

# Trabajo práctico N° 4

## Arquitectura y organización de computadoras

FECHA DE FINALIZACIÓN: 26 DE MAYO



Introducción a la computación  
Departamento de Ingeniería de Computadoras  
Facultad de Informática - Universidad Nacional del Comahue



**Objetivo:** Comprender la organización y el funcionamiento básico de una computadora simple. Se involucran conocimientos de los componentes hardware y sus interacciones para ejecutar instrucciones.

### Recursos bibliográfico:

- *Andrew S. Tanenbaum*. Organización de computadoras: un enfoque estructurado. Cuarta edición, editorial Pearson Educación, 2000. ISBN 970-170-399-5.

### Lectura obligatoria:

- Apuntes de cátedra. Capítulo 5: Arquitectura y Organización de Computadoras. Disponible en *PEDCO*: <https://pedco.uncoma.edu.ar/mod/url/view.php?id=203642>

## 1. Modelo Computacional Binario Elemental (MCBE)

1. Con respecto a la memoria de la *MCBE*, indique:
  - a) Cantidad de celdas de memoria.
  - b) Tamaño de una celda de memoria en *bits* y *bytes*.
  - c) Tamaño total en *bytes*.
  - d) Dirección de la primera y de última celda de memoria.
2. Con respecto a la *CPU* de la *MCBE*, indique:
  - a) Registros y sus propósitos.
  - b) ¿Qué representación y tamaño (en *bits*) de números utilizan las instrucciones aritméticas?
  - c) ¿En qué dirección de memoria debe ubicarse la primera instrucción del programa?
  - d) ¿Qué efecto tendría ubicar los datos del programa a partir de la posición de memoria 0 y las instrucciones a continuación?
3. Con respecto a la *Entrada/Salida* de la *MCBE*, indique:
  - a) ¿Para qué se utilizan las direcciones **30** y **31**? ¿Qué dispositivos podrían conectarse en esas direcciones?

- b) ¿Cuántas lecturas son necesarias si se quiere leer el dato: **0x01A397BCFF**? Y si en lugar de lecturas fueran escrituras ¿cuántas son necesarias?
4. Suponga la máquina *MCBE* en su estado inicial con contenido de memoria indicado en cada inciso (el resto de la memoria no indicada puede tener cualquier valor). Describir el efecto de la ejecución de cada una de las instrucciones del programa, desde su inicio hasta su finalización.

a)

Dirección	Contenido binario
0	0100 0100
1	1000 0100
2	0110 0100
3	0010 0000
4	0000 0011

Ejemplo de Resolución:

Búsqueda de la instrucción		Decodificación de la instrucción		Ejecución de la instrucción			
PC	IR	Cod. Op.	Operando	Acumulador	Memoria	Salida	PC
0000 0000	0100 0100	010	00100	0000 0011	-	-	0000 0001
0000 0001	1000 0100	100	00100	0000 0110	-	-	0000 0010
0000 0011	0110 0100	011	00100	-	(00100) ← 0000 0110	-	0000 0011
0000 0011	0010 0000	001	00000	-	-	-	-

b)

Dirección	Contenido binario
0	0100 0110
1	1010 1000
2	0110 0110
3	1010 0111
4	1110 0000
5	0010 0000
6	0000 1101
7	0000 1100
8	0000 0010

c)

Dirección	Contenido binario
0	01011110
1	10000101
2	10100110
3	01111111
4	00100000
5	00010100
6	00000101

d)

Dirección	Contenido binario
0	01011110
1	01100110
2	10000110
3	10100111
4	01111111
5	00100000
6	00000000
7	00000110

e)

Dirección	Contenido binario
0	01011110
1	10001011
2	01101011
3	01001001
4	10101010
5	01101001
6	11100010
7	11011001
8	00100000
9	00000100
10	00000001
11	00000000

f)

Dirección	Contenido binario
0	01011110
1	01100111
2	01000110
3	10000111
4	01101000
5	00100000
6	00000101
7	00000000
8	00000000

# Anexo

## Descripción del Modelo Computacional Binario Elemental (MCBE)

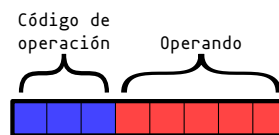
**Memoria:** consta de 32 posiciones de 8 bits. Las direcciones 0 a 29 corresponden a direcciones que pueden ser escritas y leídas. La dirección 30 es de **sólo lectura**, permite leer datos del dispositivo de entrada, por ejemplo un teclado. La dirección 31 es de **sólo escritura**, permite escribir datos en el dispositivo de salida, por ejemplo en una pantalla o una impresora.

**Registro PC:** registro de 8 bits, contiene la dirección de la próxima instrucción a ejecutar. Se inicializa en cero.

**Registro IR:** registro 8 bits donde se guarda la instrucción que se esta decodificando o ejecutando.

**Registro acumulador:** registro de 8 bits donde se almacena un número entero representado en *complemento a 2*.

**Instrucciones:** de 8 bits, los 3 bits más significativos almacenan el código de operación, y los 5 menos significativos almacenan el operando.



Código de operación <i>3 bits</i>	Operando <i>5 bits</i>	Descripción
010	<i>dirección</i>	<b>Memoria → Acumulador.</b> Copia un byte desde la dirección de memoria al acumulador.
011	<i>dirección</i>	<b>Acumulador → Memoria.</b> Copia el contenido del acumulador en esa dirección de memoria.
100	<i>dirección</i>	<b>Suma.</b> El contenido de la dirección se suma al acumulador, y el resultado se almacena en el acumulador.
101	<i>dirección</i>	<b>Resta.</b> El contenido de la dirección se resta al acumulador, y el resultado se almacena en el acumulador.
110	<i>desplazamiento</i>	<b>Salto incondicional.</b> Se suma (en complemento a 2) el desplazamiento al <b>PC</b> .
111	<i>desplazamiento</i>	<b>Salto condicional.</b> Si el acumulador es cero, se suma (en complemento a 2) el desplazamiento al <b>PC</b> , en caso contrario el <b>PC</b> se incrementa en uno.
001	<i>(sin uso)</i>	<b>Detiene la maquina.</b> No se ejecutan nuevas instrucciones. Los registros y la memoria quedan con el último valor que tenían.
000	<i>(sin uso)</i>	<b>No operación.</b> No tiene ningún efecto sobre el acumulador ni memoria. El <b>PC</b> se incrementa en uno.