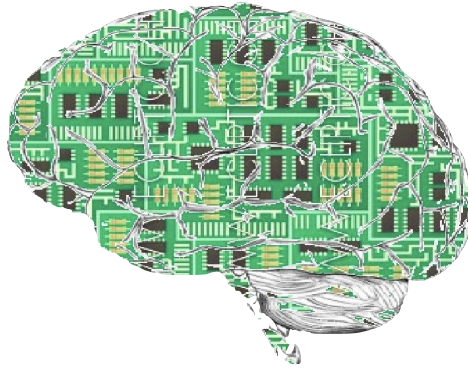
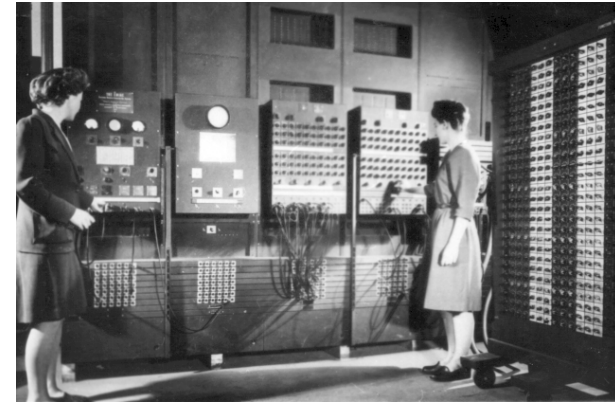


DEPARTAMENTO DE  
COMPUTACIÓN  
UNRC  
2016

## ORGANIZACIÓN DEL PROCESADOR

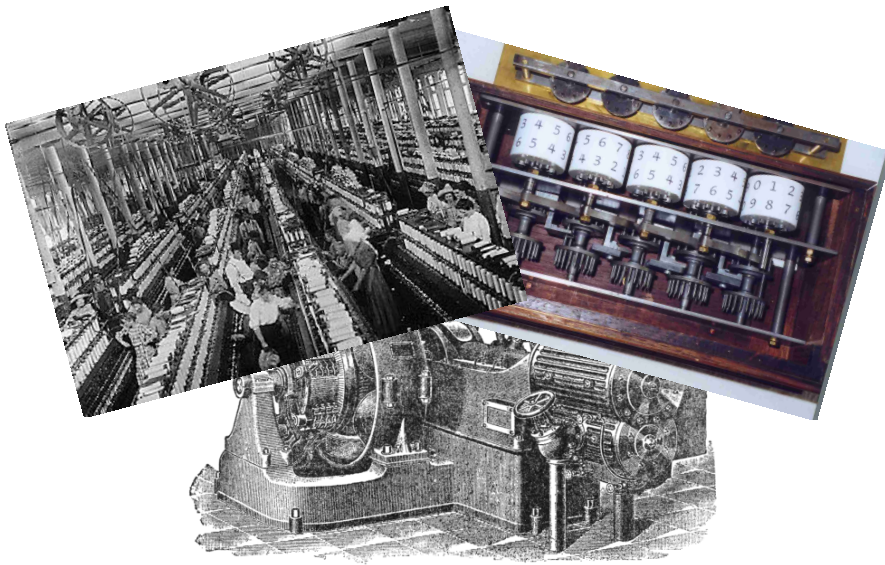


## ORGANIZACIÓN DEL PROCESADOR MAQUINAS DE COMPUTO DE PROPÓSITOS GENERALES



Electronic Numerical Integrator And Computer (ENIAC) - 1946

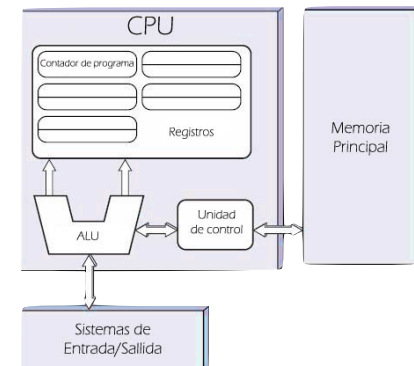
## ORGANIZACIÓN DEL PROCESADOR MAQUINAS DE PROPOSITO ESPECIFICO



## ORGANIZACIÓN DEL PROCESADOR ARQUITECTURA VON NEUMANN



John von Neumann



- **CPU** : Unidad Aritmético Lógica, Unidad de Control, Registros, Program Counter
- **Memoria**
- **Entrada Salida**
- **Capacidad de ejecutar instrucciones secuencialmente**
- **Tiene un único bus entre el CPU, Memoria y Entrada Salida**

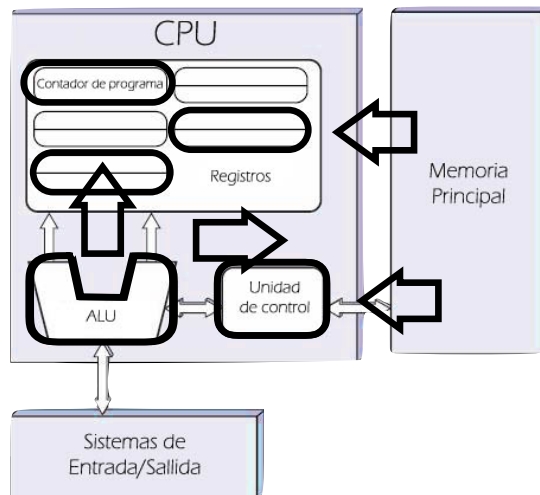
## ORGANIZACIÓN DEL PROCESADOR

### ARQUITECTURA VON NEUMANN (FETCH-DECODE-EXECUTE CYCLE)

**Fetch:** La Unidad de control obtiene de la memoria la próxima instrucción que indica el contador de programa

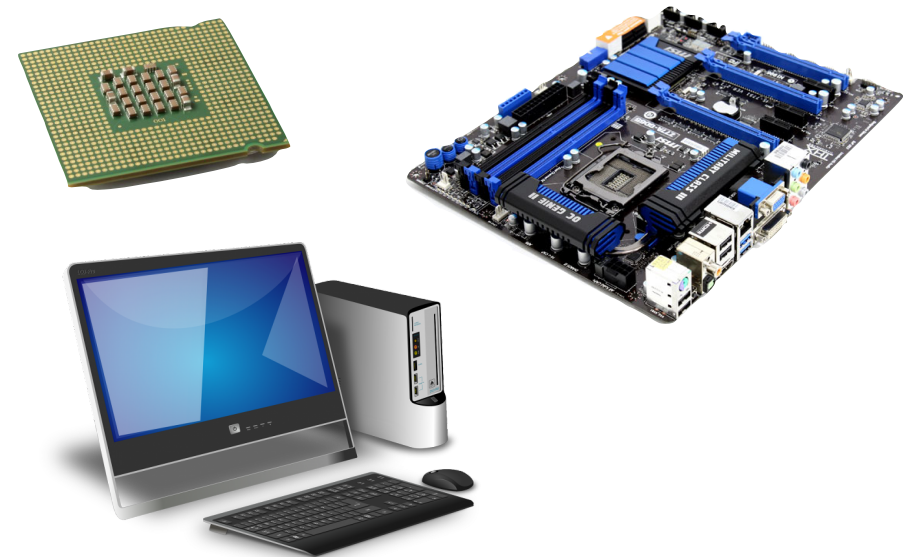
**Decode:** La Unidad de control obtiene de la memoria (si fuere necesario) y descodifica la instrucción para poder ser ejecutada por la ALU

**Execute:** La ALU ejecuta (calcula) el resultado de la operación y lo almacena en un registro o memoria



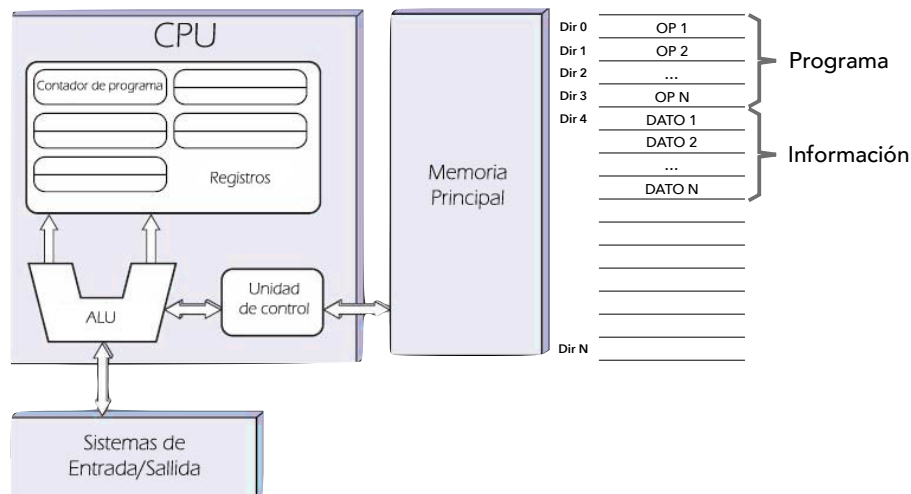
## ORGANIZACIÓN DEL PROCESADOR

### COMPUTADORAS MODERNAS



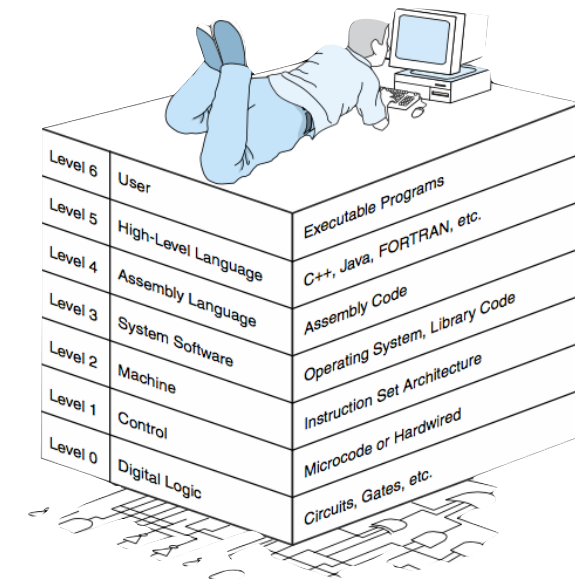
## ORGANIZACIÓN DEL PROCESADOR

### MEMORIA



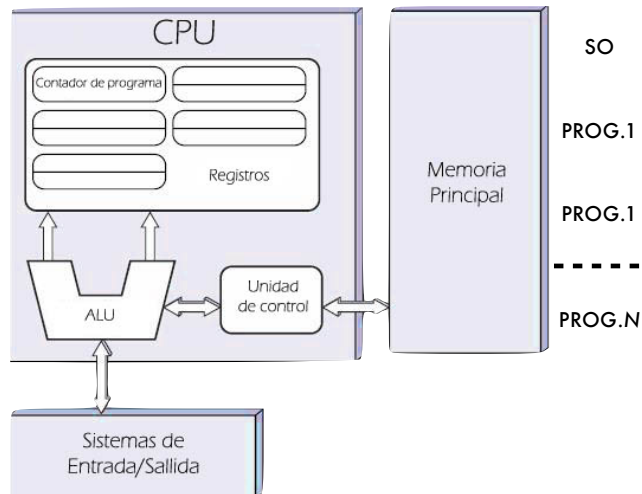
## ORGANIZACIÓN DEL PROCESADOR

### NIVELES ABSTRACTOS DE COMPUTADORAS MODERNAS



## ORGANIZACIÓN DEL PROCESADOR

### SISTEMA OPERATIVO



## ORGANIZACIÓN DEL PROCESADOR

### DIRECCIONES DE MEMORIA, TAMAÑOS Y UNIDADES

Debido al tamaño para la representación de información (enteros, flotantes, caracteres), usualmente se utilizan otros términos para referirnos a tamaños de memoria más grandes que 1 byte.

Unidad	Tamaño	Castellano
<b>byte</b>	8 <b>bits</b>	
<b>word</b>	2 bytes	palabra
<b>double word</b>	4 bytes	palabra doble
<b>quad word</b>	8 bytes	palabra cuádruple
<b>paragraph</b>	16 bytes	párrafo

## ORGANIZACIÓN DEL PROCESADOR

### DIRECCIONES DE MEMORIA, TAMAÑOS Y UNIDADES

Para poder utilizar la memoria debemos darle una identificación (**dirección**). Desde las primeras computadoras de la 4 generación la identificación se realiza agrupando de 8bits (**1 byte**).

Dependiendo de la cantidad de bits con el que trabajo para mantener las direcciones, es la capacidad máxima de memoria que puedo identificar, ej:

**8 bits:** 256 bytes de memoria

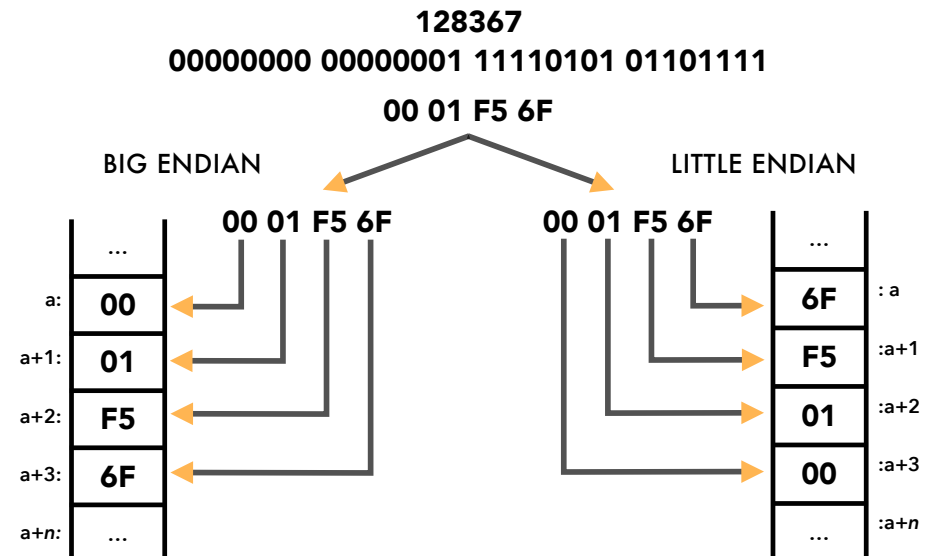
**16 bits:** 65536 bytes (64 Kbytes) de memoria

**32 bits:** 4294967296 bytes (4 Gbytes) de memoria

**64 bits:** 18446744073709551616 bytes (16,777,216 Tbytes) de memoria

## ORGANIZACIÓN DEL PROCESADOR

### LITTLE Y BIG ENDIAN



## ORGANIZACIÓN DEL PROCESADOR

### REGISTROS

Son las unidades de almacenamiento con las que cuenta el procesador para realizar las operaciones. Cada procesador varia en la **cantidad** de registros que contiene y su **tamaño** en bits.

Además existen registros, no sólo para mantener la información con la que se va a operar (**propósitos generales**), sino también para mantener información sobre el proceso, como por ejemplo, registros que contiene **FLAGS**, registros para guardar **direcciones de memoria**, para saber en qué **instrucción del programa** me encuentro, etc.

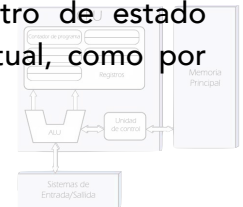
## ORGANIZACIÓN DEL PROCESADOR

### UNIDAD DE CONTROL

Es el módulo encargado de sincronizar todos los componentes para la ejecución de un programa.

Se encarga de extraer una instrucción de programa de la memoria, decodificarla y dejar en los registros correspondientes la información para que otros módulos operen con ella. Comanda el servicio de interrupciones.

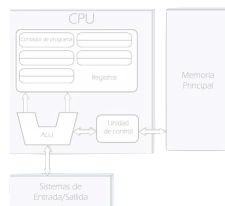
Utiliza un contador de programa (registro) para saber cuál es la próxima instrucción a ejecutar y un registro de estado donde actualiza la situación del programa actual, como por ejemplo si hubo **overflow**, etc.



## ORGANIZACIÓN DEL PROCESADOR

### ALU

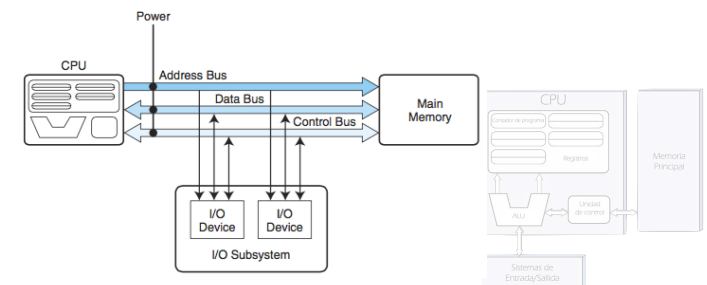
Es el módulo encargado de realizar operaciones aritmético-lógicas. Los procesadores modernos tienen además incorporados módulos más complejos (co-procesadores) para poder realizar eficientemente este tipo de operaciones, como por ejemplo para números racionales representados con punto flotante.



## ORGANIZACIÓN DEL PROCESADOR

### BUS

Todos los componentes de un CPU se encuentran interconectados entre sí mediante el BUS. Existen diferentes tipos de configuraciones de conexión, además en presencia de diferentes conexiones, cada una de ellas con diferentes velocidades.



## ORGANIZACIÓN DEL PROCESADOR

## INTERRUPCIONES

Los componentes interactúan con el procesador mediante interrupciones, además de utilizarse este mecanismo para la ejecución normal, también se utilizan para manejar situaciones anormales. Entre los tipos de interrupciones más comunes podemos mencionar:

- Requerimientos de Entrada/Salida
- Errores aritméticos (división por cero, por ejemplo)
- Overflow o Underflow (fuera de rango de representación)
- Detección de inconsistencia por paridad (memoria)
- Acceso a memoria incorrecto

## ORGANIZACIÓN DEL PROCESADOR

## CONJUNTO DE INSTRUCCIONES - LENGUAJE ENSAMBLADOR

Cada procesador provee un **conjunto de instrucciones** (Lenguaje), si bien este conjunto varia en los diferentes procesadores, en general todo proveen instrucciones para:

- Transferencia de información entre la Memoria y los Registros
- Operaciones Aritmético-lógicas
- Control de Programas
- Transferencia de Entrada Salida

## ORGANIZACIÓN DEL PROCESADOR

## PRIMERA EVOLUCIÓN DE PROCESADORES INTEL

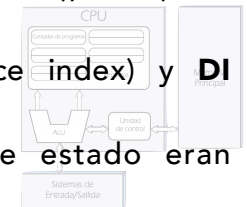
**8086** (1979): disponía de 16 bits para datos y 20 bits para direccionamiento de memoria (1Mbyte).

Tenía 4 registros de propósito general: **AX** (acumulador primario), **BX** (registro base para extensión de direcciones de memoria), **CX** (Contador) y **DX** (Datos). Además estos registros podían utilizarse de mitades AH,BH,CH,DH,AL,BL,CL,DL ("H" high, "L" low ).

Disponía de 3 apuntadores: **SP** (Stack Pointer), **BP** (Base Pointer) y **IP** (Instruction Pointer).

Utilizaba 2 registros especiales **SI** (Source index) y **DI** (Destination index) para operar con cadenas.

Finalmente los diferentes resultados de estado eran reportados en el registro **FLAGS**.



## ORGANIZACIÓN DEL PROCESADOR

## PRIMERA EVOLUCIÓN DE PROCESADORES INTEL

**8086** (cont.): El lenguaje assembler (y su ejecución) que disponía constaba de 3 partes: El **segmento de código** (programa), el **segmento de datos** (datos) y el **segmento pila** utilizado para la ejecución del programa. Para cada uno de ellos disponía de un registro para CS,DS,SS.

Hasta este procesador inclusive la memoria se operaba en **modo real** (no protegido). En este modo un programa puede acceder a cualquier dirección de memoria, ¡aún a la memoria de otros programas!

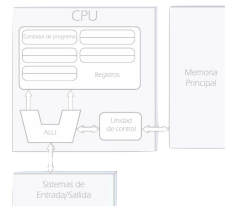




# ORGANIZACIÓN DEL PROCESADOR

## PRIMERA EVOLUCIÓN DE PROCESADORES INTEL

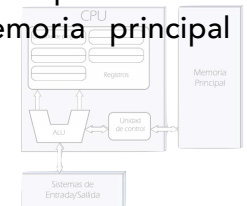
**80286 (1982):** Es una mejora de su antecesor con alguna instrucciones nuevas, además podía direccionar hasta 16 Mbyte de memoria y trabajaba con **modo protegido**.



# ORGANIZACIÓN DEL PROCESADOR

## PRIMERA EVOLUCIÓN DE PROCESADORES INTEL

**80486, Pentium, Pentium II, III, 4, Itanium:** a partir del 486, hasta el itanium, los procesadores evolucionaron haciendo hincapié en la velocidad de realizar operaciones. A partir del Itanium en 2001 se comenzó a utilizar la denominada IA-64 (Arquitectura Intel de 64 bits) y de allí se comenzaron a explotar procesadores con más de un núcleo de ejecución: CoreDuo, Core2Duo, Core i3, i5 e i7. Un aspecto importante fue la incorporación de memoria de altas prestaciones intermedia entre el procesador y la memoria principal denominada **cache**.



# ORGANIZACIÓN DEL PROCESADOR

## PRIMERA EVOLUCIÓN DE PROCESADORES INTEL

**80386 (1985):** En esta versión el gran cambio fue pasar de 16 a 32 bits, pero manteniendo compatibilidad con su antecesor. Los registros fueron extendidos a 32 bits, de allí sus nombres EAX, EBX, ECX ... . El direccionamiento cambió a 32 bits protegido. Este procesador fue la génesis de la arquitectura IA-32 (Intel Architecture 32 bits).

