

## Literals / Variables

### Literals

Strings are contained in double quotes, e.g. "Maximite".  
 Numbers may be decimal or represented as:  
 &Hnn Hex Literal, e.g. &H3C (60)  
 &Bnn... Binary Literal, e.g. &B00100011 (35)  
 n.nE+n Scientific, e.g. 1.6E+4 (16000)

### System Variables

MM.HRES Horizontal Screen Resolution  
 MM.VRES Vertical Screen Resolution  
 MM.VER Firmware Version  
 MM.DRIVE\$ Current Default Drive  
 MM.FNAME\$ Current Default File  
 MM.ERRNO Last Error Code  
 MM.I2C Last I2C Result Code

### User Variables

Variable names start with an alpha character or underscore and can contain any alpha or numeric character, period (.) and underscore (\_); maximum length is 32 characters. String variable names are terminated with a \$ symbol. Number variable names are not terminated with a \$ symbol.

## Operators

### Arithmetic

^ \* / Exponentiation, Multiplication, Division  
 MOD \ Modulus (remainder), Integer Division  
 + + - Addition, String Concatenation, Subtraction

### Logical

NOT Logical inverse  
 = <> Equality, Inequality  
 > < Greater Than, Less Than  
 <= or =< Less Than Equal To  
 >= or => Greater Than Equal To  
 AND OR Conjunction, Disjunction,  
 XOR Exclusive OR

## Codes

### Error

0 No error  
 1 No SD card found  
 2 SD card is write protected  
 3 Not enough space

4 All root directory entries are taken  
 5 Invalid filename  
 6 File does not exist  
 7 Directory does not exist  
 8 File is read only  
 9 Cannot open file  
 10 Error reading from file  
 11 Error writing to file  
 12 Not a file  
 13 Not a directory  
 15 Directory not empty / Hardware error

### Format String

#### % [flags] [width] [.prec] type

flags: - Left justify  
**0** Use 0 for the pad char, not space.  
**+** A plus sign is shown for positive values.  
**space** Space as sign, unless negative.  
 width: min. chars to output, less than this causes padding, more than this causes expansion.  
 prec: no. of fraction digits for **e**, or **f** type, or the max. no of significant digits for **g** type. Must be preceded by a dot (.) if used.  
 type: **g** format for the best presentation  
**f** format with decimal point and digits  
**e** Format in exponential format  
**G** exponential output with uppercase E  
**F** exponential output with uppercase E.

If the format specification is not specified "%g" is assumed.

### I2C

0 No error  
 1 Received NACK response  
 2 Command timed out  
 4 Received general call addr. (slave mode)

### File Open Mode

INPUT Read Only  
 OUTPUT Write (Overwrite if exists)  
 APPEND Write (Append if exists)

## Commands / Statements

### Assignment

CLEAR  
 DATA  
 DIM variable(elements...)  
 ERASE variable

LET variable =  
 READ variable[, variable]...  
 RESTORE

### Editor

AUTO [start] [, increment]  
 DELETE line  
 DELETE -lastline  
 DELETE firstline [- lastline]  
 EDIT [line-number] (*extended editing mode*)  
 LIST [line]  
 LIST -lastline  
 LIST firstline [- lastline]  
 RENUMBER [first] [,incr] [,start]

### External Pins

PIN(pin) = value  
 SETPIN pin, cfg  
 SETPIN pin, cfg, line

### File System

CHDIR dir\$  
 CLOSE [#]nbr [, [#]nbr]  
 CLOSE CONSOLE  
 DRIVE drive\$  
 FILES [search\_pattern\$]  
 INPUT #nbr, list of variables  
 KILL file\$  
 LINE INPUT #nbr, string-variable\$  
 LOAD file\$  
 MERGE file\$  
 MKDIR dir\$  
 NAME old\$ AS new\$  
 OPEN fname\$ FOR mode AS [#]fnbr  
 OPEN comspec AS [#]fnbr  
 ? or PRINT #nbr, expression [, [,]expression]...  
 RMDIR dir\$  
 SAVE [file\$]  
 SAVEBMP file\$  
 WRITE [#nbr,] expression [, expression]

### Flow Control

CONTINUE  
 DO <statements> LOOP  
 DO WHILE expression <statements> LOOP  
 DO <statements> LOOP UNTIL expression  
 ELSE  
 ELSEIF expression THEN  
 ENDIF

## Maximite Quick Reference – v2.7a

END  
 EXIT  
 EXIT FOR  
 FOR counter = start TO finish [STEP increment]  
 GOSUB  
 GOTO  
 IF expression THEN  
 IRETURN  
 NEXT [counter-variable] [, counter-variable]...  
 ON variable GOTO|GOSUB line[,line,line,...]  
 PAUSE nbr  
 RETURN  
 WHILE expression...<statements>WEND

### I2C

I2CEN speed, timeout [,int\_line]  
 I2CDIS, I2CSDIS  
 I2CSEND addr, option,sendlen, data [,data....]  
 I2CRCV addr,hold,rcvlen,rcvbuf [,sendlen,data [,data....]]  
 I2CSEN addr, mask,option, send\_int\_line,rcv\_int\_line  
 I2CSSEND sendlen,data [,data....]  
 I2CSRCV rcvlen, rcvbuf, rcvd  
 NUM2BYTE number,var1, var2,var3, var4  
 NUM2BYTE number,array(x)

### Keyboard

INPUT ["prompt string";]  
 LINE INPUT [prompt\$,], string-variable\$

### Miscellaneous

COPYRIGHT  
 DATE\$ = "DD-MM-YY" or "DD/MM/YY"  
 ERROR [error\_msg\$]  
 MEMORY  
 OPTION BASE 0 | 1  
 OPTION ERROR ABORT | CONTINUE  
 OPTION Fnn str\$  
 OPTION PROMPT str\$  
 OPTION USB ON | OFF  
 OPTION VIDEO ON | OFF  
 POKE hiword, loword, val  
 RANDOMIZE nbr  
 REM string  
 ' Comment  
 RUN [line] [file\$]  
 NEW  
 SETTICK period, line  
 TIME\$ = "HH:MM:SS"

TIMER = msec  
 TROFF / TRON

### Screen

CLS  
 CIRCLE (x, y) ,r [,c [,F]]  
 FONT #nbr [, scale] [, reverse]  
 FONT LOAD file AS #nbr  
 FONT UNLOAD #nbr  
 LINE [(x1 , y1)] - (x2, y2) [,c [,B[F]]]  
 LOCATE x, y  
 PIXEL(x,y)  
 PRINT / ? expression [[:]expression]...  
 PRESET (x, y)  
 PSET (x, y)

### Serial I/O

CLOSE CONSOLE  
 OPEN COMSPEC as CONSOLE  
 XMODEM SEND file\$  
 XMODEM RECEIVE file\$

### Sound/PWM

SOUND frequency, duration  
 SOUND frequency, duration, dutycycle

## Functions

### Date, Time, Timer

DATE\$  
 TIME\$  
 TIMER

### File System

CWD\$  
 EOF ([#]nbr)  
 INPUT\$ (nbr, [#]fnbr)

### I2C

BYTE2NUM(array(x))  
 BYTE2NUM(arg1, arg2,arg3, arg4)

### I/O

LOC ([#]nbr)  
 LOF ([#]nbr)  
 PIN (pin)  
 SPI (rx, tx, clk [,data [,speed] ])

### Math / Number

ABS (nbr)

ATN (nbr)  
 CINT (nbr)  
 COS (nbr)  
 EXP (nbr)  
 FIX (nbr)  
 HEX\$ (nbr)  
 INT (nbr)  
 LOG (nbr)  
 OCT\$ (nbr)  
 RND (nbr)  
 SGN (nbr)  
 SIN (nbr)  
 SQR (nbr)  
 STR\$ (nbr)  
 TAN (nbr)

### Memory

PEEK (hiword,loword)

### Screen

POS  
 PIXEL(x,y)

### String / Character

ASC (str\$)  
 CHR\$ (nbr)  
 FORMAT\$ (nbr [,format\$])  
 INSTR ([start,] search\$, pattern\$)  
 LEFT\$ (str\$, nbr)  
 LEN (str\$)  
 LCASE\$ (str\$)  
 MID\$ (str\$, start [,nbr])  
 RIGHT\$ (str\$, nbr)  
 SPACE\$ (nbr)  
 SPC (nbr)  
 STRING\$(nbr, val|str\$)  
 TAB(nbr)  
 UCASE\$ (str\$)  
 VAL (str\$)  
 INKEY\$

## Acknowledgements

MMBasic is Copyright 2011 by Geoff Graham

For the full Maximite User Manual and more details on the Maximite go to <http://geoffg.net/maximite.html>