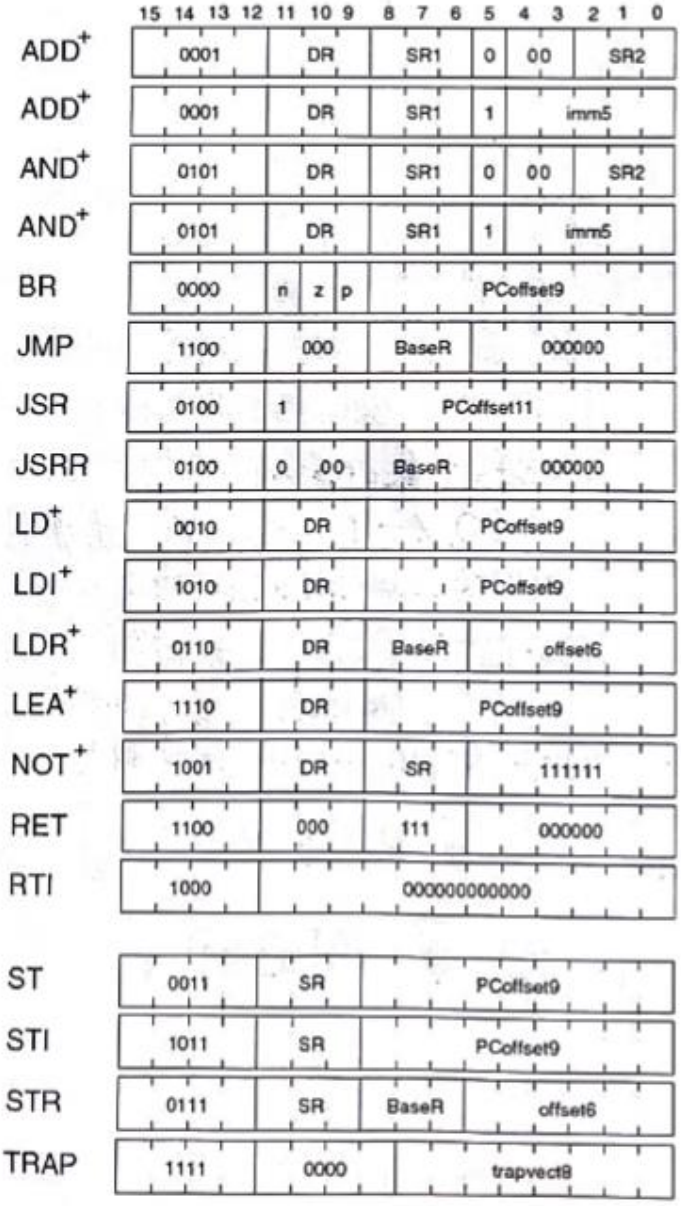
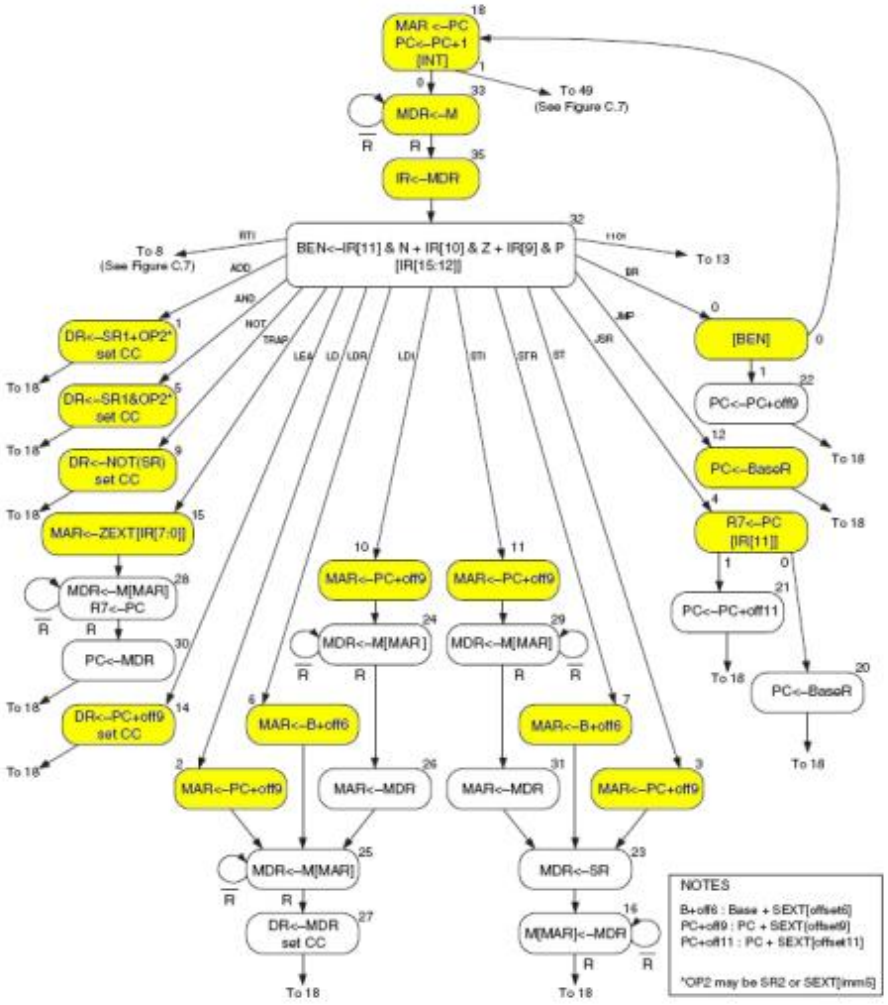
LC-3 Instruction Set

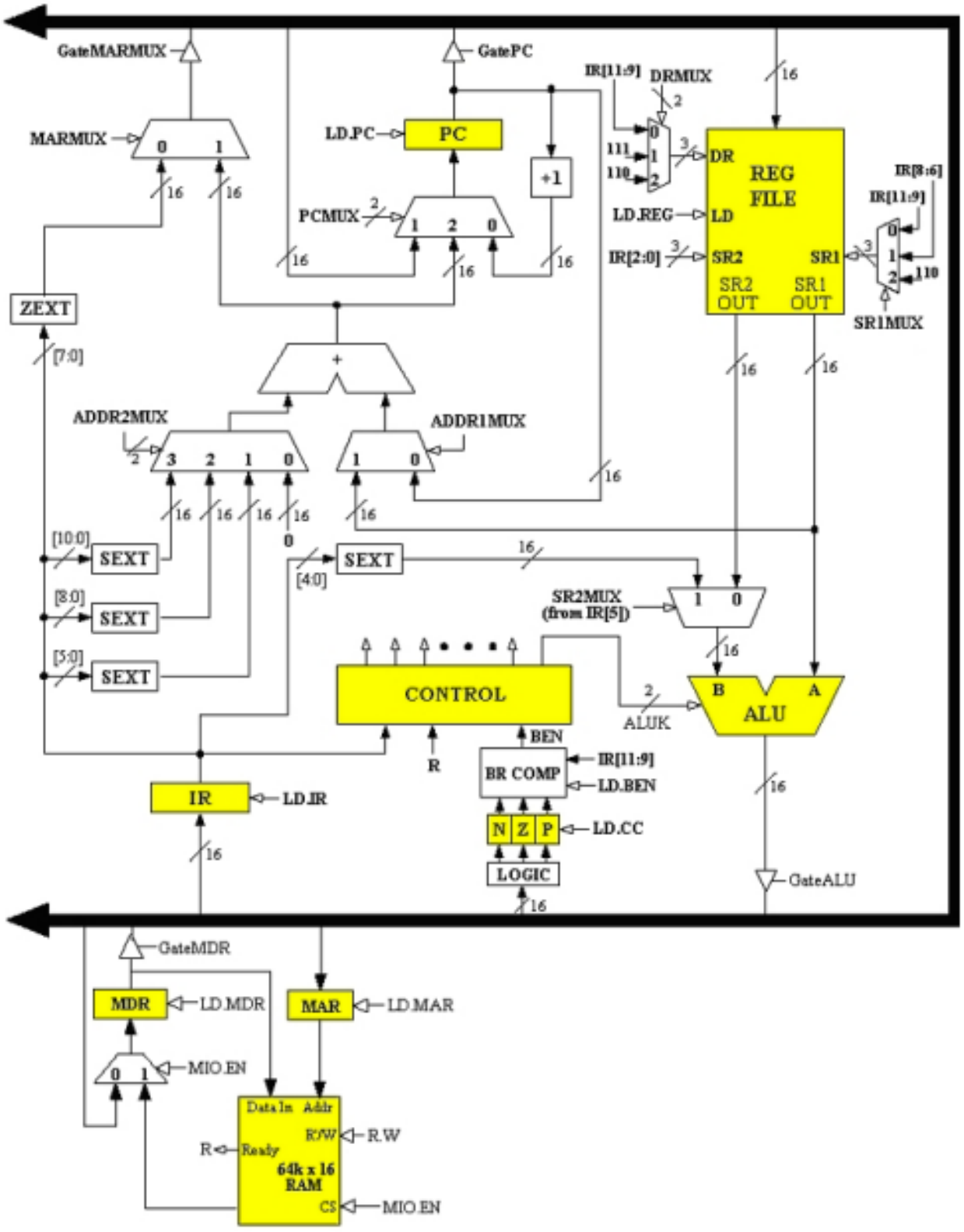


superscript "+" denotes instructions that update the condition bits NZP

LC-3 FSM


18
MAR <–PC
PC<–PC+1
[INT]
1
To 49
(See Figure C.7)
0
33
MDR<–M
R
R
35
IR<–MDR
32
BEN<–IR[11] & N + IR[10] & Z + IR[9] & P
[IR[15:12]]
BR
To 13
To 8
(See Figure C.7)
ADD
AND
NOT
TRAP
LEA
LD/LDR
LDI
STI
STR
ST
JMP
JSR
0
[BEN]
0
1
22
PC<–PC+off9
To 18
12
PC<–BaseR
To 18
4
R7<–PC
[IR[11]]
To 18
1
0
21
PC<–PC+off11
To 18
20
PC<–BaseR
To 18
1
DR<–SR1+OP2*
set CC
To 18
5
DR<–SR1&OP2*
set CC
To 18
9
DR<–NOT(SR)
set CC
To 18
15
MAR<–ZEXT[IR[7:0]]
28
MDR<–M[MAR]
R7<–PC
R
R
30
PC<–MDR
To 18
14
DR<–PC+off9
set CC
To 18
10
MAR<–PC+off9
24
MDR<–M[MAR]
R
R
6
MAR<–B+off6
2
MAR<–PC+off9
26
MAR<–MDR
25
MDR<–M[MAR]
R
R
27
DR<–MDR
set CC
To 18
11
MAR<–PC+off9
29
MDR<–M[MAR]
R
R
7
MAR<–B+off6
31
MAR<–MDR
3
MAR<–PC+off9
23
MDR<–SR
16
M[MAR]<–MDR
R
R
To 18
NOTES
B+off6 : Base + SEXT[offset6]
PC+off9 : PC + SEXT[offset9]
PC+off11 : PC + SEXT[offset11]
*OP2 may be SR2 or SEXT[imm5]

LC-3 Datapath



LC-3 Datapath Control Signals

Signal	Description
LD.MAR	= 1, MAR is loaded
LD.MDR	= 1, MDR is loaded
LD.IR	= 1, IR is loaded
LD.PC	= 1, PC is loaded
LD.REG	= 1, register file is loaded
LD.BEN	= 1, updates Branch Enable (BEN) bit
MARMUX	= 0, chooses ZEXT IR[7:0] = 1, chooses address adder output
ADDR1MUX	= 0, chooses PC = 1, chooses reg file SR1OUT
ADDR2MUX	= 00, chooses "0...00" = 01, chooses SEXT IR[5:0] = 10, chooses SEXT IR[8:0] = 11, chooses SEXT IR[10:0]
PCMUX	= 00, chooses PC + 1 = 01, chooses system bus = 10. chooses address adder output
SR1MUX	= 00, chooses IR[11:9] = 01, chooses IR[8:6] = 10, chooses "110"

Signal	Description
LD.CC	= 1, updates status bits from system bus
GateMARMUX	= 1, MARMUX output is put onto system bus
GateMDR	= 1, MDR contents are put onto system bus
GateALU	= 1, ALU output is put onto system bus
GatePC	= 1, PC contents are put onto system bus
MIO.EN	= 1, Enables memory, chooses memory output for MDR input = 0, Disables memory, chooses system bus for MDR input
R.W	= 1, M[MAR]<-MDR when MIO.EN = 1 = 0, MDR<-M[MAR] when MIO.EN = 1
ALUK	= 00, ADD = 01, AND = 10, NOT A = 11, PASS A
DRMUX	= 00, chooses IR[11:9] = 01, chooses "111" = 10, chooses "110"