DS3002 Incremental Encoder Interface 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

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

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

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

Contents

About This Reference

Content

This RTLib Reference (Real-Time Library) gives detailed descriptions of the C functions needed to program a DS3002 Incremental Encoder Interface 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.

Macros

Introduction

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

Base Address of the I/O Board

DSxxxx_n_BASE Macros

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

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

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

Example

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

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

Board Initialization

Objective

Before you can use the DS3002 board, you have to perform the initialization process.

> **Note**
>
> The initialization function of the processor board must be called before the DS3002 board's initialization function.

ds3002_init

Syntax

```
void ds3002_init(phs_addr_t base)
```

Include file

`ds3002.h`

Purpose

To initialize the DS3002 board:

- Reset on index pulse disabled
- Data sampling using SYNCIN line disabled
- All position counters are set to zero

Description

This is the basic initialization function for the DS3002 board. All setup parameters are set to default values as written below.

Initiating of input data sampling by the PHS-bus line /SYNCIN and resetting of the line counters on index pulse detection are disabled for all channels.

All line counters are preset to zero.

The function allocates and initializes some variables for internal usage and an ARCTAN table for fine position interpolation. If the allocation of dynamic memory fails, the function sets the error `DS3002_ALLOC_ERROR`. Otherwise `DS3002_NO_ERROR` is returned.

> **Note**
>
> This function must be called before using any of the DS3002 access functions described below.

Parameters

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

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
201	Error	ds3002_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.
-170	Error	ds3002_init(0x??): Board not found!	No DS3002 board could be found at the specified PHS-bus address. Check if the DSxxxx_n_BASE macro corresponds to the I/O board used.
-171	Error	ds3002_init(0x??): Memory allocation error!	The allocation of some dynamic memory for internal data storage has failed.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void main(void)
{
   init();
   ds3002_init(DS3002_1_BASE);
   ...
}
```

The DS3002 board at address DS3002_1_BASE is initialized.

Related topics

References

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

Incremental Encoder Interface

Objective

The DS3002 Incremental Encoder Interface Board features 6 independent incremental encoder interface channels and captures digital position signals and sinusoidal position signals.

Where to go from here

Information in this section

ds3002_clear_index

Syntax

```
void ds3002_clear_index(phs_addr_t base)
```

Include file

`ds3002.h`

Purpose

To clear the index flags of DS3002 board encoder channels.

Description

Since the index bits in the STB register are cleared automatically with each STB read access, a backup copy of the STB register keeps the individual index bits.

This function clears the STB register as well as the backup variable.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

Parameters

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

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   ...
   ds3002_clear_index();
   ...
}
```

The index flags of the DS3002 board are cleared.

Related topics

References

ds3002_read_count

Syntax

```
void ds3002_read_count(
   phs_addr_t base,
   long channel,
   Int64 *count)
```

Include file

`ds3002.h`

Purpose

To read the 42-bit position count of the specified encoder channel.

Description

This function reads the 42-bit position count of the DS3002 board. The count value consists of the 32-bit 4-fold line count and 10-bit fine position from sinusoidal encoder signals right-aligned within the 64-bit return value.

The returned line count corresponds to the position where the last input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

Note

The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

count Specifies the 32-bit 4-fold line count + 10-bit fine position.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   Int64 count
   ...
   ds3002_read_count(DS3002_1_BASE, 1, &count);
   ...
}
```

The 42-bit position count of channel 1 is read.

Related topics

References

Base Address of the I/O Board 7
ds3002_init 9
ds3002_read_line_count 26
ds3002_set_syncin_mode 29
ds3002_start 31
Macros 7

ds3002_read_index

Syntax

```
long ds3002_read_index(
   phs_addr_t base,
   long channel,
   long mode)
```

Include file

`ds3002.h`

Purpose

To indicate the detection of an index pulse.

Description

This function indicates whether an index pulse has been detected on the specified channel. The value 1 is returned in this case, 0 otherwise. Valid channel numbers are in the range 1 … 6.

Because the index flags for all DS3002 channels are automatically cleared by a read access to the corresponding register, they are copied to an internal variable to allow individual testing of each flag. The `mode` parameter specifies whether or not this internal variable shall be updated by an individual call to `ds3002_read_index`. Mode may be set to `DS3002_UPDATE` or `DS3002_NO_UPDATE`, respectively.

Note

The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

mode Specifies the mode of index reading:

Symbol	Meaning
`DS3002_UPDATE`	To get a new copy of index register
`DS3002_NO_UPDATE`	To keep the index register copy

Return value

The following values are returned:

Value	Meaning
0	No index detected
1	Index detected

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   ...
   if( ds3002_read_index(DS3002_1_BASE, 1, DS3002_UPDATE) )
      user_fct();
   ...
}
```

If an index has been detected on channel 1 of the DS3002 board, the function `user_fct` is called.

Related topics

References

Base Address of the I/O Board	7
ds3002_clear_index	15
ds3002_init	9
ds3002_test_index	34
Macros	7

ds3002_read_full_delta_line_pos

Syntax

```
dsfloat ds3002_read_full_delta_line_pos(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To evaluate the full 42-bit position difference (encoder lines) relative to last call.

Description

In some cases the position difference between two simulation steps is required, for example for velocity computation. This function returns the position difference relative to the previous call to the function `ds3002_read_full_delta_line_pos` scaled to encoder lines. The returned value corresponds to the previous two positions where input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

For calculation of the position difference the previous position count is saved in an internal variable. This variable is allocated and initialized with 0 by the **`ds3002_init`** function. Therefore the first call to `ds3002_read_full_delta_line_pos` will return the position difference relative to the position 0.

Note

- The function **`ds3002_init`** must be called before this function can be used.
- This function cannot be used in parallel with **`ds3002_read_full_delta_pos`** for the same channel, because both functions use the same global variable to store the previous position.
- If reset-on-index is set for the specified encoder channel, you have to regard the following situation: When an index has occurred before this function has been executed, the previously read position is set to 0. This causes a deviation between the real and the calculated delta position.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

Return value

Returns the delta position scaled to encoder lines within the range -2^{29} … $+2^{29}$.

Note

- The position value must be a 64-bit floating-point data type to yield the maximum resolution over the full position range.
- If the position value exceeds the valid range, it goes into the reverse.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat delta;
   ...
   delta = ds3002_read_full_delta_line_pos(DS3002_1_BASE, 1);
   ...
}
```

The delta position of encoder channel 1 is read and scaled to encoder lines, and stored into the **`delta`** variable.

Related topics

References

Base Address of the I/O Board	7
ds3002_init	9
ds3002_read_full_delta_pos	21
ds3002_set_syncin_mode	29
ds3002_start	31
Macros	7

ds3002_read_full_delta_pos

Syntax

```
dsfloat ds3002_read_full_delta_pos(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To evaluate the full 42-bit position difference scaled to a floating point value relative to last call.

Description

In some cases the position difference between two simulation steps is required, e.g. for velocity computation. This function returns the position difference relative to the previous call to the function `ds3002_read_full_delta_pos` scaled to the floating-point range –1.0 … +1.0. The returned value corresponds to the previous two positions where input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

For calculation of the position difference the previous position count is saved in an internal variable. This variable is allocated and initialized with 0 by the **`ds3002_init`** function. Therefore the first call to `ds3002_read_full_delta_pos` will return the position difference relative to the position 0.

Note

- The function **`ds3002_init`** must be called before this function can be used.
- This function cannot be used in parallel with **`ds3002_read_full_delta_line_pos`** for the same channel, because both functions use the same global variable to store the previous position.
- If reset-on-index is set for the specified encoder channel, you have to regard the following situation: When an index has occurred before this function has been executed, the previously read position is set to 0. This causes a deviation between the real and the calculated delta position.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 📖).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

Return value

The following values are returned:

Value	Meaning
–1.0 … +1.0	Delta position

Note

- The position value must be a 64-bit floating-point data type to yield the maximum resolution over the full position range.
- If the position value exceeds the valid range, it goes into the reverse.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat delta;
   ...
   delta = ds3002_read_full_delta_pos(DS3002_1_BASE, 1);
   ...
}
```

The delta position of encoder channel 1 is read and stored into the **`delta`** variable.

Related topics

References

Base Address of the I/O Board 7
ds3002_init 9
ds3002_read_full_delta_line_pos 19
ds3002_set_syncin_mode 29
ds3002_start 31

ds3002_read_full_line_pos

Syntax

```
dsfloat ds3002_read_full_line_pos(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To read the full 42-bit position value of the current channel in encoder lines.

Description

This function returns the current position of the specified channel scaled to encoder lines. The returned value corresponds to the position where the last input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

> **Note**
>
> - The position value must be a 64-bit floating-point data type to yield the maximum resolution over the full position range.
> - If the position value exceeds the valid range, it goes into the reverse.

Return value

Returns the position value scaled to encoder lines within the range -2^{29} … $+2^{29}$.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat position;
   ...
   position = ds3002_read_full_line_pos(DS3002_1_BASE, 1);
   ...
}
```

The full 42-bit position value, scaled to encoder lines, of encoder channel 1 is read and stored into the variable `position`.

Related topics

References

Base Address of the I/O Board	7
ds3002_init	9
ds3002_read_full_pos	24
ds3002_set_syncin_mode	29
ds3002_start	31
Macros	7

ds3002_read_full_pos

Syntax

```
dsfloat ds3002_read_full_pos(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To read the full 42-bit position of the current encoder channel as floating-point value.

Description

This function returns the current position of the specified channel scaled to a floating-point value within the range –1.0 … +1.0. The returned value corresponds to the position where the last input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

Return value

The following values are returned:

Value	Meaning
–1.0 … +1.0	Encoder position

> **Note**
>
> - The position value must be a 64-bit floating-point data type to yield the maximum resolution over the full position range.
> - If the position value exceeds the valid range, it goes into the reverse.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat position;
   ...
   position = ds3002_read_full_pos(DS3002_1_BASE, 1);
   ...
}
```

The position of encoder channel 1 is read and stored into the `position` variable.

Related topics

References

Base Address of the I/O Board........ 7
ds3002_set_syncin_mode........ 29
ds3002_start........ 31

ds3002_read_line_count

Syntax

```
Int32 ds3002_read_line_count(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To read the position counter as 32-bit signed integer.

Description

This function returns the position counter contents of the specified channel as a 32-bit signed integer. The returned line count corresponds to the position where the last input data sampling has been started either by calling **`ds3002_start`** or by a /SYNCIN signal from the DSP. Starting data sampling by a /SYNCIN signal must be enabled by calling **`ds3002_set_syncin_mode`**. The channel number must be in the range 1 … 6.

Because the DS3002 contains a 4-fold phase decoder, each encoder line yields 4 increments of the respective position counter.

Note

The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 📖).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   Int32 count;
   ...
   ds3002_start(DS3002_1_BASE, DS3001_MASK_1);
   count = ds3002_read_line_count(DS3002_1_BASE, 1);
   ...
}
```

The DS3002 encoder channel 1 is started and the line count value is read and stored into the `count` variable.

Related topics

References

Base Address of the I/O Board	7
ds3002_init	9
ds3002_read_count	16
ds3002_set_syncin_mode	29
ds3002_start	31
Macros	7

ds3002_set_counter_reset_mode

Syntax

```
void ds3002_set_counter_reset_mode(
   phs_addr_t base,
   long channel,
   long mode)
```

Include file `ds3002.h`

Purpose To enable the DS3002 board channel counter reset on index pulse.

Description The DS3002 line counters can be automatically cleared on index pulse detection. This function allows to select the counter reset mode for the specified channel.

The `mode` parameter can be `DS3002_NEVER` to disable automatic counter reset, `DS3002_ALWAYS` to clear the counter by each index pulse detected, or `DS3002_ONCE` to clear the counter on detection of the first index pulse only. The channel number must be in the range 1 … 6.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 📖).

Parameters **base** Specifies the PHS-bus base address. Refer to Base Address of the I/O Board on page 7.

channel Specifies the logical channel number within the range 1 … 6.

mode Specifies the counter reset mode:

Symbol	Meaning
`DS3002_NEVER`	To disable counter reset on index
`DS3002_ALWAYS`	To reset counter on every index pulse
`DS3002_ONCE`	To reset counter once on index pulse

> **Note**
>
> The current fine position count at the reset position depends on the input signals and is unpredictable. Thus the resolution of the automatic counter reset is limited to the 32-bit line count.

Return value None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds3002_set_counter_reset_mode(0x??): Board not initialized!	The DS3002 board has not been initialized by a preceding call to **`ds3002_init`**.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

void main(void)

```
{
   init();
   ds3002_init(DS3002_1_BASE);
   ds3002_set_counter_reset_mode(
      DS3002_1_BASE, 1, DS3002_ALWAYS);
   ...
}
```

The encoder counter of channel 1 is reset on every index pulse.

Related topics

References

Base Address of the I/O Board.. 7
ds3002_init.. 9
Macros.. 7

ds3002_set_syncin_mode

Syntax

```
void ds3002_set_syncin_mode(
   phs_addr_t base,
   long channel,
   long mode)
```

Include file

`ds3002.h`

Purpose

To enable the DS3002 data conversion by a SYNCIN signal.

Description

Input data sampling can be initiated automatically by the PHS-bus line /SYNCIN. This function can be used to enable or disable this feature.

Depending on the `mode` parameter , strobing of input data sampling by /SYNCIN is enabled or disabled for the specified channel. Mode must be set either to `DS3002_ENABLE` or `DS3002_DISABLE`. Valid channel numbers are in the range 1 … 6.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

mode Specifies the SYNCIN mode:

Symbol	Meaning
`DS3002_ENABLE`	For enabling data conversion by SYNCIN
`DS3002_DISABLE`	For disabling data conversion by SYNCIN

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds3002_set_syncin_mode(0x??): Board not initialized!	The DS3002 board has not been initialized by a preceding call to **`ds3002_init`**.

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void main(void)
{
   init();
   ds3002_init(DS3002_1_BASE);
   ds3002_set_syncin_mode(DS3002_1_BASE, 1, DS3002_ENABLE);
   ...
}
```

Data conversion by SYNCIN signal on channel 1 of the DS3002 board is enabled.

Related topics

References

Base Address of the I/O Board........ 7
ds3002_init........ 9
Macros........ 7

ds3002_start

Syntax

```
void ds3002_start(
   phs_addr_t base,
   long mask)
```

Include file

`ds3002.h`

Purpose

To start the data conversion on the specified channels.

Description

Input data sampling is started on the channels specified by the `mask` parameter. Mask can be any OR combination of the constants `DS3002_MASK_1` … `DS3002_MASK_6` to simultaneously initiate data sampling on any combination of channels, or `DS3002_MASK_ALL` to start data sampling on all channels.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

mask Specifies the bit mask with channels to be started:

Symbol	Meaning
`DS3002_MASK_1`	For channel 1
`DS3002_MASK_2`	For channel 2
`DS3002_MASK_3`	For channel 3
`DS3002_MASK_4`	For channel 4
`DS3002_MASK_5`	For channel 5
`DS3002_MASK_6`	For channel 6
`DS3002_MASK_ALL`	For all 6 channels

The definition can be combined using the logical OR operation.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   ...
   ds3002_start(
      DS3002_1_BASE, DS3002_MASK_1 | DS3002_MASK_3);
   ...
}
```

The DS3002 encoder channels 1 and 3 are started.

Related topics

References

Base Address of the I/O Board.. 7
ds3002_init.. 9
Macros.. 7

ds3002_sub64

Syntax

```
void ds3002_sub64(
   Int64 *src1,
   Int64 *src2,
   Int64 *res)
```

Include file

`ds3002.h`

Purpose

To perform a 64-bit subtraction of two 64-bit integer values.

Description

This function performs a full 64-bit subtraction of two 64-bit signed integer values. The result is also returned as a 64-bit signed integer. For more information about the 64-bit data representation refer to the description of **`ds3002_read_count`**.

The count values returned by **`ds3002_read_count`** may directly serve as source operands for `ds3002_sub64` if a full 64-bit position count difference is required.

Note

The function **`ds3002_init`** must be called before this function can be used.

Parameters

src1	Specifies the pointer to operand 1.
src2	Specifies the pointer to operand 2.
res	Specifies the pointer to the result value.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   Int64 count1, count2, result;
   ...
   ds3002_start(DS3002_1_BASE, DS3002_MASK_1);
   ds3002_read_count(DS3002_1_BASE, 1, &count1);
   ds3002_start(DS3002_1_BASE, DS3002_MASK_1);
   ds3002_read_count(DS3002_1_BASE, 1, &count2);
   ds3002_sub64(&count2, &count1, &result);
   ...
}
```

The difference between the position values `count1` and `count2` are calculated and stored into the `result` variable.

Related topics

References

ds3002_init	9
ds3002_read_count	16
Macros	7

ds3002_test_index

Syntax

```
long ds3002_test_index(
   phs_addr_t base,
   long channel)
```

Include file

`ds3002.h`

Purpose

To test the index flag of the specified channel.

Description

Since the index bits in the STB register are cleared automatically with each STB read access, a backup copy of the STB register keeps the individual index bits. The bit being tested is cleared afterwards.

Note

The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

Return value

The following values are returned:

Value	Meaning
0	No index detected
1	Index detected

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   ...
   if( ds3002_test_index(DS3002_1_BASE, 1) )
   user_fct();
   ...
}
```

If an index has been detected on channel 1 of the DS3002 board, the `user_fct` function is called.

Related topics

References

Base Address of the I/O Board........7
ds3002_clear_index........15
ds3002_init........9
ds3002_read_index........17
Macros........7

ds3002_write_full_line_pos

Syntax

```
void ds3002_write_full_line_pos(
   phs_addr_t base,
   long channel,
   dsfloat pos)
```

Include file

`ds3002.h`

Purpose

To write the position to the specified channel scaled to encoder lines within the full 42-bit position range.

Description

This function sets the position counter to the value corresponding to the position specified by the `pos` parameter. The parameter `pos` must be scaled to encoder lines.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 🕮).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

pos Specifies the position to be written to the specified encoder channel within the range -2^{29} … $+2^{29}$.

> **Note**
>
> - The position value must be a 64-bit floating-point data type to yield the maximum resolution over the full position range.
> - If the position value exceeds the valid range, it goes into the reverse.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat pos = 0.0;
   ...
   ds3002_write_full_line_pos(DS3002_1_BASE, 1, pos);
   ...
}
```

The position value 0 is written to the DS3002 encoder of channel 1.

Related topics

References

Base Address of the I/O Board 7
ds3002_init 9
ds3002_write_full_pos 37
ds3002_write_line_count 38

ds3002_write_full_pos

Syntax

```
void ds3002_write_full_pos(
   phs_addr_t base,
   long channel,
   dsfloat pos)
```

Include file

`ds3002.h`

Purpose

To write the position to the specified encoder channel scaled to a floating-point value within the full 42-bit position range.

Description

This function sets the position counter to the value corresponding to the position specified by the `pos` parameter.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

pos Specifies the position to be written to encoder channel within the range –1.0 … +1.0.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   dsfloat pos = 0.0;
   ...
   ds3002_write_full_pos(DS3002_1_BASE, 1, pos);
   ...
}
```

The position value 0 is written to the DS3002 encoder of channel 1.

Related topics

References

Base Address of the I/O Board..7
ds3002_init..9

ds3002_write_line_count

Syntax

```
void ds3002_write_line_count(
   phs_addr_t base,
   long channel,
   Int32 value)
```

Include file

`ds3002.h`

Purpose

To write the position counter to the specified channel as 32-bit integer.

Description

The 32-bit position counter of the specified channel is set to the new value given by the `value` parameter. It can be any 32-bit signed integer in the range –2,147,483,648 … +2,147,483,647. The channel number must be in the range 1 … 6.

The current fine position is evaluated and saved in an internal data structure for consideration by the position and counter readout functions.

> **Note**
>
> The function **`ds3002_init`** must be called before this function can be used.

I/O mapping

For information on the I/O mapping, refer to Mapping of I/O Signals (PHS Bus System Hardware Reference 📖).

Parameters

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

channel Specifies the logical channel number within the range 1 … 6.

value Specifies the value to be written to the counter (4-fold line count) within the range –2,147,483,648 … +2,147,483,647.

Return value

None

Execution times

For information, refer to Function Execution Times on page 41.

Example

This example shows how to use this function:

```
void sub_fct(void)
{
   Int32 count = 0;
   ...
   ds3002_write_line_count(DS3002_1_BASE, 1, count);
   ...
}
```

The value 0 is written to the DS3002 position counter of channel 1.

Related topics

References

Base Address of the I/O Board	7
ds3002_init	9
Macros	7

Function Execution Times

Objective

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

Where to go from here

Information in this section

Information on the Test Environment

Test environment

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

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

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

The properties of the processor boards used are:

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

Related topics

References

Measured Execution Times... 42

Measured Execution Times

Execution times are available for the following RTLib units:

- Initialization
- Incremental encoder interface

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.

Initialization

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

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds3002_init`	1.38 ms	363.94 µs

Incremental encoder interface

The following execution times have been measured for the incremental encoder interface functions:

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds3002_set_syncin_mode`	0.85 µs	0.69 µs
`ds3002_set_counter_reset_mode`	0.72 µs	0.61 µs
`ds3002_start`	0.038 µs	0.029 µs
`ds3002_read_index`	0.79 µs	0.61 µs
`ds3002_test_index`	0.62 µs	0.60 µs
`ds3002_clear_index`	0.62 µs	0.59 µs
`ds3002_sub64`	0.22 µs	0.048 µs
`ds3002_read_count`	1.21 µs[1]/1.87 µs[2]	1.25 µs[1]/1.87 µs[2]

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds3002_read_full_delta_line_pos`	1.29 µs[1]/1.95 µs[2]	1.33 µs[1]/1.93 µs[2]
`ds3002_read_full_delta_pos`	1.24 µs[1]/1.95 µs[2]	1.36 µs[1]/1.94 µs[2]
`ds3002_read_full_line_pos`	1.24 µs[1]/1.91 µs[2]	1.16 µs[1]/1.81 µs[2]
`ds3002_read_full_pos`	1.23 µs[1]/1.89 µs[2]	1.16 µs[1]/1.83 µs[2]
`ds3002_read_line_count`	1.19 µs[1]/1.56 µs[2]	1.15 µs[1]/1.52 µs[2]
`ds3002_write_full_line_pos`	2.83 µs[1]/2.57 µs[2]	2.83 µs[1]/2.56 µs[2]
`ds3002_write_full_pos`	2.77 µs[1]/2.54 µs[2]	2.79 µs[1]/2.53 µs[2]
`ds3002_write_line_count`	2.75 µs[1]/2.53 µs[2]	2.71 µs[1]/2.50 µs[2]

[1] The data conversion was already finished when calling this function.

[2] The data conversion has been started immediately before calling this function.

Related topics

References

Information on the Test Environment.. 41