



Ah ! si vous aviez su...

Vous ne connaissez pas votre machine à fond, et moins encore les autres machines... Ces quelques "ficelles" vous montreront comment on peut toujours en tirer un peu plus.

Quelques opérations « spéciales » (TI 58 et 59)

■ Tout utilisateur d'une TI 58 ou d'une TI 59 sait que son micropoche est équipé de 40 fonctions spéciales (41 pour la 58 C) accessibles grâce à la touche 2nd Op. Parmi ces fonctions, il s'en trouve deux, Op 18 et Op 19, qui ne retiennent pas toujours l'attention des débutants et qui sont pourtant d'un grand intérêt.

Ces deux opérations font office de détecteurs d'erreurs et mettent en œuvre le drapeau n°7 :

- 2nd Op 18 lève le drapeau 7 si aucune condition d'erreur ne s'est produite dans le déroulement d'un programme ;
- 2nd Op 19 lève le drapeau 7 dès qu'une condition d'erreur survient.

On peut ainsi obtenir facilement l'équivalent des ON ERROR GOTO

que l'on trouve dans certains Basics. Pour utiliser l'une de ces deux instructions, il suffit de la placer dans le programme immédiatement après la séquence susceptible de provoquer une condition d'erreur, de faire suivre le tout par une interrogation sur l'état du drapeau 7 et de prévoir l'instruction de transfert à l'adresse ou à l'étiquette appropriée.

Prenons un exemple : l'utilisateur veut détecter lors de l'étude des tangentes de nombres quelconques s'il n'existe pas de solutions infinies (et par conséquent indéfinies).

...		
510	95	=
511	30	2nd tan
512	69	2nd Op
513	19	19
514	87	2nd lf flg
515	07	07
516	25	CLR
...		
620	76	2nd Lbl
621	25	CLR
etc.		

Le test utilisé ici porte sur le drapeau 7 que l'opération 19 aura levé en cas d'erreur. Si c'est le cas, le programme se poursuivra au label CLR (rappelons qu'une tangente est indéfinie pour un angle de $\pi/2 + K\pi$ radians, k étant un entier).

Notons d'ailleurs que l'instruction Op 18 aurait pu être utilisée, et dans ce cas le transfert au label CLR ne

se serait produit que si la tangente avait été définie. Avec tout ceci, auquel vient s'ajouter l'instruction INV 2nd lf flg, nous pouvons obtenir une grande souplesse dans le traitement des erreurs sur ce micropoche.

Toujours au chapitre de la chasse aux bogues, le drapeau 8 est lui aussi bien utile, surtout lors de la mise au point des programmes. Rappelons que ce drapeau, s'il est levé, interrompt systématiquement le déroulement du programme à l'endroit où l'erreur a été détectée. S'il est baissé, la machine poursuit malgré les erreurs rencontrées en cours d'exécution, et seul le clignotement de l'affichage en fin de programme peut nous signaler qu'une ou plusieurs erreurs sont survenues. Encore faut-il savoir que, même en cours de programme, CE ou CLR restitue un affichage « normal ».

On a donc intérêt à laisser toujours deux pas libres au début de chaque séquence suspecte pour pouvoir y insérer, le moment venu, 2nd St flg 8. Par la suite, une fois certain que le programme est bien au point, on supprimera simplement ces pas.

Puisque nous en sommes à parler des drapeaux, il n'est pas inutile de rappeler le triple rôle de RST. Non seulement cette instruction renvoie le pointeur au pas 000 et elle efface les mémoires d'adresses de retour des sous-programmes, mais elle rabaisse également tous les drapeaux. Ainsi, lorsque tous les drapeaux levés doivent être abaissés au

Ah ! si vous aviez su...

même moment, on dispose de trois solutions.

La première (à déconseiller) consiste à placer une série d'instructions INV St flg n, mais c'est très lourd. On peut essayer l'annulation par adressage indirect (INV St flg IND n), mais ce n'est pas forcément économique en pas. La meilleure solution consiste encore à utiliser RST comme suit :

000	61	GTO
001	01	01
002	25	25
...		
124	81	RST
125	...	

On consomme alors quatre pas de programme ; autrement dit l'opération devient rentable par rapport à la première solution dès que l'on a plus de deux drapeaux à abaisser. Toutefois, RST abaissant *tous* les drapeaux et effaçant les adresses de retour des sous-programmes, on ne doit l'utiliser qu'à bon escient.

Autre opération spéciale, parfois très commode et que beaucoup d'utilisateurs ignorent : 2nd Op 10 retourne à l'affichage le signe d'un nombre x de la même façon que la fonction SGN du Basic :

- 1 si le nombre x est plus grand que 0 ;
- 0 si x est nul ;
- -1 si x est un nombre négatif.

Application possible : multiplication d'un nombre par le signe d'un autre sans pour autant passer par des tests :

...		
015	43	RCL
016	05	05
017	65	x
018	43	RCL
019	06	06
020	69	2nd Op
021	10	10
022	95	=
...		

Ici le résultat final du pas 22 sera égal à la valeur du registre 05 mais affecté d'un signe qui est fonction du signe des valeurs contenues dans

les mémoires 05 et 06 (+ et +, - et - donnent + ; + et -, - et + donnent - ; enfin on obtient 0 si l'une au moins des deux valeurs est nulle).

Pour terminer, je rappellerai une petite astuce que certains d'entre vous ne pratiquent peut-être pas encore. Elle intéresse les personnes à qui la T1 58 ou 59 a tellement rendu de services que le clavier a perdu de ses qualités. Ou bien certaines touches ne répondent plus toujours du premier coup, ou bien elles répondent deux fois à une seule et unique pression (on parle de « rebonds »).

Généralement ce sont les touches numériques qui, pour avoir servi beaucoup plus que les autres, sont affectées par cette maladie. On peut évidemment faire installer un clavier neuf, c'est un remède radical. Mais on peut aussi s'en accommoder tant bien que mal en rusant, lorsque c'est possible, avec le code des touches.

Un exemple : en mode programme, STO 34 est tapé STO, 3 puis 4 ; mais STO \sqrt{x} (code 34) revient exactement au même. L'avantage est double : on presse sur deux touches au lieu de trois et l'on élimine le risque du rebond. De la même façon SBR 232 peut être tapé SBR 2, $x \approx t$, et 2nd If flg 2nd Ind 12 554 devient 2nd If flg 2nd Ind B 5 EE (7 touches au lieu de 9), etc.

Pour entrer un double zéro, on utilisera dans la plupart des cas la touche SST : RCL SST équivaut à RCL 00 et GTO SST A s'inscrit GTO 011.

Même si certains codes ne peuvent être introduits que par l'intermédiaire des touches numériques, au total, cela fait tout de même plus de 60 combinaisons de deux chiffres qui sont ainsi disponibles : on prend très vite l'habitude de les utiliser. En diminuant le nombre de pressions de touches, on diminue les risques d'erreurs et l'on gagne un peu de temps. Il faut bien sûr que les touches autres que numériques soient restées en assez bon état, mais c'est généralement le cas.

Rien n'empêche d'ailleurs d'utiliser cette méthode pour entrer un programme dans une machine neuve. Petit avantage accessoire : cela oblige à apprendre par cœur le code des différentes instructions...

□ Jean-Charles Lemasson