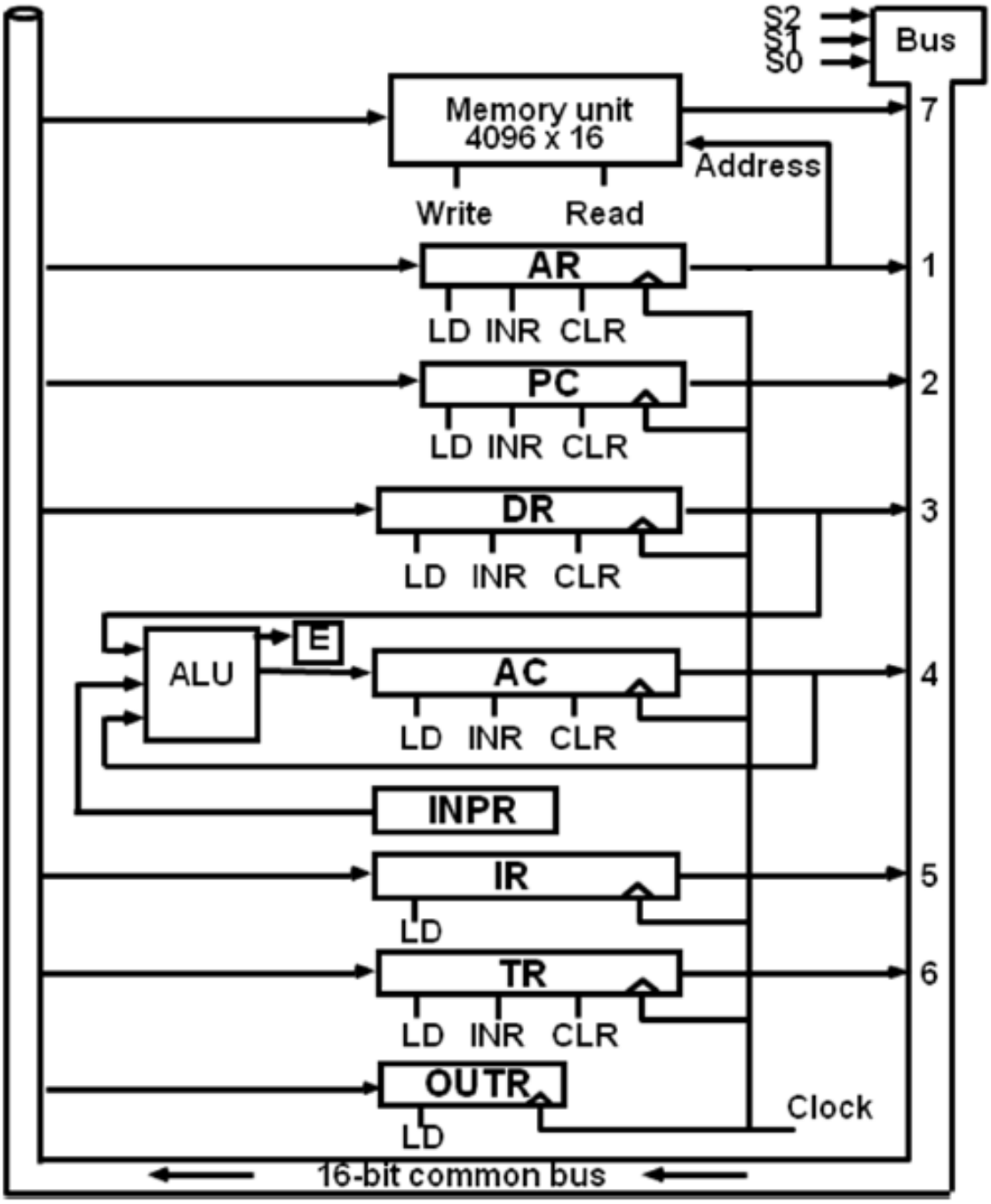
Appendix A: Common Bus of the Basic Computer



Appendix B: Instruction set of the Basic Computer

Symbol	Hex Code	Brief Description
AND	0 or 8	AND M to AC
ADD	1 or 9	Add M to AC, carry to E
LDA	2 or A	Load AC from M
STA	3 or B	Store AC in M
BUN	4 or C	Branch unconditionally to m
BSA	5 or D	Save return address in m and branch to m+1
ISZ	6 or E	Increment M and skip if zero
CLA	7800	Clear AC
CLE	7400	Clear E
CMA	7200	Complement AC
CME	7100	Complement E
CIR	7080	Circulate right E and AC
CIL	7040	Circulate left E and AC
INC	7020	Increment AC, carry to E
SPA	7010	Skip if AC is positive
SNA	7008	Skip if AC is negative
SZA	7004	Skip if AC is zero
SZE	7002	Skip if E is zero
HLT	7001	Halt computer
INP	F800	Input information and clear flag
OUT	F400	Output information and clear flag
SKI	F200	Skip if input flag is on
SKO	F100	Skip if output flag is on
ION	F080	Turn interrupt on
IOF	F040	Turn interrupt off

Appendix C: Complete BC Description using Micro-operations

Fetch	$R'T_0$: $R'T_1$:	AR ← PC IR ← M[AR], PC ← PC + 1
Decode	$R'T_2$:	D0, ..., D7 ← Decode IR(12 ~ 14), AR ← IR(0 ~ 11), I ← IR(15)
Indirect	$D_7'IT_3$:	AR ← M[AR]
Interrupt	$T_0'T_1'T_2'$(IEN)(FGI + FGO): RT_0: RT_1: RT_2:	R ← 1 AR ← 0, TR ← PC M[AR] ← TR, PC ← 0 PC ← PC + 1, IEN ← 0, R ← 0, SC ← 0
Memory-Reference		
AND	D_0T_4: D_0T_5:	DR ← M[AR] AC ← AC ∧ DR, SC ← 0
ADD	D_1T_4: D_1T_5:	DR ← M[AR] AC ← AC + DR, E ← C_{out}, SC ← 0
LDA	D_2T_4: D_2T_5:	DR ← M[AR] AC ← DR, SC ← 0
STA	D_3T_4:	M[AR] ← AC, SC ← 0
BUN	D_4T_4:	PC ← AR, SC ← 0
BSA	D_5T_4: D_5T_5:	M[AR] ← PC, AR ← AR + 1 PC ← AR, SC ← 0
ISZ	D_6T_4: D_6T_5: D_6T_6:	DR ← M[AR] DR ← DR + 1 M[AR] ← DR, if(DR=0) then (PC ← PC + 1), SC ← 0
Register-Reference ($D_7I'T_3$ = r, IR(i) = B_i)		
	r:	SC ← 0
CLA	rB_{11}:	AC ← 0
CLE	rB_{10}:	E ← 0
CMA	rB_9:	AC ← AC'
CME	rB_8:	E ← E'
CIR	rB_7:	AC ← shr AC, AC(15) ← E, E ← AC(0)
CIL	rB_6:	AC ← shl AC, AC(0) ← E, E ← AC(15)
INC	rB_5:	AC ← AC + 1, E ← C_{out}
SPA	rB_4:	If(AC(15) =0) then (PC ← PC + 1)
SNA	rB_3:	If(AC(15) =1) then (PC ← PC + 1)
SZA	rB_2:	If(AC = 0) then (PC ← PC + 1)
SZE	rB_1:	If(E=0) then (PC ← PC + 1)
HLT	rB_0:	S ← 0
Input-Output (D_7IT_3 = p, IR(i) = B_i)		
	p:	SC ← 0
INP	pB_{11}:	AC(0-7) ← INPR, FGI ← 0
OUT	pB_{10}:	OUTR ← AC(0-7), FGO ← 0
SKI	pB_9:	If(FGI=1) then (PC ← PC + 1)
SKO	pB_8:	If(FGO=1) then (PC ← PC + 1)
ION	pB_7:	IEN ← 1
IOF	pB_6:	IEN ← 0