STRING FUNCTIONS

Functions arguments N, M represent numeric integer values: X$, Y$ represent string values.

ASC(X$)	Returns ASCII code of first character in string. X = ASC(A$)
CHR$(N)	Returns character for specified ASCII code value A$ = CHR$(143)
HEX$(N)	Returns hexadecimal value of argument as a string PRINT HEX$(46)
INKEY$	Checks keyboard and returns key being pressed. K$ = INKEY$
INSTR(N,X$,Y$)	Searches for first occurrence of target string at position (N). Returns position of match. P = INSTR(8,A$,"TED")
LEFT$(X$,N)	Returns the first N characters of string X$. B$ = LEFT$(A$,9)
LEN(X$)	Returns the length of string X$. K = LEN(K$)
MID$(X$,N,M)	Returns substring of X$ of length M, starting from position N. B$ = (MID$(A$,4,1)
RIGHT$(X$,N)	Return last N character of string X$. B$ = RIGHT$(A$,3)
STRING$(N,X$)	Returns string consisting of N copies of the first character of X$. The ASCII code number for the desired character may replace X$. L$ = STRING$(32,"+") PRINT STRING$(5,41)
STR$(N)	Converts N to its string representation. X$ = STR$ (14.4)
VAL(X$)	Converts numeric characters of X$ to a number. P = VAL(D$)

NUMERIC FUNCTIONS

Function arguments

a	numeric value ± 10^{38}
n	integer value Ø-65535
r	angle expressed in radians.
x,y	low resolution screen location. *x* Ø-63: *y* Ø-31
w,z	high resolution screen location. *w* Ø-255: *z* Ø-191

ABS(*a*)	Computes absolute value. AY = ABS(Y)
ATN(*a*)	Returns arctangent in radians. X = ATN(A)
COS(*a*)	Returns cosine of angle given in radians. W = COS(2*B)
EXP(*a*)	Returns exponential of number, e*a*. Q = EXP(-5.6)
FIX(*a*)	Returns truncated value. Z = FIX(13.43)
INT(*a*)	Converts number to integer. A = INT(5.7)
JOYSTK(*a*)	Returns horizontal or vertical co-ordinate of joystick. Ø = horizontal, left joystick. 1 = vertical, left joystick 2 = horizontal, right joystick. 3 = vertical, right joystick. H = JOYSTK(Ø) V = JOYSTK(P)
LOG(*a*)	Returns natural logarithm. Z = LOG(7.842)
MEM	Returns the amount of free memory. PRINT MEM
PEEK(*n*)	Returns contents of specified memory location, *n*. P = PEEK(65082)
POINT(*x,y*)	Tests graphic cell at (*x,y*) Returns colour code if on, Ø if off, or -1 if a text character. T = POINT(15,25)
POS(*d*)	Returns current print position of device *d*. P = POS(-2)
PPOINT(*w,z*)	Tests high resolution graphics cell at (*w,z*). Returns colour code if on, Ø if off. P = PPOINT(186,54)
RND(*n*)	Generates random integer between 1 and *n*. If is Ø random number is between Ø and 1. X = RND(18)
SGN(*a*)	Returns sign of number as:1 positive: Ø zero: -1 negative. G = SGN(4*H/3)
SQR(*a*)	Returns square root of number. X = SQR(A+7*C)
TAN(*r*)	Returns tangent of angle given in radians. Z = TAN(2*A3)

TIMER	Returns current value of timer, or allows setting of timer.
	T = TIMER
	TIMER = Ø

USR(*n*)	Calls user defined machine language routine. F = USR(D)
VAPTR(*var*)	Returns address of pointer to specified variable, Z = USR(VARPTR(F))

DRAGON 32

QUICK REFERENCE GUIDE

ERROR CODES

/Ø	Division by zero
AO	File already open
BS	Bad subscript
CN	Can't continue
DD	Attempt to redimension array
DS	Direct statement in file
FC	Illegal function call
FD	Faulty data
FM	Wrong file mode
ID	Illegal direct statement
IE	Input past end of file
I/O	Input/Output error
LS	String too long
NF	NEXT without FOR
NO	File not open
OD	Out of data in READ
OM	Out of memory
OS	Out of string space
OV	Overflow, number too large
RG	RETURN without GOSUB
SN	Syntax error
ST	String formula too complex
TM	Type mismatch
UL	Undefined line

ON DISKS (NOT ATTACHED)
UF UNDEFINED FUNCTION

GRAPHICS STATEMENTS

CIRCLE$(x,y)r,c,h,s,e$.	Draws a circle with centre at point (x,y) of radius r and colour c. Height/width ratio h. (Circle can start at s and end at e (Ø to 1:Ø is 3 o'clock). CIRCLE(128,96),25 CIRCLE(1ØØ,5Ø),3Ø,4,1,Ø,Ø.5
COLOR *foreground,background*	Sets foreground and background colours from available colour set. COLOR Ø,5
DRAW *string*	Draws lines according to instruction held in *string* , as follows. M x,y Move to x,y Un, Up n points Dn, Down n points Ln, Left n points Rn, Right n points En, At 45° n points Fn, At 135° n points Gn, At 225° n points Hn, At 315° n points Ak, Displace by angle k Ø=Ø° 1=9Ø° 2=18Ø° 3=27Ø° Sk, Scale drawing by factor k/4 (k from 1 to 62). N, No update of draw position. B, Blank move next command. C, Set colour of line. X, Execute substring
GET$(x_1,y_1)-(x_2,y_2)$*array*, G	Read graphic contents of screen area defined by rectangle top left (x_1,y_1) bottom right (x_2,y_2) into *array*. G optional - specifier full graphic detail. GET(Ø,Ø)-(24,24),X,G
LINE$(x_1,y_1)-(x_2,y_2),a,b$	Draws line from (x_1,y_1) to (x_2,y_2). If (x_1,y_1) omitted last end point used a must be either PSET (select foreground colour), or PRESET (select background colour). b is optional but must be either B (draw box with corners at (x_1, y_1) and (x_2, y_2) or BF (fill box with foreground colour). LINE(1Ø,2Ø)-(35,15),PSET LINE(Ø,Ø)-(128,96),PSET,BF.
PAINT$(x,y)c,b$	Paints graphic screen starting at point (x,y) with colour c and stopping at border of specified colour b. PAINT(128,96),1,4
PCLEAR n	Reserves n graphics memory pages (n≤8) PCLEAR 6
PCLS c	Clears high resolution screen to colour c. See CLS for colour codes. PCLS 5
PCOPY a TO b	Copies graphics page a to destination page b. PCOPY 3 TO 4
PMODE *mode, page*	Selects resolution mode (from Ø to 4) and first memory page (from 1 to 8) PMODE 4,1
PRESET (x,y)	Resets point (x,y) to background colour PRESET (15,37)
PSET (x, y,c)	Sets point (x,y) to colour c. PSET (120,95,4)
PUT$(x_1,y_1)-(x_2,y_2)$,*array,action*	Places graphic detail from *array* onto rectangle on screen defined by top left (x_1,y_1) bottom right (x_2,y_2) *action* is optional and maybe PSET, PRESET, AND OR or NOT. PUT(A,C)-(A+24,C+24),
RESET(x,y)	Sets point (x,y) on low resolution screen to background colour. RESET (1Ø,12)
SCREEN *type, set*	Selects screen *type* (text = Ø, graphics = 1) and colour *set* (Ø or 1). SCREEN 1,1
SET(x,y,c)	Sets point (x,y) on low resolution screen to colour . SET (15,3Ø,8)

MATHEMATICAL AND LOGICAL OPERATORS

Symbol	Operation	Precedence
↑	Exponentiation	1
–	Unary minus	2
*	Multiplication	3
/	Division	3
+	Addition	4
–	Subtraction	4
>	Greater than	5
<	Less than	5
=	Equal to	5
<>	Not equal to	5
> =	Greater than or equal to	5
< =	Less than or equal to	5
NOT	logical NOT	6
AND	logical AND	6
OR	logical OR	6

The only mathematical operator that can be used with strings is +, the concatentation operator.

Relational operators may be used to compare strings as well as numeric items.

CONTROL KEYS

[←]	Backspace. Cancels last character.
[SHIFT][←]	Erases current line
[BREAK]	Interrupts anything in progress and returns control to keyboard
[CLEAR]	Clears the screen
[ENTER]	Carriage return, end of current input line
[SHIFT][@]	Causes executing program to pause, restart by pressing any key
[SHIFT][Ø]	Upper/lower case switch
[SPACEBAR]	Enter blank character

BASIC LANGUAGE STATEMENTS

CLEAR *n,h*
Reserves *n* bytes of string storage space and erases all variables. *h* specifies highest BASIC address.
CLEAR 5ØØ

CLS c
Clears display to specified colour c.
Ø – Black
1 – Green
2 – Yellow
3 – Blue
4 – Red
5 – Buff
6 – Cyan
7 – Magenta
8 – Orange

DATA
Stores data in program for use by READ statement
DATA 4,7,14.2,SUBJECT

DEF FN
Defines user numeric function.
DEF FNA(X) = X*X+3*X

DEFUSR *n*
Definite entry point for USR function *n*, n=Ø-9.
DEFUSR2 = 14ØØØ

DIM
Dimensions one or more arrays.
DIM X(4Ø),A$(7,6),B(1Ø,2)

END
Terminates program execution.
END

EXEC *address*
Transfers control to machine language programs at *address*. If *address* is omitted control last set in CLOADM.
EXEC 45Ø43

FOR TO STEP NEXT
Creates a program loop which is executed, for specified range of values, STEP indicates the increment. If STEP omitted, one is used.
FOR X=1 TO 1Ø...NEXT X
FOR A=-1.3 TO 7.6 STEP 0.1...NEXTA
FOR G=5Ø TO 1Ø STEP-1Ø...NEXTG

GOSUB
Calls subroutine beginning at specified line number
GOSUB 5ØØ

GOTO
Causes immediate program branch to specified line number.
GOTO 45

IF *condition* THEN *action 1* ELSE *action 2*
Tests *condition* if true performs *action 1*, if false performs *action 2*.
IF A<3 THEN 2ØØ
IF B-C>ØTHEN X=X+1 ELSE Y=Y-1

INPUT
Causes program to halt for entry from keyboard.
INPUT"ENTER NAME";N$
INPUT A,B,C,D

LET
Assigns value to variable (optional).
LET X = 47.4

LINE INPUT
Allows input of line from keyboard, including commas. Line is terminated by [ENTER].
LINE INPUT"TITLE";T$

ON...GOSUB
Multiway branch to specified lines.
ON I GOSUB 1ØØ,2ØØ,3ØØ

ON...GOTO
Multiway branch to specified lines.
ON K GOTO 245,187,31Ø

POKE *location,value*
Places *value* in specified memory *location*. *value* must be Ø – 255.
POKE 27852,Ø

PRINT
Prints contents of following list on screen. Comma (,) causes tab to next 16 column print zone. Semi colon (;) holds print head position.
PRINT"THE ANSWER"
PRINT A,B
PRINT"YOU'VE HAD";T;"TRIES"

PRINT TAB
Moves cursor to specified column position.
PRINT TAB (1Ø);"CREDITS"

PRINT USING
Prints output in specified format
# number field
$ dollar sign in front of number.
** fills leading spaces with asterisks.
↑↑↑↑ prints in exponential format.
% *spaces* % string field length *spaces* + 2.
+ causes sign to be printed.
PRINT USING "##.##";A,B
PRINT USING "%   %";A$

PRINT @ *location*
Prints at specified screen *location*. (Ø-511)
PRINT @ 8,"PAGE";N

READ
Assigns the next item in a DATA statement to specified variable.
READ A,B,C$

REM
Allows comments to be inserted in a program. Everything in a line following REM is ignored.
REM THIS IS A COMMENT LINE

RESTORE
Resets the data pointer back to the first item in the first DATA statement.
RESTORE

RETURN
Returns program from subroutine to the statement following GOSUB.
RETURN

STOP
Halts execution of program at line containing STOP. Use CONT to continue execution
STOP

SOUND GENERATION STATEMENTS

PLAY *string*
Plays music *string* made of following.
A–G (or 1–12), note
O*n*, octave *n*=1 to 5
V*n*, volume *n*=1 to 31
L*n*, note length *n*=1 to 255
T*n*, tempo *n*=1 to 255
P*n*, pause *n*=1 to 255
XA$; execute substring in A$.
Also allows sharps (#) or flats (–)
PLAY"O3L2GBO4CXY$;"

SOUND *tone,duration*
Sounds specified *tone* (1 to 255) for specified *duration*.
SOUND 18Ø,5

CASSETTE RECORDER CONTROL STATEMENTS

AUDIO	Connects or disconnects cassette output to TV speaker. AUDIO ON AUDIO OFF
CLOAD	Loads program file from cassette. First file encountered will be loaded, unless program name is specified. CLOAD CLOAD"NAME"
CLOADM	Loads machine language program from cassette. Offset to loading address may be given. CLOADM CLOAD"NAME" CLOADM"NAME",25ØØ
CLOSE	Closes open files CLOSE CLOSE #-1
CSAVE	Saves program on cassette (program name must be maximum of 8 characters). If A specified, program saved in ASCII format. CSAVE"NAME" CSAVE"NAME",A
CSAVEM *name,start,end,length*	Save machine language program onto cassette. CSAVEM"NAME",4E,6F,5F
EOF(-1)	Returns TRUE if end of file on cassette has been read. IF EOF(-1) THEN 48Ø
INPUT #-1	Inputs data from cassette. File must be open. INPUT #-1,A,B,C
MOTOR	Turns cassette motor on or off. MOTOR ON MOTOR OFF
OPEN *a*,#-1, *filename*	Opens channel to cassette recorder, *a* is either "I" (input) or "O" (output). *filename* is data file name. OPEN"I",#-1,"DATA2"
PRINT #-1	Writes data to cassette. File must be open. PRINT #-1,X9,LN(J)
SKIPF	Skips to end of specified program on cassette. SKIPF"NAME"

PRINTER CONTROL STATEMENTS

LLIST	Prints specified program lines in printer. LLIST LLIST 1Ø-95
OPEN"O",#-2,*filename*	Opens output channel to printer OPEN"O",#-2,"OUTPUT"
PRINT #-2	Prints list of items to printer. PRINT#-2,W,A$;X

SYSTEM COMMANDS

CONT	Continues program execution after a STOP statement or pressing the BREAK key.
DEL	Deletes program lines DEL 1ØØ - 35Ø DEL 1Ø - DEL - 8Ø
EDIT	Used to alter the contents of specified line. EDIT 115 [ENTER] Once in EDIT mode the following subcommands may be used. *n*C Change *n* characters *n*D Delete *n* characters I Insert characters H Delete rest of line and insert characters L List current state of line *n*Sc Search for *n*th occurrence of c X Extends the line *n* ← Moves cursor spaces to left [SHIFT][↑]Escape from subcommand *n*[SPACEBAR] Moves cursor *n* spaces to right
LIST	Displays specified lines on screen LIST LIST 1Ø - 95 LIST - 2ØØ
NEW	Clears program from memory
RENUM	*newline, startline, increment.* Allows all or part of program lines to be renumbered. RENUM RENUM 1ØØ,5Ø,1Ø RENUM,,2Ø
RUN	Executes a program starting at lowest line or specified line RUN RUN 25Ø
TROFF	Turns off program flow trace TROFF
TRON	Turns on program flow trace

SIMPLE VARIABLES

Type	Name	Range
Numeric	AB	± 10^{38}
String	AB$	Ø to 255 characters

Where A must be a letter, B is optional and may be a letter or a digit. If name used is longer than two characters only the first two are significant.

ARRAY VARIABLES

Type	Name of typical element
Numeric	AB(3,4,5)
String	AB$(17,8,2)

Names and ranges are as for simple variables. Array size is limited by available memory space.

SPECIAL CHARACTERS

'	Abbreviation for REM
?	Abbreviation for PRINT
:	Separates statements on the same line.
$	Appended to variable name makesit string type.