DELTA
PID Controller
Module

Operator's Manual

Jetter

Article # DPID HB1.0

05/1999 Printed in Germany

Edition: 1.0
May 1999

How to Contact us:

JETTER AG
Gräterstraße 2
D-71642 Ludwigsburg
Germany

Phone - Switchboard: 07141/2550-0
Phone - Technical Hotline: 07141/2550-444
Phone - Sales Hotline: 07141/2550-530

Fax: 07141/2550-425
e-mail: hotline@jetter.de
Internet: http://www.jetter.de

This Operating Instruction belongs to the PROCESS-PLC System DELTA:

Type: ______________________________

Serial #: ______________________________

Year of construction: ______________________________

Order #: ______________________________

To be entered by the customer:

Inventory #: ______________________________

Place of operation: ______________________________

Table of Contents

List of Appendices

1 Word of Advice on this Manual

1.1 Comment

This manual forms part of the DELTA PID controller module,

- and must be kept in a way that it is always at hand until the DELTA PID controller module will be disposed;
- If the module is sold, transferred or lent, this manual must be handed over.

In any case you encounter difficulties to clearly understand the manual, please contact the manufacturer.
We would appreciate any kind of suggestion and contributions on your part and would ask you to inform us or the write us. This will help us to produce manuals that are more user-friendly and to address your wishes and requirements.

Missing or inadequate knowledge of the manual results in the loss of any claim of liability on part of Jetter AG. Therefore, the operating company is recommended to have the instruction of the persons concerned confirmed- in writing.

Maintenance of the DELTA PID controller module

The DELTA PID controller module is maintenance-free. Therefore, for the operation of the module no inspection or maintenance are required.

Decommissioning and disposal of the DELTA PID controller module

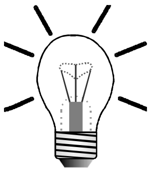
Decommissioning and disposal of the DELTA PID controller module are subject to the environmental legislation of the respective country in effect for the operator's premises.

1.2 Description of Symbols

This sign is to indicate a possible impending danger of serious physical damage or death.

This sign is to indicate a possible impending danger of light physical damage. This sign is also to warn you of material damage.

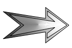
This sign is to indicate a possible impending situation which might bring damage to the product or to its surroundings.



You will be informed of various possible applications and will receive further useful suggestions.

• / - Enumerations are marked by full stops, strokes or scores.

Operating instructions are marked by this arrow.



Automatically running processes or results to be achieved are marked by this arrow.

(D) Illustration of PC and user interface keys.

2 Safety Instructions

2.1 General Safety Instructions

The DELTA PID controller module reflects the present state of the art. This DELTA PID controller module complies with the safety regulations and standards in force. Special emphasis was given to the safety of the users.

Of course, the following regulations apply to the user:

- relevant accident prevention regulations;
- accepted safety rules;
- EC guidelines and other country-specific regulations.

Usage as agreed upon

The DELTA PID controller module consists of four PID controllers observing the process with a minimum sampling period of 2 ms. These PID controllers can be parameterised at will.
The supply voltage of the DELTA PID controller module is 24 V DC. This operating voltage is classified as SELV (Safety Extra Low Voltage). Thus, the DELTA PID controller module is not subject to the "Low Voltage“ Directive (LVD) of the EC.

The DELTA PID controller module must be operated within the limits of the specified technical data only.

Usage other than agreed upon

The DELTA PID controller module must not be used in technical systems which to a high degree have to be fail save, e.g. ropeways and aeroplanes.

If the DELTA PID controller module is to be run under surrounding conditions, which differ from the conditions mentioned in chapter 4, the manufacturer is to be contacted beforehand.

Who is permitted to operate the DELTA PROCESS PLC?

Only instructed, trained and authorised persons are permitted to operate the DELTA PROCESS PLC.

Mounting, backfitting, maintenance and repair may only be carried out by specially trained personnel, as specific know-how will be required.

- Isolate the DELTA PROCESS PLC from the mains (pull out the mains plug) when working on the control system.

Modifications and alterations to the module

Important!

Due to safety reasons, no modifications and alterations to the DELTA PID controller module and its functions are allowed. Any modifications to the module not expressly authorised by JETTER AG will result in a loss of any liability claims to Jetter AG.

The original parts are specifically designed for the DELTA PID controller module. Parts and equipment of other manufacturers are not tested on our part, and are, therefore, not released by us. The installation of such parts may impair the safety and the proper functioning of the DELTA PID controller module.

For any damages resulting from the use of non-original parts and equipment any claims with respect to the liability of Jetter AG are excluded.

Malfunctions

- Malfunctions or other damages are to be reported to an authorised person at once.
  The DELTA PROCESS PLC must be protected from improper or inadvertent use.

  Only qualified experts are allowed to carry out repairs.

Information Signs and Labels

- Writings, information signs, and labels always have to be observed and kept readable.
- Damaged or unreadable information signs and labels are to be exchanged.

Earthing Procedure

- Screw down the DELTA basic housing to a highly conductive and earthed plate.
- On the top face of the DELTA basic housing an earth-terminal bolt with a M4 thread is located.
  This earthing bolt must electrically be connected to a PE terminal by means of a PE conductor (conductor area: 1,5 mm^2, colour: green-yellow) (refer to Fig. 1).

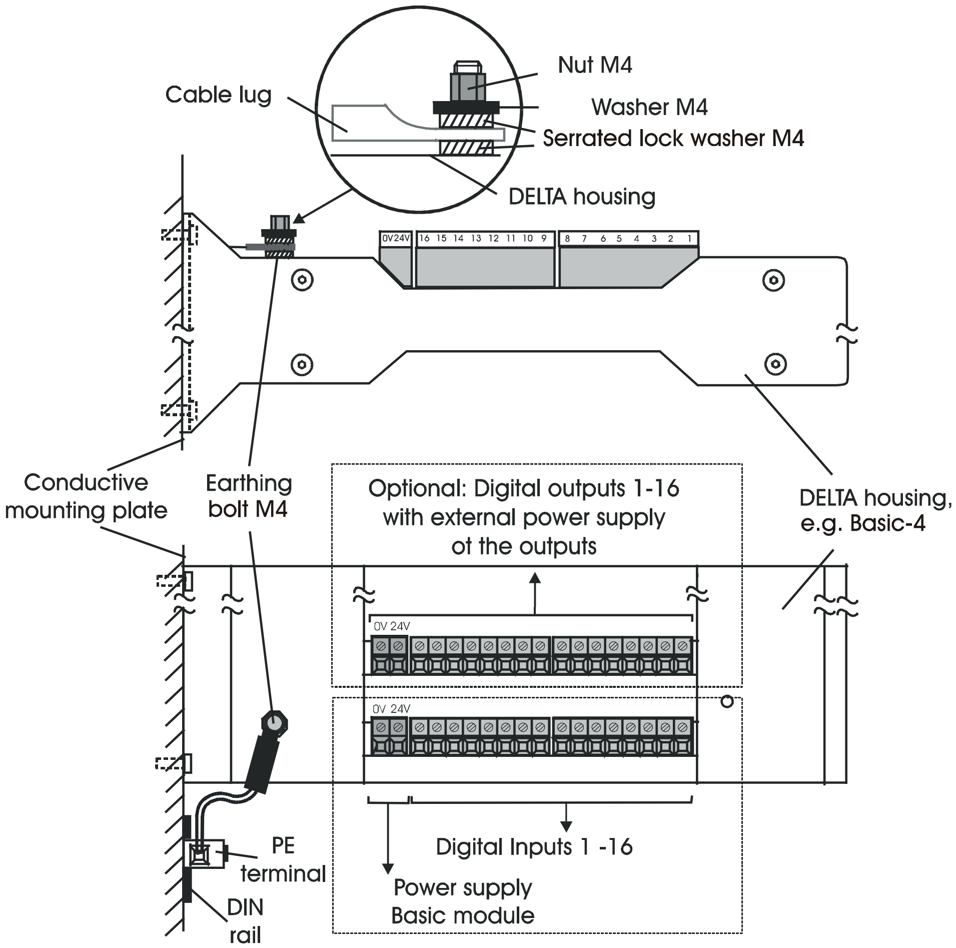


Fig. 1: Earthing the DELTA basic housing

2.2 Instructions on EMI

The noise immunity of a system corresponds to the weakest component of the system.

For this reason, correct wiring and shielding of the cables is important.

Important!

Measures for increasing immunity to interferences:

- Shield both sides of the cable.

- The entire shield must be drawn behind the isolation, and then be extensively clamped under an earthed strain relief.

When male connectors are used:

- Only use metallised connectors, e.g. SUB-D with metallised housing. Please take care of direct connection of the strain relief with the housing here as well (refer to Fig. 2).

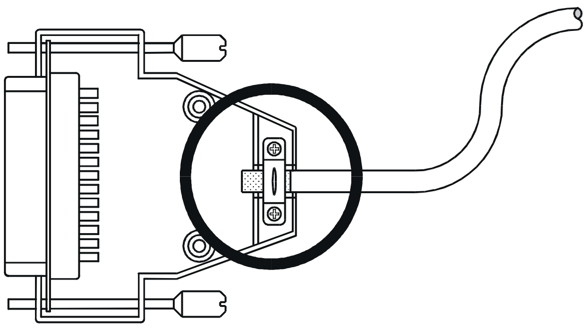


Fig. 2: Shielding of SUB-D connectors in conformity with the EMC standards.

- On principle, separate signal and voltage connections spatially.

- It is of great importance that the DELTA basic housing is screwed down to a highly conductive mounting plate.

3 Mounting Dimensions

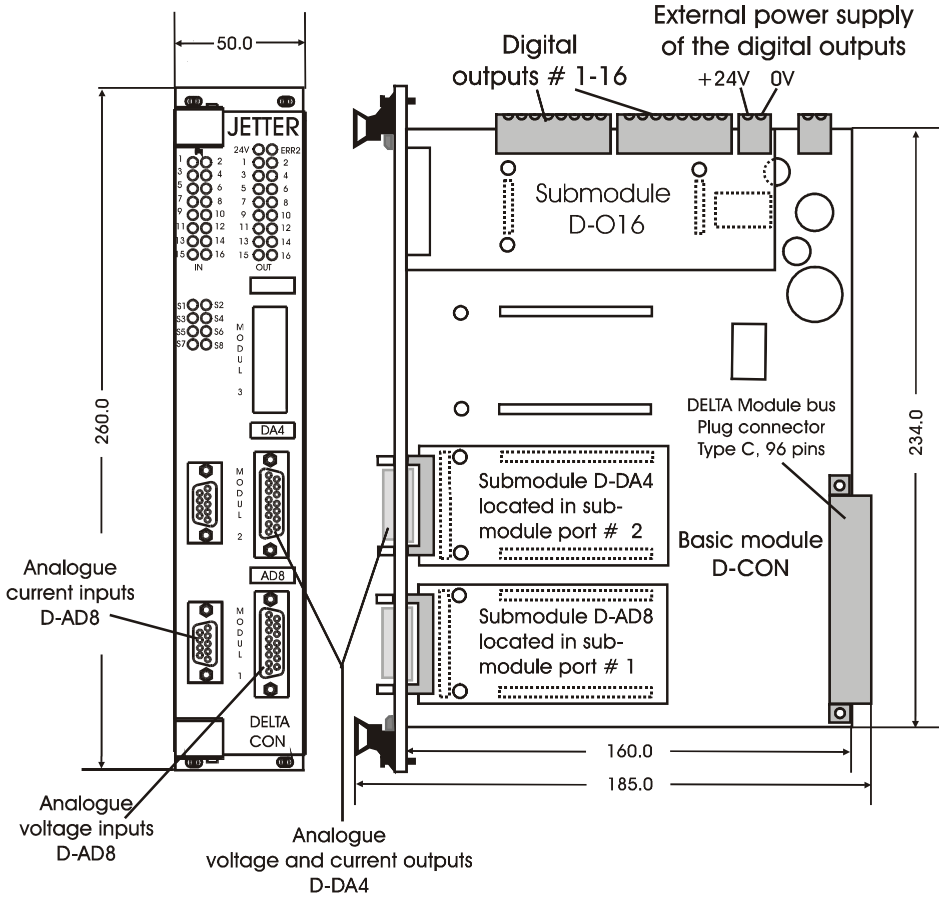


Fig. 3: Front and top view of the DELTA PID controller module

Design	
Dimensions (H x W x D in mm)	260 x 50 x 185
Components of the DELTA PID controller module Basic module D-CON with digital inputs Submodule D-AD8 with analogue inputs Submodule D-DA4 with analogue outputs Submodule D-O16 with PWM outputs	 This module can be plugged into slot # 1 to 8 of the DELTA basic housing. This module can be plugged into submodule port # 1 of the basic module D-CON This module can be plugged into submodule port # 2 of the basic module D-CON This module can be plugged onto the basic module D-CON

4 Technical Data

Functional Data	
Quantity of PID controllers	4
Parameterisation	Each PID controller can be parameterised at will Parameters: - P gain - Integral-action time T_N - Derivative-action time T_V - Sampling interval - Integral-action limitation - Slew rate limitation
Sampling interval:	Minimum sampling interval of 2 ms. With 4 activated controllers it is 8 ms per controller.
Controller Input	
Analogue inputs (controlled variable)	1 differential-mode input per activated PID controller The analogue inputs of each deactivated controller are available for direct current and voltage measurements It is possible to measure two single-ended voltages (voltage to ground) instead of one differential-mode voltage For additional data see chapter „D-AD8 Submodule“
Controller Output	
Output combinations	I: Analogue output channel # 1-4 II: PWM output channel # 1-4 III: Analogue output channel # 1-2 PWM output channel # 3-4 IV: PWM output channel # 1-2 Analogue output channel # 3-4

Functional Data	
Analogue output (manipulated variable)	4 single-ended channels (voltage to ground) Each channel can be used as voltage or current output Per deactivated controller one output channel is available enabling the output of a voltage or current. For additional data see chapter "D-DA4 Submodule“
PWM outputs Digital output # 1-8 of the D-O16 submodule	4 output channels Per channel one PWM+ and PWM- signal For additional data see chapter "D-O16 Submodule“

Environmental Operating Conditions The basic module D-CON is installed in the DELTA housing D-Basic	
Ambient Temperature	Operation: +20 °C to +50 °C Storage: -10 °C to +70 °C
Relative humidity	5 to 95%, no condensation RH2 acc. to IEC 61131-2
Protective system	IP 20
Category of protection	III acc. to IEC 61131-2
Contamination level	II acc. to IEC 61131-2
Electromagnetic Compatibility	EMI is met by appropriate filtering and shielding: Interference acc. to EN 61000-4-2 severity class 4 EN 61000-4-4 severity class 4 EN 61000-4-6 severity class 3 Emitted interference acc. to EN 55011 group 1, class B
Installation	Position: The module is to be

Environmental Operating Conditions The basic module D-CON is installed in the DELTA housing D-Basic	
	placed vertically so that cooling air can flow upward and remove heat
Oscillation fatigue limit	acc. to IEC 61131-2 and IEC 68 part 2-6

5 Functional Description

The design of the DELTA PID controller module is shown in Fig. 4.

With the help of this module voltages and currents can be measured. This module allows commercial sensors to be connected to it. The inputs can be configured for a voltage range of -10 V up to +10 V, resp. for current ranges of -20 mA up to +20 mA and +4 mA up to +20 mA.

To the 8 inputs which can be operated either as single-ended or as differential inputs a register set is assigned in which the registers are specified functionally referring to the analogue inputs.

In theses registers individual inputs are allocated to one of the four PID controllers in order to perform actual-value acquisition.

Now, 4 PID controllers follow which are independent of each other, and which can be parameterised at will. The registers necessary for this purpose are exemplary shown for the first PID controller. Above the controllers is given a description of the algorithm according to which the controllers calculate a manipulated variable on the basis of the present actual value dependent on a given setpoint.

Finally, in the output register set all relevant registers for the analogue outputs, resp. PWM outputs are given. As described under item "Output Combinations“ in Fig. 4 in these registers several combinations are possible, too.
The allocation between the outputs and the 4 PID controllers is made in one of these registers in order to output a manipulated variable.

There is a 0 to 20 mA and a -10 to +10 V (maximum current 20 mA) interface available serving as interface to the actuators. In case this quantity is not sufficient, the PWM output can be used. It provides a current of 0,5 A which allows the operation of a small heating or cooling system.

In Fig. 4: Block diagram of the DELTA PID controller module below the 4 PID controllers the registers for global configuration and control of the DELTA PID controller module are shown.

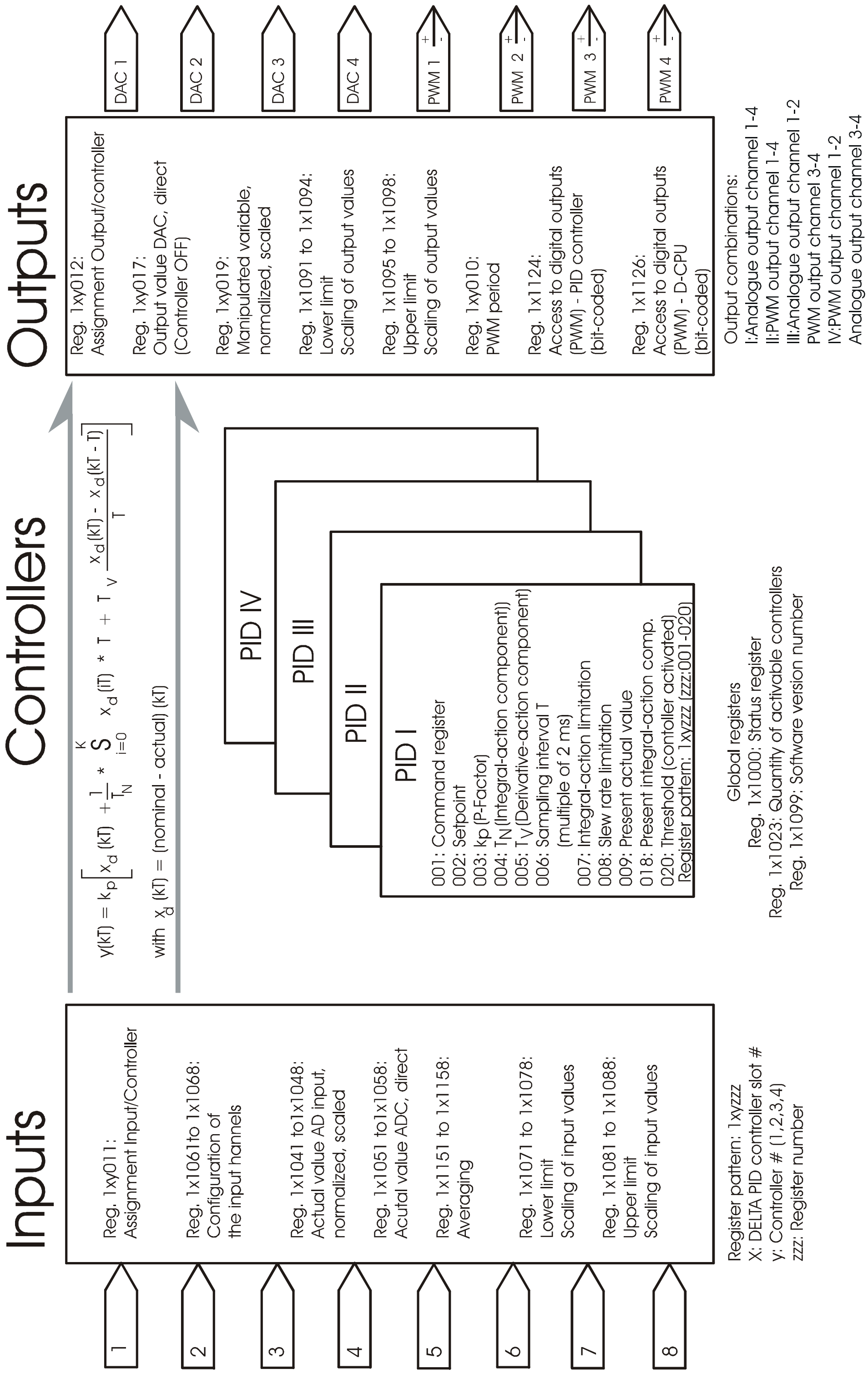


Fig. 4: Block diagram of the DELTA PID controller module

6 Power Supply

6.1 Requirements

Power supply unit requirements	
Voltage range:	20 ... 30 V DC Residual ripple < 5 % filtered
Power consumption without digital outputs	approx. 10 W

6.2 Contact Assignment of Terminals

The power supply terminals of the DELTA PID controller module are located on the top side of the DELTA basic housing. For this purpose, green Phoenix combicon screw-clamping terminals with a contact spacing (pitch) of 5.08 are used.

In Fig. 5 you will find an explanation of the contact assignment. In the left-hand column the supply terminal of the DELTA PID controller module and the digital inputs are shown.

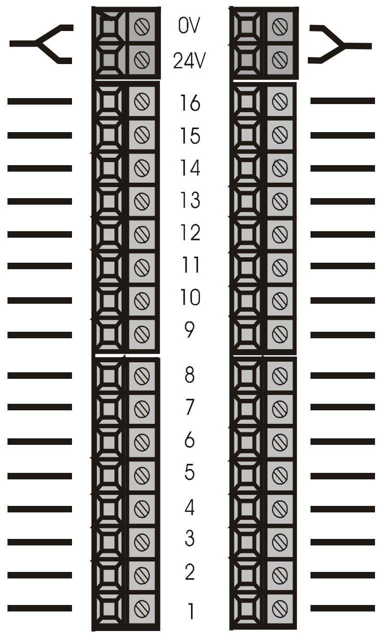


Fig. 5: Contact assignment of the power supply terminal

Power supply of the DELTA PID controller module		
Terminal	Signal	Comment
24V	24 V DC	
0V	GND	

Important!

Following installation of the DELTA PID controller module in the DELTA basic housing be sure to supply this module with voltage. If you fail to do so the control system will not be ready for operation.

7 Digital Inputs

7.1 Technical Data

Digital inputs form an integral part of the D-CON basic module.

Functional Data	
Input quantity	16 digital inputs
Nominal voltage	24V DC

Electrical Data	
Voltage range	15 ... 27 V DC
Signal voltage ON	min. 15 V
Signal voltage OFF	max. 10 V
Input current	approx. 8mA
Input resistance	3.0 kΩ
Input delay time	approx. 3 ms
Electrical isolation	None

7.2 Description of LEDs

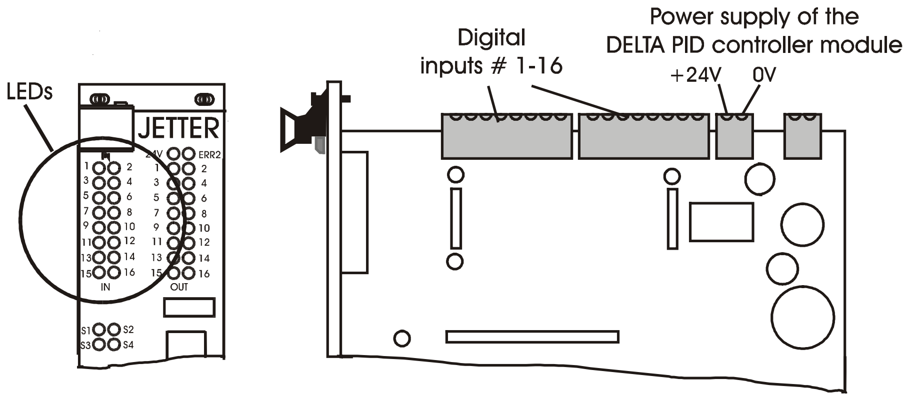


Fig. 6: LEDs of the digital inputs

LEDs of the digital inputs		
Designation	Colour	Function
IN 1 ... 16	yellow	Digital input 1 to 16 **ON:** Signal voltage ON **OFF:** Signal voltage OFF

7.3 Contact Assignment

The terminals of the digital inputs on the basic module D-CON (refer to Fig. 6) are located on the top side of the DELTA basic housing. For this purpose, green Phoenix combicon screw-clamping terminals with a contact spacing (pitch) of 5.08 are used.

In Fig. 7 you will find an explanation of the contact assignment. In the right-hand column (outputs) the terminals of the D-O16 module are shown. In the left-hand column (inputs) the terminals of the digital inputs of the basic module D-CON are shown.

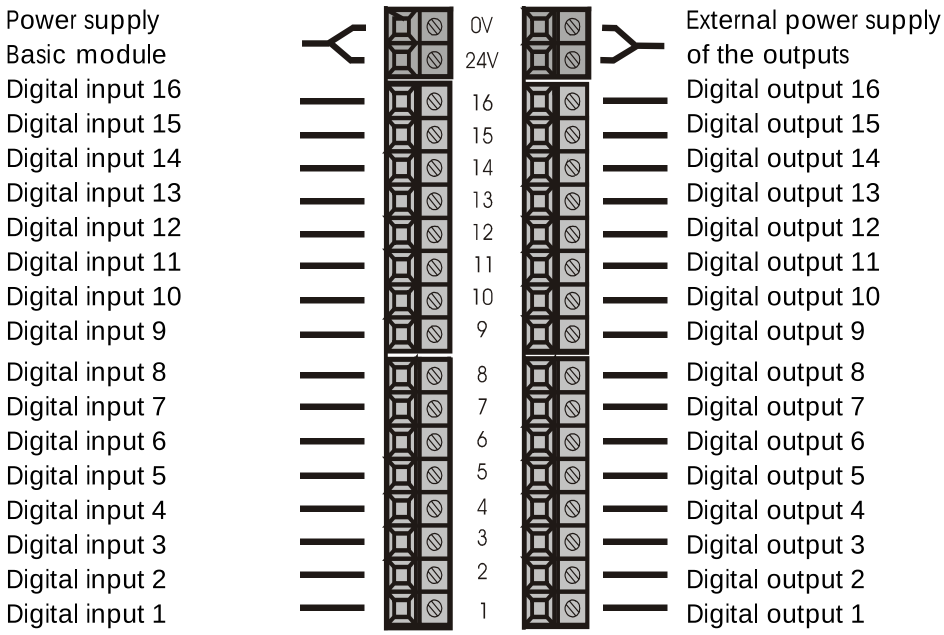


Fig. 7: Contact assignment of the digital input terminal

In Fig. 8 the assignment of the digital inputs of the basic module D-CON is shown. Reference point is the 0 V terminal to which the 0 V signal is connected-up.

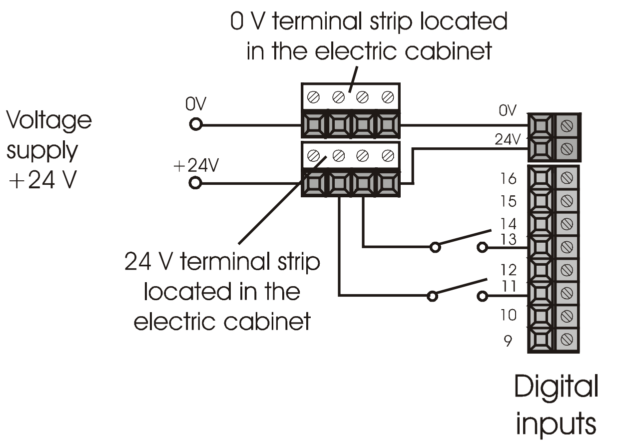


Fig. 8: External circuit of the digital inputs 11 and 13

7.4 Numbering of digital inputs

Numbering of digital inputs	
Input D-CON (DELTA PID controller module)	Number
Input 1	Slot * 100 + 1
Input 2	Slot * 100 +2
...	...
Input 16	Slot * 100 +16

8 Submodule D-O16 (digital outputs)

8.1 Technical Data

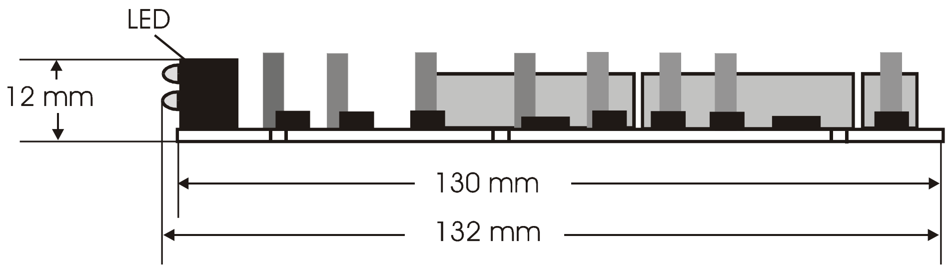


Fig. 9: Side view of the D-O16 submodule

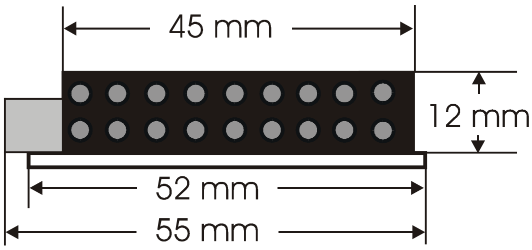


Fig. 10: Front view of the D-O16 submodule

Design	
Dimensions (H x W x D in mm)	12.0 x 52.0 x 130.0
Mounting	This module can be plugged into basic module D-CON (refer to Fig. 3)

Functional Data	
Quantity of outputs	16 digital outputs
Nominal voltage	24V DC
Type of outputs	Transistor, pnp

Electrical Data	
External power supply Voltage range	required 20 ... 30 V DC
Signal voltage ON	typically V_{supply} -0.5 V
Signal voltage OFF	typically 0.8 V
Maximum load current	0.5 A per output
Electrical isolation	None
Protective circuit	Overload, overvoltage, overtemperature A fault trip will be indicated with the red LED ERR2
Protection against inductive loads	Yes

8.2 Description of LEDs

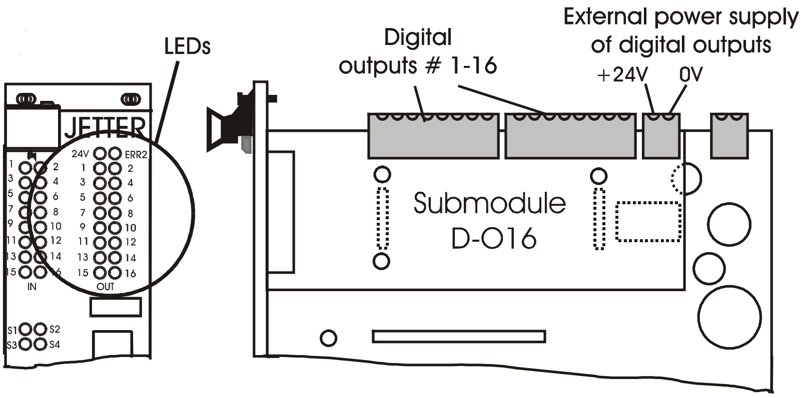


Fig. 11: LEDs of the D-O16 submodule

LEDs of the D-O16 submodule		
Designation	Colour	Function
OUT 1 ... 16	yellow	Digital output 1 to 16 **ON:** Signal voltage IN **OFF:** Signal voltage OUT
ERR2	red	**ON:** Overload, overtemperature, cable breakage of one or more outputs.
24V	green	**ON:** External voltage supply of the digital outputs is provided.

8.3 Contact Assignment

The terminals of the digital outputs on the D-O16 are located on the top side of the DELTA basic housing (see Fig. 1 and Fig. 3) For this purpose, green Phoenix combicon screw-clamping terminals with a contact spacing (pitch) of 5.08 are used.

In Fig. 12 you will find an explanation of the contact assignment. In the right-hand column (outputs) the terminals of the D-O16 are shown. In the left-hand column (inputs) the terminals of the digital inputs of the basic module D-CON are shown.

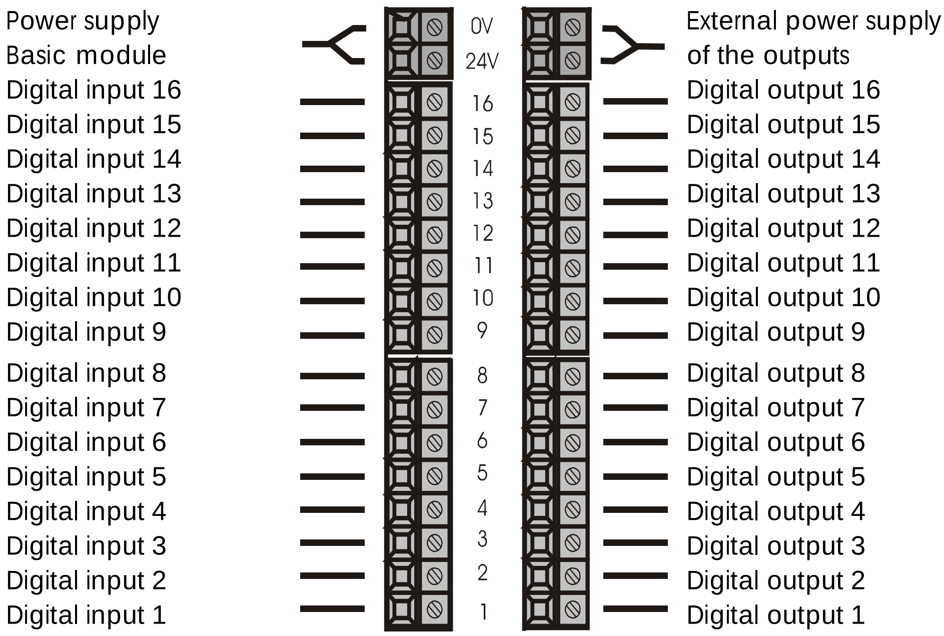


Fig. 12: Contact assignment of the digital output terminal

In Fig. 13 the external circuit of the digital outputs of the D-O16 module is shown. The 0 V terminal to which the 0 V signal is connected-up is located in the electric cabinet.

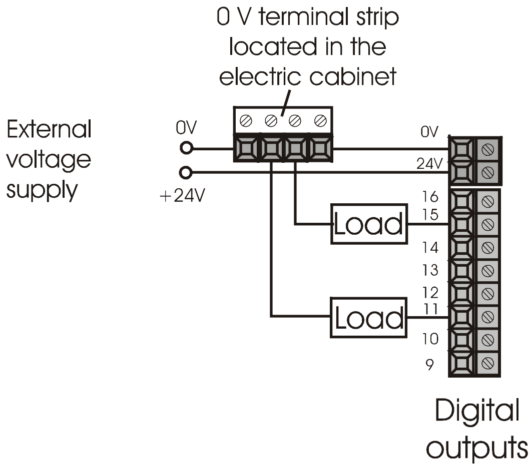


Fig. 13: External circuit of the digital outputs 11 and 15

8.4 Access to the Digital Outputs

8.4.1 Global access by the D-CPU

With the SYMPAS instruction "output number“ the digital output is set, resp. reset directly.

It is prerequisite that global access to these digital outputs is enabled (refer to chapter 8.4.3).

Numbering of digital outputs on the D-O16 submodule	
Output D-O16	Number
Output 1	Slot * 100 + 1
Output 2	Slot * 100 +2
...	...
Output 16	Slot * 100 +16

8.4.2 Local access by the DELTA PID controller module

The PWM outputs of the PID controllers are directly assigned to the digital outputs of the submodule D-O16.

It is prerequisite that local access to these digital outputs is enabled (refer to chapter 8.4.3).

Controller	Signal	Output D-O16
1	PWM+	1
1	PWM-	2
2	PWM+	3
2	PWM-	4
3	PWM+	5
3	PWM-	6

Controller	Signal	Output D-O16
4	PWM+	7
4	PWM-	8

Below follows a description of an application for the PWM signal. This description is about temperature control.

In case the manipulated variable is > 0 the actual temperature is too low. Depending on the manipulated variable a PWM+ signal is generated which is used to operate a heating coil. The width of the pulse depends on the manipulated variable.

In case the manipulated variable is < 0 the actual temperature is too high. Depending on the manipulated variable a PWM- signal is generated which is used to operate a cooling system. The width of the pulse depends on the manipulated variable.

In Fig. 14 the PWM signal is shown together with the manipulated variable.

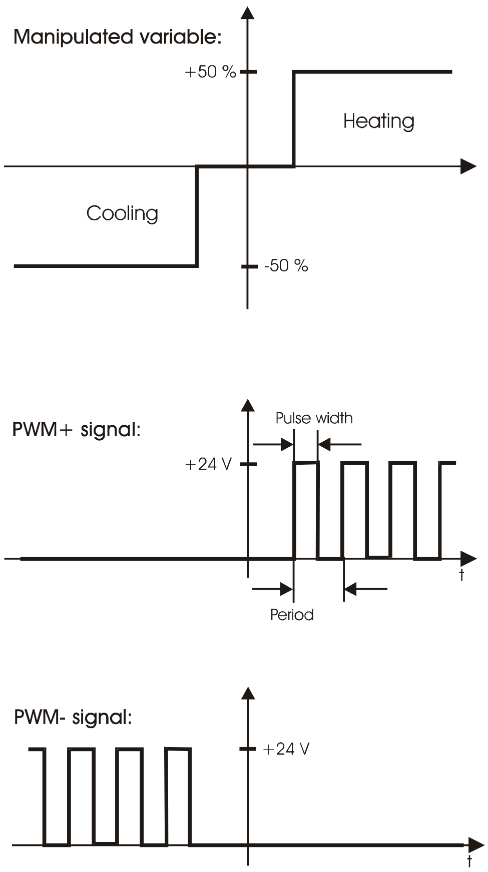


Fig. 14: PWM signal depending on the manipulated variable

8.4.3 Access rights

In order to use the PWM outputs local access to the digital outputs must be enabled.

The local access can be enabled through register 1x1124 (see chapter „Firmware“). This register is bit-coded. If the bits 0 to 7 are set the PWM signals of the four controllers are connected through to the digital outputs. Resetting the bits 0 to 7 will disable the outputs. In Fig. 15 the function of the bits 0 and 1 is shown.

If it is required that a digital output is to be set or reset by the D-CPU global access to the digital outputs must be enabled. Setting and resetting of a digital output by the D-CPU is carried out with the SYMPAS instruction "output number“.

The global access can be enabled through register 1x1126 (see chapter "Firmware“). This register is bit-coded. Through setting the bits 0 to 15 global access is enabled. Resetting the bits 0 to 15 will disable the outputs. In Fig. 15 the function of the bits 0 and 1 is shown.

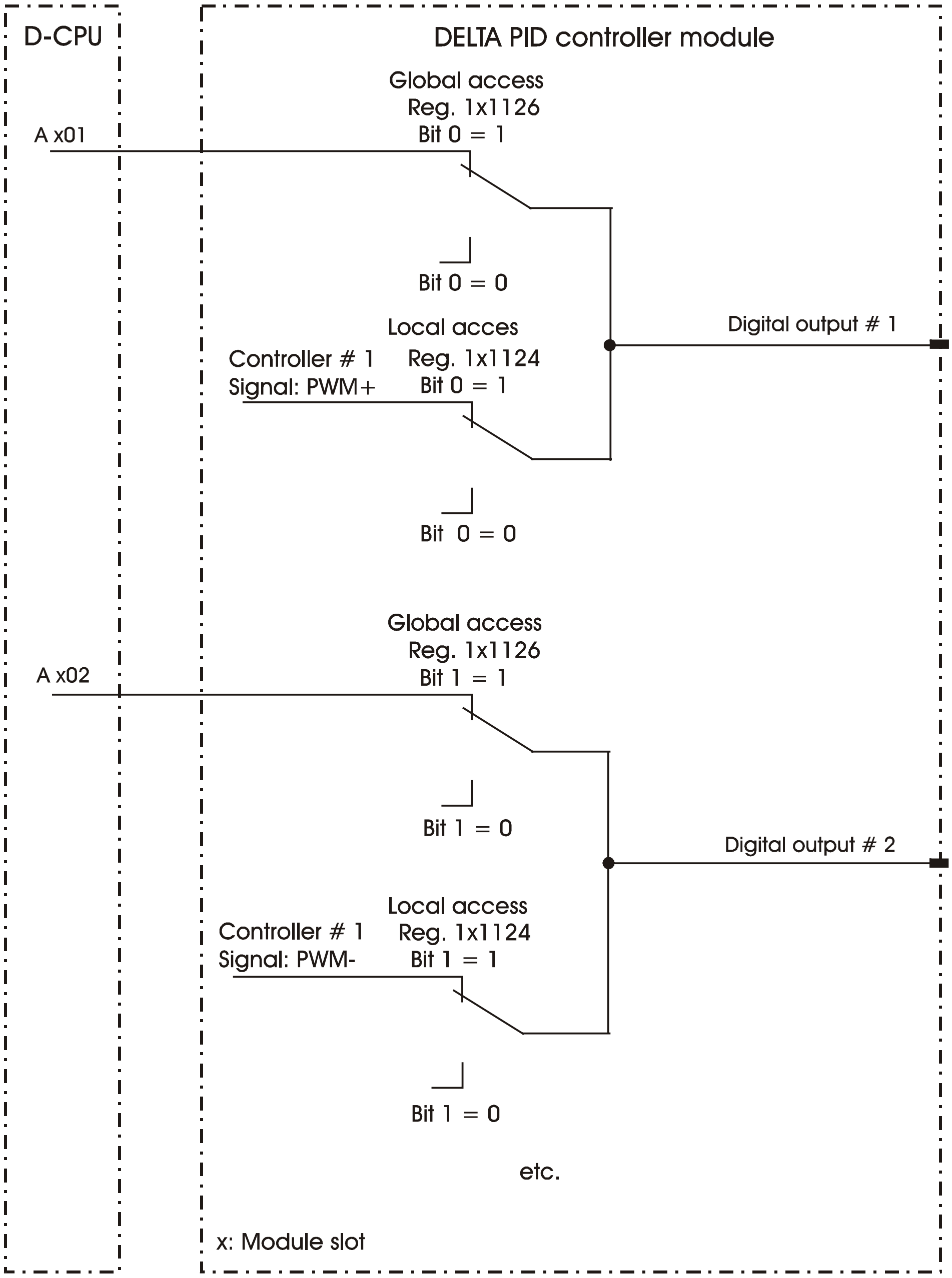


Fig. 15: Global and local access to digital outputs

9 Submodule D-AD8 (analogue input)

9.1 Technical Data

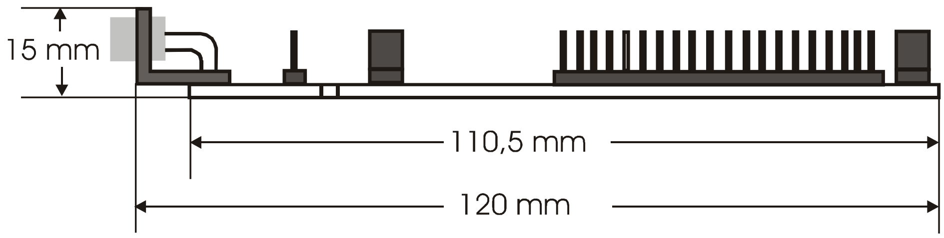


Fig. 16: Side view of the D-AD8 submodule

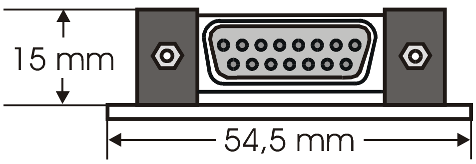


Fig. 17: Front view of the D-AD8 submodule

Design	
Dimensions (H x W x D in mm)	15.0 x 54.5 x 110.5
Mounting	This module can be plugged into submodule port # 1of the basic module D-CON (cf. Fig. 3)

Functional Data	
Input quantity	Voltage channels: max. 8 single-ended channels max. 4 differential-mode channels Current channels: max. 4 single-ended channels max. 4 differential-mode channels

Functional Data	
	Configurable channel by channel Cyclic conversion of 1 to 8 voltages (depending on input configuration)
Resolution	16 Bit
Voltage range Value range	-10 V ... +10 V -32768 ... +32767
Current range 1 Value range	-20 mA +20 mA -32768 ... +32767
Current range 2 Value range	4 mA +20 mA 6554 ... +32767
Sampling interval	Minimum sampling interval of 2 ms. When 4 controllers are activated the total sampling interval is 8 ms
Absolute error (Voltage)	max. 0.3 %
Absolute error (Current)	max. 0.4 %

Electrical Data	
Voltage supply D-AD8 +24 V and +/-15 V	This module can be plugged into submodule port # 1 of the basic module D-CON
Input impedance - Voltage: - Current:	 55 kΩ 100 Ω
Electrical isolation	None
The D-AD8 module provides	+/- 15 V / 5 mA

9.2 Input Circuit

9.2.1 Single-ended voltage channel

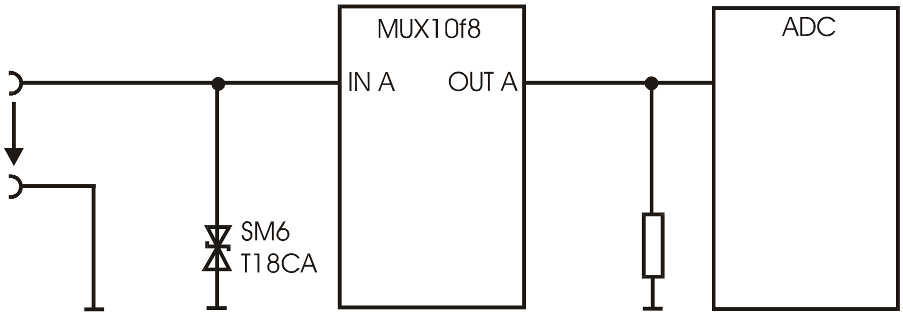


Fig. 18: Single-ended voltage channel

9.2.2 Differential-mode voltage channel

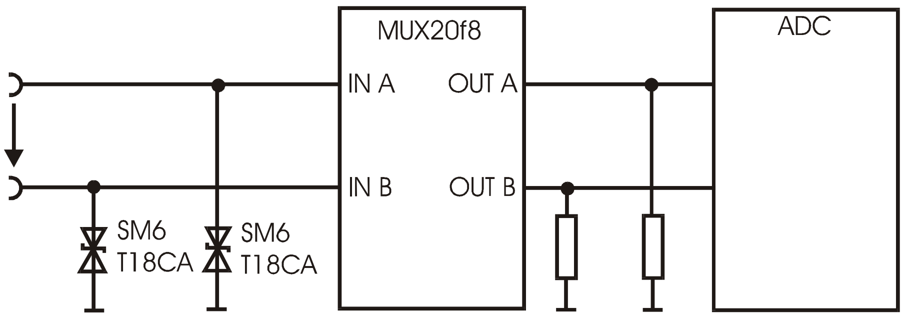


Fig. 19: Differential-mode voltage channel

9.2.3 Single-ended current channel

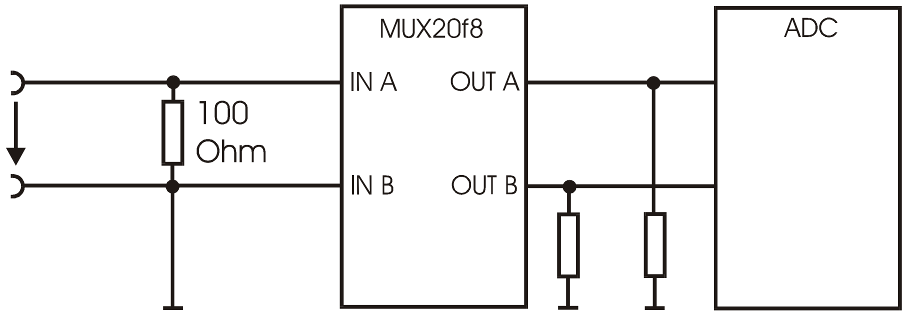


Fig. 20: Single-ended current channel

9.2.4 Differential-mode current channel

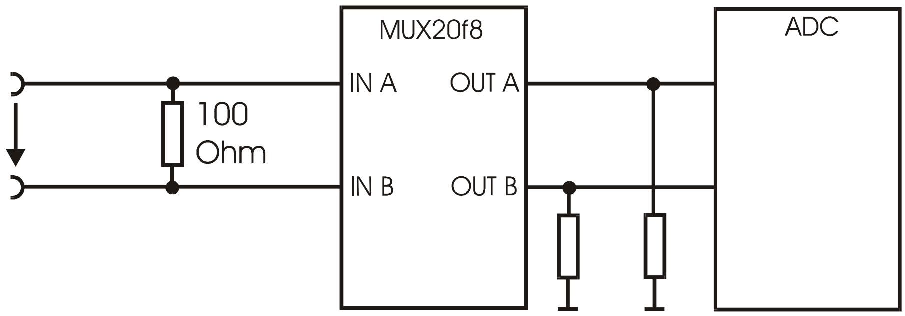


Fig. 21: Differential-mode current channel

9.3 Configuration of the Analogue Inputs

The ADC performs cyclic conversion of 1 to 8 voltages.

Through a configuration (input configuration) the following parameters can be specified:

- Voltage channel, single-ended (voltage to ground)
- Differential-mode voltage channel
- Current channel, single-ended (voltage to ground)
- Differential-mode current channel

- Voltage range: -10 ... +10 V
- Current range: -20 ... +20 mA
- Current range: 4 ... 20 mA

Only voltages (currents) of the configured inputs will be converted.

1 to 8 configurations are possible. Each configuration is written into a register.

The number and the kind of possible configurations depend on the quantity of voltages (currents) to be converted.

Reason:
Since every voltage or current measurement requires a configuration, the following configurations are possible:

- max. 8 single-ended voltage channels
- max. 4 differential-mode voltage channels
- max. 4 single-ended current channels;
- max. 4 differential-mode current channels;
- or a combination out of them.

Per current channel one differential voltage less can be converted.

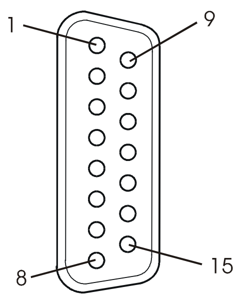
Input configuration		
		Register value
Single-ended	-20 .. +20 mA	3
	4 .. 20 mA	17
	-10 .. +10 V	8
Differential	-20 .. +20 mA	7
	4 .. 20 mA	21
	-10 .. +10 V	12

In chapter 9.4.1 an example of the input configuration is given.

9.4 Description of Connections

9.4.1 Analogue Voltage Channels

Pin Assignment - Female connector SUB-D, 15 pins



Pin	Signal: Single-ended	Signal: Differential	Comment
1	GND		Ground
2	IN1	IN1 A	Analogue input
3	IN2	IN2 A	Analogue input
4	IN3	IN3 A	Analogue input
5	IN4	IN4 A	Analogue input
6	IN5	IN1 B	Analogue input
7	IN6	IN2 B	Analogue input
8	IN7	IN3 B	Analogue input
9	IN8	IN4 B	Analogue input
10	not assigned		
11	+15V		Loadability: 5 mA
12	-15V		Loadability: 5 mA
13	GND		Ground
14	not assigned		
15	not assigned		

Important!

Do not connect any voltage sources to pin 1, 11, 12 and 13. This will result in damages to the product.

Example of an input configuration

Input 1:	Differential voltage (IN1 A)
Input 2:	Differential voltage (IN2 A)
Input 3:	Differential voltage (IN3 A)
Input 4:	Single-ended voltage (IN4)
Input 5:	Differential voltage (IN1 B)
Input 6:	Differential voltage (IN2 B)
Input 7:	Differential voltage (IN3 B)
Input 8:	Single-ended voltage (IN8)

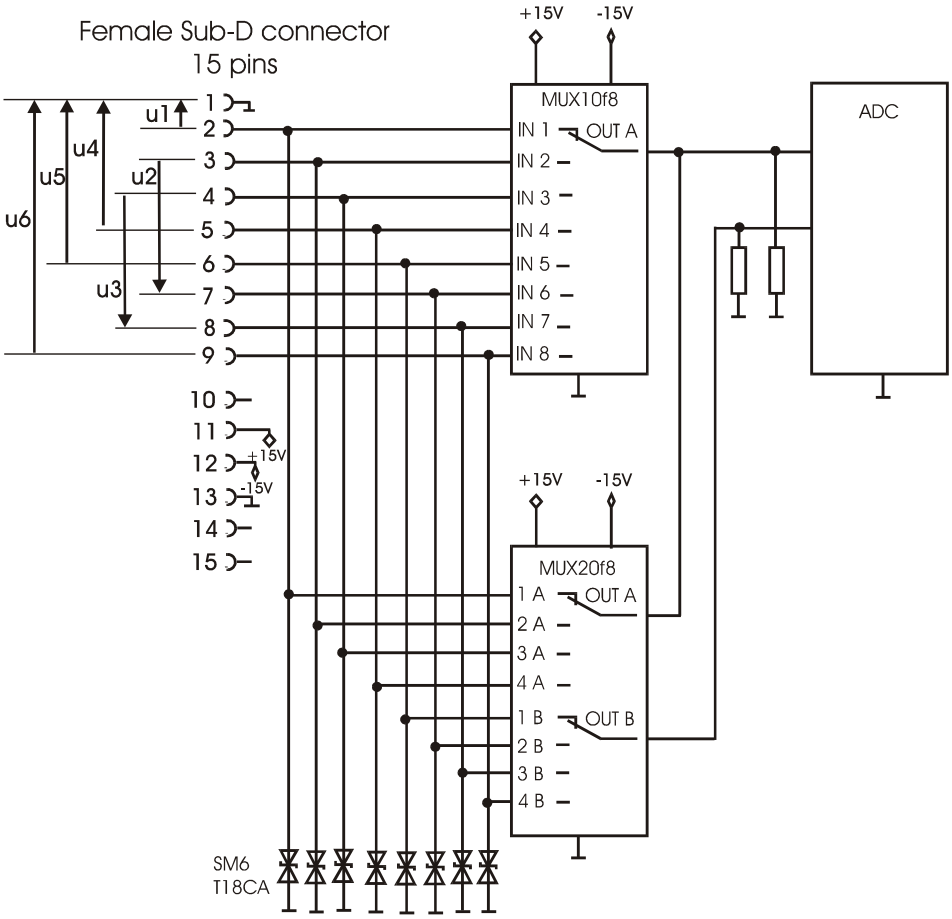
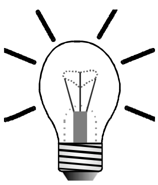


Fig. 22: Voltage channels of the D-AD8 submodule



Note!

In this example the PID controller # 4 is switched off.
Resulting from this, the inputs 4 and 8 are released. Through these inputs 2 single-ended voltages (voltage u4 and u5) are measured (voltage to ground). This configuration is shown in Fig. 22.

In the given case, altogether 5 configurations of input channels are required.

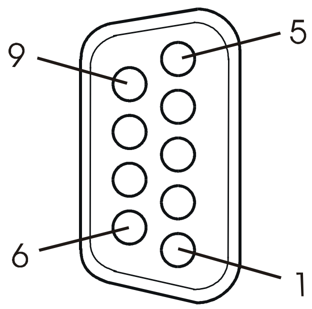
Register	Register value	Input configuration
Configuration of AD channel # 1	12	Differential -10 ... +10V
Configuration of AD channel # 2	12	Differential -10 ... +10V
Configuration of AD channel # 3	12	Differential -10 ... +10V
Configuration of AD channel # 4	8	Single-ended -10 ... +10V
Configuration of AD channel # 8	8	Single-ended -10 ... +10V

Registers with converted digital value	Analogue signal
Direct actual value - ADC channel # 1	Voltage u1
Direct actual value - ADC channel # 2	Voltage u2
Direct actual value - ADC channel # 3	Voltage u3
Direct actual value - ADC channel # 4	Voltage u4
Direct actual value - ADC channel # 8	Voltage u5

9.4.2 Analogue Current Channels

Basic module D-CON - Submodule port # 1

Pin Assignment - Female connector SUB-D, 9 pins



Pin	Signal	Comment
	Differential	
1	GND	Ground
2	IN4 B	Analogue input
3	IN3 B	Analogue input
4	IN2 B	Analogue input
5	IN1 B	Analogue input
6	IN4 A	Analogue input
7	IN3 A	Analogue input
8	IN2 A	Analogue input
9	IN1 A	Analogue input

Note!

The differential current channel can be converted into a single-ended current channel by connecting pins 2, 3, 4 resp. 5 to GND.

9.4.3 Jumper Settings

By inserting specific jumpers on the D-AD8 submodule current channels can be allocated to the female Sub-D connector, 15 pins, located on the D-AD8 submodule.
This will be necessary, if

- there is no female Sub-D connector, 9 pins, available at this port.

However, this does not apply to the DELTA PID controller module which means that no jumper setting is required.

Allocation of current channels to the female Sub-D connector, 15 pins The following jumpers have to be inserted				
Current channel	Mode	Jumper		
1	Single-ended	X4.1-2	and	X6
	Differential	X4.1-2	and	X4.9-10
2	Single-ended	X4.3-4	and	X7
	Differential	X4.3-4	and	X4.11-12
3	Single-ended	X4.5-6	and	X8
	Differential	X4.5-6	and	X4.13-14
4	Single-ended	X4.7-8	and	X9
	Differential	X4.7-8	and	X4.15-16

Example of an input configuration for 4 PID controllers

Controller 1:
Input 1: Differential-mode current (i1 A)
Input 5: Differential-mode current (i1 B)

Controller 2:
Input 2: Differential-mode current (i2 A)
Input 6: Differential-mode current (i2 B)

Controller 3:
Input 3: Differential-mode voltage (u3 A)
Input 7: Differential-mode voltage (u3 B)

Controller 4:
Input 4: Differential-mode voltage (u4 A)
Input 8: Differential-mode voltage (u4 B)

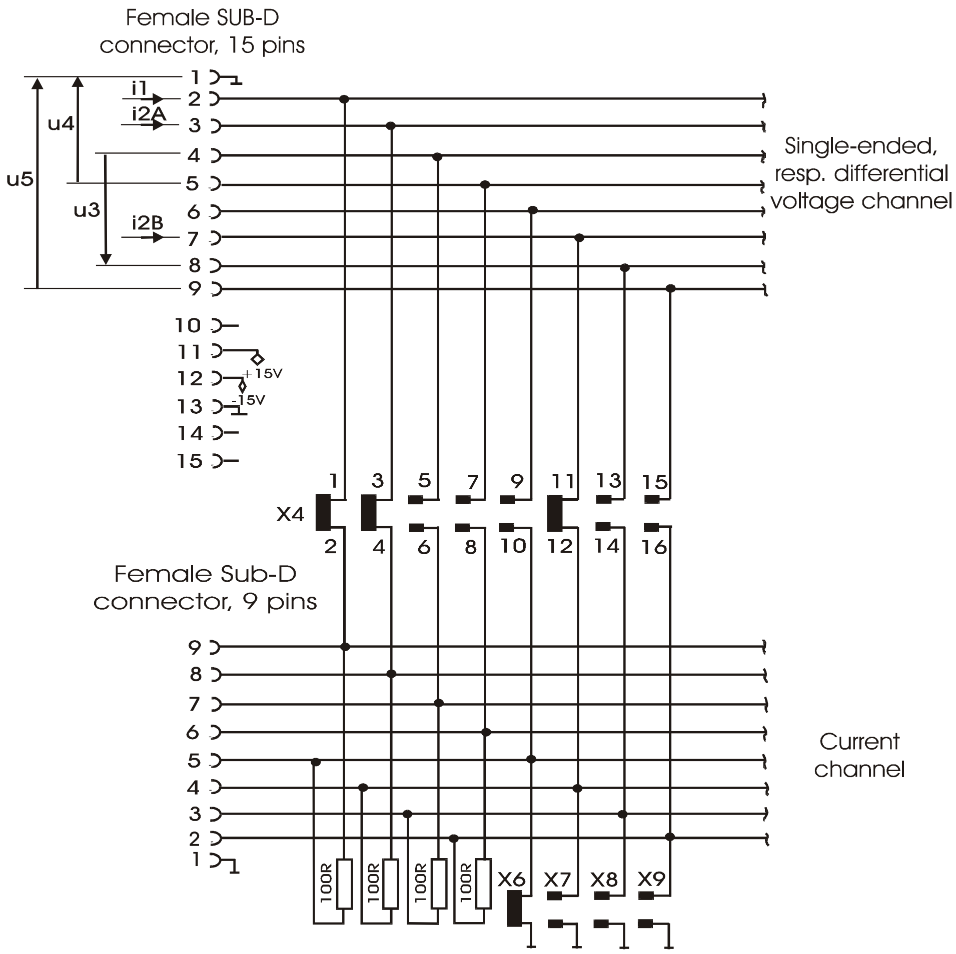


Fig. 23: Voltage and current channels of the D-AD8 submodule

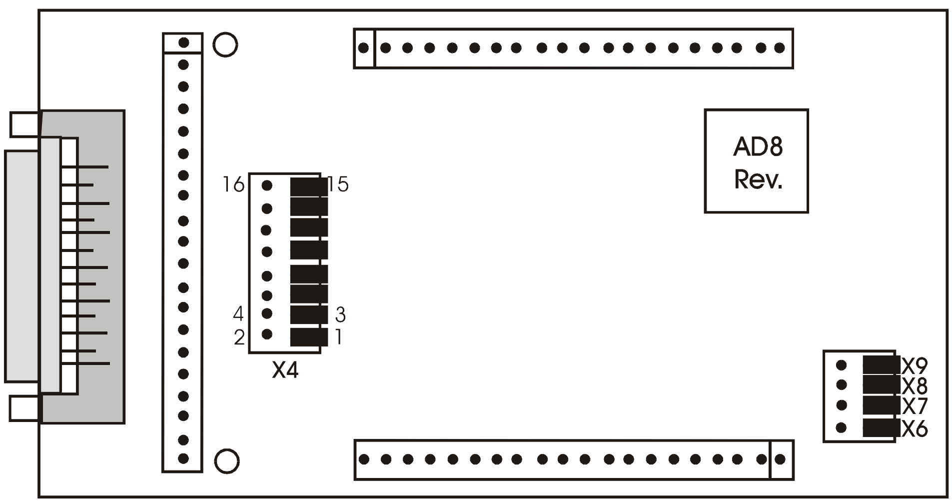


Fig. 24: Jumper settings of the D-AD8 submodule

In the given case, altogether 4 configurations of input channels are required.

Register	Register value	Input configuration
Configuration of AD channel # 1	7	Differential -20 ... +20 mA
Configuration of AD channel # 2	7	Differential -20 ... +20 mA
Configuration of AD channel # 3	12	Differential -10 ... +10V
Configuration of AD channel # 4	12	Differential -10 ... +10V

Register with converted digital value	Analogue signal
Direct actual value - ADC channel # 1	Current (i1A - i1B)
Direct actual value - ADC channel # 2	Current (i2A - i2B)
Direct actual value - ADC channel # 3	Voltage u3
Direct actual value - ADC channel # 4	Voltage u4

10 Submodule D-DA4 (analogue output)

10.1 Technical Data

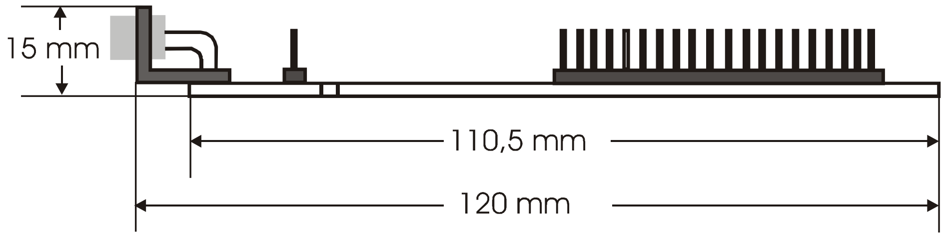


Fig. 25: Side view of the D-DA4 submodule

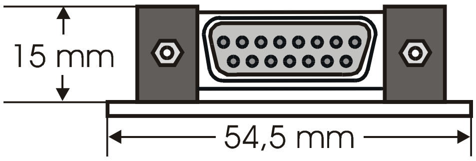


Fig. 26: Front view of the D-DA4 submodule

Design	
Dimensions (H x W x D in mm)	15.0 x 54.5 x 110.5
Mounting	This module can be plugged into the submodule port # 2of the basic module D-CON (refer to Fig. 3)

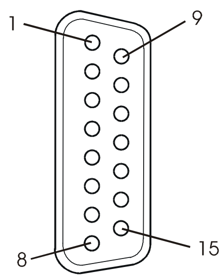
Functional Data	
Quantity of outputs:	4 single-ended channels (voltage to ground) Each channel can be used as voltage or current output
Resolution	16 Bit
Voltage range: Value range	-10 V ... +10 V -32768 ... +32767

Functional Data	
Current range Value range	0 mA ... +20 mA 0 ... +32767
Absolute error (Voltage)	0.02 %
Absolute error (Current)	0.08 %

Electrical Data	
Power supply D-DA4 +24 V	Through submodule port # 2 located on the basic module D-CON
Output impedance - Voltage - Current	 0.3 Ω 2.5 MΩ
Max. output current	20 mA
Electrical isolation	None

10.2 Pin Assignment

Pin Assignment - Female connector SUB-D, 15 pins



Pin	Signal	Comment
1	GND	Ground
2	not assigned	
3	IOUT4	Current output - Channel # 4
4	IOUT3	Current output - Channel # 3
5	IOUT2	Current output - Channel # 2
6	IOUT1	Current output - Channel # 1
7	GND	Ground
8	GND	Ground
9	GND	Ground
10	GND	Ground
11	VOUT4	Voltage output - Channel # 4
12	VOUT3	Voltage output - Channel # 3
13	VOUT2	Voltage output - Channel # 2
14	VOUT1	Voltage output - Channel # 1
15	not assigned	

11 Theory of the Digital Sampling Controller

A controller serves the following basic purposes:

- The value of the controlled variable x(k) is to follow the reference variable w(k) as accurately and instantaneously as possible (follow-up control, servo-control).
- The value of the manipulated value shall correspond with the constant value of the reference variable irrespective of any external interferences (immunity to interferences).

The mathematical model of the control loop is as follows:

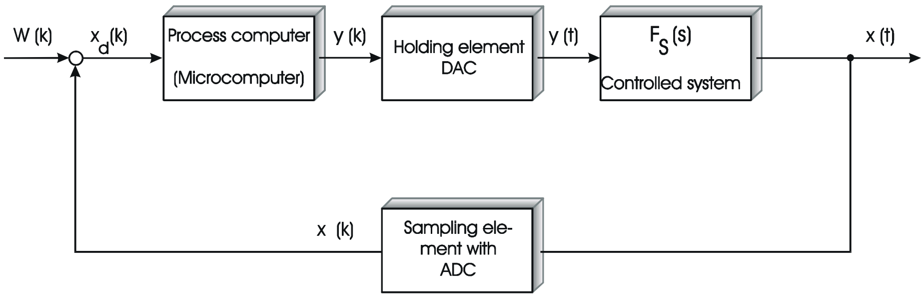


Fig. 27: Mathematical model of the process controlled by the D-PID controller:

- The time continuous signal x(t) is collected by the sampling element in fixed intervals and converted into a sequence of digitally coded numerical values x(k) by means of an ADC.
- The system deviation x_d (k) is determined by subtracting the present actual value x(k) from the setpoint value w(k).
- The processor accesses to these x_d (k) values and calculates with the help of a programmed control algorithm on the basis of these values a sequence of manipulated values y(k) which are read out in regular intervals.
- The holding element stores the latest received numerical value until it is up-dated. This element converts the numerical values with the help of a DAC into a time continuous manipulated variable y(t). Therefore, the curve of the manipulated value y(t) has a stepped characteristic.

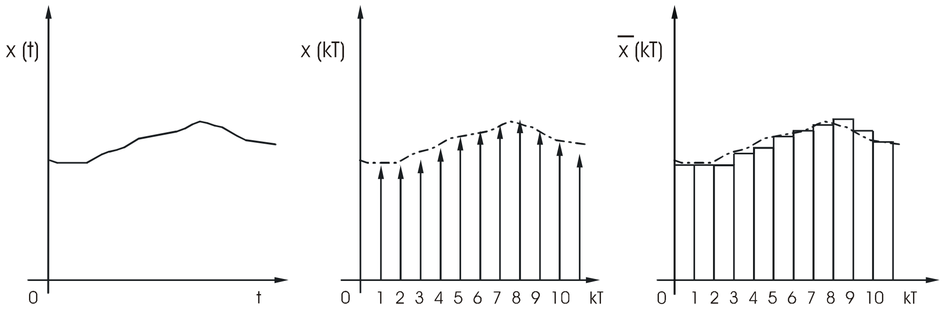


Fig. 28: Digital scanning

11.1.1 The Algorithm of PID controllers

The manipulated variable is calculated on the basis of a so called manipulating algorithm. This is a discrete PID algorithm, which is presented in the following equation:

$$y(kT) = k_p \left[x_d(kT) + \frac{1}{T_N} * \sum_{i=0}^{K} x_d(iT) * T + T_V \frac{x_d(kT) - x_d(kT - T)}{T} \right]$$

with $x_d(kT)$ = (nominal - actual) (kT)

The parameters K_p, T_n, T_v and T can be configured at will. These four parameters have the following meaning:

K_p:	Proportionality factor of the P-coefficient
T_n:	Integral-action time T_N (I component)
T_v:	Derivative-action time T_V (D component)
T:	Sampling interval (period)

The other values are:

y(kT):	Manipulated variable
$x_d(kT)$:	Present system deviation
$x_d(kT-T)$:	System deviation of the last sample

As well as there is a proportionality factor for the proportional component, there are corresponding proportionality factors for integral-action and derivative-action components, too:

$$K_T = \frac{K_p}{T_N}$$

$$K_D = T_V * K_p$$

The complete formulas for the three components of the PID algorithm are presented below. From these three equations and the manipulating algorithm result that the mathematical term is additively made up of the P, I and D components.

Special settings of the P, I and D components of the PID controller are as follows:

- If the P component by means of the factor K_p is set to zero, no control action will take place.
- Increasing T_n, the I component will become zero.
- To reset the entire derivative component to zero the factor T_v of the derivative component must be set to zero.

$$\text{P - component} = K_p * x_d$$

$$\text{I - component} = \frac{K_p * T * \sum_{i=0}^{K} x_d(iT)}{T_n}$$

$$\text{D - component} = \frac{K_p * T_v}{T} \left(x_d(kT) - x_d(kT - T)\right)$$

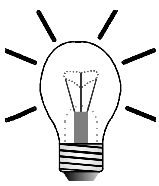
12 Firmware

12.1 Addressing the controller and the registers

Description of the register pattern: 1xyzzz

By way of example **REG 1xyzzz,** it will be demonstrated how the registers are numbered.

- The registers are addressed with the help of a 6 digit number.
- The first digit always is **1.**
- The second digit **x** specifies the **slot**, where the DELTA PID controller module is located.
  x = Slot (2 ... 8).
- The third digit **y** defines the **controller number** of the DELTA PID controller module:
  y = Controller number (1, 2, 3, 4).
- The digits four, five and six zzz specify the actual Register number, while the letters zzz correspond to the register numbers from **0 to 999.**



Note!

Before version 2.00 of the operating system for the D-CON basic module a different register pattern was used to number the registers:

xyzzz

The register numbers are upward compatible, but not downward compatible, i.e. the versions 2.00 and higher understand the old numbers (xyzzz). The versions before 2.00 do not understand the new numbers (1xyzzz).

12.1.1 Register Table

*) R/W: Read/Write; Ro: Read only

Reg. #	Type of register		R/W Ro*)
1x1000	Status register	0 ... +1023	Ro
1xy001	Command register	0 ... 38	R/W
1xy002	Setpoint value of the controller	-1000 ... +1000	R/W
1xy003	P gain	0 ... 131072	R/W
1xy004	Integral-action time T_N	1 ... 1000000	R/W
1xy005	Derivative-action time T_V	-8388608 ... +8388607	R/W
1xy006	Sampling interval	1 ... 255	R/W
1xy007	Integral-action limitation	0 ... 32767	R/W
1xy008	Slew rate limitation	1 ... 2000	R/W
1xy010	Period of the PWM signal	0 ... 65535	R/W
1xy011	Assignment Input-Controller	1 ... 8	R/W
1xy012	Assignment Output-Controller	1 ... 4	R/W
1xy017	Output value DAC, direct	-32768 ... 32767	R/W
1xy018	Present I component	-32768 ... 32767	Ro
1xy019	Manipulated value (normalised, scaled)	-1000 ... +1000	R/W
1xy020	Threshold - Activation of the controller	-32768 ... 32767	R/W
1x1023	Quantity of activable controllers	1 ... 4	R/W
1x1041	Actual value - AD input channel # 1 (normalised, scaled)	-1000 ... +1000	Ro
1x1042	Actual value - AD input channel # 2 (normalised, scaled)	-1000 ... +1000	Ro

1x1043	Actual value - AD input channel # 3 (normalised, scaled)	-1000 ... +1000	Ro
1x1044	Actual value - AD input channel # 4 (normalised, scaled)	-1000 ... +1000	Ro
1x1045	Actual value - AD input channel # 5 (normalised, scaled)	-1000 ... +1000	Ro
1x1046	Actual value - AD input channel # 6 (normalised, scaled)	-1000 ... +1000	Ro
1x1047	Actual value - AD input channel # 7 (normalised, scaled)	-1000 ... +1000	Ro
1x1048	Actual value - AD input channel # 8 (normalised, scaled)	-1000 ... +1000	Ro
1x1051	Direct actual value - ADC channel # 1	-32768 ... 32767	Ro
1x1052	Direct actual value - ADC channel # 2	-32768 ... 32767	Ro
1x1053	Direct actual value - ADC channel # 3	-32768 ... 32767	Ro
1x1054	Direct actual value - ADC channel # 4	-32768 ... 32767	Ro
1x1055	Direct actual value - ADC channel # 5	-32768 ... 32767	Ro
1x1056	Direct actual value - ADC channel # 6	-32768 ... 32767	Ro
1x1057	Direct actual value - ADC channel # 7	-32768 ... 32767	Ro
1x1058	Direct actual value - ADC channel # 8	-32768 ... 32767	Ro
1x1061	Configuration - AD channel # 1	3, 7, 8, 12, 17, 21	R/W
1x1062	Configuration - AD channel # 2	3, 7, 8, 12, 17, 21	R/W

1x1063	Configuration - AD channel # 3	3, 7, 8, 12, 17, 21	R/W
1x1064	Configuration - AD channel # 4	3, 7, 8, 12, 17, 21	R/W
1x1065	Configuration - AD channel # 5	3, 7, 8, 12, 17, 21	R/W
1x1066	Configuration - AD channel # 6	3, 7, 8, 12, 17, 21	R/W
1x1067	Configuration - AD channel # 7	3, 7, 8, 12, 17, 21	R/W
1x1068	Configuration - AD channel # 8	3, 7, 8, 12, 17, 21	R/W
1x1071	Scaling of AD input channel # 1 - lower limiting value	-1000 ... +1000	R/W
1x1072	Scaling of AD input channel # 2 - lower limiting value	-1000 ... +1000	R/W
1x1073	Scaling of AD input channel # 3 - lower limiting value	-1000 ... +1000	R/W
1x1074	Scaling of AD input channel # 4 - lower limiting value	-1000 ... +1000	R/W
1x1075	Scaling of AD input channel # 5 - lower limiting value	-1000 ... +1000	R/W
1x1076	Scaling of AD input channel # 6 - lower limiting value	-1000 ... +1000	R/W
1x1077	Scaling of AD input channel # 7 - lower limiting value	-1000 ... +1000	R/W
1x1078	Scaling of AD input channel # 8 - lower limiting value	-1000 ... +1000	R/W
1x1081	Scaling of AD input channel # 1 - upper limiting value	-1000 ... +1000	R/W
1x1082	Scaling of AD input channel # 2 - upper limiting value	-1000 ... +1000	R/W
1x1083	Scaling of AD input channel # 3 - upper limiting value	-1000 ... +1000	R/W
1x1084	Scaling of AD input channel # 4 - upper limiting value	-1000 ... +1000	R/W
1x1085	Scaling of AD input channel # 5 - upper limiting value	-1000 ... +1000	R/W

1x1086	Scaling of AD input channel # 6 - upper limiting value	-1000 ... +1000	R/W
1x1087	Scaling of AD input channel # 7 - upper limiting value	-1000 ... +1000	R/W
1x1088	Scaling of AD input channel # 8 - upper limiting value	-1000 ... +1000	R/W
1x1091	Scaling of DA output channel # 1 - lower limiting value	-1000 ... +1000	R/W
1x1092	Scaling of DA output channel # 2 - lower limiting value	-1000 ... +1000	R/W
1x1093	Scaling of DA output channel # 3 - lower limiting value	-1000 ... +1000	R/W
1x1094	Scaling of DA output channel # 4 - lower limiting value	-1000 ... +1000	R/W
1x1095	Scaling of DA output channel # 1 - upper limiting value	-1000 ... +1000	R/W
1x1096	Scaling of DA output channel # 2 - upper limiting value	-1000 ... +1000	R/W
1x1097	Scaling of DA output channel # 3 - upper limiting value	-1000 ... +1000	R/W
1x1098	Scaling of DA output channel # 4 - upper limiting value	-1000 ... +1000	R/W
1x1099	Software version	0 ... +8388607	Ro
1x1124	Enabling local access to the digital output	0 ... 65535	R/W
1x1126	Enabling global access to the digital output	0 ... 65535	R/W
1x1151	Averaging ON / OFF - AD channel # 1	0 ... 32767	R/W
1x1152	Averaging ON / OFF - AD channel # 2	0 ... 32767	R/W

1x1153	Averaging ON / OFF - AD channel # 3	0 ... 32767	R/W
1x1154	Averaging ON / OFF - AD channel # 4	0 ... 32767	R/W
1x1155	Averaging ON / OFF - AD channel # 5	0 ... 32767	R/W
1x1156	Averaging ON / OFF - AD channel # 6	0 ... 32767	R/W
1x1157	Averaging ON / OFF - AD channel # 7	0 ... 32767	R/W
1x1158	Averaging ON / OFF - AD channel # 8	0 ... 32767	R/W
1xy199**)	Recognised submodule type	1 ... 7	Ro

**) y = Submodule port (1 ... 3)

12.1.2 Register Description

For each register the following items are quoted:

1. Function of the register resulting from a „reading access“, i.e. an instruction of the following kind: **REGISTER_LOAD [220 with R(1xy1056)].**
2. Function of the register resulting from a „writing access“, i.e. an instruction of the following kind: **REGISTER_LOAD [1x1068 with R(120)].**
3. Value range, i.e. range of valid numerical register values.
4. Value of the register shortly after the PROCESS PLC was switched on (or following a RESET).
5. An example regarding the use of the register with a description of the effect resulting from the given instruction.

Register 1x1000: Status register	
Function	**Description**
Read	Reports back the state of the DELTA PID controller module
Write	Illegal
Value range	0 ... +1023 (bit-coded)
Value following a reset:	Depending on the current state

Meaning of the individual status register bits:

Bit 0:	Controller # 1 ON?	1 =	ON
		0 =	OFF
Bit 1:	Controller # 2 ON?	1 =	ON
		0 =	OFF
Bit 2:	Controller # 3 ON?	1 =	ON
		0 =	OFF
Bit 3:	Controller # 4 ON?	1 =	ON
		0 =	OFF
Bit 4:	Type of output # 1 and 2	1 =	PWM
		0 =	Analogue
Bit 5:	Type of output # 3 and 4	1 =	PWM

Meaning of the individual status register bits:

		0 = Analogue
`Bit 7:`	With input configuration 4-20 mA for controller # 1	1 = Current < 2 mA 0 = Current ≥ 2 mA
`Bit 8:`	With input configuration 4-20 mA for controller # 2	1 = Current < 2 mA 0 = Current ≥ 2 mA
`Bit 9:`	With input configuration 4-20 mA for controller # 3	1 = Current < 2 mA 0 = Current ≥ 2 mA
`Bit 10:`	With input configuration 4-20 mA for controller # 4	1 = Current < 2 mA 0 = Current ≥ 2 mA

Note!

These status bits can be queried in a simple way using the **`BIT_SET`** and **`BIT_CLEAR`** instructions.

Example:

This program section is waiting until the differential input current from controller # 1 drops below the value of 2 mA. Subsequently, an alarm message is issued.
This status bit is set only with the input configuration 4-20 mA for controller # 1.

```
....
WHEN
      BIT_SET [REG=121000, Bit=7]
    THEN
....
```

Register 1xy001: Command register	
Function	**Description**
Read	Instruction currently being executed or the last executed instruction
Write	Starts the execution of a new instruction
Value range	0 to 38
Value following a reset:	0

The DELTA PID controller module makes use of the following instructions:

1 **Controller ON:**

Controller # y is switched on.

2 **Controller OFF (Default):**

Controller # y is switched off. The manipulated variable is set to zero.

3 **CLEAR I component:**

The I component of the controller is cleared (set to zero).

4 **PWM+ ON:**

The PWM+ signal of controller # y is switched ON.

This instruction may be used only in case the controller # y is switched off.
Beforehand, local access to the digital output of controller # y must be enabled (Register 1x1124).

24 volt are applied to the digital output. For test purposes the functioning of a heater can be tested.

Do not switch on the controller in this state!

5 **PWM+ OFF (Default):**

The PWM+ signal of controller # y is switched OFF.

This instruction may be used only in case the controller # y is switched off.

The DELTA PID controller module makes use of the following instructions:

6 **PWM- ON:**

The PWM- signal of controller # y is switched ON.

This instruction may be used only in case the controller # y is switched off.
Beforehand, local access to the digital output of controller # y must be enabled (Register 1x1124).

24 volt are applied to the digital output. For test purposes the functioning of a cooling aggregate can be tested.

Do not switch on the controller in this state!

7 **PWM- OFF (Default):**

The PWM- signal of controller # y is switched OFF.

This instruction may be used only in case the controller # y is switched off.

12 **Controller OFF (keeping the manipulated variable):**

Controller # y is switched off and the last manipulated variable is kept.

35 **Activation of PWM outputs on controller # 1 and 2:**

On controller # 1 and 2 the PWM output is used.

This instruction is entered into register **1x1001**.

36 **Deactivation of PWM outputs on controller # 1 and 2 (Default):**

On controller # 1 and 2 the analogue output is used.

This instruction is entered into register **1x1001**.

37 **Activation of PWM outputs on controller # 3 and 4:**

On controller # 3 and 4 the PWM output is used.

This instruction is entered into register **1x1001**.

38 **Deactivation of PWM outputs on controller # 3 and 4 (Default):**

On controller # 3 and 4 the analogue output is used.

This instruction is entered into register **1x1001**.

Example 1:

With the following instruction controller # 2 is switched on.

```
; Transfer of instruction 1 to the DELTA PID controller module
located in slot # 3.

THEN
  REGISTER_LOAD [132001 with 1]
```

Example 2:

The manipulated variable (positive and negative value) is to be transferred to the process as PWM signal. For this purpose, switch on the PWM+ and PWM- signals of controller # 2.
The DELTA PID controller module is plugged into slot # 2.

```
....
; Activating PWM outputs on controller # 1 and 2:
    THEN
      REGISTER_LOAD [121001 with 35]
....
; Switching controller # 2 ON.
    THEN
      REGISTER_LOAD [122001 with 1]
....
```

Register 1xy002: Setpoint value of the controller	
Function	**Description**
Read	If the controller is activated the present reference variable is read out by controller # y.
Write	If the controller is activated a new reference variable is transferred to the controller # y.
Value range	-1000 ... +1000
Value following a reset:	0

Register 1xy003: P gain	
Function	**Description**
Read	Present gain of controller # y.
Write	Specification of a new gain for controller # y.
Value range	0 ... 131072
Value following a reset:	1000 (corresponds to a gain of 1)

The P gain corresponds to the proportionality factor K_p of the P controller. The P component of the PID controller results from the multiplication of the P gain by deviation x_d.

Within the controller the value stored in the register is divided by 1000, i.e., to realise a gain of 1 the content of register 1xy003 must be 1000.

Register 1xy004: Integral-action time T_N (I component)	
Function	**Description**
Read	Present integral-action time T_N of controller # y
Write	Specification of a new integral-action time for controller # y
Value range	1... 1000000 *)
Value following a reset:	1000000 (D component deactivated)

*): K_P * T_N must not exceed 131072000.

The unit of the register value is milliseconds, i.e. a register value of 1 000 specifies an integral-action time of 1 second.

Register 1xy005: Derivative-action time T_V (D component)	
Function	**Description**
Read	Present derivative-action time T_V of controller # y.
Write	Specification of a new derivative-action time T_V for controller # y
Value range	-8388608 ... +8388607 *)
Value following a reset:	0 (D component deactivated)

*): (K_P * T_V) / (number of activated controllers) must not exceed 26214400.

The proportionality factor K_D of the D component results from the derivative-action time multiplied by the gain. The derivative-action time is measured in milliseconds.

Register 1xy006: Sampling interval	
Function	**Description**
Read	Present sampling interval T of controller # y.
Write	Specification of a new sampling interval T for controller # y
Value range	1 ... 255
Value following a reset:	10

The time continuous signal x(t) is collected by the sampling element in fixed intervals and converted into a sequence of digitally coded numerical values x(k) by means of an ADC (see Fig. 29).

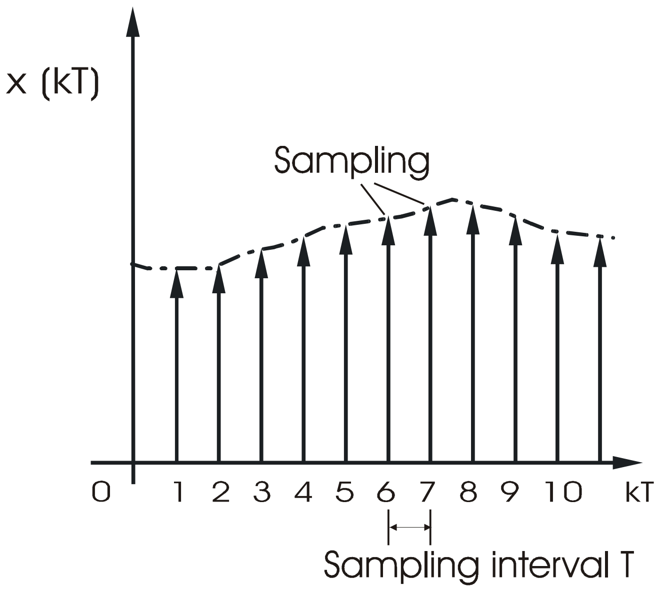


Fig. 29: Sampling of a time-continuous signal.

The sampling interval is defined in register 1xy006.

The sampling interval is calculated by the following formula:
Sampling interval T = <Reg. 1xy006> * <Reg. 1x1023> * 2 ms

The minimum sampling interval is 2 ms.

Register 1xy007: Integral-action limitation	
Function	**Description**
Read	Present limitation of the I component of controller # y *)
Write	New limitation of the I component of controller # y
Value range	0 ... 32767
Value following a reset:	1000

*): In this register a limit for the integral-action component can be specified which applies to the positive range, as well as to the negative range.

Fig. 30 shows how the limitation of the I component influences the manipulated variable.

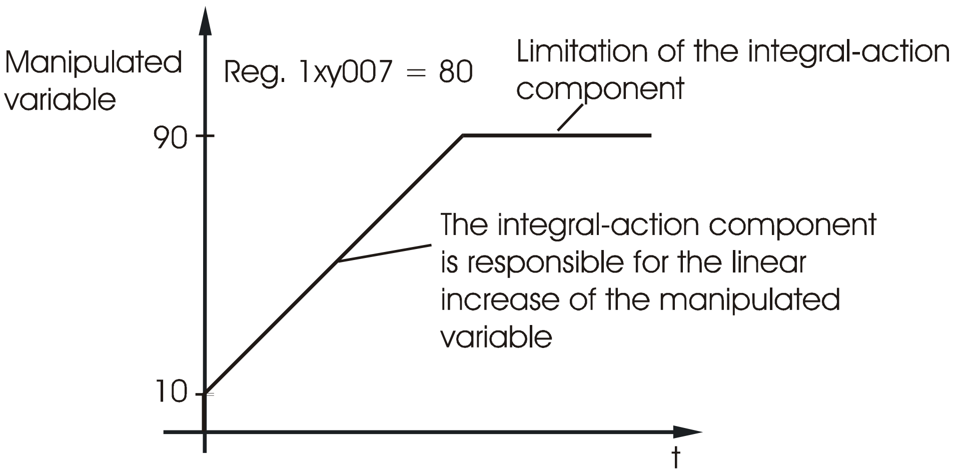


Fig. 30: Step response of a PI controller with limitation of the I component.

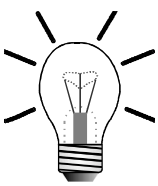
Register 1xy008: Slew rate limitation	
Function	**Description**
Read	Present slew rate limitation of controller # y.
Write	Specification of a new slew rate limitation for controller # y.
Value range	1 ... 2000
Value following a reset:	1000

The parameter „slew rate limitation“ defines the maximum rate of change of the manipulated variable per sampling cycle.

Register 1xy010: Period of the PWM signal	
Function	**Description**
Read	Present period of the PWM signal
Write	Specification of a new period.
Value range	0 ... 65535
Value following a reset:	10

The period of the PWM signal is calculated by the following formula:
Period = <Reg. 1xy010> * <Reg. 1x1023> * 2 ms

Please refer to Fig. 14.



Note!

The value of register 1xy010 must be many times over the sampling interval (value of register 1xy006).

Register 1xy011: Assignment Input-Controller	
Function	**Description**
Read	Present assignment Input-Controller
Write	Specification of a new assignment.
Value range	1 ... 8
Value following a reset:	Register 1x1011: 1 Register 1x2011: 2 Register 1x3011: 3 Register 1x4011: 4

Through this register the assignment between controller and AD channel can be changed.

Possible assignments			
Value of register 1xy011	Configuration: Single-ended Assignment of controller # y to one of the following input channels (AD channel)	Configuration: Differential Assignment of controller # y to one of the following input channels (AD channel)	Conversion of the controlled variable into a digital value and storage of same in one of the following registers for subsequent processing by controller #y
1	IN1	(IN1A - IN1B)	1x1041
2	IN2	(IN2A - IN2B)	1x1042
3	IN3	(IN3A - IN3B)	1x1043
4	IN4	(IN4A - IN4B)	1x1044
5	IN5	Illegal	1x1045
6	IN6	Illegal	1x1046
7	IN7	Illegal	1x1047
8	IN8	Illegal	1x1048

Register 1xy012: Assignment Output-Controller	
Function	**Description**
Read	Present assignment Output-Controller
Write	Specification of a new assignment.
Value range	1 ... 4
Value following a reset:	Register 1x1012: 1 Register 1x2012: 2 Register 1x3012: 3 Register 1x4012: 4

Possible assignments	
Value of register 1xy012	Assignment of controller # y to output channel (DA channel)
1	DAC 1
2	DAC 2
3	DAC 3
4	DAC 4

Through this register the assignment between controller and DA channel can be changed.

Unlike this procedure, the relation between controller and PWM output is fix and is not subject to changes (refer to chapter „Local access by the DELTA PID controller module“).

Default values:
The manipulated variable is transmitted as analogue value to the process control loop as follows:
- Manipulated variable of controller # 1 on DA channel # 1;
- Manipulated variable of controller # 2 on DA channel # 2;
- Manipulated variable of controller # 3 on DA channel # 3;
- Manipulated variable of controller # 4 on DA channel # 4.

Important!

For each of the four controllers this register must be written in such a way that access of two or more controllers to the same DA channel is excluded.

Register 1xy017: Output value DAC, direct	
Function	**Description**
Read	-
Write	Output value DAC channel # y (digital)
Value range	-32768 ... 32767
Value following a reset:	0

Important!

Register 1xy017 will perform a function only in one of the following cases:

- Controller # y is switched OFF, and not activated;
- Controller # y is switched ON, and its threshold is not exceeded. The threshold is defined in register 1xy020.

Register 1xy017 is required so as to convert a digital value by means of the DAC of channel # y to an analogue value.

For this purpose, a value is written into register 1xy017. This value is in the range between -32768 and +32767.
The voltage range between -10 V and +10 V, for example, is divided into discrete voltage values with a resolution of 16 bit (65536). One digit, i.e. the least voltage difference, is approx. 0.3 mV.

Example 1:

The DELTA PID controller module is plugged into slot # 2. A voltage of approx. +5 V is to be output on DA channel # 2.

```
THEN
    REGISTER_LOAD [122017 with 16384]
```

Example 2:

The DELTA PID controller module is plugged into slot #4. A current of 4 mA is to be output on DA channel # 3.

```
THEN
    REGISTER_LOAD [143017 with 6554]
```

Register 1xy018: Present I component	
Function	**Description**
Read	Present I component of controller # y.
Write	Illegal
Value range	-32768 ... 32767
Value following a reset:	0

The present integral-action component can be read out out of register 1xy018.

The integral-action component is calculated by the following formula:

$$I - component = \frac{kp}{T_N} \bullet \sum_{I=0}^{k} x_d(IT) \bullet dT$$

with $x_d(kT)$ = (nominal -actual) (kT)

Register 1xy019: Manipulated value (normalised, scaled)	
Function	**Description**
Read	Present manipulated variable of controller # y.
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

The manipulated variable is computed by controller # y on the basis of the control algorithm: Of course, controller # y must be switched on.

The default setting for scaled output values is -1000 (lower limit) and +1000 (upper limit). These limits are defined in the registers 1x1091 through 1x1098.

Depending on the kind of output and its configuration the manipulating variable ranging from -1000 to +1000 may correspond with the following electrical signals:

- a voltage range between -10 V and +10 V;
- a current range between 0 and 20 mA;

- a PWM-signal with a pulse width between 0 and 100 %.

Since the submodule D-DA4 can generate only positive currents the manipulating range is restricted with regard to the current output. Setting the lower limit for output scaling to zero the restriction of the manipulated range can be undone (refer to description of register 1x1091).

Register 1xy020: Threshold- Activation of the controller	
Function	**Description**
Read	Present threshold of controller # y.
Write	Specification of a new threshold.
Value range	-32768 ... 32767
Value following a reset:	-32768

First of all, switch on controller # y with command 1.
If the actual value (ADC, direct) of controller # y exceeds the threshold the corresponding controller starts to perform its control action.
The default value -32768 ensures that the controller always is switched on. Except that the operator makes use of the possibility to specify a threshold. Register 1xy017 (DAC, direct) defines the analogue output value before the threshold is reached.

Register 1x1023: Quantity of activable controllers	
Function	**Description**
Read	Present quantity of activable controllers
Write	New quantity of activable controllers
Value range	1 ... 4
Value following a reset:	4

By means of register 1x1023 the number of activated controllers is specified.
The digit in the register defines the last activated controller. The counting direction starts from controller 1.
Therefore, it is not possible to operate, for instance, only the controllers 3 and 4, because controller 1 is always active and cannot be deactivated.

Unlike controller activation by means of controller specific registers, global activation has an effect on total cycle time of all controllers. If controllers are deactivated by means of controller-specific command registers the total cycle time of all four controllers remains the same.

Example:

Each controller has a cycle time of 2 ms. Thus, 4 controllers have a total cycle time of 8 ms. If 3 of the 4 controllers are switched off by means of the corresponding command registers the total cycle time remains 8 ms.
This means: The only remaining active controller has an effective cycle time of 8 ms although the individual cycle time is 2 ms.
Using the global deactivation of controllers the total cycle time is reduced accordingly. If, for example, only controller # 1 is activated (register 1x1023 = 1) the controller-specific cycle time and the total cycle time are 2 ms.

During the cycle time the PID controller completes the following jobs:
- Acquiring the actual controlled variable.
- Comparing the controlled variable with the setpoint value (reference variable).
- Computing the manipulated variable on the basis of the resulting system deviation using the discrete PID algorithm.
- Transferring the manipulated variable to the actuator.

Register 1x1041: Actual value - AD input channel # 1 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN1 or (IN1A - IN1B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

The analogue single-ended signal IN1 or differential-mode signal (IN1A - IN1B), respectively, is converted into a digital value.
The normalised and scaled value is stored in register 1x1041 so as to be processed by the PID controller. For the controller it is signal x(k).
AD conversion is continuously being carried out in the background.

The default limits for scaled input values are -1000 (lower limit) and +1000 (upper limit). For the AD input of channel # 1 the lower limit is specified in register 1x1071 and the upper limit in register 1x1081.

Depending on the kind of input and its configuration the actual value ranging from -1000 to +1000 may correspond with the following electrical signals:

- a voltage range between -10 V and +10 V;
- a current range between -20 mA and +20 mA.

Register 1x1042: Actual value - AD input channel # 2 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN2 or (IN2A - IN2B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1043: Actual value - AD input channel # 3 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN3 or (IN3A - IN3B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1044: Actual value - AD input channel # 4 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN4 or (IN4A - IN4B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1045: Actual value - AD input channel # 5 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN5 or (IN1A - IN1B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1046: Actual value - AD input channel # 6 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN6 or (IN2A - IN2B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1047: Actual value - AD input channel # 7 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN7 or (IN3A - IN3B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1048: Actual value - AD input channel # 8 (normalised, scaled)	
Function	**Description**
Read	Normalised actual value following AD conversion Signal IN8 or (IN4A - IN4B)
Write	Illegal
Value range	-1000 ... 1000
Value following a reset:	0

Register 1x1051: Direct actual value - ADC channel # 1	
Function	**Description**
Read	Actual value following AD conversion Signal IN1 or (IN1A - IN1B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

The analogue single-ended signal IN1 or differential-mode signal (IN1A - IN1B), respectively, is converted into a digital value.
The actual value resulting from AD conversion can be read out out of register 1x1951 for subsequent processing.
The AD conversion is continuously carried out in the background regardless whether the actual value is read out.

The measured voltage ranging between -10 V and +10 V is converted into a digital value with a resolution of 16 Bit (65536).
The value range is between -32768 and +32767.
One digit, i.e. the least voltage difference subject to conversion, is approx. 0.3 mV.

Example:

```
; Querying and processing the actual value of channel # 1
  following the AD conversion.
; D-AD8 is plugged into submodule port # 1.
; D-CON (DELTA PID controller module) is placed in module slot #
  2 of the DELTA basic-4 housing.
    THEN
      LOAD_REGISTER [rADValue with R(121051)]
```

Register 1x1052: Direct actual value - ADC channel # 2	
Function	**Description**
Read	Actual value following AD conversion Signal IN2 or (IN2A - IN2B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1053: Direct actual value - ADC channel # 3	
Function	**Description**
Read	Actual value following AD conversion Signal IN3 or (IN3A - IN3B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1054: Direct actual value - ADC channel # 4	
Function	**Description**
Read	Actual value following AD conversion Signal IN4 or (IN4A - IN4B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1055: Direct actual value - ADC channel # 5	
Function	**Description**
Read	Actual value following AD conversion Signal IN5 or (IN1A - IN1B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1056: Direct actual value - ADC channel # 6	
Function	**Description**
Read	Actual value following AD conversion Signal IN6 or (IN2A - IN2B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1057: Direct actual value - ADC channel # 7	
Function	**Description**
Read	Actual value following AD conversion Signal IN7 or (IN3A - IN3B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1058: Direct actual value - ADC channel # 8	
Function	**Description**
Read	Actual value following AD conversion Signal IN8 or (IN4A - IN4B)
Write	Illegal
Value range	-32768 ... +32767
Value following a reset:	0

Register 1x1061: Configuration of AD channel # 1	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Comment

Through a configuration (input configuration) the following parameters can be specified:

Input configuration		
		Register value
single-ended	-20 .. +20 mA	3
	4 .. 20 mA	17
	-10 .. +10 V	8
Differential	-20 .. +20 mA	7
	4 .. 20 mA	21
	-10 .. +10 V	12

Register 1x1062: Configuration of AD channel # 2	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1063: Configuration of AD channel # 3	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1064: Configuration of AD channel # 4	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1065: Configuration of AD channel # 5	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1066: Configuration of AD channel # 6	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1067: Configuration of AD channel # 7	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1068: Configuration of AD channel # 8	
Function	**Description**
Read	Present configuration
Write	New configuration values
Value range	3, 7, 8, 12, 17, 21
Value following a reset:	8

Register 1x1151: Averaging ON / OFF - AD channel # 1	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Comment (the explanation refers to AD channel # 1)

In this register the number of analogue values to be averaged is specified.
The average value is then entered into register 1x1051.

Example 1: Averaging over a range of 255 values.

```
THEN
  REGISTER_LOAD [121151 with 255]
```

Example 2: Averaging OFF - AD channel # 1

```
THEN
  REGISTER_LOAD [121151 with 0]
```

Register 1x1152: Averaging ON / OFF - AD channel # 2	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1153: Averaging ON / OFF - AD channel # 3	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1154: Averaging ON / OFF - AD channel # 4	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1155: Averaging ON / OFF - AD channel # 5	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1156: Averaging ON / OFF - AD channel # 6	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1157: Averaging ON / OFF - AD channel # 7	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1158: Averaging ON / OFF - AD channel # 8	
Function	**Description**
Read	Present number of input values to be averaged.
Write	New number of input values to be averaged.
Value range	0 ... 32767
Value following a reset:	0 (averaging deactivated)

Register 1x1071: Scaling of AD input channel # 1 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 1
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1081: Scaling of AD input channel # 1 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 1
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Fig. 31 and Fig. 32 are supposed to show the influence of the lower and upper limiting values on the normalised actual value of the controlled variable in register 1x1041.

Fig. 31 shows an example with default settings of the lower and upper limiting value.

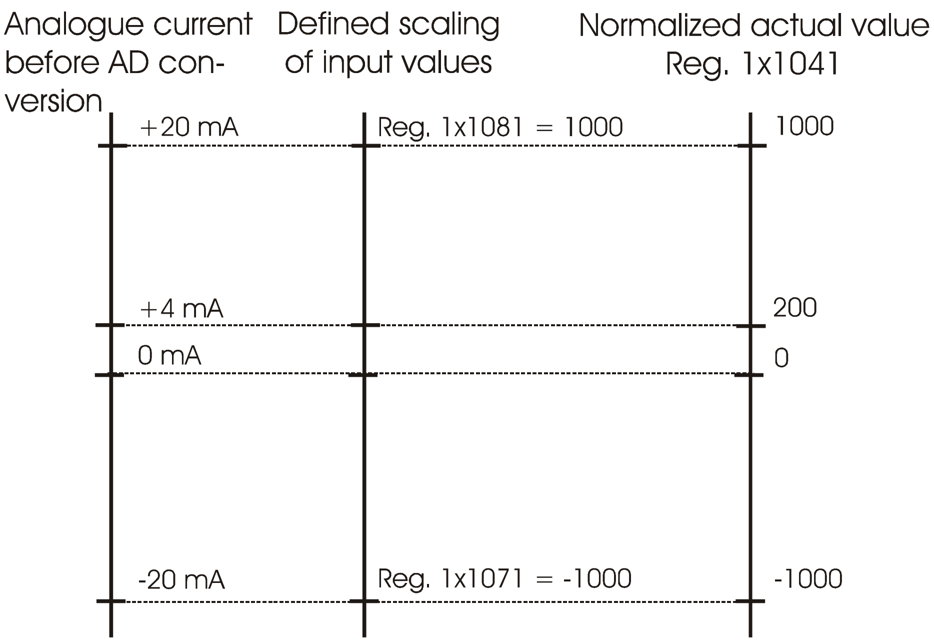


Fig. 31: Effect of the scaling of input signals - Example 1.

Fig. 32 shows an example where the lower limit is 0 and the upper limit is 1000.

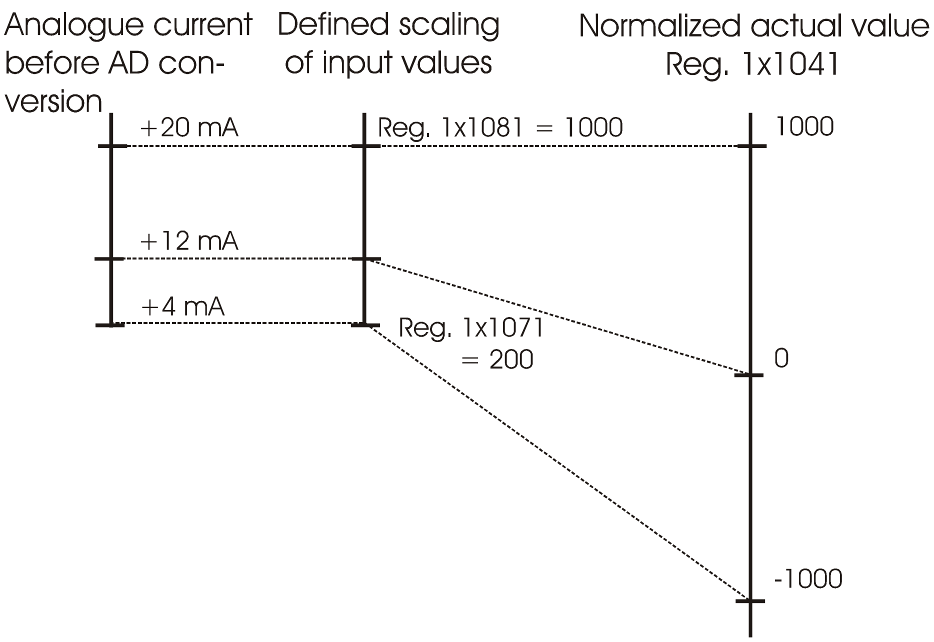


Fig. 32: Effect of the scaling of input signals - Example 2.

Register 1x1072: Scaling of AD input channel # 2 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 2
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1082: Scaling of AD input channel # 2 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 2
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1073: Scaling of AD input channel # 3 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 3
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1083: Scaling of AD input channel # 3 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 3
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1074: Scaling of AD input channel # 4 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 4
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1084: Scaling of AD input channel # 4 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 4
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1075: Scaling of AD input channel # 5 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 5
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1085: Scaling of AD input channel # 5 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 5
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1076: Scaling of AD input channel # 6 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 6
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1086: Scaling of AD input channel # 6 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 6
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1077: Scaling of AD input channel # 7 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 7
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1087: Scaling of AD input channel # 7 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 7
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1078: Scaling of AD input channel # 8 - lower limiting value	
Function	**Description**
Read	Present lower limit - AD channel # 8
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1088: Scaling of AD input channel # 8 - upper limiting value	
Function	**Description**
Read	Present upper limit - AD channel # 8
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1091: Scaling of DA output channel # 1 - lower limiting value	
Function	**Description**
Read	Present lower limit - Scaling of DA output channel # 1
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1095: Scaling of DA output channel # 1 - upper limiting value	
Function	**Description**
Read	Present upper limit - Scaling of DA output channel # 1
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Fig. 33 and Fig. 34 are supposed to show the influence of the lower and upper limiting values on the normalised actual value of the controlled variable in register 1x1019 and, thus, on the analogue output value.

Fig. 33 shows an example with default settings of the lower and upper limiting value.

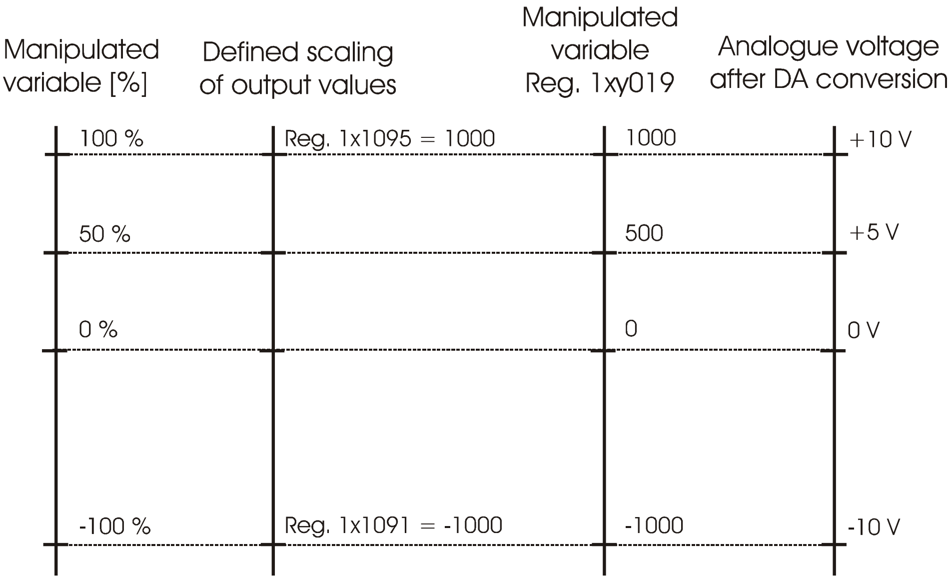


Fig. 33: Effect of the scaling of output signals - Example 1.

Fig. 34 shows an example where the lower limit is 0 and the upper limit is 1000.

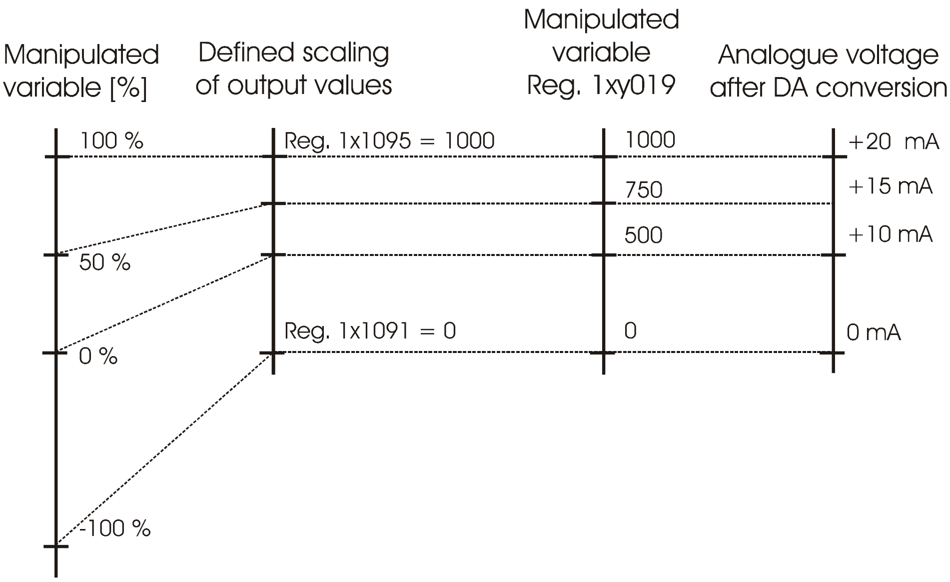


Fig. 34: Effect of the scaling of output signals - Example 2.

Register 1x1092: Scaling of DA output channel # 2 - lower limiting value	
Function	**Description**
Read	Present lower limit - Scaling of DA output channel # 2
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1096: Scaling of DA output channel # 2 - upper limiting value	
Function	**Description**
Read	Present upper limit - Scaling of DA output channel # 2
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1093: Scaling of DA output channel # 3 - lower limiting value	
Function	**Description**
Read	Present lower limit - Scaling of DA output channel # 3
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1097: Scaling of DA output channel # 3 - upper limiting value	
Function	**Description**
Read	Present upper limit - Scaling of DA output channel # 1
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1094: Scaling of DA output channel # 4 - lower limiting value	
Function	**Description**
Read	Present lower limit - Scaling of DA output channel # 4
Write	New lower limit
Value range	-1000 ... +1000
Value following a reset:	-1000

Register 1x1098: Scaling of DA output channel # 4 - upper limiting value	
Function	**Description**
Read	Present upper limit - Scaling of DA output channel # 4
Write	New upper limit
Value range	-1000 ... +1000
Value following a reset:	+1000

Register 1x1124: Enabling local access to the digital outputs	
Function	**Description**
Read	Present status of access to outputs
Write	Enabling or disabling local access to the outputs
Value range	Bit 0 ... Bit15 (bit-coded)
Value following a reset:	0

Comment:

Bit 0:	Enabling local access to output 1	1 =	PWM+ signal from controller # 1 is output via digital output 1
		0 =	Output disabled.
Bit 1:	Enabling local access to output 2	1 =	PWM- signal from controller # 1 is output via digital output 2
		0 =	Output disabled.
Bit 2:	Enabling local access to output 3	1 =	PWM+ signal from controller # 2 is output via digital output 3
		0 =	Output disabled.
Bit 3:	Enabling local access to output 4	1 =	PWM- signal from controller # 2 is output via digital output 4
		0 =	Output disabled.
Bit 4:	Enabling local access to output 5	1 =	PWM+ signal from controller # 3 is output via digital output 5
		0 =	Output disabled.
Bit 5:	Enabling local access to output 6	1 =	PWM- signal from controller # 3 is output via digital output 6
		0 =	Output disabled.

Comment:

Bit 6:	Enabling local access to output 7	1 = PWM+ signal from controller # 4 is output via digital output 7 0 = Output disabled.
Bit 7	Enabling local access to output 8	1 = PWM- signal from controller # 4 is output via digital output 8 0 = Output disabled.
Bit 8	Enabling local access to output 9	1 = No function. 0 = Output disabled.
Bit 9	Enabling local access to output 10	1 = No function. 0 = Output disabled.
Bit 10	Enabling local access to output 11	1 = No function. 0 = Output disabled.
Bit 11	Enabling local access to output 12	1 = No function. 0 = Output disabled.
Bit 12	Enabling local access to output 13	1 = No function. 0 = Output disabled.
Bit 13	Enabling local access to output 14	1 = No function. 0 = Output disabled.
Bit 14	Enabling local access to output 15	1 = No function. 0 = Output disabled.
Bit 15	Enabling local access to output 16	1 = No function. 0 = Output disabled.

Register 1x1126: Enabling global access to the digital outputs	
Function	**Description**
Read	Present status of access to outputs
Write	Enabling or disabling global access to the outputs
Value range	Bit 0 ... Bit15 (bit-coded)
Value following a reset:	0

Comment

*): It is possible to directly set or reset the digital output via SYMPAS instruction „Output number".

Bit 0:	Enabling global access to output 1	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 1:	Enabling global access to output 2	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 2:	Enabling global access to output 3	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 3:	Enabling global access to output 4	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 4:	Enabling global access to output 5	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 5:	Enabling global access to output 6	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 6:	Enabling global access to output 7	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 7	Enabling global access to output 8	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit 8	Enabling global access to	1 = Access via D-CPU is possible*).

Comment

*): It is possible to directly set or reset the digital output via SYMPAS instruction „Output number“.

		output 9	0 = Output disabled.
Bit	**9**	Enabling global access to output 10	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**10**	Enabling global access to output 11	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**11**	Enabling global access to output 12	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**12**	Enabling global access to output 13	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**13**	Enabling global access to output 14	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**14**	Enabling global access to output 15	1 = Access via D-CPU is possible*). 0 = Output disabled.
Bit	**15**	Enabling global access to output 16	1 = Access via D-CPU is possible*). 0 = Output disabled.

Register 1xy199: Recognised submodule type	
Function	**Description**
Read	Type of the inserted submodule
Write	Illegal
Value range	1 ... 7
Value following a reset:	Type of the inserted submodule

Register value	Submodule type
1	SV_MODULE_TYPE
2	AD8_MODULE_TYPE
3	DIMA3_MODULE_TYPE
4	SM_MODULE_TYPE
5	DA4_MODULE_TYPE
7	INTELLIGENT_MODULE_TYPE

Important!

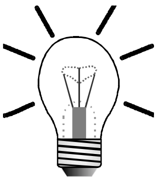
The third digit in the register number 1xy199 specifies the **`submodule port`**, but not the controller number.
y = Submodule port # (1, 2, 3).

Register 1x1099: Software version	
Function	**Description**
Read	Software version
Write	Illegal
Value range	0 ... +8388607
Value following a reset:	Present version * 1000

Comment:

In register `1x1099` the operating system's version number of the DELTA PID controller module (Software) is stored and can be read out.

Example:
Version 2.050 of the operating system is loaded.
<Reg. 1x1099> = 2050



Note!

When submitting technical support queries the version number must be quoted.

13 Installation Guide

13.1 Deinstallation of the DELTA PID controller module

Procedure:

- Switch off the supply voltage for the DELTA PROCESS PLC.
- Disconnect the 2 pin terminal (1) of the power supply for the basic module (refer to Fig. 35).
- Disconnect the two 8 pin terminals (2) of the 16 digital inputs, located on the DELTA PID controller module (refer to Fig. 35)
- If applicable:
  Disconnect the 2 pin terminal (3) of the external power supply for the outputs,
  and
  the two 8 pin terminals (3) of the 16 digital outputs (refer to Fig. 35).

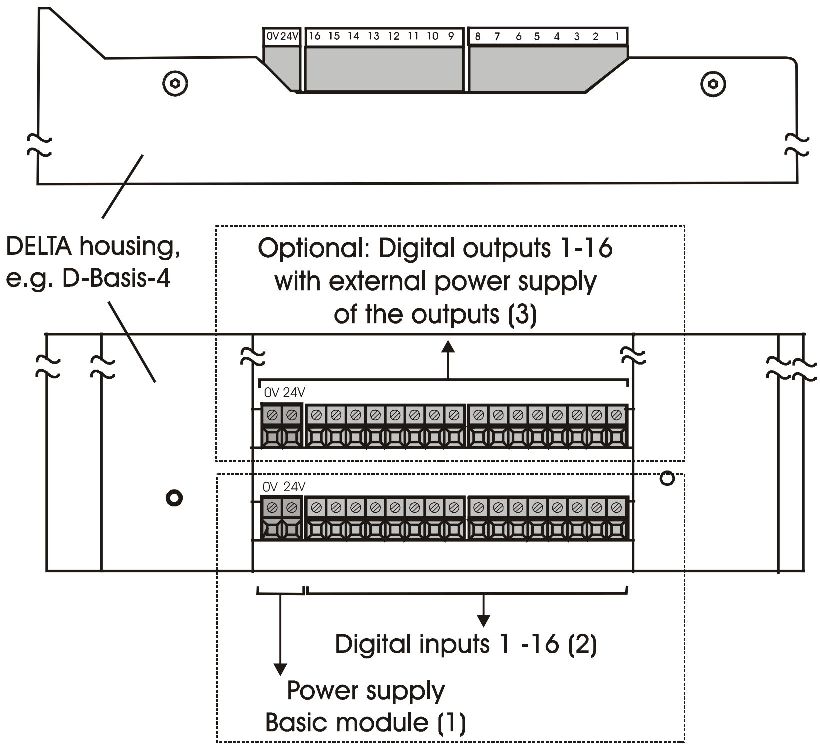


Fig. 35: Top and side view of the DELTA basic housing

- Remove all Sub-D connectors plugged into the DELTA PID controller module. These connectors are located on the front side of the controller module.
- Loosen the four screws (4) connecting the DELTA PID controller module with the basic housing of the DELTA control system using a screwdriver (refer to Fig. 36)
- Pull the DELTA PID controller module out of the DELTA basic housing using the handles (5) (refer to Fig. 36).

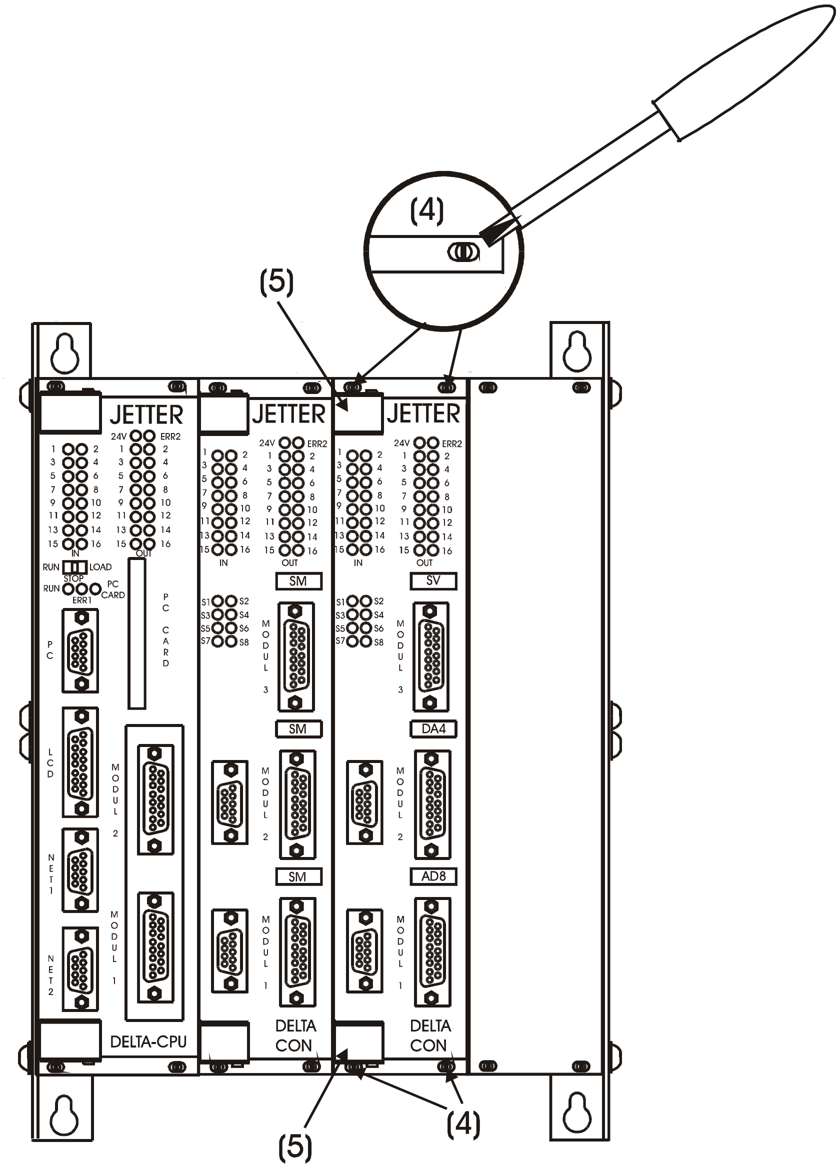


Fig. 36: Front view of the DELTA basic housing equipped with basic modules

13.2 Installation of the DELTA PID controller module

Procedure:

Installation of the D-PID 1 module is the reverse of its deinstallation described above.

Important!

Be sure to plug the green Phoenix brand screw plug connectors, COMBICON, into the correct receptacles.
Connectors plugged into the wrong socket may result in malfunctions and even may destroy the control system.

There is a danger that connectors, by mistake, are plugged into the wrong sockets especially if they are located next to each other.
A package of plastic keys is provided by Jetter AG as standard with each DELTA control system. When properly installed, these keys can guard against plugging of connectors into a wrong socket.

An example for using the keys is given in chapter „Keying“.

Important!

Be sure to supply the DELTA PID controller module with voltage after the installation of same in the DELTA basic housing.
If you fail to do so the control system will not be ready for operation.

13.3 Keying of Connectors

A package of plastic keying bands is provided with each DELTA control system. These keys are intended to guard against incorrect seating of Phoenix connectors.

Important!

Plugging connectors into the wrong socket (Phoenix name: socket = header) may result in malfunctions and even may destroy the control system.

In the following, a description of the plastic keys and their proper usage is given.

Appearance

The plastic key consists of two parts. In Fig. 37 a description of the appearance of both parts is given.

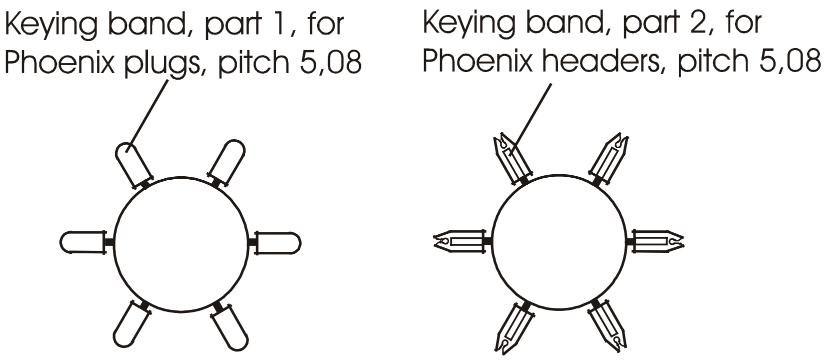


Fig. 37: Wheel of plastic keys provided with the control system.

Keying of a Phoenix connector

Fig. 38 gives an example of the keying procedure.

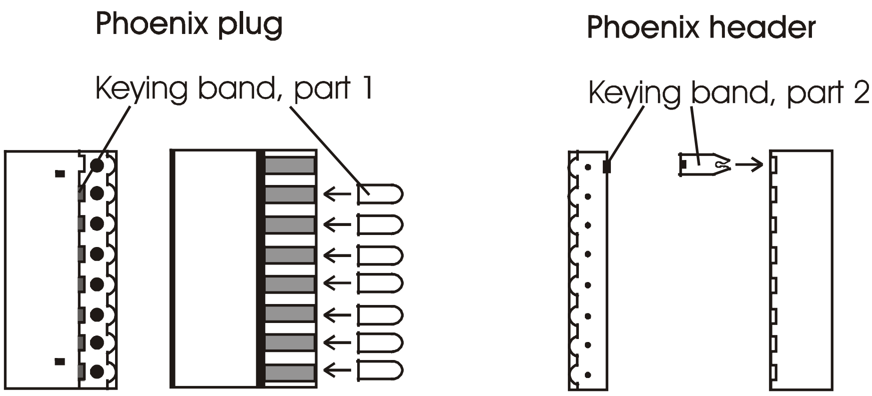


Fig. 38: Application of plastic keying bands

Suggestion concerning keying

In Fig. 39 a suggestion is given, how keying of Phoenix connectors can be carried out.

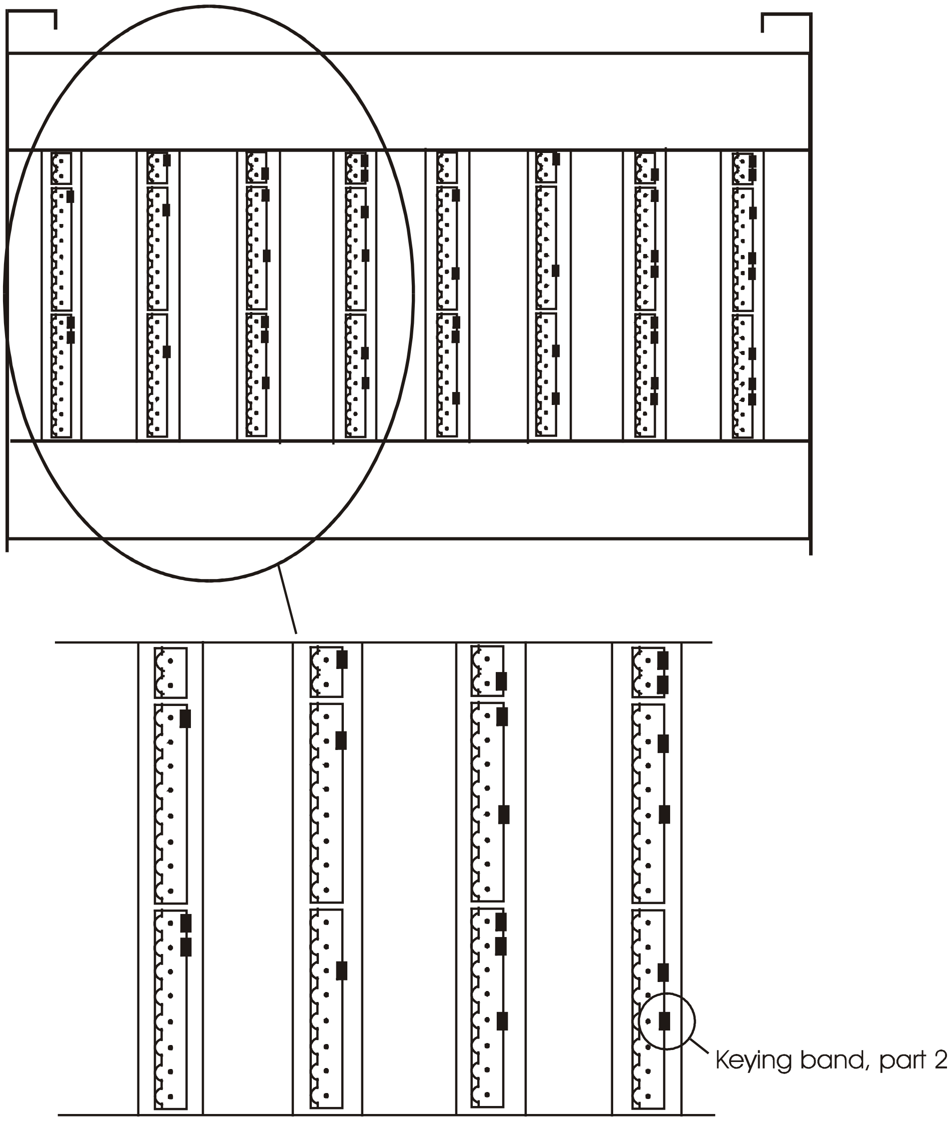


Fig. 39: Top view of the DELTA Basic-4 housing with keyed connectors.

14 Commissioning

14.1 Configuration of Input Channels

Procedure:

- Assign the controllers to the input channels (refer to description of register 1xy011 in chapter 12).
  It is advisable to keep the values contained in the registers following a reset.

  In this case, the following assignment applies:
  Controller 1: Channel # 1
  Controller 2: Channel # 2
  Controller 3: Channel # 3
  Controller 4: Channel # 4

- Connect the sensors, in most cases temperature sensors, to the analogue inputs of the DELTA PID controller module.

  Dependent on the design and the interface of the sensor, either a current or a voltage channel can be used.

  Sensors equipped with a current interface are connected to the female Sub-D connector, 9 pins, located on a level with the submodule port # 1 (refer to analogue current input shown in Fig. 3).
  Sensors equipped with a voltage interface are connected to the female Sub-D connector, 15 pins, located on a level with the submodule port # 1 (refer to analogue voltage input shown in Fig. 3).

  A description of the Sub-D connector assignment for the current and voltage inputs is given in chapter 9.

- Carry out input configuration (refer to description of registers 1x1061 through 1y1068 in chapter 12).

- In case of need, switch averaging on (refer to description of registers 1x1151 through 1y1158 in chapter 12).
  Averaging produces the same effect as a low pass.

- The converted actual value can be read out directly (refer to description of registers 1x1051 through 1y1058 in chapter 12).

The signal to be measured is converted into a digital value with a resolution of 16 Bit. The conversion is carried out continuously.

➢ The converted, normalised and scaled actual value of channels 1 through 8 is stored in the registers 1x1041 through 1048 (refer to chapter 12).

On the basis of the setpoint and the normalised actual value each of the four PID controller computes the system deviation.

➢ As far as the topic „Scaling of input signals“ is concerned, refer to the description of the registers 1x1071 and 1x1081 given in chapter 12.
As a rule, following a reset the value stored in the register can be adopted.

Section of the SYMPAS program which is responsible for this configuration:

This configuration routine is required once at the beginning of the program.

- The DELTA PID controller module is placed in module slot # 2 of the DELTA basic-4 housing.
- PID controller # 1 is to be configured for a single-ended input with a voltage ranging between -10 and +10 V.

Program section: **`REGISTER_LOAD [121061 with 8]`**

14.2 Configuration of Output Channels

14.2.1 Output of analogue manipulated variables

Procedure:

➢ Assign the controllers to the output channels (refer to description of register 1xy012 in chapter 12).
It is advisable to maintain the register value following a reset.

In this case, the following assignment applies:
Controller 1: Channel # 1
Controller 2: Channel # 2
Controller 3: Channel # 3
Controller 4: Channel # 4

➢ Connect the actuators to the analogue outputs of the DELTA PID controller module.
The analogue outputs are located on a level with the submodule port # 2 (refer to analogue current/voltage output shown in Fig. 3).

➢ Configure the PID controller in a way that the manipulated variable is output as analogue signal.

Following the instruction 36 controller # 1 and 2 will output an analogue signal as manipulated variable.
Following the instruction 38 controller # 3 and 4 will output an analogue signal as manipulated variable.
These instructions are entered into command register **1x1001**.

When the controller is switched ON:
The content of register 1xy019 is converted into an analogue voltage, respectively current.

When the controller is switched OFF:
The content of register 1xy017 is converted into an analogue voltage, respectively current.

Section of the SYMPAS program which is responsible for this configuration:

When the control system is switched on, the settings ensure that the manipulated variable is output as analogue value.
Yet the PID controller has to be switched on.

Controller # 1 directs the manipulated variable to analogue channel # 1.

14.2.2 Output of the manipulated variable as PWM signal

Procedure:

The PWM signal is output via the digital outputs of the S-O16 submodule. Each output is dedicated to a controller (refer to chapter 8.4.2).

- Enable local access to the digital outputs (refer to description of register 1x1124 in chapter 12).
  From this moment on, the PID controller has the possibility to output the manipulated variable as PWM signal

- Specify the period of the PWM signal (refer to description of register 1xy010 in chapter 12).

- Configure the PID controller in a way that the manipulated variable is output as PWM signal.

  Following the instruction 35 controller # 1 and 2 are assigned to one PWM output each.
  Following the instruction 37 controller # 3 and 4 are assigned to one PWM output each.
  These instructions are entered into command register **`1x1001`**.

  When the controller is switched ON:
  The content of register 1xy019 is converted into a PWM signal with the corresponding pulse width.

If desired:

- Set the PWM+ signal using instruction 4.
  Set the PWM- signal using instruction 6.

Reset the PWM+ signal using instruction 5.
Reset the PWM- signal using instruction 7.

For test purposes, the functioning of the heating or the cooling aggregate can be tested.

Section of the SYMPAS program which is responsible for this configuration:

This configuration is required subsequently to the configuration of the inputs.

- The DELTA PID controller module is placed in module slot # 2 of the DELTA basic-4 housing.
- PID controller # 1 is to be able to output the manipulated variable as PWM signal. For this purpose, local access to the corresponding digital outputs must be enabled.
- Furthermore, the period must be specified.
- Only one of the four PID controllers should be activated.

Program section:

- Enabling local access to digital outputs # 1 and 2.
  REGISTER_LOAD [121124 with 3]
- Activation of only one out of the four PID controllers.
  REGISTER_LOAD [121123 with 1]
- Specifying the period of the PWM signal.

  Here must be mentioned that using the PWM signal a heating or cooling aggregate is controlled. From the electrical point of view, both devices are sluggish in their response, i.e. they require plenty of power.
  The higher the power must be, the longer the period must be specified. In the given case, a period of 4 seconds should be selected.
  REGISTER_LOAD [121010 with 2000]
- Configuration of the PID controller enabling the manipulated variable to be output as PWM signal. For this purpose, enter instruction 35 into the command register.
  REGISTER_LOAD [121001 with 35]

Scaling of output signals

As far as the topic „Scaling of output signals“ is concerned, refer to the description of the registers 1x1091 and 1x1095 given in chapter 12.
As a rule, following a reset the value stored in the register can be adopted.

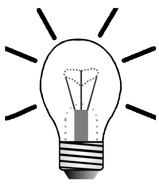
14.3 Controller Configuration

The following controller parameters have to be specified:

- Setting the P-gain (refer to description of register 1xy003 given in chapter 12).
- Setting the integral-action time T_N (refer to description of register 1xy004 given in chapter 12).
- Setting the derivative-action time T_V (D component) (refer to description of register 1xy005 given in chapter 12).
- Setting the sampling interval T (refer to description of register 1xy006 given in chapter 12).
- Specifying the number of controllers which can be activated by the user via the controller-specific command register (refer to description of register 1x1023 given in chapter 12).
- Switch on the controller.

  Controller # y is switched on by entering command 1 into command register 1xy001.
- If required, clear the integral-action component.

  The integral-action component is cleared by entering command 3 into command register 1xy001.
- Specify a setpoint (refer to description of register 1xy002 given in chapter 12).



Note!

For adjusting the controller parameters, for instance, the following adjustment criteria are suitable:
- Ziegler and Nichols
- Chien, Hrones and Reswick

15 Downloading the Operating System

In the menu **`Transfer`** of the SYMPAS programming interface the operating system can be updated.
For this purpose, operating system files (*.OS) are made available on the internet (http://www.jetter.de) by JETTER AG.

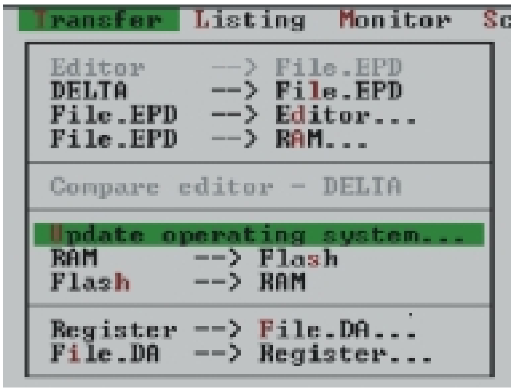


Fig. 40: SYMPAS programming interface - Menu item: `Transfer`

Note!

Prior to downloading the operating system the timeout period must be set to 4000 ms in the menu **`Special/Interface`** of the SYMPAS programming interface (Default).

In order to download the OS the RUN-STOP-LOAD selector switch of the CPU must be set to LOAD when the control system is being switched on.

Appendices

Appendix A: Glossary

Sampling interval:	An ADC requires a certain time, i.e. the sampling interval, to convert an analogue value to a digital value.
Analogue	A parameter, e.g. voltage, which is steplessly adjustable. Contrasted with digital.
Resolution	Analogue-to-Digital and Digital-to-Analogue converters:
	The resolution may be expressed as the number of bits in the digital value that corresponds to a full-scale analogue value. For example, the analogue range may be a voltage between -10 V and +10 V or a current range between 0 and 20 mA.
Bit-coded register	Individual bits of such a register can be set or reset.
Digital	Binary presentation of a parameter, e.g. time. This parameter in digital representation can be changed in given steps only, that is in binary mode. Contrasted with analogue.
Electro-Magnetic Compatibility	Definition according to the EMC regulations:
	"EMC is the ability of a device to function in a satisfactory way in electro-magnetic surroundings without causing electromagnetic disturbances itself, which would be unbearable for other devices in these surroundings."
Impedance	Impedance consists of the ohmic resistance and the reactance.
	The ohmic resistance is independent of the frequency.
	The ohmic resistance is proportionally dependent of the frequency.
Actual temperature	The actual existing temperature.
Integral-action time	Time constant of the integral-action

	component.
„Low Voltage“ Directive	To be considered when using electric devices of a rated voltage between 50 and 1000 V AC and between 75 and 1500 V DC.
Process	A physical system (environment, machine) on which the PID controller has a stabilising effect.
PROCESS PLC	Advanced control system of the JETTER AG in contrast to the conventional programmable logic controller.
Contact spacing	Standardised spacing (pitch) between two contacts of a connector (male/ female) adjacent to each other.
Register	A high-speed memory for a group of bits placed in a microprocessor or in another electronic device where data can be buffered for a specific purpose. On JETTER controllers, usually, these are 24 bit wide storage positions in a remanent RAM.
Oscillation fatigue limit	The device can permanently or shockwise be exposed to a vibration defined in the standard.
Sensor	Electronic detector, pick-up.
Actuator	An actuator boosts the output signal of a controller or converts it to another physical quantity.
Manipulated variable	A quantity that the controller applies to the controlled system.
Derivative-action time	Time constant of the derivative-action component.
Ripple - Smoothing - Filtering	Ripple: The percentage of AC left on a DC signal after rectifying. Filtering: Circuit configuration with a RC or LC component in order to achieve more smoothness or a lower ripple of the DC voltage.

Appendix B: List of Abbreviations

AC	Alternating Current
CE	Communautés Européennes Europäische Gemeinschaften
DC	Direct Current
EMC	Electro-Magnetic Compatibility
EN	Europäische Norm European Standard
GND	Ground
IEC	International Electrotechnical Commission
MUX	Multiplexer
PE	Protected Earth
PID	Proportional-Integral-Differential (Controller)
PWM	Pulse Width Modulation
SELV	Safe Extra Low Voltage: Voltage up to 60 V, galvanically separated from the network.
SUB-D	Type name of a plug-in connector
SYMPAS	Symbolic Program Processing language

Appendix C: Index of Illustrations

Appendix D: Index

T

U