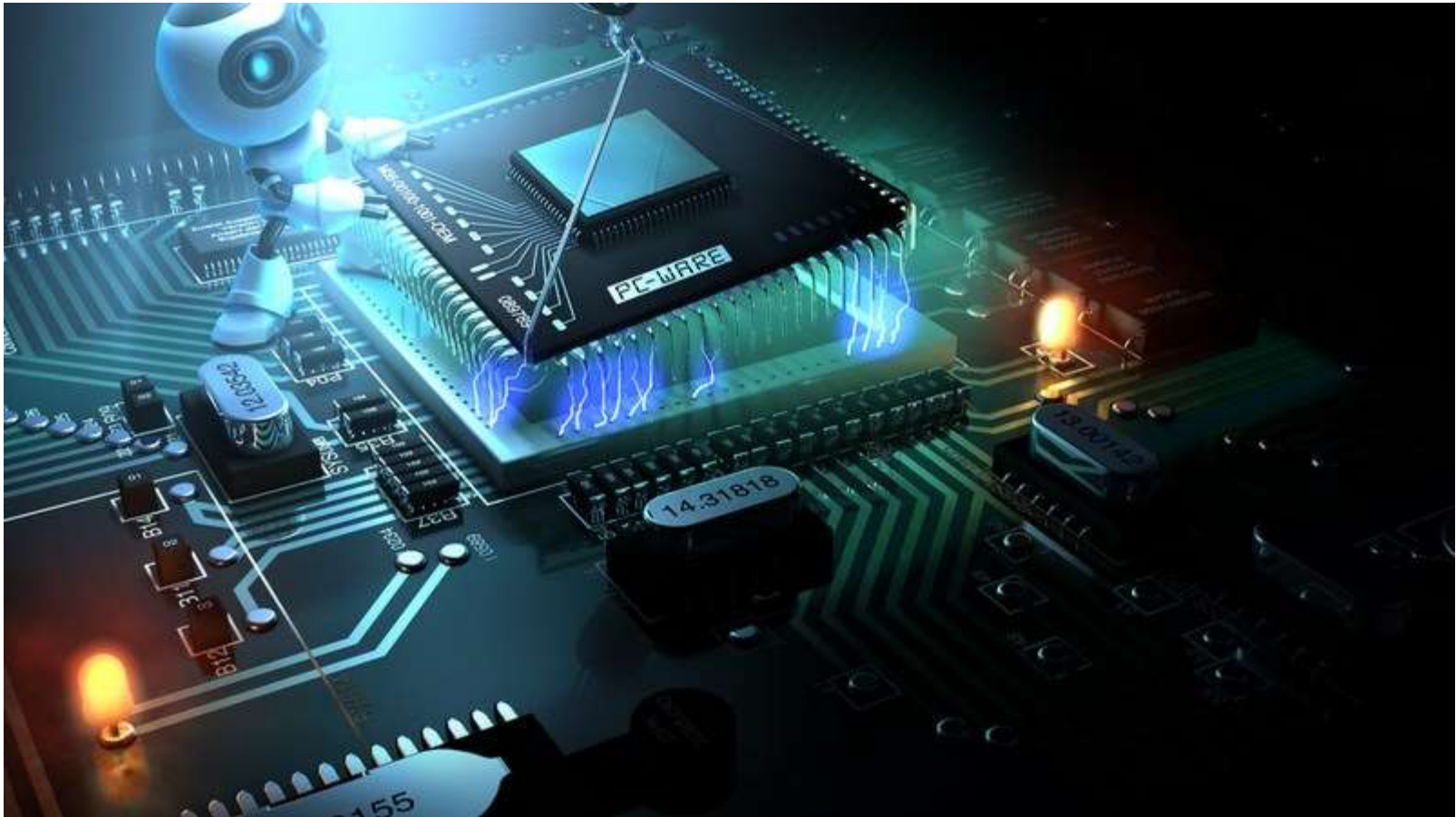


# LENGUAJE ENSAMBLADOR

## (4°B-ISC)



**Mtro. en Ing. Armando Álvarez Fdez.**  
*Ago-2024*

# **EL MODELO DE PROGRAMACIÓN DEL CPU INTEL 8086**

## EJERCICIOS Y NOTAS:



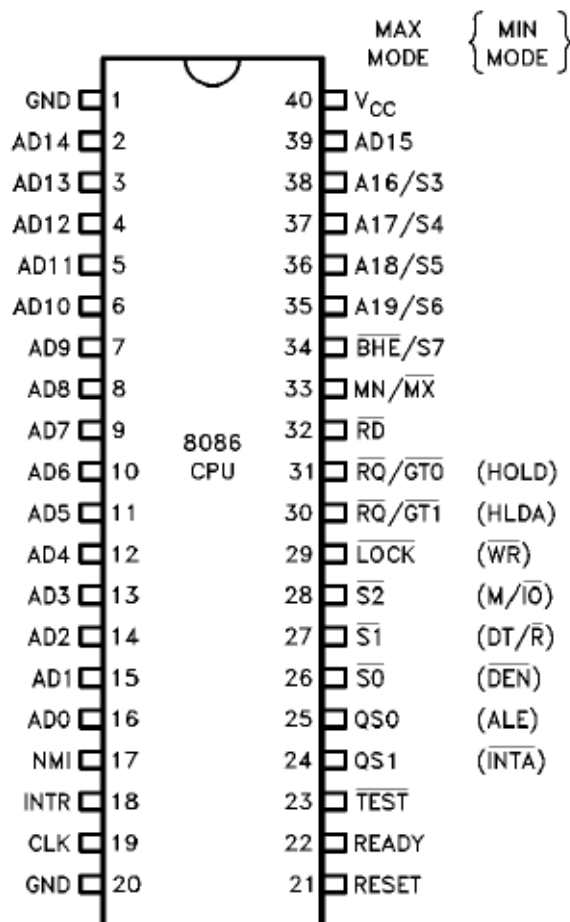
## **8086**

# **16-BIT HMOS MICROPROCESSOR**

### **8086/8086-2/8086-1**

- **Direct Addressing Capability 1 MByte of Memory**
- **Architecture Designed for Powerful Assembly Language and Efficient High Level Languages**
- **14 Word, by 16-Bit Register Set with Symmetrical Operations**
- **24 Operand Addressing Modes**
- **Bit, Byte, Word, and Block Operations**
- **Range of Clock Rates:  
5 MHz for 8086,  
8 MHz for 8086-2,  
10 MHz for 8086-1**
- **8 and 16-Bit Signed and Unsigned Arithmetic in Binary or Decimal Including Multiply and Divide**
- **Available in 40-Lead Cerdip and Plastic Package**

# Chip CPU Intel 8086 y su diagrama a bloques



231455-2

40 Lead

Figure 2. 8086 Pin Configuration

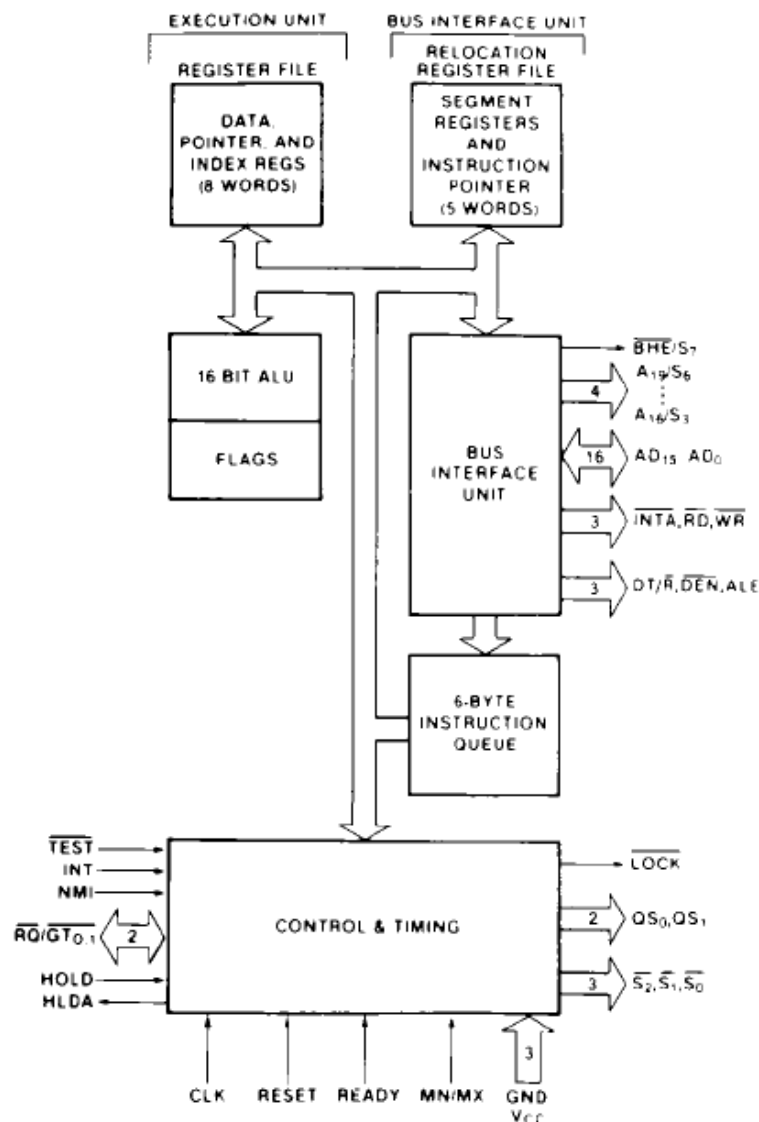


Figure 1. 8086 CPU Block Diagram

231455-1

# DESCRIPCION GENERAL DEL 8086 y 8088

El 8086 es un microprocesador de 16 bits, tanto en lo que se refiere a su estructura como en sus conexiones externas, mientras que el 8088 es un procesador de 8 bits que internamente es casi idéntico al 8086. La única diferencia entre ambos es el tamaño del bus de datos externo. Intel trata esta igualdad interna y desigualdad externa dividiendo cada procesador 8086 y 8088 en dos sub-procesadores. O sea, cada uno consta de una unidad de ejecución (EU: Execution Unit) y una unidad interfaz del bus (BIU: Bus Interface Unit). La unidad de ejecución es la encargada de realizar todas las operaciones mientras que la unidad de interfaz del bus es la encargada de acceder a datos e instrucciones del mundo exterior.

## EXPLICACION =

— Consta de 6 Partes.

— Se agrupan en 3 Bloques básicos

- EU (Execution Unit): Realiza las Ops. Lógicas y Aritméticas y calcula Direcc. efectivas-
- BIU (Bus Interface Unit): Manipula los buses de datos y direcciones y controlar cola de Instrucciones.
- Control = "Señales de Control y Temporiz. del CPU" (Técnicas)

— El CPU tiene 2 modos Fundamentales de operación

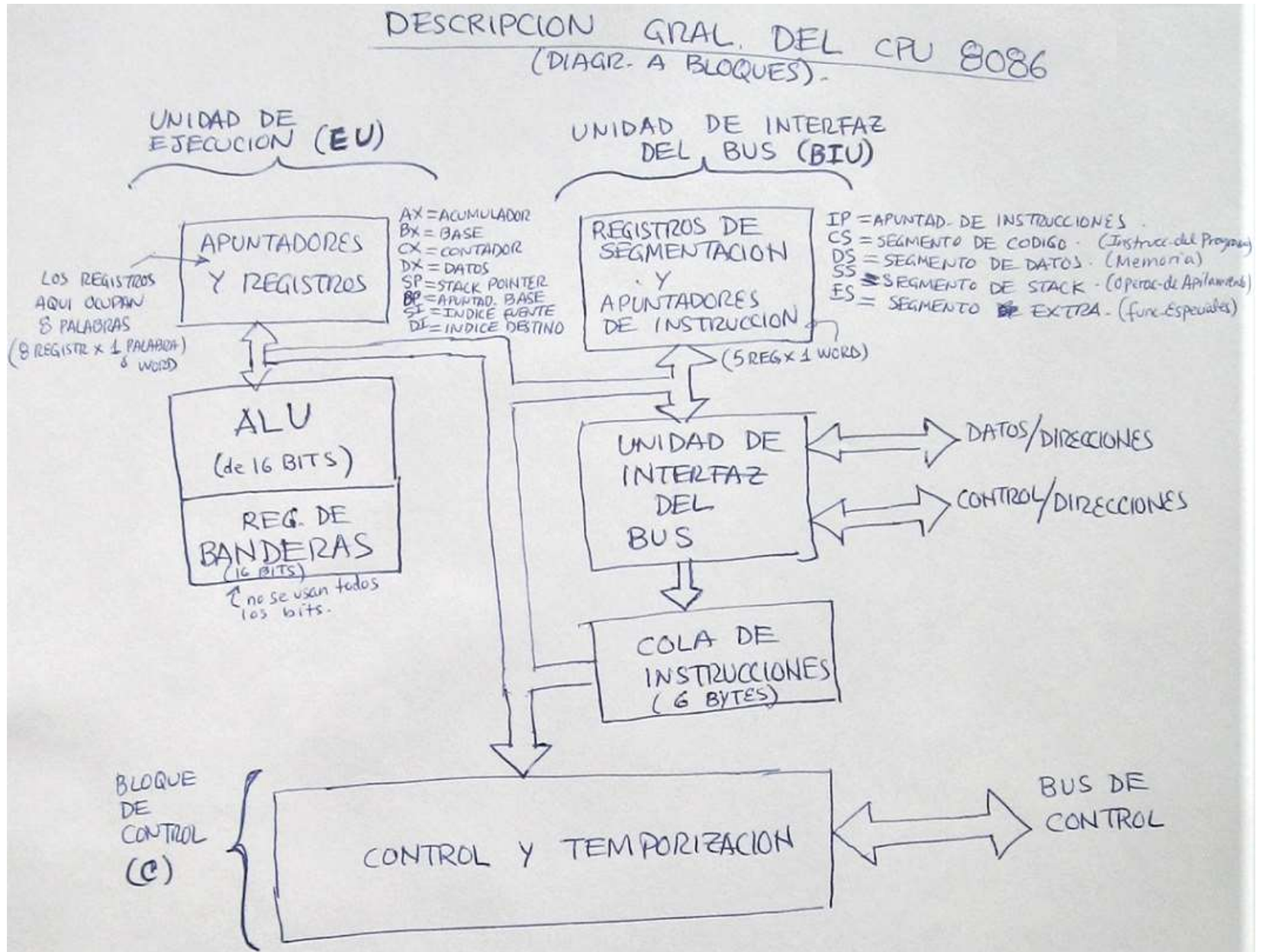
{ \* Se ajusta el modo con el <sup>pin</sup> entrada PIN (33) llamado  $\overline{MN}/\overline{MX}$

• MODO MÁXIMO = Modo de Hardware que maneja más señales. Por ejemplo, para interactuar con un Coprocesador Matem. 8087 y Arbitraje del Bus.

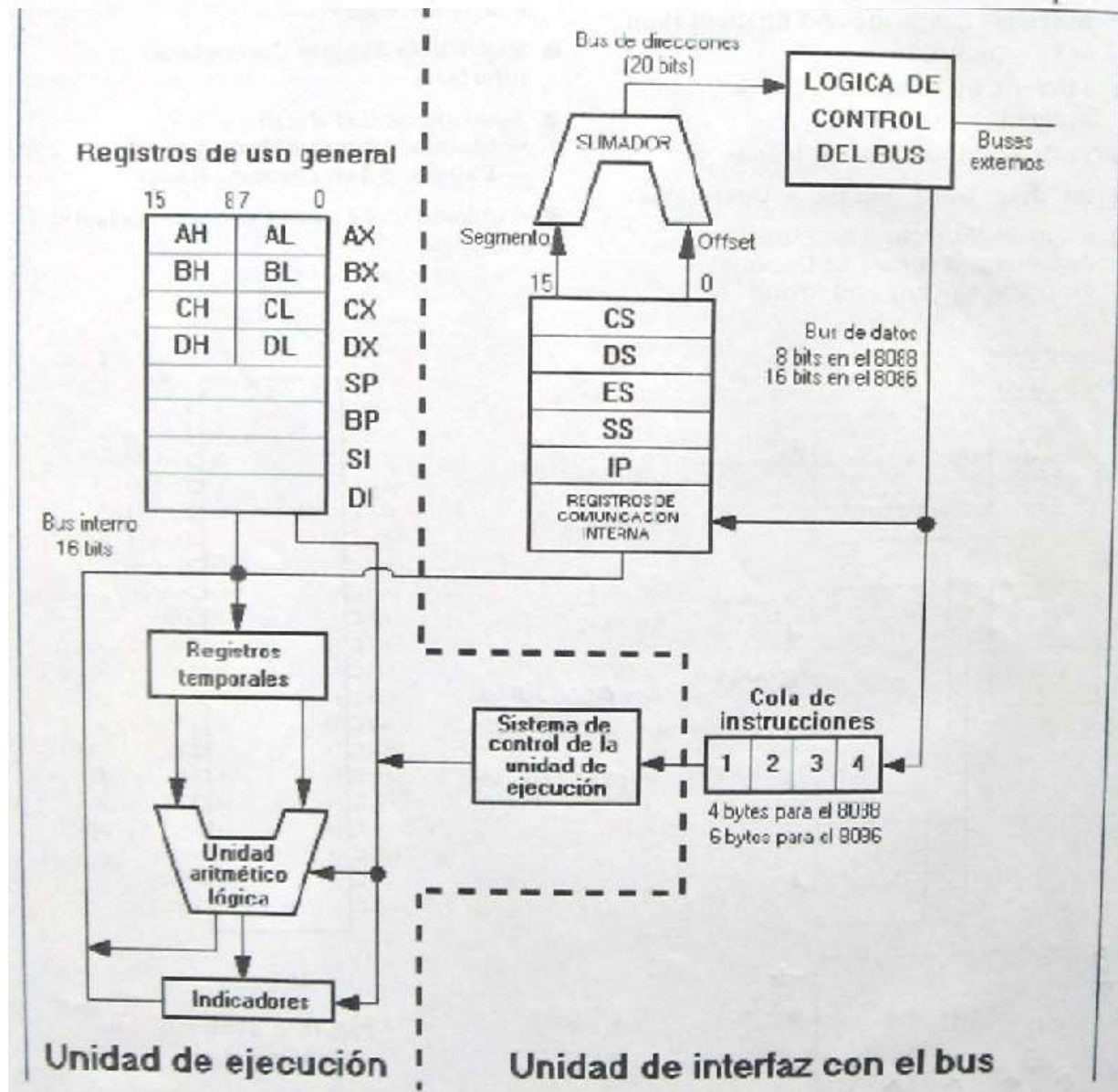
• MODO MÍNIMO = Es un modo Básico de Sistema Mínimo menos sofisticado en Hardware



# DESCRIPCION GENERAL DEL 8086 y 8088



Las unidades de ejecución son idénticas en ambos microprocesadores, pero las unidades de interfaz del bus son diferentes en varias cuestiones, como se desprende del siguiente diagrama en bloques:

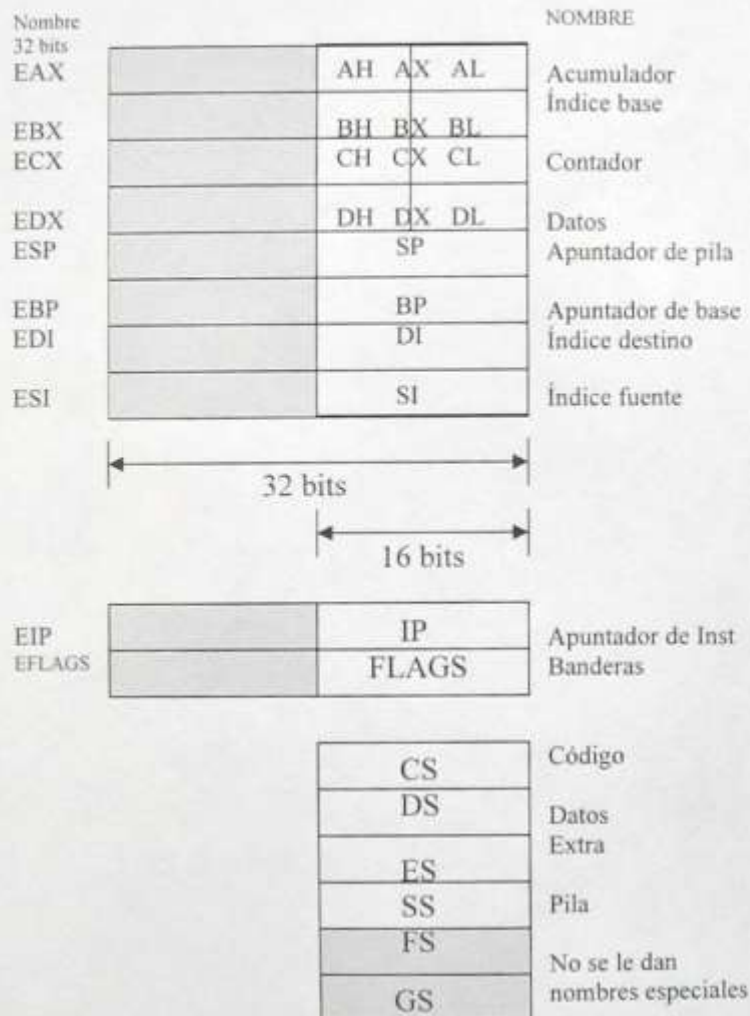


La ventaja de esta división fue el ahorro de esfuerzo necesario para producir el 8088. Sólo una mitad del 8086 (el BIU) tuvo que rediseñarse para producir el 8088.



## 2. El modelo de programación.

Se describe la estructura de los registros del microprocesador y se explica la forma en que se direcciona la memoria por medio de los registros de segmentos y de los desplazamientos de dirección.



# MODELO DE PROGRAMACION DEL 80x86 (SUS REGISTROS)

NOTA: Las áreas sombreadas EXISTEN SOLAMENTE EN LOS CPUS INTEL 80386, EN ADELANTE.

OBSERVAR QUE EL MODELO DE PROGRAMACION TIENE REGISTROS DE 8, 16 y 32 BITS.

\* (Ej. Sig. se puede cambiar por redundancia)

REGISTROS DE 8 BITS:

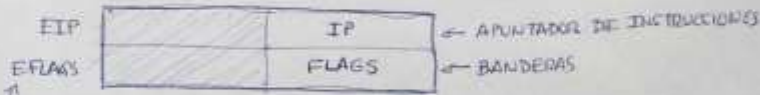
{AH, AL, BH, BL, CH, CL, DH, DL}

REGISTROS DE 16 BITS:

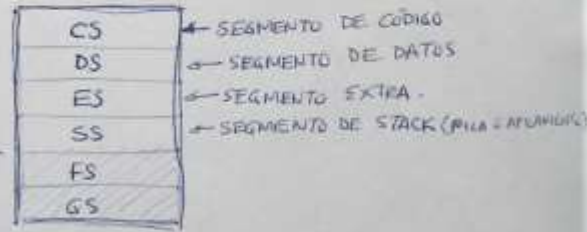
{AX, BX, CX, DX, IP, BP, DI, SI, IF, FLAGS, CS, DS, ES, SS, FS, GS}

REGISTR. EXTENDIDOS de 32 BITS:

{EAX, EBX, ECX, EDX, ESP, EBP, EDI, ESI, EIP, EFLAGS}



E, es por registro extendido o 32 bits



LOS REGISTROS FS Y GS NO TIENEN UN NOMBRE ESPECIFICO

- EXISTEN REGISTROS DE:
- DE PROPOSITO GRAL & MULTIPROPOSITO: {EAX, EBX, ECX, EDX, EBP, EDI, ESI} → Utilizados para muchos propósitos
  - DE PROPOSITO ESPECIAL: {ESP, EIP, EFLAGS, CS, DS, ES, SS, FS, GS} → Tienen una finalidad especifica o mas especializados en algo.

# NOMBRES E INICIALES DE LOS REGISTROS DEL 8086 EN INGLES

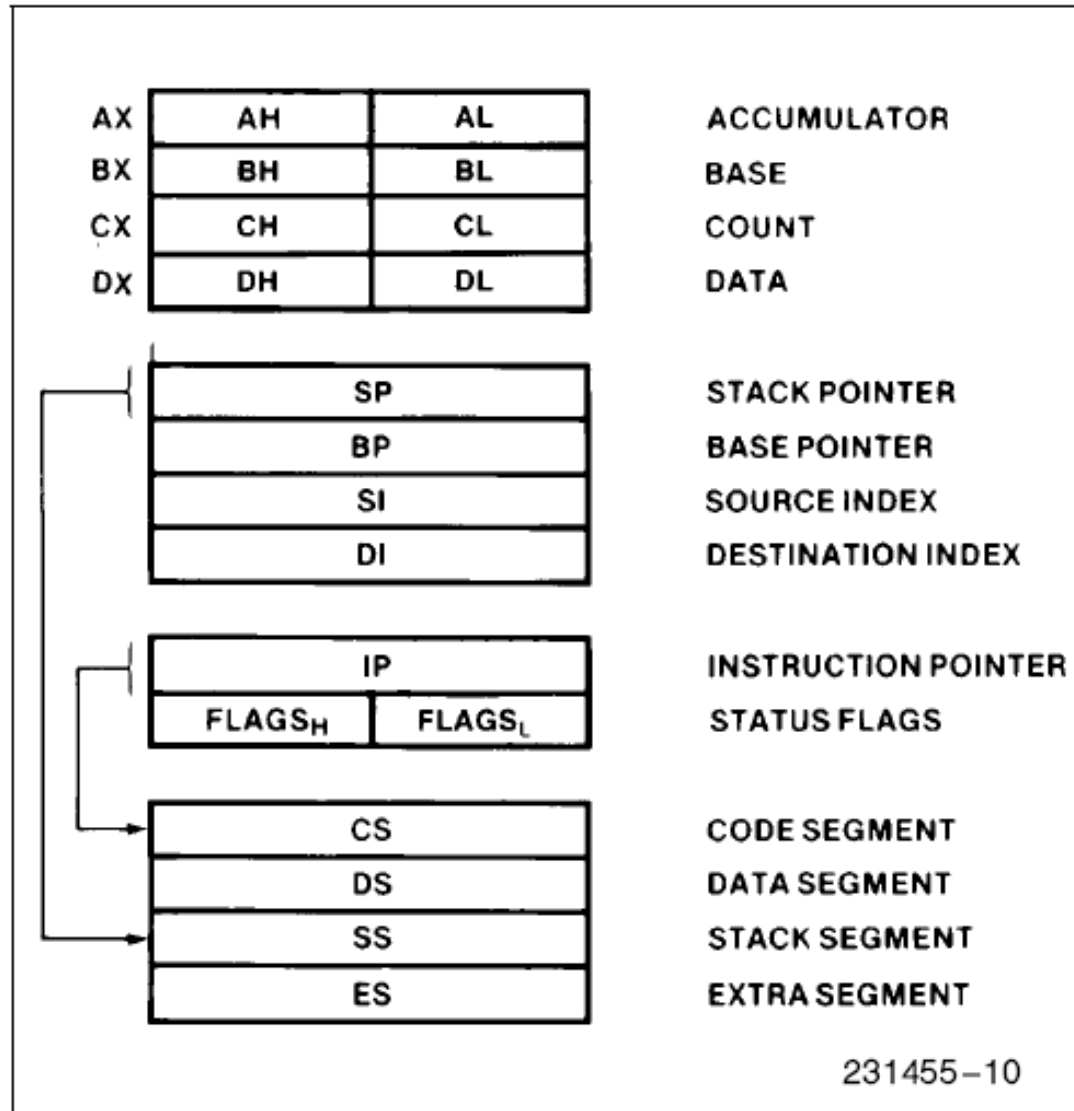


Figure . 8086 Register Model

### ***3. Registros del microprocesador***

#### **Registros de propósito general.**

Los *registros de propósito general* se utilizan en la forma en que se desee el programador. Cada registro para uso general se puede direccionar como un registro de 32 bits (EAX, EBX, ECX y EDX), como un registro de 16 bits (AX, BX, CX y DX) o como uno de 8 bits (AH,AL,BH,BL,CH,CL,DH yDL). Se debe tener en cuenta que solo el 80386 en adelante contienen el grupo de registros de 32 bits. Las funciones principales de los registros de propósito general incluyen:

***AX (Acumulador):*** Es el acumulador primario, posee dos características diferentes de los otros acumuladores. Todas las operaciones de I/O deben pasar por la parte baja (AL) de este acumulador, las instrucciones que utilizan este registro gastan menos espacio y se ejecutan más rápido.

***BX (Base):*** Es un acumulador de propósito general que puede ser utilizado para calcular direcciones, a menudo conserva la dirección base (desplazamiento) de los datos que hay en memoria. Si la memoria es accesada con este registro los programas serán más cortos y correrán más rápidos.

***CX (Contador):*** Puede ser utilizado para el conteo de ciertas instrucciones para corrimientos (CL) y rotaciones del número de bytes (CX) para las operaciones repetidas de cadenas y un contador (CX o ECX) para la instrucción LOOP.

***DX (Datos):*** Las instrucciones de I/O lo utilizan para definir la dirección del periférico, algunas multiplicaciones y divisiones también lo necesitan.

### **Registros apuntadores e índices.**

***SP (Apuntador de pila):*** Se utiliza para direccionar datos en una pila de memoria LIFO, esto ocurre más a menudo cuando se ejecutan las instrucciones PUSH y POP, cuando se llama CALL o cuando se regresa RET de una subrutina.



***BP (Apuntador de base):*** Es un apuntador que se utiliza para referenciar parámetros, especialmente cuando se programa en un ambiente de múltiples lenguajes.

***SI (Índice de fuente):*** Se emplea para direccionar datos fuente en forma indirecta para utilizarlos con las instrucciones de cadenas o arreglos.

***DI (Índice destino):*** Se suele emplear para direccionar datos destino en forma indirecta para utilizarlos con las instrucciones de cadenas o arreglos.

***IP (Apuntador de instrucciones):*** Se utiliza siempre para direccionar a la siguiente instrucción que va a ejecutar el microprocesador. Esta relacionado con el registro CS que es el segmento de código. Para formar la localidad real de la siguiente instrucción se suma el contenido de IP con CS por 10H. Corresponde al contador de programas (Program Counter o PC) de otros microprocesadores.

## Registros con segmentos.

Unos registros adicionales, a los que se da el nombre de registros de segmentos, generan direcciones en la memoria junto con otros registros en el microprocesador. A continuación aparece una lista de cada registro de segmento junto con su función en el sistema:

**CS (Código):** El segmento de código es una sección en la memoria que tiene los programas y procedimientos utilizados por los programas. El registro de segmento de código define la dirección inicial de la sección de memoria que tiene el código (*Es un registro de 16 bits que define el segmento de las instrucciones*).

**DS (Datos):** El segmento de datos es una sección en la memoria que contiene la mayor parte de los datos utilizados por un programa, (esto datos pueden ser variables, vectores, matrices, etc).

**ES (Extra o adicional):** El segmento extra o adicional de datos lo utilizan algunas instrucciones para cadenas.

**SS (Pila):** El segmento de pila define la superficie de la memoria utilizada para la pila. La ubicación del punto inicial de entrada a la pila, se determina por el registro apuntador de la pila. El registro BP también direcciona los datos que hay dentro del segmento de pila.

**FS y GS:** Estos registros de segmento adicionales están disponibles en los microprocesadores 80386 en adelante a fin de contar con dos segmentos adicionales de memoria para acceso con los programas.



## EJERCICIOS Y NOTAS:



# **APENDICES**



## **8086**

# **16-BIT HMOS MICROPROCESSOR**

### **8086/8086-2/8086-1**

- Direct Addressing Capability 1 MByte of Memory
- Architecture Designed for Powerful Assembly Language and Efficient High Level Languages
- 14 Word, by 16-Bit Register Set with Symmetrical Operations
- 24 Operand Addressing Modes
- Bit, Byte, Word, and Block Operations
- 8 and 16-Bit Signed and Unsigned Arithmetic in Binary or Decimal Including Multiply and Divide
- Range of Clock Rates:
  - 5 MHz for 8086,
  - 8 MHz for 8086-2,
  - 10 MHz for 8086-1
- MULTIBUS System Compatible Interface
- Available in EXPRESS
  - Standard Temperature Range
  - Extended Temperature Range
- Available in 40-Lead Cerdip and Plastic Package

(See Packaging Spec. Order #231369)

# Chip CPU Intel 8086 y su diagrama a bloques

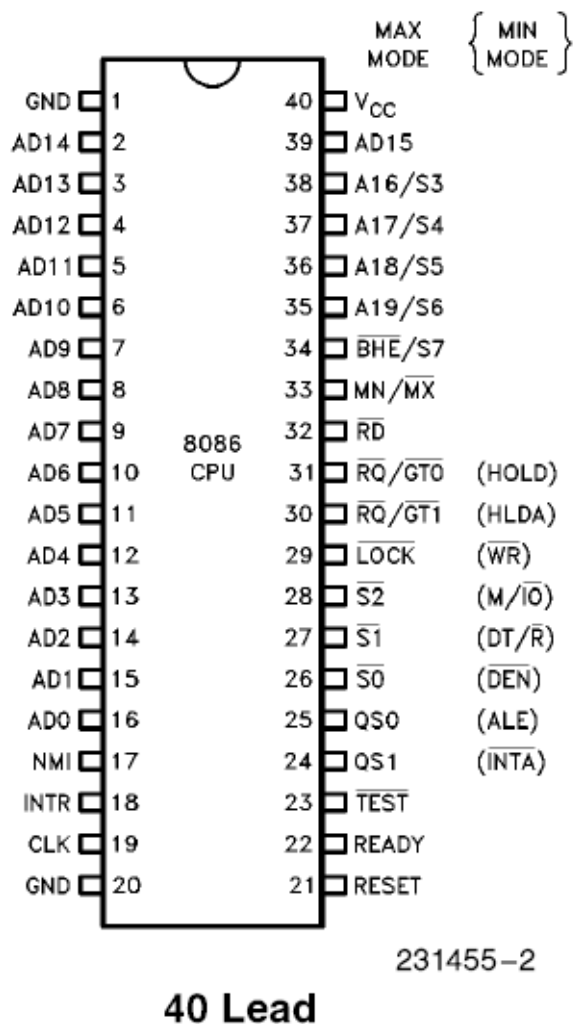


Figure 2. 8086 Pin Configuration

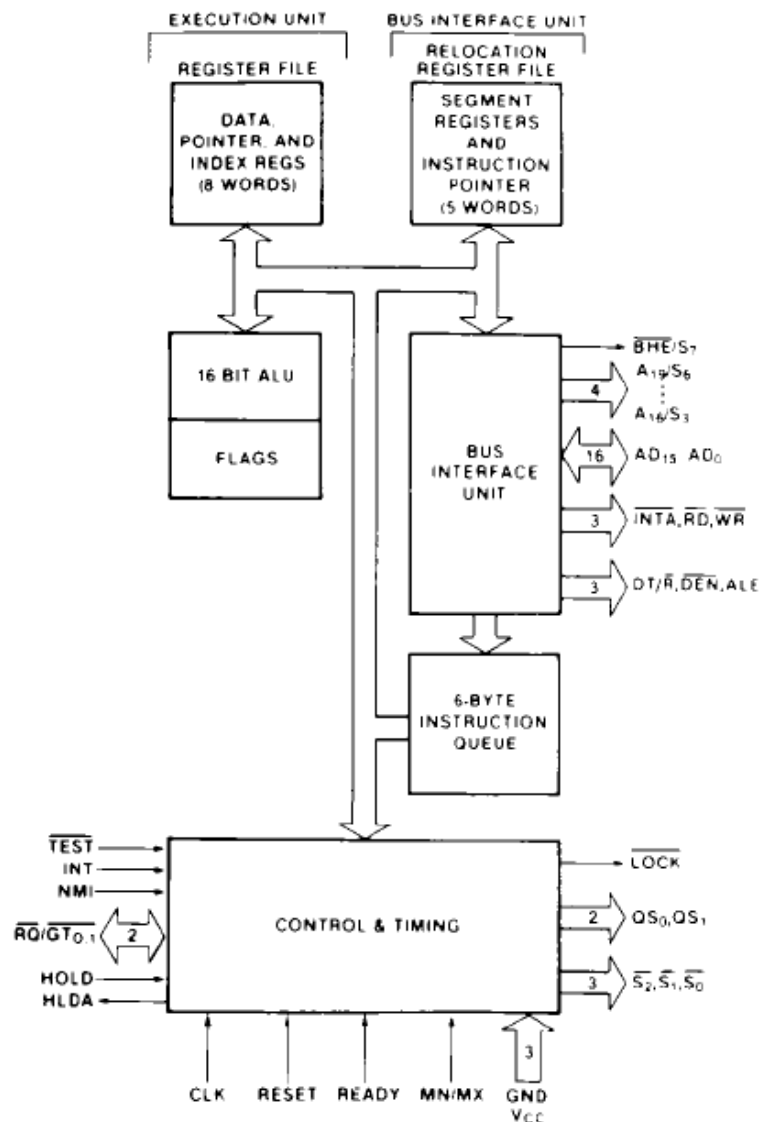


Figure 1. 8086 CPU Block Diagram

# REGISTROS DE LA UNIDAD DE EJECUCIÓN DE INSTRUCCIONES (EU).

Tienen 16 bits cada uno y son ocho:

1. **AX** = Registro acumulador, dividido en **AH** y **AL** (8 bits cada uno). Usándolo se produce (en general) una instrucción que ocupa un byte menos que si se utilizaran otros registros de uso general. Su parte más baja, **AL**, también tiene esta propiedad. El último registro mencionado es el equivalente al acumulador de los procesadores anteriores (8080 y 8085). Además hay instrucciones como **DAA**; **DAS**; **AAA**; **AAS**; **AAM**; **AAD**; **LAHF**; **SAHF**; **CBW**; **IN** y **OUT** que trabajan con **AX** o con uno de sus dos bytes (**AH** o **AL**). También se utiliza este registro (junto con **DX** a veces) en multiplicaciones y divisiones.
2. **BX** = Registro base, dividido en **BH** y **BL**. Es el registro base de propósito similar (se usa para direccionamiento indirecto) y es una versión más potente del par de registros **HL** de los procesadores anteriores.
3. **CX** = Registro contador, dividido en **CH** y **CL**. Se utiliza como contador en bucles (instrucción **LOOP**), en operaciones con cadenas (usando el prefijo **REP**) y en desplazamientos y rotaciones (usando el registro **CL** en los dos últimos casos).

# REGISTROS DE LA UNIDAD DE EJECUCIÓN DE INSTRUCCIONES (EU).

4. **DX** = Registro de datos, dividido en **DH** y **DL**. Se utiliza junto con el registro **AX** en multiplicaciones y divisiones, en la instrucción **CWD** y en **IN** y **OUT** para direccionamiento indirecto de puertos (el registro **DX** indica el número de puerto de entrada/salida).

5. **SP** = Puntero de pila (no se puede subdividir).

Aunque es un registro de uso general, debe utilizarse sólo como puntero de pila, la cual sirve para almacenar las direcciones de retorno de subrutinas y los datos temporarios (mediante las instrucciones **PUSH** y **POP**). Al introducir (push) un valor en la pila a este registro se le resta dos, mientras que al extraer (pop) un valor de la pila este a registro se le suma dos.

6. **BP** = Puntero base (no se puede subdividir). Generalmente se utiliza para realizar direccionamiento indirecto dentro de la pila.



## EJERCICIOS Y NOTAS:

