

La interrupción no enmascarable, por fin disponible

COMO USAR LA NMI (I)

Primitivo DE FRANCISCO

Las posibilidades de uso de la interrupción no enmascarable (NMI) están totalmente bloqueadas por el software incluido en la ROM. Nuestro propósito es desbloquearla con un sencillo y barato dispositivo que se conectará al slot trasero del ordenador.

La interrupción NMI genera una bifurcación obligatoria en el microprocesador a la dirección fija 0066 en hexadecimal, abandonando el programa en curso tras la total ejecución de la instrucción que se encuentra en proceso al producirse la interrupción NMI. Para generar una NMI hay que llevar a nivel bajo, el terminal 14A del conector del ordenador. Cuando el Z-80 admite la interrupción, en el stack almacena la dirección de la instrucción siguiente a la cual volverá tras la instrucción de retorno RETN con la que habrá de acabar la subrutina que trata la NMI. Esta subrutina, por la propia configuración del Z-80, habrá de comenzar necesariamente a partir de la dirección citada (0066 H), en el Spectrum esta dirección se halla ocupada por la ROM en donde el fabricante ha tratado la NMI como mejor le ha parecido para sus propios intereses, restringiendo o mejor dicho, anulando el efecto de la NMI cuando ésta se produce.

La entrada de la NMI no se puede inhibir por software de modo que, una vez posibilitada su utilización, podrá salirse de cualquier programa comercial o propio con sólo oprimir un botón. Esto servirá para analizar los programas o para recuperar el control de la CPU sin que se pierda el contenido de la memoria, si el microprocesador se

ha bloqueado al ejecutar un programa o por cualquier otra causa.

En pocas palabras, se podría decir del botón NMI que es una auténtica tecla BREAK para salir de cualquier programa en el momento que el usuario lo desee.

Las interrupciones del Z-80 fueron tratadas en los números 63 y 64 de MICROHOBBY. En ellos se ofrece una visión completa del tema para aquellos que deseen profundizar ahora que ofrecemos la posibilidad de llevarlas a la práctica.

La NMI y la ROM

En el Spectrum la NMI está bloqueada por motivos que suponemos relacionados con las desprotecciones de software, pero esto priva de sus ventajas a la gran mayoría de usuarios legales que desearían obtener de su ordenador las máximas prestaciones.

En principio, cuando fue concebido el sistema operativo del Spectrum, se pensó en dotarlo de la posibilidad de acceso a la NMI. Luego cambiaron de opinión y para ello alteraron un único BIT en la ROM. Ahora no-

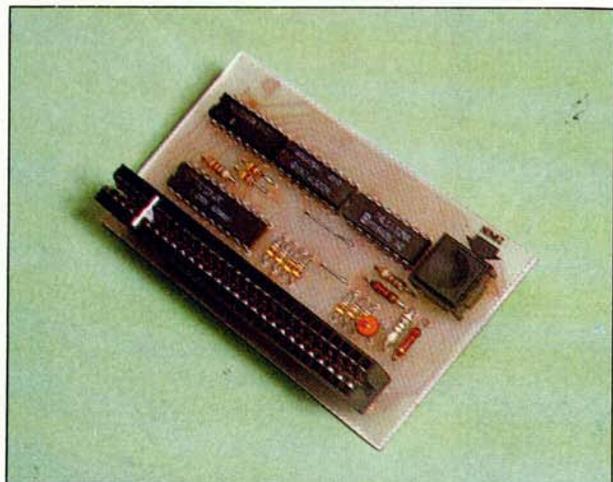
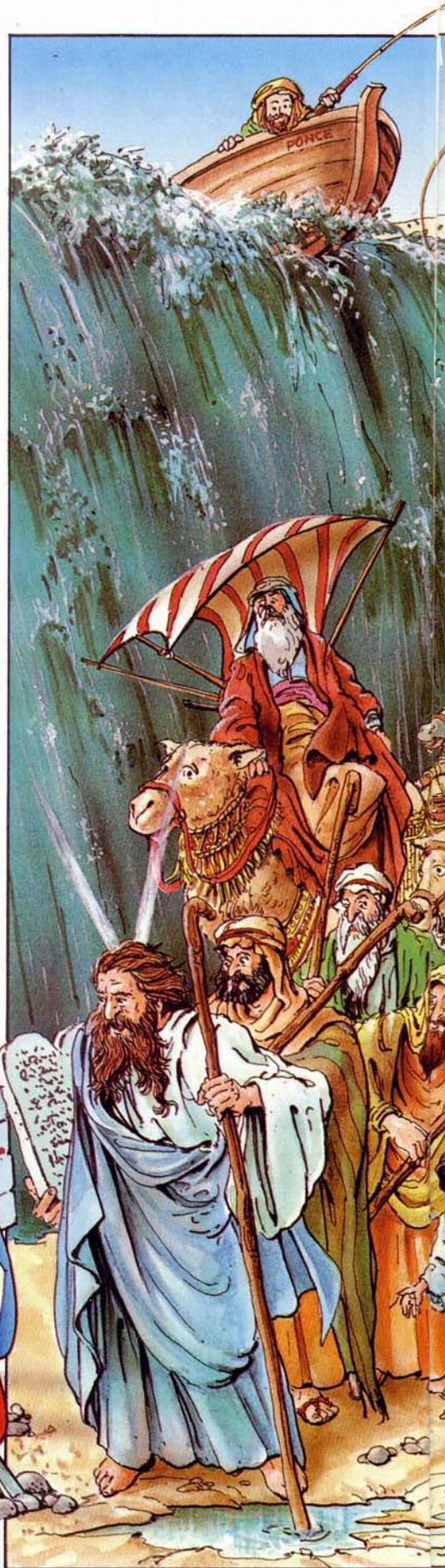
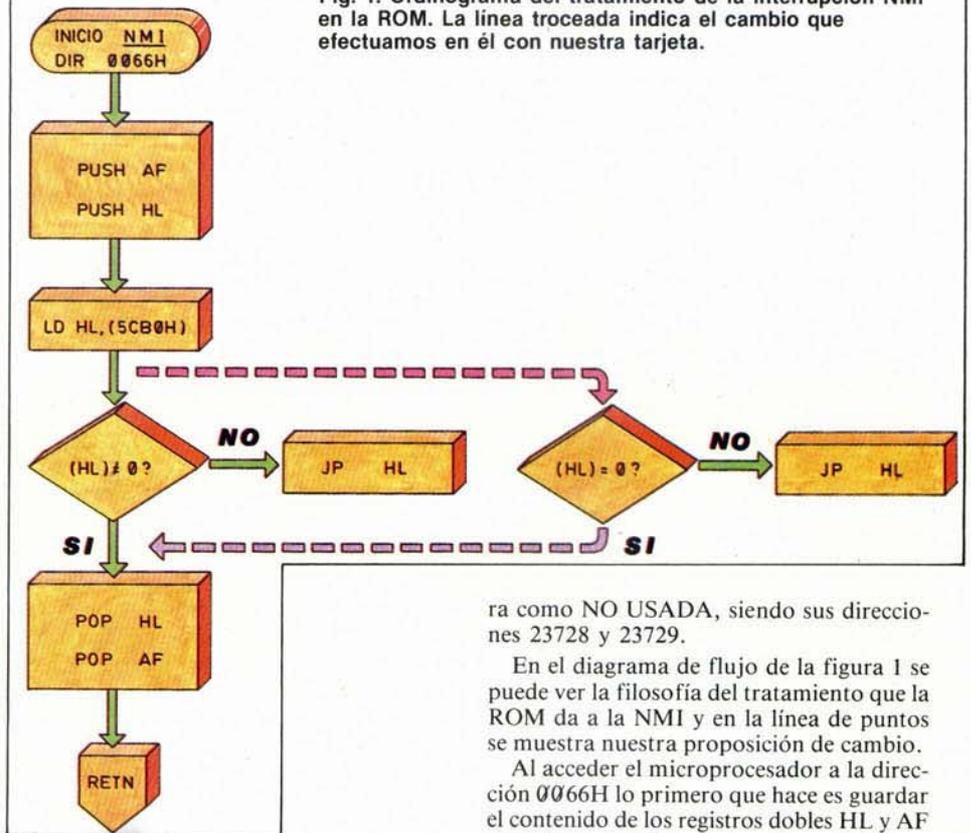




Fig. 1. Ordinograma del tratamiento de la interrupción NMI en la ROM. La línea troceada indica el cambio que efectuamos en él con nuestra tarjeta.



ra como NO USADA, siendo sus direcciones 23728 y 23729.

En el diagrama de flujo de la figura 1 se puede ver la filosofía del tratamiento que la ROM da a la NMI y en la línea de puntos se muestra nuestra proposición de cambio.

Al acceder el microprocesador a la dirección 0066H lo primero que hace es guardar el contenido de los registros dobles HL y AF en el stack, pues los va a usar inmediatamente. Luego carga en HL el contenido de la variable del sistema NMIADD, en la que el usuario habrá colocado previamente la dirección de salida de la subrutina de la NMI. Esta variable de sistema NMIADD (NMI ADDRESS) utilizada, como ya hemos indicado, las direcciones 5CB0H (23728) y 5CB1H (23729).

Al bifurcar el microprocesador a la dirección 0066H se encuentra con la ROM y por tanto su contenido es inalterable. Debido a

sotros volveremos a restituirlo con el circuito que proponemos en este artículo.

Echando una ojeada al desensamble de la ROM se observa que efectivamente existe una subrutina que trata la NMI; esta subrutina abarca desde la dirección 0066H hasta la 0073H. Con estos pocos bytes se consigue tomar control de la CPU por la NMI para entregárselo al usuario a través de una de las variables del sistema denominada NMIADD. En el manual de Spectrum figu-

| DIRECCION | C. MAQUINA | NEMONICO | LEYENDA |
|-----------|------------|----------------|--|
| 0066 | F5 | PUSH AF | ;GUARDA AF |
| 0067 | E5 | PUSH HL | ;GUARDA HL |
| 0068 | 2A005C | LD HL, (#5CB0) | ;CARGA HL CON EL DATO CONTENIDO EN #5CB0 Y #5CB1 |
| 006B | 7C | LD A, H | ;CARGA A CON H |
| 006C | B5 | OR L | ;HACE UN OR ENTRE H Y L |
| 006D | 2001 | JR NZ, #0070 | ;SALTA SI EL CONTENIDO DE HL NO ES CERO |
| 006F | E9 | JP HL | ;SALTA A DONDE INDIQUE HL |
| 0070 | E1 | POP HL | ;RESTAURA HL |
| 0071 | F1 | POP AF | ;RESTAURA AF |
| 0072 | ED45 | RETN | ;RETORNO INTERRUPCION NO ENMASCARABLE |

#20 ES EL BYTE DE CODIGO DE LA INSTRUCCION JR NZ, PARA APROVECHAR LA SUBROUTINA LA INSTRUCCION DEBE SER JR Z CUYO BYTE DE CODIGO ES #28. !SOLO SE DIFERENCIA EN UN BIT

Fig. 2. Listado en Assembler de la subrutina de la interrupción NMI

esto, la subrutina se apoya en dos posiciones de RAM en donde se carga la dirección absoluta de salto al entrar en la subrutina de la NMI. A continuación pregunta si el contenido de ambos bytes de la variable NMIADD son cero; si esto es cierto saltará a dicha dirección (con lo cual el micro quedará inicializado), si no son cero retornará al programa principal del cual salió a atender la NMI.

Estas dos posibilidades son: una catastrófica, porque pone a cero toda la memoria machacando el programa existente y la otra inútil porque retorna, es decir, no hace nada. En el primer caso el salto se produce inmediatamente según la variable NMIADD, en el segundo el retorno lo hace tras restaurar el contenido de los registros HL y AF.

Sin embargo, si se cambia la condición de salto por su opuesta el resultado es totalmente diferente y sobre todo útil. Esta segunda opción está representada en la figura 1 por líneas de trazos.

Si el contenido de HL es distinto de cero, entonces salta a la dirección apuntada por HL la cual será la prevista previamente por el usuario; por el contrario si HL contiene 0000H entonces retornará al programa principal sin ningún efecto restaurado HL y AF. Como se ve, los resultados son totalmente opuestos siempre en función de la instrucción de salto condicional (JR—salto relativo).

La figura 2 nos muestra el listado Assembler de la subrutina de la NMI contenida en la ROM con comentarios para cada línea.

La instrucción de salto relativo JR, que se halla en la dirección 006DH, es la que determina que la NMI se pueda utilizar o no. En la figura el octeto resaltado en un rectángulo muestra el byte de código de la instrucción: si es 20H el salto será JR NZ, XX es decir, salta a la dirección XX si el resultado de la comparación anterior no es cero.

Cambiando simplemente el código 20H por 28H, es decir, variando un único bit (D3) se obtiene el efecto contrario: 28H es JR Z,XX; salta a la dirección XX si el resultado de la comparación es cero. (Si (HL) contienen 0000H).

La solución por tanto, está en alterar este bit en la ROM, pero ésta es, por definición, inalterable, por lo que habrá que recurrir a otros procedimientos: cambiar la ROM por una EPROM con la corrección indicada o desde el exterior inyectar el código 28H por el bus de datos cuando se produzca en el bus de direcciones el valor 006DH. Dejamos para los más expertos la primera solución y la segunda para los que no se atreven a abrir su ordenador y desoldar la ROM.

El circuito

La figura 3 muestra el circuito eléctrico del dispositivo que ejecutará lo anteriormente dicho cuando se pulse el botón de la NMI.

Al oprimir el pulsador se dispara el monoestable formado con dos puertas lógicas, una de las cuales es tipo NAND SCHMITT para fijar los límites estables de temporización. En este monoestable el estado inestable está determinado en unos 100 ms. mediante R y C para eliminar los rebotes del pulsador. La salida de este monoestable pro-

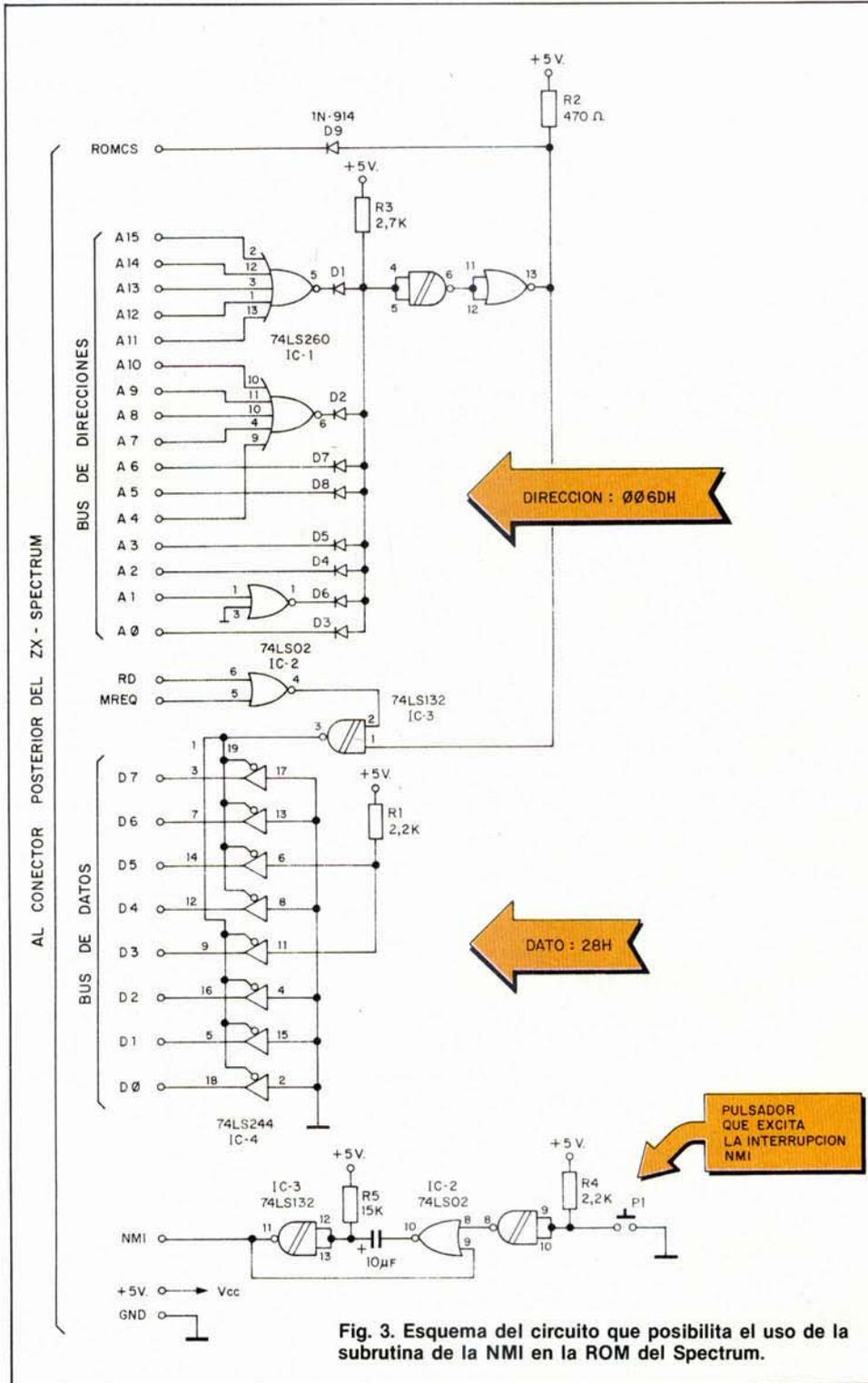


Fig. 3. Esquema del circuito que posibilita el uso de la subrutina de la NMI en la ROM del Spectrum.

porciona un nivel bajo que activa directamente la entrada NMI del ordenador.

La dirección 006DH es decodificada mediante el grupo de puertas NOR y NAND de la parte superior de la figura 3. La salida de estas puertas se agrupan con una puerta AND de ocho entradas confeccionada con diodos; a la salida de ésta hay dos puertas en serie que actúan como dos inversores, restaurando el nivel de la señal.

La salida de la última puerta NOR se aplicará a la entrada ROMCS para bloquear la ROM. Este bloqueo sólo se obtiene introduciendo un nivel alto capaz de proporcionar una intensidad próxima a los 8mA, cosa que se logra mediante R2 y D9 cuando

a la salida del conjunto de puertas decodificadoras hay un uno lógico.

IC-4 es un circuito integrado que contiene ocho triestados en cuya entrada se encuentra programado el número 28 H en formato binario.

El resto de las puertas OR se encargan de abrir los triestados para introducir el citado código 28 H (instrucción JR Z,0070H) por el bus de datos cuando el decodificador detecta la dirección 006DH de memoria en modo lectura del micro a la vez que bloquea la ROM interna, para desbloquearlo a partir de la dirección siguiente (006EH).

El montaje práctico, como habréis imaginado, lo abordaremos la próxima semana.

Montaje práctico para la habilitación de la NMI

COMO USAR LA INTERRUPCION NO ENMASCARABLE (y II)

Primitivo DE FRANCISCO

Este dispositivo ha sido concebido con pocos componentes, ya que su funcionamiento en sí mismo es simple, lo cual conduce a una corta y sencilla tarea de realización y puesta en marcha. Para colmo, resulta baratísimo.

La semana pasada analizamos la filosofía y el esquema del circuito, ahora pasaremos a su construcción y manejo.

Primeramente hay que adquirir los componentes y la placa de circuito impreso. Para la realización de la misma lo adecuado es utilizar los procedimientos ópticos y químicos que existen para estas tareas, sobre las que en las tiendas especializadas ofrecen información además de los materiales necesarios.

La figura 2 muestra, a tamaño real, la cara de pistas para la confección de las tarjetas que, como es costumbre, será de una sola cara.

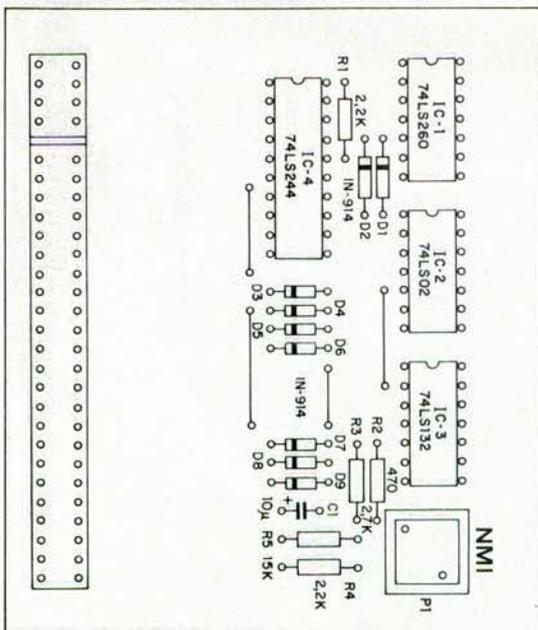


Figura 1. Cara de componentes con el detalle de la implantación de los mismos. Préstese atención en el montaje a la posición de los circuitos integrados y a la polaridad del condensador.

El montaje

Los componentes se implantarán siguiendo la figura número uno. Los circuitos integrados tienen una muesca que viene reflejada en el dibujo. Su colocación errónea o intercambio entre sí hará, lógicamente, que el dispositivo no llegue a funcionar e incluso puede ocasionar la destrucción de alguno de los propios circuitos.

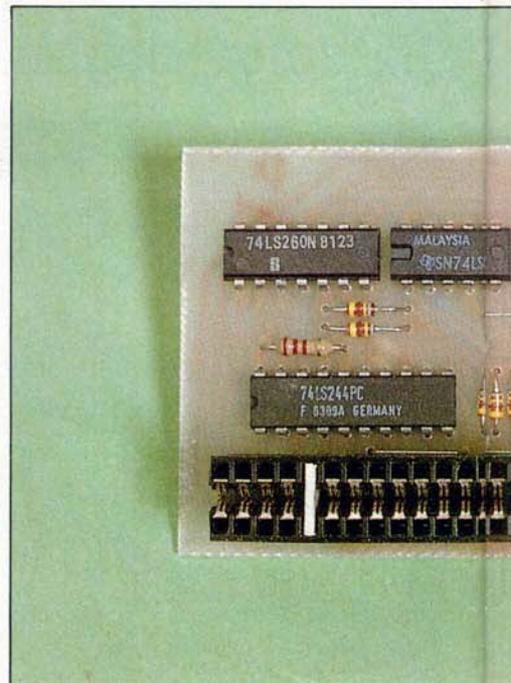
También es importante que los puntos de soldadura queden en perfecto estado dado que en la placa existen nodos de superficie muy reducida y próximos entre sí. Emplear preferentemente un soldador de punta fina de aproximadamente 15 vatios.

Los condensadores de tántalo tienen polaridad que también hay que respetar, de lo contrario al ponerlos bajo tensión se eleva su temperatura llegando a destruirse o explotar.

Si se adquiere un conector que no esté preparado para el Spectrum y hay que cortarlo a la longitud de 28 pares de terminales, el par que hace el número cinco ha de llevar un pequeño tabique para su correcto posicionamiento o simplemente, soldar los dos terminales que forman el par entre sí cortando sus prolongadores por el lado de soldadura.

Opcionalmente, se puede meter todo el montaje en una caja de plástico recortando previamente la ventana del conector y del pulsador, como algunos de nuestros lectores han hecho con montajes anteriores.

Esta tarjeta lleva bastantes diodos cuya colocación hay que cuidar fijándose en la posición del cátodo. En la figura, el cátodo viene marcado con una raya



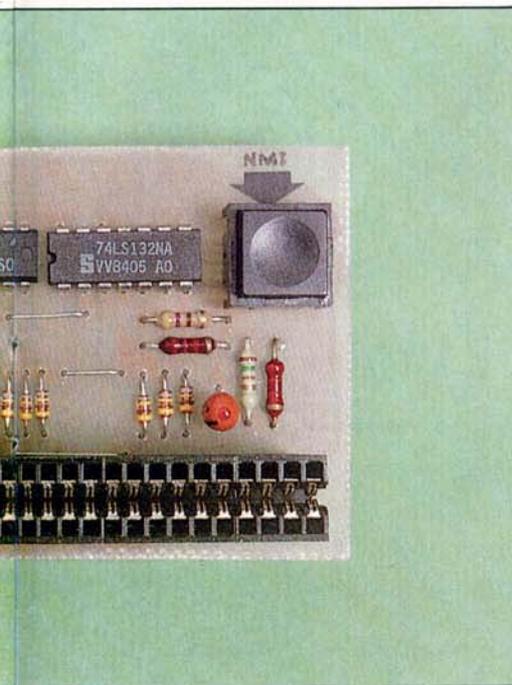
en el extremo correspondiente del cuerpo del diodo; no obstante, si se tiene duda es preferible ver la polaridad con un polímetro.

Puesta en marcha y uso

Cuando la tarjeta esté totalmente montada y revisada visualmente se podrá insertar en el conector trasero del ordenador. Es muy importante que se conecte con el Spectrum sin alimentar, pues de lo contrario un cierto desplazamiento entre macho y hembra en el momento de la inserción puede dar lugar a que alguna de las tensiones diferentes a +5V se derive por algún terminal de señales contiguo deteriorando instantáneamente alguno de los circuitos integrados del ordenador que tenga relación con el terminal derivado.

Tras la inserción, enchufar el ordenador. No pasará nada, todo funcionará normalmente. La prueba más elemental consiste en pulsar P1 para introducir la interrupción NMI. Tampoco pasará nada si todo va bien. Esto es debido a que tras la conexión, las posiciones de memoria 5CB0H y 5CB1H (en donde se apoya la subrutina de la NMI) son puestas a cero, lo que hará que se produzca un retorno de interrupción al encontrar en ambas direcciones un cero (ver lo dicho en la primera parte, la semana pasada).

Si por el contrario, tras la actuación del pulsador P1 se produce una iniciación del Spectrum o cualquier otra



La tarjeta, una vez montada, tomará este aspecto. El pulsador de la derecha iniciará el proceso de la NMI.

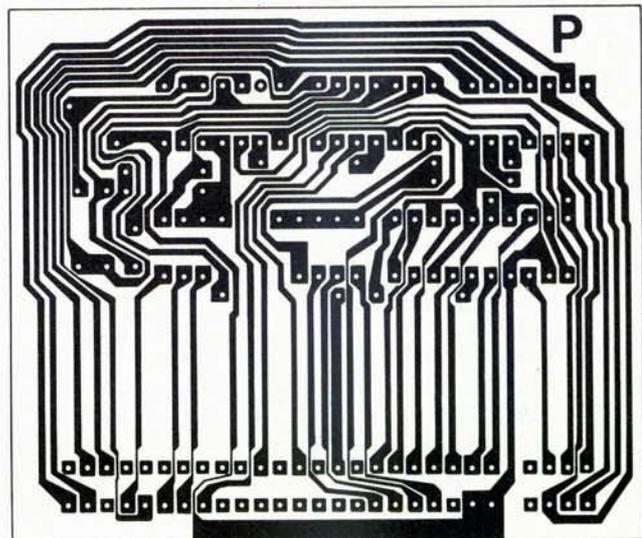


Figura 2. Placa del circuito impreso por la cara de las pistas, a tamaño real. Los taladrés serán todos de 1 mm excepto los del conector, que serán de 1,25 mm.

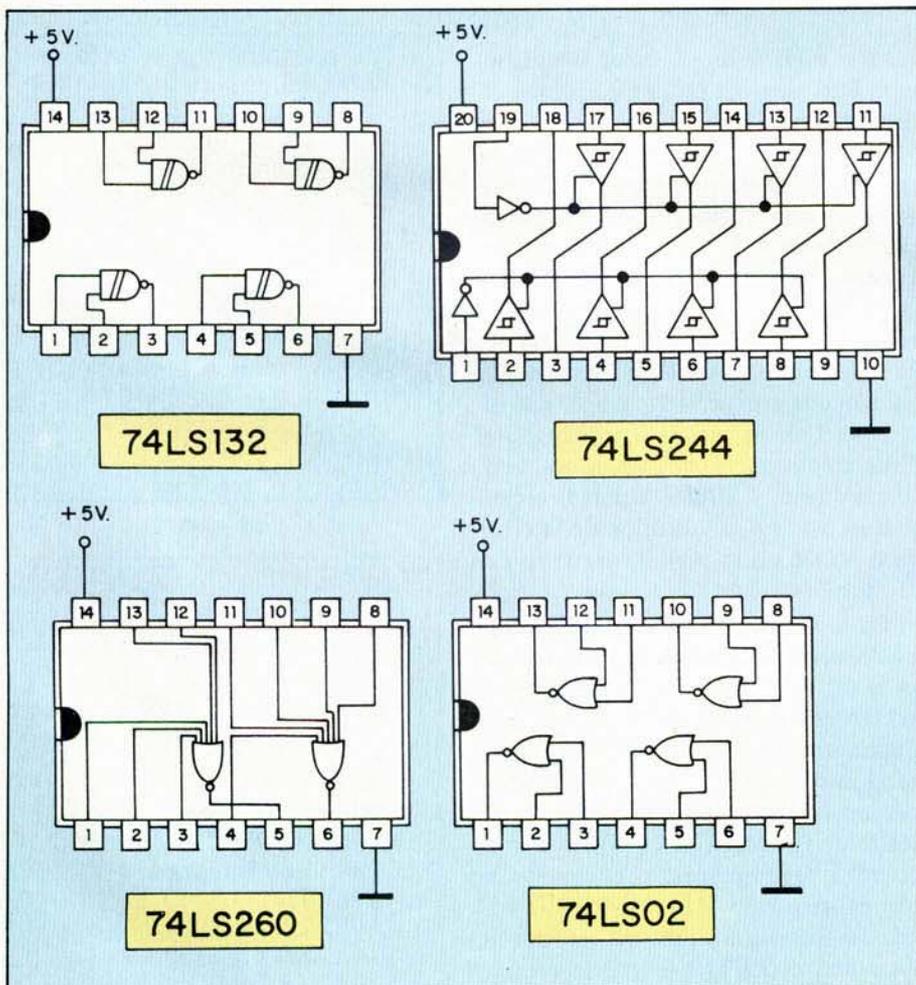


Figura 3. Diagrama interno de los circuitos integrados usados en este montaje.

alteración, algo va mal en la tarjeta, siendo preciso revisar el montaje, soldaduras, etc.

Una prueba más profunda y espectacular se consigue con la pequeña rutina

en Código Máquina que incluimos. Tras cargarla en el ordenador y ejecutarla, se verá un continuo desplazamiento hacia la izquierda de cuadrados multicolores. Oprimiendo el pulsador P1 se detiene

radicalmente el desplazamiento, para continuar de nuevo pulsando la tecla cero. Este proceso puede repetirse indefinidamente, lo que pondrá de manifiesto el buen funcionamiento de la tarjeta así como de la filosofía de la NMI.

El programa demostración se limita simplemente a cargar el contenido de la ROM en la zona de atributos de pantalla desplazando en un byte la carga cada vez que se repite la operación. Esto ocurrirá indefinidamente; pero si en un momento cualquiera se pulsa P1 se genera una interrupción no enmascarable (NMI) y el microprocesador bifurca a la subrutina de la NMI en la ROM que a su vez hará saltar a la dirección A02CH en donde se ha ubicado un bucle continuo que pregunta por el estado de la tecla cero; tras pulsarla se produce un retorno de interrupción y el programa principal continuará con el desplazamiento de colores en pantalla hasta una nueva interrupción y así sucesivamente.

Normas de uso de la NMI

Como ya se ha dicho, la subrutina de la NMI de la ROM tiene una gran potencia que podemos aprovechar, con el uso de nuestra tarjeta, para infinidad de aplicaciones. Como ejemplo citamos el empleo que le hemos dado durante el desarrollo de la pequeña rutina en código máquina que incluimos. Consistía en retornar el control al ensamblador al oprimir P1 cuando se encontraba co-

DESENSAMBLE DE LA RUTINA DEMOSTRACION NMI

| | | | |
|-----|----------------|-----|----------------|
| 10 | ORG #A000 | 210 | INC HL |
| 20 | ; PROGRAMA | 220 | INC DE |
| 30 | DI | 230 | DEC BC |
| 40 | LD HL, #5CB0 | 240 | LD A, B |
| 50 | LD (HL), #2C | 250 | CP 0 |
| 60 | INC HL | 260 | JR NZ, LA2 |
| 70 | LD (HL), #0A0 | 270 | LD A, C |
| 80 | LD HL, 0 | 280 | CP 0 |
| 90 | PUSH HL | 290 | JR NZ, LA2 |
| 100 | LA1 POP HL | 300 | JR LA1 |
| 110 | INC HL | 310 | ; SUBROUTINA |
| 120 | PUSH HL | 320 | LA3 LD A, #0EF |
| 130 | LD DE, #5800 | 330 | IN A, (#0FE) |
| 140 | LD BC, #0300 | 340 | RRA |
| 150 | EI | 350 | JR C, LA3 |
| 160 | HALT | 360 | POP HL |
| 170 | DI | 370 | POP AF |
| 180 | LA2 LD A, (HL) | 380 | RETN |
| 190 | AND #78 | 390 | END |
| 200 | LD (DE), A | | |

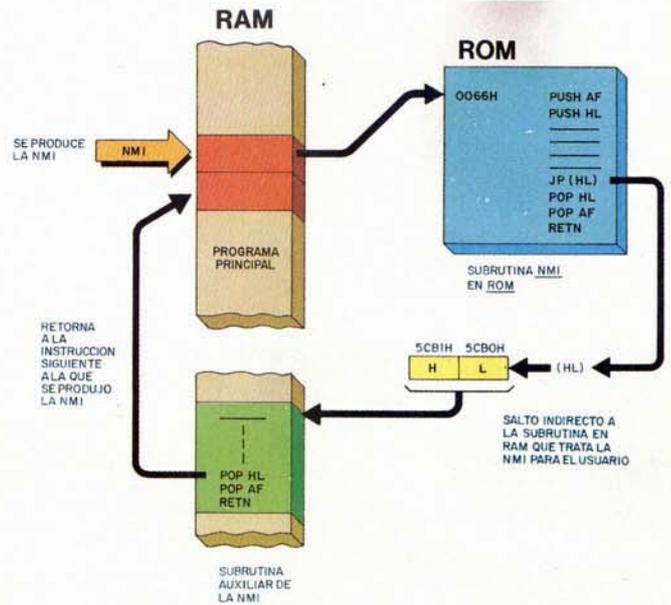


Fig. 4. Ciclo seguido por el tratamiento de la NMI desde su generación hasta el retorno.

```

S REM  "ROUTINA DE DEMOSTRACION
10 FOR X=40960 TO 41014
20 READ A
30 POKE X,A
40 NEXT X
50 PRINT AT 10,0;"=====
60 PRINT AT 11,1;"DEMOSTRACION
DEL USO DE LA NMI."
70 PRINT AT 12,0;"=====
80 RANDOMIZE USR 40960
90 DATA 243,33,176,92,54,44,35
,54,160,33,0,0,229,225,35,229,17
,0,88,1,0,3,251,116,243,126,230,
120,18,35,19,11,120,254,0,32,244
,121,254,0,32,239,24,225,62,239,
219,254,31,56,249,225,241,237,69
    
```

Ejemplo de utilización de la NMI en programas propios.

riendo la rutina demostración o cuando por error en el código máquina se perdía el control del microprocesador.

A continuación veremos los caminos que se siguen con el uso de la NMI para poder emplearla en programas propios. La primera precaución que hay que tener es cargar en los bytes de la variable NMIADD (direcciones 5CB0H y 5CB1H) la dirección a la cual saltará el microprocesador al producirse una NMI. En 5CB0H se cargará la parte baja y en 5CB1H la parte alta de la dirección absoluta de salto. Un método bastante eficaz para hacerlo desde BASIC es el siguiente:

```

RANDOMIZE dirección: POKE
23728, PEEK 23670: POKE 23729,
PEEK 23671
    
```

Cuando se encuentre en curso el programa principal y esté conectada la tarjeta objeto de nuestro montaje, al pulsar P1 el microprocesador saltará a la dirección 0066H guardando previamente

te en el stack la dirección de la instrucción siguiente del programa principal. Si el usuario ha tenido la precaución de guardar en la variable NMIADD la dirección de su rutina para el tratamiento de la interrupción, el microprocesador bifurcará hacia dicha dirección. Hay que tener en cuenta que NMIADD se pone a cero tras el reset, por lo que si no ha cargado nada en esta variable, se producirá un retorno de interrupción sin ninguna otra acción (siempre que esté conectada la tarjeta al ordenador). Esta circunstancia excluye naturalmente a la dirección 0000H como dirección usable, ya que la subrutina de la NMI en la ROM no lo permite según lo explicado.

En la figura 4 se muestra el camino que seguiría un correcto tratamiento de la NMI. Si el contenido de la variable NMIADD es distinto de cero no se producirá un retorno, sino un salto a la dirección contenida en ella. Hay que tener en cuenta que la subrutina de la ROM que empieza en 0066H lo primero que hace es guardar el contenido de los registros AF y HL, registros que habrá que restaurar antes del retorno de interrupción (RETN) al final del programa propio de tratamiento de la interrupción para que el contador de programa recoja correctamente la dirección de retorno del stack al ejecutarse la última instrucción: RETN.

Finalmente, efectuaremos un resumen de los detalles que comprenden el proceso:

- 1.º Hay que cargar en la variable del sistema NMIADD (5CB0H y 5CB1H) la dirección de salto a la subrutina de tratamiento en RAM de la NMI.
- 2.º Concluir la subrutina anterior con la siguiente secuencia de instrucciones:
POP HL
POP AF
RET N
- 3.º Mantener previamente conectada la tarjeta.
- 4.º El contenido 0000H en NMIADD no produce ningún salto.

LISTA DE MATERIALES

Resistencias

- R1 = 2,2 K
- R2 = 470 ohmios
- R3 = 2,7 k
- R4 = 2,2 k
- R5 = 15 k

Condensadores

- C1 = 10 F, 1,6 V Tántalo

Circuitos integrados

- IC 1 = 74LS260
- IC 2 = 74LS02
- IC 3 = 74LS132
- IC 4 = 74LS244

Semiconductores

- D1—D9 = 1N—914

Varios

- P1 = Pulsador para circuito impreso
- Conector hembra para el Spectrum
- Circuito impreso