INDIAN INSTITUTE OF TECHNOLOGY ROORKEE

System Software
CSN-252
Assembler

2.3.5 (Control Sections)
- L. L. Beck

Control Sections and Program Linking

- **Control Section**
  - Part of the program that maintain their identity
  - are most often used for subroutines or other logical subdivisions of a program
  - the programmer can assemble, load, and manipulate each of these control sections separately
  - instruction in one control section may need to refer to instructions or data located in another section
    - External references
    - External Symbols

2

```
int x = 7;

int p2();

int main(){
  p2();
}
```

```
#include <stdio.h>

extern int x;

int p2(){
  printf("%d\n", x);
}
```

Control Sections and Program Linking

- How assembler processes external references?
- There should be some means for linking control sections together
- Assembler generates information for each **external reference**
- Loader does required linking
- Assembler directive csect

4

External Definition and References

- Symbols in one control section can not be used directly in other control section
- External definition
  - **EXTDEF name [, name]**
  - symbols that are defined in this control section and may be used by other sections
- External reference
  - **EXTREF name [,name]**
  - symbols that are used in this control section and are defined elsewhere
- **Control section names do not need to be named in an EXTDEF**
- Assembler uses a separate location counter for each control section (starting from 0)

```
1   copy    start   0
2           extdef  buffer, bufend, length
3           extref  rdrec
4   first   stl     retadr
5   cloop   +jsub   rdrec
6           lda     length
7           comp    #0
8           jeq     endfil
9           j       cloop
10  endfil  j       @retadr
11  retadr  resw    1
12  length  resw    1
13  buffer  resb    4096
14  bufend  equ     *
15  maxlen  equ     bufend-buffer
        :
```

```
16  rdrec   csect
17          extref  buffer,length, bufend
18          clear   x
19          clear   a
20          clear   s
21          ldt     maxlen
22  loop    td      input
23          jeq     loop
24          rd      input
25          compr   a, s
26          jeq     exit
27          +stch   buffer,x
28          tixr    t
29          jlt     loop
30  exit    +stx    length
31          rsub
32  input   byte    x'f3'
33  maxlen  word    buffer-bufend
34          end     first
```

1		copy	start	0	
2			extdef	buffer, bufend, length	
3			extref	rdrec	
4	0000	first	stl	retadr	172013
5	0003	cloop	+jsub	rdrec	?
6	0007		lda	length	03200F
7	000a		comp	#0	290000
8	000d		jeq	endfil	332003
9	0010		j	cloop	3F2FF0
10	0013	endfil	j	@retadr	3E2000
11	0016	retadr	resw	1	
12	0019	length	resw	1	
13	001c	buffer	resb	4096	
14	101c	bufend	equ	*	
15	1000	maxlen	equ	bufend-buffer	

* Assembler will pass this information to loader

16	0000	rdrec	csect		
17			extref	buffer, length, bufend	
18	0000		clear	x	b410
19	0002		clear	a	b400
20	0004		clear	s	b440
21	0006		ldt	maxlen	77201f
22	0009	loop	td	input	e3201b
23	000c		jeq	loop	332ffa
24	000f		rd	input	db2015
25	0012		compr	a, s	a004
26	0014		jeq	exit	332009
27	0017		+stch	buffer,x	…
28	001b		tixr	t	b850
29	001d		jlt	loop	3b2fe9
30	0020	exit	+stx	length	…
31	0024		rsub		4f0000
32	0027	input	byte	x'f3‘	f3
33	0028	maxlen	word	buffer-bufend	…
34	002B	end	first		

This maxlen is diff

Control Sections and Program Linking

- Assembler must check that in which control section the symbol is defined
- Using it in another control section must be flagged as error unless it is in EXTREF
- Same symbol name can be used in different control sections

9

Implementation

- The assembler must include information in the object program that will cause the loader to insert proper values where they are required
- Define record
  - Col. 1 D
  - Col. 2-7 Name of external symbol defined in this control section
  - Col. 8-13 Relative address within this control section (hexadeccimal)
  - Col.14-73 Repeat information in Col. 2-13 for other external symbols
- Refer record
  - Col. 1 R
  - Col. 2-7 Name of external symbol referred to in this control section
  - Col. 8-73 Name of other external reference symbols

10

Modification Record

- Modification record
  - Col. 1 M
  - Col. 2-7 Starting address of the field to be modified (hexiadecimal)
  - Col. 8-9 Length of the field to be modified, in half-bytes (hexadeccimal)
  - Col. 10 Modification Flag (+ or -)
  - Col.11-16 External symbol whose value is to be added to or subtracted from the indicated field
  - Note: control section name is automatically an external symbol, i.e. it is available for use in Modification records.
- The same modification records can be used for relocation
  M00000705 → M00000705+COPY

11

1		copy	start	0	
2			extdef	buffer, bufend, length	
3			extref	rdrec	
4	0000	first	stl	retadr	172013
5	0003	cloop	+jsub	rdrec	4B100000
6	0007		lda	length	03200F
7	000a		comp	#0	290000
8	000d		jeq	endfil	332003
9	0010		j	cloop	3F2FF0
10	0013	endfil	j	@retadr	3E2000
11	0016	retadr	resw	1	
12	0019	length	resw	1	
13	001c	buffer	resb	4096	
14	101c	bufend	equ	*	
15	1000	maxlen	equ	bufend-buffer	

H COPY__000000101c
D BUFFER00001cBUFEND00101cLENGTH000019
R RDREC
T 000000 16 172013 4B100000 03200F 290000 332003 3F2FF0 3E2000
M000004 05 +RDREC
E 000000

- Object programs for two control section

```
H COPY__000000 00101c
D BUFFER00001cBUFEND00101cLENGTH000019
R RDREC
T 000000 16 172013 4B100000 03200F 290000 332003 3F2FF0
   3E2000
M000004 05 +RDREC
E 000000


H RDREC_ 000000 00002B
R BUFFER LENGTH BUFEND
T 000000 1D B410 … B850
T 00001D 0E 3B2FE9 … 000000
M 000018 05 +BUFFER
M 000021 05 +LENGTH
M 000028 06 -BUFEND
M 000028 06 +BUFFER
E
```

13

External References in Expression

- Earlier definitions
  - required all of the relative terms be paired in an expression (an absolute expression), or that all except one be paired (a relative expression)
- New restriction
  - Both terms in each pair must be relative within the same control section
  - Ex: BUFEND-BUFFER
  - Ex: RDREC-COPY
- In general, the assembler cannot determine whether or not the expression is legal at assembly time.
- Handled by a linking loader.
- Assembler evaluates all the terms it can and combines these to form an initial expression
- And generates modification records for loader

14