SOFTWARE OPERATING PROCEDURES

ASSEMBLER, FORTRAN AND ALGOL ERROR MESSAGES

PREREQUISITE SOP MODULES:

Introduction (5951-1369)

REFERENCE MANUALS:

HP Assembler (02116-9014)

HP FORTRAN (02116-9015)

HP ALGOL (02116-9072)

HEWLETT hp PACKARD

11000 Wolfe Road
Cupertino, California 95014

5951-1377

April, 1971

ASSEMBLER, FORTRAN AND ALGOL ERROR MESSAGES

During the compilation or assembly of programs, error messages are typed on the list output device to aid the programmer in debugging programs. This SOP module consists of three procedures as follows:

PROCEDURE 1
ASSEMBLER ERROR MESSAGES

Errors detected in the source program are indicated by a 1- or 2- letter mnemonic followed by the sequence number and the first 62 characters of the statement in error. The messages are printed on the list output device during the passes indicated:

For Extended Assembler, error listings produced during Pass 1 are preceded by a number which identifies the source input file where the error was found. Pass 2 and 3 error messages are preceded by a reference to the previous page of the listing where an error message was written. The first error will refer to page "O".

Error Code	Pass	Description
CS	1	Control statement error: a) The control statement contained a parameter other than the legal set. b) Neither A nor R, or both A and R were specified. c) There was no output parameter (B, T, or L.)
DD	1	Doubly defined symbol: A name defined in the symbol table appears more than once as: a) A label of a machine instruction. b) A label of one of the pseudo operations: BSS EQU ASC ABS DEC OCT DEF Arithmetic subroutine call DEX c) A name in the Operand field of a COM or EXT statement. d) A label in an instruction following a REP pseudo operation.

Error Code	Pass	Description
		e) Any combination of the above. An arithmetic subroutine call symbol appears in a program both as a pseudo instruction and as a label.
EN	1	The symbol specified in an ENT statement has already been defined in an EXT or COM statement.
ENØØØ <symbol>	2	The entry point specified in an ENT statement does not appear in the label field of a machine or BSS instruction. The entry point has been defined in the Operand field of an EXT or COM statement, or has been equated to an absolute value.
IF	1	An IFZ or an IFN follows either an IFZ or an IFN without an intervening XIF. The second pseudo instruction is ignored.
IL	1	Illegal instruction: a) Instruction mnemonic cannot be used with type of assembly requested in control statement. The following are illegal in an absolute assembly: NAM EXT ENT COM ORB Arithmetic subroutine calls b) The ASMB statement has an R parameter, and NAM has been detected after the first valid Opcode.
IL	2 or 3	Illegal character: A numeric term used in the Operand field contains an illegal character (e.g. an octal constant contains other than +, -, or Ø-7). Illegal instruction: ORB in an absolute assembly.
M	1, 2 or 3	Illegal operand: a) Operand is missing for an Opcode requiring one. b) Operands are optional and omitted but comments are included for: END HLT

Error Code	Pass	Description
M	1, 2 or 3	c) An absolute expression in one of the following instructions from a relocatable program is greater than 77_8. Memory Reference DEF Arithmetic subroutine calls d) A negative operand is used with an Opcode field other than ABS, DEX, DEC, and OCT. e) A character other than I follows a comma in one of the following statements: ISZ ADA AND DEF JMP ADB XOR Arithmetic JSB LDA IOR subroutine LDB CPA calls STA CPB STB f) A character other than C follows a comma in one of the following statements: STC MIB CLC OTA LIA OTB LIB HLT MIA g) A relocatable expression in the operand field of one of the following: ABS ASR RRL REP ASL LSR SPC RRR LSL h) An illegal operator appears in an Operand field (e. g. + or - as the last character). i) An ORG statement appearing in a relocatable program includes an expression that is base page or common relocatable or absolute. j) A relocatable expression contains a mixture of program, base page, and common relocatable terms.

Error Code	Pass	Description
		k) An external symbol appears in an operand expression or is followed by a common and the letter I. l) The literal or type of literal is illegal for the operation code used (e.g., STA = B7). m) An illegal literal code has been used (e.g., LDA = 077). n) An integer expression in one of the following instructions does not meet the condition $1 \leq n \leq 16$. The integer is evaluated modulo 2^4. ASR RRR LSR ASL RRL LSL o) The value of an 'L' type literal is relocatable.
NO	1, 2, 3	No origin definition: The first statement in the assembly containing a valid opcode following the ASMB control statement (and remarks and/or HED, if present) is neither an ORG nor a NAM statement. If the A parameter was given on the ASMB statement, the program is assembled starting at 2000; if an R parameter was given, the program is assembled starting at zero.
OP	1, 2, 3	Illegal Opcode preceding first valid Opcode. The statement being processed does not contain an asterisk in position one. The statement is assumed to contain an illegal Opcode; it is treated as a remarks statement.
OP	1,2, or 3	Illegal Opcode: A mnemonic appears in the Opcode field which is not valid for the hardware configuration or assembler being used. A word is generated in the object program.
OV	1,2, or 3	Numeric operand overflow: The numeric value of a term or expression has overflowed its limit:

Error Code	Pass	Description
		$1 \geq N \geq 16$ Shift-Rotate Set
		$2^{6}-1$ Input/Output, Overflow, Halt
		$2^{10}-1$ Memory Reference (in absolute assembly)
		$2^{15}-1$ DEF and ABS operands; data generated by DEC; or DEX: expressions concerned with program location counter.
		$2^{16}-1$ OCT
R?	Before 1	An attempt is made to assemble a relocatable program following the assembly of an absolute program.
SO		There are more symbols defined in the program than the symbol table can handle.
SY	1,2,3	Illegal Symbol: A Label field contains an illegal character or is greater than 5 characters. A label with illegal characters may result in an erroneous assembly if not corrected. A long label is truncated on the right to 5 characters.
SY	2 or 3	Illegal Symbol: A symbolic term in the Operand field is greater than five characters; the symbol is truncated on the right to 5 characters.
		Too many control statements: A control statement has been input both on the teleprinter and the source tape or the source tape contains more than one control statement. The Assembler assumes that the source tape control statement is a label, since it begins in column 1. Thus, the commas are considered as illegal characters and the "label" is too long. The binary object tape is not affected by this error, and the control statement entered via the teleprinter is the one used by the Assembler.
TP	1,2, or 3	An error has occurred while reading magnetic tape.
UN	1,2, or 3	Undefined Symbol:

Error Code	Pass	Description
		a) A symbolic term in an Operand field is not defined in the Label field of an instruction or is not defined in the Operand field of a COM or EXT statement. b) A symbol appearing in the Operand field of one of the following pseudo operations was not defined previously in the source program: BSS ASC EQU ORG END

PROCEDURE 2
FORTRAN ERROR MESSAGES

Errors detected in the source program are indicated by a nemeric code inserted before or after the statement in the List Output.

The format is as follows:

E-eeee: ssss + nnnn

eeee — The error diagnostic code shown below.

ssss — The statement label of the statement in which the error was detected. If unlabeled, 0000 is typed.

nnnn — Ordinal number of the erroneous statement following the last labeled statement. (Comment statements are not included in this count.)

Error Code — Description

0001 Statement label error:

a) The label is in positions other than 1-5.

b) A character in the label is not numeric.

c) The label is not in the range 1-9999.

d) The label is doubly defined.

e) The label indicated is used in a GO TO, DO, or IF statement or in an I/O operation to name a FORMAT statement, but it does not appear in the label field for any statement in the program (printed after END).

0002 Unrecognized Statement

a) The statement being processed is not recognized as a valid statement.

b) A specifications statement follows an executable statement.

Error Code — Description

c) The specification statements are not in the following order:

DIMENSION
COMMON
EQUIVALENCE

d) A statement function precedes a specification statement.

0003 — Parenthesis error: There are an unequal number of left and right parentheses in a statement.

0004 — Illegal character or format:

a) A statement contains a character other than A through Z, 0 through 9, or space =+-/(),.$".

b) A statement does not have the proper format.

c) A control statement is missing, misspelled, or does not have the proper format.

d) An indexing parameter of a DO-loop is not an unsigned integer constant or simple integer variable or is specified as zero.

0005 — Adjacent operators: An arithmetic expression contains adjacent arithmetic operators.

0006 — Illegal subscript: A variable name is used both as a simple variable and a subscripted variable.

0007 — Doubly defined variable:

a) A variable name appears more than once in a COMMON statement.

b) A variable name appears more than once in a DIMENSION statement.

c) A variable name appears more than once as a dummy argument in a statement function.

Error Code — Description

d) A program subroutine, or function name appears as a dummy parameter; in a specifications statement of the subroutine or function; or as a simple variable in a program or subroutine.

0008 Invalid parameter list:

a) The dummy parameter list for a subroutine or function exceeds 63.

b) Duplicate parameters appear in a statement function.

0009 Invalid arithmetic expression:

a) Missing operator

b) Illegal replacement

0010 Mixed mode expression: integer constants or variables appear in an arithmetic expression with real constants or variables.

0011 Invalid subscript:

a) Subscript is not an integer constant, integer variable, or legal subscript expression.

b) There are more than two subscripts (i.e., more than two dimensions.)

c) Two subscripts appear for a variable which has been defined with one dimension only.

0012 Invalid constant:

a) An integer constant is not in the range of -2^{15} to 2^{15} -1.

b) A real constant is not in the approximate range of 10^{38} to 10^{-38}.

c) A constant contains an illegal character.

Error Code   Description

0013   Invalid EQUIVALENCE statement:

a) Two or more of the variables appearing in an EQUIVALENCE statement are also defined in the COMMON block.

b) The variables contained in an EQUIVALENCE cause the origin of COMMON to be altered.

c) Contradictory equivalence; or equivalence between two or more arrays conflicts with a previously established equivalence.

0014   Table overflow: Too many variables and statement labels appear in the program.

0015   Invalid DO loop:

a) The terminal statement of a DO loop does not appear in the program or appears prior to the DO statement.

b) The terminal statement of a nested DO loop is not within the range of the outer DO loop.

c) DO loops are nested more than 10 deep.

d) Last statement in a loop is a GO TO, arithmetic IF, RETURN, STOP, PAUSE, or DO.

0016   Statement function name is doubly defined.

PROCEDURE 3
ALGOL ERROR MESSAGES

Source Program Diagnostic Message

Errors detected in the source program are indicated by a code number and an "↑" below the symbol which caused the error.

Error Code	Description
1	More than two characters used in an ASCII constant
2	@ not followed by an octal digit
3	Octal constant greater than 177777
4	Two decimal points in one number
5	Non-integer following apostrophe
6	Label declared but not defined in program
7	Number required but not present
8	Missing END
10	Undefined identifier
11	Illegal symbol
12	Procedure designator must be followed by left parenthesis
13	Parameter types disagree
14	Name parameter may not be an expression
15	Parameter must be followed by a comma or right parenthesis
16	Too many parameters
17	Too few parameters
18	Array variable not followed by a left bracket
19	Subscript must be followed by a comma or right bracket
20	Missing THEN
21	Missing ELSE
22	Illegal Assignment
23	Missing Right Parenthesis
24	Proper procedure not legal in arithmetic expression
25	Primary may not begin with this type quantity

Error Code	Description
26	Too many subscripts
27	Too few subscripts
28	Variable required
4Ø	Too many external symbols
41	Declarative following statement
42	No parameters declared after left parenthesis
43	REAL, INTEGER, or BOOLEAN illegal with this declaration.
44	Doubly defined identifier or reserved word found
45	Illegal symbol in declaration
46	Statement started with illegal symbol
47	Label not followed by colon
48	Label is previously defined
49	Semicolon expected as terminator
5Ø	Left arrow or := expected in SWITCH declaration
51	Label entry expected in SWITCH declaration
52	Real number assigned to integer
53	Constant expected following left arrow or :=
54	Left arrow or := expected in EQUATE declaration
55	Left bracket expected in array declaration
56	Integer expected in array dimension
57	Colon expected in array dimension
58	Upper bound less than lower bound in array
59	Right bracket expected at end of array dimensions
6Ø	Too many values for array initialization
61	Array size excessive (set to 2Ø47)
62	Constant expected in array initialization
63	Too many parameters for procedure
64	Right parenthesis expected at end of procedure parameter list
65	Procedure parameter descriptor missing
66	VALUE parameter for procedure not in list
67	Illegal TYPE found in procedure declaration
68	Illegal description in procedure declaratives
69	Identifier not listed as procedure parameter
7Ø	No type FOR variable in procedure parameter list
71	Semicolon found in a format declaration
72	Left parenthesis expected after I/O declaration name

Error Code	Description
73	Right parenthesis expected after I/O name parameters
74	Undefined label reference
75	Switch identifier not followed by a left bracket
76	Missing right bracket in switch designator
77	THEN missing in IF statement
78	DO missing in WHILE statement
79	FOR variable must be of type INTEGER
8Ø	FOR variable must be followed by an assign symbol
81	STEP symbol missing in FOR clause
82	UNTIL symbol missing in FOR clause or DO statement
83	DO symbol missing in FOR clause
84	Parenthesis expected in READ/WRITE statement
85	Comma expected in READ/WRITE statement
86	Free field format (*) illegal with WRITE
87	Unmatched [in I/O statement list
88	Missing BEGIN in case statement
89	Missing END in case statement
1ØØ	Program must start with BEGIN, REAL, INTEGER or PROCEDURE. Computer halts with 102077_8 in MEMORY DATA Register.
999	Table areas have overflowed, program halts with 102077_8 in MEMORY DATA Register.