

Taller de Ciclo de Instrucción

Organización del Computador 1

Segundo Cuatrimestre 2022

El presente taller consiste en una serie de ejercicios en los cuales se deberá realizar en papel el ensamblado de diversos códigos fuente y luego sus correspondientes seguimientos utilizando las *Planillas de Seguimiento*.

Será posible comparar, analizar y validar los seguimientos realizados en papel utilizando el *Simulador de la Máquina ORGA1*.

Planilla de seguimiento: Reglas de notación

Cuando un registro pasa a tener un nuevo valor, la celda que muestra el valor viejo debe ser tachada.

Todos los valores de los registros y las posiciones de memoria que no se definan explícitamente, comienzan con valor 0.

Los FLAGS deben escribirse *únicamente* después de que la instrucción ejecutada los modifique. Es decir, si la instrucción no los modifica, las celdas de flags deben quedar en blanco.

Los desplazamientos se expresan en complemento a 2 de 8 *bits*. No pueden aparecer en decimal.

Toda instrucción escrita en la columna de *Instrucción Decodificada* debe corresponder a la decodificación de la palabra presente en el registro IR según la *Cartilla de la máquina Orga 1*.

En la celda de *Ejecución* debe mostrarse cómo se produce la actualización de todos los registros afectados y los pasos intermedios que considere necesarios.

Para toda instrucción de salto debe indicarse explícitamente en la celda de *Ejecución* si se produce o no, en caso de que sí, su justificación.

Al intentar decodificar una instrucción inválida, debe figurar Instrucción Inválida en la columna de *Instrucción Decodificada* y FIN en la celda de *Ejecución*.

El simulador

El simulador se puede bajar desde la página de la materia. Requiere Java 1.5 o superior. Para ejecutarlo, teclear en una consola `java -jar simulador.jar`.

Para correr un programa tenemos que agregar el archivo con el código (extensión .asm), luego ensamblarlo (esto traduce y calcula etiquetas y saltos), y finalmente cargarlo en la memoria y encender la máquina.

Además de lo dicho en el apunte de la Arquitectura ORGA1 tenemos que tener en cuenta que todo programa en lenguaje ensamblador de ORGA1 tendrá que tener necesariamente una etiqueta llamada *main*. Dicha etiqueta indicará el contenido inicial del PC al encender la máquina.

Ejercicios

Para los próximos códigos realizar el seguimiento con la planilla y verificar el resultado con el simulador de la máquina ORGA1.

Ejercicio 1

```
main :      MOV R1 , 0 x7000
            ADD R1 , 0 x6000
            JE cero
            JL dioNegativo
            JVS huboOverFlow
cero :      DW 0 x0FE0
huboOverFlow :  MOV R2 , 0 x0030
                SUB R2 , 0 x00F0
                JVS huboOverFlow
                JE cero
                JL dioNegativo
                DW 0 x0FE0
dioNegativo :  CMP R2 , 0 xFF40
                JE finOk
                DW 0 x0FE0
finOk :  DW 0 x0000
```

Realizar el seguimiento, 13 ciclos de instrucción, respondiendo las siguientes preguntas, usando el simulador para verificar las respuestas sabiendo que el programa fue ensamblado y cargado desde la posición de memoria 0x8C00:

a) ¿Qué dirección va a tener cada etiqueta?

main:

8C00
8C02
8C04
8C05
8C06

cero:

8C07

huboOverFlow:

8C08
8C0A
8C0C
8C0D
8C0E
8C0F

dioNegativo:

8C10
8C12
8C13

finOK:

b) ¿Cuál será el desplazamiento que el ensamblador deberá poner en cada salto condicional?

c) ¿Termina el programa? (¿Se decodifica una instrucción inválida?) ¿Cuál es el último valor que toma el PC?

Ejercicio 2

```
main : MOV R1 , 0 x4228
llama : CALL subrut
        DW 0 x6008
        DW 0 xFEDE
        DW 0 xF450
        DW 0 xDEAD
subrut : DW 0 x3840
        DW 0 x2114
        JNE llama
        DW 0 xC000
```

Realizar el seguimiento, 10 ciclos de instrucción, respondiendo las siguientes preguntas, usando el simulador para verificar las respuestas sabiendo que el programa fue ensamblado y cargado desde la posición de memoria 0xFA00:

a) ¿Qué dirección va a tener cada etiqueta?

main:

FA00

FA01

llama:

FA02

FA03

FA04

FA05

FA06

FA07

subrut:

FA08

FA09

FA0A

b) ¿Cuál será el desplazamiento que el ensamblador deberá poner en el salto condicional?

JNE - 009

c) ¿Termina el programa? (¿Se decodifica una instrucción inválida?) ¿Cuál es el último valor que toma el PC? ¿Y el SP?

Sí, sí el último valor que toma el PC es FA07 y el stack pointer es FFEE

Ejercicio 3

Se tiene el siguiente programa cargado en memoria a partir de la posición 0x0001.

```

main :      MOV R2 , 0 x0004
            MOV R1 , [ R2 ]
            CMP R1 , [[0 x0004 ]]
            JE main
            JNE seguir
            DW 0 x0034
seguir :    ADD [ R2 + 0 x0008 ] ,
            0 xA64B
            SUB R1 , R2

```

i. Realice su seguimiento, 7 ciclos de instrucción, asumiendo que inicialmente el PC vale 0x0001.

ii. Suponga que el programa hubiese sido cargado en la posición 0x0000 y el PC comienza con ese valor. ¿Cambia la ejecución del programa? ¿De qué manera? ¿Por qué?