# Manual for

# DAC-42(A)

**DIGITAL AMPLIFIER AND CONTROLLER CARD FOR ALL KIND OF PROPORTIONAL VALVES**

| | Model Code | SW Version | Remarks |
|---|---|---|---|
| **Applicable for SW Versions:** | DAC-42(A)-x-PN-x<br>Operation Mode: 01, 02, 03, 04, 06, 08, 10, 11 | **V 11.xx** | xx = 01 or higher<br>* = a, b, ….<br>x = additional options |


**HCS Hydraulic Control Systems GmbH**
**Neuffener Str. 29**
**D-72636 Frickenhausen**
**Germany**
**Telefon: (+49) 7025 - 911 007**
**Telefax: (+49) 7025 - 911 008**
**Email: info@h-c-s-gmbh.de**

## Revision History

| Revision | Date | Author | Description |
|---|---|---|---|
| R 1.0 | 22.01.2024 | Bm | First edition |

# Table of Contents

# 1 Abbreviations

| Abbreviation | Description |
|---|---|
| DAC | **D**igital **A**mplifier and **C**ontroller |
| EMC | **E**lectro **M**agnetic **C**ompatibility; properties of the unit in order to comply with the EMC directive |
| USB | **U**niversal **S**erial **B**us; interface for communication between a PC and the DAC-4x(A) |
| PC | **P**ersonal **C**omputer |
| µC | **M**icro **C**ontroller; CPU used inside the unit to control all functions |
| mamsl | **m**eters **a**bove **m**edian **s**ea **l**evel |
| IP65 | Protection Class; solid particle protection and protection level against water ingress |
| LED | **L**ight **E**mitting **D**iode |
| PWM | **P**ulse **W**idth **M**odulation; method to control the output current at low power losses |
| Ub | Supply voltage |
| FS | **F**ull **S**cale |

Table 1: Abbreviations

# 2 Terms and Definitions

| Term | Description |
|---|---|
| Amplifier Card | Amplifier for 19” rack system in Eurocard format |
| HCSTool | Software tool provided by HCS in order to adjust and parametrize DAC-4x(A) units |
| CANopen | Bus interface for digital communication, defined by CIA |
| PROFIBUS | Process Field Bus interface for digital communication, defined by BMBF (German department of education and research) and used by Siemens |
| PROFINET | Process Field Net interface for digital communication, the standard is maintained and supported by PROFIBUS & PROFINET International |
| ETHERNET/IP | (IP = Industrial Protocol) is an industrial network protocol that adapts the Common Industrial Protocol to standard Ethernet |
| USB-C connector | Connector type used on the USB interface |

Table 2: Terms and definitions

# 3 General information

## 3.1 Abstract

This document describes the function of the HCS digital amplifier and controller card “DAC-4x(A)”.

This manual was prepared with great care and the contents reflect the author’s best knowledge. However, the possibility of error remains and improvements are possible. Please feel free to submit any comments regarding errors or possibly incomplete information to HCS GmbH.

**⚠ DANGER**

**Read this manual carefully before working with the digital amplifier cards!**

**The general and especially the safety instructions of this manual have to be observed in any case!**

**Electronic equipment of all kind can have component failures or software may lead to unpredictable reactions. Secondary safety measures must be taken in order to ensure safety under all circumstances. This is especially the case for safety critical applications.**

**Furthermore, it is the responsibility of the user to always comply with applicable safety standards (e.g. EN 13849) and to implement a system architecture capable to cover all safety requirements. The unit itself does not comply with any performance level given in EN 13849 or any SIL level as per EN 61508.**

**Applicable laws and safety standards have to be observed at any time. Before commissioning or using the equipment the necessary risk analysis must be conducted and suitable protection measures must be taken.**

**HCS GmbH refuses any liability in case of omission to comply with these requirements.**

## 3.2 Scope

With the help of this document a professional, competent and trained user should be able to install, wire, connect, commission, set parameters, perform fault analysis and fix issues. Also, the document will provide the user any technical and operational details necessary to safely operate the product. It remains the user’s responsibility to set-up a safe working environment while operating the product.

The information contained in this manual is valid at the time of this version's release. See footer for version number and release date of this manual.

### 3.2.1 Copyright



### 3.2.2 Documents place of storage

This manual and all other associated documentation for hardware and software must always be kept in a location where they will be readily accessible and close to the digital amplifier and controller card DAC-4x(A) or the equipment in which it is installed.

### 3.2.3 Warning signs, symbols and notes

**⚠ DANGER**

**Identifies safety instructions that are intended to warn of an immediate and impending danger to life and limb. Failure to observe these safety instructions will – with a very high probability - lead to death, serious personal injury (disablement)!**

**⚠ WARNING**

**Identifies safety instructions that are intended to warn of potential danger to life and limb. Failure to observe these safety instructions might lead to death, serious personal injury (disablement)!**

**⚠ CAUTION**

**Identifies safety instructions that are intended to warn of slight personal injury. Failure to observe these safety instructions might lead to slight personal injury.**

**NOTICE**

**Failure to observe this safety notice can result in property damage!**

ⓘ Identifiers important information

Other identifiers (when applicable):

| Identifier | Description |
|---|---|
| ●/ - | Listings |
| □ | References to another chapter, page, table or figure in this manual |
| blue text | Hyperlink within the document |
| 1., 2., … | Steps in a procedure that should be performed in consecutive order |
| «ER» | LEDs of the amplifier (for example, «ER» |
| < > | parameter name |
| [xx] | Pin number of the rear connector |
| "…" | Used for references |

Table 3: Identifiers

### *3.2.4 Safety and liability*

**NOTICE**

**The information in this document is subject to change without further notice.**
**HCS assumes no responsibility for any faults that may appear in this manual.**

**This operating manual represents the knowledge of HCS and during the drafting of this operating manual the greatest possible care was taken. Nevertheless, HCS disclaims any responsibility and liability claims for individual applications of the user. This is particularly true in cases of non-compliance, omissions, faults, misinterpretations and misunderstandings.**

**Applicable laws and safety standards have to be observed at any time. Before commissioning or using the equipment, the necessary risk analysis must be conducted and suitable protection measures must be taken.**

**This manual only describes the functionality and influence of the parameters. The described software functionality can be used in various amplifier models which can be implemented in a vast range of applications. Hence it is not possible to assume liability for the influence of the parameters.**

**HCS GmbH refuses any liability in case of omission to comply with these requirements.**

**⚠ WARNING**

**In the case of applications with critical safety requirements or where accident prevention regulations must be observed, it may be necessary to isolate the components from the solenoids with relays in e-stop circuits.**

**To switch off the enable signal (0 V at terminal 8d) is insufficient. In these cases, hydraulic and/or mechanical safety measures to stop the drive must be provided (e.g. through switching valves with position monitoring).**

**⚠ WARNING**

**All types of proportional directional valves, which may be used in all kinds of different environments and applications, can eventually fail and hence are able to cause damage.**

**It is the customers obligation to analyze all safety related aspects of the application. It is within the full responsibility of the machine builder and/or system integrator to make the final selection of products and reassure and confirm that all requirements regarding safety and performance are met.**

**The procedure of selecting the proper control system and related safety levels is covered by the machine directive EN 13849.**

**NOTICE**

**Liability for any content of this document is rejected!**

**Any claims against HCS GmbH – based on whatever legal reason – resulting from the use of the information, programs, engineering and performance data etc., described in this document shall be excluded. Exclusion in such cases shall not apply for mandatory liability, e.g. under the "Product Liability Act", in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition. Damages for breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.**

**We reserve the right to make changes to this document at any time without prior notice.**

### *3.2.5 General information*

**⚠ WARNING**

**ESD (electrostatic discharge) may damage components on the card. In order to prevent any damage please always follow the recommendations below!**

**Discharge static voltage from your body by using best practice.**
**Work in safe environment and do not use any devices in the working surrounding which could generate or hold static charge. Avoid using the product in areas where floors or surfaces are composed of materials that could generate static charges.**

**Handle all equipment very carefully and do not touch exposed pins or components.**

**Store and transport the units only in its original packaging.**

**⚠ WARNING**

**During commissioning, particular attention must be paid to the correct design and realization of the wiring. This must be checked before applying the supply voltage.**

**To avoid collisions, the safety devices and limit switches must be activated. All safety regulations are to be observed.**

**Monitoring the fault signal (terminal 14d) is recommended.**

**Malfunctions can occur with the following:**

- Changes to the settings made by the supplier.
- Neglecting of operating parameters (e.g. supply voltage, application of inadmissible signals on inputs or outputs, ambient conditions, wiring, unsafe loads such as motors, contactors, relays, ohm loads, etc.)
- Faults in series-connected controls components and set points or actual values
- Faults in the subsequent hydraulic components
- Removing of solenoids connections.

## *3.3 Terms and conditions*

Please pay attention to our general terms and conditions (available on request).

## *3.4 Delivery state (default setting)*

The product is shipped in a ready-to-use state (default settings). After correct installation and setting of all parameters relevant for the application, the amplifier card is ready for use.

## *3.5 Correct product use*

The digital amplifier and controller cards “DAC-4x(A)” are used for the following applications:

- Controlling of proportional valves (namely directional control valves); with or without feedback (depending of operation mode).
- To convert set point signals into a current signal to control one proportional valve and spool position feedback in the aforementioned valves.
- For industrial applications only.

The operation of other inductive or resistive loads is not recommended (motors, heaters, etc.). If you are considering any applications like this please contact HCS GmbH.

If used correctly, the safety of the user as well as the safe and proper function of the amplifier DAC-4x(A) is given.

## *3.6 Non-intended product use*

The digital amplifier and controller cards “DAC-4x(A)” should not be used in the following cases:

- if noticeably damaged
- when the electrical connections are damaged
- if they do not function correctly
- after incorrect handling or storage
- in non-appropriate applications or environments.

In these cases, the digital amplifier and controller card must be disabled and secured against accidental restarting.

In the case of applications with critical safety requirements or where accident prevention regulations must be observed, it may be necessary to isolate the components from the solenoids with relays in e-stop circuits. In these cases, hydraulic and/or mechanical safety measures to stop the drive must be provided (e.g. through switching valves with position monitoring).

During commissioning, particular attention must be paid to the correct design and realization of the wiring. This must be checked before applying the supply voltage.

To avoid collisions, the safety devices and limit switches must be activated. All safety regulations are to be observed.

Malfunctions can occur when:

- Changes to the settings made by the supplier.
- Neglecting of operating parameters (e.g. supply voltage, application of inadmissible signals on inputs or outputs, ambient conditions, wiring, unsafe loads such as motors, contactors, relays, ohm loads, etc.)
- Faults in series-connected controls components and set points or actual values
- Faults in the subsequent hydraulic components
- Removing of valve connection.

## 3.7 Selection and qualification of personnel

Only qualified users may work with the amplifier. Qualified users are properly trained experts with the required knowledge and experience. In particular, these experts must have the authorization to bring into operation systems and power circuits in accordance with safety engineering standards. They must be familiar with safety concepts common in automation.

### 3.7.1 Inquiries and ordering

To order the product, the complete order code is requested.

See □ Chapter "4.3 Model code", page 15

Please pay attention to our general terms and conditions (available on request) and forward orders and enquiries to the companies as listed on our web page:

# Sales Partners (h-c-s-gmbh.de)

https://www.h-c-s-gmbh.de/en/sales-partners

They are our official distributors/partners. We reserve the right to forward any direct inquiries or orders to our distributors/partners. Nevertheless, we will be available for any necessary technical support.

Germany and other countries without distributor/partner:

Neuffener Strasse 29
D-72636 Frickenhausen
Tel: (+49) 7025 - 911 007
Fax: (+49) 7025 - 911 008
Email: info@h-c-s-gmbh.de
www.h-c-s-gmbh.de

### 3.7.2 Service and repair

**⚠ WARNING**

**Do not attempt - under any circumstances - to repair the product yourself**

After repair, certain adjustments and test procedures must be performed; this can only be made by qualified and authorized personnel. Products that need repair can be sent to the addresses:

Please enclose a detailed description of the error, malfunction or failure with the sent item and state the serial number and the purchase date. This will speed up the process and guarantees a fast and reliable repair.

In the case of a fault or a malfunction, your distributor can give you instructions on the phone or in writing before accepting a repair order. For service and repair, we offer experienced and qualified personnel. In case you need our assistance, please contact the address:

See □ Chapter "3.7.1 Inquiries and ordering", page 10

### 3.7.3 Cleaning, storage, transport

The product should only be transported and stored in the original packaging to ensure suitable protection against mechanical damage as well as electrostatic discharge.

- Keep the amplifier card away from moisture and dust
- Observe the allowed storage and transportation temperature range
- If it should be necessary to clean the amplifier, we recommend sending it back to the manufacturer or any distributor and partner:

See □ Chapter "3.7.1 Inquiries and ordering", page 10

**⚠ CAUTION**

**Unpacking and handling should be left to suitably trained personnel.**
**Beware of damaging the amplifier by electrostatic discharge**

### 3.7.4 Delivery state (default setting)

The product is shipped in a ready-to-use state (default settings). After correct installation and setting of all parameters relevant for the application, the digital amplifier and controller card is ready for use.

## 3.8 Introduction

The digital amplifier and controller card DAC-4x(A) features leading edge technology. This electronic device meets the industrial standards for EMC. This ensures a high interference security and low interference emission.

The performance characteristics are possible through the use of the most current microprocessor technology (32-bit floating-point CPU) combined with other state of the art components. In addition to all control functions, the microprocessor also handles closed-loop control. The system features are essentially determined by the software and provide reserve capacity for further developments and adaptations.

The following features distinguish the DAC-4x(A) series:

- Fully digitized amplifier card with the advantage of
  - no on-board potentiometer
  - no jumper settings required
  - digital setting and display of all parameters by means of PC with HCSTool
  - user safety during parameterization
- flexible and reliable system:
  - use of a modern 32-Bit floating point µC
  - high performance and yet power reserve
  - high reliability and safety through the use of a integrated watch-dog and reset function
  - variable settings for magnetic systems making high flexibility possible
- functional use of the interface:
  - change of selected parameters “on-the-fly” without interference or interruptions of the working cycle
  - analyzing of system performance through selection of display parameters with the PC and by using the 4-channel oscilloscope function included in HCSTool
  - bus interfaces available (PROFIBUS/PROFIDRIVE, PROFINET, ETHERNET/IP, CAN-OPEN)

## 3.9 General Applications

The amplifier card DAC-4x(A) is used for:

- With or without electrical feedback transducers
  - proportional directional valves direct and pilot operated
  - proportional flow control valves
  - proportional pressure reducing valves
  - proportional pressure regulating valves
  - cartridge valves
  - servo valves with torque motors
- Controlling of hydraulic motors, - installations and – systems, e.g.:
  - position
  - speed
  - pressure
  - revolutions per minute

- torque
- power etc.

- Volume flow control and pressure control of pumps (if the occasion arrives: power limiting function; controlling valve spool position)
- Controlling of different process values:
  - P/Q controlling
  - pump controlling
  - controlling of pressures
  - controlling of pilot- and main stage
  - cascade controlling of components etc.

# 4 Product overview and description

## *4.1 Technical features*

The system features of the DAC-4x(A) are the following:

- One differential command input (set point). Range of 0 ± 10 respectively 0 or 4 ... 20 mA, resolution 16-bit.
- Two single ended command point inputs. Range of 0 ± 10 respectively 0 or 4 ... 20 mA, resolution 16-bit.
- Two single ended feedback inputs. For current and voltage. Wide range of adjustability, resolution 16-bit.
- Two optional inputs for LVDT signals. 1 or 2 inputs with optional extension board; information on request.
- Integrated reference voltage supply of ± 10 V (10 mA max.), to supply set point potentiometer or actual value transducer.
- 4 (optional 5) recallable digital adjustable set points.
- Direction externally set through “+“ and “-“ inputs.
- Enable signal for output stages.
- Reset-Ramp for fast ramp function zeroing.
- Status outputs "error“ and "comparator“.
- All digital inputs and outputs are optically isolated for functional security.
- Display with four 7-segment indications and six keys for easy handling and operation.
- Function indication through front panel by LEDs.
- Additional switching output (24 V, max 1A) to directly disable safety valves.
- Analog outputs to perform controller functions and / or enable subsequent electronic devices and monitoring (± 10 V, 12-bit resolution). One is also selectable as current output 4 ...20 mA.
- Additional front panel test jacks (2 mm) for easy commissioning. Test jacks for set point (SET)*1, for feedback 1 (FB1), for solenoid currents (A) and (B) and monitor input (MON) display for analog display of all parameters (± 10 V, 12-bit resolution).
- USB interface with USB-C connector
- Serial interface RS232 (instead of USB_C interface; optional on request)
- Optional available: PROFIBUS, PROFINET, ETHERNET/IP and CANopen interfaces available.

## 4.2 Technical data

### 4.2.1 General technical data

| Feature | Range, characteristics |
|---|---|
| Supply voltage and<br>Control voltage for digital inputs | 12 V DC - 10 % … 24 V DC + 20 %; residual ripple < 10 % (max. 50 VA power draw)<br>current draw per input < 20 mA |
| Solenoid system selection | 0.8 A; 1.1 A; 1.3 A; 1.6 A; 2.4 A; 2.7 A and 3.5 A (intermediate values adjustable)<br>Servo valve current ranges available on request! |
| Ambiant operating temperature | - 40 °C to + 70 °C |
| Storage temperature | - 45 °C … 85 ° C |
| Humidity (relative air humidity) | max. 95 % non condensing |
| Max. elevation | 2,000 m (mamsl) |
| Class of protection | IP20 when installed in 19" rack system |
| Connector | In accordance with DIN 41612, 48 pol form F gold plated |
| EMC | In accordance with the applicable industrial standards (CE - conformity) *3 |
| Dimensions front panel / PCB | 50,5 x 128,4 mm; 10TE / 3 HE; 100 x 160 mm euro format |
| Analogue set values inputs | 3 inputs with 16 Bit resolution (1 x differential; 2 x single ended; 0 … +-10 V, 0/4 … 20 mA) |
| Analogue feedback U/I | 2 inputs with 16 Bit resolution (current and voltage with wide range of adjustability) |
| Analogue feedback LVDT *2 | 1 or 2 inputs with optional extension board; detailled information on request |
| Digital inputs | 8 inputs (S1.01 … S1.04, ENABLE, RAMP 0, DIRECTION +, DIRECTION -) |
| Solenoid current (output) | 2 PWM output stages, each for up to 3,5 A (with over-energ. and quick de-energization) |
| Analogue output | 1 output with 12 Bit resolution, 0 … +-10 V or 4 ... 20 mA (selectable) |
| Monitor output | 1 output with 12 Bit resolution, 0 … +-10 V |
| Digital outputs | 2 outputs, voltage level 0 V / 24 V, 10 mA (ERROR, COMPARATOR) |
| Test jacks at front *2 | Current A and B, sensor 1 (Fb1) set value (S1.06), Monitor and GND |
| Auxiliary reference voltage | +- 10 V, max. output 10 mA |
| Optional digital I/O signals | 3 for variable usage, voltage level 24 V or 5 V |
| Interface 1 | USB-C or (optional) RS232, 9-pol SUB-D-conncetor at face plate |
| Optional interface 2 *2 | PROFIBUS DP; PROFNET; other bus interfaces on request |
| Display and keypad *2 | 4-digit 7-segment display, 6 keys, status LED's |
| PWM frequency | 22.2 kHz - independant from dither frequency (customer specific frequencies available) |
| Cycle times | Closed loop 1 and 2: 0.1 msec |

Table 4: General Technical Data

**Detailed technical data in the "DAC-4x(A) Manual E Rxx yyyymmdd.pdf"**
**please ask HCS GmbH if needed.**

### 4.2.2 Technical data for optional bus interfaces

| Feature PROFIBUS | Range, characteristics |
|---|---|
| Supply voltage | Via DAC-4(A) motherboard |
| Temperature ranges, EMC, Mounting/housing | Refer to page 2 |
| PROFIBUS-DP | - Certified by PNO<br>- Supports PROFIBUS-DP Slave in accordance with IEC 61158<br>- Supports PROFIBUS DPV1<br>- Maximum 244 Byte in-/output data<br>- Supports up to 12 Mbaud (autodetect)<br>- Electrical isolated and opto-decoupled |
| Connection / Type of connector | RS485, Sub-D 9-pole female |
| Status signals | LED „Buserror“ (red): DAC-4 error LED is used |
| Address selection | DIP switch 1- 8, each on/off |

| Feature PROFINET | Range, characteristics |
|---|---|
| Supply voltage | Via DAC-4(A) motherboard |
| Temperature ranges, EMC, Mounting/housing | Refer to page 2 |
| PROFINET | Approved by PNO<br>- Meets the standard IEC 61158 and IEC 61784<br>- LAN 10/100Base-T(X)<br>- 2 x RJ-45 LAN (Daisy Chain)<br>- Cycling data exchange RT and IRT with PROFINET IO-Controller<br>- Sending and receiving of diagnostic and process alarms<br>- I&M0...4-data available<br>- Supporting of PROFINET naming (device name) and TCP/IP addressing<br>- Fast Startup functionality supported<br>- Shared Device supported<br>- Media Redundancy Protocol support.<br>- Electrical isolated interface |
| Connection / Type of connector | In/Out: 2 x RJ45 (integrated switch) |
| Status signals | Power (green), Error (red), Maint (yellow), Sync (yellow), Status (yellow) |
| Future usage | DIP switch 1- 3, each on/off |

Table 5: Technical Data for Optional Bus Interfaces

### 4.2.3 Block diagram hardware

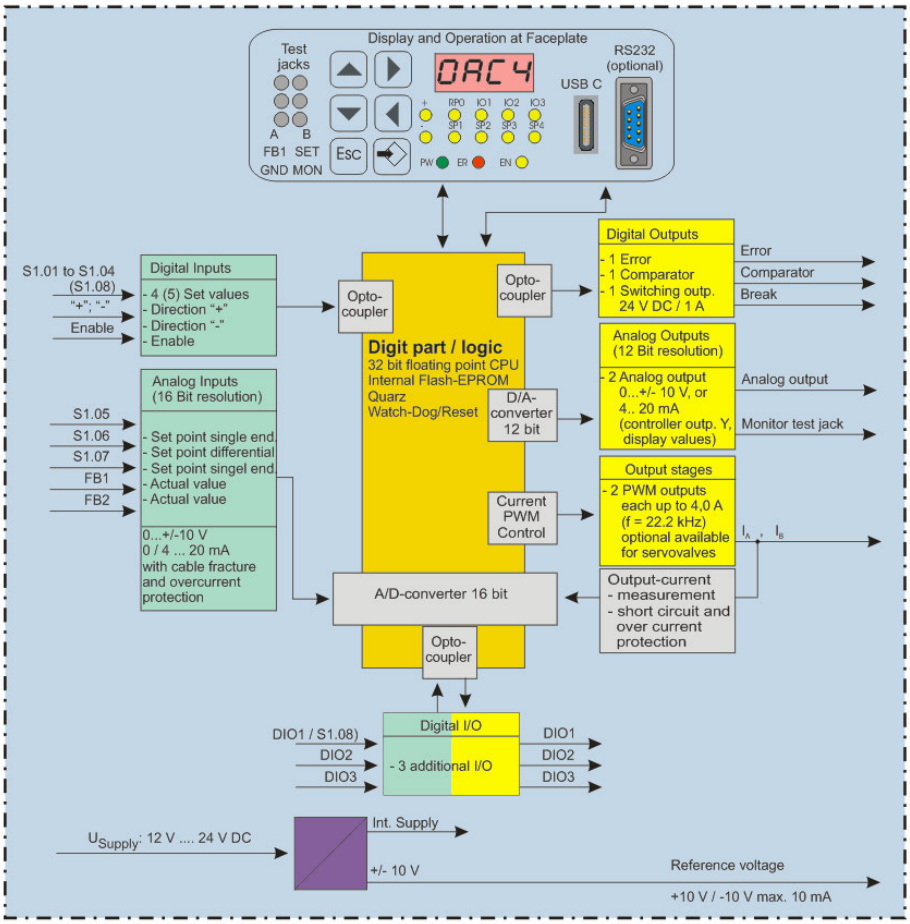
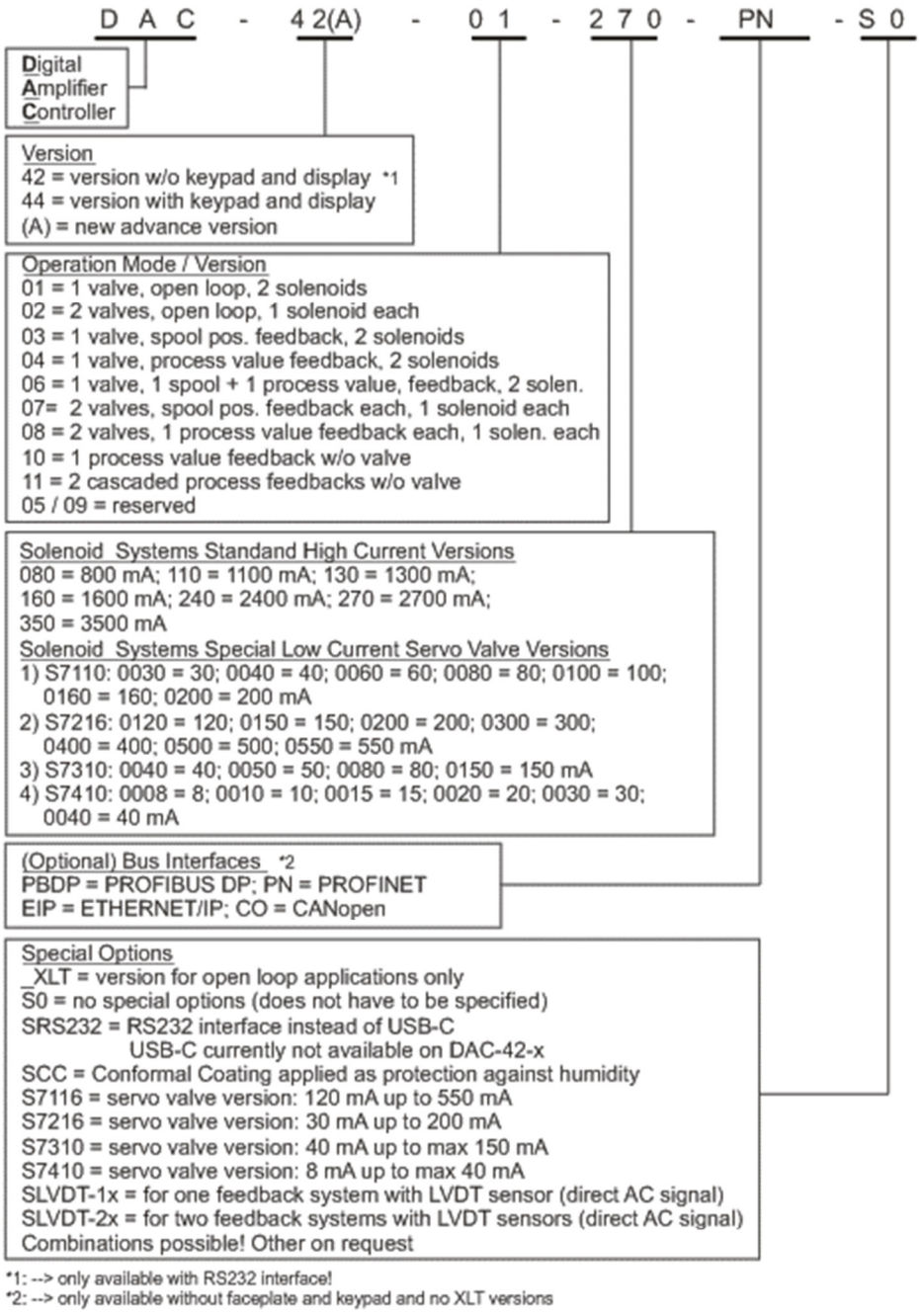



Figure 1 : Block Diagram Hardware

## 4.3 Model code




**Model code examples:**

Version with display for one valve with 2.7 A solenoids closed loop operation in mode 3. With USB-C interface:

***DAC-44(A)-03-270-S0***

Version without display for one valve with 0.8 A solenoids open loop operation in mode 1 with RS232 interface and conformal coating:

***DAC-44(A)-01-080-SRS232-CC-XLT***

Figure 2 : Model code

**Remark:** Servo valve versions are covered in a separate manual

# 5 Installation

## 5.1 Mounting

- Compare card type (refer to label on rear connector) with part list / diagram.
- The card generally could be mounted in any direction. However, it is recommended to choose a vertical orientation for better convection cooling. Also, this orientation will be obviously better suited for adjusting parameters or monitoring display values via the keypad and display (applicable for DAC-4**4**(A) versions only).
- The amplifier card should be mounted in a shielded environment (e.g. control cabinet).
- The amplifier card module has to be mounted either in a rack according to the 19" rack standards (EIA 310-D, IEC 60297 und DIN 41494 SC48D).
  it is also possible to install the card either in an appropriate card holder or HCS own universal rack:

| Card holder (HCS article number: 5 741 010 00) | Universal rack (HCS article number: 5 741 000 00) |
|---|---|
|  |  |

Figure 3 : Card holder and universal rack (not to scale!)

Dimensions Eurocard for 19" rack mounting:

For 3 HE: 100 mm × 160 mm, connector on small side
Faceplate dimensions DAC-4x(A): 3 HE x 10 TE (= 128,4 x 50,5 mm)

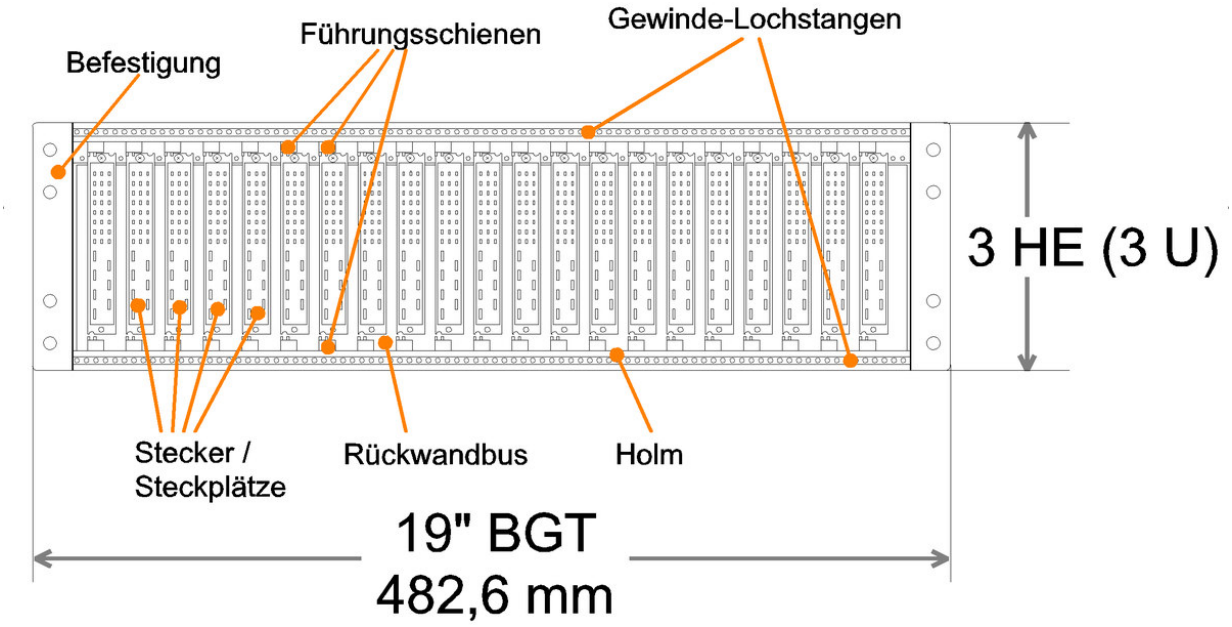



Figure 4 : General example for 19" Rack systems 3 HE

## 5.2 Connection

### 5.2.1 General recommendations

The general supply voltage for the unit is at the terminals:
[26bd] = 0 V and [32bd] = +UB is 12 V DC - 10 % … 24 V DC + 20 %; residual ripple < 10 %
(at supply voltage down to 12 V DC some limitations may be applicable).

Output stages are electronically protected against short circuit and overload. The amplifiers should be protected with a quick-acting fuse.

The solenoids are connected at terminals [22bdz] and [28bdz] for solenoid A; terminals [24bdz] and [30bdz] for solenoid B. The terminals [bdz] are bridged on the amplifier. We recommend using all three terminals for better load distribution.

If no electrical isolation of logic inputs or outputs is required, terminal [26bd] may be bridged to terminal 2d and terminal [32z] to terminal [32bd].

Terminal [18d] is the GND for the digital signals.

**When replacing amplifier cards model VRD350 and VRD355, please note that the connection for PE was at terminals 18bd. If PE is connected to these terminals, disconnect.**

### 5.2.2 Wires and preparation

Connection wires have to comply to the following specifications:

| Feature | Required |
|---|---|
| Wire type | Hookup cable; stranded |
| Cross section: Solenoids and supply | min AWG 16 / 1.5 mm² |
| Cross section: Signal, command and sensor: | min AWG 20 / 0.5 mm² |
| Wire length | Max. 50 m (> 50 m consult factory) |
| Skinning length | 7 mm |
| Soldering of wires | Not permitted! |

Table 6: Wires recommendation

⚠ WARNING

**The assembly contains electronic components. Incorrect handling or operation can lead to damage through electrostatic discharge (ESD). Only trained personnel should work with the unit. All safety instructions must be observed. Damage may result if the card is disconnected while the power supply is still on. Avoid such actions under all circumstances. The information in this document is subject to change without notice.**

⚠ WARNING

**Avoid under all circumstances to use logical signals from the card (i.e. “Error”) for switching machine safety circuits (refer also to EU standard EN13849)!**

### 5.2.3 EMC

Devices DAC-4x(A) series are class “A” equipment and therefore only suitable for industrial surroundings.

The distance between a source of interference (device emitting interference) and an interference sink (a device under the influence of interference) is very important. The greater the distance between the source of interference and the sink, the smaller the effects on the equipment will be. In other words, the closer a device is placed to the source of interference, the greater the interference amplitudes. For this reason, a minimum gap of 0.25 m should be kept between the amplifiers and strong sources of interference. The following devices are to be regarded as strong sources of interference:

- Switching power supply units
- Frequency converters
- Digital drive modules
- Mains filters with wiring (even if shielded)
- AC/DC commutator motors
- Motor cables (even if shielded)
- Switched inductances, even if anti-interference measures have been taken (solenoid valves, contactors, relays, etc.)

One of the most common input points for interference is wiring. If interfering cables are laid at least 0.25 m away from cables susceptible to interference, the influence on each other can be minimized. Parts of the amplifier wiring may be susceptible to interference (analog set point and actual value, solenoid cables). If these cables are laid parallel over a distance longer more 10 m, the necessary distance between them must be increased. Cables susceptible to interference should never be laid parallel to motor cables. The influence is the least when the cables cross particularly at an angle of 90°.

However, interference can also arise from cables in the amplifier wiring system, in particular solenoid cables.
Examples of devices which are particularly susceptible to interference:

- Office PC’s
- Sensors with small output voltages / currents
- Capacity proximity switches
- Audio equipment (television, hi-fi, radio, etc.)
- Devices which do not meet the EMC guidelines

### 5.2.4 Specific recommendations for wiring and control cabinets

The following rules and tips are by no means to be complete. Since various electronic components are used in a variety of different ambient conditions, these guidelines only represent a compromise. The actual design of the wiring also depends on the interference emission and interference sensitivity of each individual component.

- Use shielded and twisted-pair cables for the solenoid connections. The shield must be grounded (PE Protective Earth) at both ends. The capacity should be ca. 120 pF/m. If the cables are up 100 m long, their cross-section has to be 1.5 mm 2 and 2.5 mm 2 for cables longer than 100 m.
- Shields of digital signal lines are to be connected at both ends to a PE, a good conductivity of the connection should be provided.
- Set point and actual value connections should have shielded and twisted-pair cables. The analog signal line shields are to be connected, with low impedance, at both ends to PE
- In environments with high interference, use double shielded cables for set point and actual value connections. The inner shield is only grounded at one end, the outer at both ends.
- In the event of low-frequency interference on the analog signal lines (measured value fluctuations), the shield should be connected at one end. Preferably, use a corresponding compensation of potential (see also the following point).
- Use only cables with a Cu shielding grid and a covering of > 85 %. Avoid screening films. Metal foils must be avoided.
- The shield should not be interrupted along the entire length of the cable. If contactors, safety switches, chokes, etc. have to be used in the wiring, the installation of a metallic housing with a high HF shield may be necessary.
- Shield terminals are to be connected over a large area to the shield rail to function well.
- The shield rail must be installed close to the cable duct in the control cabinet.
- The metallic parts of the electrical cabinet are to be connected with low impedance on large areas. Make the desired connection with mechanical aids such as scratch plates if necessary. Connect the doors of the cabinet with the shortest possible homogeneous tapes (multiple).
- Solenoid valves, contactors, relays, brakes, etc. must be suppressed directly at the interference source. Suitable suppression devices are, for example, RC networks, diodes or varistors.
- Analog and digital signal lines should preferably enter the control cabinet from only one side.
- Non-shielded lines of an electric circuit must be twisted.
- Auxiliary wires are to be grounded at both ends.
- Avoid unnecessarily long lines. This keeps the coupling impedance low.
- Wiring should preferably not be freely hanging in the cabinet. Lay cables, including auxiliary cables, as close as possible to mounting plates and cabinet housing.

- In the case of a potential difference between the shield connections, a compensation conductor of ⟨10 mm $^2$ should be laid parallel to the shield to reduce the transient current. A multiple connection of the shield to the cabinet casing and thus PE is generally possible. Also, a multiple connection of the shield outside the cabinet is possible.
- If filters are installed, place them close to the source of interference and keep a good surface contact to the cabinet or mounting plate.
- If converters are used, converter filters must be provided. Variable speed motors may have to be connected using shielded lines. All further instructions of the converter manufacturer have to be observed.

The following two pages show illustrations of:

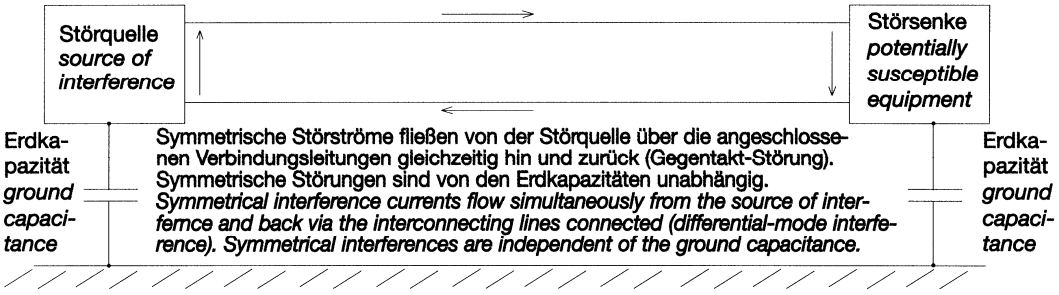
- The most important types of interference and their remedies
- Construction of EMC suitable electric cabinets and systems

The diagrams have been provided by our competent partners for all questions regarding EMC:

**NKL GmbH**
**Birckichstr. 15**
**D-74549 Wolpertshausen**

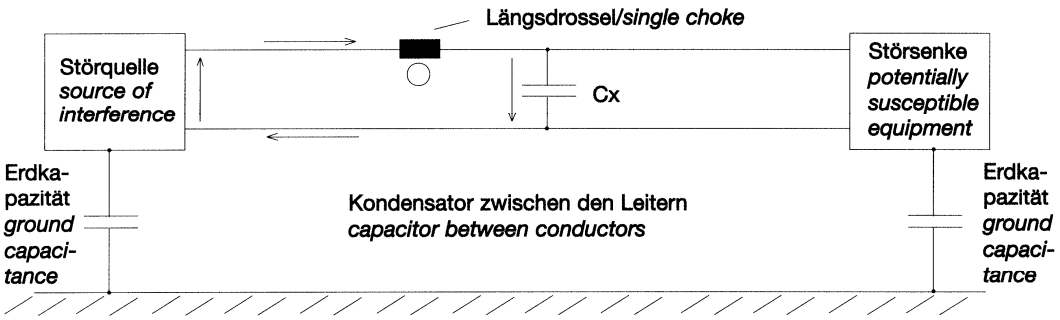
## Die wichtigsten Arten von Störungen und Abhilfen:
## *The most important types of interference and remedies:*

Symmetrische Störungen:
*Symmetrical interference:*




Typische Störquellen, die symmetrische Störungen erzeugen, sind:
*Typical sources of symmetrical interferences:*

Alle Arten von Gleichrichtern, Thyristor-Steuerungen wie z. B. Phasenanschnitt-Steuerungen, Halbleiter-Relais etc.
Symmetrische Störungen treten in der Praxis hauptsächlich leitungsgebunden im unteren Frequenzbereich bis ca. 1 MHz auf.
*All types of rectifiers, thyristor controls such as e. g. generalized phase controls, semiconductor relays, etc.*
*In practice, symmetrical interfernces occur mainly as conducted interference in the lower frequency range up to about 1 MHz.*




Symmetrische Störungen werden entstört durch:
*Symmetrical interferences are suppressed by:*

- Kondensatoren zwischen den Anschlüssen („X"-Kondensatoren): Diese schließen die Störströme kurz, bevor sie die Störsenke erreichen.
- *capacitors between the terminals („X" capacitors):*
  *they short-circuit the interference currents just before they reach the potentially susceptible equipment.*

- Einzel-Drosseln im Strompfad der Zu- und Ableitungen. Diese Drosseln sind für symmetrische Störungen wirksam und erhöhen die Impedanz des symmetrischen Störstromkreises.
- *single chokes in the current path of the incoming and outgoing lines. These chokes are effective for symmetrical interferences and inrease the impedance of the symmetrical interference circuit.*

Figure 5 : EMC – interferences and remedies

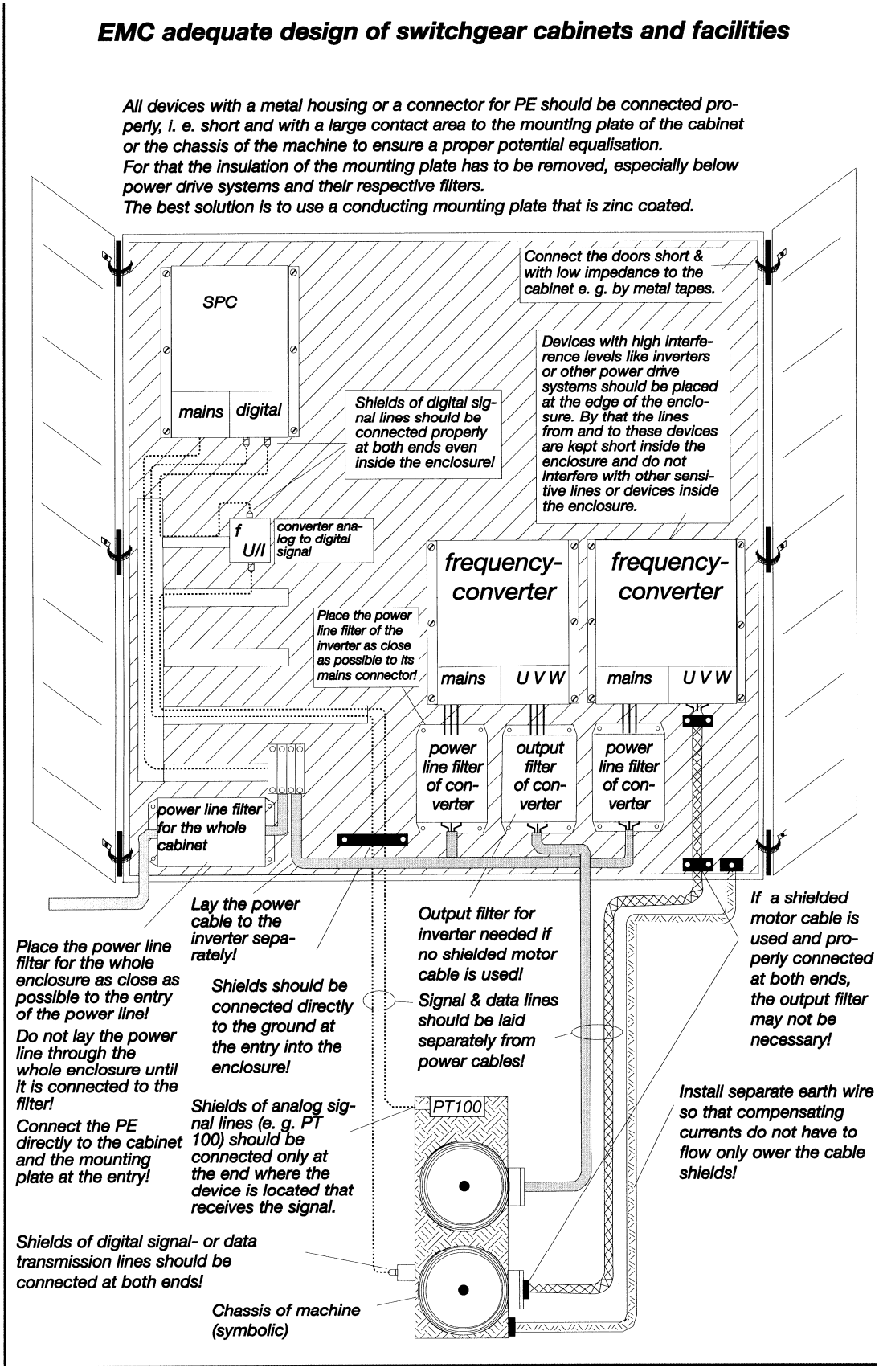



Figure 6 : EMC – adequate design of switchgear cabinets and facilities

## 5.3 Terminal assignment

| Pin | d | b | Z |
|---|---|---|---|
| **2** | 0 V (External) | DIO 1 or S1.08 | - Sign (direction) digital set values |
| **4** | Digital set value 2 (S1.02) | DIO 2 | + Sign (direction) digital set values |
| **6** | Digital set value 3 (S1.03) | DIO 3 | Digital set value 4 (S1.04) |
| **8** | ENABLE (DI 1) | Analog input 1 (Option) | Digital set value 1 (S1.01) |
| **10** | Sensor 1 (FB 1) $U_E$, $I_E$ | Analog output | n/c |
| **12** | Analog set value 6 $U_{E+}$ (S1.06) | n/c | Analog set value 5 $U_{E+}$ (S1.05) |
| **14** | ERROR (DO 1) | COMPARATOR (DO 2) | Sensor 2 (FB 2) $U_E$ |
| **16** | Analog set value 6 $U_{E-}$ (S1.06) | Analog input 2 (Option) | Analog set value 7 $I_E$ (S1.07) |
| **18** | Digital GND | PE | Reset Ramp (DI 2) |
| **20** | Reference output - 10.0 V | Break output 24 V / 1 A | Reference output + 10.0V |
| **22** | Solenoid output A - | Solenoid output A - | Solenoid output A - |
| **24** | Solenoid output B - | Solenoid output B - | Solenoid output B - |
| **26** | 0 V (Power) | 0 V (Power) | Analog GND |
| **28** | Solenoid output A + | Solenoid output A + | Solenoid output A + |
| **30** | Solenoid output B + | Solenoid output B + | Solenoid output B + |
| **32** | + 24 V (Power) | + 24 V (Power) | 24 V (External) |

Table 7: Terminal assignment

## 5.4 Picture of rear connector

Connector: IEC 60603-2 form F 48 pins

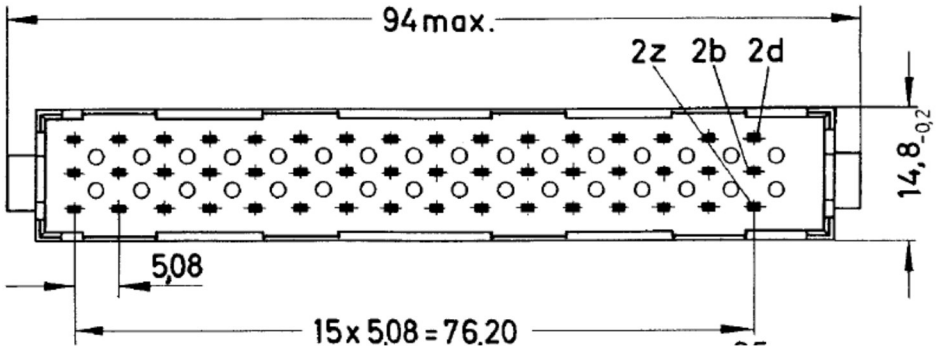



Figure 7 : Connector F48

## 5.5 Operation modes

The setting of parameter <E00> is determining which operation mode is activated. This parameter is factory preset! Only the mode relevant parameters are made available for each of the modes.

Available operation modes:

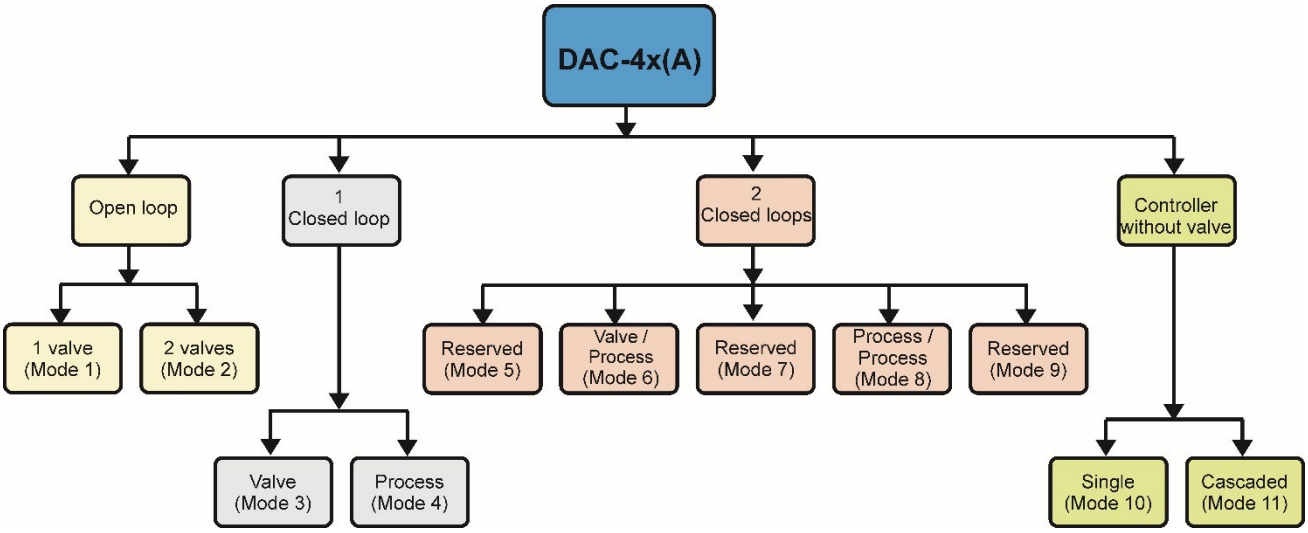



Figure 8 : Available Operation Modes

With the use of parameter <E00> any basic mode of operation may be selected. A change in operation modes will be effective immediately. Recalling the parameters for the newly chosen mode takes only a few seconds. During this time, the display flashes. Only the mode relevant parameter are made available.

| <E00> | Mode description |
|---|---|
| **1** | Open loop, 1 proportional valve with 2 solenoids without feedback |
| **2** | Open loop, 2 proportional valves with 1 solenoid each without feedback |
| **3** | Closed loop valve, single, 1 proportional valve with 2 solenoids and feedback of spool position |
| **4** | Closed loop process, single, 1 proportional valve with 2 solenoids and feedback of process value (position, velocity, pressure, force, torque etc.) |
| **5** | Reserved |
| **6** | Closed loop valve and process, double, 1 proportional valve with 2 solenoids and feedback of spool position and additional feedback of process value (cascaded controller) |
| **7** | Reserved |
| **8** | Closed loop processes, double, 2 independent proportional valve with 1 solenoid each and feedback of two independent process values (e.g. two pressure control systems) |
| **9** | Reserved |
| **10** | Controller function without valve, control of 1 process value; provide set value to follow up electronics (e.g. valve with integrated electronics, frequency converter for AC motor etc.) |
| **11** | Controller function without valve, control of 2 process values (cascaded controller, e.g. position and velocity controller); provide set value to follow up electronics (e.g. valve with integrated electronics, frequency converter for AC motor etc.) |

Table 8: Operation modes

## 5.6 Wiring diagrams for all operation modes

### 5.6.1 Wiring diagram for operation mode 01

Open loop, 1 proportional valve with 2 solenoids

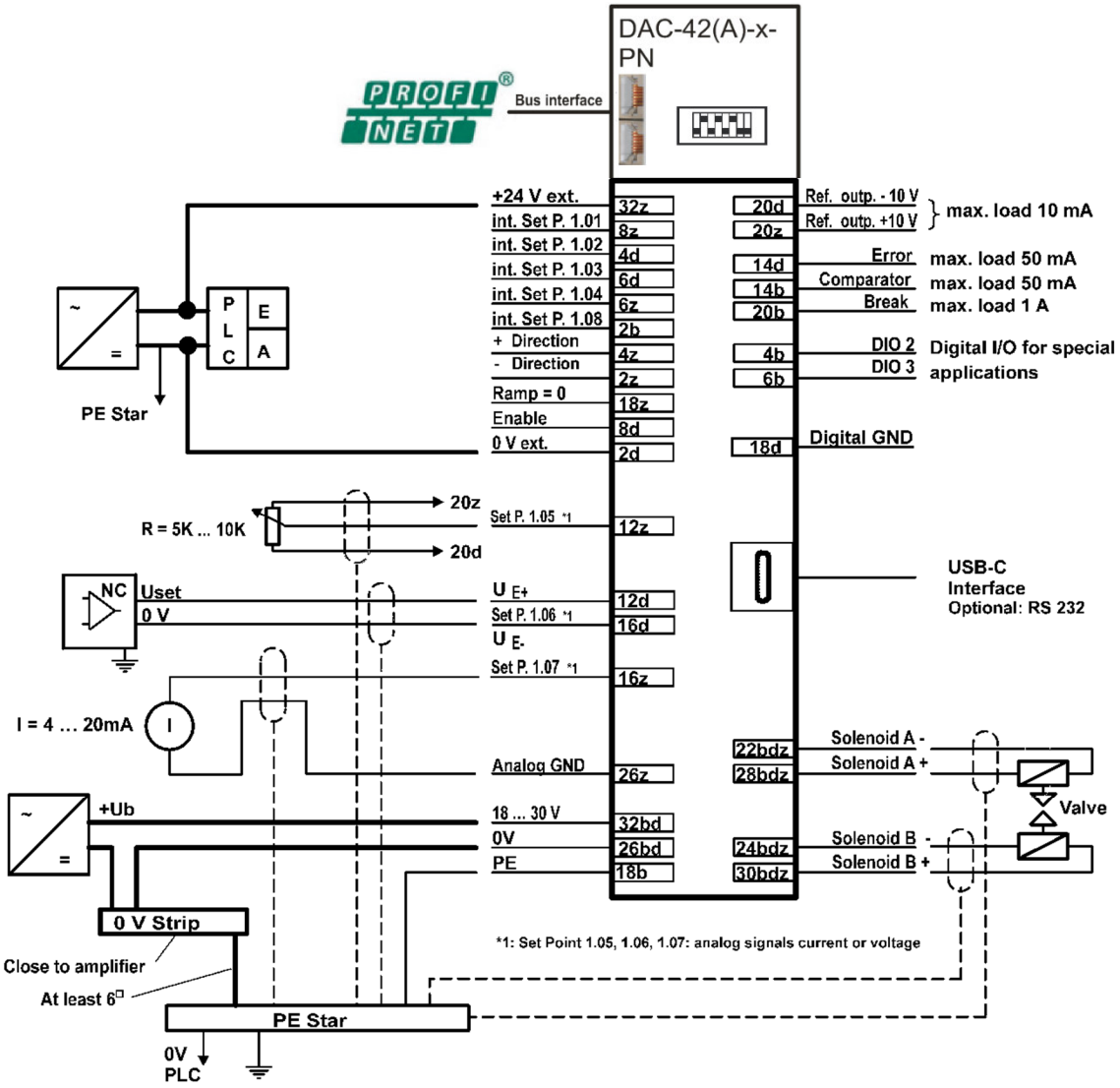



Figure 9 : Wiring diagram Mode 01

### 5.6.2 Wiring diagram for operation mode 02

Open loop, 2 proportional valves with 1 solenoid each

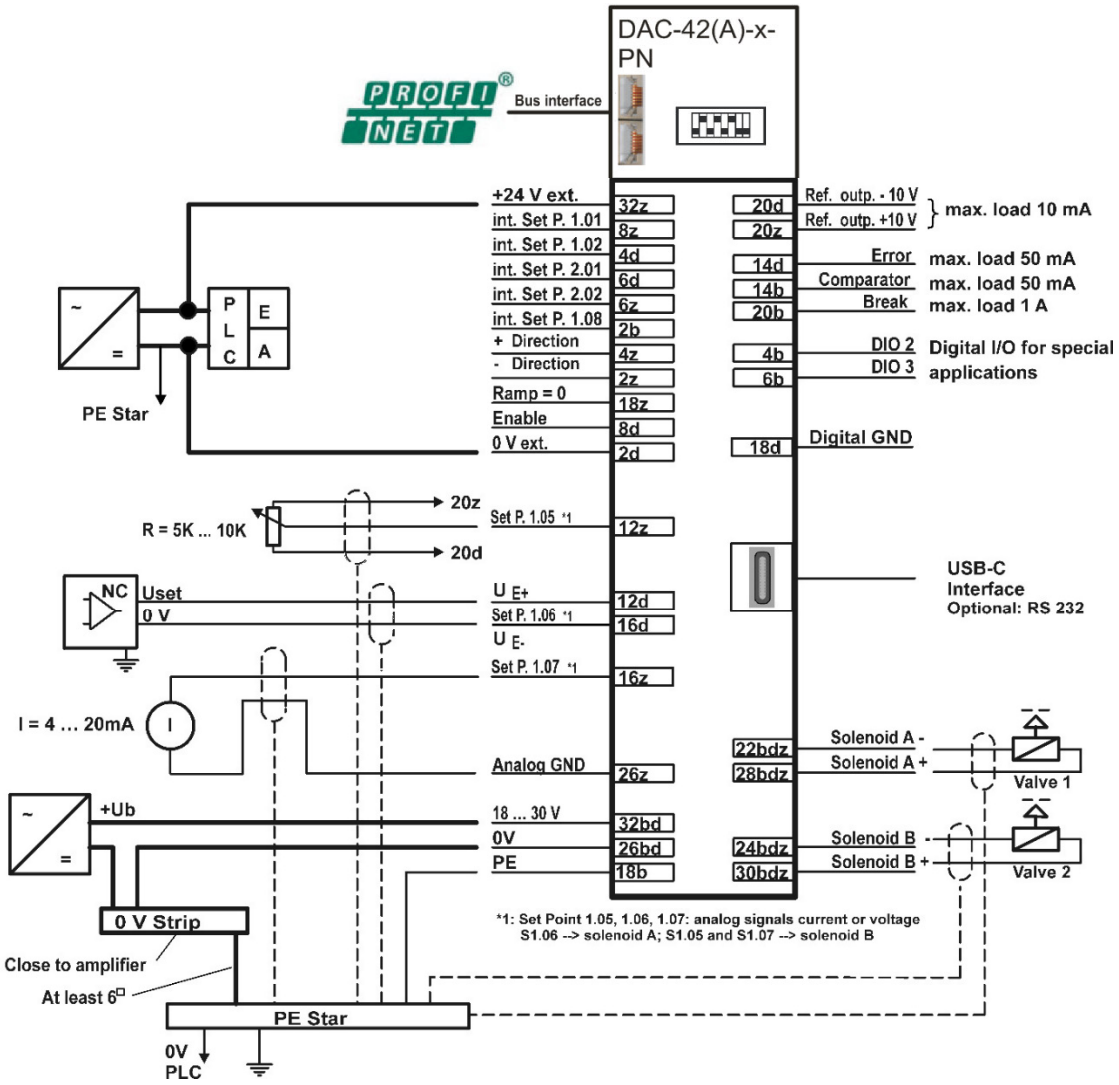



Figure 10 : Wiring diagram Mode 02

### 5.6.3 Wiring diagram for operation mode 03

Closed loop, 1 proportional valve with 2 solenoids and spool position feedback

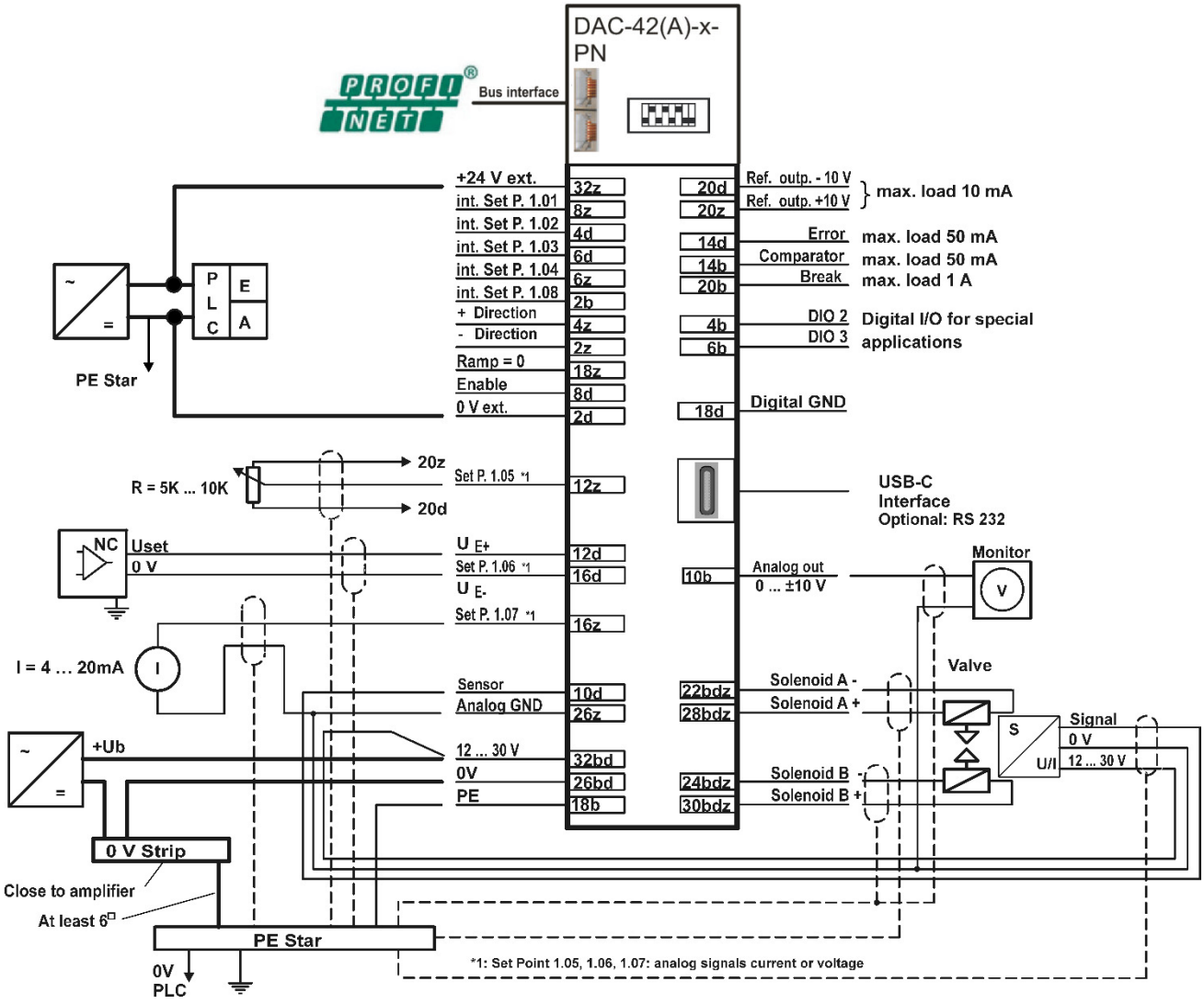



Figure 11 : Wiring diagram Mode 03

### 5.6.4 Wiring diagram for operation mode 04

Closed loop double, 1 proportional valve with 2 solenoids and feedback of a process value

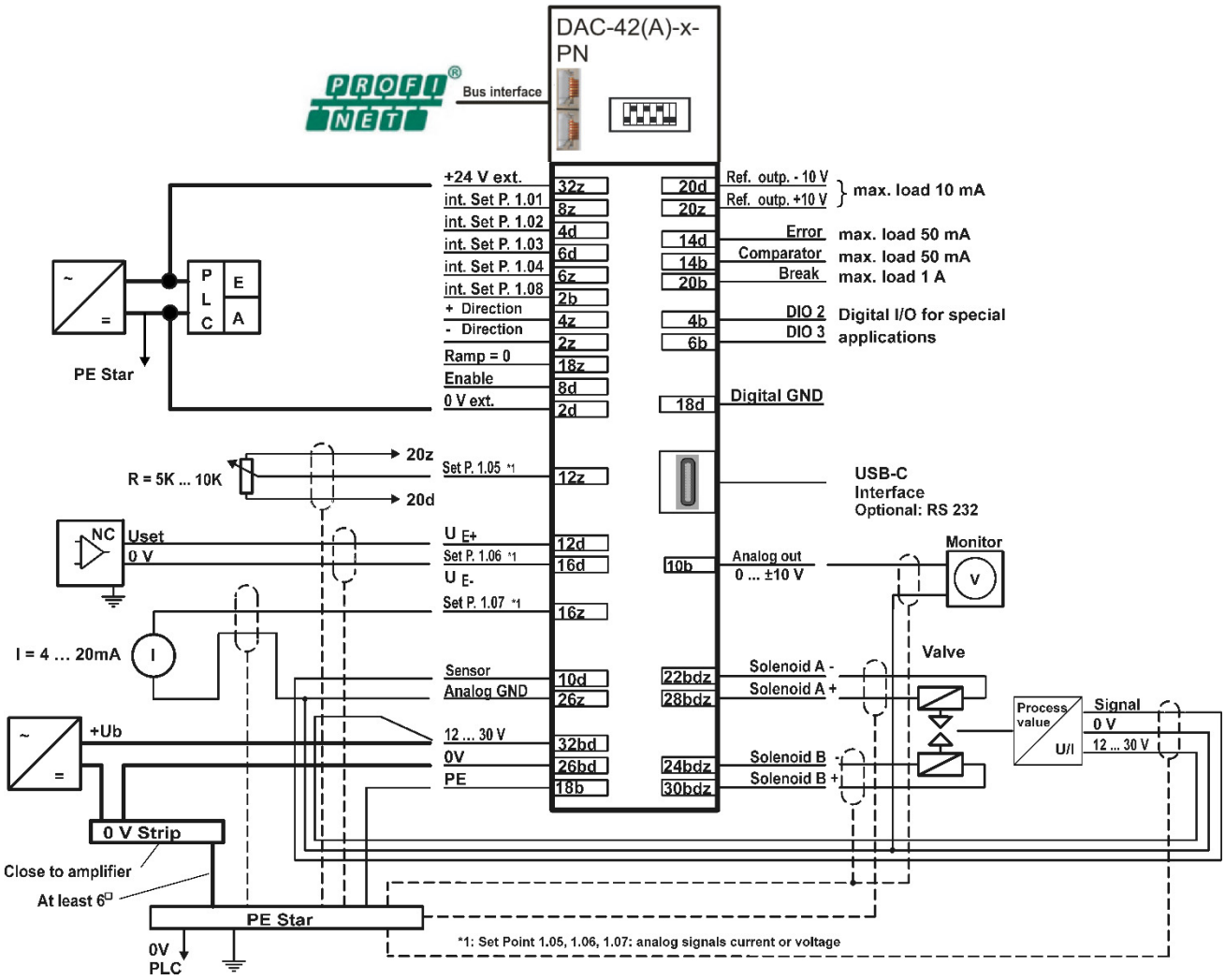



Figure 12 : Wiring diagram Mode 04

### 5.6.5 Wiring diagram for operation mode 06

Closed loop double, 1 proportional valve with 2 solenoids and spool position and process value feedback

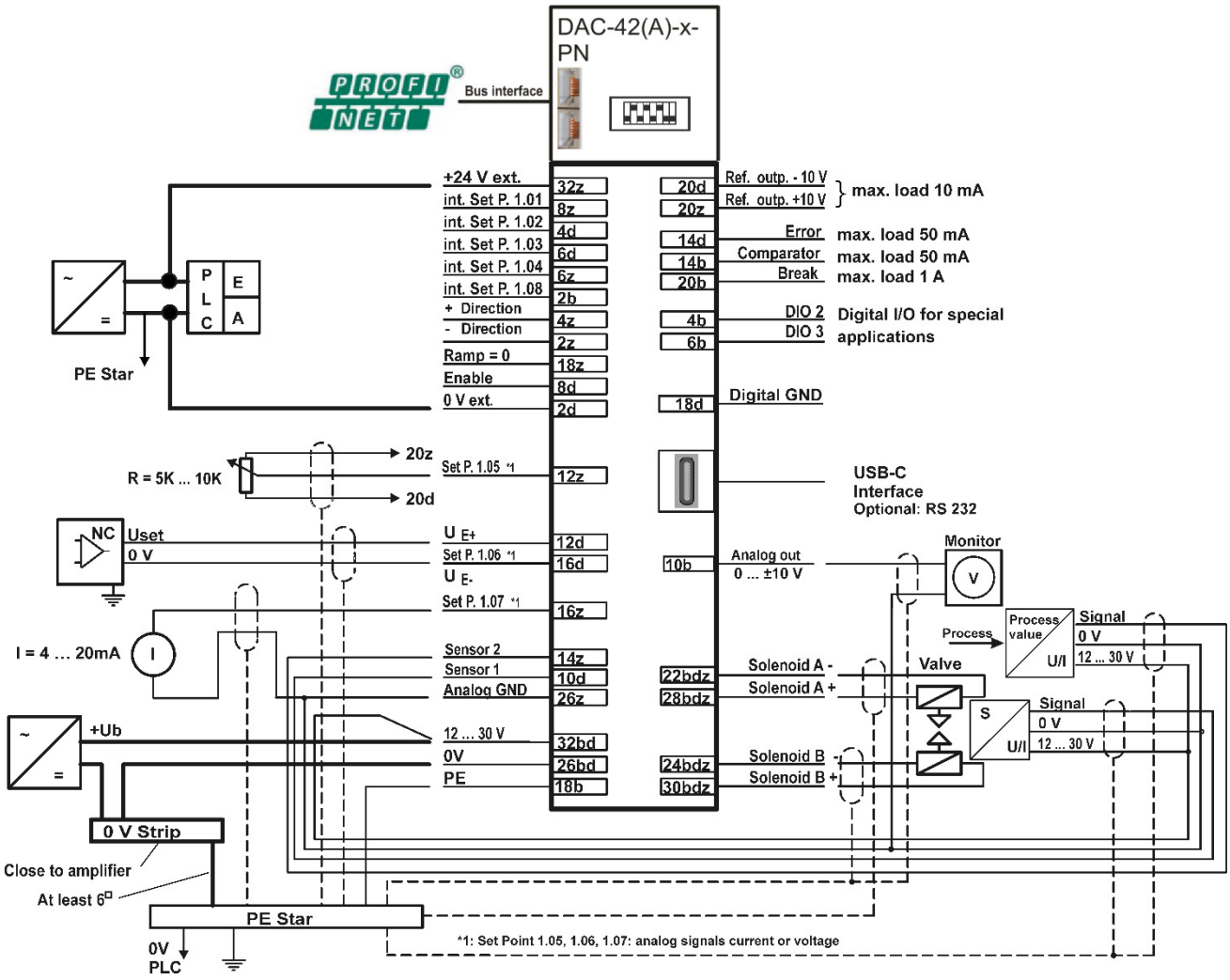



Figure 13 : Wiring diagram Mode 06

### 5.6.6 Wiring diagram for operation mode 08

Closed loop double, 2 proportional valves with 1 solenoid each and feedback of two independent process values

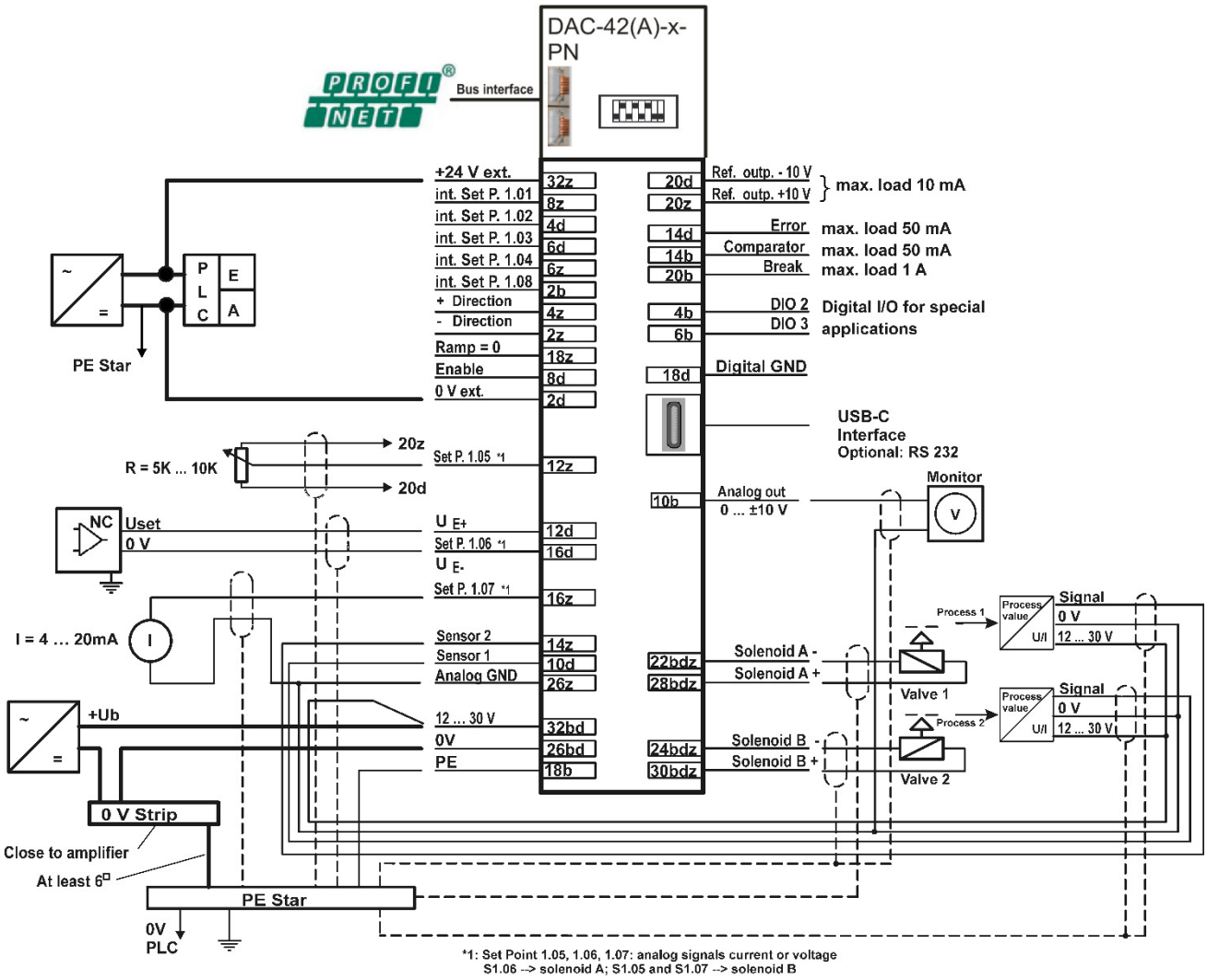



Figure 14 : Wiring diagram Mode 08

### 5.6.7 Wiring diagram for operation mode 10

Closed loop without valve, controller only

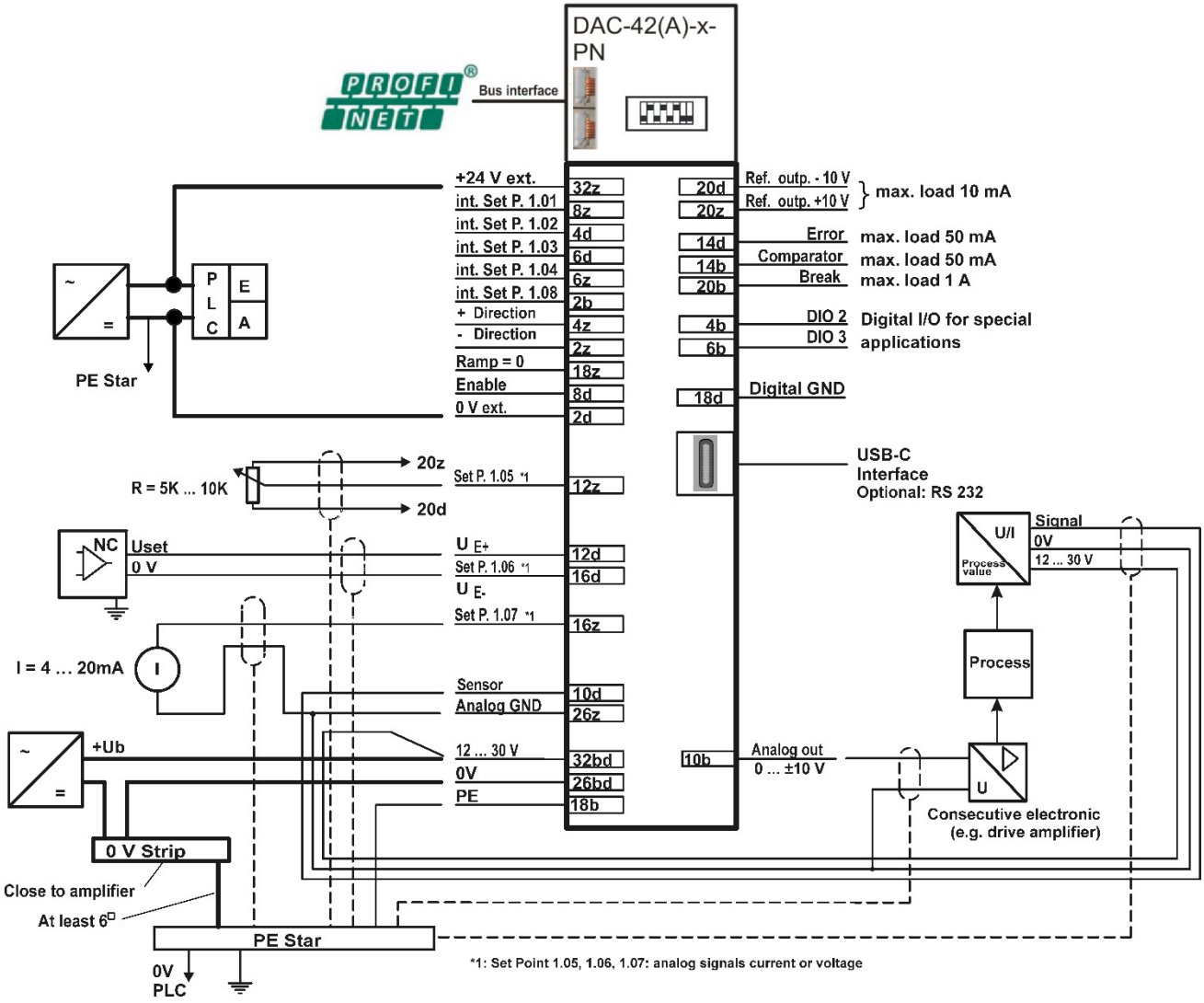



Figure 15 : Wiring diagram Mode 10

### 5.6.8 Wiring diagram for operation mode 11

Closed loop double (cascaded controller) without valve, controller only

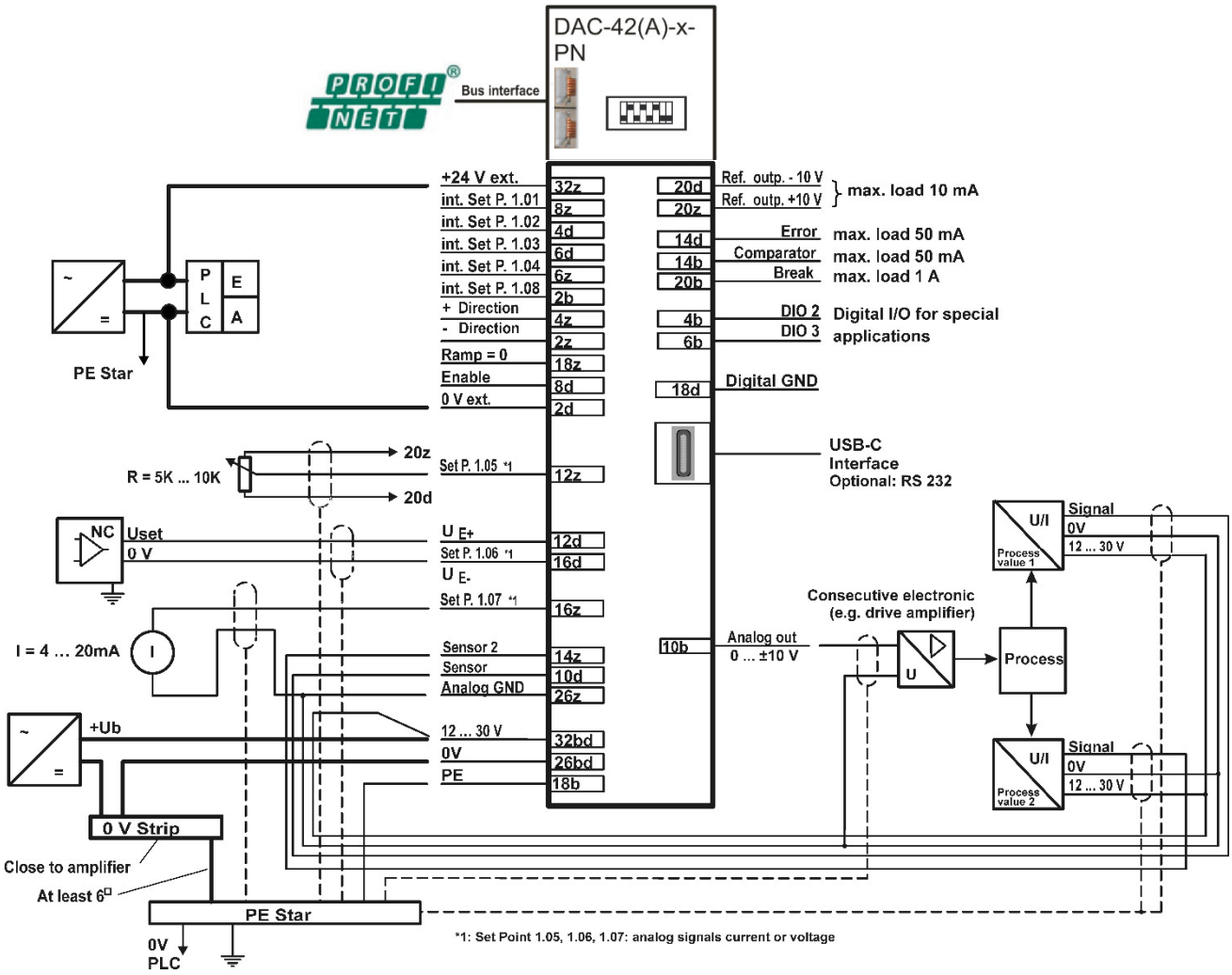



Figure 16 : Wiring diagram Mode 11

# 6 Commissioning

## 6.1 Front elements

**⚠ WARNING**

**The electrical wiring must be checked before switching on the supply voltage. Limit switches and safety devices must be activated to avoid uncontrolled movements. Carefully follow relevant safety regulations. Suitable emergency stop measures must be taken.**

### 6.1.1 DAC-42(A)-x-PN

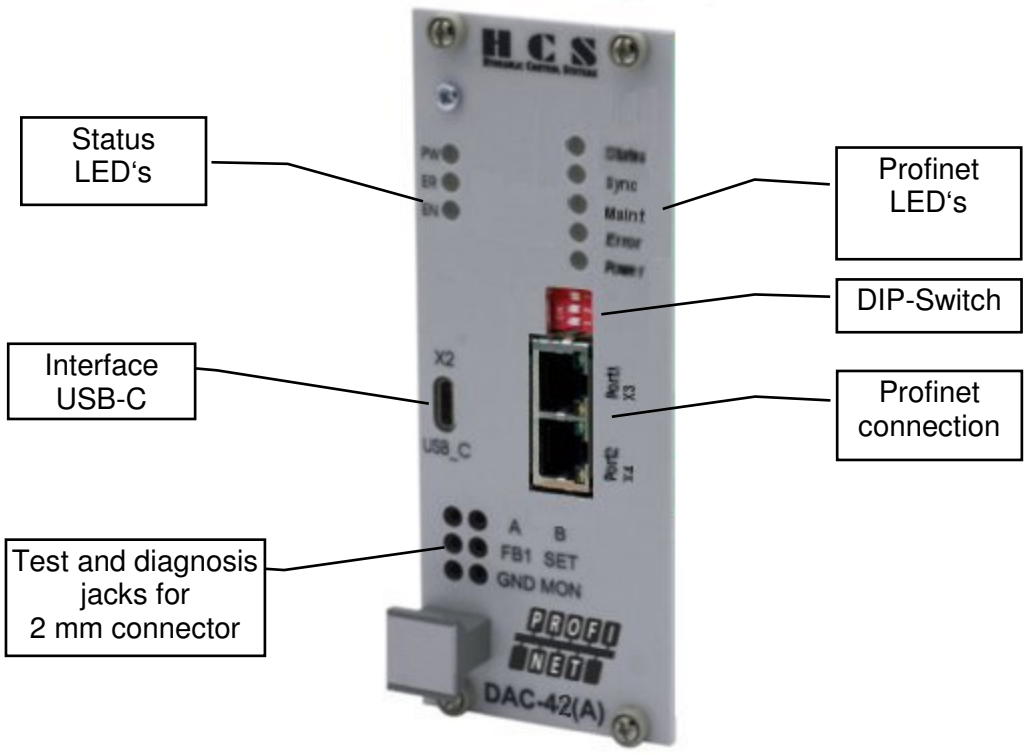



Figure 17 : DAC-42(A)-x-PN Front elements

| Element | Function |
|---|---|
| Status LED’s | Display of status, Power, Enable, Error. |
| Profinet LED’s | Display of status of the profinet connection. |
| Profinet LED’s | Display of status of the profinet connection. |
| Profinet connection’s | Ethernet 2xRJ45 input/Output to the network |
| DIP-switch | Reserved, for further use |
| USB-C interface | Trough which programing and accessing parameters via PC and HCSTool is been executed. RS232 (optional) |
| Measuring and test jacks | Direct measurement of set point, actual value, solenoid currents and internal values via the monitor output. Use 2 mm sockets (S1.06, FB1, A, B, d1.01 ... d2.13) |

Table 9: Front elements DAC-42(A)-x-PN explanation

## 6.2 Software block diagrams

**All functions and parameter inside of the block diagrams are described in detail in the "DAC-4x(A) Manual E Rxx yyyymmdd.pdf" please ask HCS GmbH if needed.**

### 6.2.1 Operation Mode 01; 1 valve with 2 solenoids, open loop

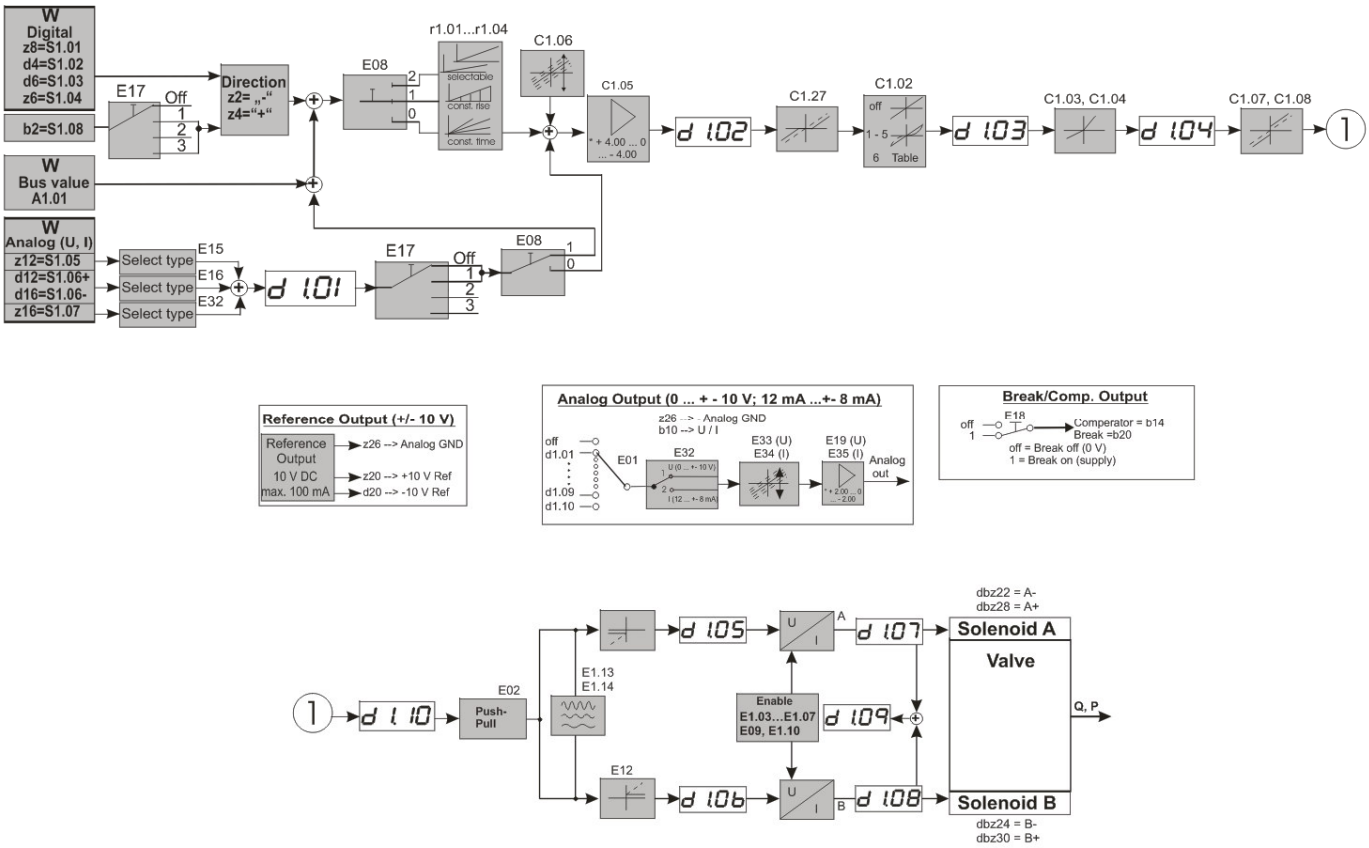



Figure 18 : Software block diagram operation mode 1

### 6.2.2 Operation Mode 02; 2 valves with 1 solenoid each, open loop

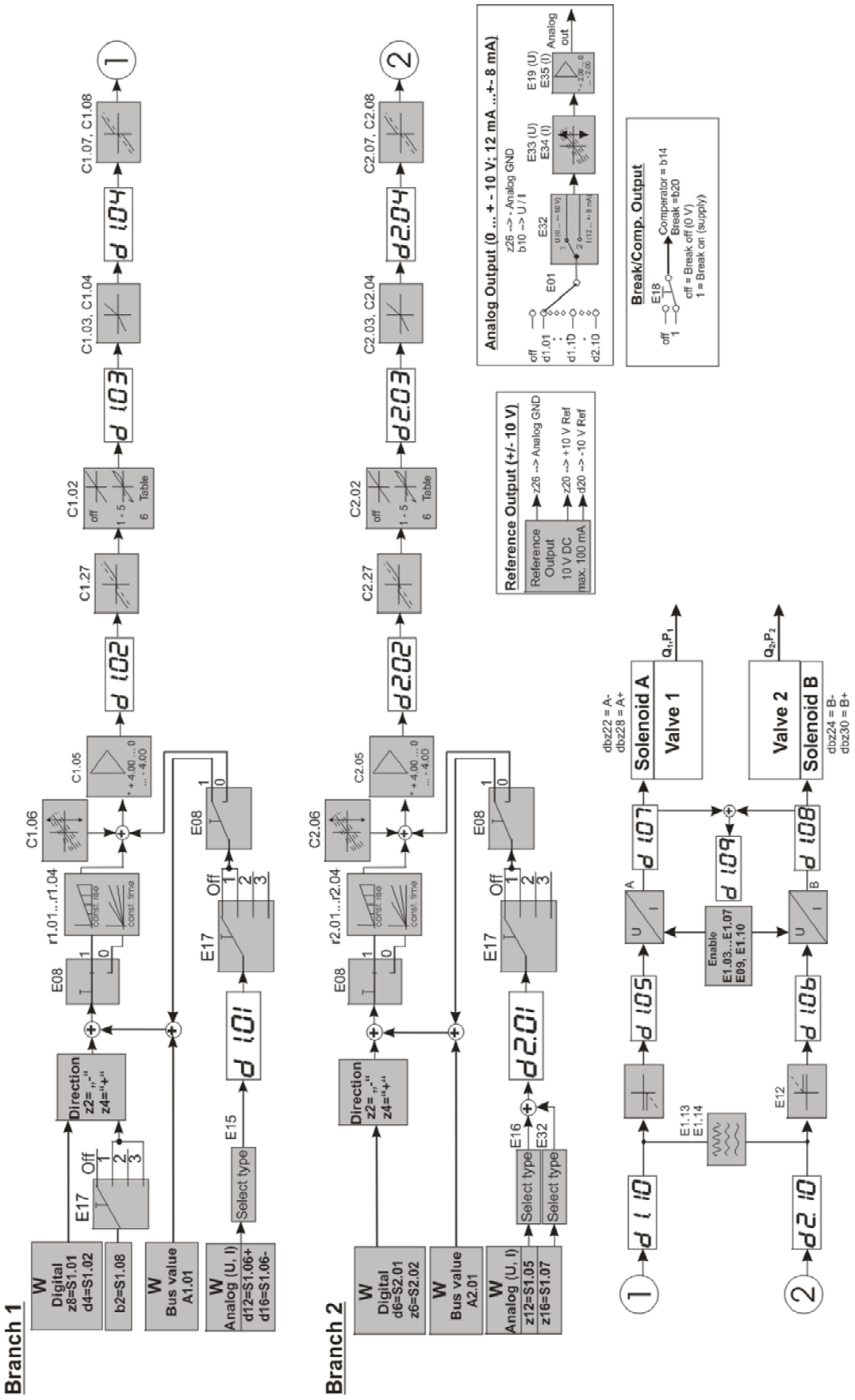



Figure 19 : Software block diagram operation mode 2

### 6.2.3 Operation Mode 03; 1 valve with 2 solenoids and spool position feedback

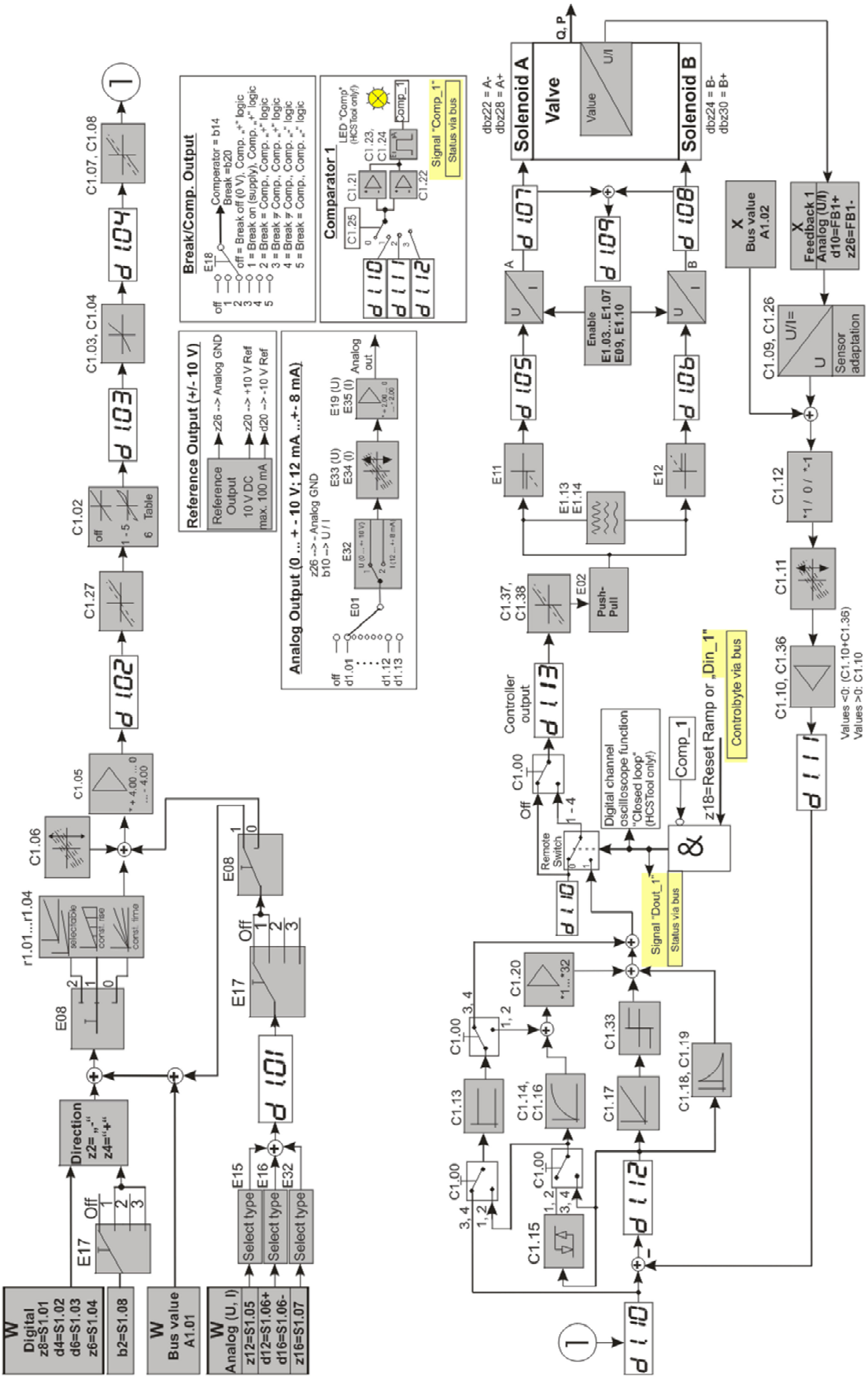



Figure 20 : Software block diagram operation mode 3

### 6.2.4 Operation Mode 04; 1 valve with 2 solenoids and process value feedback

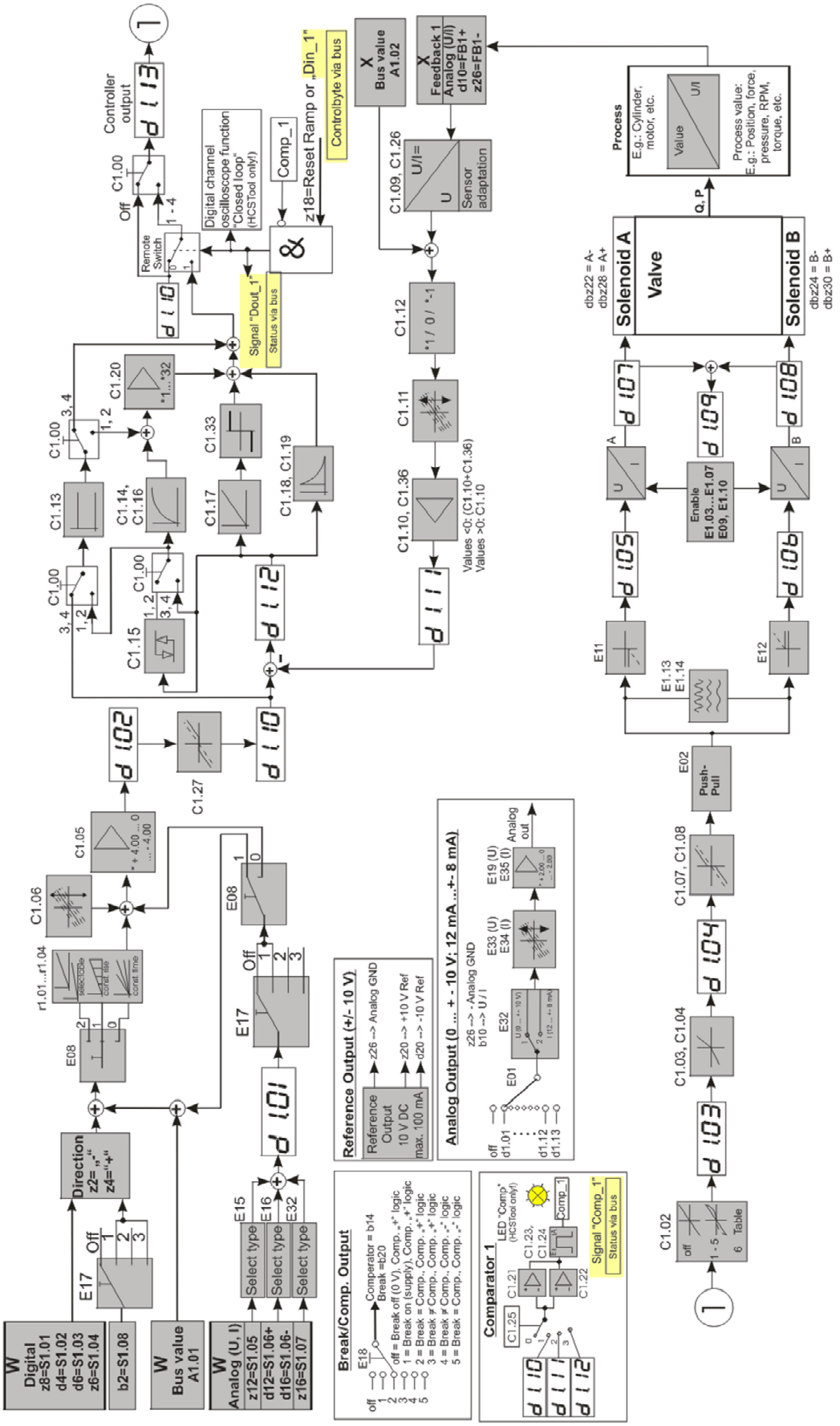



Figure 21 : Software block diagram operation mode 4

### 6.2.5 Operation Mode 06; 1 valve with 2 solenoids spool position feedback and process value feedback

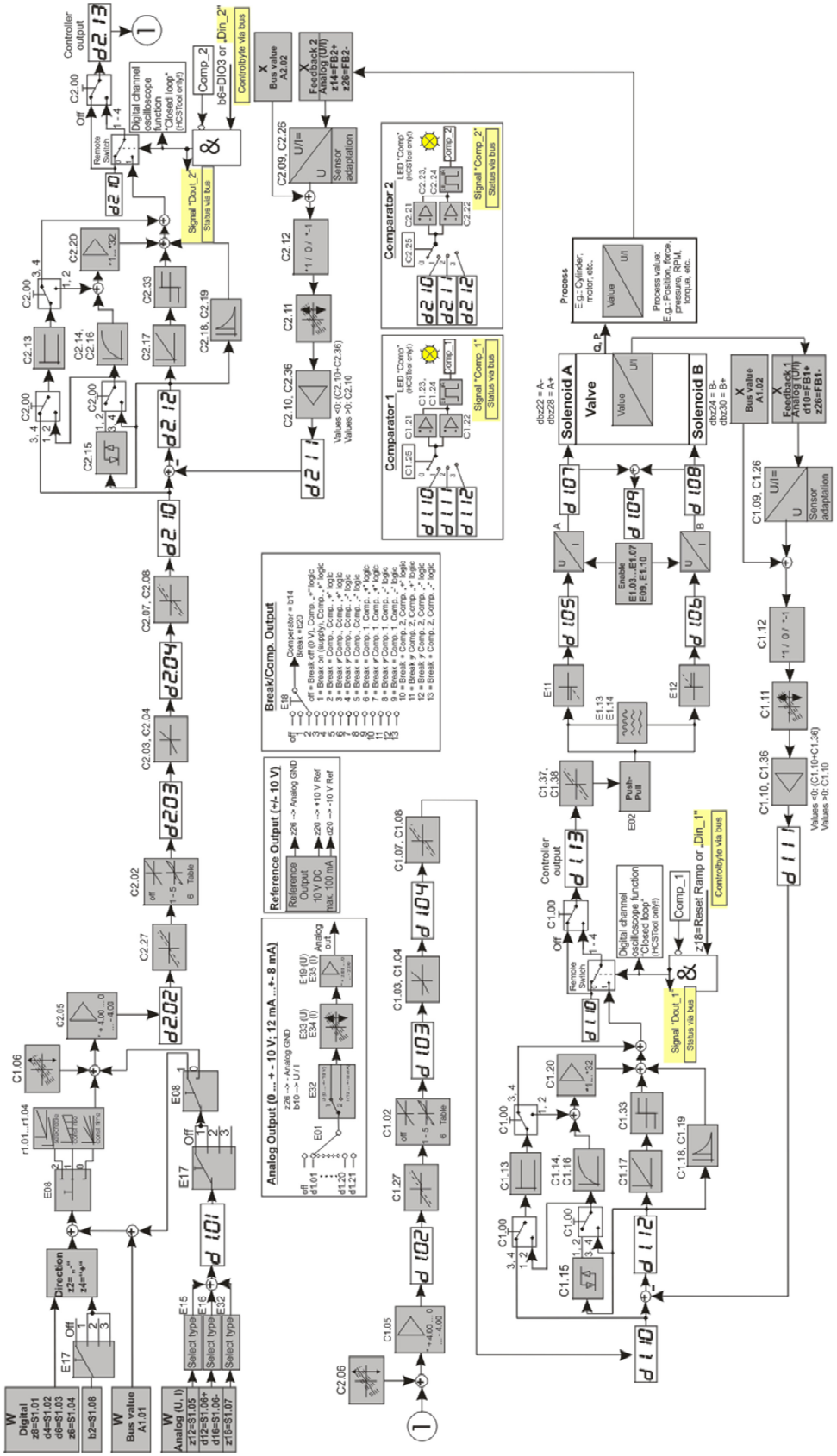



Figure 22 : Software block diagram operation mode 6

### 6.2.6 Operation Mode 08; 2 valves with 1 solenoid and process value feedback

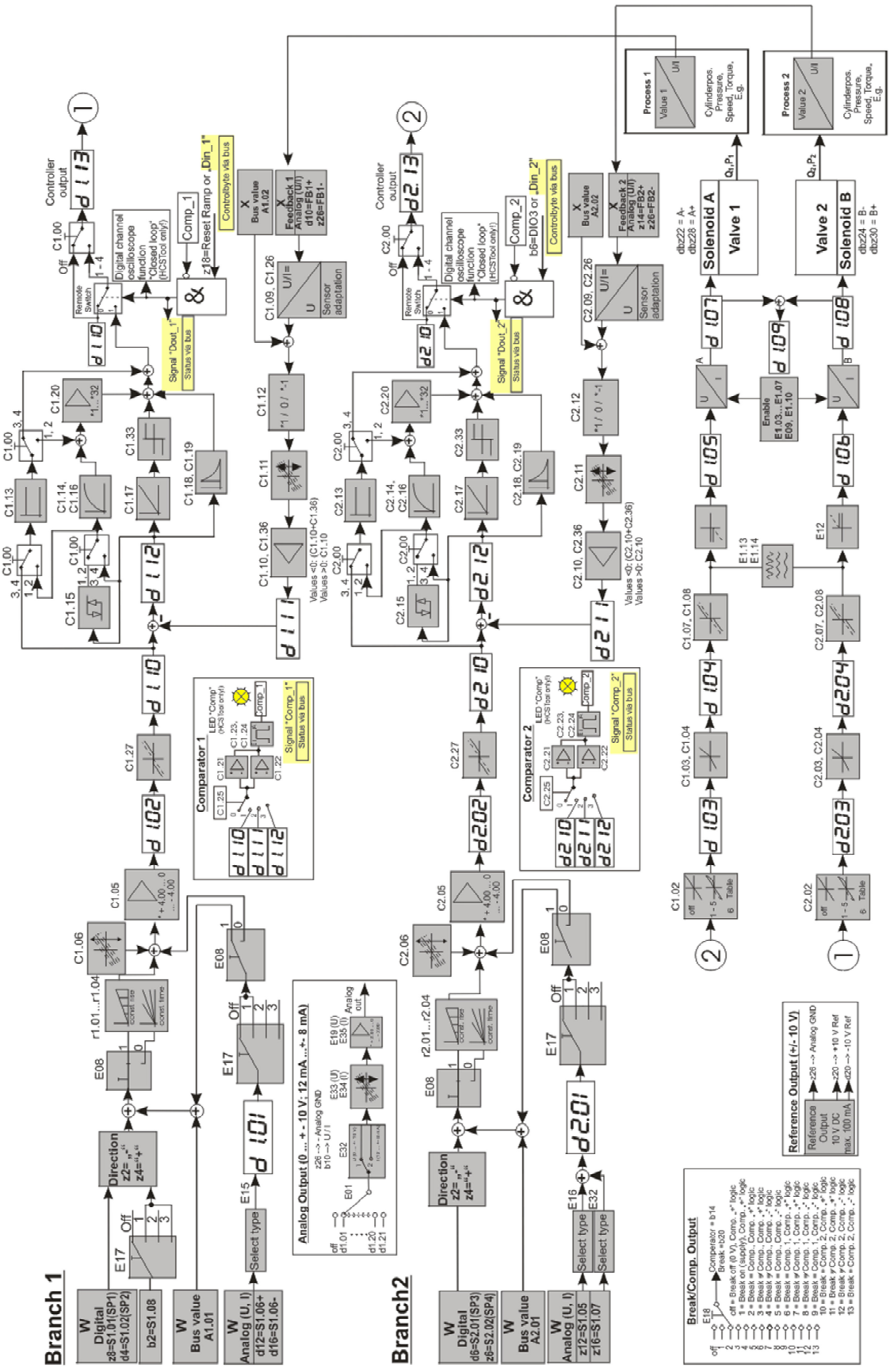



Figure 23 : Software block diagram operation mode 8

### 6.2.7 Operation Mode 10; stand-alone controller with process feedback

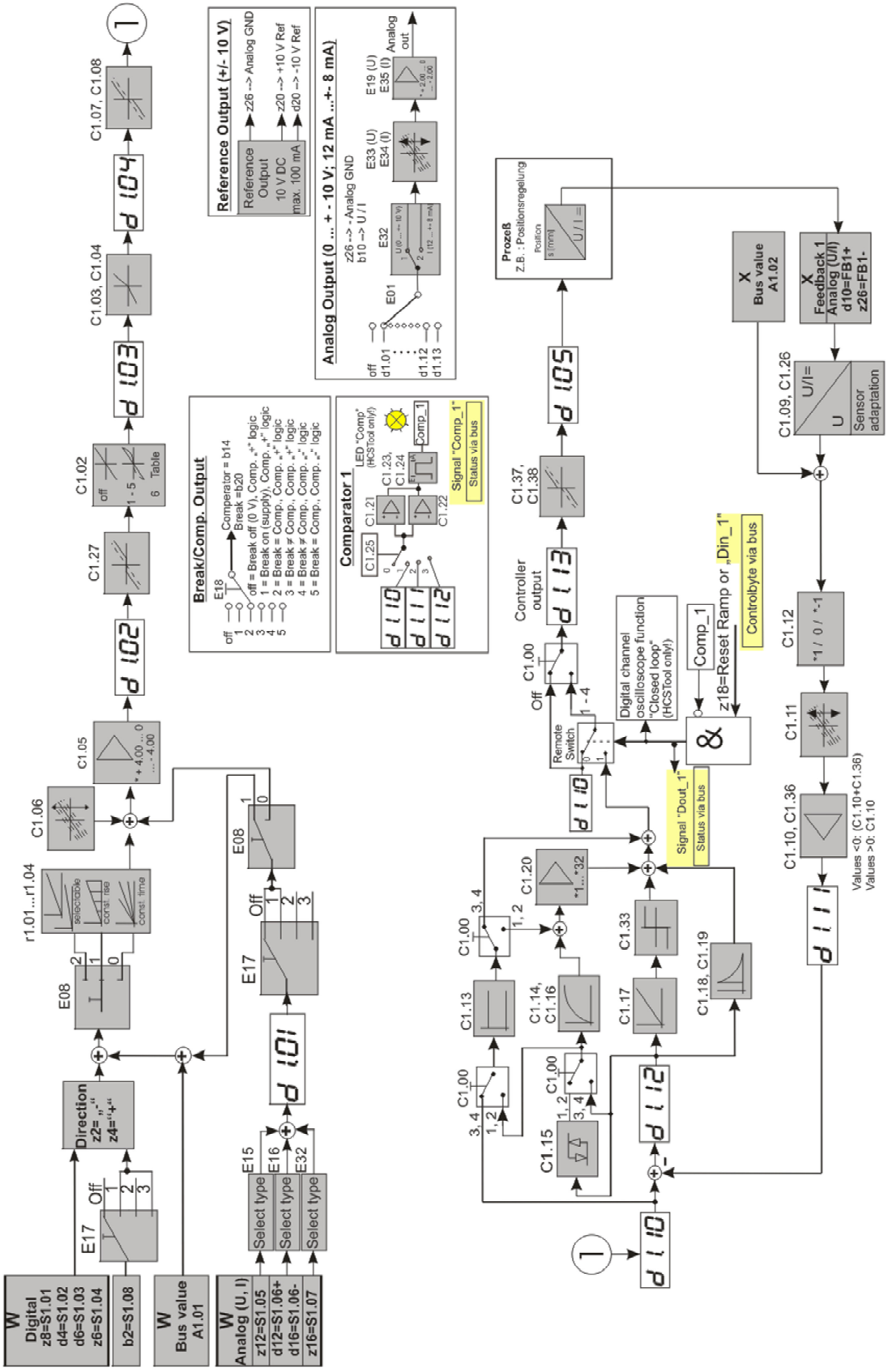



Figure 24 : Software block diagram operation mode 10

### 6.2.8 Operation Mode 11; stand-alone controller with two process feedbacks

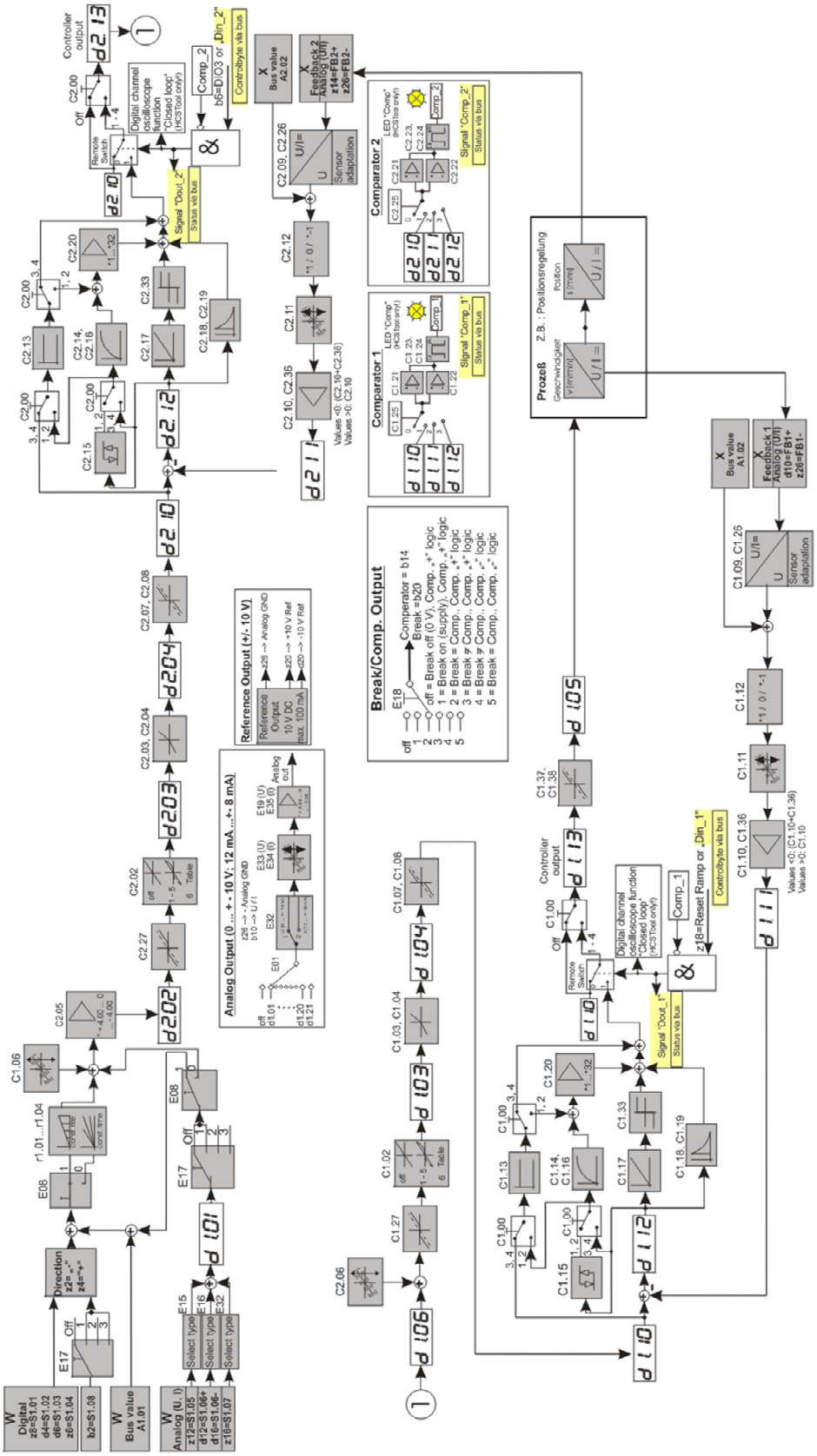



Figure 25 : Software block diagram operation mode 11

# 7 General communication information

## 7.1 Special information for bus activation

If communication via Profinet is required, there must be a connection to the power supply; supply via USB is not sufficient. The Profinet-Node connected to the PLC (Error LED is off). In addition, parameter E22 must be set to "on" or “1”.

## 7.2 General information about TADR (Telegram Address)

TADR is a value which may be written (changed) with each cycle but changing of this value is optional and not mandatory. The value will be defined by the Profinet-Master.
The purpose is to either check the telegrams by the master or also in order to force sending of a new telegram from the node.
The design of the node is made in a way that the slave or slaves (module or modules in case of multi-slave version) are only responding if a received message from the master contains changed (different) data compared to the previous message.
So TADR can be used to force a response from a slave for example in order to get an update on the current status of the slave even if the other data in the message remain unchanged.
TADR value is valid in the range of 0 to 255 or in hexadecimal 0x00 to 0xFF.

## 7.3 General information about CMD (Command) and SADR (Slave Address)

**The SADR is used to address different slaves which are connected to the same Profinet-node. In case of the DAC-4x(A) the possibility of a Multi-Slave Nodes is not possible. The slave Address in the telegram must always set to “0” or “1”**

Different ways of communication between master and slaves (modules) are possible. The difference is in the CMD instruction.

CMD = 6: Writing of a single parameter-ID to the DAC-4x(A). The Profinet node has only one slave (one DAC-4x(A) connected. In this case SADR is set to 1.
□ Chapter „8.3.1 CMD = 6, Master Write single parameter (7 bytes) ”, page 44

CMD = 3: Reading of one or more parameters with parameter-ID in rising order. The Profinet node has only one slave (one DAC-4x(A) connected. In this case SADR is set to 1
□ Chapter „8.1.1 CMD = 3, Master Read parameters (7 bytes)”, page 42

CMD = 15: This instruction allows writing or reading of a (pre-defined) set of parameters. The set of parameters itself is depending on the software version of the DAC-4x(A); refer also to
□ Chapter „9.5 CMD = 15, structure of Profinet telegram definition, depending on SW Versions“, page 49
For more information please to □ Chapter „9 Complex commands”, page 47

It is possible to „mix“ the instructions CMD = 3, CMD = 6 and CM = 15 for the DAC-4x(A).

# 8 Simple commands

## 8.1 CMD 3 Master Read

### 8.1.1 CMD = 3, Master Read parameters (7 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the user, Range 0..255, 0..0xFF |
| 1 | SADR = 01 | Slave address |
| 2 | CMD | Command: 3 = Read multiple parameters by the master |
| 3 | IDH | High byte parameter ID of first parameter |
| 4 | IDL | Low byte parameter ID of first parameter |
| 5 | N-high | High-Byte Number parameters |
| 6 | N-low | Low-Byte Number parameters |

…Write last

**TADR** Telegram address, defined by the Profinet-Master.( Set by the User )

**SADR** 1 = standard (single slave)

**CMD** Command: 3 = Read parameter by the master

**IDH,IDL** H-Byte and Low-Byte of parameter-ID in HEX
0x0000 .. 0x0109 = Valid ID-Range. Please take into account the gaps in the parameter list

**N-high** High-Byte Number parameters (words), normally zero

**N-low** Low-Byte Number parameters (words), 1 to 8 (0x08)
(maximal 8 parameters readable at once)

### 8.1.2 CMD = 3, Response from Slave (4 + CNT bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the read command, Range 0..255, 0..0xFF |
| 1 | SADR | Slave address |
| 2 | CMD | Command: 3 = Read multiple parameters by the master |
| 3 | CNT | N*2, Number of data bytes |
| 4 | DAT1H | High byte value of first parameter |
| 5 | DAT1L | Low Byte value of first parameter |

If CNT > 2: ……

| | | |
|---|---|---|
| CNT+2 = N | DATnH | High byte value of last (n)parameter |
| CNT+3 = N | DATnL | Low Byte value of last (n)parameter |

…Write last

**TADR** Telegram address, defined by the Profinet-Master. (Set by the read command)

**SADR** Slave address, 1 = standard (single slave)

**CMD** Command: 3 = Read parameter by the master

**CNT** N*2, Number of data bytes (Maximal 16 data bytes)

**IDH,IDL** H-Byte and Low-Byte of parameter-ID in HEX
0x0000 .. 0x00109 = Valid ID-Range

**DAT1H.. DAT nH**,
**DAT1L.. DAT nL**, Two data bytes in HEX, without decimal sign
0x7FFF == +32767
0x8000 == -32768

### 8.1.3 CMD = 3, Error from DAC-4x(A) (4 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the read command, Range 0..255, 0..0xFF |
| 1 | SADR | Slave address |
| 2 | ERR | Error code |
| 3 | EXCE | Exception |

…Write last

**ERR** Error code: 0x83 = Read failed

**EXCE** Exception:
1 = Command not supported
2 = Wrong ID
3 = CNT== 0 or CNT > 16 (0x10)
4 = Reading of ID failed
5 = timeout slave, address wrong or not installed or
Invalid number of connected slaves
6 = Internal checksum error (Modbus)

Example for error response:

Module 3 is not installed and **SADR** in the master telegram is set to 3

| Byte | Data | Abbreviation | Description |
|---|---|---|---|
| 0 | 0x23 | TADR | Response telegram address (defined by the read command) |
| 1 | **0x01** | SADR | Actual slave address |
| 2 | 0x83 | ERR | Error, Bit „7“ with command 0x03 is set |
| 3 | 0x05 | EXCE | timeout slave, address wrong or not installed |

## 8.2 Telegram example CMD = 3

**Master wants to read parameter „d1.07”:**

ID = 0x0007
N = 0x01, (one parameter)
TADR = 0x24
SADR = 0x01 (always 1)

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x24 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x03 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x07 | IDL |
| 5 | 0x00 | N-high |
| 6 | 0x01 | N-low |

Response from slave (module)
CNT = 2 (2 Bytes)
DATA = 0x0133 (== 0.307A)

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x24 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x03 | CMD |
| 3 | 0x02 | CNT |
| 4 | 0x01 | DAT1H |
| 5 | 0x33 | DAT1L |

## 8.3 CMD = 6, Master Write

### 8.3.1 CMD = 6, Master Write single parameter (7 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the user, Range 0..255, 0..0xFF |
| 1 | SADR = 01 | Slave address |
| 2 | CMD | Command: 6 = Write single parameter by the master |
| 3 | IDH | High byte parameter ID |
| 4 | IDL | Low byte parameter ID |
| 5 | DATH | High byte of parameter value |
| 6 | DATL | Low byte of parameter value |

…Write last

**TADR** Telegram address, defined by the Profinet-Master.(Set by the user)

**SADR** 1 = standard (single slave)

**CMD** Command: 6 = Write single parameter by the master

**IDH,IDL** H-Byte and Low-Byte of parameter-ID in HEX
0x0000 .. 0x00B0 = Valid ID-Range

**DATH,DATL** Two data bytes in HEX, without decimal sign
0x7FFF == +32767
0x8000 == -32768

### 8.3.2 CMD = 6, Response from DAC-4x(A) (7 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the write command, Range 0..255, 0..0xFF |
| 1 | SADR | Slave address |
| 2 | CMD | Command: 6 = Write single parameter by the master |
| 3 | IDH | High byte parameter ID |
| 4 | IDL | Low byte parameter ID |
| 5 | DATH | High byte of second analogue value |
| 6 | DATL | Low byte of first analogue value |

…Write last

**TADR** Telegram address, defined by the Profinet-Master.( Set by the write command )

**SADR** Slave address, 1 = standard (single slave)

**CMD** Command: 6 = Write parameter by the master

**IDH,IDL** H-Byte and Low-Byte of parameter-ID in HEX
0x0000 .. 0x0109 Valid ID-Range

**DATH,DATL** Two data bytes in HEX, without decimal sign
0x7FFF == +32767
0x8000 == -32768

### 8.3.3 CMD = 6, Error from DAC-4x(A) (4 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the write command, Range 0..255, 0..0xFF |
| 1 | SADR | Slave address |
| 2 | ERR | Error code |
| 3 | EXCE | Exception |

…Write last

**ERR** Error code: 0x86 = write failed

**EXCE** Exception:
1 = Command not supported
2 = Wrong ID
3 = Wrong Data Value
4 = Writing of ID failed
5 = timeout slave, address wrong or not installed or
Invalid number of connected slaves
6 = Internal checksum error (Modbus)

Example for error response:

SADR 3 is not valid and **SADR** in the master telegram is set to 3

| Byte | Data | Abbreviation | Description |
|---|---|---|---|
| 0 | 0x23 | TADR | Response telegram address (defined by the write command) |
| 1 | **0x03** | SADR | Actual slave address |
| 2 | 0x86 | ERR | Error, Bit „7“ with command 0x06 is set |
| 3 | 0x05 | EXCE | timeout slave, address wrong or not installed |

## 8.4 Telegram example CMD = 6

### 8.4.1 Example, write parameter

**Master wants to write parameter „C1.07” with value 1.000 V:**

ID = 0x0030
DATA = 0x3E8 ( = 1000 in decimal)
TADR = 0x23
SADR = 0x01

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x23 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x30 | IDL |
| 5 | 0x03 | DATL |
| 6 | 0xE8 | DATH |

Response from slave (module), command was successful executed
ID = 0x0030
DATA = 0x3E8 ( = 1.000 V)

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x23 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x30 | IDL |
| 5 | 0x03 | DATL |
| 6 | 0xE8 | DATH |

### *8.4.2 Example, write set value*

**Examples for set value via Profinet (ID = 0x0027):**

1.) Command signal 0.000 V = 0x0000:

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x12 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x27 | IDL |
| 5 | 0x00 | DATL |
| 6 | 0x00 | DATH |

2.) Set value 5.000 V = 0x1388:

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x12 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x27 | IDL |
| 5 | 0x13 | DATL |
| 6 | 0x88 | DATH |

3.) Set value 9.999 V = 0x270F:

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x12 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x27 | IDL |
| 5 | 0x27 | DATL |
| 6 | 0x0F | DATH |

4.) Set value –9.999 V = 0xD8F1

| Byte | Data | Abbreviation |
|---|---|---|
| 0 | 0x12 | TADR |
| 1 | 0x01 | SADR |
| 2 | 0x06 | CMD |
| 3 | 0x00 | IDH |
| 4 | 0x27 | IDL |
| 5 | 0xD8 | DATL |
| 6 | 0xF1 | DATH |

# 9 Complex commands

## 9.1 CMD = 15, Master writes multiple parameters (8 bytes)

### (Fast single-slave command)

Write first…

| Byte | Abbreviation | Description | |
|---|---|---|---|
| 0 | TADR | Telegram address, defined by the user, Range 0..255, 0..0xFF | |
| 1 | SNUM = 01 | Single slave, always 01 | |
| 2 | CMD | Command: 15 = Write multiple parameter by the master | |
| 3 | DAT1.1 | Control-Byte | Byte for special functions |
| 4 | DAT1.2H | Process Value1 | High byte of first „analogue" value |
| 5 | DAT1.2L | | Low byte of first „analogue" value |
| 6 | DAT1.3H | Process Value2 | High byte of second „analogue" value |
| 7 | DAT1.3L | | Low byte of second „analogue" value |
| | Overall telegram length in bytes | | |

**TADR** Telegram address, defined by the Profinet-Master. (defined by the user) Range 0..255

**SNUM** Only one slave is connected. SNUM = 01.

**CMD** Command: 15 = Write multiple parameters by the master

**DAT1.1** Control-Byte. One data byte in HEX, for special functions
0x00 == 0
0xFF == 255

**DAT1.2H, DAT1.2L** Process Value1: Two data bytes in HEX, without decimal sign, for „analogue" values
0x7FFF == +32767
0x8000 == -32768

**DAT1.3H, DAT1.3L** Process Value1: Two data bytes in HEX, without decimal sign, for „analogue" values
0x7FFF == +32767
0x8000 == -32768

## 9.2 CMD = 15, Response from DAC-4x(A) (9 bytes)

This response telegram with SADR = 1 is send back by the DAC-4x(A).

Write first…

| Byte | Structure | Abbreviation | Description | |
|---|---|---|---|---|
| 0 | Telegram header | TADR | Telegram address, defined by the user, Range 0..255, 0..0xFF | |
| 1 | | SADR = 01 | Slave address | |
| 2 | | CMD | Command: 15 = Write multiple parameter by the master | |
| 3 | Data of SADR | DAT.1H | Status Word | High byte of module state |
| 4 | | DAT.1L | | Low byte of module state |
| 5 | | DAT.2H | VALUE1 | High byte of first „analogue" value |
| 6 | | DAT.2L | | Low byte of first „analogue" value |
| 7 | | DAT.3H | VALUE2 | High byte of second „analogue" value |
| 8 | | DAT.3L | | Low byte of second „analogue" value |

…Write last

**TADR** Telegram address, defined by the Profinet-Master.(set by the user)

**SADR** Slave address, normally always 01

**CMD** Command: 15 = Write multiple parameters by the master

**DAT.1H,DAT.1L** Status word. Two data bytes in HEX, , for module state
0x0000 == 0
0xFFFF == 65535

**DAT.2H, DAT.2L, DAT.3H, DAT.3L** VALUE1
VALUE2. Two data bytes in HEX, without decimal sign, for analogue values
0x7FFF == +32767
0x8000 == -32768

## 9.3 CMD = 15, Error from DAC-4x(A) (4 bytes)

Write first…

| Byte | Abbreviation | Description |
|---|---|---|
| 0 | TADR | Telegram address, defined by the write command, Range 0..255, 0..0xFF |
| 1 | SADR | Slave address |
| 2 | ERR | Error code |
| 3 | EXCE | Exception |

…Write last

**ERR** Error code: 0x8F = write failed

**EXCE** Exception:
1 = Command not supported
2 = Wrong ID
3 = Wrong Data Value
4 = Writing of ID failed
5 = timeout (Modbus)
6 = Internal checksum error (Modbus)
7 = Invalid number of connected slaves
8 = timeout slave, address wrong or not installed
9 = Profinet node (SADR = 0) send, received data length to small. The data length of the selected GSD module is too small.

Example for error response:

Module 3 is not installed and SNUM in the master telegram is set to 3

| Byte | Data | Abbreviation | Description |
|---|---|---|---|
| 0 | 0x23 | TADR | Response telegram address (defined by the write command) |
| 1 | **0x03** | SADR | Actual slave address |
| 2 | 0x8F | ERR | Error, Bit „7“ with command 0x0F in decimal 15 is set |
| 3 | 0x08 | EXCE | timeout slave, address wrong or not installed |

## 9.4 Examples, CMD = 15, single slave, DAC-4x(A)

E22 is set to 1 and hardware Enable is connected to the DAC-4x(A).

**Master writes set value of 1.000 V to slave (module):**

| Byte | Data | Abbreviation | Description |
|---|---|---|---|
| 0 | 0x23 | TADR | Telegram address, defined by the user, Range 0..255, 0..0xFF |
| 1 | 0x01 | SNUM | 0x01 means single slave |
| 2 | 0x0F | CMD | 0x0F in decimal 15 |
| 3 | 0x00 | DAT1.1 | **Control byte** , 0x00 (default value, no bus-disable) |
| 4 | (1000) | DAT1.2H | **Process Value1 = A1.01, set value branch 1** |
| 5 | | DAT1.2L | 0x03E8 ( = 1000 in decimal = 1.000V) |
| 6 | (0) | DAT1.3H | **Process Value2 = A1.02, feedback value branch 1** |
| 7 | | DAT1.3L | 0x0000 ( = 0 in decimal = 0.000V) |

**Response slave (module) no.1:**

| Byte | Data | | Abbreviation | Description | |
|---|---|---|---|---|---|
| 0 | 0x23 | | TADR | Response telegram address (defined by the write command) | |
| 1 | 0x01 | | SADR | Actual slave address | |
| 2 | 0x0F | | CMD | 0x0F in decimal 15 | |
| 3 | 0x40 | | DAT1.1H | **Status word** | High byte of module state = 0x40 (HW enable = active) |
| 4 | 0x00 | | DAT1.1L | | Low byte of module state = 0x00 |
| 5 | 0x00 | (0) | DAT1.2H | **VALUE1** = d1.11 | |
| 6 | 0x00 | | DAT1.2L | 0x0000 ( = 0 in decimal = 0.000V ) | |
| 7 | 0x03 | (1000) | DAT1.3H | **VALUE2** = d1.10 | |
| 8 | 0xE8 | | DAT1.3L | 0x03E8 ( = 1000 in decimal = 1.000V) | |

(x) = value in decimal.

## 9.5 CMD = 15, structure of Profinet telegram definition, depending on SW Versions

### 9.5.1 General

#### 9.5.1.1 Error Message (indication by „Error occurred")

Errors which are indicated in the Bit (Error occurred) like for example „Over Current" or „Cable Fraction" can be reset with the following action :

- Set input signal for the hardware enable at the according module at terminal X3/1 (9) to „Low" and then go back to „High"
- Setting and resetting of the Bits (0x80) Bus-Disable

Mandatory condition for resetting an error is that the failure causing the error has been cleared or removed.

**Error numbers:**
The corresponding error numbers can be found in the respective manual of the DAC-4x(A).
Generally, the following error table is valid:

| **Slave (Module) Status** | | |
|---|---|---|
| Hi-Byte (DAT.1H) | Low-Byte (DAT.1L) | Description |
| 0x0C | 0x01 | Error of Operation |
| 0x0C | 0x02 | Wire break at FB1 (voltage/current depend on setting) |
| 0x0C | 0x03 | Excess current or short circuit at the output stages (peak current >8A!) |
| 0x0C | 0x04 | Wire break at FB2 (voltage/current depend on setting) |
| 0x0C | 0x05 | Wire break at S1.05 (voltage/current depend on setting) |
| 0x0C | 0x06 | Wire break at S1.06 (voltage/current depend on setting) |
| 0x0C | 0x07 | Wire break at S1.07 (voltage/current depend on setting) |
| 0x0C | 0x08 | Underload of the solenoid output (if monitoring is activated) |
| 0x0C | 0x0A | Timeout error -no Profinet command recognized- (defined with parameter E23) |
| 0x0C | 0x0B | Over current at feedback FB1 (current signal >22mA) |
| 0x0C | 0x0C | Over current at feedback FB2 (current signal >22mA) |
| 0x0C | 0x0F | Over current at set point S1.05 (current signal >22mA) |
| 0x0C | 0x10 | Over current at set point S1.06 (current signal >22mA) |
| 0x0C | 0x11 | Over current at set point S1.07 (current signal >22mA) |
| 0x0C | 0x17 | Selected Sensor need autocalibration. |

Table 10: Error numbers

If parameter E 23 has a value not equal to 0 (Profinet timeout is activated). Depending on parameter E 26, there are different responses from the device.

- E 26 = off (0)
  The device reacts with a blinking enable LED about to long time between valid Profinet telegrams. Those versions continue with work if a new valid telegram is detected. No separate acknowledges with the enable signal is needed.
- E 26 = 1
  The device reacts with a shining Error LED about to long time between valid Profinet telegrams. Those versions do not continue with work if a new valid telegram is detected. A separate acknowledge with the enable signal is needed.

Generally, the electronic discards every set point in error case and the outputs will be disabled.

#### 9.5.1.2 Explanation for „Hardware Enable" and „Software Disable":

Generally speaking, the hardware enable must be present („high" level) in order to get a signal (current) at the output stages. Besides the hardware enable a software disable can be used in order to „override" the hardware enable. So even if the hardware signal is present, with the according software command the module can be disabled!

This means that the hardware enable can be hardwired and the software disable can be used in order to drop the output signal to zero or also in order to quit errors.

### 9.5.1.3 Explanation of special functions and module state bits

Attention: Functions and module state bits depend on the used SW version

| | |
|---|---|
| **HW_ENABLE** | The „Enable" signal (HW Enable pin 8d) is activated at the module. |
| **ERROR** | An Error has occurred in the DAC-4x(A). Different possible problems can cause this. Please refer to the manual for the according DAC-4x(A) version. |
| **BUS_DISABLE** | Enable signal was reset by means of Profinet (DAC-4x(A) is disabled).<br>Switching off the HW enable also resets any pending **BUS_DISABLE** |
| **Din_1** | To enable the function for the „remote loop controller" (function switches from open loop to closed loop) via the comparator „Comp_1" is enabled and can be used. Additional mandatory condition: C1.00 must be set to 2 or 4!<br>Remark: d1.12 and d1.13 remain at „0.000 V" as long as the condition for activation „Din_1" has not be set. This will prevent a premature activation of the comparator „Comp_1". |
| **Dout_1/CL** | Indicates that the function „loop controller" for the branch 1 is active. When the module is working in „remote loop", the module is now working in closed loop mode. Additional mandatory conditions for transition from open loop to closed loop:<br>• C1.00 must be set to 2 or 4<br>• Signal Din_1 must be present<br>• Comparator Comp_1 has to be outside of the window, hence signal Comp_1 = 0.<br>The module will remain in closed loop operation as long as signal Din_1 stays on „High". |
| **Comp_1** | Output signal of comparator „Comp_1", s used in order to monitor the signal defined by C1.25.<br>This signal can also be used in order to switches over from open to closed loop:<br>0 = comparator is not active (out of the determined limits and time delays)<br>1 = comparator is active (inside the determined limits and time delays). |
| **Comp_11** | Output signal of comparator „Comp_11", is used in order to monitor the signal defined by C1.32<br>0 = comparator is not active (out of the determined limits and time delays)<br>1 = comparator is active (inside the determined limits and time delays). |
| **Din_2** | To enable the function for the „remote loop controller" (function switches from open loop to closed loop) via the comparator „Comp_2" is enabled and can be used. Additional mandatory condition: C2.00 must be set to 2 or 4!<br>Remark: d2.12 and d2.13 remain at „0.000 V" as long as the condition for activation „Din_2" has not be set. This will prevent a premature activation of the comparator „Comp_2". |
| **Dout_2** | Indicates that the function „loop controller" for the branch 2 is active. When the module is working in „remote loop", the module is now working in closed loop mode. Additional mandatory conditions for transition from open loop to closed loop:<br>• C2.00 must be set to 2 or 4<br>• Signal Din_2 must be present<br>• Comparator Comp_2 has to be outside of the window, hence signal Comp_2 = 0.<br>The module will remain in closed loop operation as long as signal Din_2 stays on „High". |
| **Comp_2** | Output signal of comparator „Comp_2", s used in order to monitor the signal defined by C2.25.<br>This signal can also be used in order to switches over from open to closed loop:<br>0 = comparator is not active (out of the determined limits and time delays)<br>1 = comparator is active (inside the determined limits and time delays). |
| **Comp_22** | Output signal of comparator „Comp_22", is used in order to monitor the signal defined by C2.32<br>0 = comparator is not active (out of the determined limits and time delays)<br>1 = comparator is active (inside the determined limits and time delays). |
| **TST_CMP** | This bit can be used to test the Set Error/Comp output. When set, the output is set to 24V, independent of the Comp_1/Comp_2 signal. To activate this function the parameter E 18 must be set to a value > 0 and no error is present. |
| **CNTRL_4** | Reserved for special function |
| **STAT_1..5** | Reserved for special functions |

### 9.5.2 Version for Mode 1 (open loop, one valve with two solenoids)

Master writes multiple parameters (at all 8 bytes)

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by User, Range 0..255, 0..0xFF |
| | 1 | | SNUM = 1 | Single slave = 1 |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Module Data | DAT1.1 | **Control byte** \| Byte for special functions |
| | 4 | | DAT1.2H | **A1.01 Module 1** |
| | 5 | | DAT1.2L | Set value input (Profinet) |
| | 6 | | DAT1.3H | **A1.02 Module 1 (normally not used)** |
| | 7 | | DAT1.3L | Set value input (Profinet) |
| ….Write last | | 8 | | Overall telegram length in bytes |

Explanation of Byte for special functions (**Control byte**)

| Description | Abbreviation | Function of each bit |
|---|---|---|
| Byte for special functions | DATx.1 | Can be set only if the hardware enable is active.<br>0x01 = **not used**<br>0x04 = **TST_CMP**; Set Error/Comp output direct to 24V (disable signalizing of error output)<br>0x10 = **not used**<br>0x80 = **BUS_DISABLE** (Module disable via Profinet)<br>In this mode not applicable, reserved:<br>0x02 = CNTRL_1<br>0x08 = CNTRL_2<br>0x20 = CNTRL_3<br>0x40 = CNTRL_4 |
| | | The special function byte will be set to internal 0x00 (reset) when the hardware enable is inactive. |

**Response telegram:**

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by the write command |
| | 1 | | SADR | Slave address |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Data of SADR Module | DAT.1H | **Status word** \| High byte of module state |
| | 4 | | DAT.1L | Low byte of module state |
| | 5 | | DAT.2H | **VALUE1** = d1.07 |
| | 6 | | DAT.2L | Actual current A |
| | 7 | | DAT.3H | **VALUE2**= d1.08 |
| | 8 | | DAT.3L | Actual current B |
| ….Write last | | 9 | | Overall telegram length in bytes |

Explanation of High and low byte of module state (**Status word**):

| Description | Abbreviation | Explanation of each bit |
|---|---|---|
| High byte of module state | DAT.1H | 0x01 = Digital input **SP1** (S1.01) active *)<br>0x02 = Digital input **SP2** (S1.02) active *)<br>0x04 = **HW_ENABLE**, hardware enable active<br>0x08 = **ERROR**, Error is pending<br>0x10 = Digital input **SP3** (S1.03 active *)<br>0x20 = Digital input **SP4** (S1.04 active *)<br>0x80 = **BUS_DISABLE** is set<br>In this mode not applicable, reserved:<br>0x40 = STAT_5 |
| Low byte of module state | DAT.1L | if „Error occurred" bit is set: error number<br>otherwise:<br>0x01 = **not used**<br>0x02 = **not used**<br>0x04 = **not used**<br>0x08 = **not used**<br>0x10 = **not used**<br>0x20 = **not used**<br>0x40 = **not used**<br>0x80 = **not used** |

*) The designation and availability of the input depends on the module variant.

### 9.5.3 Version for Mode 2 (open loop, two valves with one solenoid each)

Master writes multiple parameters (at all 8 bytes)

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by User, Range 0..255, 0..0xFF |
| | 1 | | SNUM = 1 | Single slave = 1 |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Module Data | DAT1.1 | **Control byte** \| Byte for special functions |
| | 4 | | DAT1.2H | **A1.01,** set value input (Profinet) for (Branch 1) |
| | 5 | | DAT1.2L | |
| | 6 | | DAT1.3H | **A2.01,** set value input (Profinet) for (Branch 2) |
| | 7 | | DAT1.3L | |
| ….Write last | | 8 | Overall telegram length in bytes | |

Explanation of Byte for special functions (**Control byte**)

| Description | Abbreviation | Function of each bit |
|---|---|---|
| Byte for special functions | DATx.1 | Can be set only if the hardware enable is active.<br>0x01 = not used<br>0x04 = **TST_CMP**; Set Error/Comp output direct to 24V (disable signalizing of error output)<br>0x10 = not used<br>0x80 = **BUS_DISABLE** (Module disable via Profinet)<br>In this mode not applicable, reserved:<br>0x02 = CNTRL_1<br>0x08 = CNTRL_2<br>0x20 = CNTRL_3<br>0x40 = CNTRL_4 |
| | | The special function byte will be set to internal 0x00 (reset) when the hardware enable is inactive. |

**Response telegram:**

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by the write command |
| | 1 | | SADR | Slave address |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Data of SADR Module | DAT.1H | **Status word** \| High byte of module state |
| | 4 | | DAT.1L | Low byte of module state |
| | 5 | | DAT.2H | **VALUE1** = d1.07 |
| | 6 | | DAT.2L | Actual current A |
| | 7 | | DAT.3H | **VALUE2** = d1.08 |
| | 8 | | DAT.3L | Actual current B |
| ….Write last | | 9 | Overall telegram length in bytes | |

Explanation of High and low byte of module state (**Status word**):

| Description | Abbreviation | Explanation of each bit |
|---|---|---|
| High byte of module state | DAT.1H | 0x01 = Digital input **SP1** (S1.01) active *)<br>0x02 = Digital input **SP2** (S1.02) active *)<br>0x04 = **HW_ENABLE**, hardware enable active<br>0x08 = **ERROR**, Error is pending<br>0x10 = Digital input **SP3** (S2.01) active *)<br>0x20 = Digital input **SP4** (S2.02) active *)<br>0x80 = **BUS_DISABLE** is set<br>In this mode not applicable, reserved:<br>0x40 = STAT_5 |
| Low byte of module state | DAT.1L | if „Error occurred" bit is set: error number<br>otherwise:<br>0x01 = not used<br>0x02 = not used<br>0x04 = not used<br>0x08 = not used<br>0x10 = not used<br>0x20 = not used<br>0x40 = not used<br>0x80 = not used |

*) The designation and availability of the input depends on the module variant.

### 9.5.4 Version for Mode 3, 4, 10 (closed loop)

Master writes multiple parameters (at all 8 bytes)

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by User, Range 0..255, 0..0xFF |
| | 1 | | SNUM = 1 | Single slave = 1 |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Module Data | DAT1.1 | **Control byte** \| Byte for special functions |
| | 4 | | DAT1.2H | **Process Value1 = A1.01** |
| | 5 | | DAT1.2L | Set value input (Profinet) for (Branch 1) |
| | 6 | | DAT1.3H | **Process Value2 = A1.02** |
| | 7 | | DAT1.3L | Feedback value input (Profinet) for (Branch 1) |
| ….Write last | | 8 | | Overall telegram length in bytes |

Explanation of Byte for special functions (**Control byte**)

| Description | Abbreviation | Function of each bit |
|---|---|---|
| Byte for special functions | DATx.1 | Can be set only if the hardware enable is active.<br>0x01 = **Din_1**; Set signal Din_1 to active<br>0x04 = **TST_CMP**; Set Error/Comp output direct to 24V (disable signalizing of error output)<br>0x10 = **Din_2**; Set signal Din_2 to active<br>0x80 = **BUS_DISABLE** (Module disable via Profinet)<br>In this mode not applicable, reserved:<br>0x02 = CNTRL_1<br>0x08 = CNTRL_2<br>0x20 = CNTRL_3<br>0x40 = CNTRL_4 |
| | | The special function byte will be set to internal 0x00 (reset) when the hardware enable is inactive. |

**Response telegram:**

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by the write command |
| | 1 | | SADR | Slave address |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Data of SADR Module | DAT.1H | **Status word** \| High byte of module state |
| | 4 | | DAT.1L | Low byte of module state |
| | 5 | | DAT.2H | **VALUE1** = d1.11 |
| | 6 | | DAT.2L | Actual value, feedback value (Branch 1) |
| | 7 | | DAT.3H | **VALUE2** = d1.10 |
| | 8 | | DAT.3L | Set value (internal value) (Branch 1) |
| ….Write last | | 9 | | Overall telegram length in bytes |

Explanation of High and low byte of module state (**Status word**):

| Description | Abbreviation | Explanation of each bit |
|---|---|---|
| High byte of module state | DAT.1H | 0x01 = Digital input **SP1** (S1.01) active *)<br>0x02 = Digital input **SP2** (S1.02) active *)<br>0x04 = **HW_ENABLE**, hardware enable active<br>0x08 = **ERROR**, Error is pending<br>0x10 = Digital input **SP3** (S2.01) active *)<br>0x20 = Digital input **SP4** (S2.02) active *)<br>0x80 = **BUS_DISABLE** is set<br>In this mode not applicable, reserved:<br>0x40 = STAT_5 |
| Low byte of module state | DAT.1L | if „Error occurred" bit is set: error number<br>otherwise:<br>0x01 = **Din_1** active<br>0x02 = **Dout_1** active<br>0x04 = **Comp_1** active<br>0x08 = **Comp_11** active<br>0x10 = **Din_2** active<br>0x20 = **Dout_2** active<br>0x40 = **Comp_2** active<br>0x80 = **Comp_22** active |

*) The designation and availability of the input depends on the module variant.

### 9.5.5 Version for Mode 6, 11 (two closed loops)

Master writes multiple parameters (at all 8 bytes)

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by User, Range 0..255, 0..0xFF |
| | 1 | | SNUM = 1 | Single slave = 1 |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Module Data | DAT1.1 | **Control byte** \| Byte for special functions |
| | 4 | | DAT1.2H | **Process Value1 = A1.01** |
| | 5 | | DAT1.2L | Set value input (Profinet) for (Branch 1) |
| | 6 | | DAT1.3H | **Process Value2 = A2.02** |
| | 7 | | DAT1.3L | Feedback value input (Profinet) for (Branch 2) |
| ….Write last | | 8 | | Overall telegram length in bytes |

Explanation of Byte for special functions (**Control byte**)

| Description | Abbreviation | Function of each bit |
|---|---|---|
| Byte for special functions | DATx.1 | Can be set only if the hardware enable is active.<br>0x01 = **Din_1**; Set signal Din_1 to active<br>0x04 = **TST_CMP**; Set Error/Comp output direct to 24V (disable signalizing of error output)<br>0x10 = **Din_2**; Set signal Din_2 to active<br>0x80 = **BUS_DISABLE** (Module disable via Profinet)<br>In this mode not applicable, reserved:<br>0x02 = CNTRL_1<br>0x08 = CNTRL_2<br>0x20 = CNTRL_3<br>0x40 = CNTRL_4 |
| | | The special function byte will be set to internal 0x00 (reset) when the hardware enable is inactive. |

**Response telegram:**

Write first…

| Comment | Byte | Structure | Abbreviation | Description |
|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by the write command |
| | 1 | | SADR | Slave address |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master |
| | 3 | Data of SADR Module | DAT.1H | **Status word** \| High byte of module state |
| | 4 | | DAT.1L | Low byte of module state |
| | 5 | | DAT.2H | **VALUE1** = d1.11 |
| | 6 | | DAT.2L | Actual value, feedback value (Branch 1) |
| | 7 | | DAT.3H | **VALUE2** = d2.11 |
| | 8 | | DAT.3L | Actual value, feedback value (Branch 2) |
| ….Write last | | 9 | | Overall telegram length in bytes |

Explanation of High and low byte of module state (**Status word**):

| Description | Abbreviation | Explanation of each bit |
|---|---|---|
| High byte of module state | DAT.1H | 0x01 = Digital input **SP1** (S1.01) active *)<br>0x02 = Digital input **SP2** (S1.02) active *)<br>0x04 = **HW_ENABLE**, hardware enable active<br>0x08 = **ERROR**, Error is pending<br>0x10 = Digital input **SP3** (S2.01) active *)<br>0x20 = Digital input **SP4** (S2.02) active *)<br>0x80 = **BUS_DISABLE** is set<br>In this mode not applicable, reserved:<br>0x40 = STAT_5 |
| Low byte of module state | DAT.1L | if „Error occurred" bit is set: error number<br>otherwise:<br>0x01 = **Din_1** active<br>0x02 = **Dout_1** active<br>0x04 = **Comp_1** active<br>0x08 = **Comp_11** active<br>0x10 = **Din_2** active<br>0x20 = **Dout_2** active<br>0x40 = **Comp_2** active<br>0x80 = **Comp_22** active |

*) The designation and availability of the input depends on the module variant.

### 9.5.6 Version for Mode 8 (closed loop)

Master writes multiple parameters (at all 8 bytes)

Write first…

| Comment | Byte | Structure | Abbreviation | Description | |
|---|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by User, Range 0..255, 0..0xFF | |
| | 1 | | SNUM = 1 | Single slave = 1 | |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master | |
| | 3 | Module Data | DAT1.1 | **Control byte** | Byte for special functions |
| | 4 | | DAT1.2H | **Process Value1 = A1.01** | |
| | 5 | | DAT1.2L | Set value input (Profinet) for (Branch 1) | |
| | 6 | | DAT1.3H | **Process Value2 = A2.01 S** | |
| | 7 | | DAT1.3L | Set value input (Profinet) for (Branch 2) | |
| ….Write last | | 8 | Overall telegram length in bytes | | |

Explanation of Byte for special functions (**Control byte**)

| Description | Abbreviation | Function of each bit |
|---|---|---|
| Byte for special functions | DATx.1 | Can be set only if the hardware enable is active.<br>0x01 = **Din_1**; Set signal Din_1 to active<br>0x04 = **TST_CMP**; Set Error/Comp output direct to 24V (disable signalizing of error output)<br>0x10 = **Din_2**; Set signal Din_2 to active<br>0x80 = **BUS_DISABLE** (Module disable via Profinet)<br>Not applicable, reserved:<br>0x02 = CNTRL_1<br>0x08 = CNTRL_2<br>0x20 = CNTRL_3<br>0x40 = CNTRL_4 |
| | | The special function byte will be set to internal 0x00 (reset) when the hardware enable is inactive. |

**Response telegram:**

Write first…

| Comment | Byte | Structure | Abbreviation | Description | |
|---|---|---|---|---|---|
| | 0 | Telegram header | TADR | Telegram address, defined by the write command | |
| | 1 | | SADR | Slave address | |
| | 2 | | CMD | Command: 15 = Write multiple parameter by the master | |
| | 3 | Data of SADR Module | DAT.1H | **Status word** | High byte of module state |
| | 4 | | DAT.1L | | Low byte of module state |
| | 5 | | DAT.2H | **VALUE1** = d1.11 | |
| | 6 | | DAT.2L | Actual value, feedback value (Branch 1) | |
| | 7 | | DAT.3H | **VALUE2** = d2.11 | |
| | 8 | | DAT.3L | Actual value, feedback value (Branch 2) | |
| ….Write last | | 9 | Overall telegram length in bytes | | |

Explanation of High and low byte of module state (**Status word**):

| Description | Abbreviation | Explanation of each bit |
|---|---|---|
| High byte of module state | DAT.1H | 0x01 = Digital input **SP1** (S1.01) active *)<br>0x02 = Digital input **SP2** (S1.02) active *)<br>0x04 = **HW_ENABLE**, hardware enable active<br>0x08 = **ERROR**, Error is pending<br>0x10 = Digital input **SP3** (S2.01) active *)<br>0x20 = Digital input **SP4** (S2.02) active *)<br>0x80 = **BUS_DISABLE** is set<br>In this mode not applicable, reserved:<br>0x40 = STAT_5 |
| Low byte of module state | DAT.1L | if „Error occurred" bit is set: error number<br>otherwise:<br>0x01 = **Din_1** active<br>0x02 = **Dout_1** active<br>0x04 = **Comp_1** active<br>0x08 = **Comp_11** active<br>0x10 = **Din_2** active<br>0x20 = **Dout_2** active<br>0x40 = **Comp_2** active<br>0x80 = **Comp_22** active |

*) The designation and availability of the input depends on the module variant.

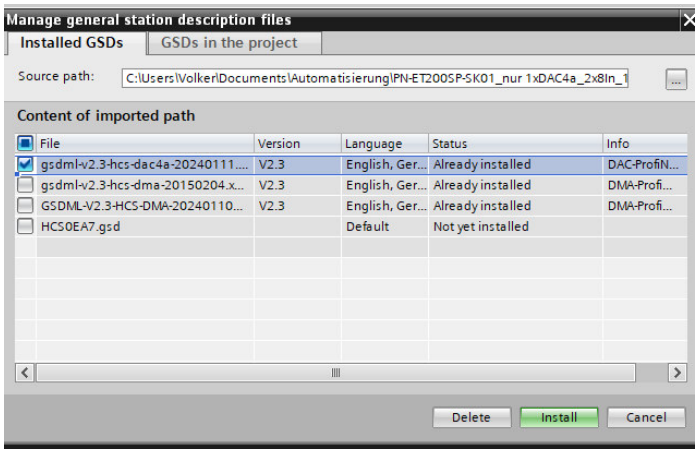
# 10 Configuration for SIEMENS TIA portal

Two Profinet function blocks for the Siemens TIA portal are available, and can be downloaded from the HCS web side. Which one, the user should take, is depending on the used CPU family.

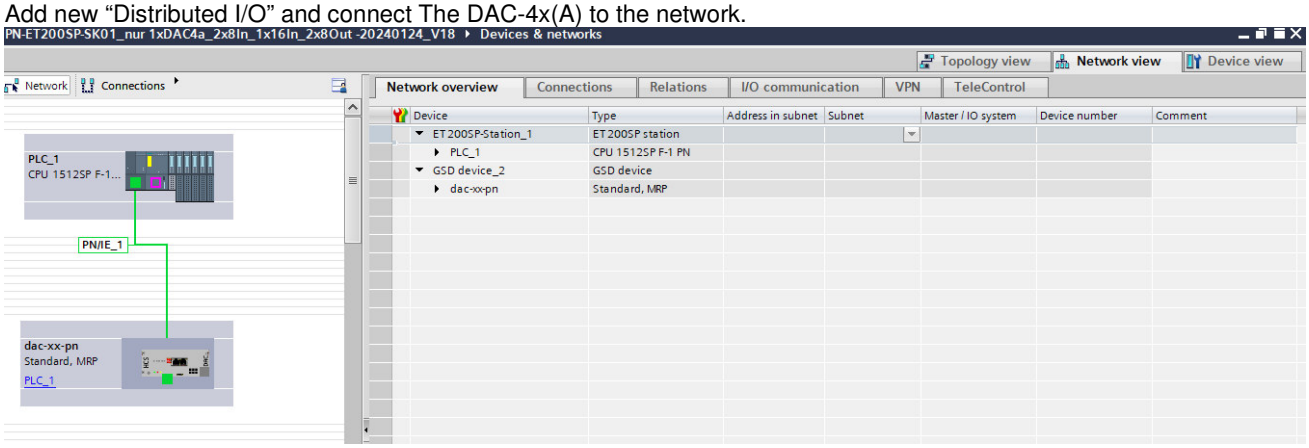
- Siemens CPU series S7-300,400 — PN_DAC4a_CLASSIC_DB(Vxx).scl
- Siemens CPU series S7-1200,1500 — PN_DAC4a_DB.scl(Vxx).scl

## *10.1 Setup the network configuration for the DAC-4x(A)*

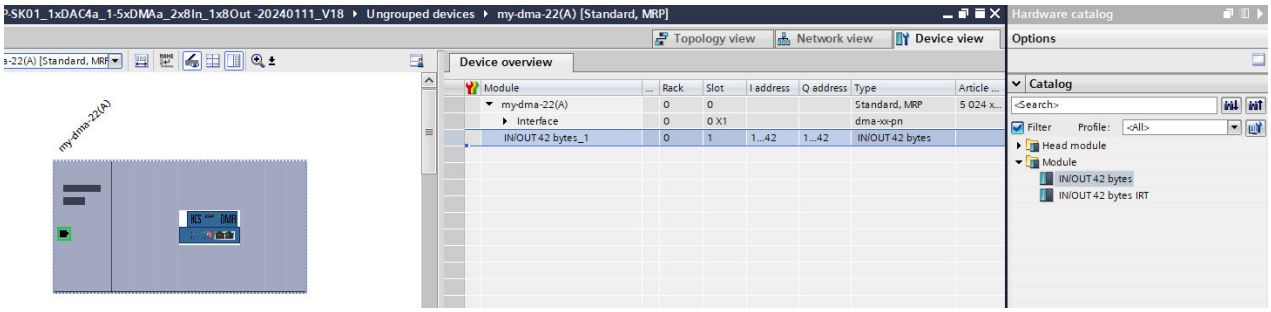
Download and install the GSD file




Add new “Distributed I/O” and connect The DAC-4x(A) to the network.



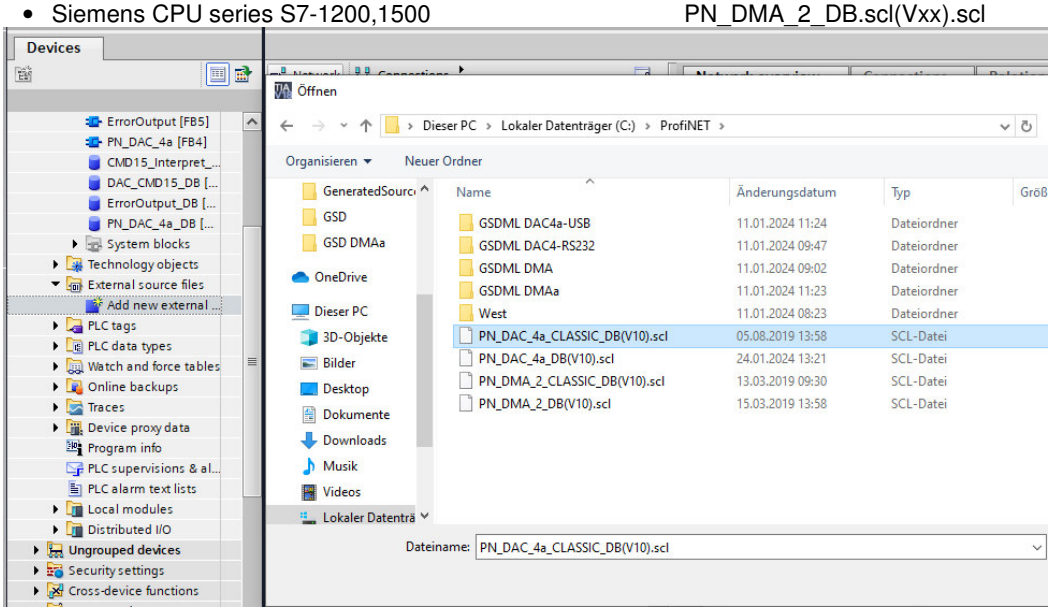

## *10.2 Assign the IO-Module from the GSD to the device*



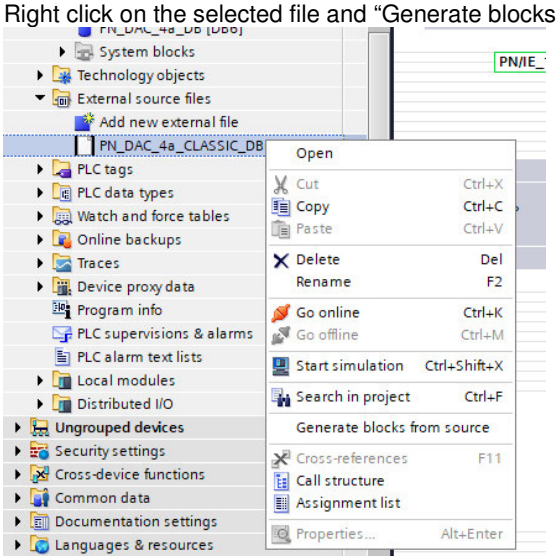
## *10.3 Install the function block*

Add new external file, Chose the function block according your used CPU series

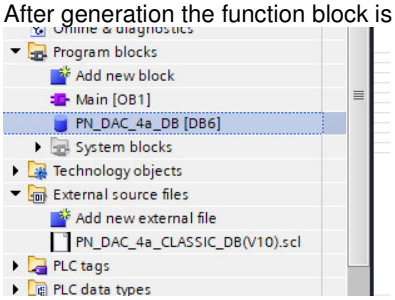
- Siemens CPU series S7-300,400 PN_DMA_2_CLASSIC_DB(Vxx).scl
- Siemens CPU series S7-1200,1500 PN_DMA_2_DB.scl(Vxx).scl




Right click on the selected file and “Generate blocks from the source”



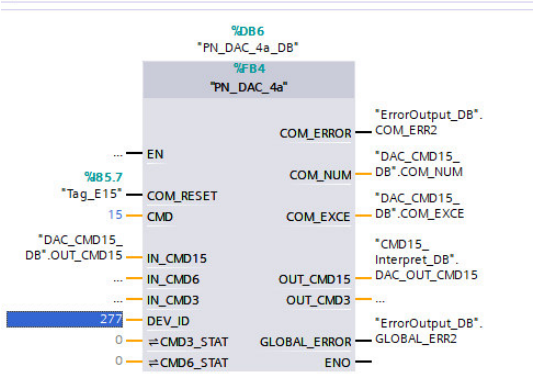

After generation the function block is now available



## 10.4 Explanation of the function block

### 10.4.1 General

In this chapter we describe, how to use the function block in the different modes.
The structure and function for CMD 3, CMD 6 is the same in every working mode of the DAC-4x(A). Only the CMD 15 differs depending on the DAC-4x(A) operation mode.




The Bus slave address SNUM of the connected DAC-4x(A) is fixed and is set to “1” (single slave).

| SNUM | Int |
|---|---|

The COM_ERROR is obtained from the telegram header, which is transmitted from the Profinet node to the PLC.
When this error is pending, the slave address source and the cause can be found out by analyze the COM_EXCE code.
(See □ Chapter “8.1.3 CMD = 3, Error from DAC-4x(A) (4 bytes)“,page 43
or □ Chapter “8.3.3 CMD = 6, Error from DAC-4x(A) (4 bytes)”, page 45

Reset an communication error can by set the input #COM_RESET

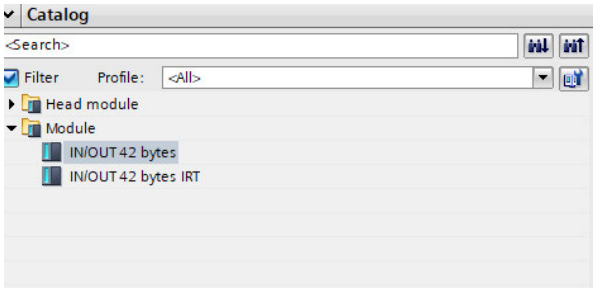
| COM_RESET | Bool |
|---|---|

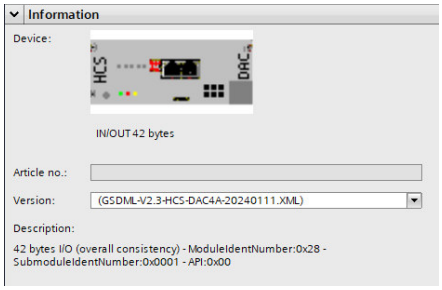
An GLOBAL_ERROR occurs if any communication error or/and internal DMA-2 error is pending.

| GLOBAL_ERROR | Bool |
|---|---|

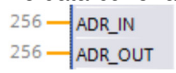
### *10.4.2 Classic CPU 300,400: Definition of the data buffer address*

Unlike the CPU 1200, 1500, the E-address and A-address must be assigned to the variables #ADR_IN and #ADR_OUT here.
Add the IN/OUT module

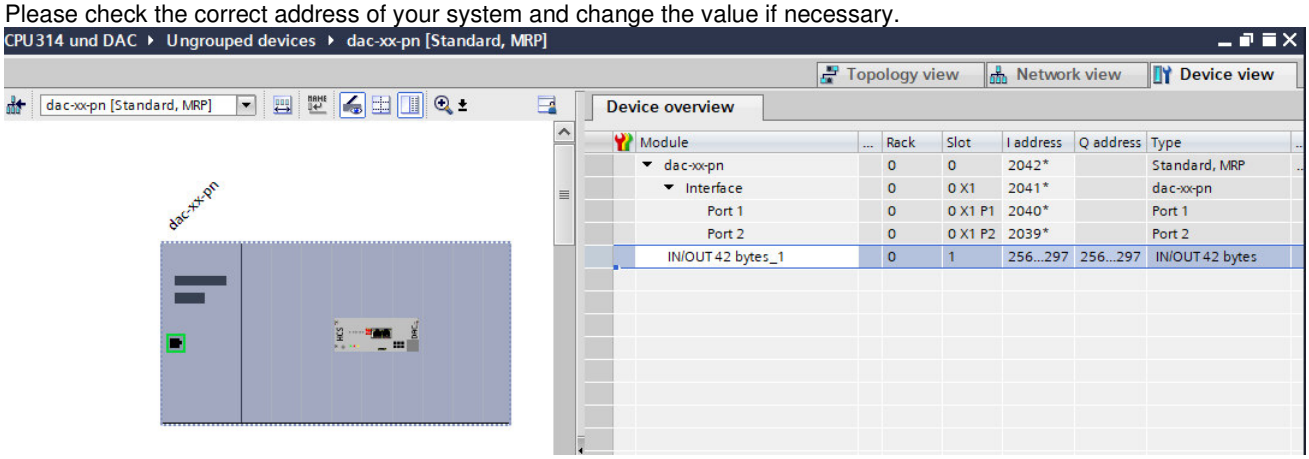



The data buffer address #ADR_IN, #ADR_OUT is predefined to 256.

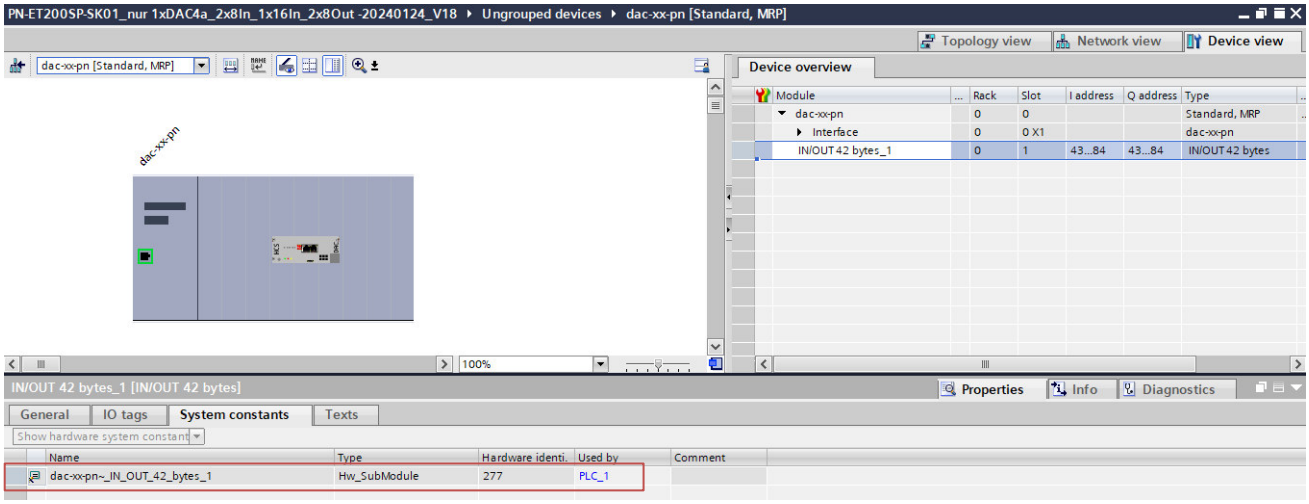



Please check the correct address of your system and change the value if necessary.

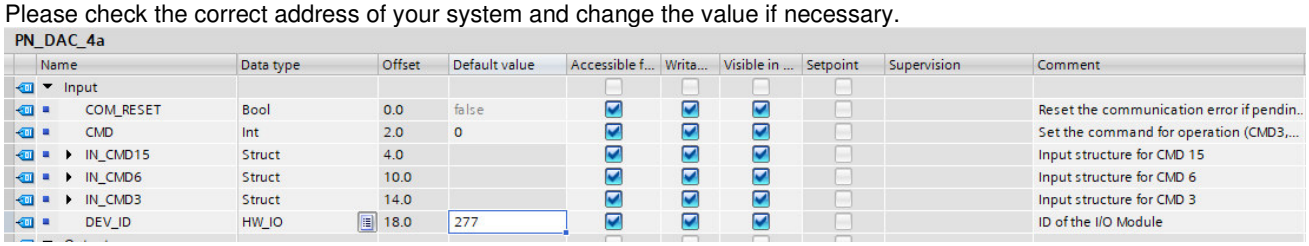


### *10.4.3 CPU 1200,1500: Specification of the data buffer address based on the hardware identifier*

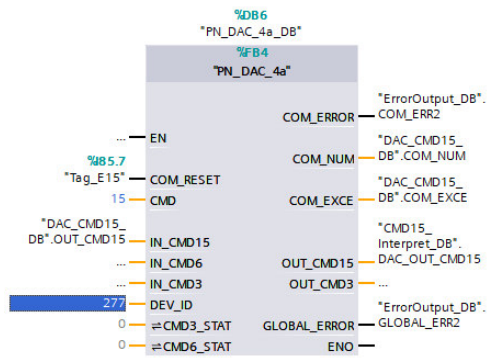
In contrast to CPU 300, 400, the hardware ID for the GSD I / O module must be assigned to the variable #DEV_ID.
The hardware-ID can be found here:



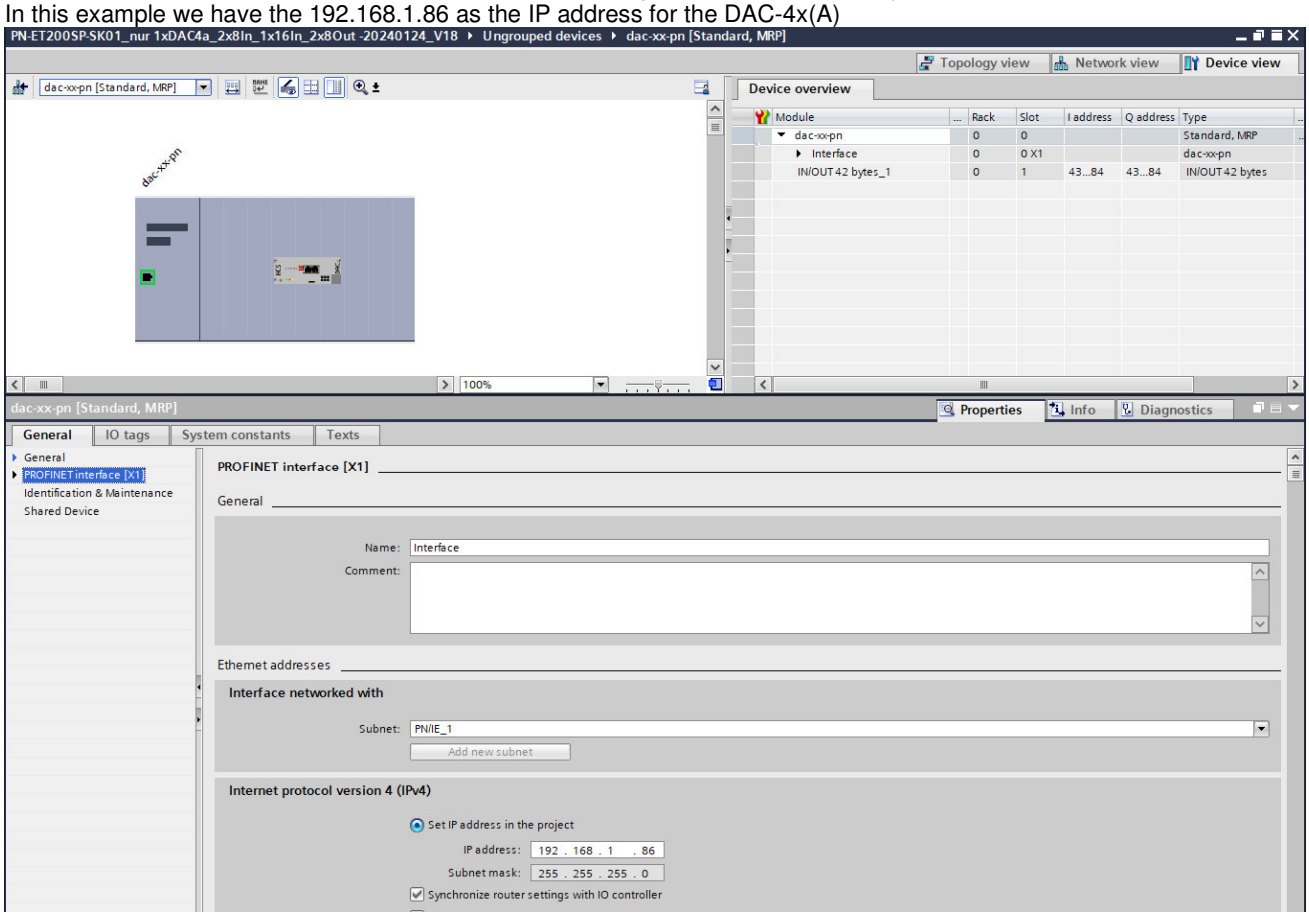

Please check the correct address of your system and change the value if necessary.




The data buffer address #ADR_IN, #ADR_OUT is predefined in this example at 277.



Please check the correct IP address of the device and change the value if necessary.
In this example we have the 192.168.1.86 as the IP address for the DAC-4x(A)




### *10.4.4 CMD 3, Reading values from one DAC-4x(A) node*

The DAC-4x(A) connected to the Profinet has a fix SNUM (slave number). Together with this number it is possible to read a set of parameter of this device.
The parameter-ID's are shown in table □ Chapter "11 List of parameters", page 64

An reading access is done as followed:

- The predefined slave number SNUM for the DAC4x(A) is 1.

| | |
|---|---|
| SNUM | Int |

For slave 1 #SNUM := 1;.

- Setup the CMD: # CMD := 3;

| | |
|---|---|
| CMD | Int |

- Write the starting address of the first parameter ID which should be read by the structure

| | |
|---|---|
| IN_CMD3 | Struct |
| PAR_ID | Int |

E.g. parameter 22, S1.01 #IN_CMD3.PAR_ID := 22;

- Define the number of parameters to be read (maximum 8).

| | |
|---|---|
| IN_CMD3 | Struct |
| PAR_ID | Int |
| PAR_N | Int |

E.g. 4 parameters S1.01..S1.04 #IN_CMD3.PAR_N := 4;

- Definition of the states in #CMD3_STAT:

| | | | |
|---|---|---|---|
| CMD_UNSED | Int | | 0 |
| CMD_START | Int | | 1 |
| CMD_PROGRESS | Int | | 2 |
| CMD_FINISH | Int | | 4 |
| CMD_FAIL | Int | | 8 |

0: no action
1: Start of reading

2: Reading in process
4: Reading finished
8: Reading error occurred.

- Start the process, by writing #CMD3_STAT := 1; (CMD_START)

| | |
|---|---|
| CMD3_STAT | Int |

- When #CMD3_STAT reaches the value 4 (CMD_FINISH), Reading is finished, and the values can be read in the structure #OUT_CMD.VALUE[1..8]. In addition, the number of received bytes will be shown I the structure #OUT_CMD.CNT. Every read parameter needs to 2 bytes.

| | |
|---|---|
| OUT_CMD3 | Struct |
| CNT | Int |
| VALUE | Array[1..8] of Int |
| VALUE[1] | Int |
| VALUE[2] | Int |
| VALUE[3] | Int |

- If a reading error occurred, the reason for the error can be analyzed by checking the #COM_xxxxx outputs of the function block.

| | |
|---|---|
| COM_ERROR | Bool |
| COM_NUM | Int |
| COM_EXCE | Int |

### *10.4.5 CMD 6, Writing parameters the DAC-4x(A)*

**The written values are stored in a non-volatile memory.**
The DAC-4x(A) connected to the Profinet has a fix SNUM (slave number). Together with this number it is possible to write a new parameter value to the parameter ID of this device.
The parameter-ID's are shown in table □ Chapter "11 List of parameters", page 64

An parameter writing access is done as followed:

- The predefined slave number SNUM for the DAC4x(A) is 1.

| | |
|---|---|
| SNUM | Int |

For slave 1 #SNUM := 1;.

- Setup the CMD: # CMD := 6;

| | |
|---|---|
| CMD | Int |

- Define the parameter ID and the value which should be changed.

| | |
|---|---|
| IN_CMD6 | Struct |
| PAR_ID | Int |
| PAR_VALUE | Int |

E.g. parameter 22, S1.01 =1000 (1.0 V) #IN_CMD3.PAR_ID := 22;
#IN_CMD3.PAR_VALUE := 1000;

- Definition of the states in #CMD6_STAT:

| | | | |
|---|---|---|---|
| CMD_UNSED | Int | | 0 |
| CMD_START | Int | | 1 |
| CMD_PROGRESS | Int | | 2 |
| CMD_FINISH | Int | | 4 |
| CMD_FAIL | Int | | 8 |

0: no action
1: Start of writing
2: Writing in process
4: Writing finished
8: Writing error occurred.

- Start the process, by writing #CMD6_STAT := 1; (CMD_START)

| | |
|---|---|
| CMD6_STAT | Int |

- When #CMD6_STAT reaches the value 4 (CMD_FINISH), writing is finished

- If an writing error occurred, the reason for the error can be analyzed by checking the #COM_xxxxx outputs of the function block.

| | |
|---|---|
| COM_ERROR | Bool |
| COM_NUM | Int |
| COM_EXCE | Int |

### *10.4.6 CMD 15, Writing and Reading once the connected DAC-4x(A)*

The CMD 15 makes it possible to send and receive process data from the DAC4-4x(A) connected to the Profinet node. The meaning of the process values depending on the mode of the DAC4-4x(A) is working at.
See □ Chapter “9.5 CMD = 15, structure of Profinet telegram definition, depending on SW Versions“, page 49

Also, the Control-Byte is depending on the working mode. Explanation of the bits
See □ Chapter “9.5.1.3 Explanation of special functions and module state bits“, page 50

An CMD 15 access is done as followed:

- Define the number of slaves SNUM of the modules connected to the Profinet node. In case of DAC-4x(A), only a single slave can be connected.

| | |
|---|---|
| SNUM | Int |

For single slave #SNUM := 1;.

- Setup the CMD: # CMD := 15;

| | |
|---|---|
| CMD | Int |

- Write the process data you want to the DAC-4x(A) in the structure provided.
-

| | |
|---|---|
| IN_CMD15 | Struct |

DAC-4x(A) input structure

Structure of the transmit data set:

| | |
|---|---|
| IN_CMD15 | Struct |
| BUS_DISABLE | Bool |
| Din_1 | Bool |
| Din_2 | Bool |
| TST_CMP | Bool |
| CNTRL_1 | Bool |
| CNTRL_2 | Bool |
| CNTRL_3 | Bool |
| CNTRL_4 | Bool |
| VALUE1 | Int |
| VALUE2 | Int |

Control Byte

Process values to DAC-4x(A)

- Structure of the receive data set

| | |
|---|---|
| OUT_CMD15 | Struct |
| HW_ENABLE | Bool |
| BUS_DISABLE | Bool |
| STAT_1 | Bool |
| STAT_2 | Bool |
| STAT_3 | Bool |
| STAT_4 | Bool |
| STAT_5 | Bool |
| ERROR | Bool |
| ERROR_NO | Byte |
| Din_1 | Bool |
| Dout_1 | Bool |
| DKOMP_1 | Bool |
| DKOMP_11 | Bool |
| Din_2 | Bool |
| Dout_2 | Bool |
| DKOMP_2 | Bool |
| DKOMP_22 | Bool |
| VALUE1 | Int |
| VALUE2 | Int |

Status Word

Process values from DAC-4x(A)

- If an error has occurred, the cause of the error can be analyzed by evaluating the outputs #COM_xxxxx of the function block.

| | |
|---|---|
| COM_ERROR | Bool |
| COM_NUM | Int |
| COM_EXCE | Int |

# 11 List of parameters

Parameters in grey letters are not yet implemented in the DMA Software.
All values in decimal, for usage as data values they must be converted into Hex
R = read only parameter.
X = implemented, but not active (no access)
W/R = writing and reading possible.

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| **0x00** | 0 | R | Vers | Software version | --- | xx.xx | xx.xx | xx.xx | Depends on HW and SW version |
| **0x01** | 1 | R | d1.01 | Sum of analogue set value | V | -9999 | 9999 | - | -1000 == -1.000 V; |
| **0x02** | 2 | R | d1.02 | Sum of all post ramp set values | V | -9999 | 9999 | - | 1000 == 1.000 V |
| **0x03** | 3 | R | d1.03 | Set values after linearization | V | -9999 | 9999 | - | |
| **0x04** | 4 | R | d1.04 | Value after gain adjustment. | V | -9999 | 9999 | - | |
| **0x05** | 5 | R | d1.05 | Signal A | V | -9999 | 9999 | - | |
| **0x06** | 6 | R | d1.06 | Signal B | V | -9999 | 9999 | - | |
| **0x07** | 7 | R | d1.07 | Current A | A | 0 | 5000 | - | 1000 == 1.000 A |
| **0x08** | 8 | R | d1.08 | Current B | A | 0 | 5000 | - | |
| **0x09** | 9 | R | d1.09 | Total current | A | 0 | 5000 | - | |
| **0x0A** | 10 | R | d1.10 | Desired value | V | -9999 | 9999 | - | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x0B** | 11 | R | d1.11 | Actual value, feedback value | V | -9999 | 9999 | - | |
| **0x0C** | 12 | R | d1.12 | Lag error | V | -9999 | 9999 | - | |
| **0x0D** | 13 | R | d1.13 | Controller output | V | -9999 | 9999 | - | |
| **0x0E** | 14 | R | d2.01 | Sum of analogue set value | V | -9999 | 9999 | - | |
| **0x0F** | 15 | R | d2.02 | Sum of all post ramp set values | V | -9999 | 9999 | - | |
| **0x10** | 16 | R | d2.03 | Set values after linearization | V | -9999 | 9999 | - | |
| **0x11** | 17 | R | d2.04 | Value after gain adjustment. | V | -9999 | 9999 | - | |
| **0x12** | 18 | R | d2.10 | Desired value | V | -9999 | 9999 | - | |
| **0x13** | 19 | R | d2.11 | Actual value, feedback value | V | -9999 | 9999 | - | |
| **0x14** | 20 | R | d2.12 | Lag error | V | -9999 | 9999 | - | |
| **0x15** | 21 | R | d2.13 | Controller output | V | -9999 | 9999 | - | |
| **0x16** | 22 | R/W | S1.01 | Set Value1 | V | -9999 | 9999 | 0 | |
| **0x17** | 23 | R/W | S1.02 | Set Value2 | V | -9999 | 9999 | 0 | |
| **0x18** | 24 | R/W | S1.03 | Set Value3 | V | -9999 | 9999 | 0 | |
| **0x19** | 25 | R/W | S1.04 | Set Value4 | V | -9999 | 9999 | 0 | |
| **0x1A** | 26 | --- | S1.08 | reserved | | --- | --- | --- | --- |
| **0x1B** | 27 | R/W | r1.01 | Ramp from 0 □ - | s | 0 | 3950 | 0 | 1 == 10ms |
| **0x1C** | 28 | R/W | r1.02 | Ramp from – □ 0 | s | 0 | 3950 | 0 | 0 == Ramp function off |
| **0x1D** | 29 | R/W | r1.03 | Ramp from 0 □ + | s | 0 | 3950 | 0 | |
| **0x1E** | 30 | R/W | r1.04 | Ramp from + □ 0 | s | 0 | 3950 | 0 | |
| **0x1F** | 31 | R/W | A1.01 | Profinet set value (Branch 1) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V This value is, when written, always active. Cleared by reset, or writing “0”. For switching off of the analogue set-value, set E17 = “2” |
| **0x20** | 32 | R/W | A1.02 | Profinet feedback value (Branch 1) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V C1.09 has to be switched off in order to avoid interferences |
| **0x21** | 33 | R/W | S2.01 | Set Value1 (Branch 2) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x22** | 34 | R/W | S2.02 | Set Value2 (Branch 2) | V | -9999 | 9999 | 0 | |
| **0x23** | 35 | R/W | r2.01 | Ramp from 0 □ - | s | 0 | 3950 | 0 | 1 == 10ms |
| **0x24** | 36 | R/W | r2.02 | Ramp from – □ 0 | s | 0 | 3950 | 0 | 0 == Ramp function off |
| **0x25** | 37 | R/W | r2.03 | Ramp from 0 □ + (Branch 2) | S | 0 | 3950 | 0 | |
| **0x26** | 38 | R/W | r2.04 | Ramp from + □ 0 (Branch 2) | S | 0 | 3950 | 0 | |
| **0x27** | 39 | R/W | A2.01 | Profinet set value (Branch 2) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V This value is, when written, always active. Cleared by reset, or writing “0”. For switching off of the analogue set-value, set E17 = “2” |
| **0x28** | 40 | R/W | A2.02 | Profinet feedback value (Branch 2) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V C2.09 has to be switched off in order to avoid interferences |
| **0x29** | 41 | R/W | C1.00 | Controller selection | --- | 0 | 4 | 1 | 0 = off<br>1 = P-$PT_1$-I-$DT_1$<br>2 =Remote<br>3 =dff<br>4 =Remote + dff |
| **0x2A** | 42 | R/W | C1.01 | Safety function | --- | 0 | 1 | 0 | 0 = off; 1 = on |
| **0x2B** | 43 | R/W | C1.02 | Linearization | --- | 0 | 6 | 0 | 0 = linear<br>1 ... 5 = standard curves<br>6 = customized curve |
| **0x2C** | 44 | R/W | C1.03 | Gain A | V/V | 0 | 200 | 100 | 100 == Factor 1.00 |
| **0x2D** | 45 | R/W | C1.04 | Gain B | V/V | 0 | 200 | 100 | 100 == Factor 1.00 |

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| **0x2E** | 46 | R/W | C1.05 | Set value sign and gain | V/V | -400 | 400 | 100 | 100 == Factor 1.00<br>Sign and gain! |
| **0x2F** | 47 | R/W | C1.06 | Set value offset | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x30** | 48 | R/W | C1.07 | Dead band compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0x31** | 49 | R/W | C1.08 | Dead band compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **0x32** | 50 | R/W | C1.09 | Sensor type<br><br>Attention:<br>No negative controller output possible when 10, 11 or 12 is selected! | --- | 1 | 28 | 4 | off = Deactivated<br>1 = 0...20mA<br>2 = 4...20mA<br>3 = 12mA +-8mA<br>4 = 0...10V<br>5 = 0...+-10V<br>6 = 6V +-2.5V<br>7 = 7.5V +-2.5V<br>8 = 6V +-5V<br>9 = 7,5 +-5V><br>10 = 0..20mA (positive contr. outp. only)<br>11 = 4..20mA (positive contr. outp. only)<br>12 = 0..10V (positive contr. outp. only)<br>13 = reserved<br>14 = 5V+-3.0V<br>15..27 = reserved<br>28 = 5V+-2.6V |
| **0x33** | 51 | R/W | C1.10 | Actual value gain | V/V | 0 | 400 | 100 | 100 == Factor 1.00 |
| **0x34** | 52 | R/W | C1.11 | Actual value offset | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x35** | 53 | R/W | C1.12 | Actual value sign | --- | - 1 | + 1 | 1 | - 1 = negative<br>0 = off<br>+ 1 = positive |
| **0x36** | 54 | R/W | C1.13 | P-Portion $K_{P1}$ | V/V | 0 | 400 | 0 | 100 == Factor 1.00 |
| **0x37** | 55 | R/W | C1.14 | T-Portion for PT1 (to C1.16) | S | 0 | 1000 | 0 | 100 == 1.00 |
| **0x38** | 56 | R/W | C1.15 | Threshold (C1.13, C1.16) | V | 0 | 9999 | 9999 | 1000 == 1.000 V |
| **0x39** | 57 | R/W | C1.16 | P-Portion $K_{P2}$ | V/V | 0 | 400 | 0 | 100 == Factor 1.00 |
| **0x3A** | 58 | R/W | C1.17 | I-Portion | V/s | 0 | 4000 | 0 | 1000 == 1.000 |
| **0x3B** | 59 | R/W | C1.18 | D-Portion | Vs | 0 | 400 | 0 | 100 == 1.00 |
| **0x3C** | 60 | R/W | C1.19 | T-Portion for DT1 | S | 0 | 1000 | 0 | 100 == 1.00 |
| **0x3D** | 61 | R/W | C1.20 | Gain ( C1.13 and C1.16) | V/V | 1 | 32 | 1 | 2 = Factor 2.00 |
| **0x3E** | 62 | R/W | C1.21 | Comparator upper level | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x3F** | 63 | R/W | C1.22 | Comparator lower level | V | -9999 | 9999 | 0 | |
| **0x40** | 64 | R/W | C1.23 | Comparator delay into window | S | 0 | 9999 | 0 | 1 == 10 ms |
| **0x41** | 65 | R/W | C1.24 | Comparator delay out of window | S | 0 | 9999 | 0 | 0 == no delay |
| **0x42** | 66 | R/W | C1.25 | Comparator selection COMP_1 | --- | 0 | 3 | 0 | 0 = off<br>1 = Set value<br>2 = Actual value<br>3 =Lag error |
| **0x43** | 67 | R/W | C1.26 | Cable fracture detection feedback | --- | 0 | 1 | 0 | 0 = off; 1 = active |
| **0x44** | 68 | R/W | C2.00 | Controller selection | --- | 0 | 4 | 0 | 0 = off<br>1 = $P\text{-}PT_1\text{-}I\text{-}DT_1$<br>2 =Remote<br>3 =dff<br>4 =Remote + dff |
| **0x45** | 69 | R/W | C2.01 | Safety function | --- | 0 | 1 | 0 | 0 = off; 1 = on |
| **0x46** | 70 | R/W | C2.02 | Linearization | --- | 0 | 6 | 0 | 0 = linear<br>1 ... 5 = standard curves<br>6 = customized curve |
| **0x47** | 71 | R/W | C2.03 | Gain A | V/V | 0 | 200 | 100 | 100 == Factor 1.00 |
| **0x48** | 72 | R/W | C2.04 | Gain B | V/V | 0 | 200 | 100 | 100 == Factor 1.00 |
| **0x49** | 73 | R/W | C2.05 | Set value sign and gain | V/V | -400 | 400 | 100 | 100 == Factor 1.00<br>Sign and gain! |
| **0x4A** | 74 | R/W | C2.06 | Set value offset | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x4B** | 75 | R/W | C2.07 | Dead band compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0x4C** | 76 | R/W | C2.08 | Dead band compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **0x4D** | 77 | R/W | C2.09 | Sensor type<br><br>Attention:<br>No negative controller output possible when 10, 11 or 12 is selected! | --- | 0 | 12 | 4 | off = Deactivated<br>1 = 0...20mA<br>2 = 4...20mA<br>3 = 12mA +-8mA<br>4 = 0...10V<br>5 = 0...+-10V<br>6 = 6V +-2.5V<br>7 = 7.5V +-2.5V<br>8 = 6V +-5V |

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 9 = 7,5 +-5V><br>10 = 0..20mA (positive contr. outp. only)<br>11 = 4..20mA (positive contr. outp. only)<br>12 = 0..10V (positive contr. outp. only)<br>13 = reserved<br>14 = 5V+-3.0V<br>15..27 = reserved<br>28 = 5V+-2.6V |
| **0x4E** | 78 | R/W | C2.10 | Actual value gain | V/V | 0 | 400 | 100 | 100 == Factor 1.00 |
| **0x4F** | 79 | R/W | C2.11 | Actual value offset | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x50** | 80 | R/W | C2.12 | Actual value sign | --- | - 1 | + 1 | 1 | - 1 = negative<br>0 = off<br>+ 1 = positive |
| **0x51** | 81 | R/W | C2.13 | P-Portion $K_{P1}$ | V/V | 0 | 400 | 0 | 100 == Factor 1.00 |
| **0x52** | 82 | R/W | C2.14 | T-Portion for PT1 (to C1.16) | S | 0 | 1000 | 0 | 100 == 1.00 |
| **0x53** | 83 | R/W | C2.15 | Threshold (C1.13, C1.16) | V | 0 | 9999 | 9999 | 1000 == 1.000 V |
| **0x54** | 84 | R/W | C2.16 | P-Portion $K_{P2}$ | V/V | 0 | 400 | 0 | 100 == Factor 1.00 |
| **0x55** | 85 | R/W | C2.17 | I-Portion | V/s | 0 | 4000 | 0 | 1000 == 1.000 |
| **0x56** | 86 | R/W | C2.18 | D-Portion | Vs | 0 | 400 | 0 | 100 == 1.00 |
| **0x57** | 87 | R/W | C2.19 | T-Portion for DT1 | S | 0 | 1000 | 0 | 100 == 1.00 |
| **0x58** | 88 | R/W | C2.20 | Gain (C1.13 and C1.16) | V/V | 1 | 32 | 1 | 2 = Factor 2.00 |
| **0x59** | 89 | R/W | C2.21 | Comparator upper level | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0x5A** | 90 | R/W | C2.22 | Comparator lower level | V | -9999 | 9999 | 0 | |
| **0x5B** | 91 | R/W | C2.23 | Comparator delay into window | S | 0 | 9999 | 0 | 1 == 10 ms |
| **0x5C** | 92 | R/W | C2.24 | Comparator delay out of window | S | 0 | 9999 | 0 | 0 == no delay |
| **0x5D** | 93 | R/W | C2.25 | Comparator selection COMP_2 | --- | 0 | 3 | 0 | 0 = off<br>1 = Set value<br>2 = Actual value<br>3 =Lag error |
| **0x5E** | 94 | R/W | C2.26 | Cable fracture detection feedback | --- | 0 | 1 | 0 | 0 = off; 1 = active |
| **0x5F** | 95 | R | E00 | Operation mode<br>(depends on HW + SW version) | --- | 1 | 11 | 3 | 1 = Open loop one valve<br>2 = Open loop two valves<br>3 = Closed loop one valve<br>4 = Closed loop on application<br>6 = Closed loop valve/application<br>8 = Closed loop application/ application<br>10 = Controller function only, one feedback<br>11 = Controller function only, two feedback |
| **0x60** | 96 | R/W | E01 | Analogue output<br>(depends on HW + SW version) | --- | 1<br>and<br>14 | 13<br>and<br>21 | 1 | 1 = d1.01 to<br>13 = d1.13 and<br>14 = d2.01 to<br>21 = d2.13 |
| **0x61** | 97 | R/W | E02 | Push-Pull function<br>(depends on HW + SW version) | --- | 0 | 1 | 0 | 0 = off v(no push pull)<br>1 = active<br>2 = common “+” for solenoids<br>(Remark: only for max 0,8 A current )<br>3 = full bridge<br>4 = off + detection low current to solenoid<br>5 = active + detection low current to solenoid |
| **0x62** | 98 | R/W | E03<br>E1.03 | Solenoid selection<br>Solenoid selection A<br>(depends on HW + SW version) | --- | 1 | 7 | 6 | Standard solenoid definition<br>1 = 0,8 A<br>2 = 1,1 A<br>3 = 1,3 A<br>4 = 1,6 A<br>5 = 2,4 A<br>6 = 2,7 A<br>7 = 3,5 A<br>(depends on HW + SW version) |
| **0x63** | 99 | R | E04<br>E1.04 | P-Portion current contr. Energization A | --- | 500 | 9999 | 500 | Default for 2,700 A solenoid |
| **0x64** | 100 | R | E05<br>E1.05 | I-Portion current contr. Energization A | --- | 500 | 9999 | 500 | |
| **0x65** | 101 | R | E06<br>E1.06 | P-Portion cur. contr. de- energization A | --- | 1300 | 9999 | 1300 | |
| **0x66** | 102 | R | E076<br>E1.07 | I-Portion cur. contr. De- energization A | --- | 500 | 9999 | 500 | |
| **0x67** | 103 | R/W | E08 | Ramp selection | --- | 1 | 2 | 1 | 0 = digital set v. (time constant)<br>1 = all set v. (rise constant.)<br>2 = selectable ramps |
| **0x68** | 104 | R/W | E09 | Time delay enable signal | s | 0 | 9999 | 0 | 1000 = 1.00s |

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| **0x69** | 105 | R/W | E10<br>E1.10 | Solenoid current adaptation A | --- | 50 | 110 | 100 | Variable adjustment of max. current<br>100 == Factor 1.00 |
| **0x6A** | 106 | R/W | E11 | Initial current solenoid A | V | 0 | 9999 | 0 | 3.000 V = 30 % of max. rated current |
| **0x6B** | 107 | R/W | E12 | Initial current solenoid B | V | 0 | 9999 | 0 | |
| **0x6C** | 108 | R/W | E13<br>E1.13 | Dither Amplitude A | V | 0 | 3000 | 0 | |
| **0x6D** | 109 | R/W | E14<br>E1.14 | Dither Frequency A | Hz | 1 | 300 | 0 | --- |
| **0x6E** | 110 | R/W | E15 | Selection set point S1.06 (U/I) | --- | 0 | 11 | 1 | off = Deactivated<br>1 = Voltage -10V..+10V<br>2 = Voltage 0..10V<br>3 = Voltage 5V+-5V<br>4 = Voltage 5V+-4.5V w/o signal observation<br>5 = Voltage 5V+-4.5V with signal observation<br>6 = 0..20mA w/o cable fract. det.<br>7 = 10+-10mA w/o cable fract. det.<br>8 = 4..20mA w/o cable fract. det.<br>9 = 4..20mA with cable fract. det.<br>10 = 12+-8mA w/o cable fract. det.<br>11 = 12+-8mA with cable fract. det. |
| **0x6F** | 111 | R/W | E16 | Reserved | --- | --- | --- | --- | --- |
| **0x70** | 112 | R/W | E17 | Set value activation mode | --- | 0 | 3 | off | off = 4 digital, 3 analog active<br>1 = 5 digital SP, 3 analog active<br>2 = only 4 digital active<br>3 = only 5 digital active |
| **0x71** | 113 | R/W | E18 | Error / Comparator output selection Modus 1,2 | --- | off | 1 | off | off = break off<br>1 = break on |
| | | | | Modus 3,4,10,11 | --- | off | 5 | off | off = break off, comp. positive logic<br>1 = break on, comp. positive logic<br>2 = break follows comp. positive logic<br>3 = break not and comp. positive logic<br>4 = break and comp. negative logic<br>5 = break not and comp neg. logic |
| | | | | Modus 6,8 | --- | off | 13 | off | off = break off, comp. positive logic<br>1 = break on (supply) comp. pos. log<br>2 = break follows comp., comp. pos. log<br>3 = break log. not comp., comp. pos. log<br>4 = break log. not comp., comp. neg. log<br>5 = break a. comp. same, comp. neg. l.<br>6 = break follow Comp1, comp. pos. log<br>7 = break log. not Comp1, comp. pos. l.<br>8 = break log. not Comp1, comp. neg. l.<br>9 = break a. Comp1 same, comp. neg. l.<br>10 = break follow Comp2; comp. pos. l.<br>11 = break log. not Comp2, comp. pos. l.<br>12 = break log. not Comp2; comp. neg. l.<br>13 = break a. Comp2 same, comp. neg. l |
| **0x72** | 114 | R/W | E19 | Factor for analog output (depends on HW + SW version) | V/V | -400 | 400 | 100 | 100 == Factor 1.00<br>Sign and gain! |
| **0x73** | 115 | X | E20 | Reserved | --- | --- | --- | --- | |
| **..** | .. | | | | | | | | |
| **0xA2** | 162 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xA3** | 163 | R/W | E 22 | Profinet Slave Address | --- | 0 | 32 | 0 | 0 = No bus function<br>1 = Standard Profinet<br>2..5 = Multiple Slave Profinet |
| **0xA4** | 164 | R/W | C1.27 | Hysteresis command A | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0xA5** | 165 | R/W | C2.27 | Hysteresis command B | 1 | -9999 | 9999 | 0 | |
| **0xA6** | 166 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xA7** | 167 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xA8** | 168 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xA9** | 169 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAA** | 170 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAB** | 171 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAC** | 172 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAD** | 173 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAE** | 174 | X | --- | Reserved | --- | --- | --- | --- | |
| **0xAF** | 175 | X | --- | Reserved | --- | --- | --- | --- | |

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| **0xB0** | 176 | R/W | E Int | Internal Digital Switches (Read/Set) | --- | 0 | 0xFFFF | | 0x0001 = Din_1 active (Read/Write)<br>0x0002 = Dout_1 active (Read only )<br>0x0004 = Comp_1 active (Read only )<br>0x0008 = reserved<br>0x0010 = Din_2 active (Read/Write)<br>0x0020 = Dout_2 active (Read only )<br>0x0040 = Comp_2 active (Read only )<br>0x0080 = reserved<br>0x0100 = Input SP1 active (Read only )<br>0x0200 = Input SP2 active (Read only )<br>0x0400 = Enable active (Read only )<br>0x0800 = Error active (Read only )<br>0x1000 = Input SP3 active (Read only )<br>0x2000 = Input SP4 active (Read only )<br>0x8000 = Bus Disable Card (Read/Write) |
| **0xB1** | 177 | R/W | E 23 | Profinet telegram timeout window | S | 0 | 9999 | 0 | 0 == function deactivated<br>1 == 10 ms |
| **0xB2** | 178 | R/W | E 26 | Profinet Timeout Error handling | --- | 0 | 1 | 0 | 0 = Blinking Enable Led shows Profinet timeout<br>1 = Profinet timeout causes error. |
| **0xB3** | 179 | R/W | C1.33 | I-portion limitation | V | 0 | 9999 | | 0 == 0.000 V; 9999 == 9.999 V |
| **0xB4** | 180 | R/W | C2.33 | I-portion limitation | V | 0 | 9999 | | 0 == 0.000 V; 9999 == 9.999 V |
| **..** | .. | | | | | | | | |
| **0xD2** | 210 | R/W | E 31 | Mode of the analog output | --- | 1 | 2 | 2 | 1 = Voltage output at pin 10b<br>2 = Current output at pin 10b. |
| **0xD3** | 211 | R/W | E 32 | Selection set point S1.07 (U/I) | --- | 0 | 11 | 1 | off = Deactivated<br>1 = Voltage -10V..+10V<br>2 = Voltage 0..10V<br>3 = Voltage 5V+-5V<br>4 = Voltage 5V+-4.5V w/o signal observation<br>5 = Voltage 5V+-4.5V with signal observation<br>6 = 0..20mA w/o cable fract. det.<br>7 = 10+-10mA w/o cable fract. det.<br>8 = 4..20mA w/o cable fract. det.<br>9 = 4..20mA with cable fract. det.<br>10 = 12+-8mA w/o cable fract. det.<br>11 = 12+-8mA with cable fract. det. |
| **0xD4** | 212 | R/W | E 33 | Offset for analog voltage output (depends on HW + SW version) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **0xD5** | 213 | R/W | E 34 | Offset for analog current output (depends on HW + SW version) | mA | -9999 | 9999 | 0 | -1000 == -1.000 mA; 1000 == 1.000 mA |
| **..** | .. | | | | | | | | |
| **0xDA** | 218 | R/W | C1.36 | Sensor signal correction factor for signal < 0. branch 1 | V/V | -100 | 100 | 100 | 100 == Factor 1.00 |
| **0xDB** | 219 | R/W | C2.36 | Sensor signal correction factor for signal < 0. branch 2 | V/V | -100 | 100 | 100 | 100 == Factor 1.00 |
| **0xDC** | 220 | R/W | C1.37 | Spool overlap compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0xDD** | 221 | R/W | C1.38 | Spool overlap compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **0xDE** | 222 | R/W | C2.37 | Spool overlap compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0xDF** | 223 | R/W | C2.38 | Spool overlap compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **..** | .. | | | | | | | | |
| **0xE5** | 229 | X | E sta | Reserved | --- | --- | --- | --- | |
| **0xB6** | 182 | R/W | E 35 | Factor analog output current | --- | -200 | 200 | 100 | 100 == Factor 1.00 |
| **..** | .. | | | | | | | | |
| **0xD4** | 212 | R/W | E 33 | Offset for analog output (depends on HW + SW version) | V | -9999 | 9999 | 0 | -1000 == -1.000 V; 1000 == 1.000 V |
| **..** | .. | | | | | | | | |
| **0xDA** | 218 | R/W | C1.36 | Sensor signal correction factor for signal < 0. branch 1 | V/V | -100 | 100 | 100 | 100 == Factor 1.00 |
| **0xDB** | 219 | R/W | C2.36 | Sensor signal correction factor for signal < 0. branch 2 | V/V | -100 | 100 | 100 | 100 == Factor 1.00 |
| **0xDC** | 220 | R/W | C1.37 | Spool overlap compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0xDD** | 221 | R/W | C1.38 | Spool overlap compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **0xDE** | 222 | R/W | C2.37 | Spool overlap compensation A | V | 0 | 9999 | 0 | 1000 == 1.000 V |
| **0xDF** | 223 | R/W | C2.38 | Spool overlap compensation B | V | 0 | 9999 | 0 | 9.999 V = max. current depending on solenoid selection |
| **..** | .. | | | | | | | | |
| **0xE6** | 230 | R/W | L1.x0 | Linearization curve branch 1 [0,0] | V | 0 | 0 | 0 | 1000 == 1.000 V |
| **0xE7** | 232 | R/W | L1.y0 | Linearization curve [0,0] | V | 0 | 9999 | 0 | |

| $ID_{Hex}$ | $ID_{dec}$ | W/R | Name | Function | Unit | Min | Max | Def | Description |
|---|---|---|---|---|---|---|---|---|---|
| **0xE8** | 232 | R/W | L1.x1 | Linearization curve [1,1] | V | 0 | 9999 | 1250 | |
| **0xE9** | 233 | R/W | L1.y1 | Linearization curve [1,1] | V | 0 | 9999 | 1250 | |
| **0xEA** | 234 | R/W | L1.x2 | Linearization curve [2,2] | V | 0 | 9999 | 2500 | |
| **0xEB** | 235 | R/W | L1.y2 | Linearization curve [2,2] | V | 0 | 9999 | 2500 | |
| **0xEC** | 236 | R/W | L1.x3 | Linearization curve [3,3] | V | 0 | 9999 | 3750 | |
| **0xED** | 237 | R/W | L1.y3 | Linearization curve [3,3] | V | 0 | 9999 | 3750 | |
| **0xEE** | 238 | R/W | L1.x4 | Linearization curve [4,4] | V | 0 | 9999 | 5000 | |
| **0xEF** | 239 | R/W | L1.y4 | Linearization curve [4,4] | V | 0 | 9999 | 5000 | |
| **0xF0** | 240 | R/W | L1.x5 | Linearization curve [5,5] | V | 0 | 9999 | 6250 | |
| **0xF1** | 241 | R/W | L1.y5 | Linearization curve [5,5] | V | 0 | 9999 | 6250 | |
| **0xF2** | 242 | R/W | L1.x6 | Linearization curve [6,6] | V | 0 | 9999 | 7500 | |
| **0xF3** | 243 | R/W | L1.y6 | Linearization curve [6,6] | V | 0 | 9999 | 7500 | |
| **0xF4** | 244 | R/W | L1.x7 | Linearization curve [7,7] | V | 0 | 9999 | 8750 | |
| **0xF5** | 245 | R/W | L1.y7 | Linearization curve [7,7] | V | 0 | 9999 | 8750 | |
| **0xF6** | 246 | R/W | L1.x8 | Linearization curve [8,8] | V | 0 | 9999 | 9999 | |
| **0xF7** | 247 | R/W | L1.y8 | Linearization curve [8,8] | V | 9999 | 9999 | 9999 | |
| **0xF8** | 248 | R/W | L2.x0 | Linearization curve branch 2 [0,0] | V | 0 | 0 | 0 | 1000 == 1.000 V |
| **0xF9** | 249 | R/W | L2.y0 | Linearization curve [0,0] | V | 0 | 9999 | 0 | |
| **0xFA** | 250 | R/W | L2.x1 | Linearization curve [1,1] | V | 0 | 9999 | 1250 | |
| **0xFB** | 251 | R/W | L2.y1 | Linearization curve [1,1] | V | 0 | 9999 | 1250 | |
| **0xFC** | 252 | R/W | L2.x2 | Linearization curve [2,2] | V | 0 | 9999 | 2500 | |
| **0xFD** | 253 | R/W | L2.y2 | Linearization curve [2,2] | V | 0 | 9999 | 2500 | |
| **0xFE** | 254 | R/W | L2.x3 | Linearization curve [3,3] | V | 0 | 9999 | 3750 | |
| **0xFF** | 255 | R/W | L2.y3 | Linearization curve [3,3] | V | 0 | 9999 | 3750 | |
| **0x100** | 256 | R/W | L2.x4 | Linearization curve [4,4] | V | 0 | 9999 | 5000 | |
| **0x101** | 257 | R/W | L2.y4 | Linearization curve [4,4] | V | 0 | 9999 | 5000 | |
| **0x102** | 258 | R/W | L2.x5 | Linearization curve [5,5] | V | 0 | 9999 | 6250 | |
| **0x103** | 259 | R/W | L2.y5 | Linearization curve [5,5] | V | 0 | 9999 | 6250 | |
| **0x104** | 260 | R/W | L2.x6 | Linearization curve [6,6] | V | 0 | 9999 | 7500 | |
| **0x105** | 261 | R/W | L2.y6 | Linearization curve [6,6] | V | 0 | 9999 | 7500 | |
| **0x106** | 262 | R/W | L2.x7 | Linearization curve [7,7] | V | 0 | 9999 | 8750 | |
| **0x107** | 263 | R/W | L2.y7 | Linearization curve [7,7] | V | 0 | 9999 | 8750 | |
| **0x108** | 264 | R/W | L2.x8 | Linearization curve [8,8] | V | 0 | 9999 | 9999 | |
| **0x109** | 265 | R/W | L2.y8 | Linearization curve [8,8] | V | 9999 | 9999 | 9999 | |
| **..** | .. | | | | | | | | |

Table 11: Parameter list

## 12 Declaration of Conformity

# EC Declaration of Conformity in accordance with EMC Directive 2014/30/EU

**HCS Hydraulic Control Systems GmbH**
**Neuffener Str. 29**
**D-72636 Frickenhausen**

Hereby declares that the product described as follows complies in terms of its design, as well as in the version placed in the stream of commerce by us, with the relevant requirements of the directive. This declaration is void in the event of any change to the product without our written agreement.

| | |
|---|---|
| Product: | Digital Amplifier and Controller Card |
| Intended use | Automation systems (industrial applications) |
| Model: | **DAC-4x(A)-x** |
| Rated voltage: | 24 V DC; SELV, PELV |
| Rated power: | max. 100 VA |
| Protection class: | III |
| Protection degree: | IP00 (IP20 on request) |
| Relevant EU Directive: | EMC Directive 2014/30/EU |
| Applicable EU Standards: | |
| Emissions: | EN 61000-6-4:2007 + A1:2011;<br>EN 61000-6-3:2007 + A1:2011 |
| Immunity: | EN 61000-6-2: 2019 |

Date/manufacturers signature

15.01.2024
Details of signatory: Dipl.-Ing. (FH) Peter Deuschle (General Manager)

Figure 26 : Declaration of conformity

**- End -**