



Anhang 1: Installationen unter Windows

Inhaltsverzeichnis

- Die Plug&Play (PnP) Funktion 2**
- Treiber-Installationen unter Windows 3**
 - PCI mit Windows 95/98/ME 3
- PCI und Windows NT 7**
- Die Windows - NT Treiber Installation 7**
 - Ermittlung der Adresslage einer I/O-Karte unter NT 8
 - Informationen zur Reaktionszeit innerhalb des NTP-Treibers 10
- PCI und Windows 2000 beziehungsweise Windows XP 11**
 - Allgemein 11
 - Voraussetzung zur Installation 11
 - Fehler im System - was nun? 11
 - Und los gehts... 12
 - Die manuelle Installation 12
 - Hardwaretest und PCI-Informationen 19
 - Mehr Komfort 20
- Kernel Drivers 21**
 - Wie Kernel-Treiber mit PCI-Karten funktionieren 21
 - Erläuterung der Abkürzungen 26



Die Plug&Play (PnP) Funktion

Nach dem Einschalten des Rechners findet das BIOS des Computers alle PCI-Devices und fragt jedes Device danach ab, welche Ressourcen gebraucht werden. Die benötigten Ressourcen bestehen meistens aus I/O-Adressen, IRQ-Nummern, DMA Kanälen und verwendeten Speicherbereichen. Überschneiden sich bei unterschiedlichen Karten die jeweils benötigten Ressourcen, so versucht das BIOS durch umkonfigurieren der PCI-Karten die Funktion beider in einem System zu ermöglichen. Ist dies einmal nicht der Fall, so schaltet das BIOS eine der beiden Karten einfach ab. Dies kommt aber relativ selten vor.

Die verwendeten Ressourcen jeder PCI-Karte werden über die ESCD-Datenbank des BIOS dem verwendeten Betriebssystem zur Verfügung gestellt. Desweiteren werden diese Informationen auch in einem bis zu 256 Bytes großem Speicherbereich auf der PCI-Karte abgelegt, dem Configuration-Space. Aus diesem Speicherbereich holt sich auch das BIOS die Informationen über die benötigten Ressourcen und stellt es anderen Anwendungen über spezielle Bios-Interrupt-Aufrufe zur Verfügung.

Da diese BIOS-spezifischen Parametrierungsregister in jedem System dynamisch geführt werden und kein einheitlicher, fester Speicherplatz (beispielsweise MEM-Adresse) vorgegeben wird, kann das Lesen dieser Informationen auch nur durch einen für alle Motherboard-Hersteller vorgegebenen Software-Interrupt erfolgen. Über Status- und Command-Register können Informationen über jede verwendete PCI-Karte ausgelesen beziehungsweise gegebenenfalls auch gesetzt werden.

Die Vendor-ID ist die Kennziffer des Herstellers der verwendeten Karte, die Device-ID gibt die Geräte-nummer des verwendeten Produkts an. Die Vendor-ID Kennziffern (kurz: VID) wird vom PCI-SIG (PCI Special Interest Group) vergeben, die Device-ID Nummern werden den Produkten vom jeweiligen Hersteller zugewiesen.

Die PCI-SIG ist ein Konsortium, das alle Spezifikationen des PCI-Busses kontrolliert. Über diese Institution können auch die detaillierten Spezifikationen zum PCI-Bus bezogen werden. In diesen finden sich auch die genauen Erläuterungen zu den Registern und Interrupt-Aufrufen.



Treiber-Installationen unter Windows

Vor der Installation der Treiber gehen wir davon aus, dass Sie die Karte in Ihren Rechner eingebaut haben. Nach dem Zusammenbauen des Rechnergehäuses müssen nun die entsprechenden Treiber installiert werden. Achten Sie darauf, dass Sie ausschließlich die zu Ihrem installierten Betriebssystem passenden Treiber verwenden. Dies scheint nicht trivial zu sein, denn bisweilen kommt es vor, dass jemand versucht die Karte mit Win9x-Treibern unter NT zu installieren, beziehungsweise umgekehrt mit NT-Treibern WIN9x zu Leibe rückt (unser Hotline könnte Ihnen einiges dazu erzählen). Nun kann man den PC einschalten und mit der Treiberinstallation beginnen.

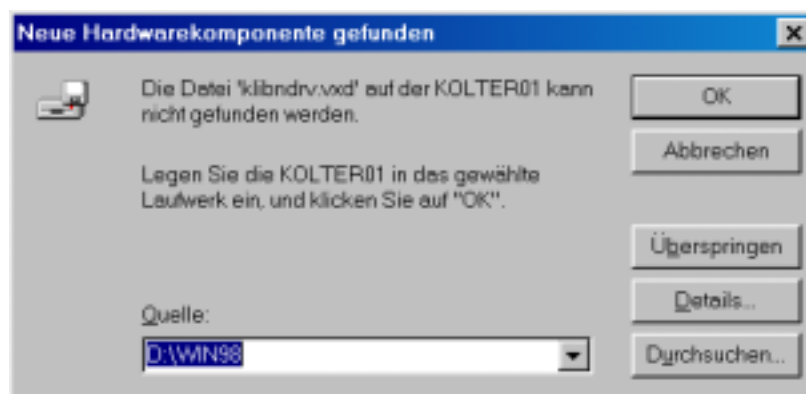
PCI mit Windows 95/98/ME

Unsere Sourcen und Treiber-Software unterstützen in erster Linie die Ansteuerung und Abfrage der karteninternen I/O-Register zu jeder KOLTER-I/O-Karte. Die Beispielprogramme (meistens in C oder CPP) sind mit Absicht relativ kurz gehalten, damit speziell die kartenspezifischen Eigenschaften der jeweiligen PC-Karte und deren Programmierung deutlich wird.

Oft sind es nur ein paar Code-Zeilen und I/O-Adressen die man kennen muss, um die ganze Sache schnell, bequem und erfolgreich in Betrieb zu nehmen. Die Beispiele sind so konzipiert, dass die jeweilige Messkarte initialisiert, ausgelesen und dann die Ergebnisse auf dem Monitor als Zahlenwerte (in einer Tabelle oder Liste) dargestellt werden. Neben kartenspezifischen Ausgaben wie Volt, Milliampere, Pegel oder Frequenz, bleiben weitergehende Datenauswertungen und Umrechnungen (zum Beispiel für Steuerungszwecke oder Statistiken) der Kreativität des Programmierers überlassen. Die Sourcen (auf der CD) sind, soweit nicht anders angegeben, frei verfügbar und können in eigene Softwareentwicklungen direkt übernommen werden.

Für alle Windows-Versionen werden spezielle, zugehörige Treiber geliefert. Die CD enthält beispielsweise Windows-9x Treiber (wie VXD, INF...) und diverse PCI-Diagnose-Tools, die neben den PCI-Karten auch unsere ISA-Bus-Karten unterstützen. Mit den PCI-Tools wie PCITree, HWT... (im CD-Verzeichnis:\Anwendungen oder PCI\Utils_PCI) können beispielsweise I/O-Adressen, Konfiguration oder der Revisionsstand zur jeweiligen PCI-Karte aus dem Rechner-BIOS ausgelesen werden.

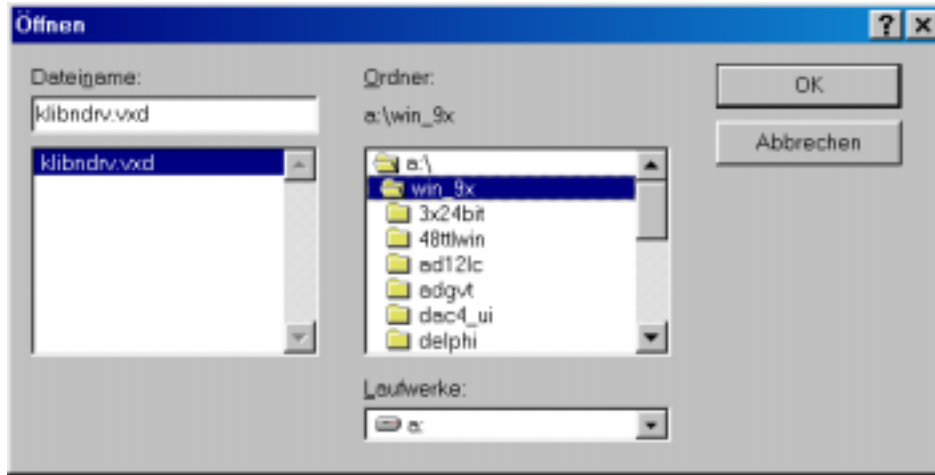
Die Installation gestaltet sich einfach: Nach dem Einbau der Karte den Rechner einschalten. Beim Bootvorgang meldet das BIOS (Version bis 2.1) im PCI-Device-Listing unter der Vendor ID 1001 und der Device ID 00xx ein unbekanntes PCI Device. Ab BIOS-Version 2.2 und aufwärts lautet hier die Meldung bei denselben IDs „Data acquisition and signal processing controller“. Beide Hinweise zeigen an, dass vom System erkannt wurde, dass eine neue PCI-Karte eingebaut worden ist. Kurz darauf öffnet sich, noch während des Windows-Bootvorgangs, das Fenster „Neue Hardwarekomponente gefunden“.



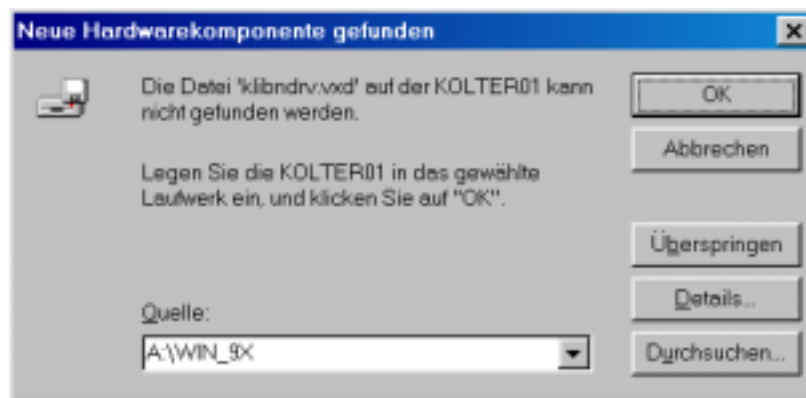
* Eine CD ist in Vorbereitung und wird nach der Fertigstellung die Disketten ersetzen.



Legen Sie nun unsere Original-CD ein und betätigen Sie die Schaltfläche „Durchsuchen“. Das Fenster „Öffnen“ wird nun angezeigt. Hier muss das CD-Laufwerk selektiert werden in dem sich



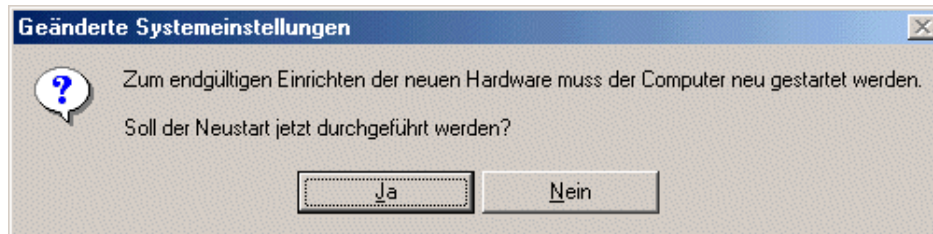
die CD mit den Treibern (im Pfad ...\Drivers\WIN9x_ME) befindet. Daraufhin wird unter „Ordner“ das Verzeichnis WIN9x_ME angezeigt, das durch Doppelklicken geöffnet werden muss. Im linken Fenster „Dateiname“ erscheint jetzt der Name des virtuellen Gerätetreibers klibndrv.vxd. Klicken Sie auf „OK“. Sie gelangen wieder in das Fenster „Neue Hardwarekomponente gefunden“ zurück.



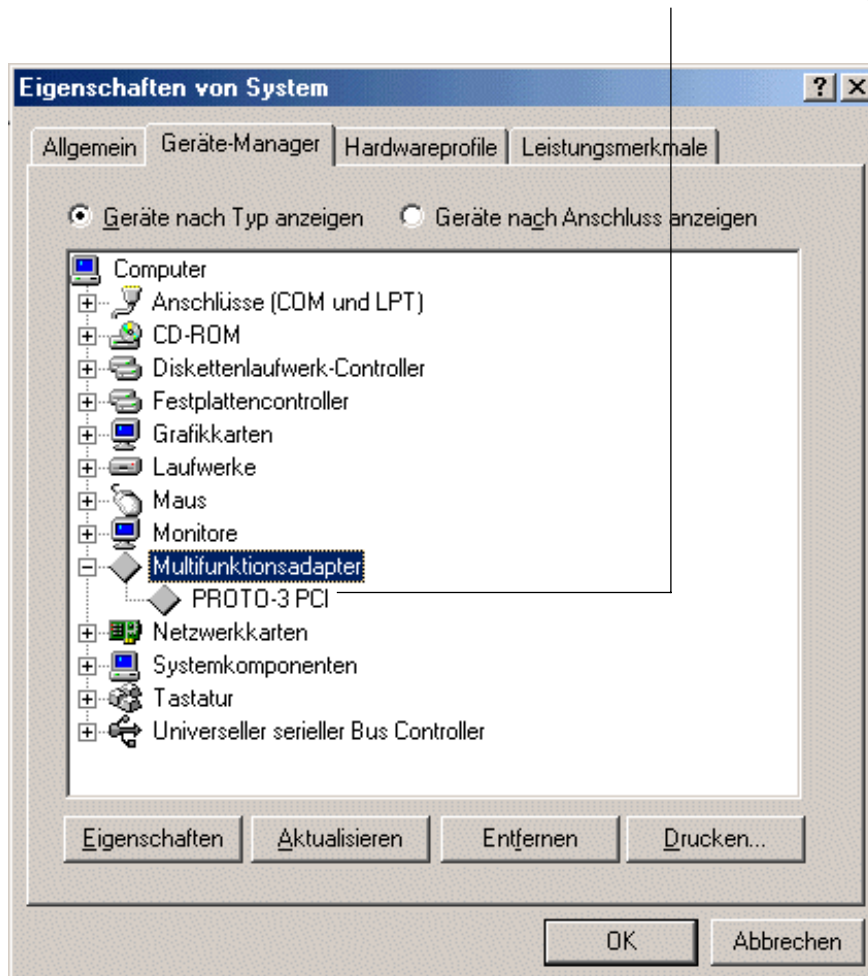
Hier ist jetzt als Quellverzeichnis der Ordner WIN_9x auf dem entsprechenden Laufwerk (A für Diskette bzw. der Buchstabe für Ihr CD-Laufwerk) eingetragen. Mit „OK“ wird die Installation des Treibers eingeleitet, was über das eingblendete Fenster „Treiber werden installiert“ angezeigt wird.



Im nächsten Fenster werden Sie zum Neustart des PCs aufgefordert. Dies ist unbedingt nötig, da erst nach einem erneuten hochfahren des Rechners die entsprechenden Treiber geladen werden.

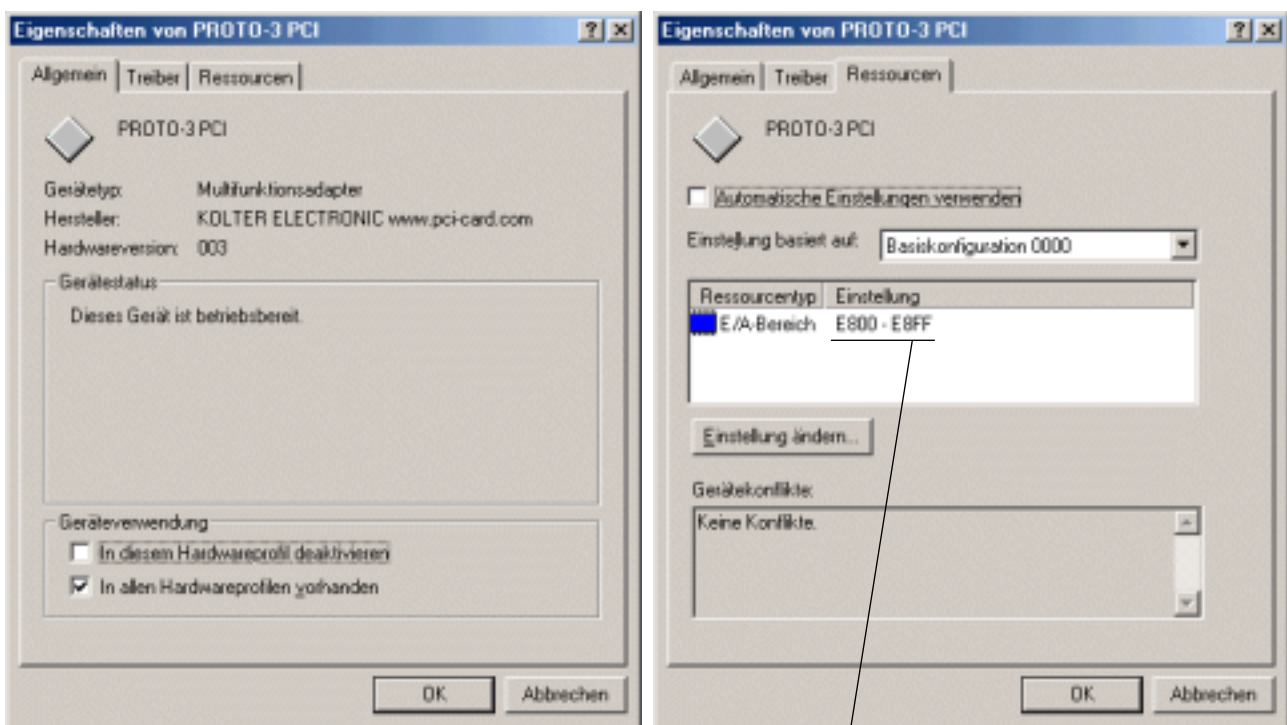


Nach dem Neustart können Sie überprüfen, ob die Treiber in das System integriert wurden. Über Start → Einstellungen → Systemsteuerung und einen Doppelklick im sich dann öffnenden Fenster auf das Symbol von „System“, werden Ihnen die „Eigenschaften von System“ angezeigt. Wählen Sie hier die Registerkarte „Geräte-Manager“ aus. Wenn Sie im dort dargestellten Verzeichnisbaum das Verzeichnis „Multifunktionskarten“ öffnen (Doppelklick auf das Icon oder anklicken des „+“ Symbols im Verzweigungsbaum), muss nun der Treiber (hier z. B. PROTO-3 PCI Karte) angezeigt werden.





Nach einem Doppelklick auf den Treibereintrag (alternativ geht das auch über Markieren des Treibers mit einem Mausklick und Betätigen der Schaltfläche „Eigenschaften“), lassen sich weitere Informationen zur neu installierten Karte gewinnen. Wichtig ist an dieser Stelle, dass man hier auf der Registerkarte „Ressourcen“ ablesen kann in welchem Adressbereich die PCI-Karte im System angemeldet ist. Diese Basis-Adresse (meistens im Bereich xx00 bis xxFF) zeigt dem Benutzer den I/O-Adressbereich, unter der die jeweilige Karte direkt angesprochen wird. Werden mehrere, gleiche Karten in einem System verwendet, ordnet der Geräte-Manager (über die zuvor in das PCI-BIOS eingelesenen Informationen) die Karten in ihrer Adresslage so an, dass kein Adressierungskonflikt entstehen kann.



Die I/O-Adresse ist dynamisch!

Im PCI-BIOS werden zwei gleiche Karten nur durch die Device-Nummer und die Adresslage unterschieden. Vendor-ID und Produkt-ID bleiben weiterhin identisch und können/sollen nicht verändert werden, da sonst eine klare Zuweisung im System nicht mehr gewährleistet ist.

Wichtiger Hinweis:

Da sich bei der Einbindung der PCI-Kartentreiber immer, entsprechend den bereits anderweitig genutzten Ressourcen, eine andere (dynamische) Adressbereichsvergabe ergibt, wäre es lästig und aufwendig, wenn dafür jedes mal der Programmcode auf diesen neuen Adressbereich angepasst werden müsste (zum Beispiel wenn man Software auf dem einen Rechner entwickelt, die dann auf einem anderen Computer laufen soll). Um dieses Problem elegant und komfortabel zu lösen, finden Sie auf der KOLTER CD im Verzeichnis „PCI\Utils_PCI“ einen Source-Code in C, der den gültigen Adressbereich ausliest, damit man ihn dann (beispielsweise über eine Parameter-Zuweisung) in den folgenden Programmcode einbauen kann. Dieser Source (Freeware) ist frei verfügbar und kann von Ihnen direkt in Ihre Programme übernommen werden.

Bitte lesen Sie dazu auch die EULA.TXT auf unserer CD im Verzeichnis ../txt.



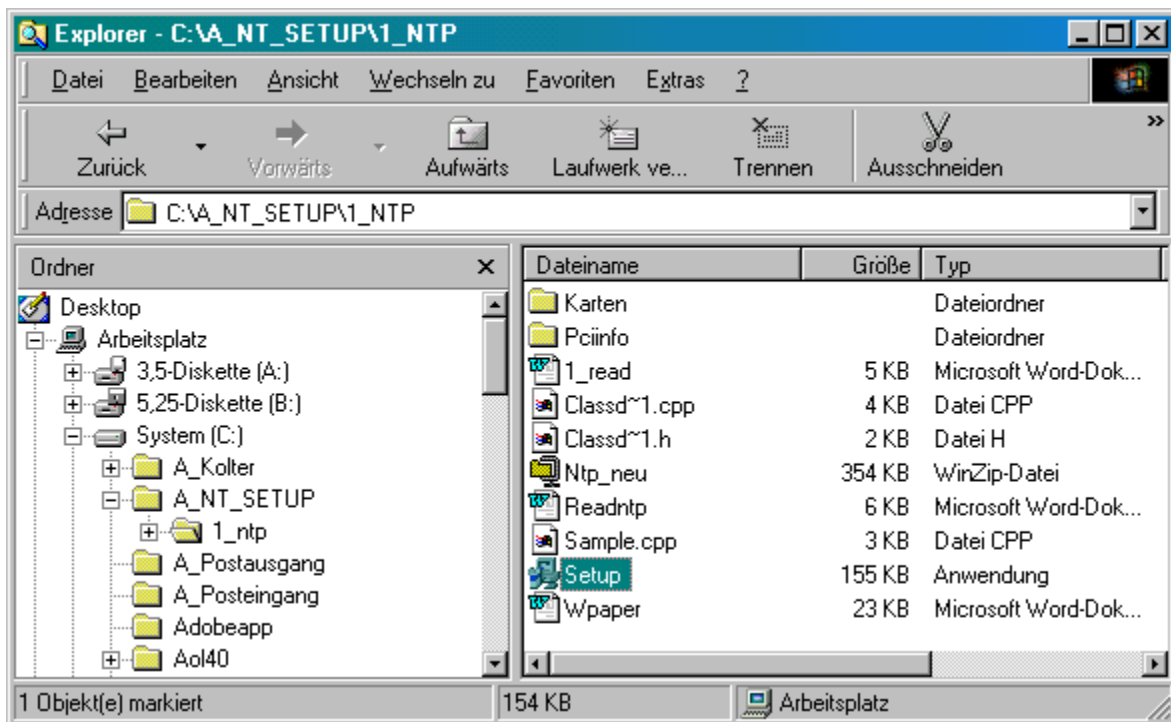
PCI und Windows NT

Die Windows - NT Treiber Installation

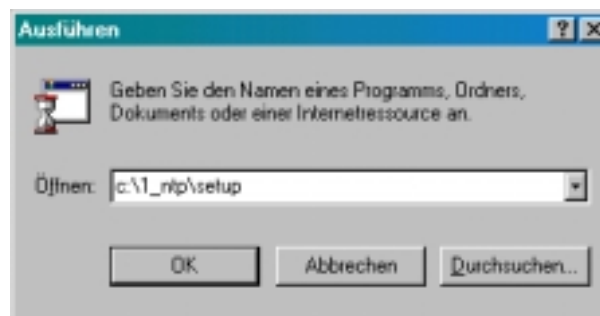
Ausgangspunkt ist wieder: Die Karte ist eingebaut, der Rechner ist eingeschaltet und der Datenträger mit den Treibern liegt bereit. Achten Sie bitte unbedingt darauf, dass Sie hierbei ausschließlich die beigelegten **NT**-Treiber benutzen. Sie sind speziell auf unsere Produkte zugeschnitten und getestet (unter NT 4 ab Servicepack 3).

Zuerst sollten Sie sich den Ordner NTP aus dem CD-Verzeichnis: Sourcen\WIN_NT4 in ein neues Verzeichnis (beispielsweise: 1_NTP) auf Ihre Festplatte kopieren.

Dann führen Sie SETUP aus, um die Installation zu starten. Dies kann auf zweierlei Art geschehen:



- 1.) Mit Hilfe des Explorers durch Öffnen des NTP-Ordners und einen Doppelklick auf „Setup“
- 2.) Über das Startmenü: Klicken Sie auf „Start“ dann auf „Ausführen...“, und geben Sie in die Eingabezeile des Fensters, das dann erscheint, manuell den gültigen Laufwerksbuchstaben und den kompletten Pfad gefolgt von SETUP ein (hier beispielsweise: C:\1_NTP\SETUP). Klicken Sie auf





„OK“, damit die Installation beginnen kann. Nach der Treiberinstallation muss der Rechner heruntergefahren und neu gestartet werden, damit sich die Treiber in das NT-System einbinden können.

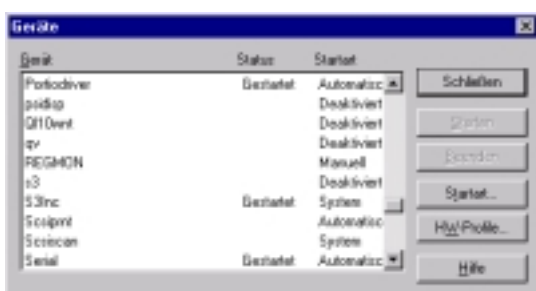
Am schnellsten geht das indem man „Neu starten“ im Beenden-Menü auswählt und „OK“ anklickt.

Nach diesem Neustart sollte kontrolliert werden, ob der Rechner beim Booten mit dem neuen KOLTER I/O-Port Treiber hochfährt. Dazu ruft man den Softwaremanager in Windows NT auf. Über Start → Einstellungen → Systemsteuerung wird ein Fenster geöffnet, in dem unter anderem ein Icon mit dem Namen „Software“ angezeigt wird. Ein Doppelklick auf dieses Symbol öffnet ein weiteres Fenster, das eine Liste mit den Programminstallationen unter Windows anzeigt. Hier muss, wenn die Installation der Treiber erfolgreich verlaufen ist, der KOLTER Portiodriver mit aufgelistet sein. Ist das der Fall, ist die Grundinstallation erfolgreich abgeschlossen.

Sollte dem nicht so sein, muss die Installation wie zuvor beschrieben wiederholt werden. Achten Sie bitte bei der zweiten Treiberinstallation auf eventuelle Fehlermeldungen.

Ermittlung der Adresslage einer I/O-Karte unter NT

Leider verfügt Windows-NT nicht über den gleichen Gerätemanager wie er aus Windows-9x bekannt ist, so dass man die I/O-Adresslage der Messkarte unter NT nicht unmittelbar nach dem Booten feststellen kann. Bei NT wurde wohl einfach vergessen beim Systemstart dafür zu sorgen, dass, wie bei WIN 9x/ME, kurz der gesamte Adressbereich gecheckt wird.



Wenn Sie sich über „Systemsteuerung“ und dort „Geräte“ die vorhandenen Einträge anschauen, sehen Sie, dass der Portdriver im System aktiviert wurde. Das ist eine Grundvoraussetzung dafür, dass sich später I/O-Ports überhaupt unter Windows-NT ansteuern lassen. Die Ausgabe sollte sein: „Portiodriver... Gestartet... Automatisch“. Drückt man nun auf den Button HW-Profile, sollte die Anzeige: „Aktiviert... Originalkonfiguration“ lauten.

Um jetzt die I/O-Adresslage der jeweiligen PCI-Karte zu erfahren (bei ISA-Karten ist das nicht nötig, denn die Adresslage wird dort in der Regel per Jumpereinstellung oder GAL fest zugewiesen), muss man sich eines weiteren Tools (zu finden im CD-Verzeichnis: PCI\Utils_PCI\Diag_HWT) bedienen, da Windows-NT über keine ausreichende Hardware-Diagnose verfügt. Weil die Windows-NT-Diagnose erst die Kartenadresse anzeigt, wenn schon einmal darauf zugegriffen wurde, ist sie also zur ersten Inbetriebnahme und zum Ermitteln der Adresslage völlig unbrauchbar, weil nur die aktivierten Devices angezeigt werden.



Ein weiteres PCI-Tool zur Kartenerkennung:

Als Shareware bieten wir mit HWT1.EXE (CD-Pfad: PCIUtils_PCI\Diag_HWT\HWT) ein Tool an, um die Adresslage der Hardware zu erkennen und um eine Zugriffsanalyse auf Port-Ebene durchzuführen. Man kann mit HWT1 prüfen, wie schnell der Rechner auf I/O-Ports zugreifen kann. Dazu werden in einer Zeitschleife 1 Millionen Port-Operationen durchgeführt und die dabei verstrichene Zeit gemessen. Gute Rechner werden dabei ca. 500.000 bis 2 Mio. Port-Zugriffe pro Sekunde selbst unter NT aufweisen können. Programmierer, die eine automatische Erkennung einer PCI- Karte unter Visual-BASIC programmieren möchten, können für einen Betrag von 49,- DM zzgl. MwSt. den Source von HWT1 käuflich erwerben.

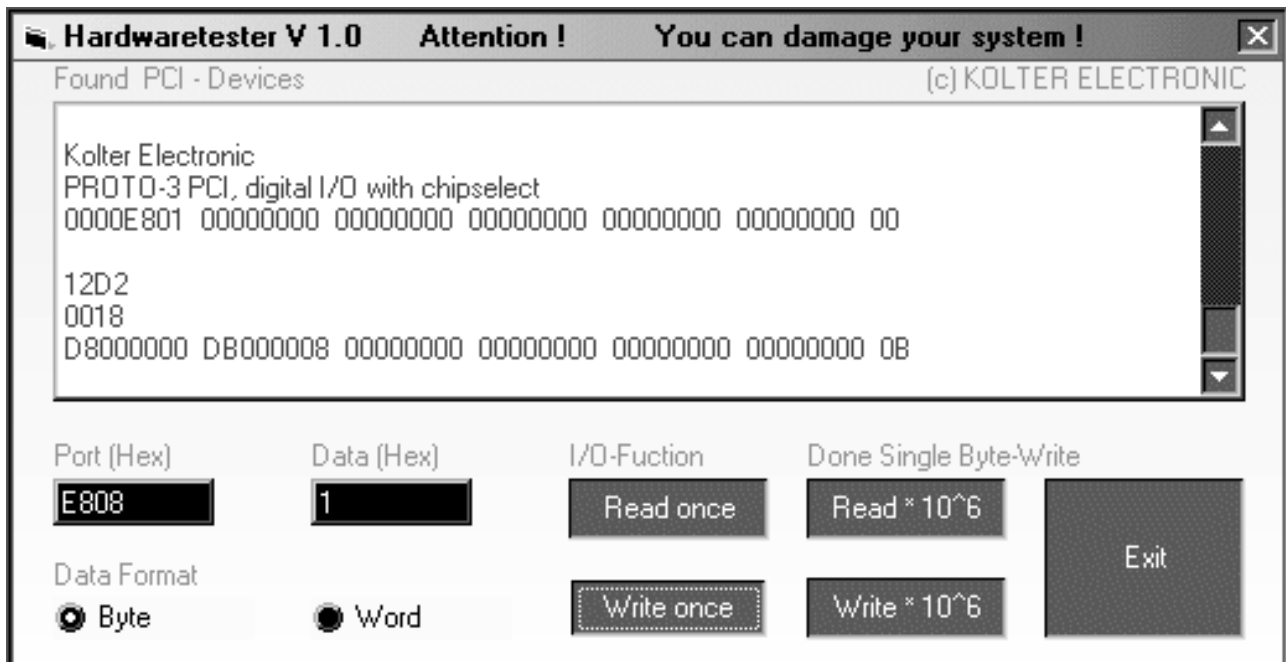


Foto: HWT1.EXE
unterstützt Windows 95/98/ME/NT4/Windows 2000 und XP.

Programm:
CD-Pfad :PCIUtils_PCI\Diag_HWT\HWT1a



Informationen zur Reaktionszeit innerhalb des NTP-Treibers

Für das NT-System gelten folgende Prioritätsklassen:

Quelle	Prioritätslevel	benutzt von
Hardware	HIGHEST	Bus Error
	POWER FAIL	Power fail Interrupt
	IPI	Interprocessor Interrupt
	CLOCK	Timer
	PROFIL	Profiling Timer
	DIRQL	Interrupts from Controllers: Devices!
Software	DISPATCH	Thread scheduler
	APC	APC execution
	PASSIVE	User threads
	- REALTIME	- User threads, real time priority (fix)
	- HIGH	- User threads, time slice (variable)
	- NORMAL	- ...
	- IDLE	- ...

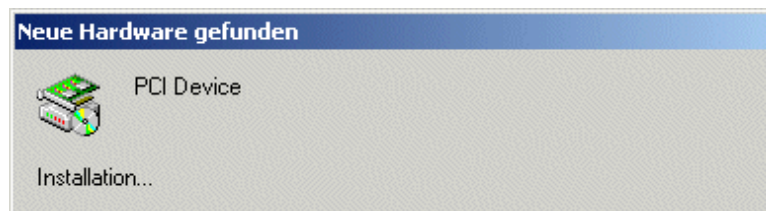
Achtung: REALTIME heißt nicht(!), dass ein Reaktionsintervall garantiert wird. Es benennt hier nur ein anderes Schedulingverfahren. Der REALTIME Thread verdrängt alles mit einer niedrigen Priorität, bis er selbst zum Stopp kommt. Interrupts, insbesondere Page fault, sind immer von höherer Priorität.



PCI und Windows 2000 beziehungsweise Windows XP

Allgemein

Die Abfolge einer Karteninstallation ist bei Windows 2000 und Windows XP nahezu identisch und gilt damit übergreifend für alle PC-Karten die mit diesen Betriebssystemen eingesetzt werden. Als Beispiel wurde hier eine PCI-1616 TTL-Karte unter Windows XP (Whistler) mit dem neuen 32-bit Device-Driver + INF installiert, die unseren PCI-Decoder mit der Vendor-ID 0x1001h verwendet. Um den Geräte-Treiber (verwendbar nur für Kolter-PC-Karten) zu installieren, gehen wir davon aus, dass Sie XP beziehungsweise Windows 2000 schon fertig installiert und bereits die neue PCI-Karte in einen freien PCI-Slot eingebaut haben. Nach dem Boot-Vorgang meldet Ihr System anschliessend, dass ein neues PCI-Device gefunden wurde und empfiehlt die Installation über den Hardware-Manager beziehungsweise Assistenten durchzuführen.



...doch halt !!! Beachten Sie die im Folgenden beschriebenen Installationsvoraussetzungen.

Voraussetzung zur Installation

- ❑ Stellen Sie zuerst sicher, dass keine gelben Ausrufezeichen im Gerätemanager angezeigt werden. Diese werden eingeblendet, wenn zuvor andere Komponenten fehlerhaft installiert worden sind.
- ❑ Installieren Sie immer eine Karte **nach** der anderen beziehungsweise stecken Sie nie zwei oder mehr Karten auf einmal in die PCI-Slots.
- ❑ Beachten Sie genau die Mitteilungen der Dialog-Boxen der Hardware- Installationsroutine und befolgen Sie deren Anweisungen genau.
- ❑ Achten Sie unbedingt darauf, dass Sie die Admin-Rechte haben beziehungsweise als Administrator angemeldet sind.
- ❑ Stecken Sie eine Karte nur im ausgeschalteten Zustand ein und warten Sie nach dem Booten (PC-Neustart) auf die Mitteilung, das Windows ein neues Gerät gefunden hat, um den Treiber jetzt zu installieren. Nie umgekehrt!

Fehler im System - was nun?

Falls Sie bereits eine fehlerhafte Treiberinstallation auf Ihrem 2000er/XP Windows-System haben, de-installieren Sie alle fehlerhaften Komponenten komplett über den Gerätemanager beziehungsweise entfernen Sie das Gerät mit einem Klick der rechten Maustaste auf dem Device-Eintrag. **Doch Achtung:** Löschen Sie nie eine Systemkomponente des Betriebssystems, da ansonsten eine komplette Neuinstallation von Windows notwendig werden kann. Falls Sie bereits Anwendersoftware zu dieser Hardware mitinstalliert haben, sollten Sie auch diese de-installieren, damit Treiber dieser Soft-



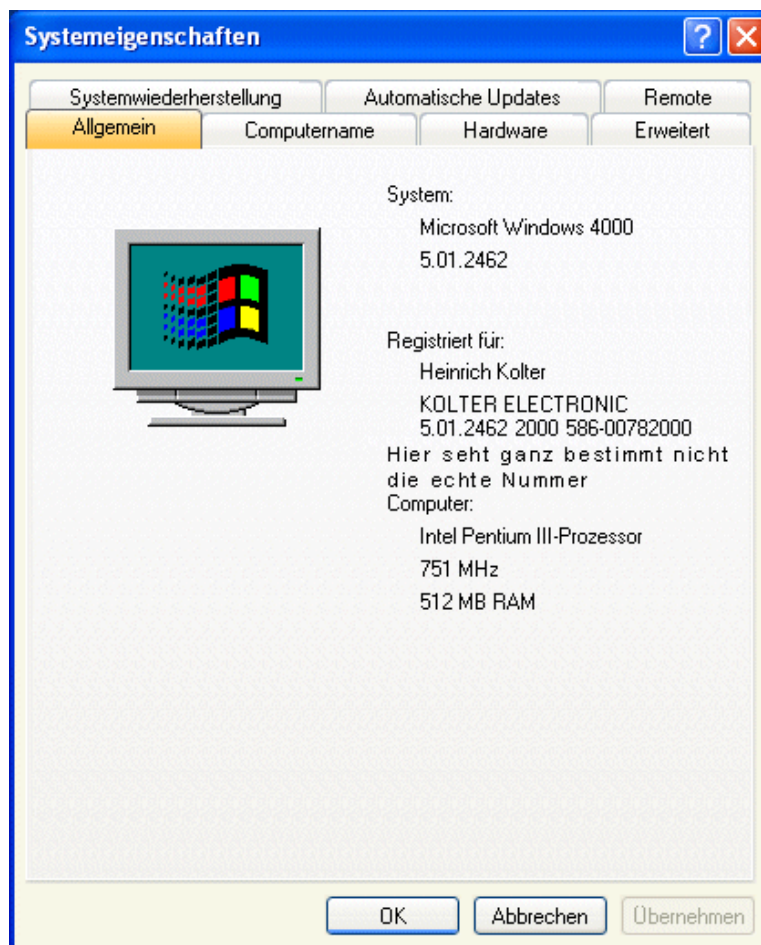
ware nicht während des Bootens auf Systemebene auf die fehlerhafte Hardwareinstallation zugreifen können. Löschen Sie die Backup OEMx.INF Script-Files aus Ihrem Windows\INF\...-Verzeichnis, damit Windows gezwungen wird, auch wirklich den neuen Karten-Treiber von Ihrem Datenträger zu installieren. Starten Sie den Rechner anschliessend ohne die PCI-Karte neu, bevor Sie den gesamten PnP-Prozess wiederholen und den Treiber installieren. Ein einfaches überspielen bei eingesteckter Hardware führt ansonsten nur zu weiteren, sonderbaren Konflikten im Windows-Gerätmanager und bringt überhaupt nichts ausser Zeitverschwendung, Katzenjammer oder unter Umständen ein völlig unbrauchbares System.

Und los gehts...

Die folgende Auswahl besteht darin, ob eine automatische oder manuelle Installation durch den Assistenten erfolgen soll. Bei der automatischen Installation brauchen Sie sich lediglich an die schrittweise Abfolge des HW-Assistenten zu halten und werden per Menü-Dialog durch die Installation geführt. Beide Wege führen letztendlich zum gleichen Ergebnis. Aus diesem Grund gehen wir nun auch nicht auf die automatische Installation durch den Hardwareassistenten näher ein und befassen uns vielmehr mit der manuellen Installation, da hierdurch die Treibereinbindung besser dargestellt werden kann.

Die manuelle Installation

Nach dem Starten des Hardware-Managers (mit rechter Maustaste über „Arbeitsplatz“), gelangt man zur Auswahl in das Fenster „Systemeigenschaften“.

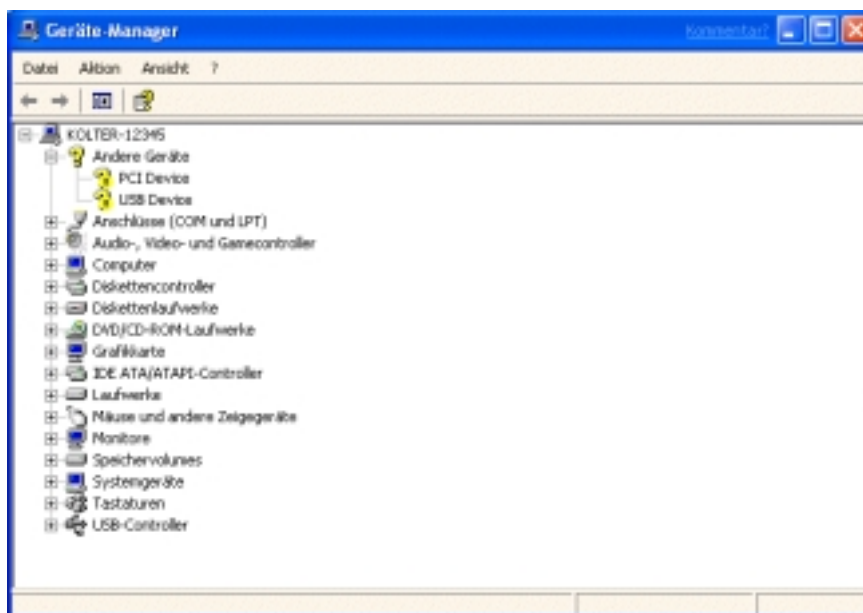




Hier wechselt man durch Anklicken der Registerlasche „Hardware“ auf die Bedienungsseite, auf der die verschiedenen Möglichkeiten zur Hardware-Installation beziehungsweise Konfiguration zur Verfügung gestellt werden.



Durch das Anklicken der Schaltfläche „Geräte-Manager“ wird das Verzeichnis des „Geräte-Managers“ angezeigt: hier kann man die Einträge der einzelnen Geräte einsehen beziehungsweise abändern.





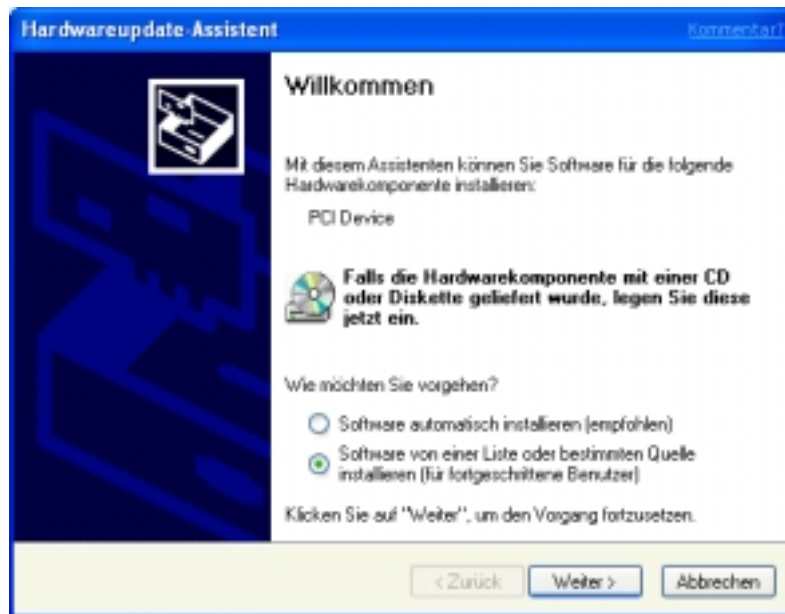
Alle Treiberinstallationen können Sie zu einem späteren Zeitpunkt auch wiederholen bzw. abändern, falls ein Fehler oder ein Irrtum während der Installation auftreten sollte. Dazu kann man das Device aus dem Gerätemanager komplett löschen und es gegebenenfalls, nach einem Rechner-Neustart, über den Assistenten wieder neu installieren.

Da grundsätzlich eine neue PCI-Karte dem System zunächst unbekannt ist und auch so gemeldet wird, wird ihr Eintrag Übergangsweise mit einem gelben Fragezeichen versehen und als unbekanntes Device unter „Andere Geräte“ deklariert. In unserem Beispiel wurde neben der unbekannten PCI-1616 Karte (erster Eintrag) gleich noch ein USB-Gerät (zweiter Eintrag) gemeldet, was für die weitere Installation unserer PCI-Karte jedoch keine Rolle spielt. Da im jetzigen Stadium der XP-Entwicklung USB noch nicht unterstützt wird (es soll laut Microsoft später aber in XP enthalten sein), wird zunächst dieser zweite Eintrag von Hand gelöscht.

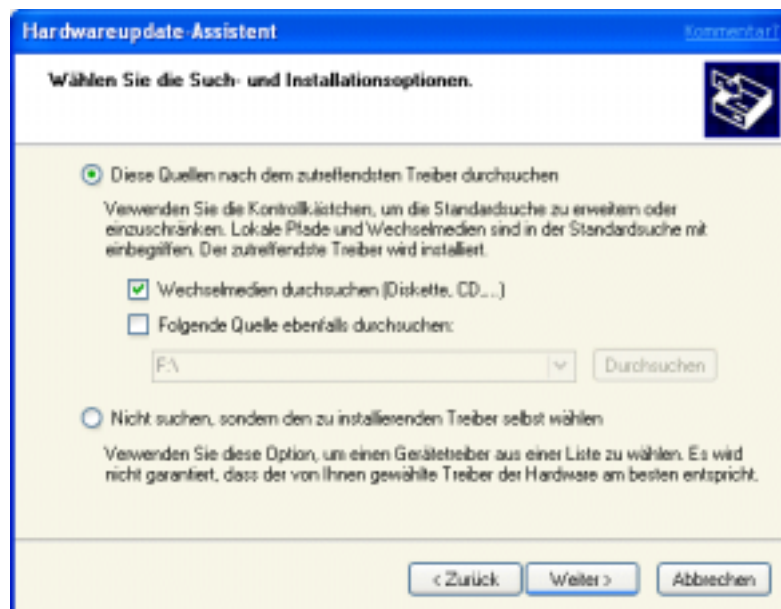
Um den Treiber für unsere PCI-1616 Karte zu installieren muss nun das gelbe Fragezeichen angeklickt werden (mit einem Doppelklick) um zum Einstellungsfenster „Eigenschaften von PCI-Device“ zu gelangen.



Durch Anklicken des Buttons: „Treiber erneut installieren“ wird jetzt der Hardwareassistent aufgerufen, der die Treiberinstallation einleitet.



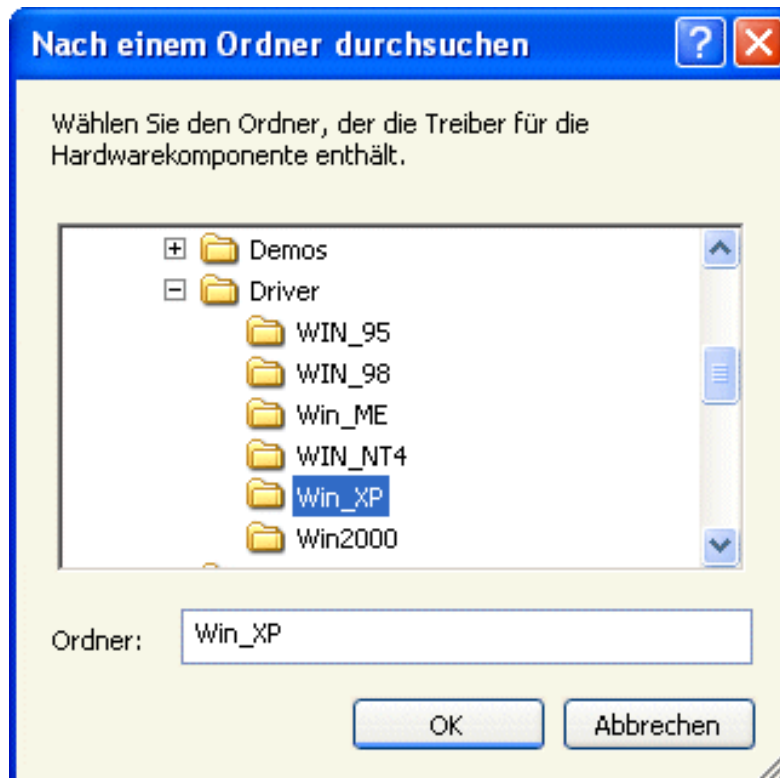
Mit der netten „Willkommen“-Begrüßung, wird gleich die Frage gestellt, ob man jetzt den Treiber automatisch oder manuell installieren möchte. Da die manuelle Installation mehr Eingriffsmöglichkeiten zulässt, haben wir uns für diese Variante entschieden. Durch die „Weiter“-Taste gelangt man zum nächsten Menüpunkt, der der Pfad-Einstellung zur Suche der Installationsdateien dient.



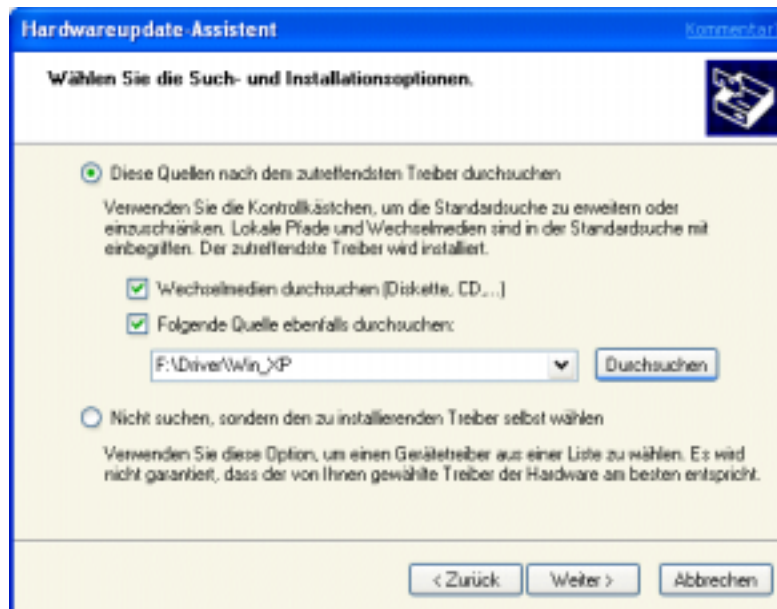
Ob CD, Diskette oder ausführlicher Ordner, spielt für die Installation keine Rolle. Wichtig ist lediglich, dass der richtige Treiber für XP gefunden wird. Dieser befindet sich für die jeweiligen Betriebssysteme auf unserer neuen CD (ab Rev.3.0) unter „Driver“, also in diesem Fall für XP unter

X:\Driver\Win2k_XP\kolter.inf

X = aktueller CD-Laufwerk-Buchstabe



Nach Wahl beziehungsweise Vorgabe des richtigen CD-Ordnern, in dem sich die Geräte-Treiber befinden, werden diese dann anschliessend automatisch, per Registry-Eintrag, in das Windows-System eingebunden.





Bei Erfolg wird dies im Fenster „Hardwareupdate Assistent“ unter „Fertigstellen des Assistenten“ sichtbar, in dem jetzt auch der reale Name des Gerätes angezeigt wird (In unserem Fall: PCI-1616).



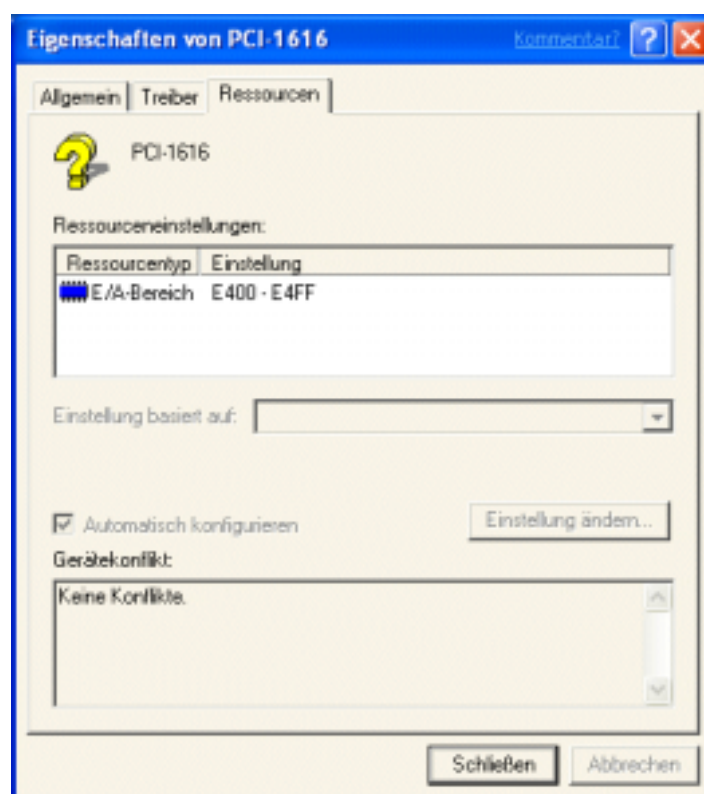
Durch Anklicken des Buttons: „Fertig stellen“ wird nun die Treiberinstallation verlassen, da die Treiber in das Systemverzeichnis kopiert wurden.

Damit nun überprüft werden kann, ob der Treiber korrekt übernommen wurde, können Sie jetzt nochmals auf das Device PCI-1616 klicken, um die Treiberdaten zu ermitteln.



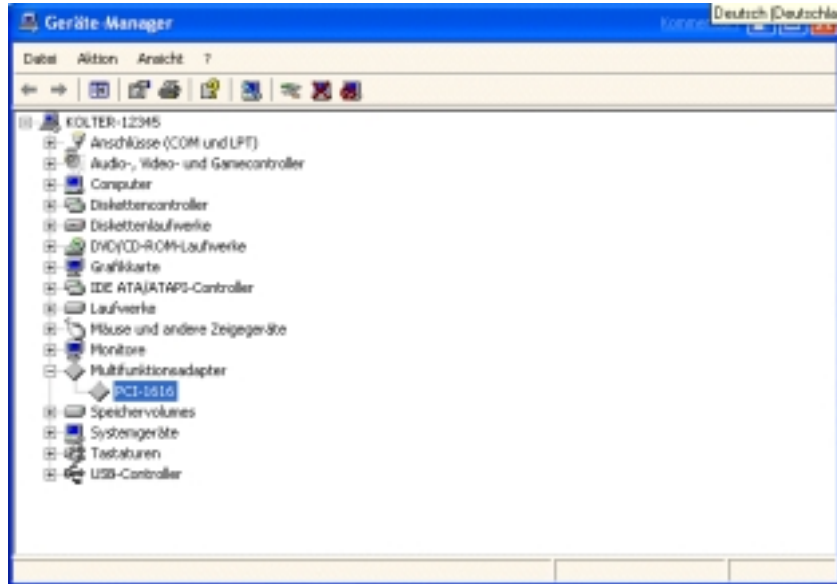


Beispielsweise können Sie im Ordner: „Ressourcen“ die I/O-Adresse der aufgerufenen Karte ermitteln und überprüfen, ob Konflikte mit der Hardware/Software entstanden sind.





Nach einem Neustart, ändert sich das gelbe Fragezeichen zum Multifunktionskartensymbol (graue Raute). Zusätzlich wird im entsprechenden Eintrag der Device-Namen (hier PCI-1616) ausgegeben.



Das war's – Geschäft!

Hardwaretest und PCI-Informationen

Möchte man testweise auf generelle I/O-Hardware direkt zugreifen (Achtung: nicht ganz ungefährlich), kann das Testprogramm HWT (CD-Pfad: PCI\Utils_PCIDiag_HWT) von grossem Nutzen sein.

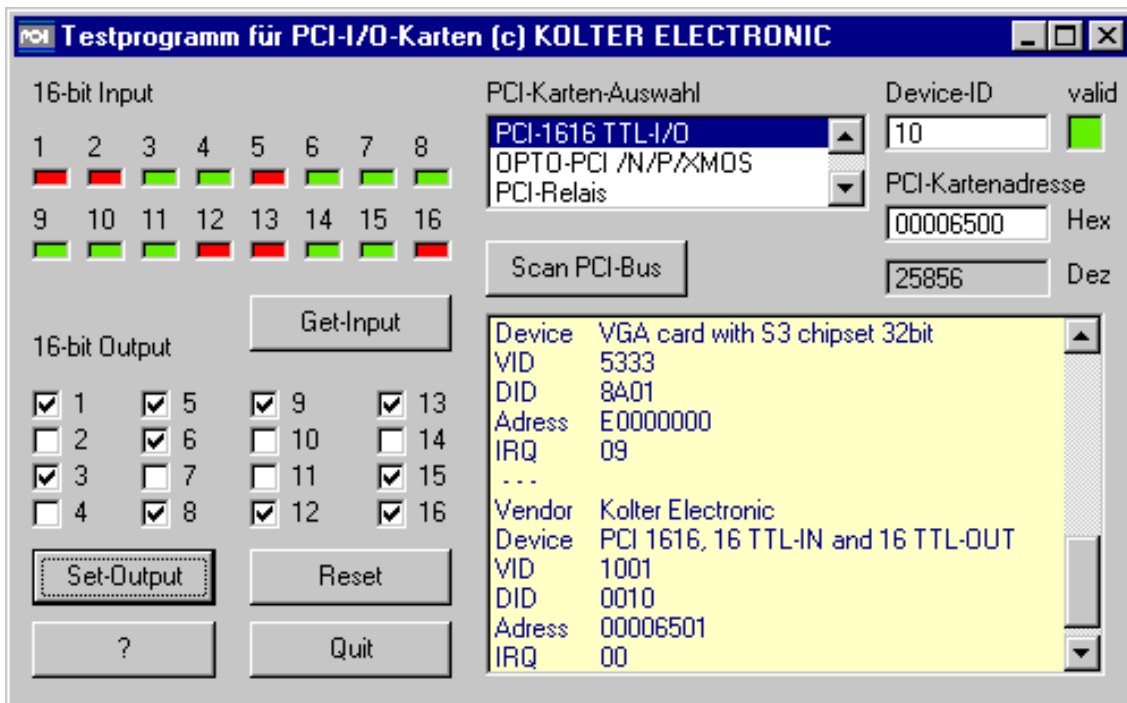




Mit HWT kann die I/O-Adresse der Karte über einen VXD/SYS Device-Driver ermittelt und über Read/Write-Befehle entsprechend angesteuert werden. Zusätzlich bietet HWT eine Geschwindigkeitsanalyse an, mit der man die maximale, systembedingte Ablaufgeschwindigkeit von Port-Zyklen austesten kann. In unserem Beispiel wird die PCI-1616 Karte an der Adresse 6500 erkannt (ein Offset von 1 zeigt an, dass es sich um eine reine I/O-Karte handelt). Über das Feld „Port(Hex)“ wird die Adresse manuell eingetragen, auf der die Daten (im rechten Feld) übermittelt werden. Je nach Wahl der Read-Once- oder Write-Once-Taste wird ungefähr 1 millionenmal ein Read-Port oder Write-Port 1x ausgeführt.

Mehr Komfort

Übersichtlicher und einfacher geht es natürlich mit einer richtigen Steuerungsanwendung wie zum Beispiel PCIIO.EXE (auf der CD im Verzeichnis Anwendungen\Win_NT_2K_XP\PCI_IO), die extra für eine beziehungsweise mehrere registrierte PCI-Karten programmiert wurde.



In unserem Beispiel wird die PCI-1616 Karte automatisch ermittelt („Scan PCI-Bus“) und mit jeweils 16 LED-Balkenanzeigen bzw. 16 Kontroll-Kästchen (für den Ausgang) übersichtlich dargestellt. Die jeweilige PCI-Karte wird im Feld: „PCI-Karten-Auswahl“ manuell vorgegeben. Wird die Karte nicht erkannt (zum Beispiel wenn eine andere PCI-Karte eingebaut wurde als angegeben), kann nicht auf das Device beziehungsweise die Ports zugegriffen werden und die Valid-Anzeige bleibt schwarz (invalid). Mit den Buttons: „Set-Output“ wird der aktuelle Stand der durch Anklicken gesetzten Kontroll-Kästchen je nach Kanal auf die Ausgänge übertragen. Die Darstellung der Eingänge (LEDs) wird über die Taste: „Get-Input“ aktualisiert.

Aktuelle Anwendungen wie PCIIO finden Sie auf unserer neuen CD (ab rev. 3.0) unter dem CD-Ordner: \Anwendungen\Win9x_ME oder Win_NT_2K_XP.

Diese kleine Test-Software unterstützt jeweils nur eine PCI-I/O-Karte im System. Dabei kann zwischen den Karten: PCI-1616, PCI-Relais, PCI-OptoRel und OPTO-PCI ausgewählt werden. Mit dieser Test-Software wird kein bestimmter, steuerungsrelevanter Zweck verfolgt; sie dient lediglich dazu, Eingänge und Ausgänge (jeweils 16) zu testen. Es wurde auf eine weitere Datenübergaben daher



verzichtet. Wenn Sie die Eigenschaften erweitern möchten und diese Software als Basis nutzen wollen, können Sie den MS-Visual-BASIC 6 Source von PCIIO.EXE käuflich erwerben. Falls Sie eigene Programme zu umfangreichen Steuerungen von PCI-I/O-Karten entwickeln möchten, empfehlen wir die Entwicklungsplattform ProfiLab-EXPERT 2.0

Kernel Drivers

Wie Kernel-Treiber mit PCI-Karten funktionieren

Installation und Systembindung

Einführung

Bei Windows XP handelt es sich um die Version des Windows-Betriebssystems von Microsoft, die als Nachfolger von Microsoft Windows 2000 und Microsoft Windows Me gilt und diese beiden Betriebssysteme ersetzt. Windows XP basiert auf der Windows 2000-Codebasis und verfügt über die Zuverlässigkeit und Leistung dieses Betriebssystems. Darüber hinaus besitzt Windows XP die neuen Features des Betriebssystems Windows Me in verbesserter Form: Systemwiederherstellung, Windows Media Player, Windows-Bilderfassung usw. Ob nun Windows 98, ME, NT, 2000 oder XP, es gilt grundsätzlich: Damit unter Windows entsprechende Hardwarekomponenten angesteuert werden können, müssen entsprechende Treiber zu jedem Gerät installiert werden, die für jedes Betriebssystem unterschiedlich ausfallen.

Der Gerätetreiber

Geräte (engl. *device*) sind Drucker, Graphikkarten, IDE-Controller oder aber auch moderne PCI-Messkarten wie A/D-Karten und Hardware mit TTL-Ports, die unter Windows installiert werden müssen. Die Plug-and-Play-Unterstützung ist Voraussetzung, damit Slot-Hardware (PCI) unter Windows erkannt, installiert und verwaltet werden kann. Diese Geräte können samt Treiber über den Geräte-Manager eingesehen (Parametrierungen), deinstalliert oder mit einem neueren UpDate versehen werden. Teilweise werden Treiber zu gängigen Geräten bereits bei der Betriebssysteminstallation von der Windows-CD gleich vorinstalliert, bei Ware die nicht als Massenprodukt vertrieben wird, muss ein hardwarespezifischer Treiber vom Hersteller nachinstalliert werden. Ein Treiber entscheidet wesentlich darüber mit, wie gut oder schlecht sich ein Gerät im Wirk-Betrieb verhält. Wird beispielsweise ein 600-dpi Drucker nur mit einem 300-dpi Treiber installiert, kann folglich auch der Ausdruck nur mit 300-dpi erfolgen obwohl das Gerät 600-dpi fähig ist.

Fehlerbehebung

Seit Windows 98 soll ein spezieller Mechanismus allgemeine Treiberprobleme verhindern helfen: Der Assistent für die Hardware-Fehlerbehebung (auch „Automatic Skip Driver Agent“ oder „ASD“ genannt) überwacht die Treibererkennung beim Hochfahren des Rechners. Erkennt ASD eine Inkompatibilität, wird der betreffende Treiber automatisch aus der Boot-Liste entfernt. Im Geräte-Manager findet der Anwender dann dazu ein gelbes Ausrufezeichen (siehe Bild) neben seinem Hardware-Symbol, das dann unter Umständen noch als „Unknown



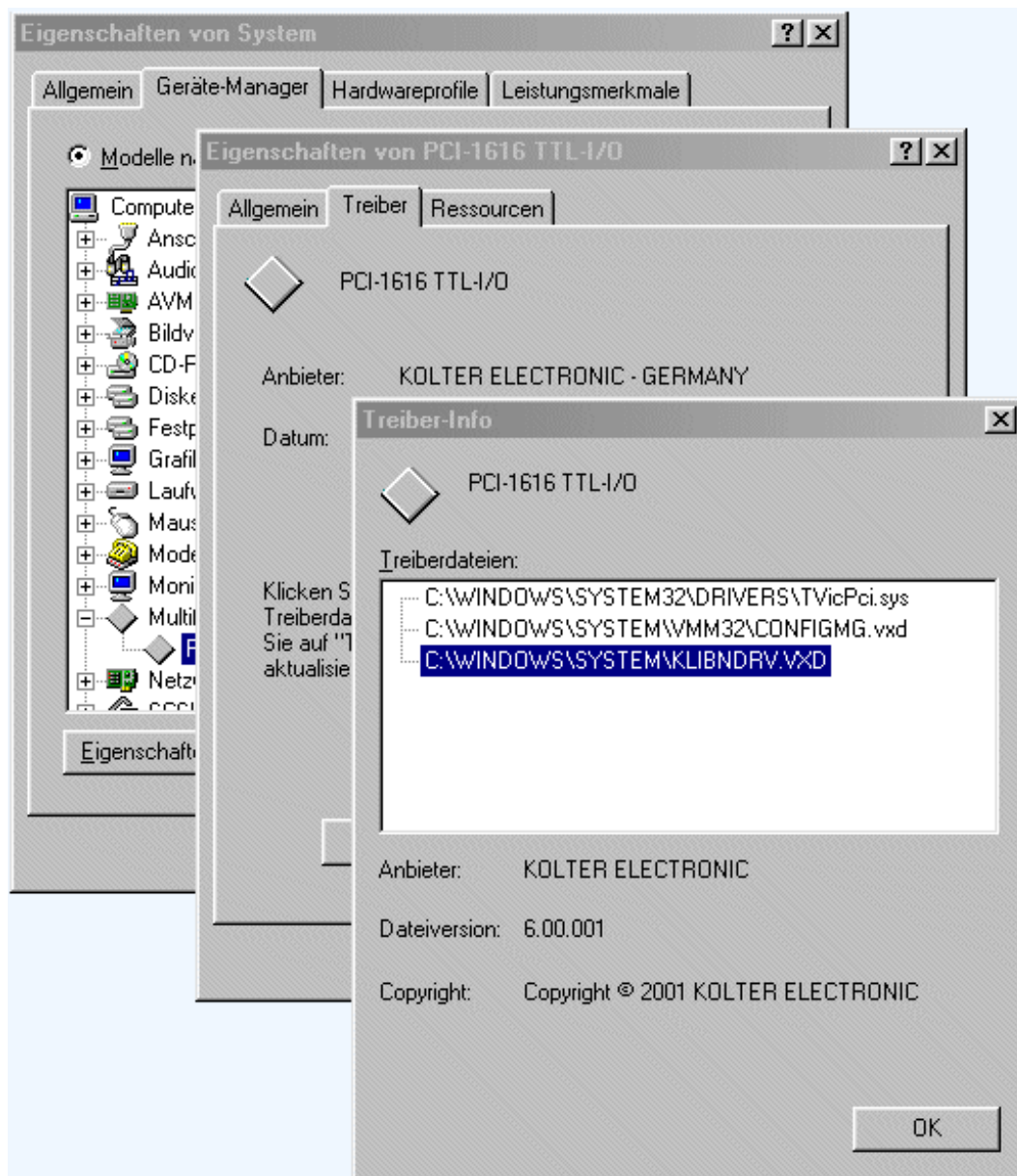
Fehler in der Treiberinstallation



device“ oder „PCI Device“ o.a. deklariert wird. Weiterhin werden fehlerhaft installierte beziehungsweise nicht erkannte Geräte in die Datei „detlog.txt“ aufgenommen, die man im Hauptverzeichnis findet. Kein Grund zur Panik: Die fehlerhaften oder nicht erkannten Karten und/oder Geräte können Sie später nach der Windows-Installation von Hand über die Systemsteuerung installieren und initialisieren.

Treiberarten

Grundvoraussetzung für eine erfolgreiche Treiberinstallation ist ein sauber installiertes Betriebssystem sowie die richtigen Treiber zur entsprechenden PnP-Hardware (PnP = Plug-and-Play) zu installieren (siehe dazu den Abschnitt zur Installation). Allgemein werden heutzutage Treiber (Kernel Device Driver) zur Einbindung von Hardware verwendet, damit sie beispielsweise unter Windows komfortabel angesprochen werden können. Dabei unterscheiden sich diese Treiber aber in mehrfacher Weise: Treiber, die unter Windows 9x/ME oder Windows NT/2000/XP verwendet werden (siehe: Anmerkung zum Kernel), damit sie vom Betriebssystem erkannt und für die Anwendung integriert





werden - und Treiber, die man benötigt, um auf die Hardware direkt zuzugreifen (Beispiel: klibdrv.sys). Im Klartext: In erster Linie sind die System-Treiber VXD und SYS gemeint, die eine Hardware im Gerätemanager eintragen und weitere Treiber, die Zugriffe mittels einer DLL von Anwendungen auf verschiedene Hardware ermöglichen soll. Beides kann ein und der selbe Treiber sein, muss es aber nicht. Eine weitere Variante ist, dass man direkt auf die Hardware ohne Treiber zugreift. Dies ist jedoch nur mit kleinen Hilfsmitteln (Port-Zugriff mittels asm-include) und Delphi oder VC++ unter Windows 9x/ME möglich - allerdings nicht aber unter NT/2000 oder XP. Da es hinsichtlich einer besseren UpDate-Fähigkeit hilfreicher ist, immer mit mehreren Treibern zu arbeiten, werden oft zwei oder mehr Treiber zu einer Hardware verwendet (beispielsweise: klibndrv.sys, klibdrv.sys pciinfo.sys...).

Anmerkung zum Kernel

Windows 2000 und XP basieren nicht auf dem Kernel von Windows 9x/ME, sondern auf dem geschützten Kernel von Windows NT, der zwar stabiler ist, aber ursprünglich nicht für USB, Multimedia-Anwendungen und Spiele konzipiert wurde. Diese waren unter dem Vorgänger Windows NT 4.0 nur eingeschränkt möglich, da alle neueren DirectX-Versionen nicht in die Windows NT-Plattform integriert werden konnten. Daher werden zu beiden Plattformen auch immer unterschiedliche Treiber zur betreffenden Hardware vom Hersteller angeboten. Beispielsweise können NT-basierende Plattformen nicht mit VXD-Treibern betrieben werden - sie benötigen echte 32-bit SYS Kernel-Treiber. Hierzu wird von Microsoft das neue Windows Driver Model (WDM-Treibermodell) empfohlen, das sich weitgehend durchgesetzt hat. WDM ist ein 32-bit Treibermodell, das sowohl von Win9x/ME als auch von Windows NT unterstützt wird. Microsoft hat dazu bestimmte NT-Kernel-Dienste als virtuelle Gerätetreiber (ntkern.vxd) zur Verfügung gestellt. Damit ist es möglich, die neuen WDM-Treiber parallel zu den alten Treibern zu nutzen. Einen Vorteil hiervon versprechen sich NT-Anwender, die für Ihre nicht unterstützte Hardware nun auf einen WDM-Treiber hoffen können.

Der INF-Script

Damit Treiber überhaupt in die Systemumgebung aufgenommen werden können, bedient man sich diverser Hilfsmittel wie Setups oder INF-Dateien. Alle Kernel- und WDM-Treiber werden mittels der INF-Dateien installiert. Sie veranlasst das Kopieren der Dateien, den Eintrag in der Registry sowie die Zuordnung der vom Treiber unterstützten Hardware. Es handelt sich um ein Microsoft-standardisiertes Installationskript im ASCII Format und kann mit jedem herkömmlichen Editor wie Word, Wordpad, Edit oder Notepad editiert werden. Eine INF-Datei besteht aus mehreren Sektionen und beginnt üblicherweise mit der „Signature“, die dem nachfolgenden Scripttext eine oder mehrere BS-Plattformmöglichkeiten zuteilt. Jede Sektion wird durch ein Schlüsselwort in eckigen Klammern eingeleitet. Dabei kann es sich um systemspezifische oder benutzerdefinierte Schlüsselworte handeln. Eine Auflistung der systemspezifischen Werte findet man in der Dokumentation zum DDK. Jede Sektion besitzt mindestens einen Eintrag. Die Art des Eintrags ist von der Bedeutung der Sektion abhängig. Welche Sektionen und Einträge (Beispiel: Geräteklasse) notwendig und welche optional sind, hängt immer von der Art des Treibers ab. Jede Geräteklasse ist in einem eigenen Abschnitt beschrieben, der im allgemeinen aus einem Header mit der Identifikation der INF-Datei und der unterstützten Geräteklasse besteht, gefolgt vom Manufacturer-Abschnitt. Darin sind Hersteller und die von ihnen produzierten Geräte aufgelistet. Im Install-Abschnitt sind der Gerätetreiber und die physikalischen Eigenschaften der Geräte erklärt. Zusätzlich laufen hier alle Installationshinweise und Pfade zusammen. Die Reihenfolge des Auftretens der Sektionen ist beliebig, da unter anderem auch Setupprogramme den Namen der Sektion mit verwenden können.

Eine reine Setup-Installation ist für den Benutzer zwar einfach zu handhaben, da sie als EXE lediglich gestartet werden muss. Leider ist sie für den Hersteller jedoch schwieriger und lästiger zu programmieren, so dass diese Variante nicht immer up-to-date zu aktuellen Betriebssystemen bzw. zur Hardware ist. Ein INF-Script ist wesentlich pflegeleichter und kann zudem vom Nutzer eingesehen



oder selber abgeändert werden, um die Installation besser auf sein System auszurichten. Damit die Treiber an ihren richtigen Ort gelangen (bei NT/2000/XP: C:\WINNT\system32\drivers, und bei W9x/ME: C:\WINDOWS\SYSTEM32\DRIVERS) sucht Windows immer nach einer aktuellen INF-Datei, in der Hersteller (VID) und Devicecode (DID) zur erkannten Hardware passen. Die INF-Datei wird dazu mit der rechten Maustaste angeklickt. Dann öffnet sich ein weiteres Fenster, mit der Möglichkeit den/die Treiber über „Installieren“ in das System zu übertragen. In der Regel werden dazu verschiedene Dateien mit der Endung: VXD, SYS und DLL in das entsprechende System-Verzeichnis kopiert und dauerhafte Einträge in die Windows-Registry vorgenommen. Nach einem Neustart werden diese Treiberdienste automatisch gestartet und die KOLTER-Hardware im Gerätemanager als „Multifunction Device“ aufgeführt.

Hier ein Script-Ausschnitt für Windows 95/98:

```
[Version]
Signature=$Windows 95/98$"
Provider=%String0%
HardwareClass=MultiFunction
ClassGUID={4d36e971-e325-11ce-bfc1-08002be10318}
Class=Multifunction
HWRevision=003
DriverVer=09/14/2001, 2.0.0.0
Driver=Multifunktionskarten
Version=2.0.0.0
```

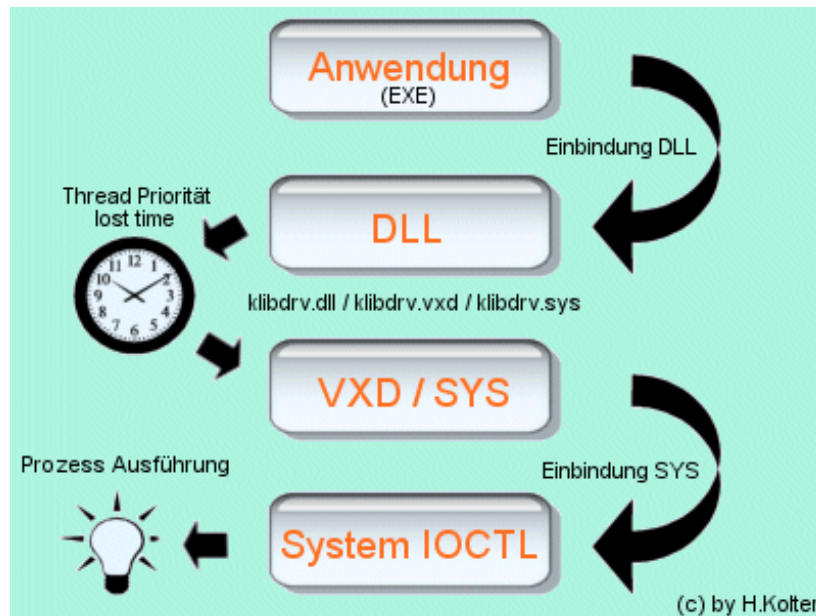
Anbindung zum System

Dynamische Bibliotheken (DLL) sind im Allgemeinen weder direkt ausführbar noch verarbeiten sie Benutzereingaben im Rahmen einer Nachrichtenschleife. Vielmehr handelt es sich bei DLLs um eigenständige Dateien, die anderen Programmen und dynamischen Bibliotheken eine Sammlung von Funktionen bereitstellen, die bestimmte Aufgaben erledigen. Eine dynamische Bibliothek tritt nur dann in Aktion, wenn ein anderes Modul eine ihrer Funktionen aufruft. Statische Bibliotheken werden hauptsächlich bei der Entwicklung von Programmen verwendet – im Gegensatz zu den dynamischen Bibliotheken, die zur Laufzeit des jeweiligen Programms verfügbar sein müssen. Windows sucht diese DLL automatisch zuerst im Anwenderverzeichnis der EXE-Datei. Fehlt sie dort, sucht es danach im Windows-Systemverzeichnis. Es gibt verschiedene Methoden, wie eine DLL angesprochen werden kann. So kann beispielsweise eine 16-bit-DLL unter Windows 9x/ME ohne SYS- oder VXD-Treiber betrieben werden, die jedoch unter NT-Systemen zum Absturz führt. Im umgekehrten Fall können 32-bit-DLLs, die für NT-Systeme mit SYS-Treiber geschrieben worden sind, nicht unter Windows 9x/ME ohne VXD-Treiber betrieben werden. Deshalb ist es elementar wichtig, dass bei der Installation von Treibern wirklich das richtige Treiber-Verzeichnis angewählt wird, da sonst das Gerät entweder nicht richtig funktioniert oder sogar das gesamte System durcheinander kommt.

Nicht zuletzt bestimmt auch der Registry-Eintrag wesentlich mit, wie Treiber in das System eingebunden werden. So kann beispielsweise ein Eintrag darüber entscheiden, ob ein Geräte-Dienst permanent oder erst auf Anforderung der Anwendung zur Verfügung gestellt wird. Liegt der Treiber permanent an, bremst er das System unnötig aus, ist dafür jedoch bei einem Zugriff schneller bereit und abgearbeitet. Im umgekehrten Fall wird das System nicht unnötig belastet, jedoch müssen bei einem Zugriff diverse Initialisierungen, Handles und Prioritäten durchlaufen werden, bis die Funktion ausgeführt werden kann.



Um mit einer selbstgeschriebenen Software, sei es über Visual-BASIC, VC++ oder Delphi, auf Hardware beziehungsweise I/O-Ports direkt zugreifen zu können, bedienen wir uns der klibdrv.dll, die mit ihren hardwarespezifischen Funktionen neue Befehle zum bestehenden Befehlsvorrat ergänzt. Sie wird üblicherweise als UNIT.PAS, MODUL.BAS oder INCLUDE.H einfach zum eigenen Source-Code dazugebunden. Da die DLL-Anbindung unterschiedlich ausfällt und in Ihrem Compiler-Handbuch ausführlich beschrieben sein sollte, gehen wir an dieser Stelle auch nicht näher darauf ein. In unseren kostenpflichtigen VB-Sourcen für NT können Sie die Anbindung der DLL und der dazu notwendigen Module leicht nachverfolgen. Nach einer Initialisierung durch den CREATE- oder OPEN-Befehl kann über einen zurückgegebenen HANDLE die DLL aus Ihrer Anwendung heraus angesprochen werden und die I/O-Anforderung über die SYS bzw. VXD an das System abgesetzt werden. Der dabei entstandene Zeitverlust im DLL -> SYS -> SYSTEM-Thread ist von der vorgegebenen Priorität abhängig. Die Prioritätenzuweisungen werden unter anderem durch Hintergrundanwendungen und System-Treiber stark belastet (beispielsweise: VGA-Karte, IDE-Treiber, Tastatur und Maus...). Selbst als Ring-0 ausgeführte Threads unter Windows 9x können Wartezeiten in der Zeitscheibe (Task) von ca. 10-100 µs und unter NT-Systemen von bis zu 200 ms entstehen lassen.



Übersicht zur Treibereinbindung unter Windows

Dynamische und statische Treiber

Je nach Treiberart können sowohl dynamische als auch statische Treiber für den Port-Zugriff auf Hardware verwendet werden. Bei der Verwendung von dynamischen Treibern, wird der DLL-Zugriff über den im gleichen Verzeichnis liegenden VXD-Treiber abgearbeitet (Win9x/ME), wo hingegen beim statischen Treiber der Zugriff auf die im System verankerten SYS-Treiber erfolgt (Win NT). Viele dynamische Treiber haben jedoch den Nachteil, dass sie bei jedem Zugriff geöffnet und geschlossen werden müssen (unnötiger Zeitverlust), wo hingegen statische Treiber erst nach Beendigung oder Abbruch der Anwendung wieder geschlossen wird. Da Windows NT aber ein anderes, sichereres Zeitscheibenverfahren verwendet als Win9x/ME sind dynamische Treiber oft wesentlich schneller als statische Treiber unter Windows NT. Fehlerhafte Zugriffe mit dynamischen Treibern unter Win9x/ME werden aus diesem Grund oftmals mit einer Schutzverletzung und einem Blue-Screen quittiert, wo Windows NT lediglich die Anwendung einstellt oder nur eine Schutzverletzung mit Abbruchmöglichkeit ausgibt - das System aber weiterhin funktionstüchtig bleibt. Über den Taskmanager kann ein fehlerhafter Prozess (Anwendung) auch jederzeit manuell abgebrochen werden.



Treiber von CD

Updates von Windows enthalten neuere Geräteinformationen zu Massenprodukte (VGA-Karte, SCSI-Contoller...), für die dann keine Diskette bzw. CD mehr erforderlich ist. Für Nieschenprodukte gelten immer die hinzugefügten Treiber auf der Hersteller-CD bzw. Diskette. Sie finden unsere aktuellen Treiber im Verzeichnis: %root CD%:\Drivers\...

Erläuterung der Abkürzungen

API

“application programming interface“

BS

“Betriebssystem“

NT

“New Technologie“. Windows NT ist ein 32-bit Betriebssystem von Microsoft.

PnP

Plug and Play besagt, dass der PC ohne Zutun des Benutzers den Einsteckkarten und Hardware-Komponenten alle Ressourcen automatisch zuordnet. Das Rechner-BIOS findet nach dem Einschalten des Computers alle PCI-Devices und fragt jedes einzelne Device nach den benötigten Ressourcen ab. Die zu verteilenden Ressourcen bestehen meistens aus I/O-Adressen, IRQ-Nummer, DMA-Kanälen und verwendeter Speicherbereiche. Die verwendeten Ressourcen jeder PCI-Karte werden anschliessend in einer ESCD-Datenbank des BIOS (im Memory) abgelegt und dem verwendeten Betriebssystem zur Verfügung gestellt.

PCI

Der PCI-Bus (Peripheral Component Interconnect Bus), ist ein aktuelles High-End-Bussystem mit 32/64-bit Datenbreite. Die Steckkarten arbeiten mit einem eigenen Systemtakt, asynchron zum PC. Über eine Hostbridge sind die Bussysteme miteinander verbunden und tauschen Daten mit 33/66 MHz aus.

Port I/O

Port I/O ist die Abkürzung für einen hardwarebezogenen „Input/Output-Anschluss“. Darunter versteht man heutzutage nicht mehr nur die Ein- und Ausgabe über Register, sondern auch über diverse Peripherie-Controller wie beispielsweise dem UPD8255 PPI Baustein.

DDK

Das Kürzel DDK steht für „Driver Development Kit“ und ist ein spezielle Ansammlung von Microsoft-Sourcen und Includes um mit einem C-Line-Compiler einen SYS- oder VXD-Gerätetreiber per Build-Befehl zu generieren. Hierzu ist tiefgehendes Fachwissen über die System- und Kernelprogrammierung nötig, dass nur sehr mühsam katalogisiert werden kann, da es keine schlüssige Referenz gibt beziehungsweise einem stetigem Wandel unterworfen ist.

DLL

Die Abkürzung „DLL“ steht für „Dynamic Link Library“. DLLs enthalten Prozeduren und Routinen, auf die Anwendungen mit Funktionsaufrufen zugreifen können (ähnlich einer API).

SYS

Das Suffix .SYS „System“ tragen auch die Startdateien von Windows (beispielsweise IO.SYS), die im Root-Verzeichnis des Systems liegen. Eine SYS-Datei ermöglicht einen statischen oder dynamischen Systemaufruf und wird vorzugsweise für Windows NT, 2000 oder XP verwendet.

**VXD**

Die virtuellen Gerätetreiber unter Windows 95, 98 und ME haben das Suffix VXD. Das x steht für den Gerätetyp „Virtual x Device“. Das heißt, man bezeichnet virtuelle Gerätetreiber für Anzeigegeräte mit VDD (Virtual Display Device) und für Drucker mit VPD (Virtual Printer Device). Ein virtueller Gerätetreiber ist ein 32-Bit-Treiber, der Systemressourcen wie Hardwarekomponenten oftmals nur dynamisch verwaltet.

INF

Die INF-Dateien sind reine Geräte-informationsdateien. Sie stellen dem Windows-Setup wichtige Treiber-Informationen zur Einrichtung einer Hardware- und Netzwerkkomponente wie Konfigurationsparameter und Laufwerkspfade zur Verfügung. In der Regel liefern Hardwarehersteller INF-Dateien samt Treiber auf einer CD mit dem Gerät aus.

WDM

Dieses Kürzel steht für „Windows Driver Modell“. Es umschreibt die Vorgehensweise und Einbindung von neuen Treibern, wie sie nach Auffassung von Microsoft für neue Betriebssysteme angewendet werden sollte.

Einen interessanten Link zum Thema finden Sie hier:

<http://www.microsoft.com/germany/ms/msdnbiblio/artikel/2000comp.htm>