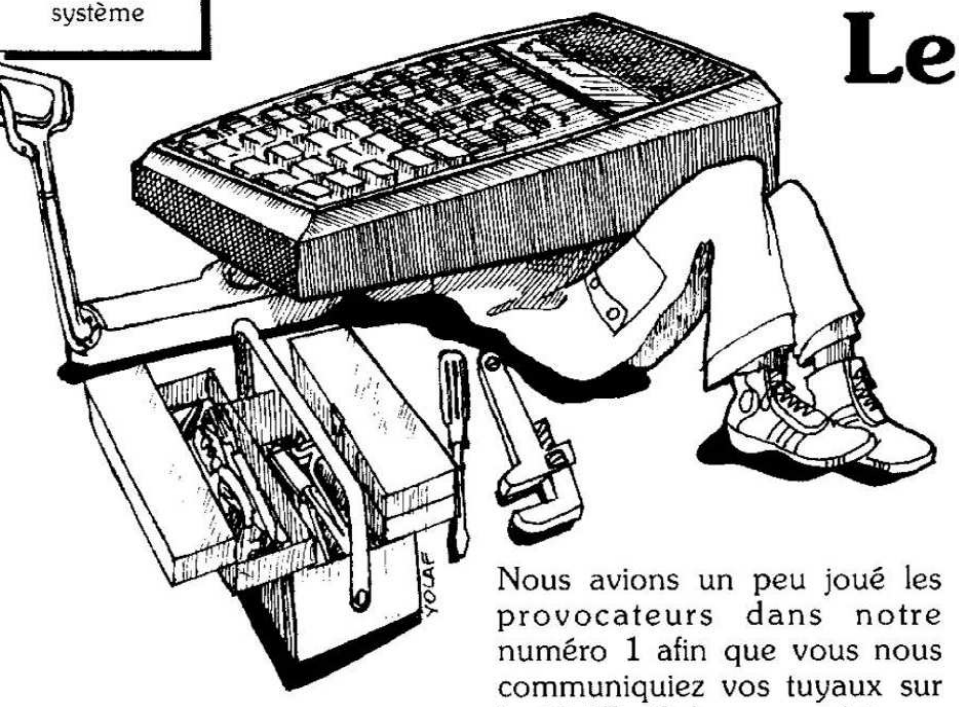
Les dessous de la TI-57



Nous avions un peu joué les provocateurs dans notre numéro 1 afin que vous nous communiquiez vos tuyaux sur la TI 57. Cela a marché au-delà de toute espérance, aussi voilà aujourd'hui quelques-unes des trouvailles que nous avons reçues. Puisse le dieu des calculatrices vous aider à partir de ces trouvailles à en découvrir d'autres !

l'Op

A ne pas faire

■ Vous devez sans doute déjà tous connaître la façon de bloquer votre machine avec la séquence
1 LE 99 +/− × 0 =.
Vous pouvez également parvenir au même résultat avec
1 EE 99 +/− Σ+.
Le seul moyen de vous en sortir sera d'éteindre votre machine. En fait, seul l'affichage est éteint par cette séquence, et à partir de là on peut envisager un petit bricolage donnant presque à la TI 57 une mémoire continue !

□ JBC

Donnez des lettres à votre TI 57, et autres choses étranges

■ Si l'on utilise SST derrière Exc, on obtient des résultats plutôt surprenants. Tapez LRN (affichage 00 00)/ 2nd Exc (00 38 0)/ SST (01 00)/ Lbl 1 (02 00)/ 3 (03 00)/ R/S (04 00)/ RST (04 71). Faites maintenant RST/ R/S/ R/S/ LRN.

Tapez un chiffre de 0 à 9, et admirez le résultat après un BST. Si vous avez tapé 0, vous voyez maintenant apparaître au pas 04 de notre exemple le code 8 ; 1 vous fournit 9 ; et surtout, 2, 3, 4, 5, 6, 7 donnent respectivement **A, b, C, d, E** et **F.** Voilà qui ressemble furieusement à un début d'alphabet hexadécimal ! Mais ce n'est pas là le plus important. Cette manipulation permet d'obtenir comme code d'instruction les codes 11 (2nd) et 12 (Inv) qui ne sont pas accessibles normalement.

Ces codes et ces lettres peuvent être mis dans un programme avec des INS et des DEL, à partir du préfixe qui *doit* être placé au pas 00, car seule l'instruction RST conserve le numéro d'ordre qui apparaît à la droite de l'affichage.

La touche RST du pas 04 peut être remplacée par n'importe quelle touche, sauf celles appelant un numéro d'ordre, telles que GTO, SBR, STO, SUM, RCL, Fix ; mais le RST doit être tapé après chaque R/S, sauf si l'on tape le programme « en double » à partir d'un pas quelconque. On a ainsi par exemple :

Exc	38	
LBL 1	86	1
3	03	
Exc	38	
LBL 1	86	1
3	03	
R/S	81	

Le code 38 (Exc) peut être remplacé par le code 48 (Fix), 39 (Prd) ou −39 (Inv Prd). Les lettres obtenues sont très « fragiles », car toute tentative de mise en mémoire ou d'utilisation dans un sous-programme les transforme en nombre : **A** devient 0, **b** devient 1, **C** devient 2, etc. Les lettres ne semblent en résister qu'à une Pause ou à un R/S, toute tentative de reprise de l'exécution du programme les « numérise ».

En fait, je dois l'avouer, je n'ai pas réussi ainsi à obtenir et conserver les « vrais » codes 11 et 12... aussi y suis-je parvenu d'une autre façon.

Exc	00	38
LBL 0	01	86 0
. ou +/−	02	83 ou 84
R/S	03	81

Si maintenant on fait R/S puis LRN, puis 0, on va obtenir le code 11 (2nd) au pas 03. Au lieu de 0, on peut taper n'importe quel chiffre de 1 à 7, et l'on obtient tous les codes de la première colonne : 1 donne 21 (LRN), 2 donne 31 (SST), 3 fournit 41 (BST), 4 fournit 51 (GTO), etc (5 : SBR ; 6 : RST ; 7 : R/S).

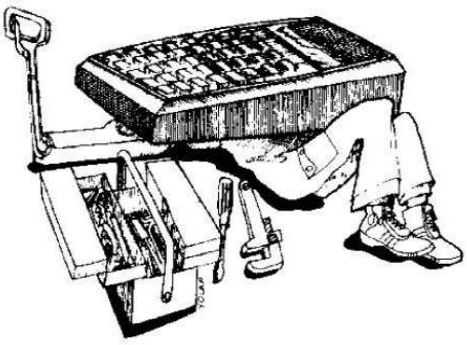
Le code 11 possède une propriété particulière : si on l'exécute, il fait apparaître un affichage semblable à celui du mode programmation, alors que la machine est en mode exécution. Si par exemple avant cette exécution l'affichage était 98 920, il va devenir 92 11 ; le soi-disant « numéro de pas » est toujours formé à partir des chiffres des centaines et des dizaines de l'affichage.

Mais je vais quand même garder quelques autres découvertes pour le prochain numéro, afin de vous laisser quelque chose à chercher et à trouver !

□ Yves Roque

Combinaison de codes

■ Les instructions suivies d'une adresse ou d'un numéro de mémoire se prêtent à des manœuvres diver-



ses. Rentrez par exemple LRN/ GTO/ STO/ 4/ BST. Vous obtiendrez Fix 4 ! Il y a en fait un groupe d'instruction « fortes » (GTO, SBR, Fix, Lbl) et un groupe « faible » (STO, SUM, Prd, Exc, RCL).

Si vous faites suivre une instruction « faible » par une forte ou une faible, c'est le dernier code tapé qui l'emporte. De même si vous faites suivre une « forte » d'une autre forte. Mais si vous faites suivre une forte par une faible, vous obtenez comme résultat une forte !

Ainsi forte suivie de STO donne Fix, tout comme forte suivie de Prd. SUM donne GTO, Exc donne Lbl et RCL donne SBR !

Plus étrange encore est la séquence LRN/ STO/ Del.

Votre affichage manque alors 00 00 0. Appuyez maintenant sur n'importe quelle touche sauf un chiffre, et votre programme contiendra alors cette touche.

Mais si vous appuyez sur n'importe quel chiffre, par exemple 2, puis sur BST, vous allez voir que le STO que vous espériez avoir détruit s'est en fait transformé en l'instruction STO 2 !

De même, LRN/ GTO/ Del/ STO/ Del/ 4/ BST vous fournira Fix 4 !

[] Jérôme Lacaille

Appui sur deux touches

■ Si l'on appuie simultanément sur deux touches de gauche situées sur une même ligne, on obtient le code opération de la touche qui est à leur droite. Exemple : x^2 et $\sqrt{x}$ donnent 1/x SST et STO donnent RCL, etc.

[] Frédéric-Julien Brunhes

Et avec SST ?

■ J'avais déjà remarqué qu'en mode programmation (tout ce qui suit se passe dans ce mode), il est possible de séparer artificiellement une instruction mémoire (STO, RCL) de son opérande, c'est-à-dire du numéro de la mémoire visée par l'opération.

Toute opération en mémoire, tout saut inconditionnel est stocké dans le même pas que son opérande. J'ai un moment espéré qu'en séparant le code opération de l'opérande, j'aurais accès à un adressage indirect, mais ce n'est hélas pas le cas. Voyons toutefois comment procéder à cette séparation, et ce qu'elle permet finalement.

Mettez-vous en LRN, et tapez STO au pas 00 ; votre affichage indique 00 32 0, la machine se mettant en attente du numéro de la mémoire visée.

Soyons sournois, et décevons son attente en appuyant sur SST. Votre machine, un instant surprise, ne perd pas pour autant son sang froid et affiche 01 00. Rentrons maintenant le numéro de la mémoire visée par STO, par exemple 3. L'affichage marque tout naturellement 02 00. Appuyons sur R/S pour clôturer le programme que nous venons de créer, revenons au début et lisons notre programme. Il apparaît sous la forme 00 32, 01 03, 02 81 ; ceci correspond à

00	STO		00	STO 3
01	3	et non à	01	R/S
02	R/S			

Ce programme fonctionne très bien, comme vous pouvez le vérifier : tapez INV c.t./ RST/ 456/ R/S/ CLR/ RCL 3, l'affichage indique bien 456.

Après cette petite mise en train d'exploration, passons à d'autres résultats. J'ai d'abord essayé SBR/ SST/ 0, m'attendant à un saut au sous-programme LBL 0. Cette fois, c'est mon attente qui a été déçue : j'obtenais à chaque fois des résultats déconcertants, parfois des clignotements, parfois des sauts au pas 01, parfois aucun effet apparent.

Et la clé du mystère apparut au milieu d'une table de résultats. Le pas 00 est la clé de tout ce système car selon son contenu, les effets de SBR/ SST/ x (x = 0, 1, 2... ou 9) sont différents.

. Si le pas 00 contient 0, SBR SST x effectue un appel au sous-programme commençant en 01 *si l'affichage est compris entre 0 et 9.* Si l'affichage contient x + 1 chiffres, aucun effet visible ; si l'affichage ne contient aucune des valeurs ci-dessus, le programme s'arrête en clignotant.

. Si le pas 00 contient autre chose que 0, le pas suivant SBR SST x est sauté si l'affichage est l'un des chiffres entre 0 et 9 ; les autres effets sont identiques.

Et que se passe-t-il si au lieu de taper un chiffre on tape une autre instruction zzz ? Et bien, si le pas 0 contient 00, et que l'affichage contient x + 1 chiffres, l'exécution se poursuivra justement au pas suivant le SBR, c'est-à-dire sur l'instruction tapée derrière SBR 2/ SBR 4.

J'allais oublier un petit détail qui a son importance : l'exécution de tout SBR SST fait monter la pile de retour des sous-programmes, tandis que INV SBR fonctionne « normalement » et renvoie au premier pas suivant le SBR ou SBR SST rencontré. Si vous passez 3 fois sous une de ces fonctions SBR SST sans avoir fait entre temps un INV SBR ou RST pour vider la pile de retour, vous aurez très normalement des ennuis.

[] Christophe Théron

C'est là une collection bien riche de « trucs » et de pistes de recherche. A vos claviers pour les mettre en œuvre et trouver des nouveautés, et n'oubliez pas de nous les communiquer : avec des astuces, on peut réaliser des tas de choses, même en 50 pas !

l'Op