

## **Hilfeinhalt**

Programm-Einleitung

Programm-Features

Systemvoraussetzungen

Installation / Deinstallation

Programmdateien

Startparameter

Danksagungen

Überblick über die Programmanalysen

Reportfunktion

FAQ – Häufig gestellte Fragen

Hinweise zur Professional-Version

Über das Programm

Hilfe-Index

## Hilfeindex

A

B

C

Chipsatz-Details

D

Danksagungen

E

F

FAQ – Häufig gestellte Fragen

G

Geräte-Details

Geräte-Fähigkeiten

H

Hilfe-Inhalt

I

Installation / Deinstallation

IRQ-Routing Tabelle

J

K

L

M

N

O

P

[PCI-Details](#)

[PCI-Headerdetails](#)

[PCI-Register](#)

[Professional-Version](#)

[Programm-Dateien](#)

[Programm-Einleitung](#)

[Programm-Features](#)

Q

R

[Reportfunktion](#)

S

[Startparameter](#)

[System-Interna](#)

[Systemvoraussetzungen](#)

T

U

[Über das Programm](#)

[Überblick über die Programmanalysen](#)

V

W

X

Y

Z

## Über das Programm



Die Registerkarte *Über* zeigt das Firmenlogo des Programmentwicklers *Devid Espenschied* und *Registrierungs-Details* an. Letztere liegen nur dann vor, wenn die Professional-Version von *ChipInfo* erworben und damit zusammenhängend ein Keyfile geliefert wurde. Dateien der Professional-Version dürfen nicht an Dritte weitergegeben werden !

Unter dem Logo erscheint die *Internet-* und *EMail-*Adresse des Programmentwicklers. Ein Mausklick auf die Adressen öffnet jeweils das voreingestellte Programm für die Darstellung von Internetseiten bzw. den EMail-Versand.

Die Registrierungs-Details bestehen aus den folgenden Bestandteilen:

- Name (Name des Kunden)
- Anschrift (Straße mit Hausnummer, sowie Postleitzahl und Stadt)
- Versionstyp (*Demo* oder *Professional*)
- Registriernummer (eindeutige Kundennummer, die jedem Kunden individuell zugewiesen wird).

In der Demo-Version werden alle Kundendaten mit Ausnahme des Feldes *Versionstyp* durch *keine Angabe* (k.A.) ersetzt.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)



## **Programm-Einleitung**

*ChipInfo* ist ein Windows-basiertes Programm, welches umfangreiche Informationen zum verwendeten Mainboard-Chipsatz und PCI-Bus mitsamt seiner angeschlossenen Geräte anzeigt. Hierzu wird der PCI-Bus nach allen verfügbaren Geräten durchsucht, und durch verschiedene Fenster und Listen übersichtlich dargestellt.

Der Chipsatz als solcher wurde mit dem 286er Prozessor eingeführt, bis zu dessen Fertigstellung mehrere Dutzend Chips auf dem Mainboard gängige Aufgaben übernommen haben. Mit dem Neat-Chipsatz von Chips & Technologies stand erstmals eine Sammlung einiger weniger Chips zur Verfügung, die alle wichtigen Controller-Funktionen beinhalteten und per dokumentierter Register programmierbar waren. Anschließend folgten weitere Hersteller, die als Opti, Symphony, SiS und Via sehr schnell den Chipsatz-Markt veränderten. Mit Einführung des PCI-Busses führte Intel die Unterscheidung zwischen North- und South-Bridge ein, die als zwei separate Bausteine unterschiedliche Aufgaben wahrnehmen. Während die Northbridge hauptsächlich den Prozessor anbindet, stellt die Southbridge eine Sammlung unterschiedlichster Einzel-Kontroller dar, die jeweils teilweise als eigene PCI-Geräte am PCI-Bus arbeiten und erkennbar sind. Dazu gehören Kontroller für AGP, IDE, SATA, SMBus, USB (1.1 und 2.0) sowie AC97 für Audio und Modem. Intel verwendete ab der 8xx-er Chipsatzserie die Bezeichnungen Northbridge und Southbridge nicht mehr, und nennt diese Memory Controller Hub (MCH) sowie I/O Controller Hub (ICH).

*ChipInfo* zeigt die Chipsatz-Einstellungen ausschließlich basierend auf dem PCI-Registerbereich an, weitere Quellen wie I/O-Register oder sogenannte Memory-Mapped-Register sind zurzeit noch nicht implementiert. Sollten Chipsätze nicht unterstützt werden, liegen höchstwahrscheinlich keine Datenblätter vor, die von einigen Herstellern häufig als vertraulich eingestuft sind. Einen genauen Featureumfang können Sie der [Featureliste](#) entnehmen, die unterstützten Chipsätze sind [hier](#) zusammengefasst. *ChipInfo* ist lauffähig unter den Betriebssystemen Windows 9x/ME sowie Windows NT4/2000/XP/2003 in jeglichen Produktfamilien.

Da die Software als mehrsprachige Anwendung entwickelt wurde, hat die aktuell gewählte Programmiersprache Auswirkungen auf alle Fenster und Listen. Als Ausnahme gelten hier die Chipsatz-Details, deren Übersetzung beispielsweise von Englisch nach Deutsch nicht mehr sinngemäß wäre. Weiterhin wird der Begriff Register komplett vermieden und stattdessen der eigentliche Registernamen verwendet.

Durch die sehr umfassende Datenerhebung und –aufbereitung ist *ChipInfo* für jeden interessant, der genau wissen möchte, welcher Chipsatz und welche PCI-Geräte in seinem System arbeiten und über welche Merkmale diese verfügen. Für den ambitionierten Benutzer erschließen die angezeigten Interna neue Sichtweisen in Bezug auf den verwendeten Rechner.

Siehe auch:  
[Hilfe-Inhalt](#)  
[Hilfe-Index](#)

## Programm-Features

Nachfolgend eine Auflistung der Programmfeatures, welche in der mit dieser Hilfedatei ausgelieferten *ChipInfo*-Version zur Verfügung stehen. Weitere Informationen zu den Unterschieden zwischen Demo- und Professional-Version finden Sie [hier](#).

- Ermittlung detaillierter Chipsatz-Informationen und dazugehöriger Chipsatz-Zusatzgeräte ([Liste der unterstützten Chipsätze](#))
- Ermittlung aller vorhandenen PCI- und AGP-Geräte mit jeweiligen Details zum Gerät, dem gerätespezifischen Header und den verwendeten Registern
- Aufschlüsselung der Geräte-Fähigkeiten mit Details zum Power Management, AGP, der Slot-Identifizierung, Message Signaled Interrupts (MSI), PCI-X und Debug Port
- Anzeige der IRQ-Routing Tabelle mit Aufschlüsselung der PCI-Slots und dazugehörigen Interrupts
- System-Internia mit Zusammenfassung aller Rechnerdaten (Prozessor, System, Betriebssystem, physikalischer- sowie virtueller Speicher und Programm-Datenbanken)
- Reportmöglichkeit wahlweise interaktiv oder kommandozeilengesteuert
- 3 Varianten des kommandozeilengesteuerten Reports (Standard mit allen Details, nur Chipsatz-Details und nur System-Internia)
- deutsche und englische Programmsprache in einem Paket verfügbar (Betriebssystem-Sprache wird ermittelt und für *ChipInfo* eingestellt, manuelle Änderung per Startparameter möglich)
- Debug-Modus mit Speicherung von Debug-Daten in der Datei *debug.txt*
- verschiedene Startparameter für Batch-Modus, Sprachwahl, Treiber-Management und Debug-Modus
- Online-Hilfe und Handbuch im PDF-Format.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)



## **Systemvoraussetzungen**

*ChipInfo* benötigt relativ geringe Systemvoraussetzungen, um lauffähig zu sein.  
Nachfolgend eine Auflistung der Voraussetzungen:

- x86-kompatibles Mainboard mit PCI-Bus
- x86-kompatibler Prozessor mit CPUID-Unterstützung
- Windows-Betriebssystem in den Versionen 95, 98, ME, NT4, 2000, XP oder 2003
- Administrationsrechte unter den Windows-Versionen NT4, 2000, XP und 2003
- ca. 1.9 MByte freier Festplattenspeicher für Programmdateien und Dokumentation
- Bildschirmauflösung von mindestens 800x600, um alle Daten auf einem Bildschirm unterzubringen.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **Installation / Deinstallation**

### **Installation**

Eine Installation im Sinne eines Setup-Programmes ist nicht notwendig, da *ChipInfo* mit der Programmdatei *CHIPINFO.EXE* startbar ist. Ein Eintrag im Software-Menü der Systemsteuerung findet nicht statt.

Die zusätzliche Datei *CHIPINFO.KEY* ist eine zwingende Voraussetzung für die Professional-Version und enthält die Kundendaten. Für die Erkennung der installierten PCI-Geräte ist die Datenbankdatei *PCI.DAT* notwendig, innerhalb der Registerkarte *System-Interna* wird zusätzlich die Datenbankdatei *MAINBRD.DAT* benötigt, um das Mainboard zu ermitteln. Eventuelle weitere Dateien sind optional und dienen ausschließlich dem Zweck der Dokumentation bzw. Online-Hilfe.

Unter den Betriebssystemen Windows NT4, 2000, XP und 2003 existiert eine zusätzliche Möglichkeit, um den von *ChipInfo* für den Hardwarezugriff benötigten Treiber manuell zu installieren (Treiberdatei *HWACCESS.SYS* ist Bestandteil von *CHIPINFO.EXE*). Dies wird grundsätzlich beim Programmstart erledigt. Der Startparameter */INSTALL* führt diesen Vorgang manuell durch, ohne *ChipInfo* zu starten. Dieser Vorgang ist aber i.d.R. nur dann notwendig, wenn *ChipInfo* während des Startvorganges Probleme beim Installieren dieses Treibers haben sollte.

### **Deinstallation**

Da keine Installation mit einem Setup-Programm durchgeführt wurde, lässt sich *ChipInfo* einfach durch Löschen der Programmdateien deinstallieren.

Der mit dem ersten Programmstart installierte Hardwaretreiber *HWACCESS.SYS* kann man anhand des Startparameters */UNINSTALL* manuell deinstallieren. Grundsätzlich geht *ChipInfo* jedoch so vor, dass dieser Treiber beim Programmende automatisch deinstalliert und gelöscht wird.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## Programmdateien

Die folgende Liste beschreibt die Dateien, welche mit jeder Programmversion von *ChipInfo* ausgeliefert werden. Eine Ausnahme stellt das Keyfile dar, dass nur Bestandteil der Professional-Version ist.

*ChipInfo* ist grundsätzlich so konzipiert, dass zur Programmausführung ausschliesslich die Dateien *CHIPINFO.EXE* und *CHIPINFO.KEY* notwendig sind. Erstere Datei ist das Programm selbst, zweitere das Keyfile mit Kundendaten. Optional sind die Gerätedatenbanken im Sinne der Dateien *PCI.DAT* und *MAINBRD.DAT* sinnvoll, weil diese die Gerätenamen innerhalb des Programmes liefern. Zusätzliche Dateien dienen ausschließlich dem Zweck der Dokumentation und Programmhilfe.

Datei	zum Start notwendig	Beschreibung
CHIPINFO.EXE	ja	Programmdatei
CHIPINFO.KEY	ja	Keyfile mit Kundendaten
PCI.DAT	nein	Datenbankdatei für PCI-Gerätenamen
MAINBRD.DAT	nein	Datenbankdatei für Mainboard-Bezeichnungen
CHIPINFO_D.HLP	nein	deutsche Hilfedatei
CHIPINFO_E.HLP	nein	englische Hilfedatei
CHIPINFO_D.PDF	nein	deutsches Handbuch
CHIPINFO_E.PDF	nein	englisches Handbuch
HISTORIE.TXT	nein	deutsche Historiedatei
HISTORY.TXT	nein	englische Historiedatei
LIZENZ.PDF	nein	deutsche Lizenzvereinbarung
LICENSE.PDF	nein	englische Lizenzvereinbarung
ORDER.TXT	nein	deutsche/englische Bestell-Details

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **Startparameter**

*ChipInfo* lässt sich mit verschiedenen Startparametern aufrufen, die einerseits die Installation bzw. Deinstallation des *ChipInfo*-Treibers und andererseits die Spracheinstellung betreffen.

Zusätzlich lässt sich ein Report direkt von der Kommandozeile aus erstellen, bei dem die grafische *ChipInfo*-Instanz nicht gestartet wird. Diese Funktion ist ideal für den Einsatz in Netzwerken gedacht. Nachfolgend eine Auflistung der Startparameter:

### **/H oder /?**

zeigt ein Hinweisfenster mit den möglichen Startparametern

### **/LANG=Language**

stellt manuell die entsprechende Sprache ein (DEU = Deutsch, ENG = Englisch)

### **/INSTALL**

installiert den *ChipInfo*-Treiber manuell (nur für Windows NT4/2000/XP/2003, siehe Installation/Deinstallation)

### **/UNINSTALL**

deinstalliert den *ChipInfo*-Treiber manuell (nur für Windows NT4/2000/XP/2003, siehe Installation/Deinstallation)

### **/DEBUG**

aktiviert den Debug-Modus und erstellt im aktuellen Verzeichnis die Datei *Debug.txt*. Diese Datei hilft den Programmentwicklern im Problemfall, weitere Vorgehensweisen zu entscheiden.

### **/REPORT=file.txt**

erstellt einen Report mit allen Optionen und speichert die Ergebnisse in der Datei *file.txt*. Wird kein Gleichheitszeichen oder ein Dateiname angegeben, verwendet *ChipInfo* automatisch den Namen *chipinfo.txt*.

### **/REPORTCHIP=file.txt**

erstellt einen Report in der Datei *file.txt*, der ausschließlich Chipsatz-Informationen enthält. Wird kein Gleichheitszeichen oder ein Dateiname angegeben, verwendet *ChipInfo* automatisch den Namen *chipinfo.txt*.

**/REPORTSYS=file.txt**

erstellt einen Report in der Datei *file.txt*, der ausschließlich System-Interna enthält. Wird kein Gleichheitszeichen oder ein Dateiname angegeben, verwendet *ChipInfo* automatisch den Namen *chipinfo.txt*.

Siehe auch:

Hilfe-Inhalt

Hilfe-Index

## **Danksagungen**

Wir bedanken uns bei allen Beta-Testern und Benutzern, die uns mit Vorschlägen, Kritiken und Fehlermeldungen geholfen haben, das Programm zu verbessern.

Für die englische Übersetzung der Dokumentation möchten wir insbesondere Dolf Westerveld danken.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## Chipsatz-Details

*ChipInfo* erkennt automatisch, welcher Chipsatz auf dem Mainboard arbeitet und welche Zusatzgeräte er anbindet. Die in diesem Bereich detailliert aufgeschlüsselten Chipsatz-Details beinhalten zunächst einen Kopfbereich mit Chipsatznamen und Stepping/Revision.

Anschließend folgen die eigentlichen Details, die in Gruppen der jeweiligen Chipsatzregister aufgeschlüsselt sind. Das bei Intel häufig vorzufindene Register *MCH Configuration* erscheint beispielsweise in einer separaten Zeile, gefolgt von den darin enthaltenen Daten. Zwischen zwei Registern wird jeweils eine Leerzeile eingefügt.

Bis auf den Kopfbereich werden alle Chipsatz-Details in englischer Sprache dargestellt, was sowohl die Registerbezeichnungen, Optionen und Ergebnisse betrifft. Durch eine Übersetzung würden diese aus der englischen Sprache stammenden Begriffe nicht mehr sinngemäß sein. Weiterhin wird der Begriff Register komplett vermieden und stattdessen der eigentliche Registername verwendet.

Handelt es sich um ein PCI-Gerät, das kein Chipsatz ist, für das aber Informationen anzeigbar sind, benennt *ChipInfo* diese Kategorie zu *Geräte-Details* um. Existiert im *ChipInfo*-Verzeichnis die Gerätedatenbank *PCI.DAT* nicht, zeigt *ChipInfo* trotzdem die anhand von Vendor- und Device-ID verfügbaren Chipsatz-Details an. Die Ermittlung des aus der Datenbank stammenden Chipsatznamens ist dann allerdings nicht möglich.

*ChipInfo* zeigt Details zu folgenden Chipsätzen an (diese Liste kann zwischen verschiedenen Programmversionen variieren):

### Intel:

Intel 434LX  
Intel 434NX

Intel 430VX  
Intel 430FX  
Intel 430TX  
Intel 430HX  
Intel 430MX

Intel 440FX

Intel 440LX/440EX  
Intel 440LX/440EX- Virtual PCI-to-PCI Bridge

Intel 440GX

Intel 440GX- PCI-to-PCI-Bridge

Intel 440BX/440ZX

Intel 440BX/440ZX- PCI-to-PCI-Bridge

Intel 450KX/GX- PCI Bridge

Intel 450KX/GX- Memory Controller

Intel 450NX- Memory & I/O Controller

Intel 450NX- PCI Expander Bridge

Intel 810(E/E2)

Intel 810(E/E2)-Graphics Device

Intel 815(E/P/EP/EM/G/EG)

Intel 815- AGP Bridge

Intel 815- Graphics Device

Intel 820

Intel 820- AGP Bridge

Intel 830M

Intel 830M- Graphics Device

Intel 830M- AGP Bridge

Intel 840

Intel 840- Hub Interface B Bridge

Intel 840- AGP Bridge

Intel 845(E/G/GL/GV/GE/PE/MP/MZ)

Intel 845- Host-to-AGP Bridge

Intel 845- Integrated Graphics Device

Intel 848(P/PE)

Intel 848- PCI-to-AGP Bridge

Intel 848- PCI-to-CSA Bridge

Intel 848- Overflow

Intel 850

Intel 850- AGP Bridge

Intel 852

Intel 852- GMCH Host-Hub Interface Bridge

Intel 852- GMCH Main Memory Control

Intel 852- GMCH Configuration Process

Intel 852- GMCH Integrated Graphics Device



Intel 855GM/GME  
Intel 855GM/GME- GMCH Host-Hub Interface Bridge  
Intel 855GM/GME- GMCH Main Memory Control  
Intel 855GM/GME- GMCH Configuration Process  
Intel 855GM/GME- GMCH Integrated Graphics Device

Intel 855PM  
Intel 855PM- Host-Hub Interface Bridge  
Intel 855PM- AGP Bridge  
Intel 855PM- Power Management

Intel 860  
Intel 860- Host-Hub Interface\_A Bridge  
Intel 860- Hub Interface\_B Bridge  
Intel 860- Hub Interface\_C Bridge  
Intel 860- AGP Bridge

Intel 865(P/PE/G/GV)  
Intel 865- PCI-to-AGP Bridge  
Intel 865- PCI-to-CSA Bridge  
Intel 865- Overflow  
Intel 865- Integrated Graphics

Intel 875P  
Intel 875P- PCI-to-AGP Bridge  
Intel 875P- PCI-to-CSA Bridge  
Intel 875P- Overflow

Intel E7205  
Intel E7205- Chipset Host RAS Controller  
Intel E7205- PCI-to-AGP Bridge

Intel E7500  
Intel E7500- DRAM Controller Error Reporting  
Intel E7500- HI\_B Virtual PCI-to-PCI Bridge

Intel E7501  
Intel E7501- Host RASUM Controller  
Intel E7501- HI\_B PCI-to-PCI Bridge  
Intel E7501- HI\_B PCI-to-PCI Bridge Error Reporting  
Intel E7501- HI\_C PCI-to-PCI Bridge  
Intel E7501- HI\_C PCI-to-PCI Bridge Error Reporting  
Intel E7501- HI\_D PCI-to-PCI Bridge  
Intel E7501- HI\_D PCI-to-PCI Bridge Error Reporting

Intel E7505  
Intel E7505- Chipset Host RAS Controller  
Intel E7505- PCI-to-AGP Bridge  
Intel E7505- HI\_B PCI-to-PCI Bridge  
Intel E7505- HI\_B PCI-to-PCI Bridge Error Reporting

Intel MISA  
Intel MPC12

Intel MPIIX

Intel PIIX/PIIX3  
- PCI to ISA Bridge  
- IDE Controller  
- USB Controller

Intel PIIX4  
- PCI to ISA Bridge  
- IDE Controller  
- USB Controller  
- Power Management

Intel ICH(0)  
- Hub Interface to PCI Bridge  
- LPC Interface Bridge  
- IDE Controller  
- USB 1.1 Controller  
- SMBus Controller  
- AC'97 Audio Controller  
- AC'97 Modem Controller

Intel ICH2(M)  
- LAN Controller  
- Hub Interface to PCI Bridge  
- LPC Interface Bridge  
- IDE Controller  
- USB 1.1 Controller  
- SMBus Controller  
- AC'97 Audio Controller  
- AC'97 Modem Controller

Intel ICH3-M/S  
- LAN Controller  
- Hub Interface to PCI Bridge  
- LPC Interface Bridge  
- IDE Controller

- USB 1.1 Controller
- SMBus Controller
- AC'97 Audio Controller
- AC'97 Modem Controller

#### Intel ICH4

- LAN Controller
- Hub Interface to PCI Bridge
- LPC Interface Bridge
- IDE Controller
- USB 1.1 Controller
- USB 2.0 Controller
- SMBus Controller
- AC'97 Audio Controller
- AC'97 Modem Controller

#### Intel ICH5

- LAN Controller
- Hub Interface to PCI Bridge
- LPC Interface Bridge
- IDE Controller
- SATA Controller
- USB 1.1 Controller
- USB 2.0 Controller
- SMBus Controller
- AC'97 Audio Controller
- AC'97 Modem Controller

#### Intel PCI 64 Hub

- Hub Interface to PCI Bridge
- Advanced Interrupt Controller (APIC)

#### Intel PCI/PCI-X 64-Bit Hub 2

- Hub Interface to PCI Bridge
- I/OxAPIC Interrupt Controller
- Hot Plug Controller

### **AMD:**

#### AMD 751

AMD 751- AGP and PCI-to-PCI Bridge

#### AMD 756

AMD 756- PCI-ISA Bridge

AMD 756- EIDE Controller

AMD 756- Power Management

AMD 756- USB Controller

AMD 761

AMD 761- PCI-to-PCI Bridge

AMD 762

AMD 762- PCI-to-PCI Bridge

AMD 766

AMD 766- PCI-ISA Bridge

AMD 766- EIDE Controller

AMD 766- Power Management

AMD 766- USB Controller

**Via:**

Via PIPC (VT82C586A/B)

Via PIPC- PCI-to-ISA Bridge

Via PIPC- IDE Controller

Via PIPC- USB Controller

Via PIPC- Power Management

Via Apollo VP (VT82C580VP)

Via Apollo VPX (VT82C580VPX)

Via Apollo VP2 (VT82C595)

Via Apollo VP3 (VT82C597)

Via Apollo VP3- PCI-to-PCI-Bridge

Via ProSavage PM133 (VT8605)

Via ProSavage PM133- PCI-to-PCI-Bridge (VT8605)

Via Apollo MVP3 (VT82C598MVP)

Via Apollo MVP3- PCI-to-PCI-Bridge (VT82C598MVP)

Via Apollo Pro (VT82C691)

Via Apollo Pro- PCI-to-PCI-Bridge (VT82C691)

Via Apollo PLE133 (VT8601A)

Via Apollo PLE133- PCI-to-AGP-Bridge (VT8601A)

Via Apollo PM601 (VT8601)

Via Apollo PM601- PCI-to-AGP-Bridge (VT8601)

Via Apollo Pro266 (VT8633)

Via Apollo Pro266- PCI-to-PCI-Bridge (VT8633)

Via KT133 (VT8363)  
Via KT133- PCI-to-PCI Bridge (VT8363)

Via KX133 (VT8371)  
Via KX133- PCI-to-PCI Bridge (VT8371)

Via CLE266  
Via CLE266- PCI-to-PCI Bridge  
Via CLE266- Graphics Controller

**SiS:**

SiS 630  
SiS 630- PCI-to-PCI Bridge  
SiS 630- IDE Controller  
SiS 630- GUI Accelerator  
SiS 630- LPC Bridge  
SiS 630- Ethernet Controller  
SiS 630- Audio Accelerator

Bitte haben Sie Verständnis dafür, dass die einzelnen Chipsatz-Details nicht in dieser Hilfe aufgeführt und erklärt sind, weil dies den Rahmen sprengen würde.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## Geräte-Details

*ChipInfo* erkennt neben dem Mainboard-Chipsatz die dazugehörigen Geräte, die zumeist als Southbridge oder wie bei Intel als ICH-Bustein bezeichnet werden und eine Fülle verschiedener Geräte beinhalten. Hierzu gehören häufig die PCI-to-ISA-Bridge, integrierte Grafikkarten, IDE-, USB- und SMBus-Controller sowie die AC97-Schnittstelle für Audio und Modem. Die in diesem Bereich detailliert aufgeschlüsselten Geräte-Details beinhalten zunächst einen Kopfbereich mit Gerätenamen und Stepping/Revision.

Anschließend folgen die eigentlichen Details, die in Gruppen der jeweiligen Gerätereister aufgeschlüsselt sind. Zwischen zwei Registern wird jeweils eine Leerzeile eingefügt.

Bis auf den Kopfbereich werden alle Geräte-Details in englischer Sprache dargestellt, was sowohl die Registerbezeichnungen, Optionen und Ergebnisse betrifft. Durch eine Übersetzung würden diese aus der englischen Sprache stammenden Begriffe nicht mehr sinngemäß sein. Weiterhin wird der Begriff Register komplett vermieden und stattdessen der eigentliche Registername verwendet.

Handelt es sich bei dem Gerät um die Host-to-PCI-Bridge und liegen anzeigbare Chipsatz-Details vor, benennt *ChipInfo* diesen Bereich in *Chipsatz-Details* um. Existiert im *ChipInfo*-Verzeichnis die Gerätedatenbank *PCI.DAT* nicht, zeigt *ChipInfo* trotzdem die anhand von Vendor- und Device-ID verfügbaren Geräte-Details an. Die Ermittlung des aus der Datenbank stammenden Gerätenamens ist dann allerdings nicht möglich.

Bitte haben Sie Verständnis dafür, dass die einzelnen Geräte-Details nicht in dieser Hilfe aufgeführt und erklärt sind, weil dies den Rahmen sprengen würde.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## Geräte-Fähigkeiten

Die *Geräte-Fähigkeiten* (englisch Capabilities List) sind immer dann sinnvoll, wenn keine Details zum entsprechenden PCI-Gerät bekannt sind. Anhand der definierten Strukturen in der PCI-Spezifikation lassen sich für diese Art von Geräten trotzdem wichtige Details aufschlüsseln. Aber auch für bekannte Geräte mit umfassenden Geräte-Details stellen die Geräte-Fähigkeiten oftmals eine sinnvolle Erweiterung dar.

Der Schalter *Geräte-Fähigkeiten* befindet sich in der Gruppe Chipsatz-Details bei Chipsätzen bzw. Geräte-Details bei Nicht-Chipsätzen. Wird diese Funktion vom jeweiligen Gerät nicht unterstützt, ist der Schalter nicht anwählbar. Kommt die Demo-Version von *ChipInfo* zum Einsatz, ist diese Funktion standardseitig deaktiviert.

Die nachfolgenden Punkte sind Bestandteil der Geräte-Fähigkeiten:

- PCI Power Management Interface
- AGP - Accelerated Graphics Port
- VPD - Virtual Product Data
- Slot Identification
- Message Signaled Interrupts
- CompactPCI - Hot Swap
- PCI-X
- Reserved for AMD
- Vendor Specific
- Debug Port
- CompactPCI - Central Resource Control
- PCI Hot-Plug

Zu beachten ist, dass insbesondere für die Bereiche *CompactPCI - Hot Swap*, *Reserved for AMD*, *CompactPCI - Central Resource Control* und *PCI Hot-Plug* kaum Details angezeigt werden können, weil diese entweder nicht bekannt sind oder die Beschaffung entsprechender Spezifikationen erfolglos war. Wir werden zukünftig versuchen, diese Bereiche mit Daten zu ergänzen.

Jedes Gerät kann jeden einzelnen Bereich der Geräte-Fähigkeiten mehrfach unterstützen, was in der Liste auch entsprechend berücksichtigt wird. Die Bereichsgruppen werden durch den Namen in eckigen Klammern gekennzeichnet, damit schnell sichtbar ist, wann eine neue Gruppe beginnt.

Bitte haben Sie Verständnis dafür, dass die einzelnen Details der Gruppen nicht in dieser Hilfe aufgeführt und erklärt sind, weil dies den Rahmen sprengen würde.

Siehe auch:  
[Hilfe-Inhalt](#)

## Hilfe-Index



## PCI-Details

*ChipInfo* ermittelt für jedes PCI-Gerät allgemeine Details, die hauptsächlich der Geräte-Identifizierung dienen und sich beim verwendeten Registerbereich vor dem PCI-Header befinden (00h bis 0Fh). Aufgeteilt werden diese Details in 4 verschiedene Gruppen, auf die nachfolgend eingegangen wird:

### **Geräte-Identifikation:**

Für eine eindeutige Geräteerkennung sind hauptsächlich die *Vendor-* und *Device-ID* notwendig, wobei erstere von der PCISIG ([www.pcisig.com](http://www.pcisig.com)) vergeben wird, und zweitere vom Hersteller frei wählbar ist. Diese Daten reichen in vielen Fällen aus, die exakte Gerätebezeichnung mithilfe der Gerätedatenbank *PCI.DAT* zu erkennen. Zusätzlich existieren Unterkennungen, die als *SubVendor-* und *SubDevice-ID* bekannt sind. Beide Kennungen sind zwar Bestandteil des PCI-Headers, sind aber ebenso wichtig für eine Identifikation und werden deswegen in diesem Bereich mit angezeigt.

In weiteren Zeilen folgt die *Gerätrevision*, die *Gerätenummer*, *-Funktion* und *-Bus*, der *Headertyp*, die Prüfung auf ein *Multifunktionsgerät* (z.B. die PCI-to-ISA-Bridge), und der *Geräte-Typ*. Letzterer ist wiederum in 3 Gruppen unterteilt, die als *Basis-Klasse*, *Unterklasse* und *Schnittstelle* bezeichnet werden. Die Rückgabergebnisse der *Basis-* und *Unterklasse* werden ermittelt und in Klammern mit angeführt.

### **Geräte-Kontrolle:**

Anhand der *Geräte-Kontrolle* (englisch Device Control) lassen sich einige Daten über die Geräte-Eigenschaften ermitteln. Bestandteil dieser Kontrolldaten ist beispielsweise die *Kontrolle für Wartezyklen*, *aufgetretene Paritätsfehler*, die Funktion *VGA Palette Snoop*, der *Schreibzugriff mit Invalidierung*, *Spezialzyklen*, ob das aktuelle Gerät *Bus-Master* ist und ob auf die Bereiche des *Speichers* sowie *I/O* zugegriffen wird.

### **Geräte-Status:**

Der *Geräte-Status* (englisch Device Status) spiegelt den aktuellen Arbeitszustand des entsprechenden PCI-Gerätes wieder. Hierzu gehört beispielsweise, ob die *Fähigkeiten-Liste* unterstützt wird, die *ChipInfo* im Fenster *Geräte-Fähigkeiten* ermittelt. Weiterhin existieren Status-Werte, ob ein *Paritätsfehler* beim Bus-Master-Gerät aufgetreten ist, ob *Systemfehler* signalisiert und erkannt wurden und mit welchem *Geschwindigkeitsmodus* ein Gerät arbeitet.

### **Sonstiges:**

Dieser Bereich beinhaltet Informationen, die in keine der anderen Kategorien passen. Bestandteil ist die *Größe der CacheLine*, die vom Master-Gerät mit Schreibrechten für den Speicher implementiert wird und gleichzeitig als Anhaltspunkt dafür dient, mit

welchen Lesebefehlen unterschiedlicher Geschwindigkeit sich auf den Speicher zugreifen lässt.

Die *Latenz-Zeit* wird in PCI-Takten angegeben und definiert bei PCI Bus-Master Geräten eine Zeitspanne, die zwischen zwei Zugriffen gewartet werden muss.

Der sogenannte *Build-In Self Test* (BIST) stellt einen im Gerät implementierten Testmechanismus zur Verfügung. Während des Tests arbeitet das Gerät nicht standardseitig am Bus, sondern schaltet in einen speziellen Testmodus. *ChipInfo* zeigt hier an, ob der Selbsttest unterstützt wird, und wenn dies der Fall ist, ob der Test erfolgreich bestanden wurde.

Anders als bei den Chipsatz-Details, werden die hier ermittelten Daten entsprechend der Programmiersprache übersetzt. Ist die deutsche Sprache eingestellt, erscheinen die angezeigten Details auch in Deutsch.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **PCI-Headerdetails**

Während der PCI-Registerbereich 00h bis 0Fh den festgelegten Standardbereich abdeckt und primär für die Geräteidentifikation zuständig ist, dient der Bereich 10h bis 3Fh dem sogenannten *PCI-Header*.

Hierbei handelt es sich um 3 verschiedene Headertypen, in die jedes PCI-Gerät eingeteilt ist. Jeder Headertyp hat eine andere Registerbelegung und spezialisiert sich zumeist auf den Typ des Gerätes. Folgende Headertypen sind verfügbar:

- Typ 0: Standard-Header für alle gängigen Geräte
- Typ 1: Header für PCI-to-PCI Bridges
- Typ 2: Header für PCI-to-CardBus Bridges

Als Bridge bezeichnet man sinngemäß eine Brücke zwischen zwei Chipsatzkomponenten, wobei beispielsweise eine PCI-to-ISA Bridge die Verbindung zwischen PCI- und ISA-Bus herstellt. Lässt sich ein Gerät nicht in Headertyp 1 oder 2 einteilen, wird es automatisch Headertyp 0 zugewiesen.

Alle 3 Header werden in die Spalten *Beschreibung* und *Ergebnis* eingeteilt und beinhalten zunächst einmal den Bereich *Allgemeine Informationen* mit dem *Headertyp*, gefolgt von *Basis-Adressen*, *Latenz-Zeiten*, *Bus-Daten* und *Interrupt-Zuweisungen*. Letztere sind vor allem interessant, um den zugewiesenen PCI- und ISA-Interrupt zu ermitteln.

Header 0 und 2 beinhalten Daten zur *Subsystem-Identifizierung*. Hierbei handelt es sich um die *SubVendor-* und *SubDevice-ID*, die vergleichbar mit der regulären Vendor- und Device-ID eine genauere Geräte-Erkennung zulassen. *ChipInfo* greift für die Geräteerkennung auf die Datenbank *PCI.DAT* zu, kann aber aufgrund der vom Hersteller frei einstellbaren ID's nicht immer ein Ergebnis liefern.

Anders als bei den [Chipsatz-Details](#), werden die hier ermittelten Daten entsprechend der Programmiersprache übersetzt. Ist die deutsche Sprache eingestellt, erscheinen die angezeigten Details auch in Deutsch.

Bitte haben Sie Verständnis dafür, dass die einzelnen Details der Header nicht in dieser Hilfe aufgeführt und erklärt sind, weil dies den Rahmen sprengen würde.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **PCI-Register**

Jedes PCI-Gerät verfügt über einen 256 Byte großen Adressraum, aus dem *ChipInfo* einen Großteil der angezeigten Informationen gewinnt. Das Fenster *PCI-Register* listet diese Register vollständig auf und teilt die Daten in folgende Spalten auf:

### **Register**

Nummeriert die Geräteregister fortlaufend in hexadezimaler Schreibweise. Begonnen wird mit 00h.

### **Dezimal**

Beinhaltet den Registerinhalt in dezimaler Schreibweise.

### **Hexadezimal**

Beinhaltet den Registerinhalt in hexadezimaler Schreibweise.

### **Binär (7-0)**

Beinhaltet den Registerinhalt in binärer Schreibweise. Bit 7 erscheint links und Bit 0 rechts.

### **Beschreibung**

Beinhaltet die Registerbezeichnung, die bis Register 3Fh aus definierten Strukturen innerhalb der PCI-Spezifikation stammen, und darüberhinaus gerätespezifisch vorliegen.

Aufgrund der Tatsache, dass einige Geräteregister als Word- oder LongWord-Register ausgelegt sind (16 bzw. 32 Bit), werden diese Register in das Byte-Format aufgeteilt. Hinter dem Beschreibungstext erscheint hierbei in Klammern, der wievielte Teil eines Registers angezeigt wird (beginnend mit 1).

Liegt die Spezifikation für das jeweilige PCI-Gerät beispielsweise als PDF-Dokument vor, lässt sich anhand der hier aufgelisteten Register selbst prüfen, welche Optionen aktiv und nicht aktiv sind.

Kommt die Demo-Version von *ChipInfo* zum Einsatz, werden die Register nur bis zum Register 7Fh aufgelistet.

Siehe auch:

Hilfe-Inhalt  
Hilfe-Index

## **System-Interna**

In der Registerkarte *System-Interna* werden alle wichtigen Rechner-Daten zusammengefasst und übersichtlich auf einer Seite angezeigt. Während des Programmstarts werden die Eckdaten ermittelt, so dass während der Programmlaufzeit keine zusätzliche Zeit dafür beansprucht wird. Aufgeteilt sind die Daten in folgende Bereiche:

### **Prozessor:**

Beinhaltet Details zum verwendeten Prozessor, inklusive *Hersteller* und *Typ*, *Taktfrequenz*, *Front Side Bus (FSB)*, *Multiplikator*, Daten zum *CPUID-Befehl* und den verwendeten Hardwarecaches *L1*, *L2* und *L3*. Ein Großteil der Daten innerhalb dieser Kategorie werden anhand des *CPUID-Befehles* gewonnen, der ab späteren 486er Prozessoren verfügbar ist und genaue Rückschlüsse auf den Prozessor mitsamt seinen Eigenschaften zulässt. Andere Ergebnisse wie beispielsweise der *Front Side Bus* oder *Multiplikator* werden mit sogenannten *Maschinen-spezifischen Registern (MSR)* ermittelt.

### **System:**

Dieser Bereich soll primär einen Überblick über die Systemkomponente *Mainboard* geben. Hierzu zählt der *Mainboardtyp*, der in erster Instanz mit der *BIOS-ID* und der Gerätedatenbank *MAINBRD.DAT* erkannt wird, und im Falle einer nicht vorhandenen Datenbank oder keines Eintrages innerhalb der Datenbank über das *Desktop Management Interface (DMI)* ermittelt wird. Letztere Erkennungsmöglichkeit liest hierbei die Attribute *BaseBoard.Manufacturer* und *BaseBoard.Product* aus. In der Zeile *PCI-Geräte* werden statistisch die Geräte des PCI-Bus ausgewertet, die *alle vorhandenen Geräte*, *Multifunktionsgeräte*, *PCI-to-PCI-Geräte* und *PCI-to-CardBus-Geräte* beinhalten.

Hinsichtlich des verwendeten BIOS zeigt *ChipInfo* den *BIOS-Typ* mit *Version* sowie in einer weiteren Zeile die *BIOS-ID* an. Diese Kennung existiert ausschließlich bei AMI- und Award-BIOSsen und beinhaltet herstellerepezifische Details – *ChipInfo* verwendet sie zusätzlich zur Erkennung des Mainboards.

### **Betriebssystem:**

Hier fasst *ChipInfo* die allgemeinen Betriebssystemangaben zusammen, wozu der *Betriebssystemname*, die *Suite*, *Version*, *Service Pack* sowie *Build* gehören. Letztere dient häufig zur Unterscheidung zwischen verschiedenen Beta-Versionen eines Betriebssystems.

### **physikalischer Speicher:**

Der *physikalische Speicher* befindet sich im Sinne von Speichermodulen auf dem

Mainboard und wird hier mit den Angaben *Gesamt*, *Benutzt* und *Frei* dargestellt. Dabei erscheinen eventuelle mehrere Speichermodule als eine feste Speichergröße, da Windows diese als eine Größe ermittelt.

### **virtueller Speicher:**

Der *virtuelle Speicher* ist die Erweiterung des physikalischen Speichers um weitere Speicherbereiche. Anhand der Auslagerungsdatei nutzt Windows Festplattenspeicher und fügt diesen dem betriebssystemtechnischen Speicherbereich hinzu. Anwendungsprogramme merken nicht, ob sie sich gerade im Arbeitsspeicher oder Festplattenspeicher befinden, da beide Angaben im Sinne des virtuellen Speichers eine zusammenhängende Größe darstellen.

Die Ergebnisse werden hier mit den Angaben *Gesamt*, *Benutzt* und *Frei* dargestellt.

### **Programm-Datenbanken:**

Als Statistik und Versionskontrolle ermittelt *ChipInfo* einige Kenndaten der Gerätedatenbanken *PCI.DAT* und *MAINBRD.DAT*, insofern sich diese im aktuellen Verzeichnis befinden. Im Feld direkt nach dem Datenbanknamen symbolisiert ein *Ja*, dass die Datenbanken verfügbar und eingebunden sind. In weiteren Feldern erscheint die jeweilige *Datenbankversion*, das *Veröffentlichungsdatum* und die *Größe* in Bytes. Fehlt eine Datenbank und kann nicht eingebunden werden, meldet *ChipInfo* diesen Umstand direkt beim Programmstart.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## IRQ-Routing Tabelle

Mainboards mit PCI-Bus besitzen physikalische PCI-Steckplätze, die externe PCI-Karten aufnehmen können, und virtuelle PCI-Steckplätze, die an Chipsatz-Komponenten angebunden sind. Jeder Steckplatz besitzt wiederum 4 Interrupt-Pins, die als *INTA#*, *INTB#*, *INTC#* und *INTD#* bezeichnet werden.

Innerhalb eines x86-Computersystems stehen 16 reguläre Interrupts (PIRQ, Programmable Interrupt Request) zur Verfügung, die bereits für die Verwendung basistechnischer Komponenten vordefiniert sind (z.B. Interrupt 4 für COM1 oder Interrupt 12 für PS/2-Mäuse). Freie Interrupts werden während des Power-On Self Test (POST) vom BIOS automatisch so verteilt, dass die verfügbaren Interrupt-Pins den freien Interrupts zugewiesen werden. Dieser Vorgang lässt sich in vielen BIOS-Versionen manuell aushebeln und nach freien Wünschen selbst anordnen.

Die *IRQ-Routing Tabelle* stellt einen Bereich innerhalb des BIOS dar, in dem diese Zuweisungen abgelegt sind. *ChipInfo* liest die eingetragenen Werte aus und zeigt sie in der Registerkarte *IRQ-Routing Tabelle* an.

Aufgeteilt werden diese Daten in folgende Bereiche:

### **Allgemeine Informationen:**

Beinhaltet allgemeine Details, die den Kopf der *IRQ-Routing Tabelle* symbolisieren. Dazu gehört die Signatur *\$PIR*, die den Beginn der Tabelle andeutet und sich im Speicherbereich F0000h bis FFFFFh befindet. Darauf folgend erscheint die *Version*, die *Größe* der Tabelle in Bytes, welche IRQs exklusiv für die Verwendung mit PCI-Geräten bestimmt sind, die sogenannten *Miniport-Daten* und eine *Prüfsumme*.

### **PCI Interrupt Routing-Gerät:**

Hier wird das entsprechende PCI-Gerät ermittelt, das primär für die Interrupt-Zuordnung zuständig ist. Dazu gehören Angaben zur jeweiligen *Bus-Nummer* (meistens Bus 00h), sowie die *Geräte-* und *Funktionsnummer* sowie die *Gerätebezeichnung*. Letztere Angabe wird aus der Gerätedatenbank *PCI.DAT* ermittelt.

### **Kompatibler PCI Interrupt-Router:**

In diesem Bereich wird die *Vendor-* und *Device-ID* sowie die dazugehörige *Bezeichnung* des *kompatiblen PCI Interrupt Routers* abgezeigt. Dieses Gerät ist häufig ein Bestandteil des Chipsatzes (z.B. PCI-to-ISA Bridge), und kann Zuweisungen zwischen den PCI-basierten Interrupt-Pins und den regulären Interrupts vornehmen. Kompatibel deswegen, weil mehrere Interrupt-Router existieren können und der kompatible Router die gleichen Methoden der Zuweisung und des Interrupt-Managements verwendet, die das *PCI Interrupt-Routing Gerät* verwendet. Der Bezeichnungstext wird hierfür aus der



Gerätedatenbank *PCI.DAT* ermittelt.

### **Slot-Einträge:**

In diesem Kernbereich werden die *PCI-Slots* (Steckplätze) nummeriert aufgelistet und ausgewertet. Unterteilt werden die Daten in folgende Unterbereiche:

#### Nr:

Nummeriert den jeweiligen PCI-Steckplatz, beginnend bei 1.

#### Bus/Gerät/Funktion:

Beinhaltet die *Bus-Nummer*, auf dem sich der PCI-Steckplatz befindet (meistens 00h). Weiterhin wird die *Geräte- und Funktionsnummer* angezeigt.

#### Slot:

Stellt die *Slot-Nummer* dar, wobei Nullwerte dafür stehen, dass es sich um einen virtuellen PCI-Steckplatz handelt. Werte ab 1 stellen hingegen die tatsächlichen physikalischen PCI-Steckplätze in nummerierter Form dar.

#### Typ:

Als *Slot-Typ* kommen physikalische Steckplätze (*PCI-Slot*) und virtuelle Steckplätze (*Mainboard*) in Frage. Die ersten stehen als reale Steckplätze für externe PCI-Karten zur Verfügung (z.B. Netzwerkkarte), die zweiten existieren ebenfalls auf dem PCI-Bus, binden aber tatsächlich Chipsatz-Komponenten an (z.B. PCI-to-ISA-Bridge).

#### IRQ:

Hier werden die 4 *Interrupt-Pins* jedes PCI-Steckplatzes aufgeführt (*INTA#*, *INTB#*, *INTC#* und *INTD#*).

#### Link-Wert:

Stellt einen Wert für die Verknüpfung zwischen dem jeweiligen *Interrupt-Pin* und regulärer Interrupts dar. Hat dieses Feld den Wert 00h, ist das jeweilige *Interrupt-Pin* an keinen Interrupt angebunden. Ist der Wert höher, deutet dies auf eine Zuordnung zu anderen *Interrupt-Pins*, aber nicht zu regulären Interrupts hin.

#### Link-Matrix:

Mit der *Link-Matrix* ist ein hexadezimaler Wert gemeint, der in verschlüsselter Form die Interrupt-Zuweisungen der *Interrupt-Pins* beinhaltet.

## IRQ-Matrix:

Die *IRQ-Matrix* beinhaltet diejenigen Interrupts, für die eine Zuordnung zum jeweiligen *Interrupt-Pin* möglich wären. Weiterführende Umstände werden in diesem Ergebnis ignoriert, beispielsweise ob Interrupts frei sind, Mehrfachbelegungen möglich sind oder Interrupt-Sharing eingerichtet ist.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **Report erstellen**

Anhand der *Reportfunktion* lassen sich die ermittelten Daten wahlweise in eine Datei speichern oder an einen Drucker senden. Hierfür wählt man in der oberen Kategorie *Reportziel* das entsprechende Ziel aus. Über den rechten Schalter *Datei wählen* bzw. *Drucker wählen* lässt sich für Dateien der *Dateiname* sowie Pfad, und für Drucker der *Druckername* sowie dessen treiberbezogene Einstellungen festlegen.

Die unteren zwei Kategorien *Chipsatz/PCI* und *Sonstiges* lassen sich einzeln ab- sowie anwählen und stellen die von *ChipInfo* bereitgestellten Analysefunktionen dar. Die Kategorie *Chipsatz/PCI* beinhaltet grundlegend Analysen, die für jedes einzelne PCI-Gerät individuell ermittelt werden können. Weitere Analysepunkte im Bereich *Sonstiges* lassen sich ungeachtet von der Anzahl vorhandener PCI-Geräte nur einmal ermitteln.

Wichtig und insbesondere bei vielen PCI-Geräten entscheidend ist die Option *Alle PCI-Geräte*, die im aktiven Zustand alle PCI-Geräte abarbeitet und die aktivierten PCI-Analysen durchführt. Ist die Option nicht aktiv, wird ausschließlich das aktuell geöffnete PCI-Gerät verwendet.

Abschließend wird der Reportvorgang durch Drücken des Schalters *Erstelle Report* gestartet. *ChipInfo* zeigt mit einem *Statusfenster* während der Reportausführung an, welche Analyse gerade aktiv ist. Abschließend wird ein *Hinweisfenster* angezeigt, durch welches man nach dessen Bestätigung zum Programm zurückgelangt.

Über zusätzliche Startparameter existieren 3 weitere Möglichkeiten, um den Report direkt von der Kommandozeile auszuführen. Diese Möglichkeiten eignen sich besonders gut zur Inventarisierung von Netzwerken, da *ChipInfo* ohne die grafische Benutzeroberfläche (GUI) gestartet wird.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **Überblick über die Programmanalysen**

Analysen, die für jedes PCI-Gerät individuell ermittelt werden können:

[Chipsatz-Details / Geräte-Details](#)

[Geräte-Fähigkeiten](#)

[PCI-Details](#)

[PCI-Headerdetails](#)

[PCI-Register](#)

Analysen, die für ein Computersystem einmalig ermittelt werden können:

[System-Intern](#)

[IRQ-Routing Tabelle](#)

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **FAQ - Häufig gestellte Fragen**

**Frage:** Beim Start unter Windows 95, 98 und NT4 erhalte ich die Fehlermeldung "LOADER ERROR: The procedure entry point VarNot could not be located in the dynamic link library oleant32.dll" (oder ähnlich). Was bedeutet das ?

**Antwort:** Die von uns verwendete Programmiersprache Borland Delphi 6 hat scheinbar einige grundlegende Änderungen im ServicePack 2 erhalten (siehe Internetseite von Borland: <http://bdn.borland.com/article/0,1410,28841,00.html>). Dadurch muss bei einigen Systemen mit Windows 95, 98 und NT4 ein Update von DCOM installiert werden.

Nachfolgende Links führen auf die Internetseite von Microsoft.

Windows 95 DCOM Update: [http://www.microsoft.com/com/dcom/dcom95/dcom1\\_3.asp](http://www.microsoft.com/com/dcom/dcom95/dcom1_3.asp)  
Windows 98 DCOM Update: [http://www.microsoft.com/com/dcom/dcom98/dcom1\\_3.asp](http://www.microsoft.com/com/dcom/dcom98/dcom1_3.asp)  
Windows NT4 ServicePack 4: <http://www.microsoft.com/ntserver/nts/downloads/recommended/NT4SvcPk4/NT4SvcPk4.asp>

Diese Updates stellen ein Minimum dar, um unsere Programme auch auf diesen Systemen starten zu können.

**Frage:** Wie kann ich schnellstmöglich Updates beziehen ?

**Antwort:** Registrierte Kunden und damit Benutzer der Professional-Version erhalten mit dem Keyfile eine Zugangskennung und ein Passwort für unseren geschützten Kundenbereich. Wird eine neue Programmversion von *ChipInfo* freigegeben, erhalten die registrierten Anwender eine Email mit einer Übersicht der verbesserten bzw. neuen Programmfunktionen und können sich nach dem Einloggen die aktuelle Professional-Version herunterladen. Nach dem Entpacken macht das bei der Bestellung übermittelte Keyfile die Professional-Version startfähig.

**Frage:** Wie wird der Debug-Modus aktiviert ?

**Antwort:** Der Startparameter */DEBUG* aktiviert den Debug-Modus und erstellt im aktuellen Verzeichnis die Textdatei *debug.txt*. Sinnvoll ist diese Funktion dann, wenn *ChipInfo* Probleme beim Startvorgang hat oder bestimmte Hardwarekomponenten nicht erkennt. Im Problemfall sollte man diese Datei an die Entwickler senden

(chipinfo@pcanalyser.de), damit das Problem umgehend gelöst werden kann. Beachten Sie bitte, dass der Programmstart mit aktiviertem Debug-Modus länger dauert. Das hängt damit zusammen, dass die Debug-Datei vor jedem Schreibvorgang geöffnet und nach dem Schreibvorgang geschlossen wird, um die Dateiintegrität zu gewährleisten.

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

## **Professional-Version**

Die Demo-Version beinhaltet ungefähr die Hälfte des Funktionsumfangs, den die kostenpflichtige Professional-Version zu bieten hat. Grundsätzlich soll die Demo-Version ausschließlich zum Vorab-Test dienen, wobei der eingeschränkte Funktionsumfang ausreichende Einblicke in das Programm gewährleistet.

Zu den vollwertigen Funktionen der Professional-Version gehören insbesondere vollständige Chipsatz-Informationen, die IRQ-Routing Tabelle und 3 verschiedene Batch-Modi für automatische Programmstarts.

Somit ist die Professional-Version einerseits für Anwender interessant, die eine Inventarisierung eines Netzwerkes benötigen, andererseits erschließen sich noch mehr Details, die der ambitionierte Anwender anhand von weiteren Quellen in seinem System ermitteln lassen kann.

Weiterhin unterstützen Sie durch den Erwerb der Professional-Version die Weiterentwicklung, die möglichst aktuell der fortlaufenden Marktveränderung im Chipsatz-Sektor nachkommt.

Alle Unterschiede zwischen der Demo- sowie Professional-Version können Sie der nachfolgenden Liste entnehmen.

<b>Programm-Funktion</b>	<b>Demo-Version</b>	<b>Professional-Version</b>
Vollständige Chipsatz-Informationen	nein	ja
Geräte-Fähigkeiten (wenn unterstützt)	nein	ja
Mehr als 128 anzeigbare PCI-Register	nein	ja
IRQ-Routing Tabelle	nein	ja
Batch-Modus für automatischen Programmstart	nein	ja
Demo-Hinweisfenster	ja	nein

Siehe auch:

[Hilfe-Inhalt](#)

[Hilfe-Index](#)

