UAI - Sistemas de Computación I

**MONOGRAFÍA VON NEUMANN EN LA COMPUTACIÓN ACTUAL**

Gerardo Tordoya - 10-sep-2020

------------------------------------------------

Arquitectura von Neumann

Una computadora se define como un dispositivo electrónico que toma una entrada, la cual procesa, y envía una respuesta. Para procesar los datos, la computadora sigue una serie de instrucciones (un programa de computadora).

Los dispositivos antiguos de cómputo tenían programas fijos. Piénsese en los vestigios actuales de esa arquitectura, por ejemplo, una calculadora. Una calculadora se puede usar para hacer matemáticas, pero no puede ser usada para ningún otro propósito. Cambiar su programa requiere un rediseño de sus conexiones.

Un programa de computadora es modificable.

Este concepto lo planteó en 1945 un matemático y físico, John von Neumann, planteando así una diferencia que hoy se conoce como la dupla hardware/software. El programa "duro" (no modificable) queda fijo en las lecturas que puede hacer la "maquina" desde el puro hardware (valga la redundancia), a diferencia del software "blando" que es modificable.

Los actores de esta arquitectura son sus componentes principales: memoria y procesador. No hace falta extenderse en lo que debería entenderse por memoria, pero sí mencionaré grosso modo los componentes nucleares del procesador y luego entenderemos esa arquitectura con un ejemplo de funcionamiento (el cual tiene como propósito ver cómo interactúan el programa "duro" con el programa "blando", es decir, hardware y software).

# CPU (Unidad Central de Procesamiento)

El propósito del procesador (CPU) es capturar y ejecutar instrucciones.

## Componentes:

* **ALU (Arithmetic Logic Unit):** (unidad lógica aritmética) realiza los cálculos y decisiones lógicas.
* **CU (Control Unit):** (unidad de control) envía microcomandos para controlar el flujo de los datos en la CPU.
* **CACHE:** (submemoria ultrarrápida) provee acceso rápido a datos e instrucciones frecuentemente usados.
* REGISTROS:
  + **PC (Program Counter):** (contador) guarda la dirección de la próxima instrucción en la memoria.
  + **MAR (Memory Address Register):** (registro de dirección de memoria) guarda la dirección desde donde los datos serán tomados o en donde serán guardados.
  + **MDR (Memory Data Register):** (registro de datos de memoria) guarda los datos obtenidos de o a guardar en la memoria.
  + **ACC (Accumulator):** (acumulador) guarda el resultado de los datos.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | REGISTRO  PC |  | REGISTRO  MAR |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | REGISTRO  ACC |  | REGISTRO  MDR |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

# Cómo el ciclo de captura y ejecución funciona

Las computadoras son "máquinas" complejas que, dentro de la CPU, son capaces de realizar miles de ciclos de captura y ejecución por segundo usando una secuencia metódica de eventos.

Veamos la secuencia en detalle:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  1 |  | MAR  1 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC |  | MDR  Cargar dir 5 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* La dirección de memoria donde se encuentra la próxima instrucción está guardada en el contador (PC).
* En este ejemplo, la primera instrucción está en la dirección 1.
* Por lo tanto, el valor en el contador será 1.
* Este valor es pasado al registro de dirección de memoria (MAR).
* La instrucción de esa dirección el luego capturada y llevada al registro de datos de memoria (MDR).

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  2 |  | MAR  5 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE  Cargar direcc. 5 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC |  | MDR  Cargar direcc. 5 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* El contador (PC) ahora necesita apuntar a la próxima instrucción, que será 2.
* Una copia de lo cargado en la CPU se aloja en submemoria ultrarrápida (CACHE). Si se necesita de nuevo, se puede acceder a ese dato rápidamente sin necesidad de ir a la memoria.
* La unidad de control (CU) decodifica la carga y entiende que hay que capturar un dato en la dirección 5 de la memoria.
* Por lo cual el registro de dirección de memoria (MAR) va a cambiar su valor a 5.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  2 |  | MAR  5 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE  Cargar D5  23 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  23 |  | MDR  23 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* Se captura el dato en la dirección 5 de memoria (un 23).
* Y (de nuevo) se lo lleva al registro de datos de memoria (MDR).
* (De nuevo) se hace una copia en la submemoria ultrarrápida (CACHE). A diferencia de los otros registros de la CPU (que son sobre escritos con cada nuevo dato), la CACHE contiene una porción mayor de submemoria y es capaz de alojar varias instrucciones y datos.
* El valor 23 ahora será enviado al acumulador (ACC)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  2 |  | MAR  2 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE  Cargar D5  23  Sum dir 6 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  23 |  | MDR  Sum dir 6 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* El contador (PC) sigue apuntando a la dirección 2 de la memoria.
* Así que ese 2 va a ser pasado al registro de dirección de memoria.
* Y se capturará lo que hay en esa dirección de memoria.
* La captura se alojará en el registro de datos de memoria (MDR).
* De nuevo, una copia se aloja en la CACHE.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  3 |  | MAR  6 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE  Cargar D5  23  Sum dir 6  12 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  23 |  | MDR  12 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* El contador se incrementará en 1 de nuevo.
* La unidad de control pasa la dirección 6 al registro de dirección de memoria.
* El cual captura lo que hay en la dirección 6 de la memoria.
* Y lo lleva al registro de datos de memoria.
* Y se hace una copia de ese 12 en la CACHE.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  3 |  | MAR  6 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU  23+12 |  | CACHE  Cargar D5  23  Sum dir 6  12 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  35 |  | MDR  12 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* Como hay que sumar 23 y 12, la unidad aritmética lógica (ALU) hará ese cómputo.
* El resultado, 35, se guardará en el acumulador.

Veamos cómo han quedado los registros ahora:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  3 |  | MAR  6 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  35 |  | MDR  12 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

El contador tiene alojado el 3. Es decir, se han completado 2 instrucciones y ahora podemos avanzar a la 3ra:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  3 |  | MAR  3 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 12 |  |
|  | ACC  35 |  | MDR  Gua en D6 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* El contador tiene alojado el 3. Es decir, se han completado 2 instrucciones y ahora podemos avanzar a la 3ra.
* Ese 3 se pasa al registro de dirección de memoria.
* Lo que hay en la dirección 3 es llevado al registro de datos de memoria.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  4 |  | MAR  6 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE  Gua en D6  35 |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 35 |  |
|  | ACC  35 |  | MDR  35 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* A continuación, simultáneamente se hace una copia en la CACHE y se incrementa el contador.
* La unidad de control examina la instrucción y cambia el valor del registro de dirección de memoria.
* El 35 del acumulador es enviado de vuelta al registro de datos de memoria.
* Y de allí a la dirección de memoria 6.
* Y se guarda en la CACHE una copia del valor.

Valores en los registros antes de comenzar con la 4ta instrucción:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  4 |  | MAR  6 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 35 |  |
|  | ACC  35 |  | MDR  35 |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

Procedamos al ciclo siguiente:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  | PC  5 |  | MAR  4 |  |  |  | **DIRECCIÓN** | **DATOS** | **MEMORIA** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | **CPU** |  | CU |  |  |  | 1 | Cargar dirección 5 |  |
|  |  |  |  |  | 2 | Sumar dirección 6 |  |
|  |  |  |  |  |  |  | 3 | Guardar en dirección 6 |  |
|  | ALU |  | CACHE |  |  |  | 4 | Fin |  |
|  |  |  |  |  | 5 | 23 |  |
|  |  |  |  |  |  |  | 6 | 35 |  |
|  | ACC  35 |  | MDR  Fin |  |  |  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

* El 4 se pasa al registro de direcciones de memoria.
* El contenido de la dirección 4 de memoria se pasa al registro de datos de memoria.
* El contador se incrementa (pasa a 5).
* La unidad de control determina que es el final del programa.

# Resumen

* El contador contiene la dirección de memoria de la próxima instrucción.
* Lo pasa al registro de direcciones.
* El contenido de esa dirección de memoria se pasa al registro de datos.
* Se incrementa el contador (dirección de memoria de la próxima instrucción).
* El dato obtenido de la memoria puede ser enviado al acumulador si lo necesita la unidad aritmética lógica.