

Installing Middle Mouse Button

Middle Mouse Button is a Windows 95 & Windows NT application which intercepts the middle mouse button of a 3-button mouse and converts a single middle button click into a default button (normally the left button) double click. The program dynamically detects if you have selected the "left handed mouse" option in the control panel. The ergonomics of this action are obvious; one click instead of two rapid clicks causes less wear and tear on both the mouse and the person using the mouse.

This application was written because although Windows 95 and Windows NT allow using a 3-button mouse, there is no built-in usage for the middle button. Also there does not appear (at this time) to be any support by OEM mouse vendors in providing such a utility for 3-button mice for use in the Windows 95 or Windows NT environments.

This application is targeted for Windows NT ver. 3.51 and Windows 95. It will work on Windows NT ver. 3.50 with the limitation of only being active within an application window. It does not simulate the double click when the mouse cursor is on the desktop when used under Windows NT ver. 3.50. It will not work under any version of Win32s running on 16-bit Windows 3.x.

To install Middle Mouse Button, copy all of the files in the self extracting archive file MBUTT32.EXE or from the file MBUTT32.ZIP (or from the floppy disk root directory) into the subdirectory (folder) of your choice. If installing on a Windows NT system we suggest either the "Win32app" subdirectory (folder) or the root "Windows\System32" subdirectory (folder). If installing on a Windows 95 system we suggest using either the "Program Files\Accessories" folder or the root "Windows" folder.

The files included in the archive are:

MBUTTON.EXE The main executable file.
MBHOOK.DLL A dynamic link library required by the system.
MBUTTREG.WRI Registration form and instructions.
MBUTTON.WRI This file.. Installation and troubleshooting information
MBUTTON.TXT Brief description of the program.

Once the Mbutton files are copied to your disk all you have to do is execute "MBUTTON.EXE". If you create an entry for "MBUTTON.EXE" in your "Startup" group (or create a Windows 95 "Startup" menu entry), it will run every time you logon to the system.

The first time it executes (and every time until the program is registered) a popup dialog box will appear and ask for registration information. Once the program has been registered, this dialog box will not reappear. This program does not contain a "time bomb" to limit its unregistered usage, but you will have to register the program to avoid having to answer the dialog box enquiry when the program starts. You may use the MBUTTREG.WRI file as an order form to obtain the Product ID code needed to complete the registration process. You may also register this program via Compuserve (GO SWREG, Registration ID: 7865)

You may pass this (unregistered) program along to your friends and associates who have 3-button mice. Please distribute the original archive file rather than the individual files contained therein. (The archive file is located in the REDIST subdirectory if you are installing from the floppy disk version). You may also post it to any BBS or on-line service. We only ask that if you continue to use this program on a regular, basis that you register it. You will be rewarded with having the program load swiftly and silently instead of nagging you when it starts.

Digital Control Systems

P.O. Box 505

Hermitage, TN. 37076

(615) 889-6357 Fax: (615) 889-9595 Email: 73347.145@compuserve.com

What to do if Mbutton reports that a 3 button mouse is not installed

Windows 95 mouse drivers will only support a 3-button mouse if the specific driver was designed to do so. The Logitech driver will support Logitech (or compatible) 3-button mice, the Mouse Systems driver supports Mouse Systems (or compatible) 3-button mice, etc. The only solution for using the middle mouse button under Windows 95 is to use the proper driver for your mouse.

All of the built-in Windows NT mouse drivers will support 3-button mice. However, sometimes the Windows NT setup program does not detect a 3-button mouse as having 3 buttons. When this happens the first thing to do is to use setup to remove the current mouse driver, and then re-install the driver. If this attempt is not successful, you will have to manually change an entry in the system registry using the registry editor (**REGEDT32.EXE**).

Using this tool requires system administrator privileges. If you are using a company owned computer you may need to contact your network administrator or tech-support coordinator in order to obtain the required security access privilege. (or to have a technician do it for you.)

On the next several pages you will first find several sections of information detailing how to edit values in the Windows NT registry. Following these sections is specific information related to the **NumberOfButtons** registry entry for the standard Microsoft supplied mouse drivers.

Editing values in the Windows NT Registry

Each value entry in Registry Editor appears as a string that consists of three components, as shown in the following figure.

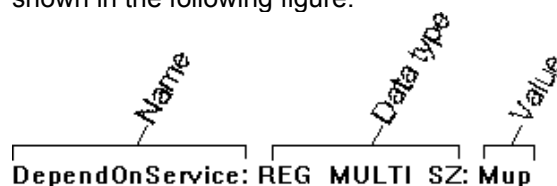


Figure 1.1 The Three Components of a Value Entry

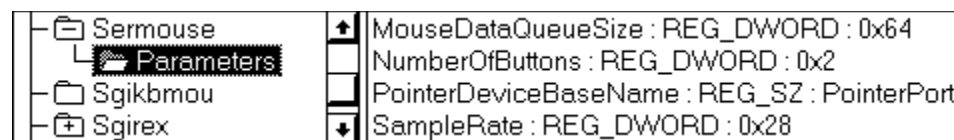


Figure 1.2 Sample view of Serial Mouse Parameters entries

The following rules govern the content of these three value entry components:

- The name of the value is a string of up to 16,000 Unicode characters (32K). This name can contain backslash (\) characters. The name itself can be null (that is, "").
- The data type of the value is REG_BINARY, REG_DWORD, REG_EXPAND_SZ, REG_MULTI_SZ, or REG_SZ. Other data types can be defined by programs, but editing of only these five specific data types is supported in Registry Editor.

The Registry preserves case as you type it for any entry but ignores case in evaluating the data. The names are case-insensitive. However, the data is defined by specific applications (or users), so it might be case-sensitive, depending on how the program that uses it treats the data.

To edit any value

1. In the right pane of the Registry Editor window, double-click the value entry.
– Or –
From the Edit menu, choose the String, Binary, DWord, or Multi String command as appropriate for the selected value.
2. Edit the value that appears in the related Editor dialog box, and then choose the OK button.
The Binary and DWord editors give you the flexibility to select the base of a number system in which you want to edit your data. In the Binary editor, you can edit your data as binary (base 2) or hex (base 16). In the DWord editor, you can edit your data in binary, hex, or decimal (base 10). Hex is the default base for both editors. These types of data will always be displayed in hex in the right pane of the Registry Editor.

Tip To view numbers in decimal format, double-click the value entry and select the Decimal format option. Cancel the dialog box when you finish checking the value. Information stored in a nonvolatile key remains in the Registry until you delete it. Information stored in a volatile key is discarded when you shut down the system. Everything under a volatile key is also volatile. Everything stored under the HKEY_LOCAL_MACHINE\HARDWARE key is volatile.

Adding a Key

You can add a key to store data in the Registry. For example, you might add a subkey under CurrentControlSet\Services to start a service process you have written or to install a device driver that doesn't have an installation program. To do this, you must have Create Subkey access permission for the key under which you are adding a subkey.

To add a key to the Registry

1. Select the key or subkey under which you want the new key to appear, and then choose the Add Key command from the Edit menu, or press the INS key.
2. In the Key Name box of the Add Key dialog box, type the name that you want to assign your key.
The key name cannot contain a backslash (\), and it must be unique in relation to other subkeys at the same level in the hierarchy. That is, Key1 and Key2 can each have a subkey named Key3, but Key1 cannot have two subkeys named Key3.
3. Leave the Class box blank, as this entry is reserved for future uses.
4. Choose the OK button to display the new key in the Registry Editor window.

Adding a Value Entry to a Registry Key

In Registry Editor, you can assign a new value entry to a key or edit the value entry of an existing key. When you do this, the value that you add appears in the data pane of the selected Registry window.

To add a value entry to a Registry key

1. Select the subkey to which you want to add a value entry.
2. From the Edit menu, choose the Add Value command.
The Value Name box is displayed.
Tip To quickly open the Value Name box, move the focus to the right pane using the TAB key or the mouse, and then press the INS key.
3. In the Value Name box, type the name you want to assign to the new value entry.
4. In the Data Type box, select the type that you want to assign to the value entry.
5. Choose the OK button, and then type the value in the Editor dialog box that appears.
Choose the OK button again to display the new entry in the Registry Editor window.

Deleting a Key or a Value Entry

You can use either the Delete command from the Edit menu or press the DELETE key to remove selected keys or value entries from the Registry. However, you cannot delete any of the predefined subtrees or change the name of a key.

Caution There is no Undo command for deletions. Registry Editor asks you to confirm the deletions if the Confirm On Delete option is checked under the Options menu. However, the message does not include the name of the key you are deleting. Check your selection carefully before proceeding when you delete a key.

The only way to retrieve a key that you mistakenly delete is to restart the computer. Press the spacebar immediately after selecting Windows NT at the Boot Loader prompt, and then choose the Last Known Good option to roll back to an earlier configuration.

You can protect the Registry from accidental deletions in these ways:

- Protect data through read-only mode
To do this, choose the Read Only command from the Options menu. When this command is checked, Registry Editor will not save any changes made, protecting the data from accidental changes.
- Protect data through confirmation
To do this, choose the Confirm On Delete command from the Options menu. When this command is checked, the Registry Editor asks you to confirm deletion of any key or value.

Entries which control the number of mouse buttons

The following section describes the registry entries which determine the number of mouse buttons which are visible to applications. This entry can be wrong if the driver installation program did not properly detect the number of actual buttons on your mouse. To correct this problem, use the registry editor to change the **NumberOfButtons** entry to the proper value for your mouse. You will need system administrator privileges to change this value in the Windows NT registry. The system will also need to be restarted in order for this change to take effect.

Microsoft Bus Mouse Port Driver Entries

The Bus Mouse driver is used for the older Microsoft Bus Mouse (green buttons) and for the Logitech Bus Mouse . Other mice which have the 9-pin DIN connector may be compatible with this driver (consult the documentation supplied with your mouse). The value entries for the Microsoft bus mouse are found in this subkey:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Busmouse\Parameters

NumberOfButtons **REG_DWORD** **>= 0x1**

Specifies the number of buttons on the bus mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.
Default: 0x2 (two buttons) Change this to 0x3 for a 3-button mouse.

Intel 8042 Port Driver Entries

The i8042prt driver handles the keyboard and mouse port mouse (uses a 6-pin DIN connector and is also known as a PS/2-compatible mouse) for the Intel 8042 controller. These value entries are found in the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\i8042prt\Parameters

NumberOfButtons **REG_DWORD** **>= 0x1**

Specifies the number of buttons on the mouse port mouse. If an incorrect number of buttons is detected at startup time and placed in the Registry, this value can be used to override it.
Default: 0x2 (two buttons) Change this to 0x3 for a 3-button mouse.

Microsoft InPort Bus Mouse Port Driver Entries

The Inport Bus Mouse driver is used for newer versions of the Microsoft Bus Mouse. Other mice which have the 9-pin DIN connector may be compatible with this driver (consult the documentation supplied with your mouse). The value entries for the Microsoft InPort bus mouse are found in the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Inport\Parameters

NumberOfButtons **REG_DWORD** **>= 0x1**

Specifies the number of buttons on the Microsoft InPort bus mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.
Default: 0x2 (two buttons) Change this to 0x3 for a 3-button mouse.

Microsoft Serial Mouse Port Driver Entries

The Serial Port Mouse driver supports serial mice from multiple vendors. The value entries for the Microsoft serial mouse are found in the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Sermouse\Parameters

NumberOfButtons **REG_DWORD** $\geq 0x1$

Specifies the number of buttons on the serial mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.
Default: 0x2 (two buttons) Change this to 0x3 for a 3-button mouse.

Other Vendor Mouse Port Driver Entries

If you have installed a driver supplied with an OEM mouse you will have to use the above sample entries as a guide to changing the appropriate **NumberOfButtons** value. Most OEM's will probably have used Microsoft's sample mouse driver as a starting point for their own mouse. Contact the mouse manufacturer if you have problems enabling the middle button.