

# JPC

# PPC PC

JPC  
AURIL

1985  
\*NUMERO-23\*

Le numero 35FF

## A PROPOS DU CLUB

Pierre DAVID	1	EDITORIAL
Pierre DAVID	2	PPC PARIS SE REUNIT
Olivier POUGEON	3	COMPTE-RENDU DE LA REUNION BRETONNE
Christian BACQUET	5	NOUS EN AVONS
Etienne POUPEE	6	SOS (REPONSE)
	7	LA SCIP DANS LA COURSE

## DUR ET MOU

Pierre ANTOINE	8	ESSAI: LE MODULE AC POUR HP71
Philippe ROMASCANO	9	MODULE DE TRADUCTION HP41-HP71

## HP-41

Pierre LANGLOIS	12	ROUTINES COMPLEXES
Michel CLABOT	13	BIORYTHME ?
Jean-Jacques DHENIN	15	LE MODULE PANAME ET L'INTERFACE VIDEO

## HP-IL

Philippe TENAND	23	L'HPIL ET SES ELEMENTS
-----------------	----	------------------------

## HP-71

Pierre DAVID	30	LES BUFFERS DU HP71B
Laurent ISTRIA	32	INTERCEPTION DE POLL
Michel MARTINET	34	REPLACE#
Stephane BARIZIEN	36	FILE?
ISTRIA/MARTINET/DAVID	37	PEEK POKE PARTOUT
Pierre DAVID	37	BIP TOUT LE TEMPS
DAVID/MARTINET	38	ATTENTION !
Michel MARTINET	38	MANIPULATION DES CHAINES ALPHA
Michel MARTINET	47	CHAINAGE DE FICHIERS LEX
Pierre DAVID	49	CHOC EN RETOUR
Daniel JACOB	49	INTEGRATION NUMERIQUE DE GAUSS
Jean-Claude FOURES	49	UTILITAIRES
Philippe DAVASE	50	VISUALISATION DE LA RAM
Daniel JACOB	50	DOW, JDS, DATE ET HEURE
Serge VAUDENAY	50	LE DEBILOMETRE

# EDITORIAL

Chers membres,

Il y a un an était annoncé en Europe le HP71B. Nous avons été parmi les premiers à vous en faire part. Aujourd'hui, notre Journal montre l'activité du Club autour de cette machine. Nous sommes parmi les plus avancés en ce qui concerne l'assembleur, et nous devons ceci à la communication rapide des informations, par l'intermédiaire du Journal.

Mais ce succès ne doit pas faire oublier celui de la HP41. Pour la première fois depuis un an, nous avons été obligés d'imprimer un article en format réduit, cet article ne faisant pas moins de 15 pages...

Alors, n'hésitez plus: prenez votre plume et, vous aussi, participez à cette grande effervescence !

Et n'oubliez cependant pas le Spécial-Sicob (6 au 11 mai). Plus vous serez nombreux, plus grande sera la fête !

A bientôt  
Pierre DAVID

## PPC PARIS SE REUNIT UNE FOIS PAR MOIS

Comme vous le savez peut être déjà, PPC PARIS se réunit une fois par mois, en plein coeur de Paris. Amenez votre matériel, votre bonne volonté et vos idées ! Plus vous en apporterez, et plus vous en trouverez chez vos collègues de PPC. Ces réunions se déroulent de manière très libre, aucun ordre du jour, discussion ou autre n'étant imposé. Un membre du Bureau est toujours présent. Ainsi, si vous désirez remettre votre article tout frais au journal, si vous avez des suggestions à faire, si vous voulez vous procurer des anciens numéros de JPC, ce sera en principe toujours possible.

Si donc cela vous intéresse, n'hésitez plus un seul instant venez nous rejoindre tous les premiers samedi de chaque mois, au CENTRE DE JEUNESSE ET DE LOISIRS JEAN VERDIER  
11, rue de Lancry  
75010 PARIS

Et en montant au deuxième étage, vous entendrez des éclats de rire et des discussions passionnées vers la salle 215. Attention, toutefois, de venir entre 16 et 19 h.

Pour l'accès en métro, trois possibilités s'offrent à vous à savoir:

Métro Strasbourg St Denis: sortie Porte St Martin / Bld St Denis, coté pairs.

Métro République: sortie Bld St Martin, coté pairs.

Métro J. Bonsergent: Sortie Bld Magenta, coté impairs.

Ah, j'oubliais: JPC est distribué en avant-première lors de ces réunions ... A bon entendeur Salut !

Les dates des prochaines réunions sont: (16h-19h)

- Le Samedi 20 Avril 1985

- Le Samedi 4 Mai 1985

- Le Samedi 1 Juin 1985

A Bientôt...  
Pierre DAVID

# DIVERS

COMPTE RENDU DE LA REUNION DES 9 ET 10 FÉVRIER 85

Une fois encore le Club s'est réuni en Bretagne, cette fois-ci, au lieu de la région Rennaise, le point de chute a été l'extrémité de la Bretagne, près de Quimper. Heureusement, la distance n'a rebuté personne puisque (et je tiens à les remercier) les participants ont afflué de toute la France: de Lille (Vincent DELORME), de Lyon (Franck LEBASTARD), de Toulouse (Jean-Daniel DODIN), certains ayant même interrompu leurs vacances (Michel MARTINET), et je ne cite pas tous ceux qui sont venus de Paris et de sa banlieue.

Étaient finalement présents: BLANCHET Gilbert (T297) de Plerin, BORNES Jean-Claude (PPC 9250 T 175) de Lisses, CAHIERRE Patrick (T340) de Guer, DELOGE Jean-Claude (T610) de Entrammes, DELORME Vincent (T350) de Lille, DHENIN Jean-Jacques (P177 T212) de Paris, DODIN Jean-Daniel (P8 T1 FFC7226) de Toulouse, JOUVE Grégoire (T400) de Rennes, LE BASTARD Franck (P43 T35) de Lyon, MARTINET Michel (P12 CHHU616) de Paris, NICOLAS Georges-Noël (P147 T438) de Crozon, NICOLAS François (T653), POUPEE Etienne (P197) et moi-même (P64 T62) de Quiberon.

Pour la première fois dans une réunion de Bretagne, nous avons réussi à avoir des membres des bureaux de nos deux Clubs français, en la personne de Michel MARTINET, secrétaire du Club de Paris, et Jean-Daniel DODIN, président du Club de Toulouse, qui ont eu tous deux la gentillesse de faire un long voyage pour venir nous rejoindre.

Le weekend a été ponctué par un certain nombre

d'exposés:

- J.D. DODIN a fait le point sur un certain nombre de projets qui se développent dans son Club:

- Au niveau des modules nous avons appris notamment que les Américains seraient intéressés par le module PANAME, comme quoi, il suffit qu'un produit soit bien fait pour qu'il ait du succès; nous avons pu consulter la liste, mise à jour des fonctions du CCD-ROM, qui promet, lui aussi, d'être un module très intéressant. Enfin nous avons appris toutes les difficultés que posent les importations de modules et qui expliquent les délais parfois nécessaires à leur obtention.

- Nous avons pu voir le catalogue IND 41 qui vient de sortir aux éditions du Cagire. Cela semble être la publication la plus complète sur les programmes édités pour la 41, tant sur le plan du contenu (puisque seuls PPC-TN et PRISMA (N.D.L.R. : et JPC) n'y sont pas recensés), que sur le plan de la forme (des fiches du style de celles des bibliothèques, très complètes).

- Forth: le module Forth de Serge VAUDENAY en est à la phase du debugging, aussi le Club est-il à la recherche de gens connaissant le Forth pour cette tâche. Par ailleurs, vu le nombre de commandes, ce module serait destiné à sortir sur EPROM, à moins que tous ceux qui sont intéressés ne s'éveillent pour pousser dans le sens d'un module HP.

- Parmi les rumeurs: un livre sur le microcode sortirait aux U.S.A., Matra sortirait une Alice Forth, une traduction française du manuel de la ZENROM serait en cours.

- Par ailleurs, le président nous a parlé d'une interface graphique vidéo et des lecteurs double

disquettes qui existent pour la boucle HP-IL. Hélas, toujours en raison des problèmes d'importation, nous n'avons pu voir ces matériels qui n'étaient pas arrivés en France au jour de la réunion.

- Un autre sujet prometteur a été abordé par Jean-Claude BORNES qui a réalisé une extension intéressante pour la 41: il a réalisé par le biais d'un convertisseur parallèle une interface HP-IL <> MINITEL.

- On peut ainsi se servir du minitel comme d'une interface vidéo aux normes minitel, soit 25 lignes de 40 colonnes (mieux que HP).

- Si on possède un minitel "retournable", il est possible d'échanger des programmes entre 2 41 par téléphone, en bref un modem HP-IL bon marché.

- On peut utiliser la 41 pour stocker des pages minitel.

- Quand on sait que la norme minitel est, au voltage près la même que la norme RS232C, on s'aperçoit que l'on peut disposer ainsi de la RS232C à un prix défiant toute concurrence.

- Dernier renseignement, le montage reviendrait de 500 à 700 Frs. si on y ajoute le millier de Francs du convertisseur parallèle, on arrive à moins de 2000 Frs. pour une interface vidéo et sa console (pour une fois que les PTT donnent quelque chose !).

- Un troisième sujet a été abordé par Jean-Jacques DHENIN et Michel MARTINET: le HP71B.

- Nous avons pu ainsi découvrir les possibilités du 71, et à ce sujet je pense que nous avons tous été étonnés. En effet, si le 75 s'est montré comme étant une machine qui n'avait comme rapport avec la 41 que l'HP-IL, nous avons pu voir que la 71, malgré des calculs en notation algébrique, était née dans la même philosophie que la 41: modularité, langage de programmation complet, petite taille, souplesse d'utilisation, possibilités de calcul importantes, microprocesseur complet, etc...

Mais avec malgré tout un plus non négligeable: dès sa sortie, HP a mis à la disposition de tous la documentation nécessaire sur la 71.

En plus il existe déjà un Forth pour le 71 qui semble très réussi (en plus du Basic résident).

- Dans un autre ordre d'idées, nous avons pu voir en fonctionnement une Thinkjet et une unité de diskette HP-IL, 2 périphériques très efficaces,

mais cependant coûteux.

- Vincent DELORME nous a montré un programme de son cru sur HP75: un cross assembleur 6800 avec comme but la possibilité d'interfacer le 6800 avec la boucle HP-IL.

- Etienne POUPEE quant à lui nous a montré sa production: ainsi, il intègre dans une même valise un calculeur (41 ou 71), une imprimante table traçante CANON, une boucle HP-IL, une unité de cassette HP-IL, une interface RS232C, et un modem acoustique. Bref l'ensemble parfait pour la personne qui voyage et qui veut avoir son système disponible et sous la main. Il nous a montré aussi le lecteur d'EPR0M que commercialise la SCIP, et qui sera sans doute utilisé pour supporter la ROM FORTH.

- Le dernier sujet abordé a été la possibilité de raccord entre la 41 et un micro ordinateur via la RS232C. Actuellement, les programmes sont développés sur un SANYO 550, à mon avis c'est le nec plus ultra en matière de périphérique (et là, je ne plaisante qu'à moitié) en effet, en regardant le SANYO afficher les programmes de la 41, les stocker sur ses disquettes, servir d'interface vidéo couleur graphique haute définition, on se demande lequel des deux commande. Et la réponse est encore plus étonnante puisque, en fait, les deux dialoguent en permanence, permettant ainsi une souplesse d'utilisation assez étonnante, et ce pour un prix de revient de l'ordre des périphériques HP, qui n'offrent hélas que des possibilités inférieures.

Voilà, j'espère que dans ce compte rendu je n'oublie personne, que ceux qui se sont déplacés pour cette réunion n'ont pas regretté le voyage et qu'ils y ont trouvé ce qu'ils espéraient. Je tiens à remercier Georges-Noël NICOLAS, qui, grâce à sa présence sur place avant la réunion, et à tout le courrier qu'il a pu faire, a permis que ce rassemblement ait eu lieu dans de bonnes conditions. Je pense que la prochaine réunion aura lieu à l'automne prochain, de toute façon vous en serez avisés dans le journal. En attendant ce jour, Heureuse Programmation à tous !

(P64 T62) Olivier POUGEON

## NOUS EN AVONS

Sont en vente au Club:

- Cartes magnétiques pour HP41 au prix de 150FF + 3,10FF de port les 50 cartes;
- Cartes magnétiques pour HP71B au prix de 71FF + 3,10FF de port les 10 cartes;
- Eeproms (2716 & 2732) vierges au prix de 60FF + 3,10FF de port pièce. Pour les Eeproms programmées, nous consulter;
- Module TIME (82182A) au prix de 400FF + 6,50FF de port. Livré avec manuel en français;
- Module QUADRAM (82170A) au prix de 400FF + 6,50FF de port;
- Module HPIL (82160A) au prix de 970FF + 6,50FF de port. Livré avec manuel en français;
- Pour les commandes des ZENROM, CCDROM et ROM PANAME nous envoyer 500FF minimum d'arrhes;
- VASM (listing des 3 premières pages de la 41) au prix de 170FF + 13,50FF de port;
- Le manuel de service de la 41 au prix de 50FF + 6,50FF de port;
- Le manuel technique du convertisseur au prix de 20FF + 3,20FF de port;
- Les anciens numéros au prix de:
  - No 1 à 3 au prix de 15FF + 3,20FF de port par numéro,
  - No 4 à 10 au prix de 20FF + 3,20FF de port par numéro,
  - No 11 à 17 au prix de 25FF + 6,50FF de port par numéro,
  - à partir du No 18 au prix de 35FF + 6,50FF de port par numéro.

Les ports sont les tarifs PTT actuels en non urgent.

## ACHATS GROUPEES

Nous assurons des commandes groupées pour ceux qui désirent profiter de notre réduction "spécial Club", soit 25% par rapport au prix HF. Pour ceux qui le désirent, nous assurons l'envoi postal en urgent ou en non urgent. Des achats concernent aussi bien des modules que des calculatrices et autres.

NOTA: ces deux rubriques "nous en avons" et "achats groupés", sont des services "Club" que nous rendons exclusivement aux membres effectivement inscrits au club et possédant leur carte de membre PPC-PARIS.

## ACCORD AVEC HP

Nous venons de conclure un accord avec Hewlett-Packard. C'est un geste d'ouverture vers notre Club, et nous ne pouvons que nous en féliciter.

Désormais, il est possible pour vous, adhérent du Club, d'acheter directement le matériel auprès d'HP, en bénéficiant d'une réduction de 53,7 % par rapport au prix catalogue.

Il serait dommage de ne pas en profiter...

S O S (réponse)

BACQUET Christian  
1, Tertre St Ewan  
28000 CHARTRES

Bonjour Amis (es)

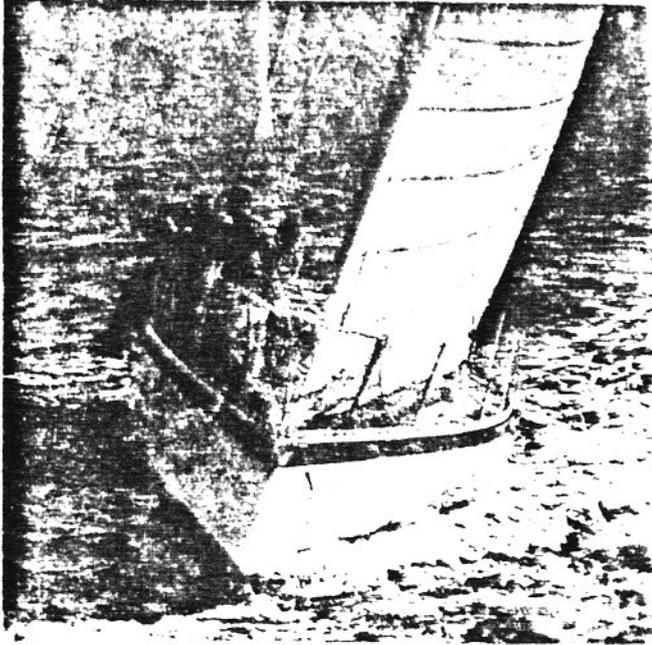
Je prends mon style pour répondre à 2 SOS de Février.  
Le premier SOS concerne une Interface Analogique 4 voies. Voici ce que je propose:  
Convertisseur analogique/digital 8 voies multiplexées  
Tension 0 à 5V avec le moins commun  
Alimentation intégrée 12V/220V  
C'est un A/D 8 bits,  
Et en plus un programme de configuration du GPIB + Connecteur.  
Le tout 1000 FF TTC, garantie 1 an, pièces et Main d'Œuvre.  
Nécessite un GPIB HPB2166A et un module I/O.  
Etudie toutes propositions 8, 12, 16 bits, 8, 16, 32 voies. Me consulter.

Le deuxième SOS de M. DUBOIF concerne le HPGL (langage graphique HP).  
Me contacter au (37) 34 24 25 p 364.

Amicalement,  
Christian BACQUET

### LES PRODUITS

. Hewlett Packard annonce dans la série 200 des ordinateurs HP 9000 le modèle 217, poste de travail pour l'ingénierie, avec 512 ko jusqu'à 4 Mo, moniteur à tube phosphore vert de 14" d'une résolution de 512 x 390 au prix de 66 822 FF ht et le modèle 237, poste de travail graphique, avec 512 ko jusqu'à 7 Mo, moniteur 17" à affichage mappé binaire de 1 024 x 768 et souris au prix de 148 821 FF ht. Le prix de la souris en option sur le 217 est de 1 620 FF ht. La compagnie annonce également l'analyseur de protocole portable HP 4953A qui permet de surveiller la transmission des données ou simuler des composants de réseaux. Peut être totalement télécommandé et déclenche jusqu'à 72 kbps en duplex intégral, vitesse 50 bps à 72 kbps.



LA S.C.I.P. DANS LA COURSE

\*\*\*\*\*

Du 29 Mars au 5 Avril va se dérouler à la ROCHELLE, la course de l'E.D.H.E.C., rassemblant plus de 300 bateaux, en faisant ainsi la plus grande course européenne par son nombre de participants. Le principe de cette épreuve est de faire concourir différentes universités, la S.C.I.P. y sera présente avec le bateau de l'université de PARIS I-TOLBIAC.

En effet Marc LE GOFFIC, le navigateur de ce bateau va utiliser durant toute la course "LE NAVIPROM" mis au point par la S.C.I.P. sous la direction de Messieurs Etienne POUPEE et de Jacques HAMMELIN.

"Ce petit boîtier miracle" comme l'appelle le navigateur se branche tout simplement sur une H.P. 41 D.V. et fait de cette machine un véritable petit ordinateur de bord. Tous les problèmes de navigations côtières et hauturières sont résolus en un "clin d'oeil".

L'originalité de ce logiciel consiste en sa rapidité exceptionnelle et dans le fait qu'il est extensible très rapidement et très facilement. C'est ainsi que des programmes de tactiques - courses sont en train d'être mis au point et vont être très prochainement introduit dans le boîtier.

En effet, après l'E.D.H.E.C., la S.C.I.P. va se lancer dans le Tour de France à la Voile (T.F.V.) où elle sera partenaire du bateau "Ville de Paris - E.S.C.P.". Christian DUMARD le navigateur a été particulièrement séduit par "LE NAVIPROM" qu'il compte utiliser (c'est officiel), pour la T.F.V. 1985.

LE NAVIPROM soyons en sûr, a un bel avenir devant lui, autant en course qu'en croisière. Souhaitons lui bonne chance.

N.B. : Pour tous renseignements complémentaires, contacter  
S.C.I.P., 10-12, rue Deltéral - 93310 LE PRE ST-GERVAIS.  
TEL. : 840-58-29

# DUR ET MOU

Pierre ANTOINE

LE MODULE AC POUR HP71  
Banc d'essais du module  
Analyse de Circuit

Philippe ROMASCANO

MODULE DE TRADUCTION HP41-HP71  
Plus de renseignements sur  
ce module. Il promet !

## LE MODULE AC POUR HP71

Le module AC Steady State Circuit Analysis abrite  
5 fichiers:

Deux fichiers LEX:

- Circuit contient le numéro de version du module.
- KEYWAIT contient la fonction KEYWAIT\$ dont l'effet est similaire à KEY\$, sauf que KEYWAIT\$ attend la pression d'une touche (voir JPC No 20).

Trois fichiers BASIC:

- CNAP est le programme principal. Il appelle le sous programme CNAP71 afin de préserver l'environnement du HP71.
- CNAPDOUT contient le sous programme DOUT, appelé par CNAP, et qui a pour tâche de formater toutes les sorties vers l'imprimante ou l'affichage.
- CNAPHEAD contient le sous programme HEAD qui génère les en-têtes des sorties imprimées ou affichées.

NAME	S	TYPE	LEN	DATE	TIME	PORT
Circuit		LEX	43	09/30/83	12:00	4
KEYWAIT		LEX	55	09/30/83	12:00	4
CNAP		BASIC	9887	09/30/83	12:00	4
CNAPDOUT		BASIC	946	09/30/83	12:00	4
CNAPHEAD		BASIC	261	09/30/83	12:00	4

CNAP permet une analyse rapide de n'importe quel  
circuit en régime AC permanent.

Il permet:

- la simulation de filtres actifs,
- l'analyse de filtres LC contenant des transformateurs et des inductances couplées,
- l'analyse des circuits amplificateurs à transistors,
- l'analyse des amplificateurs opérationnels,
- et l'analyse des amplificateurs haute fréquence.

Un HP71 peut résoudre un circuit comportant un  
maximum de 17 noeuds et 51 branches. L'ajout de  
modules mémoire permet d'étendre le circuit à 24

noeuds et 72 branches.

Les éléments reconnus par le programme sont les résistances, les inductances, les capacités, les Vccs et les éléments de transition. Par conséquent, si le circuit comporte des transistors, des ampli-op, des transformateurs, etc, il faudra déterminer les éléments équivalents. Le manuel décrit les opérations permettant d'arriver à un circuit pouvant être accepté par CNAP.

CNAP se présente sous la forme de menus. Il suffit de répondre aux questions ou de choisir parmi les options proposées.

#### ENTREE DU CIRCUIT:

Il faut au préalable numéroter tous les noeuds du circuit. Ensuite, pour entrer un élément du circuit, il suffit d'indiquer son emplacement (les noeuds de connexion) et les valeurs caractéristiques de l'élément.

Il est possible de changer, détruire ou ajouter des branches au circuit: celui-ci peut également être stocké dans un fichier DATA pour un rappel ultérieur.

#### SORTIE DES RESULTATS:

CNAP demande le noeud de sortie. L'option "group delay" consiste à analyser le circuit à des fréquences justes supérieures et inférieures à la fréquence d'analyse en cours. Le taux de modification de la phase suivant la fréquence est le "group delay". Cette option double le temps de calcul. Cette valeur est obtenue par différentielle numérique des valeurs des deux phases obtenues.

CNAP demande ensuite le type d'incrément (linéaire ou logarithmique), l'intervalle d'étude et la valeur de l'incrément (nombre de points par décade pour un balayage logarithmique).

La dernière option est la sortie des résultats vers l'imprimante.

Les sorties sont formatées par CNAPOUT: l'en-tête est contenue par CNAPHEAD. L'utilisateur a la possibilité de définir son propre en-tête et son propre formatage des

résultats (sorties quelconques vers l'imprimante, stockage des résultats sur cassette ou diskette, miniplotter ou table traçante graphiques).

La routine CNAPOUT donne la fréquence, le gain en dB, la phase et, éventuellement, le "group delay" (en sec.).

Les annexes du manuel décrivent la méthode de calcul de CNAP. elles donnent la liste des fichiers et des variables utilisés par les programmes. Les Voltage-controlled current sources et les éléments de transition, ainsi que le format sous lequel est stocké le circuit sont expliqués.

CNAP est assez satisfaisant. Il est cependant dommage qu'il faille transformer les circuits comportant certains éléments classiques en électronique, mais non reconnus par le programme, surtout que ce n'est souvent qu'une question de calcul.

Il faut également noter que tout est en anglais, mais le manuel comporte quelques exemples bien choisis, à défaut d'être nombreux.

Pierre ANTOINE  
P61 T384

---

#### MODULE DE TRADUCTION HP41-HP71

Ce nouveau module permet d'exécuter la plupart des programmes écrits pour HP41 sur un HP71B.

Quelques questions à propos de ce module:

Qu'est ce que ce module ?

Le "Translator Pac" est un module d'application pour le HP71B qui émule une HP41 et permet la traduction des programmes de ce calculateur.

Comment entrer les programmes dans le HP71B ?

Vous pouvez soit entrer le programme au clavier du 71B, soit transférer le programme à travers la boucle HPIL.

Peut-on écrire des programmes HP41 ?

Oui: ce module contient un éditeur puissant qui vous permet d'écrire des programmes HP41 à partir du clavier du HP71B.

Comment vont fonctionner les programmes traduits ?

Le module est construit de manière à produire les mêmes séquences d'affichage qu'une HP41. Les résultats peuvent quelque peu varier, la précision mathématique du HP71B étant plus grande que celle de la HP41. Si différence il y a, le résultat ne pourra qu'être plus juste ! Les messages d'erreur sont également plus explicites.

Faut-il modifier les programmes HP41 ?

Non, pour autant qu'ils contiennent les fonctions reconnues par le module de traduction (la plupart des programmes resteront inchangés).

Quelles sont les fonctions reconnues ?

Toutes les fonctions programmables des HP41C/CV, plus quelques fonctions de manipulations de données de la HP41CX et du module X Fonctions. Le module contient également toutes les fonctions non-graphiques de l'imprimante du module HPIL.

Quelles sont les fonctions HP41CX non reconnues ?

Les fonctions temps, alarme, chronomètre et mémoire étendue.

Quelle place mémoire utilisent les programmes ?

Les programmes traduits prennent 2,5 fois plus de place dans le HP71B.

Quelle est la vitesse d'exécution ?

Les programmes, suivant leur contenu, tournent de 4 à 8 fois plus vite. Les programmes contenant des chaînes alpha et des branchements tourneront plus lentement que des applications purement mathématiques.

Le module réplique-t-il le clavier de la HP41 ?

Non. Le module donne au HP71 un mode HP41, mais pas toujours une réplique touche par touche des opérations. Des différences ont été introduites pour tirer profit de la puissance et de la flexibilité du HP71. Un utilisateur HP41 n'aura aucune difficulté à passer d'une machine à l'autre.

Quelles sont les nouvelles fonctions ?

Jusqu'à 10000 registres de données directement accessibles, limités par la taille mémoire du HP71B.

Emploi de la pile de commande.

Exécution des fonctions par leur nom, sans passer par une touche XEQ alpha ... alpha.

L'écran montre l'état de la pile après chaque opération.

Que signifie traduction ?

Les programmes HP41 sont traduits en FORTH !

Faudra-t-il apprendre le FORTH ?

Non: la traduction est automatique.

Est-ce que le manuel permet d'apprendre le FORTH ?

Non: il existe bon nombre de manuels dans le commerce.

Pourra-t-on programmer en FORTH ?

Oui: le module contient le système FORTH et un dictionnaire de plus de 300 mots.

Philippe ROMASCAND

# HP-41

Pierre LANGLOIS

ROUTINES COMPLEXES

Affrontez vos complexes  
la tete haute !

Michel CLABOT

BIORYTHME ?

Calcul du biorythme.

Jean-Jacques DHENIN

LE MODULE PANAME ET L'INTERFACE VIDEO

ou comment rediger des programmes  
plus courts et plus lisibles...

## ROUTINES COMPLEXES

Après "F-" et "M<>", voici "C\*A", qui simule, lui aussi, une pile opérationnelle simplifiée - destinée, cette fois, aux complexes. Rappelons ce qu'est une pile simplifiée:

Elle comprend deux registres de calcul: CX et CY, plus un registre de stockage: CST, destiné à sauver les résultats intermédiaires dans une série de longs calculs.

Ici, le relativement faible encombrement des nombres complexes a permis de caser CX et CY dans la pile opérationnelle (la vraie, celle dont vous avez l'habitude.) de la façon suivante:  $CX=X+iY$ ;  $CY=Z+iT$ . Enfin:  $CST=(01)+i(02)$ .

Il faut donc faire un peu attention lors des manipulations: toujours introduire la partie imaginaire (ENTER) avant la partie réelle. Les habitués de la fonction R-P ne seront pas dépaysés. D'un autre côté, la lecture est plus logique: à la fin de chaque opération, c'est la partie réelle qui est affichée, X<>Y permettant d'avoir la partie imaginaire.

Je n'ai pas programmé le calcul du module  $r$  du complexe CX pour une raison évidente: il est d'origine sur nos chères HP41, directement accessible au clavier avec la fonction R-P qui, en prime, nous donne l'angle polaire correspondant [théta] en X, ils seront interprétés par le programme non pas comme  $r e^{i\theta}$ , mais comme  $r+i\theta$ , ce qui est une source d'erreur non négligeable.

Pour des raisons tout aussi évidentes, le programme ne comporte pas de module pour calculer le complexe conjugué de CX; vous pouvez toujours le rajouter si cela vous paraît indispensable.

Le programme est, bien entendu, modulaire :

"C+"; "C-"; "C\*"; "C/" parlent d'eux-mêmes: ce sont les quatre opérations appliquées aux complexes. CX contient le résultat et CY reste intact.

"C^A" et "C/A" permettent respectivement l'élevation de CX à la puissance du réel A (Si  $A=-1$ , vous obtenez  $1/CX$ , et si  $A=1/n$ , une racine nième de CX) et la multiplication de CX par A. La méthode utilisée permet de ne pas abîmer CY au cours de l'opération: vous appelez le module, qui stocke temporairement X en M et vous demande

alors A. Le procédé a deux inconvénients: il dévaste le registre alpha et rend le module difficilement appelable à partir d'un autre programme. J'ai préféré favoriser la facilité d'emploi direct. Les éventuelles modifications que vous pourriez souhaiter ne sont pas très difficiles à réaliser. Rappelez-vous cependant que l'intérêt principal de la méthode utilisée est de préserver le contenu de CY.

Enfin, "CST" sauve CX en CST, "CRC" rappelle CST en faisant monter CX en CY.

Ce programme occupe 130 octets, il tient donc à l'aise sur une carte magnétique, sans "CST" et "CRC", il tient même sur une seule piste. Vous pouvez donc largement ajouter vos propres modules. Je pense personnellement à la résolution des équations du second degré à coefficients complexes.

N'hésitez plus ! Vous pouvez désormais affronter vos complexes la tête haute.

Amicalement

Pierre Langlois

```

01+LBL "C"
02 R-P
03 "A?"
04 RVIEW
05 STO M
06 RDN
07 STOP
08 *
09 X<> M
10 LASTX
11 Y↑X
12 RCL M
13 X<>Y
14 P-R
15 RTN

```

```

16+LBL "C*A"
17 "A?"

```

18 RVIEW  
19 STO M  
20 RDN  
21 STOP  
22 ST\* M  
23 \*  
24 RCL M  
25 RTN

26\*LBL "C-"  
27 CHS  
28 X<>Y  
29 CHS  
30 X<>Y

31\*LBL "C+ "  
32 R↑  
33 ST+ Z  
34 R↑  
35 ST+ Z  
36 RDN  
37 RDN  
38 RTN

39\*LBL "C/"  
40 R-P  
41 1/X  
42 X<>Y  
43 CHS  
44 X<>Y  
45 P-R

46\*LBL "C\*"  
47 R-P  
48 R↑  
49 R↑  
50 R-P  
51 ST\* Z  
52 X<>Y  
53 ST+ T  
54 X<>Y  
55 P-R  
56 RDN  
57 RDN  
58 P-R  
59 RTN

60\*LBL "CST"  
61 STO 01  
62 X<>Y

63 STO 02  
64 X<>Y  
65 RTN

66\*LBL "CRC"  
67 RCL 02  
68 RCL 01  
69 RTN  
70 END

-----  
Vous avez dit "BIORHYME" ?

Rappelons à ceux perdus "au fond de la 41" que, suivant des études sérieuses faites par des Docteurs, notre corps subirait des influences suivant certains cycles:

Physique de 23 jours (Dr W. FLIESS)  
Emotionnel de 28 jours (Dr H.SWOBODA)  
Intellectuel de 33 jours (Pr A. TELTSCHER)

Ces cycles, dont le premier jour se situerait le jour de notre naissance, seraient tantôt positifs (période faste), tantôt négatifs (période néfaste).

Pour juger de sa forme, voici donc "BI", programme court de 72 lignes, 24 registres, permettant en fonction de sa date de naissance de connaître sa période faste ou néfaste. Pour nos amis puristes, les lignes suivantes peuvent à loisirs être modifiées:

9: D1=/Date de naissance  
14: D2=/Date de base du calcul  
26: JRS/JOURS  
29: PHY/PHYSIQUE  
32: EMO/EMOTIONNEL  
35: INT/INTELLECTUEL  
40: MOY/MOYENNE

Lancez le programme par R/S, entrez la date au format DD,MMYYYY (19,021985) pour "D1=" et "D2=",

le reste s'affichant seul.

Amicalement

Michel CLABOT

-----  
01\*LBL "BI"  
02 CF 29  
03 DATE  
04 STO 02

05\*LBL 00

06 FIX 6

07 CLST

08 STO 07

09 "D1= "

10 RCL 01

11 ARCL X

12 PROMPT

13 STO 01

14 "D2= "

15 RCL 02

16 ARCL X

17 PROMPT

18 STO 02

19 RCL 01

20 X<>Y

21 DDAYS

22 STO 00

23 FIX 0

24 CLA

25 ARCL X

26 "┌ JRS "

27 XEQ 02

28 23

29 "PHY"

30 XEQ 01

31 28

32 "EMO"

33 XEQ 01

34 33

35 "INT"

36 XEQ 01

37 RCL 07

38 3

39 /

40 "MDY"

41 XEQ 03

42 RTN

43 GTD 00

44\*LBL 03

45 E2

46 #

47 INT

48 X<0?

49 SF 01

50 "┌ "

51 ARCL X

52 FC? 01

53 "┌ "

54 "┌-X ## "

55\*LBL 02

56 -2

57 ARDT

58 AVIEW

59 FC? 01

60 BEEP

61 FS?C 01

62 TONE c

63 RTN

64\*LBL 01

65 1/X

66 RCL 00

67 #

68 360

69 #

70 SIN

71 ST+ 07

72 XEQ 03

73 END

LE MODULE PARAMÈRE ET L'INTERFACE VIDEO

Le "modèle" video pour le HP-41 est une nécessité pour peu que l'utilisateur ait à réaliser des programmes assez difficiles et longs, ce qui est souvent le cas pour des applications professionnelles.

- Il y a au moins 2 axes de travail: ou le moniteur présente un intérêt:
  - Pour une visualisation du déroulement du programme et l'affichage des résultats à un groupe de personnes notamment en milieu scolaire, dans les lieux de passage tels que les magasins etc....
  - Pour la mise au point des programmes.

Quelle que soit la situation, la nécessité d'utiliser les séquences d'échappement comprenant un caractère de code 27 et un ou plusieurs caractères placés dans le registre ALPHA au moyen de l'instruction XTION présente plusieurs inconvénients:

- Un nombre important de pressions de touches
- Une occupation importante de l'espace mémoire
- Une mémorisation fastidieuse des codes
- La destruction systématique du registre ALPHA
- Le recours inévitable à l'utilisation de chaînes synthétiques totalement illisibles dans le listing des programmes et quelquefois l'impossibilité d'imprimer correctement les listings.

Avec le module PARAMÈRE tous ces inconvénients sont surmontés. À titre d'exemple nous présentons deux versions d'un programme utilisant le moniteur: la première version est due à R. NELSON et occupe 29 octets, la seconde effectuée un affichage semblable sur la video et n'occupe que 11 octets. Vous remarquerez la différence de lisibilité des 2 versions et que l'utilisation du module PARAMÈRE rend la programmation plus aisée puisque le programmeur n'a plus à rechercher dans une table de correspondance les codes nécessaires à la construction des chaînes synthétiques.

La routine LBL "1" affiche sur l'écran:

**HELLO**

Puis :

WELCOME TO THE WORLD OF  
HP-41 VIDEO  
AND THE HP103 VIDEO INTERFACE

La routine LBL "2" affiche:

SCIEBA FEATURES  
VIDEO OR TV CHC OUTPUTS  
HP-IL CONTROLLED  
INVERSES CHARACTERS  
TRACE MODE  
190 DISPLAYABLE CHARACTERS  
16 31 LINES OF 52 CHARACTERS  
CURSOR THIS OR THIS --

La routine LBL "3" affiche:

190 DISPLAYABLE CHARACTERS  
1"##%\*(+\*+-.0123456789:;= ?@  
ABCDEFGHIJKLMNQRSTUWXYZ[\]^\_`  
abcde fghijklmnopqrstuvwxy z{}  
1"##%\*(+\*+-.0123456789:;= ?@  
ABCDEFGHIJKLMNQRSTUWXYZ[\]^\_`  
abcde fghijklmnopqrstuvwxy z{}

La routine LBL "4" affiche:

HOME XXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXX SROLL UP XXXXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

La routine LBL "5" affiche:

THE TIME IS:  
07:45:05 PM  
12/01/1985

La routine LBL "6" affiche:

0	1	2	3
123456789012345678901234567890			
2XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
3XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
4XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
5XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
6XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
7XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
8XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
9XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
10XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
11XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
12XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
13XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
14XXXXXXXXXXXXXXXXXXXXXXXXXXXXX			

Voici la première version. Dans certains cas les instructions sont regroupées sur une même ligne pour gagner de la place. Le listing est n'est pas exactement celui d'une imprimerie (en fait il n'est pas listable), sous chaque ligne synthétique se trouvent les codes décimaux nécessaires à l'écriture.

01=LBL "0"  
02 " D, " "  
10,0,68,44,2,136,0  
03 RCL I (M)  
04 STO D  
05 XEQ "1"  
06 XEQ 14  
07 XEQ "2"  
08 XEQ 14  
09 XEQ "3"  
10 XEQ 14  
11 XEQ "4"  
12 XEQ 14  
13 XEQ "5"  
14 XEQ "6"  
15 GTO "0"

Initialise le calculateur: cette anaine prend place dans le registre ALBHA puis dans le registre X pour être chargée dans le registre des indicateurs binaires. Comparez avec "CHFLAG" du module PANAME "1" affiche "HELLO" sur la video "2" énumère les différentes possibilités de l'interface. "3" liste les 190 caractères dont dispose l'interface. "4" démonstration du 'scrolling'. "5" affichage de résultats pour un groupe. "6" tabulation et graphisme.

JPC

21

23

2

10

10

16=LBL 14 SF 17 PSE PSE PSE RTN

23=LBL "K"  
24 "D"  
127,68  
25 GTO 07  
26=LBL "L"  
27 "A"  
127,651  
28 GTO 07  
29=LBL "M"  
30 "H"  
127,721  
31 GTO 07  
32=LBL "N"  
33 "B"  
127,661  
34 GTO 07  
35=LBL "O"  
36 "C"  
127,671  
37 GTO 07  
38=LBL "P"  
39 F97 05  
40 "K"  
127,601  
41 F97C 05  
42 GTO 07  
43 SF 05  
44 "B"  
127,621  
45 GTO 07

Cursor left : curseur vers la gauche. Equivalant à CSRL du module PANAME Affecté à la touche 14 (D)  
Cursor UP : curseur vers le haut. CSRU dans le module PANAME Affecté à la touche -24 (M)  
Home cursor : place le curseur à la position 0,0 et ne provoque pas de 'scrolling' de la page visible. = "HOME" (PANAME) 11 (A).  
Cursor down : curseur vers le bas. = CSRD 24 (I)  
Cursor right : curseur vers la droite. = CSRR 15 (E)  
Cursor displayed or not. Curseur visible ou non. = CSRON et CSROFF 25 (J) et -25 (M)  
Notes l'utilisation d'un indicateur binaire (BMS) pour constituer une bascule. Dans le module PANAME les ordres CSRON et CSROFF n'utilisent pas d'indicateurs binaires.

46=LBL "Q"  
47 "S"  
127,831  
48 GTO 07  
49=LBL "R"  
50 "J"  
127,741  
51 GTO 07  
52=LBL "S"  
53 "T"  
127,841  
54 GTO 07  
55=LBL "T"  
56 "E"  
127,691  
57=LBL 07 SF 17 OUTA TONE 0 RTN  
58=LBL "U"  
59 "Z"  
127,371  
64 GTO 00  
65 X<>Y  
66 ABS  
67 ENTER  
68 32  
69 MOD X#0? GTO 08 0 32 GTO 10  
75=LBL 08 64 + 91 X<=Y? GTO 09 CLX X<>Y GTO 10  
84=LBL 09 RDN 32 - 0 X<>Y  
90=LBL 10 0 X<>Y BLDSPC ARCL X RCL00 ABS ENTER  
98 16 MOD X#0? GTO 11 0 32 BLDSPC ARCL X GTO 07  
107=LBL 11 64 + 0 X<>Y BLDSPC ARCL X GTO 07  
115=LBL "V" " " [27] 0 X<>Y BLDSPC ARCL X OUTA RTN  
123=LBL "W"  
124 F97 06  
125 "R"  
127,921  
126 F97C 06  
127 GTO 07  
128 SF 06  
129 "Q"  
127,811  
130 GTO 07  
131=LBL "1"  
132 3  
133 SELECT

Scroll up : deplace le page vers le haut. = SCPLU 10 (B)  
Efface l'écran à partir de la position du curseur jusqu'à la fin de la page. = CLEAR 10 (C)  
Scroll down : deplace le page vers le bas. = SCPLD 10 (D)  
Soft reset : efface l'écran totalement et place le curseur en position 0,0. = CLEAR -12 (M)

Cursor relative : tabulation du curseur. doit être suivi de 2 caractères ASCII. Notez la complexité des opérations de la séquence et comparez avec la fonction XYTAB du module affectée à -14 (M) qui sur 2 octets (1 instruction) effectue le même travail que les lignes 62 à 114.

Insert or replace cursor. Il existe 2 types de curseurs l'un barre l'autre sous forme d'une flèche. Mais ne permet pas pour autant d'insérer caractères avant des caractères. La méthode 00M de l'interface n'est pas adaptable.  
= 07MEE 115 (M)

"1" affiche "HELLO" en gros caractères sur l'écran de la video. sélectionne le Jere périphérique de la boucle





```

295 "*****"
    (184 à 198)
296 "i-*****"
    (127,199 à 207)
297 OUTA
298 "*****"
    (208 à 222)
299 "i-*****"
    (127,223 à 231)
300 OUTA
301 "*****"
    (232 à 246)
302 "i-*****"
    (127,247 à 254)
303 OUTA

304=LBL "4"
305 SF 17
306 "E<"
    (27,69,27,60)
307 OUTA
308 " START "
309 OUTA
310 "XXXXXXXXXXXXXXXX" Pour envoyer une séquence répétitive de caractères
311 "i-XXXXXXXXXX" le module PANAME permet "x" 30 OUTAX OU 80 30 OUTVEX
312 1.021

313=LBL 04 OUTA ISG X GTO 04
317 "XJC SCROLL UP "
    (27,37,74,67,32,83,67,82,79,76,32,85,80,32)
318 OUTA
319 .03
320=LBL 05
321 "S"
    (27,83)
322 OUTA
323 ISG X
324 GTO 05
325 "XIC SCROLL "
    (27,37,73,67,32,83,67,82,79,76,32)
326 "i-DOWN "
327 OUTA

328 .03
329=LBL 06
330 "T"
    (27,84)
331 OUTA
332 ISG X
333 GTO 06

334=LBL "5"
335 SF 17
336 "E<EJZT"
    (27,60,27,72,27,74,27,37,8,4) le module TIME est indispensable.
337 OUTA

338 "THE TIME IS:"
339 OUTA
340 "X"
    (27,37,8,0)
341 OUTA CLA CLKTD FIX 6 DATE " " ADATE OUTA FIX 4

350 1.01
351 STO 00
352=LBL 01
353 "XT"
    (27,37,8,6)
354 OUTA CLA TIME ATIME OUTA ISG 00 GTO 01 RTN

362=LBL "6"
363 SF 17
364 "E<XJ 1XT 7"
    (27,69,27,60,48,27,37,74,32,49,27,37,84,32,50)
365 OUTA
366 "XJ "
    (27,37,62,32)
367 OUTA
368 "3 11" XEQ "z" XEQ "z" XEQ "z"
372 "2 X" XEQ "z" XEQ "z"
375 "3 X" XEQ "z" XEQ "z"
378 "4 X" XEQ "z" XEQ "z"
381 "5 X" XEQ "z" XEQ "z"
384 "6 X" XEQ "z" XEQ "z"
387 "7 X" XEQ "z" XEQ "z"
390 "8 X" XEQ "z" XEQ "z"
393 "9 X" XEQ "z" XEQ "z"
396 "10 " XEQ "z" XEQ "z"
399 "11 " XEQ "z" XEQ "z"
402 "12 " XEQ "z" XEQ "z"
405 "13 " XEQ "z" XEQ "z"
408 "14 " XEQ "z" XEQ "z"
411 "15 " XEQ "z" XEQ "z"
415 "XOG>"
    (27,37,79,71,27,62)
416 OUTA RTN

418=LBL "z" "i-2345678901" OUTA CLA RTN
423=LBL "#" "i-XXXXX" OUTA RTN
427=LBL "<" "XXXXXXXXXXXXXXXXX" "i-XXXXXXXXXX" OUTA RTN
432 END

01=LBL "0"
02 XEQ 99
03 CHFLAG "
04 "

DEUXIEME VERSION AVEC LE MODULE PANAME
Determiner le type d'interface video.
Initialise le calculateur.
Au moment de l'écriture du programme, désarmer tous les
indicateurs binaires inutilisés: armer l'indicateur 05:

```

exécuter "CHELAG" en mode PRG4. La chaîne sera produite automatiquement par le module PANAME.  
 Au moment de l'exécution, le calculateur s'initialisera conformément à l'état défini pendant la production.

Affiche "HELLO" en gros caractères sur la vidéo.

Enumère les différentes possibilités de l'interface.

Liste les 190 caractères disponibles avec l'interface.

Démonstration du 'scrolling'.

Exemple d'affichage des résultats.

Exemple de tabulation et de graphisme simple.

Alterne l'affichage 40 ou 90 colonnes.

40 ou 80 colonnes.

Bascule sur l'indicateur F 01

"XTGAR" est une fonction du module PANAME. Elle est semblable à "XTOA" du module XFUNCTION.

Choisit le mode 40 ou 80 colonnes.

Affiche "HELLO" en gros caractères.

Efface l'écran. = ESC E c'est à dire les lignes 55 à 61 de la version 1. De meme CSROFF efface le curseur. = ESC C

Place un espace dans le registre ALPHA.

Evite l'envoi de fin ligne après chaque "OUTA".

Il y a 7 lignes identiques à créer sur l'écran.

écriture de chaque ligne.

Definit le nombre de répétitions pour la fonction "OUTAV".

Envoie le contenu du registre ALPHA sur l'écran autant de fois qu'il est indiqué par le registre X.

Envoie le contenu du registre ALPHA sur l'écran en armant le bit de poids fort; c'est à dire que le contenu du registre ALPHA est affiché en vidéo inversée.

Envoie 3 espaces.

Envoie un " " sur la vidéo.

Envoie 1 espace.

Envoie 5 espaces sur la vidéo.

Envoie 1 " " .

05 MANIO

06 LBL 08

07 XEQ "1"

08 XEQ 14

09 XEQ "2"

10 XEQ 14

11 XEQ "3"

12 XEQ 14

13 XEQ "4"

14 XEQ 14

15 XEQ "5"

16 XEQ "6"

17 XEQ 08

18 6TO 08

19 LBL 09

20 FC7C 01

21 SF 01

22 CLA

23 27

24 XTOAR

25 91

26 F57 01

27 93

28 XTOAR

29 OUTA

30 RTN

31 LBL "1"

32 CLEAR

33 CSROFF

34 " "

35 SF 17

36 7

37 00

38 LBL 11

39 2

40 OUTAX

41 OUTA

42 3

43 OUTAX

44 OUTA

45 OUTA

46 OUTA

47 5 OUTAX

48 OUTA

49 OUTA

50 OUTAX OUTA OUTA OUTA

Envoie 5 espaces. = ESC E  
 Déplace le curseur sur la gauche.  
 = ESC D lignes 22 à 29 et 31 à 37 de la première version.  
 Place 1 " "

Envoie un retour chariot. retour en début de ligne.  
 Envoie une fin de ligne. passe à la ligne suivante.

Positionne le curseur en 22. Notez la facilité de mouvement comparativement aux lignes 52 à 61 de la première version.

Déplace le curseur d'un caractère vers la droite.  
 = ESC C lignes 25-27 et 50-51.

Déplace le curseur verticalement conformément à la valeur placée en X. De même horizontalement.

Il n'y a pas d'écoulement: il faudrait envoyer X fois ESC A (curseur up).

Déplace le curseur vers le bas d'un caractère. = ESC B

Envoie X fois le contenu du registre ALPHA en vidéo inversée

81 CSRR OUTAX CSRR OUTA 3 OUTAX OUTA

88 0 25 XYTAB CSRR OUTA 3 OUTAX OUTA

95 PSE PSE

98 CLEAR CSROFF

100 4

101 3

102 XYTAB

103 "WELCOME TO THE "

104 "1--WORLD OF"

105 OUTA

106 6

107 9

108 XYTAB

109 "HP-41 VIDEO"

110 CF 17

111 OUTA

112 SF 17

113 CSRON

114 "AND THE HP 163 "

115 OUTA

116 "VIDEO INTERFACE"

117 OUTA

118 XEQ 14

Laisse le temps de lire le texte présent à l'écran.  
 Equivalents à la ligne 225 de la première version.

Remarque: la lisibilité du listing en regard de la ligne 211 de la première version.  
 Envoie le contenu du registre ALPHA en vidéo inversée.

119#LBL "2"      Enumere les differentes possibilites de l'interface.  
170 CLEAR      CSROFF

122 " "      " "  
123 15 6 XYTAB CF 17  
127 "82163A FEATURES" OUTA SCRLUP CSRDN  
131 "VIDEO OR TV<CH3" "I->OUTPUTS" OUTA  
134 "HP-IL " "I-CONTROLLED" OUTA  
137 "FULL CURSOR " "I-CONTROL" OUTA  
140 "INVERSE " "I-CHARACTERS" OUTA  
143 "TRACE MODE" OUTA SF 17  
146 "190 DISPLAYABLE" OUTA CF 17  
149 " CHARACTERS" OUTA  
151 "16 <31> LINES " SF 17 "I-OF 32 " OUTA CF 17  
156 "CHARACTERS" OUTA  
158 "CURSOR THIS " "I-OR" OUTA

161 -3 SCRLX      Monte la page de 3 lignes sur l'ecran.  
163 -4 CSRVX 17 CSRHX

167 4      Deplace le curseur de 4 lignes vers le haut et de  
168 510 00      17 caracteres vers la droite.  
169#LBL 03      Le curseur sera affiche sous ses 2 formes 4 fois.  
170 1      Affichage du curseur sous ses 2 formes.  
171 CTYPE      Choix du curseur ( le registre X doit contenir 0 ou 1).  
172 CSRDN      Curseur visible.  
173 -5 CSRHX      Deplacement de 5 caracteres vers la droite.  
175 PSE  
176 5 CSRHX      Deplacement de 5 caracteres vers la gauche.  
178 0 CTYPE PSE      Choix du curseur de remplacement.  
181 DSE 00 610 03

183#LBL "3"      Affichage des 190 caracteres utilisables sur la video.

184 SF 17 CLEAR CSROFF 2 3 XYTAB  
190 "190 " OUTA "DISPLAYABLE CHA" "I-RACTERS" OUTA

195 4      Debut d'écriture 4eme ligne  
196 1      1er caractere de la ligne.  
197 XYTAB      Positionne le curseur.

198 33.126      Affichage des caracteres de code 33 a 126.  
199#LBL 51      Envoie sur le periphérique selectionné (video) le caractere  
200 OUIXB      correspondant a la valeur contenue dans le registre X.

201 156 X 610 51

203 OUTCR OUTLF      Retour a la ligne.

206 160.255  
207#LBL 52      Affichage des caracteres de code 160 a 255.

208 OUIXB      Les lignes 132 a 210 sont equivalentes aux lignes 257 a  
209 156 X      303 de la premiere version.  
210 610 52

211 PSE PSE

213#LBL "4"      Demonstration des fonctions de "scrolling"

214 SF 17 CLEAR CSROFF  
217 " HOME " OUTA

219 88      Le caractere de code 98 est 'x'  
220 40      specifie le nombre de caracteres a envoyer.  
221 OUIYBX      Envoie 40 fois l'octet specifie en X (CHR 98 = 'X')

222 " SCROLL UP " OUTA CLX

225 350 OUIYBX      Envoie 350 fois l'octet 88.  
227 HOME      Remplace le curseur a la position 0,0  
228 PSE

229 .014  
230#LBL 05      Deplacement de la page 15 lignes vers le haut.  
231 SCRLUP      = ESC 5 : ligne 221 de la 1ere version.  
232 156 X 610 05

234 15 SCRLX      Provoque un deplacement instantane de 15 lignes vers le bas.  
236 47 CSRHX      Positionne le curseur 47 caracteres a droite. Ceci permet la  
compatibilite du programme pour les differentes interfaces.

238 " SCROLL DOWN " OUTA  
240 PSE PSE  
242 -15 SCRLX      Deplacement instantane de 15 lignes vers le haut.

244 .014  
245#LBL 06  
246 SCRLDN      Deplacement de la page d'une ligne vers le bas.

247 156 X 610 06  
249 PSE PSE

251#LBL "5"      Affichage de resultats sur la video pour un usage collectif.

252 SF 17 CLEAR CSROFF 4 8 XYTAB  
258 "THE TIME IS:" OUTA CLA  
261 ENTER XYTAB  
263 CLKTD FIX 6 DATE " " ADATE OUTA FIX 4

270 1.01 510 00      Affichage de l'heure pendant une dizaine de secondes.

272#LBL 01 6 8 XYTAB CLA TIME ATIME OUTA  
280 156 00 610 01

282#LBL "6"      Utilisation de l'ecran pour des jeux ou graphismes simples.

283 SF 17 CLEAR CSROFF  
286 "0" OUTA 9 OUI5PX

Envoie le contenu du registre ALPHA suivi de 9 espaces.

290 "1" OUTA OUTSPX "2" OUTA OUTSPX "3" OUTA OUTCR OUTLF  
300 " " XEQ "x" XEQ "x" XEQ "x" OUTCR OUTLF

306 2.015 Préparation de la pile pour la boucle 07:  
307 2 Ecrire les lignes 2 à 15.  
308 88 Justification du texte sur 2 caractères.  
309 29 Accoler des caractères de code 88 ('X').  
310=LBL 07 Accoler 29 de ces caractères.  
311 R^  
312 " "  
313 APPX Ecrit à droite de l'espace placé dans le registre ALPHA  
314 R^ la partie entière du registre X sans la virgule.  
315 SUBS conserve l'espace à droite du nombre du registre ALPHA  
316 "1- " uniquement pour les nombres à 1 chiffre. Place un espace  
317 OUTA à droite.  
318 R^ R^ OUTYBX Envoie 29 fois 'x' à la suite.

321 OUTCR OUTLF ISG T 6TO 07 Ecrit les 15 lignes.

325 SCRLDN TONE 9 7 15 XYTAB CSRON RTN

332=LBL "x" "1-1234567890" OUTA CLA RTN

337=LBL 14 PSE PSE PSE RTN

342=LBL 99 Détermine le type d'interface, sa position sur la boucle.  
343 48 Identification de l'interface HP 82163.  
344 FINDAID Recherche la position de l'appareil N°48. Ne pas confondre  
FINDID du module HP-IL qui recherche la position d'un  
accessoire dont l'identité est spécifiée en ALPHA.  
345 X=0? 6TO 00 Si la vidéo HP n'est pas présente, continue la recherche.  
347 SELECT RTN Si l'appareil est trouvé, le sélectionne et retourne

349=LBL 00  
350 50 FINDAID Recherche l'interface MC00701A (le N° d'appareil est 50.)  
352 X=0? 6TO 00 Si aucune interface n'est trouvée...  
354 SELECT Sélectionne l'interface Mountain.

355 CLA 27 XTOAR 107 XTOAR OUTA RTN

Sélectionne le jeu de caractères standard.

362=LBL 00 "NO VIDEO" BEEP PROMPT END

# HP-IL

Philippe TENAND

L'HPIL ET SES ELEMENTS  
Le point sur l'HPIL.

## L'OUTIL HPIL ET SES ELEMENTS.

Il ne s'agit pas d'une interface permettant de relier de gros ordinateurs. Il s'agit d'une interface peu coûteuse, pratique et toutefois puissante, permettant des applications simples entre des matériels peu coûteux ou transportables. Il est donc possible, avec du matériel compatible HPIL (micro-ordinateur de poche), de saisir des données n'importe où, le tout tenant dans une mallette, voire une grande poche. L'intérêt est sa grande gamme de possibilités: le système HPIL permet de relier des calculatrices à des systèmes plus coûteux et performants, pour des applications spécifiques et professionnelles. Fabriqué par Hewlett-Packard, l'HPIL (ou Hewlett-Packard Interface Loop) est constitué par un câble à deux conducteurs qui permet un raccordement rapide et simplifié. Insensible aux parasites, ce système d'apparence fragile est capable de raccorder dans une même boucle 30 périphériques, chacun espacés de 100 mètres.

Utilisable sur une table, avec du matériel compatible, ou dans la rue, l'HPIL reçoit sans cesse par son constructeur de nouveaux périphériques, dont le nombre augmente même pendant que vous lisez ces lignes. Tous les nouveaux ordinateurs de table ou transportables HP sont prévus pour avoir une option HPIL au sein de la marque. Au temps où la boucle fut créée,

les applications étaient peu nombreuses, mais rapidement les périphériques nouveaux montrèrent les applications possibles au sein d'une entreprise, d'un laboratoire, d'une étude sur le terrain, de prises automatiques et autonomes de données physiques, aidés par un logiciel de plus en plus important et traitant tous les sujets. De très nombreuses entreprises de création de logiciels, indépendants de Hewlett-Packard créèrent des centaines de programmes tout fait, et ceci pour toutes les applications possibles (aviation, navigation, astronomie, gestion, traitement de textes, prises de données, béton armé, construction navale, etc...) ainsi que du matériel (imprimantes, extensions diverses) et des livrets d'applications, tout ceci pour l'utilisation de la boucle HPIL, ce qui est à rajouter aux logiciels et à l'aide fournie par Hewlett-Packard (ordinateurs portables personnalisés, modules ROM personnalisés, etc...).

Faisons d'abord une liste de tout le matériel compatible HPIL existant, que nous verrons avec plus de détails ensuite, et nous finirons par quelques conseils et matériaux utiles pour utiliser au mieux la boucle HPIL, que ce soit au sein de Hewlett-Packard ou par des fabricants indépendants.

## Liste des périphériques HPIL.

SYSTEMES:

- série 40: HP41C, HP41CV, HP41CX.
- série 70: HP71B, HP75C, HP75D.
- série 80: HP83, HP85, HP86, HP87.

Et par l'intermédiaire des interfaces: HP1000, HP3000, HP150, etc..., ainsi que du matériel rendu compatible HPIL avec le KIT Hewlett-Packard.

#### PERIPHERIQUES:

- HP 82161A, unité à microcassettes numériques.
- HP 82162A, imprimante / traceur thermique.
- HP 82163B, interface vidéo 32 colonnes (options toutes normes).
- HP 92198A, interface vidéo 80 colonnes (options toutes normes).
- HP 82912A moniteur vidéo
- HP 82168A, modem acoustique.
- HP 2225B, thinkjet, imprimante à jet d'encre vaporisé.
- HP 82905B, option 248, imprimante 80 colonnes à impact.
- HP 7470A, option 003, table traçante / traceur graphique.
- HP 2671A, option 048, imprimante thermique alpha numérique.
- HP 26716, option 048, imprimante thermique graphique.

Diverses imprimantes, tables traçantes, (Seikosha GP100A, Canon X710, Tandy CGP115, etc...) interfacées HPIL par des Clubs, ou des distributeurs indépendants.

- HP 5006A, analyseur de signature.
- HP 3468A, multimètre digital.
- HP 5385A compteur de fréquence.
- HP 3421A, unité de contrôle et d'acquisition de données.
- HP 3056DL, rack pour l'union de plusieurs HP 3421A.

#### INTERFACES:

- HP 82160A, interface HPIL pour série 40.
- HP 82401A, interface HPIL pour HP71.
- HP 82938A, interface HPIL pour série 80.
- HP 82166, kit de convertisseur d'interface HPIL.
- HP 82164A, interface HPIL -- RS232C.
- HP 82169A, convertisseur HPIL -- HP1B.
- HP 82165A, convertisseur GPIO.

Notons qu'avant l'existence de la boucle HPIL et du système de la série 40, le HP97 modèle "S" était équipé d'une interface DCB (prise type AMPHENOL 57-30500 ou équivalent), (décimal codé binaire), vieux modèle coûteux et peu performant, mais unique en son genre. Le système HPIL, lui, est infiniment plus pratique et moins onéreux.

Les modèles des séries 40 et 70 ont des grilles d'identification interchangeables sur claviers.

#### LES SYSTEMES COMPATIBLES HPIL

Nous n'insisterons pas sur les modèles ne représentant pas une nouveauté dans la marque.

#### SERIE 40

Système de poche utilisant le langage RPN (notation polonaise inverse), système évolutif dont un HP41 est le centre, avec mémoire permanente, affichage à cristaux liquides, touches personnalisables, système conversationnel et clavier alphanumérique. Le modèle HP41C a 441 octets, la 41CV a 2233 octets et la 41C se transforme en 41CV par l'adjonction d'un module mémoire appelé "QUAD". Possibilités d'extension jusqu'à 6,4 k-octets. La 41CX est une 41CV plus puissante en possibilités (horloge, fichiers, chronomètre) et intégrant les modules XFUNCTIONS et le module horloge, en plus de particularités propres. Il possède quatre logements pour enfichage et extensions. Parfait pour une utilisation sur le terrain et le nombre d'extensions utilisable sur pile, batterie et adaptateur AC. Le nombre de fonctions grandit par ajout de ROM. Pour une utilisation professionnelle, le HP41CX est recommandé, modèle puissant et résistant.

#### SERIE 70

Récemment distribué, le HP71B semble plus petit, moins cher et plus puissant que le HP75. Il répond à un besoin différent, le langage basic du HP75 est moins complet que celui du HP71B, mais tous deux ont des possibilités égales ou supérieures à certains modèles de table. Ils disposent d'un écran à cristaux liquides et si un lecteur de cartes magnétiques est intégré au HP75, il est en option sur le HP71B. La 75 contient au maximum de ses extensions (trois espaces de sortie) 96 k-octets de ROM et 24 k-octets de RAM, l'interface HPIL est intégrée au HP75. Le HP75D diffère du HP75C par une interface intégrée de lecture des codes barres, afin d'offrir une solution à la collecte de données sur des sites décentralisés.

Exemple d'utilisation: un ingénieur de maintenance peut utiliser le HP75D au cours d'une intervention pour enregistrer tous les détails de cette intervention, quand celle-ci est terminée, il peut transmettre son compte rendu à son centre

de maintenance grâce à un modem, à l'inverse, il peut recevoir des messages concernant ses rendez-vous ou des informations de maintenance. Il est aussi parfait pour la collecte des données de stocks en magasin, l'assistance à la vente ou le suivi des échantillons en laboratoire, par exemple à l'aide du lecteur de codes-barres (en option). Il existe un module ROM de lecture de codes-barres qui permet de lire la plupart des codes-barres industriels:

3,9

2,5 entrelacé

11

UPC A ou E

EAN 8 ou 13

CODABAR

De plus un berceau alimenté sur piles (HPB271BA), dans lequel repose le HP75D ajoute une connexion modem 300 bauds et 32 ou 64 k-octets de mémoire disque-électronique. L'ensemble autonome berceau et 75D mesure 16,5x29,2x5,7cm. Un module ROM entrées/sorties est disponible pour les applications de contrôle sous HPIL et la programmation de haut niveau. Le HP75 est un outil de calcul polyvalent personnalisable, un compagnon de voyage léger mais très puissant, avec, par la boucle HPIL un choix croissant de périphériques et d'extensions.

Le nouvel ordinateur portable HP71B est plus séduisant, et légèrement plus petit. Son basic est plus puissant que celui du 75. Le 71 reçoit 33,5k-octets de RAM et 256k-octets de ROM au maximum. Le HP71B dispose probablement du système d'exploitation le plus puissant du marché, il combine le langage basic et un mode calcul qui en fait un remarquable outil de résolution des problèmes. Il possède 4 ports d'enchâssement et est un système des plus portatifs, au même titre qu'une HP41, reprenant les possibilités de redefinissage des touches. Un affichage à une ligne et cristaux liquides de 8x132 éléments présente des caractères très lisibles. 22 caractères sont affichés à la fois, les lignes comptant 96 caractères portatifs, il est alimenté par 4 piles AAA ou par secteur.

Le HP71B créé plus intelligemment que le HP75, que ce soit par ses touches à fonctions multiples ou par son incroyable facilité d'emploi, il fait d'emblée l'unanimité et détrône sinon par le prix, du moins par les performances et la taille, ce qui était l'avatar de la HP41. Tout aussi professionnel que le HP75, il est aussi portable

que la HP41 pouvait l'être, réunissant par cet aspect polyvalent les adeptes de l'un et de l'autre. Les cartes magnétiques du HP71B sont les mêmes que celles du HP75, de plus le HP71 semble être plus rapide et sa programmation s'apprend plus rapidement que celle du HP75, grâce aux fonctions assignées et gravées à trois niveaux sur certaines touches du 71B. A l'usage aussi, il se trouve plus maniable. Si l'avantage du 75D est son lecteur optique, il reste à attendre cette extension sur le HP71B pour en faire un "state of the art" de l'informatique portable comme l'est le HP41 sur le marché des calculatrices programmables.

SERIE 80

En quittant les séries 40 et 70, nous tombons dans le matériel de table non transportable et qui ont des buts et des possibilités tout à fait différents. Nous avons là l'écran vidéo (intégrés pour certains de la série 80), des mémoires de masse de grande capacité et dont les possibilités sont encore plus élargies par la boucle HPIL. Un exemple: vous pouvez mettre dans une boîte, en pleine forêt, (ou sur une ligne électrifiée), une HP41 munie d'un module horloge et d'un multimètre, qui se mettront en marche et s'éteindront automatiquement toutes les heures, le temps de mesurer une température (ou une tension) et de stocker les résultats soit en mémoire fichier, soit dans une unité de microcassette digitale. Deux semaines plus tard, vous pouvez réunir les informations en échangeant la microcassette contre une autre, et vous retrouver avec votre ordinateur de table (série 80) grâce à la boucle HPIL qui utilise le même matériel, à savoir une unité à cassette HPIL, il vous suffit de mettre la cassette en question pour que votre ordinateur de la série 80 vous délivre graphiques et informations à souhait, et vous dicte par exemple les modifications à apporter dans un plan ou que faire des informations suivant leurs qualités par rapport à une moyenne ou à un plan préétabli.

L'union d'un ordinateur de poche HPIL avec un ordinateur de table HPIL est un outil des plus performants. Les calculateurs portables HPIL permettent des acquisitions de données aux prix les plus bas du marché. Et en utilisant les modèles HPIL plus performants, vous acquierez une performance et une rapidité d'étude la plus rapide qui soit dans la gamme de prix de ces

matériels. L'idéologie et le principe même de la boucle HPIL en fait le summum au point de vue rapport performances/prix. Un dialogue parfait peut enfin se faire entre l'ordinateur puissant (car par le biais des interfaces, on atteint les gros ordinateurs HP) et le petit ordinateur apportant des données urgentes et multiples, par le biais d'un même langage, celui qui unis les périphériques de la boucle. C'est ici la notion de prix et de simplicité qui l'emporte, par son originalité, sur d'autres boucles ou interfaces qui n'ont pas les mêmes buts ni les mêmes capacités. L'HPIL a pour base un matériel léger et performant et joignant l'utile à l'agréable, il communique avec du matériel plus lourd. Les ordinateurs de la série 80 n'étant pas des nouveautés récentes, nous abrègerons leurs caractéristiques. C'est le type de matériel qui a autant besoin d'interfaces performantes comme le HP1B (etc...) que le HPIL pour l'arrivage rapide de données nombreuses ou sur le terrain.

Au début était le HP85, complet, puis est arrivé le HP83, simplifié, le HP87 plus puissant et à l'écran plus large, et conjointement le HP86 et le principe modulaire: l'écran était séparé du reste, et dans un autre modèle on avait droit à un ordinateur type HP85 où tout était séparé: clavier, écran, mémoire de masse, unité centrale. Nous n'en parlerons pas ici.

Un défaut du HP85 était un écran très petit: 12,7cm de diagonale. Il servait donc à analyser des données plutôt qu'à faire de jolis dessins, même si la définition des caractères (matrice 5x7) n'était pas mauvaise, avec graphiquement 192x256 points. Le HP85 reçoit 32 k-octet au maximum et contient intégré une unité à cartouche performante (210 k-octets de données et 195 k-octets de programme) et une imprimante (10,8cm de largeur de papier), un vrai petit ensemble performant pour bien des applications.

Le HP83 offre une puissance de calcul identique à celle du HP85, dans un boîtier semblable, mais est destiné à ceux qui veulent choisir une autre capacité de stockage et une imprimante différente ou plus performante, l'écran étant le même que celui du HP85. Dans l'optique inverse, le HP87 est un HP85 avec écran deux fois plus large, plus puissant et sans unité à cartouches magnétiques (spécialiste HP de cartouches incassables) ni imprimante, laissées au choix et de préférence, comme disent les manuels d'utilisation, dans les

modèles de la même marque. L'écran passe à 544x240 points, la mémoire est de 544 k-octets en version pleine (d'origine à 32 k-octets).

Le HP87 est un système d'informatique personnelle très souple destiné aux scientifiques et aux ingénieurs confrontés à des calculs complexes, ayant une structure modulaire, avec interface HP1B intégrée et, (c'est pourquoi nous le nomons) l'option IL, (entre autres). Grâce à un très large choix de mémoires de masse, d'imprimantes et de traceurs de courbes et aux moyens de communications avec d'autres ordinateurs, calculateurs et systèmes d'instrumentation, le HP87 s'adapte aux travaux de laboratoire, d'atelier ou de bureau. Avec des langages quasi-identiques les modules de la série 80 voient leurs logiciels modifiés seulement pour que l'utilisateur exploite leur machine au maximum, mais ils sont pratiquement échangeables l'un à l'autre (différence d'écran pour le HP87). Le HP86 est un clavier, auquel il faut tout apporter, un moniteur vidéo de grande taille (24 lignes de texte sur 80 colonnes), des unités de disque souples compatibles avec celles du HP87 et surtout son entière compatibilité avec le HP87, et sa capacité de 576 k-octets de mémoire utilisateur (et 48 k-octets pour le système) la boucle HPIL nous amène très très loin! par rapport à ses aspirations du début...

Après les systèmes, voyons les périphériques HPIL, utilisables même à partir du petit (mais suffisamment puissant) HP41.

S'il y a plus de périphériques pour le petit matériel (série 40, 70), c'est que justement c'est le créneau de la boucle HPIL: le faible coût. Brancher un HPIL sur un matériel trop important devient ridicule, laissons cela à l'HP1B ou autres RS-232, ... sauf dans certains cas (prises de données sur le terrain).

L'avantage de la portabilité du HP41 en fait justement l'outil idéal, par son dialogue fonctionnel, ses dispositifs de contrôle intégrés, sa prévention et détection des erreurs, son affichage multifonction, ses signaux d'avertissement sonore, ses solutions simples aux problèmes difficiles par l'HPIL. La 41 devient un réel système informatique personnel.

Le module d'interface HPIL transforme la HP41 en contrôleur de boucle, il comprend les câbles,

l'électronique et les fonctions nécessaires pour gérer l'interface, la mémoire de masse et l'imprimante graphique. Les fonctions générales d'entrée/sortie du module d'interface HPIL vous permettent de gérer d'une manière précise le fonctionnement de la boucle.

Le HP82161A est le lecteur de cassette numérique, 131000 octets de données, c'est à dire 50 fois la capacité du HP41 peuvent être stockés sur une mini cassette numérique HP (cassettes spéciales de marque HP, et pas une autre). L'accès rapide fait de cette unité un dispositif idéal de stockage de masse pour les applications fréquemment mises à jour.

L'imprimante thermique HP82162A est petite et pratique, et peut juste créer une courbe ou des codes barres (grâce au module ROM "plotter") d'une manière graphique. A part cela, elle est alphanumérique et permet d'imprimer tous caractères spéciaux (créés par la HP41) ou de lister un programme. Elle imprime tout résultat et tout message (en une seule couleur: noir ou bleu), dans diverses largeurs, avec ou sans coupures de mots, ce qui est suffisant pour des petits résultats et est parfois très esthétique quand aux résultats.

Le module d'interface vidéo HP82163B possède un connecteur HPIL à une extrémité et un connecteur standard TV (radio, fréquence et vidéo) à l'autre. Il vous permet d'afficher des données et des listages de votre système sur des moniteurs de télévision: 23 ou 30cm, ou sur des récepteurs de télévision classiques équipés d'une entrée vidéo ou recevant sur le canal E4 (CCIR bande I). Passant sur les interfaces dont le fait d'exister n'apporte guère de commentaires, leur existence étant simplement la preuve d'une conception bien faite d'un système et de son universalité. Un système de si faible coût avec tant de possibilités est unique et original. Disons juste que le Kit /convertisseur d'interface HPIL HP82166C est destiné aux OEM, à l'enseignement, aux électroniciens ou aux amateurs. Il relie l'interface HPIL à deux bus bidirectionnels 8 bits parallèles (ou à un bus bidirectionnel 16 bits). Le convertisseur comprend tous les circuits d'interface permettant de réaliser une liaison entre l'HPIL et d'autres équipements électroniques. L'ensemble se compose d'une documentation complète sur les composants, de quatre jeux de composants pour la réalisation de

prototypes et du logiciel de développement HPIL utilisable sur des calculateurs de la série 40.

L'imprimante à impact HP82905B (option 34B) édite sur 80 colonnes des caractères de 9x9 points, avec impression bidirectionnelle de 80 caractères par secondes. Un jeu de caractères permet d'éditer des textes en plusieurs langues. La longueur de la page et les sauts de perforations sont également programmables. (NDLR: c'est celle-ci qui imprime le journal)

La table traçante HP7470A (option 003) permet de réaliser des travaux de qualité professionnelle. Ce traceur à 2 plumes, en couleurs, possède 40 instructions graphiques complexe HPGL (langage graphique Hewlett-Packard) et 5 jeux de caractères internes, dont 3 jeux européens. Le test peut-être écrit dans une direction quelconque, avec ou sans inclinaison des caractères et en différentes tailles, l'utilisation de symboles intégrés au traceur et de 7 types de lignes en pointillés contribue à clarifier les graphes les plus complexes.

D'autres imprimantes HP et compatibles HPIL (voir liste précédente) existaient déjà, la nouveauté étant leur interfaçage HPIL. La dernière nouveauté est en fait dans le domaine des imprimantes, la THINKJET ou HP2225B, imprimante à jet d'encre vaporisé, utilisant une nouvelle technologie, c'est la première application dans la catégorie portable, le niveau sonore est insignifiant, la qualité d'impression proche des systèmes de traitement de texte, elle imprime 80 colonnes en faisant 50 db de bruit et a une vitesse de 150 caractères par seconde (cps). En mode 'caractères compressés', elle va jusqu'à 142 colonnes. Petite, sa batterie miniature lui permet d'imprimer 200 pages, mais elle possède une alimentation secteur. Les caractères ont une matrice de 11x12 points. Le tampon d'impression permet de mémoriser un kilo-octets. Le coeur en est la tête d'impression remplaçable HP92261. Le nouveau procédé de vaporisation possède 3 avantages:

- l'encre se trouve divisée en particules infimes, permettant un dosage extrêmement précis (on peut s'attendre à l'avenir à voir venir des modèles permettant la quadrichromie).
- La vaporisation qui dégage le liquide par une pression assure le transfert des particules d'encre sur le papier dans un silence absolu. (C'est la dilatation qui permet la projection de

l'encre).

- Cette suppression de la mécanique permet une consommation d'énergie qui se résume dans la pratique au seul déplacement de la tête d'impression et à la surchauffe de quelques décigrammes d'encre pour une page. Si le jet d'encre n'est pas nouveau, le principe de la vaporisation d'encre, en revanche, l'est.

Le HP3468A est un multimètre numérique 3 chiffres 1/2 à 5 chiffres 1/2, 5 fonctions à changement de gamme automatique avec une sensibilité de 1 microvolt, il permet de répondre aux applications exigeantes. Doté de l'interface bifilaire (HPIL), il est gérable par ordinateur portable HP41, HP71B, ... Pour mesurer des températures, un modèle HP linéarise la réponse du capteur et inscrira le résultat en degrés C ou degrés F directement. On peut obtenir des lectures en dB par rapport à une impédance quelconque. Pour les mesures de tolérance sur résistances ou les tests de comportements, le 41 (ou le 71B, ou le 75, ou le 85, ...) peut sans difficulté recueillir les mesures du 3468A, effectuer un calcul de pourcentage d'erreur et inscrire le résultat directement en pourcentage. Un système immédiatement opérationnel vous est offert. Une imprimante ou une unité de cassette seront peut-être utiles. Vous pouvez mesurer des tensions continues et alternatives efficaces vraies à partir de 0,3V pleine échelle avec une sensibilité de 1 microvolt jusqu'à 300Volts. Cette plage dynamique étendue permet de détecter des signaux de faible niveau tout en réduisant le degré de conditionnement nécessaire au signal. Les mesures de résistance peuvent être faites en configuration 2 ou 4 fils, avec une gamme maximale de 300 méga-ohms et une sensibilité allant jusqu'à 1 milli-ohm sur la gamme 300 ohms. Vous pouvez utiliser la configuration 4 fils pour minimiser les erreurs dues à la résistance parasite des câbles de liaison ou utiliser la configuration 2 fils plus simple à mettre en oeuvre. Le 3468A mesure les courants continus et la valeur efficace vraie des courants alternatifs jusqu'à 3A. L'étalonnage électronique complet est très rapide, électroniquement, manuellement ou à distance. Une fonction SRQ permet d'interrompre tout programme. Le HP3468A contient aussi un test automatique, des batteries rechargeables en option ainsi que des sondes diverses et cordons de test en option, pour utilisations variées.

Deux nouveaux compteurs de fréquence économique,

le HP5384A et le HP5385A, offrent des possibilités telles qu'une gamme de fréquence étendue, une résolution et une précision élevées, une sensibilité d'entrée élevée, et d'importantes possibilités de conditionnement du signal d'entrée, qui font de ces compteurs des instruments HPIL idéaux pour de nombreuses applications de test, d'étalonnage, de recherche et de développement. La gamme de fréquence est de 10Hz à 225MHz pour le HP5384A et de 10Hz à 16Hz pour le HP5385A. Parmi les caractéristiques communes aux deux instruments, on peut citer:

- Résolution de 9 chiffres par secondes
- 15mV eff de sensibilité maximale d'entrée,
- un affichage variable de 3 à 11 chiffres,
- un mode d'affichage à distance avec un afficheur alphanumérique à cristaux liquides, programmable, à 12 caractères, une commande pour clavier, manuellement par la face avant ou à distance à partir d'un contrôleur, le tout dans un coffret portable pouvant être monté en baie et empilé, alimenté par batterie interne ou alimentation continue externe (9-15V).

Le HP5006A est, comme beaucoup de matériel dans cet article, une première mondiale quand à son inclusion dans un système de faible coût et est le premier analyseur de signature HPIL. Il offre également deux nouvelles fonctions: signature composite et mémoire de signature, qui font gagner du temps à l'utilisateur dans le dépannage manuel de circuits numériques. La signature composite est la somme binaire (modulo 2<sup>16</sup>) de nombreuses signatures individuelles. L'opérateur gagne du temps en introduisant une série de signatures individuelles avec l'interrupteur de la sonde, sans cesser de voir si la signature de chaque noeud individuel est correcte. Seule la signature composite résultante est comparée au résultat attendu. Si la signature composite est correcte, tous les noeuds individuels sont supposés corrects. Dans le cas contraire, le dépanneur n'a pas à recontrôler chaque noeud, la mémoire tampon de l'analyseur sauvegarde et étiquette les 32 dernières signatures introduites. Le dépanneur peut examiner le contenu de la pile pour déterminer le noeud défectueux. Un système de test, par petit poste spécialisé, économique et tenant dans une mallette est ainsi possible.

Philippe TENAND

# HP-71

## HP71 ASSEMBLEUR

## ASSEMBLEUR

Pierre DAVID	LES BUFFERS DU HP71B
Laurent ISTRIA	INTERCEPTION DE POLL
Michel MARTINET	REPLACE#
Sthephane BARIZIEN	FILE?
ISTRIA/MARTINET/DAVID	PEEK POKE PARTOUT
Pierre DAVID	BIP TOUT LE TEMPS
DAVID/MARTINET	ATTENTION !
Michel MARTINET	MANIPULATION DES CHAINES ALPHA

## HP71 BASIC

## BASIC

Pierre DAVID	CHOC EN RETOUR
Daniel JACOB	INTEGRATION NUMERIQUE DE GAUSS
Jean-Claude FOURES	UTILITAIRES
Philippe DAVASE	VISUALISATION DE LA RAM
Daniel JACOB	DOW,JDS,DATE et HEURE
Serge VAUDENAY	LE Debilometre

## LES BUFFERS DU HP71

Tous ceux qui se sont intéressés de près ou de loin au FORTH71 ont été intrigués par les mots traitant des Buffers. Il est vrai que le manuel du Forth/Assembleur n'est pas très explicite sur ce sujet (comme sur bien d'autres d'ailleurs).

Notre but, ce mois-ci, sera de comprendre l'utilité des buffers, leur structure, et leur emploi.

Pour cela, je vous propose quelques petits exemples d'utilisation depuis BASIC. Si Forth propose déjà des mots tout préparés, l'utilisation que j'en fait est plus facile depuis BASIC. J'ai donc fait un LEX (ADBUF) qui permet de s'affranchir des 3 Ko de FORTHRAM. D'autre part, un fichier LEX peut circuler, un module Forth/Assembleur difficilement. La compréhension des buffers ne sera donc pas (j'espère) limitée aux heureux possesseurs du module.

Mais entrons dans le vif du sujet:

### QU'EST CE QU'UN BUFFER ?

Un buffer (tampon en français) est un espace mémoire qui sert à stocker des "choses".

Un fichier a une structure et une utilisation trop rigide. Les variables sont trop spécialisées (chaînes et nombres). Réserver une zone mémoire rien qu'à la "chose" reviendrait à gaspiller trop d'espace si la dite "chose" prend moins de place que prévu.

Alors, HP nous a offert une structure très dynamique, très souple: les buffers, encore appelés les "I/O Buffers" (Littéralement: tampons d'entrées/sorties). On peut les créer, les supprimer, les allonger, les rétrécir, et bien sûr stocker et récupérer des "choses".

Ces "Choses", qui vous irritent depuis le début de cet article, ce peut être n'importe quoi.

Citons quelques exemples:

- Le STARTUP est stocké dans un buffer. Si vous

avez essayé de chercher son adresse, vous avez peut être constaté qu'elle n'est pas fixe. C'est une caractéristique des buffers.

- Quand vous faites ASSIGN#, le 71 stocke quelques informations dans le FIB, qui est un buffer.

- Le jeu de caractères défini par CHARSET est dans un buffer.

Etc... les exemples ne manquent pas. Vous pouvez utiliser aussi (et surtout) les buffers pour vos besoins propres, comme le fait le module MATHS, par exemple.

Toutes ces "choses" ont un point commun: il s'agit de données temporaires, destinées à être détruites ou remplacées à plus ou moins longue échéance. D'autre part, elles n'ont pas de taille fixe.

Ce sont là les caractéristiques principales de ces I/O Buffers. Voyons maintenant leur structure:

### STRUCTURE DES BUFFERS

Les buffers sont placés à la fin des fichiers en mémoire principale. Chaque Buffer est repéré par son ID (Identificateur). Attention: cela n'a rien à voir avec les ID de fichiers LEX. Pour accéder à un buffer, par le biais des routines système ou de la fonction ADBUF\$, il faudra donner son numéro d'identification (exemple 808 en hexadécimal pour STARTUP).

Adresses basses:

:

:

Buffer précédent

-----  
En tête du buffer

n Nombre d'adresses à réactualiser

iii Identificateur (en hexadécimal)

lll Longueur de la zone des données

Zone des données

La "Chose"

-----  
Buffer suivant

:

:

Adresses hautes

Un buffer peut servir, entre autres choses, à stocker des adresses. Or, quand on efface un fichier, par exemple, toutes les adresses ont changé. Et celles qui sont dans les buffers ? Mais oui, elles ont changé aussi.

C'est le rôle du quartet "n" (nombre d'adresses à réactualiser). Quand on stocke des adresses dans le buffer, si on veut qu'elles soient réactualisées quand la mémoire "bouge", on met le quartet n au nombre d'adresses. Le maximum est 15 adresses. Au delà, il faut utiliser un autre buffer.

iii sont les trois quartets qui codent le numéro d'identification.

lll est la longueur en hexadécimal de la zone des données. L'en-tête n'est pas comprise, de telle sorte que lll est nul pour un buffer vide (mais dont l'en-tête existe). La longueur maximum d'un buffer est de 4095 quartets (2047 octets).

Enfin, nous avons la "zone des données". Cette zone est à la responsabilité de l'utilisateur. Chacun organise cette zone comme il l'entend. Pour les buffers utilisés par le système, il faut se reporter au chapitre 12 des IDS No1 pour avoir les formats utilisés.

#### CONFIGURATION ET BUFFERS

Les buffers, nous l'avons dit, sont destinés à des stockages temporaires. Tellement temporaires que le système essaie de les détruire à chaque mise sous tension.

C'est ce qu'on appelle la "configuration": à chaque destruction, copie de fichier, à chaque allumage, le 71 fait le ménage (il répertorie tous les fichiers LEX, remet toute la mémoire en ordre, etc...). En particulier, pour chaque buffer, il positionne le bit de poids fort de l'ID à 0. En fin de configuration, il détruira tous les buffers pour lesquels ce bit est toujours à 0. Vous avez donc compris que pour conserver un buffer (CHARSET par exemple), il faut mettre ce bit à 1 avant que le système ne détruise tout. Ceci se fait en interceptant le poll pCONFIG, et en remettant ce bit à 1. Voilà, le tour est joué.

Conséquence directe de ceci: les numéros d'ID ne peuvent aller que de 800 à FFF.

Autre conséquence directe: une application complète des buffers ne peut être faite en Basic qu'à condition de se forger des outils LEX. Forth, quant à lui, permet de préserver les buffers.

#### ROUTINES UTILES

Pour utiliser les buffers en assembleur, il faut avoir à sa disposition quelques routines du système.

I/OFND permet de localiser un buffer dont l'ID est dans C(X). En sortie, on a D1 pointant sur la zone des données, A(A) contenant la longueur, si la retenue est à 1. C'est la routine utilisée par ADBUF\$.

I/DALL crée un buffer de taille donnée en B(A) et d'ID donné en C(X). Si la retenue est à 0, il n'y avait pas assez de place.

I/ODAL: inverse de la précédente routine. Dans C(X) devait se trouver l'ID. Si la retenue est à 0, le buffer n'a pas été trouvé.

I/OCOL: Remet à 0 la longueur du buffer dont l'ID est dans C(X). S'il n'existait pas, (retenue=0), il est créé avec une longueur nulle.

I/OEXP et I/OCON élargissent et réduisent respectivement la taille du buffer dont l'ID est dans C(X), et d'un nombre de quartets B(A).

Enfin, I/ORES sert à mettre à 1 le bit de poids fort de l'ID. Utile pour la configuration.

Je tiens à préciser que ces renseignements ne sont que partiels. Il faut lire les IDS No2, chapitre 3 pour avoir toutes les informations utiles.

#### EXEMPLES

Ces exemples sont écrits en Basic et nécessitent, pour fonctionner correctement, les fonctions ADBUF\$, REV\$, KEYWAIT\$ et HTA\$ décrites dans le numéro précédent (JPC No 22: Mars 85).

La fonction ADBUF\$ renvoie l'adresse de l'en-tête du buffer (et non du début de la zone des données comme en Forth), ou la chaîne nulle si le buffer

spécifié n'existe pas.

L'ID du buffer est donné en hexadécimal.

Premier exemple: impression de la zone de données d'un buffer.

```
10 DESTROY I$
20 DIM I$(3),L
30 INPUT "Buffer Id ? ";I$
  - Si le buffer n'est pas trouvé, on arrête là.
40 IF NOT LEN(ADBUF$(I$)) THEN END
  - L=Longueur du buffer en quartets.
50 L=HTD(REV$(PEEK$(ADBUF$(I$),7)[5,7]))
60 FOR I=0 TO INT(L/80)-1
70 PRINT PEEK$(DTH$(HTD(ADBUF$(I$))+I*80+7),80)
80 PRINT
90 NEXT I
100 PRINT PEEK$(DTH$(HTD(ADBUF$(I$))+I*80+7),
  MOD(L,80))
110 END
```

Deuxième exemple: lire le STARTUP, et éventuellement le modifier.

```
10 DIM A$(5),S$(96)
20 A$=ADBUF$('808')
  - Si il n'y a pas de STARTUP, on arrête là.
30 IF NOT LEN(A$) THEN 60
  - L=Longueur du buffer en quartets.
40 L=HTD(REV$(PEEK$(A$,7)[5,7]))
  - S$ contient la représentation ASCII de la zone
  données du buffer.
50 S$=HTA$(PEEK$(A$,L+5)[8])
  - On affiche, on modifie éventuellement.
60 LINPUT "Startup ",S$,S$
  - Et on le remet.
70 STARTUP S$
80 END
```

Le dernier exemple est réservé aux membres du SIG (Saturn Interest Group): Il s'agit de faire un catalogue (similaire à CAT ALL), mais pour les fichiers LEX.

En effet, le buffer d'ID "BFC" est le "LEX Entry Buffer", c'est à dire un buffer qui contient les adresses des "Main Table" de tous les fichiers LEX présents en mémoire.

Une fois le programme lancé, les touches suivantes sont redéfinies:

[v]: LEX suivant

[^]: LEX précédent

[>]: affiche l'adresse de la "Main Table"

[<]: revient à l'affichage normal

Toute autre touche termine le programme.

L'affichage normal est: "Id #EF/239 07<tk<2A" ce qui veut dire: LEX d'Id EF en hexadécimal (239 en décimal), qui comporte les tokens de 7 à 2A (en hexadécimal).

Une pression sur [>] entraine l'affichage de "Main Table: A407F", pour dire que la table principale est en A407F (on peut toujours rêver !)

```
10 DIM L$(11),H$(2),K$(4),L,I
20 I=1
30 'BOUCLE':
40 L$=PEEK$(DTH$(HTD(ADBUF$('BFC'))-4+I*11),11)
50 H$=REV$(L$(1,2))
60 DISP 'Id #';H$;'/';STR$(HTD(H$));TAB(15);
  REV$(L$(3,4));'<tk<';REV$(L$(5,6))
70 K$=KEYWAIT$
80 IF K$='#48' THEN DISP 'Main Table: ';
  REV$(L$(7)) @ GOTO 70
90 IF K$='#47' THEN 60
100 IF K$='#50' THEN I=I-(I#1) @ GOTO 'BOUCLE'
110 IF K$='#51' THEN PUT K$ @ END
120 IF L$(1,2)='00' THEN I=I+1
130 GOTO 'BOUCLE'
140 IMAGE "ID #",2A,"/",3D,2X,2A,"<tk<",2A
```

#### CONCLUSION

J'espère vous avoir bien expliqué comment fonctionnent les buffers. Je souhaite ardemment voir très prochainement vos réalisations dans ce domaine... en attendant de vous compter parmi les membres de SIG.

A bientôt

Pierre DAVID (SIG #1)

-----  
Interception de POLL.

- Microprocesseur de Hotel Papa sept unité, me recevez-vous ?
- Hotel Papa de micropro, je vous reçois 5 sur 5, à vous.
- Signal POLL en approche, demande autorisation d'interception.
- Autorisation accordée, passez en assembleur et exécutez votre mission.
- (plus tard) Mission effectuée. POLL intercepté. Hotel Papa pour quitter la fréquence.
- Terminé.

Non, vous ne rêvez pas. Cet échange de messages pourrait bien avoir lieu dans un HP 71. En effet à l'aide de volonté, et à force de Memory Lost, Nous y sommes arrivé: nous avons intercepté un POLL.

Tout d'abord, peut-être serait-il bon d'expliquer ce qu'est un POLL ?

Eh bien, il s'agit d'une sorte de petit message que certains fichiers LEX envoient à l'intention de leurs semblables, pour leur faire savoir qu'ils peuvent intervenir. Voici une explication simple de la chose: les personnes ne possédant pas la ROM Forth/Assembleur peuvent effectuer un ordre LIST sur un fichier TEXT. Il en résultera une erreur: "Type non valide"(merci FRALEX. Notons au passage que FRALEX est un immense Poll Handler) . En revanche, pour ceux qui possèdent la ROM sus nommée, cet ordre fonctionnera parfaitement et listera sans problème le fichier. Que se sera-t-il passé ? La ROM Forth possède-t-elle une fonction LIST qui prend le dessus ? Non, elle ne contient qu'une extension de l'ordre aux fichiers TEXT. Mais, lorsque l'on tente d'exécuter un ordre LIST sur un fichier non BASIC, la machine envoie un POLL vers TOUS les fichiers LEX présents. EDTEXT en est un; donc s'aperçoit de l'erreur et prend le contrôle de la machine, permettant ainsi l'exécution du listage. Il existe ainsi une soixantaine de POLLS qui sont envoyés à différents moment au cours de la vie d'un HP 71. L'un de ceux-là a pour nom pVER\$. Il est envoyé à chaque exécution de VER\$, donnant la possibilité à tous les fichiers LEX de faire parler d'eux. C'est ainsi que mon HP71 m'affiche le message suivant lorsque j'effectue VER\$:

HP71:1BBBB FRA255 FRA001 FRA240 HPIL:1A FTH:1A  
EDT:A KBD:B MATH:1A

Je me suis mis l'autre jour à chercher comment chaque fichier LEX répondait. Et j'ai trouvé. Et cela donne le Poll Handler qui suit et qui permet de faire afficher à la machine le message "ISTRIA" lors de l'appel de la fonction VER\$.

Laurent ISTRIA (PPC-PC #3, S16 #3)

```

LEX 'POLL'
ID #5D
MSG 0
POLL POLHND
ENDTXT
pVER$ EQU #00
POLHND
    SETHEX
    P= 0
    LC(2) pVER$
    ?B=C B
    GOYES CONFIG
    RTNSXM
CONFIG
    CR3EX
    D1=C
    A=R2
    D1=D1- 14
    CD1EX
    ?C>A A
    GOYES SUITE
    CD1EX
    CR3EX
    RTNSXM
SUITE
    CD1EX
    P= 0
    LCASC ' ISTRIA'
    P= 13
    DAT1=C WP
    CD1EX
    CR3EX
    CD1EX
    RTNSXM
END

```

```

-----
REPLACE#

Voici encore un petit LEX d'ID #5C,08. Pour
parfaire le LEX DESAL, il manquait une fonction
alphanumérique.

REPLACE$(A$,B$,C$,CHR$(N))

Remplace dans A$, tous les B$ par des C$. Le
dernier paramètre est optionnel, il définit un
caractère "joker" dans la chaîne B$.

Exemple: REPLACE$("PPC-PARIS CHAPTER","P","(p)")
donne : "(p)(p)C-(p)ARIS CHA(p)TER"

REPLACE$("HP-71B",".P","Hewlett-Packard",".")
donne "Hewlett-Packard-71B"

Ceci est une fonction identique mais simplifiée
à la commande (R) remplacement de l'éditeur de
texte du module Forth/Assembleur. Seulement cette
fois-ci: elle est programmable !

Voilà, avec l'article sur les chaînes, un autre
exemple de manipulation alphanumérique sur la
Math-Stack.

Heureuse Programmation

Michel MARTINET (P#12, S16#2)
-----

LEX 'REPLACE'
ID #5C
MSG 0
POLL 0

```

```

ARGERR EQU #0BF19
POF15 EQU #0BD38
EXPR EQU #0F23C
AVMEMS EQU #2F594
MEMERR EQU #0944D

ENTRY REF
CHAR #F
KEY 'REPLACE#'
TOKEN 8
ENDTXT

Egale ?B=0 A
RTNYES
D0=D0- 2
D1=D1- 2
A=DAT1 B
C=R0
?C#0 XS
GOYES NJok
?A=C B
GOYES Jok
NJok C=DAT0 B
?A#C B
RTNYES
Jok B=B-1 A
B=B-1 A
GONC Egale

Argerr GOVLNG ARGERR

NIBHEX 444434

REP P=C 15
?P# 4
GOYES NJ
GOSBVL POF15
?A=0 A
GOYES NJ
LC(5) 2
?A>C A
GOYES Argerr
C=P 2
C=DAT1 B
R0=C
D1=D1+ 2
GONC REPC

Test C=R3
?C#0 A

```

	GOYES Test2	:	C=A X
	?A=0 A	:	RO=C
	GOYES Exp1	:	
	GOTO Exp	:	TestAB A=R3
Test2	?A=0 A	:	C=R2
	GOYES Exp	:	B=C A
	RTN	:	C=D A
		:	C=C-B A
Exp1	C=R4	:	?C<A A
	D1=C	:	GOYES Stk1
Exp	D1=D1- 16	:	C=D A
	C=RSTK	:	DO=C
	GOTO Expr	:	C=R2
		:	D1=C
A=AVMS	CD1EX	:	D1=D1- 16
	D1=(5) AVMEMS	:	C=R3
	A=DAT1 A	:	B=C A
	D1=C	:	GOSUB Egale
	RTN	:	?B#0 A
		:	GOYES Stk1
NJ	P=P+1	:	C=R4
	C=P 2	:	D1=C
	RO=C	:	D1=D1- 14
		:	C=DAT1 A
REPC	GOSBVL POP15	:	D1=D1- 2
	CD1EX	:	CD1EX
	R4=C	:	A=R1
	C=C+A A	:	C=C-A A
	D1=C	:	CD1EX
		:	A=C A
	GOSBVL POP15	:	C=R4
	CD1EX	:	C=C+A A
	C=C+A A	:	CDOEX
	D1=C	:	B=A A
	R3=A	:	GOSUB A=AVMS
		:	GOSUB Stk
	GOSBVL POP15	:	C=R3
	GOSUB Test	:	D=D-C A
	CD1EX	:	GONC StkC
	R2=C	:	
	C=C+A A	:	Stk1 GOSUB A=AVMS
	D=C A	:	C=D A
	C=0 A	:	DO=C
	R1=C	:	C=R1
	C=RO	:	B=C A
	A=C X	:	C=R4
	CDOEX	:	C=C-B A
	CSL W	:	D1=C
	CSL W	:	D1=D1- 16
	CSL W	:	LC(5) 2

```

B=C A
GOSUB Stk
CDOEX
D=C A
Stk C=R2
?C=D A
GOYES Fin
GOTO TestAB

```

```

Stk CD1EX
C=C-B A
?A>C A
GOYES Memerr
C=C+B A
D1=C

```

```

Boucle ?B=0 A
RTNYES
D0=D0- 2
D1=D1- 2
C=DAT0 B
DAT1=C B
C=R1
C=C+1 A
C=C+1 A
R1=C
B=B-1 A
B=B-1 A
GONC Boucle

```

```

Memerr GOVLNG MEMERR

```

```

Fin C=R1
B=C A
C=R4
C=C-B A
D1=C
D1=D1- 16
D1=D1- 16
CD1EX
?A>C A
GOYES Memerr
D1=C
C=R0
CSR W
CSR W
CSR W
D0=C
C=B A
CSL W
CSL W

```

```

LDHEX F
DAT1=C 7
Expr GOVLNG EXPR
END

```

FILE ?

En dernière minute, Stéphane BARIZIEN nous a fait parvenir ce fichier LEX.

FILE? (nom de fichier) donne  
1 si le fichier est présent en mémoire,  
0 sinon.

Stéphane BARIZIEN (SIG #?)

```

LEX 'FILELEX'
ID #50
MSG 0
POLL 0
ENTRY RUN
CHAR #F
KEY 'FILE?'
TOKEN 240
ENDTXT
BSERR EQU #0939A
FILEXQ$ EQU #09B95
FINDF EQU #09F77
EXPR EQU #0F23C
NIBHEX 411
RUN CDOEX
RSTK=C
GOSBVL FILEXQ$

```

```

        GONC Fspce
        ?A#0 W
        GOYES OK
Fspce  GOVLNG BSERR

OK      CD1EX
        RSTK=C
        GOSBVL FINDF
        A=0 W
        GOC RTN
        A=A+1 S
        ASR W
RTN     C=RSTK
        D1=C
        D1=D1- 16
        DAT1=A W
        C=RSTK
        DO=C
        GOVLNG EXPR
        END

```

PEEK POKE PARTOUT (S.I.G.)

Voici, en attendant la fonction MNEMO\$(A\$), (on y comprend rien, vraiment rien !) un petit LEX d'ID 01 - TOKEN 63/64 qui ne sont autres que les fonctions PEEK\$ et POKE du HP71B. Avec ce LEX dans la mémoire de Titan, vous "Peekerez" et "Pokerez" où vous voudrez dans toute la mémoire (00000 à FFFFF).

Exemple:

```

SUB DEPRIVE(A$)
POKE DTH$(HTD(ADDR$(A$))+20),"00"
END SUB

```

Avec PEEK\$, ADBUF\$ et POKE, nous avons les outils du parfait "Bidouilleur" (Sig'man).

SIGement votre

Laurent ISTRIA, Michel MARTINET et Pierre DAVID.  
SIG #3, #2 et #1 ... Brand New !!!

SATURN Interest Group. (S.I.G. & Co. Ltd.)

(c) - Copyright PPC-Paris 03/1985.

BIP TOUT LE TEMPS !

Si vraiment vous ne savez plus quoi faire pour embêter votre entourage, j'ai ce qu'il vous faut ! Assemblez le petit LEX que je vous propose, et je suis sûr qu'on vous mettra à la porte en moins de temps qu'il faut pour le dire...

Le but de ce petit fichier, donc, est de faire un petit BIP à chaque pression de touche. Rien de bien excitant, si ce n'est la facilité de réalisation (et encore !).

J'explique, pour ceux qui trouveraient ceci encore trop difficile:

Il suffit d'intercepter le poll "Définition de touche". Chaque fois que vous appuyez sur une touche, vous ne vous en rendez peut-être pas compte, mais ce satané HP71 fait le tour de chaque fichier LEX présent en mémoire, en demandant si on veut redéfinir la touche. En théorie, cela sert surtout pour des touches telles que SST, BACK, -CHAR ... qui ont une action immédiate, et ne correspondent pas à un ordre BASIC.

On redonne ensuite la main à Basic, en disant n'avoir rien fait (XM=0). Ni vu, ni connu etc...

Le programme se contente donc d'intercepter le sus-dit poll, puis de faire un beep à 1000 Hz pendant 32 millisecondes. C'est tout ! et c'est très énervant. Tellement énervant que je me demande si je ne vais pas purger ce fichier. Car en plus, il ralentit le fonctionnement du tout ! Mais oui. Heureusement qu'il y a BEEP OFF. Ouf...

A bientôt  
Pierre DAVID (SIG #1)

ATTENTION !

De plus en plus fort !

Vous avez peut-être vu fleurir dans les programmes Basic quelques affreux POKE "2F441", "F" et POKE "2F441", "0". Vous avez certainement pesté contre le manque de lisibilité de cette syntaxe.

Heureusement, nous arrivons à point pour vous délivrer de ces angoisses Basikéennes. Nous proposons à votre sagacité habituelle LE mot "ATTN".

Sous trois formes différentes, il éclaircira vos programmes mieux que ne saurait le faire l'étoile la plus brillante au firmament. (snif !)

Plus prosaïquement:

ATTN OFF désactive la touche ATTN en mode programme. Attention, elle est active durant les ordres INPUT et LINPUT.

ATTN DN remet les choses en place.

Et enfin, devant vos yeux ébahis, nous révélons pour vous, en exclusivité pour vous et vous seulement l'existence de ATTN (tout simplement), qui a pour effet de passer d'un mode à l'autre.

Heureuse Bientôt,  
et à Programmation !

Pierre DAVID et Michel MARTINET  
SIG #1 et #2, PPC-PC #37 et #12  
Saturn Interest Group

MANIPULATION DES CHAINES ALPHANUMERIQUES  
SUR LA "MATH STACK" DU HP71B

Durant l'exécution d'une fonction, ses paramètres sont placés dans un endroit de la mémoire appelé Math Stack où les opérations se déroulent lorsque le système exécute une fonction BASIC.

La représentation interne des chaînes alphanumériques y est assez singulière, en effet tout est sens dessus-dessous. Les octets sont représentés à l'envers et les chaînes sont tantôt dans le bon sens tantôt dans le sens inverse.

Exemple: avec le module FORTH/ASSEMBLER, lorsque vous exécutez PEEK\$("E7BB0",12), vous obtenez le résultat suivant: 2545E43585D4. En clair, nous nous trouvons dans la table de codage des anémiques du module F/A et cela signifie RTNSXM. 25 pour R (52 en hexa), 45 pour T (54 en hexa), ..., D4 pour M (4D en hexa). Les quartets sont inversés deux à deux mais la chaîne est dans le bon sens. Jusqu'ici, tout va bien, il suffit de savoir que les octets successifs représentant les chaînes sont codés à l'envers et le tour est joué: nous sommes alors capables de comprendre toutes les chaînes alphanumériques se trouvant dans la mémoire de notre Titan préféré.

Sur la Math Stack, l'inversion des quartets deux à deux est la même mais de plus, la chaîne est entièrement inversée. Le premier caractère se trouve en dernière position et le dernier au début.

Exemple: la même chaîne y serait représentée sous la forme D48535E44525 soit "MXSNTR". Vous comprenez aisément la facilité avec laquelle nous sommes mis à faire de l'assembleur au Club.

LA MATH STACK:

Les chaînes y sont codées en deux parties, la première de longueur fixe (16 quartets) représente l'en-tête et la deuxième la chaîne proprement etid.

L'en-tête: les 16 quartets sont divisés en 4

parties. Le préfixe FO (2 quartets), toujours égal à lui-même, la longueur de la chaîne en quartets lllll (5 quartets), l'adresse de la chaîne s'il s'agit d'une variable alphanumérique aaaaa (5 quartets) et enfin la longueur maxi de la chaîne s'il s'agit là aussi d'une variable alphanumérique mmm (4 quartets). Notez enfin que sur la Math Stack, seuls le préfixe et la longueur de la chaîne sont utilisés.

Exemple: la chaîne "ABCDE"

VOIR FIGURE (I)

La chaîne: est codée à l'envers comme décrite précédemment. Noter qu'une chaîne nulle est tout de même représentable:

! FO ! 00000 ! aaaaaa !

#### MANIPULATION DES CHAINES AVEC L'ASSEMBLEUR

Définition du nombre de paramètres de la fonction.

Avant d'exécuter une fonction, le HP71B étudie, le nombre et le type de paramètres que celle-ci admet, et place dans C(S) (champ S du registre C) le nombre exact de paramètres se trouvant sur la Math Stack lorsque la fonction admet des paramètres optionnels.

Type

2 tableaux (utilisé en addition avec 4 ou 8)  
 4 paramètre numérique  
 6 tableau alphanumérique (4+2)  
 8 paramètre alphanumérique  
 A tableau numérique (8+2)

Exemples: la fonction PEEK\$ admet 2 paramètres, 1 alphanumérique et 1 numérique. Ceci est codé comme suit:

PEEK\$(A\$,N)

NIBHEX 8422

8 paramètre numérique  
 4 paramètre alphanumérique  
 2 nombre mini de paramètres  
 2 nombre maxi de paramètres

POS(A\$,B\$,N)

NIBHEX 84423

8 paramètre numérique  
 4 paramètre alphanumérique  
 4 paramètre alphanumérique  
 2 nombre mini de paramètres  
 3 nombre maxi de paramètres (1 optionnel,  
 C(S)=nombre exact de paramètres)

KEYWAIT\$

NIBHEX 00

0 nombre mini  
 0 nombre maxi

Exemple: la fonction ASC\$(A\$) (voir JPC Mars)

NIBHEX 411

Le point d'entrée de la fonction se trouve au label ASCII. Avant d'exécuter la suite des opérations, le HP71B étudie le nombre de paramètres que celle-ci admet en remontant en avant du point d'entrée. Le premier 1 rencontré définit le nombre maxi que peut accepter la fonction. Le deuxième le nombre mini et le 4 définit un paramètre alphanumérique. Le HP71B place les paramètres sur la Math Stack et D1 pointe le début du premier d'entre eux.

ASCII CD1EX

Sauvegarde de la valeur de D1 (adresse du sommet de la Math Stack) en échangeant D1 avec C(A).

D1=C

Restitution.

CDEX A

Echange de C(A) avec D(A), le registre D n'est pas utilisé par la suite, on y sauvegarde D1.

GOSBVL POP1S

Au retour de la routine POP1S, A(A) contient la longueur de la chaîne en quartets, P=0 et D1 pointe la fin de l'en-tête (le dernier caractère de la chaîne D1=D1+ 16).

ASCII1 ?A=0 A

GOYES ASCII4



Voilà, un article "explicatif" sur l'assembleur  
 du HP71B. Pourquoi n'ouvririons-nous pas une  
 rubrique sur SATURN en nous exposant mutuellement  
 (même si cela est du déjà vu) tout nos travaux ?

Michel MARTINET (P#12 E16#2)

FIGURE (I)

```

paramètre suivant : FO : A0000 : aaaaaaaaa : 5444342414 : paramètre précédent
(adresse basse)  :<----->:<----->: (adresse haute)
      En-tête           Chaîne
      :<-->:           :           :
      Préfixe
      : <----->:           :           :
      Longueur
      :           :<----->:           :
      Adresse et Longueur maxi (non utilisées)
      :           : <XXXXXX> :           :
                        E D C B A
  
```

```

LEX  'PKLEX'
ID   #01
MSG  0
POLL 0
  
```

```

REVPDP EQU #0BD31  * REV$ et POP15
ARGERR EQU #0BF19  * Renvoie l'erreur "Invalid Arg"
DRANGE EQU #1B076  * Teste si un octet est dans l'intervalle [0..9]
  
```

```

CONVUC EQU #152AA * "CONVert to Upper Case"
RANGE EQU #1B07C * Teste si un octet est dans un certain intervalle
POP1R EQU #0E8FD * Teste un nombre réel sur la math stack et renvoie une erreur si non.
FLTDH EQU #1B223 * Convertit un nombre sur 12 chiffres en un entier Hexa.
HEXASC EQU #1714B * Convertit de l'Hexa en Ascii.
EXPR EQU #0F23C * Point de retour des fonctions alpha.
REV$ EQU #1B3BE * Inverse une chaîne sur la pile.
F-R0-0 EQU #2FB9B * RAM réservée, Zone "scratch" pour les fonctions
F-R0-1 EQU #2FBA0 * Dito mais 3 octets plus loin
STRHDR EQU #0F09A * "String Header"
NXTSTM EQU #0BA4B * "scan to NeXT StateMent/jump to BASIC loop"
S-R0-0 EQU #2F871 * RAM réservée, zone "scratch" pour les ordres Basic.
POP1S EQU #0BD38 * Déjà vue
FIXDC EQU #05493 * "Expression list decompile"
POKEP EQU #0378E * Routine d'analyse de l'ordre POKE
EXPEX- EQU #0F17B * Evalue l'expression
TFORN EQU #2F59E * RAM:Zone de sauvegarde du pointeur de pile (Math Stack)
EXPEXC EQU #0F186 * Evalue l'expression (pas la même)

```

```

ENTRY PEEK
CHAR #F
ENTRY POK
CHAR #D

```

```

KEY 'PEEK$'
TOKEN 63
KEY 'POKE'
TOKEN 64

```

```

ENDTXT

```

```

REL(5) POKEd
REL(5) POKEp
POK GOSBVL EXPEXC
GOSUB Addrck
D1=(5) S-R0-0
A=B A
DAT1=A A
D0=D0+ 2
GOSBVL EXPEX-
GOSBVL POP1S
P= 5
A=0 P
P= 0
ASRB
D0=(5) S-R0-0
C=A A
D=C A
A=DAT0 A
D1=A

```

```

DO=(5) TFORM
A=DATO A
DO=A
POKE1 D=D-1 A
      GDC POKE2
      DO=DO- 2
      C=0 S
      A=DATO B
      GOSUB Aschex
      A=B A
      DAT1=A 1
      D1=D1+ 1
      GONC POKE1
POKE2 GOVLNG NXTSTM

POKEd GOVLNG FIXDC  * FIXDC: routine de décompilation des ordres : PRINT, DISP
                    * POKE, FIX, SCI, ENG, FLAG, DELAY, WAIT, INPUT et READ

POKEp GOVLNG POKEP  * POKEP: routine d'analyse de l'ordre POKE (celui de HP)

```

\* ADDRCK est une routine interne de la version HP71B:1BBBB. Adresse: 1C5A5  
 \* Ce point d'entrée n'est pas supporté, c'est-à-dire que les modifications  
 \* ultérieures peuvent altérer cette adresse dans de futures versions (:1CCCC etc.).  
 \* Pour la compatibilité, nous avons inséré la routine "Addrck"

```

Addrck GOSBVL REVPOP  *
      C=0 A           *
      LCHEX A         * A
      ?A>C A         *
      GDYES Err      * D
      C=A A           *
      P=C 0           * D
      P=P-1          *
      GDC Err        * R
      ADIEX          *
      C=C+A A        * C
      D1=A           *
      A=0 W           * K
      A=DAT1 WP      *
      D1=C           *
      B=0 A           *
      P= 0           *
Addr  GOSUB Aschex  *
      ?A=0 B         *
      RTNYES         *
      GONC Addr      *

Err   GOVLNG ARGERR * Commun à ADDRCK et ASCHEX

```

\* routine Acshex, non supportée par Hewlett-Packard.  
 \* Adresse 1C5E3 dans la version :1BBBB  
 \* Voir ADDRCK.

```

Aschex  GOSBVL LRANGE  *
        GDC  ATH      *
ATH1    BSL  W        *
        B=A  P        * A
        ASR  W        *
        ASR  W        * S
        RTN                *
                * C
ATH     GOSBVL CONVUC  *
        LCASC 'FA'    * H
        GOSBVL RANGE  *
        GDC  Err      * E
        LCHEX 37      *
        A=A-C B      * X
        GDC  ATH1     *

        NIPHEX 8422
PEEK    GOSBVL POP1R
        D1=D1+ 16
        CDOEX
        D0=(5) F-R0-0
        DAT0=C A
        GOSBVL FLTDH
        GDC  Err
        C=A  A
        C=C+C A
        GDC  Err
        R0=A
        GOSUB Addrck
        CD1EX
        D1=(5) F-R0-1
        DAT1=C A
        C=R0
        D=C  A
        A=C  A
        C=C+B A
        R1=C
        C=B  A
        R0=C
        A=A+A A
        D1=(5) F-R0-1
        C=DAT1 A
        D1=C
        C=A  A
        GOSBVL STRHDR
  
```

```

A=R0
D0=A
PEEK1 D=D-1 A
      GOC PEEK2
      C=0 S
      A=DAT0 1
      GOSBVL HEXASC
      DAT1=A B
      D1=D1+ 2
      D0=D0+ 1
      GOTO PEEK1

PEEK2 C=R1
      D1=C
      P= 0
      GOSBVL REV#
      D0=(5) F-R0-0
      C=DAT0 A
      D0=C
      GOVLNG EXPR

END

```

---

```

LEX 'KBEEP' Key BEEP . Pourquoi pas ?
ID #5C Ca ou un autre...
MSG 0
POLL POLHND C'est ce qui fait l'intérêt de ce LEX
ENDTXT Aucun mot clef

pKYDF EQU #1B Poll "Key DeFinition"
BP+C EQU #0EB40 Routine Système pour biper
Freq EQU #3EB Fréquence (1000 Hz)
Duree EQU #20 Durée. Vous pouvez essayer la valeur que vous voulez.

POLHND LC(2) pKYDF C(B)=No du Poll qu'on tente d'intercepter
      ?B=C B Si c'est celui là,
      GOYES POLL on le prend,
      RTNSXM sinon, on ne fait rien.

POLL C=D A Tout le travail du programme
      R3=C est de sauver les bons
      SETHEX registres, et de les
      P= 0 remettre après.
      LC(5) Freq
      B=C A

```

LC(5) Duree  
 GOSBVL BP+C Beep de durée C(A), de frequence D(A).  
 C=R3 On remet les bonnes  
 D=C A choses à leur place,  
 RTNSXM et on revient.  
  
 END Voila, c'est fini !

---

LEX 'ATTNLEX'  
 ID #5C  
 MSG 0  
 POLL 0

ENTRY ATTNe execution de ATTN  
 CHAR #D Ordre, exécutable au clavier, légal après THEN ELSE, programmable.

KEY 'ATTN'  
 TOKEN 33 Pourquoi pas celui-là ?

ENDTXT

tON EQU #E0 Token (code) de "ON"  
 tOFF EQU #E1 Token (code) de "OFF"  
 ATNDIS EQU #2F441 ATtN DISable: un quartet non nul empêche ATTN de tout stopper.  
 NXTSTM EQU #0BA4B Achève l'exécution de l'ordre, et revient à Basic.  
 WRDSCN EQU #02C2A Teste si la chaîne entrée contient un code contenu dans une table.  
 RESPTR EQU #03172 Remet D1 à sa position initiale.  
 FINDA EQU #023E3 ON GOTO suivant une table.  
 DROPDC EQU #05470 Routine de décompilation de DROP et ADD.  
 TACDC EQU #052FC Routine de décompilation de TRACE.

REL(5) ATTNd Adresse de la routine de décompilation.  
 REL(5) ATTNp Adresse de la routine d'analyse.

ATTNe AD1EX Sauvegarde D1 dans A  
 D1=(5) ATNDIS  
 C=0 S C(S) contient la valeur à mettre dans ATNDIS  
 B=A A On sauvegarde D1 dans B(A)  
 A=DATO B A(B)=token suivant ATTN  
 LC(2) tON  
 ?A=C B Si c'est ON  
 GOYES AO on met 0  
  
 C=C+1 B C(B)=tOFF (en fait: (tON)+1)  
 ?A=C B

```

GOYES AF      Si c'est celui-là, on met F

C=DAT1 P      Sinon, on agit comme une bascule.
?C#0 P        et si ATNDIS # 0 alors
GOYES A0      on met 0

      *
      A ce point, il faut se souvenir que C(S)=0 (n'a jamais été modifié).
AF  SETHEX
    C=C-1 S    Ruse: C(S)=F (et la retenue vaut 1)
AO  DAT1=C S   ATNDIS <- C(S)
    A=B A
    AD1EX      On restore D1
    GOVLNG NXTSTM Et on revient à BASIC.

ATTNp GOSBVL WRDSCN  Teste la chaîne entrée,
      CON(2) tON     si c'est ON,
      REL(3) Rcc     on va en Rcc,
      CON(2) tOFF    si c'est OFF,
      REL(3) Rcc     on va en Rcc,
      NIBHEX 00      Fin de la table.
      GOVLNG RESPTR Si ce n'est aucun de ceux-là, on repasse le contrôle à Basic, avec une
      *              erreur si aucun autre LEX n'a reconnu ce qui suit.
Rcc  RTNCC       Retour à l'interpréteur, tout est Ok.

ATTNd GOSBVL FINDA  <=> ON GOTO
      CON(2) tON     Si c'est tON,
      REL(3) tracdc  on va en tracdc.
      CON(2) tOFF
      REL(3) tracdc
      NIBHEX 00      Fin de la table.
      GOVLNG DROPDC routine de décompilation de ADD et DROP.

tracdc GOVLNG TRACDC routine de décompilation de TRACE.

      END

```

---

#### CHAINAGE DE FICHER LEX

Vous avez certainement constaté, en parcourant ces lignes, et en faisant vous même vos propres fonctions Basic qu'il existait un problème: le nombre grandissant de fichiers LEX et le nombre restreint de numéros ID et Token aboutit à une cacophonie de LEX.

Nous avons réfléchi au problème. Nous en reparlerons ultérieurement, et de manière plus approfondie. Pour le moment, il faut faire des petits fichiers, modifiables aisément. Lorsque l'on voudra les utiliser, il suffira de se faire UN fichier LEX contenant ceux dont on a besoin.

Mais seulement, comment créer ce fichier ?

C'est pour cela que j'ai créé "CL" (Chain Lex), en partant d'une idée de Jean-Jacques MOREAU (cf JPC No 22 Mars 85).

Ce programme demande le nom de deux fichiers LEX, et les réunit en un seul. Pour arriver à ce résultat, le programme crée le port indépendant No 0. Il est nécessaire d'avoir le LEX "PKLEX" et la fonction "REV\$" pour procéder.

#### PRINCIPE:

Le port 0 doit être libre (CLAIM PORT) avant exécution.

Le programme reconstitue dans le port 0 les deux LEX, chaînés, détruit les deux originaux dans la mémoire principale, recopie le LEX chaîné, et détruit le port 0.

Joyeux chainages !

Michel MARTINET

---

```
1 INPUT "LEX 1,2 :";N$,M$ @ DEF KEY "F*", "FREE:PORT(0)@RUNCL,A": @ USER DN
2 PUT "F*" @ STOP @ 'A': DEF KEY "F*", "CLAIM:PORT(0)@RUNCL,B": @ CREATE TEXT Z:PORT(0)
3 C$=ADDR$("Z") @ A1=HTD(ADDR$(N$)) @ A0=A1+32 @ A2=HTD(ADDR$(M$))
4 L1=32+HTD(REV$(PEEK$(DTH$(A0),5))) @ L2=HTD(REV$(PEEK$(DTH$(A2+32),5)))-5 @ F1=A1+L1
5 A0=A0+11 @ 'Z': D=HTD(REV$(PEEK$(DTH$(A0),5))) @ IF 0 THEN A0=A0+0+6 @ GOTO 'Z'
6 D=F1-A0 @ A0=A0-A1 @ POKE C$,PEEK$(DTH$(A1),L1)&PEEK$(DTH$(A2+37),L2)
7 POKE DTH$(HTD(C$)+32),REV$(DTH$(L1-32+L2)) @ POKE DTH$(HTD(C$)+A0),REV$(DTH$(D)) @ PURGE N$
8 PURGE M$ @ COPY N$ @ PUT "F*" @ STOP @ 'B': DEF KEY "F*" @ USER OFF @ CAT N$ @ END ALL
```

## CHOC EN RETOUR

Un certain nombre d'erreurs se sont infiltrées dans les derniers JPC.

Dans le No 20 (Décembre 84-Janvier 85):

page 46: en bas, il faut lire "remplacez aussi INF par MAXREAL aux lignes 740,, 560 et 930.

Page 64: il faut lire

```
2860 IF POS('ES',A$(10,10)) THEN UNSECURE  
FNS$(N$,D$) ELSE SECURE FNS$(N$,D$)
```

```
2565 IF Z=57 OR Z=1057 THEN DISP MSG$(67) @ RETURN
```

Dans le No 21 (Février 85):

Page 39: 3160 FDR J=N TD N/2 STEP -1

D'autre part, comme vous l'avez peut être remarqué, dans le No 20 (Décembre 84/Janvier 85), le deuxième B=1 de la ligne 110 n'a pas lieu d'être. Encore quelques demi-octets récupérés !

Pour ceux que ce genre de problèmes intéresse, je signale que la version que j'utilise fait 185 (cent quatre vingt cinq) octets, tout en conservant les mêmes fonctionnalités (Sous programme, précision ajustable, même format de sortie...).

Je ne prétends pas détenir la version la plus courte. Un concours est ouvert: qui fera mieux ?

Le premier prix sera la considération des membres de PPC-PARIS !

Si vous voyez des erreurs, des oublis, ou des imprécisions dans votre Journal favori, n'hésitez plus un instant: communiquez les nous, ne les gardez pas pour vous. Vous nous faciliterez ainsi la tâche, et celle de ceux qui lisent le Journal.

Merci d'avance,  
Pierre DAVID

## INTEGRATION NUMERIQUE DE GAUSS

Pour ceux qui ne possèdent pas (encore) le module MATH du HP71B, mais néanmoins désireux de se concocter des intégrales d'enfer, je leur propose, pour pallier à ce petit inconvénient, une petite routine qui donne de très bons résultats.

La routine "INTG" (INTégration) utilise la méthode numérique de GAUSS en 12 constantes. Elle peut d'ailleurs être aisément étendue à 24, voire 96 constantes. A chacun de voir !

Toutes les fonctions peuvent être intégrées, de a à b, de -Inf à a, ou de a à +Inf; Seule restriction: pour intégrer de -Inf à +Inf, deux intégrations sont à prévoir: par exemple de -Inf à 0, puis de 0 à +Inf.

Sans plus tarder, un exemple:

```
CALL INTG
```

```
F(X)= EXP(X)
```

```
a,b= -Inf,0
```

```
I= 1.0000
```

```
ou encore:
```

```
CALL INTG
```

```
F(X)= LOG(X)/(1+X*X)
```

```
a,b= 0,Inf
```

```
I= 2.800E-12 (autrement dit: Zéro !)
```

Daniel JACOB (P11)

## UTILITAIRES

Voici deux petits utilitaires qui, j'en suis sûr, vous rendront la vie plus facile.

Le premier, utilitaire d'édition, vous permet de

lister un fichier, mais en mettant des commentaires utiles pour l'archivage, ou la simple mise au point d'un programme: quelle est la bonne version, parmi tous mes listages ? Ne cherchez plus, mon programme imprimera le nom du fichier, la date et l'heure du listage, ainsi que la place mémoire disponible à ce moment.

L'autre utilitaire permet de faire le ménage dans la mémoire. Vous effacerez tous les fichiers jusqu'à ce que vous donniez le nom "EOPURGE" (End Of PURGE).

A la fin, deux petits messages vous donneront des informations sur l'état de votre machine.

Jean Claude Fourès

---

#### VISUALISATION DE LA RAM

Si vous voulez partir en chasse contre les quartets tapis au fond de votre HP71, il vous faudra utiliser les bonnes armes que constituent une imprimante, et mon programme: j'ai nommé "MEMDUMPX". C'est une adaptation d'un programme pour HP75 paru dans Micro-Revue. Il sort un tableau sur l'imprimante (80 colonnes).

L'utilisation n'est pas trop dure: laissez-vous guider par les questions du programme.

Bon vidage de mémoire !  
Philippe DAVASE

---

DOW, JDS, DATE, HEURE

Voici quelques utilitaires relatifs au calendrier:  
DOW: donne le jour de la semaine au clavier  
JDS: donne le jour de la semaine par programme (sous programme)  
DATE: affiche la date courante  
HEURE: affiche l'heure courante.

Quelques exemples:

CALL DOW  
JJ,MM,AAAA: 26,5,1965 (ma date de naissance)  
Mercredi  
CALL JDS(26,5,1965,5\$) @ DISP 5\$  
Mercredi  
CALL DATE  
Samedi 9/ 3/1985

Daniel JACOB (P11)

---

#### LE DEBILOMETRE

Amis des sourires idiots et des rongeurs, bonjour. En l'honneur du 1er Avril, je vous présente LE programme de DAO sur HP-71.

Ce programme est un débilomètre de 10 Ko, qui vous renseignera sur votre santé.

Tous les commentaires sont sous forme de DATA.

Je vous laisse donc avec ce programme, et vous rappelle que toute réclamation est à adresser au Journal.

Serge VAUDENAY

INTG (Integration)

```
=====
10 DATA .9815606342,.04717533639,.9041172564,.106939326,.7699026742,.1600783285,.5873179543
20 DATA .2031674267,.367831479,.2334925365,.1252334085,.2491470458
=====
```

```
30 DESTROY I @ RESTORE @ CFLAG 0,1,2
40 LINPUT "F(X)= ",G$;G$ @ INPUT "a,b= ";A,B
50 IF ABS(A)=INF OR ABS(B)=INF THEN 140 ELSE C=(B+A)/2 @ B=(B-A)/2
60 FOR D=1 TO 6 @ READ E,F @ IF FLAG(1) THEN 90
70 X=C+(-1)^FLAG(0)*B*E @ I=I+VAL(G$)
80 IF FLAG(0,NOT FLAG(0))=0 THEN 70 ELSE 120
90 V=(-1)^FLAG(0) @ X=C+2/(1+E*V) @ IF FLAG(2) THEN X=-X
100 I=I+F*VAL(G$)/(1+E*V)^2
110 IF FLAG(0,NOT FLAG(0))=0 THEN 90
120 NEXT D @ IF FLAG(1) THEN I=I*2 ELSE I=I*B
130 CFLAG 0,1,2 @ FIX 4 @ DISP "I=";I @ STD @ BEEP 999,.1 @ END
140 SFLAG 1 @ IF A=-INF THEN SFLAG 2 @ C=-1-B @ GOTO 60 ELSE C=A-1 @ GOTO 60
```

=====

UTIL (Utilitaires)

```
=====
1000 SUB PLIST
  - Utilitaire édition (ECR)
1010 INPUT "Fichier: ";A$
1020 PRINT "Date =";DATE$,"Heure =";TIME$
1030 PRINT
1040 PLIST A$ @ DESTROY ALL @ A=MEM
1050 PRINT
1060 PRINT "Taille mémoire disponible =";A
1070 PRINT
1080 PRINT
1090 PRINT
1100 END
```

```
=====
2000 SUB PURGE
  - Purger
2010 DESTROY ALL
2020 INPUT "Fichier: ";A$
2030 IF A$="EOPURGE" THEN GOTO 'FIN'
2040 A=A+1 @ PURGE A$ @ GOTO 2020
```

```
=====
2050 'FIN': PRINT "Nombre de fichier(s) effacé(s)=";A @ A=MEM
```



```
460 PRINT L$(I,80)
470 RETURN
```

=====  
DOW (Jour de la semaine)

```
10 INPUT "JJ,MM,AAAA: ";J,M,A
20 CALL JDS(J,M,A,S$) @ DISP S$ @ END
```

```
=====  
30 SUB JDS(A,B,C,E$)
```

```
=====  
40 DATA Vendredi,Samedi,Dimanche,Lundi,Mardi,Mercredi,Jeudi
```

```
=====  
50 RESTORE @ E=IP(.7+1/(B+1))
60 N=IP((B+1+12*E)*30.6001)+IP(365.25*(C-E))+A
70 FOR I=0 TO MOD(N,7) @ READ E$ @ NEXT I
80 END SUB
```

```
=====  
90 SUB DATE
100 A$=DATE$ @ J=VAL(A$(7)) @ M=VAL(A$(4,5)) @ A=VAL(A$(1,2))+1900
110 CALL JDS(J,M,A,J$) @ DISP USING "BA,X,2(2D, '/') ,4D";J$,J,M,A @ END
```

```
=====  
120 SUB HEURE
130 DELAY 0,0 @ DISP TIME$
140 IF NOT KEYDOWN THEN 130 ELSE END
```

=====  
LE Debilometre

```
- Debilometre.
2 POKE '2F441','F'
3 DESTROY ALL @ DIM A$(96) @ FOR A=1 TO 12 @ READ B @ A$=A$&CHR$(B) @ NEXT A @ CHARSET A$
```

```
=====  
4 DATA 56,84,86,85,24,0,56,85,86,84,24,0
```

```
=====  
5 DISP "-- D'&CHR$(128)&'bilom'&CHR$(129)&'tre... --"
6 USER ON @ RADIANS @ SFLAG 0,1,2,3,4,-25 @ DELAY 0,0 @ OPTION BASE 0
7 BEEP 349,.4
8 BEEP 349,.2
```

```

9 BEEP 440,.2
10 BEEP 440,.2
11 BEEP 392,.2
12 BEEP 440,.2
13 BEEP 392,.2
14 BEEP 349,.4
15 BEEP 349,.2
16 BEEP 440,.2
17 BEEP 440,.2
18 BEEP 392,.2
19 BEEP 100,.5
20 FOR A=5 TO 1 STEP -1 @ DISP TAB(B);'-';A;'-' @ WAIT .5 @ BEEP 10000,.05 @ NEXT A
21 DISP TAB(B);'-';0;'-' @ FOR A=1 TO 20 @ BEEP 1.5^A,A/30 @ NEXT A
22 DEF KEY '#43','' @ DEF KEY '#50','' @ DEF KEY '#51','' @ DEF KEY '#99',''
23 DEF KEY '#f0','' @ DEF KEY '#g0',''
24 CONTRAST 0 @ CFLAG 0,1,2,3,4,-25 @ DEGREES @ DISP
25 A$=' ' @ FOR A=1 TO 76 @ READ B @ A$=A$&CHR$(B) @ NEXT A

```

```

=====
26 DATA 80,79,75,69,39,50,70,52,52,49,39,44,39,70,39,64,73,78,80
27 DATA 85,84,32,39,80,97,115,115,119,111,114,100,32,63,32,39,59,66,36,64
28 DATA 80,79,75,69,39,50,70,52,52,49,39,44,39,48,39,64,39,77,101,109
29 DATA 111,114,121,32,108,111,115,116,39,64,85,83,69,82,79,70,70

```

```

=====
30 CONTRAST 15 @ CONTRAST 9 @ STARTUP A$
31 DELAY 1 @ GOSUB 178 @ BEEP

```

```

=====
32 DATA 69,116,101,115,45,118,111,117,115,32,100,128,98,105,108,101,32,63,63,63
33 DATA 13,0
34 DATA 67,101,32,112,114,111,103,114,97,109,109,101,44,32,100,101,115,32,112,108
35 DATA 117,115,32,115,128,114,105,101,117,120,44,32,118,111,117,115,32,97,105,100
36 DATA 101,114,97,32,100,97,110,115,32,99,101,32,103,114,97,118,101,32,112,114
37 DATA 111,98,108,129,109,101,46,13,0
38 DATA 86,111,117,115,32,100,101,118,114,101,122,32,115,105,109,112,108,101,109,101
39 DATA 110,116,32,115,128,108,101,99,116,105,111,110,110,101,114,32,118,111,115,32
40 DATA 114,128,112,111,110,115,101,115,32,97,118,101,99,32,108,101,115,32,102,108
41 DATA 129,99,104,101,115,32,118,101,114,116,105,99,97,108,101,115,44,13,0
42 DATA 101,116,32,108,101,115,32,118,97,108,105,100,101,114,32,112,97,114,32,69
43 DATA 78,68,76,73,78,69,46,13,0
44 DATA 66,111,110,32,99,111,117,114,97,103,101,46,46,46,13,0
45 DATA 76,111,114,115,113,117,101,32,118,111,117,115,32,118,111,121,101,122,32,113
46 DATA 117,101,108,113,117,39,117,110,32,115,101,32,114,101,112,111,115,101,114,44
47 DATA 32,46,46,46,13,0
48 DATA 49,45,32,86,111,117,115,32,108,39,97,105,100,101,122,46,13,3
49 DATA 50,45,32,86,111,117,115,32,108,39,101,110,118,105,101,122,46,13,2
50 DATA 51,45,32,86,111,117,115,32,108,101,32,109,128,112,114,105,115,101,122,46
51 DATA 13,1
52 DATA 73,109,97,103,105,110,101,122,32,113,117,101,32,121,111,117,32,104,97,115
53 DATA 116,32,108,24,115,116,105,110,103,25,101,110,32,106,97,32,14,15,4,103
54 DATA 9,32,122,104,117,114,32,2,97,104,98,116,32,84,79,32,73,78,80,85
55 DATA 84,46,13,0
56 DATA 49,45,32,86,111,117,115,32,112,97,115,115,101,122,32,99,101,116,116,101

```

57 DATA 32,113,117,101,115,116,105,111,110,46,13,1  
58 DATA 50,45,32,86,111,117,115,32,114,101,108,105,115,101,122,32,99,101,108,108  
59 DATA 101,45,99,105,46,13,2  
60 DATA 51,45,32,86,111,117,115,32,115,104,114,25,110,103,115,97,104,103,116,122  
61 DATA 32,121,111,117,114,32,17,20,15,106,12,46,13,3  
62 DATA 76,111,114,115,113,117,101,32,106,101,32,118,111,117,115,32,100,105,115,32  
63 DATA 34,115,112,104,105,110,99,116,101,114,34,44,32,118,111,117,115,32,114,128  
64 DATA 112,111,110,100,101,122,44,13,0  
65 DATA 49,45,32,34,99,114,128,116,105,110,33,34,13,1  
66 DATA 50,45,32,34,109,101,114,99,105,46,34,13,3  
67 DATA 51,45,32,34,97,103,103,103,114,101,117,117,117,117,33,33,33,34,13,2  
68 DATA 81,117,97,110,100,32,106,101,32,118,111,117,115,32,100,105,115,32,34,70  
69 DATA 114,97,110,107,105,101,32,71,111,101,115,32,116,111,32,72,111,108,108,121  
70 DATA 119,111,111,100,34,44,32,118,111,117,115,32,112,101,110,115,101,122,44,13  
71 DATA 0  
72 DATA 49,45,32,34,66,69,69,80,32,71,79,84,79,32,75,69,89,46,34,13  
73 DATA 2  
74 DATA 50,45,32,34,68,128,115,97,114,109,101,109,101,110,116,46,34,13,3  
75 DATA 51,45,32,34,81,117,101,108,32,101,115,116,32,108,101,32,99,2,12,24  
76 DATA 16,32,113,117,105,32,97,32,102,97,105,116,32,99,101,32,112,114,111,103  
77 DATA 114,97,109,109,101,32,100,101,32,109,17,6,20,23,32,63,34,13,1  
78 DATA 81,117,101,108,108,101,32,128,116,97,105,116,32,108,97,32,99,111,117,108  
79 DATA 101,117,114,32,100,117,32,99,104,101,118,97,108,32,98,108,97,110,99,32  
80 DATA 100,39,72,101,110,114,105,32,73,86,32,63,13,0  
81 DATA 49,45,32,118,101,114,116,46,13,2  
82 DATA 50,45,32,105,108,32,110,39,97,118,97,105,116,32,112,97,115,32,100,101  
83 DATA 32,99,104,101,118,97,108,46,13,1  
84 DATA 51,45,32,98,108,97,110,99,46,13,3  
85 DATA 81,117,105,32,97,32,105,110,118,101,110,116,128,32,108,97,32,100,128,98  
86 DATA 105,108,105,116,128,32,63,13,0  
87 DATA 49,45,32,67,128,115,97,114,46,13,1  
88 DATA 50,45,32,67,104,105,114,97,99,46,13,2  
89 DATA 51,45,32,69,105,110,115,116,101,105,110,46,13,3  
90 DATA 81,117,101,108,108,101,32,101,115,116,32,108,97,32,102,111,114,109,117,108  
91 DATA 101,32,100,128,118,101,108,111,112,112,128,101,32,100,101,32,108,39,97,99  
92 DATA 105,100,101,32,100,128,115,111,120,121,114,105,98,111,110,117,99,108,128,105  
93 DATA 113,117,101,32,63,13,0  
94 DATA 49,45,32,72,45,79,45,72,13,3  
95 DATA 50,45,32,65,45,66,45,67,61,68,45,69,45,70,13,1  
96 DATA 51,45,32,67,72,51,45,67,72,50,45,79,45,67,72,50,45,67,72,51  
97 DATA 13,2  
98 DATA 80,111,117,114,32,118,111,117,115,44,32,113,117,105,32,115,111,110,116,32  
99 DATA 108,101,115,32,114,111,105,115,32,100,101,32,108,97,32,100,128,98,105,108  
100 DATA 105,116,128,32,63,13,0  
101 DATA 49,45,32,67,104,105,99,111,32,100,39,65,103,110,101,97,117,32,101,116  
102 DATA 32,71,114,111,117,99,104,111,32,66,105,122,110,101,115,115,46,13,3  
103 DATA 50,45,32,77,105,116,116,101,114,97,110,100,32,101,116,32,70,97,98,105  
104 DATA 117,115,46,13,2  
105 DATA 51,45,32,83,101,114,103,101,32,86,97,117,100,101,110,97,121,32,101,116  
106 DATA 32,83,101,114,103,101,32,86,97,117,100,101,110,97,121,32,40,99,39,101  
107 DATA 115,116,32,109,111,105,41,46,13,1  
108 DATA 81,117,101,32,115,105,103,110,105,102,105,101,32,108,101,32,115,105,103,108  
109 DATA 101,32,39,77,65,84,72,79,83,39,32,63,13,0  
110 DATA 49,45,32,39,77,105,108,105,101,117,32,65,112,112,114,111,112,114,105,128

```

111 DATA 32,112,111,117,114,32,84,114,97,118,97,105,108,108,101,114,32,72,101,117
112 DATA 114,101,117,120,32,115,117,114,32,117,110,32,79,114,100,105,110,97,116,101
113 DATA 117,114,32,83,97,99,114,105,102,105,128,39,46,13,1
114 DATA 50,45,32,39,109,97,116,128,114,105,101,108,39,44,32,101,110,32,97,114
115 DATA 103,111,116,46,13,2
116 DATA 51,45,32,114,101,112,97,115,32,112,111,117,114,32,99,97,110,105,110,32
117 DATA 109,97,116,104,101,117,120,46,13,3
118 DATA 81,117,101,32,112,101,110,115,101,122,45,118,111,117,115,32,100,101,32,99
119 DATA 101,32,112,114,111,103,114,97,109,109,101,32,63,13,0
120 DATA 49,45,32,83,117,112,101,114,32,33,33,33,46,46,46,13,3
121 DATA 50,45,32,98,101,108,32,101,120,101,109,112,108,101,32,100,101,32,68,65
122 DATA 79,32,40,68,128,98,105,108,105,116,128,32,65,115,115,105,115,116,128,101
123 DATA 32,112,97,114,32,79,114,100,105,110,97,116,101,117,114,41,32,112,111,117
124 DATA 114,32,117,110,32,112,111,105,115,115,111,110,32,100,39,65,118,114,105,108
125 DATA 46,13,2
126 DATA 51,45,32,65,114,99,104,105,45,110,117,108,58,32,112,114,111,103,114,97
127 DATA 109,109,101,32,100,128,98,105,108,101,46,13,1

```

```

=====
128 S=0 @ FOR Q=1 TO 10 @ GOSUB 133 @ GOSUB 134 @ GOSUB 140 @ NEXT Q
129 DISP ' -- R sultats --' @ BEEP 100,2
130 ON IP((S-1)/5) RESTORE 141,141,141,155,167
131 GOSUB 178 @ GOSUB 178 @ GOSUB 178
132 DISP @ DELAY 0,0 @ PUT 'D' @ PUT 'F' @ PUT 'F' @ PUT '#38' @ END
133 DISP @ DISP '-- Question '&STR$(Q)&' --' @ BEEP Q*500 @ RETURN
134 DISP @ DIM A$(4)[96],P(4) @ FOR I=1 TO 4
135 A$(I)=""
136 READ C @ IF I=1 AND C<>13 THEN DISP CHR$(C);
137 IF C<>13 THEN A$(I)=A$(I)&CHR$(C) @ GOTO 136
138 IF I=1 THEN DISP
139 READ P(I) @ NEXT I @ RETURN
140 CALL MENU(A$(I),A) @ IF A<>1 THEN S=S+P(A) @ RETURN ELSE 140

```

```

=====
141 DATA 86,111,116,114,101,32,100,128,98,105,108,105,116,128,32,101,115,116,32,105
142 DATA 110,102,105,110,105,101,46,32,86,111,117,115,32,32,100,101,118,114,105,101
143 DATA 122,32,118,111,117,115,32,102,97,105,114,101,32,105,110,116,101,114,110,101
144 DATA 114,32,100,39,117,114,103,101,110,99,101,46,13
145 DATA 74,101,32,118,111,117,115,32,99,111,110,115,101,105,108,108,101,32,100,101
146 DATA 32,99,111,110,116,97,99,116,101,114,32,105,109,109,128,100,105,97,116,101
147 DATA 109,101,110,116,32,108,101,32,100,111,99,116,101,117,114,32,86,111,110,32
148 DATA 71,108,25,116,101,110,115,99,104,116,117,109,109,101,108,104,105,109,100,24
149 DATA 114,102,46,13
150 DATA 67,39,101,115,116,32,108,117,105,32,113,117,105,32,109,39,97,32,115,111
151 DATA 105,103,110,128,32,40,99,101,32,110,39,101,115,116,32,100,111,110,99,32
152 DATA 112,97,115,32,117,110,32,99,104,97,114,108,97,116,97,110,116,41,44,32
153 DATA 100,101,109,97,110,100,101,122,32,115,111,110,32,97,100,114,101,115,115,101
154 DATA 32,97,117,32,106,111,117,114,110,97,108,46,13
155 DATA 65,116,116,101,110,116,105,111,110,58,32,118,111,117,115,32,115,101,109,98
156 DATA 108,101,122,32,97,117,32,98,111,114,100,32,100,117,32,103,111,117,102
157 DATA 102,114,101,46,13
158 DATA 80,114,111,102,102,105,116,101,122,32,100,101,32,108,97,32,118,105,101,44
159 DATA 32,105,108,32,101,110,32,101,115,116,32,101,110,99,111,114,101,32,116,101
160 DATA 109,112,115,58,32,97,117,99,117,110,32,115,121,109,112,116,111,109,101,32

```

161 DATA 112,114,111,102,111,110,100,32,100,101,32,100,128,98,105,108,105,116,128,46  
162 DATA 13  
163 DATA 83,117,114,118,101,105,108,108,101,122,45,118,111,117,115,44,32,101,116,32  
164 DATA 109,97,110,103,101,122,32,100,101,115,32,99,97,114,111,116,116,101,115,32  
165 DATA 116,111,117,115,32,108,101,115,32,109,97,116,105,110,115,44,32,101,116,32  
166 DATA 116,111,117,116,32,105,114,97,32,98,105,101,110,46,13  
167 DATA 69,116,101,115,32,118,111,117,115,32,115,117,114,32,100,39,97,118,111,105  
168 DATA 114,32,98,105,101,110,32,114,128,112,111,110,100,117,32,97,117,120,32,113  
169 DATA 117,101,115,116,105,111,110,115,32,63,13  
170 DATA 86,111,117,115,32,110,101,32,115,101,109,98,108,101,122,32,112,97,115,32  
171 DATA 112,114,128,115,101,110,116,101,114,32,100,101,32,115,105,103,110,101,115,32  
172 DATA 100,101,32,100,128,98,105,108,105,116,128,44,32,98,114,97,118,111,46,13  
173 DATA 67,111,110,116,105,110,117,101,122,32,100,101,32,109,97,110,103,101,114,32  
174 DATA 100,101,115,32,99,97,114,111,116,116,101,115,32,40,99,39,101,115,116,32  
175 DATA 117,110,32,116,114,117,99,32,113,117,101,32,109,39,97,32,114,128,118,128  
176 DATA 108,128,32,108,101,32,100,111,99,116,101,117,114,32,66,101,97,116,41,46  
177 DATA 13

=====  
178 DISF  
179 READ C @ IF C<>13 THEN DISF CHR\$(C);  
180 IF C<>13 THEN 179 ELSE RETURN

=====  
181 SUB MENU(A\$( ),A)  
182 DELAY INF @ A=2 @ ON ERROR GOTO 188  
183 DISF A\$(A)  
184 B\$=KEY\$ @ IF B\$="" THEN 184  
185 ON POS(' \*43\*50\*51\*38\*162',B\$)/3+1 GOTO 184,186,188,189,190,187  
186 A=1 @ END  
187 A=1 @ GOTO 183  
188 A=A-(A>1) @ GOTO 183  
189 A=A+1 @ GOTO 183  
190 DELAY 1 @ END SUB

# PPC PARIS CHAPTER

ASSOCIATION REGIE PAR LA LOI DE 1901, ENREGISTREE  
A PARIS LE 2 DECEMBRE 1982 SOUS LE NUMERO 82/3240

## BULLETIN D'ADHESION

NOM \_\_\_\_\_  
 PRENOM \_\_\_\_\_  
 ADRESSE \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 COMMUNE \_\_\_\_\_  
 CODE POSTAL \_\_\_\_\_ PAYS \_\_\_\_\_  
 TELEPHONE DOMICILE \_\_\_\_\_ BUREAU \_\_\_\_\_

PROFESSION \_\_\_\_\_  
 INTERETS \_\_\_\_\_

MATERIEL HP EN VOTRE POSSESSION \_\_\_\_\_

AUTRE MATERIEL MICRO-INFORMATIQUE \_\_\_\_\_

COMMENT AVEZ-VOUS CONNU PPC PARIS CHAPTER ?  
 PUBLICITE \_\_\_\_\_ MAGAZINE \_\_\_\_\_  
 AUTRE CLUB \_\_\_\_\_ HP \_\_\_\_\_  
 RELATIONS, MEMBRES DU CLUB, AUTRES \_\_\_\_\_

QUE RECHERCHEZ-VOUS AU SEIN DU PPC PARIS CHAPTER ? \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

Je souhaite adhérer au club PPC PARIS CHAPTER conformément aux statuts de l'association. Au mieux de ma connaissance, je déclare avoir le droit de fournir tous les programmes et informations que je vous enverrai (sans enfreindre des obligations de secret à l'égard d'autres personnes ou organismes) pour publication dans le journal de liaison, sans obligations ni responsabilité d'aucune sorte (en cas d'utilisation frauduleuse) de la part des dirigeants du PPC PARIS CHAPTER.

DATE \_\_\_\_/\_\_\_\_/19\_\_\_\_  
 SIGNATURE, PRECEDEE DE LA MENTION "LU ET APPROUVE" \_\_\_\_\_

LE MONTANT DE LA COTISATION AU PPC PARIS CHAPTER S'ELEVE A 300.00 FF.  
 ETUDIANTS: 250.00 FF. (JUSTIFICATIF INDISPENSABLE)  
 PAIEMENT EFFECTUE LE \_\_\_\_/\_\_\_\_/19\_\_\_\_ A L'ORDRE DE PPC PARIS CHAPTER.  
 PAR  CHEQUE BANCAIRE N° \_\_\_\_\_ BANQUE \_\_\_\_\_  
 CHEQUE POSTAL 3 VOLETS N° \_\_\_\_\_  
 MANDAT LETTRE \_\_\_\_\_

EVENTUELLEMENT: JE M'ABONNE A COMPTER DU \_\_\_\_/\_\_\_\_/19\_\_\_\_  
 JOINDRE A VOTRE INSCRIPTION UNE PHOTO D'IDENTITE ET UNE ENVELOPPE TIMBREE A VOTRE ADRESSE.

VEUILLEZ ENVOYER TOUTE CORRESPONDANCE A:  
 MR PHILIPPE GUEZ, 56 RUE J.J. ROUSSEAU, 75001 PARIS (FRANCE)

