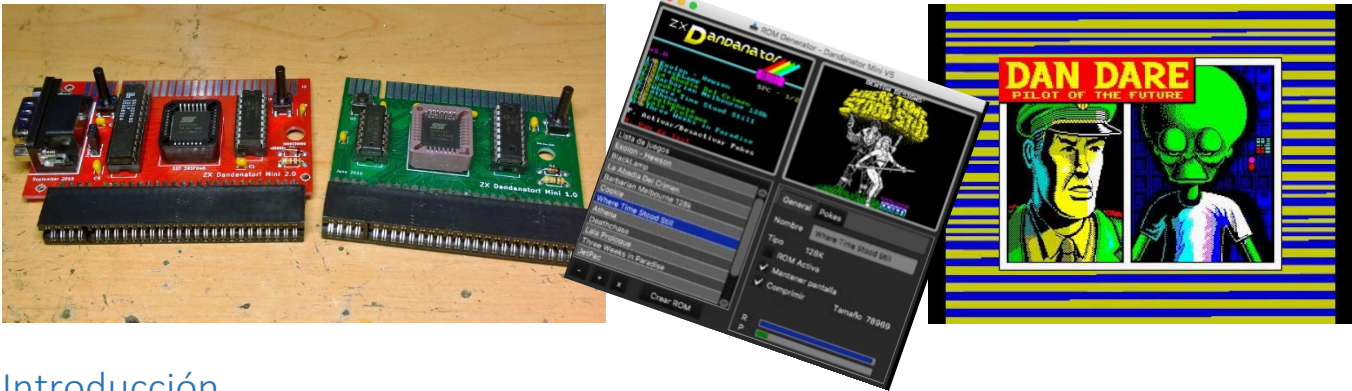


# Dan-TAP. TAP para slots de Dandanator. ( $\geq v8.1$ )



## Introducción

Dandanator, mediante el generador de ROMs, permite utilizar formatos SNA, Z80 y TAP. Todos estos formatos, al estar basados en Snapshot (estado “congelado” del spectrum), NO permiten integrar la carga de fases o datos posteriores desde la EPROM interna del dandanator.

Para permitir tener en el Dandanator juegos multicarga se desarrolló el formato MLD, pero resulta muy laborioso y requiere buenos conocimientos de ensamblador y de la propia máquina Spectrum, para así poder realizar una conversión de un juego a MLD. Las fases para llevarlo a término incluyen analizar el juego en sí, localizar las rutinas de carga, pasar los datos a Dandanator y generar un cargador para ello.

Para aumentar las funcionalidades del cartucho Dandanator, manteniendo la filosofía principal de “ser una experiencia agradable y de fácil uso para cualquier tipo de usuario” se ha desarrollado Dan-Tap.

Dan-Tap es una funcionalidad añadida al generador de ROMs y al código de menú de Dandanator que permite cargar software cuya procedencia sea un fichero TAP ¿ / TZX ? estándar de Spectrum (que no incluya bloques de carga turbo) y que utilice para la carga las rutinas de la ROM estándar de Spectrum.

Dan-Tap sirve para juegos o software multicarga, pero también se puede usar en juegos o software standard que no use multicarga.

## Funcionamiento

La idea básica detrás del Dan-Tap es la de pasar la información de un TAP a la eprom del cartucho Dandanator de forma que simule el acceso a cinta.

Para ello se utiliza, como rom del Spectrum, un slot que contiene una versión customizada de la rom de un spectrum 48k, integrando las siguientes variaciones:

- Rutina de carga estándar de cinta modificada para realizar la carga del TAP desde Dandanator.
- Autocarga. Cuando arranca la rom se pone directamente en modo carga del TAP, sin necesidad de escribir el clásico LOAD "" . Además se obvia el chequeo de RAM (fondo negro con líneas rojas verticales que desaparecen).
- Borde, fondo y color de letras en negro. Queda más agradable esa combinación durante la carga. El juego/programa del propio TAP podría cambiar posteriormente el borde durante la carga.

## Limitaciones

Debido a que la base de funcionamiento de la carga de datos es la de sustituir la rutina de cinta al sustituir la ROM del Spectrum por una ROM customizada, hay algunas limitaciones a tener en cuenta:

- a) No funcionará el Software de 128k que utilicen las roms de editor, acceso a disco (+2A, +2B, +3). Por limitaciones solo se puede utilizar software que utilice la ROM denominada de 48K. Esta limitación realmente afecta a muy pocos programas, y si acaso, a juegos realizados en BASIC 128k.
- b) No serán válidos los programas que intenten cargar datos en las zonas propias de la ROM de spectrum (direcciones 0-16383). Algunos TAP generados con ciertos Transtapes para 128k dan problemas debido precisamente a esto. Ello es debido a la forma en que Dandanator interpreta los comandos que se le envían, los cuales se realizan escribiendo en la zona 0-16383, y por ello el funcionamiento es alterado y suele acarrear un cuelgue y/o reinicios del Spectrum.
- c) No funcionará con software que utilice rutinas propias de carga.

## Informacion técnica

### Ficheros TAP

Los ficheros TAP son básicamente contenedores de los datos que antiguamente estaban grabados en las cintas que se ponía en un cassette (o el famoso computone) para cargar el juego o aplicación en el Spectrum.



El fichero TAP se compone de varios conjuntos de datos o bloques que se van cargando en memoria del Spectrum y ejecutando o conteniendo datos e información para que disfrutemos de un juego o aplicación.

Cada bloque de datos de un TAP está estructurado de la forma indicada en la Figure 1-TAP File Format.

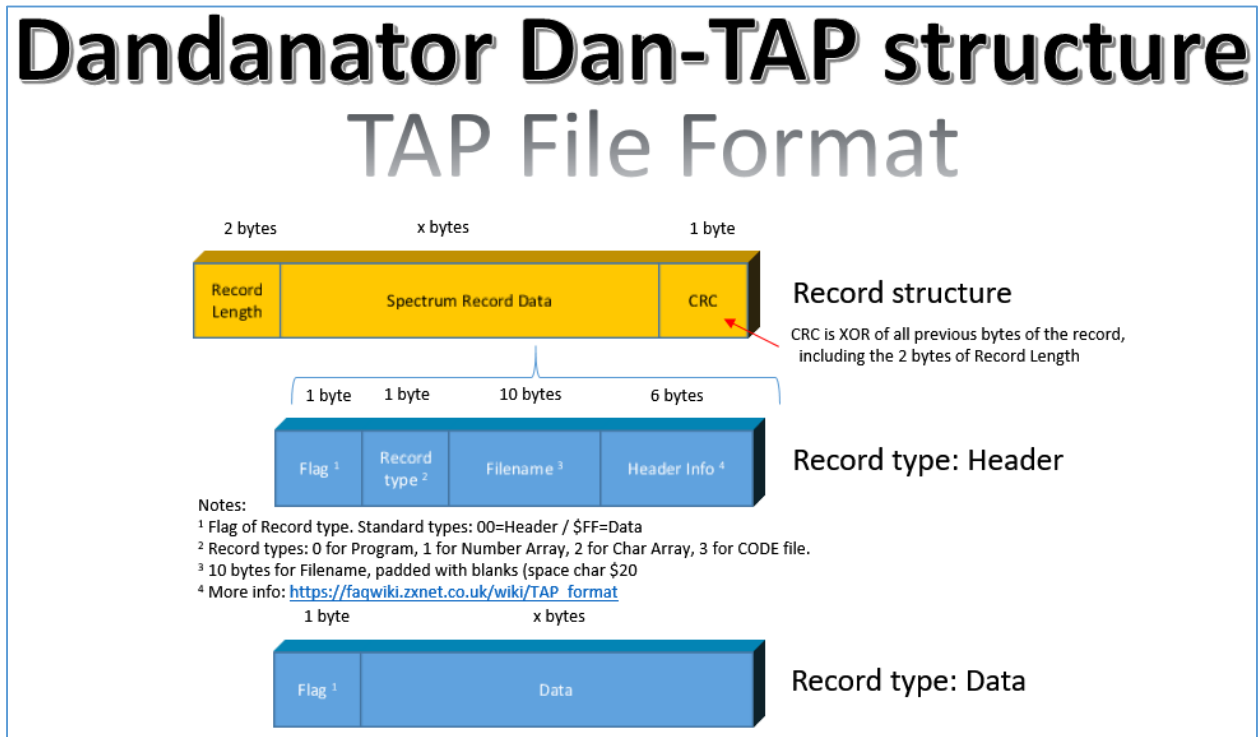


Figure 1-TAP File Format

### Slots del formato Dan-TAP

Esta información del fichero TAP se integra en los slots necesarios para contener tanto sus datos como las rutinas e información adicional que requiere Dandanator para realizar correctamente la carga. La estructura es la indicada en la Figure 2 - Composing the Slots

# Dandanator Dan-TAP structure

## Composing the Slots

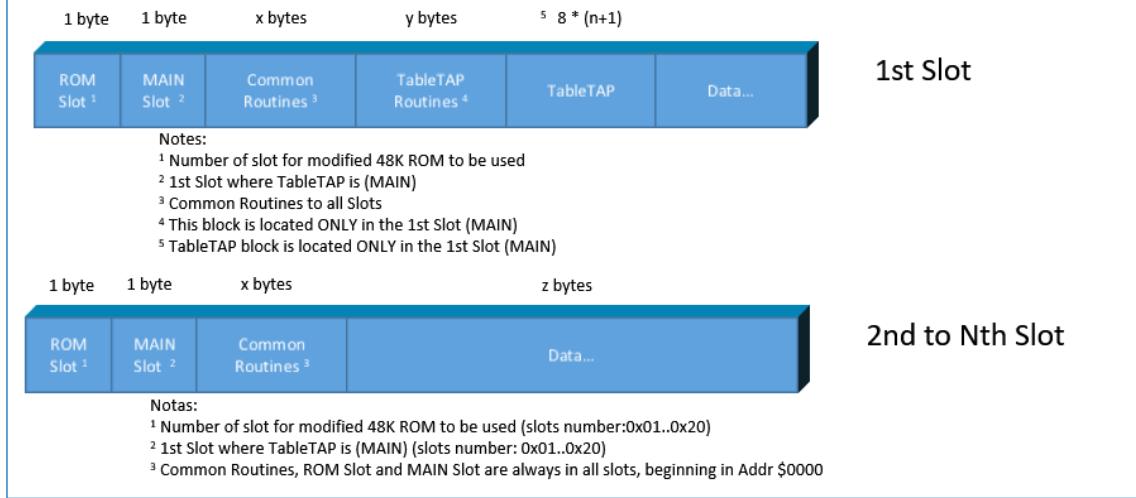


Figure 2 - Composing the Slots

En cada slot hay un área de rutinas comunes (common routines) necesarias para que Dandanator no requiera utilizar memoria RAM del spectrum más allá del uso del stack o pila (SP) y que sirven de apoyo para poder cambiar de un slot a otro de manera fluida así como para transferir y/o descomprimir los datos.

### TableTAP

La tabla de datos (TableTAP) son registros de 8 bytes que incluye información esencial para poder realizar la carga de los ficheros y datos de forma correcta. La estructura requerida se recoge en la Figure 3 - Composing data for TableTAP.

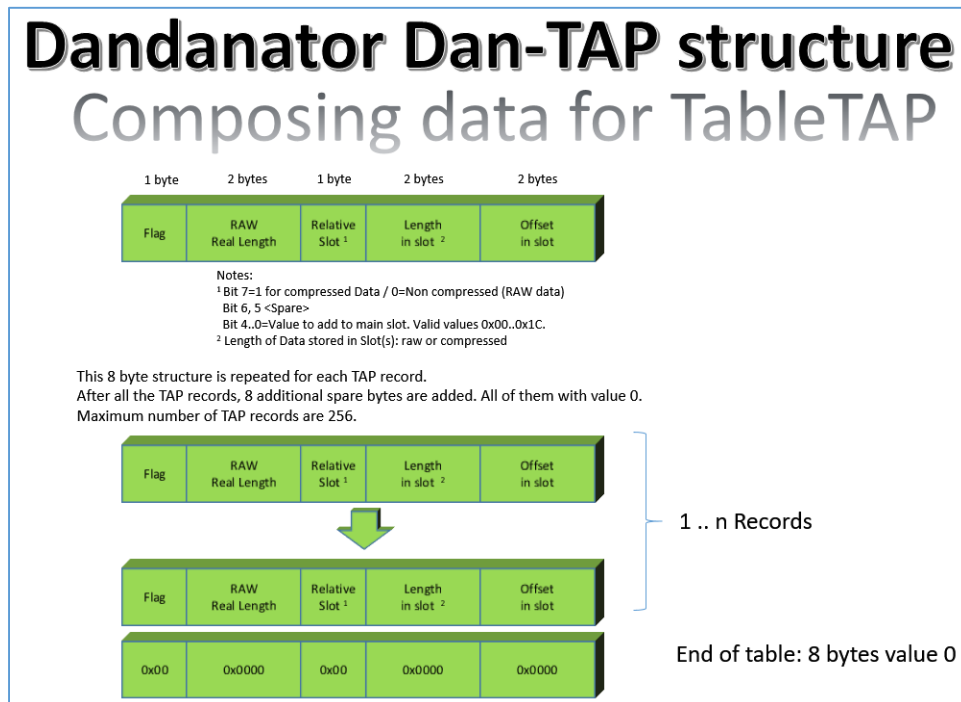


Figure 3 - Composing data for TableTAP

## Ejemplo

Veamos un ejemplo de un TAP que contiene 8 registros. Abriendo el fichero TAP en ZX-BlockEditor vemos su contenido (Figure 4 - TAP File Contents - Example).

No.:	Block type/name:	Additional info:	Length:	Address:	Skip:	Expand:	CRC:
0001	FILE TYPE	Simple tape files					
0002	FILE INFO	Format:TAP/BLK, CRC:52FE4DEC, Size:...					
0003	PROGR "FAIRLIGHT" LINE 0	Prog:319; Vars:0					393D66A2
0004	Program data		319				CA960BDD
0005	BYTES "FAIRLIGH_0" CODE 40000,4772		17				0C9CA9BF
0006	Standard data		4772	40000			4AAE389C
0007	BYTES "FAIRLIGH_1" CODE 26000,13976		17				71531DDF
0008	Standard data		13976	26000			BA49A964
0009	BYTES "FAIRLIGH_2" CODE 26000,33765		17				8E3201C9
0010	Standard data		33765	26000			8B8E4E25

Figure 4 - TAP File Contents – Example

Las 2 primeras filas “FILE TYPE” y “FILE INFO” no forman parte de los registros de datos a transferir, son solo datos informativos que ofrece el programa ZX-BlockEditor.

El fichero contiene 8 registros, 4 de ellos son de tipo Cabecera (No: 0003, 0005, 0007 y 0009) y otros 4 de tipo Datos (No: 0004, 0006, 0007 y 0010).

Puede encontrar más información sobre el formato TAP en [https://faqwiki.zxnet.co.uk/wiki/TAP\\_format](https://faqwiki.zxnet.co.uk/wiki/TAP_format)

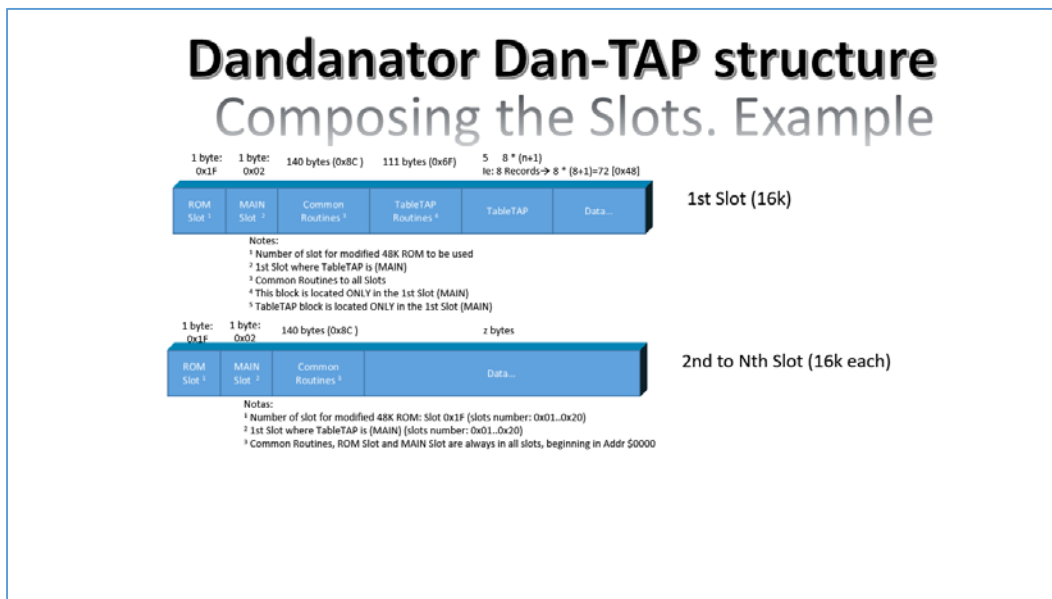


Figure 5 - Composing Slot - Example

# Dandanator Dan-TAP structure

## Composing data for TableTAP. Example

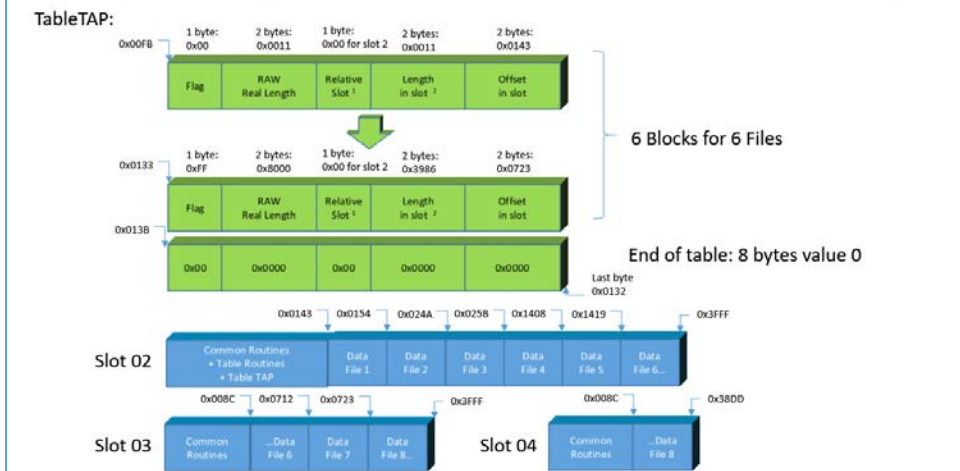


Figure 6 - Composing TableTAP - Example

Como se aprecia en la Figure 6 - Composing TableTAP - Example, si un fichero excede la posición 0x3FFF se continúa en el siguiente slot, pero siempre a partir del byte 0x008C (ya que del 0x0000 a 0x008B están las rutinas comunes).

```

HxD - [D:\Descargas\ZX juegos\fair_tap\FAIR_FINAL.ROM]
Archivo Edición Buscar Ver Análisis Extras Ventanas ?
16 ANSI hex
FAIR_FINAL.ROM
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00004000 1F 02 CD 18 00 EB 1B C3 82 00 00 00 32 00 00 3D ..í..ë.Ã,...2..=
00004010 20 F8 3E 2C 3D 20 FD C9 3E 80 ED A0 CB 74 C4 6C ø>.= ýÉ>ëi ÈtÀl
00004020 00 CD 61 00 30 F4 D5 01 00 00 50 14 CD 61 00 30 .ía.0óÖ...P.ía.0
00004030 FA D4 61 00 CB 11 CB 10 38 24 15 20 F4 03 5E 23 úôa.È.È.è$. ô.è#
00004040 CB 74 C4 6C 00 CB 33 30 0C 16 10 CD 61 00 CB 12 ÈtÀl.È30...ía.È.
00004050 30 F9 14 CB 3A CB 1B E3 E5 ED 52 D1 ED B0 E1 30 0ù.È:È.ääiRÑi°áo
00004060 C0 87 C0 7E 23 CB 74 C4 6C 00 17 C9 08 3C 67 CD À+À~#ÈtÀl..È.<gí
00004070 0A 00 7C 08 21 8C 00 C9 CB 74 C4 6C 00 ED A0 EA ..|!è.ÈÈtÀl.i è
00004080 78 00 3A 01 00 21 F2 00 E5 C3 0A 00 7D 21 F3 00 x:...!ò.ää...!}ó.
00004090 01 08 00 09 3D 20 FC 23 BE 20 08 23 BE 2B 20 03 ...=. ú#%. ##+ .
000040A0 21 FC 00 2B 08 BE 28 04 3E 01 18 E7 23 7E BB 20 !ù.+.*(.)..ç#~»
000040B0 05 23 7E BA 28 0A 21 AA 05 3A 00 00 E5 C3 0A 00 .#~*(!*.:...ää..
000040C0 23 3A 01 00 4F 11 82 00 DD 7C FE 40 38 0D 7E CB #:.0.,.Ý|p@8.~È
000040D0 7F 20 05 11 7D 00 18 03 11 02 00 D5 E6 1F 81 4F ...}.....Öæ..O
000040E0 08 79 23 4E 23 46 23 5E 23 56 EB DD 54 DD 5D C3 .y#N#F#^#VeÝTÝjÃ
000040F0 0A 00 DD 62 DD 6B 21 AF 05 18 BE 00 11 00 00 11 ..ÝbÝk!~.%.....
00004100 00 43 01 FF 3F 01 80 F6 00 54 01 00 11 00 00 11 .C.y?.èö.T.....
00004110 00 4A 02 FF A4 12 80 AD 11 5B 02 00 11 00 00 11 .J.ýH.È..[.....
00004120 00 08 14 FF 98 36 80 6D 32 19 14 00 11 00 01 11 ...ý~6Em2.....
00004130 00 12 07 FF E5 83 81 62 77 23 07 00 00 00 00 00 ...ýâf.bw#.....
00004140 00 00 00 00 46 41 49 52 4C 49 47 48 54 20 3F 01 ...FAIRLIGHT ?.
00004150 00 00 3F 01 00 00 01 65 00 DA C3 A7 3A E7 A0 03 ..?.....e.ÚÀS:ç .
00004160 D9 B0 22 37 18 22 3A FD 05 32 35 65 60 33 08 F5 Û°"7."":ý.25e~3.è
00004170 AC A7 2A A7 50 2C 0D 22 3B 22 00 14 01 20 20 46 -S*SP,":;... F
00004180 41 49 52 00 4C 49 47 48 54 20 41 20 01 50 52 45 AIR.LIGHT A .PRE
00004190 4C 55 44 45 8C 14 31 32 38 04 14 00 00 22 27 27 LUDEE.128...."'
000041A0 22 43 52 41 43 06 4B 45 44 C6 54 07 49 60 4E 07 "CRAC.KEDET.I'N.
000041B0 20 42 59 20 4D 01 41 54 41 53 4F 46 54 81 1C 0D BY M.ATASOFT...
000041C0 00 02 2F 00 F4 40 4C 33 37 33 39 22 0D 2C AF 22 ../.ò@L3739".,"
Posición: 40FB Bloque: 40FB-4142 Longitud: 48 * Modificado * Sobrescribir
    
```

Figure 7 - Volcado Slot02

En la Figure 7 - Volcado Slot02 está parte del volcado hexadecimal del Slot 2.

El primer byte 0x1F indica el slot en el cual está la ROM48k adaptada para la carga de TAP.



El segundo byte 0x02 indica que el primer slot del conjunto es el 02.

Del byte 0x4002 hasta el 0x408B (marcado en color rojo) es la zona de rutinas comunes a todos los slots.

Del byte 0x408C hasta el 0x4142 (marcado en fondo azul) es la TableTAP.

A partir del byte 0x4143 comienzan los datos de los diferentes ficheros, por ejemplo del 0x4143 al 4159 es la cabecera del primer fichero que cargará.

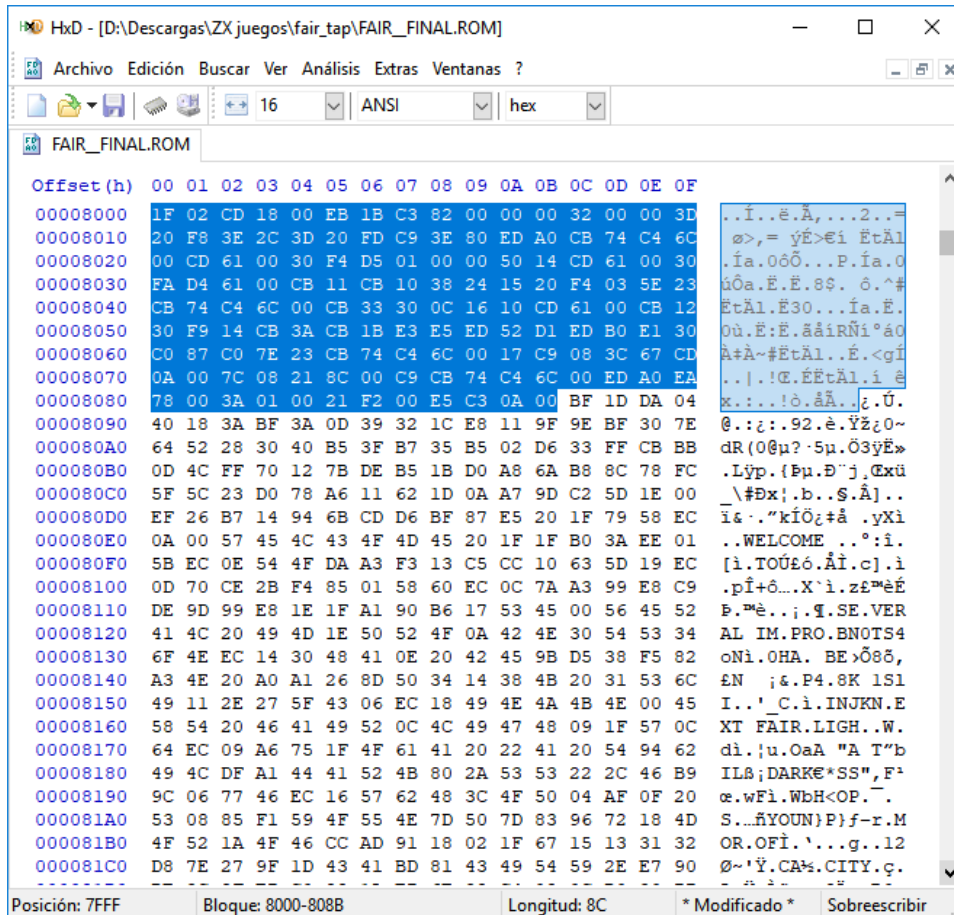


Figure 8 - Volcado del slot 03

En la Figure 8 - Volcado del slot 03 está parte del volcado hexadecimal del Slot 3.

El primer byte 0x1F indica el slot en el cual está la ROM48k adaptada para la carga de TAP.

El segundo byte 0x02 indica que el primer slot del conjunto es el 02.

Del byte 0x4002 hasta el 0x408B (marcado en color rojo) es la zona de rutinas comunes a todos los slots.

En este slot y los subsiguientes no hay TableTAP ya que la misma está únicamente en el slot 02.

A partir del byte 0x808C están los datos de los diferentes ficheros.

## Interioridades de la carga

Para que puedan utilizarse varios Dan-TAP en un mismo menú de Dandanator, el menú almacenará el número de primer slot de Dan-TAP a utilizar para la carga. La dirección (provisional) 66 (0x42) es la utilizada para ello, conteniendo un valor de 0..31 correspondiente al slot real 1..32.

Además para poder realizar correctamente la carga se requiere de un registro que guarde el número de registro/fichero que se carga en cada momento. Para ello se dispone de uno de los registros internos del PIC (memoria no volátil), concretamente la dirección (provisional) 65 (0x41).

La rutina de carga de cinta comienza en la dirección de ROM 0x0556 rutina LD-BYTES. Cuando hacemos LOAD "" se ejecutan una serie de rutinas que van llamando a esta dirección para realizar la carga de cabeceras y datos.

Pero diversos juegos y programas llaman a la rutina de carga estándar un poco más adelante, ya sea para evitar el control de BASIC de los errores de carga o bien para realizar cargas con otras combinaciones de colores de borde.

Debido a ello la rutina de carga de Dan-TAP comienza en 0x56B.

Los pasos para realizar la carga son:

1. "Despertar" al dandanator, habilitando los comandos.
2. Obtener el número de fichero que toca cargar, recuperando del PIC la variable.
3. Se cambia al primer slot del Dan-TAP.
4. Se toman los datos del fichero a cargar.
5. Se comprueba el Flag, si el flag no coincide se pasa al siguiente fichero y se vuelve al paso 4. Si llegamos al último fichero se vuelve al primer fichero nuevamente.
6. Se comprueba que la Longitud de datos a cargar coinciden con las del fichero actual. Si no coinciden se retornará de la carga con un Error de Carga y se va al paso 10.
7. Se cambia al slot donde comienzan los datos del fichero a cargar.
8. Se transfiere o descomprime los datos del fichero a cargar. Si se lleva al final de un slot (0x3FFF) se cambia al siguiente slot y se continúa la transferencia o descompresión.
9. Se vuelve al primer slot del Dan-TAP
10. Se retorna al slot de la ROM48 modificada para Dan-TAP, si la carga fue correcta (Flag y Longitud de datos a cargar correctos) se vuelve con Carry activo.

## Pendientes

1. Añadir formato Dan-TAP (¿nuevo icono?) al menú de Dandanator. A realizar en generador y en ensamblador del menú.
2. El menú de Dandanator inicializará la variable que guarda el número de slot del Dan-TAP a cargar, siendo su valor 0 para slot 1, valor 1 para slot 2, etc... A realizar en ensamblador del menú, previsto usar la dirección PIC 66.
3. El menú de Dandanator inicializará a 0 la variable que guarda el número de fichero que toca cargar. A realizar en ensamblador del menú, previsto usar la dirección PIC 65.
4. Para permitir incluir varios ficheros Dan-TAP e integrarlos en el menú correctamente, se modificará las rutinas en la ROM48K adaptada para que obtenga una variable interna del PIC (dirección 66) donde esté el valor del slot del TAP a cargar. Igualmente el menú del Dandanator deberá ser modificado para que al lanzar un Dan-TAP guarde el valor correspondiente en el registro 66 del PIC.