HYDAC ELECTRONIC

Condition Monitoring Unit
CMU 1000

User manual
(Translation of original manual)

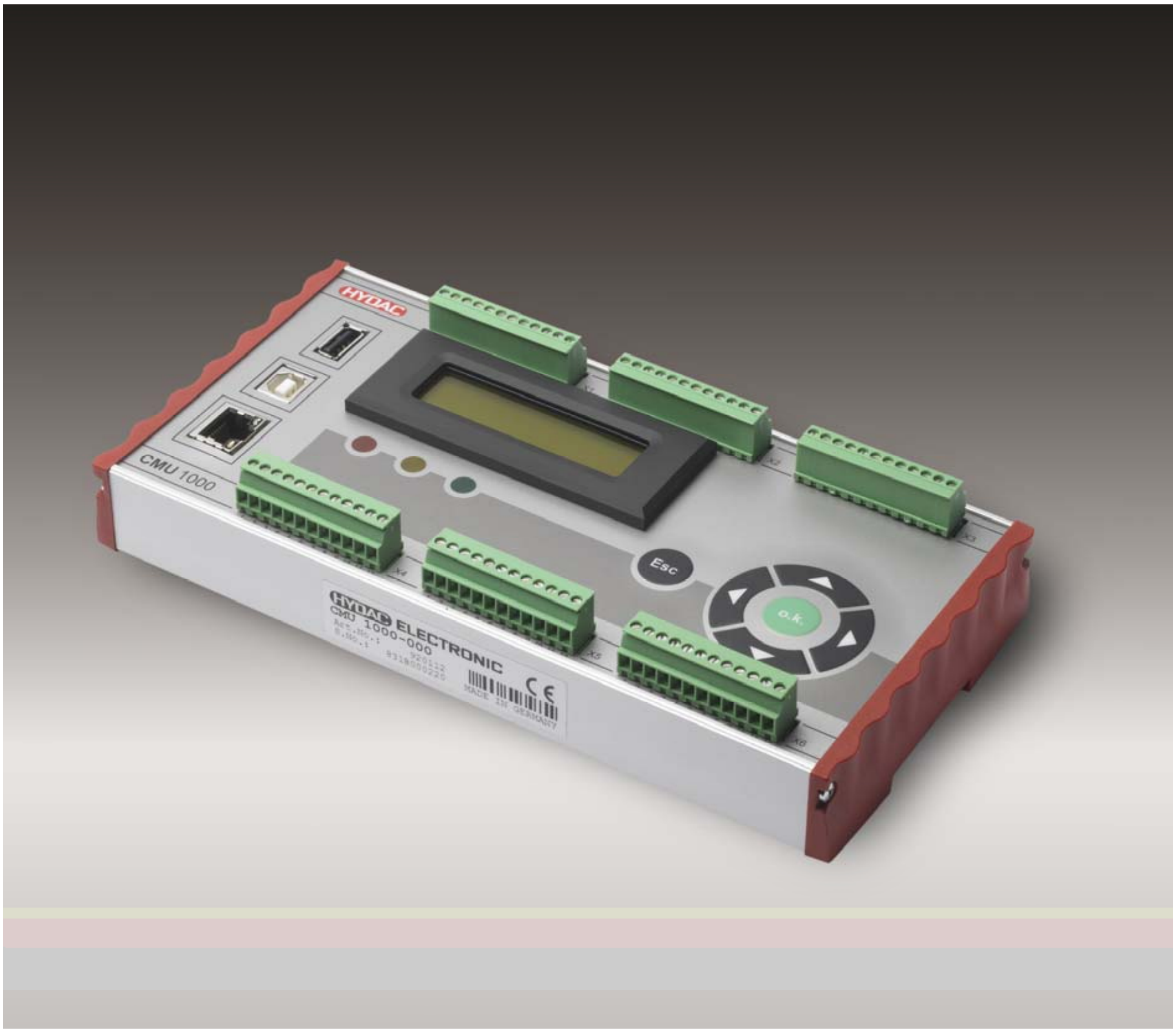


Table of Contents

Preface

We have compiled the most important instructions for the **operation** and **maintenance** of our product for you, its user, in this documentation**.**

It will acquaint you with the product and assist you in using it as intended in an optimal manner.

Keep it in the vicinity of the product so it is always available.
Note that the information on the unit's engineering contained in the documentation was that available at the time of publication. There may be deviations in technical details, figures, and dimensions as a result.

If you discover errors while reading the documentation or have additional suggestions or notes, contact us at:

HYDAC ELECTRONIC GMBH
Technical Documentation
Hauptstraße 27
66128 Saarbrücken
-Germany-
Tel: +49(0)6897 / 509-01
Fax: +49(0)6897 / 509-1726
Email: electronic@hydac.com

The editorial board would welcome your contributions.

„Putting experience into practice“

1 General

This manual is a constituent part of the device. It contains texts and graphics concerning the correct handling of the product and must be read before installation, assembly and the operation of the device.
The manual offers information concerning the safe operation of as well as the installation and programming of the Condition Monitoring Unit CMU 1000. It addresses engineers, programmers, installers and maintenance personnel with general knowledge of the automation technology involved.
Using this manual in the recommended manner will ensure that the goal of effective and operationally safe utilisation of the CMU 1000 is achieved without delay. We are providing you at this juncture with an overview of the following items:

- What previous knowledge must one have in order to be able to progam the CMU 1000?
- How is this manual structured?
- How does one find one's way around in this manual?
- What information is to be found in this manual?

1.1 Previous Knowledge

No special previous knowledge is required for programming the CMU 1000.
It is however to your advantage to have general knowledge in the area of automation technology or memory-programmable controllers, knowledge of control technology or SPS programming knowledge, and having them will shorten the familiarization period.

1.2 Structure of the Manual

We have integrated a variety of different Help functions for the purpose of making the utilization of this manual somewhat easier. Please consult the Table of Contents to find your way to a specific subject. A brief overview is provided at the beginning of each Chapter listing the contents of that particular Chapter.

Selective Reading

You will find notes in the side margins that make it easier to find particular sections. Pictograms and markings also appear, the significance of which will be explained below.

Furthermore, this manual also contains instructions regarding personal safety and the avoidance of property damage that must be observed. The instructions are highlighted by a Warning symbol and displayed as follows, depending on the seriousness of the hazard:

Danger!
means that death, severe bodily injury or considerable property damage will occur if the respective precautionary measures are not implemented.

Warning!
means that death, severe bodily injury or considerable property damage could occur if the respective precautionary measures are not implemented.

Caution!
means that some non-severe bodily injury or property damage could occur if the respective precautionary measures are not implemented.

Attention!
means that an unwanted event or condition could occur if the respective instruction is not followed.

Note!
means an important piece of information about the product, its handling or a part of the documentation to which particular attention should be paid.

In the event that several hazard levels occur simultaneously, it is always the warning notice announcing with the respectively highest level present that will be used. If the warning triangle appears in the warning notice to warn against possible personal injury, then the same warning notice may also contain a warning against possible property damage.

1.3 Copyright Protection

The dissemination and/or reproduction of this document, as well as the exploitation and communication of its content, is not permitted until specifically authorized. Violations of this stipulation will result in liability for damages. All rights reserved.

1.4 Note on Warranty

This manual was compiled with the greatest possible care. Nevertheless, errors or deviations cannot be excluded, for which reason we assume no responsibility for the complete accuracy of the content.
In view of the fact that, despite intensive endeavors, errors can never be completely avoided, we are thankful at all times for tips and suggestions for improvement.

1.5 Declaration of Conformity CE

This product is labelled with the CE Marking and thus is in compliance with currently valid German marketing authorization regulations and European standards. This ensures that applicable guidelines for electromagnetic compatibility and the safety provisions stipulated in the Low Voltage Directive are complied with.
This product is in compliance with the regulations contained in the following European guidelines: EN 61000-6-1 / 2 / 3 / 4

2 Safety

2.1 General Safety Precautions

Follow the specifications contained in this description. Non-observance of the instructions, operation outside of the following intended utilization, incorrect installation/assembly or erroneous handling of the product can lead to severe impairments with respect to the safety of personnel and systems/machines and result in the revocation of warranty and liability claims.

Immediately after unpacking, check the scope of delivery for completeness and the device for proper condition.
The device may not be commissioned or operated except by qualified personnel who can be regarded as being "competent" in the sense of the EMV and Low Voltage Directives.
Qualified personnel are individuals who are authorized to operate, ground and label devices, systems and electrical circuits in accordance with safety technology standards.
All relevant and generally recognized safety technology directives are to be complied with while doing so.

If the voltage supply to the device is not provided by an on-board electrical system V (24 V battery operation), then care must be taken to ensure that the external voltage is generated and routed in accordance with the criteria for secure low voltage (SELV [Separated Extra Low Voltage] pursuant to EN 60950), in view of the fact that this is provided for supplying the connected control system, sensor system and actuating elements without any other additional measures being implemented.
The wiring of all of the signals connected with the SELV circuit in the device must also meet the SELV criteria (safe protective low voltage, securely disconnected galvanically from other electrical circuits).
If the fed-in SELV voltage is grounded externally (PELV pursuant to EN 50178), then responsibility for this and for compliance with any national installation regulations that apply to the site of installation rest with the operator.
All of the statements made in this manual refer to devices which are not grounded in terms of the SELV voltage.
Generally speaking, DIN VDE 0100 Part 410 is to be observed for the supply voltage.

Only the signals which are respectively specified in the Technical Data and/or on the device label may be fed in at the connection terminals; only authorized HYDAC ELECTRONIC GMBH accessory components may be connected to them.

In accordance with the following technical specifications, the device can be operated in a wide range of ambient temperatures. Due to the additional self-heating of the device, high perceptible contact temperatures may develop on the housing in hot environments.

In the event of malfunction or uncertainty, please contact your HYDAC representative.
Unauthorized interventions in the device could result in grave impairments to the safety of persons and systems. These are not permitted and lead to the exclusion of all liability and warranty claims.
Troubleshooting and repairs may only be performed by our HYDAC SERVICE GMBH Customer Service Department.

2.2 Proper/Designated Use

The CMU1000 is an electronic evaluation unit designed for permanent machinery and systems condition monitoring. The device must be supplied for this purpose with machine data that is gathered through the connected sensors. The recorded data (whether processed or unprocessed) can be forwarded by the CMU 1000 through various interfaces to other units and/or monitoring levels. The device can also intervene directly in the machine or system being monitored by means of the integrated analog and digital outputs.

The units of the CMU 1000 family are designed for utilization under difficult conditions (expanded temperature range). They are thus suitable for direct installation in machines in not only stationary but also in mobile and robust utilization.

The specifications for these inputs and outputs mean that they are designed especially for such utilization. Integrated hardware and software functions (operating system) provide a greater level of protection for the machine.

Warning!

The device may be used only for the types of applications specified in the manual and only in connection with accessory components authorized by HYDAC ELECTRONIC GMBH. The trouble-free and safe operation of the product is contingent on proper transport; on proper storage, setup and installation; and on careful operation and maintenance.

The application software, the "CM Program", can be readily generated with the "CM Editor" on a PC by the operator himself/herself. The "CM Editor" is a component part of the HYDAC PC software "CMWIN", starting with Version 3.0.

Note!

All of the programming procedures and software functions subsequently described in this documentation refer to the "CM Editor" in accordance with IEC 61131.

The operator is responsible himself/herself for the safe and application-appropriate functioning of the CM Programs that he or she generates.

2.3 System Configuration

The CMU 1000 is a device concept that is intended for both single operation and utilization in series. This means that the device can be configured in optimal fashion for the respective application case. Special functions and special hardware solutions can also be implemented if required.

Note!

Generally speaking, the following applies:

All of the performances, descriptions and explanations contained in this manual are generally valid for the standard model of the CMU 1000.

A check must be made in each case before the control module is used to determine whether particular functions, hardware options, inputs and outputs described in the documentation are in fact available in the hardware.

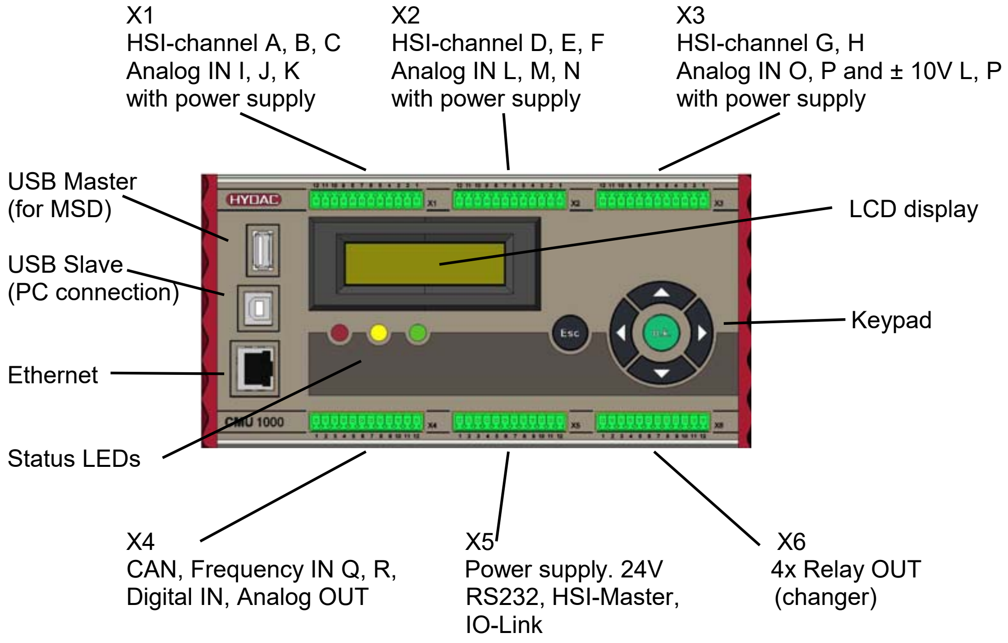
3 Setup and Function

The CMU1000 is an electronic device for regular (permanent) status monitoring of hydraulic systems or machinery.
This procedure is also referred to as *"Condition Monitoring"*.

3.1 Hardware Setup

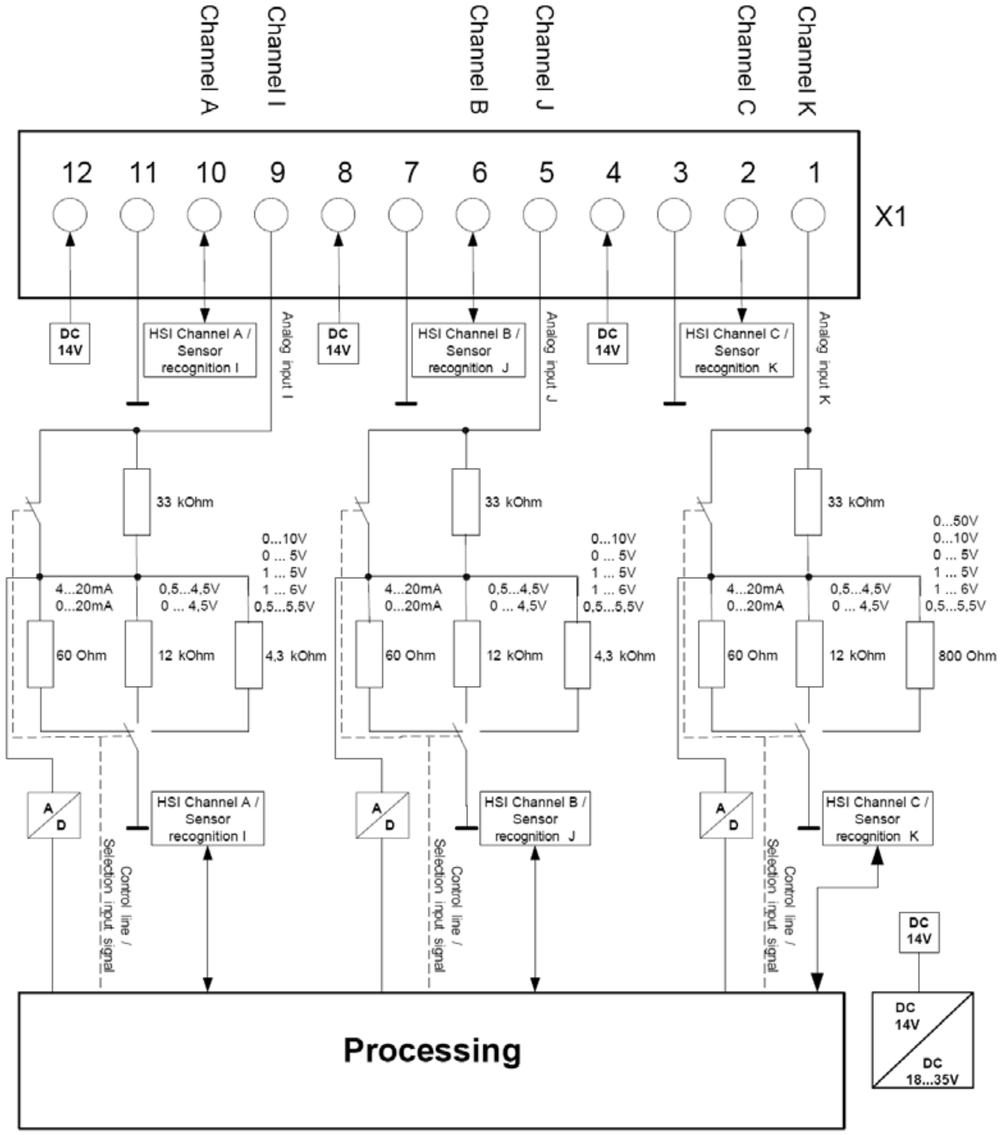
In order to fulfill the aforementioned task, the CMU 1000 must be supplied with relevant machinery and/or systems data, which it receives through connected sensors. The recorded data (whether processed or unprocessed) can be forwarded by the CMU 1000 through various interfaces to other units.
The CMU 1000 is equipped with a background-lit LCD display as well as three different-colored LEDs for the status display and presentation of messages and values.
The entering of data and commands can proceed directly at the device by means of a keypad,
within the specified menu structure, among other ways.

3.2 Control Elements/Connections

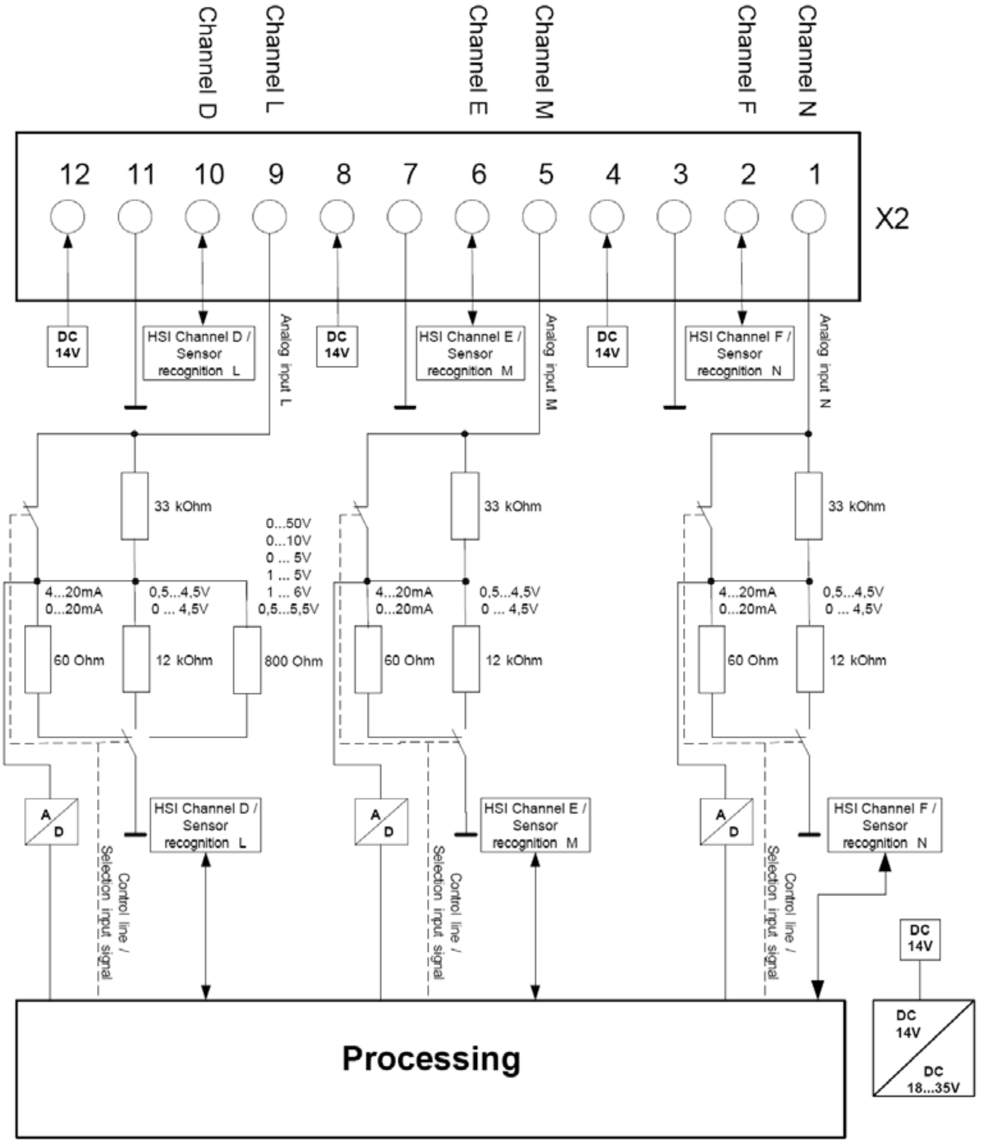


3.3 Terminal Allocations

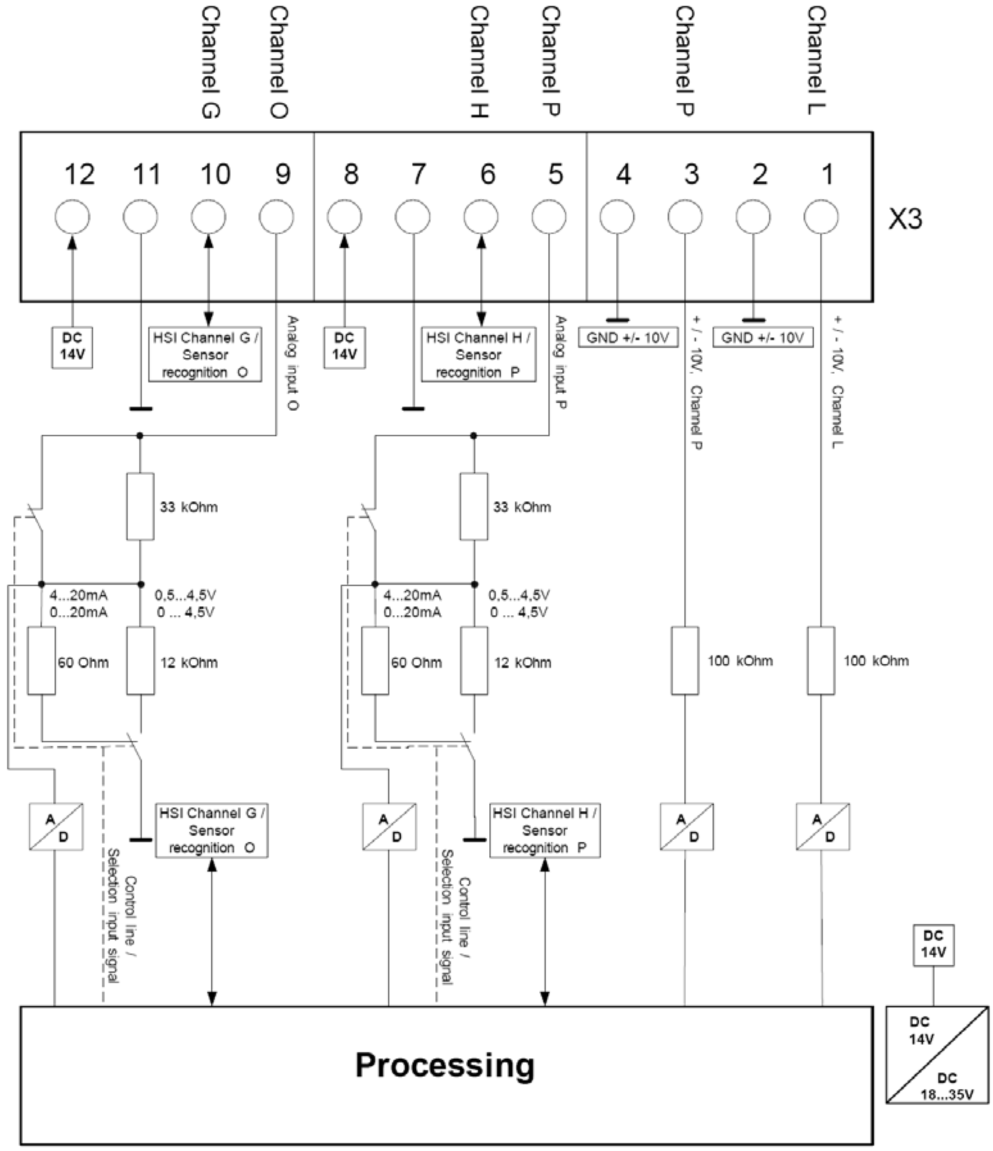
Plugs	Pin	Channel	Function	I/O
X1	1	**K**	Analog input K	IN
	2	**C**	HSI Channel C / Sensor recognition input K	IN / OUT
	3		GND	
	4		Power supply	
	5	**J**	Analog input J	IN
	6	**B**	HSI Channel B / Sensor recognition input J	IN / OUT
	7		GND	
	8		Power supply	
	9	**I**	Analog input I	IN
	10	**A**	HSI Channel A / Sensor recognition input I	IN / OUT
	11		GND	
	12		Power supply	



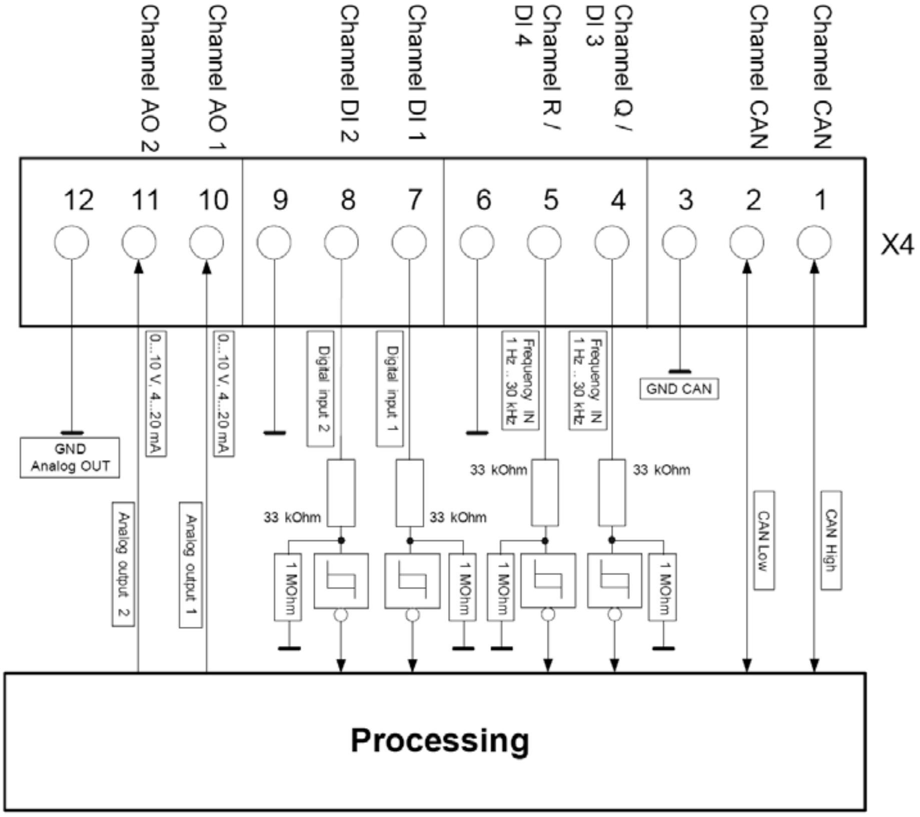
Plugs	Pin	Channel	Function	I/O
X2	1	**N**	Analog input N	IN
	2	**F**	HSI Channel F / Sensor recognition input N	IN / OUT
	3		GND	
	4		Power supply	
	5	**M**	Analog input M	IN
	6	**E**	HSI Channel E / Sensor recognition input M	IN / OUT
	7		GND	
	8		Power supply	
	9	**L**	Analog input L	IN
	10	**D**	HSI Channel D / Sensor recognition input L	IN / OUT
	11		GND	
	12		Power supply	



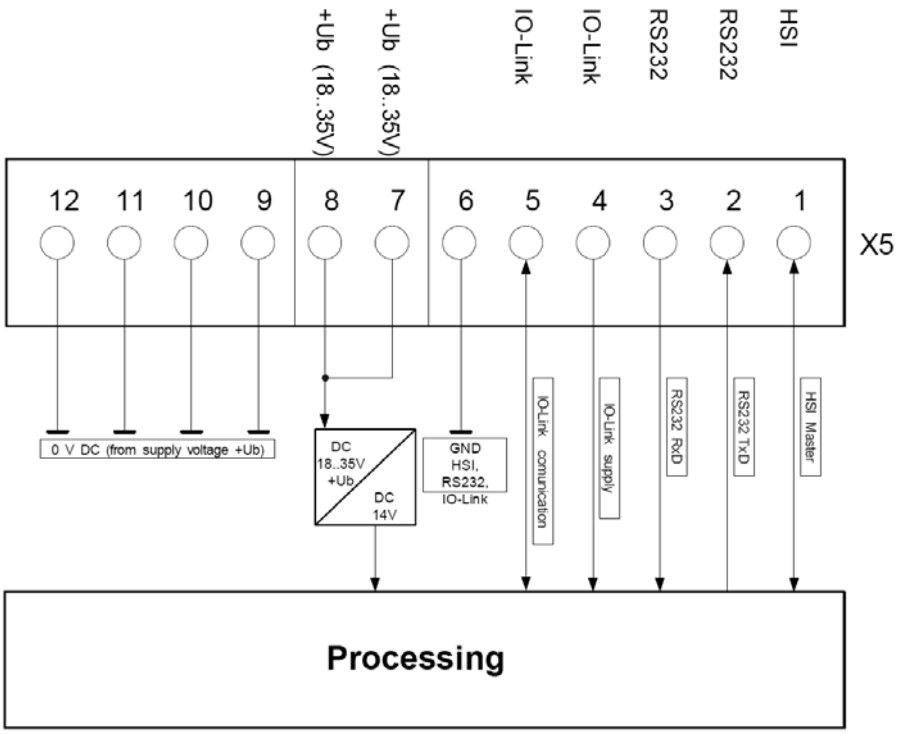
Plugs	Pin	Channel	Function	I/O
X3	1	**L**	+/-10V, Channel L	IN
	2		GND +/-10V	
	3	**P**	+/-10V, Channel P	IN
	4		GND +/-10V	
	5	**P**	Analog input P	IN
	6	**H**	HSI Channel H / Sensor recognition input P	IN / OUT
	7		GND	
	8		Power supply	
	9	**O**	Analog input O	IN
	10	**G**	HSI Channel G / Sensor recognition input O	IN / OUT
	11		GND	
	12		Power supply	



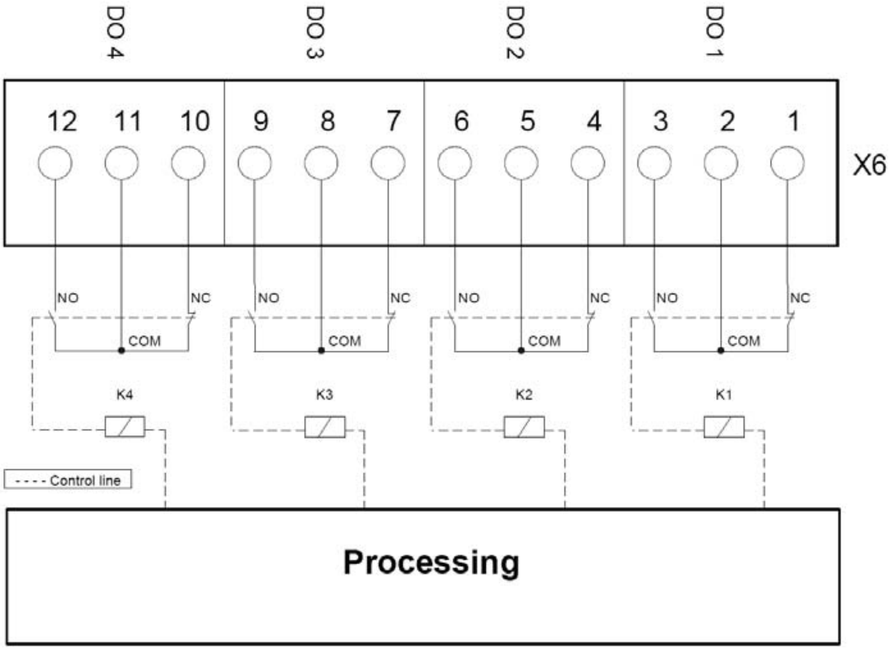
Plugs	Pin	Channel	Function	I/O
X4	1	**CAN**	CAN Bus Low	IN / OUT
	2	**CAN**	CAN Bus High	IN / OUT
	3	**GND**	CAN GND	
	4	**Q**	Frequency input Q	IN
	5	**R**	Frequency input R	IN
	6	**GND**	GND	
	7	**DI 1**	Digital In 1	IN
	8	**DI 2**	Digital In 2	IN
	9	**GND**	GND	
	10	**AO 1**	Analoge out 1	OUT
	11	**AO 2**	Analoge out 2	OUT
	12	**GND**	GND Analoge out	



Plugs	Pin	Channel	Function	I/O
X5	1	**HSI**	HSI Master	IN / OUT
	2	**RS232**	RS232	OUT
	3	**RS232**	RS232	IN
	4	**IO-Link**	Power supply IO-Link	IN
	5	**IO-Link**	IO-Link communication	IN / OUT
	6	**GND**	GND HSI / RS232 / IO-Link	
	7	**+U$_B$**	Power supply +U$_B$	
	8	**+U$_B$**	Power supply +U$_B$	
	9	**0 V DC**	Power supply 0 V DC	
	10	**0 V DC**	Power supply 0 V DC	
	11	**0 V DC**	Power supply 0 V DC	
	12	**0 V DC**	Power supply 0 V DC	

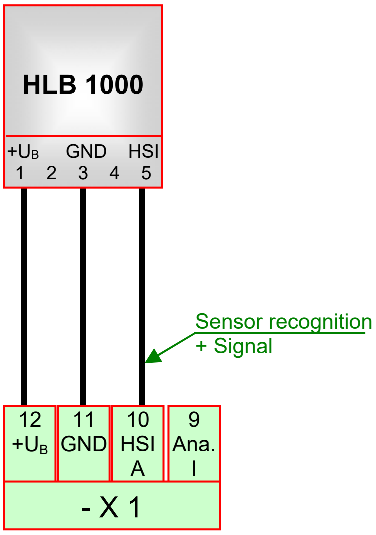


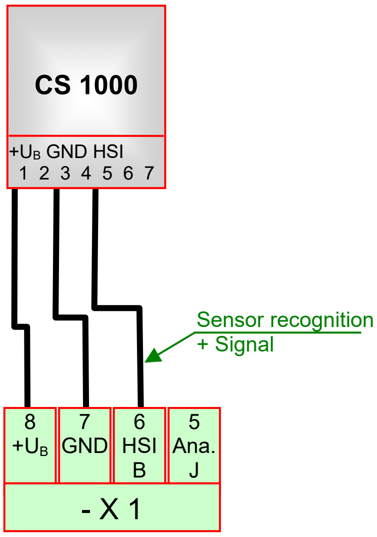
Plugs	Pin	Channel	Function	I/O
X6	1		Relay 1 NC	
	2	**DO 1**	Relay 1 COM	
	3		Relay 1 NO	
	4		Relay 2 NC	
	5	**DO 2**	Relay 2 COM	
	6		Relay 2 NO	
	7		Relay 3 NC	
	8	**DO 3**	Relay 3 COM	
	9		Relay 3 NO	
	10		Relay 4 NC	
	11	**DO 4**	Relay 4 COM	
	12		Relay 4 NO	

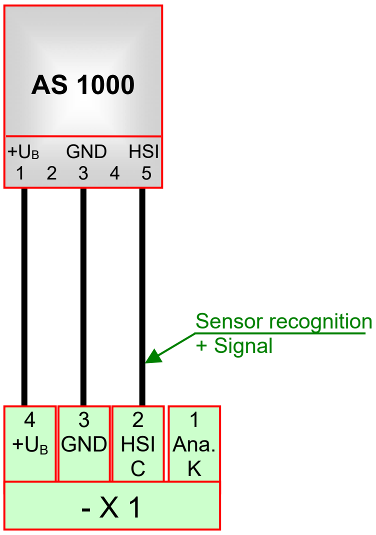


3.4 Examples of Connections

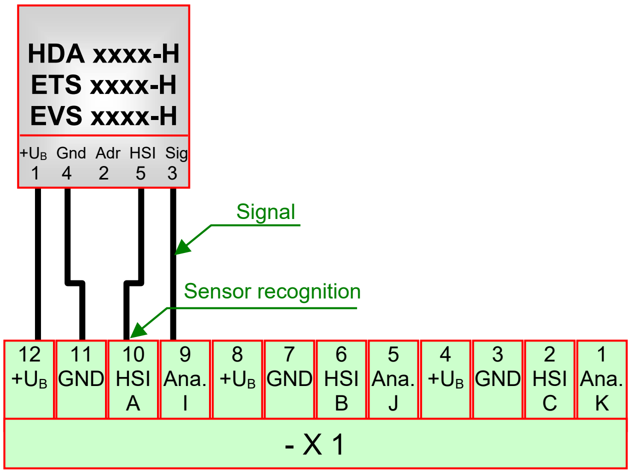
3.4.1 SMART sensors



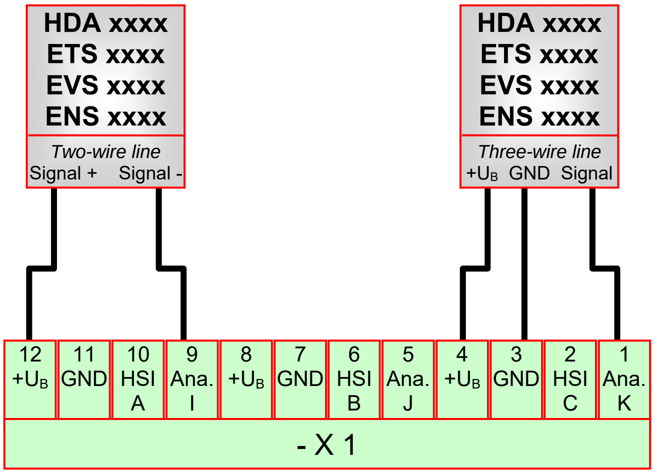




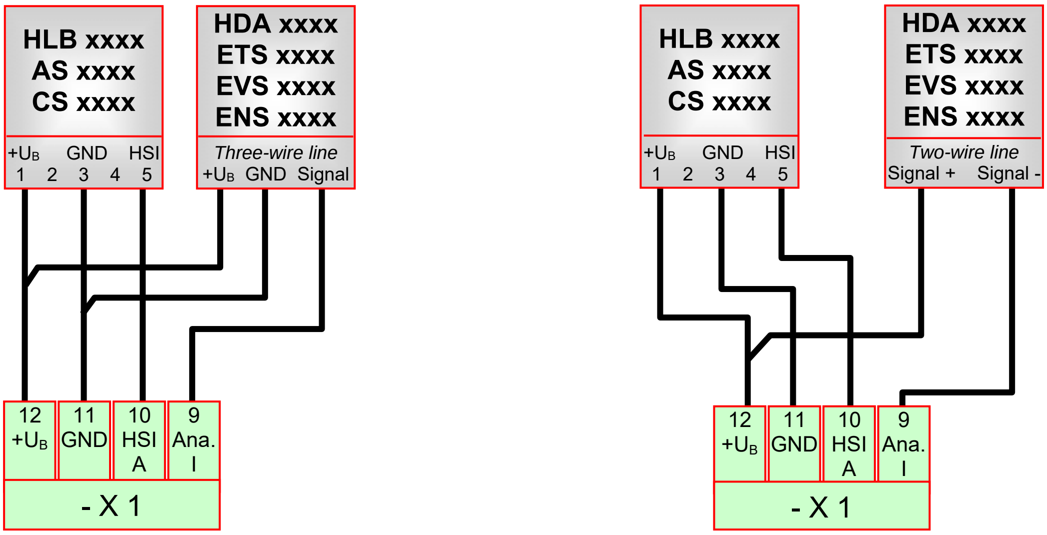
3.4.2 Standard HSI Sensors



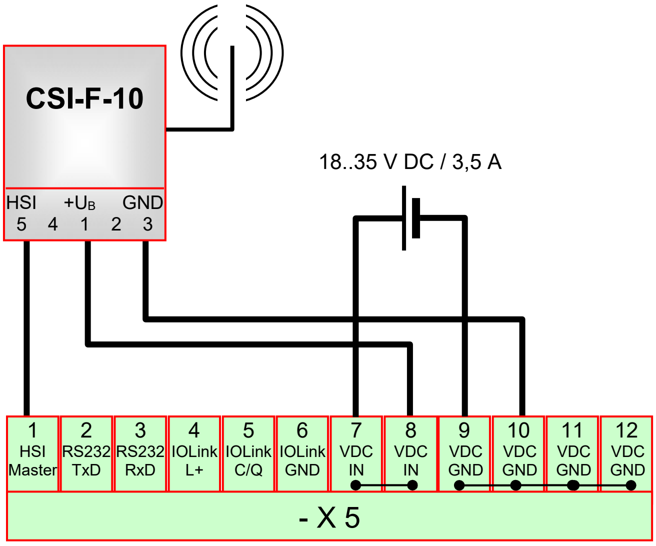
3.4.3 Standard Analog Sensors



3.4.4 SMART Sensors and Standard Analog Sensors



3.4.5 GSM Module CSI-F-10



4 Installation and Initial Operation

4.1 Installation Guidelines

We recommend the installation of the CMU 1000 in a control cabinet or switchbox. It can be mounted on a standard top hat rail either horizontally or vertically.

Caution!
Keep the CMU 1000 devices well away from heat, high voltage and electrical interference from other consumers!

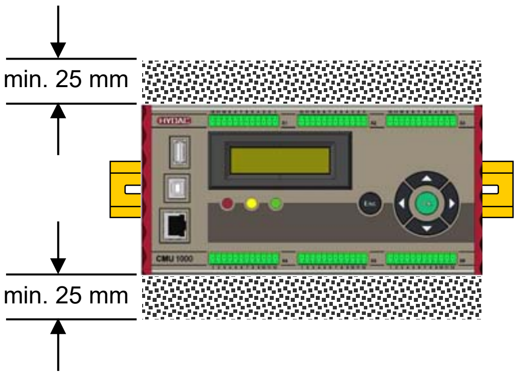
When planning the installation space of the CMU 1000 in your control cabinet, take into account any heat-generating devices present there and reserve cooler areas of the control cabinet for the CMU 1000.
If an electronic device is operated in a high ambient temperature, then this reduces the time interval between breakdowns.

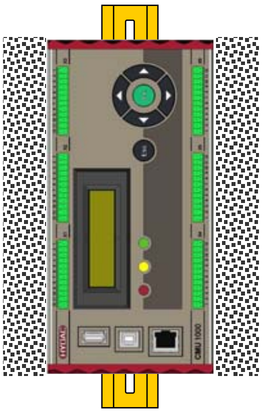
Note!
Allow sufficient clearance between the CMU 1000 and other devices for cooling and wire placement!

The CMU 1000 devices are designed for natural heat dissipation by means of convection. For that reason, allow at least 25 mm of open space both above and below the devices in order to ensure heat dissipation. Also ensure a minimum installation depth of 75 mm.
If the ambient temperature nonetheless still exceeds the maximum authorized operating temperature of the device, then a system providing sufficient cooling (e.g. control cabinet air-conditioning) must be provided for.

Note!
If the CMU 1000 is installed in vertical position, then the maximum permissible ambient temperature is reduced by 10°C!

When planning the layout for your system, leave sufficient clearance around the device both above and below between it and all other devices to allow for the wiring the peripherals and connecting of the communications cable.

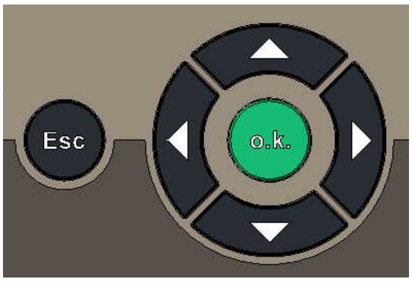




4.2 Control Element on the Device

The following control element is available on the device for operating the CMU 1000 and performing the basic settings:

Keypad

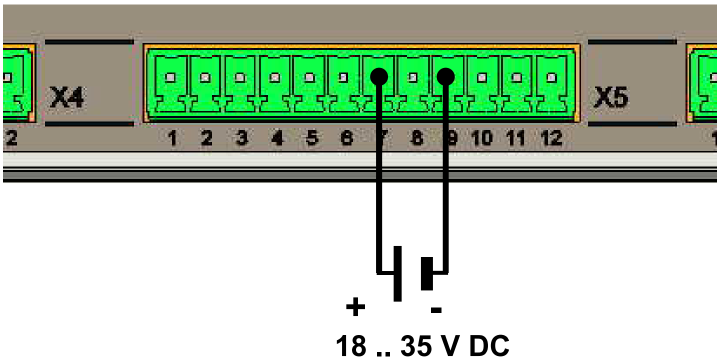


LCD display

4.3 Power Supply Connection

Before installing or removing an electrical device, you must make sure that the voltage supply to the devices is switched off. Implement all of the necessary safety precautions and make sure that the voltage supply to the CMU 1000 is switched off before installation/removal.

Warning!
If you attempt to install or to wire the CMU 1000 and/or connected accessory components while it or they are switched on, this could lead to an electrical shock and/or to device malfunction. Implement all of the necessary safety precautions and make sure that the voltage supply to the CMU 1000 and/or to connected accessory components is switched off prior to installation/removal.

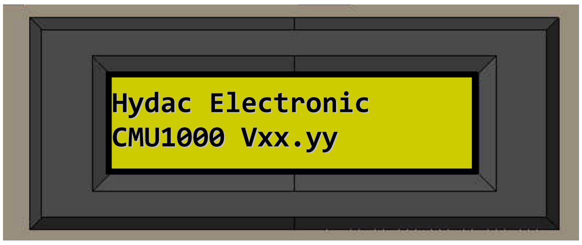


4.4 Behavior when Switching On/Restart

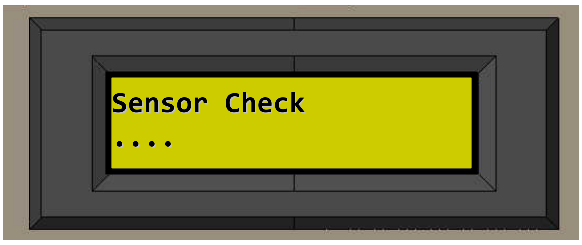
The CMU 1000 is not equipped with a power switch. The behavior of the device following switch-on depends on whether or not a CM Program has been stored in the device memory.

4.4.1 No CM Program Available in the Device

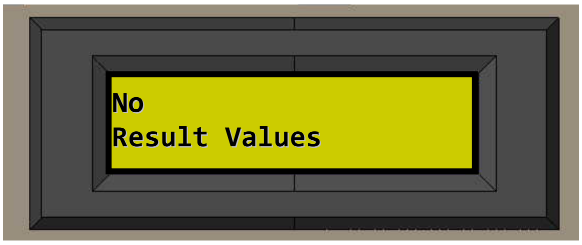
If the CMU 1000 is still in the condition it was at the time of delivery, and if no application program has yet been loaded into the CMU, then the following displays will appear in sequence in the LCD display:



```
Hydac Electronic
CMU1000 Vxx.yy
```



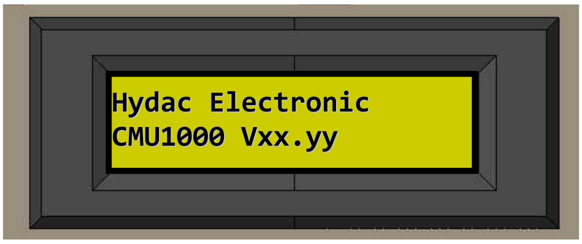
```
Sensor Check
....
```



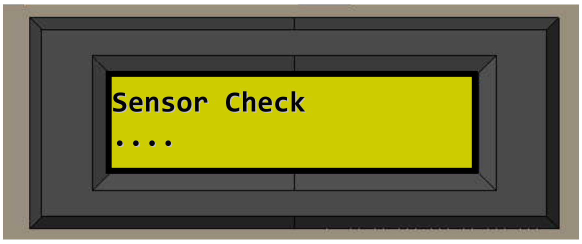
```
No
Result Values
```

4.4.2 CM Program Available in the Device

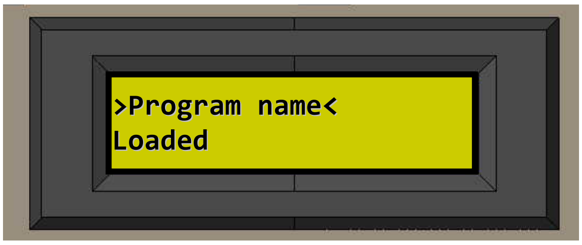
If an application program is stored in the CMU, then this will restart. The sensor configuration and the sensor constellation will also be checked at this time.
The following displays appear in sequence:



```
Hydac Electronic
CMU1000 Vxx.yy
```



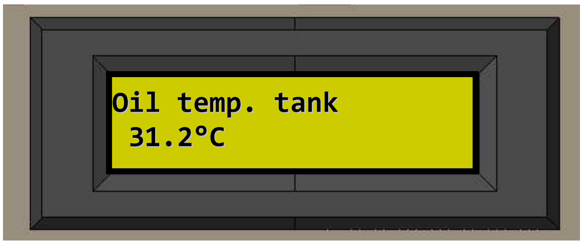
```
Sensor Check
....
```



```
>Program name<
Loaded
```

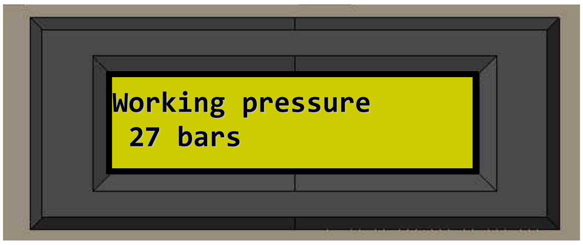
If one or more output values are programmed in the CM Program that is loaded, (see Chap. 7.9.1 and 7.9.2), then the display will jump to the first output value after startup. When several output values have been programmed, you can switch between the individual program values with the ▲ and ▼ key.

Example:



Oil temp. tank
 31.2°C

▲ ▼



Working pressure
 27 bars

Note!
No conditions are registered or saved in the device after switch-off. Exceptions to this are the input parameters (settings), which the user has defined and saved in the menu.

5 Basic Settings/Menu Structure

The CMU 1000 configuration and settings can be carried out in two different modes:

5.1 Configuration on the Device

5.1.1 Menu Structure for Operation on the Device

Level 1	Level 2	Level 3	Level 4	Level 5	Level 6
Input values	Sensor A	*[Display of current measured value]*			
	Sensor B :	*[Display of current measured value]*			
	: Sensor R	*[Display of current measured value]*			
Settings	Basic settings	Name	*[Enter name]*		
		Language	*German* *English* *French*		
		Port	*RS 232 / HSI*		
		Dateformat	*TT.MM.JJ* *MM/TT/JJ* *JJ-MM-TT*		
		Recording settings	Delete record	*Yes / No*	
			Continue record	*Yes / No*	
			Sart new record	*Yes / No*	
		HLB reset	HLB reset	*Yes / No*	
			Sensor	*Sensor A …* *Sensor H*	
		Time settings	Date	*[Enter current date]*	
			Time	*[Enter current time]*	
		Network settings	IP-Address	*[enter IP address]*	
			Subnetmask	*[enter subnetmask]*	
			Gateway	*[enter Gateway address]*	
			MAC-Address	*[MAC address is permanently set at the factory]*	
	CM Program settings	Numerical input values	*[Enter value]*	*The menu option appears only if one or more input functions are used in the CM Program!*	
		Boolean input values	*Yes / No*		
	Peripherals	Channel settings	Channel A : Channel H	Name	*[Enter name]*
			Channel I : Channel P	Name	*[Enter name]*
				Mode	*Automatic* *Off* *Manual*
				Input signal	*HSI* *0 .. 20 mA* *4 .. 20 mA* *0 .. 5 V* *0 .. 10 V* *0,5 .. 4,5 V* *0,5 .. 5,5 V* *1 .. 5 V* *1 .. 6 V*
				Low range	*[Enter value]*

Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**	**Level 6**
Settings	Peripherals	Channel settings	Channel I : Channel P	High range	*[Enter value]*
				Decimal format	*0* *0.0* *0.00* *0.00*
				Unit (of measurement)	*[Enter unit]*
			Channel Q : Channel R	Name	*[Enter name]*
				Mode	*Off* *Active*
				High range	*[Enter value]*
				Factor	*[Enter value]*
				Decimal format	*0* *0.0* *0.00* *0.00*
				Unit (of measurement)	*[Enter unit]*
		Analog outputs	Analog ouput 1	*0 to 20 mA* *4 to 20 mA* *0 to 10 V*	
			Analog ouput 2	*0 to 20 mA* *4 to 20 mA* *0 to 10 V*	

5.1.2 Key Functions during Operation on the Device

The setting parameters and displays listed in the previously shown menu structure can be called up and the corresponding settings and selections can be made with the key pad.

- Pressing the **o.k.** key once takes you out of the measured value display and into the CMU 1000 menu. .
- You can switch between the individual menu options of a level by pressing the keys ▲ or ▼ .
- The next-lower menu level can be accessed by pressing the **o.k.** key once.
- You can go back to the next-higher menu level by pressing the **Esc** key once.
- The selection of the specific setting values (e.g. interface, date format, mode,...) can be made with the ▲ ▼ keys and confirmed by pressing the **o.k.** key.
- To enter name, date and time values, addresses measurement ranges, etc., select the cursor position with the ◄ ► keys.
  Use the ▲ ▼ keys to select the required uppercase and lowercase letters, numbers and special characters.
- Press the **o.k.** key to confirm the completed entry.
- Pressing the **Esc** key aborts the entry without applying the changes.

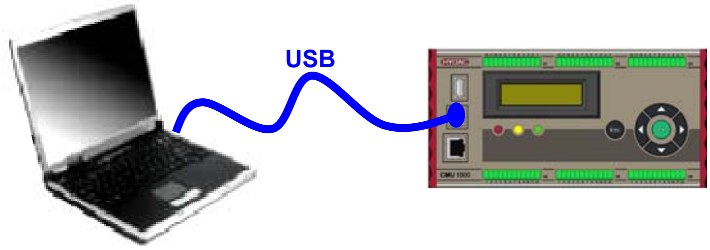
5.2 Configuration Using CMWIN PC Software

The configuration of the CMU 1000 and the carrying out of the basic settings can also be performed from a PC with the "**CM Manager**".
The "**CM Manager**" is a component part of the ***CMWIN*** HYDAC PC software, starting with Version 3, and provides you with various tools and functions for the connecting, configuring, parameterizing and reading-out of CM devices.

The following options are available for setting up the connection with the CMU 1000:

5.2.1 Direct Connection

- First connect your PC with the USB slave port on the CMU 1000

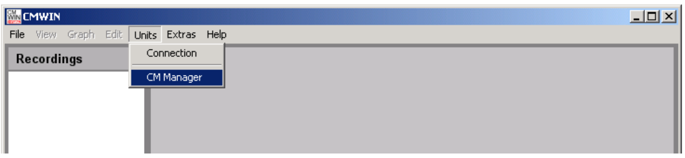


Note!
If the CMU 1000 is being connected for the first time with the PC via USB, then you must first install the HYDAC USB driver „HE-Virtual-Comport-Driver“.
The driver are included on the CD-ROM contained in the scope of delivery.

- Call up the file “***HE-VIRTUAL-COMPORT-INSTALLER.EXE***” in the “***HE-VIRTUAL-COMPORT-DRIVER***” folder and follow the instructions of the “***Setup Wizard***”.

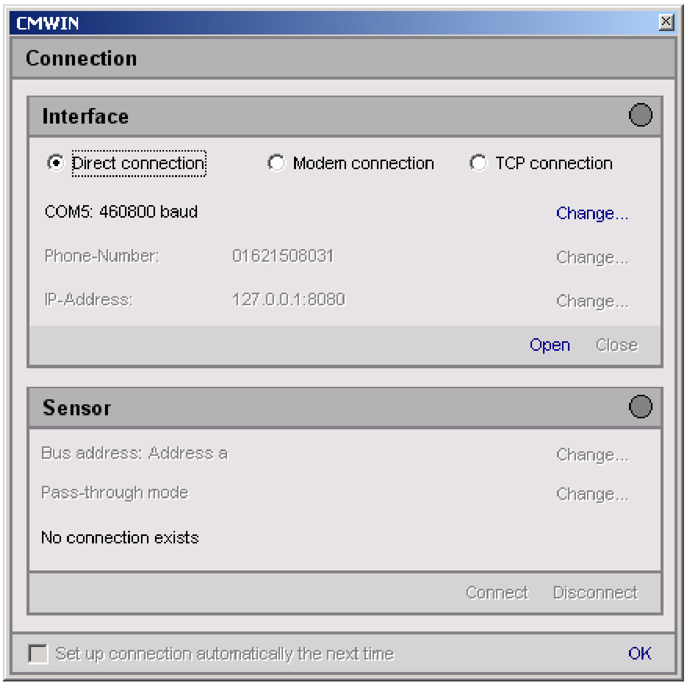
- Start the HYDAC PC software **CMWIN**

- In the ***Units*** Menu, select the **"CM Manager"** option.



- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.

- Mark the option "**Direct Connection**" option in the window that opens.
- Click on "**Change**" to open the window for the interface settings.



CMWIN

Connection

Interface

● Direct connection ○ Modem connection ○ TCP connection

COM5: 460800 baud

Phone-Number: 01621508031

IP-Address: 127.0.0.1:8080

Sensor

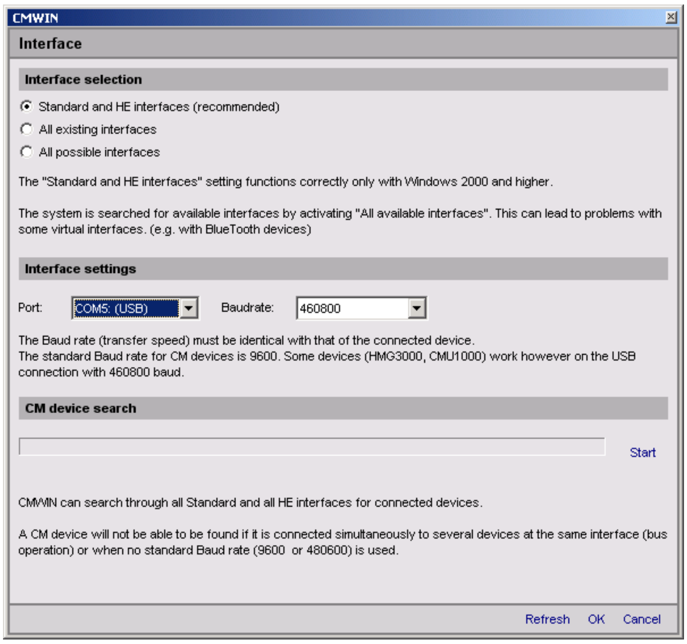
Bus address: Address a

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Make the corresponding preselection for the port settings in the window that opens under ***Interface selection***.
- Select the respective port address and Baud rate under ***Interface settings***.
- As an alternative, you can also search automatically under ***CM device search*** for CM devices connected to the PC by pressing "**Start**".
- Pressing "**Refresh**" causes the interfaces marked under ***Interface selection*** to be refreshed in terms of availability.
- Click on “**OK**“ to apply the modified settings or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.



CMWIN

Interface

Interface selection

● Standard and HE interfaces (recommended)
○ All existing interfaces
○ All possible interfaces

The "Standard and HE interfaces" setting functions correctly only with Windows 2000 and higher.

The system is searched for available interfaces by activating "All available interfaces". This can lead to problems with some virtual interfaces. (e.g. with BlueTooth devices)

Interface settings

Port: COM5: (USB) Baudrate: 460800

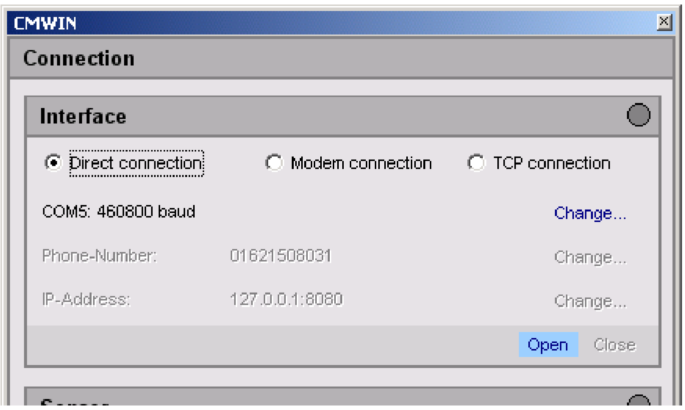
The Baud rate (transfer speed) must be identical with that of the connected device.
The standard Baud rate for CM devices is 9600. Some devices (HMG3000, CMU1000) work however on the USB connection with 460800 baud.

CM device search

CMWIN can search through all Standard and all HE interfaces for connected devices.

A CM device will not be able to be found if it is connected simultaneously to several devices at the same interface (bus operation) or when no standard Baud rate (9600 or 480600) is used.

- In the ***Interface*** field, select the option "**Open**" in order to open the selected interface (COM port).
  The opened interface will be symbolized by a green dot on the right-hand edge of the window.



CMWIN

Connection

Interface

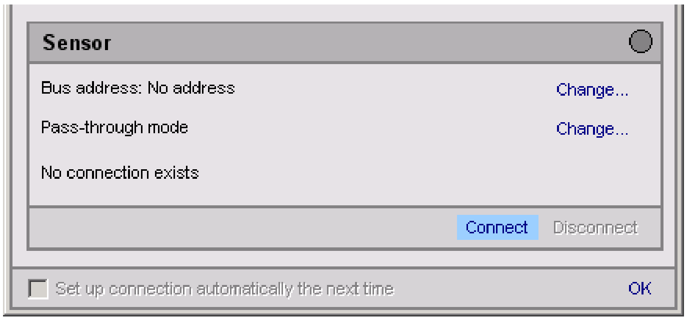
● Direct connection ○ Modem connection ○ TCP connection

COM5: 460800 baud

Phone-Number: 01621508031

IP-Address: 127.0.0.1:8080

- Afterwards click on "**Connect**" in the ***Sensor*** field to connect the CMU 1000 to the PC.
  The successful connection will be symbolized by a green dot on the right-hand edge of the window.



Sensor

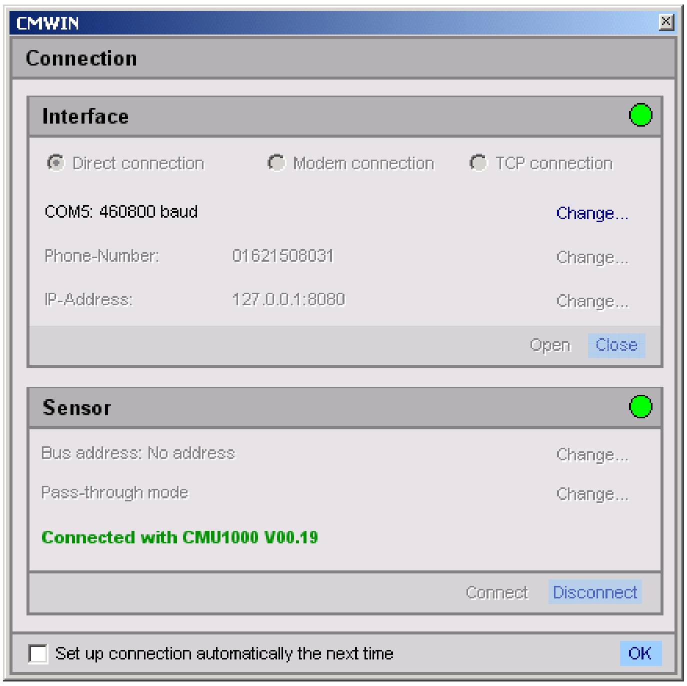
Bus address: No address

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Pressing "**Disconnect**" in the ***Device*** field allows you to interrupt the existing connection between the CMU 1000 and the PC.
- The interface (COM port) used can be closed again on the PC by pressing "**Close**" in the ***Interface*** field.
- At the end you also have the option of selecting an automatic connection setup. Placing a checkmark in the box for "**Set up connection automatically next time** " causes the CMWIN software to set up a connection automatically with the CMU 1000 that is linked via USB during startup. To ensure that this happens, no changes should be made to the interface parameter settings after the currently existing connection is disconnected.
- Click on "**OK**" to complete the connection setup and to return to the *CM Manager*.



COM5: 460800 baud

Phone-Number: 01621508031

IP-Address: 127.0.0.1:8080

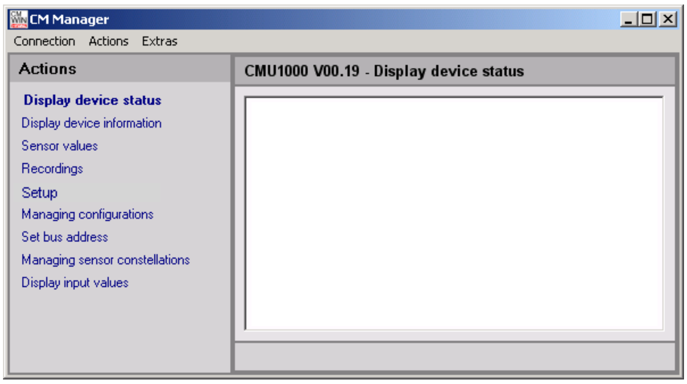
Bus address: No address

Pass-through mode

Connected with CMU1000 V00.19

□ Set up connection automatically the next time

- The following window opens after the connection has been successfully established:



The menu structure and window properties of the CM Manager are explained below in greater detail in Chapter 5.2.5 ff.

5.2.2 Direct Connection via HSI Bus

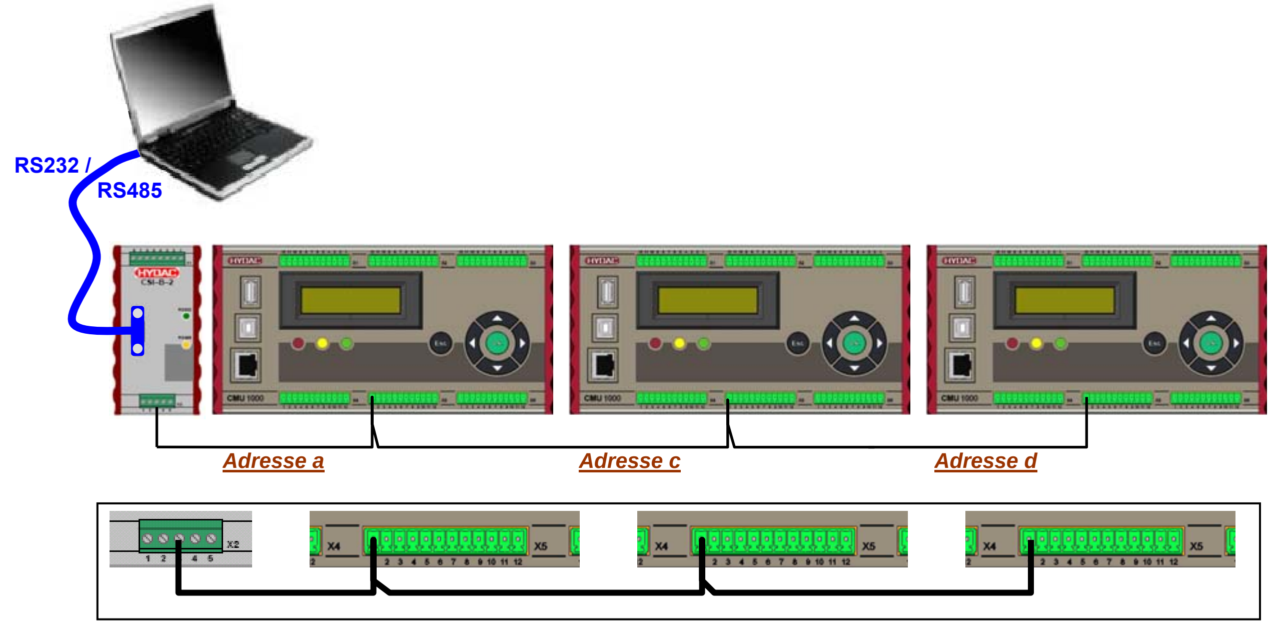
You can connect several HYDAC CM devices with one another (maximum of 26 devices) via the so-called "**HSI Bus**". Each CM device must be assigned an **HSI Bus address** for this purpose (see Chap. 5.2.5.7).

This kind of bus setup is presented below, using as an example three CMU 1000 devices. Other HYDAC CM devices such as SMART sensors (e.g. HYDACLab®, CS 1000) or CSI modules can however also be linked in random sequence and configuration in one HSI Bus.

Caution!
If you connect devices with one another that do not have the same reference voltage, this can cause unwanted currents in the HSI connection cable.
These unwanted currents can lead to communication errors or property damage in the devices.
Make sure that all of the devices that you connect with one another via HSI Bus either have the same reference wire in the electricity circuit or are electrically disconnected, so that no unwanted currents can arise.

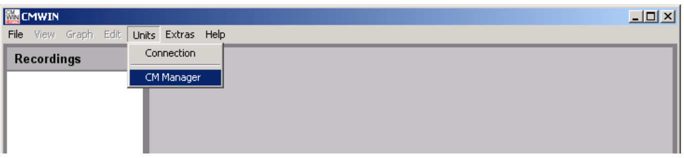
5.2.2.1 Device Connection via CSI-B-2 Interface Module

- Connect the serial interface of your PC (RS232 or RS485) with the 9-pin SUB-D socket of the HYDAC interface module CSI-B-2 via a corresponding data cable (interface module and data cable are not included in the scope of delivery for the CMU 1000).
- Connect all of the devices via the "HSI Master" connection.
  (-X2/Pin 3 on the CSI-B-2 and -X5/Pin 1 on the CMU 1000)
- Switch on the voltage supplies for all of the connected devices.

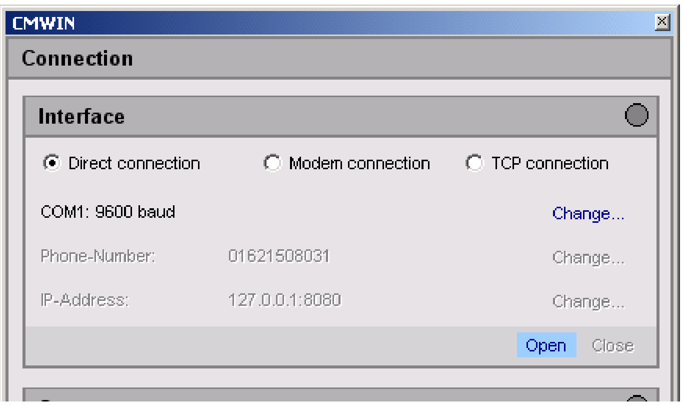


5.2.2.2 Connection Setup via CSI-B-2 Interface Module

- Start the HYDAC PC software **CMWIN**
- In the ***Units*** Menu, select the **"CM Manager"** option.



- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.
- Mark the option "**Direct Connection**" option in the window that opens.
- Click on "**Change**" to open the window for the interface settings.



CMWIN

Connection

Interface ○

● Direct connection ○ Modem connection ○ TCP connection

COM1: 9600 baud Change...

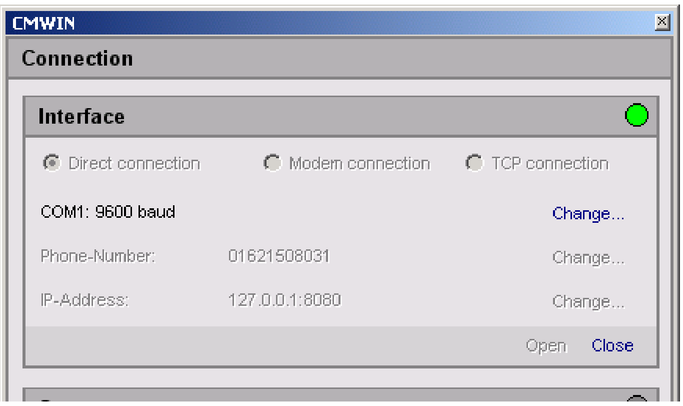
Phone-Number: 01621508031 Change...

IP-Address: 127.0.0.1:8080 Change...

Open Close

- Make the corresponding preselection for the port settings in the window that opens under ***Interface selection***.
- Select the respective port address and Baud rate under ***Interface settings*** (RS232 or RS485 with 9600 Baud).
- Pressing "**Refresh**" causes the interfaces marked under ***Interface selection*** to be refreshed in terms of availability.
- Click on “**OK**“ to apply the modified settings or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.

- Click on ***Open*** to open the selected interface. The opened interface is indicated by a green dot on the upper right.



CMWIN

Connection

Interface 🟢

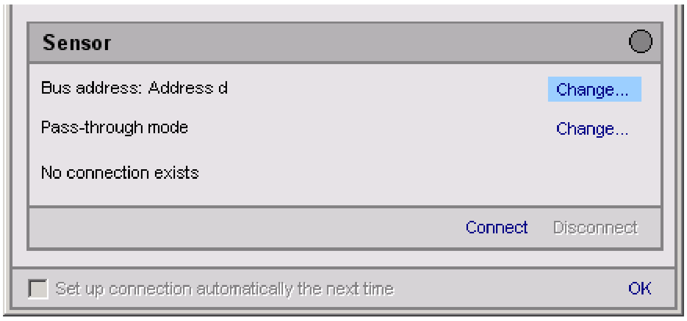
● Direct connection ○ Modem connection ○ TCP connection

COM1: 9600 baud Change...

Phone-Number: 01621508031 Change...

IP-Address: 127.0.0.1:8080 Change...

Open Close



Sensor

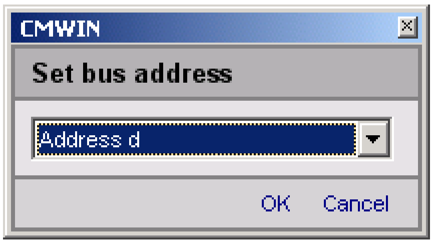
Bus address: Address d

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Select ***Change*** in the ***Bus address*** line. The following window opens:

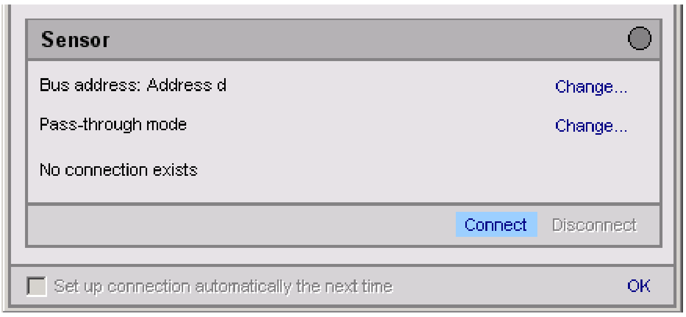


CMWIN

Set bus address

Address d

- Select the respective device address in the selection window (*Address d* in our example).
- Confirm this with ***OK***.



Sensor

Bus address: Address d

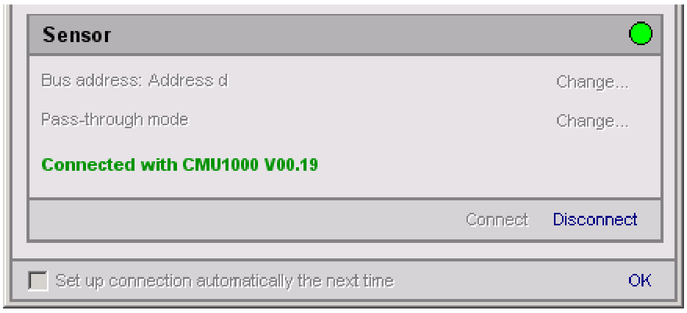
Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Afterwards click on ***Connect*** to connect the PC to the CMU 1000 (Address d).

- The successful establishment of the connection will be signalled as shown below:



Sensor

Bus address: Address d

Pass-through mode

Connected with CMU1000 V00.19

□ Set up connection automatically the next time

- End the connection setup by confirming with ***OK***.

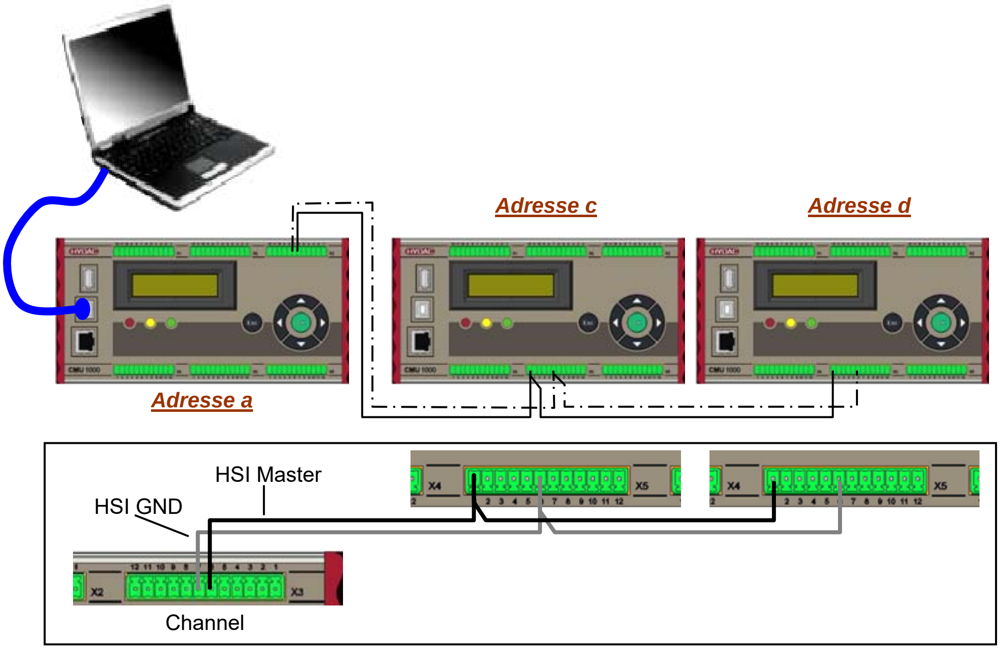
5.2.2.3 Device Connection without CSI-B-2 Interface Module

As an alternative, you can also set up an HSI Bus without a CSI-B-2 interface module and access the individual devices from the PC. For this you will need a CMU 1000 for communications with the PC, which in such cases functions as an "HSI Master".

- First establish which CMU 1000 is the "Master" for the HSI Bus, i.e. at which CMU 1000 the PC will be connected.
  The other CMU devices are connected to this device as "Slaves". Because of the fact that the "Master" treats all of the other devices on the HSI Bus as SMART sensors, the first "Slave" must be connected to one of the eight HSI channels of the CMU 1000 (Channels A ... H) (see Chap. 3.3).
- Connect your PC via USB with the "Master" CMU 1000

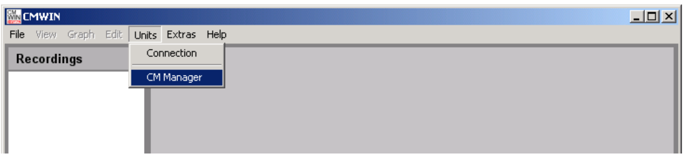
- All "Slave devices" are connected via the "HSI Master" connection (-X5/Pin 1 and Pin 6) to the HSI Bus and linked with one another.

Example:

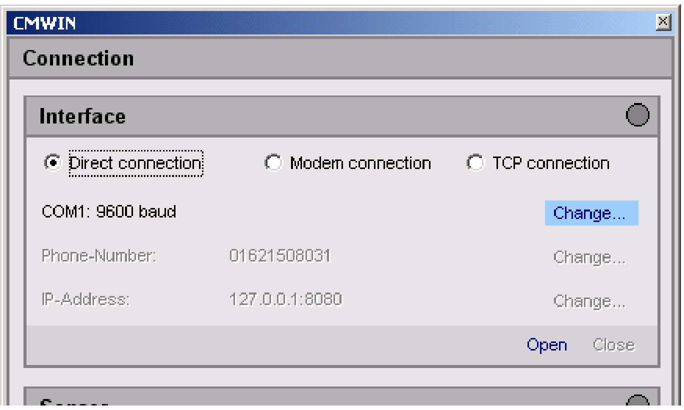


5.2.2.4 Connection Setup without CSI-B-2 Interface Module

- Start the HYDAC PC software **CMWIN**
- In the ***Units*** Menu, select the **"CM Manager"** option.



- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.
- Mark the option "**Direct Connection**" option in the window that opens.
- Click on "**Change**" to open the window for the interface settings.



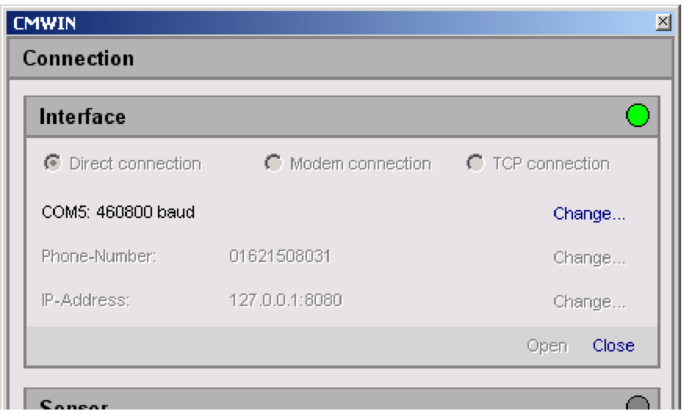
◉ Direct connection

COM1: 9600 baud

Phone-Number: 01621508031

IP-Address: 127.0.0.1:8080

- Make the corresponding preselection for the port settings in the window that opens under ***Interface selection***.
- Select the respective port address and Baud rate under ***Interface settings*** (USB with 460,800 Baud).
- Pressing "**Refresh**" causes the interfaces marked under ***Interface selection*** to be refreshed in terms of availability.
- Click on “**OK**“ to apply the modified settings or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.



◉ Direct connection

COM5: 460800 baud

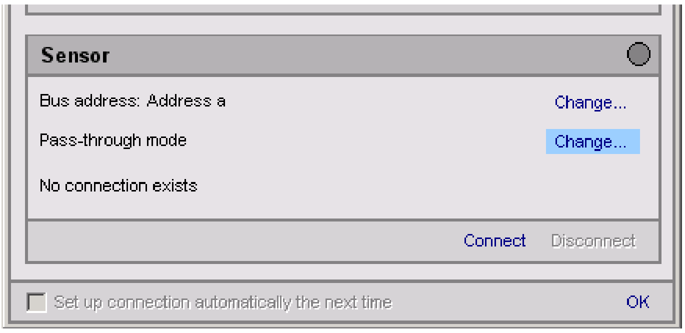
Phone-Number: 01621508031

IP-Address: 127.0.0.1:8080

- Click on ***Open*** to open the selected interface. The opened interface is indicated by a green dot on the upper right.

- To set up a connection with the "Master" CMU, simply click on ***Connect*** and then ***OK***.

- To set up a connection with one "Slave" CMU (e.g. Address d), proceed as follows:



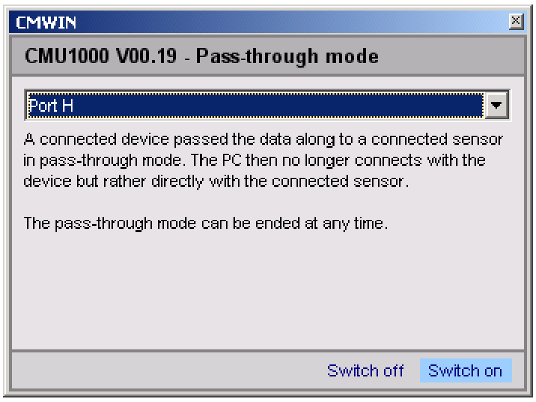
Sensor

Bus address: Address a

Pass-through mode

No connection exists

- Select ***Change*** in the ***Pass-through mode*** line. The following window opens:



CMU1000 V00.19 - Pass-through mode

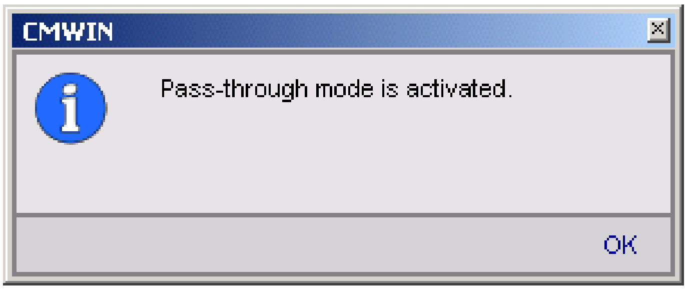
Port H

A connected device passed the data along to a connected sensor in pass-through mode. The PC then no longer connects with the device but rather directly with the connected sensor.

The pass-through mode can be ended at any time.

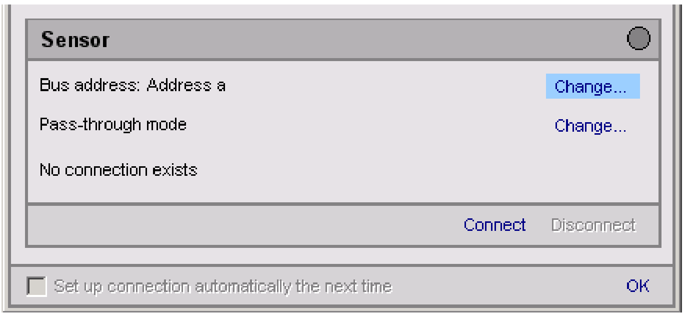
- Select the HSI channel in the selection window to which the slave devices are connected (in our example, Port H at the "Master" CMU).
- Afterwards, click on ***Switch on*** in order to switch on the pass-through mode for the selected channel.

- The following message appears:



Pass-through mode is activated.

- Confirm this with ***OK***.



Sensor

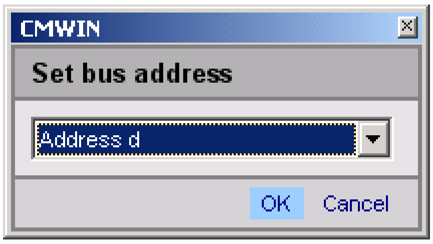
Bus address: Address a

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Select ***Change*** in the ***Bus address*** line. The following window opens:

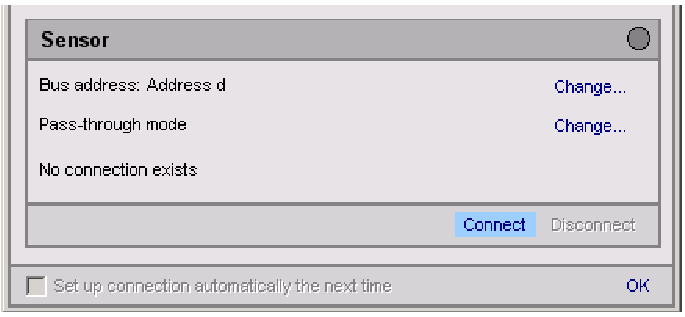


CMWIN

Set bus address

Address d

- Select the respective device address in the selection window (Address d in our example).
- Confirm this with ***OK***.



Sensor

Bus address: Address d

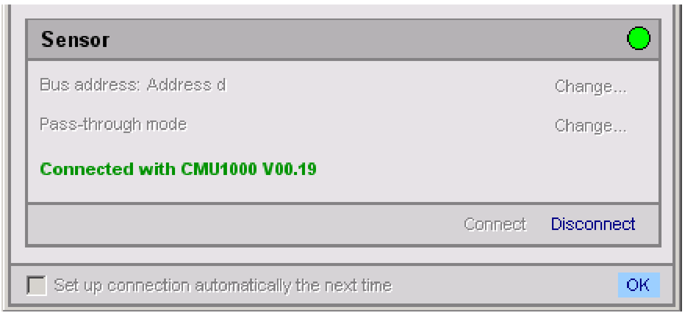
Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Afterwards click on ***Connect*** to connect the PC to the Slave CMU (Address d).

- The successful establishment of the connection will be signalled as shown below:



Sensor

Bus address: Address d

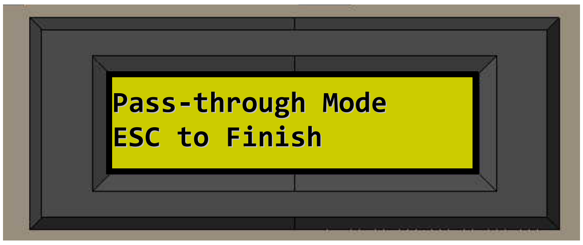
Pass-through mode

Connected with CMU1000 V00.19

□ Set up connection automatically the next time

- End the connection setup by confirming with ***OK***.

- The following message now appears in the Master CMU display:



```
Pass-through Mode
ESC to Finish
```

The message will remain in the display for as long as the "Master" CMU continues to be operated in pass-through mode.
The pass-through mode can also be switched off on the CMU itself (instead of by means of the *Connection* Menu in the CM Manager) by pressing the ***Esc*** key on the device.

Note!
Error messages can occur if the pass-through mode is switched off at the "Master" CMU (either via software or directly on the device), because the signals to the channel to which the "Slave" devices are connected can no longer be evaluated under certain circumstances.

5.2.3 Modem Connection

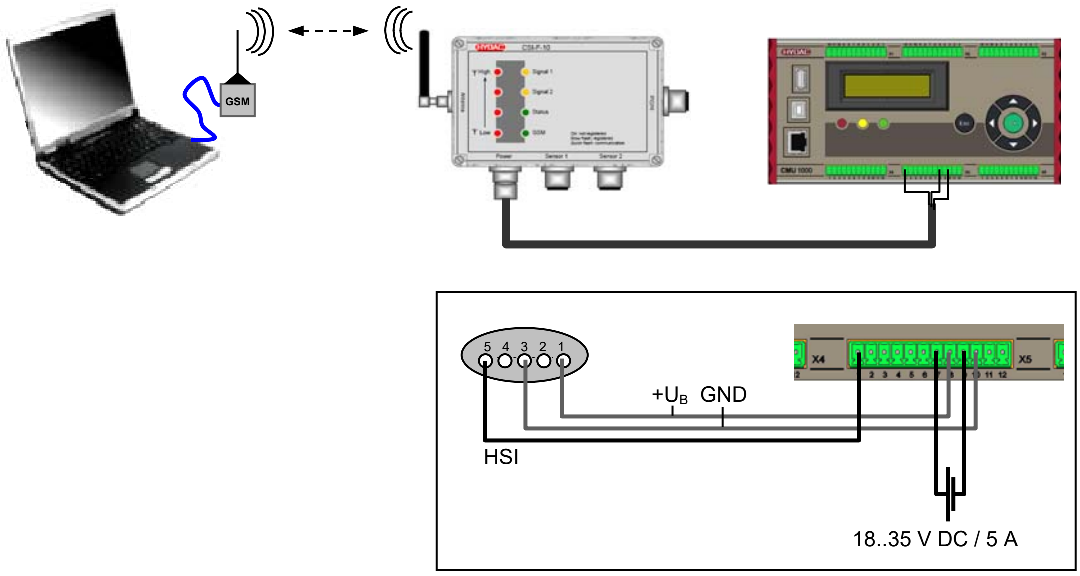
You also have the option of setting up a connection by means of the Standard GSM mobile radio network.
In the following we present an example of this kind of communication link.

Warning!
The GSM data service is not available in all countries and is not supported by all mobile phone operators.
Please contact your service provider for further information.
This information does not apply to text messages!

5.2.3.1 Device Connection/Pin Connections

Connect a Standard GSM modem to your PC and connect the CMU 1000 to the HYDAC ***GSM radio module CSI-F-10*** in accordance with the diagram.

Example:

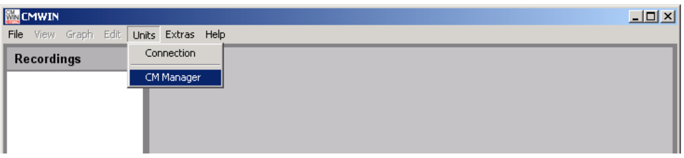


Note!
In order to be able to communicate with the CMU 1000 via the connected CSI-F-10 GSM radio module, this must first be configured. This means that the mobile phone numbers which are authorized for access must be stored in the CSI-F-10 and appropriate permissions assigned.

To configure the GSM radio module CSI-F-10, or to make changes in a configured device, first connect directly with the GSM radio module CSI-F-10 as described below.

5.2.3.2 Establishing Connection with GSM Radio Module CSI-F-10

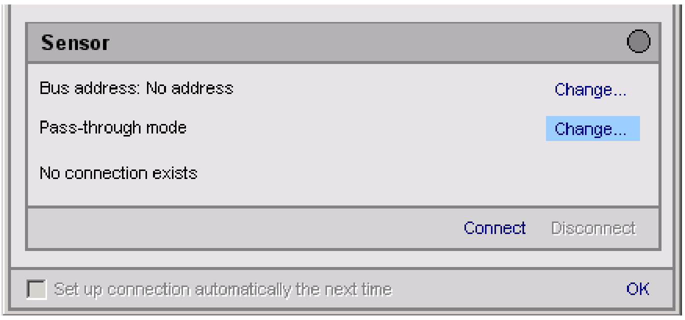
- Connect the CMU 1000 for configuring the CSI-F-10 GSM radio module to your PC as also described in *Chapter 5.2.1 Direct Connection, Chapter 5.2.2 Direct Connection via HSI Bus or Chapter 5.2.4 TCP Connection*.



- Start the HYDAC PC software **CMWIN**
- In the ***Units*** Menu, select the **"CM Manager"** option.

- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.

- Under ***Interface Settings*** make the settings necessary for the available connection type (for configuring the CSI-F-10).
- Click on "**OK**" to apply the settings or "**Cancel**" to discard these changes. In either case you will then return to the ***Connection*** window.
- Click on ***Open*** to open the selected interface. The opened interface is indicated by a green dot at the top right.



Sensor ○

Bus address: No address Change...

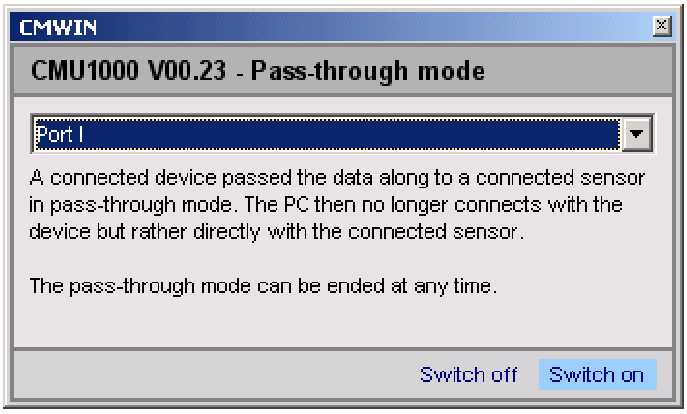
Pass-through mode Change...

No connection exists

Connect Disconnect

□ Set up connection automatically the next time OK

- Select "**Change**" in the ***Pass-through mode*** line. The following window opens:



CMWIN

CMU1000 V00.23 - Pass-through mode

Port I

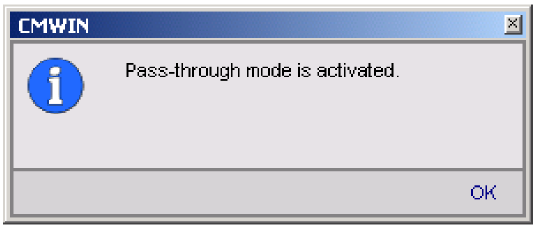
A connected device passed the data along to a connected sensor in pass-through mode. The PC then no longer connects with the device but rather directly with the connected sensor.

The pass-through mode can be ended at any time.

Switch off Switch on

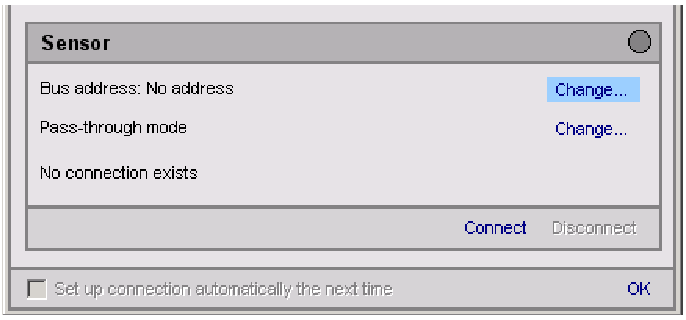
- In the drop-down menu, select "***Port I***" and then click on ***Switch on*** in order to switch on the pass-through mode.

- The following message appears:



CMWIN

Pass-through mode is activated.



Sensor

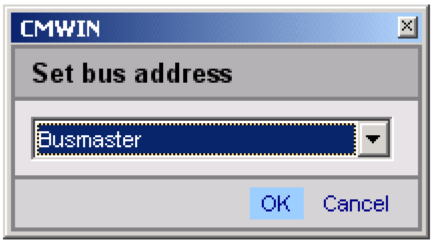
Bus address: No address

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Select "***Change***" in the ***Bus address*** line. The following window opens:

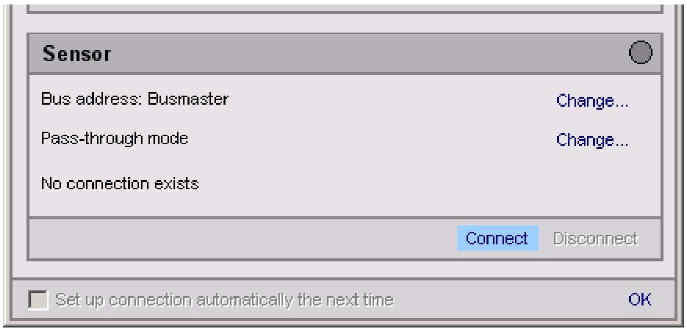


CMWIN

Set bus address

Busmaster

- In the selection window, select "***Busmaster***" and then click on ***OK*** to apply the address.



Sensor

Bus address: Busmaster

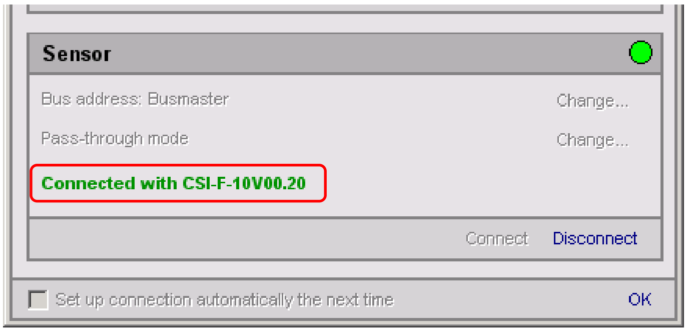
Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Then click on "***Connect***" to connect the PC to the CSI-F-10 GSM radio module (address busmaster).

- Successful establishment of the connection will be signaled as shown below:



Sensor ●

Bus address: Busmaster Change...

Pass-through mode Change...

Connected with CSI-F-10V00.20

Connect Disconnect

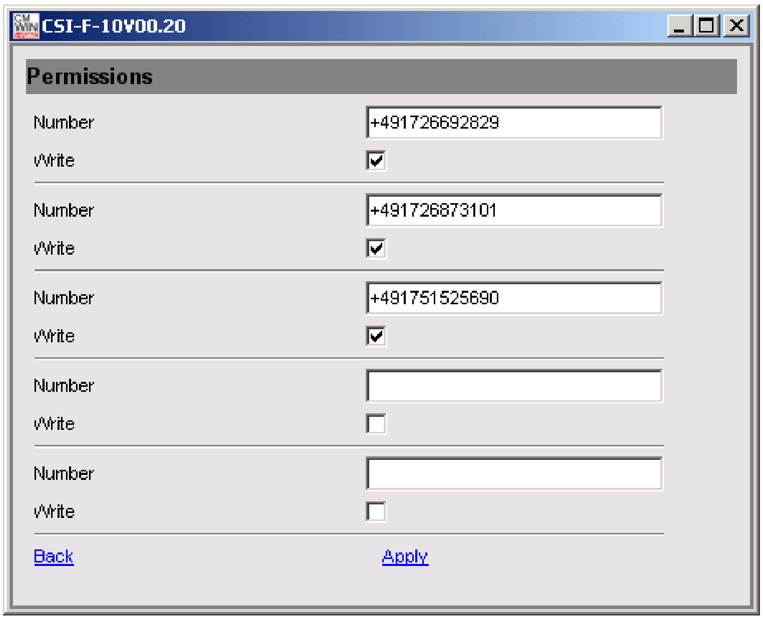
□ Set up connection automatically the next time OK

- End the connection setup by confirming with ***OK***.

- In the CM Manager under ***Actions / Setup / Permissions***, open the following input window and input the mobile phone number authorized for accessing the CMU 1000.

- **Permissions**

Number	[*input telephone number with country code*]
Write	[allow *written access*]



CSI-F-10V00.20

Permissions

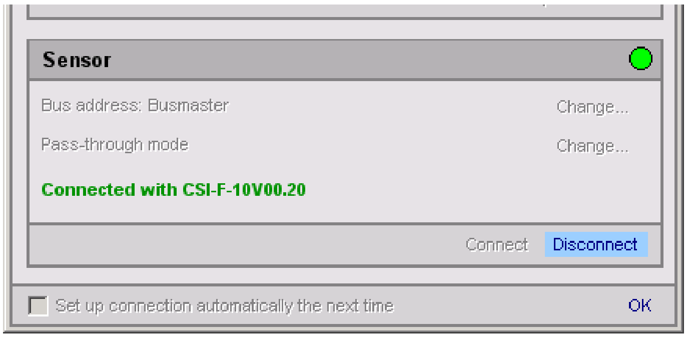
Number	+491726692829
Write	☑
Number	+491726873101
Write	☑
Number	+491751525690
Write	☑
Number	
Write	□
Number	
Write	□

Back Apply

You can input up to five telephone numbers to which the CSI-F-10 may send messages and from which the device may receive enquiry text messages.
By placing a checkmark in the "***Text"*** box, you are also allowing access by text to the CSI-F-10 from this telephone number (change settings, transfer CM program, update firmware, ...)

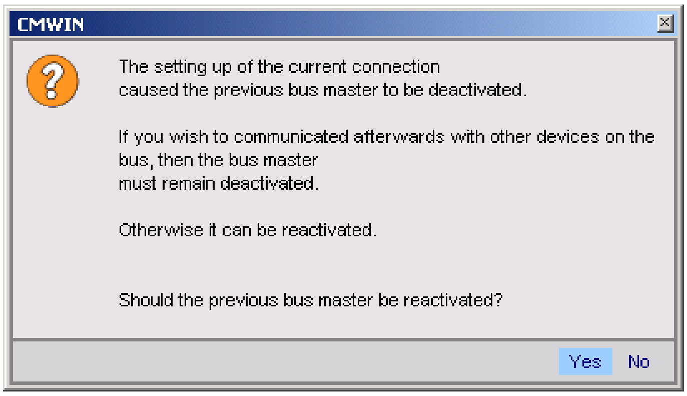
Click on "***Apply***", to apply the settings. "***Back***" takes you back to the main menu.

- Click on "**Disconnect**" under Connection in the ***Device*** box to break the existing connection with the CSI-F-10.



Connected with CSI-F-10V00.20

- The following window opens:



The setting up of the current connection
caused the previous bus master to be deactivated.

If you wish to communicated afterwards with other devices on the bus, then the bus master
must remain deactivated.

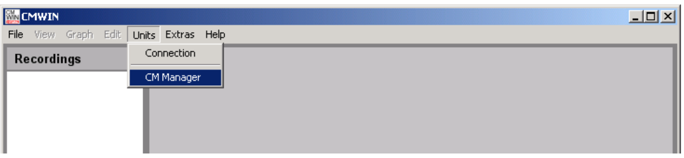
Otherwise it can be reactivated.

Should the previous bus master be reactivated?

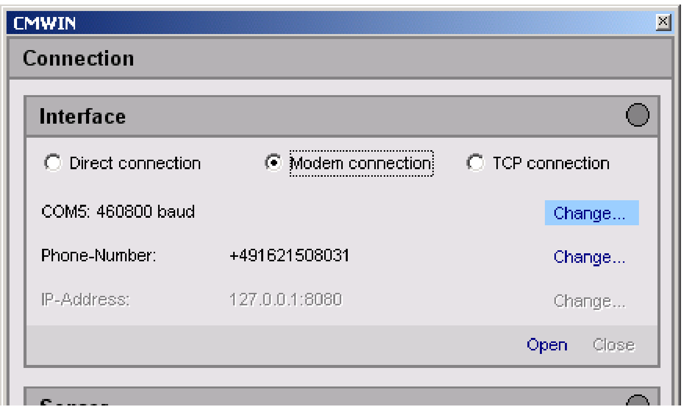
- Then click on "***Yes***" to reactivate the busmaster again.

5.2.3.3 Connection Setup with CMU 1000 using GSM Mobile Network

- Start the HYDAC PC software **CMWIN**
- In the ***Units*** Menu, select the **"CM Manager"** option.



- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.
- Mark the "**Modem Connection**" option in the window that opens.
- Click on "**Change**" to open the window for the interface settings.



○ Direct connection ● Modem connection ○ TCP connection

COM5: 460800 baud Change...

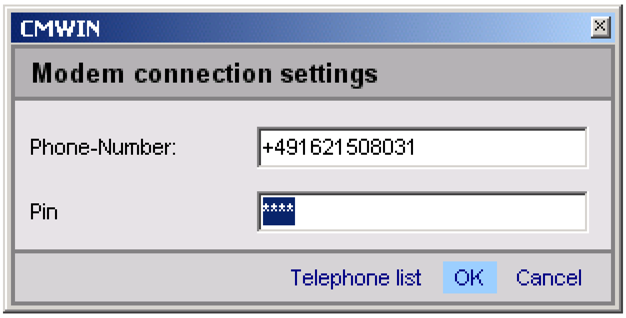
Phone-Number: +491621508031 Change...

IP-Address: 127.0.0.1:8080 Change...

Open Close

- Make the corresponding preselection for the port settings in the window that opens under ***Interface selection***.
- Select the respective port address and Baud rate under ***Interface settings***.
- Pressing "**Refresh**" causes the interfaces marked under ***Interface selection*** to be refreshed in terms of availability.
- Click on “**OK**“ to apply the modified settings or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.

- Click on "**Change**" to open the window for entering the telephone numbers.

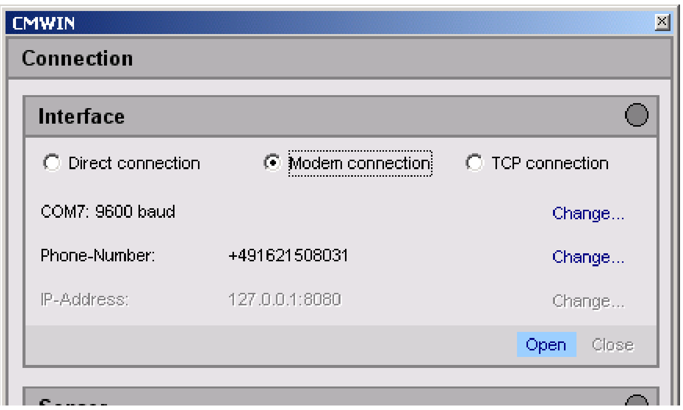


Phone-Number: +491621508031

Pin ****

- Enter the telephone number of the SIM card mounted in the GSM module CSI-F-10.
- In the ***Pin*** box, input the pin code given for the SIM card which is operated in the GSM modem in the PC (not for the SIM card in the CSI-F-10!). If no PIN code has been assigned, then leave this box empty.

- You can set up a list of telephone numbers (address book) with ***Telephone list***.
- Click on “**OK**“ to apply the entries or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.



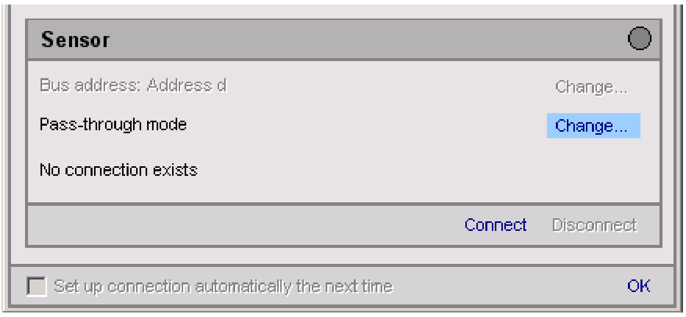
○ Direct connection ● Modem connection ○ TCP connection

COM7: 9600 baud

Phone-Number: +491621508031

IP-Address: 127.0.0.1:8080

- Click on ***Open*** to open the selected interface. The opened interface is indicated by a green dot on the upper right.
- Click on "**Change**" to open the window for the pass-through mode.



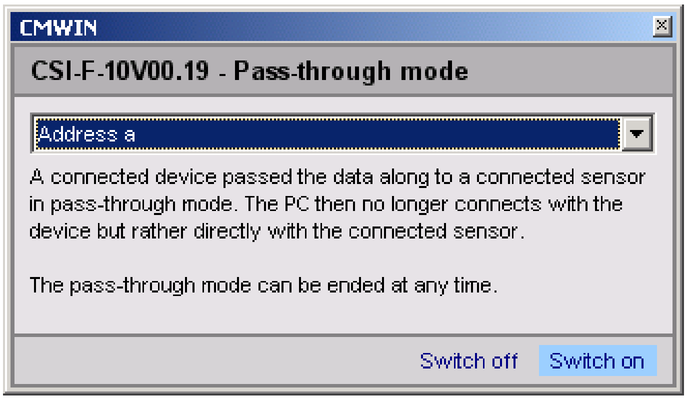
Bus address: Address d

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Select the HSI address of the CMU 1000 connected to the CSI-F-10 in the selection window (Address a in our example).
- Afterwards, click on ***Switch on*** in order to switch on the pass-through mode for the selected channel.

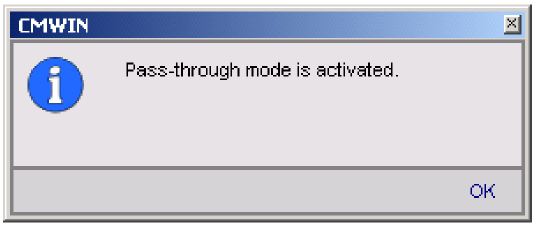


Address a

A connected device passed the data along to a connected sensor in pass-through mode. The PC then no longer connects with the device but rather directly with the connected sensor.

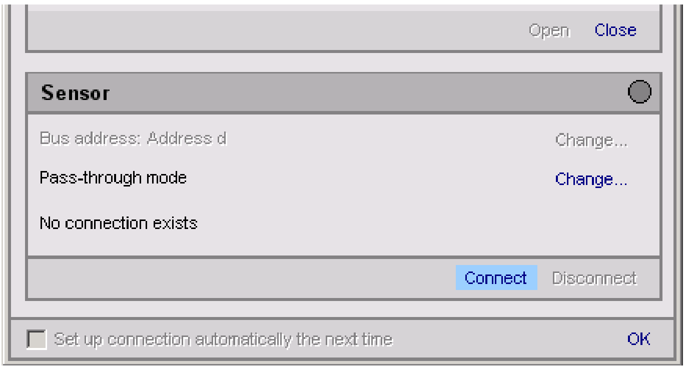
The pass-through mode can be ended at any time.

- The following message appears:



Pass-through mode is activated.

- Confirm this with ***OK***.



Bus address: Address d

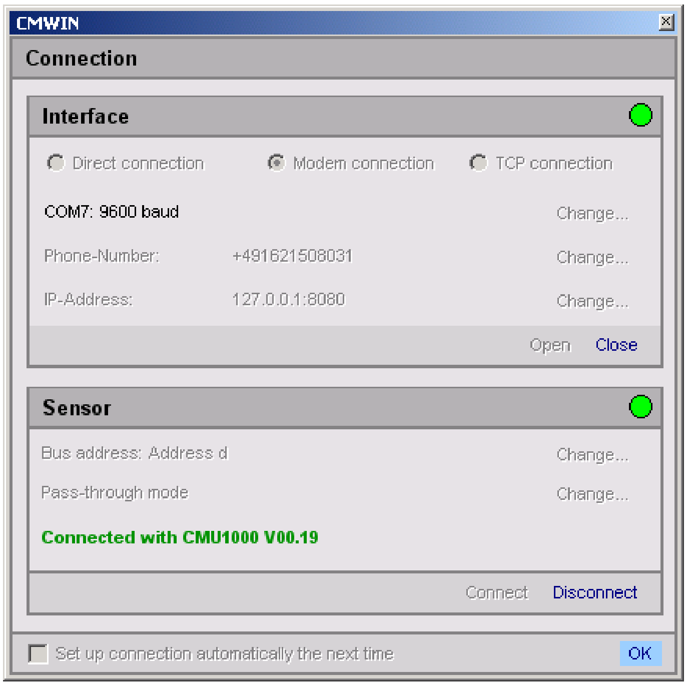
Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Afterwards click on ***Connect*** to connect the PC to the CMU 1000 that is connected with the CSI-F-10

- The successful establishment of the connection will be signalled as shown below:



○ Direct connection ● Modem connection ○ TCP connection

COM7: 9600 baud

Phone-Number: +491621508031

IP-Address: 127.0.0.1:8080

Bus address: Address d

Pass-through mode

Connected with CMU1000 V00.19

□ Set up connection automatically the next time

- End the connection setup by confirming with ***OK***.

5.2.4 TCP Connection

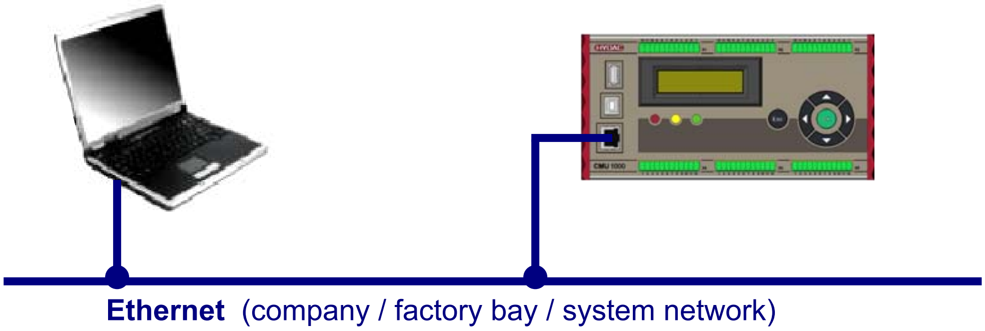
As a third option, you can set up a link between the PC and the CMU 1000 by means of an Ethernet network.
In the following we present an example of this kind of link.

5.2.4.1 Device Connection

Use one Standard RJ45 cable each to connect your PC and the CMU 1000 to a shared Ethernet network.

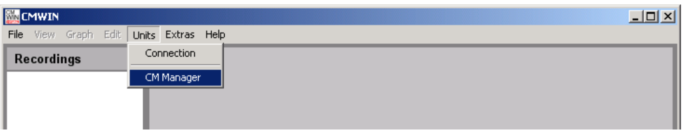
> ***Note!***
> In order to be able to set up a connection via Ethernet, the network settings and parameters in the CMU 1000 must first be correctly stored.
> See in this connection Chapter 5.1.1/5.1.2 "Operation on the Device" and 5.2.5.5 "Setup".

Example:

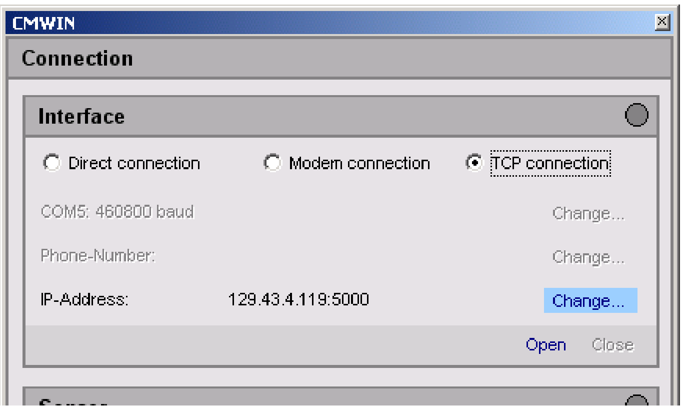


5.2.4.2 Connection Setup

- Start the HYDAC PC software **CMWIN**
- In the ***Units*** Menu, select the **"CM Manager"** option.



- If the ***Connection*** window does not open automatically, select "**Connection**" in the menu bar of the CM Manager.
- Mark the option "**TCP Connection**" option in the window that opens.
- Click on "**Modify**" to open the window for the interface settings.

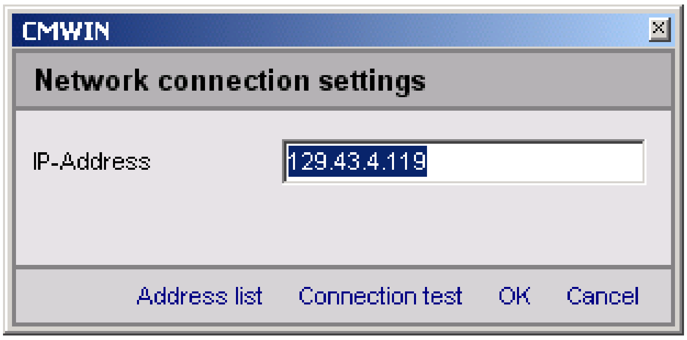


○ Direct connection ○ Modem connection ● TCP connection

COM5: 460800 baud Change...

Phone-Number: Change...

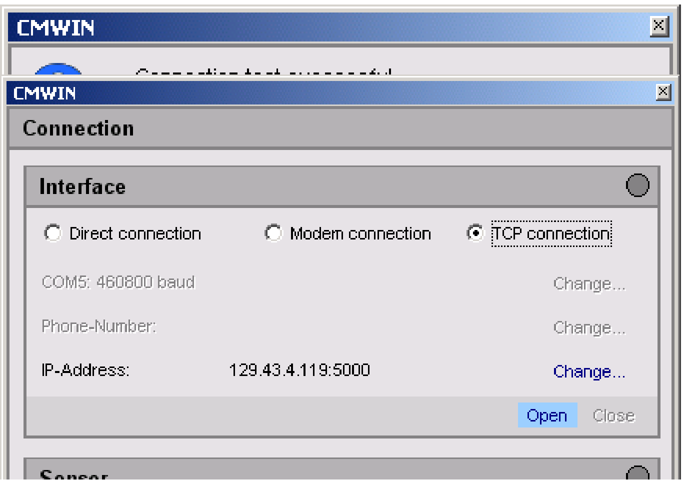
IP-Address: 129.43.4.119:5000 Change...


CMWIN

Network connection settings

IP-Address: 129.43.4.119

- Enter the IP Address stored in the network settings of the CMU 1000.
- You can set up a list of addresses (address book) with ***Address list***.
- With ***Connection test*** you can test the link from the PC to the CMU 1000. If the IP Address has been entered correctly, then the following message will appear:


CMWIN

Connection test successful

CMWIN

Connection

Interface

○ Direct connection ○ Modem connection ● TCP connection

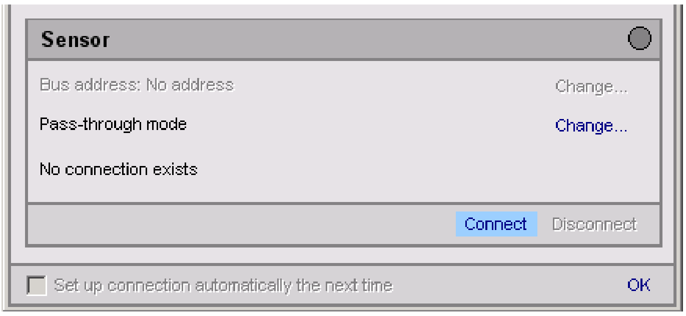
COM5: 460800 baud

Phone-Number:

IP-Address: 129.43.4.119:5000

- Click on “**OK**“ to apply the entries or “**Cancel**“ to discard these changes. In either case you will then return to the ***Connection*** window.

- Click on ***Open*** to open the selected interface. The opened interface is indicated by a green dot on the upper right.


Sensor

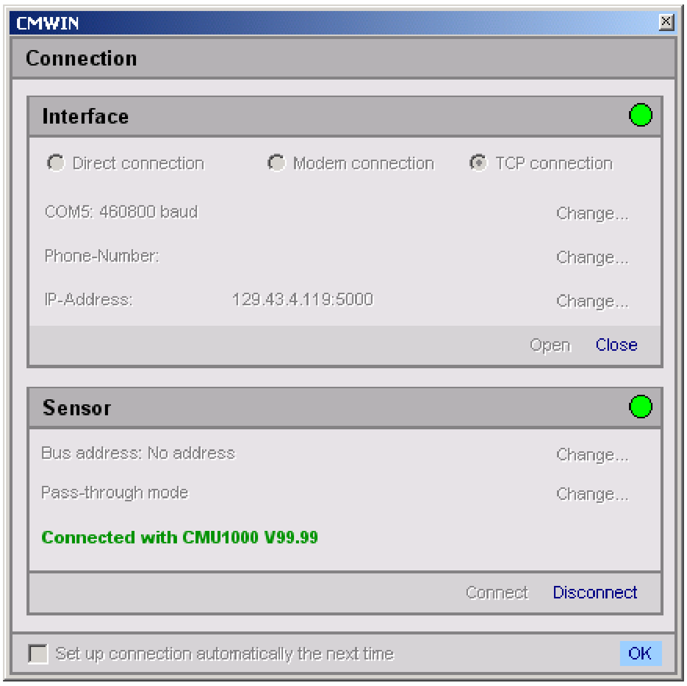
Bus address: No address

Pass-through mode

No connection exists

□ Set up connection automatically the next time

- Afterwards click on ***Connect*** to connect the PC to the CMU
- The successful establishment of the connection will be signalled as shown below:



CMWIN

Connection

Interface ●

○ Direct connection ○ Modem connection ● TCP connection

COM5: 460800 baud Change...

Phone-Number: Change...

IP-Address: 129.43.4.119:5000 Change...

Open Close

Sensor ●

Bus address: No address Change...

Pass-through mode Change...

Connected with CMU1000 V99.99

Connect Disconnect

□ Set up connection automatically the next time OK

- End the connection setup by confirming with ***OK***.

5.2.5 Actions

5.2.5.1 Display Device Status

- ***Status***

The "Status" indicates the current condition of the device. The individual conditions can be be specified in greater detail via the following table.

Ready	No active error present, device is ready for operation
Stand-By	No active error present, but device is currently not ready for operation; it may be that individual device functions are switched off or the device is in a startup phase, etc.
Minor error	A minor error is present which can be acknowledged.
Moderate error	A medium-serious error is present, which may possibly be eliminated by switching On/Off.
Serious error	A serious error is present; the unit must be sent in to the manufacturer.

- ***Status code***

The "Status code" is dependent on the CM Program present in the device and reflects the conditions of the Boolean output values used in the program.
For this, the Boolean output values are displayed from right to left in ascending binary sequence, i.e. the lowest-value bit corresponds to the Boolean output value 0.

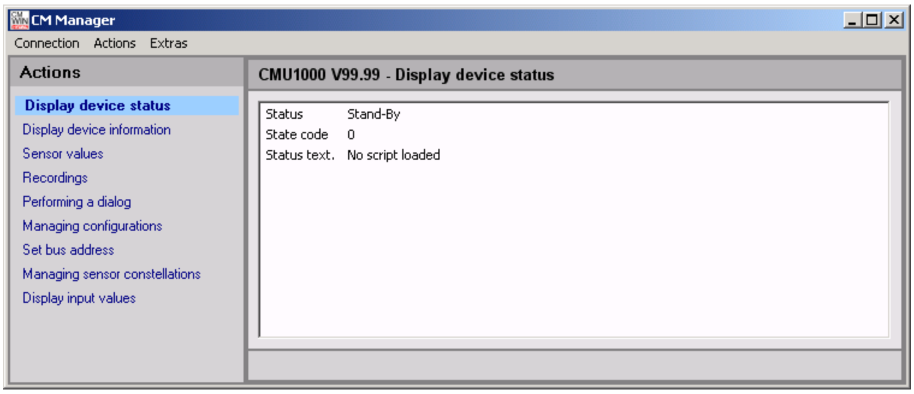
Example:

0 = No Boolean output values used in the program
1 0 1 0 = Boolean output value 0 = 0 (LSB)
Boolean output value 1 = 1
Boolean output value 2 = 0
Boolean output value 3 = 1

- ***Status text***

The "Status text" indicates whether or not a CM Program is available in the device.

No script loaded	No CM Program is available in the device.
Script in execution	A CM Program is available in the device.

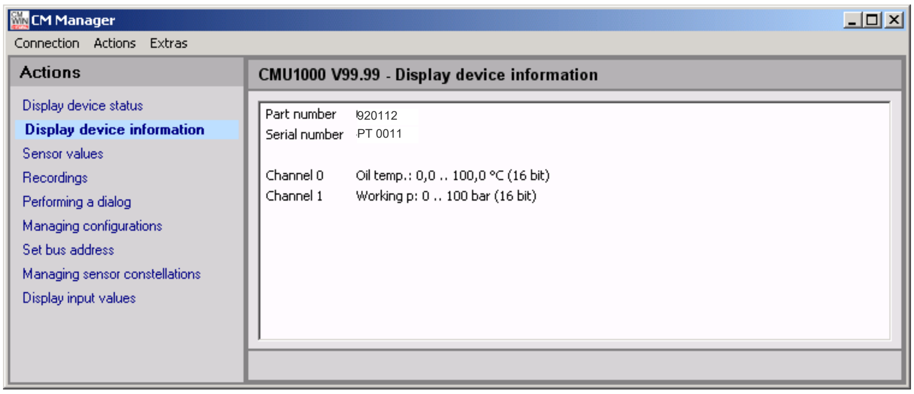


Status Stand-By
State code 0
Status text. No script loaded

5.2.5.2 Display Device Information

- Here the values of the following status parameters are displayed:
  - Part numer
  - Serial number
  - Channel information

  The Channel information reflects the numerical input values from the CM Program. Channel 0 corresponds thereby to the first numerical input value in the CM Program, Channel 1 the second one, etc.

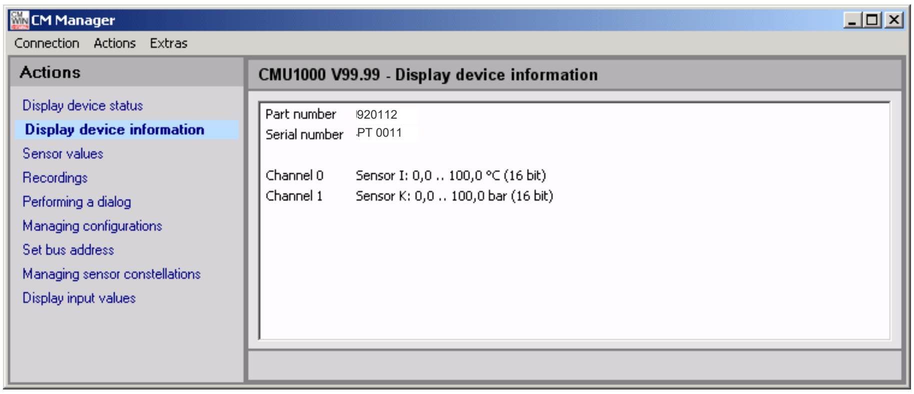


CMU1000 V99.99 - Display device information

Part number 920112
Serial number PT 0011

Channel 0 Oil temp.: 0,0 .. 100,0 °C (16 bit)
Channel 1 Working p: 0 .. 100 bar (16 bit)

If no CM Program is available in the device, then the connected sensors will be displayed in this position with measurement range and unit.



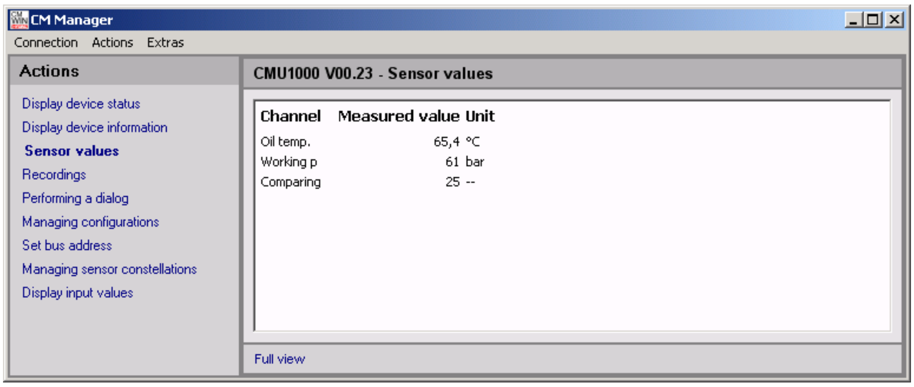
CMU1000 V99.99 - Display device information

Part number 920112
Serial number PT 0011

Channel 0 Sensor I: 0,0 .. 100,0 °C (16 bit)
Channel 1 Sensor K: 0,0 .. 100,0 bar (16 bit)

5.2.5.3 Sensor Values

- Here the result values (Numerical output values) from the CM Program in the device are displayed.
  If no CM Program is available in the device, then the measured values of the connected sensors will be displayed.

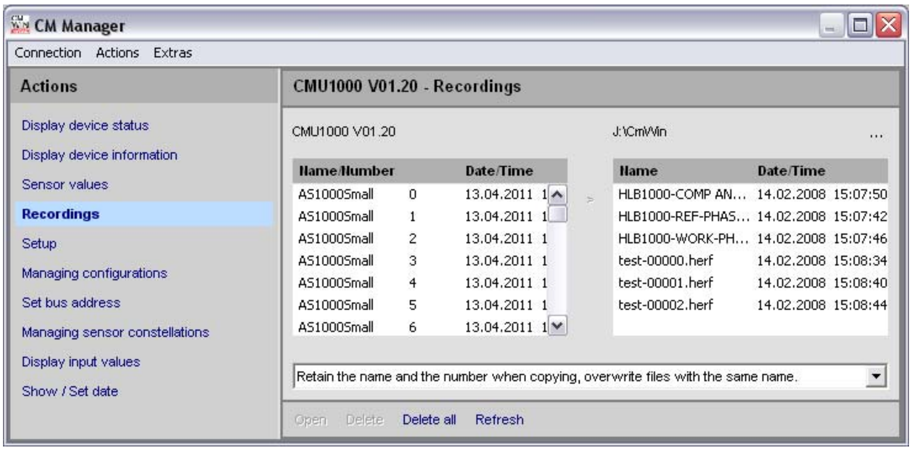


CMU1000 V00.23 - Sensor values

Channel	Measured value	Unit
Oil temp.	65,4	°C
Working p	61	bar
Comparing	25	--

5.2.5.4 Managing Recordings

- Here you can manage the recordings stored in the CMU 1000 (copying, opening, updating, deleting)



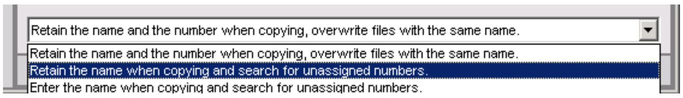
CMU1000 V01.20 J:\CmWin ...

Name/Number		Date/Time
AS1000Small	0	13.04.2011 1
AS1000Small	1	13.04.2011 1
AS1000Small	2	13.04.2011 1
AS1000Small	3	13.04.2011 1
AS1000Small	4	13.04.2011 1
AS1000Small	5	13.04.2011 1
AS1000Small	6	13.04.2011 1

Name	Date/Time
HLB1000-COMP AN...	14.02.2008 15:07:50
HLB1000-REF-PHAS...	14.02.2008 15:07:42
HLB1000-WORK-PH...	14.02.2008 15:07:46
test-00000.herf	14.02.2008 15:08:34
test-00001.herf	14.02.2008 15:08:40
test-00002.herf	14.02.2008 15:08:44

Retain the name and the number when copying, overwrite files with the same name.

You have the following options for saving and copying:

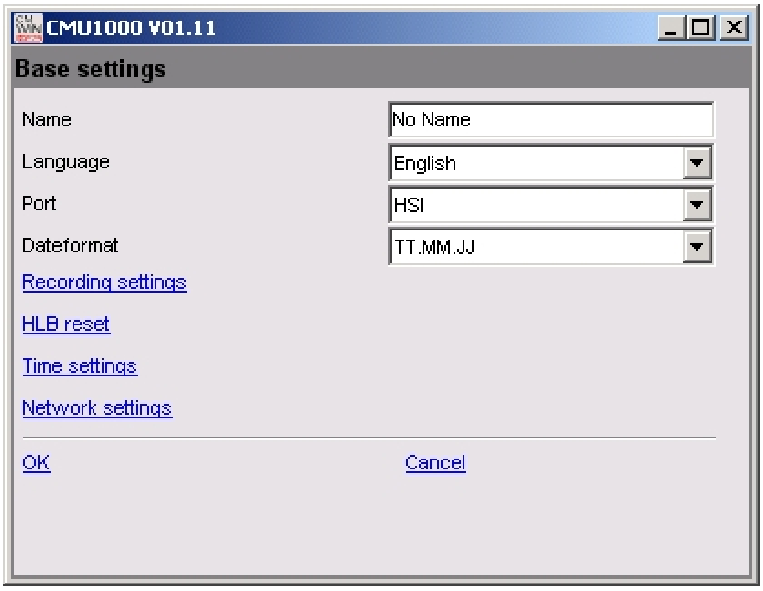

Retain the name and the number when copying, overwrite files with the same name.
Retain the name and the number when copying, overwrite files with the same name.
Retain the name when copying and search for unassigned numbers.
Enter the name when copying and search for unassigned numbers.

5.2.5.5 Setup

This menu option corresponds to the menu option "**Settings**" in the internal device structure for the purpose of configuring the CMU 1000.
The following settings can be made:

Base settings

- **Name**	
- **Port**	[RS232 / HSI]
- **Language**	[German / English / French
- **Dateformat**	[TT.MM.YY / MM/TT/YY / YY-MM-TT]

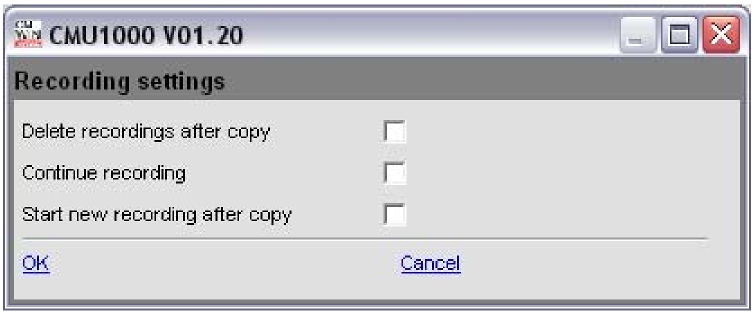


Name: No Name
Language: English
Port: HSI
Dateformat: TT.MM.JJ

Recording settings

Here you specify whether you wish the recorded data held in the internal memory to be deleted after copying onto a USB stick and whether, after the CMU reboots, to generate a new record file or to continue with the previous one.

- **Delete recordings after copy** [☑yes / □ no]
- **Continue recording** [☑yes / □ no]
- **Start new recording after copy** [☑yes / □ no]



CMU1000 V01.20

Recording settings

Delete recordings after copy □

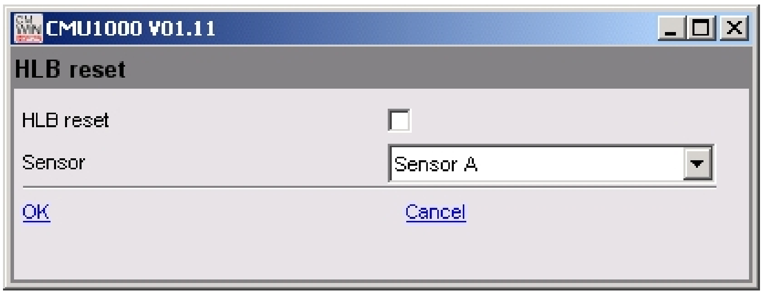
Continue recording □

Start new recording after copy □

HLB reset

Here you can reset an HLB connected to Channel A .. H (delete memory).

- **HLB Reset** [☑yes / □ no]
- **Sensor** [Select sensor A .. H]



CMU1000 V01.11

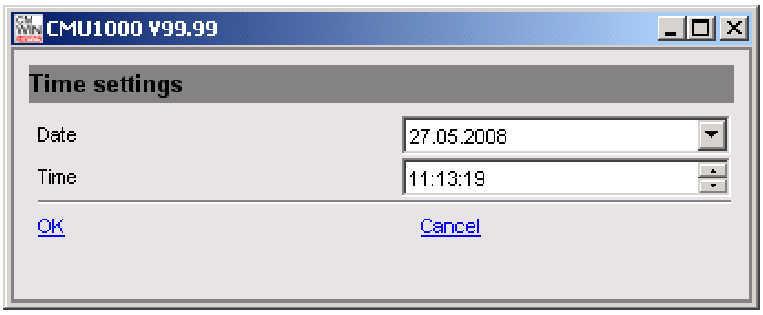
HLB reset

HLB reset □

Sensor: Sensor A

Time settings

- **Date** [*Enter Date*]
- **Time** [*Enter Time*]



CMU1000 V99.99

Time settings

Date: 27.05.2008

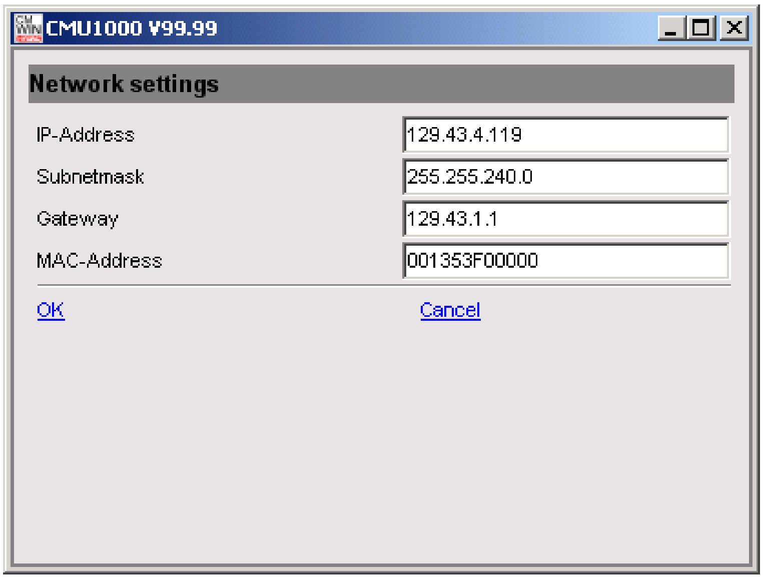
Time: 11:13:19

Network settings

If the CMU 1000 is integrated in a network at the customer site, then the following settings need to be carried out in accordance with this network:

- IP Address [*enter IP Address*]
- Subnetwork mask [*enter Subnetwork mask*]
- Gateway Address [*enter Gateway Address*]
- MAC Address [permanently set at the factory, read-only]

Example:

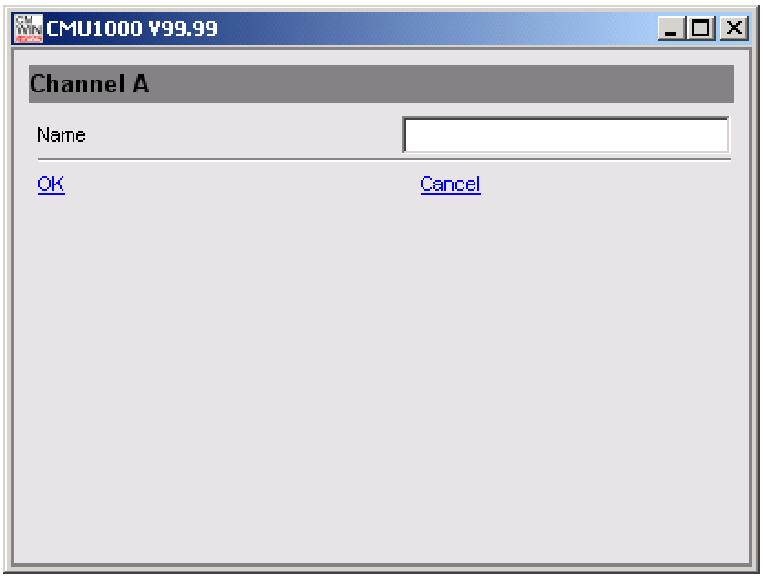


Network settings

IP-Address 129.43.4.119
Subnetmask 255.255.240.0
Gateway 129.43.1.1
MAC-Address 001353F00000

Peripherals - Channel Settings

- **Channel A to Channel H (HSI channels)**
  - Name [*Enter name*]

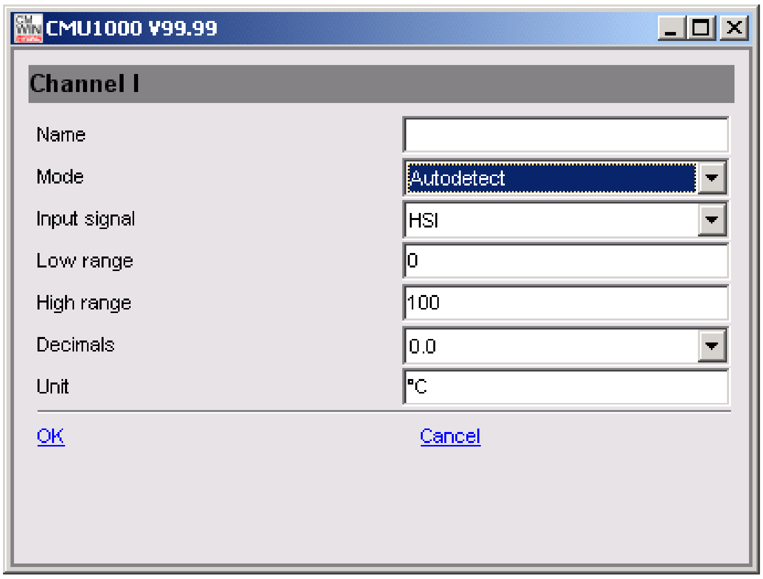


Channel A

Name

- **Channel I to Channel P**

- Name	[*Enter name*]
- Mode	[Off / Manual / Autodetect]
- Input signal ***)**	[HSI / 0..20 mA / 4..20 mA / 0..5 V / 0..10 V / 0.5..4.5 V / 0..50 V / 0.5..5.5 V / 1..5 V / 1..6 V / -10..+10 V]
- Low range	[*Enter low range*]
- High range	[*Enter high range*]
- Decimal format	[0 / 0.0 / 0.00 / 0.000]
- Unit	[*Enter unit*]



CMU1000 V99.99

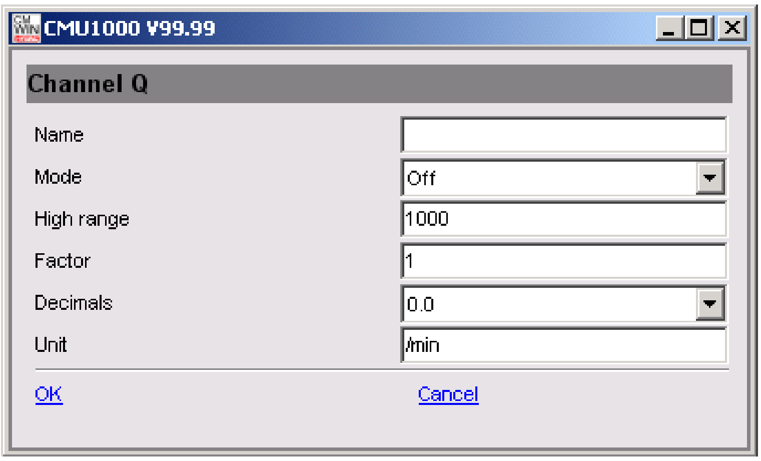
Channel I

Name	
Mode	Autodetect
Input signal	HSI
Low range	0
High range	100
Decimals	0.0
Unit	°C

*) Possible input signals channel-dependent

- **Channel Q and Channel R**

Name	[*enter Name*]
Mode	[Off / Active]
High range	[*enter High range*]
Factor	[*enter Factor*]
Decimal format	[0 / 0.0 / 0.00 / 0.000]
Unit	[*enter Unit*]



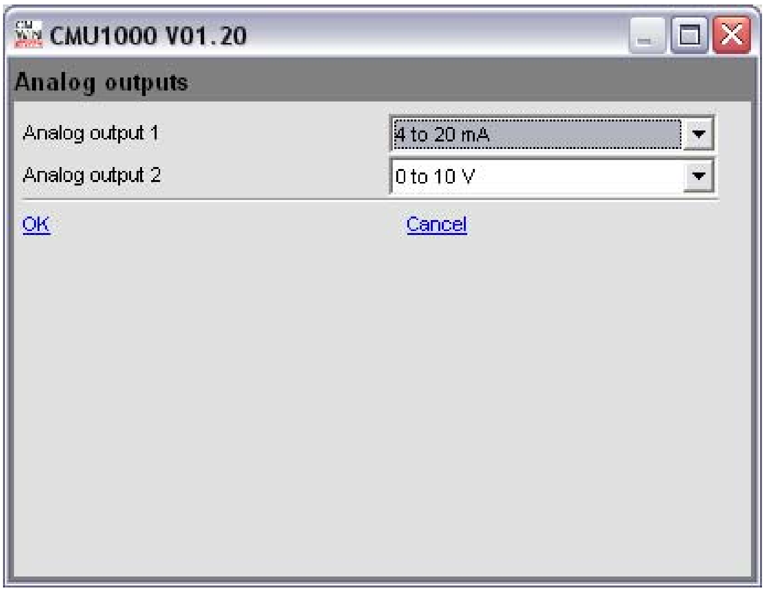
CMU1000 V99.99

Channel Q

Name	
Mode	Off
High range	1000
Factor	1
Decimals	0.0
Unit	/min

Peripherals - Analog outputs

- **Analog output 1, Analog output 2**
  You use the Analog outputs settings to establish which type of signal is to be available at the two analog ouputs.



CMU1000 V01.20

Analog outputs

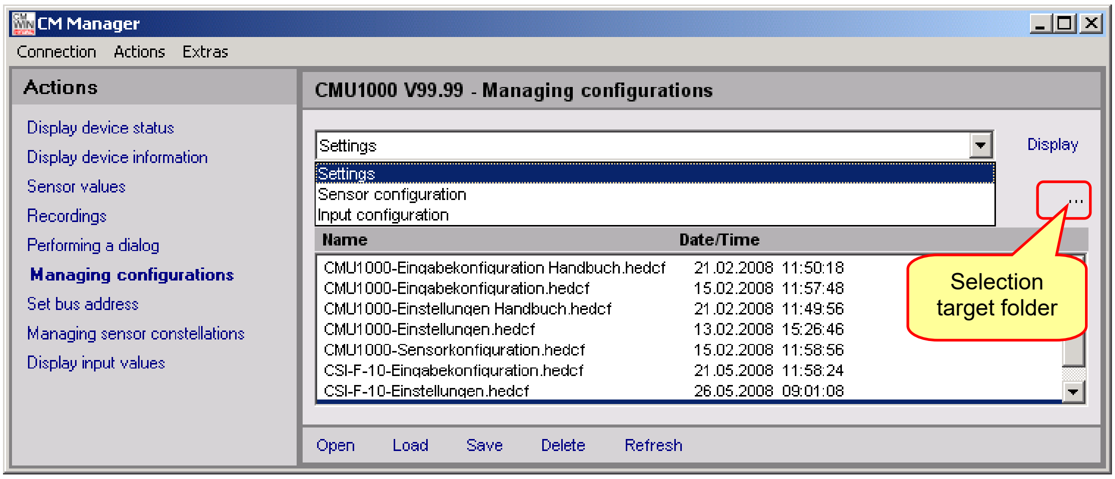
Analog output 1	4 to 20 mA
Analog output 2	0 to 10 V

5.2.5.6 Managing Configurations

Here you can generate and manage various configuration files. These configuration files can, for example, be generated in series on a "Master" CMU and then subsequently loaded onto an unlimited number of other CMUs.

The following configuration files can be generated and managed:

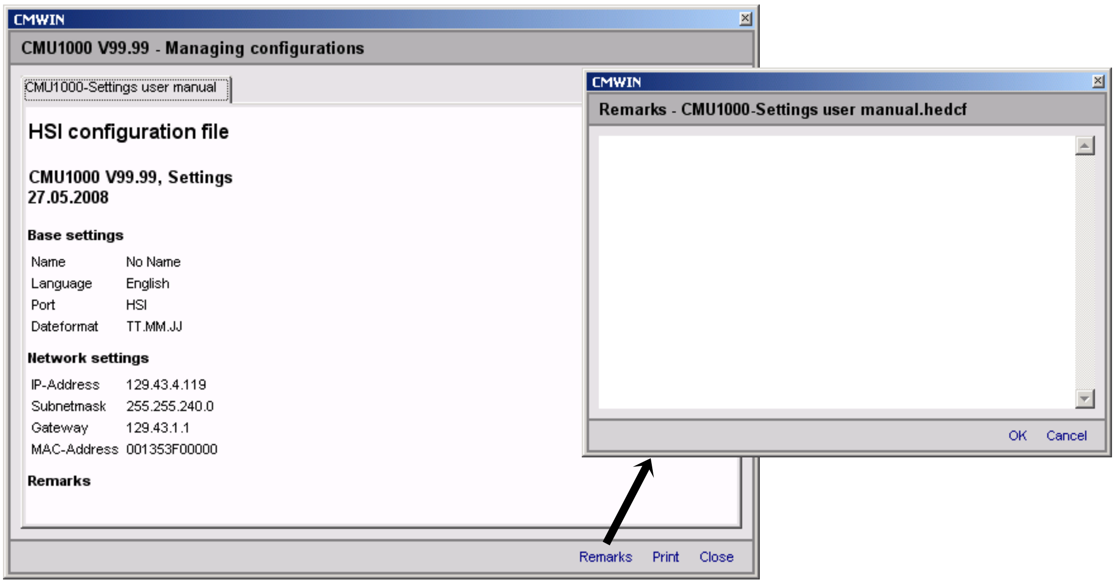
- Settings configuration
- Sensor configuration
- Input configuration



CMU1000 V99.99 - Managing configurations

Name	Date/Time
CMU1000-Eingabekonfiguration Handbuch.hedcf	21.02.2008 11:50:18
CMU1000-Eingabekonfiguration.hedcf	15.02.2008 11:57:48
CMU1000-Einstellungen Handbuch.hedcf	21.02.2008 11:49:56
CMU1000-Einstellungen.hedcf	13.02.2008 15:26:46
CMU1000-Sensorkonfiguration.hedcf	15.02.2008 11:58:56
CSI-F-10-Eingabekonfiguration.hedcf	21.05.2008 11:58:24
CSI-F-10-Einstellungen.hedcf	26.05.2008 09:01:08

- With ***Open*** you can call up the display of the content of a configuration file listed in the lower display field. To accomplish this, mark the desired file with a mouse click. You can add remarks to the content and print out the file in the window that then opens automatically.



CMU1000 V99.99 - Managing configurations

HSI configuration file

CMU1000 V99.99, Settings
27.05.2008

Base settings

Name	No Name
Language	English
Port	HSI
Dateformat	TT.MM.JJ

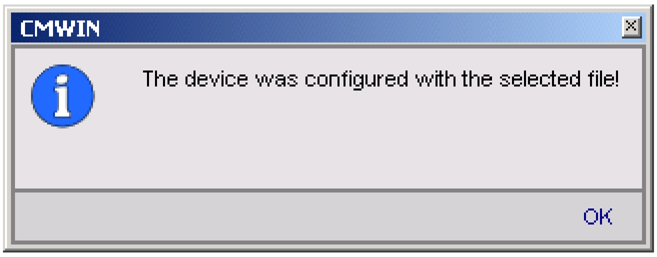
Network settings

IP-Address	129.43.4.119
Subnetmask	255.255.240.0
Gateway	129.43.1.1
MAC-Address	001353F00000

Remarks

Remarks - CMU1000-Settings user manual.hedcf

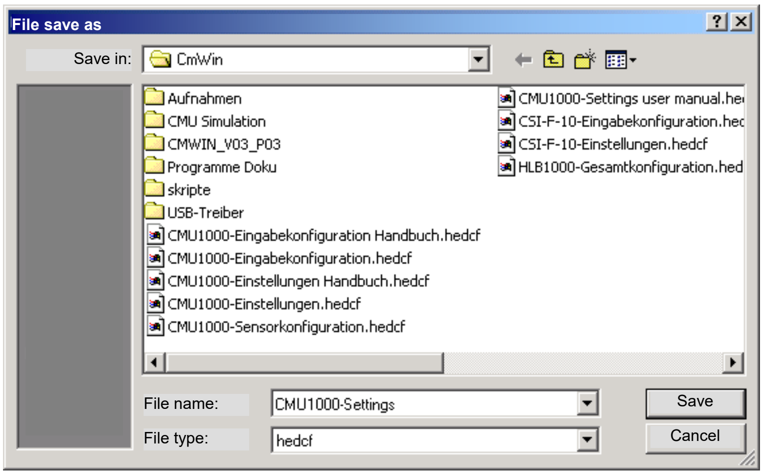
- With ***Load*** you can transfer a configuration file (marked in the lower display field) from the PC to the CMU 1000. The following message appears after the completion of the transfer:



The device was configured with the selected file!

All of the settings stored in the configuration file were carried out after the transfer of the file in the CMU 1000.

- When you press ***Save,*** you generate a new configuration file or save a modified one to the previously specified target folder (in our example: *C:\Temp*). The following window opens for this purpose:


File save as
Save in:
CmWin
Aufnahmen
CMU Simulation
CMWIN_V03_P03
Programme Doku
skripte
USB-Treiber
CMU1000-Eingabekonfiguration Handbuch.hedcf
CMU1000-Eingabekonfiguration.hedcf
CMU1000-Einstellungen Handbuch.hedcf
CMU1000-Einstellungen.hedcf
CMU1000-Sensorkonfiguration.hedcf
CMU1000-Settings user manual.he
CSI-F-10-Eingabekonfiguration.hec
CSI-F-10-Einstellungen.hedcf
HLB1000-Gesamtkonfiguration.hed
Save
Cancel

File name: CMU1000-Settings

File type: hedcf

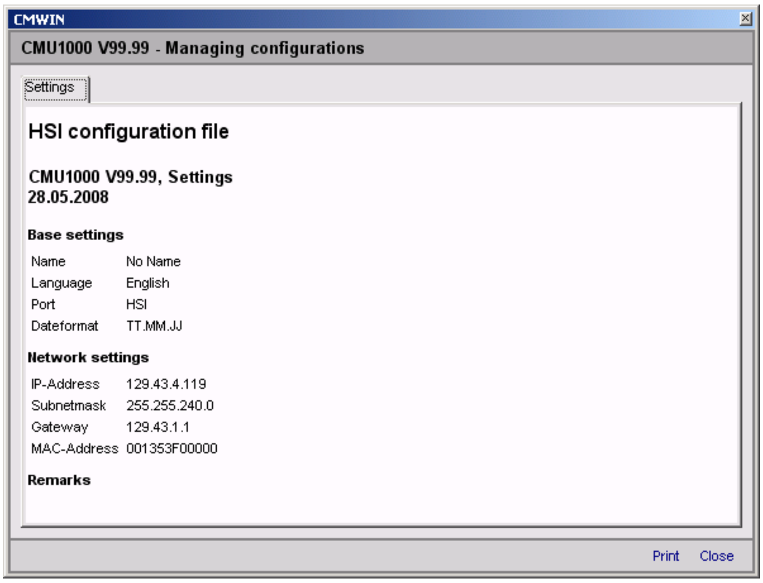
- Pressing ***Delete*** causes the configuration file in the target folder marked in the lower display field to be deleted.

- Pressing ***Refresh*** causes the display field for the configuration files to be updated. The defined target folder will be read out again for this purpose.
  This is necessary, e.g. when you copy or delete configuration files with Windows Explorer. These changes will not be displayed in the folder until after "Refresh" has been carried out in CMWIN.

- With ***Display*** you can display the respective momentary "Actual Configuration" of the connected CMU 1000.

The individual configuration files are explained briefly in the following:

- ***Settings***
  The basic settings and the network settings are stored in the Settings Configuration file.



HSI configuration file

CMU1000 V99.99, Settings
28.05.2008

Base settings

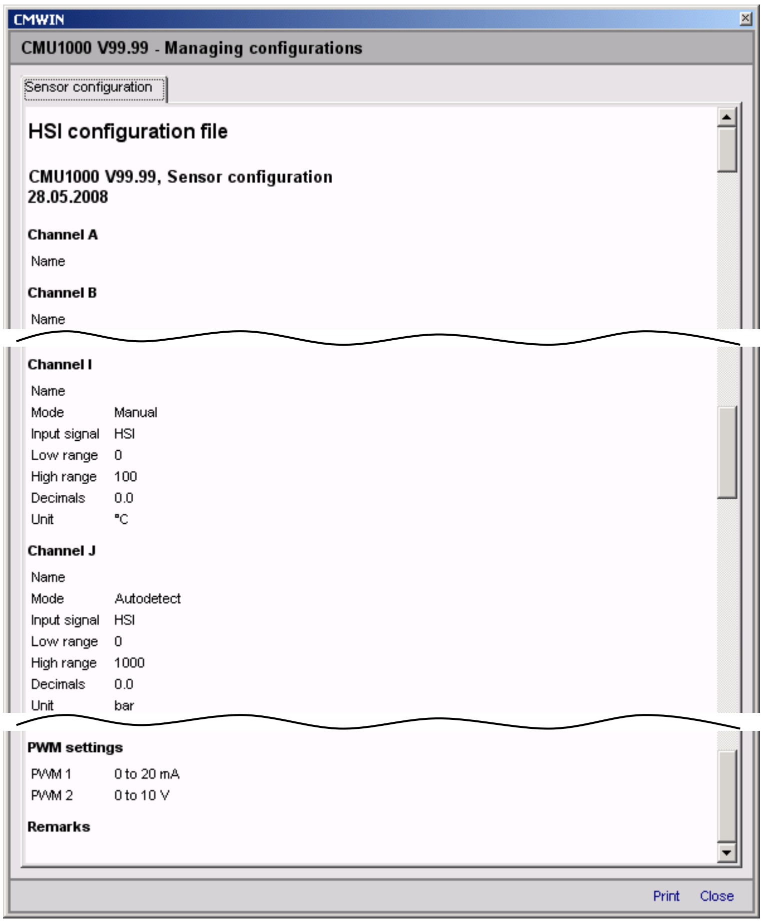
Name	No Name
Language	English
Port	HSI
Dateformat	TT.MM.JJ

Network settings

IP-Address	129.43.4.119
Subnetmask	255.255.240.0
Gateway	129.43.1.1
MAC-Address	001353F00000

Remarks

- ***Sensor configuration***
  The peripherals settings are stored in the Sensor Configuration file.



HSI configuration file

CMU1000 V99.99, Sensor configuration
28.05.2008

Channel A

Name

Channel B

Name

Channel I

Name	
Mode	Manual
Input signal	HSI
Low range	0
High range	100
Decimals	0.0
Unit	°C

Channel J

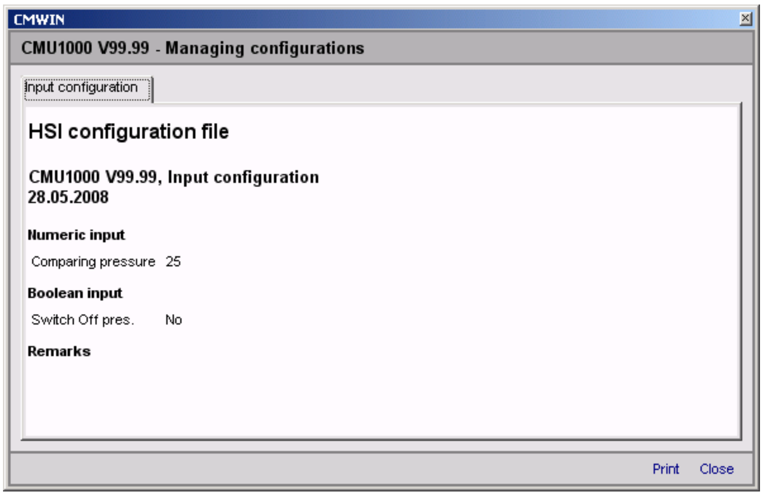
Name	
Mode	Autodetect
Input signal	HSI
Low range	0
High range	1000
Decimals	0.0
Unit	bar

PWM settings

PWM 1	0 to 20 mA
PWM 2	0 to 10 V

Remarks

- ***Input configuration***
  The values of the numerical and Boolean input values used in the CM Program are stored in the input configuration file.



CMU1000 V99.99 - Managing configurations

HSI configuration file

CMU1000 V99.99, Input configuration
28.05.2008

Numeric input

Comparing pressure 25

Boolean input

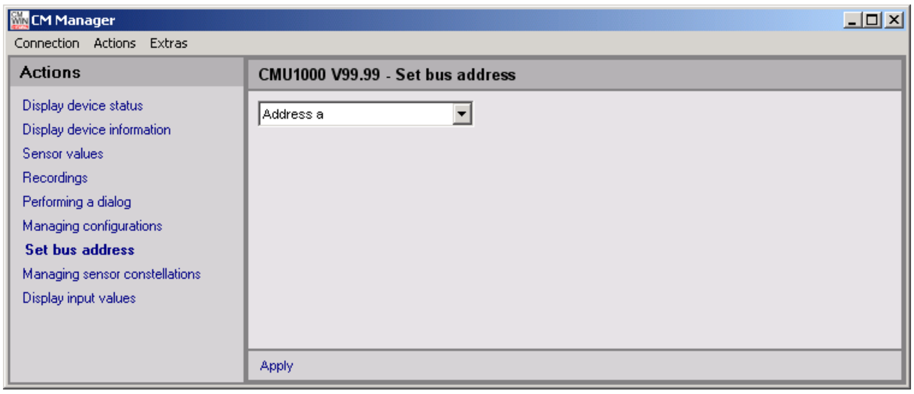
Switch Off pres. No

Remarks

5.2.5.7 Set bus address

Here you can assign the CMU 1000 an "HSI Bus address". This is required when several CMU 1000 (or other HYDAC CM devices) are connected to one HSI Bus and these devices are all set up to be be addressed by a central PC. A total of a maximum of 26 HYDAC CM devices can be linked to one HSI Bus, i.e. you can assign the addresses "a" to "z" (see Chap. 5.2.2. CMU 1000 at the HSI Bus).

If the CMU 1000 is operated as a single device, then the bus address setting is not required; it is not relevant for operational purposes.
The standard setting is "Address a".



CMU1000 V99.99 - Set bus address

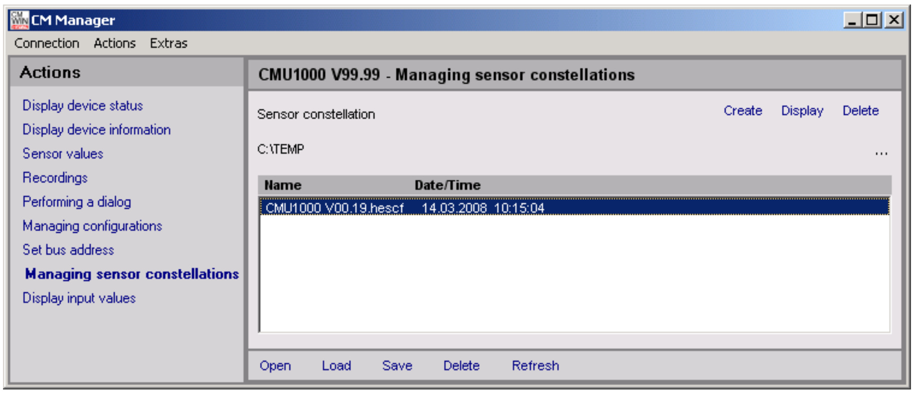
Address a

5.2.5.8 Managing Sensor Constellations

Reliable system monitoring means ensuring that exactly the same sensors that were connected at the time the CMU 1000 was configured are connected during operation.
The sensor constellation is used for this purpose. The sensor constellation is a monitoring instrument for the connected sensor system, i.e. it performs a continuous comparison between the connected "Actual" sensor system and the specified "Nominal" sensor system.
(see Chap. 6.1.5. Sensor Constellation)

You can use this function to generate and manage various sensor constellation files. These constellation files can, like the configuration files, be generated on a "Master" CMU and then subsequently loaded onto an unlimited number of other CMUs.

If no sensor constellation has yet been generated in the device then the following window opens:



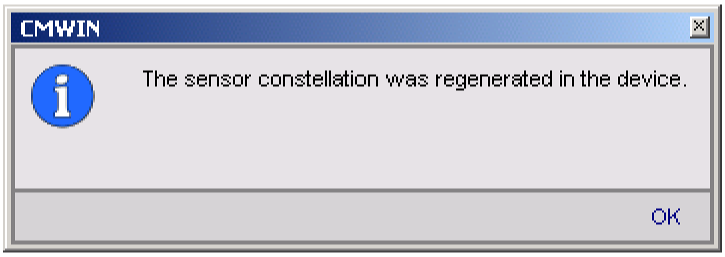
CMU1000 V99.99 - Managing sensor constellations

Sensor constellation

C:\TEMP

Name	Date/Time
CMU1000 V00.19.hescf	14.03.2008 10:15:04

- You can generate a new sensor constellation in the device by pressing ***Create*** (in the upper command bar). Afterwards, the following message appears:



The sensor constellation was regenerated in the device.

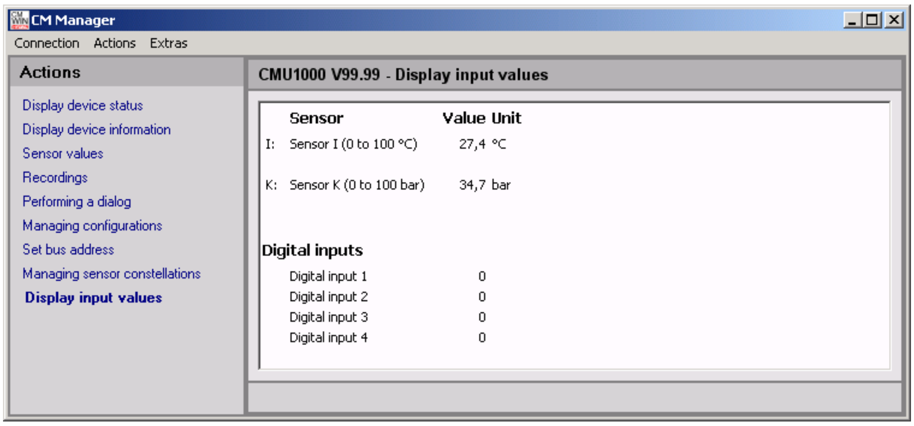
- You can have the momentarily current sensor constellation in the device displayed by pressing ***Display*** (in the upper command bar).

- You can delete the momentarily current sensor constellation in the device displayed by pressing ***Delete*** (in the upper command bar).

All other functions in the lower command bar (Open, Load, Save, Delete, etc.) and the selection of the target folder for saving the files are identical with Chapter 5.2.2.6, "Managing configurations".

5.2.5.9 Display Input Values

- Here the current measured values and statuses of the analog (also HSI and SMART) and digital sensors connected to the CMU 1000 are displayed



CMU1000 V99.99 - Display input values

	Sensor	Value	Unit
I:	Sensor I (0 to 100 °C)	27,4	°C
K:	Sensor K (0 to 100 bar)	34,7	bar

Digital inputs

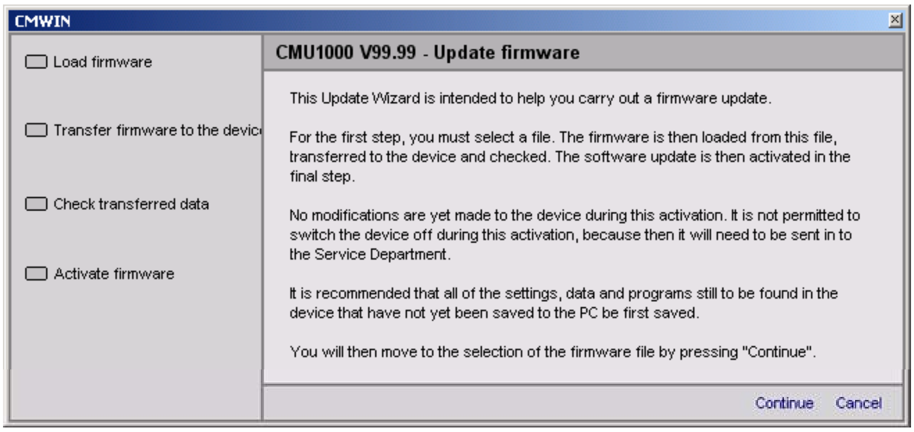
Digital input 1	0
Digital input 2	0
Digital input 3	0
Digital input 4	0

5.2.6 Extras

5.2.6.1 Update Firmware

Caution
The voltage supply to the CMU 1000 is not permitted to be interrupted during the firmware update. If the voltage supply fails during the update process, then trouble-free functioning can no longer be ensured and the device must be sent back to HYDAC SERVICE GMBH.

- You can update the firmware of your CMU 1000 after selecting this menu option. The following window opens for this purpose:



CMU1000 V99.99 - Update firmware

This Update Wizard is intended to help you carry out a firmware update.

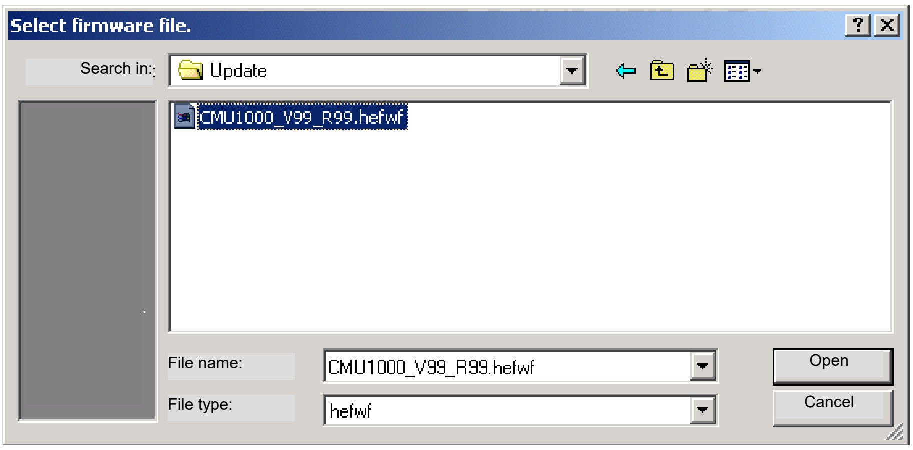
For the first step, you must select a file. The firmware is then loaded from this file, transferred to the device and checked. The software update is then activated in the final step.

No modifications are yet made to the device during this activation. It is not permitted to switch the device off during this activation, because then it will need to be sent in to the Service Department.

It is recommended that all of the settings, data and programs still to be found in the device that have not yet been saved to the PC be first saved.

You will then move to the selection of the firmware file by pressing "Continue".

- Follow the instructions and confirm with ***Continue***. In the window that opens, select the corresponding Update file and click on ***Open.***



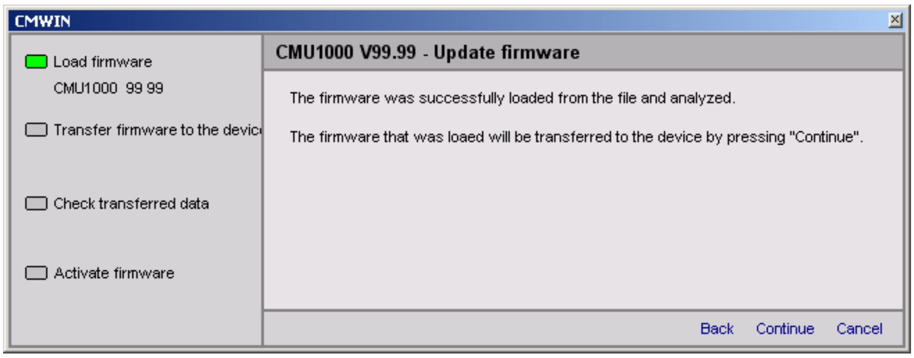
Search in: Update

CMU1000_V99_R99.hefwf

File name: CMU1000_V99_R99.hefwf

File type: hefwf

- Follow the instructions in the following window:

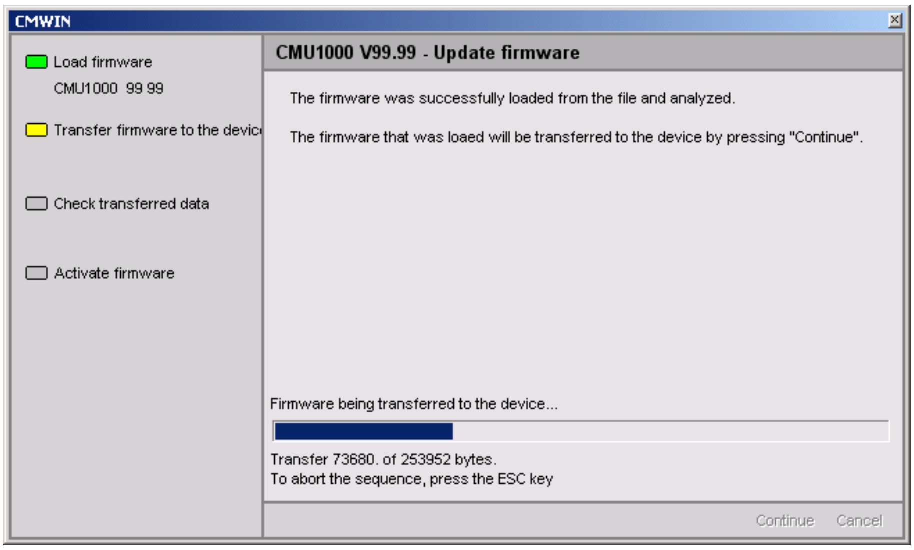


CMU1000 V99.99 - Update firmware

The firmware was successfully loaded from the file and analyzed.

The firmware that was loaed will be transferred to the device by pressing "Continue".

- Confirming with ***Continue*** will cause the data to be transferred to the CMU 1000.



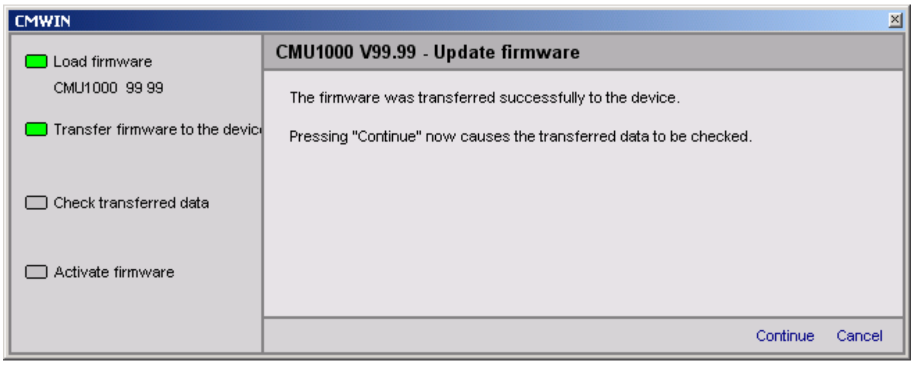
CMU1000 V99.99 - Update firmware

The firmware was successfully loaded from the file and analyzed.

The firmware that was loaed will be transferred to the device by pressing "Continue".

Firmware being transferred to the device...

Transfer 73680. of 253952 bytes.
To abort the sequence, press the ESC key

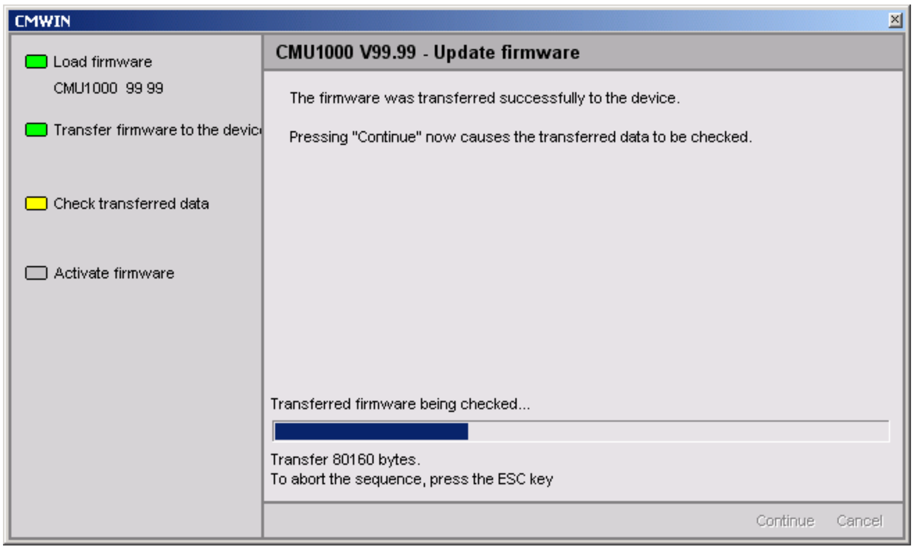


CMU1000 V99.99 - Update firmware

The firmware was transferred successfully to the device.

Pressing "Continue" now causes the transferred data to be checked.

- Confirming with ***Continue*** will cause the data in the CMU 1000 to be checked, and then the two following windows will appear one after the other:



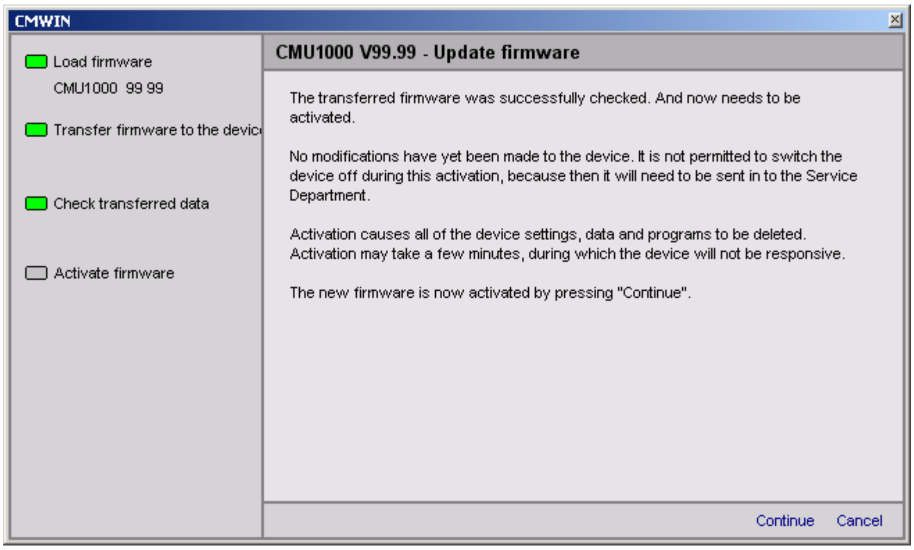
CMU1000 V99.99 - Update firmware

The firmware was transferred successfully to the device.

Pressing "Continue" now causes the transferred data to be checked.

Transferred firmware being checked...

Transfer 80160 bytes.
To abort the sequence, press the ESC key



CMU1000 V99.99 - Update firmware

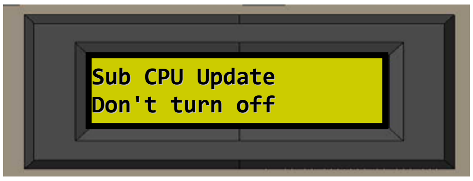
The transferred firmware was successfully checked. And now needs to be activated.

No modifications have yet been made to the device. It is not permitted to switch the device off during this activation, because then it will need to be sent in to the Service Department.

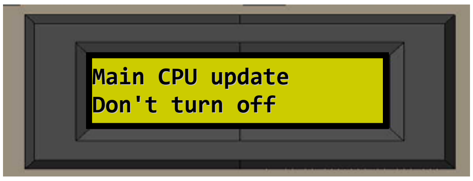
Activation causes all of the device settings, data and programs to be deleted. Activation may take a few minutes, during which the device will not be responsive.

The new firmware is now activated by pressing "Continue".

- Confirming again with ***Continue*** will activate the new firmware in the device. The following messages will appear one after the other in the CMU 1000 display for this purpose for approximately 10 seconds:



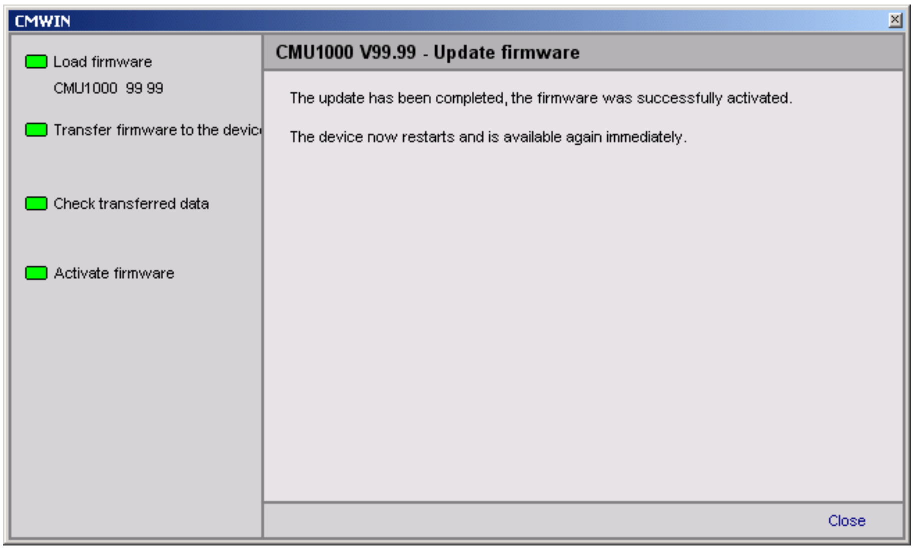
```
Sub CPU Update
Don't turn off
```



```
Main CPU update
Don't turn off
```

Afterwards, the CMU 1000 reboots with the updated firmware.

- The following window opens as the last one in the sequence in the CMWIN. Pressing ***Close*** returns you to the CM Manager.



CMU1000 V99.99 - Update firmware

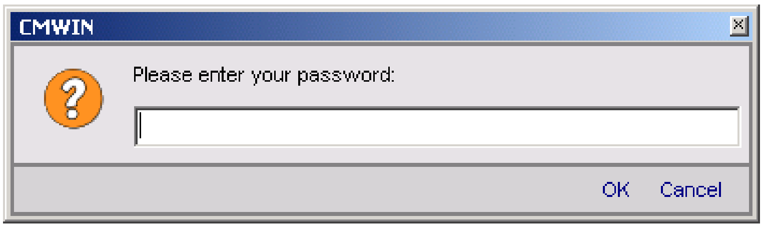
The update has been completed, the firmware was successfully activated.

The device now restarts and is available again immediately.

Note!
All settings, configurations, constellations and the CM program are retained and not overwritten when the firmware is updated.

5.2.6.2 Set Password Protection

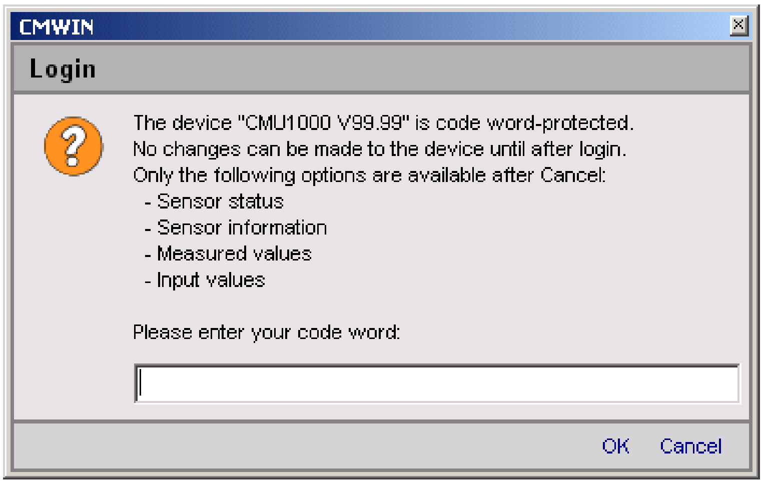
- To prevent unauthorized access to the CMU 1000, you can equip the device with password protection. Select the function ***Set password protection*** for this purpose. The following window opens:



Please enter your password:

You are prompted at this point to enter a password. If the CMU 1000 is still in the condition it was at the time of delivery or if no password protection has been set, then the standard password for this function is: **0000**.

Afterwards, disconnect the link to the CMU 1000 and then reconnect it. The following window opens:



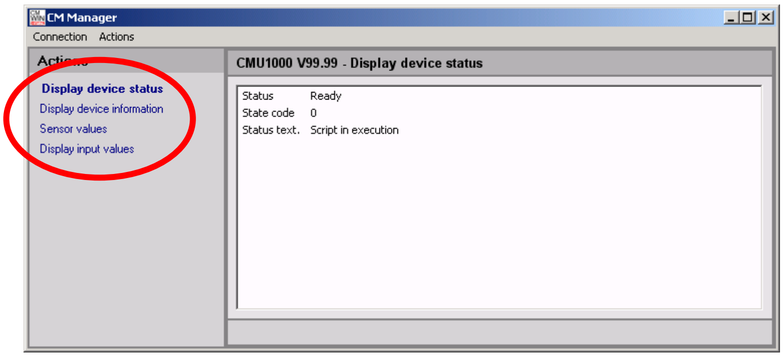
The device "CMU1000 V99.99" is code word-protected.
No changes can be made to the device until after login.
Only the following options are available after Cancel:
- Sensor status
- Sensor information
- Measured values
- Input values

Please enter your code word:

- Enter the ***Password*** and then click on ***OK*** in order to have unlimited access to the CMU 1000.

- If you click on ***Cancel*** without entering a password, then you will obtain only "Reading Rights". You can only access the following information:
  Sensor status, sensor information, measured values, input values

  The ***CM Manager*** is reduced in such cases to the following functions:



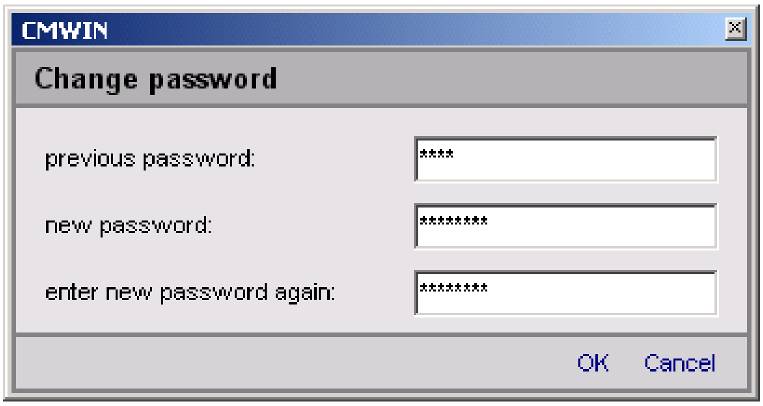
CMU1000 V99.99 - Display device status

Status Ready
State code 0
Status text. Script in execution

It was not possible to carry out any settings.

5.2.6.3 Change Password

- Here you have the opportunity of changing the password. After the ***Change passwort*** function has been selected, the following window opens:



CMWIN

Change password

previous password: ****

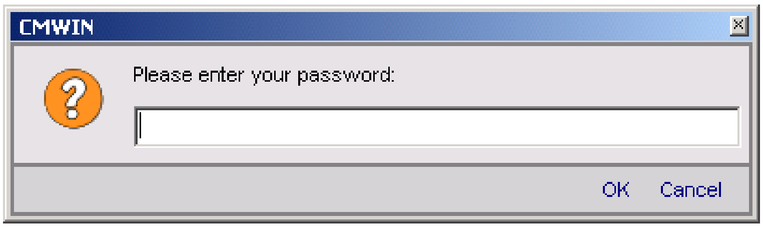
new password: ********

enter new password again: ********

- Enter the previous password in the top line.
- Enter the new password in the middle line.
- Repeat the new password in the bottom line.
- Confirm the action with ***OK*** (the new password is immediately activated) or end the action by clicking on ***Cancel*** without changing the password.

5.2.6.4 Removing Password Protection

- To remove the password protection in the device again, select the function ***Remove password protection***. The following window opens:



CMWIN

Please enter your password:

- Enter the currently valid password.
- Confirm the action with ***OK*** (password protection is immediately deactivated) or end the action by clicking on ***Cancel*** without deactivating the password protection.

5.2.6.5 Send Commands

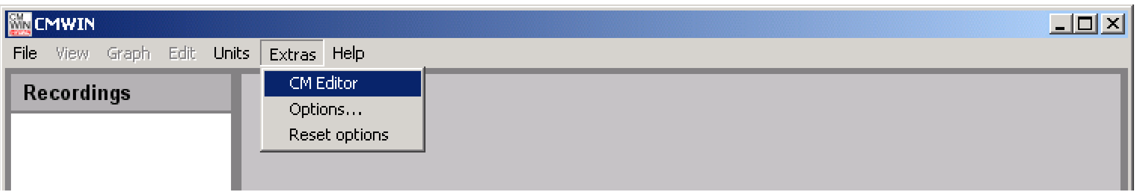
- With this menu option it will be possible with future device versions to send direct commands to the CMU 1000 in order to initiate certain functions there. This menu option has no function at present.

6 CM Editor

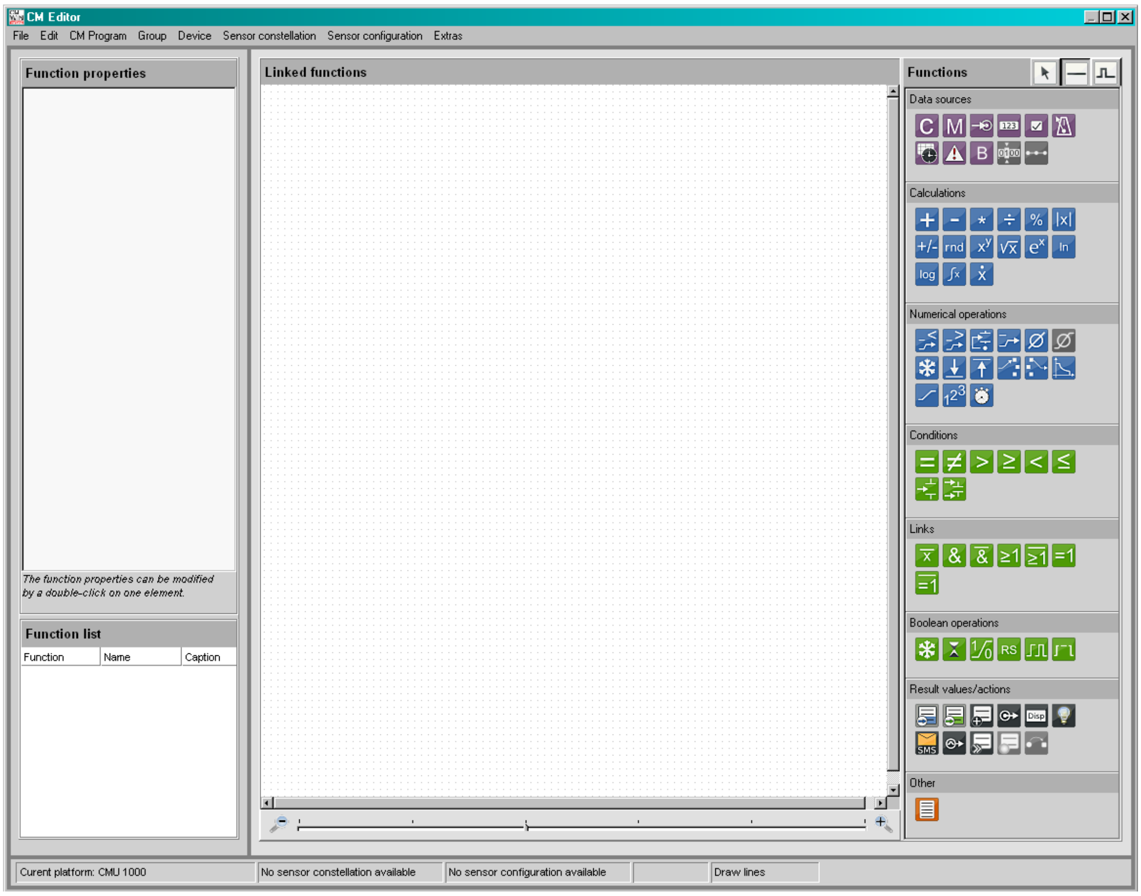
The CMU 1000 processes your program in continuous cycles. You generate the program with the ***CM Editor*** and load it afterwards into the CMU 1000.
The CM Editor is a constituent part of the HYDAC PC software ***CMWIN*** starting with Version 3 and provides you with various tools and functions for designing, integrating and testing your CM program.

To open the Editor, proceed as follows:

- Start the HYDAC PC software **CMWIN**
- In the ***Extras*** Menu, select the "**CM Editor**" option.



- The following screen opens:



The function properties can be modified by a double-click on one element.

The menu structure and window properties of the Editor are explained below in greater detail:

6.1 Menu Bar

The menu bar of the CM Editor is tailored to the MS Windows user interface and contains the following menu structure:

6.1.1 File

- With "**New**", you can establish for which platform (CM device) the CM program is to be created before beginning the actual creation of the CM program. The program functions which are not available for the selected platform will be suppressed in gray tones in the ***Functions*** window and will not be able to be used during the program generation.

- With "**Platform**", you can establish for which platform (CM device) the CM program is to be created during the creation of a CM program. The program functions which are not available for the selected platform will be suppressed in gray tones in the ***Functions*** window and will not be able to be used during the program generation.

- Pressing "**Open**" allows you to open an already generated and stored CM program. CM programs use the file extension ***.hecmp***. Select the desired file in the corresponding path for this purpose.

- Select "***Save***" to save a CM program. If the recording has not yet been saved, the “**Save as**…“ window will open. Enter the desired file name for this purpose in the respective path.

- To save a file that has not yet been saved or to save a file that has already been saved under a different name, select "**Save as ...**", Enter the desired file name for this purpose in the respective path.

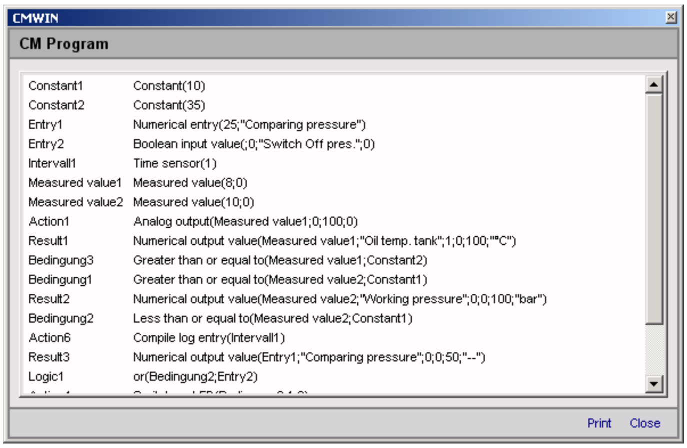
- To integrate an existing program that has already been saved into the current program generation process, select "**paste from file**". Select the desired file in the corresponding path for this purpose.

- Press "**Print**" to print out the content of the ***Linked functions*** window (= program printout).

- Press "**Quit**" to close the CM Editor.

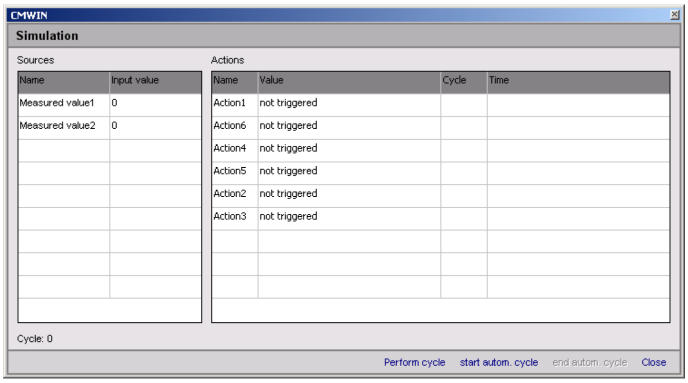
6.1.2 CM Program

- After "**Display**" is selected, a window opens in which all of the functions used in the currently opened CM program will be listed, together with inscriptions and parameters.
  The list can be printed out by selecting "**Print**".
  Pressing "**Close**“ takes you back to the CM Editor.



Constant1	Constant(10)
Constant2	Constant(35)
Entry1	Numerical entry(25;"Comparing pressure")
Entry2	Boolean input value(;0;"Switch Off pres.";0)
Intervall1	Time sensor(1)
Measured value1	Measured value(8;0)
Measured value2	Measured value(10;0)
Action1	Analog output(Measured value1;0;100;0)
Result1	Numerical output value(Measured value1;"Oil temp. tank";1;0;100;"°C")
Bedingung3	Greater than or equal to(Measured value1;Constant2)
Bedingung1	Greater than or equal to(Measured value2;Constant1)
Result2	Numerical output value(Measured value2;"Working pressure";0;0;100;"bar")
Bedingung2	Less than or equal to(Measured value2;Constant1)
Action6	Compile log entry(Intervall1)
Result3	Numerical output value(Entry1;"Comparing pressure";0;0;50;"--")
Logic1	or(Bedingung2;Entry2)

- With "**Simulate**", you can simulate and/or test the CM program that is currently open. The ***Simulation*** window opens for this purpose. No connection with the CMU 1000 is required for the simulation.



Sources

Name	Input value
Measured value1	0
Measured value2	0

Actions

Name	Value	Cycle	Time
Action1	not triggered		
Action6	not triggered		
Action4	not triggered		
Action5	not triggered		
Action2	not triggered		
Action3	not triggered		

Cycle: 0

- All of the input signals used in the CM program are listed one above the other in the ***Name*** column in the left-hand part of the Simulation window.

- You can assign a specific value to each input in the ***Input value*** column.

- All of the actions used in the CM program are listed one above the other in the ***Name*** column in the right-hand part of the Simulation window.

- The current status of each action is displayed in the ***Value*** column.

- The cycle number of the last status modification is displayed in the ***Cycle*** column.

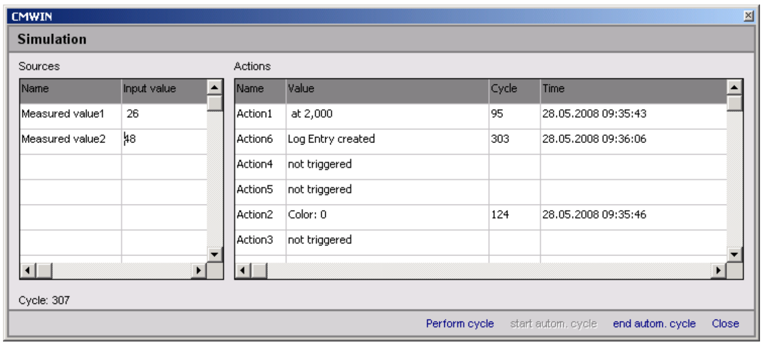
- The date and time of the last status modification is displayed in the ***Time*** column.

- With "**Perform cycle**", you can start the simulation for a single processing cycle and then view the thereby resulting status modifications of the actions afterwards.

- With "**Start autom. cycle**“, you start a permanent, continuous program simulation. You can change the input values during the simulation any way you like and observe the status modifications of the actions.

- Mit "**End autom. cycle**“, you stop the permanent, continuous program simulation.

- Pressing "**Close**“ takes you back to the CM Editor.



Sources

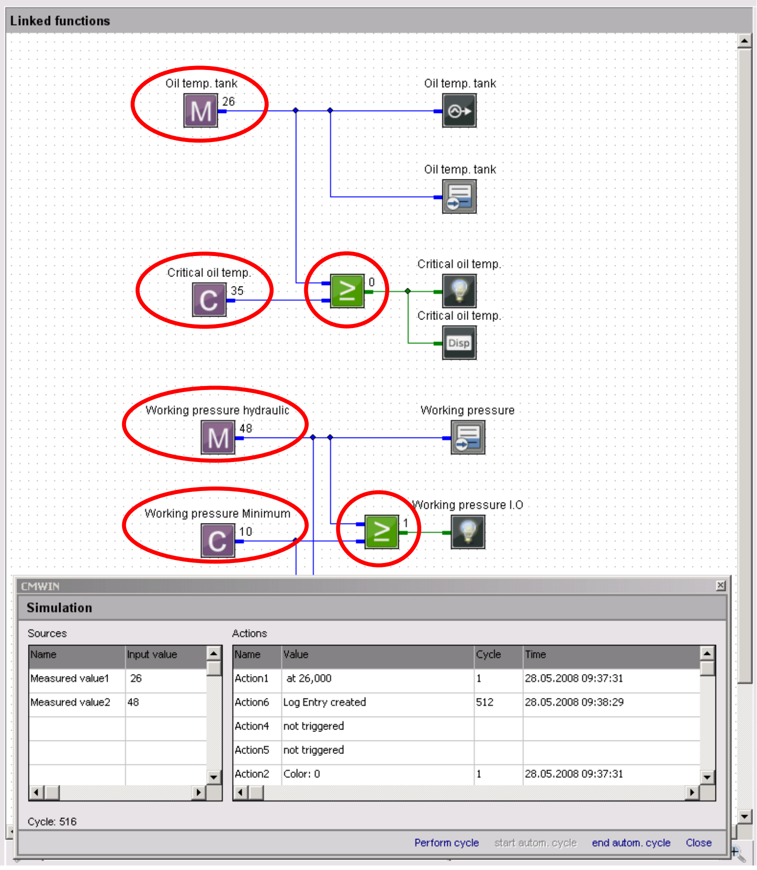
Name	Input value
Measured value1	26
Measured value2	48

Actions

Name	Value	Cycle	Time
Action1	at 2,000	95	28.05.2008 09:35:43
Action6	Log Entry created	303	28.05.2008 09:36:06
Action4	not triggered		
Action5	not triggered		
Action2	Color: 0	124	28.05.2008 09:35:46
Action3	not triggered		

Cycle: 307

- During the simulation, the assigned input values and action statuses are also displayed in the ***Linked Functions*** window at the corresponding symbols.



Sources

Name	Input value
Measured value1	26
Measured value2	48

Actions

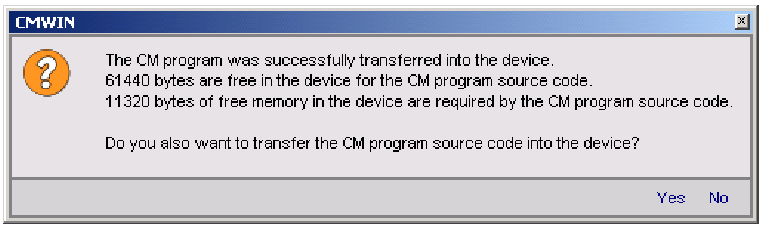
Name	Value	Cycle	Time
Action1	at 26,000	1	28.05.2008 09:37:31
Action6	Log Entry created	512	28.05.2008 09:38:29
Action4	not triggered		
Action5	not triggered		
Action2	Color: 0	1	28.05.2008 09:37:31

Cycle: 516

- You can transfer the currently opened CM program to the CMU 1000 with "**Transfer into device**“.

Note!
Only error-free programs can be transferred into the CMU 1000.

- The following message appears after the program has been successfully applied:



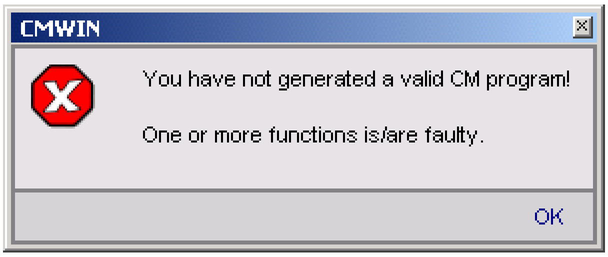
The CM program was successfully transferred into the device.
61440 bytes are free in the device for the CM program source code.
11320 bytes of free memory in the device are required by the CM program source code.

Do you also want to transfer the CM program source code into the device?

- Here you can select whether you also wish to transfer the source code of the program into the CMU 1000.

Note!
If you transfer the source code into the CMU 1000 as well, then it is possible for it to be read out again from every other connected PC and modified! If the source code is not transferred as well into the CMU 1000, then the program cannot be read out from another PC.

- The following message will appear if the CM program is not error-free:



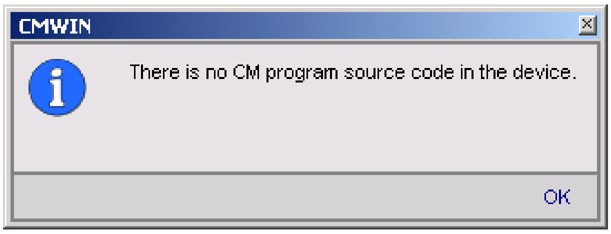
You have not generated a valid CM program!

One or more functions is/are faulty.

- Eliminate the error(s) found in the CM program (see in this connection Chap. 6.2.2., "Function List" Window and Chap. 8, "Error messages") and transfer the program again.

Note!
If you transfer a program and a power failure occurs during the transfer, then the program will not be saved in the CMU 1000 after power is restored. For that reason, first save your program on the PC on which it was originally generated or modified before making the transfer.

- You can transfer and then edit the CM program currently available in the CMU to your PC with "**Receive from device**". This will however only work if the CM program source code from the original creator has also been transferred into the CMU 1000.
  If no source code is available in the CMU 1000, then the following message will appear:



CMWIN

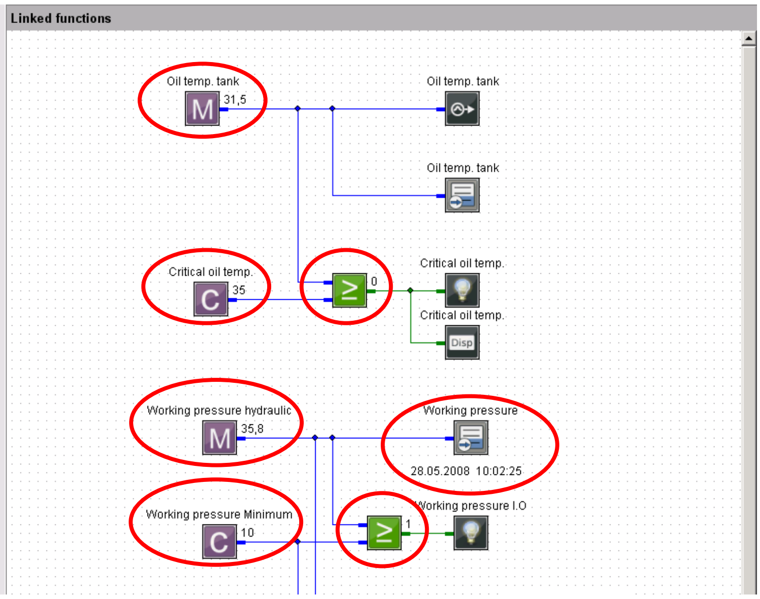
There is no CM program source code in the device.

- The "**Online Debugging**" function is a tool for observing the CM program as well as for finding, diagnosing and eliminating possible errors in the CM program and/or in the connected peripherals.

  ▪ The following prerequisites must be fulfilled for this purpose:
    - The CMU 1000 must be connected with the PC and the CMWIN software.
    - The CM program opened in CMWIN and the one active in the CMU 1000 must be identical.

  ▪ If the "Online Debugging" mode is active, then the following values will be displayed at the respective functions:
    - Measured values
    - Constants
    - Switching status of logical links
    - Date and time when initiating the corresponding event

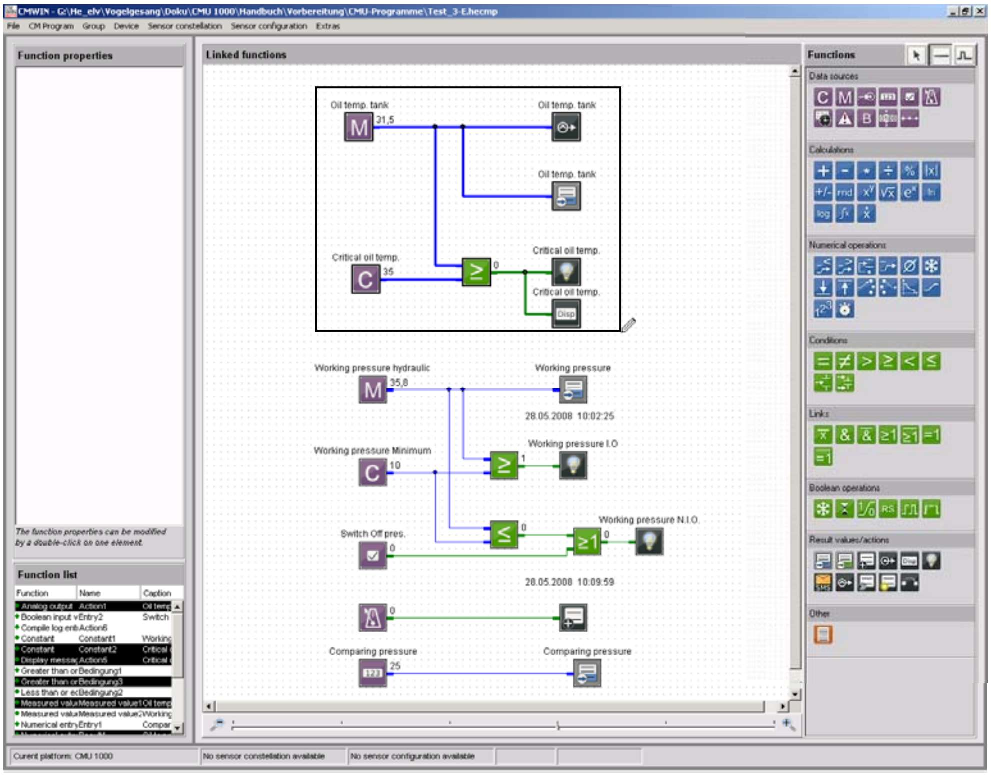
  ▪ The display in "Online Debugging" mode is shown in exemplary fashion in the following illustration:



6.1.3 Grouping

- With ***Create grouping*** you can join several functions to make an interconnected unit and transfer and copy these as a block.

  ▪ First mark the functions to be grouped by enclosing them in a frame created with the help of the cursor.



  ▪ Afterwards, select "***Group***" in the menu bar and then "***Create grouping***" in the drop-down menu that appears.

- Pressing "***Cancel grouping***" releases the functions linked together in the group from one another again.

  ▪ To accomplish this, mark the desired group with a mouse click on any one of the functions within the group.

  ▪ Afterwards, select "***Group***" in the menu bar and then "***Cancel grouping***" in the drop-down menu that appears.

6.1.4 Device

The "**Connection**" function can be used to set up a connection between the PC and a CMU 1000 from the CM Editor.

The function is identical with the connection setup in the CM Manager. See in this connection Chap. 5.2.1. ff.

6.1.5 Sensor Constellation

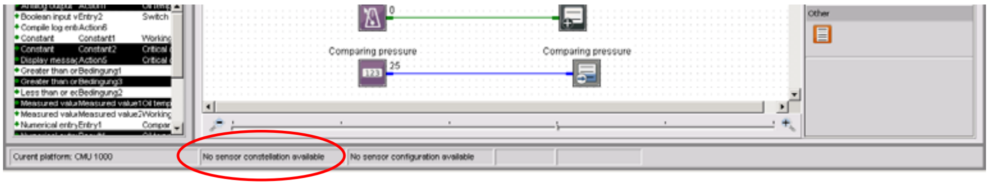
Reliable system monitoring requires that it be ensured that precisely the same sensors are connected during operation as were connected at the time the CMU 1000 was configured.
The ***Sensor constellation*** is used for this purpose. The sensor constellation is a monitoring instrument for the connected sensor system, i.e. it performs a continuous comparison between the connected "Actual" sensor system and the specified "Nominal" sensor system.

The sensor constellation is optional in the CMU 1000 and not mandatory to have. If however a sensor constellation has been saved, then the connected sensor system must match it.
When there is an activated sensor constellation, then an accidental, connected "incorrect" sensor will be recognized, thus preventing a situation in which the CM program is working with incorrect information.
A sensor constellation can be saved in files and loaded from files, and can be both received and transferred by the CMU 1000.
The constellation files all have the extension ****.hescf***.

The sensor constellation contains the following data:

- Quantity of connected sensors
- Quantity of subchannels for each connected sensor
- Status of each individual subchannel (active / inactive)
- Sensor class (Analog / HSI / SMART)
- Units of the individual measured values
- Name of each sensor
- Device designation of each sensor

If a sensor constellation is available in the CM Editor, then you can use the correct names in connection with the measurements during program generation. It will then be the case that only those sensors and subchannels that are actually present will be available and accessible for adjustment.
A status message at the lower edge of the window of the CM Editor indicates whether or not a sensor constellation is available.


Boolean input vEntry2 Switch
Compile log entAction6
Constant Constant1 Working
Constant Constant2 Critical
Display messagAction5 Critical
Greater than orBedingung1
Greater than orBedingung3
Less than or eqBedingung2
Measured valuMeasured value1Oil temp
Measured valuMeasured value2Working
Numerical entryEntry1 Compar
0
Comparing pressure
Comparing pressure
123
25
Other

Curent platform: CMU 1000

No sensor constellation available

No sensor configuration available

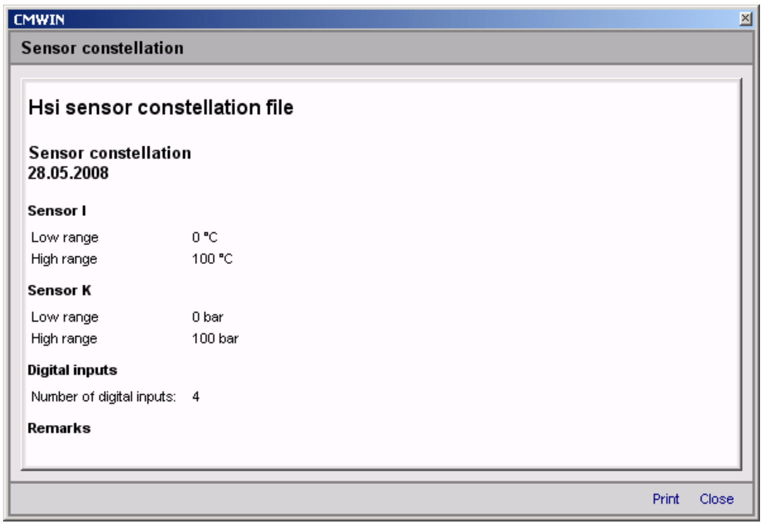
- With the function ***Apply from file,*** you can open a saved sensor constellation file and use it in the CM Editor.
  To accomplish this, enter the corresponding path and file name in the window that opens and then click on ***Open***.

- With the function ***Apply from device***, you transfer one of the sensor constellations stored in the CMU 1000 to your PC, after which you can use it in the CM Editor.

- With ***Uninstall*** you delete the currently available sensor constellation in the CM Editor, after which it is no longer available for further use when generating programs.
  No saved constellation files are deleted!

- To save a constellation file, select ***Saving to a file***. Enter the respective path and desired file name for this purpose in the window that opens.

- When the ***Display*** function is selected, a window opens in which the complete sensor constellation is displayed.



Sensor constellation

Hsi sensor constellation file

Sensor constellation
28.05.2008

Sensor I

Low range	0 °C
High range	100 °C

Sensor K

Low range	0 bar
High range	100 bar

Digital inputs

Number of digital inputs:	4

Remarks

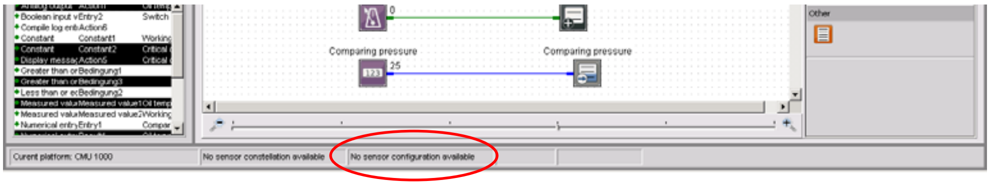
6.1.6 Sensor Configuration

Sensor configuration is understood to mean all of the input-relevant settings in the CMU 1000 (*Peripherals settings*). The sensor configuration can only be generated with the CMU 1000 itself (offline generation in CMWIN is not possible). All necessary sensors must be connected in order for this to be possible.

A sensor configuration can be saved in files and loaded from files, and can be both received and transferred by the CMU 1000.
The configuration files have the extension ***.hedcf**.

The sensor configuration contains the following data:
- Sensor name
- Mode
- Measurement range
- Decimals
- Units of the individual measured values

A status message at the lower edge of the window of the CM Editor indicates whether or not a sensor configuration is available.


Boolean input vEntry2 Switch
Compile log enbAction6
Constant Constant1 Working
Constant Constant2 Critical
Display messagAction5 Critical
Greater than orBedingung1
Greater than orBedingung3
Less than or eqBedingung2
Measured valuMeasured value1Oil temp
Measured valuMeasured value2Working
Numerical entryEntry1 Compar
0
Comparing pressure
25
123
Comparing pressure
Other

Curent platform: CMU 1000 | No sensor constellation available | No sensor configuration available

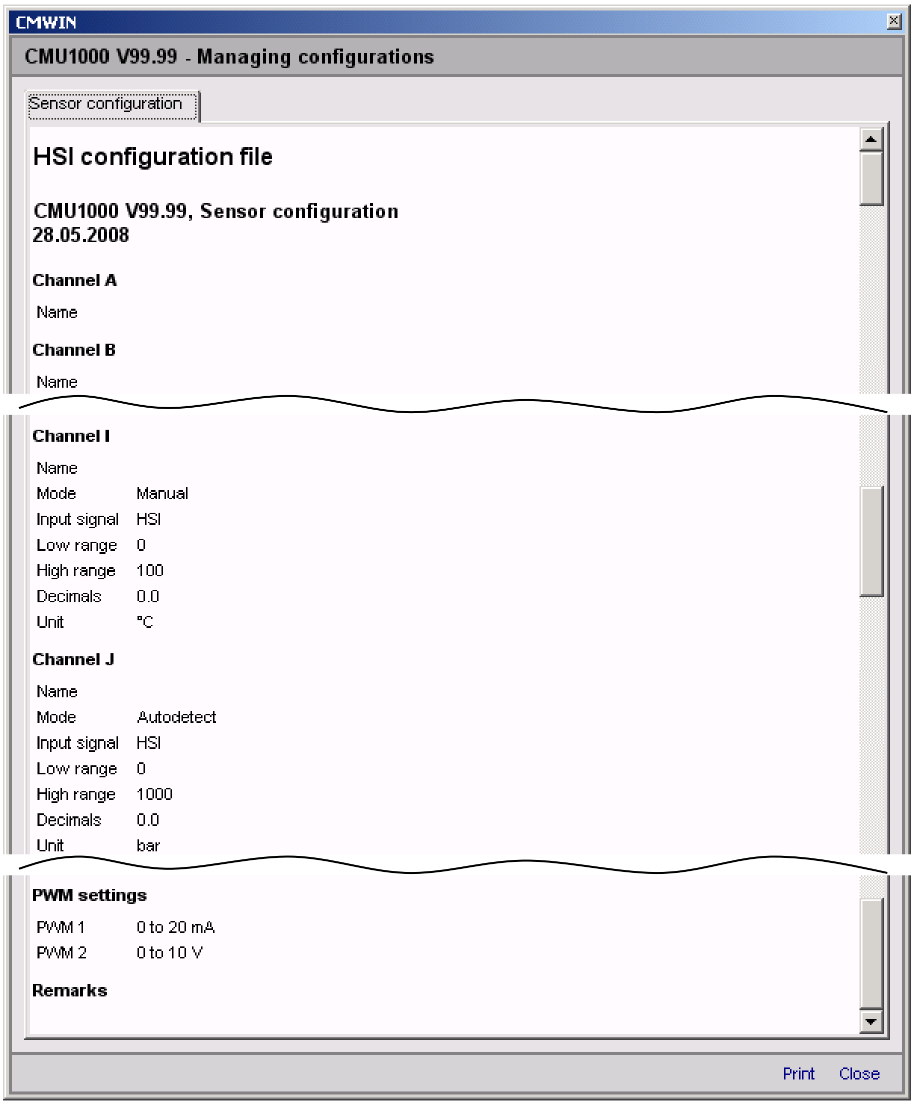
- With the function ***Apply from file,*** you can open a saved sensor configuration file and use it in the CM Editor.
  To accomplish this, enter the corresponding path and file name in the window that opens and then click on ***Open***.

- With the function ***Apply from device***, you transfer one of the sensor configuration
  stored in the CMU 1000 to your PC, after which you can use it in the CM Editor.

- With ***Uninstall*** you delete the currently available sensor constellation in the CM Editor, after which it is no longer available for further use when generating programs.
  No saved constellation files are deleted!

- To save a configuration file, select ***Saving to a file***. Enter the respective path and desired file name for this purpose in the window that opens.

- When the ***Display*** function is selected, a window opens in which the complete sensor configuration file is displayed.



HSI configuration file

CMU1000 V99.99, Sensor configuration
28.05.2008

Channel A

Name

Channel B

Name

Channel I

Name	
Mode	Manual
Input signal	HSI
Low range	0
High range	100
Decimals	0.0
Unit	°C

Channel J

Name	
Mode	Autodetect
Input signal	HSI
Low range	0
High range	1000
Decimals	0.0
Unit	bar

PWM settings

PWM 1	0 to 20 mA
PWM 2	0 to 10 V

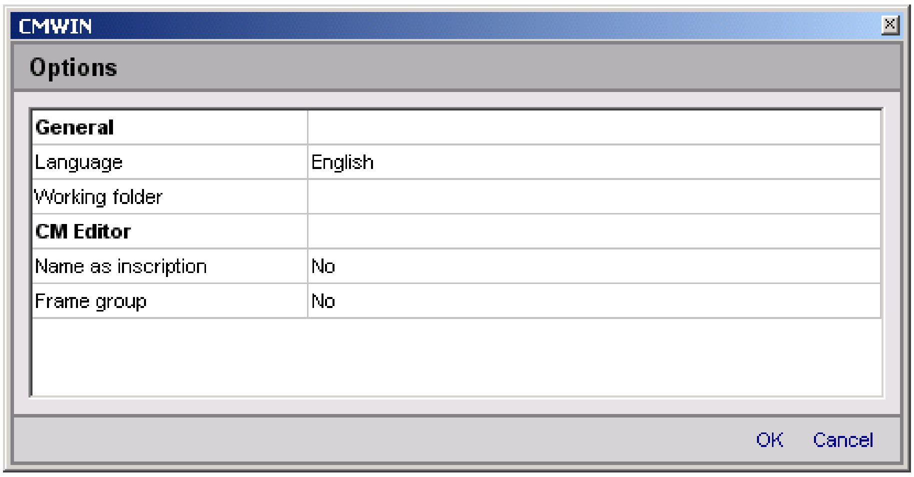
Remarks

Note!
With a CM program, one has the opportunity of integrating not only a configuration file but also a constellation file, so that one can define the configuration and the constellation at the same time the CM program is transferred into the CMU 1000. Both parts are however optional and need not be present in the CM program.

6.1.7 Extras

- The following window opens with the function ***Options*** is selected:
  The selection buttons at the right-hand edge of the window appear after clicking in the respective selection field.

  - In the ***Language*** field, you can select between the options of German, English and French for choosing the CMWIN system language.



CMWIN

Options

General	
Language	English
Working folder	
CM Editor	
Name as inscription	No
Frame group	No

  - In the field ***Working folder*** you define the path for saving the CMWIN files (CM programs, recordings, constellation and configuration files, ...).

  - In the field ***Name as inscription*** you define whether or not the respective function name (e.g. Measured value 2, Action 4) is to be displayed as function inscription in the "Linked functions" window.
    If "No" is selected, then you have the option of entering an inscription text manually into the function parameters.

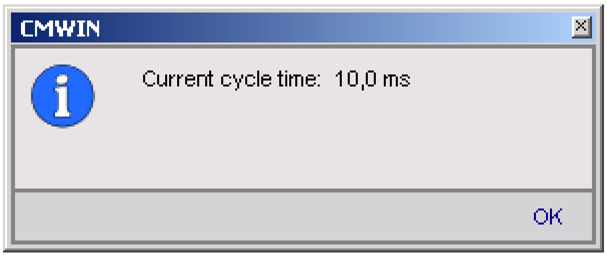
  - In the field ***Frame group*** you define whether or not a frame is to appear around generated groups in the CM program.

  - Clicking on ***OK*** applies the settings and returns you to the main CMWIN window.
    Pressing ***Cancel*** takes you back without applying any changes.

- By using the ***Reset Options*** function you can reset all the modified options and settings to the standard settings.

- Selecting the ***Display cycle time*** function causes the following window to open and the current cycle time of the CM program to be displayed.

Example:



CMWIN

Current cycle time: 10,0 ms

6.2 Window Divisions

The graphics interface of the CM Editor is divided into the following elements:

6.2.1 "Function Properties" Window

The properties of the functions currently selected in the CM program are displayed in this window. These include:

- Function name (e.g. Action 2; Constant 5; Measured value 12)
- Function type (z.B. Constant, Measured value, Timer)
- Specific properties (parameter settings)
- Comment

6.2.2 "Function List" Window

This window lists all of the functions used in the CM program with the following specifications:

- Function type (z.B. Constant, Measured value, Timer)
- Function name (e.g. Action 2; Constant 5; Measured value 12)
- Inscription (e.g. Working pressure N.I.O.)

6.2.3 "Linked Functions" Window

This window contains the actual CM program. The display can be zoomed in or out with the scroll bar on the lower edge of the window.

6.2.4 "Functions" Window

This window contains all of the functions available for program generation, sorted according to:

- Data sources
- Calculations
- Numerical operations
- Conditions
- Links
- Boolean operations
- Result values/actions
- Other

7 CM Program Functions

A CM program consists of many individual functions that are linked with one another and that are processed and evaluated in cyclical fashion.

7.1 General Information Concerning Functions

A function has *Inputs*, *Outputs* and *Parameters*. This means, for example, the function "Median value" reads a numerical value at the input, forms a median value above it and then displays this at the output. A parameter is used to define the amount of time for which the calculation is rendered.

7.1.1 Inputs / Outputs

For most functions, the outputs change during the running time, depending on the input. Functions are linked with one another in the Editor. This means, for example, that the output of a function can be linked with the input of a different function. It is possible to make one output dependent on several inputs, but not several outputs on one input.
There are two types of inputs/outputs, depending on the value type: *Numerical* and *Boolean*. A "Boolean output" and only be linked with a "Boolean input" and a numerical output can only be linked with a "numerical input".

7.1.1.1 Numerical Values

A numerical value is a decimal number, i.e. a numerical value with an optional algebraic sign and any given number of digits after the decimal point. It is accurate down to 7-8 significant places. That means that with a value of 2 million (7 places in front of the decimal point), the addition of a value of 0.001 (3 places after the decimal point) will not alter the numerical value. One would need accuracy down to at least 10 significant places for it to be altered.
Numerical inputs/outputs and the corresponding connection lines are shown in **blue**.

7.1.1.2 Boolean Values

A Boolean value is a logical status. There are only 2 statuses: "true" or "1" and false or "0".
Boolean inputs/outputs and the corresponding connection lines are shown in **green**.

7.1.2 Parameters

Parameters are defined in the Editor and do not change during the running time. An exception to this are the input parameters, which can be modified in a menu on the CMU or with a connected PC during the running time.

Parameters have one of the following types:

7.1.2.1 Numerical Parameters

A numerical parameter is a decimal number in accordance with the inputs/outputs.

7.1.2.2 Whole Number

A whole number is a natural number, i.e. it has no digits after the decimal point. As a rule, no negative numbers are permitted either. Whole numbers are used for example for numbering purposes.

7.1.2.3 Entry List

An entry from a list is selected for the input list type. The quantity and the type of list entries is dependent on the respective parameter.

7.1.2.4 Boolean Parameters

A Boolean parameter has, as was already described in connection with the inputs/outputs, only two logical statuses: "0" and "1". Nevertheless, it is not "0" and "1" that are set in the Editor, but rather such terms as "No" or "Yes", "Inactive" or "Active", "Off" or "On", depending on the context.

7.1.2.5 Character String

A character string is an arbitrary text, the length of which is usually limited. In addition, preceding and subsequent empty spaces are usually removed automatically.

7.1.2.6 Values Table

A values table is a table with several values, whereby each value is positioned in a line of its own.
Furthermore, it is also possible to enter value pairs. The individual values of a value pair are then separated from one another with a colon ":".

7.1.2.7 Time

The specification of a time of the day is accomplished in the country-specific format that is set in Windows.

7.2 Data Sources

7.2.1 Numerical Constant

The *Numerical Constant* function supplies a numerical value which is defined in the Editor and which does not change during the running time.

That means that the *Value* parameter entered in the Editor is output during the running time at the output.

Inputs:	-	
Outputs:	y:	(Numerical)
Parameters:	p_1:	Value (Numerical)

7.2.2 Measured Value

The *Measured Value* function provides the current measured value of a connected sensor.

The *Sensor connection* parameter is used to define the sensor; the channel of the sensor with *Subchannel*. If a connected sensor has no subchannel, then no selection is possible at this position.

If a sensor constellation has been saved, then only the active connections are displayed in the input lists for the *Sensor connection*, each of them with the respective sensor name. The subchannels available for this sensor are then displayed at *Subchannel*, each of them with name and unit.

If no sensor constellation is saved, then Ports "A" to "R" are offered for selection at *Sensor connection* and the channels "1" bis "32" *a Subchannel*.

Inputs:	-	
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Sensor connection* (input list)
	p_2:	*Subchannel* (input list)

7.2.3 Digital Input

The *Digital input* function supplies the status of a digital input.

The *Input terminal* parameter is used to define which digital input port is used.

Inputs:	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Input terminal* (input list)

7.2.4 Numerical Entry

The function *Numerical input* supplies a numerical value which can be adjusted in the Parameters menu of the CM device. As an alternative, it can also be set through a connected PC.

The *Inscription* parameter is used as a menu option in the input menu for this purpose. The permissible input range runs from -2,000,000.000 to +2,000,000.000.

Modifications made in the CM device during the running time also remain in effect after switch-off. The value that is set under *Start value* will used after the first-time transfer of the CM program to the CM device until the first change is made.

Inputs:	-	
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Starting value* (Numerical)
	p_2:	*Inscription* (character string)

7.2.5 Boolean Entry

The *Boolean input* function supplies a Boolean value, which can be set in the Parameters menu of the CM device. As an alternative, it can also be set through a connected PC.

The *Inscription* parameter is used as a menu option in the input menu for this purpose. The input is accomplished on the CM device by selecting "Yes" or "No".
Modifications made in the CM device during the running time also remain in effect after switch-off. The value that is set under *Start value* will used after the first-time downloading of the CM program into the CM device until the first change is made.
The Functionality paramter defines how the input is interpreted. The following settings are possible:

Switch
A menu option is generated on the CM device with which the input value can be switched on or off. . This functionality is used to active certain paths in the evaluation logic.

Key
When the value on the CM device is switched on, then only one impulse is generated in the evaluation logic and the option switches itself off in the menu again immediately. A key function is simulated, so to speak. This functionality can be used to trigger events in the evaluation logic.

Inputs:	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Starting value* (Boolean)
	p_2:	*Inscription* (character string)
	p_3:	*Functionality* (entry list)

7.2.6 Time Sensor

The *Time sensor* is a function which generates an impuls at an adjustable interval (e.g. every minute, every 5 minutes), thus setting its Boolean output to "1" for a cycles and then back to "0".

The following settings are possible fo the *Interval* parameter:

- *1; 2; 5; 10; 15; 30 Seconds*,
- *1; 2; 5; 10; 15; 30 Minutes*,
- *1; 2; 6; 12; 24 Hours*.

The output impulse is thereby always accomplished in synchronized fashion with the time of day. If, for example, "6 hours" is set, then an impulse will be generated at 6 AM, 12 AM, 6 PM and 12 PM.

If an impulse is required at particular times of the day, then you should use a clock timer with a downstream impulse generator instead of a time sensor (see Chap. 7.2.7).

Inputs:	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Interval* (input list)

7.2.7 Clock Timer

The *Clock timer* is a function which switches on its Boolean output at a certain time during a month and then switches it off again at a different point in time.

The switch-on time is set with the parameters *Switch-on day* and *Switch-on time*, while the switch-off time is set with the parameters *Switch-off day* and *Switch-off time.*

A weekday "Monday" to "Sunday" can be set. The setting "Daily" is also possible. The "Daily" setting is only possible if both days are set to "Daily". The Editor prevent erroneous inputs: If, for example, the switch-on day is changed from "Monday" to "Daily", then the switch-off day is automatically set to "Daily".

If switch-on time and switch-off time are identical, the the output will be switched on for precisely this specified second.
If only one impulse is required, i.e. if the output in intended to be set for exactly one cycle to "1" and then back to "0", then use a downstream impulse generator (see Chap. 7.8.6).

Inputs:	-	
Outputs:	y:	(Boolean);
Parameters:	p_1:	*Switch-off day* (input list)
	p_2:	*Switch-off time* (time of day)
	p_3:	*Switch-on day* (input list)
	p_4:	*Switch-on time* (time of day)

7.2.8 Error Event

An error handling can be implemented with the function *Error event*. The Boolean output is switched to "1" in the event that an error condition is present. The output is switched back to "0" if the error goes away.

The type of error event can be set with the *Event* parameter.

The following events are possible:

Below signal range
A sensor has cable breakage, for example

Above signal range
One sensor lies above the signal range with its signal

Incorrect sensor constellation
At one sensor connection port there is either no sensor connected, or an incorrect one.

Numerical error
An error occured during a calculation, e.g. division by 0 or square root of a negative number, logarithm from 0, etc.

Cycle time exceeded
The set cycle time was exceeded.

Inputs	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Event* (input list)

7.2.9 Boolean Constants

The Boolean Constant function supplies a Boolean value which is defined in the Editor and which does not change during the running time. That means that the parameter entered in the Editor is output during the running time at the output.

Inputs	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	Value (Boolean)

7.2.10 State-bit

The function *Show Stat-bit* provides the status of the bit defined by the "Sensor port" and "Bit location" parameters in the status code of the selected SMART sensor (status code is sensor-specific) at the Boolean output.

Inputs	-	
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Sensor port* (Channel A .. Channel H)
	p_2:	*Bit location* (0 .. n -depending on sensor-)

Additional points for Sequence controls in the CMU

The two interrelated function components *Sequence* and *Transition* can be used to implement the functionality of Sequence controls or finite state machines.

7.2.11 Sequence

The *Sequence* component has a numerical output. This represents the current state of the Sequence. At the same time, a number is allocated to all states. The term 'Sequence' has been taken from control engineering. Typically several steps are completed one after the other in a Sequence. Branches and backward jumps are however possible. A sequence can adopt several states which are numbered in ascending order starting at "0". "0" is always the state following switch-on. A change in state is always triggered by a *transition*.

Inputs:	-	
Outputs:	y:	(Numerical)
Parameters:	-	
	-	

7.2.12 Transition (in the field "Result values / Actions")

The *Transition* component serves to bring about a change from one state to another in a Sequence. A transition has a Boolean input and has 4 parameters.

Inputs: x: (Boolean)
Outputs: -
Parameters: p_1: *Sequence*
Name of the related Sequence in the application program

p_2: *Current state*
Number of the state while the input condition is scrutinized.
With the "-1” (ignore) setting, the current Sequence state is ignored, i.e. the input condition is continually scrutinized, irrespective of the current state of the sequence.

p_3: *Successor state*

- State which is adopted once the input condition occurs.

p_4: *Dwell time [s]*
Indicates how many seconds the Sequence must be in the current state before the input condition is scrutinized for the successor state.

For one Sequence, several transitions with identical current and successor states but with differing input conditions can be used in the application program. This enables branching into different Sequence cascades within a Sequence.

7.3 Numerical Calculations

7.3.1 Addition

The *Addition* function returns the sum of the two input values at the output:

$y = x_1 + x_2$

Inputs	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.2 Subtraction

The *Subtraction* function returns the difference between the two input values at the output:

$y = x_1 - x_2$

Inputs	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.3 Multiplication

The *Multiplication* function returns the product of the two input values at the output:

$y = x_1 * x_2$

Inputs	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.4 Division

The *Division* function returns the quotient of the two input values at the output:

$y = x_1 / x_2$

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.5 Division Remainder

The *Division remainder* function returns the division remainder of the two input values at the output (the modulo). The division remainder is determined by performing a whole number division x_1 / x_2 and outputting the remainder of this division as output value.

If the input x_1 counts upward, e.g. sequentially by 1, and the input x_2 amounts to 5, the the output will count around from 0 to 4.

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.6 Absolute Value

The *Absolute value* function returns the input value without algebraic sign at the output:

y = |x|

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.7 Change of Algebraic Sign

The *Change of algebraic sign* function returns the inverse input value at the output:

y = -x

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.8 Rounding

The *Rounding* function returns the rounded-off input value at the output. With this function, one can not only round off to whole decimal places, but also to whole-number multiples of a step.

The size of the step will be specified in the *Step* parameter.

If the step size is 20, for example, then all values between -10 and 10 will be rounded off to 0, all values between 10 and 30 to 20, and so on.

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Step* (Numerical)

7.3.9 Raising to a Higher Power

x^y

The *Raising to a higher power* function supplies the power of the input value at the output.
The exponent is set with the *Exponent* parameter.

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Exponent* (Numerical)

7.3.10 Square Root

$\sqrt{x}$

The *Square root* function supplies the square root of the input value at the output.

If the input value is negative, then the value 0 will be supplied at the output and an error flag will be set. One can react to this situation with the *Error event* function (see Chap. 7.2.8).

If a negative input can occur in everyday practice, then you should structure the corresponding behavior in accordance with your preferences with the functions *Absolute value, If-then-else* and *Less than (see Chap.* 7.3.6, 7.4.4 and 7.6.5).

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.11 Power at Base e

e^x

The *Power at base e* function supples the power at base e at the output.
The input value is used as the exponent.

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.12 Natural Logarithm

ln

The *Natural logarithm* function supplies the logarithm of the input value at base e at the output.

If the input value is negative or 0, then the value 0 will be supplied at the output and an error flag will be set. One can react to this situation with the Error event function (see Chap. 7.2.8).

If a negative input can occur in everyday practice, then you should structure the corresponding behavior in accordance with your preferences with the functions Absolute value, If-then-else and *Less than (see Chap.* 7.3.6, 7.4.4 and 7.6.5).

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.13 Decade Logarithm

The Decade logarithm function supplies the logarithm of the input value at base 10 at the output.

If the input value is negative or 0, then the value 0 will be supplied at the output and an error flag will be set. One can react to this situation with the Error event function (see Chap. 7.2.8).

If a negative input can occur in everyday practice, then you should structure the corresponding behavior in accordance with your preferences with the functions Absolute value, If-then-else and *Less than (see Chap.* 7.3.6, 7.4.4 and 7.6.5).

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.3.14 Integral

The *Integral* function supplies the integral of the input value over time at the output. The output is always calculated with the unit of seconds. This means that the input value 6 causes the output to increase every second by 6. The trapezoidal rule is applied to make the calculation.

The function still has a Boolean reset input. If the value "1" is pending there, then the value "0" will be set up at the output.

Furthermore, the integral function also has an automatic anti-wind-up mechanism. It is with this that a parameterizable *Lower limit* and *Upper limit* are set for the output.

This function adopted from control engineering has the following background:
If a control variable is not achieved, then the I ratio continues to integrate further. The controller then may require under certain circumstances a very long time to exit this range again if the actuating variable reverses its algebraic sign. This can lead to very unstable behavior.

Inputs:	x_1:	*Input value* (Numerical)
	x_2:	*Reset input* (Boolean)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Lower limit* (Numerical)
	p_2:	*Upper limit* (Numerical)

7.3.15 Differential Quotient

The *Differential quotient* function supplies the derivation of the input value over time at the output. The output is always calculated with the unit of seconds. This means that an increase of the input value from 5 to 6 in one second will yield an output value of 1.

The differential quotient is formed and filtered numerically by the differential quotient. This filtering is necessary for the following reasons:

In view of the fact that the input values usually arise from a quantized measured value, e.g. in connection with the digitization of an analog quantity, these values will have a so-called quantizing distortion. This means that the digitization causes the value fluctuate in terms of resolution. For example, with a 12-bit resolution, a value of 600 bar that was resolved with 12 bit will fluctuate back and forth by 0.15 bar. If the differential quotient is formed now every millisecond, then this quantizing distortion will be amplified by a factor of 1000. That means that, without filtering, the output would jump back and forth between + and - 150 bar/s.

The filter can be set with the *Filtering* parameter. The setting corresponds thereby to the time range during which the filtering takes place. Nonetheless, no pure median value formation will be used as a filter, but rather a special algorithm instead.

The following settings are possible:
• *switched off*, • *200 ms*, • *1 second* • *5 seconds.*

Inputs:	x_1:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Filtering* (entry list)

7.4 Numerical Operations

7.4.1 Minimum

The *Minimum* function supplies the smaller of the two input values at the output.

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.2 Maximum

The *Maximum* function supplies the larger of the two input values at the output.

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.3 Limit

The *Limit* function limits the input value x_1, and makes it available at the output.

The two limits are set by the input values x_2 and x_3. If x_1 is less than x_2, then x_2 will be output, if x_1 is greater than x_3, then x_3 will be output, otherwise x_1.

Inputs:	x_1:	*Input value* (Numerical)
	x_2:	*Lower limit* (Numerical)
	x_3:	*Upper limit* (Numerical)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.4 If - Then - Else

The function *If-Then-Else* has two numerical inputs x_1 and x_2 as well as a Boolean input x_3.

If the Boolean input value is "1", then the input value of x_1 is output at the output, otherwise the value of x_2.

Inputs	x_1:	(Numerical)
	x_2:	(Numerical)
	x_3:	(Boolean)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.5 Median Value

The Median value funtion supplies the arithmetical median value of the input values over an adjustable time range.

The time range is set in seconds with the *Time* parameter.

The formation of the median value is accomplished by means of the "Repeating Average" procedure. This means that, when the time setting is "2 seconds", for example, the input values are compiled for 2 seconds, then averaged and output at the output. The next median value interval begins afterwards. The median value remains stopped at the last median value thereby

Inputs	x:	*Input value* (Numerical)
Outputs:	y:	*Median value* (Numerical)
Parameters:	p_1:	*Time* (Numerical)

7.4.6 Extended Average

The *Extended Average* function supplies the arithmetic average of the numerical input value at the output. The interval over which the average value is generated, is controlled using a boolean reset input.

If the reset input has the value "1", then the input value is passed directly to the output. If the signal on the reset input falls from "1" to "0", then from this point, the arithmetic average of the input value is given at the output.

Inputs:	x_1:	*Input value* (Numerical)
	x_2:	*Reset* (Boolean)
Outputs:	y:	*Average* (Numerical)
Parameters:	-	

7.4.7 Note Value

The *Note value* function is used to hold on to certain values (to freeze them). It has one numerical and one Boolean input.

If the Boolean input value is "1", then the numerical input value is output at the output. If the Boolean input value is "0", then the last output value remains in effect.

If the value is only to be applied for one flank of the Boolean input, then you can put the function *Pulse generation* upstream from it (see Chap. 7.8.6).

Inputs	x_1:	*Input value* (Numerical)
	x_2:	*Switch value through* (Boolean)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.8 Note Minimum

The *Note minimum* function supplies as the output value the smallest value that the input value has yet reached.

If the input value is greater than the output value, then the output value remains unchanged. If the input value is less, then the output value will be reset.

The minimum can be reset with the Boolean input x_2. The input value wil be applied directly at the output for as long as this input is "1".

Inputs	x_1:	*Input value* (Numerical)
	x_2:	*Reset* (Boolean)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.9 Note Maximum

The *Note maximum* function supplies as the output value the largest value that the input value has yet reached.

If the input value is less than the output value, then the output value remains unchanged. If the input value is greater, then the output value will be reset.

The maximum can be reset with the Boolean input x2. The input value will be applied directly at the output for as long as this input is "1".

Inputs	x_1:	*Input value* (Numerical).
	x_2:	*Reset* (Boolean)
Outputs:	y:	(Numerical)
Parameters:	-	

7.4.10 Tabular Value

With the function *Tabular value*, the output value is obtained from a parameterized number table. The input value functions thereby as number of the table entry.

If the whole number value of the input is 0 or less, then the first value of the table will be output; if it is 1, then the second value; and so on up to the last tabular value. If the input value is greater than the number of tablular entries, then the last table entry will be output.

As a basic rule, the input value will be rounded off to a whole number.
The table is defined using the *Table* parameter. Each value must have a line of its own at the time of entry. Empty lines are removed automatically. The number of values will also be defined automatically on the basis of the available lines. It must be between 2 and 20.

The *Tabular value* function can be used for example as a downstream element of a division remainder in order to specify various values one after the other (see Chap. 7.3.5).

Inputs:	x:	*Index of the selected tabular value* (Numerical)
Outputs:	y:	*Selected value* (Numerical)
Parameters:	p_1:	*Table* (values table)

7.4.11 Tabular Index

The *Tabular index* function is the counterpart to the *Tabular value* function. The input value is sorted into a parameterizable numerical table, which must be organized in order of increasing values, and the number of the tabular entry is output.

If, for example, the first tabular entry is 4 and the second is 7.8, then a 0 will be output for all input values less than 0, the value 1 will be output for all values between 4 and 7.8, and the value 2 for all values greater than 7.8.

The table is defined using the *Table* parameter. Each value must have a line of its own at the time of entry. Empty lines are removed automatically. The number of values will also be defined automatically on the basis of the available lines. It must be between 2 and 20.

This function can be used for flexible range definition. Thus, for example, limit values can be defined in the table when a value is normal, suspicious, critical and very critical.

Inputs:	x:	*Value in the table* (Numerical).
Outputs:	y:	*Index of the value / Value range* (Numerical).
Parameters:	p_1:	*Table* (values table)

7.4.12 Characteristic Curve

The Characteristic curve function is used to convert input values from one range into another. The conversion can be subdivided into various segments through the specification of up to 20 nodes.

The *Table* parameter is used to specify the nodes. Each value pair is in a different line in the table. The values for x and y are separated by a colorn. The X values must be listed in ascending order. No x value may appear more than once. Empty lines are removed automatically. The number of values will also be defined automatically on the basis of the available lines. It must be between 2 and 20.

The ranges between the nodes are interpolated linearly; the values outside the nodes are extrapolated from the last segment. A limitation is easy to set up by simply setting another node nearby that has the same y value. If for example the range of 0 to 450 is to be converted to percent and at the same time limited to 0 and 100, then this is accomplished with the following value pairs:

-1:	0
0:	0
450:	100
451:	100

Inputs:	x:	*X value of the charactertistic curve* (Numerical).
Outputs:	y:	*Function value from the charactertistic curve* (Numerical)
Parameters:	p_1:	*Table* (values table)

7.4.13 Slope

The *Slope* function is used to prevent rapid value changes. Under stable conditions, the input value is shown at the output. Changes of the input value are however not passed along directly to the output, but rather only in small steps. Like a slope, so to speak. Different slopes can be defined thereby for positive and negative value modifications.

The parameters *Descending slope* and *Ascending slope* are used to specify the maximum permissible value changes per second.

If for example the value 5 is set for *Ascending ramp* and the input value jumps from 0 to 100, then the output will be only slowly increased, and it will take 20 seconds for the output value to reach 100.

Inputs:	x:	(Numerical)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Descending slope* (Numerical)
	p_2:	*Ascending slope* (Numerical)

7.5 Counting Functions

7.5.1 Count Pulses

The *Count pulses* function has three Boolean inputs and one numerical output. If the *Counting* input has the value "1", then the change from "0" to "1" at the *Pulses* input will be counted and the count value will be set at the output. If the *Reset* input is at "1", then the count value, and thus the output as well, will be "0".

Flexible counting structures can thus be obtained by placing the *Pulse generation* function upstream (see Chap. 7.8.5). The placement of a downstream *Note value* function (see Chap. 7.4.6) makes it possible to also implement a counter with an interim status.

Inputs:	x_1:	*Pulses* (Boolean)
	x_2:	*Counting* (Boolean)
	x_3:	*Reset* (Boolean)
Outputs:	y:	*Count value* (Numerical)
Parameters:	-	

7.5.2 Stop Clock

The *Stop clock* function has two Boolean inputs and one numerical output. If the *Start/Stop* input has the value "1", then the seconds will be counted and the time will be set at the output. If the *Reset* input is at "1", then the time, and thus the output as well, will be "0".
The behavior at the output can be controlled with the *Output* parameter.

Two settings are possible:

Current time
The output value is the current number of seconds counted.

Stopped time
The output value is not the current quantity, but rather the most recently measured quantity. This means that the current count is set to the output whenever the time is stopped with the *Start/Stop* input.

Flexible time measurement structures can thus be obtained by placing the *Pulse generation* function upstream (see Chap. 7.8.5).

Inputs:	x_1:	*Start/Stop* (Boolean)
	x_2:	*Reset* (Boolean)
Outputs:	y:	(Numerical)
Parameters:	p_1:	*Output* (entry list)

7.6 Numerical Conditions

7.6.1 Equals

The *Equals* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if the values are equivalent, otherwise a "0".

With the Precision parameter, you can adjust how precise the comparison is. For this the following explanation:

In view of the fact that numerical values are presented on computers as floating point numbers with finite precision, normal comparisons usually fail. Thus, for example, the finite precision of 2/6 is not necessarily the same as the result of 1/3. The difference is not to be found before the 8th decimal place, but nonetheless the two values are not recognized as being equivalent.

The point at which one needs to break off the comparison of numbers varies from case to case. It is for that reason that you have the option of controlling the precision of the comparison.

If you specify 0.01 for precision, for example, then the numbers 12.453 and 12.458 will still be recognized as equivalents to one another, because the difference is less than 0.01.

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Precision* (Numerical)

7.6.2 Does not Equal

The *Does not equal* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if the values are not equal, otherwise a "0".

For the *Precision* parameter, see the explanation in the *Equals* function (Chapter 7.6.1).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Precision* (Numerical)

7.6.3 Greater than

$>$

The *Greater than* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is greater than x_2, otherwise a "0".

In view of the fact that numerical values are presented on computers as floating point numbers with finite precision, it is difficult to make decisions in border ranges. (See the explanation in Chapter 7.6.1, *Equals* function). This is however usually irrelevant in everyday usage, because the imprecision is not to be found before the 8th significant decimal place.

When however it is important that a precise decision be made for a border range, then you can install the *Rounding* function upstream (see Chap. 7.3.8).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.6.4 Greater than or Equal to

$\geq$

The *Greater than or equal to* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is greater than or equal to x_2, otherwise a "0".

For more on the subject of precision, please note the explanations in the *Greater than* function (see Chapter 7.6.3).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.6.5 Less than

$<$

The *Less than* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is less than x_2, otherwise a "0".

For more on the subject of precision, please note the explanations in the *Greater than* function (see Chapter 7.6.3).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.6.6 Less than or Equal to

The *Less than or equal to* function compares two numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is less than or equal to x_2, otherwise a "0".

For more on the subject of precision, please note the explanations in the *Greater than* function (see Chapter 7.6.3).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.6.7 Within

The *Within* function compares three numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is greater than or equal to x_2, and smaller than or equal to x_3, otherwise a "0".

For more on the subject of precision, please note the explanations in the *Greater than* function (see Chapter 7.6.3).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
	x_3:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.6.8 Outside

The *Outside* function compares three numerical input values for equivalence and outputs a "1" at its Boolean output if value x_1 is smaller than x_2 or greater than x_3, otherwise a "0".

For more on the subject of precision, please note the explanations in the *Greater than* function (see Chapter 7.6.3).

Inputs:	x_1:	(Numerical)
	x_2:	(Numerical)
	x_3:	(Numerical)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7 Boolean Links

7.7.1 Not

The *Not* function supplies the negated Boolean input value at its Boolean output.

I x = "0", then a "1" is output, otherwise a "0".

Inputs:	x:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.2 And

The *And* function links the two Boolean inputs with the "and" operation and supplies the result to its Boolean output.

The output is then "1" only if both inputs are "1", otherwise it is "0".

The following log table makes this function clear.

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Inputs:	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.3 Not - And

The *Not - And* function links the two Boolean inputs with the "nand" operation and supplies the result to its Boolean output.

The output is then "0" only if both inputs are "1", otherwise it is "1".

The following log table makes this function clear.

x_1	x_2	y
0	0	1
0	1	1
1	0	1
1	1	0

Inputs:	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.4 Or

≥1

The *Or* function links the two Boolean inputs with the "or" operation and supplies the result to its Boolean output.

The output is "1" if one of the two inputs is "1". If both are "0", then the output will also be "0".

The following log table makes this function clear.

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

Inputs:	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.5 Not - Or

$\overline{\geq 1}$

The *Not - Or* function links the two Boolean inputs with the "nor" operation and supplies the result to its Boolean output.

The output is "0" if one of the two inputs is "1". If both of the inputs are "0", then the output will be "1".

The following log table makes this function clear.

x_1	x_2	y
0	0	1
0	1	0
1	0	0
1	1	0

Inputs:	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.6 Exclusive Or

=1

The *Exclusive Or* function links the two Boolean inputs with the "xor" operation and supplies the result to its Boolean output.

The output is "1" if precisely one of the two inputs is "1". If both of the inputs are "0" or if both of the inputs are "1", then the output will be "0". One can also say that the output is then precisely "1" if the two inputs are not equivalent.

The following log table makes this function clear.

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Inputs	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.7.7 Not Exclusive Or

$\overline{=1}$

The *Not Exclusive Or* function links the two Boolean inputs with the "nxor" operation and supplies the result to its Boolean output.

The output is "0" if precisely one of the two inputs is "1". If both of the inputs are "0" or if both of the inputs are "1", then the output will be "1". One can also say that the output is then precisely "1" if the two inputs are equivalent.

The following log table makes this function clear.

x_1	x_2	y
0	0	1
0	1	0
1	0	0
1	1	1

Inputs	x_1:	(Boolean)
	x_2:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.8 Other Boolean Operations

7.8.1 Note Value

The *Note value* function is used to hold on to a Boolean value (to freeze it). It has two Boolean inputs.

If the input value x_2 is "1", then the input value x_1 is output at the output. If the input value x_2 is "0", then the last output value remains in effect.

If the value is only to be applied for one flank, then you can put the function *Pulse generation* upstream at x_2 (see Chap. 7.8.6).

Inputs	x_1:	*Value* (Boolean)
	x_2:	*Note* (Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.8.2 Switching Delay

The *Switching delay* function is used to delay Boolean signals. The function has one Boolean input and one Boolean output. A change at the input will not be recognizable at the output until after a parameterizable delay time has elapsed.

The delay times can be set separately for switching on and switching off. . They are adjusted with the parameters *Switch-on delay* and *Switch-off de*lay. This makes it possible to suppress short-term signal changes from view.

Example:
Input and output are "0", the switch-on delay is set to 5 seconds. If the input switches now to "1", then the output waits 5 seconds before switching to "1". If the input switches back to "0" before the 5 seconds have elapsed, then the input remains set to "0" and the change at the input is not visible at all at the output.

Inputs:	x:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Switch-on delay* in seconds (Numerical)
	p_2:	*Switch-off delay* in seconds (Numerical)

7.8.3 T - Flipflop

The *T-FlipFlop* function is the representation of a surge relay. The output switches over every time Boolean input value changes from "0" to "1". (Toggle function, which is the reason for the name T-Flipflop).

A side effect of the T-FlipFlop is that is reduces the frequency of a counting signal down to half its size.
The output is set to 0 after initialization at the time of program start.

Inputs:	x:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.8.4 Mono Flop

The output switches on when the Boolean input value switches from "0" to "1". The function is comparable to the automatic switching of a light in a stairwell.
The output is set to 0 after initialization at the time of program start.
When the Reset input is set to "1", then the output is definitely set to "0".

- *Not retriggerable*
  When the flank switches from "0" to "1", the output for the specified time is switched to "1". After the time elapses, the output is set back to "0". It doesn't matter whether or not this flank occurs again during this time.

- *Retriggerable*
  When the flank switches from "0" to "1", the output for the specified time is switched to "1". After the time elapses, the output is set back to "0". If the flank switches from "0" to "1" once again during this time, then the output will be switched again to "1" for the time specified.

- *Prolongable*
  When the flank switches from "0" to "1", the output for the specified time is switched to "1". If no change of flanks occurred during this time, then the output remains switched to "1" until the flank switches from "1" to "0".

Inputs:	x_0:	S (Boolean)
	x^1:	R (Boolean)
Outputs:	y:	(Boolean)
Parameters:	p1:	Mode
	p2:	Time in seconds

7.8.5 RS - Flipflop

The *RS-FlipFlop* function has a Boolean input *Set* for the purpose of setting the output to "1" and a Boolean input *Reset* for setting the output back to "0".

The *Priority* parameter can be used to define how the output will react when a "1" is present at both inputs simultaneously. The following priorities are possible:

First
If the Set input is the first to switch to "1", then the output is "1"; if the Reset input is the first to switch to "1", then the output is "0". If both switch to "1" simultaneously, then the output remains in the status it was in.

Last
If the Set input is the last to switch to "1", then the output is "1"; if the Reset input is the last to switch to "1", then the output is "0". If both switch to "1" simultaneously, then the output remains in the status it was in.

Off
The output value is "0"

On
The output value is "1"
The output is set to 0 after initialization at the time of program start.

Inputs:	x_0:	*S* (Boolean)
	x_1:	*R* (Boolean)
Outputs:	y:	(Boolean)
Parameters:	p_1:	*Priority* (entry list)

7.8.6 Pulse Generation

The *Pulse generation* function is used for generating a Boolean pulse. Every time the input changes from "0" to "1", the output is set to "1" for a cycle.

This function is useful with the Note functions (Note value, Note minimum, etc.), among others.

Inputs:	x:	(Boolean)
Outputs:	y:	(Boolean)
Parameters:	-	

7.9 Result Values

7.9.1 Numerical Output Value

The *Numerical output value* function makes a numerical input value available to the outside. It publicizes the value.
Output values are displayed on the CMU as well as in the measured value display on a connected PC. Furthermore, the output values are also recorded in the log recording (see Chap. 7.10.3). A maximum of 32 values can be publicized.

The publication of values is accomplished in a fixed decimal point representation. The number of decimal places must be entered using the *Decimal format* parameter for this purpose. The current format permits only 5 significant places, wherby the first one only makes it to 3. This results in the following maximum numerical ranges:

Decimal format	maximum value range
0	-30000 .. +30000
0.0	-3000.0 .. +3000.0
0.00	-300.00 .. +300.00
0.000	-30.000 .. +30.000

The output value is always limited to the above-mentioned maximum value range. The decimal format is to be adjusted in such a way that all of the values relevant to daily practice can be presented. If needed, one can for example publicize a "bar" value in "millibars" by multiplying the value by 1000.

The *Designation* parameter defines the name under which the value is displayed. The *Unit* parameter defines the physical unit. No 2 output values with the same name are permitted to be on hand.

The *Low range* and *High range* parameters are used solely for defining the typical value range for post-processing work. If, for example, the value progression is output as a graphic, then the graphics program will first scale the values to the value range specified here. The specification of the measurement range has no effect on the limitation of the output value.

Example: "0.00" is set as the decimal format, "250.00" as the high measuring range. If the input value now lies at 450.00, then the value 300.00 will be publicized, because the limiting is accomplished with respect to the maximum value range and not to the high measuring range.
If this behavior is not wanted, limit the value using the *Limit* function (see Chap. 7.4.3).

Inputs:	x:	(Numerical)
Outputs:	-	
Parameters:	p_1:	*Designation* (character string)
	p_2:	*Low range* (Numerical)
	p_3:	*High range* (Numerical)
	p_4:	*Unit* : (character string)
	p_5:	*Decimal format* : (input list)

7.9.2 Boolean Output Value

The *Boolean output value* function makes a Boolean input value available to the outside. It publicizes this value.

The status values are displayed on the CMU. A connected PC indicates the status values in the status line. Furthermore, the status values are also recorded in the log recording (see Chap. 7.10.3). A maximum of 15 status values can be publicized.

All status values are combined to make a common status in the results log. Each individual status value is represented thereby as one Bit.
The question of which Bit will be used is defined in the *Bit number* parameter.

There must never be two status values with the same bit number.

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p_1:	*Bit number* (whole number)

7.10 Actions

7.10.1 Setting Switching Output

The *Set switch output* function transmits the Boolean input to a digital switching output on the CM device.

The *Output terminal* parameter is used to define which digital output port is used.

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p_1:	*Output terminal* (input list)

7.10.2 Setting Analog Outputs

The *Set analog output* function outputs the numerical input to an analog output of the CM device.

The *Output* parameter is used to define which analog output will be used. The scaling is defined by the parameters *Lower value* and *Upper value*.

If for example -25 is set for the lower value and 150 for the upper value, then all input values less than or equal to -25 will cause the lower signal range to be output at the output (type 0 V or 4 mA) and all input values greater than or equal to 150 will lead to an output signal that corresponds to the upper signal range. (Type 10 V or 20 mA)

Inputs:	x:	(Numerical)
Outputs:	-	
Parameters:	p_1:	*Output* (input list)
	p_2:	*Low range* (Numerical)
	p_3:	*High range* (Numerical)

7.10.3 Display Message

Disp

The *Display message* function has one Boolean input. A message will be displayed on the CMU monitor for as long as the input value is "1". The upper line includes the date and time of day that the message appeared.

The text of the message is defined with the *Message* parameter. This is to be found in the lower line of the display.

The *Acknowledgement* parameter can be used to specify that the message will not automatically disappear with the display when the input returns to "0", but rather that it must be acknowledged by pressing a key. Should this eventuality occurs, then it will be signalled by the flashing of the upper line.

Independent of that which is set with Acknowledgement, a message can be acknowledged at any time in order that the CMU can be operated. If the input remains at "1" however, then the message will be displayed again after a time.

If several messages are active simultaneously, then the keys can be used to scroll through the list of messages.

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p_1:	*Message* (character string)
	p_2:	*Acknowledgement* (Boolean)

7.10.4 Switch on LED

The *Switch on LED* function has one Boolean input. The respective LED will remain on for as long as the input value is "1".

The *Acknowledgement* parameter can be used to specify that the LED will not automatically go out when the input returns to "0", but rather that the LED must be acknowledged by pressing a key. The respective LED will start to flash in such cases.

The *Color* parameter is used to set which LED will be switched on. 3 traffic light colors are present: • red, • yellow • green.

Each color may only be used once in the CM program.

Depending on the specific CM device, the LEDs can also have designations other than the names of their colors, e.g.: LED1, LED2, etc.

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p_1:	*Color* (entry list)
	p_2:	*Acknowledgement* (Boolean)

7.10.5 Compiling a Log Entry

The *Compile log entry* function is used for saving the currently publicized values in the ongoing log recording (see Chap. 7.9.1. Numerical output value and 7.9.2 Boolean output value).

The log entry is generated every time the input value changes from "0" to "1".

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	-	

7.10.6 Compiling Quick Log Entries

The *Quick log entries* function is used for rapidly saving the currently publicized values in the ongoing log recording (see Chap. 7.9.1. Numerical output value and 7.9.2 Boolean output value).
Log entries will be generated at the specified time interval (p2) for as long as the input is set to "1".
The *Interim period* parameter specifies that for every time stamp of a log entry, the precise time value of the log entry will also be saved in ms.

The log entry is generated every time the input value changes from "0" to "1".

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p1:	Interim period (Boolean)
	p2:	Time in seconds

7.10.7 Start new log

Using the *Start new log* function, a new time-dependent or cycle-dependent log file can be generated. The numbering in the file name is increased by 1.
The previous log file is stored in the internal memory.

Eingänge:	x:	(Boolesch)
Ausgänge:	-	
Parameter:	-	

7.10.8 **Transition** (see Chap. 7.2.12)

7.10.9 Send SMS

The *Send SMS* function is used to send an SMS when an event occurs. This occurs with every switchover of the input value changes from "0" to "1".

The text of the SMS is defined with the *Message* parameter and the *Telephone number* parameter defines the number to which the SMS is sent.

Inputs:	x:	(Boolean)
Outputs:	-	
Parameters:	p_1:	*Telephone number* (character string)
	p_2:	*Message* (character string)

Caution

The "***Send SMS***" function requires that the following conditions be met:

- One HYDAC GSM Module CSI-F-10 is connected to the HSI Master of the CMU 1000 in accordance with specifications.
- The GSM module CSI-F-10 is supplied with voltage and ready for functioning.
- A valid, functionable SIM card has been inserted in the GSM module CSI-F-10.
- The GSM module has a sufficient network reception strength.

7.11 Other

7.11.1 Comment

With the *Comment* function, it is possible to insert a comment at any given point in the CM program.

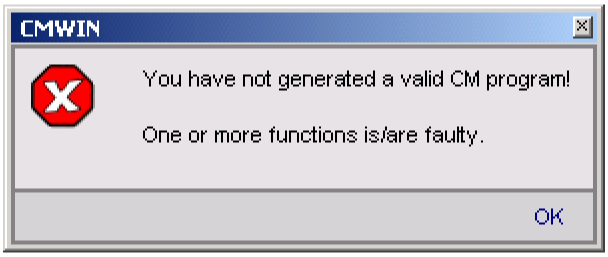
The heading of the Comment box is defined with the *Inscription* parameter.

The actual comment text is entered with the *Comment* parameter.

Inputs:	-	
Outputs:	-	
Parameters:	p_1:	*Inscription* (character string)
	p_2:	*Comment* (character string)

8 Error Messages CM Program Compilation

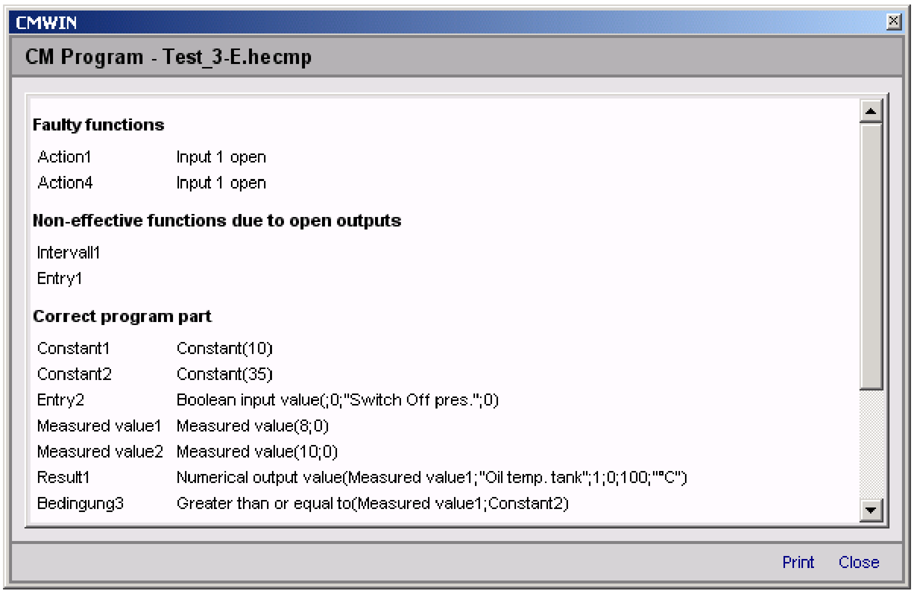
In order to ensure as high a degree of operational safety as possible, the CM program generated will be checked for possible programming errors before it is transferred into the device. If the system recognizes one or more such programming errors, then the following message will appear and the CM program will not be transferred into the CMU 1000.



You have not generated a valid CM program!

One or more functions is/are faulty.

The menu function [*CM Program / Display*] (see Chap. 6.1.2) can be used in such cases to have a more detailed program evaluation displayed, together with error messages, and to have this printed out as needed.



CM Program - Test_3-E.hecmp

Faulty functions

Action1	Input 1 open
Action4	Input 1 open

Non-effective functions due to open outputs

Intervall1
Entry1

Correct program part

Constant1	Constant(10)
Constant2	Constant(35)
Entry2	Boolean input value(;0;"Switch Off pres.";0)
Measured value1	Measured value(8;0)
Measured value2	Measured value(10;0)
Result1	Numerical output value(Measured value1;"Oil temp. tank";1;0;100;"°C")
Bedingung3	Greater than or equal to(Measured value1;Constant2)

Work through all of the error messages in your CM program and then transfer the program once again into your CMU 1000.

The following list shows all of the error messages that could occur during programming, together with the function groups present in the CMWIN to which they belong and their causes.

8.1 Overriding Error Messages

8.1.1 Function not Available in this Mode

The CM program was compiled for a platform in which the marked function does not exist.
► Check the platform setting and correct it or modify the CD program accordingly.

8.2 Error Messages with Data Sources

8.2.1 Invalid Channel Setting

A channel/subchannel was selected that is not valid.
► Check the channel setting and correct it.

8.2.2 Duplicate Channel Name

The name of a channel may only be used once in a CM program.
► Check the channel name and correct it.

8.2.3 Invalid Digital Input

A port must be set with the Digital input function.
► Check the port setting and correct it.

8.2.4 Duplicate Digital Input

A port must be set with the Digital input function. This port that is set may only be used once in a CM program.
► Check the port setting and correct it.

8.2.5 Too many Boolean Input Fields

No more than a maximum of 50 Boolean input values are permitted to be present in a single CM program.
► Reduce the Boolean input values to a maximum of 50.

8.2.6 No Inscription with Boolean Input

The Boolean input value must have an inscription.
► Enter an inscription in the function properties.

8.2.7 Duplicate Inscription with Boolean Inputs

The inscription of a Boolean input value must be unambiguous within a CM program and is only permitted to occur once for that reason.
► Check the inscription and correct it.

8.2.8 Too Many Numerical Input Values

No more than a maximum of 50 numerical input values are permitted to be present in a single CM program.
► Reduce the numerical input values to a maximum of 50.

8.2.9 No Inscription with Numerical Input

The numerical input value must have an inscription.
► Enter an inscription in the function properties.

8.2.10 Duplicate Inscription with Numerical Input

The inscription of a numerical input value must be unambiguous within a CM program and is only permitted to occur once for that reason.
► Check the inscription and correct it.

8.2.11 Duplicate Error Source

A setting is made at an error source as to which error the output of the error sources is set. No error source may be present more than once in a CM program.
► Check the setting under "Error messages" and correct it.

8.3 Error Messages with Operations/Conditions

8.3.1 Upper and Lower Measured Value Limits too Close to one another

In cases of functions with upper and lower measured value limits, the difference between the two values must amount to at least 10 steps away from one another.
► Check the values that were entered and correct them.

8.3.2 Measured Value Limits Outside the Range of -30000 to 30000

In cases of functions with upper and lower measured value limits, the entered values must be betweem -30,000 and +30,000.
► Check the values that were entered and correct them.

8.3.3 Lower Measured Value Limit Greater than Upper Measurement Value Limit

In cases of functions with upper and lower measured value limits, the lower measured value must be less than the upper measured value.
► Check the values that were entered and correct them.

8.4 Error Messages with Result Values/Actions

8.4.1 Invalid Output LED Selected

The selected LED does not exist in this device and must be set correctly.
► Check the LED setting and correct it.

8.4.2 Duplicate Usage of Output LED

The selected LED is already being used in the current CM program and is not permitted to be present twice.
► Check the LED selection and correct it.

8.4.3 Invalid Digital Output

The quantity of digital outputs is device-dependent. This error is set when an output terminal is selected that a device doesn't have.
► Check the selection and correct it.

8.4.4 Duplicate Digital Ouput

The output terminal of the digital output may not be present more than once in a CM program.
► Check the adjusted output terminal and correct it.

8.4.5 Invalid Analog Output

The quantity of analog outputs is device-dependent. This error is set when an output is selected that a device doesn't have.
► Check the selection and correct it.

8.4.6 Duplicate Analog Output

The set output of the analog output may not be present more than once in a CM program.
► Check the output settings and correct them.

8.4.7 Too Many Boolean Output Fields

The number of Boolean output fields in one CM program is device-dependent.
► Reduce the Boolean output fields to the device-specific maximum.

8.4.8 Duplicate Boolean Output Field

The inscription of a Boolean output field may not be present more than once in a CM program.
► Check the inscription and correct it.

8.4.9 The Bit Number Must Be a Figure between 0 and 14

The property "Bit number" may not lie outside the range of 0 ... 14 for the Boolean output field function.
► Check the value that was entered and correct it.

8.4.10 Too Many Numerical Output Fields

The number of numerical output fields in one CM program is device-dependent.
► Reduce the numerical output fields to the device-specific maximum.

8.4.11 Duplicate Numerical Output Field

The inscription of a numerical output field may not be present more than once in a CM program.
► Check the inscription and correct it.

8.4.12 Message and Telephone Number too Long

The length of the message + telephone number together is limited to 230 characters.
► Check the respective inputs and correct them.

9 Technical Data

9.1 Power Supply

- Input voltage: 18.0 ... 35.0 V DC
- Power consumption: max. 1.5 A (3.5 A with connected CSI-F-10)
- Reverse voltage protection: -30 V
- Dielectric strength: +40 V

9.2 Connecting Sensors

- Up to 8 sensors with HSI functionality or
  up to 8 SMART sensors and in addition up to 8 analog sensors, as well as
  up to 4 digital sensors
  (4 x digital / 2 x digital + 2 x frequency / 3 x digital + 1 x frequency)
  can be connected at the same time

  The measuring ranges of the analog inputs are designed for a variety of requirements

9.3 Analog Inputs

Channels I and J (precision)

4 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 4,5 V	(≤ ± 0,1 % FS max.)
0,5 .. 4,5 V	(≤ ± 0,1 % FS max.)
0 .. 5 V	(≤ ± 0,2 % FS max.)
1 .. 5 V	(≤ ± 0,2 % FS max.)
0,5 .. 5,5 V	(≤ ± 0,2 % FS max.)
1 .. 6 V	(≤ ± 0,2 % FS max.)
0 .. 10 V	(≤ ± 0,1 % FS max.)

Channels K and L (precision)

4 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 4,5 V	(≤ ± 0,5 % FS max.)
0,5 .. 4,5 V	(≤ ± 0,5 % FS max.)
0 .. 5 V	(≤ ± 0,5 % FS max.)
1 .. 5 V	(≤ ± 0,5 % FS max.)
0,5 .. 5,5 V	(≤ ± 0,5 % FS max.)
1 .. 6 V	(≤ ± 0,5 % FS max.)
0 .. 10 V	(≤ ± 0,25 % FS max.)
0 .. 50 V	(≤ ± 0,1 % FS max.)
-10 .. +10 V	(≤ ± 0,2 % FS max.) only L!

Channels M and N (precision)

4 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 4,5 V	(≤ ± 0,1 % FS max.)
0,5 .. 4,5 V	(≤ ± 0,1 % FS max.)

Channels O and P (precision)

4 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 20 mA	(≤ ± 0,1 % FS max.)
0 .. 4,5 V	(≤ ± 0,1 % FS max.)
0,5 .. 4,5 V	(≤ ± 0,1 % FS max.)
-10 .. +10 V	(≤ ± 0,2 % FS max.) only P!

9.4 Digital Inputs

- Quantity: 4, of which 2 are for frequency measurements (Channels Q and R)
- Trigger threshold: approx. 2 V
- Dynamics: 30 kHz

9.5 Measurement Channels

A total of up to 32 measurement channels can be processed by the CMU.
One measurement channel can be a value of a connected sensor (also a subchannel or a SMART sensor) or a value derived (calculated) from sensor data.

9.6 Analog Outputs

Quantity:	2
Type:	individually selectable, current (4 .. 20 mA) or voltage (0 .. 10 V)

9.7 Digital Outputs

Quantity:	4
Type:	Relay output, directional contact
Switching power:	30 V DC / 1 A

9.8 Calculation Unit

Analog value acquisition:	12 Bit A/D transmitter
Internal measured value memory:	
- RAM:	1 MByte
- Flash Memory	256 MByte

Real-time clock, battery-buffered (battery change only by HYDAC SERVICE GMBH)

The memory is full when:

- The maximum amount of log files, (independent of size), equals 1000. Each file can have a maximum size of 1 Mbyte
- The log file storage = 256 MByte

9.9 Interfaces

9.9.1 Keyboard

- 4 arrow keys (up, down, right, left)
- OK key
- ESC key

9.9.2 View

Two-line LED display (2x16 characters) with background illumination. Status information can also be displayed via 3 different colored LEDs.

9.9.3 USB Mass Storage Device

USB 1.1 / USB 2.0 full speed interface for connection of a Mass Storage Device (Memory Stick); connection socket Type "A".
The data recorded by the CMU can be transferred to a memory stick via this interface. The USB host supports exclusively Mass Storage Devices.

9.9.4 Ethernet

The calculation unit is equipped with an RJ 45 8/8 Ethernet interface which supports the following services/protocols:
- HTTP Server
- OPC Client

9.9.5 Serial Interface 0 (UART 0)

The serial interface 0 of the calculation unit is used either for implementation of an RS 232 or of an HSI Master interface (see below). The switchover can be programmed freely (optional IO link also possible). The connection is accomplished using plug-in terminals. No handshake lines.

9.9.6 HSI Master

HSI interface for cascading the CMU. The HSI Master interface is linked with an HSI sensor interface of the superordinate SMU for this purpose.

9.9.7 USB Device

USB 1.1 / USB 2.0 full speed interface for connecting a PC/laptop for configuration of the CMU; connection socket Type "B".

9.10 Cycle Time

The CMU determines automatically the required cycle time when the program starts. The user has the option of having the current cycle time displayed on the CM Editor. The display shows rounded values (e.g. 10 ms instead of the 8.7 ms detected).

9.11 Operating and Ambient Conditions

Operating temperature:	-20.. +70°C
Storage temperature:	-30.. +80°C
Relative moisture:	0 .. 70%, non-condensing

9.12 Dimensions and Weight:

Dimensions:	approx. 212 x 105 x 32 mm (W x H x D)
Weight:	approx. 600 g

9.13 Technical Standards

EMV:	EN 61000-6-1/2/3/4
Safety:	EN 61010
Protection type:	IP 40

9.14 Scope of Delivery

The CMU 1000 is packed in a box and delivered in ready-for-operation condition. Check the packing and the device prior to installation for transport damage and the contents of the packing for completeness.

- CMU 1000
- User manual
- CD-ROM with the PC software "CMWIN" as well as additional product information
- USB connection cable

9.15 Maintenance and cleaning

- Switch the CMU 1000 voltage-free and check it for absence of voltage.
- For reasons of electrical safety, never clean the device with water or other fluids, and under no circumstances whatsoever should the device be immersed in water or other fluids.
- Use only a dry, lint-free cloth for cleaning. Do not use any solvents, gasoline or similar, because these would cause damage to the CMU 1000.

9.16 Recycling and Disposal

The packing and the packing material are comprised solely of environmentally friendly materials. They can be disposed of in the respective local recycling containers.

Never dispose of electrical devices and electronic components in containers intended for household refuse!
Pursuant to European Guideline 2002/96/EU concerning used electrical and electronic devices and implementation in national law, used electrical devices must be collected separately and transferred to an environmentally suitable recycling process.
Consult for this purpose a company that is certified for the disposal of electronic waste in order to ensure that your device is recycled or disposed of in an environmentally friendly manner.

10 Ordering Details

CMU 1000 - 000 - X

Operator guidance and documentation
D = German
E = English
F = French

11 Accessories

- **SMART Sensors**

HLB 1300 - Series	(oil condition sensor)
AS 1000 - Series	(moisture sensor)
CS 1000 - Series	(dirt sensor)

- **HSI pressure measuring transmitter of the measuring ranges:**

- 1 ... 9 bar, 0 ... 16 bar, 0 ... 100 bar, 0 ... 250 bar, 0 ... 400 bar, 0 ... 600 bar

Mat. No.	909429	Mat. Desig.	HDA 4748-H-0009-000 (-1 ... 9 bar)
Mat. No.	909425	Mat. Desig.	HDA 4748-H-0016-000 (0 ... 16 bar)
Mat. No.	909554	Mat. Desig.	HDA 4748-H-0060-000 (0 ... 60 bar)
Mat. No.	909426	Mat. Desig.	HDA 4748-H-0100-000 (0 ... 100 bar)
Mat. No.	909337	Mat. Desig.	HDA 4748-H-0250-000 (0 ... 250 bar)
Mat. No.	909427	Mat. Desig.	HDA 4748-H-0400-000 (0 ... 400 bar)
Mat. No.	909428	Mat. Desig.	HDA 4748-H-0600-000 (0 ... 600 bar)

- **HSI temperature measuring transmitter**

Mat. No.	909298	Mat. Desig.	ETS 4548-H-000 (-25 to +100°C)

- **Level sensors**

ENS 3000 series

- **HSI volume flow measuring transmitter**

Mat.-No.	909405	Mat.-Desig.	EVS 3108-H-0020-000 (1,2 - 020 l/min)
Mat.-No.	909293	Mat.-Desig.	EVS 3108-H-0060-000 (006 - 060 l/min)
Mat.-No.	909404	Mat.-Desig.	EVS 3108-H-0300-000 (015 - 300 l/min)
Mat.-No.	909403	Mat.-Desig.	EVS 3108-H-0600-000 (040 - 600 l/min)
Mat.-No.	909409	Mat.-Desig.	EVS 3118-H-0020-000 (1,2 - 020 l/min)
Mat.-No.	909406	Mat.-Desig.	EVS 3118-H-0060-000 (006 - 060 l/min)
Mat.-No.	909408	Mat.-Desig.	EVS 3118-H-0300-000 (015 - 300 l/min)
Mat.-No.	909407	Mat.-Desig.	EVS 3118-H-0600-000 (040 - 600 l/min)

- **RPM probe**

Mat. No.	909436	Mat. Desig.	HDS 1000-002 (plug M12x1)
Mat. No.	904812	Mat. Desig.	HDS 1000 reflection foil set (25 pcs.)

- **Sensor simulator for 2 HSI measuring transmitters**

Mat. No.	909414	Mat. Desig.	SSH 1000-H-3 (Simulator for HMG 3000) ideal for learning purposes

HYDAC ELECTRONIC GMBH

Hauptstraße 27
D-66128 Saarbrücken
Germany

Web : www.hydac.com
E-mail : electronic@hydac.com
Tel.: +49-(0)6897-509-01
Fax: +49-(0)6897-509-1726

HYDAC SERVICE

If you have any questions concerning repairwork, please don't hesitate to contact HYDAC SERVICE:

HYDAC SERVICE GMBH

Hauptstr. 27
D-66128 Saarbrücken
Germany

Tel.: +49-(0)6897-509-1936
Fax: +49-(0)6897-509-1933

Notice

The information and particulars provided in this manual apply to the operating conditions and applications described herein. In the event of deviating applications and/or operating conditions, please contact the respective HYDAC department concerned.

If you have any questions, suggestions, or encounter any problems of a technical nature, please contact your HYDAC representative.

All technical details are subject to change without notice.