La Máquina de von Neumann

Modelo de von Neumann

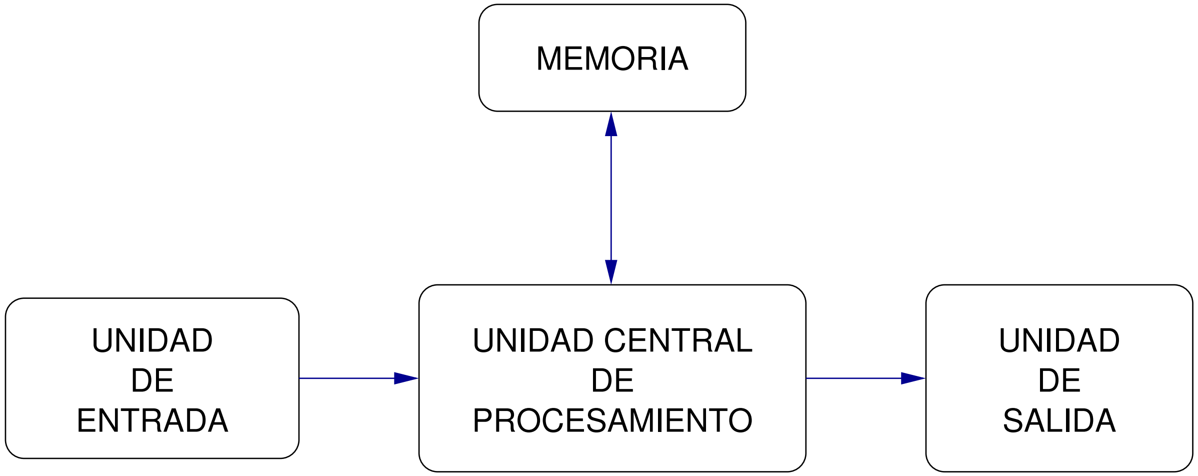
La idea central es almacenar las instrucciones del programa de una computadora en su propia memoria, logrando con ello que la máquina siga los pasos definidos por su **programa almacenado**.

Una computadora de programa almacenado tiene una configuración muy parecida al disenño original de Charles Babbage.

La unidad central de procesamiento *CPU* contiene a la unidad a-ritmética y lógica y la unidad de control.

Para poder operar bajo el modelo es necesario resolver el problema de comunicar a la computadora las operaciones por efectuar sobre los datos previamente almacenados en la memoria.

Esquema básico de una computadora actual



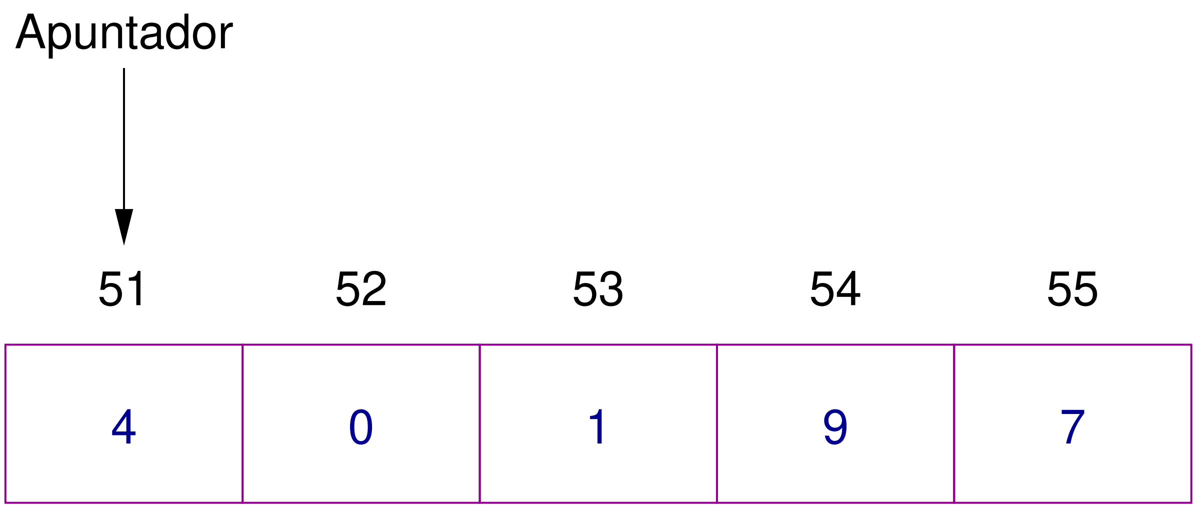
Propiedades de la memoria

La función de la memoria es guardar datos. La memoria puede ser vista como un conjunto de *celdas* con las siguientes características:

1. Cada celda puede contener un *valor numérico*
2. Cada celda tiene la propiedad de ser direccionable, se puede distinguir una de otra por medio de un número unívoco que es su *dirección*.

Las celdas de la memoria tienen que estar organizadas de modo que faciliten la localización de cualquiera de ellas con un esfuerzo mínimo. La forma más sencilla de hacer esto es organizando las celdas en forma de *vector*, es decir, las celdas son numeradas secuencialmente.

Vista de un arreglo en memoria



Operaciones elementales sobre la memoria

Leer el contenido de la memoria

1. Decidir que celda se va leer (dar la dirección).
2. Esperar a que se complete la operación de copia del dato.
3. Recoger el dato y dar por terminada la operación.

Escritura

1. Proporcionar el dato que se desea guardar.
2. Proporcionar la dirección de la celda en donde será guardado el dato.
3. Esperar un tiempo fijo para que se haga el almacenamiento.

Necesidad de la Codificación

El segundo problema es cómo alamacenar las instrucciones en la memoria.

Como en las celdas de memoria sólo caben números, entonces habrá que traducir las instrucciones a números para poder almacenarlas, que es el concepto de *codificación*.

Para codificar las instrucciones se debe considerar cuantas y cuales son las instrucciones disponibles y que esquema de codificación se empleará.

La cantidad depende básicamente de la capacidad de la unidad de control del procesador central para hacer operaciones; a mayor complejidad –y costo–, mayor será el número de instrucciones diferentes que podrá efectuar. Se debe encontrar un código adecuado para que a cada instrucción le corresponda uno, y sólo un, valor numérico. Se usa una especie de diccionario electrónico.

Ejemplo de Codificación

INSTRUCCION	CODIGO INTERNO
SUMA	57
RESTA	42
•	•
•	•
•	•

Primer Programa

Se puede escribir un primer programa completo, usando el modelo mostrado. Supongamos el problema de sumar $5 + 7$ Consideraciones:

1. Se requieren tres celdas, dos para los datos y una para dejar el resultado, sean las celdas 21, 22 y 23.
2. Se deben definir con detalle las operaciones a efectuar, el orden y la codificación adecuada.
3. Introducir todos los datos e instrucciones en memoria.

Haciendo un análisis es necesario definir varias operaciones sobre la máquina, por ejemplo: una instrucción para llevar el contenido de una celda al *registro acumulador*; otra para hacer la suma, y otra para devolver el contenido del acumulador a una celda de la memoria.

Diccionario propuesto

Instrucción Mnemnica	Código Interno	Longitud de la instrucción	Descripción
CARGA_Ac	21	2	Lleva el contenido de una celda al acumulador
GUARDA_Ac	96	2	Deposita el valor del acumulador en una celda
SUMA	57	2	Suma al acumulador el contenido de la celda de memoria descrita por la dirección
RESTA	42	2	Resta al ...

Programa desarrollado

Instrucción	Dirección	Código	Comentarios
CARGA_Ac	21	2121	Se coloca el 1er número en el acumulador
SUMA	22	5722	Se efectúa la suma
GUARDA_Ac	23	9623	El resultado queda en la casilla 23
ALTO	–	70	

De la figura se pueden apreciar dos programas: uno escrito en lenguaje mnemónico (más fácil de reconocer) y otro que está descrito en código numérico (que sólo reconoce la computadora hipotetica).

Al primero se le conoce como *programa fuente* y al otro *programa objeto*. Se puede apreciar que este programa sirve para sumar cualquier par de números.

Carga del programa objeto en memoria

Es importante elegir las casillas de la memoria que se utilizarán para almacenar el programa, se puede cargar a partir de cualquiera que esté desocupada, siempre que se esté seguro de que existen suficientes celdas secuenciales vacías, se decidió hacerlo a partir de la celda 10.

Una vez cargado, el programa objeto se verá así:

21	21	57	22	96	23	70	....	05	07	??
10	11	12	13	14	15	16	....	21	22	23

Cada celda contiene un solo número de dos dígitos. Observar que las celdas 10 y 11 contienen el número 21, en donde en la primera representa el *cdigo* de la instrucción **CARGA_Ac**, mientras que en la otra celda representa la *dirección* 21.

La Unidad de Control

Su función principal es dirigir la secuencia de pasos de modo que la computadora lleve a cabo un ciclo completo de ejecución de una instrucción, y hacer esto con todas las instrucciones de que conste el programa. Los pasos son:

1. Ir a la memoria y extraer el código de la siguiente instrucción (que estará en la siguiente celda de memoria por leer).
2. Decodificar la instrucción recién leída (determinar de qué instrucción se trata.
3. Ejecutar la instrucción.
4. Prepararse para leer la siguiente instrucción, y volver al primer paso para continuar.

La unidad de control ejecutará varias veces este ciclo de cuatro ”instrucciones alambradas”.

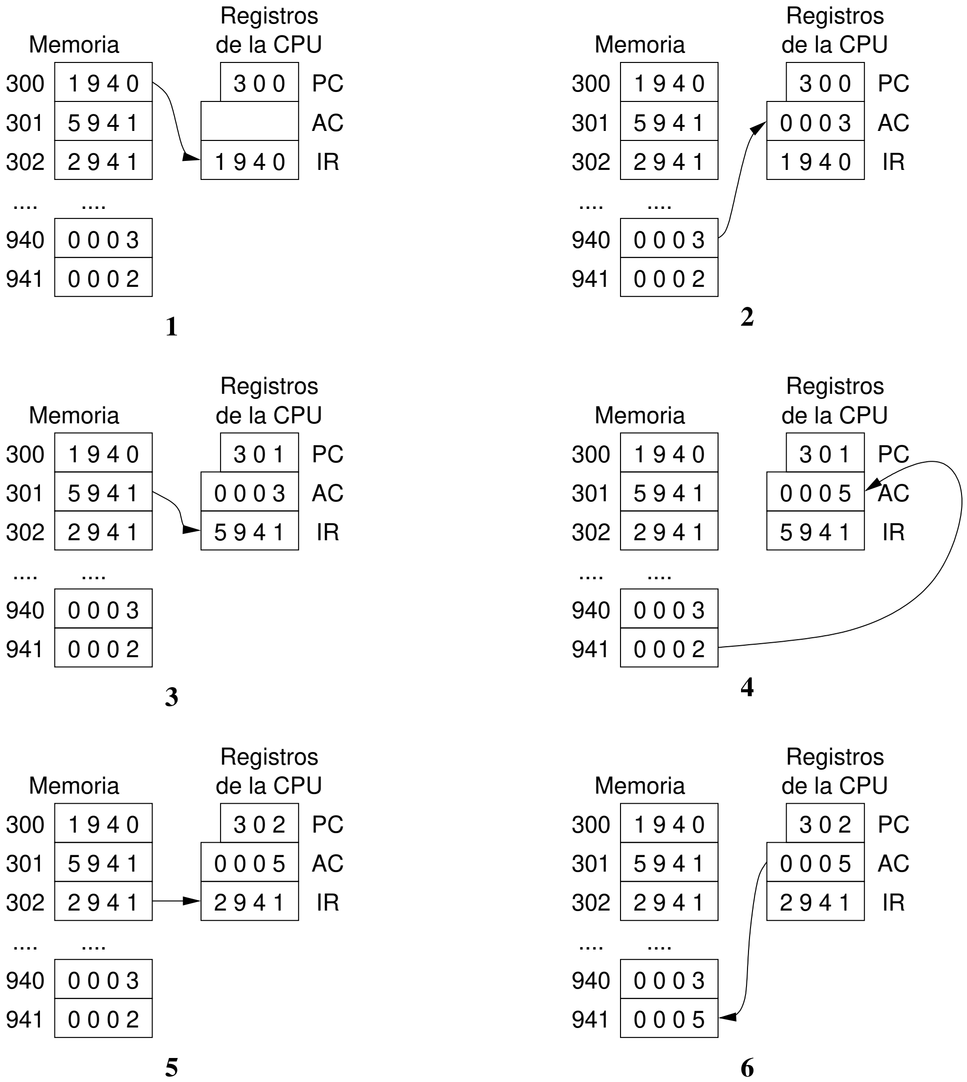
Comentarios

1. Dado el contenido de una celda, la computadora no puede distinguir si se trata de una instrucción o de un dato o de una dirección.

2. De acuerdo a lo anterior, es responsabilidad de quien maneja la máquina indicarle cuál es la celda en donde comienza el programa. La propia computadora, por medio del sistema operativo se puede encargar de esta tarea.

3. Una vez que el contador de programa apunta a la celda que contiene la primera instrucción, el resto del proceso ocurre de manera automática e invisible para el programa.

Ejemplo de ejecución de un programa


Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0002
300 PC
AC
1940 IR
1
Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0002
300 PC
0003 AC
1940 IR
2
Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0002
301 PC
0003 AC
5941 IR
3
Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0002
301 PC
0005 AC
5941 IR
4
Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0002
302 PC
0005 AC
2941 IR
5
Memoria
Registros de la CPU
300 1940
301 5941
302 2941
....
940 0003
941 0005
302 PC
0005 AC
2941 IR
6