

H1-TF-LINK

H1-Library für PC  SIMATIC S5 CP 143

Version 1.2.0.0



Betriebssystem

Windows

- 10
- 8
- 7
- Vista
- XP

Programmiersprachen

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

Hardware

PC mit Netzwerkkarte und MS-Windows Betriebssystem

Lieferumfang

Hauptverzeichnis	
H1TFLink.pdf	diese Datei □ Dokumentation
h1tflink.h	C/C++ Headerfile
h1tflink.lib	die Lib-Datei zum Linken

Verzeichnis 'WinPCAP'	Dateien für C / C++
h1tflink.dll	die DLL zur Verwendung des WinPCAP drivers
Verzeichnis 'WinPCAP/Drivers'	Hier finden Sie den den WinPCAP Treiber, falls Sie diesen verwenden möchten. Diesen mit mit Admin-Rechten installieren, und darauf achten dass der WinPCAP Service automatisch gestartet wird.
Verzeichnis WINPCAP/Demo	C++ Beispielprogramm für WinPCAP
Verzeichnis □NDIS□	die DLL zur Verwendung des NDIS-Treibers
Verzeichnis NDIS/Drivers	Hier finden Sie den den WinPCAP Treiber, falls Sie diesen verwenden möchten. Diesen mit mit Admin-Rechten installieren, und darauf achten dass der WinPCAP Service automatisch gestartet wird.
Verzeichnis NDIS/Demo	C++ Beispielprogramm für NDIS

Funktionsweise:

H1TFLINK-Lib ist eine DLL für MS-Windows (NT/XP/Vista/7/8/10), welche die Anbindung eines PC an Industrial Ethernet über das Siemens-H1-Protokoll auf ISO/MAC-Basis. Mit einfachen Funktionen kann der Anwender schnell mit C, C++, Delphi, Visual Basic oder auch Excel H1- Verbindungen aufbauen und Daten

senden und Empfangen. Zur Kopplung wird nur die MAC-Adresse, DSAP, SSAP des Partners benötigt.

Bitte beachten Sie, falls der Partner ein bestimmte MAC-Adresse als Absender fordert, müssen Sie für Ihren Adapter unter Windows die richtige MAC-Adresse setzen. Im Anhang finden Sie eine Anleitung

Funktionsbeschreibung im Detail

Bitte beachten Sie: Die Funktionen werden Synchron ausgeführt, was zur Folge hat, dass die Funktion erst nach Erfüllung der Aufgabe zum Aufrufer zurückkehrt. Zum Asynchronen Betrieb rufen Sie diese Funktionen einfach von einen separaten Thread aus auf, welcher für die Kommunikation des System zuständig ist.

Aktuell wird ein NDIS-Treiber und der WinPCAP-Treiber unterstützt. Je nach Treiber sind die entsprechenden DLLs zu verwenden.

Ja nachdem, welcher Treiber verwendet wird, ist die entsprechende DLL zu verwenden.

ACHTUNG:Der Name der DLL immer „h1tflink.dll“. Sie finden die passende DLL im entsprechenden Verzeichnis.

Ethernet-Adaptername

Das Netzwerk muss installiert sein. Der Treiber setzt auf der NDIS-Ebene auf. Die Treiber werden dynamisch geladen. Grundsätzlich versucht der Treiber den ersten NDIS-fähigen Adapter im System zu finden dieser wird dann auch verwendet. Wenn Sie also nur einen Netzwerkadapter verwenden, so wird dieser auch benutzt. Möchten Sie eine bestimmten Adapter ansprechen, so muss dieser vor der H1TFOpen-Routine mit der Funktion H1TFSetAdaptername, gesetzt werden. Achtung auch DFÜ-Adapter können NDIS-fähig sein. Damit läuft jedoch das H1-Protokoll nicht. Den Adapternamen finden Sie in der Registry als ServiceName.

H1TFSetAdaptername

Setzt den Netzwerkadapter

Aufrufparameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1		Adaptername	<p>Windows speichert die Informationen im folgenden Registrypfad: HKEY_LOCAL_MACHINE * Software * Microsoft * Windows NT * CurrentVersion * NetworkCards * 1 * ServiceName z.B: {00030...} * 2 * ServiceName</p> <p>Möchten Sie die 1. Karte verwenden, rufen Sie also z.B. auf H1TFSetAdaptername ({"00030..."}) auf</p>

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung / Reaktion
0	Adaptername ungültig	
1	Adaptername okay	erfolgreich

C/C++

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

H1-Funktionen

H1TFOpen

H1TFOpenEx

Dient zur Initialisierung der Verbindung. Wurde die Funktion erfolgreich ausgeführt, wird die Verbindung intern verwendet.

- Verbindung ist aktiv (bPassive = 0):
 - Der Treiber beginnt sofort den Verbindungsaufbau mit dem Partner zu starten
- Verbindung ist passive (bPassive = 1):
 - Der Treiber wartet auf den Verbindungsaufbau durch den Partner

Der Verbindungsstatus kann mit [H1TFGetStatus](#) abgefragt werden. Die Verbindung wird im Background automatisch aufgebaut. H1TFOpenEx erlaubt zusätzlich den Sende- / Empfangstimeout bei Fetch/Write anzugeben. Im Standard sind diese 5000ms. So empfiehlt es sich, bei Fetch/Write die H1TFOpenEx Funktion zu verwenden.

Aufrufparameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1	Zeiger auf ein Array von 6 Bytes	MACAdr	Ziel-MAC-Adresse z.B. für 08:00:06:01:00:01 BYTE Mac[6] = {0x0,0x08, 0x00, 0x06, 0x01, 0x00, 0x01}
2	Zeiger auf „0“-terminierte Zeichenkette (C-String, 32-Bit Zeiger)	SSAP	eigener SAP Es werden maximal 8 Zeichen verwendet.
3	32 Bit unsigned	SSAPLen	Länge des eigenen SAP. Es werden maximal 8 Zeichen verwendet
4	Zeiger auf „0“-terminierte Zeichenkette (C-String, 32-Bit Zeiger)	DSAP	SAP des Partners. Es werden maximal 8 Zeichen verwendet.
5	32 Bit unsigned	DSAPLen	Länge des SAP des Partners. Es werden maximal 8 Zeichen verwendet

Nr.	Speicherbreite	Bezeichnung	Funktion
6	32-Bit Wert (BOOL)	bPassive	Gibt an, ob der PC oder der Partner der die Verbindung aufbaut. 0: PC baut die Verbindung auf 1: der Partner baut die Verbindung auf
7	32 Bit unsigned	RxTimeout	Empfangs Timeout in ms bei Fetch/Write nur bei H1TFOpenEx
8	32 Bit unsigned	TxTimeout	Sende Timeout in ms bei Fetch/Write nur bei H1TFOpenEx

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung
>= 0	Alles OK	Die Rückgabe ist die Referenznummer für diese Verbindung und muss bei allen anderen Funktionen als Eingangsparameter Ref verwendet werden.
<0	siehe Returncodes	

C/C++

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

extern long WINAPI
H1TFOpenEx (BYTE *DstMacAdr, BYTE *DSAP, DWORD LenDSAP, BYTE *SSAP, DWORD LenSSAP, BOOL
bPassive, DWORD RxTimeout, DWORD TxTimeout);
```

H1TFClose

Dient zur Deinitialisierung der Verbindung, Speicher wird freigegeben und die Verbindung wird getrennt.

Aufrufparameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32 Bit unsigned	Ref	Die Referenz der Verbindung, welche mit H1TFOpen generiert wurde. Dient zur internen Identifikation der Verbindung.

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung / Reaktion
0	alles OK	Speicher wieder freigegeben und Verbindung, wenn vorhanden geschlossen
<0	siehe Returncodes	

C/C++

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

Empfangen und Senden

Der Treiber besitzt einen internen FiFo für empfangene Datenpakete.

Im Hintergrund wird dieser FiFo bedient. Dabei achtet der Treiber darauf, dass kein Overflow auftritt.

Wird die Grenze des Puffers erreicht, teilt der Treiber dem Partner rechtzeitig mit, dass die Queue gefüllt ist.

Aus diesem Grund ist es notwendig, dass die Applikation die Daten möglichst regelmässig mit H1TxDataH1TFRx abholt.

H1TFPacketInQ

Prüft, ob mindestens ein Packet in der Empfangs-Queue bereitsteht.

Aufruf Parameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32 Bit unsigned	Ref	Die Referenz der Verbindung, welche mit H1TFOpen generiert wurde. Dient zur internen Identifikation der Verbindung

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung / Reaktion
0	kein Paket vorhanden	
>0	mindesten ein Paket vorhanden	
<0	siehe Returncodes	

C/C++

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

H1TFRxFifoClear

Löscht alle Pakete im RxFifo für die angegebene Verbindung.

Dies kann nötig sein. Selbst nach Verbindungsunterbrechung bleiben die empfangen Pakete und der Queue. Diese können entweder gesamt ausgelesen oder verworfen werden.

Aufrufparameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32 Bit unsigned	Ref	Die Referenz der Verbindung, welche mit H1TFOpen generiert wurde. Dient zur internen Identifikation der Verbindung

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung / Reaktion
0	FiFo wurde gelöscht	
<0	siehe Returncodes	

C/C++

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

Lesen / Schreiben

Funktion	Beschreibung / Zweck
H1TFRx	Versucht Daten zu empfangen.
H1TFTx	Sendet Daten an den Partner.
H1TFTxUnlocked	H1TFTxUnlocked sendet ein Packet an den Partner ohne einen Lock auf die Verbindung auszuführen. Diese Funktion wird verwendet, während z.B. in einem anderen Thread H1TFRx läuft. Während H1TFRxData läuft, ist die Verbindung nämlich geblockt. Ein normaler H1TFTx würde blockieren, bis die H1TFRx wieder zurückkehrt.

H1TFRx dient zum Empfangend der Daten.

Bitte beachten Sie: Selbst bei Verbindungsunterbrechnung bleiben die empfangene Datenpaket in der Queue vorhanden.

Der Anwender ist für das Leeren der Queue verantwortlich.

Aufrufparameter

Die Empfangs- und Sendefunktionen besitzen die selben Eingangsparameter:

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32 Bit unsigned	Ref	Die Referenz der Verbindung, welche mit H1TFOpen generiert wurde. Dient zur internen Identifikation der Verbindung
2	32-Bit Adresse	Buffer	Die Adresse auf den Quell- bzw. Zielspeicher im PC.
3	32 Bit unsigned	Cnt	Bei Rx, Anzahl der maximal zu empfangenden Datenbytes. Bei Tx Anzahl der zu sendenden Datenbytes.
4	32 Bit signed	Timeout	Timeout in ms für Senden bzw. Empfangen 0 = warte nicht

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Reaktion
>=0	Anzahl der gesendeten / empfangenen Datenbytes	
<0	siehe 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

Liefert den Status der Verbindung

Aufrufparameter

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32 Bit unsigned	Ref	Die Referenz der Verbindung, welche mit H1TFOpen generiert wurde. Dient zur internen Identifikation der Verbindung

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Bedeutung / Reaktion
0	Verbindung nicht vorhanden	
1	Verbindung vorhanden	
<0	siehe Returncodes	

C/C++

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

H1TFCloseAll

Dient zum schliessen aller Verbindungen, Speicher wird freigegeben und die ISO-Verbindungen werden getrennt.

C/C++

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

Fetch / Write

Lesen / Schreiben

Funktion	Beschreibung / Zweck
H1TFRdB	byteweise lesen aus der SPS
H1TFRdW	wortweise (16-Bit) lesen aus der SPS
H1TFWrB	byteweise schreiben in die SPS
H1TFWrW	wortweise (16-Bit) schreiben in die SPS

Aufrufparameter

Die Empfangs- und Sendefunktionen besitzen die selben Eingangsparameter:

Nr.	Speicherbreite	Bezeichnung	Funktion
1	32-Bit unsigned	Ref	Die Referenz der Verbindung, welche mit IPS7Open generiert wurde. Dient zur internen Identifikation der Verbindung
2	32-Bit unsigned	Typ	Die Auswahl des Speicherbereichs in der SPS (DB, Eingang, Ausgang, Merker), welcher bearbeitet werden soll: 'D' = 68 dez. steht für Datenbaustein nur für wortweisen Zugriff 'E' = 69, dez. steht für Eingänge 'A' = 65 dez. steht für Ausgänge 'M' = 77 dez. steht für Merker

Nr.	Speicherbreite	Bezeichnung	Funktion
3	32-Bit unsigned	DBNr	Datenbausteinnummer, diese wird nur beim Typ 'D' verwendet. Ansonsten steht dort der Wert „0,,
4	32-Bit unsigned	Ab	Erstes Wort bzw. Byte ab dem gelesen bzw. geschrieben werden soll. Bei Wortoperationen Startwort Bei Byte, Doppelwort und Realfunktionen Startbyte Bei Timer oder Zähler ist dies die Nummer des ersten Elements, welches gelesen werden soll
5	32-Bit unsigned	Anz	Anzahl der Einheiten (Byte, Worte, Doppelworte Real, oder Einheiten, z.B. Timer), die gelesen bzw. geschrieben werden sollen
6	32-Bit Pointer	Buffer	Die Adresse auf den Quell- bzw. Zielspeicher im PC. Bei den Wortfunktionen ist dies ein Zeiger auf ein Feld von 16-Bit breiten Worten Bei den Bytefunktionen ist das eine Adresse auf ein Feld mit 8-Bit breiten Bytes

Rückgabewert

Die Funktionen liefert einen „32 Bit signed“ als Rückgabewert mit folgender Bedeutung:

Wert	Fehlerbeschreibung	Reaktion
0	Die Aktion war erfolgreich	
<0	siehe Returncodes	

C/C++

```

long WINAPI
H1TFRdB (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPBYTE Buffer);

long WINAPI
H1TFRdW (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPWORD Buffer);

long WINAPI
H1TFWrB (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPBYTE Buffer);

long WINAPI
H1TFWrW (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPWORD Buffer);

```

Returncodes

Konstante im Includefile	Wert	Bedeutung
E_H1TF_OKAY	0	erfolgreich, kein Fehler aufgetreten
E_H1TF_TIMEOUT	-1	Timeout aufgetreten
E_H1TF_NONETDRIVER	-2	Netzwerktreiber nicht gefunden
E_H1TF_NOCONFREE	-3	keine Verbindungen mehr frei
E_H1TF_NOTCONNECTED	-5	Verbindung wurde noch nicht hergestellt
E_H1TF_TXERR	-6	Sendefehler, Ethernetpaket konnte nicht gesendet werden
E_H1TF_RXFIFO_OVERFLOW	-15	Empfangsfifo ist übergelaufen
E_H1TF_MEMALLOC	-97	nicht genug Speicher verfügbar
E_H1TF_GENERAL	-98	genereller Fehler aufgetreten
E_H1TF_BADREF	-99	übergeben Referenz ungültig
E_H1TF_DEMO_EXPIRED	0x1234	Demozeit ist abgelaufen
FETCHWRITE_H1_E_OKAY	0	Demozeit ist abgelaufen

FETCHWRITE_H1_E_TIMEOUT	-1	Timeout aufgetreten
FETCHWRITE_H1_E_NODATA	2	Der gewünschte Datenbereich existiert nicht. Z.B Datenbaustein nicht vorhanden oder zu klein.
FETCHWRITE_H1_E_DATAAREA_NOT_VALID	-10	der angegebene Datenbereich z.B 'D' ist, ungültig, oder mit der verwendeten Funktion nicht möglich

C/C++ - Header

```

/*
    H1TFLink.h
    Version 1.2.0.0.

*/

#ifndef __H1TF_INCLUDE_INCLUDED__
#define __H1TF_INCLUDE_INCLUDED__

#define E_H1TF_OKAY                0 // okay, kein Fehler aufgetreten
#define E_H1TF_TIMEOUT             -1 // Timeout aufgetreten
#define E_H1TF_NONETDRIVER        -2 // Netzwerktreiber nicht gefunden
#define E_H1TF_NOCONFREE          -3 // keine Verbindungen mehr frei
#define E_H1TF_NOTCONNECTED       -5 // Verbindung wurde noch nicht hergestellt
#define E_H1TF_TXERR              -6 // Sendefehler, Paket konnte nicht gesendet werden
#define E_H1TF_RXFIFO_OVERFLOW    -15 // Empfangsfifo ist übergelaufen
#define E_H1TF_MEMALLOC           -97 // nicht genug speicher verfügbar
#define E_H1TF_GENERAL            -98 // genereller Fehler aufgetreten
#define E_H1TF_BADREF             -99 // übergeben Referenz ungültig
#define E_H1TF_DEMO_EXPIRED       0x1234

#define FETCHWRITE_H1_E_OKAY        0 /* alles OK */
#define FETCHWRITE_H1_E_TIMEOUT     -1 /* Zeitüberlauf */
#define FETCHWRITE_H1_E_NODATA      2 /* Datenbereich existiert nicht */
#define FETCHWRITE_H1_E_DATAAREA_NOT_VALID -10 /* der angegebene Datenbereich z.B 'D' ist, ungültig, oder mit der verwendeten Funktion nicht möglich */

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
H1FTTxUnlocked (long Ref, void *Buf, DWORD Cnt, long Timeout);

long WINAPI
H1TFRdB (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPBYTE Buffer);

long WINAPI
H1TFRdW (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPWORD Buffer);

long WINAPI
H1TFWrB (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPBYTE Buffer);

long WINAPI
H1TFWrW (long Ref, DWORD Typ, DWORD DBNr, DWORD Ab, DWORD Anz, LPWORD Buffer);

long WINAPI
H1TFOpenEx (BYTE *DstMacAdr, BYTE *DSAP, DWORD LenDSAP, BYTE *SSAP, DWORD LenSSAP, BOOL
bPassive, DWORD RxTimeout, DWORD TxTimeout);

#endif //__H1TF_INCLUDE_INCLUDED__
```

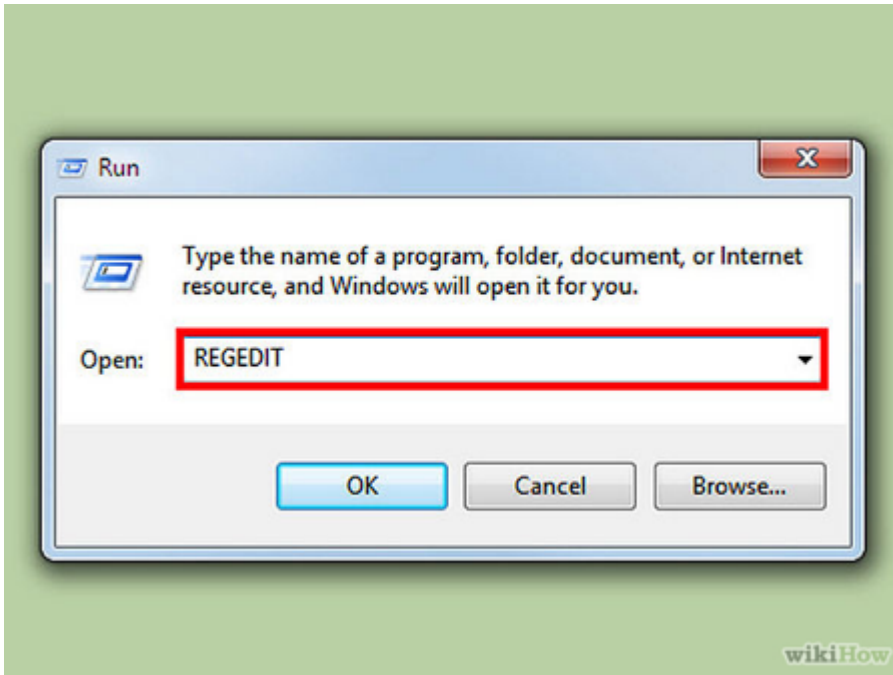
Change MAC-Address

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.

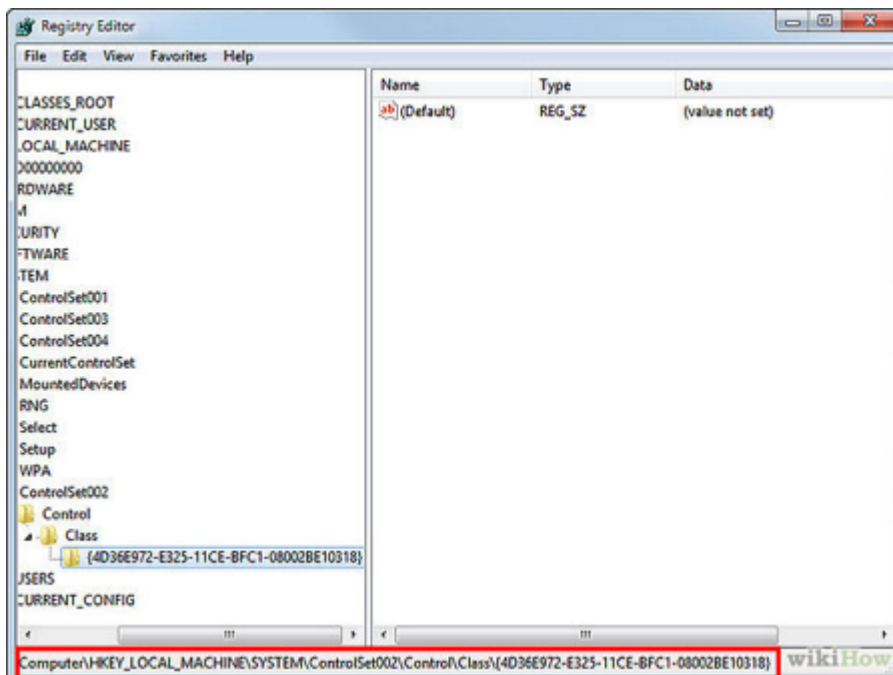
In order to easily identify your network adapter in the Windows Registry, you'll want to gather some basic information about it through the Command Prompt. You can open the Command Prompt by typing „cmd“ into the Run box (Windows key + R).

- 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.

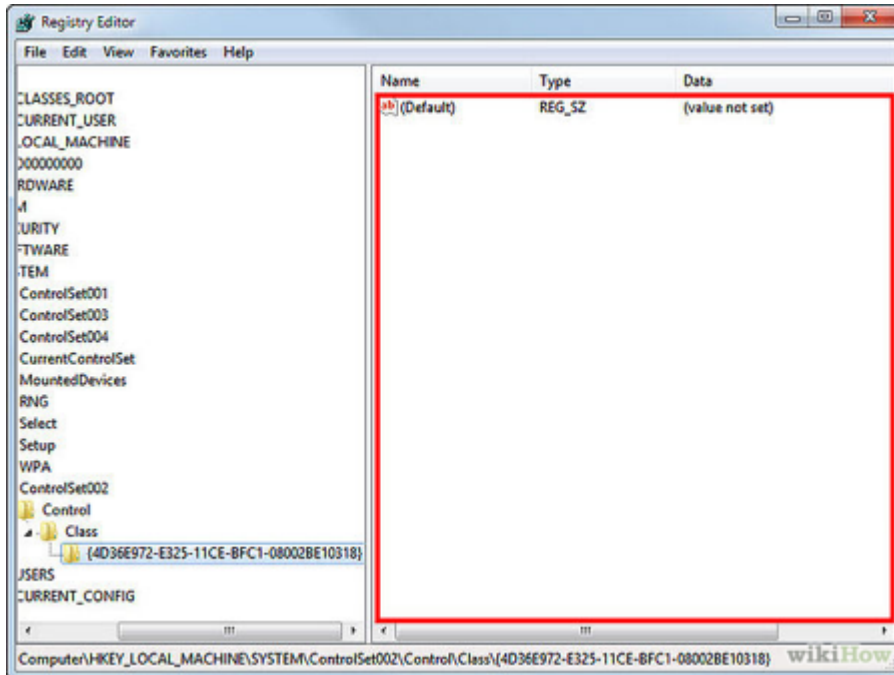


2. Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Control\Class\{4D36E972-E325-11CE-BFC1-08002BE10318}

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.

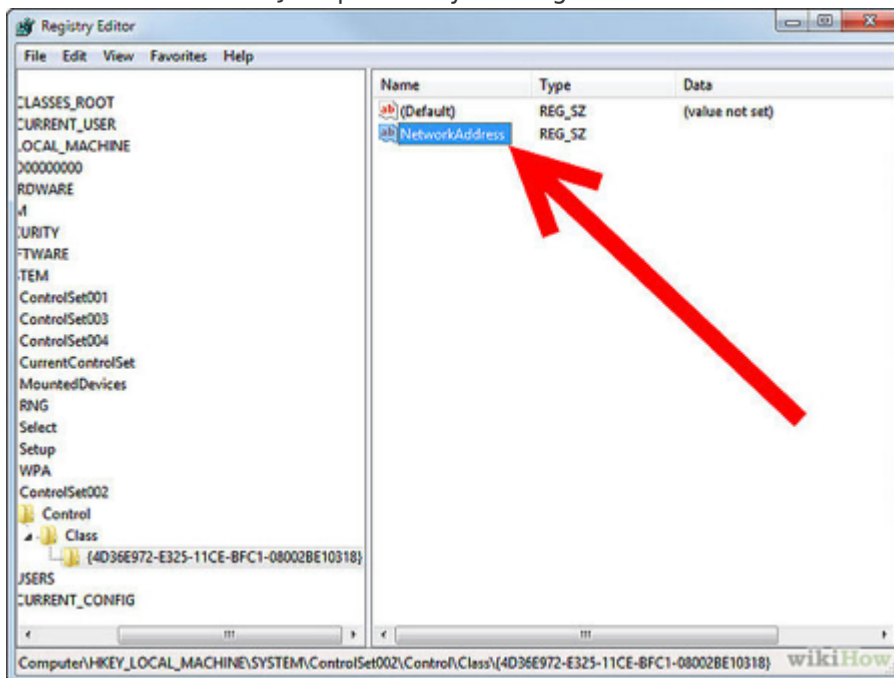
- Making incorrect changes to the registry can cause your system to malfunction.



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.

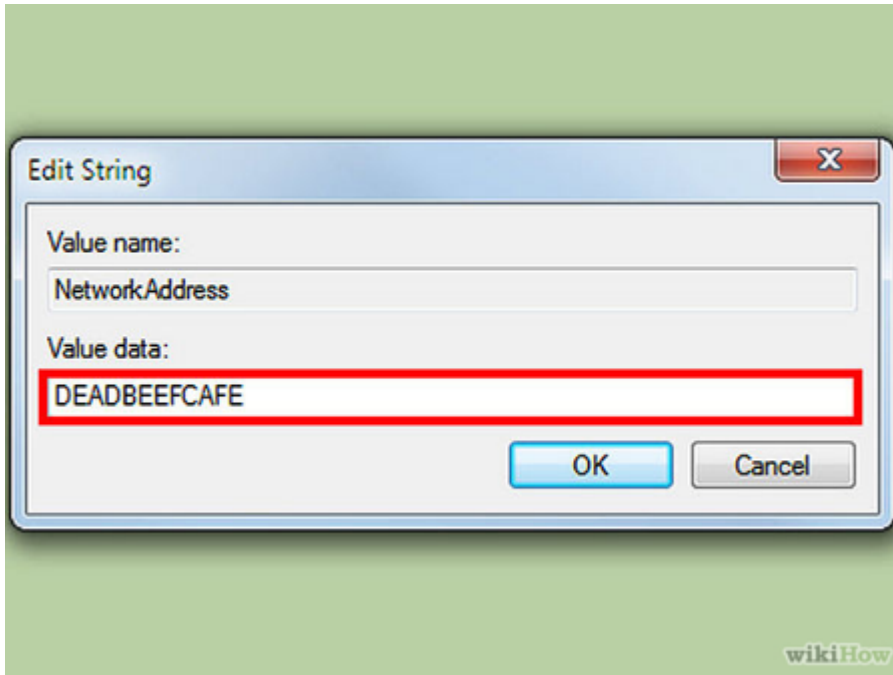
Find your adapter.

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.

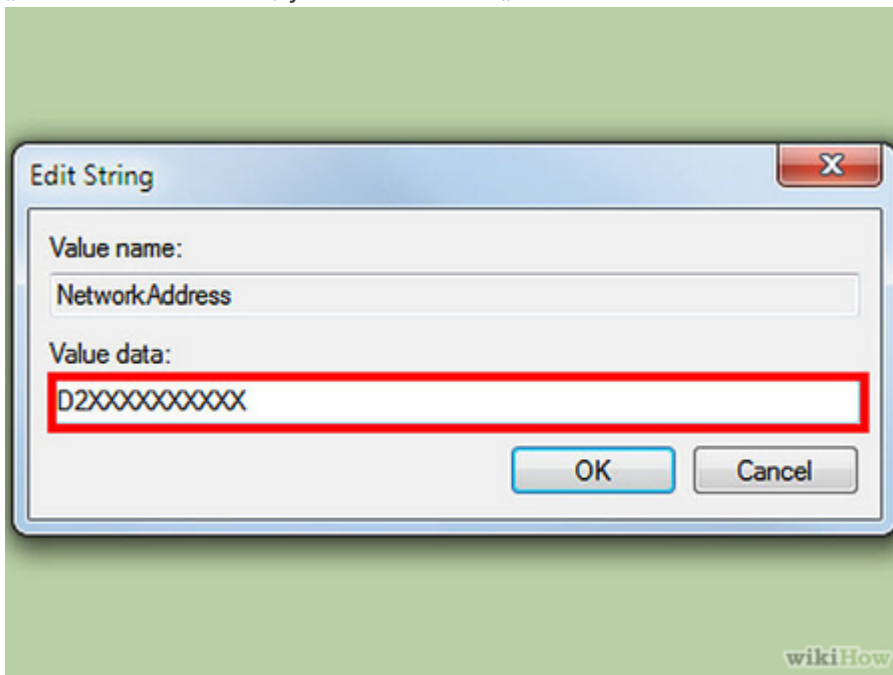
For example, if the „0001“ folder matches your device, right-click on the folder. Select New ? String Value. Name the new value „NetworkAddress“. ??



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 Vaio's 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.

Inhaltsverzeichnis

Betriebssystem	2
Programmiersprachen	2
Hardware	2
Lieferumfang	2
Funktionsweise:	2
Funktionsbeschreibung im Detail	3
Ethernet-Adaptername	3
H1TFSetAdaptername	3
Aufrufparameter	3
Rückgabewert	4
H1-Funktionen	4
H1TFOpen	4
H1TFOpenEx	4
Aufrufparameter	4
Rückgabewert	5
H1TFClose	5
Aufrufparameter	5
Rückgabewert	5
Empfangen und Senden	5
H1TFPacketInQ	6
Aufruf Parameter	6
Rückgabewert	6
H1TFRxFifoClear	6
Aufrufparameter	6
Rückgabewert	6
Lesen / Schreiben	7
Aufrufparameter	7
Rückgabewert	7
H1TFGetStatus	7
Aufrufparameter	7
Rückgabewert	8
H1TFCloseAll	8
Fetch / Write	8
Lesen / Schreiben	8
Aufrufparameter	8
Rückgabewert	9
Returncodes	9
C/C++ - Header	10
Change MAC-Address	11
Using Registry Editor	11
Using SMAC	15
Using the Device Manager	15

