

# Taller de Ciclo de Instrucción

## Segunda Parte

### Organización del Computador 1

Primer Cuatrimestre 2021

## 1. Introducción

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*.

## 2. 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 corresponderse 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*.

## 3. 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.

## 4. Ejercicios

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

### 4.1. Ejercicio 1

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

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:

- ¿Qué dirección va a tener cada etiqueta?
- ¿Cuál será el desplazamiento que el ensamblador deberá poner en el salto condicional?
- ¿Termina el programa? (¿Se decodifica una instrucción inválida?) ¿Cuál es el último valor que toma el PC? ¿Y el SP?

### 4.2. Ejercicio 2

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

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

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