JPC

PPCPC

JPC AOÛT

1983
VOLUME-1
NUMERO-5

LE NUMERO 15FF

APPLICATIONS

ROBERT SCHWARTZ	2	RIDEAU ou "DOMAINE D'ALICE"
ROBERT SCHWARTZ	4	ASN pour "sportifs"
ROBERT SCHWARTZ	6	L'alphabet de travers
ROBERT SCHWARTZ	8	BLDSPEC et codage sur 7 octets (suite)
ROBERT SCHWARTZ	9	ECONOMIE d' OCTETS
ROBERT SCHWARTZ	10	LA PILE
ROBERT SCHWARTZ	12	Nombre d'OCTET et de REG dans un PRGM
ROBERT SCHWARTZ	13	BIN, HEX et DEC
DANIEL JACOB	16	Structure des 'GTO' et 'XEQ' numériques
ANDRE OISEL	19	AFFECTIONS SYNTHETIQUES
ROBERT SCHWARTZ	20	LE "BJ" : quoi, comment ?
ROBERT SCHWARTZ	23	RAMBARDE POUR BG
ROBERT SCHWARTZ	24	IMP "IL" ou pas "IL"

PROGRAMMES

ROBERT SCHWARTZ	2	C?, CX, CU (PPC ROM)
ROBERT SCHWARTZ	7	ABC
ROBERT SCHWARTZ	12	OP
DANIEL JACOB	15	PP
DANIEL JACOB	18	PRGM in PRGM
ANDRE OISEL	19	XA, SA
ROBERT SCHWARTZ	25	"CNR" = YTOAX
ROBERT SCHWARTZ	26	"CNX" = (ATOX suite)
ROBERT SCHWARTZ	27	HD (PPC ROM)
FREDERIC POUPON	28	COMP
YANN DOLHEN	30	APPHR
YANN DOLHEN	31	MORSE
JACQUES DURAND	33	BacHe

LA REVUE DES CODES BARRES

ROBERT SCHWARTZ	35	VIVENT LES CODES BARRES
ROBERT SCHWARTZ	37	MORSE
ROBERT SCHWARTZ	39	FONCTIONS DU WAND
ROBERT SCHWARTZ	40	COMP
ROBERT SCHWARTZ	42	XA, SA

LE PETIT THEATRE DES MICROCODES

ROBERT SCHWARTZ	22	VIVE LE PETIT THEATRE
PHILIPPE GUEZ	27	CHOC EN RETOUR (V1N4P29)
PHILIPPE GUEZ	36	TONES EN MICROCODE
PHILIPPE GUEZ	37	B+1 B-1

PILOU SANABRIA	1	UGH!	EPROMS ET CARTES MAG AU PPCPC	8
ROBERT SCHWARTZ	21 22	SOS	REUNION PPCPC A MARLY (génér.)	26

Le JPC est imprimé et édité par le PPC PARIS CHAPTER 56, rue J.J. Rousseau 75001 PARIS - FRANCE

Le PPC PARIS CHAPTER est une association régie par la loi de 1901 , enregistrée à la préfecture de PARIS le 2 décembre 1982 sous le numero 82/3240.

UGH !
VOUS, SUIVRE HOI SUR
TRACES DES CB
DU LIVRE J.P.C
DU GRAND SORCIER

```
7:29PM 06/23
01♦LBL "C?"
02 RCL c
03 STO [
04 "┣♦♦A"
05 X<> [
07
08 CF 07
09 CF 04
10 CF 07
11 FS?C 10
12 SF 07
13 FS?C 11
14 SF 09
15 FS?C 12
16 SF 10
17 FS?C 13
18 SF 11
19 FS?C 14
20 SF 13
21 FS?C 15
22 SF 14
23 FS?
24 SF
25 X<
26  E38
27 /
28 INT
29 DEC
30 RTN
31♦LBL "CX"
32 XEQ "C?"
33 -
34♦LBL "CU"
35 ABS
36 RDN
37 RCL c
38 STO [
39 "┣♦♦♦♦"
40 11
41 X<> [
42 X<> d
43 STO ]
44♦LBL 00
45 RDN
46 X<> L
47 INT
48 X=0
49 GTO 14
50 2
51 /
52 RCL [
53 X<>Y
54 FRC
55 X=0?
56 GTO 13
```

RIDeau ou "domaine d'Alice" ?

Vous avez dû vous apercevoir que je confondais les 2 expressions citées dans le titre et je vais vous expliquer pourquoi je me mélangeais les "crayons" (de la comprenoire).

Pour moi, regarder derrière le RIDeau c'était regarder là où on ne devrait pas regarder. Or, j'ai fini par m'apercevoir qu'en fait, derrière le RID n'était pas (encore) le "domaine d'Alice", ce dernier étant l'ensemble de la MEM interne, c'est-à-dire tous les REG auxquels nous ne pouvons pas accéder normalement.

Le RIDeau est la limite extrême de la partie occupée par les PRGM (éventuels) situés en MEM principale. Je ne suis pas sûr que "ceux qui ne savent pas" ont compris (?).

La MEM vive dont on peut disposer sur une 41CV comporte 319 REG que nous allons représenter, empilés les uns sur les autres. Le 1er REG, en Bas, a comme ADD DEC 193 et le dernier REG (en Haut), a comme ADD DEC 512 d'où : 512-193= 319 mais il se peut que je me sois trompé d'une unité (Bm +), en ce qui concerne ces ADD et si quelqu'un se dévoue pour nous donner des précisions, j'expliquerais comment j'ai trouvé ces VAL (ADD).....

En partant du Bas, les premiers REG seront éventuellement utilisés pour les ASN de Fonctions et (ou) de LBL ALPHA globaux éventuels (1 REG pour 2 ASN).

Ensuite (toujours en montant), les REG seront éventuellement utilisés pour le stockage des ALARMes du module HORloge.

PUIS : "Buffer" qui signifie, je crois (?) : TAMPon. Jean-Daniel DODIN, dans son livre "AF", indique que cet emplacement "est destiné à recevoir des données à titre transitoire, par exemple au cours d'un transfert sur la boucle HP-IL entre le Mass storage (je pense qu'il doit s'agir du LECTeur de "K7" ?) et l'imprimante". J'aimerais bien également, avoir des précisions à ce sujet.....

Au-dessus, se trouveraient les REG disponibles avant le .END. permanent situé à la fin du dernier PRGM éventuel.

La limite supérieure du (ou des) PRGM serait notre fameux RIDeau .

Au dessus du RIDeau nous trouvrions donc les REG de données dont le nbre est déterminé par le SIZE et le dernier REG de données (situé tout en Haut) serait donc à l'ADD 512 (ou 511 ?).

« Ouf !, il fallait l'écrire ! »

Donc, si j'ai bien compris (?), après un MEMORY LOST et avec SIZE ØØØ, le RIDeau est à l'ADD DEC 512 et avec SIZE Ø12 il se trouve à l'ADD 5ØØ, etc....

```
57♦LBL 01
58 FC?C IND Y
59 SF IND Y
60 FC? IND Y
61 CHS
62 X>0?
63 GTO 13
64 FC? IND Y
65 CHS
66 DSE Y
67 GTO 01
68♦LBL 13
69 DSE [
70 GTO 00
71♦LBL 14
72 X<> ]
73 X<> d
74 STO [
75 "←ABC"
76 X<> \
77 X<> c
78 RDN
79 CLA
80 RTN
81 END
```

La définition que j'ai donnée au début est donc "imprécise" et nous dirons que le RIDeau est la limite déterminée par le SIZE, entre les REG de données et les REG de PRGM. Il est entendu que cette définition ne s'appilque qu'aux conditions normales (c'est-à-dire, prévues par bellemère HP) d'utilisation car, si cela vous intéresse, nous verrons qu'il est possible de déplacer le RID autrement qu'en agissant sur le SIZE et je vous assure que cela donne quelques applications intéressantes (avec la PS, bien entendu et en utilisant les PRGM adéquats de la PPC-ROM).

J'indique que si des collègues veulent traiter le sujet, j'en serais ravi car je préfère lire les articles des autres .....

Je tire donc le RIDeau et vous dis: à bientôt, si vous le voulez-bien ! rS (P20)

```
                        CAT 1
LBL'C?
LBL'CX
LBL'CU
END          163 BYTES
.END.        05 BYTES
```

ADD = ADresse
VAL = VALeur

"C?" est de Clifford STERN
"CU" est de Bill WICKES et Harry BERTUCCELLI
"CX" est de Keith JARETT.

CODES DEC :

P04: 244 127 0 0 65
P39: 245 0 0 0 0

CODES DEC pour le PRGM "GREC"

Etant donné que les CODES des chaînes ALPHA doivent être introduits en base 10, je me suis donné la peine indispensable de convertir les CODES indiqués Page 22 de JPC N°1.

Ligne	
15	247 16 18 8 47 128 128 131
17	247 17 7 252 16 32 65 128
19	247 16 0 3 105 48 81 0
21	247 16 4 112 16 63 160 0
23	247 16 3 240 20 169 140 0
25	247 16 113 21 217 55 81 28
27	16 1 244 153 47 128 0 (247 en tête!)
29	247 16 0 0 15 144 0 0
31	247 17 13 9 97 13 33 97
33	247 17 1 1 23 112 32 0
35	247 17 252 65 2 2 7 144
37	247 16 4 23 196 4 71 128
39	247 17 135 13 217 55 96 227
41	247 16 24 23 224 87 161 2
43	247 17 240 81 18 35 128 0
45	247 16 18 84 153 44 65 0
47	247 16 24 11 248 48 64 129
49	247 16 12 8 31 224 64 131
51	247 16 1 132 137 87 64 130
53	247 16 7 252 16 16 127 193
55	247 16 226 36 72 142 34 98
57	247 16 5 252 8 16 16 127
59	247 17 130 196 104 49 172 96
61	247 16 1 4 186 216 128 0
63	247 17 3 5 128 192 238 96
65	247 16 0 0 97 52 83 0
67	247 16 4 242 8 16 95 0
69	247 16 48 133 31 244 72 12
71	247 16 0 56 207 194 67 128
73	247 16 32 165 95 245 74 8
75	247 16 226 36 4 16 34 56

ALPHA α
BETA β
TAU Γ
DELTA Δ
OMICRON σ
LAMBDA λ
MU μ
PHI Φ

THETA Θ
OMEGA Ω
DELTA δ
SIGMA Σ
(KHI χ)

ASN pour "sportifs"

Voici une série de fonctions qui vous permettront de faire "un tas de choses" mais qui n'interressera pas ceux qui n'aiment pas "découvrir".....

Pour obtenir ces ASN, je vous conseille d'utiliser le PRGM "MKX" auquel vous ajouterez au Pas Ø2, CF 27 (sage précaution que je vous conseille d'adopter dans tous vos PRGM). "MKX" est à mon avis, le plus rapide parmi les nombreuses versions que j'ai eu l'occasion d'essayer avec, pour un "certain", des difficultés qui ne devaient être dues qu'à ma "comprenoire difficilette" ou à mon manque d'attention.... Il vous suffira donc, après avoir introduit le CODE de la touche à ASN, de taper USER ⊡ et de suivre la procédure dont le listing se trouve ci-joint. FIX Ø et CF 29 vous permettront d'obtenir la même bande (si vous avez une IMP).

Le module XF est pratiquement indispensable. J'indique à ceux qui ne le possèdent pas, que j'ai sous les yeux une lettre de la FNAC "forum" qui précise que les modules XF, XM, QuadM & Temps sont vendus séparément (sic), 275F et, avec l'achat d'une IMP ordinaire, d'un lecteur de carte ou d'un lecteur optique : 165F. Je tiens à remercier le charmant collègue qui m'a envoyé cette lettre.

UTILISATION

Il est vivement conseillé de déconnecter l'IMP avant d'utiliser les F ASN car il en est qq unes qui réagissent différemment avec l'IMP mais vous ne risquez pas grand chose d'essayer..... Rappelez-vous que l'IMP est un périphérique qui comporte ses propres fonctions qui réagissent à des CODES qui sont également utilisés par notre "41".

La touche 12 (B) est ASN au BG (je crois que j'aurais dû écrire le contraire!). J'aurais préféré adopter le BG de Lionel ANCELET publié dans JPC N°Ø P4 et "repris" dans JPC N°1 P16 mais je ne connais pas les CODES DEC à utiliser pour l'ASN. J'en profite donc pour lancer un appel à "ceux qui savent" pour qu'ils nous indiquent ces CODES et la formule mathématique qui permet, à partir de l'XROM, d'obtenir ces CODES.

La touche 22 est occupée par une F qui permet la création de GTO délocalisés. Exemple : en mode RUN taper XEQ "A" et ne pas s'inquiéter de l'affichage NONEXISTENT, passer en mode PRGM, USER E (touche 22) et vous obtiendrez GTO "A" qui n'est pas un GTO

"dirigeant" vers un LBL ALPHA local mais un GTO à un LBL ALPHA global codé sur 5 octets. Nous remarquons que le A est placé entre des guillemets.

La F ASN à la touche 33 (L) permet de créer des LBL délocalisés. En mode RUN, faire XEQ "A" comme pour le GTO, passer en mode PRGM puis USER L et vous obtenez LBL "A" qui n'est pas un LBL local codé sur 2 octets mais un LBL ALPHA global (synthétique) codé sur 5 octets qui pourra être ASN à une touche, ce qui accélèrera la "recherche". Vous m'excuserez de ne pas m'étendre sur la question mais si cela vous intéresse, je pourrais développer le système de codage des LBL, GTO et XEQ (tout un programme!).

Il est également possible d'utiliser des CODES de CAR inaccessibles directement au clavier et je vais vous indiquer la procédure que j'ai découverte. Exemple: CLA 12 (en mode RUN) XTOA (USER ✱ USER) ASTO X mode PRGM USER L et vous avez un LBL "μ".

Une procédure semblable vous permet de créer des XEQ délocalisés avec la touche X (64).

XEQ IND X (oubli = excuses): affichage NONEXISTENT

La fonction mu qui est l'équivalent français du QLoader "US" est ASN à la touche 34 (M) et sa soeur, STO Q, est ASN à la même touche SHIFTée. L'utilisation de ces 2 dernières F associées à d'autres (RCL Q STO M etc...) fera l'objet d'un "papier" ultérieur.

La touche 43 (EEX) permet d'obtenir un E qui est un 1 plus rapide et si vous tapez - et EEX (51 & 43), vous obtenez -E qui donne -1. N'oubliez pas d'effacer les 2 Pas suivants et vous serez surpris (?) de constater que le E s'est placé derrière le - (moins). Si vous faites d'autres essais (ne vous en privez pas!), vous aurez bien d'autres surprises qui ne seront pas toujours faciles à expliquer....

La F XROM 48,28 vous permet également de placer le signe moins derrière tout chiffre ou nbre : mode PRGM 123 USER - (touche 51) USER SST ← original, n'est-ce-pas ?

La touche 82 vous donne, toujours en mode PRGM, une virgule (ou un point si CF 28) mais c'est plus simple de taper sur la touche d'à côté.... Regardez tout de même au Pas suivant (?).

La touche 63 (W) permet d'obtenir le fameux sauteur d'octets sur lequel j'ai déjà écrit quelque chose mais je pense qu'il y a encore des applications à découvrir de ce côté là.

Je vous laisse découvrir des applications aux fonctions non citées ci-dessus et j'attends avec impatience le plaisir de vous lire car j'aime aussi beaucoup lire.....

Si je connaissais les CODES DEC des F"__" & μ-- publiées dans l'OP N°6 P53 (Jean THIBERGE) et qui "complètent" eGOBEEP (ASN à -22=G), j'aurais complété ces ASN.

Bonnes découvertes, rS (P20)

```
⊢* $%*+-/<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ↑abcdeΣ
ST+ IND N RCL d LBL → GTO ⊢
X<>c STO a ASTO M rS
LBL 01 SF 23 GTO 45 FC?C 67 89
```

L'alphabet de travers

Le PRGM "ABC" ci-joint permet d'imprimer tous les CARactères disponibles directement au clavier, et ceci dans le sens de la longueur de la bande de papier. Il suffit d'introduire en ALPHA un maximum de 24 CAR puis de faire XEQ "ABC".($)

Vous pouvez y ajouter les CODES correspondant aux chiffres Ø à 9 (publiés précédemment) en leur attribuant respectivement les LBL 2Ø à 29.

Je vous avais indiqué qu'il était possible d'imprimer les CAR dans le sens de la hauteur et je me suis limité aux CAR dont j'avais besoin pour imprimer le nom des fonctions synthétiques "code-barrées" mais il serait possible de traiter les 128 CAR ASCII "HP-41". Compte tenu de sa longueur, ce PRGM n'est pas très rapide et nous pourrions l'accélérer un peu en plaçant les CAR les plus utilisés, en tête de PRGM.

Signalons que ce PRGM peut très bien être utilisé avec l'IMP 82143A (non "IL") et les exemples ci-joints ont été réalisés avec cette IMPrimante.

Vous ne trouverez pas ce PRGM en CB mais si vous le désirez, je pourrais les publier dans un prochain N° de JPC.

Les CODES des chaînes de CAR ont été obtenus au moyen du PRGM "BL" de la PPC-ROM et je vous parlerais peut-être un jour, du système de compilation (?), "de mise en forme" d'une chaîne ACSPEC correspondant aux points 7x7 permettant tous les tracés désirés. rS (P20)

P.S.: Ci-dessous, la partie modifiée du PRGM "ABC" qui introduit les CODES des chiffres Ø à 9 et qui accélère le déroulement du PRGM (ajout de la routine LBL ØØ). Bonnes écritures "de travers".

Le PRGM "ABC" complet nécessiterait 85 RANGs de CB ...... rS (P20)

($) Si vous introduisez la chaîne ALPHA par PRGM, vous pouvez introduire 24 CAR en utilisant APPEND (exemple: "ABCDEFGHIJKLMNO" "⊢PQRSTUVWX").

```
63♦LBL 19
64 "ØØAαØ♦♦"
65 GTO ØØ
66♦LBL 2Ø
67 "ØμαY4Ø\"
68 GTO ØØ
69♦LBL 21
7Ø "ØØx̄Γå"
71 GTO ØØ
72♦LBL 22
73 "Øå' "
74 GTO ØØ
75♦LBL 23
76 "Øμ↓αØ⊢"
77 GTO ØØ
78♦LBL 24
79 "ØØEμØ"
8Ø GTO ØØ
81♦LBL 25
82 "Øμå‡"
83 GTO ØØ
84♦LBL 26
85 "ØαØ
A⊢"
86 GTO ØØ
87♦LBL 27
88 "ØØ AαØ"
89 GTO ØØ
9Ø♦LBL 28
91 "ØμØ' "
92 GTO ØØ
93♦LBL 29
94 "ØØ‡' "
95♦LBL ØØ
96 RCL [
97 ACSPEC
98 ADV
99 RTN
1ØØ♦LBL 32
etc....
1Ø9 REG
757 OCTETS
```

r-S

```
01♦LBL "ABC"
02 CF 27
03 9
04 PSIZE
05 2,005
06 ABS
07♦LBL "R"
08 ASTO IND L
09 ASHF
10 ISG L
11 GTO "R"
12 2,1
13 STO 06
14 SF 21
15 SF 12
16♦LBL "S"
17 5 E-3
18 STO 07
19♦LBL "T"
20 CLA
21 ARCL IND 06
22 13
23 ATOX
24 ASTO IND 06
25 X=0?
26 STOP
27 X=Y?
28 GTO 13
29 28
30 -
31 XEQ IND X
32 ISG 07
33 GTO "T"
34 ISG 06
35 GTO "S"
36 STOP
37♦LBL 13
38 "Qα8♦"
39 GTO 00
40♦LBL 01
41 "8αÖ▒@"
42 GTO 00
43♦LBL 04
44 7
45 SKPCOL
46 ADV
47 RTN
48♦LBL 08
49 "8 Δ"
50 GTO 00
51♦LBL 09
52 "Q8AαΩ"
53 GTO 00
54♦LBL 14
55 "0♦B  ♦"
56 GTO 00
57♦LBL 15
58 "0♦@α♦"
59 GTO 00
60♦LBL 17
61 "8♦♦↓♦♦"
62 GTO 00
63♦LBL 19
64 "0♦8Aα8♦"
65 GTO 00
66♦LBL 32
67 "0@@@Aα8"
68 GTO 00
69♦LBL 33
70 "8÷↓♦♦"
71 GTO 00
72♦LBL 34
73 "8Δ x̄x̄x̄"
74 GTO 00
75♦LBL 35
76 "8 ♦ΔQ@"
77 GTO 00
78♦LBL 37
79 "QΓ‡0Q@"
80 GTO 00
81♦LBL 38
82 "8μÖ' "
83 GTO 00
84♦LBL 39
85 "8Δ8 ' "
86 GTO 00
87♦LBL 40
88 "8→α(P▒"
89 GTO 00
90♦LBL 41
91 "QΔÅ
@"
92 GTO 00
93♦LBL 42
94 "0αΔÅ
@"
95 GTO 00
96♦LBL 43
97 "Qw‡8 ' "
98 GTO 00
99♦LBL 44
100 "QΓμ▒'"
101 GTO 00
102♦LBL 45
103 "0@x̄α)"
104 GTO 00
105♦LBL 46
106 "088αΔ!"
107 GTO 00
108♦LBL 47
109 "Q8σQP"
110 GTO 00
111♦LBL 48
112 "QΔ8 @"
113 GTO 00
114♦LBL 49
115 "QΓw05q"
116 GTO 00
117♦LBL 50
118 "Q↓∠01a"
119 GTO 00
120♦LBL 51
121 "0qαö0Q@"
122 GTO 00
123♦LBL 52
124 "0αΔÖ' "
125 GTO 00
126♦LBL 53
127 "QqÅö0Q@"
128 GTO 00
129♦LBL 54
130 "QΓ♦Ö' "
131 GTO 00
132♦LBL 55
133 "0α↓@"
134 GTO 00
135♦LBL 56
136 "8 @x̄α⊦"
137 GTO 00
138♦LBL 57
139 "8μö0'"
140 GTO 00
141♦LBL 58
142 "8  DH"
143 GTO 00
144♦LBL 59
145 "8
146 GTO 00
147♦LBL 60
148 "Q8QA8QA"
149 GTO 00
150♦LBL 61
151 "8 @8QA"
152 GTO 00
153♦LBL 62
154 "Q8Aα8⊦"
155 GTO 00
156♦LBL 66
157 "8 @♦Δ"
158 GTO 00
159♦LBL 69
160 "0Å↓♦"
161 GTO 00
162♦LBL 70
163 "0γδ$x̄"
164 GTO 00
165♦LBL 71
166 "0q8 G♦♦"
167 GTO 00
168♦LBL 72
169 "06&K8 "
170 GTO 00
171♦LBL 73
172 "8μQG♦♦"
173 GTO 00
174♦LBL 98
175 "00A÷Q)"
176♦LBL 00
177 RCL I
178 ACSPEC
179 ADV
180 END
 91 REG
632 OCTETS
```

CODES DEC :

```
P38= 247 17 252 144 193
     4 16 0
P41= 247 16 4 23 241 31
     208 64
P49= 247 16 32 242 131
     130 158 8
P52= 247 17 135 16 65 4
     17 195
P55= 247 16 66 163 138
     132 0 0
P58= 247 16 64 135 194 4
     0 0
P61= 247 16 7 192 0 0 0
     0
P64= 247 16 16 65 4 16 0
     0
P67= 247 16 64 64 64 65
     4 16
P70= 247 16 1 240 7 192
     0 0
P73= 247 16 8 32 130 2 2
     2
P76= 247 16 32 130 8 17
     28 0
P79= 247 17 6 15 248 48
     81 28
P82= 247 16 254 12 23
     240 96 191
P85= 247 16 250 8 16 32
     96 190
P88= 247 16 125 20 40 80
     145 31
P91= 247 17 252 8 19 224
     64 255
P94= 247 16 4 8 19 224
     64 255
P97= 247 17 123 15 16 32
     96 190
P100=247 17 6 12 31 240
     96 193
P103=247 16 248 64 129 2
     4 62
P106=247 16 56 137 18 4
     8 124
P109=247 17 5 9 17 228
     80 193
P112=247 17 252 8 16 32
     64 129
P115=247 17 6 12 25 53
     113 193
P118=247 17 7 13 25 49
     97 193
P121=247 16 113 20 24 48
     81 28
P124=247 16 4 8 23 240
     96 191
P127=247 17 113 21 24 48
     81 28
P130=247 17 6 10 23 240
     96 191
P133=247 16 254 4 7 192
     64 254
P136=247 16 32 64 129 2
     4 127
P139=247 16 250 12 24 48
     96 193
P142=247 16 32 161 68 72
     160 193
P145=247 16 138 172 153
     48 96 193
P148=247 17 5 17 65 5 17
     65
P151=247 16 32 64 129 5
     17 65
P154=247 17 252 16 65 4
     16 127
P157=247 16 32 64 129 10
     142 8
P160=247 16 241 19 196 7
     128 0
P163=247 16 121 18 36
     198 129 2
P166=247 16 113 16 32 71
     0 0
P169=247 16 241 18 38 75
     16 32
P172=247 16 112 17 228
     71 0 0
P175=247 16 249 16 65 1
     17 62
```

BLDSPEC et codage sur 7 octets (suite)

J'avais lancé un appel concernant la formule mathématique permettant d'obtenir les 7 CODES correspondant à ceux que l'on obtient au moyen de la fonction "BL" de la PPC-ROM. En attendant la réponse, j'ai commencé des recherches sur ce problème et je vous indique ce que j'ai découvert.

Si nous introduisons Ø pour le premier octet, c.à d.: Ø XEQ "BL" , nous obtenons :

Ø avec Ø à 31 introduit/ deuxième octet
1 " 32 " 63
2 " 64 " 95
3 " 96 " 127
4 " 128 " 159
5 " 16Ø " 191
6 " 192 " 223
7 " 224 " 255

Remarquons que 7 est 111 en BIN et donc le maxi codable sur 3 BITS et que "l'incrément" de 1 est conditionné par une différence de 31 qui est 11111 en BIN et le maxi codable sur 5 BITS.

Le premier octet est :

16 avec Ø à 63
17 " 64 " 127
18 " 128 " 191
19 " 192 " 255

L'incrément est alors de 63 qui est 111111 en BIN et le maxi codable sur 6 BITS.

Un professionnel de l'informatique découvrira tout de suite (?) le système. Hélas, je ne suis qu'un Amateur et vous comprendrez que je n'ai pas pu pousser plus loin mes "recherches".

<< Ohé les collègues chevronnés,
ne me laisser pas mourir idiot ! >>
MERCI de votre attention, rS (P20)

S.O.S. (?)

Est-il possible d'obtenir des LBL numériques dont la VAL est supérieure à 99 tels que LBL 154, par exemple ? Associés à des GTO compilés, ces LBL pourraient trouver qq applications ..... Mais je m'aperçois que la réponse est contenue dans ma question mais je laisse le soin aux spécialistes de développer la question....

MERCI DE VOTRE ATTENTION

Je signale à tous les lecteurs du JPC que des eproms et des cartes magnétique sont à vendre au PPC P C et qu'aprés les vacances, j'instaurerai un service de photocopie

la vente des cartes magnétiques est de 300FF les 100 cartes (je signale à tous ceux qui sont interessés que l'incription à PPCTN - Australie - est payable par cartes magnétiques)

La vente des éproms est de:

			pour les membres	pour les non membres
vierges	4K	(1 x 2716 ; 1 x 2716) =	100FF les 2	120FF les 2
	8K	(1 x 2716 ; 2 x 2732) =	150FF les 3	190FF les 3
copiées	4K		150FF les 2	200FF les 2
	8K		200FF les 3	280FF les 3

copie d'eprom (fournie) sur eprom déjà existante: 60FF (membres)

autres possibilités sur devis

PHILIPPE

ECONOMIE d' OCTETS

Les PRGM que j'ai à réaliser pour une utilisation professionnelle, sont souvent très longs, à la limite des 319 REG. Je suis donc, par la force des choses, un "radin" de l'OCTET.

Je vais donc vous indiquer quelques "trucs".

Le premier et, à mon avis, le plus beau, bien que, hélas, ce n'est pas moi qui l'ait trouvé, est ABS qui remplace avantageusement STO L en gagnant 1 octet. Merci Jacques VAUCELLE, qui est un chercheur de génie, doublé d'un programmeur très avisé. J'ai eu le chance de le rencontrer lors de son dernier passage dans la région parisienne; quel bonhomme ! Il mérite le déplacement. Peut-être ne va-t'il pas apprécier la publicité que je lui fait ; je vous demande donc de lui téléphoner avant tout comtact "physique" ou de lui écrire (il préfère le phone, après 20h...)
La Chasserie 35133 ROMAGNé (99) 98 83 69
Ceux qui adhère à PPC-T (je suis T178) ont ses coordonnées.

, x (virgule multiplier) est plus élégant que ∅ STO L mais maintenant, avec l'astuce de JV, , INT est encore plus beau. Cela nous rappelle que ABS & INT sont des opérations.....

CLA ARCL L ARCL X ACA ADV est + économe que :

CLA ARCL L ACA ACX ADV | NON!, là je me suis planté....

CLA ARCL L ARCL X ACA PRBUF STOP

CLA ARCL L ARCL X PROMPT (gain= 4 octets) (?)

Je n'ai voulu que mettre le train sur les rails. A vos machines à écrires et prenez le train en marche, comme vous le voyez, je n'ai pas mis toute la vapeur.....

R.S.

La PILE

A la demande de qq collègues, je vais essayer d'expliquer le "fonctionnement" de cette mystérieuse PILE. Compte tenu de l'importance du sujet, ce "papier" ne sera qu'une introduction car il est évident que je ne peux pas envahir les pages de JPC (?) mais si vous le désirez, je continuerais dans les N°s suivants.

Les explications du manuel de notre "41" sont pratiquement "imbuvables" et je dois avouer que j'ai eu la grande chance de posséder la première HP25 dont le Manuel était une petite merveille dans le genre, ce qui m'a permis de saisir facilement les subtilités de la PILE.

J'en profite pour indiquer que cette HP25 ainsi que son manuel m'ont été "empruntés" définitivement (?) et que je suis acheteur de ce type de machine qui m'a laissé de très bons souvenirs et vous comprendrez que j'y étais sentimentalement, très attaché..... Je suis donc prêt à payer un prix bien supérieur à la valeur vénale. Celui qui pourra m'aider à trouver cet oiseau rare aura droit à toute ma bien sincère reconnaissance.

La PILE est donc constituée de 4 REG (X,Y,Z&T) que nous allons représenter, emPILés les uns sur les autres. Le REG LAST X (L) ne semble pas faire partie de la PILE puisque la Fonction CLST n'efface pas son contenu. Le REG L est cependant étroitement associé à la PILE....

1°) Mode RUN (CF 28 SF 29 FIX Ø)

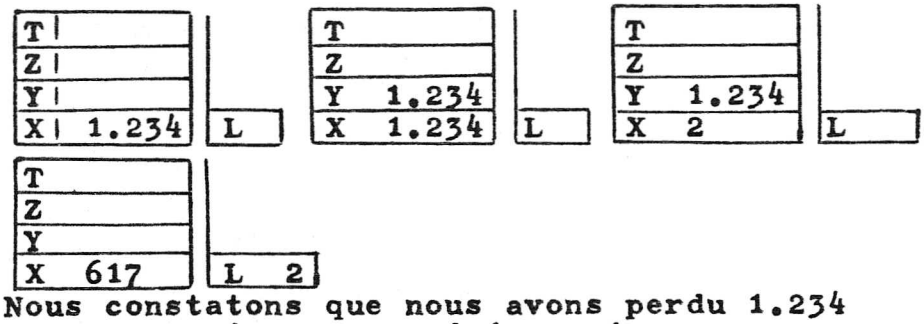
Si nous tapons un ou plusieurs chiffres (1Ø au maximum) constituant un nombre (exemple : 1234 donnant à l'affichage: 1.234_), notre Diva a besoin de savoir quand ce nbre est "complet" (terminé) et la solution proposée par HP est de taper ENTER↑.

Sachons que cette solution introduit bien la VAL en X mais également en Y et nous verrons plus loin comment faire "autrement".

Nous voulons diviser par 2 le nbre 1234 introduit précédemment par :

1234 ——— 1.234_
ENTER↑ ——— 1.234,
tapons 2 ——— 2
tapons / (÷) ——— 617,

et examinons ce qui s'est passé :



Nous constatons que nous avons perdu 1.234 mais que l'opérande 2 a été placé en L.

Si nous voulons conserver 1.234, nous allons

procéder comme suit : 1234 ENTER↑ ENTER↑ 2 /
ce qui nous donnera :

T	
Z	
Y	1234
X	617

L 2

Pour "saturer" la PILE, c'est-à-dire, pour introduire une même VAL dans ses 4 REG, il suffit d'appuyer 3 fois sur la touche ENTER↑ après avoir introduit la VAL désirée. Indiquons au passage que je pense que ce procédé est plus simple que CLST pour placer ø dans chaque REG de la PILE et si nous désirons, en plus, placer ø dans L, il nous suffira de taper x .....(OK ?).

A chaque fois que nous tapons ENTER↑ , la PILE "monte" d'un cran vers le Haut en "dupliquant" la VAL introduite, dans le REG supérieur mais, lorsque le sommet de la PILE est atteint (REG T), (donc après 3 fois ENTER↑), la VAL qui était en T est PERDUE.

Nous allons voir qu'il est possible de décaler la PILE vers le H sans perdre aucune VAL :

ø ENTER↑ ENTER↑ ENTER↑ x ——————— ø,
1 ——————————————————— 1_
ALPHA ALPHA (taper 2 fois ALPHA)___ 1,
ALPHA SHIFT CLA (facultatif)_____
ALPHA 2 ALPHA ALPHA ——————————— 2,
3 ALPHA ALPHA ——————————————— 3,
4 ALPHA ALPHA ——————————————— 4,

Nous avons maintenant :

T	1
Z	2
Y	3
X	4

L ø

Si nous avions appuyé sur ENTER↑ , nous aurions eu :

T	2
Z	3
Y	4
X	4

L ø

et le 1 aurait été PERDU (1 ENTER↑ 2 ENTER↑ 3 ENTER↑ 4 ENTER↑). Signalons cependant que si nous n'avions pas tapé le dernier ENTER↑ , le 1 aurait été sauvegardé.

Tapons: 4 ENTER↑ 3 ENTER↑ 2 ENTER↑ 1 XEQ ALPHA R SHIFT ENTER↑ ALPHA et nous aurons :

T	3
Z	2
Y	1
X	4

PILE				LAST X
X	Y	Z	T	L
1	2	3	4	ø
4	1	2	3	ø

La représentation "à plat" des REG me semble plus claire et nous verrons plus tard qu'elle est bien pratique.

La Fonction R↑ décale donc les REG de la PILE d'un cran vers le H (ou vers la Droite) et la F RDN (touche R↓) fait l'inverse. Ces 2 F permettent donc de visualiser le contenu de la PILE sans

perdre aucune VAL et il suffit de taper 4 fois sur la touche R↓ pour que toutes les VAL aient retrouvées leurs places initiales.

Il ne faut pas confondre la F RDN (R↑) avec ENTER↑ car cette dernière F fait "monter" la PILE d'un cran mais fait perdre la VAL qui se trouve en T. Sachons que toutes les F qui font "monter" la PILE en font autant.

Tapons: + ——————————— 5,

X	Y	Z	T	L
4	1	2	3	Ø
5	2	3	3	1

La F d'opération "+" a effectué l'addition des VAL de X et Y et a placé le résultat en X. L'opérande "1" a été placé en L et la PILE a été décalée d'un cran vers le Bas (vers la Gauche). Remarquons que la VAL qui se trouvait en T (3) est conservée.

Ceux qui disposent d'une IMP pourront utiliser la F PRSTK qui est bien gourmande en papier et qui, (on ne comprend pas pourquoi) n'imprime pas le contenu de L. Remarquons que si nous voulons visualiser le contenu de L (LAST X), nous décalons la PILE d'un cran vers le H et nous perdons alors le contenu de T. C'est bien ennuyeux et on aimerait bien que PRSTK soit complet.... Avis aux Amateurs de MC !

Je vous laisse donc manipuler les REG de la PILE en les visualisant au moyen de R↓ et vous observerez ainsi ce que font les F X<>Y X<>Z X<>T X<>L et X<>nn (nn étant le N° d'un REG compris entre ØØ et 99). Il est peut-être utile que je vous indique comment se font ces F :
XEQ ALPHA X SHIFT COS SHIFT TAN ALPHA . Z
XEQ ALPHA X SHIFT COS SHIFT TAN ALPHA . L
XEQ ALPHA X SHIFT COS SHIFT TAN ALPHA Ø Ø

Nous verrons la prochaine fois comment il est facile de faire des opérations en chaînes sans utiliser de parenthèses et je vous assure que c'est bien agréable. A bientôt donc, rS (P20)

```
01♦LBL "OP"
02 SAVEP
03 FLSIZE
04 RCLPT
05 PURFL
06 FIX 0
07 CF 29
08 CLA
09 ARCL Y
10 "⊢ REG"
11 AVIEW
12 PSE
13 CLA
14 ARCL X
15 "⊢ OCTETS"
16 PROMPT
17 END
```

OP

XEQ "OP"

```
7 REG
45 OCTETS
PRBC
```

XEQ "OP"

```
16 REG
105 OCTETS
```

Nombre d'OCTETS et de REG d'un PRGM

Le petit PRGM "OP" (Octets Prgm) ci-joint vous permet d'obtenir le nbre de REG et d'OCTETS occupés par un PRGM dont le nom (LBL ALPHA global) doit être placé en ALPHA.

Il nécessite le XF et éventuellement 1 ou 2 XM car avec le XF seul, un PRGM de 125 (127-2) REG au maximum peut être "mesuré".

Les fonctions FLSIZE & RCLPT permettent en effet, de rappeler en X le nbre de REG et d'OCTETS d'un PRGM placé en XM. Il suffisait donc de placer le PRGM en XM puis de l'effacer avec PURFL. C'est ce que fait le PRGM "OP".

Ce PRGM peut être utilisé pour déterminer le nbre de RANGs qui seront IMPrimés avec le PRGM "P +P P" dans lequel je n'ai pas pris la peine d'incorporer "OP" (vous pouvez le faire, si vous le désirez mais attention !).

Le nbre de RANGs est 0/9, 0 étant le nbre d'octets et le résultat doit être RND à l'entier supérieur et ajouter 1 par mesure de précaution... (RND = arrondi = excuses!). rS (P20)

BIN, HEX et DEC

Il n'est pas possible d'essayer d'aller "AU FOND de la 41C" sans connaître les bases de numération.

Dans un ORDI, les informations sont stockées de façon BINaire et sont donc constituées d'une multitude de "cases", d'éléments qui peuvent prendre 2 états : 1 et Ø. Ces éléments sont dits bistables et ont imposé le système de numération BINaire. Nous pouvons avancer que si nous maîtrisions le "tristable", nous serions amenés à partir (à prendre comme 1ère base) de la base 3 et que si la technologie nous permettait, d'une façon simple et fiable, d'obtenir des états décastable, nous n'aurions pas à utiliser d'autres bases que celle que nous avons l'habitude de manier. Je ne pense pas que les logiciels des micro-processeurs seraient plus simples pour autant mais je ne veux pas m'étendre sur la question.

Le tableau ci-après me semble être à la base (?) des systèmes de numérations utilisés "dans" un ordinateur.

	DEC	HEX
$2^{Ø}$	= 1	= 1
2^{1}	= 2	= 2
2^{2}	= 4	= 4
2^{3}	= 8	= 8
2^{4}	= 16	= 1Ø
2^{5}	= 32	= 2Ø
2^{6}	= 64	= 4Ø
2^{7}	= 128	= 8Ø
2^{8}	= 256	= 1ØØ
2^{9}	= 512	= 2ØØ
$2^{1Ø}$	= 1 Ø24	= 4ØØ
2^{11}	= 2 Ø48	= 8ØØ
2^{12}	= 4 Ø96	= 1 ØØØ

L'examen de ce tableau nous permet de découvrir les principales relations qui existent entre les 3 bases que nous utilisons.

Signalons que 2↑1Ø = 1 Ø24 est le fameux k (ka) qui est utilisé par les informaticiens pour indiquer le kilo et que les 24 unités supplémentaires sont négligées. Un ORDI dont la RAM est annoncée "1k", aura donc 1 Ø24 Octets et la "pesée" ($) sera, dans ce cas, fort appréciable.

En base 1Ø, la décomposition d'un nbre peut se faire de la façon suivante :

si nous prenons comme exemple le nbre 2832 et que nous lisons ce nbre de D à G, c'est-à-dire dans le sens : unité, dizaine, centaine, etc..., nous obtenons :

$2\ 8\ 3\ 2 = 2.1Ø^{Ø} + 3.1Ø^{1} + 8.1Ø^{2} + 2.1Ø^{3}$

Pour une VAL NEGative, nous aurons :

$\emptyset, 4\ 3\ 7 = 4.1\emptyset^{-1} + 3.1\emptyset^{-2} + 7.1\emptyset^{-3}$

Vous m'excuserez de ne pas faire la "démonstration" mais un examen attentif des exemples précités vous permettra de découvrir les relations et vous aurez peut-être un jour, l'occasion d'utiliser ces égalités.

Le procédé est le même pour toutes les bases.

En BIN :

$\emptyset 1 \emptyset 1 = 1.2^{\emptyset} + \emptyset.2^{1} + 1.2^{2} + \emptyset.2^{3}$

où nous constatons que les zéros donnent un résultat nul mais ne doivent pas pour autant, être négligés car ils ont une influence sur l'INCrément de la puissance de 2.

En HEX (base 16), la relation précitée est également vérifiée mais il faut tenir compte qu'après 9 nous avons A, B, C, D, E & F puis 1Ø, 11, 12, etc...

$D\emptyset 5 = 5.16^{\emptyset} + \emptyset.16^{1} + D.16^{2}$ (=3 333)

L'égalité étant calculée en DEC, nous prenons la valeur DEC de D qui est 13, ce qui donne :

DØ5 = 5 + (13.16↑2) = 5 + (13.256) = 3 333(OK ?)

Les VAL DEC de A à F ne nous facilitent pas les convertions HEX/DEC mais nous allons voir qu'il y a un moyen de s'en sortir....

Pour n'utiliser que des Maths "accessibles" (?), nous allons "passer" par la convertion en BIN et nous allons travailler sur un DIGit à la fois.

DØ5 sera décomposé en 3 DIG pris, cette fois-ci, de G à D : D = 13

```
           13 | 2
MOD  = ---->1 |----
              | 6 | 2
d°   = ------->Ø  |----
                  | 3 | 2
d°   = ----------->1  |----
                      | 1 | 2
d°   = --------------->1  |----
                          | Ø
```

Remarquons que puisque nous travaillons sur un DIG = 4 BITs, nous devrons toujours faire 4 divisions et que ce sont les MOD 2 qui seront pris comme résultat. Mais, me direz-vous, " DEC 13 ne nous donne pas BIN 1Ø11 !". Pour ne pas rentrer dans une démonstration "hazardeuse", nous allons nous contenter d'admettre que, puisque nous avons décidé de décomposer de G à D contrairement à ce que nous avions convenu au départ, il nous faudra enregistrer les résultats (MOD) de Bas en Haut, ce qui nous donne bien :

DEC 13 = BIN 11Ø1

Nous allons nous épargner de faire les mm opérations sur Ø et nous placerons simplement quatre zéros (ØØØØ).

```
            5 | 2
MOD  = ---->1 |----
              | 2 | 2
d°   = ------->Ø  |----
                  | 1 | 2
d°   = ----------->1  |----
                      | Ø | 2
d°   = --------------->Ø  |----
                          | Ø
```

d'où DEC 5 = Ø1Ø1 et, l'association des 3 DIG nous donnera :

BIN	11Ø1	ØØØØ	Ø1Ø1
DEC	13	ØØ	Ø5
HEX	D	Ø	5
DEC		3333	

Nous verrons ultérieurement comment effectuer la convertion DEC 13 ØØ Ø5 = DEC 3333 rS (P20)

CHOC EN RETOUR

Suite aux art. parus dans le JPC V1N1 sur "la technique du Compiled-GTO" et "(dé)privatisation",je vous propose une version plus courte du PRGM 'PP' (208 au lieu de 396). De plus,il ne nécessite plus l'utilisation du PRGM 'CODE' de Wickes.

Je redonne le mode d'emploi pour éviter toutes ambiguïtés:

1°. Chargez votre PRGM en mémoire
2°. Chargez le PRGM 'PP' (RSUB ou GETSUB)
3°. GTO "PP" puis RCL b
4°. GTO "(dernier label alpha du PRGM)" puis RCL b
    Si le PRGM ne contient pas de label alpha,levez le flag Ø1 (SF Ø1), sinon baissez-le (CF Ø1).Dans ce cas le RCL b s'effectuera à la 1ère ligne.
5°. Si vous voulez privatiser votre PRGM,levez le flag ØØ (SF ØØ),sinon baissez-le (CF ØØ)
6°. XEQ "PP"

Au sujet des GTO compilés, je comprends à tous votre impatience. Mais je voudrais préciser une chose: si on publie un (ou des) art. à ce sujet, combien pourront-ils suivre? JE crois qu'il serait plus sage d'attendre que les débutants en P.S soient familiarisés avec la stucture de la mémoire et des fonctions (LBL?,END,GTO,..),et surtout de possèder les PRGM utilitaires (CODE,DECODE,REG,..). LA technique du Compiled-GTO n'est pas une chose bien compliquée;il suffit,pour la maitriser,de bien comprendre ce que l'on fait.

Daniel JACOB (11)

```
01♦LBL "PP"
02 FIX 4
03 XEQ 14
04 X<>Y
05 STO a
06 XEQ 14
07 -
08 3
09 FS?C 01
10 CLX
11 -
12 7
13 /
14 INT
15 LASTX
16 FRC
17 7
18 *
19 ARCL X
20 RDN
21 256
22 X<=Y?
23 CLX
24 RCL I
25 X<> d
26 SF 00
27 SF 01
28 FS? 13
29 SF 04
30 FS? 14
31 SF 05
32 FS? 15
33 SF 06
34 X<>Y
35 X=0?
36 SF 07
37 X<>Y
38 X<> d
39 STO I
40 -6
41 AROT
42 ASHF
43 RT
44 256
45 MOD
46 XTOA
47 64
48 9
49 FS? 00
50 +
51 XTOA
52 RCL I
53 X<> a
54 STO I
55 "├♦♦♦♦"
56 RCL I
57 X<> d
58 FS? 15
59 SF 03
60 FS? 16
61 SF 04
62 FS? 17
63 SF 05
64 FS? 18
65 SF 06
66 FS? 19
67 SF 07
68 FS? 20
69 SF 08
70 FS? 21
71 SF 09
72 FS? 22
73 SF 10
74 FS? 23
75 SF 11
76 X<> d
77 STO I
78 "├♦♦♦"
79 RCL \
80 X<> c
81 RCL a
82 STO 00
83 X<>Y
84 STO c
85 CLA
86 PCLPS
87♦LBL 14
88 STO I
89 "├♦♦♦♦♦"
90 RDN
91 RCL I
92 X<> d
93 256
94 FC? 07
95 CLX
96 SF 07
97 X<>Y
98 X<> d
99 STO I
100 RDN
101 ATOX
102 16
103 /
104 INT
105 X<>Y
106 ATOX
107 +
108 7
109 *
110 +
111 CLA
112 END

LBL"PP
END          208 BYTES
```

Structure des 'GTO' et des 'XEQ' numériques

1°. **'GTO' numériques codés sur deux octets:**

Leur structure est de la forme: Ba lm

où: le 1er nybble 'B' est constant(en binaire 1011)

le 2ème nybble 'a' code le label (de 1 à F)

les 3 et 4ème nybbles 'lm' codent la distance et la direction entre le 'GTO' et le label. Décomposés en binaire,on a:

dooo rrrr
1 m

le bit 'd' indique la direction:

- 0 pour la direction vers le bas
- 1 pour la direction vers le haut

les 3 bits 'o' codent le nb d'octets

les 4 bits 'r' codent le nb de registres

soit au maximum: 15(1111)x7 + 7(111)= 112 octets

Exemple 1:

Ø1 LBL Ø1	Ø2	1
Ø2 ENTER	83	1
Ø3 ENTER	83	1
Ø4 ENTER	83	1
Ø5 "ABC"	F3 41 42 43	4
Ø6 GTO Ø1	B2 B1	2
		10 octets

direction vers le haut ← 1011 0001
B 1

d'où: 3 + 1x7 = 10 octets

Exemple 2:

Ø1 GTO Ø4	B5 01	
Ø2 ENTER	83	1
Ø3 "ABCDE"	F5 41 42 43 44 45	6
Ø4 LBL Ø4	Ø5	7 octets

direction vers le bas ← 0000 0001
0 1

d'où: 0 + 1x7 = 7 octets

2°. 'GTO' et 'XEQ' numériques codés sur trois octets:

Leur structure est de la forme: al mn bc

où: le 1er nybble indique le type d'instructions:

'D'(en binaire 1101) pour les 'GTO'

'E'(en binaire 1110) pour les 'XEQ'

les 2,3 et 4ème nybbles codent la distance entre le 'GTO'(ou le 'XEQ') et le label. Décomposés en binaire,on a:

```
ooor  rrrr  rrrr
 l     m     n
```

les 3 bits 'o' servant à coder le nb d'octets

les 9 bits 'r' servant à coder le nb de registres

soit au maximum: 2^9= 512 registres

les nybbles 'b' et 'c' codent la direction et le label.

Décomposés en binaire,on a:

```
dxxx  xxxx
 b     c
```

Comme pour les 'GTO' num. à 2 octets, le bit 'd' code la direction.

les 7 bits 'x' codent le label numérique (de 0 à 7F en Hexa)

Exemple 1:

```
Ø1 LBL A          CF 66              2
Ø2 ENTER          83                 1
Ø3 "ABC"          F3 41 42 43        4
Ø4 "DEF"          F3 44 45 46        4
Ø5 GTO A          DA 01 E6           1
                  1                  12 octets

          1010  0000  0001     1110  0110  → 102 (alias 'A')
           A     0     1        E     6
                                  ↳ direction vers le haut
```

d'où: 5 + 1x7= 12 octets

Exemple 2:

```
Ø1 XEQ Ø1         EØ Ø2 Ø1           2
Ø2 "ABCDEFGHIJ"   FA 41 42.....49 4A 11
Ø3 ENTER          83                 1
Ø4 LBL Ø1         02
                                     14 octets

          0000  0000  0010     0000  0001  → 1
           0     0     1        0     1
                                  ↳ direction vers le bas
```

d'où: 0 + 2x7= 14 octets

PRGM in PRGM

La P.S permet bien des choses (pas tout hélas),et souvent auquelles on n'y avait jamais pensé. Et c'est en décortiquant le PRGM 'TIX' de Lionel que je découvris la possibilité de coder un PRGM à l'intérieur d'un autre PRGM,éxécutable bien entendu. Il suffit de programmer les chaines adéquats et de s'adresser (dans la mémoire) au bon endroit. Sans plus tarder,un exemple/:

```
01◆LBL "P"
02 "◆QÜ"
03 RCL [
04 STO 03
05 "ØÅsα"
06 RCL [
07 STO 02
08 "s                    u
F"
09 RCL [
10 STO 01
11 "INΣα!"
12 RCL [
13 STO 00
14 CF 21
15 "ₓ"
16 RCL [
17 X<> b
18 END
```

```
CODES DU PRGM 'P'
(PEUVENT ETRE CHARGE
AVEC LE PRGM 'LOAD')

LIGNE 02=244,156,0,17,26
LIGNE 05=247,16,19,152,
        115,159,113,150
LIGNE 08=247,115,177,145
        117,134,243,70
LIGNE 11=247,73,78,126,
        156,4,206,124
LIGNE 15=242,1,240
```

ATTENTION: exécutable impérativement en SIZE Ø2Ø
Après (dé)assemblage,on peut extraire du PRGM 'P' le PRGM suivant:

```
01 FIX 0
02 1.03
03 VIEW X
04 TONE Z
05 ISG X
06 GTO 00
07 RDN
08 BEEP
09 "FIN"
10 AVIEW
11 FIX 4
12 X<> b
```

PRGM correspondant aux codes des lignes 2,5,8 et 11 du PRGM'P'. La ligne 15 comporte les codes de l'adresse à laquelle on va se pointer pour exécuter le PRGM ci-contre,en autre le 1er octet du registre 1FØ (alias la mémoire Ø4 en SIZE Ø2Ø);les octets étant chargés dans les registres 1EF,1EE,..(alias les mémoires Ø3,Ø2.. en SIZE Ø2Ø).On peut d'ailleurs calculer le 2ème code de l'adresse(le 1erétant toujours égale à 1 ou à Ø suivant la position dans la mémoire) avec la formule(?) suivante: 2ème code Ad.= 256-SIZE+NRU
où: NRU est le nombre de mémoire utilisées pour le stockage des octets uniquement.
Encore une précision:le stockage des octets se fait toujours de haut en bas,et n'oubliez jamais le X()b pour revenir en mémoire prog.

A bientôt: Daniel JACOB(11)

AFFECTATIONS SYNTHETIQUES

Différents programmes, tous nommés "KA (WICKES, LIND, DODIN), permettent l'affectation de fonctions synthétiques à des touches désignées.

Tous souffrent des deux handicaps suivants :
a) Les fonctions doivent être affectées par paires,
b) Les nouvelles affectations occupent toujours le bas de la table.

J'ai cherché une solution plus proche des procédures légales (ASN / PASN).
Ce programme nécessite un module XFUNCTION .

L'idée générale est de faire une affectation automatique, légale, fictive et temporaire, grace à la fonction PASN , et de la remplacer ensuite dans la table par le triplet définitif (Préfixe, Suffixe, Code touche) décrivant la fonction synthétique spécifiée.

Ceci permet : a) De n'affecter qu'une fonction à la fois,
b) De combler les vides éventuels dans le table,
c) Sans devoir manipuler les registres ⊢ et e.

La procédure est la suivante :

[Prefixe] ENTER↑ [Suffixe] ENTER↑ [Code Tche] XEQ"SA

Le programme principal ("SA) :

x Traduit Préfixe + Suffixe en chaine α (Les XTOA provoquent des messages d'erreur pour les valeurs invalides),
x Affecte temporairement la fonction % à la touche spécifiée (PASN implique un message d'erreur pour un code touche invalide),
x Traduit la touche en code interne (25 octets, en comptant pour 1 octet la ligne 60 - qui évite un ENTER↑),
x Constitue le triplet définitif, à substituer au triplet temporaire,
x Fixe une borne (101 registres) à la zone explorée (la borne ne sera jamais atteinte car le triplet temporaire sera toujours trouvé avant),
x Cherche dans la table le triplet temporaire, en recomposant la structure des registres (ligne 108 = Code synthétique 241,240),
x Opère la substitution (Lignes 112 et 113), et achève par un TONE 8 .

Le programme peut être appelé par une routine composant les codes XROM (légaux), selon la procédure :

[Code XROM] (MM,FF) ENTER↑ [Code Tche] XEQ"XA

dont les messages d'erreur se trouvent en lignes 01 à 10.
L'ensemble tient sur les deux pistes d'une carte.

```
AFF. SYNTH.
 01*LBL 13
 02 RCLFLAG
 03 FIX 2
 04 "FN : "
 05 ARCL Z
 06 "⊢ ?"
 07 STOFLAG
 08 CLST
 09 BEEP
 10 PROMPT
 11*LBL "XA"
 12 X<>Y
 13 ENTER↑
 14 X<0?
 15 GTO 13
 16 INT
 17 ST- Y
 18 32
 19 X<=Y?
 20 GTO 13
 21 RDN
 22 640
 23 +
 24 X<>Y
 25 .64
 26 X<=Y?
 27 GTO 13
 28 ST* Z
 29 RDN
 30  E2
 31 ST* Z
 32 *
 33 +
 34 ENTER↑
 35 ENTER↑
 36 256
 37 MOD
 38 ST- Y
 39 X<>Y
 40 LASTX
 41 /
 42 X<> Z
 43*LBL "SA"
 44 CLA
 45 X<> Z
 46 XTOA
 47 RDN
 48 XTOA
 49 X<> [
 50 X<>Y
 51 "%"
 52 PASN
 53 INT
 54 .5
 55 X<=Y?
 56 CLX
 57 -
 58 ABS
 59 ENTER↑
 60 "αL"
 61 6
 62 /
 63 INT
 64 7
 65 -
 66 ABS
 67 X>0?
 68 SIGN
 69 -
 70 16
 71 *
 72 159
 73 MOD
 74 XTOA
 75 X<> [
 76 X<>Y
 77 X<> [
 78 XTOA
 79 X<> [
 80 X<>Y
 81 .1
 82 ABS
 83*LBL 01
 84 XEQ 02
 85 XEQ 04
 86 XEQ 04
 87 XEQ 02
 88 ISG L
 89 FC?C 01
 90 GTO 01
 91 TONE 8
 92 CLST
 93 CLA
 94 RTN
 95*LBL 02
 96 "αααα"
 97 XEQ 03
 98 X<> IND L
 99*LBL 03
100 X<> [
101 X<> c
102 X<> [
103 RTN
104*LBL 04
105 X<> [
106 "⊢***"
107 X<> [
108 ""
109 XEQ 05
110 X=Y?
111 SF 01
112 X=Y?
113 X<> Z
114 XEQ 06
115 "⊢***"
116 XEQ 06
117*LBL 05
118 X<> [
119*LBL 06
120 "⊢****"
121 X<> \
122 END
          CAT 1
LBL"XA
LBL"SA
END
222 BYTES
```

A. DISEL (PPC-PC-52)

Ce programme, après plusieurs essais, ne possède aucun des problèmes connus sur les autres PRGM de ce style quelque soit sa position en mémoire

Philippe

Le "BJ": quoi, comment ?

Un collègue me téléphone pour me demander ce qu'est le Byte Jumper et ce qu'on peut en faire : vaste question bien embarrassante (?)....

Je lui fait remarquer gentiment que je ne me suis engagé à répondre qu'aux questions relatives à la PRGMmation "classique", n'étant qu'un simple débutant en PS et donc non encore qualifié pour répondre à ce genre de question.

Cela étant, je ne l'ai tout de même pas laissé totalement sur sa faim et je vais vous indiquer ce que je lui ai proposé.

Le BJ s'ASN avec les CODEs 241,65 et s'utilise en mode RUN (PRGM Off). Pour ASN ces CODEs, il faut disposer au minimum du PRGM "MKX" publié P92 de "HP-41 SYNTHETIC PROGRAMMING MADE EASY" de Keith JARETT, du module XF (27ONF à "La Règle à Calcul °) mais également du PRGM "LBX" (issu du livre précité) pour pouvoir créer les FS et les chaînes ALPHA..... Cela ressemble étrangement au chien qui tourne en rond (?) pour essayer de rattraper sa queue....

Il y a plusieurs façon de contourner le problème et la plus simple aurait été que j'envoie les CARTes magnétiques correspondantes mais à condition qu'il eut un lecteur (?).C'était presque la quadrature du cercle!....

Je pense que ce n'est pas le meilleur service à rendre que de livrer une "nourriture prémâchée" et il existe une citation qui s'adapte bien à ce principe....

J'ai donc demandé à ce charmant collègue de commencer par introduire dans sa "41", le BG, comme c'est indiqué dans 2 N°s de JPC ainsi que les autres ASN conseillées puis de lire attentivement les "papiers" traitant de la création des FS et des CHaînes ALPHA et ensuite, de me rappeler.

Le lendemain (très doué, collègue!), notre ami était fin prêt et je lui ai donc dicté le PRGM "LBX" qui me semblait être la suite logique. Le mode d'emploi de ce PRGM lui ayant été indiqué, j'ai dicté dans la foulée (?), le PRGM "MKX"(avec, bien entendu, également, son mode d'emploi) et nous nous sommes quittés, l'un semblant tout heureux de posséder la clé résolvant tous ses problèmes et l'autre, satisfait "du devoir accompli" (?).....

J'avais naturellement précisé que le BJ n'était pas identique au BG mais que je répondrais à ce problème dans JPC.

Je vais me contenter d'indiquer que le BJ s'utilise en mode RUN (déjà dit = excuses!) et qu'il place des CARactères en ALPHA qui peuvent être décodés au moyen de la F ATOX. Des recherches méthodiques et "patientes" doivent permettre de découvrir COMMEMT agit ce BJ et je pense que c'est un très bon exercice et je vous assure que cela permet de mieux assimiler (comprendre) qu'en

```
                    CAT 1
LBL'LBX
END          160 BYTES
LBL'MKX
END          123 BYTES
.END.        18 BYTES
                  CLKEYS
                   FIX 0
                   CF 28
                   SF 29
              241, ENTER↑
               65, ENTER↑
               12,
                XEQ "MKX"
                   PRKEYS

USER KEYS:
 12 XROM 05,01
(PRGM ON)          GTO ..
 PACKING
 01 ENTER↑
 02 "ABCDEFG"
 (PRGM OFF)              Σ0
 ←wABCDEFG (en ALPHA)
          ASN "ATOX" 11
                       Σ+
 NONEXISTENT (SF27)
                      ATOX
          131,         ***
                      ATOX
          247,         ***
                      ATOX
 1°:A      65,         ***
                      ATOX
 2°:B      66,         ***
                      ATOX
 3°:C      67,         ***
                      ATOX
 4°:D      68,         ***
                      ATOX
 5°:E      69,         ***
                      ATOX
 6°:F      70,         ***
                      ATOX
 7°:G      71,         ***
                      ATOX
            0,         ***
OK ? ,etc....
```

lisant la meilleure des explications....

Ne désirant aucunement monopoliser les pages de JPC, je demande tout de même qu'un collègue se dévoue pour nous écrire un article sur ce fameux BJ dont les subtilités sont indiquées dans "SYNTHETIC PROGRAMMING on the HP-41C" de WICKES et, comme ce bouquin est épuisé, je propose d'offrir une copie de ce livre à celui qui se dévouera à en traduire les passages les plus intéressants. Ce travail pourrait d'ailleurs être réparti entre 2 ou 3 "traducteurs" qui recevraient chacun une copie et qui devraient se répartir le travail. Il semble évident que cette traduction devrait être compréhensible par des Français qui ont l'esprit cartésien (?) et qu'il faudrait donc éviter une traduction "mot par mot"....

Merci à ceux qui ont la chance de bien maîtriser l'américain, ce qui est bien loin d'être mon cas.

(°) Publicité gratuite mais je pense, méritée.

rS (P20+T178)

S.O.S. !

La FIG.5 de "AU FOND" (JDD) nous donne les CODEs des touches trouvés dans les REG d'ASN.

Page 219 du Manuel du module PLOTTER HP82184A il est indiqué que pour créer en CB, une Fonction ASN kk "xyz...", le CODE à introduire dans l'Octet 4 doit être :

A x 16 + B (x = signe de multiplication)

et je vais recopier textuellement ce qui est indiqué et que je n'ai pas réussi "à traduire" :

A=ABS(keycode)DIV10,B=ABS(keycode)MOD10. If keycode < 0 then A= [ABS(keycode)Div1048] MOD16

Bien que cela me sembla relativement simple à priori, je n'ai pas réussi à exploiter cette formule et je dois vous avouer que c'est assez décevant.....

Il se trouvera certainement un collègue qui réussira à décrypter cette formule.....

Il est également indiqué que l'Octet 5 :
1st ASCII Character
ce qui semblerait indiquer (?):
1er CARactère ASCII,
que l'Octet N+3 :
N-1 ASCII Character
ce qui devient de plus en plus nébuleux,
et que l'Octet N+4 :
Nth ASCII Character (N < 7)
ce qui m'a plongé dans une complète perplexité.

Pour reprendre une formule (expression) bien connue :

<< Ne me laissez-pas me noyer idiot ! >>

rS (P20)

Vive le PETIT THEATRE !

Nous devons bien vivement remercier notre Président pour l'important travail long et fastidieux qu'il a réalisé dans les N°s 1 à 3 de JPC. Seuls ceux qui n'ont jamais tapé à la machine ignoreront "ce que c'est" mais il faut cependant souhaiter qu'aucune coquille ne s'est glissée subrepticement dans la mayonnaise. Nous ne travaillons pas "au mixer" et certaine "coquille pourrait nous rester en travers de la gorge..... Je sens que cette remarque ne va pas me faire prendre du galon....

Jusqu'à présent, j'ai l'impression que nous sommes derrière le rideau du théâtre, avec les machinistes et que nous rassemblons tout le matériel (je crois qu'on dit: accessoires ?) nécessaire à la future représentation.

Après ces considérations "gastrono-théâtrales", il n'en reste pas moins vraique nous attendons avec impatience que résonnent les 3 coups annonçant la Première.....

Donc un grand MERCI à Philippe mais ne nous laisse pas trop languir ! rS (P20)

S.O.S. !

Un collègue de travail que j'ai réussi à convertir à notre religion (?) m'a demandé s'il était possible d'arrêter le chrono du module HORLoge par PRGM. Il est amateur de photographie et développe lui-même ses photos et c'est donc pour cette discipline qu'il aurait besoin de cette "Fonction".

Quelqu'un a-t'il résolu ce problème ?

Il doit bien y avoir des Amateurs "PHOTO" parmi nos collègues ! rS (P20+T178)

La "41" seraient-elle MISOGyne ?

Vous avez dû constater, et j'espère à votre grand désespoir (?), que les prénoms féminins sont extrêmement rares parmi la liste des membres PPC et je vais m'abstenir d'en profiter pour en tirer des conclusions désobligeantes.....

Je suis devenu contre mon gré, un irascible misogyne, ce qui ne m'empêche pas de beaucoup aimer les femmes mais à faibles doses, comme le poison..... Rassurez-vous, je ne vais pas développer le fond de ma pensée mais je tenais cependant à poser la question : "pourquoi le sexe "dit faible" se désintéresse-t'il de ce qui nous passionne ?"

Je ne désire pas une psychanalyse du problème mais j'aimerais bien avoir votre avis et bien entendu, celui de ces dames et demoiselles. Comme à ma connaissance, il ne se trouve qu'un seul prénom féminin dans la liste des membres PPC-PC : Pilou va se dévouer pour me répondre.....

Je vais me garder de trahir le secret (?) qui entoure cette charmante présence parmi nous....

Venez à nous, Dames et Damoiselles, vous constaterez que même les misogynes savent vous Aimer !

rS (P20) (robert)

Rambarde pour BG

D'abord une petite précision au sujet du mot "rambarde" qui signifie garde-corps et que beaucoup d'auteurs n'ayant jamais mis les pieds sur un bateau, appellent "bastingage". La rambarde est destinée à retenir les marins lorsqu'il y a du roulis provoqué par les caprices de la mer ou par la fréquentation de Dionysos (les marins préfèrent les Grecs....). Le bastingage est l'emplacement où l'on range les hamacs et vous remarquerez que son séjour dans "la Royale" a marqué votre serviteur (6137 T51, pour les connaisseurs....).

J'en entends dire: "il est en train de nous monter un bateau !". Restons ou redevenons sérieux§

Vous savez que pour utiliser le BG (BJ,-PRINTER-etc...), il est nécessaire de "travailler" dans un PRGM "fermé par le Haut", c'est-à-dire, ne pas "BéGer" en tête de la MEM PRGM: un LBL quelconque fait l'affaire et si vous travaillez dans un PRGM en cours, la condition est obligatoirement remplie.

Il est également nécessaire que le PRGM soit "ouvert" vers le Bas, c'est-à-dire NON PACKé par un GTO .. mais cela présente un petit danger car nous travaillons alors à la limite du domaine d'Alice et ce n'est pas toujours une "Merveille" que de s'y retrouver..... Ne souriez-pas, s'en sortir n'est pas toujours évident pour un débutant et un CATALOG 1 ou autre GTO . ØØØ ne résolvent pas toujours le problème.....

Je vous indique donc une méthode que j'ai adoptée et qui m'a, jusqu'à présent (?), toujours donné satisfaction : placer un bon paquet d'ENTER (ou toute autre Fonction, exceptée STOP) et travailler au dessus de cette "rambarde". Si vous retrouvez des CAR du CODE DEC 131 (flèche dirigée vers la G), ne vous affolez-pas, ce sera un (ou des) ENTER qui aura été "bouffé" et vous comprendrez alors pourquoi j'appelle le BG "Bouffeur d'octets", plutôt que sauteur de je ne sais quoi ; ce dernier terme s'appliquant mieux à la Fonction (BUG ?) W" (CODE DEC 31) qui, je l'espère, a été appréciée par quelques collègues (?).... rS (P20)

Ne pliez-pas, S.V.P.!

Dans la mesure du possible, essayer d'envoyer vos feuilles 21x29,5cm dans une enveloppe du même format car le pliage détériore l'écriture et la qualité de la reproduction s'en ressentirait. Essayer donc de présenter vos copie avec le plus de soin possible. Les PRGMmeurs sont des gens soigneux, n'est-ce pas ? rS (P20)
P.S.: vous pouvez répondre: oui papa ! .....

IMPrimante "IL" ou pas "IL" ?

Etant adhérent PPC-T, j'ai écrit un "papier" semblable dans le N° 5 du journal de Toulouse (eh oui, ils ont dû supporter ma prose !) et j'ai pensé que la comparaison de ces 2 IMP pouvait également intéresser mes collègues de PPC-PC.

Je précise tout de suite que tout ce que je vais avancer, n'est en fait que mon avis personnel et j'accepte qu'il soit contesté mais essayez tout de même de ne pas écrire de bêtises comme cela m'a été reproché par notre grand Président JDD..... Je lui indique au passage qu'il est parfaitement possible d'imprimer des lignes complètes (par ACSPEC et non BLDSPEC; +PS) avec l'IMP 82143A et j'en ai envoyé la preuve à un adhérent qui m'a écrit à ce sujet. Il risque même de gagner une caisse de Champagne s'il arrive à prouver que ce je lui ai envoyé n'a pas été imprimé avec l'IMP non "IL"....(?) .

Revenons à notre comparaison.

L'IMP 82162A "IL" ne peut être utilisée qu'avec le module INTERFACE "IL" HP 82160A qui doit, normalement vous être offert gratuitement lors de l'achat de l'IMP (3 800Ⓕ, au moment où je vous écrit). L'INTERFACE est constitué(e) d'un connecteur un peu plus long que celui de l'IMP "normale" et est muni de 2 câbles (fils de 2 conducteurs chacun, genre "Scindex") permettant de constituer la fameuse boucle. Dans le cas de l'utilisation de l'IMP seule, les fiches (d'un modèle différent) placées aux extrémités des câbles sont connectées à l'arrière de l'IMP et un petit interrupteur placé au dos du module doit être placé sur la position "ENABLE". La position "DISABLE" est réservée à l'utilisation de l'IMP "normale" simultanément avec la boucle "IL" et, dans ce cas, vous avez obligatoirement 2 PORTS découpés (la boucle peut être, par exemple, fermée sur le lecteur de K7,c.q.f. Dire...

Ces renseignements sont superflus pour ceux qui disposent de ce matériel mais peuvent être utiles à d'éventuels nouveaux utilisateurs. En effet, les manuels correspondants sont tellement mal rédigés qu'il est nécessaire, pour découvrir les "effets" de la plupart des fonctions, de s'armer de patience et de faire des essais. A titre indicatif, je signale que le manuel de l'IMP "IL" n'a que 18 (dix huit) pages réellement utilisables (et encore!) y compris la table des matières. J'estime que c'est purement et simplement INADMISSIBLE de la part de belle-mère HP et, quand on tient compte du nombre de coquilles introduites sans errata et qui interdisent pratiquement l'utilisation du mode "échappement".... Le module TIME dont le prix est sans commune mesure avec celui de l'IMP "IL" (qui a été vendue 4 800Ⓕ), est accompagné d'une notice de 87 pages + Addendum, ce dernier n'étant pas traduit en français, il est vrai.(ce doit être trop fatiguant !).

A mon avis, la seule supériorité vraiment appréciable de l'IMP "IL" est la capacité de sa MEM tampon (101 octets) qui permet d'imprimer des CB et je pense même, qu'à part cette possibilité, l'IMP non "IL" est préférable. En effet, notre belle-mère à tous n'a rien trouvé de mieux que d'utiliser qq CODES ASCII pour des commandes telles que : retour chariot, saut de lignes, etc..., ce qui interdit pratiquement l'utilisation de ce CODE ASCII pour le codage-décodage de valeurs numériques (pour stockage en FICHiers ALPHA de nombres entiers positifs compris entre Ø & 255 = gain d'octets, par exemple). Les commandes spéciales comme : fin dd ligne ignorée (LF) ou justification de texte à Gauche (que l'on obtient par PRBUF) ou à Droite (que l'on obtient avec ADV) ou au centre avec FMT sont d'ailleurs issues du MODULE "IL".

Mise à part l'impression de CB, on ne voit pas l'utilité de la capacité de la MEM tampon. A quoi sert d'accumuler plus de 24 CAR (avec ACA et ACX) si une ligne ne peut pas en contenir plus ?
Ne me dites-pas qu'on peut aller à la ligne car je vous demanderais de me faire le PRGM qui fait la césure entre les syllabes.....(?).

Les défenseurs de l'IMP "IL" diront qu'elle imprime automatiquement l'heure et la date dès qu'on "PRP" ou "LIST" un PRGM, à condition de posséder le module TIME.....

Comme l'a écrit JDD, la MEM tampon sert également à stocker les CODES de qq commandes mais je crois pouvoir lui indiquer que ces CODES peuvent être incorporés aux chaînes ALPHA ASCII au moyen de la PS..... et je précise que je ne suis qu'un débutant, ce que j'ai indiqué à plusieurs reprises. J'ai toujours lu les "réponses" de JDD avec l'intérêt qu'elles méritaient.

J'espère simplement vous avoir donné qq indications vous permettant de faire votre "choix" en adultes et sans engager ma responsabilité....

Imprimez beau et bien ! rS (P20)

"CNR" = YTOAX

```
01*LBL "CNR"
 02 ENTER↑
 03 ENTER↑
 04 ALENG
 05 -
 06 AROT
 07 ATOX
 08 R↑
 09 XTOA
 10 R↑
 11  E
 12 +
 13 CHS
 14 AROT
 15 END
```

rS

```
ABCDEFGHIJKLMNOPQRSTUVWX
          47 ENTER↑
           5
          XEQ "CNR"
ABCDE/GHIJKLMNOPQRSTUVWX
```

Le PRGM ci-contre remplace le CAR dont le N° d'ordre est placé en X par le CAR dont le CODE DEC ASCII a été placé en Y et est donc l'équivalent de la fonction YTOAX du module EXTended I/O.

Dans l'exemple indiqué, 47 est le CODE de "/" que nous désirons substituer à "F" qui est le CAR N°5 de la chaîne qui a été placée en ALPHA (24 CAR au maxi).

Il ne nous restera plus qu'à réaliser le PRGM qui fera le même travail que la Fonction ANUMDEL qu'il ne faut pas confondre avec ANUM du XF. Rappelons que ANUMDEL place en X la première VAL NUM contenue dans une CH ALPHA (première en partant de la G) et efface les CAR situés à G de cette VAL. Je crois avoir écrit (?) que cette F est beaucoup plus intéressante qu'elle ne paraît.... Je vous laisse le soin de faire le PRGM correspondant.

Donc au plaisir de vous lire... rS (P20)

```
01♦LBL "CNX"
 02 ENTER↑
 03 ALENG
 04 -
 05 AROT
 06 ATOX
 07 STOP
 08 XTOA
 09 RDN
 10 CHS
 11  E
 12 -
 13 AROT
 14 END
```

rS

```
ABCDEFGHIJKLMNOPQRSTUVWX
                 9,
                 XEQ "CNX"
                    CF 29
                 74    ***
                       RUN
ABCDEFGHIJKLMNOPQRSTUVWX
                 23
                 XEQ "CNX"
                 88    ***
                       RUN
ABCDEFGHIJKLMNOPQRSTUVWX
                  3
                 XEQ "CNX"
                 68    ***
                       RUN
ABCDEFGHIJKLMNOPQRSTUVWX
```

```
01♦LBL "XTOAC"
 02 XTOA
 03 - E
 04 AROT
 05 END
```

```
ABCDEF
   71
 XEQ "XTOAC"
GABCDEF
```

"CNX" (ATOXX = suite)

Le petit PRGM ci-contre fait le même travail que la fonction ATOXX du module EXTended I/O. Il s'adresse donc à ceux qui ne disposent pas de ce module ou de la PPC-ROM.

Si nous plaçons en X le N° d'ordre d'un CAR d'une CHaîne placée en ALPHA et que nous XEQ'ons "CNX" (Code N en X), nous obtenons en X le CODE DEC ASCII et la CH est reconstituée.

Précisons que, comme avec ATOXX, le N° d'ordre doit être compté de G à D en n'oubliant pas qu'en Informatique, le premier CAR a le N° Ø et donc, si vous comptez de façon "traditionnelle" (?), il vous suffira d'ajouter 1.....

Le nbre maximal de CAR que nous pouvons introduire en ALPHA est 24, ce qui m'a semblé suffisant car, s'il serait possible de "traiter" 28 CAR, le PRGM correspondant utilisant la PS, serait plus compliqué et fera peut-être l'objet d'un "papier" ultérieur.Voici donc un sujet de PRGM....

La VAL maximale à placer en X pour le N° d'ordre est donc de 23.

Il serait possible de contrôler la longueur maximale de la CH introduite en ALPHA ainsi que la VAL max correspondante introduite en X pour le N° d'ordre mais cela "alourdirait" le PRGM. Je laisse le soin à ceux qui désirent placer ces 2 "barrières", de réaliser la modif adéquate.

Une utilisation appropriée des fonctions du module XF permet d'obtenir tout ce dont nous avons besoin mais le MC va nous permettre, je l'espère, d'avoir toutes ces fonctions en ROM (PPC-ROM N°2, ROM d'outre-Rhin, etc... ?) car hélas, ce qui a été réalisé en France est plutôt "maigre". J'en profite pour déclarer que la "politique" de belle-mère HP qui impose un minimum de 500 unités pour la réalisation d'une ROM (reproduction), est pour le moins "DISCUTABLE" et mériterait l'étude d'une réalisation "sauvage". Il est étonnant que nos vénérables collègues "US" ou australiens n'aient pas encore résolu ce problème. Je m'associerais volontiers à ceux qui voudraient se lancer dans ce genre d'entreprise.

<< Allez la FRANCE, montrons ce que nous sommes capables de faire ! .....>> rS (P20)

REUNION GENERALE

Chers amis, j'en avait parlé dans la précédente circulaire mais je m'aperçoit que personne ne m'a encore répondu à ce jour. Effectivement, je vous proposait pour la date de notre première réunion générale le dernier week-end de septembre ou début octobre; et à savoir si vous étiez intéressés par le samedi ou le dimanche. Cette réunion se passera à Marly la ville chez Pierre Ravennes (PPC P C n°53)

A bientôt pour vos réponses PHILIPPE

Conversion HEX-DEC

```
11:09AM 26.06
01♦LBL "HD"
02 CLST
03♦LBL 01
04 "├ "
05 X<> ]
06 X=0?
07 GTO 01
08 X<> [
09 R↑
10 X<> \
11 "├♦ₐ"
12 X<> \
13 RDN
14 X<> [
15  E
16 *
17 39
18 -
19 X>0?
20 DSE X
21 9
22 +
23 X<0?
24 GTO 02
25 X<>Y
26 16
27 *
28 +
29 ,
30 GTO 01
31♦LBL 02
32 RDN
33 CLA
34 END
 9 REG
57 OCTETS
```

J'ai fait à l'attention de ceux qui n'ont pas la chance d'avoir une HP-16C, un PRGM permettant les conversions HEX-DEC et vice-versa mais j'avais surestimé mes compétences et je me suis aperçu que les PRGM de la PPC-ROM étaient plus courts et plus performants.....

Le coeur meurtri (?), je me suis donc résolu à proposer le PRGM "BD" de George ELDRDGE d'après une documentation fournie par John KENNEDY. Je me suis simplement contenté de remplacer le LBL ALPHA global par "HD" (Hex-Déc) et de placer 16 à la place de RCL Ø6 au Pas 26. Dans le PRGM de la PPC-ROM, il faut placer dans le REG Ø6 la BASE du nbre à convertir et donc, si vous voulez utiliser ce PRGM pour convertir des VAL dont les BASEs sont différentes de 16, il vous suffira de replacer RCL Ø6 au Pas 26.

UTILISATION: Placer en ALPHA la VAL à convertir puis XEQ "HD" et vous aurez en X la VAL équivalente en BASE 1Ø.

Exemple: ALPHA SHIFT 2 D SHIFT Ø ALPHA
XEQ ALPHA H D ALPHA ——————→ 72Ø,

Le SIZE minimal est ØØ6 (ØØ7 avec "RCL Ø6").

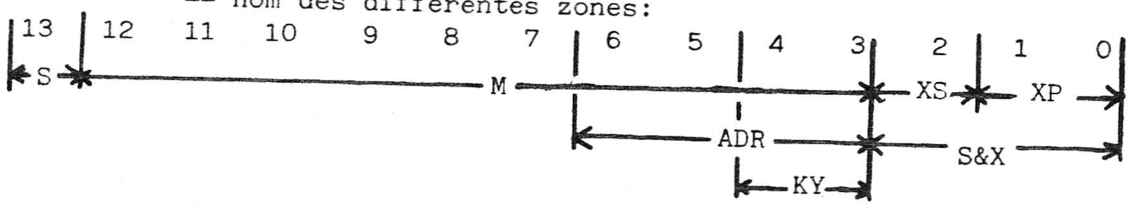
Les CODEs DEC du Pas 11 sont: 243 127 Ø 8

Le PRGM "DH" qui fait l'inverse de "HD" et qui sera la version légèrement modifiée de "TB" de la PPC-ROM, sera publié dans le prochain N° de JPC.

CHOC EN RETOUR (JPC V1N4P29)
++++++++++++++++++++++++++++++

6. - retour au registres "A, B, C"

-- nom des differentes zones:



CHAMPS

		NYBBLES	BITS
S	= Signe de la mantisse	13	52 - 55
M	= Mantisse	3 - 12	12 - 51
XS	= Signe de l'eXposant	2	8 - 11
X ou XP	= eXPosant	0 - 1	0 - 7
ADR	= ADResse	3 - 6	12 - 27
S & X	= Signe de l'exposant et eXposant	0 - 2	0 - 11
ALL	= TOUT le registre	0 - 13	0 - 55
KB ou KY	= registre KeY	3 - 4	12 - 19

+/

-- pointeurs:

R ou PT = PoinTeur sélectionné (P ou Q)
@R = au digit pointé par R
R ←- = du digit 0 au digit pointé par R
P - Q = entre les pointeurs (du Pième digit au Qième digit)
@R, + = au digit R et R+1
LD@R, - = charger au digit R et décrémenter R (R-1)
SLCT P / SLCT Q = sélectionner lequel des deux pointeurs est actif P ou Q

```
 8:47PM 06.07
01♦LBL "COMP"
02 7
03 PSIZE
04 "L"
05 EMDIR
06 CLD
07 CRFLAS
08 "P"
09 RCLPTA
10 STO 02
11 " ♦♦♦♦"
12 RCL [
13 "λ*"
14 RCL [
15 X<> c
16 RCL 00
17 STO [
18 "┣ABCDE"
19 X<> Z
20 STO \
21 "┣AB"
22 X<> \
23 STO 00
24 X<>Y
25 STO c
26 CLX
27 STO 00
28 STO 04
29 STO 06
30♦LBL 40
31 RCL 00
32 XEQ 30
33 STO 03
34 XEQ 50
35 X=0?
36 SIGN
37 FS? 00
38 GTO 60
39 STO 05
40♦LBL 07
41 RCL 00
42 XEQ 30
43 FC? 01
44 ST  06
45    03
46 RCL 04
47 7
48 /
49 INT
50 SEEKPT
51 GETX
52 STO [
53 RCL 04
54 7
55 MOD
56 X≠0?
57 XEQ 03
58 RCL \
59 X<> [
60 STO \
61 RCL 03
62 XTOA
63 RCL [
64 X<> \
65 STO [
66 RCL 04
67 7
68 MOD
69 6
70 -
71 CHS
72 X≠0?
73 XEQ 03
74 RCLPT
75  E
76 ST+ 00
77 ST+ 04
78 -
79 SEEKPT
80 RCL \
81 SAVEX
82 DSE 05
83 GTO 07
84♦LBL 41
85 RCL 00
86 RCL 02
87 X≠Y?
88 GTO 40
89 RCL 04
90 STO 02
91 CLX
92 STO 00
93♦LBL 81
94 RCL 00
95 XEQ 30
96 STO 03
97 XEQ 50
98 X=0?
99 SIGN
100 FS? 01
101 GTO 90
102 ST+ 00
103♦LBL 83
104 RCL 00
105 RCL 02
106 3
107 -
108 X>Y?
109 GTO 81
110 RCL 02
111 STO 00
112 RCL 06
113 256
114 MOD
115 XEQ 35
116 RCL 02
117  E
118 +
119 7
120 /
121 ENTER↑
122 INT
123 X≠Y?
124 ISG X
125 DEG
126 STO 00
127 SF 25
128 SEEKPT
129 ""
130 RCL [
131 FS? 25
132 SAVEX
133 "0♦♦♦"
134 RCL 02
135 16
136 /
137 INT
138 XTOA
139 RCL 02
140 16
141 MOD
142 LASTX
143 *
144 RCL 00
145 256
146 /
147 INT
148 +
149 XTOA
150 RCL 00
151 256
152 MOD
153 XTOA
154 RCL [
155 "λ*"
156 RCL [
157 X<> c
158 X<>Y
159 STO 00
160 X<>Y
161 X<> c
162 "FIN"
163 SF 11
164 OFF
165 AVIEW
166 "P"
167 STOP
168♦LBL 60
169 STO 01
170  E
171 X≠Y?
172 GTO 09
173 RCL 03
174 -
175 CHS
176 GTO 04
177♦LBL 09
178 RCL 00
179  E
180 +
181 XEQ 30
182  E2
183 X<=Y?
184 SF 06
185 RDN
186♦LBL 04
187 -
188 "L"
189 SEEKPTA
190 RDN
191 XTOA
192 RCL 04
193 / (
194
195 XTOA
196 RCL 04
197 LASTX
198 MOD
199 XTOA
200 APPREC
201 "P"
202 FLSIZE
203 RCL 01
204 STO 05
205 FS? 06
206 GTO 07
207 ST+ 00
208 GTO 41
209♦LBL 50
210 -
211 X<>F
212 RDN
213 X=0?
214 RTN
215 16
216 X>Y?
217 SF 00
218 SIGN
219 FS? 00
220 RTN
221 CLX
222 29
223 X=Y?
224 GTO 28
225  E
226 +
227 X=Y?
228 GTO 28
229 CLX
230 144
231 X>Y?
232 SF 02
233 SIGN
234 FS?C 02
235 RTN
236 CLX
237 176
238 X>Y?
239 SF 02
240 FS? 02
241 2
242 FS?C 02
243 RTN
244 CLX
245 192
246 X>Y?
247 SF 01
248 FS? 01
249 SF 04
250 FS? 01
251 2
252 FS? 01
253 RTN
254 CLX
255 206
256 X>Y?
257 GTO 51
258 X=Y?
259 SF 02
260 FS? 02
261 2
262 FS? 02
263 RTN
264 CLX
265 207
266 X=Y?
267 SF 00
268 CLX
269 2
270 FS? 00
271 RTN
272 CLX
273 223
274 X<Y?
275 SF 03
276 RDN
277 240
278 X>Y?
279 SF 01
280 -
281  E
282 +
283 FS? 01
284 3
285 RTN
286♦LBL 28
287 RCL 00
288  E
289 +
290 XEQ 30
291 240
292 -
293 2
294 +
295 RTN
296♦LBL 51
297 RCL 00
298 2
299 +
300 XEQ 30
301 240
302 X<Y?
303 GTO 51
304 3
305 RTN
306♦LBL 51
307 -
308 3
309 +
310 RTN
311♦LBL 30
312 CLA
313 STO [
314 7
315 /
316 INT
317 SEEKPT
318 GETX
319 X<> [
320 7
321 MOD
322 X≠0?
323 XEQ 03
324 STO \
325 "┣♦"
326 X<> \
327 STO [
328 ATOX
329 RTN
330♦LBL 03
331 "┣♦"
332 DSE X
333 GTO 03
334 RTN
335♦LBL 90
336 3
337 X=Y?
338 GTO 91
339 RCL 03
340 177
341 -
342 GTO 92
343♦LBL 91
344 RCL 00
345 2
346 +
347 XEQ 30
348 128
349 MOD
350♦LBL 92
351 "L"
352 -
353 SEEKPTA
354 RDN
355 XTOA
356 ASTO 04
357 SF 00
358♦LBL 05
359 CLA
360 ARCL 04
361 POSFL
362 X<0?
363 GTO 02
364 ENTER↑
365 INT
366 X≠Y?
367 GTO 05
368 GETREC
369 "┣♦♦♦♦♦"
370 CLX
371 STO \
372 "┣♦"
373 X<> [
374 ATOX
375 256
376 *
377 X<>Y
378 STO [
379 RDN
380 ATOX
381 +
382 RCL 00
383 X<>Y
384 FC? 00
385 GTO 08
386 X<=Y?
387 GTO 05
388♦LBL 08
389 -
390 CHS
391 -
392 "P"
393 SEEKPTA
394 RDN
395 CF 00
396 X<0?
397 SF 00
398 ABS
399 FC? 03
400 FC? 04
401 FS? 30
402 GTO 26
403 ENTER↑
404 SIGN
405 FC? 00
406 CHS
407 +
408 STO 05
409 7
410 MOD
411 2
412 *
413 RCL 05
414 1792
415 /
416 INT
417 +
418 FC? 03
419 208
420 FS? 03
421 224
422 +
423 ST+ 06
424 XEQ 35
425  E
426 ST+ 00
427 RCL 05
428 7
429 /
430 INT
431 256
432 MOD
433 ST+ 06
434 XEQ 35
435  E
436 ST+ 00
437 CLX
438 FS? 00
439 128
440 ST+ 06
```

```
441 XEQ 35
442  E
443 ST+ 00
444 GTO 83
445*LBL 02
446 -
447 SEEKPT
448 FS?C 00
449 GTO 05
450 "ERREUR"
451 SF 11
452 OFF
453 PROMPT
454*LBL 35
455 RCL 00
456 7
457 /
458 INT
459 SEEKPT
460 GETX
461 CLA
462 STO [
463 RDN
464 RCL 00
465 7
466 MOD
467 X≠0?
468 XEQ 03
469 X<> \
470 X<> [
471 STO \
472 R↑
473 XTOA
474 X<> [
475 X<> \
476 X<> [
477 RCL 00
478 7
479 MOD
480 6
481 -
482 CHS
483 X≠0?
484 XEQ 03
485 RCLPT
486  E
487 -
488 SEEKPT
489 RCL \
490 SAVEX
491 RTN
492*LBL 26
493 2
494 FC? 00
495 CHS
496 +
497 STO 03
498 111
499 X<>Y
500 X>Y?
501 GTO 22
502 177
503 ST+ 06
```

"Il n'y a pas de petites économies"

Tous ceux qui sont de ma génération ont été marqués par les privations de la "dernière" et nous vivons une époque où il est pratiquement indispensable d'éviter le gâchis, à moins d'avoir de l'argent à jeter par les fenêtres....

"nous n'avons pas de pétrole, nous devons donc puiser dans nos idées" et le Français, c'est bien connu, a des idées, il a des idées sur tout et il a surtout des idées, comme l'a dit ce comique si controversé....

Vous avez constaté que les CB occupent une place appréciable dans JPC, bien que j'ai essayé de la minimiser de mon mieux.

Je viens d'adopter une nouvelle présentation qui va permettre d'économiser pas mal de place et je pense que la disposition "dans le sens de la hauteur" devrait permettre une lecture plus facile.

J'ai adopté le nbre maximal (16) d'Octets "code-barrables" dans un RANG et, comme précédemment, 9 Octets PRGM étaient "code-barrés" par RANG et que maintenant il y en a 13, vous pourrez apprécier l'économie réalisée.

Vous trouverez ci-joint le PRGM (merveilleux!) COMP de Frédéric POUPON (JPC N°3 P26) que j'ai dû partager en 2 à cause de la capacité disponible dans la MEM principale, en tenant compte que les PRGM qui me servent à "code-barrer" occupent une place appréciable, principalement la partie qui permet d'écrire "dans le bon sens". Je serais donc obligé d'effectuer cette scission dans tous les PRGM d'une certaine longueur (1).

Je pense qu'il est inutile (?) de vous expliquer comment le "recollage" doit s'effectuer....

Frédéric, nous te félicitons pour ce merveilleux PRGM et nous attendons avec impatience, tes prochaines créations.

Joyeuses et bonnes PRGMmations, rS (P20)

(1) 142 REG au maximum.

```
504 XEQ 35
505  E
506 ST+ 00
507 128
508 RCL 03
509 7
510 /
511 INT
512 RCL 03
513 7
514 MOD
515 16
516 *
517 +
518 FS? 00
519 +
520 ST+ 06
521 XEQ 35
522  E
523 ST+ 00
524 GTO 83
```

```
525*LBL 22
526 "GTO 2/3 OCTS"
527 SF 11
528 OFF
529 PROMPT
530 END
```

CODES DEC :

```
P 11: 245 32 0 0 0 0
P 13: 243 11 224 1
P129: 240
P133: 243 16 0 0 0
P155: 243 11 224 1
P325: 242 127 0
P331: 242 127 0
P369: 246 127 0 0 0 0 0
P372: 242 127 0
```

ATTENTION :

LeS PaS 187,210,352,391,

eT 446 = CODE DEC 28

<cDec 28 = 0 = , >

(R.S.)

```
 01♦LBL "APPRH"
SF 12  "TEXTE "  AON
PROMPT  AOFF  ALENG  6.1
/  INT  1 E3  /  STO 04


 14♦LBL 00
ASTO IND X  ASHF  ISG X
GTO 00

 19♦LBL 01
CLA  ARCL IND 04  ATOX
X=0?  GTO 07
ASTO IND 04  CLA  XTOA
ASTO X  XEQ IND X



 30♦LBL 02
1 E2  *  FRC  LASTX  INT
X=0?  GTO 05  SF 08



 39♦LBL 03
X<>F  X<>Y  4 E-3  CLA


 44♦LBL 04
FS? IND X  "⊢"
FC? IND X  "⊢  "  ISG X
GTO 04  FS? 12  AVIEW
RDN  FS?C 08  AVIEW
GTO 02

 57♦LBL 05
14  2  FS?C 09  /
FC? 12  2  FC? 12  /
"  "

 67♦LBL 06
AVIEW  DSE X  GTO 06
GTO 01

 72♦LBL "A"
.15202015  RTN

 75♦LBL "B"
.3121211  RTN

 78♦LBL "C"
.1417171  RTN

 81♦LBL "D"
.31171714  RTN

 84♦LBL "E"
.31212117  RTN

 87♦LBL "F"
.31202016  RTN

 90♦LBL "G"
.14172106  RTN

 93♦LBL "H"
.31040431  RTN

 96♦LBL "I"
.31  RTN

 99♦LBL "J"
.0201013  RTN

102♦LBL "K"
.31041017  RTN

105♦LBL "L"
.31010101  RTN

108♦LBL "M"
.3108040831  RTN


111♦LBL "N"
.3103040231  RTN


114♦LBL "O"

115♦LBL "0"
.14171714  RTN

118♦LBL "P"
.31202008  RTN

121♦LBL "Q"
.1417211813  RTN


124♦LBL "R"
.31202011  RTN

127♦LBL "S"
.08212102  RTN

130♦LBL "T"
.1616311616  RTN


133♦LBL "U"
.3001013  RTN

136♦LBL "V"
.2802010228  RTN


139♦LBL "W"
.300102013  RTN

142♦LBL "X"
.1710041017  RTN

145♦LBL "Y"
.2404070424  RTN


148♦LBL "Z"
.17192125  RTN

151♦LBL "1"
.040831  RTN

154♦LBL "2"
.09192109  RTN

157♦LBL "3"
.1017211  RTN

160♦LBL "4"
.06102302  RTN

163♦LBL "5"
.29212118  RTN

166♦LBL "6"
.14212102  RTN

169♦LBL "7"
.16192024  RTN

172♦LBL "8"
.1021211  RTN

175♦LBL "9"
.08212114  RTN

178♦LBL "-"
.040404  RTN

181♦LBL "+"
.041404  RTN

184♦LBL "*"
.10041  RTN

187♦LBL "/"
.020408  RTN

190♦LBL "="
.050505  RTN

193♦LBL "%"
.1702040817  RTN


196♦LBL ","
.0102  RTN

199♦LBL "<"
.1417  RTN


202♦LBL ">"
.1714  RTN

205♦LBL "↑"
.02202428  RTN

208♦LBL "a"
.2114311421  RTN


211♦LBL "b"
.2727042727  RTN


214♦LBL "c"
.2110211021  RTN


217♦LBL "d"
.102114211  RTN

220♦LBL "e"
.1218091812  RTN


223♦LBL "Σ"
.17272117  RTN

226♦LBL " "
.  SF 09  RTN

230♦LBL "?"
.2416232028  RTN


233♦LBL "."
.01  RTN

236♦LBL ":"
.09  RTN

239♦LBL "'"
.0816  RTN

242♦LBL "$"
.0414311404  RTN


245♦LBL "∠"
.162416  RTN

248♦LBL "#"
.1608  RTN

251♦LBL 07
ISG 04  GTO 01  END
```

```
01♦LBL "MORSE"
"TEXTE :" AON PROMPT
AOFF ALENG 6.1 / INT
1 E3 / STO 04

13♦LBL A
ASTO IND X ASHF ISG X
GTO A

18♦LBL 00
CLA ARCL IND 04 ATOX
X=0? GTO 03
ASTO IND 04 CLA XTOA
2 STO Y ASTO X AVIEW

31♦LBL B
DSE Y GTO B XEQ IND X

35♦LBL "?"
X=0? GTO 00 10 * FRC
LASTX INT XEQ IND X
X<>Y GTO "?"

46♦LBL 01
TONE 2 RTN

49♦LBL 02
TONE 8 RTN

52♦LBL "A"
.12 RTN

55♦LBL "B"
.2111 RTN

58♦LBL "C"
.2121 RTN

61♦LBL "D"
.211 RTN

64♦LBL "E"
.1 RTN

67♦LBL "F"
.1121 RTN

70♦LBL "G"
.221 RTN

73♦LBL "H"
.1111 RTN

76♦LBL "I"
.11 RTN

79♦LBL "J"
.1222 RTN

82♦LBL "K"
.212 RTN

85♦LBL "L"
.1211 RTN

88♦LBL "M"
.22 RTN

91♦LBL "N"
.21 RTN

94♦LBL "O"
.222 RTN

97♦LBL "P"
.1221 RTN

100♦LBL "Q"
.2212 RTN

103♦LBL "R"
.121 RTN

106♦LBL "S"
.111 RTN

109♦LBL "T"
.2 RTN

112♦LBL "U"
.112 RTN

115♦LBL "V"
.1112 RTN

118♦LBL "W"
.122 RTN

121♦LBL "X"
.2112 RTN

124♦LBL "Y"
.2122 RTN

127♦LBL "Z"
.2211 RTN

130♦LBL "1"
.12222 RTN

133♦LBL "2"
.11222 RTN

136♦LBL "3"
.11122 RTN

139♦LBL "4"
.11112 RTN

142♦LBL "5"
.11111 RTN

145♦LBL "6"
.21111 RTN

148♦LBL "7"
.22111 RTN

151♦LBL "8"
.22211 RTN

154♦LBL "9"
.22221 RTN

157♦LBL "0"
.22222 RTN

160♦LBL " "
PSE CLX RTN

164♦LBL 03
ISG 04 GTO 00 END
```

Ces deux programmes requièrent un Size 005 et le module XFUNCTIONS.

Morse : TEXTE vous demande un texte de 24 caractères au maximum. Ensuite faire R/S et écouter...avec de l'entrainement cela peut devenir un jeu.

Une personne met un texte et une ou plusieurs autres doivent décripter le message.(sans regarder la fenêtre de la machine bien sûr...)

Apprh : (nécessite pour bien faire une imprimante)
Texte demande un texte de 24 caractères au maximum mais on peut cummuler plusieurs exécutions sans coupures au niveau du travail de l'imprimante.
Faire R/S et c'est parti...automatiquement en majuscules (grands caractères). Pour passer en minuscules (petits caractères) faire un R/S,CF 12,R/S.entre deux caractères de l'affichage. Remplacer CF 12 par SF 12 pour repasser en majuscules.

Remarque : en minuscule et avec un peu d'habileté on peut facilement écrire 2 fois sur le papier, dans un sens et dans l'autre.

Séparer le message en deux moitiés égales. Faire imprimer la première partie, enlever le papier de l'imprimante et remettre celui-ci dans l'autre sens, pour imprimer la suite.

JPC

Apprh permet aussi de générer vos propres caractères. Chaque caractère est codé en colonne, elle même codée en binaire (00 à 31) pour des groupes de 2 lignes. Soit une matrice 5X5. Il suffit de changer le code d'un des LBL^T (après LBL^T,)

Comme d'habitude certaines lignes restent cachées par l'inaptitude de l'imprimante. Ne vous étonnez pas s'il s'en trouve de même pour vos LBL^T alpha

Morse : au pas 47 Tone Σ-
50 Tone E/X-1

Apprh : 45 F2;7F;1F on peut mettre n'importe quel 2° caractère, les colonnes seront alors imprimées avec le caractère choisi. (voir petit tableau)

Clavier de la HP 41

codage des ASN des LBL^T alpha

	1	2	3	4	5
1	01 Σ^+ 09	17 1/x 25	33 $\sqrt{x}$ 41	49 LOG 57	65 LN 73
2	02 $x^{\circ}y$ 10	18 R↓ 26	34 SIN 42	50 COS 58	66 TAN 74
3	03 shift 11	19 XEQ 27	35 STO 43	51 RCL 59	67 SST 75
4	04 enter 12	20 28	36 CHS 44	52 EEX 60	68 ← 76
5	05 − 13	21 7 29	37 8 45	53 9 61	69 77
6	06 + 14	22 4 30	38 5 46	54 6 62	70 78
7	07 × 15	23 1 31	39 2 47	55 3 63	71 79
8	08 % 16	24 0 32	40 . 48	56 R/S 64	72 shift non shifté 80

Code décimal du 1er caractère (fantôme) du LBL^T alpha

Après avoir crée le LBL^T alpha, préassigné à une touche passer une carte magnétique d'un programme quelconque. l'assignation sera validée.

A vos claviers !!!

Dolhen Yann 9 rue Béranger
60200 Compiègne

R/C

```
01♦LBL "BacHe"
02 GTO 07
03♦LBL 03
04 "ERREUR"
05 BEEP
06 PROMPT
07♦LBL 07
08 SF 00
09 CF 01
10 "N ?"
11 PROMPT
12 STO 00
13  E
14 X<>Y
15 +
16 STO 11
17 LASTX
18 CLA
19 "e SORTIE ?"
20 PROMPT
21 X=Y?
22 SF 01
23 X=Y?
24 CLX
25 STO 01
26 X≠0?
27 CF 00
28 X>Y?
29 GTO 03
30  E
31 X<>Y
32 +
33 STO 12
34 RDN
35 LASTX
36 CLA
37 "PROB X "
38 PROMPT
39 STO 02
40 STO 13
41 CLA
42 "POUR M PRIS?"
43 PROMPT
44 STO 03
45 FC?C 01
46 GTO 00
47 -
48 LASTX
49 X<>Y
50 CHS
51 STO 02
52 X<>Y
53♦LBL 00
54 X<>Y
55 X>Y?
56 GTO 03
57 X<>Y
58 X>Y?
59 CLA
60 FACT
61 STO 04
62 RDN
```

Le PRGM "BacHe" (Problème de Bachelier), résoud l'énoncé :

énoncé 1ère partie 1ère forme : -Soit un évènement, d'un tirage exhaustif, qui s'est réalisé "n" fois en "N" tirages.(1)

Ou encore: -Un tirage exhaustif ayant eu lieu dans une population de "U" éléments dont un nbre "u", possèdent un caractère distinctif (1)

énoncé 1ère partie 2ème forme

2ème partie de l'énoncé : On demande, lors de la poursuite exhaustive du tirage, quelle est la PROBabilité de tirer "m" fois l'évènement (le caractère distinctif), en "M" tirages.

Exemple: si N = 5 --- n = 3 succès
quelle est la PROB de: m = 2 succès pour M = 7 tirages

Réponse : PROBabilité
= 0,064 pour 1/7
= 0,024 pour 0/7
= 0,109 pour 2/7
= 0,152 pour 3/7
= 0,182 pour 4/7
= 0,191 pour 5/7
= 0,170 pour 6/7
= 0,109 pour 7/7
PROBabilité Totale = 1,000(pour 0 à 7/7)

-D'une façon générale, la PROB de "m" succès en "M" tirages, est régie par une loi interne de la forme :

$$\frac{M!}{m!(M-m)!}\,p^{m}(1-p)^{M-1}$$

- qui se combine avec la loi des PROB élémentaires des événements à venir :

$$\frac{(N-1)!}{(n-1)!(N-n-d')!}\,p(1-p)^{(n-1)}\,dp^{N-n-1}$$

on rappelle :

$$\int_0^1 p^{a}(1-p)^{b}dp = \frac{a!b!}{(a+b+1)!}$$

Nota : Le PRGM est limité à N et M inf ou égal à environ 50 par les capacités de la HP (OUT OF RANGE). En modifiant le PRGM on peut certainement faire mieux. Je crois avoir vu un PRGM pour les FACT 120 ! sur HP.

(1) exhaustif- de l'anglais "exhaust"= épuiser, "I presum"- signifie seulement que l'évènement survenu n'est pas remis en jeu (ce qui, qqfois, peut modifier la PROBabilité du mm évènement évalué dans un tirage postérieur.

Non exhaustif- par contre, peut simplement signifier qqfois que la PROB demeure constante qq soit le rang du tirage (pile ou face, dés, etc...)

- et aussi, bien sûr, que l'on remet l'évènement sorti en jeu, avant chaque nouveau tirage.

Nota : La PROB d'un évènement non exhaustif, de PROB élémentaire p, qui s'est réalisé "n" fois en "N" tirages, est donnée par la loi binomiale :

$b(n,N,p) = \frac{N!}{n!(N-n)!}\,p^{n}(1-p)^{N-n}$ (b est la PROB.

La PROB élémentaire est 1/2 à pile ou face,

```
63 LASTX
64 X<>Y
65 FACT
66 STO 05
67 RDN
68 LASTX
69 -
70 LASTX
71 X<>Y
72 FACT
73 STO 06
74 RDN
75 LASTX
76 R↑
77 FC? 00
78 GTO 00
79 CLX
80 RCL 11
81 STO 00
82 X<> T
83 CLX
84 RCL 12
85 X<> T
86*LBL 00
87 +
88 LASTX
89 X<> T
90 -
91 STO 07
92 LASTX
93 X<>Y
94 CLX
95  E
96 R↑
97 X<>Y
98 -
99 FACT
100 STO 08
101 RDN
102 LASTX
103 RCL 03
104 +
105 FACT
106 STO 09
107 CLX
108 RCL 07
109  E
110 -
111 FACT
112 STO 07
113 CLX
114 RCL 00
115  E
116 -
117 X<>Y
118 -
119 LASTX
120  E
121 -
122 FACT
123 STO 10
124 RDN
```

1/6 pour 1 dé, etc...).
- La PROB d'un évènement exhaustif, de se réaliser "n" fois en "N" tirages,étant donné qu'il appartient à une population de "U" éléments dont un nbre "u" possède le caractère propre à l'évènement considéré, est donnée par la loi hypogéométrique :

$$h\left(n,N,p=\frac{u}{U}\right)=\frac{U!\ N!\ (U-N)!\quad (U-u)!}{n!(U-n)!\ U!\ (N-n)!\ [(U-u)-(N-n)]!}$$

(h est la PROB. La PROB initiale est u/U)
Le Nota ci-dessue fournit immédiatement la PROB de de la première partie de l'énoncé du problème de Bachelier. Le PRGM "BacHe" résoud la PROB de "m" succès en "M" tirages ultérieurs étant donné ce qui est dit en première partie (formes 1 & 2) de l'énoncé. C'est tout différent.

Curieusement le problème de Bachelier peut servir au jeu du "LOTO" dans le cas où un lot de boules aurait un caractère différent de celui de l'ensemble :"Sortie plus aisée ! héh !".

Je signale toutefois qu'on change une série de 10 boules après chaque "LOTO", et que de temps en temps (quand ?), on change aussi les 49 boules.

Si quelqu'un possède des renseignements ou des idées à ce sujet, vous avez mon adresse.

```
125 LASTX
126 X<>Y
127 FACT
128 RDN
129 +
130 FACT
131 X<>Y
132 CLX
133 RCL 07
134 *
135 RCL 09
136 /
137 X<>Y
138 RCL 10
139 *
140 RCL 08
141 X<>Y
142 /
143 *
144 RCL 04
145 RCL 05
146 /
147 RCL 06
148 /
149 *
150 BEEP
```

```
151 "PROB X/M"
152 AVIEW
153 PSE
154 FIX 0
155 "X="
156 FC? 01
157 GTO 00
158 CLX
159 RCL 03
160 RCL 02
161 -
162 ARCL X
163 X<>Y
164 GTO 01
165*LBL 00
166 ARCL 13
167*LBL 01
168 "├/M="
169 ARCL 03
170 FIX 3
171 "├ "
172 ARCL X
173 PROMPT
174 CLA
175 FC? 00
176 GTO 00
```

```
177 RCL 02
178 X=0?
179 "├ )="
180 X=0?
181 GTO 02
182  E
183 -
184 X=0?
185 "├ ("
186 X=0?
187 GTO 02
188 X>0?
189 "├ (="
190*LBL 02
191 X<>Y
192 GTO 01
193*LBL 00
194 "├  "
195*LBL 01
196 FIX 7
197 ARCL X
198 PROMPT
199 GTO 07
200 END
  45 REG
 309 OCTETS
```

```
                SIZE 014
                  FIX 3
              XEQ "BacHe"
N ?
            5,000    RUN
e SORTIE ?
            3,000    RUN
PROB X
            2,000    RUN
POUR M PRIS?
            7,000    RUN
PROB X/M
X=2/M=7 0,109
e SORTIE= d SORTIE
```

JACQUES DURAND
IPETAN-STCAN/BA
6, Bd VICTOR
75015 PARIS CEDEX 15

Vivent les Codes-Barres !

D'abord un grand et sincère BRAVO à Gabriel GIL pour ses CB et pour les PRGM qui permettent de les imprimer; c'est du beau travail !

Une simple petite remarque: je pense que tu aurais pu te passer d'imprimer "ROw" devant le N° de RANG car en plus je ne sais pas ce que cela veut dire.....

Quelques conseils aux utilisateurs:

- il peut être utile et c'est même qqfois indispensable, de noircir les taches blanches qui se sont introduites dans les parties noires. Il convient de faire ces retouches soigneusement et d'attendre que l'encre sèche avant de balayer avec le crayon lecteur. Les défauts sont causés par la reproduction (photocopie).
- il est conseillé de se servir d'une règle pour guider le lecteur et de chercher éventuellement la "hauteur" de balayage qui convient le mieux, c'est-à-dire déplacer légèrement la règle dans le sens de la hauteur de la feuille en conservant le même angle d'inclinaison du crayon. Il est inutile d'appuyer fort et je crois même que, moins on appuie, mieux ça va. Vous pouvez éventuellement utiliser une chemise transparente fournie avec le lecteur mais j'avoue que je ne m'en sers pas étant donné que je stocke aussitôt sur K7.....

J'indique à Gabriel que depuis que j'ai le module PLOTTER (incorporé au module "IL" par Jacques VAUCELLE), je me sers d'un nouveau PRGM qui utilise les Fonctions très performantes de ce module. Ce PRGM a été envoyé à not 'Sident et sera donc (?) publié dans un prochain JPC, en CB, noblesse oblige.....

Je pense que si tu pouvais imprimer les CB sur une hauteur double (Jacques m'a dit que c'était possible) et dans le sens de la longueur d'une feuille (29,5cm), cela permettrait de gagner pas mal de place car, le nbre d'Octets "code-barrés" par RANG (et non par ROW!, na!) serait supérieur et avec une réduction de format, nous pourrions les présenter dans le bon sens; c'est d'ailleurs le procédé que j'utilise mais ceci n'est qu'un conseil et ne doit donc être considéré que comme tel.... L'imprimante de Jacques est très tentante mais je suis également intéressé par une table traçante qui permet d'obtenir des CB de toute beauté tels que ceux qui sont reproduit sur le Manuel du module PLOTTER HP82184A (230 pages!) rédigé en anglais, malheureusement et toujours en infraction avec la législation française..... Je finirais peut-être par mettre cela entre les mains d'une Association de Consommateurs.....

rS (P20+T178)

```
12CC 085 E
12CD 00E N
12CE 30F O
12CF 114 T
12D0 379
12D1 05A   GOLONG 16DE
```

LES TONES EN MICROCODE

La routine qui permet de générer les tones, se trouve en ROM à l'adresse 16DE; et controlle l'état du bender pour toute opération "TONE". La routine accepte des données de 2digits.

TONE est une instruction de la forme 9F,xy ; ou x est normallement Ø et y prend une valeur comprise entre Ø et 9 en programmation non synthétique. La durée est déterminée par le contenu du mot ROM 16F2+xy. Cette valeur est décrémentée dans une boucle et le tone est entendu jusqu'à ce que la valeur du mot soit Ø. La fréquence est donnée par y

pour utiliser le BENDER,stockez ØØ dans le registre T et stockez FF (hex) dans le registre ST. Les tones sont créés en échangeant T et ST (2D8 T()ST). Le nombre d'échange détermine la durée du tone; le nomd'instructions entre 2 échanges détermine la fréquencedu tone. La durée et la fréquence, dépendent de la fréquence de votre calculateur. Pour une 41 non accelérée, les machines ont approximativement un cycle de 158 microsecondes par instructions microcodes

```
16DE 3B8   C=REGN 14(d)
16DF 2BC   RCR    7
16E0 3D8   CST EX
16E1 30C   ?S=1   1
16E2 3A0   RTN NC
16E3 3C4   CLR ST
16E4 2D8   FST EX
16E5 358   ST=C
16E6 04E   C=0    ALL
16E7 130   LDI
16E8 0FF
16E9 3D8   CST EX
16EA 1BC   RCR    11
16EB 10E   A=C    ALL
16EC 07C   RCR    4
16ED 2A0   SETDEC
16EE 2BE   C=-C-1 S
16EF 260   SETHEX
16F0 3F1
16F1 058   GOSUB  16FC
```

```
16FC 1B0   C=STK
16FD 21A   C=A+C  M
16FE 330   CXISA
16FF 0A6   AC EX  X
1700 27E   C=C-1  S
1701 067   GOC    *+0C
1702 27E   C=C-1  S
1703 077   GOC    *+0E
1704 27E   C=C-1  S
1705 08F   GOC    *+11
1706 11E   A=C    S
1707 2D8   FST EX
1708 1BE   A=A-1  S
1709 3FB   GONC   *-01
170A 1A6   A=A-1  X
170B 3DB   GONC   *-05
170C 3E0   RTN
```

```
170D 2D8   FST EX
170E 1A6   A=A-1  X
170F 3F3   GONC   *-02
1710 3E0   RTN
1711 2D8   FST EX
1712 000   NOP    3
1713 1A6   A=A-1  X
1714 3EB   GONC   *-03
1715 3E0   RTN
1716 2D8   FST EX
1717 000   NOP    3
1718 000   NOP    3
1719 1A6   A=A-1  X
171A 3E3   GONC   *-04
171B 3E0   RTN
```

Exemple : TONE 9

TONE 9 (9F Ø9) a une période de 3 cycles par boucles x 2 boucles par cycles tone x 158 microsecondes par cycles soit 0,000948 secondes. La fréquence est donc de 1/0,000948=1055 hertz. Pour déterminer la durée, convertir le mot ROM 16F2 + Ø9 = 16FB qui est 215 (hex), en décimal, puis ajouter 1 pendant que la boucle décrémente jusqu'à Ø. Ce nombre (534 dec) est la durée d'utilisation du bender.

```
16F2 05B
16F3 065
16F4 073
16F5 087
16F6 0A1
16F7 0C9
16F8 10B
16F9 13F
16FA 18F
16FB 215
```

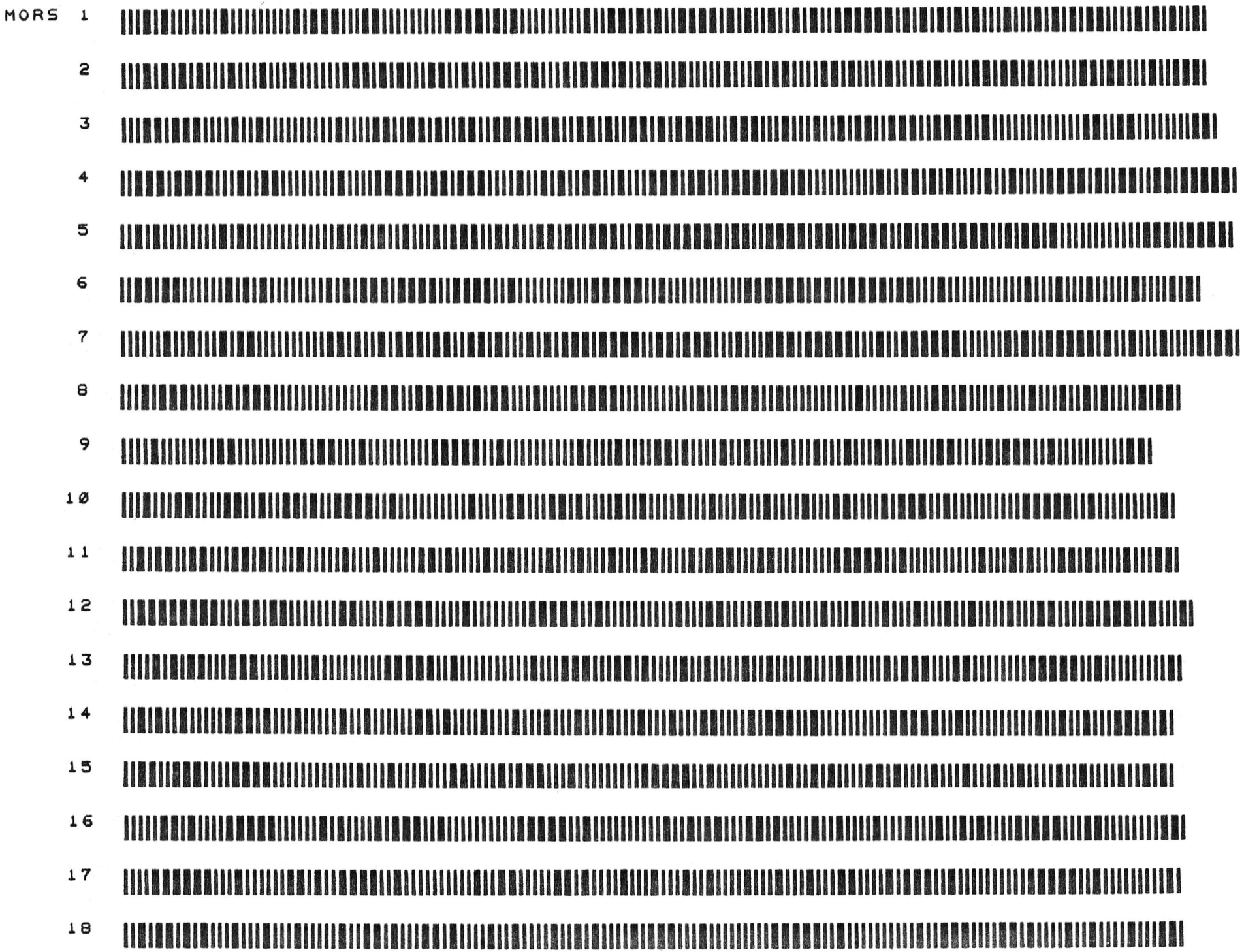
La durée du TONE 9 est:
534 boucles x 3 cycles
par boucles x 0,000158 s
par cycles = 0,253 secondes

Tous les tones d'une fréquence comprise entre deux tones existant peuvent être créés en variant le ratio entre l'alumage et l'extinction du bender. dansl'exemple de TONE 9, le bender est allumé pendant 3 cycles et étein- pendant 3cycles. essayez la même boucle avec des valeurs differentes (2c & 4c; ...). Aprés utilisation du bender,devez stocker ØØ en T, sinon un son quand le processeur tourne ou quand vous presserez une touche.

```
99B0 0B1 1
99B1 02B +
99B2 002 B
99B3 130   LDI
99B4 0FF
99B5 3D8   CST EX
99B6 2D8   FST EX
99B7 3E0   RTN
99B8 0B1 1
99B9 02D -
99BA 002 B
99BB 3C4   CLR ST
99BC 2D8   FST EX
99BD 3E0   RTN
```

RETENTIT

Ma HP41 ayant été gonflée avec deux blocs XFCT de 600 registres voici deux petites routines que j'utilise pour passer d'un bloc à l'autre. Ceux qui ont la transformation, ont l'habitude d'assigner : TONE 80 à la touche P -> R
et TONE 37 à la touche R -> P
Il est maintenant possible de remplacer TONE 80 par B-1
et TONE 37 par B+1


MORS
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18

19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36

WNDDTA

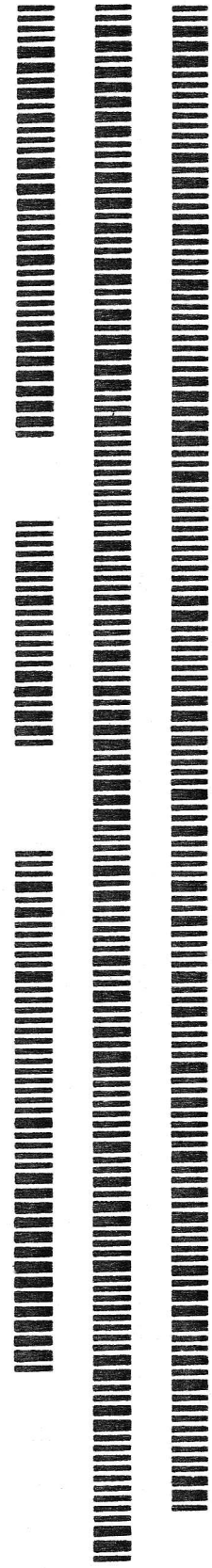
WNDDTX

WNDSUB

WNDSCN

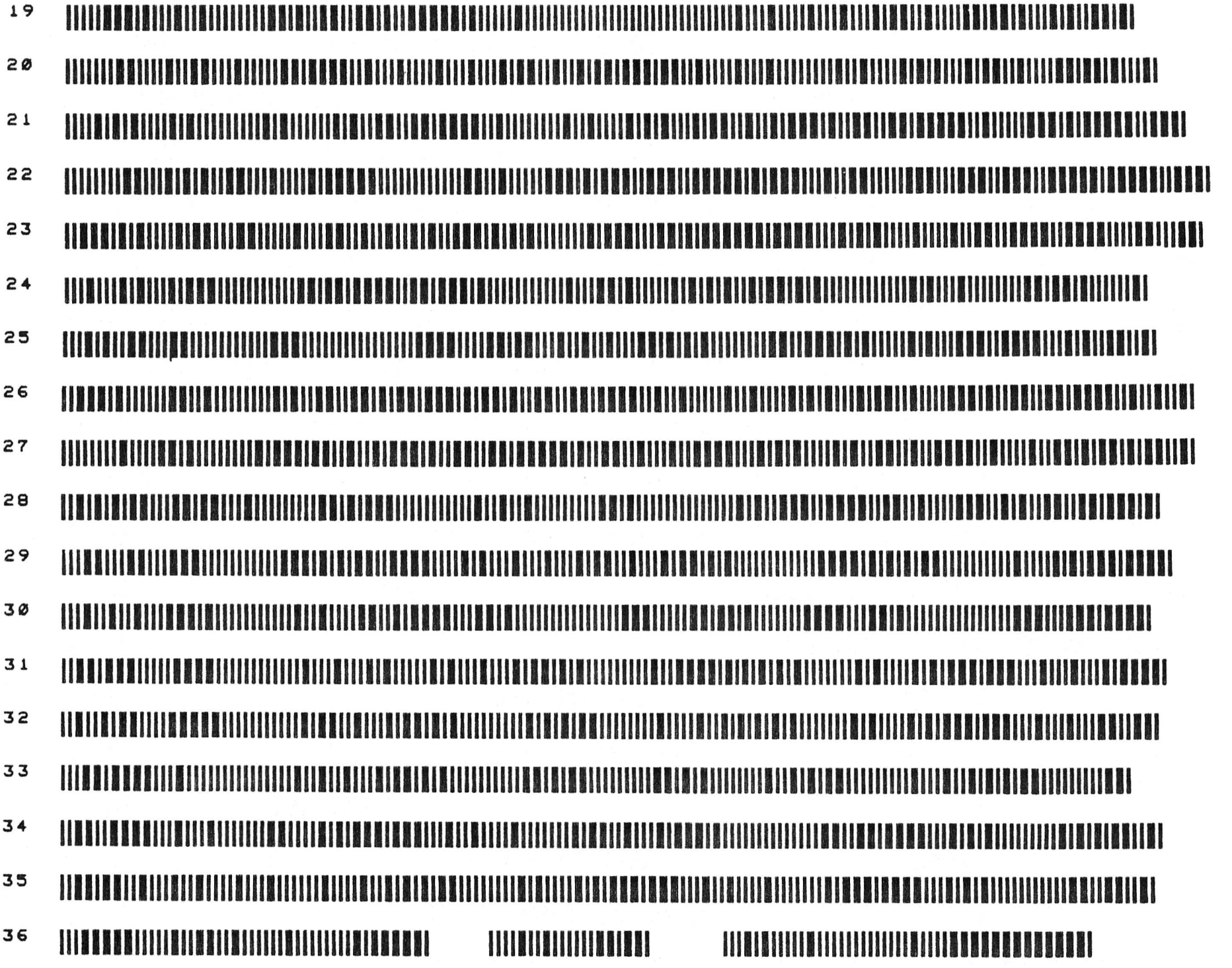
WNDLNK

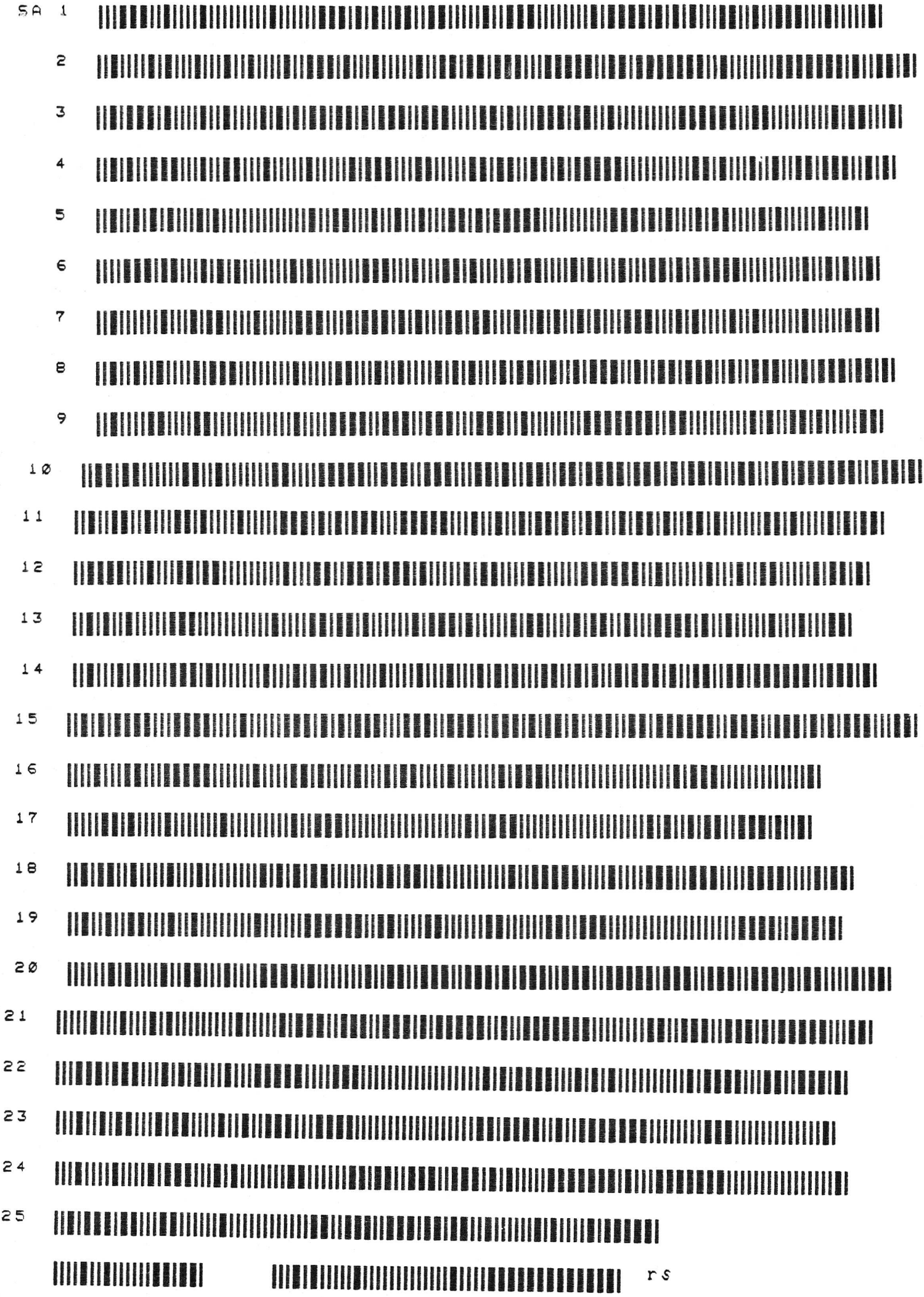
WNDTST



39 38 37

COMP 1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18


19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36


SA 1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
rs