



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

IIC2343 - Arquitectura de computadores  
2do semestre del 2015

# Informe Entrega 2

## Grupo 1

### Felipe Aguayo

Implementó el Control Unit

### Carlos García

Corrigió ALU e implementó Mux A, Mux B y Registro Status.

### Nicolás Gebauer

Corrigió ALU, hizo conexiones en Basys 3, implementó Control Unit y trabajó en el debug del Assembler.

### Andrés Matte

Programó el Assembler y trabajó en el correspondiente debug.

### Dasen Razmilic

Implementó RAM, ROM y PC.

# Estructura de Instrucciones de la CPU

A continuación se presenta la especificación de la estructura de las instrucciones de la CPU, es decir, la función de cada uno de los 33 bits de una instrucción:

## Salida Instruction Memory:

- **Literal - Opcode**  
00000000000000000000000000000000
- **Dirección de instrucción**  
00000000000000000000000000000000

## Salida Control Unit:

El opcode es analizado por el Control Unit, el cual genera las señales de control en un bus de la siguiente manera:

- Lpc: **Load PC**  
000000000000000000000000
- La: **Load A**  
000000000000000000000000
- Lb: **Load B**  
000000000000000000000000
- Sa0 y Sa1: **Selector Mux A**  
000000000000000000000000
- Sb0 y Sb1: **Selector Mux B**  
000000000000000000000000
- Sop0, Sop1, Sop2: **Selector ALU**  
000000000000000000000000
- Sadd0 y Sadd1: **Selector Mux Address**  
000000000000000000000000
- Sdin0: **Selector Mux DataIn**  
000000000000000000000000
- Spc0: **Selector Mux PC**  
000000000000000000000000
- W: **Write RAM**  
000000000000000000000000
- IncSp: **Incremento SP**  
000000000000000000000000
- DecSp: **Decremento SP**  
000000000000000000000000

Luego simplemente conectamos cada parte del bus, es decir, cada señal de control, con el sistema que controla.

## **Tabla de Instrucciones**

La Tabla de instrucciones es la que se puede apreciar en el Cuadro 1: Instrucciones soportadas por la CPU.

## **Assembler**

Para utilizar el assembler se debe arrastrar el archivo de prueba que contiene el código assembly al archivo Assembler.exe . Este último generará un archivo output.txt que tendrá el código en lenguaje vhdl que se debe copiar y pegar en el proyecto.

Cuadro 1: Instrucciones soportadas por la CPU

Instrucción	Operadores	Opcode	Instrucción	Operadores	
MOV	A B	0000000	XOR	AB	0100111
	B A	0000001		B A	0101000
	A Lit	0000010		A Lit	0101001
	B Lit	0000011		B Lit	0101010
	A Dir	0000100		A Dir	0101011
	B Dir	0000101		B Dir	0101100
	Dir A	0000110		Dir	0101101
	Dir B	0000111	NOT	A	0101110
	A DirB	0001000		B A	0101111
	B DirB	0001001		Dir A	0110000
ADD	DirB A	0001010	SHL	A	0110001
	A B	0001011		B A	0110010
	B A	0001100		Dir A	0110011
	A Lit	0001101	SHR	A	0110100
	B Lit	0001110		B A	0110101
	A Dir	0001111		Dir A	0110110
	B Dir	0010000	INC	A	0110111
	Dir	0010001		B	0111000
SUB	A B	0010010		Dir	0111001
	B A	0010011	DEC	A	0111010
	A Lit	0010100	CMP	A B	0111011
	B Lit	0010101		A Lit	0111100
	A Dir	0010110		A Dir	0111101
	B Dir	0010111	JMP	Dir	0111110
	Dir	0011000	JEQ	Dir	0111111
AND	A B	0011001	JNE	Dir	1000000
	B A	0011010	JGT	Dir	1000001
	A Lit	0011011	JGE	Dir	1000010
	B Lit	0011100	JLT	Dir	1000011
	A Dir	0011101	JLE	Dir	1000100
	B Dir	0011110	JCR	Dir	1000101
	Dir	0011111	NOP		1000110
OR	AB	0100000			
	B A	0100001			
	A Lit	0100010			
	B Lit	0100011			
	A Dir	0100100			
	B Dir	0100101			
	Dir	0100110			