DS5101 Digital Waveform Output Board

Features

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 Document

Contents

This document provides feature-oriented access to the information you need to implement the functions of the DS5101.

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.

Introduction to the Features of the DS5101

Where to go from here

Information in this section

Information in other sections

Data Sheets (PHS Bus System Hardware Reference 🕮)
Summarizes the technical specifications of the hardware components.

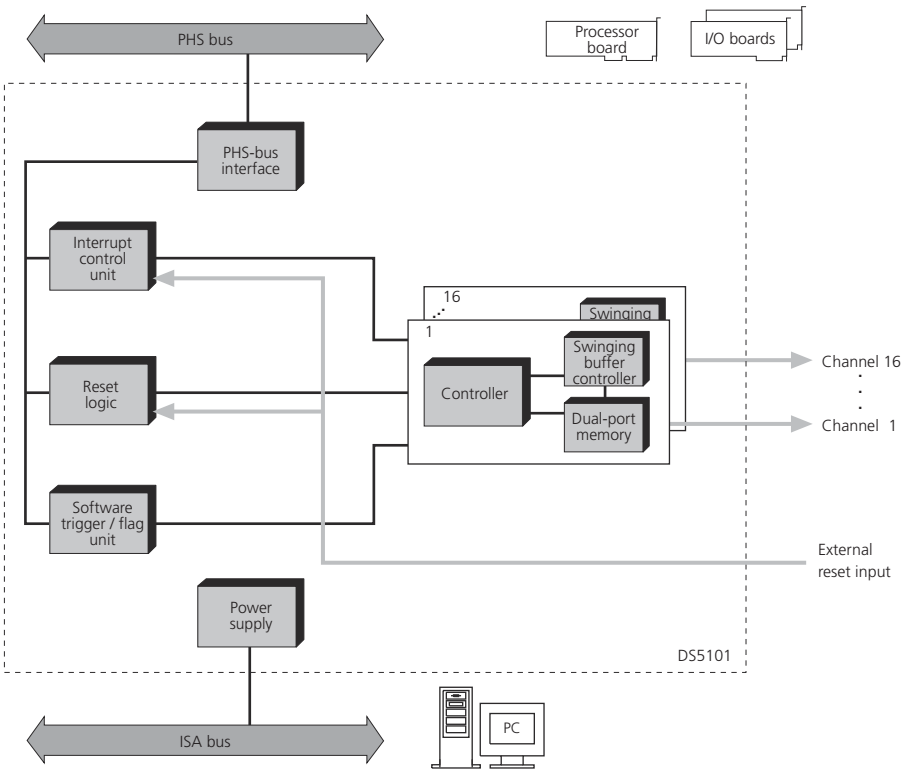
DS5101 Architecture

Introduction

All board revisions of the DS5101 have the same functional units but the board architectures are different.

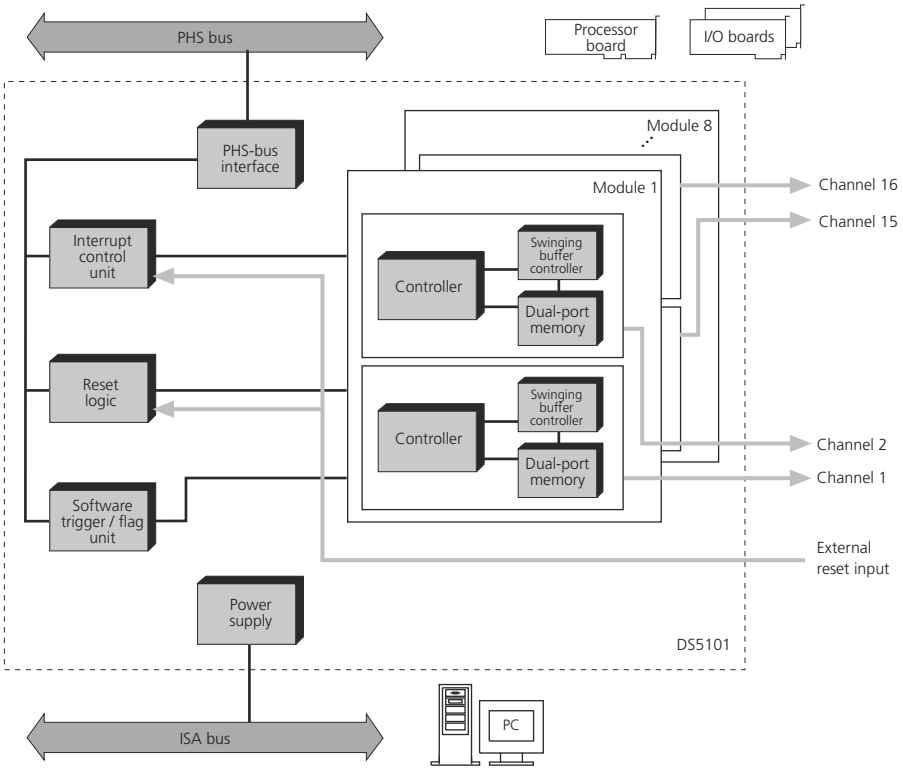
As of revision DS5101-04

The following illustration gives an overview of the functional units and architecture of the DS5101 as of revision DS5101-04:



Up to revision DS5101-02

The following illustration gives an overview of the functional units and architecture of the DS5101 up to revision DS5101-02:



Related topics

References

DS5101 Interfaces 10
Feature Overview 9

Feature Overview

Introduction

The DS5101 provides a timing I/O unit and interrupt control.

Overview

The DS5101 Digital Waveform Output Board has been designed to generate complex, high-speed digital signals at high resolution. The board can generate a multitude of signals at various frequencies, including incremental encoder signals and pulse-width modulation waveforms. It is able to vary signal pulse widths on the fly and, through the use of various trigger and interrupt mechanisms, provides a high degree of flexibility. Its main area of deployment is

hardware-in-the-loop simulation in automotive applications, for example, simulating sensors or controlling actuators.

Timing I/O unit

The timing I/O unit generates complex digital signals at TTL levels on up to 16 channels. Refer to Timing I/O Unit on page 13.

Interrupt control

The interrupt control provides various hardware interrupts to the processor board. Refer to Interrupts Provided by the DS5101 on page 43.

Limitations

There are some limitations when you work with the DS5101. Refer to Limitations of the DS5101 on page 47.

Related topics

References

DS5101 Interfaces

Introduction

The DS5101 has interfaces for connection to a PHS-bus-based system and external devices.

Integration into a PHS-bus-based system

To be used, the DS5101 must be integrated into a PHS-bus-based system. While the DS5101 carries out the generation of output signals, the processor board executes the real-time model. That is, applications using DS5101 I/O features are implemented on the processor board.

Communication between the processor board and I/O boards is performed via the peripheral high-speed bus: That is the PHS bus for a connection to a dSPACE processor board.

Partitioning the PHS bus with the DS802 With the DS802 PHS Link Board you can spatially partition the PHS bus by arranging the I/O boards in several expansion boxes.

The DS802 can be used in combination with many types of available dSPACE I/O boards. However, some I/O boards and some functionalities of specific I/O boards are not supported.

The I/O board support depends on the dSPACE software release which you use. For a list of supported I/O boards, refer to DS802 Data Sheet (PHS Bus System Hardware Reference 🕮).

Connection to external devices

There are two ways of connecting external devices to the DS5101. To access the I/O unit of the DS5101, connect external devices to

- the 37-pin, female Sub-D I/O connector P35 on the DS5101
- the BNC connectors on the optional connector panel CP5101

Related topics

References

DS5101 Architecture 7
Feature Overview 9

Timing I/O Unit

Introduction

The timing I/O unit is capable of generating fast and complex digital waveforms on up to 16 channels.

Where to go from here

Information in this section

Information in other sections

Basics

Basics of the Timing I/O Unit

Board overview

The DS5101 up to revision DS5101-02 contains up to eight output modules (DS5101M). Each module provides two channels for signal generation. The DS5101 as of revision DS5101-04 contains one FPGA providing 16 channels. Each channel consists of a channel controller, a swinging buffer controller, a local RAM, and a counter. The counter provides the time base with a resolution of 25 ns.

Timing I/O unit

The timing I/O unit of the DS5101 can be accessed by some standard DWO applications which are distributed with the DS5101 software environment. They are supported by suitable functions for application loading, i.e., the DWO Compiler generated load functions, and individual functions for parameter control and parameter readout. Using these functions you have more functionality than using the functions generated automatically by the DWO compiler.

DWO applications

To use a channel for signal generation, a DWO application (digital waveform output) to be executed by the channel controller must be downloaded to the local RAM. Downloading is performed by the processor board.

Each signal can be described as a series of states – each consisting of a delay and a level change, like "after 1 ms set output level to high". A DWO application defines actions and delays including loops and conditional branches. Examples of actions are setting the channel to high level, resetting to low level, generating an interrupt, and triggering other channels. A delay defines the time after which the next action will be executed.

Standard DWO applications

For the generation of the standard pulse patterns such as PWM or monoflop signals, ready-to-use DWO applications are provided. The following standard DWO applications are available:

- 1-phase PWM signal generation
- 3-phase PWM signal generation
- 3-phase PWM signal generation with inverted and non-inverted outputs
- Incremental encoder simulation
- Monoflop signal generation.

For more details, refer to Standard DWO Applications on page 17.

Downloading is done automatically if you use RTI or by calling special loader functions, when using RTLib.

Custom DWO applications

For the generation of arbitrary pulse patterns, you can program your own DWO applications. See Custom DWO Applications on page 38. You can use RTI to download the DWO application automatically or use RTLib and call special loader functions.

Triggering

The DS5101 provides different trigger sources:

- You can use the `ds5101_trigger` function to send a software trigger to one or more output channels.
- Each channel can be configured as an input channel for trigger signals. For custom DWO applications, you can use this feature to receive an external trigger on one input channel, which subsequently sends a software trigger to the other output channels. You can use the output signal of another output channel as trigger input, for example, to synchronize several DS5101 boards.

  For custom DWO applications, the DWO commands FALL, RISE, BOTH, or NONE are available to switch to the input mode. See also Recognizing External Timing I/O Signals (PHS Bus System Hardware Reference 🕮).

Power-up state

On power-up or reset of the DS5101, the output modules are reset, the channels are set to input, and the pull-up resistors of all channels are connected to V_{CC}. Note that a connected device with a high input impedance (> 1 kΩ) will interpret the power-up state as high level.

Pull-up resistors

By default, the pull-up resistors are connected to V_{CC}. With the `ds5101_set_pullup` function, you can connect the pull-up resistors to GND. On reset, the pull-up resistors are connected to V_{CC} again.

Using RTI, you have to incorporate the RTLib function in a Simulink S-function. For details, refer to Implementing S-Functions (RTI and RTI-MP Implementation Guide 🕮) and I/O Circuits and Electrical Characteristics (PHS Bus System Hardware Reference 🕮).

For DS5101 as of revision DS5101-04 you can set the pull-up resistor for each channel individually via jumpers. Refer to Board Overview (as of Revision DS5101-04) (PHS Bus System Hardware Reference 🕮).

Start behavior

After a DWO application is loaded, you can start signal generation. For an initial duration of 200 ns, the corresponding channels are set to output and the voltage is set to low (unless the channel is set to input mode waiting for a trigger). This is done automatically if you use RTI or by executing the `ds5101_start` function when using RTLib.

Related topics

Basics

Generation of Arbitrary Signals 38
Timing I/O Unit 13

References

DS5101 Digital Waveform Output Board (PHS Bus System Hardware Reference 🕮)
Limitations of the DS5101 47

Standard DWO Applications

Where to go from here

Information in this section

1-Phase PWM Signal Generation (PWM1)

Introduction

The timing I/O unit of the DS5101 provides generation of 1-phase PWM signals with run-time adjustable PWM period and duty cycle on up to 16 channels. The channels are not synchronized.

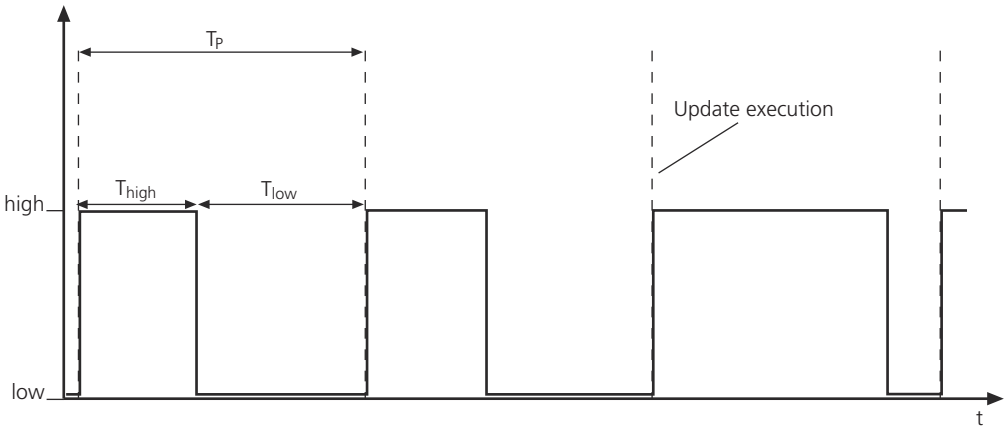
Note

When PWM signal generation starts, the initial output signal level is low for a duration of 200 ns. Thereafter, the PWM period starts with the high PWM level.

PWM period and duty cycle

You can specify the PWM period T_P (= T_{high} + T_{low}) for each 1-phase PWM signal individually within the range 500 ns … 26 s.

You can also specify the duty cycle d (= T_{high} / T_P) for each 1-phase PWM signal individually within the range 0 … 1 (0 … 100%).



If the PWM period or duty cycle is changed during run time, the new values become effective with the next PWM period, beginning with the high PWM level.

In the illustration above, an update of the duty cycle is executed at the beginning of the third PWM period.

Note

Due to quantization effects, you will encounter considerable deviations between the desired PWM period T_p and the generated PWM period, especially for high PWM frequencies. Refer to Quantization Effects on page 48.

Downloading DWO application

When using 1-phase PWM signal generation, the corresponding DWO application must be downloaded to the channel controller before you start signal generation. This is done automatically if you use RTI or by executing the loader function `ds5101_pwm_load`, when using RTLib.

RTI/RTLib support

To perform 1-phase PWM signal generation, use DS5101 Blockset and RTLib. Refer to

- RTI: DS5101PWM1_Bx_BLy (DS5101 RTI Reference 🕮).
- RTLib: 1-Phase PWM Signal Generation (DS5101 RTLib Reference 🕮).

Execution times

For execution times of the RTLib functions, refer to Function Execution Times (DS5101 RTLib Reference 🕮).

Connecting external devices

For an I/O circuit and information on electrical characteristics and signal conditioning of the timing I/O unit, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and the RTLib functions, and the corresponding pins used for 1-phase PWM signal generation.

Related RTI Block	Channel (RTI)	Related RTLib Functions	Channel (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101PWM1_Bx_BLy	Ch-block 1 (Channel 1)	See 1-Phase PWM Signal Generation	Block 1 (via mask)	P35 20	CP1	IO1
	Ch-block 2 (Channel 2)		Block 2 (via mask)	P35 21	CP2	IO2
	Ch-block 3 (Channel 3)		Block 3 (via mask)	P35 22	CP3	IO3
	Ch-block 4 (Channel 4)		Block 4 (via mask)	P35 23	CP4	IO4
	Ch-block 5 (Channel 5)		Block 5 (via mask)	P35 24	CP5	IO5
	Ch-block 6 (Channel 6)		Block 6 (via mask)	P35 25	CP6	IO6
	Ch-block 7 (Channel 7)		Block 7 (via mask)	P35 26	CP7	IO7
	Ch-block 8 (Channel 8)		Block 8 (via mask)	P35 27	CP8	IO8
	Ch-block 9 (Channel 9)		Block 9 (via mask)	P35 28	CP9	IO9
	Ch-block 10 (Channel 10)		Block 10 (via mask)	P35 29	CP10	IO10
	Ch-block 11 (Channel 11)		Block 11 (via mask)	P35 30	CP11	IO11
	Ch-block 12 (Channel 11)		Block 12 (via mask)	P35 31	CP12	IO12
	Ch-block 13 (Channel 13)		Block 13 (via mask)	P35 32	CP13	IO13
	Ch-block 14 (Channel 14)		Block 14 (via mask)	P35 33	CP14	IO14
	Ch-block 15 (Channel 15)		Block 15 (via mask)	P35 34	CP15	IO15
	Ch-block 16 (Channel 16)		Block 16 (via mask)	P35 35	CP16	IO16

Conflicting features

The I/O features of the DS5101 conflict with each other. For details, refer to Limitations of the DS5101 on page 47.

Related topics

Basics

Basics of the Timing I/O Unit..... 14

References

1-Phase PWM Signal Generation (DS5101 RTLib Reference 🕮)
ds5101_pwm_load (DS5101 RTLib Reference 🕮)
DS5101PWM1_Bx_BLy (DS5101 RTI Reference 🕮)

3-Phase PWM Signal Generation (PWM3)

Introduction

The timing I/O unit of the DS5101 provides 3-phase PWM signals with run-time adjustable PWM period, duty cycles, and interrupt shift. PWM3 signals are centered around the middle of the PWM period. The polarity of the PWM3 signals is active high.

To synchronize the output signals on the three channels, a master clock signal is generated on an additional channel. Therefore, 3-phase PWM signal generation requires 4 channels and 2 output modules (DS5101M) respectively. With the DS5101 timing I/O unit, you can implement up to four independent 3-phase PWM signals at the same time.

Note

When 3-phase PWM signal generation starts, the three PWM3 phases are set to low for a duration of 200 ns. After that, the PWM signals start at the middle of the low PWM level.

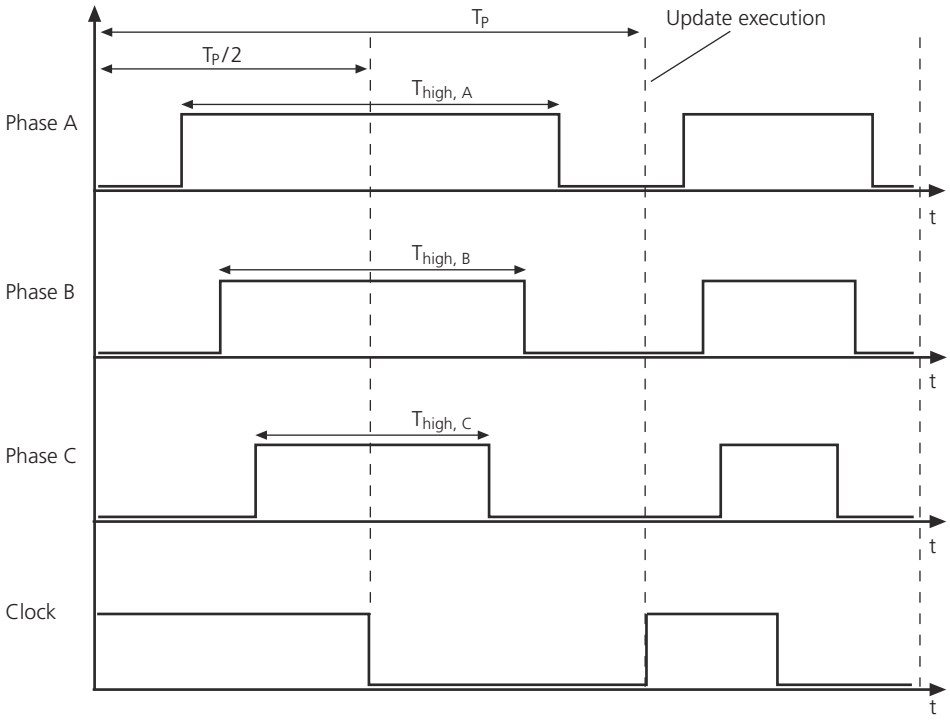
PWM period and duty cycle

For PWM3 signals, the PWM period T_P (= T_{high} + T_{low}) applies to each of the three PWM3 phases. The PWM period can be specified within the range 500 ns … 26 s.

For each of the three PWM3 phases, you can specify the duty cycle d_x (x = A, B, C) individually. The duty cycle is defined as $d_x = T_{high,x} / T_P$, where T_P is the PWM period. The available duty cycle range is 0 … 1 (0 … 100%).

If the PWM period T_P or the duty cycles d_x are changed during run time, the new values become effective synchronously for all the 3 phases with the next PWM period.

In the illustration, an update of the PWM period is executed at the beginning of the second PWM period.

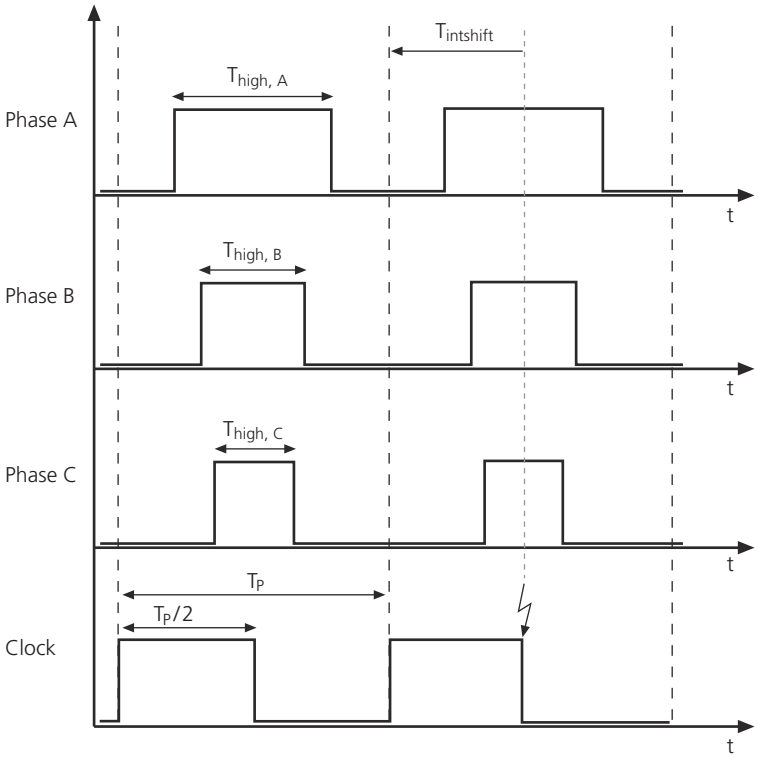


Note

Due to quantization effects, you will encounter considerable deviations between the desired PWM period T_P and the generated PWM period, especially for high PWM frequencies. Refer to Quantization Effects on page 48.

Interrupt via PWM3 signal generation

The clock channel generates an interrupt request at $0.5 \cdot T_p - T_{intshift}$ where the interrupt shift $T_{intshift}$ depends on the PWM period. It must be in the range 0.25 µs … (0.5 · T_p - 0.25 µs). See also Signal Generation Interrupt on page 43.



Downloading DWO application

When using 3-phase PWM signal generation, the corresponding DWO application must be downloaded to the channel controllers of the corresponding channels before you start signal generation. This is done automatically if you use RTI or by executing the loader function **`ds5101_pwm3_intshift_load`** when using RTLib.

RTI/RTLib support

To perform 3-phase PWM signal generation, use DS5101 Blockset and RTLib. For details, refer to

- RTI: DS5101PWM3_Bx_BLy (DS5101 RTI Reference 🕮).
- RTLib: 3-Phase PWM Signal Generation (DS5101 RTLib Reference 🕮).

Execution times

For execution times of the RTLib functions, refer to Function Execution Times (DS5101 RTLib Reference 🕮). There you will find details of the measurement setup and the results.

Connecting external devices

For an I/O circuit and information on electrical characteristics and signal conditioning of the timing I/O unit, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and the RTLib functions, and the corresponding pins used for 3-phase PWM signal generation.

Related RTI Block	Channel (RTI)	Related RTLib Functions	Channel (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101PWM3_Bx_BLy	Ch-block 1 (Channel 1: Phase A)	See 3-Phase PWM Signal Generation (DS5101 RTLib Reference 🕮)	Block 1 (via mask)	P35 20	CP1	IO1
	Ch-block 1 (Channel 2: Phase B)		Block 1 (via mask)	P35 21	CP2	IO2
	Ch-block 1 (Channel 3: Phase C)		Block 1 (via mask)	P35 22	CP3	IO3
	Ch-block 1 (Channel 4: Clock)		Block 1 (via mask)	P35 23	CP4	IO4
	Ch-block 2 (Channel 5: Phase A)		Block 2 (via mask)	P35 24	CP5	IO5
	Ch-block 2 (Channel 6: Phase B)		Block 2 (via mask)	P35 25	CP6	IO6
	Ch-block 2 (Channel 7: Phase C)		Block 2 (via mask)	P35 26	CP7	IO7
	Ch-block 2 (Channel 8: Clock)		Block 2 (via mask)	P35 27	CP8	IO8
	Ch-block 3 (Channel 9: Phase A)		Block 3 (via mask)	P35 28	CP9	IO9
	Ch-block 3 (Channel 10: Phase B)		Block 3 (via mask)	P35 29	CP10	IO10
	Ch-block 3 (Channel 11: Phase C)		Block 3 (via mask)	P35 30	CP11	IO11
	Ch-block 3 (Channel 12: Clock)		Block 3 (via mask)	P35 31	CP12	IO12
	Ch-block 4 (Channel 13: Phase A)		Block 4 (via mask)	P35 32	CP13	IO13
	Ch-block 4 (Channel 14: Phase B)		Block 4 (via mask)	P35 33	CP14	IO14
	Ch-block 4 (Channel 15: Phase C)		Block 4 (via mask)	P35 34	CP15	IO15
	Ch-block 4 (Channel 16: Clock)		Block 4 (via mask)	P35 35	CP16	IO16

Conflicting features

The I/O features of the DS5101 conflict with each other. For details, see Limitations of the DS5101 on page 47.

Related topics

Basics

Basics of the Timing I/O Unit..... 14

References

3-Phase PWM Signal Generation (DS5101 RTLib Reference 🕮)
ds5101_pwm3_intshift_load (DS5101 RTLib Reference 🕮)
DS5101PWM3_Bx_BLy (DS5101 RTI Reference 🕮)

3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (PWM6)

Introduction

The timing I/O unit of the DS5101 provides 3-phase/6-channel PWM signals with 3 inverted outputs (/A, /B, /C) and 3 non-inverted outputs (A, B, C). The PWM period, switch delay, interrupt shift, and the duty cycles of the non-inverted outputs can be changed at run time. The inverted and non-inverted signals are symmetrical to the middle of the PWM period.

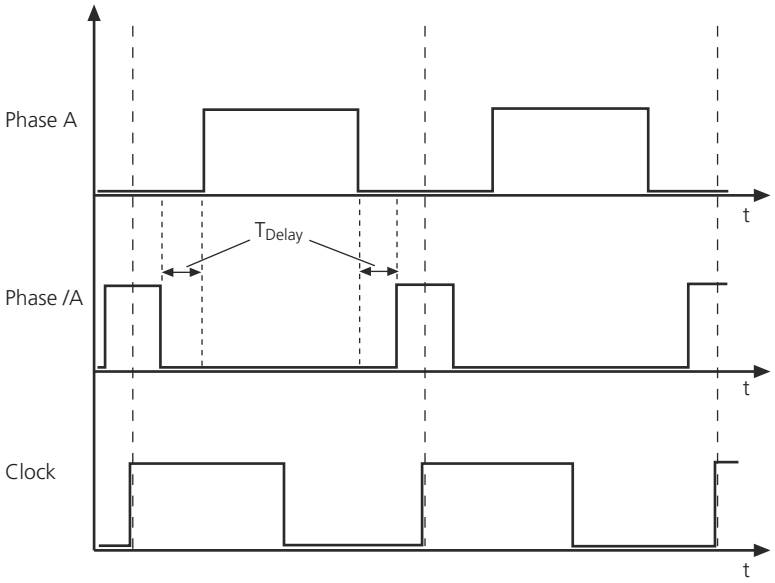
To synchronize the 6 output signals a master clock signal is generated on an additional channel. 3-phase/6-channel PWM signal generation requires 7 channels and 4 output modules (DS5101M) respectively. With the DS5101 timing I/O unit, you can generate up to 2 independent 3-phase/6-channel PWM signals at the same time. The channels 8 and 16 are not used (see also I/O mapping on page 28).

Note

When 3-phase/6-channel PWM signal generation starts, the output signals are set to low for a duration of 200 ns.

Switch delay

You can specify a switch delay T_{Delay} for the inverted outputs. With delay, the falling edge of the inverted signal leads the rising edge of the corresponding non-inverted signal. The rising edge of the inverted signal is delayed with respect to the falling edge of the non-inverted signal.



The switch delay is measured in seconds. The time required by the pulse of the inverted signal plus twice the delay must be shorter than the PWM period.

PWM period and duty cycle

The PWM period T_P (= T_{high} + T_{low}) applies to each of the inverted and non-inverted PWM3 phases. The PWM period can be specified within the range 500 ns … 26 s.

For each non-inverted PWM3 phase (A, B, C), you can specify the duty cycle d_x (x = A, B, C) individually. The duty cycle is defined as $d_x = T_{high,x} / T_P$, where T_P is the PWM period. The available duty cycle range is $0 < d < 1$.

Note

Generation of PWM signals with duty cycles $d_x = 0$ (non-inverted output is always low) and $d_x = 1$ (non-inverted output is always high) is not possible.

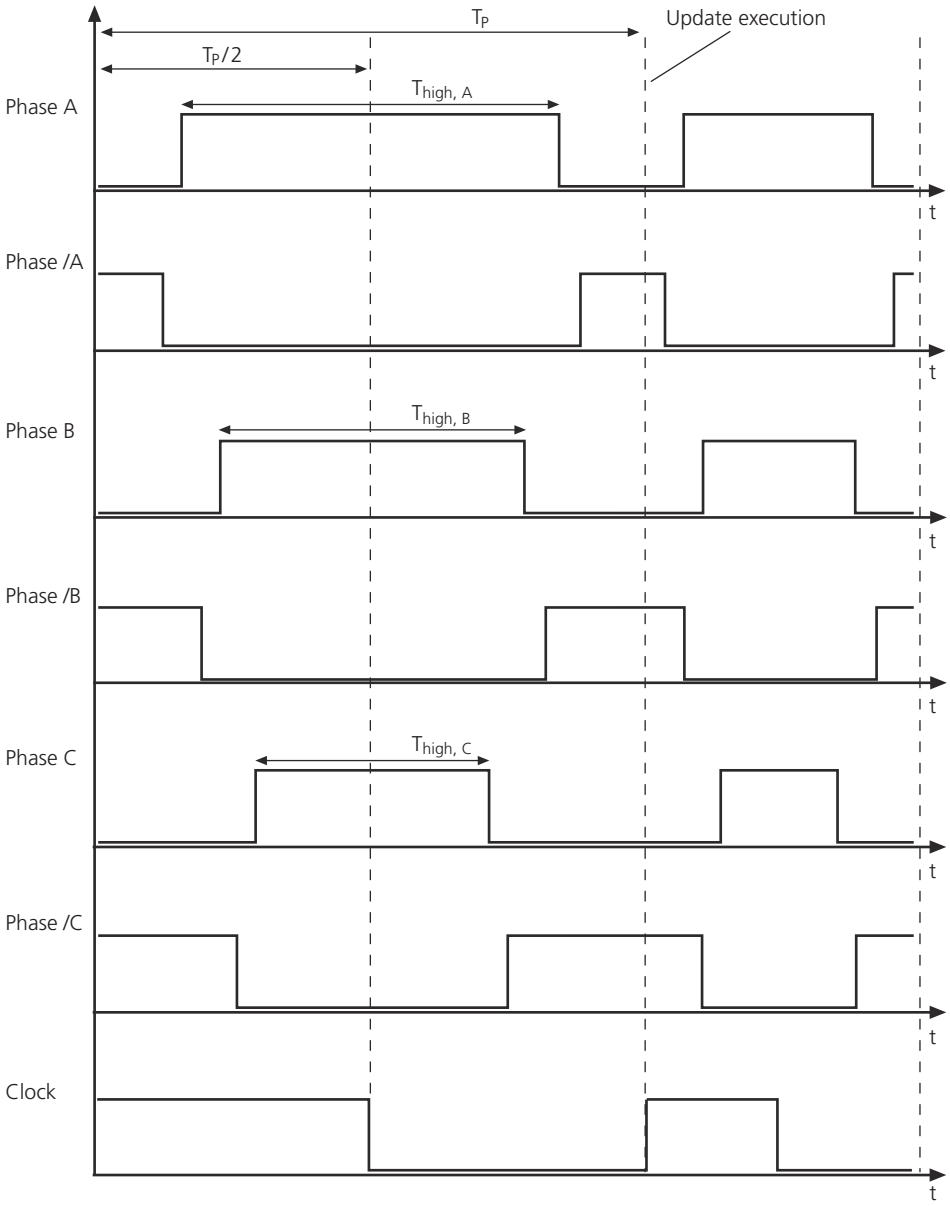
- For duty cycle values near or equal to 0, a remaining high pulse of 250 ns is generated at the non-inverted output.
- For duty cycle values near or equal to 1, a remaining high pulse of 800 ns is generated at the inverted output.

The actual duty cycle limits depend on the PWM period T_P and the switch delay T_{Delay}.

The duty cycles of the inverted PWM3 phases (/A, /B, /C) are derived from the duty cycles of the corresponding non-inverted PWM3 phases, taking into account the switch delay.

If the PWM period T_P, switch delay T_{Delay}, or duty cycles d_x are changed during run time, the new values become effective synchronously for all the 6 channels with the next PWM period.

In the illustration below, an update of the PWM period is executed at the beginning of the second PWM period.

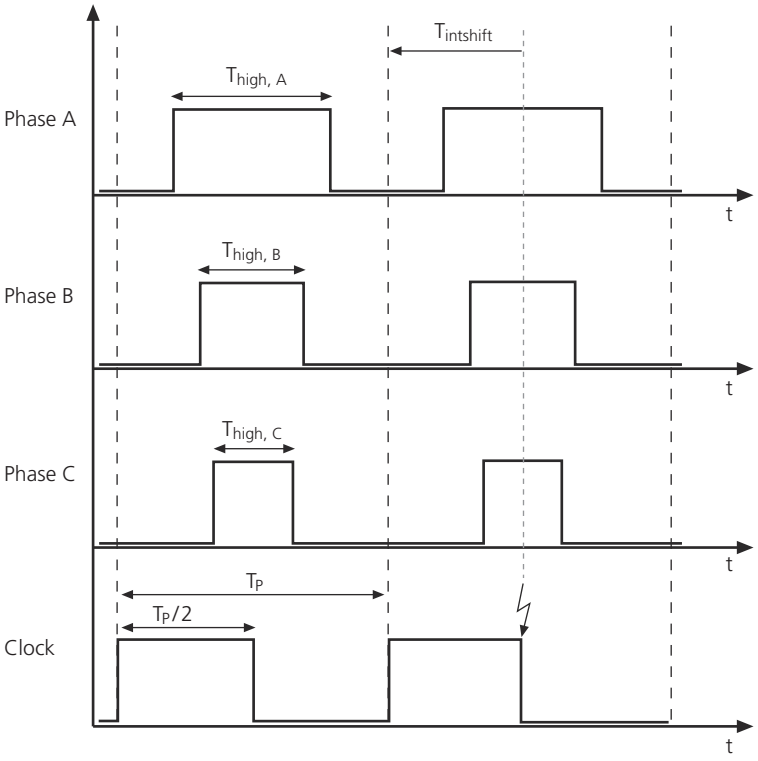


Note

Due to quantization effects, you will encounter considerable deviations between the desired PWM period T_P and the generated PWM period, especially for high PWM frequencies. Refer to Quantization Effects on page 48.

Interrupt via PWM6 signal generation

The clock channel generates an interrupt request at $0.5 \cdot T_p - T_{intshift}$ where the interrupt shift $T_{intshift}$ depends on the PWM period. It must be in the range 0.25 µs … ($0.5 \cdot T_p$ - 0.25 µs). The following illustration shows only the non-inverted phases. See also Signal Generation Interrupt on page 43.



Downloading DWO application

When using 3-phase/6-channel PWM signal generation with inverted and non-inverted outputs, the corresponding DWO application must be downloaded to the channel controllers of the corresponding channels before you start signal generation. This is done automatically if you use RTI or by executing the loader function `ds5101_pwm6_intshift_load`, when using RTLib.

RTI/RTLib support

To perform 3-phase PWM signal generation with inverted and non-inverted outputs, use DS5101 Blockset and RTLib. For details, refer to

- RTI: DS5101PWM6_Bx_BLy (DS5101 RTI Reference 🕮).
- RTLib: 3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (DS5101 RTLib Reference 🕮).

Execution times

For execution times of the RTLib functions, refer to Function Execution Times (DS5101 RTLib Reference 🕮). There you will find details of the measurement setup and the results.

Connecting external devices

For an I/O circuit and information on electrical characteristics and signal conditioning of the timing I/O unit, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and the RTLib functions, and the corresponding pins used for 3-phase/6-channel PWM signal generation with inverted and non-inverted outputs.

Related RTI Block	Channel (RTI)	Related RTLib Functions	Channel (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101PWM6_Bx_BLy	Ch-block 1 (Channel 1: Phase A)	See 3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (DS5101 RTLib Reference 🕮)	Block 1 (via mask)	P35 20	CP1	IO1
	Ch-block 1 (Channel 2: Phase /A)		Block 1 (via mask)	P35 21	CP2	IO2
	Ch-block 1 (Channel 3: Phase B)		Block 1 (via mask)	P35 22	CP3	IO3
	Ch-block 1 (Channel 4: Phase /B)		Block 1 (via mask)	P35 23	CP4	IO4
	Ch-block 1 (Channel 5: Phase C)		Block 1 (via mask)	P35 24	CP5	IO5
	Ch-block 1 (Channel 6: Phase /C)		Block 1 (via mask)	P35 25	CP6	IO6
	Ch-block 1 (Channel 7: Clock)		Block 1 (via mask)	P35 26	CP7	IO7
	Ch-block 2 (Channel 9: Phase A)		Block 2 (via mask)	P35 28	CP9	IO9
	Ch-block 2 (Channel 10: Phase /A)		Block 2 (via mask)	P35 29	CP10	IO10
	Ch-block 2 (Channel 11: Phase B)		Block 2 (via mask)	P35 30	CP11	IO11
	Ch-block 2 (Channel 12: Phase /B)		Block 2 (via mask)	P35 31	CP12	IO12
	Ch-block 2 (Channel 13: Phase C)		Block 2 (via mask)	P35 32	CP13	IO13
	Ch-block 2 (Channel 14: Phase /C)		Block 2 (via mask)	P35 33	CP14	IO14
	Ch-block 2 (Channel 15: Clock)		Block 2 (via mask)	P35 34	CP15	IO15

Conflicting features

The I/O features of the DS5101 conflict with each other. For details, see Limitations of the DS5101 on page 47.

Related topics

Basics

Basics of the Timing I/O Unit..... 14

References

3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (DS5101 RTLib Reference 🕮)
ds5101_pwm6_intshift_load (DS5101 RTLib Reference 🕮)
DS5101PWM6_Bx_BLy (DS5101 RTI Reference 🕮)

Incremental Encoder Simulation

Introduction

The timing I/O unit of the DS5101 provides incremental encoder simulation with two output signals (PHI0 and PHI90), which have a phase lag of +90° or –90° to one another. The velocity can be changed at run time.

The simulation of an incremental encoder requires 2 output channels (PHI0 and PHI90). With the DS5101 timing I/O unit, you can simulate up to 7 incremental encoders at the same time. The channels 15 and 16 are not used (see also I/O mapping on page 31).

Note

- The DS5101 does not simulate encoder index signals.
- When incremental encoder simulation starts, the output signals are set to low for a duration of 200 ns.

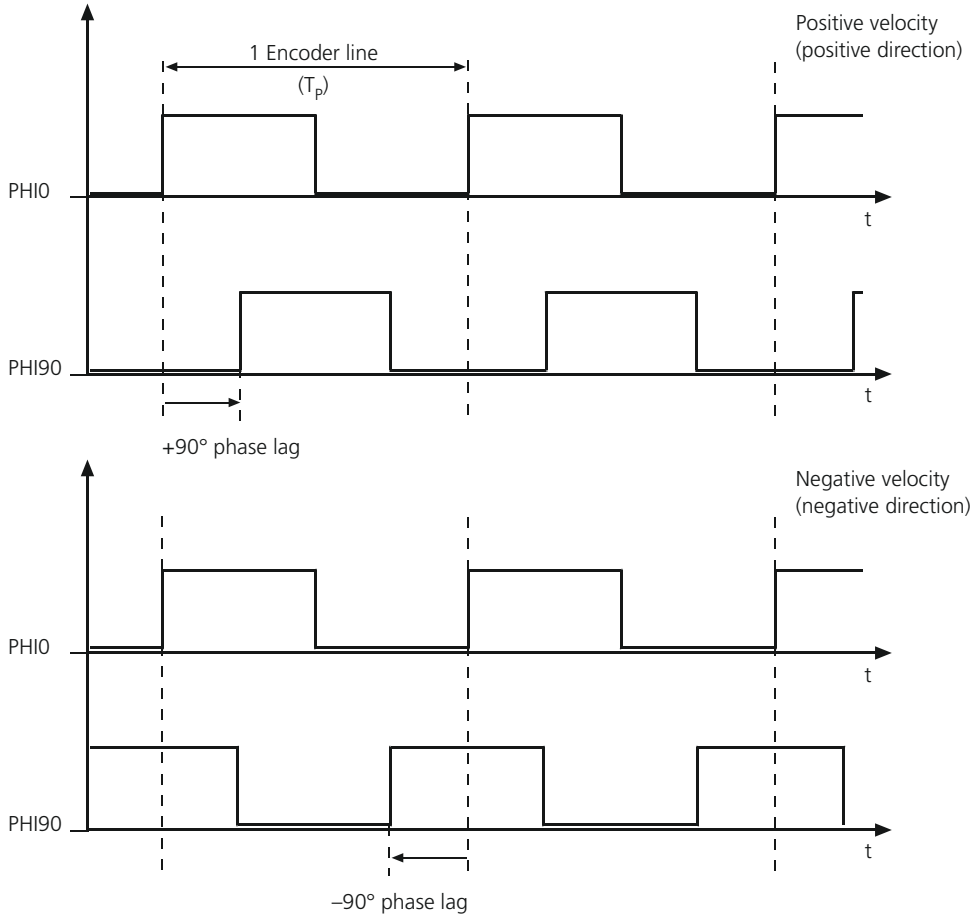
Velocity and direction

One encoder line is represented by one period (one pulse) of the output signals. The velocity can be specified within the range -1,000,000 … +1,000,000 encoder lines per second.

The direction is implicitly specified by the sign of the velocity value:

- For positive velocity values (positive direction), the PHI90 signal lags PHI0.

- For negative velocity values (negative direction), the PHI90 signal leads PHI0.



Downloading DWO application

When using incremental encoder simulation, the corresponding DWO application must be downloaded to the channel controllers of the concerned channels before you start signal generation. This is done automatically if you use RTI or by executing the loader function `ds5101_inc_load`, when using RTLib.

RTI/RTLib support

To perform incremental encoder simulation use DS5101 Blockset and RTLib. For details, see

- RTI: **DS5101INC_Bx_BLy** (DS5101 RTI Reference 🕮).
- RTLib: Incremental Encoder Simulation (DS5101 RTLib Reference 🕮).

Execution times

For execution times of the RTLib functions, refer to Function Execution Times (DS5101 RTLib Reference 🕮). There you will find details of the measurement setup and the results.

Connecting external devices

For an I/O circuit and information on electrical characteristics and signal conditioning of the timing I/O unit, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and the RTLib functions, and the corresponding pins used for incremental encoder simulation.

Related RTI Block	Channel (RTI)	Related RTLib Functions	Channel (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101INC_Bx_BLy	Ch-block 1 (Channel 1)	See Incremental Encoder Simulation (DS5101 RTLib Reference 🕮)	Block 1 (via mask)	P35 20	CP1	IO1
	Ch-block 1 (Channel 2)		Block 1 (via mask)	P35 21	CP2	IO2
	Ch-block 2 (Channel 3)		Block 2 (via mask)	P35 22	CP3	IO3
	Ch-block 2 (Channel 4)		Block 2 (via mask)	P35 23	CP4	IO4
	Ch-block 3 (Channel 5)		Block 3 (via mask)	P35 24	CP5	IO5
	Ch-block 3 (Channel 6)		Block 3 (via mask)	P35 25	CP6	IO6
	Ch-block 4 (Channel 7)		Block 4 (via mask)	P35 26	CP7	IO7
	Ch-block 4 (Channel 8)		Block 4 (via mask)	P35 27	CP8	IO8
	Ch-block 5 (Channel 9)		Block 5 (via mask)	P35 28	CP9	IO9
	Ch-block 5 (Channel 10)		Block 5 (via mask)	P35 29	CP10	IO10
	Ch-block 6 (Channel 11)		Block 6 (via mask)	P35 30	CP11	IO11
	Ch-block 6 (Channel 12)		Block 6 (via mask)	P35 31	CP12	IO12
	Ch-block 7 (Channel 13)		Block 7 (via mask)	P35 32	CP13	IO13
	Ch-block 7 (Channel 14)		Block 7 (via mask)	P35 33	CP14	IO14

Conflicting features

The I/O features of the DS5101 conflict with each other. For details, see Limitations of the DS5101 on page 47.

Related topics

Basics

Basics of the Timing I/O Unit..... 14

References

ds5101_inc_load (DS5101 RTLib Reference 🕮)
DS5101INC_Bx_BLy (DS5101 RTI Reference 🕮)
Incremental Encoder Simulation (DS5101 RTLib Reference 🕮)

Monoflop Signal Generation

Introduction

The timing I/O unit of the DS5101 provides up to 8 monoflop signal generators. Each generator requires 2 output channels, one for the non-inverted and one for the inverted monoflop signal (see also I/O mapping on page 36).

After monoflop signal generation is triggered, a single high-active pulse — the monoflop pulse — is output on the specified non-inverted channel and a single low-active pulse is output on the inverted channel.

Monoflop signals can be used to generate ignition pulses, for example.

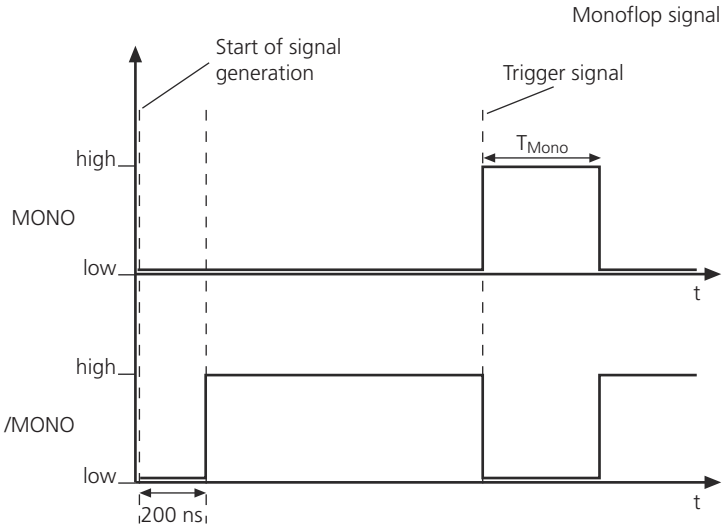
Note

When signal generation starts, the inverted signal is set to low for a duration of 200 ns. After that, the inverted signal changes to high.

Monoflop pulse length

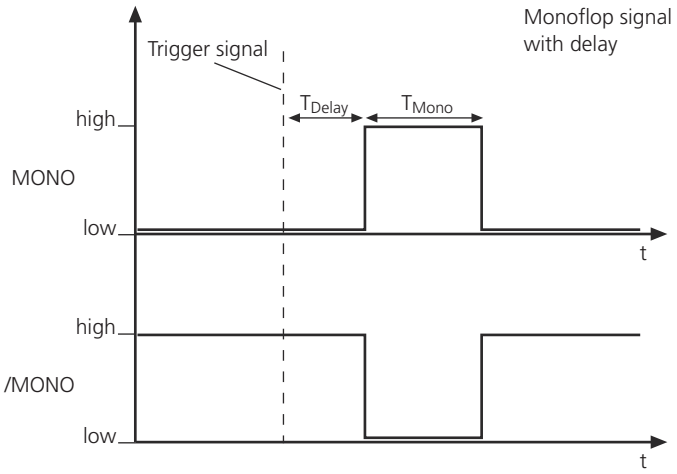
For monoflop signal generation on the DS5101, you can specify the monoflop pulse length T_{Mono} for each generator individually. You can specify T_{Mono} in the range 500 ns … 26 s.

If T_{Mono} is changed during run time, the new value becomes effective for the next monoflop pulse.



Delayed pulse generation

RTLib provides functions for monoflop signal generation with a time delay, T_{Delay}. After monoflop signal generation is triggered, the output remains unchanged for T_{Delay}. Then the monoflop pulse is output. You can specify T_{Delay} in the range 250 ns … 26 s. The following illustration shows a monoflop signal with delay.



If T_{Delay} is changed during run time, the new value becomes effective for the next delayed monoflop pulse.

Using RTI, you have to program delayed monoflop signal generation with RTLib functions, and incorporate your C code in a Simulink S-function. For details, refer to Implementing S-Functions (RTI and RTI-MP Implementation Guide 🕮).

Triggering monoflop signal generation

There are several ways to trigger monoflop signal generation.

RTLib provides the `ds5101_mono_start` function.

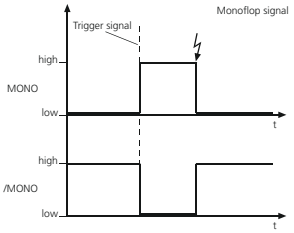
RTI Blockset contains three blocks for monoflop signal generation, each supporting a different trigger:

- Using **DS5101MONO_Bx_BLy**, the monoflop signal is triggered by a block input. You can set the trigger level.
- Using **DS5101MONOE_Bx_BLy**, the monoflop signal is triggered when the enabled subsystem is executed.
- Using **DS5101MONOT_Bx_BLy**, the monoflop signal is triggered when the subsystem is triggered.

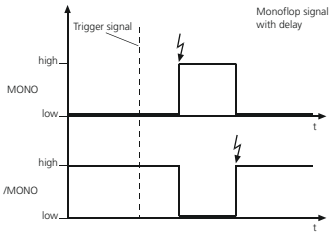
Interrupt via monoflop signal generation

With RTLib, you can enable interrupt generation for monoflop signals (refer to ds5101_int_enable (DS5101 RTLib Reference 🕮)). To make the interrupts available for RTI, you have to program monoflop signal generation with RTLib functions, and incorporate your C code in a Simulink S-function. For details, refer to Implementing S-Functions (RTI and RTI-MP Implementation Guide 🕮).

- For monoflop signals without delay, an interrupt is generated on the falling edge of the non-inverted monoflop signal.



- For delayed monoflop signals, interrupts are generated on the rising edge of the non-inverted monoflop pulse and on the rising edge of the inverted monoflop pulse.



For short monoflop pulses, you should use the `ds5101_int_enable` function to enable interrupt generation on one of the two channels only.

Downloading DWO application

When using monoflop signal generation, the corresponding DWO application must be downloaded to the channel controllers of the concerned channels before you start signal generation. This is done automatically if you use RTI or when using RTLib, by executing the loader function `ds5101_mono_load` (for monoflop signals without delay) or `ds5101_delayed_mono_load` (for monoflop signals with delay).

RTI/RTLib support	To perform monoflop signal generation use DS5101 Blockset and RTLib. For details, see ▪ RTI: Timing I/O Unit (DS5101 RTI Reference 🕮). ▪ RTLib: Monoflop Signal Generation (DS5101 RTLib Reference 🕮).
Execution times	For execution times of the RTLib functions, refer to Function Execution Times (DS5101 RTLib Reference 🕮). There you will find details of the measurement setup and the results.
Connecting external devices	For an I/O circuit and information on electrical characteristics and signal conditioning of the timing I/O unit, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and the RTLib functions, and the corresponding pins used for monoflop signal generation.

Related RTI Block	Channel (RTI)	Related RTLib Functions	Channel (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101MONO_Bx_BLy/ DS5101MONOE_Bx_BLy / DS5101MONOT_Bx_BLy	Ch-block 1 (Channel 1)	See Monoflop Signal Generation (DS5101 RTLib Reference 🕮)	Block 1 (via mask)	P35 20	CP1	IO1
	Ch-block 1 (Channel 2)		Block 1 (via mask)	P35 21	CP2	IO2
	Ch-block 2 (Channel 3)		Block 2 (via mask)	P35 22	CP3	IO3
	Ch-block 2 (Channel 4)		Block 2 (via mask)	P35 23	CP4	IO4
	Ch-block 3 (Channel 5)		Block 3 (via mask)	P35 24	CP5	IO5
	Ch-block 3 (Channel 6)		Block 3 (via mask)	P35 25	CP6	IO6
	Ch-block 4 (Channel 7)		Block 4 (via mask)	P35 26	CP7	IO7
	Ch-block 4 (Channel 8)		Block 4 (via mask)	P35 27	CP8	IO8
	Ch-block 5 (Channel 9)		Block 5 (via mask)	P35 28	CP9	IO9
	Ch-block 5 (Channel 10)		Block 5 (via mask)	P35 29	CP10	IO10
	Ch-block 6 (Channel 11)		Block 6 (via mask)	P35 30	CP11	IO11
	Ch-block 6 (Channel 12)		Block 6 (via mask)	P35 31	CP12	IO12
	Ch-block 7 (Channel 13)		Block 7 (via mask)	P35 32	CP13	IO13
	Ch-block 7 (Channel 14)		Block 7 (via mask)	P35 33	CP14	IO14
	Ch-block 8 (Channel 15)		Block 8 (via mask)	P35 34	CP15	IO15
	Ch-block 8 (Channel 16)		Block 8 (via mask)	P35 35	CP16	IO16

Conflicting features

The I/O features of the DS5101 conflict with each other. For details, see Limitations of the DS5101 on page 47.

Related topics

Basics

Basics of the Timing I/O Unit..... 14

References

ds5101_delayed_mono_load (DS5101 RTLib Reference 🕮)
ds5101_mono_load (DS5101 RTLib Reference 🕮)
ds5101_mono_start (DS5101 RTLib Reference 🕮)
DS5101MONO_Bx_BLy (DS5101 RTI Reference 🕮)
DS5101MONOE_Bx_BLy (DS5101 RTI Reference 🕮)
DS5101MONOT_Bx_BLy (DS5101 RTI Reference 🕮)
Monoflop Signal Generation (DS5101 RTLib Reference 🕮)

Custom DWO Applications

Where to go from here

Information in this section

Generation of Arbitrary Signals

Introduction

You can generate arbitrary pulse patterns for your model.

Arbitrary signal generation

To make the generation of arbitrary pulse patterns available for your model, proceed as follows:

1. Write your own DWO source code. In this code you define delay values and states for generating arbitrary signals with the DS5101 digital waveform output board.

   There are some limitations for signal generation. Refer to Limitations of the DS5101 on page 47.
2. Compile the DWO source file using the DWO compiler.

   The DWO compiler generates a C source file and a header file.

   The C source file contains the DWO application code and instructions for downloading the DWO application code to the DS5101 and starting signal generation. It can also include ready-to-use update functions to modify delay values while the DWO application is running. For basic information on updating delay values, refer to Updating Delay Values on page 39.

   The header file provides the address definitions for the timing parameters declared in the DWO source file.
3. Add the C source file and the header file to your model to be executed by the processor board.

For information on the DWO software environment and the DWO compiler, refer to DS5101BoardReference.pdf.

RTLib support

RTLib provides several functions that support generation of arbitrary signals:

- Board Initialization (DS5101 RTLib Reference 🕮)
- Arbitrary Signal Generation (DS5101 RTLib Reference 🕮)
- Interrupts (DS5101 RTLib Reference 🕮)

RTI block support

The DS5101 Blockset provides the **DS5101DWO** block to enable parameter updating of a custom DWO application in your Simulink model. For this a Simulink S-function template is serviced by RTI, where the DWO C source file, the header file, and the number of parameters must be specified. You have to adapt the S-function template according to the requirements of your DWO application. Refer to **DS5101DWO** (DS5101 RTI Reference 🕮).

Related topics

Basics

Basics of the Timing I/O Unit 14
Updating Delay Values 39

References

Limitations of the DS5101 47

Updating Delay Values

Basics on updating delay values

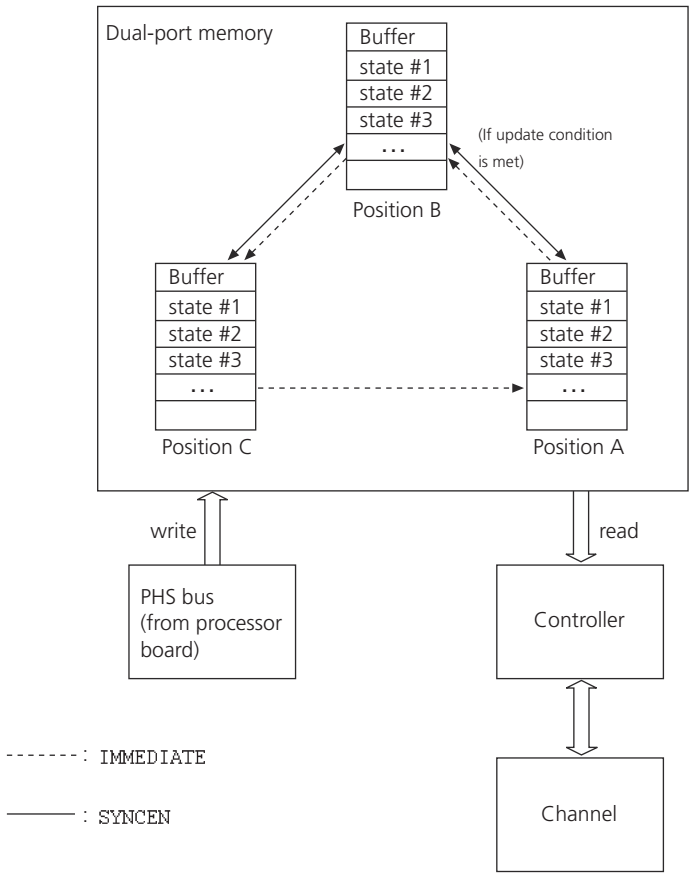
You can update delay values of a DWO application while the application is running. For this purpose you can generate a `ds5101_<application>_update_<name>` function with the DWO compiler that writes the new delay values to the memory of the DS5101 digital waveform output board. If you include your DWO application in a Simulink model you have to specify this update function in the S-function template for DWO applications.

Basics on the swinging buffer principle

For delay value updates during run-time, the dual-port memory internally uses the swinging buffer principle for each channel of the board's timing I/O unit. This avoids inconsistencies during delay value updates.

Reading from and writing to the memory

The illustration below shows the structure of the three buffers for one channel. New delay values coming from the processor board via PHS bus are always written to the buffer at position C. The controller always reads the buffer at position A.



Initial delay values

When you load a DWO application to the DS5101 digital waveform output board, the initial delay values of all the delay constants that are defined in the DWO application are loaded to all the three buffers. The controller reads the buffer at position A.

Delay value update

Delay values can be updated with the update function that is generated by the DWO compiler from the DWO source code. This function writes the new delay values to the buffer at position C.

During delay value update, the DS5101 exchanges only the buffer positions so that no buffer needs to be copied from one position to another. The way in which buffer positions are exchanged depends on the update mode which you specify in the DWO source code for each channel. You can specify the update mode with the `IMMEDIATE` and the `SYNCEN` DWO commands. If you do not specify the update mode the buffer positions are not exchanged at all, i.e. the delay values are not updated.

Update Command	Position C is Changed to …	Position A is Changed to …	Position B is Changed to …
IMMEDIATE	Position A	Position B	Position C
SYNCEN	Position B	Position A (position remains unchanged until update condition is met)	Position C

A delay value update is complete when the new values are at position A.

- In the immediate mode, the controller uses new values immediately since buffer position C is directly changed to position A.
- In the syncen mode, the position of the buffer containing new values is first changed from C to B. The position is changed to A when the update condition is met.

Related topics

Basics

Generation of Arbitrary Signals..... 38

Interrupts Provided by the DS5101

Introduction

The DS5101 provides access to various hardware interrupts – originating either from on-board devices, or from external devices connected to the DS5101.

Where to go from here

Information in this section

Information in other sections

Signal Generation Interrupt

Introduction

When you perform signal generation, you can enable interrupt generation for each channel and specify whether the interrupt is generated on the rising or falling edge of the generated signal. The following applications provide interrupts:

PWM3, PWM6

For 3-phase PWM signal generation (PWM3) and 3-phase PWM signal generation with inverted and non-inverted outputs (PWM6), the clock channel generates an interrupt at $\texttt{0.5} \cdot t_p - \texttt{intshift}$, where the interrupt shift depends on the PWM period. For more detailed information, see 3-Phase PWM Signal Generation (PWM3) on page 20 or 3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (PWM6) on page 24.

Monoflop

For monoflop signal generation, the interrupt is generated on the falling edge of the high-active monoflop signal. The interrupt cannot be accessed directly via RTI. For more detailed information, see Monoflop Signal Generation on page 32.

Monoflop with delay

For delayed monoflop signals, the interrupt is generated on the rising edge of the non-inverted high-active monoflop signal and on the rising edge of the inverted low-active monoflop signal. The interrupt cannot be accessed directly via RTI. For more detailed information, see Monoflop Signal Generation on page 32.

Custom DWO application

DWO applications can trigger interrupts on each of the 16 output channels. To make the interrupt available for arbitrary signals, you have to include the required interrupt directives in your custom DWO application. For more detailed information, see Generation of Arbitrary Signals on page 38.

RTI/RTLib support

For information on how to access the signal generation interrupts, see

- RTI: DS5101_HWINT_Bx_Iy (DS5101 RTI Reference 🕮)

  Depending on the kind of signal you generate, your Simulink model must also contain the **DS5101PWM3_Bx_BLy** or the **DS5101PWM6_Bx_BLy** block to implement interrupt generation.
- RTLib: Interrupts (DS5101 RTLib Reference 🕮)

Interrupt processing

Via the interrupt lines of the PHS bus, interrupts from the DS5101 are sent to the interrupt controller of the connected processor board. Using RTI, the interrupts of the DS5101 can therefore be used to implement interrupt-driven tasks. For details, refer to Tasks Driven by Interrupt Blocks (RTI and RTI-MP Implementation Guide 🕮).

Related topics

References

DS5101PWM3_Bx_BLy (DS5101 RTI Reference 🕮)
DS5101PWM6_Bx_BLy (DS5101 RTI Reference 🕮)

External Reset Interrupt

Introduction

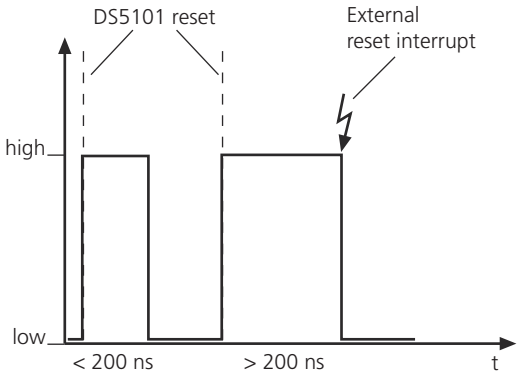
The DS5101 provides an external reset interrupt that you can use as a trigger source in a real-time application. The external reset signal has to be connected to the DS5101 I/O connector.

External reset

The rising edge of the external reset signal triggers a DS5101 reset. All DS5101 modules are reset, the channels are set to input, and the pull-up resistors of all channels are connected to V_{cc}.

External reset interrupt

If the external reset signal is kept high for at least 200 ns the interrupt controller of the DS5101 generates an interrupt request.



RTI/RTLib support

For information on how to access the external reset interrupt, see

- RTI: DS5101_HWINT_Bx_Iy (DS5101 RTI Reference 🕮)
- RTLib: Interrupts (DS5101 RTLib Reference 🕮)

Connecting external devices

For a circuit diagram and information on the electrical characteristics and signal conditioning of the external reset signal, refer to Signal Connection to External Devices (PHS Bus System Hardware Reference 🕮).

I/O mapping

The following table shows the mapping between the RTI block and RTLib functions and the corresponding pins used by the external reset interrupt.

Related RTI Blocks	Ch (RTI)	Related RTLib Functions	Ch (RTLib)	Conn. Pin	Pin on CP	Signal
DS5101_HWINT_Bx_Iy	Ext. RESET int	See Interrupts (DS5101 RTLib Reference 🕮).	–	P35 37	CP18	RESET

Related topics

Basics

Timing I/O Unit..... 13

References

DS5101_HWINT_Bx_Iy (DS5101 RTI Reference 🕮)
Interrupts (DS5101 RTLib Reference 🕮)

Limitations of the DS5101

Where to go from here

Information in this section

Information in other sections

Limitations for Signal Generation

Introduction

There are some limitations for signal generation.

Limitations

Note the following limitations for signal generation:

- The minimum pulse length is 250 ns.
- Generally it is not possible to generate PWM signals or similar signals which are constantly low or high, for example, PWM signals with 0% or 100% duty cycle. There will always be a remaining pulse of at least 250 ns length.

The reason for these limitations is that the DS5101 always processes a sequence of changes in state and delays. Each change in state (command) is followed by a delay. A single command needs 250 ns to be executed. Consequently a change in state takes effect for at least 250 ns even if the successive delay value is set to 0.

This limitation does not occur if you use the standard 1-phase PWM signals with the standard timing I/O unit.

If you absolutely need to generate arbitrary signals like PWM signals that must constantly be high or low, you can use the standard flags in the DWO source code (See "if" statement in the *DS5101 Board Reference* (../Print/DS5101BoardReference.pdf) and ds5101_flag (DS5101 RTLib Reference 🕮)).

Related topics

Basics

1-Phase PWM Signal Generation (PWM1)..... 17
3-Phase PWM Signal Generation (PWM3)..... 20
Monoflop Signal Generation..... 32

Quantization Effects

Introduction

Signal generation is only feasible within the limits of the 25 ns time base of the timing I/O unit, which causes quantization errors that increase with increasing frequencies.

When performing 1-phase PWM signal generation (PWM1), for example, you will encounter considerable deviations between the desired frequency $f_{desired}$ and the generated frequency $f_{generated}$, especially for higher frequencies. The (quantized) generated signal frequency can be calculated according to the following equation:

$$f_{generated} = 1/(n \cdot R)$$

where R is the 25 ns time base, and n is the integer part of

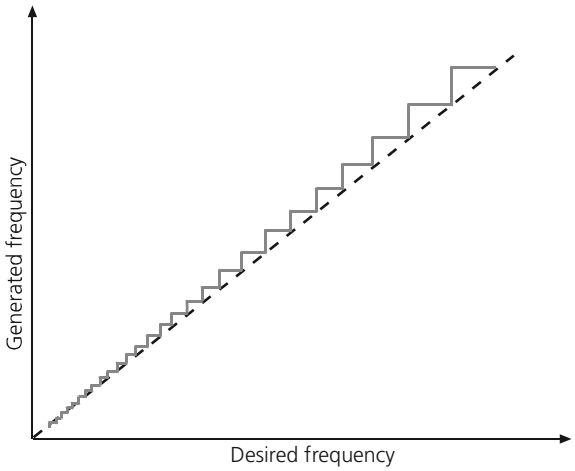
$$(1/(f_{desired} \cdot R))$$

Note that n must be ≥ 10.

Example

Suppose you want to generate a PWM signal with $f_{desired}$ = 300 kHz. Calculating the integer part of (1/(300 kHz · 25 ns)) yields n = 133. According to $f_{generated}$ = 1/(n · R), the generated frequency is 300.8 kHz.

The following illustration shows the increasing quantization effect for increasing desired frequencies:



Conflicting I/O Features

Introduction

The following I/O features of the DS5101 use signals IO1 … IO16:

- 1-Phase PWM Signal Generation (PWM1)
- 3-Phase PWM Signal Generation (PWM3)
- 3-Phase PWM Signal Generation with Inverted and Non-Inverted Outputs (PWM6) (PWM6 does not use IO8 and IO16)
- Incremental Encoder Simulation
- Monoflop Signal Generation
- Generation of Arbitrary Signals

Each of the output signals IO1 … IO16 can be used by only one I/O feature at the same time. For example, if you use IO2 for 1-Phase PWM Signal Generation (PWM1), you cannot use this channel to perform Incremental Encoder Simulation at the same time.

Numerics

A

C

D

E

F

I

L

M

P

Q

S

T