

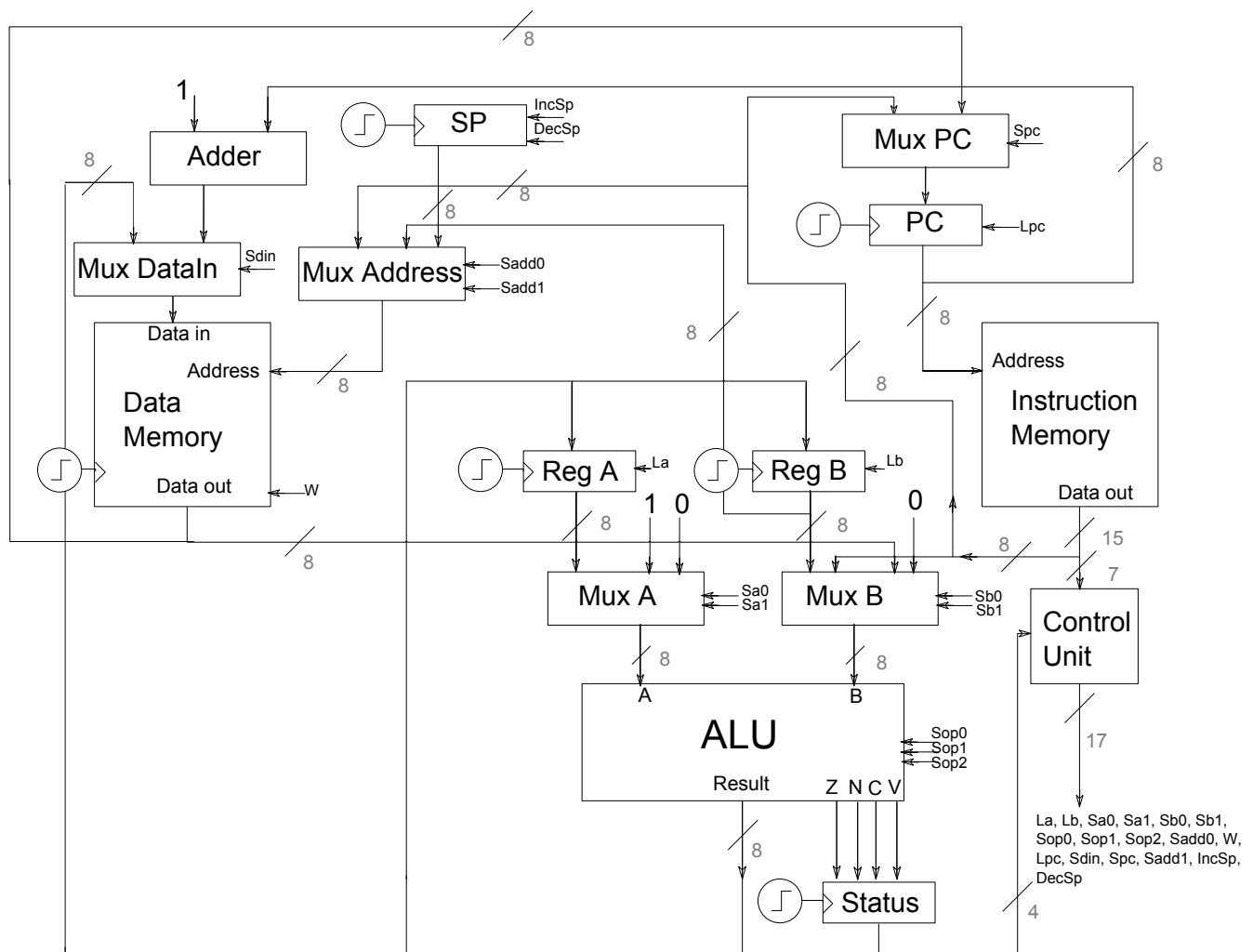


IIC2343 - Arquitectura de Computadores

Arquitectura Computador Básico

©Alejandro Echeverría, Hans Löbel

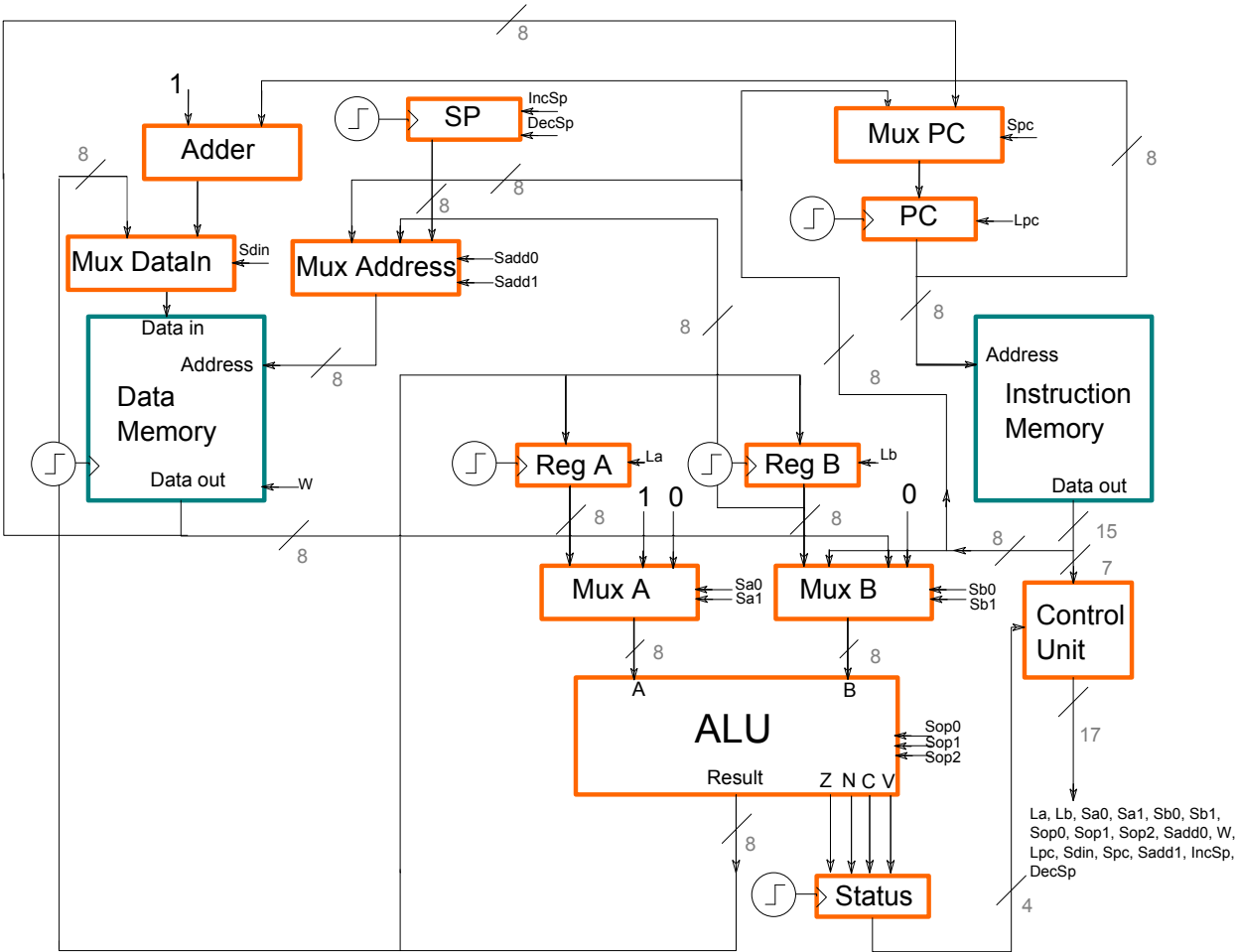
1. Microarquitectura Computador Básico



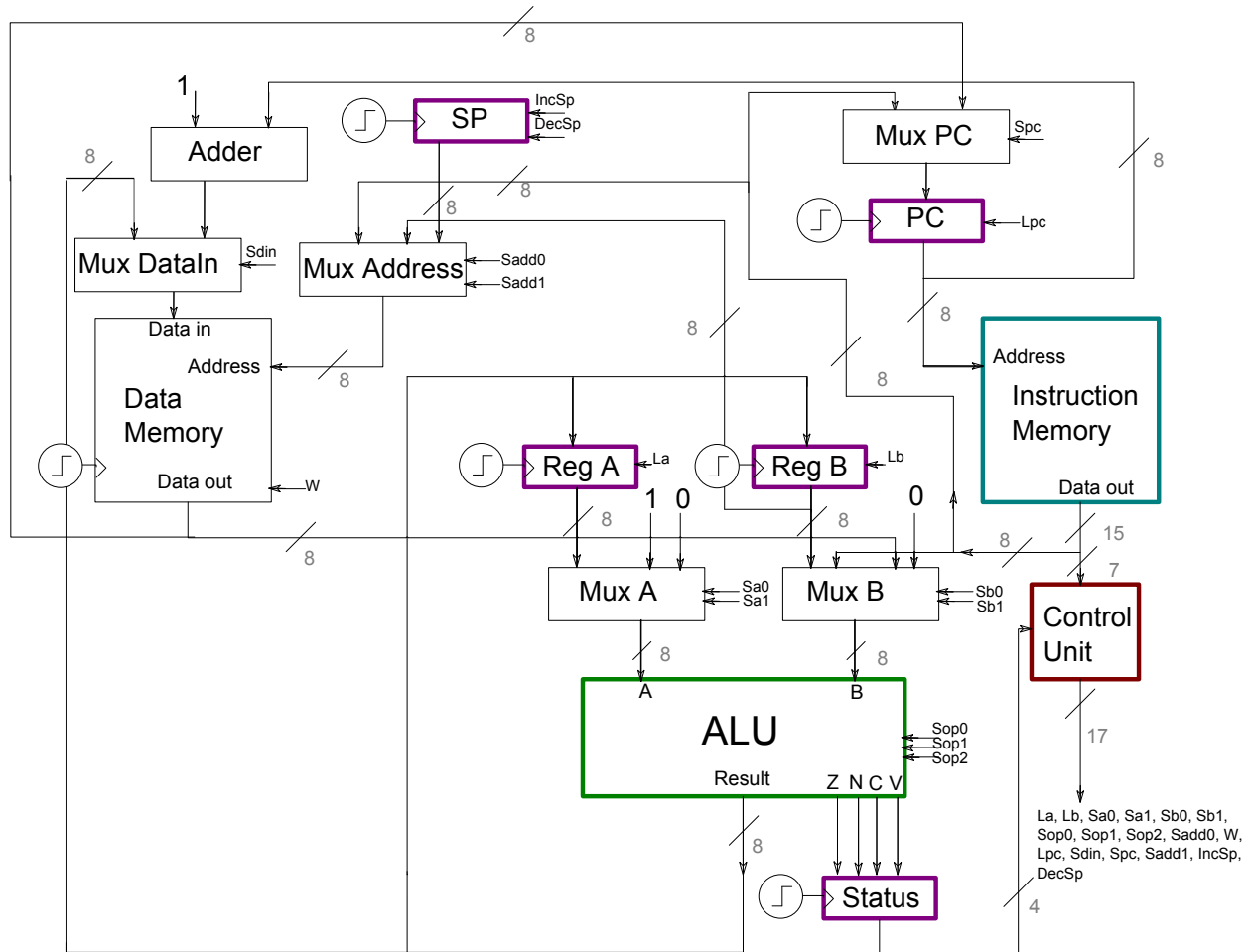
1.1. Partes del Computador Básico

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=not A		-
	B,A	B=not A		-
XOR	A,B	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		-
SHR	A,A	A=shift right A		-
	B,A	B=shift right A		-

### 2.2. Instrucciones de salto y comparación

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
CMP	A,B	A-B		
	A,Lit	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

### 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	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++		-
POP		B = Mem[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=not A		-
	B,A	B=not A		-
	(Dir)	Mem[Dir]=not A		NOT (var1)

Instrucción	Operandos	Operación	Condiciones	Ejemplo de uso
XOR	A,B	A=A xor B		-
	B,A	B=A xor B		-
	A,Lit	A=A xor Lit		XOR A,15
	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	A,A	A=shift left A		-
	B,A	B=shift left A		-
	(Dir)	Mem[Dir]=shift left A		SHL (var1)
SHR	A,A	A=shift right A		-
	B,A	B=shift right A		-
	(Dir)	Mem[Dir]=shift right A		SHR (var1)
INC	B	B=B+1		-
CMP	A,B	A-B		
	A,Lit	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]		-

### 3. Señales de control

Instrucción	Operandos	Opcode	Condition	Lpc	La	Lb	Sa0,1	Sb0,1	Sop0,1,2	Sadd0,1	Sdin0	Spc0	W	IncSp	DecSp
MOV	A,B	0000000		0	1	0	ZERO	B	ADD	-	-	-	0	0	0
	B,A	0000001		0	0	1	A	ZERO	ADD	-	-	-	0	0	0
	A,Lit	0000010		0	1	0	ZERO	LIT	ADD	-	-	-	0	0	0
	B,Lit	0000011		0	0	1	ZERO	LIT	ADD	-	-	-	0	0	0
	A,(Dir)	0000100		0	1	0	ZERO	DOUT	ADD	LIT	-	-	0	0	0
	B,(Dir)	0000101		0	0	1	ZERO	DOUT	ADD	LIT	-	-	0	0	0
	(Dir),A	0000110		0	0	0	A	ZERO	ADD	LIT	ALU	-	1	0	0
	(Dir),B	0000111		0	0	0	ZERO	B	ADD	LIT	ALU	-	1	0	0
	A,(B)	0001000		0	1	0	ZERO	DOUT	ADD	B	-	-	0	0	0
	B,(B)	0001001		0	0	1	ZERO	DOUT	ADD	B	-	-	0	0	0
	(B),A	0001010		0	0	0	A	ZERO	ADD	B	ALU	-	1	0	0
ADD	A,B	0001011		0	1	0	A	B	ADD	-	-	-	0	0	0
	B,A	0001100		0	0	1	A	B	ADD	-	-	-	0	0	0
	A,Lit	0001101		0	1	0	A	LIT	ADD	-	-	-	0	0	0
	A,(Dir)	0001110		0	1	0	A	DOUT	ADD	LIT	-	-	0	0	0
	A,(B)	0001111		0	1	0	A	DOUT	ADD	B	-	-	0	0	0
SUB	(Dir)	0010000		0	0	0	A	B	ADD	LIT	ALU	-	1	0	0
	A,B	0010001		0	1	0	A	B	SUB	-	-	-	0	0	0
	B,A	0010010		0	0	1	A	B	SUB	-	-	-	0	0	0
	A,Lit	0010011		0	1	0	A	LIT	SUB	-	-	-	0	0	0
	A,(Dir)	0010100		0	1	0	A	DOUT	SUB	LIT	-	-	0	0	0
AND	A,(B)	0010101		0	1	0	A	DOUT	SUB	B	-	-	0	0	0
	(Dir)	0010110		0	0	0	A	B	SUB	LIT	ALU	-	1	0	0
	A,B	0010111		0	1	0	A	B	AND	-	-	-	0	0	0
	B,A	0011000		0	0	1	A	B	AND	-	-	-	0	0	0
	A,Lit	0011001		0	1	0	A	LIT	AND	-	-	-	0	0	0
OR	A,(Dir)	0011010		0	1	0	A	DOUT	AND	LIT	-	-	0	0	0
	A,(B)	0011011		0	1	0	A	DOUT	AND	B	-	-	0	0	0
	(Dir)	0011100		0	0	0	A	B	AND	LIT	ALU	-	1	0	0
	A,B	0011101		0	1	0	A	B	OR	-	-	-	0	0	0
	B,A	0011110		0	0	1	A	B	OR	-	-	-	0	0	0
NOT	A,Lit	0011111		0	1	0	A	LIT	OR	-	-	-	0	0	0
	A,(Dir)	0100000		0	1	0	A	DOUT	OR	LIT	-	-	0	0	0
	A,(B)	0100001		0	1	0	A	DOUT	OR	B	-	-	0	0	0
	(Dir)	0100010		0	0	0	A	B	OR	LIT	ALU	-	1	0	0
	A,A	0100011		0	1	0	A	-	NOT	-	-	-	0	0	0
	B,A	0100100		0	0	1	A	-	NOT	-	-	-	0	0	0
	(Dir)	0100101		0	0	0	A	-	NOT	LIT	ALU	-	1	0	0



Instrucción	Operandos	Opcode	Condition	Lpc	La	Lb	Sa0,1	Sb0,1	Sop0,1,2	Sadd0,1	Sdin0	Spc0	W	IncSp	DecSp
XOR	A,B	0100110		0	1	0	A	B	XOR	-	-	-	0	0	0
	B,A	0100111		0	0	1	A	B	XOR	-	-	-	0	0	0
	A,Lit	0101000		0	1	0	A	LIT	XOR	-	-	-	0	0	0
	A,(Dir)	0101001		0	1	0	A	DOUT	XOR	LIT	-	-	0	0	0
	A,(B)	0101010		0	1	0	A	DOUT	XOR	B	-	-	0	0	0
	(Dir)	0101011		0	0	0	A	B	XOR	LIT	ALU	-	1	0	0
SHL	A,A	0101100		0	1	0	A	-	SHL	-	-	-	0	0	0
	B,A	0101101		0	0	1	A	-	SHL	-	-	-	0	0	0
	(Dir)	0101110		0	0	0	A	-	SHL	LIT	ALU	-	1	0	0
SHR	A,A	0101111		0	1	0	A	-	SHR	-	-	-	0	0	0
	B,A	0110000		0	0	1	A	-	SHR	-	-	-	0	0	0
	(Dir)	0110001		0	0	0	A	-	SHR	LIT	ALU	-	1	0	0
INC	B	0110010		0	0	1	ONE	B	ADD	-	-	-	0	0	0
CMP	A,B	0110011		0	0	0	A	B	SUB	-	-	-	0	0	0
	A,Lit	0110100		0	0	0	A	LIT	SUB	-	-	-	0	0	0
JMP	Dir	0110101		1	0	0	-	-	-	-	-	LIT	0	0	0
JEQ	Dir	0110110	Z=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JNE	Dir	0110111	Z=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JGT	Dir	0111000	N=0 y Z=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JLT	Dir	0111001	N=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JGE	Dir	0111010	N=0	1	0	0	-	-	-	-	-	LIT	0	0	0
JLE	Dir	0111011	N=1 o Z=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JCR	Dir	0111100	C=1	1	0	0	-	-	-	-	-	LIT	0	0	0
JOV	Dir	0111101	V=1	1	0	0	-	-	-	-	-	LIT	0	0	0
CALL	Dir	0111110		1	0	0	-	-	-	SP	PC	LIT	1	0	1
RET		0111111		0	0	0	-	-	-	-	-	-	0	1	0
		1000000		1	0	0	-	-	-	SP	-	DOUT	0	0	0
PUSH	A	1000001		0	0	0	A	ZERO	ADD	SP	ALU	-	1	0	1
PUSH	B	1000010		0	0	0	ZERO	B	ADD	SP	ALU	-	1	0	1
POP	A	0111111		0	0	0	-	-	-	-	-	-	0	1	0
POP		1000011		0	1	0	ZERO	DOUT	ADD	SP	ALU	-	0	0	0
		0111111		0	0	0	-	-	-	-	-	-	0	1	0
	B	1000100		0	0	1	ZERO	DOUT	ADD	SP	ALU	-	0	0	0