OPCODE (HEX)	SYMBOL	DESCRIPTION	
0x00	BRA	PC ← PC + VALUE	Y
0x01	BNE	IF Z=0 THEN PC ← PC + VALUE	Y
0x02	BEQ	IF Z=1 THEN PC ← PC + VALUE	Y
0x03	POP	SP ← SP + 1, Rx ← M[SP]	Y
0x04	PSH	M[SP] ← Rx, SP ← SP – 1	Y
0x05	INC	DSTREG ← SREG1 + 1	Y
0x06	DEC	DSTREG ← SREG1 – 1	Y
0x07	LSL	DSTREG ← LSL SREG1	Y
0x08	LSR	DSTREG ← LSR SREG1	Y
0x09	ASR	DSTREG ← ASR SREG1	Y
0x0A	CSL	DSTREG ← CSL SREG1	Y
0x0B	CSR	DSTREG ← CSR SREG1	Y
0x0C	AND	DSTREG ← SREG1 AND SREG2	Y
0x0D	ORR	DSTREG ← SREG1 OR SREG2	Y
0x0E	NOT	DSTREG ← NOT SREG1	Y
0x0F	XOR	DSTREG ← SREG1 XOR SREG2	Y
0x10	NAND	DSTREG ← SREG1 NAND SREG2	Y
0x11	MOVH	DSTREG[15:8] ← IMMEDIATE **(8-bit)**	Y
0x12	LDR (16-bit)	Rx ← M[AR] (**AR is 16-bit register**)	Y
0x13	STR (16-bit)	M[AR] ← Rx (**AR is 16-bit register**)	Y
0x14	MOVL	DSTREG[7:0] ← IMMEDIATE **(8-bit)**	Y
0x15	ADD	DSTREG ← SREG1 + SREG2	Y
0x16	ADC	DSTREG ← SREG1 + SREG2 + CARRY	Y
0x17	SUB	DSTREG ← SREG1 - SREG2	Y
0x18	MOVS	DSTREG ← SREG1, Flags will change	Y
0x19	ADDS	DSTREG ← SREG1 + SREG2, Flags will change	Y
0x1A	SUBS	DSTREG ← SREG1 - SREG2, Flags will change	Y
0x1B	ANDS	DSTREG ← SREG1 AND SREG2, Flags will change	Y
0x1C	ORRS	DSTREG ← SREG1 OR SREG2, Flags will change	Y
0x1D	XORS	DSTREG ← SREG1 XOR SREG2, Flags will change	Y
0x1E	BX	M[SP] ← PC, PC ← Rx	Y
0x1F	BL	PC ← M[SP]	Y
0x20	LDRIM	Rx ← VALUE (VALUE defined in ADDRESS bits)	Y
0x21	STRIM	M[AR+OFFSET] ← Rx (**AR is 16-bit register**) (OFFSET defined in ADDRESS bits)	Y

OPCODE (6-bit)	RSEL (2-bit)	ADDRESS (8-bit)

Figure 1: Instructions with an address reference.

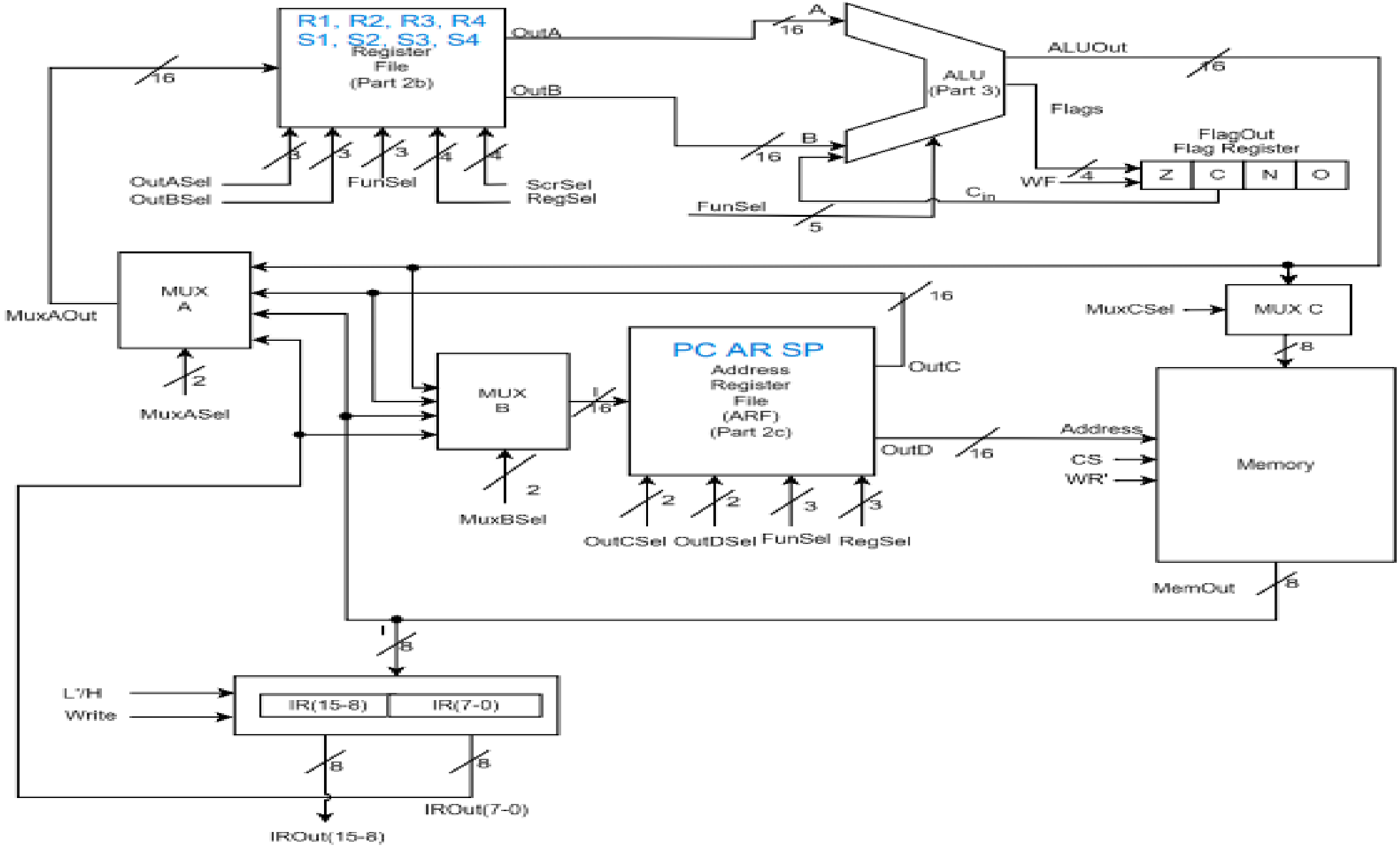
OPCODE (6-bit)	S (1-bit)	DSTREG(3-bit)	SREG1 (3-bit)	SREG2 (3-bit)

Figure 2: Instructions without an address reference.

RSEL	REGISTER
00	R1
01	R2
10	R3
11	R4

DSTREG/SREG1/SREG2	REGISTER
000	PC
001	PC
010	SP
011	AR
100	R1
101	R2
110	R3
111	R4

OFFSET in memory or IMMEDIATE ??


R1, R2, R3, R4
S1, S2, S3, S4
Register
File
(Part 2b)
16
OutA
OutB
3
3
3
4
4
OutASel
OutBSel
FunSel
ScrSel
RegSel
A
16
B
16
ALU
(Part 3)
FunSel
5
Cin
ALUOut
16
Flags
FlagOut
Flag Register
WF
4
Z
C
N
O
MUX
A
MuxAOut
2
MuxASel
MUX
B
1
16
2
MuxBSel
PC AR SP
Address
Register
File
(ARF)
(Part 2c)
16
OutC
OutD
16
2
2
3
3
OutCSel
OutDSel
FunSel
RegSel
MuxCSel
MUX C
8
Address
CS
WR'
Memory
MemOut
8
1
8
L'/H
Write
IR(15-8)
IR(7-0)
8
8
IROut(7-0)
IROut(15-8)

FUNSEL

E	FunSel	Q+
0	ϕ	Q (Retain value)
1	000	Q-1 (Decrement)
1	001	Q+1 (Increment)
1	010	I (Load)
1	011	0 (Clear)
1	100	Q (15-8) ← Clear, Q (7-0) ← I (7-0) (Write Low)
1	101	Q (7-0) ← I (7-0) (Only Write Low)
1	110	Q (15-8) ← I (7-0) (Only Write High)
1	111	Q (15-8) ← Sign Extend (I (7)) Q (7-0) ← I (7-0) (Write Low)

Instruction Register (IR)

L'H	Write	IR+
ϕ	0	IR (retain value)
0	1	IR (7-0) ← I (Load LSB)
1	1	IR (15-8) ← I (Load MSB)

Address Register File ARF

RegSel	Enable Address Registers
000	All address registers are enabled. (Function selected by FunSel will be applied to PC, AR, and SP.)
001	PC and AR are enabled. (Function selected by FunSel will be applied to PC and AR.)
010	PC and SP are enabled. (Function selected by FunSel will be applied to PC and SP.)
011	PC is enabled. (Function selected by FunSel will be applied to PC.)
100	AR and SP are enabled. (Function selected by FunSel will be applied to AR and SP.)
101	AR is enabled. (Function selected by FunSel will be applied to AR.)
110	SP is enabled. (Function selected by FunSel will be applied to SP.)
111	NO address register is enabled. (All registers retain their values.)

OutCSel	OutC
00	PC
01	PC
10	AR
11	SP

OutDSel	OutD
00	PC
01	PC
10	AR
11	SP

Register File (RF)

OutASel	OutA
000	R1
001	R2
010	R3
011	R4
100	S1
101	S2
110	S3
111	S4

OutBSel	OutB
000	R1
001	R2
010	R3
011	R4
100	S1
101	S2
110	S3
111	S4

RegSel	Enable General Purpose Registers	RegSel	Enable General Purpose Registers
0000	All general purpose registers are enabled. (Function selected by FunSel will be applied to R1, R2, R3 and R4.)	1000	R2, R3, and R4 are enabled. (Function selected by FunSel will be applied to R2, R3, and R4.)
0001	R1, R2 and R3 are enabled. (Function selected by FunSel will be applied to R1, R2, and R3.)	1001	R2 and R3 are enabled. (Function selected by FunSel will be applied to R2 and R3.)
0010	R1, R2, and R4 are enabled. (Function selected by FunSel will be applied to R1, R2, and R4.)	1010	R2 and R4 are enabled. (Function selected by FunSel will be applied to R2 and R4.)
0011	R1 and R2 are enabled. (Function selected by FunSel will be applied to R1 and R2.)	1011	Only R2 is enabled. (Function selected by FunSel will be applied to R2.)
0100	R1, R3, and R4 are enabled. (Function selected by FunSel will be applied to R1, R3, and R4.)	1100	R3 and R4 are enabled. (Function selected by FunSel will be applied to R3 and R4.)
0101	R1 and R3 are enabled. (Function selected by FunSel will be applied to R1 and R3.)	1101	Only R3 is enabled. (Function selected by FunSel will be applied to R3.)
0110	R1 and R4 are enabled. (Function selected by FunSel will be applied to R1 and R4.)	1110	Only R4 is enabled. (Function selected by FunSel will be applied to R4.)
0111	Only R1 is enabled. (Function selected by FunSel will be applied to R1.)	1111	NO general purpose register is enabled. (All registers retain their values.)

Table 4: ScrSel Control Input

ScrSel	Enable General Purpose Registers	ScrSel	Enable General Purpose Registers
0000	All general purpose registers are enabled. (Function selected by FunSel will be applied to S1, S2, S3, and S4.)	1000	S2, S3, and S4 are enabled. (Function selected by FunSel will be applied to S2, S3, and S4.)
0001	S1, S2, and S3 are enabled. (Function selected by FunSel will be applied to S1, S2, and S3.)	1001	S2 and S3 are enabled. (Function selected by FunSel will be applied to S2 and S3.)
0010	S1, S2, and S4 are enabled. (Function selected by FunSel will be applied to S1, S2, and S4.)	1010	S2 and S4 are enabled. (Function selected by FunSel will be applied to S2 and S4.)
0011	S1 and S2 are enabled. (Function selected by FunSel will be applied to S1 and S2.)	1011	Only S2 is enabled. (Function selected by FunSel will be applied to S2.)
0100	S1, S3, and S4 are enabled. (Function selected by FunSel will be applied to S1, S3, and S4.)	1100	S3 and S4 are enabled. (Function selected by FunSel will be applied to S3 and S4.)
0101	S1 and S3 are enabled. (Function selected by FunSel will be applied to S1 and S3.)	1101	Only S3 is enabled. (Function selected by FunSel will be applied to S3.)
0110	S1 and S4 are enabled. (Function selected by FunSel will be applied to S1 and S4.)	1110	Only S4 is enabled. (Function selected by FunSel will be applied to S4.)
0111	Only S1 is enabled. (Function selected by FunSel will be applied to S1.)	1111	NO general purpose register is enabled. (All registers retain their values.)

ALU

FunSel	ALUOut	Z	C	N	O
00000	A (8-bit)	+	-	+	-
00001	B (8-bit)	+	-	+	-
00010	NOT A (8-bit)	+	-	+	-
00011	NOT B (8-bit)	+	-	+	-
00100	A + B (8-bit)	+	+	+	+
00101	A + B + Carry (8-bit)	+	+	+	+
00110	A – B (8-bit)	+	+	+	+
00111	A AND B (8-bit)	+	-	+	-
01000	A OR B (8-bit)	+	-	+	-
01001	A XOR B (8-bit)	+	-	+	-
01010	A NAND B (8-bit)	+	-	+	-
01011	LSL A (8-bit)	+	+	+	-
01100	LSR A (8-bit)	+	+	+	-
01101	ASR A (8-bit)	+	+	-	-
01110	CSL A (8-bit)	+	+	+	-
01111	CSR A (8-bit)	+	+	+	-

FunSel	ALUOut	Z	C	N	O
10000	A (16-bit)	+	-	+	-
10001	B (16-bit)	+	-	+	-
10010	NOT A (16-bit)	+	-	+	-
10011	NOT B (16-bit)	+	-	+	-
10100	A + B (16-bit)	+	+	+	+
10101	A + B + Carry (16-bit)	+	+	+	+
10110	A – B (16-bit)	+	+	+	+
10111	A AND B (16-bit)	+	-	+	-
11000	A OR B (16-bit)	+	-	+	-
11001	A XOR B (16-bit)	+	-	+	-
11010	A NAND B (16-bit)	+	-	+	-
11011	LSL A (16-bit)	+	+	+	-
11100	LSR A (16-bit)	+	+	+	-
11101	ASR A (16-bit)	+	+	-	-
11110	CSL A (16-bit)	+	+	+	-
11111	CSR A (16-bit)	+	+	+	-

SYSTEM MUXOUT

MuxASel	MuxAOut
00	ALUOut
01	ARF OutC
10	Memory Output
11	IR (7:0)

MuxBSel	MuxBOut
00	ALUOut
01	ARF OutC
10	Memory Output
11	IR (7:0)

MuxCSel	MuxCOut
0	ALUOut(7-0)
1	ALUOut(15-8)