

# Registros de control y estado, y su participación en el ciclo de ejecución de instrucciones

Organización de computadoras 2014

Universidad Nacional de Quilmes

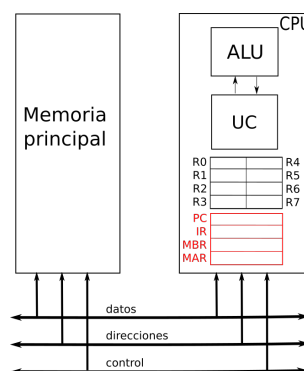
Para comprender el funcionamiento del procesador, consideremos los requisitos que tiene que cumplir:

1. Buscar instrucciones desde la memoria o desde un dispositivo de entrada y salida
2. Decodificar instrucciones para determinar que acción es necesaria
3. Buscar los datos que la ejecución de la instrucción puede requerir
4. Procesar los datos que la ejecución puede necesitar a través de una operación lógica o aritmética
5. Almacenar los resultados donde corresponda

Para llevar a cabo estas cosas es claro que el procesador debe almacenar algunos datos temporalmente. Debe recordar la posición de la última instrucción de forma de poder determinar de dónde tomar la siguiente. Necesita almacenar instrucciones y datos temporalmente mientras una instrucción está ejecutándose. En otras palabras, el procesador necesita una pequeña memoria interna.

Dentro del procesador hay un conjunto de registros clasificados en dos tipos:

- Registros visibles al programador: permiten al programador de lenguaje máquina o de ensamblador minimizar las referencias a memoria principal por medio de la optimización de uso de registros
- Registros de control y estado: Son utilizados por la unidad de control para controlar el funcionamiento del procesador y por programas privilegiados del sistema operativo para controlar la ejecución de programas.



## Registros de control y estado

Hay diversos registros del procesador que se emplean para controlar su funcionamiento. La mayoría de ellos, en la mayor parte de las máquinas no son visibles por el usuario, pero alguno de ellos puede ser visible por ciertas instrucciones máquina ejecutadas en un modo privilegiado o de sistema operativo.

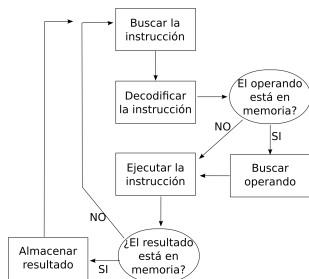
Naturalmente, diferentes máquinas tendrán distinta organización de registros y usarán distinta terminología. Son esenciales cuatro registros para la ejecución de una instrucción:

- **Contador de programa (Program Counter- PC):** Contiene la dirección de la instrucción a buscar
- **Registro de instrucción (Instruction Register- IR):** Contiene la instrucción buscada mas recientemente. Muchas veces este registro tiene mas capacidad que los demás, a los fines de *almacenar la instrucción completa*.

- **Registro de dirección de memoria (Memory Address Register- MAR)**  
Contiene la dirección de una posición de memoria
- **Registro amortiguador de memoria (Memory Buffer Register - MBR)**  
Contiene el dato a escribir en una posición de memoria o el dato contenido en una posición de memoria leído mas recientemente

No todos los procesadores tienen registros internos designados como MAR o MBR pero se necesita algún mecanismo de almacenamiento intermedio equivalente mediante el cual se de salida a los bits que van a ser transferidos por el bus de sistema se almacenen los bits leídos por el bus de datos.

Típicamente, el procesador actualiza el PC luego de cada búsqueda de instrucción (ver figura), de manera que siempre apunte a la siguiente instrucción a ejecutar. Una instrucción de bifurcación o salto también modificará el valor del registro PC. La instrucción buscada se carga en IR, donde son analizados los códigos de operación y los campos de operando. Se intercambian datos con la memoria por medio del MAR y el MBR. El MAR se conecta con el bus de direcciones y el MBR con el bus de datos. Los registros visibles por el usuario repetidamente cambian datos con MBR.



Los registros mencionados se usan para la transferencia de datos entre el procesador y la memoria principal. Dentro del procesador, los datos tienen que ofrecerse a la ALU para su procesamiento. La ALU puede tener acceso directo al MBR y a los registros visibles por el programador, pero alternativamente puede haber registros intermedios en torno a la ALU, que le sirven como entrada y salida e intercambian datos

con el MBR y los registros visibles al programador.

Muchos diseños de procesadores incluyen un registro o un conjunto de registros, conocidos a menudo como palabra del estado de programa o *program status word* (PSW), que contiene información de estado. Típicamente contiene códigos de condición, incluyendo a menudo los siguientes bits de estado:

- El signo del resultado de la última operación
- Indicador de resultado cero o nulo
- Indicador de acarreo en la última operación (suma o resta)
- Indicador de igualdad entre operadores
- Indicador de desbordamiento aritmético
- Habilitación de interrupciones
- Indicador de modo supervisor (para permitir ciertas instrucciones privilegiadas)

En algunos diseños es posible encontrar otros registros relativos a estado y control. Puede existir un puntero a bloque de memoria que contenga información de estado adicional (por ejemplo, bloques de control de procesos). En las máquinas que utilizan interrupciones vectorizadas puede existir un registro de vector de interrupciones. Si se utiliza una pila para llevar a cabo ciertas funciones, se necesita un puntero a pila del sistema. En un sistema de memoria virtual se utiliza un puntero a una tabla de paginas. por último, pueden utilizarse registros para el control de operaciones de E/S.

## 1. Ejecución de programas en la arquitectura Q

La ejecución de un programa no es ni más ni menos que la ejecución en secuencia de sus instrucciones. Para llevar cuenta de la instrucción actual, o mejor dicho, la siguiente a ejecutar, se utiliza el *Program Counter*.

Consideremos como ejemplo la instrucción que está ensamblada en la celda 0xD401. Por lo tanto PC=D401

	...
D401	3225
D402	F3F3
	...

- Búsqueda de la instrucción:** Se lee la celda indicadas por el PC y se almacenan en **IR**. Por lo tanto IR=3225. Además, se actualiza el PC para que quede preparado para la siguiente lectura PC=D402
- Decodificación de la instrucción:** A partir de la interpretación de la cadena en **IR**, la Unidad de Control entiende lo siguiente:

codop(4b)	M.D. (6b)	M.O. (6b)
0011	001000	100101

Dado que el modo de direccionamiento del operando destino es **directo**, la UC entiende que hace falta la lectura de otra celda
- Completar la búsqueda de la instrucción:** Se lee la celda indicada por el PC (D402) y se almacena también en **IR**. Por lo tanto IR=3225D402
- Completar la decodificación de la instrucción:** Con esta nueva lectura puede concluirse que:

codop	M.D.	M.O.	Dest(16b)
0011	001000	100101	1111001111110011

A los fines de la comprensión de nuestro ejemplo, es importante notar que dicho código máquina corresponde a la instrucción SUB [0xF3F3], R5

- Búsqueda de operandos:** La Unidad de control solicita una lectura de la celda **F3F3** para obtener el operando destino. Para esto realiza los siguientes pasos:
  - Copia la dirección **F3F3** del registro IR al registro **MAR**
  - Indica a la Memoria Principal una lectura a través de las líneas correspondientes en el **Bus de control**.
  - Cuando la Memoria Principal indica que el dato está disponible (también a través del bus de control), la UC tiene disponible el dato en el registro **MBR**.

**6. Ejecución de la instrucción:** La UC le indica a la **ALU** la operación **SUB** y le suministra los dos operandos:

- Operando destino: desde el registro **MBR**
- Operando origen: desde el registro **R5**

**7. Almacenamiento de resultado:** La Unidad de Control solicita una **escritura** en Memoria Principal para almacenar el resultado en el operando destino. Para esto realiza los pasos:

- Copia la dirección **F3F3** del registro IR al registro **MAR**
- Copia el resultado arrojado por la ALU en el registro **MBR**
- Indica a la Memoria Principal una **escritura** a través de las líneas correspondientes en el **Bus de control**.

**8.** Comienza un nuevo ciclo con la siguiente instrucción

## Fuente

Extraído de *Organización y Arquitectura de Computadores*, William Stallings - 7ta edición. Capítulo 12.