

# MMC I/O Development test tool for Windows

**DISCLAIMER:** This application was primarily written for in-house testing purposes only. It was written specifically for the SDMMC development boards programmed with the SDMMC I/O hex firmware. All files are available at <http://www.compsys1.com/support>. The application is provided "as-is" and use at your own risk. COMPSys Workbench is not liable for any harm caused by its use.

## IMPORTANT

**This application along with the MMC I/O microcontroller firmware writes directly to the SD/MMC card. This is not an MS FAT type of I/O. Do not use with a card that is FAT formatted (such as in Windows with a card reader) and has important data on it! MMC and SD cards are made by numerous manufacturers using the SD and MMC standards. However, there may be subtle differences. In some rare cases a particular brand/size of card may fail to initialize on the development board.**

Communications setup for the PC's serial ports. The settings must match the PIC firmware. Default is 19200 baud

Port buffers and time-out settings to adjust I/O if there are problems with incomplete data I/O. Normally the default values will work.

Advanced Settings for tweaking I/O problems. Normally they will not have to be changed. (The "D" and "E" are the "Enable changes" and "Disable changes" buttons). Some parameters are firmware dependent

### I/O options

0 - Read as hexadecimal values

1 - Read as ASCII text

2 - Write as hexadecimal values or text

3 - Erase a given number of blocks starting at a given address

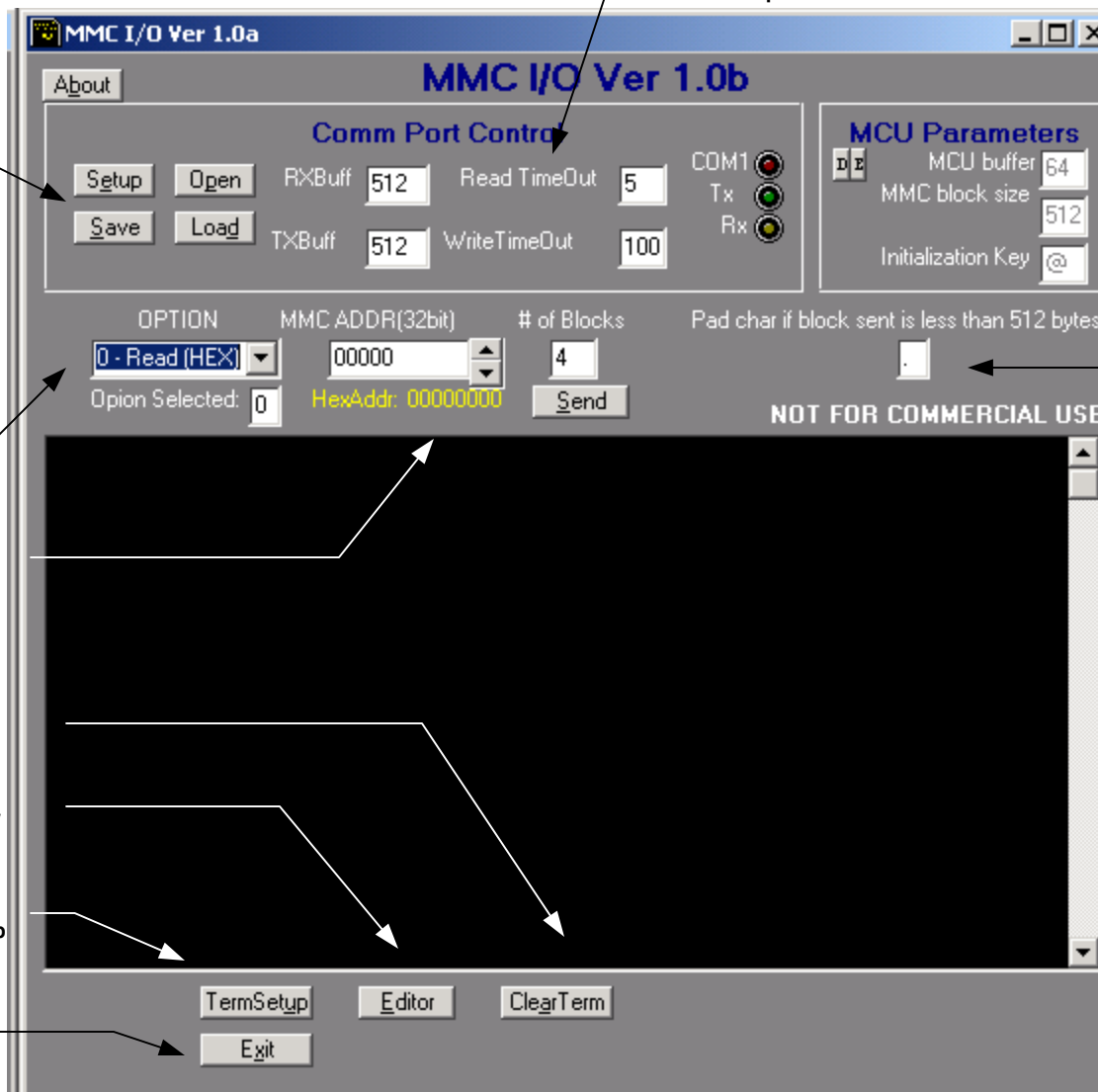
Card I/O address in decimal (incremented by 512 bytes) and the number of blocks (each block is 512 bytes) to be read

Clear the terminal window

Switch to the editor window

Terminal window setup

Exit application



The "padding" character to be used if exactly 512 bytes per block are not sent. You can set this to whatever character you wish.

## Reading from the card

Open the MMC I/O application and select the appropriate com port on the PC (default is com 1) and click the "Open" button.

Connect the SD board (which is programmed with the SDMMC hex firmware) to the PC's com port and place an SD/MMC card in the connector, apply power to the board.

If the board can initialize the card you should see the "Waiting.." message. If it hangs on the "Start" message it means that it could not initialize the card. Try recycling the power on the board or try a different card.

Next, decide whether the data read should be in hexadecimal format (for reading binary data) or in ASCII format (for reading textual data)

The example on the top right will read hex data.

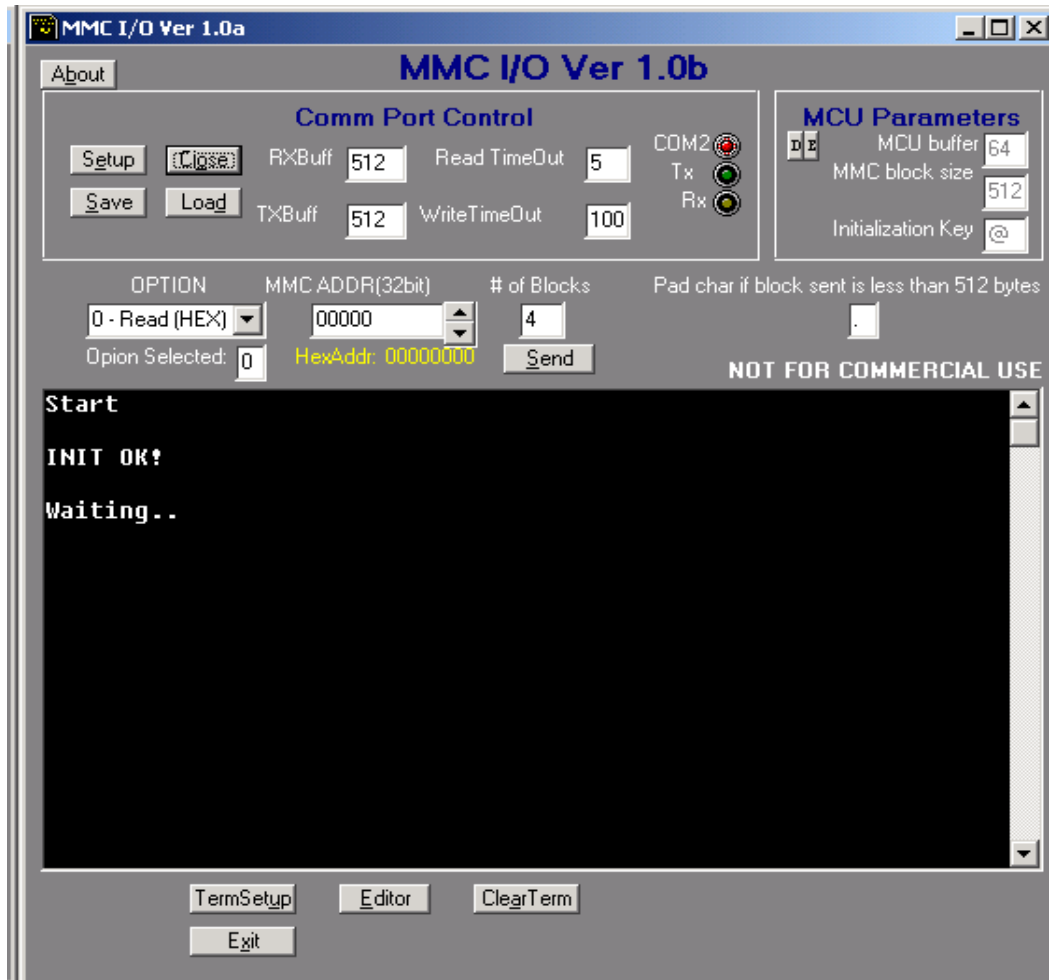
Next, select the start address (which is incremented by 512 bytes). The example is address 00000 (in decimal, the equiv. in hex is shown in yellow)

Select how many blocks (each block has 512 bytes) to read. The example is set to 1 block.

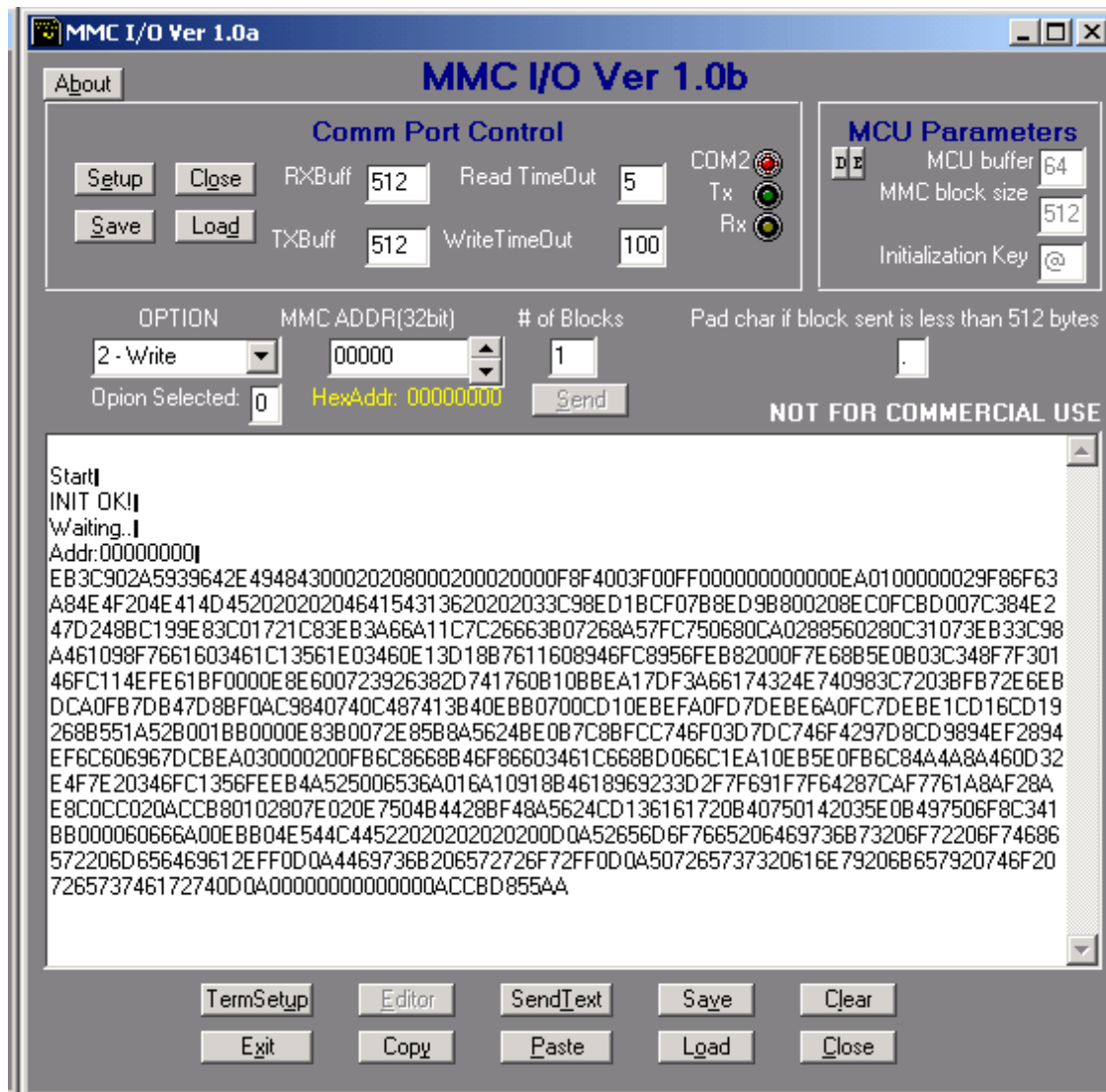
Finally, press the "Send" button. The MMC I/O program will communicate with the board and send the results to the terminal window, as shown on the right.

The data in the terminal window can be collected in the "Editor" window by pressing the Editor button (as shown on the next page).

**Info: A Lexar 64mb SD card was used for this demonstration**



## Reading data and saving it using the Editor



Data read from the card can be edited and saved using the Editor. The data can be copied to memory for pasting or it can be saved as a text file.

# Writing to an SD / MMC card

Select Write from the options box. You will be switched to the Editor automatically. You can clear the Editor by pressing the Clear button.

Enter the starting address in the Address area. 1024 is selected in the example on the right.

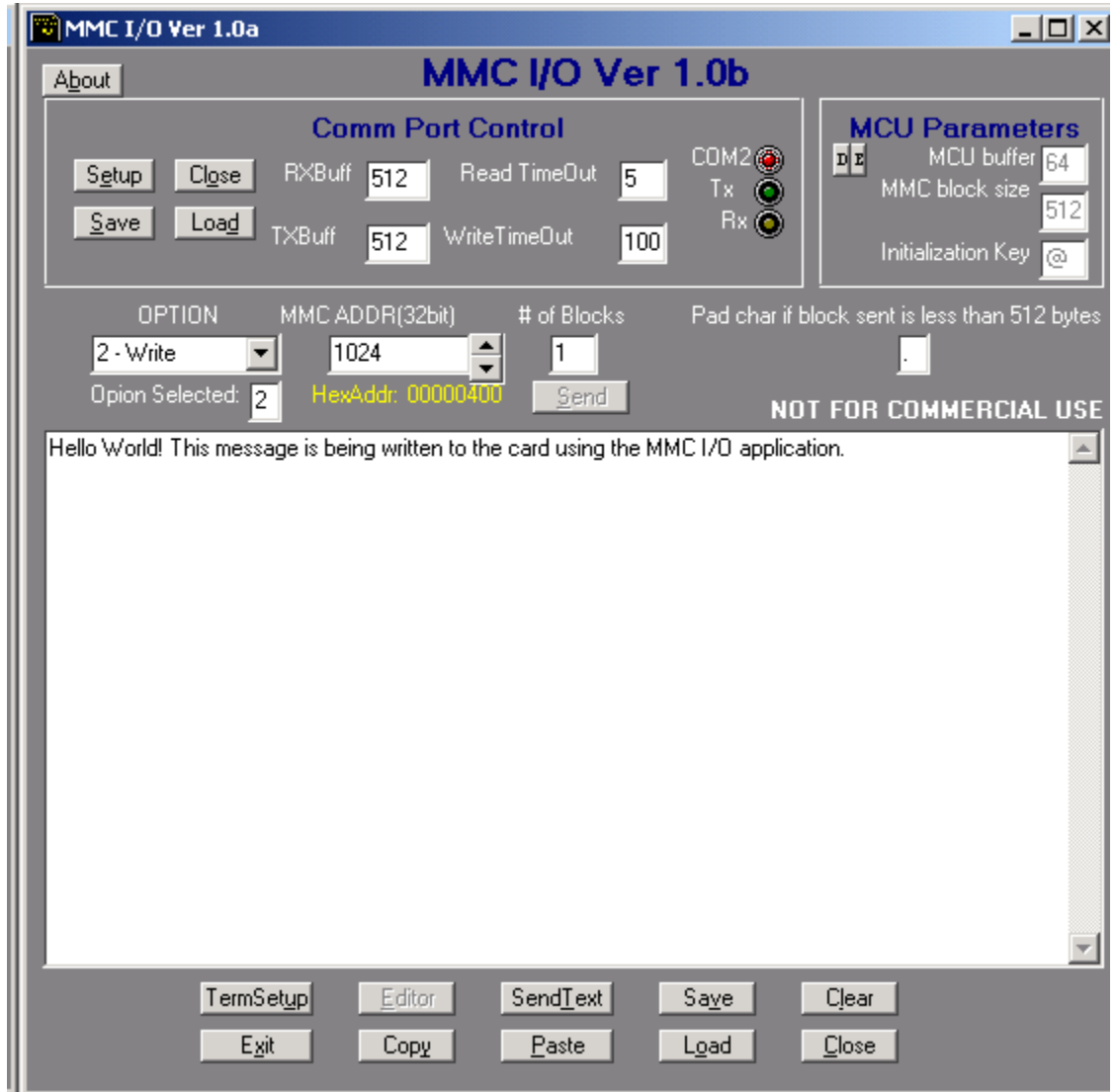
Type in or paste the data that you want to send to the card.

Optionally, enter the "padding" character.

Next, press the SendText button at the bottom of the Editor window. The empty bytes will be filled with the padding character and the data (in 512 byte blocks) will be sent to the card.

You can close the Editor and go back to the main terminal to read the data saved. Make sure you enter the same address that you wrote it to.

Example of the read is shown on the next page.



# Reading back data written (from the previous page)

Make sure that the address is correct and that Option 1 (Read ASC) is selected.

Press the Send button.

The data written previously should now be visible. All blank locations will be filled with the "padding" character (a period in this case).

