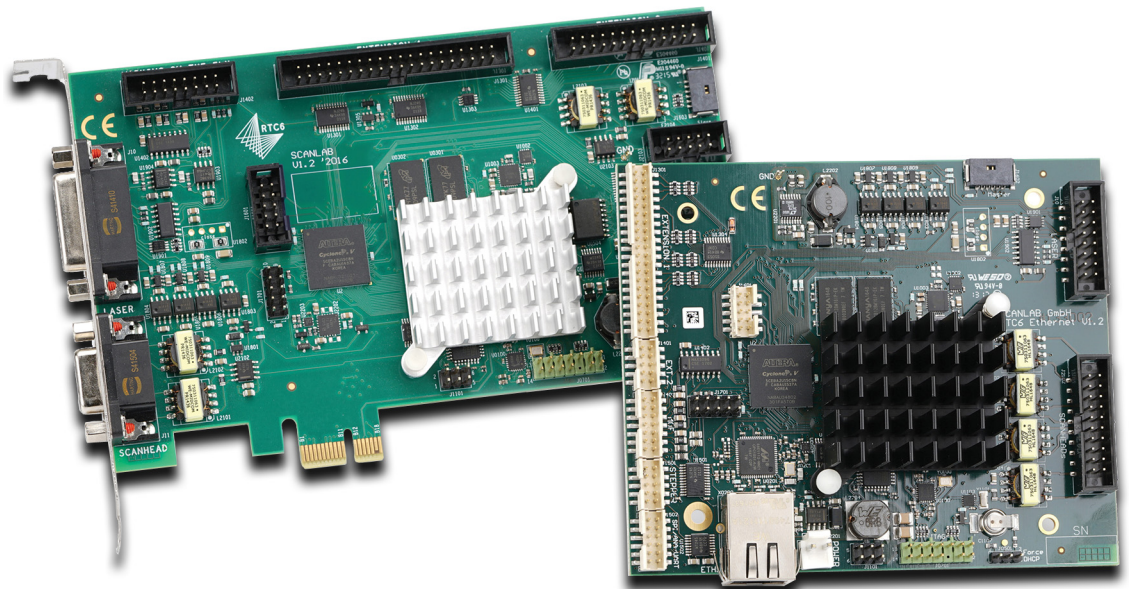


## Installation und Inbetriebnahme

### RTC6 PCI-Express-Karte und RTC6 Ethernet-Karte

für die Scan-Kopf- und Lasersteuerung in Echtzeit  
RTC6-Software-Paket V1.4.1



SCANLAB GmbH  
Siemensstr. 2a  
82178 Puchheim  
Deutschland

Tel. +49 (89) 800 746-0  
Fax: +49 (89) 800 746-199

[info@scanlab.de](mailto:info@scanlab.de)  
[www.scanlab.de](http://www.scanlab.de)

© SCANLAB GmbH 2018  
(Doc. Rev. 1.1.3 d - 23. August 2018)

SCANLAB GmbH behält sich vor, dieses Dokument jederzeit und ohne Ankündigung inhaltlich zu aktualisieren.  
Kein Teil dieses Dokuments darf in irgendeiner Form (Fotokopie, Druck, Mikrofilm oder in einem anderen Verfahren) ohne ausdrückliche schriftliche Genehmigung der SCANLAB GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

RTC ist ein eingetragenes Warenzeichen der SCANLAB GmbH.  
Andere erwähnte Marken unterliegen dem Markenschutz der jeweiligen Markeninhaber.

## Inhalt

<b>1 Die RTC6 PCI-Express-Karte .....</b>	<b>7</b>
1.1 Über dieses Handbuch .....	7
1.2 Weitere Dokumentationen .....	7
1.3 Produktübersicht .....	8
1.3.1 Bestimmungsgemäße Verwendung .....	8
1.3.2 Systemanforderungen .....	8
Hardware .....	8
Software .....	8
1.3.3 Optionale Funktionalitäten .....	9
1.3.4 Kennzeichnung .....	9
1.3.5 Typenbezeichnung .....	9
1.3.6 Auspackanleitung und typischer Lieferumfang .....	9
1.3.7 Mitgelieferte Software .....	9
1.3.8 Zubehör für die RTC6 PCI-Express-Karte .....	10
1.3.9 Ergänzende Software .....	10
1.4 Layout und Schnittstellen .....	11
1.4.1 Layout – Oberseite .....	11
1.4.2 Layout – Unterseite .....	12
1.4.3 Schnittstelle zum PC .....	12
1.4.4 Master/Slave-Taktsynchronisierung: Master-Stiftleiste, Slave-Stiftleiste .....	13
1.4.5 McBSP/ANALOG-Stiftleiste .....	14
Funktion als McBSP-Schnittstelle (Werksvoreinstellung) .....	14
Funktion als SPI-Schnittstelle (Benutzereinstellung) .....	14
Analog-Eingänge .....	14
1.5 Installation und Inbetriebnahme .....	15
1.6 Über die RTC6-Karte und Hinweise an RTC5-Kartenbenutzer .....	16
1.6.1 RTC6-Karten und RTC5-Karten im Vergleich – Neuerungen und Änderungen .....	16
Synchronisierung von RTC6-Zyklus und externem Taktsignal .....	18
1.6.2 RTC6-Karten und RTC5-Karten im Vergleich – was bleibt unverändert .....	20
1.6.3 Einschränkungen bei RTC6-Karten .....	21
Verfügbarkeit technischer Varianten .....	21
Schnittstellen zur Peripherie .....	21
Funktionalität .....	21
1.7 Quellcode für RTC6-Anwenderprogramme .....	21
1.7.1 RTC6-Quellcode neu erstellen .....	21
1.7.2 RTC5-Quellcode für die RTC6 anpassen .....	21
Zwingend erforderliche Schritte .....	21
Optionale Schritte und Hinweise zu weiteren Anpassungen und Optimierungen .....	22
1.8 RTC6-Befehlssatz .....	23
1.8.1 Neu im RTC6-Befehlssatz .....	23
Nur für Scan-Köpfe mit SCANahead-Technologie .....	23
Allgemein nutzbare RTC6-Befehle .....	23
free_rtc6_dll .....	24
get_bios_version .....	24
init_rtc6_dll .....	25
master_slave_config .....	27
rtc6_count_cards .....	28
set_laser_power .....	29

set_rtc6_mode .....	30
set_timelag_compensation .....	31
spot_distance .....	32
spot_distance_ctrl .....	32
1.8.2 Geändert im RTC6-Befehlssatz .....	33
config_list .....	33
get_dll_version .....	33
get_hex_version .....	33
get_rtc_version .....	34
get_status .....	34
get_sync_status .....	35
load_correction_file .....	37
load_program_file .....	37
load_stretch_table .....	38
mcbasp_init_spi .....	38
number_of_correction_tables .....	39
select_cor_table .....	39
set_auto_laser_control .....	40
set_auto_laser_params .....	45
set_auto_laser_params_list .....	45
set_dsp_mode .....	46
set_pixel_line .....	47
set_trigger4 .....	50
1.8.3 Nicht mehr vorhanden im RTC6-Befehlssatz .....	51
1.9 Master/Slave-Betrieb .....	52
Initialisierung .....	52
Synchronisierung der Takt-Phasen .....	52
Synchrone Starts und Stopps .....	53
Beispiel-Code .....	54
1.10 Ergänzende Informationen zu den RTC6-Pixel-Modi .....	55
1.11 Hinweise zu set_auto_laser_control mit Ctrl = 7 ("Spot Distance Control") .....	60
1.12 Technische Spezifikationen der RTC6 PCI-Express-Karte .....	62
1.13 Konformität mit der EU-Richtlinie zur elektromagnetischen Verträglichkeit (EMV) .....	63
1.14 Konformität mit FCC-Bestimmungen .....	63

<b>2</b>	<b>Anhang A: Die RTC6 Ethernet-Karte</b>	<b>64</b>
2.1	Produktübersicht	64
2.1.1	RTC6 Ethernet-Karte und RTC6 PCI-Express-Karte – Verwendung und Vergleich	64
2.1.2	Systemanforderungen	65
	Hardware	65
	Software	65
2.1.3	Optionale Funktionalitäten	65
2.1.4	Kennzeichnung	65
2.1.5	Typenbezeichnung	65
2.1.6	Auspackanleitung und typischer Lieferumfang	66
2.1.7	Mitgelieferte Software	66
2.1.8	Zubehör für die RTC6 Ethernet-Karte	66
2.1.9	Ergänzende Software	66
2.2	Layout, Schnittstellen, Jumpereinstellungen	67
2.2.1	Layout – Oberseite	67
2.2.2	Layout – Unterseite	68
2.2.3	Abmessungen und Steckverbinderpositionen	69
2.2.4	LASER-Stiftleiste	70
	Laserausgangssignale	70
	Externe Steuersignale	70
	BUSY-Status	70
	2 Bit-Digital-Eingang und 2 Bit-Digital-Ausgang	70
	12-Bit Analog-Ausgänge	71
	Ein-/Ausgangsbeschaltung	71
2.2.5	SCANHEADs-Stiftleiste	73
2.2.6	POWER-Stiftleiste	75
2.2.7	ETH-Buchse	75
2.2.8	SPI/ANA/UART-Stiftleiste	76
	Analog-Eingänge	76
	Verwendung der McBSP-Schnittstelle	76
	Verwendung der RS-232 Schnittstelle	76
2.2.9	STEPPER-Stiftleiste	77
2.2.10	MOF-Stiftleiste	77
	Encoder-Eingänge	78
	Externe Steuersignale	78
	BUSY OUT-Status	78
2.2.11	EXTENSION 1-Stiftleiste	78
	Einstellen des Ausgangssignalpegels	79
	16-Bit-Digital-Ausgang und 16-Bit-Digital-Eingang	79
	Synchronisation der Datenübergabe	79
	BUSY-Status	79
2.2.12	EXT. 2-Stiftleiste	79
	Einstellungen mittels Lötjumper	80
	8-Bit-Digital-Ausgang	80
2.2.13	Master-Stiftleiste, Slave-Stiftleiste	81
2.2.14	Jumpereinstellungen	82
	Lötjumperfeld A – Einstellen des Ausgangssignalpegels an der EXTENSION 1-Stiftleiste	82
	Lötjumperfeld B – Konfiguration von Pin (09) der EXT. 2-Stiftleiste	83
	Lötjumperfeld C – Konfiguration von Pin (08) der EXT. 2-Stiftleiste	84
	Steckjumperfeld 'Force DHCP'	85

2.3	Installation und Inbetriebnahme .....	86
2.3.1	Hardware-Installation .....	86
2.3.2	Software-Installation .....	86
2.3.3	Erst-Inbetriebnahme (Test, ohne Benutzeranwendung) .....	86
2.4	Hinweise zur Migration und Neu-Programmierung von RTC6-Anwendungen .....	87
2.4.1	RTC6 Ethernet-Karten im Netzwerk finden und Eigenschaften abfragen .....	87
2.5	RTC6 Ethernet-Kartenbefehle und Funktionen .....	88
2.5.1	Hinweise zum Arbeiten mit IP-Adressen .....	88
2.5.2	Über die Suche von RTC6 Ethernet-Karten .....	88
2.5.3	Über die RTC6-Kartenverwaltung .....	89
2.5.4	Verbindung zur RTC6 Ethernet-Karte prüfen .....	90
2.5.5	Befehlssatz für die RTC6 Ethernet-Karte .....	90
	eth_assign_card .....	91
	eth_assign_card_ip .....	92
	eth_check_connection .....	93
	eth_convert_ip_to_string .....	94
	eth_convert_string_to_ip .....	95
	eth_count_cards .....	96
	eth_found_cards .....	97
	eth_get_card_info .....	98
	eth_get_card_info_search .....	99
	eth_get_com_timeouts .....	100
	eth_get_error .....	101
	eth_get_ip .....	102
	eth_get_ip_search .....	102
	eth_get_last_error .....	103
	eth_get_port_numbers .....	105
	eth_get_serial_search .....	105
	eth_get_static_ip .....	106
	eth_max_card .....	107
	eth_remove_card .....	108
	eth_search_cards .....	109
	eth_search_cards_range .....	110
	eth_set_com_timeouts .....	111
	eth_set_port_numbers .....	112
	eth_set_search_cards_timeout .....	113
	eth_set_static_ip .....	114
	get_card_type .....	115
2.6	Sichere Einschalt- und Ausschaltreihenfolge (zur Ausführung von Steuerungs-Software) .....	116
2.7	Technische Spezifikationen der RTC6 Ethernet-Karte .....	117
2.8	Konformität mit der EU-Richtlinie zur elektromagnetischen Verträglichkeit (EMV) .....	123
2.9	Konformität mit FCC-Bestimmungen .....	123
3	Anhang B: Die UFPM-Erweiterungskarte .....	124
4	Änderungsindex .....	127

# 1 Die RTC6 PCI-Express-Karte

## 1.1 Über dieses Handbuch

Die Gesamtdokumentation der RTC6-Karte umfasst folgende Teile:

- das vorliegende Handbuch
- das RTC5-Handbuch
- das Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten"

Das vorliegende Handbuch

- ergänzt und verweist auf das RTC5-Handbuch
- beschreibt die Unterschiede zwischen RTC5-Karten und RTC6-Karten
- nennt die excelliSCAN-Befehle nur (Beschreibungen siehe Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten")
- bezieht sich auf das RTC6-Software-Paket V1.4.1:

DLL-Datei für 32-Bit-Anwenderprogramme <sup>(a)</sup>	RTC6DLL.dll	Version 609 (DLL 609 <sup>(b)</sup> )
DLL-Datei für 64-Bit-Anwenderprogramme <sup>(a)</sup>	RTC6DLLx64.dll	Version 609 (DLL 609 <sup>(b)</sup> )
Programmdatei für den PCIe-DSP	RTC6OUT.out	Version 609 (OUT 609 <sup>(b)</sup> )
Programmdatei für den Eth-DSP	RTC6ETH.out	Version 609 (OUT 609 <sup>(b)</sup> )
Firmwaredatei für das FPGA	RTC6RBF.rbf	Version 613 (RBF 613 <sup>(b)</sup> )
Hilfsdatei	RTC6DAT.dat	Version 601 (DAT 601 <sup>(b)</sup> )

(a) Software für Laser-Scan-Prozesse, die RTC6-Karten auf Basis dieser DLL-Datei ansteuert, wird in diesem Handbuch einheitlich als "Anwenderprogramm" bezeichnet.

(b) Kurzform in diesem Handbuch.

### Hinweise

- Alle Verweise auf das RTC5-Handbuch beziehen sich auf Doc. Rev. 1.13 d.

## 1.2 Weitere Dokumentationen

Informationen zu den excelliSCAN Scan-Köpfen finden Sie in den entsprechenden Handbüchern.

## 1.3 Produktübersicht

### 1.3.1 Bestimmungsgemäße Verwendung

Die SCANLAB RTC6 PCI-Express-Karte ist zur synchronen Ansteuerung von Scan-Systemen, Lasern und Peripheriegeräten vorgesehen.

Hinweis: RTC6-Karten sind nicht als PCI-Variante lieferbar.

- Die RTC6 PCI-Express-Karte besitzt ein PCI-Express-Interface und muss in einen PCI-Express-Steckplatz des PCs eingesteckt werden.
- Die RTC6 PCI-Express-Karte stellt an der "McBSP/ANALOG" Stifteiste zwei 10 V-Analogeingänge bereit, siehe [Kapitel 1.4.5 "McBSP/ANALOG-Stifteiste"](#), Seite 14.

Ansonsten bietet die RTC6 PCI-Express-Karte die gleiche Funktionalität zur Ansteuerung von Scan-Systemen, Lasern und Peripheriegeräten wie die RTC5 PCI-Karte und RTC5 PCI-Express-Karte.

Anzahl, Art und Anordnung der Anschlussstecker zur Ansteuerung von Scan-Systemen, Lasern und Peripheriegeräten und die Jumper zur Konfiguration sind identisch zur RTC5 PCI-Express-Karte.

Es gelten die gleichen Spezifikationen für die Signaleingänge und Signalausgänge wie bei der RTC5 PCI-Express-Karte.

Die Software-Installation und die Quellcodeerstellung für Anwenderprogramme erfolgt identisch zur RTC5 PCI-Express-Karte (Ausnahmen, siehe unten).

Für die RTC6 PCI-Express-Karte wird ein eigenes Softwarepaket (Treiber, DLL, Importdeklarationen, etc.) und Befehlssatz geliefert.

### 1.3.2 Systemanforderungen

#### Hardware

Die RTC6 PCI-Express-Karte kann in jeden Windows-PC mit PCI-Express-Bus und mindestens einem freien PCI-Express-Steckplatz installiert werden.

Die Abmessungen der RTC6 PCI-Express-Karte zeigt [Abbildung 1](#).

RTC6 PCI-Express-Karten, die Master/Slave-synchronisiert werden sollen, müssen in benachbarte PCI-Express-Steckplätze gesteckt werden.

#### Software

Für die RTC6 PCI-Express-Karte müssen RTC6-Treiber und RTC6-DLL-Dateien verwendet werden. Zu den unterstützten Betriebssystemen siehe [Kapitel 1.12 "Technische Spezifikationen der RTC6 PCI-Express-Karte"](#), Seite 62.



#### Vorsicht!

- Die RTC6 PCI-Express-Karte unterstützt keine Energiesparmodi, in denen die Versorgungsspannungen des PCI-Express-Bus abgeschaltet werden. Entsprechende Standby- oder Sleep-Modi des Betriebssystems sind zu deaktivieren. Siehe hierzu auch Hinweis im RTC5-Handbuch, [Seite 26](#).



#### Vorsicht!

- Falls auf Ihrem PC bereits ein auf WDM-Technologie basierender RTC3/4/5-Kartentreiber
  - installiert ist oder
  - installiert war und deinstalliert wurde oder
  - Sie sich diesbezüglich nicht sicher sind:
    - (1) Installieren Sie zuerst den RTC6-Kartentreiber.
    - (2) Führen Sie `ScanlabClassChecker.cmd` als Administrator aus (im mitgelieferten Softwarepaket enthalten; siehe dort auch Hintergrundinformationen in `ReadMe_ScanlabClassChecker.pdf`). D.h. Schritt 2 kann auf neuen PCs, auf denen noch nie ein RTC-Kartentreiber installiert war, auch ausgelassen werden.



### 1.3.3 Optionale Funktionalitäten

Die RTC6 PCI-Express-Karte kann mit den gleichen Funktionalitäten ausgestattet werden wie die RTC5 PCI-Express-Karte, siehe RTC5-Handbuch, Kapitel 2.3 "Optionale Funktionalitäten", Seite 27.

Welche Optionen tatsächlich auf einer Karte freigeschaltet sind, kann mit **get\_rtc\_version** abgefragt werden.

Nur für RTC6-Karten sind zusätzlich erhältlich:

- **Option "SCANa" – Ansteuerung von Systemen mit SCANahead-Technologie**  
Wenn die Option "SCANa" freigeschaltet ist, können auch Scan-Systeme, die mit SCANahead-Technologie ausgestattet sind<sup>(1)</sup> (excelliSCAN-Baureihe), angesteuert werden. Weitere Informationen finden Sie im Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten". Die entsprechenden Befehle sind ansonsten nicht oder nur eingeschränkt nutzbar.
- **Option "UFPM"<sup>(2)</sup> – Pixelausgabemodi mit Pixelfrequenzen über 800 kHz**  
Wenn die Option "UFPM" freigeschaltet ist, können in verschiedenen Pixelausgabemodi auch Pixelfrequenzen von > 800 kHz...3,2 MHz erreicht werden, siehe auch **set\_pixel\_line**.
- **Option "syncA"<sup>(3)</sup> – Unterstützung der syncAXIS control-Software**  
RTC6-Karten mit der freigeschalteten Option "syncA" ermöglichen die Nutzung der SCANLAB syncAXIS control-Software (Karte und Software sind im SCANLAB-Lieferumfang für XL SCAN-Systeme enthalten).

### 1.3.4 Kennzeichnung

Die Seriennummer der RTC6 PCI-Express-Karte finden Sie auf einem Aufkleber auf der Karte.

Die Artikelnummer und die Konfiguration der Karte werden auf dem jeweiligen Lieferschein aufgeführt.

### 1.3.5 Typenbezeichnung

Aus der Artikelnummer einer RTC6 PCI-Express-Karte ergibt sich, in welcher Konfiguration die Karte ausgeliefert wird. Die jeweilige Jumperkonfiguration wird zusätzlich durch eine dreiziffrige Typenbezeichnung dargestellt. Die Typenbezeichnung ist identisch zu der RTC5 PCI-Express-Karte, siehe RTC5-Handbuch, Kapitel 16.1.5 "Typenbezeichnung", Seite 673.

### 1.3.6 Auspackanleitung und typischer Lieferumfang

- ▶ Nehmen Sie die RTC6 PCI-Express-Karte vorsichtig aus der Verpackung.
- ▶ Bewahren Sie sowohl die Umverpackung als auch den antistatischen Beutel, in dem die RTC6 PCI-Express-Karte ausgeliefert wird, auf, damit die Karte im Fall einer Reparatur wieder optimal verpackt an SCANLAB geschickt werden kann.
- ▶ Entnehmen Sie auch alle weiteren Artikel aus der Verpackung. Überprüfen Sie anhand des Lieferscheins die Vollständigkeit der Lieferung. Der Lieferumfang besteht typischerweise aus einer RTC6 PCI-Express-Karte und einer Daten-CD (mit der zugehörigen Software (s.u.) und diesem Manual). Unter Umständen sind aber auch Zusatzkomponenten wie Datenkabel oder Konverter enthalten, siehe RTC5-Handbuch, Kapitel 16.1.7 "Optionale Zusatzkomponenten", Seite 673.

### 1.3.7 Mitgelieferte Software

Das mitgelieferte RTC6-Software-Paket enthält u.a. RTC6-Kartentreiber<sup>(4)</sup> für die (32 Bit oder 64 Bit) Microsoft-Betriebssysteme Windows 10, 8, 7, Korrektur-, DLL- und Programm-Dateien, sowie Utility-Dateien für mehrere Programmiersprachen.

Beachten Sie den Sicherheitshinweis zu alten RTC-Kartentreibern in **Kapitel 1.3.2 "Systemanforderungen"**, Seite 8.

(1) Diese können nicht wie herkömmliche Scan-Köpfe betrieben werden und benötigen zur Ansteuerung zwingend RTC6-Karten mit SCANa-Option.

(2) Ultra Fast Pixel Mode.

(3) syncAXIS control.

(4) Der RTC6-Kartentreiber wird nur für RTC6 PCI-Express-Karten benötigt. Für RTC6 Ethernet-Karten wird er nicht benötigt.

### 1.3.8 Zubehör für die RTC6 PCI-Express-Karte

Zusätzlich zur RTC6 PCI-Express-Karte und zum zugehörigen Software-Paket sind bei SCANLAB – wie für die RTC5 PCI-Karte und RTC5 PCI-Express-Karte – die folgenden Komponenten erhältlich (in Kombination mit der RTC6 PCI-Express-Karte sollten nur Hardware-erweiterungen von SCANLAB verwendet werden):

- XY2-100-Konverter,  
siehe RTC5-Handbuch, Seite 29
- Laseradapter,  
siehe RTC5-Handbuch, Seite 29

Weiteres erhältliches Zubehör ist:

- Datenkabel,  
siehe RTC5-Handbuch, Seite 29
- Slotblech mit einer 9-poligen Sub-D-Buchse für die "2. SCANHEAD" Stiftleiste,  
siehe RTC5-Handbuch, Seite 43
- Slotblech mit einer 15-poligen Sub-D-Buchse für die "MARKING ON THE FLY" Stiftleiste,  
siehe RTC5-Handbuch, Seite 53

#### Hinweise

- Zur UFPM-Erweiterungskarte für die RTC6 PCI-Express-Karte<sup>(1)</sup> siehe **Kapitel 3 "Anhang B: Die UFPM-Erweiterungskarte"**, Seite 124.
- Die folgenden Komponenten können *nicht* mit RTC6 PCI-Express-Karten verwendet werden:
  - EXT1 Erweiterungskarte für die RTC5,  
#123804<sup>(2)</sup>
  - ADC-Aufsteckplatine,  
#121126
  - Erweiterungskarte  
RTC5 varioSCAN 40 FLEX Extension,  
#128683
  - RTC Step Motor Extension-Karte,  
#112097
  - I/O-Erweiterungskarte für RTC3 und RTC4,  
#108285, #121721

(1) Voraussetzung für Benutzer mit ANALOG angesteuerten Lasern, die Pixelfrequenzen > 100 kHz erzielen möchten, siehe **set\_pixel\_line**.

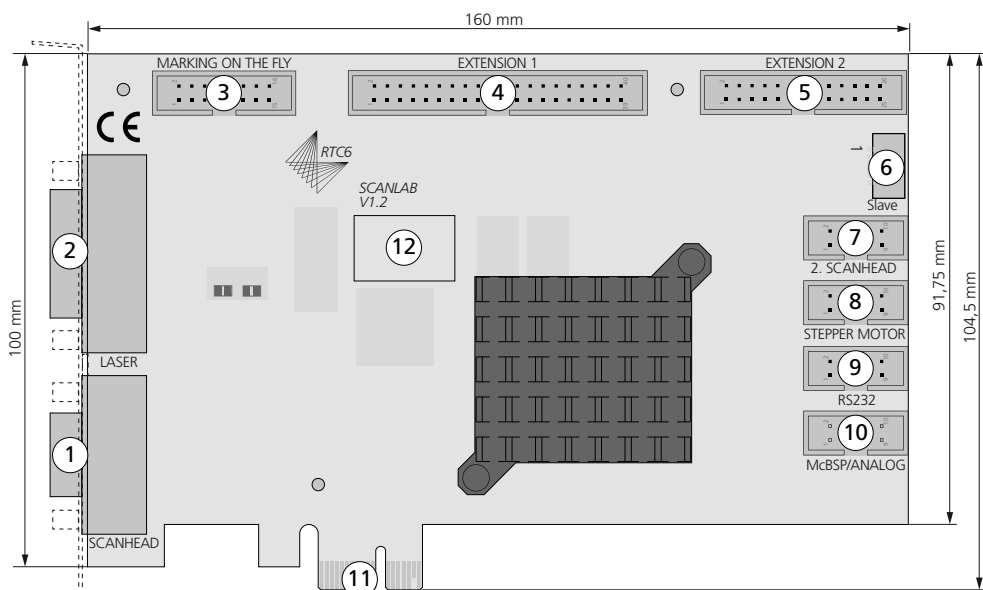
(2) Für RTC5 entwickelt. Kann aus mechanischen Gründen nicht mit der RTC6 PCI-Express-Karte verwendet werden (Kühlkörper verhindert, dass diese EXT1 Erweiterungskarte vollständig eingesteckt werden kann).

### 1.3.9 Ergänzende Software

Zur individuellen Anpassung von RTC-Korrekturdateien entsprechend der Daten eigener Testmessungen ist bei SCANLAB Software mit zugehörigem Handbuch aus der correXion-Reihe erhältlich, siehe RTC5-Handbuch, Seite 137.

## 1.4 Layout und Schnittstellen

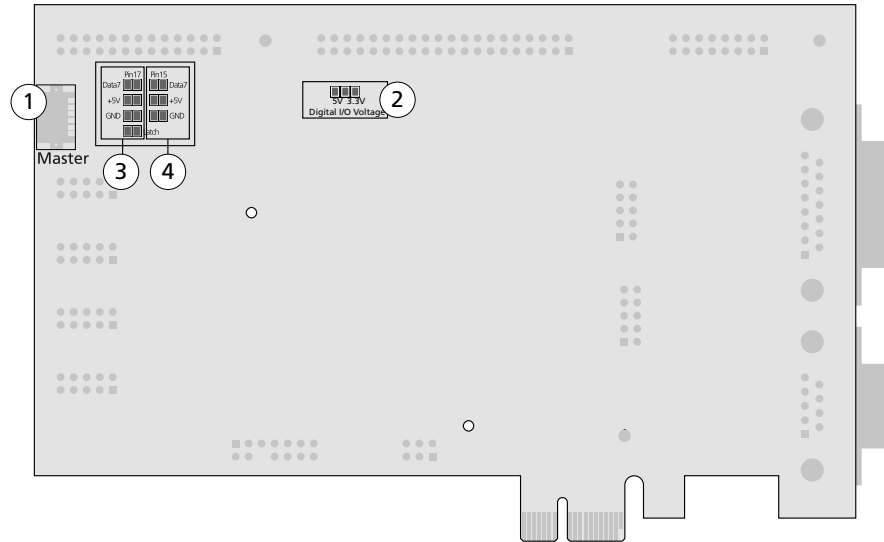
### 1.4.1 Layout – Oberseite



#### Legende

- |                       |   |
|-----------------------|---|
| 1. SCANHEAD           | 9-polige Sub-D-Buchse, weiblich. Primärer Scan-Kopf-Anschluss.<br>Pinbelegung siehe RTC5-Handbuch, Seite 42.  |
| 2. LASER              | 15-polige Sub-D-Buchse, weiblich. Laseranschluss, inkl. digitaler und analoger Ausgänge.<br>Pinbelegung siehe RTC5-Handbuch, Seite 47.  |
| 3. MARKING ON THE FLY | 16-polige Stiftleiste. Mit Encoder-Eingängen für "Processing on the fly"-Anwendungen.<br>Immer bestückt, aber aktivierte Signale sind eine zukaufbare Option.<br>Pinbelegung siehe RTC5-Handbuch, Seite 53. |
| 4. EXTENSION 1        | 40-polige Stiftleiste. Mit einem 16 Bit-Digital-Eingang und einem 16 Bit-Digital-Ausgang.<br>Pinbelegung siehe RTC5-Handbuch, Seite 51.   |
| 5. EXTENSION 2        | 26-polige Stiftleiste. Mit einem 8-Bit-Digital-Ausgang, z. B. für eine alternative Laseransteuerung.<br>Pinbelegung siehe RTC5-Handbuch, Seite 52.  |
| 6. SLAVE              | 6-polige Stiftleiste.<br>Details siehe RTC5-Handbuch, Seite 41.   |
| 7. 2. SCANHEAD        | 10-polige Stiftleiste. Anschluss für einen zweiten Scan-Kopf. Immer bestückt, aber aktivierte Signale sind eine zukaufbare Option. Pinbelegung siehe RTC5-Handbuch, Seite 43.                               |
| 8. STEPPER MOTOR      | 10-polige Stiftleiste. Zur Ansteuerung von bis zu zwei Schrittmotoren.<br>Pinbelegung siehe RTC5-Handbuch, Seite 58.  |
| 9. RS232              | 10-polige Stiftleiste. Zur Ansteuerung eines externen Geräts über RS-232-Schnittstelle.<br>Pinbelegung siehe RTC5-Handbuch, Seite 54.   |
| 10. McBSP/ANALOG      | 10-polige Stiftleiste. Serielle Erweiterungsschnittstelle.<br>Pinbelegung siehe <a href="#">Seite 14</a> .  |
| 11. PCIe-x1-Stecker   | Schnittstelle zum PC, siehe <a href="#">Seite 12</a> .  |
| 12.                   | Seriennummernaufkleber.   |

## 1.4.2 Layout – Unterseite



### Legende

1. Master . . . . . 6-polige Stiftleiste. Zum Verbinden mit einer anderen RTC6 PCI-Express-Karte zwecks Takt-synchronisierung. Details siehe RTC5-Handbuch, Seite 41.
2. Lötjumperfeld A. . . . . Zum Konfigurieren des Pegels der Ausgangssignale an der EXTENSION 1-Stiftleiste. Details siehe RTC5-Handbuch, Seite 51.
3. Lötjumperfeld B. . . . . Zum Konfigurieren des Signals an EXTENSION 2-Stiftleiste Pin (17). Details (analog zu JP2...JP8 der RTC5) siehe RTC5-Handbuch, Seite 52.
4. Lötjumperfeld C. . . . . Zum Konfigurieren des Signals an EXTENSION 2-Stiftleiste Pin (15). Details (analog zu JP2...JP8 der RTC5) siehe RTC5-Handbuch, Seite 52.

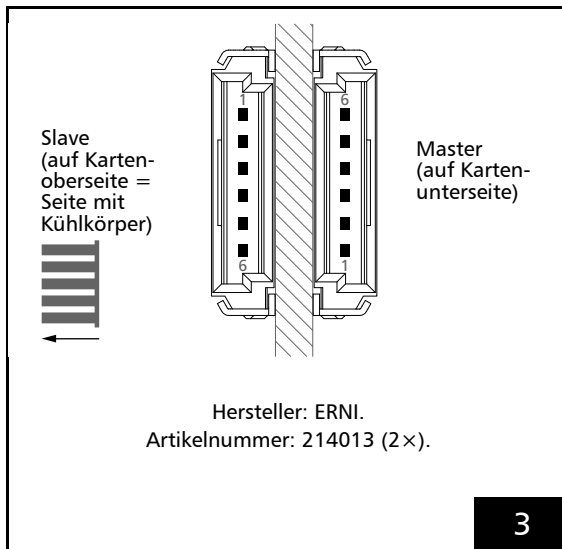
## 1.4.3 Schnittstelle zum PC

Schnittstelle zum PC ist der PCIe-x-1-Stecker.

Die RTC6 PCI-Express-Karte kann in jeden Windows-PC mit PCI-Express-Bus und mindestens einem freien PCI-Express-Steckplatz installiert werden.

## 1.4.4 Master/Slave-Taktsynchronisierung: Master-Stiftleiste, Slave-Stiftleiste

Sowohl die Slave-Stiftleiste als auch die Master-Stiftleiste ist 6-polig, siehe **Abbildung 3**. Die Slave-Stiftleiste befindet sich auf der Oberseite der RTC6 PCI-Express-Karte, siehe **Abbildung 1**, die Master-Stiftleiste auf der Unterseite, siehe **Abbildung 2**<sup>(1)</sup>.



Slave-Stiftleiste und Master-Stiftleiste.  
Das Rastermaß der Pins ist 1,27 mm.

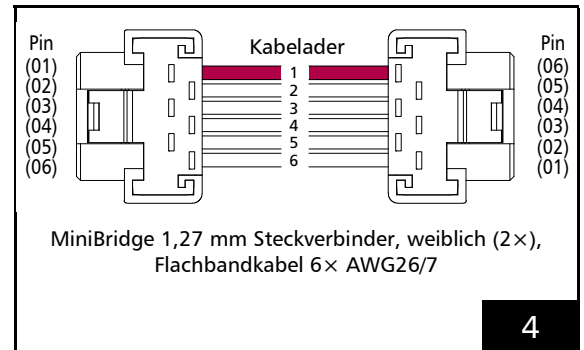
Zweck der beiden Stiftleisten ist, eine Taktsynchronisierung mehrerer RTC6 PCI-Express-Karten zu ermöglichen.

Wenn mehrere RTC6 PCI-Express-Karten mit synchroner Taktung in einem PC betrieben werden sollen, dann müssen die RTC6 PCI-Express-Karten zunächst paarweise über die Master- und Slave-Stiftleisten miteinander verbunden werden. Verbinden Sie jeweils die Master-Stiftleiste einer Karte mit der Slave-Stiftleiste einer anderen Karte mit einem geeigneten Kabel. Dazu ist ein 50 mm langes Verbindungskabel (#117241) bei SCANLAB erhältlich. Benutzer, die ein passendes Kabel selber konfektionieren wollen, finden die notwendigen Angaben in **Abbildung 4**.

Miteinander verbundene RTC6 PCI-Express-Karten müssen in benachbarte PCIe-Steckplätze gesteckt sein.

Siehe auch **Kapitel 1.9 "Master/Slave-Betrieb"**, **Seite 52**.

(1) Ist bei RTC6 Ethernet-Karten umgekehrt.



Kabel zum Verbinden von Master-Stiftleiste und Slave-Stiftleiste: Anforderungen.

#### 1.4.5 McBSP/ANALOG-Stiftleiste

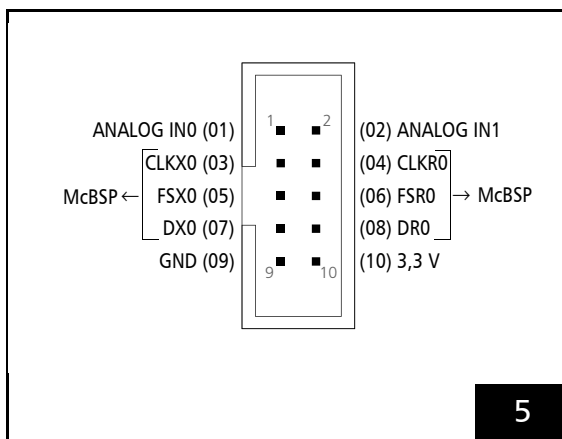
Die McBSP/ANALOG-Stiftleiste (bei RTC5-Karten mit "SPI / I2C" Stiftleiste bezeichnet) stellt zwei Analog-Eingänge sowie eine McBSP-Schnittstelle (Multi-channel Buffered Serial Port) bereit.

Der Datenübertragungsstandard mittels Software-befehl von McBSP auf SPI (Serial Peripheral Interface) kann bei RTC6-Karten *nicht* umgestellt werden.

Die Pinbelegung zeigt **Abbildung 5**.

#### Hinweise

- Die ADC-Aufsteckplatine (#121126, optionales Zubehör für die RTC5 PCI-Karte) ist mit der RTC6 PCI-Express-Karte *nicht verwendbar*.



McBSP/ANALOG-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,54 mm.

#### Funktion als McBSP-Schnittstelle (Werksvoreinstellung)

Die McBSP-Schnittstelle samt zugehöriger Befehle funktionieren identisch zur RTC5 PCI-Karte und RTC5 PCI-Express-Karte, siehe RTC5-Handbuch, Kapitel 4.4.6 "SPI / I2C-Stiftleiste", Seite 54.

#### Funktion als SPI-Schnittstelle (Benutzereinstellung)

Die SPI-Schnittstelle steht bei RTC6-Karten nicht zur Verfügung.

#### Analog-Eingänge

Genauso wie die RTC5 PCI-Express-Karte, stellt die RTC6 PCI-Express-Karte zwei 10 V-Analog-Eingänge (ANALOG IN0 und ANALOG IN1) bereit.

Zur Verwendung und zu den Spezifikationen der beiden Analog-Eingänge: siehe RTC5-Handbuch, Kapitel 4.4.8 "Analog-Eingänge (Sonderzubehör)", Seite 59.

## 1.5 Installation und Inbetriebnahme

Die Installation und Inbetriebnahme der RTC6 PCI-Express-Karte und der zugehörigen Software erfolgt analog zu RTC5-Karten, siehe RTC5-Handbuch, Kapitel 5 "Installation und Inbetriebnahme", Seite 60.



### Vorsicht!

- Die RTC6 PCI-Express-Karte unterstützt keine Energiesparmodi, in denen die Versorgungsspannungen des PCI-Express-Bus abgeschaltet werden. Entsprechende Standby- oder Sleep-Modi des Betriebssystems sind zu deaktivieren. Deaktivieren Sie insbesondere den Windows-Sleep-Modus (dieser ist bei einigen Windows-Betriebssystemen standardmäßig voreingestellt). Siehe hierzu auch Hinweis im RTC5-Handbuch, [Seite 26](#).



### Vorsicht!

- Falls auf Ihrem PC bereits ein auf WDM-Technologie basierender RTC3/4/5-Kartentreiber
  - installiert ist oder
  - installiert war und deinstalliert wurde oder
  - Sie sich diesbezüglich nicht sicher sind:
    - (1) Installieren Sie zuerst den RTC6-Kartentreiber.
    - (2) Führen Sie `ScanlabClassChecker.cmd` als Administrator aus (im mitgelieferten Softwarepaket enthalten; siehe dort auch Hintergrundinformationen in `ReadMe_ScanlabClassChecker.pdf`). D.h. Schritt 2 kann auf neuen PCs, auf denen noch nie ein RTC-Kartentreiber installiert war, auch ausgelassen werden.

## 1.6 Über die RTC6-Karte und Hinweise an RTC5-Kartenbenutzer

Mit der RTC6-Karte können Scan-Köpfe mit SL2-100-Interface und auch Scan-Köpfe mit XY2-100-Interface (zusammen mit dem XY-Konverter) angesteuert werden. Nutzer solcher Scan-Köpfe sollten das RTC5-Handbuch und dieses Handbuch lesen.

Nutzer des excelliSCAN-Scan-Kopfs müssen *zusätzlich* das Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten" durchlesen.

Es folgt eine Übersicht über die wichtigsten Änderungen, die bei der RTC6-Karte im Vergleich zur RTC5-Karte eingeführt werden.

Unter anderem wird im Folgenden eine mögliche Vorgehensweise zum Überarbeiten von RTC5-Programmcode für die Verwendung mit RTC6-Karten beschrieben.

### 1.6.1 RTC6-Karten und RTC5-Karten im Vergleich – Neuerungen und Änderungen

Die RTC6-Karte hat im Vergleich zur RTC5-Kartenfamilie neuere, leistungsstärkere Hardware-Komponenten. Das betrifft den Digitalen Signal Prozessor (DSP), das FPGA (Field Programmable Gate Array) und den Hauptspeicher.

Für Umsteiger von RTC5-Karten auf die RTC6-Karte (betrifft sowohl Betreiber von NICHT-excelliSCAN Scan-Köpfen und auch von excelliSCAN-Scan-Köpfen) bedeutet das im Detail:

- Für bestehende Anwenderprogramme sind nur wenige Anpassungen erforderlich, siehe [Kapitel 1.7.2 "RTC5-Quellcode für die RTC6 anpassen", Seite 21](#).
- Kurze Vektoren<sup>(1)</sup> führt die RTC6 schneller aus als RTC5-Karten (siehe [set\\_dsp\\_mode](#)).
- Der RTC6-Listenspeicher ist  $8\times$  größer als der der RTC5. Er umfasst insgesamt  $8.388.608 (= 2^{23})$  Listenplätze. Liste 1 und Liste 2 werden mit je  $4.194.304 (= 2^{22})$  Listenplätzen initialisiert (RTC5: 4.000 und 4.000).
- Der Pixelausgabe-Modus ist mit RTC6-Software-Paket V1.3 erheblich erweitert, siehe [set\\_pixel\\_line](#). RTC6-Karten ohne Option "UFPM" können damit out-of-the-box Pixelfrequenzen von 800 kHz erzielen (RTC5-Karten: 308 kHz), RTC6-Karten, auf denen die Option "UFPM" freigeschaltet ist, sogar bis zu 3,2 MHz. Außerdem ist es mit [set\\_pixel\\_line](#) nun möglich, die Laserleistungs-Steuerwerte digital auszugeben (bei RTC5-Karten mit Standard-Firmware: nicht möglich). Das erfolgt an der EXTENSION 2-Stiftleiste (=EXT. 2-Stiftleiste der RTC6 Ethernet-Karte).
- Erweiterte Messwert-Aufzeichnung mit der RTC6: 4 Aufzeichnungskanäle für je 8.388.608 Datenwerte (siehe [set\\_trigger4](#)).

(1) "Strichlinien"



- Acht 3D-Korrekturtabellen können auf der RTC6-Karte gleichzeitig mit vier Aufzeichnungskanälen (siehe **set\_trigger**, **set\_trigger4**) und dem kompletten Listenspeicher verwendet werden (RTC5: Acht 3D-Korrekturtabellen, die aber nicht gleichzeitig mit vier Aufzeichnungskanälen und dem kompletten Listenspeicher verwendet werden können). Mit **number\_of\_correction\_tables** kann eine benutzerseitige Begrenzung auf weniger als 8 Korrekturtabellen zur Prüfung auf fehlerhafte Benutzereingaben eingestellt werden.
- Das virtuelle Bildfeld der RTC6 ist (ab DLL 609, OUT 609)  $32 \times$  größer als das der RTC5. Es beträgt nun  $\pm 28 \text{ Bit} = 29 \text{ Bit}$  (RTC5:  $\pm 23 \text{ Bit} = 24 \text{ Bit}$ ). Damit sind für "Processing on the fly" nun X- und Y-Koordinatenwerte von  $-268.435.456 \dots +268.435.455$  möglich (RTC5:  $-8.388.608 \dots +8.388.607$ ). Für die Koordinatentransformationen im virtuellen Bildfeld gilt:  
**set\_offset\_xyz**( HeadNo = 4, X, Y, Z)  
 Wertebereich für X,Y  $\pm 28 \text{ Bit}$ , für Z  $\pm 19 \text{ Bit}$   
**set\_matrix**( HeadNo = 4, M11, M12, M21, M22 ):  
 Wertebereich für M11, M12, M21, M22  $\pm 2.0$
- Die RTC6-Karte verwendet intern für 3D-Anwendungen bei Defocus-Werten und Z-Koordinaten jetzt ebenso wie für X- und Y-Werte 20-Bit (RTC5: 16-Bit). Bezüglich der Abwärtskompatibilität zur RTC5-Karte siehe Schritt (3), Seite 22).
- Für 3D-Systeme mit i.d.R. unterschiedlichem Schleppverzugsverhalten von XY-Achsen und Z-Achse kann eine entsprechende Schleppverzugskompensation eingestellt werden, siehe **set\_timelag\_compensation**.
- Die Ausgabe-Synchronisation der RTC5 wird bei der RTC6 durch eine Zyklus-Synchronisation ersetzt, siehe folgender Abschnitt.
- Die Laser-Delays (Parameter `LaserOnDelay` und `LaserOffDelay` von **set\_laser\_delays**) sowie `LaserOnShift` bei Sky-Writing-Befehlen haben bei der RTC6 eine feinere Auflösung von  $1/64 \mu\text{s}$  (RTC5:  $0,5 \mu\text{s}$ ). Bezüglich der Abwärtskompatibilität zur RTC5-Karte siehe Schritt (3), Seite 22.
- Der Parameter `Timelag` von **set\_sky\_writing\_para** wird zwar im 64-Bit IEEE Gleitkommaformat ("double") angegeben ( $1,0 = 1 \mu\text{s}$ ), wird aber jetzt auf der RTC6 tatsächlich mit  $1/64 \mu\text{s}$  ausgeführt.

## Synchronisierung von RTC6-Zyklus und externem Taktsignal

Die Laserpulssignale eines freilaufenden Lasers und die Lasersteuerung der RTC6-Karte können synchronisiert werden. Dazu stellt die RTC6-Karte die Funktionalität bereit, den RTC6-internen 10  $\mu$ s-Takt und ein externes Taktsignal zu synchronisieren.

Die RTC6-Karte akzeptiert (im Gegensatz zu RTC5-Karten) dabei das externe Taktsignal (z. B. Laserpulssignal) als Master-Clock. Es muss am Digital-Eingang DIGITAL IN1 der LASER-Buchse eingespeist werden.

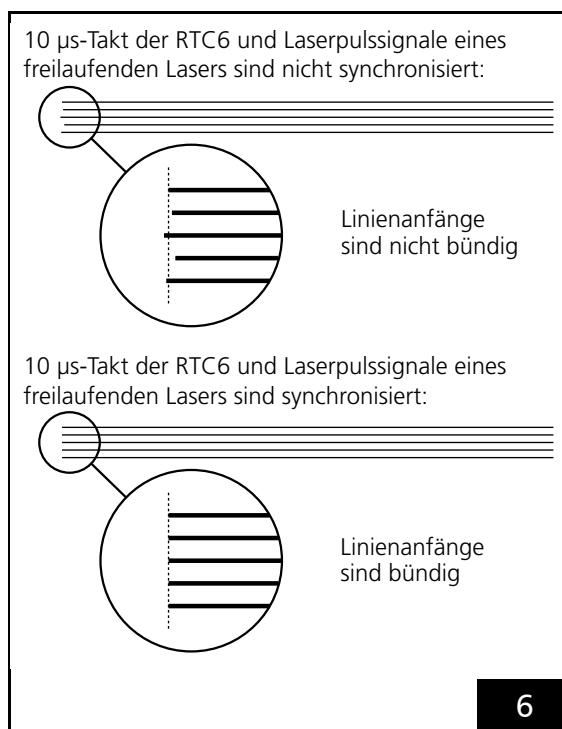
Die Synchronisierung von RTC6-Karte und externem Takt wird mit **set\_laser\_control** über das Bit#6 ein- und ausgeschaltet.

Zusätzlich kann mit **set\_laser\_control** über das Bit#5 gewählt werden, ob auf die fallende oder die steigende Flanke des externen Taktes synchronisiert werden soll.

Bei aktivierter Synchronisierung werden der interne 10  $\mu$ s-Takt der RTC6-Karte und der externe Takt an DIGITAL IN1 der LASER-Buchse synchronisiert. Deshalb wirkt sich die Synchronisierung auf die gesamte Listenausführung aus. Alle Markierbefehle (Mark, Arc und Ellipse) werden automatisch synchron ausgeführt. Nichtbündige Linienanfänge kommen dadurch nicht mehr vor, siehe **Abbildung 6**.

## Hinweise gültig für Version $\geq$ RBF 603

- Die Frequenz der Master-Clock muss ein ganzzahliges Vielfaches von 100 kHz sein. Minimal zulässig sind 100 kHz. Die maximal zulässige Abweichung der externen Frequenz gegenüber einem ganzzahligen Vielfachen von 100 kHz beträgt hierbei  $\pm 15,625$  ns pro 10  $\mu$ s.
- Bei dem eingespeisten Taktsignal muss es sich um ein TTL-Signal handeln. Die minimale Pulslänge oder Pulspause des Taktsignals sollte 80 ns betragen. Siehe hierzu auch die Informationen im RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Abschnitt "Digitaler Eingang und Ausgang", Seite 48.
- Liegt bei aktivierter Synchronisierung kein gültiges Taktsignal an DIGITAL IN1 der LASER-Buchse an, erfolgt keine Synchronisierung. Die RTC6-Karte verwendet in diesem Fall ihren internen 10  $\mu$ s Takt.
- Wird die Synchronisierung über den Befehl **set\_laser\_control** durch Setzen von Bit#6 aktiviert, erfolgt die Synchronisierung der RTC6-Karte schleichend auf den externen Takt (abhängig von der jeweiligen Phasenlage innerhalb von höchstens 1,2 ms).



Freilaufender Laser und RTC6 – Beispiel für markierte Linien.

Benutzer, die mit der RTC6-Karte excelliSCAN-Scan-Köpfe ansteuern, profitieren von deren Alleinstellungsmerkmalen:

- Nur die RTC6-Karte ermöglicht die Nutzung der SCANAhead-Technologie, z. B. bei Scan-Köpfen der excelliSCAN-Baureihe.  
*Mit der RTC5 können keine excelliSCAN-Scan-Köpfe angesteuert werden!*
- Wird die RTC6-Karte mittels **set\_scanahead\_params** für die Benutzung mit einem excelliSCAN-Scan-Kopf konfiguriert, dann wird bei der Lasersteuerung automatisch die *PreviewTime* der SCANAhead-Regelungstechnik (d.h. die Durchlaufzeit für die Berechnung und Ausführung der Scanner-Trajektorie) berücksichtigt. Dynamik und Genauigkeitsvorteile werden so optimal genutzt.
- Mit dem RTC6-Befehl **activate\_scanahead\_autodelays** kann der excelliSCAN einfach und schnell zum Einsatz gebracht werden. Weder Laser-Delays noch Scanner-Delays müssen selbst ermittelt oder gesetzt werden.
- Aktuell (DLL 603) gibt es für excelliSCAN-Scan-Köpfe folgende Einschränkungen:
  - Siehe *RTC6\_Release\_Notes\_<Datum>.pdf* im mitgelieferten Softwarepaket.
- Zu weiteren Details siehe eigenständiges Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANAhead-Regelung und Ansteuerung mit RTC6-Karten". Dort finden Sie auch die Beschreibungen der excelliSCAN-spezifischen Befehle des RTC6-Befehlssatzes.

### 1.6.2 RTC6-Karten und RTC5-Karten im Vergleich – was bleibt unverändert

Sowohl für RTC5-Karten und RTC6-Karten gilt das Folgende:

- Die gleichen zukaufbaren Optionen sind lieferbar. Das sind im Einzelnen:
  - Option “Processing on the fly” (aktuell mit DLL 603 bei RTC6-Karten nur für Scan-Köpfe ohne SCANAhead-Technologie nutzbar).
  - Option 3D (zur Steuerung eines 3-Achsen-Scan-Systems).
  - Option SSHC (zur Steuerung eines zweiten Scan-Kopfs).
  - Option DC/DC<sup>(1)</sup> (RTC6-Karten: aktuell noch nicht lieferbar).
- Alle neueren<sup>(2)</sup> Windows-Versionen werden unterstützt: 32-Bit Treiber und 64-Bit Treiber für Windows 10, 8, 7 werden mitgeliefert.
- Die kartenspezifischen DLL-Dateien zur Entwicklung von 32-Bit- und 64-Bit-Anwendungen werden mitgeliefert.
- In einem PC sind bis zu 255 RTC6 PCI-Express-Karten möglich<sup>(3)</sup>.
- Die elektrischen Anschlüsse und Jumper (technische Spezifikationen) sind gleich.
- Die Steuerung von Laser und Peripherie verhält sich identisch zur RTC5-Karte, wenn kein excelliSCAN verwendet wird.
- Die Synchronisation der Steuerung von Scan-Kopf und Laser erfolgt bei der RTC6 und der RTC5 gleich (das ist in Kap 7.2 beschrieben; mit Ausnahme der automatischen Delay-Anpassung, die bei der RTC6 im DSP-Modus 3 entfällt. Aus Kompatibilität zur RTC5 wird sie im DSP-Modus 2 wie bei der RTC5 beschrieben ausgeführt).
- Die Scan-Kopf-Schnittstelle ist gleich (SL2-100-Protokoll, 10  $\mu$ s Übertragungstakt).

- Die Steuerung von Scan-Systemen erfolgt gleich<sup>(4)</sup>.
- Die Lasersteuersignale sind gleich.
- Das Laser-Markieren erfolgt gleich.
- Das Verkabelungsprinzip ist gleich<sup>(5)</sup>.
- Sonstiges
  - RTC5 und RTC6 können nicht gemeinsam als Master/Slave betrieben werden!

(1) Optokoppler.

(2) Ab RTC6-Software-Paket V1.3: Vista und XP werden nicht mehr unterstützt.

(3) Bedingt durch die Kapazität der RTC6-Kartenverwaltung. Diese kann 255 [RTC6 PCI-Express-Karten + RTC6 Ethernet-Karten] verwalten, siehe Kapitel 2.5.3 “Über die RTC6-Kartenverwaltung”, Seite 89.

(4) Der “optionale XY2-100-Konverter” kann bei beiden Karten verwendet werden.

(5) Bezüglich einer korrekten Verkabelung mit den Systemkomponenten müssen beim Wechsel von der RTC5 auf RTC6 keine Hardware-Änderungen berücksichtigt werden, siehe RTC5-Handbuch, Kapitel 2.7.1 “Hardware-Änderungen”, Seite 31. Dieses Kapitel gilt also für die RTC6-Karte auch bzw. analog, siehe dort Abschnitte Steuerung von Scan-Systemen, Lasersteuerung, EXTENSION 1-Stiftleiste, EXTENSION 2-Stiftleiste, MARKING ON THE FLY-Stiftleiste).

### 1.6.3 Einschränkungen bei RTC6-Karten

#### Verfügbarkeit technischer Varianten

RTC6-Karten sind als RTC6 PCI-Express-Karten und RTC6 Ethernet-Karten<sup>(1)</sup> lieferbar.

RTC5-Karten sind als PCI-, PCIe-, PC/104-Plus- und PCIe/104-Versionen erhältlich, RTC4-Karten als PCIe- und Ethernet-Versionen.

#### Schnittstellen zur Peripherie

Die ADC-Aufsteckplatine (#121126, optionales Zubehör für die RTC5 PCI-Karte) ist mit der RTC6 PCI-Express-Karte nicht verwendbar.

Stattdessen stellt die RTC6 PCI-Express-Karte an der "McBSP/ANALOG" Stiftleiste – wie die RTC5-PCI-Express-Karte an der "SPI / I2C" Stiftleiste – die zwei 10 V-Analog-Eingänge ANALOG IN0 und ANALOG IN1 bereit, siehe auch [Kapitel 1.4.5 "McBSP/ANALOG-Stiftleiste"](#), Seite 14.

#### Funktionalität

Bei der RTC6 PCI-Express-Karte kann die McBSP/ANALOG Stiftleiste – im Gegensatz zur "SPI / I2C" Stiftleiste der RTC5-Karten – nicht im SPI-Modus betrieben werden, siehe auch [Kapitel 1.4.5 "McBSP/ANALOG-Stiftleiste"](#), Seite 14.

Weitere vorläufige Einschränkungen sind in RTC6\_Release\_Notes\_<Datum>.pdf im mitgelieferten RTC6-Software-Paket genannt.

### 1.7 Quellcode für RTC6-Anwenderprogramme

#### 1.7.1 RTC6-Quellcode neu erstellen

Die Neuerstellung von Quellcode für RTC6-Programme erfolgt wie im RTC5-Handbuch Kapitel 6.2 "Initialisierung und Programmstart", Seite 66 beschrieben, aber es ist [Kapitel 1.7.2 "RTC5-Quellcode für die RTC6 anpassen"](#), Seite 21 dieses Dokuments zu beachten.

#### 1.7.2 RTC5-Quellcode für die RTC6 anpassen

Anwenderprogramme, die für die RTC5 (also zur Ansteuerung von NICHT-excelliSCAN-Scan-Köpfen) geschrieben wurden, können für die RTC6 nur dann verwendet werden, wenn der Programmcode überarbeitet wird und einige Voraussetzungen erfüllt sind, wie im Folgenden beschrieben.

Dies gilt auch dann, wenn der eigentliche Ablauf des Programms unverändert bleibt und keine der neuen Funktionalitäten der RTC6 in Anspruch genommen werden soll.

Zur Ansteuerung eines excelliSCAN-Scan-Kopfs müssen zusätzliche Schritte beachtet werden, siehe Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten".

#### Zwingend erforderliche Schritte

(1) Stellen Sie sicher, dass folgende Voraussetzungen erfüllt sind:

- RTC6-Karte ist eingebaut.
- Der RTC6-Treiber für das Betriebssystem ist installiert.
- Die folgenden Dateien sind im Anwendungs-Software-Verzeichnis vorhanden:  
 RTC6DLL.dll oder RTC6DLLx64.dll,  
 RTC6OUT.out (DSP-Programmdatei für RTC6 PCI-Express-Karten),  
 RTC6ETH.out (DSP-Programmdatei für RTC6 Ethernet-Karten),  
 RTC6RBF.rbf (Firmwaredatei),  
 RTC6DAT.dat (binäre Hilfsdatei),  
 \*.CT5 (Korrekturdatei/en).

(1) Standalone-Funktionalität ist in Entwicklung.

- Die Importdeklarationen wurden in die Software importiert und es wird als Aufrufkonvention `std_call` verwendet, siehe RTC5-Handbuch, Kapitel 6.2 "Initialisierung und Programmstart", Seite 66.

(2) Berücksichtigen Sie die Befehlsnamensanpassungen des RTC6-Befehlssatzes.

- Ändern Sie *nicht* pauschal `_rtc5_` zu `_rtc6_`!
- `init_rtc5_dll` zu `init_rtc6_dll` ändern.
- `free_rtc5_dll` zu `free_rtc6_dll` ändern.
- `rtc5_count_cards` zu `rtc6_count_cards` ändern.
- Achtung: Ändern Sie *nicht* pauschal `set_rtc5_mode` zu `set_rtc6_mode`, siehe dazu Schritt (3).

(3) Der Kontrollbefehl **set\_rtc6\_mode** wird bei Programmstart automatisch eingestellt. Damit steht eine erweiterte Parameterauflösung für Z-Koordinaten von 20 Bit (RTC5: 16 Bit) und Laser-Delays von  $1/64 \mu s$  (RTC5:  $1/2 \mu s$ ) zur Verfügung.

Im Gegensatz dazu gewährleistet der Kontrollbefehl **set\_rtc5\_mode** die Abwärtskompatibilität des Quellcodes<sup>(1)</sup> zur RTC5 ("RTC5-Kompatibilitätsmodus").

- Laser-Delays:  
Im RTC5-Kompatibilitätsmodus werden die Laser-Delays (Parameter `LaserOnDelay` und `LaserOffDelay` des Befehls **set\_laser\_delays**) automatisch mit 32 multipliziert. Das gilt auch für den Parameter `LaserOnShift` von **set\_sky\_writing** und **set\_sky\_writing\_para**.
- Z-Koordinaten (betrifft 3D-Anwendungen):  
Im RTC5-Kompatibilitätsmodus werden die Z-Koordinaten und Defocus-Werte (z.B. Parameter `Shift` von **set\_defocus**) automatisch mit 16 multipliziert.

Für weitere Anpassungen und Optimierungen an Ihrem Code, siehe folgendes Kapitel.

## Optionale Schritte und Hinweise zu weiteren Anpassungen und Optimierungen

- Aus dem RTC6-Befehlssatz entfernte und nicht mehr unterstützte Befehle müssen aus dem Programmcode entfernt oder ersetzt werden, siehe Kapitel 1.8.3 "Nicht mehr vorhanden im RTC6-Befehlssatz", Seite 51.
- Prüfen Sie, ob die Änderungen an den Befehlen für Ihr Anwenderprogramm relevant sind, siehe Kapitel 1.8.2 "Geändert im RTC6-Befehlssatz", Seite 33, und passen Ihren Code entsprechend an.
- Beachten Sie das andere Verhalten von "load\_program\_file", Seite 37. Vorsicht: Der Befehl hält jetzt ohne Warnung die Liste an.
- **Hinweis zur erhöhten Parameterauflösung**  
Beachten Sie die erweiterte Parameterauflösung von Z-Koordinaten, wenn **set\_rtc6\_mode** verwendet wird. Siehe auch Schritt (3), Seite 22.
- **Hinweis zu verändertem Zeitverhalten**  
Durch den Wegfall der automatischen Delay-Anpassung kann sich für kurze Vektoren ein verändertes Zeitverhalten ergeben.  
Mit `set_dsp_mode( 2 )` kann das ursprüngliche Zeitverhalten der RTC5 wieder hergestellt werden (siehe **set\_dsp\_mode**).

(1) Betrifft nicht die Befehlsnamen im Quellcode.

## 1.8 RTC6-Befehlssatz

Grundsätzlich sind die Befehle des RTC6-Befehlssatzes und RTC5-Befehlssatzes gleich. Auf die wenigen Ausnahmen wird in den folgenden Kapiteln eingegangen:

- *Neue* Befehle, siehe **Kapitel 1.8.1 "Neu im RTC6-Befehlssatz", Seite 23**
- *Geänderte* Befehle, siehe **Kapitel 1.8.2 "Geändert im RTC6-Befehlssatz", Seite 33**
- *Entfernte* Befehle, siehe **Kapitel 1.8.3 "Nicht mehr vorhanden im RTC6-Befehlssatz", Seite 51**

Die Angaben zu den Befehlen im RTC5-Handbuch, Kapitel 10.2 "RTC5 Befehlssatz", Seite 261, gelten weiterhin, wenn nicht ausdrücklich anders vermerkt.

### Multi-Board-Befehle

Die meisten RTC6-Befehle werden – wie die RTC5-Befehle – zusätzlich auch als Multi-Board-Befehl bereitgestellt. Bei Befehlen, bei denen das nicht der Fall ist, wird in der Befehlsbeschreibung explizit darauf hingewiesen. Multi-Board-Befehlsnamen beginnen mit dem Vorsatz **n\_**.

Zur Verwendung von Multi-Board-Befehlen siehe RTC5-Handbuch, Kapitel 6.6.1 "Multi-Board-Programmierung", Seite 92.

### 1.8.1 Neu im RTC6-Befehlssatz

Im RTC6-Befehlssatz gibt es neue Befehle, die nur für Scan-Köpfe mit SCANahead-Technologie und solche, die für alle Scan-Kopftypen nutzbar sind.

#### Nur für Scan-Köpfe mit SCANahead-Technologie

Die folgenden Befehle sind nur für Scan-Köpfe mit SCANahead-Technologie vorgesehen und daher im Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten" beschrieben:

```
(n_) activate_scanahead_autodelays con (1)
(n_) activate_scanahead_autodelays_list uk (2)
(n_) get_scanahead_params con
(n_) set_scanahead_laser_shift con
(n_) set_scanahead_laser_shift_list uk
(n_) set_scanahead_line_params con
(n_) set_scanahead_line_params_list uk
(n_) set_scanahead_params con
```

#### Allgemein nutzbare RTC6-Befehle

```
free_rtc6_dll con ..... 24
(n_) get_bios_version con ..... 24
init_rtc6_dll con ..... 25
(n_) master_slave_config con ..... 27
rtc6_count_cards con ..... 28
(n_) set_laser_power uk ..... 29
set_rtc6_mode con ..... 30
(n_) set_timelag_compensation con ..... 31
(n_) spot_distance uk ..... 32
(n_) spot_distance_ctrl con ..... 32
```

(1) con Kontrollbefehl

(2) uk unverzögerter kurzer Listenbefehl

Kontrollbefehl	<b>free_rtc6_dll</b>
Funktion	Gibt alle von der RTC6-DLL für ein Anwenderprogramm belegten Ressourcen frei.
Aufruf	<code>free_rtc6_dll()</code>
Hinweise	<ul style="list-style-type: none"> <li>• Zu den von der DLL belegten Ressourcen zählt vor allem der in der DLL belegte Speicherbereich für die Kartenverwaltung, die mit <b>init_rtc6_dll</b> aufgebaut wird. Mit <b>free_rtc6_dll</b> wird die Kartenverwaltung der DLL gelöscht. Danach ist vom Anwenderprogramm keine Karte mehr ansprechbar (<b>get_last_error</b>-Returncode: <code>RTC6_ACCESS_DENIED</code>).</li> <li>• <b>free_rtc6_dll</b> löst keinen Reset der RTC6-Karten aus. Ein Karten-Reset wird nur durch <b>load_program_file</b> initiiert.</li> <li>• Der Aufruf von <b>free_rtc6_dll</b> ist nicht zwingend erforderlich, da die von der DLL belegten Ressourcen auch automatisch freigegeben werden, wenn das Anwenderprogramm beendet wird und Microsoft Windows dabei die DLL entlädt. Einige Entwicklungsumgebungen melden jedoch im Debugmodus "Memory leaks detected!" noch ehe die DLL entladen wird. Der Aufruf von <b>free_rtc6_dll</b> beseitigt diese Irritation.</li> <li>• <b>free_rtc6_dll</b> ist nicht als Multi-Board-Befehl verfügbar.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl. Die Funktionalitäten von <b>free_rtc6_dll</b> und dem RTC5-Befehl <b>free_rtc5_dll</b> sind identisch.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600.
Verweise	<b>init_rtc6_dll</b> , <b>release_rtc</b>

Kontrollbefehl	<b>get_bios_version</b>
Funktion	Gibt die BIOS-Versionsnummer der RTC6-Karte zurück.
Aufruf	<code>BiosVersion = get_bios_version()</code>
Rückgabe	BIOS-Versionsnummer in BCD-Kodierung. Als 32-Bit-Wert ohne Vorzeichen. Beispiel: <code>BiosVersion = 33 = 0x21</code> bedeutet BIOS-Version 21.
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• <b>get_bios_version</b> liefert nur ab BIOS-Version 21 zuverlässige Ergebnisse, sonst 0.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>get_dll_version</b> , <b>get_hex_version</b> , <b>get_rtc_version</b>



Kontrollbefehl	<b>init_rtc6_dll</b>
Funktion	Initialisiert die Ansteuerung der installierten RTC6-Karten für ein Anwenderprogramm.
Aufruf	<code>InitErrorNo = init_rtc6_dll()</code>
Rückgabe	Fehlercode. Als 32-Bit-Wert ohne Vorzeichen. Sind mehrere Fehler gleichzeitig aufgetreten, dann sind mehrere Bits gesetzt. Die Zuordnung zwischen Bitnummern, Fehlertypen und Fehlerkonstanten ist identisch zu <b>get_error</b> .
Hinweise	<ul style="list-style-type: none"> <li>• <b>init_rtc6_dll</b> muss von jedem Anwenderprogramm zu Beginn aufgerufen werden, damit von dem Anwenderprogramm überhaupt eine RTC6-Karte ansprechbar ist.</li> <li>• <b>init_rtc6_dll</b> sucht nach sämtlichen vorhandenen RTC6 PCI-Express-Karten (nicht nach RTC6 Ethernet-Karten! – dazu siehe <a href="#">Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten"</a>, Seite 88) und baut die RTC6-Kartenverwaltung auf. Wenn keine RTC6 PCI-Express-Karte gefunden wird (z.B. falls ausschließlich RTC6 Ethernet-Karten betrieben werden), dann wird (analog zur RTC5: <code>RTC5_NO_CARD</code>) der Fehlercode <code>RTC6_NO_PCIE_CARD_FOUND</code> zurückgegeben (siehe <b>get_error</b> und unten, Zeile RTC5→RTC6).</li> <li>• Wenn beim Lesen des PCI-Konfigurationsregisters ein Fehler aufgetreten ist (<b>get_last_error</b>-Returncode <code>RTC6_CONFIG_ERROR</code>), wird auch der Fehler <code>RTC6_ACCESS_DENIED</code> erzeugt. Auf die Karte kann dann so lange nicht mehr zugegriffen werden, bis nach einem PC-Neustart der Fehler beseitigt ist.</li> <li>• <b>init_rtc6_dll</b> vergibt dem Anwenderprogramm automatisch (wie mit <b>acquire_rtc</b>) die Zugriffsberechtigung für die gefundenen Karten, wenn die Zugriffsberechtigung nicht bereits an ein anderes Anwenderprogramm vergeben wurde (es können zwar beliebig viele RTC6-Karten oder Anwenderprogramme gleichzeitig verwendet werden, ein und dieselbe Karte kann aber nicht gleichzeitig von mehreren Anwenderprogrammen verwendet werden). Die erste Initialisierung akquiriert alle gefundenen Karten für sich. Bei nachfolgenden, durch andere Anwenderprogramme gestarteten Initialisierungen liefert <b>init_rtc6_dll</b> dann den Fehlercode <code>RTC6_ACCESS_DENIED</code> zurück. Die Karten sind dann von diesen Anwenderprogrammen nur mit ausschließlich DLL-intern wirkenden Funktionen ansprechbar, die keine Zugriffsberechtigung erfordern – z.B. mit <b>get_error</b>, <b>get_last_error</b> oder <b>select_rtc</b> (auch die meisten Multi-Board-Befehle sind nur auf zugriffsberechtigten Karten ausführbar). Hat ein Anwenderprogramm die Zugriffsberechtigung für eine Karte erhalten, dann kann diese Karte erst dann wieder – mit <b>init_rtc6_dll</b> oder <b>acquire_rtc</b> – von einem anderen Anwenderprogramm beansprucht werden, nachdem die Karte mit <b>release_rtc</b> oder <b>free_rtc6_dll</b> explizit vom zugriffsberechtigten Anwenderprogramm freigegeben wurde. Der Fehlercode <code>RTC6_ACCESS_DENIED</code> wird zurückgegeben, wenn mindestens eine der gefundenen Karten den Zugriff verweigert hat. Welche das ist oder sind, kann mit <code>get_error(CardNo)</code> (<code>CardNo</code> von 1 bis zur Zahl der gefundenen Karten) oder unmittelbar nach <b>init_rtc6_dll</b> (noch vor dem ersten anderen Befehl) mit <code>get_last_error(CardNo)</code> abgefragt werden.</li> <li>• Beim Akquirieren einer Karte durch <b>init_rtc6_dll</b> wird (wie mit <b>acquire_rtc</b>) eine Versionsprüfung durchgeführt. Liegt ein Versionsfehler vor, wird der Zugriff auf die Karte verweigert (Rückgabewert <code>RTC6_ACCESS_DENIED</code>   <code>RTC6_VERSION_MISMATCH</code>).</li> <li>• Es kann zur gleichen Zeit immer nur ein einziges Anwenderprogramm die Initialisierung durchführen. Durch andere Anwenderprogramme später gestartete Initialisierungen warten, bis die aktuelle Initialisierung abgelaufen ist.</li> </ul>

Kontrollbefehl	init_rtc6_dll
Hinweise (Forts.)	<ul style="list-style-type: none"> <li>• Wird <b>init_rtc6_dll</b> von demselben Anwenderprogramm mehrmalig aufgerufen, so wird die beim vormaligen Aufruf des Befehls aufgebaute RTC6-Kartenverwaltung für dieses Anwenderprogramm gelöscht und die dabei an das Anwenderprogramm vergebenen Zugriffsberechtigungen aufgehoben, ehe die neue RTC6-Kartenverwaltung aufgebaut wird und wieder neue Zugriffsberechtigungen erteilt werden.</li> <li>• Bei jeder Initialisierung eines Anwenderprogramms mit <b>init_rtc6_dll</b> werden DLL-interne Nummern für alle gefundenen RTC6-Karten neu vergeben. Die Zuordnung zwischen den DLL-internen Nummern und den installierten Karten kann über <b>get_serial_number</b> ermittelt werden. Die Karte mit der kleinsten DLL-internen Nummer aller zugriffsberechtigten Karten wird bei der Initialisierung gleichzeitig zur aktiven Karte und damit zum Ziel für Nicht-Multi-Board-Befehle. Ein Wechsel der aktiven Karte ist jederzeit über <b>select_rtc</b> möglich. Besteht für keine Karte eine Zugriffsberechtigung, wird die Karte mit der höchsten DLL-internen Nummer (siehe <b>rtc6_count_cards</b>) zur aktiven Karte. Im letzteren Fall können dann aber ausschließlich rein DLL-interne Befehle ausgeführt werden, die keine Zugriffsberechtigung erfordern.</li> <li>• Beachten Sie auch im RTC5-Handbuch das Kapitel 6.7.1 "Hinweise zur Übernahme einer Karte durch ein Anwenderprogramm", Seite 96.</li> <li>• <b>init_rtc6_dll</b> ist auch ohne explizite Zugriffsberechtigung für irgendeine RTC6-Karte verfügbar.</li> <li>• <b>init_rtc6_dll</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Nach der Initialisierung der DLL mit <b>init_rtc6_dll</b> ist standardmäßig der RTC6-Modus voreingestellt. Danach kann ein anderer DLL-Betriebsmodus eingestellt werden, siehe <b>set_rtc4_mode</b> und <b>set_rtc5_mode</b>.</li> <li>• <b>init_rtc6_dll</b> löst keinen Reset der RTC6-Karte aus. Ein Karten-Reset wird nur durch <b>load_program_file</b> initiiert.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	<p>Neuer Befehl.</p> <p>Die Funktionalitäten von <b>init_rtc6_dll</b> und dem RTC5-Befehl <b>init_rtc5_dll</b> sind identisch.</p> <p>Hinweis zu <b>get_error</b>: Im RTC5-Befehlssatz bedeutet Bit #0 = 1 (Fehlerkonstante: <b>RTC5_NO_CARD</b> = 1), dass keine RTC5-Karte gefunden wurde. Im RTC6-Befehlssatz bedeutet Bit #0 = 1 (Fehlerkonstante: <b>RTC6_NO_PCIE_CARD_FOUND</b> = 1), dass keine RTC6 PCI-Express-Karte gefunden wurde (<b>init_rtc6_dll</b> sucht nicht nach RTC6 Ethernet-Karten).</p>
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600.
Verweise	<b>set_rtc4_mode</b> , <b>set_rtc5_mode</b>

Kontrollbefehl	<b>master_slave_config</b>
Funktion	Setzt Einstellungen für die Master-Slave-Schnittstelle einer RTC6-Karte.
Aufruf	<code>master_slave_config( Config )</code>
Parameter	<p>Config Als 32-Bit-Wert ohne Vorzeichen.</p> <hr/> <p>Bit #0 Unterdrückung von /Slave-START-Signalen und /Slave-STOP-Signalen.</p> <p>= 0: /Slave-START- und /Slave-STOP-Signale über die Master-Slave-Schnittstelle werden empfangen und verarbeitet.</p> <p>= 1: /Slave-START- und /Slave-STOP-Signale, die diese Karte über die Master-Slave-Schnittstelle empfängt, werden auf dieser Karte ignoriert, wohl aber an weitere Karten der Master-Slave-Schnittstelle weitergeleitet.</p> <hr/> <p>Bit #1 /STOP bei Master-Slave-Störungen.</p> <p>= 0: Die Karte reagiert nicht explizit auf eine Störung der Master-Slave-Verbindung. Etwaige Auswirkungen sind undefiniert.</p> <p>= 1: Löst ein /STOP aus, wenn die Verbindung zur Master-Karte unterbrochen wird. Dies tritt z. B. auf, wenn auf einer Karte der bereits synchronisierten Master-Slave-Kette <b>load_program_file</b> ausgeführt, das Master-Slave-Kabel entfernt oder das Signal der Master-Slave-Schnittstelle elektromagnetisch gestört wird.</p> <hr/> <p>Bit #2 Weiterleitung von /STOP bei Master-Slave-Störungen.</p> <p>= 0: Ein /STOP wird nicht an andere Karten weitergeleitet.</p> <p>= 1: Wird ein Fehler der Master-Slave-Verbindung erkannt und daraufhin ein /STOP ausgelöst (siehe Bit #1) wird dieses /STOP an alle noch verbundenen Karten der Master-Slave-Kette weitergeleitet (auch aufwärts an eine Master-Karte).</p>
Hinweise	<ul style="list-style-type: none"> <li>• Zur Verwendung von <b>master_slave_config</b>, siehe <b>Kapitel 1.9 "Master/Slave-Betrieb"</b>, Seite 52.</li> <li>• Die Master-Slave-Schnittstelle einer Karte sollte normalerweise erst nach erfolgreicher Synchronisierung konfiguriert werden.</li> <li>• Wenn Bit #0 schon vor <b>sync_slaves</b> gesetzt ist, kann diese Karte nicht mit <b>sync_slaves</b> auf eine Master-Karte synchronisiert werden.</li> <li>• In bestimmten weiteren Störungsszenarien kann es vorkommen, dass ein fehlerfreier Betrieb nicht mehr möglich ist. In diesem Fall wird immer ein /STOP auf der betroffenen Karte durchgeführt, auch wenn Bit #1 = 0 ist.</li> <li>• Wenn die Verbindung zu einer Master-Slave-synchronisierten Karte gestört ist, kann dies auch zur Folge haben, dass weitere Karten in der Kette ebenfalls gestört werden, und damit einen /STOP ausführen, auch wenn Bit #2 nicht gesetzt ist.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 609, OUT 609, RBF 613.
Verweise	<b>sync_slaves</b>

Kontrollbefehl	<b>rtc6_count_cards</b>
Funktion	Gibt die Zahl der bei der Initialisierung gefundenen RTC6-Karten zurück.
Aufruf	<code>NoOfCards = rtc6_count_cards()</code>
Rückgabe	Zahl der RTC6-Karten. Als 32-Bit-Wert ohne Vorzeichen.
Hinweise	<ul style="list-style-type: none"> <li>Die Initialisierung der installierten RTC6-Karten wird für jedes Anwenderprogramm separat mit <b>init_rtc6_dll</b> aufgerufen.</li> <li><b>rtc6_count_cards</b> ist auch ohne explizite Zugriffsberechtigung für irgendeine RTC6-Karte verfügbar.</li> <li><b>rtc6_count_cards</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>Die kartenspezifischen Fehlervariablen <code>LastError</code> und <code>AccError</code> (siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98) werden von <b>rtc6_count_cards</b> weder erzeugt noch verändert.</li> </ul>
RTC4→RTC6	Neuer Befehl. Funktionalität ist analog zu <b>rtc4_count_cards</b> aus dem RTC4-Befehlssatz.
RTC5→RTC6	Neuer Befehl. Die Funktionalitäten von <b>rtc6_count_cards</b> und dem RTC5-Befehl <b>rtc5_count_cards</b> sind identisch.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600.
Verweise	<b>init_rtc6_dll</b>

Unverzögerter kurzer Listenbefehl	<b>set_laser_power</b>
Funktion	Der Befehl synchronisiert die Ausgaben für die Laserleistung an den Ausgabe-Ports (ANALOG OUT1, ANALOG OUT2, DIGITAL OUT0, DIGITAL OUT1) und die Laser-Delays.
Aufruf	<code>set_laser_power( Port, Power )</code>
Parameter	<b>Port</b> Ausgabe-Port, für den ein Laserleistungswert <i>Power</i> definiert werden soll als 32-Bit-Wert ohne Vorzeichen. Zulässige Werte: = 0: ANALOG OUT1-Ausgang. = 1: ANALOG OUT2-Ausgang. = 2: DIGITAL OUT1-Ausgang (8-Bit-Digital-Ausgang). = 3: DIGITAL OUT0-Ausgang (16-Bit-Digital-Ausgang). Für Werte >3 wird <b>set_laser_power</b> nicht ausgeführt ( <b>get_last_error</b> -Returncode: <code>RTC6_PARAM_ERROR</code> ).
	<b>Power</b> Gewünschter Laserleistungswert als 32-Bitwert ohne Vorzeichen. Zulässige Werte: Für <i>Port</i> = 0: 12-Bit-Werte [0 ... 4095]. Für <i>Port</i> = 1: 12-Bit-Werte [0 ... 4095]. Für <i>Port</i> = 2: 8-Bit-Werte [0 ... 255]. Für <i>Port</i> = 3: 16-Bit-Werte [0 ... ( $2^{16}-1$ )]. Werte über dem zulässigen Höchstwert werden auf den zulässigen Höchstwert geclippt ( <b>get_last_error</b> -Returncode: <code>RTC6_PARAM_ERROR</code> ).
Hinweise	<ul style="list-style-type: none"> <li>Der Befehl <b>set_laser_power</b> kann sinnvoll mit Scan-Köpfen der excelliSCAN-Baureihe verwendet werden, aber auch allgemein bei einer Abfolge von kurzen Vektoren mit längeren Laser-Delays (siehe "<a href="#">Hinweis zu verändertem Zeitverhalten</a>", Seite 22).</li> <li>Der Befehl <b>set_laser_power</b> ist insbesondere dann zu verwenden, wenn mit einem excelliSCAN-Scan-Kopf innerhalb eines Polygonzuges die Laserleistung geändert werden soll.</li> <li>Der Befehl <b>set_laser_power</b> ist auch notwendig, wenn z. B. kurze Vektoren unterschiedliche Laserleistungen benötigen und die Laser-Delays über den nächsten Vektor hinweg reichen (insbesondere im DSP-Mode 3).</li> <li>Im RTC4-Kompatibilitätsmodus (siehe <b>set_rtc4_mode</b>) wird der angegebene <i>Power</i>-Wert für <i>Port</i> = 0 und <i>Port</i> = 1 von der RTC6 mit 4 multipliziert (weil bei der RTC4 die Analogausgänge nur eine Auflösung von 10 Bit haben). Der zulässige Wertebereich verringert sich entsprechend.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 602, OUT 602, RBF 602.
Verweise	<b>get_last_error</b> , <b>set_dsp_mode</b> , <b>set_rtc4_mode</b> , <b>write_da_x</b>

Kontrollbefehl	<b>set_rtc6_mode</b>
Funktion	Stellt den RTC6-Modus als aktuellen DLL-Betriebsmodus ein (Standardeinstellung).
Aufruf	<code>set_rtc6_mode()</code>
Hinweise	<ul style="list-style-type: none"> <li>• Im RTC6-Modus als DLL-Betriebsmodus müssen die Z-Koordinatenwerte und die Defocus-Werte mit einer Parameterauflösung von 20 Bit angegeben werden.</li> <li>• Im RTC6-Modus als DLL-Betriebsmodus müssen die Laser-Delay-Werte mit einer Parameterauflösung von <math>1/64 \mu\text{s}</math> angegeben werden.</li> <li>• Mit <b>set_rtc5_mode</b> kann der RTC5-Kompatibilitätsmodus als DLL-Betriebsmodus eingestellt werden. Im RTC5-Kompatibilitätsmodus müssen Z-Koordinatenwerte und Defocus-Werte (z.B. Parameter <i>Shift</i> von <b>set_defocus</b>) mit einer Parameterauflösung von 16 Bit angegeben werden. Sie werden automatisch mit 16 multipliziert. Die zulässigen Wertebereiche verringern sich entsprechend. Im RTC5-Kompatibilitätsmodus müssen die Laser-Delay-Werte (Parameter <i>LaserOnDelay</i> und <i>LaserOffDelay</i> von <b>set_laser_delays</b>) mit einer Parameterauflösung von <math>1/2 \mu\text{s}</math> angegeben werden. Sie werden automatisch mit 32 multipliziert. Die zulässigen Wertebereiche verringern sich entsprechend. Siehe auch <b>Kapitel 1.7.2 "RTC5-Quellcode für die RTC6 anpassen"</b>, Schritt <b>(3)</b>, Seite 22.</li> <li>• Mit <b>set_rtc4_mode</b> kann der RTC4-Kompatibilitätsmodus als DLL-Betriebsmodus eingestellt werden.</li> <li>• Der aktuelle DLL-Betriebsmodus kann mit <b>get_rtc_mode</b> abgefragt werden.</li> <li>• <b>set_rtc6_mode</b> ist auch ohne explizite Zugriffsberechtigung für irgendeine RTC6-Karte verfügbar.</li> <li>• <b>set_rtc6_mode</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• <b>set_rtc6_mode</b> wirkt nicht kartenspezifisch, sondern global für die DLL und damit auf alle RTC6-Karten, für die das Anwenderprogramm zugriffsberechtigt ist.</li> <li>• Die kartenspezifischen Fehlervariablen <i>LastError</i> und <i>AccError</i> (siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98) werden von <b>set_rtc6_mode</b> weder erzeugt noch verändert.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600.
Verweise	<b>get_rtc_mode</b> , <b>set_rtc4_mode</b> , <b>set_rtc5_mode</b>

Kontrollbefehl	<b>set_timelag_compensation</b>
Funktion	Gleicht die unterschiedlichen Schleppverzögerungen von XY-Achsen und Z-Achse an.
Aufruf	<code>set_timelag_compensation ( HeadNoXY, TimelagXY, TimelagZ )</code>
Parameter	<p><b>HeadNoXY</b> Nummer des XY-Kopfanschlusses. Zulässiger Wertebereich [1...2].</p> <p><b>TimelagXY</b> Angabe des Schleppverzugs der XY-Achsen in [10 µs]. Zulässiger Wertebereich [0...255].</p> <p><b>TimelagZ</b> Angabe des Schleppverzugs der Z-Achse in [10 µs]. Zulässiger Wertebereich [0...255].</p> <p>Alle Parameter als 32-Bit-Werte ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>Bei unzulässigen Parameterangaben wird der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> gesetzt und der Befehl wird nicht zur RTC6-Karte gesendet.</li> <li>Wenn eine Listenausführung aktiv ist, wird der <b>get_last_error</b>-Returncode <code>RTC6_BUSY</code> gesetzt (der Kontrollbefehl <b>get_status</b> gibt entweder BUSY oder INTERNAL-BUSY oder PAUSED zurück) und der Befehl wird nicht zur RTC6-Karte gesendet.</li> <li>Die schnelleren Achsen werden zu den langsameren hin verzögert.</li> <li>Der Befehl <b>set_timelag_compensation</b> kann auch mit Scan-Köpfen der excelliSCAN-Baureihe verwendet werden. Wurde die RTC6-Karte mittels <b>set_scanahead_params</b> für die Ansteuerung eines excelliSCAN konfiguriert, dann wird <code>TimelagXY</code> ignoriert. Statt dessen wird der Wert des <code>PreviewTime</code>-Parameters aus <b>set_scanahead_params</b> verwendet. Der Befehl wartet automatisch darauf, dass ein <code>HEAD_BUSY</code> (siehe Beschreibung im Handbuch "excelliSCAN-Scan-Köpfe – Funktionsprinzip der SCANahead-Regelung und Ansteuerung mit RTC6-Karten" beschrieben) zurückgesetzt wird, d.h. die Ausführung des Befehls kann bis <code>PreviewTime</code> dauern.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600.
Verweise	<b>set_scanahead_params, get_last_error, get_status</b>

<b>Unverzögerter kurzer Listenbefehl</b>	<b>spot_distance</b>
Funktion	Wie <b>spot_distance_ctrl</b> , aber ein unverzögerter kurzer Listenbefehl.
Aufruf	spot_distance( Dist )
Parameter	Dist                      Siehe <b>spot_distance_ctrl</b> .
Hinweise	<ul style="list-style-type: none"> <li>• Siehe <b>spot_distance_ctrl</b>.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 609, OUT 609, RBF 613.
Verweise	<b>spot_distance_ctrl</b> , <b>set_auto_laser_control</b> , <b>set_auto_laser_params</b> , <b>set_auto_laser_params_list</b>

<b>Kontrollbefehl</b>	<b>spot_distance_ctrl</b>
Funktion	Legt den geometrischen Pulsabstand für die automatische Lasersteuerung mit Ctrl = 7 fest.
Aufruf	spot_distance_ctrl( Dist )
Parameter	Dist                      Pulsabstand in Bits. Im 64-Bit-IEEE-Gleitkommaformat.
Hinweise	<ul style="list-style-type: none"> <li>• Obwohl der Pulsabstand in Bruchteilen von Bits (Bildfeldkoordinaten) angegeben werden kann, wird er nur mit einer Genauigkeit von 1/40 Bit aufgelöst.</li> <li>• Dist = 0 unterdrückt die automatische Lasersteuerung mit Ctrl = 7, aber schaltet sie nicht ab.</li> <li>• Dist darf den Maximalwert 26.214 nicht überschreiten. Ein praktischer Wert von Dist hängt von der Markiergeschwindigkeit, der Laserfrequenz und dem Scan-System ab.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 609, OUT 609, RBF 613.
Verweise	<b>spot_distance</b> , <b>set_auto_laser_control</b> , <b>set_auto_laser_params</b> , <b>set_auto_laser_params_list</b>



## 1.8.2 Geändert im RTC6-Befehlssatz

Die Befehle in diesem Abschnitt sind alphabetisch geordnet.

(n_) <b>config_list</b>	con(1)	33
<b>get_dll_version</b>	con	33
(n_) <b>get_hex_version</b>	con	33
(n_) <b>get_rtc_version</b>	con	34
(n_) <b>get_status</b>	con	34
(n_) <b>get_sync_status</b>	con	35
(n_) <b>load_correction_file</b>	con	37
(n_) <b>load_program_file</b>	con	37
(n_) <b>*load_stretch_table</b>	con	38
(n_) <b>mcbasp_init_spi</b>	con	38
(n_) <b>number_of_correction_tables</b>	con	39
(n_) <b>select_cor_table</b>	con	39
(n_) <b>*set_auto_laser_control</b>	con	40
(n_) <b>*set_auto_laser_params</b>	con	45
(n_) <b>*set_auto_laser_params_list</b>	vk (2)	45
(n_) <b>set_dsp_mode</b>	con	46
(n_) <b>*set_pixel_line</b>	nor (3)	47
(n_) <b>set_trigger4</b>	nor (3)	50

\* Hinweis: RTC6-Befehl mit Funktionalitätserweiterung.

### config\_list

- Im RTC6-Befehlssatz ist der Kontrollbefehl **config\_list** in Bezug auf Initialisierung und Wertebereich geändert. Standardmäßig erfolgt die Initialisierung pro Liste mit 4.194.304 Listenplätzen. Der gesamte Listenspeicher ist auf 8.388.608 Listenplätze erweitert.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: die "bisherigen" Werte bleiben gültig (die neuen Möglichkeiten werden nur nicht ausgeschöpft).

### get\_dll\_version

- Die RTC6 DLL-Versionsnummern liegen im Bereich 600...699. Diese sind in der RTC6 DLL-Datei `RTC6DLL.dll` hinterlegt.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Im RTC6-Befehlssatz liefert der Kontrollbefehl **get\_dll\_version** Werte im Bereich 600 ... 699 zurück.

### get\_hex\_version

- Die RTC6 DSP-Programmdatei-Versionsnummern liegen im Bereich 600...699. Diese sind in der DSP-Programmdatei `RTC6OUT.out` hinterlegt.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Im RTC6-Befehlssatz liefert der Kontrollbefehl **get\_hex\_version** Werte zurück in den folgenden Bereichen:
  - ohne freigeschaltete 3D-Option 2.600...2.699 (Versionsnummer + 2.000)
  - mit freigeschalteter 3D-Option 3.600...3.699 (Versionsnummer + 3.000)

(1) con Kontrollbefehl

(2) vk verzögerter kurzer Listenbefehl

(3) nor normaler Listenbefehl

#### **get\_rtc\_version**

- Die RTC6 Firmware-Versionsnummern liegen im Bereich 600...699. Diese sind in der RTC6 Firmware-Datei `RTC6RBF.rbf` hinterlegt.
- Beim Rückgabewert bedeutet:  
 Bit #12 = 1: Die Option "SCANa" ist freigeschaltet (ab DLL 605).  
 Bit #13 = 1: Die Option "UFPM" ist freigeschaltet (ab DLL 605).  
 Bit #14 = 1: Die Option "syncA" ist freigeschaltet (ab DLL 607).
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Im RTC6-Befehlssatz liefert der Kontrollbefehl `get_rtc_version` (Bit#0...7) Werte im Bereich 0...99 (Versionsnummer – 600) zurück.

#### **get\_status**

- Im RTC6-Befehlssatz ist der Kontrollbefehl **get\_status** erweitert. Wie im RTC5-Befehlssatz liefert der Parameter `Status` den `BUSY`-, `INTERNAL-BUSY`- und `PAUSED` Status. Bei der RTC6 DLL-Datei `RTC6DLL.dll` kann die Parameterrückgabe zusätzlich auch den Status `HEAD_BUSY` über Bit #23 zurückliefern. Allerdings wird dieser Statuswert nur zurückgeliefert, wenn die RTC6-Karte mittels **set\_scanahead\_params** für die Ansteuerung eines excelliSCAN-Scan-Kopfs konfiguriert wurde (er muss nicht angeschlossen sein). excelliSCAN-Scan-Köpfe können nach Beendigung einer Liste mindestens noch um die `PreviewTime` (Vorausrechnungszeit) mit der Ausgabe beschäftigt sein).
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Nicht relevant, da mit RTC5-Karten keine excelliSCAN-Scan-Köpfe angesteuert werden können.

Kontrollbefehl	<b>get_sync_status</b>																		
Funktion	Liefert Master/Slave-Synchronisierungsinformationen zu der angesprochenen RTC6-Karte zurück.																		
Aufruf	<code>MasterSlaveSyncStatus = get_sync_status()</code>																		
Rückgabe	<p>Master/Slave-Synchronisierungsinformationen. Als 32-Bit-Wert ohne Vorzeichen.</p> <table border="1"> <tr> <td>Bit #0</td><td>Master/Slave-Synchronisierungszustand [0...640].</td></tr> <tr> <td>...</td><td>Als 10-Bit-Wert ohne Vorzeichen.</td></tr> <tr> <td>Bit #9</td><td> <p>&lt; 3: Die angesprochene Karte ist synchron zur Master-Karte (oder ihrer Vorgängerkarte in der Master/Slave-Kette).</p> <p>640: Die angesprochene Karte wird als Master-Karte betrieben.</p> </td></tr> <tr> <td>Bit #10</td><td> <p>= 0: Bit #0...Bit #9 enthalten gültige Werte.</p> <p>= 1: Bit #0...Bit #9 enthalten keine gültigen Werte. Es wurde noch kein externer Start von der Master-Karte empfangen.</p> </td></tr> <tr> <td>Bit #11</td><td>Reserviert.</td></tr> <tr> <td>Bit #12</td><td>= 1: Es wurde eine Master-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b>.</td></tr> <tr> <td>Bit #13</td><td>= 1: Es wurde eine Slave-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b>.</td></tr> <tr> <td>Bit #14</td><td>= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Master abgebrochen.</td></tr> <tr> <td>Bit #15</td><td>= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Slave abgebrochen.</td></tr> </table>	Bit #0	Master/Slave-Synchronisierungszustand [0...640].	...	Als 10-Bit-Wert ohne Vorzeichen.	Bit #9	<p>&lt; 3: Die angesprochene Karte ist synchron zur Master-Karte (oder ihrer Vorgängerkarte in der Master/Slave-Kette).</p> <p>640: Die angesprochene Karte wird als Master-Karte betrieben.</p>	Bit #10	<p>= 0: Bit #0...Bit #9 enthalten gültige Werte.</p> <p>= 1: Bit #0...Bit #9 enthalten keine gültigen Werte. Es wurde noch kein externer Start von der Master-Karte empfangen.</p>	Bit #11	Reserviert.	Bit #12	= 1: Es wurde eine Master-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b> .	Bit #13	= 1: Es wurde eine Slave-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b> .	Bit #14	= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Master abgebrochen.	Bit #15	= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Slave abgebrochen.
Bit #0	Master/Slave-Synchronisierungszustand [0...640].																		
...	Als 10-Bit-Wert ohne Vorzeichen.																		
Bit #9	<p>&lt; 3: Die angesprochene Karte ist synchron zur Master-Karte (oder ihrer Vorgängerkarte in der Master/Slave-Kette).</p> <p>640: Die angesprochene Karte wird als Master-Karte betrieben.</p>																		
Bit #10	<p>= 0: Bit #0...Bit #9 enthalten gültige Werte.</p> <p>= 1: Bit #0...Bit #9 enthalten keine gültigen Werte. Es wurde noch kein externer Start von der Master-Karte empfangen.</p>																		
Bit #11	Reserviert.																		
Bit #12	= 1: Es wurde eine Master-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b> .																		
Bit #13	= 1: Es wurde eine Slave-Karte erkannt. Jedoch kann kein /Slave-START oder /Slave-STOP empfangen werden. Siehe <b>master_slave_config</b> .																		
Bit #14	= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Master abgebrochen.																		
Bit #15	= 1: Seit dem letzten Aufruf von <b>get_sync_status</b> ist die Verbindung zum Slave abgebrochen.																		
Hinweise	<ul style="list-style-type: none"> <li>• Zur Verwendung von <b>get_sync_status</b>, siehe <b>Kapitel 1.9 "Master/Slave-Betrieb"</b>, <b>Seite 52</b>.</li> <li>• Falls die angesprochene RTC6-Karte nicht zuvor mit <b>load_program_file</b> initialisiert wurde, liefert <b>get_sync_status</b> 0 zurück.</li> <li>• Die Karten der Master/Slave-Kette sollten vor <b>get_sync_status</b> mit <b>sync_slaves</b> synchronisiert worden sein.</li> <li>• Damit <b>get_sync_status</b> tatsächlich den aktuellen Master/Slave-Synchronisierungszustand zurückliefert, muss zuvor auf der Master-Karte (per externem Start-Signal oder per Befehl) ein externer Listen-Start ausgelöst worden sein, siehe RTC5-Handbuch, Abschnitt "Externer Listen-Start", Seite 240. Dieser Listen-Start löst bei allen Slave-Karten, bei denen externe Starts und Stopps nicht unterdrückt sind (siehe <b>master_slave_config</b>) eine Messung des Zeitunterschieds zwischen dem jeweiligen /Slave-START-Puls und dem jeweiligen 10 <math>\mu</math>s-Takt aus. Dieser Zeitunterschied wird als Master/Slave-Synchronisierungszustand (in Einheiten von 1/64 <math>\mu</math>s) auf der jeweiligen Karte gespeichert, bleibt dort gespeichert, bis an der Master-Karte ein neuer externer Listen-Start ausgelöst wird und kann daher auch noch zu einem späteren Zeitpunkt mit <b>get_sync_status</b> ausgelesen werden.</li> </ul>																		

Kontrollbefehl	get_sync_status
Hinweise (Forts.)	<ul style="list-style-type: none"> <li>• Im synchronisierten Zustand ist der gemessene Zeitunterschied kleiner als der Laufzeitunterschied bei der Synchronisierung zwischen den Karten selbst (siehe <b>Kapitel 1.9 "Master/Slave-Betrieb"</b>, Seite 52): Master/Slave-Synchronisierungszustand &lt; 3.</li> <li>• Im nicht explizit synchronisierten Zustand (vor <b>sync_slaves</b>) kann der Synchronisierungszustand dennoch zufällig &lt; 3 sein. Die Karten verhalten sich dann wie synchronisierte Karten.</li> <li>• Ist Bit #12 oder Bit #13 gesetzt, können die betroffenen Karten nicht mit <b>sync_slaves</b> synchronisiert werden.</li> <li>• Ist Bit #14 oder Bit #15 gesetzt, sind /Slave-STARTs oder /Slave-STOPS evtl. nicht weitergeleitet worden. In diesem Fall wird empfohlen, <b>master_slave_config</b> anzupassen und <b>sync_slaves</b> erneut auszuführen.</li> <li>• Wenn eines oder mehrere der Bit #12...Bit #15 sporadisch 1 sind, kann dies auf elektromagnetische Störungen hinweisen, die die Kommunikation zwischen den Karten beeinträchtigt.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Unveränderte Grundfunktionalität. Beim RTC5-Befehl stehen jedoch weniger Rückgaben (und damit weniger Details zu Master/Slave-Synchronisierungsinformationen) zur Verfügung.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600. Letzte Änderung mit Version DLL 609, OUT 609, RBF 613: interne Verbesserungen, zusätzliche Rückgaben.
Verweise	<b>sync_slaves</b> , <b>get_master_slave</b> , <b>master_slave_config</b>

## load\_correction\_file

- Im RTC6-Befehlssatz können mit dem Kontrollbefehl **load\_correction\_file** vier Korrekturfiles gleichzeitig in den Speicher der RTC6 geladen werden. Es gibt keine Überschneidung mit von **set\_trigger** benutzten Speicherbereichen mehr.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: —.

## load\_program\_file

- Im RTC6-Befehlssatz führt der Kontrollbefehl **load\_program\_file** die gleichen Funktionen aus wie im RTC5-Befehlssatz. Es werden u.a. die Dateien `RTC6OUT.out`, `RTC6RBF.rbf`, `RTC6DAT.dat` geladen.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen:  
**load\_program\_file** ist abwärtskompatibel zur RTC5 und kann unverändert weiter benutzt werden. Ihr Anwenderprogramm muss nicht geändert werden. Allerdings ist zu beachten, dass:
  - Die Prüfung auf Status BUSY entfällt (d.h. es kann niemals der Fehlercode 13 ausgegeben werden).
  - Die Datenübertragung an der McBSP-Schnittstelle ohne Warnung oder Fehlermeldung deaktiviert wird.
  - Im RTC6-Befehlssatz eine laufende Liste wie mit dem Kontrollbefehl **stop\_execution**, siehe RTC5-Handbuch, Seite 596, ohne Warnung oder Fehlermeldung beendet wird. Beachten Sie dort auch die Hinweise z.B. zur Spiegelstellung.
  - andere Fehlercodes bereitgestellt werden, siehe folgende Tabelle.

### Wert Beschreibung

0	Erfolg.
1	Entfällt.
2	BOARD NOT RUNNING. Falls bei wiederholtem Aufruf nicht erfolgreich, ist ein PowerCycle erforderlich.
3	RTC6DAT.dat-Datei oder RTC6RBF.rbf-Datei nicht gefunden.
4	Entfällt.

### Wert Beschreibung

5	Nicht genügend WINDOWS-Speicher.
6	Access-Fehler: die Karte ist für anderes Anwenderprogramm reserviert.
7	Versions-Fehler: DLL-Version (der <code>RTC6DLL.dll</code> ), RTC-Version (der Firmwaredatei <code>RTC6RBF.rbf</code> ) und HEX-Version (der DSP-Programmdatei <code>RTC6OUT.out</code> ) passen nicht zusammen.
8	Systemtreiber nicht gefunden.
9	Das Laden der <code>RTC6OUT.out</code> -Datei schlug fehl oder <code>RTC6OUT.out</code> hat falsches Format oder sonstiger Fehler.
10	Entfällt.
11	Firmware-Fehler: Das Laden der <code>RTC6RBF.rbf</code> -Datei schlug fehl.
12	Fehler beim Öffnen/Lesen von Dateien ( <code>RTC6DAT.dat</code> ).
13	Entfällt.
14	DSP-Memory Fehler.
15	Verify-Memory Fehler.
16	Entfällt.
17	Ethernet-Fehler. Siehe <b>eth_get_last_error</b> und <b>eth_get_error</b> .

### load\_stretch\_table

- Im RTC6-Befehlssatz ist bei dem Kontrollbefehl **load\_stretch\_table** die Funktionalität erweitert. Im Vergleich zum RTC5-Befehl gibt es den zusätzlichen Parameter `TableNo` als 32-bit Wert ohne Vorzeichen. Der zulässige Wertebereich ist 1...4. Beispielaufruf: `NoOfDataPairs = load_stretch_table( Name, No, TableNo)`. `Name` und `No` werden wie bisher interpretiert. `TableNo` bedeutet die zugeordnete Korrekturtabelle (mit **select\_cor\_table** wird diese Korrektur eingeschaltet).  
Im RTC5-Kartenspeicher kann es nur eine Stretch-Datei oder Zoom-Table-Datei geben. Diese wird dann global verwendet. Mit der RTC6-Karte kann jeder der vier Korrekturtabellen eine eigene Stretch-Datei und/oder Zoom-Table-Datei zugeordnet werden. Damit können Benutzer z.B. leichter Vergleiche mit und ohne Korrektur oder mit zwei unterschiedlichen Korrekturen bewerkstelligen, indem einfach nur die Korrekturtabelle umgeschaltet wird.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: —.

### mcbbsp\_init\_spi

- Im RTC6-Befehlssatz ist der Kontrollbefehl **mcbbsp\_init\_spi** enthalten, jedoch nicht wirksam. Er wird mit **get\_last\_error**-Returncode `RTC6_REJECTED` verweigert und nicht ausgeführt.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Bei RTC6-Karten kann Hardware-bedingt keine SPI-Funktionalität mehr eingestellt werden. Solche Quellcodeabschnitte müssen entsprechend angepasst werden.

Kontrollbefehl	<b>number_of_correction_tables</b>
Funktion	Definiert die maximale Anzahl von zulässigen Korrekturtabellen.
Aufruf	<code>number_of_correction_tables( Number )</code>
Parameter	<p>Number      Anzahl der maximal zulässigen Korrekturtabellen.  Als 32-Bit-Wert ohne Vorzeichen.  Zulässiger Wertebereich: [1...8].  Default nach <b>load_program_file</b>: 4.</p>
Hinweise	<ul style="list-style-type: none"> <li>• Für <b>Number</b> außerhalb des zulässigen Wertebereichs wird <b>number_of_correction_tables</b> ignoriert (<b>get_last_error</b>-Returncode: <code>RTC6_PARAM_ERROR</code>).</li> <li>• <b>number_of_correction_tables</b> dient dazu, andere Befehle (wie z. B. <b>load_correction_file</b> und <b>select_cor_table</b>) vor unerwünschten Tabellennummern zu schützen.</li> <li>• Bestehende Anwenderprogramme müssen nicht geändert werden. Die Ausnahme ist, wenn künftig Benutzer-Eingaben (mittels expliziter RTC6-Fehlermeldungen) zurückgewiesen werden sollen.</li> <li>• <b>number_of_correction_tables</b> bezieht sich nur auf nachfolgende Befehlsausführungen. Bereits bestehende Zuweisungen von Korrekturtabellen können nicht automatisch korrigiert werden. Das ist insbesondere dann zu beachten, wenn RTC6-Karten nachträglich in Besitz genommen werden, die von anderen Anwenderprogrammen initialisiert wurden.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Geänderte Auswirkungen (der RTC5-Befehl wurde aufgrund des begrenzten Nutzerkreises nicht dokumentiert).
Versionsinfo	Verfügbar ab Version DLL 609, OUT 609, RBF 613.
Verweise	<b>load_correction_file</b> , <b>select_cor_table</b> , <b>select_cor_table_list</b>

#### select\_cor\_table

- Im RTC6-Befehlssatz schaltet der Kontrollbefehl **select\_cor\_table** Zoom und/oder Stretch automatisch ein (d.h. die erweiterte 3D-Korrektur ist aktiv), wenn vorher eine Tabelle mit **load\_stretch\_table** oder **load\_zoom\_correction\_file** geladen worden ist. Im RTC5-Kartenspeicher kann es nur eine Stretch-Datei oder Zoom Table-Datei geben. Diese wird dann global verwendet.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: —.

Kontrollbefehl	set_auto_laser_control	
Funktion	Initialisiert bzw. deaktiviert die positions- und/oder geschwindigkeits- oder encodergeschwindigkeitsabhängige Lasersteuerung.	
Aufruf	ErrorCode = set_auto_laser_control( Ctrl1, Value, Mode, MinValue, MaxValue )	
Parameter	Ctrl1	<p>Kontrollparameter zur Initialisierung bzw. Deaktivierung der positions- und/oder geschwindigkeits- oder encodergeschwindigkeitsabhängigen Lasersteuerung. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>= 1...7: Definiert, welcher Signalparameter durch die positions- und/oder geschwindigkeits- oder encodergeschwindigkeitsabhängige Lasersteuerung automatisch korrigiert wird.</p> <ul style="list-style-type: none"> <li>= 1: 12-Bit-Ausgabewert am ANALOG OUT1-Ausgang.</li> <li>= 2: 12-Bit-Ausgabewert am ANALOG OUT2-Ausgang.</li> <li>= 3: Ausgabewert am 8-Bit-Digital-Ausgang.</li> <li>= 4: Pulslänge (PulseLength) der Lasersignale LASER1 und LASER2.</li> <li>= 5: Ausgabeperiode (HalfPeriod) der Lasersignale LASER1 und LASER2.</li> <li>= 6: Ausgabewert am 16-Bit-Digital-Ausgang.</li> <li>= 7: Nur für excelliSCAN-Scan-Köpfe. Wie 5. Jedoch ist der Pulsabstand geometrisch konstant ("Spot Distance Control").</li> </ul> <p>= 0 oder &gt; 7: deaktiviert die positions-, geschwindigkeits- und encodergeschwindigkeitsabhängige Lasersteuerung (für Ctrl1 &gt; 7: <b>get_last_error-Returncode</b> RTC6_PARAM_ERROR).</p>
	Value	<p>Legt den 100%-Wert für den über Ctrl1 ausgewählten Parameter fest. Als 32-Bit-Wert ohne Vorzeichen. Zulässige Werte:</p> <ul style="list-style-type: none"> <li>Für Ctrl1 = 1/2: 12-Bit-Werte [0...4095]. Höherwertige Bits werden ignoriert.</li> <li>Für Ctrl1 = 3: 8-Bit-Werte [0...255]. Höherwertige Bits werden ignoriert.</li> <li>Für Ctrl1 = 4: [0...(2<sup>32</sup>-1)]. 1 Bit entspricht 1/64 µs.</li> <li>Für Ctrl1 = 5: [0...(2<sup>32</sup>-1)]. 1 Bit entspricht 1/64 µs.</li> <li>Für Ctrl1 = 6: 16-Bit-Werte [0...(2<sup>16</sup>-1)]. Höherwertige Bits werden ignoriert.</li> </ul>
	Mode	<p>Kontrollparameter zur Initialisierung bzw. Deaktivierung der geschwindigkeits- oder encodergeschwindigkeitsabhängigen (nicht aber der positionsabhängigen) Lasersteuerung. Als 32-Bit-Wert ohne Vorzeichen. Für die geschwindigkeitsabhängige Lasersteuerung (Mode = 1...5) kann ausgewählt werden, welche Größe bei der Berechnung der geschwindigkeitsabhängigen Korrektur zu Grunde gelegt wird.</p> <ul style="list-style-type: none"> <li>=1: Für alle Scan-Systeme. Sollgeschwindigkeit.</li> <li>=2: Nur für intelli-Scan-Köpfe. Rückgelesene aktuelle Geschwindigkeit.</li> <li>=3: Reserviert.</li> <li>=4: Reserviert.</li> <li>=5: Für alle Scan-Systeme. Encodergeschwindigkeit (Zählerpulsrate).</li> </ul>



Kontrollbefehl		set_auto_laser_control
Parameter (Forts.)	Mode (Forts.)	<p>=6: <i>Nur für intelli-Scan-Köpfe.</i> Kombinierte Geschwindigkeit aus 2 und 5.</p> <p>Weitere Modi können mit Mode = 18, 22, 34, 38, 50, 54 eingestellt werden. Hierfür gilt (mit M = 2 oder M = 6):</p> <ul style="list-style-type: none"> <li>• Mode = M + 0: Nur für intelliSCAN-Scan-Köpfe</li> <li>• Mode = M + 16: Nur für excelliSCAN-Scan-Köpfe</li> <li>• Mode = M + 32: Die Winkel-synchrone Galvanometerscanner-Geschwindigkeit wird mit Hilfe des zugeordneten Korrekturfiles in eine Bildfeld-synchrone Geschwindigkeit umgerechnet. (Mode = M + 32 + (0 oder 16): siehe oben).</li> </ul> <p>=18: <i>Nur für excelliSCAN-Scan-Köpfe mit Firmware-Version ≥ 5052. Nicht für intelli-Scan-Köpfe.</i> Es wird die rückgelesene und um PreviewTime korrigierte Geschwindigkeit verwendet.</p> <p>=22: <i>Nur für excelliSCAN-Scan-Köpfe mit Firmware-Version ≥ 5052. Nicht für intelli-Scan-Köpfe.</i> Kombinierte Geschwindigkeit aus 18 und 5.</p> <p>=34: <i>Nur für intelli-Scan-Köpfe. Nicht für excelliSCAN-Scan-Köpfe.</i> Wie Mode = 2. Zusätzlich wird die Winkel-synchrone Galvanometerscanner-Geschwindigkeit mit Hilfe des zugeordneten Korrekturfiles in eine Millimeter-synchrone Geschwindigkeit umgerechnet.</p> <p>=38: <i>Nur für intelli-Scan-Köpfe. Nicht für excelliSCAN-Scan-Köpfe.</i> Wie Mode = 6. Zusätzlich wird die Winkel-synchrone Galvanometerscanner-Geschwindigkeit mit Hilfe des zugeordneten Korrekturfiles in eine Millimeter-synchrone Geschwindigkeit umgerechnet.</p> <p>=50: <i>Nur für excelliSCAN-Scan-Köpfe mit Firmware-Version ≥ 5052. Nicht für intelli-Scan-Köpfe.</i> Siehe Mode = 34.</p> <p>=54: <i>Nur für excelliSCAN-Scan-Köpfe mit Firmware-Version ≥ 5052. Nicht für intelli-Scan-Köpfe.</i> Siehe Mode = 38.</p> <p>Für Mode = 0 oder (mit Ausnahme der oben genannten) &gt; 6: Deaktiviert die geschwindigkeitsabhängige und encodergeschwindigkeitsabhängige Lasersteuerung (für – mit Ausnahme der oben genannten – Mode &gt; 6: <b>get_last_error</b>-Returncode: RTC6_PARAM_ERROR).</p>
	MinValue	Legt die <i>untere</i> Bereichsgrenze für den über Ctrl1 ausgewählten Parameter fest, der durch die automatische Korrektur nicht unterschritten werden kann. Als 32-Bit-Wert ohne Vorzeichen. Zulässige Werte: siehe Value.
	MaxValue	Legt die <i>obere</i> Bereichsgrenze für den über Ctrl1 ausgewählten Parameter fest, der durch die automatische Korrektur nicht überschritten werden kann. Als 32-Bit-Wert ohne Vorzeichen. Zulässige Werte: siehe Value.

Kontrollbefehl	set_auto_laser_control														
Rückgabe	<p>ErrorCode. Als 32-Bit-Wert ohne Vorzeichen.</p> <table> <tr> <td>0</td><td>Kein Fehler.</td></tr> <tr> <td>1</td><td>Kein primärer Scan-Kopf aktiv.</td></tr> <tr> <td>2</td><td>Kein iDRIVE-Scan-System (intelliSCAN, intelliSCAN<sub>de</sub>, intelliDRILL, intellicube, intelliWELD, varioSCAN<sub>de</sub>) aktiv.</td></tr> <tr> <td>3</td><td>Ctrl unzulässig.</td></tr> <tr> <td>4</td><td>Mode unzulässig.</td></tr> <tr> <td>5</td><td>Zugriff verweigert (Karte gesperrt, <b>get_last_error</b>-Returncode RTC6_ACCESS_DENIED).</td></tr> <tr> <td>6</td><td>Nur für Mode = 18 und Mode = 22: Kein excelliSCAN-Scan-Kopf angeschlossen oder aktiv.</td></tr> </table>	0	Kein Fehler.	1	Kein primärer Scan-Kopf aktiv.	2	Kein iDRIVE-Scan-System (intelliSCAN, intelliSCAN <sub>de</sub> , intelliDRILL, intellicube, intelliWELD, varioSCAN <sub>de</sub> ) aktiv.	3	Ctrl unzulässig.	4	Mode unzulässig.	5	Zugriff verweigert (Karte gesperrt, <b>get_last_error</b> -Returncode RTC6_ACCESS_DENIED).	6	Nur für Mode = 18 und Mode = 22: Kein excelliSCAN-Scan-Kopf angeschlossen oder aktiv.
0	Kein Fehler.														
1	Kein primärer Scan-Kopf aktiv.														
2	Kein iDRIVE-Scan-System (intelliSCAN, intelliSCAN <sub>de</sub> , intelliDRILL, intellicube, intelliWELD, varioSCAN <sub>de</sub> ) aktiv.														
3	Ctrl unzulässig.														
4	Mode unzulässig.														
5	Zugriff verweigert (Karte gesperrt, <b>get_last_error</b> -Returncode RTC6_ACCESS_DENIED).														
6	Nur für Mode = 18 und Mode = 22: Kein excelliSCAN-Scan-Kopf angeschlossen oder aktiv.														
Hinweise	<ul style="list-style-type: none"> <li>• Zur Verwendung von <b>set_auto_laser_control</b> siehe RTC5-Handbuch, Kapitel 7.4.9 "Automatische Lasersteuerung", Seite 159.</li> <li>• MinValue und MaxValue werden automatisch vertauscht, sofern MinValue &gt; MaxValue ist und gegebenenfalls MinValue auf ≤ Value sowie MaxValue auf ≥ Value geclippt.</li> <li>• Die Steuerungsdaten der positions- und/oder geschwindigkeitsabhängigen Lasersteuerung werden ausschließlich aus den Daten des Scan-Systems am primären Scan-Kopf-Anschluss abgeleitet und dann auch für den sekundären Scan-Kopf-Anschluss verwendet (sofern dieser freigeschaltet ist und dort ein Scan-System angeschlossen ist). Dem primären Scan-Kopf-Anschluss muss bereits zum Zeitpunkt der Befehlserteilung eine Korrekturabelle zugewiesen sein, sonst wird Fehlercode 1 zurückgegeben und Ctrl = 0 gesetzt. Wenn nur der sekundäre Scan-Kopf-Anschluss verwendet wird und/oder das Scan-System am primären Scan-Kopf-Anschluss abgeschaltet ist, funktioniert die geschwindigkeitsabhängige Lasersteuerung nicht.</li> <li>• Für Mode = 2 muss am primären Scan-Kopf-Anschluss ein funktionsfähiges iDRIVE-Scan-System angeschlossen sein. Ist dies nicht der Fall, wird Fehlercode 2 zurückgegeben und Ctrl = 0 gesetzt. Wenn das iDRIVE-Scan-System reagiert hat, wird der rückzulesende Datentyp für den primären Scan-Kopf-Anschluss und beide Achsen per <b>control_command</b> (Data = 0506<sub>H</sub>) auf Ist-Geschwindigkeit gestellt. Erst danach wird der Befehl zur RTC6 geschickt, wobei die anderen Parameter übernommen werden. In der Folge ist der <b>control_command</b>-Befehl für den primären Scan-Kopf-Anschluss gesperrt, solange diese Mode-Auswahl besteht. Für Mode = 0 oder 1 ist der <b>control_command</b>-Befehl uneingeschränkt verfügbar.</li> <li>• Die encodergeschwindigkeitsabhängige Lasersteuerung (Mode = 5) funktioniert auch dann, wenn zur Laufzeit kein Scan-Kopf an den Scan-Kopf-Anschlüssen angeschlossen ist. Auch die "Processing on the fly"-Option muss hierfür nicht freigeschaltet sein.</li> </ul>														

Kontrollbefehl	set_auto_laser_control
Hinweise (Forts.)	<ul style="list-style-type: none"> <li>• Mode = 6 ist grundsätzlich wie Mode = 2. Jedoch werden die Encoder-Geschwindigkeiten mit den Skalierungsfaktoren aus <b>set_fly_x</b> und <b>set_fly_y</b> bzw. <b>set_fly_2d</b> in Galvanometerscanner-Einheiten (Bits/ms) umgerechnet und zu den aktuellen Galvanometerscanner-Geschwindigkeiten hinzuaddiert. Die Option "Processing on the fly" muss dazu freigeschaltet sein (i. Ggs. zu Mode = 5). Als Referenzgeschwindigkeit wird wie bei Mode = 2 die aktuelle Markiergeschwindigkeit verwendet. Der für Mode = 5 verwendete Befehl <b>set_encoder_speed</b> bleibt im Mode = 6 unberücksichtigt. Ist zur Laufzeit eine Achse nicht für "Processing on the fly" aktiviert, so bleibt deren Encodergeschwindigkeit unberücksichtigt. Sind beide Achsen nicht für "Processing on the fly" aktiviert, wirkt Mode = 6 wie Mode = 2. <b>set_fly_rot</b>, <b>set_fly_rot_pos</b>, <b>set_fly_x_pos</b> und <b>set_fly_y_pos</b> können nicht mit der Galvanometerscanner-Geschwindigkeit kombiniert werden.</li> <li>• Für Ctrl = 1...6 wird der ausgewählte Signalparameter alle 10 <math>\mu</math>s aktualisiert, für Ctrl = 5 (Ausgabeperiode) jedoch immer erst nach Ablauf einer bereits begonnenen Lasersignal-Periode wirksam. Für Ctrl = 7 wird die Ausgabeperiode dagegen kontinuierlich (von Puls zu Puls) angepasst.</li> <li>• Ctrl = 7 ("Spot Distance Control") ist nur für excelliSCAN-Scan-Köpfe verfügbar. Damit Ctrl = 7 tatsächlich wirken kann, muss mit <b>spot_distance</b> oder <b>spot_distance_ctrl</b> der gewünschte (geometrische) Pulsabstand angegeben werden. Solange der Pulsabstand 0 ist (Defaultwert) ist diese Regelung unwirksam, aber nicht deaktiviert. Die tatsächliche Regelgröße ("HalfPeriod") kann nicht mit <b>set_trigger/set_trigger4</b> und Signal 24 aufgezeichnet werden. Die Befehlsparameter MinValue, MaxValue und Value werden ignoriert. Deshalb kann die tatsächliche Regelgröße ("HalfPeriod") nicht mit MinValue und MaxValue begrenzt werden. Es wird automatisch auf die aktuelle Markiergeschwindigkeit geregelt. Siehe auch Kapitel 1.11 "Hinweise zu set_auto_laser_control mit Ctrl = 7 ("Spot Distance Control")", Seite 60.</li> <li>• Ein Aktivieren der Winkel-zu-Bildfeld-Transformation der Galvanometerscanner-Geschwindigkeit (Mode = 34, 38, 50, 54) ist insbesondere dann sinnvoll, wenn das Korrekturfile starke Verzeichnungen ausgleichen muss.</li> <li>• Für einen unzulässigen Wert von Ctrl oder Mode wird <b>set_auto_laser_control</b> nicht ausgeführt (Rückgabewert 3 bzw. 4, <b>get_last_error</b>-Returncode: RTC6_PARAM_ERROR).</li> <li>• Die automatische Lasersteuerung sollte nicht zusammen mit der variablen Laserleistung aus <b>set_multi_mcbp_in</b> verwendet werden.</li> <li>• Nur für excelliSCAN-Scan-Köpfe: Wenn ein excelliSCAN-Scan-Kopf angeschlossen und aktiv ist, stellen Sie vor dem Aufruf von <b>set_auto_laser_control</b> sicher, dass der Scan-Kopf nicht mehr beschäftigt ist (siehe <b>get_status</b> und Zustand HEAD_BUSY). Anderenfalls kann der noch nicht verarbeitete Rest der letzten Markierung gestört werden.</li> </ul>
RTC4→RTC6	<p>Neuer Befehl.</p> <p>Im RTC4-Kompatibilitätsmodus werden für Ctrl = 1/2 die Werte für Value, MinValue und MaxValue mit 4 multipliziert. Für Ctrl = 4/5 müssen im RTC4-Kompatibilitätsmodus die Werte für Value, MinValue und MaxValue in 1/8 <math>\mu</math>s angegeben werden. Sie werden dann intern auf 1/64 <math>\mu</math>s-Einheiten umgerechnet (d.h. mit 8 multipliziert). Die zulässigen Wertebereiche verringern sich entsprechend.</p>

Kontrollbefehl	set_auto_laser_control
RTC5→RTC6	<p>Veränderte Funktionalität.</p> <p>Im RTC6-Befehlssatz (ab DLL 602, OUT 602) können mit <b>set_auto_laser_control</b> weitere Lasersteuerungsmodi eingestellt werden. Dazu können jetzt für den Parameter <code>Mode</code> weitere Parameterwerte übergeben werden (siehe oben, ab RTC6-Software-Paket <math>\geq</math> V1.4.1 auch <code>Mode &gt; 6</code>). Für <code>Mode = 0</code> oder unzulässige Werte <code>Mode &gt; 6</code> wird die geschwindigkeitsabhängige und/oder encodergeschwindigkeitsabhängige Lasersteuerung deaktiviert und der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> erzeugt.</p>
Versionsinfo	<p>Verfügbar ab Version DLL 600, OUT 600, RBF 600.</p> <p>Letzte geänderte Version DLL 609, OUT 609, RBF 613: <code>Ctrl = 7</code>.</p>
Verweise	<p><b>load_position_control, load_auto_laser_control, set_auto_laser_params, set_auto_laser_params_list, set_fly_x, set_fly_y, set_fly_2d, set_encoder_speed, set_fly_rot, set_fly_rot_pos, set_fly_x_pos, set_fly_y_pos, set_multi_mcbp_in, spot_distance, spot_distance_ctrl</b></p>

Kontrollbefehl	<b>set_auto_laser_params</b>
Funktion	Legt den zu steuernden Signalparameter der automatischen Lasersteuerung, den 100%-Wert und seine ihm zugeordneten Grenzwerte fest.
Aufruf	<code>set_auto_laser_params( Ctrl, Value, MinValue, MaxValue)</code>
Parameter	Ctrl Der zu steuernde Signalparameter. Siehe <b>set_auto_laser_control</b> . Zulässiger Wertebereich: [1...7].
	Value 100%-Wert. Siehe <b>set_auto_laser_control</b> .
	MinValue Untere Bereichsgrenze. Siehe <b>set_auto_laser_control</b> .
	MaxValue Obere Bereichsgrenze. Siehe <b>set_auto_laser_control</b> .
Hinweise	<ul style="list-style-type: none"> <li>• Zur Verwendung von <b>set_auto_laser_params</b> siehe RTC5-Handbuch, Kapitel 7.4.9 "Automatische Lasersteuerung", Seite 159.</li> <li>• Die Bedeutung der Parameter ist identisch zu <b>set_auto_laser_control</b> (siehe auch Hinweise dort). Jedoch kann die automatische Lasersteuerung mit <b>set_auto_laser_params</b> weder aus- noch eingeschaltet werden.</li> <li>• Wenn der Wert von Ctrl unzulässig ist (0 oder &gt; 7), dann wird <b>set_auto_laser_params</b> nicht ausgeführt (<b>get_last_error</b>-Returncode: RTC6_PARAM_ERROR).</li> <li>• Ctrl = 7 ist nur für excelliSCAN-Scan-Köpfe.</li> </ul>
RTC4→RTC6	Neuer Befehl. RTC4-Kompatibilitätsmodus: siehe <b>set_auto_laser_control</b> .
RTC5→RTC6	Erweiterte Funktionalität ab Version DLL 609, OUT 609, RBF 613: Ctrl = 7.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600. Letzte geänderte Version DLL 609, OUT 609, RBF 613: Ctrl = 7.
Verweise	<b>set_auto_laser_control</b> , <b>set_auto_laser_params_list</b>

Verzögerter kurzer Listenbefehl	<b>set_auto_laser_params_list</b>
Funktion	Wie <b>set_auto_laser_params</b> , aber ein verzögerter kurzer Listenbefehl.
Aufruf	<code>set_auto_laser_params_list( Ctrl, Value, MinValue, MaxValue)</code>
Parameter	Ctrl Siehe <b>set_auto_laser_params</b> .
	Value Siehe <b>set_auto_laser_params</b> .
	MinValue Siehe <b>set_auto_laser_params</b> .
	MaxValue Siehe <b>set_auto_laser_params</b> .
Hinweise	<ul style="list-style-type: none"> <li>• Wenn der Wert von Ctrl unzulässig ist (0 oder &gt; 7), dann wird <b>set_auto_laser_params_list</b> durch ein <b>list_nop</b> ersetzt (<b>get_last_error</b>-Returncode RTC6_PARAM_ERROR).</li> <li>• Ctrl = 7 ist nur für excelliSCAN-Scan-Köpfe.</li> </ul>
RTC4→RTC6	Neuer Befehl. RTC4-Kompatibilitätsmodus: siehe <b>set_auto_laser_control</b> .
RTC5→RTC6	Erweiterte Funktionalität ab Version DLL 609, OUT 609, RBF 613: Ctrl = 7.
Versionsinfo	Verfügbar ab Version DLL 600, OUT 600, RBF 600. Letzte geänderte Version DLL 609, OUT 609, RBF 613: Ctrl = 7.
Verweise	<b>set_auto_laser_params</b> , <b>set_auto_laser_control</b>

### set\_dsp\_mode

- Im RTC6-Befehlssatz wird beim Kontrollbefehl **set\_dsp\_mode** der DSP-Modus 3 eingeführt. RTC6-Karten werden standardmäßig<sup>(1)</sup> im DSP-Modus 3 betrieben. *Kurze Vektoren* werden damit schneller abgearbeitet. Mit dem Modus 0 und Modus 2 kann für *kurze Vektoren* eine Kompatibilität zu RTC5-Karten hergestellt werden<sup>(2)</sup>. Modus 0 entspricht RTC5-Karten älterer Baureihen mit 500 MHz und Modus 2 RTC5-Karten mit 720 MHz. Mit den beiden Modi werden die schnelleren Zeitabläufe der RTC6 bei *kurzen Vektoren* an die langsameren der RTC5 angepasst<sup>(3)</sup>.
- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: Es ist eine inhaltliche Benutzerentscheidung, ob `set_dsp_mode( 2 )` oder `set_dsp_mode( 3 )` gewünscht ist.

(1) D.h. der Befehl `set_dsp_mode( 3 )` muss dazu nicht explizit im Quellcode stehen.

(2) Modus 1 ist reserviert.

(3) Im RTC5-Handbuch ist im Abschnitt "Automatische Delay-Anpassung", Seite 121 beschrieben, wie die "künstlichen" Scanner-Delays eingefügt werden.

Normaler Listenbefehl	set_pixel_line	
Funktion	Aktiviert den gewünschten Pixelausgabemodus und definiert verschiedene Pixelausgabeparameter.	
Aufruf	set_pixel_line( Channel, HalfPeriod, dX, dY )	
Parameter	Channel	<p>Definiert den Pixelausgabemodus sowie die Ausgabeports, an denen die Pixelinhalte ausgegeben werden sollen. Die Pixelinhalte an sich werden in <b>set_pixel</b> und <b>set_n_pixel</b>-Befehlen festgelegt. Diese müssen <b>set_pixel_line</b> unmittelbar nachfolgen.</p> <p>Es gilt: Channel = Modus + Port.</p> <p>Als 32-Bit-Wert ohne Vorzeichen.</p> <p>Zulässig sind:</p> <p>Modus = 0: Ausgaben an den Parameter <code>PulseLength</code> von <b>set_pixel</b> und <b>set_n_pixel</b> (bis 32 Bit) und Port. Die Galvanometerscannerbewegung ist nicht kontinuierlich, siehe <a href="#">Abschnitt "RTC5→RTC6", Seite 49</a>.</p> <p>Modus = 16: Zwei Ausgaben an Port. Je ein 32-Bit-Pixel pro Ausgabe.</p> <p>Modus = 32: Zwei Ausgaben an Port. Je zwei 16-Bit-Pixel pro Ausgabe.</p> <p>Modus = 64: Zwei Ausgaben an Port. Je vier 8-Bit-Pixel pro Ausgabe.</p> <p>Modus = 256: Wie Modus = 0, jedoch ist die Galvanometerscannerbewegung kontinuierlich, siehe <a href="#">Abschnitt "RTC5→RTC6", Seite 49</a>.</p> <p>Port = 1: 12-Bit-Analog-Ausgang 1 (LASER-Buchse). Siehe Hinweis, <a href="#">Seite 48</a>.</p> <p>Port = 2: 12-Bit-Analog-Ausgang 2 (LASER-Buchse. Bei der RTC6 PCI-Express-Karte auch die MARKING ON THE FLY-Stiftleiste). Siehe Hinweis, <a href="#">Seite 48</a>.</p> <p>Port = 3: 8-Bit-Digital-Ausgang (EXTENSION 2-Stiftleiste).</p> <p>Port = 4: 16-Bit-Digital-Ausgang (EXTENSION 1-Stiftleiste).</p> <p>Port = 5: Parameter <code>PulseLength</code> von <b>set_pixel</b> und <b>set_n_pixel</b>. Nicht erlaubt mit Modus = 0 und Modus = 256.</p>
	HalfPeriod	<p>Halbe Pixelausgabeperiode in <i>Bits</i>. 1 Bit entspricht 1/64 <math>\mu</math>s.</p> <p>Als 32-Bit-Wert ohne Vorzeichen.</p> <p>Zulässiger Bereich: [Min...(2<sup>32</sup>-1)] mit:</p> <p>Min = 80 für Modus = 0 und 256. Maximalfrequenz: 0,4 MHz.</p> <p>Min = 40 für Modus = 16. Maximalfrequenz: 0,8 MHz.</p> <p>Min = 20 für Modus = 32. Maximalfrequenz: 1,2 MHz. Für RTC6-Karten <i>ohne</i> Option 'UFPM' gilt: der Wert wird bei 40 geclippt.</p> <p>Min = 10 für Modus = 64. Maximalfrequenz: 3,2 MHz. Für RTC6-Karten <i>ohne</i> Option 'UFPM' gilt: der Wert wird bei 40 geclippt.</p>
	dX, dY	<p>Abstand in x- und y-Richtung zwischen zwei aufeinanderfolgenden Pixeln in <i>Bits</i>.</p> <p>Im 64-Bit-IEEE-Gleitkommaformat.</p>

Normaler Listeneintrag	set_pixel_line
Hinweise	<ul style="list-style-type: none"> <li>• Jede Bildzeile eines Pixelbilds muss mit dem Befehl <b>set_pixel_line</b> begonnen werden. Dem Befehl sollte jeweils ein Jump-Befehl oder Mark-Befehl auf den jeweiligen Startpunkt der Bildzeile vorausgehen.</li> <li>• Unmittelbar nach dem Befehl <b>set_pixel_line</b> muss die gewünschte Anzahl von <b>set_pixel</b> und <b>set_n_pixel</b>-Befehlen folgen. Diese übertragen die Parameter <i>PulseLength</i> und <i>AnalogOut</i>. Deren Bedeutung hängt von Modus und Port ab (siehe oben, Parameter <i>Channel</i>).</li> <li>• Der erste Befehl einer Liste, der auf den Befehl <b>set_pixel_line</b> folgt und <i>kein</i> <b>set_pixel</b> oder <b>set_n_pixel</b>-Befehl ist, beendet den Pixelausgabemodus. Auch der Befehl <b>set_pixel_line</b> beendet den Pixelausgabemodus, bevor er ihn wieder startet. Dabei wird ein Default-Pixel eingefügt.</li> <li>• Das Default-Pixel für Port 1...4 wird durch <b>set_port_default</b> festgelegt. Aber Achtung: die Port-Nummern bei <b>set_port_default</b> und <b>set_pixel_line</b> stimmen nicht überein! Das Default-Pixel für Port 5 (<i>PulseLength</i>) wird durch <b>set_default_pixel</b> festgelegt.</li> <li>• Bei unzulässigen <i>Channel</i>-Werten (Beispiel: 5, für Mode = 0 und Port = 5) wird <b>set_pixel_line</b> durch <b>list_nop</b> ersetzt (<i>get_last_error</i>-Returncode <i>RTC6_PARAM_ERROR</i>). Der Pixelausgabemodus wird dann <i>nicht</i> aktiviert.</li> <li>• Beachten Sie, dass unter <i>HalfPeriod</i> die <i>halbe</i> Periodendauer angegeben werden muss. <math>2 \times \text{HalfPeriod}</math> ist also der zeitliche Abstand zwischen den einzelnen Pixeln, siehe RTC5-Handbuch, Kapitel 8.8 "Pixelausgabemodus – Rasterbilder (Bitmaps) scannen", Seite 217. <i>HalfPeriod</i> darf nicht kleiner als <i>Min</i> sein (wird andernfalls automatisch geclippt).</li> <li>• Bei Ausgaben an den 12-Bit-Analog-Ausgang 1 (Port = 1) oder 12-Bit-Analog-Ausgang 2 (Port = 2) ist zu beachten: die DA-Wandlung wird für Pixelfrequenzen nur bis etwa 100 kHz (d.h. für <i>HalfPeriod</i> &lt; ca. 320) in jedem Fall vollständig abgeschlossen. Bei Pixelfrequenzen in diesem Bereich müssen Benutzer die ausreichende Funktionsfähigkeit sorgfältig überprüfen. Bei Pixelfrequenzen &gt; 100 kHz muss zwingend die UFPM-Erweiterungskarte, siehe <b>Kapitel 3 "Anhang B: Die UFPM-Erweiterungskarte"</b>, Seite 124, verwendet werden.</li> <li>• Siehe auch <b>Kapitel 1.10 "Ergänzende Informationen zu den RTC6-Pixel-Modi"</b>, Seite 55.</li> <li>• Der Pixelausgabemodus kann mit "Processing on the fly" kombiniert werden. Siehe auch RTC5-Handbuch, Kapitel 8.7 "Processing on the fly" (optional)", Seite 199.</li> <li>• Der Pixelausgabemodus sollte nicht zusammen mit der automatischen Lasersteuerung (siehe RTC5-Handbuch, Kapitel 7.4.9 "Automatische Lasersteuerung", Seite 159) verwendet werden, wenn dort der Port-Ausgang, die Pulslänge (<i>PulseLength</i>) oder Ausgabeperiode (<i>HalfPeriod</i>) der Lasersignale LASER1 und LASER2 angepasst werden.</li> <li>• Der Pixelausgabemodus ist inkompatibel mit dem Softstart-Modus. Siehe auch RTC5-Handbuch, Kapitel 7.4.5 "Softstart-Modus", Seite 152.</li> <li>• Der Pixelausgabemodus kann <i>nicht</i> mit Sky-Writing kombiniert werden. Siehe auch RTC5-Handbuch, Kapitel 7.2.4 "Sky-Writing", Seite 127.</li> <li>• Der Pixelausgabemodus kann <i>nicht</i> mit Wobbel kombiniert werden. Siehe auch RTC5-Handbuch, Kapitel 8.4 "Wobbelmode", Seite 189.</li> <li>• Siehe auch RTC5-Handbuch, Kapitel 8.8 "Pixelausgabemodus – Rasterbilder (Bitmaps) scannen", Seite 217.</li> </ul>



Normaler Listenbefehl	set_pixel_line
RTC4→RTC6	<ul style="list-style-type: none"> <li>Die RTC4 gibt im Pixelausgabemodus Analogspannungswerte nur am ANALOG OUT2-Ausgang aus. Ausgaben an die digitalen Ports sind nicht möglich.</li> <li>Die Pixelausgabeperiode wird bei der RTC6 als <i>halbe</i> Ausgabeperiode <code>HalfPeriod</code> angegeben (bei der RTC4 dagegen als <i>volle</i> Ausgabeperiode <code>PixelPeriod</code>).</li> <li>Der Befehl <b>set_pixel_line</b> belegt bei der RTC6 wie andere normale Listenbefehle nur <i>einen</i> Listenspeicherplatz (bei der RTC4 dagegen <i>zwei</i> Speicherplätze).</li> <li>Im RTC4-Kompatibilitätsmodus muss <code>HalfPeriod</code> in <math>1/8 \mu s</math> angegeben werden. Die Parameter-Werte werden dann intern auf <math>1/64 \mu s</math>-Einheiten umgerechnet (d.h. mit 8 multipliziert). Der zulässige Wertebereich verringert sich entsprechend. Die angegebenen Koordinatenwerte <code>dx</code>, <code>dy</code> werden im RTC4-Kompatibilitätsmodus von der RTC6 mit 16 multipliziert. Der zulässige Wertebereich verringert sich entsprechend.</li> </ul>
RTC5→RTC6	<p>Mit <math>\leq</math> DLL 603 ist der Pixelausgabe-Modus noch identisch zu dem im RTC5-Befehlssatz. Es sind nur Pixelfrequenzen von bis zu 308 kHz ("Classic Mode") möglich.</p> <p>Mit <math>\geq</math> DLL 604 ist die Pixelausgabe-Funktionalität bedeutend erweitert. Die <code>Channel</code>-Werte 1 (Modus = 0 an ANALOG OUT1-Ausgang) und 2 (Modus = 0 an ANALOG OUT2-Ausgang) haben noch die gleiche Wirkung wie im RTC5-Befehlssatz, erlauben jedoch eine Maximalfrequenz bis 400 kHz. Im Modus = 0 bleiben die Galvanometerscanner unmittelbar vor und nach der Pixelbildzeile stehen. Benutzer sollten die Beschleunigungsstrecke ausblenden. Dazu ist das LaserOn-Delay auf einen geeignet hohen Wert einzustellen.</p> <p>Im Modus <math>&gt; 0</math> bewegen sich die Galvanometerscanner stets mit der durch <code>dx</code>, <code>dy</code> angegebenen Geschwindigkeit. Daher können Benutzer zusätzlich Vor- und Nachlaufbewegungen im Anwenderprogramm einfügen, um gleichmäßige Pixelbildzeilen zu erzeugen (ohne den LaserOn-Delay-Wert anpassen zu müssen).</p>
Versionsinfo	Letzte Änderung mit DLL 604, OUT 604, RBF 609.
Verweise	<b>set_default_pixel</b> , <b>set_default_pixel_list</b> , <b>set_pixel</b> , <b>set_n_pixel</b> , <b>set_pixel_line_3d</b> , <b>set_port_default</b>

#### **set\_trigger4**

- Im RTC6-Befehlssatz können mit dem Normalen Listenbefehl **set\_trigger4** mehr Werte (32-bit Datenworte) als mit der RTC5 gespeichert werden. Das sind bei Verwendung von

- 4 Kanälen: je Kanal 8.388.608  
(RTC5: 524.288)
- 2 Kanälen: je Kanal 2×8.388.608  
(RTC5: 1.048.576)
- 1 Kanal: 4×8.388.608  
(RTC5 je Kanal 1.048.576)  
Datenworte.

Wie bei der RTC5-Karte endet die Aufzeichnung danach automatisch.

- Hinweise z.B. zur Migration von Quellcode von RTC5-Anwenderprogrammen: —.

### 1.8.3 Nicht mehr vorhanden im RTC6-Befehlssatz

Die Befehle in diesem Abschnitt sind alphabetisch geordnet.

<b>free_rtc5_dll</b> .....	51
<b>init_rtc5_dll</b> .....	51
<b>rtc5_count_cards</b> .....	51

Kontrollbefehl	<b>free_rtc5_dll</b>
Unterstützung	Dieser RTC5-Befehl wird nicht von der RTC6 unterstützt.
Ersatz-Befehl	<b>free_rtc6_dll</b>

Kontrollbefehl	<b>init_rtc5_dll</b>
Unterstützung	Dieser RTC5-Befehl wird nicht von der RTC6 unterstützt.
Ersatz-Befehl	<b>init_rtc6_dll</b>

Kontrollbefehl	<b>rtc5_count_cards</b>
Unterstützung	Dieser RTC5-Befehl wird nicht von der RTC6 unterstützt.
Ersatz-Befehl	<b>rtc6_count_cards</b>

## 1.9 Master/Slave-Betrieb

Sollen mehrere RTC6-Karten mit synchroner Taktung in einem PC betrieben werden, dann müssen die RTC6-Karten zuerst paarweise über die Master- und Slave-Anschlüsse miteinander verbunden und dann in benachbarte PCIe-Steckplätze gesteckt werden. Verbinden Sie jeweils den Master-Anschluss einer Karte mit dem Slave-Anschluss einer anderen Karte. Geeignete Verbindungskabel sind bei SCANLAB erhältlich.

In einer Master/Slave-Kette ist automatisch diejenige RTC6-Karte die Master-Karte, an deren Master-Anschluss eine weitere und an deren Slave-Anschluss keine weitere RTC6-Karte angeschlossen ist. Alle anderen RTC6-Karten sind Slave-Karten.

Mit `get_master_slave` kann für jede RTC6-Karte der Master/Slave-Status abgefragt werden, d.h. ob die Karte als Master-, Slave- oder Einzelkarte betrieben wird.

Für ein Quellcodebeispiel auf eine Prüfung, welche RTC6-Karte Master, und welche RTC6-Karte Slave ist, siehe [Abschnitt "Beispiel-Code", Seite 54](#).

### Initialisierung

Auf allen RTC6-Karten einer Master/Slave-Kette müssen die Befehle `load_program_file` und `load_correction_file` erteilt worden sein. Die synchrone Taktung mit stabiler Phasenlage der Master/Slave-Kette wird durch die erste nicht-initialisierte Karte unterbrochen. Eine RTC6-Karte, die zwar mit `load_program_file` initialisiert wurde, aber als Slave an eine Karte angeschlossen ist, die nicht mit `load_program_file` initialisiert wurde, unterliegt ihrem eigenen Takt mit zufälliger Phasenlage.

### Synchronisierung der Takt-Phasen

Damit die RTC6-Karten einer Master/Slave-Kette nicht nur mit synchroner Taktung, sondern auch mit einer definierten relativen Takt-Phase betrieben werden können, müssen die Karten mit `sync_slaves` synchronisiert werden.

Dazu muss `sync_slaves` ein einziges Mal nur an die Master-Karte geschickt werden. SCANLAB empfiehlt, die Synchronisation unmittelbar nach der Initialisierung aller Karten (`load_program_file` und `load_correction_file`) durchzuführen. Es genügt, `load_correction_file(0,1,2)` aufzurufen oder vorübergehend alle Scan-Köpfe abzustecken.

Nach der Synchronisierung ist die Takt-Phase jeder Karte um  $0 \dots 2 \frac{1}{64} \mu\text{s}$ -Takte (= etwa 0 ns...32 ns) gegenüber derjenigen der vorgeschalteten Karte verzögert (RTC5: 160 ns).

Ohne die Synchronisierung kann die Verzögerung bis zu 10  $\mu\text{s}$  betragen. Mit `get_sync_status` kann überprüft werden, ob eine Slave-Karte synchron zur Master-Karte (bzw. ihrer Vorgängerkarte in der Master/Slave-Kette) ist.

### Hinweise

- Encoder-Signale werden nicht von der Masterkarte an die Slave-Karte(n) weitergeleitet. Sie müssen jeweils separat in die Slave-Karte(n) eingespeist werden. Dabei sollte auch die Takt-Phasen-Verschiebung von 0 ns...32 ns berücksichtigt werden.
- Wenn eine Karte in einer synchronisierten Master/Slave-Kette separat durch eine Zyklus-Synchronisierung extern getaktet wird, so bezieht sich diese Taktung ausschließlich auf diese Karte. Alle anderen Karten in der Master/Slave-Kette werden weiterhin mit dem ursprünglichen Takt der Master-Karte synchronisiert. Wenn die Zyklus-Synchronisation wieder deaktiviert wird, bleibt die betroffene Karte asynchron. Sie kann nur durch nochmaliges Aufrufen von `sync_slaves` wieder synchronisiert werden.
- Erst nach erfolgter Synchronisierung (empfohlen) können die Eigenschaften der Master/Slave-Schnittstelle mit `master_slave_config` konfiguriert werden.

## Synchrone Starts und Stopps

In einer Master/Slave-Kette werden externe Listen-Starts (sofern mit **set\_control\_mode** freigegeben) und externe Listen-Stopps von einer Karte zu allen nachgeschalteten Slave-Karten *und*<sup>(1)</sup> der vorgeschalteten Master-Karte durchgereicht.

Daher kann ein gleichzeitiger Listen-Start aller synchronisierter Karten einer Master/Slave-Kette (mit vorgebbaren Streckenverzögerungen) durch ein externes Start-Signal, ein **simulate\_ext\_start** oder **simulate\_ext\_start\_ctrl** an einer Karte<sup>(2)</sup> ausgelöst werden; und durch ein externes Stopp-Signal oder ein **simulate\_ext\_stop** entsprechend ein gleichzeitiger Listen-Stopp.

Listen-Starts mit **execute\_list** oder **execute\_at\_pointer**, sowie Listen-Stopps mit **stop\_execution** werden dagegen nicht durchgereicht und müssen auch an Master/Slave-synchronisierten Karten separat ausgeführt werden.

Zu weiteren Informationen siehe RTC5-Handbuch, Kapitel 9.3.1 "Ausführen und Beenden von Listen mit externen Steuersignalen und Master/Slave-Synchronisation", Seite 239.

(1) RTC5: nicht an die vorgeschaltete Master-Karte.

(2) RTC5: nur an der Master-Karte.

## Beispiel-Code

Der folgende Delphi-Quellcode zeigt beispielhaft, wie man prüft, welche RTC6-Karte Master, und welche Slave ist.

Der Code muss in ein Anwenderprogramm eingebunden werden (siehe RTC5-Handbuch, Kapitel 6.2.5 "Beispiel-Code", Seite 71).

```

if init_rtc6_dll() <> 0 then halt; // Initialize the RTC6 DLL
if rtc6_count_cards() <> 2 then halt; // Are 2 RTC6 boards in the PC?
for CardNo := 1 to 2 do // Load firmware and 3D correction files onto both boards
begin
    // Stop RTC6 if a task is currently still running. Optional:
    // n_stop_execution(CardNo);
    if n_load_program_file(CardNo, nil) <> 0 then halt;
    if n_load_correction_file(CardNo, nil, 1, 2) <> 0 then halt;
end;
// Check for Errors (bits #12-#15)
if (n_get_sync_status(1) and $F000 = 0) and (n_get_sync_status(2) and $F000 = 0) then
begin
    // Detect master board
    // Is board 1 the master and board 2 a single slave?
    if (n_get_master_slave(1) = 2) and (n_get_master_slave(2) = 1) then
    begin
        Master := 1;
        Slave := 2;
    end else
    // Is board 2 the master and board 1 a single slave?
    if (n_get_master_slave(1) = 1) and (n_get_master_slave(2) = 2) then
    begin
        Master := 2;
        Slave := 1;
    end
    else
        halt; // Something wrong with master-slave configuration
    end else
        halt;

    n_select_cor_table(Master, 1, 0);
    n_select_cor_table(Slave, 1, 0);
    n_set_control_mode(Master, 1 + 8); // Master slave, activate Start Stop control
    n_set_control_mode(Slave, 1 + 8); // For external control, you must use the master
    // board's control inputs
    n_sync_slaves(Master); // Synchronize master and slave boards
    // check snychronization status at any time
    // provide an external /START or a simulated external start before checking
    n_simulate_ext_start_ctrl(Master);
    Result = n_get_sync_status(Master) and $3FF; // must be 640
    Result = n_get_sync_status(Slave) and $3FF; // must be <3

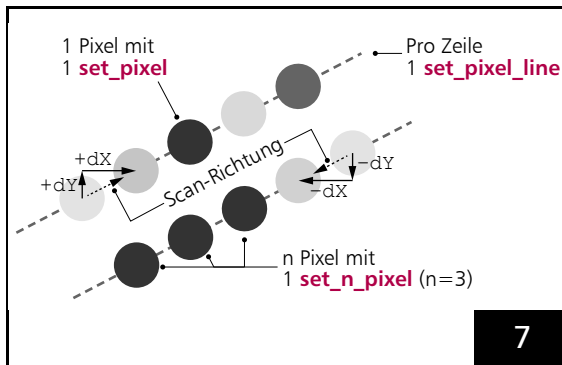
    // MasterCfg, SlaveCfg must be defined elsewhere, see manual
    n_master_slave_config(Master, MasterCfg);
    n_master_slave_config(Slave, SlaveCfg);

```

## 1.10 Ergänzende Informationen zu den RTC6-Pixel-Modi

Im RTC6-Befehlssatz ist **set\_pixel\_line** im Vergleich zum gleichnamigen RTC5-Befehl erweitert worden: Es können nun Pixelfrequenzen bis 3,2 MHz (RTC5: bis 308 kHz) erzielt werden (für Frequenzen > 800 kHz ist eine RTC6-Karte mit "UFPM" Option erforderlich).

Der grundsätzliche Ablauf zum Markieren von Pixel-Bildern mit dem Pixel-Modus ist im Vergleich zur RTC5 unverändert, siehe **Abbildung 7**.



Für jede Bildzeile wird ein eigenes **set\_pixel\_line** benötigt. Über nachfolgende **set\_pixel/set\_n\_pixel** werden dann die einzelnen Pixel dieser Zeile definiert.

- Mit **set\_pixel\_line** (siehe Befehls-Tabelle) wird eine einzelne Zeile konfiguriert:
  - der **räumliche** Abstand zwischen den einzelnen Pixeln mit den Parametern  $\Delta x$ ,  $\Delta y$  (wahlweise **set\_pixel\_line\_3d** mit zusätzlichem Parameter  $\Delta z$ ; wie bei RTC5)
  - der **zeitliche** Abstand zwischen den einzelnen Pixeln mit dem Parameter **HalfPeriod** (wie bei RTC5)
  - der Parameter **Channel** ist gegenüber der RTC5 erweitert zu **Channel = Modus + Port**, wobei nun der (Pixel-)Modus über den Wert **Modus** und der Ausgabe-Port über den Wert **Port** definiert wird, siehe **Tabelle 1**
- Erweitert gegenüber der RTC5: Abhängig vom eingestellten Modus haben die Parameter von **set\_pixel/set\_n\_pixel** im RTC6-Befehlssatz unterschiedliche Bedeutungen, siehe **Tabelle 2**.

### Hinweise

- **set\_n\_pixel** ist synonym zu **set\_pixel...set\_pixel** (n-mal wiederholt). Die Pixelzeilen sind dadurch automatisch (je nach Mode) auf 1/2/4/8 Pixel ausgerichtet (aligned). Ggf. müssen Padd-Pixel eingefügt werden.

Tabelle 1

Ausgabe-Port	Wo an der RTC6 PCI-Express-Karte	
1	12-Bit-Analog-Ausgang 1	LASER-Buchse, ANALOG OUT1 Siehe RTC5-Handbuch, Kapitel 4.4.1, Abbildung 10.
2	12-Bit-Analog-Ausgang 2	LASER-Buchse, ANALOG OUT2 Siehe RTC5-Handbuch, Kapitel 4.4.1, Abbildung 10. MARKING ON THE FLY-Stiftleiste, ANALOG OUT2 Siehe RTC5-Handbuch, Kapitel 4.4.4, Abbildung 18.
3	8-Bit-Digital-Ausgang	EXTENSION 2-Stiftleiste Siehe RTC5-Handbuch, Kapitel 4.4.3, Abbildung 17.
4	16-Bit-Digital-Ausgang	EXTENSION 1-Stiftleiste Siehe RTC5-Handbuch, Kapitel 4.4.2, Abbildung 15.
5 <sup>(a)</sup>	Pulslänge	LASER-Buchse, LASER1 Siehe RTC5-Handbuch, Kapitel 4.4.1, Abbildung 10.

(a) Port = 5 kann nicht mit Modus = 0 und nicht mit Modus = 256 kombiniert werden.

Tabelle 2

Modus	Max. Frequenz	Anzahl Pixel pro set_pixel/set_n_pixel	Erster Parameter von set_pixel/set_n_pixel (Name: PortOutValue1 <sup>(a)</sup> )	Zweiter Parameter von set_pixel/set_n_pixel (Name: PortOutValue2 <sup>(b)</sup> )
0, 256 <sup>(c)</sup>	400 kHz	1	Pixel 1 (definiert durch die Pulslänge). Die Ausgabe erfolgt als Signal LASER1.	Port-Ausgabe (aber nicht Port 5) für Pixel 1
			<div>Pixel 1</div> <div>(erster Parameter: 32-Bit; zweiter Parameter: bis 16 Bit, je nach Port)</div>	
16	800 kHz	2	Port-Ausgabe für Pixel 1 (bis zu 32 Bit pro Pixel, je nach Port)	Port-Ausgabe für Pixel 2 (bis zu 32 Bit pro Pixel, je nach Port)
			<div>Pixel 1</div>	<div>Pixel 2</div>
32 <sup>(d)</sup>	1,6 MHz	4	Port-Ausgabe für Pixel 1, 2 (bis zu 16 Bit pro Pixel, je nach Port)	Port-Ausgabe für Pixel 3, 4 (bis zu 16 Bit pro Pixel, je nach Port)
			<div>Pixel 2</div> <div>Pixel 1</div>	<div>Pixel 4</div> <div>Pixel 3</div>
64 <sup>(d)</sup>	3,2 MHz	8	Port-Ausgabe für Pixel 1, 2, 3, 4 (8 Bit pro Pixel, für alle Ports)	Port-Ausgabe für Pixel 5, 6, 7, 8 (8 Bit pro Pixel, für alle Ports)
			<div>Pixel 4</div> <div>Pixel 3</div> <div>Pixel 2</div> <div>Pixel 1</div>	<div>Pixel 8</div> <div>Pixel 7</div> <div>Pixel 6</div> <div>Pixel 5</div>
			Bit 31.....Bit 0	Bit 31.....Bit 0

(a) PulseLength in RTC6-Software-Paketen < V1.3.3.

(b) AnalogOut in RTC6-Software-Paketen < V1.3.3.

(c) Die Galvanometerscannerbewegung ist im Modus = 0 nicht kontinuierlich, im Modus = 256 kontinuierlich.

(d) Nur verfügbar mit RTC6-Karten, die mit der Option "UFPM" ausgestattet sind.



## Quellcode-Beispiel

Der Code muss in ein Anwenderprogramm eingebunden werden.

```
// UINT = uint32_t
// LONG = int32_t

// laser parameters
UINT LaserMode = 0;
UINT LaserControl = 0x0;
LONG LaserOnDelay = (LONG)round(100.0*64.0); // LaserOnDelay = 100 us
UINT LaserOffDelay = (UINT)round(100.0*64.0); // LaserOffDelay = 100 us
UINT HalfPeriod = (UINT)round(5.0 * 32.0); // Period = 5 us, PixelFrequency = 200 kHz
UINT PulseLength = (UINT)round(0.5 * 64.0); // PulseLength = 0.5 us

// pixel mode variables
enum E_PixelModes
{
    STANDARD = 0, // as RTC5-Mode; up to 400 kHz; 1 pixel per set_pixel command; 2 values of 32 bit per pixel
    ENHANCED = 16, // up to 800 kHz; 2 pixel per set_pixel command; 1 value of 32 bit per pixel
    FAST = 32, // up to 1.6 MHz; 4 pixel per set_pixel command; 1 value of 16 bit per pixel
    ULTRAFAST = 64, // up to 3.2 MHz; 8 pixel per set_pixel command; 1 value of 8 bit per pixel
    STANDARD_MOVE = 256 // as STANDARD, but with continuous scanner movement
} Mode;

enum E_PixelPorts
{
    NO_OUT_PORT = 0,
    ANALOG_OUT1 = 1, // laser connector ANALOG OUT1
    ANALOG_OUT2 = 2, // laser connector ANALOG OUT2
    DIGITAL_8Bit = 3, // EXTENSION 2 socket connector (8-bit)
    DIGITAL_16Bit = 4, // EXTENSION 1 socket connector (16-bit)
    PULSE_LENGTH = 5 // not allowed for mode = 0 or mode = 256
} Port;

Port = DIGITAL_8Bit; // pixel output port number
Mode = STANDARD; // pixel mode
UINT Channel = Port + Mode; // pixel channel

UINT Number = 1; // set_pixel repetition number

UINT PixelArray[17] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 }; // port output values
UINT PortOutValue1 = PixelArray[0];
UINT PortOutValue2 = PixelArray[0];

LONG dx = (LONG)round(0.01 * CalibrationFactorXY); // 10 um spot distance in x direction
LONG dy = (LONG)round(0.0 * CalibrationFactorXY); // 0 um spot distance in y direction

LONG PixelLineStartPosX = (LONG)round(0.0 * CalibrationFactorXY);
LONG PixelLineStartPosY = (LONG)round(0.0 * CalibrationFactorXY);

// basic configuration
set_laser_mode(LaserMode);
set_laser_control(LaserControl);

// RTC-List
set_start_list_pos(1, 0);
set_laser_delays(LaserOnDelay, LaserOffDelay);
set_default_pixel_list(0); // sets pixel default PulseLength
UINT PortDefault = Port - 1; // CAUTION: port numbers differ between set_port_default[_list] and
set_pixel_line
set_port_default_list(PortDefault, PixelArray[0]); // sets default value for specified port
```

```

switch (Mode)
{
case STANDARD:
case STANDARD_MOVE:
{
    // standard pixel mode up to 400 kHz (Mode = 0, 256)
    // 2 values of 32 bit per pixel
    // 1 pixel per set_pixel command
    //-----
    // |           pixel1           |
    // | 32 Bit | 32 Bit |
    // | PulseLength | PortOutValue |
    //-----
    HalfPeriod = (UINT)round(5.0 * 32.0); // Period = 5 us, PixelFrequency = 200 kHz
    PulseLength = (UINT)round(0.5 * 64.0); // PulseLength = 0.5 us

    jump_abs(PixelLineStartPosX, PixelLineStartPosY);
    set_pixel_line(Channel, HalfPeriod, dx, dy);
    set_n_pixel(1 * PulseLength, PixelArray[1], Number); // pixel 1
    set_n_pixel(2 * PulseLength, PixelArray[2], Number); // pixel 2
    set_n_pixel(3 * PulseLength, PixelArray[3], Number); // pixel 3
    set_n_pixel(4 * PulseLength, PixelArray[4], Number * 2); // (pixel 4)*2
}
break;

case ENHANCED:
{
    // enhanced pixel mode up to 800 kHz (Mode = 16)
    // 1 value of 32 bit per pixel
    // 2 pixel per set_pixel command
    //-----
    // | pixel1 | pixel2 |
    // | 32 Bit | 32 Bit |
    // | PortOutValue1 | PortOutValue2 |
    //-----
    HalfPeriod = (UINT)round(2.0 * 32.0); // Period = 2 us, PixelFrequency = 500 kHz
    PulseLength = (UINT)round(0.5 * 64.0); // PulseLength = 0.5 us
    set_laser_pulses(HalfPeriod, PulseLength);

    jump_abs(PixelLineStartPosX, PixelLineStartPosY);
    set_pixel_line(Channel, HalfPeriod, dx, dy);
    set_n_pixel(PixelArray[1], PixelArray[2], Number); // pixel 1, pixel 2
    set_n_pixel(PixelArray[3], PixelArray[4], Number); // pixel 3, pixel 4
    set_n_pixel(PixelArray[5], PixelArray[6], Number * 2); // (pixel 5, pixel 6)*2
    //--> pixel output: pixel 5, pixel 6, pixel 5, pixel 6
}
break;

case FAST:
{
    // fast pixel mode up to 1.6 MHz (Mode = 32) - needs UFPM-Option
    // 1 value of 16 bit per pixel
    // 4 pixel per set_pixel command
    //-----
    // | pixel2 | pixel1 | pixel4 | pixel3 |
    // | 16 Bit | 16 Bit | 16 Bit | 16 Bit |
    // | PortOutValue1 | PortOutValue2 |
    //-----
    HalfPeriod = (UINT)round(1.0 * 32.0); // Period = 1 us, PixelFrequency = 1 MHz
    PulseLength = (UINT)round(0.5 * 64.0); // PulseLength = 0.5 us
    set_laser_pulses(HalfPeriod, PulseLength);

    jump_abs(PixelLineStartPosX, PixelLineStartPosY);
    set_pixel_line(Channel, HalfPeriod, dx, dy);
    // low 16 bit - first pixel, high 16 bit - second pixel
    PortOutValue1 = PixelArray[1] | (PixelArray[2] << 16);
    PortOutValue2 = PixelArray[3] | (PixelArray[4] << 16);
    set_n_pixel(PortOutValue1, PortOutValue2, Number); // pixel 2 + pixel 1, pixel 4 + pixel 3

    PortOutValue1 = PixelArray[5] | (PixelArray[6] << 16);
    PortOutValue2 = PixelArray[7] | (PixelArray[8] << 16);
    set_n_pixel(PortOutValue1, PortOutValue2, Number * 2); // (pixel 6 + pixel 5, pixel 8 + pixel 7)*2
    //--> pixel output: pixel 5, pixel 6, pixel 7, pixel 8, pixel 5, pixel 6, pixel 7, pixel 8
}
break;

```

```

case ULTRAFAST:
{
    // ultra fast pixel mode up to 3.2 MHz (Mode = 64) - needs UFPM-Option
    // 1 value of 8 bit per pixel
    // 8 pixel per set_pixel command
    //-----
    // | pixel4 | pixel3 | pixel2 | pixel1 | | pixel8 | pixel7 | pixel6 | pixel5 |
    // | 8 Bit  | 8 Bit  | 8 Bit  | 8 Bit  | | 8 Bit  | 8 Bit  | 8 Bit  | 8 Bit  |
    // |-----|-----|-----|-----| |-----|-----|-----|-----|
    // | PortOutValue1 | PortOutValue2 |
    //-----
    HalfPeriod = (UINT)round(0.5 * 32.0); // Period = 0.5 us, PixelFrequency = 2 MHz
    PulseLength = (UINT)round(0.25 * 64.0); // PulseLength = 0.25 us
    set_laser_pulses(HalfPeriod, PulseLength);

    jump_abs(PixelLineStartPosX, PixelLineStartPosY);
    set_pixel_line(Channel, HalfPeriod, dx, dy);

    // lowest 8 bit - first pixel, highest 8 bit - fourth pixel
    PortOutValue1 = PixelArray[1] | (PixelArray[2] << 8) | (PixelArray[3] << 16) | (PixelArray[4] << 24);
    PortOutValue2 = PixelArray[5] | (PixelArray[6] << 8) | (PixelArray[7] << 16) | (PixelArray[8] << 24);

    // pixel 4 + pixel 3 + pixel 2 + pixel 1, pixel 8 + pixel 7 + pixel 6 + pixel 5
    set_n_pixel(PortOutValue1, PortOutValue2, Number);

    PortOutValue1 = PixelArray[ 9] | (PixelArray[10] << 8) | (PixelArray[11] << 16) | (PixelArray[12] << 24);
    PortOutValue2 = PixelArray[13] | (PixelArray[14] << 8) | (PixelArray[15] << 16) | (PixelArray[16] << 24);

    // (pixel 12 + pixel 11 + pixel 10 + pixel 9, pixel 16 + pixel 15 + pixel 14 + pixel 13)*2
    set_n_pixel(PortOutValue1, PortOutValue2, Number * 2);
    // --> pixel output: pixel 9, pixel 10, pixel 11, pixel 12, pixel 13, pixel 14, pixel 15, pixel 16,
    //                    pixel 9, pixel 10, pixel 11, pixel 12, pixel 13, pixel 14, pixel 15, pixel 16
}
break;

default:
break;

}

set_end_of_list();

```

## 1.11 Hinweise zu **set\_auto\_laser\_control** mit **Ctrl = 7** ("Spot Distance Control")

Ab RTC6-Software-Paket  $\geq$  V1.4.1 akzeptiert **set\_auto\_laser\_control** beim Parameter **Ctrl**<sup>(1)</sup> auch den Wert 7.

In den folgenden Punkten verhält sich die RTC6 für **Ctrl = 7** identisch wie für **Ctrl = 5**:

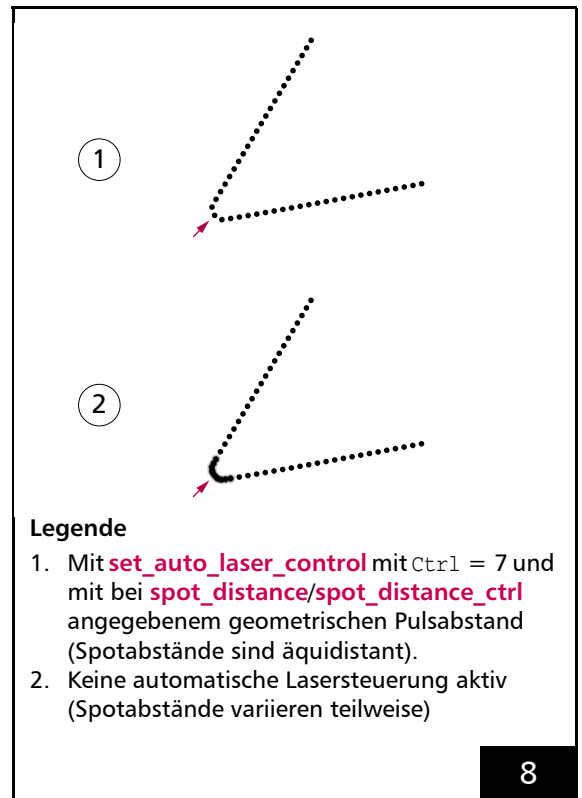
- Die Ausgabeperiode (**HalfPeriod**) des LASER1 Lasersignals wird automatisch basierend auf der aktuellen Galvanometerscanner-Geschwindigkeit korrigiert. Optional kann dazu auch eine Winkel-zu-Bildfeld-Transformation der Galvanometerscanner-Geschwindigkeit aktiviert werden.
- Für eine erfolgreiche Verwendung der automatischen Lasersteuerung (mit **Ctrl = 7** oder **Ctrl = 5**) sollte ein Laser mit "Pulse-on-Demand"-Funktionalität eingesetzt werden, der Laserpulse unverzüglich mit möglichst gleichbleibender Leistung ausgibt, sobald diese von der RTC6 mit dem LASER1-Signal getriggert werden. Die RTC6 unterstützt Pulse-on-Demand-Laser mit Frequenzmodulierungen von bis zu 1 MHz oder mehr. Man beachte dabei die Auflösung von  $1/64 \mu s$  für die **HalfPeriod**.

In den folgenden Punkten unterscheidet sich **Ctrl = 7** wesentlich von **Ctrl = 5**:

- **Ctrl = 7** ist nur für excelliSCAN-Scan-Köpfe (= mit SCANahead-Regelung) verfügbar.
- Bei **Ctrl = 7** berechnet die RTC6 (aus der aktuellen Galvanometerscanner-Geschwindigkeit) automatisch die Ausgabe passend zum (bei **spot\_distance** oder **spot\_distance\_ctrl**) angegebenen geometrischen Pulsabstand (weitere Details siehe Befehls-Tabelle für **set\_auto\_laser\_control**).  
Bei **Ctrl = 5** dagegen muss das Anwenderprogramm die Ausgabeperiode passend zur Galvanometerscanner-Geschwindigkeit vorgeben.

- **Ctrl = 7** kann für Anwendungen mit höheren Genauigkeitsanforderungen und höheren Geschwindigkeitsänderungen als bei **Ctrl = 5** verwendet werden und eignet sich dadurch auch für Mikrobearbeitungen.
- **Ctrl = 7** kann nicht sinnvoll mit **Mode = 1** (Sollgeschwindigkeit) oder **Mode = 5** (Encodergeschwindigkeit) kombiniert werden.

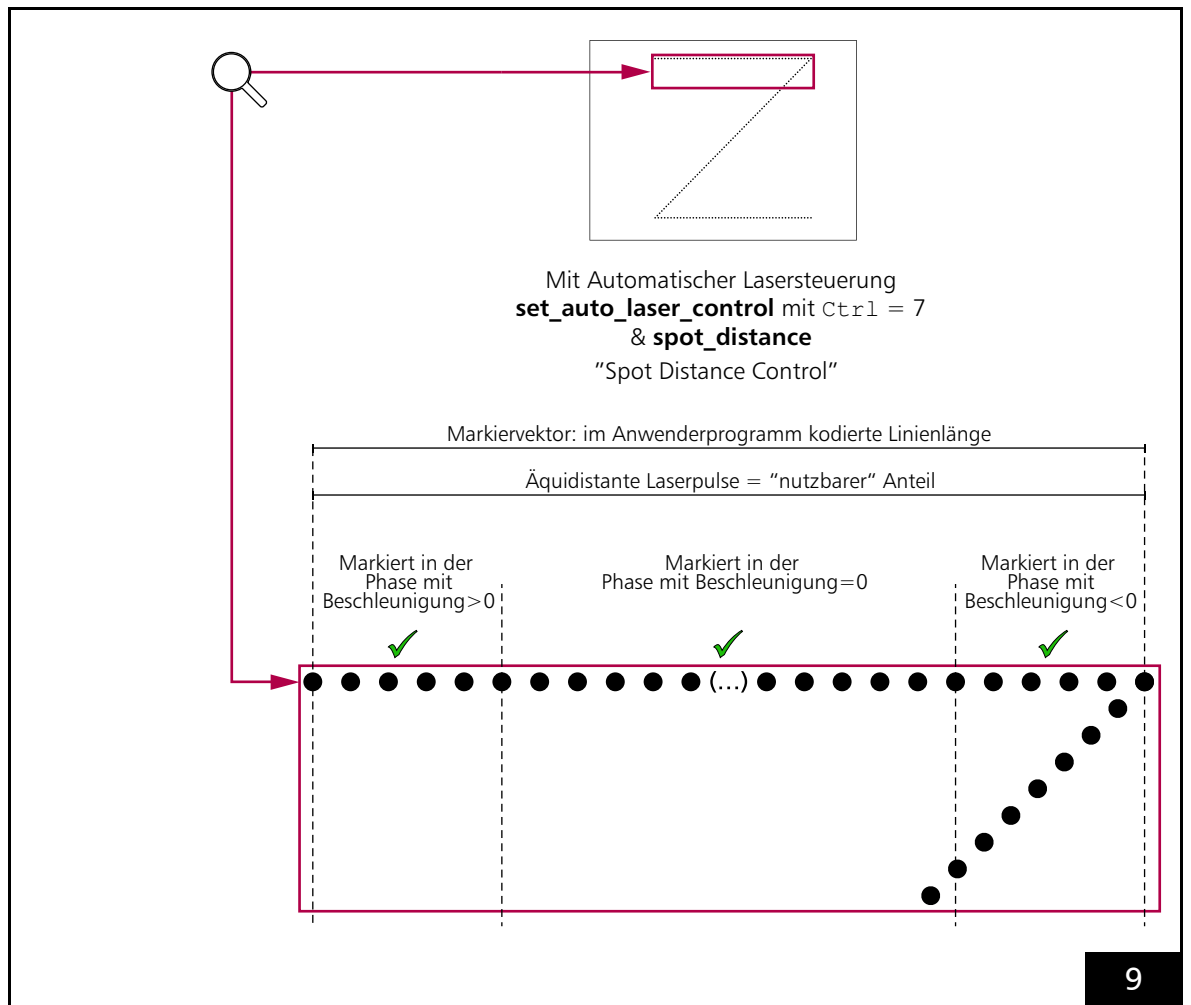
Einen Vergleich eines typischen Markierungsergebnisses mit automatischer Lasersteuerung (**Ctrl = 7**) und ohne automatischer Lasersteuerung zeigt **Abbildung 8**.



Beispiel-Markierungsergebnis (ohne anfängliche Beschleunigungsphasen).

(1) Kontrollparameter zur Initialisierung und Deaktivierung der automatischen Lasersteuerung.

Die in **Abbildung 9** gezeigte Markierung könnte alternativ zu **set\_auto\_laser\_control** mit `Ctrl = 7` auch mit eingeschaltetem Sky-Writing realisiert werden. Allerdings ist die Prozesszeit bei Verwendung von **set\_auto\_laser\_control** mit `Ctrl = 7` kürzer, da keine Vor- und Nachlaufbewegungen erforderlich sind.



Bei **set\_auto\_laser\_control** mit `Ctrl = 7` zeigt das Markierungsergebnis *durchgehend* äquidistante Punkte.

## 1.12 Technische Spezifikationen der RTC6 PCI-Express-Karte

### Systemanforderungen

Windows-PC mit freiem PCI-Express-Steckplatz

Betriebssystem Mit RTC6-Software-Paket  
≥ V1.3:  
Microsoft  
Windows 10, 8, 7  
als 32-Bit oder 64-Bit  
Version.  
Vista und XP werden nicht  
unterstützt!

### Abmessungen

Länge 160 mm

Breite 104,5 mm

### Schnittstelle zum PC

PCI-Express x-1, Version 1.0

### Stiftleiste "McBSP/ANALOG"

Anschluss 10-polige Stiftleiste

McBSP-Schnittstellen-  
funktionalität Wie bei RTC5 PCI-Karten  
und RTC5 PCI-Express-  
Karten, siehe RTC5-  
Handbuch, Seite 54

SPI-Schnittstellen-  
funktionalität Nein

2 Analogeingänge (ANALOG IN0, ANALOG IN1):

- Eingangsspannungs- 0 V ... 10 V  
bereich
- Eingangsimpedanz 4 kΩ
- ADC-Auflösung 12 Bit
- Bezugspunkt GND

### Sonstige Anschlüsse und Spezifikationen

Gesamtkapazität des 8.388.608 Listenplätze  
Listenspeichers

Virtuelles Bildfeld, *Nicht* wie bei RTC5-  
z. B. für "Processing on Karten:  
the fly" -268.435.456...  
+268.435.455  
(28 Bit mit Vorzeichen  
= 29 Bit)<sup>(a)</sup>

Sonst wie RTC5 PCI-Karte und RTC5 PCI-Express-  
Karte.

(a) Mit RTC6-Software-Paket ≥ V1.4.1.

### **1.13 Konformität mit der EU-Richtlinie zur elektromagnetischen Verträglichkeit (EMV)**

Die Konformität der RTC6 PCI-Express-Karte mit der EG-Richtlinie 2014/30/EU (Elektromagnetische Verträglichkeit) wurde festgestellt.

Dazu wurde die RTC6 PCI-Express-Karte in einen PC eingebaut und zusammen mit einem excelliSCAN 14 Scan-Kopf (SL2-100-Schnittstelle) getestet.

#### **Prüfbestimmungen**

Nachweis zur Erfüllung der Schutzziele der EG-Richtlinie 2014/30/EU (CE-Konformität für EMV) auf der Basis von

- EN 61000-6-2: 2005 + AC: 2005
- EN 61000-6-4: 2007 + A1: 2011

#### **Ergebnis**

Die Prüflinge erfüllen die Spezifikationen.

### **1.14 Konformität mit FCC-Bestimmungen**

Die RTC6 PCI-Express-Karte wurde getestet und entspricht den Grenzwerten für Digitalgeräte der Klasse A, gemäß Teil 15 der FCC-Bestimmungen.

Diese Grenzwerte sollen einen angemessenen Schutz gegen schädliche Störungen bieten, wenn die RTC6 PCI-Express-Karte in einer kommerziellen Umgebung betrieben wird. Die RTC6 PCI-Express-Karte erzeugt, verwendet und emittiert möglicherweise Hochfrequenzenergie und kann, sofern nicht in Übereinstimmung mit dieser Bedienungsanleitung installiert und verwendet, Störungen im Funkverkehr verursachen. Der Betrieb der RTC6 PCI-Express-Karte in einem Wohngebiet wird wahrscheinlich schädliche Störungen verursachen; in diesem Fall ist der Benutzer verpflichtet, die Störungen auf eigene Kosten zu beseitigen.

## 2 Anhang A: Die RTC6 Ethernet-Karte

### 2.1 Produktübersicht

#### 2.1.1 RTC6 Ethernet-Karte und RTC6 PCI-Express-Karte – Verwendung und Vergleich

Bei RTC6 Ethernet-Karten und RTC6 PCI-Express-Karten sind die *grundsätzlichen Funktionen* gleich:

- Echtzeitsteuerung von Laser und Scan-System.
- Übertragung von direkten Befehlen (Kontrollbefehle) vom Rechner auf die Karte. Diese Befehle werden sofort ausgeführt.
- Übertragung von Listebefehlen vom Rechner in den Listenspeicher der Karte. Die Befehle werden erst nach dem Start der Liste ausgeführt.
- Die Listenausführung erfolgt in Echtzeit.
- Die Berechnung der Sollpositionen erfolgt alle 10  $\mu$ s.
- Es erfolgt eine Feldkorrektur der Sollpositionen.
- Die Schnittstelle zum Scan-System benutzt das 20-Bit SL2-100-Protokoll<sup>(1)</sup>.
- Zur Softwareentwicklung ist in der RTC6DLL.dll/RTC6DLLx64.dll der komplette Befehlssatz für RTC6 Ethernet-Karten und RTC6 PCI-Express-Karten enthalten<sup>(2)</sup>.

Bei RTC6 Ethernet-Karten ist im Vergleich zu den RTC6 PCI-Express-Karten folgendes *anders*:

- Ethernet-Schnittstelle anstatt PCIe-Bus, d.h.
  - Die Kommunikation mit dem Rechner erfolgt über das TCP/IP- und UDP-Protokoll anstatt PCIe.
  - RTC6 Ethernet-Karten funktionieren *ohne* RTC6-Kartentreiber<sup>(3)</sup>.
  - Die Karte muss nicht in einen Rechner gesteckt werden, sondern kann z. B. in einen Schaltschrank installiert werden. Ein dedizierter und räumlich nahe beim Lasersystem stehender Rechner ist somit nicht nötig.

(1) Bei RTC4 Ethernet-Karten: 16-Bit XY2-100-Protokoll.

(2) Bei RTC4 Ethernet-Karten:  
 RTC4ethDLL.dll/RTC4ethDLLx64.dll ist anstatt  
 RTC4DLL.dll/RTC4DLLx64.dll zu verwenden.  
 Zusätzlich gibt es noch die  
 RTC4netD11.dll/RTC4netD11x64.dll.

(3) Der RTC6-Kartentreiber wird nur für RTC6 PCI-Express-Karten benötigt. Bei RTC6 Ethernet-Karten wird stattdessen die Netzwerk-Bibliothek Winsock verwendet, die standardmäßig in MS Windows enthalten ist. Winsock oder Windows Sockets ist eine Windows-Programmierschnittstelle (API) zum Zugriff auf Rechnernetze über Sockets.



## 2.1.2 Systemanforderungen



### Achtung!

- Die RTC6 Ethernet-Karte darf keinesfalls ohne Abschirmung betrieben werden (z. B. außerhalb eines metallischen Gehäuses), um Störeinkopplungen und Störabstrahlungen zu vermeiden.
- Die RTC6 Ethernet-Karte ist ausschließlich für den gewerblichen Gebrauch bestimmt. Sie ist zum Einbau in eine Maschine (in der Regel in ein Lasersystem) vorgesehen und entspricht **nicht** allen Anforderungen an ein gebrauchsfertiges Endprodukt. Nehmen Sie die RTC6 Ethernet-Karte erst dann in Betrieb, wenn sie in eine Maschine eingebaut ist, die den Bestimmungen aller dafür anzuwendenden Richtlinien und Normen (des betreffenden Landes) entspricht. Sie ist **nicht** für den Einsatz als Kinderspielzeug, im Haushalt oder unter widrigen Umgebungsbedingungen (z. B. im Freien) geeignet. Entsprechende Vorsichtsmaßnahmen zur Vermeidung solcher Fehlanwendungen sind von Benutzern zu treffen.
- Die Installation und Inbetriebnahme darf nur durch Personen erfolgen, die über eine ausreichende Qualifikation verfügen, u.a. über Kenntnisse im sicheren Umgang mit elektrischen Geräten. Führen Sie Installations- und Wartungsarbeiten nur bei ausgeschalteter Versorgungsspannung und ausgeschaltetem Laser aus.
- Die RTC6 Ethernet-Karte ist eine Einrichtung der Klasse A. Sie kann im Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen.

## Hardware

- Abgeschirmtes Gehäuse (z. B. Schaltschrank), mit
  - Möglichkeit zur mechanischen Befestigung incl. Befestigungsmaterial
  - entsprechender thermischer Anbindung (Kühlung) der Karte
  - Stromzufuhr
  - Ethernet-Kabel
  - Netzteil gemäß Spezifikation
  - Kabel mit Stecker für die Spannungsversorgung der RTC6 Ethernet-Karte
  - Flachbandkabel mit Stecker zu den Peripheriegeräten (Laser, Scan-Kopf, Erweiterungen)
- Windows-PC (es ist kein dedizierter PC notwendig, der an der Maschine steht)
- Gigabit-Ethernet (bevorzugt), 10/100 Mbit/s-Ethernet

## Software

- TCP/IP Protokoll IPv4
- UDP-Protokoll
- Verwendete Ports<sup>(1)</sup> (Voreinstellung)
  - 63749 (UDP)
  - 63750 (UDP, TCP)

Für die Entwicklung von Anwenderprogrammen:

- DLL/Importdeklarationen (im RTC6-Software-Paket enthalten)

## 2.1.3 Optionale Funktionalitäten

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.3 "Optionale Funktionalitäten"](#), Seite 9.

## 2.1.4 Kennzeichnung

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.4 "Kennzeichnung"](#), Seite 9.

## 2.1.5 Typenbezeichnung

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.5 "Typenbezeichnung"](#), Seite 9.

(1) Bei RTC4 Ethernet-Karten: Ports 63740, 63741, 63742.

### **2.1.6 Auspackanleitung und typischer Lieferumfang**

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.6](#) "Auspackanleitung und typischer Lieferumfang", Seite 9.

### **2.1.7 Mitgelieferte Software**

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.7](#) "Mitgelieferte Software", Seite 9.

### **2.1.8 Zubehör für die RTC6 Ethernet-Karte**

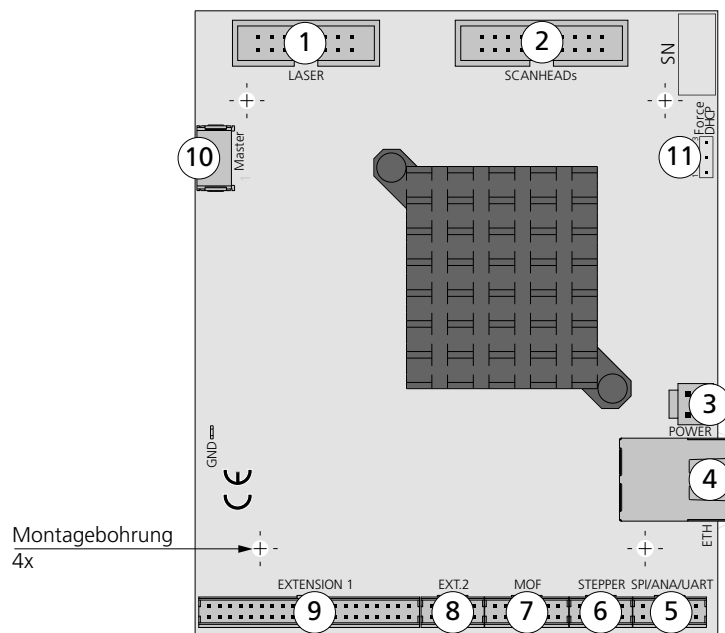
Das in [Kapitel 1.3.8](#) "Zubehör für die RTC6 PCI-Express-Karte", Seite 10 genannte Zubehör für RTC6 PCI-Express-Karten eignet sich (ohne entsprechende Adapter ) nicht für RTC6 Ethernet-Karten (z.B. wegen abweichender Steckverbinder und Rastermaße).

### **2.1.9 Ergänzende Software**

Wie bei RTC6 PCI-Express-Karten, siehe [Kapitel 1.3.9](#) "Ergänzende Software", Seite 10.

## 2.2 Layout, Schnittstellen, Jumpereinstellungen

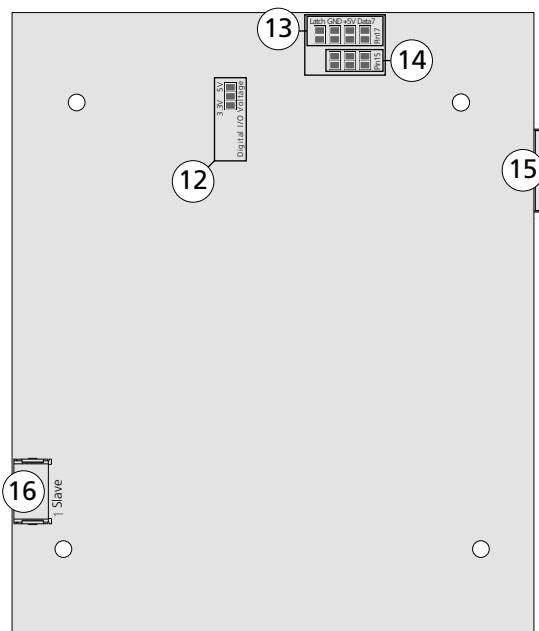
### 2.2.1 Layout – Oberseite



#### Legende

1. LASER ..... 16-polige Stiftleiste. Zum Anschließen des Lasers. Mit digitalen und analogen Ausgängen.  
Details siehe "[LASER-Stiftleiste](#)", Seite 70.
2. SCANHEADs ..... 20-polige Stiftleiste. Zum Anschließen des ersten und zweiten Scan-Kopfs.  
Details siehe "[SCANHEADs-Stiftleiste](#)", Seite 73.
3. POWER ..... 2-polige Stiftleiste. Für die Spannungsversorgung.  
Details siehe "[POWER-Stiftleiste](#)", Seite 75.
4. ETH ..... RJ-48 8P8C-Buchse. Für den Netzwerkanschluss.  
Details siehe "[ETH-Buchse](#)", Seite 75.
5. SPI/ANA/UART ..... 12-polige Stiftleiste. Hardware-Schnittstelle für verschiedene Datenaustauschverfahren.  
Mit analogen Eingängen. Details siehe "[SPI/ANA/UART-Stiftleiste](#)", Seite 76.
6. STEPPER ..... 10-polige Stiftleiste. Zur Ansteuerung von bis zu zwei Schrittmotoren.  
Details siehe "[STEPPER-Stiftleiste](#)", Seite 77.
7. MOF ..... 14-polige Stiftleiste. Für "Processing on the fly"-Anwendungen. Die Stiftleiste ist immer vorhanden, jedoch ist deren Funktionalität eine zukaufbare Option.  
Details siehe "[MOF-Stiftleiste](#)", Seite 77.
8. EXT. 2 ..... 10-polige Stiftleiste. Für Erweiterungen. Mit einem 8-Bit Digital-Ausgang.  
Details siehe "[EXT. 2-Stiftleiste](#)", Seite 79.
9. EXTENSION 1 ..... 40-polige Stiftleiste. Für Erweiterungen. Mit einem 16-Bit-Digital-Ausgang und einem 16-Bit-Digital-Eingang. Details siehe "[EXTENSION 1-Stiftleiste](#)", Seite 78.
10. Master ..... 6-polige Stiftleiste. Zum Verbinden mit einer anderen RTC6 Ethernet-Karte zwecks Taktsynchronisierung.  
Details siehe "[Master-Stiftleiste, Slave-Stiftleiste](#)", Seite 81.
11. force DHCP ..... Steckjumperfeld zum Ignorieren/nicht Ignorieren der gespeicherten statischen IP-Adresse.  
Details siehe "[Steckjumperfeld 'Force DHCP'](#)", Seite 85.

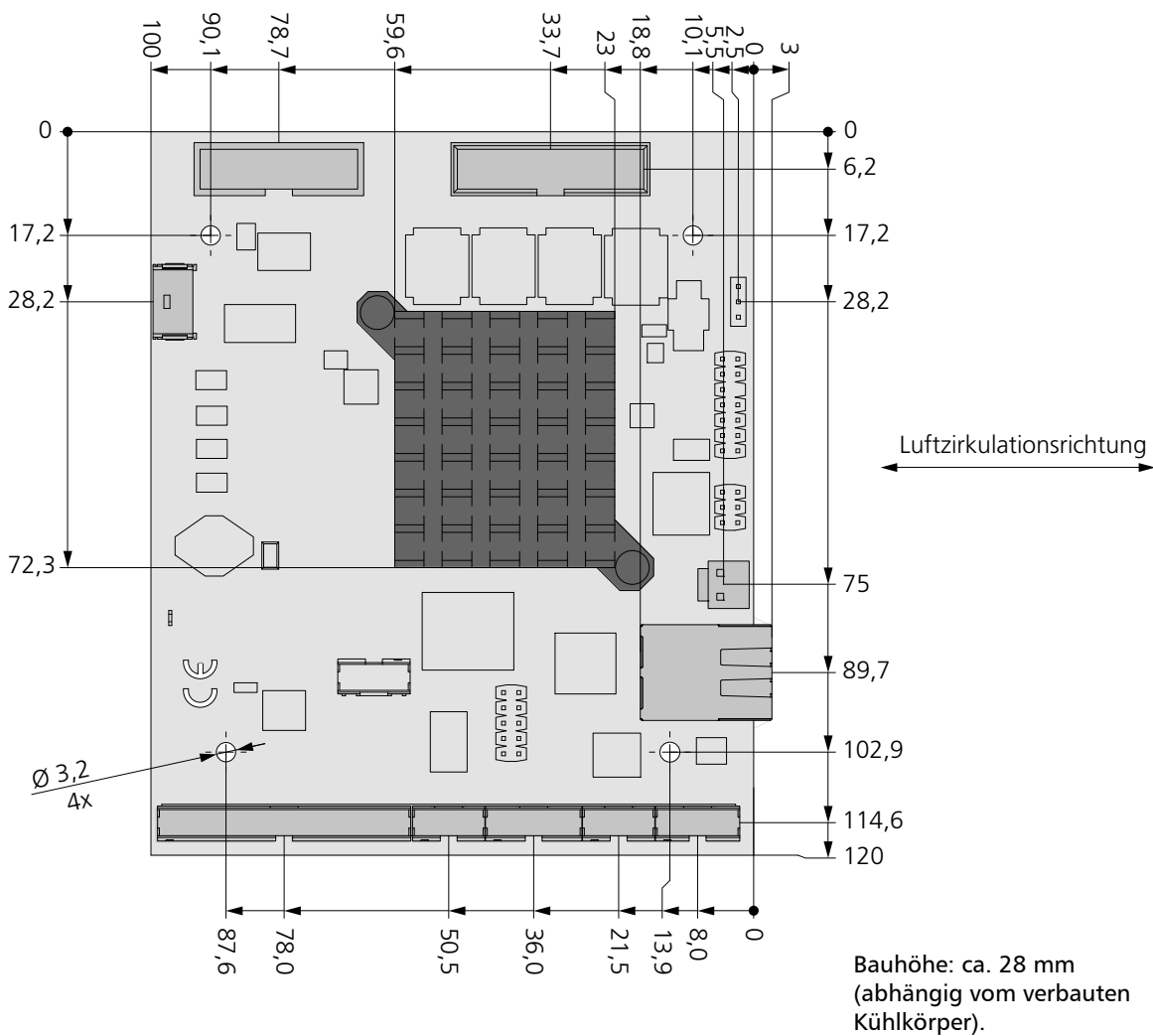
## 2.2.2 Layout – Unterseite



### Legende

- 12. Lötjumperfeld A . . . . . Zum Konfigurieren des Pegels der Ausgangssignale an der EXTENSION 1-Stiftleiste.  
Details siehe "Lötjumperfeld A – Einstellen des Ausgangssignalpegels an der EXTENSION 1-Stiftleiste",  
Seite 82.
- 13. Lötjumperfeld B . . . . . Zum Konfigurieren des Signals an EXT. 2-Stiftleiste Pin (09).  
Details siehe "Lötjumperfeld B – Konfiguration von Pin (09) der EXT. 2-Stiftleiste", Seite 83.
- 14. Lötjumperfeld C . . . . . Zum Konfigurieren des Signals an EXT. 2-Stiftleiste Pin (08).  
Details siehe "Lötjumperfeld C – Konfiguration von Pin (08) der EXT. 2-Stiftleiste", Seite 84.
- 15. ETH Buchse. . . . . Siehe **Abbildung 10**.
- 16. Slave. . . . . 6-polige Stiftleiste. Zum Verbinden mit einer anderen RTC6 Ethernet-Karte zwecks Taktsynchronisierung.  
Details siehe "Master-Stiftleiste, Slave-Stiftleiste", Seite 81.

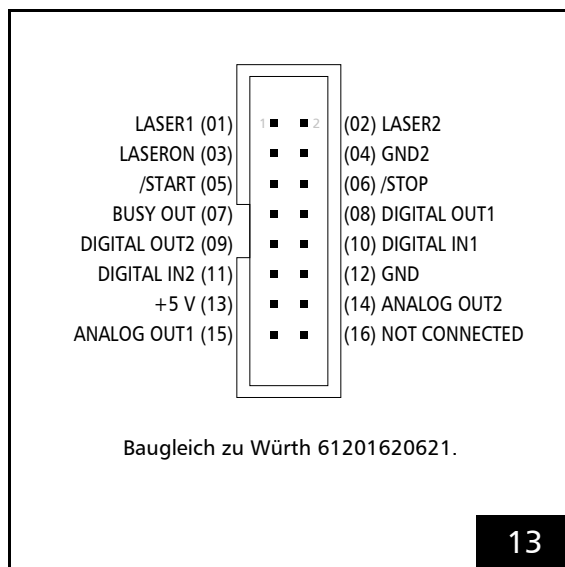
### 2.2.3 Abmessungen und Steckverbinderpositionen



## 2.2.4 LASER-Stiftleiste

Die LASER-Stiftleiste ist 16-polig<sup>(1)</sup>. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 1.

Die Pinbelegung zeigt **Abbildung 13**.



LASER-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,54 mm. Bei RTC6 Ethernet-Karten ohne der "DC/DC" Option sind die GND2 und GND identisch. Bei RTC6 Ethernet-Karten mit der "DC/DC" Option sind GND2 und die Laserausgangssignale galvanisch von GND entkoppelt.

### Hinweise

- Bei der RTC6 Ethernet-Karte ist GND die Masse am POWER-Stecker. Siehe **Kapitel 2.2.6 "POWER-Stiftleiste"**, Seite 75. Bei RTC5-Karten und der RTC6 PCI-Express-Karte ist GND die PC-Masse.

## Laserausgangssignale

Wie bei RTC5-Karten, siehe RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Seite 47. Beachten Sie aber, dass sich die Pin-Nummern unterscheiden.

Signal	RTC6 Ethernet-Karte	RTC5-Karten
LASERON	Pin (03)	Pin (02)
LASER1	Pin (01)	Pin (01)
LASER2	Pin (02)	Pin (09)

## Externe Steuersignale

Wie bei RTC5-Karten, siehe RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Seite 47.

## BUSY-Status

Wie bei RTC5-Karten, siehe RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Seite 47. Beachten Sie aber, dass sich die Pin-Nummern unterscheiden.

Signal	RTC6 Ethernet-Karte	RTC5-Karten
BUSY OUT	Pin (07)	Pin (04)

## 2 Bit-Digital-Eingang und 2 Bit-Digital-Ausgang

Wie bei RTC5-Karten, siehe RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Seite 47. Beachten Sie aber, dass sich die Pin-Nummern unterscheiden.

Signal	RTC6 Ethernet-Karte	RTC5-Karten
DIGITAL IN1	Pin (10)	Pin (13)
DIGITAL IN2	Pin (11)	Pin (06)
DIGITAL OUT1	Pin (08)	Pin (12)
DIGITAL OUT2	Pin (09)	Pin (05)

(1) Bei RTC6 PCI-Express-Karten: 15-polige Sub-D-Buchse, weiblich.

## 12-Bit Analog-Ausgänge

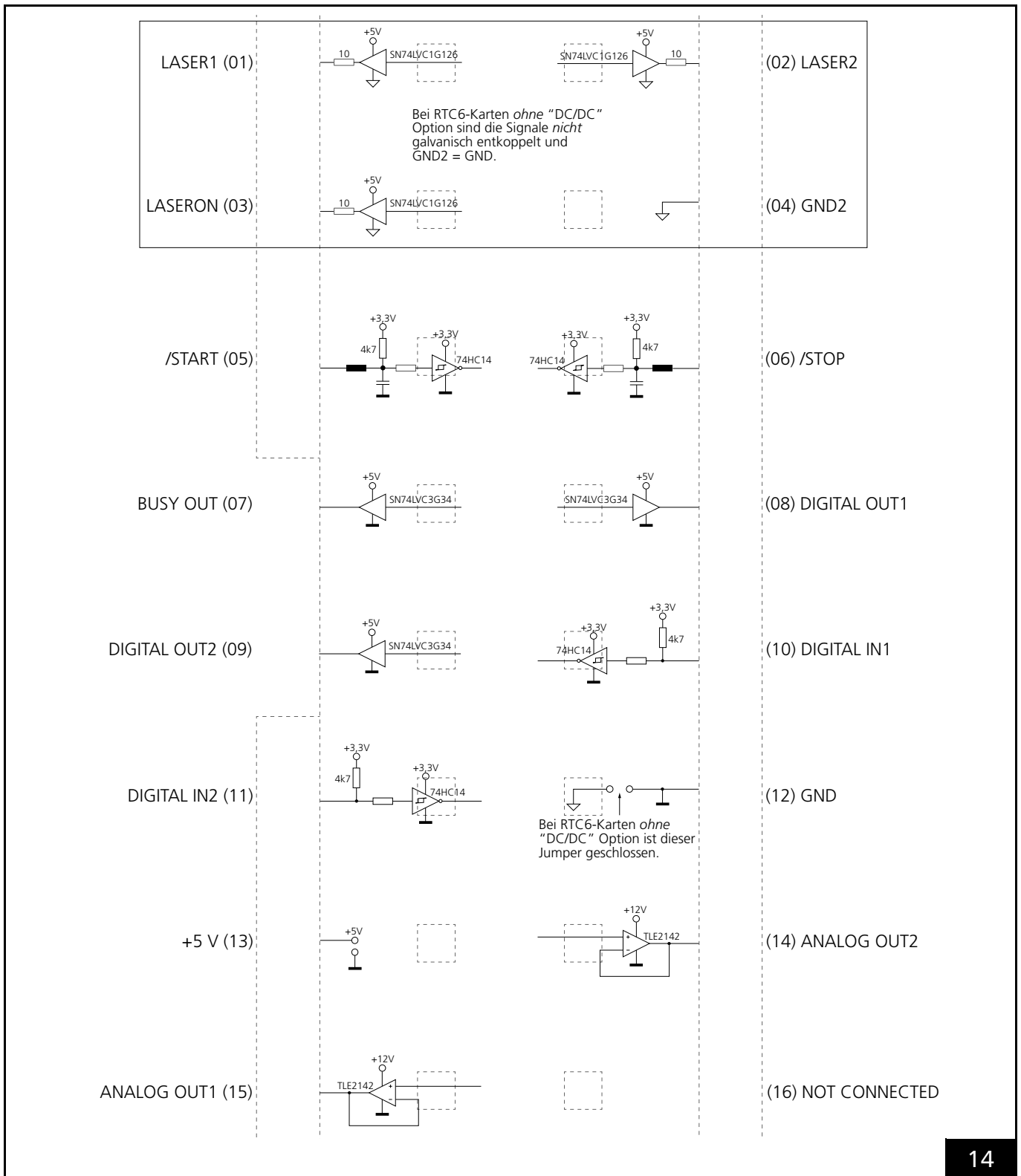
Wie bei RTC5-Karten, siehe RTC5-Handbuch, Kapitel 4.4.1 "Laseranschluss-Buchse", Seite 47. Beachten Sie aber, dass sich die Pin-Nummern unterscheiden.

Signal	RTC6 Ethernet-Karte	RTC5-Karten
ANALOG OUT1	Pin (15)	Pin (08)
ANALOG OUT2 (a)	Pin (14)	Pin (15)

(a) Bei RTC5-Karten und der RTC6 PCI-Express-Karte ist das ANALOG OUT2-Signal zusätzlich auch an der MARKING ON THE FLY-Stiftleiste verfügbar.

## Ein-/Ausgangsbeschaltung

Die Ein-/Ausgangsbeschaltung der 16-poligen LASER-Stiftleiste zeigt **Abbildung 14**. Sie ist mit Ausnahme der Pin-Nummern identisch zu RTC5-Karten und RTC6 PCI-Express-Karten.





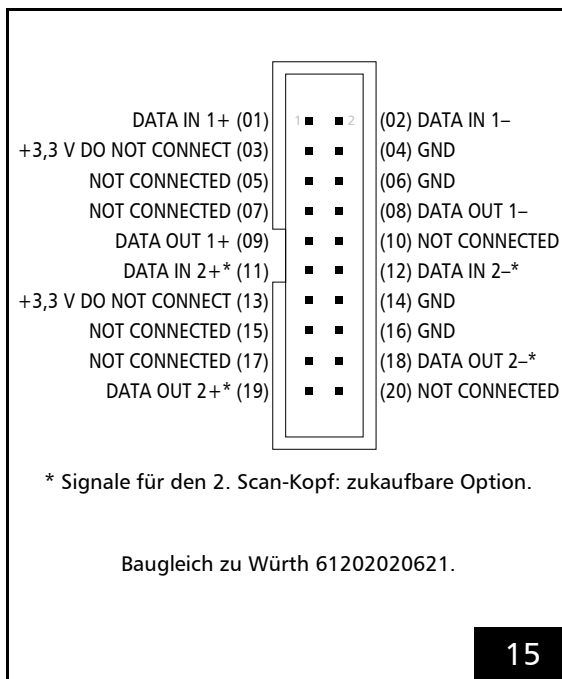
## 2.2.5 SCANHEADs-Stiftleiste

Die SCANHEADs-Stiftleiste ist 20-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 2.

Die SCANHEADs-Stiftleiste stellt die gleichen Signale bereit wie die SCAN HEAD- und 2. SCANHEAD-Stiftleisten der RTC6 PCI-Express-Karte<sup>(1)</sup>.

Es stehen zur digitalen Ansteuerung von Scan-Systemen der primäre Scan-Kopf-Anschluss (Pin (01)...Pin (10)) und der optional aktivierte sekundäre Scan-Kopf-Anschluss (Pin (11)...Pin (20)) zur Verfügung<sup>(2)</sup>. An der Stiftleiste werden Ansteuerwerte für das Scan-System ausgegeben und Statussignale empfangen, die vom Scan-System zurückgegeben werden.

Die Pinbelegung zeigt **Abbildung 15**.

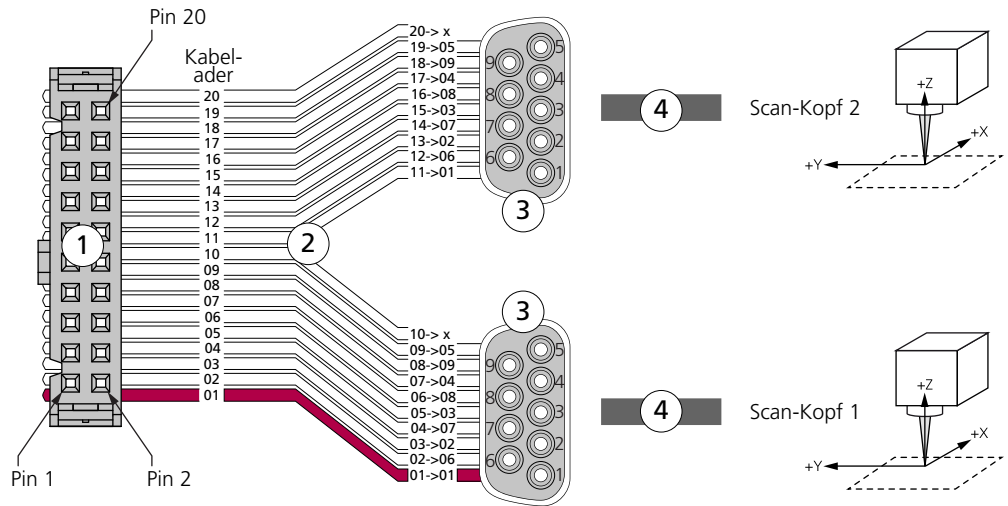


SCANHEADs-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,54 mm.

(1) RTC6 Ethernet-Karten haben nur diesen einen einzigen Anschluss. Dagegen haben RTC6 PCI-Express-Karten 2 getrennte Anschlüsse:

1 × SCANHEAD (9-polige Sub-D-Buchse, weiblich)  
1 × 2. SCANHEAD (10-polige Stiftleiste mit 2,54 mm Rastermaß)

(2) Die Ausgabe der Signale für den zweiten Scan-Kopf ist eine zukaufbare Option.



#### Legende

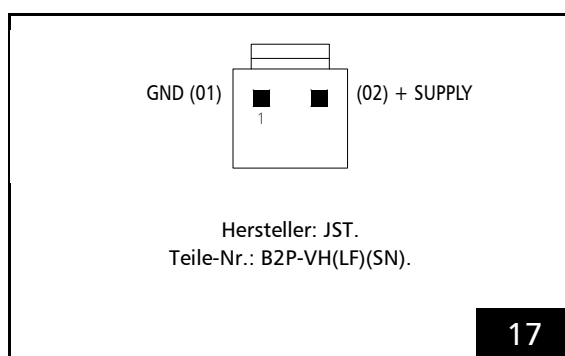
1. Steckverbinder, 20-polig, weiblich. Für SCANHEADs-Stiftleiste, siehe [Abbildung 15](#). Baugleich mit Würth 61202023021.
2. Flachbandkabel 26 AWG, 20 Leitungen, Rastermaß 1,27 mm.
3. Steckverbinder, 9-polig Sub-D, weiblich.
4. Datenkabel für SL2-100, z. B. #115428.

Verkabelung (Vorschlag). Die tatsächliche Umsetzung kann auf Grund von Kundenanforderungen abweichend sein. Die dargestellten Teile sind nicht im Standardlieferumfang der RTC6 Ethernet-Karte enthalten!

## 2.2.6 POWER-Stiftleiste

Die POWER-Stiftleiste ist 2-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 3. Über sie wird der RTC6 Ethernet-Karte die Versorgungsspannung zugeführt.

Die Pinbelegung zeigt **Abbildung 17**.



POWER-Stiftleiste. Das Rastermaß der Pins ist 3,96 mm.

Pin	Versorgungsspannung
(01) GND	Masse.
(02) + Supply	Siehe <b>Kapitel 2.7 "Technische Spezifikationen der RTC6 Ethernet-Karte"</b> , Seite 117.

### Hinweise

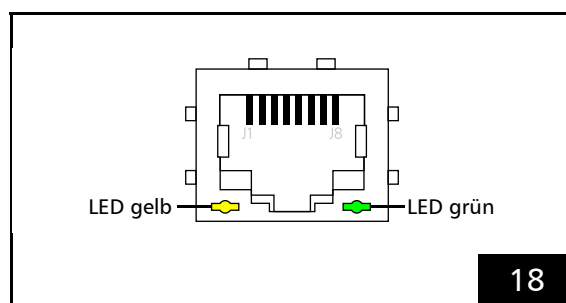
- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
Hersteller: JST  
1 × Steckergehäuse, Teile-Nr. VHR-2N  
2 × Crimpkontakt, Teile-Nr. SVH-41T-P1.1.

## 2.2.7 ETH-Buchse

Die ETH-Buchse entspricht dem Standard RJ-48 8P8C (umgangssprachlich: RJ-45). Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 4. Sie dient dazu, die RTC6 Ethernet-Karte an das Netzwerk anzuschließen.

In die Buchse sind folgende LEDs zur Statusanzeige integriert, siehe **Abbildung 18**:

- LED gelb, Funktion "IP address assigned". Die LED leuchtet *dauerhaft*, sobald die RTC6 Ethernet-Karte eine IP-Adresse bezogen hat (statische IP oder DHCP<sup>(1)</sup>). Ohne einprogrammierte statische IP zeigt die LED die erfolgreiche IP-Adressvergabe per DHCP an. Im Falle einer einprogrammierten statischen IP bedeutet eine aktive LED nicht zwingend eine im Subnetz gültige IP-Adresse. Die LED *blinkt* bei Netzwerkverkehr ("Traffic"), sobald die Karte von einem PC in Besitz genommen (akquiriert) ist.
- LED grün, Funktion "Link / Traffic". Die LED leuchtet *dauerhaft*, sobald eine Verbindung hergestellt ist ("Link"). Das ist gewöhnlich der Fall, wenn das Ethernet-Kabel angesteckt ist und auf der Gegenseite ein Router oder eine Netzwerkkarte aktiv ist. Die LED *blinkt* bei Netzwerkverkehr ("Traffic").



ETH-Buchse. RJ-48 8P8C.

(1) auto-IP wird derzeit nicht unterstützt.

## 2.2.8 SPI/ANA/UART-Stiftleiste

Die SPI/ANA/UART-Stiftleiste ist 12-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 5.

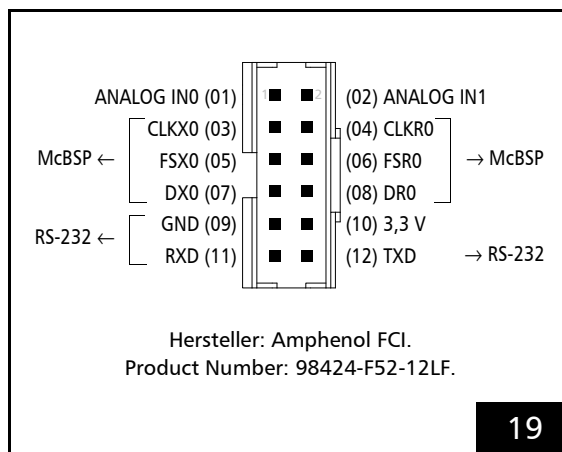
Die SPI/ANA/UART-Stiftleiste ist eine Hardware-Schnittstelle, die für Analog-Eingänge sowie die Datenaustauschverfahren McBSP (Multichannel Buffered Serial Port) und RS-232<sup>(1)</sup> vorgesehen ist.

Die SPI/ANA/UART-Stiftleiste entspricht funktional der:

- McBSP/ANALOG-Stiftleiste und RS232-Stiftleiste bei RTC6 PCI-Express-Karten
- SPI / I2C-Stiftleiste und RS232-Stiftleiste bei RTC5-Karten

Die entsprechenden Pinbelegungen zeigt **Abbildung 19**.

Die Pins können gleichzeitig verwendet werden.



SPI/ANA/UART-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,00 mm.

### Hinweise

- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
 Hersteller: Amphenol FCI  
 Serie: Minitex<sup>®</sup>  
 1 × Steckergehäuse, Teile-Nr. 90311-012LF  
 12 × Crimpkontakt, Teile-Nr. 77138-101LF

(1) SPI (Serial Peripheral Interface) und I2C (Inter-Integrated Circuit) werden nicht unterstützt.

## Analog-Eingänge

An der SPI/ANA/UART-Stiftleiste werden zwei Analog-Eingänge bereitgestellt: ANALOG IN0 und ANALOG IN1. Dort angelegte Spannungen werden unaufgefordert in einer Endlos-Serie automatisch konvertiert (Dauer ca. 0,3 ms) und als 12-Bit-Digitalwerte zum DSP übertragen.

Die anliegenden Werte können mit dem Kontrollbefehl `read_analog_in` jederzeit ausgelesen werden.

### Spezifikationen

- Eingangsspannungsbereich: 0 V...10 V
- Eingangsimpedanz: > 5 kΩ
- ADC-Auflösung: 12 Bit

Die Eingangs-Signale beziehen sich auf GND, siehe **"Hinweise"**, Seite 70.

## Verwendung der McBSP-Schnittstelle

Siehe RTC5-Handbuch, Abschnitt "Verwendung als McBSP-Schnittstelle", Seite 54.

## Verwendung der RS-232 Schnittstelle

Die RS232-Stiftleiste der RTC6 PCI-Express-Karte und der RTC5-Karten entfällt bei RTC6 Ethernet-Karten.

Stattdessen wird an der SPI/ANA/UART-Stiftleiste eine Schnittstelle für das RS-232<sup>(2)</sup> Datenaustauschverfahren bereitgestellt. Damit können externe Geräte angesteuert werden.

Spezifikationen und weitere Informationen wie RTC5, siehe RTC5-Handbuch, Kapitel 4.4.5 "RS232-Stiftleiste", Seite 54.

Die Signale beziehen sich auf GND, siehe **"Hinweise"**, Seite 70.

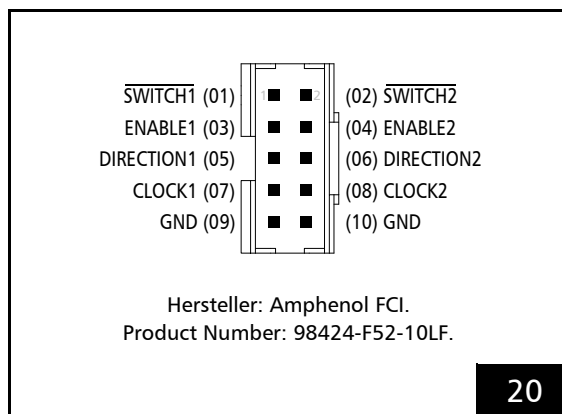
(2) Die Beschriftung der Stiftleiste lautet ".../UART". Das steht für Universal Asynchronous Receiver/Transmitter und bezieht sich auf die elektronische Schaltung, die die auf der RS-232-Schnittstelle zu übertragenden Datenbits und den dazu notwendigen Datenrahmen erzeugt.

## 2.2.9 STEPPER-Stiftleiste

Die STEPPER-Stiftleiste ist 10-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 6.

An der STEPPER-Stiftleiste können Signale zur Ansteuerung von bis zu zwei Schrittmotoren ausgegeben werden.

Die Pinbelegung zeigt **Abbildung 20**.



STEPPER-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,00 mm.

Alle Signale beziehen sich auf GND, siehe **"Hinweise"**, **Seite 70**.

Spezifikationen und weitere Informationen wie RTC5, siehe RTC5-Handbuch, Kapitel 4.4.7 "STEPPER MOTOR-Stiftleiste (Schrittmotoransteuerung)", Seite 58.

### Hinweise

- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
Hersteller: Amphenol FCI  
Serie: Minitex®  
1× Steckergehäuse, Teile-Nr. 90311-010LF  
10× Crimpkontakt, Teile-Nr. 77138-101LF.

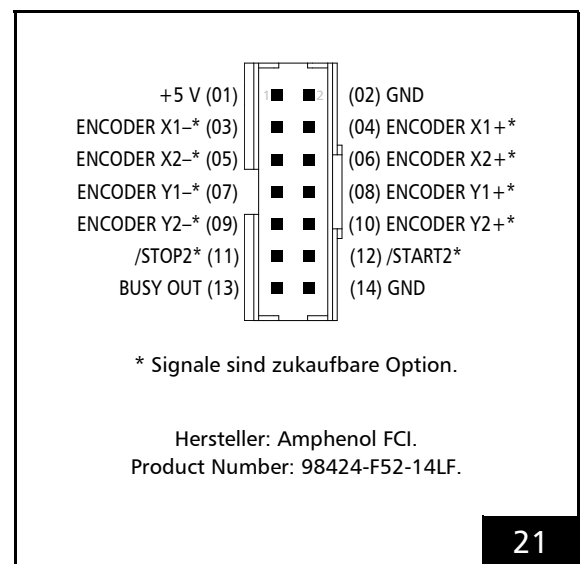
## 2.2.10 MOF-Stiftleiste

Die MOF-Stiftleiste ist 14-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 7.

Die MOF<sup>(1)</sup>-Stiftleiste stellt u. a. Pins für Encoder-Eingänge (zur Einbindung von Encoder-Signalen) zur Verfügung.

Ist die "Processing on the fly"-Funktionalität (zukaufbare Option) auf der Karte freigeschaltet, kann damit eine Lasermaterialbearbeitung bewegter Werkstücke (z. B. auf einem Förderband oder auf einer rotierenden Scheibe) realisiert werden, siehe RTC5-Handbuch, Kapitel 8.7 "Processing on the fly" (optional)", Seite 199.

Die Pinbelegung zeigt **Abbildung 21**.



MOF-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,00 mm.

### Hinweise

- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
Hersteller: Amphenol FCI  
Serie: Minitex®  
1× Steckergehäuse, Teile-Nr. 90311-014LF  
14× Crimpkontakt, Teile-Nr. 77138-101LF.
- Die RTC6 Ethernet-Karte hat an ihrer MOF-Stiftleiste *keinen* Pin für das ANALOG OUT2-Signal (im Unterschied zu MARKING ON THE FLY-Stiftleisten der RTC5-Karten und der RTC6 PCI-Express-Karte), siehe RTC5-Handbuch, Kapitel 4.4.4 "MARKING ON THE FLY-Stiftleiste", Seite 53.

(1) Marking on the fly (= "Processing on the fly").

## Encoder-Eingänge

Wie bei RTC5, siehe RTC5-Handbuch, Abschnitt "Encoder-Eingänge", Seite 53.

## Externe Steuersignale

Wie bei RTC5, siehe RTC5-Handbuch, Abschnitt "Externe Steuersignale", Seite 53.

Die Signale beziehen sich auf GND, siehe "Hinweise", Seite 70.

## BUSY OUT-Status

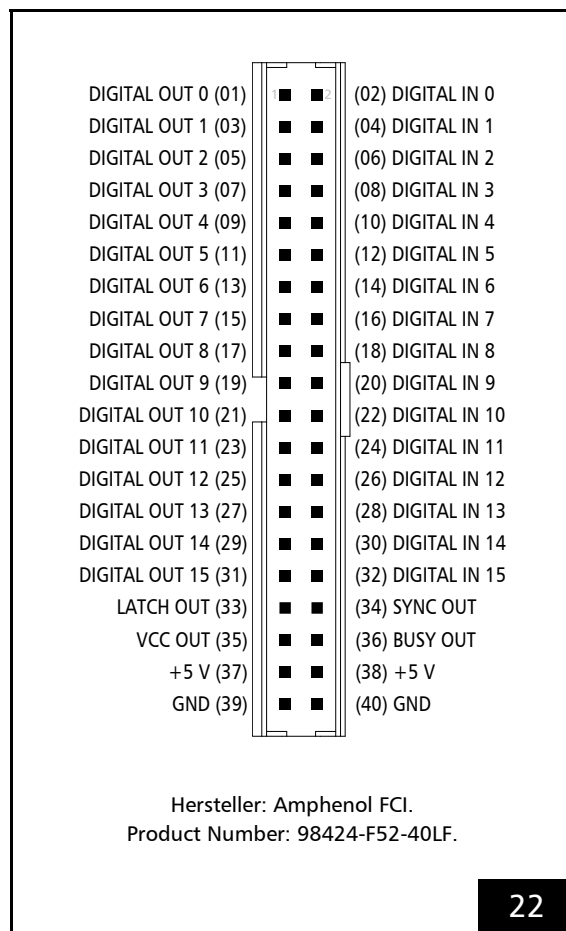
Das BUSY OUT-Signal an Pin (13) ist identisch mit dem BUSY OUT-Signal an der LASER-Stiftleiste, siehe Kapitel 2.2.4 "LASER-Stiftleiste", Seite 70.

Das Signal bezieht sich auf GND, siehe "Hinweise", Seite 70.

## 2.2.11 EXTENSION 1-Stiftleiste

Die EXTENSION 1-Stiftleiste ist 40-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in **Abbildung 10**, Nummer 9.

Die Pinbelegung zeigt **Abbildung 22**.



EXTENSION 1-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,00 mm.

## Hinweise

- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
Hersteller: Amphenol FCI  
Serie: Minitek®  
1 × Steckergehäuse, Teile-Nr. 90311-040LF  
40 × Crimpkontakt, Teile-Nr. 77138-101LF.
- Die EXTENSION 1-Stiftleiste hat bei RTC6 Ethernet-Karten ein Rastermaß der Pins von 2,00 mm, bei RTC5-Karten und RTC6 PCI-Express-Karten dagegen 2,54 mm.

## Einstellen des Ausgangssignalpegels

Mit dem Lötjumperfeld A auf der Unterseite der RTC6 Ethernet-Karte kann für alle Ausgangssignale der EXTENSION 1-Stiftleiste (DIGITAL OUT 0...DIGITAL OUT 15, LATCH\_OUT, SYNC\_OUT, BUSY\_OUT, VCC\_OUT) ein Signal-Pegel von 5 V oder 3,3 V eingestellt werden, siehe [Kapitel "Lötjumperfeld A – Einstellen des Ausgangssignalpegels an der EXTENSION 1-Stiftleiste"](#), Seite 82.

Als Kontrollmöglichkeit für Benutzer wird der eingestellte Signalpegel kontinuierlich an Pin (35) ausgegeben: Signal VCC\_OUT. VCC\_OUT bezieht sich auf GND, siehe ["Hinweise"](#), Seite 70.

Die maximale Belastung des Signals ist 100 mA.

## 16-Bit-Digital-Ausgang und 16-Bit-Digital-Eingang

Wie bei RTC5, siehe RTC5-Handbuch, Abschnitt "16-Bit-Digital-Eingang und -Ausgang", Seite 51.

Die Signale beziehen sich auf GND, siehe ["Hinweise"](#), Seite 70.

## Synchronisation der Datenübergabe

Wie bei RTC5, siehe RTC5-Handbuch, Abschnitt "Synchronisation der Datenübergabe", Seite 51.

Die Signale beziehen sich auf GND, siehe ["Hinweise"](#), Seite 70.

## BUSY-Status

Das BUSY\_OUT-Signal an Pin (36) ist identisch mit dem BUSY\_OUT-Signal an der LASER-Stiftleiste, siehe [Kapitel 2.2.4 "LASER-Stiftleiste"](#), Seite 70.

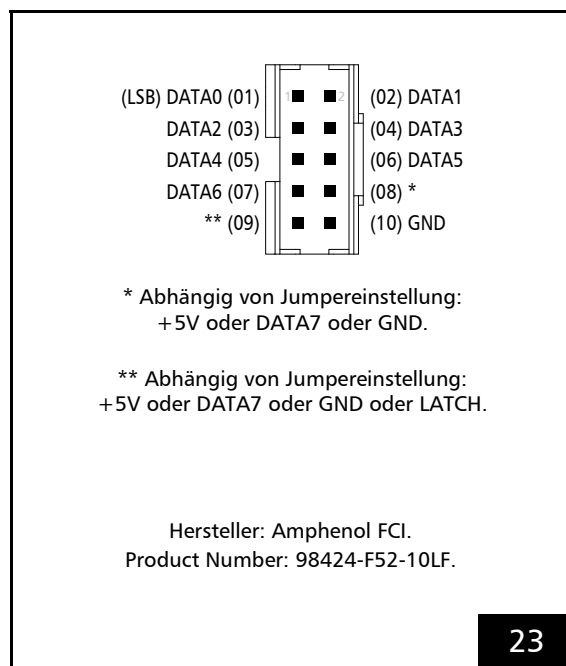
Das Signal bezieht sich auf GND, siehe ["Hinweise"](#), Seite 70.

## 2.2.12 EXT. 2-Stiftleiste

Die EXT. 2-Stiftleiste ist 10-polig. Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in [Abbildung 10](#), Nummer 8.

Sie stellt einen gepufferten 8-Bit-Digital-Ausgang bereit: DATA0...DATA7.

Die Pinbelegung zeigt [Abbildung 23](#).



EXT. 2-Stiftleiste: Pinbelegung. Das Rastermaß der Pins ist 2,00 mm.

Auf der RTC4 ist die funktional entsprechende Stiftleiste mit LASER EXTENSION, auf RTC5- und RTC6 PCI-Express-Karten mit EXTENSION 2 bezeichnet. Neben der Gesamtanzahl der Pins unterscheiden sich die bereitgestellten Signale sowie auch die Pin-Nummern, an denen sie bereitgestellt werden, siehe nachfolgende Tabelle.

Signal	RTC6 Ethernet-Karte, EXT. 2-Stiftleiste 10-polig	RTC5-Karten, EXTENSION 2, 26-polig
DATA0	Pin (01)	Pin (01)
DATA1	Pin (02)	Pin (03)
DATA2	Pin (03)	Pin (05)
DATA3	Pin (04)	Pin (07)
DATA4	Pin (05)	Pin (09)
DATA5	Pin (06)	Pin (11)
DATA6	Pin (07)	Pin (13)
*, siehe in Abbildung 23	Pin (08)	Pin (15)
**, siehe in Abbildung 23	Pin (09)	Pin (17)
GND	Pin (10)	Pin (02)
+5 V	–	Pin (06), (18), (25)
LASER1	–	Pin (22)
LASER2	–	Pin (19)
GND2	–	Pin (23)

#### Hinweise

- Die für den Gegen-Stecker benötigten Teile sind *nicht im Lieferumfang enthalten*:  
Hersteller: Amphenol FCI  
Serie: Minitex®  
1 × Steckergehäuse, Teile-Nr. 90311-010LF  
10 × Crimpkontakt, Teile-Nr. 77138-101LF.
- Pin (08) und Pin (09) können mittels Lötjumper eingestellt werden.
- RTC6 Ethernet-Karten stellen an ihren (10-poligen) EXT. 2-Stiftleisten *keine* Laserausgangssignale (d.h. LASER1 und LASER2) bereit (anders als RTC6 PCI-Express-Karten mit ihren 26-poligen EXTENSION 2-Stiftleisten).

#### Einstellungen mittels Lötjumper

Pin (08) der EXT. 2-Stiftleiste wird über das Lötjumperfeld C konfiguriert, Pin (09) über das Lötjumperfeld B. Beide Lötjumperfelder befinden sich auf der Unterseite der RTC6 Ethernet-Karte, siehe **Abbildung 11**. Für weitere Informationen siehe **Kapitel "Lötjumperfeld B – Konfiguration von Pin (09) der EXT. 2-Stiftleiste"**, Seite 83 und **Kapitel "Lötjumperfeld C – Konfiguration von Pin (08) der EXT. 2-Stiftleiste"**, Seite 84.

#### Hinweise

- Ist das DATA7-Bit an Pin (08) zugewiesen, dann ist der volle 8-Bit Ausgangswert am Ausgang (an den Pins (1) bis Pin (08) der EXT. 2-Stiftleiste) verfügbar.
- Ist Pin (08) auf +5 V (HIGH-Pegel) eingestellt, führt das zu einem Offset von 128 für die Ausgabewerte, d.h. die Ausgabewerte liegen zwischen 128...255.
- Ist Pin (08) auf GND (LOW-Pegel) eingestellt, so sind nur Ausgabewerte zwischen 0 und 127 möglich.
- Das DATA7-Bit kann für andere Zwecke verwendet werden, indem es Pin (09) zugewiesen wird.

#### 8-Bit-Digital-Ausgang

Wie bei RTC5, siehe RTC5-Handbuch, Abschnitt "8-Bit-Digital-Ausgang", Seite 52.



### 2.2.13 Master-Stiftleiste, Slave-Stiftleiste

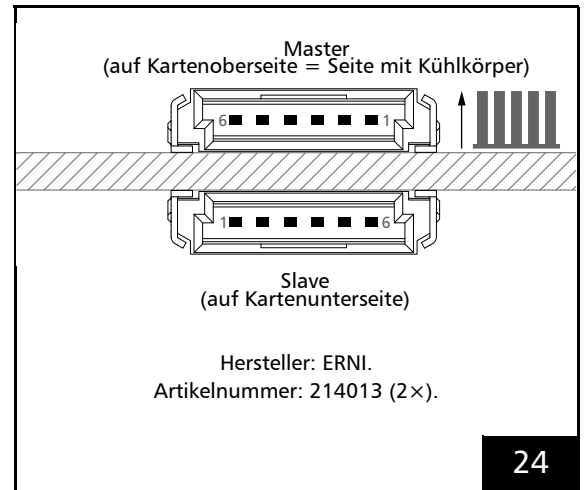
Die Master-Stiftleiste ist 6-polig, siehe [Abbildung 24](#). Sie befindet sich auf der Oberseite der RTC6 Ethernet-Karte, siehe in [Abbildung 10](#), Nummer 10.

Die Slave-Stiftleiste ist 6-polig, siehe [Abbildung 24](#). Sie befindet sich auf der Unterseite der RTC6 Ethernet-Karte, siehe in [Abbildung 10](#), Nummer 16.

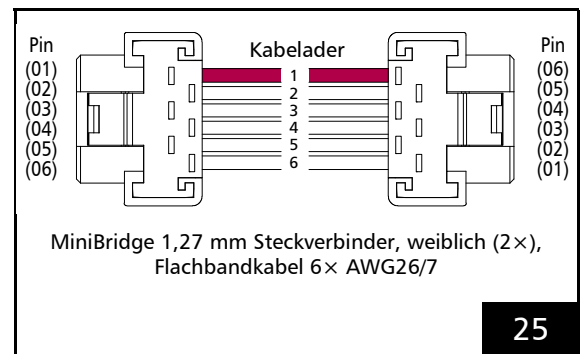
Die beiden Stiftleisten dienen dazu, um die Takte mehrerer RTC6 Ethernet-Karten synchronisieren zu können.

Sollen mehrere RTC6 Ethernet-Karten mit synchroner Taktung betrieben werden, dann müssen diese zuerst paarweise über die Master- und Slave-Stiftleisten miteinander verbunden werden. Verbinden Sie jeweils die Master-Stiftleiste einer Karte mit der Slave-Stiftleiste einer anderen Karte. Ein 50 mm langes Verbindungskabel ist bei SCANLAB (#117241) erhältlich. Benutzer, die ein passendes Kabel selber konfektionieren wollen, finden die notwendigen Angaben in [Abbildung 25](#).

Zu weiteren Informationen siehe [Kapitel 1.9 "Master/Slave-Betrieb"](#), Seite 52.



Master-Stiftleiste und Slave-Stiftleiste. Das Rastermaß der Pins ist 1,27 mm.



Kabel zum Verbinden von Master-Stiftleiste und Slave-Stiftleiste: Anforderungen.

## 2.2.14 Jumpereinstellungen

SCANLAB liefert RTC6 Ethernet-Karten in unterschiedlichen Jumperkonfigurationen aus. Die Jumper sind Verbindungen, die entweder offen oder geschlossen sind. Lötjumperverbindungen können vom Benutzer mit einem Lötkolben nachträglich umkonfiguriert werden. Die Zuweisung eines gewünschten Signals erfolgt durch Schließen des entsprechenden Lötjumpers mittels Lot oder einem 0-Ohm-Widerstand.

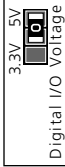

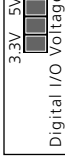


### Vorsicht!

- Nur zulässige Jumpereinstellungen vornehmen. Anderenfalls wird die Karte beschädigt!

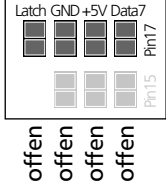
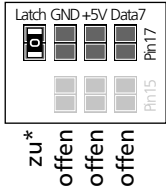
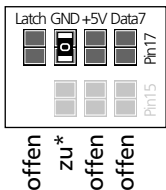
## Lötjumperfeld A – Einstellen des Ausgangssignalpegels an der EXTENSION 1-Stiftleiste

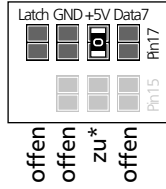
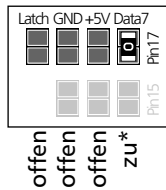
- Position auf der Karte: Unterseite, siehe in **Abbildung 11**, Nummer **12**.
- Zweck: Einstellen des Pegels (5 V oder 3,3 V) für alle Ausgangssignale der EXTENSION 1-Stiftleiste, siehe folgende Tabelle.
- Siehe auch **„Einstellen des Ausgangssignalpegels“**, Seite **79**.

Zulässige JumperEinstellung	Auswirkung
 zu* offen	An der EXTENSION 1-Stiftleiste ist der Ausgangssignalpegel 5 V.
 offen zu*	An der EXTENSION 1-Stiftleiste ist der Ausgangssignalpegel 3,3 V.
 offen offen	An der EXTENSION 1-Stiftleiste werden keine Ausgangssignale ausgegeben.
<p>* Achtung: nur <i>eine einzige</i> Position in diesem Lötjumperfeld schließen. Andere Kombinationen sind nicht zulässig und beschädigen die Karte!</p>	

## Lötjumperfeld B – Konfiguration von Pin (09) der EXT. 2-Stiftleiste

- Position auf der Karte: Unterseite, siehe in **Abbildung 11**, Nummer **13**.
- Zweck: Einstellen des Signals am Pin (09) der EXT. 2-Stiftleiste, siehe folgende Tabelle.
- Siehe auch **“Einstellungen mittels Lötjumper”**, Seite 80.
- Die Konfigurationen von Lötjumperfeld B und Lötjumperfeld C sind nicht voneinander abhängig.
- Auf der RTC6 Ethernet-Karte zeigt die Bedruckung des Lötjumperfeld B 'Pin 17'. Diese wurde bewusst gewählt, um die Konsistenz mit bisherigen RTC-Karten bewahren. Tatsächlich wird bei RTC6 Ethernet-Karten jedoch Pin (09) konfiguriert.

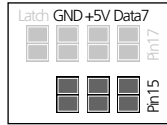
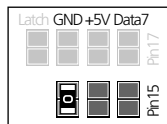
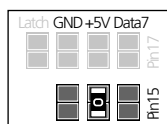
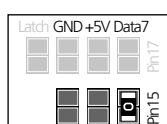
Zulässige Jumbereinstellung	Auswirkung
	An der EXT. 2-Stiftleiste Pin (09) wird ausgegeben: kein Signal.
	An der EXT. 2-Stiftleiste Pin (09) wird ausgegeben: LATCH-Signal.
	An der EXT. 2-Stiftleiste Pin (09) wird ausgegeben: GROUND (Low-Pegel).
<p>* Achtung: nur <i>eine einzige</i> Position in diesem Lötjumperfeld schließen. Andere Kombinationen sind nicht zulässig und beschädigen die Karte!</p>	

Zulässige Jumbereinstellung (Forts.)	Auswirkung (Forts.)
	An der EXT. 2-Stiftleiste Pin (09) wird ausgegeben: + 5 V (High-Pegel).
	An der EXT. 2-Stiftleiste Pin (09) wird ausgegeben: DATA7 <sup>(a)</sup> .
<p>* Achtung: nur <i>eine einzige</i> Position in diesem Lötjumperfeld schließen. Andere Kombinationen sind nicht zulässig und beschädigen die Karte!</p>	

(a) Höchstwertiges Bit (Most Significant Bit, MSB).

## Lötjumperfeld C – Konfiguration von Pin (08) der EXT. 2-Stiftleiste



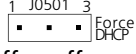
- Position auf der Karte: Unterseite, siehe in **Abbildung 11**, Nummer **14**.
- Zweck: Einstellen des Signals am Pin (08) der EXT. 2-Stiftleiste, siehe folgende Tabelle.
- Siehe auch **“Einstellungen mittels Lötjumper”**, Seite **80**.
- Die Konfigurationen von Lötjumperfeld C und Lötjumperfeld B sind nicht voneinander abhängig.
- Auf der RTC6 Ethernet-Karte zeigt die Bedruckung des Lötjumperfeld C ‘Pin 15’. Diese wurde bewusst gewählt, um die Konsistenz mit bisherigen RTC-Karten bewahren. Tatsächlich wird bei RTC6 Ethernet-Karten jedoch Pin (08) konfiguriert.

Zulässige Jumbereinstellung	Auswirkung
 offen    offen    offen	An der EXT. 2-Stiftleiste Pin (08) wird ausgegeben: Kein Signal.
 zu*    offen    offen	An der EXT. 2-Stiftleiste Pin (08) wird ausgegeben: GROUND (Low-Pegel).
 offen    zu*    offen	An der EXT. 2-Stiftleiste Pin (08) wird ausgegeben: +5 V (High-Pegel).
 offen    offen    zu*	An der EXT. 2-Stiftleiste Pin (08) wird ausgegeben: DATA7 <sup>(a)</sup> .
* Achtung: nur <i>eine einzige</i> Position in diesem Lötjumperfeld schließen. Andere Kombinationen sind nicht zulässig und beschädigen die Karte!	

(a) Höchstwertiges Bit (Most Significant Bit, MSB).

### Steckjumperfeld 'Force DHCP'

- Position auf der Karte: Oberseite, siehe in **Abbildung 10**, Nummer **11**.
- Zweck: siehe folgende Tabelle.

Zulässige JumperEinstellung	Auswirkung
 offen zu	<p>Beim Einschalten der RTC6 Ethernet-Karte soll eine IP-Adresse über DHCP bezogen werden (= "Force DHCP-Stellung").</p> <p>Es werden für alle konfigurierbaren Netzwerkparameter (statische IP, zugehörige Netzmaske, Gateway, UDP- und TCP-Ports) die Default-Einstellungen verwendet (anstatt der im Flash-Memory gespeicherten).</p>
 zu offen*	<p>Wenn benutzerdefinierte Netzwerkeinstellungen abgespeichert sind, werden diese beim Einschalten der RTC6 Ethernet-Karte verwendet.</p>
 offen offen	<p>Wie "zu offen".</p>
* Auslieferungszustand.	

## 2.3 Installation und Inbetriebnahme

### 2.3.1 Hardware-Installation



#### Vorsicht!

- Die RTC6 Ethernet-Karte sollte stets in einer elektrostatisch neutralen Umgebung im mitgelieferten antistatischen Beutel gelagert werden.
- Beachten Sie beim Einbau der Karte die ESD-Schutzvorschriften.
- Berühren Sie die RTC6 Ethernet-Karte nicht an den elektrischen Kontakten.
- Schützen Sie die Karte vor Feuchtigkeit, Staub, korrosiven Dämpfen und mechanischen Belastungen.

Der Einbau der RTC6 Ethernet-Karte in ein abgeschirmtes Gehäuse erfolgt in den folgenden Arbeitsschritten:

- ▶ Nehmen Sie die RTC6 Ethernet-Karte aus dem antistatischen Beutel. Berühren Sie die Karte nicht an den elektrischen Kontakten.
- ▶ Befestigen Sie die RTC6 Ethernet-Karte am dafür vorgesehenen Platz im abgeschirmten Gehäuse.
- ▶ Verbinden Sie die RTC6 Ethernet-Karte und
  - Netzteil mittels Kabel zur Spannungsversorgung
  - Ethernet mittels Ethernet-Kabel
- ▶ Verbinden Sie die RTC6 Ethernet-Karte mittels geeigneter Kabel mit
  - Scan-Kopf
  - Laser
  - gegebenenfalls mit einem Handling-System oder anderen Zusatzgeräten der Gesamtmaschine.
- ▶ Wenn Sie die Signale der RTC6 Ethernet-Karten-Stiftleisten verwenden wollen, dann schließen Sie entsprechende Verbindungskabel an.

### 2.3.2 Software-Installation

RTC6 Ethernet-Karten *benötigen den RTC6-Treiber nicht*<sup>(1)</sup>.

Die Installation und Inbetriebnahme der RTC6 Ethernet-Karte und der zugehörigen Software erfolgt analog zu RTC5-Karten, siehe RTC5-Handbuch, Kapitel 5 "Installation und Inbetriebnahme", Seite 60.

### 2.3.3 Erst-Inbetriebnahme (Test, ohne Benutzeranwendung)

Verbinden Sie die RTC6 Ethernet-Karte mit dem gewünschten Netzwerksegment und schalten Sie die Versorgungsspannung ein. Die grüne LED an der ETH-Buchse (siehe [Kapitel 2.2.7 "ETH-Buchse"](#), Seite 75) sollte nun aufleuchten und evtl. blinken.

Ist keine statische IP-Adresse auf der RTC6 Ethernet-Karte gespeichert, ist ein DHCP-Server notwendig, um eine dynamische IP-Adresse zu beziehen. In diesem Fall sollte die gelbe LED nach einigen Sekunden aufleuchten. Damit wird der erfolgreiche Bezug einer IP-Adresse signalisiert.

Wenn eine statische IP-Adresse auf der RTC6 Ethernet-Karte abgespeichert ist, leuchtet die gelbe LED sofort nach Einschalten der RTC6 Ethernet-Karte auf.

Wichtig: Es ist möglich, dass die auf der RTC6 Ethernet-Karte gespeicherte IP-Konfiguration für das angeschlossene Netzwerksegment nicht gültig ist. Auch in diesem Fall leuchtet die gelbe LED auf. Jedoch ist die RTC6 Ethernet-Karte im Netzwerk aber nicht ansprechbar.

(1) Siehe Fußnote auf [Seite 64](#).

## 2.4 Hinweise zur Migration und Neu-Programmierung von RTC6-Anwendungen

Anders als RTC6 PCI-Express-Karten müssen RTC6 Ethernet-Karten manuell in die RTC6-Kartenverwaltung eingetragen werden, siehe [Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung", Seite 89](#). Im Anschluss können sie prinzipiell genauso wie RTC6 PCI-Express-Karten verwendet werden.

Die Kommunikation über Ethernet ist im Allgemeinen unsicherer als eine PCIe-Bus-Verbindung innerhalb eines PCs. Daher sollte mehr Augenmerk auf eventuell auftretende Fehler und deren Fehlerbehandlung gelegt werden, als dies evtl. bei RTC6 PCI-Express-Karten getan wurde (siehe `get_last_error`).

### 2.4.1 RTC6 Ethernet-Karten im Netzwerk finden und Eigenschaften abfragen

#### Fall 1: Die IP-Adresse der RTC6 Ethernet-Karte ist bekannt

Verwenden Sie einen entsprechenden Quellcodeabschnitt wie folgt:

```
card_no = eth_assign_card_ip(ip, 0); // 0: assign to first free number
Result = select_rtc(card_no);       // If Result equals CardNo: Success
...
```

#### Fall 2: Die IP-Adresse der RTC6 Ethernet-Karte ist nicht bekannt

Verwenden Sie einen entsprechenden Quellcodeabschnitt wie folgt:

```
result = eth_search_cards(eth_convert_string_to_ip("192.168.250.1"),
eth_convert_string_to_ip("255.255.255.0"));
if (result != 0)
{
    card_no = eth_assign_card(1, 0); // take first found card, assign to first free number
    result = select_rtc(card_no);
    if (result == card_no)
    {
        ip = eth_get_ip(); // now IP address is known
        ... // do anything
    }
}
```

## 2.5 RTC6 Ethernet-Kartenbefehle und Funktionen

### 2.5.1 Hinweise zum Arbeiten mit IP-Adressen

Bei den RTC6-Befehlen werden alle IP-Adressen (stets IPv4, IPv6 wird nicht unterstützt) als Dezimalwert im Big-Endian-Format ("Big-Endian Byte-Reihenfolge") angegeben.

Beispielsweise wird für die IP-Adresse "192.168.250.1" also "33204416" angegeben, siehe dazu folgende Tabelle.

Format	IP-Adresse	Hex-Wert	Dezimalwert
Little-Endian	192.168.250.1 (a)	0xC0A8FA01	3232299521
Big-Endian	1.250.168.192	0x01FAA8C0	33204416

(a) Gewöhnliche Dezimalpunktschreibweise (engl. dotted decimal notation).

**eth\_convert\_ip\_to\_string** wandelt die IP-Adresse in Big-Endian Byte-Reihenfolge in die gewöhnliche Dezimalpunktschreibweise um.

**eth\_convert\_string\_to\_ip** wandelt die gewöhnliche Dezimalpunktschreibweise in die IP-Adresse in Big-Endian Byte-Reihenfolge um.

#### Hinweise

- Die Verwendung von **eth\_convert\_ip\_to\_string** bietet sich im Zusammenhang mit **eth\_get\_ip**, **eth\_get\_card\_info** oder anderen Befehlen wie **eth\_get\_static\_ip** etc. an. Die Parameterrückgabewerte von **eth\_get\_static\_ip** sind in Big-Endian Byte-Reihenfolge.
- Die Verwendung von **eth\_convert\_string\_to\_ip** bietet sich im Zusammenhang mit **eth\_search\_cards**, **eth\_search\_cards\_range** und **eth\_set\_static\_ip** an. Bei diesen Befehlen sind die Parameterwerte in Big-Endian Byte-Reihenfolge anzugeben. Gleiches gilt auch für **eth\_get\_card\_info** und **eth\_get\_card\_info\_search**.

### 2.5.2 Über die Suche von RTC6 Ethernet-Karten

Mit **eth\_search\_cards** kann nach allen im Netzwerk verfügbaren RTC6 Ethernet-Karten gesucht werden. Soll die Suche auf einen bestimmten Adressbereich beschränkt sein, ist **eth\_search\_cards\_range** zu benutzen.

Die Daten aller RTC6 Ethernet-Karten (die innerhalb eines konfigurierbaren Timeouts, siehe **eth\_set\_search\_cards\_timeout**) geantwortet haben, werden in die Suchergebnisliste (s.u.) eingetragen.

Mit einer Kartensuche können RTC6 Ethernet-Karten mit unbekannter IP-Adresse im Netzwerk identifiziert werden (z. B. weil sie diese durch einen DHCP-Server dynamisch zugewiesen bekommen haben).

Die gefundenen RTC6 Ethernet-Karten werden in einer eigenständigen, temporären Liste eingetragen, der Suchergebnisliste.

Index	Eintrag in der Suchergebnisliste
1	Informationen <sup>(a)</sup> zu RTC6 Ethernet-Karte 1
2	Informationen <sup>(a)</sup> zu RTC6 Ethernet-Karte 2
n	Informationen <sup>(a)</sup> zu RTC6 Ethernet-Karte n

(a) IP-Adresse, Seriennummer, Verbindungsstatus, etc., siehe **eth\_get\_card\_info\_search**.

Für einen angegebenen Suchergebnislisten-Index gibt **eth\_get\_card\_info\_search** die verfügbaren Informationen zu der RTC6 Ethernet-Karte zurück, **eth\_get\_ip\_search** dagegen nur die IP-Adresse und **eth\_get\_serial\_search** nur die Seriennummer.

Die Anzahl der gefundenen RTC6 Ethernet-Karten wird zum einen bereits durch **eth\_search\_cards** zurückgegeben, zum anderen aber auch jederzeit erneut durch den Aufruf von **eth\_found\_cards** (ohne dass die Suche im Netzwerk wiederholt werden muss).

Mehrere aufeinanderfolgende Aufrufe von **eth\_search\_cards** können unterschiedliche Ergebnisse liefern, je nachdem, wie viele RTC6 Ethernet-Karten im Netzwerk verfügbar sind, und in welcher zeitlichen Reihenfolge die Antworten eintreffen.



### 2.5.3 Über die RTC6-Kartenverwaltung

RTC6-Karten werden (wie RTC4- und RTC5-Karten) über eine eindeutige Nummer angesprochen, so z.B. bei Multi-Board-Befehlen<sup>(1)</sup> und **select\_rtc**<sup>(2)</sup>.

Deswegen müssen RTC6 PCI-Express-Karten und RTC6 Ethernet-Karten in der RTC6-Kartenverwaltung eingetragen (s.u.) sein (d.h. RTC6-Karten, die nicht eingetragen sind, können auch nicht verwendet werden).

Die RTC6-Kartenverwaltung ist eine RTC6-DLL<sup>(3)</sup>-interne Liste aus 255 RTC6-Karteneinträgen, siehe nachfolgende Tabelle.

Index	RTC6-Karteneintrag
1	RTC6 PCI-Express-Karte 1
2	RTC6 PCI-Express-Karte 2
3	RTC6 PCI-Express-Karte 3
4	"Keine Karte"
5	Informationen zur RTC6 Ethernet-Karte <sup>(a)</sup>
6	Informationen zur RTC6 Ethernet-Karte <sup>(a)</sup>
7	"Keine Karte"
...	...
42	Informationen zur RTC6 Ethernet-Karte <sup>(a)</sup>
43	"Keine Karte"
...	...
255	"Keine Karte"

(a) IP-Adresse, Seriennummer, Verbindungsstatus, etc., siehe **eth\_get\_card\_info\_search**.

(1) Alle Befehle mit Präfix n\_.

(2) Dieser Befehl definiert diejenige Karte, die standardmäßig die Single-Board-Befehle ausführt, wenn mehrere RTC6-Karten verfügbar sind.

(3) RTC6DLL.dll, RTC6DLLx64.dll.

Ein RTC6-Karteneintrag kann über seinen Index angesprochen werden. RTC6-Karteneinträge können sein: "Keine Karte", RTC6 PCI-Express-Karte, RTC6 Ethernet-Karte (mit fest zugeordneter IP-Adresse).

Die Liste beginnt mit RTC6 PCI-Express-Karten, sofern solche im PC vorhanden sind.

RTC6 PCI-Express-Karten werden durch die RTC6-DLL automatisch fortlaufend nummeriert. Benutzer können ihre Nummerierung nicht ändern.

**rtc6\_count\_cards** gibt die Anzahl der RTC6 PCI-Express-Karten in der RTC6-Kartenverwaltung zurück.

Im Gegensatz dazu tragen Benutzer selber jede RTC6 Ethernet-Karte einzeln in die RTC6-Kartenverwaltung ein. Ist deren IP-Adresse bekannt, kann **eth\_assign\_card\_ip** verwendet werden. Mit diesem Befehl wird die IP-Adresse an dem gewünschten Index in der RTC6-Kartenverwaltung (aus dem Bereich [RTC6 PCI-Express-Kartenanzahl+1]...255) angegeben. An diesem Index darf noch keine RTC6 PCI-Express-Karte oder RTC6 Ethernet-Karte stehen.

Ist die IP-Adresse einer RTC6 Ethernet-Karte nicht bekannt, kann alternativ zu **eth\_assign\_card\_ip** der Befehl **eth\_assign\_card** verwendet werden. Dazu muss aber zuvor eine Suchergebnisliste mit **eth\_search\_cards** oder **eth\_search\_cards\_range** erzeugt worden sein (Detailinformationen zu einer bestimmten Kartennummer darin gibt **eth\_get\_card\_info\_search** zurück). Bei **eth\_assign\_card** wird die gewünschte Kartennummer aus der Suchergebnisliste (= Parameter **SearchNo**) und der gewünschte Index in der RTC6-Kartenverwaltung (aus dem Bereich [RTC6 PCI-Express-Kartenanzahl+1]...255) angegeben. An diesem Index darf noch keine RTC6 PCI-Express-Karte oder RTC6 Ethernet-Karte eingetragen sein.

RTC6 Ethernet-Karten können also beliebig in der RTC6-Kartenverwaltung angeordnet sein. Deren Nummern müssen auch nicht fortlaufend sein. **eth\_max\_card** gibt den höchsten Index<sup>(4)</sup> zurück, an dem eine RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung eingetragen ist. **eth\_count\_cards** gibt die Gesamtanzahl der eingetragenen RTC6 Ethernet-Karten zurück.

Mit **get\_card\_type** kann der eingetragene Kartentyp abgefragt werden: 0 = "Keine Karte", 1 = RTC6 PCI-Express-Karte, 2 = RTC6 Ethernet-Karte.

(4) Nicht: Kartenanzahl!

## 2.5.4 Verbindung zur RTC6 Ethernet-Karte prüfen

Die Kommunikation über Ethernet ist im Allgemeinen unverlässlicher als eine PCIe-Bus-Verbindung innerhalb eines PCs (einfaches Beispiel: Ethernet-Kabel ist nicht gesteckt). Mit **eth\_check\_connection** kann einfach geprüft werden, ob die RTC6 Ethernet-Karte antwortet und somit auch die Ethernet-Verbindung noch besteht.

## 2.5.5 Befehlssatz für die RTC6 Ethernet-Karte

Die Befehle sind alphabetisch geordnet.

	<b>eth_assign_card</b> .....	91
	<b>eth_assign_card_ip</b> .....	92
(n_)	<b>eth_check_connection</b> .....	93
	<b>eth_convert_ip_to_string</b> .....	94
	<b>eth_convert_string_to_ip</b> .....	95
	<b>eth_count_cards</b> .....	96
	<b>eth_found_cards</b> .....	97
	<b>eth_get_card_info</b> .....	98
	<b>eth_get_card_info_search</b> .....	99
(n_)	<b>eth_get_com_timeouts</b> .....	100
(n_)	<b>eth_get_error</b> .....	101
	<b>eth_get_ip</b> .....	102
	<b>eth_get_ip_search</b> .....	102
(n_)	<b>eth_get_last_error</b> .....	103
(n_)	<b>eth_get_port_numbers</b> .....	105
	<b>eth_get_serial_search</b> .....	105
(n_)	<b>eth_get_static_ip</b> .....	106
	<b>eth_max_card</b> .....	107
	<b>eth_remove_card</b> .....	108
	<b>eth_search_cards</b> .....	109
	<b>eth_search_cards_range</b> .....	110
(n_)	<b>eth_set_com_timeouts</b> .....	111
(n_)	<b>eth_set_port_numbers</b> .....	112
	<b>eth_set_search_cards_timeout</b> .....	113
(n_)	<b>eth_set_static_ip</b> .....	114
(n_)	<b>get_card_type</b> .....	115

Kontrollbefehl	<b>eth_assign_card</b>
Funktion	Trägt die RTC6 Ethernet-Karten-IP-Adresse <code>SearchNo</code> aus der Suchergebnisliste am RTC6-Kartenverwaltungsindex = <code>CardNo</code> ein.
Aufruf	<code>Result = eth_assign_card( SearchNo, CardNo )</code>
Rückgabe	<p>Fehlercode. Als 32-Bit-Wert mit Vorzeichen.</p> <p>–2 Fehler: Der Eintrag kann nicht vorgenommen werden. An diesem Index ist bereits eine RTC6 Ethernet-Karte eingetragen.</p> <p>–1 Fehler: Der Eintrag kann nicht vorgenommen werden. Dieser Index ist bereits mit einer RTC6 PCI-Express-Karte besetzt.</p> <p>0 Fehler: Der Eintrag kann nicht vorgenommen werden. <code>SearchNo</code> ist ungültig (0, &gt; <b>eth_found_cards</b>). Oder: <code>CardNo</code> ist ungültig (<math>1 \leq \text{CardNo} \leq \text{rtc6\_count\_cards}</math>, &gt; 255).</p> <p><code>n (= CardNo)</code> Erfolg: Der Kartennummern-Eintrag wurde vorgenommen.</p>
Parameter	<p><code>SearchNo</code> Index der Suchergebnisliste. Zulässiger Wertebereich: [1...(Rückgabewert &gt;0 von <b>eth_found_cards</b>)].</p> <p><code>CardNo</code> Index der RTC6-Kartenverwaltung, bei dem die RTC6 Ethernet-Karte eingetragen werden soll. Zulässiger Wertebereich: [(<b>rtc6_count_cards</b> + 1)...255] oder 0.</p> <p>Alle als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_assign_card</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zum Erzeugen einer Suchergebnisliste werden <b>eth_search_cards</b> und <b>eth_search_cards_range</b> verwendet. Siehe auch Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten", Seite 88.</li> <li>• Am angegebenen Index <code>CardNo</code> darf noch keine RTC6 Ethernet-Karte oder RTC6 PCI-Express-Karte eingetragen sein. Eine RTC6 Ethernet-Karte muss zuvor explizit mit <b>eth_remove_card</b> entfernt werden. RTC6 PCI-Express-Karten können nicht entfernt werden.</li> <li>• Im Erfolgsfall wird <b>eth_count_cards</b> automatisch um 1 erhöht.</li> <li>• Mit <code>CardNo = 0</code> wird die RTC6 Ethernet-Karte an <b>rtc6_count_cards</b> + <b>eth_count_cards</b> + 1 eingetragen. Dadurch kann automatisch eine lückenlose Kartenummerierung erzeugt werden (siehe auch <b>eth_remove_card</b>). <code>CardNo = 0</code> und freigewählte Zuweisung von RTC6 Ethernet-Karten zu Indices sollten nicht zusammen verwendet werden.</li> <li>• Bei allen Fehlern wird der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> erzeugt.</li> <li>• Siehe auch Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung", Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_assign_card_ip</b> , <b>eth_count_cards</b> , <b>eth_found_cards</b> , <b>eth_max_card</b> , <b>eth_search_cards</b> , <b>eth_search_cards_range</b> , <b>eth_remove_card</b> , <b>rtc6_count_cards</b>

Kontrollbefehl	<b>eth_assign_card_ip</b>
Funktion	Trägt eine RTC6 Ethernet-Karte mit der IP-Adresse <code>Ip</code> am Index = <code>CardNo</code> in die RTC6-Kartenverwaltung ein.
Aufruf	<code>Result = eth_assign_card_ip( Ip, CardNo )</code>
Rückgabe	<p>Fehlercode. Als 32-Bit-Wert mit Vorzeichen.</p> <p>–2 Fehler: Der Eintrag kann nicht vorgenommen werden. An diesem Index ist bereits eine RTC6 Ethernet-Karte eingetragen.</p> <p>–1 Fehler: Der Eintrag kann nicht vorgenommen werden. An diesem Index ist bereits eine RTC6 PCI-Express-Karte eingetragen.</p> <p>0 Fehler: Der Eintrag kann nicht vorgenommen werden. <code>CardNo</code> ist ungültig (<math>1 \leq \text{CardNo} \leq \text{rtc6\_count\_cards}</math>, <math>&gt; 255</math>).</p> <p><code>n (= CardNo)</code> Erfolg: Der Eintrag wurde vorgenommen.</p>
Parameter	<p><code>Ip</code> IP-Adresse der RTC6 Ethernet-Karte.</p> <p><code>CardNo</code> Index der RTC6-Kartenverwaltung, an dem die RTC6 Ethernet-Karte eingetragen werden soll. Zulässiger Wertebereich: <math>[(\text{rtc6\_count\_cards} + 1) \dots 255]</math> oder 0. Alle als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_assign_card_ip</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Am angegebenen Index <code>CardNo</code> darf noch keine RTC6 Ethernet-Karte oder RTC6 PCI-Express-Karte eingetragen sein. Eine RTC6 Ethernet-Karte muss zuvor explizit mit <b>eth_remove_card</b> entfernt werden. RTC6 PCI-Express-Karten können nicht entfernt werden.</li> <li>• Im Erfolgsfall wird <b>eth_count_cards</b> automatisch um 1 erhöht.</li> <li>• Mit <code>CardNo = 0</code> wird die RTC6 Ethernet-Karte an <b>rtc6_count_cards + eth_count_cards + 1</b> eingetragen. Dadurch kann automatisch eine lückenlose Kartenummerierung erzeugt werden (siehe auch <b>eth_remove_card</b>). <code>CardNo = 0</code> und freigewählte Zuweisung von RTC6 Ethernet-Karten zu Indices sollten nicht zusammen verwendet werden.</li> <li>• Bei allen Fehlern wird der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> erzeugt.</li> <li>• Die RTC6 Ethernet-Karte wird auch dann in die RTC6-Kartenverwaltung eingetragen, wenn bei deren Initialisierung ein Ethernet-Fehler aufgetreten ist. Vor einem erneuten Initialisierungsversuch muss diese RTC6 Ethernet-Karte wieder explizit aus der RTC6-Kartenverwaltung entfernt werden.</li> <li>• Siehe auch Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung", Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_assign_card</b> , <b>eth_count_cards</b> , <b>eth_remove_card</b>

Kontrollbefehl	<b>eth_check_connection</b>
Funktion	Prüft, ob die RTC6 Ethernet-Karte antwortet.
Aufruf	CheckOK = eth_check_connection()
Rückgabe	Antwortverhalten. Als 32-Bit-Wert ohne Vorzeichen.  0:               Nicht OK. Die RTC6 Ethernet-Karte hat keine Ethernet-Verbindung oder sie antwortet nicht.  1:               OK. Die RTC6 Ethernet-Karte hat eine Ethernet-Verbindung und sie antwortet.
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• Bei CheckOK = 0 kann mit <b>get_last_error</b>, <b>eth_get_last_error</b> und <b>eth_get_error</b> die mögliche Fehlerursache abgefragt werden.</li> <li>• Wenn die Karte keine RTC6 Ethernet-Karte ist, dann wird der <b>get_last_error</b>-Return-code RTC6_TYPE_REJECTED gesetzt und <b>eth_check_connection</b> wird nicht ausgeführt.</li> <li>• Siehe auch <b>Kapitel 2.5.4 "Verbindung zur RTC6 Ethernet-Karte prüfen", Seite 90.</b></li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 608, OUT 608, RBF 611.
Verweise	<b>eth_get_error</b> , <b>eth_get_last_error</b> , <b>get_last_error</b>

Kontrollbefehl	<b>eth_convert_ip_to_string</b>	
Funktion	Wandelt eine IP-Adresse in Big-Endian Byte-Reihenfolge in die gewöhnliche Dezimalpunktschreibweise (Beispiel: "192.168.250.1") um.	
Aufruf	<code>eth_convert_ip_to_string( Ip, IpString )</code>	
Rückgabe	Keine.	
Parameter	Ip	Umzuwandelnde IP-Adresse in Big-Endian Byte-Reihenfolge. Als 32-Bit-Wert ohne Vorzeichen.
	IpString	Umgewandelte IP-Adresse. Als Pointer (bei C und C++ Datentyp ULONG_PTR, d.h. 32-Bit- bzw. 64-Bit-Wert ohne Vorzeichen) auf ein Array von vier 32-Bit-Werten ohne Vorzeichen, in dem die umgewandelte IP-Adresse als Zeichenkette in gewöhnlicher Dezimalpunktschreibweise zu finden ist (synonym zu einem Zeichen-Array der Länge 16, Low Byte first, mit abschließendem \0).
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_convert_ip_to_string</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zur Speicherung der umgewandelten IP-Adresse muss vom Anwenderprogramm an der über IpString angegebenen Adresse ein Speicherbereich der Größe <math>4 \times 4</math> Byte bereitgestellt werden.</li> <li>• Beispiel: Ip = 33204416, IpString[0] = "192.", IpString[1] = "168.", IpString[2] = "250.", IpString[3] = "1\0xx".</li> <li>• Die Verwendung von <b>eth_convert_ip_to_string</b> bietet sich im Zusammenhang mit <b>eth_get_static_ip</b> an.</li> <li>• Siehe auch Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen", Seite 88.</li> </ul>	
RTC4→RTC6	Neuer Befehl.	
RTC5→RTC6	Neuer Befehl.	
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.	
Verweise	<b>eth_convert_string_to_ip</b> , <b>eth_get_static_ip</b>	

Kontrollbefehl	<b>eth_convert_string_to_ip</b>
Funktion	Wandelt eine IP-Adresse in gewöhnlicher Dezimalpunktschreibweise (Beispiel: "192.168.250.1") in die Big-Endian Byte-Reihenfolge um.
Aufruf	<code>ResultIp = eth_convert_string_to_ip( &amp;IpString )</code>
Rückgabe	<p>IP-Adresse. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0:               IpString ist fehlerhaft. &gt;0:             Umgewandelte IP-Adresse in Big-Endian Byte-Reihenfolge.</p>
Parameter	<p>IpString       Umzuwandelnde IP-Adresse in gewöhnlicher Dezimalpunktschreibweise. Als String (char array). Zeiger auf einen nullterminierten ANSI-String. Max. 16 Bytes.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_convert_string_to_ip</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Beispiel: Beim Parameter <code>IpString = "192.168.250.1"</code> ist der Rückgabewert <code>ResultIp = 33204416</code>. Siehe auch Hinweis bei <a href="#">eth_search_cards</a>.</li> <li>• Die Verwendung von <b>eth_convert_string_to_ip</b> bietet sich im Zusammenhang mit <a href="#">eth_search_cards</a>, <a href="#">eth_search_cards_range</a> und <a href="#">eth_set_static_ip</a> an, sowie bei <a href="#">eth_get_card_info</a> und <a href="#">eth_get_card_info_search</a>.</li> <li>• Siehe auch <a href="#">Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen"</a>, Seite 88.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<a href="#">eth_convert_ip_to_string</a> , <a href="#">eth_get_card_info</a> , <a href="#">eth_get_card_info_search</a> , <a href="#">eth_search_cards</a> , <a href="#">eth_search_cards_range</a> , <a href="#">eth_set_static_ip</a>

Kontrollbefehl	<b>eth_count_cards</b>
Funktion	Liefert die Anzahl der RTC6 Ethernet-Karten zurück, die in der RTC6-Kartenverwaltung eingetragen sind.
Aufruf	Result = eth_count_cards()
Rückgabe	Anzahl der RTC6 Ethernet-Karten, die in der RTC6-Kartenverwaltung eingetragen sind. Als 32-Bit-Wert ohne Vorzeichen.
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_count_cards</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Im Gegensatz zu <b>rtc6_count_cards</b> kann aus dem Rückgabewert nicht unbedingt auf den höchsten Index aller RTC6-Karten geschlossen werden (siehe <b>eth_assign_card</b> und <b>eth_remove_card</b> mit CardNo = 0, siehe auch <b>eth_max_card</b>).</li> <li>• Es ist nur sichergestellt, dass jede RTC6 Ethernet-Kartennummer größer ist als die größte RTC6 PCI-Express-Kartennummer, siehe <b>rtc6_count_cards</b>.</li> <li>• <b>eth_count_cards</b> kann durchaus größer als <b>eth_found_cards</b> sein. Das ist der Fall, wenn RTC6 Ethernet-Karten explizit über ihre IP-Adressen in die RTC6-Kartenverwaltung eingetragen worden sind, siehe <b>eth_assign_card_ip</b>.</li> <li>• Nach der erfolgreicher Ausführung von <b>eth_assign_card</b> wird <b>eth_count_cards</b> automatisch um 1 erhöht.</li> <li>• Nach der erfolgreicher Ausführung von <b>eth_remove_card</b> wird <b>eth_count_cards</b> automatisch um 1 verringert.</li> <li>• Siehe auch Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung", Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_assign_card</b> , <b>eth_assign_card_ip</b> , <b>eth_found_cards</b> , <b>eth_max_card</b> , <b>eth_remove_card</b> , <b>rtc6_count_cards</b>



Kontrollbefehl	<b>eth_found_cards</b>
Funktion	Liefert die Anzahl der RTC6 Ethernet-Karten zurück, die in der Suchergebnisliste sind.
Aufruf	<code>Result = eth_found_cards()</code>
Rückgabe	<p>Anzahl der RTC6 Ethernet-Karten in der Suchergebnisliste. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0: Es sind keine RTC6 Ethernet-Karten in der Suchergebnisliste enthalten. Möglicherweise wurde die Suche noch nicht durchgeführt.</p> <p>n: Anzahl der RTC6 Ethernet-Karten, die in der Suchergebnisliste enthalten sind.</p>
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_found_cards</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• <b>eth_found_cards</b> stimmt <i>nicht</i> unbedingt überein mit <ul style="list-style-type: none"> <li>– der RTC6 Ethernet-Kartenanzahl in der RTC6-Kartenverwaltung</li> <li>– der Gesamtzahl aus RTC6 Ethernet-Karten und RTC6 PCI-Express-Karten</li> </ul> </li> <li>• Die Suchergebnisliste wird mit <b>eth_search_cards</b> oder <b>eth_search_cards_range</b> erzeugt. Deren Rückgabewert stimmt mit dem von <b>eth_found_cards</b> überein. Siehe auch <b>Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten"</b>, Seite 88.</li> <li>• Informationen zu einer bestimmten RTC6 Ethernet-Karte in der Suchergebnisliste können mit <b>eth_get_card_info_search</b> abgefragt werden.</li> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</b>, Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_get_card_info_search</b> , <b>eth_search_cards</b> , <b>eth_search_cards_range</b>

Kontrollbefehl	<b>eth_get_card_info</b>
Funktion	Liefert Informationen zu einer bestimmten RTC6 Ethernet-Karte zurück, die in der RTC6-Kartenverwaltung eingetragen und einmal in Besitz genommen war oder ist.
Aufruf	<code>eth_get_card_info( CardNo, Ptr )</code>
Rückgabe	Keine.
Parameter	<div>CardNo</div> <p>Index der RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung. Als 32-Bit-Wert ohne Vorzeichen.</p> <div>Ptr</div> <p>Karteninformationen. Als Pointer (bei C und C++ Datentyp ULONG_PTR, d.h. 32-Bit- bzw. 64-Bit-Wert ohne Vorzeichen) auf ein Array von 16 32-Bit-Werten ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_get_card_info</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zur Speicherung der Karteninformationen muss vom Anwenderprogramm (an der über <code>Ptr</code> angegebenen Adresse) ein Speicherbereich der Größe <math>16 \times 4</math> Byte bereitgestellt werden.</li> <li>• Karteninformationen: <ul style="list-style-type: none"> <li>[0] = Firmwareversion</li> <li>[1] = Seriennummer</li> <li>[2] = IP-Adresse (Big-Endian-Format)</li> <li>[3] = MAC-Adresse (untere 4 Bytes)</li> <li>[4] = MAC-Adresse (obere 2 Bytes)</li> <li>[5] = Ist in Besitz genommen (1 oder 0)</li> <li>[6] = IP-Adresse des verbundenen PCs (Big-Endian-Format)</li> <li>[7] = Force DHCP (1 oder 0)</li> <li>[8] = Statische IP-Adresse (Big-Endian-Format)</li> <li>[9] = Statische Netzmaske (Big-Endian-Format)</li> <li>[10] = Statisches Gateway (Big-Endian-Format)</li> <li>[11] = UDP-Port für Kartensuchvorgänge</li> <li>[12] = UDP-Port für exklusive Verbindungen</li> <li>[13] = TCP-Port für eine exklusive Verbindung</li> <li>[14] = Reserviert</li> <li>[15] = Reserviert</li> </ul> </li> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</b>, Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_count_cards</b> , <b>eth_get_card_info_search</b> , <b>eth_max_card</b> , <b>get_card_type</b>

Kontrollbefehl	<b>eth_get_card_info_search</b>
Funktion	Gibt Informationen zu einer bestimmten RTC6 Ethernet-Karte in der Suchergebnisliste zurück.
Aufruf	<code>eth_get_card_info_search( SearchNo, Ptr )</code>
Rückgabe	Keine.
Parameter	<p><b>SearchNo</b>      Index der Suchergebnisliste. Als 32-Bit-Wert ohne Vorzeichen.</p> <p><b>Ptr</b>             Karteninformationen. Als Pointer (bei C und C++ Datentyp <code>ULONG_PTR</code>, d.h. 32-Bit- bzw. 64-Bit-Wert ohne Vorzeichen) auf ein Array von 16 32-Bit-Werten ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_get_card_info_search</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Die Suchergebnisliste wird mit <b>eth_search_cards</b> oder <b>eth_search_cards_range</b> erzeugt. Deren Rückgabewert stimmt mit dem von <b>eth_found_cards</b> überein. Siehe auch <b>Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten"</b>, Seite 88.</li> <li>• Für <code>SearchNo = 0</code> und <code>SearchNo &gt; eth_found_cards</code> wird eine Default-Information zurückgegeben. Außerdem wird der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> erzeugt.</li> <li>• Wurde noch gar keine Suche ausgeführt, dann gilt automatisch: <code>SearchNo &gt; eth_found_cards</code>.</li> <li>• Zur Speicherung der Karteninformationen muss vom Anwenderprogramm (an der über <code>Ptr</code> angegebenen Adresse) ein Speicherbereich der Größe <math>16 \times 4</math> Byte bereitgestellt werden.</li> <li>• Karteninformationen: <ul style="list-style-type: none"> <li>[0] = Firmwareversion</li> <li>[1] = Seriennummer</li> <li>[2] = IP-Adresse (Big-Endian-Format)</li> <li>[3] = MAC-Adresse (untere 4 Bytes)</li> <li>[4] = MAC-Adresse (obere 2 Bytes)</li> <li>[5] = Ist in Besitz genommen (1 oder 0)</li> <li>[6] = IP-Adresse des verbundenen PCs (Big-Endian-Format)</li> <li>[7] = Force DHCP (1 oder 0)</li> <li>[8] = Statische IP-Adresse (Big-Endian-Format)</li> <li>[9] = Statische Netzmaske (Big-Endian-Format)</li> <li>[10] = Statisches Gateway (Big-Endian-Format)</li> <li>[11] = UDP-Port für Kartensuchvorgänge</li> <li>[12] = UDP-Port für exklusive Verbindungen</li> <li>[13] = TCP-Port für eine exklusive Verbindung</li> <li>[14] = Reserviert</li> <li>[15] = Reserviert</li> </ul> </li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_found_cards</b> , <b>eth_get_card_info</b> , <b>eth_search_cards</b> , <b>eth_search_cards_range</b>

Kontrollbefehl	<b>eth_get_com_timeouts</b>												
Funktion	Liefert Timing-Informationen zu einer bestimmten RTC6 Ethernet-Karte zurück, die in der RTC6-Kartenverwaltung eingetragen ist.												
Aufruf	<code>eth_get_com_timeouts( &amp;AcquireTimeout, &amp;AcquireMaxRetries, &amp;SendRecvTimeout, &amp;SendRecvMaxRetries, &amp;KeepAlive, &amp;KeepInterval )</code>												
Rückgabe	Keine.												
Parameter- rückgabe	<table border="0"> <tr><td>AcquireTimeout</td><td>Reserviert.</td></tr> <tr><td>AcquireMaxRetries</td><td>Reserviert.</td></tr> <tr><td>SendRecvTimeout</td><td>Reserviert.</td></tr> <tr><td>SendRecvMaxRetries</td><td>Reserviert.</td></tr> <tr><td>KeepAlive</td><td>Reserviert.</td></tr> <tr><td>KeepInterval</td><td>Reserviert.</td></tr> </table> <p>Jeweils als Pointer auf einen 32-Bit-Wert ohne Vorzeichen. Alle Werte sind jeweils 0, wenn die Karte keine RTC6 Ethernet-Karte ist.</p>	AcquireTimeout	Reserviert.	AcquireMaxRetries	Reserviert.	SendRecvTimeout	Reserviert.	SendRecvMaxRetries	Reserviert.	KeepAlive	Reserviert.	KeepInterval	Reserviert.
AcquireTimeout	Reserviert.												
AcquireMaxRetries	Reserviert.												
SendRecvTimeout	Reserviert.												
SendRecvMaxRetries	Reserviert.												
KeepAlive	Reserviert.												
KeepInterval	Reserviert.												
Hinweise	<ul style="list-style-type: none"> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</b>, Seite 89.</li> <li>• Zum Ändern der Einstellungen wird <b>eth_set_com_timeouts</b> verwendet (normalerweise nur im Kontext einer Problemlösung und nur auf Aufforderung durch SCANLAB).</li> </ul>												
RTC4→RTC6	Neuer Befehl.												
RTC5→RTC6	Neuer Befehl.												
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.												
Verweise	<b>eth_set_com_timeouts</b>												

Kontrollbefehl	<b>eth_get_error</b>
Funktion	Gibt einen akkumulierten Fehlercode zurück. Dieser beinhaltet alle Fehler, die bei den zuletzt ausgeführten RTC6 Ethernet-Karten-Befehlen aufgetreten sind.
Aufruf	<code>EthAccError = eth_get_error()</code>
Rückgabe	Gleicher Fehlercode wie <b>eth_get_last_error</b> . Als 32-Bit-Wert ohne Vorzeichen. Sind mehrere Fehler gleichzeitig aufgetreten, so sind mehrere Bits gesetzt.
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• Zur Fehlerbehandlung siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98.</li> <li>• Die Befehle <b>eth_get_error</b> und <b>n_eth_get_error</b> sind auch ohne explizite Zugriffsberechtigung für irgendeine RTC6 Ethernet-Karte verfügbar.</li> <li>• Die kartenspezifischen Fehlervariablen <code>LastError</code> und <code>AccError</code> (siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98) werden von <b>eth_get_error</b> weder erzeugt noch verändert.</li> <li>• Wenn bei einem RTC6 Ethernet-Karten-Befehlsaufruf (intern) mehrere Fehler nacheinander auftreten, gibt <b>eth_get_last_error</b> möglicherweise nur den wirklich letzten Fehler (z. B. nur "<b>not_acquired</b>") zurück. Die tatsächlichen Fehlerursachen können mit <b>eth_get_error</b> festgestellt werden, weil dieser die akkumulierten Fehler zurück gibt.</li> <li>• Der akkumulierte Fehler wird automatisch mit jeder Inbesitznahme der RTC6 Ethernet-Karte zurückgesetzt.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 608, OUT 608, RBF 611.
Verweise	<b>eth_get_last_error</b>

Kontrollbefehl	<b>eth_get_ip</b>
Funktion	Gibt für einen angegebenen RTC6-Kartenverwaltungsindex die zugehörige IP-Adresse zurück.
Aufruf	IP = eth_get_ip( CardNo )
Rückgabe	IP-Adresse. In Big-Endian Byte-Reihenfolge. Der Wert ist 0, wenn die Karte keine RTC6 Ethernet-Karte ist.
Parameter	CardNo      Index der RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung. Als 32-Bit-Wert ohne Vorzeichen.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_get_ip</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zur Konvertierung von der Big-Endian Byte-Reihenfolge in die gewöhnliche Dezimalpunktschreibweise dient <b>eth_convert_ip_to_string</b>. Siehe auch <b>Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen"</b>, Seite 88.</li> <li>• <b>eth_get_ip</b> erlaubt eine einfachere Abfrage der IP-Adresse aus der RTC6-Kartenverwaltung als über <b>eth_get_card_info</b>.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_convert_ip_to_string</b> , <b>eth_get_card_info</b>

Kontrollbefehl	<b>eth_get_ip_search</b>
Funktion	Gibt für eine bestimmte RTC6 Ethernet-Karte in der Suchergebnisliste die zugehörige IP-Adresse zurück.
Aufruf	IP = eth_get_ip_search( SearchNo )
Rückgabe	IP-Adresse. In Big-Endian Byte-Reihenfolge.
Parameter	SearchNo      Index in der Suchergebnisliste. Als 32-Bit-Wert ohne Vorzeichen.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_get_ip_search</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Die Suchergebnisliste wird mit <b>eth_search_cards</b> oder <b>eth_search_cards_range</b> erzeugt. Deren Rückgabewert stimmt mit dem von <b>eth_found_cards</b> überein. Siehe auch <b>Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten"</b>, Seite 88.</li> <li>• Der Rückgabewert ist 0 für SearchNo = 0 und SearchNo &gt; <b>eth_found_cards</b>. Gleichzeitig wird der <b>get_last_error</b>-Returncode <b>RTC6_PARAM_ERROR</b> erzeugt.</li> <li>• Wurde noch gar keine Suche ausgeführt, dann gilt automatisch: SearchNo &gt; <b>eth_found_cards</b>.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 608, OUT 608, RBF 611.
Verweise	<b>eth_get_ip</b>

Kontrollbefehl	eth_get_last_error
Funktion	Gibt einen Fehlercode zurück, der nur die beim zuletzt ausgeführten RTC6 Ethernet-Karten-Befehl aufgetretenen Fehler beinhaltet.
Aufruf	EthLastError = eth_get_last_error()
Rückgabe	Fehlercode. Als 32-Bit-Wert ohne Vorzeichen. Sind mehrere Fehler gleichzeitig aufgetreten, so sind mehrere Bits gesetzt.
	no_error = 0. Kein Fehler. Fehler Zahlenwert ( $2^{\text{[Bit]}}$ )
	wsasstartup_failed Bit 0 1
	wrong_winsock_version Bit 1 2
	Reserviert. Bit 2 4
	udp_create_socket_error Bit 3 8
	udp_bind_socket_error Bit 4 16
	udp_connect_socket_error Bit 5 32
	udp_excl_add_use_error Bit 6 64
	udp_bc_ena_error Bit 7 128
	Reserviert. Bit 8 256
	tcp_create_socket_error Bit 9 512
	tcp_excl_add_use_error Bit 10 1024
	tcp_bind_socket_error Bit 11 2048
	Reserviert. Bit 12 4096
	tcp_connect_socket_error Bit 13 8192
	tcp_connect_sel_error Bit 14 16384
	tcp_connect_fds_error Bit 15 32768
	tcp_nodelay_error Bit 16 65536
	create_thread_failed Bit 17 131072
	udp_rcv_error Bit 18 262144
	udp_send_error Bit 19 524288
	udp_rcv_timeout Bit 20 1048576
	already_acquired Bit 21 2097152
	not_acquired Bit 22 4194304
	access_denied Bit 23 8388608
	send_tgm_timeout Bit 24 16777216
	card_not_found Bit 25 33554432
	core1_timeout Bit 26 67108864
Parameter	Keine.

Kontrollbefehl	eth_get_last_error
Hinweise	<ul style="list-style-type: none"> <li>• Zur Fehlerbehandlung siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98.</li> <li>• Die Befehle <b>eth_get_last_error</b> und <b>n_eth_get_last_error</b> sind auch ohne explizite Zugriffsberechtigung für irgendeine RTC6 Ethernet-Karte verfügbar.</li> <li>• Die kartenspezifischen Fehlervariablen <b>LastError</b> und <b>AccError</b> (siehe RTC5-Handbuch, Kapitel 6.8 "Error-Handling", Seite 98) werden von <b>eth_get_last_error</b> weder erzeugt noch verändert.</li> <li>• Jeder <b>eth_get_last_error</b>-Fehler führt auch zu einem <b>get_last_error</b>-Fehler <b>RTC6_ETH_ERROR [Bit #13 = 1 (= 8192)]</b>: RTC6 Ethernet-Kartenfehler.</li> <li>• Wenn bei einem RTC6 Ethernet-Karten-Befehlsaufruf (intern) mehrere Fehler nacheinander auftreten, gibt <b>eth_get_last_error</b> möglicherweise nur den wirklich letzten Fehler (z. B. nur "<b>not_acquired</b>") zurück. Die tatsächlichen Fehlerursachen können mit <b>eth_get_error</b> festgestellt werden, weil dieser die akkumulierten Fehler zurück gibt.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_get_error</b> , <b>get_error</b> , <b>reset_error</b> , <b>set_verify</b>



Kontrollbefehl	<b>eth_get_port_numbers</b>
Funktion	Gibt die UDP-Ports für Kartensuchvorgänge (UDPsearch) und exklusive Verbindungen (UDPexcl) sowie den TCP-Port einer RTC6 Ethernet-Karte zurück.
Aufruf	Result = eth_get_port_numbers( &UDPsearch, &UDPexcl, &TCP )
Rückgabe	Fehlercode. Als 32-Bit-Wert ohne Vorzeichen. 0:               Erfolg: OK. 1:               Fehler: Keine RTC6 Ethernet-Karte.
Parameter- rückgabe	UDPsearch     UPD-Port für Kartensuchvorgänge. UDPexcl       UDP-Port für exklusive Verbindungen. TCP            TCP-Port für exklusive Verbindungen. Jeweils als Pointer auf einen 32-Bit-Wert ohne Vorzeichen. Alle Werte sind jeweils 0, wenn die Karte keine RTC6 Ethernet-Karte ist.
Hinweise	<ul style="list-style-type: none"> <li>Die Werte können auch mit <b>eth_get_card_info</b> abgefragt werden.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_get_card_info, eth_set_port_numbers</b>

Kontrollbefehl	<b>eth_get_serial_search</b>
Funktion	Gibt für eine bestimmte RTC6 Ethernet-Karte in der Suchergebnisliste die zugehörige Seriennummer zurück.
Aufruf	Serialnumber = eth_get_serial_search( SearchNo )
Rückgabe	Seriennummer. Als 32-Bit-Wert ohne Vorzeichen.
Parameter	SearchNo     Index in der Suchergebnisliste. Als 32-Bit-Wert ohne Vorzeichen.
Hinweise	<ul style="list-style-type: none"> <li><b>eth_get_serial_search</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>Die Suchergebnisliste wird mit <b>eth_search_cards</b> oder <b>eth_search_cards_range</b> erzeugt. Deren Rückgabewert stimmt mit dem von <b>eth_found_cards</b> überein. Siehe auch Kapitel 2.5.2 "Über die Suche von RTC6 Ethernet-Karten", Seite 88.</li> <li>Der Rückgabewert ist 0 für SearchNo = 0 und SearchNo &gt; <b>eth_found_cards</b>. Gleichzeitig wird der <b>get_last_error</b>-Returncode RTC6_PARAM_ERROR erzeugt.</li> <li>Wurde noch gar keine Suche ausgeführt, dann gilt automatisch: SearchNo &gt; <b>eth_found_cards</b>.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 608, OUT 608, RBF 611.
Verweise	<b>get_serial_number, eth_get_card_info_search</b>

Kontrollbefehl	<b>eth_get_static_ip</b>
Funktion	Gibt die auf der RTC6 Ethernet-Karte gespeicherte statische IP-Adresse, Subnetzmaske und Gateway-Adresse zurück.
Aufruf	<code>Result = eth_get_static_ip( &amp;Ip, &amp;NetMask, &amp;Gateway )</code>
Rückgabe	<p>Fehlercode. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0:           Erfolg: OK. 1:           Fehler: Keine RTC6 Ethernet-Karte.</p>
Parameter	Keine.
Parameter- rückgabe	<p>Ip            Statische IP-Adresse in Big-Endian Byte-Reihenfolge. NetMask      Subnetzmaske in Big-Endian Byte-Reihenfolge. Gateway      Adresse des Gateways in Big-Endian Byte-Reihenfolge.</p> <p>Jeweils als Pointer auf einen 32-Bit-Wert ohne Vorzeichen. Alle Werte sind jeweils 0, wenn die Karte keine RTC6 Ethernet-Karte ist oder noch keine statische IP-Adresse programmiert wurde.</p>
Hinweise	<ul style="list-style-type: none"> <li>Zur Konvertierung von der Big-Endian Byte-Reihenfolge in die gewöhnliche Dezimalpunktschreibweise dient <b>eth_convert_ip_to_string</b>. Siehe auch <b>Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen"</b>, Seite 88.</li> <li>Die Werte können auch mit <b>eth_get_card_info</b> abgefragt werden.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_convert_ip_to_string, eth_get_card_info, eth_set_static_ip</b>

Kontrollbefehl	<b>eth_max_card</b>
Funktion	Liefert den höchsten Index in der RTC6-Kartenverwaltung zurück, an dem eine RTC6 Ethernet-Karte eingetragen ist.
Aufruf	<code>Result = eth_max_card()</code>
Rückgabe	<p>Höchster Index einer RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0: In der RTC6-Kartenverwaltung ist keine RTC6 Ethernet-Karte eingetragen.</p> <p>n: Höchster Index, an dem eine RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung eingetragen ist.</p>
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_max_card</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• <b>eth_max_card</b> liefert nicht die Gesamtzahl der RTC6 Ethernet-Karten in der RTC6-Kartenverwaltung zurück. Dazu steht <b>eth_count_cards</b> zur Verfügung.</li> <li>• Mit <b>eth_max_card</b> können alle RTC6 Ethernet-Karten aus der RTC6-Kartenverwaltung durch eine einfache Schleife entfernt werden:  <pre>while ( eth_max_card() ) { release_rtc( eth_max_card() ); eth_remove_card( eth_max_card() ); }</pre> </li> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung", Seite 89.</b></li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_count_cards</b> , <b>eth_remove_card</b>

Kontrollbefehl	<b>eth_remove_card</b>
Funktion	Löscht den RTC6 Ethernet-Karten-Eintrag aus der RTC6-Kartenverwaltung am angegebenen Index.
Aufruf	<code>Result = eth_remove_card( CardNo )</code>
Rückgabe	<p>Fehler code. Als 32-Bit-Wert mit Vorzeichen.</p> <p>–2 Fehler: Der Eintrag kann nicht gelöscht werden. An diesem Index ist eine RTC6 Ethernet-Karte eingetragen, die aber noch in Besitz genommen ist.</p> <p>–1 Fehler: Der Eintrag kann nicht gelöscht werden. An diesem Index ist eine RTC6 PCI-Express-Karte eingetragen.</p> <p>0 Fehler: Der Eintrag kann nicht gelöscht werden. An diesem Index ist eine "Keine Karte" eingetragen oder <code>CardNo</code> ist ungültig (<math>&gt; 255</math>).</p> <p><code>n (= CardNo)</code> Erfolg: Der Eintrag wurde entfernt.</p>
Parameter	<p><code>CardNo</code> Index der RTC6 Ethernet-Karte in der RTC6-Kartenverwaltung. Als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_remove_card</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Bei allen Fehlern wird der <b>get_last_error</b>-Returncode <code>RTC6_PARAM_ERROR</code> erzeugt.</li> <li>• Einträge für "Keine Karte" und RTC6 PCI-Express-Karten können nicht gelöscht werden.</li> <li>• Vor dem Löschen eines RTC6 Ethernet-Karten-Eintrags muss eine in Besitz genommene Karte mit <b>release_rtc</b> explizit freigegeben werden.</li> <li>• Mit <code>CardNo = 0</code> wird der RTC6 Ethernet-Karten-Eintrag am Index (<b>rtc6_count_cards</b> + <b>eth_count_cards</b>) gelöscht (siehe auch lückenlose Kartenummerierung mit <b>eth_assign_card</b> oder <b>eth_assign_card_ip</b> und <code>CardNo = 0</code>). <code>CardNo = 0</code> und freigeählte Zuweisung von RTC6 Ethernet-Karten zu Indices sollten nicht zusammen verwendet werden.</li> <li>• Alle RTC6 Ethernet-Karten-Einträge können aus der RTC6-Kartenverwaltung durch eine einfache Schleife entfernt werden (vorausgesetzt, alle RTC6 Ethernet-Karten sind <i>nicht</i> in Besitz genommen):  <pre>while ( eth_remove_card(0) &gt; 0 );</pre> </li> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</b>, Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_assign_card, eth_assign_card_ip, eth_count_cards, release_rtc, rtc6_count_cards</b>

Kontrollbefehl	<b>eth_search_cards</b>
Funktion	Führt eine Kartensuche in Form eines Broadcasts aus und gibt die Anzahl der RTC6 Ethernet-Karten zurück, die im angegebenen Adressbereich geantwortet haben.
Aufruf	<code>Result = eth_search_cards( Ip, NetMask )</code>
Rückgabe	Anzahl der RTC6 Ethernet-Karten, die im angegebenen Adressbereich (innerhalb von <b>TimeOut</b> ) geantwortet haben. Als 32-Bit-Wert ohne Vorzeichen.
Parameter	<p><b>Ip</b> IP-Adresse in Big-Endian Byte-Reihenfolge.</p> <p><b>Netmask</b> Subnetzmaske in Big-Endian Byte-Reihenfolge.</p> <p>Alle als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_search_cards</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zur Umwandlung aus der gewöhnlichen Dezimalpunktschreibweise in die Big-Endian Byte-Reihenfolge kann <b>eth_convert_string_to_ip</b> verwendet werden. Siehe auch Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen", Seite 88.</li> <li>• Der Timeout-Wert kann mit <b>eth_set_search_cards_timeout</b> eingestellt werden.</li> <li>• Ein Broadcast erreicht zuverlässig nur Adressen innerhalb des angegebenen Netzwerksegmentes. Für andere Anwendungsfälle kann eine Kartensuche per IP-Scan (Befehl <b>eth_search_cards_range</b>) durchgeführt werden.</li> <li>• Die Anzahl der gefundenen RTC6 Ethernet-Karten kann auch zu einem späteren Zeitpunkt mit <b>eth_found_cards</b> abgefragt werden.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_convert_string_to_ip</b> , <b>eth_found_cards</b> , <b>eth_search_cards_range</b> , <b>eth_set_search_cards_timeout</b>

Kontrollbefehl	<b>eth_search_cards_range</b>
Funktion	Führt eine Suche in einem konkret angegebenen IP-Adressbereich aus und gibt die Anzahl der RTC6 Ethernet-Karten zurück, die geantwortet haben.
Aufruf	<code>Result = eth_search_cards_range( StartIP, EndIp )</code>
Rückgabe	Anzahl der RTC6 Ethernet-Karten, die (innerhalb von <b>TimeOut</b> ) geantwortet haben. Als 32-Bit-Wert ohne Vorzeichen.
Parameter	<p><b>StartIP</b> Start-IP-Adresse in Big-Endian Byte-Reihenfolge.</p> <p><b>EndIp</b> Ende-IP-Adresse in Big-Endian Byte-Reihenfolge.</p> <p>Alle als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_search_cards_range</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Zur Umwandlung aus der gewöhnlichen Dezimalpunktschreibweise in die Big-Endian Byte-Reihenfolge kann <b>eth_convert_string_to_ip</b> verwendet werden. Siehe auch Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen", Seite 88.</li> <li>• <b>eth_search_cards_range</b> führt die Suche durch Versenden von UDP-Paketen an jede IP-Adresse im angegebenen Adressbereich aus. Sie umfasst verlässlich den angegebenen Adressbereich (vgl. Broadcast bei <b>eth_search_cards</b>).</li> <li>• Der <b>TimeOut</b>-Wert kann mit <b>eth_set_search_cards_timeout</b> eingestellt werden.</li> <li>• Die Anzahl der gefundenen RTC6 Ethernet-Karten kann auch zu einem späteren Zeitpunkt mit <b>eth_found_cards</b> abgefragt werden.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_convert_string_to_ip</b> , <b>eth_found_cards</b> , <b>eth_search_cards</b> , <b>eth_set_search_cards_timeout</b>

Kontrollbefehl	<b>eth_set_com_timeouts</b>
Funktion	Stellt Timing-Informationen für eine bestimmte RTC6 Ethernet-Karte ein, die in der RTC6-Kartenverwaltung eingetragen und in Besitz genommen ist.
Aufruf	eth_set_com_timeouts( AcquireTimeout, AcquireMaxRetries, SendRecvTimeout, SendRecvMaxRetries, KeepAlive, KeepInterval )
Rückgabe	Keine.
Parameter	AcquireTimeout      Reserviert.
	AcquireMaxRetries    Reserviert.
	SendRecvTimeout     Reserviert.
	SendRecvMaxRetries   Reserviert.
	KeepAlive            Reserviert.
	KeepInterval         Reserviert.
	Alle als 32-Bit-Wert ohne Vorzeichen. Bei 0 wird der Parameter nicht verändert.
Hinweise	<ul style="list-style-type: none"> <li>• Siehe auch <b>Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</b>, Seite 89.</li> <li>• Zum Abfragen der Einstellungen wird <b>eth_get_com_timeouts</b> verwendet.</li> <li>• Normalerweise ist es nicht erforderlich, diese Parameterwerte zu ändern. Sollten Timing-Probleme auftreten, wenden Sie sich an SCANLAB, um Ihre lokalen Besonderheiten zu klären.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_get_com_timeouts</b>

Kontrollbefehl	<b>eth_set_port_numbers</b>
Funktion	Speichert die UDP-Ports für Kartensuchvorgänge (UDPsearch) und exklusive Verbindungen (UDPexcl) sowie den TCP-Port in den nichtflüchtigen Speicher (Flash Memory) der RTC6 Ethernet-Karte.
Aufruf	Result = eth_set_port_numbers( UDPsearch, UDPexcl, TCP )
Rückgabe	<p>Fehlercode. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0:           Erfolg: OK.</p> <p>1:           Fehler: Kein Zugriff auf die RTC6 Ethernet-Karte. <b>get_last_error</b>-Returncode: RTC6_ACCESS_DENIED.</p> <p>2:           Fehler: Keine RTC6 Ethernet-Karte. <b>get_last_error</b>-Returncode: RTC6_TYPE_REJECTED.</p> <p>3:           Fehler: Programmierung des Flash-Speichers ist nicht möglich. Möglicherweise ist die RTC6 Ethernet-Karte beschäftigt. <b>get_last_error</b>-Returncode: RTC6_BUSY oder RTC6_FLASH_ERROR.</p>
Parameter	<p>UDPsearch    UDP-Port für Kartensuchvorgänge.</p> <p>UDPexcl      UDP-Port für exklusive Verbindungen.</p> <p>TCP           TCP-Port für exklusive Verbindungen.</p> <p>Als 32-Bit-Wert ohne Vorzeichen. Zulässiger Wertebereich: [0...65535]. Überlaufende Werte werden geclippt.</p>
Hinweise	<ul style="list-style-type: none"> <li>• Voraussetzungen für <b>eth_set_port_numbers</b>: <ul style="list-style-type: none"> <li>– Auf der RTC6 Ethernet-Karte darf gerade keine Liste ausgeführt werden.</li> <li>– Die RTC6 Ethernet-Karte muss in der RTC6-Kartenverwaltung eingetragen sein.</li> <li>– Die RTC6 Ethernet-Karte muss in Besitz genommen sein.</li> </ul> </li> <li>• Wird für einen Parameter 0 angegeben, so wird dieser Parameter nicht überschrieben.</li> <li>• Gespeicherte Parameter werden erst nach einem Neustart der RTC6 Ethernet-Karte verwendet.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_get_port_numbers</b>



Kontrollbefehl	<b>eth_set_search_cards_timeout</b>
Funktion	Setzt einen Timeout-Wert.
Aufruf	<code>eth_set_search_cards_timeout( Timeout )</code>
Rückgabe	Keine.
Parameter	Timeout      Timeout-Wert in $\mu$ s. Als 32-Bit-Wert ohne Vorzeichen.
Hinweise	<ul style="list-style-type: none"> <li>• <b>eth_set_search_cards_timeout</b> ist nicht als Multi-Board-Befehl verfügbar.</li> <li>• Der voreingestellte Timeout-Wert ist 5 ms.</li> <li>• Der Timeout-Wert ist relevant bei <b>eth_search_cards</b> und <b>eth_search_cards_range</b>.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_search_cards</b> , <b>eth_search_cards_range</b>

Kontrollbefehl	<b>eth_set_static_ip</b>
Funktion	Schreibt eine statische IP-Adresse, eine Subnetzmaske sowie eine Gateway-Adresse auf die RTC6 Ethernet-Karte.
Aufruf	Result = eth_set_static_ip( Ip, NetMask, Gateway )
Rückgabe	<p>Fehlercode. Als 32-Bit-Wert ohne Vorzeichen.</p> <p>0:           Erfolg: OK.</p> <p>1:           Fehler: Kein Zugriff auf die RTC6 Ethernet-Karte. <b>get_last_error</b>-Returncode: RTC6_ACCESS_DENIED.</p> <p>2:           Fehler: Keine RTC6 Ethernet-Karte. <b>get_last_error</b>-Returncode: RTC6_TYPE_REJECTED.</p> <p>3:           Fehler: Programmierung des Flash-Speichers ist nicht möglich. Möglicherweise ist die RTC6 Ethernet-Karte beschäftigt. <b>get_last_error</b>-Returncode: RTC6_BUSY oder RTC6_FLASH_ERROR.</p>
Parameter	<p>Ip            Statische IP-Adresse in Big-Endian Byte-Reihenfolge.</p> <p>NetMask      Subnetzmaske in Big-Endian Byte-Reihenfolge.</p> <p>Gateway      Adresse des Gateways in Big-Endian Byte-Reihenfolge.</p> <p>Alle als 32-Bit-Wert ohne Vorzeichen.</p>
Hinweise	<ul style="list-style-type: none"> <li>• Voraussetzungen für <b>eth_set_static_ip</b>: <ul style="list-style-type: none"> <li>– Auf der RTC6 Ethernet-Karte darf gerade keine Liste ausgeführt werden.</li> <li>– Die RTC6 Ethernet-Karte muss in der RTC6-Kartenverwaltung eingetragen sein.</li> <li>– Die RTC6 Ethernet-Karte muss in Besitz genommen sein.</li> </ul> </li> <li>• Soll kein Gateway verwendet werden, dann muss <i>Gateway</i> auf 0 gesetzt werden.</li> <li>• Zur Umwandlung aus der gewöhnlichen Dezimalpunktschreibweise in die Big-Endian Byte-Reihenfolge kann <b>eth_convert_string_to_ip</b> verwendet werden. Siehe auch <b>Kapitel 2.5.1 "Hinweise zum Arbeiten mit IP-Adressen", Seite 88.</b></li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<b>eth_convert_string_to_ip, eth_get_static_ip</b>

Kontrollbefehl	<b>get_card_type</b>
Funktion	Liefert den Kartentyp zurück.
Aufruf	<code>Result = get_card_type()</code>
Rückgabe	Kartentyp. Als 32-Bit-Wert ohne Vorzeichen. 0: "Keine Karte". 1: RTC6 PCI-Express-Karte. 2: RTC6 Ethernet-Karte.
Parameter	Keine.
Hinweise	<ul style="list-style-type: none"> <li>Siehe <a href="#">Kapitel 2.5.3 "Über die RTC6-Kartenverwaltung"</a>, Seite 89.</li> </ul>
RTC4→RTC6	Neuer Befehl.
RTC5→RTC6	Neuer Befehl.
Versionsinfo	Verfügbar ab Version DLL 606, OUT 606, RBF 611.
Verweise	<a href="#">eth_assign_card</a> , <a href="#">eth_assign_card_ip</a>

## **2.6 Sichere Einschalt- und Aussaltreihenfolge (zur Ausführung von Steuerungs-Software)**

Beachten Sie beim Einschalten der Komponenten eines Lasersystems die folgende Reihenfolge:

- (1) Schalten Sie den Netzwerk-PC ein.
- (2) Schalten Sie das Netzteil für die RTC6 Ethernet-Karte ein.
- (3) Starten Sie die Steuerungs-Software.
- (4) Schalten Sie die benötigten Peripheriegeräte ein.
- (5) Schalten Sie die Spannungsversorgung des Scan-Systems ein.
- (6) Schalten Sie den Laser ein.

Gehen Sie beim Abschalten des Lasersystems in genau umgekehrter Reihenfolge vor:

- (1) Schalten Sie den Laser aus.
- (2) Schalten Sie die Spannungsversorgung des Scan-Systems aus.
- (3) Schalten Sie die Peripheriegeräte aus.
- (4) Beenden Sie die Steuerungs-Software.
- (5) Schalten Sie das Netzteil für die RTC6 Ethernet-Karte aus.
- (6) Schalten Sie den Netzwerk-PC aus.



### **Achtung!**

- Beim Ein- und Ausschalten des PCs können an den Ausgängen der RTC6 Ethernet-Karte kurzzeitige Pegeländerungen, also auch unvorhergesehene Änderungen der Lasersteuersignale stattfinden. Die obige Ein- und Aussaltreihenfolge muss deswegen unbedingt eingehalten werden, sonst besteht die Gefahr dass der Laser kurzzeitig unerwartet angeht.
- Das Scan-System muss immer nach dem PC und der Steuerungs-Software eingeschaltet bzw. vorher wieder ausgeschaltet werden, sonst kann es zu unvorhergesehenen Scan-Bewegungen des Scan-Systems kommen. Der Laser muss immer zuletzt eingeschaltet bzw. zuerst ausgeschaltet werden, sonst besteht die Gefahr, dass der Laserstrahl in eine unkontrollierte Richtung abgelenkt wird.

## 2.7 Technische Spezifikationen der RTC6 Ethernet-Karte

### Systemanforderungen

Windows-PC mit Ethernet-Schnittstelle (Gigabit-Ethernet (bevorzugt), 10/100 Mbit/s-Ethernet)

Betriebssystem      Wie bei RTC6 PCI-Express-Karte: Microsoft Windows 10, 8, 7 als 32-Bit oder 64-Bit Version

### Abmessungen (ohne eingesteckte Steckverbinder)

Länge      120 mm  
Breite      103 mm  
Höhe      ca. 30 mm

### Sonstige Anschlüsse und Spezifikationen

Versorgungsspannungsbereich      +11 V...+50 V

Maximale Leistungsaufnahme, ohne angeschlossene Peripheriegeräte      < 10 W

Um die Verlustleistung gering zu halten, sollte eine niedrige Eingangsspannung gewählt werden.

#### Betriebstemperatur

- mit natürlicher Konvektion      10 °C...50 °C
- mit erzwungener Konvektion      10 °C...60 °C

Der Benutzer hat für eine ausreichende Kühlung der RTC6 Ethernet-Karte zu sorgen.

Standalone-Funktionalität      In Planung

### Schnittstelle zum Netzwerk

Ethernet	Gigabit-Ethernet (bevorzugt), 10/100 Mbit/s-Ethernet
IP-Adressen-Bezug <sup>(a)</sup>	<ul style="list-style-type: none"> <li>• Durch einen DHCP-Server</li> <li>• Statische IP-Adresse (auf der Karte speicherbar)</li> </ul>
Benötigte lokale Ports <sup>(b)</sup>	<ul style="list-style-type: none"> <li>• 63749 (UDP)</li> <li>• 63750 (UDP, TCP)</li> </ul>

(a) auto-IP wird derzeit nicht unterstützt.

(b) Konfigurierbar mit **eth\_set\_port\_numbers**.

### Ansteuerung eines Scan-Systems

Zahl der Listenspeicherbereiche	Wie RTC6 PCI-Express-Karte: Bis zu 3, konfigurierbar
Gesamtkapazität des Listenspeichers	Wie bei RTC6 PCI-Express-Karte: 8.388.608 Listenplätze
Ausgabeintervall der Mikrovektoren	Wie bei RTC6 PCI-Express-Karte: 10 µs
Maximaler Wertebereich der Bildfeldkoordinaten	Wie bei RTC6 PCI-Express-Karte: -524.288...+524.287 (20 Bit mit Vorzeichen)
Virtuelles Bildfeld, z. B. für "Processing on the fly"	Wie bei RTC6 PCI-Express-Karte: -268.435.456...+268.435.455 (28 Bit mit Vorzeichen = 29 Bit)

## Schnittstellen zu Scan-Systemen

### • SCANHEADs-Stiftleiste<sup>(a)</sup>

Anschluss	20-polige Stiftleiste mit Rastermaß 2,54 mm. Die Signale für den ersten Scan-Kopf werden an Pin (01)...Pin (10) ausgegeben. Nur für Karten mit der "SSHC"-Option gilt: Die Signale für den zweiten Scan-Kopf werden an Pin (11)...Pin (20) ausgegeben.
Signalübertragung	Wie bei RTC6 PCI-Express-Karte: SL2-100-Protokoll. Über XY2-100-Konverter (Zubehör): XY2-100-Protokoll

(a) RTC6 Ethernet-Karten haben nur diesen einen einzigen Anschluss. Dagegen haben RTC6 PCI-Express-Karten 2 getrennte Anschlüsse:  
1 × SCANHEAD (9-polige Sub-D-Buchse, weiblich)  
1 × 2. SCANHEAD (10-polige Stiftleiste mit 2,54 mm Rastermaß)

## Schnittstellen zum Laser und zur Peripherie

### • LASER-Stiftleiste

Bei RTC6 PCI-Express-Karten: 15-polige Sub-D-Buchse (weiblich).

Anschluss	16-polige Stiftleiste mit Rastermaß 2,54 mm
Laserausgangssignale	Wie bei RTC6 PCI-Express-Karte: LASER1, LASER2, LASERON
• TTL-Pegel	Wie bei RTC6 PCI-Express-Karte: 5 V, HIGH-aktiv oder LOW-aktiv programmierbar
• Max. Last	Wie bei RTC6 PCI-Express-Karte: 20 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND bei Karten ohne "DC/DC" Option. GND2 bei Karten mit "DC/DC" Option
Analoge Ausgänge	Wie bei RTC6 PCI-Express-Karte: ANALOG OUT1 und ANALOG OUT2 <sup>(a)</sup> .
• Ausgangsspannungsbereich	Wie bei RTC6 PCI-Express-Karte: 0 V...10 V
• Auflösung	Wie bei RTC6 PCI-Express-Karte: 12 Bit
• Max. Last	Wie bei RTC6 PCI-Express-Karte: 5 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND

(a) Bei RTC5-Karten und RTC6 PCI-Express-Karten ist das ANALOG OUT2-Signal zusätzlich auch an der MARKING ON THE FLY-Stiftleiste verfügbar.

## • LASER-Stiftleiste (Forts.)

Digitaler Eingang	Wie bei RTC6 PCI-Express-Karte: 2 Bits
• LOW-Pegel	Wie bei RTC6 PCI-Express-Karte: < 0,6 V
• HIGH-Pegel	Wie bei RTC6 PCI-Express-Karte: > 2,3 V
• Max. Eingangsspannungsbereich	Wie bei RTC6 PCI-Express-Karte: -0,5 V... +5,5 V
• Eingangswiderstand (Pull-up)	Wie bei RTC6 PCI-Express-Karte: > 4,7 kΩ
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
Digitaler Ausgang	Wie bei RTC6 PCI-Express-Karte: 2 Bits, gepuffert
• LOW-Pegel	Wie bei RTC6 PCI-Express-Karte: < 0,55 V
• HIGH-Pegel	Wie bei RTC6 PCI-Express-Karte: > 3,8 V
• Max. Last	Wie bei RTC6 PCI-Express-Karte: 20 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
Eingänge für externe Start- und Stoppsignale	Wie bei RTC6 PCI-Express-Karte: TTL LOW-aktiv, intern über Pull-up-Widerstände (4,7 kΩ) mit +3,3 V verbunden
• /START	Wie bei RTC6 PCI-Express-Karte: flankensensitiv
• /STOP	Wie bei RTC6 PCI-Express-Karte: pegelsensitiv
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
Sonstige Signale	
• BUSY OUT	Wie bei RTC6 PCI-Express-Karte: 5 V, TTL HIGH-aktiv, max. 10 mA, Bezugspunkt GND
• +5 V	Wie bei RTC6 PCI-Express-Karte: max. 100 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND

## • EXTENSION 1-Stiftleiste

Anschluss	40-polige Stiftleiste mit Rastermaß 2,00 mm <sup>(a)</sup> . Ausgangssignalpegel einstellbar über Lötjumperfeld A
Digitaler Ausgang	Wie bei RTC6 PCI-Express-Karte: 16 Bits, gepuffert
• LOW-Pegel	Wie bei RTC6 PCI-Express-Karte: < 0,4 V
• HIGH-Pegel	Wie bei RTC6 PCI-Express-Karte: > 2,0 V (3,3 V oder 5 V)
• Max. Last	Wie bei RTC6 PCI-Express-Karte: ±8 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
• LATCH-Signal	Wie bei RTC6 PCI-Express-Karte: 3,3 V oder 5 V, TTL HIGH-aktiv, 5 µs-Puls, max. 10 mA
Digitaler Eingang	Wie bei RTC6 PCI-Express-Karte: 16 Bit, geschützt
• LOW-Pegel	Wie bei RTC6 PCI-Express-Karte: < 0,5 V
• HIGH-Pegel	Wie bei RTC6 PCI-Express-Karte: > 2,6 V...24 V
• Max. Eingangsspannungsbereich	Wie bei RTC6 PCI-Express-Karte: -0,5 V... +26 V
• Eingangswiderstand	Wie bei RTC6 PCI-Express-Karte: > 10 kΩ
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
• SYNC-Signal	Wie bei RTC6 PCI-Express-Karte: 3,3 V oder 5 V, TTL HIGH-aktiv, Rechtecksignal (5 µs-Puls, 10 µs-Periode), max. 10 mA

(a) Bei RTC6 PCI-Express-Karten: Rastermaß 2,54 mm.

## • EXTENSION 1-Stiftleiste (Forts.)

### Sonstige Signale

- **BUSY OUT**      Wie bei RTC6 PCI-Express-Karte: 3,3 V oder 5 V, TTL HIGH-aktiv, max. 10 mA
- **VCC**            Wie bei RTC6 PCI-Express-Karte: 3,3 V oder 5 V, max. 100 mA
- **+5 V**            Wie bei RTC6 PCI-Express-Karte: max. 100 mA
- **Bezugspunkt**    Wie bei RTC6 PCI-Express-Karte: GND

## • EXT. 2-Stiftleiste<sup>(a)</sup>

### Anschluss

10-polige Stiftleiste mit Rastermaß 2,00 mm<sup>(a)</sup>. Über Lötjumperfeld B ist Pin (09)<sup>(b)</sup>, und Pin (08)<sup>(c)</sup> über Lötjumperfeld C einstellbar

### Digitaler Ausgang

Wie bei RTC6 PCI-Express-Karte: 8 Bits, gepuffert

- **LOW-Pegel**      Wie bei RTC6 PCI-Express-Karte: < 0,4 V
- **HIGH-Pegel**    Wie bei RTC6 PCI-Express-Karte: > 2,0 V
- **Max. Last**        Wie bei RTC6 PCI-Express-Karte: ±8 mA
- **Bezugspunkt**    Wie bei RTC6 PCI-Express-Karte: GND
- **LATCH-Signal**    5 V, TTL HIGH-aktiv, 5 µs-Puls, max. 10 mA

### Laserausgangssignale

Nicht wie RTC6 PCI-Express-Karte: keine<sup>(d)</sup>

### Sonstige Signale

- **+5 V**            Wie bei RTC6 PCI-Express-Karte: max. 100 mA
- **Bezugspunkt**    Wie bei RTC6 PCI-Express-Karte: GND

(a) Bei RTC6 PCI-Express-Karten: 26-polige Stiftleiste mit Rastermaß 2,54 mm. Außerdem ist die Bedruckung auf der Karte EXTENSION 2.

(b) Pin (09) der RTC6 Ethernet-Karten = Pin (17) der RTC6 PCI-Express-Karten.

(c) Pin (08) der RTC6 Ethernet-Karten = Pin (15) der RTC6 PCI-Express-Karten.

(d) Bei RTC6 PCI-Express-Karten: LASER1 und LASER2.



## • MOF-Stiftleiste<sup>(a)</sup>

Anschluss	14-polige Stiftleiste mit Rastermaß 2,00 mm <sup>(a)</sup>
2 Encoder-Eingänge für Inkremental-Encoder	Wie bei RTC6 PCI-Express-Karte: ENCODER X(1 $\pm$ , 2 $\pm$ ) und ENCODER Y(1 $\pm$ , 2 $\pm$ ), jeweils für ein Paar von standardisierten differentiellen Eingangssignalen (RS-422). HIGH-Pegel $\geq 2,0$ V LOW-Pegel $\leq 0,8$ V $f \leq 4$ MHz
Analoger Ausgang	Nicht wie RTC6 PCI-Express-Karte: keiner <sup>(b)</sup>
Eingänge für externe Start- und Stoppsignale	Wie bei RTC6 PCI-Express-Karte: /START2, /STOP2 (siehe /START, /STOP der LASER-Stiftleiste, <a href="#">Seite 119</a> )
Sonstige Signale	
• BUSY OUT	Wie bei RTC6 PCI-Express-Karte: identisch mit BUSY OUT der LASER-Stiftleiste
• +5 V	Wie bei RTC6 PCI-Express-Karte: max. 100 mA
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND

(a) Bei RTC6 PCI-Express-Karten: 16-polige Stiftleiste mit Rastermaß 2,54 mm. Außerdem ist die Bedruckung auf der Karte MARKING ON THE FLY.

(b) Die RTC6 Ethernet-Karte hat an ihrer MOF-Stiftleiste keinen Pin für das ANALOG OUT2-Signal (im Unterschied zu MARKING ON THE FLY-Stiftleisten der RTC5-Karten und der RTC6 PCI-Express-Karte).

## • RS232-Stiftleiste<sup>(a)</sup>

Nicht wie RTC6 PCI-Express-Karte:  
Die RS232-Stiftleiste ist bei RTC6 Ethernet-Karten nicht vorhanden. Stattdessen gibt es die SPI/ANA/UART-Stiftleiste.

(a) Bei RTC6 PCI-Express-Karten: 10-polige Stiftleiste mit Rastermaß 2,54 mm.

## • McBSP/ANALOG-Stiftleiste<sup>(a)</sup>

Nicht wie RTC6 PCI-Express-Karte:  
Die McBSP/ANALOG-Stiftleiste ist bei RTC6 Ethernet-Karten nicht vorhanden. Stattdessen gibt es die SPI/ANA/UART-Stiftleiste.

(a) Bei RTC6 PCI-Express-Karten: 10-polige Stiftleiste mit Rastermaß 2,54 mm.

## • SPI/ANA/UART-Stiftleiste<sup>(a)</sup>

Anschluss	10-polige Stiftleiste mit Rastermaß 2,00 mm.
Analoge Eingänge	Wie bei RTC6 PCI-Express-Karte: ANALOG IN0 und ANALOG IN1
• Eingangsspannungsbereich	Wie bei RTC6 PCI-Express-Karte: 0 V...10 V
• Eingangsimpedanz	Nicht wie RTC6 PCI-Express-Karte: $> 5$ k $\Omega$
• ADC-Auflösung	Wie bei RTC6 PCI-Express-Karte: 12 Bit
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND

(a) Bei RTC6 PCI-Express-Karten nicht vorhanden. Diese haben stattdessen die RS232-Stiftleiste und die McBSP/ANALOG-Stiftleiste, beides 10-polige Stiftleisten mit Rastermaß 2,54 mm.

# • SPI/ANA/UART-Stiftleiste (Forts.)

In RS-232 Konfiguration

Eingang	Wie bei RTC6 PCI-Express-Karte: RxD
• Spannungsbereich	Wie bei RTC6 PCI-Express-Karte: max. –25 V...+25 V
Ausgang	Wie bei RTC6 PCI-Express-Karte: TxD
• Spannungsbereich	Wie bei RTC6 PCI-Express-Karte: max. –13 V...+13 V
Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND
Baud-Rate	Wie bei RTC6 PCI-Express-Karte: 300...115.200

In McBSP-Konfiguration, siehe RTC5-Handbuch, Abschnitt "Verwendung als McBSP-Schnittstelle", Seite 54.

- Sendersignalpegel Wie bei RTC6 PCI-Express-Karte: 3,3 V TTL
- Empfängersignalpegel Wie bei RTC6 PCI-Express-Karte: 3,3 V oder 5 V TTL
- McBSP-Mode Wie bei RTC6 PCI-Express-Karte:  
Single Phase Frame  
Single Element per Frame  
32 Bits per Element  
DataDelay N Bit
- Bezugspunkt Wie bei RTC6 PCI-Express-Karte: GND

In SPI-Konfiguration:

- RTC6 PCI-Express-Karten und RTC6 Ethernet-Karten bieten (anders als RTC5-Karten) *keine* SPI-Schnittstellenfunktionalität.

# • STEPPER-Stiftleiste<sup>(a)</sup>

Anschluss	10-polige Stiftleiste mit Rastermaß 2,00 mm.
Signale zur Ansteuerung von bis zu zwei Schrittmotoren:	
• Ausgänge ENABLE1, ENABLE2, DIRECTION1, DIRECTION2	Wie bei RTC6 PCI-Express-Karte: 5 V, TTL
• Ausgänge CLOCK1, CLOCK2	Wie bei RTC6 PCI-Express-Karte: 5 V, TTL HIGH-aktiv, 5 $\mu$ s-Puls
• Eingänge SWITCH1, SWITCH2	Wie bei RTC6 PCI-Express-Karte: TTL LOW-aktiv, intern über Pull-up-Widerstände (10 k $\Omega$ ) mit +3,3 V verbunden
• Bezugspunkt	Wie bei RTC6 PCI-Express-Karte: GND

(a) Bei RTC6 PCI-Express-Karten: 10-polige Stiftleiste mit Rastermaß 2,54 mm. Außerdem ist die Bedruckung auf der Karte STEPPER MOTOR.

## **2.8 Konformität mit der EU-Richtlinie zur elektromagnetischen Verträglichkeit (EMV)**

Die Konformität der RTC6 Ethernet-Karte mit der EG-Richtlinie 2014/30/EU (Elektromagnetische Verträglichkeit) wurde festgestellt.

Dazu wurde die RTC6 Ethernet-Karte in einen PC eingebaut und zusammen mit einem powerSCAN II Scan-Kopf (SL2-100-Schnittstelle) getestet.

### **Prüfbestimmungen**

Nachweis zur Erfüllung der Schutzziele der EG-Richtlinie 2014/30/EU (CE-Konformität für EMV) auf der Basis von

- EN 61000-6-2: 2005 + AC: 2005
- EN 61000-6-4: 2007 + A1: 2011

### **Ergebnis**

Die Prüflinge erfüllen die Spezifikationen.

## **2.9 Konformität mit FCC-Bestimmungen**

Die RTC6 Ethernet-Karte wurde getestet und entspricht den Grenzwerten für Digitalgeräte der Klasse A, gemäß Teil 15 der FCC-Bestimmungen.

Diese Grenzwerte sollen einen angemessenen Schutz gegen schädliche Störungen bieten, wenn die RTC6 Ethernet-Karte in einer kommerziellen Umgebung betrieben wird. Die RTC6 Ethernet-Karte erzeugt, verwendet und emittiert möglicherweise Hochfrequenzenergie und kann, sofern nicht in Übereinstimmung mit dieser Bedienungsanleitung installiert und verwendet, Störungen im Funkverkehr verursachen. Der Betrieb der RTC6 Ethernet-Karte in einem Wohngebiet wird wahrscheinlich schädliche Störungen verursachen; in diesem Fall ist der Benutzer verpflichtet, die Störungen auf eigene Kosten zu beseitigen.

### 3 Anhang B: Die UFPM-Erweiterungskarte

Die UFPM-Erweiterungskarte<sup>(1)</sup> wurde für RTC6 PCI-Express-Karten entwickelt und ist *nicht* separat erhältlich.

Aktuell wird sie nur in Verbindung mit der RTC6 PCI-Express-Karte #135427 (bereits montiert) geliefert, siehe **Abbildung 26**.

Die Abmessungen und Details der UFPM-Erweiterungskarte zeigt **Abbildung 27**.

Die UFPM-Erweiterungskarte wandelt mittels eines schnellen DA-Wandlers 8-Bit-Digitalsignale in Analogspannungswerte.

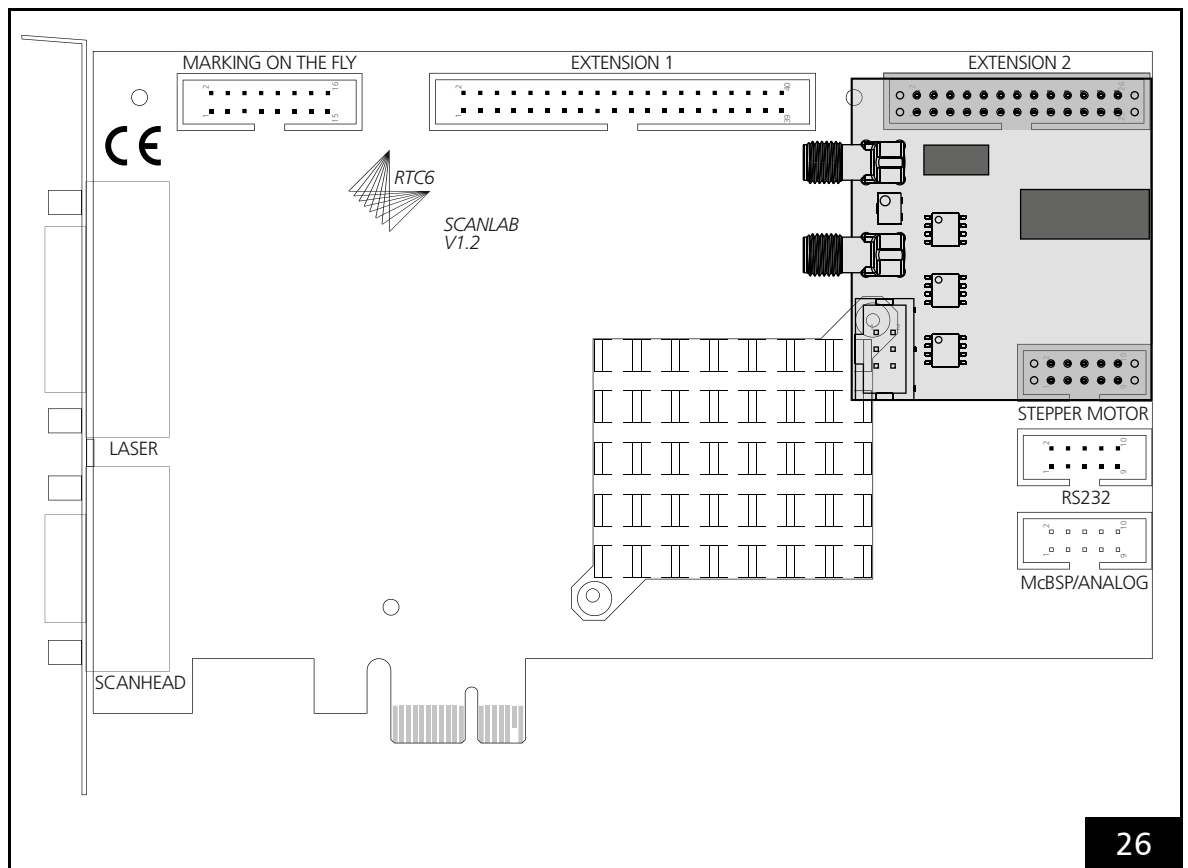
Benutzer benötigen diese zwingend, wenn sie

- *analog* angesteuerte Lasern betreiben *und*
- damit Rasterbilder (Bitmaps) mit Pixelfrequenzen  $\geq 100 \text{ kHz}$ <sup>(2)(3)</sup> markieren möchten (siehe **set\_pixel\_line**).

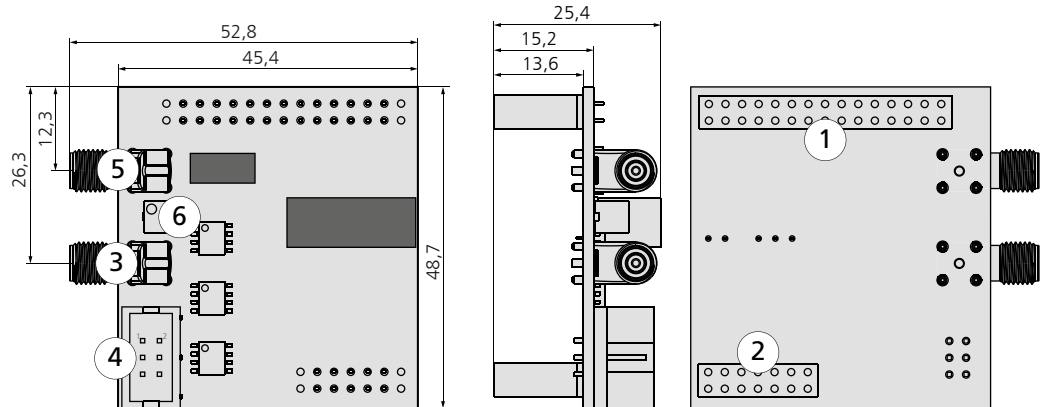
(2) Die UFPM-Erweiterungskarte unterstützt selbstverständlich auch Pixelfrequenzen  $< 100 \text{ kHz}$ .

(3) Für Pixelfrequenzen  $800 \text{ kHz} \dots 3,2 \text{ MHz}$  ist die RTC6 PCI-Express-Karte #135427 mit der Option 'UFPM' ausgestattet.

(1) #137980.



UFPM-Erweiterungskarte: Zusammenbau mit der RTC6 PCI-Express-Karte.  
Hinweis: Das SSHC-Slotblech (#115132) kann nicht verwendet werden.



## Legende

- 1 . . . J1. 30-polige Buchse zur Verbindung mit der 26-poligen EXTENSION 2-Stiftleiste der RTC6 PCI-Express-Karte. Die äußeren 4 Pins von Anschluss J1 sind nicht verbunden und verhindern lediglich ein versetztes Aufstecken der Karte.
- 2 . . . J2. 14-polige Buchse zur Verbindung mit der 10-poligen STEPPER MOTOR-Stiftleiste der RTC6 PCI-Express-Karte. Die Buchse J2 dient ausschließlich zur mechanischen Fixierung auf der RTC-Karte.
- 3 . . . J3. SMA-Buchse mit Analogausgang ANALOG OUTA.
- 4 . . . J4. 6-polige Stiftleiste mit Analogausgängen ANALOG OUTA und ANALOG OUTB.
- 5 . . . J5. SMA-Buchse mit Analogausgang ANALOG OUTB.
- 6 . . . P3. Potentiometer zum Feinabgleich der 5 V-Ausgangsamplitude der Analogausgänge ANALOG OUTA und ANALOG OUTB.

27

UFPM-Erweiterungskarte: Abmessungen und Details. Alle Maße in mm.

Bei Verwendung<sup>(1)</sup> der UFPM-Erweiterungskarte kann die Variation der Laserleistung von Pixel zu Pixel über Analogspannungswerte (in 8-Bit Auflösung, d.h. in  $2^8-1$  Stufen) gesteuert werden (Ausgangsspannungsbereich 0 V...5 V)<sup>(2)</sup>

Dazu müssen Benutzer die Ausgaben an den Port = 3 (8-Bit-Digital-Ausgang; an der EXTENSION 2-Stiftleiste) senden, siehe **set\_pixel\_line**.

- (1) Ohne UFPM-Erweiterungskarte kann die Steuerung auch über Analogspannungswerte (in 12-Bit Auflösung) mittels Ausgaben an den 12-Bit-Analog-Ausgang 1 (Port = 1) oder 12-Bit-Analog-Ausgang 2 (Port = 2) erfolgen. Allerdings kann die DA-Wandlung dieser Ausgaben nur für Pixelfrequenzen bis etwa 100 kHz (d.h. für HalfPeriod < ca. 320) in jedem Fall vollständig abgeschlossen werden.
- (2) Diese Analogspannungswerte (0 V...5 V) werden am ANALOG OUTA- und ANALOG OUTB-Analogausgang der UFPM-Erweiterungskarte ausgegeben. Die exakte Ausgangsamplitude der beiden Analogausgänge kann über das Potentiometer P3 feineingestellt werden, siehe **Abbildung 27**.

## Software-Voraussetzungen

- Die UFPM-Erweiterungskarte wird standardmäßig

Die Variation der Laserpulsdauer ist nur unter Verwendung des "Standard"-Pixelmodus (Modus = 0, Modus = 256) bis zu einer maximalen Frequenz von 400 kHz möglich. Auch kann nur in diesem Mode die Information für die Laserpulsdauer zusammen mit der Laserleistung pro Pixel übertragen werden. Die Laserpulsdauer wird mit einer Auflösung von  $1/64 \mu s$  angegeben.

Für alle "erweiterten" Pixelmodi (Modus = 16, Modus = 32, Modus = 64) ist die Laserpulsdauer vor Beginn der Pixelzeile über **set\_laser\_pulses**, **set\_laser\_pulses\_ctrl** oder **set\_laser\_timing** anzugeben.

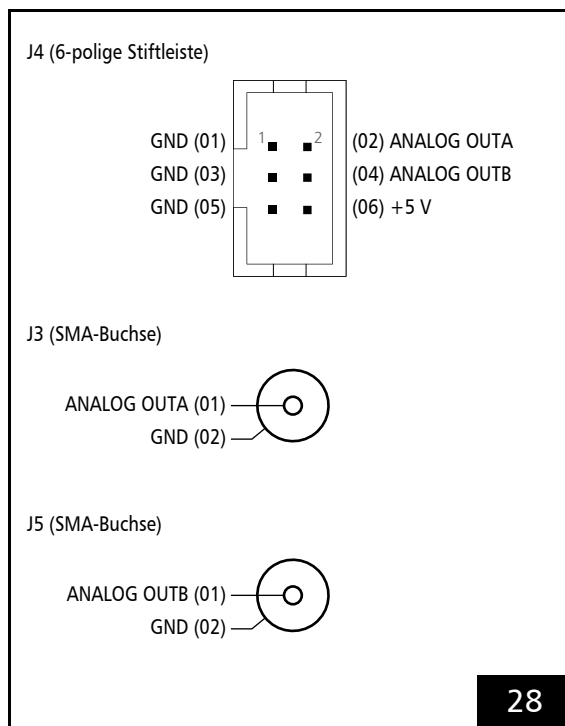
Beim Abarbeiten einer Pixel-Befehlsliste werden die Pixelwerte synchron mit dem Lasertakt am 8-Bit-Digitalausgang (an der EXTENSION 2-Stiftleiste der RTC6 PCI-Express-Karte) ausgegeben.

durch das aktuelle RTC6-Software-Paket unterstützt (d.h. es werden keine speziellen Firmware-dateien etc. vorausgesetzt).

## Hardware-Voraussetzungen

- Aktuell RTC6 PCI-Express-Karte #135427.

Die UFPM-Erweiterungskarte wandelt 8-Bit-Digitalwerte synchron in entsprechende Analogwerte von 0 V...5 V um, und stellt diese an ihren Analogausgängen ANALOG OUTA und ANALOG OUTB zur Verfügung. Die Pinbelegungen der Anschlüsse auf der UFPM-Erweiterungskarte zeigt **Abbildung 28**.



UFPM-Erweiterungskarte: Pinbelegungen der Anschlüsse.

## Technische Spezifikationen

### • Abmessungen

- Länge 52,8 mm
- Breite 48,7 mm

### • Analoge Ausgänge ANALOG OUTA, ANALOG OUTB

- Anschlüsse 6-polige Stiftleiste und zwei SMA-Buchsen (koaxiale Steckverbinder)
- Ausgangsspannungsbereich 0 V...5 V
- Auflösung 8 Bit
- max. Last 5 mA
- Bezugspunkt GND

## 4 Änderungsindex

Inhaltliche Änderungen für Doc. Rev. 1.1.3 (RTC6-Software-Paket V1.4.1)

Seite	Kapitel / Abschnitt / Befehl	Was
7	Kapitel 1.1 "Über dieses Handbuch"	Änderung. Diese Handbuchrevision gilt für das RTC6-Software-Paket V1.4.1 inkl. zugehöriger Komponenten.
9	Kapitel 1.3.3 "Optionale Funktionalitäten"	Änderung. Liste erweitert um die Option "syncA".
13	Kapitel 1.4.4 "Master/Slave-Taktsynchronisierung: Master-Stiftleiste, Slave-Stiftleiste"	Neu. Kapitel eingefügt.
17	Kapitel 1.6.1 "RTC6-Karten und RTC5-Karten im Vergleich – Neuerungen und Änderungen"	Änderung. Liste angepasst: Details zu 3D Korrekturta- bellen (mit RTC6-Software-Paket V1.4.1).
17	Kapitel 1.6.1 "RTC6-Karten und RTC5-Karten im Vergleich – Neuerungen und Änderungen"	Änderung. Liste angepasst: Virtuelles Bildfeld (mit RTC6-Software-Paket V1.4.1 29 Bit).
23	Kapitel 1.8.1 "Neu im RTC6-Befehlssatz"	Änderung. Liste "Allgemein nutzbare RTC6-Befehle" erweitert um: <ul style="list-style-type: none"> <li>• (n_) <b>master_slave_config</b></li> <li>• (n_) <b>spot_distance</b></li> <li>• (n_) <b>spot_distance_ctrl</b></li> </ul>
25	<b>init_rtc6_dll</b>	Änderung. Vollständige Befehls-Tabelle eingefügt.
27	<b>master_slave_config</b>	Neu. Befehls-Tabelle eingefügt. <b>master_slave_config</b> ist neu im RTC6-Software-Paket V1.4.1.
30	<b>set_rtc6_mode</b>	Änderung. Vollständige Befehls-Tabelle eingefügt.
32	<b>spot_distance</b>	Neu. Befehls-Tabelle eingefügt. <b>spot_distance</b> ist neu im RTC6-Software-Paket V1.4.1.
32	<b>spot_distance_ctrl</b>	Neu. Befehls-Tabelle eingefügt. <b>spot_distance_ctrl</b> ist neu im RTC6-Software-Paket V1.4.1.
33	Kapitel 1.8.2 "Geändert im RTC6-Befehlssatz"	Änderung. Liste erweitert um: <ul style="list-style-type: none"> <li>• (n_) <b>get_sync_status</b></li> <li>• (n_) <b>number_of_correction_tables</b></li> <li>• (n_) <b>set_auto_laser_control</b></li> <li>• (n_) <b>set_auto_laser_params</b></li> <li>• (n_) <b>set_auto_laser_params_list</b></li> </ul>

Seite	Kapitel / Abschnitt / Befehl	Was
35	<b>get_sync_status</b>	Neu. Befehls-Tabelle eingefügt. <b>get_sync_status</b> (siehe dort Zeile "RTC5→RTC6") ist im RTC6-Software-Paket V1.4.1 anders im Vergleich zum gleichnamigen RTC5-Befehl.
39	<b>number_of_correction_tables</b>	Neu. Befehls-Tabelle eingefügt. <b>number_of_correction_tables</b> (siehe dort Zeile "RTC5→RTC6") ist im RTC6-Software-Paket V1.4.1 anders im Vergleich zum gleichnamigen RTC5-Befehl.
40	<b>set_auto_laser_control</b>	Neu. Befehls-Tabelle eingefügt. <b>set_auto_laser_control</b> (siehe dort Zeile "RTC5→RTC6") ist im RTC6-Software-Paket V1.4.1 anders im Vergleich zum gleichnamigen RTC5-Befehl.
45	<b>set_auto_laser_params</b>	Neu. Befehls-Tabelle eingefügt. <b>set_auto_laser_params</b> (siehe dort Zeile "RTC5→RTC6") ist im RTC6-Software-Paket V1.4.1 anders im Vergleich zum gleichnamigen RTC5-Befehl.
45	<b>set_auto_laser_params_list</b>	Neu. Befehls-Tabelle eingefügt. <b>set_auto_laser_params_list</b> (siehe dort Zeile "RTC5→RTC6") ist im RTC6-Software-Paket V1.4.1 anders im Vergleich zum gleichnamigen RTC5-Befehl.
52	Kapitel 1.9 "Master/Slave-Betrieb"	Neu. Kapitel eingefügt. RTC6-Software-Paket V1.4.1 unterstützt nun Master/Slave-Synchronisation vollständig.
55	Kapitel 1.10 "Ergänzende Informationen zu den RTC6-Pixel-Modi"	Neu. Kapitel eingefügt.
60	Kapitel 1.11 "Hinweise zu set_auto_laser_control mit Ctrl = 7 ("Spot Distance Control")"	Neu. Kapitel eingefügt.
62	Kapitel 1.12 "Technische Spezifikationen der RTC6 PCI-Express-Karte"	Änderung. Virtuelles Bildfeld (mit RTC6-Software-Paket V1.4.1 29 Bit).
73	Kapitel 2.2.5 "SCANHEADs-Stiftleiste"	Änderung. In <b>Abbildung 15</b> sind die DATA OUT 1/2-Vorzeichen nun analog zum RTC5-Handbuch dokumentiert.
76	Kapitel 2.2.8 "SPI/ANA/UART-Stiftleiste"	Änderung. Information zum Gegen-Stecker hinzugefügt.
77	Kapitel 2.2.9 "STEPPER-Stiftleiste"	Änderung. Information zum Gegen-Stecker hinzugefügt.
77	Kapitel 2.2.10 "MOF-Stiftleiste"	Änderung. Information zum Gegen-Stecker hinzugefügt.
78	Kapitel 2.2.11 "EXTENSION 1-Stiftleiste"	Änderung. Information zum Gegen-Stecker hinzugefügt.
80	Kapitel 2.2.12 "EXT. 2-Stiftleiste"	Änderung. Information zum Gegen-Stecker hinzugefügt.
117	Kapitel 2.7 "Technische Spezifikationen der RTC6 Ethernet-Karte"	Änderung. Virtuelles Bildfeld (mit RTC6-Software-Paket V1.4.1 29 Bit).
124	Kapitel 3 "Anhang B: Die UFPM-Erweiterungskarte"	Neu. Kapitel eingefügt.
127	Kapitel 4 "Änderungsindex"	Neu. Kapitel eingefügt.