Appendix VIII: Error codes and reserved words

Error Numbers and Error Messages

When BASIC encounters a program statement, word or variable that it cannot understand or process, it will stop and display an error message. The form of the message will generally indicate what went wrong - and sometimes, if the error is a typographical error during program entry, BASIC will prompt in edit mode, with the line where the incorrect entry was made.

The most popular error to greet the inaccurate typist is the Syntax Error (number 2), and BASIC prompts with the line to edit if encountered in program mode. In direct mode, it simply states that an error occured, and assumes that the last line typed is visible to spot the problem.

If the ON ERROR GOTO command is included at the beginning of a program, it may refer the computer to a given line number when detecting an error. In the following example, the computer is referred to line 1000 when detecting an error:

```
10 ON ERROR GOTO 1000
        program
1000 PRINT CHR$(7):MODE 2:INK 1,0: INK 0,9: CLS :LIST
```

Whereupon the CPC464 will beep, clear the current screen, change to a suitable colour combination for the 80 column display, and list the program ready for examination. If the error is a Syntax error it will appear at the foot of the listing, awaiting correction in the line edit mode, although the Syntax error message is suppressed.

Remember to END the program on the last line before 1000 if you wish to save the results on the screen.

BASIC will not produce error messages for valid input - so it must be assumed that whenever an error does occur, it can be traced back to an error in the form of the program, usually guided by the message produced to help in the process of de-bugging. As with most things, you will learn most readily from your mistakes, so make the most of the fact that the CPC464 is the most tolerant of tutors: you will tire of trying long before the CPC464 loses its patience!

All errors generated by BASIC are listed here, in error number order. The messages produced by BASIC are given, as well a brief description of possible causes.

1 ***Unexpected NEXT***

A N E X T command has been encountered while not in a F 0 R loop, or the control variable in the N E X T command does not match that in the F 0 R.

2 ***Syntax Error***

BASIC cannot understand the given line because a construct within it is not legal.

3 ***Unexpected RETURN***

A R E T U RN command has been encountered when not in a subroutine.

4 ***DATA exhausted***

A R E A D command has attempted to read beyond the end of the last D A T A.

5 ***Improper argument***

This is a general purpose error.The value of a function's argument, or a command parameter is invalid in some way.

6 ***Overflow***

The result of an arithmetic operation has overflowed. This may be a floating point overflow, in which case some operation has yielded a value greater than 1.73-38 (approx.). Alternatively, this may be the result of a failed attempt to change a floating point number to a 16 bit signed integer.

7 ***Memory full***

The current program or its variables may be simply too big, or the control structure istoodeeplynested(nestedGOSUBs,WHILEsor FORs).
A ***MEMORY*** command will give this error if an attempt is made to set the top of BASIC's memory too low, or to an impossibly high value. Note that an open cassette file has a buffer allocated to it, and that may restrict the values that M E M 0 R Y may use.

a ***Line does not exist***

The line referenced cannot be found.

9 ***Subscript out of range***

One of the subscripts in an array reference is too big or too small.

10 ***Array already dimensioned***

One of the arrays in a D I M statement has already been declared.

11 ***Division by zero***

May occur in Real division, integer division, integer modulus or in exponentiation.

12 ***Invalid direct command***

The last command attempted is not valid in Direct Mode.

13 *Type mismatch*

A numeric value has been presented where a string value is required. and vice versa, or an invalidly formed number has been found in R E A D or I N P U T.

14 *String space full*

So many strings have been created that there is no further room available, even after 'garbage collection'.

15 *String too* long

String exceeds 255 characters in length. May be generated by adding a number of strings together.

16 *String expression too complex*

String expressions may generate a number of intermediate string values. When the number. of these values exceeds a reasonable limit, BASIC gives up, and this error results.

17 *Cannot* CONTinue

For one reason or another the current program cannot be restarted using C 0 N T. Note that C 0 N T is intended for restarting after a S T 0 P command, **[ESC][ESC]** or error, and that any alteration of the program in the meantime makes a restart impossible.

18 *Unknown user function*

No D E F F *N* has been executed for the F N just invoked.

19 RESUME missing

The end of the program has been encountered while in Error Processing Mode (ie in an ON *ERROR* GOTO routine).

20 *Unexpected RESUME*

R E S U M *E* is only valid while in Error Processing Mode (ie in an 0 N E RR 0 R G 0 T 0 routine).

21 *Direct command found*

When loading a program from cassette a line without a line number has been found.

22 *Operand missing*

BASIC has encountered an incomplete expression.

23 *Line too long*

A line when converted to BASIC internal form becomes too big.

24 EOF met

An attempt has been made to read past end of file on the cassette input stream.

25 File ***type error***

The cassette file being read is not of a suitable type. 0 PEN I N is only prepared to open ASCII text files. LO A D, ***R U N*** etc, are only prepared to deal with the file types produced by S A VE.

26 ***NEXT missing***

Cannot find a N E X T to match a F 0 R command.

27 ***File already open***

An 0 ***PEN*** I N or 0 PEN OU T command has been executed before the previously opened file has been closed.

28 ***Unknown command***

BASIC cannot find a taker for an external command.

29 ***WEND missing***

Cannot find a W E N D to match a W H I L E command.

30 ***Unexpected WEND***

Encountered a W E N D when not in a W H I L E loop, or a W EN D that does not match the current W H I LE loop.

BASIC Keywords

The following are the BASIC keywords, they are reserved and cannot be used as variable names.

ABS, AFTER, AND, ASC, ATN, AUTO

BIN$, ***BORDER***

CALL, CAT, CHAIN, CHR$, CINT, ***CLEAR, CLG,*** CLOSEIN, ***CLOSEOUT, CLS, CONT, COS, CREAL***

DATA, DEF, DEFINT, DEFREAL, DEFSTR, DEG, DELETE, DI, DIM, ***DRAW, DRAWR***

EDIT, EI, ***ELSE, END, ENT, ENV, EOF, ERASE, ERL, ERR, ERROR, EVERY, EXP***

FIX, FN, FOR, FRE

GOSUB, GOTO

HEX$, HIMEM

IF, INK, INKEY, INKEY$, INP, INPUT, INSTR, INT

JOY

KEY

LEFTS, *LEN*, *LET*, *LINE*, *LIST*, *LOAD*, *LOCATE*, *LOG*, LOG10, LOWER$

MAX, *MEMORY*, *MERGE*, *MID$*, *MIN*, *MOD*, *MODE*, *MOVE*, *MOVER*

NEXT, *NEW*, *NOT*

ON, *ON BREAK*, *ON ERROR* GOTO, *ON SQ*, OPENIN, OPENOUT, *OR*, *ORIGIN*, *OUT*

PAPER, PEEK, PEN, PI, PLOT, PLOTR, POKE, POS, PRINT

RAD, *RANDOMIZE*, *READ*, *RELEASE*, *REM*, *REMAIN*, *RENUM*, *RESTORE*, *RESUME*, *RETURN*, *RIGHT$*, *RND*, *ROUND*, *RUN*

SAVE, *SGN*, *SIN*, *SOUND*, SPACE$, *SPC*, *SPEED*, *SQ*, *SQR*, STEP, STOP, STR$, STRING$, SWAP, SYMBOL

TAB, *TAG*, TAGOFF, *TAN*, *TEST*, *TESTR*, *THEN*, *TIME*, *TO*, TROFF, TRON

UNT, UPPER$, USING
VAL, *VPOS*

WAIT, *WEND*, *WHILE*, *WIDTH*, *WINDOW*, *WRITE*

XOR, *XPOS*

YPOS

ZONE