

Facultad de la Energía, las Industrias y los Recursos
Naturales no Renovables
CARRERA DE INGENIERÍA EN SISTEMAS

Lenguaje Ensamblador

Instrucciones

René Guamán-Quinche
rguaman@unl.edu.ec

Agenda

Instrucciones

- Instrucciones de Transferencia de Datos
- Instrucciones de Control de Bucles (instrucciones simples)
- Instrucciones de Prueba, Comparación y Saltos
- Instrucciones de Llamado y Retorno de Subrutinas
- Instrucciones Aritméticas
- Instrucciones Lógicas
- Instrucciones de Desplazamiento, Rotación y Adeudos
- Instrucciones de Pila
- Instrucciones de Control del Microprocesador
- Instrucciones de Interrupción

Instrucciones

Instrucciones de Transferencia de Datos

- Estas instrucciones mueven datos de una parte a otra del sistema; desde y hacia la memoria principal, de y a los registros de datos, puertos de E/S y registros de segmentación
- Las instrucciones de transferencia de datos son las siguientes:

Instrucciones

Instrucciones de Transferencia de Datos

- **MOV** transfiere
- **XCHG** intercambia
- **IN** entrada
- **OUT** salida
- **XLAT** traduce usando una tabla
- **LEA** carga la dirección efectiva
- **LDS** carga el segmento de datos
- **LES** carga el segmento extra
- **LAHF** carga los indicadores en **AH**
- **SAHF** guarda **AH** en los indicadores
- **PUSH FUENTE** $(sp) \leftarrow \text{fuente}$
- **POP DESTINO** $\text{destino} \leftarrow (sp)$

Instrucciones

Control de Bucles (instrucciones simples)

Éstas posibilitan el grupo de control más elemental de nuestros programas. Un bucle es un bloque de código que se ejecuta varias veces. Hay 4 tipos de bucles básicos:

- Bucles *sin fin*
- Bucles por *conteo*
- Bucles *hasta*
- Bucles *mientras*

Instrucciones

Control de Bucles (instrucciones simples)

Las instrucciones de control de bucles son las siguientes:

- **INC** incrementar
- **DEC** decrementar
- **LOOP** realizar un bucle
- **LOOPZ, LOOPE** realizar un bucle *si es cero*
- **LOOPNZ, LOOPNE** realizar un bucle *si no es cero*
- **JCXZ** salta si **CX** es *cero*

Instrucciones

Instrucciones de Prueba, Comparación y Saltos

Este grupo es una continuación del anterior, incluye las siguientes instrucciones:

• TEST	verifica
• CMP	compara
• JMP	salta
• JE, JZ	salta si es igual a cero
• JNE, JNZ	salta si no igual a cero
• JS	salta si signo negativo
• JNS	salta si signo no negativo
• JP, JPE	salta si paridad par
• JNP, JOP	salta si paridad impar
• JO	salta si hay capacidad excedida
• JNO	salta si no hay capacidad excedida

Instrucciones

Instrucciones de Prueba, Comparación y Saltos

Este grupo es una continuación del anterior, incluye las siguientes instrucciones:

- **JB, JNAE**
- **JNB, JAE**
- **JBE, JNA**
- **JNBE, JA**
- **JL, JNGE**
- **JNL, JGE**
- **JLE, JNG**
- **JNLE, JG**

salta	si <i>por abajo</i> (no encima o igual)
salta	si <i>no está por abajo</i> (encima o igual)
salta	si <i>por abajo o igual</i> (no encima)
salta	si <i>no por abajo o igual</i> (encima)
salta	si <i>menor que</i> (no mayor o igual)
salta	si <i>no menor que</i> (mayor o igual)
salta	si <i>menor que o igual</i> (no mayor)
salta	si <i>no menor que o igual</i> (mayor)

Instrucciones

Instrucciones de Llamado y Retorno de Subrutinas

Para que los programas resulten eficientes y legibles tanto en lenguaje ensamblador como en lenguaje de alto nivel, resultan indispensables las subrutinas:

- **CALL** *llamada* a subrutina
- **RET** *retorno* al programa o subrutina que llamó

Instrucciones

Instrucciones Aritméticas

a. Grupo de adición:

- **ADD**
- **ADC**
- **AAA**
- **DAA**

suma

suma con acarreo

ajuste **ASCII** para la suma

ajuste decimal para la suma

b. Grupo de sustracción:

- **SUB**
- **SBB**
- **AAS**
- **DAS**

resta

resta con acarreo negativo

ajuste **ASCII** para la resta

ajuste decimal para la resta

Instrucciones

Instrucciones Aritméticas

c. Grupo de multiplicación:

- *MUL*
- *IMUL*
- *AAM*

multiplicación

multiplicación entera

ajuste **ASCII** para la multiplicación

d. Grupo de división:

- *DIV*
- *IDIV*
- *AAD*

división

división entera

ajuste **ASCII** para la división

Instrucciones

Instrucciones Aritméticas

e. Conversiones:

- **CBW** pasar octeto a palabra
- **CWD** pasar palabra a doble palabra
- **NEG** negación

f. Tratamiento de cadenas:

Permiten el movimiento, comparación o búsqueda rápida en bloques de datos:

- **MOVC** transferir carácter de una cadena
- **MOVW** transferir palabra de una cadena
- **CMPC** comparar carácter de una cadena
- **CMPW** comparar palabra de una cadena
- **SCAC** buscar carácter de una cadena
- **SCAW** buscar palabra de una cadena
- **LODC** cargar carácter de una cadena
- **LODW** cargar palabra de una cadena
- **STOC** guardar carácter de una cadena
- **STOW** guardar palabra de una cadena

Instrucciones

Instrucciones Aritméticas

f. *Tratamiento de cadenas:*

Permiten el movimiento, comparación o búsqueda rápida en bloques de datos:

- *MOVC* transferir *carácter* de una cadena
- *REP* repetir
- *CLD* poner a 0 el indicador de dirección
- *STD* poner a 1 el indicador de dirección

Instrucciones

Instrucciones Lógicas

Son operaciones bit a bit que trabajan sobre octetos o palabras completas:

- **NOT** negación
- **AND** producto lógico
- **OR** suma lógica
- **XOR** suma lógica exclusiva

Instrucciones

Instrucciones de Desplazamiento, Rotación y Adeudos

Básicamente permiten multiplicar y dividir por potencias de 2

- **SHL, SAL** **desplazar** a la **izquierda** (desplazamiento aritmético)
- **SHR** **desplazar** a la **derecha**
- **SAR** **desplazamiento** aritmético a la **derecha**
- **ROL** **rotación** a la **izquierda**
- **ROR** **rotación** a la **derecha**
- **RCL** **rotación** con acarreo a la **izquierda**
- **RCR** **rotación** con acarreo a la **derecha**
- **CLC** **borrar** acarreo
- **STC** **poner** acarreo a 1

Instrucciones

Instrucciones de Pila

Una de las funciones de la pila del sistema es la de salvaguardar (conservar) datos (la otra es la de salvaguardar las direcciones de retorno de las llamadas a subrutinas):

- *PUSH* introducir
- *POP* extraer
- *PUSHF* introducir indicadores
- *POPF* extraer indicadores

Instrucciones

Instrucciones de Control del microprocesador

Hay varias instrucciones para el control de la CPU, ya sea a ella sola, o en conjunción con otros procesadores:

- | | |
|---------------|--------------|
| • <i>NOP</i> | no operación |
| • <i>HLT</i> | parada |
| • <i>WAIT</i> | espera |
| • <i>LOCK</i> | bloquea |
| • <i>ESC</i> | escape |

Instrucciones

Instrucciones de Interrupción

Hay varias instrucciones para el control de la CPU, ya sea a ella sola, o en conjunción con otros procesadores:

- *STI* poner a 1 el indicador de interrupción
- *CLI* borrar el indicador de interrupción
- *INT* interrupción
- *INTO* interrupción por capacidad excedida (desbordamiento)
- *IRET* retorno de interrupción

Las instrucciones del ensamblador

Instrucciones de transferencia

- Son utilizadas para mover los contenidos de los operandos
- Cada instrucción se puede usar con diferentes modos de direccionamiento

MOV

MOVS (MOVSB) (MOVSW)

Las instrucciones del ensamblador

Instrucciones de transferencia

Instrucción MOV

Propósito: Transferencia de datos entre celdas de memoria, registros y acumulador

Sintaxis:

MOV Destino,Fuente

Donde Destino es el lugar a donde se moverán los datos y fuente es el lugar donde se encuentran dichos datos.

Las instrucciones del ensamblador

Instrucciones de transferencia

Instrucción MOV

Los diferentes movimientos de datos permitidos para esta instrucción son:

- entre registros generales
 - MOV ax, bx
 - MOV eax, edi
 - MOV esi, ecx
 - MOV ebx, eax
- entre registros generales y celdas de memoria
 - MOV eax, [variable]
 - MOV eax, [ebx]
 - MOV eax, [ebx + edi + 100h]
 - MOV [variable], eax

Las instrucciones del ensamblador

Instrucciones de transferencia

Instrucción MOV

Los diferentes movimientos de datos permitidos para esta instrucción son:

- MOV inmediato a memoria
 - MOV eax, 100h
 - MOV ecx, variable
 - MOV edi, 1
- MOV registros generales a registros de segmento
 - MOV cx, ds
 - MOV bx, cs

Las instrucciones del ensamblador

Instrucciones de transferencia

Instrucción MOV

Ejemplo:

```
MOV AX,0006h  
MOV BX,AX  
MOV AX,4C00h
```