

INSTITUTO TECNOLÓGICO DE NUEVO LEÓN

INGENIERÍA EN SISTEMAS COMPUTACIONALES

LENGUAJES Y AUTÓMATAS II

UNIDAD 4 GENERACIÓN DE CÓDIGO OBJETO

Proyecto 4 Elaboración de un resumen que incluya
cada uno de los subtemas de la unidad.

ALUMNO: Aaron Ulises Nuñez Avila

Nº DE CONTROL: 15480456

MAESTRO: Ing. Juan Pablo Rosas Balazo

Guadalupe, Nuevo León, al 28 de Noviembre del 2018.

1. INTRODUCCIÓN

La fase final de un compilador es la generación de código objeto, que por lo general consiste en código de máquina relocizable o código ensamblador. Las posiciones de memoria se seleccionan para cada una de las variables usadas por el programa. Después, cada una de las instrucciones intermedias se traduce a una secuencia de instrucciones de máquina que ejecuta la misma tarea. Un aspecto decisivo es la asignación de variables a registros.

2. CAPÍTULO 1: REGISTROS

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.

Los registros del procesador se emplean para controlar instrucciones en ejecución, manejar direccionamiento de memoria y proporcionar capacidad aritmética.

Los registros son direccionables por medio de una viñeta, que es una dirección de memoria. Los bits, por conveniencia se enumeran de derecha a izquierda (15, 14, 13, ..., 3, 2, 1, 0)

TIPOS

- Registros de segmento
- Registros de apuntadores de instrucciones
- Registros apuntadores
- Registros de propósitos generales
- Registro índice
- Registro de bandera

3. CAPÍTULO 2: LENGUAJE ENSAMBLADOR

El lenguaje ensamblador o assembler (assembly language en inglés) es un lenguaje de programación de bajo nivel para las computadoras, microprocesadores, microcontroladores, y otros circuitos integrados programables. Implementa una representación simbólica de los códigos de máquina binarios y otras constantes necesarias para programar una arquitectura dada de CPU y constituye la representación más directa del código máquina específico para cada arquitectura legible por un programador.

Esta representación es usualmente definida por el fabricante de hardware, y está basada en los mnemónicos que simbolizan los pasos de procesamiento (las instrucciones), los registros del procesador, las posiciones de memoria, y otras características del lenguaje.

Un lenguaje ensamblador es por lo tanto específico a cierta arquitectura de computador física (o virtual). Esto está en contraste con la mayoría de los lenguajes de programación de alto nivel, que, idealmente son portables.

Características.

1. El código escrito en lenguaje ensamblador posee una cierta dificultad de ser entendido ya que su estructura se acerca al lenguaje máquina, es decir, es un lenguaje de bajo nivel.
2. El lenguaje ensamblador es difícilmente portable, es decir, un código escrito para un microprocesador, puede necesitar ser modificado, para poder ser usado en otra máquina distinta. Al cambiar a una máquina con arquitectura diferente, generalmente es necesario reescribirlo completamente.
3. Los programas hechos por un programador experto en lenguaje ensamblador son generalmente mucho más rápidos y consumen menos recursos del sistema (memoria RAM y ROM) que el programa equivalente compilado desde un lenguaje de alto nivel.
4. Con el lenguaje ensamblador se tiene un control muy preciso de las tareas realizadas por un microprocesador por lo que se pueden crear segmentos de código difíciles y/o muy ineficientes de programar en un lenguaje de alto nivel, ya que, entre otras cosas, en el lenguaje ensamblador se dispone de instrucciones del CPU que generalmente no están disponibles en los lenguajes de alto nivel.

Almacenamiento.

Dentro del almacenamiento que maneja el lenguaje ensamblador se tiene que este tiene una relación con los componentes internos del procesador:

- **En la memoria:** En donde se almacena la información en celdas especiales llamados registros los cuales tienen un nivel alto y un nivel bajo.
- **Unidad aritmética y lógica:** Es la responsable de realizar como su nombre lo indica operaciones aritméticas y lógicas.
- **Unidad de control:** Se encarga de coordinar que los otros componentes ejecuten las operaciones correctamente.
- **Bus interno:** Son los canales por donde pasa la información que la máquina va a procesar (bus de entrada) o procesada (bus de salida).

Manejo de la Memoria: Direccionamiento (Interno y Externo)

El manejo de la memoria depende de que procesador tenga la máquina, entre los cuales a continuación se mencionan los siguientes:

- Memoria de Programa
- Memoria Externa de Datos
- Memoria Interna de Datos
- Registros de Funciones Especiales
- Memoria de Bit

4. CAPÍTULO 3: LENGUAJE MÁQUINA

El lenguaje maquina es un conjunto de instrucciones restringido y sencillo, que puede ser interpretado y ejecutado directamente por la computadora. Las instrucciones se almacenan y tratan en la computadora como cadenas de unos y ceros, aunque se representen en hexadecimal o mediante nemotécnicos. Cuenta con unos niveles que lo caracterizan de instrucciones de máquina, donde incluye los modos de direccionamiento, formato de instrucciones, codificación del código de operación y la clasificación de las arquitecturas a nivel del lenguaje maquina. Dentro de estos niveles entran los registros programables que hay que hay en el CPU donde los caracteriza por número de registro, tamaño y utilidad de cada uno.

Características.

Las instrucciones se almacenan y tratan en la computadora como cadenas de unos y ceros, aunque se representen en hexadecimal o mediante nemotécnicos.

Las instrucciones maquinas suelen, en general, cumplir las siguientes propiedades:

1. Realizan una única y sencilla función: Su interpretación es sencilla.
2. Emplean un número fijo de operandos, que podrán ser implícitos o estar representados explícitamente en la instrucción.
3. La codificación de las instrucciones es bastante sistemática, para que su codificación sea sencilla.
4. Las instrucciones son autocontenidas e independientes

Modos de Direccionamiento.

Las *instrucciones de maquina* nos permiten identificar los tipos de instrucciones de máquina, repertorio o conjunto de instrucciones y los *modos de direccionamiento* permiten llevar a cabo la función del lenguaje maquina.

La denominación de modos de direccionamiento proviene que normalmente se especifica la dirección donde se encuentra el dato o la instrucción. Por lo tanto, los modos de direccionamiento son las diversas formas de determinar el valor de un operando o la posición de un operando o una instrucción.

Cuando se utiliza una posición de memoria, la dirección real de memoria especificada por el modo de direccionamiento se denomina *DIRECCION EFECTIVA* (EA, Effective Address).

El modo de direccionamiento esta codificado en algún/algunos campos de la instrucción.

Tipos de Direccionamiento.

Todo modo de direccionamiento cuenta con el direccionamiento implícito, direccionamiento inmediato, direccionamiento directo, direccionamiento indirecto y el direccionamiento relativo donde cada uno tiene una función a registros o pilas. El propio código de operación indica la dirección del operando.

5. CAPITULO 4: ADMINISTRACIÓN DE MEMORIA

La administración de memoria consiste en determinar la posición de memoria en la que los diferentes símbolos del programa almacenan la información. Depende de la estrategia utilizada para la gestión de memoria, el mecanismo puede variar.

La administración de memoria se refiere a los distintos métodos y operaciones que se encargan de obtener la máxima utilidad de la memoria, organizando los procesos y programas que se ejecutan de manera tal que se aproveche de la mejor manera posible el espacio disponible.

Para poder lograrlo, la operación principal que realiza es la de trasladar la información que deberá ser ejecutada por el procesador, a la memoria principal. Actualmente esta administración se conoce como Memoria Virtual ya que no es la memoria física del procesador sino una memoria virtual que la representa. Entre algunas ventajas, esta memoria permite que el sistema cuente con una memoria más extensa teniendo la misma memoria real, con lo que esta se puede utilizar de manera más eficiente. Y por supuesto, que los programas que son utilizados no ocupen lugar innecesario.

El administrador de memoria tiene como objetivos:

- Ubicar, reemplazar, cargar y descargar procesos en la memoria principal.
- Proteger la memoria de acceso indeseados (accidentales o intencionados).
- Permitir la compartición de zonas de memoria (indispensable para lograr la cooperación de procesos).

Requisitos del administrador de memoria:

1. **Reubicación.**
2. **Protección.**
3. **Compartición.**
4. **Organización Lógica.**
5. **Organización Física.**

En la administración de memoria las técnicas usadas son las siguientes:

- Partición Fija
- Partición Dinámica
- Paginación Simple
- Segmentación Simple
- Memoria Virtual Paginada
- Memoria Virtual Segmentada

Partición fija: La memoria principal se divide en un conjunto de particiones de tamaño fijo durante el inicio del sistema. Un proceso se puede cargar completamente en una partición de tamaño menor o igual.

Ventajas. Sencilla de implementar. Poca sobrecarga al SO.

Desventajas. Fragmentación interna. Número fijo de procesos activos.

Partición dinámica: Las particiones se crean dinámicamente por demanda. Son variables en tamaño y número. Cada proceso se carga completamente en una única partición del tamaño del proceso.

Ventajas. No existe fragmentación interna.

Desventajas. Fragmentación externa. Se debe compactar la memoria, el compactado toma tiempo.

Paginación simple: La memoria principal se divide en un conjunto de marcos de igual tamaño. Cada proceso se divide en una serie de páginas del tamaño de los marcos. Un proceso se carga en los marcos que requiera (todas las páginas), no necesariamente contiguos.

Ventajas. No hay fragmentación externa

Desventajas. Fragmentación interna pequeña.

Segmentación simple: Cada proceso y sus datos se dividen en segmentos de longitud variable. Un proceso carga sus segmentos en particiones dinámicas no necesariamente contiguas. Todos los segmentos de un proceso se deben de cargar en memoria. Se diferencia de la partición dinámica en que un proceso puede ocupar más de un segmento.

Ventajas. No hay fragmentación interna.

Desventajas. Fragmentación externa, pero menor (compactación).

6. CONCLUSIONES

Lenguaje de la máquina que se utiliza en binario el más usamos son los lenguajes naturales que son más sencillos para su comprensión este lenguaje de maquina tiene como características que hace operaciones aritméticas lógicas y de control, también depende de la arquitectura de la máquina para su realización.

Lenguaje ensamblador es un traductor de códigos de baja nivel está directamente en la máquina para su uso. 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.

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 oras maquinas.

7. CONCEPTOS

Organización Lógica: Permitir que los programas se escriban compilables y ejecutables

Abstracción: consiste en aislar un elemento de su contexto o del resto de los elementos que lo acompañan.

Protección: Evitar el acceso a posiciones de memoria sin el permiso expreso.

Registros de segmento: Se utiliza para alinear en un límite de párrafo y su dirección en un registro

Registros de uso general: acumulador, de base, contador, y de datos.

Punteros: es un objeto del lenguaje de programación, cuyo valor se refiere a otro valor

Mnemotécnicos: sirve para ayudar a la memoria a retener una cosa

Bytecode: es un código intermedio más abstracto que el código máquina

Organización Lógica: Permitir que los programas se escriban compilables y ejecutables

Organización Física: Permitir el intercambio de datos en la memoria primaria y secundaria

8. BIBLIOGRAFÍA

Mark, F.(noviembre 29, 2016). "Generación de código objeto"., de blogspot Sitio web:
<http://acaurio.blogspot.mx/2016/11/unidad-4-generacion-de-codigo-objeto.html>

Trevizo, V.. (marzo 11, 2015). "Unidad 4". 2015, de PREZI
Sitio web:
<https://prezi.com/anzpvba2xoss/unidad-4-generacion-de-codigo-objeto/>

REPORTE

Registros

Lenguaje de la máquina que se utiliza en binario los 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.

Lenguaje ensamblador

Direccionamiento este donde se va 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 maquina

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

Administración de memoria

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.