MICRO REVUE

LA REVUE DE L'INFORMATIQUE PORTABLE

NOUVEAU

PPC - T EDITE "IND 41" (cf. p.120)

© 1984 Bimestriel

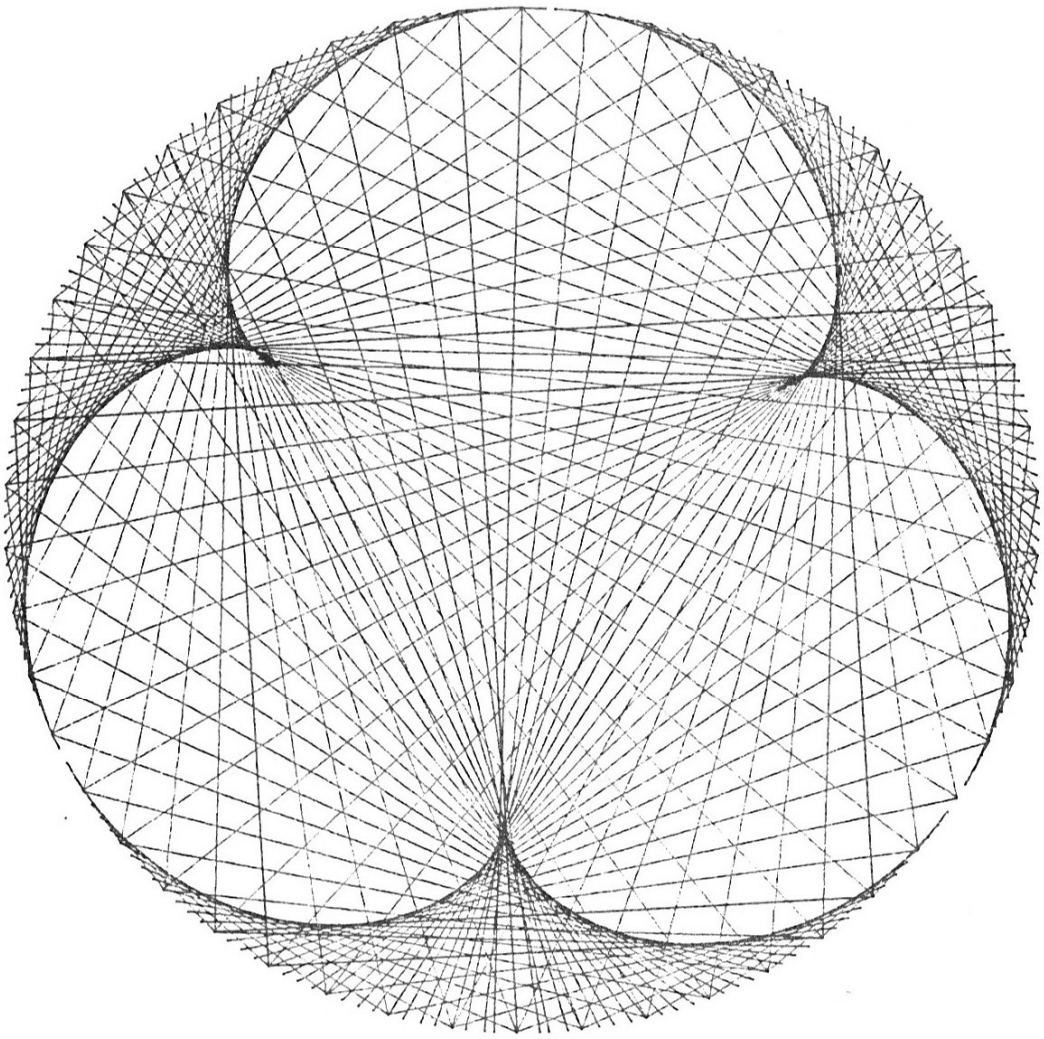
ÉDITÉ PAR PPC-T

Nº 4 JANV. - FEV. 1985

40 F

TABLE DES MATIERES

Page	Contenu
5	Editorial
	HP-41
	PROGRAMMES
8	La grande aventure
13	Dominos
16	La Grande Prophète
18	TT, traitement de texte
20	IDRE, identités remaquables
21	DIRectory
22	ARRAbes
24	VMIND, jouez au Mastermind avec des mots
	APPLICATIONS
26	Lecteur optique
27	Accéléré
29	Contre-attaque. C'est un programme qui n'est pas à sa place !
31	Messages d'erreur 41CX
32	Drapeaux, INtégration
35	Astuce
36	3 routines
37	e
38	PFA
39	RCL CAT 1
40	X-IO
45	STOd
47	PURDIR
48	Module Graphique
49	Sauts relatifs
51	Codes-barres
	MICROCODE
55	NOP
59	Divers
62	MLDL
	LE COIN DES U-PROS
70	CHIESA, calcul mécanique des lignes électriques aériennes
	HP-1x
77	Interpollation de Cornu
	HP-71
79	MEMDUMPX
81	GRAPHIQUES
	FORTH
85	STO & RCL
86	FTHUTIL
	HP-75
88	+8K
91	Au fond des ROMs
94	ROMDIR
101	CREE...SUITE
	REACTIONS
	MAGAZINE PPC-T, LE COIN DU CLUB



EDITORIAL

Chers amis,

Je vous écris ceci le 31 Décembre 84 vers 19h. C'est un bon prétexte pour vous souhaiter une très bonne année 1985. C'est aussi un bon prétexte pour faire quelques réflexions sur la relativité des choses.

Le numéro 4 de MICRO-REVUE doit être chez l'imprimeur Mercredi 2 Janvier, il sera expédié, sauf incident le Samedi 12 Janvier et vous le recevrez environ une semaine plus tard, dans presque un mois. A ce moment il sera presque temps pour moi de commencer le numéro 5. C'est pour cette raison que je pense difficile de faire un journal mensuel : il n'y a plus le temps de la réflexion.

Le présent numéro est différent des trois premiers de plusieurs façons, presque toutes regrettables. Comme je le craignais dans le dernier éditorial, il ne nous a pas été possible d'inclure un dossier dans ce numéro. Je le regrette, car je trouve les dossiers de la presse "officielle" beaucoup trop légers et je pense que nous aurions notre mot à dire. Dans ce numéro, la place du HP-71 est encore congrue. Cela n'est ni grave ni surprenant. Cet appareil est encore récent et donc peu d'entre-vous l'ont. A la cadence ou vous vous précipitez pour l'acheter (à en croire les lettres que je reçois) cela ne devrait pas durer. Mais il faut encore apprendre à l'utiliser. On annonce beaucoup de choses pour cet appareil (RAM 96K, mais à 9000F !...), il faut vous laisser le temps de réagir.

Plus gênant est la faiblesse des articles sur le FORTH. J'aurais certainement bientôt des quatités d'articles sur le FORTH/71, mais il est tellement plus puissant que les FORTH usuels qu'il est pratiquement non-compatible. Que les possesseurs d'autres machines (ils sont nombreux parmi nous, je le sais) nous envoie des articles, ils sont vivement souhaités.

Nous avons fait un gros effort pour fournir une documentation de première qualité pour les amateurs de FORTH, utilisez-la. Ne croyez pas que votre expérience est sans intérêt pour les autres, il y a toujours à gagner aux échanges. J'en profite pour vous signaler un nouveau club FORTH : contactez Georges Seibel (T638) Brassilly 74330 POISY (50) 46 21 62.

Ne déduisez pas de tout ceci que PPC-T se porte mal. Comme vous l'indique le numéro de Georges, les inscriptions continuent à arriver. Disons peut-être que l'année 84 a été dure. Souhaitons que l'année 85 soit florissante et donnons nous rendez-vous au prochain numéro...

Jean-Daniel Dodin

ADRESSE :
MICRO-REVUE, PPC-T, 77 rue du Cagire 31100 Toulouse.

EDITION :
Edité par le club de Programmation Personnelle sur Calculatrice de Toulouse (PPC-T).

REDACTION-ADMINISTRATION :
Rédacteur en chef, directeur de la publication : Jean-Daniel Dodin, Secrétaire de rédaction : Gilles Barret et Jean-Yves Pasquier, Administration : Jean-François Sibille.

VENTE ET DISTRIBUTION :
Vente en librairie et par correspondance. Diffusion par l'éditeur. De plus MICRO-REVUE est disponible à un tarif trés avantageux pour les autres clubs et associations qui le désirent, ainsi que pour les professionnels et les libraires qui souhaitent le revendre.

PARUTION :
Bimestriel, parait en début de bimestre civil (Janvier, Mars, Mai, Juillet, Septembre, Novembre), 6 numéros par an.

TARIFS (franco de port) :
Prix de vente au numéro : **40F** (50F pour envoi par avion).
Abonnement individuel : abonnement annuel **200F** (au lieu de 240F, prix de revient au numéro), 250F pour envoi par avion.

Soumission d'article :
Tous les membres du club sont invités à fournir des articles pour MICRO-REVUE. L'original à nous faire parvenir doit être tapé à la machine avec un ruban bien noir, recto seul, simple interligne, avec une largeur maximale de 15 cm.

Des articles manuscrits sont acceptés s'ils sont calligraphiés sur papier blanc quadrillé 5x5, sans sauter de ligne, recto seul, au stylo noir.

Nous faire parvenir cartes ou cassettes (qui vous seront retournée), pour les programmes, ainsi qu'un listing.

Cet original sera utilisé tel que pour la publication, soignez-le ! Ceux qui prévoient à l'avance d'écrire des articles peuvent nous demander un KIT AUTEUR qui leur donnera les modèles et toutes les indications nécessaires.

(c) 1984 REPRODUCTION INTERDITE

En dérogation aux mentions ci-dessus, la reproduction des programmes est autorisée pour l'usage personnel des lecteurs, à l'exclusion de toute reproduction commerciale.

Ont collaboré à ce numéro :

Bruno Tredez, Cyril Reigner, Pierre Renard, Jean-Pierre Reynier, Franck Lebastard, Pascal Dagornet, Constantin Sotiropoulos, Robert Schwartz, Jean Boschat, Robert Pulluart, Jean-Jacques Dhénin, J-P Magnan, René Legrand, Eric Goubault, Philippe Davase, Eric Gengoux, Janick Taillandier, Vincent Delorme, Jean-Loup Houdry, Jean Dutertre, Yves Chosson, Alain Merckx, Eric Angelini, Daniel Georgelin, Jean-Yves Pasquier, François Lafaix.

HP-41

PROGRAMMES

LA GRANDE AVENTURE

Règle du jeu

Vous êtes un aventurier téméraire "débarqué" quelque part dans une ville fortifiée abandonnée; au départ vous possédez uniquement quatre pilules de survie au pouvoir nutritif inconnu et beaucoup de bon sens (d'orientation) et de stratégie.

Pour gagner vous devez rechercher un trésor en creusant dans le champ bordant la ville, rapporter celui-çi pour l'enfermer dans le cachot, et rassembler dans un même lieu tous les objets que vous allez trouver dans la ville. Pour cela vous disposez d'une dizaine d'ordres et des quatre directions de déplacement. Mais tout n'est pas si simple ; vous devez vous nourrire pour rester en vie, et vous pouvez rencontrer quelques difficultés au cours de votre exploration. Lorsque vous transportez le sac (qui est vieux), il se peut qu'il se perce et vous perdez alors son contenu qui est redistribué au hasard dans les lieux, et le sac est recousu. Si dans le dédale vous n'avez pas la boussole, vous êtes désorienté et vous perdez du temps (et des points) en essayant de sortir au hasard. De plus vous êtes sujet à des crises d'amnésie...

Les ordres

Chaque ordre s'obtient en appuyant sur la touche correspondante et vous coute un point (sauf HALTE).

- UTILISER la machine demande QUOI?, vous devez rentrer le nom de l'objet considéré; ce nom peut être écrit en abrégé à condition qu'il n'y ai pas d'ambiguïté avec un autre mot du fichier "NOM". R/S et la machine affiche en entier le nom qu'elle prend en compte. SAC et TRESOR n'ont pas d'utilisation ici. Réponses possibles :

FLACON lorsqu'il est déja plein, il est vidé et vous ajoute 10 points. S'il est vide = REMPLIR. Il est interdit de boire le flacon lorsque vous êtes dans la fontaine ou la source.

PAIN multiplie par deux vos points, sauf si vous aviez déja au moins 100 points. Dans les deux cas le pain est perdu et un autre est placé au hasard dans un lieu.

PIC = CREUSER

BOUSSOLE lorsque vous êtes dans le dédale, vous fait ressortir et vous redonne les quatre points perdus en entrant.

GRAPPIN permet de sortir du trou ou du fossé.

CLEF ouvre ou ferme la porte du cachot dans la prison.

PILULE donne 10,15,20 ou 25 points supplémentaires. (répondre PILULE à n'importe quel ordre demandant QUOI? est équivalent à UTILISER,PILULE).

- VOIR réaffiche l'endroit où vous êtes et les objets qui s'y trouvent. Si vous êtes mort ou tombé, réaffiche la situation précédente.

- ENTRER fait passer du SOUS-TERRAIN à la GROTTE ou inversement, par un passage secret.

- LISTER montre la liste des objets possédés.

- REMPLIR remplie le flacon lorsque vous êtes dans la fontaine ou la source.

- SORTIR si vous possédez le sac et qu'il contient quelque chose, demande QUOI? , et en sort l'objet indiqué.

- RANGER si vous possédez le sac, demande QUOI? et range dedans l'objet indiqué si vous le possédez. Pour pouvoir UTILISER un objet qui est dans le sac, il faut d'abord le SORTIR.

- DEPOSER demande QUOI? et dépose l'objet indiqué dans la case où vous êtes. Il est interdit de déposer un objet dans les champs, le fossé et sur le pont-levis.

- PRENDRE demande QUOI? sauf dans les champs car seul le TRESOR peut s'y trouver, et ramasse l'objet indiqué s'il est dans la même case que vous. Vous pouvez prendre au maximum trois objets ; le contenu du sac ne compte pas.

- CREUSER lorsque vous êtes dans les champs et que vous possédez le PIC, vous permet de chercher le TRESOR caché dans une des trente cases.

- NORD,SUD,EST,OUEST déplacements. Les virages vont deux par deux et vous poseront sûrement des problèmes ; ils tournent à 45°...

- NOUVEAU DEPART lorsque vous êtes dans une situation déséspérée, cet ordre vous permet de repartir à l'aventure, comme une nouvelle partie, mais la position des objets est inchangée, et vos points sont augmentés de 50 si vous en aviez moins. N'en abusez pas, ou alors c'est trop facile...

- HALTE interromp la partie en préservant l'état du jeu (OFF seul modifie des drapeaux utilisés par le programme). Rallumez la machine pour repartir; le nombre de coups déja joués s'affiche alors, puis l'endroit où vous vous trouvez.

Chargement du programme

Ce jeu tourne sur HP-41 avec 319 registres plus module X-Fonctions (ou 41 CX) et lecteur de cassettes (il peut être adapté pour cartes magnétiques ou X-Mem, mais c'est moins pratique) ; il se compose de :

INI programme d'initialisation. Enregistrez-le avec le drapeau 11 armé.

NOM fichier ASCII de 37 registres (n'oubliez pas les espaces).

DGA 96 registres de données. Chargez "à la main" les registres 02 à 08, le programme ↑R vous aidera pour les autres : entrez un par un les codes décimaux des chaines alpha (R09 à 95),

suivit de R/S ; vous pouvez ne rien mettre si le code est 100 (ç'est déja en X).

DGA1 exemple de labyrinthe obtenu par exécution de LAB à partir de DGA. Le programme VR vous permet d'en faire la liste sur imprimante pour vérification. Vous devez obtenir la même chose si votre LAB est bon.

LAB programme de création aléatoire de labyrinthe.

GAV programme principal de jeu. Assignez "0" à STO, "1" à XEQ, "2" à CHS, "3" à RCL et "GAV" à ENTER ; enregistrez le en mode USER et sans END .

Liste du contenu des chaines alpha synthétiques en codes décimaux :

LAB 796 octets avec END

pas	code				
21	"2,3,4"	229	"17,38,60,18,39,61,19,40,62,20,		
29	"2,4,3"		41,63"		
46	"3,2,4"	270	"1"		
69	"4,2,3"	272	"127,127"		
77	"4,3,2"	281	"127,64"	INI	
217	"1"			04	"113,0,128,0"

GAV 1566 octets avec END

19	TONE 119
226	TONE 42

Il est conseillé de réenregistrer LAB et GAV sans compactage une fois qu'ils auront tournés, ainsi les branchement (GTO et XEQ) ne perdront plus de temps à rechercher le label à la première exécution.

Si vous êtes arrivés à tout rentrer sans fautes, bravo ! vous pouvez observer une minute de silence en pensant à celui qui a mis au point tout ça... mais pour les fainéants il y a la programmathèque.

Il est temps maintenant de se lancer à l'aventure.

Mise en route

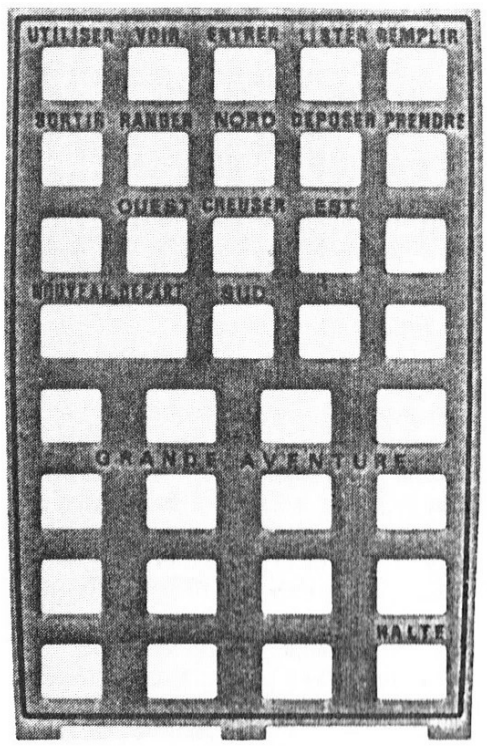
Après un MEMORY LOST , chargez le programme INI. "LAB. ?" s'affiche, si vous voulez utiliser les données déja préparées (DGA1) faites ←puis R/S, et si vous voulez créer un nouveau labyrinthe faites seulement R/S, dans ce cas LAB sera appelé, il dure trois minutes environ."ALEA ?" s'affiche, rentrez un nombre quelconque, pas trop grand et différent de kπ , puis R/S. Quand le programme GAV est lancé vous voyez "OBJETS DANS" suivit du nom d'un lieu dans lequel il faudra mettre tous les objets (sauf le trésor et les pilules) pour gagner, mais ceci n'est pas vérifié par la machine et vous pouvez les mettre dans n'importe quel lieu (sauf la prison ; devinez pourquoi). Puis la machine s'éteint, vous permettant d'enlever l'HP-IL (ou le lecteur de cartes). Préparez une feuille de papier cadrillé, un crayon et une gomme. Rallumez et c'est parti; l'endroit où vous vous trouvez est affiché; vous avez 80 points de vie au départ. Vous avez intérêt à chercher au plus vite de quoi vous nourrir et boire : un passage dans le verger ou la fontaine vous met un nombre de points décroissant avec le temps. La source vous met toujours à 25 points. Le

flacon est plein au début. Dessinez votre chemin, explorez les côtés (ILLEGAL = mur ou erreur) ; ne désespérez pas, il y a 80 cases de ville et seulement 12 lieux. Il ne peut y avoir qu'une maison (lieu) donnant sur une même case. Les virages, places et croisements ne comportent jamais de maisons. Le nombre de coups joués est affiché en fin de partie. Si vous gagnez, vous pouvez quand même continuer à jouer en faisant R/S ; "GAGNE" s'affichera tant qu'une des conditions : trésor enfermé ou tous les objets au même endroit, n'est pas annulée.

Heureux Périple

TREDEZ bruno T 120

```
01♦LBL "↑R"
02 FIX 0
03 9,096
04 STO 01
05♦LBL 00
06 VIEW 01
07 5
08 STO 00
09 CLA
10♦LBL 01
11  E2
12 PROMPT
13 XTOA
14 DSE 00
15 GTO 01
16 ASTO IND 01
17 ISG 01
18 GTO 00
19 17,037
20 STO 01
21 .END.
```




```
01♦LBL "VR"
02 CF 28
03 FIX 4
04 16 E-3
05 PRREGX
06 FIX 0
07 17,095
08 STO 00
09♦LBL 01
10 CF 29
11 CLA
12 ARCL IND 00
13 ARCL 00
14 "⊢: ""
15 SF 29
16 ,
17 POSA
18 X>0?
19 GTO 00
20 5
21 GTO 01
22♦LBL 00
23 ABS
24 XEQ 02
25 "⊢0,"
26 LASTX
27 4
28 -
29 CHS
30♦LBL 01
31 X≠0?
32 XEQ 02
33 -1
34 AROT
35 ATOX
36 "⊢""
37 PRA
38 ISG 00
39 GTO 01
40 ADV
41 STOP
42♦LBL 02
43 ATOX
44 ARCL X
45 X<>Y
46 DSE X
47 GTO 02
48 .END.
```

```
01♦LBL "INI"
02 96
03 PSIZE
04 "q♦♦"
05 ASTO d
06 CF 03
07 ΣREG 01
08 "LAB. ?"
09 ASTO X
10 STOP
11 SIGN
12 37
13 "NOM"
14 CRFLAS
15 FS?C 25
16 GETAS
17 X<>Y
18 SEEKPTA
19 "DGA"
20 X≠0?
21 ARCL X
22 READR
23 "ALEA ?"
24 TONE 3
25 PROMPT
26 STO 00
27 X<>Y
28 "GAV"
29 X=0?
30 "LAB"
31 READP
32 .END.
```

```
DGA

R00=  0,0000
R01=  17,0370
R02=  38,0590
R03=  60,0770
R04=  78,0950
R05=  13,0150
06: "3,4,2"
07: "21,42,64,22,43,65"
08: "23,44,66,24,45,67"
09: "25,100,24,0,77"
10: "25,100,24,0,94"
11: "25,100,45,0,37"
12: "25,100,45,0,94"
13: "25,100,85,0,59"
14: "25,100,67,0,37"
15: "25,100,67,0,59"
16: "25,100,85,0,77"
17: "28,100,25,101,18"
18: "28,100,17,101,19"
19: "28,100,18,101,26"
20: "28,25,100,28,101"
21: "28,100,29,101,30"
22: "28,33,100,35,101"
23: "28,100,36,101,24"
24: "28,100,23,101,37"
25: "24,100,100,20,17"
26: "24,100,19,27,100"
27: "28,26,100,31,100"
28: "24,20,100,100,29"
29: "29,100,28,33,21"
30: "28,100,21,100,31"
31: "29,27,30,100,32"
32: "28,100,31,100,0"
33: "28,29,100,22,100"
34: "24,100,100,0,35"
35: "29,22,34,100,36"
36: "29,100,35,0,23"
37: "25,100,24,0,59"
38: "28,100,47,101,39"
39: "28,100,38,101,48"
40: "28,46,100,50,101"
41: "28,48,100,53,101"
42: "28,50,100,55,101"
43: "28,51,100,57,101"
44: "28,100,56,101,57"
45: "28,100,58,101,59"
46: "24,100,100,40,47"
47: "28,100,46,100,38"
48: "29,100,39,41,49"
49: "28,100,48,100,0"
50: "28,40,100,42,100"
51: "26,52,100,43,52"
52: "26,100,51,51,53"
53: "29,41,52,100,54"
54: "28,100,53,100,0"
55: "28,42,100,56,100"
56: "29,55,100,0,44"
57: "29,43,44,0,58"
58: "28,100,57,100,45"
59: "25,100,45,0,77"
60: "27,100,101,68,100"
61: "28,68,100,72,101"
62: "24,100,101,73,69"
63: "28,100,69,101,70"
64: "28,72,100,75,101"
65: "28,100,75,101,76"
66: "28,100,76,101,67"
67: "28,100,66,101,77"
68: "28,60,100,61,100"
69: "28,100,62,100,63"
70: "28,100,63,100,71"
71: "28,100,70,100,0"
72: "28,61,100,64,100"
73: "28,62,100,74,100"
74: "28,73,100,76,100"
75: "29,64,100,0,65"
76: "29,74,65,0,66"
77: "25,100,67,0,94"
78: "28,100,89,101,0"
79: "27,100,101,100,0"
80: "28,100,87,101,88"
81: "24,100,95,91,101"
82: "28,90,100,83,101"
83: "28,82,100,92,101"
84: "28,100,93,101,85"
85: "28,100,84,101,94"
86: "29,100,95,0,92"
87: "29,100,91,90,80"
88: "26,100,80,89,89"
89: "26,88,88,100,78"
90: "28,87,100,82,100"
91: "24,81,100,100,87"
92: "29,83,86,100,93"
93: "29,100,92,0,84"
94: "25,100,85,0,37"
95: "95,14,13,14,86"
```

```
DGA1  ALEA=1,2345

R00=  0,6144
R01=  0,0000
R02=  0,0000
R03=  0,0000
R04=  0,0000
R05=  0,0000
R06=  0,0000
R07= " ≠├├0├"
R08= "σ├∆├X0"
R09=  4,0000
R10=  0,0000
R11= "*80≠δ"
R12=  17,0000
R13=  38,0590
R14=  30,0000
R15=  17,0370
R16=  4,0000
17: "28,18,100,25,12"
18: "28,19,5,17,101"
19: "28,26,6,18,101"
20: "28,101,25,100,28"
21: "28,30,100,29,101"
22: "28,101,33,100,35"
23: "28,24,100,36,101"
24: "28,37,100,23,101"
25: "24,17,100,100,20"
26: "24,100,100,19,27"
27: "28,100,26,100,31"
28: "24,29,20,100,100"
29: "29,21,100,28,33"
30: "28,31,100,21,100"
31: "29,32,27,30,100"
32: "28,93,100,31,100"
33: "28,100,29,100,22"
34: "27,35,100,100,100"
35: "29,36,22,34,100"
36: "29,23,100,35,71"
37: "25,94,100,24,77"
38: "28,47,101,39,8"
39: "28,38,101,48,4"
40: "28,100,50,101,46"
41: "28,10,53,101,48"
42: "28,100,55,3,50"
43: "28,100,57,101,51"
44: "28,56,101,57,100"
45: "28,58,101,59,100"
46: "24,100,40,47,100"
47: "28,46,100,38,100"
48: "29,39,41,49,100"
49: "28,48,100,75,100"
50: "28,100,42,100,40"
51: "26,100,43,52,52"
52: "26,51,51,53,100"
53: "29,52,100,54,41"
54: "28,53,100,76,100"
55: "28,100,56,100,42"
56: "29,100,79,44,55"
57: "29,44,78,58,43"
58: "28,57,100,45,100"
59: "25,45,94,77,100"
60: "27,68,100,11,101"
61: "28,72,101,68,7"
62: "24,73,69,100,101"
63: "28,101,70,100,69"
64: "28,75,101,72,2"
65: "28,101,76,100,75"
66: "28,1,67,100,76"
67: "28,101,77,100,66"
68: "28,61,100,60,100"
69: "28,100,63,100,62"
70: "28,100,71,100,63"
71: "28,100,36,100,70"
72: "28,64,100,61,100"
73: "28,74,100,62,100"
74: "28,76,100,73,100"
75: "29,49,65,64,100"
76: "29,54,66,74,65"
77: "25,59,37,100,67"
78: "28,100,89,101,57"
79: "27,9,101,100,56"
80: "28,100,87,101,88"
81: "24,100,95,91,101"
82: "28,90,100,83,101"
83: "28,82,100,92,101"
84: "28,100,93,101,85"
85: "28,100,84,101,94"
86: "28,100,95,100,92"
87: "29,100,91,90,80"
88: "26,100,80,89,89"
89: "26,88,88,100,78"
90: "28,87,100,82,100"
91: "24,81,100,100,87"
92: "29,83,86,100,93"
93: "29,100,92,32,84"
94: "25,100,85,37,59"
95: "95,14,13,14,86"
```

```
FICHIER NOM

00  " SAC"
01  " FLACON"
02  " PAIN"
03  " PIC"
04  "E BOUSSOLE"
05  " GRAPPIN"
06  "E CLEF"
07  " TRESOR"
08  " PILULE"
09  "E TOUR"
10  "E TERRASSE"
11  " SOUS-TERRAIN"
12  " DEDALE"
13  "E MAISON"
14  "E GALERIE"
15  "E FORGE"
16  "E TAVERNE"
17  "E CHAPELLE"
18  "E FONTAINE"
19  " VERGER"
20  "E PRISON"
21  "E GROTTE"
22  " TROU"
23  "E SOURCE"
24  " COUDE"
25  "E PLACE"
26  " VIRAGE"
27  "E IMPASSE"
28  "E RUE"
29  " CROISEMENT"
```

DOMINOS

Votre

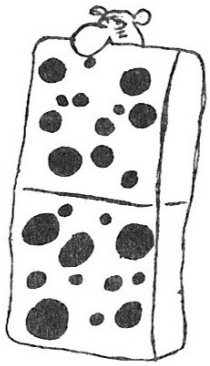
41

s'ennuie ?

Alors jouez aux dominos !

Voici une version pour HP-41 du programme de Patrick JAMET parut dans l'Ordinateur de Poche en décembre 1983.

Pour gagner aux dominos, la chance ne suffit pas. Il faut aussi une bonne stratégie. Le HP-41 le prouve: avec 704 octets de programme, il se présente comme un joueur intéressant. Pourtant, il ne possède pas la boîte de jeu. N'oubliez pas la vôtre si vous voulez vous mesurer à lui.



§ S'il est des jeux qui ne durent que le temps d'une saison, d'autres peuvent passer les siècles. C'est le cas du jeu de dominos. Aujourd'hui, il peut même être pratiqué contre un adversaire moderne: le HP-41 muni de 101 registres de programme et de 22 variables.

§ Après avoir lancé le programme, il faut introduire un à un les dominos du 41. Puis il demande "COMMENCES-TU". Si c'est «oui», il faut l'écrire en toutes lettres. Sinon, il affichera son jeu par: "MOI: X,Y" et attendra votre réponse. Si vous entrez un domino qui ne peut être posé à aucune extrémité de la chaîne, vous êtes qualifié de "TRICHEUR" et la partie s'arrête là. Tapez [A] pour continuer.

Si vous devez poser un domino qui prête à confusion, par exemple 6-2 alors que les extrémités de la chaîne sont justement 6 et 2, le 41 affichera "EXT.=", c'est-à-dire quel chiffre voulez-vous voir à l'extérieur. De même, le programme précise si ce cas se présente à lui.

Si vous devez piocher, tapez [R/S] autant de fois que vous tirez un domino lorsque c'est à vous de jouer. Si le 41 doit piocher, il l'indique par "JE PIOCHE"; il suffit de tirer pour lui un domino et de l'introduire. Si la pioche est épuisée, il indique "JE PASSE". Si la partie est bloquée le 41 s'arrête sur "PARTIE NULLE". Tapez [R/S] pour une autre partie.

Le programme obéit à une stratégie qui fait de lui un adversaire intéressant: c'est une "recherche du meilleur coup". Le HP-41 tient compte à la fois du jeu mis en mémoire et des valeurs de la chaîne qui vous ont fait piocher. Le programme possède deux limites: il ne reconnaît pas d'erreur lorsque vous introduisez deux fois un même domino en cours de partie; et bien que le 41 ne puisse pas vous empêcher de voir son jeu, évitez de le regarder. La partie en sera d'autant plus intéressante.

Je viens de me rendre compte que j'ai oublier de préciser qu'il faut rentrer les dominos sous la forme: X,Y. De plus toutes valeurs supérieures à 6,6 risque d'être mal digérée par vôtre 41.

MEMOIRES

R00 à R06: Dominos de la 41 en puissance de 10
par exemple 6,2 donnera 10^2 en R06 et 10^6 en R02
R07: Nombre de dominos restant au joueur
R08: Nombre de dominos restant au HP-41
R09: Nombre de dominos restant dans la pioche
R10 à R16: Nombre de dominos de ce numéro que possède le 41 et/ou qui sont dans le jeu (respectivement partie FRC et partie INT)
R17 & R18: Valeurs aux extrémités de la chaîne
R19 & R20: Valeurs du dernier domino joué
R21: Réference 41 pour choix du domino à jouer

FLAGS

F00: Domino double (évite le double stokage)
F01: Le HP-41 a trouvé un domino à jouer
F02: Le joueur a pioché
F03: Début de partie (1ère introduction)

LIGNES

05: Codes hexadécimauux: 21 00 80 20
arme les flags 26 31 40 50 et le 03 par ASTO d
08 à 19: initialisatisation des memoires (nombre de dominos par joueur)
20 à 25: entrées des dominos du 41
26 à 38: choix du 1er joueur
39 à 47: si le 41 commence, recherche du 1er domino à jouer puis stockage dans les memoires d'extrémités de chaîne
48 à 90: entrée du domino du joueur et stockage dans les memoires. Si aucune entrée (FC?C 22) aller piocher
91 à 97: si le 41 à un doute, entrée de la valeur extérieure du domino
98 à 100: test si le domino peut aller à un bout de la chaîne
101 à 104: reste-t-il des dominos au joueur
105 à 131: sous-programme d'entrée et stockage des dominos du 41
132 à 143: recherche du meilleur coup pour la 1ère, puis 2ème valeur d'extrémité de la chaîne si elles ne sont pas doubles
144 à 148: si le 41 pioche (FC?C 01) sinon initialisation des extrémités de chaîne
149 à 182: affichage et retrait du domino joué du jeu du 41
183 à 186: reste-t-il des dominos au 41
187 à 209: sous routine de pioche du joueur
210 à 218: sous routine de pioche du 41
219 à 234: messages de fin de partie (XROM 10,60 --> XROM "6E" du PPC-ROM; si vous ne l'avez pas mettez [STOP])
235 à 241: entrée et stockage du domino pioché par le 41
242 à 298: stratégie du programme
299 à 332: verification si le domino est valide

La stratégie du programme

■ A chaque fois que c'est son tour, le programme recherche le meilleur coup. La stratégie réside essentiellement dans la sous routine des lignes 242 à 298. Elle consiste à retenir les dominos qui peuvent être placés en bout de chaîne et choisir celui qui permettra de ne pas piocher au coup suivant mais de faire piocher l'adversaire.

■ Si aucun des dominos du 41 ne peut être placé, le RTN (ligne 261) renvoie au programme principal (pour piocher, dans ce cas). Si des dominos peuvent être retenus, le 41 analyse le coup à jouer en fonction de la valeur qui se trouverait en bout de chaîne.

■ Si cette valeur n'est posédée que par le programme (lignes 269, 270) et à condition qu'elle soit présente au moins deux fois dans son jeu (ligne 273), le coup pourra être joué sans aucun risque. Mais si cette valeur existe à l'extérieur du jeu du 41 (dans la pioche ou dans le jeu de l'adversaire), et s'il ne la possède qu'une fois (ligne 279), il jouera le coup en prenant des risques. S'il peut , il retiendra plutôt la valeur qui fera piocher l'adversaire (ligne 283).

■ Enfin, le domino retenu est introduit en R17 et R18 (lignes 285 à 298) avant le retour au programme principal.

```
LIST 097
 16:39   27.07
 01LBL "DOMINO"
 02LBL 11
"** DOMINO **" AVIEW
"!  " ASTO d PSE
CLRG 7 STO 07 STO 08
STO 09 ST+ 09  E3 /
 E + STO 17
 19LBL 00
CLA ARCL 17
"e DOMINO=" XEQ 10
ISG 17 GTO 00 CLX
STO 17 "OUI" ASTO X
"COMMENCES-TU" AON
STOP AOFF ASTO Y X=Y?
GTO A 6 E-3 STO [

 39LBL 01
XEQ 16 ISG [ GTO 01
RCL 19 STO 17 RCL 20
STO 18 GTO 06
 48LBL A
CF 27 "TOI:" CF 22
PROMPT FC?C 22 GTO 02
ABS INT STO 19 FS? 03
STO 17 LASTX FRC  E1
* STO 20 FS? 03
STO 18 LASTX ST+ Z +
ISG IND Y "" XY?
ISG IND X "" FS?C 03
GTO 04 RCL 19 RCL 20
* LASTX + RCL 19 +
RCL 17 - LASTX RCL 18
* LASTX + XY?
GTO 03 "EXT.=" PROMPT
STO 17 STO 18 GTO 04
```

```
LIST 089
 16:42   27.07
 98LBL 03
XEQ 17
100LBL 04
DSE 07 GTO B
"TU AS GAGNE" GTO 09

105LBL 10
PROMPT ABS INT STO [
LASTX FRC  E1 *
STO \ LASTX ST+ Z +
X=Y? SF 00 1,1
ST+ IND Y FC? 00
ST+ IND Z RCL \ 10^X
ST+ IND [ RCL [ 10^X
FC?C 00 ST+ IND \ RTN

132LBL B
RCL 17 STO [ RCL 18
X=Y? SF 00 XEQ 16
FS?C 00 GTO 03 RCL 18
STO [ XEQ 16
144LBL 03
FC?C 01 GTO 05 XEQ 17
CF 02
149LBL 06
CF 03 "MOI: " ARCL 19
"," ARCL 20 RCL 18
RCL 17 X=Y? " EXT. "
X=Y? ARCL X TONE 9
AVIEW PSE RCL 20 10^X
ST- IND 19 RCL 19 10^X
X=Y? SF 00 XY?
ST- IND 20 RCL 19
RCL 20  E1 ST+ Z +
LASTX 1/X ST- IND Z
FC?C 00 ST- IND Y
DSE 08 GTO A
"J'AI GAGNE" GTO 09
```

```
LIST 075
 16:44   27.07
187LBL 02
"TU PIOCHES" AVIEW
RCL 17 RCL 18 X=Y?
SF 00  E-2 ST+ IND Z
FC?C 00 ST+ IND Y
RCL 09 X=0? GTO 03
SIGN ST- 09 ST+ 07
GTO A
205LBL 03
"PIOCHE VIDE" AVIEW
SF 02 GTO B
210LBL 05
RCL 09 X0? GTO 03
"JE PASSE" AVIEW PSE
FC?C 02 GTO A PSE
"PARTIE NULLE"
221LBL 09
AVIEW BEEP STOP
"UNE AUTRE ?" AON STOP
AOFF ASTO X "OUI"
ASTO Y X=Y? GTO 11
XROM "GE"
235LBL 03
"JE PIOCHE:" XEQ 10
SIGN ST- 09 ST+ 08
GTO B
242LBL 16
, STO 21 6 E-3 STO \
RCL IND [ STO a
249LBL 07
RCL a INT  E1 /
STO a FRC X0? GTO 03

258LBL 08
ISG \ GTO 07 RTN
```

```
LIST 061
 16:47   27.07
262LBL 03
RCL \  E1 + 7
RCL IND Y STO ] X<Y?
GTO 03 FRC ,2 X<=Y?
ST+ ] GTO 04
276LBL 03
FRC ,1 X=Y? ST+ ]
RCL IND \ FRC X0?
ST+ ]
285LBL 04
RCL 21 RCL ] X<=Y?
GTO 08 STO 21 RCL [
INT STO 19 RCL \ INT
STO 20 SF 01 GTO 08

299LBL 17
RCL 17 RCL 19 XY?
GTO 03 RCL 20 STO 17
RTN
307LBL 03
RCL 18 XY? GTO 03
RCL 20 STO 18 RTN

314LBL 03
RCL 20 XY? GTO 03
RCL 19 STO 18 RTN

321LBL 03
R^
```

```
LIST 011
 16:49   27.07
XY? GTO 03 RCL 19
STO 17 RTN
328LBL 03
"TRICHEUR..." SF 27
GTO 09 .END.

*** LIGNES ALPHA ***
-=-=-=-=-=-=-=-=-=-=

22, 153 & 158 SONT PRE-
CEDEES DE "APPEND"
71 & 74 SONT "F0"
TOUS LES "XY?" OU "X0?" SONT DES
"X≠Y?" OU "X≠0?"



**********
*BONNE CHANCE*
**********
```

LA GRANDE PROPHETE

Ne vous ruinez plus en achat d'horoscope ou en visite de "diseuses de bonne aventure", la Quarantéhunus, la plus grande prêtresse de tous les temps peut vous aider. Voyez.....

N'ATTENDEZ PLUS pour lancer ce programme basé sur les dernières estimations statistiques de mortalité.

Répondez aux questions en toute sincérité. Pour les réponses numériques (unités conventionelles: m, kg) entrez vos valeurs par le pavé numérique suivies de [R/S]. Pour les autres réponses la HP ne quitte pas le mode PRGM et il suffit d'appuyer sur la touche (F, M, N ou O) donnant la réponse. La 41 vous signale quand c'est-à-vous par un TONE 73, ou qu'elle a bien reçu votre réponse par un TONE 89.

Après vous avoir donné votre espérance de vie elle vous indique quelques conseils au besoin.

 # MEMOIRES #

R00: Age de l'individu
R01: Est-il solitaire ?
R02: Nbre de cigarettes
R03: Heures de sommeil
R04: Check-up annuel ?
R05: Calculs divers
R06: Sportif ?
R07: Rapport poids-taille
R08: Nbre de verres
R09: Années possibles

 # EXTENSIONS #

Time et X-Functions

 # FLAGS #

F05: UP~ réponse N (non)
DOWN~ autre réponse
F06: UP~ Féminin
DOWN~ Masculin

La 41 baisse les flags 05 et 06 après usage, et l'affichage est en FIX 2 CF 29. Les autres flags sont inchangés

 # LIGNES #

TONEs 63 L.44, 102
73 L.7, 26, 34, 161, 173, 191
89 L.138

ALPHA non précédés de "APPEND"
2, 6, 12, 25, 33, 42, 47, 54, 65, 74, 83, 90, 109, 114, 122, 130, 144, 157, 169, 181, 229, 235, 244, 248, 251, 260, 269, 277, 285, 294, et 314

```
LIST 129
 18:54   01.08
 01LBL "CHECKUP"
"*LONGEVITE*" AVIEW
PSE CLRG "AGE" TONE 3
PROMPT ABS INT STO 00
"SEXE" XEQ 15 CF 05
CF 06 20 - X=Y?
GTO 00 3 SF 06 ST- 05

 23LBL 00
8 "TAILLE" TONE 3
PROMPT ABS  E - 110
* "POIDS" TONE 3
PROMPT X<>Y /  E1 *
- STO 07
"VOUS ETES ..." AVIEW
TONE 3 PSE PSE
"-CITADIN" FC? 06 "E"
XEQ 15 2 FC?C 05
ST- 05 "-SEUL" FC? 06
"E" XEQ 15 FS?C 05
GTO 00 RCL 05 4 -
STO 01
 64LBL 00
"-ACTI" FC? 06 "VE"
FS? 06 "F" XEQ 15 2
FC?C 05 ST+ 05
"-SPORTI" FC? 06 "VE"
FS? 06 "F" XEQ 15 3
FC?C 05 STO 06
"-SATISFAIT" FC? 06
"E" " EN AMOUR"
XEQ 15 2 FC?C 05
ST+ 05 "-ANXIEU"
FC? 06 "SE" FS?C 06
"X" XEQ 15 3 FC?C 05
ST+ 05 "AVEZ VOUS ..."
AVIEW TONE 3 PSE PSE
"FAIT DES ETUDES" " S"
AVIEW "UPERIEURES"
XEQ 15 2 FC?C 05
ST+ 05
"UNE PROFESSION " "IN"
AVIEW "DEPENDANTE"
XEQ 15 3 FC?C 05
ST+ 05
"FAIT UN BILAN D"
"E SA" AVIEW
"NTE ANNUEL" XEQ 15 2
FC?C 05 STO 04 GTO 00
```

```
LIST 124
 18:56   01.08
130LBL 15
" ?" AVIEW TONE H

134LBL 02
GETKEY X=0? GTO 02
TONE 9 41 X=Y? SF 05
RTN
143LBL 00
"NOMBRE DE CIGAR"
"ETTE" AVIEW
"S PAR JOUR ?" TONE 3
PROMPT ABS  E1 / 3
Y^X SQRT STO 02
"NOMBRE DE VERRE"
"S D'ALCOO" AVIEW
"L PAR JOUR ?" TONE 3
PROMPT ABS 2 / LASTX
- STO 08
"NOMBRE D'HEURES"
" DE SOMME" AVIEW
"IL PAR NUIT?" TONE 3
PROMPT ABS 2 / 4 -
STO 03
"COMBIEN AVEZ-VO"
"US D" AVIEW
"E GRANDS-PAR" AVIEW
"ENTS DECEDES" AVIEW
" APRES L'AGE" AVIEW
" DE 80 ANS ?" TONE 3
PROMPT 70 RCL 00 ,24
* + + RCL 05 +
RCL 07 ABS RCL 02 +
RCL 08 ABS + RCL 03
ABS + RCL 01 -
RCL 06 - RCL 04 - -
STO 09
"ESPERANCE DE VI" "E"
AVIEW PSE FIX 0 CF 29
"** " ARCL 09
" ANS**" BEEP AVIEW
PSE PSE  E LASTX
"S'" X<=Y? GTO 01 +
"VOUS POURRIEZ A"
"VOIR " AVIEW
"UN BONUS DE " AVIEW
ADATE " ANS SI ..."
AVIEW PSE CHS RCL 07
X>Y? GTO 00
"VOUS PERDIEZ DU"
" POIDS" AVIEW
```

```
LIST 092
 18:59   01.08
PSE
255LBL 00
5 RCL 02 X<Y? GTO 00
"VOUS CESSIEZ DE"
" FUMER" AVIEW PSE

264LBL 00
1,5 RCL 08 X<Y?
GTO 00
"VOUS BUVIEZ MOI" "NS"
AVIEW PSE
273LBL 00
RCL 06 X0? GTO 00
"VOUS FAISIEZ DU"
" SPORT" AVIEW PSE

281LBL 00
RCL 04 X0? GTO 00
"VOUS VOUS FAISI" "E"
AVIEW "Z EXAMINER R"
AVIEW "EGULIEREMENT"
AVIEW PSE
293LBL 00
"VOUS DORMIEZ " 1,4
CHS RCL 03 X>Y?
GTO 00 "PLUS" AVIEW
PSE
303LBL 00
,9 X>Y? GTO 00
"MOINS" AVIEW PSE

310LBL 00
RCL 01 X=0? GTO 00
"VOUS ROMPIEZ VO" AVIEW
"TRE SOLITUDE" AVIEW
PSE
319LBL 00
CLA
321LBL 01
"IL NE VOUS ARR"
"IVE AU" AVIEW
"CUN ACCIDENT" AVIEW
" FACHEUX AVA" AVIEW
"NT L'AN " DATE  E2
* FRC  E4 * RCL 00
- RCL 09 INT +
ARCL X AVIEW PSE
FIX 2 CLOCK
```

1167 OCTETS
167 REGISTRES

SIZE 010

REIGNER
Cyril
T 282

et sa HP-41CV !

CE PROGRAMME, EN DEUX PARTIES, EST UN MINI TRAITEMENT DE TEXTE PERMETTANT D'ECRIRE UN FICHIER EN MODE EDITEUR, AVEC QUELQUES POSSIBILITES EN PLUS, PUIS DE L'IMPRIMER JUSTIFIE A DROITE ET A GAUCHE.

IL NECESSITE UNE HP-41 CX (POUR "TEX"), LE MODULE IL DEVELOPMENT ET UNE IMPRIMANTE 24 COLONNES (POUR "TT").

CHAQUE ENREGISTREMENT DU FICHIER FORME UN PARAGRAPHE EN TETE DUQUEL SONT AUTOMATIQUEMENT PLACES DEUX ESPACES.

LES ESPACES ENTRE LES MOTS SONT AJUSTES; MAIS LE DERNIER MOT EST COUPE SI CES INTERVALLES SERAIENT TROP GRANDS; CE QUI ARRIVE ASSEZ SOUVENT AVEC 24 CARACTERES SEULEMENT PAR LIGNE !

```
01♦LBL "TEX"
02♦LBL I
03 "α♦♦"
04 ASTO d
05♦LBL 00
06 "TEXTE"
07 AON
08 PROMPT
09 FC? 48
10 EMDIR
11 FC? 48
12 GTO 00
13 FC?C 03
14 RTN
15 EMROOM
16 CRFLAS
17 FS?C 25
18 GTO 01
19 FLSIZE
20 +
21 RESZFL
22♦LBL 01
23 CLA
24 AON
25 ED
26 AOFF
27 CF 21
28 CF 22
29 "CODE"
30 PROMPT
31 FS? 48
32 GTO 04
33 FC? 22
34 GTO 02
35 CLA
36 XTOA
37♦LBL 04
38 INSCHR
39 GTO 01
40♦LBL 02
41 CLX
42 SEEKPT
43 SF 25
44♦LBL 03
45 ASHF
46 FC? 17
47 " "
48 ARCLREC
49 FC? 25
50 GTO 01
51 AVIEW
52 ASHF
53 ATOX
54 GTO 03
55♦LBL C
56 XEQ F
57 XEQ 00
58 SF 25
59 CREATE
60 FC?C 25
61 PROMPT
62 AOFF
63 SAVEAS
64 BEEP
65 GTO I
66♦LBL E
67 XEQ 00
68 PURFL
69 GTO I
70♦LBL A
71 XEQ 00
72 GETAS
73 GTO 01
74♦LBL F
75 CLA
76 FLSIZE
77 ASROOM
78 7
79 /
80 INT
81 -
82 RESZFL
83 END
```

(Je continue à la machine à écrire, c'est plus pratique !)

L'adaptation pour un nombre supérieur de caractères (et une autre imprimante) demandera de jongler beaucoup avec les pointeurs, alpha, et le buffer, et sûrement de créer un fichier de traitement secondaire.

mode d'emploi de TEX

SIZE 000 163 octets

Lorsque "TEXTE" s'affiche, entrez le nom d'un fichier (s'il est différent de TEXTE) puis R/S. Si vous sortez de alpha avant, vous obtiendrez un EMDIR. Le drapeau 3 armé indique qu'un fichier de taille maximale est créé au nom indiqué s'il n'existait pas, puis vous passez en mode EDiteur; faites ON pour acceder aux autres possibilités. "CODE" s'affiche, vous pouvez :

-entrer un numéro de code (1 à 255), le caractère correspondant sera inséré.

-mettre en alpha ce que vous voulez insérer,et restez en mode alpha.

-ne rien introduire, vous obtiendrez le défilement du contenu du fichier. Si vous faites SF 21 avant, il y aura arret à chaque AVIEW ou bien impression si l'imprimante est branchée.

-comme vous êtes en USER vous pouvez appuyer :

A : appel d'un fichier depuis une cassette

C : chargement sur cassette du fichier courant (F est exécuté avant). Si un fichier du nom entré existe déja, le programme s'arrête (pas 61); confirmez son remplacement par R/S ou refaites C. Un bip annonce la fin du chargement

E : efface un fichier en X-Mémoire

```
01♦LBL "TT"
02 28
03 BSIZEX
04 AIPT
05♦LBL 00
06 EMDIR
07 "α'♦♦"
08 RCL I
09 "TEXTE"
10 AON
11 PROMPT
12 STO d
13 X=Y?
14 SF 13
15 CLX
16 SEEKPTA
17 FC?C 25
18 GTO 00
19♦LBL 01
20 CLA
21 FC? 17
22 ""
23 SF 25
24 ARCLREC
25 FC?C 25
26 GTO 10
27 FC? 17
28 GTO 06
29 ATOX
30 ARCLREC
31 32
32 A-XR
33 X=Y?
34 SF 05
35 CLX
36 X<> Z
37 X-AL
38 FS?C 05
39 GTO 06
40 A-BUF
41 CLX
42 PT=
43 X<>Y
44♦LBL 02
45 POSA
46 X<0?
47 GTO 03
48  E
```

F : fin; compacte le fichier courant et affiche sa taille en registres. Faites R/S si vous voulez continuer

I : initialisation, pour changer de fichier ou en faire un autre. Pour créer un nouveau fichier il faut de la place en X-M; C,E et F en libèrent.

pas

3 hexa 71:00:80:0arme les drapeaux 25,26, 27,31 et 40

5 à 12 demande du nom du fichier concerné

15 à 21 création du fichier

22 à 30 mode éditeur et demande des ordres

35 à 39 insertion de caractères spéciaux

40 à 54 boucle d'affichage du fichier

55 à 65 sauvegarde sur cassette

66 à 69 efface un fichier

70 à 73 rappel de fichier d'une cassette

74 à 83 compactage du fichier

mode d'emploi de TT

SIZE autant qu'il peut y avoir d'espaces par ligne (006 doit suffire), 4 registres pour le buffer , 199 octets

Lancez le programme imprimante branchée; EMDIR s'exécute, entrez le nom du fichier à traiter; faites ENTER hors du mode alpha pour imprimer tout en minuscules puis R/S; si le fichier nomé n'existe pas il y a retour au EMDIR.

Je n'ai pas trouvé de moyen simple de conserver certaines lettres préalablement choisies, en majuscules.

Vitesse moyenne de traitement + impression : 4 caractères par seconde.

pas

7 hexa 04:61:00:80:00 arme les drapeaux 21,25,26,31 et 40

11 demande du nom du fichier

22 dec 160,160 deux espaces non modifiables

26 fin du fichier

28 fin de paragraphe

29 à 34 rappel le caractère suivant la ligne et teste s'il s'agit d'un espace

35 à 39 rétablit la ligne et l'imprime si elle est suivie d'un espace

44 à 56 recherche et stocke la position et le nombre d'espaces

55 NOP dec 240

57 à 66 routine de coupure sur un mot. Peut être supprimée ainsi que les pas 75 et 76 si on ne veut jamais couper les mots

67 à 69 accumulation d'une ligne complète

70 à 84 calcul de la dimension des espaces

85 à 100 accumulation des mots et des espaces

101 à 110 dernier mot et retour ligne suivante

111 à 115 fin (on peut ajouter CLA,CF 13)

TREDEZ bruno T 120

```
49 +
50 STO IND Z
51 AROT
52 +
53 A-XR
54 ISG Z
55 ""
56 GTO 02
57♦LBL 05
58 CLA
59 23
60 BUF-AX
61 "├-"
62 RCLPT
63 2 E-3
64 PT=
65 -
66 SEEKPT
67♦LBL 06
68 PRA
69 GTO 01
70♦LBL 03
71 ST+ Z
72 *
73 25
74 +
75 X>Y?
76 GTO 05
77 ,1
78 %
79 RCLPT
80 -
81 SEEKPT
82 CLX
83 7
84 *
85♦LBL 04
86 RCL IND a
87 CLA
88 BUF-AX
89 ACA
90 SIGN
91 ST+ a
92 X<>Y
93 STO Y
94 RCL Z
95 /
96 RND
97 SKPCOL
98 -
99 DSE Y
100 GTO 04
101 SIGN
102 RCL IND a
103 -
104 CLA
105 BUF-AX
106 CLX
107 STO a
108 PT=
109 ACA
110 GTO 01
111♦LBL 10
112 ADV
113 CLX
114 BSIZEX
115 END
```

I DRE

OBJET: programme developpant les identitées remarquables du type (a±b)↑ⁿ

nom: "IDRE" taille: 21 regs size: 002

extension(s): aucune

MODE D'EMPLOI

1- Une fois le programme en RAM, le lancer.

2- La 41 vous affiche alors "(a?b)↑" en mode ALPHA avec l'indicateur SHIFT allumé. Taper le signe (+ ou -) puis [R/S].

3- La HP affichera "(a#b)↑" (note: # est mis pour le signe que vous avez entré). Entrer la puissance N (N<=69) puis [R/S].

4- La 41 "TONE" et affiche un premier groupe de caractères. Elle "TONE" pour chaque groupe suivant.

5- Lorsqu'un "BEEP" retenti, alors le resultat à l'affichage est le dernier groupe de l'indentité. [R/S] fait retour au §2-.

02- codes hexadecimaux:
    f4 21 00 81 80 40
    arme les FlAgs 26-31-0-47&48
08- TONE 89 (20 ms)
19- idem ci-dessus
65- TONE 16 (990 ms)
77- X0? correspond à X≠0?

EQUATION

$(a\pm b)^N = C_N^M . a^{N-M} . b^M$

MEMOIRES

R00: M (varie de 0 à N)
R01: N puissance de l'identité

FLAGS

F00: armé si identité positive
F05: N est superieur au 3ème degré

```
 01↓LBL "IDRE"
"!↓"  RCL I  STO d
CLX  STO 00  "(a?b)↑"
TONE 9  STOP  AOFF
ASTO X  "+"  ASTO Y
X=Y?  SF 00  "(a"
ARCL X  "⊢b)↑"  TONE 9
PROMPT  STO 01  ARCL X
AVIEW  3  X<Y?  SF 05
CLA

 28↓LBL A
RCL 01  FACT  LASTX
RCL 00  FACT  X<>Y
LASTX  -  STO T  FACT  *
/  2  X<=Y?  ARCL Y  R↑
X=0?  GTO 00  "⊢a"  E
X=Y?  GTO 00  "⊢↑"
ARCL Y

 53↓LBL 00
RCL 00  X=0?  GTO 00
"⊢b"  E  X=Y?  GTO 00
"⊢↑"  ARCL Y

 63↓LBL 00
AVIEW  TONE 6  PSE
FS? 05  PSE  E  ST+ 00
"+"  FS? 00  GTO 00
RCL 00  2  MOD  X≠0?
"-"

 79↓LBL 00
RCL 01  RCL 00  X<=Y?
GTO A  BEEP  .END.
```

DIR

EXEMPLE DE LISTING DU PROGRAMME "DIR":

```
Etat d'écriture                                    ***41***
du 05.07.1984              Cassette:  ** MASS ST 2- **
à 11:01:16.39
      Initialisée à 199 fichiers & 485 enregistrements

nom et type du fichier     piste/ enr.     octets- regs
TEX       ,prgm ,S            0 /  27        1566-  224
DIR       ,prgm ,S            0 /  34         765-  109

Reste:  197 labels disponibles
```

OBJET: Ce programme permet d'imprimer le Directory sur une imprimante 80 colonnes interfacée HP-IL par le Converter HP 82166.

nom: "DIR" taille: 110 regs size: 007

extension(s): X-Functions, HP-IL Dev, HP-IL, Cassette, Iprimante

MODE D'EMPLOI

1- Chargez le prgm, branchez la boucle HP-IL et mettez tous les périphériques voulus avec l'imprimante et le cassette. Il n'est pas nécessaire de faire attention aux adresses, la 41 s'en charge; ATTENTION la 41 envoie un ordre Interface Clear qui remet à zéro tout l'HP-IL.

2- Lancer le prgm, la 41 demande après quelques secondes: "NOM CASSETTE". Entrez le nom de la cassette qui est dans le lecteur puis [R/S]. La HP imprime le Directory (≈5s. par ligne) dans une feuille 21*29,7 Cm. Si le Directory fait plus de 60 f€ P2éVoir plusieurs feuilles car l'imprimante après avoir sautée au-dessus des perforations s'arrêtera par manque de papier. La 41 affichera "TRANSMIT ERR"; faire [PRGM], [SST], [PRGM], rechargez l'imprimante en papier puis [R/S].

3- Lorsqu'elle a finie la 41 rembobine la cassette et sortir la feuille aprè un "bip", puis s'arrête.

LIGNES

52- Codes hexadécimaux:
F5 44 21 86 80 01
arme les flags 21, 17, 31, 32, 37, 38, 40 & 55

58 & 59-Definissent le 1er jeu de l'imprimante en normal pica, espacement proportionnel, page logique de 66 lignes, page texte de 60 lignes et saut par dessus les perforations.

MEMOIRES

00: adresse du lecteur de cassette
02: adresse du Converter
02: (valeur du Newm*8)-1
03: compteur d'enregistrement du Directory
04: compteur de fichier (0 à 7) par enregistrement
05: compteur nombre de fichiers réels
06: sauvergarde des flags d'origine (RCLFLAG)
Buffer de 259 octets ou de 37 registres

LIGNES

08: UP- le fichier est un programme
DOWN- fichier de données (ascii, status, etc...)

REIGNER C. T282

ARRA

OBJET: programme permettant la conversion des chiffres arabes en chiffres romains et vice-versa.

NOM: "ARRA" TAILLE: 55 regs SIZE: 002
EXTENSION(S): X-FUNCTIONS & PPC-ROM (facultatif).

MODE D'EMPLOI

1- Charger le programme le verifier et le lancer.

2- La 41 affiche "0: R-A 1:A-R". Pour une conversion Romain-Arabe taper zero; pour la conversion inverse taper un, et [R/S] pour valider votre choix.

3- A la demande "Nbe: ?", entrer le nombre voulu en romain (machine en mode ALPHA) ou arabe (mode RUN) puis [R/S].

4- Après quelques secondes la HP-41C/CV affiche le resultat. [R/S] fait retour au §3- toujours pour le même type de conversions. Pour changer de type de conversion faire [RTN] [R/S] et aller au §2-.

MEMOIRES

R00: nombre arabe
R01: pointeur: 4 à 0 pour la conversio arabe-romain
24 à 0 pour la conversion romain-arabe

FLAGS:

F00: usage interne
F01: calcul des unités
F02: calcul des dizaines
F03: calcul des centaines
F04: calcul des milliers

VALEURS ENTREES

Conversion arabe-romain: si X=0 -> "NULL" en ALPHA
X<0 -> |X|
X non entier -> E(X)
Conversion romain-arabe: si les lettres entrées sont differentes de: I, V,X,L,C,D ou M -> "NONEXISTENT"
Il faut faire alors [XEQ] 00.

NOTE 1:Si vous n'avez pas de module PPC-ROM remplacez les lignes 167 à 170 par "Nbe: ?".

NOTE 2: Toutes les valeurs alphanumerique du programme des lignes 132 (LBL 03) à 163 (RTN) sont precedés de APPEND (code hexadecimal 7F). Idem pour le "M" de la ligne 35.

REIGNER C. T282

```
§
↳PRP "DIR"§
↳ 11:06   05.07§
↳§
 01§LBL "DIR"§
↳SIZE? 7 X>Y? PSIZE §
↳RCLFLAG STO 06 CLX §
↳STO 00 STO 01 STO 05§
↳SF33 AAU  E AAD - §
↳CHS STO [ §
§
↳ 19§LBL 01§
↳RCL [ TAD SAI 64 §
↳X↳Y? GTO 00 RCL [ §
↳STO 01 RDN §
§
↳ 29§LBL 00§
↳4 / X↳Y? GTO 00 §
↳RCL [ STO 00 §
§
↳ 36§LBL 00§
↳DSE [ GTO 01 20 CMD§
↳UNT 256 BSIZEX AIPT§
↳"NOM CASSETTE" CF 21 §
↳AVIEW CLA AON STOP §
↳A-BUF "D!  §" RCL [ §
↳STO d RCL 01 SELECT §
↳PT? "§(s9h1P§&l66p" §
↳"■60f1L" OUTA §
↳"Etat d' criture" §
↳"■§&a43C###" OUTA §
↳"41###|↳du " DATE §
↳ADATE OUTA §
↳"§&a24CCassette:" "■  "§
↳OUTA CLA CLX PT= §
↳X<>Y BUF-AX OUTA §
↳"|↳  " TIME ATIME24 §
↳OUTA SF33 FIX 0 §
↳RCL 00 TAD CLX PT= §
↳64 WREG 2 DDT / §
↳INBUFX 19 PT= UNT §
↳RCL 01 LAD SIGN §
↳BUF-XB CF33 STO 03 B§
↳*  E - STO 02 §
↳"§&a5C↳Initialis" §
↳"■ e    " ARCL X OUTA§
↳" fichiers & " 510 §
↳RCL 03 - ARCL X OUTA§
↳" enregistrement" §
↳"■s|↳↳nom e" OUTA §
↳"t type du fichi" §
↳"■er    pis" OUTA §
↳"te/ enr.    oct" §
↳"■ets- regs" CF 17 §
↳OUTA SF33 RCL 00 LAD§
↳4 DDL , OUTBIN 2 §

↳OUTBIN RCL 00 UNL TAD§
↳CLX PT= 64 WREG 2 §
↳DDT X^2 * INBUFX §
§
↳§
§
↳148§LBL b§
↳7 E-3 STO 04 UNT §
§
↳§
§
↳152§LBL a§
↳RCL 01 LAD RCL 04 32§
↳* PT= 8 CLA BUF-AX§
↳X<>Y 11 + PT=  E §
↳BUF-XB X=0? GTO 16 §
↳X=Y? "■,ascii " R^ §
↳X^2 - X=0? "■,all "§
↳16 - X=0? "■,keys "§
↳LASTX - X=0? §
↳"■,stat. " 32 - X=0?§
↳"■,prgm " X=0? SF 08§
↳80 - X=0? "■,data "§
↳47 - X=0? GTO 17 R^§
↳19 + PT= SIGN BUF-XB§
↳X<>F FS? 00 "■,P" §
↳FS? 01 "■,A" FS? 03 §
↳"■,S" LASTX 16 - PT=§
↳SIGN BUF-XB CF33 ACA§
↳"§&a27C" ARCL X §
↳"■ /  " X<>Y BUF-XB §
↳ARCL X LASTX 14 + §
↳"■§&a41C" PT= 2 §
↳BUF-XB RCL X 7 FC? 08§
↳1/X / FC?C 08 X<>Y §
↳ARCL Y ACA "-§&a47C"§
↳ARCL X OUTA SF33 §
↳ISG 05 §
§
↳247§LBL 16§
↳ISG 04 GTO a RCL 00 §
↳LAD 3 DDL CLX OUTBIN§
↳UNL RCL 00 TAD CLX §
↳PT= 64 WREG CLX DDT§
↳256 INBUFX DSE 03 §
↳GTO b §
§
↳269§LBL 17§
↳RCL 00 LAD 7 DDL UNL§
↳CF33 "↳Reste: " ACA §
↳RCL 02 RCL 05 - ACX§
↳" label"  E X<Y? "■s"§
↳"■ disponible" X<Y? §
↳"■s" "■0↳" OUTA §
↳RCL 06 STOFLAG END §
```

```
                    PRP ""
 4:27PM 12/29

 01◆LBL "ARRA"
FIX 0  CF 29  .  X<>F
"0: R-A  1: A-R"  PROMPT
X=0?  GTO 00

 10◆LBL A
4  STO 01  "Nbe: ?"
PROMPT  CLA  ABS  INT
 E3  /  INT  X>0?  SF 04
LASTX  RND  LASTX  STO Z
-  X>0?  XEQ 13  LASTX
FC?C 04  GTO 15  INT
RCL b  "⊢M"  DSE Y
STO b  LASTX  ENTER↑
GTO 12

 41◆LBL a
RCL 01  2  X≠Y?  SF 00
RCL Z  4.9  X=Y?  GTO 14
5  +  X<>Y  X<=Y?
FS?C 00  FS? 30  GTO 15


 57◆LBL 14
CF 00  RND  LASTX  STO Z
-  X>0?  XEQ 13  LASTX


 66◆LBL 15
ENTER↑  INT  X=0?
GTO 12  5  MOD  4  X=Y?
SF 00  RDN  FS?C 00
SIGN  SF IND 01
XEQ IND Y  CF IND 01


 82◆LBL 12
RDN  FRC  E1  *  DSE 01
GTO a  PROMPT  GTO A


 91◆LBL 13
SF 01  RCL 01  E  -
10↑X  *  E1  X<>Y
X<=Y?  GTO 14  X<> Z
STO L  CF 01  RTN


106◆LBL 14
STO Y  5  MOD  4  X=Y?
SF 00  RDN  FS?C 00
SIGN  FS?C 04  SF 00
XEQ IND Y  FS?C 00
SF 04  RCL Z  RND  STO L
CF 01  RTN

126◆LBL 08

127◆LBL 07

128◆LBL 06
XEQ 05

130◆LBL 01

131◆LBL 02

132◆LBL 03
FS? 01  "⊢I"  FS? 02
"⊢X"  FS? 03  "⊢C"
FS? 04  "⊢M"  DSE X
GTO 03  RTN

144◆LBL 04
XEQ 03

146◆LBL 05
FS? 01  "⊢Y"  FS? 02
"⊢L"  FS? 03  "⊢D"  RTN


154◆LBL 09
XEQ 03

156◆LBL 10
FS? 01  "⊢X"  FS? 02
"⊢C"  FS? 03  "⊢M"  RTN


164◆LBL 00
.  STO 00  45  "Nbe: _?"
AVIEW  XROM "IF"  AON
STOP  AOFF  ALENG  E  -
STO 01  X=0?  GTO 11
ATOX  XEQ IND X

182◆LBL 11
ATOX  RDN  XEQ IND T
X<>Y  X<Y?  CHS  ST+ 00
RDN  DSE 01  GTO 11
ST+ 00  VIEW 00  STOP
GTO 00

197◆LBL 73
 E  RTN

200◆LBL 86
5  RTN

203◆LBL 88
 E1  RTN

206◆LBL 76
50  RTN

209◆LBL 67
 E2  RTN

212◆LBL 68
5 E2  RTN

215◆LBL 77
 E3  .END.
```

RENARD PIERRE/T 440
BORSBEEKBRUG 3C Bus 27
2600 BERCHEM-ANT
BELGIQUE

Berchem, le 13 JUILLET 1984.

Cher Jean-Daniel,

Avant tout, félicitations pour le nouveau format de la revue; MICRO-REVUE est maintenant devenue une revue adulte, de même que par son contenu. Un seul petit regret : l'interface vidéo me semble le parent pauvre de l'ensemble des périphériques. Pour y remédier, je vous propose cette fois un programme fonctionnant uniquement avec cet interface sur la boucle HP-IL. J'ai encore quelques autres programmes dans mes cartons que je me propose de vous envoyer sur un KIT AUTEUR que vous aurez la gentillesse de m'envoyer. Merci.

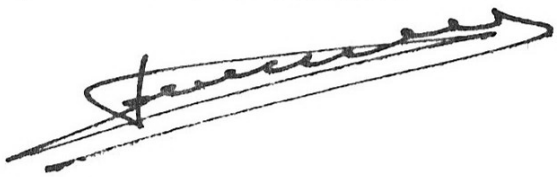
Ce programme permet de jouer au SUPER MASTER MIND avec votre écran TV. Je n'ai pas cherché à optimiser le programme avec des séquences d'échappement utilisant l'imprimante.

Le jeu se joue avec 5 positions et 8 couleurs, en l'occurence les chiffres 1 à 8. Ces données peuvent être augmentées ou diminuées en changeant les pas 17 - 20 - 27 ; Après la mise en page de l'écran, la machine affiche un x? vous demandant un nombre aléatoire. R/S et la machine affiche ESSAI? Entrer un nombre de 5 chiffres (dans ce cas çi)et la machine (l'écran) vous donnera au bout de ± 14 sec. le nombre de chiffres NOIR(correct et à sa place) et BLANC(correct mais non à sa place). Le but du jeu est évidemment de trouver la combinaison correcte des 5 chiffres. Le programme est conçu pour donner 5 chiffres différents (pas 37 à 60); si vous voulez augmentez la difficulté en mettant plusieures fois 1 ou plusieurs chiffres identiques, supprimer simplement les pas 37 à 61. Vous avez 12 essais pour trouver la soulution,sinon la machine vous affiche cette solution.

Autre chose : si vous tapez un nombre de plus de 5 chiffres, ce nombre sera présent à l'écran, mais la machine ne traitera que les 5 premiers chiffres. Si vous ne possédez pas de modules X-Fonctions ignorez les pas 3 et 4, mais prévoyez une SIZE de 017. Vous verrez à l'usage que ce programme est très simple de compréhension mais qu'il l'est moins dans la recherche de la solution.

XEQ"VMIND" et bon amusement à tous.

P.S:le pas 165 est composé de 15"space". RENARD PIERRE.



V MIND

```
 01♦LBL "VMIND"
CLRG  17  PSIZE  SF 17
"     * SUPER MA"  OUTA
"STER MIND *"  CF 17
OUTA  CLRG  CF 29  "X?"
PROMPT  STO 01  FIX 0  5
STO 00  STO 11  8
STO 09
"5 POSITIONS/8 C"
"├OULEURS"  OUTA  XEQ 00

 26♦LBL 00
8  RCL 01  9821  *
.211327  +  FRC  STO 01
*  INT  STO 06  RCL 01
X<>Y  X=Y?  XEQ 00
RCL 02  RCL 06  X=Y?
XEQ 00  RCL 03  RCL 06
X=Y?  XEQ 00  RCL 04
RCL 06  X=Y?  XEQ 00 /
RCL 05  RCL 06  X=Y?
XEQ 00  RCL 06  X=0?
XEQ 00  RCL 06
STO IND 00  DSE 00
GTO 00

 65♦LBL 01
FIX 1  RCL 11  X<> 00
X<> 08  RCL 11  X=Y?
GTO 08  RDN  RCL 13
X=0?  GTO 10  RDN  CLA
ARCL X  CF 17  OUTA  12
RCL 13  X=Y?  XEQ 11
XEQ A

 87♦LBL A
SF 17  "ESSAI?"  FIX 0
OUTA  PROMPT  CLA
ARCL X  "├     "  OUTA
STO 09  STO 10

 99♦LBL 02
XEQ 09  RCL IND 00  ABS
INT  STO IND 00  X≠Y?
GTO 03  1  ST+ 08  CHS
ST* IND 00

111♦LBL 03
DSE 00  GTO 02  RCL 09
STO 10  RCL 11  STO 00

118♦LBL 04
RCL 11  STO 07  XEQ 09
RCL IND 00  X<0?  GTO 07

125♦LBL 05
RCL 12  RCL IND 07  X≠Y?
GTO 06  .1  ST+ 08
ST+ IND 07  GTO 07

134♦LBL 06
DSE 07  GTO 05

137♦LBL 07
DSE 00  GTO 04  1
ST+ 13  GTO 01

143♦LBL 08
FIX 0  CLA  ARCL 09
"├*"  ARCL 13
"├ ESSAIS"  BEEP  OUTA
OFF

153♦LBL 09
RCL 10  INT  10  /
STO 10  FRC  10  *
STO 12  RTN

164♦LBL 10
"                "
"├ N.B"  OUTA  GTO A

169♦LBL 11
FIX 0  "SOLUTION : "
RCL 01  ABS  ARCL X
RCL 02  ABS  ARCL X
RCL 03  ABS  ARCL X
RCL 04  ABS  ARCL X
RCL 05  ABS  ARCL X
OUTA  OFF  END

LBL'VMIND
END          398 BYTES
```

APPLICATIONS

LECTEUR OPTIQUE

OBJET : UTILISATION DU LECTEUR OPTIQUE AVEC UN HP 41C

J'utilise beaucoup le lecteur optique bien que je n'ai pas de moyen d'imprimer des codes barre (j'ai bien une imprima nte 82143A mais les codes barre que l'on sait produire avec sont très limités voir L'Ordinateur de Poche n°6 p54 et PPC T n°1 p13)

- Tout d'abord, je lis les programmes qui paraissent en codes barre dans PPC CJ (USA) , et le temps à consacrer au HP étant court je ne rentre pratiquement plus que ceux-là . Je balaye aussi ceux qui se trouvent dans le manuel du PPC ROM, dans les ouvrages de WICKES et de REINSTEIN, dans PPC T (j'espère qu'il y en aura d'autres) et, occasionnellement dans l'OI (c'est arrivé une fois), et dans certaines revues techniques américaines (en particulier Petroleum Engineer International).

- Mais ma plus grosse utilisation de ce mode d'entrée en machine c'est les planches de fonctions. Depuis que j'utilise le lecteur optique, je n'assigne presque plus rien et je rentre la majorité des fonctions en balayant quelques centimètres. J'ai en permanence à portée de la main :

. les deux planches HP (CAT 3, Printer, Card Reader, Wand). Il n'est pas plus difficile de se rappeler la localisation d'une fonction sur une feuille,ou sur un clavier, et pas de SHIFT et pas de ALPHA.

. une planche des "EXTENDED FUNCTIONS" que je me suis bricolé avec l'imprimante 82143A (Réf ci-dessus). Pour cette planche l'ordre alphabétique évite toute mémorisation de la place sur la feuille.

Soit en programmation, soit en mode direct, je n'épelle pratiquement plus de fonction par XEQ ALPHA....ALPHA, sauf celles qui viennent toutes seules sous les doigts PRP, ACX, ACA et c. et celles du PPC ROM qui n'ont que 2 lettres et qui sont en fouillis.

Pas très loin de la portée de ma main, il y a aussi des planches pour la programmation synthétique:

. les 128 caractères alpha 00 à 7F (dernière page du bouquin de WICKES)

. les puissances de 10 sans 1 : E , E1 , E2 ... E-1 , E-2 etc. (PPC CJ V7N9P10 et V8N5P24)

. les 3 planches de Jake SCHWARTZ (PPC 1820) : STO , RCL , X<> , ISG DSE , VIEW , ASTO , ARCL , ST+ , ST- , ST× , ST/ et ces mêmes fonctions indirectes pour M N O P Q↓a b c d et e. Pour cette série (un peu différente du début publié dans PPC T n°9 p60-4 mais qui donne les mêmes résultats à un appui sur la flèche près), il ne faut pas oublier les END avant chaque balayage sinon on efface ce qui précède. END est en codes barre sur ces planches.

. également issue du PPC CJ (V7N8P14 et V8N3P22) une planche de APPEND n NULLS avec de 1 à 14 NULLS.

Certaines de ces planches sont lisibles après photocopie, d'autres ne le sont pas. Ne pas oublier, en cas de photocopie, de faire 2 tirages à 90°, l'un des deux sera mieux (ou moins mal) lu que l'autre.

Jean-Pierre (7809T29) REYNIER bourg de MONTREM 24110 Saint-ASTIER

ACCELERE

Franck Lebastard
10, rue de la Jalousie
35600 Redon

Redon, le 18 mai 1984

Banc d'essai du système prototype constitué par

- 1 HP 41 CX modifiée : accélérée 1,7
  compatible module rouge
  compatible recharge 4 ports
- 1 batterie 150 mAH
- 1 chargeur 220 V/ HP 41 ports

- 1 module rouge JV-XFM-C

, système élaboré par Jacques Vaucelle.

Nous nous attacherons dans ce banc d'essai à ne pas considérer la CX en tant que telle, mais plutôt par les aspects qui en font son originalité, à savoir les modifications faites pas Jacques.

Première prise de contact avec la machine, faisons nous plaisir : effectuons un BEEP. Quelle douce musique à nos oreilles, nous sommes tout simplement charmés par la vitesse de croisière de notre chère petite. Quelle différence avec une 41 non accélérée ! De ce pas, testons la vitesse : le time étant déja à l'intérieur, rentrons simplement :

```
01 LBL VIT
   1 E3   03 0
   RUNSW
05 SETSW
   LBL 00
   DSE ■Y
   GTO 00
   STOPSW
10 RCLSW
   ATIME24
   AVIEW          (fix 6)
   END
```

Faisons la tourner quelques secondeshistoire de compiler le gto oo, r/s, rtn, puis r/s : On ne voit même pas le canard à l'affichage tellement cela va vite. Enfin, l'affichage se stabilise : 57 secondes contre environ 1 Mn 35 s pour une machine classique ; ça tourne nettement plus vite. Si vous changez le pas 02 en 100 et intercalez un beep entre le lbl oo et le dse y, vous constaterez que ce nouveau prgm tourne en 1 mn 36s. Pas mal pour cent beep. Il est inutile de dire qu'accélérer une 41 ne présente aucun risque pour la machine.

En examinant les ports de la machine, je constate la présence dans le port 1 d'un xmémoire dont le nom a été coloré en rouge : c'est le nouveau triple xmémoire de Jacques, le JV-XFM-C à mémoire continue, comme l'indique le -C final, module prototype baptisé du nom de "module rouge". Ce nouveau module est très pratique. Placé dans le port 1 ou 3, il apporte à l'utilisateur son bloc de 603 registres. La CX telle qu'elle est compte donc 1046 registres, dont 319 de mémoire utilisateur, les 603 du module rouge, et les 124 de xm déjà dans la machine.

J'apprendrai par la suite qu'un tone synthétique valide l'un ou l'autre des deux blocs, le bloc de 127 xregs pouvant être étendu à également 603 registres sans nouvelle modification de la 41. Ce qui donnerait une HP 41 CX munie de 1525 registres, le tout immédiatement utilisable. Le module rouge est donc à mémoire continue : sorti de la machine, il conserve tous ses fichiers pendant plusieurs heures. Il est donc pratique de le retirer quand on fait par exemple des essais synthétiques et que l'on risque le mémory lost, ou tout simplement pour effectuer un mémory lost volontaire que l'on ne désire actif que sur la mémoire centrale. Ce module, comme les autres triples d'ailleurs permet de transférer de très longs programmes ou fichiers d'une machine à l'autre.

Ma nouvelle machine a également été dotée de batteries rechargeables un peu spéciales puisqu'elle donne à la 41 une autonomie d'environ une dizaine d'heures sans interruption, avec travail permanent de l'hp-il, ce qui est tout à fait remarquable; essayez donc de voir se que vous pouvez tirer de votre batterie hp classique. La machine est d'autres parts rechargée par l'un des quatre ports grace au chargeur 220V spécialement aménagé pour cela. L'extrémité est constituée par un ancien module xfunction et est dotée d'une diode qui permet, soit de vérifier l'état de la batterie(la diode est plus ou moins lumineuse) soit de vérifier que la recharge en cours se passe bien. L'avantage de cette alimentation par un port est la possibilité de recharger la batterie par un adaptateur sur allume-cigares ou une batterie de poche, ce qui est très pratique, par exemple pour un ingénieur en campagne n'ayant pas à sa disposition de prise 220V.

En conclusion, et après de nombreux tests, je puis vous affirmer que cette machine est tout simplement extraordinaire, non seulement bien sûr parce que c'est une CX, mais aussi parce que les extensions dont elle a été dotée la rendent très puissante : grande vitesse, grande autonomie, grande mémoire, et en plus, complètement modulable. Que demander de plus ?

Si vous aviez besoin de renseignements supplémentaires, n'hésitez pas à me contacter, ou à contacter Jacques lui-même qui ce fera un plaisir de réaliser la machine de vos rêves. Je vous donne ses coordonnées :

Jacques Vaucelle
"La Chasserie"
35133 Romagné tel.(99)98.83.69

A tous,
Heureuse Programmation

Franck Lebastard

T35

CONTRE - ATTAQUE

■ Voici une version améliorée (tous le monde n'est peut-être pas du même avis) du programme Contre-Attaque de Bruno TREDEZ (T 120) paru dans PPC-T N°7 (Septembre-Octobre 83).

Ma version (contrairement à la sienne) néccéssite le TIME pour des commiditées d'affichage. Je vais rappeler, en recopiant tout ou presque, les règles du jeu et je vous conseillerais de jeter un oeil sur l'exemple proposé en fin.

■ Des bombes tombent sur vos usines, vous devez les détruire avec un canon laser situé à l'altitude 1, avant qu'elles n'atteignent le sol. l'écran du calculateur (radar) indique, entre 2 crochets, l'altitude d'une dans chaque colonne, un "-" s'il n'y a rien, et la position du canon entre 4 points; à droite se trouve le total des bombes qui ont touchées vos usines.

Les bombes apparaissent de façon aléatoire à l'altitude 9, pour les détruire il faut placer le canon sur la même colonne puis tirer ou bien, si la bombe se trouve à la même altitude(1) et dans une case adjacente au laser, il suffit de placer celui-çi sur la bombe ce qui économise un tir.

Trois touches sont utilisées en mode USER pour agir sur le canon: [A] déplacement à gauche, [B] ou à droite et [F] tir ce qui permet de tenir la HP de la main gauche en appuyant avec le pouce. L'écran est cyclique, c'est à dire qu'un déplacement à gauche alors que vous êtes déja dans la case la plus à gauche, vous fait passer à l'autre bout à droite et inversement.

■ Après avoir lancé le programme (par la fenêtre) en SIZE 015 il demande les options:

tapez 01 ~ le jeu s'arrête lorsque 4 bombes au lieu de 8 ont touchéees vos usines (bonjour les dégats)
02 ~ il apparait une bombe sur 4 au lieu de 1/7 en moyenne par colonne (bonne chance)
03 ~ les installations couvrent les 7 colonnes au lieu de seulement 2-3-5-6

[R/S] relance le programme et l'ALERTE est donnée. A vos lasers !!!!

Au TONE vous voyez le canon à droite et une bombe dans une des autres cases, à vous de jouer (ont peut aussi ne rien faire) pendant 1s. Ensuit e l'affichage de la situation se fait en deux temps: d'abord les explosions et déplacements du canon puis descentes des bombes d'une case et autres explosions et enfin vous avez de nouveau la main. Presque tout est comptabilisé pour le calcul de la cote de force finale.

■ Une bombe détruite par un tir vaut 1 point et 2 si c'est par un déplacement du canon (BANZAï). Pour un nouveau jeu, faire [R/S] aprés affichage de la cote.

 # MEMOIRES #

R00 à R07: Contenu des colonnes
R08: Nombre pseudo-aléatoire
R09: Caractère "-"
R10 à R13: Compteurs
R14: Position du canon

 # flags #

(voir les options)

```
LIST 102
 16:07   30.07
 01LBL "C-A"
"1 " RCL [ STO d
TIME INT LASTX * FRC
STO 08
 11LBL 10
RCLSW RCL 12  E6 /
X>Y? SETSW X<Y? X<>Y
LASTX *
"HIGHT GAME TO D"
"ATE: " ARCL X AVIEW
TONE 9 BEEP TONE 9
TONE 8 TONE 6 TONE 7
PSE CLX STO 10
 35LBL 07
CLX "OPTION 1-2-3"
PROMPT FC?C IND X
SF IND X FC?C 00
GTO 07 "ALERTE" AVIEW
SF 25 SF 99 6 STO 14
7 FS? 01 3  E3 /
STO 11 "-" ASTO 09
6 E-3 RCL b ASTO IND Y
ISG Y STO b SF 00
XEQ 11 STO 12 9
STO IND Z
 67LBL 12
XEQ 13 RCL 14  E
TONE IND 11 PSE "RATE"
AVIEW TONE 3 GTO 00

 77LBL F
" ^FEU^" AVIEW ST- 10
GTO 02
 82LBL A
" ^G" - GTO 01
 86LBL B
"    D^" +
 89LBL 01
7 MOD AVIEW STO 14
RCL IND X  E XY?
GTO 00
 98LBL 02
"*" ASTO IND 14 2
ST+ 10
```

```
LIST 100
 16:09   30.07
103LBL 00
XEQ 13 TONE ^
106LBL 08   9  ..¥....
RCL IND 13 SIGN X=0?
GTO 11 TONE 7
DSE IND 13 GTO 09
FC? 03 GTO IND 13

116LBL 01
117LBL 02
118LBL 04
119LBL 05
ISG 11 GTO 06 "TOTAL="
ARCL 12 AVIEW TONE 5
BEEP TONE 5 PSE
RCL 10 RCL 12 / 50 *
"COTE=" ARCL X " %"
TONE 9 PROMPT GTO 10

140LBL 00
141LBL 03
142LBL 06
"*" ASTO IND 13 GTO 09

146LBL 11
7 FS? 02 4 RCL 08
R-D FRC STO 08 * INT
 E FS?C 00 RTN X=Y?
ST+ 12 XY? RCL 09
X=Y? 9 STO IND 13

166LBL 09
ISG 13 GTO 08 GTO 12

170LBL 13
RCL 14 6 E-3 STO 13
"["
175LBL 14
INT X=Y? ":"
ARCL IND X X=Y? ":"
X<> L ISG X GTO 14
"] " RCL 11 ADATE
AVIEW END
```

 # LIGNES #

02: codes hexadécimaux: 31 00 80 00, arme les flags 26 27 31 40
23: f6 7f 41 54 45 3a 20
96: XY? --> X≠Y?
105: TONE P, 9F 78
136: F3 7F 20 25
137: TONE 89, 9F 59
161: XY? --> X≠Y?
178 & 181: F2 7F 3A
185: F2 7F 5D

```
9 ..¥....
8 ...¥...
7 .......
6 .......
5 .......
4 ¥......
3 ......*
2 .....¥.
1 ......^
0 .■■.■■.

[4-98-2:*:] 03
```

ECRAN DU RADAR HP-41
****

"C-A" 368 OCTETS
 55 REGISTRES

REIGNER CYRIL
T 282

MESSAGES D'ERREUR

HP-41 CX

Affichage	Fonctions	Signification
ALPHA DATA	(CLAIMA) (CLAIMX) (CWPT) (AROT) (CRFLAS) (CRFLD) (DELCHR) (GETRX) (PASN) (PSIZE) (REGMOVE) (REGSWAP) (RESZFL) (SAVERX) (SEEKPT) (SEEKPTA) (STOFLAG) (X<>F)	Le registre X contient des données alphanumériques, alors que la fonction requiert un nombre.
CAT EMPTY	(CAT) 5 ou (ALMCAT)	Aucune alarme en mémoire centrale.
	(CAT) 6	Aucune touche n'est réassignée.
DATA ERROR	(CLAIMX) (RCLAIM)	$\|X\| > 999$ ou $X=0$
	(GETKEYX)	$\|X\| \geq 100$
	(RESZFL)	$\|X\| \geq 1000$ ou $X=0$
DIR EMPTY	(CAT) 4 ou (EMDIR)	Aucun fichier en mémoire étendue.
FL NOT FOUND	(ASROOM) (ED) (RESZFL)	Le registre ALPHA est vide et il n'y a pas de fichier de travail, ou le nom de fichier spécifié par le registre ALPHA n'existe pas en mémoire étendue.
FL SIZE ERR	(RESZFL)	Risque de perte de données en diminuant la taille d'un fichier AS ou DA.
FL TYPE ERR	(ASROOM) (ED)	Le fichier est un fichier DA ou PR.
	(RESZFL)	Le fichier est un fichier PR.
NONEXISTENT	(SWPT) (CLRGX)	Un (ou plusieurs) registre spécifié par le contenu de X n'existe pas.
	(X..NN?)	Le registre dont l'adresse est en Y n'existe pas.
NO ROOM	(ED)	Tentative d'addition ou d'insertion de caractères dans un fichier déjà plein durant l'édition du fichier.
	(RESZFL)	X+FLSIZE>EMROOM
NO SUCH ALM	(CLAIMA)	Aucune alarme ne contient le même message que la chaine contenue en ALPHA.
	(CLAIMX)	$\|X\|$ > nombre d'alarmes en mémoire.

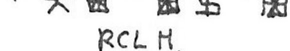
DRAPEAUX - IN

Pascal DAGORNET (T 467)
rue de la Planche
22 190 PLERIN.

<u>objet</u> : manipulation de drapeaux sur HP-41 à partir de chaînes alphanumeriques.

ceci s'adresse à des débutants (comme moi) en programmation synthétique et qui entendent souvent parler de manipulation "anormale" des flags - au début je mettais n'importe quel caractère dans d pour voir ce que ça donnait mais souvent ce que j'obtenais était très bizarre ; je me suis donc resolu à attaquer d avec une feuille et un crayon et non seulement avec les yeux fixés sur l'affichage de la machine pour le voir s'allumer comme un arbre de noël.
J'essaierais donc ici d'expliquer la methode de manipulation des drapeaux - une des methodes ?

rappel : le registre d est formé de 7 octets donc 7x8 = 56 bits qui representent les 56 drapeaux numerotés de 0 à 55 - pour manipuler les flags, nous utiliserons le registre M composé lui aussi de 56 bits
le but de la manoeuvre est de determiner une chaîne de 7 caractères Alpha (en M) qui rappelés en X puis stockée en d agitera les drapeaux choisis.

ex : 

RCL M
STO d

(ou une sequence similaire qui sauvegarderait la pile)

<u>Exemple</u> : Prenons une HP 41 <u>nue</u> *
nous la voulons en :

	Flags armés
fix 3	38-39-40
separateur de groupes de 3 chiffres	29
virgule decimale	
drapeaux 0 à 10 baissés	
mode degré	
- calcul	
bip Validé - c'est tout.	26

* la HP41 est supposée sans Peripheriques à part le lecteur de carte, le module Math ou autres roms

(Pour les possesseurs du Time, pour DMY, le flag 31 est levé et pour MDY il est baissé)

Avant tout, il serait utile d'avoir le shéma du registre d
(Cf. Au fond de la HP41, page 42 ou 110 ou PPC-T N°7 p 25)

Etude de d octet par octet, bit par bit
- l'octet n°6 - digits 13 et 12
celui ci "contient" les flags de 0 à 7. ceux ci étant desar-

més, les bits correspondant seront à 0 (le bit correspondant à un drapeau levé est egal à 1)

drapeaux 0 1 2 3 4 5 6 7

bits(oct1) 0 0 0 0 0 0 0 0 → à ce nombre en base binaire correspond le nombre decimal : [0]

le code du caractère le + à gauche de d (ou de M) sera donc egal à 0 (le caractère nul : "—")

si nous avions voulu armer le flag 5 et desarmer les autres (toujours du 1er octet) nous aurions eu :

drapeaux 0 1 2 3 4 5 6 7

bits 0 0 0 0 0 1 0 0 → nombre binaire : 100

→ nombre en base decimale : 4.

l'octet n° 6 aurait donc été le caractère "I" (cf Table des codes)

- l'octet n° 5

drapeaux 8 9 10 11 12 13 14 15

bits 0 0 0 0 0 0 0 0

→ code decimal : [0]

→ caractère : "—"

- l'octet n° 4

(drapeaux de 16 à 23 desarmés)

→ code decimal : [0]

→ caractère : "—"

- l'octet n° 3.

dans le récapitulatif des drapeaux armés pour cet état de la machine (fin 3...etc...), ceux qui sont levés sont les flags 26, 29, 38, 39, 40. ceux qui nous intéressent ici sont les 26 et 29.

drapeaux 24 25 26 27 28 29 30 31

bits 0 0 1 0 0 1 0 0

→ nombre binaire : 100100

→ — decimal : 36.

code decimal de l'octet n° 3 : [36]

caractère : "$"

- l'octet n° 2.

drapeaux 32 33 34 35 36 37 38 39.

bits 0 0 0 0 0 0 1 1

(code) nombre binaire : 11

→ code decimal : [3]

→ caractère : "▪"

- l'octet n° 1

drapeaux 40 41 42 43 44 45 46 47

bits 1 0 0 0 0 0 0 0

→ nombre binaire : 10000000

→ code decimal de l'octet n° 1 : [128]

le caractère sera : "▪"

- l'octet n° 0

drapeaux 48 49 50 51 52 53 54 55.

bits 0 0 0 0 0 0 0 0

octet à code decimal : [0]

caractère nul : "—"

mettons les caractères déterminés bout à bout :

octets 6 5 4 3 2 1 0

caractères — — — $ ▪ ▪ —

nous obtenons une chaîne de caractères de type F7, ...
ici : F7, 0, 0, 0, 36, 3, 128, 0
par ex si votre machine est en USER, SCI 4 ..., après la séquence ci-dessous dans un programme, votre machine se mettrait en FIX 3, mode calcul ...
- - - - $ ▩ ▩ -

RCL M
STO d .
:
Vu ?

Ceux qui ont des périphériques (ou autres) IHP. par ex. pourront la faire écrire en minuscules après cette séquence, il suffirait de modifier l'octet n°5
bref, affolez vos chapeaux, amusez vous bien.
Personnellement, j'utilise cette méthode pour faire un nettoyage à ma HP-41.

bonne Programmation !

Pascal Dagornet

Objet : Sur HP-41, intégration par la méthode des rectangles.

Et oui !, encore un autre programme "IN", mais il est très court et très rapide, la précision est cependant moyenne (de l'ordre de 1/10000e pour une fonction non constante et pour N = 60 ; sur 1 unité pour l'axe x'Ox)

```
01 LBL "IN
02 STO N
03 X<>Y
04 RCL Z
05 -
06 X<>Y
07 /
08 STO M
09 2
10 /
11 +
12 STO O
13 O  (ou ;)
14 LBL 00
15 RCL O
16 XEQ "FF
17 ABS
18 +
19 RCL M
20 ST+ O
21 RDN
22 DSE N
23 GTO 00
24 R↑
25 X
26 A=
27 ARCL X
28 AVIEW
29 END.
```

Principe :
a ENTER
b ENTER
N XEQ "IN
→ affichage de $\int_a^b f(x)\,dx$

N : nombre de "découpages"

ligne 02 : N stocké dans N pour le contrôle de la boucle.
ligne 08 : $\frac{b-a}{N}$ stocké dans M
- 12 : $a + \frac{b-a}{2N}$ stocké dans O

ligne 24 à la fin : $\Sigma f(x) \times \frac{b-a}{N} = \mathcal{A}$ et affichage.

Le label de f(x) doit être "FF"
Restriction : la fonction ne doit utiliser que X Y Z
car
T indifférent
Z indifférent
Y $\Sigma f(x)$
X Valeur

ASTUCE

Franck Lebastard
10, rue de la Jalousie
35600 Redon
T35

Rennes, le 3 juin 1984

Astuce.

Il arrive souvent à un ingénieur qui travaille sur le terrain d'utiliser avec son calculateur des prgms personnels pour saisir des données puis éventuellement, les transformer avant de les stocker. Il est alors utile de redéfinir le clavier par exemple avec Getkey, certaines touches correspondant aux fonctions de son choix et les autres étant inopérantes. Malheureusement, cette technique fort utile parfois est très malcommode dès qu'on a besoin de rentrer des nombres : Getkey ne permet pas la rentrée des nombres.

Voici une astuce que j'utilise personnellement dès que mon clavier doit être redéfini. Lors de l'initialisation de mon prgm, j'assigne à mes touches habituelles les fonctions dont j'ai besoin, je laisse libres les touches comportant les fonctions qui m'intéressent comme 1, 2, ...,9, chs, enter/, etc, et j'assigne à toutes les autres la fonction SST par :

```
sst
'Code touche'
pasn
```

Il ne reste plus qu'à faire que mon prgm soit privé. Le prgm ne tournant pas, et le pointeur y étant dedans, les touches correspondant à sst sont inopérantes : le sstage des prgms privés est interdit. Par contre, le pavé numérique est libre, on peut rentrer n'importe quel nombre à son gré puis exécuter une fonction du prgm qu'on a assignée quelque part. Autre avantage, les touches on, user sont libres, permettant un contrôle parfait du calculateur : on éteint ou on quitte le clavier "spécial" quand on veut. Certains objecteront que la touche shift ne peut être condamnée par la séquence de touches précédente si on n'a pas besoin des touches shiftées. Erreur ! les heureux possesseurs du ppc rom n'auront qu'à inclure dans leur prgm la séquence suivante :

```
0
8
31
xrom'lk'
```

Pour ceux qui n'ont pas de lecteur de cassettes ou de cartes et qui ne peuvent donc pas privatiser normalement leurs prgms, la manoeuvre de privatisation est la suivante, le Byte Grabber (247,63) étant assigné à une touche.

A la fin du prgm, rajouter :

```
rcl ind 16
rcl ind 64
lbl 00
hms+
```

, revenir juste avant rcl ind 16 et BGrabber. Effacer la chaine obtenue ainsi que le rcl ind 16 qui la suit. Un sst vous montrera le end que vous venez de créer et qui contient le private.

Faites xeq'pack' et voyez le private qui en résulte: votre prgm est maintenant privé.

J'indique pour finir que ceux qui le veulent peuvent déplacer la touche r/s en effectuant

```
0
5
'Code touche'
xrom'ik'
```

Même chose pour déplacer la touche shift en remplaçant le 5 par 14. Malheureusement pour ceux qui ne l'ont pas, le "ppc rom is of course needed" (disponible au club)

Bonnes redéfinitions de clavier à tous,

Franck Lebastard

3

SOTIROPOULOS C.
T 434

13 a. ACH. ARCHAMBAULT
95110 SANNOIS

LES 3 ROUTINES QUI SUIVENT FONT@

SAVEST:SAUVEGARDE DE LA PILE DANS LE X-F.

RCLST=RAPPEL D UNE PILE SAUVEGARDEE.

SWAPST:ECHANGE DE LA PILE ACTUELLE AVEC UNE PILE SAUVEGARDEE.

SIZE 000

NOTE

EVIDEMMENT NE PAS UTILISER SWAPST COMME SOUS-PROGRAMME DE 3ème NIVEAU OU PLUS.

```
01♦LBL "SAVEST"
02 SF 25
03 "ST"
04 PURFL
05 X<> L
06 RDN
07 4
08 CRFLD
09 RDN
10 LASTX
11 R↑
12 SAVEX
13 R↑
14 SAVEX
15 R↑
16 SAVEX
17 R↑
18 SAVEX
19 END
```

```
01♦LBL "RCLST"
02 SF 25
03 "ST"
04 ,
05 SEEKPTA
06 FC?C 25
07 SF 30
08 GETX
09 GETX
10 GETX
11 GETX
12 END
```

```
01♦LBL "SWAPST"
02 X<> L
03 CLX
04 "ST"
05 SF 25
06 SEEKPTA
07 FC?C 25
08 SF 30
09 RDN
10 LASTX
11 STO [
12 RDN
13 STO \
14 RDN
15 STO ]
16 RDN
17 STO a
18 GETX
19 X<> a
20 GETX
21 X<> ]
22 GETX
23 X<> \
24 GETX
25 X<> [
26 X<> L
27 CLX
28 SEEKPT
29 RDN
30 LASTX
31 R↑
32 SAVEX
33 R↑
34 SAVEX
35 R↑
36 SAVEX
37 R↑
38 SAVEX
39 RCL a
40 RCL ]
41 RCL \
42 RCL [
43 CLA
44 END
```

e

Plaçons par exemple, le POINTeur sur la ligne 6Ø d'un PRGM situé en MEM (mode RUN GTO "nom du PRGM" GTO .Ø6Ø) RCL e HEQ "NH" (JPC N6P6). Les 3 DIG de Droite seront: Ø3C qui est le N° de la ligne du PRGM (HEX Ø3C = DEC 6Ø). Vous pouvez vérifier avec toute autre valeur.

Taper le PRGM suivant: LBL "LP" CLA 8 XTOA RCL M STO e STOP BEEP et l'XEQécuter. Vous constatez, en mode PRGM, que le POINT. s'est bien placé sur la ligne Ø8 mais sans l'XEQécuter. Si vous aviez des ASN sur des touches SHIFTées, elles sont perdues !

ASN la FS 'STO M' à la touche -34 (145 ENTER↑ 117 ENTER↑ 34 CHS XEQ "XYZASN", par exemple) puis RCL e XEQ "NH". Nous obtenons ØØ4 comme premiers DIG en ALPHA et '4' indique que la touche -34 est ASN. Notre '41' ayant 35 touches, 36 BITs (9 DIG) ont été utilisés pour représenter (répertorier) la situation des touches (ASN = 1, non ASN = Ø).Vous trouverez ci-joint un tableau représentant les 36 premiers BITs du REG e (ou ⊢). Le 2ème rang de CHiffres indique les N°s de touches sous les N°s de BITs correspondants. C'est le BIT 24 qui est inutilisé et qui se trouvera donc toujours à Ø. A droite de ce tableau, j'ai représenté les N°s de BITs implantés sur le clavier pour faire apparaître la disposition adoptée. Je dois signaler que ce "papier" n'a pu être écrit que grâce à la lecture du Manuel de la PPC-ROM (PRGM "MK") qui est une véritable mine d'or (492 pages) et qui est écrit d'une façon très claire (heureusement pour mon 'anglais' minable !).

Si nous reprenons l'exemple précédent, le BIT Ø9 sera levé (à 1) et, sur 36 BITS, nous obtenons l'équivalent HEX ØØ4ØØØØØØ (9 DIG). Là, il faut avouer que la HP-16C est bien utile.....

Si nous prenons l'exemple mentionné dans la derniére rangée du tableau, les touches 11, 12, 13, 14 & 15 sont ASN; la VAL BIN correspondante sera donc: ØØØ1ØØØØØØØ1ØØØØØØØ1ØØØØØØØ1ØØØØØØØ1 ce qui donne en HEX: 1Ø1Ø1Ø1Ø1et vous pourrez vérifier.....

Les CODEs des F sont STO dans les REG d'ASN dont la 1ère ADD ABS est DEC 192. L'introduction de ces CODEs se fait de D à G et le dernier entré pousse le précédent vers la G, celui de G étant alors "poussé" dans la partie D du REG supérieur et ainsi de suite. Chaque REG peut contenir 2 ASN de la façon suivante : (ceux qui ne disposent pas d'un LECT d'EPROM et du jeu d'EPROM '-MLEPROM 1H' de Stéphane BARIZIEN, devront me faire confiance)

192 CHS XEQ "NRCL" XEQ "NH"

AFFichage : FØØØØØØØ91753B ALPHA

où 'F' est une constante indiquant qu'il s'agit de CODEs 'ASN', les 2 derniers DIG '3B' indiquent

le CODE de la touche suivant tableau Fig. 5 du bouquin de JDD (équivalent de -34) et, HEX 91 = DEC 145 qui est le préfixe de STO M et HEX 75 = DEC 117 qui est le suffixe. A la prochaine ASN, les 6 DIG de D seront poussés de 6 'crans' vers la G et, le 2ème DIG étant inutile, sera toujours Ø.

Signalons que les ASN de LBL ALPHA globaux sont STO avec le LBL correspondant et c'est donc "gratuit"; il suffit de passer en mode USER .....

Lorsque nous effaçons une ASN, seul le REG "répertoire" (e ou f) est affecté (le BIT adéquat est mis à zéro), les CODEs STO ne sont pas supprimés, ce qui occupe inutilement un demi REG par ASN. Nous verrons ultérieurement avec le PRGM "PK" (Pack Key) de la toujours aussi fameuse PPC-ROM, comment nous pouvons récupérer cette place perdue.

Vous voici, je l'espère, en possession des éléments indispensables qui pourront vous permettre de décortiquer par exemple, un PRGM tel que "MKX" de Tapani TARVANEN & Clifford STERN.....

Je dédie ce "papier" à JDD dont le bouquin m'a également été d'une grande utilité car écrit dans ma demi-langue maternelle.....

rS (T178+P2Ø)

P.S.: Un beau sujet de PRGM pour ceux qui ont compris (?) : "PKX".....

IMPRESSION de FICHIERS ALPHA

Le petit PRGM "PFA" ci-contre permet d'IMPrimer le contenu d'un FICHier ALPHA (ASCII). Il suffit de placer en ALPHA le NOM du FICH, de connecter une IMPrimante et: XEQ "PFA".

J'ai fait rapidement ce PRGM pour l'utiliser à la suite de la Fonction 'ED' de la nouvelle HP 41CX mais il est évident (et heureux!) que ce PRGM peut être utilisé dans d'autres cas. Il permet de "lire" 9Ø1 lignes de FICH mais vous pouvez aller jusqu'à 1 ØØØ (?) en plaçant ,999 au Pas Ø4.

Bien sincères amitiés à tous,

rS (T178+P2Ø)

```
01*LBL "PFA
"
02 ,
03 SEEKPTA
04 SF 25
05*LBL 00
06 GETREC
07 FC? 25
08 RTN
09 PRA
10 GTO 00
11 END
```

PFA

RCL CAT 1

S'il vous est arrivé de découvrir un CATalog 1 pour le moins bizarre à la suite d'un NONEXISTENT pour le moins surprenant, alors que vous aviez en MEM centrale un précieux PRGM inédit, les lignes qui vont suivre risquent de vous servir un jour ou l'autre; je vous indiquerai même comment vous placer dans une situation qui permette d'essayer la procédure ci-après. J'ai déjà écrit un "papier" sur une méthode qui permettait de se sortir d'un CAT 1 perturbé avec la PPC-ROM. Puisque nous disposons maintenant des fonctions MC de la merveilleuse 'MLEPROM 1H' de Stéphane BARIZIEN, nous allons utiliser six de ces fonctions. Je ne ferais pas l'affront (?) à ceux qui possèdent la PPC-ROM de leur expliquer comment ils devront procéder pour obtenir le même résultat.

En mode RUN (°) :

```
13          :adresse DEC absolue du REG c ,
CHS         :valeur négative obligatoire pour NRCL,
NRCL        :RCL en X le NNN correspondant au REG c ,
DECODE      :place en ALPHA les codes  HEX du REG c ,
3           :indique à SUB$ de ne laisser en ALPHA que
SUB$         les 3 DIG (CAR) de droite,
H-D         :convertit en DEC le contenu de ALPHA,
"C0002D"    :place en ALPHA les codes d'un .END. ,
CODE        :place en X le NNN des codes précédents,
X()Y        :place en X l'adresse DEC du .END. ,
CHS         :valeur négative obligatoire pour NSTO,
NSTO        :STO le .END. là où il faut,
PACK        :fait du "rangement".
```

(°) RUN = NO PRGM ou PRGM OFF

NRCL place en X le NNN correspondant au contenu du REG dont l'adresse DEC était indiquée en X : si l'adresse est négative, il s'agit d'une adresse absolue d'un REG de la RAM (MEV), dans le cas contraire (valeur positive), il s'agit d'un REG de donnée compris dans le SIZE. Ne pensez surtout pas que cette dernière possibilité est superflue! Ce RCL est NN (Non Normalisant).

NSTO fait l'inverse de NRCL. Le NNN placé en Y est STO dans le REG dont l'adresse est précisée en X.

Est-ce utile de rappeler que les 3 DIGits de droite du REG c contiennent (indiquent) l'adresse du .END. permanent ?

Je vous assure que lorsqu'on a goûté à des fonctions telles que les 6 précitées, on ne peut plus s'en passer.

Pour perturber un CAT 1, vous pouvez bien entendu utiliser la méthode décrite ci-dessus en introduisant par exemple, un 'C' à la place d'un zéro qui suit le 'C' de tête du code "ALPHA" du .END. L'essai que je viens de faire m'a donné 248.449 BYTES + 6 Octets pour le .END. , ce qui donne : 35.493 REG, de quoi simplement rêver! (trente-cinq-mille-quatre-cent-quatre-vingt-treize REG!) Une autre méthode consiste à faire un coup de BG après le .END. permanent (en mode PRGM) .....

Bons sauvetages! *rS* (T178)

X - IO

Notre charmant collègue, Jean BOSCHAT, a eu la gentillesse de se dévouer pour me traduire les Pages 30 à 82 du Manuel de ce module et je dois lui renouveler tous mes bien sincères remerciements pour ce travail important qu'il a réalisé avec beaucoup de soin et de compétence.

Je suis donc en mesure de vous donner des indications complémentaires sur les Fonctions de ce module.

FONCTIONS DE LA CONFIGURATION DE LA BOUCLE

NLOOP (dº) signalons que la boucle accepte jusqu'à 30 appareils !

(dº) signifiera que la définition faite dans JPC N°4 reste "valable"

RCLSEL (dº)

Identification des appareils:

Il existe 2 moyens d'identification d'un appareil sur la boucle :

- Device ID (identité de l'appareil)
- Accessory ID (identité de l'accessoire)

ID renvoie en ALPHA une CHaîne contenant le "Device ID" de l'APPareil PRINCipal sur la boucle. ID lit un maxi de 24 CAR envoyés par l'APP PRINC. ID est en relation avec la F "IL" FINDID qui trouve l'ADD "IL" d'un APP qui correspond au Device ID spécifié en ALPHA. FINDID utilise seulement les 7 premiers CAR de la CH entrée en ∝.

AID (identité d'accessoire) place l'identité d'accessoire de l'APP PRINC dans le REG X

FINDAID (trouve l'accessoire identifié) permet de localiser un APP d'une classe ou type particuliers que nous devons préciser par un nbre placé en X. Si le nbre est positif, FINDAID examine la boucle en commençant par l'APP PRINC, jusqu'à ce qu'il trouve un APP dont l'identité d'accessoire corresponde au nbre (partie entière) situé en X. Si un tel APP est présent, son ADD "IL" est placée en X. Si le nbre introduit en X est négatif, FINDAID examine la boucle de la mîn façon, cependant, la VAL retournée en X est l'ADD du 1er APP trouvé DANS LA CLASSE D'APP qui correspond à la VAL ABS du nbre situé en X.

Détermination de l'état d'un APP

Les 6 F suivantes fournissent des méthodes qui permettent de déterminer si un APP sur la boucle réclame un "entretien" ("service"), c'est-à-dire s'il réclame l'attention du contrôleur ou non (le contrôleur est la "41").

SQR? (Service demandé) est un test conditionnel qui teste la boucle pour connaître l'existence d'une demande éventuelle de service en envoyant le message d'identification.

POLLE (validation reconnaissance parallèle) autorise l'APP PRINC à répondre à une interrogation en //, attribue l'1 des 8 BITs de

reconnaissance // à l'APP et précise quelle signification aura l'utilisation de ce BIT

par l'APP. Les 2 dernières opérations sont déterminées par le PARAMETRE DE VALIDATION spécifié en X. Ce PARAMETRE doit être compris entre Ø & 15 (SIGN sans importance). Un tableau donne toutes indications utiles....

POLL (reconnaissance parallèle) envoie un message d'identification dans la boucle et retourne en X un nbre de Ø à 255 représentant la réponse de la boucle. Ce nbre réponse est l'équivalent DEC des 8 BITs de reconnaissance //. POLL met initialement tous les BITs à Ø dans le message d'identification. Si un APP quelconque de la boucle a été auparavant validé par POLLE, il peut armer le BIT qui lui est attribué en fonction d'une demande de service ou non demande de service, et le nbre qui reviendra en X sera ≠ de Ø....

POLLD (annulation de reconnaissance //) annule la réponse en reconnaissance // de l'APP PRINC, ainsi POLLD supprime l'effet d'un précédent POLLE adressé à cet APP. Après l'exécution de POLLD, l'APP PRINC ne répond pas aux exécutions ultérieures de POLL.

POLLUNC (déconfiguration de reconnaissance //) annule les réponses en reconnaissance // de tous les APP sur la boucle "IL".

STAT lit jusqu'à 23 Octets d'états de l'APP PRINC et STOcke les Octets sous forme d'une CH de CAR dans le REG ALPHA. LA CH sera toujours précédée du CAR S pour signifier "Status". Le CAR S est nécessaire car la CH d'états pourrait contenir 1 ou plusieurs CAR nuls en tête qui, sans cela, ne serait pas pris en compte dans le REG ALPHA (cette possibilité me semble très appréciable....).

Fonctions de contrôle d'un APP

CLRDEV (libère l'APP) "efface" l'APP PRINC, c'est-à-dire lui donne l'ordre de revenir à un état initial particulier. La réponse réelle dépend du TYP d'APP.

CLRLOOP "efface" tous les APP de la boucle en même temps.

NOTREM (pas de télécommande) fait passer tous les APP de la boucle en mode LOCAL et leur interdit de passer à nouveau en mode REMOTE.

LOCK (blocage) annule l'interrupteur de coupure du mode REMOTE qui permet de forcer manuellement 1 APP à se placer en mode LOCAL. Pour que LOCK ait un effet sur l'APP, ce dernier doit déjà être en mode REMOTE. Une fois déconnecté, l'inter le reste jusqu'à ce que la F NOTREM soit exécutée.

Envoi de commandes d'APP dépendants :

DEVL (récepteur dépendant)
DEVT (émetteur dépendant)
DEVL & DEVT envoient les commandes de RECEPteur dépendant et d'EMETteur dépendant (respectivement) qui correspondent au paramètre placé en X. Les commandes sont envoyées à l'APP PRINC. Le paramètre de commande doit être compris entre -31 et +31.

FONCTIONS DE TRANSFERT DE DONNEES :

INXB (introduit 1 Octet dans X)
OUTXB (prend 1 Octet dans X)
Ces 2 F permettent le transfert d'1 Octet. INXB demande à l'APP PRINC de transmettre 1 Octet de données à la "41". L'équivalent DEC de l'Octet, compris entre Ø & 255, est placé en X. OUTXB traduit un nbre situé en X en 1 Octet équivalent de 8 BITS et l'envoie à L'APP PRINC.

Transmissions s'interrompant sur un CAR particulier

INAC (entrée dans ALPHA jusqu'au CAR) efface le REG ALPHA et lit une séquence d'Octets de données envoyée par l'APP PRINC qu'il (INAC) place dans le REG ALPHA en faisant précéder ces Octets du CAR "postiche" D . Les CAR sont ajoutés au REG ALPHA jusqu'à la réception d'1 CAR dont le code est spécifié en X; ou bien jusqu'à ce que 23 CAR soient reçus (le D "postiche" exclu). Le CAR spécifié n'est pas inclus dans la CH ALPHA. Le DRAPeau 17 est baissé si ALPHA est rempli avant la réception de ce CARactère.
OUTAC (sortie d'1 CAR de ALPHA) envoie la CH du REG ALPHA (moins le 1er CAR non nul) à l'APP PRINC. La CH est suivie d'un CAR de terminaison correspondant au code spécifié en X.
XFERC (transfert de données jusqu'au CAR) transfère des données du 1er APP (APP PRINC) vers un 2ème APP dont l'ADD est spécifiée en X. La "41" arrête la transmission lorsque le CAR dont le code est spécifié en Y est envoyé. L'APP RECEPteur recevra ce CAR de terminaison.

Transfert d'un nbre spécifié d'Octets :

INAN (Introduit N CAR dans ALPHA) efface ALPHA, lit un nbre spécifié d'Octets envoyés par l'APP PRINC et les place dans le REG ALPHA en les faisant précéder d'1 CAR D . Le nbre de CAR à introduire, compris entre Ø & 23, est spécifié en X. INAN n'a aucun effet sur le DRAPeau 17.
OUTAN (prend N CAR dans ALPHA) transmet à l'APP PRINC jusqu'à 23 CAR issus du REG ALPHA; le nbre de CAR à envoyer est spécifié en X. Les CAR transmis sont ceux qui suivent le 1er CAR non nul de la CH ALPHA (le CAR de tête n'est pas envoyé) jusqu'au nbre de CAR spécifié; la CH entière est alors envoyée (1er CAR exclu).
XFERN (transfère N Octets) demande à l'APP PRINC d'envoyer des Octets de données à l'APP situé à l'ADD spécifiée en X. Le nbre d'Octets qui doivent être transférés est placé en Y.

Transferts s'arrêtant sur un message de FIN :

INAE (introduction dans ALPHA jusqu'au message de fin) lit des Octets de données envoyés par l'APP PRINC jusqu'à réception d'un message de FIN. INAE efface ALPHA et place les Octets de données, y compris l'Octet de FIN, dans le REG ALPHA sous la forme d'une CH de CAR précédée du CAR D . Le DRAP 17 est baissé si le message de FIN est reçu. Le DRAP est levé si ALPHA est plein avant réception du message de FIN.
OUTAE (sortie de ALPHA avec un message de FIN) envoie la CH contenue dans ALPHA (moins le 1er CAR non nul) à l'APP PRINC. Le dernier CAR

est envoyé comme CAR de FIN.

XFERE (transfert de données jusqu'au message de FIN) assure l'envoi de données de l'APP PRINC vers l'APP dont l'ADD est spécifiée en X. La "41" achève la transmission lorsque l'APP PRINC envoie un message de FIN (le RECEPteur recevra ce message de FIN).

Transferts s'interrompant sur un message CR/LF :

INACL (entrée dans ALPHA jusqu'au CAR LF) efface ALPHA, lit une CH d'Octets de données envoyée par l'APP PRINC et la place dans le REG ALPHA précédée d'un CAR D. L'introduction cesse quand un CAR LF (code 1Ø) est reçu (le DRAP 17 est baissé). CR (code 13) & LF n'apparaissent pas dans la CH placée en ALPHA. SI 23 CAR (CR exclu) sont introduits avant détection d'un LF, INACL arrête la transmission et lève le DRAP 17.

OUTACL envoie la CH située en ALPHA (moins le 1er CAR non nul) vers l'APP PRINC, sous forme d'Octets de données. Un CAR CR et 1 CAR LF sont envoyés à la suite de la CH de CAR prise en ALPHA

XFERCL (transfert de données jusqu'à LF) entame un transfert de données de l'APP PRINC vers l'APP spécifié par l'ADD "IL" placée en X. La "41" arrête la transmission lorqu'1 CAR LF est envoyé. Les CAR CR sont transmis sans aucun traitement spécial.

Transferts s'interrompant sur un message de fin de transmission :

XFER (transfert de données jusqu'àu message de fin de transmission) indique à l'APP PRINC de faire parvenir des données à l'APP dont l'ADD "IL" est spécifiée en X. Cette F peut être utilisée avec les APP qui envoient un message "IL" de fin de transmission (EOT) lorsqu'ils ont achevé une transmission. La "41" utilise le message de fin de transmission pour déterminer quand elle doit reprendre l'exécution du PRGM, une fois que le transfert de données est complet.

Transfert de PRGM de la "41" :

INP (entrée d'un PRGM) demande à l'APP PRINC d'envoyer une série d'Octets de données que la "41" peut transformer en un PRGM. Ce PRGM remplace le dernier PRGM en MEM. Les 4 1ers Octets reçus indiquent la longueur (en nbre d'Octets du PRGM). Une fois les 4 Octets reçus, INP teste la capacité disponible de la MEM PRGM de la "41".

OUTP (sortie de PRGM) trouve le LBL correspondant au nom du PRGM situé en ALPHA et envoie à l'APP PRINC une séquence d'Octets de données représentant le PRGM en question. Les Octets du PRGM réel sont précédés de 4 Octets représentant la longueur du PRGM en Octets et sont suivis de 2 Octets de contrôle. Si ALPHA est vide, OUTP envoie le PRGM où se trouve le pointeur. OUTP ne peut envoyer un PRGM PRIVATE. Deux "41" connectées entre elles par chacune une interface, permettent de faire passer un PRGM de l'une dans l'autre.....

FONCTIONS DE CONTROLE ETENDU

ADRON (ADressage automatique) déclenche l'adressage automatique de la boucle et les commandes de désignation des émetteurs et récepteurs utilisés par les F de transfert de données. Le DRAP 34 est utilisé par le module d'E/S pour controler les modes d'adressage : ADressing ON/OFF. ADRON baisse le DRAP 34 pour indiquer que le mode d'adressage automatique (AD. ON) est actif. Le DRAP est baissé par un MEMORY LOST mais il n'est pas affecté par la simple extinction du calculateur.

ADROFF (absence d'adressage automatique) lève le DRAP 34, ce qui place le module d'E/S en mode ADresssing-OFF. Dans ce mode, ni l'adressage auto de la boucle, ni l'attribution des rôles d'émetteur ou de récepteur ne sont assurés par les F de transfert de données (de types IN, OUT et XFER). Ainsi en mode AD. OFF, les F de transfert de données déclenchent seulement l'envoi des données (Send DATA), les messages "DATA or END" (données ou fin) et "READY for Command" (prêt pour ordre) sur le HP-IL. Les F "IL" INA & OUTA ne sont pas affectées par les modes AD.OFF & AD.ON.

Envoi des messages de commandes HP-IL :

SEND (envoi commande) transmet le message de commande "IL" spécifié en X, suivi d'un message "prêt pour ordre" (RFC). Le paramètre de commande (compris entre Ø & 255) sélectionne l'un des 256 messages de commande possibles, réunis dans un tableau (Page 61 du Manuel).

LAD (adresse récepteur) transmet un message d'ADresse-Récepteur (suivi de RFC) pour faire de l'APP dont l'ADD est spécifiée en X, un récepteur.

TAD opère de la mîm façon que LAD, en envoyant un message d'ADresse-Emetteur. Le nbre situé en X avant exécution de LAD ou de TAD doit être compris entre 1 & 3Ø (sans tenir compte su SIGN ni de la partie FRC).

DDL (récepteur Dépendant) envoie le long de la boucle la commande de RECEP DEPendant, le N° du message (de Ø à 31) est spécifié en X. Le message est reçu par tous les RECEP actifs.

DDT (émetteur Dépendant) envoie la commande d'émetteur Dépendant (de Ø à 31) spécifiée en X. Seul l'émetteur actif à ce moment répond au message.

UNL (non récepteur) envoie le message de NON-récepteur sur la boucle, enlevant à tous les récepteurs de l'instant, leur statut de récepteurs.

UNT (non émetteur) ôte à tous les émetteurs de la boucle leur statut d'émetteur en envoyant le message de NON-émetteur (suivi de RFC).

Jean BOSCHAT (+rS)

STO D

CHAINE pour STO d

Rassurez-vous, je ne vais pas exposer le système qui permet de manipuler les 56 DRAPeaux de notre '41', une prose abondante a déjà été écrite à ce sujet et les débutants pourront s'y reporter utilement. Le PRGM "STOd" n'utilise pas de PS et ne nécessite que le XF (utilisation d'une IMPrimante fortement déconseillée...). Il permet d'obtenir en ALPHA, la CHAINE correspondant aux DRAPeaux que l'on désire lever, CHaîne qui, suivie de RCL M et de STO d, placera les 56 DRAPeaux dans la configuration désirée.

La CHaîne précitée n'est pas introduite dans le PRGM (ou dans un PRGM 'spécial') mais j'invite ceux qui connaissent la "combine", à écrire un article sur la méthode adéquate. MERCI à ceux qui vont se dévouer !

Description succinte du PRGM:

Signalons tout de suite aux gens pressés que ce PRGM est relativement lent et cela est dû principalement au déroulement des boucles et également parcequ'il utilise une routine écrite en "utilisateur", routine "b-D" de Bruno TREDEZ (MR N1P31). Ceux qui disposent d'une Fonction en MC qui fait la conversion BIN-DEC, pourront l'utiliser à la place des Pas 52 à 34. La lenteur d'exécution est compensée par un affichage démonstratif qui permet de suivre les différentes "créations".

La 1ère boucle (Pas 08 à 15) permet d'introduire successivement les DRAPeaux que l'on désire lever. Il est possible d'introduire ces DRAP dans un ordre quelconque, il suffit simplement de ne pas en oublier (!). Lorsqu'ils sont tous introduits, il suffit de taper simplement 'R/S' : le TEST du Pas 14 nous fait alors sortir de la boucle. Les REG 00 à 55 sont utilisés pour STO une valeur différente de zéro (cf CLRG du Pas 07) lorsque le DRAP correspondant est indiqué (levé). Pour une question de simplification, je STO le N° du DRAP..... Dans ce cas particulier, il aurait été possible de ne pas placer de 'CF 22' (Pas 10) mais je vous déconseille vivement de prendre cette habitude (!).

La 2ème boucle (Pas 24 à 32) RCL successivement en 'x' les valeurs contenues dans les REG 00 à 55 et place en ALPHA un '1' ou un '0' en effectuant une comparaison avec zéro. Les REG sont rappelés par paquets de 8, ce qui place en ALPHA, la valeur BIN qui ne reste qu'à convertir en DEC (MERCI Bruno !)et à STO successivement dans les REG 56 à 62 (Pas 53 avec 16, 17 & 54).

Les Pas 55 à 57 incrémentent l'INCrément de la 2ème boucle de façon à récupérer le 'paquet' de REG suivant.

Le DSE 64 du Pas 58 est utilisé en TEST conditionnel et permet de parcourir 7 fois la "grande boucle" (Pas 21 à 59), 7 étant le nbre de codes correspondant aux Octets du REG d.

La boucle des Pas 63 à 79 permet de récupérer les valeurs DEC STO dans les REG 56 à 62 de façon à placer en ALPHA les codes correspondants. Signalons qu'il n'est pas possible d'introduire un CARactère NUL en tête du

REG ALPHA et au moyen de XTOA. Dans le cas où aucun DRAP de 00 à 07 ne devrait être levé, j'ai adopté de lever impérativement le DRAP 03 et d'indiquer cette particularité. Signalons également que la fonction logique 'ET' pour: s'il s'agit du 1er CAR 'ET' si la valeur est zéro, ... adopter la valeur 16 et lever le DRAP 03, est

bien expliquée dans le livre sur la '41' de Philippe DESCAMPS et Jean-Jacques DHENIN dont une lecture attentive peut (°) être riche d'enseignements.
(°) "Quiconque a beaucoup lu peut avoir beaucoup retenu".

Je n'ai pas cherché à optimiser ce PRGM de façon à ce qu'il soit plus facile à 'décortiquer' mais il est plus que probable que des collègues vont s'y atteler ainsi que pour le compléter comme précité.

Il y a d'autres procédés pour arriver au même résultat et je pense que l'utilité indéniable de ce PRGM va vous inciter à plancher sur la question.....

Nous verrons *ultérieurement comment on peut simplifier ce PRGM (et l'accélérer) en utilisant des Fonctions de la ROM "PANAME" écrite entièrement en MC par Jean-Jacques DHENIN et Stéphane BARIZIEN. Cette ROM n'étant pas encore commercialisée au moment où j'écris ces lignes, je ne désire pas vous mettre trop "l'eau-à-la-bouche". Je peux tout de même vous indiquer que rien que les Fonctions permettant de travailler sur des Tableaux, justifient presque à elles seules, l'acquisition de cette "petite merveille". Je dois bien cette pub gratuite à nos charmants collègues qui ont permis à la France de se hisser dans le peloton de tête de la PRGMmation MC. N'imaginez tout de même pas que j'ai écrit ces dernières lignes en écoutant l'air de la marseillaise.....*

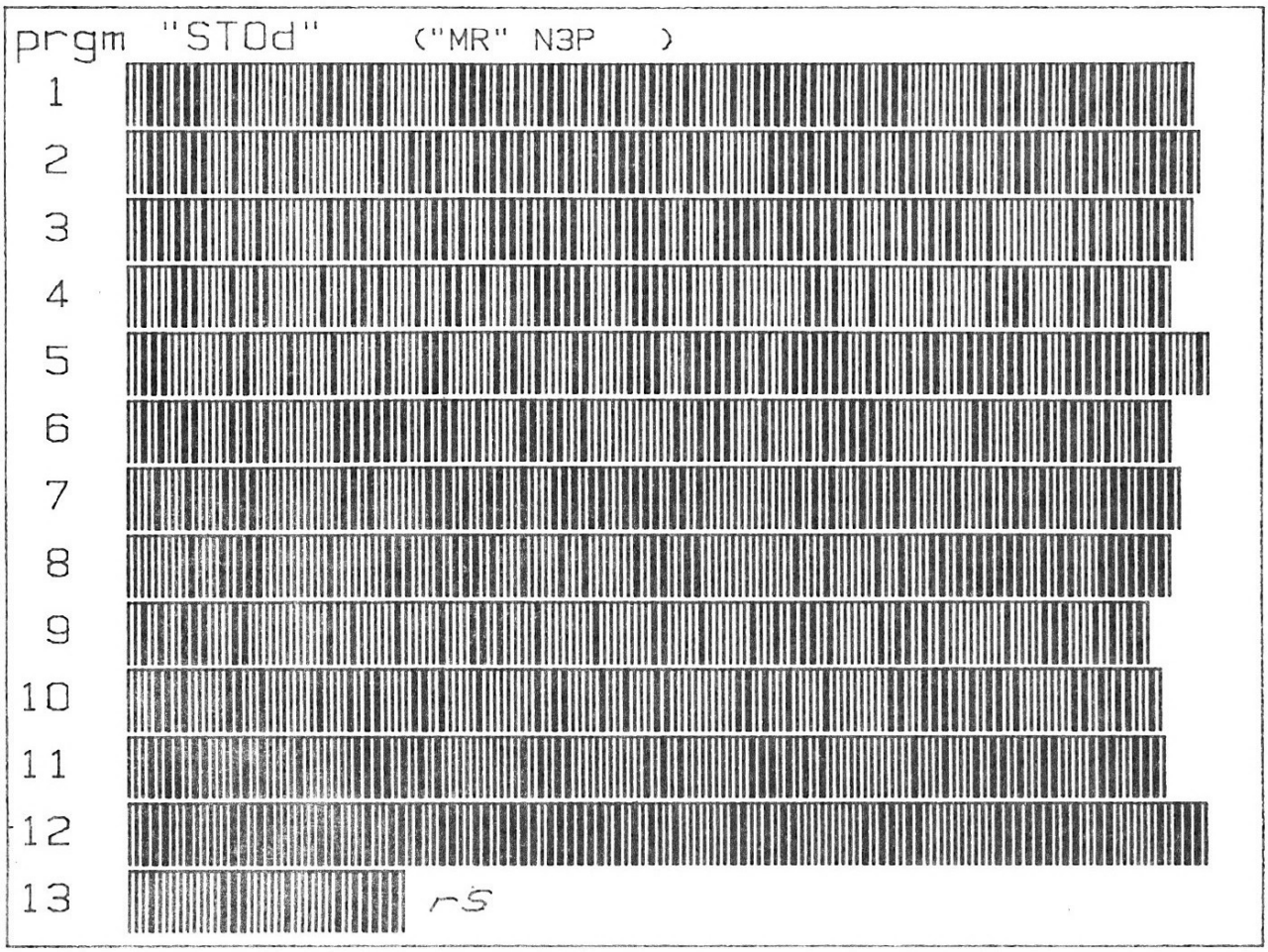
Bonnes et heureuses PRGMmations, *rS (T178)*

```
 01*LBL "STO
d"
 02 FIX 0
 03 CF 29
 04 SF 01
 05 66
 06 PSIZE
 07 CLRG
 08*LBL 00
 09 "SF..."
 10 CF 22
 11 PROMPT
 12 FS? 22
 13 STO IND
X
 14 FS?C 22
 15 GTO 00
 16 56,1
 17 STO 63
 18 7
 19 STO 64
 20 ,007
 21*LBL 04
 22 STO 65
 23 CLA
 24*LBL 01
 25 RCL IND
65
 26 X≠0?
 27 "⊢1"
 28 X=0?
 29 "⊢0"
 30 AVIEW
 31 ISG 65
 32 GTO 01
 33 2
 34*LBL "b-D
"
 35 ,
 36*LBL 02
 37 ATOX
 38 X=0?
 39 GTO 03
 40 55
 41 -
 42 7
 43 X<Y?
 44 CLX
 45 +
 46 X<>Y
 47 R↑
 48 *
 49 +
 50 GTO 02
 51*LBL 03
 52 +
 53 STO IND
63
 54 ISG 63
 55 RCL 65
 56 ,008
 57 +
 58 DSE 64
 59 GTO 04
 60 CLA
 61 56,062
 62 STO 00
 63*LBL 05
 64 RCL IND
00
 65 X=0?
 66 FC? 01
 67 FS? 30
 68 SF 00
 69 FS? 00
 70 SF 03
 71 FS?C 00
 72 16
 73 VIEW X
 74 PSE
 75 XTOA
 76 AVIEW
 77 CF 01
 78 ISG 00
 79 GTO 05
 80 AVIEW
 81 END
```

CAT 1

```
LBL'STOd
LBL'b-D
END          157 BYTES
.END.        04 BYTES
```

1 carte magnétique



PURDIR

Cher JDD,

Tout d'abord ma petite contribution au club. Après avoir potassé les albums de PPC-T, je n'ai rien trouvé d'approchant sauf erreur. Suivant la nomenclature HP, je l'ai baptisée PURDIR (PURge DIRectory).

La routine s'applique au module XF et se liste comme suit:

```
01 LBL "PURDIR
02 "§§§§§§§§§§§§§
03 X<>N
04 X<>c
05 X<>M
06 STO 00
07 X<>M
08 X<>c
09 X<>N
10 CLA
11 END
```

B. JOUSSE (T 502)
BARZUN
64530 PONTACQ

Tel : (D) (59) 53 64 27
(P) (59) 32 71 39

Résultat de la routine: le registre alpha est effacé et accessoirement un EMDIR donnera DIR EMPTY.

Ligne 02: (décimal)
254/11/224/1/105/11/224/1/255/255/255/255/255/255/245

MODULE GRAPHIQUE

28/3/84

-MODULE GRAPHIQUE

(Suite de l'article de PPC-T N9P25 n'ayant pû passé en entier)
Certains programmes contiennent des caractères que nous nommons synthétiques : des parenthèses dans "X?" et "Y?" du module, et des guillemets dans les programmes d'application "XBC" et "ABC" qu'il faut donc rentrer "obligatoirement" en codes-barres.
Pour les C.B. Data voici les correspondances des demis octets non numériques (voir PPC-T N7P16) 10 : nul (bouche-trou), 11 : virgule, 12 : + (à quoi ça sert ?), 13 : - (moins) 14 : E (exposant), 15 : illégal (d'oú ERROR).
Un C.B. Data doit faire 3 octets minimum au total.

Pour les C.B. d'exécution et de branchement d'un programme il n'est pas nécessaire de coder le nombre de caractères du nom, par exemple pour XEQ'AB : octet de contrôle (checksum), octet indicatif d'exécution directe 40_{16}, octet XEQ $1E_{16}$, et caractères A 41_{16}, B 42_{16}.

Un tableau du manuel indique comment calculer l'octet de touche (le 4ème) pour les assignations; la formule donnée est bien lourde, et fausse pour les touches shiftées. En fait le numéro de ligne de la touche forme les 4 bits de gauche, et le numéro de colonne, les 4 bits de droite; pour les touches shiftées il faut ajouter 8 au numéro de ligne, et MOD 16. Exemples :

touche	code	octet N°4	valeur décimale
EEX	43	0100,0011	67
R/S	84	1000,0100	132
☐ EEX	-43	1100,0011	195
☐ R/S	-84	0000,0100	4

Il est ainsi possible d'assigner à la touche Shift, shiftée ou non, mais ça n'est pas dit dans le manuel!

ASN X=Y? à la touche Shift

ASN BEEP à la touche Shift shiftée

efface ASN à Shift

efface ASN à Shift shiftée

Je n'avais pas trouvé les codes pour faire GTO.. en C.B. , et ceux donnés dans le manuel m'ont surpris : octet de contrôle, octet indicatif 64_{10}, 1_{10}(=GTO. ?), 15_{10}, 255_{10} !

GTO..

TREDEZ b. T 120

SAUTS RELATIFS

Il existe plusieurs types de sauts et nous allons pour l'instant nous contenter d'étudier ceux qui sont d'une longueur limitée mais qui peuvent s'effectuer à n'importe quel endroit à l'intéreir d'une Page.

Ces sauts sont commandés par des instructions codées sur 1 seul Mot. Les Bits correspondants peuvent être représentés de la façon suivante :

S1 1111 1ctt (lire '1' comme Lima)

où 'S1 1111 1' indique, en mode Complément à 2, la longueur dusaut et où 'S' : Bit de poids fort (1er Bit, Bit de tête ou Bit de Signe) indique le sens de déplacement :

0 = (Signe '+') = vers le bas ou en avant,

1 = (Signe '-') = vers le haut ou en arrière;

'c' est le Bit qui indique dans quelle situation doit se trouver le REG Carry pour que l'instruction soit exécutée; 'tt' spécifie, comme nous le savons, le type d'instruction qui est le type 3 = BIN 11.

*Exemples: 1°) Pour sauter de l'adresse 9A25 à l'adresse 9428 (+3 = saut de 2 Mots ou, au 3ème Mot en partant vers le bas ou, en avant) SI le Carry n'est PAS levé (9A25 GONC *+03 ou, en MNEMO PPC: 9A25 JNC *+03), l'instruction sera :*

BIN: 00 0001 1011 *(Carry)*

DEC: 3

HEX: 0 1 B

L'instruction placée à l'adresse 9A25 sera donc, en Mode '2 4 4': 01B .

*2°) Pour sauter de l'adresse 9A25 à l'adresse 9A22 (-3 = saut de 2 Mots ou, au 3ème Mot en partant vers le haut ou, en arrière) SI le Carry EST levé (9A25 GOC *-03 ou, en MNEMO PPC: 9A25 JC *-03), l'instruction sera :*

BIN: 11 1110 1111 *(Carry)*

DEC: -3

HEX: 3 E F

La valeur représentée par les 7 premiers Bits est bien DEC -3 mais en Mode Complément à 2; ce qui nous donne DEC 125 en Mode Non Signé. Pour simplifier, nous pouvons indiquer que dans le cas d'un saut négatif, c'est le Complément à 128 (DEC) qui est considéré : 128-125= 3 ou : 125-128= -3 .

Dans le cas d'un saut positif, la saturation des 6 Bits qui suivent le Bit de signe ne permettra qu'un saut d'une longueur maximale de DEC +63 =BIN 01 1111 1 .

Pour un saut négatif et toujours en Mode Complément à 2, le maximum codable sur 7 Bits sera DEC -64 = BIN 10 0000 0 et le minimum sera -1 = BIN 11 1111 1 = DEC 127 en Mode Non Signé.

Mode Non Signé (UNSGN) En BIN, c'est la représentation d'une valeur sans Bit de signe.

Mode Complément à 1 (1'S) Dans ce Mode ainsi que dans le suivant, le 1er Bit (à gauche) est le Bit de signe :

0 = '+' (plus)

1 = '-' (moins).

Le Complément à 1 d'une valeur représentée en BIN est l'équivalent du 'NOT' logique. Tous les Bits à '1' sont mis à '0' et tous les Bits à '0' sont mis à '1', ce qui effectue le changement de signe.

```
BIN:   0001 1101   (sur 8 Bits)
HEX:      1    D
DEC:      029      (le zéro peut être omis)
CHS ou NOT :
BIN:   1110 0010
HEX:      E    2
DEC:     -029      (le zéro peut être omis)
```

Mode Complément à 2 (2'S) Le Complément à 2 en BIN est en fait un NOT + 1 (plus un) : on effectue le Complément à 1 (ou un NOT (cf ci-avant)) et on ajoute 1 au résultat obtenu .

```
DEC:      087
BIN:   0101 0111   (Mode Non Signé)
NOT :
BIN    1010 1000   (     d°       )
1 + :
BIN:   1010 1001   (     d°       )
HEX:      A    9
DEC:      169      (     d°       ) (256-169=87)
BIN:   1010 1001   (Mode Complément à 2)
          -087     (        d°          )
BIN:    010 1001   (sans le Bit de signe)
DEC:      041      (128-41=87)
```

Ces quelques indications n'ont pas la prétention de vous initier aux différents types de notations utilisées pour aller 'AU FOND de la 41' et je me permets de lancer un appel aux professeurs qui sont assez nombreux parmi les adhérents afin qu'il y en soit au moins un qui se dévoue pour nous initier à ces connaissances indispensables (y compris les fonctions logiques de base: NOT, AND, OR & XOR ou, pour les anglophobes (dont je fais partie) : NON, ET, OU & OU exclusif.

Je me permets également de m'adresser aux quelques personnes qui sont heureusement fort peu nombreuses (?) et qui critiquent le contenu des "papiers" que j'écris et ceci quelquefois, avec un manque total de tact élémentaire. Je ferais remarquer que cette facile critique a toujours été émise par des adhérents dont on cherche vainement le nom à la suite du moindr'article ou PRGM. Je pense qu'avant d'avoir le droit de critiquer, il faudrait commencer par montrer ce que l'on sait faire et, à partir de ce moment là, je crois qu'on devrait devenir plus indulgent vis-à-vis du travail des autres. Il est vrai que "la modestie est une fantaisie que seuls les grands peuvent se permettre"..... et que nous rencontrons de plus en plus de personnes qui sont beaucoup plus exigeantes des autres qu'elles ne le sont d'elles-même. Et ne croyez surtout pas que j'ai une haute opinion de ce que j'écris. Je sais bien au contraire qu'on peut faire beaucoup mieux mais il faut pour cela y consacrer beaucoup plus de temps et avoir des dons et des connaissances que je n'ai pas. Comme je l'ai déjà écrit, seule le bonne volonté doit être prise en compte et, si vous en avez assez de lire (ou de trouver dans le Journal) ma mauvaise prose, vous pouvez me le dire ou me l'écrire gentiment, j'en tiendrais compte mais je tiens tout de même à remercier tous ceux qui m'ont encouragé à continuer.....

Bonnes PRGMmations à tous, *hS* (T178)

CODES - BARRES

Robert Pulleart (T46H) Lusterlaan 31, 2665 TH Bleiswijk Pays-Bas. suite PPC-T#11 P52

Quant au problème de Philippe GIRARDET, concernant le code-barres, il devrait trouver toutes les réponses à ses questions dans le manuel "Creating your own bar code", édité par Hewlett-Packard (ou encore, sous forme très condensée, dans "The HP-41 Synthetic Quick Reference Guide", livret de Jeremy SMITH... vous ne l'avez pas en librairie, tsss... il ne me quitte jamais: en papier plastifié, vous pouvez même faire de la programmation synthétique en plongée! ou dans votre bain... bref, l'adresse: 226 24th Place, COSTA MESA, CA 92626, USA). Si, pour l'une ou l'autre raison, Philippe GIRARDET a des difficultés à obtenir ou à consulter ces ouvrages, je peux lui fournir les quelques renseignements que voici (et éventuellement d'autres si cela l'intéresse).

Une ligne de programme comporte au maximum 16 octets, dont les trois premiers constituent l'en-tête: les 13 octets suivants contiennent le code programme proprement dit, que je laisserai de côté ici, pour m'attarder plutôt sur cet en-tête.

- Le premier octet est le nombre de contrôle, "CHECKSUM".
- Le ½ octet suivant indique le type de code-barre auquel on a affaire:
  soit 1 (0001) pour un programme non privé,
  soit 2 (0010) pour un programme privé.
- Le ½ octet suivant est l'ordre séquentiel de la ligne (modulo 16) la première ligne étant 0, la deuxième 1, etc...
- Le ½ octet d'après indique la longueur, en octets, de la toute première instruction de la ligne en question, dans le cas où cette instruction est incomplète, tronquée, avec une partie (le début) codée en fin de ligne précédente.
- Le dernier ½ octet de cet en-tête, enfin, donne la longueur, en octets toujours, de la dernière instruction, lorsque celle-ci est, là aussi, tronquée et poursuivie sur la ligne suivante.

ASN X=Y? à la touche Shift

ASN BEEP à la touche Shift shiftée

Dans ces deux derniers cas, la longueur à considérer n'est pas celle de l'instruction complète, à cheval sur deux lignes, mais seulement la longueur de la partie de cette instruction qui se trouve sur la ligne en question.

J'espère que ces quelques informations seront utiles à Philippe GIRARDET (à moins qu'un autre membre ne lui fournisse des rensiegnements plus détaillés...) S'il envisage d'introduire des instructions synthétiques au moyen de code-barres, il n'est cependant pas au bout de ses peines! Avec les code-barres de type 4 (hex40 en deuxième octet, tout de suite après le "CHECKSUM") il est possible d'introduire directement dans un programme des instructions complètes telles "RCL 03", "STO IND 29", etc... (une table des codes

d'instructions hexadécimaux est indispensable). Hélas! cela ne marche absolument pas pour des instructions synthétiques aussi banales que "RCL M", ou "STO a",... J'ognore pourquoi, mais force est de constater que le lecteur optique les refuse catégoriquement, et donne un bip désapprobateur sans même avoir la politesse d'afficher un message d'erreur!

Une solution est le "Synthetic Programming Paper Keyboard", publié dans le journal DATAFILE (Vol.2, No.3 de juin/juillet/août 1983) de PPC-UK. A vrai dire, je ne l'ai pas essayée, car j'ai ma propre méthode, qui me paraît moins lourde et moins risquée (sans doute parce que je la connais mieux...), et qui ne fait appel à aucune fonction synthétique (pas besoin de "Byte trap", "Q loader" et autres fioritures dans le genre).

Si cela peut vous intéresser, et rendre service à Philippe GIRARDET, voici comment je procède:

- Tout d'abord, chaque instruction synthétique utile est codée sous forme de programme incomplet, sans label ni END, tel que (en hexadécimal)

kk 10 00 ii ...

kk : checksum

ii ... : code hex de l'instruction (un ou plusieurs octets, éventuellement plusieurs instructions).

Par exemple, pour RCL M, ce sera: 16 10 00 90 75 (toujours en hexadécimal, ce qui facilite considérablement le traçage du code-barres correspondant, quand on le fait manuellement!) Dans cet exemple, le code-barres sera finalement:

- La procédure pour introduire une telle "instruction" dans un programme est la suivante, la seule restriction étant que le programme en question doit être de tout dernier en mémoire.

Pour clarifier un peu les choses, nous appellerons "N" l'instruction synthétique à introduire dans le programme, à la suite d'une instruction quelconque "M" non synthétique.

(1) Introduire le programme jusqu'à l'instruction M incluse.
(2) Faire GTO..
(3) Lire le code-barres de l'instruction N. Affichage: "W: RDY Ø2"
(4) Faire [←]
(5) Sortir du mode programme
(6) Faire GTO "nom du programme"
(7) Revenir en mode programme
(8) Faire [BST] . Affichage: "END"
(9) Faire [←] pour éliminer ce END
(10) Faire [SST] : l'instruction N synthétique est affichée. On vérifier qu'elle suit immédiatement l'instruction M.
(11) Poursuivre l'introduction du programme normalement.

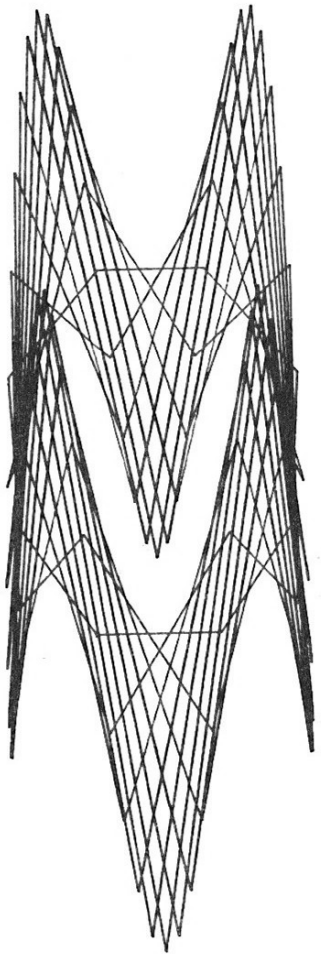
Si l'on a plusieurs instructions synthétiques consécutives à introduire, il suffit de répéter les étapes (2)(3) et (4) pour chacune d'elles, successivement, puis éliminer tous les END qui les séparent, les uns après les autres.

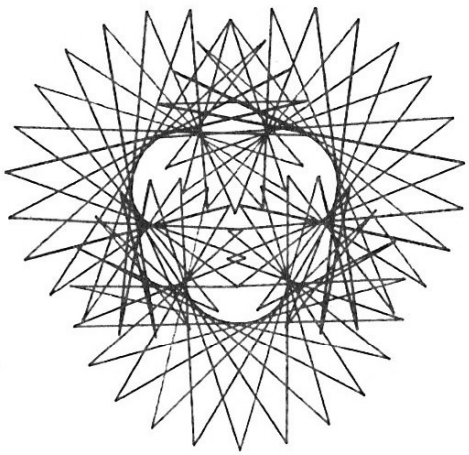
Par rapport aux autres méthodes d'introduction d'instructions synthétiques, celle-ci présente bien moins de risques de "plantage". Son inconvénient majeur est que chaque instruction synthétique doit être codée comme indiqué plus ahut, et si on désire avoir tout l'assortiment, ce n'est pas une mince affaire (ma propre collection est assez limitée)...

Je voudrais faire une dernière remarque, concernant l'article de Philippe GIRARDET: l'octet de contrôle, ou "CHECKSUM", n'est pas calculé modulo 255! La méthode exacte est celle du "end-around carry", c'est-à-dire que lorsque la somme dépasse 255 (hex FF) la retenue est elle-même ajoutée au total. C'est ainsi que dans mon exemple de "RCL M", plus haut, on obtient, en hexadécimal:

10 + 00 + 90 + 75 =: 115 + 1 =: 16

Pour en terminer avec le code-barres, je voudrais ajouter que je ne suis pas tout à fait d'accord avec votre opinion en la matière (cf. votre lettre à David MEYER, bas de la p.10 de PPC-T No.10): il est vrai qu'établir soi-même le code-barres de ses programmes est lent et fastidieux, ou bien nécessite du matériel coûteux, ou encore fait appel au bricolage (un des membres de la section PPC-Delft a réalisé un tel bricolage, matériel et logiciel, pour obtenir d'excellents résultats avec une imprimante du commerce - NEC, je crois - connectée à une HP-41). La qualité des code-barres imprimés dans PPC-US s'améliore constamment, et ceux qui accompagnent la plupart des programmes de "DATAFILE" (PPC-UK) sont en général parfaitement "lisibles". Mais le code-barres, ce peut être bien autre chose, avec un peu d'imagination, et la fonction "WNDSCN" permet de créer ses propres codes, simples et utiles dans des situations spécifiques. J'ai plusieurs applications de cet ordre, et éventuellement, à titre de curiosité, je peux vous proposer l'une ou l'autre, sous forme de petit article.





bT120

LILLE
AMIENS
ROUEN
METZ
CAEN
CHALONS
PARIS
STASBOURG
RENNES
ORLEANS
NANTES
DIJON
BESANCON
POITIER
LIMOGES
CLERMONT-
FERRAND
LYON
BORDEAUX
TOULOUSE
MONTPELLIER
MARSEILLE

NOP

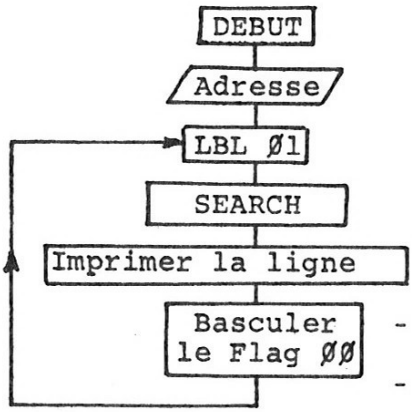
La chasse aux "NOP"

```
01♦LBL "CHERCH"
02 CLEAR
03 "DEB"
04 aVIEW
05 4
06 HEXPR
07 DECODE
08 -11
09 AROT
10 SF 00
11 CODE
12♦LBL 01
13 SEARCH
14 DECODE
15 PRA
16 FS?C 00
17 FS? 30
18 SF 00
19 GTO 01
20 END
```

```
-TIME 2C
-CX TIME
-PRINTER-
-MLEPROM 1H
-ADV PRT 3C
AID
ID
FINDAID
OUTAX
OUTCR
OUTLF
OUTLFX
OUTSPX
OUTXB
OUTYBX
OUTa
OUTaX
RCLSEL
-82163 FCNS
CLEAR
CLEARO
CSRDN
CSRHX
CSRL
CSROFF
CSRON
CSRR
CSRVX
CSRUP
CTYPE
HOME
SCRLDN
SCRLUP
SCRLX
XYTPP
```

Dans le MLDL, il est fréquent que l'utilisateur CHERCHE à connaître les zones contenant des "NOP" (codes ØØØ). Nous nous sommes proposés un exercice pratique dont l'objet est de restituer l'adresse du début de chaque bloc de NOP et l'Add de fin de bloc.

Traçons l'organigramme de notre travail :

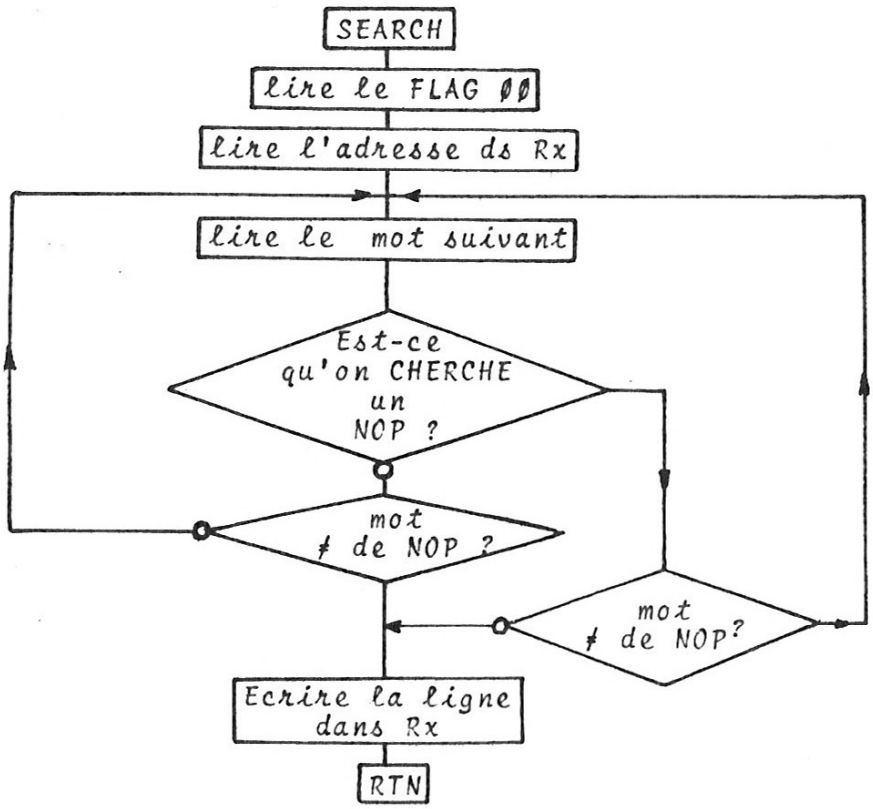


de début de recherche

la fonction "SEARCH" sera écrite en 'MC'

- *si le Flag ØØ = Ø : CHERCHER la 1ère ligne ≠NOP*
- *si le Flag ØØ = 1 : CHERCHER la 1ère ligne NOP*

PRGM utilisateur destiné à lancer la reCHERCHE et à imprimer les lignes : (ci-contre)



Organigramme du PRGM 'MC' de reCHERCHE de mots ***NOP*** *ou* ***NON NOP.***

-82162 FCNS
CLBUF
8BIT
ESCAPE
PARSE
STATUS
TABCOL
UNPARSE
-82905 FCNS
BELL
CHARSET
FFEED
FORMLEN
GRAPHX
MODE
SKIPOFF
SKIPON
TEXTLEN
VSPAC
-PLOT FCNS
AXIS
BACKSP
BACKSPX
BOX
COLOR
*CSIZE
*DRAW
*HOME
*LABEL
*LDIR
*LTYPE
*MOVE
*PLREGX
RDRAW
RESET
REVLF
REVLFX
RMOVE
SETORG
-UTILITIES
/MOD
AD-LC
ALENG
ANUM
ANUMDEL
APPX
AROT
ATOXL
ATOXR
ATOXX
BLDPT
BRKPT
CHFLAG
CLINC
COLPT
DECX
INCX
I/DINDX
GETRGX
LC-AD
LINPT
OUT

Ecriture des PRGM :

EXECUTEZ	Affichage	Commentaires
[XEQ] "N"	ORG ___	Entrez les 3 digits de droite de l'Add de la FCN dans MLI
NNN (ex: 090)		
	NOM? ALPHA	Entrez le NOM de la FCN à écrire
SEARCH	SEARCH_ ALPHA	
[R/S]	SEARCH,P0900	La '41' affiche après avoir exécuté l'écriture dans le MLI : SEARCH (NOM) "virgule" (séparat) P Page du MLI 090 Add d'entrée de la FCN 0 type de FCN (cf Manuel du ML-1H)

---

[6] [RTN]	'AS	Entrez l'Add de début d'écriture
ici: 090	ORG ___	
[RCL]	C=REGN __	Rappel du REG des drapeaux (14_{10}=d)
14		
[R]	RCR __	Positionnement du REG pour STO des Flags (cf "AF/JDD")
12		

à ce stade: C= | 8 ... 55 | 0 ... 7 |
← Flags 8 à 55 → | F 7 à 0 |
Flags utilisateurs.

[0]	TYPE0	L'instruction ST=C n'est pas directemt accessible au clav. Ns la recherchons ds un fichier en XM
ST=C		
[R/S]	ST=C	

à ce stade, le CPU contient dans son propre REG de drapeaux (ST), la copie du dernier Octet du REG d des Flags utilisateur.

[C] affichage: [C=C +1 -1 +C]
touches: [C] [D] [E]

La '41' attend la précision sur l'instruction contenant l'accumulateur C
C ici correspond à la touche +1

[C] affichage: [WPT X M S W]
touches: [A] [B] [C] [D] [E]

[C]

C ici indique que +1 s'applique à la zone MANT de l'accu

C=C+1 M

L'instruction est prise en compte.

```
POSA
PSIZE
RG
RG+-
RG*
RG/
RG+Y
RG*Y
RG/Y
RGAX
RGCOPY
RGINIT
RGNb
RGORD
RGXTR
RGSUM
RGVIEW
```

[Ø]	TYPE Ø ALPHA	
CX ISA		Recherche ds le FICHier le code de l'instruction CX ISA
[R/S]		
[6] [9]	?Sd=1 __	Instruction de test de l'1 des DRAPeaux de ST.
Ø7	?Sd=1 Ø7	Interroge l'état du Flag ØØ utilisateur.
[+]	GONC *+ __	Saut sans condition.
[←]	GOC *+ __	[BSP] transforme la demande de saut en saut conditionnel.
Ø4		Longueur du saut à effectuer.
[6] [÷]	A B C ≠Ø	Instruction de comparaison avec Ø d'1 REG du CPU.
touches:	[A][B][C]	
[C]		Choisit l'accu C
	WP' X M S W	Attend la définition du champ.
touches:	[B][C][D][E]	
[B]	?C≠Ø X	Compare le MOT de ligne lue avec Ø(NOP)

```
C08A 088 H
C08B 003 C
C08C 012 R
C08D 001 A
C08E 005 E
C08F 013 S
C090 3B8 8 C=REGN 14(d)   <--
C091 37C µ RCR    12
C092 358 Γ ST=C
C093 0F8 Γ C=REGN 3 (X)
C094 23A : C=C+1  M
C095 330 0 CXISA
C096 28C L ?S7=1
C097 027 ' GOC    *+04
C098 2E6 ♦ ?C#0   X
C099 027 ' GOC    *+04
C09A 3D3 c GONC   *-06
C09B 2E6 ♦ ?C#0   X
C09C 3C7 ' GOC    *-08
C09D 0F8 Γ REGN=C 3 (X)
C09E 3E0 ⊢ RTN
```

Pour les lignes suivantes, on procède comme ci-dessus :

	GOC +Ø4	
	GONC -Ø6	
	?C≠Ø X	
	GOC -Ø8	
[STO]	REGN =C __	Chargement d'1 REG RAM avec le contenu de l'accu C.
Ø3	REGN =C Ø3	Détermine l'écriture dans Rx .
[6] [C]	CON ___	L'introduction d'1 MOT par son code.
3 E Ø		Code du MOT 'RTN'

à vous de jouer ! J.J.D.

Jean-Jacques DHENIN (T212)

```
SAVERGX
SIZE?
SORT
STO>L
SUB$
TF55
VKEYS
X=NN?
X≠NN?
X<NN?
X<=NN?
X>NN?
X>=NN?
X<>F
XTOAL
XTOAR
Y/N
YTOAX
```

Vulgarisation du 'MC' rS (T178)

L'article ci-avant est bien de JJD et je me suis contenté de le taper à la machine. J'espère ne pas avoir introduit trop d'erreurs (?).

Jean-Jacques a proposé de donner des cours de MC à ceux qui désireraient s'initier à cette merveilleuse et puissante technique. Ces cours ont lieu pour l'instant à son domicile et les personnes intéressées devront avoir la gentillesse de s'adresser à moi (1-848 15 96 = DOMicile) afin de ne pas déranger JJD inutilement. Votre serviteur a été parmi les premiers à souscrire à cette appréciable initiative.

C'est donc à la suite de la première réunion que l'article précité a été écrit et nous avons pensé que cela pouvait vous intéresser.

Vous constaterez que la PRGMmation en MC nécessite en premier lieu, une étude sérieuse du problème à résoudre. Lorsque tout est convenablement préparé, l'écriture des MOTs est très facile grâce au merveilleux PRGM d'ASSEMB de Stéphane BARIZIEN

```
                XEQ "CHERCH"
DEB 3000
3002000
3003097
3014000
3015088
303E000
303F0B1
3062000
30630AC
3064000
306509D
306A000
306B0E3
3072000
30730F7
3080000
3082A4
```

(version "améliorée" par 'ΛS' et utilisée avec une '41CX').....

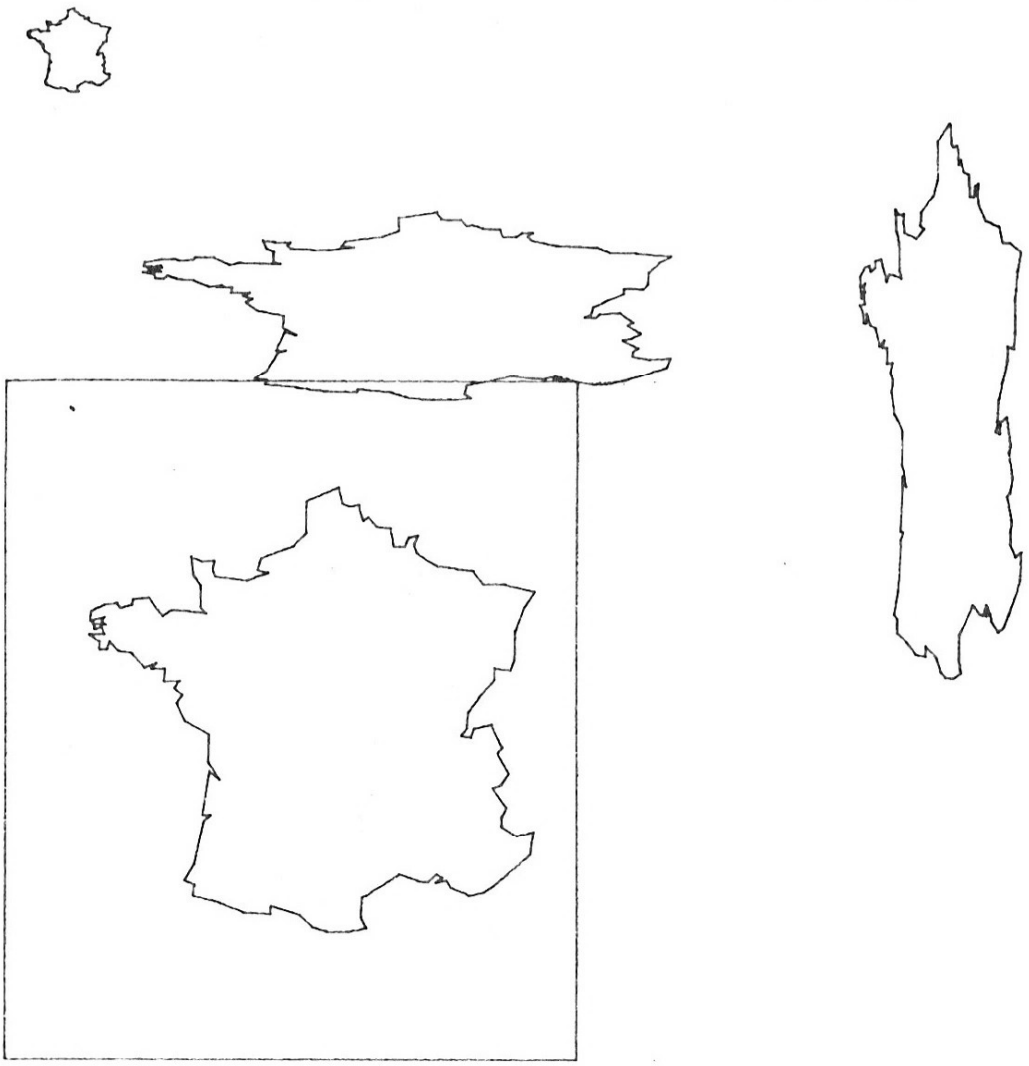
et JJD. Signalons que la ligne qui vient d'être créee est affichée mais nous ne l'avons pas mentionnée dans le texte. Il est utile, au début, d'utiliser une grille de clavier et le PRGM d'ASS fourni avec le MLDL de la SCIP et le jeu d'EPROM 'ML-1H', utilise les MNEMOniques 'HP' (qui ont ma préférence).

Les PRGM publiés, malgré leur simplicité apparente, peuvent être très riches d'enseignements pour ceux qui se donneront la peine de les étudier.

Ecrivez-moi si vous êtes intéressés par ce genre d'article car nous ne voudrions pas perdre notre temps. J'indiquerais simplement à ceux qui hésitent à se "lancer" dans le MC, qu'ils ont tort car ils se privent d'un moyen de PRGMmation puissant et passionnant. VIVE le MC !

P.S.: La F CLEAR (Pas Ø2 du PRGM "CHERCH") est facultative (?) et est issue de la ROM 'PANAME' écrite par..... JJD et StB, ROM de 8kO qui va être commercialisée par la SCIP (ROM fabriquée par HP). Vous trouverez le listing des Fonctions de cette ROM, joint (intercalé=excuses!) aux pages précédentes. Je vous assure que c'est absolument fantastique ce qu'on peut faire avec ces Fonctions. Nous ne manquerons pas de vous donner des indications sur les possibilités offertes par cette création française.....

Le 28.04.84 ΛS (T178)



DIVERS

J'envisage d'utiliser mon micro ordinateur pour lister mes programmes de HP-41 par l'intermédiaire du beeper. Je n'ai pu encore tester la faisabilité de la chose mais j'ai déjà réalisé ce petit programme qui décode le registre X et envoie tout cela dans le beeper. Pour les amateurs, j'en fais part au club.

```
3F8   C=REG 3(X)
0E0   SEL Q
39C   PT = 0
0A0   SEL P
39C   PT = 0
      [illegible]
      NOP
2FC   RCR 13
358   ST = C
000   NOP
00C   ?S3 = 1
01B   GONC *+03
0EE   BC EX W
02B   GONC *+05
0EE   BC EX W
130   LDI
010   CON 0010
01B   GONC *+03
130   LDI
020   CON 0020
0AE   AC EX W
3C4   CLR ST
2D8   FEXSB
130   LDI
0FF   CON 0255
358   ST=C
0AE   AC EX W
2D8   FEXSB
266   C=C-1 X
3F3   GONC *-02
0EE   BC EX W
33C   RCR 1
1EE   C=C+C W
3DC   INC PT
114   ?PT=8
323   GONC *-1C
33C   PT=0
0E0   SEL Q
3DC   INC PT
294   ?PT=7
360   RTN C
0A0   SEL P
2EB   GONC *-23
```

Si quelqu'un est intéressé par la routine de décodage d'un registre et envoi sur beeper, qu'il me contacte ou qu'il me rende une visite ; l'endroit est assez agréable !

x<>Y [signature]

J. P MAGNAN
Ecole du Nord
Labourdonnais MAPOU
Mauritius

Routines diverses :

1) PHP : etant donnée une adresse hexa, PHP permet de retrouver la page du microcode HP (inverse de "P8) (MLDL necessaire ou MLDE)

```
LBL T PHP
 FIX 0
 "ADR"  AVIEW
 4  HEX PR
 DECODE  ASHF  H-D
 1024  MOD  OCT
 T252.29.60.88.120.148.161.181.209.240.1.2.3"
 x<>Y  AROT  ATOX
 5  x<>Y  x>y?  GTO 01
 "242.2.30.63"
 1-  AROT  ATOX
   270 +
LBL 01
  "PAGE"  ARCL X
        R↑  PROMPT  END.
```

Utilisation   XEQ TPHP   ADR 1 ↑ A 0 (_ _ _ _)   PAGE 161
[R/S]
1640 (octal).

3) La routine 26C5 en ROM interne est très interessante (ASEARCH) Il est dommage que "ENTRY?" qui l'utilise n'ait conservé tous ses avantages.
Elle donne en effet en sortie non seulement l'adresse d'une

fonction en ROM mais aussi

- son code XROM si c'est un module :

ex : CHERCHE "TIME" E01580|A6|9 0|50EE|
XROM| adresse|

- le code de la fonction si c'est une ROM interne :

CHERCHE "AON" 00 6000|04|8C|133C
ROM interne|code hexa| adresse

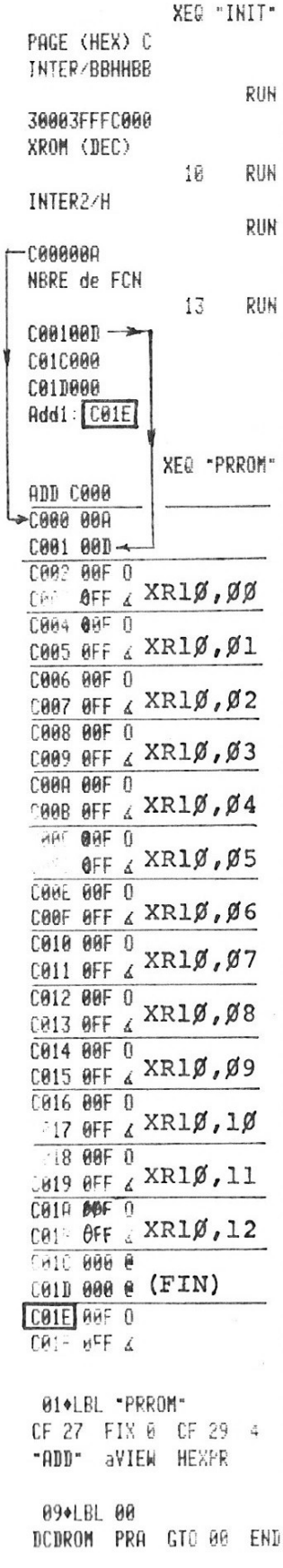
- l'adresse en RAM si c'est un programme :

CHERCHE "DIS" 01EA3C0000 C|1EA
adresse RAM du program "DIS".

```
085  E
008  H
003  C
012  R
005  E
208  H   } entrée alpha autorisée
103  C   }
278  ——> non programmable
     C = REGN 9(0)
158  M = C
315 }  GOSUB 26C5 (ASEARCH)
098 }
2EE    ?C ≠ 0 W          }
381 }                    } NONEXISTENT
00A }  GOLONG 02E0       }
0E8    REGN = C  3(X)
179 }
3F0 }  GOSUB FC5E (DECODE)
3E0    RTN
```

MLDL

PRGM "utilisateur" sur EPROM



Un MLDL permet en premier lieu de charger dans sa RAM tout PRGM écrit en langage "utilisateur", avec ou sans PRGMmation Synthétique. Il est ensuite possible, au moyen d'un PRGMeur d'EPROM, de copier le contenu de cette RAM dans un jeu d'EPROM. Cette possibilité appréciable justifie presque à elle seule, le prix d'achat d'un MLDL mais ce dernier offre heureusement d'autres possibilités non moins appréciables.

Sans faire de cinéma ni de théâtre, nous allons voir ensemble comment il faut procéder pour charger des PRGM sur EPROM. Signalons que la lecture du présent "papier" suppose que le contenu du livre "AU FOND..." de JDD, a été parfaitement assimilé car il me semble indispensable d'avoir un minimum de connaissances avant d'aborder l'utilisation d'un MLDL. Les indications qui vont suivre se rapportent à l'utilisation du MLDL '400041' d'Etienne POUPEE (SCIP), MLDL associé à l'EPROM (le jeu) de Stéphane BARIZIEN. Notons que cet ensemble, avec les PRGM "de service" en langage "utilisateur", constitue à mon avis un très bon outil de travail avec un rapport qualité/prix fort honorable. Je pense que je dois bien cette pub gratuite à Etienne qui est un vrai passionné de "hard" et qui a su s'entourer d'une équipe choisie parmi les meilleurs PRGMmeurs en 'MC' (je n'oublie pas Jean-Jacques DHENIN...). Je crois utile de préciser qu'il me semble pour le moins CORRECT, de réserver sa préférence aux productions françaises et pensons au SAV. Quant à la capacité limitée à 4kO, il est parfaitement possible d'écrire 2 ou 4 fois 4kO et, avec le soin et la rigueur indispensables, cela devrait "tourner" sans problème (à suivre...). Ceux qui désireraient faire des essais en RAM MLDL des n fois 4kO, peuvent emprunter un (ou 3 ?) MLDL à un collègue mais cela n'est pas indispensable. Pensons également au prix des EPROM 16kO et se souvenir que plus la capacité est grande, plus la place inutilisée dans l'EPROM "de pesée" est importante.....

Avant de copier en RAM MLDL nos PRGM favoris, il semble utile de tenir compte des possibilités particulières offertes par les XROM. Un code XRom utilise 2 Octets alors qu'un XEQ 'LBL ALPHA global à 1 CARactère' utilise 3 Octets ainsi d'ailleurs que tout XEQ 'LBL local'. Une utilisation judicieuse de cette particularité permet donc de faire une économie appréciable d'Octets tout en accélérant l'exécution. Souvenons-nous que la recherche des LBL ALPHA globaux (et des XR) se fait de BAS en HAUT, en partant de la position initiale du POINTeur et que nous avons de ce fait intérêt à placer en tête de MEM, le PRGM qui utilise le plus de s/PRGM (°) et de placer à la fin, le s/PRGM le plus utilisé et ainsi de suite.

(°) à LBL ALPHA globaux

INITialisation du MLDL

Avant d'utiliser le MLDL, ilfaut l'initialiser, c'est-à-dire :

- *nettoyer sa RAM*
- *préciser le N° HEX de la PAGE RAM*
- *préciser le N° XROM désiré (préfixe)*

Mon PRGM "INIT" ci-joint permet d'effectuer tout ce travail ainsi que de préparer le FAT, y compris la création automatique des 2 mots qui clôturent ce FAT. L'utilisation de "INIT" (PRGM ci-joint) est très simple et évite d'éventuels oublis.

Si le connecteur du MLDL est introduit, par exemple, dans le PORT 3, le jeu d'EPROM (ML1H) sera adressé de facto, à la PAGE 'D' et il faudra retenir ce fait.

XEQ "INIT" (avec ou sans IMPrimante) vous demandera le N° HEX de la PAGE qu'il suffit d'introduire (grâce aux merveilleuses Fonctions de Stéphane).

La position dans laquelle doit se trouver chacun des INTER est précisée : H = Haut = 1
B = BAS = Ø
et vous remarquerez que la position des 4 INTER de droite, est l'équivalent BINaire de la PAGE choisie. Il faut noter qu'il ne faut pas choisir une PAGE qui correspondrait à un PORT qui serait occupé par un module (PPC-ROM, par exemple) mais que par contre, il est possible de choisir une PAGE inférieure à 8 et supérieure à 4 en tenant compte cependant que ces PAGEs sont normalement réservées à des périphériques. Il est fortement déconseillé d'adopter la PAGE 8 qui doit être réservée au XF (PORT 1) et ceux qui possèdent une 'CX' devront introduire un XF dans ce PORT. Méfiez-vous également de la PAGE 'E' à laquelle s'adresse le lecteur de cartes.

R/S affichera quelques instants le code de "nettoyage" puis demandera le N° DEC de l'XROM désirée. L'introduire après avoir fait un choix judicieux et R/S. L'affichage signalera qu'il faut placer le second INTER en Haut (sur ON ou 1) qui est une position obligatoire à partir de ce stade.

R/S, le préfixe de l'XROM choisi est introduit automatiquement (en HEX) à la 1ère Adresse et le NomBRE de FonCtioNs est demandé. L'introduire puis R/S. Ce NBRE de FCN sera introduit automatiquement à la seconde Add et, toujours automatiquement, les 2 MOTs 'NOP' seront placés à la fin du FAT.

Pour terminer, la 1ère Adresse disponible est indiquée. Vous pouvez alors éventuellement vérifier au moyen de XCAT xr qui vous affichera successivement, autant de "4 canards" que vous avez demandé de Fonctions. Le listing ci-joint donne un exemple du résultat obtenu (PRGM "PRROM" également joint).

L'INITialisation et la préparation du FAT sont alors effectuées mais j'allais oublier d'indiquer qu'il est nécessaire de placer un XFUNCTIONS en PAGE 8 (PORT 1 et donc de ne pas utiliser cette PAGE (INTER/BBHBBB interdit).

```
 01♦LBL "INIT"
 02♦LBL 00
"PAGE (HEX)" aVIEW  E
HEXPR DECODE  E SUB$
ASTO L  CLA  ARCL Y
ATOX  5  X>Y?  GTO 00
R↑  "α!♦ˣ"  XTOA  1-
AROT  RCL E  STO d
"INTER/BB"  FS? 04  "┣H"
FC?C 04  "┣B"  FS? 05
"┣H"  FC?C 05  "┣B"
FS? 06  "┣H"  FC?C 06
"┣B"  FS? 07  "┣H"
FC?C 07  "┣B"  PROMPT
"30003FFF"  ARCL L
"┣000"  AVIEW  COPYROM

 47♦LBL 13
"XROM (DEC)"  31  PROMPT
X>Y?  GTO 13  CLA
ARCL L  INIT  "INTER2/H"
PROMPT  D-H  ASTO X
"000"  MLIADDR  "┣00"
ARCL X  AVIEW  PSE  POKE
CF 02  ASTO X

 69♦LBL 12
"NBRE de FCN"  15
PROMPT  64  X<Y?  GTO 12
RDN  X>Y?  SF 02  D-H
ASTO L  CLA  ARCL Z
"┣0"  FC?C 02  "┣0"
ARCL L  AVIEW  PSE  POKE
 E  +  2  *  D-H  E
ALENG  X>Y?  SF 02  "┣0"
FC? 02  "┣0"  1-
FC?C 02  2-  AROT
MLIADDR  XEQ 03  XEQ 03
ASTO X  "Add1: "  ARCL X
PROMPT  OFF

114♦LBL 03
"┣000"  AVIEW  PSE  POKE
END
```

CODES DEC :

```
PAS 18: 246 128 004 033
        000 128 001
AUTEUR: rS
```

Chargement des PRGM

La merveilleuse Fonction 'USER' transfère un PRGM situé en MEM centrale, dans la RAM MLDL. Il faut apprécier ici l'importance du travail effectué par 'USER' et imaginer ce que cela représenterait si nous devions le faire à la main.....

Plaçons en ALPHA une chaîne 'dddFFF' où :

ddd est l'Add à partir de laquelle le PRGM doit être transféré. Deux MOTs doivent être prévus en amont pour y STOcker les données pour la 'COPY'. Pour le 1er PRGM transféré, l'Add1 indiquée par le PRGM "INIT" peut, par exemple, être utilisée.

FFF est l'Add du 1er MOT du FAT, MOT utilisé pour ranger l'ADD du 1er LBL ALPHA global du PRGM à transférer. Cela suppose que la place nécessaire en aval pour STO les Add des LBL suivants, a été prévue. Si tous les LBL ALPHA globaux ont été comptabilisés et indiqués lors de l'exécution du PRGM "INIT", cette dernière condition est remplie.

Avant d'XEQ 'USER', il faut :

- PACKer le PRGM en mode PRGM par GTO ..
- placer le POINTeur sur la 1ère ligne du PRGM et XEQ "COMPILE", ce qui a pour effet de "COMPiler" tous les GTO & XEQ et de remplacer les éventuels GTO codés sur 2 Octets par des GTO codés sur 3 Octets, si la valeur du saut le justifie. Encore une merveilleuse Fonction !

Le calcul de l'Add du dernier MOT d'un PRGM est relativement facile et Etienne a fait un PRGM qui fait ce travail. L'Add de "FIN" (°) est égale à l'Add de début (celle qui est indiquée dans le FAT) à laquelle on ajoute le Nbre d'Octets du PRGM moins un. Exemple: le PRGM "INIT" qui occupe 293 Octets et qui débuterait à .Ø21 se terminerait à l'Add : HEX Ø21 + DEC 293-1 (= HEX 124) = HEX 145 .Ce PRGM sera donc compris entre les Add .Ø1E (!) et .124 et les VAL suivantes se trouveront dans les 2 mots du FAT : 2ØØ & Ø21 .

(°) c'est le nom du PRGM d'Etienne.

```
 01♦LBL "FIN"
3 "DEBUT" aVIEW HEXPR
CLA DECODE 3 SUB$
ASTO 00

 11♦LBL 00
"FAT" aVIEW 3 HEXPR
CLA DECODE 3 SUB$
ASTO 01 "PRGM "USER""
AVIEW CLA ARCL 00
MLIADDR ARCL 01 AVIEW
BEEP STOP

 30♦LBL "FINA"
CLA ARCL 00 H-D 2 -
D-H MLIADDR CODE
DCDROM AVIEW 5,07
SUB$ ASTO 02 CLA
ARCL 00 H-D E - D-H
MLIADDR CODE DCDROM
AVIEW 6,06 SUB$
ASTO 03 CLA ARCL 02
H-D 7 * CLA ARCL 03
H-D X=0? GTO 01 7 -
+ GTO 02

 71♦LBL 01
-

 73♦LBL 02
CLA ARCL 00 H-D + E
- D-H MLIADDR CODE
DCDROM AVIEW ,03 SUB$
ATOX H-D 3 + D-H
ASTO 00 GTO 00 END
                CAT 1
LBL'INIT
END         293 BYTES
LBL'SAVEROM
LBL'GETROM
END         191 BYTES
LBL'PRROM
END          34 BYTES
LBL'FIN
LBL'FINA
END         184 BYTES
.END.        05 BYTES
```

Nom de la ROM

Bien que cela ne soit pas indispensable, le nom de la ROM peut être affiché en tête des Fonctions listées par un CAT 2 (ou un XCAT xr). Pour obtenir ce résultat, il faut réserver les 2 premières lignes (MOTs) du FAT pour y STO l'Add d'entrée du nom choisi. Vous trouverez dans 'AU FOND' toutes les indications nécessaires pour écrire les MOTs adéquats. Remarquons que le 1er MOT du FAT ne doit pas avoir '2' comme 1er DIGit.

Derniers MOTs de la ROM

En principe, il ne faut rien écrire aux Add .FF4 à .FFA mais nous verrons dans un prochain "papier", qu'il est possible de placer notre machine dans une configuration prédéterminée dès la mise sous tension (ON). On peut ainsi créer les ASN désirées, redéfinir le clavier, etc.... Mais là commence le domaine du 'MC'.

Introduction des XROM

Avant de transférer un PRGM dans la RAM MLDL il est indispensable de remplacer les XEQ à des LBL ALPHA globaux par des XROM, pour la raison indiquée précédemment. La méthode la plus simple me semble être d'ASN ces XR à des touches et pour ce faire nous utiliserons une autre merveilleuse Fonction : 'XYZASN'.

Les codes à introduire seront :

- PREFixe issu de la carte des codes (exemple: 162 pour XR 1∅,)

- SUFFixe = 64P + S (exemple: (64 × 1∅) + 4 = 644 pour XR 1∅,∅4). Il est également possible d'utiliser 'LB ___' ou les FS adéquates.....

Clôture de la ROM

Dans le chapitre 'Derniers MOTs de la ROM', j'ai oublié d'indiquer qu'il est pratiquement indispensable d'introduire la somme de contrôle à l'Add .FFF. Cette somme de contrôle permet de vérifier si une copie a été faite correctement (sur EPROM, par exemple).

Les 4 MOTs situés en amont ne sont pas indispensables et peuvent donc être négligés.

Pour introduire la somme de contrôle, il suffit d'XEQ l'autre merveilleuse F 'RSUM'. Imaginez également quel travail vous devriez faire "à la main" (?).

Chargement sur EPROM

Vos PRGM chéris étant placés dans la RAM du MLDL et les opérations précitées ayant été effectuées avec soin, il vous restera à essayer ces PRGM, le XCAT correspondant et à envoyer le tout (?) à la personne de votre choix qui dispose d'un PRGMmeur d'EPROM. Vous avouerez que cela n'est pas pratique mais il y a heureusement une autre solution. Le PRGM 'SAVEROM' ci-joint permet, comme son nom l'indique, de sauvegarder sur K7, l'ensemble du contenu des 4kO de la RAM MLDL.Le FICHier correspondant occupe 825 REG et il serait possible de modifier ce PRGM pour utiliser le lecteur de cartes mais je vous laisse calculer le Nbre de cartes nécessaires.....

Le NOM demandé est celui sous lequel le FICHier sera STO et il suffit d'attendre que le "travail" s'effectue. Un 'SEC' peut éventuellement être fait.

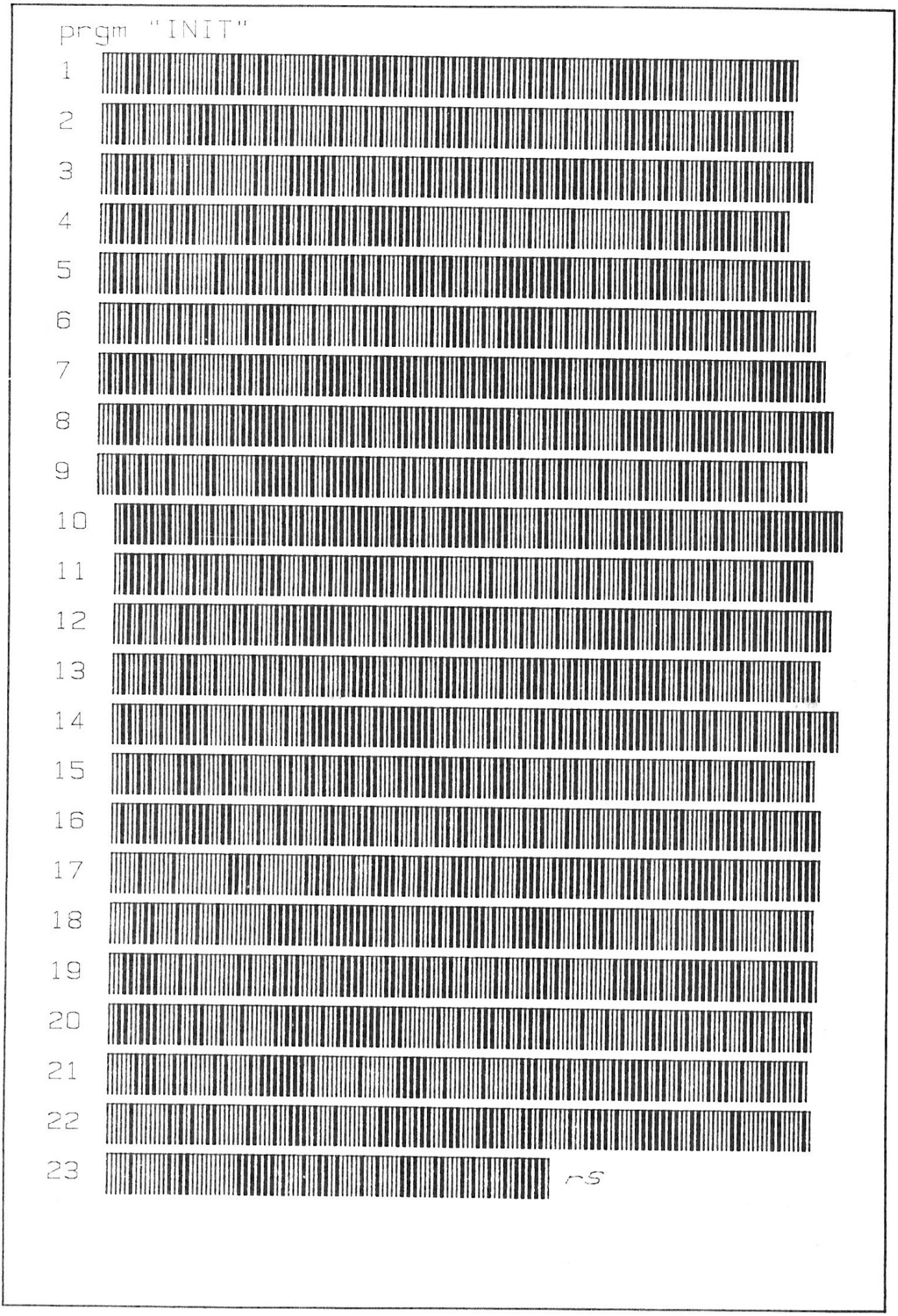
La K7 accompagnée des renseignements indispensables, pourra alors être envoyée pour copie sur EPROM (la partie "GETROM sera alors utilisée).

Signalons qu'un MLDL permet également de copier toute ROM placée dans des modules genre XF, etc... et cette possibilité n'est pas à dédaigner. Il suffit d'utiliser la F 'COPYROM' (une autre ....) de l'EPROM 'ML 1H'

J'espère ne rien avoir oublié et je vous assure que cela est beaucoup plus facile à faire qu'à expliquer....

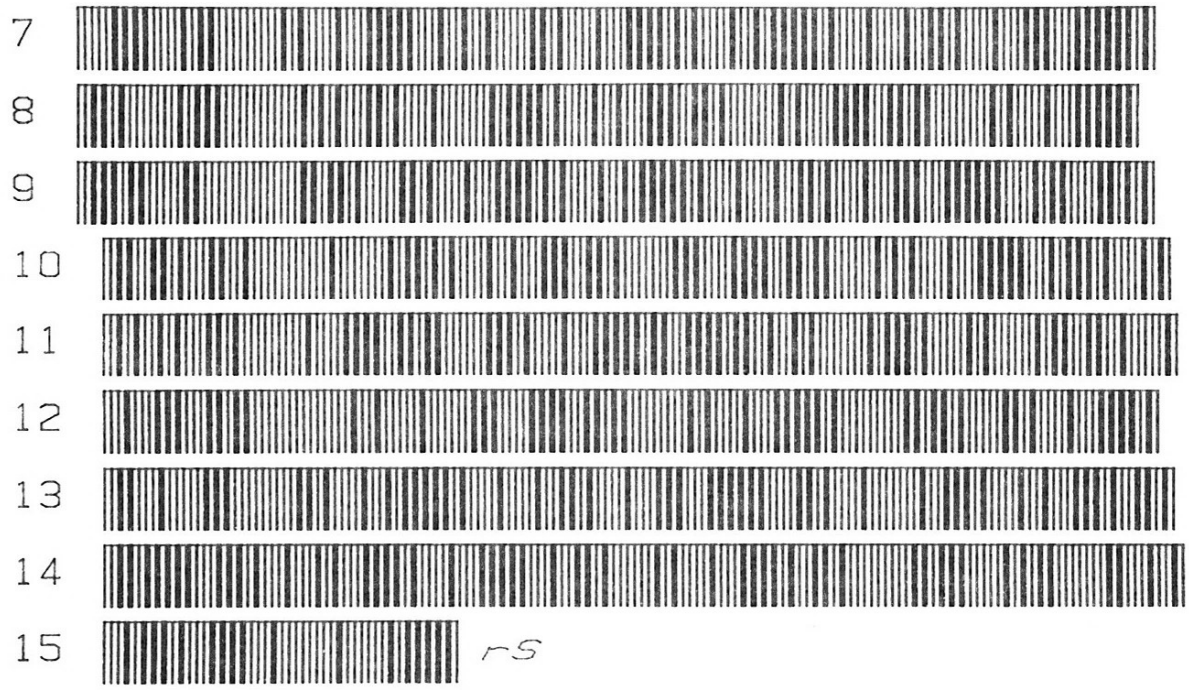
Je crois que nous devons bien un grand et sincère MERCI à JJ DHENIN et St BARIZIEN ainsi qu'à E POUPEE pour le très beau travail qu'ils ont réalisé. rS (T178)

```
 01♦LBL "L"
FIX 0  CF 01  CF 02  ,1
2,12802

 07♦LBL 00
D-H  ASTO a  ALENG  E
X=Y?  SF 01  RDN  RDN  9
RCL Z  INT  X>Y?  SF 02
X<>Y  RDN  CLA  FS?C 01
"├ "  ARCL a
"├          "  FC?C 02
"├ "  ARCL X  PRA  RDN
ISG Y  ISG X  GTO 00
END                rS

            PRP "SAVEROM"

 01♦LBL 11
"NOT PAGE F"  TONE X
PROMPT

 05♦LBL 10
275  PSIZE  "NOM"  AON
PROMPT  AOFF  825
FS? 01  CREATE  CLX
SEEKR  RTN

 18♦LBL "SAVEROM"
SF 01  XEQ 10  "559"
MLIADDR  70  POSA  X=0?
GTO 11  ASTO X  CLA
XEQ 01  "AB3"  MLIADDR
ASTO X  "55A"  XEQ 01
"FFF"  MLIADDR  ASTO X
"AB4"

 39♦LBL 01
MLIADDR  ARCL X  CLX
ROM-REG  WRTRX  RTN

 46♦LBL "GETROM"
CF 01  XEQ 10  XEQ 02
XEQ 02  XEQ 02  "000"
MLIADDR  CODE  DCDROM
6,07  SUB$  H-D  "XROM "
CF 28  CF 29  FIX 0
ARCL X  PROMPT  RTN

 66♦LBL 02
,274  READRX  RCL 00
DECODE  8,1  SUB$  CLX
REG-ROM  END
```


prgm "INIT"
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
rS

```
prgm "SAVROM"
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
```

```
prgm "FIN"
1
2
3
4
5
6
```



Deux MOTs pour COPY

Les 2 premiers MOTs d'un PRGM "utilisateur" placé en ROM contiennent les indications pour la "COPY" (cf 'AU FOND' de JDD). Le 1er MOT indique le Nbre de REG arrondi éventuellement à l'entier supérieur, REG occupés par le PRGM. Le second MOT est constitué de la façon suivante :

- le 1er DIGit est 2 pour un PRGM non PRIVATE et est 3 pour un PRGM PRIVATE (voici donc comment on peut "PRIVATiser" un PRGM à placer en ROM),

- le 2ème DIGit est le Nbre d'Octets de la "pesée" et peut donc être Ø, 1, 2, 3, 4, 5, 6

- le 3ème DIGit est toujours zéro,

en mode 244, bien entendu.

Ces 2 MOTs sont écrits automatiquement par la Fonction 'USER' de 'ML 1H' de Stéphane BARIZIEN et si vous désirez "PRIVATiser" un PRGM, il suffira d'utiliser la F 'POKE' pour insérer la bonne VAL.

Il est peut-être bon de signaler que la F 'USER' écrit également tous les MOTs du FAT correspondant au PRGM désigné et ceci à partir de l'Add indiquée.

Bonnes PRGMmations, rS (T178)

LE COIN DES U-PROS

CHIESA

T 89 LEGRAND RENE
62 Allée des Demoiselles
31400 TOULOUSE

CALCUL MECANIQUE DES LIGNES ELECTRIQUES AERIENNES.

H.P. 41CV (PRG.CHIESA)
-:-:-:-:-:-:- Sans Imprimante

Domaine d'application : Portées jusqu'à 300 M.
Dénivelées des supports ≤ 30 M.
Voltage inférieur à 60.000 Volts.
-:-:-

FEUILLE DE CALCULS.

XEQ "CHIESA" "SIZE 050 "

** DONNEES **

Tableau V

Nature du Conducteur: Cuivre dur A ou B - (Aluminium -
Aluminium et acier - Almelec-Aldrey)
Almelec-Acier - Multimétal

constitution : 19 x 30/10

-SECTION MM2 134 R/S
Diamètre MM 15 R/S
-Poids KG/M 0,3685 R/S

Tableau I

ZONE 1 (2) 3 4 R/S

H.P. affiche : VENT KG/M2 "ETE=120" "HIVER=30"

Tableau II

Coefficient d'ELASTICITE" 5500 R/S
" de DILATATION" 0,000023 R/S

Tableau III

H.P. affiche :" TENSION EN KG/MM2 "
" MAXIMUM 5 R/S (1)
" Prise EN COMPTE 6 R/S (2)

Nota : (2) peut être inférieur à (1) -H.P. ajustera

** RESULTATS **

PORTEE CRITIQUE 82 M.
H.P. affiche "FLECHE MAXIMUM = A 2% de PORTEE CRITIQUE
oui =1 non=2 1 R/S
si "NON" H.P. demande "COMBIEN ?" ___ R/S
H.P. affiche " FLECHE MAX. THEORIQUE EN M. 1,749
" PORTEE MAX. THEORIQUE EN M. 89
question" PORTEE REELLE EN M ? ..... 85 R/S
" TEMPERATURE DE POSE? 10 R/S degrés Cels

H.P. affiche alors ("CONDITIONS ETE" ou "CONDTS HIVER"
(1) (2)

(1) portée réelle ≥ à portée critique
(2) portée réelle < à portée critique

```
01♦LBL "CHIESA"
02 CF 28
03 "SIZE 050"
04 PROMPT

05♦LBL B
06 CLST
07 CLRG
08 "**DONNEES=**"
09 AVIEW
10 PSE
11 PSE
12 "SECTION MM2?"
13 PROMPT
14 STO 13
15 "DIAMETRE MM?"
16 PROMPT
17 STO 14
18 "POIDS--KG/M?"
19 PROMPT
20 STO 15
21 ENTER↑
22 RCL 13
23 /
24 STO 18
25 "ZONE1 2 3 4?"
26 PROMPT
27 STO 08
28 "VENT-KG/M2"
29 AVIEW
30 PSE
31 " ETE=120"
32 AVIEW
33 PSE
34 72
35 STO 16
36 " HIVER=30"
37 AVIEW
38 PSE
39 18
40 STO 17
41 "ELASTICITE=?"
42 PROMPT
43 STO 09
44 "COEFF-DILAT?"
45 PROMPT
46 STO 10
47 "TENS. KG/MM2"
48 AVIEW
49 PSE
50 "MAXIMUM=?"
51 PROMPT
52 STO 11
53 "EN COMPTE=?"
54 PROMPT
55 STO 12
56 -8
57 STO 19
58 50
59 STO 20
60 15
61 STO 21
62 -15
63 STO 22
64 45
```

F.MAX.

Température de pose deg.Cels.	0	5	10	15	20	25	30	35	40	45	50
Tension calculée KG/MM2			362				183			1,53	
Tension totale en KGS.			351				245			205	
Flèche en mètres.			0,95				1,36			1,69	
Longueur totale en mètres.			85,03				85,4			85,9	

Tension Tens.Tot. Flèche max. ou except. L.

Tempér.> 50 1.46 195 — 1.71 85.03

ou > θM

Autre Temp. de pose? oui=1 non=2 1 R/S

si ouiTemp. de pose? 30 R/S

45 R/S

50 R/S

FEUILLE DE CALCULS. (Suite)

si:non 2 R/S

H.P. affiche "EFFORTS EN TETE POTEAUX"

1° hypothèse " poteaux alignés"

"F EN TETE = " ..... 92 KGS.

2° hypothèse "poteau d'angle"

question "Valeur de l'angle en degrés" "ANG-DEGRES?"

introduire valeur par 10 R/S

"Valeur Tension Totale" "TENS-TOTALE?"

introduire valeur par 351 R/S

(prise dans tableau ci-dessus)

H.P. affiche "F EN TETE SANS VENT="

"F SANS VENT=" ..... 61 KGS.

affiche "F T + VENT 120"..... 153 KGS.

H.P. demande "AUTRE CAS GENERAL?" oui=1 non=2

oui:retour au chapitre **DONNEES**

non:affichage de "TERMINE" extinction de H.P.

-:-:-:-

NOTA:

A- en "Conditions hiver" H.P. demande "HIVER NORMAL"
oui=1 non=2? oui=Vent à 30 KG/M2
non=Vent à 30KG/M2 + Neige et verglas

Tens.Calculée= 2cas Tc<T maxim. le calcul continue
Tc>T maxim. H.P. affiche "REVOIR HYPOTHESES" et se remet au chapitre "DONNEES"

B- "AUTRE TEMP POSE"
Quand vous introduisez θ = à la temp: max. du Tabl. I H.P. après un indicatif musical affiche à la valeur de la flèche "FLECHE MAX."
Si vous introduisez une valeur de θ supérieure à θM suivant la zone,H.P. après un indicatif musical diff. vous indiquera "FLECHE EXCEPT."

C- "ANGLE EN DEGRES" si vous introduisez 180 degrés
H.P. vous indiquera "DEJA CALCULE EN ALIGNEMENT"

D- un exemple est porté en manuscrit

E- Pour toute explication,me contacter T.89

F- Portées supérieures à 300 M. et lignes à T Haute Tension à traiter sur un autre programme. ainsi que Dérivation > 30 m.

```
 65 STO 23
 66 15
 67 STO 24
 68 -25
 69 STO 25
 70 40
 71 STO 26
 72 8
 73 STO 27
 74 -30
 75 STO 28
 76 50
 77 STO 29
 78 8
 79 STO 30
 80 1
 81 RCL 08
 82 X=Y?
 83 GTO 08
 84 2
 85 RCL 08
 86 X=Y?
 87 GTO 09
 88 3
 89 RCL 08
 90 X=Y?
 91 GTO 10
 92 GTO 11
 93♦LBL 08
 94 RCL 21
 95 STO 32
 96 RCL 19
 97 STO 31
 98 RCL 20
 99 STO 33
100 GTO 12
101♦LBL 09
102 RCL 24
103 STO 32
104 RCL 22
105 STO 31
106 RCL 23
107 STO 33
108 GTO 12
109♦LBL 10
110 RCL 27
111 STO 32
112 RCL 25
113 STO 31
114 RCL 26
115 STO 33
116 GTO 12
117♦LBL 11
118 RCL 30
119 STO 32
120 RCL 28
121 STO 31
122 RCL 29
123 STO 33
124 GTO 12
125♦LBL 12
126 RCL 32
```

TABLEAU I

CONDITIONS CLIMATIQUES.

	REGION	θ° minim	θ' moyen	θM maxim	Ecart total
ZONE I	Climat chaud plaines méridionales	-8	+15	+50	58
ZONE II	Climat moyen FRANCE	-15	+15	+45	60
ZONE III	Climat froid montagnes à partir de 600 m.	-25	+ 8	+40	65
ZONE IV	Climat continental rigoureux. supérieur à 600 m. - Anatolie Centrale	-30	+ 8	+50	80

Vent:120 Kgs/M2 en été -30 Kgs/M2 hiver avec application coefft 3/5 pour corps cylindriques.

NOTA: Dans certaines régions montagneuses il est prudent de tenir compte d'une charge de neige ou d'un manchon de verglas en majorant le poids par mètre du conducteur du poids de la neige ou du verglas;dans ce dernier cas,on peut aussi tenir compte de l'augmentation de section pour le calcul de l'effort du vent. Aucun Réglement ne précise ce point.Une règle empirique qui, bien que discutable dans son principe,semble donner toute sécurité;est de considérer un verglas doublant le poids et triplant le diamètre des conducteurs.

Coefficient de sécurité = $\frac{\text{Charge de rupture de chaque élément}}{\text{effort le + grd auquel est soumis l'élément}}$

Cs = 3

TABLEAU II

VALEURS DE E (Elasticité) -α (Coefft. dilatation).

Nature conduct	Cuiv dur Bronzes	Alumin.	Aldrey	Aluminium et Acier 7 fils	Aluminium et Acier 37 fils	Multimtal
E(fils)	12000	6500	6500	-	-	18000
E(câble	10000	5500	5500	8500	9000	15000
α	16 10^{-6}	23 10^{-6}	23 10^{-6}	18 10^{-6}	17 10^{-6}	12 10^{-6}

TABLEAU III

TENSION MAXIM. ADMISSIBLE,Cs=3. en Kg/mm2.

Cuivre	Bronze C	Multimétal	Aldrey	Alumin	Alumin.-Acier 7 fils	Alumin.-Acier 37 fils
13	17	25	10	5	9	10

Nota : Possibilité d'adopter des valeurs inférieures -H-P ajustera le tir.

```
127 RCL 31
128 -
129 RCL 10
130 *
131 SQRT
132 RCL 13
133 *
134 RCL 11
135 *
136 70.3
137 *
138 RCL 14
139 /
140 FIX 0
141 "   PORTEE"
142 AVIEW
143 PSE
144 "CRITIQ EN M="
145 AVIEW
146 PSE
147 STO 34
148 VIEW 34
149 PSE
150 PSE
151 "PORTEE-MAXIM"
152 AVIEW
153 PSE
154 "FLECH-MAX-TH"
155 AVIEW
156 PSE
157 "2% PORT-CRIT"
158 AVIEW
159 PSE
160 PSE
161 .0212
162 STO 44
163 "OUI=1 NON=2?"
164 PROMPT
165 1
166 X=Y?
167 GTO 89
168 XEQ 91

169♦LBL 89
170 RCL 44
171 RCL 34
172 *
173 STO 45
174 "F-MAX-THEOR="
175 AVIEW
176 PSE
177 FIX 3
178 VIEW 45
179 PSE
180 PSE
181 FIX 0
182 RCL 33
183 RCL 31
184 +
185 RCL 09
186 *
187 RCL 10
188 *
189 CHS
190 RCL 11
191 +
192 STO 46
```

TABLEAU IV

DIAMETRES : FILS ET CABLES.

Composition du conducteur.	Diam. mm.	Composition du conducteur.	Diam. mm.	Composition du conducteur.	Diam. mm.
FILS: 2	2	7x1,9	5,69	37x1,4	9,81
2,5	2,5	2,0	6,00	1,5	10,50
3	3	2,1	6,31	1,6	11,19
3,5	3,5	2,2	6,60	1,7	11,90
4	4	2,3	6,90	1,8	12,60
4,5	4,5	2,4	7,19	1,9	13,31
5	5	2,5	7,50	2,0	14,00
5,5	5,5	2,6	7,81	2,1	14,69
6,0	6	2,7	8,10	2,2	15,40
6,5	6,5	2,8	8,40	2,3	16,10
7	7	2,9	8,69	2,4	16,81
7,5	7,5	3,0	9,00	2,5	17,50
8	8	3,1	9,31	2,6	18,19
8,5	8,5	3,2	9,60	2,7	18,90
9	9	3,3	9,90	2,8	19,60
9,5	9,5			2,9	20,31
10	10	19x1,2	6,00	3,0	21,00
		1,3	6,50	3,1	21,96
CABLES:		1,4	7,00	3,2	22,40
3x1,8	3,87	1,5	7,50		
1,9	4,10	1,6	8,00	61x1,5	13,50
2,0	4,31	1,7	8,50	1,6	14,40
2,1	4,53	1,8	9,00	1,7	15,30
2,2	4,74	1,9	9,50	1,8	16,19
2,4	5,17	2,0	10,00	1,9	17,10
2,3	4,96	2,1	10,50	2,0	18,00
2,5	5,39	2,2	11,00	2,1	18,90
2,6	5,60	2,3	11,50	2,2	19,81
2,7	5,82	2,4	12,00	2,3	20,69
2,8	6,03	2,5	12,50	2,4	21,60
2,9	6,25	2,6	13,00	2,5	22,50
3,0	6,46	2,7	13,50	2,6	23,40
		2,8	14,00	2,7	24,31
7x1,2	3,60	2,9	14,50	2,8	25,19
1,3	3,90	3,0	15,00	2,9	26,10
1,4	4,19	3,1	15,50	3,0	27,00
1,5	4,50	3,2	16,00	3,1	27,90
1,6	4,81			3,2	28,81
1,7	5,10				
1,8	5,40				

```
193♦LBL 85
194 XEQ 86
195 RCL 18
196 X↑2
197 *
198 RCL 09
199 *
200 24
201 /
202 RCL 11
203 X↑2
204 /
205 ENTER↑
206 RCL 18
207 8
208 /
209 RCL 45
210 /
211 +
212 STO 47
213 RCL 45
214 X↑2
215 RCL 09
216 *
217 8
218 *
219 3
220 /
221 STO 48
222 RCL 48
223 RCL 47
224 *
225 4
226 *
227 RCL 46
228 X↑2
229 +
230 SQRT
231 RCL 46
232 +
233 RCL 47
234 /
235 2
236 /
237 SQRT
238 STO 48
239 RCL 48
240 RCL 34
241 X<=Y?
242 GTO 88
243 GTO 87

244♦LBL 88
245 "PORT-MX EN M"
246 AVIEW
247 PSE
248 VIEW 48
249 PSE
250 PSE
251 GTO 84

252♦LBL 87
253 RCL 17
254 STO 16
255 GTO 85

256♦LBL 84
257 "  PORTEE"
258 AVIEW
259 PSE
260 "REELLE EN M?"
261 PROMPT
262 STO 35
263 "TEMP-POSE=?"
264 PROMPT
265 STO 36
266 RCL 34
267 RCL 35
268 X<=Y?
269 GTO 13
270 GTO 14

271♦LBL 14
272 "CONDITNS-ETE"
273 AVIEW
274 PSE

275♦LBL 20
276 RCL 36
277 RCL 31
278 +
279 RCL 09
280 *
281 RCL 10
282 *
283 CHS
284 RCL 11
285 +
286 STO 37
287 RCL 16
288 X↑2
289 RCL 14
290 X↑2
291 *
292 1000000
293 /
294 RCL 15
295 X↑2
296 /
297 1
298 +
299 ENTER↑
300 RCL 18
301 X↑2
302 *
303 RCL 09
304 *
305 24
306 /
307 RCL 11
308 X↑2
309 /
310 STO 38
311 RCL 18
312 X↑2
313 RCL 09
314 *
315 24
316 /
317 STO 39
318 RCL 12
319 RCL 11
320 X=Y?
321 GTO 15
322 GTO 16

323♦LBL 16
324 RCL 11
325 RCL 12
326 -
327 CHS
```

TABLEAU V CABLES & FILS-SECTIONS-Ø-POIDS PAR CATEGORIES.

BRONZE TYPE A dit Bronze siliceux				BRONZE TYPE B Conductibilité 60 et 72% NF C 30-010			
Section mm2	Constitut. Nb xØ mm.	Diam. Extér.	Poids Km.	Section mm2	Constitut. Nb xØ mm.	Diam. Extér.	Poids Km.
3,14	1x2	2	27,9	0,95	1x1,1	1,1	8,45
4,91	" 2,5	2,5	43,6	1,77	" 1,5	1,5	15,70
7,07	" 3,0	3	62,9	2,54	" 1,8	1,8	22,60
7,79	" 3,15	3,15	69,3	3,14	" 2	2	27,90
9,90	" 3,55	3,55	88,0	4,91	" 2,5	2,5	43,60
9,62	" 3,50	3,50	85,5	7,07	" 3	3	62,90
12,56	" 4,00	4,00	111,7	7,79	" 3,15	3,15	69,30
15,90	" 4,50	4,50	141,40				
19,63	" 5,0	5	174,50	12,40	7x1,50	4,5	112,40
28,30	" 6	6	251,30	22,00	" 2,00	6	199,50
10,80	7x1,4	4,2	97,9				
12,40	" 1,5	4,5	112,4				
14,10	" 1,6	4,8	127,8				
17,8	" 1,8	5,4	161,4				
19,84	" 1,9	5,7	178,8				
22,0	" 2,0	6,0	199,5				
27,6	" 2,24	6,7	250,0				
25,0	19x1,3	6,5	228,5	33,6	19x1,5	7,5	306,00
29,3	" 1,4	7,0	266	48,3	" 1,8	9,0	440
38,2	" 1,6	8,0	348	93,2	" 2,5	12,5	850
48,3	" 1,8	9,0	440	148,0	" 3,15	15,8	1349
59,7	" 2,0	10,0	544				
74,9	" 2,24	11,2	682				
93,3	" 2,5	12,5	850				
74,4	37x1,6	11,2	676	35,1	37x1,1	7,7	322
104,9	" 1,9	13,3	953	65,1	" 1,5	10,5	598
116,2	" 2,0	14,	1062	94,1	" 1,8	12,6	860
145,8	" 2,24	15,7	1332	116,2	" 2,	14,0	1062
153,7	" 2,3	16,3	1396				
181,6	" 2,5	17,5	1660				
227,8	" 2,8	19,6	2082				
261,5	" 3	21	2390				
288,	" 3,15	22	2632				
211,3	61x2,10	18,9	1923				
376,	" 2,8	25,2	3450				

CABLES ALUMINIUM (99,5%)				CABLES ALU. A AME ACIER.			
Section mm2	Constitut. Nb xØ mm.	Diam. Extér.	Poids Km.	Section mm2	Constitut. Nbx Ø mm.	Diam. Extér.	Poids Km.
11,9	3x2,25	4,85	32,8	10,8	1+6 1,4	4,2	37,6
14,7	" 2,5	5,4	40,5	14,1	" " 1,6	4,8	49,1
18,5	" 2,8	6,0	50,9	17,8	" " 1,8	5,4	61,9
23,4	" 3,15	6,8	64,4	22,	" " 2,	6,	76,6
				27,8	" " 2,25	6,75	96,7
27,8	7x 2,25	6,75	76,6	34,4	" " 2,5	7,5	119,7
34,4	" 2,5	7,5	94,7	43,1	" " 2,8	8,4	150,
43,1	" 2,8	8,4	118,7	54,6	" " 3,15	9,45	190,
46,2	" 2,9	8,7	126,5				
54,6	" 3,15	9,45	150,				
69,3	" 3,55	10,65	191,				

```
328 RCL 37
329 +
330 STO 37
331 RCL 38
332 RCL 12
333 X↑2
334 *
335 RCL 11
336 X↑2
337 /
338 STO 38
339 GTO 15

340♦LBL 15
341 XEQ "ROOT"
342 RCL 04
343 RCL 13
344 *
345 "TENSION TOT="
346 AVIEW
347 PSE
348 FIX 0
349 VIEW X
350 PSE
351 PSE
352 FIX 2
353 RCL 35
354 X↑2
355 RCL 18
356 *
357 8
358 /
359 RCL 04
360 /
361 STO 41
362 RCL 36
363 RCL 33
364 X=Y?
365 XEQ 99
366 XEQ 96

367♦LBL 97
368 AVIEW
369 PSE
370 VIEW 41
371 PSE
372 PSE
373 RCL 41
374 X↑2
375 8
376 *
377 RCL 35
378 /
379 3
380 /
381 RCL 35
382 +
383 "LONG-TOTALE="
384 AVIEW
385 PSE
386 VIEW X
387 PSE
388 PSE
389 "OTRE T POSE?"
390 AVIEW
391 PSE
392 "OUI=1 NON=2?"
```

75,5	19x2,25	11,25	209
85,9	2,4	12,0	236,5
93,3	2,5	12,5	258
117,	2,8	14,	324
148,1	3,15	15,75	410
188,	3,55	17,75	520
181,6	37x2,5	17,5	501
228,	2,8	19,6	633
288	3,15	22,05	799
366	3,55	24,85	1016
349,3	61x2,7	24,3	965,5

Suite TABLEAU V

116,2	7+30	2,	14,	437
147,1		2,25	15,75	553
181,6		2,5	17,5	683
228,		2,8	19,6	857
288,		3,15	22,05	1083
366,		3,55	24,85	1376
398,		3,7	25,9	1496

CABLES EN ALMELEC

Section	Constitut	Diam.	Poids
9,4	3x2,	4,3	25,9
11,9	2,25	4,85	32,6
14,7	2,5	5,4	40,5
18,5	2,8	6,	50,9
22,	7x2,	10,	165,
27,8	2,25	11,25	209,
34,4	2,5	12,5	258,
43,1	2,8	14,	324,
54,6	3,15	15,75	410,
181,6	37x2,5	17,5	504,
228,	2,8	19,6	633,
288,	3,15	22,05	799,
366,	3,55	24,85	1016,
475,	61x3,15	28,35	1324,

CABLES EN ALMELEC-AME D'ACIER.

Section	Constitut	Diam.	Poids
17,8	1+6 1,8	5,4	61,9
22,	2,	6,	76,6
27,8	2,25	6,75	96,7
34,4	2,5	7,5	119,7
43,1	2,8	8,4	150,
54,6	3,15	9;45	190,
69,3	3,55	10,65	241,
94,2	7+30 1,8	12,6	354,
116,2	2,	14,	437,
147,1	2,25	15,75	553,
181,6	2,5	17,5	683,
228,	2,8	19,6	857,
288,	3,15	22,05	1083,
366,	3,55	24,85	1376,

MULTIMETAL.

FILS.

Section mm2.	Constitut. Nb x ø	Diam. mm.	Poids Km.
3,14	1x1,	2,	25,29
7,07	1	3,	56,90
12,56	1	4,	101,16
19,63	1	5,	158,00

CABLES.

Section mm2.	Constitut Nb x ø	Diam. mm.	Poids Km.
9,42	3x2	4,3	76,58
21,20	3x3	6,5	172,35
21,99	7x2	6,	179,50
49,48	7x3	9,	403,89
59,69	19x2	10,	490,11
134,30	19x3	15,	1027,37

```
393 PROMPT
394 1
395 X=Y?
396 GTO 18
397 GTO 82

398♦LBL 18
399 "TEMP-POSE?"
400 PROMPT
401 STO 36
402 ENTER↑
403 RCL 31
404 +
405 RCL 09
406 *
407 RCL 10
408 *
409 CHS
410 RCL 12
411 +
412 STO 37
413 RCL 11
414 RCL 12
415 X=Y?
416 GTO 15
417 RCL 11
418 RCL 12
419 -
420 CHS
421 RCL 37
422 +
423 STO 37
424 GTO 15

425♦LBL 19
426 "OTRE-K-GAL?"
427 AVIEW
428 PSE
429 "OUI=1 NON=2?"
430 PROMPT
431 1
432 X=Y?
433 GTO B
434 GTO 79

435♦LBL 79
436 "  TERMINE"
437 AVIEW
438 PSE
439 PSE
440 OFF

441♦LBL 82
442 "EFFORTS-POTO"
443 AVIEW
444 PSE
445 PSE
446 "POT-ALIGNES"
447 AVIEW
448 PSE
449 "PSE"
450 GTO 80

451♦LBL 80
452 72
453 ENTER↑
454 RCL 14
```

```
      455 *
      456 RCL 35
      457 *
      458 1000
      459 /
      460 FIX 0
  461 "F EN TETE="
      462 AVIEW
      463 PSE
      464 VIEW X
      465 PSE
      466 PSE
      467 GTO 81

      468♦LBL 81
      469 DEG
470 "ANGL-DEGRES?"
      471 PROMPT
      472 STO 49
      473 180
      474 X=Y?
      475 GTO 70
      476 GTO 71

      477♦LBL 71
478 "TENS-TOTALE?"
      479 PROMPT
      480 ENTER↑
      481 2
      482 *
      483 ENTER↑
      484 RCL 49
      485 2
      486 /
      487 SIN
      488 *
489 "F SANS VENT="
      490 AVIEW
      491 PSE
      492 VIEW X
      493 PSE
      494 PSE
      495 ENTER↑
      496 RCL 14
      497 RCL 35
      498 *
      499 72
      500 *
      501 1000
      502 /
      503 +
504 "F + VENT 120"
      505 AVIEW
      506 PSE
      507 VIEW X
      508 PSE
      509 PSE
      510 GTO 19

      511♦LBL 70
512 "DEJA CALCULE"
      513 AVIEW
      514 PSE
      515 PSE
  516 "EN ALIGNT"
      517 AVIEW
      518 PSE
      519 PSE
      520 GTO 19

      521♦LBL 13
      522 FIX 2
523 "CONDIT-HIVER"
      524 AVIEW
      525 PSE
      526 RCL 17
      527 STO 16
528 "IVER-NORMAL?"
      529 AVIEW
      530 PSE
531 "OUI=1 NON=2?"
      532 PROMPT
      533 1
      534 X=Y?
      535 GTO 20
536 "NEGE-VERGLAS"
      537 AVIEW
      538 PSE
      539 RCL 14
      540 3
      541 *
      542 STO 14
      543 RCL 15
      544 2
      545 *
      546 STO 15
      547 GTO 20

      548♦LBL 96
      549 RCL 33
      550 RCL 36
      551 X>Y?
      552 XEQ 95
      553 XEQ 98

      554♦LBL 86
      555 RCL 16
      556 X↑2
      557 RCL 14
      558 X↑2
      559 *
      560 1000000
      561 /
      562 RCL 15
      563 X↑2
      564 /
      565 1
      566 +
      567 ENTER↑
      568 RTN

      569♦LBL 91
  570 "  COMBIEN?"
      571 PROMPT
      572 STO 44
      573 RTN

      574♦LBL 95
      575 BEEP
      576 BEEP
      577 BEEP
      578 BEEP
      579 TONE 8
580 "FLECH-EXCEPT"
      581 GTO 97

      582♦LBL 99
      583 BEEP
      584 TONE 9
      585 TONE 6
586 "FLECH-MAX=ML"
      587 GTO 97

      588♦LBL 98
589 "FLECHE= EN M"
      590 GTO 97

      591♦LBL "R"
      592 ENTER↑
      593 ENTER↑
      594 ENTER↑
      595 RCL 35
      596 X↑2
      597 RCL 38
      598 *
      599 RCL 37
      600 -
      601 *
      602 *
      603 RDN
      604 *
      605 *
      606 +
      607 RCL 35
      608 X↑2
      609 RCL 39
      610 *
      611 -
      612 RTN

  613♦LBL "ROOT"
    614♦LBL A
      615 FIX 2
      616 0
      617 STO 01
      618 RCL 11
      619 2
      620 *
      621 STO 02
      622 RCL 01
      623 STO 04
      624 XEQ "R"
      625 STO 05
      626 RCL 02
      627 STO 04
      628 XEQ "R"
      629 STO 06
      630 RCL 05
      631 *
      632 X>0?
      633 GTO 05

      634♦LBL 00
      635 RCL 02
      636 RCL 02
      637 RCL 01
      638 -
      639 RCL 06
      640 RCL 05
      641 -
      642 /
      643 RCL 06
      644 *
      645 -
      646 STO 04
      647 XEQ "R"
      648 STO 07
      649 X=0?
      650 GTO 04
      651 ABS
      652 1 E-8
      653 -
      654 INT
      655 X=0?
      656 GTO 04
      657 RCL 07
      658 RCL 06
      659 *
      660 X>0?
      661 GTO 01
      662 RCL 02
      663 STO 01
      664 RCL 06
      665 STO 05

      666♦LBL 02
      667 RCL 04
      668 STO 02
      669 RCL 07
      670 STO 06
      671 GTO 00

      672♦LBL 01
      673 2
      674 ST/ 05
      675 GTO 02

      676♦LBL 04
  677 "T CALCULEE="
      678 AVIEW
      679 PSE
      680 "PSE"
      681 VIEW 04
      682 PSE
      683 "PSE"
      684 RCL 11
      685 RCL 04
      686 X<=Y?
      687 GTO 06
      688 GTO 90

      689♦LBL 90
      690 BEEP
      691 BEEP
692 "REVOIR-HYPTH"
      693 AVIEW
      694 PSE
      695 PSE
      696 BEEP
      697 BEEP
      698 GTO B

      699♦LBL 05
    700 "F1*F2>0"
      701 AVIEW
      702 PSE
      703 GTO A

      704♦LBL 06
      705 RTN
      706 "END"
      707 "END"
      708 .END.
```

8 Cartes

HP-1x

Eric Goubault T 536
5, rue des canadiens
76420, Bihorel

Le 1 août 1984

HP-34C T536
INTERPOLATION DE CORNU

INTERPOLATION DE CORNU (ou formule d'HARTMANN)

Cette formule est une simple interpolation hyperbolique qui trouve son utilité principale en optique et en spectroscopie. Elle permet d'interpoler l'indice de réfraction N d'un verre ou d'un matériau optique quelconque en fonction de la longueur d'onde L de la lumière incidente.De même elle permet d'interpoler la longueur d'onde d'une raie en fonction de son abscisse X mesurée sur un spectrogramme prismatique.

Cette formule s'écrit (ici dans le cas de la relation entre N et L) $N=N0+c/(L-L0)$, où N0,L0,et c sont trois constantes que l'on détermine au moyen des indices connus pour trois radiations.Ces indices sont: N1 pour la longueur d'onde L1
N2 - - - - - L2
N3 - - - - - L3

Nous résolvons les trois équations à trois inconnues en posant d'abord: $d1N=N2-N1$ $d2N=N3-N2$
$d1L=L2-L1$ $d2L=L3-L2$
$d1(LN)=L2.N2-L1.N1$ $d2(LN)=L3.N3-L2.N2$

et $L=d1(LN).d2L-d2(LN).d1L$
$M=d2(LN).d1N-d1(LN).d2N$
$D=d1N.d2L-d2N.d1L$

enfin, on obtient $L0=L/D$ $N0=M/D$
et $c=(L1-L0).(N1-N0)=(L2-L6).(N2-N0)=(L3-L0).(N3-N0)$

On peut donc maintenant calculer l'indice du verre pour n'importe quelle longueur d'onde en ne connaissant pourtant que les quelques indices donnés par les catalogues de verres.

Néammoins cette interpolation n'est optimale que dans un certain domaine de longueur d'onde entourant L1,L2,L3.C'est donc à l'utilisateur, s'il en a la possibilité , de choisir des valeurs de L1,L2,L3 les plus homogènement réparties dans le domaine de valeurs les plus utiles pour ses calculs.Un bon choix lui permettra d'obtenir des résultats quasiment exacts. Toutefois, pour des indices , l'interpolation est bonne à cinq décimales.(erreur absolue maximale:environ 5E-5;le plus souvent pas plus de 1E-5)

Nous avons écrit les formules utilisées dans le cas de l'interpolation de N en fonction de L, elles sont formellement identiques dans le cas de l'interpolation de L en fonction de X (spectroscopie): $L=L0+c/(X-X0)$.Ainsi pour notre programme on introduira les couples connus (L1,X1),(L2,X2),(L3,X3) comme plus haut on devait introduire les couples (N1,L1) (N2,L2),(N3,L3) pour permettre le calcul des coefficients d'interpolation du premier du couple en fonction du second du couple.Enfin, un dernier exemple d'utilisation en optique de cette formule: l'interpolation de l'indice d'un verre en fonction de l'indice d'un verre voisin; on démontre en effet que la relation de Cornu reste valable.Les couples à entrer sont (N1,N1')...(pour les même longueurs d'onde)

```
001 hLBL A
002 5
003 EEX
004 3
005 CHS
006 STOfI
007 hLBL 1
008 RCLfI
009 R/S
010 STOf(I)
011 gISG
012 GTO 1
013 hLBL 0
014 RCL 3
015 RCL 2
016 *
017 RCL 1
018 RCL 0
019 *
020 -
021 STO 6
022 RCL 5
023 RCL 4
024 *
025 RCL 3
026 STO- 5
027 RCL 2
028 STO- 4
029 *
030 -
031 STO 7
032 RCL 0
033 STO- 2
034 RCL 1
035 STO- 3
036 RCL 2
037 RCL 5
038 *
039 RCL 4
040 RCL 3
041 *
042 -
043 STOfI
044 RCL 6
045 RCL 5
046 *
047 RCL 7
048 RCL 3
049 *
050 -
051 RCLfI
```

Examinons maintenant le programme.Initiallement développé sur HP-34C, ce programme est suffisament simple pour être directement adapté sur n'importe quelle autre calculatrice HP.(HP-11,15,67,41...) Pour entrer les trois couples (yi,xi) connus afin d'obtenir les coefficients d'interpolation de y en fonction de x: appuyer A.

A l'affichage, on voit: 0 , il faut entrer: y1,R/S
1 -- ---- ------ x1,R/S
2 -- ---- ------ y2,R/S
3 -- ---- ------ x2,R/S
4 -- ---- ------ y3,R/S
5 -- ---- ------ x3,R/S

la machine calcule (une dizaine de secondes) puis s'arrete. Les coefficients NO,LO,c sont maintenant rangés dans les mémoires 2,1,0 respectivement.

Pour calculer y en fonction de x: il faut entrer x (dans la même unité que les xi) puis appuyer B.Quelques secondes plus tard, on voit à l'affichage la valeur de y interpolée.(dans la même unité que les yi) On peut recommencer le calcul de y en fonction d'un x: x',B...

Si on veut des valeurs interpolées de x en fonction de y, au lieu de recommencer le calcul des coefficients en appuyant A et en introduisant les couples (x1,y1)..., il suffit d'échanger les valeurs des mémoires 2 et 1.En effet si y=NO+c/(x-LO) on a de même: x=LO+c/(y-NO).

Le calcul des coefficients d'interpolation NO,LO,c peut être appelé en sous programme en faisant GSBO; mais il faut alors que les registres mémoires soient remplies comme ils le sont à la fin du stade de demande.(sauf pour le registre I)

Voici le détail des registres utilisés:

valeur prise à la fin de	la demande (lignes1-13)	calcul des coefficients(114-76)
R0	y1	c
R1	x1	LO
R2	y2	d1N puis NO
R3	x2	d1L puis LO
R4	y3	d2N
R5	x3	d2L
R6		d1(LN)
R7		d2(LN)
I	6,005 (compteur)	D

Le label B ne change pas la valeur des registres.Il peut être utilisé lui aussi en sous programme si on a mis dans R0:c R1:LO,R2:NO et bien sur x dans X.

```
052 /
053 STO 3
054 RCL 7
055 RCL 2
056 *
057 RCL 6
058 RCL 4
059 *
060 -
061 RCLf I
062 /
063 STO 2
064 RCL 1
065 RCL 3
066 -
067 RCL 0
068 RCL 2
069 -
070 *
071 STO 0
072 RCL 3
073 STO 1
074 hRTN
075 hLBL B
076 RCL 1
077 -
078 RCL 0
079 X<>Y
080 /
081 RCL 2
082 +
```

Eric Gaubert

HP-71

MEMDUMPX

```
1 ! DAVASE PHILIPPE                                LE 20/11/84
2 ! T404-PC67
3 ! 78500 SARTROUVILLE
4 !
5 ! VOICI UN "MEMDUMPX" POUR HP71B
6 ! QUI TRAVAILLE EN DECIMAL,HEXA.,ET SUR NOM DE FICHIER
7 !
8 ! SUR UNE IDEE DE E.GENGOUX SUR HP75 (MICRO-REVUE Nr1 du 07-08/84)
9 !
100 ! MEMDUMPX visualisation de la RAM
110 DESTROY ALL @ FIX 0
120 DIM H1$[5],H2$[5],H3$[4],L$[80],L0$[6],L1$[2],N1$[8],N2$[45],Z2$[6],Z1$[6]
130 DISP "memory dumP" @ WAIT 1 @ PRINTER IS ":PR"
140 INPUT "Dec[1] Hex[2] File[3]";K
150 IF K<1 OR K>3 THEN 140
160 ON K GOTO 310,240,170
170 INPUT "Filename is?";N1$ @ I0=0
180 I0=I0+1 @ N2$=CAT$(I0) @ IF N2$="" THEN 170
190 IF N2$[1,LEN(N1$)]#N1$ THEN GOTO 180
200 D1=HTD(ADDR$(N1$))
210 A2=VAL(N2$[18,22]) @ D2=D1+A2+1
220 PRINT "FILE: ";N1$;" FROM ADDR.(dec.) ";D1;" TO ";D2 @ PRINT
230 D1=D1-RMD(D1,16) @ D2=D2+16-RMD(D2,16) @ GOTO 330
240 INPUT "TOP ADDR(Hex)?";H1$ @ H1$[5,5]='0'
250 D1=HTD(H1$)
260 IF D1>1048575 OR D1<0 THEN GOSUB 440 @ GOTO 220
270 INPUT "BOTTOM ADDR(Hex)?";H2$ @ H2$[5,5]='0'
280 D2=HTD(H2$)
290 IF D2<0 OR D2>1048575 OR D1>D2 THEN GOSUB 440 @ GOTO 270
300 GOTO 330
310 INPUT "TOP ADDR.(Absol,Dec)?";D1
320 D1=D1+16-RMD(D1,16)
330 IF D1>1048575 OR D1<0 THEN GOSUB 440 @ GOTO 310
```

```
340 INPUT "BOTTOM ADDR.(Absol.Dec)?";D2
350 D2=D2-RMD(D2,16)
360 IF D2<0 OR D2>1048575 OR D1>D2 THEN GOSUB 440 @ GOTO 340
370 PRINT
380 PRINT "ADDR   0 1   2 3   4 5   6 7   8 9   A B   C D   E F     ASCII
ADDR.DEC"
390 PRINT "------+-----------------------------------------------+----------+-----
----------"
400 FOR I=D1 TO D2 STEP 16
410 GOSUB 450
420 NEXT I
430 PRINT @ PRINT "END DUMP "&DATE$&"  "&TIME$ @ END
440 BEEP @ DISP "OUT OF RANGE" @ WAIT 2 @ RETURN
450 L$[1,5]=DTH$(I) @ L$[6,6]="-"
460 FOR J=0 TO 15 STEP 2
470 L0$=PEEK$(DTH$(I+J+1),1)&" "&PEEK$(DTH$(I+J),1)
480 A1=3*J+8 @ A2=3*J+10 @ A3=3*J+7
490 L$[A1,A2]=L0$ @ L$[A3,A3]=" "
500 L3$=CHR$(HTD(PEEK$(DTH$(I+J+1),1)&PEEK$(DTH$(I+J),1)))
510 IF NUM(L3$)<32 OR NUM(L3$)>254 THEN L3$="."
520 L$[J+55,J+55]=L3$
530 NEXT J
540 L$[74,80]=STR$(I)
550 PRINT L$[1,80]
560 RETURN

1 ! VOICI MAINTENANT UN UTILITAIRE SUR HP71B
2 ! PERMETTANT DE REMBOBINER LA CASSETTE AUTOMATIQUEMENT
3 ! OU SOIT LA POSITION DU LECTEUR SUR LA BOUCLE
4 !
5 ! EN ANNEXE ,CE MEME PROGRAMME "REW"
6 ! LISTE PAR LE PROGRAMME "MEMDUMPX" PAR HP71B
7 ! IMPRIMANTE GP100IL-A SICAPE
100 ! PROGRAMME "REW" REMBOBINAGE CASSETTE
110 A1=DEVADDR("TAPE") @ SEND LISTEN A1 DDL 7 UNT UNL



EXECUTION DE "MEMDUMPX" SUR "REW"
BONNE PROGRAMMATION A TOUS



FILE: REW FROM ADDR.(dec.)  195182.  TO  195554.

ADDR   0 1   2 3   4 5   6 7   8 9   A B   C D   E F     ASCII       ADDR.DEC
------+-----------------------------------------------+----------+-----------
2FA60- 0 0   F F   F F   F F   F F   F F   F 0   5 2  . . . . . R    195168.
2FA70- 4 5   5 7   2 0   2 0   2 0   2 0   2 0   1 4  E W          .    195184.
2FA80- E 2   0 0   0 3   1 9   0 2   1 1   8 4   F 6   . . . . .      195200.
2FA90- 0 2   0 0   0 0   0 0   0 0   0 0   0 0   1 F  . . . . . .      195216.
2FAA0- 0 0   6 0   C 5   6 F   F 5   9 4   3 4   9 4  . ` U o   4      195232.
2FAB0- 0 4   D 2   1 4   9 4   E 4   4 4   5 5   E 4  . r .   D U      195248.
2FAC0- 1 4   E 4   4 4   0 5   5 2   E 5   0 4   5 2  .   D . R . R    195264.
2FAD0- 4 5   9 5   C 4   9 4   4 4   1 5   9 4   2 4  E   T   D .   $    195280.
2FAE0- 5 5   0 4   3 2   5 5   2 5   0 5   8 2   0 4  U . 2 U % .   .    195296.
2FAF0- 7 5   1 3   2 3   D 4   0 0   2 F   0 0   E 0  u . # t . / .      195312.
2FB00- C 6   0 F   5 5   2 4   D 5   5 4   4 4   4 5  V . U $ u T D E    195328.
2FB10- 1 5   E 4   4 4   0 5   4 2   5 4   0 4   2 2  .   D . B T . "    195344.
2FB20- 5 5   D 4   2 4   F 4   2 4   9 4   E 4   5 4  U t $   $   T    195360.
2FB30- 2 4   0 5   C 2   1 4   0 4   3 2   1 4   3 4  $ . R . . 2 . 4    195376.
2FB40- 3 5   5 5   4 4   4 5   5 5   0 4   1 2   5 4  5 U D E U . . T    195392.
2FB50- 4 5   F 5   D 4   1 4   4 4   9 5   1 4   5 5  E   t . D   . U    195408.
2FB60- 5 5   D 4   5 4   E 4   4 4   D 5   0 0   3 F  U t T   D u . ?    195424.
2FB70- 0 0   E 0   C 5   1 F   1 5   E 5   3 2   F 5  .   U . .   2      195440.
2FB80- 9 4   4 4   0 5   C 2   1 4   0 4   0 2   F 5    D . R . . .      195456.
```

GRAPHIQUES

POUR FAIRE DE BEAUX GRAPHIQUES SUR HP 71 ou 75.

un livre à lire absolument (compte-rendu par Eric GENGOUX, T136)

Possédant depuis déja pas mal de temps ces deux appareils, je n'avais pas été très convaincu par les essais tentés par nos amis de PPC-US sur imprimantes dites "graphiques" HP 82905 ou THINKJET (voir les deux derniers "Computer Journal"): ces imprimantes ne sont à l'évidence pas faites pour cela... N'ayant pas les moyens de m'offrir une table traçante HP, j'ai fait comme beaucoup, et acquis un "plotter" genre PIL83 (SICAPE/TANDY) ou SCIP. Mais si ces appareils fonctionnent à merveille, convenons que, jusqu'à présent, leur mode d'emploi, orienté vers la conduite par une 41, obligeait à une programmation "à ras de terre", et aucun logiciel général n'était fourni avec. Même remarque à propos de la table Packard, d'ailleurs, avec laquelle il faut également, sinon réinventer la roue, du moins le cercle, et pas mal d'autres choses avant de pouvoir enfin faire du travail sérieux...

Mais voici que vient de paraître chez MASSON un livre vraiment remarquable et, pour tout dire, précieux: "GRAPHISME SCIENTIFIQUE SUR MICRO-ORDINATEUR", de R.DONY. C'est un véritable manuel de base du dessin sur micro-ordinateur, et il propose une méthode générale pour produire des courbes et tracés dans le plan ET DANS L'ESPACE, et ce dans les unités de l'utilisateur. J'ai aussitôt entrepris des essais et, ô bonheur, les programmes se sont révélés très faciles à utiliser, peu gourmands en octets et efficaces. Voici donc ce que contient ce livre, quelques petites astuces d'adaptation des programmes et un bref rappel des possibilités des deux traçeurs précités (riches possesseurs des tables HP, dont le manuel est également assez pauvre en applications, ceci vous concerne AUSSI!).

La méthode a été écrite pour un APPLE II, dont l'écran graphique est adressable par point (axe des Y orienté vers le bas avec 280*192 points, les tables SICAPE et SCIP offrant 480 "steps" de 0.2mm en largeur de bande (x), et bien davantage en longueur(y). Elle permet tout d'abord de simuler des fonctions graphiques évoluées (déplacements absolus ou relatifs, plume baissée ou levée, tracé d'axes gradués ou non et de "grilles"...), et résout de manière générale les problèmes d'échelle et de "limites de l'épure" en considérant des "FENETRES" dans l'espace utilisateur, les étendues correspondantes sur la feuille de papier, de sorte que toutes les conversions d'unités soient automatiques, l'utilisateur (vous et moi...) n'ayant plus à s'en préoccuper, non plus que de contrôler la table au niveau le plus bas!

Les fonctions de base étant définies, on passe au traitement des parties sortant de la fenêtre (par exemple, branches infinies d'une courbe plane). De nombreux et très jolis exemples sont donnés, qui nous réconcilieront avec les "belles fleurs" de la géométrie analytique (courbes et réseaux en paramétriques, polaires; magnifiques napperons et autres "jolygones"...). Sans oublier les histogrammes, pyramides des âges et lissages de séries temporelles par des moyennes mobiles d'ordre 12, chers aux statisticiens...

Dans l'espace, l'auteur traite complètement deux problèmes essentiels: vues d'un même corps sous différents angles (transformations et changements de repère traités par matrices de passage) et élimination du dessin des parties masquées (algorithme des lignes de crête), avec rappel sous forme simple de la théorie mathématique. Les exemples sont spectaculaires: une même maison vue en perspective sous différents angles, et des surfaces d'équation z=f(x,y)...

Pour adapter les prgrammes donnés dans cet ouvrage à nos HP favoris, il suffit de renuméroter lignes et étiquettes de 1 à 9999 (car l'APPLE va, lui, jusqu'à 99999), et de simuler les fonctions graphiques de base comme suit, avec des CALL SUB (auxquels les arguments sont transmis par liste), en adressant la table traçante par son "accessory ID" (%64), au moyen de l'instruction OUTPUT (en effet, la table contient un convertisseur HP82166, et cette méthode permet de ne pas avoir à la déclarer PRINTER IS INTRFCE (donc de conserver une "vraie" imprimante dans la boucle). Voici quelques SUBs, que je dois en grande partie à Etienne POUPEE, d'ailleurs:

```
SUB MGRAPH @ OUTPUT:%64 ;CHR$(18) ! table>mode graphique
SUB MTEXT @ OUTPUT :%64 ;CHR$(17) ! table>mode texte
SUB COLOR(n) @ CALL G1('C',n) ! n=0(noir), 1(bleu), 2(vert) ou 3(rouge)
SUB LTYPE(x) @ CALL G1('L',x) ! Trait allant du continu(0)
    au pointillé(1) et aux tirets(2 à 15 selon grosseur!)
    Pour ce qui suit, X et Y sont exprimés en "steps" de 0.2mm :
SUB MOVE(X,Y) @ CALL G2('M',X,Y)
SUB DRAW(X,Y) @ CALL G2('D',X,Y)
    Noter que RMOVE et RDRAW sont simulées par le programme lui-même.
SUB G1(C$,X) @ OUTPUT :%64 USING "A,4Z";C$,X
SUB G2(C$,X,Y) @ OUTPUT :%64 USING "A,4Z,',',4Z";C$,X,Y
```

Et voilà. A présent, il est possible de convertir les programmes se trouvant pages 29(pour le plan) et 177(espace). Ces programmes s'utilisent en MERGEant l'un ou l'autre selon le cas à la fin de vos propres programmes, et en utilisant des GOSUB nnnn pour accéder aux différentes fonctions. Mais trève d'explications; voici plutôt quelques belles images, et la version adaptée du premier programme (dans lequel tous les points d'entrée sont numérotés COMME DANS LE LIVRE, à la suppression près du dernier zéro (60000 devient ainsi 6000, de façon à ne pas dépasser le fatidique 9999!). Votre programme dispose donc en propre des lignes 0 à 5999! Une dernière remarque pour les possesseurs d'un HP 75: les SUB sont maintenant possibles avec la ROM I/O, et OUTPUT :%64 s'écrira OUTPUT":I1". Happy Plotting!

AIDE-MEMOIRE COMMANDES TABLES TRACANTES

PRINCIPE: Chaque commande ou groupe de commandes est envoyé à la table sous forme de chaîne de caractères, au moyen d'un ordre PRINT ou, mieux, OUTPUT. Une commande graphique est constituée d'une lettre majuscule, suivie de ses paramètres, qui sont séparés par des virgules.

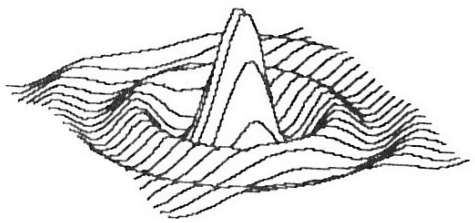
Le papier en rouleau offre 96 mm utiles en largeur, soit 480 "steps" de 0.2 mm, la longueur n'étant pas limitée. Dans ce qui suit, h et v désignent respectivement les déplacements horizontaux ou transversaux (x), et verticaux ou longitudinaux (y), en "steps".

CARACTERES DE CONTROLE: Au nombre de 7, ils commandent certaines fonctions en mode "texte" (commandes graphiques non utilisables):

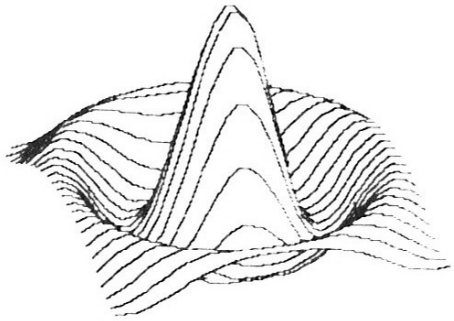
CHR$(8) Backspace 1 caractère
CHR$(10) Saut à la ligne suivante sans retour à la marge (Line Feed, LF)
CHR$(11) Saut à la ligne PRECEDENTE sans retour à la marge (Inverse LF)
CHR$(13) Retour à la marge sans saut de ligne en mode texte (CR, Carriage Return)
CHR$(17) Passage en mode texte (mode par défaut du traceur)
CHR$(18) Passage en mode graphique
CHR$(29) Changement de couleur

COMMANDES EN MODE GRAPHIQUE:

A	Retour en mode texte.
Cn	Sélection couleur (1:noir, 2:bleu, 3:vert, 4:rouge)
Dh1,v1[,h2,v2...]	DRAW ABSOLUTE, trace une droite entre le point courant et le point de coordonnées par rapport à l'origine courante h1,v1 [puis entre ce dernier point et celui h2,v2, et ainsi de suite...]



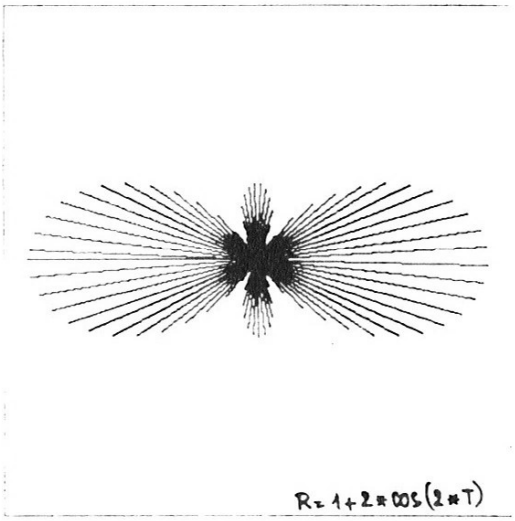
H	Retour à l'origine courante, sans tracer de trait
I	définit la position actuelle du stylo comme nouvelle origine
Jh1,v1[,h2,v2...]	DRAW RELATIVE, tire un trait du point courant au point situé en h1,v1 par rapport à ce point [et ainsi de suite,...]
Ln	Définition du type de trait, n=0(continu), 1(pointillé), 2(tirets fins) à 15(grands tirets)
Mh1,v1	MOVE ABSOLUTE, sans tracé
Pchaine	En mode graphique, écrit à partir de la position courante du stylo la chaîne de caractères
On	Définit la direction dans laquelle s'imprimera une chaaîne avec la commande P: 0:sens normal), 1: vers le basV, 2: à l'envers et de droite à gauche(, et 3: de bas en haut^
Rh1,v1	MOVE RELATIVE, déplacement comme avec J mais sans tracé
Sn	Taille des caractères (n allant de 0:80 car/ligne, à 63: un seul caractère par ligne)

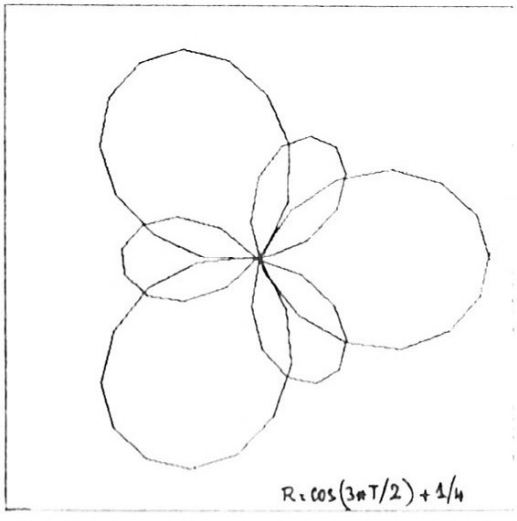


Programme pour HP71

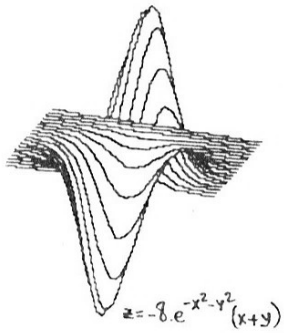
```
100 ! POLRAD, courbes polaires rayonnees.
110 PRINTER IS :INTRFCE
120 OPTION ANGLE RADIANS
130 CALL MGRAPH
140 DISP 'Ligne 200 DEF FNT(R)'
150 WAIT 2 @ DISP 'puis faire RUN 200' @ STOP
200 DEF FNR(T)=1+2*COS(2*T)          -> voir graphe ci-contre
210 DISP 'Donnees RADIANS' @ WAIT 2
220 LINPUT 'F periodique?O/N','O';P$  -> Oui ou Non (Ici, Oui)
225 IF P$='N' THEN 300
230 INPUT 'Valeur per.(ou Ø)';B       -> (pour l'exemple, 2*PI)
235 ! Recherche auto. periode            (Ø si inconnu)
240 X0=FNR(0) @ Y0=0 @ A=0 @ B=2*PI
250 X=FNR(B)*COS(B) @ Y=FNR(B)*SIN(B)
260 IF ABS(X-X0)<.000001 AND ABS(Y-Y0)<.000001 THEN 310
270 B=B+2*PI @ IF B<200*PI THEN 250
300 INPUT 'Intervalle etude [A,B]';A,B  -> (ici, Ø,2*PI)
310 INPUT 'Pas increm.';P               -> (ici, Ø.08)
330 INPUT 'ABS(Rmax) ou Ø';M            -> (ici, 3) (Ø si inconnu)
332 IF M#0 THEN 400
335 ! Rech.fenetre mini.
340 M=0 @ FOR I=A TO B+.5*P STEP P
350 F=FNR(I) @ X=F*COS(I) @ Y=F*SIN(I)
360 IF M<ABS(X) THEN M=ABS(X)
370 IF M<ABS(Y) THEN M=ABS(Y)
380 NEXT I
400 LINPUT 'Cadre?','O';C$            -> Oui ou Non
405 Z=.1*M @ W1=-M-Z @ W2=M+Z @ W3=-M-Z @ W4=M+Z
406 CALL MGRAPH @ CALL COLOR(1)
407 V1=0 @ V2=400 @ V3=0 @ V4=400 @ GOSUB 6000
410 IF C$='O' THEN GOSUB 6350
415 ! Trace courbe
420 FOR I=A TO B+.01*P STEP P
425 W$='U' @ GOSUB 6230 @ W$='D'
430 X=0 @ Y=0 @ W$='U' @ GOSUB 6230
440 F=FNR(I) @ X=F*COS(I) @ Y=F*SIN(I) @ W$='D' @ GOSUB 6230
450 NEXT I
480 BEEP 1000,1 @ DISP 'TERMINE'
490 CALL MTEXT @ FOR I=1 TO 50 @ PRINT @ NEXT I
500 END
```

600 et suivantes : programme DOMY & MERGE2

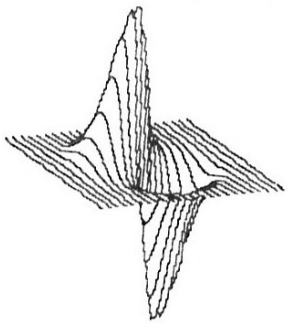


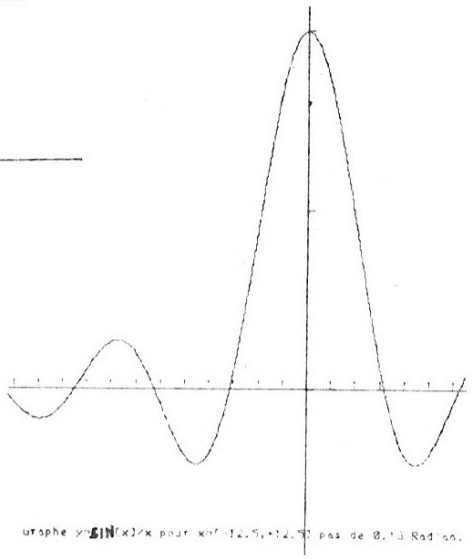


```
5990 ' Programme DONY, adaptation E.GENGOUX et PPCT, 11/1984.
5991 ' MERGEr (sans RENUMéroter) en fin prgm.ppal.
5992 ' Choisir fenêtre(Wi) et clôture(Vi), LTYPE et COLOR,
5993 ' activer SR0. Pour tout point à tracer, passer(X=,Y=),
5994 ' puis W$=Up/Down et GOSUB 6000 (équiv. resp. MOVE/DRAW)
5995 ' SR0 initialisation ligne 6000
E 6000 A8=(V2-V1)/(W2-W1) @ B8=(V1*W2-V2*W1)/(W2-W1)
6010 A9=(V4-V3)/(W4-W3) @ B9=(V3*W4-V4*W3)/(W4-W3) @ RETURN
6090 ' SR1 principale
E 6100 X5=X @ Y5=Y @ GOSUB 6150 @ IF W$='D' THEN 6120
6110 C8$=C5$ @ X8$=C5$ @ X8=X @ Y8=Y @ IF C8$<>'0000' THEN RETURN
6115 X6=X @ Y6=Y @ W$='U' @ GOSUB 6231 @ RETURN
6120 C9$=C5$ @ X9$=C5$ @ X9=X @ C7=X @ Y9=Y @ D7=Y
6125 GOSUB 6200 @ C8$=X9$ @ X8$=X9$ @ X8=C7 @ Y8=D7 @ RETURN
6140 ' SR2 codage binaire
E 6150 G9$='0' @ D9$='0' @ B9$='0' @ H9$='0'
6155 IF X5<W1 THEN G9$='1' @ GOTO 6165
6160 IF X5>W2 THEN D9$='1'
6165 IF Y5<W3 THEN B9$='1' @ GOTO 6175
6170 IF Y5>W4 THEN H9$='1'
6175 C5$=H9$&B9$&D9$&G9$ @ RETURN
6190 ' SR3 Découpage
E 6200 IF C8$='0000' AND C9$='0000' THEN 6215
6201 E9=0 @ FOR K9=1 TO 4 @ X$=C8$[K9,K9] @ Y$=C9$[K9,K9]
6202 E9=E9+VAL(X$)*VAL(Y$) @ NEXT K9
6203 IF E9<>0 THEN RETURN
6204 C7$=C8$ @ IF C7$='0000' THEN C7$=C9$
6205 G9$=C7$[4,4] @ IF G9$<>'1' THEN 6207
6206 Y6=Y8+(Y9-Y8)*(W1-X8)/(X9-X8) @ X6=W1 @ GOTO 6213
6207 D9$=C7$[3,3] @ IF D9$<>'1' THEN 6209
6208 Y6=Y8+(Y9-Y8)*(W2-X8)/(X9-X8) @ X6=W2 @ GOTO 6213
6209 B9$=C7$[2,2] @ IF B9$<>'1' THEN 6211
6210 X6=X8+(X9-X8)*(W3-Y8)/(Y9-Y8) @ Y6=W3 @ GOTO 6213
6211 H9$=C7$[1,1] @ IF H9$<>'1' THEN 6213
6212 X6=X8+(X9-X8)*(W4-Y8)/(Y9-Y8) @ Y6=W4
6213 IF C7$=C8$ THEN X8=X6 @ X5=X6 @ Y8=Y6 @ Y5=Y6 @ GOSUB 6150 @ C8$=C5$ @ GOTO 6200
6214 X9=X6 @ X5=X6 @ Y9=Y6 @ Y5=Y6 @ GOSUB 6150 @ C9$=C5$ @ GOTO 6200
6215 IF X8$<>'0000' THEN X6=X8 @ Y6=Y8 @ W$='U' @ GOSUB 6231
6216 X6=X9 @ Y6=Y9 @ W$='D' @ GOSUB 6231 @ RETURN
6220 ' SR4 Projection et Tracé
E 6230 X6=X @ Y6=Y ' Entrée tracé rapide sans coupage
6231 X5=A8*X6+B8 @ Y5=A9*Y6+B9
6232 IF W$='U' THEN CALL MOVE(X5,Y5) @ RETURN
6233 CALL DRAW(X5,Y5) @ RETURN
6240 ' SR5 Trace axes
E 6250 P9=W1 @ Q9=W3
6251 IF W1<0 AND W2>0 THEN P9=0
6252 IF W3<0 AND W4>0 THEN Q9=0
6253 X6=W1 @ Y6=Q9 @ W$='U' @ GOSUB 6231 @ X6=W2 @ W$='D' @ GOSUB 6231
6254 X6=P9 @ Y6=W3 @ W$='U' @ GOSUB 6231 @ Y6=W4 @ W$='D' @ GOSUB 6231
6255 RETURN
6290 ' SR6 Graduations
E 6300 T8=.01*(W2-W1) @ T9=.01*(W4-W3)
6301 U8=((P9-W1)/U-IP((P9-W1)/U))*U
6302 V8=((Q9-W3)/V-IP((Q9-W3)/V))*V
6303 FOR X6=W1+U8 TO W2 STEP U
6304 Y6=Q9 @ W$='U' @ GOSUB 6231 @ Y6=Q9+T9 @ W$='D' @ GOSUB 6231
6305 NEXT X6
6306 FOR Y6=W3+V8 TO W4 STEP V
6307 X6=P9 @ W$='U' @ GOSUB 6231 @ X6=P9+T8 @ W$='D' @ GOSUB 6231
6308 NEXT Y6
6309 W$='U' @ GOSUB 6231 @ RETURN
6340 ' SR7 Tracé du cadre
E 6350 X6=W1 @ Y6=W3 @ W$='U' @ GOSUB 6231 @ X6=W2 @ W$='D' @ GOSUB 6231
6360 Y6=W4 @ GOSUB 6231 @ X6=W1 @ GOSUB 6231 @ Y6=W3 @ GOSUB 6231 @ RETURN
6400 SUB MGRAPH @ OUTPUT :%64 ;CHR$(18) @ END SUB
6410 SUB MTEXT @ OUTPUT :%64 ;CHR$(17) @ END SUB
6420 SUB COLOR(X) @ CALL G1('C',X) @ END SUB
6430 SUB LTYPE(X) @ CALL G1('L',X) @ END SUB
6440 SUB MOVE(X,Y) @ CALL G2('M',X,Y) @ END SUB
6450 SUB DRAW(X,Y) @ CALL G2('D',X,Y) @ END SUB
6460 SUB G1(C$,X) @ OUTPUT :%64 USING "A,4Z";C$,X @ END SUB
6470 SUB G2(C$,X,Y) @ OUTPUT :%64 USING "A,4Z,',',4Z";C$,X,Y @ END SUB
```



entre $\emptyset$ et 4*PI (M=1.25)





FORTH

STO 8 RCL

Les définitions FORTH que vous trouverez ci-dessous ont pour objet de donner au HP71 muni d'une ROM FORTH l'équivalent des registres de stockage de la 41. On constate en effet que les mots de gestion de la pile XYZTL et de fonctions arithmétiques flottantes donnent au 71 pratiquement toutes les instructions d'une 41, seuls manquaient les registres.

Ces registres font 8 octets (16 nibbles) et sont crées dans un I/O buffer et par conséquent n'utilisent pas de place dans FORTHRAM:

SETREG crée un buffer; par exemple 15 SETREG alloue la place pour 15 registres. On peut allouer au maximum 255 registres si la mémoire disponible le permet.

REG permet d'acceder aux registres ainsi crées, en retournant l'adresse du registre demandé:

10 REG RCL rappelle dans X le contenu du registre X
10 REG STO stocke X dans le registre 10.

Ces deux expressions sont équivalentes sur 41 à RCL 10 ou STO 10.

Les registres sont conservés tant que le sous-programme interne de configuration n'est pas executé ce qui est le cas lors d'un FREE/CLAIM PORT ou au démarrage de la machine. Il est possible de préserver le buffer grâce à VARID (REGBUF VARID !); inversement on peut les détruire par REGBUF @ KILLBF.

Bien entendu, vous êtes prévenus si l'opération demandée n'est pas possible. Par rapport à la 41 il ne manque que l'équivalent de STO/RCL IND n que je vous laisse le soin de réaliser.

Bon travail!

Janick TAILLANDIER T8 CCHU178
9 rue Delezenne
59000 LILLE

```
( Pseudo HP41 registers

HEX

VARIABLE REGBUF VARIABLE NUMREG 0 REGBUF ! 0 NUMREG !

( SETREG create register buffer )
: SETREG ( n -- )
  DUP 0< IF ABORT" Invalid reg #" THEN
  DUP NUMREG !
  10 * MAKEBF
  IF REGBUF ! DROP
  ELSE ABORT" Create error"
  THEN ;
```

```
( FINDREG find REG in buffer )
  FINDREG ( n -- addr )
  REGBUF @ FINDBF DUP
  IF SWAP 10 * +
  ELSE ABORT" No register"
  THEN ;

( REG find address of the register )
  REG
  DUP 0( IF ABORT" REG ( 0" THEN
  DUP NUMREG @ )
  IF ABORT" Nonexistent"
  ELSE FINDREG
  THEN ;

DECIMAL
```

FTHUTIL

HP vient de sortir aux USA un "HP-71 software developer's handbook", référence non communiquée. Il coûte $50 avec une cassette contenant un certain nombre de fichiers LEX. Je n'ai pas fini de lire celui que j'ai dans les mains. Mais je peux déja dire que c'est du très bon travail. On peut dire qu'il s'agit d'un guide de la bonne façon de programmer le HP-71. Hélas il est destiné à des professionnels (d'où son prix) et il est doûteux qu'il soit jamais traduit en Français. Il contient beaucoup d'éléments de FORTH, je vous en livre quelques-uns ci-dessous, qui devraient pouvoir être utilisés par tous. Disons quand même pour ceux qui ne connaissent pas le HP-71 que cet appareil permet des transferts de pointeur entre FORTH et BASIC et réciproquement, d'où les "BASICX...". Ca, ce sera difficile à "porter" sur d'autres appareils.

```
(                          FTHUTILF                                    )
( FORTH Utilities   Secondaries.  FTHUTILA file should assembled prior to )
(     loading this file.                                                 )
BASE @

( PAUSELEN: Variable to hold pause length )
VARIABLE PAUSELEN 0 PAUSELEN !

( PAUSE : Pause for PAUSELEN/1000 seconds )
: PAUSE PAUSELEN @ 1+ 0 DO LOOP ;

( N-A: convert a nibble to its hex ascii equivalent.  n -> )
DECIMAL
: N-A 10 - DUP 0< IF 58 + ELSE 65 + THEN EMIT ;

: DUMP OVER + SWAP ( addr n -> print n nibs starting at addr )
    DO I N@ N-A LOOP ;

: DUMP+ 2DUP + ROT ROT DUMP ; ( Do DUMP, leave next addr on stack )

( SHOW: disp contents of n consecutive memory cells.  ( addr n -> ) )
: SHOW 1+ 1 DO DUP H. DUP @ 5 SPACES H. PAUSE CR 5+ LOOP ;

: BASE? BASE @ DUP DECIMAL . BASE ! ; ( put current base in X and display)
```

```
( DELAY00: Set DELAY0,0)
: DELAY00 " DELAY0,0" BASICX ;

( D- {D P R *}: Do DISPLAY IS {DISPLAY  PRINTER RS232 *} )
: D-P " DISPLAY IS PRINTER" BASICX ;
: D-* " DISPLAYIS*" BASICX ;
: D-D " DISPLAYISDISPLAY" BASICX ;
: D-R " DISPLAYISRS232"BASICX ;

( ROOM?:  Display number of nibbles available in dictionary )
: ROOM? SP0 @ HERE - 458 - . ;

( S.: Print stack contents bottom first )
: S. ." [ " DEPTH 0> IF DEPTH 1 SWAP DO I PICK U. -1 +LOOP THEN ." ] " ;

HEX
( ADDR-     get the addr of the namefield of the previous word.)
(           NFAddr1 -> NFAddr2 )
: ADDR- 5- @ ;

( NFA:  GIVEN CFA, GET NFA.  [ CFA -> NFA ] )
: NFA 2- -1 TRAVERSE ;

( NAME: From NFA, type name  [ nfa -> ] )
: NAME DUP C@ 1F AND 2DUP 1- SWAP 2+ SWAP TYPE
      2* + C@ 7F AND EMIT ;

( NFASTR: Convert NFA to name string.  [NFA -> str] )
: NFASTR DUP 2+ SWAP C@ 1F AND ;

( SPECIAL: Array containing list of words with remote CFA's. )
( 1st value is # of entries. )
CREATE SPECIAL
   D ,
   E701A , ( COLON )

  E71E8 , ( SEMI )
  E1C54 , ( number )
  E22ED , ( F-number )
  E1C67 , ( DO )
  E3FF1 , ( LOOP )
  E3F81 , ( +LOOP )
  E5D86 , ( IF / UNTIL / WHILE )
  E5D89 , ( ELSE / REPEAT )
  E0640 , ( " )
  E0EFA , ( ." )
  E580E , ( ABORT")
  E0168 , ( J )

( SPEC?: Find CFA in SPECIAL. Return # of entry, or 0 if not present. )
( [ cfa -> cfa # ] )

: SPEC? 0 SPECIAL @ 1+ 1 DO  ( CFA 0 )
   OVER SPECIAL I 5 * + @ =  ( CFA 0 CFA SPEC1 )
      IF DROP I LEAVE THEN LOOP ; ( CFA # )

( 'NAME: Given CFA, type name.  { CFA -> } )
: 'NAME DUP NFA SWAP OVER - 2/ 1- SWAP NFASTR DUP 4 ROLL =
      IF 1- 2DUP TYPE 2* + C@ 7F AND EMIT
      ELSE 2DROP ." Unknown" THEN ;

( HEREN: Find the start of the n+1th link field.  { n -> addr } )
: HEREN DUP C = IF DROP E6FAB ELSE 1+ 5 * E0000 + @  ( NFA )
   BEGIN 5- DUP @ DUP 0 <> WHILE SWAP DROP REPEAT DROP THEN ;
```

etc

HP-75

+8K

CONCEPTION D'UN MODULE RAM 8K POUR LE HP-75

L'objet de cette étude est la conception d'un module RAM 8 Koctets complètement interchangeable avec le module d'extension vendu par Hewlett-Packard, mais à un prix bien inférieur à 1000 F contre plus de 2000 F pour l'accessoire d'origine H.P.

DESCRIPTION DU CONNECTEUR INTERNE DU HP-75

Le connecteur de ce module est accessible à l'intérieur du 75 en enlevant la batterie. Il est réalisé à même le circuit imprimé et comporte 14 contacts dorés sur chaque face, au pas de 1/10 de pouce (2,54 mm). On considère la machine retournée, le clavier contre la surface de travail. Les contacts du DESSUS sont tournés vers le FOND de l'ordinateur, ceux du DESSOUS, vers le CLAVIER. Le contact du DESSUS le plus vers l'INTERIEUR de la machine porte le numéro 1 (voir figure 1).

Description des broches

désignation	numéro	commentaire
VCC	1,15	Alimentation positive (environ 6 Volts)
VDD	2,16	Masse électrique
R/W	5	Mode lecture écriture (0=écr.,1=lec.)
CS	3,4,17,18	Sélection boitier (0=sél.,1=non sél.)
A	6,7,8,9,10, 19,20,21,22 23,28	Lignes d'adresse (11) espace adressable 2 Koctets
D	11,12,13,14 24,25,26,27	Bus de données (largeur 8 bits)

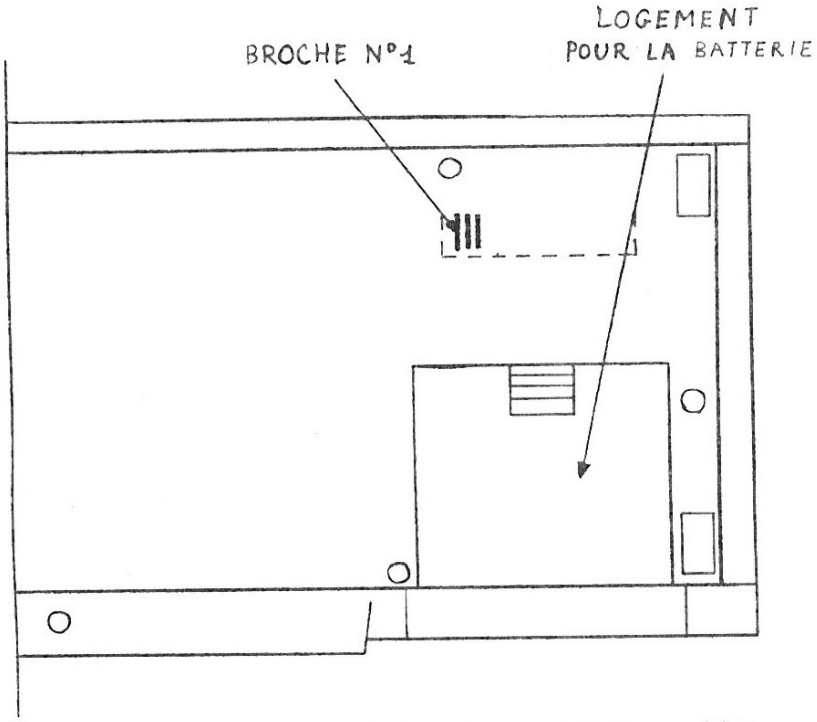
CONCEPTION DU MODULE

On remarque l'existence de 4 lignes de sélection de circuit car à l'époque de la conception du HP-75, les mémoires CMOS monolithiques de 8 Koctets n'étaient pas encore disponibles. A priori le module H.P. est constitué d'un assemblage de 4 puces de 2 Koctets, chacune d'entre elles possédant une entrée de sélection de circuit (la mémoire interne du 75 est constituée par 8 circuits de 2 Koctets). Il s'agit donc de créer à partir de ces 4 lignes de sélection de circuit deux lignes d'adresse et une ligne de sélection du boitier 8 Koctets employé (voir figure 2). On utilise 8 portes ET (2 circuits CMOS 74HC08 à environ 10 F pièce), 1 mémoire RAM CMOS 8 Koctets (environ 350 F) 1 connecteur encartable 2*14 contacts au pas de 2,54 mm. La place dans le compartiment batterie étant très limitée, le module a été realisé en soudant la RAM et les deux autres circuits de part et d'autre d'une plaquette percée et pastillée, en epoxy, pour le wrapping. On emploit pour les liaisons entre les circuits et le connecteur encartable du fil pour le wrapping, par exemple.

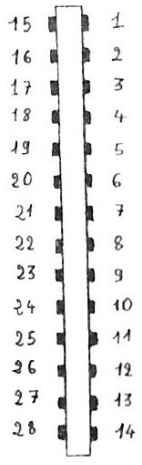
Le montage proposé est assez délicat du fait de la faible taille du module. Cependant, son prix de revient est intéressant (environ 600 F) par rapport au coût du module d'origine H.P.

V. DELORME (T350)

FIGURE 1



VUE DU HP-75 RETOURNE
(VUE DU DESSUS DU CONNECTEUR)

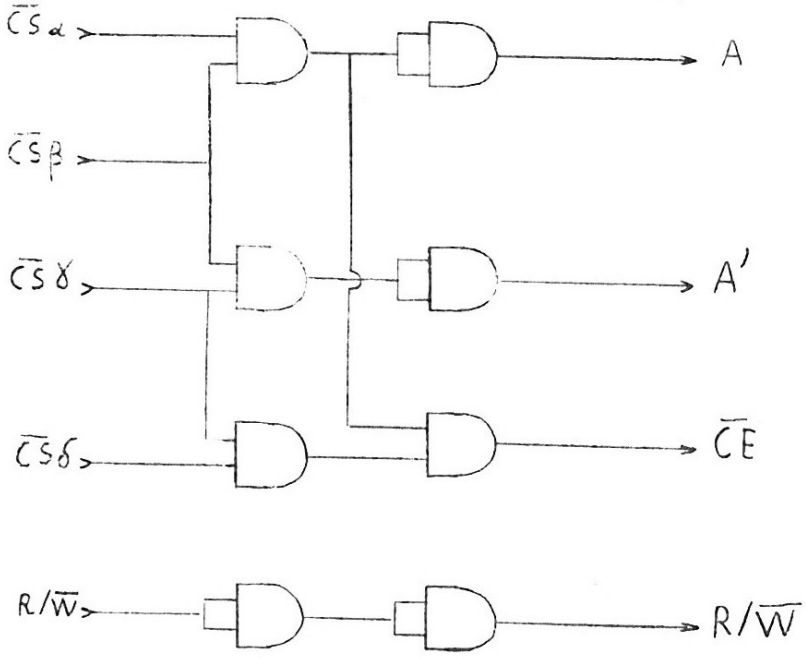


1	Vcc	15	Vcc
2	V_{DD}	16	V_{DD}
3	CS3	17	CS2
4	CS1	18	CS0
5	R/$\overline{W}$	19	A9
6	A8	20	A7
7	A6	21	A5
8	A4	22	A3
9	A2	23	A1
10	A0	24	D7
11	D6	25	D5
12	D4	26	D3
13	D2	27	D1
14	D0	28	A10

CONNECTEUR
INTERNE

FIGURE 2

SCHEMA LOGIQUE DU DECODAGE

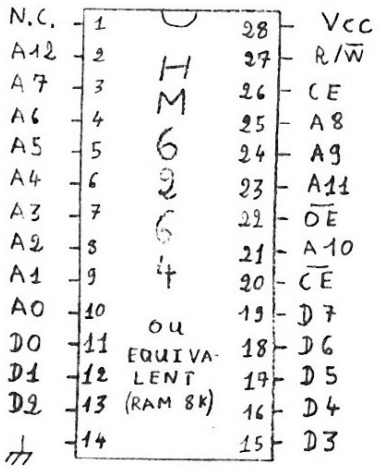


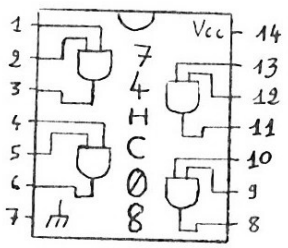
LES LETTRES $\alpha, \beta, \gamma, \delta$ INDIQUENT QU'IL FAUT UTILISER LES 4 LIGNES $\overline{CS}$ MAIS DANS N'IMPORTE QUEL ORDRE.

DE MEME A ET A' PEUVENT ETRE RELIEES A N'IMPORTE QUELLE BROCHE D'ADRESSE DE LA RAM 8 K

DU CONNECTEUR

VERS LA RAM 8K





RELIER LES BROCHES PORTANT LA MEME LETTRE A CELLES DU CONNECTEUR (LA CONCORDANCE DES CHIFFRES EST SUPERFLUE POUR L'APPLICATION DECRITE)

RELIER BROCHES 28 (Vcc) A 26 (CE)
RELIER BROCHES 22 ($\overline{OE}$) A 20 ($\overline{CE}$)

ROMS

AU FOND DES ROMS

Après la publication par le club PPC du listing des ROMs internes du HP-75, le dernier coin de voile restant à soulever à propos de cette machine cachait pudiquement le contenu des ROMs externes : MATH, VISICALC, TEXT FORMATTER et DATACOMM.

Etant par nature excessivement curieux, les mystères entourant ces ROMs me causaient presque des insomnies, et, c'est effectivement après quelques nuits blanches, que les programmes suivants ont été mis au point.

La première difficulté à résoudre était le transfert des octets des ROMs externes vers la RAM. Pour surmonter ce problème, c'est une LEXfile qui a été écrite : ROMPEEK. Cette LEXfile apporte une nouvelle fonction BASIC dénommée elle aussi ROMPEEK qui permet de lire un octet dans une ROM externe.

Il convient de rappeler ici que les ROMs externes partagent, avec deux ROMs internes, 8K de mémoire entre les adresses H6000 et H7FFF inclusivement ; les deux ROMs internes s'appellent "altrom" et "melrom". MELROM contient l'essentiel du code nécessaire à la gestion des communications HP-IL et ALTROM contient pour la plus grande part le code de ce que HP appelle "Pocket Secretary", c'est à dire la gestion du mode APPT.

Chaque appel à une routine résidant entre H6000 et H7FFF nécessite que l'on précise outre l'adresse de cette routine, l'identification de la ROM dans laquelle se trouve cette routine afin de pouvoir valider cette ROM et masquer en quelque sorte tout autre code se situant entre ces deux adresses.

Il existe trois routines internes qui permettent d'appeler un sous-programme entre H6000 et H7FFF ; ce sont :

- SYSJSB = adr. 466D ; permet d'appeler une routine dans ALTROM
- MELJSB = adr. 4661 ; permet d'appeler une routine dans MELROM
- ROMJSB = adr. 4694 ; permet d'appeler une routine dans n'importe quelle ROM,

qui sont décrites p.458 et 459 de PPC 75 NOMAS VOL. I.

Une première conclusion s'impose donc : pour lire quelque chose dans une ROM externe il faut la valider.

Un examen attentif des routines permettant les manipulations sur les ROMs paginées, c'est à dire validable entre H6000 et H7FFF, montre qu'il existe une routine appelée NXTROM (H47AC) qui valide les pages des ROMs à partir d'un pointeur dans la zone de RAM appelée ROMTAB. ROMTAB est une table d'adresses de validation des ROMs connectées dans le HP-75 qui est créée ou recréée à chaque allumage (warmstart ou coldstart) par la routine ROMSET (p.463). Ces adresses sont du type FFxx :

```
FF30   port # 3  première page de 8K
FF32   port # 1          - " -
FF34   port # 2          - " -
FF38   port # 3  deuxième page de 8K
FF3A   port # 1          - " -
FF3C   port # 2          - " -
FF40   port # 3  troisième page de 8K
FF42   port # 1          - " -
FF44   port # 2          - " -
FF46             altrom
```

```
FF48   port # 3  quatrième page de 8K
FF4A   port # 1          - " -
FF4C   port # 2          - " -
FF4E             melrom
```

En fonction des ROMs trouvées par le système à l'allumage, la table ROMTAB contiendra certaines de ces adresses plus, obligatoirement, à la fin, les trois adresses suivantes :

[0001] = système
[FF46] = altrom1
[FF4E] = melrom

A titre d'exemple, si l'on a :

MATH dans le port#1 (à gauche) = 16K (pages 3 et 4 : FF4A, FF42)

VISICALC dans le port#2 (au milieu) = 32K (FF4C, FF44, FF3C, FF34)

TEXT FORMATTER dans le port#3 (à droite) = 8K (page 3 : FF40)

ROMTAB contiendra (début à l'adresse H8520), les adresses étant rangées dans l'ordre :

```
8520   FF34   port#2
8522   FF3C   port#2
8524   FF40   port#3  page 3
8526   FF42   port#1  page 3
8528   FF44   port#2
852A   FF4A   port#1  page 4
852C   FF4C   port#2
852E   0001   système
8530   FF46   altrom
8532   FF4E   melrom
8534   0000   fin de ROMTAB
```

Le décor étant maintenant à peu près planté -évitez trop de questions : certaines choses dans le domaine étudié ici ne s'expliquent pas ; c'est en particulier le choix des adresses de validation ...-, voyons comment d'une façon pratique on peut aller au fond des ROMs.

Ainsi que cela a été indiqué plus haut, tout repose sur la routine NXTROM qui demande simplement dans les registres R20/R21 un pointeur dans ROMTAB, c'est à dire une adresse comprise entre H8520 (début de ROMTAB) et H8548 ; ceci correspond théoriquement à 21 fois 8K ce qui est un peu plus que la pratique qui, elle, correspond à 3 fois 32K + 3 fois 8K pour les ROMs systèmes + 1 adresse nulle indiquant la fin de la table, soit 128K. Il se peut néanmoins qu'il existe d'autres adresses de validation (FF5x par exemple ?) qui permettent de pouvoir "monter" chaque port jusqu'à 48K ...

Le principe de la LEXfile ROMPEEK est donc le suivant :

- acquisition d'une adresse (comprise entre H6000 et H7FFF),
- acquisition d'une adresse à l'intérieur de ROMTAB,
- appel à NXTROM pour validation de 8K de ROM correspondant à l'adresse dans ROMTAB,
- transfert de la valeur de l'octet à l'adresse spécifiée dans un registre CPU tant que les 8K de ROM sont validés,
- validation des ROMs-système,
- résultats.

La fonction ROMPEEK apportée par la LEXfile du même nom, a la syntaxe suivante :

ROMPEEK (<adr.>,<pointeur>)

- <adr.> est la valeur décimale de l'adresse comprise entre H6000 et H7FFF inclusivement (soit entre 24576 et 32767).
- <pointeur> est la valeur décimale de l'octet le moins significatif de l'adresse dans ROMTAB qui doit être paire (soit une valeur comprise entre H20=d32 et H48=d72, inclusivement).

Trois erreurs peuvent être générées par cette LEXfile :

250 = "wrong ROM address" : <adr.>inf.H6000 ou sup.H7FFF

251 = "invalid ROM #" : <pointeur>inf.32d ou sup.72d ou impair

252 = ("end of ROMTAB") : la valeur du pointeur bien que valide ne correspond pas à une page existante (le message d'erreur n'est pas affiché, seulement le no., c.à.d. 252).

La source commentée de cette LEXfle figure en principe ci-après.

Le seul problème qui reste malgré tout, c'est que l'on ne sait pas en donnant une valeur de pointeur dans quelle page de ROM on va lire ...
Cette remarque a amené l'écriture d'un programme BASIC appelé ROMDIR qui permet d'obtenir le catalogue des différentes pages existantes et qui, par conséquent, permet de savoir ce que l'on fait. De plus, et pendant que l'on dechiffrait les en-têtes des ROMs, ce programme ROMDIR indique aussi pour chaque page de 8K de ROM, les programmes qui y sont contenus, et, nec plus ultra, donne aussi la possibilité à chaque fois de transférer un ou plusieurs programmes vers une unité de mémoire de masse et donc par conséquent, de pouvoir dans un deuxième temps lister les programmes BASIC ou désassembler les programmes en assembleur (comme MATH ou VISICALC...). Le programme de désassemblage existe et fonctionne très bien : il fera l'objet d'un prochain article.

ROMDIR a été développé sans le module I/O et pourrait en utilisant certaines des fonctions de ce module être probablement raccourci ; toutefois on ne pourra pas le faire tourner à partir du PMS.

On trouvera ci-après le listing et le mode d'emploi de ROMDIR. Ceux que ces deux programmes interessent peuvent m'envoyer soit une cassette initialisée soit quatre cartes magnétiques accompagnées de Frs. 10,00 pour la réexpédition du support magnétique adressé.

Jean-Loup HOUDRY /T239
51, Bd. MURAT
75016 PARIS

PEEKROM 200 Bytes:

Line	Data															Check
1	BB	92	B6	00	BD	4C	B1	D8	6B	9F	50	45	45	4B	52	4F	4D	20	0D	31	18
2	0C	00	14	00	10	00	1C	00	4B	00	00	00	4E	00	00	00	FF	FF	52	4F	87
3	4D	50	45	45	CB	FF	FA	77	72	6F	6E	67	20	52	4F	4D	20	61	64	64	77
4	72	65	73	F3	69	6E	76	61	6C	69	64	20	52	4F	4D	20	A3	65	6E	64	34
5	20	6F	66	20	52	4F	4D	54	41	C2	FF	61	17	9E	20	2D	CE	8B	3E	FB	56
6	0B	66	C9	49	00	FB	05	C9	20	00	FB	05	CE	99	4C	FB	9E	12	A2	53	C7
7	A8	85	CE	CD	3E	70	20	A1	5E	A9	FF	5F	CE	8A	01	99	60	30	C5	FB	E8
8	0D	5E	A9	FF	7F	CE	8A	01	99	60	30	C5	FB	05	CE	99	4C	FA	9E	60	8E
9	30	A1	CE	61	46	FA	7D	50	12	A0	51	13	A0	CE	AC	47	F6	0E	5E	93	82
10	21	A4	CE	1D	48	CE	8A	01	60	0A	E5	9E	CE	99	4C	FC	CE	1D	4D	9E	CC
11	55DA																				

" RONDIR "

Ce programme, décrit ci-après, fournit :
- la liste de toutes les pages de 8K octets des ROM connectées dans le HP-75,
- le catalogue des programmes contenus dans chaque page de 8K,
- la possibilité de transférer les programmes (BASIC ou LEX) contenus dans chaque ROM vers une unité de mémoire de masse.

Pour pouvoir "tourner", ce programme requiert que les fichiers LEX suivants soient présents en RAM :
- MEMLEX pour PEEK(#)
- ROMPEEK pour ROMPEEK (#,#)
- HPILCMDS pour SENDIO $,$,$ et ENTIO$($,$)
- DISPLEX pour CR$ et INV$($)

Il convient de noter que le dernier fichier n'est pas obligatoire et qu'il est possible de remplacer CR$ par CHR$(13) et INV$($) par CHR$(#) avec la valeur correspondante à l'inverse vidéo du caractère concerné.
Le programme utilise aussi quelques fonctions de la ROM MATH : BVAL($,#), BSTR$($,#), et MAT x = ZER. Toutefois il est possible encore de suppléer à ces fonctions de conversion ainsi qu'à MAT x = ZER assez facilement.

Une fois le programme chargé en RAM dans un environnement fonctionnel et lancé, il fonctionne tout seul.

INFORMATIONS FOURNIES PAR LE PROGRAMME :

Pour chaque page de 8K octets, on obtient sur l'imprimante les informations suivantes :
- "Slot address" : adresse de validation, de la forme FFxx ;
- ROMTAB : valeur du pointeur dans ROMTAB, comprise entre H8520 et H8548, correspondant à la page de ROM ayant l'adresse de validation "Slot address" ;
- puis pour chaque programme présent dans cette page de ROM :
  - no. d'entrée (ENTRY #)
  - nom (NAME)
  - Type : B=basic, L=LEX, 0=system lex ou "null lex"
  - Adresse (Address) comprise entre H6000 et H7FFF
  - Longueur (Length) : nombre d'octets en hexadécimal.

Après ces renseignements, il est demandé si l'on veut le "dump" d'un programme de cette page de ROM sur mémoire de masse :

"Dump PRGR on K7, Y/N"

si la réponse est "N" on passe à la page de ROM suivante,
si la réponse est "Y", le nom des différents programmes sont affichés successivement pendant 3 secondes :

"Which PRGR ? : xxxxxxxx"

Il suffit alors d'appuyer sur [RTN] pour lancer le transfert du programme dont le nom figure à l'affichage. Avant de lancer le transfert proprement dit, le programme va générer une entrée au catalogue du support de mémoire de masse puis imprimer le message ci-après :

```
Programme requested :  xxxxxxxx
Catalog :   T = x
            R = xxx
            B = xxx
Address :   T = x
            R = xxx
```

avec :
T = no. de piste du support de mémoire de masse
R = no. d'enregistrement du support de mémoire de masse
B = no. d'octet dans l'enregistrement "R"

Il est ensuite demandé le type d'accès au programme :

"Access mode : RXELPCNT"

Le curseur se place alors sur le "R" ; suivant le type d'accès voulu, on tape à la place de chaque lettre un "0" ou un "1" :

```
R :  1 = RAM file ; 0 = ROM file
X :  1 = executable (BASIC) ; 0 = non executable
E :  1 = editable ; 0 = non editable
L :  1 = listable ; 0 = non listable
P :  1 = purgeable ; 0 = non purgeable
C :  1 = copyable ; 0 = non copyable
N :  1 = line numbered ; 0 = not line numbered
T :  1 = Token file (LEXfile) ; 0 = not token file
```

Ainsi pour un programme BASIC on aura : 11111110 (=HFE) et pour un fichier LEX : 10001101 (=H8D).

Le programme, une fois ces données acquises procède au transfert en affichant, avec un "beep" à chaque enregistrement (256 octets) du support de mémoire de masse, la progression du transfert :

"Recd. : x/n R2 = hh"

avec :

n = nombre total d'enregistrements nécessaires,
x = compteur des enregistrements
R2 = valeur du no. d'enregistrement en hexadécimal.

Quand le transfert est fini (cela peut prendre un petit moment ...), le programme demande si l'on souhaite le transfert d'un autre programme de la page courante.

REMARQUES :

Pour les fichiers LEX, il y a relativisation automatique des adresses des tables figurant dans le noyau du programme : RUNTAB, ASCIIS, PARTAB, ERRMSG et INTCPT, ainsi que des adresses des RUNTIME et PARSE routines (du premier octet du programme jusqu'à "FF FF", juste avant la table ASCIIS).

INFORMATIONS SUR LA STRUCTURE DES ROM :

Les ROMs externes du HP-75 sont toutes organisées par page ou bloc de 8K octets que nous appelerons dorénavant "ROM" afin de simplifier l'exposé.

Chaque ROM commence par deux octets (RMHEAD) : hE3, h1C situés en h6000 et h6001, qui sont donc les mêmes pour toutes les ROMs et qui signalent au système (routine CHKADD à h4836) qu'il existe bien une ROM à l'adresse de validation considérée.
A l'allumage de la machine, le système par le biais de la routine ROMSET (à h47ED) considère successivement toutes les adresses possibles de validation (de hFF30 à hFF5E) et, après validation de chacune de ces adresses lit les deux premiers octets situés à partir de l'adresse h6000 (routine CHKADD, appelée par ROMSET) ; si ces octets sont effectivement hE3, h1C, il y a une ROM à l'adresse de validation considérée, et cette adresse est alors placée dans ROMTAB à la suite des autres adresses déjà trouvées, sinon, il ne se passe rien et on poursuit le "scanning". La validation se fait en écrivant l'adresse de validation (FFxx) à l'adresse FFxx (par une instruction "STBD R#,R#", "#" étant un no. pair de registre du CPU ; le no. impair suivant contient FF). En fait semble-t-il, c'est le fait d'écrire quelque chose à cette adresse qi assure la validation et non ce qu'on écrit en particulier. Notons que HP appelle "Slot address" ou plus simplement "SLOT" ces adresses de validation. Les deux adresses hFF46 et hFF4E, correspondant respectivement à "altrom" et "melrom", sont traitées à part et sont omises lors de l'analyse séquentielle des adresses de validation.

Ensuite, à partir de l'adresse h6002, on trouve le catalogue de la ROM considérée. Ce catalogue comporte au moins une entrée qui correspond alors à une LEXfile car chaque ROM contient

impérativement **une seule** LEXfile qui donne son no. d'identification à la ROM. Le cas échéant cette LEXfile peut être vide ("null lex"). S'il existe plus d'une entrée au catalogue, les autres entrées correspondent à des programmes BASIC ou autres, mais plus à des fichiers LEX.
Chaque entrée du catalogue comporte 18 octets, tout comme les entrées du catalogue RAM. La structure des entrées des catalogues ROM et RAM est identique :

```
Octets Nos.
-----------
    0/1  = adresse du fichier
    2/3  = longueur du fichier en octets
     4   = code d'accès (type RXELPCNT : voir ci-avant)
     5   = type (B,L,I,A,R,T,V,?)
    6/9  = date de création (nbre de secondes depuis 1900)
   10/17 = nom du fichier (8 octets)
```

La signification des caractères indiquant le type est la suivante : B = Basic, L = LEXfile, I = LIF (communication), A = Rendez-vous, R = ROM image, T = Texte, V = Volatile et "?" = type étrange ...

La fin du catalogue est indiquée par deux octets nuls suivant immédiatement la dernière entrée du catalogue. Après ces deux octets nuls, se trouve le premier fichier de la ROM ; ce fichier est, comme cela a déjà été dit, un fichier LEX qui est donc la LEXfile de la ROM. Si ce fichier est vide ("null lex"), il comprend cependant 17 octets :

```
  Adres.     Octet no.
--------     ---------
  xx,60        0/1      = Mybpgm# : LEXfile ID = ROM ID
  xx,60        2/3      = DEF RUNTAB
  xx,60        4/5      = DEF ASCIIS
  xx,60        6/7      = DEF PARTAB
  xx,60        8/9      = DEF ERRMSG
xx+4,60       10/11     = DEF INTCPT
  FF,FF       12/13     = Relocation marker (RELMAR)
                          Nota : l'octet no. 12 est à h60xx
    yy          14      = révision
    zz          15      = Attribut
    9E          16      = RTN
```

L'octet d'attribut des fichiers LEX se trouve toujours à (INTCPT-1) et est construit de la façon suivante :

```
Bit no.
-------
   0 = RAMable (bit 2 doit être aussi égal à 1)
   1 = ROMable
   2 = Position independent code
   3 = Mergeable
   4 = LEX-ID independent code
   5 |
   6 |= réservé pour usage futur
   7 |
```

Les fichiers LEX en ROM utilisent des adresses absolues et, si l'on veut faire passer un fichier LEX de ROM en RAM via la mémoire de masse, il faut relativiser toutes les adresses utilisées par rapport au début du fichier ; ceci est fait automatiquement par ROMDIR pour les adresses figurant dans les tables du début du fichier (RUNTAB, ASCIIS, PARTAB, ERRMSG et INTCPT). Toutefois ROMDIR, pour l'instant, ne modifie pas les adresses qui se trouvent avec des instructions du type JSBX= et LDMDX= ; aussi convient-il d'être prudent et de ne pas faire appel pour exécution à un tel fichier LEX copié en RAM sans s'être assuré que ce genre d'adresse est valable. Aussi est-il recommandé de désassembler préalablement un tel fichier LEX et de faire manuellement les modifications nécesaires en utilisant les instructions BASIC "ADDR($)" et "POKE #,#" avant de faire appel à une des nouvelles instructions qu'apporte ce fichier.

Pour terminer, on trouvera ci-dessous deux configurations de début de ROM, la première avec une seule entrée au catalogue, la seconde avec deux, la deuxième entrée dans ce dernier cas étant un programme BASIC.

a/- Une seule entrée :

Adresse	Valeur de l'octet à l'adresse
6000	E3
6001	1C
6002/6013	Entrée catalogue
6014	00
6015	00
6016/7FFF	Fichier LEX

b/- Deux entrées (le fichier LEX étant vide)

Adresse	Valeur de l'octet à l'adresse
6000	E3
6001	1C
6002/6013	Entrée catalogue no.1 (= "null lex")
6014/6025	Entrée catalogue no.2 (programme BASIC)
6026	00
6027	00
6028/6029	LEX ID
602A/602B	34,60 = DEF RUNTAB
602C/602D	34,60 = DEF ASCIIS
602E/602F	34,60 = DEF PARTAB
6030/6031	34,60 = DEF ERRMSG
6032/6033	38,60 = DEF INTCPT
6034/6035	FF,FF = RELMAR
6036	xx = Révision
6037	1F = Attribut (= 00011111 b)
6038/6042	Programme Control Block (10 octets) du programme BASIC
6043	Début du programme BASIC proprement dit

```
Slot address = FF30  ROMTAB = 8520
Entry #   Name     Type Address  Length
   1     vcassem3  L     6028    12C2
   2     vcbasic3  B     72EA    0CD9

Slot address = FF3A  ROMTAB = 8522
Entry #   Name     Type Address  Length
   1     iorom1    L     6016    1FE6

Slot address = FF40  ROMTAB = 8524
Entry #   Name     Type Address  Length
   1     vcassem1  L     6016    1FE9

Slot address = FF42  ROMTAB = 8526
Entry #   Name     Type Address  Length
   1     iorom2    L     6016    1FE8

Slot address = FF44  ROMTAB = 8528
Entry #   Name     Type Address  Length
   1     math1     L     6016    1FE9

Slot address = FF48  ROMTAB = 852A
Entry #   Name     Type Address  Length
   1     vcassem2  L     6016    1FE9

Slot address = FF4A  ROMTAB = 852C
Entry #   Name     Type Address  Length
   1     iorom3    L     6016    1FE4
```

```
100 OPTION BASE 1
110 DIM V(144),A(8),L(8),T(8)
120 DIM N$[64],N1$[8],T$[1],S$[4],A1$[4],L1$[4]
125 DIM V$[256],A$[256]
130 B1=BVAL("6002",16)
140 ON ERROR GOTO 855
150 FOR N1=32 TO 72 STEP 2 @ DELAY 1
300 J=0 @ I1,J1=1
305 MAT V=ZER @ MAT A=ZER
310 GOSUB 900
315 V(J1)=ROMPEEK(B1+J,N1)
320 J=J+1 @ J1=J1+1
325 IF MOD(J1,18)=1 THEN GOSUB 900 ELSE 315
330 IF A1 THEN 315 ELSE I1=I1-1
340 N2=BVAL('85',16)*256+N1
350 S=PEEK(N2)+256*PEEK(N2+1)
360 S$=FNH$(S,4)
370 PRINT "Slot address = ";S$;"  ROMTAB = ";FNH$(N2,4)
380 PRINT "Entry #   Name     Type Address  Length"
390 N$=""
400 FOR I=1 TO I1 @ K=18*(I-1)
410 A1$=FNH$(A(I),4)
420 T(I)=V(K+6) @ IF T(I)=0 THEN T$="0" ELSE T$=CHR$(T(I))
430 L(I)=V(K+3)+256*V(K+4) @ L1$=FNH$(L(I),4)
440 FOR K1=11 TO 18 @ N$=N$&CHR$(V(K+K1)) @ NEXT K1
450 N1$=N$[1+8*(I-1),8+8*(I-1)]
460 PRINT USING 870 ; I,N1$,T$,A1$,L1$
470 NEXT I
480 PRINT CR$
490 GOSUB 1080 @ DISP "Dump PRGR on K7 : ";INV$('Y')&'/'&INV$('N')
495 B$=KEY$ @ IF B$="" THEN 495
496 IF B$='Y' THEN GOSUB 1090 @ GOTO 500
497 IF B$='N' THEN 850 ELSE 495
500 FOR I=1 TO I1 @ K=8*(I-1)
505 DELAY 0 @ BEEP @ DISP "Which PRGR. ? :";
510 DISP N$[1+K,8+K] @ ON TIMER # 1,3 GOTO 530
520 B$=KEY$ @ IF B$=CHR$(13) THEN 545 ELSE 520
530 DELAY 1 @ OFF TIMER # 1 @ NEXT I
540 GOTO 850
545 OFF TIMER # 1 @ B1$=UPRC$(N$[1+K,8+K])
550 T2,R2=0 @ B3=MOD(18+L(I),256) @ B4=(18+L(I))\256
560 IF B3 THEN B4=B4+1
570 GOSUB 1060 @ C=NUM(A$[20,20])
580 R1=C+2 @ T1=0 @ T$=CHR$(T(I))
590 FOR R2=2 TO C+1 @ GOSUB 1060
600 FOR K=1 TO 8 @ B=1+32*(K-1)
610 IF A$[B,B+7]#"" THEN 670
620 IF K<8 THEN 650
630 IF R2#C+1 THEN 650
640 GOSUB 1080 @ DISP TAB(9);"Directory full" @ GOTO 850
650 IF R1+B4<513 THEN T1=R1\256 @ R1=MOD(R1,256) @ GOTO 750
660 GOSUB 1080 @ DISP TAB(11);"Medium full" @ GOTO 850
670 L2=NUM(A$[B+18,B+18])*256+NUM(A$[B+19,B+19])
680 IF NUM(A$[B+11,B+11])=0 AND L2<=B4 THEN 690 ELSE 700
690 T1=NUM(A$[B+14,B+14]) @ R1=NUM(A$[B+15,B+15]) @ GOTO 750
700 R1=NUM(A$[B+15,B+15])+L2
710 NEXT K @ NEXT R2
750 PRINT USING 940 ; B1$,T2,R2,B
751 PRINT USING 945 ; T1,R1
755 IF T$="B" THEN T3=136 @ GOTO 765
```

```
760 IF T$="L" THEN T3=137 ELSE 848
765 B1$=B1$&"  "&CHR$(224)&CHR$(T3)&""&CHR$(T1)&CHR$(R1)
770 B5=L(I)\256 @ IF MOD(L(I),256) THEN B5=B5+1
775 B1$=B1$&""&CHR$(B5)
780 D$=DATE$&TIME$ @ D$=D$[1,2]&D$[4,5]&D$[7,8]&D$[9,10]&D$[12,13]&D$[15,16]
785 FOR K=0 TO 5 @ B1$[21+K,21+K]=CHR$(BVAL(D$[2*K+1,2*K+2],16)) @ NEXT K
790 B1$[27,32]=CHR$(128)&CHR$(1)&"     " @ V$=B1$
795 B=B-1 @ GOSUB 1020
800 D$=BSTR$(TIME+86400*((IP(DATE/1000)-1)\4+IP(DATE/1000)*365+FP(DATE/1000)*100
0),16)
801 OFF ERROR @ INPUT "Access mode : ","RXELPCNT";A2$
802 ON ERROR GOTO 801
803 K2=BVAL(A2$,2) @ GOSUB 1090 @ ON ERROR GOTO 855
805 V$=CHR$(0)&CHR$(160)&CHR$(MOD(L(I),256))&CHR$(L(I)\256)&CHR$(K2)&T$
810 FOR K=1 TO 8 STEP 2 @ K1=11-(K+1)\2 @ V$[K1,K1]=CHR$(BVAL(D$[K,K+1],16)) @ N
EXT K
812 V$=V$&B1$[1,8] @ STOP
815 B2=A(I) @ T2=T1 @ R2=R1 @ B=0
820 C1=0 @ C2,C4=1 @ L2=256 @ C3=19
822 IF C1>=2 AND T$="L" AND C4 THEN GOSUB 1100 @ GOTO 830
825 V$=V$&CHR$(ROMPEEK(B2+C1,N1))
830 C1=C1+1 @ C3=C3+1
835 IF C3>L2 THEN GOSUB 1020 ELSE 822
837 BEEP @ DISP "Recd. : ";C2;"/";B4;"  R2 = ";FNH$(R2,2)
840 C2=C2+1 @ IF C2=B4 THEN L2=B3
841 IF C2>B4 THEN 500
842 C3=1 @ V$="" @ R2=R2+1
843 IF R2>255 THEN R2=0 @ T2=1
845 GOTO 822
848 DELAY 2 @ DISP TAB(12);"Type 0 prgm."
850 GOSUB 1091 @ NEXT N1
855 OFF ERROR
860 IF ERRN=252 THEN 862 ELSE 865
862 DISP TAB(7);"*** No More ROMs ***" @ END
865 DISP "Error n ";ERRN;"  Line n ";ERRL @ STOP
870 IMAGE 3x,d,5x,8a,2x,a,5x,4a,4x,4a
900 A1=ROMPEEK(B1+J,N1)+256*ROMPEEK(B1+J+1,N1)
905 IF A1 THEN A(I1)=A1 @ I1=I1+1
910 RETURN
915 DEF FNH$(Z,L)
920 Z$=BSTR$(Z,16)
925 IF LEN(Z$)<L THEN Z$="0"&Z$ @ GOTO 925
930 FNH$=Z$ @ END DEF
940 IMAGE "Programme requested : ",8a,/,"Catalog : T = ",3d,/,10x,"R = ",3d,/,10
x,"B = ",3d,/
945 IMAGE "Address : T = ",3d,/,10x,"R = ",3d,/
1020 SENDIO ":m1","lad#,ddl4",CHR$(T2)&CHR$(R2)
1030 SENDIO ":m1","lad#,ddl3",CHR$(B)
1040 SENDIO ":m1","lad#,ddl6",V$
1050 SENDIO ":m1","lad#,ddl8","" @ RETURN
1060 SENDIO ":m1","lad#,ddl4",CHR$(T2)&CHR$(R2)
1070 A$=ENTIO$(":m1","tad#,ddt2,sda") @ RETURN
1080 FOR K1=1 TO 8 @ BEEP 437,.05 @ BEEP 874,.05 @ NEXT K1 @ RETURN
1090 DISP TAB(6);"*** WORKING HARD ! ***" @ RETURN
1091 DISP TAB(9);"Next ROM now ..." @ RETURN
1100 D1=ROMPEEK(B2+C1,N1) @ D2=ROMPEEK(B2+C1+1,N1)
1110 IF D1=255 AND D2=255 THEN C4=0 @ D3=D1+256*D2 @ GOTO 1130
1120 D3=D1+256*D2-B2
1130 V$=V$&CHR$(MOD(D3,256))&CHR$(D3\256)
1140 C1=C1+1 @ C3=C3+1 @ RETURN
```

```
-ROMDIR     B 4096 09:12 07\10\84
 ROMPEEK    L  256 22:12 18\12\84
 HPILCMDS   L 1536 22:14 18\12\84
 DISPLEX    L  512 13:44 22\07\84
* OPCODE     B  10K 21:48 19\12\84
 MEMLEX     L  512 22:09 12\12\84
```

← cette version de MEMLEX contient

- PEEK (#)
- POKE #,#
- ROMPEEK (#,#)
- ADDR ($)
- DIRECT ($) — comme ADDR, mais pointant à l'adresse du début de l'entrée catalogue du programme spécifié

retournent des adresses > 32767 sans problème (en théorie !)

MEMLEX 440 Bytes:

Line	Data																Check
1	03	C9	A6	01	8D	4C	0B	F4	C9	9F	4D	45	4D	4C	45	58	20	20	C8	32	BC
2	0C	00	26	00	18	00	40	00	83	00	00	00	B6	00	CA	00	03	01	2B	01	BF
3	9B	01	00	00	9B	00	9C	00	9B	00	9B	00	9B	00	FF	FF	50	45	45	CB	4E
4	50	4F	4B	C5	41	44	44	D2	52	4F	4D	50	45	45	CB	44	49	52	45	43	4B
5	D4	FF	F9	61	64	64	72	65	73	73	20	6E	6F	74	20	69	6E	20	52	41	D5
6	CD	77	72	6F	6E	67	20	52	4F	4D	20	61	64	64	72	65	73	F3	69	6E	6D
7	76	61	6C	69	64	20	52	4F	4D	20	6E	6F	AE	65	6E	64	20	6F	66	20	1C
8	52	4F	4D	54	41	C2	FF	E1	17	9E	42	06	E3	6D	A8	B4	6E	21	A1	6D	74
9	06	E5	4C	E4	42	E5	9E	6C	06	E3	0A	E5	9E	9E	42	B1	A3	82	02	C6	4A
10	84	00	5D	A8	02	CE	64	12	5E	0A	E2	F0	E6	CE	99	4C	F9	9E	10	2D	7F
11	CE	CD	3E	CE	61	46	FA	7D	5E	93	21	A4	CE	8A	01	60	0A	E5	9E	A1	6C
12	CE	8B	3E	CE	CD	3E	70	20	A1	5E	A9	FF	7F	CE	8A	01	99	60	28	A3	4D
13	30	C5	FA	02	F0	CB	5E	A9	00	60	CE	8A	01	99	60	28	C3	30	C5	FB	4A
14	02	F0	BA	60	30	A1	CE	61	46	FA	7D	7E	21	A6	9E	18	2D	CE	77	4A	89
15	CE	92	24	5E	B1	B1	82	5C	A9	FF	7F	98	5E	1C	C5	CE	8A	01	60	28	0B
16	A3	5E	A9	FF	7F	CE	8A	01	CE	DF	3C	CE	D7	37	9E	20	2D	CE	8B	3E	D2
17	66	91	F2	0A	C9	49	00	FB	05	C9	20	00	FB	07	CE	99	4C	FB	9E	F0	36
18	CA	66	12	A2	53	A8	85	CE	CD	3E	70	20	A1	5E	A9	FF	5F	CE	8A	01	36
19	99	60	30	C5	FB	0D	5E	A9	FF	7F	CE	8A	01	99	60	30	C5	FB	07	CE	9C
20	99	4C	FA	9E	F0	D1	60	30	A1	CE	61	46	FA	7D	50	12	A0	51	13	A0	6B
21	CE	AC	47	F6	0E	5E	93	21	A4	CE	1D	48	CE	8A	01	60	0A	E5	9E	CE	CB
22	1D	48	CE	99	4C	FC	9E	18	2D	CE	77	4A	CE	92	24	5E	18	A1	F0	C8	E2
23	C879																				

DISPLEX 414 Bytes:

Line	Data																Check
1	2E	91	8C	01	BD	4C	44	F7	0C	9F	44	49	53	50	4C	45	58	20	08	30	B2
2	0C	00	4A	00	2A	00	90	00	93	00	00	00	DC	00	FA	00	03	01	08	01	89
3	0E	01	26	01	2C	01	32	01	3E	01	48	01	69	01	70	01	7A	01	84	01	FB
4	00	00	CC	00	CC	00	BF	00	BF	00	CC	00	BF	00	BF	00	BF	00	BF	00	E4
5	CD	00	BF	00	CC	00	CC	00	CC	00	FF	FF	49	4E	56	A4	45	43	A4	4D	01
6	46	44	20	4F	46	C6	4D	46	44	20	4F	CE	43	55	52	A4	44	53	50	43	D7
7	4C	D2	43	4C	52	43	55	52	D3	44	53	50	38	B0	44	53	50	34	B0	43	A0
8	4C	52	53	43	52	CE	43	4C	52	4C	49	4E	C5	4C	46	A4	43	52	A4	46	99
9	46	A4	FF	61	17	40	C8	01	F6	0F	60	A9	2E	76	69	73	69	70	74	72	BF
10	CE	6D	46	D6	1F	9E	06	E3	6D	A8	B4	00	6E	21	A1	6D	06	E5	4C	E4	87
11	42	E5	9E	6C	06	E3	0A	E5	9E	42	B1	A3	82	02	C6	A8	00	CE	9A	0C	AC
12	F0	ED	9E	42	B1	A3	82	02	C6	A8	00	CE	6D	0E	F0	DF	18	2E	5E	CE	97
13	12	1F	6E	E3	CE	25	FC	F8	10	5C	A8	80	6E	8B	FA	09	50	1E	E0	1C	6C
14	94	16	E4	F0	F3	9E	00	2E	66	A9	1B	00	44	A9	7C	FC	A1	44	A9	14	77
15	F0	A1	44	A9	0B	F0	20	2E	CE	BE	3E	6E	27	A2	A9	04	00	CE	25	FC	6D
16	F8	08	65	A8	25	64	A8	1B	16	A7	9E	A1	CE	19	08	45	9E	A1	CE	19	B7
17	08	4A	9E	A1	CE	19	08	5B	51	A8	4B	B2	CB	82	9E	A1	CE	19	08	5D	B1
18	51	A8	26	F0	F2	A1	98	42	0A	A1	D5	57	82	CD	08	00	99	F4	D1	CE	E0
19	8B	3E	60	3E	A0	CE	19	08	25	CE	1D	08	00	CE	14	08	F0	C4	A1	CE	23
20	19	08	4B	9E	00	2E	66	A9	0A	00	44	A9	7C	FC	00	2E	66	A9	0D	00	06
21	44	A9	7C	FC	00	2E	66	A9	0C	00	44	A9	7C	FC							19
22	A5F3																				

CREE ... SUITE

27 oct 84

Jean Dutertre T416
à
Mr Jean-Daniel Dodin T1

Cher Ami,

Je vous ai proposé, vers mai ou juin, je ne ne sais plus, trois programmes, CREECAS, ECRICAS, LIRECAS, pour créer directement dans la cassette un fichier à partir de la HP-75, pour écrire directement dans ce fichier, enfin pour relire des données. (Par ,petit, malheur je ne retrouve plus trace de mon texte, j'avais l'intention de le reprendre pour le développer. Reste à y ajouter le présent post scriptum).

Naturellement (non, ce n'est pas si naturel que ça) LIRECAS peut lire, directement sur la cassette, un fichier écrit d'abord en mémoire centrale, puis COPYé sur la cassette, selon la procédure habituelle. Donc, un fichier "normal".

J'ai utilisé longtemps ces programmes tels quels, mais j'étais agaçé par le fait que l'on ne peut pas COPYer les fichiers écrits par ECRICAS de la cassette vers la mémoire centrale: la cassette est lue sans arrêt jusqu'à ce qu'un message annonce "not enough memory". Il est bien entendu que cela n'empêche pas de les utiliser et que c'est même le cas général des fichiers sur disque. Cependant j'ai essayé de reprendre la question pour donner à mes fichiers créés, puis écrits en mémoire de masse, la "vraie" structure attendue par la HP-75. A mesure que progressait le déchiffrage les choses s'arrangeaient. On pouvait les reCOPYer en mémoire centrale, les EDITer, les relire avec les flèches qui montent et descendent les lignes dans l'affichage LCD. Ce sont des lignes numérotées de DATA tout à fait correctes. Mais... dès que l'on voulait PURGEr ces fichiers, ou en COPYer un autre, ou en EDITer un autre, c'etait le crash immédiat.

Actuellement il ne s'en faut plus que d'un octet... Mais peut-être peut-on déjà diffuser ces programmes, à charge pour quelqu'un d'inventif de dénicher le dernier? J'ai trouvé la structure des lignes dans le listage de la ROM interne de la HP-75. Pour l'en-tête, il faut deviner... **Dès maintenant tout va bien.** Mes fichiers peuvent être créés sans danger sur une cassette en cours d'emploi, sans altérer le répertoire, sans écraser rien. Ils peuvent aussi être écrits, toujours sans danger, et bien sûr relus et exploités. On peut les reCOPYer en mémoire centrale, les EDITer, les relire comme j'ai dit et tout se passe bien, il n'y a plus de crash. Evidemment, pour les COPYer en mémoire centrale, il y faut une place suffisante et tout l'intérêt de l'écriture directe en mémoire de masse vient de la possibilité de créer de GRANDS fichiers.

Voici donc ce que l'on sait déjà de la structure des fichiers de la cassette:

Les deux premiers enregistrements ont déjà été décrits par PPC-T. La structure du répertoire a été décrite aussi. Ce qui suit ne concerne donc que les fichiers eux mêmes, et encore, les fichiers BASIC en lignes numérotées, avec sur chaque ligne un DATA suivi d'une chaîne de caractères entre simples guillemets. Ce fichier peut être utilisé pour des données chiffrées de tout type, entier, flottant, long, puisqu'il suffit des faire des STR$ avant de les écrire et d'en refaire des VAL après les avoir lues. Est-ce clair?

CREECAS2. Création d'un fichier HP75 normal, directement en mémoire de masse (cassette). La cassette est initialisée, neuve ou en cours d'emploi. Donner un nom de fichier et le nombre de registres nécéssaires pour contenir les données. Seul le repertoire est mis à jour.

ECRICAS4. Ecriture des données directement dans les registres de la cassette réservés par creecas2. Demande seulement le nom du fichier. Produit un fichier complet de son en-tête et de son message EOF. Peut être copié en mémoire centrale, relu, purgé sans incident (s'il y a de la place, mais c'est inutile).

LIRECAS2. Relecture des données enregistrées en mémoire de masse par ecricas4. Demande seulement le nom du fichier à lire. C'est en fait une subroutine à incorporer dans un programme d'exploitation de données (exemple fourni).

DUMPCAS3. Programme de "dump" pour cassette HP75. Cette version est adaptée à l'interface video et à l'imprimante thermique (ou autre). La fenêtre LCD redéfinit les touches situées au dessous (de APPT à FET). Adresses des registres en hexadécimal. Registrs de 0 à n. Octets de 1 à 256.

Cela paraît idiot, je le crains, mais c'est un bon moyen de n'utiliser qu'un seul type de fichier. Pour le décrire, je me démarque un peu de l'habitude informatique consistant à parler de "3 objets numérotés respectivement 0, 1 et 2". Ceci parce que les chaînes sont traitées avec les instructions Basic du type A1$[1,5] dans lesquelles 1 désigne le premier caractère. Il vaut mieux que le premier octet porte le numéro 1 plutôt que le numéro zéro si l'on veut s'y retrouver. En revanche, les enregistrements commenceront toujours par zéro à cause de notre habitude prise de trouver le répertoire en numéro 2. De plus je crois que dans toutes les descriptions d'enregistrements, il est bon de désigner leur adresse en hex, c'est beaucoup plus facile qu'en décimal, car le (ou les) programmes de DUMP donnent directement l'adresse hex. Piste 0 et enregistrement 12hex, c'est l'enregistrement 12. Piste 1 et enregistrement A3hex, c'est l'enregistrement 1A3. Un seul paramètre, no problem, sir. Le fichier se présente ainsi:

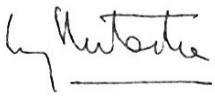
OCTETS	CONTENU
1	? pas de structure lisible (bombe à retardement?)
2	un numéro d'ordre de création. (Sans importance?)
3	LSB en binaire — Ces octets représentent la taille
4	MSB — du fichier en octets moins 18.
5	octet d'accès — (128 pour R.... 1 pour T). Ici c'est RXELPCNt, donc FE hex.
6	type de fichier — Ici c'est 42 hex (B, pour Basic)
7	LLSB — Date en secondes depuis le
8	LSB en binaire — début du siècle. Il fallait le
9	MSB — faire. La date prend 4 octets ici
10	MMSB — au lieu de 6 dans le répertoire.
11 à 18	le nom du fichier en ASCII, complété au besoin par des blancs, 20 hex. On peut remarquer qu'ici commence le décompte des octets du fichier (au delà de 18).
19 à 28	des nuls 00 hex. Il n'y a rien dans cette zone.
29	LSB en décimal codé hexadécimal — C'est le numéro
30	MSB " " — de ligne basic.
31	longueur de ligne, compris les deux suivants et EOL.
32	code symbolique (token) de DATA, soit 86 hex.
33	code symbolique du guillemet simple, soit 96 hex.
34	longueur de la chaîne de caractères (nue...).
35 et suivants,	la chaîne de caractères, terminée par le caractère EOL qui est 0E hex.
...	Les lignes suivantes, sur le modèle 29 à 35 etc.
...	"
...	"
-	99 hex — Ces 5 octets constituent le message EOF,
-	A9 hex — fin de fichier. Il y a successivement
-	02 hex — la dernière ligne A999, la longueur du
-	8A hex — reste (2 octets), le code symbolique
-	0E hex — basic END (8A) et le EOL (0E)...

Et voilà, c'est tout. J'offre personnellement un caramel mou et toute mon estime à qui décrira l'abominable premier octet manquant. (Cette offre est valable jusqu'au 30 novembre à midi, le cachet de la poste etc.). Ceci pour la beauté de la chose, puisque son absence ne nuit pas.

Les trois programmes, en leur état actuel, sous le nom de CREECAS, ECRICAS4, LIRECAS2, suivent sur cassette par courrier séparé, en même temps que DUMPCAS3, dernière version améliorée de l'indispensable DUMP. DUMPCAS3 utilise l'affichage LCD pour indiquer les nouvelles fonctions des touches placées au dessous. L'une d'elles provoque l'impression sur papier de l'écran video. Ces programmes ont été "traduits" maintenant pour le module I/O et font appel aux instructions SENDIO, ENTIO$, DTH$, HTD, RTRIM etc. Par ailleurs, j'ai appris par Eric Gengoux l'existence d'un lex MASSLEX qui positionne la tête de lecture plus facilement qu'avec des subroutines nourries d'ENTIO$ et de SENDIO. Mais

malgré cette simplification, le problème du fichier normal subsisterait... et la course au caramel mou reste ouverte. J'y ajoute encore "ENPAGE62", dernière version du programme ENPAGE pour la petite thermique 162. Il n'y a plus le problème rencontré par Damien Debril et D. Delefortrie, il suffisait d'ajouter PWIDTH INF n'importe où au début. Ce nouveau programme a l'interêt de faire imprimer des lettres accentuées. Le couvre clavier requis est celui qui a déjà été reproduit dans PPC-T en même temps que le premier "enpage", le "keys" est l'AZERTAC décrit aussi. Les lettres accentuées y figurent, sauf pour les accents circonflexes et les trémas que l'on tape, comme sur une machine à écrire française, avant la lettre à accentuer. On peut coller les bandes les unes à coté des autres en colonnes de journal. Sur la lancée, ci-joint enfin, même punition même motif, un "ENPAGE83" accentué pour la PIL83 graphique de la SICAPE.

Heureuse Programmation,



Sauf erreur de ma part (je n'ai pas fait l'essai moi-même), le premier octet d'un fichier **en RAM** et sans doute en ROM indique, si il est à zéro, un fichier "système" qui doit rester invisible au catalogue du 75. Je viens également d'apprendre par Janick que les fichiers à écriture directe sur cassette utilisés par MASSLEX (et qui ressemblent à ceux ci-dessus) sont de type "G". Ce type est un type officiel HP qui a failli (!!!!) être introduit dans le ROM I/O. Ce ROM reconnait ce type de fichier au niveau du COPY, ce qui permet de copier ces fichiers en RAM, mais ne les reconnait pas au niveau de l'exécution, ce qui provoque les plantages.

Les fichiers à en tête zéro peuvent être copiés sur cassette, édité,... mais pas catalogués (sauf par CAT$(0)).

```
340 DISP USING 110 ; DTH$(A),B10 !     DUMPCAS3
20 !     M.Martinet
30 !     ppct #8p22
40 !     modif J-D.Dodin
50 !     modif J.Dutertre
60 !     02 avr 84
70 !     pour video 163
80 !     et imprim. 162
90 !
100 DISPLAY IS ":D1" @ DELAY 0 @ WIDTH INF @ PWIDTH INF
110 IMAGE 'Enreg',x,3a,x,'Octets',x,3z,x,3z,x,3z,x,3z
120 IMAGE 'Enr-',3a,x,'Oct-',3z
130 IMAGE 23a,x,8a
140 DIM E$[256],R$[256],D$[2],D0$[2],D1$[64],D2$[256]
150 INTEGER A,B,B1,C,D,E
160 A,B,F1=0 @ E$,R$="" @ GOSUB 950
170 DISP CHR$(27)&"E" @ GOTO 770
180 GOSUB 880
190 SENDIO ":M1","LAD#,DDL3",CHR$(0)
200 E$=ENTIO$(":M1","TAD#,DDT0,SDA") @ R$=E$
210 GOSUB 960 @ DISP "patientez...";
220 FOR I=0 TO 63
230 C=NUM(E$[B+I+1,B+I+1])
240 IF C>31 AND C<128 OR C>160 THEN D1$[I+1]=CHR$(C) @ GOTO 290
250 IF C=8 THEN D1$[I+1]="B" @ GOTO 290
260 IF C=10 THEN D1$[I+1]="L" @ GOTO 290
270 IF C=13 THEN D1$[I+1]="C" @ GOTO 290
280 IF C=27 THEN D1$[I+1]="E" ELSE D1$[I+1]="."
290 D0$=DTH$(C) @ IF LEN(D0$)<2 THEN D0$="0"&D0$
300 D2$[3*I+1,3*I+2]=D0$
310 NEXT I
320 DISP CHR$(27)&"E";
330 FOR I=0 TO 1
340 DISP USING 110 ; DTH$(A),B+32*I+1,B+32*I+9,B+32*I+17,B+32*I+25 @ DISP
```

```
350 FOR J=0 TO 3
360 DISP USING 130 ; D2$[96*I+24*J+1,96*I+24*J+23],D1$[32*I+8*J+1,32*I+8*J+8]
370 NEXT J @ DISP @ NEXT I
380 B1=B @ B=B+64 @ IF B)=255 THEN B=0
390 GOSUB 960 @ DISP "visu ed  r- r+ e- e+ imp reg";
400 D$=KEY$ @ IF D$="" THEN 400
410 IF D$=CHR$(132) THEN B=0 @ GOSUB 960 @ GOTO 650
420 IF D$=CHR$(133) THEN B=0 @ GOSUB 960 @ GOTO 600
430 IF D$=CHR$(134) THEN B=MAX(B-128,0) @ GOTO 210
440 IF D$=CHR$(135) THEN 210
450 IF D$=CHR$(131) THEN 700
460 IF D$=CHR$(137) THEN 750
470 IF D$=CHR$(130) THEN B=MAX(B-64,0) @ GOTO 210
480 IF D$=CHR$(136) THEN GOSUB 810
490 IF D$=CHR$(160) THEN GOSUB 880 @ GOSUB 960 @ DISP "termine" ELSE 390
500 GOTO 390
510 SENDIO ":M1","LAD#,DDL7","" @ END
520 GOSUB 960 @ DISP USING 120 ; DTH$(A),B1+1;
530 B2=MOD(B1,64)
540 DISP " Hex=(";D2$[B2*3+1,B2*3+2];")=? "; @ INPUT "";D$
550 IF D$="" THEN 390
560 C=HTD(D$)
570 E$[B1+1,B1+1]=CHR$(C)
580 B1=B1+1 @ GOTO 520
590 GOTO 250
600 B=0 @ GOSUB 880
610 IF A=511 THEN A=0 ELSE A=A+1
620 GOSUB 950
630 GOSUB 880
640 GOTO 200
650 GOSUB 880
660 IF A=0 THEN A=511 ELSE A=A-1
670 GOSUB 950
680 GOSUB 880
690 GOTO 200
700 GOSUB 960 @ DISP "modifier octet numero? ";
710 INPUT ""; B1$ @ IF B1$="" THEN 700
720 ON ERROR GOTO 700
730 B1=VAL(B1$)-1 @ OFF ERROR
740 GOTO 520
750 B=0 @ GOSUB 880
760 GOSUB 950
770 GOSUB 960 @ DISP "aller a enreg. numero(hex)? "; @ INPUT "";A1$
780 ON ERROR GOTO 770
790 A=HTD(A1$) @ OFF ERROR
800 GOTO 180
810 FOR I=0 TO 1
820 GOSUB 960 @ DISP "patientez...";
830 PRINT USING 110 ; DTH$(A),B1+32*I+1,B1+32*I+9,B1+32*I+17,B1+32*I+25 @ PRINT
840 FOR J=0 TO 3
850 PRINT USING 130 ; D2$[96*I+24*J+1,96*I+24*J+23],D1$[32*I+8*J+1,32*I+8*J+8]
860 NEXT J @ PRINT @ NEXT I
870 RETURN
880 IF E$=R$ THEN 940
890 SENDIO ":M1","LAD#,DDL4",CHR$(MOD(A DIV 256,2))&CHR$(MOD(A,256))
900 SENDIO ":M1","LAD#,DDL3",CHR$(0)
910 SENDIO ":M1","LAD#,DDL6",""
920 IF R$=CHR$(14) AND E$="!" THEN 940
930 SENDIO ":M1","LAD#,DDL6",E$
940 RETURN
950 E$="!" @ R$=CHR$(14) @ RETURN
960 DISP CHR$(27)&"%"&CHR$(0)&CHR$(13);CHR$(27)&"J" @ RETURN
970 ! upd.#6 841023
10 ! *  CREECAS2    *
20 ! *  J.Dutertre  *
30 ! * 08 avril 84  *
40 !
50 ! The program creates
60 ! an (empty) file
70 ! on a mass media.
80 !
90 DIM A9$[32],A$[256]
```

```
100 F0,T,T0,T9=0
110 K$="media full."
120 M$=":M1"
130 INPUT "file name? ";F$
140 INPUT "# of regs. needed? ";N9
150 GOTO 260
160 D4=NUM(D5$[1,1]) @ E4=NUM(D5$[2,2])
170 IF D4<58 THEN C4=VAL(CHR$(D4))*16 ELSE C4=(D4-55)*16
180 IF E4<58 THEN C4=C4+VAL(CHR$(E4)) ELSE C4=C4+E4-55
190 RETURN
200 SENDIO M$,"LAD#,DDL4",CHR$(MOD(A DIV 256,2))&CHR$(MOD(A,256))
210 SENDIO M$,"LAD#,DDL3",CHR$(0)
220 SENDIO M$,"LAD#,DDL6",""
230 IF T9=1 THEN T9=0 @ GOTO 250
240 SENDIO M$,"LAD#,DDL6",A$
250 RETURN
260 ! looking for free space
270 A=0 @ T9=1 @ GOSUB 200
280 A$=ENTIO$(M$,'TAD#,DDT2,SDA')
290 N=NUM(A$[20,20]) @ E=N+2
300 IF N9>510-N THEN DISP K$ @ BEEP @ GOTO 650
310 A=2 @ T9=1 @ GOSUB 200
320 A$=ENTIO$(M$,'UNL,TAD#,DDT2,SDA')
330 FOR I=1 TO N
340 A=I+1
350 FOR J=0 TO 7
360 J1=J*32 @ T$=A$[J1+11,J1+12]
370 IF T$=CHR$(255)&CHR$(255) THEN 490
380 IF T$=CHR$(0)&CHR$(0) THEN T9=1
390 F0=F0+1 @ IF F0=N*8-1 THEN DISP K$ @ GOTO 650
400 E=NUM(A$[J1+15,J1+15])*256+NUM(A$[J1+16,J1+16])
410 T=NUM(A$[J1+19,J1+19])*256+NUM(A$[J1+20,J1+20]) @ T0=T0+T
420 IF T9=1 THEN T9=0 @ GOTO 440
430 GOTO 450
440 IF T>=N9 THEN N9=T @ GOTO 500
450 NEXT J
460 SENDIO M$,'LAD#,DDL3',CHR$(0)
470 A$=ENTIO$(M$,'UNL,TAD#,DDT0,SDA')
480 NEXT I
490 E=E+T
500 FOR K=1 TO 32 @ A9$[K,K]=CHR$(0) @ NEXT K
510 F$=UPRC$(F$) @ F$=FILL$("",F$," ",8) @ A9$[1,8]=F$
520 D$=DATE$
530 D5$=D$[1,2] @ GOSUB 160 @ A9$[21,21]=CHR$(C4)
540 D5$=D$[4,5] @ GOSUB 160 @ A9$[22,22]=CHR$(C4)
550 D5$=D$[7,8] @ GOSUB 160 @ A9$[23,23]=CHR$(C4)
560 D$=TIME$
570 D5$=D$[1,2] @ GOSUB 160 @ A9$[24,24]=CHR$(C4)
580 D5$=D$[4,5] @ GOSUB 160 @ A9$[25,25]=CHR$(C4)
590 A9$[19,20]=CHR$(MOD(N9 DIV 256,2))&CHR$(MOD(N9,256))
600 A9$[15,16]=CHR$(MOD(E DIV 256,2))&CHR$(MOD(E,256))
610 A9$[9,12]=CHR$(32)&CHR$(32)&CHR$(224)&CHR$(136)
620 A$[J1+1,J1+32]=A9$
630 GOSUB 200
640 DISP F$;N9;"regs. Addr.(hex) ";DTH$(E)
650 END
660 ! updated 841017
10 !   ECRICAS4
20 !   J.Dutertre
30 !   12 avril 84
40 !
50 ! This program writes
60 ! data on mass memory
70 ! in a file created
80 ! with "creecas2".
90 !
100 INTEGER E8,E9,F,I,J,K9,L,L8,L9,N,R9,T8,T9
110 DIM A9$[256],C7$[1],C9$[2],F1$[40],L8$[2],M$[3],D(6),M(12)
120 ! initialize
130 C7$=CHR$(14) @ C9$=CHR$(134)&CHR$(150)
140 F$="" @ L8$="  " @ M$=":M1" @ L8,T9=0
150 DATA 0,31,59,90,120,151,181,212,243,273,304,334
```

```
160 GOTO 720
170 ! Look for reg. # E9
180 SENDIO M$,"LAD#,DDL4",CHR$(MOD(E9 DIV 256,2))&CHR$(MOD(E9,256))
190 SENDIO M$,"LAD#,DDL3",CHR$(0)
200 IF T9=1 THEN T9=0 @ GOTO 230
210 SENDIO M$,"LAD#,DDL6",A9$
220 SENDIO M$,"LAD#,DDL8",""
230 RETURN
240 ! file header
250 FOR I=1 TO 6 @ D(I)=VAL(DTH$(NUM(D$[I,I]))) @ NEXT I
260 FOR I=1 TO 12 @ READ M(I) @ NEXT I
270 D0=D(1)*365+D(1) DIV 4+M(D(2))+D(3)-1
280 D0=D0*86400+D(4)*3600+D(5)*60+D(6)
290 D1=D0 DIV 256^3 @ D2=MOD(D0,256^3) DIV 256^2
300 D3=MOD(D0,256^2) DIV 256 @ D4=MOD(D0,256)
310 T9=1 @ GOSUB 180
320 A9$="" @ F$=FILL$("",F$," ",8)
330 K9=1
340 F1$=RPT$(" ",2)&L8$&CHR$(254)&"B"&CHR$(D4)&CHR$(D3)&CHR$(D2)&CHR$(D1)&F$&RPT
$(CHR$(0),22)
350 F1$=F1$[1,28]
360 A9$=F1$
370 RETURN
380 ! look for file F$
390 E9=0 @ T9=1 @ GOSUB 180
400 A9$=ENTIO$(M$,'TAD#,DDT2,SDA')
410 N=NUM(A9$[20,20])
420 E9=2 @ T9=1 @ GOSUB 180
430 A9$=ENTIO$(M$,'UNL,TAD#,DDT2,SDA')
440 FOR I=1 TO N
450 FOR J=0 TO 7
460 J1=J*32 @ T$=A9$[J1+11,J1+12]
470 IF T$=CHR$(255)&CHR$(255) THEN DISP "no file ";F$ @ GOTO 1090
480 IF T$=CHR$(0)&CHR$(0) THEN 540
490 P$=A9$[J1+1,J1+8] @ D$=A9$[J1+21,J1+26]
500 P$=RTRIM$(" ",P$)
510 E=NUM(A9$[J1+15,J1+15])*256+NUM(A9$[J1+16,J1+16])
520 T8=NUM(A9$[J1+19,J1+19])*256+NUM(A9$[J1+20,J1+20])
530 IF P$=F$ THEN E8,E9=E @ T8=E+T8 @ GOTO 580
540 NEXT J
550 T9=1 @ GOSUB 190
560 A9$=ENTIO$(M$,'UNL,TAD#,DDT0,SDA')
570 NEXT I
580 RETURN
590 ! formatting data
600 L9=LEN(N9$) @ N8$=CHR$(HTD(STR$(MOD(K9,100))))&CHR$(HTD(STR$(INT(K9/100))))
610 N7$=CHR$(L9+4) @ L8=L8+L9+7 @ N9$=N8$&N7$&C9$&CHR$(L9)&N9$&C7$ @ L9=L9+7
620 R9=256-LEN(A9$) @ IF R9)=L9 THEN 640
630 A9$=A9$&N9$[1,R9] @ N9$=N9$[R9+1] @ GOSUB 670
640 A9$=A9$&N9$
650 K9=K9+1
660 RETURN
670 GOSUB 180
680 E9=E9+1 @ A9$=""
690 IF E9)T8 THEN BEEP @ DISP "error, file full!" @ GOTO 1090
700 RETURN
710 ! main program
720 INPUT "file name? ";F$
730 F$=UPRC$(F$[1,MIN(LEN(F$),8)])
740 DISP "looking for file ";F$ @ GOSUB 390 @ GOSUB 250
750 INPUT "lab's name? ";L$
760 IF L$="" THEN L$="xxx"
770 L=1 @ F=0
780 INPUT "# of data per point? ";N
790 N9$=F$ @ GOSUB 600
800 N9$=DATE$ @ GOSUB 600
810 N9$=L$ @ GOSUB 600
820 N9$=STR$(L) @ GOSUB 600
830 N9$=STR$(F) @ GOSUB 600
840 N9$=STR$(N) @ GOSUB 600
850 N9$="tzyx" @ GOSUB 600
860 FOR I=1 TO 1000
```

```
870 FOR J=1 TO N
880 I$=STR$(I) @ J$=STR$(J)
890 DISP "data # (";I$;",";J$;")= "; @ INPUT N9$
900 IF UPRC$(N9$)="FIN" THEN 980
910 P9=POS(N9$,";") @ IF P9#0 THEN N9$[P9,P9]="."
920 ON ERROR GOTO 890
930 N9$=STR$(VAL(N9$))
940 OFF ERROR
950 GOSUB 600
960 NEXT J
970 NEXT I
980 IF J#1 THEN DISP "no, go on!" @ GOTO 890
990 FOR J=1 TO N
1000 N9$=STR$(1.E35) @ GOSUB 600
1010 NEXT J
1020 ! end of file line
1030 N9$=CHR$(153)&CHR$(169)&CHR$(2)&CHR$(138)&C7$ @ L8=L8+15 @ L9=5 @ GOSUB 620
1040 GOSUB 180
1050 L8$=CHR$(MOD(L8,256))&CHR$(MOD(L8 DIV 256,2))
1060 E9=E8 @ T9=1 @ GOSUB 180 @ A9$=ENTIO$(M$,'TAD#,DDT2,SDA') @ A9$[3,4]=L8$ @
T9=0
1070 GOSUB 180
1080 DISP "done!"
1090 END
1100 ! upd. 841026
10 ! This is your own
20 ! main program
30 !
40 INTEGER L1,F1,N1,K9,L9,M9
50 INPUT "file name? ";F$
60 DISP "looking for file ";F$
70 GOSUB 340
80 IF A9=2 THEN BEEP @ GOTO 50
90 ! reading header.
100 GOSUB 580
110 DISP "file: ";F1$;" on ";D1$
120 DISP "lab.: ";L1$;".";N1;"data per point"
130 ! reading data
140 DISP "press any key to get next data"
150 GOSUB 810
160 IF N9=1.E35 THEN 200
170 DISP "data= ";N9
180 R$=KEY$ @ IF R$="" THEN 180
190 GOTO 150
200 DISP "done!" @ END
210 !   LIRECAS1
220 !   J.Dutertre
230 !   12 avril 84
240 !
250 ! This subprogram
260 ! reads a file in
270 ! mass memory.
280 !
290 ! With a file name
300 ! look for the
310 ! address A9 of
320 ! the first register
330 ! (INTEGER K9,L9,M9)
340 F$=UPRC$(F$[1,MIN(LEN(F$),8)])
350 U$=CHR$(14) @ M$=":M1" @ K9=0
360 DIM E9$[300],E8$[50],E7$[20]
370 A9=0 @ GOSUB 730
380 N=NUM(E9$[20,20])
390 A9=2 @ GOSUB 730
400 FOR I=1 TO N
410 FOR J=0 TO 7
420 J1=J*32 @ T$=E9$[J1+11,J1+12]
430 IF T$=CHR$(255)&CHR$(255) THEN BEEP @ DISP "no file ";F$ @ GOTO 530
440 IF T$=CHR$(0)&CHR$(0) THEN 500
450 P$=E9$[J1+1,J1+8]
460 P$=RTRIM$(" ",P$)
470 E=NUM(E9$[J1+15,J1+15])*256+NUM(E9$[J1+16,J1+16])
480 IF P$=F$ THEN A9=E @ GOTO 530
```

```
490 T8=NUM(E9$[J1+19,J1+19])*256+NUM(E9$[J1+20,J1+20])
500 NEXT J
510 GOSUB 730
520 NEXT I
530 RETURN
540 ! with address A9
550 ! open the file and
560 ! read F1$,D1$,L1$,
570 ! L1,F1,N1,X1$.
580 GOSUB 730
590 K9=28 @ L9=NUM(E9$[K9+6,K9+6])
600 F1$=E9$[K9+7,K9+L9+6]
610 FOR I9=2 TO 7
620 K9=POS(E9$[K9+1],U$)+K9 @ L9=NUM(E9$[K9+6,K9+6])
630 E7$=E9$[K9+7,K9+L9+6]
640 IF I9=2 THEN D1$=E7$
650 IF I9=3 THEN L1$=E7$
660 IF I9=4 THEN F1=VAL(E7$)
670 IF I9=5 THEN L1=VAL(E7$)
680 IF I9=6 THEN N1=VAL(E7$)
690 IF I9=7 THEN X1$=E7$
700 NEXT I9
710 RETURN
720 ! read reg. # A9
730 SENDIO M$,"LAD#,DDL4",CHR$(MOD(A9 DIV 256,2))&CHR$(MOD(A9,256))
740 SENDIO M$,"LAD#,DDL3",CHR$(0)
750 SENDIO M$,"LAD#,DDL6",""
760 SENDIO M$,"LAD#,DDL3",CHR$(0)
770 E9$=ENTIO$(M$,"TAD#,DDTO,SDA")
780 RETURN
790 ! read next data
800 ! as var. N9
810 IF LEN(E9$)-K9(32 THEN E8$=E9$[K9] @ A9=A9+1 @ GOSUB 730 @ E9$=E8$&E9$ @ K9=
1
820 K9=POS(E9$[K9+1],U$)+K9
830 L9=NUM(E9$[K9+6,K9+6]) @ N9=VAL(E9$[K9+7,K9+L9+6])
840 RETURN
850 ! updated 841026
10 !      ENPAGE62
20 !      J.Dutertre
30 !      25.04.84
40 !
50 DIM A$[192],B$[192],B1$[96],B2$[96],T$[96]
60 INTEGER H,I,J,K,L,M,N,N1,O,P,Q,R,S
70 T$="" @ G$=CHR$(27)&"*b7G" @ K,M,R,S=0 @ N1,O,P=1 @ H=60 @ L=24
80 PWIDTH INF @ DISP "patience" @ GOSUB 830
90 INPUT "fichier a imprimer? ";F$
100 ASSIGN # 1 TO F$
110 ON ERROR GOTO 160
120 FOR I=1 TO 10
130 READ # 1 ; A$
140 IF A$[1,1]#" " THEN S=1 @ GOTO 160
150 NEXT I
160 RESTORE # 1
170 ON ERROR GOTO 420
180 READ # 1 ; A$ @ IF S=1 THEN 200
190 IF LEN(A$)>1 THEN A$=A$[2]
200 IF A$[1,1]#" " THEN R=0 @ GOTO 230
210 IF A$[1,1]#" " OR T$="" THEN 230
220 B$=T$ @ T$="" @ GOSUB 690 @ PRINT B$ @ M=M+1 @ R=R+1 @ GOSUB 590 @ GOTO 240
230 IF T$#"" THEN A$=T$&" "&A$
240 J=L @ IF J>=LEN(A$) THEN T$=A$ @ A$="" @ GOTO 180
250 IF A$[J+1,J+1]#" " THEN J=J-1 ELSE 270
260 IF J>0 THEN 250 ELSE B$=A$[1,L-1]&"-" @ A$=A$[L] @ GOTO 280
270 B$=A$[1,J] @ A$=A$[J+2]
280 J=INT(L/2)+1
290 IF LEN(B$)>J THEN 340
300 IF LEN(B$)=J THEN 320
310 IF POS(B$," ")=0 THEN B$=B$&" "
320 IF LEN(B$)<J THEN GOSUB 450 @ GOTO 320
330 GOSUB 550 @ GOTO 380
340 J=L
```

```
350 IF POS(B$," ")=0 THEN B$=B$&" "
360 GOSUB 640
370 IF LEN(B$)<J THEN GOSUB 450 @ GOTO 370
380 GOSUB 690
390 O=O*(-1) @ PRINT B$ @ M=M+1
400 IF M>=H THEN GOSUB 600
410 GOTO 240
420 ASSIGN # 1 TO * @ B$=T$ @ GOSUB 690 @ PRINT B$
430 FOR I=1 TO 6 @ PRINT @ NEXT I
440 GOTO 950
450 N=3 @ Q=LEN(B$)
460 IF O<0 THEN N=LEN(B$) @ Q=3
470 FOR I=N TO Q STEP O
480 IF B$[I,I]#" " THEN K=0 @ GOTO 530
490 IF K=1 THEN 530
500 B1$=B$[1,I] @ B2$=B$[I+1]
510 B$=B1$&" "&B2$ @ K=1 @ I=I+1
520 IF LEN(B$)=J THEN 540
530 NEXT I
540 RETURN
550 FOR I=1 TO L-2 STEP 2
560 B$=B$[1,I]&" "&B$[I+1]
570 NEXT I
580 RETURN
590 IF M<H-5 OR R>1 THEN 610
600 PRINT CHR$(12) @ PRINT @ M=3 @ P=P+1 @ PRINT P @ PRINT
610 RETURN
620 K=POS(B$[I+1],R$) @ IF K#0 THEN I=I+K @ J=J+1 @ GOTO 620
630 RETURN
640 R$="^" @ I=0 @ GOSUB 620 @ R$="~" @ I=0 @ GOSUB 620
650 RETURN
660 K=POS(B$,R$) @ IF K=0 THEN 680
670 B$=B$[1,K-1]&V$&B$[K+N1] @ GOTO 660
680 RETURN
690 R$="@" @ V$=G$&A1$ @ N1=1 @ GOSUB 660
700 R$="^a" @ V$=G$&A2$ @ N1=2 @ GOSUB 660
710 R$="{" @ V$=G$&E1$ @ N1=1 @ GOSUB 660
720 R$="}" @ V$=G$&E2$ @ N1=1 @ GOSUB 660
730 R$="^e" @ V$=G$&E3$ @ N1=2 @ GOSUB 660
740 R$="~e" @ V$=G$&E4$ @ N1=2 @ GOSUB 660
750 R$="^i" @ V$=G$&I1$ @ N1=2 @ GOSUB 660
760 R$="~i" @ V$=G$&I2$ @ N1=2 @ GOSUB 660
770 R$="^o" @ V$=G$&O1$ @ N1=2 @ GOSUB 660
780 R$="|" @ V$=G$&U1$ @ N1=1 @ GOSUB 660
790 R$="^u" @ V$=G$&U2$ @ N1=2 @ GOSUB 660
800 R$="~u" @ V$=G$&U3$ @ N1=2 @ GOSUB 660
810 R$="\" @ V$=G$&C1$ @ N1=1 @ GOSUB 660
820 RETURN
830 FOR I=1 TO 91
840 READ K @ T$=T$&CHR$(K)
850 NEXT I
860 A1$=T$[1,7] @ A2$=T$[8,14]
870 E1$=T$[15,21] @ E2$=T$[22,28] @ E3$=T$[29,35] @ E4$=T$[36,42] @ I1$=T$[43,49
]
880 I1$=T$[43,49] @ I2$=T$[50,56] @ O1$=T$[57,63]
890 U1$=T$[64,70] @ U2$=T$[71,77] @ U3$=T$[78,84] @ C1$=T$[85,91]
900 T$="" @ RETURN
910 DATA 0,32,84,85,86,120,0,0,32,86,85,86,120,0,0,56,84,86,85,8,0,0,56,84,85,86
,8,0
920 DATA 0,56,86,85,86,8,0,0,56,85,84,85,8,0,0,0,74,121,66,0,0,0,0,74,120,66,0,0
930 DATA 0,56,70,69,70,56,0,0,56,65,66,32,120,0,0,56,66,65,34,120,0,0,56,66,64,5
6,120,0
940 DATA 0,24,36,100,36,16,0
950 DISP "termine!"
960 END ! 25-04-84
```

ENPAGE62. Version d'enpage 75 adaptée à l'imprimante thermique 162 qui devient accentuée. Ce texte est imprimé de cette façon. Devrait fonctionner avec l'imprimante matricielle à 80 colonnes. Modifier la ligne 70 (L=80).

T416 le 841106.

REACTIONS

La rubrique "Réactions" avait disparu depuis quelques mois, en grande partie du fait du grand retard pris dans la publication des articles. C'est pour la même raison que vous ne trouvez guère de codes barre. Ce retard va disparaitre du fait de l'abandon des "dossiers", revoilà donc les Réactions. N'étant pas publiés vous avez peu écrit, les lettres ci-jointes sont donc anciennes, un petit coup de rétro ne fait pas de mal en début d'année...

Ecrivez donc. Mais n'attendez pas trop que je vous réponde, le temps manque, hélas. Donnez votre avis, sollicitez celui des autres, faites ce que vous voulez, mais n'oubliez pas d'écrire lisiblement (ou mieux à la machine, sur papier à lettre au format commercial standard en NOIR et RECTO SEUL. Merci.

St Herblain le 30 oct. 83

M CHOSSON Yves
10 Rue St Exupery
44 800 St Herblain

Jean Daniel Dodin
77 Rue du Cagire
31100 Toulouse

Monsieur,

je vous ai commandé et lu votre publication "Au fond de la 41C". Je tiens à vous faire part que je suis très surpris de son contenu :

Vous consacrez la moitié de votre livre au micro-code auquel ne peuvent accéder la très grande majorité de détenteurs de HP41, faute d'avoir l'un des deux appareil, (exemple lecteur d'Eprom). Il m'apparait de toute evidence que ce texte microcode n'a rien à faire dans une publication séparé de l'achat d'un lecteur d'Eprom

Quant à la premiere moitié ; Elle est tout aussi décevante : Vous écrivez que le module 3 (voir II.5) pourrait servir à stocker des programmes ! Or vous ne démontrez même pas que ceci est possible. Peut on vous croire ? (à moins qu'il faille acheter un lecteur d'Eprom) Vous nous expliquez qu'il existe des registres (R.N. O.P. etc) Oh surprise ! là aussi vous ne démontrez par leur utili

té et pas le moindre mot sur leurs conditions d'utilisation. Alors pourquoi tant d'engouement et d'éloges sur la programmation synthetique ?

En conséquence tout le reste (B.G., programme KA et LB) deviennent eux aussi inutiles. A quoi cela servirait-il d'assigner ou de programmer des fantaisies. Est-ce fait pour épater le copain ?

J'aurais mieux compris si vous aviez expliqué par exemple que l'on pourrait faire (sous module complementaire bien sûr) :

Alpha L O G Alpha puis X E Q M et qu'effectivement la fonction LOG_{10} s'effectue sur le registre x. ou encore qu'on puisse programmer SizE xxx ou Size I xxx, ou GT. xxx ou GT. Ind xxx

Ceci vous aurait pris moins d'une page et vous auriez été bien plus utile ! avec beaucoup moins de peine.

Comprenez que si j'étais membre du PPC.T. j'aurais été gêné qu'un des membres. publie un tel recueil. inutilité.

Sincères Salutations

Phonon

ALAIN MERCKX
PPC-T 348
PPC 10859
av. des Traquets 142
B-1150 BRUXELLES
BELGIQUE

Bruxelles, le 14 août 1983.

Jean-Daniel DODIN
rue du Cagire 77
F-31100 TOULOUSE

Cher Jean-Daniel,

Possesseur, depuis bientôt 2 ans, d'une HP-41 CV, mon adhésion récente aux clubs PPC (USA) et PPC-Toulouse, et surtout la lecture de ton livre "Au Fond de la HP-41", m'ont permis d'entrevoir des possibilités d'exploitation de ma machine que j'étais loin de soupçonner. J'utilisais déjà depuic plus d'un an la programmation synthétique, mais le microcode est pour moi une nouvelle découverte et je suis fort tenté de passer à la pratique de ce genre d'exercices. Seulement voilà : je dois d'abord me procurer l'équipement nécessaire à ce nouveau sport. J'essaie donc en ce moment de rassembler un maximum d'informations concernant le matériel actuellement disponible, et c'est là une des raisons pour lesquelles je t'écris : pourrais-tu m'envoyer plus amples renseignements sur le MLDL dont la souscription a été lancée en juin ? D'après le bulletin (n° 6, p 22), il semble que la capacité du lecteur d'EPROMs ne soit que de 4 k, ce qui me paraît fort limitant : une EPROM ("Operating System") étant fournie avec le MLDL et apparemment nécessaire à sa programmation, il me faudrait donc acheter un lecteur d'EPROMs supplémentaire pour pouvoir exploiter simultanément le Toulrom et le HP-IL Monitor que je compte acquérir. Dis-moi si je suis dans l'erreur... Autre question : il est possible, avec tous les systèmes semble-t-il, de copier des programmes microcodés en RAM sur cassette, mais ces systèmes sont-ils compatibles entre eux ? Autrement dit, peut-on charger la RAM d'un MLDL PPC-T, par exemple, avec des programmes écrits sur un autre MLDL et enrégistrés sur cassette, ou vice versa ? Si l'annonce m'étais parvenue plus tôt, je ne serais peut-être lancé tête baissée dans la souscription à 1700 F, mais si le nouveau prix est bien 2500 F, cela demande plus mûre réflexion : cela ne m'enchanterait guère de commander un tel appareil outre-Atlantique (longs délais, ou alors coût du transport aérien vraiment exorbitant), mais j'avoue être quelque peu attiré par le MLDL de 24 k ROM + 8 k RAM produit par ERAMCO aux Pays-Bas pour 400 dollars...

Si tu pouvais me répondre en quelques mots à toutes ces questions, cela m'aiderait beaucoup, car je dois prendre une décision très rapidement : en effet, je devrais, dans les prochains mois, partir travailler pour plusieurs années au Cameroun, où j'enseignerai l'alimentation animale à l'Ecole Nationale Supérieure Agronomique. J'attends incessamment une confirmation de la part des autorités locales. De toute façon, mon adresse à Bruxelles est celle de mes parents et reste valable, jusqu'à nouvel ordre, pour l'envoi du bulletin.

Tout autre chose : j'ai essayé sans succès le programme de transformation de fichiers XF (PPC-T n° 5, p 6) : le programme s'arrête sur GETX au pas 42 et affiche FL NOT FOUND... Effectivement, si je fais MLDIR à ce stade, je constate que tout a disparu (DIR EMPTY). La cause semble être la normalisation, inévitable avec le STO IND L du pas 98. J'ai constaté que lorsque X est le seul fichier présent en EM, on peut sans problème modifier sa taille alternativement à 1 ou 360 régistres (SF 04 XEQ 03, CF 04 XEQ 03), mais si d'autres fichiers le suivent, on peut le ramener à 1 régistre mais l'EM est perdue si l'on tente de l'étendre à 360 régistres. D'autres que moi ont-ils eu des problèmes avec ce programme ?

Enfin, je termine par une commande : afin de compléter ma collection, pourrais-tu m'envoyer les numéros 3 et 4 du bulletin de PPC-T, pour lesquels je joins un eurochèque de F. 30.-. Par la même occasion, voudrais-tu y joindre une photocopie des pages 13 et 14 du n° 6, car cette feuille manque dans mon exemplaire. D'avance, merci.

J'espère que je n'ai pas été trop long, ni trop ennuyeux. Je joins une enveloppe timbrée afin que tu puisses m'envoyer quelques mots en réponse à mes questions concernant les MLDL, et plus particulièrement quelques données techniques sur celui du club (n'hésite pas à écrire à la main pour ne pas perdre trop de ton temps !). Comme tu le vois, je ne suis pas encore rodé aux clubs PPC, mais j'espère que d'ici quelques mois je pourrai t'envoyer du courrier un peu plus concret (notamment des programmes...)

Dès à présent, je te remercie vivement de ton amabilité et, dans l'attente du plaisir de te lire, je t'envoie mes plus amicales salutations et te souhaite une bonne programmation !

Alain Merckx (T348)

REIGNER Cyril (T282)
4,rue de MORLAIX
44800 SAINT-HERBLAIN

Cher Jean Daniel,

J'ai tellement de questions à poser que je pense que je vais en oublier la moitié pendant que j'écris cette lettre.

J'ai acheté il y a peut de temps le module HP-IL et une unité de stockage de masse et,naturellement,je me suis amusé avec.J'ai donc essayé de "CREATE" un fichier de 16000(500 enregistrements), ma 41 l'a bien créé mais pendant le rembobinage de la bande j'ai eut droit à un "TRANSMIT ERR" lorsqu'il restait environ 1/5 de bande à gauche du lecteur!J'ai exécuté ensuite un DIR et tout a bien fonctionné,puis j'ai stocké quelques données par-çi par-là puis j'ai fait VERIFY:aucun problème,mais quand j'ai demandé ZERO à ma machine j'ai de nouveau eut un "TRANSMIT ERR" à la même position de la bande!

Quelqu'un a t'il eu ce problème et pourra m'explier pourquoi?Le seul cas possible d'une telle erreur serai un support defectueux mais ma bande enregistre bien les données(prgm,alpha,...)?HELP

Au passage je signale que HP a fait une erreur dans le manuel de l'HP-IL;en effet au message "DUP FL NAME" on peut lire que ce type d'erreur survient si vous voulez créer un fichier du même type et de même nom que celui déjà sur le support,ce qui est faux(essayer)!Prenez le manuel à la page d'une des fonctions pour lesquelles s'applique l'erreur (par exemple WRTK),ont y lire:si un fichier du même type et de même nom existe dans le support la 41 ~~si~~ stocke ses valeurs dans ce fichier et par conséquent les anciennes valeurs sont perdues.Je vous laisse faire la correction(m on manuel est de juillet 82).

Je voudrais savoir si il existe au club de la documentation sur le Microcode autre que ce qu'il y a dans"au fond" car ayant commandé les listings des ROMs de la HP je butte souvent sur des fonctions où je n'ai pas assez d'information.Par exemple,quelle est l'action veritable de GOTO ADR?Tu nous dit

dans l'annexe sur l'afficheur que les bits 0 à 5 doivent avoir une valeur de 0 à F(action sur le bit 8, pour avoir des caractères sinon ont obtient que des "SPACE",cela ne fait que 16 caractères or la 41 en affiche beaucoups plus et qui ne sont pas du rang 4! Il me semble qu'il y a un point que je n'ai pas suivit.Autre chose la fonction SST utilise 3 fois de suite WRIT 4(L) ce qui pousse 12caractères dans l'afficheur or le registre C a dans les digits 0 a 11 des valeurs particulieres.Où sont pris les groupe de quatre caractères dans C?A droite,a gauche,au centre?Un caractère équivaut à combient de digits? J'ai essayé de m'y reperer avec la table des caractères affichables en Microcode mais ce n'a rien donné.

Pour toutes ces questions et bien d'autres je reclame ton aide,ou celle d'un autre membre.

J'ai aussi étudié la "fonction" STO 12 mais je demande si quelqu'un a trouvé le point d'entrée dans le microcode de cette fonction ?

STO 12 charge en X (pas de mouvement de pile)l'équivalent des flags 00 à 43 (aucun lien avec RCLFLAG du X-FONCTIONS qui utilise le code 31 ou W) suivant le codage suivant:

DIGIT:	13	12	11	10	09	08	07	06
N° des flags:	00-03	04	05-08	09-12	13-16	17-20	21-24	25-28

DIGIT:	05	04	03	02	01	00
N° des flags:	29-32	33-36	37-40	41-43	met -5 si executé en mode PRGM,met 3 si executé en mode RUN ??	

Pourquoi le flag 04 est-il tout seul ?

Je pense que c'est à cause du fait que c'est le dernier drapeau visible à l'afficheur,Il en resulte un decalage des autres digits.

Quel peut-être la signification de l'exposant ?

Ce que j'ai écrit plus haut pour l'exposant n'est pas valable à tous les coups mais ce sont les valeurs que j'ai rencontées le plus souvent.Le fait que les flags empiètent sur le digit 02 ne gène pas la 41 pour les exposants négatifs est extrêmement bizarre(vous avez dit bizarre ?).

Si le flag 44(ON continu) est armé l'affichage devient "BLANC" ??

Tant que je n'aurai pas trouvé le point d'entrée de cette fonctionsje ne pense pes pouvoir expliquer entièrement STO 12;or pour trouver cette entrée il me faudrait plus de renseignements sur le microcode....

Dernier point à cette lettre j'ai un programme MGN plus court et à base d'instructions synthétiques,que celui presenté dans le journal de janvier-fevrier.A titre d'exemple il permet un resultat de 735 chiffres!(Mais on peut aller jusqu'à 790 chiffres pour les cas spéciaux !)

Ca y est j'ai enfin fini ma lettre (ouf) et je dit bonjour à tous les membres de PPC de la par de la Société d'Astronomie Amateur de Nantes (où il y a des HPistes enragés qui ne sont pas,encore,de la maison).

HEUREUSE PROGRAMMATION

Reignier (5282)

Jean Daniel Dodin
77 rue du Cagire
31100 TOULOUSE

Eric Angelini T210
14 Av. Marianna
06000 NICE
Tel DOM (93)530799
FAC (93)519100 p1334

Jean Daniel

Je me permets de t'ecrire ce petit mot pour trois raisons:
-Mon role de correspondant régional.
-Mes projets au niveau HP 75.
-Une application par HP du protocole HPIL en instrumentation (réseau INET apparu fin 83).

Mon activité de coordinateur se résume à l'adhesion de 1 ou 2 nouveaux membres, et à l'affichage de plaquettes du club. Je pourrais faire mieux...faute de temps... Au vu de la liste des adéhrents (qui s'arrete vers 450) très peu gravitent dans ma region .Je lance donc un appel à tout ceux désireux de se rencontrer .Ils peuvent me joindre par téléphone ou par courrier à l'adresse ci-dessus.Je ferai un courrier à tous ceux qui ont peur de s'intoxiquer en collant un timbre ou de s'électrocuter en touchant au téléphone.
Il s'est glissé une erreur de frappe dans mon adresse de coordinateur ,ce n'est pas T102 mais T210.

Pour le HP75, de plus en plus de choses trottent dans ma tete... ROM APRIL ... .. ASSEMBLEUR .. .. DISQUETTES .. .. FORTH ..
Pour la ROM APRIL je suis trés interessé. Permettra-t-elle effectivement d'effectuer une gestion de fichier digne de ce nom (équivalente à celle du 85 par exemple etc)?
De plus dans le numéro 1 de Micro Revue 2 articles parlent de ROM april et I/O ,il sagit bien de la meme ?
Partant de là, l'unite de disquette de 360K devient aussi intéressante.Le nouveau lectuer HP9114 (720 Ko par disquette 3 pouce en HPIL bien sur 795$) l'est encore plus.
L'assembleur et le FORTH vu ma configuration limitée actuellement (un HP75 tout seul) me paraissent peu utilisables ...(j'aimerai que tu me dises le contraire).
Enfin tout cela augure des investisements plus ou moins importants...(AIE AIE)

Enfin dernier point de ma lettre :le réseau INET.
Il s'agit d'une boucle(qui l'eut cru) mettant en communication un controleur unique (un intégrateur HP3392) avec deux périphériques (injecteur automatique HP7672 et un chromatographe HP5890). Les cables et le protocole sont HPIL mais oui . Je vous vois venir .NON, mon 75 ne reconnait pas ,à juste titre ,ces nouveaux périphériques.
Que fait cette boucle ?
Elle permet la transmission d'un signal analogique vers l'integrateur.
L' intégrateur accumule les informations concernant les divers appareils présents sur la boucle.
Enfin l'ensemble passe d'un mode "STAND BY" à un mode actif simultanément.
D'aprés un spécialiste HP ,un HP 86 possédant l'interface HP80/HPIL peut traiter directement le signal chromato. et le ressortir sur écran etc.(me contacter pour plus ample information).

Pour conclure la nouvelle présentation est presque parfaite. La mise en page et la frappe des articles sont encore un peu en désordre. Il est vrai que moi le premier, je contribue à ce désordre en frappant cette lettre sur Apple/ImageWriter. Ne pourrait-on pas envoyer nos textes sous forme de fichiers sur cartes ou casettes ? (En en standardisant la forme) Ce serait tout à fait envisageable pour les 75, 71 ...Il n'y aurait plus qu'à les éditer sur une seul et meme imprimante.
En fait ceci n'est qu'une suggestion, le journal devient "impressionnant" par sa quantité et sa qualité.
Bravo a tous.

Bonne rentrée!

TENDRESSES

Lorsque l'on est abonné à PPC CJ (le journal américain), ce qui frappe sans doute le plus au premier abord c'est la présentation. En effet ce journal est très bien mis en page et les articles qui le composent sont apparement d'une grande régularité géographique. Bravo donc ! On a l'impres-sion qu'un équilibre a été trouvé entre la clarté visuelle et le compactage de l'information. Seulement voilà : au deuxième abord, lorsque l'on lit les articles : l'impression est tout autre.

En effet la majorité des textes envoyés par les membres commencent par quelques lignes de remerciements, de commande de matériels, de réabonnement, etc... De fait, on ne sait jamais, du moins au premier coup d'oeil, ni où commence véritablement l'article, ni quel est son objet ! Et pourtant, qu'il serait facile de commencer par un titre, qu'il serait simple de joindre les commentaires personnels et étrangers à l'article sur une feuille séparée, qu'il serait tentant de supprimer ces abominables signatures manuscrites, qu'il serait utile de tous taper nos articles au format du club et qu'il serait précieux de vérifier l'état des rubans de

***JE DISAIS** DONC :*

nos machines à écrire avant de commencer à dactylographier.

Par exemple : si ces quelques conseils avaient été suivis dans le n°5 de notre journal, on aurait pu lire une bonne page d'articles supplémentaires.

Alors ce plaisir ne vaut-il pas la peine d'un peu d'effort? Ne peut on vraiment pas respecter ces dix comman-dements :

- *tu n'oublieras pas le titre de ton article*
- *tu ne traiteras pas trente six sujets sous le même titre*
- *tu ne signeras pas autrement que par ta machine à écrire*
- *tu n'entèteras pas tes articles*
- *tu ne taperas pas dans un autre format que celui du club : 13,5 cm de large et une interligne* ← ce n'est plus vrai, utilisez le format commercial ordinaire pour MICRO-REVUE
- *tu ne termineras pas par une formule de politesse*
- *tu banniras le Post-Criptum de ton univers dactylogra-phique*
- *tu n'oublieras pas de remercier, commander, te réabonner, féliciter sur des feuilles séparées de ton article*
- *tu ne négligeras pas l'entretien de ta machine*

et enfin :

- *tu essayeras une fois, pour voir, de mettre en page tous les articles que tu as écrits jusqu'ici.*

Alors, avec un peu de bonne volonté, nous auront bientot un journal aussi beau et bien construit que celui du PPC USA avec, en plus, le cadeau Toulouse : la joie de ne pas perdre son temps en lisant son journal.

En fait, on pourrait résumer toutes ces recommandations en une seule : observer attentivement la présentation des articles de J.D. Dodin (T1).

Voilà, je vous remercie de m'avoir subi pendant ces quelques lignes. A bientôt.

GEORGELIN Daniel T 57

PPC-T

MAGAZINE

LE COIN DU CLUB

LES JOURNAUX

Reçus depuis MICRO-REVUE 3 :

PPC Calculator Journal Septembre-Octobre 84 (V11N8) avec en particulier une astuce permettant de connecter la HP-41 au monde extérieur en utilisant des contacts prévus sur la plaquette du module Time. Computer Journal of PPC Septembre-Octobre 84. PPC POBOX 9599 Fountain Valley California 92728-9599 USA.

CHHU Chronicle, Octobre 84, Le nouveau journal de Richard Nelson. Il y a environ 350 inscrits à CHHU pour l'instant, ce qui n'est pas mal en si peu de temps. Ce premier numéro est très beau, avec en particulier des photos très bien imprimées montrant l'intérieur du HP-71 et la façon d'y installer des modules. Compte tenu des nouveautés prévues, il sera bientôt possible d'avoir 369,5 Koctets (RAM et ROM confondus) dans la poche. Le prix de revient est hallucinant ! Egalement dans CHHU Chronicle, comment envoyer sur la Thinkjet avec une taille variable le contenu de l'afficheur du 71. Exellent. CHHU 2545 Camden Place, Santa Anna CA 92704 USA. Richard a investi $5000 dans cette affaire, et il n'est pas seul. Au moins deux de ses amis ont une Laserjet et de plus en plus d'articles seront sortis avec...

USER n°5, le journal Danois, 93 pages. Annonce le travail de Vaudenay sur le FORTH/41. PPC Danmark Postbox 2, 3500 Vaælrose, Danemark.

HP-Journal de Octobre, Novembre et Décembre (3 numéros) consacrés à l'instrumentation (chez HP France).

PRISMA Octobre/Novembre. Les allemands semblent faire du mauvais esprit. Ils n'ont annoncé ni le ROM Paname ni la préparation du FORTH/41. Je ne suis même pas sur qu'ils aient annoncé le ZENROM (mais là je peux me tromper). Tout cela sans doute pour ne pas faire concurrence au CCD ROM. Lequel n'est toujours pas disponible. Ce numéro de PRISMA contient de nombreux programmes professionnels (Batiment, RDM, Météo...).

JEDI n°6 (Octobre) et 6 (Novembre) (9 et 11 pages). Se diversifie entre l'astronomie, le FORTH et les langages nouveaux (LISP, LOGO). Annonce le sommaire (FORTH) de MICRO-REVUE à chaque fois, merci. JEDI, 8 rue Poirier de Narcay 75014 Paris.

JPC Novembre 84. 42 pages. De plus en plus axé sur le HP-71. JPC, Philippe Guez, 56 rue JJ Rousseau, 75001 Paris.

HP Club Spiegel n°10, le petit journal des suisses Alémaniques. Annonce lui aussi le FORTH/41 de Serge. HP club (Schweiz) Franco dal Molin, Platten strasse 44, 8152 Glattbrugg.

DATAFILE V3N4/5, le journal anglais (annonce aussi le FORTH/41...) Contiendra sans doute une rubrique consacrée au ZENROM. 52 pages. PPC UK, c/o Astage Rectory Lane, GB-Windlesham,GU20 6BW Angleterre.

FORTH Dimension V6N1. J'ai l'impression d'avoir manqué un numéro, à la suite d'une mauvaise transmission de mon renouvellement de cotisation, je vais vérifier. Ce numéro suit une nouvelle présentation, avec une classification des articles par niveau de difficulté. Il semble toujours pêcher par défaut d'homogénéité, ce qui rend sa lecture décevante. C'est quand même une source appréciable. J'ai également reçu du FIG-FORTH un jeu de photocopies qui sont une reproduction de journaux internes aux chapitres. Il y en a un assez gros paquet (env. 100 pages) de qualité très variable. Ce lot ainsi que les suivants quand nous les recevront sera placé dans notre service photocopie pour que tout le monde puisse en profiter. J'ai également reçu (toujours à titre de chapitre) un gors polycop de 352 pages "1983 FORML Conference proceedings". C'est remarquable pour ceux qui s'intéressent à l'avancement du FORTH. La reproduction en est interdite, mais vous pouvez vous le procurer (je crois $35) au FIG FORTH Po Box 1105 San carlos CA 94070. Le FIG accepte les règlements par carte VISA, si vous n'êtes pas à quelques $ près, vous pouvez donc commander même sans connaitre le prix exact en joignant le numéro de votre carte et sa date de fin de validité.

Enfin nous avons reçu PROMPT, une petite revue photocopiée de 14 pages du club flamand (Gerlof Donga, Quellijnstr. 47, 1072 XP Amsterdam Belgique, et SCHEDIR n°21 de R. Pulluart, décidément très actif R Pulluart, Lusterlaan 31, 2665 TH Bleiswuk Pays Bas.

PETITES ANNONCES

T40 vends -40% sur prix discount, modules mémoire, 1 quadram et 4 simples, X-fonctions, Horloge, lecteur de cartes, imprimante 82143, boucle HP-IL, interface vidéo, port d'extension SICAPE, tel 19-20h (1) 875 22 30.

Carlos F. Cano Lanzadera (T609) vends une HP-41 + module mémoire Quad Janvier 81, 1650F. A. Sainz de Baranda 99, 28007 Madrid Espagne.

T488 Jean-Claude Liehn 8 rue de l'écaille 51100 Reims tel (26) 85 50 52 vend (cause achat disquette) lecteur de cassettes HP 82161A Juin 83 avec 7 cassettes. Parfait état, le tout 3000F, livraison possible sur Paris.

T301 Jaworski Daniel 50 Bd du Barlet 62640 Montigny en Gohelle, (21) 75 42 27 vends module HP-IL neuf, HP-41CV, lecteur de cartes, imprimante 82143, lecteur optique, modules Time, X-F, X-mem(2), chargeurs (2), 6 étuis de cartes vierges, 40 rouleaux de papier bleu, matériel neuf, prix à débattre.

T266 Stéphane Bartial, 2 rue Grande, 89340 Villeneuve la Guyard (86) 66 40 77 vends HP-41CV, lecteur de cartes, HP-IL, X-F, Math, accus + chargeur, documentation, programmes, 2 libres (Prog 41 et AF) le tout 5700F (cause achat HP-71).

T441 vends module Time (aout 82) avec manuel en francais ou en anglais 300F.

Thierry Pethuy (T14) 13 Bd de Baudricourt 54600 Villers-lès-Nancy (8) 398 72 07 vends tout un matériel HP-41 avec manuels et documentation, dont PPC ROM. le contacter pour détails.

T391 François Le Grand rue de la poterie Trévignon, 29128 Tregunc, vends une imprimante 83143 et Cherche un lecteur de cassettes 83161.

T259 Henri Bourboulon 156 av de la libération, 59140 Dunkerque donne un module de mémoire simple et vend 200F un module X-F (mai 83).

Remarque: si vous avez des modules simples inutilisé (ou la boite en plastique et les contacts, vous pouvez en faire cadeau au club. Nous avons usage des connecteurs, et ne rechignons pas à souder entre eux les modules pour en faire des banques de données. Merci d'avance.

COTISATION

NON, n'ayez pas peur, la cotisation n'augmente pas. On ne peut pas en dire autant pour tout en cette nouvelle année ! Simplement c'est la méthode de rappel de cotisations qui change.

Jusqu'à présent nous mettions avec le dernier journal une feuille d'"appel de cotisation". Cela nous posait des problèmes pratiques. Nous allons essayer une nouvelle méthode. Dés le présent numéro, un bulletin d'adhésion figurera en fin de chaque MICRO-REVUE. Sur ce bulletin, en haut à gauche, se trouve un espace blanc. Si il est bien blanc, vous n'avez rien à craindre. Si il porte en rouge **DERNIER NUMERO,** il est temps de renouveler votre cotisation, dont vous trouverez le montant à la page précédente dans "Organisation de PPC-T". N'oubliez-pas de nous retourner le bulletin d'adhésion rempli. Son emplacement a été choisi pour que votre MICRO-REVUE ne soit pas abimé quand vous détacherez le bulletin. Ainsi, tant que vous n'aurez-pas renouvelé, à chaque fois que vous lirez MICRO-REVUE, vous pourrez vérifier si vous n'avez pas oublié de renouveler... Vous pouvez également vérifier l'étiquette adresse qui vous donne le dernier mois de l'abonnement et le dernier numéro de MICRO-REVUE que vous devez recevoir. En cas d'erreur, prévenez-nous, merci.

SUR LE MARCHE

Le Point sur les affaires en cours :

Le ZENROM est disponible et toutes les commandes en instance ont été livrées. Il arrive au rythme de 10 par mois. Essayez d'un voir un, il faut l'avoir utilisé pour en comprendre l'intérêt.

A ce sujet, une remarque générale. PPC-T est une association loi de 1901, ce qui signifie que le club ne peut pas distribuer de bénéfice à ses membres (ni même à ses responsables). Du coup il est considéré vis à vis de la TVA (taxe à la Valeur Ajoutée) comme un simple particulier. Il en suit que nous ne pouvons **pas mentionner une TVA remboursable sur nos factures.** Cela n'est pas gênant pour nos clients particuliers, et guère gênant pour les professionnels qui commandent de la librairie car la TVA y est faible (7%). Mais pour des matériels comme les ROM, compte tenu de leur prix et de la TVA à 18,6%, cela peut devenir significatif. vous êtes nombreux à faire acheter votre matériel par votre entreprise (et c'est normal pour du matériel professionnel). A vous de voir, le club fait en général moins cher qu'ailleur, mais vous ne pouvez déduire la TVA. Si vous tenez à cette déduction, adressez-vous à votre magasin le plus proche (mais nous ne pourrons pas lui fournir les produits, il devra se les procurer autrement). Vous pouvez également vous procurer ces ROMS aux Editions du Cagire. Je tiens à préciser encore une fois pour les nouvaux que les éditions du Cagire (dont ma femme est la gérante) n'ont aucun lien organique avec le club, il ne faut donc pas les confondre, même si l'adresse est la même. J'essaie autant que possible de ne pas favoriser les EdC par rapport aux autres fournisseurs cités dans ces colonnes, j'espère y parvenir.

Le ROM Paname est en principe disponible. En fait nous n'avons pas encore reçu les modules que l'on doit nous livrer, mais il s'agit semble-t-il d'une erreur de BCMW et cela devrait être réglé quand vous lirez ces lignes.

Toujours pas de nouvelle du module CCD. Nous pouvons vous fournir le descriptif complet contre 3 timbres à 2,10.

Notre catalogue hiver/document de présentation a été expédié fin Décembre. Les nouveaux adhérents qui n'étaient pas au fichier le recevront avec le présent numéro. Il contient plusieurs éléments remarquables : Le premier est **IND41**. Il s'agit d'un index de tous les programmes pour HP-41C publiés dans le monde entier en français, anglais, allemand et hollandais. Il y a environ 32 revues répertoriées et 440 programmes dans cette première édition. Robert Pulluart, le maitre d'oeuvre de ce travail bénévole continue à éditer des fiches, et la prochaine édition (dans un an pourrait comporter plus de 900 programmes). Attention, IND41 ne contient pas de programmes, seulement une fiche descriptive **dans la langue d'origine du programme.** Connaissant l'adresse de la revue (adresse fournie dans IND41) il vous appartient alors de vous procurer le numéro concerné.

Nous avons essayé de maintenir le prix d'IND41 le plus bas possible, car il nous semble être nécessaire à tout utilisateur sérieux de la 41. Mais faute de connaitre son succés à l'avance nous n'avons fait qu'un petit tirage (300 ex) en offset. Si le succès est grand, le prix de la prochaine édition devrait être stable, même si il y a beaucoup plus de pages. Selon le succès, toujours, nous étudierons avec Robert de quelle façon pourront être faites les mises à jour. Le livre est déja imprimé de façon à permettre la perforation des feuilles pour le rangement en classeur, mais les perforations ne sont pas faites. Il me semble compliqué de faire des mises à jour avec classement des pages. Le plus simple est de tout réimprimer (un peu comme l'annuaire du téléphone). Dites nous votre avis.
prix de IND41 : à l'unité F80 (F90 par avion), de 10 à 50 exemplaires : F60 (F70 par avion), par 50 ex et au-dessus F50 (F60 par avion). Pour quantités plus importantes, nous contacter.

Ensuite nous avons au catalogue tous les modules (PPC ROM, ZENROM, PANAME).

Puis nous avons décidé de vous fournir (surtout pour les nouveaux adhérents) sur cassette l'ensemble des programmes publiés. Pour des questions de commodité, nous avons préparé 4 cassettes, correspondant a l'album 82/83 (PPC-T grand format) pour la cassette n°1, au 4 numéros PPC-T petit format (cassette n°2) et à MICRO-REVUE 1 à 3 (cassette n°3). Il n'y a dans ces cassettes que les programmes que l'on nous a fournis sur cartes ou cassette lors de la publication, plus quelques gâteries. Liste ci-jointe. Si vous avez tapé d'autres programmes de ces numéros et que vous pouvez nous les envoyer, nous les ajouterons. Deux ou trois programmes ont été perdus suite à des ennuis de lecteur de cassette nous les ajouterons si nous les retrouvons. Le prix nous parait raisonnable (160F chaque) compte tenu du prix des cassettes. Bien sûr les programmes sont toujours disponibles gratuitement à l'unité chez Damien Debril. Suite à exés de travail, j'ai beaucoup tardé à lui envoyer les programmes des derniers numéros, mais c'est maintenant fait.

Nous avons aussi de plus en plus de photocopies sur le FORTH, ce qui commence à constituer

une documentation respectable.

Les livres

Nous avons reçu un livre très intéressant de Marc Petreman et Michel Rousseau, "Tours de FORTH". Ce livre est un complément très appréciable à Débutez en FORTH de Brodie. En effet Brodie décrit les Bases du FORTH, mais peu d'applications. Petreman et Rousseau reprennent les base de la programmation (et non plus du FORTH) en les appliquant à ce langage. Nous revoyons ainsi les algorithmes et les organigrammes (y compris ceux spécifiques au FORTH), ainsi que de nombreuses applications. Le type des applications est sans importance, par contre chaque application est expliquée en détails et permet de découvrir de nombreuses astuces de programmation. Les programmes sont développés surtout sur ORIC, mais FORTH merci, sont aisément transférables sur toutes les autres machines. Un livre qui m'a paru très utile.

Le ROM MATH (proposition de Eric Aubourg) et le ROM FORTH

Après examen des réponses pour le FORTH (manque encore la publication dans PPC Journal), il apparait qu'il n'y a pas assez de clients pour faire un ROM/custom. J'ai donc étudié une autre possibilité, qui est celle du lecteur d'EPROM. La plus-part des acheteurs éventuels ne possède pas cet accessoire. J'ai donc pensé qu'il serait possible d'effectuer un achat groupé de lecteurs d'EPROM).

En laissant un petit bénéfice à Serge Vaudenay et de quoi couvrir les frais pour le club, le jeu d'EPROM de 8K (équivallent à un module double) revient "rempli" à environ 500F. Il y a actuellement à la SCIP des lecteurs d'EPROM (ancien modèle) en réclame à 600F pièce. Le total de 100F n'est pas supérieur à celui d'un module. Par contre, quand le stock en solde sera épuié il faudra compter le double (1200F TTC) pour le lecteur, mais avec la possibilité d'avoir 16K d'éproms). Cela fait un prix de 1700F. C'est un peu cher pour un seul module, mais si il y a un deuxième ROM/EPROM cela devient beaucoup plus intéressant. Si vous vous sentez concerné, écrivez-moi, vous serez contacté directement dès que l'affaire sera plus avancée.

Cette façon de concevoir les choses redonne un grand intérêt à la proposition d'Eric Aubourg (MR3, p127). B. Jousse (T502) (Barzun, 64530 Pontacq) veut s'y associer. Il propose que ce projet soit repris par le club tout entier avec les règles suivantes : participation bénévole des auteurs, recensement des programmes déja publiés pour incorporation dans le module, écriture si nécessaire de routines en langage machine pour accélérer le traitement...

Comme il se propose pour coordonner le travail, je suis tout à fait d'accord.

N'oublions pas, cependant, que le plus difficile à faire est le **manuel**. Je propose que Bernard et Eric se partagent le travail, l'un coordonnant les programmes et l'autre le manuel. Il me parait souhaitable de définir un module de 8K. Mais il n'est pas difficile de prévoir deux modules de 8K chacun, la plus part des lecteurs d'EPROM acceptant 16K. Le gros avantage des EPROMS est de permettre des mises à jour et des corrections quand des erreurs sont découvertes.

Je vous propose, dès maintenant, de porter l'effort sur **les procédures d'entrée et de sortie.** Il est important que des programmes différents ayant le même but utilisent les données de la même façon et donnent les résultats de la même façon. Ainsi, par exemple, un programme d'intégration en langage utilisateur pourra un jour être remplacé par un programme en langage machine sans qu'il y ai à réécrire les programmes qui l'utilisent. Il faut également que les programmes du module puissent être appelé comme sous programme en courcircuitant les demandes d'entrée et les messages de sortie. Une caractéristique intéressante serait d'avoir tous les messages pris dans un fichier placé en X-M, ce qui permettrait une traduction dans toutes les langues. Une approximation acceptable de ce système serait de localiser dans un fichier du ROM les messages, ce fichier étant disponible en plusieurs versions (par exemple une version en français, une en anglais... L'utilisation des Eproms facilite ce genre de choses.

Donc que tous les intéressés écrivent à Bernard et Eric (envoyez-moi un double, et peut-être bientôt un TOULMATH ?

Bernard Jousse (adresse ci-dessus) peut vous fournir gratuitement des programmes pour pétroliers. Ajoutez 2 ou 4 cartes à celles nécessaires pour le programme pour les photocopies de la notice et les frais de port retour : DIRIGE (4 cartes) effectue les calculs nécessaires pour le suivi de trajectoire en forage dirigé, HYDRO (2cartes) effectue divers calculs liés aux pertes de charges aux duses (?) de l'outil, KICK (3 cartes) effectue les calculs nécessaires pour un contrôle de venue par la méthode dite "driller's method". Je vous signale par la même occasion qu'une société américaine (Petrophysical Data Consultant, 14614 Falling Creek, suite 210, Houston, TX 77068, phone (713) 537 5500) diffusait en 1982 un jeu de programmes pour pétroliers et qu'il était question d'en faire un ROM de 8K (certainement très cher).

On me signale que dans un catalogue EDUCALC récent (que je n'ai pas reçu) est montré un écran à cristaux liquides de 25 lignes de 80 caractères (1,5kg, sur piles) avec interface parallèle et série (interface HP-IL extérieur disponible), au prix de $599 (plus éventuellement $144.95 pour l'interface). HP doit aussi annoncer quelque chose de ce genre bientôt.

On nous signale aussi la mise sur le marché prochaine par un spécialiste en topographie d'une extension mémoire 32K sur HP-IL (interface HP-IL et RS232) au prix astronomique de 20 000F !

La société Wild France vient d'éditer une brochure décrivant une modification simple de l'interface HPIL/RS232 Hewlett-Packard, permettant la connection de la HP-41 au T2000 Wild. Avec cette brochure la modification semble très facile.

cassette PPC-1

NAME	TYPE	REGS
CREA	PR	26
LISTE	PR	19
BADTA	DA	37
BA	PR	15
DF	PR	32
CDR	PR	100
DEFI	PR	7
DEFILE	PR	12
INTER	PR	37
TAB	PR	52
651	PR	56
K	PR	13
T	PR	15
PILO	PR	11
FACTO	PR	27
FACTO2	PR	32
ALFA	PR	21
TTT	PR	12
TTT2	PR	15
T3	PR	5
LON	PR	17
DH	PR	12
DN2	PR	14
PILOTE	PR	45
PILO2	PR	11
CFA	PR	82
LFA	PR	13
DD	DA	23
DH	PR	20
RE	PR	66
WI	PR	83
P*	PR	33
BASES	PR	46
aRM	PR	9
INVA	PR	8
CALAN	PR	43
WATT	PR	64
BAIRST	PR	79
X	PR	81
MGN	PR	26
HIDDEN	PR	35
LEM	PR	144
AA4	PR	17
AA3	PR	13
EE	PR	3
DISBIN	PR	175
LA	PR	16
LEFT	PR	41
HIDDEN2	PR	30
MIND	PR	125
DPC	PR	31
DRVTST	PR	180
MA	PR	29
CREAT	PR	9
PD	PR	16
P	PR	16
SPEC	PR	96
HDUMP	PR	43
QUAD	PR	10
CAISSE	PR	111
CA	PR	74
TRANS	PR	41
IGR	PR	16
EDITEUR	PR	42
MASMIND	PR	44
CAT3	PR	32
TX	PR	9
IN	PR	14
EAO	PR	15
PX	PR	41
ASS	PR	109
DP	PR	32
ROT	PR	146
Y	DA	5
DX	PR	8
HPSOC	PR	64
M	PR	109
REVEIL	PR	6
CBG1	PR	96
IP	PR	13
AV	PR	186
AVDATA	DA	130
SL	PR	14
DATA	PR	12
CDE	PR	6
ACC	PR	26
CAN	PR	30
PND	PR	48
PCB	PR	52
TRI	PR	11
SAV	PR	46
HIST	PR	40
COD	PR	8
FAC	PR	32
ORP	PR	18
SAVE	PR	38
DEV	PR	25
B5	DA	10
TABMUL	PR	36
FERMEC	PR	77
FACTO3	PR	30
TSYNT	PR	12
C21	PR	40
PDIV	PR	11

cassette PPC 2

NAME	TYPE	REGS
CODEHP	PR	155
ASSEMB	PR	158
INPUT	PR	25
INPUT2	PR	28
DESAS	PR	27
T0	PR	57
XQ	PR	30
CHR	PR	36
C10	PR	31
CHR2	PR	28
COPCAS	??	8224
ENPAGE7	??	8224
ENPADOC	??	8224
NETTCAS	??	8224
LISTOLY	??	8224
DUMPCAS	??	8224
AZERTAC	??	8224
ENPAGOL	??	8224
OLY	??	8224
OLYDOC	??	8224
BENCH	??	8224
COPYDAT	??	8224
PEL	PR	52
PELIN	PR	30
FORMAT7	??	8224
TIR	PR	48
JACK	PR	80
IN2	PR	15
CACR	PR	21
CACRDA	DA	32
GAM	PR	35
APRO	PR	62
RENAME	PR	32
F	PR	28
PL	PR	11
HPILCMD	??	8224
AUTOASN	??	8224
MCOPY	??	8224

La cassette PPC-4 est prête. elle ne contient pour l'instant que les programmes de MR4 mais sera complétée avec MR 5 et 6 même prix (160F) voir Damien pour mag à l'unité

cassette PPC-3

NAME	TYPE	REGS
DCB	PR	19
SYMB	PR	53
INVC	PR	119
HIST	PR	52
CB	PR	14
XR	PR	16
TRIPLOT	PR	76
PUISS	PR	22
VM	PR	12
PM	PR	20
CHRO	PR	20
CHROV	PR	23
BCPLOT	PR	11
BA	PR	50
DC	PR,S	6
DCPGDV	PR	29
DCPGH	PR	16
HPILCMD	??	8224
AUTOASN	??	8224
MCOPY	??	8224
CANONK	KE	8
DUMP	??	8224
MAKEFIL	??	8224
PEEK	??	8224
INSTALL	??	8224
MEMDUMP	??	8224
MEMDIR	??	8224
UTIL3	??	8224
UNECOPI	??	8224
CATALPH	??	8224
FORMAT7	??	8224
CREECAS	??	8224
ECRICAS	??	8224
LIRECAS	??	8224
RIOWIO	??	8224
SECURE	??	8224
UNSECUR	??	8224
EP44	??	6150
KEYBOAR	??	2569
DON	PR	171
DONSOTI	PR,S	221
TRACE	PR,S	60
LST	PR,S	13
BAPBORN	PR,S	71
CBD	PR,S	61
CBNN	PR,S	26
TF	PR,S	28
DERIVEE	PR,S	14
CAR	PR,S	21

LE ROM PANAME

Dans Micro-Revue N°3, était annoncé le ROM PANAME. J-D a pu m'en prêter un et, avec seulement quelques feuilles photocopiées, je m'y suis attaqué pour voir ce qu'il y a dedans.

C'est un ROM de 8K, tout en microcode, occupant les XROM 05 et09 (comme le standard pac, le home management et hélas le ZENROM). Au catalogue 2, apparaissent 122 fonctions réparties sous 6 entêtes. Donc, 128 points d'entrée, c'est du tout plein. Malgré l'absence de documentation

et le nombre impressionnant de fonctions, il est assez simple à utiliser parce que:

Les mnémonniques sont bien choisis pour être explicites.

Les modes et paramètres de fonctionnement rappellent ceux des fonctions standard et restent logiques pour toutes les fonctions du modules.

Certaines fonctions reprennent les XF, l'XIO et la CX.

N'allez tout de même pas penser que ce module n'est q'une compilation de fonctions intéressantes piquées ici et là, c'est bien plus que tout cela et je vais essayer de vous en convaincre.

LES PREFIXES

Comme vous pouvez le constater sur le listage du CAT 2, il ya 9 fonctions commençant par OUT et 14 par RG. Pour aider à la frappe de toutes ces fonctions, vous affectez OUT ou RG à une touche et, pour entrer OUTLFX, frapper la touche assignée, ALPHA L F X ALPHA et la fonction OUTLFX est exécutée (ou mise en MEV si vous êtes en mode PRGM). Si vous avez l'IL, cela marche aussi avec OUTA qui pourtant, n'est pas reprise dans le PANAME. A l'usage, c'est vraiment très pratique. Avec deux assignations et un minimum de pressions de touches, un grand nombre de fonctions est épelé. De plus, le danger d'affecter OUTA ou RGINIT à une seule touche est écarté. C'est un très bon compromis entre la disponibilite des fonctions et la sécurité. On y prend rapidement gout et on cherche du coté de READ, WRT ou PR mais de ce coté, rien! Dommage, mais on ne peut pas tout avoir.

LES FONCTIONS POUR MINI PLOTTER

Elles permettent de commander le traceur de l'X07 ou de la Tandy CGP115 (modèles interfacés via l'HP IL et un convertisseur). Assez explicites, notons qu'étant assignées, ces fonctions permettent de faire travailler le traceur en mode manuel. Adieu chaines ALPHA bizarres et OUTA et bonjour la lisibilité des programmes. La même chose vaut aussi pour les fonctions de controle de la boucle, de l'imprimante 162 ou 905 ainsi que pour l'interface vidéo. Avec le PANAME, vous n'aurez plus à vous arracher les cheveux à rechercher la séquence d'échappement qui, de toutes façons, ne marche pas pour les conditions dans lesquelles vous vous trouvez.

LES UTILITAIRES

ALENG:	idem XF
ANUM:	idem XF
ANUMDEL:	idem XIO
AROT:	idem XF
ATOXL:	idem XIO
ATOXR:	idem XIO
ATOXX:	idem XIO
SIZE?:	idem XF
XechF:	idem XF
XTOAL:	idem XIO
XTOAR:	idem XIO
YTOAX:	idem XIO
PSIZE:	idem XF
POSA:	idem XF
X=NN?:	idem 41CX
XdiffNN?:	idem 41CX
XinfNN?:	idem 41CX
Xinf=NN?:	idem 41CX
XsupNN?:	idem 41CX
Xsup=NN?:	idem 41CX

NOP: pas d'opération, sans commentaire.

APPX: concatène à ALPHA la partie entière de X sans le séparateur décimal, mais précédé du signe si négatif.

TF55: valide ou invalide l'imprimante suivant l'état antérieur. Très utile : essayez avec une 41, une cassette et une iprimante sur la boucle de faire un DIR seulement sur l'écran de la 41.

VKEYS: visualise les touches assignées. En fait, est identique à PRKEYS mais sur l'écran exclusivement. Bien utile quand on n'a pas la chance d'avoir d'imprimante. R/S arrête le listage. Une pression sur une touche le suspend tant que la touche est maintenue.

Y/N: permet de répondre à une question binaire. Y/N? est concaténé à ALPHA et le tout est affiché. Un petit TONE est émis et toutes les touches sont verrouillées sauf :

ON qui éteint le calculateur,

R/S qui arrête le programme

CHS (0) ou * (Y) pour répondre OUI

ENTER (N) pour répondre NON

Si cette fonction est exécutée en mode calcul (peu d'intérêt), elle fonctionne comme un test sur les flags : la réponse YES ou NO est affichée. Pour les choix binaires, Y/N est bien plus souple que GETKEY.

WRTEM: sauvegarde de l'XM sur la cassette. Cette fonction crée sur cassette un fichier dont le nom doit préalablement se trouver en ALPHA. Dans ce fichier est stocké tout le contenu de l'XM. Ce fichier est d'un type spécial et ne peut pas servir à autre chose qu'à être relu par

READEM. Au DIR, le type apparait sous la forme ?? (ce qui est normal). Les autres fichiers non reconnus par la 41 (HP71, HP75...) ne sont pas valides à la relecture. Après chargement d'une sauvegarde, l'XM est remplie avec le groupe de fichiers constituant la sauvegarde. Les fichiers antérieurs sont supprimés même si la place était suffisante pour tout loger.

LES FONCTIONS SUR LES BLOCS

Elles nécessitent toutes 1 mot de controle en X et même quelque fois en Y, mot ayant la même structure que ceux de ISG et DSE. Ces pointeurs peuvent être assemblés par BLDPT et désassemblés par BRKPT.

RGINIT: initialise un bloc de registres pointé par X à zéro ou les remplit avec des nombres croissants à partir de 1.

RGNb: Met en X le nombre d'éléments du tableau dont le mot de controle doit être placé en X. Le mot de controle est sauvegardé en LASTX.

RGCOPY: 1 mot de controle en X, 1 mot de controle en Y copie les registres de X dans ceux de Y si X est positif, l'échange est opéré si X est négatif.

RGVIEW: Liste le contenu des registres de données désignés par le mot de contrôle placé en X. Pendant le listage, R/S l'arrete ; R/S à nouveau, et il repart.Quand le listage est arreté, SST est accepté, BST aussi, mais, si vous l'avez affecté, elle ne sera pas valide, vous devrez faire SHIFT SST. En mode SST, CLX arrete définitivement le listage. Pour modifier un registre, se placer sur celui ci avec BST et SST, puis, presser les touches numériques, vous composez alors le nouveau contenu du registre, vous le validez par R/S et le registre suivant est affiché (toujours en mode pas à pas). Si au lieu de presser les touches numériques, vous appuyez sur ALPHA, le mode ALPHA devient actif et vous pouvez rentrer 6 lettres par registre (si vous en rentrez plus, seules les 6 dernières seront prises en compte). La validation est aussi effectuée par R/S et le mode ALPHA reste actif pour le registre suivant. Bref, du bon boulot qui rapelle un peu le fonctionnement du module TIME.

SORT: Trie le bloc de registres désigné en X par ordre croissant ou décroissant. Cette routine trie aussi bien l'ALPHA que le numérique. Si le tri est mixte, l'ALPHA est toujours supérieur au numérique.

RG+-: Ajoute (soustrait) terme à terme les contenus des blocs de registres pointés par X et par Y (si X est négatif, la soustraction est effectuée). Le résultat se trouve dans le bloc pointé par X.

RG*, RG/: Effectuent la multiplication (la division) terme à terme des registres pointés par X et par Y. Attention, ce n'est pas une multiplication matricielle au sens mathématique du terme.

RG+Y, RG*Y, RG/Y: Effectue l'opération souhaitée entre le bloc de registres pointés par X et le nombre placé en Y.

RGAX: Suivant le signe du mot de contrôle placé en X, décompose ALPHA dans des registres ou recompose ALPHA avec les registres. Laisse en X le pointeur de registres incrémenté du nombre de REG utilisés. Très utile pour travailler sur la boucle avec INA par exemple.

LC-AD: Entrez le numéro de ligne, le numéro de colonne, le pointeur de bloc, cette fonction renverra l'adresse du registre, vous pourrez l'utiliser comme pointeur d'indirection.

AD-LC: Une adresse de registres en Y et un pointeur de bloc en X, cette fonction renverra la colonne et la ligne respectivement en X et en Y.

CHFLAG: Jetez RCLFLAG et STOFLAG au panier. Vousavez écrit un programme, en mode calcul, vous configurez la 41, repassez en mode PRGM, insérez en deuxième ligne l'instruction CHFLAG. En mémoire, sera enregistrée CHFLAG suivie d'une chaine synthétique. A l'exécution, le calculateur sera reconfiguré dansl'état original.

Il y a encore beaucoup à dire sur toutes ces fonctions et sur celles dont je n'ai pas parlées(CLINC, COLPT, GETRX, LINPT, RGORD, RGXTR, RGSUM, SAVERGX, STO L et SUB$) mais, je vais m'arrêter ici.

EN conclusion, c'est un rom super puissant, développé par des gens qui connaissent très bien les ROM de la 41 et de ses périphériques mais qui sont aussi des utilisateurs, ils savent donc ce dont les utilisateurs ont besoin. Un rom tel que celui ci va raccourcir vos programmes d'application, augmenter leur vitesse et les rendre plus lisibles. Je n'ai pas pu tout décortiquer mais je pense vous avoir donner un petit aperçu de la puissance de ce module. Bravo messieurs Dhénin et Barizien, il ne vous reste plus qu'a faire un manuel de référence à la hauteur de cette petite merveille.

Jean-Yves Pasquier

Note de l'éditeur : Depuis que Jean-Yves a écrit cet article nous avons reçu un exemplaire de pré-série du manuel. Il s'agit d'un gros volume de plus de cent pages en petits caractères qui reprend les 128 fonctions un peu comme le manuel de référence du HP-71, de façon très systématique et avec de nombreux exemples. L'ensemble a l'air très intéressant et valorise le module.

EDITIONS DU CAGIRE 77 RUE DU CAGIRE 31100 TOULOUSE
FRANCE - TEL 61 44 03 06
TARIF LIVRES ET MATERIEL. VENTE AUX PARTICULIERS
Prix Franco de port hors, valables jusqu'au 31-12-84

Facture sur demande. Seuls les ouvrages non disponibles à la bibliothèque de PPCT sont référencés ci-dessous.

(ref. EL3) **HET ONDERSTE uit de HP-41**, (door Dodin), prix **160F.**
(ref. EM15) **La cassette Programmation synthétique**, prix **150F.**
(ref. EL8) **Débutez en FORTH** (Brodie), prix **130F.**
(ref. EL9) **Calcul sur béton armé** (HP-41) (SOCOTEC), ISBN 2-3824, prix **110F.**
Revue australienne PPC-TN (club HP):
(ref. EL11) Numéros 1 à 6, les 6, prix **100F.**
(ref. EL12) Nos 7 à 12, les 6 prix **120F.**
(ref. EL13) abonnement en cours, 6 numéros, prix **150F.**
(ref. EL14) abonnement pour les numéros 19 à 24, prix **150F.**
(ref. EL16) **Manuel du tailleur et polisseur de verres d'optique** (Dodin), ISBN 2-86811-003-7, prix **150F.**
(ref. EL17) **Manuel dictionnaire du petit offset** (Dodin), ISBN 2-86811-005-3, prix **150F.**
(ref. EL18) **Aventures d'un individu au XXème siècle**, (Dodin), ISBN 2-86811-006-1, prix **120F.**
(ref. EM5) **Imprimante HP-IL.** Prix complète, prête à l'emploi **4130F.**
(ref. EM6) **Interface spécial** HP-IL/imprimante X-710. Prix interface seul **675F.**
(ref. EM7) **Lecteur d'Eprom** avec une cartouche (sans EPROM). Prix **1265F.**
(ref. EM8) **Cartouche** supplémentaire. Nous consulter.
(ref. EM9) Mise sur EPROM de vos programmes. Nous consulter.
(ref. EM10) **Lecteur de disquettes HP-IL.** Nous consulter.
(ref. EM11) **Interface vidéo HP-IL 80 colonnes/graphique.** Nous consulter.
Réclamez-nous notre catalogue général et notre catalogue logiciels gratuits HP-41, HP-75, HP-71, vous aurez tous les détails sur les produits ci-dessus.

TARIF HORS TAXES

PROGRAMMES. VENTE AUX PARTICULIERS

DECEMBRE 84

Prix en francs français, franco de port par voie de surface (ajouter **20F** pour expédition par avion), hors taxes, valables jusqu'au 31-12-84. Règlement à la commande (ne pas oublier d'inclure la TVA au taux de 18,6% pour les livraisons en France). Détails sur les produits dans le catalogue général PROGRAMMES gratuit. La notice du programme est vendue séparément pour vous permettre d'obtenir plus de détails par l'acquisition de cette notice, n'oubliez pas de l'inclure dans votre commande si vous ne la possédez-pas déja.

Recherche documentaire pour le compte des clients, l'heure **150F.**

PROGRAMMES POUR HP-41

(ref. EP1) **Jeu d'échec** sur **14** cartes magnétiques, prix **180F.** Notice **20F.** Remplacement des cartes : 20F en timbres.
(ref. EP2) **Calcul de poutres selon le CM66** (première partie) sur cassette, prix **650F**, sur cartes magnétiques prix **1000F.** Notice prix **60F**
(ref. EP3) **Analyse de données** sur cassette, prix **500F.** Notice prix **60F.**
(ref. EP4) **Calcul de béton armé selon BAEL.1** module HP-41, prix **3000F** avec notice.
(ref. EP5) **IMP41 Devis pour imprimeur**, sur cartes magnétiques, prix **1000F.** Notice **50F.**
(ref. EP6) **Programmes pour calculs du Bâtiment LOGABAT** nous consulter.
(ref. EP7) **Calcul de prix de revient dans l'industrie du plastique** nous consulter.
(ref. EP29) **Mailing 41CX** sur cartes prix **100F**, sur cassette prix **200F. Notice contre 10F en timbres.**

LES PROGRAMMES SUR EPROMS pour HP-41

(ref. EP8) **ASSEMBLEUR 3** par Richard Collet, Michael Thompson et John McGechie. Prix **300F.** Notice **20F** (en anglais).
(ref. EP9) **I. CODE** par Paul Cooper, prix **300F.** Notice **20F** (en anglais).

LES PROGRAMMES POUR HP-75

(ref. EP28) **FORTH POUR HP-75C** sur cassette, prix **1200F.** Manuel seul **50F**
(ref. EP10) **CHANGE 4** traitement d'un portefeuille de devises. Sur cassette **1000F**, notice **50F.**
(ref. EP11) **ATLAS** gestion d'une base de données de différents pays avec leur devise. Sur cassette avec notice, **500F**, notice seule contre 10F en timbres.
(ref. EP12) **Gestion de fichier** (nécessite Visicalc) sur cartes magnétiques, prix **500F.** Notice **50F**

(en anglais, en cours de traduction).
(ref. EP13) **MERGEVC** assemblage de feuilles Visicalc. Nous consulter.
(ref. EP14) **ED** Editeur plein écran et calculateur. Nous consulter.
(ref. EP15) **DECIDE** Aide à la décision. Nous consulter.
(ref. EP16) **STOCKMKT** stock market, jeu de bourse. Nous consulter.
(ref. EP17) **ACYDUCY** jeu de caractères. Nous consulter.
(ref. EP18) **RPN - RPNTV** Calculatrice polonaise inverse. Nous consulter.
(ref. EP19) **BASE DE DONNEE** Nous consulter.
(ref. EP20) **MASSLEX** gestion de fichiers sur cassette. Nous consulter.
(ref. EP21) **DSPLEX** Gestion d'écran. Nous consulter.
(ref. EP22) **UTIL3** PEEK, POKE et ADDR$. Sans notice. Gratuit contre une carte magnétique (pour le programme) et 10 timbres à 2,10 pour frais.
(ref. EP31) **POKER** sur cassette prix avec notice **300F**. Notice seule contre 10F en timbres.
(ref. EP33) **TAKE A MEMO** Traitement de texte pour HP905B ou thinkjet avec caractères spéciaux. Nous consulter.

LES PROGRAMMES POUR HP-71

(ref. EP23) **GRAPHIE** gestion de l'écran graphique, sur cassette **250F**. Notice **20F**.
(ref. EP24) **CALCUL DE ROULEMENTS A BILLES ET ROULEAUX** avec base de données, nous consulter.
(ref. EP25) **MEMORY** jeu. sur cartes **100F** sur cassette **200F**.
(ref. EP26) **Aide au diagnostic médical**. Sur cartes **300F**, sur cassette **350F**.
(ref. EP27) **Gestion financière d'un cabinet médical**. Nous consulter.
(ref. EP30) **Othello-Reversi** sur cassette **300F**
(ref. EP32) **ATLAS** nous consulter.

François LAFAIX
2 Rue de la résistance
44290 Guémene-Penfao.

pour le PPC-T

Le 15 décembre 1984

Ayant réalisé un compilateur FORTH fonctionnant sous environnement MS-DOS sur un VICTOR S1 (et sans doute adaptable aux appareils type IBM PC), je suis pret à prendre contact avec qui serait interessé par les sources, ou par le langage en soi.

Voici les principales caracteristiques de cette implantation:

-Au standard FORTH-79 (La compatibilité au standard FIG peut être assurée moyennant l'ajout de quelques mots - je ne dispose pas du texte de norme 84 !- mais je pense que dans l'ensemble , Tout programme FIG tourne, bien que certaines caractéristiques diffèrent: implantation mémoire differante afin de permettre une meilleure utilisation des Segments memoire du 8086, pas de véritable zone usager car l'environnement est mono-tache ....

-Contient un emulateur MS-DOS 1.25 (les commandes DIR, COPY, DEL, REN, A:, ... sont acceptées)

-Primitives de gestion des E/S etendues: sortie IMPRIMANTE, entree/sortie auxiliaire, Gestion de fichier à accés direct (Sans distinction entre la memoire de masse et les periphériques). Gestion d'horloge et de calendrier.

-Utilisation possible en pages de 64K (operations de transfert d'octets, de blocs de mev possible sur 1 Méga octet, la zone contenant le dictionnaire, la pile des données et les divers buffers ne pouvant exceder 64K par souci de compatibilité)

-Traitement de chaines inclus en version de base (Type STRING, comparaisons , copie entre deux variables avec justification à droite ... , une chaine peut etre considerée comme une constante ...)

-Editeur de BLOCKS pleine page, editeur pleine ligne en mode immediat ...

-Assembleur 8086 et editeur FIG disponibles sur BLOCKS

-Nombreuses autres extensions au standard disponibles dès la mise en route.

-Quelques Caractéristiques:
Vitesse d'execution: "10000 0 DO LOOP " execute en 0.7 s
Taille disponible entre HERE et S0 : plus de 59 Ko.
Pile de retour contenant 1024 mots (extensible)
Declenchement d'un programme possible par Ctrl C lors de l'execution d'un autre programme ....
Taille du code objet: moins de 9 Ko.
Messages d'erreur en français.
Ecrit grace aux logiciels suivants: MACRO86, LINK, on peut donc si on dispose du source l'adapter facilement à ses propres besoins où à un autre environnement.

En esperant que cela vous interesse, Veuillez accepter les salutations d'un amateur de FORTH et des HP-41 .

NOTE:

Si vous désirez aquerir le code objet et le manuel de reférence pour VICTOR S1, envoyez moi une disquette vierge 5 pouces 1/4 80 pistes 96 TPI simple ou double face et 150 Frs.
Pour avoir en plus le source sur disquette, rajoutez 200 Frs.
Enfin si vous disposez d'un autre appareil ne pouvant lire les disquette format victor, je peut vous faire parvenir mon source pour 150 Frs (Bon courage pour ceux qui auraient envie de le retranscrire manuellement : il y en a pour de nombreuses Heures : plus de 10 Ko de texte)

```
FORTH
FORTH 8086       Rev 2.01
↑Version MS-DOS§
Copyright 1984 (C) by F.Lafaix
Created JULY 84
Bytes free : 59873

OK
VLIST
FORTH DEFINITIONS VOCABULARY " ." L" R/W --> LIST EDIT LOAD UPDATE EMPTY-BUFFER
S SAVE-BUFFERS BLOCK BUFFER LIB Dà D! CLOSE REN COPY DIR DEL A: B: HMS! HMS DAT
E! DATE CSEG DSEG SSEG BP CODE L! LC! Là LCà LMOVE LCMOVE IF ELSE THEN DO I I'
J K LEAVE LOOP +LOOP BEGIN AGAIN UNTIL WHILE REPEAT BRANCH 0BRANCH COMPILE * */
 */MOD + - / /MOD M* M/ M/MOD S->D 0< 0= 0> 1+ 1- 2+ 2- < = > ABS D+ D< D= MAX
MIN MOD NEGATE U* U/MOD U< SAR SHL SHR ROL ROR AND NOT OR XOR ! C! *! +! -! /!
à Cà TOGGLE NNULL à= à> à< LSET CMOVE <CMOVE MOVE MCSWAP FILL CAPS -TRAILING MA
TCH CDEL CINS à à+ =" EXPECT EDLIN DEPTH DROP DUP ?DUP PICK OVER ROLL ROT R> Rà
 >R SWAP INKEYà KEY CON_SIZE LST_SIZE PRINT AUXIN AUXOUT EMIT CONOUT CR LCR SPA
CE SPACES AT LOCATE ( WORD HERE <# # #S HOLD SIGN #> . U. D. COUNT TYPE CONTYPE
 LTYPE ? DUMP ↑COMPILE§ CONSTANT VARIABLE STRING RECORD VLIST FORGET IMMEDIATE
SMUDGE ↑ § ALLOT , C, <BUILDS DOES> : ; CREATE DIGIT CONVERT DNEGATE DLITERAL L
ITERAL ASCII EXECUTE INTERPRET QUERY FIND ' QUIT EXIT ABORT CLEAR SYSTEM WIDTH
VOC_LINK USE TIB S0 STATE SP SCR R0 PREV PAD LIMIT >IN IP HLD FIRST FENCE ERR E
RROR ALTC DP CURRENT CSP CONTEXT BLK BASE BIN DECIMAL HEX B/SCR B/BUF
OK
çi dessus: exemple d'emploi du programme: contenu du dictionnaire de base
```

L'ORGANISATION DE PPC-TOULOUSE

Cette page sera publiée dans chaque numéro de MICRO-REVUE. Seules les indications répertoriées dans le dernier numéro du journal font foi. PPC-T étant une association sans but lucratif, elle ne fait pas de bénéfice, elle est administrée par des bénévoles et ses produits et services sont **réservés à ses membres.** La marge prise sur les produits vendus est minimale et destinée à couvrir les frais de fonctionnement. Une grande partie de nos produits vient des USA et leur prix est donc soumis aux variations du $. En cas de variation brusque du $ nous nous réservons le droit de demander un supplément.

ASSOCIATION PPC-T

Association à buts non lucratifs (loi de 1901), PPC-T accepte des adhérents dans toute la France et à l'étranger. La cotisation est de 50 F plus 150 F d'abonnement groupé pour MICRO-REVUE, pour un total de 200 F. DOM TOM et reste du monde par avion ajouter **50F**. Règlements par chèques et mandats postaux **à l'ordre de PPC-T** (Eurochèques acceptés, mandats postaux uniquement pour les DOM TOM). l'inscription prend effet au premier jour du bimestre qui **précède** la réception au club du bulletin d'adhésion (1er Janvier, 1er Mars...). En cas d'adhésion d'un abonné, l'adhésion est de 50 F et prends effet à la date de début de l'abonnement. Les adhérents s'engagent à apporter leur contribution au journal et au club dans la mesure de leurs moyens, même modestement.

RESPONSABLES DE PPC-T

Président: Jean-Daniel Dodin (T1) 77 rue du Cagire 31100 Toulouse (nouvelles adhésions, journaux de l'année en cours, édition du journal, réception des articles).
Trésorier: Jean-François Sibille (T15) Résidence du Pays d'Oc, rue des Genets 31500 Toulouse (renouvellement d'adhésions, problèmes financiers...).
Trésorier adjoint: Jean-Pierre Baudoin (T131) Lieu dit "Les Carabiniers" Castelnau d'Estretefond 31620 Fronton (fichier).
Secrétaire: Olivier Monachon (T13) 10 rue Louis Vignes 31200 Toulouse (librairie et coopérative)
Secrétaire adjoint: Gilles Barret (T22) 92 av de Muret 31300 Toulouse (photocopies).
Secrétaire adjoint: Jean-Yves Pasquier (T18) Apt. 10, 2 rue de Grèce 31000 Toulouse.
Programmathèque: Damien Debril (T33) 38 rue du 8 Mai 1945 59190 Hazebrouk (envoyez les cartes vierges, elles reviennent pleines).
Coordinateurs régionaux:
Bretagne: Olivier Pougeon 6 Bd Chanard 56170 Quiberon
Paris: Olivier Arbey (T164) 34 av de la République 94100 St Maur, Robert Schwartz (T178) 141 av Carnot 93140 Bondy, Eric Gengoux 8 rue de Furstenberg 75006 Paris, J.J. Dhenin 83 rue de Gergovie 75014 Paris, Jean Thiberge (T165) 104 Quai Louis Blériot 75016 Paris, Serge Vaudenay (T270) 62 av Ardouin B3 94420 Le Plessis Trévise, Stéphane Barizien (T149) 39 rue St Fargeau 75020 Paris
Lorraine: Dominique Talon (T311) 90100 Courcelle
Sud Est: Eric Angelini (T240) 14 av Marianna 06000 Nice
Besançon: E. Piérunek (T76) chez Mme Faivre 27 rue O. David 25000 Besançon.
Picardie: Bruno Tredez (T120), 16 chemin St Jean Allonville 80260 Villers Bocage.
Région Rhône: Yves Alajouanine (T144), 108 bis rue Hénon 69004 Lyon, et Franck Lebastard Apt 1085 15 Chemin des Plates 69120 Vaulx en Velin tel (7) 880 77 34.
BELGIQUE: Bernard Branquart (T372) 32 Thiarmont, 7190 Ecaussines Belgique.
Des coordinateurs sont recherchés pour toutes les autres régions. Contactez le coordinateur de votre région pour organiser avec lui des réunions, démonstrations,...Vous pouvez obtenir un budget du club sur présentation d'un projet. Nous pouvons vous imprimer vos convocations, des affiches,...Contactez vous-même les concessionnaires locaux, les journaux, les autres clubs. Contactez les mairies pour avoir des salles de réunions (montrez leur cette page du journal pour confirmer votre position, vous nous représentez officiellement dans votre région !)

Renseignements facultatifs

Sujets d'intéret, domaines préférentiels d'utilisation de votre ordinateur de poche :

Je dispose des matériels informatiques suivants (type et modèle, date d'achat) :

J'ai eu connaissance de PPC-T par :

N°PPCUS : N°CHHU :

Observations :

club de Programmation Personnelle
sur Calculatrice de Toulouse
77 rue du Cagire 31100 Toulouse France

Nouvel adhérent
Renouvellement
Changement d'adresse
(rayer la mention
inutile)

PPC-T N°_____

BULLETIN D'ADHESION

NOM _____
Prénom _____
ADRESSE _____

_____
Pays _____
Profession _____
Téléphone: 16( ): : : :

Ecrivez clairement en majuscules ; indiquez l'adresse à laquelle vous souhaitez recevoir les publications.

==========

Je demande à rejoindre PPC-T dans le seul but d'échanger des programmes et des idées sur la programmation. Mes seules obligations sont d'aider les nouveaux venus, d'encourager l'information sur les activités de PPC-T et d'offrir mon expérience aux autres adhérents. J'ai compris que l'adhésion à PPC-T entraine une participation active à l'échange d'informations.

Date signature

==========

paiement par: Chèque - CCP - mandat - espèces -
à l'ordre de PPC-T

Votre adhésion sera prise en compte à réception par PPC-T du présent bulletin correctement rempli.

==========

Remplir le verso, SVP, merci.

PPC

PROGRAMMATION SYNTHETIQUE, LCD, IMPRIMANTE, REGISTRES, MODULE, TOUCHES, OVER LAYS, MATRICES, MEMOIRE CONSTANTE, LECTEUR DE CODES A BARRES, LECTEUR DE CARTES, BASIC, RPN, FORTH, LANGAGE SPECIALISE, MICROCODE, ASSEMBLEUR, MATHEMATIQUES, SOUS PROGRAMME, NOMBRE ALEATOIRE, DRAPEAUX, LUCASIEWICZ, PILE, ALPHANUMERIQUE, PARASITES, RIDEAU, SWAP, NOTATION POLONAISE INVERSE, DUP, SHIFT, ECRAN, MOT, COMPILER, EXECUTION, TESTS, BOUCLES, DO LOOP, BEGIN UNTIL, IF ELSE THEN, : ;, BOUCLES CONDITIONNELLES, LABELS, TABLEAUX, TRACE, SST, BST, R/S

FORTH Interest Group

Edité par PPC-T — TITRE : MICRO-REVUE
77, rue du Cagire 31100 TOULOUSE

ISSN 0754-6335

Directeur de publication : J. D. Dodin