System Programming

Lecture #5

Professor Chung Yung

CSIE Dept., NDHU, Fall 2022

Machine-Dependent Features

- Instruction Formats and Addressing Modes
- Translation to Machine Code
- Program Relocation

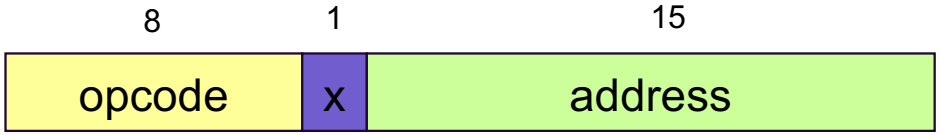
Instruction Formats

And Addressing Modes

SIC Instructions

There are two addressing modes available in SIC

- Indicated by *x* bit in the instruction
- (*X*) represents the contents of register *X*



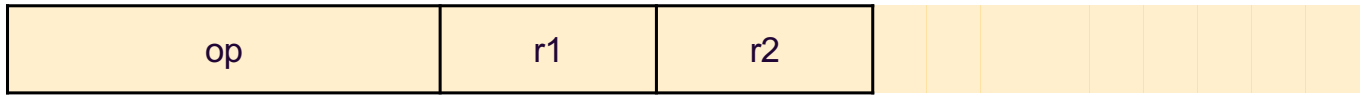
Direct vs. Indexed Addressing Modes

Mode	Indication	Target Address
Direct	x = 0	TA = address
Indexed	x = 1	TA = address + (X)

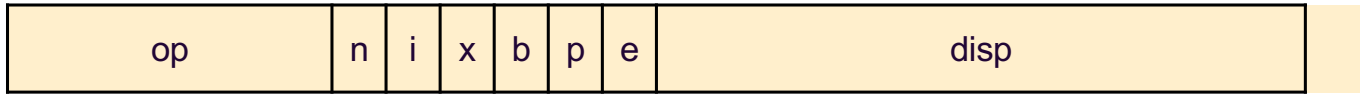
SIC/XE Instructions

Format 1 (1 byte)

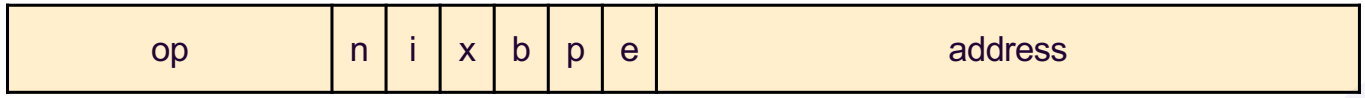
Format 2 (2 bytes)



Format 3 (3 bytes)



Format 4 (4 bytes)



Addressing Modes - 1/2

Base relative (format 3)

- b=1, p=0, TA=(B)+disp (0 ≤ disp ≤ 4095)

Program-counter relative (format 3)

- b=0, p=1, TA=(PC) + disp (-2048 ≤ disp ≤ 2047)

Direct (simple) addressing (formats 3 & 4)

- n=0, i=0 – Standard SIC
- n=1, i=1 – Format 3 of SIC/XE
- b=0, p=0, TA = disp

Addressing Modes - 2/2

Indexed addressing

- x=1, (X) is added in the target address calculation

Immediate addressing

- i=1, n=0, TA itself is the operand value.

Indirect addressing

- i=0, n=1, Target Address = (TA)

Format 4

- e=1

Note that we have introduced:

- Indexed base relative
- Indexed program-counter relative

Basic Formats

Format	Addressing Mode
op m	Simple, Base-relative, PC-relative
op @m	Indirect addressing
op #c	Immediate addressing
+op m	Extended format
op m, X	Indexed addressing
opR	Register-register instructions

An Example Program (1/3)

Line	Source statement			
5	COPY	START	0	COPY FILE FROM INPUT TO OUTPUT
10	FIRST	STL	RETADR	SAVE RETURN ADDRESS
12		LDB	#LENGTH	ESTABLISH BASE REGISTER
13		BASE	LENGTH	
15	CLOOP	+JSUB	RDREC	READ INPUT RECORD
20		LDA	LENGTH	TEST FOR EOF (LENGTH = 0)
25		COMP	#0	
30		JEQ	ENDFIL	EXIT IF EOF FOUND
35		+JSUB	WRREC	WRITE OUTPUT RECORD
40		J	CLOOP	LOOP
45	ENDFIL	LDA	EOF	INSERT END OF FILE MARKER
50		STA	BUFFER	
55		LDA	#3	SET LENGTH = 3
60		STA	LENGTH	
65		+JSUB	WRREC	WRITE EOF
70		J	@RETADR	RETURN TO CALLER
80	EOF	BYTE	C'EOF'	
95	RETADR	RESW	1	
100	LENGTH	RESW	1	LENGTH OF RECORD
105	BUFFER	RESB	4096	4096-BYTE BUFFER AREA

An Example Program (2/3)

```
110     .
115     .       SUBROUTINE TO READ RECORD INTO BUFFER
120     .
125     RDREC   CLEAR   X           CLEAR LOOP COUNTER
130             CLEAR   A           CLEAR A TO ZERO
132             CLEAR   S           CLEAR S TO ZERO
133             +LDT    #4096
135     RLOOP   TD      INPUT       TEST INPUT DEVICE
140             JEQ     RLOOP       LOOP UNTIL READY
145             RD      INPUT       READ CHARACTER INTO REGISTER A
150             COMPR   A,S         TEST FOR END OF RECORD (X'00')
155             JEQ     EXIT        EXIT LOOP IF EOR
160             STCH    BUFFER,X    STORE CHARACTER IN BUFFER
165             TIXR    T           LOOP UNLESS MAX LENGTH
170             JLT     RLOOP         HAS BEEN REACHED
175     EXIT    STX     LENGTH      SAVE RECORD LENGTH
180             RSUB                RETURN TO CALLER
185     INPUT   BYTE    X'F1'       CODE FOR INPUT DEVICE
```

An Example Program (3/3)

```
195         .
200         .       SUBROUTINE TO WRITE RECORD FROM BUFFER
205         .
210         WRREC   CLEAR   X           CLEAR LOOP COUNTER
212                 LDT     LENGTH
215         WLOOP   TD      OUTPUT      TEST OUTPUT DEVICE
220                 JEQ     WLOOP       LOOP UNTIL READY
225                 LDCH    BUFFER,X    GET CHARACTER FROM BUFFER
230                 WD      OUTPUT      WRITE CHARACTER
235                 TIXR    T           LOOP UNTIL ALL CHARACTERS
240                 JLT     WLOOP         HAVE BEEN WRITTEN
245                 RSUB                RETURN TO CALLER
250         OUTPUT  BYTE    X'05'       CODE FOR OUTPUT DEVICE
255                 END     FIRST
```

Addressing Modes Used 1/2

Indirect addressing

- Adding the prefix @ to operand (line 70)

Immediate addressing

- Adding the prefix # to operand (lines 25, 55, 133)

Base-relative addressing

- TA = Address – (B) (lines 160, 225)

Addressing Modes Used 2/2

PC-relative addressing

- TA = Address – (PC) (lines 10, 12, 20, 30, 40, 45, 50, 60, 70, 135, 140, 145, 155, 170, 175, 212, 215, 220, 230, 240)

Extended format

- Adding the prefix + to OP code (lines 15, 35, 65)

Register-register instructions

- Faster and no extra memory reference (Line 125, 130, 132, 150, 165, 210, 235)

Translation to Machine Code

Translated Object Code (1/3)

```
 5      0000    COPY      START   0
10      0000    FIRST     STL     RETADR          17202D
12      0003              LDB     #LENGTH         69202D
13                        BASE    LENGTH
15      0006    CLOOP     +JSUB   RDREC           4B101036
20      000A              LDA     LENGTH          032026
25      000D              COMP    #0              290000
30      0010              JEQ     ENDFIL          332007
35      0013              +JSUB   WRREC           4B10105D
40      0017              J       CLOOP           3F2FEC
45      001A    ENDFIL    LDA     EOF             032010
50      001D              STA     BUFFER          0F2016
55      0020              LDA     #3              010003
60      0023              STA     LENGTH          0F200D
65      0026              +JSUB   WRREC           4B10105D
70      002A              J       @RETADR         3E2003
80      002D    EOF       BYTE    C'EOF'          454F46
95      0030    RETADR    RESW    1
100     0033    LENGTH    RESW    1
105     0036    BUFFER    RESB    4096
```

36

Translated Object Code (2/3)

```
110                 .
115                 .       SUBROUTINE TO READ RECORD INTO BUFFER
120                 .
125     1036        RDREC   CLEAR   X               B410
130     1038                CLEAR   A               B400
132     103A                CLEAR   S               B440
133     103C                +LDT    #4096           75101000
135     1040        RLOOP   TD      INPUT           E32019
140     1043                JEQ     RLOOP           332FFA
145     1046                RD      INPUT           DB2013
150     1049                COMPR   A,S             A004
155     104B                JEQ     EXIT            332008
160     104E                STCH    BUFFER,X        57C003
165     1051                TIXR    T               B850
170     1053                JLT     RLOOP           3B2FEA
175     1056        EXIT    STX     LENGTH          134000
180     1059                RSUB                    4F0000
185     105C        INPUT   BYTE    X'F1'           F1
```

Translated Object Code (3/3)

```
195                 .
200                 .       SUBROUTINE TO WRITE RECORD FROM BUFFER
205                 .
210     105D        WRREC   CLEAR   X               B410
212     105F                LDT     LENGTH          774000
215     1062        WLOOP   TD      OUTPUT          E32011
220     1065                JEQ     WLOOP           332FFA
225     1068                LDCH    BUFFER,X        53C003
230     106B                WD      OUTPUT          DF2008
235     106E                TIXR    T               B850
240     1070                JLT     WLOOP           3B2FEF
245     1073                RSUB                    4F0000
250     1076        OUTPUT  BYTE    X'05'           05
255                         END     FIRST
```

Register and Address

Address translation

- Most register-memory instructions use PC-relative or Base-relative addressing.
- Format 3: 12-bit disp field
  - PC-relative: -2048 ~ 2047
  - Base-relative: 0 ~ 4095

Register translation									
Register	A	X	L	B	S	T	F	PC	SW
#	0	1	2	3	4	5	6	8	9

START and Instructions

START assembler directive

- Specifies a beginning address of 0.

Register-register instructions

- CLEAR, TIXR, COMPR

Register-memory instructions

- Program-counter (PC) relative addressing
- The program counter is advanced after each instruction is fetched and before it is executed.
- PC contains the address of the next instruction

ni And xbpe

n	i	Opcode+?	Addressing Mode
1	1	Opcode+3	Simple
1	0	Opcode+2	Indirect
0	1	Opcode+1	Immediate

x	b	p	e	Addressing Mode
0	0	1	0	PC-relative
0	1	0	0	Base-relative
1	0	0	0	Indexed
0	0	0	1	Extended

Example 1

PC-relative

- 10 0000 FIRST STL RETADR 17202D

  TA - (PC) = disp = 30H – 3H = 02D

  Opcode + ni = 14 + 3 = 17

- 40 0017 J CLOOP 3F2FEC

  TA – (PC) = disp = 0006 – 001A = -14 = FEC

  Opcode + ni = 3C + 3 = 3F

Example 2

Indexed Base-relative

- 160 104E STCH BUFFER, X 57C003

  TA - (B) = disp = 0036 – 0033 = 003

  Opcode + ni = 54 + 3 = 57

Examples 3 and 4

Extended

- 15 0006 CLOOP +JSUB RDREC 4B101036

  Opcode + ni = 4A + 1 = 4B

Immediate

- 55 0020 LDA #3 010003

  Opcode + ni = 00 + 1 = 01

- 133 103C +LDT #4096 75101000

  Opcode + ni = 74 + 1 = 75

Example 5

PC-relative + Indirect

- 70 002A J @RETADR 3E2003

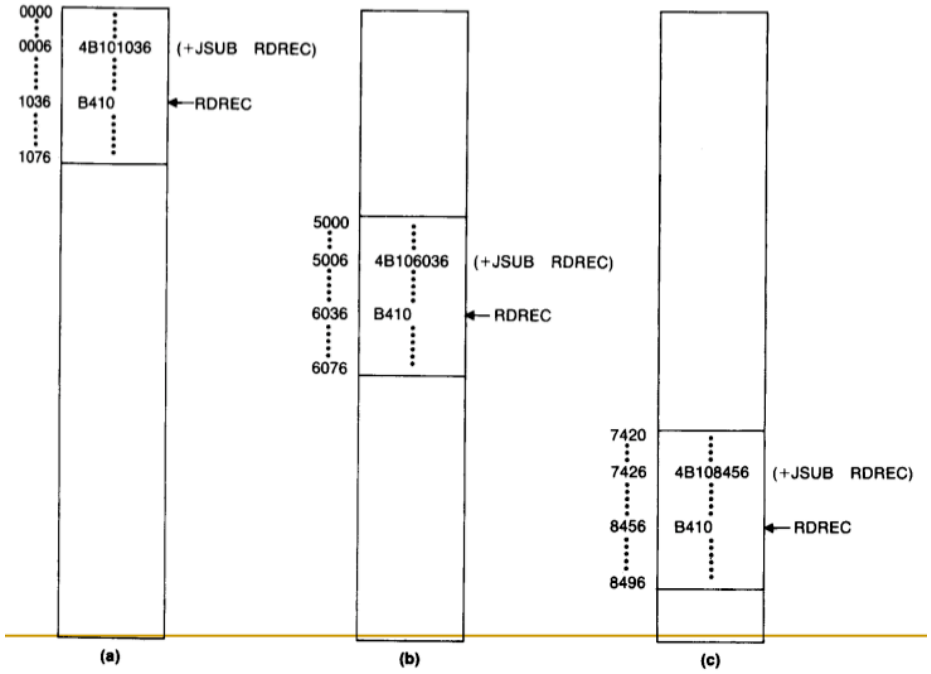
  TA = (operand) = (RETADR – (PC))

  = (0030 – 002D) = (003)

  Opcode + ni = 3C + 2 = 3E

Program Relocation

Basic Idea



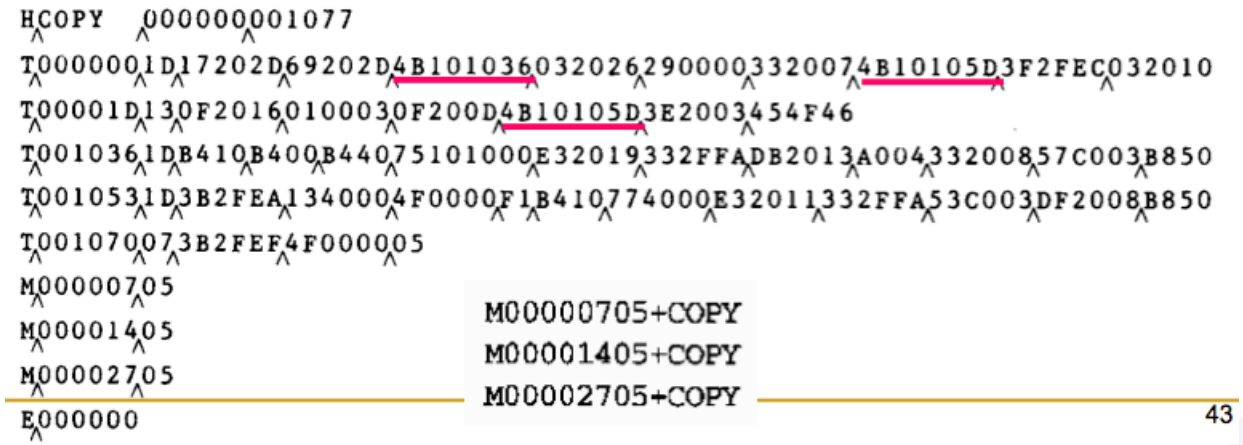
Method

We can solve the relocation problem in the following way:

- When the assembler generates the object code for the JSUB instruction, it will insert the address of RDREC relative to the start of the program. This is the reason we initialize the location counter to 0.
- The assembler produces a command for the loader, instructing it to add the beginning address of the program to the address field in the JSUB instruction at load time.

Modify Records

1	M
2-7	Starting location of the address field to be modified, relative to the beginning of the program
8-9	Length of the address field to be modified, in half-bytes



Quiz #4

Exercise

Quiz #4 Exercise

Translate each statement in the following assembly program into SIC/XE object code.

- LDA (00)
- LDS (6C)
- ADD (18)
- SUB (1C)
- STA (0C)
- ADDR (90)

```
1000        LDS   INCR
1003        LDA   ALPHA
1006        ADDR  S, A
1008        SUB   #1
100B        STA   BETA
            .
100E ALPHA  RESW  1
1011 BETA   RESW  1
1014 INCR   RESW  1
1017
```