

FloatSort

This is an embedded C module which will sort an array of floating point numbers (floats) into ascending numerical order. It runs much faster than a BASIC program and was written by Peter Mather of the Back Shed forum.

Note that this will only work on the standard Micromite which uses single precision floating point. It will not work on the Micromite Plus which uses double precision floating point.

Adding FloatSort to MMBasic

To add the FloatSort function to MMBasic you must insert the following code somewhere in your BASIC program (you can use copy and paste from this document). The exact spot is not important but at the end of the program is typical.

```
CSub FloatSort
00000000
27BDFFA8 AFBF0054 AFBE0050 AFB7004C AFB60048 AFB50044 AFB40040 AFB3003C
AFB20038 AFB10034 AFB00030 AFA40058 8CA50000 AFA50018 00A09021 10000034
3C159D00 8FA20020 8C5E0000 8FA3001C 0072102A 14400019 0060B821 8FB30020
8FB10028 10000009 8FB00024 AE620000 8FA40014 02048021 02368821 02901021
0052102A 1440000C 02769821 0214B821 AFB00010 8EA20068 03C02021 0040F809
8E250000 2403FFFF 5043FFF0 8E220000 10000003 0017B880 8FB70010 0017B880
8FA40058 0097B821 AEFE0000 8FA2001C 24420001 AFA2001C 8FA30024 24630001
AFA30024 8FA40020 24840004 AFA40020 8FA20028 24420004 AFA20028 8FA30018
8FA4001C 0083102A 1440FFCF 8FA20020 001217C2 00529021 00129043 12400010
8FA30018 0243102A 1040FFFA 001217C2 0240A021 00121080 8FA40058 00821021
AFA20020 AFA40028 AFB2001C AFA00024 00121023 AFA20014 1000FFBA 0002B080
00001021 00001821 8FBF0054 8FBE0050 8FB7004C 8FB60048 8FB50044 8FB40040
8FB3003C 8FB20038 8FB10034 8FB00030 03E00008 27BD0058
End CSub
```

Parameters

The FloatSort command (created by adding the above code) takes two parameters:

FloatSort ArrayOfFloat!(), NbrOfFloats

Where:

ArrayOfFloat = The array of floating point numbers to be sorted. Note that it is passed with empty brackets (ie, ArrayOfFloat!).

NbrOfFloats = The number of floating point numbers to be sorted (ie, the number of elements in the array). Note that unless the command OPTION BASE is used an array will start with an index of zero and contain one more element than that specified in the DIM command.

Using the Command

This example creates a small array of three floating point numbers, populates the array and then uses FloatSort to sort the array into numeric order:

```
DIM N!(2)
N!(0) = 100.56
N!(1) = 5.4
N!(2) = 50.27
FloatSort N!(), 3
```

Note that OPTION BASE was not used which means that the base for an array will default to zero which in turn means that the array will start with an index of zero and contain one more element than that specified in the DIM command.