DS2002 Multi-Channel 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 DS2002 Multi-Channel 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

Before you can use the DS2002, you have to perform the initialization process.

Note

The initialization function of the processor board `init` must be called before the DS2002's initialization function `ds2002_init`.

ds2002_init

Syntax

```
void ds2002_init(phs_addr_t base)
```

Include file

`ds2002.h`

Purpose

To initialize the DS2002.

Description

All DS2002 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 DS2002 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	ds2002_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	ds2002_init(0x??): Board not found!	No DS2002 board could be found at the specified PHS-bus address. Check if the DSxxxx_n_BASE macro corresponds to the I/O board used.
-142	Error	ds2002_init(0x??): Memory allocation error!	The allocation of some dynamic memory for internal data storage has failed.
-53	Warning	ds2002_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 29.

Example

This example shows how to initialize a DS2002:

```
void main(void)
{
   init();
   ds2002_init(DS2002_1_BASE);
   …
}
```

Related topics

References

ADC Unit

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

For further information about the ADC unit of the DS2002 board, refer to ADC Unit (DS2002 Features 🕮).

Note

You have to initialize the DS2002 with `ds2002_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 select a channel of the A/D converter

To start the conversion

To poll the end of conversion (EOC) flag

To read values from the A/D converter

To initialize and read values in blocks

ds2002_set_range

Syntax

```
void ds2002_set_range(
   phs_addr_t base,
   int group,
   int range)
```

Include file

`ds2002.h`

Purpose

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

Description

This function selects the input voltage range of one channel group or all channels on the DS2002 at the specified PHS-bus base address.

Note

The `ds2002_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 (DS2002 Features 🕮).

Parameters

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

group Specifies the channel group. The following symbols are predefined:

Predefined Symbol	Meaning
DS2002_GROUP1	Selects channel group 1 (channel 1 … 16)
DS2002_GROUP2	Selects channel group 2 (channel 17 … 32)
DS2002_CH_ALL	Selects all channels

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

Predefined Symbol	Input Voltage Range
DS2002_RNG5	–5 … +5 V
DS2002_RNG10	–10 … +10 V (initial value)

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2002_set_range(0x??): Board not initialized!	The DS2002 has not been initialized by a preceding call to the `ds2002_init` function.
-54	Error	ds2002_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 29.

Example

This example shows how to use the function:

```
ds2002_set_range(DS2002_1_BASE, DS2002_GROUP1, DS2002_RNG5);
```

Channel group 1 of the DS2002 is set to the input voltage range –5 … +5 V.

Related topics

References

ds2002_set_shmode

Syntax

```
void ds2002_set_shmode(
   phs_addr_t base,
   int group,
   int shmode)
```

Include file

`ds2002.h`

Purpose

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

Description

This function selects the sample/hold mode of one channel group or all channels on the DS2002 at the specified PHS-bus base address.

Note

The `ds2002_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 (DS2002 Features 🕮).

Parameters

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

group Specifies the channel group. The following symbols are predefined:

Predefined Symbol	Meaning
`DS2002_GROUP1`	Selects channel group 1 (channel 1 … 16)
`DS2002_GROUP2`	Selects channel group 2 (channel 17 … 32)
`DS2002_CH_ALL`	Selects all channels

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
DS2002_TRK	Tracking input during conversion
DS2002_HLD	Holding input during conversion (initial value)

Note

The DS2002 that is equipped with fast ADCs does not support the tracking mode. Independently from this parameter setting, the A/D converter is always working in holding mode. Nevertheless, for an optimal signal quality, it is recommended to use the DS2002_TRK setting.
This board variant is specially marked with an imprint (Fast ADC) on its bracket.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2002_set_shmode(0x??): Board not initialized!	The DS2002 has not been initialized by a preceding call to the ds2002_init function.
-54	Error	ds2002_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 29.

Example

This example shows how to use the function:

```
ds2002_set_range(DS2002_1_BASE, DS2002_GROUP2, DS2002_TRK);
```

Channel group 2 (channel 17 … 32) of the DS2002 is set to tracking during conversion.

Related topics

References

Base Address of the I/O Board....7
ds2002_init....9
Macros....7

ds2002_set_wordlen

Syntax

```
void ds2002_set_wordlen(
   phs_addr_t base,
   int group,
   int wordlen)
```

Include file

`ds2002.h`

Purpose

To select the word length of the A/D converters.

Description

This function selects the word length of one channel group or all channels on the DS2002 at the specified PHS-bus base address.

Note

The `ds2002_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 (DS2002 Features 🕮).

Parameters

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

group Specifies the channel group. The following symbols are predefined:

Predefined Symbol	Meaning
`DS2002_GROUP1`	Selects channel group 1 (channel 1 … 16)
`DS2002_GROUP2`	Selects channel group 2 (channel 17 … 32)
`DS2002_CH_ALL`	Selects all channels

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

Predefined Symbol	Word Length in Bit
`DS2002_LEN4`	4
`DS2002_LEN8`	8
`DS2002_LEN12`	12
`DS2002_LEN16`	16 (initial value)

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-50	Error	ds2002_set_wordlen(0x??): Board not initialized!	The DS2002 has not been initialized by a preceding call to the `ds2002_init` function.
-54	Error	ds2002_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 29.

Example

This example shows how to use this function:

```
ds2002_set_wordlen(DS2002_1_BASE,DS2002_CH_ALL,DS2002_LEN12);
```

All channels of the DS2002 are set to 12-bit ADC word length.

Related topics

References

Base Address of the I/O Board....7
ds2002_init....9
Macros....7

ds2002_select_channel

Syntax

```
void ds2002_select_channel(
   phs_addr_t base,
   int channel)
```

Include file

`ds2002.h`

Purpose

To select the input channel on a DS2002.

Description

This function selects the input channel which is accessed by the `ds2002_start`, `ds2002_in`, `ds2002_read` functions.

Note

- The `ds2002_init` function must be called before this function can be used.
- This function is not reentrant.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2002 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 … 32.

Return value

None

Execution times

For information, refer to Function Execution Times on page 29.

Example

This example shows how to use the function:

```
ds2002_select_channel(DS2002_1_BASE, 30);
```

Channel 30 of the DS2002 is selected.

Related topics

References

ds2002_start

Syntax

```
void ds2002_start(phs_addr_t base)
```

Include file	`ds2002.h`
Purpose	To start the conversion of a DS2002 A/D converter.
Description	This function starts the A/D conversion of one channel on the DS2002 at the specified PHS-bus base address. The channel must be selected with `ds2002_select_channel` function before. **Note** The `ds2002_init` function must be called before this function can be used.
Parameters	**base** Specifies the PHS-bus base address. Refer to Base Address of the I/O Board on page 7.
Return value	None
Execution times	For information, refer to Function Execution Times on page 29.
Example	This example shows how to use the function: `ds2002_start(DS2002_1_BASE);` The conversion of the selected channel of the DS2002 is started.
Related topics	References Base Address of the I/O Board....7 ds2002_in....21 ds2002_init....9 ds2002_read....23 ds2002_ready....20 ds2002_select_channel....17 Macros....7

ds2002_ready

Syntax

```
int ds2002_ready(
   phs_addr_t base,
   int group)
```

Include file

`ds2002.h`

Purpose

To indicate the conversion status of a DS2002's channel group.

Description

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

Note

- The `ds2002_init` function must be called and the conversion must be started by a preceding call to `ds2002_start` before this function can be used.
- The DS2002's interrupt control unit (ICU) must be initialized to polling mode. This is done in the processor board's initialization function. If the ICU is in interrupt mode, the function will return erroneous results. For further information, refer to Limitations (DS2002 Features 🕮).

Tip

You can use this function to make the `ds2002_read` function waiting on the *end of conversion* flag.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2002 Features 🕮).

Parameters

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

group Specifies the channel group. The following symbols are predefined:

Predefined Symbol	Meaning
`DS2002_GROUP1`	Selects channel group 1 (channel 1 … 16)
`DS2002_GROUP2`	Selects channel group 2 (channel 17 … 32)

Return value

The following values are returned:

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

Execution times

For information, refer to Function Execution Times on page 29.

Example

This example shows how to use the function:

```
while( !ds2002_ready(DS2002_1_BASE, DS2002_GROUP1) );
```

The code is waiting until the conversion of the selected group is finished.

Related topics

Basics

Limitations (DS2002 Features 🕮)

References

ds2002_in

Syntax

```
dsfloat ds2002_in(phs_addr_t base)
```

Include file

`ds2002.h`

Purpose

To read values from a DS2002 channel after end of conversion.

Description

The *end of conversion* flag of the selected 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

Before you can use this function, you have to call:
1. `ds2002_init` to initialize the board
2. `ds2002_select_channel` to select the channel
3. `ds2002_start` to start the conversion

The DS2002's interrupt control unit (ICU) must be initialized to polling mode. This is done in the processor board's initialization function. If the ICU is in interrupt mode, the function will block the processor board. For further information, refer to Limitations (DS2002 Features 🕮).

Parameters

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

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 29.

Example

This example shows how to use the function:

```
void sub_fct(void)
{
   float adc_value;
   ds2002_select_channel(DS2002_1_BASE, 1);
   ds2002_start(DS2002_1_BASE);
   adc_value = ds2002_in(DS2002_1_BASE);
   …
}
```

The ADC value of channel 1 is read.

Related topics

Basics

Limitations (DS2002 Features 🕮)

References

ds2002_read

Syntax

```
float ds2002_read(phs_addr_t base)
```

Include file

`ds2002.h`

Purpose

To read values from a DS2002 channel immediately.

Description

The ADC value is read immediately without polling the end-of-conversion flag 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

Before you can use this function, you have to call:

1. `ds2002_init` to initialize the board
2. `ds2002_select_channel` to select the channel
3. `ds2002_start` to start the conversion

Tip

With the `ds2002_ready` function, you can poll the *end of conversion* flag. Instead of using `ds2002_ready` and `ds2002_read`, it is recommended to use the `ds2002_in` function.

Parameters

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

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 29.

Example

This example shows how to use the function:

```
float adc_value;
void adc_service(void)
{
   adc_value = ds2002_read(DS2002_1_BASE);
}
```

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

Related topics

References

ds2002_single_in

Syntax

```
dsfloat ds2002_single_in(
   phs_addr_t base,
   int channel)
```

Include file

`ds2002.h`

Purpose

To read values from a specified DS2002 channel after end-of-conversion.

Description

The specified channel is selected and started. The EOC flag is polled until the conversion is finished. Hence, this function is not suitable for an EOC interrupt service routine. The ADC value is read and scaled to a floating-point value within the range –1.0 … +1.0.

Note

- The `ds2002_init` function must be called before this function can be used.
- It is not necessary to call the `ds2002_start` function.
- This function is not reentrant.
- The DS2002's interrupt control unit must be initialized to polling mode. This is done in the processor board's initialization function. If the ICU is in interrupt mode, the function will block the processor board. For further information, refer to Limitations (DS2002 Features 🕮).

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2002 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 … 32.

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 29.

Example

This example shows how to use the function:

```
dsfloat adc_value;
adc_value = ds2002_single_in(DS2002_1_BASE, 1);
```

The ADC value of channel 1 is read.

Related topics

Basics

Limitations (DS2002 Features 🕮)

References

ds2002_block_init

Syntax

```
void ds2002_block_init(
   phs_addr_t base,
   int count,
   int *channels)
```

Include file

`ds2002.h`

Purpose

To initialize the ADC block input.

Description

This function initializes the DS2002 block input.

The channels to be used for block input must be specified in arbitrary order in the `*channels` array. This array must be supplied by the calling program. The number of channels to be used is limited to 16 per A/D converter. This means that 16 channels within the range 1 … 16 and 16 channels within the range 17 … 32 can be specified.

The number of channels to be used must be specified in the `count` parameter.

Note

- The `ds2002_init` function must be called before this function can be used.
- This function is not reentrant.

I/O mapping

For details on the I/O mapping, refer to ADC Unit (DS2002 Features 🕮).

Parameters

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

count Specifies the number of channels to be used for block input within the range 1 … 32.

channels Specifies the pointer of an array of channel numbers to be used for block input.

Return value

None

Messages

The following messages are defined:

ID	Type	Message	Description
-143	Error	ds2002_block_init(0x??): Initialization error!	More than 16 channels per A/D converter are specified in the `*channels` array.
-50	Error	ds2002_block_init(0x??): Board not initialized!	The DS2002 has not been initialized by a preceding call to the `ds2002_init` function.

Execution times

For information, refer to Function Execution Times on page 29.

Example

This example shows how to use the function:

```
int channels[5] = {1, 5, 11, 20, 32};
int count = 5;              /* number of used channels */
void main(void)
{
   …
   ds2002_block_init(DS2002_1_BASE, count, channels);
   …
}
```

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

Related topics

References

ds2002_block_in

Syntax

```
void ds2002_block_in(
   phs_addr_t base,
   dsfloat *data)
```

Include file

`ds2002.h`

Purpose

To read values from the specified DS2002 block after end of conversion.

Description

The DS2002 channels specified in the preceding call to `ds2002_block_init` are converted and read. The ADC values are scaled to floating-point values within the range –1.0 … +1.0 and stored into the `*data` array. The values are stored in the same order as specified in the `*channels` array from the function call to `ds2002_block_init`.

Note

- The `ds2002_init` and `ds2002_block_init` functions must be called before this function can be used.
- This function is not reentrant.
- The array lengths of the `data` and `channels` parameter must be equal.
- The DS2002's interrupt control unit (ICU) must be initialized to polling mode. This is done in the initialization function of the processor board. If the ICU is in interrupt mode, the function will block the DSP. For further information, refer to Limitations (DS2002 Features 🕮).

Parameters

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

data Specifies the address of the data array for ADC values.

Return value

None

Execution times

For information, refer to Function Execution Times on page 29.

Example

This example shows how to use the function:

```
void sub_fct(void)
{
   int channels[5] = {1, 5, 11, 20, 32};
   int count = 5;              /* number of used channels */
   float adc_array[5];
   …
   ds2002_block_init(DS2002_1_BASE, count, channels);
   ds2002_block_in(DS2002_1_BASE, adc_array);
   …
}
```

The ADC values are read and written into `adc_array`.

Related topics

Basics

Limitations (DS2002 Features 🕮)

References

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<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.
- The execution times are also listed for the DS2002 that is equipped with fast ADCs. This board variant is specially marked with an imprint (Fast ADC) on its bracket.

Initialization

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

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds2002_init`	44.12 µs	55.93 µs

ADC unit

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

Function	Mean Execution Time	
	DS1006 with 2.6 GHz	DS1006 with 3.0 GHz
`ds2002_set_range`	2.23 µs	2.22 µs
`ds2002_set_shmode`	2.23 µs	2.22 µs
`ds2002_set_wordlen`	2.23 µs	2.22 µs
`ds2002_select_channel`	3.49 µs	3.48 µs
`ds2002_start`	2.23 µs	2.22 µs
`ds2002_ready`	0.59 µs	0.58 µs
`ds2002_in`	2.04 µs	2.02 µs
`ds2002_read`	1.45 µs	1.45 µs
`ds2002_single_in`	9.50 µs	9.93 µs
`ds2002_block_init`	1.301 + n[1)] · 0.004 µs	1.32 + n[1)] · 0.003 µs
`ds2002_block_in`	2.590 + n[1)] · 2.878 µs	3.087 + n[1)] · 2.83 µs

[1)] n stands for the number of channels used.

B

C

D

F

L