DS2001 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 DS2001 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.

Macros

Introduction

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

Base Address of the I/O Board

DSxxxx_n_BASE Macros

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

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

Note

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

Example

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

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

Board Initialization

Introduction

Before you can use the DS2001, you have to perform the initialization process.

Note

The initialization function of the processor board `init` must be called before the DS2001's initialization function `ds2001_init`.

ds2001_init

Syntax

```
void ds2001_init(phs_addr_t base)
```

Include file

`ds2001.h`

Purpose

To initialize the DS2001.

Description

All DS2001 registers are initialized to default values:

- –10 … +10 V input voltage range
- Hold input during conversion
- 16-bit word length
- Interrupt line 0
- Polling mode of the Interrupt Control Unit

Note

This function must be called before any other DS2001 function can be used.

Parameters

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

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
201	Error	ds2001_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.
-140	Error	ds2001_init(0x??): Board not found!	No DS2001 board could be found at the specified PHS-bus address. Check if the DSxxxx_n_BASE macro corresponds to the I/O board used.
-53	Warning	ds2001_init(0x??): Jumper setup is not matching SW default initialization! STP register: 0x???????? instead of 0x????????	The value of the STP register could not be verified because the jumper setting is not correct.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to initialize a DS2001:

```
void main(void)
{
   init();
   ds2001_init(DS2001_1_BASE);
   …
}
```

Related topics

References

ADC Unit

Introduction

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

For further information about the ADC unit of the DS2001 board, refer to ADC Unit (DS2001 Features 🕮).

Note

You have to initialize the DS2001 with the `ds2001_init` function before you can use one of these functions.

Where to go from here

Information in this section

To change the ADC settings

To start the A/D conversion

To poll the end of conversion (EOC) flag

To read values from the A/D converter

ds2001_set_range

Syntax

```
void ds2001_set_range(
   phs_addr_t base,
   int channel,
   int range)
```

Include file

`ds2001.h`

Purpose

To select the input voltage range of the specified A/D converter channels.

> **Note**
>
> The `ds2001_init` function must be called before this function can be used.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

Parameters

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

channel Specifies the channel number within the range 1 … 5. To select all 5 channels, use `DS2001_CH_ALL`.

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

Predefined Symbol	Input Voltage Range
`DS2001_RNG5`	–5 V … +5 V
`DS2001_RNG10`	–10 V … +10 V (initial value)

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2001_set_range(0x??): Board not initialized!	The DS2001 has not been initialized by a preceding call to the `ds2001_init` function.
-54	Error	ds2001_set_range(0x??): Specified value is not matching jumper setup! STP register: 0x???????? instead of 0x????????	The value of the STP register could not be verified because the jumper setting is not correct.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to set the input voltage range of channel 1 to -5 V … +5 V:

```
ds2001_set_range(DS2001_1_BASE, 1, DS2001_RNG5);
```

Related topics

References

ds2001_set_shmode

Syntax

```
void ds2001_set_shmode(
   phs_addr_t base,
   int channel,
   int shmode)
```

Include file

`ds2001.h`

Purpose

To select the sample/hold mode of the specified A/D converter channels.

Note

The `ds2001_init` function must be called before this function can be used.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

Parameters

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

channel Specifies the channel number within the range 1 … 5. To select all 5 channels, use `DS2001_CH_ALL`.

shmode Specifies the sample/hold mode. After the board initialization the sample/hold mode is set to holding input during conversion. If you start the A/D conversion the actual value of the input signal is stored. This analog value is

converted into a digital value. If you change the sample/hold mode to tracking input during conversion, the A/D conversion is done directly with the input signal that can change during conversion. The following symbols are predefined:

Predefined Symbol	Meaning
`DS2001_TRK`	Tracking input during conversion
`DS2001_HLD`	Holding input during conversion (initial value)

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2001_set_shmode(0x??): Board not initialized!	The DS2001 has not been initialized by a preceding call to the `ds2001_init` function.
-54	Error	ds2001_set_shmode(0x??): Specified value is not matching jumper setup! STP register: 0x???????? instead of 0x????????	The value of the STP register could not be verified because the jumper setting is not correct.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to set channel 4 to tracking during conversion:

```
ds2001_set_shmode(DS2001_1_BASE, 4, DS2001_TRK);
```

Related topics

References

ds2001_set_wordlen

Syntax

```
void ds2001_set_wordlen(
   phs_addr_t base,
   int channel,
   int wordlen)
```

Include file

`ds2001.h`

Purpose

To select the word length of the specified A/D converter channels.

Note

The `ds2001_init` function must be called before this function can be used.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

Parameters

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

channel Specifies the channel number within the range 1 … 5. To select all 5 channels, use `DS2001_CH_ALL`.

wordlen Specifies the word length of ADC data as number of bits. The following symbols are predefined:

Predefined Symbol	Word Length / Bit
`DS2001_LEN4`	4
`DS2001_LEN8`	8
`DS2001_LEN12`	12
`DS2001_LEN16`	16 (initial value)

Return value

None

Messages

The following message is defined:

ID	Type	Message	Description
-50	Error	ds2001_set_wordlen(0x??): Board not initialized!	The DS2001 has not been initialized by a preceding call to the `ds2001_init` function.
-54	Error	ds2001_set_wordlen(0x??): Specified value is not matching jumper setup! STP register: 0x???????? instead of 0x????????	The value of the STP register could not be verified because the jumper setting is not correct.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to set all channels to 12-bit ADC word length:

```
ds2001_set_wordlen(DS2001_1_BASE, DS2001_CH_ALL,
        DS2001_LEN12);
```

Related topics

References

ds2001_start

Syntax

```
void ds2001_start(
   phs_addr_t base,
   Ulnt32 mask)
```

Include file

`ds2001.h`

Purpose

To start the conversion of the specified A/D converter channels.

Description

The specified A/D converters start synchronously.

Note

The `ds2001_init` function must be called before this function can be used.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

Parameters

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

mask Specifies the channels to be converted. You can combine the following predefined symbols using the logical OR operation:

Predefined Symbol	Meaning
`DS2001_CVT1`	Starts A/D conversion for channel 1
`DS2001_CVT2`	Starts A/D conversion for channel 2

Predefined Symbol	Meaning
DS2001_CVT3	Starts A/D conversion for channel 3
DS2001_CVT4	Starts A/D conversion for channel 4
DS2001_CVT5	Starts A/D conversion for channel 5
DS2001_CVT_ALL	Starts A/D conversion for all 5 channels

Return value

None

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to start A/D conversion of channels 1 and 4 synchronously:

```
ds2001_start(DS2001_1_BASE, DS2001_CVT1 | DS2001_CVT4);
```

Related topics

References

ds2001_ready

Syntax

```
int ds2001_ready(
   phs_addr_t base,
   int channel)
```

Include file

ds2001.h

Purpose

To indicate the conversion status of the specified channel.

Description

The *end of conversion* (EOC) flag of the specified channel is polled.

Note

- The `ds2001_init` function must be called and the conversion must be started by a preceding call to `ds2001_start` before this function can be used.
- The DS2001 board interrupt control unit must be initialized to polling mode. This is done in the processor board's initialization function. If the interrupt control unit is not in the polling mode, the function will return erroneous results. For further information, refer to Limitations (DS2001 Features 🕮).

Tip

You can use this function to make the `ds2001_read` function waiting on the *end of conversion* flag.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

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 … 5.

Return value

The following values are returned:

Value	Meaning
0	Conversion of specified channel has not finished
1	Conversion of specified channel has finished

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to use the function:

```
while( !ds2001_ready(DS2001_1_BASE, 1) );
```

The code is waiting until the conversion of channel 1 is finished.

Related topics

Basics

Limitations (DS2001 Features 🕮)

References

ds2001_in

Syntax

```
dsfloat ds2001_in(
   phs_addr_t base,
   int channel)
```

Include file

`ds2001.h`

Purpose

To read values from a specified A/D converter channel after end of conversion.

Description

The *end of conversion* flag of the specified channel is polled until conversion is complete. The ADC value is read and scaled to a floating-point value within the range –1.0 … +1.0.

Note

- The `ds2001_init` function must be called and the conversion must be started by a preceding call to `ds2001_start` before this function can be used.
- The DS2001 board interrupt control unit must be initialized to polling mode. This is done in the processor board's initialization function. If the interrupt control unit is not in the polling mode, the function will block the processor board. For further information, refer to Limitations (DS2001 Features 🕮).

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

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 … 5.

Return value

This function returns the A/D value within the range –1.0 … +1.0.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to read the ADC value of channel 1:

```
void sub_fct(void)
{
   dsfloat adc_value;
   ds2001_start(DS2001_1_BASE, DS2001_CVT1);
   adc_value = ds2001_in(DS2001_1_BASE, 1);
   …
}
```

Related topics

Basics

Limitations (DS2001 Features 🕮)

References

ds2001_read

Syntax

```
dsfloat ds2001_read(
   phs_addr_t base,
   int channel)
```

Include file

`ds2001.h`

Purpose

To read values from a specified A/D converter channel immediately.

Description

The ADC value is read immediately and scaled to a floating-point value within the range –1.0 … +1.0. This function can be used in a service routine for the *end of conversion* interrupt.

Note

The `ds2001_init` function must be called, the corresponding interrupt must be initialized, and the conversion must be started by a preceding call to `ds2001_start`.

Tip

With the `ds2001_ready` function, you can poll the *end of conversion* flag. Instead of using `ds2001_ready` and `ds2001_read`, it is recommended to use the `ds2001_in` function.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2001 Features 🕮).

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 … 5.

Return value

This function returns an A/D value within the range –1.0 … +1.0.

Execution times

For information, refer to Function Execution Times on page 23.

Example

This example shows how to use the function:

```
dsfloat adc_value;
void adc_service(void)
{
   adc_value = ds2001_read(DS2001_1_BASE, 1);
}
```

The ADC value of channel 1 is read in an interrupt service routine.

Related topics

References

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

The properties of the processor boards used are:

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

Measured Execution Times

Execution times are available for the following RTLib units:

- Initialization
- ADC unit

Note

The following execution times contain mean values for a sequence of I/O accesses. The execution time of a single call might be lower because of buffered I/O access.

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
`ds2001_init`	43.50 µs	55.85 µs

ADC unit

The following execution times have been measured for the functions of the ADC unit:

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds2001_set_range`	2.18 µs	2.23 µs
`ds2001_set_shmode`	2.19 µs	2.24 µs
`ds2001_set_wordlen`	2.19 µs	2.24 µs
`ds2001_start`	1.59 µs	1.65 µs
`ds2001_ready`	1.40 µs	1.46 µs
`ds2001_in`	1.98 µs	2.04 µs
`ds2001_read`	1.41 µs	1.47 µs

B

C

D

F

L