BASIC Commands for TTGO T7 v1.4 Mini32

## Structured Programming

|  |  |  |  |
| --- | --- | --- | --- |
| **Command** | **Description** | **Parameters** | **Example** |
| LET | Assign a value to a variable. | variable, value | LET A = 10 |
| FOR ... TO ... STEP ... NEXT | Loop with a counter. | start, end, step | FOR I = 1 TO 10 STEP 2 : NEXT I |
| GOTO | Jump to a specific line number. | line number | GOTO 100 |
| GOSUB ... RETURN | Call and return from subroutines. | line number | GOSUB 200 |
| END | Terminate the program. | None | END |
| STOP | Halt execution. | None | STOP |
| IF ... THEN ... ELSE | Conditional branching. | condition, statement | IF A > 10 THEN PRINT 'Large' ELSE PRINT 'Small' |
| REM | Add comments. | comment text | REM This is a comment |
| WHILE ... WEND | Loop until a condition is false. | condition | WHILE A < 10 : PRINT A : WEND |
| REPEAT ... UNTIL | Loop until a condition is true. | condition | REPEAT PRINT A : UNTIL A > 10 |
| DO ... DEND | Group multiple statements. | None | DO PRINT "Hello" : DEND |
| SWITCH ... CASE ... SWEND | Case-based branching. | value, cases | SWITCH A CASE 1 : PRINT "One" SWEND |
| AFTER | Execute a command after a delay. | milliseconds, command | AFTER 1000, PRINT "Done" |
| EVERY | Repeat a command periodically. | milliseconds, command | EVERY 500, PRINT "Tick" |

## General Utility Commands

|  |  |  |  |
| --- | --- | --- | --- |
| **Command** | **Description** | **Parameters** | **Example** |
| PRINT | Output text or numbers. | expression(s) | PRINT "Hello, World!" |
| SIZE | Get free memory space. | None | PRINT SIZE |
| NEW | Clear the program in memory. | None | NEW |
| RUN | Execute the program. | Optional start line | RUN |
| CLS | Clear the screen. | None | CLS |
| NETSTAT | Display current network status. | None | NETSTAT |
| MILLIS | Get system uptime in milliseconds. | None | MILLIS |

## File, Memory, and Array Commands

|  |  |  |  |
| --- | --- | --- | --- |
| **Command** | **Description** | **Parameters** | **Example** |
| SAVE | Save a program to storage. | filename | SAVE "PROGRAM.BAS" |
| LOAD | Load a program from storage. | filename | LOAD "PROGRAM.BAS" |
| PEEK | Read a memory address. | address | PRINT PEEK(1024) |
| POKE | Write to a memory address. | address, value | POKE 1024, 255 |
| CATALOG | Lists files, optionally filtered by a string pattern. | Optional filter | CATALOG "abc" |
| DELETE | Deletes a file on the SD card. | filename | DELETE "prog.bas" |
| FDISK | Format the SD card. | None | FDISK |
| DIM | Declare an array. | array, size | DIM A(10) |

## I/O, Networking, and Arduino-Specific Commands

|  |  |  |  |
| --- | --- | --- | --- |
| **Command** | **Description** | **Parameters** | **Example** |
| PINM | Set pin mode. | pin, mode | PINM 13, OUTPUT |
| DWRITE | Digital write to pin. | pin, value | DWRITE 13, HIGH |
| DREAD | Digital read from pin. | pin | PRINT DREAD(13) |
| AWRITE | Analog write to pin (PWM). | pin, value | AWRITE 11, 128 |
| AREAD | Analog read from pin. | pin | PRINT AREAD(0) |
| DELAY | Pause execution for a duration. | milliseconds | DELAY 1000 |
| TONE | Play a tone on a pin. | pin, frequency, duration | TONE 8, 440, 500 |
| INPUT | Accept user input. | variable(s) | INPUT A |

## Graphics and Display Commands

|  |  |  |  |
| --- | --- | --- | --- |
| **Command** | **Description** | **Parameters** | **Example** |
| PLOT | Plot a point on the screen. | x, y | PLOT 10, 20 |
| LINE | Draw a line between two points. | x1, y1, x2, y2 | LINE 10, 10, 20, 20 |
| RECT | Draw a rectangle. | x1, y1, x2, y2 | RECT 5, 5, 15, 15 |
| FRECT | Draw a filled rectangle. | x1, y1, x2, y2 | FRECT 5, 5, 15, 15 |
| CIRCLE | Draw a circle. | x, y, radius | CIRCLE 10, 10, 5 |
| FCIRCLE | Draw a filled circle. | x, y, radius | FCIRCLE 10, 10, 5 |
| COLOR | Set the color for graphics. | color code | COLOR 2 |