

## A PROPOS DU CLUB

Le Bureau	Editorial	1
J. Belin	Des nouveaux statuts pour PPC PARIS ?	2

## HP28

G. Toublanc	Rappel aux auteurs d'articles	4
G. Toublanc	Passage en modes Symbolique/Numérique	4

## HP48

G. Toublanc	<i>HP48 en Prépa</i> : 2ème édition.	6
G. Toublanc	Courrier des lecteurs	8
G. Toublanc	Numéro de semaine (acte 2)	10
G. Toublanc	Voyage bizarre ... et code nul.	12
G. Toublanc	Trucs sur l'éditions de chaînes	12
A. Ripoll	Traduction des messages d'erreurs	14
P. De Sacy	Attachement de librairies	15
G. Toublanc	Choc en retour	20

## HP95 / HP100

J. Belin	<i>PC de Poche</i> : HP100 & 95LX	22
J. Belin	Configuration commune des cartes Ram	24
J. Belin	Conversion d'images (Acte II)	26
J. Belin	Commande de sélection de mode écran	27

Le coin des codes	29
-------------------	----

## EDITORIAL

Tout d'abord, excusez nous pour les possibles erreurs de mise en page de ce JPC. Alors qu'il aurait dû sortir trois semaines seulement après le précédent (ce qui aurait pu réduire notre retard de parution) nous avons dû retarder de plus d'une semaine la parution à une date qui ne nous était pas favorable, suite à une avalanche d'incidents, et un d'accident. Le principal incident étant un délai important (plus de 5 jours !) dans l'acheminement d'une disquette par le poste. Quant à l'accident, il s'agit du "crash" d'une 205 bien connue des derniers partants de nos réunions mensuelles... Pourtant, malgré ces contretemps de dernière minute, nous avons réussi à tenir notre engagement d'éditer tous les JPC prévus cette année, ce que peu de clubs ont pu faire pendant cette période. Cependant, si le bureau a ainsi prouvé qu'il était solide, la taille et les délais de parution de JPC restent notre grand problème. Car nous ne pouvons pas faire plus pour le résoudre. Il est inutile de répéter encore une fois que l'incident précédemment mentionné n'aurait eu aucune incidence sur notre emploi du temps si nous avions un stock suffisant d'articles pour remplacer le contenu de cette disquette. Cela ne fait que confirmer une décision que nous avons prise auparavant, et dont nous avons discuté à de nombreuses reprises avec les adhérents, consistant à modifier certaines règles de fonctionnement de l'association. Lisez donc la page suivante pour tous les détails et soyez conscients que nous sommes les premiers à penser que ces changements n'auraient jamais dû avoir lieu.

Maintenant, parlons un peu de ce journal. Comme promis le mois dernier, vous trouverez dans ce JPC notre opinion sur deux des derniers livres édités par les éditions Dunod. Selon des rumeurs persistantes, cet éditeur aurait d'autres livres consacrés à nos machines préférées dans ses cartons. Alors patience...

Les utilisateurs de HP28 pourront constater que leur rubrique est très réduite ce mois-ci. Toujours les problèmes d'arrivage au "coup par coup"... Les utilisateurs de palmtops pourront aussi constater que leur rubrique est relativement courte dans ce numéro. Cela est probablement dû au fait que le gros programme en assembleur qui y paraît habituellement est ici réduit à quelques dizaines de lignes. Cependant, si au cours des précédents numéros nous avons souvent parlé du HP100, ce numéro contient pour la première fois des articles qui lui sont vraiment dédiés. Et ce n'est qu'un début !

Avant de terminer cet éditorial, nous vous souhaitons de joyeuses fêtes et une bonne année 1994. Nous vous donnons aussi rendez vous dans le *JPC* de février, qui devrait avoir (enfin !) une couleur plus "civilisée" que ceux de cette année...

## DES NOUVEAUX STATUTS POUR PPC PARIS ?

Depuis quelques années, le journal (qui faisait auparavant plus de 50 pages) n'a cessé de voir sa taille décroître et ses délais de parution compromis. Comme nous l'avons dit à de maintes reprises, la seule et unique cause en est que les adhérents actuels ne nous transmettent pas d'articles. En fait, au cours de ces deux dernières années, seuls 23 adhérents ont fait paraître au moins un article. Certains auteurs réguliers, n'utilisant plus de HP, sont partis, ce qui ne fait qu'aggraver la chose. Aux débuts du club, plus de la moitié des adhérents envoyaient régulièrement des articles. Aujourd'hui, le journal ne paraît plus que par la prédominance involontaire de deux auteurs.

Cette situation n'est pas spécifique à PPC Paris, puisque tous les clubs sont touchés par la raréfaction des contributions. La grande majorité d'entre eux ont des retards de parution compris entre un et deux mois. D'autres, ayant "baissé les bras", ne font plus paraître leur journal qu'épisodiquement. Pourtant, la plupart d'entre eux ont un nombre d'adhérents égal ou supérieur à ce qu'il était il y a quelques années, quand ils n'avaient aucun problème pour alimenter leur publication. Paradoxal quand on sait aussi qu'il n'y a jamais eu autant d'activité sur les machines HP...

Cela est peut-être dû au fait qu'on a un peu oublié que le principe de base d'une *association* est la mise en commun du travail et des connaissances de chacun. Or, certains ne considèrent le club que comme le fournisseur de programmes ou d'un journal (au même titre que l'*Ordinateur Individuel*).

De plus, il semble que certains auteurs réguliers sont de moins en moins motivés pour participer activement quand ils voient peu d'articles d'autres auteurs en échange de leur propre contribution. Pour être franc, certains de ceux-ci sont souvent irrités par le fait que certains ne font rien pour participer, alors qu'ils ont souvent des choses à transmettre.

Pourtant, une grande partie des adhérents se déclarent prêts à nous envoyer des articles, mais ne sautent jamais le premier pas ou reportent l'écriture d'un article à "La Semaine Prochaine". Il faut donc leur donner un petit "coup de pouce".

Pour toutes ces raisons, nous avons décidé de proposer, lors de la prochaine assemblée générale une mesure imposant à tous les adhérents d'envoyer au moins un article par an. Cette mesure aurait donc pour effet de rendre au moins symbolique la participation de certains, et donner le petit "coup de pouce" nécessaire aux autres.

Suite à de nombreuses discussions avec les habitués de nos réunions, il semble que cette mesure soit favorablement envisagée. Cependant, nous pensons qu'il est indispensable qu'elle soit soumise au vote de tous les adhérents, afin qu'elle ne puisse plus être discutée par certains. Pratiquement, et afin de rendre cette mesure officielle, nous avons décidé de modifier les statuts de notre association, ce qui donnera à tous une règle commune.

La modification toucherait l'article 7, dont le texte actuel est le suivant :

*Pour devenir membre de l'association, il faut avoir acquitté sa cotisation et être agréé par le bureau.*

Nous ajouterions le texte suivant (le libellé exact pourra être discuté pendant l'AG) :

*Pour pouvoir réadhérer à l'association, il faudra en plus avoir transmis au club au moins un article pour le journal JPC, au cours de l'année d'adhésion précédente.*

Comme vous pouvez le constater, nous n'offrons toujours aucune obligation au nouvel adhérent. Ce n'est qu'au bout d'une année, et si il veut continuer de faire partie du club, qu'il est soumis à cette mesure.

De façon plus procédurière, puisque nos statuts l'imposent, il est nécessaire de convoquer une Assemblée Générale Extraordinaire afin de délibérer d'une modification des statuts. Elle aura donc lieu le 22 Janvier 1994, en même temps que notre assemblée générale annuelle.

Comme je l'ai dit plus haut, nous n'avons encore rencontré aucune personne opposée à cette proposition. En fait cela nous pose même un problème, puisque nous ne pouvons pas donner le nom d'une personne sur laquelle les absents de l'Assemblée Générale pourraient reporter leur pouvoir. A cet effet, Eric Gengoux a accepté de jouer le rôle de l'"avocat du diable" et de les représenter.

Cependant, notez que si cette mesure est approuvée, nous pourrions ne la mettre en vigueur qu'après un certain délai (deux ou trois mois) afin de ne pas brusquer les personnes dont l'adhésion se termine prochainement.

Pour terminer, je vous rappelle encore une fois que nous serons toujours à votre écoute (comme nous le sommes toujours d'ailleurs !) afin de vous aider à trouver des sujets d'articles possibles, ainsi que vous conseiller pour les préparer !

Jacques Belin (123)

## HP28

G. Toublanc  
G. Toublanc

Rappel aux auteurs d'articles	4
Passage en modes Symbolique/Numérique	4
<b>Le coin des codes</b>	<b>29</b>

## LES BONS COMPTES FONT LES BONS AMIS

Je rappelle que depuis avril 93 les utilisateurs de HP28S ont à leur disposition la fonction *BYTES* (JPC 85 pages 9-10) comme sur les HP48. Vous avez remarqué que la rubrique *le coin des codes* fait toujours mention du Checksum et du nombre d'octets fournis par *BYTES* pour tous les programmes HP48 et HP28. C'est la seule façon fiable et rapide de s'assurer que les objets dans les machines sont bien conformes aux listings correspondants. Cette fonction permet aussi d'identifier les différentes versions de programmes très rapidement sans avoir à les éditer ni les lister, ce qui peut-être très long (ou même impossible pour l'assembleur) sans vous éviter les erreurs de comparaison.

Je fais ce rappel car les personnes contribuant à la rubrique HP28 n'ont pas toujours les mêmes possibilités que leurs homologues HP48 pour nous transmettre leurs travaux sur support magnétique. C'est ce qui est arrivé à notre ami Paul Jebeily qui nous a livré un ensemble de programmes très intéressants sur le graphisme mais sur support papier. Ce mode de transmission méritoire n'est pas sans quelques risques de fautes de frappes qui ne peuvent être reprochées ni à l'auteur, ni à la personne qui met en page (en l'occurrence notre dévoué Jacques). Ainsi malgré l'erratum du dernier JPC je n'ai pas réussi à faire tourner les *demos*. Si la fonction *BYTES* avait été utilisée et que le Checksum des programmes que j'ai tapés était le même je pourrais être certain que je n'ai pas fait de fautes de frappes ou dans le cas contraire je pourrais identifier le ou les programmes en défaut ce qui limiterait le travail de vérification.

Ce programme *BYTES* avec son nom n'occupe que 199 octets et les vérifications qu'il vous permet ne sont pas superflues car beaucoup n'ont pas la possibilité de sauvegarder leurs objets HP28 et sont à la merci de leur perte par utilisation de programmes en assembleur mal introduits. *BYTES* est aussi utile aux programmeurs qui veulent optimiser leurs travaux.

Merci à tous pour ce petit travail supplémentaire lors de l'envoi de leurs programmes.

Guy Toublanc (276)

## TRUCS ASTUCES ET PETITS UTILITAIRES

Il ne doit pas y avoir de ségrégation entre HP28 et HP48. Les idées ou programmes peuvent être adaptés d'une machine à l'autre. Ici on pourrait trouver des découvertes d'utilisateurs de HP28 mais pouvant avoir leur place côté HP48 ou au contraire des éléments spécifiques HP28 parce que cela existe déjà dans la Rom HP48.

Ouvre cette rubrique avec un petit utilitaire comblant une lacune des HP28 par rapport aux HP48: passer sans effort en modes symbolique/numérique. Cette bascule entre les 2 modes affiche YES si l'on passe en mode symbolique, sinon NO.

L'idéal serait un label SYMB avec carré en mode symbolique actif comme les labels du menu MODE.

Exemple : passage en mode numérique par appui sur [SYMB]. Réponse NO (affichage d'une seconde au sommet gauche) : c'est bon, sinon deuxième appui.

En User Rpl le programme serait :

```
« 35 FC?C IF THEN 35 SF "YES" ELSE "NO" END
  1 DISP 1 WAIT »
```

Donc en 81 octets se réduisant à 40 en System Rpl :

### SYMB

Cksum: #37E6h 40 octets

### ASSEMBLE

```
con(5) #02c67 * DOCOL
con(5) #07307 * THIRTYFIVE
con(5) #0204a * DUP
con(5) #0c8dc * TestUserFlag
con(5) #3dca3 * ITE
con(5) #0c8c8 * ClrUserFlag
con(5) #0c8a7 * SetUserFlag
con(5) #07307 * THIRTYFYVE
con(5) #0c8dc * TestUserFlag
con(5) #3dca3 * ITE
con(5) #3ef53 * "YES"
con(5) #3ef63 * "NO"
con(5) #1bba3 * DISPROW1
con(5) #112c9 * %1
con(5) #1a16e * wait_real
con(5) #02f90 * SEMI
```

Voir la liste des codes dans le *Coin des codes*.

Guy Toublanc (276)

## HP48

G. Toublanc	<i>HP48 en Prépa : 2ème édition.</i>	6
G. Toublanc	Courrier des lecteurs	8
G. Toublanc	Numéro de semaine (acte 2)	10
G. Toublanc	Voyage bizarre ... et code nul.	12
G. Toublanc	Trucs sur l'éditions de chaînes	12
A. Ripoll	Traduction des messages d'erreurs	14
P. De Sacy	Attachement de librairies	15
G. Toublanc	Choc en retour	20
	<b>Le coin des codes</b>	29

## LITTERATURE HP48

### HP 48 G ◦ GX ◦ S ◦ SX en prépa

Il y a un an j'avais analysé la première édition de cet ouvrage. A l'époque j'y avais trouvé des choses intéressantes mais je ne cachais pas que des améliorations étaient souhaitables. Depuis, un deuxième tirage a vu le jour (la HP48 a du succès en France).

Aujourd'hui cette édition se veut bi-valente en faisant profiter à la fois les possesseurs de HP48 séries S et G des améliorations de cette deuxième mouture.

L'ouvrage a gardé la même structure générale mais avec des améliorations et des compléments très substantiels faisant passer son volume de 398 pages à 508. Je ne reviendrai pas sur sa structure car cela avait été développé assez largement dans le JPC n°82 (pages 16-21) auquel je renvoie le lecteur pour l'analyse du contenu.

Parmi les améliorations il y a celles que j'avais suggérées :

- Indication du checksum et du nombre d'octets pour chaque programme ou objet pour permettre d'être assuré que l'on a bien reproduit les listings à la fois en User Rpl, System Rpl et assembleur. Cela permet également de déceler les programmes non identiques d'une édition à l'autre.

- Indication du temps d'exécution pour obtenir les résultats des exemples.

- Abandon de la méthode des frappes de touches qui était pour moi le point noir de l'ouvrage.

- Fourniture des listings des fichiers sources avec des commentaires. Cela a été fait pour certains programmes et même consciencieusement. On conçoit que cela ne pouvait être fait pour l'intégralité des programmes et bibliothèques sans poser quelques problèmes économiques. Evidemment que cela aurait comblé les curieux comme moi, mais pour un livre qui n'est pas spécialisé dans ce domaine nous ne pouvons pas en demander trop.

- Partitionnement de la liste des internes par familles. Cela n'est pas aussi détaillé que dans les ouvrages de Jean-Michel Ferrard mais ce qui a été fait va dans le sens du progrès. Là encore ce domaine n'est pas l'unique objet du livre.

- Amélioration de la rapidité des programmes en faisant appel à des programmes en System Rpl. Ceci a été réalisé pour des sous-programmes souvent appelés. Pour moi il aurait fallu aller plus loin dans ce domaine et peut-être même aborder les routines en virgule flottante via l'assembleur.

- Utilisation d'un sous-programme d'aide en ligne et de vérification des arguments pour de nombreux programmes qui doivent faire ces vérifications.

J'ai pu constater qu'une grande partie de mes doléances concernant la première édition n'était pas restée lettre morte et que cela a permis de substantielles améliorations. Suis-je totalement comblé ? Non, il reste encore quelques points que j'aurais aimé voir améliorés. Par exemple la disparition totale dans les programmes des suites de IF THEN ELSE IF THEN ELSE ..... lorsque cela peut être très avantageusement remplacé par la structure CASE qui est plus simple et améliore la rapidité. J'avais donné un exemple (JPC 82 p 19) et le scénario pourrait se répéter pour d'autres programmes, par exemple pour le premier programme LOOK du livre où chaque condition exclut les autres et pourtant tous les tests sont effectués ligne par ligne :

```
IF ..... THEN ..... END
IF ..... THEN ..... END
.. ..... ..
.. ..... ..
IF ..... THEN ..... END
IF ..... THEN ..... END
```

Ce programme dont l'algorithme utilisé est un peu trop simple pour sa fonction de recherche de mots et mène parfois à des résultats inattendus. Un allègement de 121 octets sur 650 peut se réaliser sans nécessiter d'acrobaties. Il est resté une faute d'étourderie: la séquence SWAP 3 ROLLD SWAP est équivalente à ROT. D'autres ont échappé ici et là à la sagacité de l'auteur comme par exemple dans le programme FIND avec DROP DROP au lieu de DROP2.

Un autre motif d'alourdissement de programmation que l'on rencontre dans le livre et que j'avais déjà dénoncé est la répétition des tests :

```
IF 0 == THEN au lieu de IF NOT THEN
IF 0 ≠ THEN au lieu de IF THEN
```

Ces types de programmation sont très persistants chez d'autres auteurs alors que certains ont su s'en dispenser rapidement.

Concernant les listings, si j'ai reproché à un autre auteur l'abus d'artifices typographiques pour gonfler le volume d'un ouvrage, à l'inverse certains pourraient

reprocher à ce livre la très grande compacité des listings, sans indentation et donc rendant plus difficile la lisibilité des programmes et leur compréhension. Je pense qu'une solution raisonnable est celle qui consiste, dans un ouvrage, à distinguer deux choses :

- D'un côté les *leçons* où les listings respectent bien la structure des programmes et ceux-ci étant commentés ligne par ligne.

- De l'autre côté les outils pouvant être mêmes de gros programmes et que l'auteur livre comme des utilitaires sans but pédagogique mais dont le nombre sera un critère d'achat du livre pour un utilisateur courant. Ici on se contente de savoir ce qu'il faut fournir à un programme et ce qu'il nous donnera en retour.

Si l'aspect économique n'entraîne pas en compte, l'idéal serait évidemment d'avoir sur toutes les pages des listings type *leçon*. Mais quant à avoir, dans un livre que je ne reciterai pas, des lignes du genre :

*programme toto*

«  
DUP  
»

*programme ...*

au lieu de:

*programme toto*

« DUP »

*programme ....*

ce qui me semble aussi compréhensible, je dis non.

Du point de vue des programmes outils, je pose la question : quel est le pourcentage parmi les utilisateurs de bibliothèques de programmes en simple User Rpl qui font l'effort d'analyser les listings pour les comprendre ?

Le reproche des listings compacts de ce livre peut être tempéré par le fait qu'une disquette de tous les outils est jointe à l'ouvrage ce qui permet d'éviter les fautes de saisie pour entrer les programmes. Libre à chacun de lister ces outils à partir de la HP-48 avec les indentations données par celle-ci lors de l'impression ou même de la transmission en mode ascii à un PC, puis de les analyser plus confortablement.

Maintenant chaque chapitre de programmes débute par une liste de ceux-ci avec ce qu'ils font. Une bonne

chose pour avoir une vue d'ensemble et s'y retrouver parmi tous ces programmes.

Dans cette catégorie de chapitres, deux ont vu le jour :

#### Physique (8 pages)

Avec BODE programme principal de tracé et d'étude de fonctions de transfert.

et GIBBS pour le calcul de l'enthalpie

#### Chimie (12 pages)

Pour le calcul du pH avec des sous-programmes de saisie des données.

Et le calcul de la masse moléculaire en se servant d'une liste des masses moléculaires atomiques à compléter suivant les besoins de chacun.

Le chapitre Arithmétique voit maintenant les deux programmes EDIVI et PGCD bien listés avec des fichiers sources très bien commentés. Pour EDIVI la liste des codes est à modifier : à la ligne 2 on doit avoir CCD 203 et non CCD 201 sinon cela ne fonctionne pas. Il y a eu aussi une erreur avec ce programme sur la disquette. L'assemblage a été mal réalisé.

Pour les autres programmes de ce chapitre deux versions sont données : une en User Rpl et une en System Rpl plus rapide. Cela va dans le sens de ce que je souhaitais: l'amélioration de la rapidité d'exécution, en particulier pour des programmes souvent appelés.

Le chapitre *Assembleur* se termine maintenant par une initiation accessible à la programmation dans ce langage en partant d'exemples de programmes bien commentés. Cette partie sera bien accueillie par les débutants en assembleur. Evidemment qu'il n'est pas possible de tout dire en 20 pages, mais ce qui a été tenté est méritoire.

Le chapitre *Internals* possède maintenant sa petite introduction à la programmation en System Rpl en partant d'un exemple en User Rpl et traduit dans l'autre langage. La liste des adresses fournies a été partitionnée par grandes familles avec une colonne HP48S/X et une colonne HP48G/X. Ce chapitre se termine par un paragraphe *Remarques* contenant un programme PIK pour accéder à la Rom cachée pour la partie #C0000h à #FFFFFFh et un programme pour l'adresse d'un objet en Ram.

Le chapitre *Applications* est devenu *Extensions* et a grossi avec la fourniture de 4 programmes :

- 3 compacteurs pour HP48 présentant des avantages et des inconvénients suivant le type d'objet HP48 sur lequel ils opèrent.

- 1 décompacteur pour les trois types de compactage. Il s'agit en réalité de 3 décompacteurs réunis en un programme.

Tous ces programmes sont en assembleur et livrés sous forme de codes hexa à assembler.

Le dernier chapitre donne le mode d'utilisation de la disquette DOS 3"1/2 720 Ko utilisable sur PC et sur Macintosh (avec lecteur HD 1.44 Mo).

Contrairement aux récentes éditions de livres du domaine HP48 et qui ont coupé les ponts avec les HP48S/X, *HP48 en prépa* se veut bi-valent par son titre. Qu'en est-il exactement ? Il y a 2 versions pour séries S et G des programmes :

- SYS pour rappeler un objet en Rom
- LIB→ pour décomposer une bibliothèque.

A cela on peut ajouter dans le chapitre *Internals* les doubles listes d'adresses pour HP48 G et S. Des renseignements concernant les ports des deux types de machines font aussi les différences, et enfin le PEEK en Rom cachée pour les 128 derniers Ko de la Rom HP48G/X.

Pour le reste tout est commun et cette très large compatibilité pourra peut-être apparaître comme contraignante pour les nouveaux venus dans le clan HP48 mais série G ou ceux qui renient leur HP48 série S. En effet ces derniers auraient peut-être préféré une mise à profit, pour les programmes en User Rpl, des nouvelles possibilités de leur HP48. Le choix fait pour ce livre bi-valent peut se justifier si on considère qu'il réapparaît dans une période de transition pour les HP48 et ainsi peut rendre service à une large clientèle.

Dans ma précédente analyse j'avais fait allusion à la plus ou moins grande rapidité des programmes. Un certain nombre de ceux-ci travaillant avec les rationnels, ce qui est un plus, perdent en rapidité du fait que les manipulations d'objets sont plus compliquées. Il doit être possible d'améliorer cela, ce qui est un excellent exercice de reprogrammation pour qui veut s'y atteler.

Conclusion :

Cet ouvrage constitue une boîte à outils qui s'est améliorée en de nombreux points. Comme tout ce qui a un caractère polyvalent, elle ne satisfera pas le futur spécialiste d'un domaine particulier mais les

chapitres satellites de la partie principale (les programmes mathématiques) inciteront à aller plus loin. Les possibilités de la HP-48 peuvent laisser envisager d'autres extensions de l'ouvrage qui n'ont pas été abordées ou très peu développées et nous ne pouvons que souhaiter une autre suite à ce livre et donner rendez-vous pour la troisième édition où quelques défauts auront été gommés.

Guy Toublanc (276)

HP 48 G GX S SX en prépa

Matthieu Cornillault

M. de Courville E. Lesueur

DUNODTECH

175 francs (disquette comprise)

---

## COURRIER DES LECTEURS

Je rappelle que cette rubrique est toujours à la disposition de chacun pour poser des questions et que les éventuelles réponses peuvent intéresser d'autres lecteurs. Celles-ci pourront être plus ou moins rapides suivant leur niveau de difficulté ou même ne pas voir le jour si personne ne peut les fournir. Mais dans tous les cas il faut tenter sa chance. Cette rubrique est particulièrement utile aux lecteurs qui ne peuvent pas assister aux réunions.

Guy Toublanc (276)

Edmond-Jean Dougados

BP 144.09

75422 Paris Cedex 09

NDLR: extrait d'une lettre qui nous a été adressée

..... cherche une solution de remplacement de la plaque métallique du clavier (HP48SX) dont la peinture s'égratigne facilement, cette plaque ne pouvant être fournie par HP ..... la solidité des HP48 ne vaut pas celle des HP41 et 28 (utilisation sur chantier bâtiment ou génie-civil) .....

### Réponse du Dr JPC

Malheureusement, si l'on peut déplorer cette plus grande vulnérabilité extérieure, il n'existe pas de solution curative peu coûteuse hors celle de récupérer une plaque en bon état sur une machine inutilisable par ailleurs.

Les lecteurs qui possèderaient une telle machine seraient gentils de se manifester.

Les solutions préventives existent :

- Recouvrement de cette plaque (pas les touches) par l'un des caches du kit :

HP 82220A  
Overlay Kit for HP48

ce kit comprend :

- 1 cache transparent avec des caractères qui n'apparaissent pas sur la plaque du clavier
- 3 caches transparents
- 3 caches opaques blancs pour clavier personnalisé

- protection par un sachet transparent étanche et épousant la forme de la machine, que l'on trouve chez EduCalc (j'ignore si cela est disponible en France) :

- #543X Standard Comp-U-Bag \$2.50
- #543XT 10 Standard Comp-U-Bag \$19.95
- #543U Anti-Static Comp-U-Bag \$2.75
- #543UT 10 Anti-Static Comp-U-Bag \$21.95

Il y a aussi la solution de réaliser soi-même ces sachets le problème est d'avoir un plastique qui n'altère pas la lisibilité du clavier et de l'écran, ce plastique doit bien épouser la forme de la machine et posséder une fermeture étanche et pratique pour accéder aux trappes ou permettre une éventuelle connection. La solution de raccordement des bords du plastique par soudure thermique peut donner de bons résultats.

---

Joel Thibaut  
65, impasse Morand  
44000 Nantes

Que font les points d'entrée (HP48S/X):

- 62160h ?
- 621F3h ?

### Réponse du Dr JPC

Vous ne dites pas si vous avez trouvé ces adresses lors d'une décompilation de type Rpl ou assembleur. Les décompilateurs en mode automatique ne donnent pas à tout coup des listings exacts. En assembleur le code peut correspondre à des tables qui n'ont rien à voir avec des instructions. D'autre part pour protéger une librairie un programmeur peut très bien inclure du

code qui ne sert à rien sinon qu'à empêcher un désassemblage automatique. Dans ce cas il faut désassembler en pas à pas ce qui n'est pas à la portée du débutant.

En ce qui concerne les deux adresses données, cela ne correspond à rien. Voici la méthode permettant d'affirmer cela :

Chercher dans une liste de points d'entrées une adresse voisine en amont. Pour 621560h nous avons #62159h de mnémonique HP TYPECSTR? ou st? chez J.M. Ferrard. C'est une primitive qui teste si un objet est une chaîne et dont le code décompilé donne :

62159	6215E		(primitive)
6215E	34C2A20	lchex	=DOCSTR
62165	65FE	goto	#6205B

puis nous trouvons:

62169	CC	A=A-1	A
-------	----	-------	---

# 62160h ne correspond donc à rien. Si vous faites un Syseval à cette adresse vous obtiendrez sur la pile une chaîne étrange dont SIZE donnera 423848 pour longueur !! Cela s'explique avec le code en mémoire à partir de # 62160h :

C2A20	65FEC	C.....
prologue chaîne	longueur	données

#CEF56h - 5 = #CEF51h      #CEF51h / 2 = #677A8h  
# 677A8h = # 423848d

Pour 621F3h nous trouvons #621EC de mnémonique J.M. Ferrard *rpl?* qui est aussi une primitive qui teste si un objet est un programme Rpl et dont une décompilation nous donne :

621EC	621F1		primitive
621F1	34D9D20	lchex	=DOCOL
621F8	626E	goto	#6205B

puis nous trouvons:

621FC	CC	A=A-1	A
-------	----	-------	---

Ceci permet d'affirmer que #621F3h ne correspond à rien non plus. Un Syseval avec cette adresse ne produira que l'exécution d'un programme incohérent qui ne s'arrêtera qu'avec un arrêt système. Cela s'explique encore avec le code en mémoire à partir de # 621F3h:

D9D20	626EC	C.....
prologue programme		

Pour des programmeurs voici donc deux moyens parmi d'autres de brouiller les cartes et d'empêcher les regards indiscrets ou les indéclicats (et il n'en manque pas) de s'approprier le travail des autres.

Vous n'êtes pas le premier à poser des questions de ce type et qui aboutissent à des réponses de non-existence. Mais j'espère que les explications données ici vous satisferont plus que celles que l'on rencontre sur certains serveurs et qui sont du genre "c'est l'anniversaire de ma grand-mère".

Guy Toublanc (276)

## NUMERO DE SEMAINE

### Acte II

Robert Pulluard nous proposait (JPC 91 page 11) un petit programme calculant le numéro de semaine dans l'année et correspondant à une date donnée. Robert nous fournissait l'algorithme et sa méthode en laissant deviner qu'il livrait en pâture son programme.

Quelques essais ont permis d'apporter de sensibles améliorations:

- 1- en User Rpl un gain de 57 octets sur 158,5.
- 2- la traduction en System Rpl fait encore gagner 12,5 octets.
- 3- un peu d'assembleur permet encore d'alléger de 14 octets.

Rappel du listing du programme de Robert (ici les encombrements mémoire sont obtenus avec l'objet sur la pile):

**NSEM** 158.5 octets

```
« DUP NOT ( DROP DATE ) IFT                @ 1
-2 ALOG MOD LASTARG ROT + 1 +              @ 2
DUP ROT DDAYS 3 +                          @ 3
( "MON" "TUE" "WED" "THU" "FRI" "SAT" "SUN" ) @ 4
ROT 0 TSTR                                  @ 5
1 3 SUB POS                                 @ 6
4 - 7 MOD +                                 @ 7
7 / FLOOR »                                 @ 8
```

### 1ère étape : programmation en User Rpl

Je n'ai pas découvert la recette miracle pour me dispenser des chaînes de caractères. De ce point de vue Robert va être déçu. Les 57 octets disparus sont la conséquence de points de détails :

- Ligne 1 : la neutralité de zéro pour l'addition permet de gagner 5 octets.
- Ligne 2 : pas de gain mais affranchissement de LASTARG.
- Ligne 3 : en combinant l'algorithme des lignes 3 et 7 on gagne ici 5 octets.
- Ligne 4 : pour identifier le jour de la semaine les 2 premiers caractères suffisent. Une seule chaîne nécessaire en utilisant l'algorithme de la ligne 7 ce qui économise 42 octets.
- Ligne 7 : la combinaison de l'algorithme des lignes 3 et 7 permet de gagner ici 5 octets.

Ce programme est légèrement plus rapide mais ce n'est pas le point fort.

**NSEMuser**

Cksum: # A98Bh 101.5 octets

```
« DUP 0 DATE IFTE +                        @ 1
-2 ALOG OVER OVER MOD + 1 +              @ 2
DUP ROT DDAYS                             @ 3
"MOTUWETHFRSASU"                          @ 4
ROT 0 TSTR                                  @ 5
1 2 SUB POS                                 @ 6
2 / +                                       @ 7
7 / FLOOR »                                 @ 8
```

### 2ième étape : programmation en System Rpl

A partir de maintenant les gains d'octets ne sont plus à mettre au compte d'améliorations de détails par rapport au programme de Robert, mais seulement à l'actif d'un autre langage. En effet je suis pas à pas mon programme en User Rpl dont je conserve quelques instructions de ce langage. Donc ici 12,5 octets de moins seulement. %-2 %ALOG est remplacé par %100 %1/ ce qui donne le même résultat mais c'est histoire de varier. Je rappelle que lorsque des instructions User Rpl sont utilisées, elles sont préfixées par x pour les distinguer de celles en System Rpl.

**NSEMsys** 89 octets

```
ASSEMBLE
=xDUP EQU #1fb87
=xPOS EQU #1cab4
=xTSTR EQU #19992
RPL
```

```

::
xDUP %0= ITE DATE DUP
SWAPDROPDUP %100 %1/ SWAPOVER %MOD %+ %1+
DUPROT DDAYS
"MOTUWETHFRSASU"
ROT %0 xTSTR
THREE 1_#1-SUB$ xPOS
%2 %/ %+
%7 %/ %FLOOR
;

```

### 3ième étape : programmation en System Rpl et assembleur

Ici nous pouvons nous dispenser de la chaîne des débuts de noms de jours grâce à l'assembleur. La première instruction en User Rpl a été conservée car elle permet un gain d'octets par rapport au System Rpl. C'est donc le programme le plus court et aussi le plus rapide (3 fois plus que celui de Robert).

**NSEMass** 75 octets

```

ASSEMBLE
=xDUP EQU #1fb87
RPL
::
xDUP %0= ITE DATE DUP
SWAPDROPDUP %100 %1/ SWAPOVER %MOD %+ %1+
DUPROT DDAYS
COERCESWAP
CODE
    gosbvl =POPDATE%
    gosbvl =YMD>Ticks
    gosbvl =dowutil
    sethex
    govlng =PUSH#ALOOP
ENDCODE
#+-1 SEVEN #/ SWAPDROP UNCOERCE
;

```

### 4ième étape : pour les puristes

Tous les programmes ci-dessus (y compris celui de Robert) renvoient des messages d'erreurs qui ne satisferont pas les puristes. Aussi voici l'ultime version un peu plus octivore mais renvoyant des des messages beaucoup plus orthodoxes. Il ne reste donc plus une seule instruction User Rpl.

### NSEM2

Cksum: # 4D43h 85 octets

```

::
CK1NoBlame CK&DISPATCH1 real
::
DUP%0= ITE .....
..... UNCOERCE

```

```

;
;

```

Je rappelle que ces listings en assembleur et System Rpl sont compilables avec les compilateurs HP sur PC. Pour ceux qui utilisent la librairie <-RPL-> sur HP48 il suffit de mettre en majuscules les mnémoniques de la partie CODE et pour les chaînes de faire \$ "cccc..." où cccc... sont les caractères.

A titre expérimental voici les listings sources, du programme NSEM2, compilables sur PC avec les mnémoniques et les compilateurs de Jean-Michel Ferrard. Le fichier source DOW.ASM doit d'abord être compilé avec ASMPc pour générer le fichier objet ASM.OBJ. Puis le fichier source NSEM.SRC devra être compilé avec EXTERNPC pour obtenir le programme HP48 NSEM2.

### NSEM.SRC

```

begin
    depth=0?:TFA! ChkArg <1h>
    begin
        dup;=0(r)? pifte date dup
        drop(2);dup %1E2 inv(r) swap;over mod(2r)
        +(2r) 1+(r)
        dup;rot ddays(2r)
        r=>s;swap
    end
end

INCLUDE DOW.OBJ

    +(2s);1-(s) <7h> mdv(2s) drop(2) s=>r
end
end

```

### DOW.ASM

```

gosbvl #0D92C
gosbvl #0DB3A
gosbvl #0DB51
sethex
govlng #0357C * PushA<><

```

Cet article n'a pas la prétention d'avoir fermé la parenthèse sur ce problème, aussi le champ est libre pour des solutions plus astucieuses quel que soit le langage de programmation. Merci à Robert de nous avoir tendu la perche et dont l'article me dispense de fournir d'autres explications sur le fond du problème.

Vous trouverez la liste des codes de NSEM2 dans le *Coin des codes*.

Guy Toublanc (276)

## VOYAGE BIZARRE....

### ...et code nul

Ceux qui ont utilisé le coin des codes pour les programmes BANKN2 et GO de Pierre de Sacy et dans le dernier JPC-91, ont dû trouver étrange :

- la fin des codes avec tous ses zéros
- la non conformité du checksum ainsi que du nombre d'octets (plus) donnés dans le coin des codes et ce qui est obtenu après assemblage dans la HP48 .
- que la liste des codes du programme XMOD-KER ne posait pas ces problèmes.

L'explication est simple : Ces programmes de Pierre ont fait le *voyage* ou transfert HP48-PC sous protocole Xmodem et les outils que le club possède pour établir les listes de codes, sont des programmes sur PC et qui ont été réalisés pour des HP48S/X donc pour des programmes HP48 transférés sous protocole Kermit.

Le résultat obtenu après assemblage de ces listes de codes trop longues n'a pas d'incidence grâce à NEWOB du programme d'assemblage qui remet tout ça en ordre.

Si les programmes fournis par Pierre sur disquette devaient faire le *voyage* PC-HP48, il y aurait deux possibilités (JPC 91 page 14) :

- Utiliser le protocole Xmodem.
- Ou utiliser le protocole Kermit et récupérer le programme, de la chaîne obtenue après transfert, avec le programme XMOD-KER (JPC 91 page 14).

Pour ceux qui devraient entrer les codes de ces programmes voici le début et la fin des listes à modifier par rapport à celles de JPC 91:

```
BANKN2          (HP48)
# ECE2h          353 octets

      0123 4567 89AB CDEF  sm

000: D9D2 0FDE 8111 9201  CAC
... ..
... ..
02C: 30          7A5

GO              (HP48)
# F4D1h          359.5 octets
```

```
0123 4567 89AB CDEF  sm
```

```
000: D9D2 0ECE 81B2 040D  E3D
... ..
... ..
02C: 4423 0B21 30B2 130   8AA
```

Moralité : Lors de l'envoi au club de programmes sur disquette, utilisez le protocole Kermit qui est pour les petits objets (quelques centaines d'octets) aussi rapide que Xmodem car d'accès plus rapide ce qui compense la lenteur du transfert lui-même. Ainsi nous n'aurons plus de *VOYAGES* à problèmes avec un surplus de *CODE NUL*.

Guy Toublanc (276)

---

## TRUCS ASTUCES ET PETITS UTILITAIRES

On sait que l'éditeur des HP-48 a toujours été par sa lenteur la partie faible de cette machine.

Ce qui suit intéressera surtout ceux qui éditent leurs programmes et chaînes avec l'éditeur de leur machine et gagneront un peu de temps sans effort avec une méthode qui, si elle n'est pas géniale, a l'avantage d'être simple (et même simplette).

### Le problème des larges césures

1- Couper une chaîne par larges séquences lorsque celle-ci est éditée peut se faire normalement avec les commandes de l'éditeur +DEL ou DEL+ si des changements de lignes existent dans la chaîne ou caractère par caractère avec [+ ] (backspace) pour des parties de lignes ou lorsqu'il n'y a pas de changements de lignes. Ceci peut-être laborieux. Il existe une solution simple à laquelle certains ne pensent peut-être pas. Il suffit d'insérer des "" entre les suites de caractères à conserver et celles à éliminer lorsque l'éditeur est actif. La sortie de l'éditeur se réalisant avec [ENTER] vous trouverez *n* chaînes sur la pile au lieu d'une. Il ne restera plus qu'à éliminer les indésirables puis à concaténer ce qui reste (avec [+ ]) et dans le bon ordre pour restituer la chaîne allégée. Cette façon de faire n'est évidemment rentable que pour de larges coupures dans de longues chaînes, ce qui est le cas des fichiers sources *assembleur* compilés dans une HP-48.

2- en ce qui concerne les coupures de programmes la méthode peut être appliquée de la même manière : transformer le programme en chaîne avec -STR puis opérer comme pour les chaînes et modifier la chaîne par les ajouts nécessaires à la nouvelle structure du programme (ceci peut aussi se faire en phase finale). Après concaténation des parties subsistantes et, si cela n'a déjà pas été réalisé, modifier la structure du programme avant de passer à la transformation en programme avec OBJ-. L'avantage de travailler sur la chaîne au lieu du programme lui-même est que l'éditeur est moins lent avec le type d'objet chaîne.

Lorsque les chaînes ou programmes contiennent des chaînes cette méthode n'est plus applicable ce qui m'amène au problème suivant.

### Du bon usage de C\$

Ceci pourra intéresser plus particulièrement les possesseurs de HP48G/X car à ma connaissance C\$ n'est documentée dans aucun des manuels des HP48G. C\$ permet de générer des chaînes comptées (manuel d'utilisation des HP48SX page 88 ou tome 1 de *La maîtrise de la HP48SX* par J.M Ferrard, page 144) ce qui peut permettre des acrobaties avec certains types de chaînes. Si vous tapez "TOTO "bis"" pour obtenir une seule chaîne, en validant avec [ENTER] vous obtiendrez:

```
"TOTO "  
  'bis'  
  ""
```

Ce qui n'est pas le résultat escompté.

Pour obtenir la chaîne tapée ci-dessus il faut taper une chaîne comptée :

```
C$ 10 TOTO "bis"
```

avec le nombre de caractères précisé: 10

ou C\$ \$ TOTO "bis"

Rappel: \$ s'obtient avec [ø] [shift gauche] [4].

Après validation nous trouverons sur la pile :

```
"TOTO "bis""
```

Si l'on édite ce résultat la machine fournira:

```
C$ 10 TOTO "bis"
```

Si, en restant dans l'éditeur, vous modifiez cette chaîne, deux cas se présentent :

1- Réduction: alors modifiez sans compter.

```
C$ 10 TOTO "as"
```

la machine vous donnera: "TOTO "as"".

2- Expansion: alors modifiez la chaîne et l'en-tête pour laisser à la machine le soin de compter.

```
C$ $ TOUTOU "as"
```

Cette technique des chaînes comptées ne permet donc pas les coupures par la méthode exposée au début de l'article. Donc il faudra couper par la méthode normale les chaînes contenant elles-mêmes des chaînes ou le caractère "". Le recours à ces chaînes comptées sera particulièrement utile à ceux qui font de l'assembleur directement dans leur HP48.

### A propos d'éditeur

Pour les objets éditables avec l'éditeur HP il devient particulièrement inconfortable, lorsque le volume des objets grossit, de visionner puis de traiter ces objets. Il existe des solutions de remplacement parmi lesquelles il faut distinguer les visionneurs (*viewers*) et les éditeurs avec des fonctions de traitement de ces objets.

Un certain nombre de programmeurs ont créé des programmes ou des bibliothèques qui entrent dans une de ces catégories.

Dans la catégorie "Editeur" je signalerai, parmi les outils importants, deux de ceux-ci :

GED de Lutz Vieweg (aussi auteur du très célèbre compacteur RFU). La version que je connais est contenue dans le répertoire UTILS des Goodies Disk #6 (et avec une documentation complète). Ceci est diffusé en shareware. Cette version datant de deux ans, je suppose que l'auteur a dû l'améliorer.

L'excellente et bien connue bibliothèque STRING WRITER qui nécessite de la place disponible car si la version 1 occupe 9 Ko, la version la plus récente atteint les 23 Ko, donc les possesseurs de HP48S ou HP48G seront peut-être obligés de s'en dispenser.

C'est une bibliothèque qui est plus qu'un éditeur car ayant les fonctionnalités d'un traitement de texte très performant. Lorsqu'on la possède les 'trucs' exposés ci-dessus sont inutiles.

Elle fut créée à l'origine par deux programmeurs : Jean Yves Avenard et Christian Bourgeois. Le premier fait maintenant cavalier seul. Cette bibliothèque a évolué dans le temps avec toujours plus de nouvelles

fonctionnalités ou raffinements en tenant compte des remarques d'utilisateurs. Si les premières versions se trouvent en téléchargement sur certains serveurs, ou sont diffusées par d'autres voies, à partir de la version 4 cette librairie est en shareware ce que certains ont critiqué mais, pour ma part cette solution ne me choque pas. Quand on fait le compte de ce qui peut rester sur les 50 francs demandés, l'auteur ne va certainement pas se faire une fortune. En effet les souscripteurs ont droit à une actualisation avec sa documentation.

Pour ceux qui doivent éditer chaînes et programmes (ce que font les dernières versions alors que les premières n'étaient que des chaînes) c'est un outil essentiel et très performant par sa vitesse car très largement en assembleur.

La version 4.2, dont j'ai vu une démonstration, est compatible HP48SX et HP48GX et peut être mise dans l'un quelconque des ports de la HP48GX. La dernière version a encore un bug connu (au moment où j'écris : problème lorsque les batteries sont faibles et que l'on éteint la machine) qui devrait disparaître avec la version suivante. En attendant celle-ci voici les coordonnées de l'auteur pour qui voudrait souscrire (50 francs) et recevoir la disquette avec la documentation complète plus des suppléments (fontes, utilitaires, jeux ainsi que les anciennes versions 4.1 de 17 Ko) :

Jean-Yves Avenard  
16 rue de Savigny  
91390 Morsang sur Orge

Je signale que je ne connais l'auteur que très indirectement mais il me semble qu'il est utile de révéler l'existence des outils performants qui, s'ils sont bien connus des HPistes parisiens le sont moins des provinciaux isolés pour qui le recours aux serveurs est très onéreux.

A votre tour de nous donner des informations sur d'autres produits que vous connaissez.

Guy Toublanc (276)

---

## TRADUISEZ LES MESSAGES !

Les messages d'erreur ou d'environnement (STAT, PLOT, TIME...) utilisés par la machine sont ceux des librairies de numéros 0, 1, 2, 3, 5, 6, A(10), B(11), C(12) et D(13). Hormis la librairie 2, ce ne sont que des fragments de librairie qui, il me semble, ne

servent qu'à générer les messages. Rappelons au passage que les messages sont stockés dans toutes les librairies sous forme de tableaux de chaînes (*array of string*).

Pour l'utilisation de la table des messages d'une de ces librairies, deux cas se présentent :

- ou bien la librairie n'est pas attachée au répertoire courant : la machine se sert alors directement de l'adresse en ROM (Les différentes adresses de ces tables sont #72000h, #72704h, #72DCfh, #72F1Eh, #736F9h, #72FE6h, #72281h, #726A5h, #7232Ch et #7260Ah).

- ou bien la librairie est attachée au répertoire courant : la machine se sert alors de l'adresse de la table dans l'objet-répertoire courant (après le numéro de lib et l'adresse de la hash-table)

Pour mettre les messages d'une de ces librairies en français, il suffit donc d'attacher cette librairie au répertoire courant (HOME de préférence) et de remplacer l'adresse de sa table des messages par l'adresse d'un autre tableau de chaînes.

Cela peut se faire par le programme :

```
PROG
<n de lib>   Ce programme attache la librairie
$ 07709     dans HOME.
<n de lib>
ARR1
$ 0746E
END
```

Remarque : Il est très important que ARR1 ait une adresse fixe. Il faut donc que ce tableau soit stocké dans une variable ou un nom XLIB.

Ne faites pas comme moi qui, devant la taille des messages (environ 4Ko), avais essayé de les compacter.

L'adresse #0746Eh prend en argument le numéro de librairie sous forme d'entier-système et un tableau de chaînes à adresse fixe (pas de NEWOB). Elle remplace l'adresse de la table des messages de la librairie dans le répertoire HOME par celle de ARR1.

Pour remettre les messages en anglais, il suffit de détacher la librairie :

```
PROG
<n de lib>   Ce programme détache la librairie
$ 076AE     de HOME.
END
```

## Cas particulier : la librairie 2

Elle contient des fonctions RPL-utilisateur. Dans le premier programme, il n'y a donc pas besoin de l'attacher (elle s'est déjà attachée grâce à sa config), et dans le second il faut la réattacher après l'avoir détachée.

Pour convertir une liste de messages en un tableau de chaînes, le plus simple est encore d'utiliser LIBMAKER 3.2, de BouHP (sur la Goodies Disk 7).

Remarque : cette méthode ne permet pas franciser les mots "Array of" qui, suivis du nom en français d'un type d'objet, apparaissent pour désigner un tableau d'objet de ce type. "Array of string" apparaît sous la forme "Array of Chaîne". Je n'ai pas encore trouvé de remède à cela.

Arthur Ripoll (590)

---

## UNE LIBRAIRIE TRES ATTACHANTE

Cet article a pour but de présenter une librairie présentant la particularité d'être totalement en écriture en assembleur.

Mais cette librairie aura une fonction particulière puisqu'elle sera articulée autour de son config (programme à exécuter lors d'un Arrêt Système [ON] [C]).

Pour tout dire, ce config permettra de choisir les autres librairies que l'on voudra Attacher (en les Attachant lui même pour ne pas exécuter leurs configs), ou Détacher (en ne les Attachant pas et en n'exécutant pas leurs configs).

Il faut savoir que lors de l'appel de tout config il y a sur la pile tous les numéros des libs, dont on n'a pas encore exécuté le config, sous forme d'entiers système, plus le nombre total de ces numéros.

Il suffit de remplacer certains de ces numéros par un numéro fixé (par exemple celui d'une librairie en ROM qui n'a pas de config, ici la lib numéro 3) pour que leur config soit "détournée"...

Cette librairie s'appelle Config Picker.

LIBATTACH prend en argument une liste de numéros de librairie à attacher (même si vous mettez un numéro de librairie inexistante, il n'y a pas de problème).

LIBDETACH prend une liste de numéros de librairie à ne pas attacher (même si vous mettez un numéro de librairie inexistante, il n'y a pas de problème).

Si vous voulez autoriser tous les configs, faites :

```
{ 0 } LIBATTACH { 0 } LIBDETACH
```

Si vous voulez attacher toutes les librairies, il suffit d'armer le flag 62.

Si vous voulez que Config Picker s'attache, désarmez le Flag 63, si celui-ci est armé, Config Picker sera détaché...

CHANGEN°LIB prend un numéro de librairie inférieur à 1792 et renumérote la librairie (attention ne pas donner le numéro d'une autre lib existante ou il risque d'y avoir des conflits) puis reboote. Notez que j'ai rendu impossible l'accès de cette fonction par la ligne de commande en ne mettant pas d'offset sur les noms de 11 caractères...

Bug Connu : si vous donnez à votre librairie les numéros 4, 7, 8 ou 9 et que le flag 62 est armé, il apparaît que la HP48G/X (version M) se rallume une seconde après l'avoir éteinte... (La mienne est restée allumée comme cela environ une heure avant que je le voie !!!). De plus, pour tous numéros inférieurs à 242, si le flag 62 est allumé, les librairies supérieures en Rom sont attachées, ce qui occasionne une lenteur lors de l'accès au menu Library.

Le listing étant assez long je vous laisse découvrir par vous même la structure de cette lib. J'ai essayé de faire un effort sur les commentaires... car ce n'est pas mon point fort.

```
ASSEMBLE
```

```
TOUTDEB
```

```
CON(5) #02B40 * =DOLIB
```

```
TAIL_LIB
```

```
REL(5) FinL
```

```
CON(2) 17
```

```
NIBASC 'Config Picker !!!'
```

```
CON(2) 17
```

```
No1
```

```
CON(3) 1642 * Numéro de librairie
```

```
REL(5) HashTab * Offset sur Hash-Table
```

```
REL(5) MsgTab * Offset sur Table des messages
```

```
REL(5) LinkTab * Offset sur Table des liens
```

```
REL(5) Config * Offset sur Config
```

Config

CON(5) #02D9D \* =DOCOL  
 CON(5) #02DCC \* =DOCODE  
 REL(5) FinConfigCod  
 GOSBVL #0679B \* =SAVPTR  
 C=0 A  
 R2=C.F A \* nombre de lib a attacher  
 A=DAT1 A  
 AD1EX  
 R4=A.F A \* sauvegarde @ pile  
 D1=D1+ 5  
 C=DAT1 A \* nombre de configs restant  
 AD1EX  
 D1=D1+ 5  
 D0=(5) #537D7 \* @ @ drapeaux user  
 A=DAT0 A  
 D0=A  
 D0=D0+ 15  
 A=DAT0 B  
 C=0 A  
 ?ABIT=0 1  
 GOYES Flag\_62\_Clear

FLag\_62\_Set

A=DAT1 A  
 ?A#0 A  
 GOYES NoFin  
 GOTO FIN

NoFin

D0=A  
 D0=D0+ 5  
 A=DAT0 A  
 CD1EX  
 R3=C.F A \* sauvegarde D1  
 R0=A.F A  
 GOSBVL #06537 \* =PUSH# : depose numero  
 GOSBVL #0679B \* =SAVPTR  
 C=R3.F A  
 CD1EX \* restore D1  
 C=R2.F A  
 C=C+1 A  
 R2=C.F A \* 1 de + a attacher  
 LCHEX 0400D  
 DAT1=C A \* vire le config de la lib  
 D1=D1+ 5  
 GOTO FLag\_62\_Set

Flag\_62\_Clear

BCL R1=C.F A  
 ?C=0 A \* fin des configs  
 GOYES FIN  
 A=DAT1 A  
 AD1EX  
 D1=D1+ 5  
 C=DAT1 A \* numéro de lib  
 AD1EX  
 B=C A  
 A=PC  
 CON(2) #43  
 REL(5) ATTACH

A=A+C A  
 D0=A  
 D0=D0- 1 \* libs a attacher  
 D0=D0+ 3  
 A=DAT0 X  
 ?A=0 X \* tous les numéros ont  
 GOYES NOMOR \* ete examinés  
 ?A#B X  
 GOYES B1  
 GOTO REPL1 \* les numéros correspondent  
 FIN C=R2.F A  
 C=C+1 A  
 R1=C.F A \* nombre de lib a attacher  
 D0=(5) #537D7 \* @ @ drapeaux user  
 A=DAT0 A  
 D0=A  
 D0=D0+ 15  
 A=DAT0 B  
 C=0 A  
 ?ABIT=1 2  
 GOYES Flag\_63\_Set  
 CON(2) #43

No15 CON(5) 1642 \* attache Config Picker  
 PUSH R0=C.F A  
 GOSBVL #06529 \* =PUSH2#  
 GOVLNG #2D564 \* LOOP

Flag\_63\_Set

LC(1) 2 \* n'attache pas Config Picker  
 GOC PUSH

NOMOR

A=PC  
 CON(2) #43  
 REL(5) NOTATTACH  
 A=A+C A  
 D0=A

B2

D0=D0+ 2 \* libs a detacher  
 A=DAT0 X  
 ?A=0 X \* tous les numéros ont  
 GOYES NOMOR2 \* été examinés  
 ?A=B X  
 GOYES REPL2 \* les numéros correspondent  
 D0=D0+ 3  
 GONC B2  
 NOMOR2 D1=D1+ 5  
 C=R1.F A  
 C=C-1 A \* nombre de configs restant  
 GOTO BCL

REPL1

A=0 A  
 A=DAT0 X \* numéro lib à attacher  
 CD1EX  
 R3=C.F A \* sauvegarde D1  
 R0=A.F A  
 GOSBVL #06537 \* =PUSH# dépose numéro  
 GOSBVL #0679B \* =SAVPTR  
 C=R3.F A

CD1EX		* restore D1	No4	CON(3)	1642	* Entier Binaire des
C=R2.F	A			CON(3)	6	* libs a attacher
C=C+1	A			CON(5)	#02E92	* =DOROMP
R2=C.F	A	* 1 de + à attacher				
REPL2	LCHEX	0400D	No5	CON(3)	1642	
DAT1=C	A	* vire le config de la lib		CON(3)	5	* modifie Entier Binaire
GOTO	NOMOR2			CON(5)	#03258	* =2DROP efface list et Ent Bin
				CON(5)	#02E92	* =DOROMP
XLIB6	CON(5)	#02A4E * =DOHSTR	No6	CON(3)	1642	
REL(5)	FAT			CON(3)	3	* Recalcul CRC
ATTACH				CON(5)	#0312B	* =SEMI
CON(3)	0	* nombre		CON(5)	#0312B	* =SEMI
CON(3)	0	* de xlib				
:		* que vous		CON(1)	8	* commande
:		* voulez				
CON(3)	0	* attacher-1	No7	CON(3)	1642	* numéro de lib
FAT				CON(3)	1	* numéro de xlib
XLIB7	CON(5)	#02A4E * =DOHSTR	XLIB1	CON(5)	#02D9D	* =DOCOL
REL(5)	FNAT			CON(5)	#18ECE	* =CK1&Dispatch
NOTATTACH				CON(5)	#04021	* =FIVE
CON(3)	0	* nombre		CON(5)	#02D9D	* =DOCOL
CON(3)	0	* de xlib		CON(5)	#02E92	* =DOROMP
:		* que vous				
:		* voulez				
CON(3)	0	* detacher-1	No13	CON(3)	1642	
FNAT				CON(3)	8	* Controle Port2 ?
FinConfigCod				CON(5)	#02E92	* =DOROMP
CON(5)	#073C3	* =ZERO_DO boucle	No16	CON(3)	1642	
CON(5)	#07709	* =TOSRRP attache		CON(3)	4	* ListReel>ListSysBinary
CON(5)	#07334	* =LOOP libs		CON(5)	#02E92	* =DOROMP
CON(5)	#0312B	* =SEMI				
MsgTab	CON(5)	#029E8 * =DOARRY * Array	No8	CON(3)	1642	
REL(5)	FinMT			CON(3)	7	* Entier Binaire DETACH LIB
CON(5)	#02A2C	* =DOCSTR * of String		CON(5)	#02E92	* =DOROMP
CON(5)	1	* Dimension Tableau	No9	CON(3)	1642	
CON(5)	1	* Nombre de MSG		CON(3)	5	* modifie Entier Binaire
CON(5)	43	* Taille Chaîne		CON(5)	#03258	* =2DROP efface list et Ent Bin
NIBASC	'Library In Bad Port'	* Msg d'erreur 01		CON(5)	#02E92	* =DOROMP
FinMT	CON(1)	8 * commande	No10	CON(3)	1642	
No3	CON(3)	1642 * numéro lib		CON(3)	3	* Recalcul CRC
	CON(3)	0 * numéro xlib		CON(5)	#0312B	* =SEMI
XLIB0	CON(5)	#02D9D * =DOCOL		CON(5)	#0312B	* =SEMI
	CON(5)	#18ECE * =CK1&Dispatch				
	CON(5)	#04021 * =FIVE prend une liste		CON(1)	8	* commande
	CON(5)	#02D9D * =DOCOL en argument	No11	CON(3)	1642	* numéro de lib
	CON(5)	#02E92 * =DOROMP		CON(3)	2	* numéro de xlib
No12	CON(3)	1642 * Controle que l'on	XLIB2	CON(5)	#02D9D	* =DOCOL
	CON(3)	8 * est pas en port 2		CON(5)	#18ECE	* =CK1&Dispatch
	CON(5)	#02E92 * =DOROMP		CON(5)	#03FF9	* =ONE un reel
No17	CON(3)	1642		CON(5)	#02D9D	* =DOCOL
	CON(3)	4 * ListReel>ListSysBinary		CON(5)	#18CEA	* =COERCE Reel>SysBinary
	CON(5)	#02E92 * =DOROMP		CON(5)	#02E92	* =DOROMP

No14 CON(3) 1642  
 CON(3) 8 \* Lib en Port 2 ?  
 CON(5) #02DCC \* =DOCODE  
 REL(5) FinC4  
 GOSBVL #06641 \* =POP#  
 R0=A.F A  
 GOSBVL #0679B \* =SAVPTR  
 LCHEx 00700  
 ?A<C A  
 GOYES NOFIN3  
 GOTO FIN3 \* numéro de lib trop grand

NOFIN3 GOSUB DON.No  
 REL(5) No1 \* liste  
 REL(5) No2 \* d'offset  
 REL(5) No3 \* sur numéros  
 REL(5) No4 \* de la lib  
 REL(5) No5 \* dans  
 REL(5) No6 \* la lib  
 REL(5) No7  
 REL(5) No8  
 REL(5) No9  
 REL(5) No10  
 REL(5) No11  
 REL(5) No12  
 REL(5) No13  
 REL(5) No14  
 REL(5) No15  
 REL(5) No16  
 REL(5) No17  
 CON(5) 0

DON.No C=RSTK  
 D0=C  
 BCL2 A=DAT0 A  
 ?A=0 A \* liste finie  
 GOYES FIN3  
 CDOEX  
 A=A+C A \* @ numéro  
 D0=A  
 A=R0.F A  
 DAT0=A X \* change le numéro  
 D0=C  
 D0=D0+ 5 \* offset suivant  
 GOTO BCL2

FIN3 GOTO LBCRC \* recalcule CRC

FinC4 CON(5) #21B4E \* WarmStart (ON-C)  
 CON(5) #0312B \* =SEMI  
 CON(5) #0312B \* =SEMI

XLIB3 \* recalcule CRC  
 CON(5) #02DCC \* =DOCODE  
 REL(5) FinC3  
 GOSBVL #0679B \* =SAVPTR

LBCRC A=PC  
 CON(2) #43  
 REL(5) TAIL\_LIB  
 A=A+C A  
 D0=A  
 D0=D0+ 2  
 A=DAT0 A \* taille lib  
 A=A-CON A,4  
 CDOEX  
 GOSBVL #0597E \* =DoCRCc  
 DAT0=A 4 \* nouveau CRC  
 GOSBVL #05143 \* =GETPTRLOOP

FinC3

XLIB4 \* ListReel>ListSysBinary  
 CON(5) #02D9D \* =DOCOL  
 CON(5) #054AF \* =INNERCOMP  
 CON(5) #61891 \* =DUP#0=case  
 CON(5) #11006 \* WRONGCOUNTERR  
 CON(5) #03188 \* =DUP  
 CON(5) #073C3 \* =ZERO\_DO  
 CON(5) #628EB \* =DUP#1+  
 CON(5) #03325 \* =ROLL  
 CON(5) #62169 \* =DUPTYPEREAL?  
 CON(5) #63B46 \* =NcaseTYPEERR  
 CON(5) #18CEA \* =COERCE  
 CON(5) #06657 \* =TOTEMPOB  
 CON(5) #03223 \* =SWAP  
 CON(5) #07334 \* =LOOP  
 CON(5) #05459 \* =( )N  
 CON(5) #0312B \* =SEMI

XLIB5 \* modifie Entier Binaire

CON(5) #02DCC \* =DOCODE  
 REL(5) FinCod1  
 GOSBVL #0679B \* =SAVPTR  
 A=DAT1 A \* @ Entier Bin  
 D1=D1+ 5  
 C=DAT1 A \* @ liste SB  
 D1=A  
 D0=C  
 D1=D1+ 5  
 A=DAT1 A  
 D1=D1+ 5  
 B=A A \* taille Ent Bin  
 B=B-CON A,10 \* - taille - vide - premier + 1  
 GOC FIN1  
 D0=D0+ 5  
 BCL1 A=DAT0 A  
 LCHEx 02911  
 ?A#C A  
 GOYES FIN1  
 D0=D0+ 5  
 A=DAT0 X  
 DAT1=A X  
 D1=D1+ 3  
 D0=D0+ 5  
 B=B-CON A,3

```

GOC    FIN1
GOTO   BCL1

FIN1   C=0    A
      DAT1=C  X
      GOSBVL #05143 * =GETPTRLOOP

FinCod1

XLIB8          * Controle Lib en Port 2 ?
              * => Erreur

CON(5) #02DCC * =DOCODE
REL(5) FinCod4
A=PC
CON(2) #43
REL(5) TOUTDEB
A=A+C  A
ADOEX
D0=D0+ 2      * @ debut de lib
C=DAT0  A
ADOEX
A=C      A
LCHEX   02B40
?A=C    A      * si le prologue est bon
GOYES   NOTP2  * on est pas en port 2
CON(4) #0143  * LC NoErreur

No2     CON(3) 1642
      A=C      A      * sinon Erreur
      GOVLNG #05023 * =Errjmp

```

```

NOTP2 LOOP
FinCod4

```

```

HashTab
CON(5) #02A4E * =DOHSTR
REL(5) FinHashTab
CON(5) 0
REL(5) NOM_DE_9_CHR
CON(5) 0
CON(5) 0      * pas d'offset a 11
CON(5) 0
CON(5) 0
CON(5) 0
CON(5) 0
REL(5) Offset1
NOM_DE_9_CHR
CON(2) 9
NIBASC 'LIBATTACH'
CON(3) 0
CON(2) 9

```

```

NIBASC 'LIBDETACH'
CON(3) 1
NOM_DE_11_CHR
CON(2) 11
NIBASC 'CHANGEN°LIB'
CON(3) 2

Offset1
CON(5) #00049 * Offset negatif ( -REL(5) )
CON(5) #00037 * on doit les calculer
CON(5) #00025 * a la main ...

FinHashTab

LinkTab
CON(5) #02A4E * =DOHSTR
REL(5) FinLinkTab
REL(5) XLIB0 * offset
REL(5) XLIB1 * sur
REL(5) XLIB2 * tous
REL(5) XLIB3 * les
REL(5) XLIB4 * xlibs
REL(5) XLIB5 * de
REL(5) XLIB6 * la
REL(5) XLIB7 * librairie
REL(5) XLIB8

FinLinkTab

CRC    CON(4) #0000 * CRC
FinL
RPL

```

Une fois la librairie assemblée, il faut exécuter ce petit programme pour calculer pour la première fois le CRC (librairie au niveau 1, lancer ce programme).

```

CON(5) #02DCC * =DOCODE
REL(5) FinCode
GOSBVL #0679B * =SAVPTR
A=DAT1  A
D0=A
D0=D0+ 5
A=DAT0  A
A=A-CON A,4
CDOEX
GOSBVL #0597E * =DoCRCc
DAT0=A  4
GOSBVL #05143 * =GETPTRLOOP

```

FinCode

Maintenant vous pouvez stocker la librairie en Port.

La librairie donnée dans le coin des codes permet de donner des listes de 30 librairies à attacher ou à détacher. NDLR : Suite aux délais de mise en pages de ce JPC, le listing du "Coin des codes" est soumis aux mêmes remarques que celui cité dans l'article "Voyage... Code Nul". Référez vous donc à cet article pour un bon transfert sur votre HP48.

Pierre Silvestre de Sacy (572).

## CHOC EN RETOUR

Dans l'article *programme bizarre ...* (JPC-90 p16) et dans le paragraphe *principe du programme* au lieu de:

'ni1/di1'

il fallait lire:

'nI/dI'

et ainsi de suite

J'ai pu constater que l'utilisation de ASC-, qui parut dans JPC-79 page 11, posait parfois des problèmes avec certaines structures de programmes. Cela est dû au début du programme. Pour que se soit correct il faut ajouter CK1NoBlame juste devant CK&DISPATCHO. Le début du fichier source devient donc:

```
::  
CK1NoBlame  
CK&DISPATCHO  
.....
```

Pour ne pas retaper le code vous pouvez utiliser le programme suivant:

```
MODIFASC-  
cksum # 9984h 131 octets  
«  
  'ASC-' RCL BYTES DROP # FC91h #  
  IF THEN # 203h DOERR END  
  "D9D20D2951" 'ASC-' RCL  
  -'ASC 6 OVER SIZE SUB +  
  ASC-  
»
```

Sinon le code se trouvera dans le coin des codes.

Guy Toublanc (276)



## HP95 / HP100

J. Belin	<i>PC de Poche : HP100 &amp; 95LX</i>	22
J. Belin	Configuration commune des cartes Ram	24
J. Belin	Conversion d'images (Acte II)	26
J. Belin	Commande de sélection de mode écran	27
	<b>Le coin des codes</b>	<b>29</b>

## PC DE POCHE : LE LIVRE

Alors que de nombreux livres ont été écrits en France pour les HP28 et HP48, les HP95 et HP100 avaient jusqu'à présent été oubliés.

Cette omission est maintenant corrigée, puisque un livre dédié à ces machines est maintenant disponible. Son titre est *PC de Poche : HP100 & 95LX*. Écrit par Luc Smeesters, il est édité par Dunod au prix de 145 Francs.

Cependant, alors que les livres consacrés aux machines précédentes dépassent souvent les 500 pages et sont plutôt orientés sur la programmation sous toutes ses formes, celui-ci ne dépasse pas les 170 pages et se destine plutôt (ainsi que le dit l'auteur dans son introduction) aux utilisateurs débutants et aux acheteurs potentiels.

Ensuite, le livre est constitué de deux parties principales : La première présente les machines et l'utilisation des applications intégrées. La seconde partie contient une présentation de produits et logiciels adaptés aux deux machines.

Parcourons maintenant les différents chapitres.

### **Palmtops : Pour qui ? Pour Quoi ?**

Sous ce titre, ce chapitre retrace en fait l'histoire et l'évolution des ordinateurs de poche, ainsi que leur différences fondamentales par rapport aux ordinateurs portables. Les Canon X07, Psion Organizer, Poqet, Atari Portfolio, Psion Série 3, HP95, Sharp PC3000 et HP100LX y sont donc brièvement décrits par ordre chronologique. Ensuite l'auteur décrit brièvement ce que pourrait être le futur, avec l'apparition récente des PDA.

Ma seule observation dans cette partie serait peut-être que le HP-71 a été oublié dans ce panorama. Après tout, il possède sûrement plus le titre d'ordinateur de poche que le Psion Organiser, et c'est aussi un peu l'ancêtre du HP95...

### **Compagnon du PC**

Après avoir bien dit qu'il était illusoire de demander à un palmtop les mêmes services et performances qu'à un PC, l'auteur décrit de façon assez détaillée les deux HP (Ecrans, claviers, Communications...). Cependant, et c'est peut être un oubli de sa part, il n'y a aucun mot dans cette partie sur l'unité centrale (types de microprocesseurs...). Cela aurait peut être été intéressant pour un débutant.

Par contre, la partie consacrée à l'alimentation électrique est bien traitée, avec quelques courbes de décharge des types de piles et des conseils d'utilisations.

Ce chapitre contient aussi une courte description de la norme PCMCIA et donne les différents types et versions admises pour chaque machine. Pour une fois, l'auteur n'est pas tombé dans le piège consistant à confondre *type* et *version*. Ce qu'on en peut pas dire pour de nombreux journalistes ou distributeurs...

Enfin, cette partie se termine par quelques mots concernant la protection de la machine par password, avec quelques conseils sur la façon de les choisir.

### **Logiciels Intégrés**

Après une petite explication sur ce qu'est un "ordinateur" (son mode de fonctionnement, ses différents composants...) et du vocabulaire de base du MS-Dos, l'auteur décrit brièvement l'interface du Système Manager, pour les deux machines.

Ensuite, il nous décrit de façon assez détaillée les différentes applications intégrées : Setup, Filer, Editeur, Agenda...

Bien sûr, pour un utilisateur expérimenté, cette partie semblera au premier coup d'oeil n'être qu'un résumé des notices d'utilisations des deux machines.

Cependant, nous avons ici (en plus des renseignements de base) de nombreux conseils pratiques ou trucs d'utilisation qui intéresseront souvent des utilisateurs expérimentés. Par exemple, la suggestion d'une petite syntaxe normalisée dans les fiches de l'agenda. Il y a aussi un petit script permettant de remplacer le bip d'alarme par l'hymne Européen ! Contenant malheureusement une petite fausse note que je vous laisse trouver par vous même...

Mon seul regret dans cette partie serait qu'il aurait été intéressant, pour les utilisateurs de HP100, d'y voir une initiation au RPN, puisque celle-ci a disparue de la notice de cette machine.

### **Interconnexions**

Ce chapitre décrit succinctement (un peut trop, peut être) les différents accessoires permettant de connecter les palmtops HP à des ordinateurs : Les kits de connections HP (émulation des machines sur PC, conversions des fichiers HP95/HP100...), la transmission par infra-rouge avec les Compatibles HP-XM et le kit de connection HP-MacIntosh de Dataviz.

## Accessoires Matériels

Pour les utilisateurs expérimentés, ce chapitre et les suivants seront probablement ceux qui les intéresseront le plus. En effet, même les lecteurs assidus du catalogue EduCALC y feront quelques découvertes. De plus, de nombreuses photos montrent l'aspect de certains d'entre eux.

Suivant une petite présentation de deux étuis de protection de la machine, l'auteur présente en quelques mots les différents types de mémoire (Rom, Ram, Flash...).

Ensuite, suit une présentation de différents produits ou technologies dont les lecteurs de disquettes portables, les cartes d'interface PCMCIA/SCSI ou les lecteurs de cartes sur PC, puis quelques imprimantes portables (Diconix, Citizen PN48 et BJ10), avec les indispensables convertisseurs série/parallèle.

Une autre grande section est ensuite consacrée aux modems, avec d'abord une bonne introduction technique suivie d'un petit exposé des différences existant entre le HP95 et le HP100 dans ce domaine. Pour la première machine, le WorldPort d'US Robotics et le modem PCMCIA de NewMedia sont bien sûr cités. Ensuite nous trouvons quelques mots à propos des autres types de moyens de communications tels que le "pager" de Motorola.

Suivent ensuite quelques périphériques plus exotiques, tels que le *Linalyzer* de ByteCom que je ne connaissais pas. Il s'agit d'une interface permettant au HP95 d'analyser le comportement et la précision d'une alimentation secteur 220V. L'intérêt du HP95 étant que son alimentation par pile l'isole totalement de la source électrique et lui permet de continuer de travailler en cas de coupure du secteur. D'autres périphériques sont cités, comme les lecteurs de code-barre ou de cartes PCMCIA, et les systèmes de réception radio (Alphapage ou système GPS).

## Programmes d'applications

En ce qui concerne les logiciels, l'auteur cite tout d'abord quelques applications commerciales (*Pocket Sales Forces*, *ACT!*...), indique que le monde médical est aussi touché par la "HP-ite aiguë" et présente *Derive* en quelques mots. Puis au cours d'une courte présentation des langages développés pour le HP95, il a la bonne idée de parler du travail de notre ami Craig Finseth (l'auteur de MEMUTIL et FREYJA) et de laisser ses coordonnées.

Pour terminer (et après avoir dit qu'on pouvait aussi jouer sur son HP !), il nous indique qu'il est possible

d'obtenir de nombreux programmes par l'intermédiaire de serveurs tels que Compuserve.

Cependant, une première lecture de ces trois derniers chapitres donne une sensation de manque à un utilisateur averti. Une rapide relecture lève rapidement le doute : certaines sociétés parmi les plus importantes dans le monde HP, telles que SPARCOM, ont été totalement oubliées. Par exemple pour cette société (dont Maubert importe la majorité des produits), on omet de citer son kit de connection MacIntosh, alors qu'on cite le MacLinkPlus de Dataviz, qui est 40% plus cher... De même il n'y a rien sur le *Drive95* (qui est disponible chez plusieurs revendeurs Français dont la Fnac) alors qu'on parle à la place du *HandyDisc* qui est beaucoup plus difficile à trouver. D'autres sociétés importantes ont aussi été oubliées, par exemple ACE et TDS... De même, en ce qui concerne les lieux où l'on trouve des programmes, il ne semble connaître ni le *Palmtop Paper* ni l'existence des clubs utilisateurs qui sont pourtant d'un coût de revient très inférieur à CompuServe...

## Annexes

Enfin, on trouve en annexe quelques renseignements intéressants tels que le brochage des connecteurs séries ou des informations sur les codes de commande des modems.

Cependant, certaines de ces annexes laissent un peu perplexe. Par exemple, la première a la bonne idée de présenter les fichiers APNAME.LST, choses absentes dans la documentation fournie par HP. Le problème est que la création de ces fichiers est déjà difficile à maîtriser (tous les utilisateurs peuvent en témoigner !) et que l'auteur non seulement ne donne aucun conseil réel sur eux (par exemple le fait qu'il est indispensable de taper les noms de fichiers en majuscules), mais ajoute des éléments pouvant induire le pauvre utilisateur en erreur !

La première erreur est de présenter la syntaxe de la ligne du fichier ainsi :

```
[Unité:Répertoire\Programme.EXM,[codeTouche],[Descripteur]][RC]
```

Tout informaticien débutant sait que les crochets servent à indiquer un paramètre optionnel. Le problème est que sur le HP95, tous les paramètres de la ligne sont obligatoires !

La seconde chose pouvant induire les utilisateurs en erreur se trouve dans la liste des codes de touche. Si ces codes en eux-mêmes sont exacts, il aurait fallu indiquer (par une marque) les touches qui peuvent être effectivement utilisables comme "Hot Key" dans

APNAME.LST. Car certaines d'entre elles (par exemple la plupart des combinaisons [shift]+Touches bleues) ne sont pas utilisables dans ce fichier.

Enfin la dernière partie, donnant une liste de fournisseurs de logiciels et périphériques, laisse tout aussi perplexe. En effet, si on y voit de nombreuses sociétés Européennes (dont certaines me sont inconnues), comment ce fait-il qu'aucune société Américaine n'y soit directement mentionnée ? Sparcom, ACE ou TDS sont pourtant les principales compagnies fournissant des extensions pour ces machines ! Bizarre. De plus, à moins de connaître ces sociétés, il n'est pas vraiment possible de savoir exactement qui fait quoi car si le nom des produits est cité dans les présentations, la société productrice l'est rarement. Ce qui rend impossible l'utilisation de cette liste. Par exemple, il est cité (dans sa présentation) que CONTACT95 est créé par des Français. Heureusement qu'il n'y a qu'une seule société Française conceptrice de logiciels dans cette liste, car autrement comment en déduire que Dynaxx en est la source ? En fait cette déduction nous est simplifiée par le fait que les logiciels financiers de Bruckert sont brièvement décrits, mais sans aucune mention de ce nom, que ce soit dans la description ou dans la liste... Enfin, certaines produits et sociétés cités (Compuserve, par exemple) n'y sont pas du tout listés, alors qu'elles ont souvent une adresse en France.

En résumé, ce livre devrait servir son but : inciter les acheteurs potentiels et aider les débutants. Il devrait aussi bien aider à faire comprendre les différences entre le HP95 et le HP100 en ce qui concerne les différences dans les possibilités internes et les extensions possibles. Il devrait même intéresser de nombreux utilisateurs expérimentés par les petits trucs qui y sont cités dans la première partie. Par contre, pour ces mêmes utilisateurs, au courant des différents accessoires diffusés, ses nombreuses omissions et erreurs dans la deuxième partie ne peuvent que donner une impression de manque de préparation, et peut-être la présence d'un certain parti-pris. Et c'est dommage, car l'idée de base était bonne...

Jacques Belin (123)

## DUEL DE CARTES HP95 / HP100

Au cours de notre dernière réunion, un de nos utilisateurs de HP95, se trouva fort ennuyé lorsque il voulut booter, avec son nouvel HP100, sur sa carte Ram configurée pour son HP95,

Le symptôme consécutif à cette manoeuvre était l'apparition du message suivant :

```
Bad or missing Command Interpreter
```

N'ayant pas alors le temps de chercher la cause de ce problème, ma première réaction fût d'utiliser la possibilité qu'a le HP100 de laisser choisir le disque de boot, afin de le faire redémarrer à partir du disque Rom.

Ce n'est que le lendemain, repensant à ce problème que j'en trouvais la cause exacte. L'erreur se trouvait dans le fichier CONFIG.SYS, à la ligne :

```
SHELL=C:\COMMAND.COM /E /P:512
```

En effet, si cette ligne est valide sur le HP95, elle ne peut en aucun cas fonctionner sur le HP100 puisque le fichier COMMAND.COM est maintenant situé dans le répertoire \DOS de l'unité D !

Ceci m'ammena à la question suivante : comment faire en sorte que les personnes ayant accès aux deux machines puissent booter sur la carte de façon transparente ?

Si cette question peut paraître une hypothèse d'école (puisque'il semble peu logique que les possesseurs des deux machines continuent à utiliser leur HP95). Cependant, cela pourrait être intéressant dans le cadre de nos réunions, ou de nombreux échanges peuvent se faire. Cet article a donc pour but de montrer une méthode de configuration, en espérant que les habitués en tiennent compte.

### Adaptations du fichier CONFIG.SYS

MS-DOS ne permettant pas de faire des tests ou des sauts dans le fichier CONFIG.SYS, il n'y a pas de moyen d'influer sur son déroulement. Nous n'avons donc aucun moyen de sélectionner les bonnes coordonnées du fichier...

Une solution pouvant venir à l'esprit serait de copier le fichier COMMAND.COM sur la carte et de modifier

le CONFIG.SYS en conséquence. En fait, cette solution est à éviter, pour plusieurs raisons :

- Le fichier COMMAND.COM est dépendant de la version du DOS. Les deux HP ayant une version différente du système d'exploitation, il y aura donc une incompatibilité pour les deux machines.

- Même si ce problème ne se posait pas, il faut savoir que (sur les deux HP), ce programme n'est en fait qu'un "Loader", faisant un saut dans la Rom, où se trouve réellement le code du programme. L'adresse de destination du saut n'étant pas la même suivant les machines (peut être même suivant les versions d'une même machine), le plantage est assuré.

- Enfin, si vous déclarez le fichier COMMAND.COM sur la carte et que vous tentez d'utiliser la machine lorsque cette carte est absente (ou que vous ayez une autre carte insérée), vous aurez aussi un plantage.

Afin de contourner ces problèmes, j'ai donc choisi une solution radicale : copier le fichier COMMAND.COM du HP100 dans le répertoire racine du disque C. Cela n'est peut être pas élégant, puisqu'il faut "préparer" le HP100 récepteur de la carte, mais c'est ce qu'il y a de plus simple. De toutes façons, ce fichier ayant une taille de moins de 512 octets, elle est probablement la plus économique.

Un autre partie du fichier HP95 peut sembler poser un problème. Il s'agit de la ligne :

```
COUNTRY=33
```

Lors de la sequence boot, le HP100 affiche fugitivement le message :

```
Bad or missing \COUNTRY.SYS
```

En fait, nous pouvons ignorer ce problème, car le HP100 gère les informations nationales à l'aide du programme KEYBEZ, pouvant être lancé dans le fichier AUTOEXEC.BAT (voir page 22-13 de la doc du HP100, ou l'exemple plus bas).

Notez cependant qu'il existe un problème que je n'ai pas réglé, c'est la désactivation de l'installation de certains drivers sur l'une ou l'autre des deux machines. Par exemple, 95BUDDY qui n'est pas utilisable sur le HP100. Quelqu'un aurait-il une idée ?

### Modifications du fichier AUTOEXEC.BAT

Si il n'est pas possible d'influer sur le fonctionnement du fichier CONFIG.SYS, nous n'avons pas ces problèmes pour le fichier AUTOEXEC.BAT. La question est ici : Comment savoir sur quelle machine est exécuté ce fichier ?

Il est possible d'utiliser un programme nommé 1S95 qui renvoie la valeur 1 si il est utilisé sur le HP95. Cette valeur pouvant être testée grace à la fonction ERRORLEVEL du Dos.

Cependant, je vous propose une autre méthode, qui possède l'avantage de ne pas nécessiter de programme extérieur. Elle consiste à utiliser un test du type IF [NOT] EXIST permettant d'effectuer une action suivant l'existence (ou l'absence) d'un fichier.

Dans le cas qui nous intéresse, il s'agit de faire ce test sur un fichier dont on sait qu'il n'est présent que sur une seule des deux machines. Notez que le type ou le contenu du fichier n'a aucune importance, puisque la seule chose qui nous intéresse est son existence.

Pour ma part j'ai porté ma préférence sur un des deux fichiers exemples de Lotus, présent dans le répertoire racine du disque C. Ces fichiers étant présents (à ma connaissance) dans toutes les versions du HP95 et absents du HP100, ils offrent toutes les garanties nécessaires.

Pratiquement, nous utiliserons une structure de programme telle que celle-ci :

```
. (Début du fichier)
.
.
IF NOT EXIST C:\_cf\low.wk1 goto hp100
:HP95
. (Code d'initialisation HP95)
.
.
GOTO common
:HP100
. (Code d'initialisation HP100)
.
.
:common
. (Suite de l'initialisation)
.
.
```

Notez bien que même si vous n'avez pas de suite pour l'utilisation, le label (et son instruction de branchement) doivent tous deux exister. Dans le cas contraire, le code destiné au HP100 serait exécuté par le HP95, à la suite de celui destiné à cette machine.

A titre d'exemple, voici un exemple de fichier sensiblement équivalent à celui que j'ai défini pour mon usage personnel :

```
@echo off
cls
PROMPT $P$G
```

```

IF NOT EXIST C:\_cflow.wk1 goto hp100
:HP95
PATH=A:\BIN;A:\BIN95;C:\;C:\_SYS;C:\DOS
GOTO common
:HP100
PATH=A:\BIN;A:\BIN100;D:\DOS;D:\BIN
keybez fr
:common
SET TMP=a:\TMP
SET TEMP=a:\TMP
serctl /o
ctype /ea
qshell95 -sx

```

Vous pouvez constater que dans les deux déclarations du PATH, j'ai spécifié (en plus d'un directory BIN), deux directories BIN95 et BIN100. Cela permet d'avoir des programmes ayant le même nom, mais spécifiques à chaque machine (des commandes DOS 3.2 et 5.0, par exemple).

Jacques Belin (123)

## CONVERSION D'IMAGES

### ACTE II

Le mois dernier, je vous ai présenté une méthode permettant de convertir des fichiers graphiques, afin de les visualiser sur le HP95.

Ce mois-ci, je vous propose de faire la même chose pour le HP100LX, toujours en utilisant le programme Graphic WorkShop.

Le principe général étant quasiment le même que pour le HP95, je ne me bornerai ici qu'à expliquer les différences pour le mode CGA. Je vous engage donc à vous reporter sur l'article du mois dernier pour l'explication de certaines étapes.

Cependant, alors que le HP95 ne supporte que des images monochromes, le HP100 et son mode CGA permettent aussi d'afficher des images en 4 niveaux de gris. Dans le premier cas la résolution est de 640x200, dans l'autre 320x200. Je vous expliquerai donc les étapes nécessaires permettant d'arriver aux deux formats de fichiers.

Notez aussi que nous devons actuellement considérer le format 4 niveaux de gris du point de vue théorique, car je n'ai pas trouvé actuellement de programme permettant d'afficher des fichiers 4 niveaux de gris sur le HP100. En fait, de nombreux utilitaires sont sensés le faire, mais ils ne produisent pas le résultats escompté ou plantent la machine. D'autres programmes devraient fonctionner, mais leur taille (plus de 100ko) est excessive pour la taille mémoire de la machine. Je poursuis donc mes recherches. Si vous possédez un tel utilitaire fonctionnant convenablement sur le HP100, n'hésitez pas à me prévenir. Dans le cas contraire, il faudra encore que je sorte mon assembleur, mais ceux qui sont venus aux dernières réunions savent que j'ai déjà commencé à travailler dans ce domaine...

D'autre part, j'ai constaté que pour certaines images, il était plutôt préférable les convertir en monochrome plutôt qu'en 4 niveau de gris, car ce format a quelques fois tendance à "écraser" tous les détails.

#### Conversion initiale

Comme pour les fichiers décrits dans l'article précédent, Il est conseillé de convertir l'image originale en format GIF 256 Couleurs.

#### Conversion en image tramée, 640x200

Tout d'abord, si vous désirez un petit aperçu du gain de résolution par rapport au HP95, reportez vous tout simplement aux illustrations du mois dernier, car la première image à la même résolution que ce que nous allons obtenir. Toutefois, l'écran du HP100 étant plus large que celui du HP95 (de 7 mm), nous aurons légèrement plus de pertes dues au passage en "Cinemascope".

#### Conversion en image anamorphosée, 640 colonnes

Même si la largeur de l'image originale est de 640 colonnes, il est toutefois nécessaire de passer par cette étape afin de l'anamorphoser.

Je vous avais présenté la dernière fois une formule permettant de connaître la hauteur de l'image intermédiaire. La voici pour le HP100 :

$$\text{lignes} = 640 * \frac{\text{nb\_lignes}}{\text{nb\_colonnes}} * \frac{0.86}{\text{format\_pixel}}$$

Ce qui donne pour les tailles courantes de fichiers :

320x200 -> image de 640x344  
640x400 -> image de 640x344

640×480 → image de 640×413

800×600 → image de 640×413

#### *Conversion en image 640×200*

Utilisez la même méthode que le mois dernier (fonction *Crop*).

#### *Conversion en image monochrome tramée*

Avant de convertir l'image en monochrome, rappelez-vous que vous pouvez toujours modifier certains paramètres de l'image (couleur, contraste) afin d'affiner la qualité de l'image finale. Ceci est aussi valable pour la conversion en 4 niveaux de gris.

Ensuite, il suffit d'utiliser la même méthode que le mois dernier : le *Dithering* (accessible par la touche [F3]).

#### **Conversion en image 320×200, 4 niveaux de gris**

Par rapport à la partie précédente, il y a deux différences : la résolution horizontale et l'étape de conversion des couleurs. N'oubliez pas d'utiliser la fonction *Crop* entre les deux.

#### *Conversion en image anamorphosée, 320 colonnes*

La méthode est strictement identique à précédemment, si ce n'est que la formule de conversion est ici :

$$\text{lignes} = 320 * \frac{\text{nb\_lignes}}{\text{nb\_colonnes}} * \frac{1.72}{\text{format\_pixel}}$$

En fait, le résultat est le même que pour le mode 640×200, puisque le fait que nous ayons une résolution horizontale deux fois moindre est contrebalancée par le fait que chaque pixel d'information est affiché sur deux pixels physiques de l'écran.

#### *Conversion en 4 niveaux de gris*

L'option à utiliser ici est quasiment la même que celle que vous avez peut-être utilisé pour l'étape de préparation de l'image, si vous aviez un fichier graphique de plus de 256 couleurs. Utilisez donc la fonction *Special Effects*, accessible par la touche [F9], puis sélectionnez l'option *Grey Scale*, puis *4 Levels* et enfin le format de fichier *PCX*.

#### **Inversion de l'image**

Que vous ayez une image monochrome ou en 4 niveaux de gris, il vous reste toujours à rendre l'image en "négatif". Bien que le HP100 permette d'inverser l'écran (avec la combinaison de touches [ON] [/]),

cette étape est conseillée, car vous éviterez ainsi une manipulation de touches. Le fichier résultant sera bien sûr en format *PCX*.

#### **Visualisation de l'image sur le HP100**

Si vous n'avez pas d'utilitaire de visualisation de fichier *PCX*, la méthode la plus simple est de redéfinir le *Topcard*. Comme pour le HP95, cela se fait dans l'application *SETUP*.

Malheureusement, le HP100 n'accepte pas que le *TopCard* soit en 4 niveaux de gris. Pour ces fichiers, il nous faut donc vraiment utiliser un utilitaire spécifique.

Donc, affaire à suivre...

Jacques Belin (123)

---

## **UNE COMMANDE A LA MODE**

Dans le directory *\BIN* du disque D du HP100 se trouve une commande *MONO8025.COM* qui n'est pas documentée dans la notice de la machine. Son rôle est de faire passer l'écran en mode 80×25 et de modifier la palette de niveaux de gris, afin de rendre l'écran plus lisible avec certains programmes.

Cependant, hormis cette commande restreinte à un seul mode, nous n'avons aucun moyen de contrôler la taille de l'écran autrement qu'en pressant les touches [Fnt] [ZOOM]. Cela aurait pourtant été intéressant de pouvoir configurer de façon logicielle (dans un fichier batch) le mode d'affichage avant et après l'exécution de ces programmes.

C'est pour cela que j'ai créé la commande *SCRN100.COM* que je vous présente aujourd'hui.

Sa syntaxe est la suivante :

```
SCRN100 ( 8 | 6 | 4 )
```

Ceci veut dire qu'elle admet l'un des paramètres 8, 6 ou 4 indiquant que l'on veut passer respectivement en mode 80, 64 ou 40 colonnes.

#### **Fonctionnement du programme**

Le principe de programme est simple, puisque il s'agit en fait d'utiliser la fonction standard du Bios de l'IBM PC permettant de sélectionner le mode vidéo courant. C'est la fonction 0 de l'interruption 10. Le mode désiré se place dans le registre AL

Normalement, sur un IBM PC, il est possible de connaître le mode vidéo courant en utilisant la fonction 0Fh de l'int 10h. Ou plus directement en lisant l'octet situé à l'adresse 40:49. Cependant, afin de garder une parfaite compatibilité des programmes, les développeurs du HP100 ont préféré indiquer le mode réel dans un octet situé à l'adresse 40:9F, et renvoyer toujours la valeur de mode "3" dans les fonctions standard.

Si on teste l'octet situé en 40:9F, en faisant varier le nombre de colonnes d'affichage, le HP100 rapporte le code 3 pour le mode 80 colonnes, 81h pour le 60 colonnes et 85h pour le 40 colonnes. Or, suite à quelques tests j'ai constaté qu'il était préférable d'utiliser les codes 80h au lieu de 81h et 84h au lieu de 85h. Ceci parce qu'il semble qu'en sortie de certains modes vidéos, le passage dans un autre mode n'est pas effectué...

En fait, le HP100 se révèle très "chatouilleux" dès qu'on manipule certains modes vidéo. Désagréments qui peuvent aller jusqu'au plantage "dur" du HP100. Par exemple, certains problèmes pouvant survenir en sortie du mode graphique compatible avec celui du HP95 si on utilise certains programmes tels que QSHELL95. La cause semble due au fait que le HP100 a peut être un bug sur la gestion du mode numéro 7 (texte 80x25, monochrome) qui est utilisé par le HP95 (le HP100 utilisant le mode numéro 3). Il en résulte que certains programmes fonctionnant parfaitement sur HP95 posent des problèmes sur le HP100. Je n'ai pas encore trouvé la raison exacte de ce dysfonctionnement, mais je vous tiendrais au courant dès que j'aurais plus de détails. En attendant, j'ai constaté que si on créait un fichier batch appelant le programme graphique puis la commande SCRNM100, il n'y avait plus de plantage... Je vous encourage donc à faire ainsi si vous avez des problèmes équivalents.

Pour terminer, si vous regardez bien le listing du programme, vous constaterez que la routine de test limitant l'utilisation de la commande aux seuls HP, autorise aussi l'utilisation sur HP95. Cela est intentionnel, car cela nous permet de l'inclure dans des fichiers batch de lancement de programmes pouvant être utilisés sur les deux machines, sans afficher de message d'erreur. J'ai bien sûr vérifié qu'il n'y avait aucune action de la commande sur le HP95.

### Listing du programme

```
DOSSEG
code segment para 'CODE'
    assume cs:code,ds:code
    ORG 100h
start: jmp main ; saut sur première inst.
```

```

; message d'erreur
usage DB "usage : scrn100 { 8 | 6 | 4 }"
DB 13,10,10,"$"
Not_100 DB "This program runs only on an HP100LX"
DB 13,10,"$"

main: mov ax,4dd4h ;vérification d'exécution
int 15h ; sur un palmtop HP
cmp bx,'HP'
jne main_2
cmp ch,1
je main_3 ;si hp100, on passe
main_2: mov dx,OFFSET Not_100
jmp err02

main_3: mov si,128 ; SI ^ sur longueur param
lodsb
cmp al,1 ; si pas de paramètres..
jbe error ; ... erreur

; SAUT DES ESPACES DU PARAMETRE
xor ch,ch ; CX = nbre caractères param
mov cl,al
main_4: lodsb ; lecture caractère param
cmp al," " ; si caract # espace..
jne main_5 ; passe à la suite de l'analyse
dec cx ; si encore des espaces..
jnz main_4 ; ..on boucle
jmp error ; ..sinon erreur

; ANALYSE SIMPLIFIEE PARAMETRE
main_5: cmp al,"8" ; si parametre = "8"
jne main_6 ; non, on passe
mov dl,03h ; sinon, dl = mode...
jmp main_8 ; .. et on va plus bas
main_6: cmp al,"6" ; si parametre = "6"
jne main_7 ; non, on passe
mov dl,80h ; sinon, dl = mode...
jmp main_8 ; .. et on va plus bas
main_7: cmp al,"4" ; si parametre # "4"..
jne error ; ..erreur
mov dl,84h ; sinon, dl = mode...

main_8: xor ah,ah ; AH=0 (fonction SET MODE)
mov al,dl ; AL = Mode vidéo
int 10h ; activation fonction vidéo
jmp exit ; ...et on sort.

error: mov dx,OFFSET usage
err02: mov ah,09h ; affichage message d'erreur
int 21h
exit: mov ah,4ch ; retour au DOS
int 21h
Code ends
END start ; That's All Folks !
```

Jacques Belin (123)



Donc à partir de maintenant pour tout assemblage de chaîne de codes procédez de manière suivante :

- 1- lancez le programme ASSCOD48 ou ASSCOD28
- 2- donnez le nombre d'octets (1/2 octet compris) puis validez avec ENTER.
- 3- tapez chaque ligne de codes, correspondant au numéro de ligne à 3 chiffres, sans les espaces et validez.
- 4- tapez la somme de contrôle et validez. S'il y a erreur la ligne de codes sera demandée à nouveau après émission d'un BEEP. L'appui sur EDIT fera apparaître la ligne des codes qui pourra être corrigée. Alors relancez avec CONT.
- 5- après l'affichage de "fin" stockez le programme assemblé dans la variable donnée en tête.
- 6- si tout s'est bien déroulé vous pouvez purger tmpcod qui contient la chaîne de codes.

### Programmes MS-DOS

Afin d'être utilisé par tous, ce programme est destiné à être écrit en GWBASIC. Il devrait cependant être facile de le convertir pour un autre programme (QBASIC, Turbo BASIC...).

#### Programme MAKEDOS.BAS

```
10 INPUT "Nom du fichier : ",NOM$ : INPUT "Nombre d'octets : ",N : N=N*2
20 OPEN NOM$ FOR OUTPUT AS #1 : CLOSE #1 : KILL NOM$
30 OPEN "bin.tmp" FOR OUTPUT AS #1 : S=0 : P$="-----"
40 NLINES=N\16 : LENLAST=(N MOD 16)+((N MOD 16)\5)
50 IF (N MOD 16)=0 THEN NLINES=NLINES-1 : LENLAST=LEN(P$)
60 FOR X=0 TO NLINES
70 IF X=NLINES THEN P$=LEFT$(P$, LENLAST)
80 C$=P$
90 X2$="00"+HEX$(X) : PRINT RIGHT$(X2$,3);";";
100 Y=CSRLIN : LOCATE Y,6 : PRINT C$; : LOCATE Y,6 : INPUT "",C$ : IF Y=24 THEN Y=Y-1
110 LOCATE Y,27 : PRINT " sm = ---" : LOCATE Y,33 : INPUT "",D$
120 M=S
130 FOR Z=1 TO LEN(C$)
140 IF MID$(C$,Z,1)<>" " THEN M=(M+((Z-(Z\5))*ASC(MID$(C$,Z,1)))) MOD 4096
150 NEXT Z
160 D2$="00"+HEX$(M) : D2$=RIGHT$(D2$,3)
170 IF D2$<>D$ THEN PRINT "Erreur de somme" : BEEP : GOTO 90
180 FOR Z=1 TO LEN(C$) STEP 2
190 IF MID$(C$,Z,1)=" " THEN Z=Z-1 : GOTO 230
200 CH=ASC(MID$(C$,Z,1))-48 : IF CH>9 THEN CH=CH-7
210 CL=ASC(MID$(C$,Z+1,1))-48 : IF CL>9 THEN CL=CL-7
220 PRINT#1,CHR$((16*CH)+CL);
230 NEXT Z
240 S=M
250 NEXT X
260 CLOSE #1 : NAME "bin.tmp" AS NOM$ : END
```

Mode d'emploi :

- 1- Lancer le programme : GWBASIC MAKEDOS.BAS
- 2- Entrer le nom du fichier destination.
- 3- Entrer la taille du fichier.
- 4- Entrer les listes de codes puis le checksum (en prenant soin d'entrer les codes hexadécimaux en majuscules). En cas d'erreur corriger la ligne, en prenant soin de placer le curseur après le dernier caractère avant de taper sur la touche d'entrée.
- 5- Une fois que toutes les lignes sont entrées, sortir du GWBASIC en exécutant la commande SYSTEM. Le nouveau programme est immédiatement disponible.

Note : La taille du fichier résultant peut être supérieure d'un octet à ce qui est affiché dans le listing. Cela n'est pas un problème.

SYMB28S (HP28)  
# 37E6h 40 octets

0123 4567 89AB CDEF sm  
000: 76C2 0703 70A4 020C C8C  
001: D8C0 3ACD 38C8 C07A C0F  
002: 8C07 0370 CD8C 03AC B24  
003: D335 FE33 6FE3 3ABB B54  
004: 19C2 11E6 1A10 9F20 87B

NSEM2 (HP48)  
# 4D43h 85 octets

0123 4567 89AB CDEF sm  
000: D9D2 0D29 512B F819 E0A  
001: FF30 D9D2 0AAB 368D D7B  
002: A16E 0CC0 8813 0038 9DF  
003: 261F 514F AAA2 0831 776  
004: 6CDB A247 9A22 6205 43E  
005: 1BF2 693C C0B7 E26C 3AA  
006: CD20 3200 08FC 29D0 170  
007: 8FA3 BD08 F15B D004 F98  
008: 8DC7 5308 0836 5304 B6C  
009: 07FE 30B9 F06F BD81 AFF  
00A: B213 0B21 30 611

CNFGPICK.LIB (HP48)  
# 8461h 1015.5 octets

0123 4567 89AB CDEF sm  
000: 04B2 0917 0011 34F6 C5B  
001: E666 9676 0205 9636 87E  
002: B656 2702 1212 1211 30F  
003: A669 F500 0B20 06A6 FEB  
004: 0050 000D 9D20 CCD2 E96  
005: 0382 008F B976 0D28 C6B  
006: 1AF0 A143 1338 1AF0 9CC  
007: 4174 1471 3317 41B7 625  
008: D735 1421 3016 E14A 2F0  
009: D280 861C 5143 8AC6 0E5  
00A: 0659 0130 1641 4213 BCB  
00B: 781A F0B8 1AF0 08F7 AA4  
00C: 3560 8FB9 7608 1AF1 8E5  
00D: B137 81AF 1AE6 81AF 8AA  
00E: 0A34 D004 0145 1746 466  
00F: 9AF8 1AF0 98AA B314 303  
010: 3133 1741 4713 3D58 F93  
011: 1B43 45F0 00CA 1301 C00  
012: 8016 2152 3938 6593 827  
013: 41F6 4808 1AF1 AE68 719  
014: 1AF0 91B7 D735 1421 3B1  
015: 3016 E14A D280 872D 1AA  
016: 134A 6600 81AF 088F 020

017: 9256 08D4 65D2 3034 CA0  
018: 8E81 B434 CE00 OCA1 AAB  
019: 3016 1152 3938 D093 738  
01A: 0151 625E E174 81AF 5DB  
01B: 19CE 6E3F D015 2313 2DA  
01C: 781A F0B8 1AF0 08F7 1B3  
01D: 3560 8FB9 7608 1AF1 FF4  
01E: B137 81AF 1AE6 81AF FB9  
01F: 0A34 D004 0145 6CAF E21  
020: E4A2 0560 0000 0000 841  
021: 0000 0000 0000 0000 1C1  
022: 0000 0000 0000 0000 B41  
023: 0000 0000 0000 0000 4C1  
024: 0000 0000 0000 0000 E41  
025: 0000 0000 0000 0000 7C1  
026: 0000 0000 00E4 A205 3A1  
027: 6000 0000 0000 0000 D27  
028: 0000 0000 0000 0000 6A7  
029: 0000 0000 0000 0000 027  
02A: 0000 0000 0000 0000 9A7  
02B: 0000 0000 0000 0000 327  
02C: 0000 0000 0000 0000 CA7  
02D: 0000 3C37 0907 7043 86A  
02E: 370B 2130 8E92 0F30 55E  
02F: 00C2 A201 0000 1000 F95  
030: 0B20 00C4 9626 2716 BBA  
031: 2797 0294 E602 2416 7BE  
032: 4602 05F6 2747 8A66 526  
033: 000D 9D20 ECE8 1120 2A4  
034: 40D9 D202 9E20 A668 04C  
035: 0029 E20A 6640 029E D84  
036: 20A6 6600 29E2 0A66 AA6  
037: 5008 5230 29E2 0A66 798  
038: 300B 2130 B213 08A6 43C  
039: 6100 D9D2 0ECE 8112 23E  
03A: 040D 9D20 29E2 0A66 FDC  
03B: 8002 9E20 A664 0029 C1A  
03C: E20A 6670 029E 20A6 956  
03D: 6500 8523 029E 20A6 63E  
03E: 6300 B213 0B21 308A 2D4  
03F: 6620 0D9D 20EC E819 1C0  
040: FF30 D9D2 0AEC 8129 05A  
041: E20A 6680 0CCD 204B EA6  
042: 0008 F146 6081 AF00 BB1  
043: 8FB9 7603 4007 008B 82E  
044: 2606 A807 A505 EBFF 84F  
045: 9C10 08DE FF80 FFFE A4B  
046: 0FFF 91FF F52F FF55 B11  
047: FFFF 5FFF 66FF F27F CB3  
048: FFAC EFF6 2FFF 78FF F08  
049: F2DC FF22 FFFC BEFF 26C  
04A: 0000 0071 3414 28A8 EAC  
04B: E113 6CA1 3081 AF10 C00  
04C: 1503 1341 646F DF63 A3C  
04D: 20E4 B12B 2130 B213 68D  
04E: 0CCD 20D3 0008 FB97 4DB  
04F: 6081 B434 D0BF FCA1 4DD  
050: 3016 1142 818F 8313 135

051: 68FE 7950 1583 8F34 E9D  
052: 150D 9D20 FA45 0198 C18  
053: 1660 0118 8130 3C37 845  
054: 0BE8 2652 3309 6126 428  
055: 64B3 6AEC 8175 6603 154  
056: 2230 4337 0954 50B2 D86  
057: 130C CD20 C600 08FB BFF  
058: 9760 1431 7414 7131 72A  
059: 1341 7414 3174 D881 3C1  
05A: 8F99 4231 6414 2341 F1A  
05B: 1920 8A60 2164 1523 A9A  
05C: 1513 1721 6481 8F92 77E  
05D: 4606 6DFD 2155 38F3 596  
05E: 4150 CCD2 0340 0081 16A  
05F: B434 C0AF FCA1 3216 FA0  
060: 1146 132D A340 4B20 C3D  
061: 8A22 1341 0A66 DA8D B0E  
062: 3205 08D4 65D2 E4A2 949  
063: 02B0 0000 0000 0000 303  
064: 0000 0000 0000 0000 C83  
065: 0000 0000 0000 00D2 74F  
066: 0000 0000 0000 0000 0CF  
067: 0000 0000 0000 0000 A4F  
068: 0000 00E4 0009 0C49 6C4  
069: 4241 4454 5143 4840 232  
06A: 0090 C494 2444 5445 E3E  
06B: 1434 8410 0B03 4841 9EF  
06C: 4E47 454E 40BC 4942 797  
06D: 4200 9400 0730 0052 236  
06E: 000E 4A20 2300 064C E82  
06F: FF2A CFFE FCFF 5EDF 224  
070: F22E FFD6 EFF5 FAFF 543  
071: A5BF FFCE FFFB A400 60C  
072: 0000 0000 0000 0000 F8C  
073: 0000 0000 0000 0000 90C  
074: 0000 0000 0000 0000 28C  
075: 0000 0000 0000 0000 C0C  
076: 0000 0000 0000 0000 58C  
077: 0000 0000 0000 0000 F0C  
078: 0000 0000 0000 0000 88C  
079: 0000 0000 0000 0000 20C  
07A: 0000 0000 0000 0000 B8C  
07B: 0000 0000 0000 0000 50C  
07C: 0000 0000 0000 0000 E8C  
07D: 0000 0000 0000 0000 80C  
07E: 0000 0000 0000 000 E8C

SCRN100.COM (MS-DOS)  
159 octets

0123 4567 89AB CDEF sm  
000: EB49 7573 6167 6520 BFB  
001: 3A20 2073 6372 6E31 863  
002: 3030 207B 2038 207C 500  
003: 2036 207C 2034 207D 19C  
004: 0D0A 0A24 5468 6973 E78

005: 2070 726F 6772 616D BEF	011: 750A B284 32E4 8AC2 53F	002: 170D 9D20 BB72 6FD5 B2D
006: 2072 756E 7320 6F6E A40	012: CD10 EB07 BA02 01B4 2CA	003: 502C 230A 1626 A162 7B0
007: 6C79 206F 6E20 616E 82D	013: 09CD 21B4 4CCD 21 A8E	004: 6C1C 1632 230C CD20 53A
008: 2048 5031 3030 4C58 456		005: 2500 08F1 4660 8FB9 374
009: 0D0A 24B8 D44D CD15 388		006: 760C CD81 4313 0174 F92
00A: 81FB 5048 7505 80FD 1B0	ASC→ (HP48)	007: 1431 3117 9319 315B C02
00B: 0174 05BA 2401 EB37 F6B	# A0F5h 91 octets	008: 19EA 8081 8686 1580 86F
00C: BE80 00AC 3C01 762C D30		009: 1711 60CD 54E8 D341 65E
00D: 32ED 8AC8 AC3C 2075 BC8	0123 4567 89AB CDEF sm	00A: 50B9 F06B 2130 B213 2E7
00E: 0549 75F8 EB1E 3C38 AF3		00B: 0B21 30 714
00F: 7504 B203 EB0E 3C36 93D	000: D9D2 0D29 51D9 F81D F14	
010: 7504 B280 EB06 3C34 6D3	001: 0040 D9D2 0902 36A2 BE2	

*Ouf, c'est terminé pour cette année !*



Le Journal JPC est le bulletin de liaison entre les membres de l'Association "PPC Paris", régie par la loi de 1901. Le Club est éditeur de JPC, et son siège social est au 56, rue Jean-Jacques Rousseau, 75001 Paris.

PPC Paris est le représentant Français de **HEX** (**H**andhelds **E**uropean **C**lubs **E**Xchange), la fédération des principaux clubs Européens d'utilisateurs de calculateurs HP.

La maquette de ce numéro a été préparée et réalisée par Jacques Belin et Asdin Aoufi.

Les dessins sont de Jean-Jacques Dhénin et Paul Courbis.

Les informations et programmes parus dans ce journal sont publiées "Tels quels" et ne peuvent en aucun cas engager la responsabilité de Hewlett-Packard ou de PPC Paris. Hewlett-Packard se réserve le droit de ne pas répondre aux questions concernant le sujet de certains articles.

Les programmes publiés peuvent être utilisés librement. Cependant, ils ne peuvent être vendus ou fournis dans un ensemble commercialisé, sous quelque forme que ce soit, sans l'accord écrit de l'auteur ou de PPC Paris.

Directeur de la publication : Jacques Belin  
Numéro ISSN : 0762 - 381X

Veuillez adresser toute correspondance à :  
PPC Paris, BP 604, 75028 Paris Cedex 01.

Imprimé par Paris Copie, 4 rue Linné, 75005 Paris