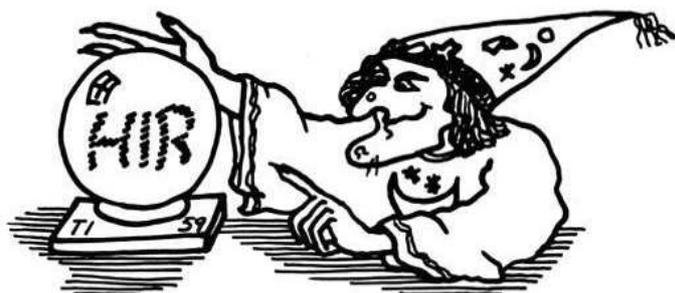


# La función HIR de la TI 58/59



*Tenemos una buena noticia: su TI 58 ó 59 se ha enriquecido últimamente con una nueva función, muy cómoda, en caso de necesitar memorias suplementarias.*

No se dice todo en los manuales de operación. Así el de las TI 58/58C/59, muy extenso, silencia una función que da acceso a una de las zonas prohibidas de la calculadora. ¿Cómo resistir la tentación de investigar? Esta intrusión es aprovechable, ya que para llevar a cabo los cálculos, la TI se reserva algunos registros de datos utilizados mediante la función *HIR*. Haga como ella.

¿Por qué *HIR*? Simplemente, porque la impresora PC-100 A (B ó C) imprime **82** (código) y *HIR* (abreviatura) cuando lista un programa donde ha introducido esta función. De acuerdo, me dirán, pero ¿Qué significa *HIR*? **44** (código) *SUM*, ó **25** *CLR*, se entienden fácilmente, pero no *HIR*.

Han sido barajadas varias hipótesis. La más plausible es que se trate de las abreviaciones de Handling on Internal Registers (manipulación de registros internos) pues es eso lo que permite esa función. Evidentemente gestiona la pila AOS de la calculadora y almacena los códigos alfanuméricos destinados a la impresora. Es un poco "la chica para todo" de su máquina. Recordemos sin más demora cómo se introduce esta función en un programa (ver figura 1).

Ocho teclas, pues, para un sólo paso de programa: sin lugar a dudas esta

STO	(ou RCL ou SUM)
82	
BST	
BST	
2nd Del	
SST	

Fig. 1

función es distinta de las demás. Si no va seguida de un código de dos cifras, *HIR* es inoperante. En el listado de un programa, este código ocupa un sólo paso e inmediatamente le sigue el código **82**. Ejemplo:

```
01 82 HIR
02 34 34
```

Para introducir este código de dos cifras en un solo paso, utilizaremos las teclas del uno al nueve para los códigos **01** a **09**, las teclas de las etiquetas para los códigos **10** a **19**, y las teclas de función para los códigos **20** (*2nd CLR*) a **98** (*2nd Adv*). El código de las teclas está —recordémoslo— compuesto por la fila del teclado donde se encuentra esta tecla (primera cifra) y su columna (segunda cifra). Así *RCL*, código **43**, significa tecla de la cuarta fila, tercera columna.

Llamaremos **a** la cifra de la izquierda del código y **b** la de la derecha. El primero identifica la operación ejecutada por *HIR*:

a = 0	Pasa el contenido de la pantalla a <b>b</b> . ( <i>STOb</i> )
a = 1	Pasa <b>b</b> a la pantalla ( <i>RCLb</i> )
a = 2	(¿puede ser una prueba de un indicador?).
a = 3	Incrementa en el contenido de la pantalla <b>b</b> ( <i>SUM b</i> )
a = 4	multiplica el contenido de la pantalla por <b>b</b> ( <i>Prd b</i> )
a = 5	Resta el contenido de la pantalla a <b>b</b> ( <i>INV SUM</i> )
a = 6	divide <b>b</b> por el contenido de la pantalla ( <i>INV Prd b</i> )
a = 7	( <i>INV Prd b</i> )
a = 8	( <i>INV Prd b</i> )
a = 9	( <i>INV Prd b</i> )

La cifra de la derecha **b** define qué registro es el utilizado (de uno a ocho). Cada registro tiene ya una función específica: el almacenamiento de resultados intermedios en los cálculos con parentesis, pero también:

registros 1 a 4 pila AOS  
 5 Op 01 + pila AOS  
 6 Op 02 + pila AOS  
 7 Op 03 + pila AOS  
 8 Op 04 + pila AOS

y puede que también tenga otros usos (¿Tiene alguna idea?).

Atención: cuando utiliza una función **a** de 3 a 9 y el número de la pantalla tiene un valor absoluto inferior a 1, debe poner la máquina en notación científica antes de introducir *HIR*, si nó tendrá problemas. Después de este resumen pasemos a las aplicaciones.

## MEMORIAS SUPLEMENTARIAS:

puede utilizar *HIR* para introducir números en memoria. La misma función permitirá sacarlos a la pantalla. Ejemplo:

00	76	Lbl
01	11	A
02	82	HIR
03	01	01 guarda lo visualizado en 1
04	91	R/S
05	76	Lbl
06	12	B
07	82	HIR
08	11	11 Devuelve 1 a pantalla
09	91	R/S

Para introducir este programa, debe pulsar sucesivamente las teclas siguientes: *LRN 2nd SBR A STO 8 2 BST*

**2nd BST SST 1 R/S 2nd SBR B STO  
8 2 BST BST 2nd BST A R/S LRN.**

Ensaye ahora esto: **1 2 3 A CLR B**. Después de haber introducido **123** en la memoria 1, la máquina lo saca en la pantalla. Inténtelo con otros números: **0,002 A CLR B, 1,77 EE 35 A CLR B, - 3,16 EE 45 A CLR B**, etc. Cada vez el número es almacenado y luego visualizado.

Calcule ahora **3 + 4 =** y pulse **B**. Vaya, la máquina ha visualizado el primer número, lo había introducido ella misma en el registro interno n° 1. Si hacemos la operación **3 x 4 + 2 =**, al pulsar **B** se visualiza **12** (es el producto de **3 x 4**). Después del cálculo **2+3x4**, al pulsar **B** aparece **2** en la pantalla. ¿Cómo funciona la pila AOS?. Pongámonos un momento en el lugar de la máquina y calculemos:

Primer paréntesis	6 + 5 = 11	divisor	(4 x (3 x (5 - 2 x 11)))
Calculado	2 x 11 = 22	divisor	(4 x (3 x (5 - 22)))
del segundo paréntesis	5 - 22 = -17	divisor	(4 x (3 x -17))
Tercer paréntesis	3 x (-17) = (-51)	divisor	(4 x (-51))
Resultado	4 x (-51) = -204	divisor	-204

**3x2+5x(2+3)/(4x(3x(5-2x(6+5))))**.

Ejecutaremos primero las operaciones del dividendo. Empecemos por el más pequeño de los paréntesis: **(2 + 3) = 5**. El dividendo es pues **(3x2+5x5)**. Ahora los productos: **3x2 = 6 y 5x5 = 25**. Que dan **6 + 25**. Y por fin la suma **6 + 25 = 31**. Dividendo = **31**.

000	03	3	012	01	1
001	93	.	013	07	7
002	01	1	014	03	3
003	07	7	015	04	4
004	03	3	016	65	x
005	04	4	017	01	1
006	65	x	018	04	4
007	02	2	019	95	=
008	95	=	020	91	R/S
009	66	PAU	021	00	0
010	03	3	022	00	0
011	93	.	023	00	0

Guardamos en memoria el dividendo y pasamos al divisor:

Ya podemos efectuar la división: **31/- 204 = - 0,1519607843**. Para obtener este cociente hizo falta retener sucesivamente 9 resultados. Es utilizando la función **HIR**, como la TI almacena esos resultados intermedios durante un

**Programa n° 2:  
24 teclas a pulsar para introducir este programa que ocupa 17 pasos.**

010	82	HIR	000	03	3
011	11	11	001	93	.
012	65	x	002	01	1
013	01	1	003	07	7
014	04	4	004	03	3
015	95	=	005	04	4
016	91	R/S	006	65	x
017	00	0	007	02	2
018	00	0	008	95	=
019	00	0	009	66	PAU

cálculo. Pero nada nos prohíbe utilizar esta función para optimizar un programa. Ganaremos 4 pasos, por ejemplo, en el programa 1 que multiplica **3,1734** por **2** (visualización del resultado en el tiempo de una pausa) y luego por **14**.

Utilizando **HIR**, obtenemos el programa n°2:

**HIR 11** lleva a la pantalla **3,1734** que la máquina había almacenado para hacer la primera multiplicación. Se puede objetar que podíamos igualmente ganar 4 pasos colocando **3,1734** en una memoria (**STO nn**) antes de sacarlo a la pantalla (**RCL nn**) para la segunda multiplicación lo que es cierto. Pero —y es aquí donde la función **HIR** nos presta unos servicios inestimables— ¿Qué hacer si no quedan memorias disponibles?. Si su programa es muy largo, y la partición escogida no le deja

◀ **Programa n° 1.**

**Programa n° 3:**  
Ninguna instrucción **STO**  
Ninguna instrucción **RCL**

suficientes registros de datos, descubre entonces los límites de su máquina. Se trata de optimizar, no por capricho, si no por necesidad.

La función **HIR**, utilizada adecuadamente, le facilitará los registros suplementarios que necesita.

En general será preferible utilizar **HIR** para manipular los últimos datos, es decir, las de los números más elevados. Se reducirán los riesgos de que la máquina almacene, sin saberlo nosotros, un resultado intermedio a lo largo de un cálculo complejo (varios niveles de paréntesis, por ejemplo). Si se tiene la menor duda al respecto, habrá que asegurarse de que no lo haga.

Muy a menudo, no es necesario utilizar **HIR 01, 02**, etc. Para almacenar los números en los registros internos: la máquina lo habrá hecho, ella misma, durante el cálculo. Bastará con determinar en qué memoria (1 a 8) se encontrará el n° que se quiere extraer.

Un ejemplo sencillo: introduzca un número cualquiera y pulse una de las teclas siguientes: **X, /, +, -** ó **YX**. Borre la pantalla (**CLR**). Es inútil hacer **HIR 01** para introducir el número en el registro interno número 1, ya se encuentra en él. La TI lo ha colocado en previsión de la operación **X, /, +, -** ó **YX** que anunció. **HIR 11** visualizará el número.

000	76	LBL	039	82	HIR
001	15	E	040	07	07
002	65	x	041	25	CLR
003	09	9	042	91	R/S
004	08	8	043	76	LBL
005	93	.	044	12	B
006	02	2	045	94	+/-
007	01	1	046	85	+
008	85	+	047	82	HIR
009	00	0	048	18	18
010	93	.	049	95	=
011	02	2	050	69	DP
012	01	1	051	10	10
013	01	1	052	82	HIR
014	03	3	053	37	37
015	02	2	054	82	HIR
016	07	7	055	17	17
017	95	=	056	69	DP
018	22	INV	057	10	10
019	59	INT	058	91	R/S
020	65	x	059	76	LBL
021	01	1	060	13	C
022	00	0	061	32	X:T
023	00	0	062	82	HIR
024	95	=	063	18	18
025	59	INT	064	67	EQ
026	82	HIR	065	44	SUM
027	08	08	066	00	0
028	25	CLR	067	91	R/S
029	91	R/S	068	76	LBL
030	76	LBL	069	44	SUM
031	11	A	070	08	8
032	94	+/-	071	93	.
033	85	+	072	08	8
034	82	HIR	073	08	8
035	18	18	074	91	R/S
036	95	=	075	00	0
037	69	DP	076	00	0
038	10	10	077	00	0

**HIR y TI 58C:** si utiliza una TI 58C, almacene un número en el registro interno 01 (número y luego HIR 01) y apague la calculadora. Enciéndala y compruebe el resultado del registro 01 (HIRM). Resultado: ¡cero!. Los registros internos no forman parte de la memoria permanente. Cuando apagamos la TI 58C, la información contenida en la pila AOS se pierde.

A título de ejemplo, introduzca el pequeño programa de juego ( $n^o$  3). Se trata de adivinar un número comprendido entre 1 y 100 que la máquina ha sacado al azar. La particularidad del programa es que no se utiliza instrucción *STO* ó *RCL*. Todas las memorias usuario están vacías. De hecho, el número secreto está conservado en la pila AOS (registro interno 08).

### JUGUEMOS CON HIR.

**Utilización:** Después de haber introducido un número semilla (generación del número secreto), proponemos un intervalo. Este intervalo no comprende los extremos inferior y superior con los que lo define. Si los extremos son 15 y 20, el intervalo está compuesto de 16, 17, 18 y 19. ¿Es 20 el número

	Introducir	Pulsar	Visualización
1.	Semilla	F	0
2.	Supuesto alto	A	0
3.	Supuesto bajo	B	1,0 ó - 1

secreto?. La TI 58 le contestará que es mayor que el extremo superior del intervalo. Si el número secreto está comprendido en ese intervalo, la pantalla se pone a cero. Si es más pequeño que el intervalo, la máquina responde -1. Si es más grande responde 1.

Vuelva a empezar con las etapas 2 y 3 hasta que crea haber encontrado el número secreto. Introdúzcalo y pulse C.

8,88 en la pantalla: ha acertado. 0 en la pantalla: ha fallado, puede escoger entre un nuevo intervalo (secuencias 2 y 3) o proponer otro número (tecla C).

Para hacer trampa, nada más sencillo: introduzca a partir del paso 100 un pequeño apéndice al programa:

100	76	LBL
101	14	D
102	82	HIR
103	18	18
104	91	R/S
105	00	0
106	00	0

*Lbl D HIR 18 R/S.* Le bastara con pulsar D y sin utilizar intervalo, conocerá el número secreto.

Para terminar, señalemos a los usuarios de TI57 que aún no hemos encontrado el equivalente de la función HIR en su máquina. En cambio, vamos descubriendo las funciones escondidas de la HP41C.

*Vincent Brunetta.*