

BASC is a D-BASIC V to C compiler which generates a pure C code output from a D-BASIC V program code input. Thus executable programs are generated by using the computer's normal C compiler.

BASC is based upon a translator/compiler program combined with libraries of runtime functions.

The translator/compiler uses D-BASIC V '.bac' code as input, and writes C source code as output. The output is linked to the runtime support functions to form an executable program. BASC is totally written in the C language and is available on many different UNIX systems on the market, in combination with D-BASIC V.

BASC features

- o D-BASIC V code conforms to ANSI X3.60 standard with many extensions
- o D-BASIC V long variable names
- o Multiline recursive functions and procedures with local variables
- o REPEAT...UNTIL and WHILE...WEND statements
- o Multiline IF...IFEND with ELIF
- o Open Pipe statements for effective fork handling
- o Request statements for access to all UNIX system calls
- o Floating point arithmetic with trigonometric functions
- o Advanced file handling with path/name translations tables
- o Program chaining with common statement generates one executable module
- o Formatted print commands
- o Date and time functions
- o C subroutines easily added
- o Writes separate compilable programs
- o 3 to 50 times faster execution after using BASC
- o Since programs are compiled they become list-protected
- o Mixes with other languages e. g. Fortran, Pascal.

Programming development

All programming is done in D-BASIC V with interactive and immediate syntax control and error monitoring. Multi-line functions and procedures permit both parameters and results to be transferred for easy program structuring. Automatically indented loops provide for easy readability. Any type of error can be handled without stopping the program, promoting very stable application execution.

Trace and single step functions simplify debugging of the finished program.

The finished D-BASIC V program, working correctly, is compiled by BASC to C language, using a C compiler. The code, being compiled to object code, is linked to the BASC runtime function into a very fast executing program module. The original D-BASIC V code is still intact making maintenance very easy.

ISAM (option)

ISAM is a database handling tool which is optional to D-BASIC V. Special instructions are used to create anything

from simple telephone lists to advanced database systems. The ISAM is working in a B-tree fashion and permits the user to work with one or more databases at the same time. Index sequential access with fixed record length allows up to 10 indices in the same database. All indices and keys are updated during writing, eliminating the need for sorting.

MIMER (option)

MIMER is a multi-user relational database management system which is optional to D-BASIC V through the addition of database access statements. MIMER, now a whole family of database access products, contains an active data dictionary controlling data access, usage and security.

This option may be included in D-BASIC V provided MIMER/DB is operating on the system

Communication with other programs

D-BASIC V programs can co-operate with assembly routines located in internal memory areas. Communication between processes in the system is handled with 'PIPE' statements. With the REQUEST statement direct use can be made of the UNIX system calls. C subroutines may easily be integrated with programs produced with BASC.

Data types and variables

D-BASIC V has 32 bit integers, 32 or 64 bit floats and may contain strings up to 64 kbytes. Names can have a length of 160 characters. All datatypes can be used in vectors and matrices with any number of dimensions.

Operations

- o Arithmetics: Exponential functions (**), Multiplication (*), Division(/), Addition (+), Subtraction (-).
- o Logic: and, eqv, imp, not, or, xor.
- o Relations: =, <>, >, <, >=, <=.
- o ASCII: String arithmetics with up to 126 character precision.

Data instructions

LET, READ, RESTORE, DIM, COMMON, DATA, SINGLE, DOUBLE, SHORT INT, LONG INT, EXTEND, NO EXTEND, INTEGER, FLOAT, OPTION BASE, SWAP, RANDOMIZE, FIELD, CASEUP, CASEDN



Program flow instructions

STOP, END, BYE, CHAIN, DEF FN, RETURN, FNEND, DEF PROC, PROC, PROCEND, REPEAT...UNTIL, FOR, FOR...TO...STEP, NEXT, GOSUB, GOTO, ON...GOSUB, ON...GOTO, IF...THEN...ELSE, ELIF, IFEND, WHILE, WEND, ON ERROR GOTO, RESUME, ON...RESTORE, TRACE, NO TRACE

I/O instructions

DIGITS, OPTION EUROPE, INPUT, INPUT LINE, POSIT, PRINT, GET, PUT, PRINT USING, NAME, KILL, OPEN, PREPARE, CLOSE, LOCK, UNLOCK, FIND, INKEY\$, CLS, CRT, (CURUP, CURDN, CURLT, CURRT, HOM, CR, CLE, CLL, IL, DL, BEL, REVON, REVOFF, UNDON, UNDOFF, HFON, HFOFF)

Mathematical functions

ABS(x), FIX(x), INT(x), MOD(x,y), PI, RND, SGN(x), SQR(x), EXP(x), LOG(x), LOG10(x), ATN(x), COS(x), TAN(x), SIN(x), HEX\$(x), OCT\$(x)

String handling functions

ADD\$, DIV\$, MUL\$, SUB\$, COMP%, NUM\$, VAL, ASCII, INSTR, LEN, MID\$, LEFT\$, RIGHT\$, CHR\$, SPACE\$, STRING\$

Miscellaneous instructions and functions

CUR, ERRCODE, FN, REM or !, SLEEP, TAB, TIME\$, ARGV\$, ARGC%, SYSTEM, REQUEST, SCAN, ATIME\$, ESC\$, SYN\$

Advanced instructions and functions

CALL, CVT, OPEN...MODE, PREPARE...MODE, OUT, INP, PEEK, PEEK2, PEEK4, POKE, SWAP%, SWAP2%, SYS, VAROOT, VARPTR

ISAM instructions (option)

ISAM OPEN, ISAM READ, ISAM WRITE, ISAM UPDATE, ISAM DELETE

MIMER instructions and functions (option)

MIMER BEGIN, MIMER OPEN, MIMER GETFIRST, MIMER GETNEXT, MIMER WRITE, MIMER UPDATE, MIMER DELETE, MIMER TRANSACTION, MIMER COMMIT, MIMER ABORT, MIMER END

Installation considerations

- o DS90-XX hardware including D-NIX basic system
- o D-BASIC V 072-8715-XX
- o D-NIX Extension Package 079-8711-XX

Ordering information

BASC 072-8286-XX