DS2004 High-Speed A/D Board

RTLib Reference

Release 2021-A – May 2021

dSPACE

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

Contents

About This Reference

Content

This RTLib Reference (Real-Time Library) gives detailed descriptions of the C functions needed to program a DS2004 High-Speed A/D 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
DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazard that, if not avoided, could result in property damage.
Note	Indicates important information that you should take into account to avoid malfunctions.
Tip	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.

General Information on the DS2004 Real-Time Library

Introduction	The DS2004 Real-Time Library provides the C functions which make the A/D conversion capabilities of the DS2004 High-Speed A/D Board available for RTI-specific Simulink S-functions or for handcoded control models.

Overview of the DS2004 Real-Time Library

Introduction	The DS2004 Real-Time Library provides functions to perform A/D conversions for various use cases.
A/D conversion channels	You can independently configure and use 16 input channels for A/D conversion.
A/D conversion modes	You can perform A/D conversion in different modes: ▪ Burst conversion modes for digitizing and buffering a sequence of analog values. The conversion results can be read when the specified number of A/D conversions is finished. ▪ Single conversion mode for digitizing and buffering one analog value. The conversion result can be read when the conversion has finished.
Burst size	You can define how many conversion results are to be buffered in a burst, with a burst size of up to 16384.

Trigger sources

You can choose between various trigger sources for the burst trigger and A/D conversion trigger:

- Software trigger
- 4 external trigger inputs with selectable edge polarity of the trigger signal
- Trigger by independent channel timer (in burst conversion mode as conversion trigger only)
- SYNCIN signal on the PHS bus

Interrupts

You can use 4 interrupts per channel to control A/D conversion via interrupt-driven tasks.

Voltage range

You can select the analog input voltage range (±5 V or ±10 V).

Handling of defective channels

During initialization of the DS2004 board, a self test is performed and messages are issued if errors were detected:

- If a single A/D conversion channel did not pass the test, the function issues a warning and the channel is marked as defective. In this case you can continue to use the DS2004 board, if you change your application to use one of the correct working channels. If your application tries to use a defective channel, an error message is issued and the application is terminated.
- If a primary hardware component affecting all channels did not pass the test, the function issues an error message and terminates the application.

Related topics

Basics

General Information on the Implementation Features of the DS2004 (DS2004 Features 🕮)

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 refer to 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 RTLib Reference 🕮 or 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 distinct 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 DS2004, you have to perform the initialization process.

Note

The processor board must be initialized before you can initialize the DS2004.

ds2004_init

Syntax

```
void ds2004_init(phs_addr_t base)
```

Include file

`ds2004.h`

Purpose

To initialize the DS2004 with default settings.

Description

This function causes a reset and a self test of the specified DS2004 board, disables all channels, and initializes them with default settings.

The following table shows the default settings:

Setting	Default Setting
Voltage range	–10 ... +10 V
Burst conversion mode	Triggered sample mode
Conversion trigger	Software
Burst trigger	Software

Setting	Default Setting
Timer period	1 s
Burst size	1

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
201	Error	ds2004_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.
240	Error	ds2004_init(0x??): Board not found!	No DS2004 board could be found at the specified PHS-bus address. Check if the DSxxxx_n_BASE macro corresponds to the I/O board used.
241	Error	ds2004_init(0x??): Memory allocation error!	The allocation of some dynamic memory for internal data storage has failed.
242	Error	ds2004_init(0x??): Board is not responding! Hardware reset failed.	The DS2004 board is not responding after a FPGA reset.
243	Warning/Error	ds2004_init(0x??): Power-on self test of DS2004 board failed!	The self test of the DS2004 board during initialization failed.
254	Error	ds2004_init(0x??): Access to I2C bus failed!	Access to the DS2004 internal I2C bus during the update of the ADC correction values failed.

Note

After the reset, messages are issued if the self test of the board detected errors:

- If a primary hardware component failed, the function issues an error message and terminates the application.
- If a single A/D conversion channel failed, the function issues a warning and the channel is marked as defective. If you try to enable a failed channel using the following functions an error message is issued and the application is terminated:
  - `ds2004_set_burst_mode`
  - `ds2004_single_init`
  - `ds2004_burst_init`

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to initialize the first DS2004 board connected to the PHS bus:

```
void main(void)
{
    init();
    ds2004_init(DS2004_1_BASE);
    ...
}
```

Related topics

References

ds2004_burst_init....36
ds2004_set_burst_mode....22
ds2004_single_init....47

A/D Conversion

Where to go from here

Information in this section

Converter Channel Initialization

Introduction

The DS2004 provides functions to initialize the A/D converter channels.

Where to go from here

Information in this section

ds2004_set_range

Syntax

```
void ds2004_set_range(
   phs_addr_t base,
   UInt32 channel,
   UInt32 range)
```

Include file

`ds2004.h`

Purpose

To set the input voltage range of an A/D conversion channel.

Description

This function sets the input voltage range of an A/D conversion channel.

Note

- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

range Specifies the input voltage range. The symbols are as follows:

Symbol	Meaning
`DS2004_RNG_5`	To set the input voltage range to –5 ... +5 V.
`DS2004_RNG_10`	To set the input voltage range to –10 ... +10 V.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2004_set_range(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the `ds2004_init` function.
245	Error	ds2004_set_range(0x??): Access to I2C bus failed!	Access to the DS2004 internal I2C bus during the update of the ADC correction values failed.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to set channel 1 of a DS2004 to the ±5V input voltage range:

```
ds2004_set_range(DS2004_1_BASE, 1, DS2004_RNG_5);
```

Related topics

References

ds2004_burst_init....36
ds2004_channel_control....28
ds2004_init....11
ds2004_single_init....47

ds2004_set_conversion_trigger

Syntax

```
void ds2004_set_conversion_trigger(
   phs_addr_t base,
   UInt32 channel,
   UInt32 trigger)
```

Include file

`ds2004.h`

Purpose

To set the trigger source for the conversion trigger.

Description

This function sets the trigger source which is used as the conversion trigger. Each conversion trigger effects the digitalization of one analog value and produces one conversion result.

Note

- A separate function is used to set the trigger source for the burst trigger which starts an A/D conversion burst. For setting the burst trigger, see `ds2004_set_burst_trigger`.
- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

trigger Specifies the trigger source for the conversion trigger. For the external trigger inputs, `trigger` additionally selects the evaluated edge polarity of the trigger signal. The symbols are as follows:

Symbol	Meaning
`DS2004_TRIG_SW`	To set the conversion trigger source to software via `ds2004_sw_trigger`.
`DS2004_TRIG_TIMER`	To set the conversion trigger source to channel timer.
`DS2004_TRIG_EXT1_RISING`	To set the conversion trigger source to external trigger input 1, rising edge of the signal.
`DS2004_TRIG_EXT2_RISING`	To set the conversion trigger source to external trigger input 2, rising edge of the signal.
`DS2004_TRIG_EXT3_RISING`	To set the conversion trigger source to external trigger input 3, rising edge of the signal.
`DS2004_TRIG_EXT4_RISING`	To set the conversion trigger source to external trigger input 4, rising edge of the signal.
`DS2004_TRIG_EXT1_FALLING`	To set the conversion trigger source to external trigger input 1, falling edge of the signal.
`DS2004_TRIG_EXT2_FALLING`	To set the conversion trigger source to external trigger input 2, falling edge of the signal.
`DS2004_TRIG_EXT3_FALLING`	To set the conversion trigger source to external trigger input 3, falling edge of the signal.
`DS2004_TRIG_EXT4_FALLING`	To set the conversion trigger source to external trigger input 4, falling edge of the signal.
`DS2004_TRIG_SYNCIN`	To set the conversion trigger source to SYNCIN signal via PHS bus.

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2004_set_conversion_trigger(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the `ds2004_init` function.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to assign the external trigger input 1 to channel 1 of a DS2004 board. The A/D conversion on channel 1 is triggered by the rising edge of a signal on input 1:

```
ds2004_set_conversion_trigger(DS2004_1_BASE, 1, DS2004_TRIG_EXT1_RISING);
```

Related topics

References

ds2004_burst_init....36
ds2004_channel_control....28
ds2004_init....11
ds2004_set_burst_trigger....26
ds2004_single_init....47
ds2004_sw_trigger....30

ds2004_set_timer_period

Syntax

```
void ds2004_set_timer_period(
   phs_addr_t base,
   UInt32 channel,
   dsfloat period)
```

Include file

`ds2004.h`

Purpose

To set the period value of a channel timer.

Description

This function sets the period value. The channel timer with a resolution of 10 ns can be used to trigger an A/D conversion or to trigger an A/D conversion burst.

Note

- A timer is started by enabling the corresponding A/D channel using the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

period Specifies the period value in the range 0.8 µs ... 1.342 s.

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2004_set_timer_period (0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the `ds2004_init` function.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to set channel 1 of a DS2004 to the 1 ms period value:

```
ds2004_set_timer_period(DS2004_1_BASE, 1, 1.0e-3);
```

Related topics

References

ds2004_set_burst_mode

Syntax

```
void ds2004_set_burst_mode(
   phs_addr_t base,
   UInt32 channel,
   UInt32 mode)
```

Include file

`ds2004.h`

Purpose

To set the burst conversion mode.

Description

This function initializes the burst conversion mode, in which a sequence of analog values is digitized and buffered. The A/D conversion results of the burst can be read when the specified number of A/D conversions is finished.

You can choose between two burst conversion modes:

- Triggered sample mode

  One trigger signal from the selected burst trigger source effects the execution of one burst of A/D conversions. The next burst must be started with a new burst trigger.
- Continuous sample mode

  One trigger signal from the selected burst trigger source effects the continuous execution of A/D conversion bursts. After a burst has finished, the next burst is started automatically. To stop the continuous bursts, the channel must be disabled by the `ds2004_channel_control` function.

Note

- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

mode Specifies the burst conversion mode. The symbols are as follows:

Symbol	Meaning
`DS2004_TRIGGERED_MODE`	To set triggered burst conversion mode.
`DS2004_CONTINUOUS_MODE`	To set continuous burst conversion mode.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2004_set_burst_mode(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the `ds2004_init` function.
244	Error	ds2004_set_burst_mode(0x??): Power-on self test of channel ?? failed!	The self test of the specified channel during initialization failed. This channel must not be used. The other channels of the board can be used.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to set channel 1 of a DS2004 to the continuous sample mode:

```
ds2004_set_burst_mode(DS2004_1_BASE, 1, DS2004_CONTINUOUS_MODE);
```

Related topics

References

ds2004_burst_init....36
ds2004_channel_control....28
ds2004_init....11
ds2004_single_init....47

ds2004_set_burst_size

Syntax

```
void ds2004_set_burst_size(
   phs_addr_t base,
   UInt32 channel,
   UInt32 count)
```

Include file

`ds2004.h`

Purpose

To set the number of A/D conversions per burst.

Description

This function specifies the number of A/D conversions per burst. It also specifies the size of the write, free, and read buffer in the swinging buffer. For details on the swinging buffer, refer to Swinging Buffer (DS2004 Features 🕮).

Note

- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

count Specifies the number of conversions per burst in the range 1 ... 16348. This value also specifies the buffer size of each of the write, free, and read buffers in the swinging buffer.

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2004_set_burst_size(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the `ds2004_init` function.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to set channel 1 of a DS2004 to 150 conversions per burst:

```
ds2004_set_burst_size(DS2004_1_BASE, 1, 150);
```

Related topics

Basics

Swinging Buffer (DS2004 Features 🕮)

References

ds2004_burst_init....36
ds2004_channel_control....28
ds2004_init....11
ds2004_single_init....47

ds2004_set_burst_trigger

Syntax

```
void ds2004_set_burst_trigger(
   phs_addr_t base,
   UInt32 channel,
   UInt32 trigger)
```

Include file

`ds2004.h`

Purpose

To set the trigger source for triggering an A/D conversion burst.

Description

This function selects the trigger source for starting a burst of A/D conversions. The burst conversion runs in triggered mode or in continuous mode, depending on the mode which is set by the `ds2004_set_burst_mode` function.

Note

- A separate function is used to set the trigger source for the conversion trigger which starts an A/D conversion. For setting the conversion trigger, see `ds2004_set_conversion_trigger`.
- The `ds2004_init` function must be called before this function can be used.
- This function may be used only before the A/D channel is enabled by one of the following functions:
  - `ds2004_channel_control`
  - `ds2004_single_init`
  - `ds2004_burst_init`
  - and after the channel is disabled by the `ds2004_channel_control` function.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

trigger Specifies the trigger source for starting an A/D conversion burst. For the external trigger inputs, `trigger` additionally selects the evaluated edge polarity of the trigger signal. The symbols are as follows:

Symbol	Meaning
DS2004_TRIG_SW	To set the burst trigger source to software via **ds2004_sw_trigger**.
DS2004_TRIG_TIMER	To set the burst trigger source to channel timer.
DS2004_TRIG_EXT1_RISING	To set the burst trigger source to external trigger input 1, rising edge of the signal.
DS2004_TRIG_EXT2_RISING	To set the burst trigger source to external trigger input 2, rising edge of the signal.
DS2004_TRIG_EXT3_RISING	To set the burst trigger source to external trigger input 3, rising edge of the signal.
DS2004_TRIG_EXT4_RISING	To set the burst trigger source to external trigger input 4, rising edge of the signal.
DS2004_TRIG_EXT1_FALLING	To set the burst trigger source to external trigger input 1, falling edge of the signal.
DS2004_TRIG_EXT2_FALLING	To set the burst trigger source to external trigger input 2, falling edge of the signal.
DS2004_TRIG_EXT3_FALLING	To set the burst trigger source to external trigger input 3, falling edge of the signal.
DS2004_TRIG_EXT4_FALLING	To set the burst trigger source to external trigger input 4, falling edge of the signal.
DS2004_TRIG_SYNCIN	To set the burst trigger source to SYNCIN signal via PHS bus.

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2004_set_burst_trigger(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the **ds2004_init** function.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to assign external trigger input 1 to channel 1 of a DS2004 board. The A/D conversion bursts on channel 1 are triggered by the rising edge of a signal on external trigger input 1:

```
ds2004_set_burst_trigger(DS2004_1_BASE, 1, DS2004_TRIG_EXT1_RISING);
```

Related topics

References

General Channel Handling

Introduction

The DS2004 provides functions for general channel handling, like resetting a channel or starting an A/D conversion using a software trigger.

Where to go from here

Information in this section

ds2004_channel_control

Syntax

```
__INLINE void ds2004_channel_control(
   phs_addr_t base,
   UInt32 ch_mask,
   UInt32 cmd)
```

Include file

`ds2004.h`

Purpose

To control the DS2004 A/D channels.

Description

With this function you can control the specified A/D channels:

- Enable A/D channels

  The specified channels are enabled. The channel timers are reset and started, all conversion result buffers are cleared. As all channels are disabled after DS2004 initialization, the channels which are to be used must be enabled.

  The `ds2004_set_...` functions must not be used to change the channel configuration while the channel is enabled.

- Disable A/D channels

  The specified channels are disabled. The interrupt flags are cleared. Unread finished buffers can be read using `ds2004_burst_read`, unfinished buffers can be read using `ds2004_burst_in_immediate`
- Reset A/D channels

  The specified channels are disabled and then reenabled. The channel timers are reset and started, and all related conversion result buffers are cleared.
- Abort A/D channels

  The conversion burst currently in progress on the specified channels is aborted. A data ready interrupt is generated, if enabled. The currently available conversion results can be read using the `ds2004_burst_read` function. The number of currently available conversion results can be evaluated using the `ds2004_read_buffer_count` function.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

ch_mask Specifies the channels to be controlled. The symbols are as follows for the 16 channels and can be combined by logical OR operation:

Symbol	Meaning
`DS2004_CTRL_CH1`	To control channel 1.
`DS2004_CTRL_CH2`	To control channel 2.
...	...
`DS2004_CTRL_CH15`	To control channel 15.
`DS2004_CTRL_CH16`	To control channel 16.

cmd Specifies the channel operation. The symbols are as follows:

Symbol	Meaning
`DS2004_CH_ENABLE`	To enable the specified channels.
`DS2004_CH_DISABLE`	To disable the specified channels.
`DS2004_CH_RESET`	To reset the specified channels.
`DS2004_CH_ABORT`	To abort burst A/D conversions on the specified channels.

Return value	None
Execution times	For information on the execution times, refer to Function Execution Times on page 85.

Example	This example shows how to enable channels 1 and 3. The related channel timers are reset and started, and all related conversion result buffers are cleared:
	`ds2004_channel_control(DS2004_1_BASE,`
	`  DS2004_CTRL_CH1 | DS2004_CTRL_CH3,`
	`  DS2004_CH_ENABLE);`

Related topics	References ds2004_burst_in_immediate....45 ds2004_burst_read....42 ds2004_init....11 ds2004_read_buffer_count....32

ds2004_sw_trigger

Syntax	`__INLINE void ds2004_sw_trigger(`
	`  phs_addr_t base,`
	`  UInt32 mask)`

Include file	`ds2004.h`
Purpose	To trigger an A/D conversion burst or to trigger an A/D conversion.
Description	This function can be used: ▪ As conversion trigger. In this case the function can trigger a new A/D conversion on multiple channels in burst or single conversion mode. ▪ As burst trigger. In this case the function can trigger a new A/D conversion burst on multiple channels in triggered sample mode. In continuous sample mode, the function can act as initial trigger to start successive A/D conversion bursts.

Note

- The `ds2004_init` function must be called before this function can be used.
- This function must not be used to trigger a burst and a conversion on the same channel at the same time. A settling time of 50 ns is required between a burst trigger and the first conversion trigger in the related burst. Distributing the burst trigger and the first conversion trigger to separate function calls fulfills this precondition.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

mask Specifies the channels to be triggered and selects if a conversion or a conversion burst is to be triggered on the respective channel. The symbols are as follows and can be combined by a logical OR operation:

Symbol	Meaning
`DS2004_CONV_START_1`	To trigger an A/D conversion on channel 1.
…	…
`DS2004_CONV_START_16`	To trigger an A/D conversion on channel 16.
`DS2004_BURST_START_1`	To trigger a burst on channel 1.
…	…
`DS2004_BURST_START_16`	To trigger a burst on channel 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to trigger an A/D conversion on channel 1 and a burst on channel 5:

```
ds2004_sw_trigger(DS2004_1_BASE, DS2004_CONV_START_1 | DS2004_BURST_START_5);
```

Related topics

References

ds2004_init..... 11

ds2004_read_buffer_count

Syntax

```
__INLINE void ds2004_read_buffer_count(
   phs_addr_t base,
   UInt32 channel,
   UInt32 buffer,
   UInt32 *count)
```

Include file

`ds2004.h`

Purpose

To query the number of conversion results in the read or write buffer.

Description

This function queries the number of currently available A/D conversion results in the read buffer or in the write buffer. The number can be used to read the correct number of valid conversion results after aborting a burst using the **`ds2004_burst_in_immediate`** function. The function can also be used for debugging, for example, to query the number of conversion results in the write buffer if a conversion burst is not finished.

Note

The **`ds2004_init`** function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

buffer Specifies the buffer to be queried. The symbols are as follows:

Symbol	Meaning
`DS2004_READ_BUFFER`	To get the number of conversion results in the read buffer.
`DS2004_WRITE_BUFFER`	To get the number of conversion results in the write buffer.

count Specifies the pointer to the variable containing the number of read conversion results after the function call.

Return value	None
Execution times	For information on the execution times, refer to Function Execution Times on page 85.
Example	This example shows how to read the number of available A/D conversion results in the read buffer on channel 1: `ds2004_read_buffer_count(DS2004_1_BASE, 1, DS2004_READ_BUFFER, &count);`
Related topics	References ds2004_burst_in_immediate....45 ds2004_init....11

ds2004_data_ready

Syntax	`__INLINE UInt32 ds2004_data_ready(`
	`  phs_addr_t base,`
	`  UInt32 channel)`

Include file	`ds2004.h`
Purpose	To query if a new buffer is available.
Description	This function can be used to poll until a new read buffer with unread conversion results is available. Therefore the **new_flag** of the channel is evaluated. New results will be available after: ▪ A burst of conversions is finished in burst conversion mode. ▪ An A/D conversion is finished in single conversion mode. If new data is available you can use the **ds2004_single_read** or **ds2004_burst_read** function to read the new data. The parameter **new_flag** of these functions must be ignored, because the **new_flag** is already cleared by using this function.

Note

- The `ds2004_init` function must be called before this function can be used.
- When this function is used, the new data ready flag of the channel is cleared after new data are available. Therefore the `ds2004_burst_in` and `ds2004_burst_in` functions must not be called after this function has been used because they are polling the ready flag.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 … 16.

Return value

This function returns a status value. The symbols are as follows:

Symbol	Meaning
`DS2004_OLD_BUFFER`	A new buffer is not available.
`DS2004_NEW_BUFFER`	A new buffer is available.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to poll on channel 1 in single conversion mode until a new buffer with a new A/D conversion result is available. The result is read using the `ds2004_single_read` function which is not polling:

```
UInt32  dummy;      // ignored new_flag
dsfloat adc_value;  // conversion value
// wait until new conversion value is available
while(ds2004_data_ready(DS2004_1_BASE, 1) == DS2004_OLD_BUFFER)
{
   my_time_out_check();
}
// read conversion value
ds2004_single_read(DS2004_1_BASE, 1, &dummy, &adc_value);
```

Related topics

References

ds2004_burst_in....40
ds2004_burst_read....42
ds2004_init....11
ds2004_single_read....52

A/D Burst Conversion

Introduction

The functions for burst A/D conversion are used to initialize and apply the burst capabilities of the DS2004.

Where to go from here

Information in this section

ds2004_burst_init

Syntax

```
void ds2004_burst_init(
   phs_addr_t base,
   UInt32 channel,
   UInt32 range,
   UInt32 burst_mode,
   UInt32 burst_size,
   UInt32 burst_trigger,
   UInt32 conv_trigger,
   dsfloat period)
```

Include file

`ds2004.h`

Purpose

To initialize the burst A/D conversion mode.

Description

This function initializes the burst conversion mode of a DS2004 channel.

This function disables the A/D channel, and sets the voltage range, the burst conversion mode, the burst trigger source, the conversion trigger source, and, if required, the channel timer period. Then the channel is enabled.

Therefore, this function implies the following functions:

- `ds2004_set_range`
- `ds2004_set_conversion_trigger`
- `ds2004_set_burst_trigger`
- `ds2004_set_timer_period`
- `ds2004_set_burst_size`
- `ds2004_channel_control`

The A/D conversions must be triggered by the selected conversion trigger source. The conversion results are available after the required number of conversions per burst have finished.

For the burst conversion, two sample modes are available:

- Triggered sample mode
- Continuous sample mode

Triggered sample mode After receiving the specified burst trigger, a burst of conversions is started once. The next burst must be started with a new burst trigger.

Continuous sample mode After receiving an initial burst trigger, bursts of conversions are executed continuously. After a burst has finished, the next burst is started automatically.

There are two ways to stop the successive bursts:

- The burst mode of the channel can be reinitialized to triggered sample mode using the `ds2004_set_burst_mode` function. The burst will stop after the current burst in progress has finished. To restart the continuous burst, the channel must be set back to continuous mode using `ds2004_burst_init` and a new initial burst trigger must be performed.
- The channel can be disabled using the `ds2004_channel_control` function. The burst is aborted immediately. The `ds2004_burst_in_immediate` function can be used to read the conversion results of the incomplete burst. To restart the continuous burst, the channel must be reenabled using the `ds2004_channel_control` function and a new initial burst trigger must be performed.

Note

- The `ds2004_init` function must be called before this function can be used.
- A burst trigger and the first A/D conversion trigger within this burst must not be issued at the same time on the same channel. A settling time of 50 ns is required between them. Distributing the burst trigger and the first conversion trigger to separate function calls fulfills this precondition.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

range Specifies the input voltage range. The symbols are as follows:

Symbol	Meaning
`DS2004_RNG_5`	To set the input voltage range to –5 V … +5 V.
`DS2004_RNG_10`	To set the input voltage range to –10 V … +10 V.

burst_mode Specifies the sample mode. The symbols are as follows:

Symbol	Meaning
`DS2004_TRIGGERED_MODE`	To set triggered sample mode.
`DS2004_CONTINUOUS_MODE`	To set continuous sample mode.

burst_size Specifies the number of conversions per burst in the range 1 ... 16348.

burst_trigger Specifies the trigger source for starting the bursts. For the external trigger inputs, you additionally can choose the evaluated edge polarity of the trigger signal.

This parameter is evaluated only if triggered sample mode is selected.

The symbols are as follows:

Symbol	Meaning
`DS2004_TRIG_SW`	To set the burst trigger source to software via `ds2004_sw_trigger`.
`DS2004_TRIG_EXT1_RISING`	To set the burst trigger source to external trigger input 1, rising edge of the signal.
`DS2004_TRIG_EXT2_RISING`	To set the burst trigger source to external trigger input 2, rising edge of the signal.
`DS2004_TRIG_EXT3_RISING`	To set the burst trigger source to external trigger input 3, rising edge of the signal.
`DS2004_TRIG_EXT4_RISING`	To set the burst trigger source to external trigger input 4, rising edge of the signal.
`DS2004_TRIG_EXT1_FALLING`	To set the burst trigger source to external trigger input 1, falling edge of the signal.
`DS2004_TRIG_EXT2_FALLING`	To set the burst trigger source to external trigger input 2, falling edge of the signal.
`DS2004_TRIG_EXT3_FALLING`	To set the burst trigger source to external trigger input 3, falling edge of the signal.
`DS2004_TRIG_EXT4_FALLING`	To set the burst trigger source to external trigger input 4, falling edge of the signal.
`DS2004_TRIG_SYNCIN`	To set the burst trigger source to SYNCIN signal via PHS bus.

conv_trigger Specifies the trigger source for starting the A/D conversions. For the external trigger inputs, you additionally can choose the evaluated edge polarity of the trigger signal. The symbols are as follows:

Symbol	Meaning
DS2004_TRIG_SW	To set the conversion trigger source to software via **ds2004_sw_trigger**.
DS2004_TRIG_TIMER	To set the conversion trigger source to channel timer.
DS2004_TRIG_EXT1_RISING	To set the conversion trigger source to external trigger input 1, rising edge of the signal.
DS2004_TRIG_EXT2_RISING	To set the conversion trigger source to external trigger input 2, rising edge of the signal.
DS2004_TRIG_EXT3_RISING	To set the conversion trigger source to external trigger input 3, rising edge of the signal.
DS2004_TRIG_EXT4_RISING	To set the conversion trigger source to external trigger input 4, rising edge of the signal.
DS2004_TRIG_EXT1_FALLING	To set the conversion trigger source to external trigger input 1, falling edge of the signal.
DS2004_TRIG_EXT2_FALLING	To set the conversion trigger source to external trigger input 2, falling edge of the signal.
DS2004_TRIG_EXT3_FALLING	To set the conversion trigger source to external trigger input 3, falling edge of the signal.
DS2004_TRIG_EXT4_FALLING	To set the conversion trigger source to external trigger input 4, falling edge of the signal.
DS2004_TRIG_SYNCIN	To set the conversion trigger source to SYNCIN signal via PHS bus.

period Specifies the period value of the channel timer in the range 0.8 µs … 1.342 s. Values that do not comply to the resolution of 10 ns are rounded. This parameter is evaluated only if the conversion trigger source is set to **DS2004_TRIG_TIMER**.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2004_burst_init(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the **ds2004_sw_trigger** function.
244	Error	ds2004_burst_init(0x??): Power-on self test of channel ?? failed!	The self test of the specified channel during initialization failed. This channel must not be used.
245	Error	ds2004_burst_init(0x??): Access to I2C bus failed!	Access to the DS2004 internal I2C bus during the update of the ADC correction values failed.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to initialize burst conversion mode on channel 1 of the first DS2004 in a PHS-bus-based system. The input voltage range is set to –10 ... +10 V, the burst conversion mode is set to triggered sample mode, the burst size is set to 50, the burst trigger source is set to software, the conversion trigger source is set to timer with a period of 0.5 ms:

```
ds2004_burst_init(DS2004_1_BASE, // PHS-bus base address
   1,                    // use channel 1
   DS2004_RNG_10,        // -10 ... +10V input voltage range
   DS2004_TRIGGERED_MODE,  // select triggered sample mode
   50,                   // 50 conversions per burst
   DS2004_TRIG_SW,       // burst is triggered by software
   DS2004_TRIG_TIMER,    // conversion is triggered by timer
   500.0e-6);            // conversion every 500µs
```

Related topics

References

ds2004_burst_in

Syntax

```
__INLINE void ds2004_burst_in(
   phs_addr_t base,
   UInt32 channel,
   UInt32 offset,
   UInt32 length,
   dsfloat *data)
```

Include file

`ds2004.h`

Purpose

To read the new A/D conversion results of a burst with polling for the availability of new conversion results.

Description

This function waits until a burst of A/D conversions has completed and the buffer containing the conversion results is available. The buffer is read and the conversion results are scaled to floating-point values in the range –1.0 ... +1.0.

Using the `offset` and `length` parameters enables the read-out of sections of the read buffer.

Note

- The `ds2004_init` function must be called before this function can be used.
- The burst conversion mode must have been initialized by using the `ds2004_burst_init` function. If in triggered sample mode, a burst must have been started by the selected burst trigger source and the A/D conversions must have been started by the selected A/D conversion trigger source.
- The following conditions must be met to avoid that the application remains in a polling loop:
  - It is important that the required number of A/D conversions is triggered by the selected source.
  - The data ready flag must not be deleted using one of the functions `ds2004_single_in`, `ds2004_single_read`, `ds2004_burst_read`, `ds2004_burst_in_immediate`, or `ds2004_data_ready` on the same channel.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 … 16.

offset Specifies the offset address in the buffer in the range 0 … 16383. Reading starts from this address in the read buffer. The sum of length and offset must not exceed 16384. If the entire buffer is to be read, `offset` has to be set to zero.

length Specifies the number of conversion results to be read in the range 1 … 16384. The following symbol is predefined:

Symbol	Meaning
`DS2004_BUFFER_SIZE_LENGTH`	To set the number of conversion results to be read to `burst_size` used in the `ds2004_burst_init` function.

If only a segment of the conversion results in the buffer is to be read, offset and length have to be set to appropriate values. The destination data array size must match the initialized burst size or the length of the segment to be read. The sum of length and offset must not exceed 16384.

data Specifies the pointer to the array containing the read A/D conversion results after the function call. Each value is scaled to a floating-point value in the range –1.0 V … +1.0 V. The earliest value is stored in `data[0]`.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Examples

This example shows how to read the A/D conversion results of a complete burst on channel 1:

```
dsfloat data[20];
...
ds2004_burst_in(DS2004_1_BASE, 1, 0, DS2004_BUFFER_SIZE_LENGTH, &data);
```

This example shows how to read the A/D conversion results of a segment of a burst on channel 1:

```
dsfloat data[20];
UInt32 offset = 5;
UInt32 length = 20;
...
ds2004_burst_in(DS2004_1_BASE, 1, offset, length, &data);
```

Related topics

References

ds2004_burst_in_immediate....45
ds2004_burst_init....36
ds2004_burst_read....42
ds2004_data_ready....33
ds2004_init....11
ds2004_single_in....50
ds2004_single_read....52

ds2004_burst_read

Syntax

```
__INLINE void ds2004_burst_read(
   phs_addr_t base,
   UInt32 channel,
   UInt32 offset,
   UInt32 length,
   UInt32 *new_flag,
   dsfloat *data)
```

Include file

`ds2004.h`

Purpose

To read conversion results of a burst without polling for the availability of new conversion results.

Description

This function does not wait until an A/D conversion burst has completed and the new conversion results are available in the read buffer. The current read buffer is read and the conversion results are scaled to floating-point values in the range –1.0 … +1.0. Using the `offset` and `length` parameters enables the read-out of sections of the read buffer.

If a new buffer is not available, the old one of the preceding A/D conversion burst is read again. The parameter `new_flag` indicates if new conversion results were available.

This function can also be used in a data ready interrupt service routine. In burst conversion mode, this interrupt is generated when the burst has finished and the buffer with the new conversion results is available.

Note

- The `ds2004_init` function must be called before this function can be used.
- The burst conversion mode must have been initialized by using the `ds2004_burst_init` function. If in triggered sample mode, a burst must have been started by the selected burst trigger source and the A/D conversions must have been started by the selected A/D conversion trigger source.
- This function must not be used in conjunction with one of the following functions on the same channel. In other case this may hang up your application.
  - `ds2004_single_in`
  - `ds2004_burst_in`
  - `ds2004_burst_in_immediate`

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 … 16.

offset Specifies the offset address in the buffer in the range 0 … 16383. Reading starts from this address in the read buffer. The sum of length and offset must not exceed 16384. If the entire buffer is to be read, `offset` has to be set to zero.

length Specifies the number of conversion results to be read. The following symbol is predefined:

Symbol	Meaning
DS2004_BUFFER_SIZE_LENGTH	To set the number of conversion results to be read to **burst_size** used in the **ds2004_burst_init** function.

If only a segment of the conversion results in the buffer is to be read, offset and length have to be set to appropriate values. The destination data array size must match the initialized burst size or the length of the segment to be read. The sum of length and offset must not exceed 16384.

new_flag Specifies the pointer to the variable where the flag value is returned. The symbols are as follows:

Symbol	Meaning
DS2004_OLD_BUFFER	An old buffer was read.
DS2004_NEW_BUFFER	A new buffer was read.

data Specifies the pointer to the array containing the read conversion results after the function call. The earliest value is stored in **data[0]**.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to read the A/D conversion results of a burst immediately on channel 1 without polling for the availability of new conversion results:

```
ds2004_burst_read(DS2004_1_BASE,         // PHS-bus base address
   1,                                    // use channel 1
   0,                                    // without offset
   DS2004_BUFFER_SIZE_LENGTH,   // full buffer size as initialized)
   &new_flag,            // returned flag (new/old buffer)
   &data);               // array with returned values
```

Related topics

References

ds2004_burst_in....40
ds2004_burst_in_immediate....45
ds2004_burst_init....36
ds2004_init....11
ds2004_single_in....50

ds2004_burst_in_immediate

Syntax

```
__INLINE void ds2004_burst_in_immediate(
   phs_addr_t base,
   UInt32 channel,
   UInt32 *len,
   dsfloat *data)
```

Include file

`ds2004.h`

Purpose

To abort the current burst and read the available A/D conversion results.

Description

The burst in progress is aborted. The conversion results in the write buffer which is incompletely filled are immediately available in the read buffer. The conversion results are read and scaled to floating-point values in the range –1.0 ... +1.0 V. The destination data array size must match the initialized **`burst_size`**. The number of conversion results is returned by the **`len`** parameter.

In continuous sample mode, the next burst is started automatically after the current burst has been aborted by this function.

Note

- The **`ds2004_init`** function must be called before this function can be used.
- This function must not be used in conjunction with one of the functions because this may hang up your application.
  - **`ds2004_single_in`**
  - **`ds2004_single_read`**
  - **`ds2004_burst_in`**
  - **`ds2004_burst_read`**
  - **`ds2004_data_ready`**

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

len Specifies the pointer to the variable containing the number of read conversion results after the function call.

data Specifies the pointer to the array containing the read A/D conversion results after the function call. Each value is scaled to a floating-point value in the range –1.0 ... +1.0 V. The earliest value is stored in `data[0]`.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to read a burst of A/D conversion results immediately on channel 1:

```
ds2004_burst_in_immediate(DS2004_1_BASE,         // PHS-bus base address
   1,                    // use channel 1
   &len,                 // returned number of values
   &data);               // array with returned values
```

Related topics

References

ds2004_burst_in....40
ds2004_burst_read....42
ds2004_data_ready....33
ds2004_init....11
ds2004_single_in....50
ds2004_single_read....52

A/D Single Conversion

Introduction

The DS2004 can be used like a standard A/D conversion board without utilizing its burst capabilities. The standard conversion method is called single conversion mode. The single conversion mode is a special case of the burst conversion mode. For ease of use the DS2004 RTLib offers special functions to make the single conversion mode available.

Where to go from here

Information in this section

ds2004_single_init

Syntax

```
void ds2004_single_init(
   phs_addr_t base,
   UInt32 channel,
   UInt32 range,
   UInt32 trigger,
   dsfloat period)
```

Include file

`ds2004.h`

Purpose

To initialize single A/D conversion mode.

Description

This function initializes the single conversion mode of a DS2004 channel. In this mode, the converter of the channel acts as a standard A/D converter without utilizing the burst capabilities of the DS2004.

The single conversion mode is a special case of the burst conversion mode with the following burst settings:

- Burst size of 1
- Sample mode set to continuous
- Automatically generated initial burst trigger

This function disables the A/D channel and sets the voltage range, the conversion trigger source, and, if required, the channel timer period. Then the channel is enabled. Therefore, this function implies the following functions using the predefined settings for single conversion mode:

- `ds2004_set_range`
- `ds2004_set_conversion_trigger`
- `ds2004_set_timer_period`
- `ds2004_channel_control`

A conversion must be started by the selected conversion trigger source. For example, a software triggered conversion can be started by calling the `ds2004_sw_trigger` function.

The conversion result is available immediately after the conversion has finished. The conversion result can be read by using the `ds2004_single_in` or the ds2004_single_read function.

Note

The ds2004_init function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 … 16.

range Specifies the input voltage range. The symbols are as follows:

Symbol	Meaning
`DS2004_RNG_5`	To set the input voltage range to –5 V … +5 V.
`DS2004_RNG_10`	To set the input voltage range to –10 V … +10 V.

trigger Specifies the trigger source for starting the A/D conversion. For the external trigger inputs, you additionally can choose the evaluated edge polarity of the trigger signal.

The symbols are as follows:

Symbol	Meaning
DS2004_TRIG_SW	To set the conversion trigger source to software via ds2004_sw_trigger.
DS2004_TRIG_TIMER	To set the conversion trigger source to channel timer.
DS2004_TRIG_EXT1_RISING	To set the conversion trigger source to external trigger input 1, rising edge of the signal.
DS2004_TRIG_EXT2_RISING	To set the conversion trigger source to external trigger input 2, rising edge of the signal.
DS2004_TRIG_EXT3_RISING	To set the conversion trigger source to external trigger input 3, rising edge of the signal.
DS2004_TRIG_EXT4_RISING	To set the conversion trigger source to external trigger input 4, rising edge of the signal.
DS2004_TRIG_EXT1_FALLING	To set the conversion trigger source to external trigger input 1, falling edge of the signal.
DS2004_TRIG_EXT2_FALLING	To set the conversion trigger source to external trigger input 2, falling edge of the signal.
DS2004_TRIG_EXT3_FALLING	To set the conversion trigger source to external trigger input 3, falling edge of the signal.
DS2004_TRIG_EXT4_FALLING	To set the conversion trigger source to external trigger input 4, falling edge of the signal.
DS2004_TRIG_SYNCIN	To set the conversion trigger source to SYNCIN signal via PHS bus.

period Specifies the period value in the range 0.8 µs … 1.342 sec. Values that do not comply to the resolution of 10 ns are rounded. This parameter is evaluated only if the conversion trigger source is set to DS2004_TRIG_TIMER.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2004_single_init(0x??): Board not initialized!	The DS2004 board has not been initialized by a preceding call to the ds2004_init function.
244	Error	ds2004_single_init(0x??): Power-on self test of channel ?? failed!	The self test of the specified channel during initialization failed. This channel must not be used.
245	Error	ds2004_single_init(0x??): Access to I2C bus failed!	Access to the DS2004 internal I2C bus during the update of the ADC correction values failed.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to initialize single mode on channel 1 of the first DS2004 in a PHS-bus-based system. The input voltage range is set to –10 V … +10 V, and the conversion trigger source is set to software:

```
ds2004_single_init(DS2004_1_BASE, // PHS-bus base address
   1,                 // use channel 1
   DS2004_RNG_10,     // -10 ... +10 V input voltage range
   DS2004_TRIG_SW,    // conversion is triggered by software
   0);                // timer period does not matter
```

Related topics

References

ds2004_channel_control....28
ds2004_init....11
ds2004_set_conversion_trigger....18
ds2004_set_range....16
ds2004_set_timer_period....20
ds2004_single_in....50
ds2004_single_read....52
ds2004_sw_trigger....30

ds2004_single_in

Syntax

```
__INLINE void ds2004_single_in(
   phs_addr_t base,
   UInt32 channel,
   dsfloat *data)
```

Include file

`ds2004.h`

Purpose

To read a new A/D conversion result with polling for conversion completion.

Description

This function polls until an A/D conversion has completed and the new conversion result is available. Then the function reads the conversion result and it is scaled to a floating-point value in the range –1.0 ... +1.0 V.

Note

- The `ds2004_init` function must be called before this function can be used.
- The single conversion mode must have been initialized by using the `ds2004_single_init` function. A conversion must have been started by the selected trigger source.
- The following conditions must be met to avoid that the application remains in a polling loop:
  - It is important that the A/D conversion is triggered by the selected source.
  - The data ready flag must not be deleted using one of the functions `ds2004_single_in`, `ds2004_single_read`, `ds2004_burst_read`, `ds2004_burst_in_immediate`, or `ds2004_data_ready` on the same channel.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

data Specifies the pointer to the variable in which the conversion result is returned.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to read the next available A/D conversion result on channel 1:

```
ds2004_single_in(DS2004_1_BASE, 1, &data);
```

Related topics

References

ds2004_single_read

Syntax

```
__INLINE void ds2004_single_read(
   phs_addr_t base,
   UInt32 channel,
   UInt32 *new_flag,
   dsfloat *data)
```

Include file

`ds2004.h`

Purpose

To read a new A/D conversion result without polling for conversion completion.

Description

This function does not wait until an A/D conversion is finished and a new conversion result is available. The current conversion result is read immediately and it is scaled to a floating-point value in the range –1.0 ... +1.0 V.

If a new conversion result is not available, the old one of the preceding conversion is read again. The return parameter **`new_flag`** indicates, if a new conversion result was available.

This function can also be used in a data ready interrupt service routine. In single conversion mode, this interrupt is generated when the conversion has finished and the new conversion result is available. For further information on the data ready interrupt, refer to Data Ready Interrupt on page 63.

Note

- The `ds2004_init` function must be called before this function can be used.
- The single conversion mode must have been initialized by using the `ds2004_single_init` function. A conversion must have been started by the selected trigger source.
- This function must not be used in conjunction with one of the functions `ds2004_single_in`, `ds2004_burst_in`, and `ds2004_burst_in_immediate` on the same channel. This may hang up your application.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 … 16.

new_flag Specifies the pointer to the variable in which the flag value is returned. The symbols are as follows:

Symbol	Meaning
`DS2004_OLD_BUFFER`	An old conversion result was read.
`DS2004_NEW_BUFFER`	A new conversion result was read.

data Specifies the pointer to variable in which the conversion result is returned.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to read an A/D conversion result immediately on channel 1:

```
ds2004_single_read(DS2004_1_BASE, 1, &new_flag, &data);
```

Related topics

References

Interrupts

Introduction

Interrupts give information on specific states of the A/D conversion and can be evaluated by your applications.

Where to go from here

Information in this section

Burst Start Interrupt

Introduction

The DS2004 provides functions to react to a A/D conversion burst start. For further information, refer to Burst Start Interrupt (DS2004 Features 🕮).

Where to go from here

Information in this section

ds2004_burst_start_int_enable

Syntax

```
void ds2004_burst_start_int_enable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To enable a burst start interrupt.

Description

This function enables the burst start interrupt on the specified channel. Enabling the interrupt is a precondition for using the interrupt in your application.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to enable the burst start interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_burst_start_int_enable(DS2004_1_BASE, 1);
```

Related topics

Basics

Burst Start Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_burst_start_int_disable

Syntax

```
void ds2004_burst_start_int_disable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To disable a burst start interrupt.

Description

This function disables the burst start interrupt on the specified channel. Every interrupt that is no longer used should be disabled.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to disable the burst start interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_burst_start_int_disable(DS2004_1_BASE, 1);
```

Related topics

Basics

Burst Start Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_burst_start_int_ack

Syntax

```
__INLINE void ds2004_burst_start_int_ack(
   phs_addr_t base,
   UInt32 channel)
```

Include file	`ds2004.h`
Purpose	To acknowledge a burst start interrupt.
Description	This function acknowledges the burst start interrupt on the specified channel. The acknowledgement resets the associated flag in the interrupt flag register. An interrupt, that has been evaluated, must always be acknowledged. **Note** The `ds2004_init` function must be called before this function can be used.
I/O mapping	For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).
Parameters	**base** Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9). **channel** Specifies the channel number in the range 1 ... 16.
Return value	None
Execution times	For information on the execution times, refer to Function Execution Times on page 85.
Example	This example shows how to acknowledge the burst start interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system: `ds2004_burst_start_int_ack(DS2004_1_BASE, 1);`
Related topics	Basics Burst Start Interrupt (DS2004 Features 🕮) References ds2004_init 11

ds2004_burst_start_int_pending

Syntax

```
__INLINE UInt32 ds2004_burst_start_int_pending(phs_addr_t base)
```

Include file

`ds2004.h`

Purpose

To query if burst start interrupts are pending.

Description

This function returns the number of the channel with currently pending burst start interrupt. An interrupt is pending, if it has been requested but not yet acknowledged by the program.

If more than one channel have pending burst start interrupts, the channel number of the channel with the highest priority is returned. Channel 1 has the highest priority, channel 16 has the lowest priority.

If the function returns 0, no burst start interrupt is pending.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_burst_start_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

Return value

This function returns the channel number of the channel with pending burst start interrupt and with the highest priority. This function returns 0, if no burst start interrupt is pending.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if a burst start interrupt is pending on the 1st DS2004 board in a PHS-bus-based system. If so, the number of the involved channel with the highest priority is returned and stored in int_nr:

```
int_nr = ds2004_burst_start_int_pending(DS2004_1_BASE);
```

Related topics

Basics

Burst Start Interrupt (DS2004 Features 🕮)

References

ds2004_burst_start_int_enable....56
ds2004_init....11

ds2004_burst_start_int_read

Syntax

```
__INLINE void ds2004_burst_start_int_read(
   phs_addr_t base,
   UInt32 *mask)
```

Include file

`ds2004.h`

Purpose

To read the interrupt flag register of the burst start interrupt.

Description

This function returns a bit mask indicating which channels have requested a burst start interrupt. This function can be used to process the currently pending burst start interrupts in user-defined order.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_burst_start_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

mask Returns a bit mask indicating the channel numbers with pending burst start interrupt. The symbols are as follows. They can be used to evaluate the bit mask:

Symbol	Meaning
`DS2004_INT_CH1`	Pending burst start interrupt on channel 1.
`DS2004_INT_CH2`	Pending burst start interrupt on channel 2.
`...`	...
`DS2004_INT_CH15`	Pending burst start interrupt on channel 15.
`DS2004_INT_CH16`	Pending burst start interrupt on channel 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if burst start interrupts are pending on the 1st DS2004 board in a PHS-bus-based system. If so, `int_mask` contains a bit mask indicating the numbers of the involved channels:

```
ds2004_burst_start_int_read(DS2004_1_BASE, &int_mask);
```

Related topics

Basics

Burst Start Interrupt (DS2004 Features 🕮)

References

ds2004_burst_start_int_enable....56
ds2004_init....11

Data Ready Interrupt

Introduction

The DS2004 provides functions to react to a conversion burst completion. For further information, refer to Data Ready Interrupt (DS2004 Features 🕮).

Where to go from here

Information in this section

ds2004_data_ready_int_enable

Syntax

```
void ds2004_data_ready_int_enable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To enable a data ready interrupt.

Description

This function enables the data ready interrupt on the specified channel. Enabling the interrupt is a precondition for using the interrupt in your application.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to enable the data ready interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_data_ready_int_enable(DS2004_1_BASE, 1);
```

Related topics

Basics

Data Ready Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_data_ready_int_disable

Syntax

```
void ds2004_data_ready_int_disable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To disable a data ready interrupt.

Description

This function disables the data ready interrupt on the specified channel. Every interrupt that is no longer used should be disabled.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to disable the data ready interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_data_ready_int_disable(DS2004_1_BASE, 1);
```

Related topics

Basics

Data Ready Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_data_ready_int_ack

Syntax

```
__INLINE void ds2004_data_ready_int_ack(
   phs_addr_t base,
   UInt32 channel)
```

Include file	`ds2004.h`
Purpose	To acknowledge a data ready interrupt.
Description	This function acknowledges the data ready interrupt on the specified channel. The acknowledgement resets the associated flag in the interrupt flag register. An interrupt, that has been evaluated, must always be acknowledged. **Note** The `ds2004_init` function must be called before this function can be used.
I/O mapping	For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).
Parameters	**base** Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9). **channel** Specifies the channel number in the range 1 ... 16.
Return value	None
Execution times	For information on the execution times, refer to Function Execution Times on page 85.
Example	This example shows how to acknowledge the data ready interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system: `ds2004_data_ready_int_ack(DS2004_1_BASE, 1);`
Related topics	Basics Data Ready Interrupt (DS2004 Features 🕮) References ds2004_init 11

ds2004_data_ready_int_pending

Syntax	`__INLINE UInt32 ds2004_data_ready_int_pending(phs_addr_t base)`
Include file	`ds2004.h`
Purpose	To query if data ready interrupts are pending.
Description	This function returns the number of the channel with currently pending data ready interrupt. An interrupt is pending, if it has been requested but not yet acknowledged by the program. If more than one channel have pending data ready interrupts, the channel number of the channel with the highest priority is returned. Channel 1 has the highest priority, channel 16 has the lowest priority. If the function returns 0, no data ready interrupt is pending. **Note** ▪ The `ds2004_init` function must be called before this function can be used. ▪ The `ds2004_data_ready_int_enable` function must be called before the function returns valid information.
I/O mapping	For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).
Parameters	**base** Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).
Return value	This function returns the channel number of the channel with pending data ready interrupt and with the highest priority. This function returns 0, if no data ready interrupt is pending.
Execution times	For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if a data ready interrupt is pending on the 1st DS2004 board in a PHS-bus-based system. If so, the number of the involved channel with the highest priority is returned and stored in int_nr:

```
int_nr = ds2004_data_ready_int_pending(DS2004_1_BASE);
```

Related topics

Basics

Data Ready Interrupt (DS2004 Features 🕮)

References

ds2004_data_ready_int_enable.....63
ds2004_init.....11

ds2004_data_ready_int_read

Syntax

```
__INLINE void ds2004_data_ready_int_read(
   phs_addr_t base,
   UInt32 *mask)
```

Include file

`ds2004.h`

Purpose

To read the interrupt flag register of the data ready interrupt.

Description

This function returns a bit mask indicating which channels have requested a data ready interrupts. This function can be used to process the currently pending data ready interrupts in user-defined order.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_data_ready_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

mask Returns a bit mask indicating the channel numbers with pending data ready interrupt. The symbols are as follows. They can be used to evaluate the bit mask:

Symbol	Meaning
`DS2004_INT_CH1`	Pending data ready interrupt on channel 1.
`DS2004_INT_CH2`	Pending data ready interrupt on channel 2.
…	…
`DS2004_INT_CH15`	Pending data ready interrupt on channel 15.
`DS2004_INT_CH16`	Pending data ready interrupt on channel 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if data ready interrupts are pending on the 1st DS2004 board in a PHS-bus-based system. If so, `int_mask` contains a bit mask indicating the numbers of the involved channels:

```
ds2004_data_ready_int_read(DS2004_1_BASE, &int_mask);
```

Related topics

Basics

Data Ready Interrupt (DS2004 Features 🕮)

References

ds2004_data_ready_int_enable.....63
ds2004_init.....11

Conversion Trigger Overflow Interrupt

Introduction

The DS2004 provides functions to react to the loss of a conversion trigger. For further information, refer to Conversion Trigger Overflow Interrupt (DS2004 Features 🕮).

Where to go from here

Information in this section

ds2004_conv_trigger_ovfl_int_enable

Syntax

```
void ds2004_conv_trigger_ovfl_int_enable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To enable a conversion trigger overflow interrupt.

Description

This function enables the conversion trigger overflow interrupt on the specified channel. Enabling the interrupt is a precondition for using the interrupt in your application.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to enable the conversion trigger overflow interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_conv_trigger_ovfl_int_enable(DS2004_1_BASE, 1);
```

Related topics

Basics

Conversion Trigger Overflow Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_conv_trigger_ovfl_int_disable

Syntax

```
void ds2004_conv_trigger_ovfl_int_disable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To disable a conversion trigger overflow interrupt.

Description

This function disables the conversion trigger overflow interrupt on the specified channel. Every interrupt that is no longer used should be disabled.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to disable the conversion trigger overflow interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_conv_trigger_ovfl_int_disable(DS2004_1_BASE, 1);
```

Related topics

Basics

Conversion Trigger Overflow Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_conv_trigger_ovfl_int_ack

Syntax

```
__INLINE void ds2004_conv_trigger_ovfl_int_ack(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To acknowledge a conversion trigger overflow interrupt.

Description

This function acknowledges the conversion trigger overflow interrupt on the specified channel. The acknowledgement resets the associated flag in the interrupt flag register. An interrupt, that has been evaluated, must always be acknowledged.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to acknowledge the conversion trigger overflow interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_conv_trigger_ovfl_int_ack(DS2004_1_BASE, 1);
```

Related topics

Basics

Conversion Trigger Overflow Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_conv_trigger_ovfl_int_pending

Syntax

```
__INLINE UInt32 ds2004_conv_trigger_ovfl_int_pending(phs_addr_t base)
```

Include file

`ds2004.h`

Purpose

To query if conversion trigger overflow interrupts are pending.

Description

This function returns the number of the channel with currently pending conversion trigger overflow interrupt. An interrupt is pending, if it has been requested but not yet acknowledged by the program.

If more than one channel have pending conversion trigger overflow interrupts, the channel number of the channel with the highest priority is returned. Channel 1 has the highest priority, channel 16 has the lowest priority.

If the function returns 0, no conversion trigger overflow interrupt is pending.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_conv_trigger_ovfl_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

Return value

This function returns the channel number of the channel with pending conversion trigger overflow interrupt and with the highest priority. This function returns 0, if no conversion trigger overflow interrupt is pending.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if a conversion trigger overflow interrupt is pending on the 1st DS2004 board in a PHS-bus-based system. If so, `int_mask` contains a bit mask indicating the numbers of the involved channels:

```
int_nr = ds2004_conv_trigger_ovfl_int_pending(DS2004_1_BASE);
```

Related topics

Basics

Conversion Trigger Overflow Interrupt (DS2004 Features 🕮)

References

ds2004_conv_trigger_ovfl_int_enable....70
ds2004_init....11

ds2004_conv_trigger_ovfl_int_read

Syntax

```
__INLINE void ds2004_conv_trigger_ovfl_int_read(
   phs_addr_t base,
   UInt32 *mask)
```

Include file

`ds2004.h`

Purpose

To read the interrupt flag register of the conversion trigger overflow interrupt.

Description

This function returns a bit mask indicating which channels have requested a conversion trigger overflow interrupts. This function can be used to process the currently pending conversion trigger overflow interrupts in user-defined order.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_conv_trigger_ovfl_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

mask Returns a bit mask indicating the channel numbers with pending conversion trigger overflow interrupt. The symbols are as follows. They can be used to evaluate the bit mask:

Symbol	Meaning
`DS2004_INT_CH1`	Pending conversion trigger overflow interrupt on channel 1.
`DS2004_INT_CH2`	Pending conversion trigger overflow interrupt on channel 2.
...	...
`DS2004_INT_CH15`	Pending conversion trigger overflow interrupt on channel 15.
`DS2004_INT_CH16`	Pending conversion trigger overflow interrupt on channel 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if conversion trigger overflow interrupts are pending on the 1st DS2004 board in a PHS-bus-based system. If so, `int_mask` contains a bit mask indicating the numbers of the involved channels:

```
ds2004_conv_trigger_ovfl_int_read(DS2004_1_BASE, &int_mask);
```

Related topics

Basics

Conversion Trigger Overflow Interrupt (DS2004 Features 🕮)

References

ds2004_conv_trigger_ovfl_int_enable..... 70
ds2004_init..... 11

Data Lost Interrupt

Introduction

The DS2004 provides functions to react to the loss of conversion results. For further information, refer to Data Lost Interrupt (DS2004 Features 🕮).

Where to go from here

Information in this section

ds2004_data_lost_int_enable

Syntax

```
void ds2004_data_lost_int_enable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To enable a data lost interrupt.

Description

This function enables the data lost interrupt on the specified channel. Enabling the interrupt is a precondition for using the interrupt in your application.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to enable the data lost interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_data_lost_int_enable(DS2004_1_BASE, 1);
```

Related topics

Basics

Data Lost Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_data_lost_int_disable

Syntax

```
void ds2004_data_lost_int_disable(
   phs_addr_t base,
   UInt32 channel)
```

Include file

`ds2004.h`

Purpose

To disable a data lost interrupt.

Description

This function disables the data lost interrupt on the specified channel. Every interrupt that is no longer used should be disabled.

Note

The `ds2004_init` function must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

channel Specifies the channel number in the range 1 ... 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to disable the data lost interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system:

```
ds2004_data_lost_int_disable(DS2004_1_BASE, 1);
```

Related topics

Basics

Data Lost Interrupt (DS2004 Features 🕮)

References

ds2004_init 11

ds2004_data_lost_int_ack

Syntax

```
__INLINE void ds2004_data_lost_int_ack(
   phs_addr_t base,
   UInt32 channel)
```

Include file	`ds2004.h`
Purpose	To acknowledge a data lost interrupt.
Description	This function acknowledges the data lost interrupt on the specified channel. The acknowledgement resets the associated flag in the interrupt flag register. An interrupt, that has been evaluated, must always be acknowledged. **Note** The `ds2004_init` function must be called before this function can be used.
I/O mapping	For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).
Parameters	**base** Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9). **channel** Specifies the channel number in the range 1 ... 16.
Return value	None
Execution times	For information on the execution times, refer to Function Execution Times on page 85.
Example	This example shows how to acknowledge the data lost interrupt on channel 1 of the 1st DS2004 board in a PHS-bus-based system: `ds2004_data_lost_int_ack(DS2004_1_BASE, 1);`
Related topics	Basics Data Lost Interrupt (DS2004 Features 🕮) References ds2004_init..... 11

ds2004_data_lost_int_pending

Syntax

```
__INLINE UInt32 ds2004_data_lost_int_pending(phs_addr_t base)
```

Include file

`ds2004.h`

Purpose

To query if data lost interrupts are pending.

Description

This function returns the number of the channel with currently pending data lost interrupt. An interrupt is pending, if it has been requested but not yet acknowledged by the program.

If more than one channel have pending data lost interrupts, the channel number of the channel with the highest priority is returned. Channel 1 has the highest priority, channel 16 has the lowest priority.

If the function returns 0, no data lost interrupt is pending.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_data_lost_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

Return value

This function returns the channel number of the channel with pending data lost interrupt and with the highest priority. This function returns 0, if no data lost interrupt is pending.

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if a data lost interrupt is pending on the 1st DS2004 board in a PHS-bus-based system. If so, the number of the involved channel with the highest priority is returned and stored in int_nr:

```
int_nr = ds2004_data_lost_int_pending(DS2004_1_BASE);
```

Related topics

Basics

Data Lost Interrupt (DS2004 Features 🕮)

References

ds2004_data_lost_int_enable....78
ds2004_init....11

ds2004_data_lost_int_read

Syntax

```
__INLINE void ds2004_data_lost_int_read(
   phs_addr_t base,
   UInt32 *mask)
```

Include file

`ds2004.h`

Purpose

To read the interrupt flag register of the data lost interrupt.

Description

This function returns a bit mask indicating which channels have requested a data lost interrupts. This function can be used to process the currently pending data lost interrupts in user-defined order.

Note

- The `ds2004_init` function must be called before this function can be used.
- The `ds2004_data_lost_int_enable` function must be called before the function returns valid information.

I/O mapping

For information on the I/O mapping, refer to I/O Mapping of the DS2004 (DS2004 Features 🕮).

Parameters

base Specifies the PHS-bus base address of the board, see DSxxxx_n_BASE macros (refer to Base Address of the I/O Board on page 9).

mask Returns a bit mask indicating the channel numbers with pending data lost interrupt. The symbols are as follows. They can be used to evaluate the bit mask:

Symbol	Meaning
`DS2004_INT_CH1`	Pending data lost interrupt on channel 1.
`DS2004_INT_CH2`	Pending data lost interrupt on channel 2.
…	…
`DS2004_INT_CH15`	Pending data lost interrupt on channel 15.
`DS2004_INT_CH16`	Pending data lost interrupt on channel 16.

Return value

None

Execution times

For information on the execution times, refer to Function Execution Times on page 85.

Example

This example shows how to query if data lost interrupts are pending on the 1st DS2004 board in a PHS-bus-based system. If so, `int_mask` contains a bit mask indicating the numbers of the involved channels:

```
ds2004_data_lost_int_read(DS2004_1_BASE, &int_mask);
```

Related topics

Basics

Data Lost Interrupt (DS2004 Features 🕮)

References

ds2004_data_lost_int_enable....78
ds2004_init....11

Function Execution Times

Introduction

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

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 and the usage of inlining
- Parameters used

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<xxxx>` tool (optimization and inlining). The execution times in the tables below are always the mean measurement values.

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.

The properties of the processor boards used are:

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

Initialization

The following execution times have been measured for the initialization functions:

Function	**Mean Execution Time**	
	DS1006 with 2.6 GHz	**DS1006 with 3.0 GHz**
`ds2004_init`	131.96 ms	137.65 ms

ADC Unit

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

Function	**Mean Execution Time**	
	DS1006 with 2.6 GHz	**DS1006 with 3.0 GHz**
`ds2004_set_range`	10.00 ms	10.00 ms
`ds2004_set_conversion_trigger`	0.73 µs	0.73 µs
`ds2004_set_timer_period`	0.04 µs	0.03 µs
`ds2004_set_burst_mode`	0.74 µs	0.74 µs
`ds2004_set_burst_size`	0.04 µs	0.04 µs
`ds2004_set_burst_trigger`	0.73 µs	0.73 µs
`ds2004_channel_control`	0.73 µs	0.73 µs

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2004_sw_trigger	0.03 µs	0.01 µs
ds2004_read_buffer_count	0.73 µs	0.73 µs
ds2004_data_ready	0.01 µs	0.01 µs
ds2004_burst_init	10.01 ms	10.00 ms
ds2004_burst_in	1.29 + n[1)] · 0.315 µs	1.31 + n[1)] · 0.32 µs
ds2004_burst_read	1.29 + n[1)] · 0.315 µs	1.31 + n[1)] · 0.32 µs
ds2004_burst_in_immediate	2.85 + n[1)] · 0.315 µs	2.89 + n[1)] · 0.32 µs
ds2004_single_init	10.01 ms	10.00 ms
ds2004_single_in	0.66 µs	0.67 µs
ds2004_single_read	0.73 µs	0.73 µs
1) n stands for the number of values to be read (= burst size)		

Interrupts

The following execution times have been measured for the interrupt functions:

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2004_burst_start_int_enable	0.73 µs	0.73 µs
ds2004_burst_start_int_disable	0.73 µs	0.73 µs
ds2004_burst_start_int_ack	0.03 µ s	0.01 µs
ds2004_burst_start_int_pending	0.72 µs	0.72 µs
ds2004_burst_start_int_read	0.72 µs	0.72 µs
ds2004_data_ready_int_enable	0.73 µs	0.73 µs
ds2004_data_ready_int_disable	0.73 µs	0.73 µs
ds2004_data_ready_int_ack	0.03 µs	0.01 µs
ds2004_data_ready_int_pending	0.72 µs	0.72 µs
ds2004_data_ready_int_read	0.72 µs	0.72 µs
ds2004_conv_trigger_ovfl_int_enable	0.73 µs	0.73 µs
ds2004_conv_trigger_ovfl_int_disable	0.73 µs	0.73 µs
ds2004_conv_trigger_ovfl_int_ack	0.03 µs	0.01 µs
ds2004_conv_trigger_ovfl_int_pending	0.72 µs	0.72 µs
ds2004_conv_trigger_ovfl_int_read	0.72 µs	0.72 µs
ds2004_data_lost_int_enable	0.73 µs	0.73 µs
ds2004_data_lost_int_disable	0.73 µs	0.73 µs
ds2004_data_lost_int_ack	0.03 µs	0.01 µs

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
ds2004_data_lost_int_pending	0.72 µs	0.72 µs
ds2004_data_lost_int_read	0.72 µs	0.72 µs