CS2422 Assembly Language & System Programming

December 7, 2006

Today's Topic

- Assembler: Machine Dependent Features
  - SIC/XE
  - Program Relocation
  - Modification Records in an Object File.

Study Guide

- Sections 2.2 (especially 2.2.2) of Beck's "*System Software*" book.
  - Section 2.2: Program Relocation

SIC/XE Assembler

- We have learned the 2-pass assembler for SIC.
- What's new for SIC/XE?
  - More addressing modes.
  - Program Relocation.

An SIC/XE Example (Figure 2.6)

Line	Loc	Source statement			Object code
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	

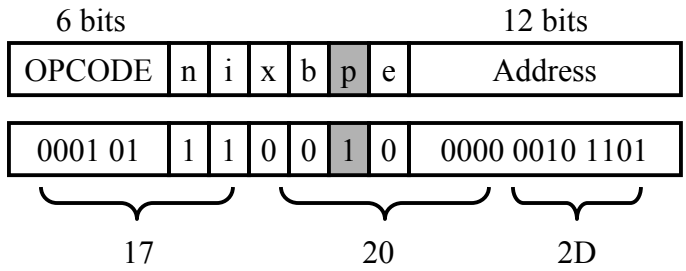
115		.	READ RECORD INTO BUFFER		
120		.			
125	1036	RDREC	CLEAR	X	B410
130	1038		CLEAR	A	B400
132	103A		CLEAR	S	B440
133	103C		+LDT	#4096	75101000
135	1040	RLOOP	TD	INPUT	E32019
140	1043		JEQ	RLOOP	332FFA
145	1046		RD	INPUT	DB2013
150	1049		COMPR	A,S	A004
155	104B		JEQ	EXIT	332008
160	104E		STCH	BUFFER,X	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
195		.			
200		.	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
		...(omitted)			

A Case of Object Code Generation

- Figure 2.6, Line 10

  `STL RETADR → 17 20 2D`

- The mode bit p=1, meaning PC relative addressing mode.



Instruction Format and Addressing Mode

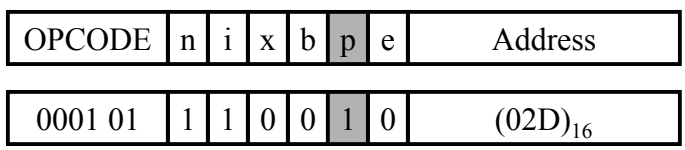
- SIC/XE
  - PC-relative or Base-relative addressing: op m
  - Indirect addressing: op @m
  - Immediate addressing: op #c
  - Extended format: +op m
  - Index addressing: op m,x
  - register-to-register instructions
  - larger memory -> multi-programming (program allocation)

Translation

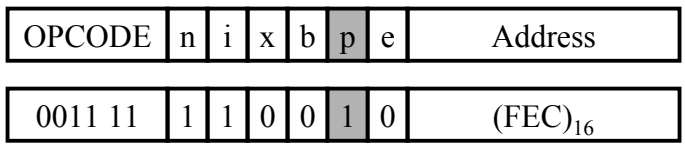
- Register translation
  - Register name (A, X, L, B, S, T, F, PC, SW) and their values (0,1, 2, 3, 4, 5, 6, 8, 9)
  - Preloaded in SYMTAB
- Address translation
  - Most register-memory instructions use program counter relative or base relative addressing
  - Format 3: 12-bit address field
    - Base-relative: 0~4095
    - PC-relative: -2048~2047
  - Format 4: 20-bit address field

PC-Relative Addressing Mode

- PC-relative
  - 10 0000 FIRST STL RETADR 17202D



    - Displacement= RETADR - PC = 30-3 = 2D
  - 40 0017 J CLOOP 3F2FEC



    - Displacement= CLOOP-PC= 6 - 1A= -14= FEC

Base-Relative Addressing Modes

- Base-relative
  - Base register is under the control of the programmer
  - 12 LDB #LENGTH
  - 13 BASE LENGTH
  - 160 104E STCH BUFFER, X 57C003

OPCODE	n	i	x	b	p	e	Address
0101 01	1	1	1	1	0	0	$(003)_{16}$

    - Displacement= BUFFER - B = 0036 - 0033 = 3
  - NOBASE is used to inform the assembler that the contents of the base register no longer be relied upon for addressing

Immediate Address Translation (1/2)

- Immediate addressing
  - 55 0020 LDA #3 010003

OPCODE	n	i	x	b	p	e	Address
0000 00	0	1	0	0	0	0	$(003)_{16}$

  - 133 103C +LDT #4096 75101000

OPCODE	n	i	x	b	p	e	Address
0111 01	0	1	0	0	0	1	$(01000)_{16}$

Immediate Address Translation (2/2)

- Immediate addressing
  - 12 0003 LDB #LENGTH 69202D

OPCODE	n	i	x	b	p	e	Address
0110 10	0	1	0	0	1	0	$(02D)_{16}$

  - 12 0003 LDB #LENGTH 690033

OPCODE	n	i	x	b	p	e	Address
0110 10	0	1	0	0	0	0	$(033)_{16}$

  - The immediate operand is the symbol LENGTH
  - The address of this symbol LENGTH is loaded into register B
  - LENGTH=0033=PC+displacement=0006+02D
  - If immediate mode is specified, the target address becomes the operand

Indirect Address Translation

- Indirect addressing
  - Target addressing is computed as usual (PC-relative or BASE-relative)
  - Only the n bit is set to 1
  - 70 002A J @RETADR 3E2003

OPCODE	n	i	x	b	p	e	Address
0011 11	1	0	0	0	1	0	$(003)_{16}$

  - TA=RETADR=0030
  - TA=(PC)+disp=002D+0003

Program Relocation

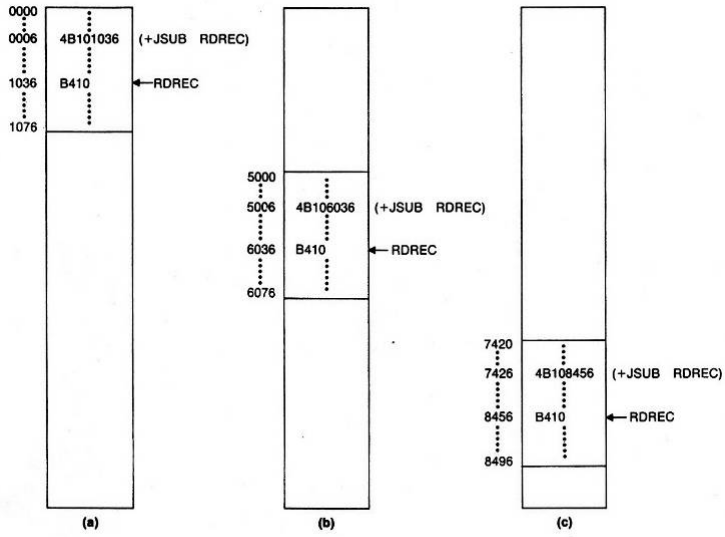


Figure 2.7 Examples of program relocation.

Examples of Program Relocation (1/2)

- Example Fig. 2.2
  - *Absolute program*, starting address 1000

5	1000	COPY	START	1000	
10	1000	FIRST	STL	RETADR	141033
15	1003	CLOOP	JSUB	RDREC	482039
20	1006		LDA	LENGTH	001036
25	1009		COMP	ZERO	281030
30	100C		JEQ	ENDFIL	301015
35	100F		JSUB	WREC	482061
40	1012		J	CLOOP	3C1003
45	1015	ENDFIL	LDA	EOF	00102A
50	1018		STA	BUFFER	0C1039
55	101B		LDA	THREE	00102D
60	101E		STA	LENGTH	0C1036
65	1021		JSUB	WREC	482061
70	1024		LDL	RETADR	081033
75	1027		RSUB		4C0000
80	102A	EOF	BYTE	C'EOF'	454E46
85	102D	THREE	WORD	3	000003
90	1030	ZERO	WORD	0	000000
95	1033	RETADR	RESW	1	
100	1036	LENGTH	RESW	1	
105	1039	BUFFER	RESB	4096	

Examples of Program Relocation (1/2)

- Example Fig. 2.2
  - *Absolute program*, starting address ~~1000~~ → 2000

```
5     2000   1000   COPY     START   ~~1000~~ → 2000     141033  142033
10    2000   1000   FIRST    STL     RETADR              482039  483039
15    2003   1003   CLOOP    JSUB    RDREC               001036  002036
20    2006   1006            LDA     LENGTH              281030  282030
25    2009   1009            COMP    ZERO                301015  302015
30    200C   100C            JEQ     ENDFIL              482061  483061
35    200F   100F            JSUB    WREC                3C1003  3C2003
40    2012   1012            J       CLOOP               00102A  00202A
45    2015   1015   ENDFIL   LDA     EOF                 0C1039  0C2039
50    2018   1018            STA     BUFFER              00102D  00202D
55    201B   101B            LDA     THREE               0C1036  0C2036
60    201E   101E            STA     LENGTH              482061  483061
65    2021   1021            JSUB    WREC                081033  082033
70    2024   1024            LDL     RETADR              4C0000  4C0000
75    2027   1027            RSUB                        454E46  454E46
80    202A   102A   EOF      BYTE    C'EOF'              000003  000003
85    202D   102D   THREE    WORD    3                   000000  000000
90    2030   1030   ZERO     WORD    0
95    2033   1033   RETADR   RESW    1
100   2036   1036   LENGTH   RESW    1
105   2039   1039   BUFFER   RESB    4096
```

Examples of Program Relocation (2/2)

- Example Fig. 2.6:
  - Except for absolute address, the rest of the instructions need not be modified
    - not a memory address (immediate addressing)
    - PC-relative, Base-relative
  - The only parts of the program that require modification at load time are those that specify direct addresses

```
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   0036   BUFFER   RESB    4096
```

Examples of Program Relocation (2/2)

- Example Fig. 2.6:
  - Except for absolute address, the rest of the instructions need not be modified
    - not a memory address (immediate addressing)
    - PC-relative, Base-relative
  - The only parts of the program that require modification at load time are those that specify direct addresses

5	1000	0000	COPY	START	~~0~~ → 1000		
10	1000	0000	FIRST	STL	RETADR	17202D	17202D
12	1003	0003		LDB	#LENGTH	69202D	69202D
13				BASE	LENGTH		
15	1006	0006	CLOOP	+JSUB	RDREC	4B101036	4B102036
20	100A	000A		LDA	LENGTH	032026	032026
25	100D	000D		COMP	#0	290000	290000
30	1010	0010		JEQ	ENDFIL	332007	332007
35	1013	0013		+JSUB	WRREC	4B10105D	4B10205D
40	1017	0017		J	CLOOP	3F2FEC	3F2FEC
45	101A	001A	ENDFIL	LDA	EOF	032010	032010
50	101D	001D		STA	BUFFER	0F2016	0F2016
55	1020	0020		LDA	#3	010003	010003
60	1023	0023		STA	LENGTH	0F200D	0F200D
65	1026	0026		+JSUB	WRREC	4B10105D	4B10205D
70	102A	002A		J	@RETADR	3E2003	3E2003
80	102D	002D	EOF	BYTE	C'EOF'	454F46	454F46
95	1030	0030	RETADR	RESW	1		
100	1036	0036	BUFFER	RESB	4096		

How to Make Program Relocation Easier

- Use program-counter (PC) relative addresses
  - Did you notice that we didn't modify the addresses for **JEQ**, **JLT** and **J** instructions?
  - We didn't modify the addresses for **RETADR**, **LENGTH**, and **BUFFER** in Figure 2.6 either.
- Virtual memory! (*Not covered in this course*)

Relocatable Program

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

Object File with M-Records

- Modification records are added to the object files. (See pp.64-65 and Figure 2.8.)
- Example:

```
HCOPY 001000 001077
T000000 1D 17202D…4B101036…
T00001D ……
…
M000007 05   ← Modification Record
……
E000000
```

Object Code

```
HCOPY  000000001077
T0000001D17202D69202D4B10103603202629000033200074B10105D3F2FEC032010
T00001D130F20160100030F200D4B10105D3E2003454F46
T001036 1DB410B400B44075101000E32019332FFADB2013A00433200857C003B850
T001053 1D3B2FEA1340004F0000F1B410774000E32011332FFA53C003DF2008B850
T00107007 3B2FEF4F000005
M00000705
M00001405
M00002705
E000000
```

Figure 2.8 Object program corresponding to Fig. 2.6.