Capítulo 12 Estructura y funcionamiento del procesador

12.1 Organización del procesador:

Requisitos a cumplir por el procesador

1. Captar instrucción.
2. Interpretar instrucción.
3. Procesar datos.
4. Escribir datos.

12.2 Organización de los registros

Los registros del procesador son de dos tipos

Registros visibles por el usuario

1. Uso general
2. Datos
3. Direcciones
4. Códigos de condición

Registros de control y de estado: Son esenciales.

1. PC (Program Counter) (Contador de Programa)
2. IR (Instruction Registrer) (Registro de Dirección de Memoria)
3. MAR (Memory Address Register) (Registro de Dirección de Memoria)
4. MBR (Memory Buffer Register) (Registro Intermedio de Memoria)

PSW (Program Status Word):

Contiene información de estado y códigos de condición. Entre los campos e indicadores comunes se tienen 7:

1. Signo
2. Cero
3. Acarreo
4. Igual
5. Desbordamiento
6. Interrupciones
7. Supervisor

Figura 12.1 El procesador y el bus del sistema.

12.3 Ciclo de instrucción

1. Captación: Llevar la siguiente instrucción de la memoria al procesador.
2. Ejecución: Interpretar el código de operación y llevar a cabo la operación
3. Interrupción: Si las interrupciones están inhabilitadas
4. Ciclo Indirecto: La ejecución de una operación puede involucrar uno o más operandos que requerirán acceso a memoria.

Figura 12.4 Ciclo de instrucción

12.4 Segmentación de instrucciones

Estrategias de segmentación:

Como en una cadena de montaje que el producto pasa por una serie de etapas, bueno en la segmentación de instrucciones se utiliza esta misma técnica.

Segmentación de Cauce (PIPELINING): Técnica de diseño de procesadores, basada en la división de las instrucciones en etapas, consiguiendo que una instrucción comience a ejecutarse antes que las anteriores terminen de ejecutarse.

Figura 12.9 Cauce de instrucciones de dos etapas

Salto o Sentencias Condicionales

Es un punto de un programa informático donde el flujo se ve alterado con una instrucción que causa un salto, puede ser efectiva o no efectiva, si no es efectivo el flujo del programa no será alterado, y la siguiente instrucción a ejecutarse es la posterior en el código, si es efectiva la siguiente instrucción a ejecutarse será aquella marcada como destino del salto.

Formas de instrucción de salto

1. Salto condicional: Puede ser efectivo o no según una determinada condición
2. Salto incondicional: Siempre se realizara

Descomposición del procesamiento de una instrucción son 6:

1. FI (Fetch Instruction)
2. DI (Decode Instruction)
3. CO (Calculate Operand)
4. FO (Fetch Operand)
5. EI (Execute Instruction)
6. WO (Write Operand)

Prestaciones de un cauce segmentado:

Tratamiento de saltos:

El primer obstáculo es la instrucción de salto condicional. Consideremos varias aproximaciones al tratamiento de saltos condicionales: Son 5.

1. Flujos múltiples: Debe hacer una elección por cada instrucción de salto y se puede equivocar
2. Pre-captar el destino del salto: Cuando se capta una instrucción de salto condicional, se precapta la instrucción destino del salto
3. Buffer de bucles: Memoria pequeña de gran velocidad
4. Predicción de saltos:

* Predecir que nunca se salta.
* Predecir que siempre se salta.
* Predecir según el código de operación.
* Conmutador Saltar/No Saltar.
* Tabla de historia de Saltos.

1. Salto retardado: Reordenar las instrucciones automáticamente para que la instrucciones de salto tengan lugar después de lo realmente deseado