

**ILERNA**

Online

# **Videotutoría 1: Presentación e introducción**

Módulo 05: Entornos de Desarrollo



# INTRODUCCIÓN

## Tema 1: Desarrollo de Software

# OBJETIVOS VT 01

- Definir el desarrollo del SW
- Estructura de un microprocesador (Von Neumann)
- SW basado en distribución
- Etapas de un programa
- Ensamblador

## ¿QUÉ ES EL DESARROLLO DEL SW?

Estudiar las necesidades de un cliente para realizar el programa más adaptado a él siguiendo sus distintas fases, como por ejemplo su análisis, programación, diseño, implementación, pruebas, mantenimiento y evolución, desarrollo del manual correspondiente...

Dentro de mantenimiento y evolución, debemos dar como mínimo un año cuando terminamos un software para que no haya ningún problema entre ni con el software ni el hardware del cliente. También el cliente puede requerir algún tipo de evolutivo. Para añadir una nueva funcionalidad dentro de su aplicativo.

# Arquitectura Von Neumann

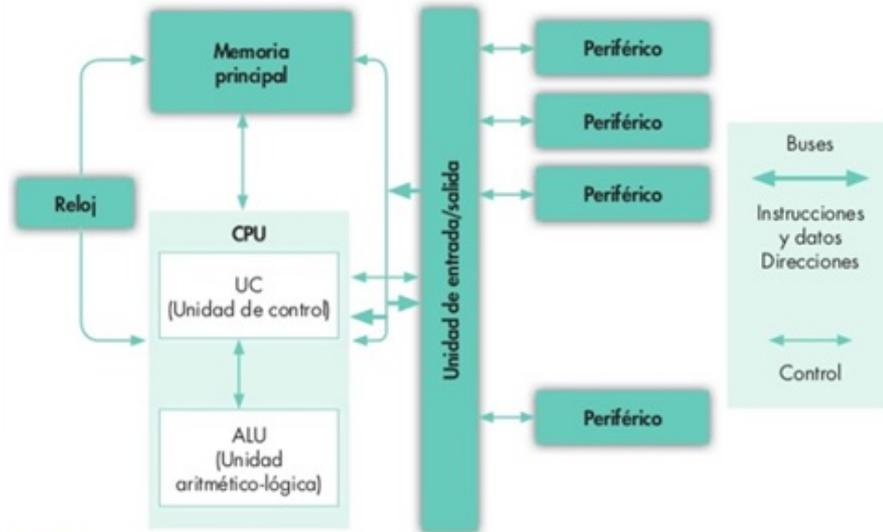


Fig. 2.1. Estructura general de un ordenador con arquitectura Von Neumann.

Montaje y mantenimiento de equipos  
McGraw Hill 2ª Edición

# SW de distribución



**Freeware:** totalmente libres y gratuitos.

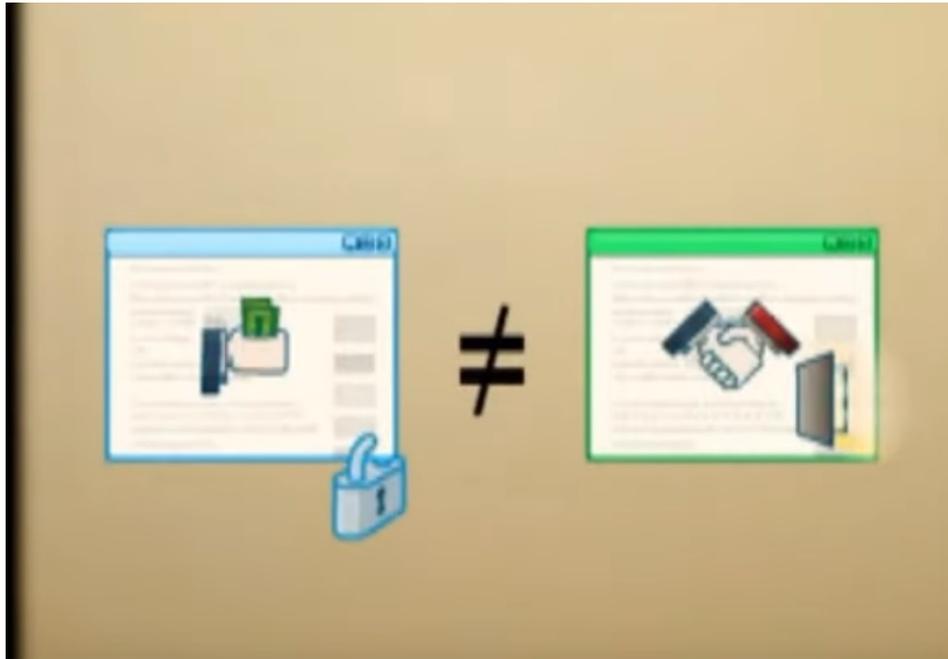
**Shareware:** podemos utilizarlos gratuitamente, pero debemos pagar para desbloquear funcionalidades extra.



**Adware:** tiene mucha publicidad y tampoco tiene presentes todas sus funcionalidades.

# SW Libre vs Proprietario

El software propietario oculta su código.



Software libre permite acceder a su código pudiendo ser modificado por sus usuarios.

# Etapas de un programa

CODIGO FUENTE → CODIGO OBJETOS → CODIGO EJECUTABLE

```
namespace HelloWorld
{
    References
    class Program
    {
        References
        static void Main(string[] args)
        {
            Console.WriteLine("\nWhat is your name? ");
            var name = Console.ReadLine();
            var date = DateTime.Now;
            Console.WriteLine($"Hello, {name}, on {date:d} at {date:t}!");
            Console.WriteLine("\nPress any key to exit...");
            Console.ReadKey(true);
        }
    }
}
```



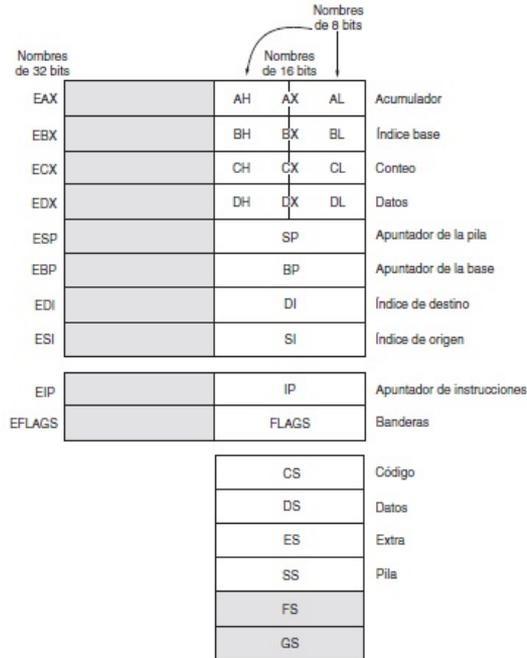
```
0001A1E 4D 4B LDR R3, =(stdout_ptr - 0xC000)
0001A20 E3 58 LDR R3, [R4,R3] ; stdout
0001A22 1B 68 LDR R3, [R3]
0001A24 1B 46 MOV R0, R3 ; stream
0001A26 FF F7 52 EA BLX fileno
0001A28 03 46 MOV R3, R0
0001A2C 1B 46 MOV R0, R3
0001A2E 4B 4B LDR R3, -(a1ChangeDisplay - 0x1
0001A30 7B 44 ADD R3, PC ; "\t) Change displ
0001A32 19 46 MOV R1, R3
0001A34 0B F0 34 FB BL print
0001A38 4B 4B LDR R3, -(stdin_ptr - 0xC000)
0001A3A E3 58 LDR R3, [R4,R3] ; stdin
0001A3C 1B 68 LDR R3, [R3]
0001A3E 1B 46 MOV R0, R3 ; stream
0001A40 FF F7 44 EA BLX fileno
0001A42 02 46 MOV R2, R0
0001A44 07 F5 43 63 ADD.V R3, R2, #0xC30
0001A46 1B 46 MOV R0, R2
0001A4C 19 46 MOV R1, R3
0001A4E AF F0 02 02 MOV.V R2, #2
0001A52 AF F0 00 03 MOV.V R3, #0xA
0001A56 0B F0 77 FB BL read
0001A5A 03 46 MOV R3, R0
0001A5C 00 2B CIP R3, #0
```



ENSAMBLADOR

# ENSAMBLADOR

FIGURA 2-1  
El modelo de programación de los microprocesadores Intel, del 8086 hasta el Pentium 4.



Observaciones:

1. Los registros sombreados sólo existen en los microprocesadores del 80386 hasta el Pentium 4.
2. Los registros FS y GS no tienen nombres especiales.

Programación de los microprocesadores del 8086 al Pentium 4

**Bibliografía:**

Microprocesadores de Intel Barry B. Brey

