

SEMINARIO DE SOLUCION DE PROBLEMAS DE TRADUCTORES DE LENGUAJES I - PRACTICA 1

OBJETIVO DEL PROYECTO

EL ALUMNO DISEÑARÁ Y PROGRAMARÁ UN ENSAMBLADOR APLICANDO LAS PRINCIPALES TÉCNICAS DE DISEÑO Y GENERACIÓN DE PROGRAMAS DE SISTEMAS POR COMPUTADORA, EMPLEANDO ALGUN LENGUAJE DE ALTO NIVEL.

Conocimientos previos

- Aplicar y emplear algún estilo de programación.
- Manejar y programar en algún lenguaje de alto nivel.
- Conocer la Arquitectura básica de una computadora.
- Aplicar y emplear la estructura de datos (algoritmos de búsqueda y ordenación).
- Aplicar y emplear archivos.
- Aplicar y emplear bases numéricas.
- Aplicar y emplear combinacional.
- Aplicar y emplear diseño de autómatas.
- Aplicar y emplear el manejo de cadenas de caracteres.
- Aplicar y emplear aritmética binaria.

Habilidades y destrezas a desarrollar

Habilidad para diseñar programas de sistemas como es el ensamblador.

Actitudes y valores a fomentar

Trabajo en equipo y trabajo colaborativo.

METODO DE EVALUACION

Exposición en clase(1)	10%
Proyecto	20%
Practicas (10)	70%
ordinario =	100%

CARACTERISTICAS DE LAS PRACTICAS

- Fecha de entrega de acuerdo al programa,
- Para mas información sobre este tema consulte la bibliografía recomendada:

- a. System programming - John Donovan
- b. Microprocesadores avanzados de Intel - Barry Brey
- c. Reference Manual HC12

CARACTERISTICAS DEL PROYECTO FINAL

1. Fecha de entrega de proyecto final: CLASSROOM
2. El proyecto deberá ser subido a la plataforma + un reporte del mismo
3. El proyecto deberá contener lo siguiente :
 - a. MD simple
 - b. MD relativos
 - c. MD indexados
 - d. MD multiple
 - e. TABSIM, DIRECTVAS
 - f. Sistemas de numeración
 - g. Formato de salida S19
 - h. Macros

CRONOGRAMA DE ENTREGA DE PRACTICAS

DE ACUERDO A CLASSROOM

INTRODUCCION

El lenguaje de programación es el medio de comunicación entre el hombre y la máquina. Sistema de símbolos y reglas que permite la construcción de programas con los que la computadora puede operar así como resolver problemas de manera eficaz. La programación tiene como objetivo el tratamiento de la información correctamente, con lo que se espera que un programa de el resultado correcto y no uno erróneo.

Un sistema es un conjunto de componentes que interaccionan entre si para lograr un objetivo común.

La programación es el proceso de convertir las especificaciones a grandes rasgos de los sistemas en instrucciones de máquina que produzcan los resultados deseados.

Por lo tanto la **Programación de Sistemas** se refiere a la creación de programas cuya finalidad es servir a otros programas (software): ejemplos: Ensambladores, Cargadores y Ligadores, Compiladores, Intérpretes.

El **Software de Sistemas** es el software que controla y coordina la operación del equipo que existe en un sistema computacional. El tipo más importante de Software de Sistema es el Sistema Operativo. Conjunto de programas que se encargan de controlar el funcionamiento de los programas que se ejecutan y de la gestión interna de los recursos físicos de la computadora. Como es natural, el sistema operativo forma parte del software de sistema.

Elementos generales que conforman la arquitectura de computadora HC12 de Motorola:

✓ **Conjunto de instrucciones.** Se refiere al nombre de las instrucciones que se utilizan en el lenguaje ensamblador, cada una de las instrucciones tiene asociado un modo de direccionamiento y tanto la instrucción como su modo de direccionamiento tienen un equivalente en lenguaje máquina

(representación hexadecimal). El conjunto de instrucciones esta representado en una tabla que incluye lo siguiente:

- ✓ **Memoria.** La memoria es el espacio de almacenamiento temporal para datos e instrucciones. La dirección máxima de memoria que se puede representar en la máquina es el equivalente al valor hexadecimal \$FFFF.
- ✓ **Palabra:** La palabra es una medida estandar de la arquitectura, que en este caso se llama WORD y equivale a 2 bytes.
- ✓ **Registros:** Son pequeñas áreas de almacenamiento de 8 o de 16 bits y que tienen los siguientes nombres: A, B (son de 8 bits); X, Y, SP, PC, D (son de 16 bits), existe otro registro que se llama CCR de (8 bits).
 - Modos de direccionamiento. Los modos de direccionamiento son una parte implícita del conjunto de instrucciones, los más comunes son:
 - INH. Inherente.
 - IMM. Inmediato.
 - DIR. Directo.
 - EXT. Extendido.
 - REL. Relativo (de 8 y 16 bits)
 - IDX. Indizado (de 5 bits, predecremento, preincremento, postdecremento, postincremento, acumulador).
 - IDX1. Indizado de 9 bits.
 - IDX2. Indizado de 16 bits.
 - [IDX2]. Indizado indirecto de 16 bits.
 - [D,IDX]. Indizado indirecto de acumulador.

PRACTICA 1: ESTRUCTURA DE LA LINEA DE PROGRAMA

IMPLEMENTARA UN PROGRAMA PARA DEFINIR LA ESTRUCTURA DE LA LINEA DE PROGRAMA EN LENGUAJE ENSAMBLADOR, PREVIO ANALISIS DE LA MISMA. TOMANDO EN CUENTA LOS 20 PRIMEROS MNEMONICOS.

Los ensambladores son programas que procesan los enunciados del programa origen en lenguaje ensamblador y los traducen en archivos en lenguaje máquina que son ejecutados por un microprocesador o un micro controlador, estos permiten que los programas origen se escriban y se editen en una computadora para generar un código ejecutable en otra computadora. El archivo en lenguaje objeto ejecutable resultante se carga (CARGADOR) y se ejecuta en el sistema destino.

Una instrucción es un código binario con un significado ya establecido de antemano. Para no tener que memorizar estos códigos y facilitar de esta manera la tarea del programador, a cada instrucción se le asigna una abreviatura que dé una idea de la acción que realiza. Así por ejemplo si se pretende cargar un dato en el acumulador se emplea una instrucción cuyo nemónico es LDAA abreviatura de Load Accumulator.

Pero además hay que indicar en la instrucción cual es el dato o donde se encuentra el dato con el que hay que operar. Así por ejemplo la instrucción LDAA (Cargar el acumulador) da lugar a los nueve códigos de operación que se indican en la tabla de su manual de referencia, que tienen como denominador común que todas las instrucciones cargan en el acumulador un dato, pero difieren en la forma de obtener ese dato.

Un programa consiste en declaraciones (llamadas línea código) línea por línea. Su sintaxis es la siguiente: nombre del campo, operación del campo, operando del campo, sintaxis del campo, ejemplo:

- ✓ **Campo de Etiquetas:** esta puede aparecer por sí sola en una línea. El ensamblador interpreta esto como "establece el valor de la etiqueta igual al valor actual del contador de programa (PC)".
- ✓ **Campo de Operaciones/Instrucciones:** El campo de operaciones aparece después del campo de etiquetas y debe de estar precedido por al menos un espacio en blanco.
- ✓ **Código de Operación:** Estos símbolos corresponden directamente a instrucciones de máquina. El código de operación incluye a cualquier nombre e registro asociado con la instrucción.
- ✓ **Directiva / Pseudoinstrucción:** Estos son códigos de operación especiales conocidos por el ensamblador, los cuales más bien controlan el proceso de ensamblado en vez de ser traducidos a instrucciones máquina.
- ✓ **Campo de Operandos:** La interpretación del campo de operandos depende del contenido del campo de operaciones.
- ✓ **Campo de Comentarios:** El último campo de un enunciado origen del ensamblador es el campo de comentarios.

ETIQUETA	MNEMONICO	Operando (S)	COMENTARIO (antepone ";")
	LDAA	#5	;carga al registro A un 5
Num1:	JMP	\$10	;salto incondicional a la dirección 10