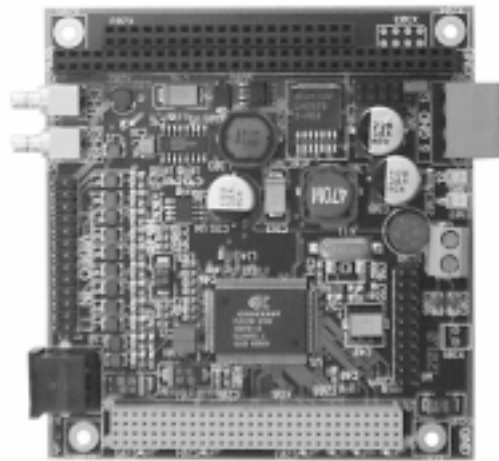


eGrabber-4*plus*



Hardware-Manual

Ausgabe Juni 2004

Im Buch verwendete Bezeichnungen für Erzeugnisse, die zugleich ein eingetragenes Warenzeichen darstellen, wurden nicht besonders gekennzeichnet. Das Fehlen der © Markierung ist demzufolge nicht gleichbedeutend mit der Tatsache, daß die Bezeichnung als freier Warenname gilt. Ebenso wenig kann anhand der verwendeten Bezeichnung auf eventuell vorliegende Patente oder einen Gebrauchsmusterschutz geschlossen werden.

Die Informationen in diesem Handbuch wurden sorgfältig überprüft und können als zutreffend angenommen werden. Dennoch sei ausdrücklich darauf verwiesen, daß die Firma PHYTEC Meßtechnik GmbH weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgeschäden übernimmt, die auf den Gebrauch oder den Inhalt dieses Handbuches zurückzuführen sind. Die in diesem Handbuch enthaltenen Angaben können ohne vorherige Ankündigung geändert werden. Die Firma PHYTEC Meßtechnik GmbH geht damit keinerlei Verpflichtungen ein.

Ferner sei ausdrücklich darauf verwiesen, daß PHYTEC Meßtechnik GmbH weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgeschäden übernimmt, die auf falschen Gebrauch oder falschen Einsatz der Hard- bzw. Software zurückzuführen sind. Ebenso können ohne vorherige Ankündigung Layout oder Design der Hardware geändert werden. PHYTEC Meßtechnik GmbH geht damit keinerlei Verpflichtungen ein.

© Copyright 2004 PHYTEC Meßtechnik GmbH, D-55129 Mainz.

Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form ohne schriftliche Genehmigung der Firma PHYTEC Meßtechnik GmbH unter Einsatz entsprechender Systeme reproduziert, verarbeitet, vervielfältigt oder verbreitet werden.

Informieren Sie sich:

	EUROPA	NORD AMERIKA
Adresse:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 203 Parfitt Way SW, Suite G100 Bainbridge Island, WA 98110 USA
Angebots Hotline:	+49 (800) 0749832 order@phytec.de	+1 (800) 278-9913 sales@phytec.com
Technische Hotline:	+49 (6131) 9221-31 support@phytec.de	+1 (800) 278-9913 support@phytec.com
Fax:	+49 (6131) 9221-33	+1 (206) 780-9135
Web Seite:	http://www.phytec.de	http://www.phytec.com

4. Auflage Juni 2004

1	Lieferumfang / Technische Daten	3
1.1	Zubehör.....	4
1.2	Technische Daten.....	5
1.3	Anwendungsgebiete und Sicherheitshinweise	8
1.4	Adressen und Ressourcen.....	10
1.5	Anschlußbelegung der Buchsen	11
1.5.1	Composite-Eingänge	12
1.5.2	S-Video-Anschluß	15
1.6	I/O-Pin	16
1.7	I ² C-Schnittstelle.....	18
1.8	Der Erweiterungsanschluß Optionport.....	19
1.9	Spannungsversorgungsanschluss.....	20
1.10	Integrierte Spannungsversorgung (nur Modelle - X2)	24
1.10.1	Anschluß der Eingangsspannung	25
1.10.2	Ausgang Spannungsversorgung	26
1.10.3	Optionen Spannungsversorgung.....	28
1.11	Device Selector.....	30
1.12	Hinweise zur CE-Konformität und Störsicherheit	31
2	Einbau und Installation der Grabberkarte	33
2.1	Einsetzen der Steckkarte.....	33
2.2	Spannungsversorgung.....	36
2.3	Installieren der Treiber	38
2.3.1	Zusätzliche Treiber (optional)	42
2.4	Installation des Demoprogrammes	43
3	Das Anschließen der Videoquellen	45
3.1	Anschlußmöglichkeiten von Videoquellen	46
3.1.1	Das S-Video Kabel.....	46
3.1.2	Die Composite-Anschlüsse	47
4	Inbetriebnahme des Grabbers mit Demoprogramm	49
4.1	Beschreibung des Demoprogramms.....	54
4.2	Bildkontrolle.....	60
4.3	Weitere Funktionen unter <i>Image</i>	61
4.4	Fadenkreuze einblenden	62
4.5	Grundeinstellungen.....	62
4.6	Die Sonderfunktionen.....	64
4.7	Bilder speichern, Programm beenden.....	70

5	Treiber-Software	75
5.1	Technische Grundlagen.....	76
5.1.1	Aufbau des eGrabber-4plus.....	76
5.1.2	Videosignal und Digitalisierungsvorgang.....	78
5.1.3	Farbübertragung und Farbspeicherung	81
5.1.4	Datenspeicherung durch DMA und RISC-Programm ..	83
5.2	Treiber für Microsoft Windows	88
5.2.1	Voraussetzungen	89
5.2.2	Einrichten des VxD-Treibers für Windows'95™	90
5.2.3	Einrichten des Gerätetreibers für Windows NT4.0.....	93
5.2.4	Einrichten des Gerätetreibers für Windows'98™ und Windows 2000.....	96
5.2.5	Anwendung der DLL	97
5.2.6	Anwenden der Windows'95/98™ / Windows NT4.0™ / Windows 2000™ DLL	98
5.2.7	Programmierung unter Delphi.....	99
5.2.8	Beschreibung der in der DLL vorhandenen Funktionen.....	101
5.3	Treiber für DOS-Anwendungen.....	156
5.3.1	Voraussetzungen	156
5.3.2	Entwicklungsplattform.....	157
5.3.3	Funktionen des DOS-Treibers <i>PCI4GRAB</i>	158
5.3.4	Programmbeispiel DOS.....	177
6	Änderungen zum eGrabber-2 und Kompatibilität	181
7	Störungssuche	185
Index	193

Bild- und Tabellenverzeichnis

Bild 1:	Zubehörkabel (links SMB-BNC, rechts S-Video).....	4
Bild 2:	Anschlüsse des eGrabber-4 <i>plus</i>	11
Bild 3:	Typische Beschaltungen des I/O-Pins als Eingang	17
Bild 4:	Typische Ausgangsbeschaltung des I/O-Ports	17
Bild 5:	Belegung des Erweiterungssteckers X6	19
Bild 6:	Jumperstellung JP301	21
Bild 7:	Jumperstellung JP301	21
Bild 8:	Jumperstellung JP301	22
Bild 9:	Jumperstellung JP301	22
Bild 10:	Jumperstellung JP301	23
Bild 11:	Belegung des Versorgungsspannungseingangs X300	25
Bild 12:	Belegung des Spannungsversorgungsausgangs X302.....	26
Bild 13:	JP301: 5 V auf PC/104-Busstecker	28
Bild 14:	JP301: 12 V auf PC/104-Busstecker	28
Bild 15:	Anschlußfeld X301	29
Bild 16:	Jumperstellungen des Device-Selectors JP201	30
Bild 17:	Jumperstellungen JP201 (Slotauswahl).....	34
Bild 18:	Anschlußbelegung Versorgungsspannungseingang	36
Bild 19:	Spannungsversorgungs-Ausgang.....	37
Bild 20:	PHYTEC Installationsmenü	43
Bild 21:	Übersicht der eGrabber4- <i>plus</i> Anschlüsse	45
Bild 22:	Video-Abschlußkabel – Legende (Bezeichnung, PHYTEC Best.Nr.).....	46
Bild 23:	Oberfläche des Demoprogramms	49
Bild 24:	Menüauswahl Image.....	50
Bild 25:	Oberfläche zum Einrichten der Bildausgabe	51
Bild 26:	Livebild der Videoquelle	52
Bild 27:	Menü „Image Settings“	54

Bild 28:	Entstehung des Vollbildes aus zwei Halbbildern mit je 7 Zeilen.....	57
Bild 29:	Kammeffekt bei schnell bewegten Objekten	58
Bild 30:	Fenster zur Bildkontrolle.....	60
Bild 31:	Basic Settings Menü.....	62
Bild 32:	Histogramm.....	64
Bild 33:	Color Meter	65
Bild 34:	Arithmetics Menü.....	66
Bild 35:	Normierungswert-Wahl.....	67
Bild 36:	Anzahl der Bilder	68
Bild 37:	I/O-Test Menü	69
Bild 38:	Blockschaltbild eGrabber4plus	76
Bild 39:	Zeilensprungverfahren (Beispiel mit 9 Zeilen)	78
Bild 40:	Halb- und Vollbilder	79
Bild 41:	Kammeffekt bei bewegten Objekten im Vollbild-Modus.....	80
Bild 42:	Übersicht über Pixel und Kontrolldatenfluß	85
Bild 43:	Verzeichnisstruktur der Windows Treiber	88
Bild 44:	Windows'95 Registrierungs-Editor	91
Bild 45:	Hinzufügen eines VxD-Eintrages.....	91
Bild 46:	Konfigurieren des VxD	92
Bild 47:	Windows NT Registrierungs-Editor.....	94
Bild 48:	Hinzufügen eines Gerätetreiber-Eintrages	94
Bild 49:	Konfigurieren des Treibers.....	95
Bild 50:	Skalierung und Ausschnittsbildung.....	123
Bild 51:	Bild zur Skalierung: alle Werte gleich bis auf ppl	124
Bild 52:	Farbformate des eGrabber-4plus	130
Bild 53:	Rückgabewert der Funktion data_present.....	136
Bild 54:	Timing Diagramm der Rückgabe-Parameters von data_present()	137

Tabelle 1: Modellvarianten eGrabber-4plus	3
Tabelle 2: Belegung der SMB-Buchsen u. Pfostenleiste, Modell EPC-032, EPC-032-X2	13
Tabelle 3: Belegung der SMB-Buchsen u. Pfostenleiste, Modelle EPC-032-X1, EPC-032-X1-X2	14
Tabelle 4: Beschaltung der S Video-Eingangs an der Pfortenleiste X 3	15
Tabelle 5: Beschaltung des I/O-Pins an den Pfostenleisten X3 und X6.....	16
Tabelle 6: Beschaltung der I ² C-Schnittstelle an den Pfostenleisten X3, X6	18
Tabelle 7: Belegung des Option-Port – Verbinders (alle Modelle).....	19
Tabelle 8: Spannungsversorgungsausgang	20
Tabelle 9: Optionen für den Versorgungsspannungsausgang (X3, Pin 25)	20
Tabelle 10: zulässige Eingangsspannungen EPC-032-X2	25
Tabelle 11: Belegung der Buchse X303.....	27
Tabelle 12: Speicherbedarf eines Pixel in den einzelnen Modi	132
Tabelle 13: Speicherbedarf eines Pixel in den einzelnen Modi	166
Tabelle 14: Belegung des Option-Port – Verbinders (alle Modelle).....	183

Teil 1

Installation und Inbetriebnahme

1 Lieferumfang / Technische Daten

Der eGrabber-4*plus* ist eine Framegrabberkarte für die Verwendung mit CPU-Karten des PC/104plus-Industriestandards.

Die Karte ist in verschiedenen Ausführungen erhältlich:

Best.-Nr.	Composite-Eingänge	S-Video-Eingang	integrierte Spannungsversorgung
EPC-032	3	1	nein
EPC-032-X1	9	1	nein
EPC-032-X2	3	1	ja
EPC-032-X1-X2	9	1	ja

Tabelle 1: Modellvarianten eGrabber-4*plus*

Bitte prüfen Sie anhand der Lieferunterlagen, welches Modell Sie besitzen, um Ihre Framegrabber-Karte optimal einsetzen zu können.

Zum Lieferumfang des eGrabber-4*plus* gehören:

- eGrabber-4*plus* Board im PC/104plus Format
- Installations-CD mit:
 - Demosoftware
(Windows '95/98/ME/XP, NT4.0 und Windows 2000)
 - Treiber-Bibliothek für DOS (mit DOS4GW)
 - Treiber- Software für Windows'95, 98, ME, XP, NT4.0 und Windows 2000
 - Twain-Treiber für Applikationen mit Twain-Schnittstelle
 - Labview-Treiber für Bildverarbeitungsanwendungen unter Labview (National Instruments; IMAQ – Paket erforderlich)
- dieses Manual

1.1 Zubehör

Bei PHYTEC können Sie folgendes Zubehör zum *eGrabber-4plus* erhalten:

- SMB auf BNC Anschlußkabel für den Anschluß von Farbkameras mit BNC-Buchse: Länge ca. 2 m – Bestell-Nr. WK-023
- S-Video- Anschlußkabel für den Anschluß von Farbkameras mit 4poliger Mini-DIN-Buchse (S-Video-Ausgang): Länge ca. 2 m – Bestell-Nr. WK051
- Ersatzsicherung 2A T TR5 für Spannungsversorgung (F1) – Best.Nr. KF012



Bild 1: Zubehörkabel

1.2 Technische Daten

Abmessungen: PC/104plus Format

Grabberteil:

Datenbus: PCI-Bus 5 V, Master-Steckposition erforderlich
(PCI Rev. 2.1 compliant)
(PC/104plus Version 1.0 compliant)

Versorgung: +5 V (250 mA Idle, 300 mA Digitizing)
(werden dem PCI-Bus entnommen)

Eingänge: 3 od. 9 (je nach Modell, *siehe Kapitel 1.5.1*)
Composite-Videoeingänge, 75 Ω , 1 V_{ss} ¹⁾
1 S-Video-Eingang 75 Ω (0,7 V_{ss} / 0,3 V_{ss})

Videoformate: PAL (B,G,H,I), NTSC (M)
bzw. entsprechende CCIR-Formate monochrom

Synchronisation: Composite-Sync bzw. Sync auf Y-Signal
externe Synchronisation nicht möglich

Datenformate: 16 Mio. Farben: RGB32, RGB24, YCrCb 4:2:2,
YCrCb 4:1:1
64.000 Farben: RGB16
32.000 Farben: RGB15
256 Graustufen: Y8 Gray Scale

Bildauflösung: maximal 752 x 582 Pixel (PAL)
bzw. 640 x 480 Pixel (NTSC)
Auflösung frei skalierbar in X- und Y-Richtung
bis 14:1

Bildeinzug: Halbbild: 20 ms
Vollbild: 40 ms
Bildtransfer in den Arbeitsspeicher in Echtzeit
(Busmaster-Transfer)

verwendete

Ressourcen: 4 kByte Arbeitsspeicher (Registerbereich) /INTA

¹⁾: wenn der S-Video-Eingang nicht benutzt wird, steht ein weiterer Composite-Eingang zur Verfügung

Bildkorrektur: Gammakorrektur (zuschaltbar)
Helligkeit (+/-50 %)
Kontrast (0 % ... 235 %)
Farbsättigung (U: 0...201 %, V: 0...283 %)
Farbton (+/-90°, nur NTSC)

Bildspeicher: 630 Byte FIFO on-board,
Echtzeit-Speicherung im PC-Arbeitsspeicher
Even-/Odd-Speicher getrennt oder
gemeinsamer Vollbildspeicher (wählbar)

Ports: 12-bit Parallel-I/O, TTL-Pegel (Multipurpose)

Parameter	Symbol	Min	Max
Input High Voltage	V _{IH}	2,0 V	5 V
Input Low Voltage	V _{IL}	-0,5 V	0,8 V
Output High Voltage	V _{OH}	2,4 V	-
Output Low Voltage	V _{OL}	-	0,4 V
Input Low Current	I _{IL}	-	-70 µA
Input High Current	I _{IH}	-	70 µA

1 I/O-Anschluß (transistorgetrieben, 28 V/0,8 A_{max})

Parameter	Symbol	Min	Max
Input High Voltage	V _{IH}	2,0 V	28 V
Input Low Voltage	V _{IL}	-0,5 V	0,5 V
Output High Voltage	V _{OH}	5 V	25 V
Output Low Voltage	V _{OL}	0 V	1,4 V
Input Low Current	I _{IL}	-	-700 µA
Input High Current	I _{IH}	-	70 µA
Output HiZ Current	I _{OZ}	-	500 µA
Output On Current	I _{OON}	-	800 mA
Schaltfrequenz	f _{IO}		200 Hz

1 I²C-Schnittstelle (Master)

Parameter	Symbol	Min	Max
Übertragungsrate ¹	f _{I2C}	99,2 kHz	396,8 kHz
Input High Voltage	V _{IH}	3,5 V	5 V
Input Low Voltage	V _{IL}	-0,5 V	1,5 V
Hysteresis	V _{hys}	0,2 V	
Input High Current	I _{IH}	-	10 µA
Input Low Current	I _{IL}	-	-10 µA
Output Low Voltage	V _{OL}	-	0,4 V

EEPROM: 256 Bytes internes EEPROM
Schreibzyklen: mehr als 10.000

¹ : die beiden Frequenzen sind softwaremäßig umschaltbar

Spannungsversorgung:

(nur Modelle EPC-032-X2 / EPC-032-X1-X2)

Eingangsspannung: 8 ... 28 VDC

Ausgangsspannungen: +5 VDC / 2 A
+12 VDV / 1A
bei Verwendung des +12 V-Ausgangs muß die Eingangsspannung im Bereich 15...28 V liegen

Ausgangsleistung: 22 W max.

Abschaltung: elektronisch, TTL-Signal

Anschlüsse:

2 SMB-Buchsen:

- X1 = Composite Channel 1
- X2 = Composite Channel 2

26poliger Pfostenstecker X3:

- Composite Kanal 1-3
(EPC-032-X1: Kanal 1-9)
- S-Video Chroma Signal
(Luma auf Composite 9)
- 1 I²C-Schnittstelle
- 1 Spannungsversorgungsausgang 1,5 A_{max}
- 1 I/O-Anschluß (transistorgetrieben)

Mini-DIN-Buchse X4:

- 1 S-Video-Eingang

Pfostenleiste 2x10 X6:

- GPIO-Port, 12 x TTL I/O
- 1 I²C-Schnittstelle
- 1 I/O-Anschluß, transistorgetrieben

Spannungsversorgungsbuchse X300:

- PC-Floppy-Steckverbinder (3,5“) zur Einspeisung einer Kamera-Versorgungsspannung. Die Spannung wird an X3 durchgeschleift (nicht Modelle -X2, siehe nächste Seite)

Modelle EPC-032-X2 und EPC-032-X1-X2 zusätzlich:

Spannungsversorgungsbuchse X300:

- PC-Floppy-Steckverbinder(3,5“) zur Einspeisung der Versorgungsspannung (8 – 28 VDC, bei Verwendung des 12 V Ausgangs 15 – 28 V)

3pol. steckbare Schraubklemme X302:

- Spannungsversorgungsausgang +5 V/2 A, +12 V/1 A

8pol. Pfostenleiste X303 (optional):

- Spannungsversorgung für Jumptec-MOPS CPU-Karte (aufsteckbar)

Steckverbinder X301 (optional):

- Shutdown-Anschluß zur Abschaltung der Spannungsversorgung

1.3 Anwendungsgebiete und Sicherheitshinweise

Achten Sie beim Einsatz des eGrabber-4plus auf die Einhaltung der spezifizierten Betriebsbedingungen. Lesen Sie vor der Inbetriebnahme diese Anleitung sorgfältig.

- Der eGrabber-4plus dient zur Digitalisierung von Videosignalen von Standard-TV-Kameras. Es können Signale von Composite-Videokameras verarbeitet werden, die den Normen CCIR B,G,H,I und Unternorm CCIR B,G,H,I/PAL entsprechen. Zusätzlich können Signale nach CCIR M/NTSC eingespeist werden. Die Kamerasignale können auch nach Luma- und Chromaanteil getrennt gemäß der S-Video-Norm zugeführt werden (nur Kanal 9 / S-Video-Anschluß).

Die Digitalisierung der Bilder erfolgt in Echtzeit. Die Übertragung der Bilddaten erfolgt über den PCI-Bus. Die Übertragungsgeschwindigkeit entspricht dabei der für den Master-PCI-Buszugriff des verwendeten PCs spezifizierten Zugriffszeit.

Die effektive Transferrate muß ausreichend für die angeforderte Bilddatenmenge sein, ansonsten ist der Verlust von Bilddaten möglich.

- Der eGrabber-4*plus* ist für die Verwendung in PC/104plus - Systemen bestimmt. Der eGrabber-4*plus* muß auf einen masterfähigen PCI-Slot konfiguriert werden. Bei der Zusammenstellung und dem Gehäuseeinbau des PC/104-plus-Systems müssen die geltenden CE-Normen beachtet werden. Das Gerät ist für den Einsatz in trockener und sauberer Betriebsumgebung konzipiert. Bei Verwendung in Maschinen, industrieller Umgebung und Geräten ist zu prüfen, ob zusätzliche Maßnahmen zur Einhaltung von Sicherheits- und Funkstörnormen sowie zur Aufrechterhaltung der Betriebssicherheit erforderlich sind.
- Die Verwendung des Geräts in sicherheitsrelevanten Bereichen, in der Luft-/Raumfahrt und in kerntechnischer oder militärischer Anwendung, bedarf unserer Prüfung und Zustimmung.
- Bei gewerblicher Anwendung sind die Unfallverhütungsvorschriften des Verbands der gewerblichen Berufsgenossenschaften für elektrische Anlagen und Betriebsmittel zu beachten.
- Vor der Inbetriebnahme ist generell zu prüfen, ob das Gerät für den vorgesehenen Anwendungsfall und Einsatzort geeignet ist. Im Zweifelsfall sind unbedingt Rückfragen bei Fachleuten, Sachverständigen oder dem Hersteller erforderlich.
- Das Produkt ist vor starken Erschütterungen und Vibrationen zu schützen. Erforderlichenfalls ist eine Federung oder Polsterung vorzusehen, die jedoch nicht die Belüftung des Geräts behindern darf.

Eine eventuell notwendige Reparatur darf nur vom Fachmann unter Verwendung von Originalbauteilen durchgeführt werden. Beim Anschluß des Geräts, nur zugelassene und geprüfte Anschlußkabel verwenden. Es muß auf korrekte Abschirmung und Entstörung der Kabel geachtet werden.

1.4 Adressen und Ressourcen

Der eGrabber-4plus belegt einen Speicherbereich von 4 kByte im Arbeitsspeicher des PCs für die lokalen Register. Der Adressbereich wird durch das PCI-BIOS des Rechners automatisch festgelegt.

Lediglich die Position der Karte im PC/104plus System muß mit Hilfe von Jumper JP201 eingestellt werden (*siehe Kapitel 2.1*).

Es können bis zu drei eGrabber-4plus in einem System installiert werden. Die Karten werden dann vom BIOS automatisch auf verschiedene Adressen konfiguriert.

Es kann nicht vorab bestimmt werden, welche Karte auf welche Adresse konfiguriert wird. Die Basisadresse jeder Karte kann über das PCI-BIOS ermittelt werden. Beim eGrabber-4plus ermittelt die Treibersoftware über das BIOS die Adresse und setzt sie in eine „Gerätenummer“ um. Es kann über den Treiber ermittelt werden, wieviel Karten sich im System befinden und dann jede Karte über ihre individuelle Gerätenummer angesprochen werden.

Hinweis:

Es kann nicht sicher vorhergesagt werden, welche Karte mit welcher Gerätenummer verbunden wird. Dies wird allein vom PCI-BIOS und der Architektur des CPU-Boards bestimmt. In der Regel werden die Adressen aufsteigend in der Reihenfolge der Nummerierung der PCI-Slots vergeben. Dies kann jedoch bei manchen Herstellern abweichen.

Der eGrabber-4plus kann bei einer ganzen Reihe von Ereignissen und Betriebszuständen einen Interrupt auslösen. Da es sich bei ihm um ein *single function device* handelt, kann er nur die Interruptleitung /INTA des PCI-Busses belegen. Diesem PCI-Bus-Interrupt ist über das BIOS ein Interrupt des PCs zugeordnet, über den programmtechnisch auf das Ereignis reagiert werden kann.

Die Ursache des Interrupts kann über das Interrupt-Statusregister des Grabbers ermittelt werden.

Achtung!

Da mehrere Karten den gleichen Interrupt /INTA benutzen, muß zunächst geprüft werden, welche Karte den /INTA ausgelöst hat.

- Beim Einschalten des Rechners wird das PCI-BIOS die neue Karte erkennen. Dies wird i.a. durch eine entsprechende Meldung angezeigt.
- Zur Funktionskontrolle empfehlen wir die Demosoftware auf der mitgelieferten CD.

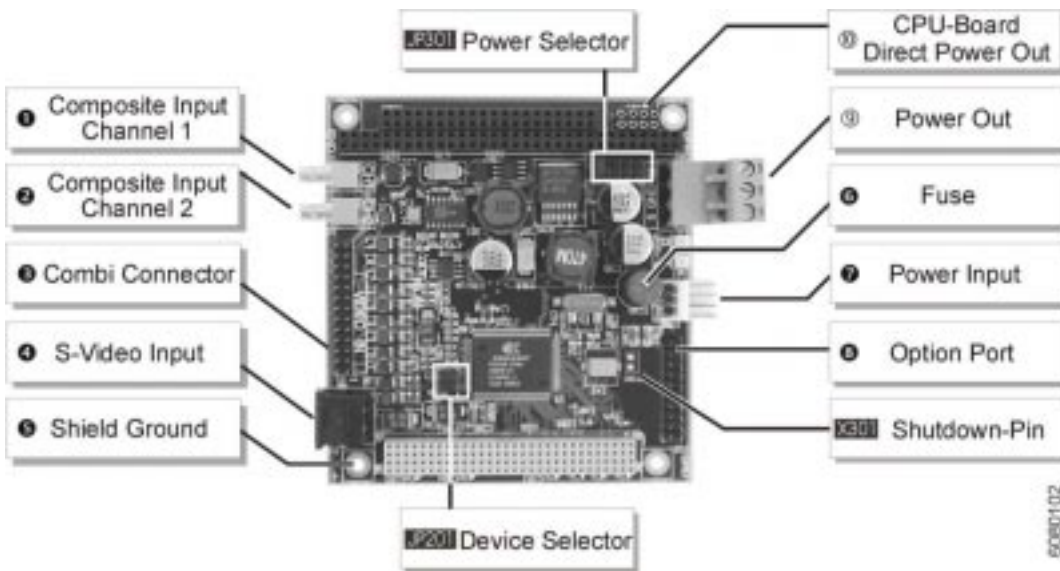
1.5 Anschlußbelegung der Buchsen

Hinweis:

Die nachfolgende Beschreibung der Anschlüsse des Grabbers ist als technische Referenz zu verstehen.

Mehr über die zur Inbetriebnahme erforderlichen Verbindungen und Anschlußkabel erfahren Sie in der detaillierten Anleitung in *Kapitel 2*.

Zur ersten Inbetriebnahme ist die Lektüre dieses *Kapitels 1.5* zunächst nicht erforderlich



ⓐ = bei allen Modellen vorhanden ⓑ = nur Modelle X2

Bild 2: Anschlüsse des eGrabber-4plus

1.5.1 Composite-Eingänge

Als Videoquellen eignen sich alle Composite-Quellen mit einem Signalhub von 1 V_{ss} und einem Abschlußwiderstand von 75 Ω. Die unterstützten Videonormen entnehmen Sie bitte *Kapitel 1.2*.

- **Versionen EPC-032, EPC-032-X2**

Der Grabber besitzt drei Composite-Eingänge. Kanal 1 und 2 sind auf die SMB Buchsen ❶ und ❷ geführt. Alle drei Kanäle sind an der 26pol-Pfostenleiste X3 ❸ herausgeführt.

Der Anschluß der Kanäle 1 und 2 kann wahlweise entweder an den SMB-Buchsen X1 / X2 oder an der Pfostenleiste X3 erfolgen.

Achtung!

Es dürfen nicht beide Anschlußmöglichkeiten gleichzeitig verwendet werden!

- **Versionen EPC-032-X1, EPC-032-X1-X2**

Der Grabber stellt neun Composite-Eingänge zur Verfügung. Kanal 1 und 2 sind auf die beiden SMB Buchsen ❶ und ❷ geführt. Alle neun Kanäle sind an der 26pol-Pfostenleiste X3 ❸ herausgeführt.

Der Anschluß der Kanäle 1 und 2 kann wahlweise entweder an den SMB-Buchsen X1 / X2 oder an der Pfostenleiste X3 erfolgen.

Achtung!

Es dürfen nicht beide Anschlußmöglichkeiten gleichzeitig verwendet werden!

Zusätzlich zu den Composite-Eingängen ist ein Versorgungsspannungspin vorhanden, über den Kameras mit Spannung versorgt werden können, sofern eine externe Spannungsversorgung an die Buchse X300 ❶ angeschlossen ist (*siehe Kapitel 1.9*).

An dem Combi-Verbinder X3 ❸ sind noch weitere nützliche Signale vorhanden, vgl. *Tabelle 2* und *Tabelle 3*.

**eGrabber-4plus mit 3 Composite-Eingängen
(EPC-032, EPC-032-X2)**

SMB-Buchsen		Pfostenleiste X3 ③	
Buchse	Funktion	Pin	Funktion
X1 ①	Composite 1	1	Composite 1
X2 ②	Composite 2	2	Signal Ground
		3	Composite 2
		4	Signal Ground
		5	Composite 3
		6	Signal Ground
		7	
		8	
		9	
		10	
		11	
		12	
		13	
		14	
		15	
		16	
		17	S-Video LUMA
		18	
		19	S-Video CHROMA
		20	Signal Ground
		21	I ² C-Bus: SDA
		22	I ² C-Bus: SCL
		23	Signal Ground
		24	Signal Ground
		25	Pwr. Supply out
		26	I/O-Pin

Tabelle 2: Belegung der SMB-Buchsen u. Pfostenleiste, Modell EPC-032, EPC-032-X2

**eGrabber-4plus mit 9 Composite-Eingängen
(EPC-032-X1, EPC-032-X1-X2)**

SMB-Buchsen		Pfostenleiste X3 ③	
Buchse	Funktion	Pin	Funktion
X1 ①	Composite 1	1	Composite 1
X2 ②	Composite 2	2	Signal Ground
		3	Composite 2
		4	Signal Ground
		5	Composite 3
		6	Signal Ground
		7	Composite 4
		8	Signal Ground
		9	Composite 5
		10	Signal Ground
		11	Composite 6
		12	Signal Ground
		13	Composite 7
		14	Signal Ground
		15	Composite 8
		16	Signal Ground
		17	Composite 9 und S-Video LUMA
		18	Signal Ground
		19	S-Video CHROMA
		20	Signal Ground
		21	I ² C-Bus: SDA
		22	I ² C-Bus: SCL
		23	Signal Ground
		24	Signal Ground
		25	Pwr. Supply out
		26	I/O-Pin

Tabelle 3: Belegung der SMB-Buchsen u. Pfostenleiste, Modelle EPC-032-X1, EPC-032-X1-X2

PHYTEC bietet Ihnen als Zubehör fertig konfektionierte Anschlußkabel an, die die Signaleinspeisung über BNC-Stecker an den SMB-Buchsen ① und ② ermöglichen. An die beiden SMB-Buchsen (Videoeingänge 1 und 2) paßt das Kabel WK023 (siehe Kapitel 1).

1.5.2 S-Video-Anschluß

Der Vorteil des S-Video-Systems ist die getrennte Führung von Helligkeits- und Farbsignal. Dies verhindert Moiréstörungen an feinen Bildstrukturen und verbessert so die tatsächliche Auflösung bei Farbbildern.

Es bestehen zwei Möglichkeiten, eine S-Video-Quelle an den eGrabber-4*plus* anzuschließen:

- An der 4poligen Mini-DIN-Buchse (X4, ④) kann ein *S-Video*-Signal eingespeist werden. Die Buchse ist entsprechend der *S-Video*-Norm beschaltet. Der Anschluß einer Kamera kann mit einem handelsüblichen *S-Video*-Kabel erfolgen.
- An die 26polige Buchsenleiste (X3, ③) kann eine S-Video-Kamera angeschlossen werden. Dazu wird das S-Video Chroma Signal an Pin 19 angeschlossen. Für die Einspeisung des S-Video Luma Signals (Helligkeit) wird einer der Composite-Eingänge verwendet (bei Modellen mit 9 Kanälen ist dies Kanal 9, bei Modellen mit 3 Kanälen der Kanal 3).

Das Signal befindet sich bei allen Modellen auf Pin17 von X3.

Wenn Sie ein entsprechendes Anschlußkabel selbst herstellen möchten, sind also folgende Pins zu beschalten (die Beschaltung der Spannungsversorgung (Pin 25) ist optional, dazu muß Spannung an Buchse X300 anliegen):

26pol Pfostenleiste X3 ③	
Pin	Funktion
17	S-Video: Luma
18	Signal Ground
19	S-Video: Chroma
20	Signal Ground
24	GND
25	Spg. out (Kamera Spg.)

Tabelle 4: Beschaltung der S Video-Eingangs an der Pfortenleiste X 3

Achtung!

Es dürfen nicht beide *S-Video*-Eingänge gleichzeitig beschaltet werden!

Es kann entweder der Mini-DIN-Eingang ④ **oder** der Anschluß an der 26poligen Pfostenleiste X3 ③ verwendet werden.

Im Anwenderprogramm kann die Auswahl der Buchse automatisch erfolgen.

1.6 I/O-Pin

Sowohl an der Pfostenleiste X3 als auch an der Option Port - Pfostenleiste X6 (*siehe auch Kapitel 1.8*) ist ein universell verwendbarer I/O-Pin vorhanden.

Dieser Pin kann sowohl als Eingang als auch als Ausgang verwendet werden. Damit der I/O-Pin genutzt werden kann, muß das Anwenderprogramm diese Funktion unterstützen.

Bei der Funktion als Eingang können dem Anwenderprogramm Steuersignale übermittelt werden. Der Eingang kann vom Programm frei abgefragt werden und ist an keine spezielle Funktion gebunden.

Bei der Funktion als Ausgang kann das Anwenderprogramm ein Steuersignal an ein anderes Gerät übermitteln. Der Ausgang kann vom Programm beliebig gesetzt oder gelöscht werden.

Funktionsweise des I/O-Pins

- **Eingang**

An den I/O-Pin kann eine externe Spannung (gegen Ground) angelegt werden. Ist dieser Spannungswert zwischen 0 V und 0,5 V, bekommt das Programm eine „0“ angezeigt. Beträgt die Spannung 2 V bis 30 V, so ist der Signalpegel logisch „1“. Der positive Anschluß muß dabei an Pin 26 (X3) bzw. an Pin 15 (X6) angeschlossen sein (*vgl. 5*).

Pfostenleiste X3		Pfostenleiste X6	
Pin	Funktion	Pin	Funktion
24	Ground	19	Ground
26	I/O-Pin (+)	15	I/O-Pin (+)

Tabelle 5: Beschaltung des I/O-Pins an den Pfostenleisten X3 und X6

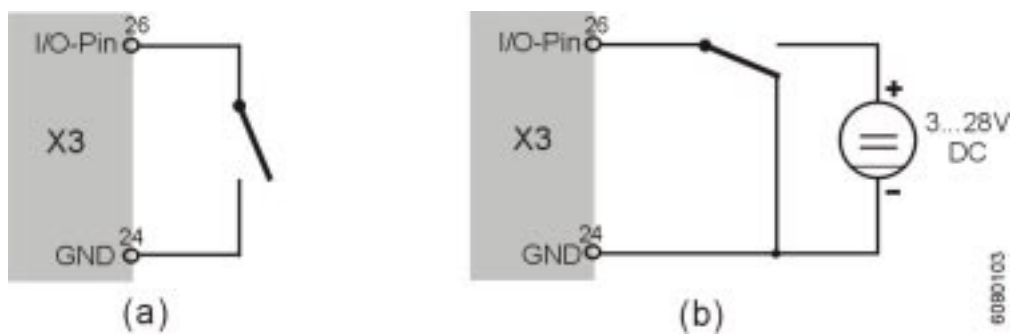


Bild 3: Typische Beschaltungen des I/O-Pins als Eingang

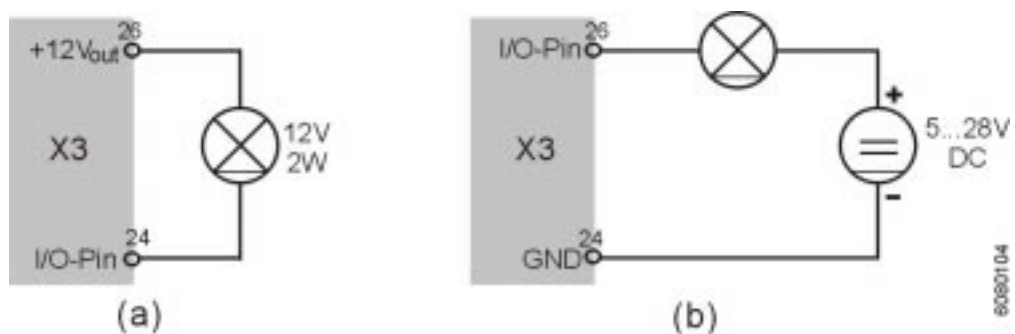
- **Ausgang**

Bei Verwendung als Ausgangs-Pin wird Pin 26 bzw. Pin 15 durch einen Schalt-Transistor mit Ground (z.B. Pin 24 bei X3, Pin 19 bei X6) verbunden, wenn das Programm den Pin auf logisch „1“ setzt.

Setzt das Programm den Pin auf logisch „0“, so ist der Transistor gesperrt und es besteht keine Verbindung zwischen Pin 26 bzw. Pin 15 und Ground.

Um den Ausgang zu benutzen, wird eine externe Versorgungsspannung benötigt. Der mögliche Spannungsbereich beträgt 5 V bis 28 V. Möglicherweise kann der Spannungsversorgungspin (Pin 25 bei X3) zur Spannungsversorgung benutzt werden.

Bild 3 und Bild 4 zeigen zwei typische Beschaltungsvarianten.



- a) Speisung aus Versorgungspin des Grabbers
- b) Speisung aus externer Gleichspannungsquelle

Bild 4: Typische Ausgangsbeschaltung des I/O-Ports

Achtung!

Die Polarität der angelegten Spannung muß so gewählt werden, daß der I/O-Pin 26 (X3) bzw. 15 (X6) stets positives Potential hat.

Negatives Potential an Pin 26 bzw. 15 (in Bezug auf Ground) kann zur Zerstörung der Grabberkarte führen!

Die angelegte Betriebsspannung (I/O-Pin gegen Ground) muß in der Funktion als Ausgang zwischen +5 V und +28 V liegen. In der Eingangs-Funktion darf die maximale Betriebsspannung von +28 V nicht überschritten werden.

Der I/O-Pin ist gegenüber den Video-Leitungen, der CPU und den übrigen Signalleitungen nicht galvanisch getrennt.

1.7 I²C-Schnittstelle

Über die I²C-Schnittstelle können externe Geräte abgefragt oder angesteuert werden, die eine I²C-Schnittstelle im Slave-Modus besitzen.

Die I²C-Schnittstelle steht an der Pfostenleiste X3, und an der Stiftleiste des *Option Ports* X6 (siehe auch Kapitel 1.8) zur Verfügung. Es können mehrere I²C-Geräte an den Bus angeschlossen werden, sie müssen sich jedoch in ihrer Geräteadresse unterscheiden.

Tabelle 6 zeigt die relevante Anschlußbelegung der Pfostenleisten.

Pfostenleiste X3		Pfostenleiste X6	
Pin	Funktion	Pin	Funktion
21	I ² C Bus: SDA	18	I ² C Bus: SDA
22	I ² C Bus: SCL	17	I ² C Bus: SCL
23	Ground	19	Ground

Tabelle 6: Beschaltung der I²C-Schnittstelle an den Pfostenleisten X3, X6

Hinweis:

Die I²C-Schnittstelle wird mit TTL-Pegel betrieben. Dadurch ist die maximale Leitungslänge begrenzt. Bei einem angeschlossenen Gerät können je nach eingestellter Übertragungsrate Leitungslängen von ca. 1 m – 2 m erreicht werden.

Verwenden Sie zum Anschluß nur ausreichend geschirmte Kabel.

Informationen, wie Sie die I²C-Schnittstelle in Ihrer Software ansprechen können, finden Sie im *Kapitel 5.2.8* unter der Funktionsgruppe „Daten über die I²C-Schnittstelle übertragen“.

Hinweis:

Der Geräte-Adressbereich 0xA0 bis 0xA3 wird auf der Grabberkarte intern verwendet. Daher darf ein extern angeschlossenes I²C-Gerät nicht auf diese Geräteadressen konfiguriert werden.

1.8 Der Erweiterungsanschluß Optionport

Der Erweiterungsport stellt dem Anwender 12 digitale I/O-Leitungen und eine I²C-Schnittstelle zur Verfügung. Die Signale sind auf einen Pfostenstecker mit 10 x 2 Pins geführt. Die Leiste ist mit X6 (Ⓢ) bezeichnet. *Bild 5* zeigt die Belegung der Steckleiste.

Beachten Sie, daß der +5V Versorgungspin X6-1 mit max. 100 mA belastet werden darf.

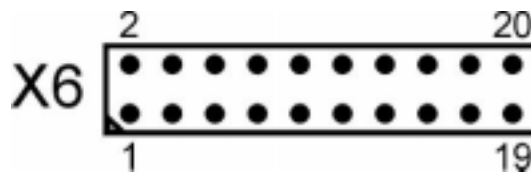


Bild 5: Belegung des Erweiterungssteckers X6

Erweiterungsanschluß (Option Port, X6)					
Pin	Funktion	Pin	Funktion	Pin	Funktion
1	+5 V out	8	I/O 6	15	I/O-Pin
2	I/O0	9	I/O 7	16	I/O Clk
3	I/O1	10	I/O 8	17	I ² C SCL
4	I/O2	11	I/O 9	18	I ² C SDA
5	I/O3	12	I/O 10	19	GND
6	I/O4	13	I/O 11	20	GND
7	I/O5	14	N.C.		

Tabelle 7: Belegung des Option-Port – Verbinders (alle Modelle)

Hinweis:

Der I/O-Pin (X6 – 15) ist identisch mit dem I/O-Pin an der Stiftleiste X3 (X3 – 26). Es darf daher nur einer dieser Pins beschaltet sein. Weitere Informationen über den I/O-Pin finden Sie im *Kapitel 1.6*.

1.9 Spannungsversorgungsanschluss

An der Stiftleiste X3 steht eine Spannungsversorgungsanschluss zur Verfügung, mit dem externe Komponenten wie z.B. eine Kamera mit Spannung versorgt werden können (vgl. Tabelle 8).






Die Stromentnahme darf 1,5 A nicht überschreiten. Die Ausgangsspannung kann nur positiv sein.

Pfostenleiste X3	
Pin	Funktion
24	Ground
25	Versorgungsspannung (+)

Tabelle 8: Spannungsversorgungsanschluss

Mit dem Steckjumper JP301 können Sie zwischen verschiedenen Möglichkeiten der Spannungsversorgung wählen. Beachten Sie, daß manche Optionen nur bei bestimmten Grabber-Typen möglich sind.

Optionen im Überblick:

Versorgung X3-25:	Jumper JP301	Modell	
		EPC-032 und EPC-032-X1	EPC-032-X2 und EPC-032-X1-X2
Eingangsspannung (X300)		●	●
+12 V aus PCI-Bus		●	○
+ 5 V aus PCI-Bus		●	
+12 V aus integr. Netzteil			●
+ 5 V aus integr. Netzteil			●

● = möglich ○ = möglich jedoch nicht empfohlen

Tabelle 9: Optionen für den Versorgungsspannungsanschluss (X3, Pin 25)

Beschreibung der Optionen:

Hinweis:

Pin 1 des Jumperfelds ist durch eine abgeschrägte Ecke auf dem Bestückungsdruck gekennzeichnet.

- Versorgung aus der Eingangsspannung



Bild 6: Jumperstellung JP301

An Pin X3-25 liegt die Eingangsspannung (X300 – Pin 4) an. Die Spannung ist unregelt. In dem Stromkreis befinden sich eine Verpolungsschutzdiode und die Sicherung SI301 (2A).

Achtung!

Die Spannung an Pin 25 entspricht beim Durchschleifen der Spannung des Netzgerätes und dessen gewählter Spannung.

Beachten Sie dies beim Anschluß von Verbrauchern.

In keinem Fall darf die angelegte Spannung an Buchse X300 +28 V überschreiten. Die Stromentnahme darf 1,5 A nicht überschreiten. Der Ausgang ist durch die Sicherung F1 geschützt (Ersatzsicherungen sind bei PHYTEC erhältlich, Best.Nr. KF012).

- Versorgung aus dem PCI-Bus mit +12 V

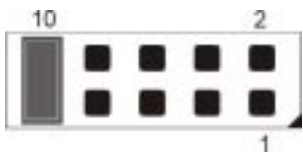


Bild 7: Jumperstellung JP301

An Pin X3-25 liegt die +12 V-Spannung an, die über den PC/104plus- Stecker auf die Grabberkarte geführt wird.

Diese Einstellung wird nicht empfohlen, wenn Sie eine Grabberkarte mit integrierter Spannungsversorgung besitzen (Versionen EPC-032-X2 / EPC-032-X1-X2).

Beachten Sie, daß die Strombelastbarkeit der PCI-Bus-Versorgung begrenzt ist. Sie sollten nicht mehr als ca. 500 mA entnehmen.

Hinweis:

Diese Option funktioniert nur, wenn in das PC/104-System eine Versorgungsspannung von +12 V eingespeist wird.

- Versorgung aus dem PCI-Bus mit +5 V

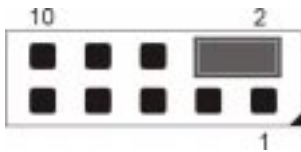


Bild 8: Jumperstellung JP301

An Pin X3-25 liegt die +5 V-Spannung an, die über den PC/104plus-Stecker auf die Grabberkarte geführt wird.

Diese Einstellung ist nicht möglich, wenn Sie eine Grabberkarte mit integrierter Spannungsversorgung besitzen (Versionen EPC-032-X2 / EPC-032-X1-X2).

Beachten Sie, daß die Strombelastbarkeit der PCI-Bus-Versorgung begrenzt ist. Sie sollten nicht mehr als ca. 500 mA entnehmen.

- Versorgung aus integriertem Netzteil mit +12 V
- **nur Versionen EPC-032-X2 / EPC-032-X1-X2**

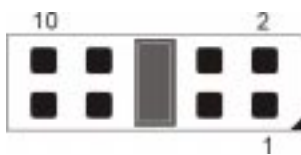


Bild 9: Jumperstellung JP301

- An Pin X3-25 liegt eine geregelte Spannung +12 V aus der internen Spannungsversorgung an.

Hinweis:

Diese Option ist nur verfügbar, wenn Sie eine Grabber-Version mit integrierter Spannungsversorgung besitzen und die Eingangsspannung (an X300) mehr als 15 VDC beträgt. Die maximale Stromentnahme aus dem 12 V-Zweig der integrierten Spannungsversorgung beträgt in der Summe 1 A.

- Versorgung aus integriertem Netzteil mit +5 V
– nur Versionen EPC-032-X2 / EPC-032-X1-X2 –



Bild 10: Jumperstellung JP301

An Pin X3-25 liegt eine geregelte Spannung +5 V aus der internen Spannungsversorgung an.

Hinweis:

Diese Option ist nur verfügbar, wenn Sie eine Grabber-Version mit integrierter Spannungsversorgung besitzen. Die maximale Stromentnahme aus dem 5 V-Zweig der integrierten Spannungsversorgung beträgt in der Summe 2 A.

1.10 Integrierte Spannungsversorgung (nur Modelle - X2)

Der *eGrabber-4plus* ist in den Modellversionen EPC-032-X2 und EPC-032-X1-X2 mit einem Spannungsversorgungsteil ausgestattet.

Die integrierte Spannungsversorgung erlaubt den Betrieb des Grabbers und weiterer Komponenten des PC/104-Systems an einer unregelmäßigen Gleichspannung, zum Beispiel aus einer Schaltschrankversorgung oder einer Fahrzeugbatterie.

In vielen Anwendungsfällen kann so der Einsatz einer speziellen Spannungsversorgungskarte entfallen.

Der *eGrabber-4plus* kann zwei Ausgangsspannungen zur Verfügung stellen:

- +5 V, max 2 A
- +12 V, max 1A

Hinweis:

Die Spannungsversorgung ist nur bei folgenden Modellen vorhanden:

- EPC-032-X2
- EPC-032-X1-X2

1.10.1 Anschluß der Eingangsspannung

In die Spannungsversorgung kann eine unregelte Gleichspannung eingespeist werden.

Zulässiger Eingangsspannungsbereich:

verwendete Ausgangsspannungen		Eingangsspannungsbereich
+5 V	+12 V	
●		8 V... 28 VDC
●	●	15 V... 28 VDC

Tabelle 10: zulässige Eingangsspannungen EPC-032-X2

Achtung!

Die Eingangsspannung darf den angegebenen Maximalwert nicht überschreiten. Bei stark gestörter Eingangsspannung, besonders beim Auftreten hoher Spannungsspitzen, ist gegebenenfalls ein zusätzlicher Entstörfilter erforderlich.

Die Eingangsspannung wird am Steckverbinder X300 ⑦ eingespeist. Der Steckverbinder entspricht den Spannungsversorgungssteckern von 3,5“-Floppylaufwerken. Bild 11 zeigt die Anschlußbelegung.

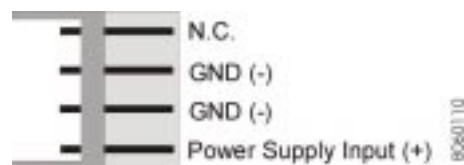


Bild 11: Belegung des Versorgungsspannungseingangs X300

Der Grabber ist mit einer Verpolungsschutzdiode ausgestattet.

Die maximal zulässige Eingangsleistung beträgt 22 W. Der Spannungsversorgungsteil wird durch die Sicherung SI301 ⑥ geschützt. Der maximal zulässige Eingangsstrom beträgt 2 A.

1.10.2 Ausgang Spannungsversorgung

Die geregelten Ausgangsspannungen stehen an der Buchse X302 (Ⓢ) zur Verfügung. *Bild 12* zeigt die Anschlußbelegung.



Bild 12: Belegung des Spannungsversorgungsausgangs X302

Verbinden Sie den Ausgang X302 mit dem Versorgungsspannungseingang der von Ihnen eingesetzten CPU-Karte. Beachten Sie dabei die Anschlußbelegung der CPU-Karte.

Hinweise:

- Die Ausgangsspannungen liegen nicht direkt auf den PC/104-Bus leisten an. Da CPU-Karten einen hohen Strombedarf haben, wird die Versorgung für das PC/104-System in der Regel über die CPU-Karte eingespeist und von dort über den Bus verteilt.
- Der Framegrabber-Teil wird nicht direkt aus der Spannungsversorgung gespeist, sondern über die Bus-Stecker des PC/104-Systems.
- An der Buchse X302 können auch Peripheriegeräte (Floppy-Laufwerk, Festplatte usw.) angeschlossen werden. Bei Festplatten und ähnlichen Komponenten muß darauf geachtet werden, daß der Anlaufstrom beim Einschalten den maximal zulässigen Ausgangsstrom nicht überschreitet. In diesem Fall spricht der Überstromschutz des Netzteils an und die Komponenten starten nicht korrekt.
- Das Vorhandensein der Betriebsspannungen wird durch die LEDs D307 / D308 neben dem Stecker X302 angezeigt. Leuchtet eine LED nicht oder nur dunkel, so ist es möglich, daß der integrierte Überstromschutz des Netzteils angesprochen hat. Bei Verringerung der Last liefert das Netzteil wieder die korrekte Ausgangsspannung.

CPU-Karten aus der MOPS-Serie der Firma Jumptec können auf Wunsch direkt mit dem Netzteil des eGrabber-4plus verbunden werden. Dazu befindet sich die Buchse X303 (Ⓜ) exakt an der gleichen Position wie der Spannungseingang der MOPS-CPU-Karten.

Mittels geeigneter Pfostenverbinder kann die Spannungsversorgung ohne Verbindungskabel einfach durch Aufeinanderstecken der Karten hergestellt werden.

Die Buchse X303 ist speziell auf die genannte CPU-Karte zugeschnitten und liefert alle nötigen Spannungen.

Ziehen Sie dazu auch das Handbuch der MOPS-CPU Karte zu Rate.

Wichtig:

Prüfen Sie genau, ob die Anschlußbelegung des Spannungsversorgungssteckers Ihrer CPU-Karte mit der Belegung der Buchse X303 übereinstimmt (vgl. *Tabelle 11*).

Pin	Funktion
1	GND
2	+5 V
3	N.C.
4	+12 V
5	N.C.
6	N.C.
7	GND
8	+5 V

Tabelle 11: Belegung der Buchse X303

1.10.3 Optionen Spannungsversorgung

- Versorgung über PCI-Bus

Die Ausgangsspannungen des Netzteils können direkt auf den PCI-Bus geführt werden. Die Kabelverbindung zwischen Grabber- und CPU-Karte kann dann entfallen.

Diese Option ist in der Regel nicht zu empfehlen, da die Einspeisung an einer gesonderten Buchse von den CPU-Kartenherstellern optimiert ist.

Zur Verbindung des Netzteil-Ausgangs mit den entsprechenden Pins der PC/104plus-Busstecker ist JP 301 folgendermaßen zu jumpern:

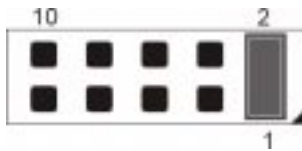


Bild 13: JP301: 5 V auf PC/104-Busstecker

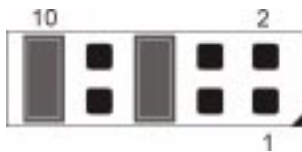


Bild 14: JP301: 12 V auf PC/104-Busstecker

Hinweis:

Die Verbindung des +12 V-Ausgangs mit dem PC/104Busstecker ist nur möglich, wenn die +12 V-Ausgangsspannung auch auf den Versorgungsspannungsausgang X3-25 gelegt wird.

- Spannungsversorgung ausschalten (Power-Down Pin)

Das integrierte Netzteil kann elektronisch abgeschaltet werden. Dies kann durch einen Schalter an dem Anschlußfeld X301 oder einem TTL-Signal geschehen.



Bild 15: Anschlußfeld X301

Die Spannungsversorgung wird abgeschaltet, wenn X301-Pin1 auf GND gelegt wird.

Dazu können die beiden Pins von X301 verbunden werden, z.B. durch einen Schalter.

An X301-1 kann auch ein Steuersignal angeschlossen werden, über welches das Netzteil ein- und ausgeschaltet werden kann.

1.11 Device Selector

Bei PC/104plus-Systemen muß mittels eines Jumpers die Position einer Steckkarte im PCI-Bussystem festgelegt werden. Die Position kann man sich als logische Position der Karte im PCI-System oder als „Geräteadresse“ vorstellen.

Die Position wird auch als „Slot“, „Slot-Nummer“ oder manchmal als „Device-ID“ bezeichnet.

Zwei Karten in einem PCI-System dürfen nicht auf die selbe Position eingestellt werden. Das System bootet in diesem Fall meist gar nicht.

PC/104plus-CPU-Karten stellen in der Regel bis zu vier Positionen zur Verfügung.

- Prüfen Sie, welche Positionen ihre CPU-Karte zur Verfügung stellt. Die für den eGrabber-4plus benötigte Position muß masterfähig sein.
- Prüfen Sie, welche Positionen von anderen PCI-Erweiterungskarten in Ihrem System benutzt werden. Suchen Sie für den Grabber eine freie, masterfähige Position aus.

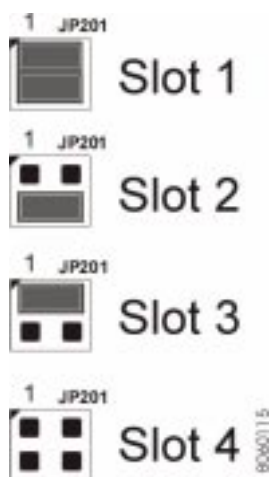


Bild 16: Jumperstellungen des Device-Selectors JP201

1.12 Hinweise zur CE-Konformität und Störsicherheit

Beachten Sie bitte beim Aufbau des PC/104-Systems die Regelungen bezüglich CE-Konformität und Störsicherheit.

Da PC-Systeme mit hohen Taktfrequenzen arbeiten, besteht die Gefahr der Aussendung und Störung von und durch elektro-magnetischen Feldern. Dem kann durch sorgfältigen Aufbau des Systems entgegengewirkt werden.

Beachten Sie besonders die Masseführung innerhalb des Systems. Hochfrequente Störungen auf der Masse können über die Kabelschirmung der Videokabel abgestrahlt werden. Günstig ist in diesem Fall die Verbindung der Betriebsmasse mit einer Schirmmasse (Gehäuse oder PE-Anschluß) in der Nähe der Eingangsbuchsen.

Der eGrabber-4plus bietet unterschiedliche Möglichkeiten der Masseanbindung zur Anpassung an unterschiedliche Betriebssituationen. Drei der vier Befestigungsbuchsen liegen auf Betriebsmasse (GND). Über metallische Stehbolzen oder Kabelverbindungen kann GND der Grabberkarte mit der Masse der CPU-Karte, dem Gehäuse oder PE verbunden werden. Durch Kunststoff-Stehbolzen kann eine Isolierung der Massepotentiale erreicht werden. Die kann wünschenswert sein, um z.B. Masseschleifen zu vermeiden.

Der neben der S-Video-Buchse befindliche Shield-GND Anschluß ⑤ stellt eine Verbindung zum Schirmanschluß der S-Video-Buchse her.

Um Störabstrahlungen über die Masseleitungen zu verhindern, kann es sinnvoll sein, die Anschlußleitungen durch einen Ferrit zu führen. Der Ferrit ist dabei dicht am Grabber anzubringen und das Kabel sollte je nach notwendiger Dämpfungswirkung ein- oder zweimal durch ihn hindurchzuführen (eine Schleife).

Für Videoleitungen eignet sich z.B. der Klappferrit Nr. 742.712.2 der Fa. Würth, Kupferzell.

Achtung!

Beachten Sie, daß starke Störspitzen auf den Videosignalen oder auf dem Kabelschirm die Eingangsstufe des *eGrabber-4plus* zerstören können.

In stark gestörten Industrieumgebungen und bei Verwendung langer Zuleitungen sind daher unbedingt zusätzliche Entstörmaßnahmen vorzusehen.

Bei Verlegung langer Videoleitungen oder dem Einbau der Bildverarbeitungskomponenten in Maschinen oder Anlagen können Mantel- bzw. Potentialausgleichsströme auftreten, die durch geeignete Vorrichtungen von den Eingängen des *eGrabber-4plus* fernzuhalten sind.

PHYTEC kann keine Haftung für Schäden übernehmen, die durch unsachgemäßen Anschluß der Signalquellen entstanden sind.

2 Einbau und Installation der Grabberkarte

Die Grabberkarte hat die Aufgabe, die analogen Bildsignale der Kamera umzuwandeln und dem Computer bzw. der Software in digitaler Form zu präsentieren.

Wenn Sie mit dem Zusammenbau von PC/104plus-Steckkarten und Systemen noch keine Erfahrung haben, nehmen Sie sich dafür bitte etwas Zeit.

Es ist nicht schwer, sollte aber sorgfältig ausgeführt werden.

2.1 Einsetzen der Steckkarte

Achtung!

Das PC/104plus-System muß von der Spannungsversorgung getrennt sein. Es muß sichergestellt sein, daß das System spannungsfrei ist.

Die Spezifikation schreibt vor, daß die Karte an den virtuellen Positionen (Slots oder Device-ID)1-4 zu installieren ist. Wählen Sie eine dieser Positionen aus, beachten Sie aber, daß es sich um eine masterfähige Position handelt (in manchen Systemen sind nicht alle Positionen masterfähig).

- Zwei Karten dürfen nicht die gleiche Position besitzen (z.B. zwei Karten verwenden Position 3)
- Wenn Sie sich für eine Position entschieden haben, muß diese vor der Installation durch Jumper JP201 eingestellt werden (*siehe Bild 17*).

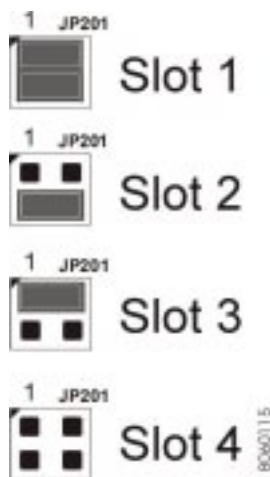


Bild 17: Jumperstellungen JP201 (Slotauswahl)

Hinweis:

Ziehen Sie das Manual des CPU-Board-Herstellers zu Rate, falls nicht sicher sein sollte, ob es sich um eine Master-Position handelt.

Achtung!

Wenn Sie den eGrabber-4plus an einer Slave-Position installieren, kann das System möglicherweise nicht mehr starten (booten). In jedem Fall arbeitet der eGrabber-4plus dann nicht korrekt. An Position 3 oder 4 darf bei manchen Systemen nur jeweils *ein* Master installiert sein.

- Stecken Sie nun den eGrabber-4plus vorsichtig auf die PC/104 und PC/104plus-Steckerleiste der Hauptplatine auf. Der Grabber kann auch auf andere Karten im System gesteckt werden, es muß nur gewährleistet sein, daß alle relevanten Signale der CPU-Karte zur Verfügung stehen.
- Die Grabberkarte kann nur in einer Ausrichtung richtig verbunden und eingesteckt werden.
- Drücken Sie die Karte nicht mit Gewalt in die Steckleiste. Die Pins könnten sich verbiegen oder abbrechen, und die Karte könnte Schaden nehmen!
- Achten Sie darauf, daß die Steckerleisten ganz eingesteckt werden und mit der Buchse übereinstimmen bzw., daß keine Pins sich neben bzw. außerhalb der Leisten befinden.

- Prüfen Sie, daß kein mechanischer oder elektrischer Kontakt zwischen Bauelementen von darüber- oder darunterliegenden Karten besteht. Bedingt durch die Konstruktionsweise der PC/104-Karten kann es u. U. vorkommen, daß hohe Bauelemente auf den Karten sich beim Aufstecken berühren können oder ein vollständiges Aufstecken der Karten dadurch nicht möglich ist. Wählen Sie dann nach Möglichkeit eine andere Reihenfolge der Karten oder verwenden Sie einen Zwischenstecker als Abstandshalter.
- Befestigen Sie nun die Karte mit Hilfe von Schrauben und Abstandsbolzen oder –rollen an den vorgesehenen Befestigungsbohrungen.
- Prüfen Sie abschließend nochmals, daß keine Kurzschlüsse zwischen den Komponenten benachbarter Leiterplatten oder ggfs. deren Steckverbinder aufgetreten sind.

Achtung!

Aus Stabilitätsgründen und damit die Karte eine sichere Masseverbindung über die Bolzen bekommt, befestigen Sie die Karte durch die vier Schraublöcher mit den Bolzen.

2.2 Spannungsversorgung

Grundsätzlich wird das PC/104-System über die CPU-Karte versorgt. Prüfen Sie anhand der Anleitung der CPU-Karte, welche Spannungen benötigt werden und wie sie angeschlossen werden müssen.

Der Grabber benötigt eine Versorgungsspannung von +5 V. Die CPU-Karte leitet diese Spannung normalerweise über den PCI-Bus-Stecker an die Grabberkarte weiter. Diese benötigt daher keine besondere Spannungsversorgung.

- Die an der Pfostenleiste X3 angeschlossene Kamera kann auch über die Framegrabberkarte erfolgen. Nähere Informationen darüber finden Sie in *Kapitel 1.9*.

Die Grabbermodelle **EPC-032-X2** und **EPC-032-X1-X2** besitzen ein separates Spannungsversorgungsteil, welches das gesamte PC/104-System versorgen kann.

Das Spannungsversorgungsteil ist vom Grabberteil völlig unabhängig.

Um die integrierte Spannungsversorgung zu nutzen, schließen Sie an der Eingangsklemme X300 eine Eingangsgleichspannung im Bereich 15 bis 28 VDC an.

Wenn Sie den +12 V-Ausgang der Spannungsversorgung nicht benutzen, kann der Eingangsspannungsbereich 8 bis 28 VDC betragen. Die Anschlußbelegung von X300 zeigt *Bild 18*.

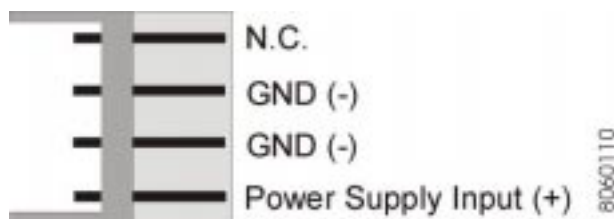


Bild 18: Anschlußbelegung Versorgungsspannungseingang

An den Schraubklemmen X302 liegen die geregelten Ausgangsspannungen +5 V und +12 V an (siehe Bild 19). Sie können die Stromversorgungs-Eingänge der CPU-Karte dort anschließen. Mit den beiden grünen Leuchtdioden können Sie kontrollieren, ob die Spannungen anliegen. Beachten Sie, daß die Ausgangsspannungen nicht an den Grabberteil weitergeleitet oder über die PC/104-Stecker verteilt werden. Daher müssen die Verbraucher in der Standard-Konfiguration immer an X302 angeschlossen werden. *Weitere Informationen über die integrierte Spannungsversorgung finden Sie in Kapitel 1.10.*



Bild 19: Spannungsversorgungs-Ausgang

Hinweis:

Die Spannung an der Buchse X300 darf 28 Volt nicht überschreiten.

Die Stromentnahme ist durch die Sicherung F1 auf 1,5 A begrenzt (Sicherungen gibt es bei PHYTEC unter Best.Nr.: KF012)

2.3 Installieren der Treiber

Als nächstes müssen noch der Treiber und die Demosoftware der Karte installiert werden.

Dies fällt für den Treiber, je nach Betriebssystem, unterschiedlich aus. Wie, wird im folgenden erklärt.

- Schalten Sie das PC/104plus-System ein.

Beim Hochfahren sollte das BIOS des Rechners die Karte automatisch erkennen. Viele BIOS-Versionen zeigen kurz eine Liste der gefundenen PCI-Geräte. Der *eGrabber-4plus* wird in dieser Liste zwei Mal als „Multimedia Device“ aufgeführt.

Es gibt nun zwei Möglichkeiten:

1. Das Betriebssystem erkennt die Karte und fragt Sie nach Treibern oder
2. Das Betriebssystem erkennt die Karte nicht automatisch (z.B. Windows NT) und Sie müssen die Treiber manuell installieren.

Die Treiberinstallation läuft je nach Betriebssystem folgendermaßen ab:

- **Windows‘95™:**

Nachdem die Karte erkannt wurde, werden Sie gebeten, den Treiber zu installieren. Es erscheint ein Fenster *Neue Hardwarekomponente gefunden*. Darin wählen Sie die Option *Andere Position* aus und bestätigen mit *OK*.

Nun erscheint ein weiteres Fenster *Andere Position auswählen*.

Legen Sie die **PHYTEC Vision Utilities** CD in Ihr CD-ROM Laufwerk. Wählen Sie *Durchsuchen* und im darauf erscheinenden Fenster das CD-ROM Laufwerk aus.

Wechseln Sie den Pfad auf *pciGrab4\driver\win95_98*. Bestätigen Sie mit *OK*.

Es erscheint eine Liste mit den gefundenen Treibern auf der CD. Wählen Sie aus dieser Liste den *PHYTEC PCI-Grabber* aus.

Der Treiber sollte nun automatisch von CD auf Ihren Rechner installiert werden

Bestätigen Sie im folgenden Fenster den Neustart des Rechners.

Der Rechner sollte nun ganz normal bis zum Betriebssystem hochfahren.

Mit diesem Schritt haben Sie den Treiber erfolgreich installiert.

Beachten Sie bitte noch *Kapitel 2.3.1*. Gehen Sie dann zu *Kapitel 2.4*, um die Demosoftware zu installieren.

- **Windows‘98/ME/2000™:**

Nachdem die Karte erkannt wurde, werden Sie gebeten, den Treiber zu installieren.

Wählen Sie in dem Fenster *Hardware Assistent* die Option *Nach dem besten Treiber für das Gerät suchen* aus und bestätigen Sie mit *OK*.

Im darauffolgenden Fenster wählen Sie *Geben Sie eine Position an*. Legen Sie die **PHYTEC Vision Utilities** CD in Ihr CD-ROM Laufwerk. Wählen Sie *Durchsuchen* und im darauf erscheinenden Fenster das CD-ROM Laufwerk aus.

Wechseln Sie den Pfad auf *pciGrab4\driver\win2K_98*. Bestätigen Sie mit *OK*.

Es erscheint eine Liste mit den gefundenen Treibern auf der CD. Wählen Sie aus dieser Liste den *PHYTEC PCI-Grabber* aus.

Der Treiber sollte nun automatisch von CD auf Ihren Rechner installiert werden

Mit diesem Schritt haben Sie den Treiber erfolgreich installiert.

Beachten Sie bitte noch *Kapitel 2.3.1*. Gehen Sie dann zu *Kapitel 2.4*, um die Demosoftware zu installieren.

- **Windows NT4.0™ (mit ServicePack 6):**

WindowsNT erkennt die Karte nicht automatisch. Der Treiber muß manuell installiert werden. Legen Sie die **PHYTEC Vision Utilites** CD in Ihr CD-ROM Laufwerk. Wählen Sie unter WindowsNT im Hauptverzeichnis der CD das Programm *Start.exe* aus.

Im erscheinenden Fenster wählen Sie *PCI-Grabber*, dann *Install drivers* und *WindowsNT4.0*.

Folgen Sie den Anweisungen des Installationsprogrammes. Die notwendigen Treiber werden dann automatisch installiert. Bestätigen Sie im folgenden Fenster den Neustart des Rechners. Der Rechner sollte nun ganz normal bis zum Betriebssystem hochfahren.

Mit diesem Schritt haben Sie den Treiber erfolgreich installiert.

Beachten Sie bitte noch *Kapitel 2.3.1*. Gehen Sie dann zu *Kapitel 2.4*, um die Demosoftware zu installieren.

2.3.1 Zusätzliche Treiber (optional)

Sie können noch zusätzliche Treiber von der CD-ROM installieren. Diese sind nicht unbedingt notwendig, um die Karte wie in diesem Handbuch beschrieben, zu betreiben.

Der **Twain-Treiber** ist ein Standardtreiber, der z.B. in Grafikprogrammen oder Photo- bzw. Scannerprogrammen eingesetzt werden kann, um Bilder einzulesen, und mit dem jeweiligen Programm zu bearbeiten. So werden Grabber und Kamera quasi wie ein Scanner behandelt.

Für die weitere Bedienung des Programmes in Verbindung mit dem Twain-Treiber konsultieren Sie bitte das Handbuch des verwendeten Grafikprogrammes.

Der „**LabView**“-**Treiber** ist ein Treiber für das Mess- und Automatisierungsprogramm von National Instruments (IMAQ-Paket erforderlich). Für die Anwendung des Programms oder des Treibers konsultieren Sie bitte das LabView Handbuch.

Wenn Sie die Treiber installieren möchten:

Legen Sie die **PHYTEC Vision Utilities** CD in Ihr Laufwerk und starten Sie aus dem Hauptverzeichnis der CD die Datei *start.exe*.

Im folgenden Fenster wählen Sie *PCI-Grabber* aus. Sie sehen in der darauf erscheinenden Installationsfenster unter anderem die zwei Einträge:

- **Install Twain**
- **Install LabView**

2.4 Installation des Demoprogrammes

Das Demoprogramm ermöglicht Ihnen, in Verbindung mit einer Kamera, die Karte zu testen, Bildparameter zu modifizieren, und einfache Bildoperationen durchzuführen.

Die Installation des Programmes:

- Legen Sie die **PHYTEC Vision Utilities** CD in Ihr CD-ROM Laufwerk.
- Dann wird das CD-ROM Laufwerk ausgewählt und das Programm *start.exe* aus dem Hauptverzeichnis der CD gestartet.
- Im erscheinenden Installationsmenü (siehe Bild 20) wählen Sie *PCI-bus grabber* aus.
- Klicken Sie danach auf *Install Windows demosoftware*

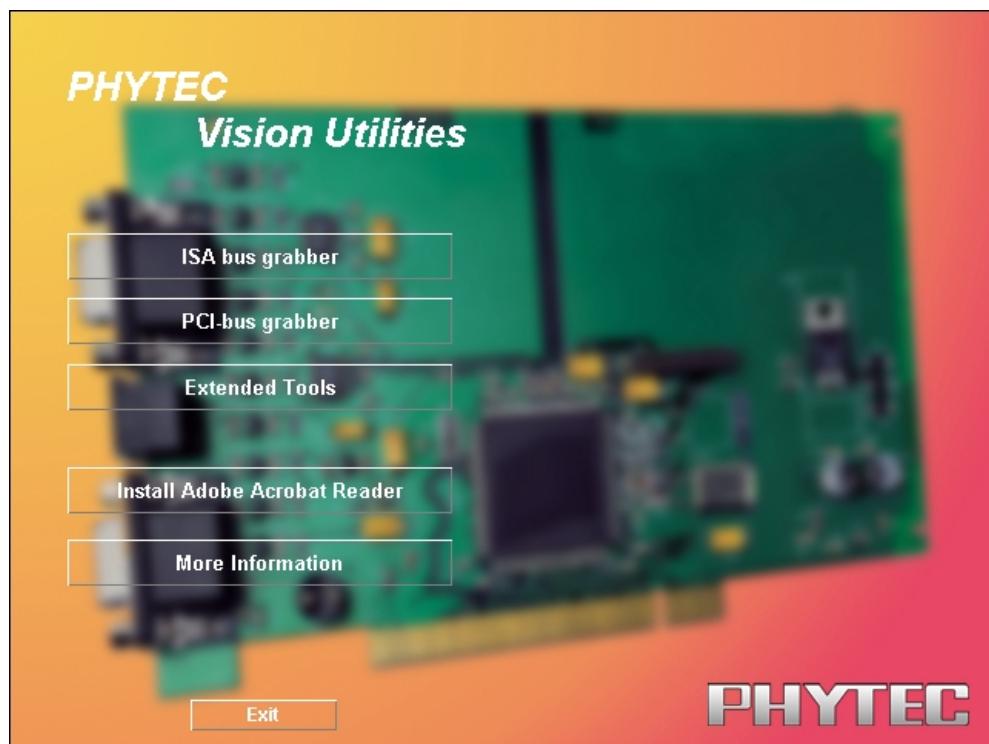


Bild 20: PHYTEC Installationsmenü

- Folgen Sie einfach den Installationsanweisungen und das Demoprogramm wird automatisch auf dem Rechner installiert.

3 Das Anschließen der Videoquellen

Beim eGrabber4-plus (siehe Bild 21) haben Sie die Möglichkeit, eine oder mehrere Videoquellen anzuschließen. Dies können z.B. Videokameras oder auch Videorekorder mit den entsprechenden Ausgängen sein (Composite oder S-Video).

Je nach Grabber-Modell können 3 Composite und eine S-Video (EPC-032-X1, EPC-032-X1-X2) oder 9 Composite und eine S-Video-Quelle (EPC-032, EPC-032-X2) angeschlossen werden.

Die Umschaltung der einzelnen Kanäle erfolgt per Software bzw. durch das mitgelieferte Demoprogramm.

Es kann jedoch immer nur eine Kamera aktiv sein und Bilder liefern.

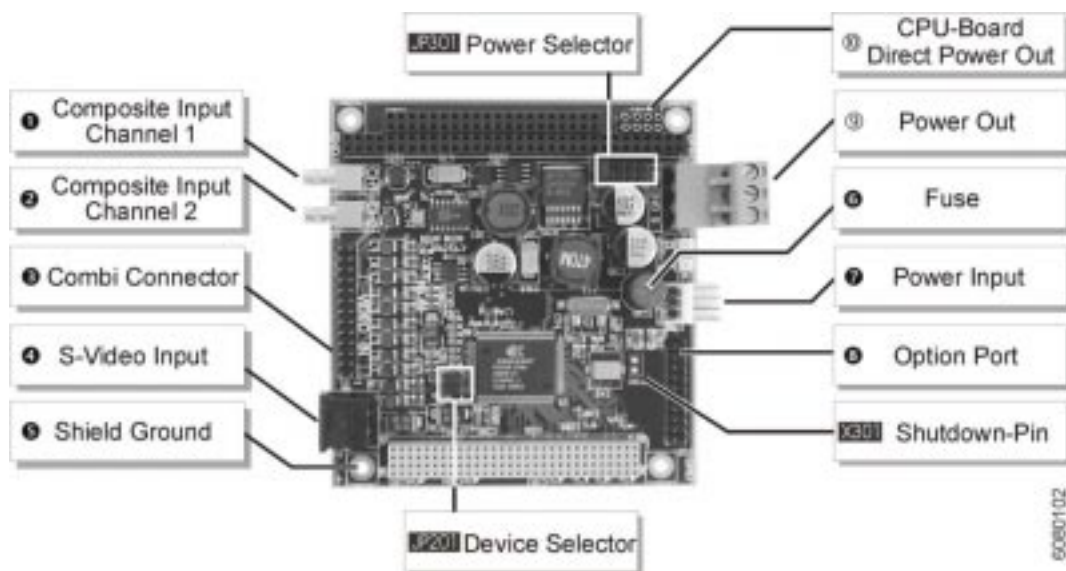


Bild 21: Übersicht der eGrabber4-plus Anschlüsse

Die Composite-Eingänge liegen an den SMB-Buchsen 1/2 und an der 26pol Pfostenleiste X3 3. Durch die Leiste X3 ist auch die Spannungsversorgung einer Kamera möglich.

Die dafür benötigten Kabel können Sie bei PHYTEC beziehen (siehe Kapitel 1, „Zubehör“).

S-Video-Signale können an der Mini-DIN-Buchse 4 eingespeist werden. Wahlweise können S-Video-Quellen auch über die Leiste X3 angeschlossen werden.

Genaue Informationen zu den Anschlußbelegungen der Buchsen finden Sie vorne im „Kapitel 1, „Technische Daten“.

3.1 Anschlußmöglichkeiten von Videoquellen

Hier werden kurz verschiedenen Anschlußmöglichkeiten der Videoquellen an den Grabber gezeigt.

Die gezeigten Kabel können Sie alle bei PHYTEC beziehen.

Die Legende der verwendeten Kabel beinhaltet die Bezeichnung und die PHYTEC-Bestellnummer (*siehe Bild 22*).

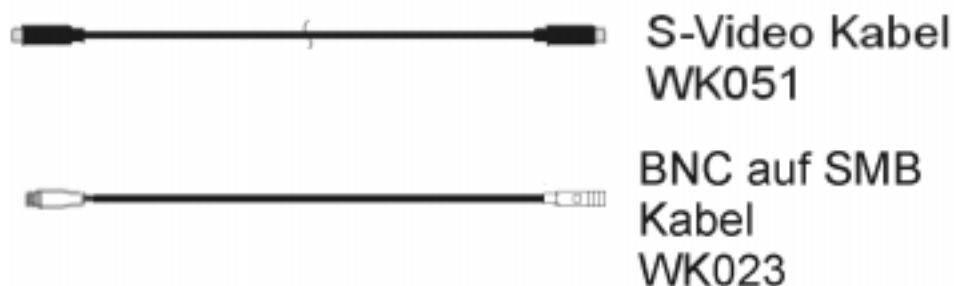


Bild 22: Video-Abschlußkabel – Legende: (Bezeichnung, PHYTEC Best.Nr.)

Ziehen Sie bitte die Anleitung ihrer Videoquelle zu Rate, um die Anschlußkompatibilität zu überprüfen.

Im folgenden Kapitel werden die verschiedenen Kabelarten kurz erklärt.

3.1.1 Das S-Video Kabel

Das S-Video Kabel wird grabberseitig mit der runden Mini-DIN Buchse ④ verbunden. Die verwendete Videoquelle (z.B. Kamera mit S-Video Ausgang) sollte ebenfalls eine solche Buchse besitzen.

3.1.2 Die Composite-Anschlüsse

Die BNC-Stecker können mit den Composite-Ausgängen (BNC-Buchsen) einer Videoquelle verbunden werden.

Hinweis:

Sollte die Compositequelle eine Cinch-Buchse besitzen, muß ein Cinch-auf-BNC-Adapter (75 Ω) verwendet werden.

Das Ende mit dem SMB-Stecker wird in den Grabber (Buchse ❶ oder ❷) gesteckt.

Um ein Bild einer Kamera darzustellen, ist in der Anwendungssoftware bzw. im Demoprogramm der richtige Kanal zu wählen. Gegebenenfalls kann die mitgelieferte Software automatisch erkennen, welcher Kanal ein Signal führt (*siehe Kapitel 4*).

4 Inbetriebnahme des Grabbers mit Demoprogramm

Das Demoprogramm und der Grabbertreiber müssen korrekt installiert sein (*siehe Kapitel 2*), um in diesem Kapitel weiterarbeiten zu können.

Unter **START/Programme/Phytec/eGrabber4plus/Grab4PCI** finden Sie das Demoprogramm. Nach dem Start erscheint folgendes, zunächst noch leeres Programmfenster mit Menüleiste (*siehe Bild 23*).

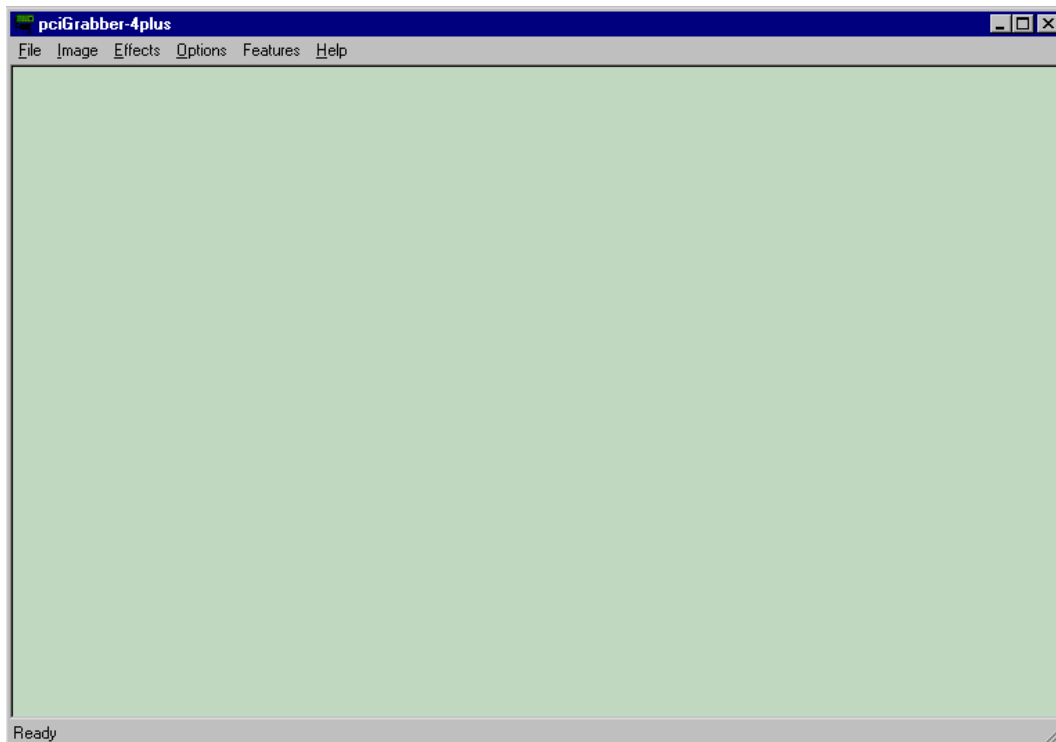


Bild 23: Oberfläche des Demoprogramms

Wir wollen zunächst ein bewegtes Live-Bild von einer Kamera darstellen. Stellen Sie deshalb zunächst sicher, daß eine Videokamera o.ä. an den Grabber angeschlossen ist und ein Bildsignal liefert.

In der Menüleiste finden Sie den Eintrag *Image*. Klicken Sie darauf und es öffnet sich folgendes Menü (*siehe Bild 24*).

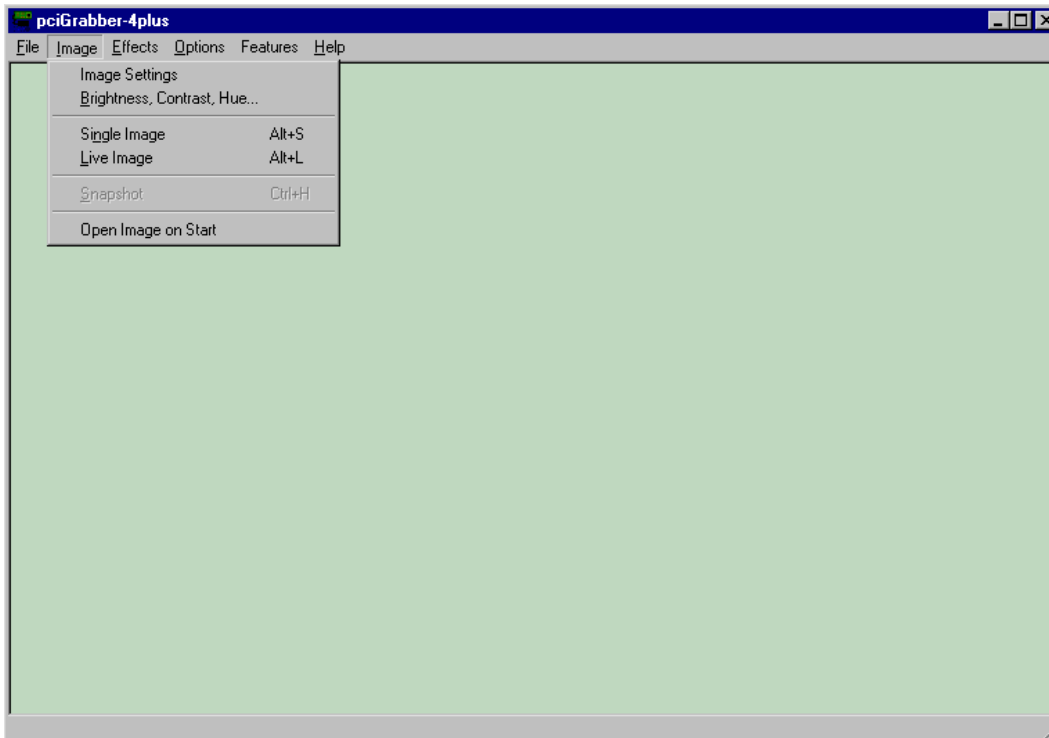


Bild 24: Menüauswahl Image

Wählen Sie den Eintrag *Image Settings*, um die Bildparameter des zu grabbenden Bildes einzustellen (siehe Bild 25):

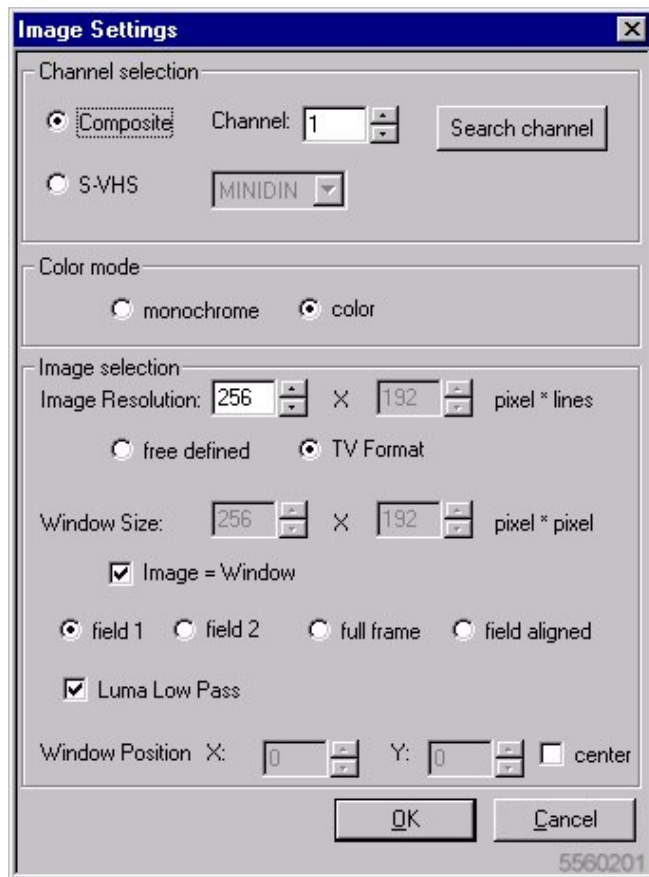


Bild 25: Oberfläche zum Einrichten der Bildausgabe

Die genaue Bedeutung der einzelnen Parameter wird später noch erklärt. Zunächst soll aber das Live-Bild auf den Bildschirm gebracht werden, um den Grabber zu testen. Dazu müssen nur wenige Angaben gemacht werden:

Es ist wichtig, den richtigen Videoeingang des Grabbers zu wählen. Im Bereich *Channel selection* geben Sie den Typ der Videoquelle (Composite / S-VHS) und den verwendeten Eingangskanal an.

Sie können den Eingangskanal manuell wählen oder auch automatisch suchen lassen. Für letzteres klicken Sie auf die Schaltfläche *Search channel* und der erste gefundene Kanal mit einer aktiven Videoquelle wird eingestellt.

Im Bereich *Color Mode* können Sie auch wählen, ob die Bildarstellung in Farbe oder monochrom erfolgen soll.

Die restlichen Einträge unter *Image selection* können wir zunächst bei den Vorgabewerten belassen.

Durch ein Klick auf *OK* verlassen Sie das Menü.

Wählen Sie erneut den Eintrag *Image* und dann *Live Image*.

Es erscheint nun ein Livebild der gewählten Videoquelle in einem neuen Fenster (*siehe Bild 26*).

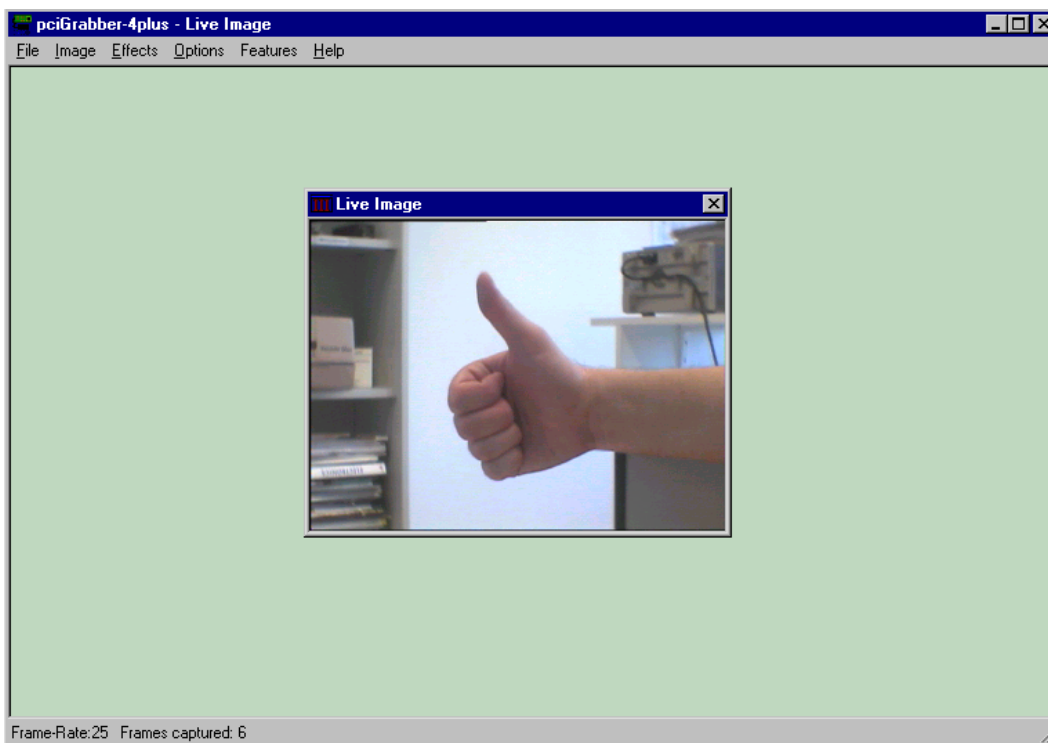


Bild 26: *Livebild der Videoquelle*

Wenn nur ein blaues Bild erscheint, überprüfen Sie bitte alle Verbindungen. Prüfen Sie bitte, ob die Kamera mit Spannung versorgt wird.

Weiterhin besteht die Möglichkeit, daß der falsche Kanal oder Grabber gewählt wurde.

Weitere mögliche Fehlerquellen sind im *Kapitel 7, „Störungssuche“* beschrieben.

Hinweis:

Wenn Sie mehrere Grabber in einem System betreiben, muß zunächst ausgewählt werden, welcher angesteuert werden soll. Dies geschieht unter *Options*.

In der unteren Leiste des Hauptfensters sehen Sie die Anzeige *Frame-Rate xx* (*xx=Zahl*). Die Zahl gibt die Bilder pro Sekunde an, die im Live-Fenster aufgebaut werden. Der Wert ist abhängig von der Bildgröße und der Leistungsfähigkeit des Rechners, da das digitalisierte Bild vom RAM in die Grafikkarte übertragen werden muß, damit es schließlich auf dem Bildschirm erscheint.

Hinweis:

Die Bilddaten werden vom Grabber (unabhängig von der Rechenleistung des Prozessors) immer in Echtzeit im Arbeitsspeicher (RAM) des PC/104 Systems abgelegt.

Die weitere Verarbeitung ist abhängig von der Rechenleistung des Prozessors (CPU) des Systems.

Weiterhin wird in der Statuszeile ein Zähler, der die Gesamtzahl der erfassten Livebilder (*Frames Captured*) anzeigt, dargestellt.

Wenn dieser Zähler bei 255 angelangt ist, fängt er wieder bei 0 an.

Dies kann z.B. als Indikator verwendet werden, ob der Grabber überhaupt aktiv ist.

4.1 Beschreibung des Demoprogramms

In diesem Kapitel sehen wir uns nun die Funktionen bzw. Menüs des mitgelieferten Demoprogramms genauer an.

Das Menü *Image Settings* (siehe Bild 27) beinhaltet Parameter, welche die Bilderzeugung und –darstellung beeinflussen:

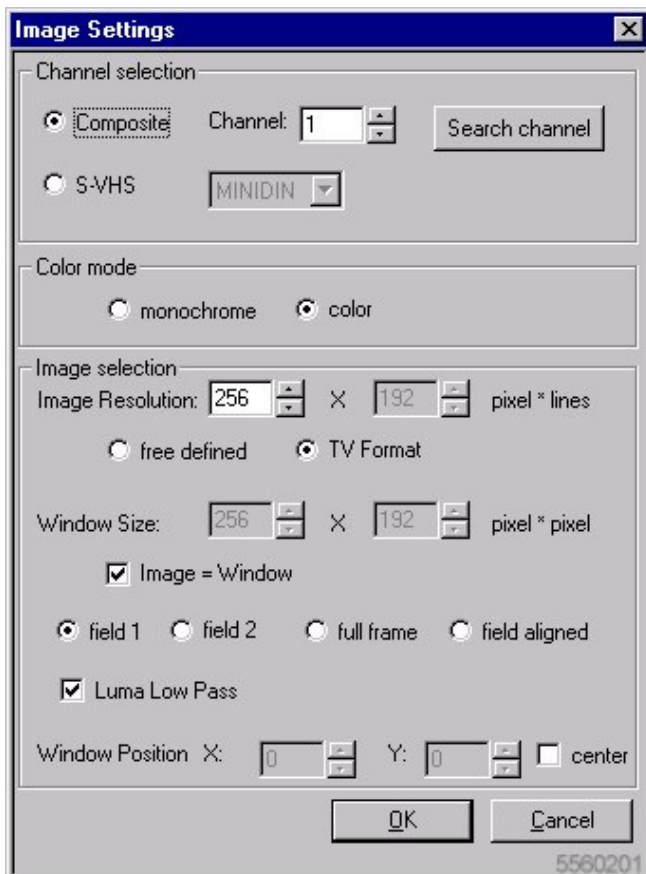


Bild 27: Menü „Image Settings“

Sie können hier Einstellungen vornehmen, bevor Sie ein Livebild sehen. Während dem Livebildbetrieb können diese Einstellungen nicht geändert werden.

Das Kapitel *Channel selection* ermöglicht die Einstellung des Videoquellentyps und die Kanalwahl.

Wählen Sie zunächst die richtige Signalart mit den Buttons *Composite* und *S-VHS*.

- **Composite-Quellen**

Composite bezieht sich auf die beiden SMB-Buchsen ❶/❷ bzw. auf die Composite Kanäleingänge an der Pfostenleiste X3 ❸.

Wählen Sie unter dem Eintrag *Channel* den richtigen Eingangskanal für die angeschlossene Kamera.

Sie können die Kamera auch suchen lassen, indem Sie auf *Search Channel* klicken. Das Programm stellt dann den ersten Kanal ein, auf dem ein Videosignal gefunden wurde.

- **S-Video - Quellen**

S-Video (oder „*S-VHS*“) – Quellen können auf zwei Arten an den Grabber angeschlossen werden. Über den Mini-DIN Stecker ❹ oder über die Pfostenleiste X3 ❸.

Bei der Kanalwahl steht die Auswahl *S-VHS* in Verbindung mit *MINIDIN* für beide Anschlußarten.

Achtung!

Es darf immer nur **eine** *S-Video* Anschlußmöglichkeit verwendet werden.

Mit den Buttons *color* und *monochrom* kann das Bild in Farbe (bei Verwendung einer Farb-Videoquelle) oder monochrom wiedergegeben werden.

Im unteren Fensterbereich „*Image selection*“ kann die Größe und Auflösung des gegrabten und dargestellten Bilds eingestellt werden.

Mit der Einstellung *Image Resolution* kann die Bildauflösung (= „Qualität“) eingestellt werden. Die Einstellung teilt sich in x-Richtung für die Pixelzahl und in y-Richtung für die Zeilenzahl. Beide Werte können getrennt verändert werden, wenn der Button *free defined* selektiert wurde.

Dabei ist jedoch zu beachten, daß das Bild verzerrt (gedehnt oder gestaucht) wiedergegeben wird, wenn das Seitenverhältnis nicht 4:3 beträgt. (Dieses Verhältnis von Breite zu Höhe ist durch die Fernsehnorm vorgegeben).

Mit dem Button *TV Format* verhindern Sie, daß das Bild verzerrt dargestellt wird, da dann bei eingegebener Pixelzahl die Zeilenzahl im Verhältnis 4:3 (Breiten/Höhenverhältnis) automatisch berechnet wird.

Mit dem Button *Window Size* kann ein Ausschnitt aus dem gesamten digitalisierten Bildfeld herausgeschnitten und anstelle des gesamten Bildes auf dem Bildschirm angezeigt werden.

Dieser kann entsprechend kleiner als z.B. das Sichtfeld einer Kamera sein. Soll der gesamte digitalisierte Bereich dargestellt werden, setzen Sie das Häkchen bei *Image=Window*.

Die Einstellung *Window Size* verzerrt die Bildgeometrie nicht, da es kein Skalieren, sondern ein Ausschneiden ist.

Hinweis:

Beachten Sie, daß sowohl Skalierung als auch Ausschnittsbildung im Grabber in Echtzeit durchgeführt werden. Das dargestellte Bild wird also genau so vom Grabber im Speicher abgelegt, wie es auf dem Bildschirm erscheint. Das ist ein großer Vorteil, da keine CPU-Leistung für diese Aufgabe benutzt werden muß.

Ein kurzer Exkurs in die analoge Fernsehtechnik, nach der die Kamera arbeitet, trägt zum besseren Verständnis der Buttons *field1*, *field2*, *full frame* und *field aligned* bei.

Ein Fernsehbild (Norm-Videosignal), besteht aus zwei ineinander verschachtelten Teilbildern, den sog. Halbbildern (*siehe Bild 28*). Diese Halbbilder werden analog hintereinander erzeugt und nachfolgend im darstellenden Gerät (z.B. Fernseher) verschachtelt.

Die Verschachtelung dient der Verringerung des Flimmereffekts eines TV-Bildes.



Bild 28: Entstehung des Vollbildes aus zwei Halbbildern mit je 7 Zeilen

Das Signal besteht bei der PAL-Norm aus 625 Zeilen, die in ein erstes Halbbild (*ungerade/odd, field1 mit Zeilen 1-625*) und ein zweites Halbbild (*gerade/even, field2 mit Zeilen 2-624*) aufgeteilt sind.

Man kann die Bildfläche schon mit einem Halbbild vollständig erkennen, die vertikale Auflösung ist jedoch um die Hälfte reduziert, da dann nur noch 288 Zeilen vorhanden sind (abzüglich nicht sichtbarer Vor- und Nachlaufzeilen, sowie Prüf- und Datenzeilen bleiben ca. 576 sichtbare Zeilen von insgesamt 625 Zeilen).

Der Vorteil, wenn nur ein Halbbild digitalisiert wird, ist jedoch die Schnelligkeit von 20 ms im Gegensatz zu 40 ms bei einem Vollbild (*full frame*), also beiden Halbbildern.

Wenn man immer das gleiche (z.B. das erste) Halbbild digitalisieren möchte, entsteht allerdings immer eine Pause von 20 ms zwischen den Digitalisierungsvorgängen.

Ein weiterer Nachteil der Vollbilddigitalisierung ist, daß Objekte, die sich schnell bewegen, einen Versatz im Bild erzeugen können. Das Objekt ist im ersten Halbbild an einer anderen Position als im zweiten Halbbild, so daß eine Art Kammefekt entsteht und das Bild dann ähnlich wie *Bild 29* dargestellt aussieht.



Bild 29: Kammefekt bei schnell bewegten Objekten

Sie haben im Demoprogramm die Möglichkeit, die oben erklärten Parameter zu beeinflussen.

Bei vertikalen Auflösungen kleiner 288 Zeilen ist es demnach günstiger, ein Halbbild zu digitalisieren. Dies kann mit den Buttons *field1* (erstes, ungerades Halbbild) oder *field2* (zweites, gerade Halbbild) ausgewählt werden.

Ist die Zeilenzahl größer 288, müssen beide Halbbilder digitalisiert werden, dazu klicken Sie *full frame* an. Wenn Sie bei *Image Resolution* eine Zeilenzahl größer 288 eingestellt, wird diese Option automatisch gesetzt.

Der Button *field aligned* ermöglicht bei Halbbild-Darstellung die doppelte Anzahl an Halbbildern pro Sekunde. Dabei entfällt die 20 ms-Pause zwischen der Halbbilddigitalisierung.

Würde man einfach beide Halbbilder nacheinander darstellen, so würde der dargestellte Bildinhalt optisch immer eine halbe Zeile nach oben und unten springen.

Dies geschieht, da die räumliche Verschachtelung der Zeilen zu einem Vollbild nicht durchgeführt wird. Bei der Einstellung *field aligned* verschiebt der Grabber das zweite Halbbild automatisch um eine halbe Zeile, so daß es genau auf das erste Halbbild zum Liegen kommt.

Dadurch tritt dieser Sprungeffekt nicht auf. Diese Einstellung *field aligned* ist also dann nützlich, wenn man Bilder mit höchstens 288 Zeilen so schnell wie möglich nacheinander digitalisieren will (1 Halbbild in jeweils 20 ms).

Bei horizontalen Auflösungen kleiner 360 Pixel sollten Sie das Häkchen bei *Luma Low Pass* setzen. Dieser Tiefpaß bewirkt quasi eine „Glättung“ des in der Größe reduzierten Videobildes. Es wird automatisch aktiviert bei Auflösungen kleiner 360 Pixel und deaktiviert bei größeren Auflösungen.

Mit *Window Position* können Sie innerhalb des oben gewählten Bildausschnitts die Position des darzustellenden Bildfensters vorgeben. Die Werte repräsentieren die Position der linken oberen Ecke. Soll der Ausschnitt genau in der Mitte des TV-Bildes liegen, kann der Haken bei *center* gesetzt werden.

Die Einstellung bewirkt also eine Verschiebung des Bildausschnitts im Gesamtbild. Sie kann natürlich nur dann vorgenommen werden, wenn der Bildausschnitt *Window Size* kleiner ist als das Gesamtbild *Image Resolution*.

4.2 Bildkontrolle

Während der Darstellung des Live-Bilds können Sie das Bildkontrollfenster öffnen, indem Sie unter *Image* auf *Brightness*, *Contrast*, *Hue...* klicken. Es öffnet sich der dargestellte Dialog.

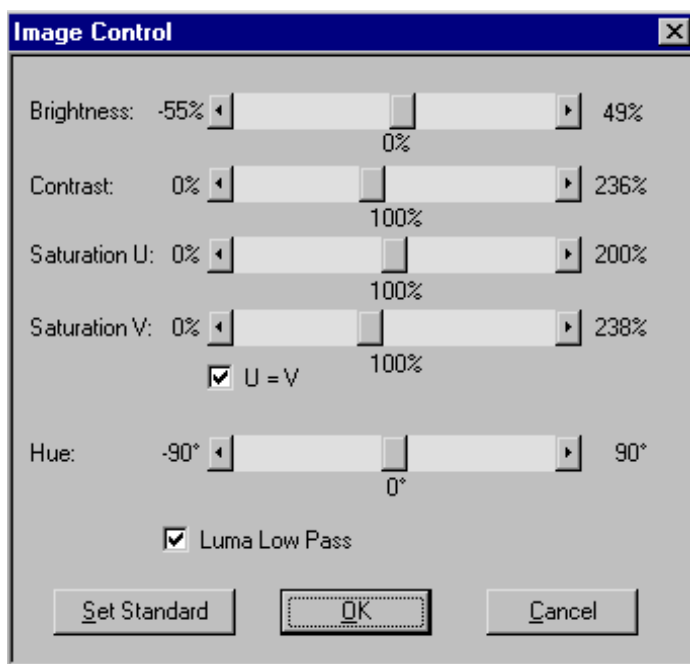


Bild 30: Fenster zur Bildkontrolle

Über Schieberegler lassen sich die Einstellungen von Helligkeit, Kontrast, Farbsättigung und Farbton verändern.

Die Werte werden sofort in den Grabber übertragen, so daß sich die Auswirkung der Justierung sofort im Live-Bild beurteilen läßt.

Über den Regler *Brightness* wird die Helligkeit eingestellt und kann an die Umgebungsbedingungen angepasst werden.

Der Regler *Contrast* verändert den Kontrast im Bild.

Für die Einstellung der Farbsättigung sind zwei Regler vorhanden: *Saturation U* und *Saturation V*. Dies ermöglicht eine getrennte Manipulation der Sättigung im Rot- und im Blauviolett-Bereich. Mit der Kontrollbox *U=V* lassen sich beide Regler zusammenschalten. Auf diese Weise kann man Farbsättigungsänderungen vornehmen, ohne den Farbton zu verändern.

Der Einstellregler für den Farbton, *Hue*, ist nur bei Verwendung des NTSC-Farbsystems sinnvoll. Er dient zur Korrektur von Farbtonverfälschungen durch Phasenfehler auf dem Übertragungsweg. Solche Störungen treten nur beim NTSC-System auf.

Das PAL-System korrigiert Farbtonfehler automatisch, so daß die Veränderung des Farbtonreglers keine Auswirkungen hat. Belassen Sie hier bitte den Regler in der Mittelposition.

Hinweis:

Korrekturen des Farbtons (z.B. Weißabgleich) können Sie bei PAL und NTSC dadurch durchführen, daß Sie die Farbsättigungsregler einzeln verschieben.

(Grundsätzlich ist es aber besser, den Weißabgleich nach Möglichkeit schon an der Kamera vorzunehmen.)

4.3 Weitere Funktionen unter *Image*

- Mit dem Eintrag *Single Image* wird ein Schnappschuß aufgenommen und auf dem Bildschirm dargestellt. In diesem Modus führt der Grabber nur eine Digitalisierung durch.
- Die Bilddarstellung wird durch die Parameter in *Image settings* bestimmt.
- Unter *Live Image* kann – wie zuvor beschrieben - das Livebild auf den Bildschirm gebracht werden. Auch hier bestimmen die Parameter in *Image settings* die Bilddarstellung.
- Mit der Option *Snapshot* können Schnappschüsse während des Livebetriebs erstellt werden.
- Das in diesem Moment gezeigt Bild wird dann in einem eigenen Fenster dargestellt. Es können mehrere Momentaufnahmen gemacht werden.
- Die erscheinenden Fenster werden automatisch durchnummeriert.
- Der Eintrag *Open Image on Start* ermöglicht es, mit den vorher unter *Image Settings* eingestellten Parametern, bei jedem Programmstart automatisch das Livebild der Videoquelle auf dem Bildschirm darzustellen.
- Wenn zusätzlich das Demoprogramm in der Autostart-Gruppe abgelegt wird, kann der Rechner nach dem Hochfahren, ohne Eingreifen des Benutzers, ein Livebild mit den vorher eingestellten Parametern auf den Bildschirm bringen.

4.4 Fadenkreuze einblenden

Wenn Sie Objekte in der Mitte des Bildes zentrieren wollen, können Sie verschiedene Fadenkreuze in das Livebild ein- und ausblenden.

Die Einstellungen dafür befinden sich in der Menuleiste unter *Effects*. Es können auch Kombinationen der Kreuze bzw. alle auf einmal eingeblendet werden.

4.5 Grundeinstellungen

Unter dem Menüpunkt *Options* befinden sich verschiedene Grundeinstellung bezüglich des Grabbers und den arithmetischen Operationen.

Der Eintrag *Basic Settings* beinhaltet folgendes Menü:

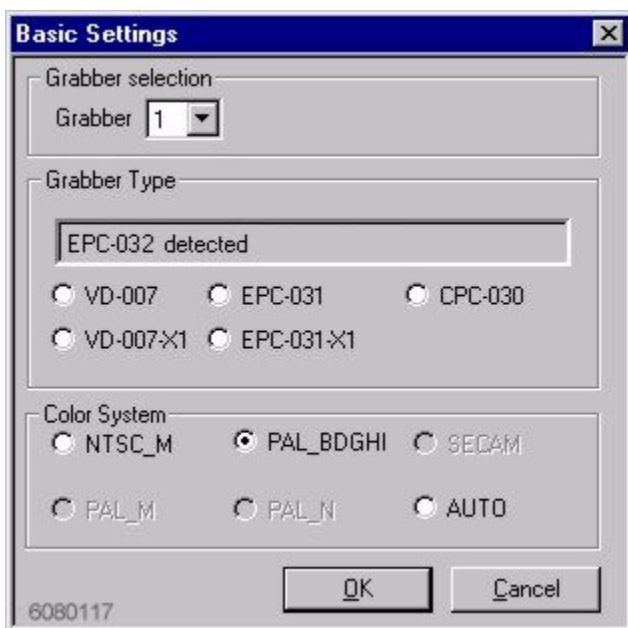


Bild 31: Basic Settings Menü

Sie können mehrere eGrabber-4plus in einem PC/104-System betreiben. Um dem Demoprogramm mitzuteilen, welchen Grabber es ansprechen soll, wählen Sie die entsprechende Nummer im Feld Grabber selection. Ist nur ein Grabber installiert, wird dieser automatisch aktiviert und mit 1 bezeichnet.

Im Bereich *Grabber Type* wird angezeigt, welches Grabbermodell im Rechner installiert ist.

Der eGrabber4*plus* meldet sich mit *EPC-032*, *EPC-032-X1* bzw. *EPC-032-X2*, *EPC-032-X1-X2* (je nach Grabbermodell).

Unter *Color System* wird das Farbsystem, mit dem der Grabber arbeiten soll, eingestellt.

PAL wird hauptsächlich in Europa, NTSC in USA eingesetzt.

Diese Parameter können bei Livebild-Betrieb nicht geändert werden.

Die Einträge *Addition Settings* und *Type Casting Settings* werden später im Zusammenhang mit den Punkten *Add Live Images* und *Arithmetics* im Menüpunkt *Features* erklärt.

4.6 Die Sonderfunktionen

Das Demoprogramm bietet einige Sonderfunktionen, mit denen der Bildinhalt manipuliert und ausgewertet werden kann.

- **Histogramm anzeigen:**

Der Menüpunkt *Histogramm* ermöglicht es, von einem Standbild, das z.B. über die *Snapshot* Funktion erstellt wurde, eine Histogramm zu errechnen.

Ein Histogramm stellt die Farbwert- bzw. Grauwertverteilung dar. Die relative Häufigkeit der einzelnen Intensitätswerte wird über der Helligkeit bzw. der Intensität aufgetragen (*siehe Bild 32*).

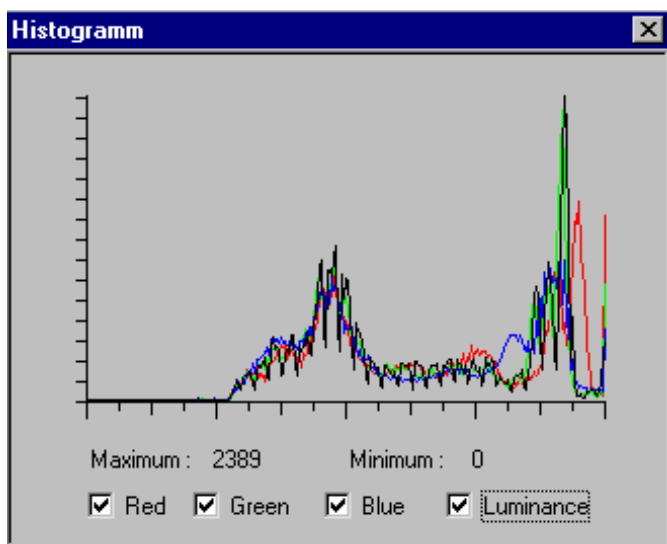


Bild 32: *Histogramm*

Der Wertebereich auf der x-Achse erstreckt sich von 0 bis 255. Die einzelnen darzustellenden Farben können über Checkboxes ausgewählt werden.

Achtung!

Das Histogramm kann nur von einem Standbild, nicht von einem Livebild dargestellt werden. Erzeugen Sie ein Standbild aus einem Livebild, indem Sie die Snapshot-Funktion verwenden.

- **Farben auswerten:**

Der Eintrag *Color Meter* öffnet die Oberfläche in *Bild 33*.

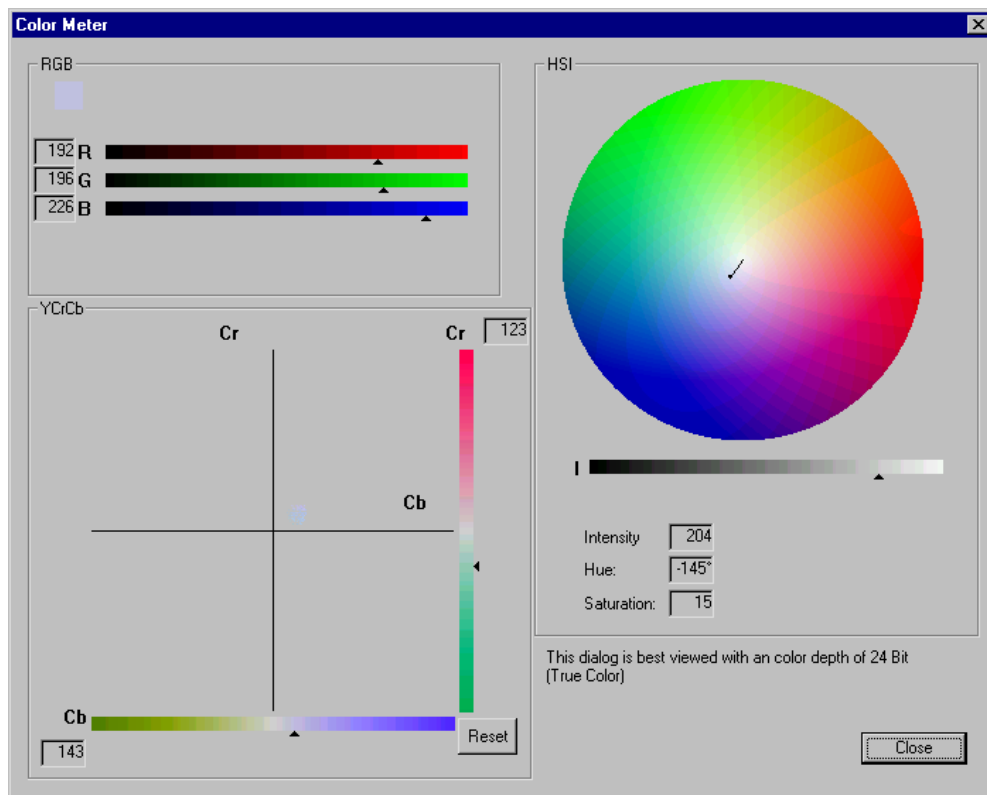


Bild 33: Color Meter

Das Color Meter funktioniert nur in der Livebilddarstellung. In ihm wird der Farbwert des Pixels, das sich in der Bildmitte befindet, in verschiedenen Farbmodellen angezeigt.

Ein kleines im Livebild erscheinendes Fadenkreuz zeigt die Bildmitte an.

Das RGB-Modell zeigt die Farbwerte für Rot, Grün und Blau als Zeiger auf Intensitätsbalken sowie als Zahlenwert an.

Das YCrCb-Modell stellt den Farbwert als Farbbalken sowie zusätzlich in einem Koordinatensystem dar.

Der Farbverlauf kann dadurch über einen längeren Zeitraum beobachtet werden.

Mit dem Button *Reset* wird die Koordinatenanzeige gelöscht und es wird erneut aufgezeichnet.

Das HSI-Modell stellt den Farbwert auf einem Kreis dar. Der Vektor in dem Kreis zeigt durch seine Länge die Sättigung (Saturation), und durch seine Richtung den Farbton (Hue). Die Helligkeit wird auf einem Balken darunter angezeigt.

Die einzelnen Werte werden auch numerisch ausgegeben.

- **Farbbalken einblenden:**

Mit der Option *Color Bars* kann der Grabber getestet werden.

Die Farbbalken werden von der Hardware, nicht vom Demoprogramm generiert. Je nach gewähltem Bildausschnitt werden mehr oder weniger Balken zu sehen sein.

Alle Farbbalken werden bei einer horizontalen Auflösung von ca. 515 Pixeln sichtbar.

- **Arithmetische Operationen an Standbildern:**

Der Menüpunkt *Arithmetics* ermöglicht einfache arithmetische Operationen auf Standbildern (*siehe Bild 31*).

Bilder können pixelweise addiert, subtrahiert, multipliziert oder dividiert werden. Zusätzlich können Konstanten addiert oder multipliziert werden.

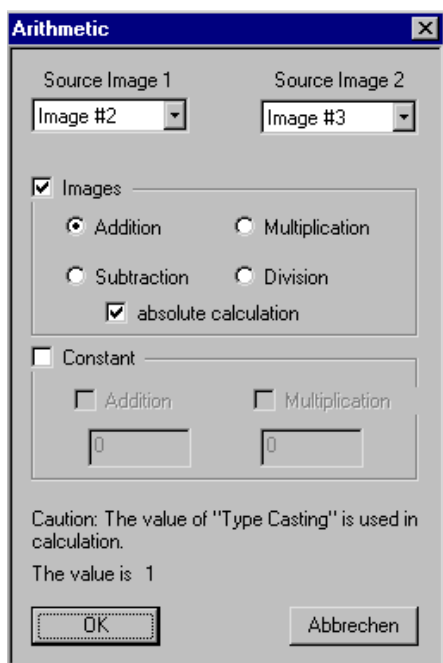


Bild 34: *Arithmetics Menü*

Die zu verarbeitenden Bilder werden unter *Source Image 1* und *2* ausgewählt. Die Nummern hinter *Image#* korrespondieren mit den Nummern der Bildfenster.

Unter dem Eintrag *Images* kann die arithmetische Operation gewählt werden. Ebenfalls kann angegeben werden, ob betragsmäßig gerechnet werden soll (*absolut calculation*).

Bei betragsmäßiger Berechnung können keine negativen Werte entstehen, die eventuell kein sinnvolles oder ein falsches Ergebnis darstellen.

Unter dem Punkt *Constant* kann eine Konstante zu jedem Pixel addiert (Helligkeitsänderung) oder multipliziert (Kontraständerung) werden.

Alle Operationen werden einer Normierung unterzogen.

Dies ist wichtig, da der Wertebereich des Ergebnisses weit außerhalb des Darstellungsbereiches von 0 bis 255 pro Farbkanal liegen kann.

Ergebnisse größer 255 werden auf 255, und Werte kleiner 0 auf 0 gesetzt.

Dabei wird verhindert, daß z.B. bei der Multiplikation weiße Bilder entstehen.

Der Normierungswert wird unter *Options/ Type Casting Settings* gewählt (*siehe Bild 35*).

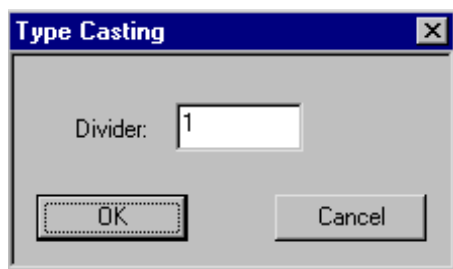


Bild 35: Normierungswert-Wahl

Der aktuelle Wert wird im unteren Fensterteil des *Arithmetic* – Menü angezeigt.

Achtung!

Inkorrekte Einstellung der Normierungsvariable führt zu unbefriedigenden Ergebnissen bei den arithmetischen Operationen (z.B. ganz schwarze oder weiße Bilder).

- Unter *Add Live Images* können bis zu 1000 aufeinanderfolgende Livebilder zu einem Bild summiert werden.
- Unter *Options / Addition Settings* kann die gewünschte Bilderanzahl gewählt werden (*siehe Bild 36*).

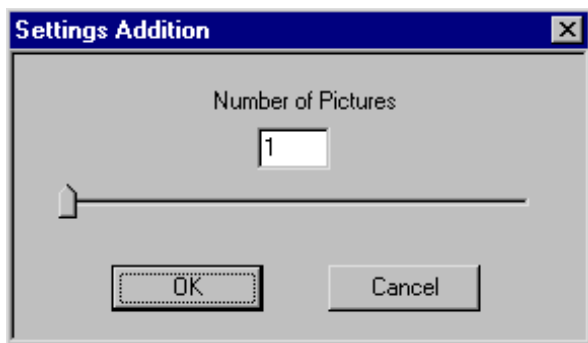


Bild 36: Anzahl der Bilder

So kann z.B. das Bildrauschen einer Aufnahme verringert oder bewegte Gegenstände gegenüber dem Hintergrund reduziert werden.

Nach der Addition wird das darzustellende Bild so normiert, daß die Gesamthelligkeit erhalten bleibt.

Der Vorgang dauert entsprechend der zu addierenden Bilder und Leistungsfähigkeit des Rechners.

Der Fortschritt des Prozesses wird am unteren Fensterrand in Prozent angezeigt.

Achtung!

Die Einstellung der Bildanzahl verändert gleichzeitig den Wert der Normierungsvariablen (Type Casting) so, daß das addierte Bild die gleiche Helligkeit hat wie ein Einzelbild.

Bei anschließender Verwendung der *Arithmetic* – Funktionen muß die Normierungsvariablen eventuell neu eingestellt werden.

- **I/O-Port Test:**

Der Menüpunkt *Test Hardware* beinhaltet den Unterpunkt *I/O Test*.

Es erscheint das Fenster in *Bild 37*.

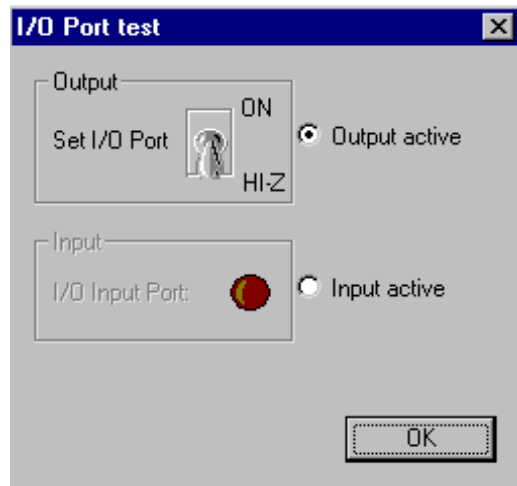


Bild 37: I/O-Test Menü

Mit diesem Dialog ist es möglich, die Funktion des I/O-Ports zu testen. Der Port ist ein Schaltein- oder –ausgang, welcher über einen auf dem Grabber befindlichen Transistor gesteuert wird.

Der Port kann über die Pfostenleiste X3 oder über den Option Port X6 des Grabbers beschaltet werden (*siehe Kapitel 1.6*).

Bei der Wahl *Output active* arbeitet der Portpin als Ausgang. Mit dem Schalter *Set I/O Port* kann der Ausgang aktiviert (ON = Verbindung gegen Masse) oder deaktiviert (HI-Z= hochohmig) werden.

Bei der Wahl *Input active* kann durch die simulierte rote Leuchte *I/O Input Port* beobachtet werden, ob der Port auf ein externes Signal reagiert.

Ist der Pin offen (unbeschaltet) so ist der Eingang nach Wahl dieses Buttons aktiviert. Dies wird durch das Leuchten der roten Lampe signalisiert.

4.7 Bilder speichern, Programm beenden

Unter dem Eintrag *File* können Sie Livebilder (Momentaufnahme), Standbilder bzw. arithmetisch bearbeitete Bilder abspeichern. Mit *Save* oder *Save as* können die Bilder mit der vom Programm vergebenen Indexnummer bzw. mit einem eigenen vergebenen Namen abgespeichert werden.

Die Bilder werden im bmp-Format abgespeichert und können mit jedem Grafikprogramm betrachtet bzw. bearbeitet werden.

Mit *Close* kann das Standbild- sowie das Livefenster nacheinander geschlossen werden.

Mit *Exit* verlassen Sie das Programm.

4.8 Inbetriebnahme unter Linux

Zum Betrieb des pciGrabber-4plus / eGrabber-4plus kann der BTTV-Grabbertreiber verwendet werden. Dieser Treiber ist bereits in den meisten Linux-Distributionen enthalten. Sie können also ohne besonderen Treiber arbeiten.

Sie benötigen dazu die BTTV-Treiberversion 0.7.107 oder höher. Falls in Ihrer Treiberversion diese Kartendefinitionen noch nicht enthalten sein sollten, können Sie sich die neuste Version des Treibers von der Homepage des bttv-Treiber www.bytesex.org herunterladen.

Beachten Sie, daß der BTTV-Treiber noch für den Betrieb mit dem pciGrabber-4plus konfiguriert werden muß. Es erfolgt keine automatische Erkennung der Karte.

Gehen Sie zur Einrichtung der Karte wie folgt vor:

Schritt 1: Sehen Sie bitte zunächst in der Liste unten nach, welche bttv-Kartenummer Ihrer Grabber-Kartenummer zugeordnet ist.

Schritt 2: Öffnen Sie mit einem Texteditor die Datei `etc/modules.conf` und tragen Sie die Kartenummer wie folgt ein:

/etc/modules.conf:

```
....
alias char-major-81 videodev
alias char-major-81-0 bttv
options bttv card=106 ← diese Nr. an Ihre Karte anpassen
...
```

bttv-Kartennummern für PHYTEC-eGrabber-Karten		
Kartennr.	Grabbertyp	S-Video-Eingang
106	EPC-032	Mini-DIN-Buchse
107	EPC-032	Pfostenleiste X3
108	EPC-032-X1	Mini-DIN-Buchse
109	EPC-032-X1	Pfostenleiste X3

Anmerkung: Die Einstellungen gelten auch für die entsprechenden –X2-Varianten

Hinweise:

- Bei der Einstellung der Karte wird festgelegt, ob als S-Videoeingang die Mini-DIN-Buchse oder der Kombi-Eingang verwendet werden soll. Eine spätere Auswahl im Anwendungsprogramm (wie unter Windows) ist nicht möglich, da dies in der bttv-Treiberstruktur nicht vorgesehen ist.
- Zum Testen verwenden Sie am besten das Programm XawTV

Teil 2

Programmierhandbuch

5 Treiber-Software

Dieser Kapitel erklärt, wie Sie den eGrabber-4*plus* aus eigenen Programmen heraus ansprechen können.

Die Treiber-Bibliotheken stellen Ihnen einen Satz von Funktionen zur Verfügung, mit denen Sie die Einstellungen des Grabbers verändern können, Statusinformationen abfragen und die Digitalisierung von Bildern einleiten können.

Softwaretreiber stehen Ihnen für verschiedene Betriebssysteme zur Verfügung.

In diesem Handbuch werden die Treiber für

- Windows '95/98/ME/XP
- Windows NT4.0
- Windows 2000
- DOS

erläutert.

Achtung!

Lesen Sie die *readme.txt* - Datei auf der Installations-CD, um neueste Informationen zu den Treibern und der Verfügbarkeit weiterer Treiber zu erhalten.

Im nächsten Kapitel werden einige technischen Grundlagen des Grabbers und der verwendeten Fernsehnormen erläutert, um Ihnen die Funktionsweise des Grabbers transparent zu machen.

5.1 Technische Grundlagen

5.1.1 Aufbau des eGrabber-4plus

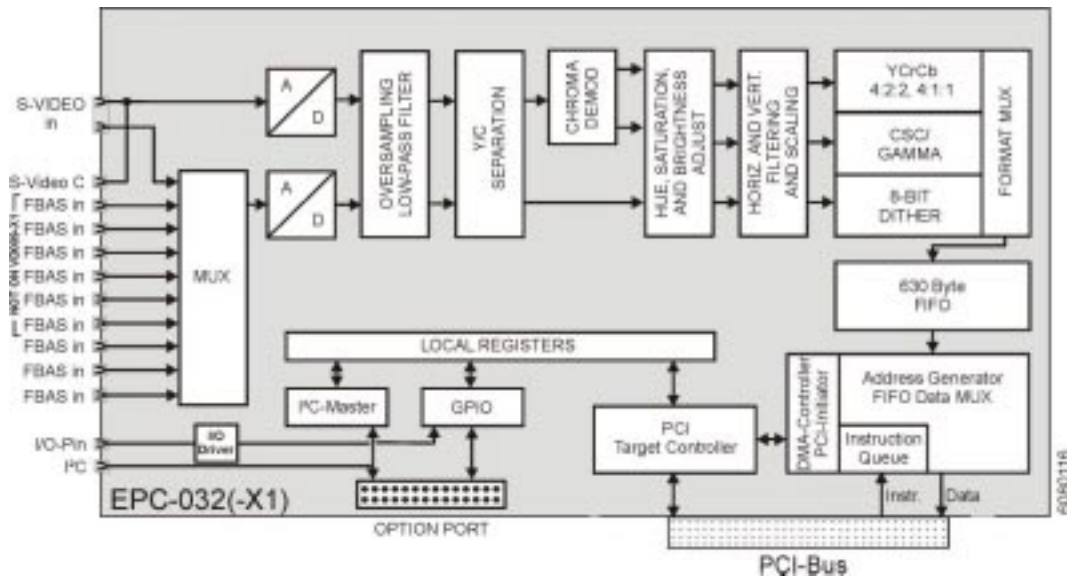


Bild 38: Blockschaltbild eGrabber4plus

Bild 38 zeigt das Blockschaltbild des eGrabber-4plus. Das Composite-Eingangssignal wird zunächst über einen 9:1-Video-multiplexer geführt, den der Anwender über den PCI-Bus steuern kann. Der nachfolgende A/D-Wandler digitalisiert dieses Signal. Als Bildquellen eignen sich alle Geräte, die ein Farb-Videosignal gemäß den CCIR-Definitionen „PAL (B,G,H,I)“ und „NTSC (M)“.

Achtung!

In Deutschland benutzte Bildquellen liefern i.a. PAL-Fernsehsignale. In diesem Manual wird daher durchgehend die Verwendung von PAL-Quellen vorausgesetzt.

Über den *S-VIDEO*-Eingang können Luma- und Chroma-Signal getrennt eingespeist werden (z.B. von einer *S-Video*-Kamera oder einem *S-VHS*-Videorecorder). Für den Farbanteil wird hier ein getrennter A/D-Wandler benutzt, was die Bildqualität verbessert.

An den eGrabber-4*plus* lassen sich selbstverständlich auch Schwarzweiß-Bildquellen anschließen.

Die Verarbeitung von Grauwert-Bildern mit 256 Graustufen ist im Grabber vorbereitet und kann per Software aktiviert werden. Beim Einsatz von Schwarzweiß-Bildquellen können Sie die Bildschärfe verbessern, indem Sie die Farbfalle deaktivieren. Dies ist softwaremäßig möglich.

Nach den A/D-Wandlern folgen Funktionsgruppen, die den Bilddatenstrom in seine Komponenten zerlegen: Hinter dem Chroma-Demodulator liegen die Bilddaten getrennt nach Helligkeit (Y) und Farbanteilen (Cr/Cb) vor. Danach erfolgt die digitale Korrektur von Helligkeit, Kontrast und Farbsättigung und die Skalierung auf die gewünschte Bildgröße und -auflösung.

Im nachfolgenden *Video Format Converter* werden die vom eGrabber-4*plus* bereitgestellten Datenformate erzeugt. Über einen Datenmultiplexer wird das vom Anwender gewünschte Format ausgewählt und dem 630 Byte großen Pufferspeicher (FIFO) zugeführt.

Der FIFO dient als Bindeglied zum nachfolgenden PCI-Bus Interface, welches die Datenübertragung über den PCI-Bus organisiert.

Die Bilddaten werden per DMA direkt in den Arbeitsspeicher des PCs übertragen. Für jedes Halbbild wird ein eigener DMA-Kanal verwendet. Die Übertragung kann vom Anwender sehr flexibel gestaltet werden. Dazu dient eine *pixel instruction list* für jedes Halbbild, die im folgenden als *RISC-Programm* für den im eGrabber-4*plus* integrierten PCI-Controller bezeichnet wird.

Das Prinzip der *pixel instruction list* wird in *Kapitel 5.1.2* näher erläutert.

Über den PCI-Controller wird des weiteren der Zugriff zu den *lokalen Registern* abgewickelt. Diese ermöglichen die Einstellungen der Grabber-Parameter und das Rücklesen der aktuellen Statusinformation. Über diese Register erfolgt auch das Ansprechen der benutzerdefinierten I/O-Leitungen und der in den Grabber integrierten I²C-Schnittstelle.

5.1.2 Videosignal und Digitalisierungsvorgang

Das vom Grabber verarbeitete analoge Norm-Videosignal besteht aus 625 Zeilen, die in zwei Halbbilder (fields) aufgeteilt sind und von der Videoquelle nacheinander geliefert werden. Das erste Halbbild (ungerade / odd) beinhaltet die Zeilen 1 bis 313, das zweite (gerade / even) die Zeilen 314 bis 625. Die Halbbilder sind ineinandergeschachtelt, um im TV-Bild den Flimmereffekt zu verringern. Räumlich gesehen folgt daher auf Zeile 1 die Zeile 314. Soll ein Vollbild dargestellt werden, müssen die beiden nacheinander empfangen Halbbilder, entsprechend *Bild 39*, verschachtelt werden (Fernseher per elektrischer Schaltung, im PC-Graphikspeicher per Software).

Neben einigen Vor- und Nachlaufzeilen beinhaltet das Videosignal noch Prüf- und Datenzeilen, sowie Zeilen mit Videotext-Information, so daß die effektive Bildfläche aus zwei Halbbildern mit je 288 Zeilen besteht.

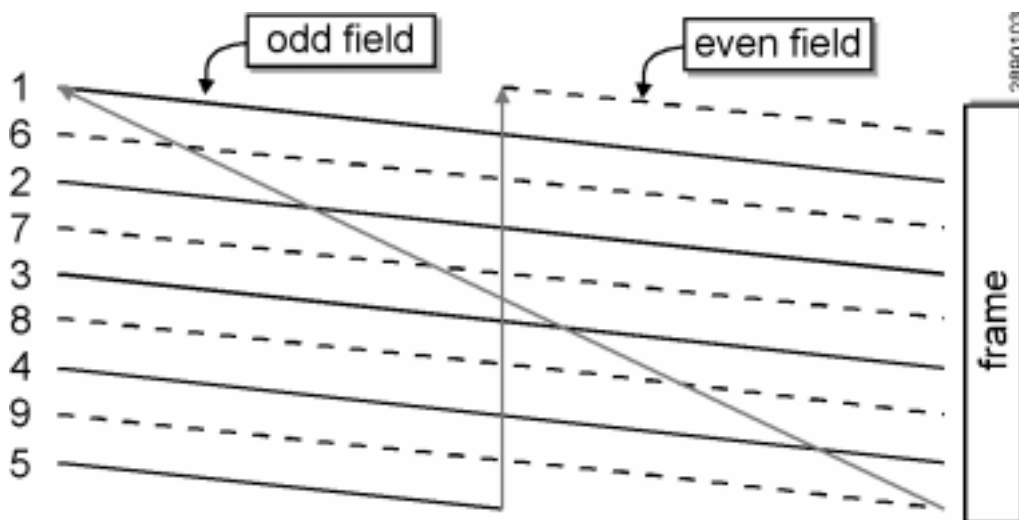


Bild 39: Zeilensprungverfahren (Beispiel mit 9 Zeilen)

Jedes Halbbild wird in 20 ms aufgebaut. In einem Halbbild ist bereits die vollständige Bildfläche zu erkennen, die vertikale Auflösung ist jedoch um die Hälfte reduziert. In vielen Anwendungen ist dies jedoch bereits ausreichend, so daß ein vollständiges Digitalisierungsergebnis bereits nach 20 ms vorliegt. Gegebenenfalls kann die Auflösung in X-Richtung ebenfalls verringert werden, um ein unverzerrtes Bild zu erhalten.

Eine Auflösungsverminderung in X-Richtung kann die Digitalisierung jedoch nicht weiter beschleunigen, da das Zeitraster des Bildaufbaus vorgegeben ist.

Wird die volle TV-Bildauflösung benötigt, so muß der Einzug beider Halbbilder abgewartet werden (40 ms). Die beiden Halbbilder folgen dabei zeitlich nacheinander.

Um die Verschachtelung der Halbbilder zu ermöglichen, ist die letzte Zeile des ungeraden (ersten) Halbbilds auf die Hälfte verkürzt. Demzufolge besteht die erste Zeile des zweiten Halbbilds nur aus der zweiten Zeilenhälfte.

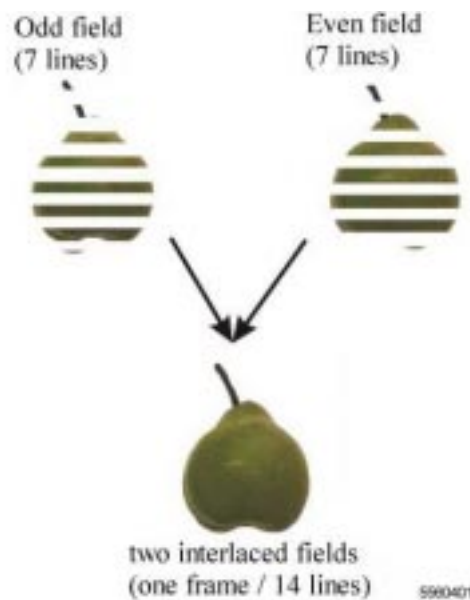


Bild 40: Halb- und Vollbilder

Ein Problem bei der Digitalisierung von TV-Bildern besteht darin, daß ein Objekt bei schneller Bewegung zwischen der Aufnahme des ersten und des zweiten Halbbilds schon ein merkliches Stück vorgerückt ist; die beiden Halbbilder passen also nicht mehr zueinander (Unschärfe / Verzerrungen). Auch aus diesem Grund benutzt man daher häufig - auf Kosten der vertikalen Auflösung - nur ein Halbbild.



Bild 41: Kammefekt bei bewegten Objekten im Vollbild-Modus

5.1.3 Farbübertragung und Farbspeicherung

Farbinformation und Bildhelligkeit werden zur Bildübertragung von den TV-Systemen prinzipiell getrennt. Übertragen wird das Bildhelligkeitssignal (Luma-Signal, Y-Signal) und das Farbdifferenzsignal (Chroma-Signal). Letzteres charakterisiert die Farbe eines Bildpunkts durch die beiden Parameter Farbton und Farbsättigung.

Die Fernsehsysteme reduzieren die Übertragungsbandbreite des Farbsignals gegenüber dem Helligkeitssignal. Die Farbe wird also „unschärfer“ übertragen als die Helligkeit eines Bildpunkts. Dies entspricht anschaulich gesprochen dem Vorgehen eines Zeichners, der die Objektkonturen zunächst mit einer spitzen Feder zieht und das Bild anschließend mit einem groben Pinsel coloriert.

Die Y-Bandbreite beträgt beim PAL (B,G,H,I) - System 5 MHz, die des Chroma-Signals 1,5 MHz.

Das Chroma-Signal wird bei PAL auch als U/V-Signal, bei NTSC als Q/I-Signal bezeichnet. V- bzw. I-Signal charakterisieren Rottöne während das U- bzw. Q-Signal blauvioletten Farbtönen zugeordnet ist. Zusammenfassend spricht man vom Cr/Cb-Signal für (Chroma Red / Chroma Blue).

Über die Werte-Tripel (Y,Cr,Cb) werden Helligkeit und Farbe eines Bildpunkts vollständig definiert. Diese Werte lassen sich einem Bildverarbeitungssystem zur Farberkennung oder -kontrolle ohne weitere Vorverarbeitung zuführen. Häufig wird die Darstellung eines Bildpunkts in den Farbauszügen Rot, Grün und Blau (R,G,B) gewünscht.

Die Umrechnung ist nach CCIR-Empfehlung für PAL durch folgende Matrix definiert:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1,371 & -191,45 \\ 1 & -0,338 & -0,698 & 116,56 \\ 1 & 1,732 & 0 & -237,75 \end{pmatrix} \cdot \begin{pmatrix} Y \\ U \\ V \\ 1 \end{pmatrix}$$

Der eGrabber-4plus kann die Bilder in das RGB-Format umrechnen und als RGB-Farbtripel im Bildspeicher ablegen. Bilder in diesem Format können gut weiterverarbeitet und angezeigt werden.

Zur Speicherung der Bildinformation oder zu Übertragungszwecken ist das YCrCb-Format oftmals günstiger, da es weniger Datenvolumen impliziert. Statt drei Byte pro Pixel sind nur zwei Byte (ein Wort) erforderlich. Die unteren acht Bit spezifizieren dabei die Helligkeit, die oberen acht den Farbanteil (Cr/Cb).

Als Farbanteil wird dabei zu jedem Y-Wert alternierend Cr und Cb geliefert. Man erhält so zu jedem Pixel nur die „halbe“ Farbinformation, entweder den Rot- oder den Blauanteil. Die jeweils fehlende Information kann durch Übernahme des Werts des Nachbarpixels gewonnen werden. Übertragen und gespeichert wird die Farbe also nur in der halben Auflösung der Helligkeit. Da die Farbinformation jedoch vom Fernsehsystem ohnehin bandbreitenreduziert übertragen wird (s.o.), wird durch diese Vorgehensweise kein Verlust verursacht. Dieses Datenformat wird als YCrCb 4:2:2 bezeichnet.

Mit dem ersten Pixel jeder Zeile wird $Y_1, Cr_{1/2}$ geliefert, mit dem zweiten $Y_2, Cb_{1/2}$ usw.

Achtung!

Zur richtigen Erkennung der Farbinformation sind vier aufeinanderfolgende Halbbilder erforderlich. Es ist deshalb nicht möglich, die Videoquelle nur zum Digitalisierungszeitpunkt kurz anzuschließen bzw. eine Videoquelle zu benutzen, die nur einzelne Halbbilder liefert.

Ebenso arbeitet die Halbbildererkennung möglicherweise zunächst nicht korrekt, wenn auf ein anderes, unsynchronisiertes Kamera-signal umgeschaltet wird.

Beim schnellen Wechsel zwischen zwei Bildquellen muß daher damit gerechnet werden, daß ein direkt im Anschluß digitalisiertes Bild fehlerhaft ist. Besser ist es, eine entsprechende Wartezeit einzuhalten.

5.1.4 Datenspeicherung durch DMA und RISC-Programm

Dieser Kapitel befaßt sich eingehender mit der Übertragung der Bilddaten in den Hauptspeicher und der Ablage der Pixel an den vom Anwender gewünschten Adressen.

Wie bereits oben erwähnt, erfolgt der Transfer der Bilddaten durch zwei DMA-Kanäle, einen für das Odd- und einen für das Even-Halbbild. Bezogen auf den PCI-Bus bedeutet das, daß im Augenblick der Bilddigitalisierung der DMA-Controller des eGrabber-4*plus* die Kontrolle über den PCI-Bus beansprucht, also *Master* auf dem Bus wird.

Die Bilddaten werden in Echtzeit auf dem Bus in den Arbeitsspeicher übertragen. Dies wird durch die hohe Datenübertragungsrate des PCI-Busses ermöglicht.

Verzögerungen der Datenübertragung bzw. Zeiträume, in denen der PCI-Bus dem Grabber nicht zur Verfügung steht, (er also die Kontrolle als Master an eine andere Karte abgeben muß) werden mit Hilfe des FIFO-Speichers überbrückt. Dies schließt jedoch längere Untbrechungen des Datentransfers während eines Digitalisierungsvorgangs aus, da dann der FIFO-Speicher überläuft und Teile des Bildes verloren gehen. Im allgemeinen ist dies jedoch unproblematisch.

Mittel zur Kontrolle der Busvergabe sind die Parameter *Maximum_Latency* und *Minimum_Grant* der PCI-Karten. Diese werden vom Treiber automatisch eingestellt.

Im Bedarfsfall müssen sie an die zu übertragende Datenrate, die Systemkonfiguration und die Bus-Performance angepaßt werden.

Der eGrabber-4*plus* weist eine hohe Flexibilität auf, was die Datenspeicherung betrifft. Der Anwender kann Ziel und Format der Bilddaten in gegebenen Rahmen frei bestimmen. Dazu ist ein Mechanismus erforderlich, den kontinuierlichen Bilddatenstrom zu partitionieren und an die gewünschte Speicheradresse zu lenken.

Beim *eGrabber-4plus* wird dies mit Hilfe der *pixel instruction list* gelöst. Dabei handelt es sich um ein RISC-Programm, welches den DMA-Controller entsprechend steuert.

Dieses RISC-Programm muß von der Anwendung entsprechend der Aufgabenstellung, dem Daten- und Bildformat erstellt werden. Es ist also individuell an jedes Problem angepaßt. Dies bedingt in den meisten Fällen, daß das RISC-Programm erst zur Laufzeit des Anwenderprogramms erstellt wird, denn oft sind Parameter, die das RISC-Programm beeinflussen, variabel (z.B. die Bildgröße).

Die mit dem *eGrabber-4plus* gelieferten Software-Treiber erstellen das jeweils passende RISC-Programm automatisch bei der Einstellung der Bildgröße. Dieser Vorgang ist für das Anwenderprogramm transparent. Dennoch sollte dem Programmierer auch bei der Anwendung der Treibersoftware dieser Zusammenhang bewußt sein.

Bild 42 zeigt die Verhältnisse im Überblick. Das Anwenderprogramm benutzt zur Einstellung von Bildgröße und Datenformat die Funktion **set_image()** des Treibers. Der Treiber führt daraufhin zwei Aktionen durch: Er stellt über einen PCI-Bus-Zugriff auf die lokalen Register des Grabbers die Bildgröße im *VideoScaler* ein. Dies sorgt dafür, daß der *eGrabber-4plus* die richtige Bildgröße erzeugt, der Datenstrom also richtig formatiert und am Ende einer Zeile bzw. eines Bildes mit entsprechenden Synchronzeichen versehen wird.

Auf dem gleichen Weg wird der *DataFormatConverter* auf das gewünschte Datenformat eingestellt. In das FIFO gelangt damit ein formatierter Pixeldatenstrom im gewünschten Format (z.B. RGB).

Die zweite Aktion, welche die Treibersoftware durchführt, ist die Erstellung eines auf den Bilddatenstrom angepaßten RISC-Programms, das im Arbeitsspeicher des PCs abgelegt wird. Der DMA-Controller des *eGrabber-4plus* erhält Kenntnis von der Startadresse dieses Programms. Bei der Durchführung eines Digitalisierungsvorgangs holt sich der DMA-Controller sequentiell per DMA die RISC-Befehle aus dem Arbeitsspeicher, verarbeitet sie und speichert den Bilddatenstrom entsprechend diesen Anweisungen.

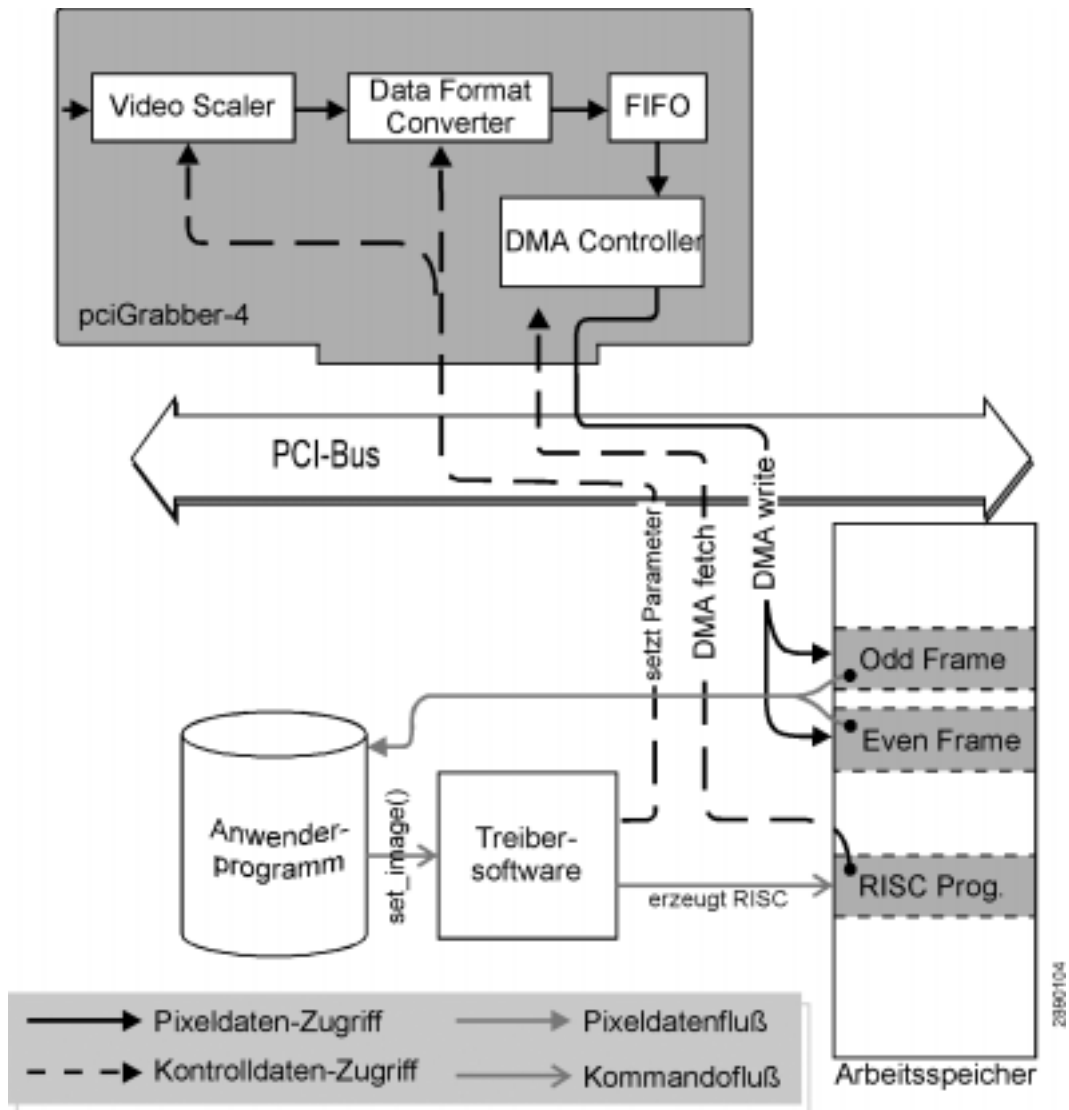


Bild 42: Übersicht über Pixel und Kontrolldatenfluß

Der Bilddatenstrom wird über DMA-Schreibzugriffe direkt in den Arbeitsspeicher geleitet und zwar an die Adressen, die im RISC-Programm spezifiziert sind. Dies sind Adreßbereiche, die das Anwenderprogramm zuvor reserviert hat (z.B. durch die Definition von Arrays).

Es kann sich dabei - wie in *Bild 42* gezeigt - um zwei getrennte Bereiche für Odd- und Even-Bild handeln oder aber um einen Speicherbereich, in dem der *eGrabber-4plus* ein Vollbild entsprechend der oben diskutierten Verzahnung von Odd- und Even-Bild aufbaut. Selektiert werden diese Optionen durch einen Übergabeparameter von **set_image()**, der wiederum die Erstellung des RISC-Programms durch den Treiber entsprechend steuert.

Da das Anwenderprogramm die Adreßbereiche, in denen die Bilddaten gespeichert werden, selbst erzeugt hat, ist ihm die Position dieser Bereiche im Arbeitsspeicher (z.B. als Zeiger darauf) bekannt. Es kann somit direkt und wahlfrei auf diese Daten zugreifen, ohne den Treiber benutzen zu müssen.

Der Treiber stellt Statusinformationen zur Verfügung, die dem Anwenderprogramm mitteilen, wann der Bildaufbau im Speicher beendet ist, die Daten also vollständig vorliegen.

Die realisierte Struktur stellt sicher, daß das Anwenderprogramm schnellstmöglich auf die Daten zugreifen kann. Die Geschwindigkeit ist Echtzeit, bezogen auf das Standard-TV-Format. Für einen Halbbildaufbau bedeutet dies eine Zeitdauer von 20 ms, ein abgeschlossener Vollbildaufbau dauert 40 ms. Zu dieser Zeitspanne, in der die Bildaufnahme erfolgt, sind gegebenenfalls Verzögerungszeiten hinzuzurechnen, um die tatsächliche Zeitdauer zwischen Bildanforderung und Abschluß des Digitalisierungsvorgangs zu bestimmen. Dies ist zunächst die Wartezeit bis zum Beginn des gewünschten Halbbilds.

Wird beispielsweise die Digitalisierung eines Even-Halbbilds gewünscht, von der Kamera jedoch gerade ein angebrochenes Even-Bild geliefert, so vergeht der Rest dieses Bildes und das darauffolgende Odd-Bild, bis wieder der Anfang eines Even-Bilds erkannt wird und die Digitalisierung beginnen kann.

Die Wartezeit beträgt in diesem Fall fast 40 ms (zwei Halbbilder, worst case). Darauf wird das Bild digitalisiert und liegt 20 ms später vollständig vor. Die nächsten 20 ms geschieht in diesem Speicherbereich nichts, da in dieser Zeit ein Odd-Bild angeliefert wird.

Wird eine kontinuierliche Digitalisierung gewünscht, so wird dann in den darauffolgenden 20 ms erneut der Speicherbereich des Even-Bilds mit Bildinformation beschrieben.

Es muß beachtet werden, daß in diesem Zeitraum die alte Bildinformation in diesem Speicherbereich kontinuierlich durch neue ersetzt wird. Dies kann zu Fehlinterpretationen des Bildinhalts durch die Software führen, insbesondere bei bewegten Szenen.

Abschließend sei der Sonderfall der Vollbild-Digitalisierung in einem Speicherbereich betrachtet. Hier tritt der gleiche Effekt auf, jedoch in anderer Ausprägung. Nach 20 ms Digitalisierungszeit liegt die Information des Odd-Bildes vollständig vor, alle ungeraden Bildzeilen sind also definiert, während die geraden undefiniert sind. In den folgenden 20 ms werden die Zeilen des geraden Halbbilds ergänzt. Da sofort darauf wieder ein Odd-Bild geliefert und bei kontinuierlicher Digitalisierung im Arbeitsspeicher abgelegt wird, gibt es in dieser Betriebsart keinen Zeitpunkt, in dem kein Einschreibvorgang in den Speicherbereich erfolgt (von den Austastlücken abgesehen). Es gibt hier also immer einen Punkt (X,Y) an dem alter Bildinhalt in neue Bildinformation übergeht und damit ein Bildversatz entstehen kann.

5.2 Treiber für Microsoft Windows

Bei der Ausführung des Installationsprogramms für das Windows Demoprogramm werden die Dateien entpackt und auf der Festplatte gespeichert. Die Verzeichnisstruktur nach der Installation sieht wie in *Bild 43* dargestellt aus. Der im linken Fensterbereich dargestellte Pfadname kann bei der Installation editiert werden und daher auf Ihrem System anders erscheinen. Die zur Compilierung erforderlichen Library- und Include-Dateien sind in den entsprechend bezeichneten Unterverzeichnissen enthalten.

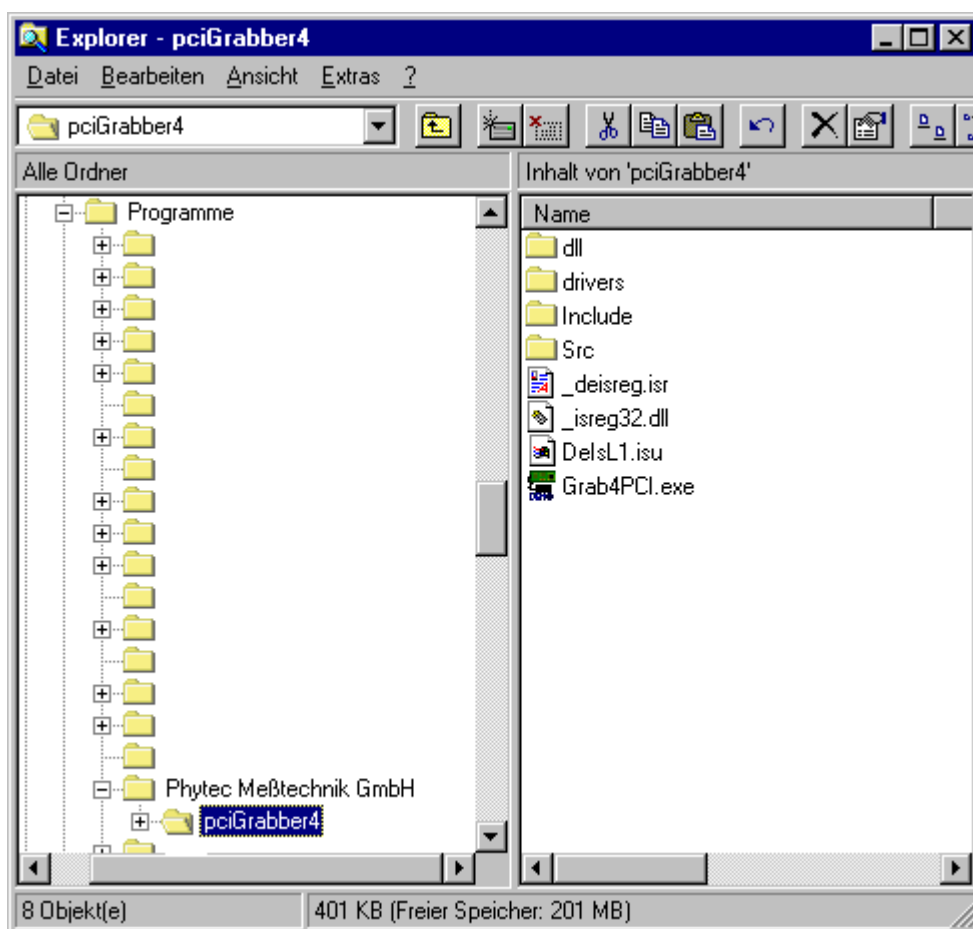


Bild 43: Verzeichnisstruktur der Windows Treiber

5.2.1 Voraussetzungen

Sie können Programme für den eGrabber-4*plus* mit Hilfe verschiedener Entwicklungsumgebungen erstellen.

Die erstellte Applikation ist unter den Betriebssystemen Microsoft Windows`95/98/ME/XP™ und Windows NT4.0/2000 einsetzbar.

Achtung!

Zum Betrieb des eGrabber-4*plus* unter Windows ist es erforderlich, daß der Gerätetreiber und die zugehörige DLL in das Windows-Systemverzeichnis kopiert werden. Zusätzlich muß der Treiber dem System durch Eintrag in die Registry-Table bekannt gemacht werden. Das von uns gelieferte Setup-Programm auf der CD führt diese Vorgänge auf Ihrem Rechner automatisch durch. Daher sind dort die Voraussetzungen für den Betrieb bereits geschaffen.

Um Ihre Applikation auf anderen Rechnern betriebsbereit einzurichten, empfehlen wir Ihnen die Erstellung entsprechender Installations-Routinen

5.2.2 Einrichten des VxD-Treibers für Windows'95™

Der VxD-Treiber dient dazu, physikalisch zusammenhängenden Speicher zu allokkieren, der als Bildspeicher fungiert. Da der eGrabber-4plus per DMA-Zugriff auf diesen Speicher zugreift, ist es erforderlich, daß es sich um physikalisch fixierten und kontinuierlich adressierbaren Speicher handelt.

Die Reservierung eines solchen Speichers ist in Windows'95 nur mittels eines VxD möglich.

Durch den VxD werden ferner die linearen Speicheradressen in physikalische Speicheradressen umgewandelt.

Der VxD-Treiber wird vom Anwenderprogramm nicht direkt angesprochen. Der gesamte Zugriff auf ihn erfolgt durch die DLL.

Zur Auslieferung des Windows'95 Treibers mit Ihrer eigenen Applikation können Sie die von uns mitgelieferte Installations-CD verwenden.

Diese befindet sich auf der Installations-CD im Verzeichnis

PCIGRAB4\DRIVER\WIN95_98.

Sie können die Dateien in diesem Verzeichnis auf eine Diskette kopieren und zusammen mit Ihrer Applikation vertreiben.

Alternativ dazu können Sie auch ein eigenes Installationsprogramm erstellen.

Dazu sind folgende Punkte durchzuführen:

Dem Windows'95-System muß der VxD-Treiber als *statischer Device-Treiber* bekanntgemacht werden. Dazu gehen Sie folgendermaßen vor:

- Zunächst muß die Datei ***PCIGRAB4.VXD*** in das Windows-System-Verzeichnis kopiert werden.
- Als nächstes wird der Treiber in die *Registry-Table* eingetragen: Benutzen Sie dazu das Programm REGEDIT im WINDOWS95 - Verzeichnis. Durchlaufen Sie den Registrierungsschlüsselbaum bis zu der Eintragsgruppe *VxD* (*siehe Bild 44*).



Bild 44: Windows'95 Registrierungs-Editor

Erweitern Sie nun die Schlüsselgruppe *VxD* um den Eintrag „Grab4PCI“, indem Sie über den Menüpunkt „*Bearbeiten - Neu - Schlüssel*“ einen neuen Schlüssel erzeugen und ihm den Namen „Grab4PCI“ zuweisen, wie in *Bild 45* gezeigt.

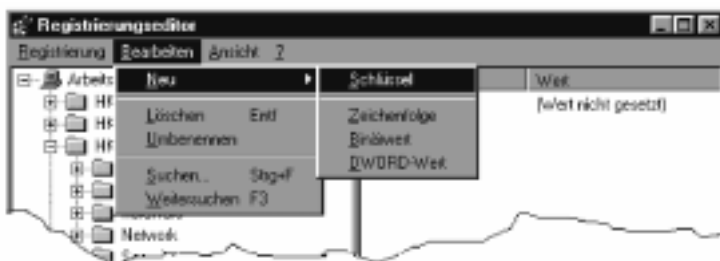


Bild 45: Hinzufügen eines *VxD*-Eintrages

Jetzt konfigurieren Sie die neue Schlüsselgruppe: Markieren Sie den Eintrag „Grab4PCI“ wie in *Bild 46*.

Wählen Sie aus *Bearbeiten/Neu* die Option „Zeichenfolge“. Innerhalb des Schlüssels „Grab4PCI“ wird ein Texteintrag erstellt, der die Bezeichnung „Neuer Wert #1“ hat. Ändern Sie diesen in „StaticVxD“. Klicken Sie dann mit der rechten Maustaste auf den erstellten Eintrag und wählen *Ändern* aus. In dem nun erscheinenden Dialog tragen Sie in dem Feld *Wert* die Zeichenfolge „pciGrab4.VxD“ ein.

Anschließend wählen Sie aus *Bearbeiten/Neu* den Menüpunkt *Binärwert* und erzeugen analog zu oben den binären Eintrag „Start“ mit dem Wert „00“. Das Endergebnis muß so wie in *Bild 46* dargestellt aussehen.

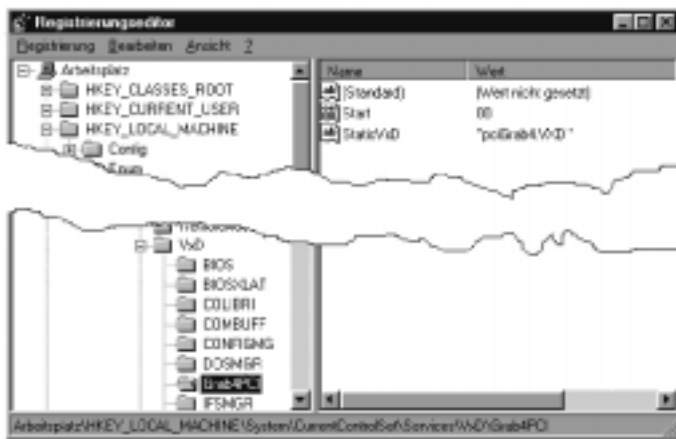


Bild 46: Konfigurieren des VxD

- Da Windows'95 die Werte in der Registry-Table erst bei einem Neustart übernimmt, sollten Sie jetzt das System neu starten.

Achtung!

Bei der Deinstallation des Anwenderprogramms sollte in jedem Fall der VxD-Treiber entfernt werden. Dazu ist er aus der Registry-Table auszutragen und aus dem System-Verzeichnis zu löschen.

Der VxD-Treiber belegt für den eGrabber-4plus einen Arbeitsspeicherbereich von ca. 1,2 MByte, der anderen Applikationen nicht zur Verfügung steht.

Achtung!

Bitte gehen Sie bei der Änderung des Registry-Eintrags mit der gebotenen Sorgfalt vor, da Sie bei fehlerhaften Änderungen die Konfiguration zerstören können. Dies kann den vollständigen Verlust der Funktionsfähigkeit Ihres Windows'95-Systems zur Folge haben. Für den Endanwender sollten Sie ein entsprechendes Install- und Uninstall-Programm vorsehen, das diese Aufgaben automatisiert.

5.2.3 Einrichten des Gerätetreibers für Windows NT4.0

Wie unter Windows`95 dient der Gerätetreiber dazu, physikalisch zusammenhängenden Speicher zu allokkieren, der als Bildspeicher fungiert. Der Treiber erlaubt außerdem den Zugriff auf die Register des eGrabber-4*plus*.

Zur Auslieferung des Windows NT4.0 Treibers mit Ihrer eigenen Applikation können Sie die von uns ausgelieferte Installationsdiskette verwenden.

Diese befindet sich auf der Installations-CD im Verzeichnis **PCIGRAB4\DRIVER\WINNT40**. Sie können die Dateien in diesem Verzeichnis auf eine Diskette kopieren und zusammen mit Ihrer Applikation vertreiben.

Alternativ dazu können Sie auch ein eigenes Installationsprogramm erstellen. Dabei sind folgende Punkte zu beachten:

Dem Windows NT4.0-System muß der Gerätetreiber durch einen Eintrag in der Registry bekanntgemacht werden. Dazu gehen Sie folgendermaßen vor:

- Zunächst muß die Datei **PCIGRABBER4.SYS** in das Verzeichnis **<Windows>\System32\drivers** kopiert werden.

- Als nächstes wird der Treiber in die *Registry-Table* eingetragen: Benutzen Sie dazu das Programm REGEDIT im WINDOWSNT - Verzeichnis. Durchlaufen Sie den Registrierungsschlüsselbaum bis zu der Eintragsgruppe *Services* (siehe Bild 47).

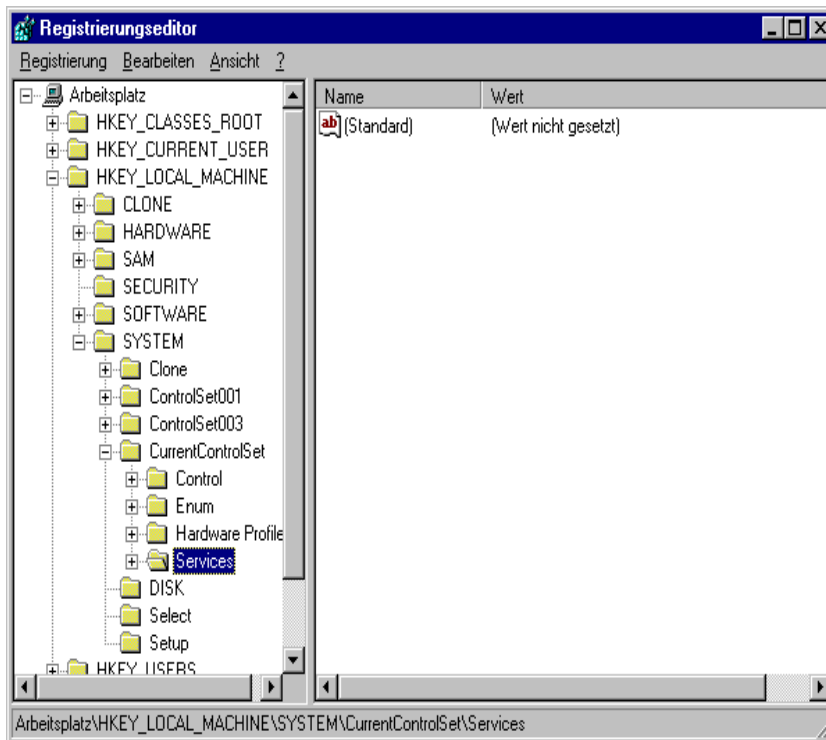


Bild 47: Windows NT Registrierungs-Editor

Erweitern Sie nun die Schlüsselgruppe *Services* um den Eintrag „pciGrabber4“, indem Sie über den Menüpunkt *Bearbeiten/Neu/Schlüssel* einen neuen Schlüssel erzeugen und ihm den Namen „pciGrabber4“ zuweisen, wie in Bild 48 gezeigt.

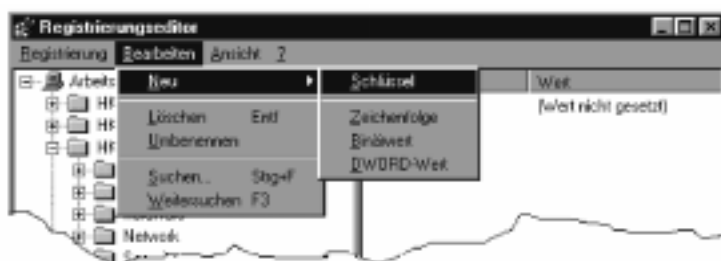


Bild 48: Hinzufügen eines Gerätetreiber-Eintrages

Jetzt konfigurieren Sie die neue Schlüsselgruppe: Markieren Sie den Eintrag „pciGrabber4“ wie in Bild 48.

Wählen Sie aus *Bearbeiten/Neu* den Menüpunkt *DWORD-Wert*. Innerhalb des Schlüssels „*pciGrabber4*“ wird ein Eintrag erstellt, der die Bezeichnung „*Neuer Wert #1*“ hat.

Ändern Sie diesen in „*Start*“. Klicken Sie dann mit der rechten Maustaste auf den erstellten Eintrag und wählen *Ändern* aus. In dem nun erscheinenden Dialog tragen Sie in dem Feld *Wert* die Zahl 2 ein. Wählen Sie aus *Bearbeiten/Neu* die Option *DWORD-Wert*. Ändern Sie die Bezeichnung in „*Type*“ und geben Sie dem Eintrag den Wert 1.

Anschließend wählen Sie aus *Bearbeiten/Neu* den Menüpunkt *DWORD-Wert* und erzeugen analog zu oben den DWORD Eintrag „*ErrorControl*“ mit dem Wert „1“. Das Endergebnis muß so wie in *Bild 49* dargestellt aussehen.

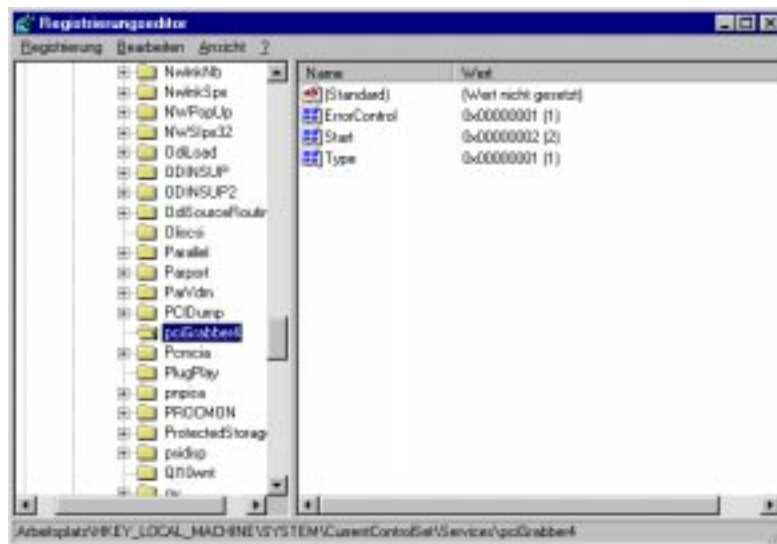


Bild 49: Konfigurieren des Treibers

Nach einem Neustart des Rechners wird jetzt der Treiber automatisch beim Start von Windows NT geladen.

Achtung!

Bei der Deinstallation des Anwenderprogramms sollte in jedem Fall der Gerätetreiber entfernt werden. Dazu ist er aus der Registry-Table auszutragen und aus dem System-Verzeichnis zu löschen.

Der Treiber belegt für den eGrabber-4*plus* einen Arbeitsspeicherbereich von ca. 1,2 MByte, der anderen Applikationen nicht zur Verfügung steht.

Achtung!

Bitte gehen Sie bei der Änderung des Registry-Eintrags mit der gebotenen Sorgfalt vor, da Sie bei fehlerhaften Änderungen die Konfiguration zerstören können. Dies kann den vollständigen Verlust der Funktionsfähigkeit Ihres Windows NT-Systems zur Folge haben. Für den Endanwender sollten Sie ein entsprechendes Install- und Uninstall-Programm vorsehen, das diese Aufgaben automatisiert.

5.2.4 Einrichten des Gerätetreibers für Windows'98™ und Windows 2000 / XP

Wie unter Windows'95 dient der Gerätetreiber dazu, physikalisch zusammenhängenden Speicher zu allokatieren, der als Bildspeicher fungiert. Der Treiber erlaubt außerdem den Zugriff auf die Register des eGrabber-4plus.

Die Reservierung eines solchen Speichers ist in Windows'98 und Windows 2000 nur mittels eines Gerätetreibers möglich.

Durch den Treiber werden ferner die linearen Speicheradressen in physikalische Speicheradressen umgewandelt.

Der Treiber wird vom Anwenderprogramm nicht direkt angesprochen. Der gesamte Zugriff auf ihn erfolgt durch die DLL.

Zur Auslieferung des Windows'98/2000 Treibers mit Ihrer eigenen Applikation können Sie die von uns ausgelieferte Installationsdiskette verwenden.

Diese befindet sich auf der Installations-CD im Verzeichnis

PCIGRAB4\DRIVER\WIN2k_98 .

Sie können die Dateien in diesem Verzeichnis auf eine Diskette kopieren und zusammen mit Ihrer Applikation vertreiben.

5.2.5 Anwendung der DLL

Aufgabe der DLL ist die Kommunikation zwischen Anwenderprogramm und eGrabber-4plus. Über die DLL können die Einstellungen des Grabbers vorgenommen und Digitalisierungsvorgänge gesteuert werden. Sie ermöglicht weiterhin den Zugriff auf die digitalisierten Bilddaten im Arbeitsspeicher.

Achtung!

Die DLL wird nicht in das Anwenderprogramm eingebunden, sondern erst zur Laufzeit aufgerufen. Sie muß daher zur Programmlaufzeit im Windows-Systemverzeichnis vorhanden sein.

Zusätzlich zur *GR4CDLL.DLL* sind folgende weitere DLLs zum Betrieb erforderlich:

- **MSVCRT.DLL**
- **CTL3D32.DLL**
- **MFC42.DLL**

Die DLL wird dynamisch eingebunden. Dazu stellt Windows verschiedene API-Funktionen zur Verfügung. Zum Laden der DLL wird die Funktion **LoadLibrary(...)** benutzt. Man erhält einen Handle auf die DLL zurück. Zur Ermittlung der Einsprunganadressen der verschiedenen DLL-Funktionen wird die API-Funktion **GetProcAddress(...)** verwendet. Die Freigabe der DLL am Programmende wird durch Aufruf der Funktion **FreeLibrary(...)** vorgenommen.

Weitere Informationen können Sie aus der Dokumentation Ihrer Entwicklungsumgebung oder den im beiliegenden SDK befindlichen Beispiel-Quellen entnehmen.

5.2.6 Anwenden der Windows'95/98 / Windows NT4.0 / Windows 2000 / XP – DLL

Zur Verwendung der DLL *Gr4CDLL.DLL* in eigenen Applikationen muß der Softwareentwickler zunächst für jede Funktion, die in der Applikation verwendet wird, einen Funktionszeiger deklarieren.

Beispiel:

- zu verwendende Funktion: **WORD Get_Error(void)**
- Deklaration des Funktionszeigers:

```
WORD (PASCAL * lpfn_GetError)(void);
```

Um den Bezug zwischen dem Funktionszeiger und der DLL herzustellen, wird die Funktion **GetProcAddress(...)** benutzt.

Beispiel:

```
lpfn_GetError = (WORD(PASCAL *)(void))  
                GetProcAddress(handle, „Get_Error“);
```

Nun kann die Funktion wie folgt aufgerufen werden:

```
WORD Fehlerstatus;  
...  
Fehlerstatus = lpfn_GetError();
```

Achtung!

Prüfen Sie **unbedingt** den Wert des Funktionszeigers (Rückgabewert von `GetProcAddress`) auf `= NULL`, um sicherzustellen, daß die auf dem (Anwender-) Rechner installierte Treiber-Version die zu verwendende Funktion unterstützt und Sie einen gültigen Handle zurückerhalten haben!

5.2.7 Programmierung unter Delphi

Um die Funktionen der DLL in Delphi bekanntzumachen, muß eine entsprechende *Unit* definiert werden. Beachten Sie dabei die Angabe der richtigen Aufrufkonvention, um die Kompatibilität der DLL sicherzustellen. Definieren Sie Funktionen vom Typ `stdcall`. Bei falscher Deklaration können ansonsten Stacküber- bzw. unterläufe auftreten, die zu einer Schutzverletzung führen.

Im folgenden ist eine Beispiel-Unit definiert:

```
unit grab4dll;

interface

{ Die Aufrufkonvention 'stdcall' legt die Reihenfolge der
Parameterübergabe auf dem Stack fest und signalisiert Delphi, daß
die gerufene Funktion den für die Parameter benötigten
Stackbereich selbst wieder freigibt }

function Grab4_Get_Error: word;
    stdcall; external 'gr4cdll.dll' name 'Get_Error';

function Grab4_Max_Device_Number: word;
    stdcall; external 'gr4cdll.dll' name
'Max_Device_Number';

function Grab4_Data_Present(nDevNo: word): word;
    stdcall; external 'gr4cdll.dll' name 'Data_Present';

function Grab4_GetPictureBufferAdress(nDevNo: word;
    dwBitsSize: Cardinal):
cardinal;
    stdcall; external 'gr4cdll.dll' name 'Data_Present';

procedure Grab4_Initialize(nDevNo: word);
    stdcall; external 'gr4cdll.dll' name 'Initialize';

procedure Grab4_Set_Channel(nDevNo, nChannel: word);
    stdcall; external 'gr4cdll.dll' name 'Set_Channel';

procedure Grab4_Start_Grabber(nDevNo: word);
    stdcall; external 'gr4cdll.dll' name 'Start_Grabber';

procedure Grab4_Stop_Grabber(nDevNo: word);
    stdcall; external 'gr4cdll.dll' name 'Stop_Grabber';
```

```
procedure Grab4_Set_Image(nDevNo: word;
                        nOhpos, nOvpos, nOhsize, nOvsize,
                        nOppl, nOlines, nOColformat :word;
                        nEhpos, nEvpos, nEhsize, nEvsiz,
                        nEppl, nElines, nEColformat :word;
                        nColsystem:word;
                        nInterlaced:word;
                        nSingleShot:word);
    stdcall; external 'gr4cdll.dll' name 'Set_Image';

const
    NTSC_M:    word    = 0;
    PAL_BDGHI: word    = 1;
    SECAM:     word    = 2;
    PAL_M:     word    = 3;
    PAL_N:     word    = 4;
    AUTO:      word    = 5;

    RGB32:     word    = 0;
    RGB24:     word    = 1;
    RGB16:     word    = 2;
    RGB15:     word    = 3;
    YUY2:      word    = 4;
    BtYUV:     word    = 5;
    Y8:        word    = 6;
    RGB8:      word    = 7;

implementation

{ DLL Funktionen }

end.
```

5.2.8 Beschreibung der in der DLL vorhandenen Funktionen

Über die Funktionen der DLL steuern Sie alle Vorgänge im eGrabber-4*plus* und können den aktuellen Status sowie die eingestellten Werte zurücklesen. Im folgenden werden diese Funktionen ausführlich beschrieben.

Zur besseren Übersicht sind die Funktionen in fünf Gruppen aufgeteilt. Die Nummer der jeweiligen Gruppe ist der Funktionsbeschreibung in einem schwarzen Kreis vorangestellt.

Die Funktionen sind wie folgt klassifiziert:

❶ Routinen zur Initialisierung / Hardwareabfrage

Diese Gruppe beinhaltet Routinen, die Sie ein Mal vor der Verwendung des Grabbers aufrufen müssen, damit der Grabber korrekt arbeitet. Des weiteren finden Sie hier zwei Funktionen, die Ihnen Auskunft über die installierte Hardware und deren Fähigkeiten geben.

❷ Routinen, die den Grabber zur Bildaufnahme einstellen

Funktionen aus dieser Gruppe stellen den Grabber auf die angeschlossene Bildquelle (Kamera) ein und legen fest, wie das vom Grabber erzeugte Ergebnisbild im Speicher aussieht (Bildgröße, Farbformat usw.). Sie sollten für jede Funktion aus dieser Gruppe prüfen, ob sie verwendet werden muß und welche Parameter zutreffend sind. Gegebenenfalls werden diese Routinen im Programmverlauf mehrfach aufgerufen (z.B. wenn der Eingangskanal umgeschaltet werden soll oder die Bildgröße verändert wird).

❸ Routinen zur Durchführung und Kontrolle des Grabbvorgangs

Mit diesen Funktionen starten Sie die Bild-Digitalisierung, überwachen den Grabbvorgang und beenden die Digitalisierung.

④ Routinen zur Einstellung von Bildparametern

Funktionen aus dieser Gruppe ermöglichen die Einstellung von Parametern wie Bildhelligkeit, Kontrast, Farbsättigung usw. Sie müssen nicht verwendet werden, können aber jederzeit aufgerufen werden, um das Ergebnisbild an die Bedürfnisse anzupassen.

⑤ Routinen zur Steuerung der Erweiterungsports

Unter diese Kategorie fallen alle Funktionen, die nicht direkt mit der Bildaufnahme zu tun haben, sondern die weiteren Features des Grabbers wie I/O-Port, I²C-Schnittstelle usw. betreffen. Diese Funktionen müssen nur aufgerufen werden, wenn ein entsprechendes Grabber-Feature verwendet wird.

Die meisten Funktionen sind in der Windows- und in der DOS-Treiberversion identisch. Aufgrund des Treibermodells gibt es jedoch einige wenige Unterschiede. Um die Portierung von Programmen zu erleichtern, werden folgende Symbole verwendet:

⇔ Der Funktionsaufruf ist unter DOS und Windows gleich
Beachten Sie jedoch, daß sich die Variablentypen betriebssystembedingt unterscheiden können.

☒ Diese Funktion ist Windows-spezifisch

DOS Diese Funktion ist DOS-spezifisch

Achtung!

In allen nachfolgend beschriebenen Routinen wird der Parameter `nDevNo` verwendet. Dieser Parameter identifiziert den gewünschten eGrabber-4plus, wenn sich mehrere eGrabber-4plus im System befinden. Die Anzahl der vorhandenen eGrabber-4plus kann mit der Funktion `Max_Device_Number()` bestimmt werden.

Kompatibilität zum eGrabber-2

Der Treiber ist grundsätzlich abwärtskompatibel zum eGrabber-2 (EPC-031 bzw. EPC-031-X1). Programme, die für den eGrabber-2 geschrieben wurden, arbeiten also auch mit dem eGrabber-4*plus*.

Um die neuen Funktionen des eGrabber-4*plus* nutzen zu können, besitzt der Treiber einige neue oder anders gestaltete Funktionen.

Funktionen, die nicht zu alten Treiber-Versionen für den eGrabber-2 kompatibel sind, sind im folgenden mit einem Stern (☆) gekennzeichnet.

Wenn Sie eine bestehende Applikation mit den neuen Features des eGrabber-4*plus* ausstatten möchten, beachten Sie bitte die mit ☆ gekennzeichneten Funktionen.

Diese Applikationen arbeiten grundsätzlich auch mit dem eGrabber-2 zusammen, allerdings können manche Funktionen nicht genutzt werden, weil die entsprechenden Hardware-Voraussetzungen nicht erfüllt sind. In jedem Fall sollten Sie mit einer neuen Applikation auch die neue Treiberversion verwenden.

Kompatibilität zum pciGrabber-4*plus*

Der eGrabber-4*plus* ist voll kompatibel zum pciGrabber-4*plus* (VD-009 / VD-009-X1). Applikationen, die für den pciGrabber-4*plus* geschrieben wurden, laufen uneingeschränkt auch auf dem eGrabber-4*plus*.

⇔ ● **Fehlermeldungen auswerten**

WORD Get_Error(void);

Returnwert:	0 = kein Fehler aufgetreten
	1 = Device Nummer nicht gefunden
	2 = Register nicht vorhanden
	3 = Initialisierung fehlgeschlagen
	4 = Grabber nicht gefunden
	5 = unbekannter Parameterwert
	6 = nicht unterstützt
	7 = neuere Treiberversion wird benötigt (Update)
	8 = kein PHYTEC-Grabber gefunden
	9 = kein Acknowledge
	10 = ungültige Adresse
	11 = Schreiben nicht möglich

Jede Ausführung einer Treiberfunktion sollte auf ihren Erfolg überprüft werden. Dazu dient die Funktion **Get_Error**. Direkt nach der Ausführung einer Funktion wird die treiberinterne Fehlervariable auf den aktuellen Status gesetzt.

Diese Variable wird über **Get_Error** dem Anwenderprogramm zugänglich gemacht, das auf diese Weise auf auftretende Fehler reagieren kann.

Die Überprüfung der Fehlervariable ist so lange möglich, bis ein neuer Aufruf einer Treiberfunktion erfolgt. Dadurch wird der Fehlerstatus auf das Resultat des neuen Aufrufs gesetzt.

❶ Versionsnummer der DLL bestimmen

DWORD GetVersionNumber(void);

Rückgabewert: Versionsnummer der Grab4CDLL
HighWord: Major_Version_Number
LowWord: Minor_Version_Number

Über sie kann die Versionsnummer der geladenen Grabber-DLL ermittelt werden.

Diese Funktion ist nur in der Windows-DLL, nicht unter DOS vorhanden.

Hinweis:

Prüfen Sie die Versionsnummer und stellen Sie bei Anwendungen für den eGrabber-4*plus* sicher, daß die Major-Version-Number größer/gleich 5 ist.

⇔ **1 Anzahl der vorhandenen eGrabber-4plus bestimmen**

WORD Max_Device_Number();

Rückgabewert: Anzahl der gefundenen eGrabber-4plus

Diese Funktion bestimmt, wieviele eGrabber-4plus sich im Rechner befinden. Dies ist erforderlich, weil PCI-Geräte nicht über Jumper oder ähnliches vom Anwender konfiguriert werden, sondern durch das PCI-BIOS automatisch einen Adreßbereich zugewiesen bekommt. Über das PCI-BIOS kann dann dieser Adreßbereich für diese bestimmte Karte ermittelt werden. Befinden sich mehrere gleiche Karten im System, so werden deren Adressen nacheinander geliefert (*siehe auch Kapitel 1.4, "Adressen und Ressourcen"*).

Der Anwender braucht sich bei Verwendung des Treibers nicht um Adressen und Adreßbereiche zu kümmern. Diese werden treiberintern in *Gerätenummern* (*device number = nDevNo*) umgesetzt.

Jede im System befindliche eGrabber-4plus - Karte erhält im Treiber eine eindeutige Gerätenummer zugeordnet. Welche Karte welcher Nummer zugeordnet wird, ist nicht unbedingt vorhersagbar, da dies von der Topologie des PCI-Busses und der Funktion des BIOS abhängt.

Damit die verschiedenen eGrabber-4plus unabhängig voneinander über den Treiber angesprochen werden können, wird die Gerätenummer als Parameter bei jedem Funktionsaufruf übergeben. Die Funktion **Max_Device_Number()** wird eingesetzt, um herauszufinden, wieviele eGrabber-4plus sich im Rechner befinden. Es wird die höchste zulässige Gerätenummer zurückgegeben. Dies ist gleichzeitig die Anzahl der Grabber im System, da die niedrigste Gerätenummer = 1 ist. Wird der Wert 0 zurückgeliefert, so hat das PCI-BIOS keinen eGrabber-4plus erkannt.

Achtung!

Für nDevNo dürfen im folgenden nur Werte

$0 < nDevNo \leq \text{Max_Device_Number}()$

angegeben werden.

⇔ ❶ Grabber und Treiber nach dem Einschalten initialisieren

void Initialize(WORD nDevNo);

Diese Routine initialisiert den Grabber und die Treibersoftware nach dem Einschalten. Sie **muß** vor dem ersten Zugriff auf eine Grabberkarte einmal aufgerufen werden. Die Initialisierung muß für jeden Grabber, der benutzt werden soll, durchgeführt werden. Das bedeutet, daß **Initialize** mehrfach mit allen zulässigen Werten für `nDevNo` aufgerufen werden muß.

⇔ ❷ Informationen über die installierten Grabber abfragen

SHORT Read_GrabberInfo(WORD nDevNo, WORD wInfoType)

`wInfoType`: spezifiziert, welche Grabbereigenschaft abgefragt wird

Returnwert: Wert der spezifizierten Eigenschaft

Damit sich die Applikation optimal an die im Rechner installierte Grabber-Hardware anpassen kann, ist es mit dieser Funktion möglich, Informationen über die Hardware abzurufen.

So kann z.B. bestimmt werden, wieviele Eingangskanäle der Grabber unterstützt und der Kanalauswahl-Dialog kann eine entsprechende Bereichsprüfung vornehmen.

Die Eigenschaften werden grundsätzlich als numerischer Wert (vom Typ `WORD`) zurückgeliefert. Die Bedeutung der Schlüsselnummern können Sie aus den Einträgen dem Header-File entnehmen.

Mit dem Parameter `wInfoType` wird ausgewählt, welche Information abgefragt werden soll. Die Bedeutung des Parameters kann ebenfalls aus der Header-Datei ersehen werden. Ist ein Parameter nicht definiert, so gibt die Funktion den Wert -1 zurück. Der Fehlerstatus ist in diesem Fall `6 = „NOT_SUPPORTED“`.

Folgende Parameter können beispielsweise abgefragt werden:

GRABBER_TYPE : Spezifiziert die Typennummer des Grabbers. Der Rückgabewert ist numerisch, mittels der Header-Datei kann die Typenbezeichnung bestimmt werden. Zum Beispiel:

`Read_GrabberInfo(1, GRABBERTYPE) = 0x03`
so ist Grabber Nr. 1 ein EPC-032.

MAX_CHANNEL : Gibt die Anzahl der Composite-Eingangskanäle zurück. Beispielsweise für EPC-032 = 3, für EPC-032-X1 = 3.

Hinweis:

Die abfragbaren Features können sich mit neueren Kartenversionen erweitern. Welche Informationen zur Verfügung stehen, ist in der Header-Datei ersichtlich.

⇔ **1 Grabbername als Klartextstring lesen**

**WORD Read_OrderCode (WORD nDevNo,
 unsigned char* sCodeString,
 DWORD dwSizeOfString)**

***sCodeString:** Zeiger auf einen zuvor deklarierten String (25 Bytes min.), in das die Funktion den Grabbernamen einschreibt.

dwSizeOfString: Größe des reservierten Arrays

Returnwert: Errorcode

Mit dieser Funktion kann die Typenbezeichnung des Grabbers im Klartext ausgelesen werden. Die Übergabe des Namens erfolgt in einem nullterminierten String. Dazu muß zuvor ein Character Array reserviert werden und ein Zeiger auf dieses Array in **sCodeString* übergeben werden. Die verfügbare Größe des Arrays wird der Funktion im Parameter *SizeOfString* mitgeteilt. Sie sollte mindestens 25 Zeichen betragen.

Falls die Typenbezeichnung nicht in das reservierte Array paßt, liefert der Rückgabewert der Funktion den Fehlercode 5 zurück.

Die Typenbezeichnung entspricht der Bestellnummer. Sollte dem Treiber bei einer Karte keine Klartext-Information zur Verfügung stehen, so wird der String „TYPE CODE=xx“ zurückgeliefert mit xx = codierter Typnummer (siehe *Read_GrabberInfo*)

Hinweis:

Falls der abgefragte Grabber ein pciGrabber-4 (VD-007), eGrabber-2 (EPC-032) oder ein verwandtes Produkt ist, wird als Fehlercode 6 und als String „VD-007 or compatible“ geliefert

↔ **2 Grabber auf verwendetes Farbsystem einstellen**

void Set_Color_System(WORD nDevNo, WORD nColSys);

nColSys: Code für Farbsystem

Über die Funktion **Set_Color_System** wird das Farbsystem eingestellt, nach dem der Grabber arbeiten soll. Taktfrequenz und Eingangsregister des Videoprozessors werden entsprechend gesetzt. Der Anwender kann für *nColSys* vordefinierte Konstanten einsetzen:

- PAL_BDGHI stellt den Grabber für die Verwendung an PAL-Videoquellen ein
- NTSC_M konfiguriert für NTSC-Quellen

⇔ ② Erkennung des Videoformats

WORD Get_Video_Status(WORD nDevNo);

< Returnwert: 0 = 525 Zeilen Format (NTSC / PAL-M)
1 = 625 Zeilen Format (PAL / SECAM)

Diese Funktion erkennt das Videoformat der Kamera am eingestellten Kanal und liefert zurück, ob es sich um eine Bildquelle nach NTSC-oder PAL/SECAM-System handelt.

Als Erkennungsmerkmal wird die unterschiedliche Zeilenzahl der Fernsehsysteme herangezogen. Es dauert nach Anlegen eines Bildsignals 32 aufeinanderfolgende Halbbilder, bis die Erkennung abgeschlossen ist.

⇔ ② Einstellen des Composite-Modus (Composite-Eingänge)

void Set_Composite(WORD nDevNo);

Der Aufruf dieser Routine schaltet den Grabber in den Composite-Modus. Der Chroma-ADC wird ausgeschaltet und die Farbfalle wieder aktiviert. Dieser Modus muß eingestellt werden, wenn Composite-Signale digitalisiert werden sollen.

Der Composite-Modus muß für alle Standard-Kameras gewählt werden (schwarzweiß oder Farbe), die keinen *S-Video*-Ausgang haben, über den sie mit dem Grabber verbunden sind. Composite-Kameras werden z.B. über das Kabel WK-023 an die Buchsen ① und ② angeschlossen. Bei der Standard-Initialisierung wird der Composite-Modus eingestellt.

Hinweis:

Nach Aufruf von *Set_Composite* muß noch mit *Set_Channel* angegeben werden, von welchem Eingangskanal das Bild digitalisiert werden soll.

⇔ ② ☆ **Einstellen des S-Video-Modus'**

BYTE Set_S_VideoEx(WORD nDevNo, BYTE input);

input:	MINIDIN	= Eingang ist Mini-DIN Buchse
	COMBI	= Eingang ist Combi-Stecker (Buchse X3)
	AUTO	= automatische Einstellung

Rückgabe:

(a) <i>input</i> = MINIDIN oder COMBI		
SUCCESSFUL		= kein Fehler aufgetreten
NOT_SUPPORTED		= pciGrabber-4, EPC-031 und Einstellung COMBI
(b) <i>input</i> = AUTO		
MINIDIN		= Einspeisung von Mini-DIN-Buchse
COMBI		= Einspeisung von Combi-Buchse
NO_SIGNAL		= kein Signal gefunden (AUTO-Mode)

Beim Anschluß von S-Videoquellen muß der zweite ADC des Videoprocessors aktiviert werden. Die Funktion **Set_S_Video** aktiviert diesen *Chroma ADC*. Außerdem wird die nun überflüssige Farbfalle im Luma-Pfad deaktiviert, wodurch das Bild an Schärfe gewinnt. **Set_S_Video** schaltet den Eingangskanal automatisch auf den *S-Video*-Eingang.

S-Video-Quellen sind Farbkameras, die über einen speziellen Ausgang verfügen, an dem Helligkeits- und Farbsignal getrennt herausgeführt sind. Diese Kameras werden z.B. über das Kabel mit dem Grabber verbunden.

Achtung!

Die *S-Video*-Quelle kann entweder an der Mini-DIN-Buchse oder alternativ an der Buchse X3 angeschlossen werden. **Es dürfen nicht an beide Buchsen gleichzeitig S-Video-Quellen angeschlossen werden.**

Die Funktion besitzt einen Übergabeparameter, mit dem spezifiziert wird, an welcher Buchse die *S-Video*-Kamera angeschlossen ist. Wird als Parameter `AUTO` angegeben, sucht der Treiber nach folgendem Schema die *S-Video*-Kamera: Zuerst wird geprüft, ob ein Signal an der Mini-DIN-Buchse anliegt.

Ist dies der Fall, wird der Grabber auf diese Buchse eingestellt und der Rückgabe-Parameter ist MINIDIN. Andernfalls wird geprüft, ob am S-Video-Anschluß der Kombi-Buchse (Buchse X3) ein Signal anliegt. In diesem Fall stellt die Funktion den Grabber auf diese Buchse ein und der Rückgabe-Parameter ist COMBI.

Wird an dieser Buchse auch kein Signal festgestellt, wird der Grabber auf die Mini-DIN-Buchse eingestellt und der Rückgabewert ist NO_SIGNAL.

Hinweis:

- Ist keine S-Video-Quelle angeschlossen, aber am Kanal 3 (EPC-032) bzw. Kanal 9 (EPC-032-X1) eine Composite-Quelle vorhanden, so stellt sich der Grabber bei Verwendung der AUTO-Funktion auf die Kombi-Buchse ein und das Bild der Composite-Quelle ist als schwarz/weiß-Bild zu sehen.
- Die AUTO-Funktion arbeitet nicht, wenn die angeschlossene Bildquelle kein Videosignal liefert.
Diese Funktion ist nicht kompatibel zu alten Treiberversionen.

↔ ② ☆ Einstellen des Eingangskanals

void Set_ChannelEx(WORD nDevNo, WORD nChannel);

nChannel: Einzustellender Eingangskanal (1..9 / 1..3)

Über diese Funktion kann der Eingangskanal gewählt werden. Die Signalführung erfolgt über einen zweistufigen Multiplexer. Die erste Stufe befindet sich extern auf dem Grabberboard, während die zweite in den Videomultiplexer integriert ist.

Wertebereich:

EPC-032 : zulässig sind Kanalnummern 1 bis 3

EPC-032-X1: zulässig sind Kanalnummern 1 bis 9

EPC-032-X2 : zulässig sind Kanalnummern 1 bis 3

EPC-032-X1-X2: zulässig sind Kanalnummern 1 bis 9

Diese Funktion ist nicht kompatibel zum pciGrabber-4 (VD-007) und eGrabber-2 (EPC-031).

Hinweis:

Das Einstellen des Eingangskanals ist bei S-Video-Quellen *nicht* erforderlich! Bei *S-Video* erfolgt die Kanalumschaltung automatisch mit Aufruf der Funktion *Set_S_VideoEx()*.

Achtung!

Beachten Sie, daß bei der Umschaltung des Eingangskanals Wartezeiten einzuhalten sind, bis ein Bild des neuen Kanals digitalisiert werden kann.

Dies hat mehrere Ursachen:

- Es dauert aufgrund der Definition des Videosignals bis zu 4 Halbbilder, bis die Synchronisation eingerastet ist und das Farbsystem korrekt arbeitet.
- Aufgrund der Gleichspannungsentkopplung in Grabber und Kamera entstehen unterschiedliche mittlere Gleichspannungspegel auf den Signalleitungen. Dadurch können bei der Umschaltung Umladungseffekte auftreten.

- Die AGC der Grabberkarte muß sich erst wieder auf den neuen Signalpegel einstellen.
- Es ist daher nicht möglich, von Bild zu Bild den Kanal umzuschalten. Sie sollten Wartezeiten von mindestens 80 ms einhalten. Dies ist jedoch mit abhängig von den angeschlossenen Signalquellen und kann nur als Richtwert dienen.

Die Funktion unterstützt auch ältere Modelle. In diesem Fall sind folgende Übergabeparameter zulässig:

VD-007 und EPC-031: zulässig sind Kanalnummern 1 bis 9
VD-007-X1 und EPC-031-X1: zulässig sind die Kanalnummern

1 = Eingang 1
5 = Eingang 2
9 = Eingang 3

⇔ **Ein-/Ausschalten der Farbfalle für s/w-Betrieb**

void Set_BW(WORD nDevNo, WORD nOn);

nOn: 0 = Composite-Signal am Eingang (Farbfalle einschalten),
1 = s/w-Signal am Eingang (Farbfalle ausschalten)

Wird statt einer Farb- eine Schwarzweißkamera an den Grabber angeschlossen, so ist die Farbfalle, die störendes Farbmoiré aus dem Helligkeitssignal entfernt (*Cross-Color-Effekt*), überflüssig.

Mit dieser Funktion kann die Farbfalle softwaremäßig ein- und ausgeschaltet werden. Das Ausschalten der Farbfalle bei s/w-Signalen ist sinnvoll, da durch den Wegfall des Filters die Bildschärfe erhöht wird. Defaultmäßig ist die Farbfalle eingeschaltet.

↔ ② Ein-/Ausschalten des Interlaced-Modus'

void Set_Interlace(WORD nDevNo, WORD nInterlace);

nInterlace: 0 = Non-Interlace
 1 = Interlace
 2 = Field Aligned

Mit dieser Funktion kann dem Grabber angezeigt werden, ob es sich bei dem ankommenden Videosignal um ein Interlaced oder ein Non-Interlaced-Signal handelt. Dies beeinflusst die Funktion des vertikalen Skalierungsfilters. Bei Vollbildauflösung beispielsweise sollte *Interlace* eingestellt werden, bei Verwendung eines einzelnen Fields für die vertikale Skalierung sollte *Non-Interlace* gewählt werden, um die Bewegungs-Artifakte zu reduzieren.

Möchte man nur Halbbilder darstellen, so entsteht normalerweise zwischen zwei Digitalisierungsvorgängen eine Pause von 20 ms, da das andere Halbbild wegen des Halbbildversatzes nicht angezeigt werden kann. Im Modus *Field Aligned* wird das zweite Halbbild intern um eine halbe Zeile verschoben, so daß es räumlich auf das erste Halbbild zu liegen kommt. Dadurch kann die Halbbild-Wiedergabe im 20 ms – Rhythmus erfolgen.

Da das Halbbild durch die elektronische Filterung verändert wird, ist diese Funktion für Meß- und Automatisierungsaufgaben nur bedingt geeignet.

⇔ **② Ein-/Ausschalten der AGC**

**void Set_AGC(WORD nDevNo,WORD nCAGC, WORD nAGC,
WORD nCrush);**

nCAGC: 0 = Chroma AGC ausschalten
 1 = Chroma AGC einschalten

nAGC: 0 = AGC einschalten
 1 = AGC ausschalten

nCrush: 0 = Nicht-adaptive AGC
 1 = Adaptive AGC

Der eGrabber-4plus verfügt über zwei AGC-Regelkreise. Die allgemeine AGC überwacht die Signalpegel des Composite- bzw. Y-Signals und regelt die Eingangs-Amplitude entsprechend nach. Zusätzlich sorgt die Chroma-AGC für eine Anpassung der Farbträgeramplitude.

Für die AGC ist zusätzlich der Modus *Adaptive AGC* einstellbar. Dabei wird zusätzlich das Overflow-Bit des A/D-Wandlers überwacht. Bei einem festgestellten Overflow wird automatisch die A/D-Referenzspannung erhöht, wodurch sich der Eingangsspannungsbereich der eGrabber-4plus weiter erhöht.

Im allgemeinen ist es ausreichend, den eGrabber-4plus mit nicht adaptiver Verstärkungsregelung zu betreiben.

In Anwendungen, bei denen sich das Regelverhalten der **adaptiven** AGC störend auswirken kann (das ist besonders dann der Fall, wenn mit absoluten Helligkeitswerten gearbeitet wird), sollte die nicht-adaptive AGC verwendet werden.

Bei Applikationen, die häufig zwischen mehreren Kameras umschalten (z.B. Video-Überwachung), kann die Umschaltzeit durch die Verwendung der adaptiven AGC unter Umständen verringert werden.

↔ ② Farbtöter Ein-/Ausschalten

void Set_CKill(WORD nDevNo, WORD nCKill);

nCKill: 0 = Ausschalten des Farbtöters
 1 = Einschalten des Farbtöters

Werden an ein Farbsystem Schwarz/Weiß-Bildquellen angeschlossen, so kann es zu einem leichten Farbrauschen kommen. Der Farbtöter verhindert diesen Effekt, indem er die Anwesenheit des Color-Bursts prüft und gegebenenfalls die Farbauswertung abschaltet.

Unter Umständen möchte man ein Farbsignal digitalisieren, dessen Farbträger sehr schwach ist. In diesem Fall ist es möglich, daß die Erkennungsschwelle des Farbtöters bereits unterschritten ist und das Bild nur in Graustufen digitalisiert wird.

Über *nCKill* = 0 kann der Farbtöter explizit ausgeschaltet werden, so daß das Bild dann farbig - möglicherweise verrauscht - digitalisiert wird.

Defaultmäßig ist der Farbtöter eingeschaltet, die Umschaltung zwischen Farb- und Schwarz/Weiß-Bildquellen erfolgt also automatisch.

⇔ ② Auslassen von Fields, Frames aus dem Videosignal

```
void TemporalDect(WORD nDevNo,  
                  WORD nDecField,  
                  WORD nFldAlign,  
                  WORD nDecRat);
```

nDecField: 0 = Frame(s) auslassen
 1 = Field(s) auslassen
nAlign: 0 = odd Field wird als erstes ausgelassen
 1 = even Field wird als erstes ausgelassen
nDecRat: Anzahl auszulassender Fields / Frames

Der eGrabber-4plus ermöglicht es, bei kontinuierlicher Digitalisierung, die Anzahl der pro Sekunde digitalisierten Bilder einzustellen. Defaultmäßig werden 50 bzw. 60 (NTSC) Bilder pro Sekunde digitalisiert (Video-Norm).

Mit Hilfe dieser Funktion kann bestimmt werden, wieviele Bilder von diesen 50 bzw. 60 Bildern bei der Digitalisierung *ausgelassen* werden sollen.

Über die beiden anderen Parameter wird die Art des Auslassungsvorgangs gesteuert:

nAlign gibt an, ob die Auslassung mit geraden oder ungeraden Halbbildern beginnen soll.

nDecField bestimmt, ob sich die Auslassung auf Voll- oder Halbbilder beziehen soll.

Beispiele (PAL):

- $nDecField=0, nDecRate=2$

Die Reduzierung wird auf Vollbild-Ebene durchgeführt. Pro 50 Bilder werden zwei Bilder ausgelassen. Die Bilder 1-24 werden normal ausgegeben, dann wird ein Bild ausgelassen. Die Bilder 26 bis 49 werden ausgegeben, gefolgt von einem ausgelassenen Bild.

- $nDecField=1, nDecRate=25$

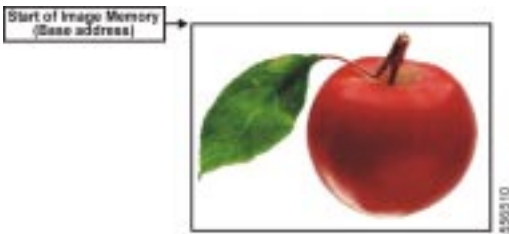

Die Reduzierung wird auf Halbbild-Ebene durchgeführt. Fünfundzwanzig Halbbilder werden pro 50 Halbbilder ausgelassen. Dies resultiert darin, daß jedes zweite Halbbild ausgegeben wird. Es werden also immer gleiche Halbbilder ausgelassen. Welches Halbbild zuerst ausgelassen wird, hängt von *nAlign* ab.

🏠 ☆ 🔄 Bildorientierung einstellen

DWORD FlipPicture(WORD nDevNo, unsigned char flip)

nDevNo : System-Nummer des Grabbers
 flip : 0 = Bild aufrecht
 : 1 = Bild vertikal gespiegelt (default)

Mit dieser Funktion wird festgelegt, wie das vom Grabber digitalisierte Bild im Speicher abgelegt wird:

flip = 0	flip=1 (default)
<p>Das Bild wird aufrecht dargestellt, an der niederwertigsten Bildspeicheradresse befindet sich die linke <u>obere</u> Ecke des Bilds:</p>	<p>Das Bild wird <u>vertikal</u> gespiegelt geschrieben. An der niederwertigsten Bildspeicheradresse befindet sich die linke <u>untere</u> Ecke des Bilds:</p>
	
	<p>Diese Variante ist z.B. nützlich, wenn Bilder im BMP-Format verarbeitet werden.</p>

Achtung:

- Die Einstellung der Bildorientierung mit *FlipPicture()* muß vor Aufruf der *Set_Image()* – Funktion erfolgen. Die Einstellung wirkt sich erst nach Aufruf der *Set_Image()* – Funktion aus. Wird die Einstellung im laufenden Betrieb geändert, so muß der Grabber angehalten werden, über *FlipPicture()* der gewünschte Wert gesetzt und dann *SetImage()* erneut aufgerufen werden.
- Die Default-Einstellung ist **flip=1**, das Bild wird also horizontal gespiegelt eingeschrieben (aus Kompatibilitätsgründen)

2 Bildgröße und Skalierung einstellen

**void Set_Image (WORD nDevNo,
WORD nOhpos, WORD nOvpos,
WORD nOhsize, WORD nOvsize,
WORD nOppl, WORD nOlines,
WORD nOColformat,
WORD nEhpos, WORD nEvpos,
WORD nEhsize, WORD nEvsized,
WORD nEppl, WORD nElines,
WORD nEColformat,
WORD nColSystem,
WORD nInterlaced,
WORD nSingleShot);**

- nOhpos, nOvpos : Position der linken oberen Ecke des Odd-Bildausschnitts im Videobild
(hpos = horizontal, vpos = vertikal)
- nOhsize : Größe des Odd-Bildausschnitts in X-Richtung
- nOvsize : Größe des Odd-Bildausschnitts in Y-Richtung
- nOppl : gewünschte Größe des Odd-Videobilds in X-Richtung
(ppl = Pixel Per Line)
- nOlines : gewünschte Zeilenzahl des Odd-Videobilds
- nOColformat: gewünschtes Farbformat: (RGB32, RGB24, RGB16, RGB15, Y8, YCrCb 4:2:2, YCrCb 4:1:1)
- nEhpos, nEvpos : Position der linken oberen Ecke des Even-Bildausschnitts im Videobild
(hpos = horizontal, vpos = vertikal)
- nEhsize : Größe des Even-Bildausschnitts in X-Richtung
- nEvsized : Größe des Even-Bildausschnitts in Y-Richtung
- nEppl : gewünschte Größe des Even-Videobilds in X-Richtung
(Pixel Per Line)
- nElines : gewünschte Zeilenzahl des Even-Videobilds
- nEColformat: gewünschtes Farbformat: (RGB32, RGB24, RGB16, RGB15, Y8, YCrCb 4:2:2, YCrCb 4:1:1)
- nColSystem : Code für Farbsystem (vgl. *Set_Color_System*)
- nInterlace: 0 = Non-Interlace
1 = Interlace
2 = Field Aligned
- nSingleShot : 0 = kontinuierliche Digitalisierung
1 = **ein** Bild grabben

Mit der Routine **Set_Image ()** werden Größe, Position und Skalierung des vom Grabber gelieferten Bildausschnitts für Even- und Odd-Bilder getrennt festgelegt. Des weiteren wird das Datenformat, in dem das Bild im Speicher abgelegt wird, definiert.

Beachten Sie, daß diese Funktion in der DOS-Version des Treibers andere Parameter hat (siehe dort).

Achtung!

Der Aufruf der Funktion darf nur erfolgen, wenn sich der Grabber im Stop-Modus befindet. *Set_Image* darf **nicht** aufgerufen werden, wenn der Grabber digitalisiert oder eine Single-Shot – Digitalisierung noch nicht mit *Stop_Grabber* abgeschlossen wurde.

Die Einstellungen können grundsätzlich für beide Halbbilder getrennt vorgenommen werden. Der Parametersatz ist entsprechend gegliedert in Parameter mit vorgestelltem 'E' = Parameter für Even-Bild und vorgestelltem 'O' = Odd-Bild - Parameter. Parameter ohne diese Kennzeichnung gelten für beide Halbbilder.

Für die beiden Halbbilder können völlig verschiedene Bildgrößen angegeben werden. Sie werden im Bildspeicher hintereinander abgelegt und können getrennt voneinander bearbeitet werden.

Im *Interlaced-Modus* werden die beiden Halbbilder richtig verzahnt in einen gemeinsamen Speicherbereich geschrieben, wodurch sich die volle Auflösung von bis zu 720 x 576 Bildpunkten (PAL) bzw. 640 x 480 Bildpunkten (NTSC) ergibt (*siehe auch Abschnitt ⇔ ② Ein-/Ausschalten des Interlaced-Modus*).

Im folgenden wird die Einstellung der Werte ohne Voranstellung der Halbbild-Spezifikation diskutiert (z.B. *hsize* für *nEhsize / nOhsize*). Es ist dann jeweils 'E' für 'Even' bzw. 'O' für 'Odd' voranzustellen. Ist die Festlegung auf einen bestimmten Halbbildtyp erforderlich, so wird der Parameter voll ausgeschrieben.

7Bei der Einstellung der Parameter legt man zuerst ein Fenster mit der Größe des gewünschten Bilds (in Pixeln) fest. Dies geschieht mittels der Werte von *hsize* und *vsze*. *Hsize* gibt also die Anzahl der Pixel in x-Richtung an, die das aufgenommene Bild später hat, *vsze* die Anzahl der Pixel in y-Richtung.

Die Werte *ppl* und *lines* geben hingegen an, wieviel Pixel aus dem ankommenden Videobild generiert werden sollen. *ppl* definiert die Anzahl der Pixel pro Bildzeile, *lines* gibt an, wieviele Zeilen („Pixel in y-Richtung“) erzeugt werden.

Mit diesen beiden Werten kann also die Auflösung des Bilds festgelegt werden. Ein Vollbild im PAL-Format hat 720 Pixel x 576 Zeilen. Um es in höchstmöglicher Auflösung zu digitalisieren, wird *ppl* der Wert 720 zugewiesen und *lines* = 576 gesetzt. Die kleinsten Auflösungen liegen für PAL bei 50 x 40 Pixeln pro Vollbild.

Werden einzelne Halbbilder benutzt, so liegt die maximale Auflösung bei jeweils 720 x 288 Zeilen.

Aufgrund der Tatsache, daß sich die Definition von *ppl* und *lines* stets auf *Halbbilder* bezieht, die tatsächliche Zeilenzahl des zu digitalisierenden TV-Bilds also doppelt so hoch ist, ist das Breiten-/Höhenverhältnis bei den digitalisierten *Halbbildern* (maximal dann 720 x 288 Pixel) um den Faktor zwei verzerrt. Um dies zu beheben, digitalisiert man entweder im Interlaced-Modus (wenn die hohe Auflösung erforderlich ist) oder man reduziert die Auflösung auf *ppl*=360, *lines*=288 und arbeitet mit der proportionsgerechten Auflösung 360 x 288.

Bei Vermessungs- und Automatisierungsanwendungen ist oft nicht unbedingt eine proportionsgerechte Darstellung nötig, sofern die Verzerrung im Algorithmus berücksichtigt wird oder nicht ins Gewicht fällt. So kann man z.B. die volle Halbbildauflösung 720 x 288 bei Vermessungen benutzen, die in X-Richtung genauer als in Y-Richtung arbeiten. Gegebenenfalls kann man die Kamera entsprechend der benötigten Vermessungsachse ausrichten.

Das mit *hsize* und *vsize* aufgespannte Fenster ist nun der Ausschnitt des Bildes, den man von dem digitalisierten Bild der Größe *ppl* x *lines* sieht. Ist *hsize* = *ppl* und *vsize* = *lines*, so sieht man das ganze digitalisierte Bild; sind sie kleiner, nur einen entsprechenden Ausschnitt. Das Verhältnis von *hsize* und *vsize* verändert nicht die Proportionen des Bildes, da hier keine Skalierung, sondern nur eine Ausschnittsbildung aus dem bereits skalierten Bild vorgenommen wird. *Bild 50* und *Bild 51* illustrieren die Bedeutung der Parameter.

Die Parameter *vsize* und *lines* sind in Bezug auf ein Halbbild anzugeben, wenn ein Halbbild digitalisiert und abgeholt wird.

Ist der mit *hsize* und *vsize* definierte Ausschnitt kleiner als die über *ppl* und *lines* festgelegte Bildgröße, so kann das Fenster mit den Werten von *hpos* und *vpos* in dem digitalisierten Bild verschoben werden. Bei *hpos*=0 und *vpos*=0 liegt es in der linken oberen Ecke des digitalisierten Bildes.

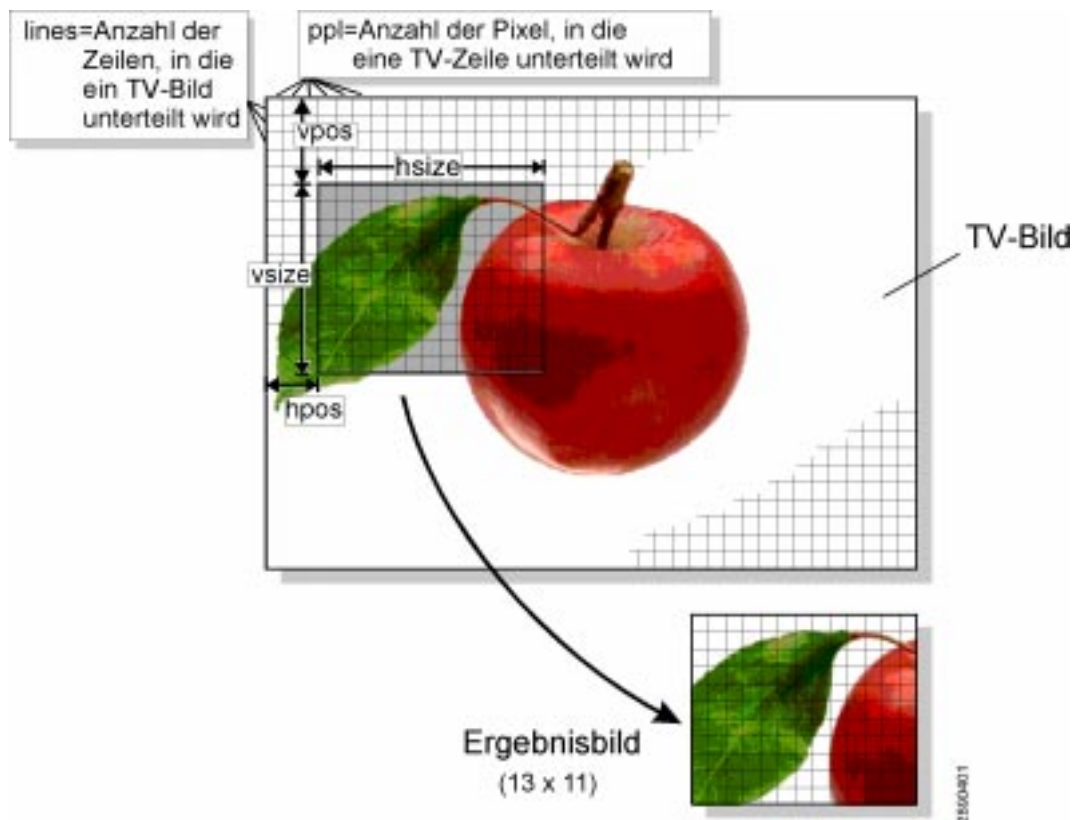


Bild 50: Skalierung und Ausschnittsbildung

Achtung!

- Der über *hsize* und *vsize* in der Größe und *hpos* und *vpos* in der Position definierte Fensterbereich darf an keiner Stelle den durch *ppl* und *lines* definierten Bereich des digitalisierten Bilds verlassen:

hpos=100, *hsize*=200, *ppl*=200 : nicht zulässig, da die letzten 100 Pixel nicht definiert sind

hpos=0, *hsize*=200, *ppl*=202 : zulässig; alle Pixel liegen im Bildbereich

hpos=100, *hsize*=100, *ppl*=200 : zulässig

hpos=100, *hsize*=200, *ppl*=300 : zulässig

hpos=300, *hsize*=300, *ppl*=800 : nicht zulässig: Bildbereich hat mehr Pixel als die TV-Norm liefert

(entsprechend in Y-Richtung)

- Alle Parameter in horizontaler Richtung müssen *gerade* Werte haben!
(*ppl*=123 ist nicht erlaubt, *ppl*=124 ist zulässig.)

- Werden die Parameter, die die Bildgröße beeinflussen, auf 0 gesetzt, so wird das entsprechende Halbbild nicht erzeugt.

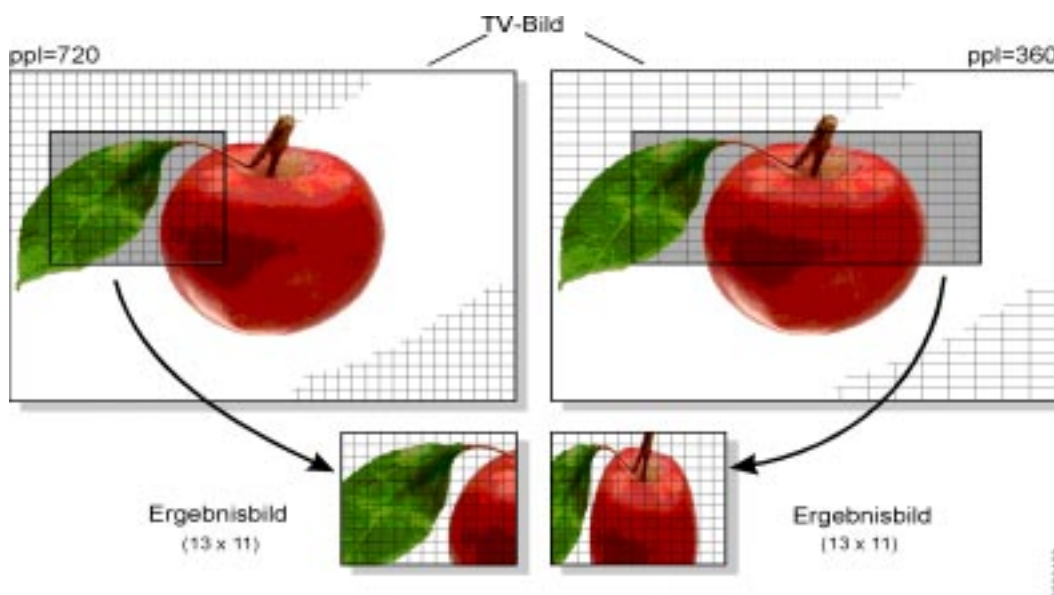


Bild 51: Bild zur Skalierung: alle Werte gleich bis auf *ppl*

Beispiele:

- ***Halbbild-Digitalisierung***

Es soll ein quadratisches Bild der Größe 256 x 256 Pixel digitalisiert werden, dessen Auflösung und Proportionen dem TV-Bild entsprechen.

(a) Auflösung und Skalierung

Zur Digitalisierung genügt ein Halbbild mit 288 Zeilen. Damit das Höhen-/Breitenverhältnis stimmt, muß die Auflösung in X-Richtung ebenfalls auf die Hälfte verringert werden (denn: Halbbild = halbe Höhe).

Also $720 : 2 = 360$.

Es ergeben sich $ppl = 360$, $lines = 288$

(b) Ausschnittsgröße

Das Bild soll quadratisch 256 x 256 Pixel groß sein. Daraus ergibt sich direkt:

$hsize = 256$, $vsize = 256$

(c) Positionierung

Es ist sinnvoll, den Bildausschnitt zu zentrieren.

In X-Richtung werden von 360 Pixeln nur 256 im Fenster dargestellt. Es entsteht ein Rand von $360 - 256 = 104$ Pixeln, der gleichmäßig auf beide Seiten verteilt werden soll, also jeweils 52 Pixel links und rechts. $hpos$ ist die Größe des linken Randes, also $hpos = 52$.

Entsprechend ist in Y-Richtung: $(288 - 256) : 2 = 16$; $vpos = 16$.

Hinweis:

Es wäre hier falsch, $ppl = 256$ und $lines = 256$ zu setzen. Dadurch würde das Breiten-/Höhenverhältnis verändert (TV-Norm = 4:3) zu 1:1 und das Bild verzerrt. Es wäre jedoch möglich, $lines = 256$ zu setzen und ppl über das Verhältnis von Bildbreite und -höhe zu berechnen. So wäre die Bildhöhe optimal ausgenutzt.

- **Halbbild-Digitalisierung mit Zoom**

Es soll ein Bild der Größe 120 x 100 geliefert werden, in dem das Originalbild um den Faktor 2 in X- und Y-Richtung vergrößert dargestellt wird.

(a) Auflösung / Skalierung

Ausreichend ist wieder ein Halbbild. Ohne Vergrößerung wären 360 x 288 Pixel anzufordern (s.o.). Um die Dehnung zu erzeugen, werden nun 180 x 144 Pixel definiert: $ppl = 180$, $lines = 144$. Das Breiten-/Höhenverhältnis wird gewahrt.

(b) Ausschnittsgröße

Entsprechend der Fenstergröße wird gesetzt:
 $hsize = 120$, $vsize = 100$.

(c) Positionierung

Mit $hpos$ und $vpos$ kann der Fensterausschnitt nach Bedarf maximal $180-120 = 60$ Pixel in X-Richtung und $144-100 = 44$ Pixel in Y-Richtung verschoben werden.

- **Vollbild-Digitalisierung**

Es soll ein Bild der Größe 700 x 500 geliefert werden, das in Auflösung und Proportionen dem TV-Bild entspricht.

(a) Auflösung und Skalierung

Zur Digitalisierung wird hier ein Vollbild benötigt. Da also als Grundlage ein volles TV-Bild dienen soll, beträgt die zugrundeliegende Bildauflösung $ppl = 720$ und $lines = 576$. In vertikaler Richtung muß der lines-Wert nun auf beide Halbbilder verteilt werden:

$$nOlines = nElines = \frac{1}{2} lines$$

$$nOlines = \frac{1}{2} 576 = 288$$

$$nElines = \frac{1}{2} 576 = 288$$

Da das erzeugte Vollbild insgesamt 576 Zeilen hat, werden in X-Richtung 720 Pixel benötigt, damit das Höhen-/Breitenverhältnis stimmt:

$$nOppl = nEpl = 720$$

(b) Ausschnittsgröße

Das Bild soll 700 x 500 Pixel groß sein. Daraus ergibt sich direkt: $hsize=700$.

Für die vertikale Ausschnittsgröße muß wieder die Aufteilung in zwei Halbbilder erfolgen:

$$nOvsize = nEvsize = \frac{1}{2} 500 \text{ Pixel} = 250$$

(c) Positionierung

Es ist sinnvoll, den Bildausschnitt zu zentrieren.

In X-Richtung werden von 720 Pixeln nur 700 im Fenster dargestellt. Es entsteht ein Rand von $720-700 = 20$ Pixeln, der gleichmäßig auf beide Seiten verteilt werden soll, also jeweils 10 Pixel links und rechts. $hpos$ ist die Größe des linken Randes, also $hpos=5$:

$$nOhpos = nEhpos = 4 \text{ (gerader Wert)}$$

Entsprechend ist in Y-Richtung: $(288-250):2 = 19$;

$$nOvpos = nEvpos = 18 \text{ (gerader Wert)}.$$

Der Parameter $nInterlaced$ sollte auf 1 gesetzt werden, damit die Bilder automatisch verzahnt werden.

Nachdem nun Größe und Auflösung des Bilds definiert sind, muß im nächsten Schritt das Datenformat ausgewählt werden. Der Parameter Colformat beschreibt, in welcher Form ein Pixel im Arbeitsspeicher der CPU abgelegt wird und damit auch, wieviel Bytes von einem Pixel belegt werden.

Das Datenformat ist in erster Linie von der Anwendung bestimmt. Grundsätzlich sind drei Formate zu unterscheiden, die teilweise nochmals in Unterformate unterteilt sind. *Bild 52* zeigt, wie die Pixel in den einzelnen Formaten im Speicher abgelegt werden.

- **RGB:** Die Helligkeitsinformation wird für die drei Farbkanäle Rot, Grün und Blau getrennt abgelegt. Dies ist die Form, in der allgemein Farbinformation betrachtet und verarbeitet wird.
 - Bei *RGB32* werden 32-bit, also ein Doppelwort pro Pixel benutzt. Das unterste Byte eines jeden Doppelworts enthält die Blau-Information (8-bit breit), das zweite Grün- und das dritte die Rotwerte. Das oberste Byte enthält keine Information. Es wird lediglich eingefügt, damit die Zuordnung von einem Pixel zu einem Doppelwort erreicht wird: Mit jeder DWort-Grenze beginnt auch ein neues Pixel (zusammengehörige Information ist in *Bild 52* in gleicher Weise schraffiert). Das Alignment auf DWords hat gegebenenfalls den Vorteil, daß schnellere Zugriffsbefehle benutzt werden können. Die Farbtiefe beträgt 16 Mio. Farben ($2^{38} = 16777 \cdot 10^6$).
 - *RGB24* überträgt den gleichen Informationsgehalt wie *RGB32*, es entfällt jedoch das Füllbyte. Das Bild wird bei gleicher Farbauflösung kompakter im Speicher dargestellt.
 - *RGB16* hat eine geringere Farbauflösung. In diesem Format werden jeweils fünf Bit für Blau und Rot-Kanal verwendet, der Grün-Kanal verfügt über sechs Bit. Die Farbtiefe beträgt damit 65.536 Farben ($2^5 \cdot 2^5 \cdot 2^6 = 65536$). Der Speicherplatzbedarf eines Pixels beträgt dementsprechend 16-bit = 1 Wort. In *Bild 52* ist die Aufteilung des Worts in die drei Farbkanäle gezeigt.

Die Farbinformation wird dabei „linksbündig“ ausgerichtet; die niederwertigen Bits fallen also weg, wodurch die Farbtiefe reduziert wird.

Die Farbtiefe des Grün-Kanals ist doppelt so hoch wie die der beiden anderen Kanäle. Das RGB16-Format entspricht dem Farbsystem von Grafikkarten mit 65.535 Farbstufen.

- **RGB15:** Die Aufteilung entspricht dem RGB16-System, jedoch ist die Farbtiefe für alle drei Farbkanäle gleich (je 5-bit = 32 Stufen). Daraus ergeben sich 32.818 darstellbare Farben. Insgesamt werden nur 15-bit benötigt, wodurch das oberste Bit eines jeden Wortes unbenutzt bleibt. Es enthält den Wert 0 (*siehe Bild 52*).

- **YCrCb:** In diesem Format liegen Grauwert und Farbinformation getrennt vor. Der Parameter Y beschreibt die Helligkeit des Pixels (Grauwert), der Wertetupel (Cr,Cb) enthält die Farbinformation. (Cr,Cb) ist als Vektor im Farbkreis zu verstehen. Der Farbton entspricht dem Winkel des Zeigers, die Sättigung wird durch den Betrag dargestellt. Anwendung findet dieses Format z.B. beim Weiterleiten der Daten an eine für das YCrCb-System ausgelegte Grafikkarte, beim getrennten Verarbeiten von Helligkeit und Farbwert und - aufgrund seiner Kompaktheit - beim (Zwischen-)Speichern von Bildinformation.
 - **YUY2:** Dieses Format entspricht dem Format YCrCb 4:2:2. In einem DWord befindet sich die Information zweier Pixel. Y0 und Y1 ist die Helligkeitsinformation der benachbarten Pixel, Cb0 und Cr0 die Farbinformation des ersten Pixels, die für beide Pixel genutzt wird. Die Farbinformation des zweiten Pixels wird nicht genutzt.
 - **BtYUV:** Entspricht YCrCb 4:1:1. Vier Pixel teilen sich eine Farbinformation. Die Anordnung der Information im Speicher ist aus *Bild 52* zu ersehen. Jeweils drei DWorte sind logisch zusammengefaßt. Sie enthalten die Information für acht Pixel. Dies ist die Helligkeit für jedes Pixel (Y0..Y7) und die Farbinformation des ersten (Cb0/Cr0) und fünften Pixels (Cb4/Cr4).

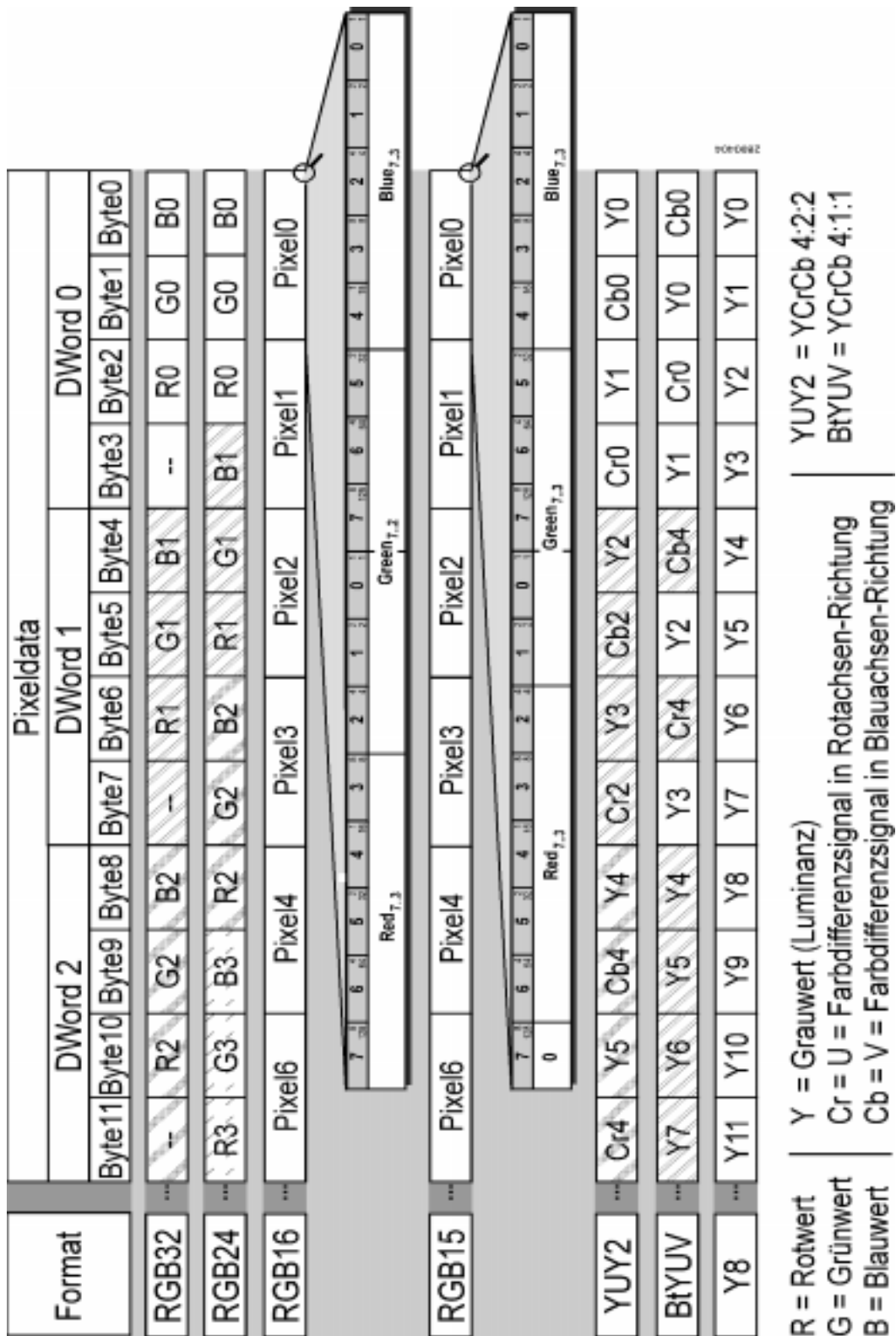


Bild 52: Farbformate des eGrabber-4plus

- **Y:** Im Grayscale-Format wird nur der Grauwert, also die Helligkeitsinformation eines Pixels abgespeichert. Die Farbinformation wird nicht berücksichtigt. Dieses Format empfiehlt sich, wenn bei der Auswertung Farbe keine Rolle spielt.
 - **Y8:** Im Y8-Format ist die Grauwertinformation jedes Pixels als 8-bit-Wert hintereinander im Speicher abgelegt. Ein Byte entspricht also einem Pixel.

Nun ist das Format des zu digitalisierenden Bildes genau definiert. Der Grabber muß jetzt angewiesen werden, wo und in welcher Weise die erzeugten Bilddaten im Arbeitsspeicher abgelegt werden. Dazu ist es zunächst erforderlich, einen ausreichend großen Arbeitsspeicherbereich zu reservieren.

Dies geschieht mit den bekannten Befehlen zur Allokation von Speicher in der jeweiligen Programmiersprache (z.B. *malloc(...)*).

Der Windows-Treiber reserviert selbständig einen entsprechenden Bildspeicher, in dem das Bild abgelegt wird.

Wieviel Speicherplatz wird benötigt? Dies errechnet sich aus der Größe des Bilds (Anzahl der Pixel) und der pro Pixel benötigten Byte (Farbauflösung):

$$\text{Speicherbedarf pro Halbbild} = \text{hsize} \cdot \text{vsize} \cdot \text{pixelgröße [Byte]}$$

Der Wert *pixelgröße* bestimmt sich wie in *Tabelle 12* gezeigt.

Bei den Formaten YUV2 und BtYUV muß beachtet werden, daß 2 bzw. 8 Pixel logisch zusammengehören und die Bildauflösung entsprechend passend gewählt wird. Bei der Berechnung ist zu beachten, daß die Pixelzahl *hsize* *vsize* für jedes Halbbild einzeln zu berechnen und gegebenenfalls bei Vollbildern zu addieren ist.

Format	<i>pixelgröße</i> [Byte]
RGB32	4
RGB24	3
RGB16, RGB15	2
YUY2	4 Byte pro 2 Pixel
BtYUV	12 Byte für 8 Pixel
Y8	1

Tabelle 12: Speicherbedarf eines Pixel in den einzelnen Modi

Werden Vollbilder benötigt, weil die vertikale Auflösung mehr als 288 Zeilen betragen soll, so kann der Grabber angewiesen werden, dieses Vollbild in einem Speicherbereich komplett aufzubauen. Er nimmt dann die erforderliche Verzahnung der beiden Halbbilder automatisch vor. Wird diese Option gewünscht, so setzt man $nInterlaced = 1$.

Abschließend wird angegeben, in welcher Weise die Bildaufnahme vorgenommen werden soll: Mit $nSingleShot = 0$ wird der Grabber angewiesen, kontinuierlich zu digitalisieren. Das bedeutet, daß ab dem Moment, in dem der Grabber den Befehl zum Start erhält, andauernd in Echtzeit (50 Halbbilder pro Sek.) die angegebenen Speicherbereiche mit aktuellen Bilddaten beschrieben werden. Im Halbbild-Modus ($nInterlaced = 0$) wird ein Speicherbereich beschrieben (20 ms), dann der andere (wieder 20 ms) und weiter wechselweise. Dies bedeutet, daß jeder Halbbildspeicher mindestens 20 ms lang nicht vom Grabber belegt wird.

Im Vollbildmodus ($nInterlaced = 1$) beschreibt der Grabber den gemeinsamen Bildspeicher ständig, 20 ms die ungeraden, dann 20 ms die geraden Zeilen.

Bei der Auswertung des Bildinhalts kann es störend sein, wenn man in dem Bildfeld arbeitet, das der Grabber gerade aktualisiert, da ein Versatz von bewegten Objekten ab einer zufälligen Bildzeile auftreten kann. In diesem Fall ist es ratsam, im Start/Stop-Betrieb zu arbeiten oder die Halbbilder in zwei getrennte Bildspeicher zu digitalisieren, die man entsprechend abwechselnd bearbeitet.

nSingleShot = 1 bewirkt, daß nur genau ein Digitalisierungsvorgang durchgeführt wird. Es werden zwei Halbbilder (ein Odd- und ein Even-Bild) aufgenommen, bzw. ein Vollbild.

Der Anwender kann die Bilder aufnehmen und dann per erneutem Start-Befehl die Bildspeicher aktualisieren. Diese Betriebsart ist empfehlenswert, wenn nur gelegentlich Bilder digitalisiert werden müssen und es nicht auf Echtzeit-Darstellung ankommt.

Egal, ob kontinuierlich gegrabbt wird oder nur Einzelbilder geschossen werden: **Set_Image()** stellt nur die Parameter ein, legt also fest, *wie* die Bilder aufgenommen werden sollen. Der Grabbvorgang selbst wird dadurch noch nicht gestartet. Dies geschieht durch den Befehl **Start_Grabber()** (*siehe unten*).

↔ ③ Bildaufnahme starten

void Start_Grabber(WORD nDevNo);

Die Funktion **Start_Grabber()** startet den Grabvorgang für den über *nDevNo* spezifizierten Grabber. Dies bedeutet, daß der Digitalisierungsvorgang mit dem Beginn des nächsten verfügbaren Bildes beginnt. Ist kontinuierliches Grabben gewählt worden, so wird fortgesetzt digitalisiert, bis **Stop_Grabber()** aufgerufen wird. Bei Einzelbild-Betrieb wird die Digitalisierung nach dem ersten vollständigen Bild beendet.

Wann wird die erste Digitalisierung durchgeführt?

Der Treiber arbeitet auf Vollbild-Basis. Es werden immer zusammenhängende Even/Odd-Bildkombinationen betrachtet. Das führt zu einem Zeitverhalten, das abhängig ist vom gewünschten Halbbild und dem zum Startzeitpunkt gerade aktuell am Videoeingang anliegenden Halbbild. Folgende Fälle müssen unterschieden werden:

(1) Es soll ein Even-Bild digitalisiert werden

a) Am Eingang liegt gerade ein Even-Bild an

⇒ Da das laufende Bild nicht mehr ausgewertet werden kann (der Beginn fehlt), läuft der Rest des Even-Bilds und das folgende Odd-Bild durch, danach beginnt die Digitalisierung. Es wird also das nächste vollständige Even-Bild digitalisiert. Die Zeitverzögerung Startbefehl bis Digitalisierungsbeginn beträgt < 40 ms.

b) Am Eingang liegt gerade ein Odd-Bild an

⇒ Das auf das Odd-Bild folgende Even-Bild wird digitalisiert. Maximale Verzögerung: 20 ms.

(2) Es soll ein Odd-Bild digitalisiert werden

a) Am Eingang liegt gerade ein Even-Bild an

⇒ Da der Treiber zuerst Even-Bilder verarbeitet, vergeht zunächst der Rest des unvollständigen Even-Bilds und das folgende Odd-Bild, bevor der Grabber auf dem nächsten Even-Bild einrastet. Das auf dieses Even-Bild folgende Odd-Bild wird digitalisiert. Die maximale Verzögerung beträgt < 60 ms.

b) Am Eingang liegt gerade ein Odd-Bild an

⇒ Es vergeht das unvollständige Odd-Bild, danach rastet der Grabber im Even-Bild ein und digitalisiert das folgende Odd-Bild. Verzögerung: < 40 ms.

↔ **3 Bildaufnahme stoppen**

void Stop_Grabber(WORD nDevNo);

Die Funktion **Stop_Grabber** stoppt den Grabvorgang. Der Bildeinzug wird sofort abgebrochen, das aktuell digitalisierte Bild ist eventuell unvollständig (*siehe Bild 49*).

Achtung!

Stop_Grabber muß auch dann aufgerufen werden, wenn die Digitalisierung bei Single-Shot-Betrieb (*nSingleShot = 1* in *Set_Image*) automatisch beendet wird. Der Grabber ist so lange verriegelt (in Wartestellung), bis **Stop_Grabber** aufgerufen wird. Danach kann ein neues Einzelbild mit **Start_Grabber** angefordert werden.

⇔ ③ Digitalisierungsfortschritt erkennen

WORD Data_Present(WORD nDevNo);

Returnwert: zeigt den Digitalisierungsfortschritt an
(kann Werte zwischen 0 und 15 (4-bit) annehmen)

Die Funktion **Data_Present** signalisiert, ob ein Even oder Odd Bild digitalisiert und im Speicher abgelegt wurde.

In den einzelnen Bits des Rückgabewerts ist codiert, wie weit die Digitalisierung bei Einzelbild- oder kontinuierlichem Betrieb fortgeschritten ist (*siehe Bild 53*).

Bit 0 und 2 zeigen an, daß ein Even-Bild eingelesen wurde, Bit 1 und 3 ein Odd-Bild. Die Bits 0 und 1 wechseln bei jedem Einlaufen eines Even- bzw. Odd-Bildes ihren Wert.



Bild 53: Rückgabewert der Funktion data_present

Bei kontinuierlichem Grabben kann damit bestimmt werden, wann der Inhalt des entsprechenden Bildspeichers mit einem neuen Bild vollständig ersetzt wurde. Jeder *Wechsel* des Zustands (0 auf 1 und 1 auf 0) zeigt an, daß ein neues Halbbild eingeschrieben wurde.

Die Bits 2 und 3 werden auf 1 gesetzt, sobald ein Even- bzw. Odd-Bild vollständig eingelesen wurde. Diese Bits werden verwendet, wenn nur ein Bild eingelesen werden soll (Single-Shot, durch **Set_Image** festgelegt). Sie bleiben dann so lange gesetzt, bis ein neuer Digitalisierungsvorgang eingeleitet wird.

Achtung!

Fragen Sie den Status während der Digitalisierung nicht zu häufig ab, da jede Abfrage den PCI-Bus belegt und dadurch den Grabber an der Datenübertragung hindert. Fügen Sie gegebenenfalls Wartezeiten zwischen den Abfragen ein, um die Digitalisierung nicht unnötig zu verlangsamen.

Achtung!

Achten Sie darauf, die richtigen, dem aktuellen Modus entsprechenden Bits auszuwerten, da ansonsten Ihr Programm möglicherweise zum falschen Zeitpunkt auf die Bilddaten zugreift.

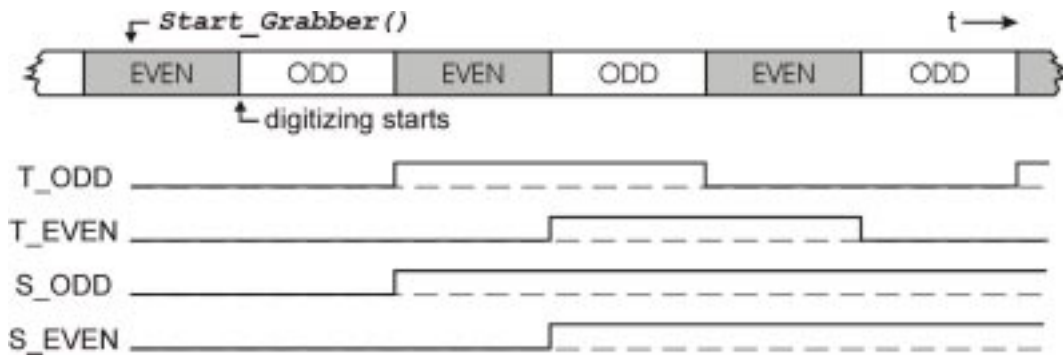


Bild 54: Timing Diagramm der Rückgabe-Parameters von `data_present()`

③ Bilddaten auslesen

**DWORD GetPictureBufferAddress(WORD nDevNo,
DWORD dwBitsSize) ;**

dwBitsSize: Größe des für das Bild benötigten Speicherbereichs

Returnwert: Adresse des Beginns des Speicherbereichs

Um die digitalisierte Bildinformation aus dem vom Gerätetreiber verwalteten Bildspeicher auslesen zu können, muß dem Anwenderprogramm die Startadresse dieses Bildspeichers bekannt sein.

Die Funktion **GetPictureBufferAddress** liefert dem Anwenderprogramm die Adresse zurück, an welcher der vom Treiber reservierte Speicherbereich beginnt. In diesem Speicher legt der Grabber die digitalisierten Bilder wie folgt ab:

(a) Nur Even- oder nur Odd-Bilder werden digitalisiert

(d.h. die Dimensionsparameter des anderen Halbbilds sind Null)

⇒ Das digitalisierte Halbbild steht am Beginn des Bildspeicherbereichs.

(b) Even- und Odd-Bilder werden digitalisiert

(nInterlaced=0)

⇒ Das Even-Bild steht am Beginn des Bildspeicherbereichs. Bündig hinter dem Even-Bild beginnt das Odd-Bild. Die Startadresse des Odd-Bilds ist also die Startadresse des Bildspeichers plus $nEhszize$ $nEvsize$ Anzahl der Byte pro Pixel.

(c) Vollbilder werden im Interlace-Modus digitalisiert

(nInterlaced=1)

⇒ Das digitalisierte Vollbild besteht aus richtig ineinandergeschachtelten Halbbildern. Es beginnt an der Startadresse des Bildspeichers.

Der Parameter *dwBitsSize* gibt die Gesamtgröße des benötigten Bildspeichers an. Diese errechnet sich für ein Bild aus der Anzahl der Pixel und der pro Pixel benötigten Bytes: *hsize* *vsize* Pixelgröße. Beachten Sie, daß *dwBitsSize* die Größe des **insgesamt** benötigten Bildspeichers angibt. Werden Even- und Odd-Halbbild getrennt digitalisiert, so müssen Sie den Speicherplatzbedarf beider Bilder addieren. Mehr zur Berechnung des Speicherplatzbedarfs erfahren Sie bei dem Befehl **SetImage()**.

Achtung!

Bei Verwendung des **Windows'95**-Treibers benutzen alle Grabber im System den selben Bild-Puffer. Es können daher nicht zwei Grabber *gleichzeitig* eine Digitalisierung durchführen, wenn die Standard-Routinen des Treibers verwendet werden, da sie ansonsten ihre Bilderdaten im Puffer gegenseitig überschreiben würden. Bei den Treibern für alle anderen Betriebssysteme besteht diese Einschränkung nicht.

Interrupt aktivieren

DWORD ActivateFieldInterrupt (WORD nDevNo, HANDLE *hEvent);

nDevNo : System-Nummer des Grabbers

HANDLE: Pointer auf ein Event, daß beim triggernden Ereignis ausgelöst wird.

Die Framegrabberkarte kann über einen Interrupt das Ende eines Halbbild-Digitalisierungsvorgangs anzeigen. Dazu wird ein Event definiert, das der Grabber beim Auftreten des Ereignisses (Halbbild digitalisiert) auslöst. Damit kann sehr schnell auf dieses Ereignis reagiert werden und das Polling (siehe *DataPresent()*) entfallen.

Die Funktion *ActivateFieldInterrupt()* dient zur Aktivierung der Interrupt-Funktion des Grabbers. Gleichzeitig wird ein Zeiger auf ein Event übergeben, das durch das triggernde Ereignis ausgelöst wird. Dieses Event wird vom Treiber angelegt.

Sie können in Ihrer Applikation auf dieses Event triggern, indem Sie die Windows-API-Funktion *WaitForSingleObject()* verwenden.

Beachten Sie, daß Sie das Event nach der Auswertung durch Aufruf der Funktion *ResetEvent()* zurücksetzen müssen, bevor Sie erneut auf das Event triggern können.

Achtung:

- Das Event tritt immer ein, wenn die Digitalisierung eines Halbbilds abgeschlossen ist.

Dies ist nur der Fall, wenn der Grabber tatsächlich gestartet wurde. Nach Eintreten des Ereignisses bleibt das Event bestehen (aktiv). Sie können also auch zu einem späteren Zeitpunkt erkennen, ob eine Digitalisierung abgeschlossen wurde.

Um das Event zurückzusetzen, muß explizit die Funktion *ResetEvent()* aufgerufen werden. Ansonsten würden Sie beim nächsten Aufruf von *WaitForSingleObject()* sofort auf die noch anstehende Signalisierung des letzten Events reagieren.

Sie sollten *ActivateFieldInterrupt()* daher nur bei gestopptem Grabber aufrufen. Gegebenenfalls kann es sinnvoll sein, sicherheitshalber ein *ResetEvent()* durchzuführen, um alte Signalisierungen zu löschen.

- Sofort nach Eintreffen des Events können Sie mit Hilfe der *DataPresent()* – Funktion herausfinden, welches Halbbild digitalisiert wurde. Bei kontinuierlicher Digitalisierung geschieht dies durch Abfrage der T_EVEN / T_ODD – Flags. Beachten Sie, daß diese Flags ihren Zustand nur für 20 ms nach dem Ende der Digitalisierung behalten, da dann schon wieder ein neues Halbbild eingelaufen ist.

Hinweis: Die Exklusiv-Oder-Verknüpfung von T_EVEN und T_ODD ergibt die Parität des fertig digitalisierten Halbbilds (1=ODD, 0=EVEN): $field_ready := T_EVEN \wedge T_ODD$;

- Der Event wird vom Grabber-Treiber mit einer nur sehr geringen Verzögerung erzeugt. Beachten Sie jedoch, daß die Signalisierung durch das System mit einer (variablen) Zeitverzögerung erfolgen kann.

⇔ ③ Erkennung eines Videosignals

WORD Get_Signal_Status(WORD nDevNo);

Returnwert: 0 = kein Videosignal am Eingang vorhanden
1 = undefiniert
2 = Videosignal liegt an
3 = Videosignal liegt an und Zeile ist eingerastet

Mit Hilfe dieser Funktion kann festgestellt werden, ob am eingestellten Kanal eine Kamera angeschlossen ist oder nicht. Des weiteren kann die Signalqualität beurteilt werden.

Der Returnwert 0 zeigt an, daß keine Videoquelle am aktuell ausgewählten Kanal anliegt. Werden für 31 Zeilen keine Synchronimpulse detektiert, so wird angenommen, daß kein Signal anliegt.

Wird 2 zurückgegeben, so ist eine Videoquelle erkannt worden. Zufällige Störungen am Eingang können jedoch zu einer fehlerhaften Aussage führen.

Der Rückgabewert 3 zeigt an, daß ein stabiles Videosignal anliegt, das mit sehr großer Wahrscheinlichkeit von einer Bildquelle kommt. Der Wert wird generiert, wenn der Horizontal-Synchronimpuls innerhalb von ± 1 Clock von der erwarteten Position gefunden wird. Dies muß für 32 aufeinanderfolgende Zeilen gegeben sein. Umgekehrt wird der „HLOCK“-Zustand verlassen, wenn das Kriterium für 32 Zeilen nicht erfüllt wird. Es handelt sich also um ein sehr zuverlässiges Kriterium. Einige Bildquellen, wie z.B. Videorecorder, liefern von Natur aus kein stabiles Zeilentiming. Bei diesen Geräten kann es möglich sein, daß der Return-Wert 3 nie erreicht wird. Die Bilddigitalisierung erfolgt wegen Verwendung des ULTRALOCK™-Synchronisationsverfahrens in der Regel trotzdem einwandfrei.

⇔ ③ ☆ **Anzahl der digitalisierten Bilder verfolgen**

Beim eGrabber-4plus ist es möglich, die Anzahl der digitalisierten Bilder zu verfolgen. Dazu stehen die folgenden zwei Funktionen zur Verfügung.

Diese Funktionen sind nicht kompatibel zum pciGrabber-4.

BYTE Get_CaptureCounter (WORD nDevNo)

Rückgabewert: Anzahl der gegrabten Fields modulo 256

Die Funktion gibt die Anzahl der digitalisierten Halbbilder wieder. Das Ergebnis ist ein Byte-Wert, der Zähler springt also von 255 wieder auf 0 um.

void Reset_CaptureCounter (WORD nDevNo)

Der Aufruf dieser Routine setzt den Halbbildzähler auf Null zurück.

⇔ ④ **Einstellen der Bildhelligkeit**

void Set_Brightness(WORD nDevNo, short nBright);

nBright: Bildhelligkeit (-128..127)

Stellt das Register für die Bildhelligkeit im Videoprozessor ein. Der Wert bestimmt eine Konstante, die zu dem Helligkeitswert eines Pixels im Videoprozessor addiert wird. Die Helligkeit läßt sich so im Bereich

-50 % bis +49,6 % variieren. Ein LSB entspricht einer Helligkeitsänderung um 0,39 %:

$$nBright = \text{Helligkeit [\%]} \cdot 2,5601 [1/\%]$$

↔ ④ **Helligkeitseinstellung zurücklesen**

short Get_Brightness(WORD nDevNo);

Returnwert: Inhalt des Helligkeitsregisters des Videoprocessors

Diese Funktion ermöglicht das Auslesen des aktuellen Helligkeitswerts aus dem Videoprocessor.

↔ ④ **Einstellen des Kontrasts**

void Set_Contrast(WORD nDevNo, WORD nContr);

nContr: Bildkontrast (0..511)

Stellt den Kontrast des aufgenommenen Bilds ein. Der Kontrastwert stellt einen konstanten Faktor dar, mit dem der Helligkeitswert der Pixeldaten im Videoprocessor (entsprechend skaliert) multipliziert wird. Die Einstellung ist im Bereich 0 % bis 236,57 % möglich:

$nContr = \text{Kontrast [\%]} \cdot 2,1598 [1/\%]$

↔ ④ **Kontrasteinstellung zurücklesen**

WORD Get_Contrast(WORD nDevNo);

Returnwert: momentan eingestellter Kontrastwert

Die Funktion liest den Inhalt der Kontrast-Register aus dem Videoprocessor und liefert sie als Ganzzahl-Wert zurück.

⇔ ④ Einstellen der Farbsättigung

**void Set_Saturation(WORD nDevNo,WORD nSat_U,
WORD nSat_V);**

nSat_U: Sättigung des U-Farbanteils (0..511, Default = 254)

nSat_V: Sättigung des V-Farbanteils (0..511, Default = 180)

Über diese Funktion wird die Farbsättigung des Bilds getrennt für U- und V-Farbanteil eingestellt. Es handelt sich dabei um eine Veränderung des Verstärkungsfaktors getrennt für die beiden Farbanteile.

Normalerweise ist das Verhältnis Werte $nSat_U$ und $nSat_V$ gleich. Durch eine Differenz von U- und V-Anteil können Farbstiche (z.B. von nicht abgeglichenen Farbkameras) ausgeglichen werden, also die Farbe des Bilds verändert werden.

$nSat_U = \text{U-Sättigung [\%]} \cdot 2,5400 [1/\%]$; 0% bis 201,18 %

$nSat_V = \text{V-Sättigung [\%]} \cdot 2,1396 [1/\%]$; 0% bis 238,83 %

⇔ ④ Inhalte der Farbsättigungsregister lesen

WORD Get_Sat_U(WORD nDevNo);

WORD Get_Sat_V(WORD nDevNo);

Returnwert: Wert der aktuellen U- bzw. V-Farbsättigung

Der Inhalt der Farbsättigungsregister kann durch Aufruf dieser Routinen ermittelt werden.

↔ ④ **Farbton korrigieren (nur bei NTSC)**

void Set_Hue(WORD nDevNo, short nHue);

nHue: Farbton, Phasenlage des Farbträgers (-128..127)

Mit dieser Funktion wird bei Digitalisierung von NTSC-Farbbildern der Farbton eingestellt, indem die Phasenlage des Farbträgers variiert wird. Bei PAL hat dieser Wert keine Bedeutung, da hier durch das Farbsystem selbst Phasenfehler automatisch ausgeglichen werden.

Ein LSB entspricht einer Phasenwinkelkorrektur um $0,7^\circ$, der Farbträger kann somit im Bereich $-89,3^\circ$ bis $+90^\circ$ nachgestellt werden. Damit der Farbdekoder unter PAL korrekt arbeitet, muß dieser Wert auf 0 gesetzt werden.

↔ ④ **Inhalt des Farbton-Registers zurücklesen**

short Get_Hue(WORD nDevNo);

Returnwert: äquivalent der eingestellten Phasenlage des Farbträgers

Diese Funktion dient dem Rücklesen des eingestellten Hue-Wertes.

⇔ ④ Ein-/Ausschalten des Luma-Tiefpassfilters

void Set_LDec(WORD nDevNo, WORD nOn, WORD nHFilt);

nOn: 1 = Luma Decimation einschalten
 0 = Luma Decimation ausschalten

nHFilt: 0 = Automatische Filterauswahl
 1 = CIF Filter
 2 = QCIF Filter
 3 = ICON Filter

Bei kleinen Bildformaten erzielt man eine bessere Bildwiedergabe, wenn die Auflösung des Eingangssignals reduziert wird (man also die Schärfe des Bilds an die spätere Auflösung anpaßt). Dazu kann mit dieser Routine optional ein Tiefpaßfilter in den Luma-Pfad eingeschaltet werden.

Mit dem Parameter *nHFilt* wird der verwendete Filter an die Bildgröße angepaßt:

Automatische Filterwahl macht die Filtereinstellung abhängig von der gesetzten Bildgröße (*siehe Set_Image*). Daneben kann die Filter manuell auf eine der normierten Bildformate CIF (= 1/2 Vollbild), QCIF (1/4 Vollbild) und ICON (1/8 Vollbild) angepaßt werden.

Default: Luma-TP ist ausgeschaltet.

⇔ ④ Ein-/Ausschalten des Testbilds

void Set_ColorBars(WORD nDevNo, WORD nColorBars);

nColorBars: 0 = Testbild ausblenden
 1 = Testbild einschalten

Diese Funktion blendet ein Testbild ein. Das Testbild besteht aus senkrechten, farbigen Balken. Das Testbild ist unabhängig von einem Eingangssignal. Um das Bild vollständig sehen zu können, sollte die Bildgröße etwa CIF-Format betragen.

⇔ ④ **Kontrolle des Wertebereichs**

```
void LumaControl(WORD nDevNo,  

                 WORD nRange,  

                 WORD nCore);
```

nRange: 0 = Luma Spannweite 16 - 253
 1 = Luma Spannweite 0 - 255
 nCore: 0 = 0 alle Helligkeitswerte werden übermittelt
 2 = 16 alle Helligkeitswerte <= 16 werden als 0 interpretiert
 3 = 32 alle Helligkeitswerte <= 32 werden als 0 interpretiert

Mit dieser Funktion kann das Ausgabeformat der Helligkeits- und Farbwerte an die Applikation angepaßt werden.

nRange bestimmt den Wertebereich der Helligkeit (mögliche Grauwerte).

- *nRange* = 0 entspricht dem normalen Wertebereich, der in CCIR 601 spezifiziert ist. Der Helligkeitsbereich ist damit auf die Werte 16 bis 253 begrenzt, wobei der Wert $Y = 16$ schwarz entspricht. Der Farbwertebereich ist 2...253 mit $Cr/Cb = 128$ als Null (vorzeichenbehaftet).
- *nRange* = 1 ermöglicht die Nutzung des vollen Wertebereichs, das ist für Y der Bereich 0...255 mit 0 = schwarz, der Chroma-Bereich ist wie bei *nRange* = 0 definiert.

⇔ ⑥ Daten über den GPIO-Port lesen und ausgeben

An der Stiftleiste Option-Port sind 12 I/O-Leitungen verfügbar, die über die folgenden drei Funktionen gesteuert werden können:

```
void Set_GPIO_Direction (WORD nDevNo, WORD nDirection);
```

```
void Set_GPIO_Data (WORD nDevNo, WORD nData);
```

```
WORD Get_GPIO_Data (WORD nDevNo);
```

nDirection: Richtungskontrolle für GPIO-Port (0...4095), bitcodiert

nData: Daten, die über den GPIO-Port ausgegeben werden sollen

Rückgabewert Get_GPIO_Data(): Daten, die vom GPIO-Port gelesen werden

Der eGrabber-4plus besitzt 12 Erweiterungsportpins, welche getrennt zum Einlesen oder zum Ausgeben von beliebigen Digitalsignalen genutzt werden können. Mit den GPIO-Funktionen steuern Sie den Erweiterungsport. Sie können definieren, welche der Pins als Eingang und welche als Ausgang fungieren, Ausgangspins auf High- oder Low-Pegel setzen und den Status von Eingangspins lesen.

Mit Hilfe der Funktion **Set_GPIO_Direction** kann jeder Portpin getrennt als Ein-oder Ausgabepin konfiguriert werden. Hierzu werden die unteren 12-bits des Parameters *nDirection* ausgewertet. Ist ein Bit auf 1 gesetzt, wird der entsprechende Portpin auf Ausgabe konfiguriert. Eine 0 bewirkt eine Umschaltung des Portpins auf Einlesen.

Achtung!

Achten Sie darauf, daß ein Portpin erst dann auf Ausgabe geschaltet wird, wenn kein externes Signal an diesem Pin anliegt. Ansonsten ist es möglich, daß der Pin zerstört wird.

Beim Starten des Rechners sind alle Pins auf Eingabe konfiguriert. Beachten Sie, daß die Pins dann hochohmig sind und damit der logische Pegel nicht definiert ist. Um das Verhalten von angesteuerten Komponenten während des Starten des Rechners bis zur Konfigurierung des GPIO-Ports zu bestimmen, müssen daher hardwaremäßig entsprechende Maßnahmen vorgesehen werden (z.B. Pull-Up-Widerstände).

Mit **Set_GPIO_Data** können Sie denjenigen Portpins, die auf „Ausgang“ geschaltet sind, den Pegel „High“ = 1 oder „Low“ = 0 zuordnen. Bei `nData` handelt es sich wieder um einen 12-bit-Wert, wobei jedes Bit einem Pin zugeordnet ist. Es wirken sich nur die Bits aus, für die die Datenflußrichtung auf „Ausgang“ gestellt ist.

Die Funktion **Get_GPIO_Data** liest den Zustand der als „Eingang“ geschalteten Portpins ein und liefert diesen als 12-bit-Wert zurück. Für die auf „Ausgang“ geschalteten Pins wird der Wert zurückgeliefert, auf den der Pin gesetzt wurde.

⇔ ⑤ ☆ Daten über die I²C-Schnittstelle übertragen

Mit diesen Funktionen können Sie Bausteine, die an der I²C-Schnittstelle angeschlossen sind, lesen und beschreiben.

Wie Sie Bausteine oder Geräte an den I²C-Bus anschließen, können Sie *Kapitel 1.7* entnehmen.

Achtung!

Das auf der Grabberkarte befindliche I²C-EEPROM ist vor versehentlichem Überschreiben geschützt. Daher sind keine Zugriffe auf den Geräteadressenbereich 0xA0 bis 0xA3 möglich.

Zum Zugriff auf den internen EEPROM-Speicherbereich benutzen Sie bitte die dafür vorgesehenen gesonderten Funktionen.

void I2C_Set_BR_Mode (WORD nDevNo, BYTE bMode)

bMode Baudrate
eGrabber-4plus: 0 = 99,2 kHz, 1 = 396,8 kHz
pciGrabber-4: 0 = 33 kHz, 1 = 290 kHz

Diese Funktion bestimmt die Baudrate, mit der die Übertragung auf dem I²C-Bus stattfindet. Es sind eine hohe und eine niedrige Übertragungsraten wählbar.

BYTE I2C_ReadByte (WORD nDevNo, BYTE bChipAddress, BYTE bSubAddress, BYTE *bByteRead)

bChipAddress: Geräteadresse des I²C-Geräts auf dem Bus
bSubAddress: Speicher-Adresse innerhalb des I²C-Geräts
*bByteRead: Zeiger auf eine Byte-Variable, in die das Ergebnis geschrieben wird

Returnwert: SUCCESS, NOACK, INVALID_ADDRESS

Mit *I2C_ReadByte* wird ein Byte aus einer Speicherstelle eines I²C-Geräts gelesen werden. Das Ergebnis wird in einer Variablen vom Typ Byte abgelegt, die zuvor definiert werden muß.

Als Returnwert gibt die Funktion einen Fehlercode zurück.. NOACK zeigt an, daß sich unter der angegebenen Geräteadresse kein I²C-Gerät gemeldet hat. INVALID_ADDRESS bedeutet, daß versucht wurde, auf den geschützten Bereich des auf dem Grabber befindlichen EEPROMs zuzugreifen.

**BYTE I2C_WriteByte (WORD nDevNo, BYTE bChipAddress,
BYTE bSubAddress, BYTE bData)**

bChipAddress: Geräteadresse des I²C-Geräts auf dem Bus
bSubAddress: Speicher-Adresse innerhalb des I²C-Geräts
bData: Byte, das in die spezifizierte Adresse geschrieben wird

Returnwert: SUCCESS, NOACK, INVALID_ADDRESS,
WRITE_FAILED

Schreibt ein Byte in eine Speicherstelle des spezifizierten I²C-Geräts.
Als Returnwert gibt die Funktion einen Fehlercode zurück (s.o.).

Hinweis:

Manche Geräte – wie z.B. EEPROMs – benötigen u.U. intern längere Verarbeitungszeiten für einen Schreibbefehl als der I²C-Buszyklus dauert. Folgen mehrere Schreibbefehle schnell aufeinander, so liefern diese Geräte eine Fehlermeldung (NOACK) zurück, da sie evtl. noch nicht wieder schreibbereit sind.

Um dies zu vermeiden, können Sie nach dem Schreiben auf einen solchen Baustein einen Lesezugriff durchführen und prüfen, ob der zurückgelesene Wert mit dem Eingeschriebenen übereinstimmt. Erst wenn dies der Fall ist, wurde der interne Schreibvorgang abgeschlossen und Sie können erneut schreibend auf den Baustein zugreifen.

Beachten Sie diesbezüglich auch die Dokumentation des I²C-Geräts.

↔ ⑤ ☆ **internes EEPROM verwenden**

Der pciGrabber-4*plus* besitzt einen internen, nichtflüchtigen Speicher, der vom Anwender benutzt werden kann, um beliebige Parameter zu speichern.

Insgesamt stehen in diesem Speicher 252 Bytes zur Verfügung.

Hinweis:

Da das Vorgängermodell eGrabber-2 keinen internen Speicher besitzt, sind die nachfolgenden Funktionen nicht kompatibel.

**BYTE I2C_ReadEEProm (WORD nDevNo, BYTE bSubAddress,
BYTE *bByteRead)**

bSubAddress: zu lesende Speicheradresse (0x00 ... 0xFB)

*bByteRead : Zeiger auf eine Byte-Variable, in die das
Ergebnis geschrieben wird

Returnwert: Fehlercode = SUCCESS, NOACK

Liest aus dem EEPROM einen Byte-Wert aus. Spezifiziert wird bei Aufruf die Speicheradresse, die gelesen werden soll. Das Ergebnis wird in einer Variablen vom Typ Byte abgelegt, die zuvor definiert werden muß.

Die Funktion gibt als Returnwert einen Fehlercode zurück (*siehe I2C_ReadByte*).

**BYTE I2C_WriteEEProm (WORD nDevNo, BYTE bSubAddress,
BYTE bData)**

bSubAddress: Speicheradresse, auf die geschrieben wird (0x00...0xFB)

bData: Datenbyte, das geschrieben wird

Returnwert: Errorcode = SUCCESS, NOACK, WRITE_FAILED

Schreibt ein Byte in das interne EEPROM. Spezifiziert wird die gewünschte Speicheradresse und das dorthin zu schreibende Byte. Die Funktion liefert einen Fehlercode zurück (*siehe I2C_WriteByte*).

Hinweis:

Die Lebensdauer des internen EEPROM-Speichers ist mit 1 Million Schreibzugriffen spezifiziert. Die Anzahl der Lesezugriffe ist nicht begrenzt.

⇔ 5 ☆ Kontrollieren des I/O-Pins

Der eGrabber-4*plus* besitzt einen extern verfügbaren, transistorgetriebenen I/O-Pin, der wahlweise zur Eingabe oder Ausgabe von Steuersignalen dienen kann. Dieser Pin kann mit den folgenden beiden Funktionen kontrolliert werden.

Wie Sie Geräte an den I/O-Pin anschließen, erfahren Sie in *Kapitel 1.6*.

Diese Funktion ist beim eGrabber-2 nicht vorhanden.

void Set_Ext_IO (WORD nDevNo, BYTE bData)

bData: Steuerwert, 0 = CLOSE, 1 = HI-Z

Mit dieser Funktion wird der Schaltausgang des Portpins gesteuert. Der Transistor des Ausgangspins arbeitet wie ein Schalter gegen Masse. Wird der Steuerwert auf 0 gesetzt, so wird dieser Schalter geschlossen und es kann Strom in den I/O-Pin hineinfließen.

Wird der Steuerwert auf 1 gesetzt, so wird der Transistor gesperrt (hochohmig / Schalter geöffnet).

Die technische Spezifikation des I/O-Ausgangs entnehmen Sie bitte *Kapitel 1.2*.

Hinweis:

Beim Start des Rechners ist der Transistor gesperrt (hochohmig). Falls extern mittels eines angeschlossenen Pull-up-Widerstands ein logischer Pegel erzeugt wird, so ist dieser dann logisch „1“. Angeschlossene Verbraucher wie z.B. ein Relais sind stromlos (abgeschaltet).

BYTE Read_Ext_IO (WORD nDevNo)

nDevNo = Device Number

Rückgabewert: Zustand des I/O-Pins

Mit dieser Funktion läßt sich der Zustand des I/O-Pins zurücklesen. Damit kann der Pin als Eingang verwendet.

Liegt die Spannung am I/O-Pin im Bereich „logisch 0“ so liefert die Funktion eine 0 zurück, liegt sie im Bereich „logisch 1“ so wird eine 1 zurückgeliefert. Die entsprechenden Spannungsbereiche entnehmen Sie bitte der Spezifikation in *Kapitel 1.2*.

Achtung!

Um den I/O-Pin als Eingang benutzen zu können, muß der Transistor gesperrt sein (hochohmig).

⇔ **5 Direktes Lesen und Beschreiben von Registern**

**WORD Read_Local_DWord (WORD nDevNo,
 WORD nRegister_Number,
 DWORD *IContent);**

nRegister_Number: Nummer des Registers
IContent: Inhalt des Registers

**WORD Write_Local_DWord (WORD nDevNo,
 WORD nRegister_Number,
 DWORD IContent);**

nRegister_Number: Nummer des Registers
IContent: Inhalt des Registers

Fast alle Funktionen des eGrabber-4plus lassen sich über die Routinen des Treibers steuern. Wir empfehlen ausdrücklich die Verwendung der Standard-Funktionen.

Für den Fall, daß der Anwender direkt Register des eGrabber-4plus beeinflussen möchte, stehen diese beiden Funktionen zur Verfügung, mit denen sich alle Register des Videoprozessors lesen und beschreiben lassen. *Sollten Sie weitere Informationen über die Register des eGrabber-4plus benötigen, wenden Sie sich bitte an den PHYTEC-Support.*

Achtung!

Manche Register des eGrabber-4plus müssen in der vorliegenden Gerätekonfiguration mit bestimmten Werten belegt werden, um die Funktion der Karte sicherzustellen.

Prüfen Sie daher vor Änderungen genau die Bedeutung der veränderten Register / Bits und die Auswirkungen.

PHYTEC kann keine Haftung für Schäden übernehmen, die durch Manipulation der Register möglicherweise entstehen.

5.3 Treiber für DOS-Anwendungen

Auf der CD finden Sie in den Verzeichnissen unter **PCIGRAB4\DRIVER\DOS\DRIVER** die Treiber-Bibliothek PCI4GRAB, mit der Sie den eGrabber-4plus unter DOS ansprechen können.

5.3.1 Voraussetzungen

Zum Betrieb des eGrabber-4plus unter DOS ist es erforderlich, daß der gesamte physikalische Adreßbereich des PCs linear adressierbar ist. Dies ist deshalb notwendig, da das PCI-BIOS den Registerbereich der Karte beliebig im Speicher konfiguriert, ohne daß der Anwender darauf Einfluß nehmen kann. Gewöhnlich werden hohe Adressen oberhalb des RAM-Bereichs verwendet, für die besondere Mechanismen zur Adressierung angewendet werden müssen.

Achtung!

Um den DOS-Treiber benutzen zu können, muß der DOS-Extender *DOS/4GW* installiert werden. Programme, die unter DOS auf den eGrabber-4plus über die Treiberrouinen zugreifen sollen, müssen für die Verwendung mit *DOS/4GW* konzipiert sein.

DOS/4GW von Rational Systems ist ein DOS-Extender, der Protected-Mode-Zugriffe ermöglicht. Viele Compiler unterstützen die Anwendung von DOS/4GW und ermöglichen die Einbindung in das EXE-File der Anwendung.

Achtung!

Manche DOS-Systemprogramme können Probleme bei der Zusammenarbeit mit DOS/4GW verursachen. Schwierigkeiten können insbesondere bei der Verwendung von *EMM386.EXE* auftreten. Wir empfehlen daher, DOS/4GW und *EMM386.EXE* nicht gleichzeitig anzuwenden.

5.3.2 Entwicklungsplattform

Die DOS-Treibersoftware wurde objektorientiert unter C++ gestaltet. Es ist möglich, den Treiber auch in Programme, die in anderen Programmiersprachen erstellt werden, einzubinden. Hierbei sind jedoch möglicherweise Unterschiede in den Aufrufkonventionen zu beachten.

Voraussetzung für die Verwendung der Library ist, daß der benutzte Compiler 32-bit-Code Anwendungen unterstützt, da es sich bei PCI4GRAB um eine 32-bit-Library handelt.

Als Memory-Modell ist *32-bit-Flat* zu wählen. Wir empfehlen die Einstellung der Option '*80386 Register Based Calling*'.

Zur Programmierung des Treibers wurde der Watcom C/C++-Compiler der Fa. Powersoft in der Version 10.6 verwendet. Dieser Compiler bietet die Möglichkeit der 32-bit-Programmierung und der Einbindung von DOS/4GW.

Der Treiber wird als Library (***.LIB**) - Datei geliefert. Sie kann mit allen Compilern eingebunden werden, die das verwendete Format unterstützen. Die Headerfiles (***.H**) werden in das Anwenderprogramm eingebunden (*include*). Sie beibhalten die Deklaration der verschiedenen Funktionen, welche von der Library zur Verfügung gestellt werden.

Beim Start des Anwenderprogramms muß der DOS/4GW-Extender geladen sein, sofern der Compiler nicht die Einbindung und damit den automatischen Start unterstützt. DOS/4GW ermöglicht den DPMI-Zugriff, der für die Funktion der Treiberrouninen erforderlich ist.

5.3.3 Funktionen des DOS-Treibers *PCI4GRAB*

In der Bibliothek *PCI4GRAB.LIB* ist die Klasse *PCI_GRABBER4* enthalten, die alle Funktionen zur Initialisierung und Konfigurierung zur Verfügung stellt. Über sie wird auch der Grabbvorgang gesteuert. Mit einem Objekt dieser Klasse werden alle im System befindlichen eGrabber-4plus kontrolliert.

Zur besseren Übersicht sind die Funktionen in fünf Gruppen aufgeteilt. Die Nummer der jeweiligen Gruppe ist der Funktionsbeschreibung in einem schwarzen Kreis vorangestellt.

Die Funktionen sind wie folgt klassifiziert:

❶ Routinen zur Initialisierung

Diese Funktionen müssen Sie ein Mal vor der Verwendung des Grabbers aufrufen, damit der Grabber korrekt arbeitet.

❷ Routinen, die den Grabber zur Bildaufnahme einstellen

Funktionen aus dieser Gruppe stellen den Grabber auf die angeschlossene Bildquelle (Kamera) ein und legen fest, wie das vom Grabber erzeugte Ergebnisbild im Speicher aussieht (Bildgröße, Farbformat usw.). Sie sollten für jede Funktion aus dieser Gruppe prüfen, ob sie verwendet werden muß und welche Parameter zutreffend sind. Gegebenenfalls werden diese Routinen im Programmverlauf mehrfach aufgerufen (z.B. wenn der Eingangskanal umgeschaltet werden soll oder die Bildgröße verändert wird).

❸ Routinen zur Durchführung und Kontrolle des Grabbvorgangs

Mit diesen Funktionen starten Sie die Bild-Digitalisierung, überwachen den Grabbvorgang und beenden die Digitalisierung.

❹ Routinen zur Einstellung von Bildparametern

Funktionen aus dieser Gruppe ermöglichen die Einstellung von Parametern wie Bildhelligkeit, Kontrast, Farbsättigung usw. Sie müssen nicht verwendet werden, können aber jederzeit aufgerufen werden, um das Ergebnisbild an die Bedürfnisse anzupassen.

⑤ Routinen zur Steuerung der Erweiterungsports

Unter diese Kategorie fallen alle Funktionen, die nicht direkt mit der Bildaufnahme zu tun haben, sondern die weiteren Features des Grabbers wie I/O-Port, I²C-Schnittstelle usw. betreffen. Diese Funktionen müssen nur aufgerufen werden, wenn ein entsprechendes Grabber-Feature verwendet wird.

Die meisten Funktionen sind in der Windows- und in der DOS-Treiberversion identisch. Aufgrund des Treibermodells gibt es jedoch einige wenige Unterschiede. Um die Portierung von Programmen zu erleichtern, werden folgende Symbole verwendet:

- ↔ Der Funktionsaufruf ist unter DOS und Windows gleich
Beachten Sie jedoch, daß sich die Variablentypen betriebs-systembedingt unterscheiden können.
- ☒ Diese Funktion ist Windows-spezifisch

DOS Diese Funktion ist DOS-spezifisch

Achtung!

In allen nachfolgend beschriebenen Routinen wird der Parameter `nDevNo` verwendet. Dieser Parameter identifiziert den gewünschten eGrabber-4plus, wenn sich mehrere eGrabber-4plus im System befinden. Die Anzahl der vorhandenen eGrabber-4plus kann mit der Funktion **Max_Device_Number()** bestimmt werden.

Kompatibilität zum eGrabber-2

Der Treiber ist grundsätzlich abwärtskompatibel zum eGrabber-2 (EPC-032 bzw. EPC-032-X1). Programme, die für den eGrabber-2 geschrieben wurden, arbeiten also auch mit dem eGrabber-4*plus*.

Um die neuen Funktionen des eGrabber-4*plus* nutzen zu können, besitzt der Treiber einige neue oder anders gestaltete Funktionen.

Neue Programme, die diese Funktionen verwenden, können nicht mit dem eGrabber-2 zusammenarbeiten.

Funktionen, die nicht mit dem eGrabber-2 kompatibel sind, sind im folgenden mit einem Stern (☆) gekennzeichnet.

Wenn Sie eine bestehende Applikation mit den neuen Features des eGrabber-4*plus* ausstatten möchten, beachten Sie bitte die mit ☆ gekennzeichneten Funktionen.

⑤ Fehlermeldungen auswerten

short Get_Error(void);

Returnwert: 0 = kein Fehler aufgetreten
 1 = Device Nummer nicht gefunden
 2 = Register nicht vorhanden
 3 = Initialisierung fehlgeschlagen
 4 = Grabber nicht gefunden
 5 = unbekannter Parameterwert

⇔ ❶ Anzahl der vorhandenen eGrabber-4*plus* bestimmen

unsigned short Max_Device_Number();

Rückgabewert: Anzahl der gefundenen eGrabber-4*plus*

⇔ ❶ Grabber und Treiber nach dem Einschalten initialisieren

void Initialize(unsigned short nDevNo);

↔ **❶ Informationen über die installierten Grabber abfragen**

**short Read_GrabberInfo(unsigned short nDevNo,
 unsigned short wInfoType)**

wInfoType: spezifiziert, welche Grabbereigenschaft abgefragt wird

Returnwert: Wert der spezifizierten Eigenschaft

↔ **❷ Grabbername als Klartextstring lesen**

**short Read_OrderCode (unsigned short nDevNo,
 unsigned char* sCodeString,
 unsigned long dwSizeOfString)**

*sCodeString: Zeiger auf einen zuvor deklarierten String
 (25 Bytes min.), in das die Funktion den
 Grabbernamen einschreibt.

dwSizeOfString: Größe des reservierten Arrays

Returnwert: Errorcode

↔ **❸ Grabber auf verwendetes Farbsystem einstellen**

**void Set_Color_System(unsigned short nDevNo
 unsigned short nColSys);**

nColSys: Code für Farbsystem

↔ **❹ Erkennung des Videoformats**

short Get_Video_Status(unsigned short nDevNo);

Returnwert: 0 = 525 Zeilen Format (NTSC / PAL-M)
 1 = 625 Zeilen Format (PAL / SECAM)

⇔ **② Einstellen des Composite-Modus (Composite-Eingänge)**

void Set_Composite(unsigned short nDevNo);

⇔ **② ☆ Einstellen des S-Video-Modus'**

**unsigned char Set_S_VideoEx(unsigned short nDevNo,
unsigned char input);**

input: MINIDIN = Eingang ist Mini-DIN Buchse
 COMBI = Eingang ist Combi-Stecker (X3)
 AUTO = automatische Einstellung

Rückgabe: MINIDIN = Einspeisung von Mini-DIN-Buchse
 COMBI = Einspeisung von Combi-Buchse
 NO_SIGNAL = kein Signal gefunden (AUTO-
 Mode)
 NOT_SUPPORTED = eGrabber-2 und Einstellung
 COMBI

⇔ **② ☆ Einstellen des Eingangskanals**

void Set_ChannelEx(WORD nDevNo, WORD nChannel);

nChannel: Einzustellender Eingangskanal (1..9 / 1..3)

⇔ **② Ein-/Ausschalten der Farbfalle für s/w-Betrieb**

void Set_BW(unsigned short nDevNo, unsigned short nOn);

nOn: 0 = Composite-Signal am Eingang (Farbfalle einschalten),
 1 = s/w-Signal am Eingang (Farbfalle ausschalten)

↔ **② Ein-/Ausschalten des Interlaced-Modus'**

**void Set_Interlace (unsigned short nDevNo,
 unsigned short nInterlace);**

nInterlace: 0 = Non-Interlace
 1 = Interlace

↔ **② Ein-/Ausschalten der AGC**

**void Set_AGC (unsigned short nDevNo,
 unsigned short nCAGC,
 unsigned short nAGC,
 unsigned short nCrush);**

nCAGC: 0 = Chroma AGC ausschalten
 1 = Chroma AGC einschalten

nAGC: 0 = AGC einschalten
 1 = AGC ausschalten

nCrush: 0 = Nicht-adaptive AGC
 1 = Adaptive AGC

↔ **② Farbtöter Ein-/Ausschalten**

**void Set_CKill (unsigned short nDevNo,
 unsigned short nCKill);**

nCKill: 0 = Ausschalten des Farbtöters
 1 = Einschalten des Farbtöters

⇔ ② **Auslassen von Fields, Frames aus dem Videosignal**

**void TemporalDect (unsigned short nDevNo,
 unsigned short nDecField,
 unsigned short nFldAlign,
 unsigned short nDecRat);**

nDecField: 0 = Frame(s) auslassen
 1 = Field(s) auslassen
nAlign: 0 = odd Field wird als erstes ausgelassen
 1 = even Field wird als erstes ausgelassen
nDecRat: Anzahl auszulassender Fields / Frames

② **DOS Bildgröße und Skalierung einstellen**

**void Set_Image (unsigned short nDevNo,
 unsigned short nOhpos,
 unsigned short nOvpos,
 unsigned short nOhsize,
 unsigned short nOvsize,
 unsigned short nOppl,
 unsigned short nOlines,
 unsigned short nOColformat,
 unsigned char *pOImgBuf,
 unsigned short nEhpos,
 unsigned short nEvpos,
 unsigned short nEhsize,
 unsigned short nEvsize,
 unsigned short nEppl,
 unsigned short nElines,
 unsigned short nEColformat,
 unsigned char *pEImgBuf,
 unsigned short nColSystem,
 unsigned short nInterlaced,
 unsigned short nSingleShot);**

nOhpos, nOvpos :	Position der linken oberen Ecke des Odd-Bildausschnitts im Videobild (hpos = horizontal, vpos = vertikal)
nOhsize :	Größe des Odd-Bildausschnitts in X-Richtung
nOvsize :	Größe des Odd-Bildausschnitts in Y-Richtung
nOppl :	gewünschte Größe des Odd-Videobilds in X-Richtung (ppl = Pixel Per Line)
nOlines :	gewünschte Zeilenzahl des Odd-Videobilds
nOColformat:	gewünschtes Farbformat: (RGB32, RGB24, RGB16, RGB15, Y8, YCrCb 4:2:2, YCrCb 4:1:1)
pOImgBuf :	Adresse des Odd-Bildspeichers
nEhpos, nEvpos :	Position der linken oberen Ecke des Even-Bildausschnitts im Videobild (hpos = horizontal, vpos = vertikal)
nEhsize :	Größe des Even-Bildausschnitts in X-Richtung
nEvsize :	Größe des Even-Bildausschnitts in Y-Richtung
nEppl :	gewünschte Größe des Even-Videobilds in X-Richtung (Pixel Per Line)
nElins :	gewünschte Zeilenzahl des Even-Videobilds
nEColformat:	gewünschtes Farbformat: (RGB32, RGB24, RGB16, RGB15, Y8, YCrCb 4:2:2, YCrCb 4:1:1)
pEImgBuf :	Adresse des Even-Bildspeichers
nColSystem :	Code für Farbsystem (vgl. <i>Set_Color_System</i>)
nInterlace:	0 = Non-Interlace 1 = Interlace 2 = Field Aligned
nSingleShot :	0 = kontinuierliche Digitalisierung 1 = ein Bild grabben

Mit der Routine **Set_Image ()** werden Größe, Position und Skalierung des vom Grabber gelieferten Bildausschnitts für Even- und Odd-Bilder getrennt festgelegt. Des weiteren wird das Datenformat, in dem das Bild im Speicher abgelegt wird, definiert und die Adresse dieses Bildspeicherbereichs im Hauptspeicher bekannt gemacht.

Die Einstellungen können grundsätzlich für beide Halbbilder getrennt vorgenommen werden. Der Parametersatz ist entsprechend gegliedert in Parameter mit vorgestelltem 'E' = Parameter für Even-Bild und vorgestelltem 'O' = Odd-Bild - Parameter. Parameter ohne diese Kennzeichnung gelten für beide Halbbilder.

Beachten Sie, daß sich die Parameter dieser Funktion zwischen DOS- und Windows-Variante des Treibers unterscheiden.

Unter DOS müssen Sie den Bildspeicher im RAM des Computers selbst reservieren und einen Zeiger auf den 7 der reservierten Speicherbereiche übergeben.

Die Einstellung der Parameter für Auflösung, Skalierung und Position erfolgt so, wie unter Windows (*siehe dort*). Damit wird das Format des zu digitalisierenden Bildes genau definiert.

Der Grabber muß jetzt angewiesen werden, wo und in welcher Weise die erzeugten Bilddaten im Arbeitsspeicher abgelegt werden.

Dazu ist es zunächst erforderlich, einen ausreichend großen Arbeitsspeicherbereich zu reservieren.

Dies geschieht mit den bekannten Befehlen zur Allokation von Speicher in der jeweiligen Programmiersprache (z.B. *malloc(...)*). Wieviel Speicherplatz wird benötigt? Dies errechnet sich aus der Größe des Bilds (Anzahl der Pixel) und der pro Pixel benötigten Byte (Farbauflösung):

$$\text{Speicherbedarf pro Halbbild} = \text{hsize} \cdot \text{vsize} \cdot \text{pixelgröße [Byte]}$$

Der Wert *pixelgröße* bestimmt sich wie in *Tabelle 13* gezeigt.

Bei den Formaten YUV2 und BtYUV muß beachtet werden, daß 2 bzw. 8 Pixel logisch zusammengehören und die Bildauflösung entsprechend passend gewählt wird. Der errechnete Speicherplatzbedarf gilt für **ein** Halbbild (*even* bzw. *odd*).

Format	<i>pixelgröße</i> [Byte]
RGB32	4
RGB24	3
RGB16, RGB15	2
YUY2	4 Byte pro 2 Pixel
BtYUV	12 Byte für 8 Pixel
Y8	1

Tabelle 13: Speicherbedarf eines Pixel in den einzelnen Modi

Wird überwiegend mit kleinen Formaten gearbeitet, empfiehlt es sich, mit getrennten Halbbildspeichern zu arbeiten. Man allokiert in diesem Fall zwei Speicherbereiche, wodurch man zwei Zeiger erhält, welche die Anfangsadresse dieser Bereiche angeben. Diese Zeiger übergibt man als Argumente *pOImgBuf* und *pEImgBuf* für Odd- und Even-Halbbild der Funktion. Der Parameter *nInterlaced* wird in diesem Fall auf 0 gesetzt, um anzuzeigen, daß die Halbbilder in getrennte Speicherbereiche geschrieben werden sollen.

Werden Vollbilder benötigt, weil die vertikale Auflösung mehr als 288 Zeilen betragen soll, so kann der Grabber angewiesen werden, dieses Vollbild in einem Speicherbereich komplett aufzubauen. Er nimmt dann die erforderliche Verzahnung der beiden Halbbilder automatisch vor. Wird diese Option gewünscht, so setzt man *nInterlaced* = 1.

In diesem Fall wird in *pEImgBuf* der Anfang des Speicherbereichs übergeben, in dem das Vollbild abgelegt werden soll.

Hinweis:

Dieser Bereich muß beide Halbbilder aufnehmen können; er ist also doppelt so groß zu bemessen wie für ein Halbbild dieser Größe. Der Odd-Bereich *pOImgBuf* wird im Interlaced-Modus (*nInterlaced*=1) nicht benutzt.

Soll nur ein Halbbild (nur Odd oder nur Even) digitalisiert werden, so übergibt man für das Halbbild, das nicht digitalisiert werden soll, als Zeiger den Wert `NULL`. Wird also nur ein Even-Halbbild gewünscht, ist *pOImgBuf* = `NULL` zu setzen.

Abschließend wird angegeben, in welcher Weise die Bildaufnahme vorgenommen werden soll: Mit *nSingleShot* = 0 wird der Grabber angewiesen, kontinuierlich zu digitalisieren. Das bedeutet, daß ab dem Moment, in dem der Grabber den Befehl zum Start erhält, andauernd in Echtzeit (50 Halbbilder pro Sek.) die angegebenen Speicherbereiche mit aktuellen Bilddaten beschrieben werden. Im Halbbild-Modus (*nInterlaced* = 0) wird ein Speicherbereich beschrieben (20 ms), dann der andere (wieder 20 ms) und weiter wechselweise.

Dies bedeutet, daß jeder Halbbildspeicher mindestens 20 ms lang nicht vom Grabber belegt wird.

Im Vollbildmodus ($nInterlaced = 1$) beschreibt der Grabber den gemeinsamen Bildspeicher ständig, 20 ms die ungeraden, dann 20 ms die geraden Zeilen.

Bei der Auswertung des Bildinhalts kann es störend sein, wenn man in dem Bildfeld arbeitet, das der Grabber gerade aktualisiert, da ein Versatz von bewegten Objekten ab einer zufälligen Bildzeile auftreten kann. In diesem Fall ist es ratsam, im Start/Stop-Betrieb zu arbeiten oder die Bilddaten in zwei getrennte Halbbild-Speicherbereiche zu grabben und abwechselnd getrennt auszuwerten (synchron im 20 ms – Zyklus der Bildaktualisierung).

$nSingleShot = 1$ bewirkt, daß nur genau ein Digitalisierungsvorgang durchgeführt wird. Es werden zwei Halbbilder (ein Odd- und ein Even-Bild) aufgenommen bzw. ein Vollbild.

Der Anwender kann die Bilder aufnehmen und dann per erneutem Start-Befehl die Bildspeicher aktualisieren. Diese Betriebsart ist empfehlenswert, wenn nur gelegentlich Bilder digitalisiert werden müssen und es nicht auf Echtzeit-Darstellung ankommt.

Egal, ob kontinuierlich gegrabbt wird oder nur Einzelbilder geschossen werden: **Set_Image()** stellt nur die Parameter ein, legt also fest *wie* die Bilder aufgenommen werden sollen. Der Grabbvorgang selbst wird dadurch noch nicht gestartet. Dies geschieht durch den Befehl **Start_Grabber()** (*siehe unten*).

↔ **③ Bildaufnahme starten**

void Start_Grabber(unsigned short nDevNo);

↔ **③ Bildaufnahme stoppen**

void Stop_Grabber(unsigned short nDevNo);

↔ **③ Digitalisierungsfortschritt erkennen**

short Data_Present(unsigned short nDevNo);

Returnwert: zeigt den Digitalisierungsfortschritt an
(kann Werte zwischen 0 und 15 (4-bit) annehmen)

DOS **③ ☆ Interrupt konfigurieren**

unsigned short Set_Interrupt(unsigned short nDevNo, unsigned long nInterrupt);

nInterrupt: OFF = keine Interrupterzeugung (default)
 VSYNC = Interrupt am Ende eines Halbbilds

Returnwert: Fehlerstatus

Der Grabber kann einen Interrupt erzeugen, wenn ein Halbbild-Ende am Videoeingang erkannt wurde. Wenn ein Digitalisierungsvorgang aktiv ist, kann mit Hilfe des Interrupts ohne Polling bestimmt werden, wann dieser abgeschlossen ist. Nach Auftreten eines Interrupts muß mit *Data_Present()* abgefragt werden, was die Ursache des Interrupts war.

Hinweis: Die Interrupt-Funktion sollte nur von erfahrenen Anwendern benutzt werden.

DOS ③ ☆ Interrupt-Flag zurücksetzen

unsigned short Reset_Interrupt(unsigned short nDevNo);

Returnwert: Fehlerstatus

Nach Erkennung des Interrupts ist es erforderlich, das Interrupt-Flag zurückzusetzen. Dazu muß von der Interrupt-Behandlungsroutine die Funktion `Reset_Interrupt` aufgerufen werden.

⇔ ③ ☆ Anzahl der digitalisierten Bilder verfolgen

Beim *eGrabber-4plus* ist es möglich, die Anzahl der digitalisierten Bilder zu verfolgen. Dazu stehen die folgenden zwei Funktionen zur Verfügung.

Diese Funktionen sind nicht kompatibel zum *pciGrabber-4*.

unsigned char Get_CaptureCounter (unsigned short nDevNo)

Rückgabewert: Anzahl der gegrabten Fields modulo 256

Die Funktion gibt die Anzahl der digitalisierten Halbbilder wieder. Das Ergebnis ist ein Byte-Wert, der Zähler springt also von 255 wieder auf 0 um.

void Reset_CaptureCounter (unsigned short nDevNo)

Der Aufruf dieser Routine setzt den Halbbildzähler auf Null zurück.

⇔ ③ Erkennung eines Videosignals

short Get_Signal_Status(unsigned short nDevNo);

Returnwert: 0 = kein Videosignal am Eingang vorhanden
1 = undefiniert
2 = Videosignal liegt an
3 = Videosignal liegt an und Zeile ist eingerastet

↔ ④ **Einstellen der Bildhelligkeit**

void Set_Brightness(unsigned short nDevNo,short nBright);

nBright: Bildhelligkeit (-128..127)

↔ ④ **Helligkeitseinstellung zurücklesen**

short Get_Brightness(unsigned short nDevNo);

Returnwert: Inhalt des Helligkeitsregisters des Videoprocessors

↔ ④ **Einstellen des Kontrasts**

**void Set_Contrast(unsigned short nDevNo,
 unsigned short nContr);**

nContr: Bildkontrast (0..511)

↔ ④ **Kontrasteinstellung zurücklesen**

unsigned short Get_Contrast(unsigned short nDevNo);

Returnwert: momentan eingestellter Kontrastwert

↔ ④ **Einstellen der Farbsättigung**

**void Set_Saturation(unsigned short nDevNo,
 unsigned short nSat_U,
 unsignedshort nSat_V);**

nSat_U: Sättigung des U-Farbanteils (0..511, Default = 254)

nSat_V: Sättigung des V-Farbanteils (0..511, Default = 180)

⇔ **④ Inhalte der Farbsättigungsregister lesen**

unsigned short Get_Sat_U(unsigned short nDevNo);
unsigned short Get_Sat_V(unsigned short nDevNo);

Returnwert: Wert der aktuellen U- bzw. V-Farbsättigung

⇔ **④ Farbton korrigieren (nur bei NTSC)**

void Set_Hue(unsigned short nDevNo, short nHue);

nHue: Farbton, Phasenlage des Farbträgers (-128..127)

⇔ **④ Inhalt des Farbton-Registers zurücklesen**

short Get_Hue(unsigned short nDevNo);

Returnwert: äquivalent der eingestellten Phasenlage des Farbträgers

⇔ **④ Ein-/Ausschalten des Luma-Tiefpassfilters**

void Set_LDec (unsigned short nDevNo,
unsigned short nOn,
unsigned short nHfilt);

nOn: 1 = Luma Decimation einschalten
 0 = Luma Decimation ausschalten
nHfilt: 0 = Automatische Filterauswahl
 1 = CIF Filter
 2 = QCIF Filter
 3 = ICON Filter

⇔ **④ Ein-/Ausschalten des Testbilds**

void Set_ColorBars (unsigned short nDevNo,
unsigned short nColorBars);

nColorBars: 0 = Testbild ausblenden
 1 = Testbild einschalten

⇔ ④ Kontrolle des Wertebereichs

**void LumaControl (unsigned short nDevNo,
 unsigned short nRange,
 unsigned short nCore);**

nRange: 0 = Luma Spannweite 16 - 253
 1 = Luma Spannweite 0 – 255
nCore: 0 = 0
 1 = 8 alle Helligkeitswerte <= 8 werden als 0 interpretiert
 2 = 16 alle Helligkeitswerte <= 16 werden als 0 interpretiert
 3 = 32 alle Helligkeitswerte <= 32 werden als 0 interpretiert

⇔ ⑤ Daten über den GPIO-Port lesen und ausgeben

Die Ansteuerung des GPIO-Ports (Buchse X6) erfolgt über folgende drei Funktionen:

**void Set_GPIO_Direction(unsigned short nDevNo,
 unsigned short nDirection);**

nDirection: kann Werte zwischen 0 und 4095 annehmen

**void Set_GPIO_Data(unsigned short nDevNo,
 unsigned short nData);**

nData: Daten, die über den Erweiterungsport ausgegeben werden sollen

unsigned short Get_GPIO_Data(unsigned short nDevNo);

Rückgabewert: Daten, die vom Erweiterungsport gelesen werden

⇔ ⑤ ☆ **Daten über die I²C-Schnittstelle übertragen**

Mit diesen Funktionen können Sie Bausteine, die an der I²C-Schnittstelle angeschlossen sind, lesen und beschreiben.

**void I2C_Set_BR_Mode (unsigned short nDevNo,
 unsigned char bMode)**

bMode	Baudrate
	eGrabber-4plus: 0 = 99,2 kHz, 1 = 396,8 kHz
	pciGrabber-4: 0 = 33 kHz, 1 = 290 kHz

**unsigned char I2C_ReadByte (unsigned char nDevNo,
 unsigned char bChipAddress,
 unsigned char bSubAddress,
 unsigned char *bByteRead)**

bChipAddress:	Geräteadresse des I ² C-Geräts auf dem Bus
bSubAddress:	Speicher-Adresse innerhalb des I ² C-Geräts
*bByteRead:	Zeiger auf eine Byte-Variable, in die das Ergebnis geschrieben wird

Returnwert: SUCCESS, NOACK, INVALID_ADDRESS

**unsigned char I2C_WriteByte (unsigned short nDevNo,
 unsigned char bChipAddress,
 unsigned char bSubAddress,
 unsigned char bData)**

bChipAddress:	Geräteadresse des I ² C-Geräts auf dem Bus
bSubAddress:	Speicher-Adresse innerhalb des I ² C-Geräts
bData:	Byte, das in die spezifizierte Adresse geschrieben wird

Returnwert: SUCCESS, NOACK, INVALID_ADDRESS,
WRITE_FAILED

⇔ ⑤ ☆ **internes EEPROM verwenden**

Der eGrabber-4*plus* besitzt einen internen, nichtflüchtigen Speicher, der vom Anwender benutzt werden kann, um beliebige Parameter zu speichern.

Insgesamt stehen in diesem Speicher 256 Bytes zur Verfügung.

Hinweis:

Da das Vorgängermodell eGrabber-2 keinen internen Speicher besitzt, sind die nachfolgenden Funktionen nicht kompatibel.

**unsigned char I2C_ReadEEProm (unsigned short nDevNo,
 unsigned char bSubAddress,
 unsigned char *bByteRead)**

bSubAddress: zu lesende Speicheradresse (0x00 ... 0xFF)

*bByteRead : Zeiger auf eine Byte-Variable, in die das Ergebnis geschrieben wird

Returnwert: Fehlercode = SUCCESS, NOACK

**unsigned char I2C_WriteEEProm (unsigned short nDevNo,
 unsigned char bSubAddress,
 unsigned char bData)**

bSubAddress: Speicheradresse, auf die geschrieben wird (0x00...0xFF)

bData: Datenbyte, das geschrieben wird

Returnwert: Errorcode = SUCCESS, NOACK, WRITE_FAILED

⇔ ⑤ ☆ Kontrollieren des I/O-Pins

Der eGrabber-4plus besitzt einen extern verfügbaren, transistorgetriebenen I/O-Pin, der wahlweise zur Eingabe oder Ausgabe von Steuersignalen dienen kann. Dieser Pin kann mit den folgenden beiden Funktionen kontrolliert werden.

Diese Funktion ist beim eGrabber-2 nicht vorhanden.

void Set_Ext_IO (unsigned short nDevNo, unsigned char bData)

bData: Steuerwert, 0 = CLOSE, 1 = HI-Z

unsigned char Read_Ext_IO (unsigned short nDevNo)

nDevNo = Device Number

Rückgabewert: Zustand des I/O-Pins

⇔ ⑥ Direktes Lesen und Beschreiben von Registern

Mit den folgenden Funktionen lassen sich die Register des Video-Prozessors direkt lesen oder beschreiben:

**short Read_Local_DWord(unsigned short nDevNo,
 unsigned short nRegister_Number,
 unsigned long *lContent);**

nRegister_Number: Nummer des Registers

lContent: Inhalt des Registers

**short Write_Local_DWord(unsigned short nDevNo,
 unsigned short nRegister_Number,
 unsigned long lContent);**

nRegister_Number: Nummer des Registers

lContent: Inhalt des Registers

5.3.4 Programmbeispiel DOS

In diesem Kapitel wird ein einfaches Programmbeispiel unter DOS erarbeitet, das ein Bild im Arbeitsspeicher ablegt.

Der Einfachheit halber beschränken wir uns auf ein festes Bildformat und eine einzige Digitalisierung des Bilds (Snapshot). Unser Bildformat soll 256 x 256 Bildpunkte betragen. Wir gehen davon aus, daß sich nur ein eGrabber-4*plus* im Rechner befindet. Sollten mehrere Grabber vorhanden sein, wird nur der erste Grabber benutzt.

Zuerst wird ein Objekt „Grabber“ erzeugt, das die Schnittstelle zum eGrabber-4*plus* darstellt. Unser Bild soll später in einem Array abgelegt werden, dem wir die Bezeichnung `pEWert` geben. An dieser Stelle ist es erforderlich, die Datenmenge des angeforderten Bilds zu bestimmen. Die Bilddimension beträgt 256 x 256 Pixel, es soll ein RGB-Farbbild abgelegt werden, wofür wir das RGB24-Format benutzen. RGB24 benötigt 3 Byte pro Pixel. Es sind also 256 x 256 x 3 Byte für ein Bild erforderlich.

Die Bedienung des Grabbers beginnt mit der Kontrolle, ob ein Grabber vorhanden ist. `Max_Device_Number()` gibt die Anzahl der im System installierten Grabberkarten zurück. Ist die Anzahl = 0, also kein Grabber im Rechner, so wird das Programm vorzeitig beendet.

Falls ein - oder mehrere - Grabber gefunden wurden, wird der Grabber initialisiert, indem `Initialize(1)` aufgerufen wird. Der Parameter 1 gibt an, daß der erste Grabber initialisiert werden soll. Sind mehrere Grabber vorhanden, so muß `Initialize()` für alle Grabberkarten aufgerufen werden.

Es ist sinnvoll, durch Abfrage von `Get_Error()` nach jedem Aufruf zu prüfen, ob ein Fehler auftrat oder ob die Operation erfolgreich durchgeführt wurde. Um die Übersichtlichkeit im Beispiel zu wahren, wurde dies hier nicht durchgeführt.

Mit **Set_Channel(1,1)** wird für den Grabber Nr. 1 der Eingangskanal 1 ausgewählt. Die Digitalisierung wird für diesen Eingangskanal vorgenommen. Die nachfolgende Wartezeit dient dazu, daß der Grabber sich auf das Kamerasignal synchronisieren kann.

Der nächste Funktionsaufruf ist der aufwendigste: Mit der Routine **Set_Image(1,...)** werden die Bildparameter für Grabber Nr. 1 eingestellt. Wir möchten nur ein Halbbild, nämlich das Even-Bild digitalisieren. Die Parameter für die Odd-Halbbilder sind daher für uns nicht von Interesse und werden alle Null gesetzt (Null bedeutet, daß die Werte im Videoprozessor nicht verändert werden). Wichtig ist, den Zeiger auf den Odd-Speicherbereich auf NULL zu setzen, denn dadurch wird dem Grabber mitgeteilt, daß kein Odd-Bild erzeugt werden soll.

Bei den Even-Bild-Parametern wird als erstes die Position des Bildausschnitts festgelegt. Die Koordinaten der linken oberen Ecke werden als (1,1) definiert, der Ausschnitt befindet sich also ebenfalls links oben im TV-Bild. (Achtung: (0,0) würde die vorher eingestellten Werte unverändert lassen.) Wie in der Aufgabenstellung gefordert, definieren wir als nächstes die Pixelanzahl des Ergebnisbilds auf 256 x 256. Das von der Bildquelle gelieferte Halbbild wird in voller Größe formatgetreu 4:3 digitalisiert. Dazu wird die Bildgröße 344 x 258 eingestellt.

Maximal wäre 360 x 288 möglich, 344 x 258 ist jedoch insofern günstiger, als daß die gesamte Bildhöhe im Ausschnitt sichtbar ist. Da die Bildbreite unterschiedlich ist, bleibt rechts ein Bereich von 88 Pixeln unsichtbar.

Mit RGB24 wird das Farbformat für das Even-Bild eingestellt. Pro Pixel werden drei Byte erzeugt, jeweils eines für den Rot-, Grün- und Blauwert (*vgl. dazu den Befehl Set_Image() und Bild 48*).

Als Zeiger auf den Bildspeicherbereich wird `pEWert` übergeben, der auf den Beginn des deklarierten Bild-Arrays weist.

Zur Berechnung der Videoprocessor-Einstellungen durch den Treiber muß das verwendete Videosystem bekannt sein. In unserer Anwendung wird als Bildquelle eine PAL-Kamera angenommen, daher wird PAL_BDGHI übergeben.

Da wir nur ein Halbbild benutzen, wird *Interlace* auf 0 gesetzt (beide Halbbilder werden in getrennten Speichern abgelegt; in unserem Beispiel entfällt zusätzlich noch das Odd-Bild).

Schließlich wird der Single-Shot-Modus gewählt, da nur ein Bild digitalisiert werden soll und keine kontinuierliche Folge von Bildern.

Nun kann der Digitalisierungsvorgang gestartet werden. Dies geschieht mit dem Aufruf von **Start_Grabber(1)**. In der folgenden while-Schleife wartet das Programm darauf, daß das gewünschte Halbbild fertig digitalisiert im Speicher vorliegt. Dazu wird die Funktion **Data_Present()** für den Grabber 1 aufgerufen. Der Rückgabewert wird mit 0x04 UND-verknüpft, um das dritte Bit (Bit Nr. 2) zu maskieren. Es handelt sich um das *S_EVEN*-Bit (*siehe Bild 49*). Es wird gesetzt, wenn das Even-Bild sich vollständig im Bildspeicher befindet. Da uns nur diese Information interessiert, werden die übrigen Status-Bits weggeblendet. Die Verzögerungszeit in der Schleife verhindert, daß ein ständiges Auslesen der Statusinformation aus dem Grabber die Datenübertragung über den PCI-Bus unnötig behindert.

Ist die Digitalisierung abgeschlossen, muß der Befehl **Stop_Grabber()** aufgerufen werden, damit der Grabber in den Ruhezustand zurückversetzt wird. Andernfalls verharrt er im Wartezustand, und es kann keine weitere Digitalisierung ausgeführt werden.

Das Bild befindet sich nun im definierten Array und kann weiterverarbeitet werden. Um ein neues Bild mit gleichen Dimensionen vom gleichen Kanal anzufordern, kann das Programm ab dem Befehl **Start_Grabber()** wiederholt werden.

```
#include <dos.h>
#include <malloc.h>
#include <stdio.h>
#include <stdlib.h>
#include "pci4grab.h"

PCI_GRABBER4 Grabber; // Objekt zum Zugriff auf den Grabber
static unsigned char pEWert[256*256*3]; // Speicher fuer EVEN-Bild

main ()
{
    if((Grabber.Max_Device_Number()==0)
    {
        printf("Es wurde kein eGrabber-4plus gefunden!!") ;
        exit(1) ;
    }
    Grabber.Initialize(1) ; // Grabber zuerst initialisieren!
    Grabber.Set_Channel(1,1) ; // Eingangskanal wählen und
    delay(100); // warten, bis synchronisiert
    Grabber.Set_Image( 1, //Grabber-Nummer
        0,0, //Alle ODD-Parameter sind...
        0,0, //...unerheblich und werden...
        0,0, //...deshalb...
        0, //...auf 0 gesetzt.
        NULL, //Zeiger auf ODD-Bild=NULL
        //=> kein ODD-Bild
        0, 0 , // linke obere Ecke des EVEN-Bilds
        256,256, // Groesse des Bildausschnitts
        344,258, // Groesse des Bildes
        RGB24, // Farbinformation: RGB24-Bild
        pEWert, // Zeiger auf EVEN-Bildspeicher
        PAL_BDGHI, // Color System angeben
        0, // Interlaced-Modus aus
        1); // SingleShot einschalten

    Grabber.Start_Grabber(1); // Grabvorgang starten
    while(!(Grabber.Data_Present(1)&0x4))
        delay(10); // Warten bis Daten vorhanden sind
    Grabber.Stop_Grabber(1); // Grabvorgang stoppen
    // ** Bild befindet sich im Speicher und kann bearbeitet werden **
}
```

6 Änderungen zum eGrabber-2 und Kompatibilität

Der eGrabber-4*plus* ist als Nachfolger des eGrabber-2 konzipiert. Daher wurde beim Design Wert auf Abwärtskompatibilität gelegt. Dies bedeutet:

- **Windows-Applikationen:**

In Systemen, in denen bisher ein eGrabber-2 installiert war, kann dieser durch einen eGrabber-4*plus* ersetzt werden. Unter Windows sind keine Änderungen an vorhandener Applikationssoftware notwendig.

Es muß jedoch der neue Gerätetreiber für den eGrabber-4*plus* installiert werden.

Der eGrabber-4*plus* stellt dann die gleichen Leistungsmerkmale zur Verfügung wie der eGrabber-2. Die zusätzlichen Funktionen des eGrabber-4*plus* können natürlich nicht zusammen mit alter Applikationssoftware für den eGrabber-2 genutzt werden.

- **DOS-Applikationen:**

Bei DOS-Applikationen wird kein getrennter Gerätetreiber verwendet, sondern der Treiber wird als Bibliothek beim Compilieren in das Anwendungsprogramm eingebunden. Damit der eGrabber-4*plus* mit diesen Programmen zusammenarbeiten kann, muß die neue Treiberbibliothek eingebunden werden. Dazu muß das Programm neu compiliert werden. Es sind aber keine Änderungen am anwenderspezifischen Programmcode erforderlich. Die Software-Schnittstelle der Treiber-Bibliotheken von eGrabber-4*plus* und eGrabber-2 ist also kompatibel.

- **alter Grabber und neue Applikation**

Gegebenenfalls möchte man auf einem System, in dem bereits ein Grabber vom Typ eGrabber-2 (EPC-032 etc.) installiert ist, eine Software installieren, die für den eGrabber-4plus entwickelt wurde und den neuen Treiber verwendet.

Dies ist grundsätzlich möglich, der neue Treiber unterstützt weiterhin die Karten älteren Typs. Allerdings ist dann zu beachten, daß manche Eigenschaften der Hardware nicht zur Verfügung stehen.

Werden Funktionen oder Parameter verwendet, die der alte Grabber nicht unterstützt, so liefert der Treiber einen entsprechenden Fehlercode an das Anwenderprogramm zurück.

Mittels den Funktionen *Read_GrabberInfo* und *Read_OrderCode* kann vorab bestimmt werden, welcher Grabber im System vorhanden ist und welche Eigenschaften zur Verfügung stehen.

- **Hardware-Kompatibilität**

Alle Signale des eGrabber-2 sind beim eGrabber-4plus auf den gleichen Buchsen bzw. Kontakten vorhanden. Es können daher Anschlußkabel, die beim eGrabber-2 verwendet wurden, auch beim eGrabber-4plus eingesetzt werden.

An der **Stiftleiste X3** sind beim eGrabber-4plus einige neue Signale hinzugekommen. Die Stiftleiste besitzt daher mehr 6 Pins als die des eGrabber-2. Möchten Sie den eGrabber-4plus in einer bestehenden Applikation als Ersatz für einen eGrabber-2 verwenden, so können Sie einfach die letzten sechs Pins der Stiftleiste X3 mit einem Seitenschneider abtrennen. Die vorderen 20 Pins sind identisch zu denen des eGrabber-2 und Sie können Ihren vorhandenen Steckverbinder aufstecken.

Um Ihnen das Entfernen der Pins zu erleichtern, sind die Pins, die zum eGrabber-2 kompatibel sind, mit weißen Linien abgegrenzt.

Im Vergleich zum eGrabber-2 sind die **SMB-Buchsen** P3 und P4 entfallen. An diesen Buchsen konnten Sie ein S-Video-Signal einspeisen. Diese Funktion übernimmt nun die normgerechte Mini-DIN-Buchse X4.

Am Erweiterungsanschluß (**Option Port, X6**) haben sich zwei Pins verändert (siehe Tabelle 14).

Erweiterungsanschluß (Option Port, X6)					
Pin	Funktion	Pin	Funktion	Pin	Funktion
1	+5 V out	8	I/O 6	15	I/O-Pin
2	I/O0	9	I/O 7	16	I/O Clk
3	I/O1	10	I/O 8	17	I ² C SCL
4	I/O2	11	I/O 9	18	I ² C SDA
5	I/O3	12	I/O 10	19	GND
6	I/O4	13	I/O 11	20	GND
7	I/O5	14	N.C.	fett = neue Funktion	

Tabelle 14: Belegung des Option-Port – Verbinders (alle Modelle)

- **Spannungsversorgung von Kameras**

Um angeschlossene Kameras mit Betriebsspannung zu versorgen, kann an der Pfostenleiste eine Gleichspannung entnommen werden. Sie wird von der Buchse X300 durchgeschleift oder je nach Jumper JP301 über den PC/104plus Stecker oder der integrierten Spannungsversorgung entnommen.

Beim eGrabber-2 war diese Funktion nicht vorhanden.

- **Umstieg von älteren Treiberversionen (vor V3.0)**

Sollten Sie Software für den eGrabber-4 entwickelt haben, die noch auf einer älteren Treiberversion aufsetzt (vor Version 3.0), so beachten Sie bitte folgende Hinweise:

- Die Windows-DLL wurde um die Funktion **GetVersionNumber** erweitert. Bitte beachten Sie: Erhalten Sie mit der Funktion **GetProcAddress** keinen gültigen Funktionszeiger, benötigen Sie eine neuere Version der **GR4CDLL.DLL**.
- Funktion **Set_Image**:

In früheren Versionen mußten in der Windows-DLL die Parameter *hpos* und *vpos* auf 1 gesetzt werden, um den Bildausschnitt in die linke obere Ecke des digitalisierten Bildes zu legen. Diese Parameter akzeptieren nun nur noch gerade Werte, d.h. *hpos* und *vpos* müssen auf 0 gesetzt werden.

7 Störungssuche

Falls Ihr Grabber nicht wie erwartet funktioniert, können Sie kleinere Störungen selbst anhand der folgenden Liste beheben. Bitte prüfen Sie zunächst, ob Sie die Störung selbst beheben können, bevor Sie den PHYTEC-Support konsultieren.

- Die Farbdarstellung im Windows-Demo-Programm ist stark reduziert.
 - Prüfen Sie die Einstellung Ihrer Grafikkarte. Um die volle Farbtiefe des eGrabber-4*plus* darstellen zu können, muß die Grafikkarte auf mindestens 16 Mio. Farben konfiguriert sein.
- Es ist nur ein blaues Bild zu sehen.
 - An dem gewählten Eingang ist keine Videoquelle angeschlossen. In der Regel wird dann ein blaues Bild geliefert. Prüfen Sie, ob der richtige Eingang eingestellt ist.
- Das digitalisierte Bild hat Schlieren / Streifen
 - Es kann sich um Moiré-Störungen bedingt durch den Farbträger handeln. Prüfen Sie, ob die Farbfalle eingeschaltet ist.
 - Die Signalleitung von der Kamera ist defekt (Abschirmung prüfen).
 - Eine Masseschleife durch eine zusätzliche Masseverbindung zwischen PC und Kamera ist Ursache von Brummstörungen. Prüfen Sie, ob z.B. durch Erdung oder Spannungsversorgung eine Brummschleife aufgebaut wurde.
- Am S-Videoeingang wird kein Bild digitalisiert.
 - Ist der *S-Video*-Eingang korrekt verbunden?
 - Wurden versehentlich beide *S-Video*-Eingänge beschaltet?
 - Wurde der eGrabber-4*plus* richtig auf *S-Video*-Betrieb konfiguriert?
 - Wurde die falsche Eingangsbuchse gewählt (Mini-DIN / Combi)?
 - Wurde nach Umschaltung auf *S-Video* mit Set_Channel ein anderer Kanal gewählt?

- Bei *S-Video*-Betrieb ist das Bild nur schwarz/weiß
 - Wird das Chroma-Signal korrekt zugeführt?
 - Ist der Grabber auf *S-Video*-Betrieb umgestellt?

- Beim *S-Video*-Betrieb ist das Bild schwarz/weiß mit starken Farbstörungen.
 - Ist das Chroma-Signal korrekt angeschlossen?
 - Handelt es sich um eine *S-Video*-Quelle?
 - Ist keine Bildquelle mehr angeschlossen?

- Das Bild wird unvollständig / sehr langsam angezeigt
 - Erhöhen Sie die Wartezeit zwischen den Statusabfragen. Der PCI-Bus wird durch zu häufige Statusabfragen blockiert.
 - Wurde die zeitliche Auflösungsreduzierung benutzt?

- Das Bild ist kontrastarm / zu hell / zu dunkel
 - Prüfen Sie die Einstellungen von Helligkeit, Kontrast usw.
 - Schalten Sie ggfs. die AGC ein

- Das Bild springt oder zeigt Halbbildversatz
 - Es wurde keine genügend große Wartezeit zwischen dem Umschalten der Eingangskanäle eingehalten.

- Das Bild erscheint ohne Farbe / im falschen Format
 - Ist das richtige Farbsystem eingestellt?
 - Wurde der Grabber korrekt initialisiert?
 - Wurde eine ausreichende Wartezeit bei der Kanalumschaltung eingehalten?

- Der Rechner stürzt unter DOS beim Programmstart ab / eine Fehlermeldung erscheint
 - Möglicherweise stört ein anderer Gerätetreiber oder Speicherverwalter, z.B. EMM386. Deinstallieren Sie diese Komponente probeweise. Verwenden Sie ggfs. einen anderen Treiber.

- Am unteren Bildrand fehlt ein Stück / es wird kein Bild digitalisiert
 - Die vertikale Bildgröße wurde zu groß gewählt.
 - Es wurde nicht beachtet, daß ab einer Zeilenzahl von mehr als 288 (PAL) bzw. 262 (NTSC) mit Vollbildern gearbeitet werden muß.

- Die Farben werden nicht korrekt dargestellt.
 - Die Werte *hpos* und *hsize* müssen gerade sein, damit die Farbückgewinnung korrekt arbeitet.
 - In eigenen Programmen: Sie haben Cr/Cb vertauscht.
 - Haben Sie das richtige Farbformat gewählt?
 - Die Farbsättigungsregister wurden verstellt. Beachten Sie, daß für eine korrekte Farbtondarstellung die U- und V-Farbsättigung im *Prozentwert* identisch sein müssen. Die *Registerwerte* sind dabei jedoch unterschiedlich!

- Beim kontinuierlichen Grabben springt das Bild um eine Zeile nach oben / unten.
 - Es wird nicht die Parität des Halbbilds beachtet. Achten Sie darauf, stets das gleiche Halbbild anzufordern. Speicherbereich von Even- und Odd-Bild dürfen nicht gleich sein.

Hinweis:

Der Versatz beträgt genaugenommen eine halbe Zeile, weswegen man nicht durch Versatz um eine Zeile das Fehlerbild kompensieren kann.

- Die Videoquelle liefert kein sauberes Signal
- Die Kanalschaltung zwischen zwei Kameras erfolgt zu schnell

- Bei der Darstellung von Vollbildern sind diagonale Linien / Kreise auffällig gerastert oder haben gezackte Linien.
 - Gerades und ungerades Halbbild werden vertauscht dargestellt (nur möglich, wenn Sie einzelne Halbbilder benutzen).

- Nach der Digitalisierung eines Bildes ist keine weitere Digitalisierung mehr möglich.
 - Sie müssen vor der Anforderung eines neuen Digitalisierungsvorgangs den vorherigen durch **Stop_Grabber()** abschließen.

- Im DOS-Demoprogramm kann nur ein Schwarz/Weiß-Bild dargestellt werden (die ersten beiden Menüpunkte fehlen).
 - Das DOS-Programm benötigt zur Darstellung von Farbbildern und Bildern mit höherer Auflösung eine Grafikkarte, die mit dem Grafikprozessor Typ ET4000 / ET6000 (Tseng Labs) ausgestattet ist. Es handelt sich nur um ein Programmierbeispiel unter DOS, in dem nicht die Besonderheiten aller Grafikkarten berücksichtigt werden konnten. Zur Darstellung von Echtfarben / einer Auflösung von mehr als 640 x 480 Bildpunkten kann der Standard-VGA-Modus nicht mehr verwendet werden.

- Im Vollbild-Modus entsteht bei kontinuierlicher Darstellung ein Nachzieheffekt, obwohl doch der Grabber zwei zueinander gehörende Halbbilder in Echtzeit digitalisiert.
 - Der Effekt ist in der endlichen Darstellungsgeschwindigkeit des Rechners begründet. Die Aktualisierung des Bildschirms wird nicht genügend schnell vorgenommen, obwohl die Bilder in Echtzeit im Arbeitsspeicher vorliegen.

- Der Versorgungsspannungsausgang für die Kamera funktioniert nicht.
 - Überprüfen Sie die Sicherung.
 - Ist die Spannungsversorgung richtig konfiguriert (vgl. Kapitel 1.9)?
 - Bei Modellen -X2: leuchten die grünen LEDs? Ist Jumper JP301 richtig gesteckt. Bei Überlast wirkt die interne Strombegrenzung und die Ausgangsspannung wird reduziert

- Die integrierte Spannungsversorgung funktioniert nicht und die grünen LEDs leuchten nicht.
 - Prüfen Sie, ob es sich um ein Modell EPC-032-X2 / EPC032-X1-X2 handelt. Das Netzteil ist nur bei diesen Modellen vorhanden.
 - Liegt die Eingangsspannung an (8 bis 25 V / 15 bis 25 VDC)?
 - Ist die Eingangsspannung richtig gepolt?
 - Ist der Stecker an X300 richtig beschaltet?
 - Ist die Sicherung SI301 in Ordnung?
 - besteht ein Kurzschluß / eine Überlast (Netzteil schaltet ab)?
 - Wurde das Netzteil über den Anschluß X301 abgeschaltet? Zum Betrieb muß X301 offen sein bzw. Pin 1 auf HIGH-Pegel liegen. Ziehen Sie evtl. probeweise einen Stecker auf X301 ab.

- Die grüne LED für +12 V leuchtet nicht und die 12 V-Ausgangsspannung ist nicht vorhanden. Die +5 V-Versorgung funktioniert.
 - Die Eingangsspannung liegt unter +15 VDC. Zum Betrieb des 12 V-Zweigs des Netzteils sind mindestens 15 V Eingangsspannung erforderlich.
 - Im 12-V-Zweig besteht ein Kurzschluß oder es wird zuviel Strom entnommen.

- Die integrierte Spannungsversorgung funktioniert nicht obwohl die grünen LEDs leuchten.
 - Prüfen Sie mit einem Multimeter die Spannungen an X302.
 - Die Spannungsversorgung ist nicht mit Grabber oder CPU-Karte verbunden. Diese Verbindung müssen Sie selbst herstellen. Üblicherweise geschieht dies über ein Kabel an X302. Lesen Sie in der Beschreibung Ihrer CPU-Karte nach, wo dort die Versorgungsspannungen am besten eingespeist werden.
 - Wenn die LEDs nur schwach leuchten, kann es sein, daß dem Netzteil zu viel Strom entnommen wird. Die Überlastsicherung reduziert dann die Ausgangsspannung, um eine Zerstörung des Netzteils zu verhindern.

- Wenn Sie die Spannung über die PC/104-Leisten in das System einspeisen möchten, müssen Sie das Jumperfeld JP301 entsprechend konfigurieren (*siehe Kapitel 1.10.3*).
 - Wenn Sie Jumptec MOPS-CPU-Karten verwenden, müssen Sie ggf. den Steckverbinder X302 bestücken, um die CPU-Karte kabellos mit Spannung zu versorgen
- Der Steckverbinder X302 für die CPU-Spannungsversorgung ist nicht vorhanden.
- Der Steckverbinder ist standardmäßig nicht bestückt, da dieser nur für die Karten bestimmter Hersteller geeignet ist (z.B. Jumptec, MOPS-Serie). Damit es nicht zu Konflikten mit anderen Komponenten oder Steckern von nicht-kompatiblen Karten kommt, wird der Stecker von Phytec nicht aufgelötet.
- Beim Einschalten läuft das System nicht richtig hoch oder die grünen Kontroll-LEDs flackern.
- Die angeschlossenen Komponenten haben einen zu hohen Einschaltstrom (z.B. Festplatten, CD-ROM-Drives). Dadurch spricht die Überlast-Sicherung des Netzteils an. Wir empfehlen die Verwendung von Komponenten, die für Laptops entwickelt wurden (z.B. 2,5“-Festplatten). Notfalls kann versucht werden, den Einschaltstrom durch einen Widerstand in der Zuleitung des Geräts zu reduzieren (0,5 bis 2 Ω).
- Das dargestellte Bild ist durch horizontale Streifen unterbrochen, in denen man möglicherweise den Inhalt vorhergehender Bilder sieht. Bei bewegten Objekten ergeben sich horizontale Nachzieheffekte.
- Der Grabber kann die Bilddaten nicht in Echtzeit über den PCI-Bus übertragen, da andere Karten den Bus zu stark beanspruchen oder die Bus-Einstellungen des BIOS nicht korrekt sind. Überprüfen Sie die Einstellungen der anderen PCI-Karten und die PCI-Einstellungen des BIOS.

- Wie können mehrere eGrabber-4*plus* gleichzeitig unter Windows'95 betrieben werden?
 - Unter Windows'95 benutzen alle eGrabber-4*plus* den gleichen Bildspeicher im Arbeitsspeicherbereich des PCs. Es kann daher nur ein Digitalisierungsvorgang zu einem Zeitpunkt durchgeführt werden. Während ein Grabber digitalisiert, können andere Grabber jedoch angesprochen werden und ihre Parameter ausgelesen oder verändert werden. Dies ist vorteilhaft, wenn man z.B. die Kanaleinstellungen ändern möchte und dabei die Vorlaufzeit, bis das analoge Signal vom Grabber richtig erkannt wird, verkürzen möchte. Unter DOS kann der Programmierer für jeden Grabber einen eigenen Speicherbereich definieren. Damit ist auch parallele Digitalisierung möglich. Es muß jedoch beachtet werden, daß die Datenübertragungsrate des PCI-Busses nicht überschritten wird. Unter DOS oder Windows 98/2000/NT/ME/XP kann jeder Grabber einen eigenen Bildspeicher benutzen.

Index

A

Adressbereich	10
AGC	
(Treiber)	114
analoge Fernsehtechnik	56
Anschlußbelegung	11
Anschlüsse	7
Arithmetics	66
Arithmetische Operationen an	
Standbildern	66
Auflösung	119
Ausgangsspannungen	24
Ausschnitt	120
AUTO-Funktion	
Treiber	110

B

Basisadresse	10
Bewegung	78
Bild	
Auflösung	119
Ausschnitt	120
blau (Störung)	183
Digitalisierungsmodus	129
falsches Format (Störung)	184
Farbformat	125
Farbsättigung (Treiber)	142
Farbstörungen	184
Größe	119
Halbbildversatz (Störung)	184
horizontale Streifen	
(Störung)	188
kein (Störung)	183
kleine Formate (Treiber)	144
kontrastarm (Störung)	184
Parameter (Treiber)	120
Raster/Zacken (Störung)	186
schwarz/weiß (Störung)	184

springen (Störung)	185
unvollständig / fehlt	
(Störung)	185
unvollständig / langsam	
(Störung)	184
Bildauflösung	5
Bilddaten	
Übertragung in Speicher	81
Bilder	
auslassen	116
Bilder abspeichern	
(Demoprogramm)	70
Bildgröße	118
Bildhelligkeit	140
Bildhelligkeitssignal	79
Bildrauschen	
verringern	
(Demoprogramm)	68
Bildspeicher	128
Auslesen unter Windows	135
DOS	164
Blockschaltbild	74
BNC-Stecker	47
BtYUV	126

C

Chroma-Signal	79
CIF	144
Color Bars	66
Color Meter	65
Color Mode	52
Composite-Eingänge	12, 45
Compositequelle	
anschließen	47
Composite-Quellen	55
Cross-Color-Effekt	112

D

Datenformat	125
Datenformate.....	5
Delphi	
Programmieren unter	97
Demoprogramm	
automatisch starten	61
Bedienung	49
Channel selection.....	54
Grundeinstellung.....	62
Image Resolution	56
Image Selection.....	56
Image Settings.....	54
Installation	43
Luma-Low-Pass	59
Normierung	67
Device-ID.....	30
Digitalisierung	
Grundlagen.....	85
nur eine (Störung)	186
Zeitpunkt	131
Digitalisierungsvorgang	
starten.....	130
DLL.....	87
Anwendung	95
DMA-Kanäle.....	81
DOS	
Programmierbeispiel.....	175
DOS/4GW.....	154
DOS-Treiber	154
DOS-Treibersoftware.....	155

E

EEPROM	150
Eingangsspannung	25
EMM386.EXE	154
Entwicklungsumgebungen.....	87
Ersatzsicherung	4
Erweiterungsport.....	19
programmieren.....	146
even (Halbbild)	76

F

Fadenkreuz	
einblenden	62
Farbauszüge	79
Farbbalken.....	66
Farbdarstellung	
reduziert (Störung)	183
Farbdifferenzsignal	79
Farbfalle	112
Farbsättigung.....	60, 142
Farbstörungen.....	184
Farbton	60
Korrektur (NTSC).....	143
Farbtöter	115
Farbwerte	
messen.....	65
Field Aligned.....	113
Flimmereffekt.....	76
Frame-Rate	53
Funktionen	
GetVersionNumber	
(Win).....	103
Data_Present().....	133
FreeLibrary(..).....	95
Get_Brightness()	141
Get_CaptureCounter().....	140
Get_Contrast().....	141
Get_Error().....	102
Get_GPIO_Data()	146
Get_Hue().....	143
Get_Sat_U()	142
Get_Sat_V()	142
Get_Signal_Status()	139
Get_Video_Status().....	108
GetPictureBufferAddress()	
(Win).....	135
GetProcAddress(..).....	95
I2C_ReadByte	148
I2C_ReadEEProm()	150
I2C_Set_BR_Mode()	148
I2C_WriteByte()	149
I2C_WriteEEProm()	150

-
- Initialize() 105
 Klassifizierung 99
 LoadLibrary(...) 95
 LumaControl() 145
 Max_Device_Number() 104
 Read_Ext_IO() 152
 Read_GrabberInfo() 105
 Read_Local_DWord() 152
 Read_OrderCode() 106
 Reset_CaptureCounter 140
 Reset_Interrupt() 138
 Set_AGC() 114
 Set_Brightness() 140
 Set_BW() 112
 Set_ChannelEx() 111
 Set_CKill() 115
 Set_Color_System() 107
 Set_ColorBars() 144
 Set_Composite() 108
 Set_Contrast() 141
 Set_Ext_IO() 151
 Set_GPIO_Data() 146
 Set_GPIO_Direction 146
 Set_Hue() 143
 Set_Image() (DOS) 162
 Set_Image() (Win) 117
 Set_Interlace() 113
 Set_Interrupt() 137
 Set_LDec() 144
 Set_S_VideoEx() 109
 Set_Saturation() 142
 Start_Grabber() 130
 Stop_Grabber() 132
 TemporalDect() 116
 Write_Local_DWord() 152
- G**
- Gerätetreiber
 Win 98/ME 40
 Win NT 4.0 41
 Win95 39
 Gr4CDLL.DLL 96
- Grabber
 mehrere ~ unter Win'95 189
 Grabberkarte
 Einbau 33
- H**
- Halbbild 77, 113
 Halbbild-Digitalisierung 122
 Halbbild-Digitalisierung mit
 Zoom 123
 Halbbilder 56
 Halbbildversatz 184
 Helligkeit 60, 140
 Histogramm 64
- I**
- I/O-Pin 16, 151
 I/O-Port
 testen (Demoprogramm) 69
 I²C-Schnittstelle 6, 18
 programmieren 148
 ICON 144
 IMAQ 42
 Interlace
 (Treiber) 113
 Interrupt 10, 137
 IRQ 138
- J**
- JP 201 30
 JP 301 28
- K**
- Kammeffekt 58
 Kompatibilität zum
 pciGrabber-4 179
 Kontrast 60, 141
- L**
- LabView-Treiber 42
 Lieferumfang 3
 Live Image 61
 Live-Bild 49
-

M

Master-Slot..... 34
mehrere pciGrabber 100
Moiré-Störungen 183

N

Nachzieheffekt 186
Non-Interlace
(Treiber) 113

O

odd
(Halbbild)..... 76
Open Image on Start 61

P

Parameterspeicher 149
PCI4GRAB.LIB..... 156
pciGrabber-4
Kompatibilität 179
PCI-Slots 34
PCI-Steckplatz 33
Pfostenstecker 19
PHYTEC Vision Tools Drivers
and Demos CD
Treiberinstallation 39
Portpin..... 146, 151
Ports 6
Protected-Mode..... 154

Q

QCIF 144

R

R,G,B 79
Rechnerabsturz (Störung) 184
Register 153
Registry-Eintrag
Windows'95 91
RGB 125
RGB15 126

RGB16..... 125
RGB24..... 125
RGB3 125

S

s/w-Signal
(Treiber) 112
Schlieren / Streifen (Störung) . 183
Schnappschüsse..... 61
Sicherheitshinweise..... 8
Signalqualität 139
Single Image 61
Slot 30
Slot-Nummer..... 30
Snapshot 61
Softwaretreiber..... 73
Spannungsversorgung 7, 24, 36
Spannungsversorgung von
Kameras
Kompatibilität 181
Spannungsversorgungsteil 36
Speicher
Windows'95 88
S-Video 15
S-Video - Quellen 55
Synchronisation..... 5

T

Testbild..... 144
Treiber
AGC Ein/Ausschalten..... 114
Anzahl der digitalisierten
Bilder verfolgen 140
Anzahl der Grabber
bestimmen 104
Auslassen von Fields, Frames
..... 116
ältere Versionen 181
Bildaufnahme starten 130
Bildaufnahme stoppen 132
Bilddaten auslesen (Win)..... 135
Bildgröße und Skalierung
einstellen (DOS)..... 162

-
- Bildgröße und Skalierung
 - einstellen (Win/DOS)..... 117
 - Bildhelligkeit einstellen 140
 - Composite Eing. einstellen .. 108
 - Digitalisierungsfortschritt
 - erkennen 133
 - Ein-/Ausschalten der
 - Farbfalle 112
 - Eingangskanals einstellen 111
 - Erkennung des
 - Videoformats..... 108
 - Farbsättigung einstellen 142
 - Farbsättigung lesen 142
 - Farbsystem einstellen 107
 - Farbton korrigieren (NTSC). 143
 - Farbton-Register lesen 143
 - Farbtöter Ein-/Ausschalten .. 115
 - Fehlermeldungen auswerten 102
 - GPIO-Port programmieren... 146
 - Grabber und Treiber
 - initialisieren..... 105
 - Grabbername als
 - Klartextstring lesen 106
 - Helligkeitseinstellung
 - zurücklesen..... 141
 - I/O-Pin programmieren 151
 - I²C-Schnittstelle verwenden. 148
 - Informationen über
 - Grabber abfragen 105
 - installieren 38
 - Interlaced-Modus
 - ein/ausschalten 113, 118
 - internes EEPROM
 - verwenden 149
 - Interrupt konfigurieren..... 137
 - Interrupt-Flag zurücksetzen . 138
 - Kontrast einstellen..... 141
 - Kontrastwert auslesen 141
 - Lesen / Schreiben von
 - Registern 152
 - Luma Werte kontrollieren 145
 - Luma-Tiefpass
 - ein/ausschalten 144
 - S-Video-Modus einstellen.... 109
 - Testbild ein/ausschalten 144
 - Versionsnummer der DLL
 - bestimmen (Win)..... 103
 - Videosignals erkennen 139
 - Windows'98 / NT 94
 - Treiber-Bibliothek..... 73
 - TV-Bild 76
 - Twain-Treiber..... 42
 - Type Casting Settings..... 67
- U**
- Umrechnung
 - RGB / YUV 79
- V**
- Vermessungsanwendungen 119
 - Versorgung 5
 - Versorgungsspannungs-
 - ausgang 27
 - Störung 186, 187, 188
 - Videoquellen
 - anschließen 46
 - Videoquellen anschließen 45
 - Videosignal..... 76
 - Vollbild..... 57, 129
 - Vollbilddigitalisierung 58
 - Vollbild-Digitalisierung 123
 - Grundlagen 85
 - Vollbildmodus 129
 - VxD-Treiber 88
- X**
- X6 (Optionport)..... 19
-

Y	YCrCb	126
	YUY2	126
Y (Helligkeitsinformation)		128
Y,Cr,Cb	Z	
Y8 (Grauwert-Format)	Zubehör	4

Dokument: eGrabber-4plus
Dokumentnummer: L-608d_4, Juni 2004

Wie würden Sie dieses Handbuch verbessern?

Haben Sie in diesem Handbuch Fehler entdeckt?

Seite

Eingesandt von:

Kundennummer: _____

Name: _____

Firma: _____

Adresse: _____

Einsenden an:

PHYTEC Technologie Holding AG
Postfach 100403
D-55135 Mainz, Germany
Fax : +49 (6131) 9221-33

Published by

PHYTEC

© PHYTEC Meßtechnik GmbH 2004

Ordering No. L-608d_4
Printed in Germany