CON. CORPOR.

Sheryl Modlinger

CYBER CROSS SYSTEM VERSION 1 DIAGNOSTIC HANDBOOK

CONTROL DATA®
CYBER 170 SERIES
CYBER 70 SERIES MODELS 72, 73, 74
6000 SERIES COMPUTER SYSTEMS
CYBER 18 COMPUTER SYSTEMS
255X HOST COMMUNICATIONS PROCESSORS

REVISION RECORD

REVISION	DESCRIPTION
A	Manual released.
(12/75)	
B	Manual revised to reflect NOS/BE 1.1 changes and manual name change from CCP Support Software to
(4/76)	CYBER Cross System Version 1.

Publication No.
96836300

Printed in the United States of America

Address comments concerning this manual to:
Control Data Corporation
Publications and Graphics Division
4455 Eastgate Mall
La Jolla, California 92037
or use Comment Sheet in the back of this manual.

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

Page	Revision	Feature
Cover	--	
Title Page	--	
ii	B	
iii	B	
v	B	
vii	B	
1-1	B	
2-1	B	
2-2	B	
2-3	B	
2-4	B	
2-5	B	
2-6	B	
3-1	B	
3-2	B	
4-1	B	
4-2	B	
4-3	B	
4-4	B	
5-1	B	
5-2	B	
6-1	B	
6-2	B	
6-3	B	
7-1	B	
7-2	B	
7-3	B	
Comment Sheet	B	
Envelope	--	
Back Cover	--	

Page	Revision	Feature

PREFACE

This handbook is intended to provide a complete and convenient source of the error and diagnostic messages produced by the CONTROL DATA® CYBER Cross System. The CYBER Cross System software operates under control of the CYBER 170/70/6000 NOS or NOS/BE operating system.

The reader is assumed to be familiar with the operation of the CYBER 18 computer.

Additional information for the CYBER Cross System software can be found in the following publications:

Description	Publication No.
CYBER Cross System PASCAL Compiler Reference Manual	96836100
CYBER Cross System Macro Assembler Reference Manual	96836500
CYBER Cross System Micro Assembler Reference Manual	96836400
CYBER Cross System Link Editor and Library Maintenance Programs Reference Manual	60471200

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

CONTENTS

FIGURES

TABLES

INTRODUCTION 1

The CONTROL DATA® CYBER 18 Cross System provides a means of developing software for the CYBER 18 computer and the CDC 255x Host Communications Processor on a CYBER 170/70/6000 computer. The CYBER Cross System consists of the following components:

- PASCAL Compiler
- Macro Assembler (CLASS)
- Micro Assembler
- Library Maintenance Program (MPLIB)
- Link Editor (MPLINK)

Error messages and diagnostics for these components are given in the following sections.

PASCAL ERROR MESSAGES 2

1: SCALAR TYPE EXPECTED

The subrange must be explicit or there must be a subrange-type identifier.

2: INTEGER TOO LARGE

The integer is greater than 32,767.

3: ERROR IN CONSTANT

A + or - was encountered instead of a legal constant.

4: '=' EXPECTED

In a CONST or VALUE section, each identifier should be followed by =.

5: FIELDNAME DECLARED TWICE

There is a duplicate field name within the same record.

6: BAD RANGE

An error occurred in the subrange.

7: TAG FIELD TYPE BAD

In a variant record definition, the tag field-type identifier was not in the numeric subrange or a symbolic type.

8: NAME DECLARED TWICE

9: ')' EXPECTED

10: ':' EXPECTED

11: IDENTIFIER EXPECTED

12: IDENTIFIER NOT DECLARED

An identifier was encountered which has not been defined.

13: INDEX MUST BE OF SCALAR TYPE

In an array definition, the index type must be a subrange or symbolic type; i.e., cannot be pointer, power, array, or record.

14: 'OF' EXPECTED

15: TEN OR MORE ERRORS ON THIS LINE

16: PROCEDURE DECLARED TWICE

17: 'END' EXPECTED

18: ERROR IN TYPE DECLARATION

An error occurred in the record definition or in the TYPE declaration.

19: RANGE RESTRICTED TO 0 THROUGH 15

A constant less than 0 or greater than 15 was used in the sets.

20: ERROR IN VALUE PART

In the VALUE section, an attempt was made to initialize an array with an illegal symbol in the value list or with too many values listed for the size of the array.

21: TOO MANY ARGUMENTS FOR THIS PROCEDURE

More than 20 arguments were in the procedure call.

22: VALUE IS OUT OF RANGE

An attempt was made to assign a value outside the subrange to a variable whose type is a subrange.

23: TOO MANY RELATIVE PROCEDURE NAMES OR EXIT LABELS

More than 100 combined relative procedure names and exit labels declared.

24: ERROR IN DECLARATION PART

The next symbol, following all declarations (LABEL, CONST, TYPE, VAR, PROCEDUREs, FUNCTIONs) in the block, was not BEGIN.

25: LOWBOUND>HIGHBOUND

An error was made in the subrange definition.

26: NOT A VARIABLE IDENTIFIER

In the VALUE section, an identifier was not variable.

27: LABEL TOO LARGE

The label was greater than 9,999.

28: SYMBOLIC SUBRANGE TYPE NOT ALLOWED

29: PARAMETER MISSING IN FUNCTION DECLARATION

The FUNCTION identifier was not followed with a (.

30: TOO MANY UNIQUE EXTERNAL TYPE REFERENCES

More than 10 pointer-type declarations pointed to undefined types.

31: TOO MANY UNIQUE EXTERNAL REFERENCES

More than 40 external names were in the procedure.

32: VARIABLE OR FIELD IDENTIFIER EXPECTED

33: EXPRESSION TOO COMPLICATED

34: TYPE OF VARIABLE SHOULD BE ARRAY

A [was not preceded by the keyword ARRAY.

35: TYPE OF EXPRESSION MUST BE SCALAR

Scalar-type expressions are required for array indices and set elements.

36: CONFLICT OF INDEX TYPE WITH DECLARATION

The array index used does not agree with the type of array declaration.

37: ']' EXPECTED

38: TYPE OF VARIABLE SHOULD BE RECORD

A nonrecord appeared in a WITH statement.

39: NO SUCH FIELD IN THIS RECORD

40: TYPE OF VARIABLE SHOULD BE POINTER

A nonpointer-type variable preceded an ↑.

41: FIELD NAME EXPECTED

A record identifier was followed by a . with no field name specified.

42: ILLEGAL SYMBOL IN EXPRESSION

While processing an expression, a symbol was encountered that was not a constant, variable, function identifier, (, ¬ (not), or [(set).

43: UNDEFINED LABEL

GOTO EXIT 1, where 1 has not been declared in a LABEL declaration in any encompassing block; GOTO 1, where 1 never occurs in block.

44: ILLEGAL TYPE OF PARAMETER IN STANDARD FUNCTION OR PROCEDURE

Standard function and procedures expect predefined types.

45: TYPE IDENTIFIER IN STATEMENT PART

While processing an expression, a type identifier was encountered.

46: PROCEDURE USED AS FUNCTION

While processing an expression, a procedure-name identifier was encountered.

47: TYPE OF STANDARD FUNCTION PARAMETER SHOULD BE INTEGER

48: INDEX OUT OF RANGE

The explicit index value for an array is outside of the declared index range.

49: '[' EXPECTED

50: ILLEGAL TYPE OF OPERAND

The type of operand does not match the operator.

51: 'v' CANNOT BE USED AS MONADIC OPERATOR

52: ':=' EXPECTED

Occurs if a statement begins with a variable identifier or a FOR statement and variable identifier that are not followed by : =.

53: ASSIGNMENT NOT ALLOWED

Assignment was made in which the type of assigned variable and the expression do not agree.

54: ILLEGAL SYMBOL IN STATEMENT

55: TYPE OR CONSTANT IDENTIFIER

The statement begins with a type or constant identifier.

56: 'THEN' EXPECTED

THEN does not appear after IF and a Boolean expression.

57: TYPE OF EXPRESSION IS NOT BOOLEAN

58: ';' EXPECTED

59: 'DO' EXPECTED

60: ILLEGAL PARAMETER SUBSTITUTION

The actual parameter type does not agree with the formal declaration in the procedure call.

61: LABEL EXPECTED

There is no label before each statement in CASE or no label following GOTO.

62: ILLEGAL TYPE OF EXPRESSION

The type of expression in the CASE or FOR statement is not numeric or symbolic.

63: CONSTANT EXPECTED

64: TYPE DECLARED TWICE

65: BAD FUNCTION TYPE

The identifier following the function header must be a type identifier.

66: TAG FIELD MISSING FOR THIS VARIANT

The tag field is missing in the standard procedure NEW.

67: 'UNTIL' EXPECTED

68: ONLY '=' AND '≠' ALLOWED HERE

When comparing arrays or records, only = or ≠ can be used.

69: LOOP CONTROL VARIABLE MUST BE SIMPLE AND LOCAL OR GLOBAL

70: 'TO' OR 'DOWNTO' EXPECTED

71: TOO MANY CASES IN CASE STATEMENT

More than 100 cases were in the CASE statement.

72: NUMBER OF PARAMETERS DOES NOT AGREE WITH DECLARATION

73: MIXED TYPES

When using set notation [ ], all expressions appearing between brackets must be of the same type. With the FOR statement, initial and final values must be of the same type.

74: TOO MANY LABELS IN THIS PROCEDURE

More than 100 labels appeared in any procedure.

75: TOO MANY CONSTANTS, YET UNDEFINED LABELS, OR TEMP. STORAGE REFERENCES

76: DEPTH OF PROCEDURE NESTING TOO LARGE

Procedures were nested to more than seven levels.

77: LABEL DEFINED MORE THAN ONCE

78: TOO MANY EXIT LABELS

More than 10 exit labels were used.

79: '(' EXPECTED

80: ',' EXPECTED

No , was used between parameters in standard procedures PACK, UNPACK, INSERT, APPEND, or ADDR.

81: TOO MANY EXIT LABELS OR FORWARD PROCEDURES

More than 10 exit labels or forward procedures were used.

82: TOO MANY NESTED WITH STATEMENTS

More than 20 nested WITH statements were used.

83: VALUE DECLARATION IN RECURSIVE PROCEDURE

84: TOO MANY CONSTANTS IN THIS PROCEDURE

There are more than 40 constants in the procedure.

85: ASSIGNMENT TO FUNCTION IDENTIFIER MUST OCCUR IN FUNCTION ITSELF

86: ACTUAL PARAMETER MUST BE A VARIABLE

The actual parameter is not a variable; the procedure declaration specified a VAR-type parameter.

87: PACKED FIELD NOT ALLOWED HERE

A packed field was used as a parameter in a procedure call where the procedure declaration specified a VAR-type parameter.

88: OPERATORS '<' AND '>' ARE NOT DEFINED FOR POWERSETS

89: REDUNDANT OPERATION ON POWERSETS

An attempt to test for the null set ([ ]) contained in any power set.

90: PROCEDURE TOO LONG

Too much code was generated for the procedure or program.

91: BEGIN COMMENT CHARACTER (⌐→) IMBEDDED IN COMMENT

The special character which indicates the beginning of a comment may not be used in the body of the comment.

MACRO ASSEMBLER ERROR MESSAGES 3

3.1 ASSEMBLY ERROR MESSAGES

When an error occurs during assembly, it is flagged with a two-character error mnemonic in columns 5-6 of the line in error on the listing. If more than one error occurs in the same line of code, the most recent error encountered is the one that is flagged. However, all errors are reflected in the total error count.

Error Mnemonic	Meaning
DS	Doubly defined symbol
UD	Undefined symbol
EX	Illegal expression
OP	Illegal operation code
RL	Illegal *relocation*
OV	Numeric operand overflow
IS	Illegal symbol name
PD	Programmer-defined error

3.2 MACRO ERROR DIAGNOSTIC MESSAGES

When an error is encountered in a macro definition or call, a two-character error mnemonic appears on the line following the erroneous code, along with a diagnostic message.

Error Mnemonic	Diagnostic Message
MD	*/MAC DEF ERROR NO NAME
MD	*/MAC DEF ERROR BAD PARAM
MD	*/MAC DEF ERROR NO OPCODE
MD	*/MAC DEF ERROR BUFFER OVE
MD	*/MAC DEF ERROR BAD TERM
MC	*/MACRO CALL ERROR NEST GT 10
MC	*/MACRO CALL ERROR BUFFER OV

Error Mnemonic	Diagnostic Message
MC	*/MACRO CALL ERROR PARAM ERR
MC	*/MACRO CALL ERROR BAD CONT
MC	*/MACRO CALL ERROR BAD DEF

3.3 MISSING END CARD

If no END card is encountered by the Macro Assembler at the end of the program, the following message is output at the end of the listing:

/// END CARD MISSING ///

3.4 DAYFILE ERROR MESSAGES

If the Macro Assembler encounters a control card error or insufficient field length has been assigned for the job, an error message is output to the dayfile and the job is aborted. The following messages are output to the dayfile, indicating the erroneous condition:

CONTROL CARD ERROR
NO INPUT FILE SPECIFIED
FILE NAME CONFLICT
INVALID TAB COLUMNS (for tidy feature)
INVALID LIST OPTIONS
INSUFFICIENT FL FOR CLASS
MEMORY OVERFLOW IN PASS 1
MEMORY OVERFLOW IN PASS 2

The following messages that are output to the dayfile, indicate erroneous conditions which exist, but which do not cause the job to abort.

MACTXT OVERFLOW
Q99 ILLEGAL TO CLASS
SYSTEM ERROR IN PASS 1 ASSEMBLY
(If this message occurs, there is probably a bug in the Macro Assembler)
nn ERRORS IN xxxxxx
(Where xxxxxx is the program name.)

MICRO ASSEMBLER ERROR CODES 4

The assembler prints numeric error codes to flag and diagnose incorrect assembly statements. When a statement is in error, one to four error codes are listed to the right of the statement describing the problem. Table 4-1 contains a list of the Micro Assembler error codes. Figure 4-1 is an example of a listing containing error codes.

Table 4-1. Micro Assembler Error Codes

NUMERIC CODE	MEANING OR CAUSE
1	The A code is set by a shift operation in F.
2	The A code is undefined.
3	The B code is set by a shift operation in F.
4	The B code given is undefined.
5	A C and M field conflict occurred.
6	Cannot reach page; the S code is set and cannot be used to reach the page specified in the C field for the jump command.
7	C and B field conflict.
8	C code is undefined.
9	*A multiply-defined label was encountered.*
10	Not used
11	The D code is setting an S code in conflict with A or B.
12	The D code is undefined.
13	The EQU pseudo instruction needs a symbol in the label field.
14	The F code is undefined.
15	The M code is undefined.
16	A different S code is set by the A and B fields.
17	The S code is already set by A, B, or D.
18	The S code is undefined.
19	An illegal T code was given for a jump, N=, or K=.
20	A T code is required but is not specified; it is assumed to be U.

Table 4-1. Micro Assembler Error Codes (Continued)

NUMERIC CODE	MEANING OR CAUSE
21	The T code is undefined.
22	An undefined symbol was encountered; the field containing the symbol is specified by the next integer.
23	There is an illegal character in the HEX, DEC, or OCT constant.
24	Not used
25	The numeric value is not in the range (0-$FFF)
26	The micro memory location is greater than 4,095.
27	The first character on the card is not *, $, +, -, or blank.
28	The shift code in the C field is illegal when the S field contains the L8EA instruction.
29	The D field must be an NOP if: • The F field is a shift or scale, or • The B field is MMU or MML
30	KZU in the T field is illegal if the C field is INCK.
31	NZU in the T field is illegal if the C field is INCN.
32	Macro-memory read in the C or S fields is illegal if the instruction time is E or F.
33	The A field may not be blank on an EQU card. The symbol in the L field is undefined.
34	This location has already been used. The next integer specifies the number of the card that caused the location to be previously assembled. This instruction overrides the one previously assembled.
35	Not used
36	The address is out of range (less than zero or greater than 4,095).
37	A symbol in an address expression is longer than eight characters.
38	The number of spaces requested between characters on the deadstart object cards is less than zero or greater than three. If this error occurs, one space will be punched between characters.

Table 4-1. Micro Assembler Error Codes (Continued)

NUMERIC CODE	MEANING OR CAUSE
39	An error has caused the label in columns 2 through 9 of an EQU card not to be defined. The error may be one of the following: • Use of a symbol that has not been previously defined in the EQU expression. • Use of a symbol that is larger than eight characters in the EQU expression. • The value of the EQU expression is greater than 4,095 or less than zero.

In addition to the error codes in Table 4-1, there are four other conditions:

Message	Printed	Meaning
*****CHECKSUM ERROR*****	At end of the source code listing. The error is detected in pass 2.	Contents of the address in object code output to contain twos complement checksum was not zero. The object code output is produced without a checksum.
STOP 5	CYBER 18-17 System: On list output device when the error occurs. The error is detected in pass 2. CYBER System: In dayfile.	Binary object code; the page read from mass storage was not the page requested. An error was detected in the subroutine GETPAG.
STOP 444	CYBER 18-17 System: On list output device when error occurs. The error is detected in pass 1. CYBER System: In dayfile.	Symbol table overflow; more than 10,000 symbols have been defined.
STOP 777	CYBER 18-17 System: On list output device when a FINIS card is read. CYBER System: In dayfile.	This is not an error. It signifies normal job termination.

```
CARD VALUE T P/MA  MICRO-MEM   LOCATION  F     A     B     D     S     C       MT    COMMENT                          DIAGNOSTICS
   1                           IDENT DIAGNOSTIC EXAMPLE FOR THE MP ASSEMBLER

   2                           ********************************************************************************
   3                           *                                                                              *
   4                           *          DIAGNOSTIC EXAMPLE                                                  *
   5                           *                                                                              *
   6                           ********************************************************************************

   7   000                      TROUBLE  EQU                                                  UNDEFINED EQUATE          33
   8   000                               EQU   5                                            UNDEFINED LABLE           13
   9   00A                               OPG   AX
  10       0  00A  5808 0000    SYM                                                        FIRST SYMBOL DUP           9
  11       1  00A  5806 2000                                 NOP
  12       0  008  5808 0000    SYM                                                        DUPLICATE SYMBOL           9
  13       1  008  9808 200A                                          SYN     J          USE OF DUP. SYMBOL
  14       0  00C  5808 0000    LOC                                 NOP                    GOOD STATEMENT
  15       1  00C  5806 2000              MP                                             UNDEFINED FUNCTION        14
  16       0  00D  5808 0000                    Q                                        CAN'T USE Q AS A  INPUT    2
  17       1  00D  5808 2000                          A                                  CAN'T USE A AS B  INPUT    4
  18       0  00E  5806 0000                                XM1                          UNDEFINED DESTINATION     12
  19       1  00E  5808 2008                                     PAP                     UNDEFINED S CODE          22   18
  20       0  00F  5808 0000    X                                                          ILLEGAL CHAR IN COL 1     27
  21       1  00F  5808 2008                                          JOE                UNDEFINED C FIELD         22    6
  22       0  010  8FFF DFFF                                               X            ILLEGAL M CODE            15
  23       1  010  5808 2080                                               BTL          ILLEGAL T CODE            21
  24       0  011  7C80 8000 E            ALOE  X                                        A,B MUST BE BLANK          1
  25       1  011  7C80 2000 E            ALOE        X                                  A,B MUST BE BLANK          3
  26       0  012  5570 8700 B                  A*    INTA                               A,B CAN'T BOTH BE PRIME   16
  27       1  012  5E10 2700 B                  SM1         A*                           A',D'' ILLEGAL            11
  28       0  013  5E18 0700 B                  SM1                 READ                 A' S CONFLICT             17
  29       1  013  5400 2000 B                        INRD          LOEA                 B' S CONFLICT             17
  30       0  014  5800 0000                                A*    GATEI                  D' S CONFLICT             17
  31       1  014  54C8 2000                          N                 OECK             B AND C CONFLICT           7
  32       0  015  0808 0FFF                                          N=35    R          C AND N FIELD CONFLICT     5
  33       1  015  9808 2300 G                                       READ  100X    J      NEED S FIELD FOR PAGE      6
  34       0  016  D808 100F                                          N=15     LQL       ILLEGAL T FOR N=          19
  35       1  016  9808 200C                                          LOC     JLQL       ILLEGAL T FOR JUMP        19
  36       0  017  9808 200C                                          LOC     JU         LEGAL T FOR JUMP
  37   00A                      ORG   10
  38       0  00A  565D 0000    A+B   P     X     A                                      ORG OVER EXISTING CODE    34   10
  39   000                      ORG   5000                                               ORG TO NON-EXISTANT LOC.  25   36
  40                           END

     27 LINES CONTAIN ERRORS

CARD VALUE T P/MA  MICRO-MEM   LOCATION  F     A     B     D     S     C       MT    COMMENT                          DIAGNOSTICS
```

Figure 4-1. Sample Micro Assembler Listing

MPLIB ERROR MESSAGES 5

I/O ERROR — READING LGO FILE

An I/O error occurred while reading the LGO file.

TOO MANY PROGRAMS ON THE LGO FILE

More than the maximum allowable number of programs was encountered while processing the LGO file.

NO END-OF-TABLE WORD FOR ENTRY POINT TABLE RECORD ON OLD LIBRARY FILE

The file specified as the old library file is in the wrong format.

I/O ERROR — READING OLD LIBRARY FILE

An I/O error occurred while reading the old library file.

UNRECOGNIZABLE DIRECTIVE

This directive is not legal.

PUT DIRECTIVE DOES NOT HAVE MATCH ON LGO FILE

A PUT directive was input for a program that could not be found on the LGO file.

DEL DIRECTIVE DOES NOT HAVE MATCH ON OLD LIBRARY FILE

A DEL directive was input for a program that could not be found on the old library file.

AN ATTEMPT WAS MADE TO WRITE TOO MANY PROGRAMS TO THE NEW LIBRARY FILE

The combined number of programs transferred from the LGO and old library files to the new library file exceeded the maximum allowable.

I/O ERROR — WRITING INTERMEDIATE LIBRARY FILE

An I/O error occurred while writing the intermediate library file.

NO XFR BLOCK FOR PROGRAM ON RANDOM LGO FILE

There is a bad object program on the LGO file.

I/O ERROR — WRITING NEW LIBRARY FILE

An I/O error occurred while writing the new library file.

I/O ERROR — READING INTERMEDIATE LIBRARY FILE

An I/O error occurred while reading the intermediate library file.

NON-ASCII (NOT $20-$5F) CHARACTER IN PROGRAM NAME OR ENTRY POINT ON THE LGO FILE

An illegal program name or entry point symbol appears in the program on the LGO file.

NEW LIBRARY ENTRY POINT TABLE OVERFLOW

The capacity of the entry point table for the new library was exceeded.

MPLINK ERROR MESSAGES 6

Error messages from MPLINK are flagged by a leading up-arrow character (↑) followed by the error message. In the case of a recognized syntax error, the up-arrow points to the last character scanned when the error was recognized.

ENTRY POINT TABLE IS FULL

The maximum number of entry point names, module names, or synonyms has exceeded the capacity of the entry point table.

xxxxxxx
AN ILLEGAL END OF FILE READ

Where xxxxxxx = TEMP OBJ or TEMP ABS. A temporary work file has encountered an end of file illegally. Indicative of an internal error.

ILLEGAL OBJECT TEXT

Either the LGO or the NEWLIB library file is improperly formatted. Possibly an incorrect file is attached.

ILLEGAL LINK PHASE DIRECTIVE

The keyword immediately following an asterisk was not a recognized address.

ILLEGAL SYMBOL IN EXPRESSION

The expression evaluator encountered a symbol it did not expect. Possibly an illegal character.

EXPRESSION OPERAND STACK OVERFLOW

The expression evaluator encountered too many operands during evaluation. This problem may be alleviated by restarting the expression with additional nested parenthesis groupings.

ILLEGAL ARITHMETIC OPERATOR

The expression evaluator expected an arithmetic operator.

xxxxxx MULTIPLE ENTRY DEFINITION

An entry point name has appeared in the object text in more than one module, or the name specified in an ENT directive has been previously encountered. xxxxx is the multiple defined name.

OPERAND EXPECTED

The expression evaluator expected a numeric valued operand.

NO ASSOCIATED LEFT PARENTHESIS

The expression evaluator encountered no corresponding left parenthesis for the right parenthesis.

MISSING RIGHT PARENTHESIS

The expression evaluator did not encounter a right parenthesis to associate with a previously encountered left parenthesis.

PERIOD EXPECTED

COMMA EXPECTED

TOO MANY EXPLICIT LINK NAMES

The number of specified *L directives specified exceeds the maximum number permitted.

TOO MANY OVERLAY AREAS

The number of overlay areas declared, via the *OVLY directive, exceeds the maximum number permitted.

TOO MANY SYNONYMS

The number of synonyms declared, via the *SYN directive, exceeds the maximum number permitted.

TOO MANY MODS IN OBJECT LIBRARY

The maximum number of object text modules that may be encountered on a single library, either LGO or NEWLIB, has been exceeded.

UNSATISFIED EXTERNAL OVERFLOW

The maximum number of unsatisfied external references or forward external references has exceeded the capacity of the Link Editor.

COMMON AREA EXCEEDED

The blank COMMON specification encountered in an object text module exceeds the area allocated by the *COM directive.

DATA AREA EXCEEDED

A named COMMON specification encountered in an object text module exceeds the area allocated by the *DAT directive.

*DAT AND GLOBL$ CONFLICT

The *DAT directive has been specified and the GLOBL$ object text module has been encountered; the assigned GLOBL$ area does not fit within the designated *DAT area.

LINK BEYOND MP MEMORY LIMIT

The Link Editor has assigned memory locations beyond the memory limit designated in the *COR directive.

ANOTHER LOADER MOD FOUND

More than one object module with the name LOADER has been encountered.

EXT REFERENCE OUT OF BOUNDS, POSSIBLE MODULE TOO LARGE xxxxxx

An external reference link address has crossed the boundary of a memory image block being developed. This situation occurs when attempting to generate a module image that exceeds the internal buffer size. xxxxxx is the module name in which the external reference exists.

MPEDIT ERROR MESSAGES 7

Error messages from MPEDIT are flagged by a leading up-arrow character (↑) followed by the error message. In the case of a recognized syntax error, the up-arrow points to the last character scanned when the error was recognized.

ILLEGAL SYMBOL IN EXPRESSION

The expression evaluator encountered a symbol it did not expect.

EXPRESSION OPERAND STACK OVERFLOW

The expression evaluator encountered too many operands during evaluation. This problem may be alleviated by restarting the expression with additional nested parenthetical groupings.

ILLEGAL ARITHMETIC OPERATOR

The expression evaluator expected an arithmetic operator.

xxxxxx MULTIPLE ENTRY DEFINITION

A local identifier, array name, or overlay name has been multiply defined. xxxxxx is the multiply defined name.

OPERAND EXPECTED

The expression evaluator expected a numeric valued operand.

NO ASSOCIATED LEFT PARENTHESIS

The expression evaluator encountered no corresponding left parenthesis for the right parenthesis.

MISSING RIGHT PARENTHESIS

The expression evaluator did not encounter a right parenthesis to associate with a previously encountered left parenthesis.

PERIOD EXPECTED

COMMA EXPECTED

:= EXPECTED

= EXPECTED

[EXPECTED

] EXPECTED

; EXPECTED

(EXPECTED

) EXPECTED

OF EXPECTED

The keyword OF was expected.

BEGIN EXPECTED

The keyword BEGIN was expected.

END EXPECTED

The keyword END was expected.

TO EXPECTED

The keyword TO was expected.

DO EXPECTED

The keyword DO was expected.

ILLEGAL END

The keyword END is not permitted in this position.

ILLEGAL KEYWORD

The keyword encountered is incorrectly specified.

NOT A LOCAL VARIABLE

The indicated local identifier was not previously specified in the VAR section.

OUT OF RANGE

The array index value is not within the range declared for the array within the ARRAY section.

ILLEGAL DIMENSIONS COUNT

The number of index values specified for the array does not agree with the number declared within the ARRAY section.

STUFF TABLE OVERFLOW

The number of requested editing values exceeds the internal capacity of MPEDIT. It is possible to present the editing information in sections by recalling MPEDIT for each section.

ARRAY TABLE OVERFLOW

The number of arrays declared within the ARRAY section exceeds the capacity of MPEDIT.

DIMENSIONS TABLE OVERFLOW

The gross number of dimensions declared for all arrays within the ARRAY section exceeds the capacity of MPEDIT.

Within the track listing, any address reference to a location that does not exist on the memory image, load file is indicated with the following comment suffixed to the edit information:

**** OUT OF RANGE

COMMENT SHEET

MANUAL TITLE CONTROL DATA® CYBER Cross System Version 1
Diagnostic Handbook

PUBLICATION NO. 96836300 REVISION B

FROM NAME:

BUSINESS ADDRESS:

COMMENTS: This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number to which your comment applies.

CUT ALONG LINE

STAPLE

STAPLE

FOLD

FIRST CLASS
PERMIT NO. 333

LA JOLLA, CA.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
PUBLICATIONS AND GRAPHICS DIVISION
4455 EASTGATE MALL
LA JOLLA, CALIFORNIA 92037

CUT ALONG LINE

FOLD

STAPLE

STAPLE