

DS2003 Multi-Channel A/D Converter Board

RTLib Reference

Release 2021-A – May 2021

How to Contact dSPACE

Mail:	dSPACE GmbH Rathenaustraße 26 33102 Paderborn Germany
Tel.:	+49 5251 1638-0
Fax:	+49 5251 16198-0
E-mail:	info@dspace.de
Web:	http://www.dspace.com

How to Contact dSPACE Support

If you encounter a problem when using dSPACE products, contact your local dSPACE representative:

- Local dSPACE companies and distributors: <http://www.dspace.com/go/locations>
- For countries not listed, contact dSPACE GmbH in Paderborn, Germany.
Tel.: +49 5251 1638-941 or e-mail: support@dspace.de

You can also use the support request form: <http://www.dspace.com/go/supportrequest>. If you are logged on to mydSPACE, you are automatically identified and do not need to add your contact details manually.

If possible, always provide the relevant dSPACE License ID or the serial number of the CmContainer in your support request.

Software Updates and Patches

dSPACE strongly recommends that you download and install the most recent patches for your current dSPACE installation. Visit <http://www.dspace.com/go/patches> for software updates and patches.

Important Notice

This publication contains proprietary information that is protected by copyright. All rights are reserved. The publication may be printed for personal or internal use provided all the proprietary markings are retained on all printed copies. In all other cases, the publication must not be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written consent of dSPACE GmbH.

© 2001 - 2021 by:
dSPACE GmbH
Rathenaustraße 26
33102 Paderborn
Germany

This publication and the contents hereof are subject to change without notice.

AUTERA, ConfigurationDesk, ControlDesk, MicroAutoBox, MicroLabBox, SCALEXIO, SIMPHERA, SYNECT, SystemDesk, TargetLink and VEOS are registered trademarks of dSPACE GmbH in the United States or other countries, or both. Other brand names or product names are trademarks or registered trademarks of their respective companies or organizations.

Contents

About This Reference	5
Macros	7
Base Address of the I/O Board.....	7
Board Initialization	9
ds2003_board_init.....	9
ADC Unit	13
ds2003_set_range.....	14
ds2003_set_trigger.....	16
ds2003_set_wordlen.....	17
ds2003_init_scantbl.....	19
ds2003_start.....	21
ds2003_conversion_start.....	22
ds2003_buffer_clear.....	24
ds2003_in.....	25
ds2003_read.....	27
ds2003_block_read.....	30
ds2003_fifo_half_full_block_read.....	33
ds2003_ready.....	36
Function Execution Times	39
Information on the Test Environment.....	39
Measured Execution Times.....	40
Index	43

About This Reference

Content

This RTLib Reference (Real-Time Library) gives detailed descriptions of the C functions needed to program a DS2003 Multi-Channel A/D Converter Board. The C functions can be used to program RTI-specific Simulink S-functions, or to implement your control models manually using C programs.

Symbols

dSPACE user documentation uses the following symbols:

Symbol	Description
	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
	Indicates a hazard that, if not avoided, could result in property damage.
	Indicates important information that you should take into account to avoid malfunctions.
	Indicates tips that can make your work easier.
	Indicates a link that refers to a definition in the glossary, which you can find at the end of the document unless stated otherwise.
	Precedes the document title in a link that refers to another document.

Naming conventions

dSPACE user documentation uses the following naming conventions:

%name% Names enclosed in percent signs refer to environment variables for file and path names.

< > Angle brackets contain wildcard characters or placeholders for variable file and path names, etc.

Special folders

Some software products use the following special folders:

Common Program Data folder A standard folder for application-specific configuration data that is used by all users.

%PROGRAMDATA%\dSPACE\<InstallationGUID>\<ProductName>

or

%PROGRAMDATA%\dSPACE\<ProductName>\<VersionNumber>

Documents folder A standard folder for user-specific documents.

%USERPROFILE%\Documents\dSPACE\<ProductName>\<VersionNumber>

Local Program Data folder A standard folder for application-specific configuration data that is used by the current, non-roaming user.

%USERPROFILE%\AppData\Local\dSPACE\<InstallationGUID>\<ProductName>

Accessing dSPACE Help and PDF Files


After you install and decrypt dSPACE software, the documentation for the installed products is available in dSPACE Help and as PDF files.

dSPACE Help (local) You can open your local installation of dSPACE Help:

- On its home page via Windows Start Menu
- On specific content using context-sensitive help via **F1**

dSPACE Help (Web) You can access the Web version of dSPACE Help at www.dspace.com.

To access the Web version, you must have a *mydSPACE* account.

PDF files You can access PDF files via the  icon in dSPACE Help. The PDF opens on the first page.

Macros

Introduction

The base address of an I/O board in a PHS-bus-based system has to be defined by using the `DSxxxx_n_BASE` macro.

Base Address of the I/O Board

DSxxxx_n_BASE Macros

When using I/O board functions, you always need the board's base address as a parameter. This address can easily be obtained by using the `DSxxxx_n_BASE` macros, where `DSxxxx` is the board name (for example, `DS2001`) and `n` is an index which counts boards of the same type. The board with the lowest base address is given index 1. The other boards of the same type are given consecutive numbers in order of their base addresses.

The macros reference an internal data structure which holds the addresses of all I/O boards in the system. The initialization function of the processor board (named `init`) creates this data structure. Hence, when you change an I/O board base address, it is not necessary to recompile the code of your application. For more information on the processor board's initialization function, refer to [ds1006_init \(DS1006 RTLib Reference\)](#) or [init \(DS1007 RTLib Reference\)](#).

Note

The `DSxxxx_n_BASE` macros can be used only after the processor board's initialization function `init` is called.

Example

This example demonstrates the use of the `DSxxxx_n_BASE` macros. There are two `DS2001` boards, two `DS2101` boards, and one `DS2002` board connected to a PHS bus. Their base addresses have been set to different addresses. The following table shows the I/O boards, their base addresses, and the macros which can be used as base addresses:

Board	Base Address	Macro
DS2001	00H	DS2001_1_BASE
DS2002	20H	DS2002_1_BASE
DS2101	80H	DS2101_1_BASE
DS2001	90H	DS2001_2_BASE
DS2101	A0H	DS2101_2_BASE

Board Initialization

Introduction

Before you can use the DS2003, you have to perform the initialization process.

Note
The processor board must be initialized before you can initialize the DS2003.

Where to go from here

Information in this section

To initialize the DS2003
[ds2003_board_init..... 9](#)

ds2003_board_init

Syntax

`void ds2003_board_init(phis_addr_t base)`

Include file

`ds2003.h`

Purpose

To initialize the DS2003.

Description

All DS2003 registers are initialized to default values:

- -10 ... +10 V input voltage range
- 16-bit word length
- Disable external trigger and SYNCIN

Note

This function must be called before any other DS2003 function can be used.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
201	Error	ds2003_init(): Invalid PHS-bus base address 0x????????	The value of the base parameter is not a valid PHS-bus address. This error may be caused if the PHS-bus connection of the I/O board is missing. Check the connection.
-144	Error	ds2003_init(0x??): Board not found!	No DS2003 board could be found at the specified PHS-bus address. Check if the DSxxxx_n_BASE macro corresponds to the I/O board used.
-145	Error	ds2003_init(0x??): Memory allocation error!	The allocation of some dynamic memory for internal data storage has failed.
-53	Warning	ds2003_init(0x??): Jumper setting is not matching SW default initialization! STP register: 0x???????? instead of 0x????????	The value of the STP register could not be verified successfully. May be the DS2003 jumper setting is not correct. Remove all jumpers.

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to initialize a DS2003:

```
void main(void)
{
    init();
    ds2003_board_init(DS2003_1_BASE);
    ...
}
```

Related topics

References

Base Address of the I/O Board.....	7
Macros.....	7

ADC Unit

Objective

The following functions are used to program the A/D converter.

Note
You have to initialize the DS2003 with the `ds2003_board_init` function before you can use one of these functions.

Where to go from here	Information in this section
	To initialize the A/D converter
	ds2003_set_range..... 14
	ds2003_set_trigger..... 16
	ds2003_set_wordlen..... 17
	To initialize the scan table
	ds2003_init_scantbl..... 19
	To start the conversion on the selected DS2003
	ds2003_start..... 21
	ds2003_conversion_start..... 22
	To clear the buffer
	ds2003_buffer_clear..... 24

To read ADC values	
ds2003_in.....	25
ds2003_read.....	27
ds2003_block_read.....	30
ds2003_fifo_half_full_block_read.....	33
To poll the scan process completed flag	
ds2003_ready.....	36

Information in other sections

[ADC Unit \(DS2003 Features !\[\]\(eafc244b53721dd1ec133f0772f70fc7_img.jpg\)\)](#)

ds2003_set_range

Syntax

```
void ds2003_set_range(
    phs_addr_t base,
    int channel,
    int range)
```

Include file

ds2003.h

Purpose

To set the input voltage range of the A/D channels.

Description

This function selects the input voltage range of the specified channel on the DS2003 at the PHS-bus base address.

The input voltage range cannot be set on hardware for each of the 32 channels, because the DS2003 contains only 2 A/D converters with multiplexer. The range setting is stored in an internal data structure and is taken into account by the ADC when the respective channel is converted.

Note

- The `ds2003_board_init` function must be called before this function can be used.
- This function must be called before you initialize the scan table via the `ds2003_init_scantbl` function.

I/O mapping

For details on the I/O mapping, refer to [ADC Unit \(DS2003 Features !\[\]\(9dfdaff1d86ba3c1f8353b4d1b61b8c5_img.jpg\)\)](#).

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

channel Specifies the channel number within the range 1 ... 32. The `DS2003_CH_ALL` symbol selects all 32 channels.

range Specifies the input voltage range. The following symbols are predefined:

Symbol	Input Voltage Range
<code>DS2003_RNG5</code>	–5 ... +5 V
<code>DS2003_RNG10</code>	–10 ... +10 V

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	<code>ds2003_set_range(0x??): Board not initialized!</code>	The DS2003 has not been initialized by a preceding call to the <code>ds2003_board_init</code> function.

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to use the function:

```
ds2003_set_range(DS2003_1_BASE, 1, DS2003_RNG5);
```

Channel 1 of the DS2003 is set to the –5 ... +5 V input voltage range.

Related topics**References**

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_init_scantbl.....	19
Macros.....	7

ds2003_set_trigger

Syntax

```
void ds2003_set_trigger(  
    phs_addr_t base,  
    int trigger)
```

Include file

ds2003.h

Purpose

To enable the external trigger and SYNCIN signal of the PHS bus.

Description

This function enables or disables the external trigger of the DS2003 by external input or PHS-bus SYNCIN signal.

Note

The `ds2003_board_init` function must be called before this function can be used.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

trigger Specifies the status of the external trigger input. The following symbols are predefined:

Symbol	External Trigger Input
DS2003_TRIGON	Enabled
DS2003_TRIGOFF	Disabled

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2003_set_trigger(0x??): Board not initialized!	The DS2003 has not been initialized by a preceding call to the <code>ds2003_board_init</code> function.

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to use the function:

```
ds2003_set_trigger(DS2003_1_BASE, DS2003_TRIGON);
```

The external trigger input of the DS2003 is enabled.

Related topics**References**

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
Macros.....	7

ds2003_set_wordlen

Syntax

```
void ds2003_set_wordlen(
    phs_addr_t base,
    int channel,
    int wordlen)
```

Include file

ds2003.h

Purpose

To select the ADC word length.

Description

This function selects the data word length of the specified channel on the DS2003 at the PHS-bus base address.

The word length cannot be set on hardware for each of the 32 channels, because the DS2003 contains only 2 A/D converters with a multiplexer. The word length setting is stored in an internal data structure and is taken into account by the ADC when the respective channel is converted. Using a word length less

than 16 bit makes the A/D conversion faster. For further information, refer to [Faster A/D Conversion via Short-Cycling \(DS2003 Features !\[\]\(529949c2c3dadbaa4e538e8c643454bc_img.jpg\)](#)).

Note

- The `ds2003_board_init` function must be called before this function can be used.
- This function must be called before you initialize the scan table via the `ds2003_init_santbl` function.

I/O mapping

For details on the I/O mapping, refer to [ADC Unit \(DS2003 Features !\[\]\(339a16584d5da0f0a3ca4e9ec17bf6a1_img.jpg\)](#)).

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

channel Specifies the channel number within the range 1 ... 32. The `DS2003_CH_ALL` symbol selects all 32 channels.

wordlen Specifies the word length of ADC data in bit. The following symbols are predefined:

Symbol	Word Length in Bit
<code>DS2003_LEN4</code>	4
<code>DS2003_LEN8</code>	8
<code>DS2003_LEN10</code>	10
<code>DS2003_LEN12</code>	12
<code>DS2003_LEN13</code>	13
<code>DS2003_LEN14</code>	14
<code>DS2003_LEN15</code>	15
<code>DS2003_LEN16</code>	16

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	<code>ds2003_set_wordlen(0x??): Board not initialized!</code>	The DS2003 has not been initialized by a preceding call to the <code>ds2003_board_init</code> function.

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to set all channels of a DS2003 to a word length of 10-bit ADC.

```
ds2003_set_wordlen(
    DS2003_1_BASE,
    DS2003_CH_ALL,
    DS2003_LEN10);
```

Related topics**Basics**

[Faster A/D Conversion via Short-Cycling \(DS2003 Features !\[\]\(cbe2492b119e39e02a1dab2af4a4b296_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_init_scantbl.....	19
Macros.....	7

ds2003_init_scantbl

Syntax

```
void ds2003_init_scantbl(
    phs_addr_t base,
    int count,
    int *channels)
```

Include file

ds2003.h

Purpose

To initialize the DS2003 scan table.

Description

Depending on the specified parameters, this function generates an appropriate scan table which is downloaded to the DS2003.

The number of input channels to be used is selected by the **count** parameter.

The **channels** parameter is a pointer to an array of channel numbers. This array must contain at least as many channel numbers as given by the **count** parameter in an arbitrary sequence. Multiple occurrences of the same channel number are possible. The **channels** array must be allocated and filled by the calling program.

The resolutions and the input voltage ranges for the respective channels can be set using the **ds2003_set_range** and **ds2003_set_wordlen** functions.

The scan table is organized in channel pairs where the resolution is specified in common for both ADCs. Inputs being converted with a higher resolution than specified are automatically masked to the required resolution.

Note

- The `ds2003_board_init` function must be called before this function can be used.
- If you want to specify the resolution and the input voltage range using `ds2003_set_range` and `ds2003_set_wordlen`, you must call these functions before you initialize the scan table via `ds2003_init_scantbl`.
- This function must be called before one of the read functions (`ds2003_in`, `ds2003_block_read`, `ds2003_read`, `ds2003_fifo_half_full_block_read`) can be used.
- If you want to use one of the DS2003 hardware interrupts, `ds2003_init_scantbl` must be called before initializing the interrupt, otherwise the execution of this function fails and causes an error message. For further information, refer to [Limitations \(DS2003 Features !\[\]\(633dd45d48d71eb51a85c6dd83ee51e9_img.jpg\)\)](#).
- The DS2003 data buffer is cleared.

I/O mapping

For details on the I/O mapping, refer to [ADC Unit \(DS2003 Features !\[\]\(23d9fc146e83b5c3013cfa32c784f8d5_img.jpg\)\)](#).

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

count Specifies the number of channels being used within the range 1 ... 32.

channels Specifies the pointer to the array of channel numbers used for the scan table. Valid channel numbers are 1 ... 32.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2003_init_scantbl(0x??): Board not initialized!	The DS2003 has not been initialized by a preceding call to the <code>ds2003_board_init</code> function.
-190	Error	ds2003_init_scantbl(0x??): Board is not responding!	The scan process completed flag of the DS2003 was not set during a time period of 0.1 seconds. This error can occur if the <code>ds2003_init_scantbl</code> function is called after the DS2003 interrupt controller has been initialized to a non-polling mode. It can also be caused by a hardware failure of the DS2003.

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to use the function:

```
/* select channels 1, 5, 11, 20 and 32 */
int channels[5] = {1, 5, 11, 20, 32};
int count = 5; /* account of used channels */
void main(void)
{
    ...
    ds2003_init_scantbl(DS2003_1_BASE, count, channels);
    ...
}
```

Channels 1, 5, 11, 20 and 32 are used for input.

Related topics**Basics**

[Limitations \(DS2003 Features !\[\]\(8bba887393ca45b761e5cb49e755e762_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_block_read.....	30
ds2003_board_init.....	9
ds2003_fifo_half_full_block_read.....	33
ds2003_in.....	25
ds2003_read.....	27
ds2003_ready.....	36
ds2003_set_range.....	14
ds2003_set_wordlen.....	17
ds2003_start.....	21
Macros.....	7

ds2003_start

Syntax

```
void ds2003_start(phs_addr_t base)
```

Include file

ds2003.h

Purpose

To clear the FIFO buffer and start the conversion.

Description

This function is used to start the conversion of a complete data set as initialized by the `ds2003_init_scantbl` function.

Note

- The DS2003 data buffer is cleared by this function and thus old conversion results that have not been read out are lost.
- The `ds2003_board_init` function must be called before this function can be used.
- The scan table must have been properly initialized by a preceding call to the `ds2003_init_scantbl` function.
- Use this function only in combination with the `ds2003_in` and `ds2003_read` functions.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

Return value

None

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to start the conversion on the DS2003:

```
ds2003_start(DS2003_1_BASE);
```

Related topics**References**

Base Address of the I/O Board	7
ds2003_board_init	9
ds2003_conversion_start	22
ds2003_in	25
ds2003_init_scantbl	19
ds2003_read	27
ds2003_ready	36
Macros	7

ds2003_conversion_start

Syntax

```
void ds2003_conversion_start(phs_addr_t base)
```

Include file	ds2003.h										
Purpose	To start the conversion without clearing the FIFO buffer.										
Description	<p>This function is used to start the conversion of a complete data set as initialized by the <code>ds2003_init_scantbl</code> function. Old conversion results remain unchanged.</p> <div> <p>Note</p> <ul style="list-style-type: none"> ▪ The <code>ds2003_board_init</code> function must be called before this function can be used. ▪ The scan table must have been properly initialized by a preceding call to the <code>ds2003_init_scantbl</code> function. ▪ Use this function only in combination with the <code>ds2003_block_read</code> and <code>ds2003_fifo_half_full_block_read</code> functions. </div>										
Parameters	<p>base Specifies the PHS-bus base address. Refer to Base Address of the I/O Board on page 7.</p>										
Return value	None										
Example	<p>The following example shows how to start the conversion on a DS2003 board.</p> <pre>... ds2003_conversion_start(DS2003_1_BASE); ...</pre>										
Related topics	<p>References</p> <table> <tr> <td>Base Address of the I/O Board.....</td> <td>7</td> </tr> <tr> <td>ds2003_block_read.....</td> <td>30</td> </tr> <tr> <td>ds2003_board_init.....</td> <td>9</td> </tr> <tr> <td>ds2003_fifo_half_full_block_read.....</td> <td>33</td> </tr> <tr> <td>ds2003_init_scantbl.....</td> <td>19</td> </tr> </table>	Base Address of the I/O Board.....	7	ds2003_block_read.....	30	ds2003_board_init.....	9	ds2003_fifo_half_full_block_read.....	33	ds2003_init_scantbl.....	19
Base Address of the I/O Board.....	7										
ds2003_block_read.....	30										
ds2003_board_init.....	9										
ds2003_fifo_half_full_block_read.....	33										
ds2003_init_scantbl.....	19										

ds2003_buffer_clear

Syntax

```
void ds2003_buffer_clear(phs_addr_t base)
```

Include file

ds2003.h

Purpose

To clear the FIFO buffer.

Description

If you use the `ds2003_conversion_start` function the FIFO buffer is not cleared before starting the conversion. You can use the `ds2003_buffer_clear` function to clear the FIFO buffer and internal stored data (see also `ds2003_block_read`).

If you use the `ds2003_start` function the FIFO buffer is cleared before starting the conversion.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

Return value

None

Example

The following example shows how to clear the FIFO buffer.

```
...
ds2003_buffer_clear(DS2003_1_BASE);
...
```

Related topics

References

Base Address of the I/O Board	7
ds2003_block_read	30
ds2003_conversion_start	22
ds2003_start	21

ds2003_in

Syntax

```
void ds2003_in(
    phs_addr_t base,
    dsfloat *data)
```

Include file

ds2003.h

Purpose

To read one data record waiting on the *buffer not empty* flag.

Description

This function reads input data from the DS2003 data buffer and stores it in an array pointed to by the **data** parameter. The function waits until the first conversion results are available. Then it reads the results while the next conversion is in process. This will be done until all conversions specified by the scan table are completed.

The values read from the DS2003 data buffer are converted to float values in the range $-1.0 \dots +1.0$. The results are stored in the **data** array in the same sequence as the channel numbers in the channel number array **channels** of the **ds2003_init_scantbl** function.

Note

- The **data** array must be allocated by the calling program. Its length must be equal to or greater than the **channels** parameter specified during initialization.
- The **ds2003_board_init** function must be called before this function can be used.
- The DS2003 must have been properly initialized by the **ds2003_init_scantbl** function and the conversion process must have been started by **ds2003_start** before the **ds2003_in** function can be used to read the conversion results. Otherwise **ds2003_in** will return unpredictable results or may even block the DSP.
- If you use one of the DS2003 hardware interrupts, the execution of this function fails and blocks the DSP. For further information, refer to [Limitations \(DS2003 Features !\[\]\(896151ec231b70900e969d67696ca48d_img.jpg\)](#)).
- Do not use this function in combination with the **ds2003_conversion_start** function.
- This function is not reentrant.

I/O mapping

For details on the I/O mapping, refer to [ADC Unit \(DS2003 Features !\[\]\(b4eeff342f60cc7bcd67d869b4fedca2_img.jpg\)](#))

Parameters	<p>base Specifies the PHS-bus base address. Refer to Base Address of the I/O Board on page 7.</p> <p>data Specifies the pointer to the destination data array.</p>
Return value	None
Execution times	For information, refer to Function Execution Times on page 39.

Example

This example shows how to use the function:

```
#define CHANNELS 5 /* number of input channels being used */
int chn[CHANNELS] = { 1, 3, 32, 17, 8 };
dsfloat u_in[CHANNELS]; /* ADC input data */

void timerA_interrupt(void) /* sample rate isr */
{
    /* start conversion process*/
    ds2003_start(DS2003_1_BASE);
    /* get conversion results*/
    ds2003_in(DS2003_1_BASE, u_in);
}

void main(void)
{
    init();
    ds2003_board_init(DS2003_1_BASE);
    ds2003_set_range(DS2003_1_BASE, 1, DS2003_RNG5);
    ds2003_set_range(DS2003_1_BASE, 17, DS2003_RNG5);
    ds2003_set_wordlen(DS2003_1_BASE, 1, DS2003_LEN4);
    ds2003_set_wordlen(DS2003_1_BASE, 32, DS2003_LEN8);
    ds2003_set_wordlen(DS2003_1_BASE, 17, DS2003_LEN12);
    ds2003_init_scantbl(DS2003_1_BASE, CHANNELS, chn);
    RTLIB_SRT_START(1.0e-3, timerA_interrupt);

    /* model background loop */
    while(1)
    {
        RTLIB_BACKGROUND_SERVICE(); /* background service */
    }
}
```

In the example, the channel 1 input signal (specified by `chn[0]`) is converted by using an input voltage range of ± 5 V (`DS2003_RNG5`) and 4-bit resolution. The corresponding conversion result is available through `u_in[0]`. A range of ± 10 V and 16-bit resolution are used for channel 3 (specified by `chn[1]`). The result is obtained in `u_in[1]`. The same holds for the remaining channels 32, 17, and 8.

Related topics

Basics

[Limitations \(DS2003 Features !\[\]\(4729e517bc6a7cd81c8025b9646574fb_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_conversion_start.....	22
ds2003_init_scantbl.....	19
ds2003_start.....	21
Macros.....	7

ds2003_read

Syntax

```
void ds2003_read(
    phs_addr_t base,
    dsfloat *data)
```

Include file

ds2003.h

Purpose

To read one data record without polling the *buffer not empty* flag.

Description

This function reads input data from the DS2003 data buffer immediately without polling the *buffer not empty* flag and stores it in an array pointed to by the **data** parameter.

The values read from the DS2003 data buffer are converted to float values in the range $-1.0 \dots +1.0$. The results are stored in the **data** array in the same sequence as the channel numbers in the channel number array **channels** of the **ds2003_init_scantbl** function.

The **data** array must be allocated by the calling program. Its length must be equal to or greater than the **channels** parameter specified during initialization.

This function can be used in an interrupt service routine triggered by the *scan process completed* interrupt. Specify "0" for the slave interrupt number of the **install_phs_int_vector** function. For further information on DS2003 interrupts, refer to [Interrupts \(DS2003 Features !\[\]\(84f47badaad7772cd95667a7c387a639_img.jpg\)\)](#).

Note

- The `ds2003_board_init` function must be called before this function can be used.
- The DS2003 must have been properly initialized by the `ds2003_init_scantbl` function, the conversion process must have been started by `ds2003_start`, and the scan process must be completed. Otherwise `ds2003_read` will return unpredictable results or may even block the DSP.
- Do not use this function in combination with the `ds2003_conversion_start` function.
- This function is not reentrant.

Parameters	base Specifies the PHS-bus base address. Refer to Base Address of the I/O Board on page 7. data Specifies the pointer to the destination data array.
Return value	None
Execution times	For information, refer to Function Execution Times on page 39.

Example

This example shows how to use the function:

```
#define CHANNELS 5 /* number of input channels being used */
int chn[CHANNELS] = { 1, 3, 32, 17, 8 };
dsfloat u_in[CHANNELS]; /* ADC input data */

void adc_service()
{
    /* get conversion results */
    ds2003_read(DS2003_1_BASE, u_in);
}

/* sample rate isr */
void timerA_interrupt(void)
{
    /* start conversion process */
    ds2003_start(DS2003_1_BASE);
}

void main(void)
{
    init();
    ds2003_board_init(DS2003_1_BASE);
    ds2003_set_range(DS2003_1_BASE, 1, DS2003_RNG5);
    ds2003_set_range(DS2003_1_BASE, 17, DS2003_RNG5);
    ds2003_set_wordlen(DS2003_1_BASE, 1, DS2003_LEN4);
    ds2003_set_wordlen(DS2003_1_BASE, 32, DS2003_LEN8);
    ds2003_set_wordlen(DS2003_1_BASE, 17, DS2003_LEN12);
    ds2003_init_scanb1(DS2003_1_BASE, CHANNELS, chn);
    install_phs_int_vector(DS2003_1_BASE, 0, adc_service);
    RTLIB_SRT_START(1.0e-3, timerA_interrupt);

    /* model background loop */
    while(1)
    {
        RTLIB_BACKGROUND_SERVICE(); /* background service */
    }
}
```


In the example, the channel 1 input signal (specified by `chn[0]`) is converted by using an input voltage range of ± 5 V (`DS2003_RNG5`) and 4-bit resolution. The corresponding conversion result is available through `u_in[0]`. A range of ± 10 V and 16-bit resolution are used for channel 3 (specified by `chn[1]`). The result is obtained in `u_in[1]`. The same holds for the remaining channels 32, 17, and 8.

The results are read in an interrupt service routine.

Related topics**Basics**

[Interrupts \(DS2003 Features !\[\]\(eafc244b53721dd1ec133f0772f70fc7_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_conversion_start.....	22
ds2003_init_scantbl.....	19
ds2003_start.....	21
install_phs_int_vector (DS1006 RTLib Reference )	
install_phs_int_vector (DS1007 RTLib Reference )	
Macros.....	7

ds2003_block_read

Syntax

```
void ds2003_block_read(
    phs_addr_t base,
    long record_count,
    long *record_len,
    dsfloat *data)
```

Include file

ds2003.h

Purpose

To read ADC values stored as complete data records.

Description

This function reads the conversion results of one or more scan processes in polling mode. Each new scan process writes a data record with the conversion results into the FIFO buffer of the DS2003. The size of a data record depends on the number of channels which have been specified by the **ds2003_init_scantbl** function.

This function returns only complete data records. If the FIFO buffer is empty before a complete data record has been read out, the incomplete record is stored internally and completed with the next call of this function. If there is a FIFO buffer overflow, the data will be inconsistent and you have to clear the buffer using the **ds2003_buffer_clear** function before reading again. You can detect a FIFO buffer overflow, for example, by using the following source code:

```

state = PHS_REGISTER_READ(DS2003_1_BASE,6);
if (state & 0x00000008) /* if FIFO overflow */
{
    discard_values ...(); /* user function */
    /* clear FIFO buffer */
    ds2003_buffer_clear(DS2003_1_BASE);
    error_message ...(); /* user function */
}

```

The values read from the DS2003 data buffer are converted to float values in the range -1.0 ... +1.0.

Note

- The `ds2003_board_init` function must be called before this function can be used.
- The DS2003 must have been properly initialized by the `ds2003_init_scbtbl` function and the conversion process must have been started by `ds2003_conversion_start` or by an external trigger. Otherwise `ds2003_block_read` will return no results.
- This function must not be used in combination with the `ds2003_start`, `ds2003_fifo_half_full_block_read`, `ds2003_in` or `ds2003_read` functions.
- If you use one of the DS2003 hardware interrupts, the execution of this function fails and blocks the DSP. For further information, refer to [Limitations \(DS2003 Features !\[\]\(9dc885fa0d6d341860a6e69645e59475_img.jpg\)](#)).
- This function is not reentrant.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

record_count Specifies the number of scan table data records to be read from the FIFO buffer. It can be within the range 1 ... 512.

record_len Specifies the pointer to the number of completely read data records.

data Specifies the pointer to the destination data array. The dimension of the array depends on `record_count` and the number of channels specified in the scan table (`dim = channels * record_count`).

The data array must be allocated by the calling program.

Return value

None

Example

This example shows how to use the function:

```
#define CHANNELS 5 /* number of input channels being used */
int chn[CHANNELS] = { 1, 3, 32, 17, 8 };
dsfloat u_in[50]; /* ADC input data */
long records = 10;
long len;

void timerA_interrupt() /* sample rate isr */
{
    /* get conversion results */
    ds2003_block_read(DS2003_1_BASE, records, &len, u_in);
    ...
}

void main(void)
{
    init();
    ds2003_board_init(DS2003_1_BASE);
    /* initialize range and wordlen */
    ds2003_set_range(DS2003_1_BASE, 1, DS2003_RNG5);
    ds2003_set_range(DS2003_1_BASE, 17, DS2003_RNG5);
    ds2003_set_wordlen(DS2003_1_BASE, 1, DS2003_LEN4);
    ds2003_set_wordlen(DS2003_1_BASE, 32, DS2003_LEN8);
    ds2003_set_wordlen(DS2003_1_BASE, 17, DS2003_LEN12);
    /* specify scan table */
    ds2003_init_scantbl(DS2003_1_BASE, CHANNELS, chn);
    /* enable external trigger */
    ds2003_set_trigger(DS2003_1_BASE, DS2003_TRIGON);
    RTLIB_SRT_START(1.0e-3, timerA_interrupt);
    while(1) /* model background loop */
    {
        RTLIB_BACKGROUND_SERVICE(); /* background service */
    }
}
```

In the example, the channel 1 input signal (specified by `chn[0]`) is converted by using an input voltage range of ± 5 V (`DS2003_RNG5`) and 4-bit resolution. The corresponding conversion result is available through `u_in[0 + n * CHANNELS]`, where `n` is the number of the data record. A range of ± 10 V and 16-bit resolution are used for channel 3 (specified by `chn[1]`). The result is obtained in `u_in[1 + n * CHANNELS]`. The same holds for the remaining channels 32, 17, and 8. The conversion must be started by an external trigger signal.

Related topics**Basics**

[Limitations \(DS2003 Features !\[\]\(666e09182d4cd268646ea700ea60dcdf_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_buffer_clear.....	24
ds2003_conversion_start.....	22
ds2003_fifo_half_full_block_read.....	33
ds2003_in.....	25
ds2003_init_scantbl.....	19
ds2003_read.....	27
ds2003_start.....	21

ds2003_fifo_half_full_block_read

Syntax

```
void ds2003_fifo_half_full_block_read(
    phs_addr_t base,
    long record_count,
    long *record_len,
    dsfloat *data)
```

Include file

ds2003.h

Purpose

To read ADC values stored as complete data records within an interrupt service routine.

Description

This function reads the conversion results of one or more scan processes in the interrupt service routine of the FIFO half full interrupt. Each new scan process writes a data record with the conversion results into the FIFO buffer of the DS2003 until the FIFO half full interrupt is requested. The size of a data record depends on the number of channels which have been specified by the `ds2003_init_scantbl` function. This function returns only complete data records.

The values read from the DS2003 data buffer are converted to float values in the range -1.0 ... +1.0.

Note

- The `ds2003_board_init` function must be called before this function can be used.
- The DS2003 must have been properly initialized by the `ds2003_init_scantbl` function and the conversion process must have been started by `ds2003_conversion_start` or by an external trigger. Otherwise `ds2003_fifo_half_full_block_read` will return no results.
- The FIFO half full interrupt must have been initialized before the function can be used to read the conversion results (specify the slave interrupt number parameter of the `install_phs_int_vector` function with 2). Otherwise `ds2003_fifo_half_full_block_read` will return erroneous results.
For further information on DS2003 interrupts, refer to [Interrupts \(DS2003 Features\)](#).
- This function must be used in an interrupt service routine.
- Do not use this function in combination with the `ds2003_start` and `ds2003_block_read` functions.
- This function is not reentrant.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

record_count Specifies the number of scan table data records to be read from the FIFO buffer. It can be within the range 1 ... 256.

record_len Specifies the pointer to the number of completely read data records.

data Specifies the pointer to the destination data array. The dimension of the array depends on **record_count** and the number of channels specified in the scan table ($\text{dim} = \text{channels} \cdot \text{record_count}$).

The data array must be allocated by the calling program.

Return value

None

Example

This example shows how to use the function:

```
#define CHANNELS 5    /* number of input channels being used */
int chn[CHANNELS] = { 1, 3, 32, 17, 8 };
dsfloat u_in[50];    /* ADC input data */
long records = 10;
long len;
```

```

/* FIFO half full interrupt service routine */
void int2_respond()
{
    /* get conversion results */
    ds2003_fifo_half_full_block_read(DS2003_1_BASE, records , &len, u_in);
    ...
}

void main(void)
{
    init();
    ds2003_board_init(DS2003_1_BASE);
    ...
    ds2003_init_scantbl(DS2003_1_BASE, CHANNELS, chn);
    /* enable external trigger */
    ds2003_set_trigger(DS2003_1_BASE, DS2003_TRIGON);
    error = install_phs_int_vector(DS2003_1_BASE, 2, int2_respond);
    if (error)
    {
        msg_error_printf(0, 0, "Error %ld while initializing interrupts", error);
        exit(1);
    }
    while(1) /* model background loop */
    {
        RTLIB_BACKGROUND_SERVICE(); /* background service */
    }
}

```

In the example, the DS2003 board is initialized to start scan processes on external trigger. The FIFO half full interrupt is initialized.

With each interrupt (requested when the FIFO contains 256 data words) the `ds2003_fifo_half_full_block_read` function reads the results of 10 conversion processes with the results of 5 channels each.

The channel 1 input signal (specified by `chn[0]`) is converted and the corresponding conversion result is available through `u_in[0 + n * CHANNELS]`, where `n` is the number of the data record. The same holds for the remaining channels 32, 17, and 8. The conversion must be started by an external trigger signal.

Related topics

Basics

[Interrupts \(DS2003 Features !\[\]\(e474458956c9a37fbf9586ddb60a7fa1_img.jpg\)\)](#)

References

Base Address of the I/O Board.....	7
ds2003_block_read.....	30
ds2003_board_init.....	9
ds2003_conversion_start.....	22
ds2003_init_scantbl.....	19
ds2003_start.....	21
install_phs_int_vector (DS1006 RTLlib Reference )	
install_phs_int_vector (DS1007 RTLlib Reference )	

ds2003_ready

Syntax

```
int ds2003_ready(phis_addr_t base)
```

Include file

ds2003.h

Purpose

To poll the *scan process complete* flag.

Description

The *scan process completed* flag is set when the DS2003 has finished the conversion of all channels specified in the scan table.

Note

The `ds2003_board_init` function must be called before this function can be used.

Parameters

base Specifies the PHS-bus base address. Refer to [Base Address of the I/O Board](#) on page 7.

Return value

The following values are returned:

Value	Description
0	Conversion has not finished
1	Conversion has finished

Execution times

For information, refer to [Function Execution Times](#) on page 39.

Example

This example shows how to use the function:

```
while( !ds2003_ready(DS2003_1_BASE) );
```

The application waits until the conversion is finished.

Related topics

References

Base Address of the I/O Board.....	7
ds2003_board_init.....	9
ds2003_start.....	21
Macros.....	7

Function Execution Times

Objective The execution times of the C functions can vary, since they depend on different factors. The measured execution times are influenced by the test environment used. This section gives you basic information on the test environment and contains the mean function execution times.

Where to go from here	Information in this section
	Information on the Test Environment..... 39
	Measured Execution Times..... 40

Information on the Test Environment

Test environment The execution time of a function can vary, since it depends on different factors, for example:

- CPU clock and bus clock frequency of the processor board used
- Optimization level of the compiler
- Use of inlining parameters

The test programs that are used to measure the execution time of the functions listed below have been generated and compiled with the default settings of the **down<xxx>** tool (optimization and inlining). The execution times in the tables below are always the mean measurement values.

The properties of the processor boards used are:

	DS1006
CPU clock	2.6 GHz / 3.0 GHz
Bus clock	133 MHz

Measured Execution Times

Execution times are available for the following RTLib units:

- Initialization
- ADC unit

Note

- The following execution times contain mean values for a sequence of I/O accesses. The execution time of a single call might be lower because of buffered I/O access.
- As of July 2009 the DS2003 is equipped with fast ADCs for which the execution times are also listed. This board variant is specially marked with an imprint (Fast ADC) on its bracket.

Initialization

The following execution time has been measured for the initialization function:

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2003_board_init	44.40 μ s	57.38 μ s

ADC Unit

The following execution times have been measured for the A/D converter functions:

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2003_set_range	0.02 μ s	0.02 μ s
ds2003_set_trigger	0.03 μ s	0.03 μ s
ds2003_set_wordlen	0.02 μ s	0.02 μ s
ds2003_init_scantbl	$1.298 + n^1 \cdot 0.072 \mu$ s	$1.565 + n^1 \cdot 0.03 \mu$ s
ds2003_start	0.01 μ s	0.01 μ s
ds2003_conversion_start	0.01 μ s	0.01 μ s
ds2003_buffer_clear	0.03 μ s	0.02 μ s
ds2003_in (after conversion finished)	$0.624 + n^1 \cdot 0.567 \mu$ s	$0.622 + n^1 \cdot 0.566 \mu$ s

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2003_start + ds2003_in	$4.346 + n^1 \cdot 2.193 \mu\text{s}$	$4.349 + n^1 \cdot 2.193 \mu\text{s}$
ds2003_read	$0.325 + n^1 \cdot 0.286 \mu\text{s}$	$0.319 + n^1 \cdot 0.286 \mu\text{s}$
ds2003_block_read	$1.310 + n^1 \cdot 0.584 \mu\text{s}$	$1.449 + n^1 \cdot 0.584 \mu\text{s}$
ds2003_fifo_half_full_block_read	$0.337 + n^1 \cdot 0.294 \mu\text{s}$	$0.336 + n^1 \cdot 0.294 \mu\text{s}$
ds2003_ready	$1.32 \mu\text{s}$	$1.43 \mu\text{s}$

¹⁾ n stands for the number of channels used.

B

base address 7
buffer not empty flag 25

C

Common Program Data folder 6

D

Documents folder 6
ds2003_block_read 30
ds2003_board_init 9
ds2003_buffer_clear 24
ds2003_conversion_start 22
ds2003_fifo_half_full_block_read 33
ds2003_in 25
ds2003_init_scanfbl 19
ds2003_read 27
ds2003_ready 36
ds2003_set_range 14
ds2003_set_trigger 16
ds2003_set_wordlen 17
ds2003_start 21
DSxxx_n_BASE 7

E

external trigger 16

F

FIFO half full interrupt 33

I

interrupt
 FIFO half full 33
 scan process completed 27

L

Local Program Data folder 6

S

scan process completed interrupt 27

