

Introducción al NASM

The Netwide Assembler

MIA J. Rafael Rojano C.

1

Convenciones de colores

- Rojo= Directivas propias del nasm sin traducción directa a ensamblador
- Morado= Directivas del ensamblador
- En fondo naranja la sintaxis general
- En fondo azul los ejemplos concretos

MIA J. Rafael Rojano C.

2

Esqueleto básico

```
[section .text]
```

```
..start:
```

```
    ;colocar código
```

```
    mov ah,4ch
```

```
    int 21h
```

```
[section .data]
```

```
    ;variables inicializadas
```

MIA J. Rafael Rojano C.

3

Pseudo-op

- **ORG 100h**
 - Define la dirección de “origen”. En el caso de programas **com** la dirección es **100h**
- **SECTION .text**
 - Define el inicio de un grupo de instrucciones para ser ensambladas en un código ejecutable.
- **SECTION .data**
 - Define el inicio de un grupo de declaraciones que son inicializadas en el segmento de datos.
- **SECTION .bss**
 - Define el inicio de un grupo de declaraciones no inicializadas.

El uso de la directiva SECTION puede ser intercalada en cualquier orden, ya que cuando el código se compila todas las secciones son concatenadas.

Primero .text, luego .data y .bss. (la sección.bss no genera código en realidad pero el proceso de reservación de bloques de memoria provoca que las etiquetas asociadas tomen los valores correctos).

MIA J. Rafael Rojano C.

4

Etiquetas

- Tanto en los lenguajes de alto nivel como en los de bajo nivel, se puede hacer referencia a las posiciones (direcciones) de memoria mediante un símbolo.
- Tales símbolos se llaman etiquetas. Así pues tendremos dos tipos de etiquetas esencialmente:

Etiquetas

- 1) Etiquetas para hacer referencia a posiciones dentro del programa (líneas)
- 2) Etiquetas para hacer referencia a variables del programa

MIA J. Rafael Rojano C.

5

Declarando etiquetas(1)

- En el primer caso (líneas de código) la definición de una etiqueta se hace mediante un nombre que termine en dos puntos, por ejemplo:
ciclo:
salto:
- En el segundo caso (variables) las etiquetas se declaran mediante un nombre, la instrucción DB y el valor de inicialización.
- **Recordar:** *cada declaración ya sea variable o línea de código esta haciendo referencia a una posición de la memoria particular.*

MIA J. Rafael Rojano C.

6

Declarando etiquetas(2)


- Veamos ejemplos ahora de la declaración de variables. No debemos olvidar que la declaración de las “variables” se hace en el segmento de datos
- El formato general de la declaración es:

Nombre DB lista

- Donde nombre es la etiqueta que usaremos como una “variable”
- DB define el tamaño del dato a asignar (pueden usarse DW, DQ o DT)
- Y lista, representa el valor o valores con los que se inicializa

Ejemplo:

Nombre DB “juanito”, 0dh, 0ah ; CR y LF



MIA J. Rafael Rojano C.

7

Declarando etiquetas(3)

- Los valores que son asignados a la variable pueden ser numéricos o cadenas
- Para los numéricos el default es decimal, pero si se desea otra base se usan:
 - h para hexadecimal
 - q para octal
 - b para binario
- Los datos alfanúmericos se colocan entre apostrofes ´ o comillas ”

MIA J. Rafael Rojano C.

8

Ejemplos de distintas declaraciones

- Nombre db "juan perez" ; declara una cadena de longitud 10, ya que cada letra ocupa un byte
- Edad db 15h ; aquí juan tiene 21 puesto que la h indica decimal
- Sexo db 'h' ; sexo ocupa un byte de longitud
- Direccion db "av. Xalapa s/n" ; otra cadena
- Contador db 0 ; contado se inicializa a cero
- Aux dw 0 ; aux ocupa dos byte ya que es word la declaración
- arreglo1 db 0,0,0,0,0 ;este arreglo ocupa 5 bytes, uno por cada cero
- arreglo2 dw 1,2,3,4,5 ;este arreglo ocupa 10 bytes, dos por cada digito, ya que se declararon como word

MIA J. Rafael Rojano C.

9

Reserva de espacio

- Cuando se trata de declarar espacio para cadenas muy grandes no tiene caso que enumeremos todos los bytes, en su lugar usamos RESB o RESW
- RESB reserva un bloque de memoria de tamaño de N bytes, ejemplos:

Var1 resb 8 ;reserva 8 bytes para var1

Var2 resb 4 ;reserva 4 bytes para var2

- También se pueden reservar bytes de tamaño word con la directiva RESW

Var3 resw 4 ;reserva 8 bytes

MIA J. Rafael Rojano C.

10

Declaración de constantes

- No olvidar que comunmente en la declaración también se usan constantes. Para ello se emplea la instrucción EQU
- La sintaxis de EQU es similar a la de DB, solo que en lugar del db o dw se pone EQU, ejemplo:

Etiqueta EQU valor

- En conjunto a la declaración de constantes, el nasm define dos pseudo operaciones que permiten el cálculo de direcciones estas dos constantes especiales son \$ y \$\$, a continuación se explican

MIA J. Rafael Rojano C.

11

Constantes especiales(1)

- El \$ define la posición de la dirección de memoria al inicio de donde lo coloquemos, ejemplo:

```
msj db "pepito"  
tamaño EQU $ - msj
```

Ya que \$ regresa la posición en donde éste se encuentra; si realizamos una resta del valor que tiene \$ menos la posición de la etiqueta msj,

lo que obtendremos como resultado es la longitud de la declaración msj. En este caso el resultado lo asignamos a la variable tamaño. Tamaño es igual a 6

MIA J. Rafael Rojano C.

12

Constantes especiales(2)

- De forma similar, el uso de la constante \$\$ regresa la dirección segmento en donde incluyamos esta constante. Por ejemplo:

```
var1 db "variable 1$"  
var2 db "variable 2$"  
valor db 10  
espacio EQU $ - $$
```

- En este caso lo que se obtiene como resultado es la longitud de espacio ocupada por todo el segmento hasta antes la declaración de la variable espacio. Para este ejemplo el valor es **23d** o **17h**. Un byte por cada caracter

MIA J. Rafael Rojano C.

13

Compilación

- Para compilar en un ambiente de windows se opta por tener dos formatos de salida:
 - Los **obj** que representan aplicaciones típicas .exe de 16 o 32 bits
 - Los **bin** que representan programas típicos como los .com
- Para compilar desde la línea de comandos se hace tecleando:

```
nasm -f obj archivo.asm
```

MIA J. Rafael Rojano C.

14

Ligado

- Un proceso necesario para la generación de archivos ejecutables es el ligado a través del cual se agregan todos los elementos necesarios para el ambiente de ejecución del programa.
- La compilación en línea de comandos se hace tecleando:
alink archivo.obj
- Como resultado obtenemos un archivo con extensión .exe

MIA J. Rafael Rojano C.

15

Ejemplos de programa com

```
:[File]: uniseg.asm
:[Author]: J. Rafael Rojano C.
;NASM-IDE ASM Assistant Assembler Project File
[BITS 16] ;Set code generation to 16 bit mode
[ORG 0x0100] ;Set code start address to 0000h

[SEGMENT .text] ;Main code segment
mov dx, msj
mov ah,09
int 21h

int 20h

[SEGMENT .data] ;Initialised data segment
msj db 'simple mensaje$'

[SEGMENT .bss] ;Uninitialised data segment
```

MIA J. Rafael Rojano C.

16

Ejemplos de programa exe

```
:[File]: firstexe.asm
:[Author]: J. Rafael Rojano C.
segment code
..start:
mov ax,data
mov ds,ax
mov dx,msg
mov ah,9
int 21h

mov ah,4ch
int 21h

segment data
msg db 'hola muchachos...$'
```

MIA J. Rafael Rojano C.

17

Diferencia de los com y exe(1)

- En los ejemplos anteriores las diferencias principales aparecen en negritas y con fuente color azul.
- Los com inician en la posición 100h
- Los exe emplean la directiva **..start:** para marcar el punto de inicio que puede ser 0
- Los com finalizan el programa mediante la int 20
- Los exe emplean la función 4ch de la interrupción 21

MIA J. Rafael Rojano C.

18

Diferencia de los com y exe(2)

- Otra diferencia substancial es como se compilan y ligan:
 - Los com solo se compilan
 - Nasm -f bin hola.asm
 - Los exe se compilan y ligan.
 - Nasm -f obj hola.asm
 - Alink hola.obj

MIA J. Rafael Rojano C.

19

Desde la interfase

- Si trabajas compilando desde la interfase debes hacer dos adecuaciones importantes para compilar los com y los exe
- Para ello deberás cambiar el tipo de archivo de salida desde el menú options/assembler y luego en la sección de target elegir la salida adecuada

MIA J. Rafael Rojano C.

20

Donde esta el menú

Tranquilo, aquí esta el menú options, luego buscas el submenú Assembler



MIA J. Rafael Rojano C.

21

Seleccionar target para com

Del paso anterior abrirá esta ventana, aquí eliges la opción com executable



MIA J. Rafael Rojano C.

22

Si el target es exe

Del paso anterior abrirá esta ventana, aquí eliges la opción sys device o bien Dos 16 para generar el archivo exe



MIA J. Rafael Rojano C.

23