

## PRÁCTICA 5: DEFINICIÓN DE MACROS. OPERACIONES ARITMÉTICAS CON NÚMEROS EN DISTINTAS REPRESENTACIONES

### OBJETIVOS:

1. Introducir al alumno en el uso de macros, como herramienta que facilita la labor de programación y la legibilidad del código fuentes.
2. Definir procedimientos para la realización de operaciones aritméticas de números representados en los distintos formatos.
3. Practicar con llamadas a macros ya definidas en el emulador.
4. Definir macros propias para realizar operaciones aritméticas auxiliares que agilicen la programación.

Las macros estarán definidas dentro de un archivo que se incluye en el programa ensamblador mediante la directiva INCLUDE 'nombre\_archivo.inc'

### CONTENIDOS

1. Definición de macros y estructura. Análisis de las macros frente a los procedimientos
2. Uso del fichero de macros del emulador: emu8086.inc
3. Creación y uso de ficheros de macros propias
4. Ejemplos de procedimientos para operaciones aritméticas

### EJEMPLOS

1. Dado un número entero tipo byte, pasarlo al formato de coma flotante siguiente

|   |         |         |
|---|---------|---------|
| 1 | 5       | 10      |
| s | caract. | mantisa |

**La mantisa es totalmente fraccionaria y en M y S. El exponente se representa polarizado.**

Guardar el número obtenido en el segmento de datos

2. Dado el número D3C0 h representado en el formato de coma flotante anterior, pasarlo a su valor decimal, y mostrarlo por pantalla.
3. Programa para la multiplicación de dos números de 16 bits con signo, los números se introducen como constantes en el segmento de datos. Los resultados se guardan en memoria. Como opción podría presentarse el resultado en pantalla.
3. Definición de macros para el tratamiento de los signos.

### EJERCICIOS:

1. Construir un fichero de macros propio pasando los procedimientos de la practica 5 a macros: leer\_datos, escribir, ascdec, decasc, cursor y borrar\_pantalla
2. Realizar un programa que obtenga los divisores de un número de dos cifras, introducido por teclado, mostrando el mayor y el menor divisor del número.
3. Realizar un programa par sumar números enteros (positivos y negativo), de dos cifras decimales; los números se introducen precediéndolos con el carácter “+” para los positivos y ”-“ para los negativos.