APPENDIX K

ERROR MESSAGES

K.1 MACRO ERROR CODES

MACRO error codes are printed following a field of six asterisk characters and on the line preceding the source line containing the error. For example:

```
 ******A
26 00236  000002'    .WORD REL1+REL2
```

The addition of two relocatable symbols is flagged as an A error.

Error Code	Meaning
A	Addressing error. An address within the instruction is incorrect. Also may indicate a relocation error.
B	Bounding error. Instructions or word data are being assembled at an odd address in memory. The location counter is updated by +1.
D	Doubly-defined symbol referenced. Reference was made to a symbol which is defined more than once.
E	End directive not found. (A listing is generated).
I	Illegal character detected. Illegal characters which are also non-printing are replaced by a ? on the listing. The character is then ignored.
L	Line buffer overflow, i.e., input line greater than 132 characters. Extra characters on a line, (more than 72_{10}) are ignored.
M	Multiple definition of a label. A label was encountered which was equivalent (in the first six characters) to a previously encountered label.
N	Number containing 8 or 9 has decimal point missing.
O	Opcode error. Directive out of context.

Error Code	Meaning
P	Phase error. A label's definition of value varies from one pass to another. A P error code also appears if a .ERROR directive is assembled.
Q	Questional syntax. There are missing arguments or the instruction scan was not completed or a carriage return was not immediately followed by a line feed or form feed.
R	Register-type error. An invalid use of or reference to a register has been made.
T	Truncation error. A number generated more than 16 bits of significance or an expression generated more than 8 bits of significance during the use of the .BYTE directive.
U	Undefined symbol. An undefined symbol was encountered during the evaluation of an expression. Relative to the expression, the undefined symbol is assigned a value of zero.
Z	Instruction which is not compatible among all members of the PDP-11 family (11/15, 11/20, 11/45).

ERROR MESSAGES

The MACRO assembler outputs the following messages when one of the related errors is detected.

COMMAND I/O ERROR
ILLEGAL SWITCH
INPUT FILE MISSING
INSUFFICIENT MEMORY TO COMPLETE ASSEMBLY
I/O ERROR ON OUTPUT FILE
OPEN FAILURE ON INPUT FILE
OPEN FAILURE ON OUTPUT FILE
OUTPUT DEVICE FULL
TOO MANY OUTPUT FILES

The error messages are self-explanatory.

K.2 FORTRAN ERROR DIAGNOSTICS

K.2.1 FORTRAN Compiler Error Diagnostics

Compiler error diagnostics are of three basic classes.

- F - Fatal errors, which must be corrected before the program can run correctly.
- W - Warning errors, which should be corrected. The program is not likely to run with W class errors.
- I - Information errors, which consist of minor syntax errors that do not affect the code generated by the Compiler.

One other error class is of interest to the user. This is the "S" (special) class. Error messages in this class are never issued unless the user specifies the /ER switch in the command string to the Compiler. "S" class errors are those errors that may be in conflict with ANSI usage or that may be errors only under unusual usages in the program. Each "S" error should be carefully considered within the context of the program. For most programs, these errors may be safely ignored.

Error diagnostics issued by the Compiler consist of two or three lines. The first line consists of a short section of the source line where the error was assumed to occur. The second line contains the error number; the third, if present, consists of the text of the error message. This text message is not printed if the Compiler error diagnostic file, FORCOM.DGN, is not present on the system disk under the [1,1] area.

The following is an example of a section of a FORTRAN program with an error diagnostic, as issued by the Compiler:

```
ØØØ1                          DIMENSION NLIST(100)
ØØØ2                          DIMENSION LIST (100)
ØØØ3                          DIMENSION A(5,5)
ØØØ4                          DEFINE FILE LIST (100,10,U,IND)
[LE LIST(1)]
ERROR 122
F ILLEGAL FORM FOR UNIT NUMBER IN DEFINE FILE.
```

In a few cases, notably EQUIVALENCE errors, errors are detected after the source line containing the error is no longer available for printing. In this situation, no source line is printed with the rest of the diagnostic.

A list of the error diagnostics, with explanatory comments is presented on the following pages.

Ø I REDUNDANT CONTINUATION MARK: IT IS IGNORED.

A continuation mark that does not make sense has been seen. The Compiler ignores it and continues as if the mark had not existed.

1 I CONTINUATION MARK NOT IN RANGE 1 to 9; IT IS IGNORED.

This error occurs when a special character has been typed after a TAB in the free form input specifications. If a continuation is to be specified when using this type of Compiler input, the only meaningful form for the continuation mark is numeric. If it is desired to use non-numeric characters for continuation marks, the standard column 6 convention should be used.

2 I ILLEGAL STMT. NUMBER, NON-NUMERIC CHAR. IN COLS. 1-5.

This error occurs when any non-numeric characters appear in the statement number field of a line.

3 W ILLEGAL TYPE OR IMPLICIT STATEMENT, INTEGER IS ASSUMED.

This error is caused by the recognition of the optional word TYPE or the word IMPLICIT not followed by a recognizable type descriptor. The recognizable type descriptors are BYTE, LOGICAL*1, LOGICAL, LOGICAL*2, INTEGER, INTEGER*2, REAL, REAL*4, DOUBLE, DOUBLE PRECISION, REAL*8 and COMPLEX.

4 W NON-DECLARATIVE STATEMENT IN BLOCK DATA.

A statement other than a COMMON, EQUIVALENCE, DIMENSION, TYPE or DATA statement occurred within a BLOCK DATA subprogram.

5 F SYMBOL TABLE FULL.

The Compiler has run out of space to process the program. It is necessary to reduce the number of variables, constants or arrays, cut down on the number of continuation lines requested, specify I/O devices that use less memory, or reduce the number of I/O devices in use.

6 W TOO MANY CONTINUATION LINES, REMAINDER IGNORED.

More continuation lines were used than are allowed in the compilation. The default number of continuation lines is 5. If more or fewer are needed, specify the /CO:n switch in the command string, where "n" is the number of continuations desired; "n" may range from Ø to 99. Note that the space assigned to continuation lines in the Compiler subtracts from the symbol table space. Each continuation line requested uses approximately the same space as 5 symbols in the symbol table.

7 S MIXED MODE IS USED IN AN EXPRESSION.

This error is issued only if the /ER switch has been specified. This message flags all expressions where an implied mode conversion occurs.

8 F ILLEGAL UNARY OP., ONLY +,-, OR .NOT. ARE ALLOWED.

A binary operator was used in an expression as though it were a unary operator.

9 F CLOSING "/" MISSING ON BLOCK NAME.

The block name specifier in a COMMON statement is missing the final /. It is impossible to determine where the block name ended. This error is also issued if a block name is longer than 6 characters.

1Ø W ALL PORTIONS OF FORMAT MUST BE WITHIN OUTER LEVEL PARENS.

The only part of a FORMAT statement that may appear outside the enclosing parentheses is the key-word FORMAT.

11 W TOO MANY RIGHT PARENTHESES IN FORMAT.

There are unbalanced parentheses in the FORMAT statement.

12 W ILLEGAL CHARACTER(S) TERMINATING A STATEMENT.

This error is issued whenever characters remain after the Compiler finishes processing a statement. This usually occurs with other errors, but may occur in expressions that have illegal characters or operations specified.

13 F ILLEGAL FORM FOR A NUMERIC CONSTANT.

The number being processed by the Compiler has an illegal form. For example, a "." is flagged as a bad constant.

14 F INSUFFICIENT COMPILER SPACE TO EVALUATE THIS CONSTANT.

The Compiler ran out of space while trying to evaluate a constant. This problem has the same remedy as error 5 (symbol table overflow).

15 W INTEGER CONSTANT TOO BIG. LARGEST POS. VALUE ASSUMED.

The number in question was not in the range -32768 to +32767. It has been assigned the value 32767.

16 F ILLEGAL SYNTAX IN LIST ITEM.

One of the declarative list items is improperly constructed; perhaps a subscript was omitted or an extra oomma was added.

17 F ILLEGAL LIST TERMINATION.

The last declarative list item was not terminated by a carriage return/line feed. This may be caused by a variable name longer than six characters or an unrecognizable construct.

18 W ILLEGAL CHARACTER IN FORMAT STATEMENT.

A character was found that could not be part of any legal format specification.

19 F DIMENSION MUST BE SPECIFIED FOR EACH VAR. IN THE LIST.

A variable was specified in a DIMENSION statement without the required dimension list.

20 F DIMS. CONFLICT WITH THOSE OF AN EARLIER STATEMENT.

A DIMENSION was declared that contradicts an earlier statement.

21 W UNRECOGNIZED STATEMENT

The statement could not be recognized as a legal FORTRAN construct. It is probably misspelled. The Compiler made no attempt to compile it and has ignored it.

22 F ADJUSTABLE ARRAY NAME OR INDEX NOT A SUBPROGRAM PARAMETER.

Any adjustable array names or indices must be specified in the parameter list of the SUBROUTINE or FUNCTION statement. They are not allowed as local variables. They may not be specified in common.

23 W MISSING EXPONENT IN CONSTANT.

An E or D format constant was specified, but the exponent was omitted.

24 F MAXIMUM FUNCTION DEPTH (2Ø) EXCEEDED.

In an expression, function calls may be nested to a maximum depth of 2Ø.

25 F MISMATCHED PARENTHESES.

Unbalanced parentheses were found in an expression.

26 F NON-ARRAY REFERENCE TO ARRAY ITEM.

This error occurs in an arithmetic expression where an item that has been declared as an array was referenced as a simple variable. All array element references must specify a distinct subscript expression.

27 F CANNOT ASSIGN TO A CONSTANT.

A constant appeared on the left side of an arithmetic assignment.

28 F CANNOT ASSIGN TO A FUNCTION.

A function reference appeared on the left side of an arithmetic assignment.

29 W ILLEGAL CHAR. TERMINATING A STMT. OR POSS. BAD OPERATOR.

Similar to error 12; characters remain after processing an expression. The probable cause is a bad operator in the expression.

30 F SUBSCRIPT ON NON-ARRAY VARIABLE.

An attempt was made to use a variable that had not been declared in a DIMENSION statement as an indexed variable or array.

31 W NAME MUST BE 1-6 ALPHANUMERICS, THE FIRST ALPHABETIC.

The name specified was longer than six characters or did not begin with an alphabetic character.

32 F ILL. SUBSCRIPT IN AN ARRAY ASSIGNMENT OR MISSING SUBSCRIPT.

The subscript form used with an array variable was not recognizable to the Compiler.

33 F ILLEGAL OPERAND OR POSSIBLE ADJACENT OPERATORS.

The character(s) following a legal operator in an expression could not be recognized as a legal operand in the expression. Two operators were probably placed together.

34 F TOO MANY SUBSCRIPTS OR NO CLOSING PAREN FOR SUBSCRIPT.

More than three subscripts were specified for a subscripted variable, or the closing parenthesis of the subscript was omitted.

35 F NO FUNCTION ARGUMENTS PRESENT

All function references must have at least one argument specified.

36 F UNRECOGNIZABLE PARAMETER IN FUNCTION CALL.

An argument in a function reference could not be recognized.

37 F FUNCTION CALL MISSING A ")".

The closing parenthesis of a function reference was missing. A possible reason was that an ASCII string without a closing quotation mark was specified as one of the arguments.

38 F ILLEGAL ROUTINE NAME.

A function or subroutine name must consist of from 1 to 6 alphanumeric characters, the first of which must be alphabetic.

39 W MISSING END STATEMENT, END IS ASSUMED.

The Compiler did not find an END statement at the end of the file. The Compiler has inserted one to allow completion of the compilation.

40 W IMPROPERLY NESTED DO STATEMENT.

A DO loop, when nested, cannot overlap the range of any other DO statement.

41 W DO LIST OVERFLOW, NO MORE THAN 1Ø NESTED DO'S ARE ALLOWED.

The Compiler table that contains information about DO loop nesting has overflowed because an attempt was made to nest deeper than 1Ø DO loops.

42 W ILLEGAL SYNTAX IN COMMON/EQUIVALENCE..

A list item in a COMMON or EQUIVALENCE statement was incorrectly written.

43 W TABLE OVERFLOW IN COMMON/EQUIVALENCE.

The Compiler ran out of space while attempting to compile the statement. This error has the same solution as error 5. Also, if possible, reduce the number of COMMON or EQUIVALENCE declarations used.

44 W DUMMY VARIABLE OR ADJUSTABLE ARRAY USED IN COMMON.

It is illegal to use a dummy argument in a COMMON declaration because of possible data allocation conflicts. Also, if adjustable arrays are specified, they cannot occur in common.

45 W VARIABLE ALREADY IN COMMON, CANNOT BE REDEFINED.

An attempt was made to place in common a variable that was already there.

46 F ILLEGAL DO STATEMENT SYNTAX.

The DO statement specified had one or more unrecognizable parts.

47 F DO CONTROL VARIABLE IS NOT SIMPLE INTEGER VARIABLE OR CONSTANT.

The control variable can only be an integer variable or constant; it cannot be an array or any type other than integer.

48 F DO PARAMETER IS NOT A SIMPLE INTEGER VARIABLE OR CONSTANT.

The initial or terminal value of a DO loop is not a simple integer variable or constant.

49 W BAD STEP VALUE IN DO, IT IS ASSUMED TO BE 1.

The step value of a DO loop is not a simple integer variable. A value of 1 has been substituted.

50 W ILLEGAL CONSTANT IN PAUSE/STOP.

The constant in a PAUSE or STOP statement does not consist of a 1- to 6-digit octal constant with a value less than 177777.

51 W ILLEGAL OR MISSING STATEMENT LABEL, IT MUST BE NUMERIC.

The statement label field did not have a proper label consisting of one to five numeric digits.

52 F ILLEGAL SYNTAX IN GOTO/ASSIGN STATEMENT.

One or more unrecognizable items were found in a GOTO or ASSIGN statement.

53 W ILLEGAL PARAMETER IN STATEMENT FUNCTION LIST.

A parameter in an arithmetic statement function may consist only of simple variables; no constants or subscripted variables are allowed.

54 F ROUTINE NAME CANNOT BE A NUMERIC CONSTANT.

The name of a subroutine or function must consist of one to six alphanumeric characters, the first alphabetic.

55 W "SUBROUTINE" OR "FUNCTION" NOT FIRST STMT. OF ROUTINE.

A SUBROUTINE or FUNCTION statement cannot occur except as the first statement of a subprogram.

56 W ILLEGAL PARAMETER IN SUBROUTINE OR FUNCTION LIST.

A parameter in a subroutine or function list can only be a simple variable; no constants or subscripted variables are allowed.

57 W TOO MANY PARAMETERS (>127) IN ROUTINE LIST.

A parameter list for a subroutine or function may have no more than 127 parameters.

58 W CONSTANT MAY NOT BE DECLARED IN EXTERNAL.

The EXTERNAL statement does not allow constants to be specified in the list.

59 W TOO MANY LEFT PARENTHESES IN FORMAT.

The parentheses in a FORMAT statement are not correctly balanced.

60 W MISSING COMMA OR) IN COMMON/EQUIVALENCE.

An item described in a COMMON or EQUIVALENCE statement is terminated incorrectly.

61 W MISSING (IN COMMON/EQUIVALENCE.

A left parenthesis has been omitted in a COMMON or EQUIVALENCE statement.

62 W DUMMY ARGUMENT OR ADJUSTABLE ARRAY USED IN EQUIVALENCE.

It is illegal to attempt to equivalence an item to any item that was passed as a parameter to a subroutine or function.

63 W INCONSISTENT EQUIVALENCE.

An equivalence was specified that, if attempted, would produce inconsistent allocation of the variables in question.

64 F TWO OR MORE COMMON ITEMS ARE EQUIVALENCED.

An item in common may only be equivalenced to an item not in common.

65 F I/O UNIT IS NOT SIMPLE INTEGER VARIABLE OR CONSTANT.

The I/O unit number specified must be a simple integer variable or constant; it must not be an array element, or be of any type other than integer.

66 F ARRAY OR FUNCTION NAME NOT ALLOWED AS UNIT IN I/O STMT.

The I/O unit number specified must be a simple integer variable or constant; it must not be an array or function name, or be of any type other than integer.

67 F ILLEGAL SYNTAX IN I/O OR ENCODE/DECODE.

An unrecognizable form was found in an I/O, ENCODE, or DECODE statement.

68 F MISSING ARGUMENT IN FIND.

The FIND statement must have exactly two arguments.

69 F ILLEGAL RECORD DESIGNATOR IN RANDOM ACCESS READ/WRITE.

The record designator in a random access READ or WRITE must be a simple integer variable or constant.

70 F MISSING RIGHT PARENTHESIS IN I/O OR ENCODE/DECODE.

The closing parenthesis in an I/O, ENCODE or DECODE statement was omitted, or an unrecognizable parameter followed the last necessary parameter.

71 W ILLEGAL FORM FOR END= AND/OR ERR=.

The END= or ERR= specifications must refer to a legal 1- to 5-digit statement number.

72 W ILLEGAL FORM FOR LIST ITEM.

An I/O list item is unrecognizable. List items may consist of variables, arrays, subscripted variables and/or implied DO lists.

73 F ILLEGAL SYNTAX FOR REWIND, BACKSPACE, OR ENDFILE.

The REWIND, BACKSPACE and END FILE statements must specify a simple integer variable or constant as the logical unit number.

74 F NON-INTEGER PARAMETER IN REWIND, BACKSPACE, OR ENDFILE.

The parameter must be a simple integer variable or constant.

75 W ILLEGAL H CONSTANT IN FORMAT.

The Hollerith constant specified has been incorrectly formed.

76 W HOLLERITH CONSTANT COUNT TOO BIG.

The number of characters in a Hollerith field cannot exceed 255.

77 W SYNTAX ERROR IN IMPLICIT STATEMENT.

A variable specification in an IMPLICIT statement has been incorrectly formed.

78 W HOLLERITH CONSTANT IMPROPERLY TERMINATED BY END OF LINE.

A Hollerith constant count was larger than the available number of characters on the line.

79 W .NOT. MAY BE USED AS A UNARY OPERATOR ONLY.

An attempt was made to use .NOT. as a binary operator.

80 W EXPONENT MAY NOT BE LOGICAL*1, LOGICAL*2 OR COMPLEX.

An attempt was made to do exponentiation with an exponent of an illegal type.

81 W INTEGER**REAL OR COMPLEX**DOUBLE NOT ALLOWED.

The attempted form of exponentiation was illegal.

82 W COMPLEX**REAL OR COMPLEX**DOUBLE NOT ALLOWED.

The attempted form of exponentiation was illegal.

83 F IMPROPER LABEL SYNTAX IN IF STATEMENT.

A transfer label was specified that was not a 1- to 5-digit number.

84 W ANYTHING **COMPLEX NOT ALLOWED.

Complex exponentiation of this form is not allowed.

85 F MISSING COMMA IN READ, PRINT, OR PUNCH.

A comma must immediately follow the FORMAT statement number in a READ or PRINT statement.

86 F INCORRECT SYNTAX IN DEFINE FILE STATEMENT.

A DEFINE FILE was stated incorrectly.

87 W ARRAY MAY NOT BE DECLARED AS EXTERNAL.

Only functions and subroutines may be declared as EXTERNAL.

88 F ARRAY IS TOO LARGE.

An array was specified that could not possibly fit on a PDP-11.

89 F ILLEGAL ROUTINE NAME.

A subprogram name must consist of one to six alphanumeric characters, the first of which is alphabetic.

90 F ILLEGAL DO SPECIFICATION IN I/O OR ENCODE/DECODE.

The implied DO specification in the statement was incorrect.

91 F ILLEGAL LIST IN IMPLIED DO.

The I/O list inside an implied DO specification was incorrectly formed.

92 F ILLEGAL FORMAT SPEC. IN I/O OR ENCODE/DECODE.

An array specifier did not consist of a legal numeric FORMAT statement number or array name.

93 W SYNTAX ERROR IN THE EXPRESSION OF AN ASF.

The expression in an arithmetic statement function must follow the same rules as a normal arithmetic statement.

94 W MISSING "," OR ")" IN ASF.

An argument was not followed by a comma or right parenthesis. The argument may possibly be badly formed.

95 W MISPLACED "=" IN ASF.

The assignment portion of an arithmetic statement function is either missing or incorrectly specified.

96 F ASF NAME HAS BEEN PREVIOUSLY USED.

An arithmetic statement function must have a uniquely defined name.

97 W SUBSCRIPTS OUT OF BOUNDS IN DATA OR EQUIVALENCE.

An attempt was made to use DATA or EQUIVALENCE outside the range of an established array.

98 F ILLEGAL EXTENSION OF COMMON ORIGIN BY EQUIVALENCE.

The beginning of common cannot be extended using an EQUIVALENCE statement.

99 F OPENING "/" MISSING FROM DATA GROUP.

The start of the data group portion of a DATA statement could not be found.

100 W UNEQUAL NUMBER OF VARIABLES AND CONSTANTS.

When using the DATA statement, the number of constants specified must exactly match the number of variable or array items to be filled.

101 W DATA NOT ALLOWED IN COMMON EXCEPT IN "BLOCKDATA".

Common cannot be initialized by a DATA statement except in a BLOCK DATA subprogram.

102 F SUBSCRIPTS ON UNDIMENSIONED ELEMENT IN DATA.

An unsubscripted variable cannot be referenced with a subscript in any form.

103 F ADJUSTABLE ARRAY NOT ALLOWED IN DATA.

It is illegal to attempt to initialize items passed as parameters to a subprogram.

104 F PRESETTING NAMED COMMON ALLOWED ONLY IN "BLOCKDATA".

A common block may be initialized with a DATA statement only within a BLOCK DATA subprogram.

105 F ILLEGAL FORM FOR CONSTANT IN DATA.

A constant specified in a DATA statement was not recognizable to the Compiler.

106 F ILLEGAL REPEAT COUNT.

The repeat count in a DATA statement was incorrectly specified.

107 W MISMATCHED DATA TYPES.

The data types of the constants specified do not match the variable types in a DATA statement.

108 W DATA MUST FOLLOW ALL OTHER DECLARATIVES.

The DATA statement must come after DIMENSION, COMMON, EQUIVALENCE, or TYPE declaration statements.

109 I NO PATH TO THIS STATEMENT.

It is not possible for the program to execute this statement. This is usually caused by an unlabeled statement immediately following a GOTO or IF form.

110 W VARIABLE MAY NOT HAVE BEEN REDEFINED AFTER USE IN "ASSIGN".

A variable used in an expression was previously used in an ASSIGN statement and has not been redefined.

111 F ILLEGAL FORM FOR COMPLEX CONSTANT.

A complex constant may consist only of a left parenthesis followed by a real constant, a comma, and another real constant, followed by a right parenthesis.

112 W NUMBER OF DIMENSIONS NOT THE SAME AS DECLARED.

An attempt was made to use a subscripted variable with a different number of subscripts than was declared.

113 W RETURN IS ILLEGAL IN MAIN PROGRAM.

A RETURN statement is only legal within a subroutine or function.

114 W MISSING DO LOOP TERMINATION(S).

Every DO loop must be terminated by a numbered executable statement.

115 F DIMENS. NOT TERM. BY RIGHT PAREN OR TOO MANY DIMENSIONS.

A dimensioned variable may have, at most, three dimensions and must be terminated by a right parenthesis.

116 W ROUTINE NAME CANNOT BE DECLARED EXTERNAL.

A function or subroutine cannot be external to itself.

117 F ENCODE/DECODE LACKS LEGAL BUFFER DESCRIPTOR.

The buffer descriptor in an ENCODE or DECODE statement can only be the name of a legal array.

118 F ENCODE/DECODE BUFFER SIZE NOT SIMPLE INTEGER.

The buffer size descriptor in an ENCODE or DECODE statement must be a simple integer variable or constant.

119 F ARRAY OR FUNCTION CANNOT DESCRIBE BUFFER SIZE.

The buffer size descriptor in an ENCODE or DECODE statement must be a simple integer variable or constant.

120 S VARIABLE USED BUT NOT PREVIOUSLY DEFINED.

The variable in question has been used in an expression before a value was assigned to it. This error is occasionally issued in extraneous cases in CALL statements.

121 F DIAGNOSTIC TABLE OVERFLOW.

More than eight errors occurred on this statement. The extra error messages cannot be printed.

122 F ILLEGAL FORM FOR UNIT NUMBER IN DEFINE FILE.

The unit number must be a simple integer variable or constant.

123 F MISSING "(" IN DEFINE FILE.

The left parenthesis in a DEFINE FILE statement was omitted.

124 F ILLEGAL RECORD COUNT (M) IN DEFINE FILE.

The record count designator in a DEFINE FILE statement has been incorrectly stated or omitted.

125 F ILLEGAL RECORD LENGTH (L) IN DEFINE FILE.

The record length designator in a DEFINE FILE statement has been incorrectly stated or omitted.

126 F DEFINE FILE ONLY ALLOWS UNFORMATTED (U) MODE.

It is illegal to specify any mode other than U (unformatted mode).

127 F DEFINE FILE ASSOCIATED VARIABLE NOT A SIMPLE INTEGER.

The associated variable in a DEFINE FILE statement must be a simple integer variable.

128 W MISSING ")" IN DEFINE FILE.

The right parenthesis in a DEFINE FILE statement was omitted.

129 F EXPRESSION STACK OVERFLOW.

The specified arithmetic expression overflowed the stack space reserved for evaluation. Break the expression into smaller components.

130 W ILLEGAL FORM FOR OCTAL CONSTANT.

An octal constant must consist of a quotation mark followed by one to six octal digits; the value may not exceed 177777. The digits 8 and 9 may not appear in such a constant.

131 W OCTAL CONSTANT TOO LARGE.

An octal constant larger than 177777 was specified.

132 W OCTAL CONSTANT MUST HAVE AT LEAST ONE CHARACTER.

An octal constant must have at least one octal digit following the quotation mark.

133 W IMPROPER FORM FOR REAL CONSTANT.

A real constant was incorrectly formed. Possibly only the "." was specified.

134 I NO EXECUTABLE STATEMENTS IN A MAIN PROGRAM.

The main program unit contains no executable statements; it is therefore meaningless.

135 W MISSING COMMA

A declarative list is missing at least one comma.

136 W REDUNDANT COMMA.

An extra comma was found in a declarative list. This error occasionally occurs in combination with error 135 when illegal forms exist in the list and the Compiler is unable to recognize them.

137 W IMPLICIT STATEMENT APPEARS AFTER STATEMENT IT AFFECTS.

If an IMPLICIT statement appears after a statement it affects, the implicit typing may be done in an incomplete fashion. In some cases, the usage of the variables affected will be changed but the data attributes and array information will not change.

138 W LOGICAL OP. MEANINGFUL ONLY ON BYTE,LOGICAL, OR INTEGER.

Logical operations cannot be done in a meaningful fashion on real, double precision, or complex data items.

139 F EQUIVALENCE GROUP TOO LARGE FOR ADDRESS SPACE.

Items have been combined by an EQUIVALENCE statement so that the resulting total size of the items equivalenced exceeds 32K words. Correct by reducing the size of arrays or by reorganizing the equivalence relationships.

140 W ATTEMPTED EQUIVALENCE OF MISALIGNED BYTE ITEM.

EQUIVALENCE can be used only with quantities that are aligned on word boundaries. In a LOGICAL*1 (BYTE) array, the elements having odd subscripts are aligned on word boundaries. Correct by referencing only bytes having odd subscripts. For example:

```
                    LOGICAL*1 A,B(9)
     correct:       EQUIVALENCE(A,B(3))
     incorrect:     EQUIVALENCE(A,B(4))
```

141 W ILLEGAL EXPRESSION MODE IN IF STATEMENT.

In a logical IF statement, the expression in parentheses must be of type LOGICAL. In an arithmetic IF statement the expression must be of type BYTE, INTEGER, REAL or DOUBLE PRECISION.

142 W DATA VALUE EXCEEDS BYTE MAGNITUDE.

A value to be assigned to a BYTE variable in a DATA statement is not within the limits -128 to 127.

143 F DELIMITER > MISSING FROM VARIABLE FORMAT EXPRESSION.

The > delimiter is missing or an expression error has occurred in a variable format expression.

144 W EMPTY VARIABLE FORMAT EXPRESSION.

No expression appeared between < and > format expression delimiters.

145 F VARIABLE FORMAT EXPRESSION NOT ALLOWED WITH H FORMAT.

A variable format expression cannot be used as the character count in a Hollerith format specification.

146 W NEGATIVE OR ZERO CONSTANT AS SUBSCRIPT.

Subscripts less than +1 are illegal.

147 F INTERNAL ERROR DURING CODE GENERATION.

An internal consistency check in the code generator for expressions has shown something amiss. This may be caused by other errors also reported for the same statement. If it occurs alone, a Software Performance Report should be submitted to Digital Equipment Corporation.

K.2.2 FORTRAN Compiler Assembly Phase Errors

Three types of diagnostic messages are associated with Compiler assembly phase errors. They are:

P - Phase error: two or more statements have the same statement number.

U - Undefined statement number: reference has been made to a nonexistent statement number.

M - Multiply-defined symbol: reference was made to a multiply-defined statement number. "M" errors are always associated with "P" errors and disappear when the condition causing the "P" error(s) is corrected.

Assembly phase error messages are printed following the list of block names in the block summary. The error code is printed, followed by the entire line of assembly code generated by the erroneous statement. FORTRAN statement numbers are shown preceded by a "." character and may be followed by a ":" character.

A short example program is shown below, along with the listing of error messages and block summary.

```
FORTRAN VØ6.12                    15:Ø1.59   21-JUL-73   PAGE   1

     ØØØ1            GO TO 25
     ØØØ2     3Ø     IF (X-1) 31,32,33
     ØØØ3     32     STOP
     ØØØ4     32     END

              BLOCK       LENGTH
              MAIN.   4Ø      (ØØØ12Ø)*
     U ØØØØ24  ØØØØØØ          .25
     U ØØØØ44  ØØØØØØ          .31
     M ØØØØ46  ØØØØ52'         .32
     U ØØØØ5Ø  ØØØØØØ          .33
     P ØØØØ62  ØØØØØØG.32      $SEQ,ØØØØØ4

             **COMPILER ----- CORE**
               PHASE       USED  FREE
             DECLARATIVES ØØ446 17398
             EXECUTABLES  ØØ446 17398
             ASSEMBLY     ØØ897 19864
```

K.2.3 FORTRAN OTS ERROR DIAGNOSTICS

Each run-time error message issued by the FORTRAN OTS is of the form:

FORTØØCnnn

where "C" is the error class number and "nnn" is the error number within the class.

Version 22 FORTRAN OTS has nine error classes:

Error Class	General Error Area
0	Very severe errors. Impossible to continue execution or to print standard error message and traceback information. The DOS Monitor diagnostic FØ3Ø ØØØnnn is issued for error "nnn" of class Ø.
1	Physical I/O errors: parity, checksum, end-of-file, etc.
2	Format specification syntax error.
3	Arithmetic overflow or division by zero.
4	Argument errors in function or subroutine calls.
5	Arithmetic underflow.

Error Class	General Error Area
6	Conversion error for format-controlled I/O.
7	Subscript errors.
8	Errors resulting from improper linking of OTS routines.

Some diagnostic numbers are not assigned to a specific error condition. The message

FORTØØCnnn SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED

appears for these diagnostic numbers. These diagnostics should never occur during execution of a FORTRAN program.

As discussed in the description of the SETERR subroutine each error class has a maximum occurrence count and a default system interpretation. The user can change the error-handling procedure through SETERR calls.

Class Ø Errors

FORTØØØØØØ INVALID CALL TO ERROR

FORTRAN system error; an illegal argument was passed to the OTS error routine.

FORTØØØØØ1 NO SPACE TO DO I/O

An attempt to open a file failed because there was not enough free memory. The program memory requirements must be reduced.

FORTØØØØØ2 SUBROUTINE DIRECTLY(INDIRECTLY) REFERENCES ITSELF

A recursive subroutine call was attempted. This error may also result if the rules on overlay transfer paths are violated.

FORTØØØØØ3 ILLEGAL FLOATING POINT INSTRUCTION

The PDP-11/45 Floating Point Processor attempted to execute an illegal floating point instruction.

Class 1 Errors

Class 1 errors result from physical I/O errors and illegal operations on files. As described earlier, the END= and ERR= options in I/O statements can be used to transfer control on Class 1 errors.

FORTØØ1ØØØ SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED

This message should never occur.

FORTØØ1ØØ1 DEVICE PARITY ERROR

A parity error was detected during the I/O transfer. Either the file is bad or the I/O device has malfunctioned.

FORTØØ1ØØ2 CHECKSUM/PARITY ERROR - END OF DATA ERROR (RANDOM)

a. A checksum or parity error occurred during unformatted I/O.

b. A random-access I/O statement attempted to read or write beyond the physical end of file.

FORTØØ1ØØ3 UNDIAGNOSABLE I/O ERROR

The DOS Monitor returned an error indication during a read or write operation. The condition was not as described under FORTØØ1ØØ1 or FORTØØ1ØØ2.

FORTØØ1ØØ4 END OF FILE OR END OF MEDIUM

A formatted or unformatted I/O statement has attempted to read or write beyond the physical end of the file. END= may be used to transfer control on this end of file condition.

FORTØØ1ØØ5 UNABLE TO LOCATE CONTIGUOUS FILE

An attempt to allocate a contiguous file as requested by SETFIL or DEFINE FILE has failed because there is not a large enough segment of physically contiguous blocks on the device. Fragmentation of available space may be reduced by using PIP.

FORTØØ1ØØ6 DEFINE FILE NOT DONE (RANDOM ACCESS)

FORTØØ1ØØ7 DEFINE FILE DONE (NOT RANDOM ACCESS)

A DEFINE FILE statement must be entered for each random-access file, but not for any formatted or unformatted files.

FORTØØ1ØØ8 INVALID PROTECTION FOR FILE ACCESS

The protection code for the specified file does not permit access of the type attempted by the program. See the description of the /PR switch in the PIP manual for a discussion of file protection codes.

FORTØØ1ØØ9 FILE DOES NOT EXIST / OR IS ALREADY OPEN

On input, a READ statement attempted to open for input a nonexistent file. On output, a WRITE statement attempted to create and open a linked file; a file of that name already exists in the user file directory.

FORTØØ1Ø1Ø UNABLE TO OPEN FILE

An attempt to open a file failed, but not for any of the more specific reasons described under FORTØØ1ØØ5 through FORTØØ1ØØ9.

FORTØØ1Ø11 WRONG MODE FOR FILE ACCESS

The program attempted to perform I/O on a file that does not support the requested type of I/O; e.g.,

a. Random-access I/O on a DOS linked file,
b. Unformatted I/O on a formatted ASCII file,
c. Formatted I/O on an unformatted file.

FORTØØ1Ø12 INVALID DEVICE NUMBER

The unit number specified in an I/O statement is not a legal FORTRAN logical unit number as specified in the FORTRAN device table.

FORTØØ1Ø13 INVALID RECORD NUMBER (RANDOM ACCESS)

The record number specified in a random-access I/O statement is less than 1 or greater than the maximum specified in the DEFINE FILE statement.

Class 2 Errors

FORTØØ2ØØØ CANNOT DO CONVERSION WITH FORMAT SPECIFIED

Execution of the format specification, including all explicit and implied repetitions, failed to convert all items of the I/O list.

FORTØØ2ØØ1 PARENTHESES NESTING TOO DEEP (>2) IN FORMAT

PDP-11 FORTRAN permits a maximum parentheses nesting depth of 2 within a format specification (not counting the outermost level).

```
legal:   FORMAT (F1Ø.2, 2(I3, F1Ø.2))
legal:   FORMAT (F1Ø.2, 2(I3,3(I3,I2)))
illegal: FORMAT (F1Ø.2, 2(I3,3(I3,4(I2))))
```

FORTØØ2ØØ2 SYNTAX ERROR IN FORMAT

A format specification is constructed incorrectly. PDP-11 FORTRAN does minimal syntax checking of format specifications during compilation.

FORTØØ2ØØ3 REFERENCE OUTSIDE OF RECORD BOUNDARIES

a. An I/O list has requested transmission of more items than exist in an unformatted or direct-access record.

b. A format specification has attempted to read or write more than 133 characters in a single ASCII record.

c. A format specification has attempted to process more characters than specified in the buffer size parameter for ENCODE/DECODE.

Class 3 Errors

All errors in class 3 are arithmetic overflow errors. In each case, the computed result was outside the range of the representation used for that data type on the PDP-11. Arithmetic ranges are specified in Part 7, FORTRAN. Some of these errors (e.g., division by zero) are mathematically undefined operations.

```
FORTØØ3ØØØ SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED
FORTØØ3ØØ1 EXPONENT OVERFLOW IN DOUBLE PRECISION ADDITION
FORTØØ3ØØ2 EXPONENT OVERFLOW IN REAL ADDITION
FORTØØ3ØØ3 DOUBLE PRECISION DIVISION BY ZERO
FORTØØ3ØØ4 EXPONENT OVERFLOW IN DOUBLE PRECISION DIVISION
FORTØØ3ØØ5 INTEGER DIVISION BY ZERO
FORTØØ3ØØ6 EXPONENT OVERFLOW IN REAL DIVISION
FORTØØ3ØØ7 COMPLEX DIVISION BY ZERO
FORTØØ3ØØ8 REAL DIVISION BY ZERO
FORTØØ3ØØ9 SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED
FORTØØ3Ø1Ø EXPONENT OVERFLOW IN DOUBLE PRECISION MULT.
FORTØØ3Ø11 INTEGER OVERFLOW DURING NEGATION
FORTØØ3Ø12 EXPONENT OVERFLOW IN REAL MULTIPLICATION
FORTØØ3Ø13 INTEGER OVERFLOW ON ADDITION OR SUBTRACTION
FORTØØ3Ø14 PRODUCT OUTSIDE OF RANGE ON INTEGER MULT.
FORTØØ3Ø15 INTEGER BASE = Ø, INTEGER EXPONENT < = Ø
FORTØØ3Ø16 DOUBLE BASE = Ø, INTEGER EXPONENT < = Ø
FORTØØ3Ø17 DOUBLE BASE = Ø, DOUBLE EXPONENT < = Ø
FORTØØ3Ø18 DOUBLE BASE < Ø, DOUBLE EXPONENT < = Ø
FORTØØ3Ø19 REAL BASE = Ø, REAL EXPONENT < = Ø
FORTØØ3Ø2Ø REAL BASE < Ø, REAL EXPONENT < = Ø
FORTØØ3Ø21 REAL BASE = Ø, INTEGER EXPONENT < = Ø
FORTØØ3Ø22 REAL OUTSIDE RANGE ON REAL TO INTEGER CONVERSION
FORTØØ3Ø23 EXPONENT OVERFLOW ON DOUBLE TO REAL CONVERSION
FORTØØ3Ø24 FLOATING POINT EXPONENT OVERFLOW
FORTØØ3Ø25 FLOATING POINT DIVISION BY ZERO
FORTØØ3Ø26 INTEGER OVERFLOW ON DO-LOOP VARIABLE
FORT003Ø27 COMPLEX BASE = Ø, INTEGER EXPONENT < = Ø
```

Class 4 Errors

Class 4 errors are issued by library subroutines and functions. Each message identifies the routine issuing the message and the cause. See Part 7 for descriptions of these routines.

```
FORTØØ4ØØØ SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED
FORTØØ4ØØ1 SYSTEM ERROR NO DIAGNOSTIC MESSAGE ASSIGNED
FORTØØ4ØØ2 DEXP CALLED WITH EXPONENT GREATER THAN 88
FORTØØ4ØØ3 DLOG ARGUMENT LESS THAN OR EQUAL TO ZERO
FORTØØ4ØØ4 DSQRT ARGUMENT LESS THAN ZERO
FORTØØ4ØØ5 EXP CALLED WITH EXPONENT GREATER THAN 88
FORTØØ4ØØ6 ARGUMENTS OUT OF RANGE FOR "TIME" CONVERSION
FORTØØ4ØØ7 IABS ABS(X) GREATER THAN 2**14 - 1
FORTØØ4ØØ8 IDIM RESULT OUTSIDE OF RANGE -2**15-1 TO 2**15-1
FORTØØ4ØØ9 ISIGN RESULT GREATER THAN 2**15 - 1
FORTØØ4Ø1Ø ALOG ARGUMENT LESS THAN OR EQUAL TO ZERO
FORT004Ø11 SQRT ARGUMENT LESS THAN ZERO
FORTØØ4Ø12 SNGL EXPONENT OVERFLOW ON ROUND
FORTØØ4Ø13 RANDU/RAN WRONG NUMBER OF ARGUMENTS
FORTØØ4Ø14 PDUMP WRONG NUMBER OF ARGUMENTS
```

FORTØØ4Ø15 INVALID ARGUMENT TO ASSIGN OR SETFIL

Issued by SETFIL or ASSIGN for any of the following reasons:

a. The unit number is less than 1 or greater than the maximum unit number permitted,

b. The name string is not a syntactically valid device/filename, or

c. The name string is too long.

FORTØØ4Ø16 FILE ALREADY OPEN ON UNIT - ASSIGN OR SETFIL IGNORED

The file on the logical unit specified has been opened and not closed. END FILE can be used to close a file and make the logical unit available for other uses.

Class 5 Errors

All errors in this class are arithmetic exponent underflow errors. An attempt was made to compute a non-zero number smaller than the smallest representable number on the PDP-11. The result is set to Ø.Ø.

FORTØØ5ØØØ CSQRT UNDERFLOW
FORTØØ5ØØ1 EXPONENT UNDERFLOW ON DOUBLE PRECISION ADDITION
FORTØØ5ØØ2 EXPONENT UNDERFLOW ON REAL ADDITION
FORTØØ5ØØ3 EXPONENT UNDERFLOW ON REAL DIVISION
FORTØØ5ØØ4 DEXP CALLED WITH EXPONENT LESS THAN -89.4
FORTØØ5ØØ5 EXP CALLED WITH EXPONENT LESS THAN -89.4
FORTØØ5ØØ6 EXPONENT UNDERFLOW ON DOUBLE MULTIPLICATION
FORTØØ5ØØ7 EXPONENT UNDERFLOW ON REAL MULTIPLICATION
FORTØØ5ØØ8 EXPONENT UNDERFLOW ON DOUBLE PRECISION DIVISION
FORTØØ5ØØ9 FLOATING POINT EXPONENT UNDERFLOW

Class 6 Errors

FORTØØ6ØØØ CONVERSION ERROR

An error occurred during a format conversion.

On output conversions, the error may result from:

a. A value too large to fit in specified field, or
b. An L conversion done on a value that is neither .TRUE. nor .FALSE.

The output field is filled with asterisks (*)

On input conversions, the error may result from:

a. A value too large to be represented in specified PDP-11 data type, or
b. Illegal characters in the input field.

Class 7 Errors

FORTØØ7ØØØ SUBSCRIPT LESS THAN OR EQUAL TO ZERO

An array subscript is < Ø. This is never legal in FORTRAN IV. A lower limit check is made on every subscript reference in PDP-11 FORTRAN.

FORTØØ7ØØ1 SUBSCRIPT GREATER THAN DIMENSIONED

An array subscript is greater than the maximum value specified in the DIMENSION statement for the array. The upper limit check is optional and is made only if the /CK switch is set prior to compilation of the program or subroutine.

FORTØØ7ØØ2 VALUE OUT OF BOUNDS ON ASSIGNED GOTO

This error results when the value of the variable is not among those in the optional label list for an assigned GOTO. The GOTO is not performed and execution continues at the next statement in the FORTRAN program.

FORTØØ7ØØ3 FLOATING POINT UNDEFINED VARIABLE

A PDP-11/45 floating-point undefined variable interrupt has occurred. PDP-11 FORTRAN does not enable this interrupt. This message should not occur.

Class 8 Errors

The FORTRAN OTS I/O packages contain a number of interdependent modules. If a FORTRAN program does not use certain I/O routines (e.g., unformatted I/O), short dummy routines are substituted for those unused routines during the linking process. A Class 8 message is issued, and the program run terminated, if a call is made to one of these dummy routines.

Class 8 errors ordinarily result from improper structuring of an overlay system or the use of a format specification stored in an array or input at execution time. Forcing the loading of the required I/O and conversion routines will correct the problem.

FORTØØ8ØØØ LINKAGE ERROR (MISSING FORMAT CONVERSION ROUTINE)

A numeric format conversion routine, required by the program, was not included. Including a dummy FORMAT statement in the resident section of the program will force loading of the necessary conversion routines.

FORTØØ8ØØ1 LINKAGE ERROR (UNFORMATTED I/O)

An overlay attempted to do unformatted I/O, but the required control routines were not included in the resident section.

FORTØØ8ØØ2 LINKAGE ERROR (DIRECT-ACCESS I/O)

An overlay attempted to do direct-access I/O, but the required control routines were not included in the resident section.

K.3 EDIT ERROR MESSAGES

Error Code	Meaning	Cause
S2Ø2	Device Full	Output device does not have sufficient room to continue.
S2Ø3	Switch Error	Too many switches or illegal switch. A switch appeared which was not equal to /B or which followed a file other than primary input; or more than one switch appeared.
S2Ø4	Too Many Output Files	More than two output files were specified or a switch appeared after secondary output.
S2Ø5	Too Many Input Files	More than two input files were specified or a switch followed the secondary input.
S227	Illegal File Specification	Argument indicates specific violation. 1 - No primary output specified 2 - Secondary input equals secondary output 3 - Secondary input equals primary output 4 - Primary input equals secondary output 5 - Primary input equals secondary input 6 - Primary output equals secondary output
Command Syntax Errors -	Command syntax errors are reported by printing the command up to and including the character at which the scan terminated, followed by a question mark and vertical tab. This does not mean the last character typed is the cause of the syntax error.	
W3Ø3	Buffer Overflow	Command Input Buffer, Text Input Buffer, Save Buffer or Page Buffer overflow.
W3Ø4	Macro Overflow	Macro as stored in Save Buffer is too long to execute.
W3Ø5	Recursive Macro	Macro contains an EM command.
W3Ø6	Empty Save Buffer	An EM or U command was issued with nothing in the Save Buffer.
W3Ø7	Search Failure	Search object was not found in available text.
W31Ø	No Room to Unsave	Not enough available room to unsave required text.
W311	End of Data	End of input medium or end of input file reached during read. Last page read was last in file.

Error Code	Meaning	Cause
W312	Illegal Line Feed	A line feed character was encountered in command string.
W313	Illegal Negative Argument	The command specified does not accept negative arguments.
W314	No Arguments Allowed	The command specified does not recognize any arguments.
W315	Illegal Argument	Command does not accept given argument.
W316	Illegal Text String	Usually caused by missing second delimiter.
W317	Illegal Command	EDIT cannot execute command as requested. Usually caused by secondary I/O commands when no secondary I/O was specified at initialization time.
W32Ø	Page Buffer Almost Full	Page Buffer within 128 characters of being full.
W321	File Closed	Attempt to Read or Write primary files after EF.

K.4 LINK Error Diagnostics

The following error diagnostics are issued by LINK:

ALLOCATION FAILURE ON FILE filnam.ext

There was not sufficient space on the disk to allocate the output file (filnam.ext) contiguously.

BLANK CONTROL SECTION NAME IS NOT LEGAL

A .PSECT command has specified a blank name. This is illegal; all .PSECT commands, when used, must specify non-blank names.

COMMAND SYNTAX ERROR

The command string last issued to LINK was not a valid command. It must be re-entered correctly.

CTRL SECTION secnam HAS OVERFLOWED

The control section secnam has overflowed machine address boundaries. No segment can exceed 32K words.

FILE filnam.ext HAS ILLEGAL FORMAT

The file filnam.ext does not have the correct format for LINK.

ILLEGAL /B OR /T SWITCH VALUE value

The /B or /T switch value specified (value) was not a 1- to 6-digit octal constant.

ILLEGAL /CO VALUE PARAMETER value

The /CO switch value specified (value) was not a decimal number that is an integral multiple of 64.

ILLEGAL CONTROL SECTION ATTRIBUTE

An unrecognizable .PSECT attribute has been encountered.

ILLEGAL MULTIPLE PARAMETER SETS

The specified option allows only one parameter set; more have been specified.

ILLEGAL OVERLAY DESCRIPTION OPERATOR

An illegal ODL operator has been encountered.

ILLEGAL OVERLAY DIRECTIVE

An unrecognizable overlay directive has been encountered.

ILLEGAL SWITCH filnam.ext

The switch(es) specified for the file filnam.ext cannot be recognized or processed correctly.

ILLEGAL /TR SWITCH VALUE value

The /TR switch value specified (value) was one of the following: (1) an odd number, (2) an undefined symbol, or (3) an out-of-range symbol.

INDIRECT COMMAND SYNTAX ERROR

An indirect command line has been specified incorrectly. Probably, no file name has been specified following the @ character.

INDIRECT FILE DEPTH EXCEEDED

An attempt has been made to nest more than five indirect files.

INDIRECT FILE OPEN FAILURE

An indirect file that has been specified cannot be found.

INSUFFICIENT PARAMETERS

Not enough parameters have been supplied for the option specified.

INVALID KEYWORD IDENTIFIER keynam

The name keynam has been specified and is not a legal options keyword. (See Chapter 9-8 for legal options.)

I-O ERROR ON OUTPUT FILE filnam.ext

An unrecoverable output error has occurred on the file filnam.ext.

I/O FAILURE ON INPUT FILE filnam.ext

LINK cannot correctly read data from the file filnam.ext.

LABEL OR NAME IS MULTIPLY DEFINED

A label or name has been defined more than once in an overlay description. This is illegal; labels and names must be uniquely defined.

LOAD ADDR OUT OF RANGE IN MODULE modnam

An address has been specified within a segment of the module modnam that does not fall within the range specified for the segment.

MISSING /B OR /T SWITCH VALUE

A /B or /T switch has been specified without a value. The /B and /T switches, when specified, must have an associated value.

MODULE modnam AMBIGUOUSLY DEFINES CTRL SECT secnam

LINK has found two or more P-section descriptions in the same segment whose attributes are not identical.

MODULE modnam AMBIGUOUSLY DEFINES SYMBOL symnam

The module modnam has defined a reference (symnam) that has been previously defined. Such a reference cannot be uniquely resolved.

MODULE modnam ILLEGALLY DEFINES XFR ADDRESS transf

A transfer address (transf) is incorrectly defined in a module (modnam). Possibly, transf was specified within an overlay segment.

MODULE modnam MULTIPLY DEFINES CTRL SECT secnam

The P-section (secnam) described in a module (modnam) is not the original definition.

MODULE modnam MULTIPLY DEFINES SYMBOL symnam

Two definitions for the same symbol (symnam) have occurred on the same path within a module (modnam).

NO DYNAMIC STORAGE AVAILABLE

LINK has no more memory available to complete a link. The link can be re-executed only if the memory requirement for linking is reduced.

NO ROOT SEGMENT SPECIFIED

An overlaid program does not have a root segment specified. This is illegal; overlaid programs must specify a root segment.

OPEN FAILURE ON FILE filnam.ext

LINK cannot find a specified file filnam.ext.

OPTION SYNTAX ERROR

The format of an option command is incorrect.

OVERLAY DIRECTIVE HAS NO OPERANDS

An overlay directive has been supplied without operands. The only directive that allows no operands is the .END directive.

OVERLAY DIRECTIVE SYNTAX ERROR

An overlay directive has been specified in an incorrect format.

PASS CTRL STACK OVERFLOW ON SEGMENT segnam

Too many overlay levels have been specified. Overlays must not be nested to a depth greater than 16 levels.

PREMATURE EOF COMMAND INPUT FILE

An end-of-file condition was encountered when LINK was expecting additional command input.

REQUIRED INPUT FILE MISSING

At least one input file must be specified to LINK.

ROOT SEGMENT IS MULTIPLY DEFINED

One .ROOT command (and no more than one) must be specified per program. This program has defined more than one .ROOT command.

SEARCH STACK OVERFLOW ON SEGMENT segnam

Too many overlay levels have been specified. Overlays must not be nested to a depth greater than 16 levels.

SEG segnam HAS ADDR OVERFLOW-ALLOCATION DELETED

The program has attempted to allocate more than 32K words within an overlay segment (segnam). This results in deletion of the program image file; a map is produced, but the program image file is not.

SEGMENT segnam HAS RO CONTROL SECTION

Overlay segment segnam contains an RO control section. RO control sections can be specified for root segments only.

TOO MANY NESTED .ROOT-.FCTR DIRECTIVES

An attempt has been made to nest .FCTR statements to a depth greater than 32 levels.

TOO MANY PARAMETERS

Too many parameters have been specified with an options keyword.

TOO MANY PARENTHESIS LEVELS

An attempt has been made to nest parentheses to a depth greater than 32 levels in an overlay description.

TRUNCATION ERROR IN MODULE modnam

A byte value specified as relocatable in the module modnam exceeded 8 bits after relocation bias was added. The low-order eight bits are loaded into the byte.

UNBALANCED PARENTHESIS

An overlay description contains mismatched parentheses (e.g., an odd number of parentheses).

cnt UNDEFINED SYMBOLS

Undefined symbols have been encountered during a link. The value cnt specifies the number of undefined symbols.

"@" COMMAND FILE SYNTAX ERROR

An indirect command file has been incorrectly specified. The string following the @ character was not recognized.

K.5 LIBR ERROR MESSAGES

Error Code	Additional Information	Meaning
S2Ø2	File Name and Error Status Byte	Fatal I/O error; due to truncated line, checksum, character parity, or device parity error.
S2Ø3	File Name	Switch error or semantic error; due to illegal switch, too many switches on a file, or illegal combination of file specifications.
S2Ø4		Illegal file specification format; more than two output files specified.
S213	File Name	Error on input file; illegal object module format; first line not a GSD, or EOF prior to reading end module line.
S244	File Name	Out of order; already past requested position for Insert.
S245	File Name	Object module error; object module not found, or /R or /D out of order.
S246		Error on input library; illegal library format, first two lines incorrect.
S247		Listing error; output library cannot be read from output library device, i.e., PP:.

K.6 PIP ERROR MESSAGES

As a system program under the DOS/BATCH Operating system, error messages received when using PIP conform to the standards for error handling. PIP's error messages are of the form:

```
        Sxxx     n
```

where Sxxx indicates a system program error number and n is one octal word displaying additional information. Error messages that PIP may issue are listed below.

ERROR CODE	ADDITIONAL INFORMATION	MEANING
I1ØØ	Hardware Status Register Content.	Single verification failure on cassette tape.

ERROR CODE	ADDITIONAL INFORMATION	MEANING
I354	Ø	Illegal response to CONFIRM; when attempting to zero on RK11 disk cartridge. Legal responses are: H - for high density disks (RKØ3/Ø5) L - for low density disk (RKØ2) blank - to cancel the request
FØ76	Device (RAD5Ø)	Cassette tape encountered enough read-after-write failures to exhaust the retry count.
S2Ø2	Error Status Byte	End of Data (EOD) or device error on .WRITE or .READ.
S2Ø3	Ø	Illegal switch, or too many switches, or illegal switch value, switch value not given or illegal switch in output field.
S2Ø4	Ø	Too many or too few output files.
S2Ø5	Ø	Too many or too few input files.
S2Ø7	Error Status Byte	EOD or device error on .TRAN.
S231		Illegal command, file-structured device required.
S232		More than one action switch (only one permitted).
S233		Specified UIC not found in MFD.
S234		Null filename or * given where filename required.
S235		No files found in UFD.
S236		Operation applicable to DECtape only.
S237		File not found during file recovery operation.
S24Ø		No space for file allocate.
S241		MFD is full.
S242		Meaningless command (no action taken).
S252	Ø	Filename given when none allowed.
S257	File Block Error Code dev:file,ext.	Illegal file operation. For example, protect code does not allow transfer of file; UIC different from login UIC thus making certain "wild card" operations illegal. The operation in question is not performed.

ERROR CODE	ADDITIONAL INFORMATION	MEANING
S26Ø	Ø	Same device needed for input and output in fast copy operation.
S262	Ø	Record size too big for buffer.
S263	File Number	File record sizes do not agree on verify (/VE).
S264	Ø	Conflict in standard filename extension which determines mode of transfer. Use explicit switches to resolve.
S265	Ø	Operation attempted on device which is not legal for non-privileged user.
WØ51		Timing errors have exhausted the retry count.

K.7 FILCOM ERROR MESSAGES

The FILCOM program issues three types of error messages to identify the source of errors during execution:

1. Command Syntax Errors
2. I/O Device Initialization Errors and I/O Errors
3. Runtime Errors

K.7.1 Command Syntax Errors

Message	Meaning
TOO MANY GLOBAL SWITCHES	More than one GLOBAL switch has been specified in a command string. The command must be retyped with only one GLOBAL switch per string.
UNKNOWN OPTION	FILCOM does not recognize a switch name in a command string; possible mistyping of switch. Command must be retyped using correct switch designation.
BAD NUMERIC FIELD	An invalid number has been typed as the argument of a switch (e.g., the SC switch). Command must be retyped using valid number for the switch.
SYNTAX ERROR IN COMMAND	A typed command does not conform to rules of the CSI; command must be retyped to conform to CSI rules.

Message	Meaning
TOO MANY SWITCHES	The number of switches in a command exceeds the capacity of the switch buffer. Commands must be retyped using fewer switches per string.
WRONG # OF INPUT DEVICES	A command has been entered containing more (or less) than two input dataset specifications. Two input datasets (master and newmaster) must be specified for a comparison. Command must be retyped correctly.

K.7.2 I/O Device Initialization Errors and I/O Errors

The following message formats indicate an I/O error:

```
xxxx DEVICE INIT ERROR

xxxx DEVICE OPEN ERROR

xxxx IO ERROR
```

where xxxx indicates a file in error, and can be MASTER, NEWMAS, LIST, or COMMAND (indirect command file). These messages are issued if one or more of the following conditions exist:

1. An attempt is made to initialize a nonexistent device.
2. Output is attempted to an input-only device.
3. Input is attempted from an output-only device.
4. An output device (list or log dataset) is duplicated in two commands.

K.7.3 Runtime Errors

Message	Meaning
COMPARE CAPACITY EXCEEDED	There is not enough core available to compare two specified files.
RAN OUT OF BUFFER HEADERS	Program error in FILCOM. Kill program and re-run. If error message is issued again, report via SPR.*

*Software Performance Report; submit to Digital Equipment Corporation, Software Information Services, Maynard, Massachusetts 01754.

K.8 VERIFY Error Messages

Message — Meaning

```
BLOCK NUMBER OUT OF RANGE = n
```

This message can occur during execution of any VERIFY option. It is produced by the subroutine MASK during bit map reconstruction. It indicates that an octal block number (n) is too large for the array MAP, which is allocated by VERIFY to hold the reconstructed bit maps. One of two conditions is possible:

(1) The named block is garbage in the file, and is a valid error.

(2) VERIFY is not capable of handling a disk being verified. VERIFY typically has enough space in MAP to verify an RK disk, RP disk, or 4-platter RF disk.

```
DEVICE ERROR DURING TRAN
BLOCK NUMBER   = n
FUNCTION WORD  = f
```

A hardware error has been encountered during a TRAN operation. The octal number of the block being read (n) and the octal number of the TRAN function word (f) are listed. See the DOS/BATCH Monitor Programmer's Manual for a complete description of the TRAN function word.

```
FILE SIZE IS INCORRECT
filnam.ext[uic]
SYSTEM THINKS FILE SIZE = s
ACTUAL FILE SIZE = a
```

The file (filnam.ext) with UIC (uic) has an incorrect file length. The number of blocks actually used by the file (a) does not agree with the size (s) stored in the file directory.

```
FILE END BLOCK IS INCORRECT
filnam.ext[uic]
SYSTEM THINKS END BLOCK = e
ACTUAL END BLOCK = a
```

The file (filnam.ext) with UIC (uic) has incorrect directory information. The last block used by the file (a) is different from the last block (e) indicated by the file directory.

```
MAP VERIFICATION ERROR
MAP NUMBER = n, WORD NUMBER = w
RECONSTRUCTED ENTRY = rrrrrr
SYSTEM ENTRY = ssssss
{LOST}
{    } BLOCK NUMBER - d DECIMAL (o OCTAL)
{FREE}

TOTAL FREE BLOCKS = f TOTAL LOST = t
```

This message is printed during map verification when the bit map on the device does not agree with the reconstructed bit map produced by VERIFY. The map number (n) and word number (w) are provided. The reconstructed bit map entry (rrrrrr) and system bit map entry (ssssss) are followed by a list of the block numbers (both decimal and octal) of the blocks in error. A block in error is marked as FREE or LOST. A FREE block is one marked as unused (bit = Ø) in the system map, but is used (bit = 1) according to the reconstructed map. A LOST block is one marked as used (bit =1) in the system map, but is unused (bit =Ø) according to the reconstructed map. LOST blocks cannot be used by any file, but do not endanger file integrity on the device. If the device being verified is

Message

Meaning

a disk, the FIX option of VERIFY can be run to recover LOST blocks. However, a file containing one or more FREE blocks can be damaged by other files created later on the device. Files containing FREE blocks should be copied (to another device) using the PIP program, and then deleted from the original device. To determine the names of the files containing FREE blocks, run VERIFY in search mode (SEARCH or ALL option), specifying the block(s) in error. After all map verification error messages have been listed at the printer.

LOCKED FILE
filnam.ext[uic]

The file (filnam.ext) with UIC (uic) was locked, and therefore in an error state, when VERIFY attempted to perform file verification on it. The UNLOCK switch (/UN) in the PIP program can be used to unlock this file; once the lock bit has been turned off, the file can be renamed or deleted.

OPEN FILE, USAGE COUNT = n
filnam.ext[uic]

The file (filnam.ext) with UIC (uic) has a nonzero USAGE count. This indicates that the file is in an intermediate state, and is likely to be the cause of bit map errors. The file's USAGE count can be zeroed through use of the UNLOCK switch (/UN) in the PIP program.

MULTIPLE ALLOCATION OF BLOCK n
filnam.ext[uic]

The file (filnam.ext) with UIC (uic) contains octal block n, which is also contained in another file on the same device. This indicates that two or more files are "cross-allocated," i.e., they intersect at block n. This situation is dangerous, since deletion of one of those files would cause part of the other(s) to be deleted also. For linked files, the data contained in such cross-allocated blocks is data for the file created most recently. It may be possible to recover all or part of such crossed files by transferring them individually to other devices through the PIP program. When two or more files are found to intersect, none should be deleted until the desired files have been backed up by PIP transfers.

This message is not sufficient by itself to identify all intersecting files for block n. After this error message has been issued, VERIFY should be run in search mode (SEARCH or ALL option) to identify all files that intersect at block n.

K.9 FILDMP ERROR MESSAGES

The following error messages are used by FILDMP.

Message	Most Probable Cause
S2Ø2	An error occurred during reading of the command input. (Recall that the maximum line length is 72 decimal characters.)
S2Ø3	An error occurred in the switches. Either: 1) FILDMP could not understand the switch; 2) too many switches on input or output; 3) no value or more than two values to /BL:
S2Ø5	More than one input file specification in the command string. This error will appear even if the extraneous input file specifications are null.
S2Ø6	FILDMP could not find the input file to /CH.
S256	A /CH request accompanied a file specification in which the input device is not directory structured, or input device will not support input.

Other error messages can occur by virtue of the user having requested FILDMP to do something illegal. For example, DOS/BATCH will issue an FØ12 message if FILDMP attempts to read a file which is protected so that the current user cannot access it. The user should consult the appropriate DOS/BATCH documentation upon receiving such error messages.

FILDMP does not terminate processing (or inform the user via S2Ø2) if a read error occurs while-the input file is being read. The user should search the dump for E flags, and then consult the status byte. Recall that the E flag appears physically between the line number and the status byte number on dumps of files which were read in either formatted ASCII or formatted binary mode.

Errors detected during the command string input and during the dumping process will result in standard DOS/BATCH Monitor error message printout. See Appendix K.1 for a complete list and explanation of the error codes.

K.10 ROLLIN ERROR MESSAGES

Error messages printed by the ROLLIN program are succinct and require no further explanation.

```
SYNTAX ERROR, COMMAND IGNORED.
DISK ERROR,--REQUEST KILLED.
```

```
THE REEL LABEL INDICATES THAT THE REST OF THE TAPE WAS NOT DUMPED, TYPE K
TO KILL REQUEST AT THIS POINT, ANYTHING ELSE TO PROCEED IN THE FACE OF DANGER:

LABEL INDICATES THAT THE TAPE IS OUT OF SEQUENCE.  TYPE P TO PROCEED, M TO
MOUNT ANOTHER REEL, OR K TO KILL REQUEST:

PREMATURE END-OF-FILE, REQUEST KILLED.

TAPE FULL, TYPE M TO MOUNT ANOTHER REEL AND CONTINUE.  ANYTHING ELSE TO
ABORT REQUEST:

SELECT ERROR ON MTn:

MAGTAPE WRITE PROTECT ERROR.

FATAL MAGTAPE ERROR

SPECIFIED DEVICE DOES NOT EXIST.

REACHED END-OF-DATA ON SKIP, OPERATION KILLED.

HUNG DEVICE DTn
TYPE K TO ABORT, ANYTHING ELSE TO TRY AGAIN:

MOUNT TAPE ON DTn, TYPE RETURN TO CONTINUE WHEN READY.

TOO FEW DECTAPE UNITS WERE SPECIFIED.  REQUEST KILLED.

DISK ERROR ON UNIT n -- REQUEST KILLED.

END OF FILE DURING READ, TYPE M TO MOUNT ANOTHER REEL, OR K TO KILL REQUEST:

MAGTAPE FILENAME DOES NOT MATCH SPECIFIED NAME.

CAN'T FIND SPECIFIED FILE ON TAPE

NO OUTPUT FILENAME SPECIFIED.

MAGTAPE RECORD TOO LONG FOR BUFFER

VERIFICATION ERROR-COPY IS BAD

VERIFY IS NOT IMPLEMENTED FOR THIS COMMAND

ERROR DURING FORMAT PASS - RESTART

DISK NOT READY - TYPE CR TO TRY FORMAT AGAIN
```

K.11 DSKINT ERROR MESSAGES

Message	Meaning
INPUT COMMAND STRING ERROR. TRY AGAIN	The operator has entered an input command string incorrectly. The command must be re-entered.
INITIALIZES RP DISKS ONLY. TRY AGAIN	The operator specified a device other than an RP disk in an input command string. The command must be re-entered.
INVALID UNIT NO. TRY AGAIN	The operator specified a unit number greater than 7 for a disk in the input command string.
NOT ENOUGH MEMORY	Not enough memory is available to contain BADB.SYS. No remedial action is possible.
BAD BLOCK ADDRESS OR SWITCH ERROR	Operator error when entering a command string in Mark mode. Possible reasons are:

Message	Meaning
	(1) invalid switch specified, (2) block address too high, (3) cylinder:track:sector address too high, (4) operator attempted to add block Ø or block 1 to BADB.SYS. These blocks are reserved for the system.
CRITICAL ERROR IN MFD OR BIT MAP DISK PACK UNSUITABLE FOR USE UNDER DOS/BATCH	An error has been detected on disk blocks Ø or 1, or verification of the test patterns has failed. The disk being initialized cannot be used under DOS/BATCH. After issuing this message, DSKINT exits to the Monitor without performing further initialization.
S254 Ø	The operator has issued a command string to DSKINT that would result in the zeroing of the system disk. This is illegal.
AØ43 bbbbbb	This message is typed each time a bad block is detected during initialization. The address of the bad block is listed (bbbbbb). To continue initialization, the operator types CO at the keyboard. This message may be issued several times for the same block, as DSKINT retries the block before entering it into BADB.SYS.
FØ42 eeeeee	This message is issued when the disk controller is unable to perform a successful home seek. Hardware failure or a damaged disk pack is indicated. The contents of the controller error register (eeeeee) are listed. DO NOT ATTEMPT TO USE THE DISK FURTHER UNTIL IT HAS BEEN VERIFIED BY FIELD SERVICE. SAVE THE DSKINT PRINTOUTS AND CHECK THE DISK DRIVE FOR A "FILE UNSAFE" LIGHT.

K.12 DOS/BATCH ERROR MESSAGES

Following is a complete summary of all error messages which can appear when using the DOS Monitor and system programs.

K.12.1 Action Message

Action messages are printed and the program is suspended. The Monitor expects the operator to take some action such as "continue the program" (type CONTINUE), or "kill the program" (type KILL).

CODE/ISSUER	ADDITIONAL INFORMATION/MEANING
AØØ1 Monitor	User Call Address Disk address error.
AØØ2 Monitor	Device (RAD5Ø) Device not ready. For example, the desired device/unit may be off-line or it may not be write-enabled. For DECtape or magtape, the proper unit may not have been selected. Make the device ready and type CO.
AØØ3 Monitor	Link Block Address The Link Block contains either an illegal device code or no device code at all. Use the MODIFY command to display the contents of Link Block+2, which is the dataset name (RAD5Ø), and then use the ASSIGN command to assign a device and/or file; type CO when ready. An attempt was made to associate (INIT) a second link block to a device driver that does not support multiple users.
AØØ4 Monitor	User Call Address DECtape error. Try adjusting the tape; type CO to retry the operation.
AØØ5 OTS	Pause Number A PAUSE was encountered in a FORTRAN program. Type CO to continue.
AØØ6 LINK	Correct Module Name Paper tape loaded out of order on Pass 2 of Linker. Load correct module and type CO to continue.
AØØ7 Monitor	Call Address The name of the output file being created on magtape is the same as that of an existing file. Type CO to write over the old file or mount another tape and then type CO.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
AØ1Ø	Monitor	Unit Number A parity error occurred when trying to open a file on magtape or on a cassette. Type CO to continue searching. If the file being sought has a parity error in its label, it cannot be found.
AØ11	Monitor	Ø if date is bad, 1 if time is bad System date or time is not valid. Reenter date or time via the console keyboard and type CO to continue.
AØ12	Monitor	Status Register Magtape error. After having made 15 entries on a WRITE or WRITE EOF, the operation is still unsuccessful. Type CO to ignore the error and proceed, or type KI to stop the program and start over with a good tape.
AØ43	PIP	Disk Pack Block Number This is the block that is bad; issued by the RP11 pack initializer to provide a list of bad blocks and to permit job termination if too many are bad. Type CO if number of bad blocks thus far is tolerable.
AØ5Ø	Monitor	Ø Batch Stream Wait. Type CO to continue.
A1ØØ		nnnnnn (Hardware Status Information) The cassette tape unit identified has encountered the physical end of tape while performing the indicated operation. The status word indicates the selected unit number in bits 1 - 3. A code of 2 indicates write and a 4 indicates read. The operator should mount a scratch cassette for subsequent writing, mount the next volume for reading, or otherwise respond to the keyboard request for input.
A35Ø	Monitor	Ø Power has come up following a power failure. Any I/O in progress has been lost, but information in core and in the registers has been retained. If you wish to continue, type CO. Note, however, that if I/O was in progress, the driver(s) may have been left in a state which will not permit your program to be continued.
A36Ø	DECCOM	The System Interface Block (SIB) or the User Interface Block (UIB) overflowed.
A37Ø		The output stacker on the card reader is full. Empty the stacker and continue. Do not reread the last card.

K.12.2 Error Messages

Error messages are printed on the teleprinter in the following format.

```
cnnn xxxxxx
```

where c is one of five letters identifying the type of message:

I Information
A Action required by the operator
W Warning to the operator
F Fatal error
S System program error

nnn is the message number; and xxxxxx gives appropriate additional information. Information, Warning, and System program messages are printed and the program continues.

Action messages are printed and the program is suspended. The Monitor expects the operator to take some action such as "continue the program" (type CONTINUE), or "kill the program" (type KILL).

Fatal error messages are printed if possible, and the program is suspended. The Monitor will not allow the operator to CONTINUE the program, but expects to see either a BEGIN, RESTART or KILL command. If a fatal error is a system disk failure and the error message cannot be printed, the central processor halts. This is the only time that a halt occurs in the Monitor. If the error has been caused by a stack overflow, the stack pointer is reset before the message is printed.

K.12.3 Fatal Messages

Fatal error messages are printed, if possible, and the program is suspended. The Monitor will not allow the operator to continue the program, but eventually expects to see a BEGIN, RESTART or KILL command. If a fatal error is a system disk failure and the error message cannot be printed, the central processor halts. This is the only time that a halt occurs in the Monitor.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
FØØØ	Monitor	Request Address Dataset not INITed. Program must issue .INIT before any other requests to a dataset.
FØØ1	Monitor	Request Address Stack overflow. Once loaded, a program requires additional space for its stack, buffers and control blocks. These are allocated as they are needed. Reduce the size of the program.

CODE/ISSUER — ADDITIONAL INFORMATION/MEANING

FØØ2 — Request Address
Monitor — Invalid EMT call. The EMT code issued by the program has not been assigned.

FØØ3 — Request Address
Monitor — Invalid .TRAN function or .TRAN to an open file.

FØØ4 — Error Code
Monitor — Incorrect OPEN on industry compatible magnetic tape. Caused by program error or improperly assigning devices via datasets. Defined error code values:
Ø - another file currently opened on tape,
1 - attempt to READ or WRITE to unopened file.

FØØ5 — Request Address
Monitor — .RLSE error. If a file has been OPENed, it must be CLOSEd before a .RLSE can be issued.

FØØ6 — Request Address
Monitor — Device full. No more space exists on the device being referenced by the request. For a file-structured device, use PIP to look at the number of free blocks and delete any files which are not needed.

FØØ7 — Request Address
Monitor — No buffer space available. Insufficient space for completion of required operation. Reduce program size or close open files.

FØ1Ø — Request Address
Monitor — Illegal .READ/.WRITE. Incorrect mode for device or file not opened correctly.

FØ11 — Request Address
Monitor — Illegal OPEN. OPEN code is not used or is unsuitable for device.

FØ12 — Request Address
Monitor — File access violation. You are trying to OPEN a file that cannot be opened for the requested purpose. See Table K-1 for details. Assure that the name of the file requested was correct.

FØ14 — Request Address
Monitor — Device error on trying to read bit map. The system cannot proceed if it cannot read the bit map. New files cannot be created on the device nor can old files be extended. Existing files may be copied to a backup medium for recovery.

FØ15 — Request Address
Monitor — DECtape error. Nonexistent memory addressed or end-zone reached during transfer.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
FØ16	Monitor	Block Number DECtape search failure. Block requested cannot be found.
FØ17	Monitor	Device (RAD5Ø) Parity error on file-structured device.
FØ2Ø	Monitor	Irrelevant Too many datasets using low-speed paper tape. A maximum of one each for input or output is allowed. Restart your job and use the ASSIGN command to reassign the excess datasets.
FØ21	Monitor	Irrelevant Checksum error or device parity error while typing to load a program. Type KILL then try again. If that doesn't work, try relinking the program. Try recreating the file. If the error persists, hardware may be faulty. Call field service.
FØ22	Monitor	Irrelevant An attempt was made to load for execution a dataset which is not formatted binary or which has no start address. Typically this means that the dataset being loaded is not a load module.
FØ23	Monitor	Program Size Program too large for core available. Try to overlay the program or make it smaller.
FØ24	Monitor	Request Address File access violation. You are trying to perform an operation that violates the Monitor's user and file protection scheme. See Table K-1 for details. Resolve access problems with owner.
FØ25	PIP	Device (RAD5Ø) Master directory full when attempting to add UIC. No more UIC's can be added.
FØ26	Monitor	Disk Control Status Register Disk (RF11 or RC11) transfer failure. Hardware error or persistent parity failure.
FØ27	Monitor	Error Register Disk (RK11) transfer failure.
FØ3Ø	OTS	Error Class, Number FORTRAN system error. An illegal call to the FORTRAN Error Processor was made.
FØ31	OTS	Addr. of Log Device No more room on FORTRAN logging device, or illegal end-of-file was encountered while a FORTRAN READ was in progress.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING	
FØ32	Monitor	Status Register	Magtape hardware error.
FØ33	Monitor	Special Function Block Address	Invalid special function block.
FØ34	Monitor	Call Address	The call code passed to a conversion request was invalid, e.g., 5 means binary-to-octal, but 63 is not defined.
FØ35	Monitor	Block Number	Illegal block number (RK11).
FØ36	RSX	Lowest Slot Used by Tasks	No slot available.
FØ37		Lowest Slot used by Tasks	Illegal slot specified.
FØ4Ø	RSX	Low Address of Task Code	Attempted to overlay the executive for another task.
FØ41	RSX	Load address of Binary Block	Attempted to load outside limits defined in the command.
FØ42	Monitor	Error Register	Disk (RP11) transfer failure.
FØ43	Monitor	Block Number	Illegal block number (RP11).
FØ44	OLD LINK-11	Ø	Error in command string passed by a Compiler via the .RUN request.
FØ45	Monitor	Request Address	The RUN EMT cannot find the requested entry in the specified core image library. Add proper entry to CIL or use correct name.
FØ46	Monitor	Ø	RD11C Descriptor table was overwritten. Reload the system.
FØ5Ø	Monitor	Request Address	Illegal I/O to batch stream. Either an illegal mode (e.g., unformatted binary when not in "OWN" mode) or a byte count less than 83, on formatted read.
FØ51	Monitor	Request Address	Too many successive read errors or EOF's while reading the batch stream.

CODE/ISSUER ADDITIONAL INFORMATION/MEANING

FØ52 Monitor

PC

Illegal Open to one of the Batch Datasets. OPENO and OPENI are the only legal OPEN/s and OPENO (OPENI) to an input (output) dataset is also illegal.

FØ53 Monitor

PC

Illegal request to the BATCH stream flush EMT. Request code must be Ø, 1, or 2.

FØ54 Monitor

Address of DDB

An attempt was made to load a new program via the RUN request (EMT) before releasing all of the datasets INITed by the current program. Correct the program by closing and releasing all datasets before the RUN request is issued.

FØ55 Monitor

PC

The time limit for the current job has expired. The current job has been aborted.

FØ75

PC

Relative to the device driver's base. Unrecoverable hardware failure on a cassette.

FØ76

Device (RAD5Ø) Cassette Tape

The read after write verification attempt failed enough times to exhaust the verification retry count. The cassette tape is probably unusable.

FØ77

Device (RAD5Ø)

There is insufficient memory space available to allocate an I/O buffer of the requested length.

F1ØØ RSX

Address in Call Sequence

Insufficient arguments in call sequence or in console command.

F24Ø Monitor

Irrelevant

An attempt was made to allocate a contiguous file, but not enough contiguous blocks are free.

F274 Monitor

Irrelevant

The stack base address has not been properly set. Thus the stack could not be moved by the RUN EMT as requested. This is probably a program error. The .STSTK request may be used to set the stack base prior to issuing the .RUN request.

F275 OTS

Ø

Incorrect argument to link subroutine.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
F276	Monitor	Request Address The transfer address of the program or overlay to be loaded (by the RUN or GET commands or by the .RUN request) was not specified or is not legal. Specify a transfer address in your source program (END statement) or correct the /TR specification in your linking procedure.
F277	Monitor	Request Address The program or overlay could not be loaded because it was outside the legal load area (on top of the Monitor or the main program or outside actual memory). Relink the program to conform to allowable boundaries. Assure that the section being improperly loaded does not overlay the resident portion of your program.
F3ØØ	FORTRN	Ø FORTRAN Compiler overlays cannot be executed. FORTRN.OVR may be nonexistent or improperly constructed.
F3Ø1	FORTRN	Ø No output file specified for the "/GO" options.
F3Ø2	Monitor	Action Word Illegal options requested in short form of RUNΔEMT.
F3Ø3	OVERLAYS	Ø I/O transfer failure during autoload (local) overlay.
F3Ø4	OVERLAYS	Ø Autoload or manual load overlay files must be contiguous.
F34Ø	Monitor	PC at Time of IOT The DOS/BATCH error routine was called with an invalid error code. This might happen if the program branched into a data area since the integer 4 would be executed as an IOT instruction (the error routine is called via an IOT).
F342	Monitor	Contents of PC Error trap. Probably caused by a reference to a byte boundary or to nonexistent memory or to a nonexistent device. Could also be caused as a consequence of the stack pointer being below 4ØØ or by executing JMP or JSR with register mode destination.

CODE/ISSUER	ADDITIONAL INFORMATION/MEANING
F344 Monitor	Contents of PC Reserved instruction trap. The instruction just executed is not a valid PDP-11 instruction. Perhaps you jumped to a point outside your program or perhaps you have stored information over an instruction.
F346 Monitor	Contents of PC Trace trap. Bit 4 of the Processor Status Register is on. Look for traps in the PDP-11 Processor Handbook.
F35Ø	Power Failure Recovery
F352 Monitor	Contents of PC Trap Instruction trap. A trap instruction was issued by your program and you did not previously specify a trap address with the .TRAP request.
F356 Monitor	Contents of PC Unexpected device interrupt. Either a new device has been added to your system without initializing the interrupt vector or a hardware failure has occurred.
F36Ø	Memory Parity CSR Address Memory parity error. The address of the offending memory parity CSR is given.
F37Ø	Limit error or illegal SIB (system interface block) or UIB (user interface block) entry.

Table K-1

Recovery from FØ12 or FØ24 File Access Violations

CONDITION	ACTION
Are you logged in?	LOGin.
Is your UIC entered?	Enter it with PIP.
Are you attemtping to create a file which already exists?	Run PIP and DELETE.
Does the Input file you are accessing exist?	Use PIP with /BR or /DI switch to check.
Are you attempting to delete a non-existent file?	Use PIP with /BR or /DI switch to check.
Are you attempting to delete a locked file? (The command to delete is correct, and the file exists.)	Run PIP and UNlock.
Are you attempting to access another user's file illegally?	Ask PIP to list the user's directory and see if an access error results.

K.12.4 Information Messages

Information messages are printed and the program generally continues.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
I1ØØ		Hardware Status Register Content For cassette tape, a read-after-write verification attempt failed, but did not exhaust the verification retry count. Refer to FØ76. This message indicates that the tape may be of poor quality.
I35Ø	OTS	STOP Number A STOP statement was executed in a FORTRAN program.
I351	FORTRN	Ø More errors of a specified type occurred than were allowed. The program is terminated.
I352	FORTRN	Address of DEVTB Entry The logical device specified is not available, (see FORTRAN device table, DEVTB, for a layout).
I353	OTS	Error Class Number No logging device. The command input device was in use when a run-time diagnostic message was to be issued. Because of a device conflict, the normal message could not be issued.
I354	PIP	Ø Illegal response to CONFIRM; when attempting to zero an RK11 disk cartridge. The disk was not zeroed. Legal responses are: H for high-density disks (RKØ3/Ø5) L for low-density disk (RKØ2).

K.12.5 Keyboard Command Messages

If a command cannot be executed satisfactorily, an appropriate message will be printed at the teleprinter and the command will be ignored. The message will be one of the following.

Message	Meaning
ILL CMD!	Command requested does not exist.
INV CMD!	Command cannot be accepted at this time (e.g., KILL with no program to kill).
SYN ERR!	Syntax of command is faulty.

Message	Meaning
ILL DEV!	The device specified is illegal.
NO FILE!	File specified does not exist.
ILL ADR!	Address is illegal (not on word-bound or in core).
NO CORE!	Insufficient core capacity to execute command (SAVE).

K.12.6 System Program Messages

System program messages are printed and the program continues. This class of error may be issued by a variety of system programs. If an ISSUER is specified, the error is unique to the indicated program. See the appropriate program manual for greater detail.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
SØØ1		Ø FORTRAN Compiler has exhausted symbol table space during the assembly phase of compilation.
S2ØØ		Ø Too many .CSECT directives.
S2Ø1		Ø Conditionals nested too deeply.
S2Ø2		Error Status Byte. Dev: file, ext. EOD or device error on .WRITE or .READ; the disk may have filled up.
S2Ø3	OLD LINK-11	Relative address of error call Illegal switch, or too many switches, or illegal switch value, or switch value not given, or switch in output field.
S2Ø4	OLD LINK-11	Relative address of error call Too many or too few output files.
S2Ø5	OLD LINK-11	Ø Too many or too few input files.
S2Ø6	OLD LINK-11	Relative address of error call No input files specified.
S2Ø7		Error Status Byte EOD or device error on .TRAN.
S21Ø	OLD LINK-11	Ø, dev:file.ext Unrecognized symbol table entry in indicated file.
S211	OLD LINK-11	Ø, dev:file.ext An RLD of the given file references a global name which cannot be found in the symbol table.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
S212	OLD LINK-11	Ø, dev:file.ext An RLD of the given file contains a location counter modification command which is not last.
S213	OLD LINK-11	Ø, dev:file.ext Object module does not start with a GSD in the indicated file.
S214	OLD LINK-11	Ø, dev:file.ext The first entry in the module is not the module name of the indicated file.
S215	OLD LINK-11	Ø dev:file.ext An RLD of the given file references a section name which cannot be found.
S216	OLD LINK-11	Ø The TRA specification references a nonexistent module name.
S217	OLD LINK-11	Relative address at error call. Insufficient core.
S22Ø	OLD LINK-11	Ø An internal jump table index is out of range.
S223	OLD LINK-11	Ø No more room for CSI input buffer or Monitor's file manager routine, or Monitor's library search buffer.
S225	OLD LINK-11	Ø Program too large or top too low (program has been linked below zero in memory).
S226	OLD LINK-11	Ø An open angle bracket, <, is present in a line other than the first.
S227		Error Code Illegal file combinations due to name conflicts. Defined error codes are: 1 No Primary File (PRI) output, 2 Secondary File (SEC) input = SEC output. 3 SEC input = PRI output, 4 PRI input = SEC output, 5 PRI input = SEC input, 6 PRI output = SEC output.
S23Ø		Error Status Byte Error on.BLOCK I/O.
S231		Illegal command, file-structured device required.
S232		No more than one action switch permitted.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
S233		Specified UIC not found in MFD.
S234		Null filename of "*" given where filename required.
S235		No files found in UFD.
S236		Operation applicable to DECtape only.
S237		File not found during file recovery operation.
S24Ø.		No space for file allocate.
S241		MFD is full.
S242		Meaningless command, no action taken.
S243	Ø	An open angle bracket, <, is not present in the first line.
S244	Ø	Already past requested position.
S245	Ø	Object module not found, could be out of order.
S246	Ø	Illegal library format.
S247	Ø	Listing requested, but unable to read output library from specified output device.
S25Ø	Ø	Core library symbol table not specified first or consecutively.
S251	Ø	No files found for "*" request.
S252	Ø	Filename given when none allowed.
S253	Ø OLD LINK-11	Linker error.
S254	Ø	It is illegal to zero the system resident disk.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
S255	Ø OLD LINK-11	Match found in third of later binary block in a paper tape library.
S256	Ø	Illegal input device.
S257		File Block Error Code, dev:file.ext Illegal file operation. For example, protect code does not allow transfer of file; UIC different from Login UIC, thus making certain "wildcard" operations illegal. The operation in question is not performed.
S26Ø		Ø Same device needed for input and output in fast copy operation.
S262		Ø Record size too big for buffer.
S263		File Number File record sizes do not agree on verify, "/V".
S264		Ø Conflict in standard file name extension which determines mode of transfer. Use explicits to resolve.
S265		Ø Operation attempted on device which is not legal for nonprivileged user. For example, /PK PIP switch attempted by a user not logged in under [1,1].
S266		Ø An attempt was made either to: rename a nonexisting file or to rename an already existing name.

K.12.7 Warning Messages

Warning messages are printed and the program generally continues.

CODE/ISSUER	ADDITIONAL INFORMATION/MEANING
WØØ2	Device Name (RAD5Ø) Device time out.
WØØ3	Request Address The selected label record does not contain a type code of 7 and is therefore of unknown (suspect) origin.
WØ43	Block Number Transfer error while using .TRAN to zero the disk.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
WØ5Ø		Device mnemonic (RAD5Ø) Cassette tape The handler of the cassette device driver encountered EOT. The message is also issued if rewinds are suppressed and any error is detected while writing the sentinel label record.
WØ51		Device (RAD5Ø) Cassette Tape An I/O timing or CRC failure sufficient to exhaust the specified retry count.
W1Ø1	RSX	Number of Task Called Task called by number not present or call number illegal. Request ignored.
W1Ø2	RSX	Addr. in Call Sequence Delay units not correct in call start. Request ignored.
W1Ø3	RSX	Addr. in Call Sequence Delay time too large in call start. Request ignored.
W1Ø4	RSX	Addr. in Call Sequence No time slot available. Request ignored.
W1Ø5	RSX	Current Run-Time A level 1 task has exceeded its maximum run time. Task continued.
W1Ø6	RSX	Illegal or unrecognized console command. Command ignored.
W1Ø7	RSX	Report Number Illegal system report number in system command. Command ignored.
W11Ø	RSX	Addr. in Call Sequence Attempted to start a background task while the background is busy. Request ignored.
W111	RSX	Addr. in Call Sequence Attempted to clock a background task. Request ignored.
W112	RSX	Symbolic task name not found. Request ignored.
W113	RSX	Command syntax error. Command ignored.
W114	RSX	Addr. in Call Sequence Illegal clock (call TRNON) time. Request ignored.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING
W3ØØ		Ø, Module Name Non-unique object module detected in first pass. Second and subsequent occurrences of the module are ignored.
W3Ø1	OLD LINK-11	Addr. of Byte Error Byte relocation error. Linker automatically continues.
W3Ø2	OLD LINK-11	Ø, Symbol and Module Names Multiple definitions of global symbol. Second definition is ignored and linking continues.
W3Ø3	EDIT	Buffer overflow. Overflow of one of the following Editor buffers: Command Input Buffer Save Buffer Page Buffer
W3Ø4	EDIT	Macro overflow. The command string as stored in the Save Buffer was too long to execute, when requested to do so by an EM (Execute Macro) command.
W3Ø5	EDIT	Recursive macro. The command string as stored in the Save Buffer contains an EM command.
W3Ø6	EDIT	Empty Save Buffer. An EM or U (Unsave) command was issued with nothing in the Save Buffer.
W3Ø7	EDIT	Search failure. The nth occurrence of the search object was not found in the available test.
W31Ø	EDIT	Unsave failure. Insufficient room to copy the contents of the Save Buffer into the Page Buffer at dot.
W311	EDIT	End-of-data detected. The end of the input file or the end of the input medium was reached during the last read of text into the Page Buffer, last page read was last in the file.
W312	EDIT	Illegal line feed. A line feed character was encountered in the command string.

CODE/ISSUER		ADDITIONAL INFORMATION/MEANING	
W313	EDIT		Illegal negative argument. A negative argument was used with a command that does not accept negative arguments.
W314	EDIT		Arguments not permitted. The command specified does not permit any argument with it.
W315	EDIT		Illegal argument. The given argument was not acceptable to the specified command.
W316	EDIT		Illegal text string.
W317	EDIT		Illegal command. The Editor was unable to execute the specified command. The command may be an illegal character, one that is not an EDIT command character.
W32Ø	EDIT		Page Buffer almost full. The Page Buffer was within 128 characters of being full. Write out part or all of the Page Buffer and then delete from the Buffer the part that was written.
W321	EDIT		File closed. An attempt to Read from or Write to a primary file after an EF (End-of-File) command was issued.
W322	OLD LINK-11	Ø	Undefined global symbols in load module. Linking continues.
W323	OLD LINK-11 RSX		Illegal size of named .CSECT or illegal entry in named .CSECT or task's named .CSECT size too large.
W324	OLD LINK-11 RSX		Too many entries in tasks named .CSECT.
W325	OLD LINK-11 RSX		Illegal priority specification in real-time header.
W35Ø	RSX	Number of Failures	Powerfail interrupt occurred.
W352	RSX	Disk Error Code	Disk error detected by RSX. Codes are: 3 transmission error 5 illegal error 6 undefined file 7 illegal file, i.e., linked 8 block of file out of range

CODE/ISSUER	ADDITIONAL INFORMATION/MEANING
W36Ø	Ø = Syntax Error; 1 = Illegal Command. Illegal command or syntax error has occurred. The command is ignored.
W361	A message without a legal start character was encountered during a receive operation. The message is received, but ignored. (This situation is equivalent to an R1 error in the core-only version.)
W362	A message without a legal termination character was encountered during a receive operation. The system aborts the receive. (This situation is equivalent to the R2 error in the core only version.)
W363	Ø = Bit Message; n = Data Message with n as the active dataset identifier (i.e., 1, 2, or 3) An illegal response or no response was sent to acknowledge a transmit operation. The system aborts the transmit operation. (This situation is equivalent to R2 error in the core-only version.)
W364	Ø = Bit Message; n = Data Message with n as the active dataset identifier (i.e., 1, 2, or 3) "RVI" response to transmission during a transmit operation. (This message is equivalent to T1 message in the core-only version.)
W365	The dataset or a line printer became Not Ready during a receive operation. The system aborts the receive operation defaults to the line printer for subsequent receive operations until a new receive transfer command is specified.
W366	Ø = Initial Preparation; 1 = During Receive File-Chain During receive transfer preparation, the dataset cannot be prepared for transfer. The system defaults to the line printer for subsequent operations until a new receive transfer command is specified.
W367	Transmit Operation Active -- Dataset (or card reader) became not ready before the end of file was detected. Either make the device ready or type TERM to terminate the transmission.
W37Ø	Ø = Modem Not Ready; n = Dataset At Fault with n as the data set identifier (i.e., 1, 2, or 3) During transmit transfer preparation, transmission cannot be initiated because the modem is not in data mode or the dataset cannot be prepared for read (INIT or OPEN). The command is ignored.