



TallerIAS

Lenin G. Falconí

11 de diciembre de 2025

Índice

1. Recursos	1
1.1. Observaciones y Recomendaciones	1
2. Instrucciones	1
3. Taller Máquina de Von Neuman	1
3.1. Conjunto de Instrucciones IAS	2
3.2. Ejercicio	3
3.2.1. Preguntas	3
3.2.2. Código Assembler:	3
3.2.3. Código RTL	3

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 L^AT_EX 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 RTL¹. Se suministra la Tabla en código LATEX para tal efecto. Edite el archivo en Emacs usando el modo principal ORG y cambie al modo 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 LATEX el siguiente código recibe colores sobre las palabras clave.

Cuadro 1: Instrucciones Maquina IAS

Opcode	Opcode Hex	Simbolo	RTL
00001010	0x0A	LOAD MQ	$[AC] \leftarrow [MQ]$
00001001	0x09	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] $

¹Register Transfer Language o Lenguaje de Transferencia de Registros

3.2. Ejercicio

En la máquina IAS, las instrucciones se dividen en dos segmentos: i) izquierdo, desde el bit 0 a 19, y ii) derecho, desde el bit 20 al 39. Primero se ejecuta el lado izquierdo y luego el derecho. El contador de programa inicia en la posición 300. El set de instrucciones del computador o ISA² está definido en la Tabla 1. A fin de terminar el programa se agrega la instrucción 0x99 que lleva al computador al estado de HALT. Conteste las siguientes preguntas considerando la Tabla 2 como el mapa de Memoria de la máquina y obtenga el programa en lenguaje ensamblador y en RTL.

Cuadro 2: Mapa de Memoria				
Dirección	<i>Opcode₁</i>	<i>X₁</i>	<i>Opcode₂</i>	<i>X₂</i>
0x300	01	940	06	941
0x301	21	940	14	000
0x302	21	941	99	000
...
0x940	00	000	00	005
0x941	00	000	00	002

3.2.1. Preguntas

1. ¿Qué resultado se tiene al final en el registro del acumulador?
2. ¿Se sobrescribe algún registro como resultado de la ejecución? Si verdadero, indique qué registro y con qué valor.
3. ¿Cuál es el valor del bit del Carry? ¿Qué operación modifica el bit?

3.2.2. Código Assembler:

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

LOAD M(940)

3.2.3. Código RTL

Escriba en notación de transferencia de registros el programa que ejecuta el computador IAS
[AC] ← [940]

Referencias

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

²Instruction Set Architecture: los códigos de programación