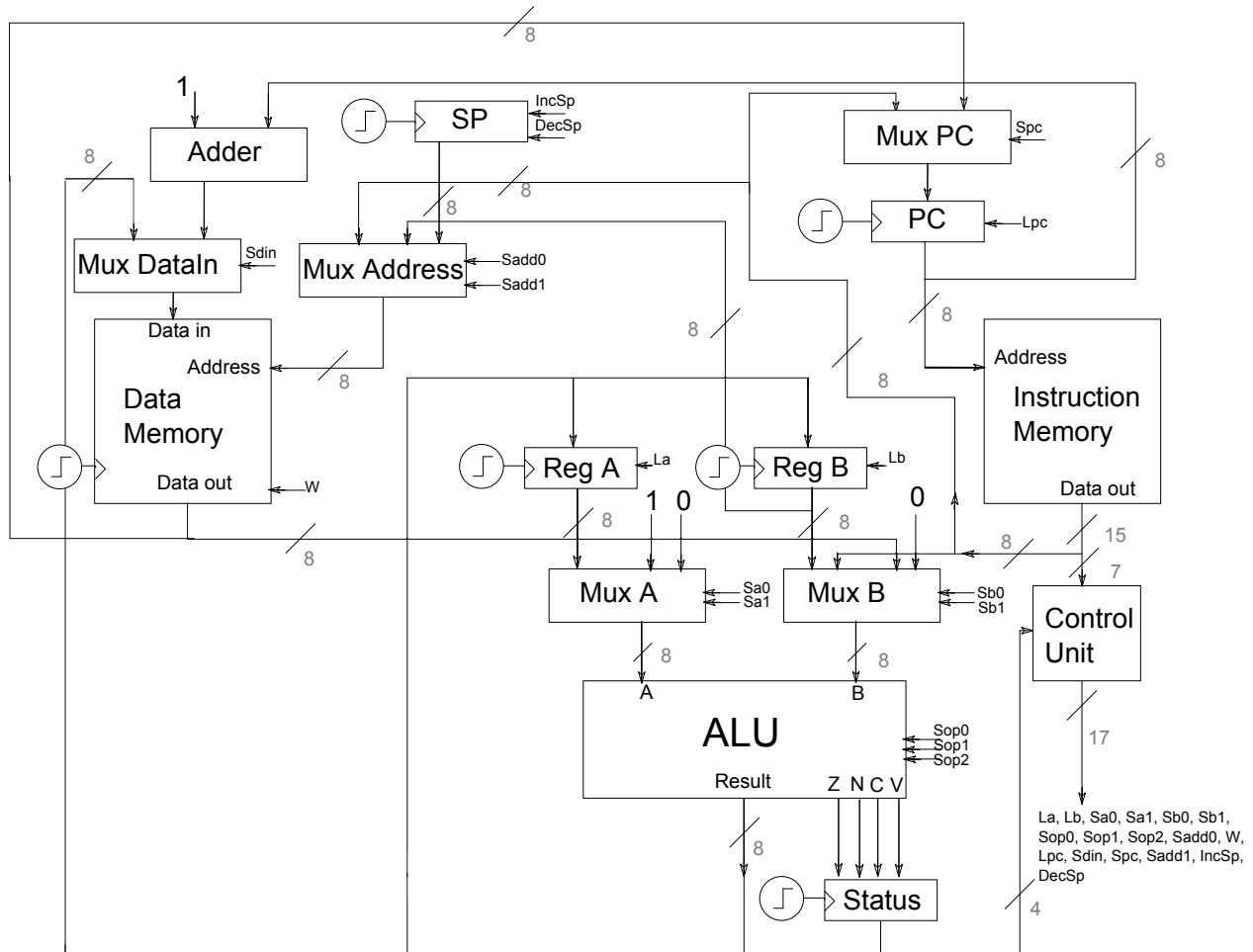




IIC2343 Arquitectura de Computadores  
**Arquitectura Computador Básico**

©Alejandro Echeverría, Hans-Albert Löbel

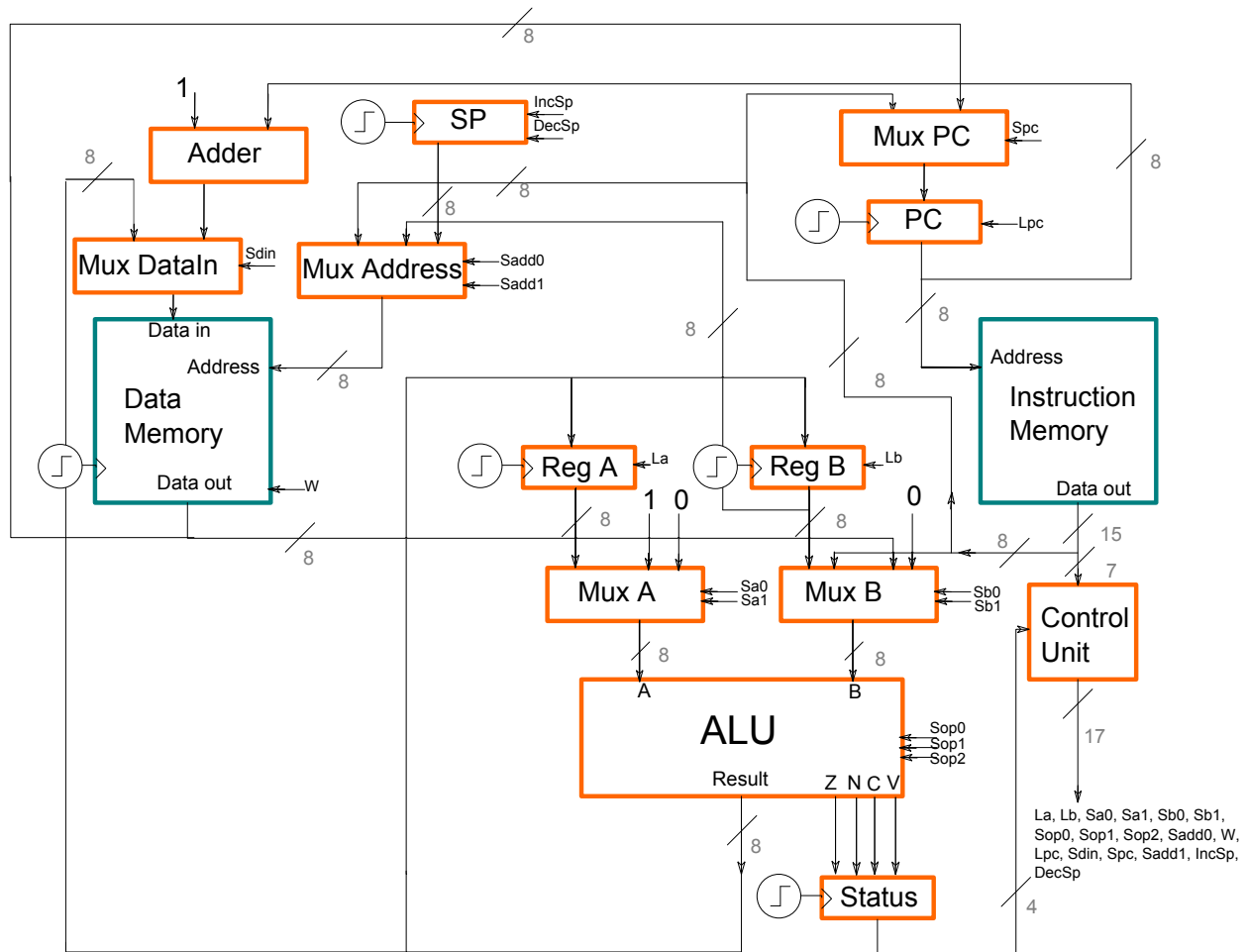
## 1. Microarquitectura Computador Básico



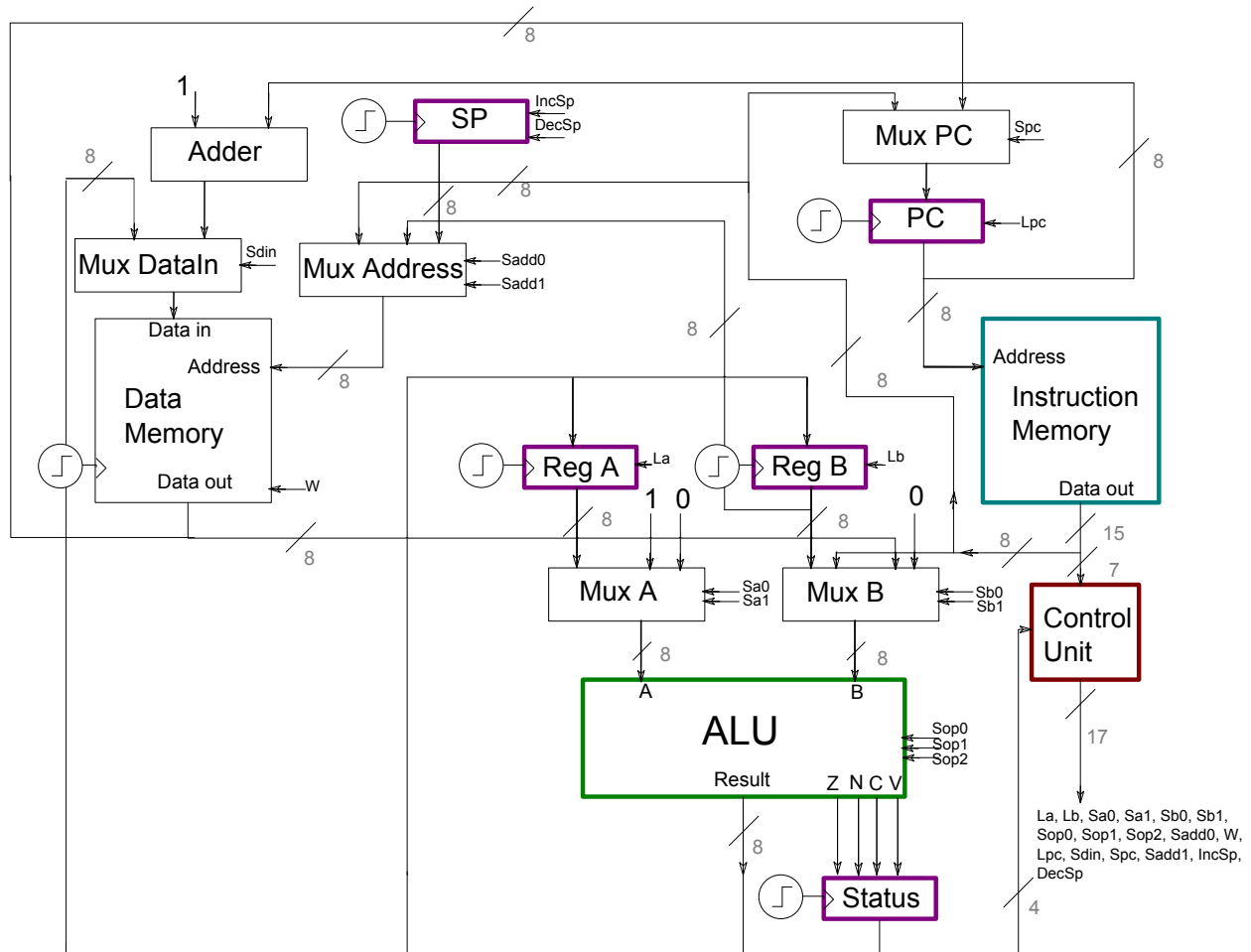
## 1.1. Partes del Computador Básico

### Procesador (CPU)

### Memorias



## Registros, Unidad de ejecución, Unidad de control



## 2. Set de instrucciones computador básico

### 2.1. Instrucciones de carga, aritméticas y lógicas

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
MOV	A,B	A=B		-
	B,A	B=A		-
	A,Lit	A=Lit		MOV A,15
	B,Lit	B=Lit		MOV B,15
ADD	A,B	A=A+B		-
	B,A	B=A+B		-
	A,Lit	A=A+Lit		ADD A,5
SUB	A,B	A=A-B		-
	B,A	B=A-B		-
	A,Lit	A=A-Lit		SUB A, 2
AND	A,B	A=A and B		-
	B,A	B=A and B		-
	A,Lit	A=A and Lit		AND A,15
OR	A,B	A=A or B		-
	B,A	B=A or B		-
	A,Lit	A=A or Lit		OR A,5
NOT	A,A	A=notA		-
	B,A	B=notA		-
	A,Lit	A=notLit		NOT A,2
XOR	A,A	A=A xor B		-
	B,A	B=A xor B		-
	A,Lit	A=A xor Lit		XOR A,15
SHL	A,A	A=shift left A		-
	B,A	B=shift left A		-
	A,Lit	A=shift left Lit		SHL A,5
SHR	A,A	A=shift right A		-
	B,A	B=shift right A		-
	A,Lit	A=shift right Lit		SHR A,2

## 2.2. Instrucciones de salto y comparación

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
CMP	A,B	A-B		CMP A,0
	A,Lit	A-Lit		JMP end
JMP	Dir	PC = Dir		JEQ label
JEQ	Dir	PC = Dir	Z=1	JNE label
JNE	Dir	PC = Dir	Z=0	JGT label
JGT	Dir	PC = Dir	N=0 y Z=0	JLT label
JLT	Dir	PC = Dir	N=1	JGE label
JGE	Dir	PC = Dir	N=0	JLE label
JLE	Dir	PC = Dir	Z=1 o N=1	JCR label
JCR	Dir	PC = Dir	C=1	JOV label
JOV	Dir	PC = Dir	V=1	

## 2.3. Instrucciones de memoria y direccionamiento

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
MOV	A,(Dir)	A=Mem[Dir]		MOV A,(var1)
	B,(Dir)	B=Mem[Dir]		MOV B,(var2)
	(Dir),A	Mem[Dir]=A		MOV (var1),A
	(Dir),B	Mem[Dir]=B		MOV (var2),B
	A,(B)	A=Mem[B]		-
	B,(B)	B=Mem[B]		-
	(B),A	Mem[B]=A		-
ADD	A,(Dir)	A=A+Mem[Dir]		ADD A,(var1)
	A,(B)	A=A+Mem[B]		-
	(Dir)	Mem[Dir]=A+B		ADD (var1)
SUB	A,(Dir)	A=A-Mem[Dir]		SUB A,var1
	A,(B)	A=A-Mem[B]		-
	(Dir)	Mem[Dir]=A-B		SUB (var1)
AND	A,(Dir)	A=A and Mem[Dir]		AND A,(var1)
	A,(B)	A=A and Mem[B]		-
	(Dir)	Mem[Dir]=A and B		-
OR	A,(Dir)	A=A or Mem[Dir]		OR A,(var1)
	A,(B)	A=A or Mem[B]		-
	(Dir)	Mem[Dir]=A or B		OR (var1)
NOT	(Dir)	Mem[Dir]=not A		NOT (var1)
XOR	A,(Dir)	A=A xor Mem[Dir]		XOR A,(var1)
	A,(B)	A=A xor Mem[B]		-
	(Dir)	Mem[Dir]=A xor B		XOR (var1)
SHL	(Dir)	Mem[Dir]=shift left A		SHL (var1)
SHR	(Dir)	Mem[Dir]=shift right A		SHR(var1)
INC	B	B=B+1		-

## 2.4. Instrucciones de subrutinas y stack

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
CALL	Dir	$\text{Mem}[\text{SP}] = \text{PC} + 1$ , $\text{SP}-$ , $\text{PC} = \text{Dir}$		CALL func
RET		$\text{SP}++$ $\text{PC} = \text{Mem}[\text{SP}]$		-
PUSH	A	$\text{Mem}[\text{SP}] = \text{A}$ , $\text{SP}-$		-
PUSH	B	$\text{Mem}[\text{SP}] = \text{B}$ , $\text{SP}-$		-
POP	A	$\text{SP}++$ $\text{A} = \text{Mem}[\text{SP}]$		-
POP	B	$\text{SP}++$ $\text{B} = \text{Mem}[\text{SP}]$		-

## 2.5. Set de instrucciones completo

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
MOV	A,B	A=B		-
	B,A	B=A		-
	A,Lit	A=Lit		MOV A,15
	B,Lit	B=Lit		MOV B,15
	A,(Dir)	A=Mem[Dir]		MOV A,(var1)
	B,(Dir)	B=Mem[Dir]		MOV B,(var2)
	(Dir),A	Mem[Dir]=A		MOV (var1),A
	(Dir),B	Mem[Dir]=B		MOV (var2),B
	A,(B)	A=Mem[B]		-
	B,(B)	B=Mem[B]		-
	(B),A	Mem[B]=A		-
ADD	A,B	A=A+B		-
	B,A	B=A+B		-
	A,Lit	A=A+Lit		ADD A,5
	A,(Dir)	A=A+Mem[Dir]		ADD A,(var1)
	A,(B)	A=A+Mem[B]		-
	(Dir)	Mem[Dir]=A+B		ADD (var1)
SUB	A,B	A=A-B		-
	B,A	B=A-B		-
	A,Lit	A=A-Lit		SUB A, 2
	A,(Dir)	A=A-Mem[Dir]		SUB A,(var1)
	A,(B)	A=A-Mem[B]		-
	(Dir)	Mem[Dir]=A-B		SUB (var1)
AND	A,B	A=A and B		-
	B,A	B=A and B		-
	A,Lit	A=A and Lit		AND A,15
	A,(Dir)	A=A and Mem[Dir]		AND A,(var1)
	A,(B)	A=A and Mem[B]		-
	(Dir)	Mem[Dir]=A and B		AND (var1)
OR	A,B	A=A or B		-
	B,A	B=A or B		-
	A,Lit	A=A or Lit		OR A,5
	A,(Dir)	A=A or Mem[Dir]		OR A,(var1)
	A,(B)	A=A or Mem[B]		-
	(Dir)	Mem[Dir]=A or B		OR (var1)
NOT	A,A	A=notA		-
	B,A	B=notA		-
	(Dir)	Mem[Dir]=not A		NOT (var1)

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
XOR	A,A B,A A,Lit A,(Dir) A,(B)	A=A xor B B=A xor B A=A xor Lit A=A xor Mem[Dir] A=A xor Mem[B]		- - XOR A,15 XOR A,(var1)
SHL	(Dir) A,A B,A	Mem[Dir]=A xor B A=shift left A B=shift left A		XOR (var1) - -
SHR	(Dir) A,A B,A	Mem[Dir]=shift left A A=shift right A B=shift right A		SHL (var1) - -
INC	(Dir) B	Mem[Dir]=shift right A B=B+1		SHR (var1) -
CMP	A,B A,Lit	A-B A-Lit		CMP A,0
JMP	Dir	PC = Dir		JMP end
JEQ	Dir	PC = Dir	Z=1	JEQ label
JNE	Dir	PC = Dir	Z=0	JNE label
JGT	Dir	PC = Dir	N=0 y Z=0	JGT label
JLT	Dir	PC = Dir	N=1	JLT label
JGE	Dir	PC = Dir	N=0	JGE label
JLE	Dir	PC = Dir	Z=1 o N=1	JLE label
JCR	Dir	PC = Dir	C=1	JCR label
JOV	Dir	PC = Dir	V=1	JOV label
CALL	Dir	Mem[SP] = PC + 1, SP-, PC = Dir		CALL func
RET		SP++ PC = Mem[SP]		- -
PUSH	A	Mem[SP] = A, SP-		-
PUSH	B	Mem[SP] = B, SP-		-
POP	A	SP++ A = Mem[SP]		- -
POP	B	SP++ B = Mem[SP]		- -





Instrucción	Operandos	Opcode	Condition	Lpc	La	Lb	Sa0,1	Sb0,1	Sop0,1,2	Sadd0,1	Sdin0	Spc0	W	IncSp	DecSp
XOR	A,B	0101000		0	1	0	A	B	XOR	-	-	-	0	0	0
	B,A	0101001		0	1	1	A	B	XOR	-	-	-	0	0	0
	A,Lit	0101010		0	1	0	A	LIT	XOR	-	-	-	0	0	0
	A,(Dir)	0101011		0	1	0	A	DOUT	XOR	LIT	-	-	0	0	0
	A,(B)	0101100		0	1	0	A	DOUT	XOR	B	-	-	0	0	0
SHL	(Dir)	0101101		0	0	0	A	B	XOR	LIT	ALU	-	1	0	0
	A,A	0101110		0	1	0	A	-	SHL	-	-	-	0	0	0
	B,A	0101111		0	0	1	A	-	SHL	-	-	-	0	0	0
	(Dir)	0110011		0	0	0	A	B	SHL	LIT	ALU	-	1	0	0
	A,A	0110100		0	1	0	A	-	SHR	-	-	-	0	0	0
SHR	B,A	0110101		0	0	1	A	-	SHR	-	-	-	0	0	0
	(Dir)	0111001		0	0	0	A	B	SHR	LIT	ALU	-	1	0	0
INC	B	0111010		0	0	1	ONE	B	ADD	-	-	-	0	0	0
CMP	A,B	0111011		0	0	0	A	B	SUB	-	-	-	0	0	0
	A,Lit	0111100		0	0	0	A	LIT	SUB	-	-	-	0	0	0
JMP	Dir	0111101		1	0	0	-	-	-	-	-	LIT	0	0	0
JEQ	Dir	0111110	Z=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JNE	Dir	0111111	Z=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JGT	Dir	1000000	N=0 y Z=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JLT	Dir	1000001	N=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JGE	Dir	1000010	N=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JLE	Dir	1000011	N=1 o Z=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JCR	Dir	1000100	C=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JOV	Dir	1000101	V=1	1	0	0	-	-	-	-	-	LIT	0	0	0
CALL	Dir	1000101		1	0	0	-	-	-	SP	PC	LIT	1	0	1
RET		1000110		0	0	0	-	-	-	-	-	-	0	1	0
		1000111		1	0	0	-	-	-	SP	-	DOUT	0	0	0
PUSH	A	1001000		0	0	0	A	ZERO	ADD	SP	ALU	-	1	0	1
PUSH	B	1001001		0	0	0	ZERO	B	ADD	SP	ALU	-	1	0	1
POP	A	1001010		0	0	0	-	-	-	-	-	-	0	1	0
		1001011		0	1	0	ZERO	DOUT	ADD	SP	ALU	-	0	0	0
POP	B	1001100		0	0	0	-	-	-	-	-	-	0	1	0
		1001101		0	0	1	ZERO	DOUT	ADD	SP	ALU	-	0	0	0