



PCI4 SDK Programmer's Reference

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

Contents

| | |
|---|----|
| General Information | 1 |
| About this manual | 1 |
| The PCI4 API | 2 |
| API Function Quick Reference List | 2 |
| pci4_auto_count_channel | 3 |
| pci4_auto_counts_board | 4 |
| pci4_available_boards | 5 |
| pci4_configure_channel | 6 |
| pci4_get_count | 8 |
| pci4_get_status | 9 |
| pci4_isready | 10 |
| pci4_start | 11 |
| Visual C++ Notes | 12 |
| Borland C++ Notes | 12 |
| Visual Basic Notes | 12 |

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

General Information

The PCI4 SDK is provided for integration with customer software to simplify use of the PCI4 transducer interface. The PCI4 SDK includes a driver package for the PCI4 transducer interface, a 32-bit Windows DLL, sample programs written in Microsoft Visual C++, Microsoft Visual Basic, and Borland C++, and this document.

About this manual

This manual provides a reference for the PCI4 API, descriptions of API functionality, and instructions for use of the PCI4 DLL.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

The PCI4 API

The PCI4 API provides functions to allow use of the PCI4 transducer interface.

API Function Quick Reference List

| API Function Name | Purpose | Pg |
|--------------------------------------|--|----|
| <code>pci4_auto_count_channel</code> | Easily update position on one PCI4 channel. | 3 |
| <code>pci4_auto_counts_board</code> | Easily update position data for all channels of a PCI4 board. | 4 |
| <code>pci4_available_boards</code> | Determine the number of PCI4 cards installed in the system. | 5 |
| <code>pci4_configure_channel</code> | Configure a PCI4 channel before use. | 6 |
| <code>pci4_get_count</code> | Determine the count returned by a PCI4 channel. | 8 |
| <code>pci4_get_status</code> | Determine the status of a PCI4 channel. | 9 |
| <code>pci4_isready</code> | Determine whether good position data is available from a PCI4 channel. | 10 |
| <code>pci4_start</code> | Begin interrogation of a PCI4 channel. | 11 |

API Usage Overview

Before position can be returned, the channels of the PCI4 you plan to use must be configured. The sequence of events required to retrieve position from a transducer is the following:

1. Configure the channel with a call to `pci4_configure_channel`.
2. Interrogate the transducer with a call to `pci4_start`.
3. Wait until one of the following occurs:
 - a. Timeout with no position returned. Return to step 1 (multi-magnet) or 2 (single magnet) to retry.
 - b. `pci4_isready` indicates new position information is available. Continue with step 4.
4. Read the channel's status with a call to `pci4_get_status` if desired.
5. Read the channel's position with a call to `pci4_get_position`.
6. If this channel is using a multi-magnet transducer, return to step 1 to configure the channel for a different magnet. If the channel is using a single-magnet transducer, return to step 2 to begin requesting new position information.

For easier retrieval of positions, `pci4_auto_count_channel` or `pci4_auto_counts_board` can be used. These functions access a state machine that automatically follows the steps outlined above.

The channels of the PCI4 operate in parallel; you can perform the above steps synchronously, asynchronously, or serially for each channel.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

```
pci4_auto_count_channel
```

Syntax

```
S32 pci4_auto_count_channel(  
    S32 board,  
    S32 chan,  
    S32 *count  
);
```

Description

Automatically update the count of a selected board and channel.

Parameters

| | |
|-------|---|
| board | The PCI4 board to read |
| chan | The channel of the selected board to read |
| count | A pointer to a 32-bit buffer where the transducer count of will be stored |

Returns

| | |
|----|--------------------|
| -1 | Invalid parameters |
| 0 | Failure |
| 1 | Success |

Notes

This function will update the count information when good data is available and interrogate the selected channel when good data is received or after 20ms of no response.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_auto_counts_board

Syntax

```
S32 pci4_auto_counts_board(  
    S32 board,  
    S32 *counts  
);
```

Description

Automatically update the counts of all channels from a selected PCI4 board.

Parameters

| | |
|--------|--|
| board | The PCI4 board to read |
| counts | A pointer to a 16-byte buffer where the transducer counts will be stored |

Returns

| | |
|----|--------------------|
| -1 | Invalid Parameters |
| 0 | Failure |
| 1 | Success |

Notes

This function will update the count information when good data is available and interrogate the selected channel when good data is received or after 20ms of no response.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_available_boards

Syntax

```
S32 pci4_available_boards(  
    );
```

Description

Returns the number of PCI4 boards detected in the PC. **This function also resets the configuration for all channels of all PCI4 boards.** If this function is used after channel configuration, you must reconfigure any used channels before attempting to retrieve position information.

Parameters

none

Returns

The number of PCI4 boards detected in the PC.

Notes

none

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_configure_channel

Syntax

```
S32 pci4_configure_channel(
    S32 board,
    S32 chan,
    S32 xdcr_type,
    S32 recircs_ssibits,
    float gradient_resolution,
    float scale,
    float offset
);
```

Description

Configures a channel of a PCI4 board.

Parameters

| | | | | | | | | | | | |
|---------------------|---|---|---------------|---|------------|---|-----|---|-----------------------------|---|-------------------------|
| board | The PCI4 board to configure | | | | | | | | | | |
| chan | The channel of the selected board to configure | | | | | | | | | | |
| xdcr_type | The type of transducer to configure the channel to be: <table> <tr> <td>0</td> <td>No transducer</td> </tr> <tr> <td>1</td> <td>Start/Stop</td> </tr> <tr> <td>2</td> <td>SSI</td> </tr> <tr> <td>3</td> <td>Pulse width modulated (PWM)</td> </tr> <tr> <td>4</td> <td>Multi-magnet Start/Stop</td> </tr> </table> | 0 | No transducer | 1 | Start/Stop | 2 | SSI | 3 | Pulse width modulated (PWM) | 4 | Multi-magnet Start/Stop |
| 0 | No transducer | | | | | | | | | | |
| 1 | Start/Stop | | | | | | | | | | |
| 2 | SSI | | | | | | | | | | |
| 3 | Pulse width modulated (PWM) | | | | | | | | | | |
| 4 | Multi-magnet Start/Stop | | | | | | | | | | |
| recircs_ssibits | If xdcr_type is set to indicate a start/stop transducer, recircs_ssibits indicates the number of recirculations desired, 1 – 16. If xdcr_type is set to indicate a SSI transducer, recircs_ssibits indicates the SSI bitlength, 24 – 25. If xdcr_type is set to indicate a PWM transducer, recircs_ssibits indicates the number of recirculations performed in the head of the transducer. If xdcr_type is set to indicate a multi-magnet start/stop transducer, recircs_ssibits indicates the magnet to be interrogated, 1 – 16. | | | | | | | | | | |
| gradient_resolution | If xdcr_type is set to indicate a SSI transducer, gradient_resolution specifies the resolution of the transducer. Otherwise, gradient_resolution specified the gradient of the transducer in microseconds per inch. | | | | | | | | | | |
| scale | The transducer position is multiplied by scale when requests for floating-point positions are answered. | | | | | | | | | | |
| offset | The offset is subtracted from the scaled position when answering requests for floating point positions. | | | | | | | | | | |

Returns

| | |
|---|--------------------|
| 0 | Invalid parameters |
| 1 | Success |

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

Notes

This function must be called before using a channel with functions such as `pci4_start` and `pci4_get_count`.

The additional parameters `gradient_resolution`, `scale`, and `offset` are not currently used, but still must be filled with legal data.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_get_count

Syntax

```
S32 pci4_get_count(  
    S32 board,  
    S32 chan  
);
```

Description

Returns the count from the transducer connected to the selected board and channel.

Parameters

| | |
|-------|---|
| board | The PCI4 board to read |
| chan | The channel of the selected board to read |

Returns

| | |
|-----------|--------------------------------|
| -1 | Invalid parameters |
| Otherwise | The count from the transducer. |

Notes

The count from the transducer is not scaled in any way. The count is generated using a 56 MHz oscillator unless specified otherwise by the customer. The count returned by pci4_get_count is undefined if pci4_start is not called prior to pci4_get_count.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_get_status

Syntax

```
S32 pci4_get_status(  
    S32 board,  
    S32 chan  
);
```

Description

Returns the status byte from the selected board and channel.

Parameters

| | |
|-------|---|
| board | The PCI4 board to read |
| chan | The channel of the selected board to read |

Returns

| | |
|-----------|--|
| -1 | Invalid parameter |
| Otherwise | Returns the status byte as described in the PCI4 Manual. |

Notes

See pci4_isready.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_isready**Syntax**

```
S32 pci4_isready(  
    S32 board,  
    S32 chan  
);
```

Description

Use to determine if a channel has position data ready.

Parameters

| | |
|-------|---|
| board | The PCI4 board to read |
| chan | The channel of the selected board to read |

Returns

| | |
|----|---|
| -1 | parameter error |
| 0 | Call successful, position data is not ready |
| 1 | Call successful, position data is ready |

Notes

See `pci4_get_status`.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

pci4_start

Syntax

```
S32 pci4_start(  
    S32 board,  
    S32 channel,  
    S32 recircs_ssibits  
);
```

Description

Begins interrogation of the specified channel's transducer.

Parameters

| | |
|-----------------|---|
| board | Indicates the PCI4 board to be used. This must be a value ranging from 0 through one less than the number of PCI4 boards installed in the PC. |
| channel | Selects the channel to be used. This must be a value ranging from 0 through 3. |
| recircs_ssibits | Specifies whether to use the stored settings for, or replace the number of recirculations, the magnet number, or the number of SSI bits, depending on the type of transducer. A value of -1 indicates that the previous setting will be used. Otherwise, a legal value must be provided as shown in the pci4_configure_channel section. |

Returns

| | |
|----|--------------------|
| -1 | Invalid parameters |
| 0 | Failure |
| 1 | Success |

Notes

This function will begin interrogation regardless of the current state of the channel. The pci4_get_status function can be used to determine the state of the channel.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com

Visual C++ Notes

The PCI4 SDK has been tested with Microsoft Visual C++ 6.0. Include pci4.h in any modules you wish to call the PCI4 SDK API from. Add pci4_vc.lib to projects you wish to call the PCI4 SDK API from.

Borland C++ Notes

The PCI4 SDK has been tested with Borland C++ 5.02. Include pci4.h in any modules you wish to call the PCI4 SDK API from, and add pci4_bc.lib to projects you wish to call the PCI4 SDK API from.

Visual Basic Notes

Integration with Microsoft Visual Basic requires several considerations. The function API can be loaded by adding the module pci4.bas to your project.

Calls to pci4_get_counts_board must pass the first member of the array instead of the array.

Rapid Controls Inc.

Box 8390 ▪ Rapid City, SD ▪ 57709

Phone: 605-348-7688 ▪ Fax: 605-341-5496

<http://www.rapidcontrols.com/> ▪ email: info@rapidcontrols.com