



**Future Technology Devices
International Ltd.**

MProg 2 Manual

Table of Contents

Part I MProg Introduction	3
1 Installing MProg.....	5
2 Installing D2XX Drivers.....	10
Part II MProg Basics	14
1 Modes of Operation.....	14
Part III Edit Mode Functions	16
1 Create a New EEPROM Template.....	16
2 Modify an Existing EEPROM Template.....	18
Part IV Device EEPROM Parameters	20
1 Basic Details.....	20
2 USB Power Options.....	21
3 USB Serial Number Control.....	22
4 USB Remote Wake Up.....	23
5 Plug & Play (FT232 Series Only).....	24
6 BM/C Device Specific Options.....	25
7 FT2232C Device Options.....	26
8 FT232R Device Options.....	27
9 FT245R Device Options.....	28
10 Product and Manufacturer Descriptor Strings.....	29
11 Programming Options.....	30
Part V Program Mode Functions	31
1 Scan for Devices.....	32
2 Erase All Existing Devices.....	33
3 Program All Existing Devices.....	34
4 Read Devices.....	35
5 Read and Parse Devices.....	36
Part VI Appendix	37
1 Customising FTDI's VCP Drivers.....	37
Changing the FTDIBUS.INF File.....	37
Changing the FTDIPOINT.INF File.....	38
Changing the VCP Uninstaller INI Files.....	39
2 Customising FTDI's D2XX Drivers.....	40

Changing the FTD2XX.INF File.....	40
Changing the D2XX Uninstaller INI File.....	41
3 Uninstalling MProg.....	42
Index	45

1 MProg Introduction

MProg is an easy to use multiple device EEPROM programming utility for use with FTDI devices. Devices supported by the latest version of MProg are:

- [FT232RL & FT232RQ](#)
- [FT245RL & FT245RQ](#)
- [FT2232C & FT2232L](#)
- [FT232BM, FT232BL & FT232BQ](#)
- [FT245BM, FT245BL & FT245BQ](#)
- [FT8U232AM](#)
- [FT8U245AM](#)

MProg is available as a free download from the [Utilities](#) page of the [FTDI website](#).

These devices support 93C46 EEPROMs organised in 16-bit words, except for the R series devices which have an internal EEPROM. The BM and C devices also support 93C56 and 93C66 EEPROMs organised in 16-bit words. When fitted, an external EEPROM allows customisation of the device default parameters such as [VID](#)^[20], [PID](#)^[20], [Serial Number](#)^[22], [Manufacturer Descriptor](#)^[29] and [Product Descriptor](#)^[29]. MProg provides a fast and simple way of programming these parameters.

A convenient way of programming the EEPROM is to use a blank device on the PCB during manufacturing assembly and program it via USB during the manufacturing test process. All these devices support direct programming over USB using a programming utility such as this one.

MProg - Multi Device EEPROM Programmer V 2.8a

File Device Tools Help

Basic Details

Device Type:

USB VID / PID:

Vendor ID: Product ID:

BM / C Device Specific Options

USB Version Number:

☐ Disable USB Serial Number

☐ Pull Down IO Pins in USB Suspend

USB Power Options

Max Bus Power

☒ Bus Powered ☐ Self Powered

milli Amps

USB Serial Number Control

Serial Number Prefix (2 digits)

☐ Use Fixed Serial Number

Fixed Serial Number (8 digits)

USB Remote Wake Up

☐ Enable USB Remote Wake Up

Plug & Play (FT232 Series Only)

☒ Enable Plug And Play

Product and Manufacturer Descriptor Strings

Manufacturer: Product Description:

Programming Options

☒ Only Program Blank Devices

FT2232C Options FT232R

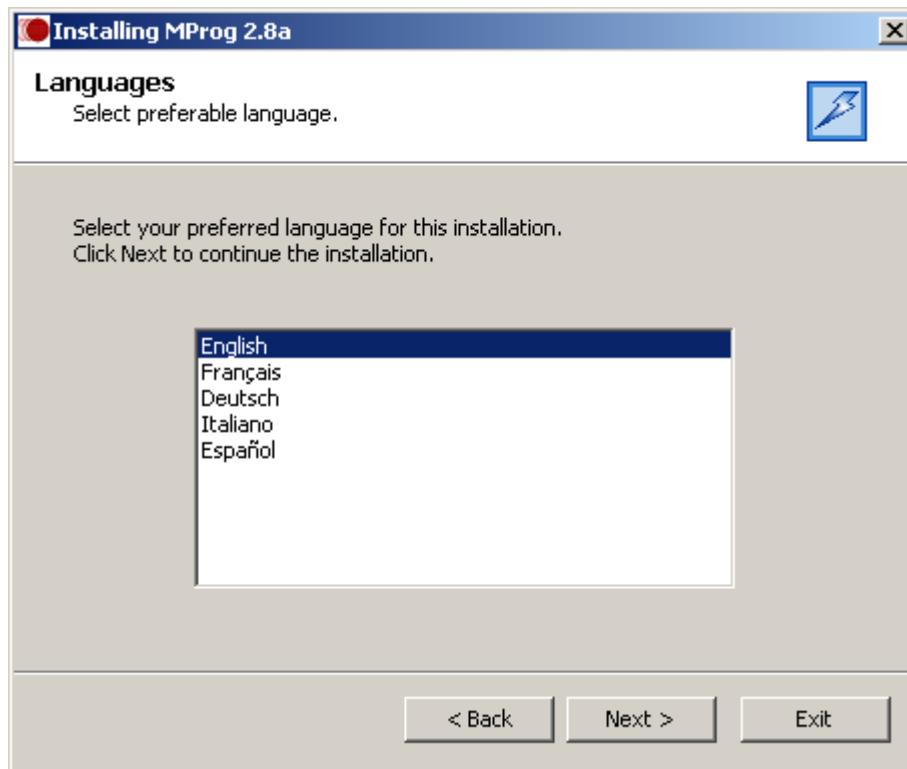
1.1 Installing MProg

Prior to installation, please uninstall any existing FTDI drivers from the target PC using the Add/Remove Programs utility in the Windows control Panel. MProg uses FTDI's D2XX drivers, the current versions of which can be downloaded free of charge from the [Drivers page](#) of the [FTDI website](#). A link to the D2XX drivers page is included in the MProg installation directory.

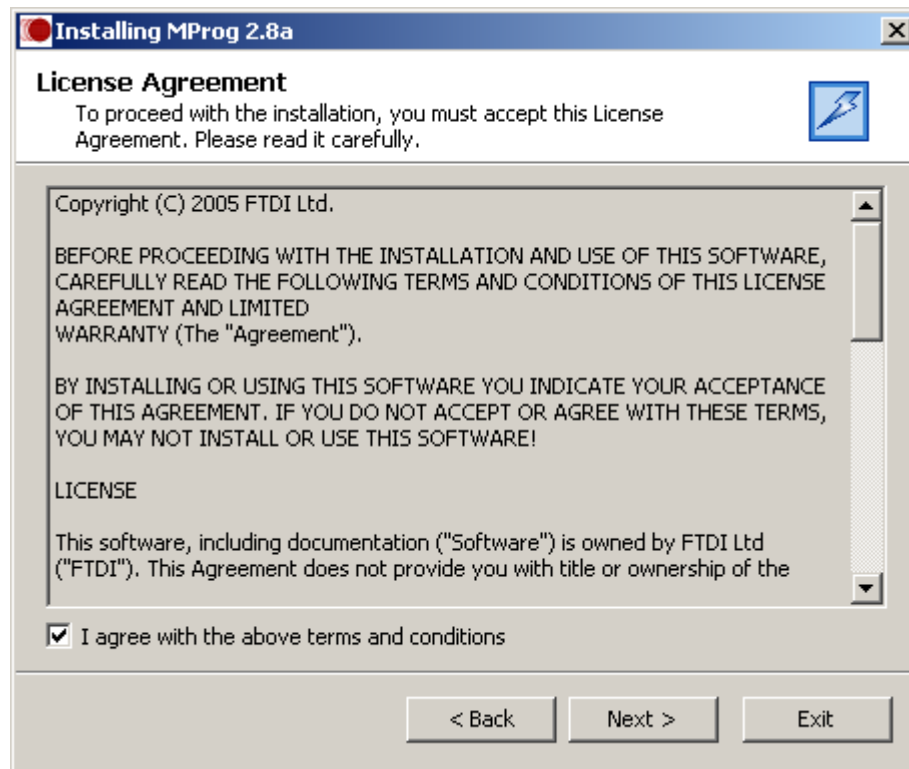
The latest version of MProg can be downloaded from the [Utilities page](#) of the [FTDI website](#) in a zipped setup executable. Extract the file, run the setup executable file and the following screen should appear:



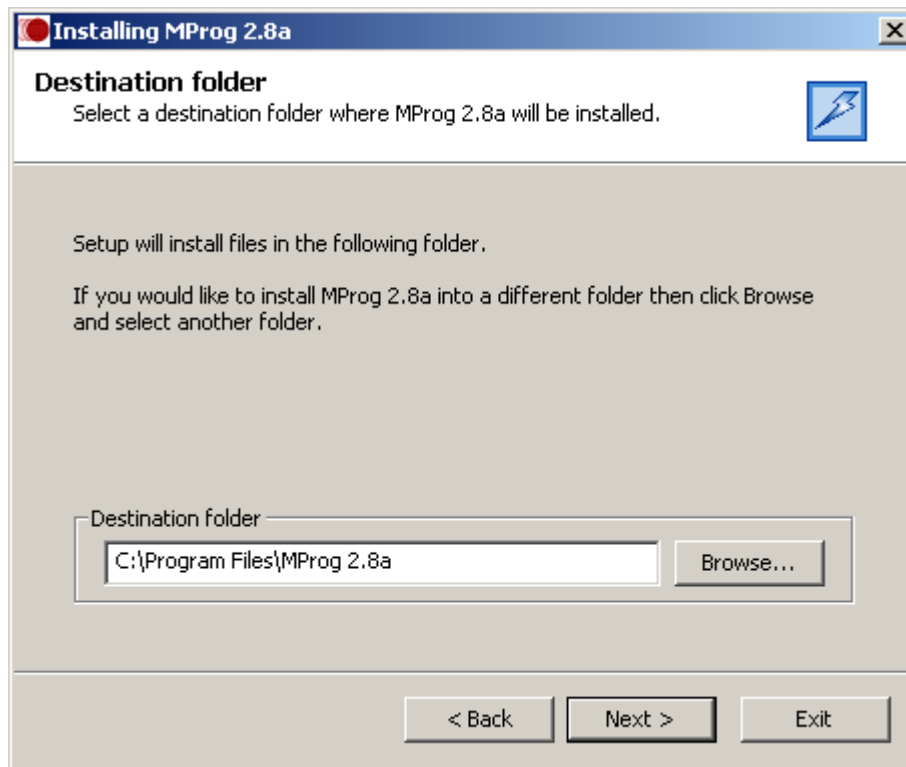
Click Next and the installation program will prompt you to select the language you want to use for the rest of the setup process. The default value is English.



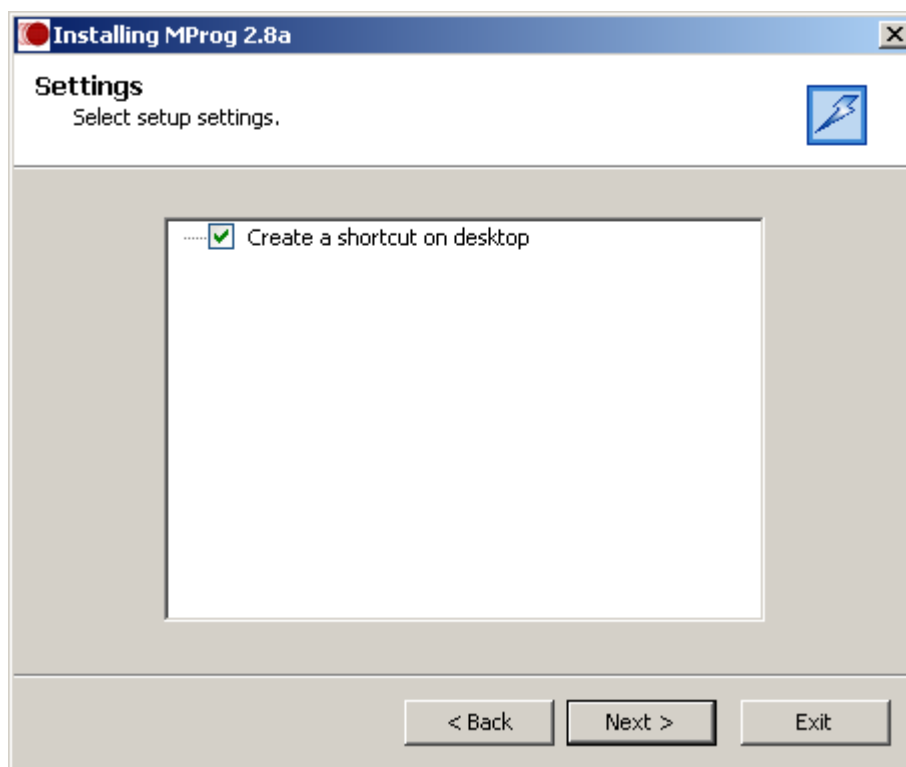
Clicking Next displays the license agreement. Read the license agreement and if you agree, check the box next to "I agree with the above terms and conditions" and click next to proceed with the installation. If you do not agree, the installation process will not continue.



The next screen prompts the user for a location to install MProg to. The default location is "C:\Program Files\MProg" with the MProg version number. Once a location has been selected, click Next to commence installation.



Clicking Next displays a screen which allows the user to select if they would like a desktop shortcut created. If so, check the box and click OK.



Clicking Next installs MProg and copies the necessary files.



This completes the installation of MProg. To start MProg, either follow the desktop shortcut or locate MProg in the Start menu.

Please note that MProg requires D2XX drivers to be installed before devices can be programmed. See [Installing D2XX Drivers](#) ¹⁰ for details of driver installation.

1.2 Installing D2XX Drivers

To install D2XX drivers for FTDI devices, follow the instructions below. The screen shots are from a Windows 2000 installation, but other Windows versions are very similar.

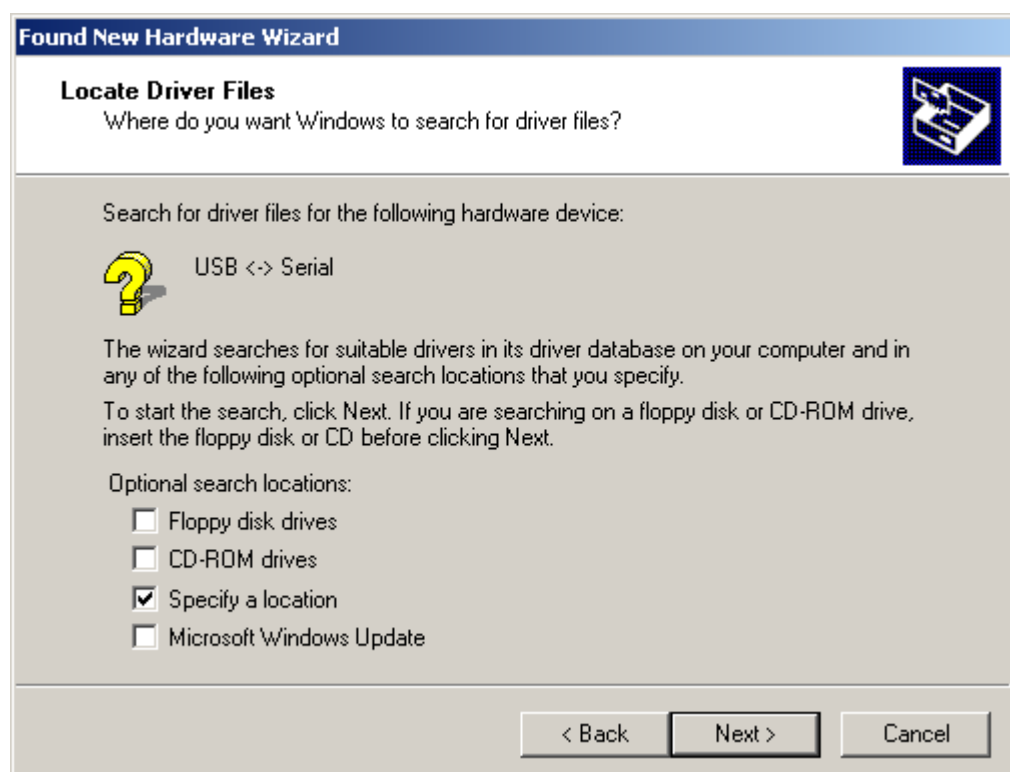
- Download the latest available D2XX drivers from the [Drivers section](#) of the [FTDI website](#) and unzip them to a location on your PC. The MProg installation program copies a link to this web page into the Drivers subdirectory of the MProg folder. For instance if you installed MProg into path "C:\Program Files\MProg 2.8a" then the link can be found in path "C:\Program Files\MProg 2.8a\Drivers".
- Connect the device to a spare USB port on your PC. This will launch the Windows Found New Hardware Wizard. Click "Next" to proceed with the installation.



- Select "Search for a suitable driver for my device (recommended)" as shown below and then click "Next".



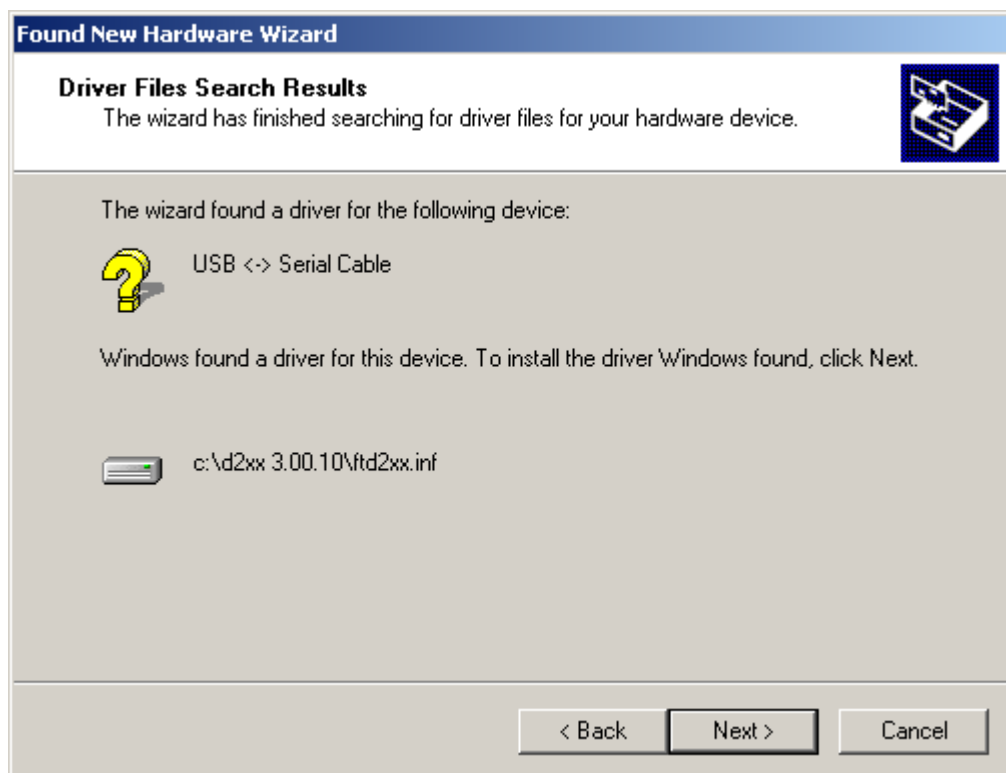
- Check the box next to "Specify a location" and uncheck all others as shown below.



- Clicking "Next" displays a dialog box for you to enter to the location of the drivers.



- Click "Browse" to display an open file dialog box.
- Locate the folder containing the driver files and click "Open", then click "OK". Once Windows has found the required driver .INF file, click "Next" to proceed.



- Windows should then display a message indicating that the installation was successful. Click "Finish" to complete the installation. In the case of FT2232C devices, the second port will also need to be installed. If prompted by the new hardware wizard, repeat the steps above to install the second port. Once completed, the device is ready for use.



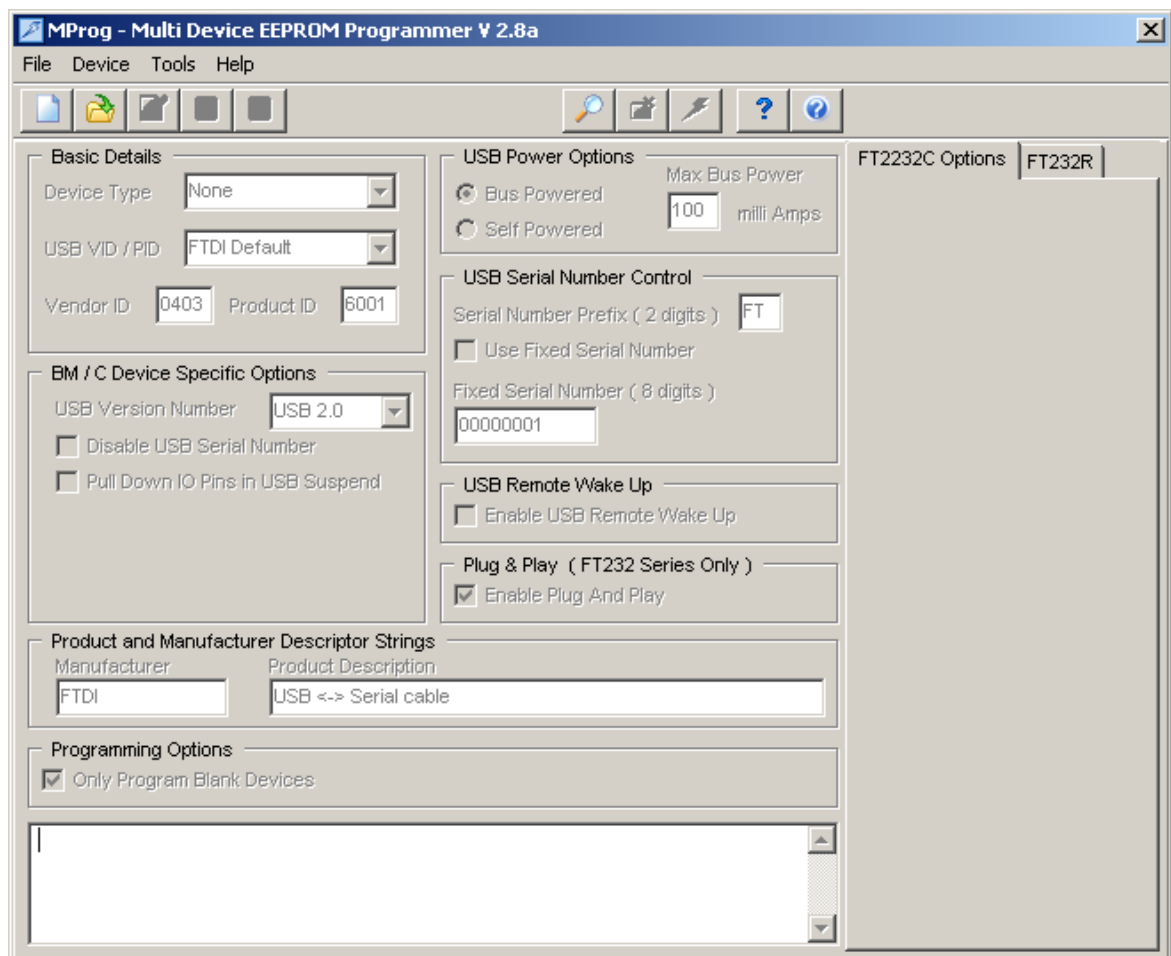
2 MProg Basics

2.1 Modes of Operation

MProg has 3 modes of operation: Idle Mode, Program Mode and Edit Mode. MProg programming parameters are saved in files referred to here as EEPROM templates. Once defined, these EEPROM templates can be loaded by MProg and used to program EEPROMs.

- Idle Mode is the initial mode of operation when the program is launched.
- Edit Mode is used to edit the settings of a EEPROM template.
- Program Mode is used to Program and Erase the device EEPROM(s).

When MProg is launched, it is initially in Idle Mode and the screen will look like this.



User choices in Idle Mode are:

- Create a new EEPROM template. This will change the mode to Edit Mode and allow the user to select the parameters for their project. The new EEPROM Template must then be saved to disk which will then put MProg into Program Mode.
- Open an existing EEPROM template. This will set the MProg parameters to that of the selected

template and put MProg into Program mode, ready to Program devices.

- Scan the USB Bus for available devices. MProg will report how many programmed and blank devices are attached to the target PC.

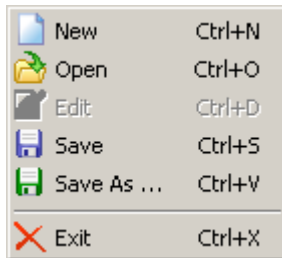
3 Edit Mode Functions

3.1 Create a New EEPROM Template

To create a new EEPROM template click on the Create New Programming Template button on the toolbar



or select New from the File Menu.



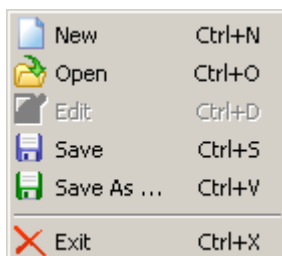
The EEPROM template can define parameters in the following categories:

- [Basic Details](#) ^[20]
- [USB Power Options](#) ^[21]
- [USB Serial Number Control](#) ^[22]
- [USB Remote Wake Up](#) ^[23]
- [Plug & Play \(FT232 Series Only\)](#) ^[24]
- [BM/C Device Specific Options](#) ^[25]
- [FT2232C Device Options](#) ^[26]
- [FT232R Device Options](#) ^[27]
- [FT245R Device Options](#) ^[28]
- [Product and Manufacturer Descriptor Strings](#) ^[29]
- [Programming Options](#) ^[30]

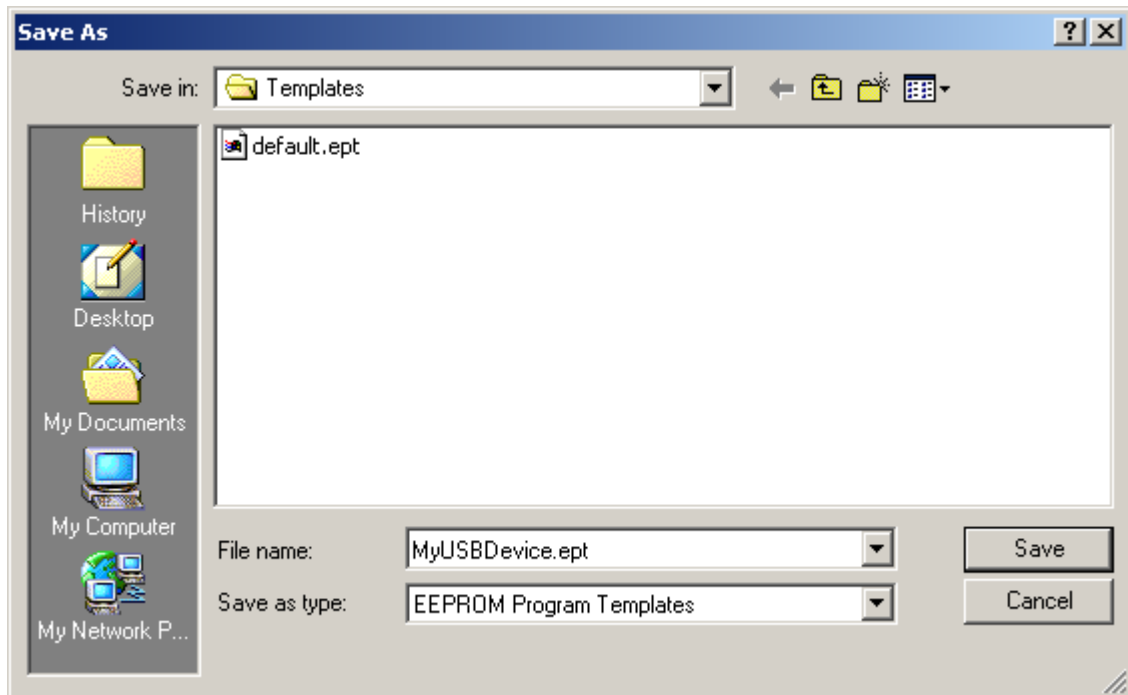
When the EEPROM template is defined, use the "Save As" toolbar button



or select Save As ... from the File menu.



Enter a file name for the EEPROM template, perhaps your product name (MyUSBDevice.ept in this example), and click on "Save"



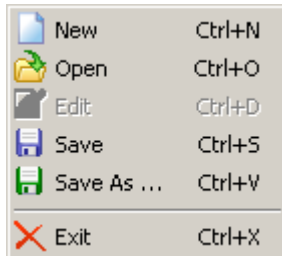
MProg will save the EEPROM template and switch into Program mode.

3.2 Modify an Existing EEPROM Template

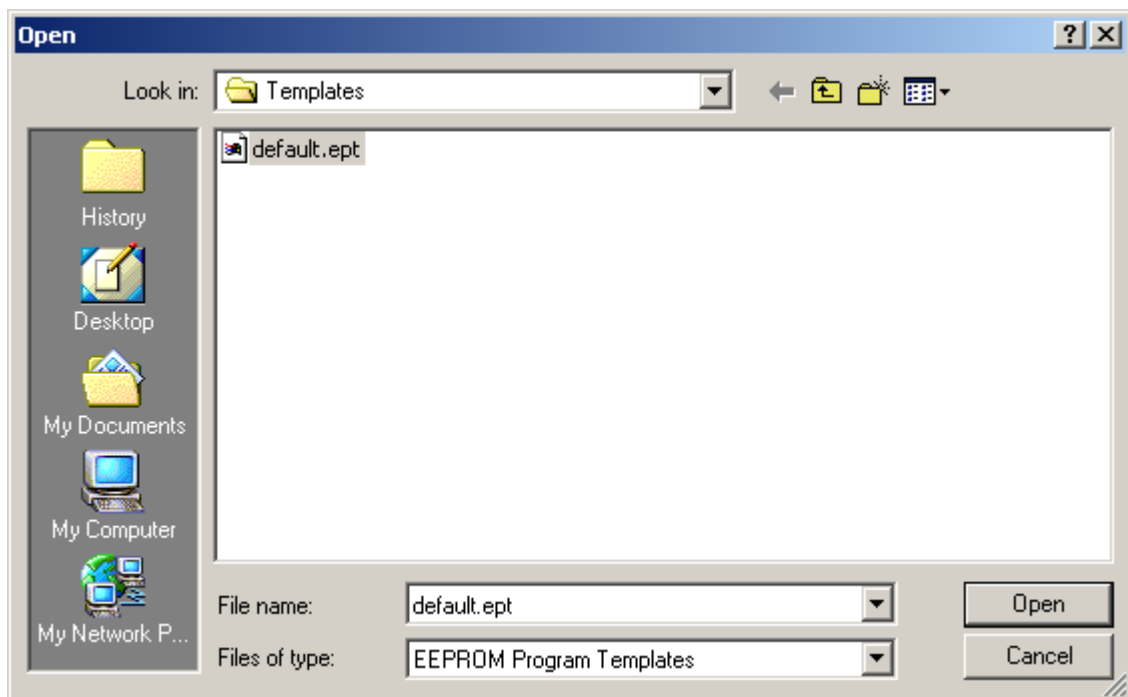
To modify an existing EEPROM template, click on the "File Open" button on the toolbar



or select "Open" from the "File" menu.



This will launch the Open dialog box. Select the template you wish to modify and click on "Open".



MProg will load the template parameters, however it will be in the Program mode. Click on the "Edit Mode" button on the toolbar to switch MProg from Program mode to Edit mode.



You will now be able to modify the template parameters.

Changing any parameter will activate the "Save" button on the toolbar. When the changes are complete, click on the "Save" button to save the changes



or on the "Save As" button to save the resulting template to a different file name.

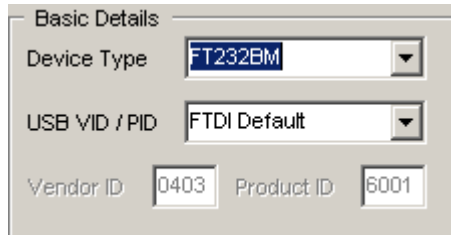


Clicking on "Save" or "Save As" will return MProg to Program mode. To make further changes, click on the "Edit Mode" button again.

4 Device EEPROM Parameters

4.1 Basic Details

Parameters that can be defined for an EEPROM template in the Basic Details section are:



The image shows a 'Basic Details' dialog box with the following fields:

- Device Type: FT232BM (dropdown menu)
- USB VID / PID: FTDI Default (dropdown menu)
- Vendor ID: 0403 (text box)
- Product ID: 6001 (text box)

Device Type

Select the device type you wish to program from the Basic Details box. When selected, this will enable the menu options appropriate to the device you have selected.

USB VID / PID

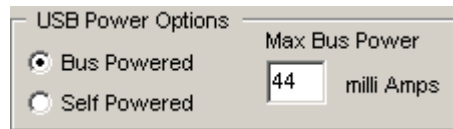
The choices available from the combo-box for VID and PID are

- FTDI Default. If you use the FTDI default values, the VID and PID are fixed and you do not need to enter any values.
- FTDI Supplied PID. If you wish to use FTDI's Vendor ID and FTDI have supplied you with a Product ID for use with your products, please enter the Product ID supplied by FTDI (a 4 digit hexadecimal number) in the Product ID box provided. If you wish to use FTDI's Product ID and do not have a unique Product ID assigned for you by FTDI then please e-mail support2@ftdichip.com with your request. There is no charge for this service to bona-fide FTDI customers.
- Other VID & PID. If you have your own USB Vendor ID and Product ID for your device, then this option will allow the user to enter both the USB VID and USB PID for your product in the boxes provided.

Note that if non-standard VID and/or PID values are used the driver INF and INI files must be modified to accommodate the new VID and PID combination. Please refer to the Appendix of this manual for the modifications that should be made.

4.2 USB Power Options

Details of the USB device power requirements can be set in the USB Power Options section.



Select Bus Powered (the product gets its power from USB) or Self Powered (the product has its own power supply) from the power options. The Max Bus Power value defaults to 100mA for a Bus Powered product and 0mA for a Self Powered product, however these are only default values and can be changed by entering a new value in the Max Bus Power field. A summary of the rules and options are given below:

All USB Devices

- The maximum current that can be supplied over the USB bus without violating the USB specification is 500mA. Do not enter a value of more than 500mA in the Max Bus Power field.
- On plug-in, no USB product is allowed to consume more than 100mA from the USB bus until USB enumeration is complete i.e. the product and its USB drivers are successfully loaded by the host PC. Special care has to be taken in the case of high power Bus Powered products, however the PWREN# pin of the BM series devices simplifies such designs.

Bus Powered USB Devices

These fall into two categories:

- Low power bus powered products. These products consume 100mA or less from the USB bus and can be plugged into any USB host or hub port.
- High power bus powered products. These products consume between 101mA and 500mA from the USB bus and can be plugged into any USB host. However, such products can only be plugged into self powered USB hubs (hubs that have their own power supply) as bus powered USB hubs can only supply 100mA per hub port.

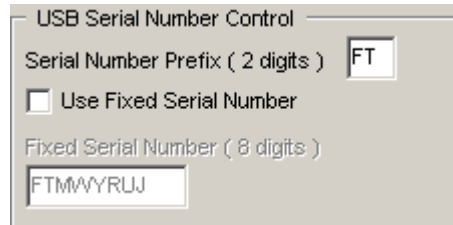
Self Powered USB Devices

These fall into two categories:

- True self powered products. These products have their own power supply and do not require any current from the USB bus. Enter a value of zero in the Max Bus Power field.
- Hybrid self powered products. These products have their own power supply, but also draw some of their operating current from the USB bus. Enter the current draw from the USB bus in the Max Bus Power field.

4.3 USB Serial Number Control

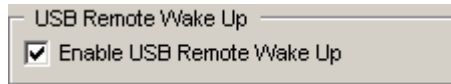
USB products have the ability to embed a unique serial number for each product into the low level descriptors that are accessed by the host controller drivers when the product is plugged into the USB bus. In turn, the serial number of a product can be used to identify the product. The device serial number can be set through the USB Serial Number Control panel.



MProg can generate unique serial numbers automatically for you (recommended). These consist of a 2 letter prefix (e.g. FT) followed by 6 alphanumeric digits generated using an algorithm based on the date and time of programming the device. It is also possible to fix the serial number to a fixed 8-digit alphanumeric string, however we do not recommend this as systems will only recognise the first instance of such a device connected to a PC.

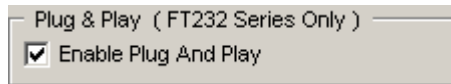
4.4 USB Remote Wake Up

FT8U232AM and FT232BM devices are capable of resuming a PC from the USB suspend (sleep) state via the RI# pin. FT245BM also has this capability via a dedicated pin (SI/WU), as does the FT2232C (SI/WUA, SI/WUB). FT8U245AM does not have support for remote wake up. To use the remote wake-up capability, enable this option by checking the box.



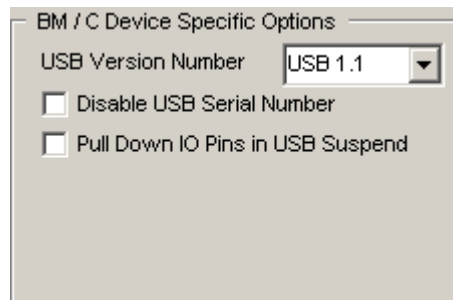
4.5 Plug & Play (FT232 Series Only)

Windows Plug and Play (PnP) - for general purpose USB to RS232 serial converters it is desirable that the device tries to recognise any attached secondary peripheral that conforms to the Windows Plug and Play standard. These devices are typically modems, or serial mice. On recognition of such a device Windows will attempt to identify and load the drivers for the secondary peripheral. For other application areas PNP is not usually required so this option should be turned off.



4.6 BM/C Device Specific Options

The FT232BM and FT245BM 2nd generation devices and FT2232C 3rd generation device have some additional options compared to the AM devices. These options can be controlled via the optional EEPROM.



- **USB Version Number.** The BM and C series devices have the option to set the USB version number to USB 1.1 or USB 2.0. This has no effect on the functionality of the device, however for USB compliance testing we recommend that you choose the USB 2.0 setting. For AM series devices the version number is hard coded to USB 1.1.
- **Disable the USB serial number.** This allows multiple identical USB devices to be plugged into the same host PC. However, it is then impossible for the USB drivers to uniquely identify the devices. We recommend that for most applications, this option is not selected.
- **Pull Down IO Pins in USB suspend.** BM and C devices feature a Power Control Pin (PWREN#). This pin can be used to control power to external peripheral circuitry via a P-Channel MOSFET. When used in this way, select this option to minimize the USB suspend current, otherwise leave this option unselected. See the product data sheets, application schematics and designers guides for more details.

4.7 FT2232C Device Options

The FT2232C incorporates all of the BM specific features described in the [BM/C Device Specific Options](#)^[25] section. It also has the following additional options which can be configured in the external EEPROM. These options will only be displayed when the device type in basic details is set to FT2232C.

The screenshot shows a software configuration window titled 'FT2232C Options'. It has two tabs at the top: 'FT2232C Options' and 'FT232R'. The 'FT2232C Options' tab is active. The window is organized into two main sections, 'Side A' and 'Side B'. Each side contains a 'Hardware' section and a 'Driver' section. In the 'Hardware' section, there are two radio buttons: 'RS232 UART' (which is selected) and '245 FIFO'. Below these is a checkbox labeled 'High Current I/O's'. In the 'Driver' section, there are two radio buttons: 'Virtual COM Port' (which is selected) and 'D2XX Direct'.

- Each channel can be independently configured as FT232BM-style USB UART or a FT245BM-style USB FIFO.
- Each channel can also be individually configured in Fast Opto-Isolated Serial Interface mode. See the FT2232C device datasheet (available from the [Datasheets page](#) of the [FTDI website](#)) for more information on these modes.
- High Current I/O's. Setting this option will enable the high output drive level. The UART/FIFO IO pins will drive out at 12mA instead of their normal 4mA.
- The driver that is installed and used by each channel can be set by selecting Virtual COM Port or D2XX Direct option (not currently supported).

4.8 FT232R Device Options

Additional features available on the FT232R device allow RS232 signals to be inverted and the CBUS pins to be configured through combo-boxes.

FT2232C Options FT232R

Invert RS232 Signals

- ☐ Invert TXD
- ☐ Invert RXD
- ☐ Invert RTS#
- ☐ Invert CTS#
- ☐ Invert DTR#
- ☐ Invert DSR#
- ☐ Invert DCD#
- ☐ Invert RI#

I/O Controls

PWRON# C0

PWRON# C1

PWRON# C2

PWRON# C3

SLEEP# C4

☐ High Current I/O's

☐ Load D2XX driver

Like the FT2232C, the FT232R supports High Current IOs. Setting this option will enable the high output drive level. The UART IO pins will drive out at 12mA instead of their normal 4mA.

The check box to load the D2XX driver rather than the VCP driver is currently not supported.

4.9 FT245R Device Options

Unlike the FT232R, the FT245R does not allow inverting of signals or configuration of the CBUS pins as they are all required.

The screenshot shows a software window titled 'FT245R' with a tab labeled 'FT245R'. Below the title bar, there is a large empty rectangular area. Underneath this area is a section titled 'I/O Controls'. This section contains five rows, each with a label in a text box and a corresponding pin number in a dropdown menu:

Label	Pin
RXF#	C0
TXE#	C1
RD#	C2
WR#	C3
PWRON#	C4

Below the 'I/O Controls' section, there are two unchecked checkboxes:

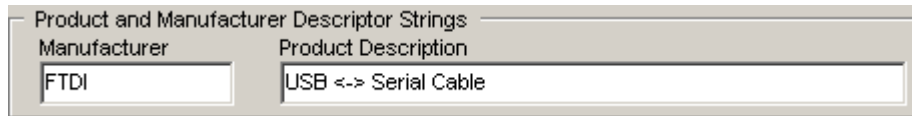
- ☐ High Current I/O's
- ☐ Load D2XX driver

Like the FT232C, the FT232R supports High Current IOs. Setting this option will enable the high output drive level. The FIFO IO pins will drive out at 12mA instead of their normal 4mA.

The check box to load the D2XX driver rather than the VCP driver is currently not supported.

4.10 Product and Manufacturer Descriptor Strings

The Product and Manufacturer Description Strings in the fields allow customisation of the manufacturer and product description strings the device uses.

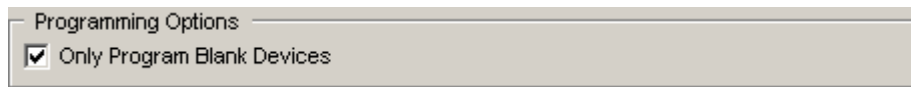


Product and Manufacturer Descriptor Strings	
Manufacturer	Product Description
FTDI	USB <-> Serial Cable

- Manufacturer: the product manufacturer string e.g. "FTDI" - best kept short and abbreviated if possible.
- Product Description: a brief description of the product, maybe including the part number e.g. "US232 USB to RS232 Converter"

4.11 Programming Options

If the "Only Program Blank Devices" box is checked then MProg will only program any blank devices it finds. If the box is unchecked, MProg will program any blank devices it finds and also re-program any programmed devices it finds.

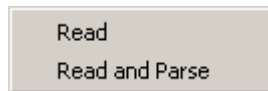


5 Program Mode Functions

The program mode functions of MProg can be accessed through the toolbar



Functions for reading the EEPROM contents can be found in the Tools menu.



The functions available through the toolbar are:

- Scan for Devices
- Erase All Existing Devices
- Program All Existing Devices

The read functions available through the Tools menu are:

- Read
- Read and Parse

5.1 Scan for Devices

To scan the USB bus for available FTDI devices, click on the "Scan for Devices" button on the toolbar.



MProg will display the number of blank devices it finds and the number of programmed devices it finds on the host PC, for example 0 and 1 as shown below.



5.2 Erase All Existing Devices

To erase devices you must be in Program mode. Clicking on the "Erase All Devices" button on the toolbar will erase the EEPROM contents of all FTDI devices connected to the host PC except any FT232R or FT245R devices.



A message will be displayed for each device found and successfully erased. An example is shown below.

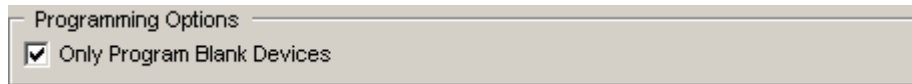


5.3 Program All Existing Devices

To program a device, MProg must be in Program mode. Click on the "Program Devices" button on the toolbar.

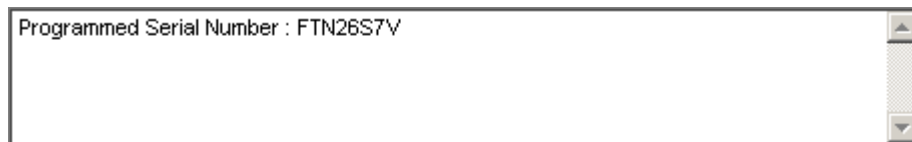


If the "Only Program Blank Devices" box is checked, then MProg will not try and reprogram any pre-programmed FTDI devices it finds. If this selection is unchecked, then MProg will attempt to program every device it finds on the host PC.



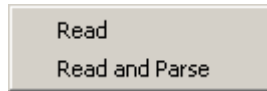
Note: to change the "Only Program Blank Devices" box, you must first enter Edit Mode by clicking on the "Edit Mode" button. After changing the option, click on the "Save" button which will return you to Program mode.

After programming is complete, MProg will display a message showing how many devices it has programmed.

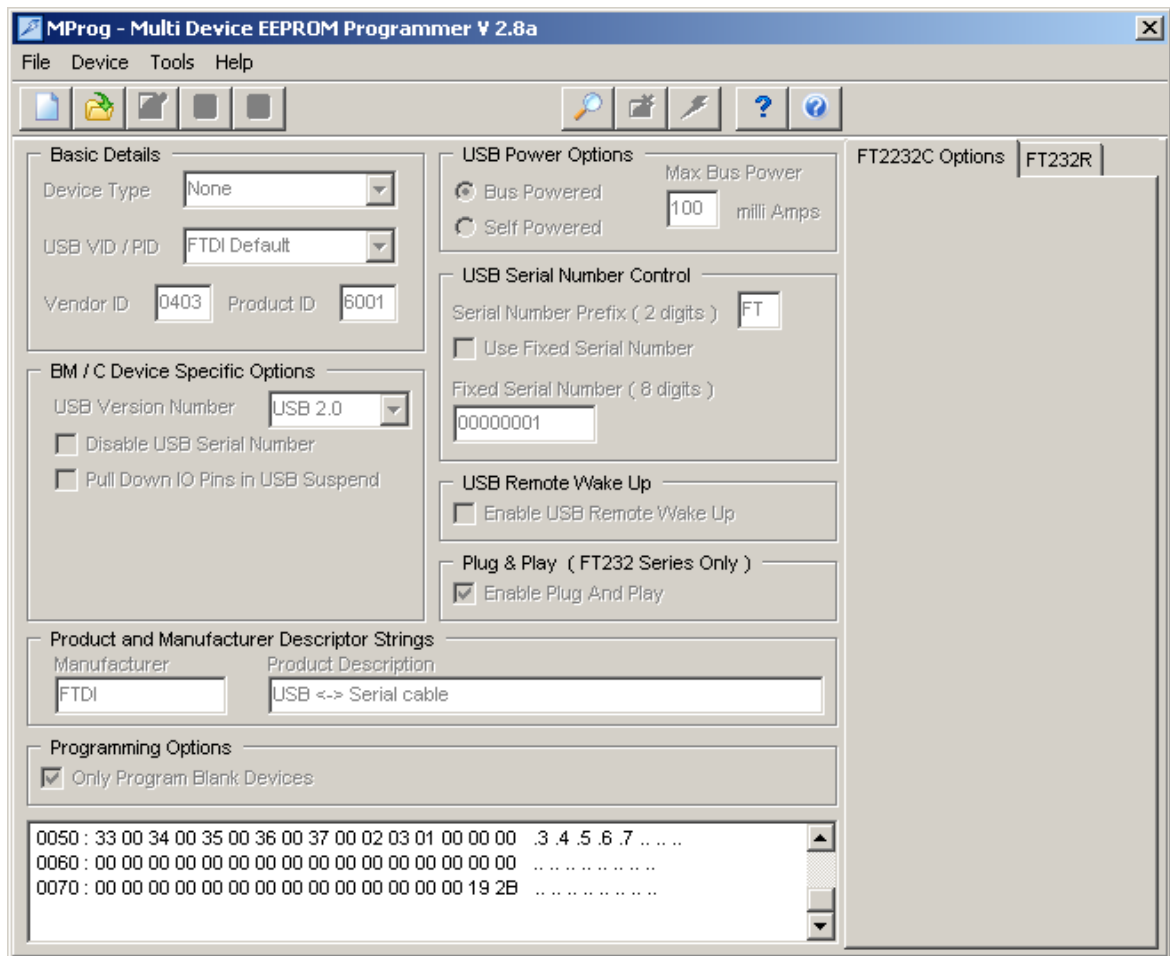


5.4 Read Devices

To read the data in a programmed FTDI device EEPROM, select Read from the tools menu.

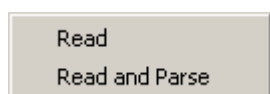


The programmed data will be displayed in the dialog window. For example:

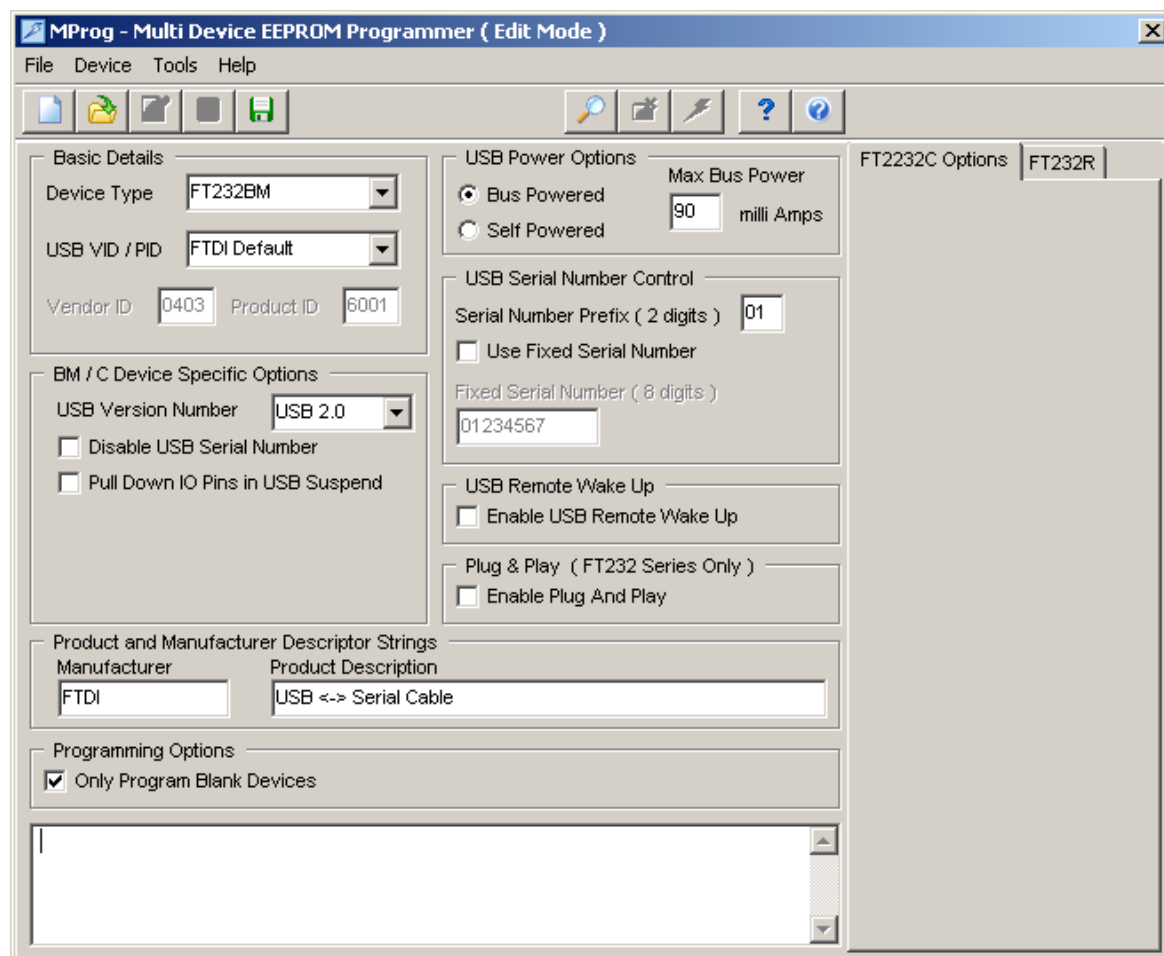


5.5 Read and Parse Devices

To read and parse the data in a programmed FTDI device EEPROM, select Read and Parse from the tools menu.



The programmed data will then be shown in each field in MProg. The device serial number will be shown, but will be greyed out.



6 Appendix

6.1 Customising FTDI's VCP Drivers

6.1.1 Changing the FTDIBUS.INF File

If you are using a VID and PID which is different from the default FTDI ones (VID = 0x0403 and PID = 0x6001 and 0x6010), you must edit the FTDIBUS.INF file using a text editor such as NotePad. You will need to add extra lines into the [FtdiHw] and [ControlFlags] sections containing the new VID and PID. You should also add a description of your product into the [Strings] section as shown.

Do not change any other parts of the supplied INF files unless you are an expert and capable of debugging the results for yourself.

The following example shows a new device with a VID of 0x1234 and a PID of 0x5678 which we shall call "My New USB Device". In this example, the additional lines are shown in red to highlight them. If the device you are adding is an FT2232C, an entry must be created for each of the ports.

```
[FtdiHw]
%USB\VID_0403&PID_8372.DeviceDesc%=FtdiBus,USB\VID_0403&PID_8372
%USB\VID_0403&PID_6001.DeviceDesc%=FtdiBus,USB\VID_0403&PID_6001
%USB\VID_1234&PID_5678.DeviceDesc%=FtdiBus,USB\VID_1234&PID_5678

[ControlFlags]
ExcludeFromSelect=USB\VID_0403&PID_8372
ExcludeFromSelect=USB\VID_0403&PID_6001
ExcludeFromSelect=USB\VID_1234&PID_5678

[Strings]
Ftdi="FTDI"
DriversDisk="FTDI USB Drivers Disk"
USB\VID_0403&PID_8372.DeviceDesc="USB Serial Converter"
USB\VID_0403&PID_6001.DeviceDesc="USB High Speed Serial Converter"
USB\VID_1234&PID_5678.DeviceDesc="My New USB Device"
WINUN="Software\Microsoft\Windows\CurrentVersion\Uninstall"
FtdiBus.SvcDesc="USB Serial Converter Driver"
```

6.1.2 Changing the FTDI.PORT.INF File

If you are using a VID and PID which is different from the default FTDI ones (VID = 0x0403 and PID = 0x6001 and 0x6010), you must edit the Ftdiport.inf File using a text editor such as NotePad. You will need to add an extra line into the [FtdiHw] section containing the new VID and PID. You should also add a description of your product into the [Strings] section as shown.

Do not change any other parts of the supplied INF files unless you are an expert and capable of debugging the results for yourself.

The following example shows a new device with a VID of 0x1234 and a PID of 0x5678 which we shall call "My New USB Device". In this example, the additional lines are shown in red to highlight them. If the device you are adding is an FT2232C, an entry must be created for each of the ports.

```
[FtdiHw]
%VID_0403&PID_8372.DeviceDesc%=FtdiPort,FTDIBUS\COMPORT&VID_0403&PID_8372
%VID_0403&PID_6001.DeviceDesc%=FtdiPort232,FTDIBUS\COMPORT&VID_0403&PID_6001
%VID_1234&PID_5678.DeviceDesc%=FtdiPort232,FTDIBUS\COMPORT&VID_1234&PID_5678

[Strings]
FTDI="FTDI"
DriversDisk="FTDI USB Drivers Disk"
PortsClassName = "Ports (COM & LPT)"
VID_0403&PID_8372.DeviceDesc="USB Serial Port"
VID_0403&PID_6001.DeviceDesc="USB Serial Port"
VID_1234&PID_5678.DeviceDesc="My New USB Device"
FtdiPort.SvcDesc="USB Serial Port Driver"
SerEnum.SvcDesc="Serenum Filter Driver"
```

6.1.3 Changing the VCP Uninstaller INI Files

If you are using a VID and PID which is different from the default FTDI ones (VID = 0x0403 and PID = 0x6001), you must edit the uninstaller Ftdiunin.ini and Ftdiun2k.ini files using a text editor such as NotePad. In the [Uninstall] section of each file, edit the line

```
Device=VID_0403&PID_6001
```

to reflect the actual VID and PID used.

The following example shows a new device with a VID of 0x1234 and a PID of 0x5678. In this example, the altered line is shown in red to highlight it.

In file FTDIUNIN.INI:

```
[Uninstall]
Device=VID_1234&PID_5678
InfFiles=FTDIBUS,FTDIPOINT,FTSERMOU
Key=FTDICOMM
```

In file FTDIUN2K.INI:

```
[Uninstall]
Device=VID_1234&PID_5678
Converter=FTDIBUS
Serial=FTSER2K
Key=FTDICOMM
```


6.2 Customising FTDI's D2XX Drivers

6.2.1 Changing the FTD2XX.INF File

If you are using a VID and PID which is different from the default FTDI ones (VID = 0x0403 and PID = 0x6001), you must edit the FTD2XX.INF file using a text editor such as NotePad. You will need to add extra lines into the [FtdiHw] and [ControlFlags] sections containing the new VID and PID. You should also add a description of your product into the [Strings] section as shown.

Do not change any other parts of the supplied INF files unless you are an expert and capable of debugging the results for yourself.

The following example shows a new device with a VID of 0x1234 and a PID of 0x5678 which we shall call "My New Device". In this example, the additional lines are shown in red to highlight them. If the device you are adding is an FT2232C, an entry must be created for each of the ports.

```
[FtdiHw]
%USB\VID_0403&PID_6001.DeviceDesc%=FTD2XX, USB\VID_0403&PID_6001
%USB\VID_1234&PID_5678.DeviceDesc%=FTD2XX, USB\VID_1234&PID_5678

[ControlFlags]
ExcludeFromSelect=USB\VID_0403&PID_6001
ExcludeFromSelect=USB\VID_1234&PID_5678

[Strings]
Ftdi="FTDI"
USB\VID_0403&PID_6001.DeviceDesc="FTDI FT8U2XX Device"
USB\VID_1234&PID_4567.DeviceDesc="My New Device"
FTD2XX.SvcDesc="FTD2XX.SYS FT8U2XX device driver"
WINUN="Software\Microsoft\Windows\CurrentVersion\Uninstall"
DriversDisk="FTDI FTD2XX Drivers Disk"
```

6.2.2 Changing the D2XX Uninstaller INI File

If you are using a VID and PID which is different from the default FTDI ones (0xVID = 0403 and PID = 0x6001), you must edit the uninstaller FTD2XXUN.INI file using a text editor such as NotePad. In the [Uninstall] section edit the line

```
Device=VID_0403&PID_6001
```

to reflect the actual VID and PID used.

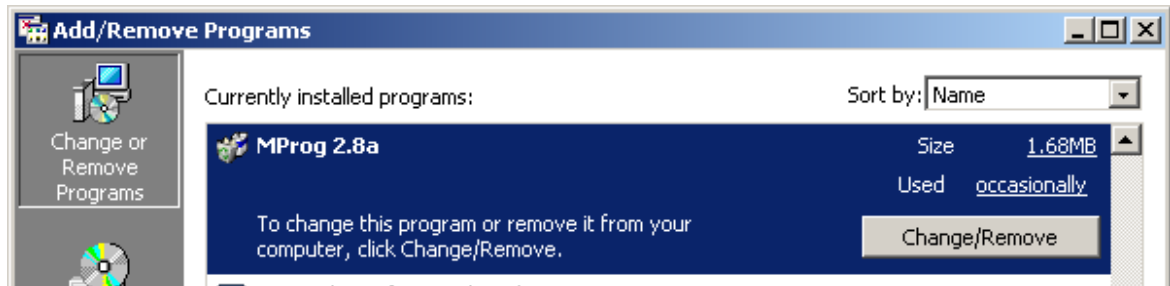
The following example shows a new device with a VID of 0x1234 and a PID of 0x5678. In this example, the altered line is shown in red to highlight it.

```
[Uninstall]  
Device=VID_1234&PID_5678  
Converter=FTD2XX
```

6.3 Uninstalling MProg

To uninstall MProg, follow the instructions below:

- Disconnect any FTDI devices attached to the PC.
- Open the Add/Remove Programs utility which is located in the Windows Control Panel.
- Select MProg from the list of installed programs and click Change/Remove. This will launch the MProg uninstaller.

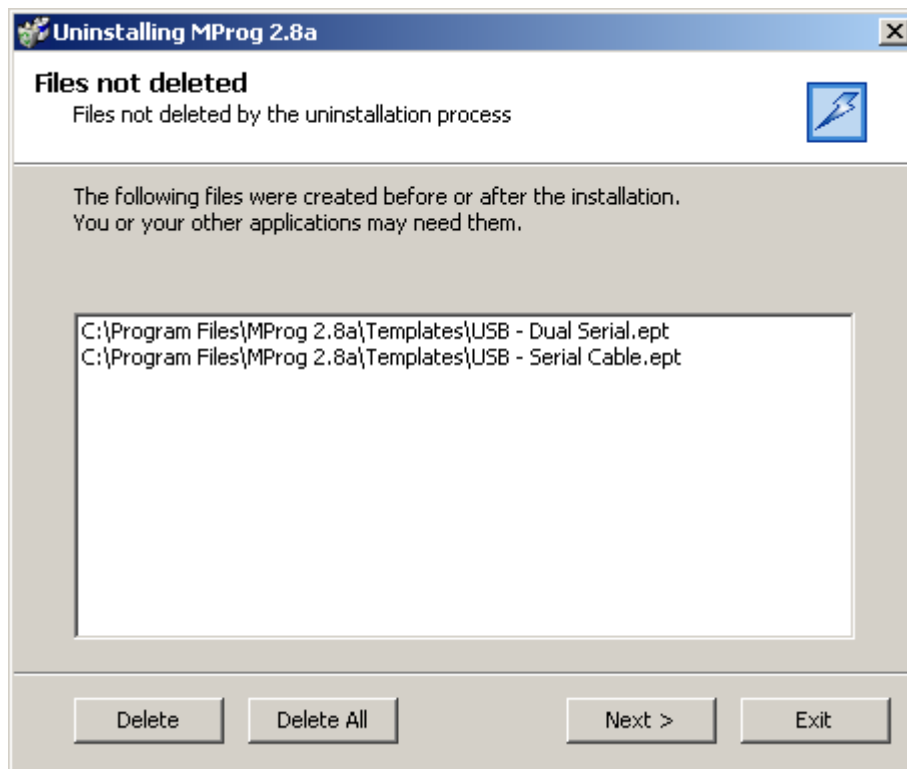


- The user will be asked for confirmation to uninstall MProg. Click Next to proceed with the uninstallation or click Cancel to abort.



- The files and directories required for MProg will be removed. If there are files other than the original installation files present (e.g. user defined templates) the following screen will be

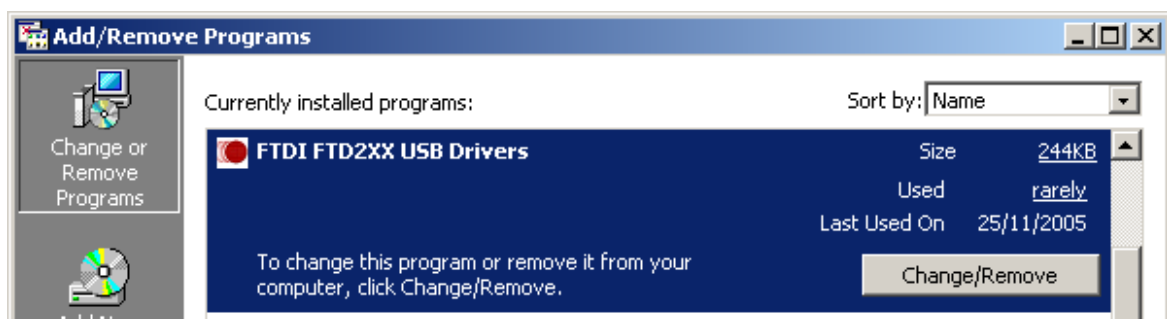
displayed to give the option of deleting them.



- The following screen is displayed when the process is complete. Clicking Finish will close the uninstall program.



- The MProg uninstaller does not remove the D2XX drivers from the system. To remove the drivers, use the driver uninstaller in Add/Remove Programs as shown below.



Index

- B -

Basic Details 20
Bus Powered 21

- C -

Customising Drivers 37, 38, 39, 40, 41

- D -

Device Type 20
Driver Installation 10

- E -

EEPROM Templates 16, 18
EPT 16, 18
Erase All Existing Devices 33
Erasing Devices 33

- F -

FT2232C EEPROM Options 25, 26
FT232BM EEPROM Options 25
FT232R EEPROM Options 25, 27
FT245BM EEPROM Options 25
FT245R EEPROM Options 25, 28

- I -

INF Files 37, 38, 40
INI Files 39, 41
Installing MProg 5
Introduction 3

- M -

MProg Modes 14
MProg Toolbar 31

- P -

PID 20
Plug & Play 24
Program All Existing Devices 34
Programming Devices 34

- R -

Read and Parse EEPROM 36
Read EEPROM 35

- S -

Scan for Devices 32
Self Powered 21
Serial Number 22

- U -

Uninstalling MProg 42
USB Power Options 21
USB Remote Wake Up 23
USB Serial Number Control 22

- V -

VID 20