

MM Universal Uploader

The ZIP has a single exe file. Put it anywhere convenient but it does require write permission so not in Program Files etc. An inf file gets created remembering the window position and the last serial port used.

Getting your code into the uploader.

Create a shortcut on the desktop and Drag a file from Windows Explorer.

Or

With the program running,

Drag a file from Windows Explorer.

Highlight code in your editor and drag onto the program's text window.

Copy the code into the clipboard and right-click in the program text window.

Provided you have a serial port selected, the upload will start immediately.

If there is no serial port, you will be prompted to select a port then press "Program" The "program" button is usually hidden.

Alternatively, you can specify the serial port and speed in the *'onload* line of the BAS file.

Eg *'onload* com7:38400 will set the port to com7 and speed to 38400.

If the first 5 characters of the file are "MMHEX" we have an encoded file and it is decoded before uploading.

If the first line is

'encode

The code is encoded, not uploaded. You will get the ability to save the encoded file ready for distribution.

If the first line is

' encode crunch

The code is crunched before encoding.

Crunching does not remove any *'echo ...* lines

Otherwise, the code is uploaded

Any line starting with

'echo

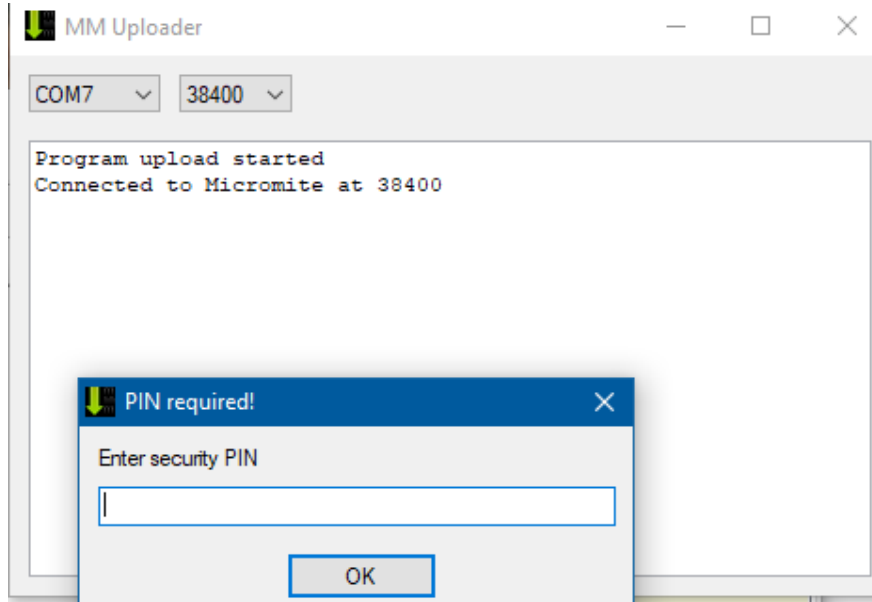
is sent to the text window and NOT uploaded.

This lets you send messages to the user.

If the micromite has a PIN set, you are given the opportunity to enter the PIN

You only get one go. If you get it wrong, you have to start from the beginning again.

If you have a device that doesn't use AUTOSAVE and ctrl-Z, you need to specify the device in the *'onload* options.



```
'encode crunch
'onload settime norun
'OPTION PIN 0
'echo Program version 99.9
DO
  FOR n = 1 TO 20 'comment
    PRINT n
  NEXT n
LOOP
' spare lines
' more comments

'echo Hi there, you are up and running again!
```

The first line starts with *'encode* so that is what happens'.

If the line also contains crunch as in the example, the file is crunched before encoding.

The first line gets discarded before encoding.

The next line is the *'onload* options

If you are not encoding, the *'onload* line must be the first line of the program.

there can only be one *'onload* line but you can have multiple options on the line

In the example, the file gets crunched when encoding and when decoded/uploaded, the time is set and the program doesn't run automatically.

Current options are

```
one of: maximate picromite micromite cmm2 command
(default micromite)
```

```
plus any combination of:
crunch norun nonew settime setrtc com file/
```

Commands are processed in the following order, not in the order entered.

```
com          sets the serial port and speed eg com7:38400
              If com is not present, it uses the port previously set
crunch       removes comments and blank lines before uploading
nonew        inhibits the NEW command before loading
settime      sets the time based on the computer time
setrtc       sets the RTC based on the computer time
              using "RTC SETTIME "
```

AUTOSAVE or equivalent is now sent followed by the file or text.
(If the command option is set, the following lines in the file are sent without any AUTOSAVE and assume you are at the command prompt).

For the CMM2

file/ or file\ followed by a filename will use that filename instead of "temp.bas" when uploading. This allows you to upload *.inc or *.fnt files more easily. There is NO space between file/ and the filename. Names with spaces are not handled.

The correct 'end of file' is then sent.

```
norun        inhibits the RUN command after loading
```

The order of options is not important and they can be separated by a space or comma.

Without any options the default is to load uncrunched to a micromite and RUN with

NEW

AUTOSAVE

send file

ctrl-Z

RUN

When you insert or remove a USB serial port, the port number is shown in the text window.

An end-user could start MMUUploader before inserting the cable. Then he/she would know what the port number is.

It also handles BAS file that have the MMEdit header in them. The header is stripped out even if you haven't selected crunch.

I have been experimenting with the command option

```
'onload command
OPTION LCDPANEL ILI9341, RL, 4,5,6

OPTION TOUCH 7,2

GUI CALIBRATE
```

This configures a module I have, something that has to be done after firmware updates. The extra blank lines were needed to give the micromite time to reboot when the display is configured.

GUI CALIBRATE has to be the last command because it takes a long time while you do the calibration.

I intend to have one BAS file with all the different setups I have for the various devices. It is then a case of highlight the one I want and drop it into the uploader.

MMUploader waits a few seconds after the RUN command is sent before disconnecting. That way any immediate messages from the micromite will be captured by MMUploader. It might be useful as a way of confirming the program is running (or capturing any error messages).