



Sujet favori depuis le début de la rubrique : HIR. Cette fonction tracasse beaucoup d'utilisateurs, d'autant plus qu'elle n'est pas décrite dans les manuels « officiels ». Avez-vous que c'est un peu l'attrait du fruit défendu.

Nous cherchions la signification de ces lettres HIR, et Jean-Pierre Guibert nous propose une solution qui semble plausible, ce serait les initiales de Handling on Interior Registers, manipulations sur les registres internes. Et cela explique bien ce que nous connaissons déjà de ce fameux code 82.

Une autre hypothèse a été avancée, notamment par Jacques Pellaton : HIR viendrait de HIRE, louer. Quelqu'un pourra-t-il trancher et répondre de manière définitive à cette oh combien angoissante question ?

Dans notre rubrique aujourd'hui, on jette un coup d'œil sur la manière d'essayer toutes les ficelles de HIR, sans passer des heures à pianoter.

Puis on rentre plus à fond dans les sous-programmes internes de la TI59. Ce qui est proposé aujourd'hui à ce sujet n'est que le point de départ de bien d'autres aventures dont nous attendons vos récits.

Alors, à vos claviers !

Xavier de la Tullaye

## L'HEURE AFFICHEE SUR TI 58-59

Une idée m'est venue à la lecture, dans le n°12 de L'OI, du programme en Basic sur la « Pendule électronique de table » : pourquoi nous autres, possesseurs de « calculettes », n'aurions nous pas la même possibilité, puisque nous disposons de la fonction D.M.S. et du programme 23 du module standard ?

D'où la rédaction du petit programme ci-contre qui fait afficher l'heure (presque) exacte à une TI 58-59, toutes les 4 à 5 secondes :

Au pas 0, on introduit l'heure exacte sous la forme DD, MMSS et l'on appuie R/S. Au pas 15, la T.I. affiche l'heure modifiée.

Bon d'accord on n'atteint sans doute pas la précision d'une montre à quartz ! Mais pour une durée de quelques

|     |    |     |
|-----|----|-----|
| 000 | 91 | R/S |
| 001 | 24 | Pgm |
| 002 | 23 | 23  |
| 003 | 11 | R   |
| 004 | 00 | 0   |
| 005 | 93 | -   |
| 006 | 00 | 0   |
| 007 | 00 | 0   |
| 008 | 00 | 0   |
| 009 | 04 | 4   |
| 010 | 02 | 2   |
| 011 | 05 | 5   |
| 012 | 36 | Pgm |
| 013 | 23 | 23  |
| 014 | 12 | B   |
| 015 | 56 | Pgm |
| 016 | 61 | GTO |
| 017 | 00 | 00  |
| 018 | 01 | 01  |

minutes, la précision obtenue suffit. D'autant plus que l'on peut raffiner en agissant sur les instructions 9 à 11, qui constituent un véritable réglage de l'avance ou du retard de notre pendule (remplacer « 425 » par 421, par exemple, ou par 429 etc.)

J'imagine que l'intérêt

principal d'un tel programme est d'être facilement transformable en sous-programme, par exemple lors d'un jeu où l'on souhaite minuter le temps de réponse. Il faudrait alors remplacer l'instruction 00 par l'introduction du temps limite :

supposons que le temps de réponse ne doive pas excéder 1'30". Le sous-pro-

gramme introduirait alors 0.0130 suivi de « x t ». Puis, entre les instructions 15 et 16, il faudrait intercaler le test « x t » qui renverrait à une adresse de fin de partie en cas de dépassement de ce temps-limite.

Mais il y a sans doute bien d'autres utilisations et améliorations possibles.

Alain Léger

## Les TI 58-59 à cœur ouvert

Voici quelques manipulations quasi-ésotériques sur TI 58-59 qui permettent d'espionner le micro-logiciel de certaines fonctions internes.

Ainsi essayez GTO 004 LRN puis 2nd Pgm 02 SBR 240 RST RST RST LRN

Le programme chargé est donc celui ci-contre.

Les trois zéros du début, ainsi que les trois RST de la fin, sont indispensables.

En faisant partir le programme RST R/S, l'utilisateur va, s'il repasse en mode

|     |    |     |
|-----|----|-----|
| 000 | 00 | 0   |
| 001 | 00 | 0   |
| 002 | 00 | 0   |
| 003 | 00 | 0   |
| 004 | 36 | Pgm |
| 005 | 02 | 02  |
| 006 | 71 | SBR |
| 007 | 02 | 02  |
| 008 | 40 | 40  |
| 009 | 81 | RST |
| 010 | 81 | RST |
| 011 | 81 | RST |
| 012 | 00 | 0   |
| 013 | 00 | 0   |
| 014 | 00 | 0   |
| 015 | 00 | 0   |

LRN après l'arrêt, pouvoir constater qu'il se trouve à un endroit de la mémoire bien éloigné !

|     |    |     |     |    |     |     |    |     |
|-----|----|-----|-----|----|-----|-----|----|-----|
| 546 | 01 | RST | 573 | 43 | RCL | 601 | 04 | 04  |
| 547 | 01 | RST | 574 | 03 | 03  | 602 | 82 | HIR |
| 548 | 01 | RST | 575 | 54 | 5   | 603 | 48 | 48  |
| 549 | 00 | 0   | 576 | 65 | 6   | 604 | 82 | HIR |
| 550 | 00 | 0   | 577 | 53 | 3   | 605 | 07 | 07  |
| 551 | 00 | 0   | 578 | 43 | RCL | 606 | 33 | 33  |
| 552 | 06 | 6   | 579 | 02 | 02  | 607 | 44 | SUM |
| 553 | 75 | 7   | 580 | 75 | 7   | 608 | 05 | 05  |
| 554 | 43 | RCL | 581 | 43 | RCL | 609 | 01 | 1   |
| 555 | 04 | 04  | 582 | 01 | 01  | 610 | 61 | GTO |
| 556 | 65 | 6   | 583 | 33 | 33  | 611 | 02 | 02  |
| 557 | 43 | RCL | 584 | 55 | 5   | 612 | 36 | 36  |
| 558 | 01 | 01  | 585 | 43 | RCL | 613 | 94 | +/  |
| 559 | 55 | 5   | 586 | 03 | 03  | 614 | 44 | SUM |
| 560 | 43 | RCL | 587 | 54 | 5   | 615 | 01 | 01  |
| 561 | 03 | 03  | 588 | 54 | 5   | 616 | 82 | HIR |
| 562 | 54 | 5   | 589 | 34 | 34  | 617 | 00 | 00  |
| 563 | 55 | 5   | 590 | 54 | 5   | 618 | 33 | 33  |
| 564 | 52 | 2   | 591 | 92 | RTH | 619 | 94 | +/  |
| 565 | 52 | 2   | 592 | 44 | SUM | 620 | 44 | SUM |
| 566 | 43 | RCL | 593 | 01 | 01  | 621 | 02 | 02  |
| 567 | 05 | 05  | 594 | 82 | HIR | 622 | 32 | 32  |
| 568 | 75 | 7   | 595 | 00 | 00  | 623 | 22 | INV |
| 569 | 43 | RCL | 596 | 33 | 33  | 624 | 44 | SUM |
| 570 | 04 | 04  | 597 | 44 | SUM | 625 | 04 | 04  |
| 571 | 33 | 33  | 598 | 02 | 02  | 626 | 82 | HIR |
| 572 | 55 | 5   | 599 | 32 | XIT | 627 | 45 | 45  |
|     |    |     | 600 | 44 | SUM |     |    |     |

Il ne faudra pas chercher à aller en arrière par une commande BST (ou alors l'affichage sera incohérent). Si l'on procède au listage (de ce qui est un programme) par l'ordre SST alors l'utilisateur aura droit à un cours de programmation offert par Texas. Avec l'imprimante, il suffit de faire : 2nd list. Dans ce cas, il manque un chiffre à l'adresse du pointeur de programme. Aussi l'affichage 6881 sera imprimé 881.

On est en effet, en train de lister le micro-programme

de fonction telles que +, INV +, P R, INV P R D, MS OP 11, 12, 13, 14, 15, x INV x etc...

Ainsi la routine + commence au pas 6613, P R au pas 6684 etc etc...

La même manipulation avec un seul RST (81) ou deux RST, à la fin de la routine d'appel, donne accès à d'autres programmes internes. On remarquera dans ces programmes, que le code HIR est très utilisé. En fait, il semble avoir été prévu pour cela.

En ce qui concerne la question posée dans « Calculatrics » de novembre au sujet du code HIR 20, voici quelques renseignements (encore incomplets).

J'ai cru, quelques temps, qu'il s'agissait d'un code neutre (NOOP). En fait, il n'en est rien.

Appelons le micro-programme avec un seul RST (81)

GTO 004 LRN 2nd Pgm 02  
SBR 240 RST LRN

Initialisons RST et lançons R/S. Listons ensuite jusqu'aux pas 082/083. Le microprogramme utilise une fonction HIR 20.

Or la liste en question est celle du calcul de x et de OP 11 (variances). Elle débute au pas 082.

La fonction HIR 20 semble être un test pour un flag levé par le clavier. Lorsque l'on presse la touche x le test HIR 20 branche à 106 et au code RTN 92. Lorsqu'on appelle OP 11 le test laisse la routine se dérouler en 084 jusqu'en 106 et l'on calcule la variance.

Je ne suis pas arrivé à lever ce drapeau depuis un programme externe... Enfin, pas encore !

Jacques Laporte

## ESSAI AUTOMATIQUE DES POSSIBILITES DE HIR : AUTOPROGRAMMATION SUR TI 58-59

J'ai découvert dans L'OI de novembre la fonction HIR qui est d'ailleurs une fonction assez extraordinaire, et j'ai voulu en essayer toutes les possibilités. Mais entre les STO 82, BST, DEL, LRN, etc... on se lasse très vite !

Ma première idée, a été de faire HIR Ind 00. Mais bien sûr, ça ne marche pas.

J'ai donc réalisé un petit programme qui simule un adressage indirect de la fonction HIR.

La liste des programmes

| LIST TI 58 | LIST TI 59 |
|------------|------------|
| 000 76 LBL | 000 76 LBL |
| 001 11 R   | 001 11 R   |
| 002 32 X1T | 002 32 X1T |
| 003 05 S   | 003 01 1   |
| 004 69 OP  | 004 00 0   |
| 005 17 17  | 005 69 OP  |
| 006 32 X1T | 006 17 17  |
| 007 65 X   | 007 32 X1T |
| 008 01 1   | 008 65 0   |
| 009 00 0   | 009 01 1   |
| 010 00 0   | 010 00 0   |
| 011 85 *   | 011 00 0   |
| 012 09 9   | 012 85 *   |
| 013 02 2   | 013 09 9   |
| 014 00 0   | 014 02 2   |
| 015 00 0   | 015 00 0   |
| 016 08 8   | 016 00 0   |
| 017 02 2   | 017 08 8   |
| 018 95 *   | 018 02 2   |
| 019 42 STO | 019 95 *   |
| 020 49 49  | 020 42 STO |
| 021 03 3   | 021 99 99  |
| 022 69 OP  | 022 06 6   |
| 023 17 17  | 023 69 OP  |
| 024 25 CLR | 024 17 17  |
| 025 91 R/S | 025 25 CLR |
| 026 71 SBR | 026 91 R/S |
| 027 00 00  | 027 71 SBR |
| 028 85 85  | 028 01 01  |
| 029 91 R/S | 029 65 65  |
| 030 00 0   | 030 91 R/S |
| 031 00 0   | 031 00 0   |
|            | 032 00 0   |

Après avoir introduit le programme faites par exemple 35 A, puis allez voir au pas 85 (ou 165 sur TI 59) ce qui se passe. Aux pas 85, 86, 87 (165, 166, 167) on trouve 82 35 92 soit : HIR 35 INVSBR : TI à programmé elle-même la fonction HIR 35 !

**Exemple :** pour stocker dans le registre HIR 6 faire : 6 ; A ; ; R/S ; pour rappeler faire : 16 ; A ; R/S.

Bien sûr, ce programme a une utilité très restreinte, mais il permet de faire rapidement le tour des caractéristiques de la fonction HIR.

Patrick Jost