H1-TF-LINK

H1-Library for PC SIMATIC S5 CP 143

Version 1.1.0.4

Operating systems

Windows

- 10
- 8
- 7
- Vista
- XP

Programming languages

- C/C++
- VB
- C#
- VB.net
- Delphi
- Excel

Hardware

PC with network card and MS-Windows operating system

Delivery

Main directory	
H1TFLink.pdf	the file documentation
h1tflink.h	C/C++ Headerfile
h1tflink.lib	the Lib-file for linking

Directory ´WinPCAP´	**Files for C / C++**
h1tflink.dll	the DLL to use the WinPCAP driver
directory ´WinPCAP/Drivers´	Here you can find the WinPCAP driver, if you want to use it. Install this with admin rights and make sure that the WinPCAP service is started automatically.
directory WINPCAP/Demo	C++ sample program for WinPCAP
directory NDIS	the DLL to use the NDIS driver
directory NDIS/Drivers	Here you will find the WinPCAP driver, if you want to use it. Install this with admin rights and make sure that the WinPCAP service is started automatically.
directory NDIS/Demo	C++ sample program for NDIS

How it works:

H1TFLINK-Lib is a DLL for MS-Windows (NT / XP / Vista / 7/8/10), which connects a PC to Industrial Ethernet via the Siemens H1 protocol on ISO / MAC basis. With simple functions, the user can quickly build C / C ++, Delphi, Visual Basic or Excel H1 connections and send and receive data. Only the MAC address, DSAP,

SSAP of the partner is required for the coupling. **Please note, if the partner requests a specific MAC: address as sender, you must set the correct MAC address for your adapter under Windows. In the appendix you will find a guide**

Functional description in detail

Please note: The functions are executed synchronously, which means that the function returns to the caller only after the task has been completed. For asynchronous operation, simply call these functions from a separate thread, which is responsible for communicating the system.

Currently, an NDIS driver and the WinPCAP driver are supported. Depending on the driver, the corresponding DLLs are to be used.
Yes after which driver is used, the appropriate DLL is to be used.
NOTE: The name of the DLL is always “h1tflink.dll”. You can find the appropriate DLL in the corresponding directory.

Ethernet-Adaptername

The network must be installed. The driver is set to the NDIS level. The drivers are loaded dynamically. Basically, the driver tries to find the first NDIS-capable adapter in the system this is then also used. So if you only use a network adapter, it is also used. If you want to address a particular adapter, it must be set before the H1TFOpen routine with the function H1TFSetAdaptername “Attention also dial-up adapters can be NDIS-capable, but the H1 protocol does not run As a service name.

H1TFSetAdaptername

Set the Networkadapter

Call parameter

Nr.	Memory width	Designation	Function
1		Adaptername	Windows stores the information in the following registry path: HKEY_LOCAL_MACHINE * Software * Microsoft * Windows NT * CurrentVersion * NetworkCards * 1 * ServiceName eg: {00030 ...} * 2 * ServiceName .... If you want to use the first card, call eg On H1TFSetAdaptername (?? {00030 ...} ”)

Return value

The functions provide a “32 Bit signed” as a return value with the following meaning:

Value	Error description	Meaning / Reaction
0	Adaptername invalid	

Value	Error description	Meaning / Reaction
1	Adaptername okay	success

C/C++

```
extern BOOL WINAPI
H1TFSetAdaptername (LPCSTR AdapterName);
```

H1-Functions

H1TFOpen

Initializes the connection. If the function was successfully executed, the connection is used internally.

- Connection is active (bPassive = 0):
  - The driver immediately starts the connection setup with the partner
- Connection is passive (bPassive = 1):
  - The driver waits for the connection to be established by the partner

The connection status can be queried with H1TFGetStatus.

Call parameters

Nr.	Memory width	Designation	Function
1	Pointer to an array of 6 bytes	MACAdr	Destination-MAC-Address z.B. for 08:00:06:01:00:01 BYTE Mac[6] = {0x0,0x08. 0x00, 0x06, 0x01, 0x00, 0x01}
2	Pointer to 0 “-termined string (C-string, 32-bit pointer)	SSAP	own SAP. A maximum of 8 characters are used.
3	32 Bit unsigned	SSAPLen	Length of your own SAP. A maximum of 8 characters are used
4	Pointer to 0 ”-termined string (C-string, 32-bit pointer)	DSAP	SAP of the Partner. A maximum of 8 characters are used.
5	32 Bit unsigned	DSAPLen	Length of the partner's SAP. A maximum of 8 characters are used
6	32-Bit value (BOOL)	bPassive	Indicates whether the PC or the partner that is establishing the connection. 0: PC establishes the connection to 1: the partner establishes the connection

Return value

The functions provide a “32 Bit signed” as a return value with the following meaning:

Value	Error description	Designation
>= 0	All OK	The return is the reference number for this connection and must be used as input parameter Ref for all other functions.
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFOpen (BYTE *DstMacAdr, BYTE *SSAP, DWORD LenSSAP, BYTE *DSAP, DWORD LenDSAP, BOOL
bPassive);
```

H1TFClose

Deinitializes the connection, frees memory, and disconnects.

Call parameters

Nr.	Memory width	Designation	Function
1	32 Bit unsigned	Ref	The reference of the connection generated with H1 FOpen. Used to identify the connection internally.

Return value

The functions provide a “32 Bit signed” as a return value with the following designation:

Value	Error description	Designation / Reaction
0	all OK	Memory enabled again and connection closed, if available
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFClose (long Ref);
```

Receiving and sending

The driver has an internal FiFo for received data packets.
In the background this FiFo is operated. The driver ensures that no overflow occurs.
If the limit of the buffer is reached, the driver informs the partner in good time that the queue is filled.
For this reason it is necessary that the application retrieve the data as regularly as possible with H1TxData H1TFRx.

H1TFPacketInQ

Checks if at least one packet is ready in the receive queue.

Call parameters

Nr.	Memory width	Designation	Function
1	32 Bit unsigned	Ref	The reference of the connection generated with H1TFOpen. Used to identify the connection internally

Return value

The functions returns a “32 Bit signed” value with following designation:

Value	Error description	Designation / Reaction
0	No package available	
>0	at least one package available	

Value	Error description	Designation / Reaction
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFPacketInQ (long Ref);
```

H1TFRxFifoClear

Deletes all packages in the RxFifo for the specified connection.
This may be necessary. Even after connection interruption, the received packets and the queue remain.
These can be either completely read out or discarded.

Call parameters

Nr.	Memory width	Designation	Function
1	32 Bit unsigned	Ref	The reference of the connection generated with H1TFOpen. Used to identify the connection internally

Return value

The functions returns a “32 Bit signed” value with following designation::

Value	Error description	Designation / Reaction
0	FiFo is deleted	
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFRxFifoClear (long Ref);
```

Read / Write

Function	Description / Purpose
H1TFRx	Try to receive data.
H1TFTx	Send data to partner.
H1TFTxUnlocked	H1TFTxUnlocked send a Packet to the partner without a Lock on the connection. This function is used, while e.g. In another thread H1TFRx is running. While H1TFRxData is running, the connection is blocked. A normal H1TFTx would block until the H1TFRx returns.

H1TFRx is used to receive data.
Please note: The received data packet remains in the queue even if the connection is not connected.
The user is responsible for emptying the queue.

Call parameters

The receive- and send functions have the same input parameter:

Nr.	Memory width	Designation	Function
1	32 Bit unsigned	Ref	The reference of the connection generated with H1TFOpen. Used to identify the connection internally

Nr.	Memory width	Designation	Function
2	32-Bit Address	Buffer	The address to the source or target memory in the PC.
3	32 Bit unsigned	Cnt	For Rx, the maximum number of data bytes to be received. For Tx, the number of bytes to be sent.
4	32 Bit signed	Timeout	Timeout in ms for sending or receiving data 0 = don't wait

Return value

The functions returns a “32 Bit signed” value with following designation::

Value	Error description	Reaktion
>=0	Count of the sent / received data bytes	
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFRx (long Ref, void *Buf, DWORD MaxCnt, long RxTimeout);

extern long WINAPI
H1TFTx (long Ref, void *Buf, DWORD Cnt, long Timeout);

extern long WINAPI
H1TFTxUnlocked (long Ref, void *Buf, DWORD Cnt, long Timeout);
```

H1TFGetStatus

Returns the state of the connection

Call parameters

Nr.	Memory width	Designation	Function
1	32 Bit unsigned	Ref	The reference of the connection generated with H1TFOpen. Used to identify the connection internally

Return value

The functions returns a “32 Bit signed” value with following designation::

Value	Error description	Designation / Reaction
0	no available connection	
1	connection available	
<0	see Returncodes	

C/C++

```
extern long WINAPI
H1TFGetStatus (long Ref);
```

H1TFCloseAll

Deinitializes the connection(s), releases memory, and disconnects the TCP/IP connection(s).

C/C++

```
extern void WINAPI
H1TFCloseAll (void);
```

Returncodes

KConstant in the include file	Value	Designation
E_H1TF_OKAY	0	Successful, no error occurred
E_H1TF_TIMEOUT	-1	Timeout occurred
E_H1TF_NONETDRIVER	-2	Didn´t found the network driver
E_H1TF_NOCONFREE	-3	No more connections
E_H1TF_NOTCONNECTED	-5	Connection has not yet been established
E_H1TF_RXFIFO_OVERFLOW	-15	Reception Fifo has overflowed
E_H1TF_MEMALLOC	-97	Not enough memory available
E_H1TF_GENERAL	-98	general error occurred
E_H1TF_BADREF	-99	Invalid reference invalid

C/C++ - Header

```
/*
  H1TFLink.h
  Version 1.1.0.4
*/

#define E_H1TF_OKAY             0 //Okay, no error occurred
#define E_H1TF_TIMEOUT             -1 // Timeout aufgetreten
#define E_H1TF_NONETDRIVER       -2 // Didn´t found the network driver
#define E_H1TF_NOCONFREE   -3 // No more connections
#define E_H1TF_NOTCONNECTED        -5 // Connection has not yet been established
#define E_H1TF_RXFIFO_OVERFLOW  -15 // Reception Fifo has overflowed
#define E_H1TF_MEMALLOC       -97 // Not enough memory available
#define E_H1TF_GENERAL        -98 // general error occurred
#define E_H1TF_BADREF      -99 // Invalid reference invalid


BOOL WINAPI
H1TFSetAdaptername (LPCSTR AName);

long WINAPI
H1TFOpen (BYTE *DstMacAdr, BYTE *DSAP, DWORD LenDSAP, BYTE *SSAP, DWORD LenSSAP, BOOL
bPassive);


long WINAPI
H1TFRx (long Ref, void *Buf, DWORD MaxCnt, long RxTimeout);

long WINAPI
H1TFTx (long Ref, void *Buf, DWORD Cnt, long Timeout);

long WINAPI
H1TFGetStatus (long Ref);

long WINAPI
H1TFClose (long Ref);

void WINAPI
H1TFCloseAll (void);

long WINAPI
H1TFPacketInQ (long Ref);

long WINAPI
H1TFRxFifoClear (long Ref);

long WINAPI
H1TFTxUnlocked (long Ref, void *Buf, DWORD Cnt, long Timeout);
```

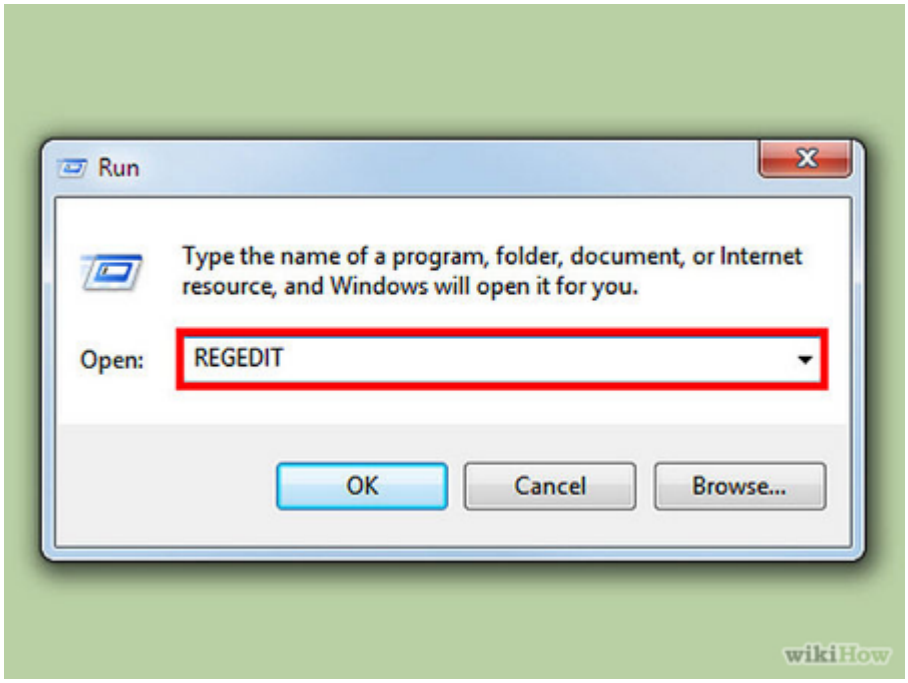
Change MAC-Address

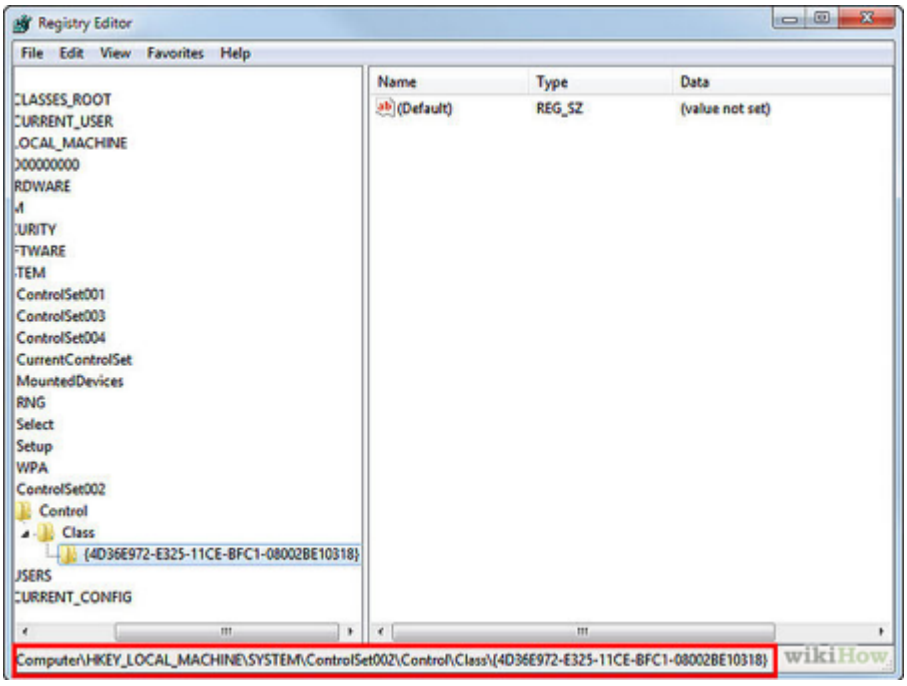
How to Change a Computer□s Mac Address in Windows
Source: http://m.wikihow.com/Change-a-Computer%27s-Mac-Address-in-Windows

There might be a time when you want to change the MAC address of your network adapter. The MAC address (Media Access Control address) is a unique identifier which is used to identify your computer in a network. Changing it can help you diagnose network issues, or just have a little fun with a silly name. See

Step 1 below to learn how to change the MAC address of your network adapter in Windows.

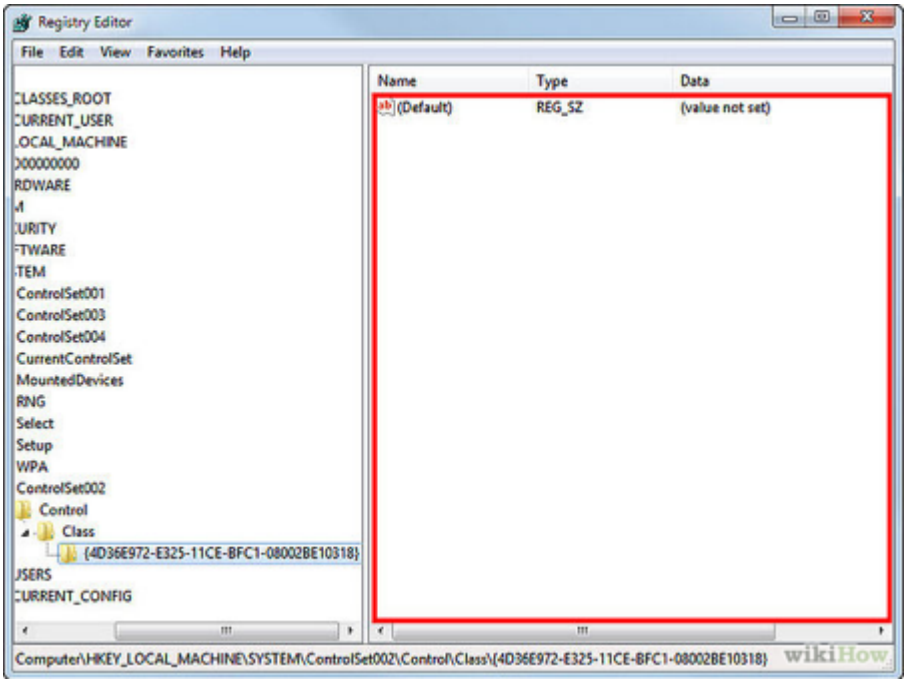
Using Registry Editor



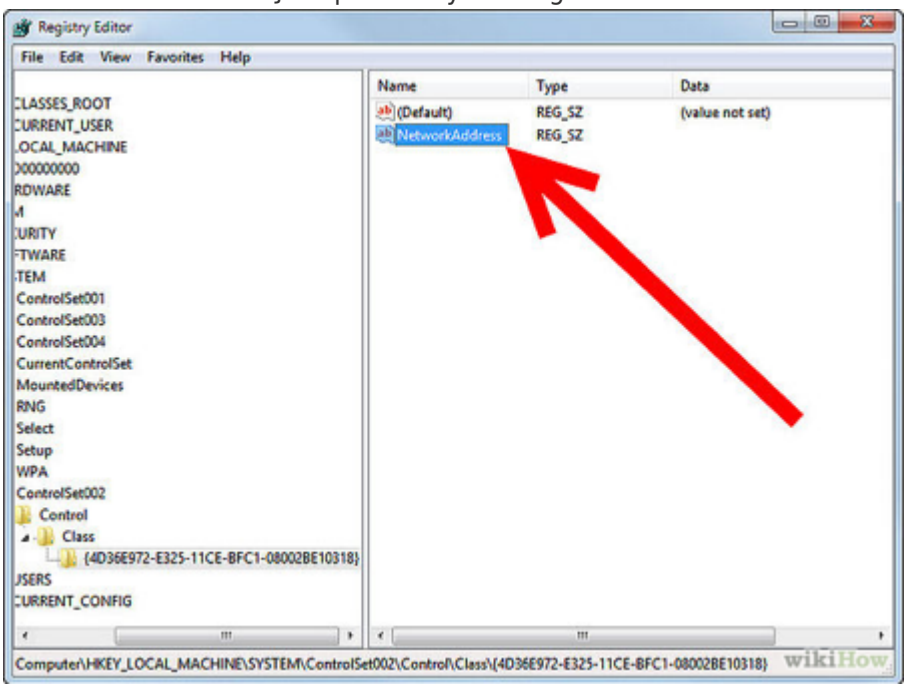
1. **Find your network adapter's ID information**.
   In order to easily identify your network adapter in the Windows Registry, you'll want to gather some basic information about it through the Command Prompt. You can open the Command Prompt by typing "cmd" into the Run box (Windows key + R).
   - Type ipconfig /all and press Enter.
     Note the Description and Physical Address for the active network device. Ignore devices that aren't active (Media Disconnected).
   - Type net config rdr and press Enter.
     Note the GUID, which is displayed between the "{}" brackets next to the Physical Address you recorded earlier.
2. 

Open the Registry Editor.
You can start the Registry Editor by opening the Run dialog box (Windows key + R) and typing “regedit”. This will open the Registry Editor, which will allow you to change the settings for your network card.

- Making incorrect changes to the registry can cause your system to malfunction.

3. 

Navigate to the registry key.
Go to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002BE10318}. Expand it by clicking the arrow. ??

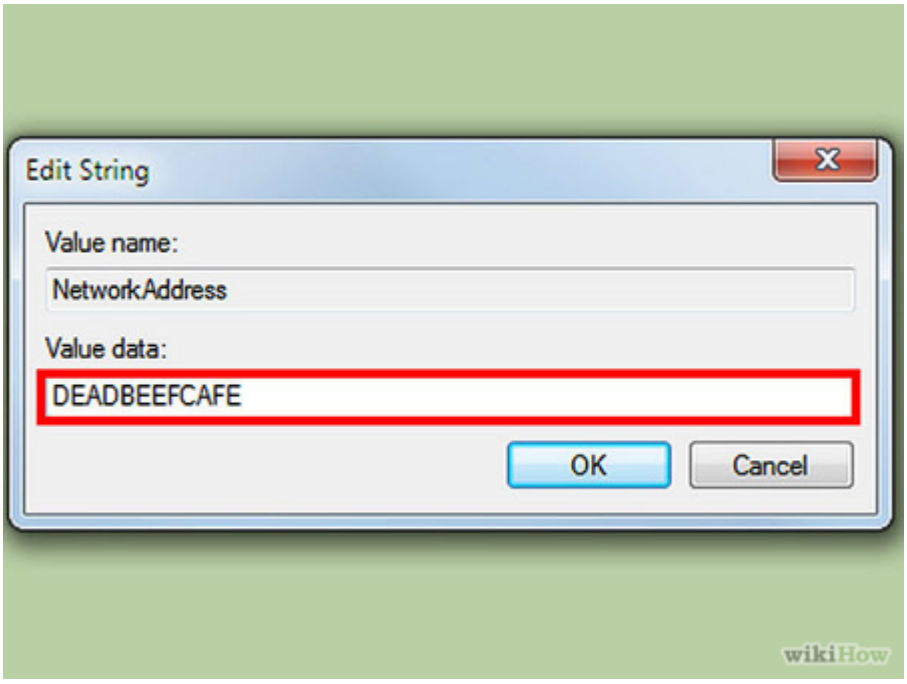
4. 

Find your adapter.
There will be several folders labeled “0000”, “0001”, etc. Open each of these and compare the DriverDesc field to the Description you noted in the first step. To be completely sure, check the NetCfgInstanceID field and match it with the GUID from the first step. ??

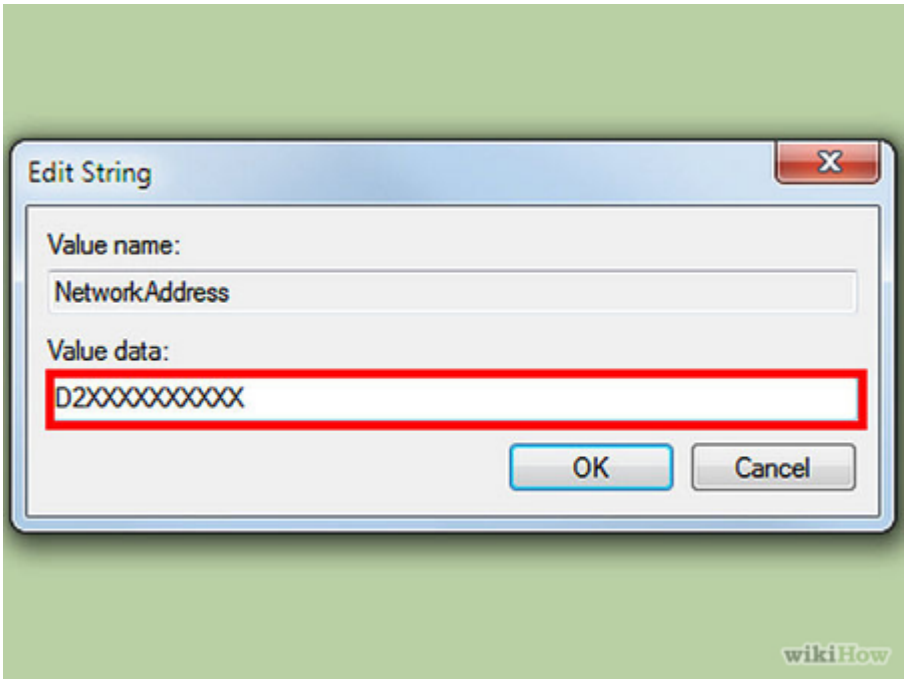
5.

Right-click on the folder that matches your device.
For example, if the “0001” folder matches your device, right-click on the folder. Select New ? String

Value. Name the new value “NetworkAddress”. ??

6. 

Double-click the new NetworkAddress entry.
In the “Value data” field, enter your new MAC address. MAC addresses are 12-digit values, and should be entered without any dashes or colons. For example, if you want to make the MAC address “2A:1B:4C:3D:6E:5F”, you would enter “2A1B4C3D6E5F”. ??

7. 

Ensure that the MAC address is formatted properly.
Some adapters (especially Wi-Fi cards) are unforgiving of MAC addresses changes if the first octet□s 2nd half isn□t a 2,6,A,E or begins with a zero. This requirement has been observed as far back as Windows XP and is formatted as:

- D2XXXXXXXXXX ?
- D6XXXXXXXXXX ?
- DAXXXXXXXXXX ?
- DEXXXXXXXXXX ?

8. **Reboot your computer to enable the changes**.
You can also disable and re-enable your adapter within Windows for the change to become effective

without rebooting. Just sliding the Wi-Fi's On/Off switch like the slider found on ThinkPad's and VaiO's won't satisfactorily disable/re-enable the card.?

9. **Check that the changes took effect**.
   Once you've rebooted the computer, open the Command Prompt and enter ipconfig /all and note the Physical Address of your adapter. It should be your new MAC address.[1]?

Using SMAC

1. **Download the SMAC program**.
   SMAC is a paid tool with a free demo that will allow you to quickly change your MAC address. It is compatible with Windows XP, Vista, and 7. Be sure to only download from trusted sources.
   - Install the software after downloading it. Most users will be fine with the default settings.
2. **Select your adapter**.
   When you open SMAC, you will see a list of all of your installed network devices. Select the adapter that you want to change the address for.?
3. **Enter your new address**.
   In the fields under “New Spoofed MAC Address”, enter in the new MAC address.?
4. **Ensure that the MAC address is formatted properly**.
   Some adapters (especially Wi-Fi cards) are unforgiving of MAC addresses changes if the first octet's 2nd half isn't a 2,6,A,E or begins with a zero. This requirement has been observed as far back as Windows XP and is formatted as:
   - D2XXXXXXXXXX
   - D6XXXXXXXXXX
   - DAXXXXXXXXXX
   - DEXXXXXXXXXX
5. **Click Options**.
   Select the “Automatically Restart Adapter” option from the menu. It should have a check mark next to it.?
6. **Click the “Update MAC” button**.
   Your network connection will be temporarily disabled as your MAC address is updated. Verify that the address changed in the grid listing your devices.[2]?

Using the Device Manager

1. **Open the Device Manager**.
   You can access the Device Manager from the Control Panel. It will be located in the System and Security section if you are using Category View.?
2. **Expand the Network Adapters section**.
   In your Device Manager, you will see a list of all of the hardware installed on your computer. These are sorted into categories. Expand the Network Adapters section to see all of your installed network adapters.
   - If you are not sure which adapter you are using, see Step 1 at the beginning of this article to find your device's Description.
3. **Right-click on your adapter**.
   Select Properties from the menu to open the network adapter's Properties window.?
4. **Click the Advanced tab**.
   Look for the “Network Address” or “Locally Administered Address” entry. Highlight it and you will

see a “Value” field on the right. Click the radio button to enable the “Value” field.
    - Not all adapters can be changed this way. If you can't find either of these entries, you will need to use one of the other methods in this article.
5. **Enter your new MAC address**.
   MAC addresses are 12-digit values, and should be entered without any dashes or colons. For example, if you want to make the MAC address “2A:1B:4C:3D:6E:5F”, you would enter “2A1B4C3D6E5F”.?
6. **Reboot your computer to enable the changes**.
   You can also disable and re-enable your adapter within Windows for the change to become effective without rebooting. Just sliding the Wi-Fi's On/Off switch like the slider found on ThinkPads and VaiOs won't satisfactorily disable/re-enable the card.?
7. **Check that the changes took effect**.
   Once you've rebooted the computer, open the Command Prompt and enter ipconfig /all and note the Physical Address of your adapter. It should be your new MAC address.

Table of Contents