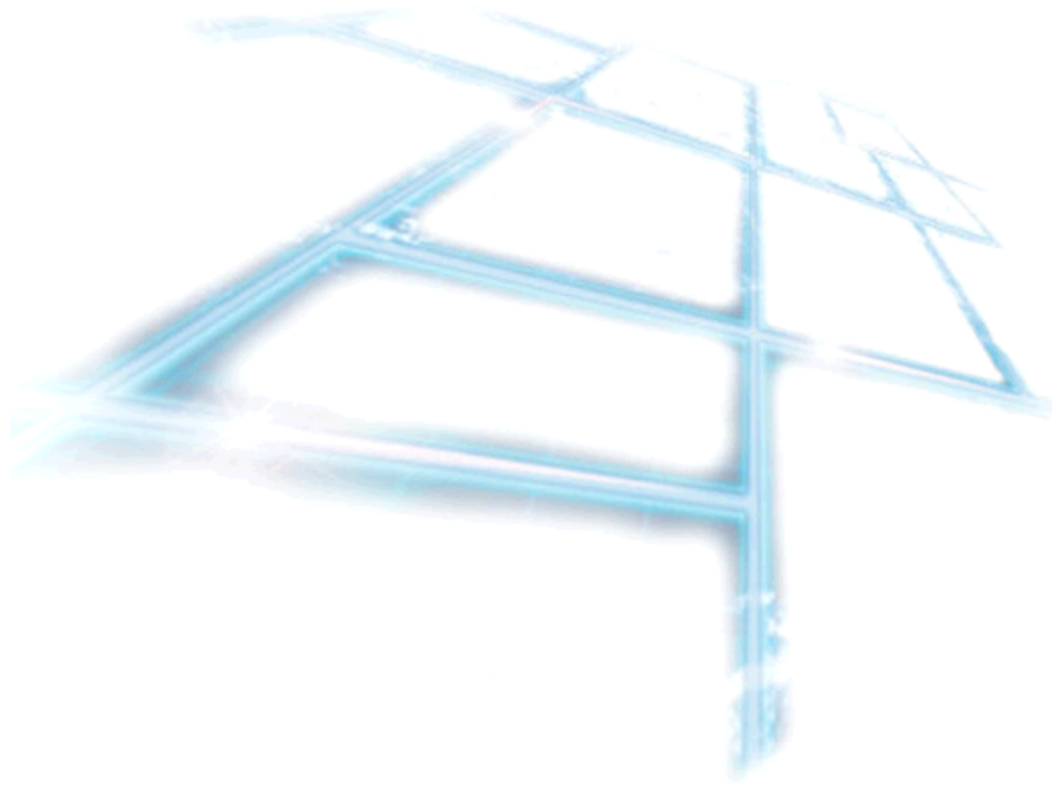


PCI-DECODER2 / PCI-DECODER3



Copyright © QUANCOM Informationssysteme GmbH

Alle Angaben in diesem Handbuch sind nach sorgfältiger Prüfung zusammengestellt worden, gelten jedoch nicht als Zusicherung von Produkteigenschaften. QUANCOM haftet ausschließlich in dem Umfang, der in den Verkaufs- und Lieferbedingungen festgelegt ist. Weitergabe und Vervielfältigung dieses Handbuches und die Verwertung seines Inhaltes sowie der zum Produkt gehörenden Software sind nur mit schriftlicher Erlaubnis von QUANCOM gestattet. Änderungen, die dem technischen Fortschritt dienen, bleiben vorbehalten.

Wesseling, Oktober 2010 Version 3.9.9

Inhaltsverzeichnis

Kapitel I	Überblick	1
1	Einführung.....	1
2	Unsere Erfahrung ist ihr Gewinn.....	1
3	Kommunikation mit unseren Kunden.....	1
4	Änderungen zu diesem Handbuch und Softwareupdates.....	2
5	Lieferumfang.....	2
Kapitel II	Technische Hardware Beschreibung	3
1	Decoderinformationen.....	3
2	Technische Daten.....	3
	DAIGLED0 und DIAGLED1.....	3
Kapitel III	Eigene Entwicklungen mit dem Decoder	5
1	Hardware Register.....	5
	Registerübersicht des PCI-Decoders.....	5
	Das IRQ-Konfigurationsregister 0xF8.....	5
	Das IRQ-Statusregister 0xF9.....	6
	TTL – Eingänge.....	7
	TTL – Ausgänge.....	8
2	Steuerleitungen für Entwicklungen.....	9
	Programmierung durch I/O-Zugriffe.....	9
	Beispiel für eine Abfrage der Eingangsports.....	9
	Beispiel für die Ansteuerung der Ausgangsports.....	10
	Einfache A/D- Wandlung mit dem PCI-Decoder.....	10
	Verwendung von Interrupts.....	10
	D0..D31 und CS0..CS3.....	11
3	Programmierung der eigenen Hardware.....	13
	Welche Software brauche ich ?.....	13
	QLIB: High Level Programmierung (Windows XP / 2000 / NT / ME / 98 / 95).....	15
	QLIB (QUANCOM Driver Library).....	15
	Installation und Allgemeines über die QLIB.....	16
4	Installation der Treiber und der QLIB unter Windows XP/2000.....	17
5	Installation der Treiber und der QLIB unter Windows NT4.x.....	19
6	Installation der Treiber und der QLIB unter Windows ME/98/95.....	20
Kapitel IV	QLIB Befehle	23
1	Einfache QLIB Befehle (QAPI...).....	23
	Digitale Lese- und Schreibfunktionen.....	23
	Verwaltungs- und sonstige Funktionen.....	24
2	Erweiterte QLIB Befehle (QAPIExt...).....	25
	Digitale Lesefunktionen.....	25
	Digitale Schreibfunktionen.....	26
	Verwaltungsfunktionen.....	27
	Sonstige Funktionen.....	27
	Jobcodes für QAPISpecial und QAPIExtSpecial.....	29

3	Programmbeispiele für die QLIB.....	30
	Ansteuerung der TTL Ausgänge.....	30
	Ansteuerung der TTL- Eingänge.....	30
	Schreiben in eine Speicherstelle.....	31
	Lesen einer Speicherstelle.....	31
4	Ansteuerung im Memory- Modus.....	32
Kapitel V	Anhang	33
1	Allgemeine Informationen.....	33
	Welche Arten von Problemen können auftreten wenn die Karte unter DOS läuft.....	34
	Probleme mit Karten unter Windows 98/95 und Windows 2000/NT.....	34
2	Kunden Support und Hilfe.....	36
3	Technisches Support Formular.....	39
4	Hardware und Software Konfigurationsformular.....	40
5	Dokumentations Formular.....	42
6	Warenzeichen.....	43

1.1 Einführung

Wir beglückwünschen Sie zum Kauf Ihres QUANCOM PCI-Decoders. Sie haben sich für ein Produkt entschieden, dessen Eigenschaften und Funktionalität den neuesten Stand der Technik darstellen.

Zu den besonderen Eigenschaften dieser Karte gehören:

Eigenschaften des Decoders

- geringer Entwicklungsaufwand für eigene Schaltungen
- einfach programmierbar
- diverse Beispielprogramme in verschiedenen Programmiersprachen
- Treiberunterstützung unter Windows XP, 2000, NT und Me/98/95 mit der QLIB
(QUANCOM Driver Library)

1.2 Unsere Erfahrung ist ihr Gewinn

Wir von QUANCOM sind auf die Entwicklung für Hardware und Software spezialisiert und gehören mittlerweile zu einem der führenden Lieferanten für Meßtechnik und Automatisierung. In ihrem Entwicklungszentrum hat QUANCOM eine eindrucksvolle Produktpalette entwickelt.

1.3 Kommunikation mit unseren Kunden

QUANCOM möchte gerne Ihren Kommentar zu unseren Produkten und zu unseren Handbüchern erhalten und ist außerdem an Ihren Anwendungen interessiert, die Sie mit unseren Produkten entwickeln. Wir möchten gleichzeitig helfen, wenn Sie Probleme haben und um dies zu vereinfachen enthält dieses Handbuch Kommentar- und Konfigurationsformulare, mit denen man direkt mit uns in Verbindung treten kann. Diese Formulare befinden sich in dem Kapitel "**Dokumentationsformular**" am Ende dieses Handbuches.

1.4 Änderungen zu diesem Handbuch und Softwareupdates

QUANCOM - Produkte zeichnen sich u.a. durch stetige Weiterentwicklung aus. Aktuelle Informationen über Änderungen können Sie den README - Dateien auf der Installationsdiskette oder CD entnehmen. Weitere Informationen und kostenlose Softwareupdates können Sie jederzeit auf den QUANCOM Internet – WWW-Seiten unter www.quancom.de erhalten.

1.5 Lieferumfang

- PCI-Decoder Chip
- QUANCOM CD mit Benutzerhandbuch (PDF)

Sollten eine oder mehrere Komponenten fehlen wenden Sie sich bitte an Ihren Händler. QUANCOM behält sich das Recht vor, Änderungen im Lieferumfang ohne Vorankündigung vorzunehmen.

2.1 Decoderinformationen

Für Entwicklung von PC-Einsteckkarten für den PCI-Bus wurde dieser PCI-Decoder der PCIPROTO nun auch für Ihre eigenen Entwicklungen bereitgestellt. Der Chip wird von uns fertig programmiert und kann bei Verwendung des ISP-download Anschlusses auch nach der Montage auf Ihrer Platine aktualisiert werden. Verwenden Sie dazu einen Anschluß, wie er in Schaltbild der PCIPROTO dargestellt ist.

Die Dekodierung des PCI-Busses erfolgt im PCI-Decoder-Chip. Erstellt diverse Signale, wie z.B. einen 32 Bit-Datenbus, IOWR, IORD und weitere Bussignale zur Verfügung.

2.2 Technische Daten

- **Allgemein**
 - Datenbus:** D0...D31
 - Adreßbus:** A2...A7
 - Steuerbus:** RD, WR zur Steuerung der Datenrichtung, RESET (Reset vom PC), INT (Interruptanforderung), CLKX (500 Hz)
 - Freigabesignale:** für max. 256 I/O Adressen, CS0..CS3
- **Geschwindigkeit** 7 Millionen Zugriffe a 32 Bit pro SekundeMemory-Modus
- **I/O Adressen** PCI
- **Temperaturbereich** 0...70 °C
- **Abmessungen**
 - PCI-DEC3:** Gemäß PQFP 176
 - PCI-DEC2:** Gemäß TQFP 208

2.2.1 DAIGLED0 und DIAGLED1

Die **DIAGLED0** (*Konfigurations- LED*) leuchtet so lange bis das PCI-Interface konfiguriert ist. Dieses geschieht beim Start oder Reset des Computers, oder wird von dem Betriebssystem durchgeführt.

Die **DIAGLED1** (*Access LED*) leuchtet auf, wenn Zugriffe (z.B. lesen, schreiben) auf den PCI-Decoder stattfinden.

3.1 Hardware Register

3.1.1 Registerübersicht des PCI-Decoders

Offset	r/w	Bit	Erklärung
0	w		Bar-LED (Offset ist im GAL festgelegt)
0	r		8* DIP Schalter (Offset ist im GAL festgelegt) zurücklesen
0xF8	w		IRQ-Konfigurationsregister schreiben
0xF8	r		IRQ-Konfigurationsregister zurücklesen
0xF9	r		IRQ-Statusregister lesen
0xFC	r		TTL-IN 1..8 lesen
0xFD	r		TTL-IN 9..16 lesen
0xFE	w		TTL-OUT 1..8 setzen
0xFF	w		TTL-OUT 9..16 setzen

*r/w = read (lesen)/write (schreiben)
0x= hex

3.1.2 Das IRQ-Konfigurationsregister 0xF8

r/w	Bit	Wertigkeit	Erklärung
w	0	1	IRQ enable (bei Bit = 1)
r			Inhalt des Registers zurücklesen
w	7	128	zurücksetzen des IRQ-Statusregisters 0xF9 (beim schreiben von 0x80 in das Register 0xF8)

*r/w = read (lesen)/write (schreiben)
0x= hex

3.1.3 Das IRQ-Statusregister 0xF9

r/w	Bit	Wertigkeit	Erklärung
r	0	1	IRQ ist ausgelöst worden (1=aktiv). Dies wird durch das Schreiben des Wertes 0x80 auf das Register 0xF8 zurückgesetzt.

*r/w = read (lesen)/write (schreiben)
0x= hex

Der PCI-Decoder erlaubt dem Anwender durch setzen des Pins IRQIN auf der Stiftheiste JP8 einen Interrupt auszulösen. Der PCI-Decoder löst damit einen IRQ auf der INTA Leitung im PCI-Bus Systems aus. Der IRQ wird in dem Register 0xF9 signalisiert und kann durch einen Schreibvorgang auf Bit 7 Register 0xF8 wieder zurückgesetzt werden. Sehen Sie dazu das Bild im Kapitel 4.5 "Verwendung von Interrupts"

Um den IRQ auszuwerten kann z. B. die folgende Codesequenz genutzt werden:

Beispiel : Auswertung des Interrupts

```
unsigned int iobase;          /* Basisadresse der Karte */  
  
void InterruptServiceRoutine () {  
    if (inp(iobase + 0xF9) & 1) {  
        printf("Interrupt wurde von dieser Karte ausgelöst");  
    }  
    if (inp(iobase + 0xF9) & 2) {  
        printf("Interrupt wurde durch ein externes Ereignis  
        ausgelöst");  
        /*reset Interruptflag*/  
        outp(iobase+0xF8,0x81);  
    }  
}
```

3.1.4 TTL – Eingänge

Lesen Eingang	Wertigkeit	Zugriff: Basis +0xFC (TTL-IN 1..8)	Zugriff: Basis +0xFD (TTL-IN 9..16)
Bit 0	1	TTL-IN 1	TTL-IN 9
Bit 1	2	TTL-IN 2	TTL-IN 10
Bit 2	4	TTL-IN 3	TTL-IN 11
Bit 3	8	TTL-IN 4	TTL-IN 12
Bit 4	16	TTL-IN 5	TTL-IN 13
Bit 5	32	TTL-IN 6	TTL-IN 14
Bit 6	64	TTL-IN 7	TTL-IN 15
Bit 7	128	TTL-IN 8	TTL-IN 16

Basis ist die I/O-Basisadresse der Karte

Beispiel : Zustand des Eingangs TTL-IN 8 anzeigen

```
unsigned int iobase;          /* Basisadresse der Karte */
lesen() {
    int in;
    printf("TTL-IN 8");

    in = inp(iobase + 0xFC);   /* 8-Bit Wert bei Basisadress + 0xFC lesen*/
    if (in & 128)              /* Bit 7 (= TTL-IN 8) isolieren */
        printf("=1");
    else
        printf("=0");

    getch();                  /* Warten auf Tastendruck */
}
```

3.1.5 TTL – Ausgänge

Schreiben Ausgang	Wertigkeit	Zugriff: Basis +0xFE (TTL-OUT 1..8)	Zugriff: Basis +0xFF (TTL-OUT 9..16)
Bit 0	1	TTL-OUT 1	TTL-OUT 9
Bit 1	2	TTL-OUT 2	TTL-OUT 10
Bit 2	4	TTL-OUT 3	TTL-OUT 11
Bit 3	8	TTL-OUT 4	TTL-OUT 12
Bit 4	16	TTL-OUT 5	TTL-OUT 13
Bit 5	32	TTL-OUT 6	TTL-OUT 14
Bit 6	64	TTL-OUT 7	TTL-OUT 15
Bit 7	128	TTL-OUT 8	TTL-OUT 16

Basis ist die I/O-Basisadresse der Karte

Beispiel : TTL – OUT 1 setzen

```
unsigned int iobase;      /* Basisadresse der Karte */
setzen() {
    outp(iobase+0xFE,1);  /* 8 Bit Wert auf BASIS+0xFE übergeben: */
                        /* TTL-OUT1 an und der Rest aus */
    getch();              /* Auf Tastendruck warten */
}
```

3.2 Steuerleitungen für Entwicklungen

Signalname	Richtung	Beschreibung
WR#	out, aktiv low	Schreiben auf eine I/O- Adresse
RD#	out, aktiv low	Lesen von einer beliebigen I/O- Adresse
CS0#	out, aktiv low	zeigt gültigen Chipselect für D0...D7
CS1#	out, aktiv low	zeigt gültigen Chipselect für D8...D15
CS2#	out, aktiv low	zeigt gültigen Chipselect für D16...D23
CS3#	out, aktiv low	zeigt gültigen Chipselect für D24...D31
CLKOUT	out	Takt mit 503,54 Hz (ca. 2 ms Takt)
A2..A7	out	Adreßleitungen für Portbereich BASE + 0...255
D0..D31	in/out	32 Bit Datenbus
IN0.. IN15	in	16 Bit Eingang auf JP3 (BASE + 0xFC/0xFD)
OUT0.. OUT15	out	16 Bit Ausgang auf JP3 (BASE + 0xFE/0xFF)
IRQIN	in (pullup)	Interrupt-Signal zum PCI-Bus
TDI, TDO, TMS, TCK	In / out	ISP download gem. PCI-Decoder
DIAGLED0	Out	PCI-Config LED
DIAGLED1	out	PCI-Access LED
DEVSEL0...DEVSEL7	In (pullup)	Device-ID (0-0xFF)
DEVSEL8	In (pullup)	
DEVSEL9	In (pullup)	
RESET#	out, aktiv low	Zurücksetzen aller Signalleitungen

3.2.1 Programmierung durch I/O-Zugriffe

Die unten aufgeführten Beispiele zeigen, wie der PCI-Decoder durch I/O- Zugriffe programmiert werden kann. Damit diese Beispiele funktionieren, muss die Karte mit dem SW10 = On konfiguriert worden sein.

3.2.2 Beispiel für eine Abfrage der Eingangsports.

Das folgende Programmbeispiel zeigt, wie einfach Programme für den QUANCOM PCI-Decoder geschrieben werden können. (I/O- Zugriff)

Beispiel : Programmierung einer Port- Eingangsabfrage unter C und DOS / Windows

```
#include <stdio.h>

void main () {
    unsigned int BaseIO=0xXXXX; /*Hier die Basisadresse der Karte
                                eintragen (durch PCINFO setzen/lesen )
                                */
    unsigned char port=0; /* Portnummer des Eingangs */

    while !(kbhit()) {
        printf("Port %d hat Wert %0x\n" /* Ausgabe des */
            ,port /* Eingangs am Port */
            ,inp(BaseOI+port));

        for (long int j=0;j <=200000;j++); /* warten*/
    }
}
```

3.2.3 Beispiel für die Ansteuerung der Ausgangsports.

Die Ausgangsports können wie folgt angesprochen werden: (I/O- Zugriff)

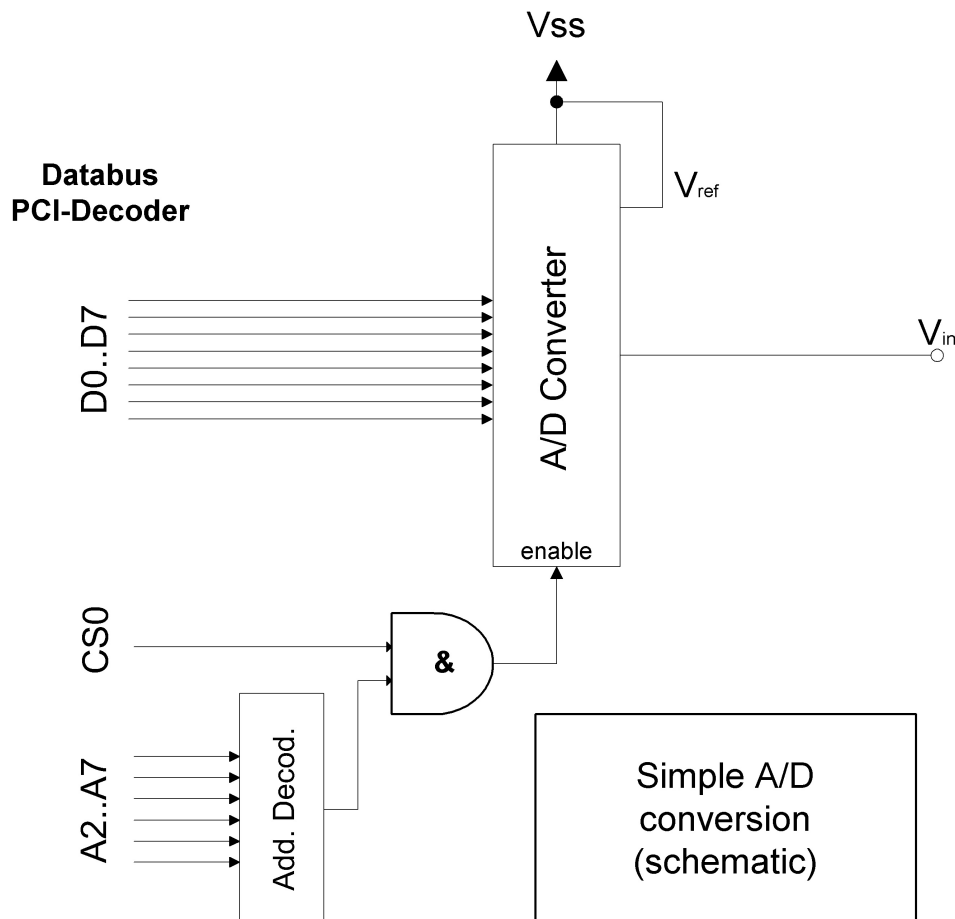
Beispiel : Programmierung der Ausgabeports unter C und DOS / Windows

```
#include <stdio.h>

void main () {
    unsigned int BaseIO=0xXXXX; /*Hier die Basisadresse der Karte
                                eintragen (durch PCINFO setzen/lesen )
                                */
    unsigned char port=0;      /* Portnummer des Eingangs */
    outp(BaseIO + port, 0xFF); /*Setzen aller Bits im Register port*/
}
```

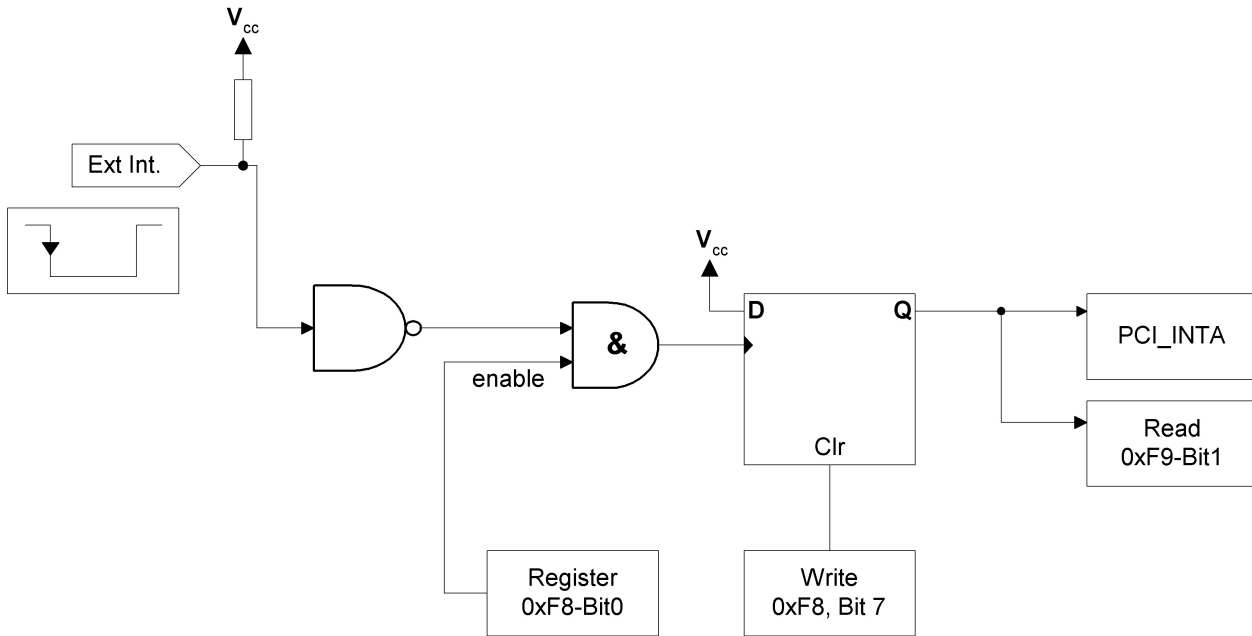
3.2.4 Einfache A/D- Wandlung mit dem PCI-Decoder

Das Beispiel zeigt den schematischen Aufbau eines A/D- Wandlers



3.2.5 Verwendung von Interrupts

Wenn Ihre Anwendung Interrupts auslösen soll, so verwenden Sie das Konfigurationsregister (0xF8) um die entsprechenden Ereignisse zu maskieren und abzufragen. Das folgende Flußdiagramm erläutert das IRQ- Handling auf dem PCI-Decoder.



3.2.6 D0..D31 und CS0..CS3

Nachdem die Basisadresse bekannt ist, wird auf der PCI-Decoder (durch PCIINFO unter DOS), durch lesende oder schreibende Befehle zugegriffen.

Da der PCI-Bus ein 32- / 64 bittiger Bus ist und die Datenzugriffe über diesen Bus mit 32 Bit erfolgen, werden für die Adresselektion nur die Adressleitungen ab A2 übertragen. Daher sind auf dem PCI-Decoder auch keine A0 und A1 vorhanden.

Datenzugriffe mit 8 / 16 / 32 Bit werden auf demj PCI-Decoder durch setzen der unterschiedlichen Chipselect (CS0..CS3) Leitungen signalisiert.

Die folgende Tabelle erläutert den Zusammenhang zwischen den CS- Leitungen und den verschiedenen Zugriffen auf die Karte.

Zugriff	Datenbreite	C S0	C S1	C S2	C S3
base + 0	8	■			
base + 1	8		■		
base + 2	8			■	
base + 3	8				■
base + 0	16	■	■		
base + 2	16			■	■
base + 0	32	■	■	■	■

Eine Adressdekodierung muss daher zusätzlich über die Leitungen A2..A7 erfolgen. Der Anwender hat damit die Möglichkeit einen Adressbereich von 0..255 abzudecken. Für die Verwaltung des PCI-Decoder sind folgende Adressen bereits vergeben:

Adresse	Beschreibung
0x0	LED-Bar und DIL-Schalter durch das GAL
0xF8	Konfigurations / Statusregister
0xFC + 0xFD	TTL-IN 0..15
0xFE + 0xFF	TTL-OUT 0..15

3.3 Programmierung der eigenen Hardware

Die Programmierung Ihrer Anwendung erfolgt je nach Konfiguration des PCI-Decoders über die direkte Port- Adressierung oder durch Zugriff auf die entsprechenden Speicherbereiche. Ein Zugriff kann in 8- / 16- oder 32-bittiger Breite erfolgen, wobei die CS0..CS3 Leitungen dann wie oben beschrieben von dem PCI- Controller gesetzt werden. Für bis zu 7 Signale kann die Chipselect und Adressdekodierung von dem auf des PCI-Decoders befindlichen GAL erfolgen.

3.3.1 Welche Software brauche ich ?

Die Software die ich brauche ist abhängig von der jeweiligen Anwendung und dem jeweiligen Betriebssystem. Um Zugriff per Programm auf die Karte zu bekommen bestehen folgende Möglichkeiten:

- **Methode 1:** Direkter I/O Zugriff (Direkter Zugriff auf die Hardware Register des Betriebssystems)
- **Methode 2:** High-Level Programmierung (Zugriff auf die Karte über die QLIB) unter Windows XP / 2000 / NT4.x / ME / 98 / 95. Hiermit ist es möglich die Karte z.B. über Visual-C, Visual-Basic, Borland Delphi, Lotus Notes u.a. Compilern und Interpretern anzusprechen.
- **Methode 3:** Installation der QLIB in Zusammenhang mit einem anderen Programm. Beispiele zu dem Einsatz mit Labview von National Instruments finden Sie nach der Installation der QLIB im Verzeichnis samples\gpib des QLIB-Verzeichnisses oder auf unseren Internetseiten unter:

http://www.quancom.de/qprod01/deu/files/download.gpib_vi.zip /\$file/gpib_vi.zip

Wenn Sie **Methode 1** oder **2** anwenden möchten, benötigen Sie den Quelltext der Anwendung. Sie sind selbst für das Hinzufügen der Befehle in Ihre Anwendung verantwortlich. Um diese Methoden zu benutzen sind Programmierkenntnisse erforderlich..

Methode 3 erlaubt einem das man die QUANCOM Karte mit einer bestehenden Software laufen lassen kann z.B. LabView oder Agilent VEE . Dafür müssen Sie als erstes die QLIB von der Installations CD installieren. Hinweise rund um die QLIB und Installation entnehmen

Sie bitte dem QLIB Handbuch, welches sich ebenfalls auf der CD befindet. Auf der CD finden Sie gleichzeitig einige Beispielprogramme für LabView und Agilent VEE.

3.3.2 QLIB: High Level Programmierung (Windows XP / 2000 / NT / ME / 98 / 95)

3.3.2.1 QLIB (QUANCOM Driver Library)

Die QLIB (die Abkürzung für **QUANCOM LIBrary**) bietet die Möglichkeit, alle QUANCOM-Karten unter den Betriebssystemen Windows XP/2000/NT und ME/98/95 und den Programmiersprachen C/C++/Delphi/Visual Basic anzusprechen. Sie wird zu allen QUANCOM-Karten mitgeliefert und gestattet dem Anwender durch die Einfachheit der Befehle, die QLIB in eigene Applikationen einzubinden. Die Befehle und Funktionen gelten für alle Betriebssysteme.

Unterstützte Betriebssysteme:

- Microsoft Windows Vista / XP / 2000 / NT 4.0 / ME / 98 / 95 und Linux

Unterstützte Compiler:

C / C++

- Borland C++ 3.1, 4.x, 5.x
- Microsoft® Visual C++ 1.x, 2.x, 4.x, 5.x, 6.x

Pascal

- Borland Turbo Pascal

Delphi

- Borland Delphi

Basic

- Microsoft® Visual Basic 3.x, 4.x, 5.x; 6.x

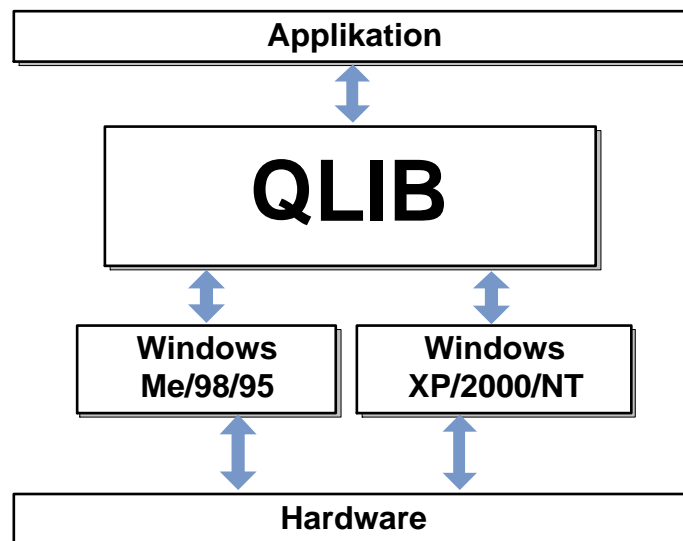
Grafische Programmiersprache:

- Agilent VEE von Agilent
- LabView® von National Instruments

3.3.2.2 Installation und Allgemeines über die QLIB

Für weitere Informationen über den Installationsprozess und die Programmierung mit der QLIB (z.B. vorbereitende Maßnahmen, Linken von Libraries und Header-Files hinzufügen, etc.) schlagen Sie bitte in dem Handbuch für die QLIB nach oder schauen Sie in die Hilfedatei zur QLIB (qlib.hlp auf der Installations CD).

Die folgenden Kapitel beschreiben die Installation der Treiber für das jeweilige Betriebssystem (Windows XP/2000/NT und Me/98/95) und das jeweilig verwendete Bussystem. Die Installationsvorgänge differieren je nach verwendeter Hard-/software.



Programmieren Sie alle QUANCOM-Karten Betriebssysteme unabhängig. Die QLIB (**QUANCOM LIBRARY**) bietet die Möglichkeit, alle QUANCOM-Karten und Modulen unter den Betriebssystemen Windows XP/2000 / NT4.x und ME/98/95 (NT4.x/95 ohne USB Unterstützung) und den Programmiersprachen C / C++ / Delphi / Visual Basic mit einfachen Befehlen anzusprechen. Es sind also keine Kenntnisse in der Treiberentwicklung notwendig. Sie wird ausserdem zu allen QUANCOM-Karten/-Modulen mitgeliefert und erleichtert dem Anwender die Integration in eigene Applikationen.



Je nach Betriebssystem und Bussystem (PCI, ISA, USB, PAR) sind die Installationen unterschiedlich. Achten Sie auf die Überschriften der folgenden Kapitel und wählen Sie sich die passende Installation aus.

3.4 Installation der Treiber und der QLIB unter Windows XP/2000

Wenn Sie die QLIB von unserer Internet Seite heruntergeladen haben, lesen Sie bitte die Informationen der Readme Datei vor der Installation der Treiber durch.



Die Installation der Software ist nur mit Administratorenrechten möglich. Sollten Sie diese Rechte nicht besitzen, so wenden Sie sich bitte an den zuständigen Administrator.

1. Schritt : Treiber Installation:

Nachdem Sie die Karte in einen freien Steckplatz gesteckt haben und Ihr System neu gestartet haben erkennt Windows automatisch eine neue Karte. Windows wird daraufhin den passenden Treiber für die Karte verlangen. Legen Sie die **QLIB Installations CD** ins CD-Laufwerk.

- Wenn Windows die Dialogbox “Neue Hardware gefunden” beim Start anzeigt, wählen Sie “Weiter”.
- Danach wählen Sie bitte “Nach einem passenden Treiber für das Gerät suchen (empfohlen)” und bestätigen anschließend mit “Weiter”.
- Als Quelle für die Suche nach den Treibern wählen Sie bitte “Andere Quelle angeben” und drücken auf “Weiter”.
- Windows öffnet jetzt einen Dateiauswahl Dialog. Klicken Sie bitte auf die Schaltfläche “Auswählen”, wechseln Sie auf das CD- Laufwerk und wählen Sie die Datei “**QLIBXDRV.INF**” im Verzeichnis **Win2000** oder **WinXP** aus. Klicken Sie auf “OK” um mit der Installation des Treibers fortzufahren.
- Bestätigen Sie in dem nächsten Fenster mit “Weiter”, dass Sie den Treiber installieren wollen und schließen die Installation des Treibers durch Drücken des Buttons “Fertigstellen” im folgenden Fenster ab.

2. Schritt : QLIB Installation:

Nach der Treiber Installation der QUANCOM Karte müssen Sie noch die **QLIB Software**

installieren.

- Klicken Sie auf **Start | Ausführen**. Wählen Sie das Programm **QUANCOM.EXE** aus und klicken auf "OK".
- Sollte nach dem Starten des Programmes **QUANCOM.EXE** eine Meldung erscheinen, die aussagt, dass Sie Ihr System zur Installation der Software neu starten müssen, klicken Sie auf "JA". Nach dem Neustart wird die Installation automatisch fortgeführt. (Dieser Schritt wird nur ausgeführt, wenn Sie noch keine Version des Software-Installers auf Ihrem System installiert haben)
- Im Fenster nach dem Neustart wird die Installation mit "Next" fortgeführt. (Bitte beachten Sie, dass Sie vor Beginn der Installation alle anderen laufenden Windowsprogramme beendet haben)
- Danach bestätigen Sie bitte die Lizenzbedingungen der QLIB Software durch Anklicken von "I accept the license agreement" und führen die Installation mit Drücken des "Next" Buttons fort.
- Bitte geben Sie nun Ihre persönlichen Benutzerdaten (Name; Organisation) ein, wählen ob Sie die QLIB Software nur für den aktuellen Benutzer oder für alle Benutzer dieses Computers installieren wollen und bestätigen Sie Ihre Auswahl mit "Next".
- Zum Ändern des Installationspfades für die QLIB Software klicken Sie bitte auf "Browse". Mit Klicken auf "Next" wird die Installation fortgeführt.
- Jetzt wählen Sie bitte noch die Installationsart der Software aus und bestätigen diese mit "Next". Es stehen drei Möglichkeiten zur Verfügung :
 - **Typical** (ohne Beispiele und Hilfedateien)
 - **Complete** (mit Beispielen und Hilfedateien)
 - **Custom** (frei wählbar)
- Die Softwareinstallation wird mit der Schaltfläche "Next" gestartet.
- Nach der erfolgreichen Installation der QLIB Software wird das Installationsprogramm mit drücken des Buttons "Fertigstellen" geschlossen.
- Um die Änderungen an Ihrem Rechner zu übernehmen werden Sie aufgefordert Ihren Rechner neu zu starten. Wählen Sie "YES" um Ihren PC neu zu starten bzw. "NO" um den PC später neu zu starten.

3.5 Installation der Treiber und der QLIB unter Windows NT4.x

Wenn Sie die QLIB von unserer Internet Seite heruntergeladen haben, lesen Sie bitte die Informationen der Readme Datei vor der Installation der Treiber durch.



Die Installation der Software ist nur mit Administratorenrechten möglich. Sollten Sie diese Rechte nicht besitzen, so wenden Sie sich bitte an den zuständigen Administrator.

2. Schritt : QLIB Installation :

Nachdem Sie die Karte in einen freien Steckplatz gesteckt haben und Ihr System neu gestartet haben legen Sie die **QLIB Installations CD** ins CD-Laufwerk.

- Klicken Sie auf **Start | Ausführen**. Wählen Sie das Programm **QUANCOM.EXE** aus und klicken auf "OK".
- Sollte nach dem Starten des Programmes **QUANCOM.EXE** eine Meldung erscheinen, die aussagt, dass Sie Ihr System zur Installation der Software neu starten müssen, klicken Sie auf "JA". Nach dem Neustart wird die Installation automatisch fortgeführt. (Dieser Schritt wird nur ausgeführt, wenn Sie noch keine Version des Software-Installers auf Ihrem System installiert haben)
- Im Fenster nach dem Neustart wird die Installation mit "Next" fortgeführt. (Bitte beachten Sie, dass Sie vor Beginn der Installation alle anderen laufenden Windowsprogramme beendet haben)
- Danach bestätigen Sie bitte die Lizenzbedingungen der QLIB Software durch Anklicken von "*I accept the license agreement*" und führen die Installation mit Drücken des "Next" Buttons fort.
- Bitte geben Sie nun Ihre persönlichen Benutzerdaten (Name; Organisation) ein, wählen ob Sie die QLIB Software nur für den aktuellen Benutzer oder für alle Benutzer dieses Computers installieren wollen und bestätigen Sie Ihre Auswahl mit "Next".
- Zum Ändern des Installationspfades für die QLIB Software klicken Sie bitte auf "Browse". Mit Klicken auf "Next" wird die Installation fortgeführt.
- Jetzt wählen Sie bitte noch die Installationsart der Software aus und bestätigen diese mit "Next". Es stehen drei Möglichkeiten zur Verfügung :
 - **Typical** (ohne Beispiele und Hilfedateien)

- **Complete** (mit Beispielen und Hilfedateien)
 - **Custom** (frei wählbar)
- Die Softwareinstallation wird mit der Schaltfläche "*Next*" gestartet.
 - Nach der erfolgreichen Installation der QLIB Software wird das Installationsprogramm mit drücken des Buttons "*Fertigstellen*" geschlossen.
 - Um die Änderungen an Ihrem Rechner zu übernehmen werden Sie aufgefordert Ihren Rechner neu zu starten. Wählen Sie "YES" um Ihren PC neu zu starten bzw. "NO" um den PC später neu zu starten.

3.6 Installation der Treiber und der QLIB unter Windows ME/98/95

Wenn Sie die QLIB von unserer Internet Seite heruntergeladen haben, lesen Sie bitte die Informationen der Readme Datei vor der Installation der Treiber durch.

1. Schritt : Treiber Installation:

Nachdem Sie die Karte in einen freien Steckplatz gesteckt haben und Ihr System neu gestartet haben erkennt Windows automatisch eine neue Karte. Windows wird daraufhin den passenden Treiber für die Karte verlangen. Legen Sie die **QLIB Installations CD** ins CD-Laufwerk.

- Wenn Windows die Dialogbox "*Neue Hardware gefunden*" beim Start anzeigt, wählen Sie "*Weiter*".
- Danach wählen Sie bitte "Nach einem passenden Treiber für das Gerät suchen (empfohlen) und bestätigen anschließend mit "*Weiter*".
- Als Quelle für die Suche nach den Treibern wählen Sie bitte "Geben Sie eine Position an" und drücken auf "Durchsuchen".
- Wechseln Sie auf das CD- Laufwerk und wählen Sie das Verzeichnis **WinME**, **Win98** oder **Win95**. Klicken Sie auf "OK" um mit der Installation des Treibers fortzufahren.
- Bestätigen Sie in den nächsten beiden Fenstern mit "*Weiter*", dass Sie den Treiber installieren wollen.
- Im folgenden Fenster schließen Sie die Installation des Treibers durch Drücken des Buttons "*Fertigstellen*" ab.

2. Schritt : QLIB Installation:

Nach der Treiber Installation der QUANCOM Karte müssen Sie noch die QLIB Software installieren.

- Klicken Sie auf **Start | Ausführen**. Wählen Sie das Programm **QUANCOM.EXE** aus und klicken auf "OK".
- Sollte nach dem Starten des Programmes **QUANCOM.EXE** eine Meldung erscheinen, die aussagt, dass Sie Ihr System zur Installation der Software neu starten müssen, klicken Sie auf "JA". Nach dem Neustart wird die Installation automatisch fortgeführt. (Dieser Schritt wird nur ausgeführt, wenn Sie noch keine Version des Software-Installers auf Ihrem System installiert haben)
- Im Fenster nach dem Neustart wird die Installation mit "Next" fortgeführt. (Bitte beachten Sie, dass Sie vor Beginn der Installation alle anderen laufenden Windowsprogramme beendet haben)
- Danach bestätigen Sie bitte die Lizenzbedingungen der QLIB Software durch Anklicken von "*I accept the license agreement*" und führen die Installation mit Drücken des "Next" Buttons fort.
- Bitte geben Sie nun Ihre persönlichen Benutzerdaten (Name; Organisation) ein, wählen ob Sie die QLIB Software nur für den aktuellen Benutzer oder für alle Benutzer dieses Computers installieren wollen und bestätigen Sie Ihre Auswahl mit "Next".
- Zum Ändern des Installationspfades für die QLIB Software klicken Sie bitte auf "Browse". Mit Klicken auf "Next" wird die Installation fortgeführt.
- Jetzt wählen Sie bitte noch die Installationsart der Software aus und bestätigen diese mit "Next". Es stehen drei Möglichkeiten zur Verfügung :
 - **Typical** (ohne Beispiele und Hilfedateien)
 - **Complete** (mit Beispielen und Hilfedateien)
 - **Custom** (frei wählbar)
 -
- Die Softwareinstallation wird mit der Schaltfläche "Next" gestartet.
- Nach der erfolgreichen Installation der QLIB Software wird das Installationsprogramm mit drücken des Buttons "*Fertigstellen*" geschlossen.
- Um die Änderungen an Ihrem Rechner zu übernehmen werden Sie aufgefordert Ihren Rechner neu zu starten. Wählen Sie "YES" um Ihren PC neu zu starten bzw. "NO" um den

PC später neu zu starten.

Stellen Sie sicher, dass die QLIB (QUANCOM Library) installiert ist.

Für weitere Informationen über die Installation und wie man die entsprechenden Dateien in die Anwendung einbindet entnehmen Sie bitte der “QLIB ” Dokumentation.

Das folgende Kapitel erläutert spezielle Befehle die benötigt werden, um die Karte mit der QLIB benutzen zu können.

Die folgende Auflistung enthält alle von dieser Karte verwendeten QLIB- Befehle. Diese unterscheiden sich in erweiterte (QAPIExt...) und einfache (QAPI...) Funktionen. Bei den einfachen Funktionen werden keine Kartenhandle´s (Kartenparameter) übergeben. Dadurch wird auch nur die erste vom System erkannte Karte eines Typs angesprochen und verwendet. Sollen mehrere Karten eines Typs in einem System verwendet werden, so müssen die erweiterten Funktionen verwendet werden.

4.1 Einfache QLIB Befehle (QAPI...)

4.1.1 Digitale Lese- und Schreibfunktionen

QAPIGetDI

```
ULONG QAPIGetDI ( ULONG cardid ULONG channel );
```

Mit der Funktion **QAPIGetDI** wird der Zustand eines 32 Bit breiten Digitalkanals einer DI-Karte eingelesen. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIPutDO

```
ULONG QAPIPutDO ( ULONG cardid ULONG channel ULONG value );
```

Mit der Funktion **QAPIPutDO** wird ein 32 Bit breiter Digitalwert auf den Kanal einer DO-Karte

ausgegeben. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

4.1.2 Verwaltungs- und sonstige Funktionen

QAPINumOfCards

ULONG **QAPINumOfCards** (void);

Die Funktion **QAPINumOfCards** gibt Auskunft darüber, wieviele Karten durch die QLIB abgedeckt werden.

QAPIGetCardInfo

LPCARDDATAS **QAPIGetCardInfo** (ULONG **cardid**);

Mit der Funktion **QAPIGetCardInfo** lassen sich Informationen über eine Karte holen.

QAPIGetCardInfoEx

ULONG **QAPIGetCardInfoEx**(ULONG **cardid** LPCARDDATAS **lpcd**);

Mit der Funktion **QAPIGetCardInfoEx** können Informationen einer Karte abgerufen werden. Diese werden in dem vom der Anwendung übergebenem Speicher geschrieben.

QAPISpecial

ULONG **QAPISpecial** (ULONG **cardid** ULONG **jobcode** ULONG **para1** ULONG **para2**);

Mit der Funktion **QAPISpecial** wird das Ausführen von kartenspezifischen Funktionen ermöglicht. Bei dem PCI-Decoders werden hiermit die einzelnen Speicherstellen und Register angesprochen. Eine Auflistung der jobcodes finden Sie im Kapitel **Jobcodes für QAPISpecial und QAPIExtSpecial**

4.2 Erweiterte QLIB Befehle (QAPIExt...)

4.2.1 Digitale Lesefunktionen

QAPIExtReadDI1

```
ULONG QAPIExtReadDI1 ( ULONG cardhandle ULONG channel ULONG mode );
```

Mit der Funktion **QAPIExtReadDI1** wird der Zustand eines 1 Bit breiten Digitalkanals einer DI-Karte eingelesen. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtReadDI8

```
ULONG QAPIExtReadDI8 ( ULONG cardhandle ULONG channel ULONG mode );
```

Mit der Funktion **QAPIExtReadDI8** wird der Zustand eines 8 Bit breiten Digitalkanals einer DI-Karte eingelesen. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtReadDI16

```
ULONG QAPIExtReadDI16 ( ULONG cardhandle ULONG channel ULONG mode );
```

Mit der Funktion **QAPIExtReadDI16** wird der Zustand eines 16 Bit breiten Digitalkanals einer DI-Karte eingelesen. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtReadDI32

```
ULONG QAPIExtReadDI32 ( ULONG cardhandle ULONG channel ULONG mode );
```

Mit der Funktion **QAPIExtReadDI32** wird der Zustand eines 32 Bit breiten Digitalkanals einer DI-Karte eingelesen. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

4.2.2 Digitale Schreibfunktionen

QAPIExtWriteDO1

```
void QAPIExtWriteDO1 ( ULONG cardhandle ULONG channel ULONG value ULONG mode );
```

Mit der Funktion **QAPIExtWriteDO1** wird ein 1 Bit breiter Digitalwert auf den Kanal einer DO-Karte ausgegeben. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtWriteDO8

```
void QAPIExtWriteDO8 ( ULONG cardhandle ULONG channel ULONG value ULONG mode );
```

Mit der Funktion **QAPIExtWriteDO8** wird ein 8 Bit breiter Digitalwert auf den Kanal einer DO-Karte ausgegeben. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtWriteDO16

```
void QAPIExtWriteDO16 ( ULONG cardhandle ULONG channel ULONG value ULONG mode );
```

Mit der Funktion **QAPIExtWriteDO16** wird ein 16 Bit breiter Digitalwert auf den Kanal einer DO-Karte ausgegeben. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

QAPIExtWriteDO32

```
void QAPIExtWriteDO32 ( ULONG cardhandle ULONG channel ULONG value ULONG  
mode );
```

Mit der Funktion **QAPIExtWriteDO32** wird ein 32 Bit breiter Digitalwert auf den Kanal einer DO- Karte ausgegeben. Bei dem PCI-Decoders sind das die TTL Ein- und Ausgänge.

4.2.3 Verwaltungsfunktionen

QAPIExtOpenCard

```
ULONG QAPIExtOpenCard ( ULONG cardid, ULONG devnum );
```

Mit der Funktion **QAPIExtOpenCard** wird eine Karte geöffnet.

QAPIExtCloseCard

```
void QAPIExtCloseCard( ULONG cardhandle );
```

Mit der Funktion **QAPIExtCloseCard** wird eine Karte geschlossen.

QAPIExtNumOfCards

```
ULONG QAPIExtNumOfCards (void);
```

Mit der Funktion **QAPIExtNumOfCards** wird abgefragt, welche momentan verwendeten Karten von der QLIB unterstützt werden.

4.2.4 Sonstige Funktionen

QAPIExtSpecial

```
ULONG QAPIExtSpecial ( ULONG cardhandle ULONG jobcode ULONG para1 ULONG
```

para2);

Mit der Funktion **QAPIExtSpecial** wird das Ausführen von kartenspezifischen Funktionen ermöglicht. Bei der PCIPROTO werden hiermit die einzelnen Speicherstellen und Register angesprochen. Eine Auflistung der jobcodes finden Sie im Kapitel Jobcodes für

QAPISpecial und **QAPIExtSpecial**

QAPIExtGetCardInfo

LPCARDDATAS **QAPIExtGetCardInfo**(ULONG **cardid**);

Mit der Funktion **QAPIExtGetCardInfo** lassen sich Informationen über eine Karte einlesen.

QAPIExtGetCardInfoEx

ULONG **QAPIExtGetCardInfoEx**(ULONG **cardid** LPCARDDATAS **lpcd**);

Mit der Funktion **QAPIExtGetCardInfoEx** können Informationen einer Karte abgerufen werden. Diese werden in dem vom der Anwendung übergebenem Speicher geschrieben.

QAPIExtReleaseCardInfo

void **QAPIExtReleaseCardInfo**(LPCARDDATAS **carddatas**);

Die Funktion **QAPIExtReleaseCardInfo** gibt mit **QAPIExtGetCardInfo** abgefragten Karteninformationen frei.

4.2.5 Jobcodes für QAPISpecial und QAPIExtSpecial

Jobcode	Funktion	Param1	Param2	Rückgabe
JOB_IOREAD_BYTE	Ein Byte von der Karte lesen	Offset ab Basis	-	Gelesenes Byte
JOB_IOREAD_WORD	Zwei Byte von der Karte lesen	Offset ab Basis	-	Gelesenes Word
JOB_IOREAD_LONG	Vier Byte von der Karte lesen	Offset ab Basis	-	Gelesener Long-Wert
JOB_IOWRITE_BYTE	Ein Byte zur Karte schreiben	Offset ab Basis	Zu schreibendes Byte	-
JOB_IOWRITE_WORD	Zwei Byte zur Karte schreiben	Offset ab Basis	Zu schreibendes Word	-
JOB_IOWRITE_LONG	Vier Byte zur Karte schreiben	Offset ab Basis	Zu schreibender Long-Wert	-
JOB_READ_DIP_SWITCH	Stellung derDIP-Schalter einlesen	-	-	Stellung der DIP-Schalter
JOB_FASTMEM_INIT	In Memory-mode: Pointer auf Basis-adresse holen	-	-	32 Bit Pointer auf Basis-adresse der Karte
JOB_FASTMEM_RELEASE	Pointer freigeben	Pointer von JOB_FASTMEM_INIT	-	-

4.3 Programmbeispiele für die QLIB

4.3.1 Ansteuerung der TTL Ausgänge

Beispiel : Programmieren der TTL Ausgänge mit der QLIB unter C

```
#include <windows.h>
#include <stdio.h>
#include <conio.h>

#include "qlib.h"

/*=====*/
/* Hauptprogramm      */
/*=====*/

void main () {

ULONG handle;      /* Handle der PCIPROTO */

/* Handle der Karte holen, bei Fehler raus */
if ((handle=QAPIExtOpenCard(PCIPROTO,0L)) == 0L) {
    printf("PCIPROTO konnte nicht geoeffnet werden\n");
    return;
}

/* Schleife durchlaufen und Werte an Karte senden*/
for (;;) {
    if (kbhit()!=0 && getch()==27) break;

    QAPIExtWriteD08(handle,0L,0x00L,0L);    /* 0b00000000 ausgeben */
    Sleep(500);
    QAPIExtWriteD08(handle,0L,0xFFL,0L);    /* 0b11111111 ausgeben */
    Sleep(500);
    QAPIExtWriteD08(handle,0L,0x55L,0L);    /* 0b01010101 ausgeben */
    Sleep(500);
    QAPIExtWriteD08(handle,0L,0xAAL,0L);    /* 0b10101010 ausgeben */
    Sleep(500);
}

QAPIExtCloseCard(handle);
}
```

4.3.2 Ansteuerung der TTL- Eingänge

Beispiel : Einlesen der TTL Eingänge mit der QLIB unter C

```
#include <windows.h>
#include <stdio.h>
#include <conio.h>

#include "qlib.h"

/*=====*/
/* Hauptprogramm      */
/*=====*/

void main () {

/* Handle der Karte holen, bei Fehler raus */
ULONG handle;      /* Handle der PCIPROTO */

if ((handle =QAPIExtOpenCard(PCIPROTO,0L)) == 0L) {
    printf("PCIPROTO konnte nicht geoeffnet werden\n");
    return;
}

/* Schleife liest Werte aus Karte und gibt diese aus */
for (;;) {
    if (kbhit()!=0 && getch()==27) break;

    printf("%04lx\n",QAPIExtReadDI16(handle,0L,0L));
    Sleep(500);
}

QAPIExtCloseCard(handle);
}
```

4.3.3 Schreiben in eine Speicherstelle

Beispiel : Schreiben einer Speicherstelle mit der QLIB unter C

```
#include <windows.h>
#include <stdio.h>
#include <conio.h>

#include "qlib.h"

/*=====*/
/* Hauptprogramm */
/*=====*/

void main () {

/* Handle der Karte holen, bei Fehler raus */
ULONG handle; /* Handle der PCIPROTO */

if ((handle =QAPIExtOpenCard(PCIPROTO,0L)) == 0L) {
    printf("PCIPROTO konnte nicht geoeffnet werden\n");
    return;
}

/* Schleife durchlaufen und Werte an Karte senden*/
for (;;) {
    if (kbhit()!=0 && getch()==27) break;

    /* 0b00000000 an Base+10 ausgeben */
    QAPIExtSpecial(handle,JOB_IOWRITE_BYTE,10L,0x00L);
    Sleep(500);
    /* 0b11111111 an Base+10 ausgeben */
    QAPIExtSpecial(handle,JOB_IOWRITE_BYTE,10L,0xFFL);
    Sleep(500);
    /* 0b01010101 an Base+10 ausgeben */
    QAPIExtSpecial(handle,JOB_IOWRITE_BYTE,10L,0x55L);
    Sleep(500);
    /* 0b10101010an Base+10 ausgeben */
    QAPIExtSpecial(handle,JOB_IOWRITE_BYTE,10L,0xAAL);
    Sleep(500);
}

QAPIExtCloseCard(handle);
}
```

4.3.4 Lesen einer Speicherstelle

Beispiel : Einlesen einer Speicherstelle der PCIPROTO mit der QLIB unter C

```
#include <windows.h>
#include <stdio.h>
#include <conio.h>

#include "qlib.h"

/*=====*/
/* Hauptprogramm */
/*=====*/

void main () {

/* Handle der Karte holen, bei Fehler raus */
ULONG handle; /* Handle der PCIPROTO */

if ((handle =QAPIExtOpenCard(PCIPROTO,0L)) == 0L) {
    printf("PCIPROTO konnte nicht geoeffnet werden\n");
    return;
}

/* Schleife liest Werte aus Karte und gibt diese aus */
for (;;) {
    if (kbhit()!=0 && getch()==27) break;

    /* Lesen und ausgeben der Speicherstelle BASE+10 */
    printf("%04lx\n",QAPIExtSpecial(handle,JOB_IOREAD_BYTE,10L,0L));
    Sleep(500);
}

QAPIExtCloseCard(handle);
}
```

4.4 Ansteuerung im Memory- Modus

Die Programmierung der PCIPROTO durch Speicherzugriffe erfolgt für Sie als Anwender fast genauso, wie eine I/O- Programmierung. Der größte Unterschied dabei ist, dass eine höhere Bandbreite als mit Zugriffen über die I/O- Ports erreicht werden kann.

Der Zugriff auf die Karte über das Speicherinterface geschieht prinzipiell genauso, wie jeder Zugriff auf einen anderen Speicherbereich, nur mit dem Unterschied, dass wir hierbei unsere Hardware ansteuern können.

Dieses Beispiel funktioniert nur, wenn die Karte auf Speicherzugriff konfiguriert wurde (SW10=off)

Beispiel : Ansteuerung der Karte über Speicherzugriffe

```
#include <windows.h>
#include <stdio.h>
#include "qlib.h"

void main () {

    ULONG handle=NULL;          /* Handle für Karte merken */
    char *BaseIO=NULL;         /* Speicherbasis wird hier gemerkt */

    unsigned char port=0;      /* Portnummer des Eingangs */

    /* Handle der Karte holen*/
    handle = QAPIExtOpenCard(PCIPROTO,0);

    /* Basisspeicheradresse der Karte holen */
    BaseIO = (char*) QAPIExtSpecial(handle,JOB_FASTMEM_INIT,0,0);

    while !(kbhit()) {
        printf("Port %d hat Wert %0xX\n"      /* Ausgabe des */
              ,port                          /* Eingangs am Port */
              ,*(BaseIO+port));

        for (long int j=0;j <=200000;j++);    /* warten*/
    }

    /*Memory-pointer wieder freigeben*/
    QAPIExtSpecial(handle, JOB_FASTMEM_RELEASE,(ULONG)BaseIO,0);
    /*Karte wieder freigeben*/
    QAPIExtCloseCard(handle);
}
```

5.1 Allgemeine Informationen

Kann ich Probleme bekommen wenn ich Netzwerkkarten, Soundkarten oder sonstige Erweiterungskarten in meinem PC habe?



Ja, es ist abhängig davon auf welchen I/O-Adressen die QUANCOM-Karte und die anderen liegen. Es können Ressourcenkonflikte auftreten, wenn mehrere Geräte die selbe I/O-Adresse benutzen. Entweder ändern Sie die Adresse der QUANCOM-Karte (Kapitel Leitfaden zur Schnellkonfiguration Hard- und Software) oder die Adresse der anderen Komponente.

Welchen Zweck erfüllt das Programm PCIInfo?

Das Programm PCIINFO zeigt die I/O-Adressen von allen PCI-Karten an, die sich im Computer befinden.

Wofür brauche ich das Programm PCISSETIO?

Das Programm erlaubt Ihnen ein manuelles Setzen der I/O- Adresse einer QUANCOM PCI Karte.

Wofür brauche ich das Programm PCIGETIO?

Das Programm PCIGETIO liest die Basis I/O-Adresse der QUANCOM PCI Karte aus dem Konfigurationsspeicher des PCI-Decoders aus.

5.1.1 Welche Arten von Problemen können auftreten wenn die Karte unter DOS läuft

- Kann ich die Karte mit einem Speicher Manager nutzen ? (QEMM 386,EMM386)
- Ja, aber durch einen Speicher Manager kann der Computer langsamer werden.

PCIINFO erkennt die PCI Karte nicht korrekt !

- Die Meldung "PCI Karte unbekannt" zeigt, dass es sich nicht um eine QUANCOM Karte handelt. PCIINFO ist nur für QUANCOM Karten hergestellt worden. Andere Komponenten, wie z. B. Grafikkarten, HDD Controller oder PCI-Bridges stehen in der Liste als "unbekannt".

5.1.2 Probleme mit Karten unter Windows 98/95 und Windows 2000/NT



Warum ist in der "Systemsteuerung" der Karten Dialog "QLIB" leer ?

- Es ist keine QUANCOM PCI Karte im System.
- Es sind keine Treiber für eine QUANCOM ISA Karte installiert.



Nach der Installation kommt die Meldung "QLIBNDRV.SYS nicht gefunden" oder "QLIBNDRV.VXD nicht gefunden" . Was kann ich machen ?

- Überprüfen Sie, ob die QLIB korrekt installiert ist. Für weitere Informationen rund um die Installation der QLIB sehen Sie im "QLIB" Handbuch nach, das auf der CD enthalten ist.
- Wenn Sie eine QUANCOM ISA Karte benutzen überprüfen Sie, ob die Treiber korrekt installiert sind.



Nach der Installation kommt die Meldung "Direct-IO interface cannot be initialized qmulti32.dll could not be initialized" - Was kann ich machen?

- Überprüfen Sie, ob die QLIB korrekt installiert ist. Für weitere Informationen rund um die Installation der QLIB sehen Sie im "QLIB" Handbuch nach, welches auf der CD enthalten ist.
- Wenn Sie eine QUANCOM ISA Karte benutzen überprüfen Sie, ob die Treiber korrekt installiert sind.



Warum gibt QAPIExtOpenCard (...) den Wert 0 zurück, obwohl die Karte installiert ist ?

- Überprüfen Sie, ob die QLIB korrekt installiert ist. Für weitere Informationen rund um die Installation der QLIB sehen Sie im “QLIB” Handbuch nach, welches auf der CD enthalten ist.
- Die Karte ist nicht richtig konfiguriert. (“Systemsteuerung” => QLIB, Kapitel “QLIB” Handbuch auf der Installations CD)



Warum bekomme ich die Meldung "Driver QLIBNDRV.SYS" oder "Driver QLIBNDRV.VXD" konnte nicht geladen werden ?

- Überprüfen Sie, ob die QLIB korrekt installiert ist. Für weitere Informationen rund um die Installation der QLIB sehen Sie im “QLIB” Handbuch nach, welches auf der CD enthalten ist.
- Die Treiber für die QUANCOM Karte ist nicht geladen. (Systemsteuerung => System)



Kann die QLIB nur mit Administratorenrechte installiert werden ?

- Ja, installieren Sie die QLIB nur mit Administratorenrechten.



Warum bekomme ich die Meldung "Treiber konnte nicht geladen werden" während der Installation?

- Die Installation wurde ohne Administratorenrechte ausgeführt.



Warum bekomme ich die Meldung "Treiber QLIBNDRV.SYS konnte nicht geladen werden"?

- Installation der Treiber ist fehlgeschlagen, weil die QLIB ohne Administratorenrechte installiert wurde.
- Die QLIB-Software ist auf einem Netzlaufwerk installiert worden. Installieren Sie die QLIB immer auf der lokalen Festplatte.



Warum muss ich den Treiber nach jedem Neustart wieder starten ?

- Die Startart des Treibers steht auf "*Manuell*". Wenn Sie die Einstellungen ändern möchten wählen Sie die Startart "*Automatisch*" und starten Sie Ihr System neu.



Wie kann ich den QLIBNDRV.SYS Treiber manuell installieren ?

- Wenn die QLIB Installation fehlschlägt kann es nötig sein, dass Sie den Treiber manuell installieren.
- Suchen Sie auf der CD in dem Verzeichnis "Tools" die Datei **instdrv.exe**. Mit diesem Tool können Sie den Treiber manuell installieren.
- Führen Sie das Programm folgendermaßen aus :
instdrv qlibndrv d:\directory\qlibndrv.sys . Wechseln Sie dazu das Verzeichnis, in der sich die Datei qlibndrv.sys befindet.
- Wechseln Sie unter Systemsteuerung -> Dienste die Startart auf "*Automatisch*" (für den Dienst QLIBNDRV). Starten Sie Ihren Computer neu.

5.2 Kunden Support und Hilfe



Sie benötigen Hilfe?

Wenn Sie nicht wissen was Sie während einer Installation tun müssen, oder wie die Karte in Betrieb genommen wird, lesen Sie bitte dieses Handbuch.

! Tip !

Im Kapitel "Frequently asked questions" (Häufig gestellte Fragen) sind einige Antworten auf häufig gestellte Fragen. Sie können Ihnen bei der Problemlösung behilflich sein. Auf der QUANCOM Installations CD finden Sie in Textform die Datei README.TXT, welche alle wichtigen Änderungen beinhaltet.

! Wichtig !

Wenn Sie weitere Fragen haben, kontaktieren Sie unser Support-Team. Für diesen Fall halten Sie bitte folgende Informationen bereit:

- Genauer Karten-Typ
- Version der Treiber
- Version der QLIB
- Betriebssystem, Hardware-Ausstattung und

Bussystem

- Name und Version von dem Programm, welches den Fehler ausgibt
- Eine genaue Fehlerbeschreibung (versuchen Sie den Fehler zu wiederholen, um diesen besser beschreiben zu können)

Wen kann ich erreichen?

Die QUANCOM Internet Webseite
www.quancom.de

Per Fax
+49 22 36 / 89 92 - 49

Per E-Mail:
support@quancom.de

Adresse:
QUANCOM INFORMATIONSSYSTEME GmbH
In der Flecht 14
50389 Wesseling

Wenn Sie Hilfe brauchen, erreichen Sie uns unter:
QUANCOM Hotline Deutschland
0 22 36 / 89 92 - 20

Montags - Donnerstag
von 9:00 bis 18:00
Freitags
von 9:00 bis 17:00

Aktuelle Treiber

Auf unserer Internetseite <http://www.quancom.de> können sie immer die neusten Treiber Versionen und Updates finden. Zudem finden Sie ebenfalls viele andere Informationen und die "Frequently asked questions (FAQ's)". Bevor Sie uns kontaktieren, überprüfen Sie ob die neueste Version der QUANCOM Software installiert ist.

Reparatur

Wenn Sie nicht genau wissen, ob die QUANCOM Karte defekt ist, rufen Sie unsere QUANCOM Hotline an:

Tel.: **+49 22 36 / 89 92 – 20**

Bevor Sie uns die Karte zur Reparatur schicken, rufen Sie unsere Hotline an:

Tel.: **+49 22 36 / 89 92 – 20**

Wenn Sie uns die Karte zurückschicken, legen Sie diese bitte in die Originalverpackung oder eine adäquate Verpackung, um einen Transportschaden zu verhindern. Zusätzlich bitten wir Sie, uns eine Kopie der Originalrechnung mitzuschicken.

5.3 Technisches Support Formular

Wenn Sie einen Internetzugang haben, öffnen Sie folgende URL in Ihrem Browser:

<http://www.quancom.de/quancom/qshop.nsf/techniksupport?OpenForm&deu>

Füllen Sie das Formular komplett aus bevor Sie sich an QUANCOM Informationssysteme GmbH wenden. Wenn Sie andere QUANCOM Hardware oder Software nutzen, fügen Sie das bitte dem Formular hinzu.

Name: _____

Firma: _____

Adresse: _____

Telefon: _____

Fax: _____

Computer / Prozessor: _____

Betriebssystem: _____

Grafikkarte: _____

Maus: _____

QUANCOM Karte _____

Andere installierte Karten: _____

Festplatte (Kapazität, frei): _____

Das Problem ist: _____

Auflistung der Fehlermeldung: _____

Folgende Schritte führen zur Wiederholung des Problems: _____

5.4 Hardware und Software Konfigurationsformular

Dieses Formular hilft Ihnen die Einstellungen der Hardware und Software aufzulisten. Füllen Sie das Formular komplett aus bevor Sie sich an QUANCOM Informationssysteme GmbH wenden und nutzen Sie das Formular ebenfalls um die aktuelle Konfiguration nachzuschlagen.

• QUANCOM Produkt:

Name / Name der Karte _____

Interrupt Level _____

DMA Kanal _____

Basis I/O Adresse _____

Betriebssystem _____

• Andere Informationen

Computer Model _____

Prozessor _____

Taktfrequenz _____

Grafikkarte _____

Betriebssystem _____
Programmiersprache _____
Programmiersprachen-Version _____

• **Andere Karten im System**

Basis I/O-Adresse anderer Karten _____
DMA Kanäle anderer Karten _____
Interrupt Level anderer Karten _____

5.5 Dokumentations Formular

QUANCOM Informationssysteme GmbH möchte Ihren Kommentar zum Produkt und zu der über diese Dokumentation oder eines Produktes. Diese Informationen helfen uns unsere Qualität zu verbessern.

Titel: PCI-DECODER2 / PCI-DECODER3

Erstellungsdatum: 14.10.2010

Nehmen Sie Stellung zur Kompetenz, Übersichtlichkeit und Inhalt dieses Handbuchs. Wenn Sie Fehler im Handbuch entdecken notieren Sie sich bitte die Seitenzahl.

Vielen Dank für Ihre Hilfe.

Name: _____

Firma: _____

Adresse: _____

Telefon: _____

Fax: _____

Kommentar: _____

Email an: support@quancom.de

Fax an: +49 2236 89 92 49

Adresse: **QUANCOM Informationssysteme GmbH**
In der Flecht 14
50389 Wesseling

5.6 Warenzeichen

Linux ist ein eingetragenes Warenzeichen von Linus Torvalds.

MS, MS-DOS, Microsoft, Visual Basic, Windows, Windows Vista/XP/2000/NT/ME/98/95 sind eingetragene Warenzeichen von Microsoft Corporation.

XT und PS/2 sind Warenzeichen und IBM, OS/2 und AT sind eingetragene Warenzeichen der International Business Machines Corporation.

Intel, Pentium ist ein eingetragenes Warenzeichen von Intel Corporation.

USB ist ein eingetragenes Warenzeichen von USB Implementers Forum Inc.

JAVA ist ein eingetragenes Warenzeichen von Sun Microsystems.

DELPHI und Pascal sind ein eingetragene Warenzeichen von Borland Corporation.

PCI ist ein eingetragenes Warenzeichen von PCI Special Interest Group.

PCI Express ist ein eingetragenes Warenzeichen der PCI-SIG.

Nationalinstruments, LABVIEW ist ein eingetragenes Warenzeichen von Nationalinstruments Corporation.

Agilent VEE ist ein eingetragenes Warenzeichen von Agilent Technologies.

Ethernet ist ein eingetragenes Warenzeichen der Xerox Corporation.

Bei anderen Produkt- und Firmennamen, die in dieser Anleitung erwähnt werden, könnte es sich um Marken ihrer jeweiligen Eigentümer handeln.