Herramientas Básicas De Programación Assembler

Arancibia Márquez Deysi [1]

[1] Departamento de Informática y Sistemas U.A.J.M.S.

Correo electrónico: pdam@uajms.edu.bo

RESUMEN

Realizar un programa en ensamblador es una tarea de todos los informáticos que estamos en contacto diario con un ordenador, un programa assembler está asociado a las funciones de un microprocesador que es el cerebro de una computadora.

El lenguaje ensamblador es un lenguaje de bajo nivel, sencillo en su programación, no obstante esta sencillez en su sintaxis dirige a crear programas con muchas líneas de código, con la evolución de los lenguajes de alto nivel, se logra solucionar este problema a través de la combinación de lenguajes de alto nivel y bajo nivel.

Se precisa del lenguaje ensamblador por las ventajas y características que posee como ser : Rapidez en la ejecución de procesos, ya que trabaja directamente con el microprocesador y ocupar menos espacio de memoria.

En este artículo se mencionan las herramientas básicas que existen para programar en assembler así como los editores, compiladores, enlazadores, mostrando las aplicaciones en MASM y Debug.

PALABRAS CLAVE : Ensamblador, microprocesador, editores, depuradores, compiladores, MASM,Debug.

INTRODUCCIÓN

El lenguaje ensamblador o código simbólico (en inglés Assembly language) es el lenguaje más básico, sin embargo el más complejo, posee una notación derivada del lenguaje de máquina. El lenguaje ensamblador se inventó para facilitar la tarea de los primeros programadores que hasta ese momento tenían que escribir directamente en código binario, se introducen seudoinstrucciones llamados también mnemónicos, para mejorar la legibilidad del programa, posteriormente surgieron los lenguajes de alto nivel que cambiaron la sintaxis de programación a expresiones directamente entendidas por los programadores. Programar en assembler significa trabajar directamente con el microprocesador; por lo cual se debe de conocer el funcionamiento interno de este, los programas en ensamblador ocupan menos espacio en memoria.

Entre las ventajas de los programas en ensamblador podemos citar:

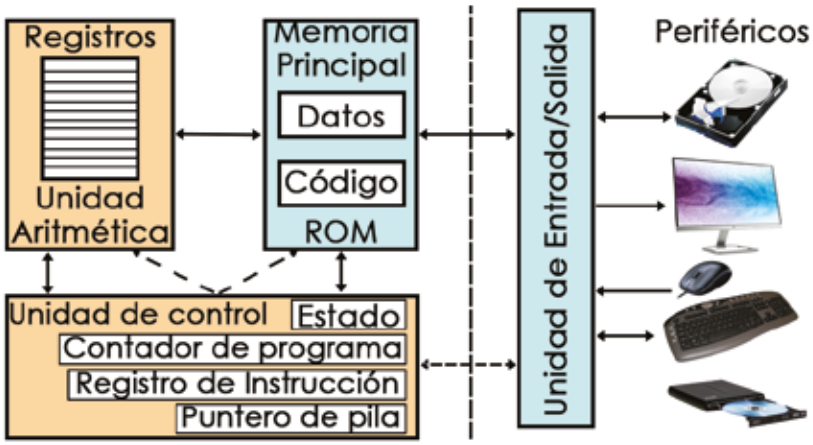
- Es veloz, ya que trabaja directamente con el microprocesador al ejecutar un programa
- Ocupa menos espacio en memoria que los lenguajes de alto nivel.
- Es posible explotar al máximo los recursos de la computadora.

Algunas desventajas son:

- Al ser un lenguaje de bajo nivel requiere más instrucciones para realizar un programa, por lo que los programas fuentes son grandes.
- Dificulta el mantenimiento de programas

COMPONENTES INTERNOS DEL MICROPROCESADOR:

Figura1: *Componentes del Micropocesador*



1. Memoria:
En la memoria se almacena información en celdas especiales llamados registros.

2. Unidad Aritmética Lógica UAL:
La unidad Aritmética Lógica es la responsable de realizar operaciones aritméticas y lógicas.

3. Unidad de Control UC:
Se encarga de coordinar que los otros componentes ejecuten las operaciones correctamente.

4 Unidades periféricas:
Son los dispositivos de Entrada y Salida que ingresan instrucciones o reciben información procesada

5 Registros Internos del microprocesador:
Son utilizados por la memoria principal y son 14 registros:

- 4 Registros de Datos o de almacenamiento temporal : Cada registro tiene 2 byte superior e inferior
- 4 Registros de Segmentos: Contienen la dirección de comienzo de ciertos segmentos de memoria.
- 2 registros puntero de la pila.
- 2 registros índices: Se utilizan como desplazamiento relativo a un campo de datos.
- 1 puntero de instrucción
- 1 registro bandera: Sirven parar indicar el estado actual de la máquina y el resultado del procesamiento, Cuando algunas instrucciones piden comparaciones o cálculos aritméticos cambian el estado de las banderas.

Las banderas están en el registro de banderas en las siguientes posiciones: bits

HERRAMIENTAS DE PROGRAMACIÓN ASSEMBLER

- Editor para introducir código
- Un ensamblador
- Un enlazador
- Depuradores

1. Editores:

Entre los **editores** tenemos:

- Dos: edit, assembler editor, etc.
- Windows: bloc de notas, visual assembler (en fase de desarrollo)
- Linux: emacs

1. Ensambladores:

Las herramientas ensambladores traducen el código fuente al lenguaje máquina (Código objeto no ejecutable).

Entre sus variantes tenemos:

- Tasm: usado bajo dos.
- Masm: usado bajo dos y windows.
- Nasm: usado bajo dos, windows y linux.

2. Enlazador:

La herramienta encargada de tomar el código objeto generado por el ensamblador, añadir los encabezados apropiados y producir un archivo ya ejecutable es el conocido como **linker o enlazador.**
Las variantes son:

- Masm: link,
- Tasm: tlink
- Nasm: no tiene un enlazador propio pero puede utilizar el alink.

a. Depuradores:

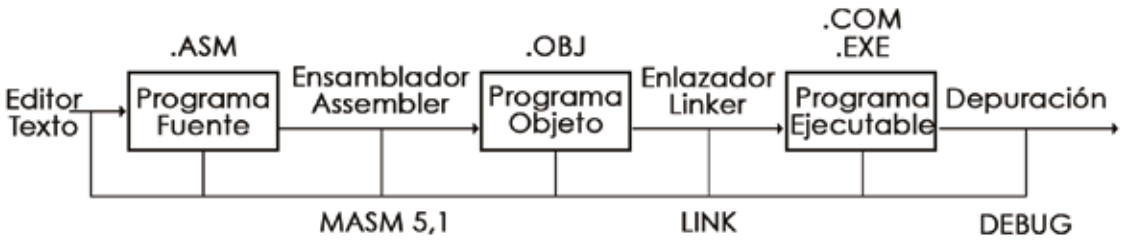
Las herramientas depuradores consideran que una de las fases más importantes del desarrollo de cualquier programa es el proceso de depuración. Dicho proceso adquiere aún más importancia al programar en ensamblador, dado que las operaciones efectuadas son de muy bajo nivel y cualquier fallo puede provocar un funcionamiento erróneo o incluso el fallo del sistema.

Entre sus variantes tenmos:

Dos: debug, grdbdl09.
Windows: turbo debugger, codeview.
Linux: gdb

b. Traducción de ensamblador a máquina:

Figura 2. *Ejemplo de pasos de traducción de ensamblador a máquina*



APLICACIONES

1 Masm (Microsoft Assembler)

El Microsoft Macro Assembler (MASM) es un ensamblador para la familia x86 de microprocesadores. Fue producido originalmente por Microsoft para el trabajo de desarrollo en su sistema operativo MS-DOS, y fue durante cierto tiempo el ensamblador más popular disponible para ese sistema operativo

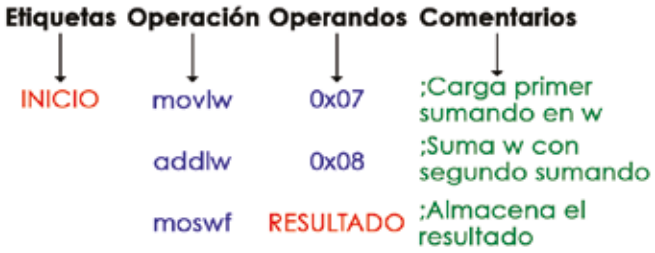
1.1. Variables:

Tabla 1. *Variables*

Expresión	Significado	Tamaño (bytes)
DB	Define byte	1
DW	Define word	2
DD	Define doubleword	4
DF	Define farword	6
DQ	Define quadword	8
DT	Define ten-byte	10

1.2. Codificación:

Figura 3. *Partes de una codificación*



1.3. Estructura básica de un programa en MASM:

```
STACK    SEGMENT STACK          ;

Segmento de pila
DW     64 DUP; Define espacio en la pila

STACK    ENDS

DATA     SEGMENT
   ; Segmento de datos

DATA     ENDS

CODIGO SEGMENT
      ASSUME CS:CODIGO, DS:DATA, SS:STACK

COMIENZO:
MOV DL,41h
          MOV AH,2h
OTRA_VEZ:
INT 21h
          CMP DL,46h
          JE FIN
```

```
        INC DL
        JMP OTRA_VEZ
FIN:
 MOV AH,4Ch
        INT 21h
CODIGO   ENDS
     END COMIENZO
```

2 Debug

Debug funciona en plataforma del sistema operativo DOS y ejecuta líneas de comandos accediendo a posiciones de memoria para editar código assembler y también para visualizarlo. Debug trabaja en el sistema hexadecimal para el ingreso de datos y para visualizar sólo muestra los caracteres o símbolos disponibles en código ASCII.

Para ensamblar un programa en el Debug se utiliza el comando "a" (assemble); cuando se utiliza este comando se le puede dar como parámetro la dirección donde se desea que se inicie el ensamblado, si se omite el parámetro el ensamblado se iniciará en la localidad especificada por CS:IP, usualmente 0100H, que es la localidad donde deben iniciar los programas con extensión .COM, y será la localidad que utilizaremos debido a que debug solo puede crear este tipo específico de programas.

A iniciar un programa se presiona es recomendable el uso del registro rip para indexarlo con el código de segmento y comenzar a ensamblar el programa :

- a **0100** [Enter]

Al hacer esto se indica el direccionamiento de inicio para luego introducir las instrucciones:

Ejemplo:

0C1B:0100 mov ax,0002
;coloca el valor 0002 en el registro ax
0C1B:0103 mov bx,0004
;coloca el valor 0004 en el registro bx
0C1B:0106 add ax,bx
; adiciona al contenido de ax en bx
0C1B:0108 int 20
; provoca la terminación del programa.

Para ejecutar el programa que escribimos se utliza el comando "g",

2. 1. Ejemplo de programas en debug :

Figura 4. *Ejemplo de un programa en debug para dividir dos números*

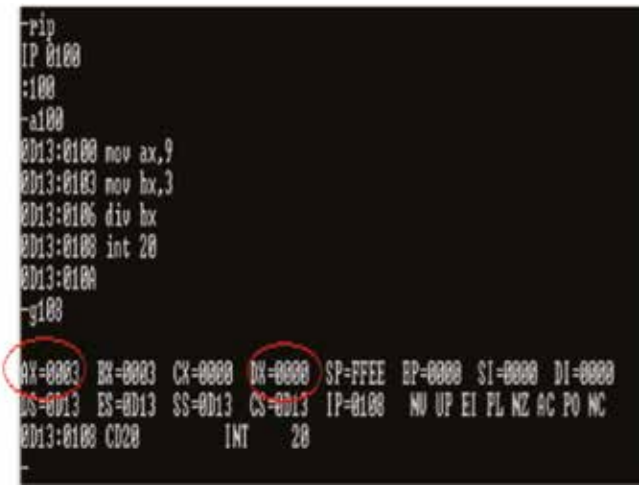


Figura 5. *Ejemplo de un programa en debug para imprimir los números del 0 al 9 del código Ascii*

BIBLIOGRAFÍA

Abel, P. (1996) , Lenguaje Ensamblador para IBM PC y Compatibles.

Alcalde E.(1988) García M.; Peñuelas S." Informática Básica"

Aguilar, L .J, (1990) "Programación en Turbo Pascal", España

Brey, B (1995) . "Los microprocesadores de Intel: Arquitectura,
Programación e Interfaces", 3ª Edición.