

EL PROGRAMA MICROSOFT CODEVIEW

El programa **CodeView** es una utilidad para la depuración de programas para entornos tipo PC, que se incluye en el Microsoft Macro Assembler versión 5.1.

Para el mejor aprovechamiento de las posibilidades del CodeView es preciso generar el programa ejecutable con las siguientes opciones:

- Ensamblar: MASM /Zi PROGRAMA;
- Enlazar: LINK /CO PROGRAMA;

LA PANTALLA DEL PROGRAMA CODEVIEW

En la pantalla del programa CodeView pueden observarse distintas zonas y ventanas:

- La **barra de menús** en la parte superior.
- La **ventana de visualización**, bajo la barra de menús, que permite observar los valores almacenados en la memoria.
- La **ventana de registros**, en la parte derecha, donde se muestra el contenido de todos los registros de la CPU accesibles al programador, incluidos los flags del registro de estado.
- En la **ventana principal** o **ventana del código**, aparecen las líneas de código fuente (*.ASM) numeradas, marcando en vídeo inverso la línea correspondiente a la siguiente instrucción a ejecutar y, sobre iluminadas, las líneas en las que se hayan activado los *puntos de break* o *puntos de parada*.

```
File View Search Run Watch Options Language Calls Help | F8=Trace F5=Go
----- suma32.ASM -----
0) dato1 l 8 : 5054:0006 FF FF FF FF 03 4E 42 30 .....NB0
1) dato2 : 5054:0008 FF .
2) resul : 5054:000A 4E03

9:
10: .CODE
11: Inicio: mov ax,@data
12: mov ds,ax
13: xor si,si ;índice
14: mov cx,NumBy ;Número de bytes a sumar
15: clc ;Acarreo inicial a cero
16: Bucle:
17: mov al,Dato1[si]
18: adc al,Dato2[si]
19: mov Resul[si],al
20: inc si
21: loop bucle ;Repite hasta terminar
22: mov al, 0 ;No se utiliza XOR para no perder el

AX = 5054
BX = 0000
CX = 0000
DX = 0000
SP = 0100
BP = 0000
SI = 0000
DI = 0000
DS = 5054
ES = 5041
SS = 5055
CS = 5051
IP = 0017

NV UP
EI PL
ZR NA
PE NC

>wb dato2
>ww resul
>Program terminated normally (1)
```

- La **ventana de comandos**, en la parte inferior, en la cual se pueden escribir los diferentes comandos que admite el programa.

LAS TECLAS

- F1 Ayuda del sistema.

- F2 Muestra/Oculto la ventana de registros.
- F3 Alterna la presentación del código fuente, mezclado y modo ensamblado.
- F4 Muestra la ventana de salida.
- F5 Ejecuta el programa hasta el final o hasta el siguiente *punto de parada*.
- F6 Intercambia el cursor entre la ventana de comandos y la ventana principal.
- F7 Ejecuta el programa hasta la posición del cursor en la ventana principal.
- F8 Ejecuta la siguiente línea en modo paso a paso.
- F9 Pone/Quita un punto de parada en la línea actual del cursor.
- F10 Ejecuta la siguiente línea o rutina.
- Ctrl+G Aumenta el tamaño de la ventana en la que está el cursor.
- Ctrl+T Disminuye el tamaño de la ventana en la que está el cursor.
- PgUp Desplaza hacia arriba la ventana sobre la que está el cursor.
- PgDn Desplaza hacia abajo la ventana sobre la que está el cursor.
- Home Desplaza la ventana del cursor al extremo superior de su contenido.
- End Desplaza la ventana actual al extremo inferior de su contenido.
- ↑↓ Mueve el cursor una línea en la dirección de la flecha.

LOS MENÚS

En cada uno de los menús se despliega una ventana de opciones disponibles; sobre la ventana desplegada se puede realizar la selección deseada. Los menús disponibles son:

- **File.** Permite abrir un archivo, salir provisionalmente al DOS y salir del programa.
- **View.** Incluye las opciones para seleccionar el modo en el que se presenta el programa en la pantalla principal, visualizar la ventana de registros y visualizar la pantalla de salida.
- **Search.** Permite realizar búsquedas en el código fuente de la pantalla principal.
- **Run.** Incluye las opciones para ejecutar el programa desde su inicio, inicializar el programa, ejecutar las líneas restantes y borrar todos los puntos de parada.
- **Watch.** Permite añadir líneas de observación de expresiones y puntos de parada condicionales.
- **Options.** Permite seleccionar diversos aspectos de presentación y funcionamiento del programa CodeView.
- **Language.** Permite seleccionar el lenguaje de programación del código fuente.
- **Calls.** Muestra las diferentes llamadas a subrutinas que se han realizado en el programa.
- **Help.** Incluye los distintos capítulos sobre los que el sistema proporciona ayuda.

LAS EXPRESIONES

FORMATOS NUMÉRICOS EN LOS COMANDOS

En algunos comandos se pueden utilizar los siguientes formatos en la representación de datos:

- d ó i Entero decimal con signo.
- u Entero decimal sin signo.
- o Entero octal sin signo.
- x ó X Entero hexadecimal.
- f Real en coma flotante.

- e ó E Real en notación científica.
- g ó G Real (lo más compacto entre e/E ó g/G).
- c Carácter ASCII correspondiente.
- s Cadena finalizada con el carácter nulo.

TIPOS DE DATOS

En algunos comandos se puede seleccionar el modo de introducción o de presentación de los datos, de acuerdo con los siguientes tipos:

- B Bytes en formato hexadecimal.
- A Caracteres ASCII.
- I Números enteros con signo.
- U Números enteros sin signo.
- W Números de 2 bytes en hexadecimal.
- D Números de 4 bytes en hexadecimal.
- S Números de 4 bytes en coma flotante.
- L Números de 8 bytes en coma flotante.
- T Números de 10 bytes en coma flotante.

NÚMEROS DE LÍNEA

Para hacer referencia a alguna de las **líneas del programa en código fuente**, se indicará el número de línea precedida por un punto:

.númerolínea

DIRECCIONES DE MEMORIA

En la línea de comandos las referencias a una **dirección** deben hacerse con el formato:

[segmento:]offset

de modo que si no se indica *segmento*, se toma como segmento por defecto el contenido del registro DS (segmento de datos). El término *offset* se considera expresado en hexadecimal y puede indicarse de forma numérica o mediante una expresión válida (etiquetas o identificadores).

RANGOS DE DIRECCIONES DE MEMORIA

Los **rangos** de direcciones pueden expresarse de cualquiera de los modos siguientes:

dirección_inicial dirección_final
dirección_inicial L número

donde *dirección_inicial* y *dirección_final* representan direcciones o expresiones válidas para una dirección; *número* indica el número de posiciones a las que se hace referencia desde la *posición inicial* (el offset).

LOS COMANDOS

COMANDOS DE EJECUCIÓN

- **Trace** (F8). Ejecuta la línea o instrucción actual entrando en las rutinas, procedimientos o interrupciones. *número* indica cuantas líneas o instrucciones se ejecutarán.

T [*número*]

- **Program step** (F10). Ejecuta la línea o instrucción actual saltando sobre las rutinas, procedimientos o interrupciones. *número* indica cuantas líneas o instrucciones se ejecutarán
P [*número*]
- **Go** (F5). Ejecuta el programa restante hasta el final o hasta el próximo *punto de parada*. *Dirección* es un símbolo, un número de línea o una dirección en el que parará la ejecución.
G [*dirección*]
- **Execute**. Ejecuta el programa restante en modo lento hasta el final o hasta el próximo *punto de parada*
E
- **Restart**. Reinicializa el programa para su ejecución desde el principio.
L

COMANDOS DE EXAMEN DE DATOS Y EXPRESIONES

- **Display expression**. Evalúa el término *expresión* y muestra el resultado en el *formato* expresado opcionalmente.
? *expresión* [,*formato*]
- **Examine symbol**. Muestra los símbolos o etiquetas utilizados y su dirección.
X
- **Dump**. Muestra los datos contenidos en la posición (*dirección*) o posiciones de memoria indicadas (*rango*), expresadas en el *tipo* indicado.
D[*tipo*] [*dirección*|*rango*]
- **Compare memory**. Compara los bytes en la memoria indicada por la expresión *rango* con las correspondientes posiciones a partir de *dirección* (*rango* debe expresar posición inicial y número de posiciones). Como resultado aparecen las posiciones que contienen valores diferentes.
C *rango dirección*
- **Search memory**. Busca en las posiciones de memoria especificadas en *rango* los valores indicados en *lista* y muestra las posiciones en las que se encuentran.
S *rango lista*
- **Port input**. Lee y muestra el byte leído en la dirección de E/S dada por *puerto*.
I *puerto*
- **Register**. Muestra el contenido de todos los registros, la siguiente línea de código que se va a ejecutar, la correspondiente instrucción en ensamblador, la dirección y el código de operación que contiene.
R
- **8087**. Muestra en la ventana de diálogo el contenido de los registros del chip 8087 o del emulador instalado.

COMANDOS DE PUNTOS DE PARADA

- **Break set.** Pone un punto de parada en la línea actual o en la posición de memoria o línea representada por *dirección*; *número* indica las veces que ha de pasar el programa por ese punto para que se produzca la parada. Pueden incluirse uno o varios comandos separados por *punto y coma*, que se ejecutarán tras producirse la parada.

BP [*dirección* [*número*] "*comando*"]]

- **Breakpoint clear.** Borra uno o varios puntos de parada. El término *lista* representa a una relación de números de los puntos de parada (* equivale a todos).

BC [*lista*]

- **Breakpoint disable.** Desactiva temporalmente los puntos de parada incluidos en la relación que representa *lista*.

BD [*lista*]

- **Breakpoint enable.** Activa los puntos de parada incluidos en la relación que representa *lista*.

BE [*lista*]

- **Breakpoint list.** Muestra la lista numerada de los puntos de parada existentes en el programa e informa del estado (*enable/disable*) de cada uno de ellos.

BL

COMANDOS DE VISUALIZACIÓN

- **Watch expression.** Muestra el valor del término *expresión*, que representa a una variable o a una combinación válida de variables y operadores en el *formato* que de forma opcional puede seleccionarse.

W? *expresión* [, *formato*]

- **Watch memory.** Muestra el valor contenido en la posición o posiciones de memoria determinadas por el término *dirección|rango*, de acuerdo con el *tipo* indicado opcionalmente (byte, ascii, word, etc.).

W[*tipo*] *dirección|rango*

- **Watchpoint.** Realiza un *punto de parada* cuando el término *expresión* es evaluado como cierto (distinto de cero), siempre que se trate de una expresión válida.

WP? *expresión*

- **Tracepoint (expresión).** Realiza un *punto de parada* cuando cambia de *expresión*.

TP? *expresión*

- **Tracepoint (memoria).** Realiza un *punto de parada* cuando cambia el valor contenido en la posición o posiciones de memoria determinadas por el término *dirección|rango*.

TP [*tipo*] *dirección|rango*

- **Watch delete.** Borra el punto de parada señalado por el término *número* y establecido con los comandos Watch, Watchpoint o Tracepoint (Y* borra todos).

Y *número*

- **Watch list.** Muestra la lista numerada de los puntos de parada establecidos con los comandos Watchpoint y Tracepoint.

W

COMANDOS DE ANÁLISIS DEL CÓDIGO

- **Set mode.** Cambia el modo de presentación del código en la ventana principal entre el modo fuente (+), modo ensamblado (-) o modo mezclado (&).

S [+|-|&]

- **Unassemble.** Muestra el contenido y el código en ensamblador correspondiente a la *dirección* o al *rango* de direcciones indicado.

U [*dirección*|*rango*]

- **View.** Hace que aparezca en la pantalla principal la línea en la que aparece la *expresión* que se indica o la línea de código señalada (*. línea*).

V [*expresión*]

- **Current location.** Un punto en la línea de comandos hace que aparezca en el centro de la pantalla la próxima línea que se va a ejecutar.

COMANDOS DE MODIFICACIÓN DE CÓDIGO O DATOS

- **Assemble.** Permite ensamblar una instrucciones e introducir los códigos de operación correspondientes a partir de la *dirección* que se señala.

A [*dirección*]

- **Enter.** Carga a partir de la posición de memoria indicada por *dirección*, los datos que se incluyen en el término *lista*. El término *tipo* indica el tipo de dato que será introducido (byte, word, ascii, etc.).

E[*tipo*] *dirección* [*lista*]

- **Fill memory.** Rellena las posiciones de memoria especificadas por *rango* con los valores incluidos en *lista*.

F *rango lista*

- **Move memory.** Copia el contenido del bloque de memoria especificado por *rango* a otro bloque del mismo tamaño que comienza en *dirección*.

M *rango dirección*

- **Port output.** Envía *dato* al puerto de entrada salida cuya dirección es *puerto*.

O *puerto dato*

- **Register.** Muestra el valor contenido en el *registro* indicado y permite cambiar ese valor.

R [*registro* [=] *expresión*]

Pueden utilizarse los siguientes nombres de registros:

AX BX CX DX CS DS SS ES SP BP SI DI IP F

Los bits del registro de estado (F) pueden modificarse individualmente entre cada uno de los dos estados posibles (RF *SET*|*CLEAR*):

| NOMBRE | SET | CLEAR |
|------------------|-----|-------|
| Overflow | OV | NV |
| Dirección | DN | UP |
| Interrupción | EI | DI |
| Signo | NG | PL |
| Cero | ZR | NZ |
| Acarreo auxiliar | AC | NA |
| Paridad | PE | PO |
| Acarreo | CY | NC |

COMANDOS DE CONTROL DEL SISTEMA

- **Help.** Presenta la primera pantalla de ayuda.
H
- **Quit.** Abandona el programa CodeView.
Q
- **Radix.** Cambia la base de numeración en la que se expresan los números para entrar argumentos y visualizar el valor de las expresiones por defecto. Los valores posibles de *número* son 8 (octal), 10 (decimal) y 16 (hexadecimal).
N [*número*]
- **Redraw.** Redibuja la pantalla de CodeView cuando aparecen interferencias en esta causadas por funcionamientos anómalos.
@
- **Screen exchange** (F4). Presenta la ventana de salida de los programas.
\
- **Search.** Sitúa el cursor sobre la línea en la que aparece *expresión*.
/*expresión*
- **Shell escape.** Permite salir temporalmente al sistema operativo DOS. (se regresará tecleando EXIT). Si se incluye *comando* será un comando del DOS. que ejecutará fuera del entorno del programa CodeView de forma temporal.
![*comando*]
- **Tab set.** Establece el *número* de caracteres a los que equivale el código del tabulador .
#*número*
- **Option.** Activa (+) o desactiva (-) una de las opciones (F, B, C, 3) del menú Options.
O[*opcion*[+|-]]