INSTITUTO TECNOLÓGICO DE NUEVO LEÓN

Lenguajes y Autómatas 2

UNIDAD # 4

Nombre de la Unidad:

Generación de código objeto

EVIDENCIA # 1: Proyecto 4

Catedrático: Ing. Juan Pablo Rosas Baldazo

Nombre: Mauricio Tamez Botello

Matrícula: 13480527

Guadalupe N.L. a 04 de Mayo del 2018

Introducción

Los lenguajes de programación de bajo nivel son muy apegados a la máquina, algunas empresas de antivirus y videojuegos, siguen programando pequeñas rutinas en ensamblador por su rapidez y a pegamiento al computador.

Esto es debido, a que un programa en ensamblador, es ejecutado casi directamente por el computador, ya que hablan casi "el mismo idioma". Hoy en día es un oficio minucioso de artesanía... sería como el carpintero de lápiz en oreja que va al bosque, tala el árbol, lo lleva al taller, lo pule, lima, da forma y saca el mueble.

En este documento vamos a conocer algunos conceptos básicos del lenguajeensamblador creo que es esencial para cualquier informático.

Capítulo 1: Registros

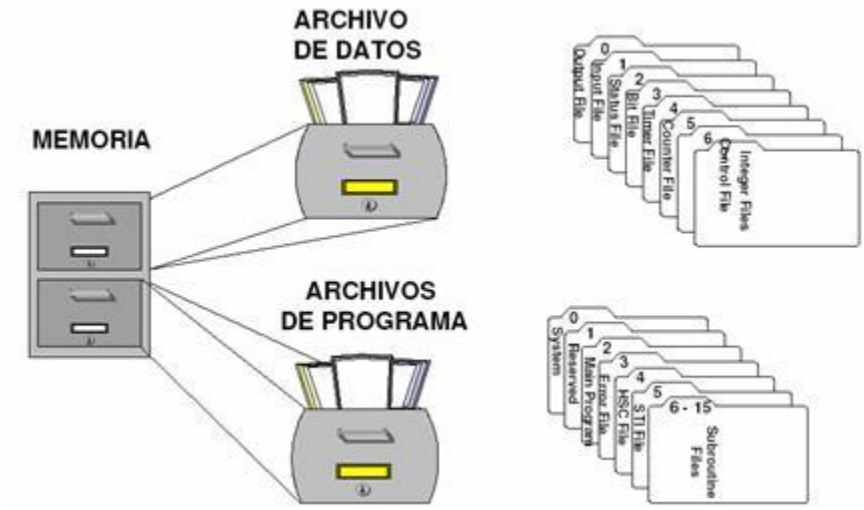
¿Qué son?

Los registros son la memoria principal de la computadora. Existen diversos registros de propósito general y otros de uso exclusivo. Algunos registros de propósito general son utilizados para cierto tipo de funciones. Existen registros acumuladores, puntero de instrucción, de pila, etc.

Los registros son espacios físicos dentro del microprocesador con capacidad de 4 bits hasta 64 bits dependiendo del microprocesador que se emplee.

¿Quiénes lo utilizan?

Antes de nada, para el desarrollo de esta parte hablaremos indistintamente de registros de activación o de marcos de pila. Esto se debe a que en la documentación encontrada sobre el manejo de los registros ebp y esp se hace mención a dicho concepto de marco de pila. Puesto que el lenguaje permite recursividad, los registros de activación se asignan dinámica mente.



Distribución

La UCP o CPU tiene 14 registros internos, cada uno de ellos de 16 bits (una palabra). Los bits están enumerados de derecha a izquierda, de tal modo que el bit menos significativo es el bit 0.
Los registros se pueden clasificar de la siguiente forma:

Registros de datos:

AX: Registro acumulador. Es el principal empleado en las operaciones aritméticas.
BX: Registro base. Se usa para indicar un desplazamiento.
CX: Registro contador. Se usa como contador en los bucles.
DX: Registro de datos.

Estos registros son de uso general y también pueden ser utilizados como registros de 8 bits, para utilizarlos como tales es necesario referirse a ellos como por ejemplo: AH y AL, que son los bytes alto (high) y bajo (low) del registro AX. Esta nomenclatura es aplicable también a los registros BX, CX y DX.

Registros de segmentos:

CS: Registro de segmento de código. Contiene la dirección de las instrucciones del programa.
DS: Registro segmento de datos. Contiene la dirección del área de memoria donde se encuentran los datos del programa.
SS: Registro segmento de pila. Contiene la dirección del segmento de pila. La pila es un espacio de memoria temporal que se usa para almacenar valores de 16 bits (palabras).
ES: Registro segmento extra. Contiene la dirección del segmento extra. Se trata de un segmento de datos adicional que se utiliza para superar la limitación de los 64Kb del segmento de datos y para hacer transferencias de datos entre segmentos.

Registros punteros de pila:

SP: Puntero de la pila. Contiene la dirección relativa al segmento de la pila.
BP: Puntero base. Se utiliza para fijar el puntero de pila y así poder acceder a los elementos de la pila.

Registros índices:

SI: Índice fuente.
DI: Índice destino.

¿Cuáles su aplicación en la generación de códigos?

1. usar el registro de y si está en un registro que no tiene otra variable, y además y no
Está viva ni tiene uso posterior. Si no:
2. usar un registro vacío si hay. Si no:
3. usar un registro ocupado si op requiere que x esté en un registro o si x tiene uso Posterior. Actualizar el descriptor de registro. Si no:
4. usar la posición de memoria de x

Capítulo 2: Lenguaje Ensamblador

¿Qué es?

El lenguaje Assembly (Urbina, 2011) (a veces mal llamado "Ensamblador" por su traducción literal al español) es un tipo de lenguaje de bajo nivel utilizado para escribir programas informáticos, y constituye la representación más directa del código máquina específico para cada arquitectura de computadora

Segunda generación de lenguajes

Versión simbólica de los lenguajes máquina (Urbina, 2011) (MOV, ADD).La comunicación en lenguaje de máquina es particular de cada procesador que se usa, y programar en este lenguaje es muy difícil y tedioso, por lo que se empezó a buscar mejores medios de comunicación con ésta. Los lenguajes ensambladores tienen ventajas sobre los lenguajes de máquina.

Este lenguaje fue usado ampliamente en el pasado para el desarrollo de software, pero actualmente sólo se utiliza encontradas ocasiones, especialmente cuando se requiere la manipulación directa del hardware o se pretenden rendimientos inusuales de los equipos

Características:

El programa lee un archivo escrito en lenguaje ensamblador y sustituye cada uno de los códigos mnemotécnicos por su equivalente código máquina. Los programas se hacen fácilmente portables de máquina a máquina y el cálculo de bifurcaciones se hace de manera fácil.

Clasificación:

- **Ensambladores básicos:** Son de muy bajo nivel, y su tarea consiste básicamente, en ofrecer nombres simbólicos a las distintas instrucciones, parámetros y cosas tales como los modos de direccionamiento
- **Ensambladores modulares, o macro ensambladores:** Descendientes de los ensambladores básicos, fueron muy populares en las décadas de los 50 y los 60, fueron antes de la generalización de los lenguajes de alto nivel. Un macroinstrucción es el equivalente a una función en un lenguaje de alto nivel.

Operaciones básicas

Las operaciones básicas en un lenguaje ensamblador son la suma la resta la multiplicación y la división y Necesitara un poco más de información sobre la arquitectura y SO para el cual programas.
Pero la idea básica es:

--Definir que parámetros tendrá la función.
--Hacer el programa, propiamente dicho, en assembler.

Siguiendo la convención de pasaje de parámetros, manejará registros y posiciones de memoria, devolviendo los resultados en donde deba (una posición de memoria, el registro eax, etc.).

Capítulo 3: Lenguaje maquina

Es el que proporciona poca o ninguna abstracción del microprocesador de un ordenador. El lenguaje máquina solo es entendible por las computadoras. Se basa en una lógica binaria de 0 y 1, generalmente implementada por mecanismos eléctricos. En general el lenguaje maquina es difícil de entender para los humanos por este motivo hacemos uso de lenguajes más parecidos a los lenguajes naturales.

Se denomina lenguaje máquina a la serie de datos que la parte física de la computadora o hardware, es capaz de interpretar. El lenguaje máquina fue el primero que empleo el hombre para la programación de las primeras computadoras. Una instrucción en lenguaje máquina puede representarse de la siguiente forma: 01101100101001001111010. Esta secuencia es fácilmente ejecutada por la computadora, pero es de difícil interpretación, siendo aún más difícil la interpretación de un programa (conjunto de instrucciones) escrito de esta forma.

Esta dificultad hace que los errores sean frecuentes y la corrección de los mismos costosa, cuando no imposible, al igual que la verificación y modificación de los programas.

Características:

El lenguaje máquina realiza un conjunto de operaciones predeterminadas llamadas micro operaciones. Las micro operaciones sólo realizan operaciones del tipo aritmética (+,- ,*, /), lógicas (AND, OR, NOT) y de control (secuencial, de control y repetitiva). El lenguaje maquina es dependiente del tipo de arquitectura. Así un programa máquina para una arquitectura Intel x86 no sé ejecutara en una arquitectura Power PC de IBM (al menos de manera nativa).

Algunos microprocesadores implementan más funcionalidades llamado CISC, pero son más lentos que los RISC ya que estos tienen registros más grandes.

Ventajas

- Mayor adaptación al equipo.
- Máxima velocidad con mínimo uso de memoria.

Desventajas

- Imposibilidad de escribir código independiente de la máquina.
- Mayor dificultad en la programación y en la comprensión de los programas.
- El programador debe conocer más de un centenar de instrucciones.
- Es necesario conocer en detalle la arquitectura de la máquina.

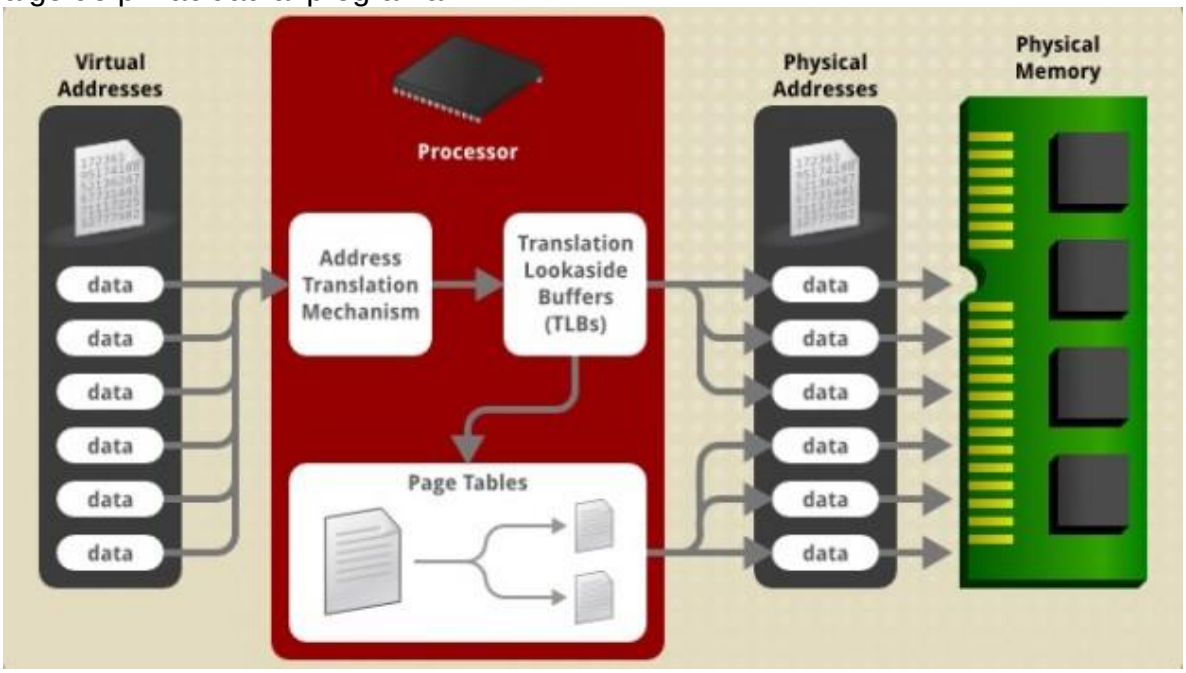
Capítulo 4: Administración de memoria

La administración de la memoria es un proceso hoy en día muy importante, de tal modo que su mal o buen uso tiene una acción directa sobre el desempeño de memoria. En general un ensamblador tiene un administrador de memoria más limitado que un compilador; en la mayoría de los lenguajes de programación el uso de punteros no estaba vigilado por lo que se tienen muchos problemas con el uso de memoria. Los lenguajes más recientes controlan el uso de punteros y tienen un programa denominado recolector de basura que se encarga de limpiar la memoria no utilizada mejorando el desempeño.

La memoria principal puede ser considerada como un arreglo lineal de localidades de almacenamiento de un byte de tamaño. Cada localidad de almacenamiento tiene asignada una dirección que la identifica

Se distinguen los siguientes propósitos del sistema de administración de memoria:
Protección.
Si varios programas comparten la memoria principal, se debería asegur ar que el programa no sea capaz de cambiar las ubicaciones no pertenecientica él. Aunque una acción de escritura puede tener efectos más graves que

una de lectura, esta última tampoco debería estar permitida, para proporcionar algo de privacidad al programa.



Compartimiento.
Este objetivo parece contradecir al anterior, sin embargo a veces es necesario para los usuarios poder compartir y actualizar información (por ejemplo, en una base de datos) y, si se organiza la tarea de entrada a la misma, se puede evitar el tener varias copias de la rutina.

Reubicación.
La técnica de multiprogramación requiere que varios programas ocupen la memoria al mismo tiempo. Sin embargo no se sabe con anticipación donde será cargado cada programa por lo que no es práctico usar direccionamiento absoluto de memoria.

Organización física.
Debido al costo de una memoria principal rápida, éste se usa en conjunto con una memoria secundaria mucho más lenta (y por consiguiente, barata) a fines de extender su capacidad.

Organización lógica.
Aunque la mayor parte de las memorias son organizadas linealmente con un direccionamiento secuencial, esto difícilmente concuerde con el camino seguido por el programa, debido al uso de procedimientos, funciones, subrutinas, arreglos, etc.

Conclusión

Lenguaje de la máquina que se utiliza en binario el que más usamos son los lenguajes naturales que son más sencillos para su comprensión este lenguaje de máquina tiene como características que hace operaciones aritméticas lógicas y de control, también depende de la arquitectura de la máquina para su realización.
Direccionamiento este donde se va a ejecutar el programa en cual debe estar almacenado en la memoria principal puede ser directa o indirecta es cuando se inicia la ejecución del programa.
Lenguaje ensamblado es un traductor de códigos de baja nivel, esta directamente en la máquina para su uso. Su característica es que se puede utilizar en su máquina.
Ensambladores moduladores son lenguajes de alto nivel.
Almacenamiento no permite que maneje la memoria de la máquina.
Registros son las instrucciones que guardan en la memoria distribución es que se puede ejecutar o usar en otras máquinas.

Bibliografía

http://acaurio.blogspot.mx/2016/11/unidad-4-generacion-de-codigo-objeto.html

http://www.academia.edu/10686636/GENERACI%C3%93N_DE_C%C3%93DIGO_OBJETO

Reporte

La memoria principal de la computadora son los registros y tienen diferentes usos y funciones pero cumplen con su propósito general ya que pueden almacenar hasta 64 bits. Dentro de nuestro equipo tenemos hasta 14 registros internos acomodados de izquierda a derecha, o en términos mejor entendidos, desde el menos significativo hasta el más significativo.

En la actualidad, existe una gran variedad de registros y tienen un objetivo específico, hay desde registros para almacenar operaciones aritméticas, hasta segmentos enfocados en código o datos almacenados en el equipo. Para que nosotros podamos referirnos a estos registros, basta con saber que los bytes tienen dos tipos de nomenclaturas existentes que son los bytes y los bytes bajos.

El lenguaje ensamblador es un traductor de códigos de baja nivel está directamente en la máquina para su uso. Su característica es que se puede utilizar en su máquina. Ensambladores básicos solo ejecuta símbolos e instrucciones del direccionamiento del mismo. Ensambladores moduladores son lenguajes de alto nivel.

La administración de memoria es un proceso muy importante y su uso ya sea bueno o malo repercute sobre el desempeño de la memoria, los ensambladores tienen memoria más limitada de los compiladores, la memoria principal se considera como un arreglo lineal de localidades las cuales almacenan un byte de tamaño.