

Nombre del alumno: Johanne Joaquín Arriaga Díaz

Nombre del profesor: Jonathan Gabriel Hernández.

Nombre del trabajo: Cuadro sinóptico unidad III y IV.

Materia: Microprocesadores.

PASIÓN POR EDUCAR

Grado: Séptimo cuatrimestre

Grupo: ISC13SDC0119-F

1) Formato general de un programa en lenguaje ensamblador
2) Señales en los pines del microprocesador 80x86

3.1.- Procedimiento para generar un programa ejecutable.

Un programa en ensamblador: compuesto por líneas, conteniendo cada una un comentario, una única instrucción o una directiva.

Se utilizan los símbolos <> para encerrar un identificador o un número; los símbolos [] encierran elementos opcionales; los {} elementos que se puede escribir consecutivamente varias veces; el carácter | separa elementos opcionales.

3.2.- Debugger.

Para usar Debugger

- Paso 1. Llame al depurador: start (). La ventana Monitor muestra información sobre los procesos depurados, módulos interpretados y opciones seleccionadas. Inicialmente no hay procesos depurados. Debe especificarse qué módulos se van a depurar (también llamado interpretado).
- Paso 2. Seleccione Módulo> Interpretar... en la ventana Monitor. La ventana Interpretar módulos es desplegado.
- Paso 3. Seleccione los módulos apropiados de la ventana de diálogo de interpretación.
- Paso 4. En la ventana Monitor, seleccione Módulo> el módulo a interpretar> Ver. El archivo de origen se muestra en la ventana Ver módulo.
- Paso 5. Establezca los puntos de interrupción, si los hay.
- Paso 6. Inicie el programa a depurar. Esto se hace de la forma habitual desde el shell de Erlang.
- Paso 7. Para adjuntar a uno de estos procesos, haga doble clic en él o seleccione el proceso y luego elija Proceso> Adjuntar. Adjuntar a un proceso abre una ventana Adjuntar proceso para este proceso.
- Paso 8. Desde la ventana Adjuntar proceso, puede controlar la ejecución del proceso, inspeccionar valores variables, establecer puntos de interrupción, etc

3.3.- Introducción a las interrupciones.

Una interrupción es una operación que suspende la ejecución de un programa de modo que el sistema pueda realizar una acción especial. La rutina de interrupción ejecuta y por lo regular regresa el control al procedimiento que fue interrumpido, el cual entonces reanuda su ejecución.

Tabla de servicio de interrupción.

La computadora enciende, el BIOS y el DOS establecen una tabla de servicios de interrupción en las localidades de memoria 000H-3FFH. Para el uso de 256 (100H) interrupciones, cada una con un desplazamiento: segmento relativo de cuatro bytes en la forma IP:CS. El operando de interrupción como INT 05H identifica el tipo de solicitud. Existen 256 entradas, cada una de cuatro bytes, la tabla ocupa los primeros 1, 024bytes de memoria, desde 000H hasta 3FFH. Cada dirección en la tabla relaciona a una rutina de BIOS o del DOS para un tipo específico de interrupción. Por lo tanto los bytes 0-3 contienen la dirección para la interrupción 0, los bytes 4-7 para la interrupción 1, y así sucesivamente:

3.4.1.- Interrupciones de software.

El BIOS contiene rutinas de entrada/salida y tablas que indican el estado de los dispositivos. El dos y los programas usuarios pueden solicitar rutinas del BIOS para la comunicación con los dispositivos conectados al sistema. El método para realizar la interfaz con el BIOS es el de las interrupciones de software.

3.4.2.- Interrupciones de hardware

Los módulos del DOS, IO.SYS y MSDOS.SYS, facilitan el uso del BIOS. Proporcionan las pruebas adicionales necesarias, las operaciones del DOS por lo general son más fáciles de usar que sus contrapartes del BIOS y por lo común son independientes de la máquina

3.5.- Programación modular (MACROS).

Uno de los componentes que caracterizan los computadores personales es su sistema operativo. Una PC puede correr varios sistemas operativos: CP/M, CP/M-86, XENIX, Windows, PC-DOS, y MSDOS. Lo que los define es la forma en que están integrados sus servicios y la forma en que se accesa a ellos. Esto es precisamente lo que el linker debe enlazar y resolver.

- Comandos Internos (reconocidos y ejecutados por el COMMAND.COM)
- Comandos Externos (.EXEs y .COMs)
- Utilerías y drivers (programas de administración del sistema)
- Shell (Interfaz amigable, sólo versiones 4.0 o mayores)
- Servicios (Interrupciones)

4.1.- Generador de reloj.

El 8086 requiere una única señal de reloj. No genera su señal, usa un generador de reloj 8284, que usa un cristal oscilador para determinar la frecuencia de señal. Intercambiando este cristal, se puede seleccionar diferentes velocidades de operación.

Para un rendimiento óptimo, requiere una señal de reloj que este activo un tercio y desactivado dos tercios del tiempo. En búsqueda de una instrucción, su dirección se obtiene sumando el desplazamiento, contenido en el IP, al valor del registro de segmento CS, multiplicado por 16. Normalmente al terminar la ejecución de una instrucción, el IP se incrementa, con lo que se pasa a direccionar la siguiente instrucción. Las instrucciones de salto y salto a subrutinas pueden modificar el contenido de IP de tres formas diferentes

- Direccionamiento relativo. Al contenido del IP se suma, un desplazamiento de 8 a 16 bits, con signo, proporcionado por la misma instrucción.
- Direccionamiento directo. Se carga, en el IP, una nueva dirección presente en la instrucción.
- Direccionamiento indirecto. El dato, es interpretado por las instrucciones de salto, como la dirección a la que se debe saltar.

4.2.- Temporización del canal.

Temporización en General

- Las operaciones de transferencia de datos hacia o desde el 8086 ocupan al menos un bus cycle
- Cada bus cycle consiste en 4 períodos de reloj del sistema (T), T1, T2, T3, T4.
- Si el reloj funciona a 5 MHz: - $T = 1/5 \text{ MHz} = 0.2 \text{ ms}$ - Bus cycle = $4 T = 0.8 \text{ ms} = 800 \text{ ns}$ - Velocidad máxima de lect. o esc. de datos de la memoria o del espacio de E/S = $1/0.8 = 1.25$ millones de operaciones por seg. - El 8086 puede ejecutar 2.5 millones de inst. por seg. (MIPS) debido a que posee una cola interna

4.3.- Interfaz de memoria.

Los microprocesadores 8086 y 8088 poseen registros de un tamaño máximo de 16 bits. Con esto podemos direccionar como máximo un total de 64 KB de memoria. Pero puede direccionar hasta 1 MB. Tendríamos que tener registros de 20 bits. Hay que recurrir a un mecanismo para direccionar toda la memoria. Que es la segmentación que divide la memoria en segmentos de 64 KB. Cada segmento se asocia con un registro de segmento, el desplazamiento (u offset) dentro de ese segmento lo proporciona otro registro de 16 bits. La dirección absoluta se calcula multiplicando por 16 el valor del registro de segmento y sumando el desplazamiento.: Dirección = segmento * 16 + desplazamiento

Los registro básicos que posee el 8088 son el CS, DS, ES y SS, son registros de 16 bits como el resto de los registros de microprocesador pero su uso interno al componer direcciones es especial. Definen una dirección base o de segmento sobre la que aplicar el desplazamiento de 16 bits

4.4.- Interface de entrada/salida

Las interfaces de entrada y de salida: para transferir información entre dispositivos de (E/S) y de (E/S). Los periféricos necesitan enlace de comunicación para funcionar como interfaces con la unidad de procesamiento central para resolver diferencias en la computadora central y de cada periférico. Las funciones de la interfase son almacenar los datos y realizar conversiones que se requieran. Detecta errores en la transmisión y puede reiniciar la transacción en casos de error. Puede testear, arrancar y detener el dispositivo según las directivas impartidas por la CPU. En algunos casos la interfase puede consultar a la CPU si algún dispositivo está requiriendo atención urgente.

4.4.1.- Interface programable.

Integrado universalmente el 8255. Existen componentes que conservan, toda su utilidad. Como UART 8251, controlador de interfaz paralelo PPI 8255. Es versátil y económico de fácil conexión, tiene un modo elegante y sencillo puertos E/S disponibles. Está en tarjetas de expansión de puertos para el bus ISA del PC, Es también ideal para expansión de puertos E/S en monoplacas o gestión de periféricos como conversores analógicos/digital y otros.

4.5.- Temporizador programable.

El 8086 y 8088 fue usado como generador de reloj en el IBM PC. Se usaba para generar la frecuencia de 4.77 MHz para el microprocesador del IBM PC, otra de 3.58 MHz para la tarjeta de video y una de 1.19 MHz para los tres temporizadores, de los cuales salían otras frecuencias (ver abajo).

Soportaba hasta 5 MHz, necesitando un cristal de 15 MHz para el 8284 IBM uso una frecuencia de cristal ligeramente menor, a 14.31818 MHz, que al dividirla entre 3 resultaba en el microprocesador del IBM PC corriendo a 4.7727267 MHz. Permitía dividir la frecuencia base del cristal entre cuatro y generar una frecuencia de 3.579545 MHz usada por la tarjeta de video Color Graphics Adapter (CGA) para generar el burst de color en el estándar de televisión NTSC, ahorrándose así algunas piezas que hubieran sido necesarias para generar esa frecuencia independientemente.