
Transposition des programmes de TI 57 sur TI 58 et 59

Voici les quelques règles qui vous permettront d'adapter facilement les programmes écrits pour la petite machine.

■ Il se peut qu'après avoir découvert l'informatique de poche sur une TI 57, vous soyez passé à un modèle supérieur du même constructeur : TI 58 ou 59. Il se peut aussi que vous utilisiez l'une de ces deux dernières machines sans connaître la TI 57. Dans un cas comme dans l'autre vous avez certainement des difficultés pour utiliser sur votre TI 58 ou 59 les programmes écrits pour TI 57. Voici qui devrait vous faciliter la tâche.



Les indications qui suivent ne constituent en rien la meilleure façon d'adapter un programme. En fait, elles sont très gourmandes en pas de programme et elles n'utilisent pas au mieux les possibilités des TI 58 et 59. Leur intérêt est autre : il s'agit de transposer **rapidement** la liste d'un programme de TI 57 en appliquant seulement quelques règles faciles à retenir. Cela permettra par ailleurs de faire apparaître les

caractéristiques respectives de ces différents matériels.

La touche 2nd ayant le même rôle sur les deux machines (nous entendons par là sur la TI 57 d'une part, et, d'autre part, la TI 58/59 considérée comme une seule machine), nous ne la mentionnerons pas dans cette étude. Une précision importante cependant : la TI 57 admet que l'on écrive indifféremment 2nd Inv ou Inv 2nd sans risque d'erreur, mais la TI 58/59 ne tolère qu'Inv 2nd : une habitude à prendre.

La TI 57 utilise dix étiquettes numérotées de 0 à 9. Le plus simple est de leur faire correspondre sur la 58/59 les étiquettes A à E' selon la convention suivante :

TI 57	Lbl 1	Lbl 2	Lbl 3	Lbl 4	Lbl 5
TI 58/59	Lbl A	Lbl B	Lbl C	Lbl D	Lbl E

TI 57	Lbl 6	Lbl 7	Lbl 8	Lbl 9	Lbl 10
TI 58/59	Lbl A'	Lbl B'	Lbl C'	Lbl D'	Lbl E'

Le démarrage d'un programme par 5 SBR 2 (TI 57) devient ainsi 5B sur 58/59 : inutile en effet de taper SBR.

Parmi les huit mémoires indifférenciées de la 57, numérotées de 0 à 7, deux jouent en plus un rôle spécifique, le registre 0, mémoire de décrémentation et le registre 7, mémoire de test. Pour conserver les possibilités de calcul dans les mémoires, d'échange, etc., nous garderons la correspondance sur 58/59. Cela n'est pas gênant pour la mémoire 0, mais on devra se livrer à une petite gymnastique concernant la mémoire 7. On veillera par ailleurs, même si ce n'est pas toujours indispensable, à frapper 00 pour 0, 01 pour 1, 02 pour 2, ... et 07 pour 7 ; STO 3 devient donc STO 03 et Exc 3 devient Exc 03.

Sur la TI 57, l'instruction Dsz décrémente la mémoire 0 et effectue un test sur son contenu. Nous garderons donc la mémoire 0 pour les décréments en précisant simplement Dsz 00.

Le GTO traditionnel des TI 57 (renvoi à une étiquette numérique) disparaît au profit des étiquettes A, B, C, ..., D', E' : il suffit d'inscrire le nom de l'étiquette où doit s'effectuer le renvoi.

TI 57	GTO 2	Dsz GTO 6	x=t GTO 9
58/59	B	Dsz 00 A'	x=t D'

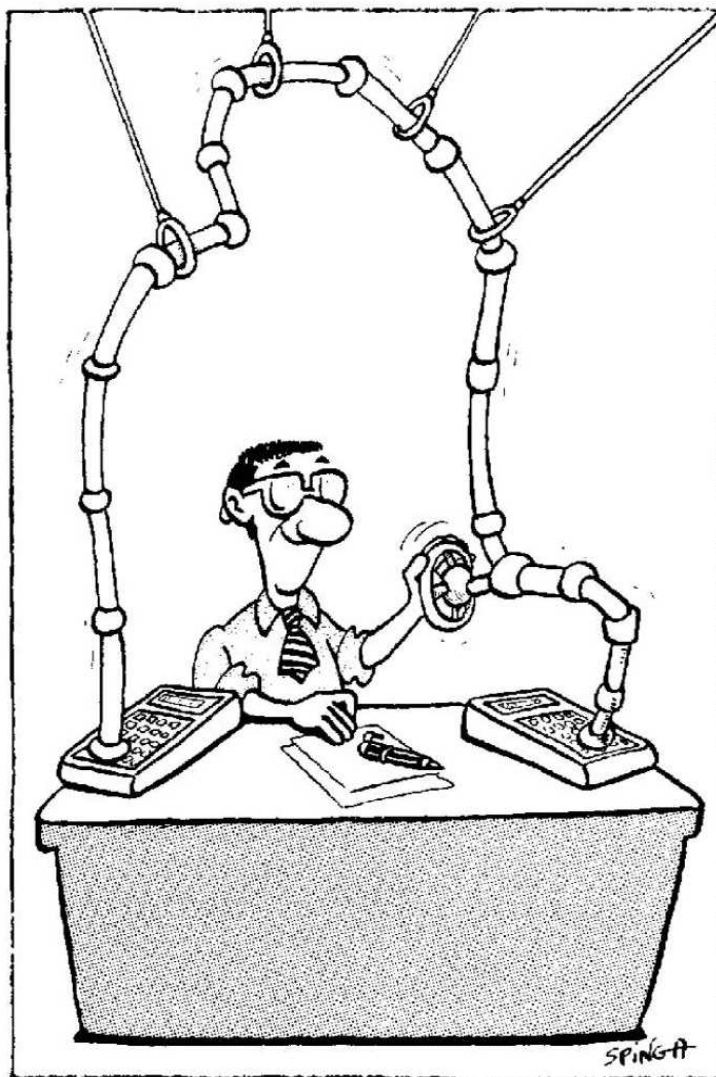
Un cas particulier : l'instruction RST qui, sur 58/59, ne peut pas être utilisée après un test. On devra donc utiliser un renvoi direct au pas de programme n° 000.

TI 57	STO 2 RST	Dsz RST	x=t RST
58/59	STO 02 RST	Dsz 00 000	x=t 000

Comme on l'a vu, sur la TI 57, la mémoire de test est confondue avec la mémoire 7, elle joue le rôle d'une mémoire indifférenciée. Pour con-

server cette indifférenciation sur les 58/59, nous utiliserons aussi la mémoire 07, mais cela nous obligera à prendre deux précautions concernant les tests et l'effacement du registre 07.

L'instruction C.t. de la 57 se traduira par Exc 07 0 Exc 07, et tout test devra systématiquement être précédé d'un transfert entre la mémoire 07 et le registre indépendant t sans modification de l'affichage ; de deux pas de programme, on passe à sept : RCL 1 x = t (TI 57)



devient RCL 01 x = t RCL 07 x = t x = t (58 et 59).

L'échange entre l'affichage et le registre de test (x = t) s'écrira Exc 07. Si cet échange a lieu juste avant un test, il est possible de simplifier. La séquence x = t x = t des 57 s'écrira STO 07 x = t x = t sur 58/59.

Nous en arrivons maintenant à la partie la plus délicate, à savoir le fait qu'après un test, les TI 58/59 n'acceptent que des instructions de renvoi, alors que les codes combinés de la TI 57 autorisent d'autres opérations.

Si l'instruction TI 57 qui suit le Dsz ou le test est un renvoi inconditionnel ou un appel de sous-programme : pas de problème.

TI 57	Dsz GTO 7	x ≥ t GTO 3	Inv x ≥ t SBR 4
58/59	Dsz 00 B'	x ≥ t C	Inv x ≥ t D

Par contre, si Dsz ou si le test sont suivis d'une autre indication (STO n, x², etc.), il faut obligatoirement utiliser une touche étiquette renvoyant inconditionnellement à un programme situé en fin de mémoire, programme qui lui-même devra renvoyer à l'adresse suivant celle de la touche étiquette : on est obligé de faire un détour... Ainsi Dsz STO 1 et x = t • 1 sur TI 57 deviennent respectivement sur 58/59 : Dsz 00 x² Lbl √x (...) Lbl x² STO 01 GTO √x d'une part et x = t RCL Lbl SUM 1 (...) Lbl RCL • GTO SUM d'autre part.

On aura bien évidemment tout intérêt à utiliser des étiquettes faciles à associer (x² et √x, sin et cos, etc.).

Quelques autres petits problèmes : l'instruction Pause est en général plus courte sur la TI 58/59 ; il conviendra donc de la répéter deux ou trois fois pour obtenir une durée d'affichage équivalente. A noter aussi que l'instruction n Inv Log qui, sur la TI 57, permet d'obtenir facilement des puissances de 10, risque, en raison d'erreurs d'arrondi, de donner des résultats aberrants sur TI 58/59 ; il faudra faire :

TI 57	n Inv Log
TI 58/59	n Inv Log EE Inv EE

Enfin, il sera sans doute préférable de se servir du générateur de nombres aléatoires inclus dans le module de base de la TI 58/59, plutôt que de recopier les générateurs souvent très simplifiés (et donc de moindre qualité) utilisés sur la TI 57.

Voilà... Bien évidemment, cet article ne prétend pas épuiser le sujet, et la méthode indiquée est assez lourde, mais elle convient pour la plupart des programmes, et elle permet d'effectuer des transpositions d'une façon presque automatique.

□ Bernard Elman