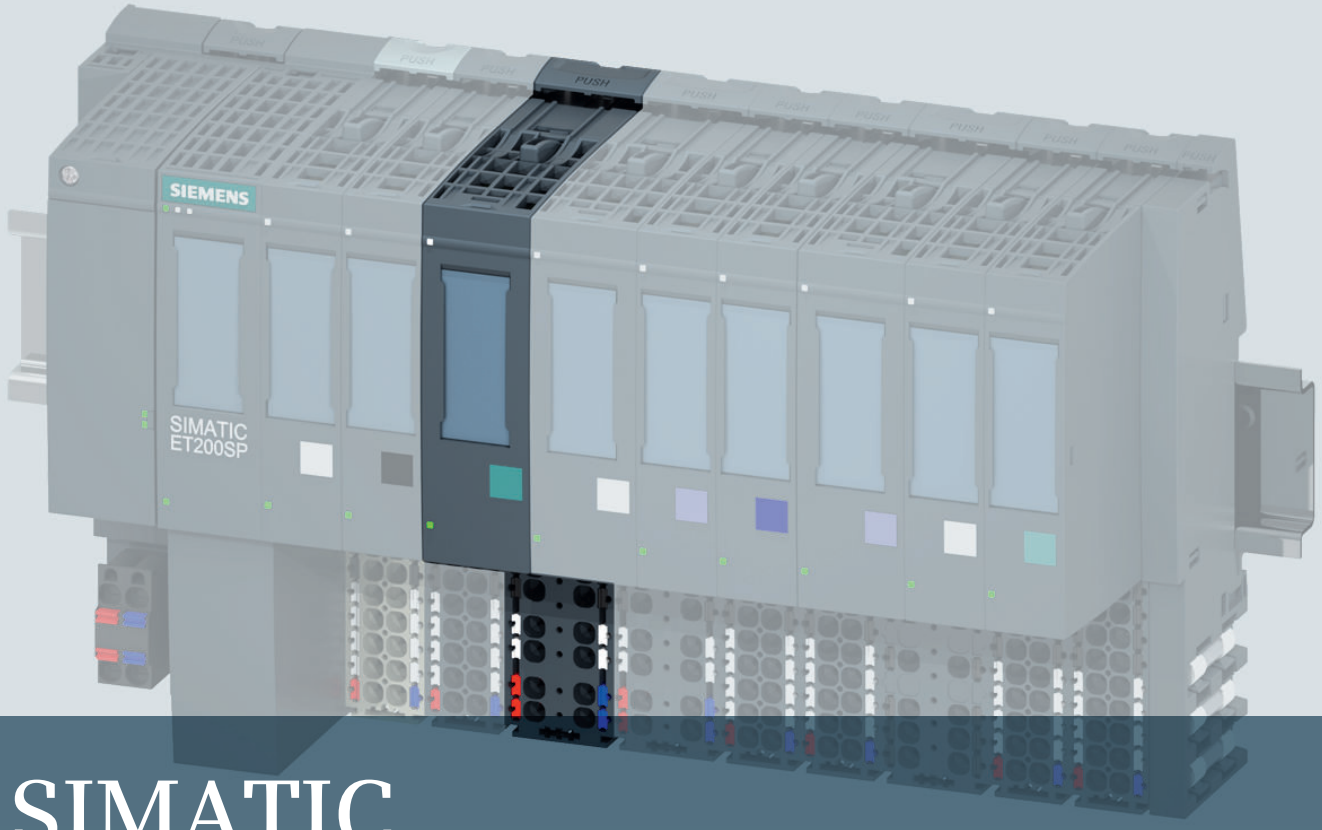
SIEMENS



SIMATIC

ET 200SP

Technology Module TM Pulse 2x24V (6ES7138-6DB00-0BB1)

Manual

Version 09/2015

Answers for industry.

SIEMENS

SIMATIC

ET 200SP
Technology module TM Pulse 2x24V (6ES7138-6DB00-0BB1)

Manual

Preface

Documentation guide 1

Product overview 2

Modes and Functions 3

Connecting 4

Configuring 5

Program control and feedback interface 6

Interrupts/diagnostic messages 7

Technical specifications 8

Parameter data record A

Open Source Software B

09/2015

A5E35061186-AA

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER

indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING

indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E35061186-AA
℗ 08/2015 Subject to change

Preface

Purpose of the documentation

This manual includes module-specific information on wiring, diagnostics and the technical specifications of the technology module.

General information regarding the design and commissioning of the ET 200SP is available in the ET 200SP system manual.

Conventions

Please observe notes marked as follows:

Note

A note contains important information on the product described in the documentation, on the handling of the product, and on the section of the documentation to which particular attention should be paid.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, solutions, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens' products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. You can find more information about industrial security on the Internet (http://www.siemens.com/industrialsecurity).

To stay informed about product updates as they occur, sign up for a product-specific newsletter. You can find more information on the Internet (https://support.industry.siemens.com/cs/?lc=en-US).

Open source software

Open source software is used in the firmware of the product described. Open source software is provided free of charge. We are liable for the product described, including the open source software contained in it, pursuant to the conditions applicable to the product. Siemens accepts no liability for the use of the open source software over and above the intended program sequence, or for any faults caused by modifications to the software.

For legal reasons, we are obliged to publish the original text of the license conditions and copyright notices. Please read the information relating to this in the "Open Source Software (Page 143)" appendix.

Table of contents

Preface 4
1 Documentation guide 8
2 Product overview 10
2.1 Properties 10
3 Modes and Functions 14
3.1 Overview 14
3.2 Pulse output (single pulse) mode 17
3.3 Pulse width modulation (PWM) mode 25
3.4 Pulse train mode 37
3.5 On/Off-delay mode 47
3.6 Frequency output mode 58
3.7 DC motor mode 66
3.8 Function: High-speed output 75
3.9 Function: Sequence counter 76
3.10 Function: Current measurement 77
3.11 Function: Current control 79
3.12 Function: Dither PWM output 83
3.13 Function: Isochronous mode 86
3.14 Function: Direct control of DQ digital outputs 87
4 Connecting 91
4.1 Pin assignment, sensor, load, and power wiring 91
5 Configuring 98
5.1 Configuration software 98
5.2 Configuration overview 99
5.3 Required I/O address space 100
5.4 TIA portal Device configuration 101
5.4.1 TIA Portal Device configuration 101
5.4.2 General information 102
5.4.3 Potential group 102
5.4.4 Channel configuration: (4 A) single or (2 A) dual channel operation 102

5.4.5 Channel parameters 103
5.4.5.1 Operating mode 103
5.4.5.2 Reaction to CPU STOP 103
5.4.5.3 Diagnostics 105
5.4.5.4 Parameter (Channel parameters) 106
5.4.6 I/O addresses 109
6 Program control and feedback interface 110
6.1 TM Pulse 2x24V control interface 110
6.2 SLOT parameter handling (control interface) 113
6.3 TM Pulse 2x24V feedback interface 115
7 Interrupts/diagnostic messages 117
7.1 Status and error displays 117
7.2 Parameter validation errors 120
7.3 Error detection and diagnostics 121
8 Technical specifications 124
8.1 Programming reference 130
A Parameter data record 137
B Open Source Software 143
Index 197

Documentation guide

1

Introduction

This modular documentation of a SIMATIC product covers automation system topics.

The complete documentation for the ET 200SP system consists of the respective system manuals, function manuals, and device manuals.

The STEP 7 information system (TIA Portal online Help) also helps you configure and program your automation system.

Overview of the documentation for TM Pulse 2x24V technology module

The following table lists further documentation that you will need when using the TM Pulse 2x24V technology module.

Table 1- 1 Documentation for TM Pulse 2x24V technology module

Topic	Documentation	Most important contents
System description	System manual ET 200SP Distributed I/O System (https://support.industry.siemens.com/cs/mdm/58649293?c=76156021003&t=1&s=BaseUnit&lc=en-US)	• Application planning • Installation • Connecting • Commissioning
	Device manuals Interface Module (https://support.industry.siemens.com/cs/products?dtp=Manual&pnid=14034&lc=en-US)	• Connecting • Interrupts, diagnostics, error, and system messages • Technical specifications • Dimension drawing
	Device manual for the ET 200SP BaseUnit 6ES7-193-6B20-0BB1 compatible with the TM Pulse 2x24V module. ET 200SP BaseUnits (https://support.industry.siemens.com/cs/mdm/59753521?c=72851856267&t=1&s=ET 200SP BaseUnit&lc=en-US)	Technical specifications

Topic	Documentation	Most important contents
Configuring interference-free controllers	SIMATIC S7-1500, ET 200MP, ET 200SP, ET 200AL Designing interference-free controllers (https://support.industry.siemens.com/cs/mdm/59193566?t=1&s=Designing interference free controllers function manual&lc=en-US) Function Manual	• Basics • Electromagnetic compatibility • Lightning protection
Isochronous mode	SIMATIC PROFINET with STEP 7 (https://support.industry.siemens.com/cs/mdm/49948856?c=73850691339&t=1&s=PROFINET with STEP 7 V13 SP1&lc=en-US) function manual	• Benefits • Use • Parameter settings

SIMATIC manuals

All current manuals for the SIMATIC products are available for download free of charge from the Internet (https://support.industry.siemens.com/cs/?lc=en-US).

Product overview

2

2.1 Properties

Article number

6ES7138-6DB00-0BB1

View of the module

Figure 2-1 View of the TM Pulse 2x24V module

Properties

The TM Pulse 2x24V technology module has the following properties:

- 2 pulse output channels with up to 2 A output current per channel
  - One-channel operation: The two channels are merged together into one logical channel and are connected in parallel to generate pulse signals with up to 4 A output current.
  - Two-channel operation: The two channels can work independently of each other.
- Programmable pulse timing.

Pulse timing	Minimum		Maximum	
	High-speed disabled	High-speed enabled	High-speed disabled	High-speed enabled
Pulse duration	10 µs [1]	1.5 µs [1]	85,000,000 µs (85 s)	
Period duration	100 µs	10 µs		
On-delay	0 µs			
Off-delay				
Frequency	0.02 Hz		10 kHz	100 kHz

[1] A smaller value is possible but not guaranteed for Pulse train, PWM, On/Off-delay, and DC motor modes

Parallel mode	Maximum pulse output current	
	High-speed disabled	High-speed enabled
Disabled	2 A (two channels)	100 mA (two channels)
Enabled	4 A (one channel)	Not allowed

- You can configure DI0.0 (Channel 0) and DI1.0 (Channel 1) digital inputs as hardware enable inputs that start the output pulse sequence, or as inputs directly usable by your program and independent of the pulse generation. In DC motor mode, you can stop the motor by using the digital inputs for external stop control.
- If you configure one channel 4 A operation, the two output channels are connected in parallel. Only Channel 0 is parameterized to control the parallel connection and only DI0.0 is available as a digital input/hardware enable.
- Each channel's digital output provides two load connections:
  - All modes except DC motor mode use a single unipolar output connection (DQn.A).
  - In DC motor, connect a load between the bipolar output connections (DQn.A and DQn.B). For example, a channel output can drive a DC motor in both directions by reversing the voltage polarity. Note that not all functions are available in DC motor mode; for example, current measurement and current control are not allowed.
- Digital outputs have integrated protection diodes to prevent voltage overstress due to inductive kickback. No external protection diodes are required for inductive loads.

Operating modes

- Pulse output: output a single pulse with variable pulse duration and On-delay.
- Pulse width modulation (PWM):
  - Output a frequency with a defined period duration and variable ratio of pulse width to period duration (duty cycle).
  - PWM can vary current in the driven loads. You can use PWM mode to control temperature in a heating resistor or the force from a coil in a proportional valve.
- Pulse train: output a train of pulses with an assigned pulse quantity, period duration, ratio of pulse width to period duration (duty cycle), and On-delay.
- On/Off-delay: the pulse output follows the signal at the DIn.0 digital input after an assigned On-delay and Off-delay.
- Frequency output: output pulses at a variable frequency with a fixed pulse width to pulse period duty cycle of 50%.
- DC motor: drive a motor in both directions with a bipolar PWM output. You can assign a digital input as an "External stop" signal, for the motor.

Functions

- Sequence counter: count completed output sequences and provide a feedback signal. The sequence counter can count short output sequences that are too fast for your program to monitor. For example, the counter is useful when the output sequence is faster than your program's cycle time and the output sequence is triggered by the DI digital input. The sequence counter is limited to 4 bits (count range 0 to 15).
- The sequence counter is available only for:
  - Pulse output and Pulse train modes.
    Without hardware enable, the counter is set to 1 after completion of the output sequence.
    With hardware enable, the counter is incremented after every completed output sequence.
  - On/Off-delay mode.
    The counter is incremented with each positive and negative edge at the DQn.A digital output.
- Current measurement: for Pulse train and PWM modes.
- Current control: PID loop control for proportional current control in PWM mode.
- Dithering: superimpose a dither signal on the PWM output to prevent valve sticking and improve proportional valve control.
- Direct control of the digital output by your control program.

- Programmable output response to CPU/master STOP condition.
- Error detection and diagnostics:
  - Missing or under voltage L+ supply voltage
  - Short-circuit/overload of a digital output
  - Short-circuit/under voltage of a sensor power supply
  - Over temperature error
  - Parameterization fault
  - Module/firmware error

Configuration

TM Pulse 2x24V configuration software options:

- TIA Portal V13 + SP1 with HSP 0131 (Hardware Support Package from the Internet)
- STEP 7 version V5.5 + SP4 with HSP 0240
- GSD file links:
  - PROFIBUS GSD files (https://support.industry.siemens.com/cs/document/73016883?dti=0&lc=en-US)
  - PROFINET GSD files (https://support.industry.siemens.com/cs/document/57138621?dti=0&lc=en-US)

Firmware update

Firmware updates can be downloaded to the memory of the TM Pulse 2x24V module by means of the STEP 7 TIA Portal software or the HW Config software.

Accessories

The following accessories can be used with the module and are not included in the product package:

- Labeling strip
- Color identification labels
- Reference identification labels

A BaseUnit of the B1 type is required to operate the technology module. For an overview of the BaseUnits to be used with the technology module, refer to the product information on the documentation for the ET 200SP Distributed I/O System (https://support.industry.siemens.com/cs/document/73021864?dti=0&lc=en-US).

For detailed information on the installation procedure, refer to the system manual for the ET 200SP Distributed I/O System (https://support.industry.siemens.com/cs/document/58649293?dti=0&lc=en-US).

Modes and Functions

3

3.1 Overview

Modes and functions

The TM Pulse 2x24V has two channels. You can assign a different mode for each channel.

Configuration of the operating mode is made using the TIA Portal or HW Config.

You can select one of six operating modes:

- Pulse output (single pulse)
- PWM
- Pulse train
- On/Off-delay
- Frequency output
- DC motor (PWM: forward and reverse rotation)

In addition to the operating mode, the TM Pulse 2x24V has the following functions:

- If enabled, the high-speed output option lets you generate a 1.5 µs minimum pulse duration at a current of 100 mA and a non-high-speed mode (high-speed mode disabled) with a minimum pulse duration of 10 µs and a maximum current of 2 A (two channels) or 4 A (one channel).
- The Sequence counter counts completed output sequences and provides a feedback signal.
- Current measurement for PWM and Pulse train modes
- Current control for the output current in PWM mode
- Dithering: superimpose a dithering signal on the PWM output to improve proportional valve control.
- You can directly control digital outputs with your control program, with separate control for each channel.
- Parallel connection of both channels creates one logical channel that can drive a 4 A output current.
- You can configure the response to CPU/master STOP. The outputs are put in the state that you assign for the control program STOP condition.
- The TM Pulse 2x24V provides module diagnostics and channel error detection.

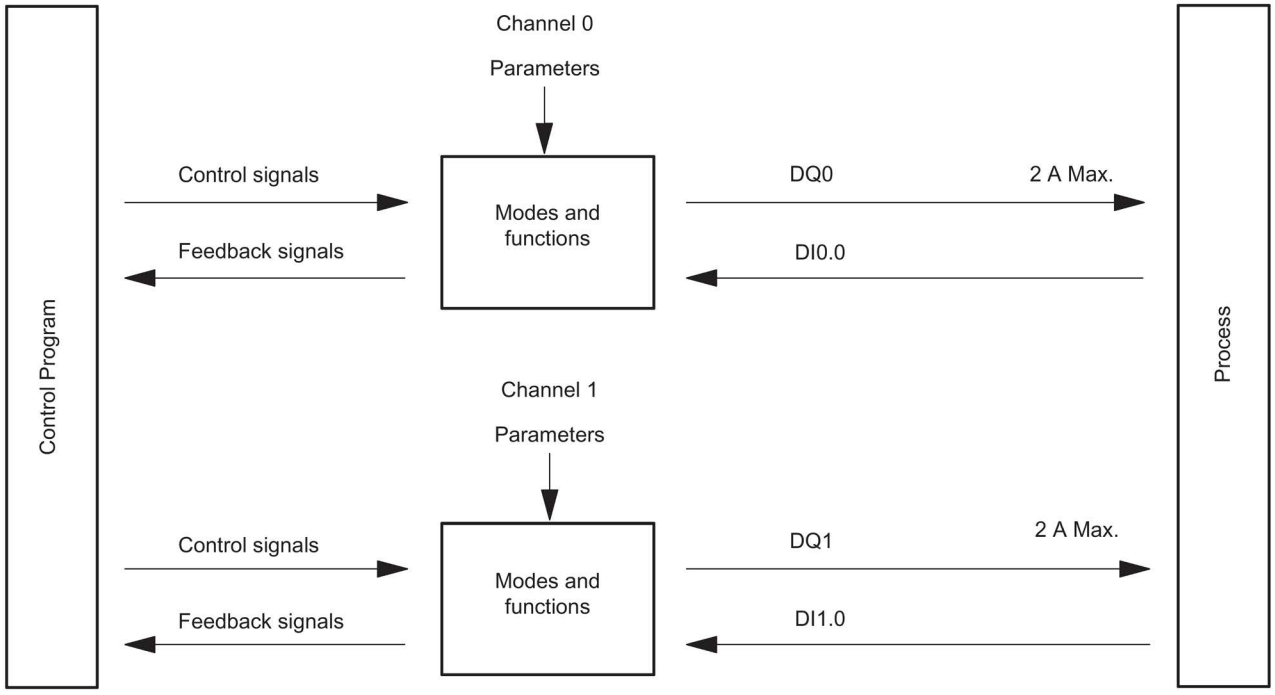


Figure 3-1 Dual channel operation

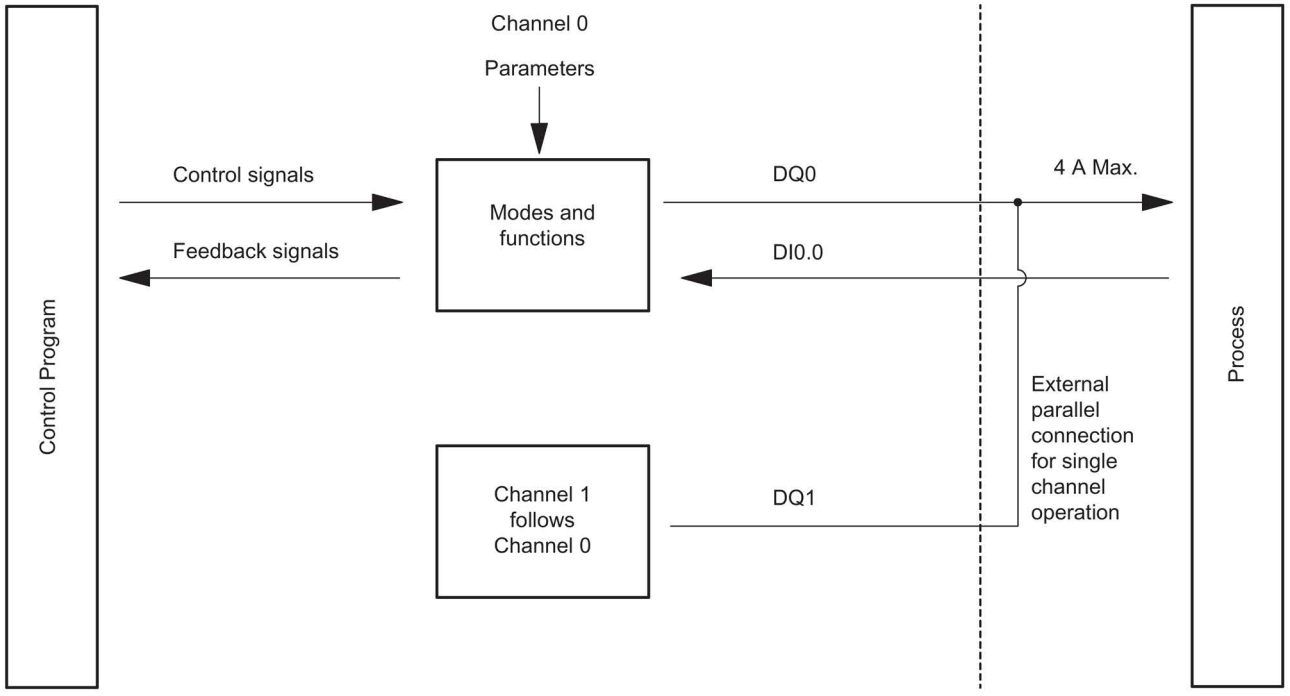


Figure 3-2 Single channel operation

Interfaces to the control program and the process under control

The TM Pulse 2x24V has the following I/O BaseUnit pin connections to the process under control:

Channel 0:

- DI0.0 (digital input 0)
- DQ0.A and DQ0.B (digital output 0)
  - Two connections are provided for each channel output (DQ0.A and DQ0.B). Load wiring depends on a channel's mode assignment.

Channel 1:

- DI1.0 (digital input 1)
- DQ1.A and DQ1.B (digital output 1)

You can find a table of configuration parameters in the "Parameter assignment and structure of the parameter data records (Page 137)" section.

You can modify and monitor the modes and functions with your control program using control and feedback signals. These parameters are listed in the "Control and feedback interface (Page 110)" section.

You will find the following in the "Modes and functions" section:

- Operation descriptions
- The relevant parameters
- The relevant control and feedback signals

The "Description of operations" for modes and functions applies to both channels. The terms DIn.0, DQn.A, and DQn.B are used in descriptions that are true for both channels.

3.2 Pulse output (single pulse) mode

Definition

After the assigned On-delay time expires, the TM Pulse 2x24V outputs a pulse at the DQn.A digital output (output sequence) for the pulse duration that you set.

Pulse diagram

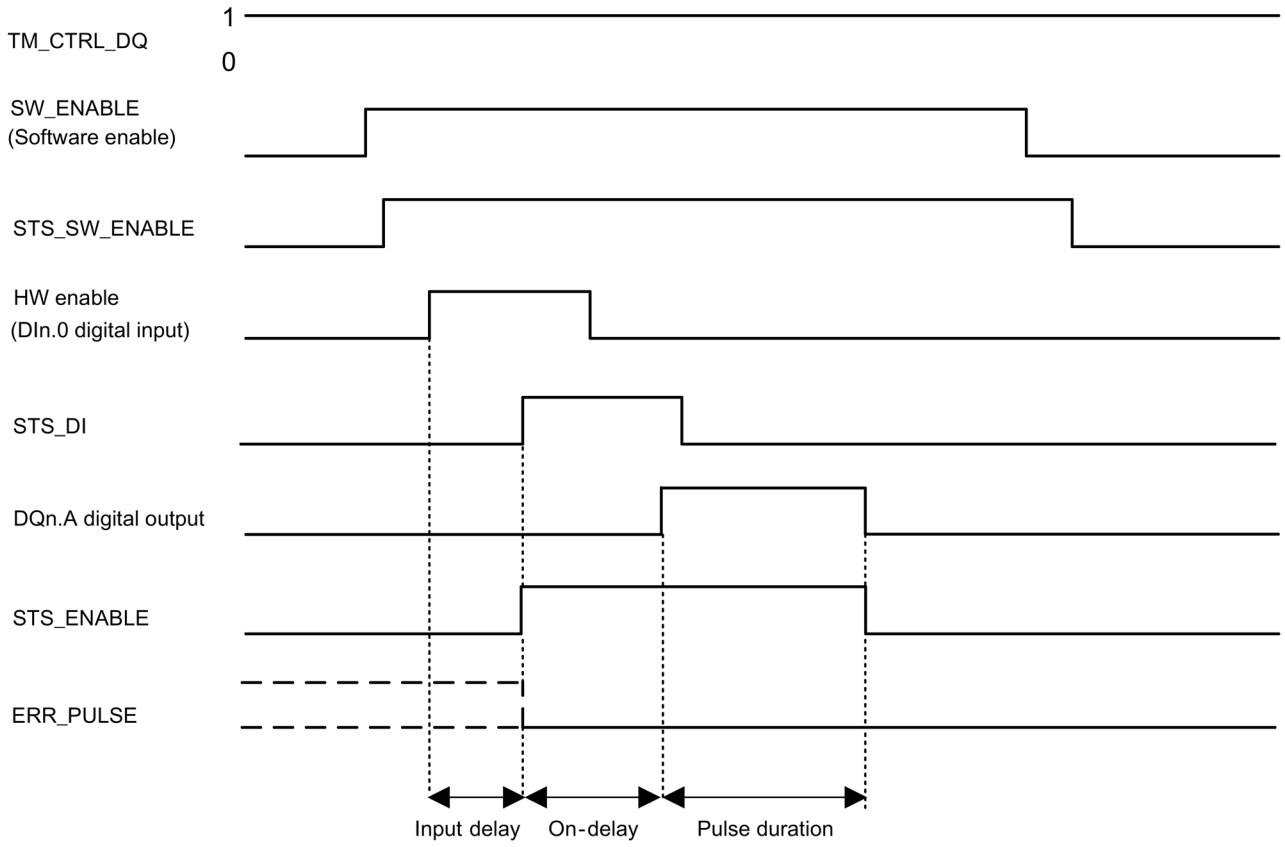


Figure 3-3 Pulse output mode output sequence using optional HW enable signal to start the output sequence

The timing diagram above has the "Function DI" parameter set to "HW enable". The other option is to set "Function DI" to "Input". If the "Function DI" parameter is set to "Input", then the On-delay phase starts at the rising edge of SW_ENABLE.

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates that a software enable is pending in the TM Pulse 2x24V.

You can also set the DIn.0 digital input of a TM Pulse 2x24V channel to be a hardware enable (HW enable) with the "Function DI" parameter. The input delay (noise filter) of the hardware enable can be set using the "Input delay" parameter.

If you want to use the hardware enable, it has to be combined with the software enable. When the software enable has been enabled, the output sequence starts at the first positive edge of the hardware enable. Further positive edges of the hardware enable during the current output sequence are ignored. When the HW enable goes high (positive edge) and remains high for the input delay time, the On-delay is started and the STS_ENABLE is set. After the On-delay expires, the pulse is output with the assigned pulse duration. The output sequence finishes with the end of the pulse and STS_ENABLE is cleared.

If you reduce the pulse duration to a time that has already expired, then the ERR_PULSE signal indicates a pulse output error and the output sequence stops. To resume the pulse output, you must restart the output sequence after an ERR_PULSE error occurs. The next time the output sequence starts, TM Pulse 2x24V clears the ERR_PULSE feedback bit.

Note

TM_CTRL_DQ technology module output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A outputs.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQn.A/DQn.B outputs directly with the SET_DQA/SET_DQB control bits.

Canceling the output sequence

Disabling the software enable (SW_ENABLE = 1 → 0) cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A digital output are immediately reset to 0.

You must restart the output sequence to begin new pulse output.

Truth Table

Software enable SW_ENABLE	Function DI parameter	Hardware enable (DIn.0 digital input)	DQn.A digital output (when TM_CTRL_DQ = 1)	STS_ENABLE	Output Sequence
1	HW_ENABLE	0 → 1 and remains 1 during the input delay. Only active for the first positive edge, additional positive edges are ignored and no start occurs.	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0 → 1	Input	Not used	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0	HW_ENABLE or Input	Any status	0	0	Terminate
1	HW_ENABLE or Input	Any status	0, if On-delay is not expired or if the pulse duration is expired. 1, if On-delay is expired and pulse duration is not expired.		-
0 → 1	HW_ENABLE	0	0	0	-

Setting and changing the pulse duration

Your control program can set the pulse duration directly with the control interface field OUTPUT_VALUE as a DWord number value in microseconds:

- High-speed output enabled, from 2 µs to 85,000,000 µs
- High-speed output disabled, from 10 µs to 85,000,000 µs

If you change the pulse duration when an output sequence is running, the pulse time already elapsed is subtracted from the new pulse duration and the pulse output continues.

Reducing the pulse duration

If you have reduced the pulse duration to a time that is less than the pulse time already elapsed, then the output sequence is terminated. STS_ENABLE and the DQn.A digital output are cleared, and the ERR_PULSE status bit is set. At the next output sequence, the ERR_PULSE status bit is cleared.

Setting and changing the On-delay

- **Permanent update**

  The On-delay can be controlled permanently using the control interface. The MODE_SLOT bit has to be 1 (permanent-update); LD_SLOT must have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the control interface field SLOT.

- **Single Update**

  Set the On-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters. Alternatively, you can do a single update using the control interface. MODE_SLOT has to be 0 (single-update); LD_SLOT has to have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the control interface field SLOT.

If you change the On-delay value during the output sequence, then the new On-delay is activated on the next output sequence.

For more details about the use of the SLOT value, see SLOT parameter handling (Page 113)

Using the Sequence counter

The Sequence counter counts completed pulse output sequences and provides the count value variable SEQ_CNT in the feedback interface. The sequence counter can count short output sequences that are too fast to be monitored by your program. See Sequence counter (Page 76) for details.

Isochronous mode

General information is available in Function: Isochronous mode (Page 86).

Isochronous mode does not have any influence on the functionality of Pulse output operating mode.

If you want to synchronize the output sequence with T_o, then set the Function DI parameter to "Input" and the Pulse output sequence starts at T_o.

Pulse output parameters

Pulse output parameter	Meaning	Value Range	Default
Mode	0 = Set the Pulse output operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC Motor	1
High-speed output [1]	The output supports higher frequencies at smaller loads when enabled.	0 = Disabled 1 = Enabled	Disabled
Function DI	You can use the DIn.0 digital input as an input or as a hardware enable. The signal at DIn.0 is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	0 = Input 1 = HW enable	Input
Input delay	DIn.0: a digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
On-delay	The time from the start of the output sequence to the output of the pulse. You can change the On-delay in your control program using the SLOT parameter.	0 µs to 85,000,000 µs	0 µs

[1] Only if the module is configured as "2 channels (2 A)"

Control and feedback signals for Pulse output mode

Control interface: Offset to the start address		**Parameter**	**Meaning**				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (DWord)	Pulse duration: the time that the DQn.A digital output remains set after the On-delay time expires. If you violate the lower or upper limit of the range, then ERR_OUT_VAL is returned in the feedback interface and the last valid value is used.				
			High-speed output disabled:				**High-speed output enabled:**
			10 µs to 85,000,000 µs				2 µs to 85,000,000 µs
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	The On-delay can be changed before the start of the output sequence. See MODE_SLOT.				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: all other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Idle state; nothing is done with the value
			0	0	1	0	On-delay in µs
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start/enable and terminate/disable the output sequence.			
			0	Output disabled/terminated			
			0 → 1	Starts output sequence on positive edge when "Function DI" = "Input"			
			1	Enable output sequence, when start is dependent on HW enable with "Function DI" = "HW enable"			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQn.A output source: Select either CPU program or module's output sequence.			
			0	DQn.A and DQn.B are controlled by the CPU (your program logic) using the SET_DQA and SET_DQB control bits.			
			1	DQn.A is controlled by the module's pulse output sequence. DQn.B is always 0.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ is cleared.			
			0	0 on DQn.A			
			1	1 on DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ and SET_DQA are cleared.			
			0	0 on DQn.B			
			1	1 on DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).			
			0	Reset of errors is not active			
			1	Reset of errors is active			

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1[1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage
			1	PWR is detected, but under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface) to reset this error.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT mode).
			0	No load error pending.
			1	Load error pending: you must set the RES_ERROR (control interface) to reset this error and be able to use the SLOT again.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Indicates a pulse output error.
			0	No pulse output error
			1	Pulse output error
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.B. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQs manually using SET_DQA, SET_DQB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B, or attempt to set both DQn.A and DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Indicates that an invalid value is detected in OUTPUT_VALUE.
			0	OUTPUT_VALUE is valid.
			1	OUTPUT_VALUE is not valid. This bit is reset automatically when a valid value is read by the module.
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT mode).
			0	SLOT value is valid.
			0 → 1	SLOT value is not valid. This bit is reset automatically when a valid value is read by the module.
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of the SLOT in "single-update" SLOT mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running.
			0	Output sequence not running
			1	Output sequence running

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	**Channel 1[1]**			
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 on DQn.A digital output
			1	1 on DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 on DQn.B digital output
			1	1 on DQn.B digital output
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	0 on DIn.0 digital input
			1	1 on DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Sequence counter is incremented after completion of an output sequence	
			With SW_ENABLE: 0 to 1	
			With HW enable: 0 to 15	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for Pulse output mode

Input and output signals	Meaning	Value range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
HW enable	You can select the HW enable with the "Function DI" parameter and select the input delay with the "Input delay" parameter. The signal at DIn.0 is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	0 = HW enable cleared 1 = HW enable issued 0 → 1 = Start of the output sequence after the input delay, dependent on the software enable (SW_ENABLE)	3	4
Output signal				
Pulse at the DQn.A digital output	A pulse is output at the DQn.A digital output for the set pulse duration.	0 = no pulse 1 = pulse	9	10

See also Pin assignment and load/sensor wiring (Page 91)

3.3 Pulse width modulation (PWM) mode

Definition

You control the pulse width duty cycle with the control interface field OUTPUT_VALUE. The TM Pulse 2x24V generates continuous pulses based on this value. The OUTPUT_VALUE control interface field determines the duty cycle (pulse duration/period duration) for PWM. The period duration can be adjusted.

After expiration of the assigned On-delay, the DQn.A output pulses begin (output sequence).

In PWM mode, you can enable current control using the internal PID loop function to control the output load current. When the current control option is enabled, the TM Pulse 2x24V control of the duty cycle and the OUTPUT_VALUE control interface field is used to assign the target current as the ratio of target current/reference current.

Pulse diagram

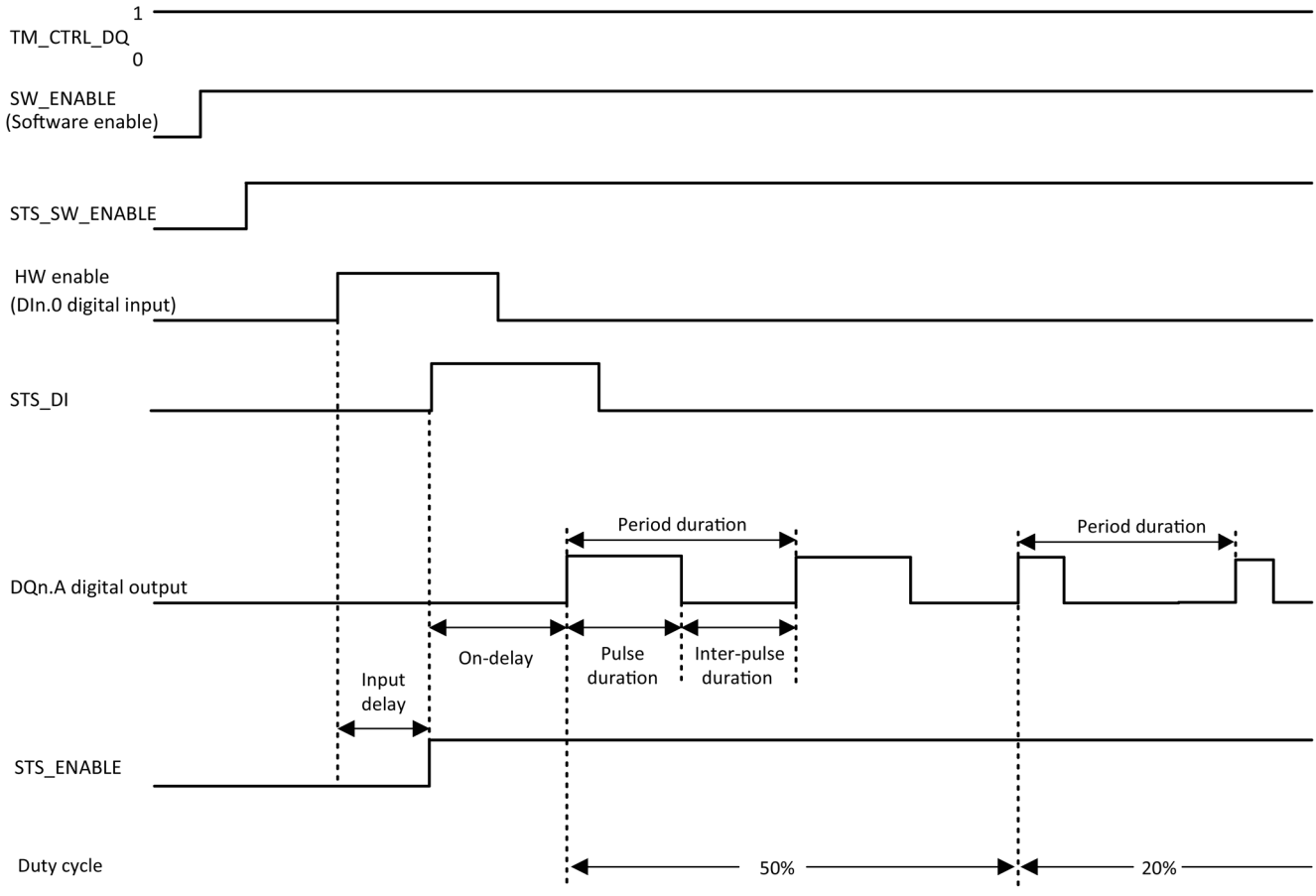


Figure 3-4 PWM output sequence

The timing diagram above has the "Function DI" parameter set to "HW_ENABLE". The other option is to set "Function DI" to "Input". If the "Function DI" parameter is set to "Input", then the On-delay phase starts at the rising edge of SW_ENABLE.

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates the software enable pending at the TM Pulse 2x24V.

You can also assign the DIn.0 digital input of the TM Pulse 2x24V as a HW enable with the "Function DI" parameter. The input delay (noise filter) of the hardware enable can be set using the parameter "Input Delay".

If you want to use the hardware enable, it has to be combined with the software enable. When the software enable has been enabled, the output sequence starts at the first positive edge of the hardware enable. Further positive edges of the hardware enable during the current output sequence are ignored by the TM Pulse 2x24V. The hardware enable option is not supported for isochronous mode.

When the enable is issued (positive edge) and remains high for the input delay time, the On-delay is started and the STS_ENABLE set. The PWM pulse train is output on expiration of the On-delay. The output sequence runs continuously as long as SW_ENABLE is set.

Note

Technology Module TM_CTRL output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A outputs.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQn.A/DQn.B outputs directly with the SET_DQA/SET_DQB control bits.

Canceling the output sequence

Disabling the software enable (SW_ENABLE = 1 → 0) cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A digital output are immediately reset to 0.

You must restart the output sequence to begin new pulse output.

Truth table

Software enable SW_ENABLE	Function DI parameter	Hardware enable (DIn.0 digital input)	Digital output DQn.A (when TM_CTRL_DQ = 1)	STS_ENABLE	Output sequence
1	HW_ENABLE	0 → 1 and remains 1 during the input delay. Only active for the first positive edge, additional positive edges are ignored and no start occurs.	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0 → 1	Input	Not used	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0	HW_ENABLE or Input	Any state	0	0	Terminate
1	HW_ENABLE or Input	Any state	0, if On-delay is not expired or you are in the interpulse time 1, if On-delay is expired and during the pulse duration		-
0 → 1	HW_ENABLE	0	0	0	-

Minimum pulse duration and minimum interpulse duration

The minimum pulse duration and minimum interpulse duration are superimposed on the proportional output characteristic.

You assign the minimum pulse duration and minimum interpulse duration using the "Minimum pulse duration" parameter; they always have the same value.

- A pulse duration calculated by the TM Pulse 2x24V that is shorter than the minimum pulse duration is suppressed.
- A pulse duration calculated by the TM Pulse 2x24V that is longer than the period duration minus the minimum interpulse duration is set to the period duration value (duty cycle 1,000‰).

Note

PWM current control and minimum pulse duration

If current control is activated by the current control bit in the parametrization record, then the parametrized minimum pulse duration and minimum interpulse period are ignored.

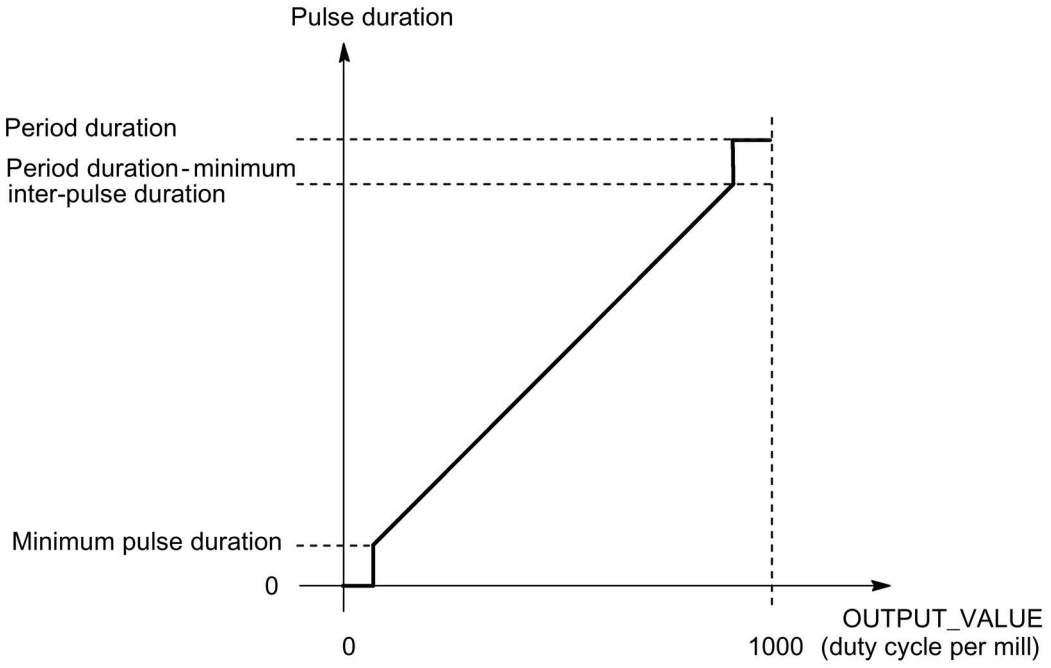


Figure 3-5 Modulation of the Pulse duration

Note

If Dithering is activated without current control, then the minimum pulse duration and minimum inter-pulse period are used by the module. In this case, the superimposed Dithering is reduced so the effective pulse duration fits in the allowed range.

Setting and changing the pulse duty cycle

OUTPUT_VALUE assigns the duty cycle for the current period duration. You select the range of the OUTPUT_VALUE control interface field with the "Output format" parameter.

- "Per 100 (%)" output format: Value range between 0 and 100
  Pulse duration = (OUTPUT_VALUE/100) x period duration.
- "Per 1,000 (‰)" output format: Value range between 0 and 1,000
  Pulse duration = (OUTPUT_VALUE/1,000) x period duration.
- "Per 10,000" output format: Value range between 0 and 10,000
  Pulse duration = (OUTPUT_VALUE/10,000) x period duration.
- "S7 analog output" output format: Value range between 0 and 27,648
  Pulse duration = (OUTPUT_VALUE/27,648) x period duration.

You assign OUTPUT_VALUE directly with your control program. A new OUTPUT_VALUE is applied at the next rising edge of the output.

When the current control option is enabled, the TM Pulse 2x24V module takes control of the duty cycle and the OUTPUT_VALUE control interface field is used to assign the target current as a ratio of target current/reference current. See the Current control (Page 79) function for details.

Setting and changing the period duration

- Permanent update
  The period duration is controlled permanently using the control interface. The MODE_SLOT bit has to be set ("1" means permanent update); LD_SLOT has to have the value 1 ("1" means Period duration).
  Set the period value in the field SLOT. The unit is always microseconds.
  - High-speed output **enabled**: between 10 µs and 85,000,000 µs in the field SLOT.
  - High- speed output **disabled**: between 100 µs and 85,000,000 µs in the field SLOT.
- Single Update
  Set the period duration in the configuration parameters. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared ("0" means single-update); LD_SLOT has to have the value 1 ('1' means Period duration). Set the period duration value in the field SLOT. The unit is always microseconds.
  - High-speed output **enabled**: between 10 µs and 85,000,000 µs in the parameters.
  - High-speed output **disabled**: between 100 µs and 85,000,000 µs in the parameters.

The new period duration is applied with the next rising edge of the output.

For more details about SLOT parameter handling, see "Slot parameter handling (control interface) (Page 113)".

Isochronous mode

General information is available in the "Function: Isochronous mode (Page 86)" topic.

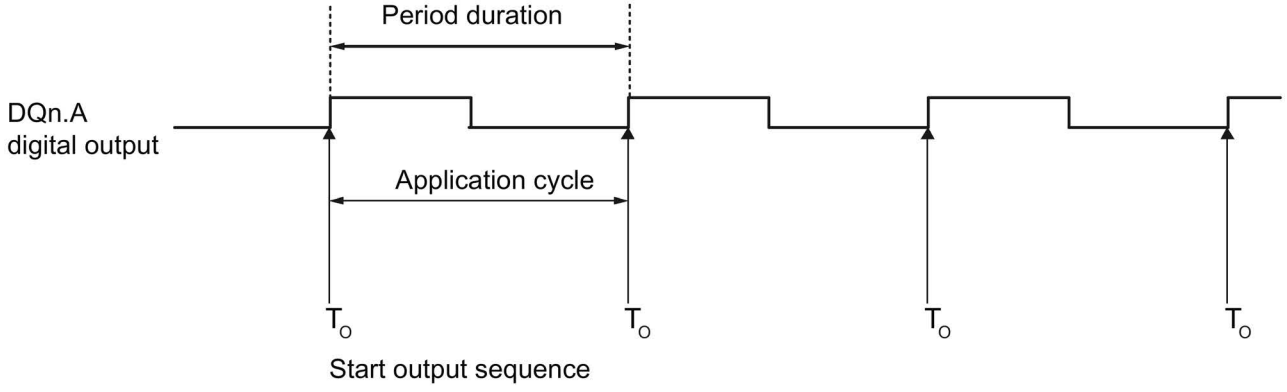
In isochronous mode, the output sequence is synchronized with the moment T_O. The period duration is coordinated to the application cycle (the synchronous cycle, a multiple of the PROFINET cycle). It is possible that your assigned period duration cannot be implemented exactly. The configured value of the period duration is adjusted by the TM Pulse 2x24V to the application cycle based on a calculation algorithm. The calculation is performed to minimize the difference between the assigned and calculated period duration. In the most unfavorable case, the deviation amounts to half the application cycle. The table below shows examples.

Application cycle T_{CAC} [1]	Assigned period duration $T_{Setpoint}$	$T_{CAC} \setminus T_{Actual}$	Calculated actual period duration T_{Actual} [1]	
10 ms (10000 µs)	5000 µs	2:1	5000 µs	
10 ms (10000 µs)	2000 µs	5:1	2000 µs	
10 ms (10000 µs)	3000 µs	3:1	3333 µs	Next possible value is used as actual value
10 ms (10000 µs)	1800 µs	6:1	1666 µs	Next possible value is used as actual value
10 ms (10000 µs)	6000 µs	2:1	5000 µs	Next possible value is used as actual value
10 ms (10000 µs)	12000 µs	1:1	10000 µs	Next possible value is used as actual value
10 ms (10000 µs)	16000 µs	1:2	20000 µs	Next possible value is used as actual value
10 ms (10000 µs)	26000 µs	1:3	30000 µs	Next possible value is used as actual value

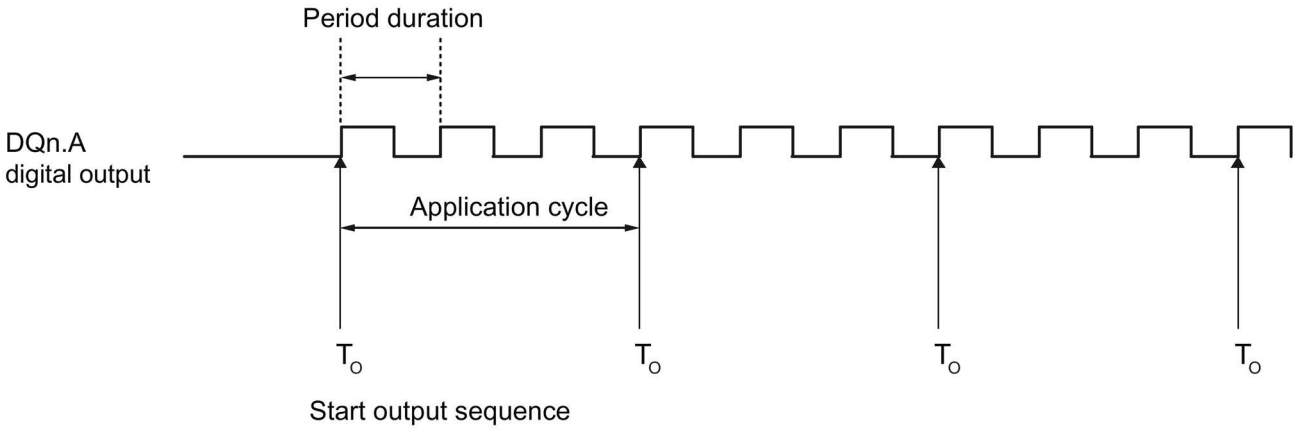
[1] The calculated actual period duration and the application cycle time always have an integer ratio (1:1, 1:2, 1:3, ..., 2:1, 3:1, ...) rounded down to the next possible value.

The timing between the digital output and the application cycle is shown below. The DQ duty cycle (On/Off) ratio is shown in the examples at 50%.

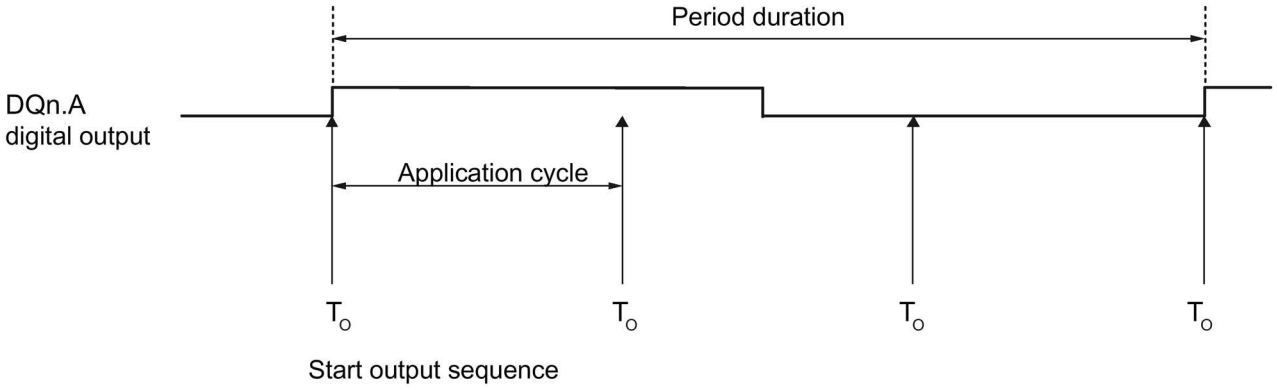
- Example 1: The period duration 10000 µs is equal to the application cycle time 10 ms (10000 µs).



- Example 2: The period duration 3333 µs is less than the application cycle time 10 ms (10000 µs).



- Example 3: The period duration 30000 µs is greater than the application cycle time 10 ms (10000 µs)



Note

Isochronous PWM operation

On-delay is not used by the module (always considered as zero) and the parameter "Function DI" is always "Input". Only the software-enable SW_ENABLE 0 → 1 is used to start the output sequence.

Setting the minimum pulse duration and minimum inter-pulse duration

You assign the minimum pulse duration and the minimum inter-pulse duration as a DWord number value between 0 and 85,000,000 µs using the "Minimum pulse duration" channel parameter configuration.

The unit is always microseconds. This value can only be changed using the configuration parameter record.

Setting and changing the On-delay

- Permanent update
  The On-delay can be controlled permanently using the control interface. The MODE_SLOT bit has to be set (permanent update); LD_SLOT must have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT. The unit is always microseconds.
- Single update
  Set the On-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters. The unit is always microseconds. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared (single-update); LD_SLOT must have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT.

If you change the On-delay value during the output sequence, then the new On-delay is activated at the next output sequence. For more details about the use of the SLOT parameter see SLOT parameter handling (Page 113).

Current measurement

Current measurement is available in PWM mode. Your control program can use current measurement for control and diagnostic purposes.

For more information, refer to Function: Current measurement (Page 77).

Current control

Current measurement is available in PWM mode. Your control program can use current measurement for control and diagnostic purposes.

For more information, refer to Function: Current control (Page 79).

Parameters of PWM operating mode

Parameter	Meaning	Value range	Default
Operating mode	1 = Set the PWM operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC Motor	1
High-speed output [1]	The output supports higher frequencies.	0 = disabled 1 = enabled	Disabled
Function DI	You can use the DIn.0 digital input as an input or as a hardware enable. The signal at DIn.0 is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	0 = Input 1 = HW enable	Input
Output format	Defines the format of the ratio value (duty cycle).	0 = S7 analog format 1 = Per 100 (%) 2 = per 1,000 3 = Per 10,000	Per 100 (%)
Input delay	DIn.0 digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
Minimum pulse duration	Minimum pulse duration and minimum inter-pulse duration.	0 µs and 85,000,000 µs	0 µs
Period	Period duration of the output pulse cycle in µs. You can change the period duration in your control program with the control interface SLOT field.	• High-speed output disabled 100 µs to 85,000,000 µs • High-speed output enabled 10 µs to 85,000,000 µs	2,000,000 µs
On-delay	The time from the start of the output sequence to the output of the pulses. You can change the On-delay in your control program with the control interface SLOT field.	0 µs and 85,000,000 µs	0 µs

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Control and feedback signals for PWM mode

Control interface Offset to the start address		Parameter	Meaning				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (DWord)	The OUTPUT_VALUE determines the duty cycle (pulse duration/period duration ratio) within a period (pulse width modulation). The period duration can be adjusted. The new output value is applied at the next rising edge of the output. When the current control option is enabled, the TM Pulse 2x24V module takes control of the duty cycle and the OUTPUT_VALUE control interface field is used to assign the target current as a ratio of target current/reference current. See the Current control function for details. UDInt data type: Only 2 least significant bytes are used For channel 0: bytes 2 and 3 For channel 1: bytes 14 and 15				
			Output format	"Per 100": value range is 0 to 100			
				"Per 1000": value range is 0 to 1,000			
				"Per 10000": value range is 0 to 10,000			
				"S7 analog output": value range is 0 to 27,648			
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	Your program can change the parameters shown in the byte 8 LD_SLOT table below, before the start of the output sequence using the SLOT and MODE_SLOT parameters.				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: all other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Parameter**
			0	0	0	0	Idle state; nothing is done with the value
			0	0	0	1	Period in µs
			0	0	1	0	On-delay in µs
			0	1	0	1	Dither ramp in ms
			0	1	1	0	Dither amplitude in per mill
			0	1	1	1	Dither period in µs
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: start and terminate the output sequence.			
			0	Output canceled			
			0 → 1	Starts output sequence on positive edge when "Function DI" = "Input".			
			1	Enable output sequence, when start is dependent on HW enable with "Function DI" = "HW enable".			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQn.A output source: Selects either CPU program or module's output sequence.			
			0	DQn.A and DQn.B are controlled by the CPU (in your program) using the SET_DQA and SET_DQB control bits.			
			1	DQn.A is controlled by the module's pulse output sequence. DQn.B is always 0 for TM_CTRL_DQ = 1.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ = 0.			
			0	0 on DQn.A			
			1	1 on DQn.A			

Control interface Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ = 0 and if SET_DQA is cleared.
			0	0 on DQn.B
			1	1 on DQn.B
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).
			0	Reset of errors is not active
			1	Reset of errors is active

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage
			1	PWR is detected, but under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface), to reset this error.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT-mode).
			0	No load error pending
			1	Load error pending: you must set the RES_ERROR (control interface) to reset this error and be able to use SLOT again.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQn.A and DQn.B manually using SET_DQA, SET_DB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B, or attempt to set both DQn.A and DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT-mode).
			0	SLOT value is valid
			0 → 1	SLOT value is not valid
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of SLOT in "single-update" SLOT-mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running.
			0	Output sequence not running
			1	Output sequence running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 on DQn.A digital output
			1	1 on DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 on DQn.B digital output
			1	1 on DQn.B digital output
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	0 on DIn.0 digital input
			1	1 on DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Sequence counter = 0. The sequence counter is not used in pulse width modulation mode.	
Word 2	Word 6	MEASURED_CURRENT	S7 Analog format positive value 0 to 32,767: • 2 Ampere channel output: Full scale value 27,648; means 2,000 mA • 4 Ampere channel output: Full scale value 27,648; means 4,000 mA • 32,767 means no valid current measurement available; for example, during the very first PWM period.	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for PWM operating mode

Input and output signal	Meaning	Value range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
HW enable Note: HW enable not supported for PWM in isochronous mod	You can select the HW enable with the "Function DI" parameter and select the input delay with the "Input delay" parameter. The signal at the DIn.0 digital input is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	0 = HW enable cleared 1 = HW enable issued 0 → 1 = Start of the output sequence after the input delay; dependent on the software enable (SW_ENABLE)	3	4
Output signal				
Pulse at the DQn.A digital output	A pulse is output at the DQn.A digital output for the set duty cycle and period duration.	0 = no pulse 1 = pulse	9	10

See also Pin assignment and load/sensor wiring (Page 91)

3.4 Pulse train mode

Definition

On expiration of the assigned On-delay, the TM Pulse 2x24V outputs the number of pulses you assigned as a pulse train (output sequence). The period duration and pulse duration can be adjusted.

Pulse diagram

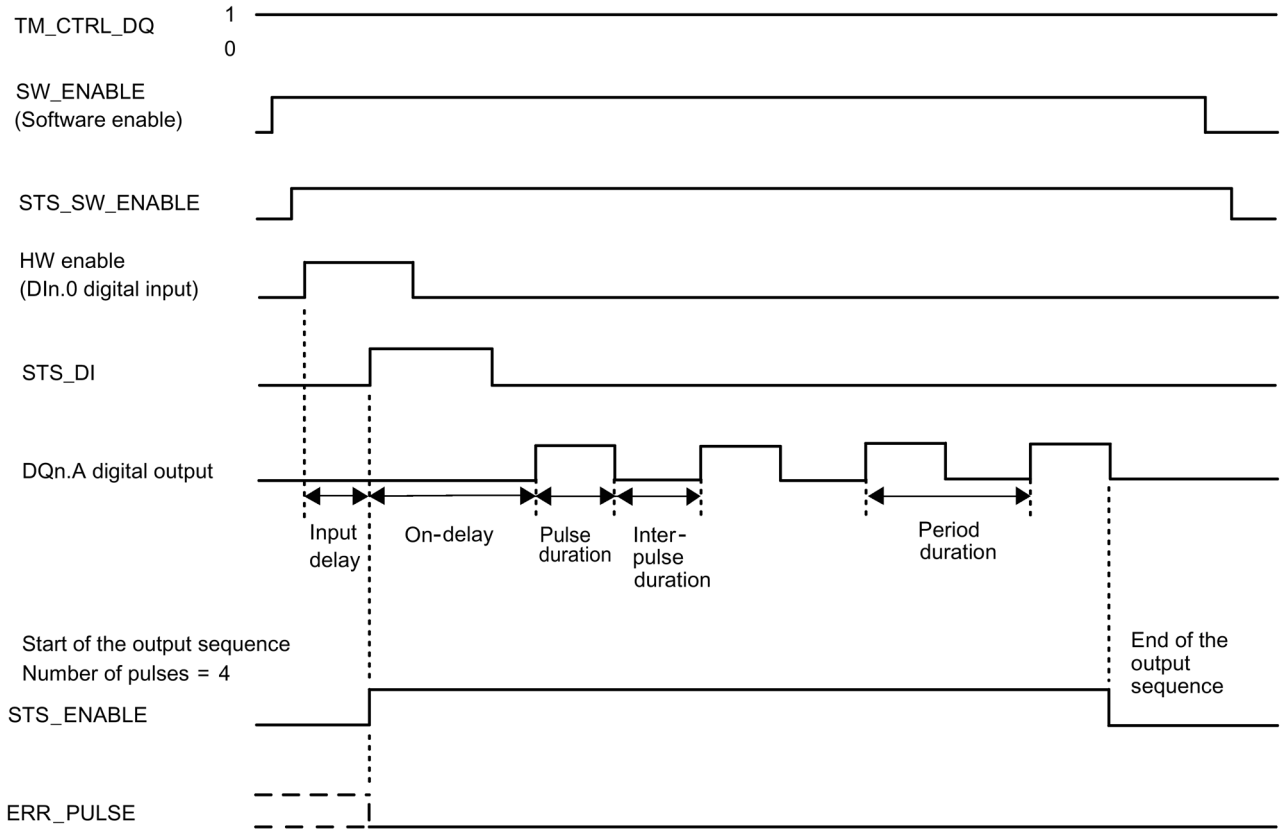


Figure 3-6 Output sequence of the pulse train

The timing diagram above has the "Function DI" parameter set to "HW enable". The other option is to set "Function DI" to "Input". If the "Function DI" parameter is set to "Input", then the On-delay phase starts at the rising edge of SW_ENABLE.

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates the software enable pending at the TM Pulse 2x24V.

You can also set the DIn.0 digital input of the TM Pulse 2x24V as a HW enable with the "Function DI" parameter. The input delay of the hardware enable can be set using the parameter "Input Delay".

If you want to use the hardware enable, it has to be combined with the software enable. When SW_ENABLE has been enabled, the output sequence starts at the first positive edge of the hardware enable. Further positive edges of the hardware enable during the current output sequence are ignored by the TM Pulse 2x24V. When the software enable has been issued, a positive edge of the hardware enable (detected after the input delay) starts the next output sequence.

When the enable is issued (positive edge) and remains high for the input delay (noise filter) time, the On-delay is started and STS_ENABLE is set. On expiration of the On-delay, the pulse train is output with the assigned pulse duration. The output sequence finishes with the end of the pulse and STS_ENABLE is cleared.

If you change the number of pulses during the output sequence to a value that has already been reached in the sequence, then the bit ERR_PULSE indicates a pulse output error

The next time the output sequence starts, the TM Pulse 2x24V clears the ERR_PULSE feedback bit.

Note

TM_CTRL_DQ technology module output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A output.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQn.A/DQn.B outputs directly with the SET_DQA/SET_DQB control bits.

Canceling the output sequence

Disabling the software enable (SW_ENABLE = 1 → 0) during the On-delay or pulse train cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A digital output are immediately reset to 0.

You must restart the output sequence to begin a new pulse output.

Truth table

Software enable SW_ENABLE	Function DI parameter	Hardware enable (DIn.0 digital input)	Digital output DQn.1 (when TM_CTRL_DQ = 1)	STS_ENABLE	Output sequence
1	HW_ENABLE	0 → 1 and remains 1 during the input delay. Only active for the first positive edge, additional positive edges are ignored and no start occurs.	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0 → 1	Input	Not used	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0	HW_ENABLE or Input	Any status	0	0	Cancel
1	HW_ENABLE or Input	Any status	0, if On-delay is not expired or you are in the inter-pulse time or the output sequence is over 1, if On-delay is expired and during the pulse duration		-
0 → 1	HW_ENABLE	0	0	0	-

Setting and changing the number of pulses

Your control program can set the pulse count directly with the control interface parameter OUTPUT_VALUE.

Set the number of pulses directly as a DWord number value between 1 to 4,294,967,295 (2^{32}-1).

If you change the number of pulses, the new value takes effect immediately whether the On-delay is expired or not.

- If the new number of pulses has not yet been reached in the current output sequence, then the new number of pulses will be used in the current output sequence.
- If the number of pulses is reduced to a number that is less than the current number of pulses already output, then the output sequence is terminated, STS_ENABLE and the DQn.A digital output are cleared, ERR_PULSE is set. At the next start of an output sequence, ERR_PULSE is cleared.
- If the number of pulses is zero the output sequence is terminated, STS_ENABLE and the DQn.A digital output are cleared, and ERR_PULSE is set. At the next start of an output sequence, ERR_PULSE is cleared.

Setting and changing the period duration

- Permanent update
  The period duration can be controlled permanently using the control interface. The MODE_SLOT bit has to be set ('1' means permanent update); LD_SLOT has to have the value 1 (for Period duration). Set the period duration as a DWord number value of microseconds in the parameter field SLOT.
  - High-speed output **enabled** (see parameters): between 10 µs and 85,000,000 µs.
  - High-speed output **disabled** (see parameters): between 100 µs and 85,000,000 µs.
- Single update
  Set the period duration in the configuration parameters. Alternatively, you can do a single update using the control interface. MODE_SLOT has to be 0 (single-update); LD_SLOT has to have the value 1 (for Period duration). Set the period duration as a DWord number value of microseconds in the parameter field SLOT.
  - High-speed output **enabled** (see parameters): between 10 µs and 85,000,000 µs.
  - High-speed output **disabled** (see parameters): between 100 µs and 85,000,000 µs.

Setting and changing the On-delay

- Permanent update
  The On-delay can be controlled permanently using the control interface. MODE_SLOT bit has to be set ('1' means permanent update); LD_SLOT must be the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the parameter field SLOT.
- Single update
  Set the On-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared ("0" means single update); LD_SLOT must be the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT.
- If you change the On-delay value during the output sequence, then the new On-delay is activated at the next output sequence.

For more details about the use of the SLOT parameter, see the SLOT parameter handling (Page 113).

Setting and changing the duty cycle

The range of the duty cycle parameter is selected using the "Output format" parameter. The TM Pulse 2x24V uses this assigned duty cycle value to calculate the pulse duration. If the number value you assign exceeds the upper limit, then a duty cycle of 100% of the period duration is used and this action does not cause an error.

- Output format "Per 100 (%)": Value range 0 to 100
  - Pulse duration = (duty cycle/100) x period duration.
- Output format "Per 1000": Value range 0 to 1,000
  - Pulse duration = (duty cycle/1,000) x period duration.
- Output format "Per 10000": Value range 0 to 10,000
  - Pulse duration = (duty cycle/10,000) x period duration.
- Output format "S7 analog output": Value range 0 to 27,648
  - Pulse duration = (duty cycle/27,648) x period duration.

Update the duty cycle

- Permanent update
  The duty cycle can be controlled permanently using the control interface. MODE_SLOT bit has to be set ("1" means permanent update); LD_SLOT must be the value 4 (for duty cycle). Set the duty cycle as a value in the control interface field SLOT. The unit depends on the assigned output format.
- Single update
  Set the duty cycle in the configuration parameters. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared ("0" means single update); LD_SLOT must be the value 4 (for duty cycle). Set the duty cycle as a value in the control interface field SLOT.
- The unit depends on the assigned output format. If you change the duty cycle value during the output sequence, then the new duty cycle is activated at the next output sequence.

For more details about the use of the SLOT parameter, see SLOT parameter handling (Page 113).

Isochronous mode

General information is available in Function: Isochronous mode (Page 86).

Isochronous mode does not have any influence on the functionality of Pulse train operating mode.

If you want to synchronize the output sequence with T_o, then set the Function DI parameter to "Input" and the Pulse train output sequence starts at T_o.

Current measurement

Current measurement is available in Pulse train mode. Your control program can use current measurement for control and diagnostic purposes.

For more information, refer to Function: Current measurement (Page 77).

Parameters for Pulse train operating mode

Parameter	Meaning	Value range	Default
Operating mode	2 = Set the "Pulse train" operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC motor	1
High-speed output [1]	The output supports higher frequencies (see pulse duration range).	0 = disabled 1 = enabled	Disabled
Function DI	You can use the DIn.0 digital input as an input or as a hardware enable. The signal at DIn.0 is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	• Input • HW enable	Input
Output format	Defines the format of the duty cycle ratio value.	0 = S7 analog format 1 = Per 100 (%) 2 = Per 1,000 3 = Per 10,000	Per 100 (%)
Input delay	DIn.0 digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
Period	Period duration of the output sequence: You can change the period duration in your control program using the control interface field SLOT.	• High-speed output disabled 100 µs to 85,000,000 µs • High-speed output enabled 10 µs to 85,000,000 µs	2,000,000 µs

Parameter	Meaning	Value range	Default
On-delay	The time from the start of the output sequence to the output of the pulse train. You can change the On-delay in your control program using the SLOT parameter.	0 µs to 85,000,000 µs	0 µs
Duty-cycle	The pulse duration results from the duty cycle and the period duration. The format of the duty cycle is determined by the parameter "Output format". You can change the duty-cycle in your control program using the control interface field SLOT.	Output format: minimum to maximum • Per 100: 0 to 100 • Per 1000: 0 to 1,000 • Per 10000: 0 to 10,000 • S7 analog output: 0 to 27,648	50%

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Control and feedback signals for Pulse train operating mode

Control interface Offset to the start address		Parameter	Meaning				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (DWord)	Assign the number of pulses to the OUTPUT_VALUE as a DWord number value 0 to 4,294,967,295 (2^{32}-1).				
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	The On-delay, period duration, and duty cycle can be changed before the start of the output sequence. See SLOT parameter handling (Page 113).				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: All other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Idle-state; nothing is done with the value
			0	0	0	1	Period duration in microseconds
			0	0	1	0	On-delay in microseconds
			0	1	0	0	Duty cycle in the format defined by the parameter "Output format"
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start/enable and terminate/disable the output sequence when "Function DI" = "Input".			
			0	Output disabled/terminated			
			0 → 1	Starts output sequence on positive edge when "Function DI" = "Input"			
			1	Enable output sequence, when start is dependent on HW enable with "Function DI" = "HW enable"			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQn.A output source: Selects either CPU program or module's output sequence.			
			0	DQn.A and DQn.B are controlled by the CPU (in your program) using the SET_DQA and SET_DQB control bits			
			1	DQn.A is controlled by the module's pulse output sequence. DQn.B is always 0			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ is cleared.			
			0	0 on DQn.A			
			1	1 on DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ is cleared and if SET_DQA is cleared.			
			0	0 on DQn.B			
			1	1 on DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).			
			0	Reset of errors is not active			
			1	Reset of errors is active			

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage
			1	PWR is detected, but under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface) to reset this error.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT-mode).
			0	No load error pending
			1	Load error pending: you must set the RES_ERROR (control interface) to reset this error and be able to use the SLOT again.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Indicates a pulse output error.
			0	No pulse output error
			1	Pulse output error
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQs manually using SET_DQA, SET_DQB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B, or attempt to set both DQn.A and DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT-mode).
			0	SLOT value is valid
			1	SLOT value is not valid
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of the SLOT in "single-update" SLOT-mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running.
			0	Output sequence not running
			1	Output sequence running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 at the DQn.A digital output
			1	1 at the DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 at the DQn.B digital output

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
			1	1 at the DQn.B digital output
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	Signal 0 at the DIn.0 digital input
			1	Signal 1 at the DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Sequence counter: is incremented after completion of an output sequence.	
			With SW_ENABLE: 0 to 1	
			With HW enable: 0 to 15	
Word 2	Word 6	MEASURED_CURRENT	S7 Analog format positive value 0 to 32,767: • 2 Ampere output: Full scale value 27,648, means 2,000 mA • 4 Ampere output: Full scale value 27,648, means 4,000 mA • 32,767 means no valid current measurement available; for example, during the very first period	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for Pulse train operating mode

Input and output signal	Meaning	Value range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
HW enable	You can select the HW enable with the "Function DI" parameter and select the input delay with the "Input delay" parameter. The signal of the DIn.0 digital input is interpreted by the TM Pulse 2x24V, after filtering by the input delay, as the start of the output sequence.	0 = HW enable cleared 1 = HW enable issued 0 → 1 = Start of the output sequence; dependent on the software enable (SW_ENABLE)	3	4
Output Signal				
Pulse train at the DQn.A digital output	Pulses are output at the DQn.A digital output for the set pulse duration.	0 = no pulse 1 = pulse	9	10

See also Pin assignment and load/sensor wiring (Page 91)

3.5 On/Off-delay mode

Definition

The signal pending at the TM Pulse 2x24V DIn.0 digital input is output with an assigned On/Off-delay at the DQn.A digital output.

Pulse diagram

SW_ENABLE is set, while DIn.0 digital input = 0:

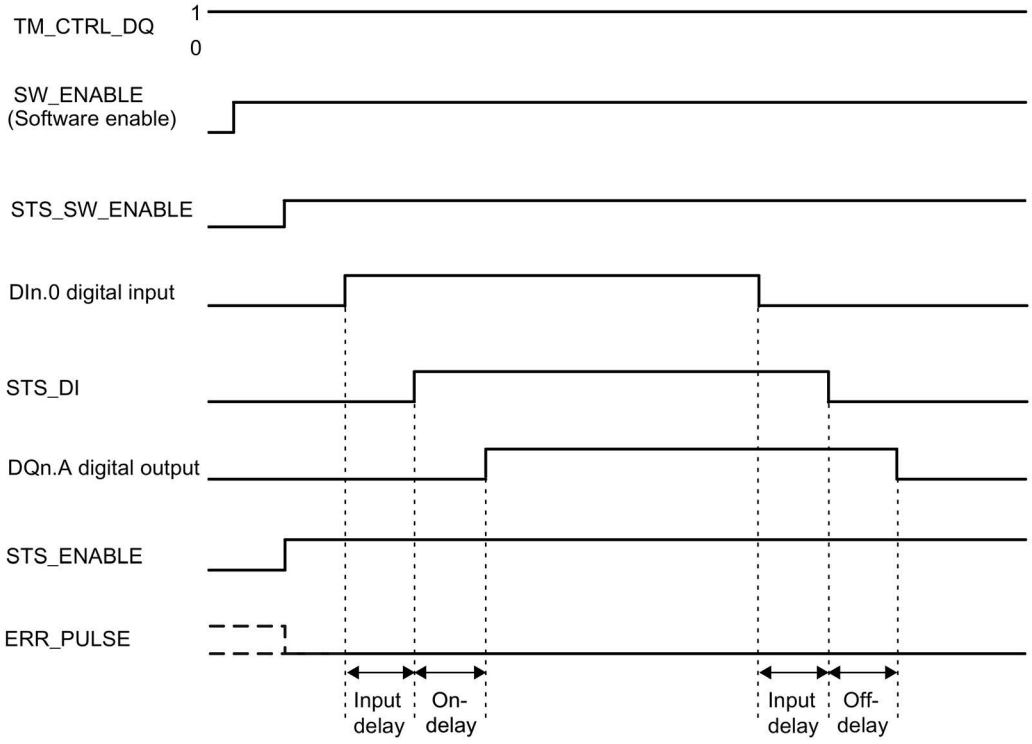


Figure 3-7 Output sequence On/Off-delay (at the start DIn.0 digital input = 0)

SW_ENABLE is set, while DIn.0 digital input = 1:

If SW_ENABLE is set while DIn.0 digital input = 1, the first edge of a digital input (falling edge) is ignored.

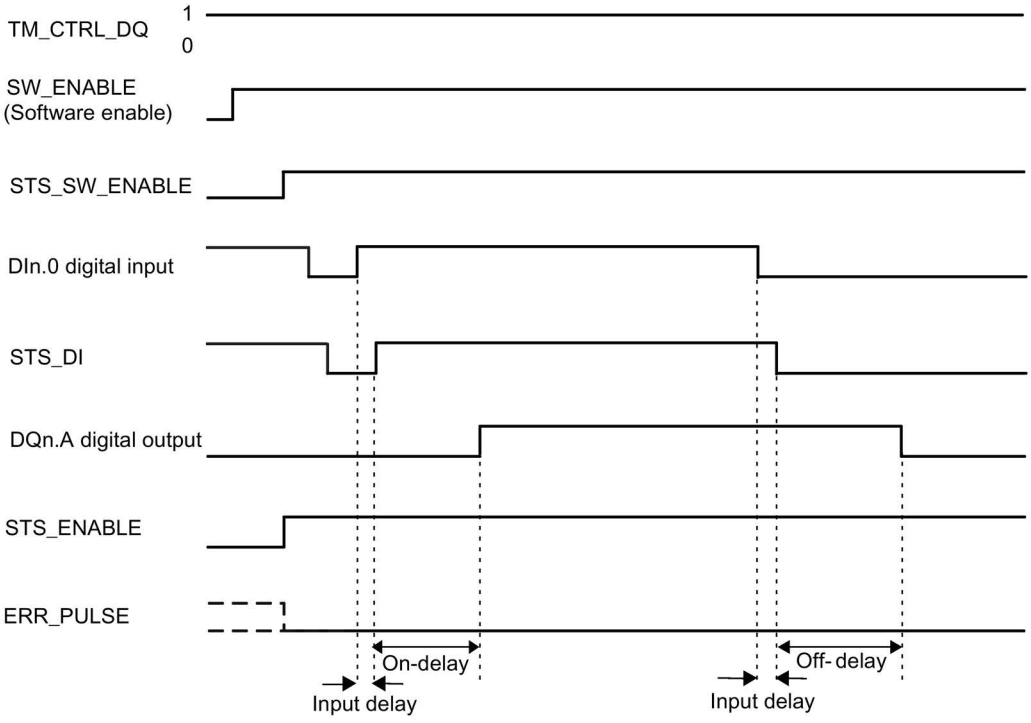


Figure 3-8 Output sequence On/Off-delay (at the start DIn.0 digital input = 1)

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates the software enable pending at the TM Pulse 2x24V.

- When the DIn.0 digital input goes high (positive edge) and remains high for the input delay time, the On-delay is started. After the On-delay expires, the DQn.A digital output is set.
- When the DIn.0 digital input goes low for the input delay time, the Off-delay is started. After the Off-delay expires, the DQn.A digital output is cleared.
- If the TM Pulse 2x24V recognizes a pulse duration or inter-pulse duration at the DIn.0 digital input shorter than the input delay time, then the input is ignored and the DQn.A digital output is unchanged.
- If the TM Pulse 2x24V recognizes a pulse duration or inter-pulse duration at the DIn.0 digital input longer than the input delay time, but too short to produce the parameterized pulse duration or inter-pulse duration, then the ERR_PULSE bit is set and the DQn.A digital output is unchanged.
- At the next edge of the DIn.0 digital input, the TM Pulse 2x24V clears the ERR_PULSE feedback bit

Note

TM_CTRL_DQ technology module output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A output.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQn.A/DQn.B outputs directly with the SET_DQA/SET_DQB control bits.

Canceling the output sequence

Disabling the software enable (SW_ENABLE = 1 → 0) during the On-delay or pulse cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A digital output are immediately reset to 0.

You must restart the output sequence to begin new pulse output.

Truth Table

Software enable SW_ENABLE	DIn.0 digital input	DQn.A digital output (when TM_CTRL_DQ = 1)	STS_ENABLE	Output sequence
1	0 → 1 and remains 1 during the input delay	0, if On-delay > 0 1, if On-delay = 0	1	Start
1	1 → 0	1, if Off-delay > 0 0, if Off-delay = 0	1	Start
0	Any status	0	0	Cancel
1	Any status	0, if On-delay is not expired or if Off-delay is expired 1, if On-delay is expired and Off-delay is not expired	1	-
0 → 1	0	0	1	-

Minimum pulse duration/minimum interpulse duration of the DQn.A digital output

The minimum pulse duration/minimum interpulse duration of the DQn.A digital output is 1.5 µs (high-speed active) and 10 µs (high-speed inactive). Note that lower values are possible but not guaranteed because this is hardware dependent.

Make sure you take this into consideration when you set the On/Off-delay and the pulse duration/interpulse duration of the DIn.0 digital input. Otherwise, the response at the DQn.A digital output is not guaranteed.

The pulse duration of the DIn.0 digital input is too short

Case 1: The DIn.0 pulse duration < input delay time: The DIn.0 pulse is filtered out and ignored (no error occurs).

Case 2: The TM Pulse 2x24V detects a DIn.0 pulse duration that is too short if: DIn.0 pulse duration + Off-delay time ≤ On-delay (error is set)

Case 3: The TM Pulse 2x24V detects a DIn.0 pulse duration that is too short if: DIn.0 pulse duration + Off-delay time + minimum DQ pulse duration < On-delay time (error is set) .

Response of the TM Pulse 2x24V to a pulse duration that is too short:

- ERR_PULSE is set.
- The current On-delay is cleared.
- The Off-delay is not started.
- The signal level at the DQn.A digital output remains at 0.

ERR_PULSE is cleared at the next positive edge on the DIn.0 digital input.

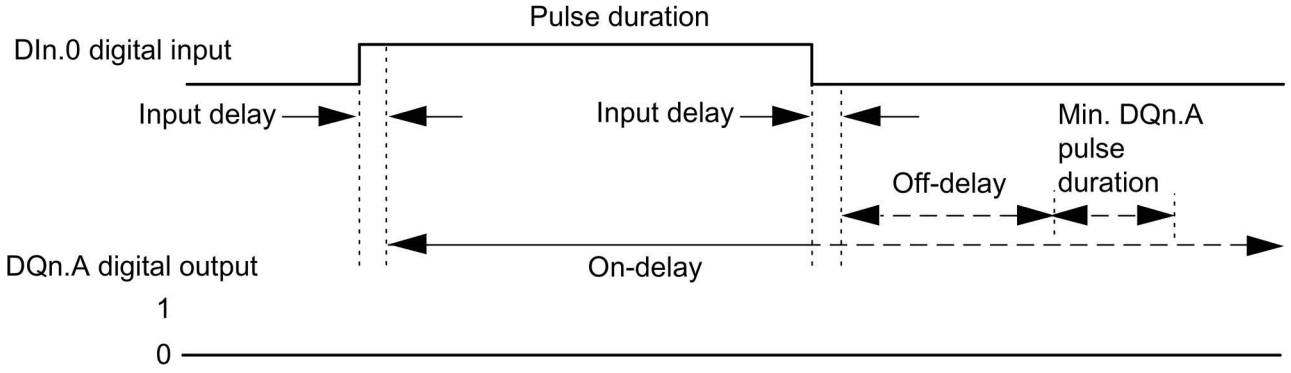


Figure 3-9 DIn.0 pulse duration too short

The interpulse duration of the DIn.0 digital input is too short

The TM Pulse 2x24V detects an interpulse duration that is too short on the positive edge on the DIn.0 digital input if:

Interpulse duration + On-delay ≤ Off-delay.

- Case 1: Interpulse duration < input delay time: The DIn.0 input interpulse is filtered out and ignored (no error occurs).
- Case 2: Interpulse duration + On-delay time ≤ Off-delay time (error is set).
- Case 3: Interpulse duration + On-delay time < Off-delay time + minimum inter-pulse duration (error is set).

Response of the TM Pulse 2x24V to an interpulse duration that is too short:

- ERR_PULSE is set.
- The current Off-delay is cleared.
- The On-delay is not started.
- The signal level at the DQn.A digital output remains at 1.

ERR_PULSE is cleared with the next negative edge on the DIn.0 digital input.

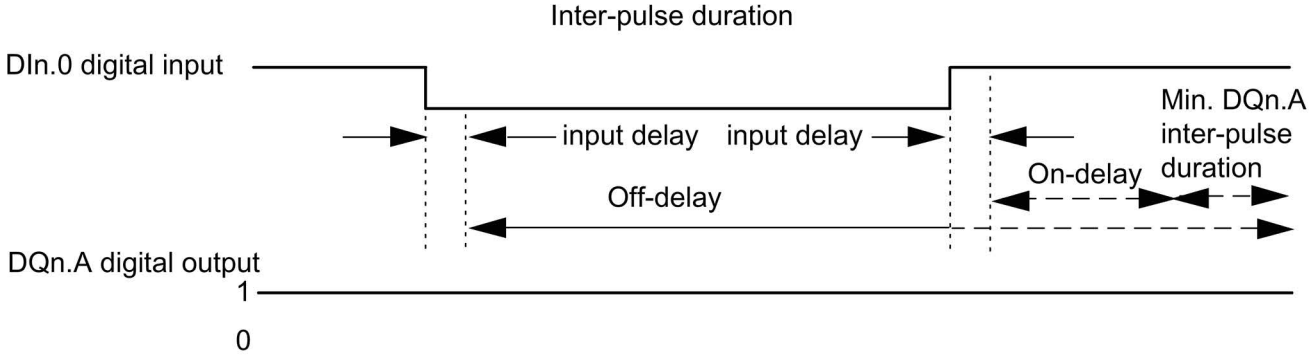


Figure 3-10 DIn.0 inter-pulse duration too short

Retriggering the current On-delay

The TM Pulse 2x24V starts a new On-delay on the positive edge on the DIn.0 digital input if:

On-delay > pulse duration + inter-pulse duration

This clears the current Off-delay.

The DQn.A digital output is only set if signal level 1 is present on the DIn.0 digital input longer than the On-delay. This enables you to filter rapid pulses.

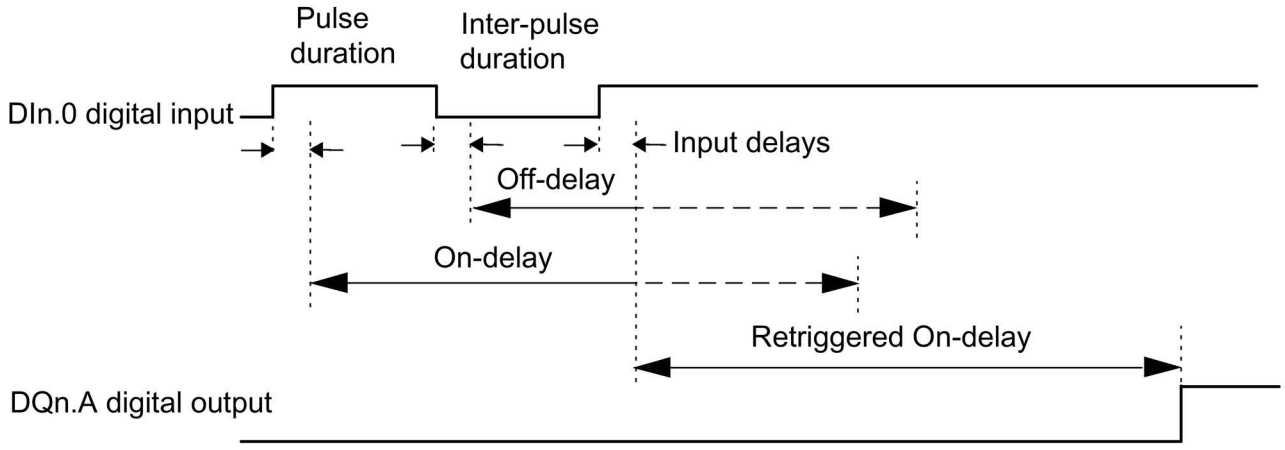


Figure 3-11 DIn.0 retriggering the current On-delay

Retriggering the current Off-delay

The TM Pulse 2x24V starts a new Off-delay on the negative edge on the DIn.0 digital input if:

Off-delay > pulse duration + inter-pulse duration

The DQn.A digital output is only cleared if signal level 0 is present on the Dn.0I digital input longer than the Off-delay.

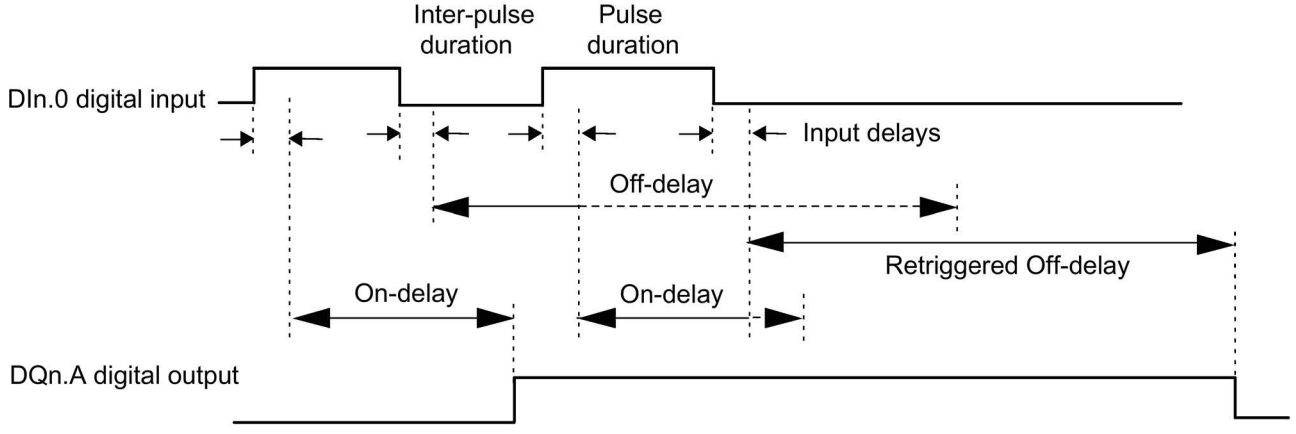


Figure 3-12 Retriggering the current Off-delay

Setting and changing the OUTPUT_VALUE (On-delay)

- You set the On-delay OUTPUT_VALUE directly using your control program to assign a value.
- The unit is always microseconds. Possible range: between 0 µs and 85,000,000 µs. If an invalid value is assigned by your program, the module will send back error ERR_OUT_VAL in the feedback interface and continue using the last valid value.
- The new On-delay value is activated with the next positive edge on the DIn.0 digital input.

Setting and changing the Off-delay

- Permanent update
  The Off-delay can be controlled permanently using the control interface. MODE_SLOT bit has to be set (permanent update); LD_SLOT must be the value 3 (for Off-delay). Set the Off-delay as a value between 0 µs and 85,000,000 µs in the field SLOT. The unit is always microseconds.
- Single update
  Set the Off-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared (single-update); LD_SLOT must be the value 3 (for Off-delay). Set the Off-delay as a value between 0 µs and 85,000,000 µs in the field SLOT.

If you change the Off-delay value during the output sequence, the new Off-delay is activated at the next negative edge on the DIn.0 digital input.
For more details about the use of the SLOT parameter, see SLOT parameter handling (Page 113).

Isochronous mode

General information is available in Function: Isochronous mode (Page 86).

Isochronous mode does not have any influence on the functionality of On/Off-delay operating mode.

Parameters for On/Off-delay operating mode

Parameter	Meaning	Value Range	Default
Mode	3 = Set the "On/Off-delay" operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC-Motor	1
High Speed Output [1]	The output supports higher frequencies.	0 = disabled 1 = enabled	Disabled
Function DI	You can only use the DIn.0 digital input as an input.	0 = Input 1 = reserved (do not use)	Input
Input Delay	DI digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
Off-delay	The time from the falling edge of the DIn.0 digital input to the falling edge of the DQn.A. You can change the Off-delay in your control program using the SLOT parameter.	0 µs to 85,000,000 µs	0 µs

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Control and feedback signals for On/Off-delay operating mode

Control interface Offset to the start address		Parameter	Meaning				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (DWord)	On-delay				
			0 µs to 85,000,000 µs				
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	The Off-delay can be changed anytime, but only takes effect on the next falling edge of the DIn.0 digital input.				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: All other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Idle-state; nothing is done with the value
			0	0	1	1	Off-delay in microseconds
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start and terminate the output sequence.			
			0	Output canceled			
			0 → 1	Starts output sequence on positive edge			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQn.A output source: Selects either CPU program or module's output sequence.			
			0	DQn.A and DQn.B are controlled by the CPU (in your program) using the SET_DQA and SET_DQB control bits.			
			1	DQn.A is controlled by the module's pulse output sequence. DQn.B is always 0 for TM_CTRL = 1.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ is cleared.			
			0	0 on DQn.A			
			1	1 on DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ is cleared and if SET_DQA is cleared.			
			0	0 on DQn.B			
			1	1 on DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).			
			0	Reset of errors is not active			
			1	Reset of errors is active			

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage.
			1	PWR is detected, but under voltage.
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface), to reset this error and be able to use the SLOT again.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT-mode).
			0	No load error pending.
			1	Load error pending: you must set the RES_ERROR (control interface) to reset this error and be able to use the SLOT again.
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	**Bit 3**	Indicates a pulse output error.
			0	No pulse output error
			1	Pulse output error
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQn.A and DQn.B manually using SET_DQA, SET_DB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B or attempt to set both DQn.A and DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Indicates that an invalid value is detected in OUTPUT_VALUE.
			0	OUTPUT_VALUE is valid
			1	OUTPUT_VALUE is not valid
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT-mode).
			0	SLOT value is valid
			0 → 1	SLOT value is not valid
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of the SLOT in "single-update" SLOT-mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running.
			0	Pulse output is not running

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	**Channel 1** [1]			
			1	Pulse output is running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 at the DQn.A digital output
			1	1 at the DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 at the DQn.B digital output
			1	1 at the DQn.B digital output
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	0 at the DIn.0 digital input
			1	1 at the DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Counts rising and falling edges of DQn.A output	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for On/Off-delay operating mode

Input and Output Signals	Meaning	Value Range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
DIn.0 Digital input	The signal of the DIn.0 digital input is output with an On/Off-delay on DQn.A digital output.	0 = no pulse 1 = pulse	3	4
Output signal				
Pulse at the DQn.A digital output	The signal of the DIn.0 digital input is output with an On/Off-delay on the DQn.A digital output.	0 = no signal 1 = signal	9	10

See also Pin assignment and load/sensor wiring (Page 91)

3.6 Frequency output mode

Definition

This mode allows you to assign a frequency value at high frequencies more precisely than PWM period and Period duration.

A square wave signal with an assigned frequency and a constant duty cycle of 50% is produced at the digital output of the TM Pulse 2x24V.

The output sequence is started after expiration of the configured On-delay on the DQn.A digital output.

Pulse diagram

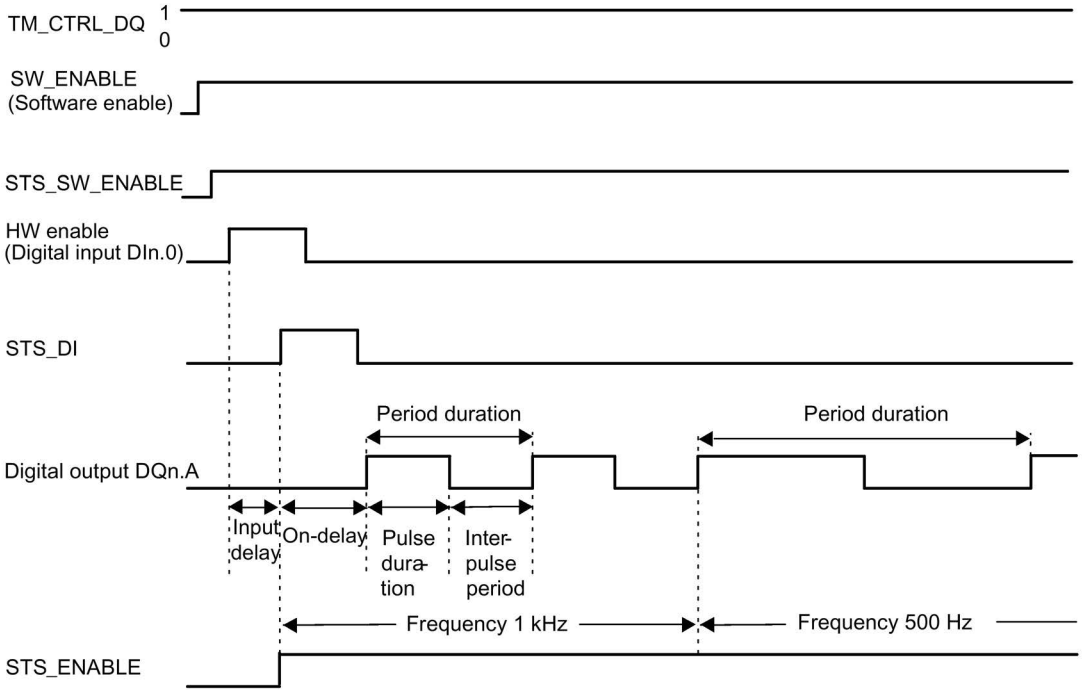


Figure 3-13 Frequency output - output sequence

The timing diagram above has the "Function DI" parameter set to "HW enable". The other option is to set "Function DI" to "Input". If the "Function DI" parameter is set to "Input", then the On-delay phase starts at the rising edge of SW_ENABLE.

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates the software enable pending at the TM Pulse 2x24V.

You can also set the DIn.0 digital input of the TM Pulse 2x24V as a HW enable with the "Function DI" parameter.

If you want to use the hardware enable, it has to be combined with the software enable. When SW_ENABLE has been enabled, the output sequence starts at the first positive edge of the hardware enable. Further positive edges of the hardware enable during the current output sequence are ignored by the TM Pulse 2x24V.

When the enable is issued (positive edge) and remains high for the time of the input delay (noise filter), the On-delay is started and the STS_ENABLE set. The frequency sequence is output on expiration of the On-delay. The output sequence runs continuously as long as SW_ENABLE is set.

Note

TM_CTRL_DQ technology module output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A output.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQn.A/DQn.B outputs directly with the SET_DQA/SET_DQB control bits.

Canceling the output sequence

Disabling the software enable (SW_ENABLE = 1 → 0) during the On-delay or frequency output cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A digital output are immediately reset to 0.

You must restart the output sequence to begin new pulse output.

Truth table

Software enable SW_ENABLE	Function DI parameter	Hardware enable (DIn.0 digital input)	DQn.A digital output (when TM_CTRL_DQ = 1)	STS_ENABLE	Output sequence
1	HW_ENABLE	0 → 1 and remains 1 during the input delay. Only active for the first positive edge, additional positive edges are ignored and no start occurs.	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0 → 1	Input	Not used	0, if On-delay > 0 1, if On-delay = 0	0 → 1	Start
0	HW_ENABLE or Input	Any status	0	0	Cancel
1	HW_ENABLE or Input	Any status	0, if On-delay is not expired or you are in the inter-pulse time 1, if On-delay is expired and during the pulse duration		-
0 → 1	HW_ENABLE	0	0	0	-

Setting and changing the output value (Frequency)

You set the OUTPUT_VALUE directly using your control program in the control interface. The value is in real format and the unit is always "Hz". The possible range is dependent on the parameter "High Speed Output" as follows:

- High Speed Output disabled
  - Frequency (OUTPUT_VALUE): 0.02 Hz to 10,000 Hz
- High Speed Output enabled
  - Frequency (OUTPUT_VALUE): 0.02 Hz to 100,000 Hz

The new frequency value is applied at the next rising edge of the output.

Output frequency accuracy

The configured output frequency is output with an accuracy of +/- 100 ppm at the DQn.A digital output.

Setting and changing the On-delay

- Permanent update
  The On-delay can be controlled permanently using the control interface. MODE_SLOT bit has to be set (permanent-update); LD_SLOT must be the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT. The unit is always microseconds.
- Single update
  Set the On-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters, the unit is always microseconds. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared (single-update); LD_SLOT must be the value 2 (for On-Delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT.

If you change the On-delay value during the output sequence, the new On-delay is activated at the next output sequence.

For more details about the use of the SLOT parameter, see Slot parameter handling (Page 113).

Isochronous mode

General information is available in the "Function: Isochronous mode (Page 86)" topic.

Isochronous mode does not have any influence on the functionality of Frequency output operating mode.

If you want to synchronize the output sequence with T_o, then set the Function DI parameter to "Input" and the frequency output sequence starts at T_o.

Parameters of Frequency output operating mode

Parameter	Meaning	Value range	Default
Mode	4 = Set the Frequency output operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC Motor	1
High Speed Output [1]	The output supports higher frequencies.	0 = disabled 1 = enabled	Disabled
Function DI	You can use the DIn.0 digital input as an input or as a hardware enable.	0 = Input 1 = HW enable	Input
Input delay	DIn.0 digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
On-delay	The time from the start of the output sequence to the output of the pulses. You can change the On-delay in your control program using SLOT parameter.	0 µs to 85,000,000 µs	0 µs

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)".

Control and feedback signals for Frequency output operating mode

Control interface Offset to the start address		**Parameter**	**Meaning**				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (Real)	DQn.A digital output frequency as a real number. The unit is always Hz. If you violate the lower or upper limit of the range, the TM Pulse 2x24V will work with the last valid value and the error ERR_OUT_VAL is activated.				
			High-speed output disabled				**High-speed output enabled**
			0.02 Hz to 10,000 Hz				0.02 Hz to 100,000 Hz
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	The On-delay can be changed before the start of the output sequence. See MODE_SLOT.				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: All other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Idle-state; nothing is done with the value
			0	0	1	0	On-delay in microseconds
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start/enable and terminate/disable the output sequence.			
			0	Output disabled/terminated			
			0 → 1	Starts output sequence on positive edge when "Function DI" = "Input"			
			1	Enable output sequence, when start is dependent on HW enable with "Function DI" = "HW enable"			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQn.A output source: Selects either CPU program or module's output sequence.			
			0	DQn.A and DQn.B are controlled by the CPU (in your program) using the SET_DQA and SET_DQB control bits.			
			1	DQn.A is controlled by the module's pulse output sequence. DQn.B is always 0.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ is cleared.			
			0	0 on DQn.A			
			1	1 on DQn.A			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ is cleared and if SET_DQA is cleared.			
			0	0 on DQn.B			
			1	1 on DQn.B			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).			
			0	Reset of errors is not active			
			1	Reset of errors is active			

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	**Channel 1[1]**			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage
			1	PWR is detected, but under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface), to reset this error.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT-mode).
			0	No load error pending
			1	Load error pending; you must set the RES_ERROR (control interface) to reset this error and be able to SLOT again.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQn.A and DQn.B manually using SET_DQA,SET_DB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B or attempt to set both DQn.A and DQn.B
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	**Bit 6**	Indicates that an invalid value is detected in OUTPUT_VALUE.
			0	OUTPUT_VALUE is valid
			1	OUTPUT_VALUE is not valid
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT-mode).
			0	SLOT value is valid
			0 → 1	SLOT value is not valid
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of the SLOT in "single-update" SLOT-mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running
			0	Pulse output is not running
			1	Pulse output is running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 at the DQn.A digital output
			1	1 at the DQn.A digital output

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	**Channel 1[1]**			
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 on DQn.B digital output
			1	1 on DQn.B digital output
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	0 on DIn.0 digital input
			1	1 on DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Sequence counter = 0. The sequence counter is not used in frequency output mode.	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for Frequency output operating mode

Input and Output Signals	Meaning	Value Range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
HW enable	You can select the HW enable with the "Function DI" parameter and select the input delay with the "Input delay" parameter. The signal of the DIn.0 digital input is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	0 = HW enable cleared 1 = HW enable issued 0 → 1 = Start of the output sequence after the input delay; dependent on the software enable (SW_ENABLE)	3	4
Output signal				
Pulse at the DQn.A digital output	A pulse is output at the DQn.A digital output for the assigned frequency.	0 = no pulse 1 = pulse	9	10

See also Pin assignment and load/sensor wiring (Page 91)

3.7 DC motor mode

Definition

Each channel has an A and a B output for connection to your DC motor load. The bipolar output switch and pulse width modulation lets your assign rotation direction and output voltage duty cycle.

You can use the two channels for 2 A per channel maximum or use a parallel connection that provides a single channel 4 A maximum.

Two channel motor load connection:

- Channel 0: motor connected between DQ0.A and DQ0.B
- Channel 1: motor connected between DQ1.A and DQ1.B

Single channel motor load connection

- 4 A Channel 0: motor connected between DQ0.A and DQ0.B.
  DQ0.A and DQ1.A are connected together; DQ0.B and DQ1.B are connected together.

For more details concerning the connection, see the Connecting chapter (Page 91)of this manual.

Motor rotate

Rotation direction

- Forward during high phase of the PWM signal: DQn.A is 1 and DQn.B is 0.
- Backward during high phase of the PWM signal: DQn.A is 0 and DQn.B is 1.
- Forward/Backward during low phase of the PWM signal: DQn.A is 0 and DQn.B is 0.

Motor stop

The motor can be stopped using an external signal connected to the DIn.0 digital input or using the signal SW_ENABLE in the control interface. Configure the "Function DI" accordingly.

Function DI setting:

- "HW enable" starts the motor on the rising edge of DI and stops the motor on the falling edge of DI.
- "External stop" starts the motor on the rising edge on SW_ENABLE and stops the motor on the rising edge on DI, or the falling edge of SW_ENABLE.
- "Input" starts the motor on the rising edge of SW_ENABLE and stops the motor on the falling edge of SW_ENABLE (DI has no effect on the control of the motor).

You control the output pulse width duty cycle with the control interface field OUTPUT_VALUE. The TM Pulse 2x24V generates continuous pulses based on this value. The OUTPUT_VALUE determines the duty cycle (pulse duration/period duration) within a period for pulse width modulation. The period duration can be adjusted.

After expiration of the assigned On-delay, the DQn.A and DQn.B output pulses begin (output sequence).

High-speed mode, current measurement, and current control are not available in DC motor mode.

Starting the output sequence

Your control program must issue the enable for the output sequence, using the software enable (SW_ENABLE 0 → 1).

The STS_SW_ENABLE feedback bit indicates the software enable pending at the TM Pulse 2x24V.

You can also assign the DIn.0 digital input of the TM Pulse 2x24V as a HW enable with the "Function DI" parameter. The input delay (noise filter) of the hardware enable can be set using the parameter "Input Delay".

If you want to use the hardware enable, it has to be combined with the software enable. When SW_ENABLE has been enabled, the output sequence starts at the first positive edge of the hardware enable. Further positive edges of the hardware enable during the current output sequence are ignored by the TM Pulse 2x24V. The hardware enable option is not supported for isochronous mode.

When the enable is issued (positive edge) and remains high for the input delay time, the On-delay is started and the STS_ENABLE set. The PWM pulse train is output on expiration of the On-delay. The output sequence runs continuously as long as SW_ENABLE is set.

Canceling the output sequence

- Using the SW_ENABLE signal: Disabling the software enable (SW_ENABLE = 1→0) cancels the current output sequence and the last period duration is not completed. STS_ENABLE and the DQn.A and DQn.B digital outputs are immediately set to 1 (motor stop).

You must restart the output sequence to begin new pulse output.

- Using the digital input DIn.0:
  - If "Function DI" is parameterized as "HW_ENABLE": a falling edge on DIn.0 will stop the output sequence with the same behavior as using the SW_ENABLE.
  - If "Function DI" is parameterized as "External stop": a rising edge on DIn.0 will stop the sequence with the same behavior as using the SW_ENABLE.

Truth table

Software enable SW_ENABLE	Function DI parameter	Hardware enable (DIn.0 digital input)	Digital output DQn.A and DQn.B	STS_ENABLE	Output sequence
1	HW_ENABLE	0 → 1 and remains 1 during the input delay. Only active for the first positive edge, additional positive edges are ignored and no start occurs.	If On-delay > 0: DQn.A: 0 DQn.B: 0	0 → 1	Start
			If On-delay = 0: Forward: DQn.A: 1 DQn.B: 0 Backward: DQn.A: 0 DQn.B: 1		
0 → 1	Input or External stop	Any state	If On-delay > 0 DQn.A: 0 DQn.B: 0	0 → 1	Start
			If On-delay = 0 Forward: DQn.A: 1 DQn.B: 0 Backward: DQn.A: 0 DQn.B: 1		
0	HW_ENABLE or Input or External stop	Any state	DQn.A: Tri-state DQn.B: Tri-state	0	Cancel
1	Input	Any state	If On-delay is not expired or during interpulse time: DQn.A: 0 DQn.B: 0		-
			If On-delay is expired and during the pulse duration: Forward: DQn.A: 1 DQn.B: 0 Backward: DQn.A: 0 DQn.B: 1		
1	HW_ENABLE	0	DQn.A: Tri-state DQn.B: Tri-state	1 → 0	Stop
1	External stop	0 → 1	DQn.A: Tri-state DQn.B: Tri-state	1 → 0	Stop
0 → 1	HW_ENABLE	0	DQn.A: Tri-state DQn.B: Tri-state	0	-

Setting and changing the pulse duty cycle and rotation direction

OUTPUT_VALUE assigns the duty cycle and the direction for the current period duration.

OUTPUT_VALUE is given as an S7 analog value, the sign gives the direction of the motor rotation ("+" means forward, "-" means backwards). The possible range is -27648 to +27648.

Only the two least significant bytes of OUTPUT_VALUE are used, the two other bytes are ignored. A value of the two least significant bytes higher than +27648 will be interpreted as a value of +27648 (100% forward). A value of the two least significant bytes below -27648 will be interpreted as a value of -27648 (100% backward).

Before changing the rotation direction, it is recommended that the OUTPUT_VALUE is first set to 0 long enough, to first stop the motor. Note that the TM Pulse 2x24V module does not override our duty cycle assignment in order to protect the motor. If a ramp-up or ramp-down is required by the motor, this ramp has to be implemented in the automation program and transmitted to the module by controlling the value of the field OUTPUT_VALUE accordingly.

- "S7 analog output" output format: Value range between -27,648 and +27,648. The sign determines the motor rotation direction.
- Pulse duration = (OUTPUT_VALUE/27,648) x period duration.

You assign OUTPUT_VALUE directly with your control program. A new OUTPUT_VALUE is applied at the next rising edge of the output.

Setting and changing the period duration

- Permanent update
  The period duration is controlled permanently using the control interface. The MODE_SLOT bit has to be set ("1" means permanent update); LD_SLOT has to have the value 1 ("1" means Period duration).
  Set the period duration value in the field SLOT between 100 µs and 85,000,000 µs.
- Single update
  Set the period duration in the configuration parameters. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared ("0" means single-update); LD_SLOT has to have the value 1 ('1' means Period duration).
  Set the period duration value in the field SLOT between 100 µs and 85,000,000 µs.

The new period duration is applied with the next rising edge of the output.

For more details about SLOT parameter handling, see "Slot parameter handling (control interface)".

Isochronous mode

General information is available in the "Function: Isochronous mode" topic.

In isochronous mode, the output sequence is synchronized with the moment T_O. The period duration is coordinated to the application cycle (the synchronous cycle, a multiple of the PROFINET cycle). The behavior in the DC-motor mode is the same than the behavior in the operating mode PWM. Please refer to the corresponding chapter in this manual. Only the differences are listed below:

- During the "pulse duration phase" (high phase), the status of the outputs is as follows:
  - In forward direction: DQn.A is 1 and DQn.B is 0
  - In backward direction: DQn.A is 0 and DQn.B is 1
- During the "interpulse duration phase" (low phase), the status of the outputs is as follows:
  - DQn.A is 0 and DQn.B is 0
- The parameter "Function DI" can be parameterized as "External stop" in isochronous mode to use the rising edges on the DIn.0 to stop the motor. Note that if "Function DI" is parameterized as "HW_ENABLE" in isochronous mode which is not supported, it will be interpreted as "External stop" by the TM Pulse 2x24V module.

Setting and changing the On-delay

- Permanent update
  The On-delay can be controlled permanently using the control interface. The MODE_SLOT bit has to be set (permanent update); LD_SLOT must have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT. The unit is always microseconds.
- Single update
  Set the On-delay as a value between 0 µs and 85,000,000 µs in the configuration parameters. The unit is always microseconds. Alternatively, perform a single update using the control interface. MODE_SLOT has to be cleared (single-update); LD_SLOT must have the value 2 (for On-delay). Set the On-delay as a value between 0 µs and 85,000,000 µs in the field SLOT.

If you change the On-delay value during the output sequence, then the new On-delay is activated at the next output sequence. For more details about the use of the SLOT parameter see SLOT parameter handling.

Parameters for DC motor operating mode

Parameter	Meaning	Value range	Default
Operating mode	5 = Set the DC motor operating mode.	0 = Pulse output 1 = Pulse width modulation 2 = Pulse train 3 = On/Off-delay 4 = Frequency output 5 = DC Motor	1
Function DI	You can use the DIn.0 digital input as an input, for HW_ENABLE, or External stop signal. The signal at DIn.0 is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay. If HW_ENABLE is used in non-isochronous mode, a rising edge on DIn.0 starts the output sequence (If SW_ENABLE is set) and a falling edge on DIn.0 stops the output sequence. If HW_ENABLE is used in isochronous mode, it will be interpreted by the module as External stop. If External Stop is used, the output sequence starts with the rising edge of SW_ENABLE and stops with the rising edge of DIn.0.	0 = Input 1 = HW enable 2 = External stop	Input
Output format	Defines the format of the ratio value (duty cycle).	0 = S7 analog format	S7 analog format
Input delay	DIn.0 digital input must be stable over the delay time (signal noise suppression).	0 = Off (4 µs) 1 = 0.05 ms 2 = 0.1 ms 3 = 0.4 ms 4 = 0.8 ms 5 = 1.6 ms 6 = 3.2 ms 7 = 12.8 ms 8 = 20 ms	0.1 ms
Period	Period duration of the output pulse cycle in µs. You can change the period duration in your control program with the control interface SLOT field.	100 µs to 85,000,000 µs	1000 µs
On-delay	The time from the start of the output sequence to the output of the pulses. You can change the On-delay in your control program with the control interface SLOT field.	0 µs to 85,000,000 us	0 µs

Control and feedback signals for DC motor mode

Control interface Offset to the start address		Parameter	Meaning				
Channel 0	**Channel 1** [1]						
Bytes 0 to 3	Bytes 12 to 15	OUTPUT_VALUE (DWord)	The OUTPUT_VALUE determines the duty cycle (pulse duration/period duration ratio) within a period (PWM). The period duration can be adjusted. The new output value is applied at the next rising edge of the output. The OUTPUT_VALUE sign indicates direction of rotation (positive for forward) and (negative for backward). S7 analog output format: value range is -27,648 to +27,648 DInt data type: Only 2 least significant bytes are used For channel 0: bytes 2 and 3 For channel 1: bytes 14 and 15				
Bytes 4 to 7	Bytes 16 to 19	SLOT (DWord)	Your program can change the On-delay and Period duration before the start of the output sequence using the SLOT and MODE_SLOT parameters.				
			0 µs to 85,000,000 µs				
Byte 8	Byte 20	LD_SLOT	Interpretation of the value SLOT: All other values not listed below are invalid and produce the error ERR_LD (in single-update mode) or ERR_SLOT_VAL (in permanent-update mode).				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
			0	0	0	0	Idle-state; nothing is done with the value
			0	0	0	1	Period in µs
			0	0	1	0	On-delay in µs
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT	**Bit 4**	Mode for use of the field SLOT.			
			0	Single-update mode			
			1	Permanent-update mode			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start and terminate the output sequence.			
			0	Output canceled			
			0 → 1	Starts output sequence on positive edge when "Function DI" = "Input" or "Function DI" = "External stop".			
			1	Enable output sequence, when start is dependent on HW enable with "Function DI" = "HW enable"			
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	The outputs are always controlled by the module in DC-motor mode. This bit is ignored.			
			X	Don't care: No effect on the outputs.			
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ = 0.			
			X	Don't care: No effect on the outputs.			
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ = 0.			
			X	Don't care: No effect on the outputs.			
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR	**Bit 0**	Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V).			
			0	Reset of errors is not active			
			1	Reset of errors is active			

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and should be 0.

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1 [1]			
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	**Bit 0**	Indicates under voltage on the Power supply. Note that the bit is not set if the voltage is not present.
			0	PWR is not under voltage
			1	PWR is detected, but under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	**Bit 1**	Indicates a short-circuit or overload on the output 24 V DC. You must set the RES_ERROR (control interface), to reset this error.
			0	No short-circuit on 24 V DC
			1	Short-circuit on 24 V DC
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	**Bit 2**	Indicates an error while loading a value using the field SLOT (only in "single-update" SLOT-mode).
			0	No load error pending
			1	Load error pending: you must set the RES_ERROR (control interface) to reset this error and be able to use SLOT again.
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	**Bit 7**	Indicates that an invalid value is detected in SLOT (only in "permanent-update" SLOT-mode).
			0	SLOT value is valid
			0 → 1	SLOT value is not valid
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT	**Bit 2**	Toggle acknowledge bit for each action of SLOT in "single-update" SLOT-mode. Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY	**Bit 4**	Indicates the module is ready and parameterized.
			0	Module is not parameterized
			1	Module is parameterized
Byte 1: Bit 5	Byte 9: Bit 5	STS_SW_ENABLE	**Bit 5**	Indicates the status of SW_ENABLE (control interface).
			0	SW_ENABLE cleared
			1	SW_ENABLE set
Byte 2: Bit 0	Byte 8: Bit 0	STS_ENABLE	**Bit 0**	Indicates an output sequence is running.
			0	Output sequence not running
			1	Output sequence running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	0 on DQn.A digital output
			1	1 on DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	0 on DQn.B digital output
			1	1 on DQn.B digital output

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	**Channel 1** [1]			
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI	**Bit 3**	Indicates the signal level at the DIn.0 digital input.
			0	0 on DIn.0 digital input
			1	1 on DIn.0 digital input
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT	Sequence counter = 0. The sequence counter is not used in PWM mode.	
Word 3	Word 7	Reserved	Read as 0	

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note: All bytes and bits not described in the table above are reserved and are read as 0.

Input and output signals for DC motor operating mode

Input and output signal	Meaning	Value range	Channel 0 BaseUnit pin number	Channel 1 BaseUnit pin number
Input signal				
HW enable /External stop Note: HW enable not supported for PWM in isochronous mode	You can select the interpretation of the digital input signal on the output sequence with the "Function DI" parameter and select the input delay with the "Input delay" parameter. The signal at the DIn.0 digital input is interpreted by the TM Pulse 2x24V, after noise filtering by the input delay, as the start of the output sequence.	Function DI = "HW_ENABLE": 0 = HW enable cleared 1 = HW enable issued 0→1 = Start of the output sequence after the input delay, dependent on the software enable (SW_ENABLE) 1→0 = Stop of the output sequence after the input delay, dependent on the software enable (SW_ENABLE). Function DI = External stop: 0→1 = Stop of the output sequence after the input delay; dependent on the software enable (SW_ENABLE)	3	4
Output signal				
Pulse at the DQn.A and DQn.B digital outputs	A pulse is output at the DQn.A and DQn.B digital outputs for the assigned duty cycle and period duration.	**No pulse or interpulse period:** DQn.A is 0 V and DQn.B is 0 V **Pulse duration in forward direction:** DQn.A is 24 V and DQn.B is 0 V **Pulse duration in backward direction:** DQn.A is 0 V and DQn.B is 24 V	9 (DQ0.A) 11 (DQ0.B)	10 (DQ1.A) 12 (DQ1.B)

See also Pin assignment and load/sensor wiring

3.8 Function: High-speed output

High-speed mode improves the signal timing of the DQ digital outputs. The switching edges have less delay, variation, jitter, and smaller rise/fall times.

High-speed mode is designed to produce more precise timing for pulse signals, but provides less maximum load current.

High-speed output is only available in dual channel operation; it is not available in single channel operation.

You can use the STEP 7 (TIA Portal) or STEP 7 hardware configuration to select "High-speed output (0.1 A)" option for each channel separately. Also, you can change the parameter assignment at runtime with your program using data record 128 (Page 137).

STEP 7 (TIA Portal) and STEP 7 assist you during parameter assignment, by disabling keyboard input for invalid parameters and range checking your value assignments. Depending on previous parameter selections, other options will be disabled. For example, if you select the single channel operation (the parallel connection of the two output channels), then the parameter options for Channel 1 and high-speed output are disabled.

High-speed output option is available for these operation modes:

- Pulse output
- PWM
- Pulse train
- On/Off-delay
- Frequency output

High-speed output is not available in DC motor mode.

High-speed output option

Pulse timing	Minimum		Maximum	
	High-speed disabled	High-speed enabled	High-speed disabled	High-speed enabled
Pulse duration	10 µs	1.5 µs	85,000,000 µs	
Period duration	100 µs	10 µs		
On-delay	0 µs			
Off-delay				
Frequency	0.02 Hz		10 kHz	100 kHz

High-speed output load current

Parallel mode (single channel operation)	Maximum pulse output load current	
	High-speed disabled	High-speed enabled
Disabled	2 A (Two channels)	100 mA (Two channels)
Enabled	4 A (One channel)	Not allowed

3.9 Function: Sequence counter

The TM Pulse 2x24V has a sequence counter for each channel that counts completed output sequences. Successfully completed and unsuccessfully completed output sequences are counted.
You can monitor the completion of an output sequence with the sequence counter SEQ_CNT variable in the feedback interface (Page 115).

The counter has a width of 4 bits. After a count overflow, the counter jumps back to 0.

Function of the sequence counter

The sequence counter has the following function in the individual operating modes:

- Pulse output and pulse train
  - Function DI configuration is set to "Input": the counter is set to 1 after completion of the output sequence (range 0 to 1).
  - Function DI configuration is set to "HW enable": the counter is incremented after every completed output sequence (range 0 to 15).
- On/Off-delay
  - The counter is incremented, with each edge (positive and negative) at the DQ output (range 0 to 15).
- PWM, frequency output, and DC motor
  - The counter does not have any function.

When the software enable (SW_ENABLE = 0) occurs, the counter is reset to 0.

Application options

The sequence counter can be used for:

- Detecting (counting) very short pulse sequences
- Counting of output sequences controlled with the hardware enable

3.10 Function: Current measurement

Principle of operation

Your program logic can use load current measurements with a control loop for proportional control of the energy transferred to an inductive or resistive load. The current measurements are provided in the feedback interface (Page 115) MEASURED_CURRENT value in SIMATIC S7 analog value format.

Current measurement is possible:

- For PWM and Pulse train operating modes and if the high-speed output option is inactive
  - When there is no pulse output (at STS_ENABLE = 0), 7FFFH is supplied as the measured value.
  - The measured value is valid at (STS_ENABLE = 1) after the first period duration. The returned current measurement value is a mean value of measured values sampled over the duration of at least one period. If Dithering is active, then the mean value is sampled over the duration of the entire Dither period.

Current measurement is not possible and returns a MEASURED_CURRENT value of 0:

- For all other operating modes or if the high-speed output option is active.

Note

In order for the current measurement to work correctly, do not connect a freewheeling diode (snubber/suppressor diode) to the output load.

Measuring range and measured value

- **Channel configuration is 2 channels (2 A)** (the parallel channel connection is deactivated).
  - A measured mean current of 2 A corresponds to the SIMATIC S7 analog value of 27,648 ($6C00_H$).
  - Measurements of current are possible up to a SIMATIC S7 analog value of 32,511 ($7EFF_H$) which corresponds to a current of 2.37 A.
  - Currents exceeding 2 A may only occur briefly.
- **Channel configuration is 1 channel (4 A)** (the parallel channel connection is activated).
  - A measured mean current of 4 A corresponds to the SIMATIC S7 analog value of 27,648.
  - Measurements of current are possible up to a SIMATIC S7 analog value of 32,511 which corresponds to a current of 4.74 A.
  - Currents exceeding 4 A may only occur briefly.

Output current limit diagnostic message

If diagnostics are enabled and the module is in PWM or PTO mode, then an over current diagnostic error will be reported when the module senses the current to be higher 2.37 A (4.74 A in single channel mode).

Current measurement accuracy

Current measurement accuracy is ±2% of the full scale 27,648 measuring range

- ±40 mA in dual channel mode
- ±80 mA in single channel mode

The current measurement accuracy degrades above 3 kHz pulse frequency to -2% +5% accuracy, when driving resistive loads at a frequency of 10 kHz.

Note

2% current measurement accuracy is only possible, if the period duration is not changed during the measurement process.

3.11 Function: Current control

Current control (PWM mode only)

In PWM mode, the TM Pulse 2x24V module can use a proportional-integral-derivative (PID) algorithm to control output current. The target value of the current (set point) is determined by your program and the module controls the duty cycle of the PWM output to follow the set point, with a response that is based on the assigned PID parameters.

The set point and the measured value of the load current are compared by the PID controller. If the resulting difference (error) is outside a symmetrical dead band, a reaction is calculated by adding the selected proportional, integral and/or derivative parts. The manipulated value is limited by the defined high/low limits and finally the output pulse duration time (duty cycle) is modified, depending on the PWM period duration.

Assign the reference current value to the current measured in the channel's DQn.A output load when the output is continuously in the high (On) state.

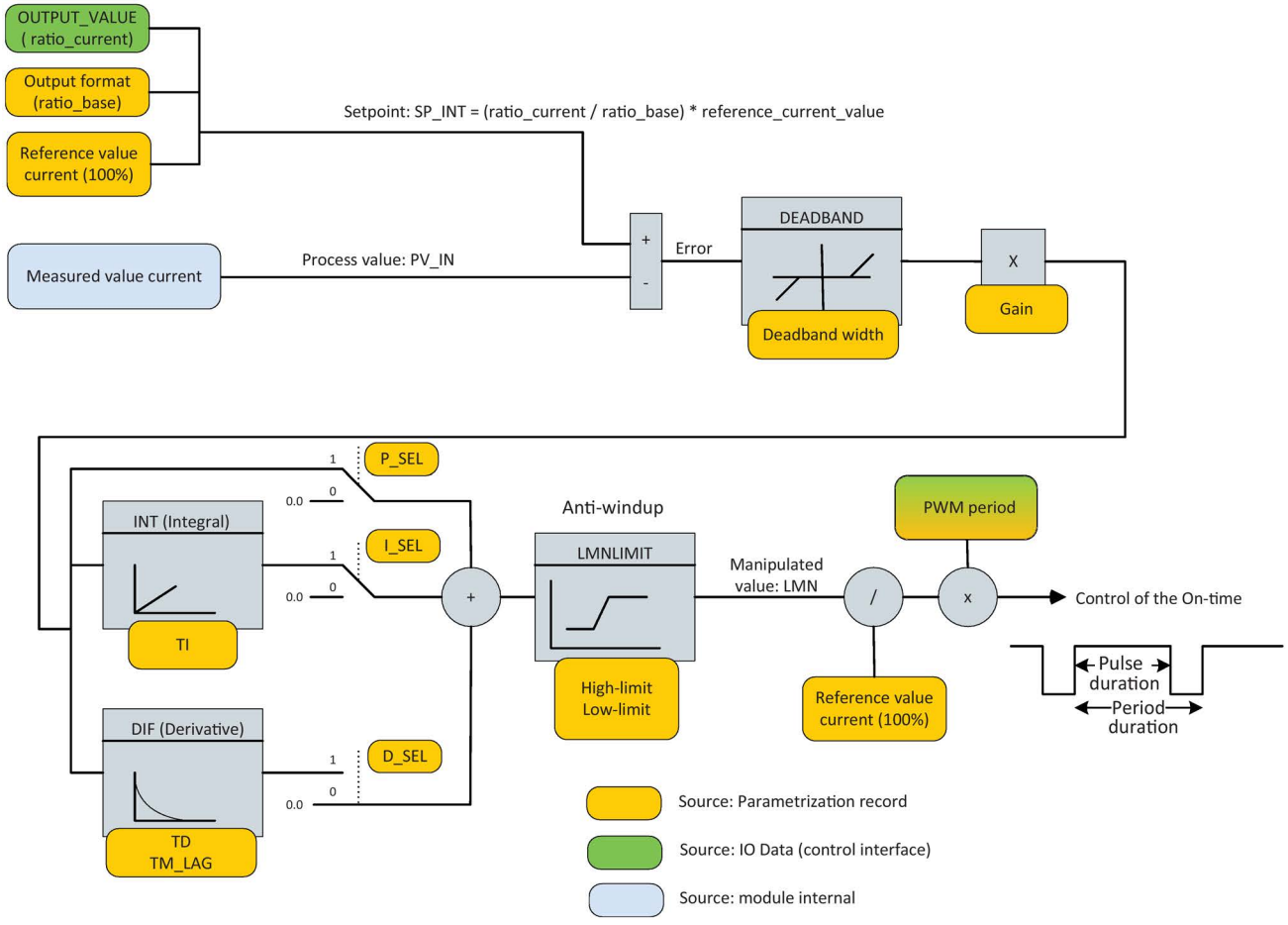


Figure 3-14 Current control PID function

For more information about this PID control method, refer to the CONT_C instruction topic in TIA Portal online help.

Setting up current control

In order to use the current control feature, the module must be parametrized correctly and the set point (target value of the current) must to be controlled by the user program. In addition, your program can also read the limit-reached flags.

PID parameters

The PID parameters can be set in TIA Portal device configuration or by your program writing parametrization record (record 128) in the module. See the parametrization record description (Page 137) for details.

- Current Control: "1" activates the PID controller. Limitations: Current control can only be activated if the selected mode of operation is PWM and the high-speed output is disabled.
  All the following parameters have no effect, if current control is disabled.
- P_Sel, I_Sel, D_Sel: defines the PID control calculation. The proportional, integral, and derivative parts can be enabled ("1") or disabled ("0") in the device configuration.
- Reference value current (mA): The reference value is used to define the maximum set point and the high and low limits of the controlled current. Typically, the maximum current can be measured in PWM mode with current control disabled and a duty cycle set to 100%. The value measured can be set as a reference for the current control. The maximum value is 4000 mA for single channel operation (parallel channel connection enabled) and 2000 mA per channel for dual channel operation (parallel connection disabled).
- Dead band width (µA): A dead band is applied to the output current deviation from the set point current. The "Dead band width" is half of the size of the dead band. The dead band is symmetrical: For example, if the dead band width = 1500 µA, then the dead band is from -1500 µA to +1500 µA .
- High limit (S7 analog value): The manipulated value is always restricted to a high limit and low limit. The "High limit" parameter assigns the high limit in S7 analog format, relative to the reference value current. A value higher or equal than 27648 means 100% of the reference value current. The High limit value must be higher than the Low limit.

  High limit example:

  – The reference current is set to 1000 mA which should be the load current measured with 100% duty cycle.
  – If the high limit is 20000, this means the maximum current the controller will drive is 20000/27648 * 1000 mA = 723.4 mA that is equivalent to a duty cycle of 72.34%.
  – If the controller calculates a control value higher than 72.34%, it will drive 72.34% and never go higher than that value.

- **Low limit (S7 analog value):** The manipulated value is always restricted to a high limit and low limit. The "Low limit" parameter assigns the low limit in S7 analog format, relative to the reference value of the current. A value higher or equal to 27648 is not allowed. The Low limit value must be lower than the High limit.
  Low limit example:
  - The reference current is set to 1000 mA which should be the load current measured with 100% duty cycle.
  - If the low limit is 100, this means the minimum current the controller will drive is 100/27648 * 1000 mA = 3.6 mA that is equivalent to a duty cycle of 0.36%.
  - If the controller calculates a control value below 0.36%, it will drive 0.36% and never go lower than that value.
- **Gain:** The proportional gain parameter assigns the amplification factor for the P part of the PID algorithm.
- **TI (s)**: The integration time parameter determines the time interval of the integral action. If TI is smaller than the controller cycle time, TI will be set internally to the controller cycle time.
- **TD (s)**: The derivative time parameter determines the time interval of the derivative action. If TD is smaller than the controller cycle time, TI will be set internally to the controller cycle time.
- **TM_LAG (s)**: Time lag of the derivative action. The algorithm of the D-action contains a delay of TM_LAG. If TM_LAG is smaller than half the controller cycle time, TI will be set internally to half the controller cycle time.

Set point control

Your program controls the target value of the output current by setting the control interface field OUTPUT_VALUE. The output format is selected in the parametrization record (see parameter "output format"). For example, if the output format "per 100" is selected, writing the value 60 in OUTPUT_VALUE means the target current value is 60% of the reference current value (see parameter "reference value current").

Limit reached flags

The limit reached flags are available in the feedback interface.

QLMN_HLM: "1" means the manipulated value is at the high limit.

QLMN_LLM: "1" means the manipulated value is at the low limit.

For more information about the PID control parameters, refer to the CONT_C instruction topic in TIA portal online help.

Controller cycle time

The internal controller cycle time depends on the configured automation system and on the PWM period. The controller cannot control the current faster than the parametrized PWM period, as the measured current is averaged over an entire PWM period.

If Dithering is active simultaneously with current control, the PID controller uses the Dither period duration as the internal controller cycle time.

Resetting the PID current controller

The internal data of the current controller are reset in the following cases:

- SW_ENABLE (see control interface) is low.
- A new parametrization record is sent to the module.

See also

TM Pulse 2x24V feedback interface (Page 115)

3.12 Function: Dither PWM output

Dither overview

The Dither function creates a vibration in a proportional valve when the desired valve position is controlled with current supplied from the PWM output. The vibration is induced by superimposing the dither current fluctuation around the target current, in a PWM output load (valve coil).

The dither vibration improves the accuracy and linearity of proportional control valves. You can enable and configure the Dither function to minimize valve control problems caused by static friction, stiction, and hysteresis.

Dither parameters

The dither parameters can be set in the TIA Portal device configuration or in the parameterization record (record 128) sent by the program to the module. See the Parameterization record (Page 137) description for details. Some dither parameters can be changed during operation by using the slot mechanism in the control interface for calibration purposes. It is recommended to restart the PWM output sequence after such a parameter change.

- **Dither**: "1" enables the dither feature. In addition to enabling "Dither" in the module's PWM configuration, you must set the DITHER bit in the Control interface (Page 110) to start the Dither signal.
- **Dither amplitude (‰)**: Assign the amplitude ratio of the superimposed dither signal in a per mill ratio. The allowed range is 0 to 500‰ duty cycle. If a higher value is assigned, then 500‰ is used by the module.

  For example: if the dither amplitude is set to 100‰ and the duty cycle in PWM mode is set to 50%, then the effective duty cycle of the signal will vary periodically between 40% and 60%.

  The dither amplitude is adapted (reduced) dynamically by the module if the calculated effective duty cycle is higher than 100% or lower than 0%, so that the dither signal remains symmetrical.

  For example, if the dither amplitude is set to 100‰ and the duty cycle in PWM mode is set to 95%, the effective duty cycle of the signal will vary periodically between 90% and 100%. The dither amplitude is always corrected to remain symmetrical. The dither amplitude will return to the assigned value, as soon as the output signal allows enough duty cycle margin for the superimposed Dither duty cycle variation to be symmetrical.

- **Dither period (µs)**: Assign the period duration in microseconds for the superimposed dither signal. The allowed range is from (4 x PWM period) to 100,000 µs. Also, the Dither period must be greater than 2 ms. If a value less than 2 ms or less than (4 x PWM period) is parametrized, then an error occurs (see parameter validation (Page 120) for ERR_SLOT_VAL and ERR_LD). If a Dither period higher than 100 ms is assigned, the module will use the value 100 ms. The Dither period used by the module can only be an even multiple of the PWM period. The module will use the nearest possible value to the assigned Dither period.
- **Dither ramps**: The dither ramp parameter is one double-word made of two different words. The low word is the dither ramp-up time and the high word is the dither ramp-down time.
  - **Dither ramp-up time (ms):** Assign the time in milliseconds for the duty cycle to rise from 0% to 100%. The allowed range is 0 ms to 30,000 ms. If a higher value is assigned, then 30,000 ms is used by the module.
    The effective ramp-up time is dependent on the dither amplitude and the nominal value of the duty cycle.
    For example: if your nominal value of the duty cycle is 50% and the dither amplitude is 100‰, the effective duty cycle will vary between 40% and 60%, the ramp-up time from 50% to 60% will be 100‰ x Dither ramp-up time.
  - **Dither ramp-down time (ms):** Assign the time in milliseconds for the duty cycle to fall from 100% to 0%. The allowed range is 0 ms to 30,000 ms. If a higher value is assigned, then 30,000 ms is used by the module.
    The effective ramp-down time is dependent on the Dither amplitude and the nominal value of the duty cycle.
    For example: if your nominal value of the duty cycle is 50% and the dither amplitude is 100‰, the effective duty cycle will vary between 40% and 60%, the ramp-down time from 50% to 40% will be 100‰ x Dither ramp-down time.

Note

Changing parameters during Dither output

If the PWM period is changed during operation using the slot mechanism, so that the dither period is less than (4 x PWM period), then dither current is deactivated and feedback bit STS_DITHER = 0, until a valid combination of Dither and PWM periods is assigned.

If you change parameter values using the SLOT mechanism during dither output, then you should restart the output sequence.

Note

PWM with dithering in isochronous mode

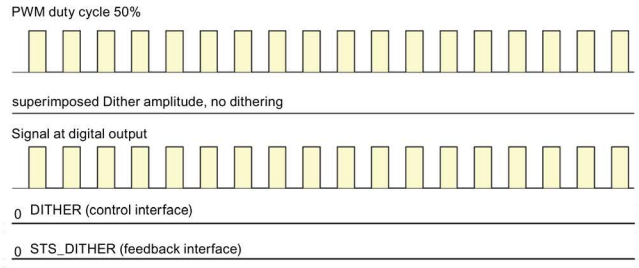
For best results, set the dither period duration to an integral even-numbered multiple of the application cycle.

Example: If the application cycle is 1ms, then the minimal optimal dither period is 2ms. The next optimal values are 4ms, 6ms, ...

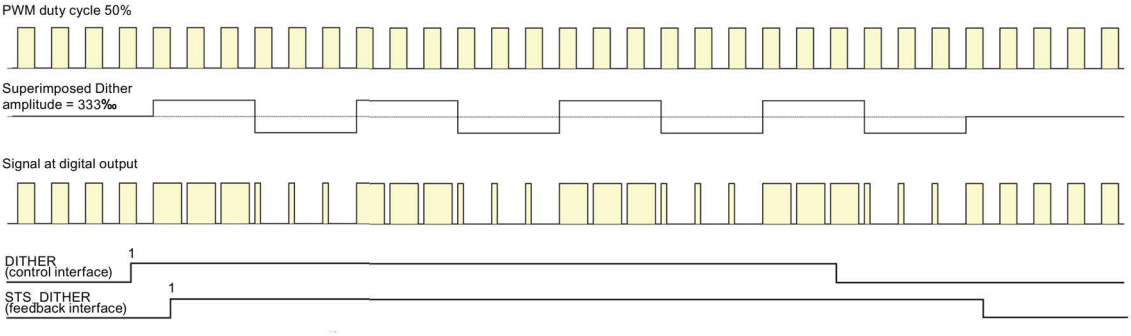
Dither starting and stopping

The ramp-up of the dither current starts as soon as the bit DITHER is set in the control interface and the current Dither period has ended and a new Dither period starts. The feedback interface provides an acknowledge bit STS_DITHER in the feedback interface that goes high when the ramp-up phase is started (by setting the DITHER bit when the output sequence is running) and goes low when the ramp-down phase is over or the output sequence is stopped. The ramp-down starts when the bit DITHER is cleared in the control interface and the current Dither period has ended and a new Dither period starts.

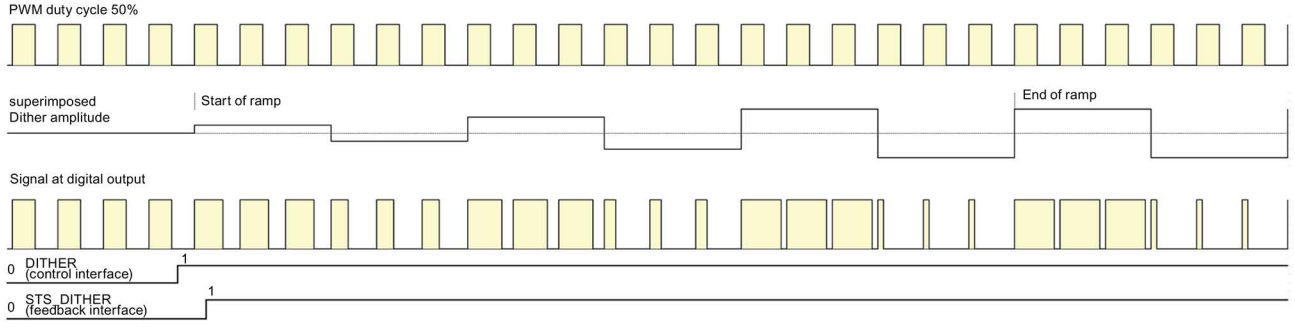
Example 1: No dither (Dither bit = 0)



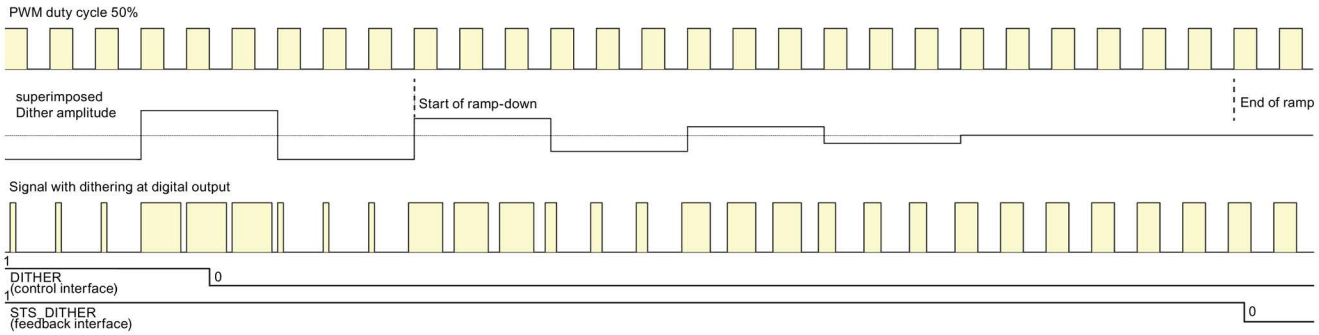
Example 2: Dither with no ramp (Dither period = 6 x PWM period)



Example 3: Dither with ramp-up



Example 4: Dither with ramp-down



3.13 Function: Isochronous mode

Note

For basic information on isochronous mode, refer to the SIMATIC PROFINET with STEP 7 (https://support.industry.siemens.com/cs/mdm/49948856?c=73850691339&t=1&s=PROFINET in STEP 7&lc=en-US) manual.

Requirements

You will require the following for the TM Pulse 2x24V in isochronous mode:

- A CPU that supports isochronous mode
- An IM (interface module) that supports isochronous mode
- Engineering software, such as TIA portal or STEP 7, to parameterize isochronous mode

Response of the TM Pulse 2x24V

Depending on the system parameter assignment, the TM Pulse 2x24V works in either non-isochronous or isochronous mode.

In isochronous mode

- The output sequences are started at the moment T_O, when only the software enable is used.
- Data communication between the PROFINET controller and TM Pulse 2x24V is isochronous to the cycle.
- All 12 bytes of a channel's control interface are consistent (24 bytes for both channels).
- All 8 bytes of a channel's feedback interface are consistent (16 bytes for both channels).
- In the PWM operating mode, the period duration is synchronized to the application cycle time (PROFINET). See the PWM chapter (Page 25) for details.

3.14 Function: Direct control of DQ digital outputs

Definition

You can directly set the TM Pulse 2x24V DQ digital outputs with your control program. Select the DQ direct control function, by clearing the Technology Module output control bit (TM_CTRL_DQ = 0), in the control interface.

Direct control of a digital output can support you when commissioning an automation control system.

If you select direct control of the DQ during a pulse output sequence, the sequence will continue to run in the background, so that when the module gets control again (by setting TM_CTRL_DQ = 1), the output sequence continues.

You assign the state of a digital output DQn.A and DQn.B with the SET_DQA and SET_DQB control bits.

You cannot use the Direct control function to set both DQn.A and DQn.B outputs high on the same channel. If an attempt to do so occurs, then error ERR_DQB is set in the feedback interface and only the DQn.A output is set high.

When you set TM_CTRL_DQ = 1, you deselect the direct control of digital output function. If the output sequence is still running (STS_ENABLE still active), then the TM Pulse 2x24V module regains control of a channel's DQn.A and DQn.B outputs.

Note

TM_CTRL_DQ technology module output control signal

- If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQn.A and DQn.B outputs. DQn.B is always 0 except in DC motor mode.
- If TM_CTRL_DQ = 0, then the CPU has control and your program can set DQ outputs directly with the SET_DQA/SET_DQB control bits. In DC motor mode, it is not possible to control the DQn.A and DQn.B outputs manually; TM_CTRL_DQ, SET_DQA, and SET_DQB have no effect.

Pulse diagram

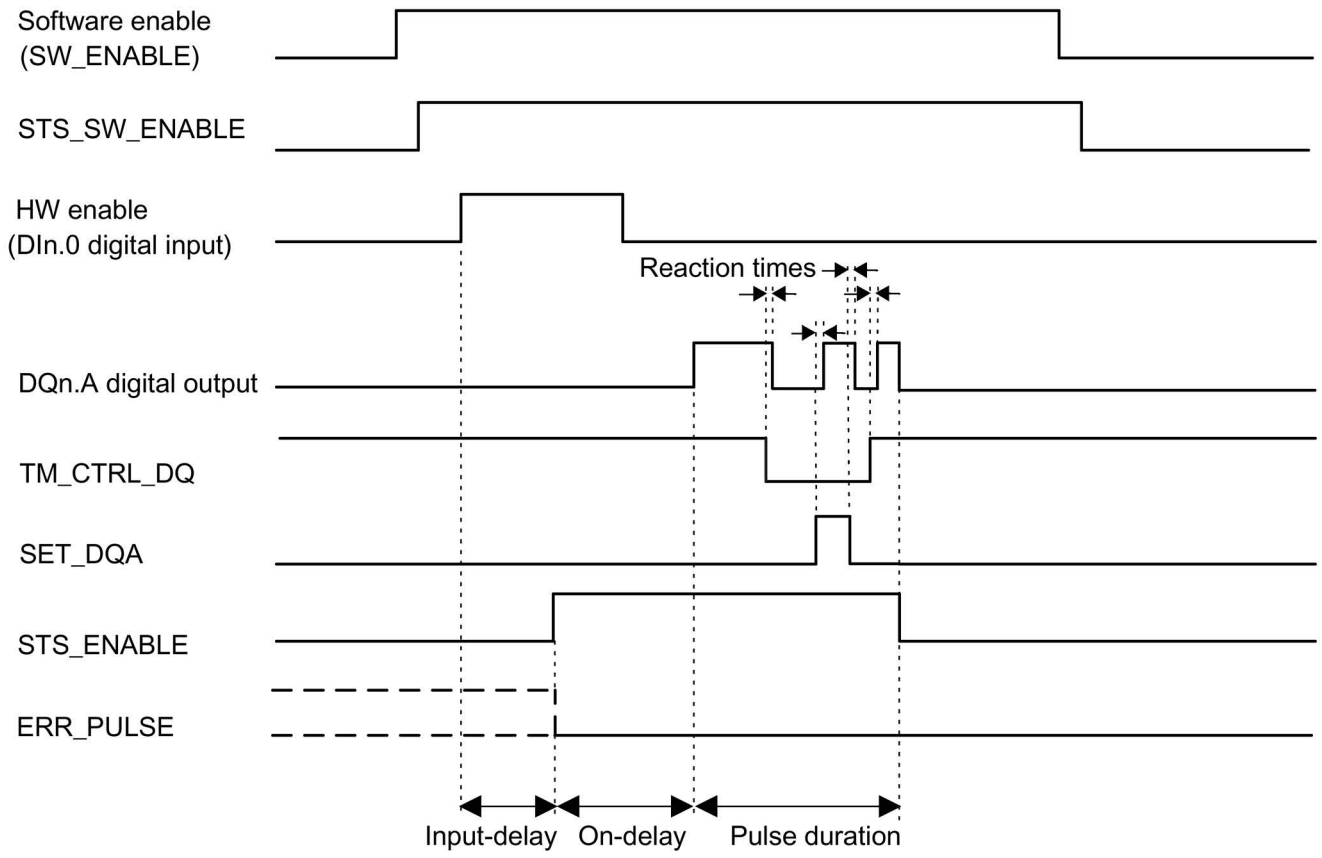


Figure 3-15 Direct control of DQ timing during Pulse output mode

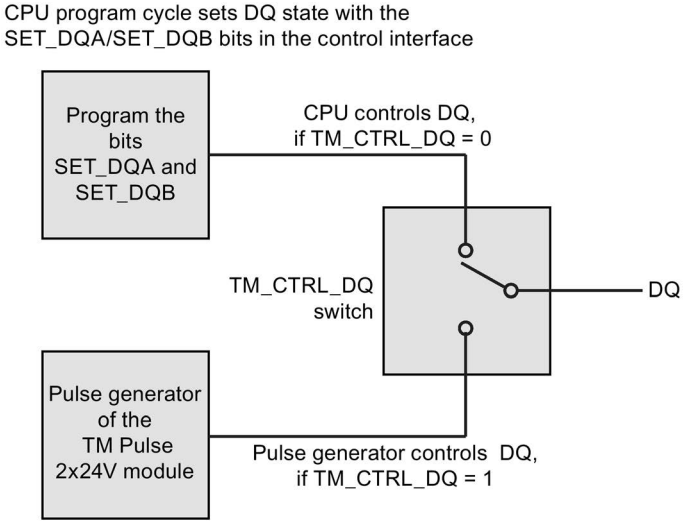


Figure 3-16 TM_CTRL_DQ output switch

Control and feedback signals

Control interface Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1			
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE	**Bit 0**	Software enable: Start and terminate the output sequence.
			0	Output canceled.
			0 → 1 1	Starts output sequence on positive edge; may be dependent on the hardware enable.
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ	**Bit 1**	Set DQ output source: Selects either PLC program or module's output sequence.
			0	DQn.A and DQn.B are controlled by the PLC program using SET_DQA and SET_DQB.
			1	DQn.A is controlled by the module's pulse output sequence.
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQA	**Bit 3**	Controls the value of the digital output DQn.A, if TM_CTRL_DQ is cleared.
			0	0V on DQn.A
			1	24V on DQn.A
Byte 9: Bit 4	Byte 21: Bit 4	SET_DQB	**Bit 4**	Controls the value of the digital output DQn.B, if TM_CTRL_DQ is cleared and if SET_DQA is cleared.
			0	0V on DQn.B
			1	24V on DQn.B

Feedback interface: Offset to the start address		Parameter	Meaning	
Channel 0	Channel 1			
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	**Bit 4**	Indicates a short-circuit on the output DQn.A. You must set RES_ERROR (control interface) to reset this error.
			0	No short- circuit on DQn.A
			1	Short-circuit on DQn.A
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	**Bit 5**	Indicates a short-circuit on the output DQn.B or an attempt to set both DQn.A and DQn.B manually using SET_DQA,SET_DB, and TM_CTRL_DQ. You must set RES_ERROR (control interface) to reset this error.
			0	No short-circuit on DQn.B
			1	Short-circuit on DQn.B or attempt to set both DQn.A and DQn.B
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA	**Bit 1**	Indicates the signal level at the DQn.A digital output.
			0	Signal 0 at the DQn.A digital output
			1	Signal 1 at the DQn.A digital output
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB	**Bit 2**	Indicates the signal level at the DQn.B digital output.
			0	Signal 0 at the DQn.B digital output
			1	Signal 1 at the DQn.B digital output

States of DQ bits				
TM_CTRL_DQ	**SET_DQA**	**SET_DQB**	**Reaction at DQn.A**	**Reaction at DQn.B**
0	0	0	0	0
0	1	0	1	0
0	1	1	1	0 (ERR_DQB is set)
0	0	1	0	1
1	Don't care	Don't care	State controlled by pulse processing	State controlled by pulse processing

Note

Setting both DQn.A and DQn.B to the 1 state not allowed

You cannot set both DQn.A and DQn.B to 1 at the same time. If SET_DQA and SET_DQB are set high and TM_CTRL_DQ is low, only output DQn.A will go high and an error bit ERR_DQB is set in the feedback interface. This error must be acknowledged in the control interface by using the RES_ERROR bit.

Connecting

4

4.1 Pin assignment, sensor, load, and power wiring

The TM Pulse 2x24V must use the B1 type BaseUnit.

Digital inputs, digital outputs, 24 V DC sensor power outputs and an external 24 V DC power source are connected to the BaseUnit of the technology module.

BaseUnit

The BaseUnit is not included in the TM Pulse 2x24V product package and must be ordered separately.

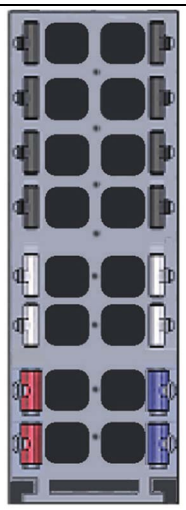
For an overview of the BaseUnits to be used with the technology module, refer to the product information on the documentation for the ET 200SP Distributed I/O System (http://support.automation.siemens.com/WW/view/en/73021864).

You can find information about selecting a suitable BaseUnit in the ET 200SP Distributed I/O System (http://support.automation.siemens.com/WW/view/en/58649293) system manual and ET 200SP BaseUnits (http://support.automation.siemens.com/WW/view/en/58532597/133300) device manual.

Pin assignment of the BaseUnit

The table below shows the pin assignment, using the BaseUnit BU20-P12+A0+4B (6ES7193-6BP20-0BB1). For this BaseUnit, the TM Pulse module's L+ pins are always isolated from adjacent modules. The L+ voltage from adjacent modules are connected together through a bypass in the BaseUnit.

Table 4- 1 Pin assignment of the BaseUnit BU20-P12+A0+4B

Designation	Pin name		View	Pin name		Designation
24 V DC supply output for sensor power	24VDC	1		2	24VDC	24VDC supply output for sensor power
Channel 0 digital input	DI0.0	3		4	DI1.0	Channel 1 digital input
Channel 0 ground for digital inputs	M	5		6	M	Channel 1 ground for digital inputs
Channel 0 ground for digital outputs	M	7		8	M	Channel1 ground for digital outputs
Channel 0 digital output A	DQ0.A	9		10	DQ1.A	Channel 1 digital output A
Channel 0 digital output B	DQ0.B	11		12	DQ1.B	Channel 1 digital output B
External 24VDC supply input for digital output and sensor power	L+	13		14	M	Ground for supply voltage
External 24VDC supply input for digital output and sensor power. Pins 13 and 15 are connected internally.	L+	15		16	M	Ground for supply voltage Pins 14 and 16 are connected internally.

24 V DC sensor power output

To power digital input sensors, the technology module supplies 24 V DC with reference to M. The 24 V DC supply is monitored for short-circuits and overload conditions.

L+ external power supply

Connect an external 24 V DC power to the L+ and M connections to supply power for the TM Pulse 2x24V module, output loads, and sensors. An internal protection circuit protects the technology module against damage due to reversed polarity of the supply voltage.

Unexpected conditions can occur at the digital outputs when L+ is connected to the supply voltage and M is disconnected from the supply voltage return, due to a wire break. The technology module monitors the connection of the supply voltage.

⚠ WARNING

Supply voltage M connections

Connect both the M potential pins to the power supply return with separate wires. If one wire breaks, then the other wire maintains the electrical connection from M to the power supply return

If the electrical connection between the M potential and the power supply return is broken, then unexpected conditions can occur and the digital outputs may go high even though your program is not setting a high state.

Note

L+ and M isolation on TM Pulse 2x24V module using BaseUnit type B1

The L+ and M connections on the type B1 BaseUnit are electrically isolated, from adjacent BaseUnits plugged in on the left-side or right-side. The L+ and M power bus passes through the type B1 BaseUnit (with no connections) and extends the power bus to connect left-side and right-side BaseUnits.

DI0.0 and DI1.0 digital inputs

The digital inputs are not electrically isolated from each other or from the digital outputs. The digital inputs are electrically isolated from the ET 200SP system bus.

When you connect input signals, depending on the configured input delay and the potential effect of interference, ground the shield on both ends of a cable that connects sensor to BaseUnit pin.

Note

Electromagnetic interference shielding for inputs

Input connections on the type B1 BaseUnit used by the TM Pulse 2x24V module do not have shield ground connections. You must connect cable shields to electrical ground at the DIN rail or the system cabinet.

Input noise filter for digital inputs

You can configure an input delay for each digital input to suppress interference. Signals must have a steady state during the configured input delay before a signal is accepted as a valid input state.

You can assign the following values for the input delay:

- Off (means input delay of 4 µs and requires a minimum pulse width of 3 µs)
- 0.05 ms
- 0.1 ms (default)
- 0.4 ms
- 0.8 ms
- 1.6 ms
- 3.2 ms
- 12.8 ms
- 20 ms

The input delay that you assign affects the detection time of input signals. The detected state change is always offset in time by the assigned input delay time.

Note

For input delay settings of "Off" or "0.05 ms" and longer wire lengths, use shielded cables at digital input connections. Shielding improves input response accuracy.

Digital outputs of channel 0 (DQ0.A, DQ0.B) and channel 1 (DQ1.A, DQ1.B)

- The digital outputs are not electrically isolated from each other or from the digital inputs. The digital outputs are electrically isolated from the ET 200SP system bus.
- The digital outputs are protected from overload and short-circuit.
- DQ0.B and DQ1.B are only used in DC motor mode or directly controlled by a SET_DQB bit in the control interface.
- Digital outputs have integrated protection diodes to prevent voltage overstress due to inductive kickback. No external protection diodes are required for inductive loads.

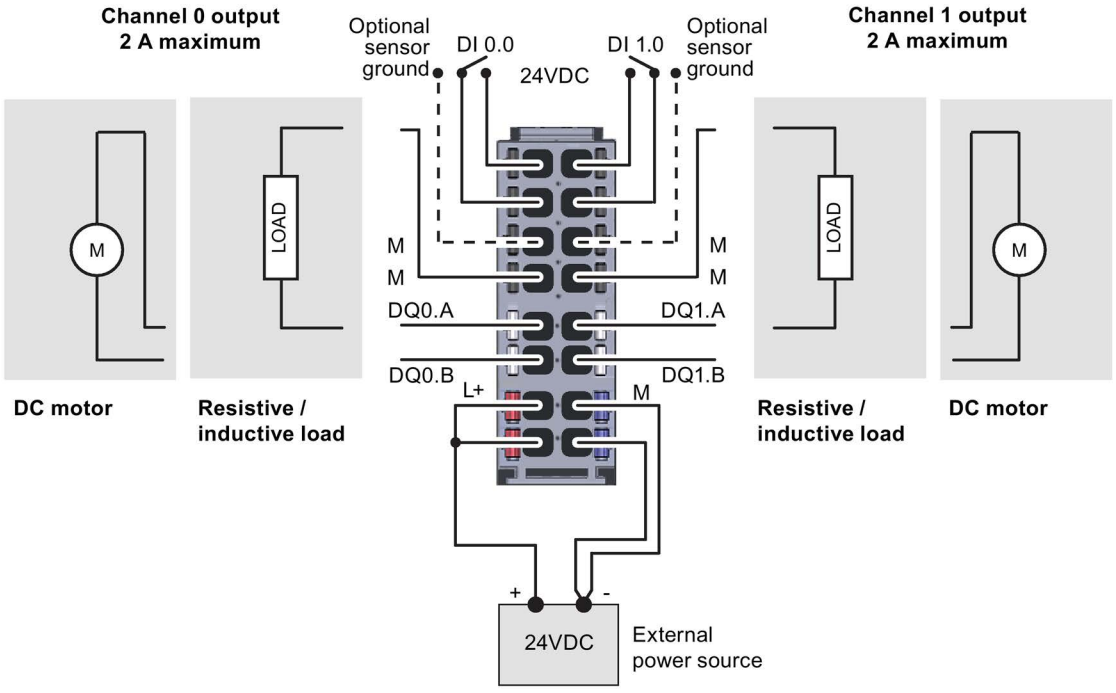


Figure 4-1 Dual channel wiring

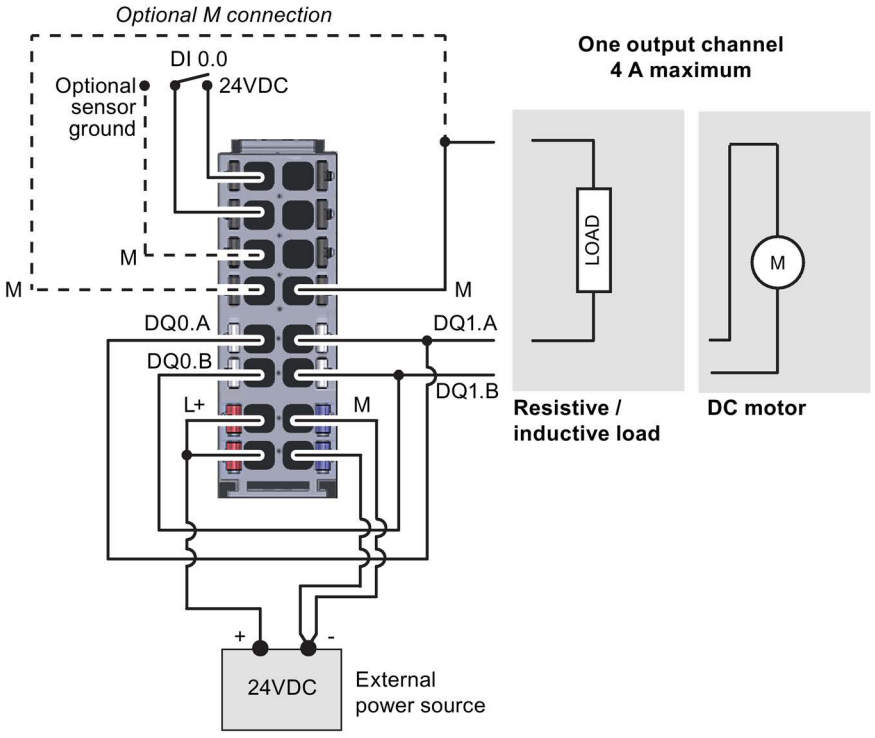


Figure 4-2 Single channel with parallel connection wiring

WARNING

Supply voltage M connections

Connect both the M potential pins to the power supply return with separate wires. If one wire breaks, then the other wire maintains the electrical connection from M to the power supply return

If the electrical connection between the M potential and the power supply return is broken, then unexpected conditions can occur at the digital outputs.

Note

Load wiring resistance

In single channel mode with parallel connected wiring, the wires from DQ0.A and DQ1.A must have equal resistance (length and size) for proper current sharing between the two outputs. Also, the wiring for DQ0.B and DQ1.B must have equal resistance when using DC motor mode.

Maximum current is limited and current measurement errors occur if the wiring does not have equal resistance.

Note

Excessive temperature from unsuitable loads

A high-speed output generates edges that are very steep. This creates very powerful charge reversals for the connected load, which can overheat the load at very high switching frequencies.

The connected load must therefore be approved for high input frequencies. See Function: High-speed output (Page 75) topic for details.

Note

The digital output switch-off response / switch-off edge depends on the load. Thus, it is possible that very short pulses cannot be output correctly.

Note

Relays and contactors can be connected direct without external circuitry.

Configuring

5

5.1 Configuration software

Introduction

The TM Pulse 2x24V module is configured and assigned parameters with the configuration software.

The module's pulse output sequences are controlled and monitored by your program.

System environment

The technology module can be used in the following system environments:

Table 5- 1 Applications of the technology module with PROFINET I/O

Applications	Components required	Configuration software	In your program
Decentralized operation in an S7-1500 system	• S7-1500 automation system • ET 200SP decentralized I/O system • TM Pulse 2x24v	STEP 7 (TIA Portal): Device configuration and parameter settings with hardware configuration (HWCN)	Direct access to the control and feedback interface (Page 110) of the TM Pulse 2x24V in the I/O data
Centralized or decentralized operation in an ET 200SP system	• ET 200SP automation system • TM Pulse 2x24v	STEP 7 (TIA Portal): Device configuration and parameter settings with hardware configuration (HWCN)	Direct access to the control and feedback interface of the TM Pulse 2x24V in the I/O data
Decentralized operation in an S7-300/400 system	• S7-300/400 automation system • ET 200SP decentralized I/O system • TM Pulse 2x24V	STEP 7 (TIA Portal): Device configuration and parameter settings with hardware configuration (HWCN) STEP 7: Device configuration and parameter settings with HSP	Direct access to the control and feedback interface of the TM Pulse 2x24V in the I/O data
Decentralized operation in a PROFINET controller or PROFIBUS master	PROFINET controller • ET 200SP decentralized I/O system • TM Pulse 2x24V	Engineering system with GSD file	Direct access to the control and feedback interface of the TM Pulse 2x24V in the I/O data

5.2 Configuration overview

You can use the STEP 7 (TIA Portal) or STEP 7 hardware configuration to set these parameters. Also, you can change the parameter assignment at runtime with your program using data record 128.

STEP 7 (TIA Portal) and STEP 7 assist you during parameter assignment by disabling keyboard input for invalid parameters and range checking your value assignments. Depending on previous parameter selections, other options will be disabled. For example, if you select the one channel operation (the parallel connection of the two output channels), then the parameter options for channel two and high-speed output are disabled.

When you make a runtime parameter assignment that uses the WRREC (Write Record) instruction to modify data record 128, you must ensure that you do not attempt to write an invalid record data. WRREC execution with invalid data fails and returns error code. For example, if you are using one channel operation and include parameter data for two channels, then the record length is too long and WRREC execution fails. In addition, you must follow the parameter validation rules (Page 120).

If you use STEP 7 (TIA Portal), you can find the module in the Hardware catalog under "Technology Modules". If you use STEP 7, you can find the module following installation of the corresponding HSP file in the Hardware catalog.

STEP 7 has two entries for Technology Module - Pulse output, with one entry for "TM Pulse 2x24V 2x2A" and one entry for "TM Pulse 2x24V 1x4A.

The following table shows how a channel's "Parameter" group is affected by mode selection.

Table 5- 2 Channel parameters and mode selection: ✓ means the parameter is available for configuration

Channel "Parameter" group	Pulse output mode	PWM mode	Pulse train mode	On/Off delay mode	Frequency output mode	DC motor mode
[1] High-speed output (.1 A)	✓	✓	✓	✓	✓	
HW enable option on DIn.0 input	✓	✓[3]	✓		✓	✓[3]
Input delay	✓	✓	✓	✓	✓	✓
Output format	✓	✓	✓	✓	✓	✓
Minimum pulse duration		✓[4]				
Period duration		✓	✓			✓
On delay	✓	✓[3]	✓		✓	✓[3]
Off delay				✓		
Duty cycle			✓			
[2] Current control		✓				
Activate P (Proportional)		✓				
Activate I (Integral)		✓				
Activate D (Derivative)		✓				
Reference value current		✓				
Dead band width		✓				
High limit S7 analog		✓				
Low limit S7 analog		✓				
Gain		✓				

Channel "Parameter" group	Pulse output mode	PWM mode	Pulse train mode	On/Off delay mode	Frequency output mode	DC motor mode
Integration time		✓				
Derivative action time		✓				
Time lag		✓				
Dithering		✓				
Dither ramp up time		✓				
Dither ramp down time		✓				
Dither amplitude		✓				
Dither period		✓				

[1] High speed output is not available for the one channel (4 A) configuration that uses a parallel connection of the two (2 A) channels.

[2] In PWM mode, you can enable either High-speed or Current control, but not both options at once.

[3] Parameter is disabled in isochronous mode.

[4] Parameter is disabled when current control is active

5.3 Required I/O address space

Address space of the technology module

Table 5- 3 TM Pulse 2x24V I/O address space usage

Function	Bytes per channel	Total I/O bytes Single channel (4 A)	Total I/O bytes Dual channel (2 A)
Program control of module operation	12 output bytes (Q addresses)	12 output bytes	24 output bytes
Feedback module status to control program	8 input bytes (I addresses)	8 input bytes	16 input bytes

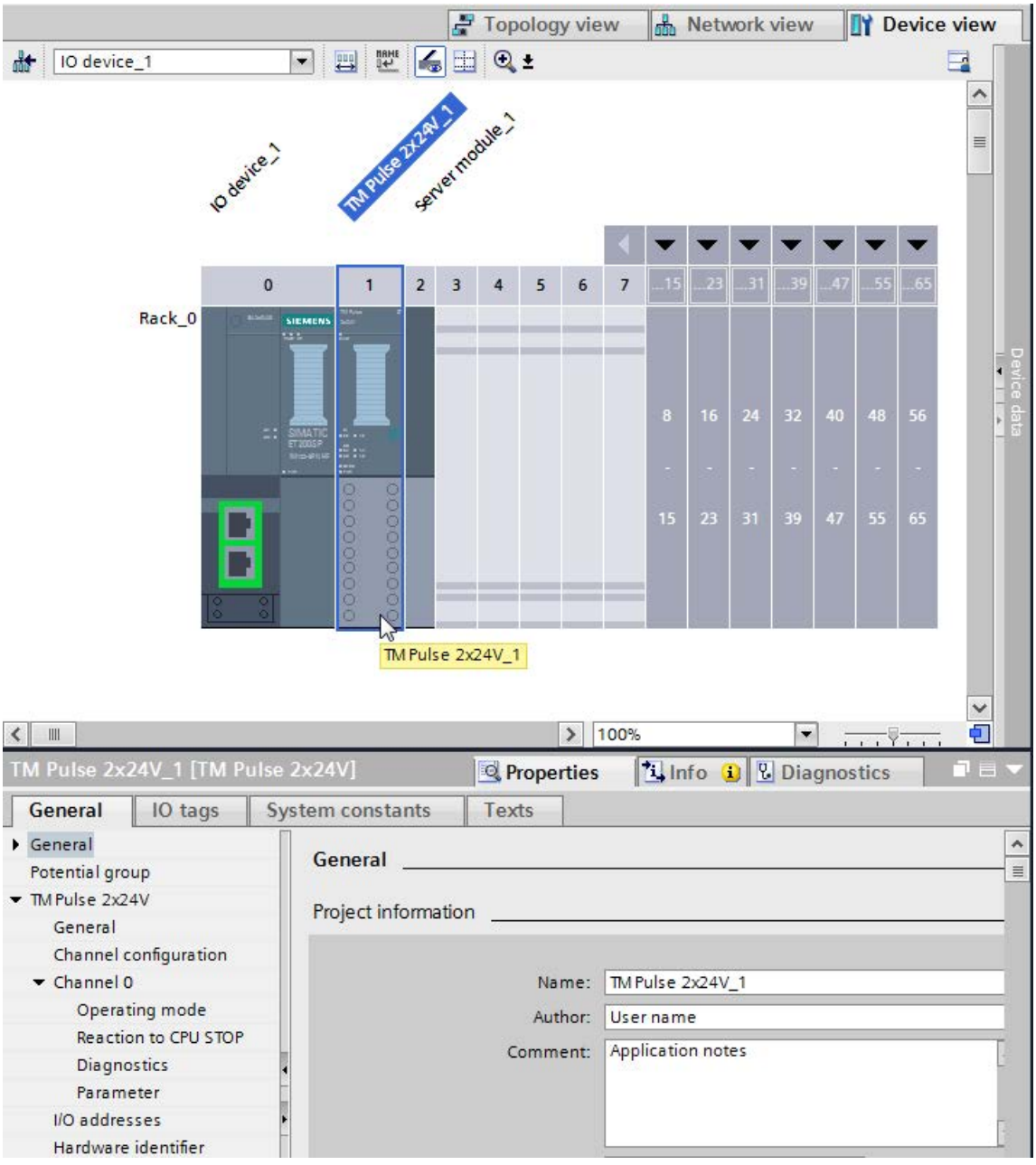
Additional information

A description on how to use the control and feedback interface of TM Pulse 2x24V can be found in the chapter Program control and feedback interface (Page 110).

5.4 TIA portal Device configuration

5.4.1 TIA Portal Device configuration

Drag the TM Pulse 2x24V module from the hardware catalog and drop it in a rack image. The example rack below uses the TM Pulse 2x24V module in a decentralized I/O system where it is possible to enable and configure isochronous mode. When you click on the TM Pulse 2x24V image in a rack, a blue line highlights the module and you can set parameters that appear on the Properties tab.



5.4.2 General information

Enter the general project and identification & maintenance information.

5.4.3 Potential group

The Potential group parameters are disabled for the type B1 BaseUnit used by the TM Pulse 2x24V. The TM Pulse 2x24V module is isolated from the other BaseUnit potential groups on the right and on the left of the TM Pulse 2x24V module. Therefore, the TM Pulse 2x24V requires an external supply.

5.4.4 Channel configuration: (4 A) single or (2 A) dual channel operation

Select the channel configuration:

- Two separate channels with 2 ampere maximum output current per channel. The high-speed option (100mA maximum current) is allowed.
- One logical channel where the two output channels connected in parallel to provide 4 ampere maximum output current. The high-speed option is not allowed.

1 channel (4 A) operation

If the "1 channel (4 A)" option is selected, then the TM Pulse 2x24V operates as a single-channel module. All parameter assignments and operations using the control interface must use Channel 0 addresses. Feedback signals are only provided for Channel 0.

The control and feedback interface is reduced to only one channel. Channel 1 cannot be configured or operated.

The current measurement adds the measured values of both channels together. This results in a measuring range up to 4 A.

The measured value of the current is supplied as a SIMATIC S7 analog value in the feedback interface. A current of 4 A corresponds to the SIMATIC S7 analog value of 27,648.

See also Pin assignment and load/sensor wiring (Page 91)

5.4.5 Channel parameters

5.4.5.1 Operating mode

Select an operating mode.

- Pulse output (single pulse)
- Pulse width modulation PWM
- Pulse train
- On/Off-delay
- Frequency output
- DC motor

5.4.5.2 Reaction to CPU STOP

You can configure the reaction of the TM Pulse 2x24V to the failure of a higher-level controller differently for each channel.

Table 5- 4 Reaction to CPU/Master STOP options for all modes except DC motor

Response to CPU/master STOP	Channel-specific response and status of the TM Pulse 2x24V
DQ substitute a value	• Output of the channel-specific, substitute value that you assigned. • STS_ENABLE goes to the 0 state. • Terminate the current output sequence.
Continue working mode	The current output sequence is continued.

Reaction to CPU/Master STOP options for DC motor mode

For DC motor mode, there are no configuration options for the reaction to CPU STOP. The outputs always behave as follows.

Both DQs (DQn.A and Dn.B) go to tri-state (high impedance state) when the CPU goes to STOP.

Substitute values

If you select the "DQ substitute a value" option as a reaction to CPU STOP, then you must configure the substitute value (0 or 1) for the DQn.A and DQn.B outputs.

Each channel has two outputs (A and B). A resistive or inductive load is wired to a channel's A output, so you must assign a substitute value to the DQn.A output. A reversible direction DC motor load is wired to a channel's A and B outputs, so you must assign substitute values to the DQn.A and DQn.B outputs.

You cannot set both substitute values of one channel to the high state (DQn.A and DQn.B). If you do so, an error code is returned to the parameterization attempt. See the Parameter validation errors (Page 120) topic for details.

Startup

To start a new output sequence after CPU/master STOP with STS_SW_ENABLE set, first reset SW_ENABLE. Keep SW_ENABLE reset until STS_SW_ENABLE is also reset.

If the "Continue working mode" option is used, then during a change from CPU-/Master-STOP to RUN (startup), the CPU/Master cannot clear the outputs.

Possible solution: In the part of your program that is executed during startup, set the "Software enable" (SW_ENABLE = 1) control bit.

Modified parameter assignment

The status assumed by the TM Pulse 2x24V at CPU/master STOP remains even in the case of parameter assignment or configuration of the ET 200SP station. This occurs, for example, at POWER ON of the CPU/master, or the IM 155-6, or at the resumption of DP transfer.

In "Continue working mode", however, and after loading changed parameters or configuration of the ET 200SP station to the CPU/master, the TM Pulse 2x24V terminates the process. As a result, the TM Pulse 2x24V does the following:

- Resets the DQ digital output.
- Resets STS_ENABLE
- Terminates the current output sequence.

5.4.5.3 Diagnostics

Module monitoring is always active. A detected error only triggers a diagnostic alarm if the diagnostics type is enabled in the Diagnostics check boxes.

TM Pulse 2x24V diagnostics

Diagnostic type	TM Pulse 2x24V module error	Default option
Group diagnostics	• Supply voltage error • Short-circuit on 24 V DC sensor supply	Disabled
Diagnostics DQA	Short-circuit of DQn.A digital output	Disabled
Diagnostics DQB	Short-circuit of DQn.B digital output	Disabled

When a TM Pulse 2x24V error event triggers a diagnostic alarm, the following happens:

- The DIAG light flashes red when a diagnostics alarm is pending. Once you have remedied the error, the DIAG light goes out.
- The diagnostics are displayed as plain text in the STEP 7 (TIA Portal) online and diagnostics view.
- Options for the reaction of a CPU running your control program
  - CPU goes to STOP and interrupts processing of your program. The diagnostic interrupt OB (for example, OB 82) is called. The event that triggered the interrupt is written in the start information of the diagnostic interrupt OB.
  - CPU remains in RUN even if no diagnostic interrupt OB is present in the CPU. The technology module continues working unchanged if this is possible, while the error exists.
- Detailed information on the error event is available through execution of the RALRM (read alarm information) program instruction.

Note

Error message details

The Diagnostics information provided through OB 82 and RALRM does not indicate which channel has an error.

The alternative error information provided by the feedback interface ERR_xxx bits does indicate which channel has an error.

See also Error detection and diagnostics (Page 121)

5.4.5.4 Parameter (Channel parameters)

The different operating modes restrict the configuration to a subset of these parameters and options.

Channel parameters

High-speed output

- Enabled
  - High-speed (0.1 A) option single channel output limit is 0.1 A.
  - 1 channel (4 A) parallel channel connection is not possible.
  - PWM mode current control is not possible.
- Disabled
  - Single channel output limit is 2 A

Function DI (digital input)

- Input: DIn.0 functions as a digital input
- HW enable: DIn.0 functions as a Hardware enable for the pulse output sequence. The HW enable option is available for these modes.
  - Pulse output
  - PWM (not available for isochronous mode)
  - Pulse train
  - Frequency output
  - DC motor (not available for isochronous mode)
- External stop (DC motor only)

See also Function: High-speed output (Page 75)

Input delay

A digital input state change must remain stable over the input delay time, before the state change is accepted and processed. The input delay time provides noise filtering for the input wiring.

- Off (4 µs)
- 0.05 ms
- 0.1 ms (default value)
- 0.4 ms
- 1.6 ms
- 3.2 ms
- 12.8 ms
- 20 ms

Output format

Set the format and value range for ratio variables like duty cycle.

Output format options	Value range
• S7 analog output	0 to 27,648 -27648 to 27648 (in DC motor mode)
• Per 100	0 to 100
• Per 1000	0 to 1000
• Per 10000	0 to 10000

Minimum pulse duration

Assign the minimum pulse duration that you allow in µs.

Period duration

Assign the period duration in µs.

Actual period duration

This read-only field is displayed only when isochronous mode is enabled. The period duration value that is displayed is the actual isochronous compatible period duration that is calculated from the value you entered in the Period duration parameter.

On-delay

Assign the On-delay in µs

Pulse timing	Minimum		Maximum	
	High-speed disabled	High-speed enabled	High-speed disabled	High-speed enabled
Pulse duration	10 µs	1.5 µs	85,000,000 µs	
Period duration	100 µs	10 µs		
On-delay	0 µs	0 µs		

Dither parameters

- Dither ramp-up time: 0 ms to 30,000 ms.
- Dither ramp-down time: 0 ms to 30,000 ms.
- Dither amplitude: 0 to 500‰
- Dither period: From (4 x PWM period) to 100,000 µs. Also, the Dither period must be greater than 2 ms.

Current control

- Enabled
- Disabled

Current control parameters

If current control is enabled, then the following parameters are available for modification.

- **Activate P**: Enable/disable the proportional part in the PID algorithm
- **Activate I**: Enable/disable the integral part in the PID algorithm
- **Activate D**: Enable/disable the derivative part in the PID algorithm
- **Reference value current**: The reference value is used to define the maximum set point and the high and low limits of the controlled current. Typically, the maximum current can be measured in PWM mode with current control disabled and a duty cycle set to 100%. The value measured can be set as a reference for the current control. The maximum value is 4000mA for single channel operation (parallel channel connection enabled) and 2000mA per channel for dual channel operation (parallel connection disabled).
- **Deadband width**: A dead band is applied to the output current deviation from the set point current. The "Deadband width" determines the size of the dead band. The dead band is symmetrical: Actual dead band value range = -Deadband width to +Deadband width.
- **High limit S7 analog**: The manipulated value is always restricted to a high limit and low limit. The "High limit" parameter assigns the high limit in S7 analog format, relative to the reference value current. A value higher or equal than 27648 means 100% of the reference value current. The High limit value must be higher than the Low limit.
- **Low limit S7 analog**: The manipulated value is always restricted to a high limit and low limit. The "Low limit" parameter assigns the low limit in S7 analog format, relative to the reference value of the current. A value higher or equal than 27648 is not allowed. The Low limit value must be lower than the High limit.
- **Gain**: The proportional gain parameter assigns the amplification factor for the P part of the PID algorithm.
- **Integration time**: The proportional gain parameter assigns the amplification factor for the P part of the PID algorithm.
- **Derivative action time**: The derivative time parameter determines the time interval of the derivative action. If TD is smaller than the controller cycle time, TI will be set internally to the controller cycle time.
- **Time lag**: Time lag of the derivative action. The algorithm of the D-action contains a delay of TM_LAG. If TM_LAG is smaller than half the controller cycle time, TI will be set internally to half the controller cycle time.

5.4.6 I/O addresses

You can assign the base addresses for the control interface (12 output Q byte addresses/channel) and the feedback interface (8 input I byte addresses/channel). Your program logic uses the values stored in these addresses to control the TM Pulse 2x24V output and read feedback signals from the module.

I/O addresses

Output addresses

Start address: Assign a starting address to 12 bytes (Q addresses) for a channel's control interface.

End address: The control interface end address is a calculated read-only field.

Input addresses

Start address: Assign a starting address to 8 bytes (I addresses) for a channel's feedback interface.

End address: The feedback interface end address is a calculated read-only field.

Isochronous mode

The Isochronous mode checkbox is only displayed if your system hardware supports isochronous mode.

Organization block

Accept the default OB assignment or select a different OB. If isochronous mode is enabled and the Organization block (OB 61) "Synchronous cycle" is assigned, then TM Pulse 2x24V PWM operation in a decentralized I/O system becomes coordinated with bus master PROFINET cycles.

In isochronous mode, the PWM output sequence is synchronized with the moment T_o. The period duration is coordinated to the application cycle (the synchronous cycle, a multiple of the PROFINET cycle).

Process image

Accept the default assignment or assign a different Process image partition.

Program control and feedback interface

6

6.1 TM Pulse 2x24V control interface

Your program uses this interface to control the behavior of the technology module.

Control interface

The following table shows control interface assignment for one channel:

Bit → / Byte↓	7	6	5	4	3	2	1	0
00	OUTPUT_VALUE							
01								
02								
03								
04	SLOT							
05								
06								
07								
08	Reserved[1)]			MODE_SLOT	LD_SLOT			
09	Reserved[1)]		DITHER	SET_DQB	SET_DQA	Reserved[1)]	TM_CTRL_DQ	SW_ENABLE
10	Reserved[1)]							RES_ERROR
11	Reserved[1)]							

[1)] Must be set to 0

Control interface parameters

Two use cases are possible while using TM Pulse 2x24V module.

Case 1: Only the main parameter OUTPUT_VALUE is controlled every program scan cycle by your program logic. All the other parameters required for the output sequence are fixed.

Case 2: The main parameter OUTPUT_VALUE and another parameter SLOT are controlled every program scan cycle by your program logic. All other parameters required for the output sequence are fixed.

Other parameters required for the output sequence are defined prior to the start of an output sequence using one of two possible methods.

- TIA portal device configuration, STEP 7 hardware configuration, or WRREC execution modifies the parameterization data record.
- Use the Control interface in single-update mode and before switching to the permanent-update mode (case 2 only) with the assigned LD_SLOT value selecting the control parameter.

OUTPUT_VALUE

The interpretation of the value in OUTPUT_VALUE depends on the mode setting. The OUTPUT_VALUE is always updated. If an invalid value is detected (outside the allowed range), then the error flag ERR_OUT_VAL is set until a valid value is detected. During the error condition, the invalid value is ignored and the module continues with the last valid OUTPUT_VALUE.

Note that PWM ratio values are not checked. If a ratio value is higher than the format allows, then a ratio of 100% is used by the module.

When the PWM current control option is activated, the TM Pulse 2x24V module takes control of the duty cycle and the OUTPUT_VALUE control interface field is used to assign the target current as a ratio of target current/reference current.

Mode	OUTPUT_VALUE meaning	Data type
Pulse output	Pulse duration High-speed disabled: 10 µs to 85,000,000 µs High-speed enabled: 2 µs to 85,000,000 µs	UDInt
PWM	Current control disabled: Duty cycle Current control enabled: Ratio current	UDInt: Only 2 least significant bytes are used For channel 0: bytes 2 and 3 For channel 1: bytes 14 and 15
Pulse train	Number of pulses	UDInt
On/Off delay	On-delay in microseconds	UDInt
Frequency output	Frequency in Hz	Real
DC motor	Duty cycle	DInt: Only 2 least significant bytes are used For channel 0: bytes 2 and 3 For channel 1: bytes 14 and 15

SLOT, MODE_SLOT, and LD_SLOT

The TM Pulse 2x24V module has a SLOT field in the control interface for each channel that allows several parameters to be single-updated, or one parameter to be permanently controlled (in addition to OUTPUT_VALUE). The interaction of SLOT, MODE_SLOT, and LD_SLOT is described in the SLOT parameter handling (Page 113) topic.

SW_ENABLE

If 0 → 1, then activate the output sequence. For some output modes, you can configure HW_ENABLE (hardware enable) to combine with SW_ENABLE (software enable) and trigger an output sequence with a hardware signal.

TM_CTRL_DQ

- If 1, then outputs are controlled by the module and produce the pulse sequences.
- If 0, then the outputs are controlled directly by your program through SET_DQA and SET_DQB assignments
  Note: This bit has no effect in DC motor mode

SET_DQA

- If 1, then set the output A to 1, when TM_CTRL_DQ is inactive.
- If 0, then set the output A to 0, when TM_CTRL_DQ is inactive.
  Note: This bit has no effect in DC motor mode.

SET_DQB

- If 1, then set output B to 1, when TM_CTRL_DQ is inactive and SET_DQA is 0.
- If 0, then set the output B to 0, when TM_CTRL_DQ is inactive.
  Note: This bit has no effect in DC motor mode

DITHER

- Activates the start (rising edge on DITHER) of the dithering function and begins dithering ramp-up, if a ramp-up is configured.
- Activates the end (falling edge on DITHER) of the superimposed dithering and begins dithering ramp-down, if a ramp-down is configured.

RES_ERROR

Resets the error flags ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V in the feedback interface.

See also

Parameter data record (Page 137)

6.2 SLOT parameter handling (control interface)

SLOT and MODE_SLOT

SLOT has the following modes.

- **Single-update mode** (MODE_SLOT = 0)
  Use this mode if some parameters have to be changed sometimes, prior to starting the output sequence. Using the SLOT parameter is an alternative to sending a new parameterization record to the module. A pulse operation change made with the SLOT parameter does not reset the module. A pulse operation change made by sending a new parameterization record does reset the module.
  - The value in SLOT is used each time the value in LD_SLOT changes.
  - An acknowledge bit is toggled in the feedback interface STS_LD_SLOT.
  - The interpretation of the SLOT is defined by the value of LD_SLOT (see the following table).
  - If the LD_SLOT value is invalid, a parameter error is triggered ERR_LD, and the user has to reset the error using the RES_ERROR control bit to reactivate SLOT.
  - The changes done in this mode can be read back from the module in the parameterization record.
  - The changes done in this mode are not permanently stored in the CPU. A restart of the CPU resets the parameters to the values specified in the hardware configuration.
- **Permanent-update mode** (MODE_SLOT = 1)
  Use this mode, if in addition to the main controlled parameter, another parameter has to be controlled continuously by the program.
  - The value in SLOT is transferred each module cycle.
  - No acknowledge bit is available.
  - The interpretation of SLOT is defined by the value of LD_SLOT (see the following table).
  - If the value in SLOT is not valid, then the error ERR_SLOT_VAL occurs. The error is automatically reset once a valid value is loaded. Note that the ratio values are not checked, if the value is higher than the format allows, a ratio of 100% will be used by the module.
  - Using this mode, the value is not updated in the parameterization record. If LD_SLOT is changed in this mode, the last valid value controlled previously is retained.
  - The permanent update mode can be stopped by setting LD_SLOT to 0 and MODE_SLOT to 0. By stopping permanent update mode, changes made to parameters during permanent update mode are retained.

Interpretation of SLOT parameter value

The value written to the SLOT parameter is interpreted as shown in the following table, depending on the LD_SLOT value and operating mode.

LD_SLOT	SLOT value meaning	Valid modes for SLOT value use	SLOT Data type
0	No action / idle state	All modes	
1	Period duration	PWM Pulse train DC motor	UDInt
2	On-delay	Pulse output PWM Pulse train Frequency output DC motor	UDInt
3	Off-delay	On/Off-delay	UDInt
4	Duty cycle (On-ratio)	Pulse train	UDInt: Only 2 least significant bytes are used. For channel 0: bytes 6 and 7 For channel 1: bytes 18 and 19
5	Dither ramp (includes ramp-up time and ramp-down time)	PWM	UDInt
6	Dither amplitude	PWM	UDInt
7	Dither period	PWM	UDInt

6.3 TM Pulse 2x24V feedback interface

Your program receives current values and status information from the technology module by means of the feedback interface.

Feedback interface

The following table shows the assignment of the feedback interface for one channel

Bit → / Byte ↓	7	6	5	4	3	2	1	0
00	ERR_SLOT_VAL	ERR_OUT_VAL	ERR_DQB	ERR_DQA	ERR_PULSE	ERR_LD	ERR_24V	ERR_PWR
01	Reserved		STS_SW_ENABLE	STS_READY	Reserved	STS_LD_SLOT	Reserved	
02	Reserved			STS_DITHER	STS_DI	STS_DQB	STS_DQA	STS_ENABLE
03	Reserved				SEQ_CNT			
04	MEASURED_CURRENT							
05								
06	Reserved						QLMN_HLM	QLMN_LLM
07	Reserved							

Feedback parameters

Table 6- 1 Status feedback

Feedback parameter	Meaning	Value range
STS_READY	Module is parameterized correctly, running, and delivering valid data.	0: Not ready 1: Ready
STS_SW_ENABLE	Current state of the software enable	0: SW_ENABLE is not active 1: Recognized SW_ENABLE
STS_LD_SLOT	Toggle acknowledge bit for each action of the SLOT in single-update SLOT-mode.	Each toggle of this bit indicates a successful LD_SLOT action.
STS_ENABLE	Output sequence is active.	0: No output sequence running 1: Output sequence running
STS_DQA	State of digital output DQn.A	0: DQn.A is not active 1: DQn.A is active
STS_DQB	State of digital output DQn.B	0: DQn.B is not active 1: DQn.B is active
STS_DI	DIn.0: State of digital input	0: DIn.0 is not active 1: DIn.0 is active
STS_DITHER	Dithering state	0: Dithering inactive 1: Dithering active with ramp-up phase started and ramp-down not complete.
SEQ_CNT	Counts completed output sequences	0 to 15

Feedback parameter	Meaning	Value range
MEASURED_CURRENT	S7 analog value	0 to 32,767 27,648 means 4 A for "1 channel (4 A)" 27,648 means 2 A for "2 channels (2 A)"
QLMN_LLM	The low limit of the manipulated value has been reached.	0: Low limit not reached 1: Low limit reached
QLMN_HLM	The high limit of the manipulated value has been reached.	0: High limit not reached 1: High limit reached

Table 6- 2 Error feedback

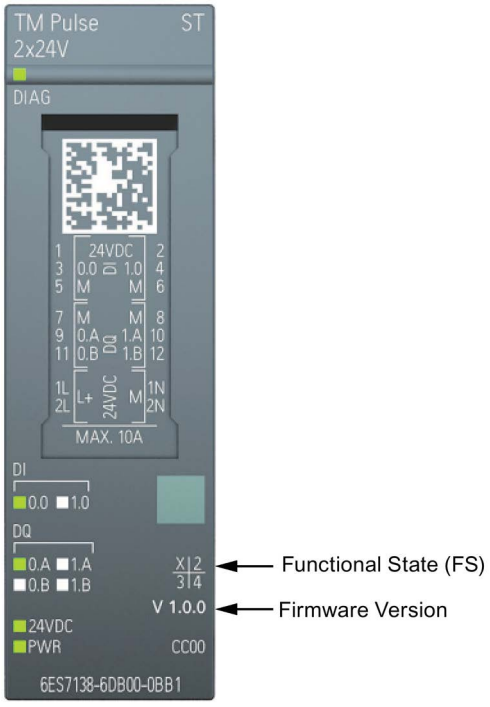
Feedback parameter	Meaning	Value range
ERR_PWR	24 V DC present, but is not in the correct range.	0: No error 1: Error
ERR_24V	Short-circuit/overload, in the 24 V DC sensor supply output.	0: No error 1: Error
ERR_LD	Error while loading a parameter value using single-update mode.	0: No error 1: Error
ERR_PULSE	Pulse duration reduced to smaller than the minimum allowed value during operation.	0: No error 1: Error
ERR_DQA	Short-circuit/overload on the digital output DQn.A detected	0: No error 1: Error
ERR_DQB	Short-circuit/overload on the digital output DQn.B or attempt to manually set both DQn.A and DQn.B high.	0: No error 1: Error
ERR_OUT_VAL	The value in OUTPUT_VALUE is not valid.	0: No error 1: Error
ERR_SLOT_VAL	The value in SLOT is not valid where MODE_SLOT = 1 (permanent update)	0: No error 1: Error

Interrupts/diagnostic messages

7

7.1 Status and error displays

TM Pulse 2x24V front view



LED status display

The following tables explain the meaning of the status and error displays. Refer to Error correction and diagnostics (Page 121) for details.

Table 7- 1 DIAG LED

DIAG LED	Meaning	To correct or avoid errors
□ Off	Backplane bus supply of the ET 200SP not OK	Check or switch On the supply voltage of the head station. Ensure the TM Pulse2x24V is correctly inserted in the BaseUnit.
☼ Flashes green	Technology Module not configured	
■ On green	Technology Module configured and no module error exists	
☼ Flashes red	Technology Module configured and module diagnostics (at least one error pending) Note:	Evaluate the diagnostic alarms and eliminate the error.

Note

The DIAG LED only shows an error if the Diagnostic alarm is activated during device configuration. By default, the Diagnostic alarm is not activated.

Table 7- 2 PWR and 24 V DC LED status

LEDs		Meaning	To correct or avoid errors
PWR	24 V DC		
□ Off	□ Off	No supply voltage	• Check the external 24 V DC power supply voltage connected between L+ and M.
■ On green	■ On green	Supply voltage is present and OK	
■ On green	□ Off	Short-circuit or overload at the sensor supply or supply voltage too low	• Check sensor wiring. • Check the loads connected to the sensor supply. • Check the supply voltage.

Channel status LEDs

The LEDs for DI digital inputs and DQ digital outputs indicate the digital state of the associated channel signals.
The LEDs of the digital outputs DQ indicate the desired state.

- DIn.0 digital inputs
  - 0.0 - Channel 0 input
  - 1.0 - Channel 1 input
- DQn.A and DQn.B digital outputs
  - 0.A and 0.B - Channel 0 outputs
  - 1.A and 1.B - Channel 1 outputs

The flashing frequency of the channel LEDs is limited to approximately 12 Hz. If higher frequencies are present, the channel LEDs will flash at 12 Hz instead of indicating the actual status.

Table 7- 3 Status displays DI digital inputs and DQ digital outputs

LED status	Meaning
□ Off	Digital input / digital output is at 0 level.
■ On green	Digital input / digital output is at 1 level.

7.2 Parameter validation errors

If the TM Pulse 2x24V parameter record is modified with an incorrect parameter value, then the module returns the error codes shown in the following table.

During TIA Portal Device configuration, parameter values are verified before transfer to the module. This prevents parameter errors during the static TIA Portal Device configuration.

You can modify the parameter data record during runtime with the WRREC (Write Record) program instruction.

For example, an illegal mode value is written to the module by WRREC execution. After execution the WRREC STATUS output is an ARRAY[1..4] of BYTE data with the value 16#DF80E111.

Example WRREC STATUS data	Address	Meaning
DF_H	STATUS[1]	Write data record error
80_H	STATUS[2]	Error according to IEC 61158-6
$E1_H$	STATUS[3]	Module specific error
11_H	STATUS[4]	Module error code from row 1 in the following table: Mode parameter has an incorrect reserved value.

Error code	Parameter	Validation criteria	Mode
11_H	Mode	Reserved value	All modes
12_H	Reaction to CPU STOP	Reserved value	All modes
$1A_H$	Input delay	Reserved value	All modes
$1B_H$	Period duration	Greater than maximum	PWM Pulse train DC motor
$1C_H$	On-delay	Greater than maximum	Pulse output PWM Frequency output DC motor
$1D_H$	Minimum pulse duration	Greater than maximum	PWM
$1E_H$	Off-delay	Greater than maximum	On/Off-delay
20_H	Reference value current	Greater than maximum	PWM
21_H	Period duration	Lower than minimum	PWM Pulse train DC motor
22_H	Dithering	Not disabled	Pulse output Pulse train On/Off-delay Frequency output DC motor
23_H	High-speed output	Not disabled	DC motor All modes with parallel connection
24_H	Current control	Not disabled	Pulse output Pulse train On/Off-delay Frequency output DC motor

Error code	Parameter	Validation criteria	Mode
25_H	Current control and high-speed output	Both enabled	PWM
26_H	Function DI	HW enable activated	On/Off-delay
27_H	Low limit and High limit	High limit <= Low limit	PWM with current control
28_H	Dithering period	Period is lower than 4 times the PWM period or lower than 2 ms.	PWM with dithering
29_H	Substitue values DQA and DQB	Substitute values DQA and DQB are both 1.	All modes

7.3 Error detection and diagnostics

Feedback interface ERR bits

Your program can access power supply error and output load status directly, through a channel's feedback interface.

Channel 0 address	Channel 1 address	Feedback bit	Meaning	Value range
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR	Indicates under voltage on the power supply. Note that the bit is not set, if the voltage is not present.	0 = PWR is not in under voltage 1 = PWR is detected but in under voltage
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V	Indicates a short-circuit or under voltage condition on the 24V sensor power supply.	0 = 24 V DC sensor supply working normally 1 = 24 V DC sensor supply short-circuit or under voltage
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD	Indicates an error occurred when loading a parameter value through the SLOT field using the single-update method.	0 = No SLOT single-update error 1 = SLOT single-update error
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE	Indicates a pulse output error in the Pulse output, On/Off-delay, and Pulse train modes	0 = No pulse output error 1 = Pulse output error
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA	Indicates a short-circuit or overload of DQn.A digital output.	0 = No DQn.A digital output short-circuit or overload 1 = DQn.A digital output short-circuit or overload
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB	Indicates a short-circuit, overload of DQn.B digital output. Also, indicates an attempt to set one channel's DQn.A and DQn.B to the high state (using SET_DQA, SET_DQB, and TM_CTRL_DQ).	0 = No DQn.B digital output short-circuit or overload 1 = DQn.B digital output short-circuit, overload, or attempt to set DQn.A and DQn.B high.
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL	Indicates the value in OUTPUT_VALUE is not valid	0 = No OUTPUT_VALUE error 1 = Error in OUTPUT_VALUE
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL	The value in SLOT is not valid (when MODE_SLOT = permanent update)	0 = No SLOT value error 1 = SLOT value error

Note

If an output short-circuit occurs, the digital output is able to briefly supply a current substantially higher than the rated value.

Diagnostic alarms

When a TM Pulse 2x24V error event triggers a diagnostic alarm, the following happens:

- The DIAG light flashes red when a diagnostics alarm is pending. Once you have remedied the error, the DIAG light changes to green.
- The diagnostics are displayed as plain text in the STEP 7 (TIA Portal) online and diagnostics view.
- Options for the reaction of a CPU running your control program
  - CPU goes to STOP and interrupts processing of the user program. The diagnostic interrupt OB (for example, OB 82) is called. The event that triggered the interrupt is written in the start information of the diagnostic interrupt OB.
  - CPU remains in RUN even if no diagnostic interrupt in present in the CPU. The technology module continues working unchanged if this is possible, while the error exists.
- Detailed information on the error event is available through execution of the RALRM (read alarm information) program instruction

Diagnostic configuration options

If a short-circuit or overload occurs on the 24 V DC sensor supply or digital output, the TM Pulse 2x24V generates a diagnostic message for the connected CPU/master. During the TM Pulse 2x24V device configuration, you must enable a channel's Group diagnostics and Diagnostics DQA / Diagnostics DQB parameters to enable the corresponding diagnostic messages.

See also TIA portal Diagnostics configuration (Page 105)

Configuration parameter	Meaning	Default
Group diagnostics	When Group diagnostics is enabled, the TM Pulse 2x24V generates a diagnostic message for the CPU/master in the following cases: • Supply voltage error • Short-circuit on 24 V DC sensor supply • Over-temperature	Disabled
Diagnostics DQA	The TM Pulse 2x24V detects a short-circuit/overload on the DQn.A digital output when Diagnostics DQA is enabled.	Enabled
Diagnostics DQB	The TM Pulse 2x24V detects a short-circuit/overload of the DQn.B digital output when Diagnostics DQB is enabled.	Enabled

Diagnostic alarms

Diagnostic alarm	Error Code	Meaning	To correct or avoid errors
Class A diagnostics (cannot be deactivated by user)			
Internal error	100_H	Technology Module is defective	Replace Technology Module
Watchdog tripped	103_H	Firmware Error, Technology Module is defective.	Run firmware update. If the problem remains, replace Technology Module.
Parametrization Fault	010_H	Possible causes: • The received parametrization record is not valid. • The configured BaseUnit is not the actual BaseUnit.	• Check the parametrization record. • Check the BaseUnit.
Module temporarily not available	$01F_H$	Possible causes: • Firmware-update is currently running. • Firmware-update has been interrupted.	• Wait for the end of the firmware update • Retry firmware update
Class B diagnostics (can be deactivated by user)			
No supply voltage	011_H	No L+ supply voltage for the technology module	Connect L+ supply voltage to the Technology Module.
Short-circuit or overload at the 24 V DC sensor supply	$10E_H$	Error on 24 V DC sensor supply Possible causes: • Short-circuit • Overload	• Correct wiring to 24 V DC. • Check loads connected to 24 V DC.
Error at the digital outputs	$10F_H$	Error at the digital outputs Possible causes: • Short-circuit • Overload	• Correct wiring at the digital outputs. • Check loads connected to the digital outputs.
Supply voltage error	110_H	Error at L+ supply voltage Possible causes: • Low voltage • Wiring of L+ supply voltage defective	• Check the L+ supply voltage. • Check the wiring of L+ supply voltage.
Over temperature	506_H	Possible causes: • Short-circuit or overload at the digital outputs • Ambient temperature outside specifications	• Correct process wiring • Improve cooling • Check connected loads

Technical specifications

8

	6ES7138-6DB00-0BB1
Product type designation	**TM Pulse 2x24 V**
General information	
Firmware version	V1.0
FW update possible	Yes
usable BaseUnits	BU type B1
Color code for module-specific color identification plate	CC40
Product function	
I&M data	Yes; I&M 0
Isochronous mode	Yes
Engineering with	
STEP 7 TIA Portal configurable/integrated as of version	V13 SP1
STEP 7 configurable/integrated as of version	V5.5 SP4 and higher
PROFIBUS as of GSD version/GSD revision	GSD Revision 5
PROFINET as of GSD version/GSD revision	GSDML V2.31
Supply voltage	
Load voltage L+	
Rated value (DC)	24 V
permissible range, lower limit (DC)	19.2 V
permissible range, upper limit (DC)	28.8 V
Short-circuit protection	Yes
Reverse polarity protection	Yes; against destruction
Input current	
Current consumption, max.	70 mA; without load
Encoder supply	
Number of outputs	2; A common 24V encoder supply for both channels

	6ES7138-6DB00-0BB1
24 V encoder supply	
24 V	Yes; L+ (-0.8 V)
Short-circuit protection	Yes; per module, electronic
Output current, max.	300 mA
Power loss	
Power loss, typ.	1.7 W
Address area	
Occupied address area	
Inputs	16 byte; 8 per channel
Outputs	24 byte; 12 per channel
Digital inputs	
Number of digital inputs	2; 1 per channel
Digital inputs, parameterizable	Yes
Input characteristic curve in accordance with IEC 61131, type 3	Yes
Digital input functions, parameterizable	
Freely usable digital input	Yes
HW enable for digital output	Yes
Input voltage	
Type of input voltage	DC
Rated value (DC)	24 V
for signal "0"	-30 to +5V
for signal "1"	+11 to +30V
permissible voltage at input, min.	-30 V
permissible voltage at input, max.	30 V
Input current	
for signal "1", typ.	2.5 mA
Input delay (for rated value of input voltage) for standard inputs	
parameterizable	Yes; none / 0.05 / 0.1 / 0.4 / 0.8 / 1.6 / 3.2 / 12.8 / 20 ms
at "0" to "1", min.	4 µs; for parameterization "none"
at "1" to "0", min.	4 µs; for parameterization "none"
Cable length	
shielded, max.	1000 m; Depending on load and cable quality
unshielded, max.	600 m; Depending on load and cable quality

	6ES7138-6DB00-0BB1
Digital outputs	
Type of digital output	P- and M-switching
Number of digital outputs	2; 1 per channel
Current-sinking	Yes
Current-sourcing	Yes
Digital outputs, parameterizable	Yes
Short-circuit protection	Yes; electronic/thermal
Response threshold, typ.	6.8 A with Standard output, 2 A with High Speed output
Limitation of inductive shutdown voltage to	-0.8 V
Accuracy of pulse duration	±100 ppm ±0.5 µs with high-speed output, ±100 ppm ±9 µs with standard output
minimum pulse duration	1.5 µs; With High Speed output, 10 µs with Standard output
Controlling a digital input	Yes
Digital output functions, parameterizable	
Freely usable digital output	Yes
PWM output	Yes
Number, max.	2; 1 per channel
Cycle duration, parameterizable	Yes; Max. 85 s
ON period, min.	0 %
ON period, max.	100 %
Resolution of the duty cycle	0.0036 %; For S7 analog format, min. 20 ns
Connection of a proportional valve	Yes
Dithering	Yes
Frequency adjustable	Yes
Amplitude adjustable	Yes
Current measurement	Yes
Current control	Yes
Connection of a DC motor	Yes
ON-delay	Yes
OFF-delay	Yes
Frequency output	Yes
Pulse train	Yes
Pulse output	Yes
Switching capacity of the outputs	
with resistive load, max.	2 A
on lamp load, max.	10 W; 1 W with High Speed output
Load resistance range	
lower limit	12 Ω; 240 ohm with High Speed output
upper limit	12 kΩ

	6ES7138-6DB00-0BB1
Output voltage	
Type of output voltage	DC
for signal "0", max.	1 V
for signal "1", min.	23.2 V; L+ (-0.8 V)
Output current	
for signal "1" rated value	2 A; 0.1 A with High Speed output, observe derating
Output delay with resistive load	
"0" to "1", typ.	0 µs; With high-speed output, 4.5 µs with standard output
"0" to "1", max.	0.8 µs; With high-speed output, 9 µs with standard output
"1" to "0", typ.	0 µs; With high-speed output, 4.5 µs with standard output
"1" to "0", max.	0.8 µs; With high-speed output, 9 µs with standard output
Parallel switching of 2 outputs	
for uprating	Yes
Switching frequency	
with resistive load, max.	100 kHz; With High Speed output, 10 kHz with standard output
with inductive load, max.	100 kHz; With High Speed output, 10 kHz with standard output
on lamp load, max.	10 Hz
Total current of the outputs	
Current per channel, max.	2 A
Current per group, max.	4 A
Current per module, max.	4 A
Cable length	
shielded, max.	1000 m; Depending on load and cable quality
unshielded, max.	600 m; Depending on load and cable quality
Isochronous mode	
Isochronous operation (application synchronized up to terminal)	Yes
Bus cycle time (TDP), min.	250 µs; with 1 channel configuration, 375 µs with 2 channel configuration
Jitter, max.	1 µs; typically ±
Interrupts/diagnostics/status information	
Substitute values connectable	Yes; Parameterizable
Alarms	
Diagnostic alarm	Yes

	6ES7138-6DB00-0BB1
Diagnostic messages	
Diagnostics	Yes
Monitoring the supply voltage	Yes
Short-circuit	Yes
Diagnostics indication LED	
Monitoring of the supply voltage (PWR-LED)	Yes; green PWR LED
Channel status display	Yes
for module diagnostics	Yes; green/red DIAG LED
Potential separation	
Potential separation digital inputs	
between the channels and the backplane bus	Yes
Potential separation digital outputs	
between the channels and the backplane bus	Yes
Potential separation channels	
between the channels	No
between the channels and the backplane bus	Yes
Permissible potential difference	
between different circuits	75 V DC/60 V AC (base isolation)
Isolation	
Isolation tested with	707 V DC (type test)
Ambient conditions	
Ambient temperature during operation	
horizontal installation, min.	0 °C
horizontal installation, max.	60 °C; Observe derating
vertical installation, min.	0 °C
vertical installation, max.	50 °C; Observe derating
Decentralized operation	
to SIMATIC S7-300	Yes
to SIMATIC S7-400	Yes
to SIMATIC S7-1200	Yes
to SIMATIC S7-1500	Yes
to standard PROFIBUS master	Yes
to standard PROFINET controller	Yes
Dimensions	
Width	20 mm
Weights	
Weight, approx.	50 g

Ambient temperature output current derating information

Use the following graphs to find the maximum output current, when operating at higher loads and temperatures.

The output current derating for single channel (4 A) is twice (2x) the derating of the dual channel (2 A per channel) values shown in the derating graphs.

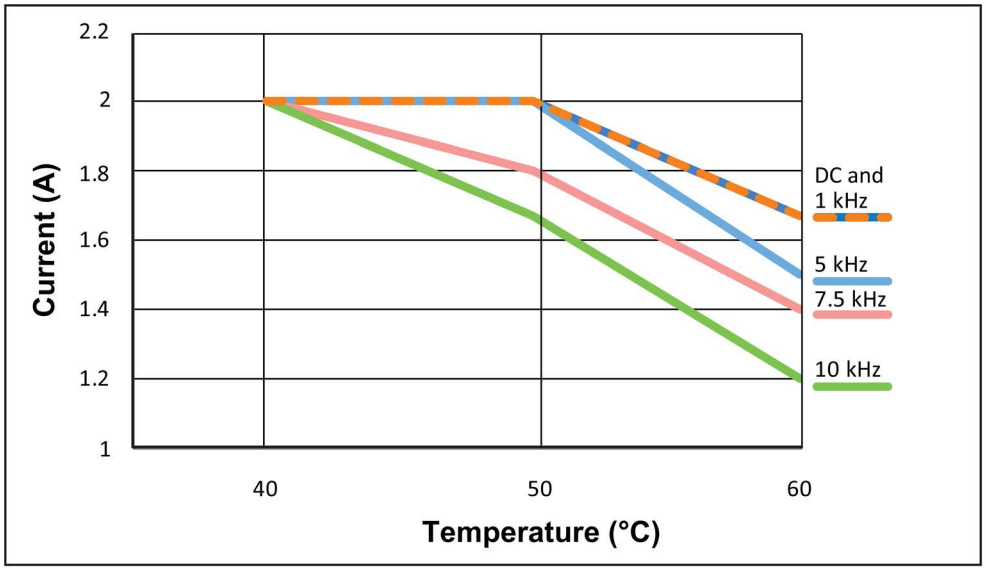


Figure 8-1 Maximum output current temperature derating for DC and 90%/10% duty cycle

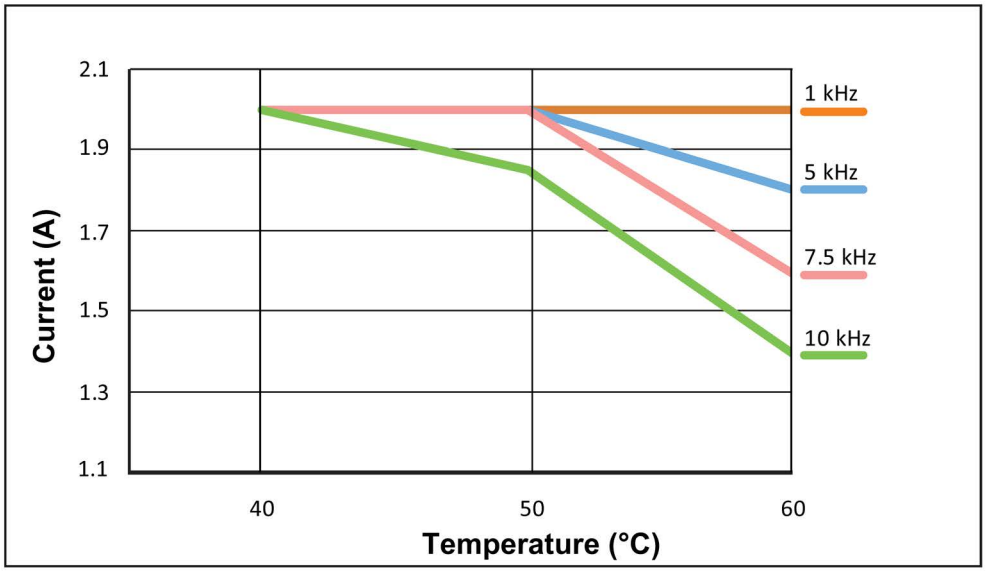


Figure 8-2 Maximum output current temperature derating for 50% duty cycle

BaseUnit technical specifications

Refer to the user manual for ET 200SP BaseUnits (http://support.automation.siemens.com/WW/view/en/58532597/133300)

8.1 Programming reference

Control interface: 2 channels, 24 output bytes (Q addresses)

CPU input address		Description
Channel 0	**Channel 1[1]**	
DWord 0	DWord 12	Depending on the mode: • Pulse output mode: Pulse duration in µs • PWM mode: Duty cycle On-ratio (Number range set by Output format configuration) – "Per 100": 0 to 100 – "Per 1000": 0 to 1,000 – "Per 10000": 0 to 10,000 – "S7 analog output": 0 to 27,648 • PWM mode with current control: Target current is assigned as a ratio of target current/reference current • Pulse train mode: Number of pulses to output as a DWord number value between 1 to 4,294,967,295 (2^{32}-1) • On/Off-delay mode: Off-delay in µs • Frequency output mode: Output frequency in Hz
DWord 4	DWord 16	SLOT value: Behavior depends on operating mode, MODE_SLOT(1 bit), and LD_SLOT (four bits).
Byte 8: Bits 0 to 3	Byte 20: Bits 0 to 3	LD_SLOT value controls interpretation of SLOT value. • 0 = No action / idle state • 1 = Period duration µs (PWM, Pulse train, and DC motor) • 2 = On-delay µs (Pulse output, PWM, Pulse train, Frequency output, and DC motor) • 3 = Off-delay µs (On/Off-delay) • 4 = Duty cycle On-ratio (Pulse train) • 5 = Dither ramp-up time and ramp-down time (PWM) • 6 = Dither amplitude (PWM) • 7 = Dither period (PWM)
Byte 8: Bit 4	Byte 20: Bit 4	MODE_SLOT value controls the SLOT update process. • 0 = single update (SLOT changed sometimes, prior to output sequence) • 1= permanent update (SLOT controlled continuously)
Byte 9: Bit 0	Byte 21: Bit 0	SW_ENABLE: Transition from 0 → 1 and remaining 1 during the input delay starts the output sequence. Only active for the first positive edge, additional positive edges are ignored and no start occurs.
Byte 9: Bit 1	Byte 21: Bit 1	TM_CTRL_DQ: Set DQ output source: Selects either CPU program or module's output sequence. • 0 = DQn.A and DQn.B are controlled by the CPU (in your program) using the SET_DQA and SET_DQB control bits. • 1 = DQn.A and DQn.B are controlled by the module's output sequence.

CPU input address		Description
Channel 0	**Channel 1[1]**	
Byte 9: Bit 2	Byte 21: Bit 2	SET_DQA: Controls the value of the digital output DQn.A, if TM_CTRL_DQ = 0 • 0 = 0V on DQn.A • 1 = 24V on DQn.A
Byte 9: Bit 3	Byte 21: Bit 3	SET_DQB: Controls the value of the digital output DQn.B, if TM_CTRL_DQ = 0 • 0 = 0V on DQn.B • 1 = 24V on DQn.B
Byte 10: Bit 0	Byte 22: Bit 0	RES_ERROR: Reset pending errors (ERR_LD, ERR_DQA, ERR_DQB, and ERR_24V). • 0 = Reset of errors is not active. • 1 = Reset of errors is active.

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Feedback Interface: 2 channels, 16 input bytes (I addresses)

CPU output address		Description
Channel 0	**Channel 1[1]**	
Byte 0: Bit 0	Byte 8: Bit 0	ERR_PWR: 1 = 24 V DC present, but is not in the correct range. 0 = no error
Byte 0: Bit 1	Byte 8: Bit 1	ERR_24V: 1 = Short-circuit/overload, in the 24 V DC sensor supply output. 0 = no error
Byte 0: Bit 2	Byte 8: Bit 2	ERR_LD: 1 = Error while loading a parameter value using single-update mode. 0 = no error
Byte 0: Bit 3	Byte 8: Bit 3	ERR_PULSE: 1 = Pulse duration reduced to smaller than the minimum allowed during operation. 0 = No error
Byte 0: Bit 4	Byte 8: Bit 4	ERR_DQA: 1 = Short-circuit/overload on the digital output DQn.A. 0 = no error
Byte 0: Bit 5	Byte 8: Bit 5	ERR_DQB: 1 = Short-circuit/overload on the digital output DQn.B or attempt to set both DQn.A and DQn.B to high. 0 = no error
Byte 0: Bit 6	Byte 8: Bit 6	ERR_OUT_VAL: 1 = The value in OUTPUT_VALUE is not valid. 0 = no error
Byte 0: Bit 7	Byte 8: Bit 7	ERR_SLOT_VAL: 1 = The value in SLOT is not valid where MODE_SLOT = 1 (permanent update). 0 = no error
Byte 1: Bit 2	Byte 9: Bit 2	STS_LD_SLOT: Toggle acknowledge bit for each action of the SLOT in single-update SLOT-mode Each toggle of this bit means a successful LD_SLOT action.
Byte 1: Bit 4	Byte 9: Bit 4	STS_READY: 1 = Module is parameterized correctly, running, and delivering valid data. 0 = not ready
Byte 1: Bit 5	Byte 9: Bit 5	ST_SW_ENABLE: 1 = SW_ENABLE active. 0 = SW_ENABLE not active
Byte 2: Bit 0	Byte 10: Bit 0	STS_ENABLE: 1 = Output sequence running. 0 = No output sequence running
Byte 2: Bit 1	Byte 10: Bit 1	STS_DQA: 1 = DQn.A output active. 0 = DQn.A output not active.
Byte 2: Bit 2	Byte 10: Bit 2	STS_DQB: 1 = DQn.B output active. 0 = DQn.B output not active.
Byte 2: Bit 3	Byte 10: Bit 3	STS_DI: 1 = DIn.0 digital input active. 0 = DIn.0 digital input not active.
Byte 3: Bit 0 to 3	Byte 11: Bit 0 to 3	SEQ_CNT: Sequence counter: Is incremented after completion of an output sequence (Range 0 to 15)
Word 4	Word 12	MEASURED_CURRENT: Current measurement uses a SIMATIC S7 analog value. Full-scale value depends on module configuration, as 2 channel (2 A) or 1 channel (4 A). • 2 channel (2 A): 0 to 32767 corresponds with 0 to 2.4 A • 1 channel (4 A): 0 to 32767 corresponds with 0 to 4.8 A

CPU output address		Description
Channel 0	**Channel 1[1]**	
Byte 6: Bit 0	Byte 14: Bit 0	QLMN_LLM: The low limit of the manipulated value has been reached.
Byte 6: Bit 1	Byte 14: Bit 1	QLMN_HLM: The high limit of the manipulated value has been reached.

[1] Only if the module is configured as "2 channels (2 A)" and not "1 channel (4 A)"

Note

If the TM Pulse 2x24V external supply voltage is interrupted, then, 16#00000000 is returned as feedback value (substitute value).

Controlling the different operating modes

You select an output channel's operating mode during device configuration. Configuration data is stored in parameter data record 128.
The following table shows the program variables that the different operating modes use.

Program control variable	Notes
Software enable	
SW_ENABLE	Transition from 0 → 1 and remaining 1 during the input delay starts the output sequence. Only active for the first positive edge, additional positive edges are ignored and no start occurs. You must always issue the software enable in your control program. If you don't use a HW enable, the output sequence will be started by the positive edge of the software enable. If you reset the software enable, the current output sequence will be terminated.
Direct control of the digital output	
TM_CTRL_DQ	• If TM_CTRL_DQ = 1, then the TM Pulse 2x24V module has control and produces pulse sequences at the DQ outputs. • If TM_CTRL_DQ = 0, then the CPU has control and your program can set outputs DQn.A and DQn.B directly with the SET_DQA and SET_DQB control bits
SET_DQA SET_DQB	These control bits set/reset a channel's DQn.A and DQn.B outputs while TM_CTRL_DQ = 0. Note: You cannot set a channel's DQn.A and DQn.B to high at the same time. Otherwise, error ERR_DQB is set and only DQn.A is set high.
Pulse output operating mode	
Pulse duration	Assign the pulse duration directly with the control interface parameter OUTPUT_VALUE, as a DWord number value in µs.
On-delay	The time from the start of the output sequence to the start of the DQ output pulse. Assign the On-delay in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 2.

Program control variable	Notes
PWM operating mode	
Duty cycle or Target current (current control enabled)	**Current control disabled**: PWM: OUTPUT_VALUE assigns the duty cycle (On/Off ratio) for the current period duration. You select the range of the OUTPUT_VALUE control interface field with the "Output format" configuration. • Output format "Per 100 (%)": Value range between 0 and 100 Pulse duration = (OUTPUT_VALUE/100) x period duration. • Output format "Per 1,000": Value range between 0 and 1,000 Pulse duration = (OUTPUT_VALUE/1,000) x period duration. • Output format "Per 10,000": Value range between 0 and 10,000 Pulse duration = (OUTPUT_VALUE/10,000) x period duration. • "S7 analog output" output format: Value range between 0 and 27,648 Pulse duration = (OUTPUT_VALUE/27,648) x period duration. **Current control enabled:** OUTPUT_VALUE assigns the target current as a ratio of target current/reference current. The reference current value is used to define the maximum set point and the high and low limits of the controlled current. Typically, the maximum current can be measured in PWM mode with current control disabled and a duty cycle set to 100%. The value measured can be set as a reference for the current control. The maximum value is 4000 mA for single channel operation (parallel channel connection enabled) and 2000 mA per channel for dual channel operation (parallel connection disabled).
Period duration	The Period duration of an output PWM cycle. Assign the period duration value in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 1. When you assign the period duration, take into account the minimum pulse duration configuration and the response time of the control element connected to the DQ digital output.
On-delay	The time from the start of the output sequence to the start of the DQ output pulse. Assign the On-delay in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 2.
Pulse train operating mode	
Number of pulses	Number of pulses that are output at the DQ digital output on expiration of the On-delay. Your control program can set the pulse count directly with the control interface parameter (OUTPUT_VALUE). Set the number of pulses directly as a DWord number value between 0 to 4,294,967,295 (2^{32}-1).
Period duration	The Period duration of an output pulse cycle. Assign the period duration in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 1. When you assign the period duration take into account the minimum pulse duration configuration and the response time of the control element connected to the DQ digital output.
Duty cycle	Assign the duty cycle with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 4. The range of the duty cycle parameter is selected using the "Output format" configuration. If the number value you assign exceeds the upper limit, then a duty cycle of 100% of the period duration is used and this action does not cause an error. • Output format "Per 100 (%)": Value range 0 to 100 Pulse duration = (duty cycle/100) x period duration. • Output format "Per 1000": Value range 0 to 1,000 Pulse duration = (duty cycle/1,000) x period duration. • Output format "Per 10000": Value range 0 to 10,000 Pulse duration = (duty cycle/10,000) x period duration. • Output format "S7 analog output": Value range 0 to 27,648 Pulse duration = (duty cycle/27,648) x period duration.

Program control variable	Notes
On/Off-delay operating mode	
On-delay	The time between a positive edge of the DIn.0 digital input and DQn.A digital output (DQ follows DI state). Assign the On-delay in µs directly using the OUTPUT_VALUE control interface field.
Off-delay	The time between a negative edge of the DIn.0 digital input and its output on the DQn.A digital output (DQ follows DI state). Assign the Off-delay in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 3.
Frequency output operating mode	
Output frequency	Frequency output at the digital output DQ. Assign the frequency in real format as Hz using the OUTPUT_VALUE control interface field. The possible range is dependent on the "High Speed Output" configuration. • High Speed Output disabled Frequency (OUTPUT_VALUE): 0.02 Hz to 10,000 Hz • High Speed Output enabled Frequency (OUTPUT_VALUE): 0.02Hz to 100,000Hz
On-delay	The time from the start of the output sequence to the output of the frequency. Assign the On-delay in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 2.
DC motor operating mode	
OUTPUT_VALUE	The OUTPUT_VALUE determines the duty cycle (pulse duration/period duration ratio) within a period (PWM). The period duration can be adjusted. The new output value is applied at the next rising edge of the output. The sign indicates direction of rotation (positive for forward) and (negative for backward). S7 analog output format: value range is -27,648 to +27,648 DInt data type: Only 2 least significant bytes are used For channel 0: bytes 2 and 3 For channel 1: bytes 14 and 15
On-delay	The time from the start of the output sequence to the output of the frequency. Assign the On-delay in µs with control interface parameter SLOT, after setting up MODE_SLOT (0 or 1) and LD_SLOT = 2.
If "Function DI" is parameterized as "External stop":	A rising edge on DIn.0 will stop the output sequence and stop the DC motor.

Device configuration (Assignments stored in parameter data record 128)

Parameters	Value Range	Default
Channel configuration	• 2 channels (2 A) • 1 channel (4 A)	2 channels (2 A)
Channel (0 and 1)		
Reaction to CPU STOP	• Continue working mode • DQ substitute a value	DQ substitute a value
Substitute value DQA	0 or 1	0
Substitute value DQB	0 or 1	0
Group diagnostics	Disable/enable	Disabled
Diagnostics DQA		Disabled
Diagnostics DQB		Disabled
Operating mode	• Pulse output • PWM • Pulse train • On/Off-delay • Frequency output • DC motor	PWM
High-speed output (0.1 A)	Disable/enable	Disabled
Current control (for PWM mode only)	Disable/enable	Disabled
Function DI HW enable option available for Pulse output, Pulse train, Frequency output and DC motor modes.	• Input • HW enable • External stop (DC motor only)	Input
Activate P	P-SEL: Add Proportional term for current control	Enabled
Activate I	I-SEL: Add Integral term for current control	Enabled
Activate D	D-SEL: Add Derivative term for current control	Disabled
Input delay	• Off (4µs) • 0.05 ms • 0.1 ms • 0.4 ms • 0.8 ms • 1.6 ms • 3.2 ms • 12.8 ms • 20 ms	0.1 ms
Output format: PWM and Pulse train modes DC motor mode, (only S7 analog output format is possible)	• S7 analog output format • Per 100 • Per 1000 • Per 10000	Per 100

Parameters	Value Range			Default
Output format (in the "Frequency output" operating mode)	1 Hz			1 Hz
Dithering (PWM mode only): Superimpose dithering waveform on PWM output sequence.	Disable/enable			Disabled
	Minimum		Maximum	
	High-speed **disabled**	High-speed **enabled**		
DWord: Minimum pulse duration for PWM and DC motor modes	10 µs	1.5 µs	85,000,000 µs	0
DWord: Period duration for PWM, Pulse train, and DC motor modes	100 µs	10 µs		2,000,000 µs
DWord: On-delay for all modes except On/Off-delay	0 µs	0 µs		0
DWord: Value depends on mode				
PWM: Dither ramp	Low word: Dither ramp-up time 0 to 30000 ms High word: Dither ramp-down 0 to 30000 ms			0 ms 0 ms
Pulse train: Duty cycle	S7 analog output format: 0 to 27648			13824 (50%)
	Per 100 format: 0 to100			50 (50%)
	Per 1000 format: 0 to 1000			500 (50%)
	Per 10000 format: 0 to 10000			5000 (50%)
On/Off delay: Off delay	0 to 85,000,000 µs			0
DWord: Dither amplitude for PWM only	0 to 500 ‰ (per mill)			50 ‰
DWord: Dither period for PWM only	From (4 times the PWM period µs) to 100000 µs (must be higher than 2000 µs)			50000 µs
Word: Reference value current for PWM with current control only	• 0 mA to 2000 mA for "2 channels (2 A)" operation • 0 mA to 4000 mA for "1 channel (4 A)" operation to 4000 mA for "1 channel (4 A)" operation			0 mA
Word: Dead band width (µA) for current control	0 µA to 65535 µA			0 µA
Word: High current limit for current control	S7-analog value relative to the reference value current: Range is 1 to 65535 (>= 27648 means 100%)			27648
Word: Low current limit for current control	S7-analog value relative to the reference value current: Range 0 to 27647 (Low limit must be less than the High limit)			0
Gain for current control	Real value (DWord size)			2.0 s
TI: Integration time (s) for current control	Real value (DWord size)			20.0 s
TD: Derivative action time (s) for current control	Real value (DWord size)			10 .0 s
TM LAG: Time lag of the derivative action (s)	Real value (DWord size)			2.0 s

Parameter data record

A

The TM Pulse 2x24V parameter data record is modified and stored for you by the TIA portal when you perform a Device configuration, successful configuration block compilation, and download a new configuration block to the system hardware.

You may also directly edit the module parameters with the CPU in RUN mode. The WRREC instruction is used to transfer parameters to the module using data record 128.

If errors occur during the transfer or validation of parameters with the WRREC instruction, the module continues operation with the previous parameter assignment. A corresponding error code is then written to the STATUS output parameter. If no errors occur, the STATUS output parameter contains the length of the data actually transferred.

The description of the WRREC instruction and the error codes is available in the STEP 7 online help (TIA Portal).

Structure of data record 128

The following table shows you the structure of data record 128 for the TM Pulse 2x24V. The values in byte 0 to byte 3 are fixed and may not be changed. Default values are indicated in bold font.

- A total of 108 bytes (4 header bytes + 2(52 channel bytes) is required for the 2 channel configuration (parallel connection disabled).
- A total of 56 byes (4 header bytes + 52 channel bytes) is required for the 1 channel configuration (parallel connection enabled).
- Bytes 4 to 55 are the channel 0 parameters
- Bytes 56 to 107 are the channel 1 parameters.
- Channel 1 parameters use the same data structure as channel 0. Add a 52 byte offset to the channel 0 byte numbers, to determine the channel 1 byte numbers.

Table A- 1 Header and channel 0 basic configuration

Bit → / Byte ↓	7	6	5	4	3	2	1	0
0 to 3	**Header**							
0	Reserved[1]		Major version = 0		Minor version = 1			
1	Channel parameter data length = 52 bytes							
2	Reserved[1]							
3								
4 to 55	**Channel 0 parameters**							
4	Current control	Dithering	High-speed output		Mode			
	0$_B$: Disabled	**0**$_B$: Disabled	**00**$_B$: Disabled					
	1$_B$: Enabled (not if High-speed mode is enabled)	1$_B$: Enabled	01$_B$: Enabled Not allowed for "1 channel (4A)" configuration that connects the two channels in parallel.					
			10$_B$-11$_B$: Reserved					
					0: Pulse output			
					1: PWM (Pulse width modulation)			
					2: Pulse train			
					3: On/Off-delay			
					4: Frequency output			
					5: DC motor			
5	PID loop calculation for current control			Reserved[1]		Diagnostic interrupt	Reaction to CPU STOP	
	P-SEL: Add Proportional term	I-SEL: Add Integral term	D-SEL: Add Derivative term			**0**$_B$: Disabled	**00**$_B$: DQ substitute a value	
						1$_B$: Enabled	01$_B$: Reserved	
	0$_B$: Disabled	0$_B$: Disabled	**0**$_B$: Disabled				10$_B$: Continue working mode	
	1$_B$: Enabled	**1**$_B$: Enabled	1$_B$: Enabled				11$_B$: Reserved	
6	Reserved[1]		Input delay				Function DI	
			0$_H$: Off (4 µs)				**00**$_B$: Input	
			1$_H$: 0.05 ms00				01$_B$:HW enable	
			2$_H$: 0.1 ms				10$_B$:External stop (DC motor mode only)	
			3$_H$: 0.4 ms					
			4$_H$: 0.8 ms					
			5$_H$: 1.6 ms					
			6$_H$: 3.2 ms					
			7$_H$: 12.8 ms					
			8$_H$: 20 ms					
			9$_H$ to F$_H$ Reserved					

<table>
<tr><td>Bit →
Byte ↓</td><td colspan="2">7</td><td colspan="2">6</td><td colspan="2">5</td><td colspan="2">4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="5">7</td><td colspan="4" rowspan="5">Reserved[1]</td><td colspan="4">Output format</td><td>Diagnostics DQB</td><td>Diagnostics DQA</td><td>Substitute value DQB</td><td>Substitute value DQA</td></tr>
<tr><td rowspan="2">Bits 5 and 4</td><td rowspan="2">PWM or Pulse train</td><td rowspan="2">Frequency output</td><td rowspan="2">DC motor</td><td>0B: Disable</td><td>0B: Disable</td><td>0B: 0 V</td><td>0B: 0 V</td></tr>
<tr><td>1B: Enable</td><td>1B: Enable</td><td>1B: 24V</td><td>1B: 24V</td></tr>
<tr><td>00B</td><td>S7 Analog format</td><td>Reserved</td><td>S7 Analog format</td><td colspan="4" rowspan="4"></td></tr>
<tr><td>01B</td><td>Per 100 (%)</td><td>1 Hz</td><td>Reserved</td></tr>
<tr><td>10B</td><td>Per 1,000</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>11B</td><td>Per 10,000</td><td>Reserved</td><td>Reserved</td></tr>
</table>

[1] Must be set to 0.

Table A- 2 Channel 0 Word, DWord, and Real values configuration

Byte	Channel 0 mode and variable usage	Value range
8 to 11	**Minimum pulse duration DWord**	
	PWM (only if current-control is inactive)	Minimum pulse duration: Default value = **0 µs**
	Pulse output, Pulse train, On/Off-delay, and Frequency output	Reserved
12 to 15	**Period duration DWord**	
	PWM, Pulse train	Period duration: High-speed disabled: 100 µs to 85,000,000 µs High-speed enabled: 10 µs to 85,000,000 µs Default value = **2,000,000 µs**
	DC motor	Period duration: 100 µs to 85,000,000 µs Default value = 1000 µs
	Pulse output, On/Off-delay, and Frequency output	Reserved
16 to19	**On-delay DWord**	
	Pulse output, PWM, Pulse Train, Frequency output, and DC motor	On-delay: **0 µs** to 85,000,000 µs
	On/Off-delay	Reserved
20 to 23	**Value DWord**	
	PWM	Dithering ramp times (2 bytes each): • Ramp-up time (low word): **0 ms** to 30,000 ms • Ramp-down time (high word): **0 ms** to 30,000 ms
	Pulse train	Duty cycle %: 0 to 27648, Default = **50** (50%)
	On/Off-delay	Off-delay: **0 µs** to 85,000,000 µs
	Pulse output, Frequency output, and DC motor	Reserved
24 to 27	**Dither amplitude DWord**	
	PWM	Dither amplitude (per mil): 0 to 500 the default value is **50**.
	Pulse output, Pulse train, On/Off-delay, Frequency output, and DC motor	Reserved
28 to 31	**Dither period DWord**	
	PWM	Dither period: Valid from ((4 x PWM period) AND (> 2000)) µs to 100,000 µs. The default value is **50,000 µs**.
	Pulse output, Pulse train, On/Off-delay, Frequency output, and DC motor	Reserved

Byte	Channel 0 mode and variable usage	Value range
32 to 55	**PWM mode only: Current control parameters**	
32 to 33	Reference value current (mA)	Word size: **0 mA** to 2000 mA for "2 channels (2 A)" operation **0 mA** to 4000 mA for "1 channel (4 A)" operation
34 to 35	Dead band width (µA)	Word size: **0 µA** to 65535 µA
36 to 37	High limit - S7-analog value relative to the reference value current	Word size: 1 to 65535 (>= **27648** means 100%)
38 to 39	Low limit - S7-analog value relative to the reference value current	Word size: **0** to 27647 (because Low limit must be less than the High limit)
40 to 43	Gain	Real (DWord size): Default value = **2.0 s**
44 to 47	TI: Integration time (s)	Real (DWord size): Default value = **20.0 s**
48 to 51	TD: Derivative action time (s)	Real (DWord size): Default value = **10.0 s**
52 to 55	TM LAG: Time lag of the derivative action (s)	Real (DWord size): Default value = **2.0 s**

Open Source Software

B

For Resellers: In order to avoid infringements of the license conditions by the reseller or the buyer, the instructions and license conditions provided here must be passed on to the buyers.

License Conditions and Disclaimers for Open Source Software and other licensing software

The Open Source software listed below is used in unmodified form or in a form we have modified as well as other license software listed below in the "digital modules, analog modules, technology modules, communication modules and power supply modules of the SIMATIC S7-1500, ET 200MP", ET 200SP Copyright Siemens AG, 2013-2014 (hereinafter referred to as "Product").

Liability for Open Source Software

The Open Source software is provided free of charge. We are liable for the Product including the Open Source software contained in it pursuant to the license conditions applicable to the Product. We explicitly reject any liability for the use of Open Source software beyond the program sequence intended for the Product. Furthermore, any liability for defects resulting from modifications to the Open Source software is excluded.

We do not provide any technical support for the Product if it has been modified.

Please read the license conditions and copyright information of Open Source software as well as other licensing software:

Component	Open Source Soft-ware[Yes/No]	Acknowledgements	Copyright Information / File
Dinkumware C/C++ Library - 5.01	NO		LICENSE AND COPYRIGHT INFORMATION FOR COMPONENT DINKUMWARE C/C++ LIBRARY - 5.01
GNU GCC libstdc++ / libsupc++ - 4.4.1	YES		LICENSE AND COPYRIGHT INFORMATION FOR COMPONENT GNU GCC LIBSTDC++ / LIBSUPC++ - 4.4.1
libgcc - 4.4.1	YES		LICENSE AND COPYRIGHT INFORMATION FOR COMPONENT LIBGCC - 4.4.1

Commercial Software: Dinkumware C/C++ Library - 5.01

Enclosed you'll find the license conditions and copyright notices applicable for Commercial Software Dinkumware C/C++ Library - 5.01

License conditions:

1 Copyright (c) 1991-1999 Unicode, Inc. All Rights reserved. This file is provided as-is by Unicode, Inc. (The Unicode Consortium).No claims are made as to fitness for any particular purpose. No warranties of any kind are expressed or implied. The recipient agrees to determine applicability of information provided. If this file has been provided on optical media by Unicode, Inc., the sole remedy for any claim will be exchange of defective media within 90 days of receipt.Unicode, Inc. hereby grants the right to freely use the information supplied in this file in the creation of products supporting the Unicode Standard, and to make copies of this file in any form for internal or external distribution as long as this notice remains attached.

2 © Copyright William E. Kempf 2001 Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. William E. Kempf makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

3 Copyright © 1994 Hewlett-Packard Company Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Hewlett-Packard Company makes no representations about the suitability of this software for any purpose. It is provided ``as is'' without express or implied warranty.

Copyrights:

copyright 1992 - 2006 by p.j. plauger and jim brodie. all rights reserved.

copyright 1992-2006 by p.j. plauger. all rights reserved.ip

copyright 1992-2006 by p.j. plauger. portions derived from work copyright 1994 by hewlettpackard company. all rights reserved

Copyright 1992-2006 by dinkumware, ltd. all rights reserved

copyright 1992-2006 by dinkumware, ltd. portions derived from work copyright 2001 by william e. kempf. all rights reserved

copyright 1994 hewlett-packard company

copyright 1994 by hewlett-packard company

copyright william e. kempf 2001

copyright 1989-2006 by p.j. plauger. all rights reserved

copyright (c) by p.j. plauger. all rights reserved.

copyright (c) by dinkumware, ltd. all rights reserved.

copyright (c) unicode, inc. all rights reserved.

(c) copyright william e. kempf 2001

copyright (c) hewlettpackard company

copyright (c) by p.j. plauger. all rights reserved.

copyright 2006 by dinkumware, ltd.

copyright (c) by p.j. plauger, licensed by dinkumware, ltd. all rights reserved.

the dinkum cec++ library reference is copyright (c) by p.j. plauger. this code is protected by copyright. all rights reserved.

the dinkum cc++ library reference is copyright (c) by p.j. plauger. this code is protected by copyright. all rights reserved.

dinkum complete library, vc++ package (vc++ compilers only) the dinkum compleat library and the dinkum compleat library reference are copyright (c) by p.j. plauger. all rights reserved.

Open Source Software: GNU GCC libstdc++ / libsupc++ - 4.4.1

Enclosed you'll find the license conditions and copyright notices applicable for Open Source Software GNU GCC libstdc++ / libsupc++ - 4.4.1

License conditions:

1 License

There are two licenses affecting GNU libstdc++: one for the code, and one for the documentation.

There is a license section in the FAQ regarding common questions. If you have more questions, ask the FSF or the gcc mailing list.
The Code: GPL

The source code is distributed under the GNU General Public License version 3, with the addition under section 7 of an exception described in the “GCC Runtime Library Exception, version 3.1” as follows (or see the file COPYING.RUNTIME):

GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright (C) 2009 Free Software Foundation, Inc.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyright.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyright requirements of the license of GCC.

Hopefully that text is self-explanatory. If it isn't, you need to speak to your lawyer, or the Free Software Foundation.
The Documentation: GPL, FDL

The documentation shipped with the library and made available over the web, excluding the pages generated from source comments, are copyrighted by the Free Software Foundation, and placed under the GNU Free Documentation License version 1.2. There are no Front-Cover Texts, no Back-Cover Texts, and no Invariant Sections.

For documentation generated by doxygen or other automated tools via processing source code comments and markup, the original source code license applies to the generated files. Thus, the doxygen documents are licensed GPL.

If you plan on making copies of the documentation, please let us know. We can probably offer suggestions.

2 This library is free
software; you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the
Free Software Foundation; either version 3, or (at your option)
any later version.
This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
Under Section 7 of GPL version 3, you are granted additional
permissions described in the GCC Runtime Library Exception, version
3.1, as published by the Free Software Foundation.
You should have received a copy of the GNU General Public License and
a copy of the GCC Runtime Library Exception along with this program;
see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
<http://www.gnu.org/licenses/>.

```
3   // This file is part of the GNU ISO C++ Library. This library is free
    // software; you can redistribute it and/or modify it under the
    // terms of the GNU General Public License as published by the
    // Free Software Foundation; either version 3, or (at your option)
    // any later version.
    // This library is distributed in the hope that it will be useful,
    // but WITHOUT ANY WARRANTY; without even the implied warranty of
    // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    // GNU General Public License for more details.
    // Under Section 7 of GPL version 3, you are granted additional
    // permissions described in the GCC Runtime Library Exception, version
    // 3.1, as published by the Free Software Foundation.
    // You should have received a copy of the GNU General Public License and
    // a copy of the GCC Runtime Library Exception along with this program;
    // see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
    // <http://www.gnu.org/licenses/>.
    // (C) Copyright Jeremy Siek 2000. Permission to copy, use, modify,
    // sell and distribute this software is granted provided this
    // copyright notice appears in all copies. This software is provided
    // "as is" without express or implied warranty, and with no claim as
    // to its suitability for any purpose.
```

```
4   // This file is part of the GNU ISO C++ Library. This library is free
    // software; you can redistribute it and/or modify it under the
    // terms of the GNU General Public License as published by the
    // Free Software Foundation; either version 3, or (at your option)
    // any later version.
    // This library is distributed in the hope that it will be useful,
    // but WITHOUT ANY WARRANTY; without even the implied warranty of
    // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    // GNU General Public License for more details.
    // Under Section 7 of GPL version 3, you are granted additional
    // permissions described in the GCC Runtime Library Exception, version
    // 3.1, as published by the, 2009 Free Software Foundation.
    // You should have received a copy of the GNU General Public License and
    // a copy of the GCC Runtime Library Exception along with this program;
    // see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
    // <http://www.gnu.org/licenses/>.
    /*
    *
    * Copyright (c) 1994
    * Hewlett-Packard Company
    *
    * Permission to use, copy, modify, distribute and sell this software
    * and its documentation for any purpose is hereby granted without fee,
    * provided that the above copyright notice appear in all copies and
    * that both that copyright notice and this permission notice appear
    * in supporting documentation. Hewlett-Packard Company makes no
    * representations about the suitability of this software for any
    * purpose. It is provided "as is" without express or implied warranty.
    *
    * Copyright (c) 1996,1997
    * Silicon Graphics
    *
    * Permission to use, copy, modify, distribute and sell this software
    * and its documentation for any purpose is hereby granted without fee,
    * provided that the above copyright notice appear in all copies and
    * that both that copyright notice and this permission notice appear
    * in supporting documentation. Silicon Graphics makes no
    * representations about the suitability of this software for any
    * purpose. It is provided "as is" without express or implied warranty.
```

```
// This file is part of the GNU ISO C++ Library. This library is free
// software; you can redistribute it and/or modify it under the
// terms of the GNU General Public License as published by the
// Free Software Foundation; either version 3, or (at your option)
// any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
// Under Section 7 of GPL version 3, you are granted additional
// permissions described in the GCC Runtime Library Exception, version
// 3.1, as published by the Free Software Foundation.
// You should have received a copy of the GNU General Public License and
// a copy of the GCC Runtime Library Exception along with this program;
// see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
// <http://www.gnu.org/licenses/>.
/*
*
* Copyright (c) 1996,1997
* Silicon Graphics Computer Systems, Inc.
*
* Permission to use, copy, modify, distribute and sell this software
* and its documentation for any purpose is hereby granted without fee,
* provided that the above copyright notice appear in all copies and
* that both that copyright notice and this permission notice appear
* in supporting documentation. Silicon Graphics makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied warranty.
*
*
* Copyright (c) 1994
* Hewlett-Packard Company
*
* Permission to use, copy, modify, distribute and sell this software
* and its documentation for any purpose is hereby granted without fee,
* provided that the above copyright notice appear in all copies and
* that both that copyright notice and this permission notice appear
* in supporting documentation. Hewlett-Packard Company makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied warranty.
```

6 This library is free
// software; you can redistribute it and/or modify it under the terms
// of the GNU General Public License as published by the Free Software
// Foundation; either version 3, or (at your option) any later
// version.
// This library is distributed in the hope that it will be useful, but
// WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
// General Public License for more details.
// Under Section 7 of GPL version 3, you are granted additional
// permissions described in the GCC Runtime Library Exception, version
// 3.1, as published by the Free Software Foundation.
// You should have received a copy of the GNU General Public License and
// a copy of the GCC Runtime Library Exception along with this program;
// see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
// <http://www.gnu.org/licenses/>.
// Copyright (C) 2004 Ami Tavory and Vladimir Dreizin, IBM-HRL.
// Permission to use, copy, modify, sell, and distribute this software
// is hereby granted without fee, provided that the above copyright
// notice appears in all copies, and that both that copyright notice
// and this permission notice appear in supporting documentation. None
// of the above authors, nor IBM Haifa Research Laboratories, make any
// representation about the suitability of this software for any
// purpose. It is provided "as is" without express or implied
// warranty.
/**
* @file splay_tree_.hpp
* Contains an implementation class for splay_tree_.
*/
/*
* This implementation uses an idea from the SGI STL (using a "header" node
* which is needed for efficient iteration). Following is the SGI STL
* copyright.
*
* Copyright (c) 1996,1997
* Silicon Graphics Computer Systems, Inc.
*
* Permission to use, copy, modify, distribute and sell this software
* and its documentation for any purpose is hereby granted without fee,
* provided that the above copyright notice appear in all copies and
* that both that copyright notice and this permission notice appear
* in supporting documentation. Silicon Graphics makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied warranty.
*
*
* Copyright (c) 1994
* Hewlett-Packard Company
*

* Permission to use, copy, modify, distribute and sell this software
* and its documentation for any purpose is hereby granted without fee,
* provided that the above copyright notice appear in all copies and
* that both that copyright notice and this permission notice appear
* in supporting documentation. Hewlett-Packard Company makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied warranty.

7 This library is free
// software; you can redistribute it and/or modify it under the
// terms of the GNU General Public License as published by the
// Free Software Foundation; either version 3, or (at your option)
// any later version.
// This library is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
// Under Section 7 of GPL version 3, you are granted additional
// permissions described in the GCC Runtime Library Exception, version
// 3.1, as published by the Free Software Foundation.
// You should have received a copy of the GNU General Public License and
// a copy of the GCC Runtime Library Exception along with this program;
// see the files COPYING3 and COPYING.RUNTIME respectively. If not, see
// <http://www.gnu.org/licenses/>.
/*
* Copyright (c) 1997
* Silicon Graphics Computer Systems, Inc.
*
* Permission to use, copy, modify, distribute and sell this software
* and its documentation for any purpose is hereby granted without fee,
* provided that the above copyright notice appear in all copies and
* that both that copyright notice and this permission notice appear
* in supporting documentation. Silicon Graphics makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied warranty.

8 Verbatim copying and distribution of this entire article are
permitted worldwide, without royalty, in any medium, provided this
notice is preserved.

9 GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyright license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work

conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

10 Disclaimer and Copyright</h2>
Revised 16 February, 2004 © Copyright Ami Tavory and Vladimir Dreizin, IBM-HRL, 2004, and Benjamin Kosnik, Red Hat, 2004.
Permission to use, copy, modify, sell, and distribute this software is hereby granted without fee, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation.
None of the above authors, nor IBM Haifa Research Laboratories, Red Hat, or both, make any representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.<

11 # This file file be copied and used freely without restrictions. It can
be used in projects which are not available under the GNU Public License
but which still want to provide support for the GNU gettext functionality.
Please note that the actual code is *not* freely available.

12 This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

13 Free Software Foundation, Inc.
This file is free software; the Free Software Foundation gives unlimited permission to copy and/or distribute it, with or without modifications, as long as this notice is preserved.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, to the extent permitted by law; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

14 This file is free software; the Free Software Foundation gives unlimited permission to copy and/or distribute it, with or without modifications, as long as this notice is preserved.

15 Permission to use, copy, modify, sell, and distribute this software is hereby granted without fee, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation. None of the above authors, nor IBM Haifa Research Laboratories, make any representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

16 This configure script is free software; the Free Software Foundation
gives unlimited permission to copy, distribute and modify it.

17 This Makefile.in is free software; the Free Software Foundation
gives unlimited permission to copy and/or distribute it,
with or without modifications, as long as this notice is preserved.
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY, to the extent permitted by law; without
even the implied warranty of MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

18 Distributed under the Boost
Software License, Version 1.0. (See accompanying file
LICENSE_1_0.txt or copy at http:www.boost.org/LICENSE_1_0.txt)

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO
EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE
BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR
OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER
DEALINGS IN THE SOFTWARE.

Copyrights:

C) Copyright Jeremy Siek 2000

Copyright 1994 Hewlett-Packard Company

Copyright 1998 by Information Technology Industry Council

Copyright © 2007 Free Software Foundation, Inc

Copyright (C) Microsoft Corporation 1984-2002

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2002, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2002, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2001, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2001, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2002, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2002, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2000, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2000, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 1999, 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2002, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 1999, 2001, 2002, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 1999, 2001, 2003, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2001, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2001, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2002, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 1999, 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2003, 2009 Free Software Foundation

Copyright (C) 1999, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 1999, 2009 Free Software Foundation, Inc.

Copyright (C) 1999-2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 1999-2001, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002 Free Software Foundation

Copyright (C) 2000, 2001, 2002 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2003, 2007, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2009 Free Software Foundation

Copyright (C) 2000, 2001, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2002, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2001, 2009 Free Software Foundation

Copyright (C) 2000, 2002, 2003, 2004, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 2000, 2002, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 2000, 2002, 2003, 2009 Free Software Foundation

Copyright (C) 2000, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2004, 2005, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2005, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2002, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 2000, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2003, 2009 Free Software Foundation

Copyright (C) 2000, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2000, 2009 Free Software Foundation, Inc.

Copyright (C) 2001 Free Software Foundation, Inc.

Copyright (C) 2001 Free Software Foundation, Inc Benjamin Kosnik <bkoz@redhat.com>, 2001.

Copyright (C) 2001, 2002, 2003 Peter Dimov

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2004, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2005 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2005, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2003, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2005, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2005, 2008 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2005, 2009 Free Software Foundation

Copyright (C) 2001, 2002, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2005, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2005, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2002, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2003, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2003, 2005 Free Software Foundation, Inc.

Copyright (C) 2001, 2003, 2005, 2009 Free Software Foundation

Copyright (C) 2001, 2003, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2003, 2009 Free Software Foundation

Copyright (C) 2001, 2003, 2009 Free Software Foundation

Copyright (C) 2001, 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2004, 2005, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2001, 2009 Free Software Foundation

Copyright (C) 2001, 2009 Free Software Foundation, Inc.

Copyright (C) 2002 Peter Dimov

Copyright (C) 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2002, 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2002, 2003, 2004, 2005, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2002, 2003, 2004, 2005, 2009 Free Software Foundation

Copyright (C) 2002, 2003, 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2002, 2003, 2004, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2003, 2005 Free Software Foundation, Inc.
Copyright (C) 2002, 2003, 2005, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2003, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2003, 2009 Free Software Foundation
Copyright (C) 2002, 2003, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2004, 2005, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2004, 2005, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2004, 2006, 2008, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2004, 2009 Free Software Foundation
Copyright (C) 2002, 2004, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2005, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2002, 2009 Free Software Foundation
Copyright (C) 2002, 2009 Free Software Foundation, Inc.
Copyright (C) 2003 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation
Copyright (C) 2003, 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2006, 2009 Free Software Foundation
Copyright (C) 2003, 2004, 2005, 2006, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2007, 2008, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2007, 2009 Free Software Foundation
Copyright (C) 2003, 2004, 2005, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2005, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2006, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2004, 2009 Free Software Foundation
Copyright (C) 2003, 2004, 2009 Free Software Foundation, Inc.
Copyright (C) 2003, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2005, 2009 Free Software Foundation

Copyright (C) 2003, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2003, 2009 Free Software Foundation

Copyright (C) 2003, 2009 Free Software Foundation, Inc.

Copyright (C) 2004 Ami Tavory and Vladimir Dreizin, IBM-HRL.

Copyright (C) 2004, 2005 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2007, 2009 Free Software Foundation

Copyright (C) 2004, 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2004, 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2007, 2009 Free Software Foundation

Copyright (C) 2004, 2005, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2005, 2009 Free Software Foundation

Copyright (C) 2004, 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2006, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2004, 2009 Free Software Foundation

Copyright (C) 2004, 2009 Free Software Foundation, Inc.

Copyright (C) 2005 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2007, 2009 Free Software Foundation

Copyright (C) 2005, 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2005, 2008, 2009 Free Software Foundation

Copyright (C) 2005, 2009 Free Software Foundation

Copyright (C) 2005, 2009 Free Software Foundation, Inc.

Copyright (C) 2006 Free Software Foundation, Inc.

Copyright (C) 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2006, 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2006, 2007, 2009 Free Software Foundation

Copyright (C) 2006, 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2006, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2006, 2009 Free Software Foundation

Copyright (C) 2006, 2009 Free Software Foundation

Copyright (C) 2006, 2009 Free Software Foundation, Inc.

Copyright (C) 2006-2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2007, 2008, 2009 Free Software Foundation

Copyright (C) 2007, 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2007, 2009 Free Software Foundation

Copyright (C) 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2008, 2009 Free Software Foundation

Copyright (C) 2008, 2009 Free Software Foundation, Inc.

Copyright (C) 2009 Free Software Foundation

Copyright (C) 2009 Free Software Foundation, Inc.

Copyright (c) 1994 Hewlett-Packard Company

Copyright (c) 1996 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1996,1997 Silicon Graphics

Copyright (c) 1996,1997 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1996-1997 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1996-1998 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1996-1999 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1997 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1997-1999 Silicon Graphics Computer Systems, Inc.

Copyright (c) 1998 Silicon Graphics Computer Systems, Inc.

Copyright (c) 2001, 2002, 2003 Peter Dimov and Multi Media Ltd.

Copyright 2008 FSF

Copyright Ami Tavory and Vladimir Dreizin, IBM-HRL, 2004, and Benjamin Kosnik, Red Hat, 2004.

Copyright © 1999 The Open Group/The Institute of Electrical and Electronics Engineers, Inc.

Copyright © 1999 ISO

Copyright © 2000 Addison Wesley Longman, Inc.

Copyright © 2000 Addison Wesley, Inc.

Copyright © 2002 OOPSLA

Copyright © 2004, 2005, 2006, 2007 Free Software Foundation, Inc

Copyright © 2004, 2005, 2006, 2007 Free Software Foundation, Inc

Copyright © 2007 FSF

Copyright © 2007 Free Software Foundation, Inc.

Open Source Software: libgcc - 4.4.1

Enclosed you'll find the license conditions and copyright notices applicable for Open Source Software libgcc - 4.4.1

License conditions:

1 GCC is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3, or (at your option) any later version.
GCC is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
Under Section 7 of GPL version 3, you are granted additional permissions described in the GCC Runtime Library Exception, version 3.1, as published by the Free Software Foundation.
You should have received a copy of the GNU General Public License and a copy of the GCC Runtime Library Exception along with this program; see the files COPYING3 and COPYING.RUNTIME respectively. If not, see <http://www.gnu.org/licenses/>.

GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright (C) 2009 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyright.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyright requirements of the license of GCC.

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyright license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.
TERMS AND CONDITIONS

0. Definitions.

“This License refers to version 3 of the GNU General Public License.

“Copyright also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided),

that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and non-commercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

2 This configure script is free software; the Free Software Foundation
gives unlimited permission to copy, distribute and modify it.

Copyrights:

Copyright (C) 2003 Free Software Foundation, Inc.

Copyright (C) 2005, 2006, 2009 Free Software Foundation

Copyright (C) 2007, 2009 Free Software Foundation, Inc.

Copyright (C) 2008, 2009 Free Software Foundation, Inc.

Copyright 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Index

B

BaseUnit wiring, 91

C

Channel status LEDs, 117
Configuration
  channel configuration 1 (4 A) or 2 (2 A) channels), 102
  control system hardware/software options, 98
  diagnostics options, 105
  general parameters, 102
  I/O address space usage, 100
  I/O addresses, 109
  operating mode, 103
  overview, 99
  Parameter (channel parameters), 106
  potential group, 102
  reaction to CPU STOP, 103
  TIA Portal module selection, 101
Control interface, 110
Current control function, 79
  dithering and cycle time, 82
  limit reached flags, 81
  PID function diagram, 79
  PID parameters, 80
  resetting the PID controller, 82
  set point control, 81
Current measurement function, 77
  accuracy, 78

D

DC motor mode, 66
  control and feedback signals of PWM mode, 72
  input and output signals of PWM mode, 74
  isochronous mode, 70
  parameters of DC motor mode, 71
  setting and changing the period duration, 69
  setting and changing the pulse duty cycle and rotation direction, 69
  truth table, 68
DIAG LEDs, 117
Diagnostics
  error detection and diagnostics, 121
  options, 105
Direct control of a digital output, 87
Dither PWM output function, 83
  amplitude, 83
  example diagrams, 85
  period, 84
  ramp-up and ramp-down, 84

E

Errors
  error detection and diagnostics, 121
  parameter validation, 120

F

Feedback interface, 115
Firmware version, 117
Frequency output mode, 58
  control and feedback signals, 63
  input and output signals, 65
  isochronous mode, 61
  on-delay, 61
  output value (frequency), 60
  parameters, 62
  truth table, 60
Functional state (FS), 117
Functions
  current control, 79
  current measurement, 77
  direct control of a digital output, 87
  dither PWM output, 83
  high-speed output, 75
  isochronous mode, 86
  overview, 14
  sequence counter, 76

H

High-speed output function, 75

I

I/O address configuration, 109

I/O address space usage, 100
Interfaces
  control, 110
  feedback, 115
  SLOT, 113
isochronous mode
  Pulse output (single pulse) mode, 20
Isochronous mode
  DC motor mode, 70
  frequency output mode, 61
  On/Off-delay mode, 53
  overview, 86
  pulse width modulation (PWM) mode, 30

L

LD_SLOT, 113
LED status display, 117
Load wiring, 91

M

MODE_SLOT, 113
Modes
  DC motor, 66
  Frequency output, 58
  On/Off-delay, 47
  overview, 14
  pulse output (single pulse), 17
  pulse train, 37
  pulse width modulation (PWM), 25

O

On/Off-delay mode, 47
  control and feedback signals, 55
  input and output signals, 57
  isochronous mode, 53
  minimum pulse duration, 50
  Off-delay, 53
  On-delay, 53
  parameters, 54
  retriggering Off-delay, 52
  retriggering On-delay, 52
  truth table, 50
Open source software, 143
Overview
  TM Pulse 2x24V, 10

P

Parameter data record, 137
Parameter validation errors, 120
PID parameters, 80
Pin assignment, 91
Potential group, 102
Power source wiring, 91
Programming reference, 130
Pulse output (single pulse) mode, 17
  control and feedback signals, 22
  input and output signals, 24
  On-delay, 20
  parameters, 21
  pulse duration, 19
  sequence counter, 20
  truth table, 19
Pulse train mode, 37
  control and feedback signals, 44
  current measurement, 42
  duty cycle, 41
  input and output signals, 46
  on-delay, 40
  period duration, 40
  pulse count, 39
  truth table, 39
Pulse width modulation (PWM) mode, 25
  control and feedback signals, 34
  current control, 32
  current measurement, 32
  duty cycle, 29
  input and output signals, 36
  isochronous mode, 30
  minimum pulse duration, 27
  On-delay, 32
  parameters, 33
  period duration, 29
  setting the minimum pulse duration, 32
  truth table, 27

R

Reaction to CPU STOP, 103
Record 128 (module parameters), 137

S

sensor wiring, 91
Sequence counter, 76
SLOT, 113

T

Technical specifications
ambient temperature derating, 129
hardware/software, 124
programming reference, 130
TM Pulse 2x24V overview, 10
TM_CTRL_DQ signal, 87
Truth table
DC motor mode, 68
frequency output mode, 60
On/Off-delay mode, 50
pulse output (single pulse) mode, 19
pulse train mode, 39
Pulse width modulation (PWM) mode, 27

W

Wiring, 91
WRREC parameter write errors, 120