

# Guía de ejercicios # 3

## Memoria, Buses, Q2 y Accesos

Organización de Computadoras 2018

UNQ

### Objetivos

Que el estudiante pueda:

- Escribir un programa utilizando los elementos de Q2 (Modo de direccionamiento directo)
- Comprender el funcionamiento de la memoria principal y los componentes del sistema que participan en la lectura y la escritura (buses, otros registros de uso específico)
- Entender para qué sirve la memoria principal desde el punto de vista del programador
- Revisar el ciclo de ejecución de cada instrucción en un programa

Para resolver esta práctica se aconseja consultar los apuntes de la materia *Memoria y buses*, disponible en <http://orga.blog.unq.edu.ar/descargas/> y

Los ejercicios marcados con ★ forman un conjunto minimal para aprender e integrar los conceptos. Los demás ejercicios son redundantes y permiten seguir entrenando

## 1 Escribir Programas en Q2

1. Hacer un programa que multiplique por 12 el valor de la celda 0x0007 ★
2. Escriba un programa que inicialice la celda 0xA000 con la cadena que representa el valor 25 en hexadecimal de 4 dígitos.
3. Escriba un programa que sume el valor de la celda 0xB000 con el valor de la celda 0x2000
4. Escriba un programa que duplique el valor de la celda 0xA305
5. Hacer un programa que sume el valor de la celda 0x7000 con el valor de R1 y guarde el resultado en la celda 0xABCD★
6. Escribir un programa que sume los valores de las celdas 0x0F80 y 0x0F81 y ponga el resultado en R2 (sin modificar las celdas).
7. Escribir un programa que calcule el **promedio** entre los valores almacenados en las celdas 0x089A y 0x089B. ★
8. Escribir un programa que intercambie los valores de las celdas 0x0F80 y 0x0F81★

9. Suponiendo que la celda 0x089A tiene almacenado el costo en pesos de un producto, y la celda 0x089B contiene el porcentaje descuento a aplicar, calcule el valor final de venta a ser almacenado en la celda 0x089C. Por ejemplo, si se un producto de \$50 y se le aplica un descuento del 20%, el resultado final es

$$50 - (50 * 20/100)$$

• No puede usar registros para resolverlo. ★

### 1.1 Probar los programas

Para estar seguro de que un programa cumple con su objetivo, se realizan **pruebas de escritorio** que son ejecuciones paso a paso. Considerar por ejemplo el siguiente programa

```
MOV R6, R0
ADD R6, R1
MUL R6, 0x0010
```

Si se toma como precondition R0=0x0060 y R1=0x0025 entonces la ejecución paso a paso es como sigue:

Instrucción	Efecto
MOV R6, R0	R6 ← 0060
ADD R6, R1	R6 ← 0085
MUL R6, 0x0010	R7,R6 ← 0000 0850

10. Hacer una prueba de escritorio para el programa del ejercicio 7 (promedio)
11. Hacer una prueba de escritorio para el programa del ejercicio 8 (intercambio).
12. Hacer una prueba de escritorio para el programa del ejercicio 9 (descuento). ★

## 2 Ensamblar programas

Para incorporar un nuevo modo de direccionamiento se necesita modificar la tabla de modos de direccionamiento como sigue:

Modo	Codificación
Inmediato	000000
<b>Directo</b>	001000
Registro	100rrr

donde **rrr** es una codificación (en 3 bits) del número de registro.

Además, el siguiente es el formato de las instrucciones de **Q2**, de dos operandos (origen y destino). Es importante destacar en comparación con **Q1**, que se agrega un campo mas: **Destino**.

Cod.Op (4b)	Modo Destino (6b)	Modo Origen (6b)	Destino (16b)	Origen (16b)
----------------	----------------------	---------------------	------------------	-----------------

Con este formato, el campo **Destino** puede contener una dirección de memoria (si el modo correspondiente es *directo*) o no se utiliza si es un registro. Por otro lado, el campo **Origen** puede contener una dirección de memoria, un registro o una constante (modo *inmediato*). Algunos ejemplos:

Código fuente	Código máquina
MOV R7, [0x0001]	0001 100111 001000 0000000000000000
ADD [0x0001], R7	0010 001000 100111 0000000000000000

**Nota:** Es bueno tener a mano la especificación completa de **Q** (ver sección **Referencias** al final)

### Ejercicios

- Suponer la instrucción `MOV [0x0001], R0`
  - Ensamblarla
  - Si se la carga en la memoria a partir de la celda 0000, ¿Qué celdas ocupa?
- Suponer la instrucción `MUL [0x00FE], 0x00A1`
  - Ensamblarla
  - Si se la carga en la memoria a partir de la celda 0000, ¿Qué celdas ocupa?
- ★ Suponer el siguiente programa
 

```
MUL [0x00FE], 0x00A1
      ADD [0xFFAB], [0xBBA7]
      SUB R0, [0x2DC6]
```

  - Ensamblarlo
  - Si se la carga en la memoria a partir de la celda 0000, ¿Qué celdas ocupa?
- ★ A partir del siguiente mapa de memoria:

	...
0x9999	29C8
0x999A	A0A0
	...

y sabiendo que a partir de la primer celda (0x9999) hay ensamblada una instrucción.

- ¿Cuál es dicha instrucción?
  - ¿Cuántas celdas ocupa?
  - ¿Qué se modifica al ejecutarla?
- A partir del siguiente mapa de memoria:

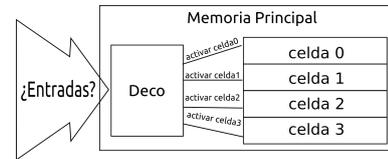
	...
0x9999	1200
0x999A	FFFF
0x999B	0000
	...

y sabiendo que a partir de la primer celda (0x9999) hay ensamblada una instrucción.

- ¿Cuál es dicha instrucción?
- ¿Cuántas celdas ocupa?
- ¿Qué se modifica al ejecutarla?

## 3 Memoria

- ★ Considerar el siguiente esquema del circuito de la memoria principal



- ¿Cuántas entradas debería tener el decodificador?
- Teniendo en cuenta lo analizado en el ejercicio 18, te parece que las direcciones se guardan en la memoria?
  - Generalizando tu respuesta al ejercicio 18, cuantas entradas se necesitan para **direccionar n celdas**?
  - ★ ¿Es cierto que en la memoria se almacenan solamente las instrucciones mientras que los datos en registros del CPU? Justificar con un ejemplo
  - ¿Es cierto que el contenido de la memoria no es volátil? Justificar
  - ¿Porqué se dice que en la memoria RAM la lectura de cualquier dirección insume el mismo tiempo?
  - ★ Sabiendo que 1 byte equivale a 8 bits, y suponiendo una memoria principal con tamaño de celda de **2** bytes y direcciones de 16 bits. ¿Qué cantidad de bits puede almacenar?
  - Sabiendo que 1 byte equivale a 8 bits, y suponiendo una memoria principal con tamaño de celda de **2** bytes y tamaño total de **64** bytes. ¿Cuántas celdas tiene?

## 4 Buses del sistema

Para cada instrucción completaremos un **cuadro de actividad de buses** que describe el contenido de los buses en cada operación de lectura o escritura a memoria al *ejecutar todo el ciclo de la instrucción*. Para todas las instrucciones asumir:

- que están ensambladas a partir de la celda **A000**,
- el siguiente estado parcial de registros y memoria

R0	0001	9000	...
R1	9002	9001	AB02
R2	9004	9002	9004
R3	0003	9003	0043
		9004	BBBB
			OFFF
			...

Como ejemplo, considerar la instrucción:  
MOV R0, [0x9001].

Bus de ctrl	Bus de dir	Bus de datos
L=1	A000	1808 (codigo maquina)
L=1	A001	9001 (codigo maquina)
L=1	9001	9004 (operando)

**Nota:** en el bus de control se debe indicar L=1 cuando es una lectura o E=1 cuando es una escritura

## Ejercicios

26. Hacer el cuadro de actividad para MOV [0x9001], R0
27. Hacer el cuadro de actividad para ADD R0, [0x9000]
28. Hacer el cuadro de actividad para ADD [0x9000], R0
29. Hacer el cuadro de actividad para ADD [0x9001], [0x9002]
30. Hacer el cuadro de actividad para SUB [0x9001], [0x9002]
31. Si la memoria tiene 8 celdas, cada una de 1 byte:
  - (a) ¿Cuántas líneas de direcciones se necesitan?
  - (b) ¿Cuántas líneas de datos se necesitan?
32. Sabiendo que 1 byte equivale a 8 bits, suponga una memoria principal con tamaño de celda de **2** bytes y tamaño total de **64** bytes. ¿Cuántos bits tendrá el bus de direcciones?
33. Si se quiere cuadruplicar el tamaño de una memoria ¿Cómo debe modificarse el bus de direcciones? (No es posible alterar el bus de datos)
34. Si se quiere cuadruplicar el tamaño de una memoria ¿Cómo debe modificarse el bus de datos? (No es posible alterar el bus de direcciones)
35. La arquitectura **Q2** tiene un bus de direcciones y un bus de datos ambos de **16 bits**. ¿Qué capacidad tiene la memoria principal?

## 5 Ciclo de ejecución de instrucción

Par los siguientes ejercicios asumir PC=9999

36. Suponer el siguiente mapa de memoria:

0x9999	...
	39E0
	...

- (a) ¿Qué celdas se **leen** durante la **busqueda de instrucción**?

- (b) ¿Qué celdas se **leen** durante la **busqueda de operandos**?
- (c) ¿Qué celdas se **escriben** durante la **almacenamiento de resultados**?

37. ★ Suponer el siguiente mapa de memoria:

0x9999	...
0x999A	29C0
	A0A0
	...

- (a) ¿Qué celdas se **leen** durante la **busqueda de instrucción**?
- (b) ¿Qué celdas se **leen** durante la **busqueda de operandos**?
- (c) ¿Qué celdas se **escriben** durante la **almacenamiento de resultados**?

38. ★ Suponer el siguiente mapa de memoria:

0x9999	...
0x999A	29C8
	A0A0
	...

- (a) ¿Qué celdas se **leen** durante la **busqueda de instrucción**?
- (b) ¿Qué celdas se **leen** durante la **busqueda de operandos**?
- (c) ¿Qué celdas se **escriben** durante la **almacenamiento de resultados**?

39. ★ Suponer el siguiente mapa de memoria:

0x9999	...
0x999A	1200
0x999B	FFFF
	0000
	...

- (a) ¿Qué celdas se **leen** durante la **busqueda de instrucción**?
- (b) ¿Qué celdas se **leen** durante la **busqueda de operandos**?
- (c) ¿Qué celdas se **escriben** durante la **almacenamiento de resultados**?

40. Analice las siguientes instrucciones en forma individual indicando **qué** celdas de memoria son leídas y escritas en cada etapa del ciclo. Asuma que las instrucciones están alojadas consecutivamente a partir de 0x0000.

Instrucción	BI	BO	AR
ADD [0xC0C0], R7	0x0000, 0x0001	C0C0	C0C0
ADD R2, R5	0x0003		
MOV [0xA001], R0			
SUB R0, 0x0005			
ADD [0x9123], R3			
MUL [0x3401], [0xA001]			

41. Idem anterior, pero además indicar el valor del Operando destino en la columna **efecto**, dado el siguiente estado de memoria y registros.

R0	0001	9000	...
R1	9002	9001	AB02
R2	9004	9002	9004
R3	0003	9003	0043
		9004	BBBB
			OFFF
			...

Instrucción	BI	BO	AR	Efecto
MOV R0, [0x9000]				
ADD [0x9001], R1				
MOV R1, 0x9002				

42. ★ Considerando el siguiente estado parcial de registros y memoria:

R5	0001	PC	9999	...
				0x9999
				0x999A
				0x999B
				0x999C
				0x999D
				...
				0xAAAA
				0xAAAB
				...

Simular la ejecución del programa (que comienza donde indica el registro PC) para completar el siguiente cuadro de actividad de buses, indicando además **con qué etapa del ciclo se corresponde cada acceso**.

Tomar como ejemplo el siguiente cuadro para la instrucción `SUB [0xABCD],R5`, ensamblada en la celda 0000, y siendo `R5=0001` y `[ABCD]=000A`

Bus de ctrl	Bus de dir	Bus de datos	Etapas
L=1	0000	3225	B.I.
L=1	0001	ABCD	B.I.
L=1	ABCD	000A	B.O.
E=1	ABCD	0009	A.R.

43. Teniendo en cuenta lo resuelto en el último ejercicio ¿En qué etapas del ciclo de ejecución se accede a la memoria?

## 6 Tiempo de ejecución

En los siguientes ejercicios compararemos diferentes programas a partir del tiempo que lleva su ejecución. Para cada uno responder:

- a ¿Cuántos bytes (1 byte = 8 bits) ocupa una vez ensamblado?
- b ¿Cuántos accesos a memoria se llevan a cabo durante su ejecución?
- c Si cada acceso a memoria se realiza en  $0.1\mu s$ , y el tiempo de computo de CPU es despreciable. ¿Cuanto tarda la ejecución del programa?

44. MOV R0,0xE1E1  
ADD R0,R1
45. MOV R0,[0xE1E1]  
ADD R0,0x0111  
MUL [0x67AB],R0
46. ADD [0x5001],R6  
MUL [0x5000],[0x5001]
47. MOV R6,0x9867  
ADD [0xA000],R6  
MUL [0xA001],[0xA000]

## 7 Ejercicios adicionales

48. Ensamblar el siguiente programa a partir de la celda E000:

```
MOV R0, [0x9000]
ADD [0x9001], R1
MOV R1, 0x9002
```

49. A partir del siguiente mapa de memoria:

...
0x9999
...
39E0
...

y sabiendo que a partir de la primer celda (0x9999) hay ensamblada una instrucción.

- (a) ¿Cuál es dicha instrucción?
- (b) ¿Cuántas celdas ocupa?
- (c) ¿Qué se modifica al ejecutarla?
50. ¿Cuántos bits transmite a la vez una línea de un bus?
51. ¿Es cierto que el bus de datos está relacionado con cuantas celdas tiene una memoria?
52. ¿Es cierto que el bus de direcciones lleva señales tales como: leer, escribir, quiero usar el bus?
53. Completar la siguiente tabla de **cantidad de accesos**, que es similar a las de la sección 5 pero se pide que se complete la cantidad de accesos a memoria (no los registros!)

Instrucción	BI	BO	AR	Total
DIV R0, 0x9876	2	0	0	2
MOV R0, R1				
ADD R0, 0xFOCA				
SUB [0x1111], 0x1111				
MUL [0x0010], [0xFEDE]				
DIV R1, [0x43AE]				

54. Dado el siguiente estado de registros y memoria:

R0	000F	00AB	R4	0x2000	...
R1	0013	000A	R5	0x2001	2001
R2	0085	0009	R6		FF00
R3	00E2	2001	R7	0x200B	:
					0050
					...

- (a) ¿Qué registro o celda de memoria se modifica al ejecutar `ADD R0, [0x2000]`? ¿Qué valor toma?
- (b) ¿Qué registro o celda de memoria se modifica al ejecutar `ADD R3, 0x2000`? ¿Qué valor toma?
- (c) ¿Qué registro o celda de memoria se modifica al ejecutar `ADD [0x2000], R3`? ¿Qué valor toma?
- (d) ¿Qué registro o celda de memoria se modifica al ejecutar `ADD [0x2001], [0x2000]`? ¿Qué valor toma?
- (e) ¿Qué registro o celda de memoria se modifica al ejecutar `ADD R3, R7`? ¿Qué valor toma?

**Nota:** Analice cada caso **independientemente**, no considere las instrucciones como un programa.

## References

- [1] Williams Stallings, *Computer Organization and Architecture*, octava edición, Editorial Prentice Hall, 2010. **Capítulos 10 y 11**
- [2] Especificación completa de Q: <http://orga.blog.unq.edu.ar/wp-content/uploads/sites/5/2015/08/Especificacion-Q.pdf>