

default



# TallerIAS

Marco Marcillo

23 de diciembre de 2024

Outline

## Índice

<b>1. Recursos</b>	<b>2</b>
1.1. Observaciones y Recomendaciones . . . . .	2
<b>2. Instrucciones</b>	<b>2</b>
<b>3. Taller Máquina de Von Neuman</b>	<b>2</b>
3.1. Conjunto de Instrucciones IAS . . . . .	3
3.2. Ejercicio . . . . .	4
<b>4. Escriba el código Assembler del programa que ejecuta el computador IAS.</b>	<b>4</b>

## 1. Recursos

Para este taller consulte el libro *Organización y arquitectura de computadores 7ma Edición* disponible en EPN-SHAREPOINT(Clic Me) y use su WSL con Emacs para editar el archivo.

### 1.1. Observaciones y Recomendaciones

- Tenga en cuenta la ruta a las imágenes y a la bibliografía. Se le

recomienda que realice una clonación o un fork del repositorio de la clase a fin de que no tenga problemas generando el documento.

- Instale  $\text{\LaTeX}$  completo en el computador que generará el archivo pdf.

## 2. Instrucciones

Complete las solicitudes del taller y suba el archivo .ORG y .PDF al aula virtual. Cuando trabaje en Emacs trate de utilizar las combinaciones de teclas para navegar por el documento.

## 3. Taller Máquina de Von Neuman

La máquina de Von Neuman realiza operaciones en un ciclo repetitivo de captación y ejecución. Para esto, el computador IAS usa una memoria de 40 bits. Cuando la memoria se usa numéricamente, se representa el número en complemento a 2, reservando el bit inicial para el signo. Cuando la memoria se usa para el registro de instrucciones, se divide en dos partes de 20 bits. Los 8 primeros bits de cada parte corresponden al *opcode*

y los restantes 12 bits de cada parte a la dirección (operando) desde la que se debe leer o a la que se debe escribir.

### 3.1. Conjunto de Instrucciones IAS

Complete la Tabla 2.1 de instrucciones del computador IAS (Stallings et al., 2006, p.47) usando la notación RTL<sup>1</sup>. Se suministra la Tabla en código  $\text{\LaTeX}$  para tal efecto. Edite el archivo en Emacs usando el modo principal `ORG` y cambie al modo  $\text{\LaTeX}$  para editar la Tabla. El cambio de modo se realiza usando `M-x latex-mode RET` y para retornar al modo `ORG` haga `M-x org-mode RET`

- M equivale a *Alt*
- x es la tecla *x*
- RET es presionar *enter*

Una vez activado el modo  $\text{\LaTeX}$  el siguiente código recibe colores sobre las palabras clave.

Cuadro 1: Instrucciones Maquina IAS

Opcode	Opcode Hex	Simbolo	RTL
00001010	0xA	LOAD MQ	$[AC] \leftarrow [MQ]$
00001001	0x9	LOAD MQ, M(X)	$[MQ] \leftarrow [X]$
00100001	0x21	STOR M(X)	$[X] \leftarrow [AC]$
00000010	0x02	LOAD M(X)	$[AC] \leftarrow [X]$
00000011	0x03	LOAD $ M(X) $	$[AC] \leftarrow  [X] $
00000100	0x04	LOAD $- M(X) $	$[AC] \leftarrow - [X] $
00001101	0xD	JUMP M(X.0:19)	$[PC] \leftarrow [X]$
00001110	0xE	JUMP M(X.20:39)	$[PC] \leftarrow [X]$
00001111	0xF	JUMP $M+(X.0:19)$	$Si[AC] \geq 0, [PC] \leftarrow [X]$
00010000	0x10	JUMP $M+(X.20:39)$	$Si[AC] \geq 0, [PC] \leftarrow [X]$
00000101	0x05	ADD M(X)	$[AC] \leftarrow [AC] + [X]$
00000110	0x06	SUB M(X)	$[AC] \leftarrow [AC] - [X]$
00000111	0x07	ADD IM(X)	$[AC] \leftarrow [AC] +  [X] $
00001000	0x08	SUB IM(X)	$[AC] \leftarrow [AC] -  [X] $
00001011	0xB	MUL M(X)	$[AC] \leftarrow [AC] * [X]$
00001100	0xC	DIV M(X)	$[AC] \leftarrow [AC] / [X]$
00010010	0x12	STOR M(X.8:19)	$[X] \leftarrow [AC]$
00010011	0x13	STOR M(X.28:39)	$[X] \leftarrow [AC]$
00010100	0x14	LSH	$[AC] \leftarrow [AC] * 2$
00010101	0x15	RSH	$[AC] \leftarrow [AC] / 2$

<sup>1</sup>Register Transfer Language o Lenguaje de Transferencia de Registros

### 3.2. Ejercicio

En la máquina IAS, las instrucciones se dividen en dos segmentos: izquierdo desde el bit 0 a 19 y derecho desde el bit 20 al 39. Primero se ejecuta el lado izquierdo (bits 0 a 19) y luego el derecho. El contador de programa inicia en la posición 300. El set de instrucciones del computador o ISA<sup>2</sup> está definido en la Tabla 1. A fin de terminar el programa se agrega la instrucción 0x99 para terminar el programa (i.e. HALT). Conteste las siguientes preguntas considerando la Tabla 2 como el mapa de Memoria de la máquina y obtenga el programa en Asmembre y en RTL.

si se tiene el siguiente mapa de memoria?

Si es un mapa de memoria ya que posee:

- Cada word es de 40 bits.
- Cada word se subdivide en 2 instrucciones del 20 bits en este caso del [0 al 19] y del [20 al 39].
- Tenemos la instruccion opcode HALT la cual finaliza el programa.
- Entre otras.

Cuadro 2: Mapa de Memoria

Dirección	<i>Opcode<sub>1</sub></i>	<i>X<sub>1</sub></i>	<i>Opcode<sub>2</sub></i>	<i>X<sub>2</sub></i>
0x300	0x01	0x940	0x06	0x941
0x301	0x21	0x940	0x99	0x000
...	...	...	...	...
0x940	0x00	0x000	0x00	0x005
0x941	0x00	0x000	0x00	0x002

¶label=sec:org8581f17]Solucion • Paso 1 (Opcode1)

LOAD M(940) AC  $\leftarrow$  0x003

PC 0x300 AC 0x003 IR 0x940

• Paso 2 (Opcode2)

STORE M(941) M(941)  $\leftarrow$  0x003

PC 0x300 AC 0x003 IR 0x941

• Paso 3 (Opcode1)

ADD M(941) AC  $\leftarrow$  0x003 + 0x003 AC  $\leftarrow$  0x006

PC 0x301 AC 0x003 IR 0x941

• Paso 4 (Opcode2)

HALT

Detiene la ejecución

¶label=sec:orgbd47b58]Preguntas

1. ¿Qué resultado se tiene en el registro del acumulador?

El acumulador al final tiene la ejecución 0x006.

2. ¿Se sobrescribe algún registro como resultado de la ejecución? Sí, se sobrescribe la dirección de memoria 0x941 con el valor 0x003

¶label=sec:org6305171]Código Assembler:

## 4. Escriba el código Assembler del programa que ejecuta el computador IAS.

- LOAD M(940) --> Cargar contenido de 0x940 en AC

<sup>2</sup>Instruction Set Architecture: los códigos de programación

- STORE M(941) --> Almacenar contenido de AC en 0x941
- ADD M(941) --> Sumar contenido de 0x941 a AC
- HALT --> Detener la ejecución)

¶label=sec:orge4c0327]Código RTL Escriba en notación de transferencia de registros el programa que ejecuta el computador

IAS  $[AC] \leftarrow [940]$

$AC \leftarrow M[940]$  -> Cargar contenido de 0x940 en el AC(acumulador)

$M[941] \leftarrow AC$  -> Almacenar contenido de AC en 0x941

$AC \leftarrow AC + M[941]$  -> Sumar contenido de 0x941 al acumulador

HALT(0x99) -> Detener la ejecución

## Referencias

Stallings, W., Vargas, A. C., & Espinosa, A. P. (2006). *Organización y arquitectura de computadores 7ma Edición*. Pearson Educación.