

AOUT 1979

INT 95/79-2

PROGRAMME MONITEUR ETENDU POUR

MICROPROCESSEUR M6800

ADAPTE A L'UTILISATION D'UN TELETYPE TELEPRINT 390

(réalisé sur la base du programme Minibug révision 4
et utilisable à sa place : mêmes adresses ROM)

J.-M. Peiry

Centre de Recherches en Physique des Plasmas

ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE

INTRODUCTION

Ce programme a été écrit essentiellement afin de pouvoir conserver sur bande perforée et/ou sur papier les informations récoltées à l'aide d'un dispositif d'acquisition de données utilisé sur le Belt-Pinch. Aux endroits critiques (retour de ligne), le débit de caractères a été adapté aux capacités du Télétype Teleprint 390.

Par ailleurs, quelques options décrites ci-dessous permettent à l'opérateur un contrôle de l'état du microprocesseur (hardware et software) et facilitent le développement, la correction, la modification et l'exploitation des programmes.

Après la mise sous tension du télétype (position Duplex) et du microprocesseur, et l'initialisation du microprocesseur (Reset manuel), l'une des options décrites ci-dessous peut être sélectionnée en frappant la lettre correspondante sur le clavier du télétype.

Dans ce qui suit, nous utiliserons les conventions suivantes :

- 1) les caractères soulignés représentent ceux qui sont frappés par l'opérateur, les autres constituant le cas échéant la réponse du microprocesseur. Une attention particulière doit être portée aux espaces qui jouent un rôle important dans la sélection des options.
- 2) les lettres utilisées dans les formats représentent symboliquement des chiffres hexadécimaux. Les nombres ou les adresses relatives sont constitués par 2 chiffres hexadécimaux = 8 bits = 1 byte, alors que les adresses absolues en comportent 4 (2 bytes). L'adresse "A" sera écrite symboliquement AAAA ou si l'on veut insister sur les deux bytes qui la composent:
AHAL où
AH = "A high" = byte le plus significatif et
AL = "A low " = byte le moins significatif.

- 3) dans le listing qu'on trouvera en appendice, on trouvera des instructions du type :

STX DEB,UT

DEB et UT sont les noms symboliques de deux positions de mémoire successives (80B6 et 80B7). Si XHXL représente le contenu du registre d'index IX, l'instruction ci-dessus est équivalente à

XH → DEB

XL → UT

LISTE DES OPTIONS

Dans la liste ci-dessous, on trouvera en plus de la page où l'option recherchée est décrite, l'adresse ROM où le sous-programme correspondant commence. Le programme moniteur occupe 1K de mémoire (FC00 → FFFF), les positions FC00 → FC7F étant encore libres pour des adjonctions futures.

A (adresses) page 3 (FC80)

Calcul des adresses relatives ou absolues lors des instructions "branch".

B (belt) page 5 (FF75)

Initialisation et exécution d'un programme fixe commençant à l'adresse 2000.

C (copies) page 5 (FFA8)

Transfert du contenu d'une région de la mémoire dans une autre mémoire (RAM) non-protégée.

G (go) page 6 (FF01)

Ordre d'exécution d'un programme préalablement initialisé.

L (load) page 6 (FE27)

Lecture et mise en mémoire de nombres hexadécimaux, à partir d'une bande perforée en code ASCII ou à partir du clavier.

(Liste des options) suite

M (mémoire) page 8 (FF47 étiquette CHANGE)

Visualisation et modification éventuelle du contenu d'une adresse.

P (print/punch) page 9 (FD00 étiquette PUNCH 1)

Impression et stockage éventuel (bande perforée) d'une portion de la mémoire en vue d'une lecture subséquente.

R (registres) page 10 (FF60 étiquette PRINTS)

Contenu des registres du processeur central avant ou après l'exécution d'un programme.

S (store) page 12 (FF83)

Stockage du même nombre dans une portion de la mémoire RAM.

T (temporaire) page 13 (Adresse RAM 80C0)

Option réservée pour une adjonction temporaire au programme moniteur ou pour l'initialisation rapide d'un programme de l'utilisateur.

DESCRIPTION DES OPTIONS ET FORMATS

Option A (= adresses)

Ce sous-programme a deux fonctions :

- 1) Connaissant les adresses absolues de départ et d'arrivée d'une instruction "branch", le sous-programme calcule et imprime la différence positive ou négative (offset) qui doit suivre l'instruction "branch".

(Option A) suite

Format: A IHIL/FHFL (RH)RL

où: IHIL: adresse de départ (= adresse du 1er byte de l'instruction branch)

/ : n'importe quel caractère sauf espace

FHFL: adresse d'arrivée

(RH)RL: adresse relative (offset)

Si le saut peut être exécuté par le "branch", le résultat est affiché sous la forme RL (2 chiffres hexadécimaux).

Si le saut dépasse l'intervalle $\overline{-128}, \overline{127}$ (base 10) ou $\overline{80}, \overline{7F}$ (base 16 - "complément à deux") à partir de l'adresse suivant le "branch", le résultat est affiché sous la forme RHRL pour un saut en avant et sous la forme FRHRL pour un saut en arrière: l'instruction "branch" doit alors être remplacée par une instruction "jump".

Exemples :

A 1EA3/1F10 6B offset positif

A 1EA3/1F24 7F

A 1EA3/1E66 C1 offset négatif

A 1EA3/1E25 80

A 1EA3/1F25 0080 offset positif > 127₁₀

A 1EA3/DEA5 C000

A DEA5/1EA3 F3FFC offset négatif < -128₁₀

A 1EA3/1E24 FFF7F

- 2) Connaissant l'adresse absolue du début d'une instruction "branch", ainsi que l'offset qui la suit, le sous-programme calcule l'adresse absolue d'arrivée :

(Option A) suite

Format: A IHIL RL FHFL

IHIL : adresse absolue de départ

RL : offset 80 → FF négatif
 00 → 7F positif

FHFL : adresse absolue d'arrivée

Exemples :

A 1EA3 6B 1F10 offset positif

A 1E00 03 1E05

A 1EA3 C1 1E66 offset négatif

A 1E00 FD 1DFE

Option B (= belt)

Ce sous-programme initialise le microprocesseur pour un programme fixe commençant à l'adresse 2000 (20 → 80AE; 00 → 80AF) et en ordonne l'exécution (RTI) comme l'instruction G (=go). Pour les détails de l'initialisation, se référer aux options G, M et R.

Option C (= copies)

Ce sous-programme permet de copier séquentiellement le contenu d'une partie de la mémoire dans une région non-protégée de la mémoire RAM, qui doit être prévue de grandeur suffisante. Après chaque pas, les deux positions de mémoire contenant l'original et la copie sont comparées. En cas d'erreur de copie (mémoire protégée, essai de copier en ROM, mémoire défectueuse, etc.) l'exécution est arrêtée et un point d'interrogation apparaît. L'adresse de la copie où l'interruption s'est produite se trouve alors dans le registre d'index et peut être déterminée à l'aide de option R.

Format: C IHILFHFLCHCL

IHIL : adresse du début de l'original

FHFL : adresse de la fin de l'original +1

CHCL : adresse du début de la copie

(Option C) suite

Il faut faire attention de ne pas copier sur l'original avant qu'il ait été lu en entier. Ce cas se présente si: FHFL > CHCL > IHIL. Une copie intermédiaire est alors nécessaire.

Par ailleurs, si on copie des instructions, il est évident que les adresses des "jump" doivent être modifiées, ainsi que les offsets des "branch" qui produisent des sauts par-dessus la frontière de la section déplacée.

Option G (= go)

Cette option permet de commencer l'exécution d'un programme de l'utilisateur dont l'adresse de départ (DHDL) doit avoir été préalablement définie par la séquence suivante :

```
M 80AE  
*80AE XX DH  
*80AF XX DL  
G
```

où XX représente le contenu précédent (quelconque) des adresses 80AE et 80AF.

Pour plus de détails, se référer aux options M, R et au manuel de programmation (instruction Return from Interrupt RTI), notamment si d'autres registres que le compteur de pas de programme (PC) doivent être initialisés.

Option L (= load ou lecture)

Cette option permet de charger un programme ou des données à partir d'une bande perforée, ou éventuellement à partir du clavier en respectant le format suivant :

```
L S1XXYHYLAABBCC.....CS  
2 bytes      n bytes  1 byte  
S9030000FC
```

(Option L) suite

S1 : début de transmission
XX : nombre (hexadécimal) de bytes à transmettre + 3 (n + 3)
YHYL : adresse initiale, où sera chargé le 1er byte (AA);
BB sera chargé en YHYL +1, etc...
AA, BB, CC: bytes à transmettre
CS : "checksum", complément à un de la somme des n+3 bytes:
XXYHYLAABBCC.....
S9 : fin de transmission, arrêt du lecteur de bande (FC est
le complément à un de la somme des bytes: 03 + 00 + 00
suivant S9)

Pour lire une bande perforée, il suffit de l'introduire dans le
lecteur puis de taper un L.

Si une erreur de lecture intervient, la valeur lue de CS est
différente de celle qui est calculée par le microprocesseur pendant
le chargement. Cela provoque l'arrêt automatique de la lecture
et l'impression d'un point d'interrogation. Toutefois, les valeurs
lues sont enregistrées jusqu'à CS. Puisqu'il y a eu erreur, la
bande doit être relue, au moins depuis le dernier S1.

Si l'on charge manuellement à partir du clavier, la valeur de CS
n'est généralement pas connue. Il faut cependant attribuer une
valeur quelconque à CS de manière à terminer le chargement. Il est
clair que dans ce cas, le point d'interrogation n'implique pas une
erreur de lecture mais indique qu'on est sorti de la subroutine L.
Une nouvelle lecture doit donc recommencer par L S1... Par ailleurs,
la longueur d'un enregistrement est limitée par le nombre de caractères
(90) que comporte une ligne du télétype, des caractères tels que CR
(retour de chariot) ou LF (avance de ligne) ne devant pas être inter-
calés entre S1 et CS.

Option P (= print/punch)

Cette routine permet d'imprimer une partie de la mémoire (données ou instructions) et, en enclenchant la perforatrice du télétype, d'en tirer une bande perforée dont le format est identique à celui utilisé pour l'option L (load), et qui peut donc être relue à posteriori.

Le format d'impression ou de perforation est le suivant:

P XXXXYYYY

où XXXX est l'adresse initiale

et YYYY est l'adresse finale +1 (1ère mémoire non imprimée)

L'impression est découpée en lignes de 16 bytes, chaque byte étant représenté par deux caractères hexadécimaux.

Si les adresses XXXX et/ou YYYY ne se terminent pas par zéro, la première et/ou la dernière ligne ne sont pas complètes, comme le montre l'exemple suivant (où les nombres 00 à 26 ont été préalablement chargés dans les mémoires 0016 à 003C)

P 0016003D

S10D001600010203040506070809AF

S11300200A0B0C0D0E0F10111213141516171819B4

S11000301A1B1C1D1E1F202122232425261F

S9030000FC

Le dernier byte de chaque ligne ne représente pas le contenu d'une mémoire, mais c'est la valeur "checksum" calculée par la machine, qui est telle que la somme de tous les bytes qui suivent le signal S1 (début d'enregistrement) soit égale à FF (seuls les 2 derniers digits hexadécimaux étant pris en considération).

(Option P) suite

La dernière ligne, commençant par S9 (fin d'enregistrement), sert notamment à arrêter le lecteur de bande et, lors d'une lecture subséquente, à sortir de la subroutine L. (On voit que $03+00+00+FC=FF$, ce qui constitue un moyen mnémotechnique du calcul de "checksum").

Remarquons finalement que si on enclenche la perforatrice avant de donner l'instruction P XXXXYYYY (ce qui est prudent car l'impression commence immédiatement après le dernier caractère de l'adresse YYYY), cette instruction est aussi perforée. Cela n'a pas d'importance puisque la subroutine L ne tient compte que des enregistrements commençant par S1. En outre, cela permet une identification facile de la bande à la lecture.

Option R (= registres)

En tapant le caractère R, on produit l'impression suivante:

R CC AB AA XHXL PHPL SHSL

où chaque lettre suivant le R représente la place d'un chiffre hexadécimal. La même impression (sans le R) est produite automatiquement à la suite de l'interruption (software interrupt SWI) d'un programme de l'utilisateur et indique le retour au programme moniteur. Les paramètres imprimés représentent alors l'état du processeur central (CPU) juste avant l'interruption où :

- CC : registre des codes de condition (condition code register)
- AB : accumulateur B (1 byte)
- AA : accumulateur A (1 byte)
- XHXL : registre d'index (2bytes)
1er byte X - "high"
2^e byte X - "low"
- PHPL : compteur de pas de programme
(program counter PC) (2 bytes)
- SHSL : pointeur de stack +1 (stack pointer SP) (2 bytes)

(Option R) suite

Une remarque sur la structure interne du programme moniteur s'impose ici. En plus des adresses ROM (FC80 → FFFF), le programme moniteur dispose pour ses variables internes d'une petite mémoire RAM (8080 → 80FF) de 128 bytes. C'est à certaines de ces positions de mémoire que sont placées les diverses adresses intervenant dans les options décrites ici. Un tableau d'occupation de cette mémoire se trouve en appendice.

Il suffit de savoir pour l'instant que lors du "Reset" manuel qui doit suivre la mise sous tension du microprocesseur, le "stack pointer" est défini à l'adresse 80A8. Il s'en suit que si l'on désire présélectionner le contenu des divers registres du processeur central avant le début de l'exécution, il faut introduire ces valeurs aux adresses 80A9 à 80AF (à l'aide de l'option L ou M) selon le schéma suivant:

(CC) → 80A9
(AB) → 80AA
(AA) → 80AB
(XH) → 80AC
(XL) → 80AD
(PH) → 80AE
(PL) → 80AF

On se souvient qu'avant l'utiliser l'option G, le compteur de pas de programme doit être initialisé en plaçant le byte le plus significatif de cette adresse en 80AE et le moins significatif en 80AF. Lors du début d'exécution (voir instruction RTI du manuel de programmation), les divers registres du processeur central sont chargés successivement du contenu des adresses suivant celle du "stack pointer" (80A8).

Remarquons encore que l'instruction SWI ne "sauve" pas la valeur du "stack pointer" (SP). La subroutine R (appelée soit manuellement soit automatiquement) imprime également le contenu (+1) de ce registre

(SP) + 1 = 80A9, et place les deux bytes en mémoire :
(SH) = 80 → 80B0
(SL) = A9 → 80B1

(Option R) suite

Il est clair que toutes ces valeurs aboutiront ailleurs, si le programme de l'utilisateur définit un autre stack à une adresse \neq 80A8. L'impression automatique du contenu des registres permettra de voir où l'on a abouti, mais un appel subséquent de l'option R redonnera le contenu des adresses 80A9 \rightarrow 80B1 car le stack est redéfini à chaque passage à l'étiquette CONTRL (FEDD).

Option S (= store)

Cette option donne la possibilité de charger rapidement une partie de la mémoire RAM avec le même byte.

Le format est le suivant:

S XXXXYYYZZ

où XXXX : adresse initiale

YYYY : adresse finale +1

ZZ : byte à stocker entre XXXX et YYYY-1.

Si on essaye de modifier le contenu d'une adresse ROM ou d'une adresse RAM protégée, un point d'interrogation apparaît et l'exécution est arrêtée. Il faut remarquer que le stockage commence à l'adresse la plus élevée YYYY-1 et se termine en XXXX.

Ce programme peut être utilisé pour tester l'état de mémoires RAM : en chargeant successivement les bytes 00 puis FF dans une portion de mémoire, on vérifie que tous ses bits peuvent prendre leurs deux états*. Si une cellule de mémoire est défectueuse, le point d'interrogation apparaît. L'adresse du défaut est alors dans le registre d'index et peut être trouvée à l'aide de l'option R par exemple.

* (Il arrive cependant que des défauts dynamiques, où un bit rechange de valeur après un bref délai, passent inaperçus.)

Option T (= temporaire)

Cette routine permet d'ajouter temporairement une option au programme moniteur ou d'initialiser de manière commode un programme de l'utilisateur souvent utilisé. L'effet de cette routine est simplement de faire sauter le programme moniteur à l'adresse RAM 80C0.

Les 64 positions de mémoire 80C0 → 80FF peuvent alors être intégrées au programme moniteur, auquel cas cette suite d'instructions doit se terminer par une instruction de retour du type JMP CONTRL (7E FE DD) et non par un RTS (39). Il est aussi possible de sauter à partir de l'adresse 80C0 ou d'une des adresses suivantes à un endroit quelconque (UUVV) d'un programme de l'utilisateur par la séquence:

CE UUVV	LDX# \$UUVV	UUVV → IX
FF 80AE	STX 80AE	UU → 80AE
3B	RTI	VV → 80AF
		EXECUTION

ou plus simplement par l'instruction:

7E UUVV	JMP UUVV
---------	----------

Toutefois, dans ce dernier cas, les registres du processeur central (sauf le PC) gardent les valeurs qu'ils avaient au moment du saut en 80C0. De ce fait, si l'option T se termine par un SWI sans que le stack ait été modifié, les registres du processeur seront stockés de 80A2 à 80AA comme suit.

(CC)	→ 80A2
(ACCB)	→ 80A3
(ACCA)	→ 80A4
(XH)	→ 80A5
(XL)	→ 80A6
(PH)	→ 80A7
(PL)	→ 80A8

(Option T) suite

(SH)=80 → 80A9

(SL)=A1+1=A2 → 80AA

On voit donc que le contenu précédent des 2 dernières adresses est détruit, ce dont il faut éventuellement tenir compte si l'on appelle ensuite l'une des options B, G ou R.