DS2301 Direct Digital Synthesis 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

Introduction

This reference gives detailed descriptions of the C functions needed to program a DS2301 Direct Digital Synthesis Board. The C functions can be used to program RTI-specific Simulink S-functions, or to implement your real-time 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

Initialization Function

Introduction

Before you can use the DS2301 board, you have to perform the initialization process.

Note

The initialization function of the processor board must be called before the DS2301 board's initialization function.

ds2301_init

Syntax

```
void ds2301_init(phs_addr_t base)
```

Include file

`ds2301.h`

Purpose

To initialize the DS2301 board.

Note

This function must be called before any other DS2301 function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

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	ds2301_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.
-163	Error	ds2301_init(0x??): Board not found!	No DS2301 board could be found at the specified PHS-bus address. Check if the DSxxx_n_BASE macro corresponds to the I/O board used.
-507	Error	ds2301_init(0x??): Memory allocation error!	The allocation of some dynamic memory for internal data storage has failed.

Example

This example shows how to use the function:

```
void main(void)
{
   init();
   ds2301_init(DS2301_1_BASE);
   ...
}
```

The DS2301 board at address DS2301_1_BASE is initialized.

Related topics

References

Base Address of the I/O Board 7
Macros 7

I/O Functions

Introduction

The C functions allow you to exchange data between your main application and an application running on one of the six DSPs. The communication is established via the dual-ported memories of the DS2301 board.

Where to go from here

Information in this section

ds2301_get_board_type

Syntax

```
void ds2301_get_board_type(
   phs_addr_t base,
   long *board_type)
```

Include file

`ds2301.h`

Purpose

To detect the type of the DS2301 board.

Description

The `board_type` parameter returns the type of the DS2301 board specified by the `base` parameter.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

board_type Specifies the pointer to the type number of the detected DS2301 board:

Symbol	Meaning
`230101`	DS2301 board with 50 MHz DSP
`230102`	DS2301 board with 60 MHz DSP

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   UInt32 type
   ...
   ds2301_get_board_type(DS2301_1_BASE, &type);
   ...
}
```

The type of the DS2301 board at the PHS-bus address DS2301_1_BASE is read and stored in the `type` variable.

Related topics

References

Base Address of the I/O Board 7
ds2301_init 9
Macros 7

ds2301_INT3

Syntax

```
void ds2301_INT3(
   phs_addr_t base,
   int channel,
   UInt32 value)
```

Include file

`ds2301.h`

Purpose

To request the DS2301 INT3 interrupt.

Description

The `value` parameter is written to the dual-port memory location 0x0FFF (address 0x1FFF as seen by the DSP) of the DS2301 channels specified by the `base` and `channel` parameters. This requests an interrupt INT3 on the specified DSPs. The dual-port memory location 0x0FFF is reserved for INT3 interrupts and cannot be used for other purposes.

The channels to be written are selected by a 6-bit mask. This allows writing to multiple channels with a single write operation. Use the channel definitions from the `ds2301.h` header file or a combination of them.

The data value being written to the dual-port memory location 0x0FFF can be chosen arbitrarily. It can be read by the respective DSP(s) and may be used to implement an interrupt-driven data transfer.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channels:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6
`DS2301_ALL_CH`	For all 6 channels

The definitions can be combined using the logical OR operation.

value Specifies the data value to be written to DS2301 dual-port memory location 0x0FFF.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   ...
   ds2301_INT3(DS2301_1_BASE, DS2301_CH1 | DS2301_CH2, 33);
   ...
}
```

An interrupt INT3 is requested on the first two channels of the DS2301 by writing 33 to the dual-port memory location 0x0FFF of channels 1 and 2.

Related topics

References

Base Address of the I/O Board....7
ds2301_init....9
Macros....7

ds2301_read

Syntax

```
void ds2301_read(
   phs_addr_t base,
   int channel,
   long offs,
   void *value)
```

Include file

`ds2301.h`

Purpose

To read data from the DS2301's dual-port memory.

Description

The 32-bit `value` parameter returns the contents of the DS2301's dual-port memory location specified by the `base`, `channel` and `offs` parameters.

Note

- The `channel` parameter must be specified as a 6-bit mask. It is recommended to use the channel definitions from the `ds2301.h` header file. Only a single channel can be read at a time.
- The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channel:

Symbol	Meaning
DS2301_CH1	For channel 1
DS2301_CH2	For channel 2
DS2301_CH3	For channel 3
DS2301_CH4	For channel 4
DS2301_CH5	For channel 5
DS2301_CH6	For channel 6

offs Specifies the dual-port memory source address within the range 0x0000 … 0x0FFF.

value Specifies the pointer to the memory location.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   long count;
   ...
   ds2301_read(DS2301_1_BASE, DS2301_CH1, 0, &count);
   ...
}
```

The contents of the dual-port memory location at offset 0 is read and written to the `count` variable.

Related topics

References

ds2301_read_block

Syntax

```
void ds2301_read_block(
   phs_addr_t base,
   int channel,
   long offs,
   long count,
   void *data)
```

Include file

`ds2301.h`

Purpose

To read a data block from the DS2301's dual-port memory.

Description

A block of 32-bit words is read from the DS2301's dual-port memory specified by `base` and `channel` and starting at the `offs` address. The `data` pointer must point to the destination data block where the data is to be stored. The destination data block must be allocated by the calling program.

Note

- The `channel` parameter must be specified as a 6-bit mask. It is recommended to use the channel definitions from the `ds2301.h` header file. Only a single channel can be read at a time.
- The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channel:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6

offs Specifies the dual-port memory start address within the range of 0x0000 … 0x0FFF.

count Specifies the number of data words to be read.

data Specifies the pointer to the source data array.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   float x[8];
   ...
   DS2301_read_block(DS2301_1_BASE, DS2301_CH2, 0, 8, x);
   ...
}
```

The contents of the dual-port memory of channel 1 from address 0x000 to 0x007 are read and written to the array x[].

Related topics

References

ds2301_read_block_float

Syntax

```
void ds2301_read_block_float(
   phs_addr_t base,
   int channel,
   long offs,
   long count,
   dsfloat *data)
```

Include file

`ds2301.h`

Purpose

To read a data block of float type from the DS2301's dual-port memory.

Description

A block of count 32-bit words is read from the DS2301's dual-port memory specified by `base` and `channel` and starting at the `offs` address. The `data` pointer must point to the destination data block where the data is to be stored. The destination data block must be allocated by the calling program.

Implicitly, the function converts the value from TI to IEEE754 floating-point format.

Note

- The `channel` parameter must be specified as a 6-bit mask. It is recommended to use the channel definitions from the `ds2301.h` header file. Only a single channel can be read at a time.
- The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channel:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6

offs Specifies the dual-port memory start address within the range of 0x0000 … 0x0FFF.

count Specifies the number of data words to be read.

data Specifies the pointer to the source data array.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   dsfloat x[8];
   ...
   DS2301_read_block(DS2301_1_BASE, DS2301_CH2, 0, 8, x);
   ...
}
```

The contents of the dual-port memory of channel 1 from address 0x000 to 0x007 is read and written to the array x[].

Related topics

References

Base Address of the I/O Board....7
ds2301_init....9
ds2301_read_block....18
Macros....7

ds2301_read_custom_register

Syntax

```
void ds2301_read_custom_register(
   phs_addr_t base,
   long *value)
```

Include file

`ds2301.h`

Purpose

To read the custom bit field in the interrupt select register (ISR) of the DS2301 board.

Description

The `value` parameter returns the contents of the 14-bit custom bit field in the DS2301's interrupt select register (ISR), i.e. bits D0 … D13. Bits D14 … D31 are read as zeros.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the pointer to the custom bit field in the ISR register within the range 0x0000 … 0x3FFF.

Return value

None

Example

For an example, refer to ds2301_write_custom_register on page 33.

Related topics

References

ds2301_read_float

Syntax

```
void ds2301_read_float(
   phs_addr_t base,
   int channel,
   long offs,
   dsfloat *value)
```

Include file

`ds2301.h`

Purpose

To read data of float type from the DS2301's dual-port memory.

Description

The `value` parameter returns the contents of the DS2301's dual-port memory location specified by the `base`, `channel` and `offs` parameters.

Implicitly, the function converts the value from TI to IEEE754 floating-point format.

Note

- The `channel` parameter must be specified as a 6-bit mask. It is recommended to use the channel definitions from the header file. Only a single channel can be read at a time.
- The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channel:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6

offs Specifies the dual-port memory source address within the range 0x0000 … 0x0FFF.

value Specifies the pointer to the memory location.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   dsfloat count;
   ...
   ds2301_read(DS2301_1_BASE, DS2301_CH1, 0, &count);
   ...
}
```

The contents of the dual-port memory location at offset 0 are read and written to the `count` variable.

Related topics

References

ds2301_read_isr

Syntax

```
void ds2301_read_isr(
   phs_addr_t base,
   UInt32 *value)
```

Include file

`ds2301.h`

Purpose

To read the interrupt select register (ISR) of the DS2301 board.

Description

The `value` parameter returns the contents of the 18-bit interrupt select field in the DS2301's interrupt select register (ISR), i.e. bits D14 … D31. Bits D0 … D13 are read as zeros.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the pointer to the ISR register within the range 0x00000000 … 0xFFFC0000.

Return value

None

Example

For an example, refer to ds2301_write_isr on page 36.

Related topics

References

ds2301_read_osr

Syntax

```
void ds2301_read_osr(
   phs_addr_t base,
   long value)
```

Include file

`ds2301.h`

Purpose

To read the output summing register (OSR) of the DS2301 board.

Description

The `value` parameter returns the contents of the DS2301's output summing register (OSR). Bits D30 and D31 are read as zeros.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the contents of the OSR register within the range 0x00000000 … 0x3FFFFFFF.

Return value

None

Example

For an example, refer to ds2301_write_osr on page 37.

Related topics

References

ds2301_speedchk

Syntax

```
void ds2301_speedchk(
   phs_addr_t base,
   dsfloat *exec_min,
   dsfloat *exec_max)
```

Include file

`ds2301.h`

Purpose

To get the execution time of the DS2301 timer interrupt service routine.

Description

The minimum and maximum execution time values of the 6 DS2301 board channels are stored in arrays pointed to by `exec_min` and `exec_max`.

The applications of the DS2301 board channels to be measured must execute the `speed_check()` macro.

The execution time values of unused channels or channels which are not executing the `speed_check()` macro are set to 0.0.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

exec_min Specifies the pointer to the array for the minimum execution times of all 6 DS2301 board channels (in µs).

exec_max Specifies the pointer to the array for the minimum execution times of all 6 DS2301 board channels (in µs).

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   Float32 exec_min[6];
   Float32 exec_max[6];
   ...
   ds2301_speedchk(DS2301_1_BASE, exec_min, exec_max);
   ...
}
```

The execution times of the 6 DS2301 channels timer interrupt service routines are read. The minimum execution times are stored in the `exec_min` array and the maximum execution times are stored in the `exec_max` array.

Array element 0 contains the value of channel 1, element 1 contains the value of channel 2 and so on.

Related topics

References

Base Address of the I/O Board....7
ds2301_init....9
Macros....7

ds2301_write

Syntax

```
void ds2301_write(
   phs_addr_t base,
   int channel,
   long offs,
   void *value)
```

Include file

`ds2301.h`

Purpose

To write data to the DS2301's dual-port memory.

Description

The 32-bit `value` parameter is written to the DS2301's dual-port memory location specified by the `base`, `channel`, and `offs` parameters.

The channels to be written are selected by a 6-bit mask. This allows writing to multiple channels with a single write operation. You may use the channel definitions from the `ds2301.h` header file.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channels:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6
`DS2301_ALL_CH`	For all 6 channels

The definitions can be combined using the logical OR operation.

offs Specifies the dual-port memory source address within the range 0x0000 … 0x0FFF.

value Specifies the data value to be written to the dual-port memory (can be either float or long).

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   float freq = 100.5;
   ...
   ds2301_write(DS2301_1_BASE, DS2301_CH1, 0x120, &freq);
   ...
}
```

The contents of the `freq` variable are written to the dual-port memory location at offset 0x120.

Related topics

References

Base Address of the I/O Board 7
ds2301_init 9
ds2301_write_block 29
ds2301_write_float 34
Macros 7

ds2301_write_block

Syntax

```
void ds2301_write_block(
   phs_addr_t base,
   int channel,
   long offs,
   long count,
   void *data)
```

Include file

`ds2301.h`

Purpose

To write a data block to the DS2301's dual-port memory.

Description

A block of count 32-bit words is written to the DS2301's dual-port memory specified by `base` and `channel` and starting at the `offs` address. The `data` pointer must point to the source data block.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channels:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6
`DS2301_ALL_CH`	For all 6 channels

The definitions can be combined using the logical OR operation.

offs Specifies the dual-port memory start address within the range of 0x0000 … 0x0FFF.

count Specifies the number of data words to be written.

data Specifies the data value to be written to the dual-port memory (can be either float or long).

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   float x[8] = {1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0};
```

```
   ...
   DS2301_write_block(DS2301_1_BASE,
   DS2301_CH1 | DS2301_CH2 | DS2301_CH3, 0, 8, x);
   ...
}
```

The contents of the **x** array is written to the dual-port memory of channels 1, 2, and 3 starting at address 0x0000.

Related topics

References

ds2301_write_block_float

Syntax

```
void ds2301_write_block_float(
   phs_addr_t base,
   int channel,
   long offs,
   long count,
   dsfloat *data)
```

Include file

`ds2301.h`

Purpose

To write a data block of float type to the DS2301's dual-port memory.

Description

A block of count 32-bit words is written to the DS2301's dual-port memory specified by **base** and **channel** and starting at the **offs** address. The pointer data must point to the source data block.

Implicitly, the function converts the value from IEEE754 to TI floating-point format.

Note

The **ds2301_init** function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channels:

Symbol	Meaning
DS2301_CH1	For channel 1
DS2301_CH2	For channel 2
DS2301_CH3	For channel 3
DS2301_CH4	For channel 4
DS2301_CH5	For channel 5
DS2301_CH6	For channel 6
DS2301_ALL_CH	For all 6 channels

The definitions can be combined using the logical OR operation.

offs Specifies the dual-port memory start address within the range of 0x0000 … 0x0FFF.

count Specifies the number of data words to be written.

data Specifies the data value to be written to the dual-port memory (can be either float or long).

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   float x[8] = {1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0};
   DS2301_write_block(DS2301_1_BASE,
   DS2301_CH2 | DS2301_CH3, 0, 8, x);
...
}
```

The contents of the `x` array are written to the dual-port memory of channels 2 and 3 starting at address 0x0000.

Related topics

References

ds2301_write_custom_register

Syntax

```
void ds2301_write_custom_register(
   phs_addr_t base,
   UInt16 value)
```

Include file

`ds2301.h`

Purpose

To write to the custom bit field in the interrupt select register (ISR) of the DS2301 board.

Description

The `value` parameter is written to the 14-bit custom bit field in the DS2301's interrupt select register (ISR), i.e. bits D0 … D13. Bits D14 … D31 remain unchanged.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the value to be written to the custom bit field in the interrupt select register (ISR) of the DS2301 board within the range 0x0000 … 0x3FFF.

Return value

None

Example

This example shows how to use this function:

```
#define ISR1_MASK 0x0001C000
void sub_fct()
{
   UInt16 cust;
   ...
   ds2301_read_custom_register(DS2301_1_BASE, &cust);
   ds2301_write_custom_register(DS2301_1_BASE, cust | 0x04);
   ...
}
```

Bit 2 in the custom bit field of the ISR register is set without changing the remaining bits.

Related topics

References

ds2301_write_float

Syntax

```
void ds2301_write_float(
   phs_addr_t base,
   int channel,
   long offs,
   dsfloat *value)
```

Include file

`ds2301.h`

Purpose

To write data of float type to the DS2301's dual-port memory.

Description

The `value` parameter is written to the DS2301's dual-port memory location specified by the `base`, `channel`, and `offs` parameters.

Implicitly, the function converts the value from IEEE754 to TI floating-point format.

The channels to be written are selected by a 6-bit mask. This allows writing to multiple channels with a single write operation. You may use the channel definitions from the `ds2301.h` header file.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

channel Specifies the bit mask with selected channels:

Symbol	Meaning
`DS2301_CH1`	For channel 1
`DS2301_CH2`	For channel 2
`DS2301_CH3`	For channel 3
`DS2301_CH4`	For channel 4
`DS2301_CH5`	For channel 5
`DS2301_CH6`	For channel 6
`DS2301_ALL_CH`	For all 6 channels

The definitions can be combined using the logical OR operation.

offs Specifies the dual-port memory source address within the range 0x0000 … 0x0FFF.

value Specifies the data value to be written into dual-port memory.

Return value

None

Example

This example shows how to use this function:

```
void sub_fct()
{
   dsfloat freq = 100.5;
   ...
   ds2301_write(DS2301_1_BASE, DS2301_CH1, 0x120, &freq);
   ...
}
```

The contents of the `freq` variable are written to the dual-port memory location at offset 0x120.

Related topics

References

ds2301_write_isr

Syntax

```
void ds2301_write_isr(
   phs_addr_t base,
   UInt32 value)
```

Include file

`ds2301.h`

Purpose

To write data to the DS2301's interrupt select register (ISR).

Description

The `value` parameter is written to the 18-bit interrupt select field in the DS2301's interrupt select register (ISR), i.e. bits D14 … D31. Bits D0 … D13 remain unchanged.

Note

- The `ds2301_init` function must be called before this function can be used.
- The `ds2301_write_isr` function does not shift the bits to be written into position. They must be placed to the proper positions by the user.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the data value to be written to the ISR register within the range 0x00000000 … 0xFFFC0000.

Return value

None

Example

This example shows how to use this function:

```
#define ISR1_MASK 0x0001C000
void sub_fct()
{
   UInt32 isr;
   ...
   ds2301_read_isr(DS2301_1_BASE, &isr);
   isr &= ~ISR1_MASK;
   isr |= (0x04 << 14);
   ds2301_write_isr(DS2301_1_BASE, isr);
...
}
```

The current status of the interrupt select register is read. The 3 bits of the channel 1 interrupt select field are set to the new value 0x04 without changing the interrupt select fields of the remaining channels. This selects the XF0 line of channel 5 as the interrupt source for the INT1 interrupt of channel 1.

Related topics

References

ds2301_write_osr

Syntax

```
void ds2301_write_osr(
   phs_addr_t base,
   UInt32 value)
```

Include file

`ds2301.h`

Purpose

To write data to the DS2301's output summing register (OSR).

Description

The `value` parameter is written to the DS2301's output summing register (OSR). Bits D30 and D31 remain unchanged.

Note

The `ds2301_init` function must be called before this function can be used.

I/O mapping

For details on the I/O connector pinouts of the DS2301, refer to *C:\Program Files <x86>\Common Files\ dSPACE\Help <ReleaseVersion>\ Print\DS2301HardwareReference.pdf*.

Parameters

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

value Specifies the data value to be written to the OSR register within the range 0x00000000 … 0x3FFFFFFF.

Return value

None

Example

This example shows how to use this function:

```
#define OSR1_MASK 0x0000001F
void sub_fct()
{
   UInt32 osr;
   ...
   ds2301_read_osr(DS2301_1_BASE, &osr);
   osr &= ~OSR1_MASK;
   osr |= 0x05;
   ds2301_write_osr(DS2301_1_BASE, osr);
   ...
}
```

The current status of the output summing register is read. The 5 bits of the channel 1 output summing select field are set to the new value 0x05 without changing the output summing select fields of the remaining channels. This selects the output signals of channel 2 and channel 4 to be added to the channel 1 output signal.

Related topics

References

Base Address of the I/O Board....7
ds2301_init....9
ds2301_write_isr....36
Macros....7

B

C

D

L