UVR 1611

Version A4.03 EN

Manual version 2

Freely programmable universal controller

Operation
Programming
Installation instructions

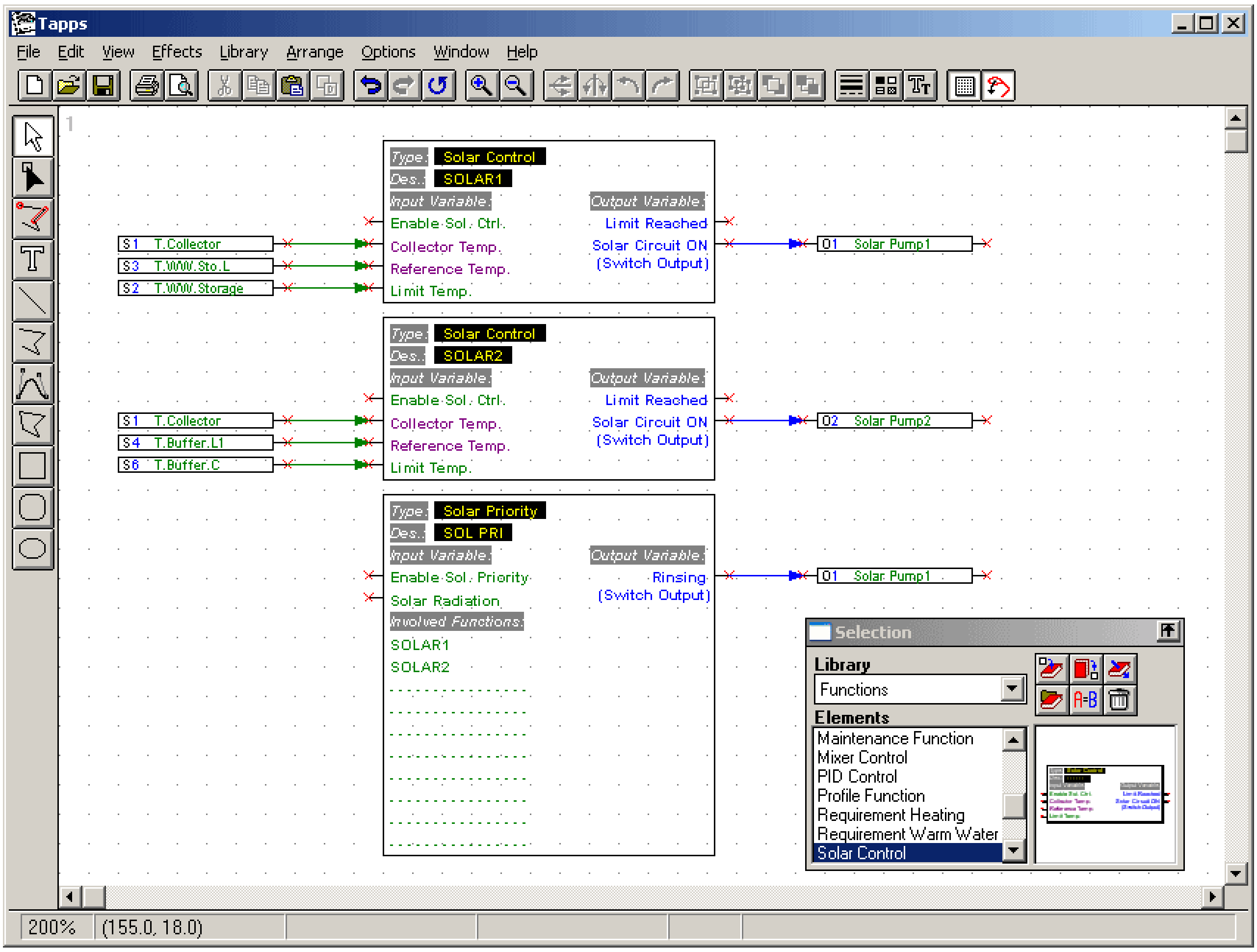
en

TECHNISCHE
ALTERNATIVE

NOTICE

This manual was written in order to provide the specialist with both an overview of the various control possibilities of the unit and its corresponding basic standards. It serves especially as a programming tool while operating the unit. Even using the **TAPPS** – system (the „**T**echnische **A**lternative **P**lanning and **P**rogramming **S**ystem") which you can find on our homepage **www.ta.co.at**, it is sometimes to know the programming mechanisms of the unit in order to be able to make local changes remotely.
In principle however, **TAPPS** is still recommended. This enables the specialist to design (= program) and parameterize the entire functionality in a graphical flow chart. In order to load the data into the control unit, the C.M.I. or the CAN-TOUCH is required.

Example with TAPPS:



The present manual describes exclusively the direct programming of the control unit and does not refer to TAPPS.

Contents

Safety requirements

These instructions are intended exclusively for authorised contractors.

All installation and wiring work on the controller must only be carried out in a zero volt state.

The opening, connection and commissioning of the device may only be carried out by competent personnel. While doing so, they must observe all local safety requirements.

This device is state of the art and meets all necessary safety regulations. It may only be used in accordance with the technical data and the safety requirements and regulations listed below. When using the device, also observe the statutory and safety regulations apposite to the particular use.Any other use will automatically void all warranty rights.

► The device must only be installed in a **dry** interior room.

► It must be possible to isolate the controller from the mains using an omnipolar isolating facility (plug/socket or 2-pole isolator).

► Before starting installation or wiring work, the controller must be completely isolated from the mains and protected against reconnection. Never interchange the safety low voltage connections (e.g. sensor connections) with the 230 V connections. Destruction and life threatening voltages at the device and the connected sensors may occur.

► Solar thermal systems can become very hot. Consequently there is a risk of burns. Proceed with caution when fitting temperature sensors.

► For safety reasons, the outputs should only be left in manual mode when testing. In this operating mode, no maximum temperatures or sensor functions are monitored.

► Safe operation is no longer possible if the controller or connected equipment exhibits visual damage, no longer functions or has been stored for lengthy periods in unsuitable conditions. If this is the case, disable the controller and equipment and secure against unintentional use.

Maintenance

If treated and used correctly, the device will not require any maintenance. Use a cloth moistened with mild alcohol (such as methylated spirits) to clean. Never use corrosive cleaning agents or solvents such as chloroethylene or trichloroethylene.

No components relevant to long term accuracy are subject to loading if the device is used correctly. Consequently long term drift is extremely low. The device therefore cannot be calibrated. Thus applying any compensation is impossible.

The design characteristics of the device must not be changed during repairs. Spare parts must correspond to the original spare parts and must be used in accordance with the build version.

Function mode

This unit is a very compact control system that has many applications for solar warm water and heating systems and for pumps and valves used in such systems.

The 16 sensor signals pass through overvoltage protection, a low-pass filter, and the multiplexer before they reach the processor's A/D transformer. An adjustable reference is used to calculate the value of the measurement signal. In addition, the computer periodically checks all of the operator's controls, describes the display, and handles the CAN bus.

Once the temperatures have been calculated and links set, the power driver switches the respective outputs. To prevent a loss of data, the device has nonvolatile memory (EEPROM) and a super capacitor (for around three days) for reserve power.

Planning basics

To ensure efficient programming, the following order has to be observed:

1 The basic condition of writing the desired controller functions and its parameterizing is an **accurate hydraulic diagram**!

2 This diagram **must** show **what** should be regulated and **how**.

3 The **sensor positions** are to be defined according to the desired controller functions and drawn into the diagram.

4 The next step consists in providing all sensors and “consumers” with the desired **input or output numbers**. As all the sensor inputs and outputs have different characteristics, it is not possible to provide consecutive numbers. Therefore, the input and output assignment must be made according to the following description:

Inputs:

All 16 inputs are suitable for standard sensors of the types KTY (2 kΩ) and PT1000 or for digital inputs. In addition, the following inputs have special functions:

S8: Current loop (4 – 20 mA) or control voltage (0 - 10V=)

S15, S16: Pulse input, such as for volume flow encoder

Signal voltages of **more than** 5 V at the inputs S1 to S7 and S9 to S16 or **more than** 10 V at S8 are **not allowed**.

Outputs (supply voltage side):

A1: Speed-adjustable output **(!!!!!!!!!! max. 0.7A !!!!!!!!!!)** with integrated interference filter

A2, 6, 7: Speed-adjustable outputs for pumps (max. 1A)

A3: Relay output (closer) for undefined consumers

A4: Relay output with opener and closer for undefined consumers, preferably for valves without retracting spring. A4 combined with A3 is also suitable for mixer motors.

A5: Relay output – **potential-free** with opener and closer for burner demand with the legally prescribed distance to the supply voltage.

A8, A9: Relay outputs (closer) for undefined consumers, preferably together with mixer engines, as only one commonly used neutral conductor clamp is available for both outputs

A10, A11: Relay outputs (A10 with closer, A11 with opener and closer) for undefined consumers, preferably together with mixer engines, as only one commonly used neutral conductor clamp is available for both outputs

Outputs (protective low voltage side):

Hirel 1, 2: Control lines for a relay module for two other relay outputs **A12** and **A13**, which can be installed as module in “Slot 1”.

DL (14): DL bus as bus link for various sensors and/or for data recording via data logger to a PC. This connection is not only able to be used through the parameterization but also to activate an additional relay.

0-10 V / PWM (A15, A16): Control output with a standardized voltage level of 0 to 10 V e.g. for boiler modulation. Switchable to PWM (level approx. 10 V, cycle duration 0.5 ms). Referred to as analogue output in the user software.

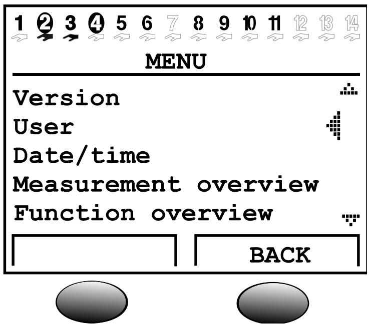
5 Now the calling of the functions and its parameterizing is effected.

Basic standards

Basic operation

Display

The display consists of four information fields



The top line constantly provides information about the actual output states.

Blank field instead of number 5 = output five has not yet been parameterized

5 Output five is active, runs in automatic mode and is temporarily **switched off**

❺ Output five is active, runs in automatic mode and is temporarily **switched on**

5➔ Output five is active, runs in **manual mode** and is temporarily switched off

❺➔ Output five is active, runs in **manual mode** and is switched on at the moment

The second line is the headline for the following menu and/or parameter lines.

The middle display area is the operative range. Within this range the programming, parameterizing and indicating takes place.

The lowest line exclusively serves to mark the two keys below in order to be able to assign different functions to it.

Keys

The control unit has two keys below the display. They are constantly assigned with the required functions via the display.

x10 – The changeable value changes for 10 steps each per increment of the scroll wheel.

BROWSE – This function allows the direct “switch" from one menu level to the same level of the next menu by means of the scroll wheel

MENU – To switch from the opening image (after starting-up) to the menu

SERVICE – To switch from the function overview (the most important menu for the user) into all other menus

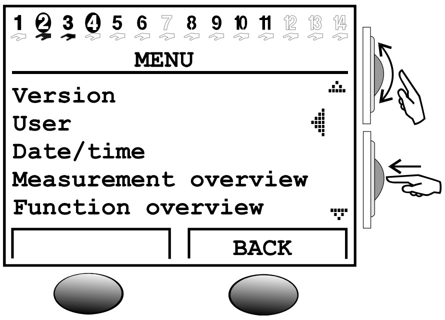
BACK – The PC switches immediately into the next-higher menu level

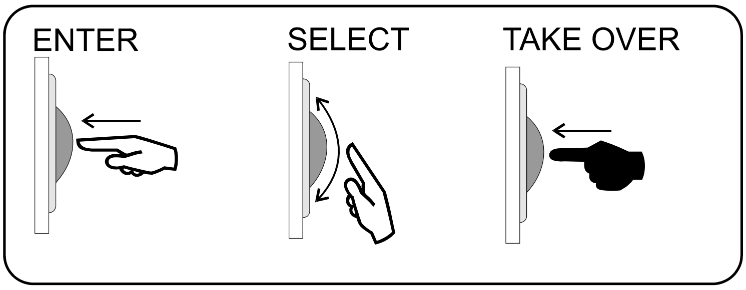
CANCEL - The current entry or change of a value is stopped

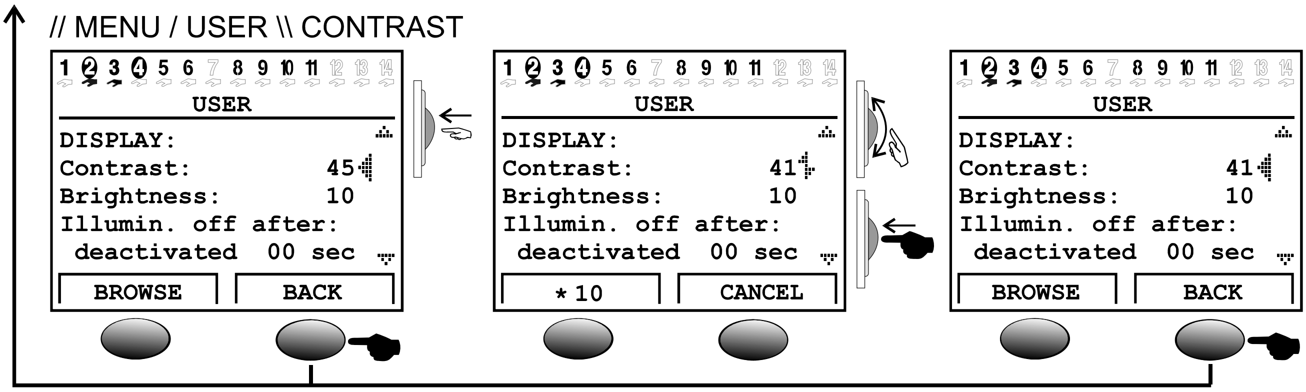
Scroll-wheel

By means of the scroll-wheel, the selected menu can be gone through by the right pointer in the display. Small upward or downward showing arrows symbolize the possibility of further menu lines above or below the visible display range.

If a parameter is to be changed, the pointer must be put in the desired position. Press the wheel to change the background lighting of the scroll wheel frame to orange to indicate programming. Now the value can be adjusted using the wheel (possibly also with "* 10"). You may cancel at any time by pressing CANCEL. Press the wheel again to turn the screen light green and take over the parameter.







Terms used

Operating system = The software (operating system) of the control unit (e.g.: version A3.28EN) with indicator of the user language

C.M.I. = Control and monitoring interface, web server for data transfer between CAN bus network and LAN network

Bootloader = Accessory equipment for data transfer between control unit and PC (no longer available)

Boot sector = Protected storage area in the processor containing a basic menu for “auto-programming" of the chip (e.g.: B2.00)

CAN-Bus = Data bus for data exchange within the unit family

Function data = Customized programming and parameterizing

Function module / **Function** / **Module** = Available functions (e.g.: solar thermal control), which constitute the control characteristics.

Infrared interface – CAN bus on infrared basis (below the two keys) allowing a slack connection to the Bootloader

Measuring data = Measured values, output states, results of computation such as kW and others

User interface

After starting up, the display indicates this menu.

```
   TECHN. ALTERNATIVE
----------------------
Homepage: www.ta.co.at
----------------------
          UVR1611
Operat. sys: Ax.xxEN
```

Operating system: Version number of the software. The latest software (higher number) is available for download under http://www.ta.co.at. It can be transferred to the control unit by means of accessory equipment – the C.M.I..

The key ***MENU*** offers an entrance to the unit menu:

```
       MENU
---------------------
Version
User
Date/time
Measurement overview
Function overview
```

and by scrolling downwards:

```
Inputs
Outputs
Functions
Messages
Network
Data administration
```

Version – shows only the same indication as after the starting up - i.e. the operating system of the unit.

User – This menu permits the adjustment of the control level, the indication contrast and the background lighting as well as the entrance into a so-called “User interface editor”, which allows the creation of an own menu surface.

Date / Time – to set the date and time. It is also possible to switch between normal time and summer time.

Measurement overview – to display all measured values and network inputs in a table.

Function overview – All important information and parameters (e.g.: ambient temperature) of the determined function modules are written by the programmer (specialist) in an editor (“User interface editor”) and displayed here clearly. The computer switches automatically to this overview after a few minutes, since it is the most important control panel for the user.

Inputs - This menu offers an exact overview of all input values. Furthermore, parameters for all inputs are set here. For details, see chapter “Menu Inputs”.

Outputs - for the complete parameterizing and manual operation of all outputs. For details, see chapter “Menu Outputs”.

Functions - This is the menu where all function modules of use are listed. Also, the control tasks and all corresponding parameters are specified here.

Messages – Events determined by the programmer via this menu can trigger status and error messages as well as an alarm tone.

Network – In this menu, all settings (node number, network in- and outputs, …) concerning the integration of the control unit in a CAN open bus network are defined.

Data administration – This menu contains for the specialist all commands for the data administration and protection as well as for an update of the operating system.

MENU User

Here the following entries are listed:

```
       USER
----------------------
OPERATING MODE:
User
Technician
Expert                ✓
```

and by scrolling downwards:

```
DISPLAY:
Contrast:        41
Brightness:      10
Illumin. off after
 deactivated   00 sec.
autom. changeover to
func. overview:yes

DATE / TIME:
aut. changeover std/
summertime: yes

Time since leaving
the expert level:
              0 Days

USER INTERF.EDITOR

USER BLOCK:
Parameter:        yes
Outputs:          yes
MENU:              no

SIMULATION:       no

CHANGE EXPERT
CODE TO:     0 0 0 0
```

Only visible for experts

User - All indication possibilities, but only the most important settings are permitted.

Technician – In addition, all the settings are permitted. Access only possible via a key number. This number can be detected by solving a "little riddle” hidden in the manual.

Expert – In addition, the programming of all functions is possible. The necessary key number is only passed on to trained personnel by email or by telephone.

***DISPLAY:* Contrast** – Adaptation of the display contrast to the lighting conditions.

DISPLAY: **Brightness** – The display has a background lighting, which is integrated in the circuit so that it does not need additional energy. The attenuation of the 12V relays to the 5V computer tension in many devices is transformed into heat, but in case of the UVR1611 also into light! Thus disconnecting does not save energy. The intensity of the background lighting is variable and can be switched off after an adjustable time, while no control element is used.

DISPLAY: **Automatic changeover to Function overview**– In the user surface the most important information for the user is listed in an overview of functions. This command can be used to activate an automatic switch when no control element has been used for several minutes.

DATE / TIME: **Automatic changeover standard / summer time** - This command allows the automatic switching between summer and standard time.

Time since leaving the expert level: **-** A negligent passing on of the expert key number often leads to changing of important parameters and linkages by unauthorized persons. This function allows you to check.

USER INTERFACE EDITOR: Opens an editor menu to program the dialogue (the overview of functions) between control unit and user (only for "expert").

USER BLOCK: **Parameter** – If set to yes, the user is not allowed to change any of the parameters (exception: function overview, all parameters in the user menu and outputs (MANUAL / AUTO)).

USER BLOCK: **Outputs** – If set to yes, if set to yes, the output conditions can also no longer be changed by the user.

USER BLOCK: **MENU –** if set to yes, user and technician now only have access to the function overview and the user menu (switching using the left key). After logging in as an expert, it is possible, to go to the main menu from the function overview by pressing the "SERVICE" key.

SIMULATION: option to activate the simulation mode (in expert mode only):

- no average determination of the outside temperature in heating circuit control
- inputs defined as PT1000 sensors are measured as KTY
- no evaluation of a room sensor

The simulation mode is ended automatically when closing the expert level.

CHANGE EXPERT CODE TO: **-** Change the ex-works key number. Without knowledge of this number no read-out of the program (function data) is possible later on.
Under normal conditions, the control unit automatically returns to the user mode two hours after the last key actuation. Since this is unwanted in devices used for programming or test purposes, the key number 0 0 0 0 blocks the resetting.
WARNING: The loss of the selected key number can only be cancelled by resetting to the factory setting – under complete loss of the function data.

MENU Date / time

Here the following entries are listed:

```
       DATE / TIME
----------------------
        Thursday
      16. 12. 2010

   Std time: 00 : 00
```

All values can be selected and changed accordingly by means of the scroll-wheel. The date and time function has a power reverse of approximately three days in case of a blackout. The indication of the “Std time“ corresponds to the winter time. The changeover to the summer time is possible manually or automatically (see user menu).

MENU Measurement overview

In this menu, all entries of the measured values are listed in a table:

```
  MEASUREMENT OVERVIEW
----------------------
 1:  60.3 °C  27.6 °C
 3:  49.2 °C  88.4 °C
 5:  29.0 °C  47.5 °C
                 …
                 …
                 …
NETWORK INPUTS:
1:OFF         ON
17: 25.4 °C   10.6 °C
```

In other words, the temperature at sensor 1 is 60.3°C; the one at sensor 2 is 27.6°C, etc.

If there is a network connection to other devices, the analogue values and digital conditions of the defined network inputs are subsequently displayed too.

In the example, network input 1 (=digital input 1) is in the "OFF" state, network input 2 is in the "ON" state, network input 17 (= analogue input 1) is equal to 25.4°C and network input 18 is equal to 10.6°C.

MENU Function overview

All of the function modules offer a wide range of information, measurement values, and parameters that can be viewed via the menu "Functions." To give users a quick overview of the main settings, experts can use the "user interface editor" to display all of the information that users need to see from all the menus. This information later appears in the menu "Function overview." Only the most important information and parameters should be entered in the menu "Function overview," and otherwise the overview would simply be too long. In other words, this menu is by far the most important interface to users.

The following display is an example of the system with one heater circulation, one heat meter and a DHW demand function:

```
HTG.CIRC.1      F: 5
OPERAT.:   RS
           TIME/AUTO
           STANDARD

T.roomSETBACK:  15 °C
T.room STD:     20 °C
            TIME PRG:
----------------------
DHW_DEM         F: 9
DHW_TEMP.:
T.DHW ACT:    60,5 °C
----------------------
INPUTS
```

The controller automatically switches from any menu to the function overview when it is switched on or if no operator's control is used for a few minutes, provided that the automatic option has been activated in the user menu (recommended).

Code for Technicians:

In order to enable all of the setting parameters, open the function "User" in the device’s basic menu and then select "Technician". Enter the product of 2^6 as the code!

The user interface editor

To keep the dialogue between users and the controller as simple as possible, an overview menu is automatically provided to present the most essential information that users need to know from the wide array of information available. The *FUNCTION OVERVIEW* serves that purpose in this device.
Experts can use the "user interface editor" at any time to create this overview. **The dialogue is complicated in accordance with the scope of information as to cover; the PC user interface TAPPS simplifies it.** We recommend that you use it in any case to provide an easy-to-follow overview of the most important information that users need.
The command can be found under the entry "USER INTERF. EDIT" in the menu *USER.* Once the menu is open, the cursor will be to the left of the display. Press the scroll wheel to open it and then select from the following commands:

S...	A source can be entered in the following dialogue for the entry. The first entry from a "source" always begins with this command. The next source command closes the previous one and opens a new one.
A...	If the value in the following entry can be changed, users may also make these changes. User area A
B…	--- " --- User area B
C…	--- " --- User area C
T...	If the value in the following entry can be changed, only **technicians** but not users may make these changes.
E...	If the value in the following entry can be changed, **experts** may make these changes. Only experts and technicians can see this entry, which is hidden for users.
>...	Enter lines. About your current position (line), information is to be entered. The number of lines must be entered.
<...	Delete lines. Information in and below your current line is to be deleted. The number of lines must be entered.
-...	Empty line that only appears in the editor; and entry can be made here at any point.

User areas A, B, and C are only important if you are using the CAN monitor. For the unit itself, it does not matter whether the entry is made with A, B, or C.

Assumption: A house with three apartments (three heating circuits in one control unit), each of which has its own CAN monitor:
Each of the three parties should only be able to access its own heating circuit; therefore, the first heating circuit is programmed for user area A in the function overview, while the second one is programmed for B and the third one for C. Experts can set the user level (such as A) on the CAN monitor. This ensures that user A only sees that heating circuit on the CAN monitor.

Programming example:

The example in the function overview we will start with is the date, the time (both of which users can change), and the collector temperature. Enter the command ***S*** (source). Now, the display shows:

```
S        User
```

User is a special feature as it does not have anything to do with commands or entries and is the only source information that does not produce a heading. It only serves as an indication of the date and time (summertime, wintertime). Enter ***A*** in the next line after you have entered the source of the information. Now the user can change the value. The current date will immediately be displayed.

```
S     User
A    Fr.  24.04.2009
```

When ***A*** is entered in the next line, the date appears again. It can be set to summertime or wintertime, depending on the date. Now, the current time appears in the function overview next term (such as summertime). Now, the display shows:

```
S     User
A     Fr.  24.04.2009
A    Summertime:
```

Use the command ***S*** to enter the collector temperature, but instead of *User* enter *Input*, and this information is in the input menu:

```
S     User
A     Fr.  24.04.2009
A     Summertime:
S    Input
```

Every time the command ***S*** is entered, the function overview displays a new bar across the width of the display indicating the new function along with a heading (in this example: *input*). ***T*** is in entered in the next line to set the collector temperature. It does not matter whether ***A***, ***E*** or ***T*** is selected for information that cannot be changed, such as collector temperature. In case of doubt (can it really not be changed?), select ***T***.

S	User	
A	Fr. 24.04.2009	Date
A	Summertime:	Time
S	Input	Bar and heading INPUTS
T	1: T.collector	The information (temperature) is always displayed for this purpose.

The function overview should now look as follows:

```
Fr.  24. 04. 2009
Summertime: 13:08
---------------------
INPUTS
1:  T.collector
         86.7 °C
```

Tips and tricks

- The commands Delete < and Insert > acquire the inputs of the number of lines.
- The overview is all the more useful for users if the information is provided in a proper sequence. Begin with the functions for maintenance and control of the heater.
- Each source command ***S*** inserts a separation bar in the function overview and the name of the "source" and is used whenever information is to be added for an additional function. In other words, ***S*** is always at the beginning of any function.
- If no new source command is set, the only selection in the subsequent lines concerns information for the previously inserted function.
- Select a device input output to display the respective values automatically (temperature, automatic / manual) in the function overview as headings.
- When entering outputs assigned to mixtures, proceed in descending order (such as makes or 8.9 before 8).
- Entries of INPUT or OUTPUT VARIABLES are admissible and do help users reach this menu directly in the function overview, but do not provide users with any truly valuable information. In other words, they can be confusing and should not be used. In addition,
- With any function is called (via ***S***), the heading of the selected function that allows the user a direct entrance into the function is always automatically entered in the overview. Users can thus reach all the areas of the selected function from the overview.
- All of the entries concerned are automatically deleted when the expert deletes this function in the menu *Functions* or turns it into a different function.
- A set monitoring function of the system from the “messages” module is always entered at the start of the function overview, but only if it is actually active.
- To make sure that the function overview truly provides an overview, you should only enter the most important information.
- Only a few parameters (mainly from the heating circuit control function) should be set by the user. We thus recommend that you use command ***A*** (user may change value) sparingly.
- Parameters that can be changed (set values) cannot be changed in the function overview (nor in the functions themselves) if these set values are transferred from another function via INPUT VARIABLE.
- Users only see "one level up" -- in other words, the information stored with the commands ***A*** (***B***, ***C***) and ***T***. Only experts see the information marked with E (expert), but experts are also not able to change this information.

MENU Inputs

The menu "***Inputs***“ primarily serves as overview over the measured values of the inputs and/or sensors. Furthermore, it allows the expert the parameterizing of all used inputs if employed the following procedure:
The line “*Inputs*” has already been selected and then pressed the scroll-wheel. Hereby the indication example is as follows:

```
1: T.collector
    78.3 °C   PAR?
2: T.DHW1
    45.8 °C   PAR?
3: T.DHW2
    61.2 °C   PAR?
4:  -----
   unused     PAR?
```

The temperature of the collector is currently 78.3°C etc.

Input 4 has to be set before

In the above indication example the sensor inputs 1 to 3 were already defined by the expert, while the input 4 is not yet fixed. In order to assign e.g. the cylinder sensor *buffer, bottom* to the input 4 the arrow has to be induced to the corresponding entrance into the parameterizing level PAR by means of the scroll-wheel. By pressing the wheel the entrance is effected and the indication "TYPE unused” appears.
First, it has to be determined which basic characteristic (TYPE) the sensor owns. Possible selections:

unused = The input is not used
ANLG. = Temperature, ambient temp., radiation sensors and others
DIGITAL = Direct ON/OFF control input (possible at each input!) from another function or connection of a **potential-free** switch contact between sensor connection and sensor earth (no voltage)
IMPULSE = Volume flow encoder, wind sensor (only for the inputs 15.16)

After having selected the type (according to example *ANALG.*; as it represents the analogue measured variable "temperature") all available parameter lines are faded in.
Indication example:

```
TYPE:      ANALG.
MEAS VAR: Temp.

DESIGNATOR
GROUP: General
DES:   T.solar flow

SENSOR:       Pt 1000
SENSOR CHECK:      no
SENSOR CORR.:  0.0 K

AVERAGE:    1.0 sec
```

A temperature sensor has the measured variable *Temp.* which is already faded in. A radiation sensor would need the measured value *Solar rad.*.

Menu Inputs

In the next step the name (designator) *Buffer, bottom* is to be assigned to the input 4. To do so, superset “designator groups” were specified such as *General, Generator, Consumer, Line, Climate* and others. *General* is a group which had to be taken over from old operating systems (< A1.21). Many names out of it are represented also in the other groups. *T.buffer bott* is put down in the group *Consumer*.
When selecting the "designator“, the computer suggests different texts with sequential index up to 9 by scrolling forward. T.buffer bott2). Instead of the “0” the index is faded out (e.g.: T.Buffer.C). In order to proceed quickly from one designator to the next, the key (x10) must be pressed at the same time. According to our example we select *T.Buffer.L.*

Indication example:

```
TYPE:      ANALG.
MEAS.VAR: Temp.

DESIGNATOR
GROUP: Consumer
DES:  T.buffer bott

SENSOR:        Pt 1000
SENSOR CHECK:       no
SENSOR CORR:    0.0 K

AVERAGE:     1.0 sec
```

Under "**SENSOR**“ the sensor type has to be specified. Possible selections are: *RAS* (KTY) or RASPT (Pt1000) for the ambient temperature sensor, *Pt 1000* for the standard temperature sensor and *KTY 10*.

If a short circuit and/or interruption occur, an active "**SENSOR CHECK**" issues **automatically** en error message in the **function overview**.
If "**SENSORCHECK**" is active, the **sensor status** is also available: OFF for a correctly working sensor and ON for a defective sensor (short circuit or circuit interruption). As the **sensor status** can also be set as source of an input variable (see menu “Functions”), it is thereby possible to react accordingly if e.g. the external sensor fails. The sensor status can optionally be selected for individual sensors or for all sensors together ("sensor status 17").

If there is a “**SENSOR CORR**“ such as 0.5K and a measured temperature of 60.0°C, 60.5°C is indicated. This corrected value is then used also internally for all calculations.

“**AVERAGE**“ means the temporal averaging of the measured values. Calculating a mean of 0.3 seconds leads to an extremely rapid reaction of the display and the unit. However, this can be expected to cause fluctuations of the value. A large mean slows everything down and is only recommended for the sensors for the heat meter. For simple measurements, around 1 – 3- sec. should be selected, for the provision of hygienic warm water with the ultra-speed sensor 0.3 – 0.5 sec.

Special abilities of the inputs

The inputs also permit as measured analogue variable *Voltage* including the necessary scaling. By this the determination of the value range is to be effected.
For program reasons, the same configuration options are available with all inputs for the ***voltage*** measurement value.
The following points must therefore be taken into account:

- **Inputs 1-7** and **9-16** can process a maximum ***voltage*** of **5 volts**
- The heat meter function cannot establish the *flow rate* for **inputs 15** and **16** from a voltage signal.
- **Input 8** also allows ***current*** and ***resistance*** as measurement.
- The process values of *voltage*, *current* and *resistance* are processed as dimensionless values (without decimal point).

Example:

```
TYPE:       ANALG.
MEAS.VAR:  Voltage
PROC VAR:  Voltage

DESIGNATOR
GROUP: General
DES: Fill level

SCALING:
 0.00 V   :    0
10.00 V   :  100
AVERAGE:    1.0 sec
```

Specification of the value range using scaling

The calculated voltage rate is averaged over 1 sec.

In addition, the **inputs 15 and 16** are able to detect faster pulses (pulse duration min. of 50 ms, pause of min. 50 ms). Thereby they are suitable as inputs for volume flow encoders.
The parameterizing of a pulse input leads to the following indication:

```
TYPE:       IMPULSE
MEAS.VAR:  Flow rate

DESIGNATOR
GROUP: General
DES: Flow rt sol

QUOTIENT:  0.5 l/Imp
AVERAGE:    1.0 sec
```

Per each 0.5 litres a pulse is received
The calculated flow rate is averaged over 1 sec

When selecting the measured variable *Flow rate*, also the “QUOTIENT” is to be entered. It describes which flow rate creates a pulse. Some of the function modules such as the heat meter have the ability to directly handle these pulses. The control unit calculates at the same time the effective flow rate as number by putting together the received pulses, the quotient and the calculation of a mean. This number is available as information also internally. All functions linked with pulse input decide independently on which pulses or flow rate they want to receive as numerical value.

Menu Inputs

With “TYPE“ *Pulse* and “MEAS.VAR” *Impulse* there is also available a “DIVIDER” at the **inputs 15 and 16**. It indicates how many pulses have to arise at the input, so that a pulse is passed on to the functions. Thus it is possible to realize a slow pulse meter in conjunction with a meter module. (see “Functions”)
That leads to the following indication:

```
TYPE:      IMPULSE
MEAS.VAR: Impulse

DESIGNATOR
GROUP: General
DES: Flow rt sol

DIVIDER:    10
```

Only each tenth pulse is passed on.

With “TYPE” *pulse* and “MEAS.VAR” *Wind speed* a “QUOTIENT” must also be stated for **inputs 15 and 16**. Here the frequency per one km/h must also be set.
Example: A wind sensor issues one pulse (=1Hz) per second at a wind speed of 20 km/h (= 1Hz). Therefore, the frequency at one km/h is equal to 0.05Hz.

Connection of electronic sensors in version DL

Electronic sensors for temperature, pressure, humidity, differential pressure, etc. are also available in the **DL** version. In this case, the supply and signal transmission takes place via the **DL bus**.

Due to the relatively high power requirement, the "**bus load"** must be considered:
The controller UVR 1611 has the maximum bus load, 100%. For example, the electronic sensor FTS4-50**DL** has a bus load of 39%, therefore up to a max. 2 FTS4-50**DL** can be connected to the DL bus. The bus loads of the electronic sensors are listed in the technical data of the respective sensors.
The advantage of this signal transfer lies in the fact that sensor inputs are not necessary, but rather the information (signals) are transferred as a network variable as with the CAN bus (see: MENU network/input variables).

MENU Outputs

The menu *"**Outputs**"* primarily serves for the switching between the automatic and manual mode of the outputs. As in the status line of the outputs (top symbol line on the display) it is not possible to share information concerning the speed stages (if active), this indication was put down in the output menu. The parameterizing of all used outputs is effected by employing the following procedure:
The line "*Outputs*" has already been selected and then pressed the scroll-wheel. Hereby the indication example is as follows:

```
1: Solar pump1
   MAN./ON     PAR?
2: HC pump1
   AUTO/OFF    PAR?
   Speed stage:   0
3: HC mixer1
       AUTO    PAR?
   open:  OFF
4: clse:  OFF
5:    -----
      -----     PAR?
```

Pump solar 1 is switched on in **manual mode.**

Heating circuit pump 1 is switched off in **automatic mode.**

Output A4 together with A3 is configured as a mixer output.
Input 5 has to be set before.

and so on

Hence, the output 1 as solar pump, the output 2 as heating circuit pump and the outputs 3 and 4 as mixer (open/closed) were already determined.
According to the example the outputs 1 and 4 are set to the automatic mode and indicate the current operating state (OFF). If the pointer is placed behind that position, the switching to manual mode on/off is possible (pressing the wheel / selecting the state / pressing the wheel). The current status of the output is immediately shown up in the status line of the outputs. As the speed control function at output 1 is active, the current speed stage is also faded in. This only can be changed in manual mode for experimental purposes.
As seen at the output 5, neither the "Designator" nor the status of the output appears before the parameterizing (similar to the parameterizing of the inputs). Thus, the corresponding symbol would also be missing in the top display line of the output status.
If e.g. the solar pump is to be assigned to the (not yet determined) output 1, the arrow must be induced to the corresponding entrance of the parameterizing level *PAR?* by means of the scroll-wheel. Pressing the wheel causes the entering and the following indication appears:

```
TYPE: unused
```

First, it has to be determined which basic characteristic (TYPE) the output should own. Possible selections:

SWITCH.OUTP = Output only can effect switches (no speed control)
SPEED CTRL = Output is prepared for the speed control

At the outputs 3, 8, 10 and 12 the suggestion MIXER appears instead of the type *SPEED CTRL*, while each first output means "Mixer open" and the next-following (4, 9, 11 and 13) "Mixer closed". In other words, if output 4 is defined as switching output and if output 3 is parameterized afterwards as mixer, output 4 automatically becomes the second mixer output!

Menu Outputs

After selecting the type (such as *SPEED CTRL*, since a solar pump shall run speed-controlled at output 1 later on) all available parameter lines are faded in.

```
OUTPUT STATUS:
TYPE: SPEED CTRL

DESIGNATOR
GROUP: General
DES:       -------

MODE:  Wave packet         (This line is suppressed in SWITCH OUTP)
DELAY:          0 Sec      Rise-delay time
RUN-ON:     0 Sec          After-running time
```

In the next step the name (designator) *Solar pump 1* is to be assigned to the output 1. As in the parameterizing of the sensors superset "designator groups" and a sequential index until 9 (e.g. *Solar pump 4*) were specified. Most suggestions like *Solar pump 1* are to be found in *General*. In order to proceed quickly from one designator to the next, the key (x10) must be pressed at the same time.
The waveform can be selected via the parameter "**MODE**" of the speed control. While usual commercial pumps are controlled by wave packets (fast switching on/off of the motor), ventilated motors need a phase control (such as a light dimmer).
NOTICE: The menu allows a choice between wave packet and phase angle however in the standard version the output of waveform "phase angle" is not possible. Special versions on request.
"**DELAY**" permits the setting of an adjustable rise-delay time.
Via "**RUN-ON**" the cutoff-delay time of the output can be determined.

If after the entrance the TYPE *MIXER* has been selected, the following indication appears:

```
OUTPUT STATUS:
TYPE: MIXER

DESIGNATOR
GROUP: General
DES:       -------

RUNTIME:  2.5 min
```

With "**RUN TIME**" the total running time of the mixer motor has to be fixed.
Where there are stability problems in the mixer control circuit, the mixer running time can be increased or reduced to lengthen or shorted the pulses or pauses. This has no influence on the remaining running time, as this is always loaded with 20 minutes upon a direction change or enable.

WARNING:

The factory setting of the total running time of the motor is 0 seconds! Thus the mixer is not controlled. Unfortunately it is not possible for reasons due to the programming technique to preset another value in the factory setting. Therefore this parameter must be entered while setting the mixer output.

The submenu option "**OUTPUT STATUS**" represents a special characteristic. Here a list of all functions and messages (including status) controlling the output is put down. By this it is easier to understand at the system, why a pump is being controlled right now or not. In addition, it is possible to enter the corresponding functions out of the output status in order to check the function status there (see function modules).
If an output is controlled by multiple functions, the output switches ON, if at least one function is active (OR – function)!

The outputs (manual and automatic) are controlled only 30 sec. after the starting up of the control unit.

Particularities of output 14

Output 14 basically serves as a data link (DL-Bus9 but can also be used as a switch output for switching an external relay and can be configured accordingly (*unused / SWITCH.OUTP / DATA LINK*).

Output 14 as data link:

Output 14 serves as a data link (DL bus) for the recording of measured values ("data logging") via C.M.I. or D-LOGG and/or as a bus line for various sensors.

If the control receives data via the network, it is possible to send a second data record with the network input data via DL. In this case the data logger recognises the second data record as a virtual second UVR1611 controller. However this option can only be used if the data logger's second DL input is unused.

```
OUTPUT STATUS:
TYPE:  DATA LINK

DESIGNATOR
GROUP: General
DES: Data link

NETW.INP.=>DL.: no
```

Output 14 as data line / DL-bus

For a “yes” input: Network inputs are output to the DL bus as a second data record

Output 14 as switching output:

When required, output 14 together with an external 12 V / 20 mA relay (laid to ground) can be used as an additional switch output. The relay must be fitted with a suitable free-wheeling diode.
Recording of the measured values with the data logger and capturing sensors via the DL bus link are not possible in this mode.
With device type UVR1611E (special type for control cabinet fitting) output 14 is able to be used **simultaneously** as a switch output and a DL link (DL bus). That is why for this type of device in the setting "**`UVR1611E: yes`**" the data link can be activated in addition to the switch output. This option may only be activated for type UVR1611E and results in a malfunctioning of the output with other types of devices.

```
OUTPUT STATUS:
TYPE:  SWITCH.OUTP

DESIGNATOR
GROUP: General
DES: HC Pump

DELAY:      0 sec
RUN-ON:     0 sec

UVR1611E: no
NETW.INP.=>DL.: no
```

Output 14 as switch output

This option may only be activated with type UVR1611E.

Particularities of outputs 15, 16

Output 15, 16 = analogue outputs. These outputs provide a voltage between 0 and 10V for performance control of modern burners (burner modulation). They can be controlled by a PID function module, but also by other functions with an analogue value. The "scaling" offers the possibility of adapting the arithmetic value to the control range of the downstream device.

If several functions act simultaneously on one analogue output, the higher value is output.

By activating the analogue output via a **digital command** (ON), a **dominating** output voltage between 0.00 and 10.00 V can be specified.

Output of the calculated value takes place either as a voltage (0-10 V) or as a PWM signal. In PWM (pulse width modulation), a square wave signal is created with a voltage level of about 10 V and a frequency of 2 kHz with a variable duty cycle (0 - 100%).

Examples for different scaling:

Correcting variable for PID function: Mode 0-10 V, correcting variable 0 should correspond to 0 V, correcting variable 100 should correspond to 10 V:

```
OUTPUT STATUS:
MODE:  0-10 V

SCALING:
      0  :  0.00 V
    100  : 10.00 V

Outp. voltage digital
command:   10.00 V
```

The correcting value is imported without a decimal point

Temperature value, e.g. of an analogue function: Mode PWM, the temperature 0°C should correspond to 0 %, 100°C should correspond to 100 %:

```
Output status:
MODE:  PWM

SCALING:
      0  :  0.0 %
   1000  : 100.0 %

Outp. voltage digital
command:   10.00 V
```

The temperature value is imported in 1/10°C **without** a decimal point

Burner performance, e.g. from the functions DHW demand or maintenance: Mode 0-10V, a burner performance of 0% should correspond to 0 V, 100% should correspond to 10 V:

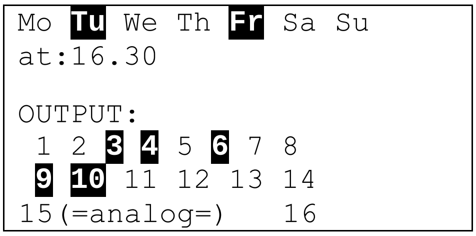
```
Output status:
MODE:  0 - 10 V

SCALING:
      0  :  0.00 V
    100  : 10.00 V

Outp. voltage digital
command:   10.00 V
```

The percentage value is imported without a decimal point

Anti-Blocking Protection

Circulating pumps, which do not run for longer periods (such as: heating circuit pump during the summer), often have starting problems as a result of inner corrosion. This problem can be easily avoided by activating the pump periodically for 30 seconds.

The menu ***ANTI-BLOCKING PROT*** added after the output 16 permits to indicate the time and the outputs which are to receive this anti-skid control.



```
Mo Tu We Th Fr Sa Su
at:16.30

OUTPUT:
 1 2 3 4 5 6 7 8
 9 10 11 12 13 14
15(=analog=)   16
```

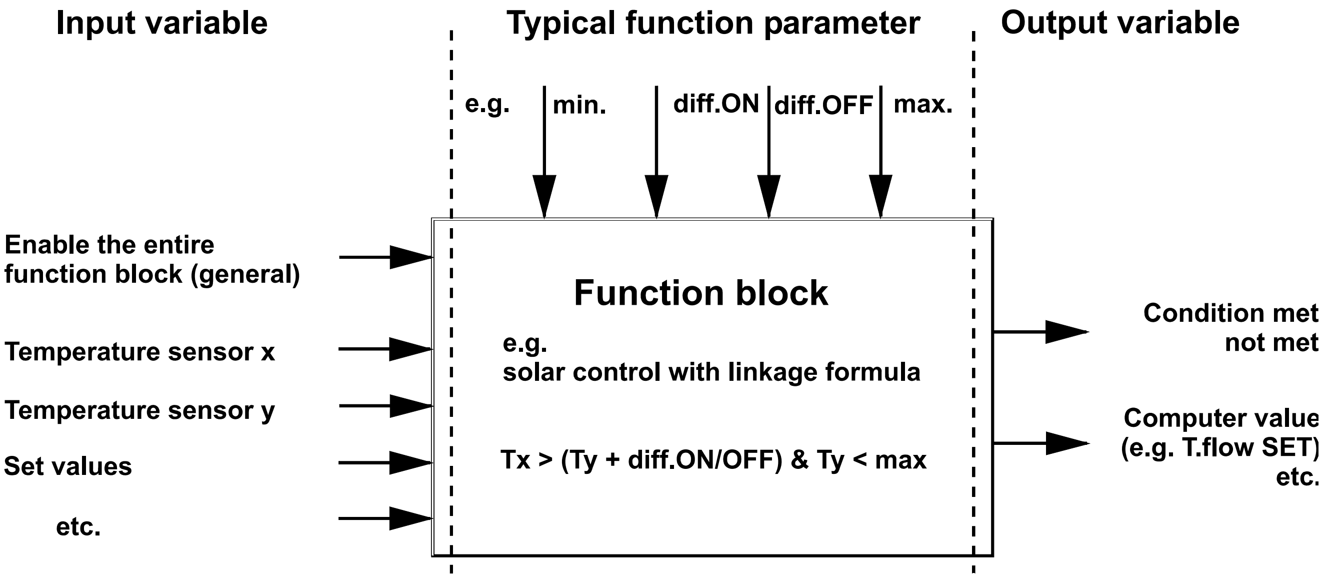
In the example the pumps 3, 4, 6, 9 and 10 are activated for 30 seconds on Tuesday and Friday at 16,30 if the output has not been active since the controller start or the last call of the anti-skid control. However, the computer does not connect all outputs at the same time, but begins with output 3, switches after 30 seconds to output 4 and so on. In order to save energy a switching time is selected when neither the industry nor the typical households are using the mains supply with maximum consumption. In addition, it will be sufficient to set one day per week.

MENU Functions

The basic standards of the function menu

In the menu “Functions“ all linkages concerning the control are to be set and parameterized (therein the control engineering of the entire solar and heating installation is described!). To this end, the unit has a set of function modules which can be registered successively and also several times in the list “Functions”.

Schematic diagram of a function module:



By means of the input variables of the function module the module receives all data necessary for the internal decision. Most of it will be temperature data. In addition, each module has the input variable “*ENABLE*” which means a general permission for processing the task.

Within the function module the decisions and the set values are calculated by means of the data and settings and made available as output variable.

Thus, a function module only can fulfill tasks when connected with the other parts of the system (inputs, outputs, other modules) by its input and output variables.

The following example shows how to set a new function.
Indication example from the menu Functions:

```
5: CHARGING PUMP
   CHRG PUMP1   PAR?
6: NEW FUNCTION
   -----       PAR? ◄
```

The function module already has been assigned to function 5 “Charging pump”.
A new module can be entered.

Employing the following procedure a new function module can be added: Induce the pointer to *PAR?* and press the scroll-wheel. The following text appears in the display:

```
TYPE: HTG CIRC.CTRL
DES.:    -----

Scope of TIME PROG:
Number of prog.:     1
```

At present the computer suggests the module *HTG CIRC.CTRL* as a new function with all its options. Assuming that the module *SOLAR CONTROL* is to be added, press the wheel once again under “TYPE” in order to switch to the selection of a function module. Now, the desired module can be selected by means of the scroll-wheel. By pressing the wheel once again the new module *SOLAR CONTROL* is called.
Indication example:

```
TYPE:  SOLAR CONTROL
DES.:    -----

ADD?              no
```

In the line DES.: a text for the module can be selected (by the usual operation – press / select text / press). Assuming that a solar circuit already exists as function number 1 with the designation ”SOLAR 1” “SOLAR 2” will be selected.
In addition, the question "ADD? *no*” is to be answered with *yes*. The computer has now taken over the module *SOLAR CONTROL* in the list under the number 6 and immediately shows the menu of this function which is now the module solar control SOLAR 2.
Hereby the **indication example** is as follows:

```
DES.:   SOLAR2
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:

COLLECTOR TEMP.:
T.coll.ACT:  -----
T.coll.MAX:    130 °C
       .....
```

and so on

Input variables

Input variables serve as link to sensors and also to output variables from other function modules or definable parameters. The collector and the cylinder sensor are the typical input variable of the module *SOLAR CONTROL*. Another typical input variable for the module *HEATING DEM.* is the calculated set temperature of the flow (T.flow SET) of the module *HTG CIRC.CTRL*.
Sometimes also simple parameters can be defined as input variable, when it is reasonable to apply the computation results of a function module (= its output variable) as thermostat threshold in the new block. Like the threshold min of the module *CHARGING PUMP* which is not a function parameter but an input variable.
Each function module has the input variable “Enable …” which represents a general permission of the entire function. This enables a simple interlock and/or enable of the entire module by another one.
Indication example:

```
DES.:   HTG CIRC.2
FUNCTION STATUS:
INPUT VARIABLE:        ◄
OUTPUT VARIABLE:

OPERAT.: TIME/AUTO
```

other text lines by scrolling

Induce the pointer to “INPUT VARIABLE” and press the wheel (in the following designated as “Enter”). Hereby the **indication example** is as follows:

```
HTG CIRC.2

ENABLE HEATING CIRC.:
Source:   User
Status:   ON
```

The line “ENABLE HEATING CIRC.:” represents the general permission of the entire function module. The user has effected the enable (ON) as “(signal) source”.

Instead of ***User*** another source of enable can be selected, such as:

♦ ***Input***	The must be an input that is set to ***DIGITAL*** during input parameterization.
♦ ***Output***	In some cases the outputs of the control unit are controlled by multiple modules (such as a common solar pump). Via ***Output*** it is also possible to use a common output as enable control.
♦ ***NW status***	The enable is effected according to the **status of the network** (See chapter Network/Timeouts). The network status can optionally be selected for individual inputs or for all network inputs together ("network status 33").
♦ ***Sensor st***	The enable is effected according to the **status of the sensors**. A correctly working sensor has status **OFF** and a defective (circuit interruption or short circuit) **ON**. Thus, for example, failure of the external sensor can be correspondingly reacted to (e.g. for the function "**messages**"). The sensor status can optionally be selected for individual sensors or for all sensors together ("sensor status 17").
♦ ***Message***	The enable of the function module depends on the status of a ***message***.
♦ ***Network***	A function module of another device from the CAN network is responsible for the enable of the function *HTG CIRC.2*. (**digital** network input variable).

Any other previously defined function can release the function HTG CIRC.2.

If another function module (also from the network) has been selected as a source, its first output variable (or first network input variable) in the following. **An analogue value** (temperature, computation result) **is not suitable for the enable control**. An enable control only can be a switch, thus a digital value such as the output status of an already registered function module. If a module has several output variables it is possible to choose between these variables.
If the enable is to be made via a digital input, output or another module, it is also possible to select the enable via ***normal*** and/or ***inverse***. Thus, a module also can be enabled by a switch-off status.
Indication example of the input variable “ENABLE PUMP”: of the function module *HTG CIRC.* The heating circuit pump only should run, when the cylinder charging is not active at the moment (cylinder priority) via the function module *CHARGING PUMP*.

`ENABLE PUMP:`	
`Source:  CHRG PUMP1`	Enable via the module with that specification
`1 : Stat.chrg pump`	via the output status of the module
`Mode:    invrs.`	via the inverse output status of the module
`Status:  ON`	heating circuit pump is currently enabled

Hence, the enable of the heating circuit pump is controlled by the function module *CHARGING PUMP* with the specification *CHRG PUMP 1*. Since the mode is ***inverse***, the enable is made always when the charging pump is not running. This also happens at the moment, since the status of the enable indicates *ON* (enabled).

By scrolling the next input variables of the module *HTG CIRC.2* appear right after the enables.

```
ROOM TEMPERATURE:
Source:   Input
14 : Temp.room2

FLOW TEMPERATURE:
Source:   Input
11 : THC flow2
    .......
```

etc.

Hence, the module *HTG CIRC.2* needs further input information such as room temperature, flow temperature etc.
Just as the input variable “ENABLE“ also an input of an unit from the CAN-network can be set as source for the temperatures via *network*. By this it is possible to transfer the information of the ambient temperature to multiple control units.

Output variables

They represent the result of a function module. They can be used directly for the switching of hardware output or serve as input variable for another module. If this output variable is to be used directly for the switching of a pump, the assignment can be made in the menu “OUTPUT VARIABLE” of the module. The output variable is always available as input variable for the other modules **with or without** assignment to a real output.
The previous example, the module solar control unit, normally creates an output out of a difference function together with a thermostat function (such as: difference already reached, cylinder temperature limit not yet reached => output variable (= *ON*). This information can now be assigned to a hardware output under ”output variable”.
Indication example (we are already in the menu of the function 6 = SOLAR 1):

```
DES:    SOLAR 1
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:      ◄

COLLECTOR TEMP.:
```

Other text lines by scrolling

After entering the menu “OUTPUT VARIABLE” the following menu appears:

```
SOLAR 1

SOLAR CIRC:
Status:   OFF
OUTPUT:
 1 2 3 4 5 6 7 8
 9 10 11 12 13 14
 15 (=analog=) 16
```

The output variable actually shows the status *OFF*, i.e. either the set temperature difference has yet not been reached or the cylinder limit has been exceeded. Now, the variable is to be assigned to the real (hardware) output 1.
To do this, induce the pointer to 1 and make the assignment by the usual operation - / press / darken the 1 / press.
Now, the display shows:

Above the visible range

```
SOLAR 1

SOLAR CIRC:
Status:   OFF
OUTPUT:
 1◄2 3 4 5 6 7 8
 9 10 11 12 13 14
 15 (=analog=) 16
```

Below the visible range

Hence, the function *SOLAR 1* influences the output 1.

A common pump with valves is often used in solar thermal systems with multiple consumers. Therefore, we assume the following:

Double circuit solar thermal system with common pump and three-way valve
Output 1 = Common pump
Output 3 = Three-way valve

In this example both the output 1 and the output 3 have to be activated in *SOLAR 2* (1 and 3 darkened). Output 1 has already been correctly assigned in the function *SOLAR 1* according to the above example.
Now, the display shows:

```
SOLAR2

SOLAR CIRC:
Status:   OFF
OUTPUT:
 1 2 3◄4 5 6 7 8
9 10 11 12 13 14
15 (=analog=) 16
```

Above the visible range

Below the visible range

Hence, *SOLAR 1* with output 1 (only with pump) would switch the first solar circuit and *SOLAR 2* with output 1 and 3 (pump and valve) the second one.
No matter if all these assignments were made or not, the variable is available for other function modules.
The output variable (switch status pump on/off) from *SOLAR 2* and possibly also from *SOLAR 1* in the example can be assigned as input variable to the module *PID-CONTROL* (speed control). By this the starting of the speed control for the common solar pump can be switched via the input variable “ENABLE”.

The following output variables are available in the heating control unit function:

- Set Flow temperature – To further use in the burner demand
- Effective set room temperature – As a set value for the speed control when instead of the mixer the room temperature is only controlled by the module *PID-CONTROL* via the circulating pump.
- Heating circuit pump – Switches the respective hardware output
- Mixer - Assignment to the two hardware outputs *Mixer OPEN / CLOSED*
- Maintenance mode - ON If the maintenance function requires the operation of the heating circuit
- Frost protection mode - ON If the heating circuit runs in frost protection mode

Function parameter

Function parameters are set values which provide the user with the possibility to adapt the ready (i.e. with all preset function modules) control unit to the characteristics of his system.

In the module *SOLAR CONTROL* these are parameters such as switch-on/off difference, maximum limit to the possible sensors. In the module *HEAT.CIRC.CONT* these would be parameters such as heating characteristic, desired room temperature in the heating and lowering mode and others.

In addition, the function parameters of the time windows in some of the modules allow a time-controlled enable and blockage of the module or of parts of the module. There are maximum 5 time programs with each 3 time windows available per function module. Each time program can be assigned separately to undefined days.

As the function parameters are an essential integrated part of a function module, they are explained extensively in the description of the various function modules.

Time programs

They are set up in the same way in almost each function module and thus can be described here in a general way.

Assumption: The module *HTG CIRC.CTRL* has already been defined twice as the function (F3 = HTG CIRC.1, F4 = HTG CIRC.2) with two time programs. We will now define separate time programs for “HTG CIRC.2” on weekdays and the weekend.

In the menu "*FUNCTIONS*" where we are, scroll to the display:

```
   SOLAR2     PAR?
3: HTG CIRC.CTRL
   HTG CIRC.1 PAR?
4: HTG CIRC.CTRL
   HTG CIRC.2 PAR?      ◄
```

Remaining text from function 2

And after entering into “HTG CIRC.2”:

```
DES: HTG CIRC.2
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:

MODE: RS
      NORMAL
ROOM TEMPERATURE:
T.room.ACT:   20.7 °C
T.roomSETBACK:   15 °C
T.room STD:  20 °C
          TIME PRG: ◄
```

Other text lines by scrolling

After entering the menu “TIME PRG:” all time programs are listed sequentially with their time windows.
Indication example:

```
Mo Tu We Th Fr Sa Su
05.00 - 07.00  h
12.00 - 22.00  h
00.00 - 00.00  h
```

Time window not used

If the first time program applies to the period from Monday to Friday, these five symbols are darkened after each other. **Indication example**:

```
Mo Tu We Th Fr Sa Su
06.00 - 07.30  h
12.00 - 21.00  h
00.00 - 00.00  h
```

The first time program provides for two heating periods on workdays (Mo-Fr) from 6 a.m. to 07:30 a.m. and then again from 12 noon to 9 p.m.
It is suggested that only one time program be used when the first settings are made for a function module. This can be changed for up to five time programs before the module's entry is injured in the function list by using "add function?" Each module has five entries at the end of the menu to change a number of time windows and programs for a function module already entered.

```
DELETE FUNCTION
CHANGE FUNCTION
INSERT FUNCTION
```

Use “**CHANGE FUNCTION**” to open the following menu for the example above:

```
TYPE: HTG CIRC.CTRL
DES:  HTG CIRC.2

Scope of TIME PROG:
Number of prog.:  1
No. of windows:   3
with set value?   no

CHANGE?          no
```

Here, the number of time programs desired and the number of windows per program can be set again.
The question "with set value? ***no***" means that the same set value for the module is to be used for all time programs (such as desired room temperature during heating periods). The question "with set value? ***yes***" allows you to assign a separate set value to any time window of a time program. In the example used about of *HTG CIRC.CTRL*, it is possible to set a room temperature for each heating period.
As soon as the assignment has been made (such as two programs with three time windows each), the change has to be acknowledged; to do so, scroll to the end of the menu and confirm CHANGE? by pressing *yes*.

Menu Functions

Functions that have already been entered can be deleted at any time. It is recommended that a function be deleted if function data are available from a similar project and the changes that have to be made are minor. To do so, you will find the command "**DELETE FUNCTION**" at the end of the menu in each function module.
The command "**INSERT FUNCTION**" allows you to insert a function module before the one currently selected. This function allows you to add a function module before or after another one in the overview at any time.

Function status

Function modules have a multitude of parameters that influence the output variables accordingly. It is thus not that easy to tell why an output is blocked or cleared any given time. For example, *HEAT.CIRC.CONT* has some 10 function parameters, such as enable, mood, switch-off conditions, that determine whether the heating circuit pump is enabled. The entry "FUNCTION STATUS" at the beginning of every basic menu provides a quick overview of the status.
This submenu lists all of the effective functional elements, with a ✓ to the right indicating that is activated. Each functional element that does not have a check at the end of the line blocks the output variables at that moment.

Indication example:

```
ENABLE HTG CIRC.2      ✓
ENABLE PUMP            ✓
ENABLE MIXER           ✓

HEATING CIRCUIT PUMP:
Status:   OFF
MIXER:
Status:   off
Rem rutime 0.0 Sec

FROST PROT.MODE         ✓

LIMIT T.flowSET :
T.flow: SET < MAX.     ✓
T.flow: SET > MIN.     ✓

SHUTDOWN CONDITIONS:
T.room: ACT < SET
T.flow: SEt > MIN.     ✓
```

The remaining running time of the mixer is reloaded after a direction change or granted enable and always equals 20 minutes independent of the set mixer running time.

In this example, the heating circuit pump is currently blocked (status = OFF) because the specified room temperature has been exceeded (Switching condition *T.room: ACT < SET* is not fulfilled).
In addition to the main functional elements, this submenu also always indicates all of the output variables along with their current output status.

MENU Messages

This module allows messages (error, malfunction, and others) to be triggered by specified events **that last longer than 10 seconds**. Any messages issued are automatically entered in the function overview. In addition, output variables provide switching signals as long as the message is current. A total of eight message lines can be set up, each of which should be understood as an independent module.

As no message lines are preset ex works, "unused PAR?" is displayed on all eight lines when the menu is opened. Open the parameter level to assign variables and set parameters as with all functions. Each message line consists of the following variables:

Input variables:	**Output variables:**
Enable message **Activate message** = Triggering event Delete message = Input to delete the Message	Status message Output (stanard ON) = Single output signal during the message Output (dominant ON) = Overwrites the assignment with an "on" signal Output (dominant OFF) = Overwrites the assignment with an "off" signal Output (Reset fault) = Creates a pulse three seconds long when resetting

Special features:

- Each message line has a deletion input that can be assigned to an acknowledgment key via a **digital input** or that allows for automatic reset with another function. The message can only be deleted by using the scroll wheel with ***User/*OFF**. ***User/*ON** causes the **automatic** deletion of the message, as soon as the cause of the message clears.
- An **acoustic warning signal** can be activated.
- Outputs assigned under the output variables ***Output dominant…*** are set for the output status of the message line regardless of any assignments from other modules or from a current manual mode.
- A separate output variable that generates a pulse lasting three seconds when the message is deleted is available in order to reset external devices (only possible if the message type is "malfunction").
- The line “DELETE MESSAGE” appears in the function overview only if the event is no longer current. If deleted, the message will automatically be removed from the function overview.

Menu Messages

In the **example** below, a compare function as boiler thermostat shall issue the message "excess temperature" to set off an acoustic warning signal when excess temperature is in the boiler (=event), switch on dominantly the heating circuit pump and the cylinder charging pump and switch off the burner demand:

`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`MESSAGE TYPE:`	Header selection: MESSAGE, FAULT, WARNING, ERROR
`WARNING`	A warning is issued
`MESSAGE GROUP:`	What group was the name of the message selected from?
`Default`	General or user-defined (only with TAPPS)
`MESSAGE CAUSE:`	What event caused the message to be issued?
`Excess temp.`	Excess temperature triggered the message
`WARNING TONE   yes`	As soon as this event occurs, a warning tone is issued

In addition, appears as the message type "FAULT":

`Reset fault?`	Pass on the scroll wheel to set off a pulse that last three seconds in the output variable "Reset fault"

The respective pump outputs are darkened to show that they are assigned in the output variables under the heading "OUTPUT (dominant ON)." This ensures that the pumps are always reliably switched on when the event occurs. At the same time, assigning this output for the burner demand via "OUTPUT (dominant OFF)" ensures that the burner is switched off in all cases.

In general: If outputs are triggered by means of "dominant" commands (even for other modules that have such options), all control signals from simple assignments are overwritten -- **including manual mode**. If two different dominant signals come into an output at the same time (ON and OFF), the signal "dominant OFF" has priority.

If the event sets off the message as described above, the function overview will begin with:

`---------------------`	
`WARNING TONE OFF:`	
`---------------------`	
`WARNING`	
`Excess tTemp.`	
`since:`	
`29,01.    at    15:18`	
`DELETE MESSAGE:`	Is not displayed if the event is still current
`---------------------`	

As soon as the cursor is on "**WARNING TONE OFF**", press the scroll wheel to switch off the acoustic warning signal and delete this line from the display.

Only for the message "**FAULT**": The additional line "reset fault?" sets off a pulse that lasts for three seconds with the scroll wheel in the output variable "Unlock Malfunction" regardless of whether the event to trigger this reaction is still current at this point or not. If this event no longer occurs after the pulse, the entire message is deleted at the same time.

MENU Network

This menu contains all of the information and settings needed to set up a CANopen network.

Entire menu view:

```
Node No.:        1
ENABLE:         ON
Autooperat.:   yes
Status:     operat

OUTPUT VARIABLE:
DIGITAL:
ANALOGUE:
Transmission cond.:

INPUT VARIABLE:
DIGITAL:
ANALG.:
Timeouts:

DATALOGGING

NETWORK NODES:
```

The device has network address 1
Participation in bus communication admissible
Device communicates with other bus participants without master and is active

- **Node No**. - Each device in the network must be allocated its **own** address (node number 1-62)
- **Enable** - Without network release ***ON***, the device cannot send or receive messages; it would thus not be able to take part in communication.
- **Autooperat**. - If the network only consists of devices from the UVR1611- family (UVR1611, CAN monitor, C.M.I., etc.), autooperat. must be set to *yes* (normal case). If there is a master or network manager in the network**,** autooperat. must be set to *no*.
- **Status** – Autooperat. is set to *yes*, the status automatically switches from *init* → *pre-op(erational)* → *operat(ional)* after the controller is started according to a preset procedure. Only then is communication possible. If there is a bus master, it will switch the nodes to *operational*.

Output variable

A total of 16 digital and 16 analogue network outputs can be programmed. All of the input and output statuses, output variables for the functions, network status, sensor status, and the status of messages can be used.

```
DIGITAL NETW. OUTPUT
-------------------
OUTPUT  1:
Source: HTG CIRC.1
 2: Pump Status
Status:  ON
Target:        CAN
```

Example: The digital network output 1 has been allocated the status of heating circuit pump 1, the current pump status is "ON"

With input Target "DL" it is possible, to switch specific sensors on or off via the DL bus. More precise instructions are contained in the data sheets for these sensors.

Parameterizing of the analogue output variables takes place in the same way.

Transmission conditions:
This menu determines the conditions for the transmission of output variables.

```
DIGITAL OUTP.:  1..16
with change      yes
Block. time:    10 Sec
Interv tm mtr:   5 Min

ANALG. OUTP.:    1..4
with change    > 30
Block. time:    10 Sec
Interv tm mtr:   5 Min
        …
        …
```

The transmission conditions are divided into five groups:

- digital network outputs 1-16
- analogue network outputs 1-4, 5-8, 9-12, and 13-16

with change yes/no:	Transmission of a digital message if status is changed.
with change > 30:	If the last analogue value transmitted has been changed by more than 3.0 K, the data are transmitted again (= 30 because numbers are transmitted without a comma).
Block. time 10 sec:	If the value is changed within 10 seconds of the last transmission by more than 30 (3,0K), the value is not transmitted anew for another 10 seconds.
Interv tm mtr 5 min.:	The value is transmitted every five minutes even if it has not changed by more than 30 (3.0K) since the last transmission.

Input variable

A total of 16 digital and 16 analogue network inputs can be programmed. They are indicated by the **transmission** node number and the number of the network output variable of the **transmission** node.

```
INPUT 1:
NW Node:           2
analg.NW.outp.:    1
Source:          CAN
Value:           234
```

Assumption: The analogue network output 1 on CAN node 2 is assigned to outdoor temperature. Transmission thus does not include the unit or a label. Therefore, the reception node only receives the number 234. Only when linked with a function, such as input variable outdoor temperature in the function module HEATING CIRCUIT is the correct value displayed: 23.4°C.
When the controller is started, the entire analogue network input variables are set to 0 and all of the digital ones to OFF.

Another **example**: input of the volume flow from the electronic volume flow encoder FTS4-50DL via the **data link (DL-Bus)**. Sensor address = 1.

```
INPUT 2:
NW Node:          1
analg.NW.outp.:   1
Source:          DL
Value:          357
```

sensor address
index of the volume flow (see data sheet of respective sensor)

Parameterizing of the digital input variables takes place in the same way.

Timeouts

Timeouts are monitoring functions that can cause reactions in the control strategy, if bus messages are missing (such as a result of a device failure). The timeouts are divided into 8 groups of network inputs:

- Digital network inputs 1-4, 5-8, 9-12 and 13-16
- Analogue network outputs 1-4, 5-8, 9-12, and 13-16

```
DIGITAL INP.:   1...4
Timeout:         60 Min
```

As long as the information is being read from the CAN bus, the **network status** is OFF. If the value has not been updated since the set timeout, the network status changes from ***OFF*** to ***ON***. You can then program the system to react to the failure of a network node or a lack of information. The network status can optionally be selected for individual inputs or for all network inputs together ("network status 33").

Network and network status are also available as sources for input variables in all function modules and messages.

Data logging

There are 2 data logging possibilities:

Via the data link (DL-bus):
When data logging via the DL-bus, there is a constant data flow from the controller to the C.M.I. or the data converter D-LOGG. The values or states of all inputs, switch outputs and the values of up to two heat meters are specified as a data record. For more information see chapter **Outputs/Special Features of Output 14**.

Via the CAN-bus:
CAN data logging is only possible with the C.M.I.. In contrast to data recording via the DL-bus, the data to be logged via the CAN-bus are freely selectable. Likewise, there is no constant data output. Upon querying of a C.M.I. the controller stores the actual values in a logging buffer and locks this to prevent it from being overwritten (requests received from a second C.M.I.) until the data are read out and the logging buffer has been released again. In the menu **Network/Data Logging** the parameters for data logging are defined via a CAN bus connection.

The essential settings of the C.M.I. for the data logging via CAN bus are described in the online help of the C.M.I..

Menu overview:

```
MASTER NODE:  56
Timeout:  60 Sec

digital values
analogue values

delete all def.
```

WARNING! Settings are immediately deleted without prior security prompt!

Master Node – The controller is allocated a C.M.I. as logging master. The logging command of this master has absolute priority; this means that the logging buffer is **always** updated at the command of the master (even if it was blocked by another C.M.I.) unless the data is being output (sent) to another C.M.I. at that time.

Timeout – The duration of the logging buffer data block has a time limit. Once this time span has expired, the buffer is again released by the controller.

Digital and Analogue Values - Each controller can output a maximum of 26 digital and 32 analogue values using 2 data records that are defined in the menu "**Network/Data Logging**" of the UVR 1611. Each data record comprises 16 analogue and 13 digital values as well as 2 heat meters:

	Digital	**Analogue**	**Heat meter**
Data record 1	**1 – 13**	**1 – 16**	**1 – 2**
Data record 2	**14 – 26**	**17 – 32**	**3 – 4**

If, for example, a digital value is to be stored in data record 2, then it must be defined as digital value 14 or higher.

Speed stages of the outputs:

If the speed stages of an output are also to be captured, the digital value must have the same number as the corresponding output, e.g., output 6 must have been allocated digital value 6. If the output is allocated to another digital value, then status output still occurs (ON/OFF), but no speed stage output.

Heat meter:

The output variables of the heat meter function are **automatically bound into data records** according to the sequence in the function list (heat meters 1 and 2 in data record 1, heat meters 3 and 4 in data record 2). The values of the heat meter function cannot be defined as analogue values.

```
   DATALOGGING
-----------------
DIGITAL VALUE 1:
Source:  Output
 1 : Solar pump1
Status:       ON

DIGITAL VALUE 2:
Source:   Output
2 : Solar pump2
Status:       OFF
```

Digital values – in this sub-menu the total 26 digital parameters of the two data records are defined:
Data record 1: digital values 1 - 13
Data record 2: digital values 14 - 26

```
   DATALOGGING
------------------
ANALOGUE VALUE 1:
Source:   Input
 1 : T.collector
Status:    105.6 °C

ANALOGUE VALUE 2:
Source: HTG CIRC.1
 1 : Set flow temp.
Status:     58.2 °C
```

Analogue values – in this sub-menu the total 32 analogue parameters of the two data records are defined:
Data record 1: analogue values 1 - 16
Data record 2: analogue values 17 - 32

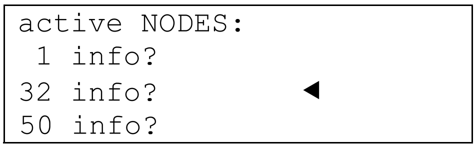
Functions of type "Heat meter" cannot be selected as a source. Their parameters are automatically bound into the two data records.

Delete All Def. – is only displayed at expert level. All settings (definitions) in data logging can be deleted by pressing the scroll wheel once. All logging values are in this respect set to *Source: User <unused>*.

CAUTION! Settings are immediately deleted without prior security prompt.

Important instruction re CAN data logging: One controller (version > E3.18) or the C.M.I. must be assigned node number 1 in the CAN network, so that the time stamp of this controller can be accepted by other devices.

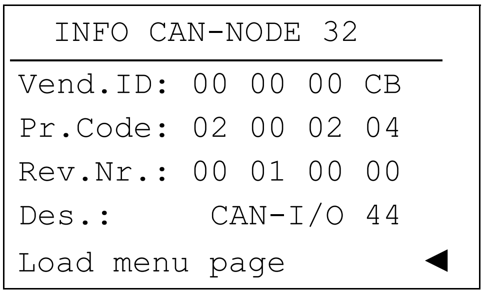
Network nodes



```
active NODES:
 1 info?
32 info?        ◄
50 info?
```

All network nodes in the network, with which the controller is linked, are listed here. This means that I/O modules and bus converters can be set from the controller. For CAN monitors, the room temperature (and room humidity for respective version) is displayed.
Setting CAN monitors and accessing other controllers is not possible.

Example CAN-I/O module, node 32:

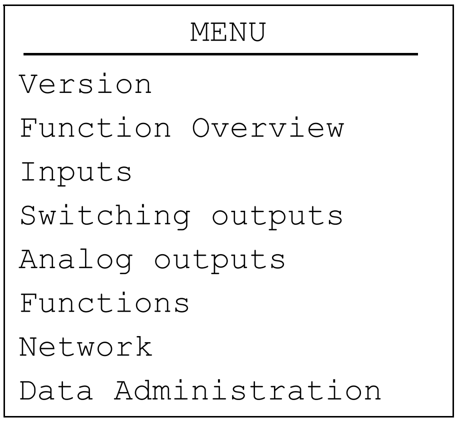


```
  INFO CAN-NODE 32
Vend.ID: 00 00 00 CB
Pr.Code: 02 00 02 04
Rev.Nr.: 00 01 00 00
Des.:      CAN-I/O 44
Load menu page      ◄
```

- selected node number

Vend.ID:	Manufacturer identification number (CB for Technische Alternative GmbH)
Pr.Code:	Product code of the selected node (here for a CAN-I/O module)
Rev.Nr.:	Revision number
Des.:	Node product description

These data are fixed values specified by Technische Alternative GmbH and cannot be changed.



```
         MENU
Version
Function Overview
Inputs
Switching outputs
Analog outputs
Functions
Network
Data Administration
```

Load menu page: This is used to access the menu level of the selected network node. The controller now serves as a display for this device.

MENU Data administration

In this menu the commands concerning the administration and protection of the function data as well as the update of the operating system are listed. The menu items for data transfer are valid only for Bootloader BL-NET. Data transfer via C.M.I. is described in the online manual of the C.M.I.

```
current func. data:
TA FACTORY SETTINGS
Status: original

Save settings as
factory settings
Load factory setting

Create backup copy
Load backup copy

Delete functions
Execute total reset

DATA <=> BOOTLOADER:
Upload data:
CTRLLR => BOOTLD.
Download data:
BOOTLD. => VTRLLR

OPER.SYSTEM<=BOOTLD.:
Download oper.system :
BOOTLD. => CTRLLR
```

Name of the current function data (TAPPS)
if function data have already been changed: modified

(Only appears if a backup copy exists!)

Data administration is valid only for Bootloader BL-NET.

Internal Data administration

Current function data:
TA FACTORY SETTING – The function data with this specification have been transferred to the control unit. **The TA factory setting can be loaded by simultaneously pressing the two input keys and scroll wheel when starting up the controller.**

Status: original – Since transmission, nothing has been changed in the function data.

Save settings as factory setting – The function data of two heating circuits including solar and charging pump system are put down in the unit as factory setting. If proprietary programming has been tested, it uses this command to replace the original setting as a factory setting.

Load factory settings - Call via the following commands in a security query YES / NO. **WARNING:** By this, the own function data will be deleted and replaced by the factory setting (ex works or an own one previously created). Nevertheless, a previously created backup copy (see following commands) is maintained.

Menu Data administration

Create backup copy – The function data can be saved in a backup copy. By this, it is possible to change the program or the parameters without losing the existing function data. If a backup copy has been created, than appears as a further menu option:

Load backup copy – The backup copy is loaded back instead of the current function data and thus overwrites all previous settings and programs – except the factory setting.

Delete functions – In order to start a new programming all function modules from the function list are deleted.

Execute total reset – This calling causes a complete loss of all entries (function data) except the data record of the factory setting and the backup copy. Besides the function modules it also clears the parameterizing of all in- and outputs.

Data exchange with the PC and/or Bootloader

DATA <=> BOOTLOADER:

Upload data – All function data are transferred via the CAN- Bus or the infrared interface to the Bootloader for the data protection in the PC. Choosing this command leads to the following indication:

```
CTRLLR => BOOTLD.
-----------------
DATA SOURCE: Ctrllr
Function data

DATA TARGET: BOOTLD.
Storage point:  1

REALLY START
DATA UPLOAD?   no

Activate CAN
IR interface?  yes
```

Transfer of function data or backup copy

One of 7 storage locations of the Bootloader is assigned.

Starting the upload by ***yes***
and pressing the start key of the Bootloader

Transfer is possible via cable as well as via IR

Notice: Depending on its operating system (update via internet is possible) and starting from boot sector version B1.01 the control unit has up to seven storage locations for function data.

DATA <=> BOOTLOADER:

Download data – By means of the Bootloader the protected function data in the PC are transmitted into the control unit via the CAN Bus or the infrared interface and thus the current programming is overwritten. The call contains similar commands as the upload; however it is possible to choose between several “**data targets**”.

BOOTLD. => CTRLLR -----------------	
DATA SOURCE: BOOTLD. Storage point: 1	Data are coming from storage location 1 of the Bootloader (from 7 possible storage points)
DATA TARGET: Ctrllr Function data overwrite ? yes	The function data are loaded into the main store
Factory settings overwrite ? no	The function data are not loaded into the factory settings
!!! CAUTION !!! ALL METER READINGS WILL BE LOST!	
REALLY START DATA DOWNLOAD? no	Starting the download by ***yes*** **and** pressing the start key of the Bootloader
Activate CAN IR interface? yes	Transfer is possible via cable as well as via IR

OPER. SYSTEM <= BOOTLD.:

Download Operating System: By its flash technology the unit offers the possibility to replace the own operating system (unit software) by a more current version (purchase from the download area of the address http://www.ta.co.at) by means of the Bootloader.
The application of a new operating system is only advisable, if it contains new **necessary** functions. An update of an operating system always represents a risk (comparable with the flashing of the PC Bios) and requires an examination of all function data as compatibility problems due to new function parts are to be expected!

Since the update of the operating system takes its time, it is advisable to carry out an update of the operating system **ONLY** via the cable wiring! After a missed update via IR the updating is permitted only via the cable wiring.

BOOTLD. => CTRLLR -----------------	
REALLY START OPERAT **SYSTEM DOWNL.?** no	Start the download by pressing ***yes*** **and** pressing the start key of the Bootloader
RECOMMENDATION: CABLE TIE	
Activate CAN IR interface? yes	Transfer via IR is possible but is not recommended

Description of the function modules

The following modules are available at present:

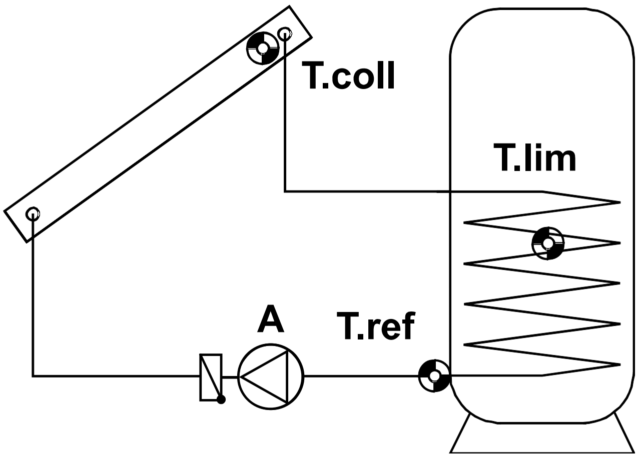
Solar control	Differential controller including various auxiliary functions
Solar priority	Priority assignment between multiple solar differential controllers
Start function	Start help for solar thermal systems
Cooling function	Cooling of an overheated solar cylinder during the night
Heating circuit controller	A mixer controller including heating circuit pump
Mixer regulation	Stabilizing the temperature variations by means of a mixer
Comparison	Comparison of two temperatures with one another (= thermostat
Charging pump	Differential and thermostat control of a charging pump
Heating demand	Burner demand by the buffer cylinder
DHW demand	DHW demand
Boiler cascade	Controls the burner demand of max. three boilers
DHW Circulation	Time and temperature control of a circulation pump
PID Control	Pump speed control
Analogue function	Searches for the lowest / highest temperature or the average
Profile function	Creates time-related (temperature) values (such as: for age-hardening of Floor pavement)
Logic function	AND-, OR-, holding function (Flip-Flop)
Time switch	Freely usable time switch clock
Timer function	Freely usable time interval function
Synchronization	Creates date-related switching signals
Heat meter	Energy evaluation
Meter	Freely usable interval or operating hours meter
Maintenance function	As a chimney sweep assistance and for exhaust measurement
Function control	Freely usable monitoring of sensors and differences
Menu Messages	System monitoring and output of error messages (The messages module is listed directly in the basic menu due to its properties.)

A maximum of 44 modules can be entered in the functions list.
When using memory-intensive functions (e.g. heating circuit controller), this number can be reduced.

Input variables which are absolutely necessary are highlighted in **bold** in the following function module description. The other input variables are optional.

Solar control

Basic diagram:



Input variables:	**Output variables:**
Enable solar circuit **Collector temperature** = T.coll **Reference temperature** = T.ref Limit temperature = T.lim	Status solar circuit Indication of the output A Status MAX limit reached = cylinder limit has been reached

Simple description of the function:

Release of the solar pump A, if the temperature in the collector T.coll is greater by a difference than the reference temperature T.ref, which is the (outlet) temperature of the cylinder. In addition, T.ref must not have reached its upper limit yet.

Special features:

- The system comes to a standstill when the collector exceeds the temperature of 140°C to prevent damage from steam. This means that the heat medium is no longer circulated, so that T.coll has a set upper limit (T.coll.MAX) including hysteresis.
- The differential temperature does not have a hysteresis that can be adjusted and is divided into a switch on and a switch of differential.
- If the cylinders have bare-tube heat exchangers, the reference temperature sensor should be screwed into the heat exchanger outlet using a t-shaped connector and an immersion sleeve (see the section on installing sensors in the instruction manual). If the surface of the collector is too large, the return flow temperature will increase too quickly, causing the limiter at T.ref to switch of the system to quickly. However, T.ref also cools down quickly in the standstill medium in the cold section of the cylinder. The pump will then start operating again, etc. To prevent this "cycling" and to prevent the cylinder from overheating if good layered cylinders are used, an **additional** optional upper limit has been defined in the module "solar control" for T.lim.
- The output variable “MAX limit reached:” shows that the upper limit has been reached (status: OFF/ON).
- If no additional limit sensor T.lim is used, it suffices to indicate *User* as the "source:" in the input variables.

Solar control

Entire menu view:

`DES:   SOLAR1`	
`FUNCTION STATUS:`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`COLLECTOR TEMP.:`	
`T.coll.ACT:  74.3 °C`	Current collector temperature
`T.Ccoll.MAX:   130 °C`	Pump is blocked when T.coll.MAX has been reached
`Hysteresis:    10  K`	Release at T.coll.MAX minus hysteresis
`REFERENCE TEMP:`	
`T.ref.ACT:   65.7 °C`	Current cylinder temperature (bottom/return)
`T.ref.MAX:     70 °C`	Cylinder limit
`Hysteresis:   3.0  K`	Release at T.ref.MAX minus hysteresis
`DIFFEREN. COLL-REF:`	
`DIFF.ON:      7.0  K`	Switch-on differential T.coll – T.ref
`DIFF.OFF:      4.0  K`	Switch-off differential T.coll – T.ref
`LIMIT TEMP.:`	
`T.lim.ACT:   54.0 °C`	Current temperature of the limit sensor
`T.lim.MAX:     70 °C`	Blocked by sensor
`Hysteresis:   3.0  K`	Release at T.lim.MAX minus hysteresis

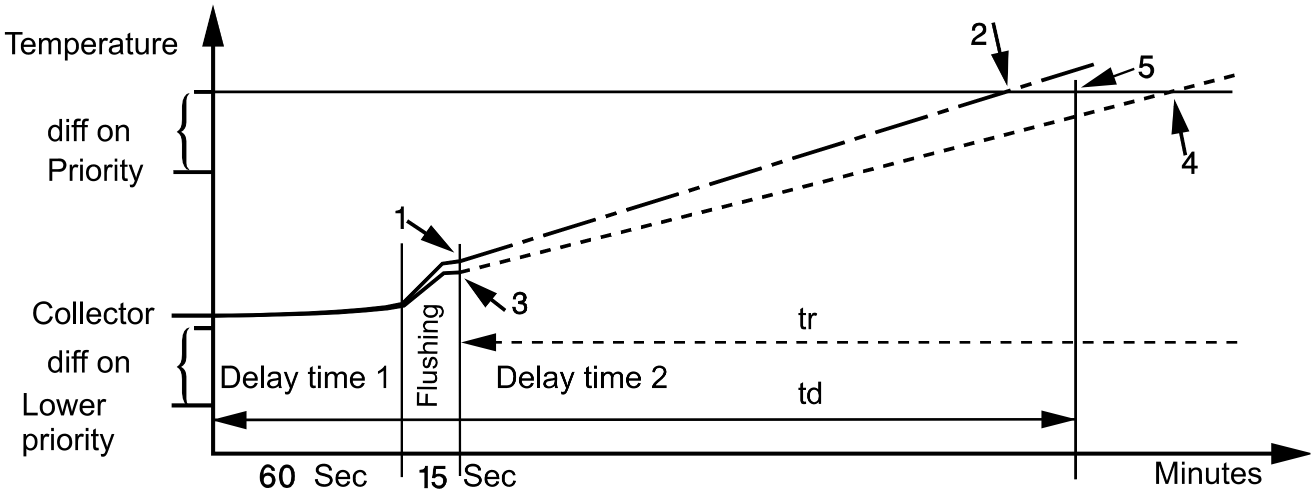
Because the module is used for all kinds of consumers, the labels "reference temperature" and "limit temperature" have been specified as generally applicable.
When using the third sensor as a limiter, the upper limit of reference sensor "T.ref.MAX" should be set high enough so that it does not affect operation.
The output variable “MAX limit reached:” has the status "ON" if the reference sensor's **or** the limit sensor's upper limit has been reached.

Solar priority

In solar heating systems used to charge more than one consumer (such as a cylinder, a buffer, and a basin), priorities must be set for the various circuits. There are basically two sets of rules used for a system of priority and lower priority assignment.

- **Absolute priority**: Only when a cylinder that has priority has reached its upper limit for temperature does the system switch to the next lower priority.
- **Relative priority**: The charge begins with the colder cylinder (because the collector will reach the differential here first), even if this cylinder does not have priority.

While the consumer that does not have priority is being charged, the device monitors the collector temperature. If the collector temperature once again reaches the switch-on differential for the consumer currently being charged while the pump is running, the priority timer is activated. If a radiation sensor is used, it must exceed at threshold value in lieu of the switch-on differential.



The priority timer switches the pump off for delay time 1 (60 sec). After flushing (1, 3), the computer then calculates the increasing collector temperature. It recognizes whether the set delay time 2 td will suffice to heat up the collector to the priority temperature (5). If not, the system waits for priority temperature to be reached before switching (case 2). If the computer determines that the increase will not suffice within the delay time 2 td (case 4), it interrupts the procedure and disables the priority timer until runtime tr has expired.

If runtime = 0, the lower priority is allowed once the maximum threshold of the current priority has been reached. In other words, the system is switched to absolute priority assignment mode.

Solar priority

Input variables:	Output variables:
Enable solar priority Solar radiation = Radiation sensor **Involved functions** = Entry of all solar functions in the function list	Status flushing process Indication of the output for the flushing

Special features:

- In this function blocks, the "involved functions" are not individual values, but rather entire function module input variables.
- The program automatically looks for all of the values needed in the function modules involved and also automatically blocks the modules involved that are lower in the hierarchy.

Entire menu view:

(Assumption: Six solar functions are entered in the function list)

`DES: SOL. PRI.`	
`FUNCTION STATUS:`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`SOLAR1   1`	SOLAR 1 is the highest priority
`SOLAR2   2`	SOLAR 2 is the second priority
`SOLAR3   3`	SOLAR 3 is the last priority
`SOLAR4   1`	SOLAR 4 is the highest priority
`SOLAR5   2`	SOLAR 5 is the second priority
`SOLAR6   3`	SOLAR 6 is the last priority
`SEC. TIMER:`	
`from prim. stage 2`	SOLAR 1 and 4 are loaded as "absolute" without time element
`Solar rad:  488 W/m²`	Current solar radiation (does not apply without radiation sensor)
`Thres val:  200 W/m²`	Activation threshold for timer (does not apply without radiation sensor)
`Runtime:     20 Min`	Runtime for the consumer of lower priority until timer starts
`delay time:   5 Min`	The collector must reach the temperature of the priority cylinder within five minutes; otherwise, the cylinder of lower priority will be charged

As this example makes clear, it is also possible to assign the same priority. Generally, this approach is only useful if the system has multiple collector fields. The priorities in the example correspond to the system with two collector fields charging three consumers (such as Solar 1 = collector 1 for cylinder 1 and Solar 2 = collector 1 for cylinder 2, etc.).
SOLAR 1 and SOLAR 4 are admissible first as the lower priority time element is only active starting at priority level 2 until the consumer has reached its maximum temperature (absolute). Only then does the priority treatment shift to the other solar functions via the priority timer (relative).

Start function

Simple description of the function:

Sometimes, the collector sensor is not immersed in the heated heat medium quickly enough. In other words, the system goes into operation too late. There may not always be enough gravity force if the collector fields are completely horizontal, if the connections between the absorber strips meander, and especially if **vacuum tubes** are used with forced circulation.
This module puts the solar pump into operation in preset intervals, thus transporting the content of the collector to the sensor. To reduce energy losses, this interval operation is only launched within a certain time window and only if the solar radiation reaches a certain level (determined using **GBS** radiation sensor - special accessory) **or** with constant monitoring of the collector temperature. Without a radiation sensor, the computer first attempts to determine the current weather conditions by constantly measuring the collector temperature. By this, it finds the right time for a brief rinsing interval so that the temperature for normal operation can be maintained. Each collector field must have its own start function.

Input variables:	**Output variables:**
Enable start function Solar radiation = Radiation sensor Reference temperature = Input of the collector sensor **Involved functions** = Entry of all solar functions for the collector field in the function list	Status flushing process Indication of the output for the flushing

Entire menu view:

```
DES: SOL START
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:

Activation time:
07:00   -  20:00 h        Time window for permission of start function
Runtime:      15 Sec      Rinsing time
Interval:      20 Min     Maximum delay between rinses
Activ.grad.:        20    or radiation threshold - see description below
Start attempts:   13      Sum of attempted starts today
unsuccessful:     11      Number of unsuccessful starts
since last run:    6      Number of attempts since the system last operated correctly
```

If a radiation sensor is used, instead of "Activ.grad." the computer displays the desired radiation threshold above which the start function is to be active. In many cases, you can do without this sensor altogether. Then, a mean value is calculated from the collector temperature, with special attention paid to the lowest temperatures reached. The start function is enabled when the collector temperature is warmer than the mean value by the difference of the activation gradient. A lower activation gradient therefore leads to an earlier start attempt, a higher gradient to later attempts.
If you need more than ten tries to start the system, the activation gradient must be increased and must be reduced, if the number of attempts is below four.
If the activation gradient is set to zero, then only the activation or interval time applies without consideration of the temperature curve at the collector sensor.

Cooling function

Simple description of the function:

Solar heating systems with partial solar heating produce more warm water in the summer than can be used. This function can be used at night to remove some of the excess energy from the lower section of the cylinder back into the collector when a critical temperature has been reached in the buffer cylinder by regulating the speed of circulation. During the day, the system will then not have to be brought to a standstill as often due to excess temperature.

Input variable:	**Output variable:**
Enable cooling function **Reference temperature** = Measuring point that sets off the function Maximum temperature reference = Temperature that sets off the function	Correcting variable = Speed stage RPM output Indication of the RPM control-output Status switching output, Indication of the switching output

Special features:

- Generally, the maximum value for the thermostat threshold can be set. This value is defined as an input variable to make it easy to link it. You only need to enter "source" *User* as the setting value. The user can see it as a common function parameter when it appears in the menu of the function.
- In addition to the output indicated with a set speed, the function module also provides a switchable output. This output can be used to block other functions during the cooling phase.
- The maximum set value does not have a hysteresis that can be set, but rather a switch-on and a switch-off differential.

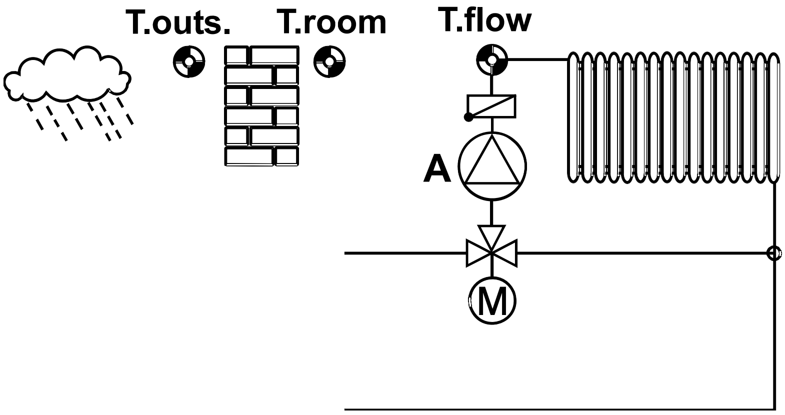
Entire menu view:

DES: COOL FUNC	
FUNCTION STATUS:	
INPUT VARIABLE:	
OUTPUT VARIABLE:	
TIME WINDOW:	
00:00 - 06:00 h	Time window for active cooling
REFERENCE TEMP.:	
T.ref.ACT: 65.7 °C	Current cylinder temperature (bottom/return)
T.ref.MAX: 90 °C	Cylinder limit
DIFF.ON: 5.0 K	Cooling active between 12 midnight and 6 a.m. above 95°C
DIFF.OFF: 0.0 K	The cooling function is switched off once the temperature drops below 90°C
CORRECT.VAR.: 15	The pump runs at speed stage 15

Tests have shown that cooling is also sufficient at the lowest speed stages. We therefore recommend that you use a speed stage just above circulation standstill. For instance, at the stage 5 the pump only consumes 10% as much energy.

Heating circuit controller

Basic diagram:



Input variable:	Output variable:
Enable heating circuit controller Enable pump Enable mixer Room temperature = T.room Flow temperature = T.flow Outdoor temperature = T.outs. External Switch = Switching to frost protection mode (status: ON) / Operation according to the setting of the unit (status: OFF)	Set flow temp. = Temperature of the flow calculated by the control unit T.flow SET Effective set room temperature = Valid indoor temperature according to time program T.room EFF Status of the heating circuit pump, indication of the output Status of the mixer, indication of the output Status of the maintenance mode Status of the frost protection mode

Simple description of the function:

Mixer controller based on outdoor and indoor temperature with consideration of the heating and lowering temperature specified in the switching times. The heating pump can be enabled in various parameters.

Special features:

- The input variable "**EXTERNAL SWITCH**" can be used to switch a remote switch between frost protection mode and normal operation according to the device settings. Furthermore, the external setting of an operating mode is possible via a dimensionless number (64 to 67).
- In addition to the pump and the mixer, this function also provides a calculated flow temperature (T.flow SET) and the status of maintenance and frost protection modes, e.g. for messages.
- Another output variable is the effective room temperature (T.room EFF), which is influenced by the timer and other functions. A heating control unit **without a mixer** can thus be set up with a downstream speed control module.
- Under "**OPERATION**," special functions such as PARTY and HOLIDAY, etc. are callable.
- A **derivating time** which can be selected and is based on the outdoor temperature also affects the switching between lowering and heating modes.
- Four criteria can be selected to switch off the pump.
- If, when the function is first called, or using "CHANGE FUNCTION" **yes** is specified to the time program "with set value?", then each time window receives its own adjustable room temperature, which replaces the value "T.room STD".
- If a room sensor is indicated in the input variables but the sensor is short-circuited, the heating circuit controller will operate as though no room sensor were indicated in the parameters.
- The mixer runtime is reloaded when the mixer output is in manual mode, is triggered by a message (dominant ON or OFF), or a trigger switches from OPEN to CLOSED. If mixer enable is OFF the mixer remains stationary in the last position.

Heating circuit controller

Basic menu view:

DES: HEAT.CIR.1	
FUNCTION STATUS:	
INPUT VARIABLE:	
OUTPUT VARIABLE:	
OPERAT.: RS	The heater is controlled by a room sensor RAS
STANDARD	and is currently running in heating mode (*STANDARD*)
ROOM TEMPERATURE:	
T.room ACT: 20.7 °C	Current room temperature
T.roomSETBACK:16 °C	Desired room temperature doing lowering time
T.room STD: 20 °C	Desired room temperature doing heating time
TIME PRG:	Submenu for heating times (see **Time programs**)
Deriv. time: 30 Min	The heating time begins 30 min earlier if the outdoor temperature is below -10°C
T.room EFF: 20°C	Current desired room temp. = 20°C (current heating operation)
FLOW TEMPERATURE:	
T.flow ACT 58.4 °C	Current flow temperature
T.flow SET: 58.2 °C	Calculated flow temperature
HEAT CURVE:	Submenu for the calculation of the flow temperature
OUTDOOR TEMPERATURE:	
T.outs.ACT: 3.6 °C	Current outdoor temperature
AVERAGE:	Settings for calculation of outdoor temperature for the calculation of flow temperature and to switch off the pump
SHUTDOWN CONDITION:	Submenu to switch off the pump and conditions for mixer
FROST PROT:	Submenu where the outdoor temperature is indicated to keep the room at a certain minimum temperature

OPERATION

***TIME*/AUTO** may also be found under "**OPERAT**:" if "unused" has been entered as an input variable for the room sensor. Furthermore, you can also switch to the following heating functions here regardless of whether a room sensor is being used:

- ***STANDBY*** Switches the control to standby (frost protection remains activated)
- ***SETBACK*** The controller is switched to manual mode - setback
- ***STANDARD*** The controller is switched to manual mode - heating (standard)
- ***HOLIDAY*** The controller takes the heating times for Saturday starting today and the times of Sunday for the last day indicated
- ***BANK HOL.*** Up to date xx, 00:00 hours, the controller will only work in lowering mode
- ***PARTY*** The heating mode remains in operation until the indicated time xx

For the modes of operation ***HOLIDAY***, ***BANK HOL.*** and ***PARTY***, the controller switches back to automatic mode after the time indicated has expired.
The room sensor is not evaluated in simulation mode; therefore there is **no** "**RS**" operating display.

Further possible displays under "OPERATION":

FROST PROT	The frost protection function is activated. The activation conditions are described in the section "Frost protection".
EXT/STANDBY	The input variable „External Switch“ is a digital "**ON**" signal.
MAINTENANCE	The maintenance function is active (see function "Maintenance"). The flow temperature is controlled to match the setting T.flow MAX set in the menu HEAT CURVE. The function module remains active for three minutes after maintenance mode has been switched off.
FAULT	If the line to the external sensor (measuring value > 100°C) is interrupted, the heating circuit is switched off. In the worst-case scenario, the system may be damaged by frost. To prevent such damage, the heating circuit is operated according to a set outdoor temperature of 0°C if the outdoor temperatures are clearly too high, and ***FAULT*** is displayed under "OPERAT:".

Status of the heating circuit pump and the mixer

relative to the operation mode and releases:

Operation mode		Enable heating circuit	Enable pump	Enable mixer		Status: pump	Status: mixer
X		OFF	x	x		OFF	OFF
Maintenance		x	x	x		ON	AUTO[1]
Standby, external standby		x	x	x		OFF	OFF
Frost protection, fault		ON	x	ON		ON	AUTO
				OFF		ON	OFF
Time / Auto, standard, setback party, bank holiday, holiday		ON	OFF	OFF		OFF	OFF
			ON	OFF		AUTO	OFF
			OFF	ON		OFF	OFF[2]
			ON	ON		AUTO	AUTO
RS		ON	as with Standby, Time/Auto, Standard, Setback,				

x… Status and operation mode do not matter

[1]… In this case, AUTO means that the settings for T.flow MAX are used in the menu HEAT CURVE.

[2]… OFF does not apply if in "SHUTDOWN CONDITION" the setting "control" is set under "if heating circ. OFF => MIXER:"

EXTERNAL SWITCH

The “EXTERNAL SWITCH” input variable also accepts analogue values on external operating mode switching:

Value (dimensionless):	Operating mode:
64	Standby/frost protection
65	Time/Auto
66	Standard
67	Setback
127	Switch back to internal operation

These analogue values can originate from another function or also come from the GSM module from the Bootloader as a network input. Values **64 - 67** are dominant, i.e. it is not possible to set any other controller operating mode, as long as the value "external switch" exists.

NB: If nevertheless at attempt is made to set another operating mode, then the controller display jumps back from the operating mode specified from the "External switch" and remains in the original operating mode. However, the controller "notices" this change and adopts this operating mode after resetting using the value 127 at the "External switch". If during this time an operating mode **other** than "**RS**" is selected, then this operating mode cannot be changed on the **RS**, rather only on the controller, on the CAN monitor or via the browser. As soon as the value at the "External switch" is 127, manual alteration of the operating mode is possible at any time.

Important note: The external switch must never be linked with a temperature sensor because the controller can be damaged otherwise.

TIME PROGRAM

Parameterizing of the time program is described in the chapter "Menu functions".
The room temperature T.room STD or the set value applies in the time window. T.room SETBACK applies outside the time window. The switchover causes a corresponding parallel shift of the heating curve and therefore a change to the set flow temperature, T.flow SET.
"CHANGE FUNCTION" is used to change the number of required time programs and the number of windows per program and to specify the use of a special set value per time window:

```
Scope of TIME PROG:
Number of Prog.: 3
No. of windows:  3
with set value ? no



CHANGE?         no
```

Maximum 5 time programs can be set
Maximum 3 time windows per time program can be set
The input "*no”* means that the same set value T.room STD is used for all time windows.
The input "*yes*" makes it possible to allocate a special set value instead of T.room STD.
Activation of the change with “CHANGE? ***yes***“

Derivation TIME

Depending on the outdoor temperature, fixed heating times can cause the system to switch on or off too quickly. The rate time can shift the switch point relative to the outdoor temperature. This entry refers to an outdoor temperature of -10°C and is 0 at plus 20°C. For example, if the rate time is 30 minutes and the outdoor temperature is 0°C, the system will switch on 20 minutes earlier under normal operation.

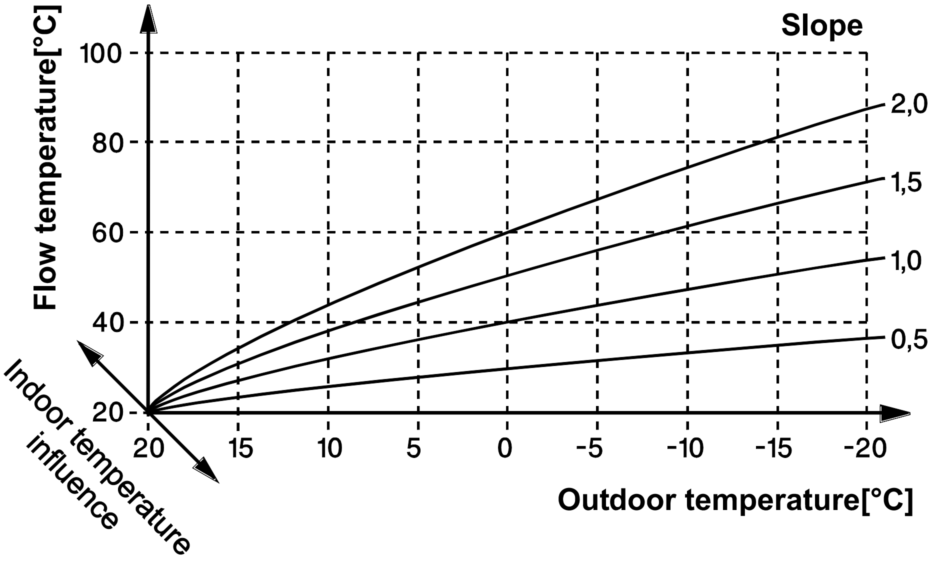
HEAT CURVE

The flow temperature is usually calculated from the outdoor temperature and the heating curve. The heating curve is calculated based on a room set temperature of +20°C and is subject to an appropriate parallel offset for other room set temperatures. An exception is the fixed value control. The flow is set in setback mode to the listed temperature of +10°C and in standard mode to that of -20°C.
The module allows you to set parameters for the heating curve in two ways:

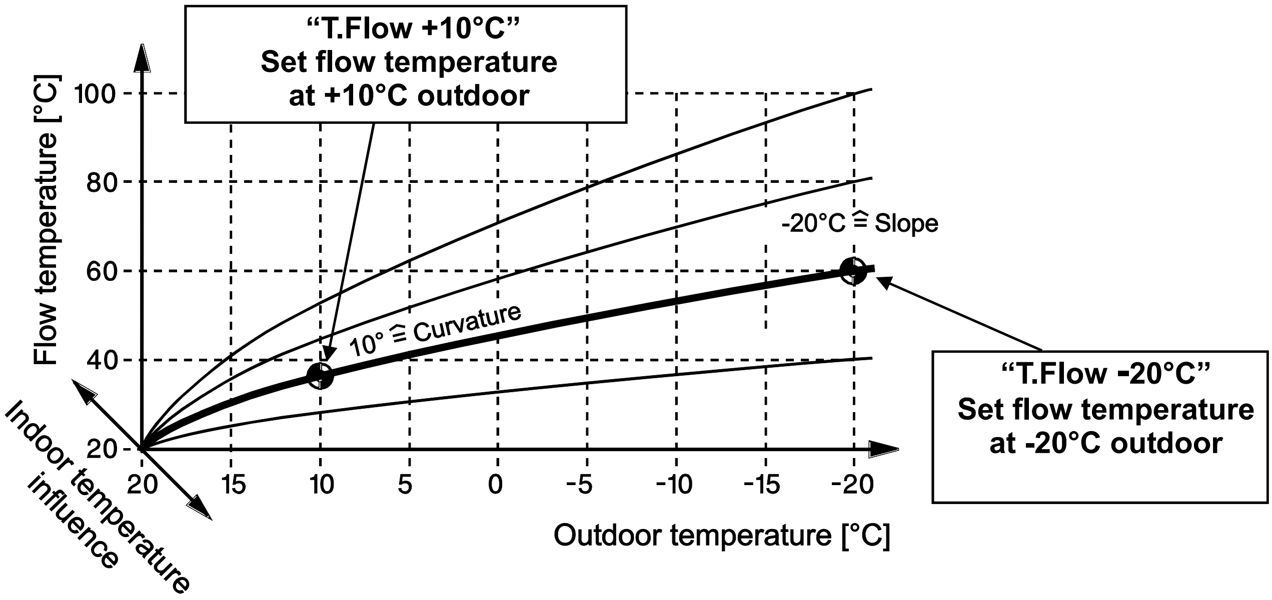
- With reference to the slope as is common in many heating control units.
- As a product of the relation between the outdoor temperature (at +10°C and -20°C) and the flow temperature. Here, another reference point is set at plus 20°C outdoor temperature = plus 20° flow temperature.

In both of these methods, the influence of the outdoor temperature on the flow temperature is **not linear**. Via "***Slope***" the curvature is set according to the standards. Via "***Temperature***" a curvature of the heating characteristic is created with the desired flow temperature at 10°C in order to take into account the different heat emissions of various heating systems.

Heat Curve "Slope"



Heat Curve "Temperature"



The following entries are found in this submenu “Heat curve”:

HTG CIRC.1 MODE: CONTROL : Outs.temp. ***or*** Fixed val.	Control based on outdoor temperature and heating curve The flow is set in setback mode to the indicated temperature at +10°C and in standard mode to the one at -20°C.
HEAT CURVE: Temp. ***or*** Slope	Heating curve via temperature points +10°C and -20°C Heating curve via input of the slope (0.05-2.50)
Room inf.: 0 %	The room temperature is taken into consideration for the calculation of the flow at xx%, setting range 0 – 90% The room influence is also active in fixed value mode.
Start excess: 0 %	The previous switch-off time of the heating circuit pump leads to a (decreasing over time) excessive increase in flow temperature. (maximum up to T.flow MAX). Setting range 0 – 20% *) for detailed explanation see below
T.flow+10°C: 35 °C T.flow-20°C: 60 °C ***or:*** Slope 0.60	Desired flow temp. at +10°C outdoor temp. (heating curve) Desired flow temp. at -20°C outdoor temp. (heating curve) Specification of the slope (for selection Heat Curve: Slope)
T.flow MAX: 65 °C T.flow MIN: 20 °C	The flow must not exceed this value The flow must not drop below this value

*) Start excess (“Increasing on Start”)

The precise formula for the start excess is:

T.flow SET Start excess = T.flow SET + T.flow SET* (Start excess / 100) * (Meter/ 30)

The meter is increased by 1 every 20 minutes with a **switched-off** heating circuit pump, with a **switched-on** heating circuit pump, it is lowered by 1 every minute down to 0.

The maximum meter value is 255. It is therefore reached after 85 hours switch-off time (= 255/3 hours or approx. 3.5 days). The maximum run-down time is 4.25 hours (= 255 minutes). The set excessive increase in % is effective after a switch-off time of 10 hours (= 30 x 20 minutes).

Example: T.flow SET=40°C, Start excess = 10%, Switch-off time 8 hours

The excess temperature starts at +3.2 K (above the set value) and falls uniformly to zero within 24 minutes.

Protection of heat-sensitive systems parts:

Heat-sensitive systems parts (e.g. plastic ducts) must be equipped with additional protecting devices (e.g. thermal temperature limit for floor heating) which prevent overheating in the event of failure of the controller or of another system component.

AVERAGE of outdoor temperature

Sometimes, fluctuations in outdoor temperature may even not be desirable in calculation of flow temperature or when determining the disconnection of the heating pump. Therefore, a separate calculation of the mean is available for the outdoor temperature to calculate the heating curve and the disconnection of the pump. The following entries are found in this submenu:

HTG CIRC.1	
for flow control: Aver time: 10 min	The average outdoor temperature is calculated for 10 minutes for the flow
T.outs.av.c: 13.6 °C	The current outdoor temperature average is 13.6°C
for shutdown: Aver time: 30 min	The average outdoor temperature is calculated for 30 minutes for switching off
T.ou.av.off: 13.8 °C	The current outdoor temperature average is 13.8°C

SHUTDOWN CONDITIONS and mixer behaviour

The controller allows the following switch-off conditions for the heating circuit pump:

if T.room ACT > SET ? no Hysteresis: 1.0 K	when the desired room temperature has been reached
If T.flow SET < MIN ? yes Hysteresis: 2.0 K	when the flow temperature falls below the lower limit T.FlowMIN
If T.outs. Av.off> MAX ? no T.outs.MAX: 20 °C Hysteresis: 2.0 K	when the average outdoor temperature T.Out.MAX exceeds a set value in the heating or lowering mode
If setback mode and T.outside ACT > MIN. ? no T.outs.MIN: 5 °C Hysteresis: 2.0 K	when the outdoor temperature T.Out exceeds a set value in the lowering mode
If T.flow ACT > MAX ? no	when the flow temperature is larger than T.FlowMAX (settings in the heat curve) plus a fixed hysteresis of 3K, reactivation if T.FlowACT. < T.FlowMAX
If heating circ. OFF MIXER: close	**Mixer behaviour:** In addition, in this menu you can determine how the mixer is to respond once the pump has been switched off (*close, open, unchanged, (continue to) control*). If mixer enable is "OFF" the mixer remains stationary in the last position it took up (Status Mixer: OFF).

The hystereses of the switch-off conditions generally have an **increasing** effect.

As both the outdoor temperature and the room temperature are taken into consideration when the set flow temperature is calculated (provided a sensor is used), the best method is to switch off under the limit T.flow MIN.

Heating circuit controller

FROST PROTECTION

This function is only active in the standby mode or when the input variable "EXTERNAL SWITCH" is active, and even then only if the module has been partially blocked by the input variable ENABLE PUMP or a switch-off condition would block the heating circuit pump**. However, if the function is blocked via Enable heating circuit, there is no frost protection mode.**

If frost protection is activated, the set flow temperature is maintained at least at T.flow MIN (setting in the sub-menu heating curve), until the temperature, which triggered the frost protection function, increases by 2 K above the frost protection limit. This submenu has the following entries:

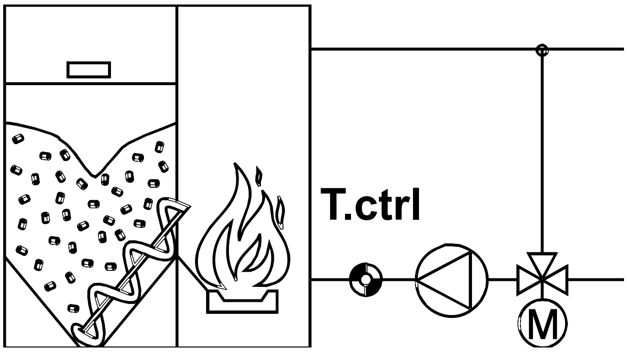
```
Activation if
T.outs.av.c  <   5°C
T.room FROST:    5°C
```

Frost protection function is active below +5°C (outdoor) and keeps the room at a temperature of 5°C

Heating circuit condition	**Frost protection function**
Operating mode STANDBY Set to RAS/RASPT	without activated frost protection function: T.flow SET set to +5°C, mode display: STANDBY Activation of the frost protection function: If T.outs.ACT < T.outs.av.c, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)
Operating mode STANDBY Controller setting	without activated frost protection function: T.flow SET set to +5°C, mode display: STANDBY Activation of the frost protection function: If T.outs.ACT < T.outs.av.c **or** (if RAS available) T.room ACT < T.room FROST, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)
Switching via digital "ON" at the "external switch" to EXT/STANDBY	without activated frost protection function: T.flow SET set to +5°C, mode display: EXT/STANDBY Activation of the frost protection function: If T.outs.ACT < T.outs.av.c **or** (if RAS available) T.room ACT < T.room FROST, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)
Switching via analogue 64 at "external switch" to STANDBY	without activated frost protection function: T.flow SET set to +5°C, mode display: STANDBY Activation of the frost protection function: If T.outs.ACT < T.outs.av.c **or** (if RAS available) T.room ACT < T.room FROST, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)
Enable pump off	without activated frost protection function: T.flow SET according to the setting heat curve, Mode display: selected operating mode Activation of the frost protection function: If RAS available: If T.room ACT < T.room FROST the pump is switched on, **independent of the outside temperature**, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)
Switching off of the pump via a switch-off condition	without activated frost protection function: T.flow SET set to +5°C, mode display: STANDBY Activation of the frost protection function: If RAS available: If T.room ACT < T.Room FROST the pump is switched on, **independent of the outside temperature**, then T.flow SET ≥ T.flow MIN (mode display: FROST PROT.)

Mixer control

Basic diagram:



Input variables:	Output variables:
Enable Mixer **Control Temperature** = T.ctrl - indication of a sensor Set Value = Control to this value (+diff)	Control Temperature = T.ctrlEFF, set temperature calculated by the controller from the control temperature and the difference Status Mixer = M, indication of the outputs

Simple description of the function:

This function allows a mixer to be constantly controlled for a set value.

Special features:

- Generally, the set value will be an adjustable one. This value has been defined as an input variable to make it easy to link it. It will then appear in the menu for the function that the user can see as a common function parameter if *User* is indicated as the "source."
- Another differential can also be used to set the set value.
- In addition to the mixer output, the function also sets the overall set value as the effective control temperature (T.ctrlEFF) for the output variable.
- As the module can only be switched via its enable, the mixer setting is preset to "enable OFF."
- In addition to *standard*, the mixer mode can also be set to *inverse* (such as a cooling function for wall heaters, etc.). Under *inverse*, the mixer opens as the temperature increases.
- The mixer running time (20 minutes) is reloaded, if the mixer output is in manual mode, triggered by a message (dominant ON or OFF), or the triggering direction changes from OPEN to CLOSED or vice versa or the enable is switched from OFF to ON.

Entire menu view:

`DES: MIX CTRL`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`MODE:    std`	The mixer closes as the temperature increases
`CONTROL TEMP.:`	
`T.ctrl ACT:   30.4 °C`	Current control temperature
`T.ctrl SET:     30 °C`	Preset control temperature
`Differen.:   0.0  K`	Additional control differential for the set value
`if ENABLE = Off`	Mixer behaviour with enable = off:
`MIXER: unchanged`	*close, open, unchanged*

Comparison

(Thermostat/differential function)

Simple description of the function:

Two values, Va and Vb + differential, are compared to produce the two output variables Va > Vb and Va < Vb.

Input variables:	**Output variables:**
Enable comparison	Status Va > Vb + diff = Value a is greater than value b
Comparative Value a = First comparison value	Status Va < Vb + diff = Value a is lower as value b, indication of the output
Comparative Value b = Second comparison value	

Special features:

- Only one sensor input / one output variable for another function is admissible for the value a. Value b can also be a flexible (temperature) value. If so, *User* has to be indicated as the "*source*." The user can see the value b as a common function parameter when it appears in the menu of the function.
- Generally this function applies to a thermostat. Use the "**function variable**" (FUNC.VAR.) to make comparisons of any figures. Possible selections: temperature, dimensionless, flow rate, output, heat amKWh, heat amMWh, number of impulses, time, solar radiation, relative humidity, wind speed and pressure.
- The comparison difference consists in the differential between the switch-on and switch-off.
- Both Va > Vb and Va < Vb can be used as output variables. When comparing a temperature sensor with a threshold value (value b is entered as "user" under the input variables), this basically functions as a mechanical thermostat with mutual contact (Va > Vb = make contact and Va < Vb = break contact).
- If sensors have been assigned to both values, the result is a simple differential function.
- If enable is set to "OFF", then **both** output variables are set to "OFF".

Entire menu view:

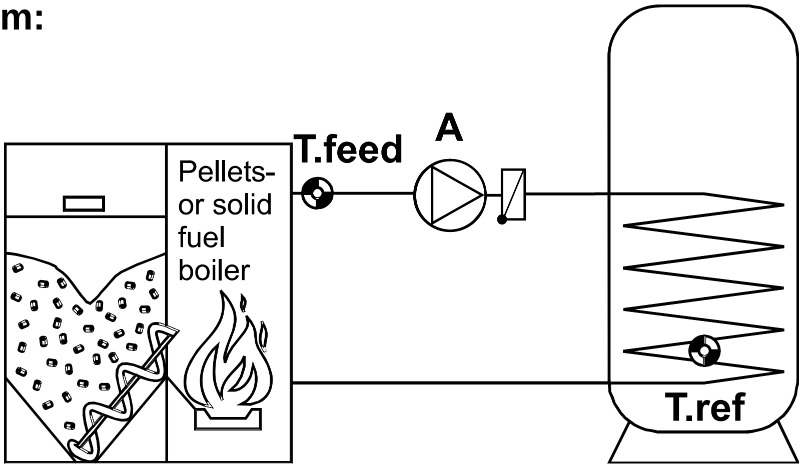
```
DES:   COMP.1
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:

FUNC.VAR: Temp.            Comparison of two temperatures

VALUEa:       39.1 °C
VALUEb:       44.3 °C

DIFF.ON:       5.0  K      Switch on if value a exceeds 49.3°C (44.3 + 5.0)
DIFF.OFF:      2.0  K      Switch on if value a falls below 46.3°C (44.3 + 2.0)
```

WARNING: The output status of the second output variable is the inverse of the first output variable Va > Vb + diff. The designation Va < Vb + diff for the second output variable is thus inaccurate. This representation was selected because the display cannot indicate the inverse symbol.

If to sensors are compared, the connection of the warmer sensor (generator) to Va is recommended. If the link between values a and b are adversely linked in the input variables, switching will then be based on a negative differential.

Charging pump

Basic diagram:



Input variables:	**Output variables:**
Enable charging pump **Feed temperature** = T.feed **Reference temperature** = T.ref Minimum temp. Feed = Min. threshold at T.feed Maximum temp. ref. = Max. threshold at T.ref	Status of the charging pump = A Indication of the output A

Simple description of the function:

Release of charging pump A if the temperature on the boiler (feed temperature T.feed) is higher than the minimum temperature and one difference higher than the reference temperature T.ref.. In addition, T.ref must not have reached its upper limit yet.

Special features:

- In most applications, the minimum threshold for T.feed and the maximum threshold for T.ref can be set. These two thresholds are defined as input variables to make it easy to link them.
- The example used is a link with the burner demand for DHW preparation. The function *DHW demand* provides the set temperature for the cylinder as an output variable. The set temperature can therefore be used simultaneously as the maximum temperature for the charging pump function.
- If the two input variables can be set, all you need to enter is *User* as the "source." The user can see it as a common function parameter when it appears in the menu of the function.
- Neither thermostat threshold has hysteresis, but rather a switch-on/off differential to the adjustable threshold value.
- **Example**: Minimum threshold = 60°C
  DIFF.ON = 5.0 K
  DIFF.OFF = 1.0 K

  In other words, if the temperature T.feed exceeds 65°C (= 60°C + 5°K), the output is activated, whereas it is switched off if the temperature drops below 61°C (= 60°C + 1°k).

Charging pump

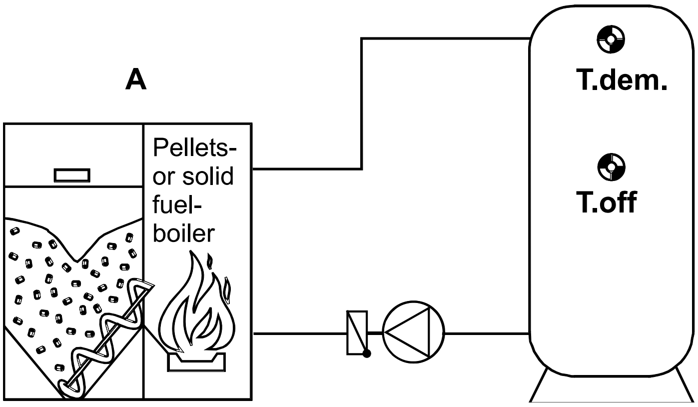
Entire menu view:

```
DES:   LD PUMP 1
FUNCTION STATUS:
INPUT VARIABLE:
OUTPUT VARIABLE:

FEED TEMPERATURE:
T.feed ACT:  74.3 °C      Current temperature of the "energy generator"
T.feed MIN:    60 °C      Basic switch-on threshold at sensor T.feed
DIFF.ON:      5.0  K      Switch-on differential to T.feed MIN (here, 65°C)
DIFF.OFF:     0.0  K      Switch-off differential to T.feed MIN (here, 60°C)

REFERENCE TEMP.:
T.ref.ACT:   65.7 °C      Current cylinder temperature
T.ref.MAX:     90 °C      Cylinder limit
DIFF.ON:      1.0  K      Switch-on differential to T.ref.MAX (here, 91°C)
DIFF.OFF:     5.0  K      Switch-off differential to T.ref.MAX (here, 95°C)

DIFFERENCE FEED-REF:
DIFF.ON:      6.0  K      Switch-on differential FEED - REF
DIFF.OFF:     3.0  K      Switch-off differential FEED - REF
```

While the minimal feeder temperature has to have constantly DIFF.ON > DIFF.OFF, DIFF.ON must always be < DIFF.OFF for the maximal reference temperature.

Heating demand

Basic diagram:



Input variables:	Output variables:
Enable heating demand **Demand temperature** = T.dem. Shutdown Temperature = T.off Set value demand = Min. threshold at T.dem Set value shutdown = Max. threshold at T.off	Status demand Indication of the output A (= Enable burner)

Simple description of the function:

Release of burner A if temperature in the buffer cylinder at the top (demand temperature T.dem.) falls below the "set value demand" (corresponds to a minimum threshold) and switch-off if the temperature in the bottom of the cylinder (shutdown temperature T.off) exceeds the "set value shutdown" (corresponds to a maximum threshold).

Special features:

- ♦ Generally, the values for demands and switch-off as thermostat threshold can be set. Here, both thresholds are defined as input variables. If the values can be set, you only have to enter *User* as the "source" in order to have it here for the user as a function parameter in the menu of the function.
- ♦ The system is switched on / off via separate threshold values and sensors, so that the two thresholds do not have any hysteresis. Instead, both thresholds have a differential to the value that can be cumulated.
  Switch-on threshold = Set value demand +DIFF.ON at sensor T.dem.
  Switch-off threshold = Set value shutdown +DIFF.OFF at sensor T.off
- ♦ The method of burner demand via a sensor and switch-off via another one is called "holding circuit" If a switch function has separate switch on/off thresholds **for just one sensor**, the input variable "SHUTDOWN TEMPERATURE" has to be set to *User / unused*. If the boiler sensor is entered instead of the cylinder sensor, the result is a sliding boiler operation. The "DEMAND TEMPERATURE" then has a differential for switch on/**off** in addition to the threshold value.
  Switch-on threshold = Set value demand + DIFF.ON
  Switch-off threshold = set value demand + DIFF.OFF
- ♦ A minimum temperature can be set via "**Low end temperature**" T.dem.MIN.
  Switch-on threshold = T.dem.MIN +DIFF.ON at sensor T.dem.
  Switch-off threshold = T.dem.MIN +DIFF.OFF at sensor T.off
  The base temperature is only effective, if the set value demand > 5°C. A value > 30°C is only useful if the function is used for sliding boiler operation. In this case, the switch off and on thresholds relate to the sensor T.dem..

Heating demand

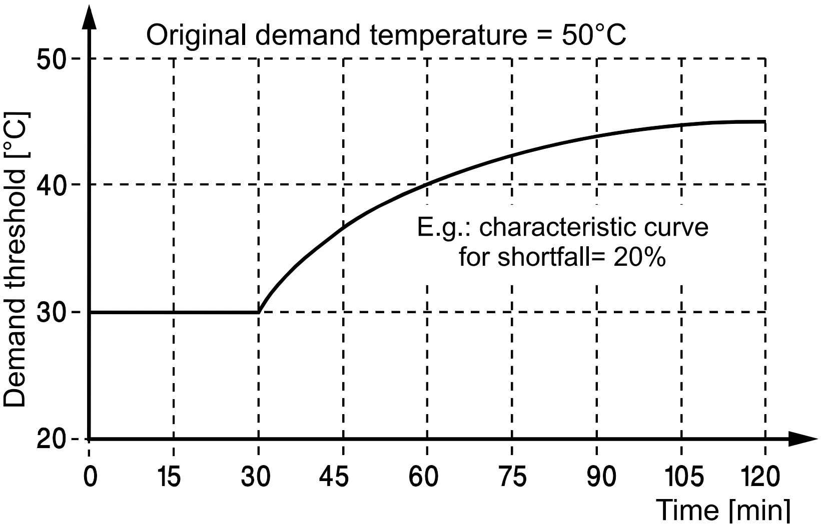
Eco mode:

Eco operation is related by "shortfall" to a particular time span. The shortfall stage always applies for 60 minutes. For a demand temperature of 50°C shortfall of 20% has the following effect: demand after 30 minutes below 30°C or below 40°C (= 20%) after an hour or below 45°C after 2 hours. The threshold value remains the same below 30 minutes.

Formula: dT * dt = Shortfall * set value demand temperature = constant

Example:
Demand temp. = 50°C
Shortfall = 20%

=> 20% of 50°C = 10K
dt= 30min => dT= 20K
dt= 60min => dT= 10K
dt= 120min => dT= 5K
dt= 240 min => dT= 2.5K
dt= 480 min => dT= 1.25K
dt= 1440 min => dT= 0.42K



In other words, there is a demand if the demand temperature (current temperature) is 20°K below the set value for 30 minutes or 0.42K below the set value for 1440 minutes (= one day).
If the value falls below the double shortfall * set value demand temperature (corresponds to the value at 30 minutes), the characteristic curve is limited. If the difference between the set value demand and the current value of the demand temperature is greater than the double shortfall * set value demand temperature, the burner is started immediately (such as when the heating circuit switches from setback to standard mode or when a shutdown condition is no longer fulfilled and the heating circuits go into operation again).

In practice, neither the demand temperature nor the set value are constant. The difference between the two values over time will normally become greater, thus making the product of dt*dT greater, which is added to the register of sums and compared to the characteristic curve. This does not happen if the heating circuits switch, such as from standard mode into setback mode, or if the heating circuit pump shuts down due to a shutdown condition, etc. However, in such cases the energy is saved that the burner would have consumed if immediately demanded as soon as the set value has been underrun. In certain intervals, the program adds up the difference between the set value of the demand and the current value of the demand temperature. If this sum is greater than the product of shortfall * set value demand temperature for one hour with consideration of the immediate switch-on of the burner when the double shortfall is underrun, the burner is started.

Entire menu view:

DES: HEAT.REQ.	
FUNCTION STATUS:	
INPUT VARIABLE:	
OUTPUT VARIABLE:	
DEMAND TEMP.:	
T.dem.ACT: 64.3 °C	Current temperature of the sensor T.dem.
T.dem.SET: 60 °C	(Switch-on) threshold value at sensor T.dem.
DIFF.ON: 1.0 K	Switch-on difference to T.dem. (here, 61°C)
SHUTDOWN TEMP.:	
T.off ACT: 44.3 °C	Current temperature of the sensor T.off
T.off SET: 60 °C	(Switch-off) threshold value at sensor T.off
DIFF.OFF: 9.0 K	Switch-off difference to T.off (here, 69°C)
Low end temp.:	Burner demand, if T.dem falls below this value (only effective, if
T.dem.MIN: 20 °C	T.den.SET > +5°C)
Minimum runtime	
Burner: 90 Sec	
ECO MODE:	
Shortfall: 0 %	No eco mode

Most common example: Burner demand if the buffer cylinder is colder than the calculated heating circuit flow with the input variables:

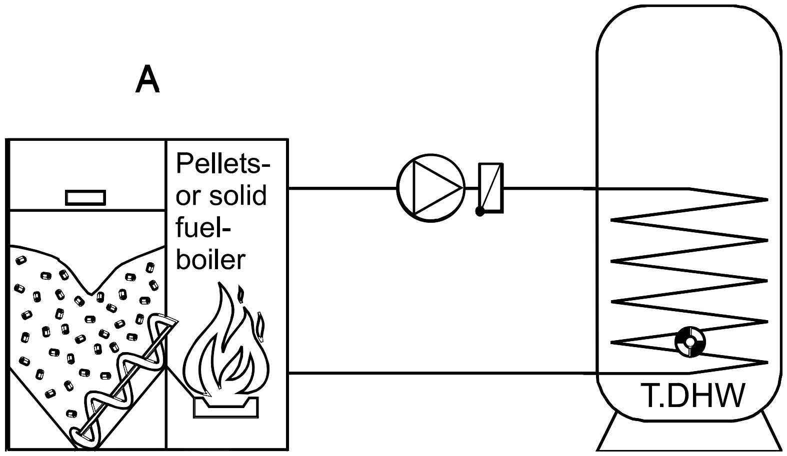
- ENABLE / User / ON = The function is enabled
- DEMAND TEMPERATURE: = Source: / input / sensor buffer top
- SHUTDOWN TEMP.: = Source: / User / unused = only one sensor is being used
- SET VALUE DEMAND: = Source: / HEATING CIRCUIT / T.flow SET = is thermostat value

The calculated flow temperature of the function *heating circuit 1* is given as the set value (as the thermostat threshold). The controller compares this value to the "demand" temperature *buffer top* along with the switch-on/off difference. Thus, the burner is demanded when the cylinder is colder than the calculated set flow temperature + DIFF.ON and is switched off if the cylinder is hotter than the set flow temperature + DIFF.OFF.

If the boiler sensor is entered instead of the cylinder sensor, the result is a sliding boiler operation, for which a low end temperature can also be set.

DHW demand

Basic diagram:



Input variables:	Output variables:
Enable DHW demand **DHW temperature** = T.DHW Set temperature = Desired DHW temperature External Switch = Switching between “standard operation” according to the time program (Status: OFF) and demand only of T.DHW MIN (Status: ON)	Effective set temperature = Time-related DHW set value T.DHW EFF Set Temperature = Desired cylinder temperature T.DHW SET Status demand, indication of the output A Burner output = Assignment only makes sense to analogue outputs A15 or A16

Simple description of the function:

Release of burner A if temperature in the cylinder (DHW temperature T.DHW) falls below the set temperature set via the time window.

Special features:

- The set temperature is also defined as an input variable in this function block. If it is only to be used as a simple set value, you need only enter *User* as the "source." Now, it will appear as a common function parameter in the function menu.
- The set temperature indicates the “desired temperature“ within definable time windows. In order to ensure a minimum cylinder temperature even outside the time windows, T.DHW MIN (minimum DHW temperature) can be used to generate a demand outside these specified times.
- The effective set temperature T.DHW EFF currently set by the time window is available as an output variable. If the cylinder exceeds this temperature, 5°C is output. The burner can then be demanded via another module (such as heating demand) by comparing T.DHW EFF with the buffer temperature.
- The *set value* T.DHW SET as another output variable is the temperature that the user determines. Therefore, the setting of the desired cylinder temperature can be transferred to other function modules.
- The input variable "**EXTERNAL SWITCH**" can be used via a remote wage to switch between standard operation in accordance with the time program and a demand based only on T.DHW MIN (for instance vacation).

- Neither thermostat threshold has hysteresis, but rather a common switch-on/off difference to the adjustable threshold value.
  Example: T.DHW SET = 50°C
  DIFF. ON = 1.0 K
  DIFF.OFF = 8.0 K
  In other words, if the temperature falls below T.DHW 51°C (= 50°C + 1°K), the output is activated, while it is switched off if the temperature exceeds 58°C (= 50°C + 8°K).
- The function block provides the burner output as an output variable. It can be assigned to a speed output or to the analogue output. For instance, the burner performance can be controlled (assuming you have the appropriate burner technology) via hardware output 15 or 16 (analogue output 0 - 10V). This is recommended when the burner's performance compared to the heat exchanger's performance sets off the excess temperature protection in the boiler.
- It is also possible to charge the cylinder outside the programmed time window to the set temperature once by pressing the key.

Entire menu view:

DES: WW-REQ	
FUNCTION STATUS:	
INPUT VARIABLE:	
OUTPUT VARIABLE:	
SINGLE CHARGING:	Charge the cylinder outside of the main time by pressing the key
START	
DHW TEMP.:	
T.DHW ACT: 54.3°C	Current temperature of the DHW cylinder
T.DHW SET: 50 °C	Set temperature of the DHW cylinder
TIME PRG.:	Opens the time menu (see **Time programs**)
T.DHW MIN: 40 °C	Minimum temperature of the DHW cylinder
DIFF.ON: 0.0 K	Switch-on difference to T.DHW SET and T.DHW MIN
DIFF.OFF: 4.0 K	Switch-off difference to T.DHW SET and T.DHW MIN
Burner outp.: 100 %	Specification for burner performance

Code for Technicians:

In order to enable all of the setting parameters, open the function "User" in the device’s basic menu and then select "Technician". Enter the product of 2^6 as the code!

Boiler cascade

Simple description of the function:

The coordination of up to three burner demands based on runtime and delay time by comparing the current demand temperature with a common flow temperature.
The indication of the functions involved (demand modules) automatically gives the module permission to control the burner via its internal signals "burner demand" and "set temperature." The highest set temperature is compared to the common flow temperature and issues a burner demand if need be. After the set delay time, the next burner stage is enabled if the conditions have been fulfilled, etc.

Input variables:	**Output variables:**
Enable boiler cascade (starting with first boiler stage)	Set Value = Highest demand temperature
Enable from second, third boiler stage	Status burner demands for boiler A, B, C Indication of the outputs
Flow temperature = Common flow	Status of boilers (1, 2, 3) = Status of burners demanded
involved functions = Indication of the demand modules involved	

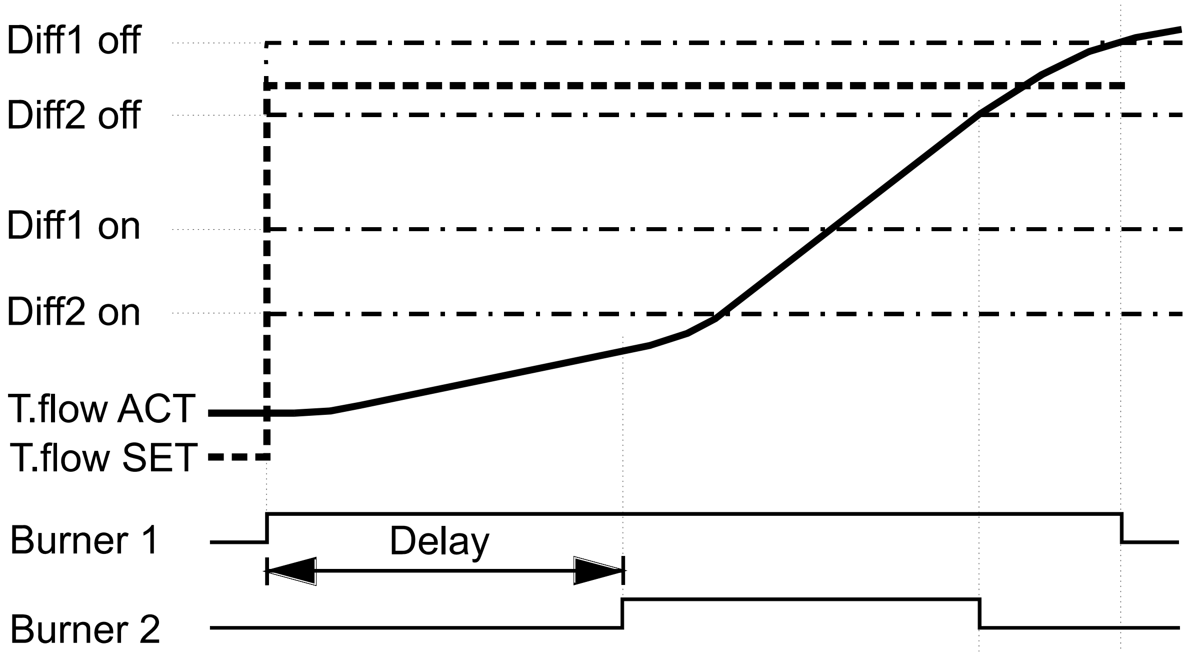
Special features:

- Few input variables as the module automatically communicates with them internally by means of the indication of the functions involved.
- Recording of the burner runtimes. You can therefore automatically switch which boiler is dominant by putting a limit on runtime.
- In addition to the necessary burner demands, the highest demand temperature (set flow value) and the switched stages are available as output variables.

Warning:

Sometimes it makes sense to link one of the output variables directly to the analogue output to create a 0 to 10 V or PWM signal. Linking this function is only allowed using analogue output A15, not with output A16.

The following runtime chart assumes that two boilers are to be controlled:



If there is demand (flow set temperature T.flow SET suddenly increases) and the flow temperature is less than the switch on temperature of the controlling boiler (=T.flow SET + DIFF1 ON), the first demand is generated. If the flow temperature remains under the switch-on temperature for the second boiler (T.flow SET + DIFF2 ON), a second demand is generated. The boilers are switched off in the same sequence as the flow temperatures exceed the switch-off temperatures (T.flow SDET + DIFF OFF).

The set flow temperature **T.flow SET** is linked with the following values of the involved functions and is determined from the highest of these temperatures:

1. From the function module **Heating demand**:
   Switch off temperature T.off SET+ DIFF.OFF,
   or demand temperature T.dem.SET+ DIFF.OFF, if no self-provided sensor is used for switching off,
   or base temperature T.dem.MIN + DIFF.OFF.
   The demand itself is generated, when the temperature falls below the demand temperature T.dem.SET + DIFF.ON or below the base temperature T.dem.MIN + DIFF.ON. An approximate burner minimum running time is not considered.
2. From the function module **DHW demand**:
   DHW set temperature T.DHW SET + DIFF.OFF
   or minimum temperature T.DHW MIN + DIFF.OFF (outside the time window)
   The demand itself is generated when the temperature falls below the hot water set temperature T.DHW SET + DIFF.ON or the minimum temperature T.DHW MIN + DIFF.ON.

If no demand arises out of the involved functions or the release is set to "OFF", then T.flow SET is +5°C.

Overall menu view (for two boilers as shown in the chart):

`DES:   BOIL.CASC`	
`FUNCTION STATUS:`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`SERVICE MENU:`	
`T.flow ACT:   34.6 °C`	Current flow temperature
`T.flow SET:     55 °C`	Desired set flow temperature from the demand
`BOILER 1:`	
`DIFF.ON:      -8.0 K`	Switch-on difference to T.flow SET (here, 47°C)
`DIFF.OFF:      2.0 K`	Switch-off difference to T.flow SET (here, 57°C)
`Time delay:    0 sec`	Switch-on delay for the first boiler (usually zero)
`BOILER 2:`	
`DIFF.ON:       -13 K`	Switch-on difference to T.flow SET (here, 42°C)
`DIFF.OFF:     -1.5 K`	Switch-off difference to T.flow SET (here, 53.5°C)
`Time delay:   15 min`	Switch-on delay for the second boiler 15 minutes

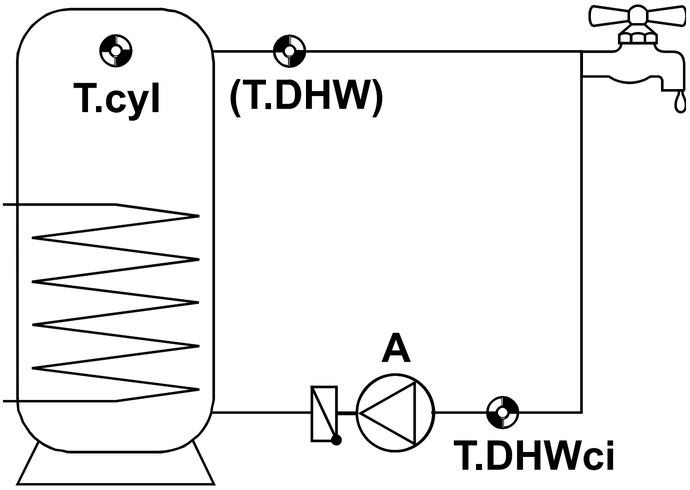
Boiler cascade

The variables in the service menu (according to example):

BOIL.CASC	
Boiler sequence:	
Boiler A: 1	Boiler A is top priority (= dominant boiler)
Boiler B: 2	Boiler B is second priority
Boiler A:	
automatic boiler change: yes	Change dominant boiler if A - B = 200 hours
Operating time 284 hrs	Total boiler runtime A = 284 hours
RESET METER: no	“yes” to reset the meter
Boiler B:	
automatic boiler change: yes	Change dominant boiler if B - A = 200 hours
Operating time 91 hrs	Total boiler runtime B = 91 hours
RESET METER: no	“yes” to reset the meter
Difference hours run for boiler change: 200 hrs	Once a difference of 200 operating hours between A and B is reached, the control boiler is changed over, provided an automatic boiler changeover is required (setting: yes).

DHW circulation

Basic diagram:



Input variables:	Output variables:
Enable DHW circulation **Return Temperature** = T.DHWci DHW temperature = T.DHW Set DHW circulation return temperature = maximum temperature allowed at T.DHWci Cylinder Temperature = T.cyl cylinder sensor for mixture protection	Effective set temperature T.DHWci.eff (takes into account also the mixture protection) Status DHW circulation, indication of the output

Simple description of the function:

Release of circulation pump A via time window as long as return sensor T.DHWci has not reached its upper limit (set temperature). In a simple application, the DHW sensor T.DHW does not have a function and is thus not used.

Special features:

- **Mixture protection 1**: Below a minimum cylinder temperature (T.cyl.MIN), the circulation function is blocked so that the layered remaining energy in the cylinder is not lost through pump operation.
- **Mixture protection 2**: To prevent a mixture above this threshold, the temperature difference is used between the cylinder temperature and the return temperature (DIFF.MIXT). If the cylinder temperature minus "DIFF.MIXT" is less than the set return temperature T.DHWciSET, this value is considered the limit temperature. Mixture protection is deactivated without a cylinder sensor ("source" user).
- If hygienic warm water is to be provided instead of a DHW cylinder, the pulse mode can be used as an alternative control method using a DHW sensor **T.DHW**. This approach requires a properly dimensioned plate heat exchanger including an **ultrafast** temperature sensor (MSP... = special accessory) at the warm water outlet. **T.DHW** also controls the water heating and circulation.

  If a faucet is opened, the temperature at **T.DHW** changes. If the temperature rises or falls by a set value within 1 second at **T.DHW**, the controller switches on the circulation pump. The pump is then switched off either after a set runtime or once the preset set value on **T.DHWciSET** has been exceeded. This ensures that DHW is available quickly at the faucet even when it is closed.
- In the **Time/pulse** mode, the time mode is active within the time window; the pulse mode, outside.

DHW circulation

Entire menu view:

`DES:   DHW CIRC.`	
`FUNCTION STATUS:`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`OPERAT: Time`	Switching to "Pulse" or "Time/pulse" mode (see special features)
`DHW CIRC. RTN:`	
`T.DHWciACT:    34.7 °C`	Current temperature of the return
`T.DHWciSET:      50 °C`	Set (maximum) temperature of the return
`          TIME PRG:`	Open the switch time menu
`DIFF.ON:       0.0 K`	Switch-on difference to T.DHWciSET (here, 50°C)
`DIFF.OFF:      5.0 K`	Switch-off difference to T.DHWciSET (here, 55°C)
`DHW TEMP.:`	
`T.DHW ACT:    53.2 °C`	Current DHW temperature

Other menu lines appear when a sensor has been indicated for the cylinder temperature:

`MIXT PROTECTION:`	
`T.cyl.ACT:    58.2 °C`	Current temperature of the cylinder
`T.cyl.MIN:      30 °C`	No circulation is allowed below this cylinder temperature (Hysteresis = 3K)
`DIFF.MIXT:   8.0  K`	If the cylinder temperature minus DIFF.MIXT falls below T.DHWciSET, the new value is “T.DHWciSET” (= Effective circulation return temperature)

For the option *Pulse* in lieu of *Time* the following menu options are displayed in place of the time program:

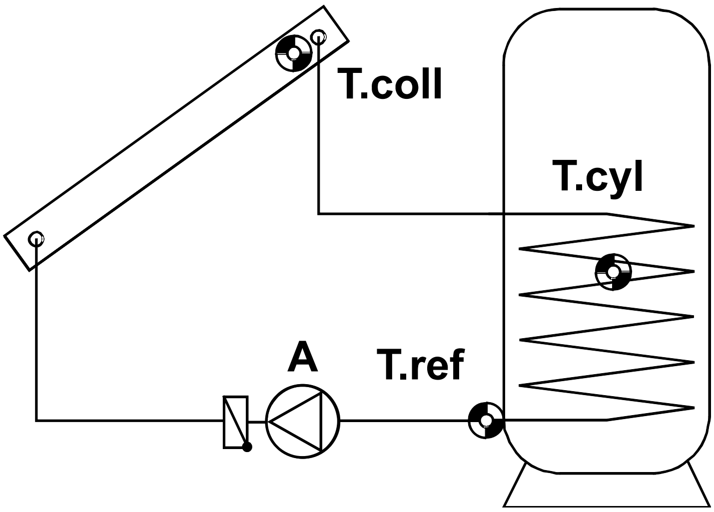
`Ddiff_on:      2.0 K`	Temperature change of 2K / seconds starts the pump
`Runtime:     90 Sec`	Maximum runtime per interval
`Pause time:   10 Min`	Minimum interval time (= minimum time between two pump runs)

If the operating mode *Time/pulse* is selected rather than *Time*, then within the time window, time mode is active, while outside it pulse mode is active.

In combination with hygienic service water preparation, the pulse mode runs very reliably with an ultrafast Sensor (special accessory). If standard sensors are used, the temperature change is recognized much more slowly. If there are problems, a flow switch for the circulation function can also be used instead of temperature measurement.
The suddenly increasing digital signal of the flow switch at the function input "DHW temperature T.DHW" causes immediate switching on of the circulation pump.

PID control (speed control)

The PID control can be used to change the delivered quantity – i.e. the volume flow – of usual commercial circulating pumps. That allows the system to maintain temperatures (differences). It can be used not only for the speed control but also for the control of the burner performance and others.
The following simple solar diagram illustrates the possibilities of this process:



Absolute value control = Stabilizing a sensor

T.coll can be stabilized very well at one temperature (such as 60°C) via the speed control. If solar radiation decreases, **T.coll** becomes cooler. The control unit reduces the speed and hence the flow rate. That leads to a long dwell time for the heat transfer medium in the collector, which in turn increases **T.coll**.
As an alternative, a constant return (**T.ref**.) can make sense in various systems (such as cylinder load). That requires an inverse control characteristic. If **T.ref** increases, the heat exchanger adds too little energy to the cylinder. Thus the flow rate is reduced. Greater dwell time in the exchanger cools the heat transfer medium more, thus reducing **T.ref**.
Stabilizing **T.cyl** would not make sense because the varying flow rate would not affect **T.cyl** directly and thus not lead to a functioning regulating circuit.

Differential control = Keeping the temperature between two sensors stable.

Keeping the temperature difference constant between, for example, **T.coll** and **T.ref** leads to a “sliding“ operation of the collector. If **T.coll** drops as a result of reduced irradiation, the difference between **T.coll** and **T.ref** also falls. The control unit reduces the speed leading to a greater dwell time of the medium in the collector and hence to a greater difference **T.coll – T.ref**.

Event control = If a set temperature event occurs, the speed control starts, thus keeping a sensor constant.

If, for instance, **T.cyl** reaches 60°C (activation threshold), the collector is to be kept stable at a certain temperature. The respective sensor is stabilized just as in the absolute value control.

Note: If the absolute value control (stabilization of a sensor) and the differential control (stabilizing the difference between the two sensors) are both active at the same time, the slower speed of the two methods “wins”. The event control “overwrites” the speed results from other regulation methods. This means that a defined event can block the absolute value or differential control.

Waveform

Two waveforms are available for motor control (in the menu “Outputs”).

Wave packet - only for circulating pumps with standard motor dimensions. Here, individual half-waves are blended in to the pump motor. The pump is run via pulses; the “smooth running” of the motor is only due to the moment of inertia.
Benefit: Great dynamics of 1:10 well suited for usual commercial pumps without internal electronics and a motor length of around 8 cm.
Drawback: Linearity depends on the pressure loss; there is some noise, not suitable for pumps with evidently deviating motor diameters and / or length from 8 cm.

Phase angle - for pumps and ventilation motors. The pump is switched to the mains within each half-wave at a certain point (phase).
Benefit: Suitable for almost all types of motors
Drawback: Low dynamics of 01:03 for pumps. A filter has to be inserted upstream from the unit with at least 1.8mH and 68nF to fulfill the CE standards for interference suppression (except A1 which on the other hand is able to carry a current of only up to 0.7A)
NOTICE: The menu allows a choice between wave packet and phase angle however in the standard version the output of waveform "phase angle" is not possible. Special versions on request.
Speed control via a phase angle control is not possible at outputs 2, 6 and 7.

Stability problems

The **proportional part P** represents the reinforcement of the deviation between the set and the actual value. The speed is changed in increments every **X***0.1 K of deviation from the set value. A large number makes the system more stable and leads to a greater deviation from the regulation.

The **integral part I** periodically adjusts the speed relative to the deviation remaining from the proportional part. For **1** K of deviation from the set value, the speed is changed one increment every **X** seconds. A larger number makes the system stable, but the adjustment to the set value is slower.

The **differential part D** leads to a short-term overreaction the faster a deviation occurs between the set and the actual value in order to provide the fastest compensation possible. If the set value deviates at a speed of **X***0.1 K per second, the speed is changed by one increment. Large numbers provide a more stable system, but it then takes longer to reach the desired value.

In some cases an adjustment of the PID values is necessary. The pump should be running in automatic mode if the system is ready for operation and has the appropriate temperatures. With I and D are set to zero, the proportional part P is reduced from 10 each 30 seconds until the system is instable and the pump speed changes its rhythm. It can be read in the menu above the PID parts. The proportion at which instability begins is noted as P_{crit}, with the duration of the oscillation (= time between the two highest speeds) noted as t_{crit}. The correct parameters can be determined with the following formulae.

$$P = 1{,}6 \times P_{crit} \qquad I = \frac{t_{crit} \times P}{20} \qquad D = \frac{P \times 8}{t_{crit}}$$

A typical result of **hygienic service water preparation** with an ultrafast sensor is PRO= 8, INT=9, and DIF=3. For reasons not entirely understood, the settings PRO=3, INT=1, and DIF=4 have proved to be effective. Probably, the control unit is so instable here that it oscillates very quickly, appearing balanced due to the systems and the fluids inertia.

Pump standstill

The wave packet method (standard) allows the volume flow to be changed by a factor of 10 in 30 increments. If the flow rate is too low, return flaps may cause a standstill as well as a low output of the pump at low speeds of the control unit. This may even be desirable, which is why the lower limit of 0 is admissible. A reasonable speed limit is found in a simple test. Select manual mode in the menu “Outputs“ and set a speed. The rotor can be observed with its hood removed. Now, the revolutions are reduced until the rotor comes to a standstill. Three stages above this limit will provide safe pump operation. The indication of the lower speed stage occurs in the respective function *Speed control*.

Input variables:	**Output variables:**
Enable PID control	Correcting variable = Calculated speed stage Indication of the speed control control output
Temperature ***absolute value control*** = Sensor which should be kept stable at the set temperature. Set value absolute value control = Desired control temperature	
Temperature (+) ***differential control*** = Base sensor (the warmer sensor, e.g. collector) of the differential control Temperature (-) differential control = Reference sensor (the colder sensor, e.g. cylinder) of the differential control	
Activation temperature ***event control*** = Sensor where an event is expected. Activation threshold = Temperature event at the above sensor Control temperature event control = Sensor which is kept stable after the occurrence of an event. Set value event control = Desired set regulation temperature for the event control	

Simple description of the function:

With the indication of temperature sensors the volume flow in the hydraulic system is controlled via the variability of the pump speed to keep the respective sensor constant at a set temperature.

Special features:

- The **correcting variable** is available as output variable also for other functions for further use. In addition, it also can be linked to an analogue output (A15, A16) instead of the pump outputs.
- It is possible to set all control processes separately to ***standard*** control operation (speed increases along with the temperature), to ***inverse*** operation (speed decreases as temperatures increases) or to ***OFF*** (control process is not active).
- If the absolute value control (stabilization of a sensor) and the differential control (stabilizing the difference between the two sensors) are both active at the same time, the “slower speed wins out”.
- If 2 PID controllers act simultaneously on an output, then the faster speed "wins".

- Event control “overwrites” the results of speed control from other control processes. Hence, a set event can block the absolute value control or differential control. **Example**: Keeping the collector temperature at 60°C with the absolute value control is blocked when the cylinder has already reached 50°C at the top = the fast provision of DHW is complete and is now to be continued with full volume flow (and hence a lower temperature but slightly better efficiency). To do so, a value that automatically requires full speed has to be entered as the new set temperature in the event control (e.g. coll = 10°C).
- If absolute value as well as differential control are switched off (output: maximum correcting variable), then there will be a switch from maximum correcting variable to the value that complies with the event control when the event control is activated.

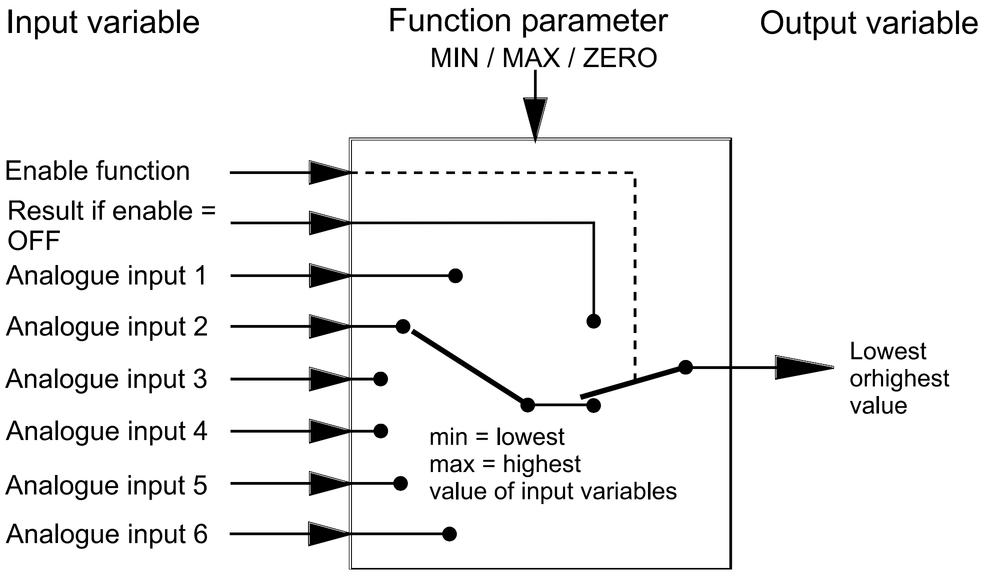
Entire menu view:

`DES:    PID CTRL`	
`FUNCTION STATUS:`	
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`ABSOL.VAL. CTRL:`	
`MODE: std`	Speed increases as temperature increases
`T.abs.ACT:    50.3 °C`	The sensor is currently measuring 50.3°C
`T.abs.SET:      50 °C`	The sensor is kept at 50°C
`DIFFERENTIAL CONTROL:`	
`MODE: std`	The speed increases with the difference T.diff+ to T.diff-
`T.diff+.ACT: 50.3 °C`	The sensor at the source is currently measuring 50.3°C
`T.diff-.ACT: 42.7 °C`	The reference sensor is currently measuring 42.7°C
`SET DIFF       8.0  K`	The desired difference (T.diff+ to T.diff-) should be 8 K
`EVENT CONTROL:`	
`MODE: off`	No event control is allowed. If *std,* then:
`COND.:    ACT < THRES`	Condition
`T.activACT:     48.1 °C`	The sensor that activates the function is measuring 48.1°C
`T.activTHR:     60 °C`	Event control should launch at 60°C on (act-) sensor (fixed activation threshold, no hysteresis)
`T.ctrl ACT    50.3 °C`	The sensor control starting with the event is displaying 50.3°C
`T.ctrl SET      90 °C`	Starting with this event, the sensor is controlled to 90°C
`CORRECTVAR.:`	
`maximum:   30`	Highest admissible speed stage is the stage 30 (full run)
`minimum:    8`	Lowest admissible speed stage is stage 8 (0 also admissible)
`current:   14`	Stage 14 is currently output
`CONTROL PARAMETER:`	
`P: 10  I: 0  D: 0`	PID shares for stable operation

The control parameters P=8, I=5, and D=2 generally ensure stable operation. If the speed periodically changes (generally for 20-30 seconds), I and D should be set to zero for simple systems. Drawback: A low, constant temperature is incorrectly controlled for, and the system is a bit slower.
If a speed control is used in service water systems, the PID settings must be determined in a test (see "stability problems") to ensure optimal performance.

Analogue function

Basic diagram:



Input variables:	Output variables:
Enable analogue function Result (Enable = Off) Analogue input variables 1 - 6	Result Indication of the speed control output

Simple description of the function:

According to the basic diagram it looks for the highest (lowest) value of the analogue inputs. Besides the heating circuit and charging pump module this module represents an extremely versatile and important link to the burner demand. In addition, it offers also simple calculating operations.

Special features:

- When an entry is made in the function list, the number of analogue inputs can also be entered. Thus, not all six inputs have to be assigned.
- The function produces the following result as an output variable via a control command from the inputs:
  - ***MIN***: Output of the minimum value of the input variables.
  - ***MAX***: Output of the maximum value of the input variables.
  - ***AVERAGE***: The output variable is the **mathematical average** of all input variables. In this way, the average of multiple measurement values can be calculated.
  - ***FILTER***: The output variable is the **temporal average** of the first input variable. All other inputs are ignored. The average time can also be set.
  - ***SUM***: The output variable is created based on the following formula from the sum of input variables E(1-6): sum = E1 - E2 + E3 - E4 + E5 - E6. For example: if the input variable E2 is set to *User* and zero is entered in the parameters for E2, E1 + E3 are simply added.
  - ***ZERO***: Output of the number zero as an output variable.
- If the module is blocked (enable = off), a value is output that either the user determines via "RESULT (ENABLE: = Off)" or that is created from an own input variable. It is therefore possible to switch between analogue values via the enable.
- The specification of *User* and analogue input creates an adjustable numeric value in the menu of the function.
- In the input variables, an offset that is added to the value of the variables can also be set.
- **Digital** conditions can also be processed at the inputs: if the condition is ***OFF***, 0 is used to calculate the value; if the condition is ***ON***, the set value “Offset” of the respective input variables is used for the calculation.

Analogue function

Example of use:

The greatest temperature that the system currently demands is calculated from the three functions "heating circuit 1." "heating circuit 2" (output variable = set flow temperature), and DHW demand (output variable = effective set temperature) so that the burner demand is later correctly compared to the buffer cylinder temperature. In addition, the customer also wishes to have a constant buffer standby temperature. When this function was called, the number of input variables was already set at four. The following parameters are now set in the submenu *INPUT VARIABLES*:

```
INPUT VARIABLE  1:
Source: HTG CIRC.1
1: Set flow temp.
Offset:   0.0 K
```

Input variable 1 is the set flow temperature of the function HTG CIRC. 1

```
INPUT VARIABLE  2:
Source: HTG CIRC.2
1: Set flow temp.
Offset:   0.0 K
```

Input variable 2 is the set flow temperature of the function HTG CIRC. 2

```
INPUT VARIABLE  3:
Source: DHW_DEM.
1: eff.set temp.
Offset:   0.0 K
```

Input variable 3 is the effective temperature of the function DHW_DEM.

```
INPUT VARIABLE  4:
Source: User
```

The user sets the socket temperature in the menu

Entire menu view:

```
DES: MAX(an)
INPUT VARIABLE:
OUTPUT VARIABLE:

FUNC.VAR:  Temp.
```

All inputs are temperatures

```
FUNCTION:  MAX
VAR. 1:    53.6 °C
VAR. 2:    66.4 °C
VAR. 3:     5.0 °C
VAR. 4:    40.0 °C
```

Output of the highest temperature of the inputs
Set flow temperature of the function HTG CIRC.1
Set flow temperature of the function HTG CIRC.2
Effective temperature of the function DHW_DEM
Low end temperature set by the user

```
if ENABLE = Off
             0 °C
```

If the analogue mode has not been released, the module outputs 0°C

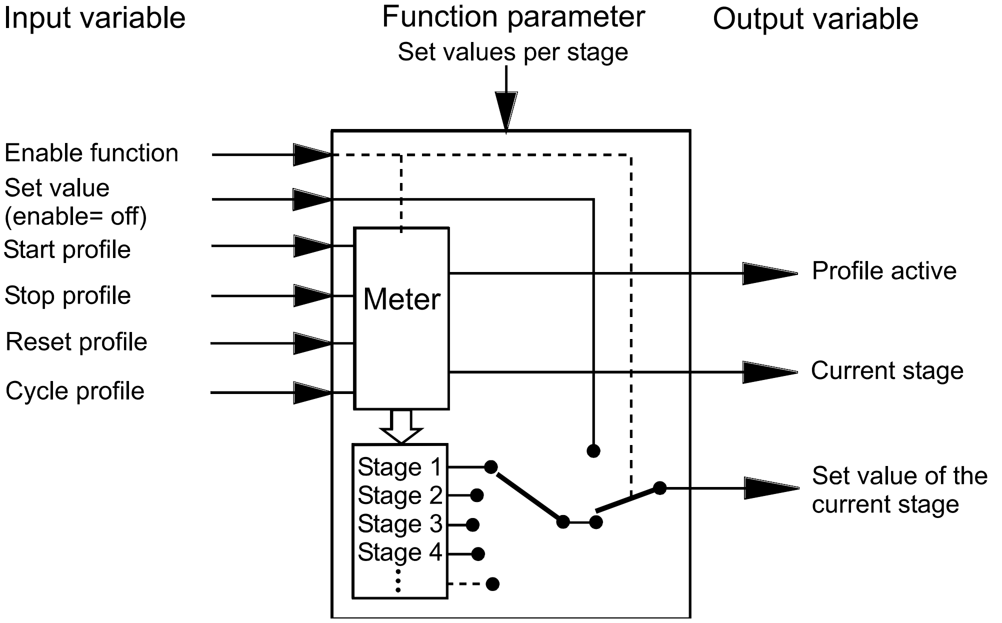
```
RESULT:    66,4 °C
```

Result of the analogue function

The function therefore provides 66 .4°C as the greater value for the output variable. As an input variable, this temperature now allows a comparison with the temperature at the top of the buffer cylinder in the function *heating demand*. If the buffer is colder than 66 .4°C (+ diff), the burner is required.

Profile function

Basic diagram:



Input variables:	**Output variables:**
Enable profile Set value if (Enable = Off) Start profile = Starting the time controlled process Stop profile = Stopping the time controlled process Reset profile = Reset to stage 0 (profile disabled) Cycle profile = More forward by 1 step (from step 1)	Status profile active = Output ON when set value is not OFF Indication of the output Set value = Value of the current stage Current stage

Simple description of the function:

This function generates a time-controlled output of **up to 64 values**. In each stage, the system switches from one value to the next in a set table and outputs the value as a "set value." In this way, a profile can be set up, e.g. a temperature profile that is suitable for a floor screed drying out program.

Special features:

- The input variables Start, Stop, Reset or Cycle profile must be digital commands (ON/OFF) (e.g. digital input, switching output or another function, etc.)
- Each of the input variables can be manually operated directly from the function by entering *User*. However, the command "STOP PROFILE" behaves differently than a linked input variable in manual mode. In the link, the meter is only stopped as long as the stop signal is active, thereafter the meter continues running. In the manual mode, "STOP PROFILE" also generates a reset. The meter then starts from the beginning again when restarted.
- A cyclical process is possible: the first value is called again after the last one.
- If the module is blocked (enable = off), a value is output that is either determined via "Set value if (enable = Off" or that another module creates as an input variable. It is therefore possible to use the enable to switch between the profile and an external analogue value.
- The table entry OFF means: During this step, the profile is not active. A value is output, that can either be specified via “Set value if (enable = Off)” or which originates from another module as the input variable.
- The following functional variables can be set as the set value: temperature, dimensionless, output, heat amMWh, heat amkWh, number of impulses, time and solar radiation.

Profile function

Although the profile stage is registered every 6 hours in the internal storage, it gets lost while loading new function data (load factory settings, load backup copy, data transfer from the C.M.I.).
If an internal cycle > 23.5 hours is set (such as heat drying of a pavement floor), profile stage 1 is saved in the internal memory immediately after the profile function starts. This ensures that the heat drying program continues to run when the controller is reconnected after a blackout just after the heat drying of the floor pavement has been launched.

Example:

A temperature profile for heat drying of floor pavement is to be created. It is assumed that all input variables are set to *User* so that the function can be manually changed at any time.

Entire menu view:

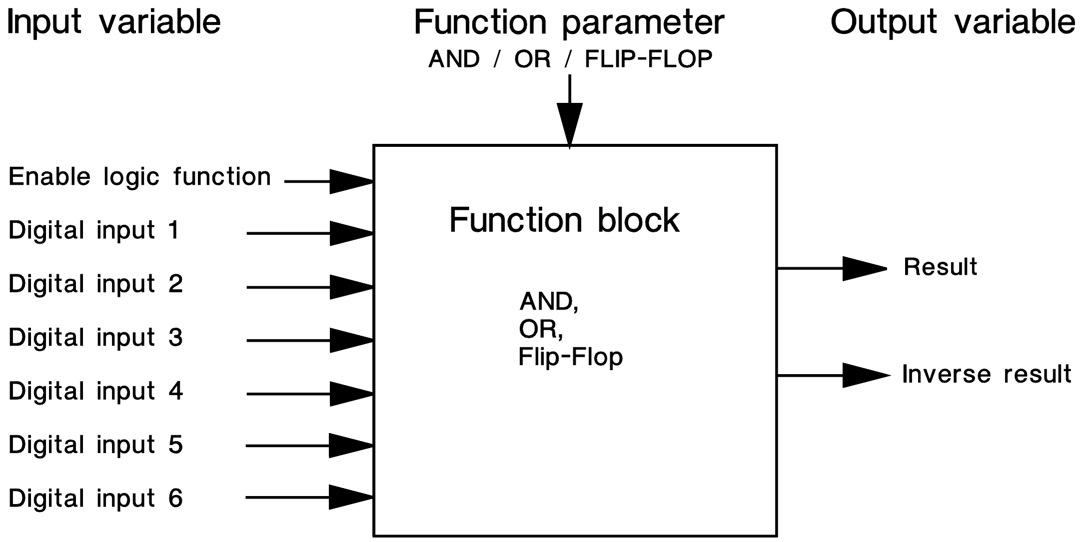
`DES: PROFILE`
`INPUT VARIABLE:`
`OUTPUT VARIABLE:`	

`FUNC.VAR:    Temp.`	The values are interpreted as temperature
`cyclical:       no`	No repeat after the profile has ended
`Int. Sync:  24.0 Hrs`	Switch to the next value every 24 hours (Adjustment range 1 sec. to 48 hours)
`START PROFILE`	Press the scroll wheel to manually start the function The following will appear when started: `STOP PROFILE` (Display only if the input variable "Start Profile“ is set to *User*)
`CURRENT STAGE:  3`	
`SET VALUE:   26.0 °C`	The set value of stage 3 is 26°C
`Stage 1:   20 °C`	
`Stage 2:   23 °C`	
`Stage 3:   26 °C`	
`Stage 4:   30 °C`	
`Stage 5:   35 °C`	
`Stage 6:    OFF`	No profile is active on the sixth day, output of the set value, if Enable = OFF
`Stage 7:   30 °C`	
`Stage 8:   26 °C`	
`Stage 9:   22 °C`	
`if ENABLE = off` `0.0 °C`	If the profile mode has not been released, the module outputs 0°C

If the output variable "PROFILE ACTIVE" is now assigned to the heating circuit pump and the function module "MIXER CONTROL" handles the set value, a pavement floor heat drying program has been created for nine days. Here, make sure that a heating circuit control function does not simultaneously control the output. To be on the safe side, set the enable for the heating circuit control function to *User OFF* during the process.

Logic function

Basic diagram:



Input variables:	**Output variables:**
Enable logic function Digital input variables 1 - 6	Status result, indication of the output Status inverse result, indication of the output

Simple description of the function:

AND- function:	Output = ON only if all inputs are ON.
OR- function:	Output = ON if at least one input is ON.
FLIP FLOP- function:	Output = Saves the status of the inputs

Special features:

- After having registered the function in the function list the indication of the number of digital inputs is possible. Thus, not all six inputs have to be assigned.
- The **FLIP-FLOP** function (also known as holding circuit) works according to the following formula:
  - Output = constantly ON if at least one of the inputs I1, I3, I5 were set to ON (set holding circuit), even if the input decreases again afterwards (set pulse).
  - Output = constantly OFF if at least one of the inputs I2, E4, I6 were set to ON (delete holding circuit). The “Delete” command dominates. Hence, no setting is possible when a delete input is ON (reset pulse).
- The function “OFF” is also available. In this way, the function is inactivated by the easiest means. The status *OFF* applies to the direct output and the status *ON* to the inverse output.
- In addition to the direct output, the inverse output variable is also available.
- If the module is blocked via enable, both the direct and the inverse output are ***OFF***.

Logic function

Example:

The heating circuit is to be released based on the two thermostat functions "comparison_1" and "comparison_2" when one of the two (OR-function) is triggered. When this function was called, the number of input variables was already set at two. The following parameters are now set in the submenu *INPUT VARIABLE*:

`INPUT VARIABLE  1:`	
`Source: COMP.1`	Input variable 1 is the output of the thermostat function COMP.1
`1 : Va > Vb + diff`	
`Mode:   normal`	Adoption of normal start condition of the module
`Status:   ON`	We see current status ON
`INPUT VARIABLE  2:`	
`Source: COMP.2`	Input variable 2 is the output of the thermostat function COMP.2
`1 : Va > Vb + diff`	
`Mode:   normal`	Adoption of normal start condition of the module
`Status:    OFF`	We see current status OFF

Thus, the function forms the command ON as the output variable. In the function *heating circuit controlr*, it now allows the release of the pump as an input variable if either the "boiler thermostat" **or** the "buffer thermostat" has exceeded the required temperature.

Value table based on two inputs + enable:

AND

Enable:		Input 1:	Input 2:		Result:	Inv. result		Commentary:
ON		OFF	OFF		OFF	ON		
ON		ON	OFF		OFF	ON		
ON		OFF	ON		OFF	ON		
ON		ON	ON		ON	OFF		
OFF		X	X		OFF	OFF		Both outputs OFF

OR

Enable:		Input 1:	Input 2:		Result:	Inv. result		Commentary:
ON		OFF	OFF		OFF	ON		
ON		ON	OFF		ON	OFF		
ON		OFF	ON		ON	OFF		
ON		ON	ON		ON	OFF		
OFF		X	X		OFF	OFF		Both outputs OFF

FLIP FLOP

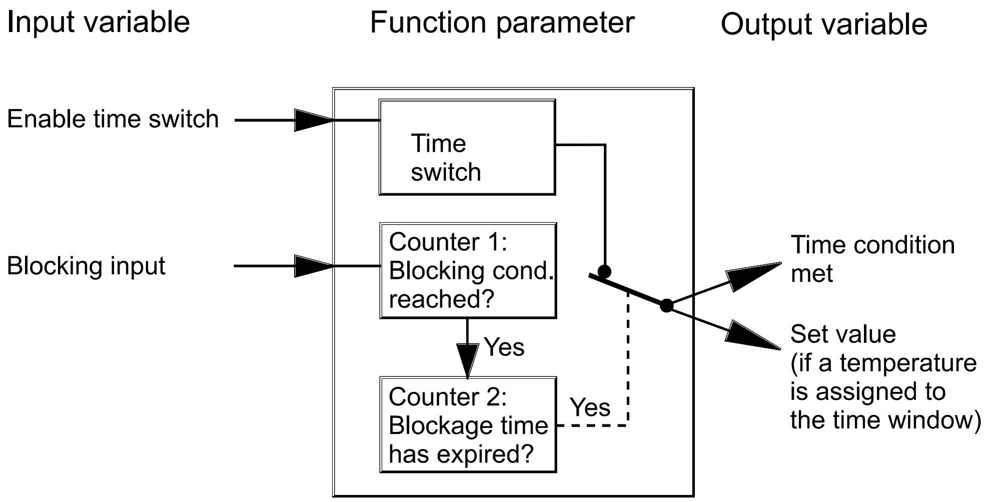
Enable:		Input 1:	Input 2:		Result:	Inv. result		Commentary:
ON		OFF	OFF		OFF	ON		Previous status
ON		ON	OFF		ON	OFF		I1 is saved!
ON		OFF	OFF		ON	OFF		Previous status
ON		OFF	ON		OFF	ON		I2 deletes output
ON		ON	ON		OFF	ON		I2 dominant
OFF		X	X		OFF	OFF		Both outputs OFF

OFF

Enable:		Input 1:	Input 2:		Result:	Inv. result		
ON		X	X		OFF	ON		
OFF		X	X		OFF	OFF		Both outputs OFF

Time switch

Basic diagram:



Input variables:	**Output variables:**
Enable time switch	Set value (if a temperature is assigned to the time window)
	Status time condition met
Blocking input	Indication of the output

Simple description of the function:

There are a maximum of 5 time programs each with 3 time windows available per module.
As a freely usable time switch clock this function can be employed in various ways. For instance, this function could be used to provide a time control for filter pumps in swimming pools or for fan motors in hot-air heating systems. The function block has the same operational structure as all other time switch functions, such as in the heating circuit control function.
If the function “time switch” is put before another function (such as “charging pump”) as an INPUT VARIABLE/ENABLE, the function receives additional time conditions. As with all other function modules, the time switch can be entered several times in the function list, i.e. multiple time switches can be available.

Special features:

- When the function is set up, the question "with set value?" *yes/no* appears next to the questions about scope (time programs, window). If you enter *no*, you'll have a normal digital time switch. If yes is entered, the user can assign a temperature to each time window that will later be available as an output variable according to the time windows. A set value can be entered simultaneously if the time program is not completed.
- If *User* is entered as the "source" for the input variable BLOCKING INPUT, a simple time switch function is the result.
- If another function is assigned as the "source" to the input variable BLOCKING INPUT, the time switch can be blocked for a certain period if certain events occur.

Time switch

Example:
Time switch with two time programs, each with three time windows

Entire menu view:

```
DES: TIME
INPUT VARIABLE:
OUTPUT VARIABLE:

Mo Tu We Th Fr Sa Su
06.00 - 07.30  h
12.00 - 21.00  h
00.00 - 00.00  h

Mo Tu We Th Fr Sa Su
05.00 - 07.00  h
12.00 - 22.00  h
00.00 - 00.00  h
```

The first time program is active on all workdays
Switched on on workdays at 06:00 AM and off at 07:30 AM
etc.
Time window not used

The second time window is active on the weekend
Switched on at 05:00 AM and off at 07:00 AM
etc.
Time window not used

If using a set value, the following lines appear according to the time matrix:

```
Set val if TP not
cmpl.:        5° C
```

Entry of a set value outside the time window, during blockage or with enable = OFF

If another function uses the blockage input, the following will appear:

```
Min.time block.cond:
     0 Days  5.0 min
Blocking Time:
     0 Days 10.0 hr
```

The condition has to be met for at least five minutes

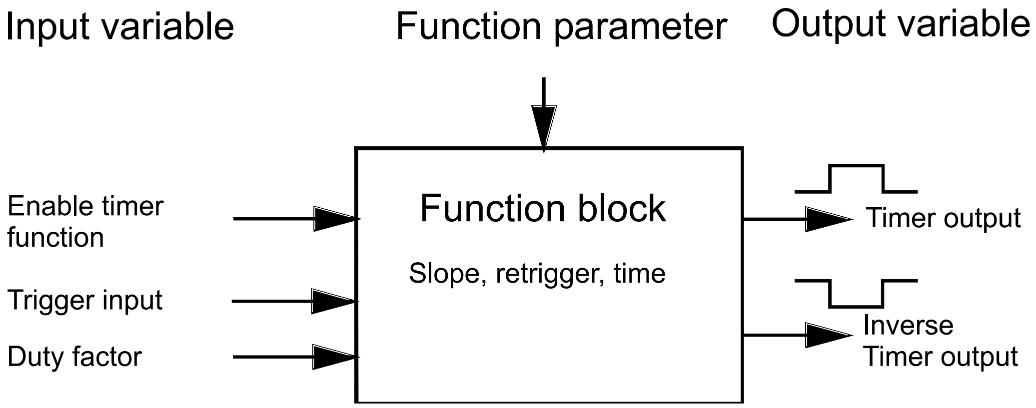
Then, the time switch is blocked for 10 hours

Legionella protection is another **example**. Here, the time switch function is used to heat up the cylinder to 60°C at night to prevent legionella from forming. If this temperature is reached anyway during the day (such as by the solar system), there is no need of additional heating, and the function is blocked:
A comparison function (thermostat) at the blocking input lets the first meter run (“Min.time block.cond”) as long as the cylinder is hotter than 60°C. If the set meter time has been reached (five minutes), a second time meter blocks the time switch until it has expired (10 hours). The cylinder is therefore not heated up once again at night using oil, gas, or electricity if the protective temperature was already reached during the day.

While the time switch is already blocked once the first meter time (“Min.time block.cond ”) has been reached, the second meter (blocking time of time switch) only begins to run when the blockage input returns to the status "OFF."

Timer function

Basic diagram:



Input variables:	**Output variables:**
Enable timer Trigger input = Input signal for starting the timer Duty factor = Relation between input and output signal	Status Timer Output, indication of the output Status Inverse Timer Output, indication of the output

Simple description of the function:

Independent time elements can switch time sequences between functions. An input status triggers a time laps of the timer function (= impulse time), which runs irrespective of the time of day. The impulse time can be set up to 90 seconds in increments of seconds in addition to various stages of up to 48 hours.

Special features:

- The set impulse time can be varied from 0-100 percent via the input "DUTY FACTOR." The impulse time can thus be varied via signals and computer values. To make it an adjustable value in the menu, *User* has to be entered as "source".
- Use the command MODE to choose between six basic functions.
- If enable = OFF, then both output variables are set to OFF.

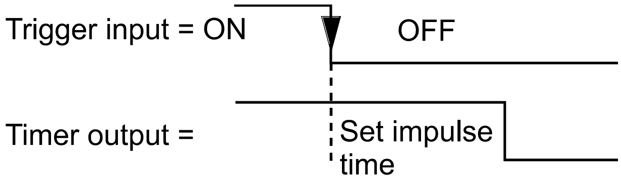
Entire menu view:

`DES: TIMER`
`FUNCTION STATUS:`
`INPUT VARIABLE:`
`OUTPUT VARIABLE:`	

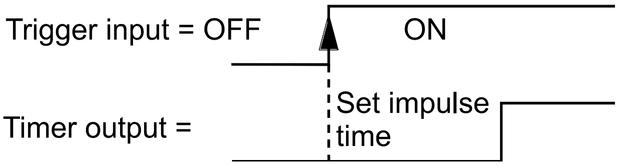
`MODE: Delay`	Input affects output with a certain delay
`TRIGGER:` `retrigger:     yes`	A second trigger while the timer is running restarts the timer
`IMP. TIME:     8 Sec`	Timer runtime
`DUTY FAC.:      100 %`	100% of 8 seconds = 8 seconds!
`MAN.: TIMER START`	The timer can be launched using the scroll wheel and stopped before it has expired.

Timer function

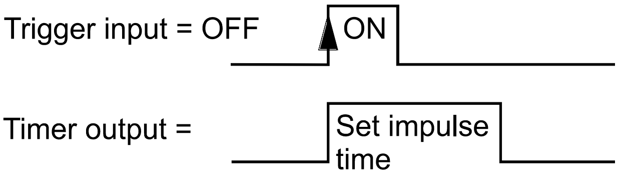
Run-on time: The ON signal at the trigger input immediately switches the output on. If the input drops (OFF), the output remains ON for the duration of the timer period.



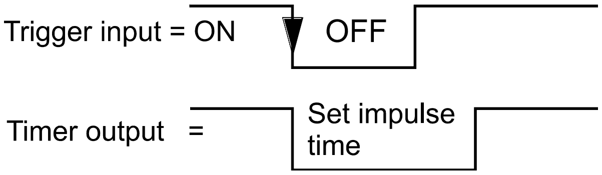
Delay: The ON signal at the trigger input is only passed on when the timer period at the output has elapsed. An OFF signal at the trigger input causes immediate switching off of the output.



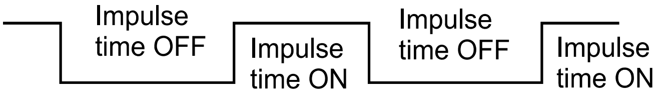
Min. runtime: The ON signal at the trigger input immediately switches the output on. If the input drops during the timer window (OFF), the output remains switched on nonetheless until the timer window has expired. If the trigger input has the status ON after the impulse time has elapsed, then the output stays switched on.



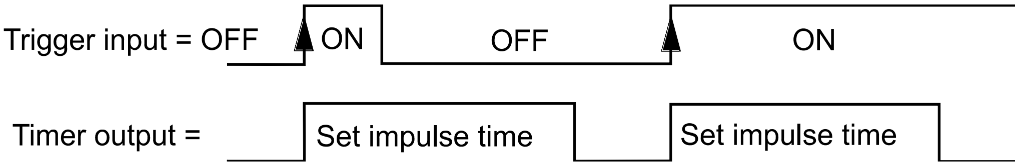
Blockage: The ON signal at the trigger input only switches the output on, after the timer window has expired since the last ON signal.



Astable: By indicating a switch-on and off time, a pulse generator is created without a trigger input. If the pulse-duty factor is also used for controls, it changes the switch-on time. The setting switch-off time = 0 is a special case: The switch-on time then corresponds to the entire period, and the pulse-duty factor to the relationship between the switch-on and the switch-off time
Example: A duty factor of 30% results in 30% ON and 70% OFF for the entered switch-on time.

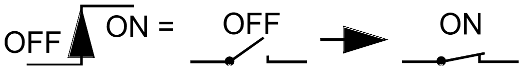


Impulse: If the selected trigger slope occurs, the output for the timer time switches on. A trigger input status change during the impulse time does not change the output status.

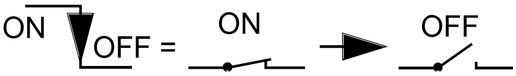


The trigger slope is positive if the input status changes from "OFF" to "ON" or from "switch open" to "switch closed" (= closing). A change from closed to open (= opening) is a negative trigger slope. TRIGGER SLOPE = *pos/neg* starts the timer anytime there is a status change at the input.

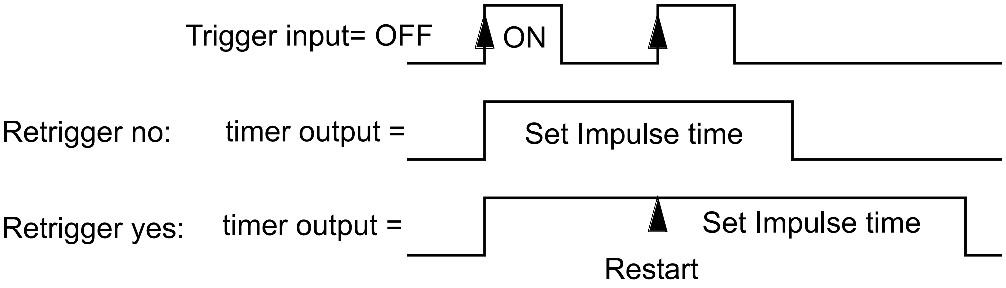
Trigger slope positive:



Trigger slope negative:



The properties of **retriggering** based on the example of a positive trigger slope:



Synchronisation

Simple description of the function:

This module provides an output variable relative to the date and time based on the date and time information of the device. In this way, periodic signals are available that directly relate to the time, day of the week, date, or season and allow for date or time-specific releases to control other function modules.

Input variables:	**Output variables:**
Enable synchronisation	Status time condition met, indication of the output Status summer time ON/OFF Status controller start = Starting up of the control unit

Special features:

- The function allows up to five date or time windows. The number has to be set after calling the module.
- Via the command “MODE”: periodic time windows in intervals of hours up to one year can be programmed.
- The setting "cyclical/once" determines whether the window whose parameters are set is to occur once or repeatedly (cyclical).
- The output "Contr. Start" only generates a 30 second pulse when the device is switched on or reset.

Example:

Assuming that a damp basement room is to be periodically heated, a time laps is prepared for other modules that then handle the heating. This procedure is to take place four times a year in the summer when there is enough solar energy in the buffer cylinder.

Entire menu view:

```
DES:   SYNC.
INPUT VARIABLE:
OUTPUT VARIABLE:

MODE:      Year
       cyclical

Day Mon     Day Mon
15.  06. -  17. 06.

05.  07.  -  07. 07.
25.  07.  -  27. 07.
10.  08.  -  12. 08.
```

Procedure within one calendar year
repeating every year

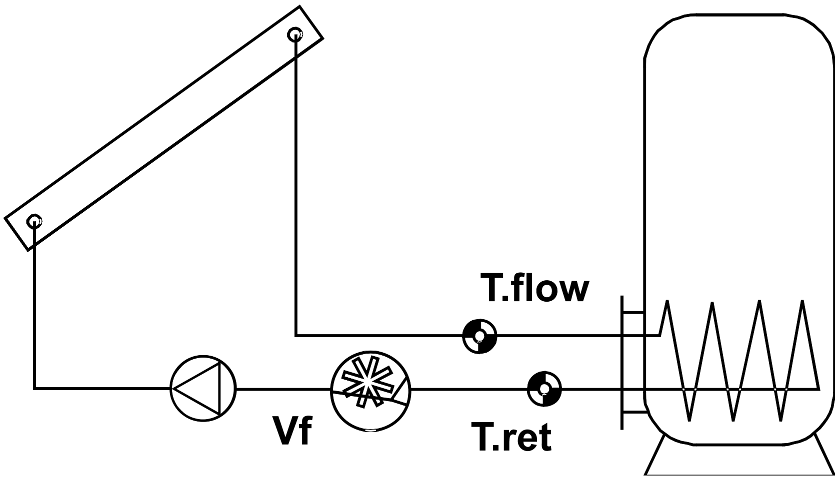
Output variable ON from the 15th 00:00 hours to the 17th of June 00:00 hours, etc.

N.B.: In modes "Year" and "Month" the time window begins and ends respectively with 00:00 hours for the specified days.

In modes “Hour” and “Day” the time window begins and ends respectively with the start of the specified minute.

Heat meter

Basic diagram:



Input variables:	Output variables:
Enable heat meter **Flow temperature** = T.flow **Return temperature** = T.ret Volume flow = Vf Meter reset	Current output Kilowatt hours Megawatt hours

Simple description of the function:

Calculation of the heat output and quantity via the temperature difference and volume flow with consideration of the share of frost protection in the heat medium.

Use as an electrical energy meter:

1. The sources of the input variables flow temperature and return temperature are set to *User / unused*.
2. The impulses of the electrical meter are captured as input 15 or 16 (Setting: Type: Impulse, Measurement variable: Flow rate). In this case the quotient setting does not correspond to litre/impulse, rather Wh/Impulse. This input must be defined as the input variable "*Flow rate*".
3. If the setting range (Wh/Impulse) of the input is insufficient, then this can be increased in the function menu by a factor (between 1 and 100).

For each impulse, the heat meter is increased by the Quotient * Factor (Wh).

Special features:

- When the temperature difference is calculated, the tolerance of the sensors and the measuring device may lead to large errors (if the difference is 10°K, the error is approximately 30%). To compensate for these errors, the device has a patented **calibration method** that can be found in the service menu.
- The collector sensor can also be used as the flow sensor. However, it always has to be installed on the flow outlet of the collector bar using an immersion sleeve. The heat measured then also contains the losses from the solar flow line.
- Meter reset function in the input variables and in the service menu.
- The output variables (output, MWh, kWh) can be taken up by other modules and input variables.
- A fixed value can be set as the flow rate instead of the transducer if *User* is in the input variable "volume flow."

WARNING: Although the meter of the function module Heat meter is recorded **every 6 hours** in internal memory, it is lost when loading new function data (load factory settings, load backup copy, data transfer from the C.M.I.)! Therefore for this reason, should a power failure occur, up to 6 hours meter data can be lost.

Calibration mode

By simultaneously measuring the same temperature with both sensors, the deviation of the sensors from each other can be calculated and included as a correction factor when calculating the heat amount in the future.
The calibration only has an influence on the sensor values in the function "Heat meter" and is not considered in other functions.
During calibration, it is important that both temperatures (flow and return) measure the same temperatures. Here, both of the sensor tips are tied together using tape or a wire. In addition, the two sensors should already be equipped with the line expansions. When using the collector sensor, the required line length has to be estimated approximately and integrated. The sensors must be connected to both parameterized inputs of the flow and return lines and are **both** submerged in a hot water bath (i.e. both measure the same temperature).

Overall submenu view - SERVICE MENU:

RESET	Reset the heat meter
METER: no	
HEAT AMOUNT:	
123.4 kWh	Total heat amount in kWh
CALIBRATION	
START: no	Start command for calibration
Status: UNCALIBRATED	The heat meter has not been calibrated
DIFFEREN. 0.56 K	Display of the difference measured during the calibration process

Calibration process:

1. Submerge the sensors in the water bath.
2. Start the calibration process by selecting "START *yes*"
3. After successful calibration, the status "CALIBRATED" appears. The measured difference value is displayed.

If the calibration was carried out incorrectly or erroneously, the result can only be corrected by carrying out a new calibration.

Entire menu view:

DES: HQC.1	
INPUT VARIABLE:	
SERVICE MENU:	
Status: CALIBRATED	
FROST PROT: 45 %	Indication of frost protection share in %
T.flow: 62.4 °C	The flow temperature is 62.4°C
T.rtn: 53.1 °C	The return temperature is 53.1°C
DIFF: 9.3 K	The difference between the flow and the return is 9.3°K
FLOW RT: 372 l/h	The current flow rate is 372 l/h
OUTPUT: 3.82 kW	The current output is 3.82 kW
HEAT AMOUNT:	
19 834.6 kWh	The current heat amount is 19,834.6 kWh

Meter

Simple description of the function:

As a operating hours meter or pulse meter (such as for the burner demand) this function represents another service function.

Input variables:	Output variables:
Enable meter (Max. 6 digital) Input variables Meter reset	Meter reading

Special features:

- When the meter function is entered in the function list, the number of "input variables" must be given. This can be changed later via "CHANGE FUNCTION." Both sensor inputs and other functions or output can be considered functions involved.
- In MODE *HRS RUN MTR (*hours run meter)*,* the meter runs if **at least** one function involved is switched on. Only whole minutes are counted.
- In MODE *IMPULSE CTR,* as long as the status of one variable is "ON" with multiple input variables, the impulses of the other input variables are ignored. In addition, a divider can be indicated. If this divider is set to, say, 2, only every second pulse at the input variables raises the meter by one. The meter can read impulses up to a **maximum frequency of 1 Hz** (=1 pulse per second). The **minimum pulse duration** across inputs 1 to 14 is 500 ms, across inputs 15 and 16, 50 ms.
- The meter reading can be reset using input variables or the service menu.
- The output variable "Meter reading", which is not visible, can be taken over by other modules as an input variable.

Entire menu view:

```
DES: METER
INPUT VARIABLE:
SERVICE MENU:

MODE: HRS RUN MTR

Operating time:
    324 hr   18 min

Day meter prev day:
      4 hr   37 min
```

WARNING: Although the meter reading of the function module "Meter" is recorded **every 6 hours** in internal memory, it is lost when loading new function data (load factory settings, load backup copy, data transfer from the C.M.I.)! Therefore for this reason, should a power failure occur, up to 6 hours meter data can be lost.

Maintenance FN

This function is intended as a service function for the chimney sweeper and/or as a simple burner switch for the exhaust measurement. The preset output (generally 100%) is switched on for the set time after the burner has been started. In addition, the heating circuits set in the input variables are activated **with the maximum admissible flow temperature T.flow MAX**. The value of the output variable T.flow SET of this heating circuit is displayed as 5°C during the active maintenance function.

These targets could also be reached via the manual mode (switch corresponding outputs to MANUAL/ON). Assuming that the user does not have a manual for the controller or that the person does not have enough time to read the entire manual, this function should make things easier.
Use the input variable "**EXTERNAL SWITCH**" to activate the maintenance function via a specially installed switch or via a switch output of another function without any inputs at the controller being activated. An "External switch" must be set to "ON" for the duration of the maintenance function (**no runtime limit**). The function must be deactivated again via this switch.

Input variables:	**Output variables:**
External switch	Status burner demand, indication of the output
involved functions = Indication of the heating circuits	Burner output, indication of the speed control output

Entire menu view:

`FUNCTION START`	Press the scroll wheel to activate the burner and the heating circuits => number `FUNCTION STOP`
`---------------------`	
`DES:   FL G INSP`	
`Status:     OFF`	The function is deactivated (stopped)
`Runtime:      0 Min`	Remaining burner runtime
`INPUT VARIABLE:`	
`OUTPUT VARIABLE:`	
`Ttl runtime:   20 Min`	Total burner runtime after function start
`Burner outp.:    100%`	Desired burner output during service time

The function block provides the burner output as an output variable. It can be assigned to a speed output or an analogue output. For instance, the burner performance can be controlled (assuming you have the appropriate burner technology) via analogue output 15 or 16 (analogue output 0 - 10V).
The output of the burner performance is dominant in the maintenance function. In other words, no other analogue signal is allowed during maintenance (such as from DHW demand). However, digital signals can overwrite the analogue value at any time.
Once the burner demand has been switched off (function stopped), the heating circuits involved still remain active for three minutes to take the residual heat from the boiler. If the mixer behaviour is detected as "close" in the heating circuit, then the mixer is set to "closed" for 20 minutes (= maximum remaining runtime) and the heating circuit pump is switched off. Only then does the heating circuit return to the set operating mode.

Function control

In the solar and heating section, a number of functions perform important tasks that could lead to wrong reactions if there's a malfunction. For instance, if a defective cylinder sensor in the solar system detects temperatures that are too low, the solar system will run under false premises and take heat out of the cylinder. The module FUNCTION CONTROL can be used to monitor various operating modes and will issue an error message if there's a malfunction or block the inoperable function via its enable.

Input variables:	**Output variables:**
Control value a Control value b Enable differential control	Status error value, indication of the output Status error difference, indication of the output

Simple description of the function:

This function allows for two sensors (control value a, b) to be monitored in order to detect a short-circuit, an interruption and the maximum admissible temperature difference. Likewise, it is possible to monitor a sensor or temperature via a defined threshold value.

Special features:

- If there is interruption or short-circuit that affects the basic function of the module, an error message is issued only after 30 seconds.
- In addition, a temperature threshold or difference can be monitored using “ENABLE DIFF CTRL” If this control has been enabled, then the following applies:
- If sensors are assigned to both control values, the monitoring of the differences is active.
- If control value b is set to *User*, it is an adjustable temperature threshold that applies for the control value a as a limit value to be monitored.
- If the monitoring of the difference is not enabled, the message DIFFEREN. OK appears in the error display nevertheless. It generally suffices to monitor the circulation of just one circuit in solar warm water systems with multiple consumers (via enable). If another circuit is running, the message of monitoring should still be displayed.
- If only one sensor is monitored (control value b = *User*) or if the difference is monitored, a malfunction is only messaged after an error time that can be set. This helps to prevent unjustified error messages caused by temperature peaks when the system is starting.
- The parameters are found in their own parameter menu to provide an overview of the error evaluation at all times.
- Via the command “Save error: yes", the display ERROR remains until it is manually deleted even after the error has disappeared.

Warning:

Sometimes it makes sense to link one of the output variables directly to the control output to create a 0 to 10 V or PWM signal. Linking this function is only allowed using control output A15, not with output A16.

Function control

Entire menu view:

(No error)

```
DES: FUNC CTRL
INPUT VARIABLE:
OUTPUT VARIABLE:
PARAMETER:

T.collector      OK
57.4  °C
T.DHW1           OK
48.9  °C
DIFFEREN.        OK
 8.5  K

Save error:    yes

Delete error disp.?
```

(With error)

```
BEZ.: KONTR.SOL1
INPUT VARIABLE:
OUTPUT VARIABLE:
PARAMETER:

T.Collector      ERROR
9999  °C     lead brk
T.DHW1           OK
48.9  °C
DIFFEREN.        ERROR
9999  K       too high

Save error:    yes

Delete error disp.?
```

The parameter menu contains the following if monitoring a difference:

```
Error if over
at least      30 min
CVa - CVb  >    50  K
```

Error minimum time setting
Difference threshold setting

Or if value a is being monitored, for example:

```
Error if over
at least      30 Min
CVa        >    30°C
```

Error minimum time setting
Error threshold setting

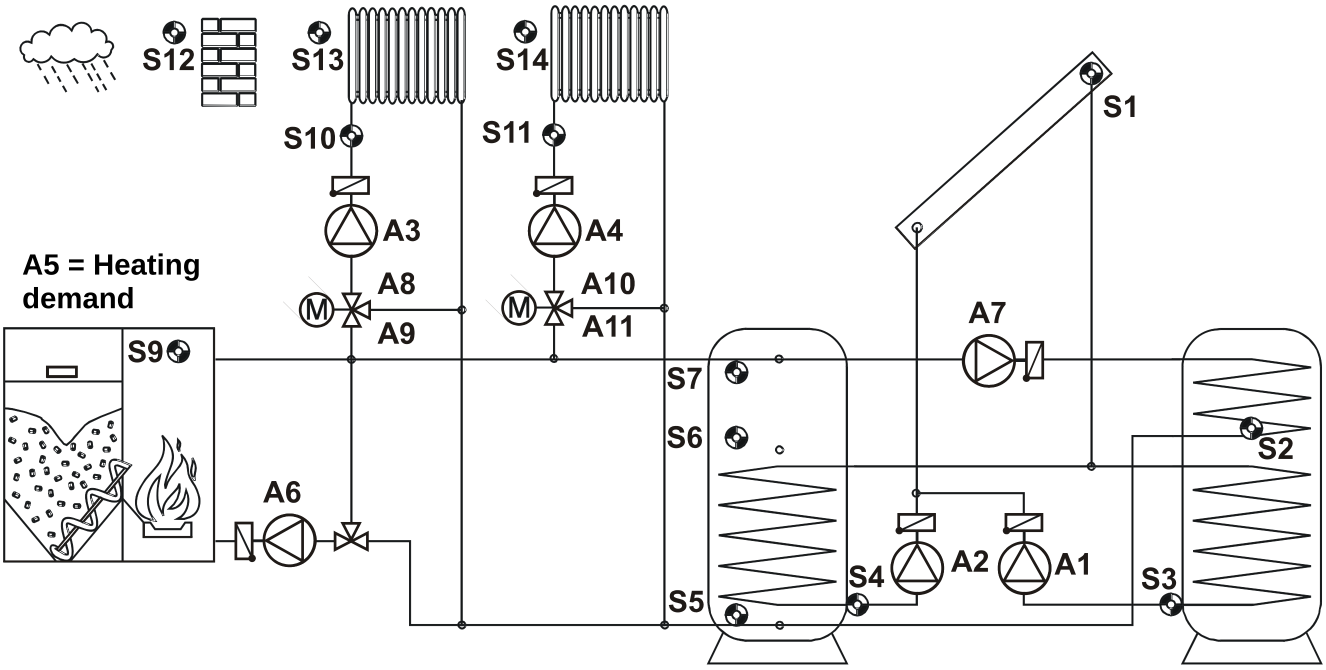
Troubleshooting:

"Save error: yes" ERROR remains displayed even after the remedy has been provided until the user presses the scroll wheel to confirm "Delete error display?" If the error continues after deletion, the message will be reissued after a certain delay.
"Save error: no" ERROR is automatically deleted once the error has disappeared.
If an output has been assigned in the output variables, it responds the same as the display.
The status lines of the function control should also be entered in the function overview using the user interface editor. In this way, users will always have the information they need in their menu.

Factory setting

TA_FACTORY SETTING - The function data with this specification have been transferred to the control unit. **The TA factory setting can be loaded by simultaneously pressing the two input keys and scroll wheel when starting up the controller.**

The factory settings are based on the following hydraulic diagram for solar warm water system with a buffer and DHW cylinder, a boiler fired with wood pellets or oil/gas, and two heating circuits:



A detailed description of the programming can be found on our homepage www.ta.co.at.

Installation instructions

Sensor installation

The sensors must be arranged and installed properly for the system to function correctly. To this end, make sure also that they are completely inserted in the immersion sleeves. The threaded cable connections provided serve as a strain relief. The clip-on sensors must be insulated to protect them from being influenced by the ambient temperature. Water must be kept out of the immersion sleeves when used outdoors (**damage from freezing**).
In general, the sensors may not be exposed to moisture (such as condensation water), which might enter the cast resin and damage the sensor. If this happens, heating the sensor to 90°C for an hour might help. When using immersion sleeves in NIRO cylinders (inoxydable) or pools, pay attention to their **non-corrosion properties**.

Collector sensor (red or grey cable with connection box): Insert either in the tube directly soldered or riveted to the absorber and sticking out of the collector's frame or set a t-shaped connector on the outer collector's flow collector tube. Screw an immersion sleeve with an MS (brass) threaded cable connection (= to protect from moisture) into the t-shaped connector and insert the sensor. To protect from lightening, the connection box has an overvoltage protection which is fixed in a parallel way between the sensor and the expansion cable.

Boiler sensor (boiler flow): This sensor is either screwed into the boiler using an immersion sleeve or at a short distance from the boiler on the flow.

Cylinder sensor: The sensor that the solar power system needs should be used with an immersion sleeve for fin coil heat exchangers just above the exchanger or, if integrated bare-tube heat exchangers are used, on the lower third of the exchanger or the exchanger's return line so that the immersion sleeve is inside the exchanger's tube. The sensor that monitors the heating of the cylinder from the boiler is installed at the level of the desired amount of warm water during the heating season. The plastic threaded cable connections provided can be used to provide strain relief. It should not be installed below the respective register or heat exchanger in any case.

Buffer sensor: The sensor that the solar power system needs is installed on the bottom of the cylinder just below the solar heat exchanger using the immersion sleeve provided. The plastic threaded cable connections provided can be used to provide strain relief. It is recommended that the sensor between the middle and the upper third of the buffer cylinder be used together with the immersion sleeve as a reference sensor for the heater's hydraulics or - flush with the cylinder's wall – inserted under the insulation.

Pool sensor (swimming pool): Put a T-shaped connector on the suction line immediately on the line leading from the pool and screw the sensor in with an immersion sleeve. In the process, make sure that the material used is non-corroding. Another option is to put the sensor on the same spot using hose clamps or adhesive tape and to provide thermal insulation for ambient influences.

Clip-on sensor: Use pipe clamps, hose clamps, and the like must be attached to the respective line. Make sure the material used is adequate (corrosion, temperature resistance, etc.). Then, the sensor has to be well insulated so that the tube temperature is measured exactly and the ambient temperature does not influence the measurement.

DHW sensor: When the control system is used in DHW systems with an external heat exchanger and speed-controlled pump, changes in the amount of water have to be **reacted to quickly**. Hence, the DHW sensor has to be put directly on the heat exchanger's outlet. A T-shaped connector should be used to insert the ultrafast sensor (special accessory) in the outlet using an O-ring. The heat exchanger has to be installed upright with the warm water outlet on top.

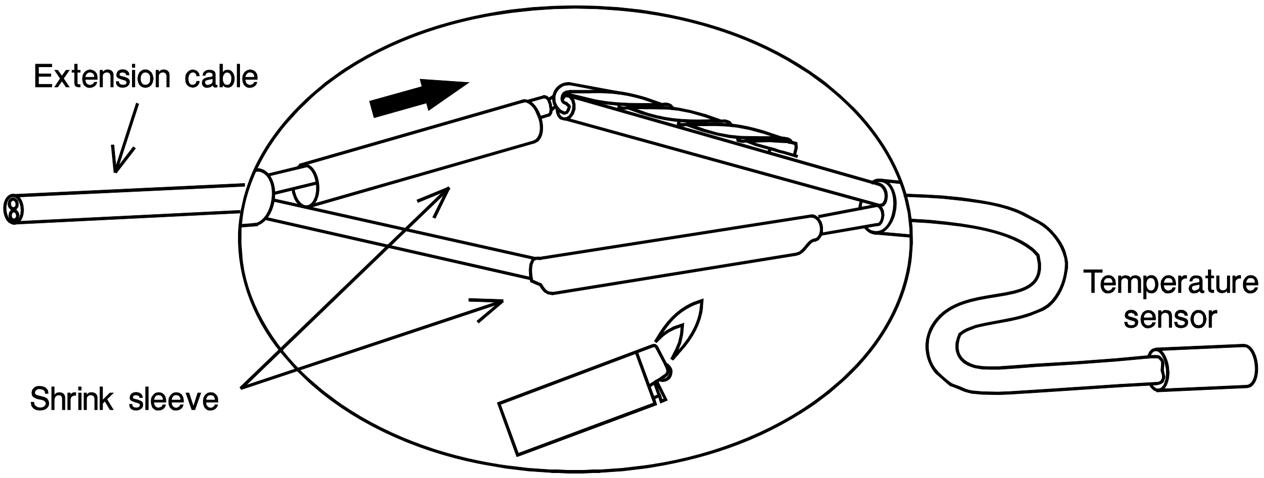
Radiation sensor: To get a measurement according to the collector's position, it should be parallel to the collector. It should thus be screwed onto the metal sheet or next to the collector along an expansion of the assembly rail. To this end, the sensor case has a blind hole that can be opened at any time. The sensor is also available as a wireless sensor.

Room sensor: This sensor is intended for installation in the living room (as a reference room). The room sensor should not be installed near a source of heat or near a window. The sensor is also available as a wireless sensor.

Outdoor temperature sensor: This sensor is installed on the coldest wall side (usually the north) some two meters above ground. Avoid temperature influences from nearby air shafts, open windows, etc.

Sensor lines

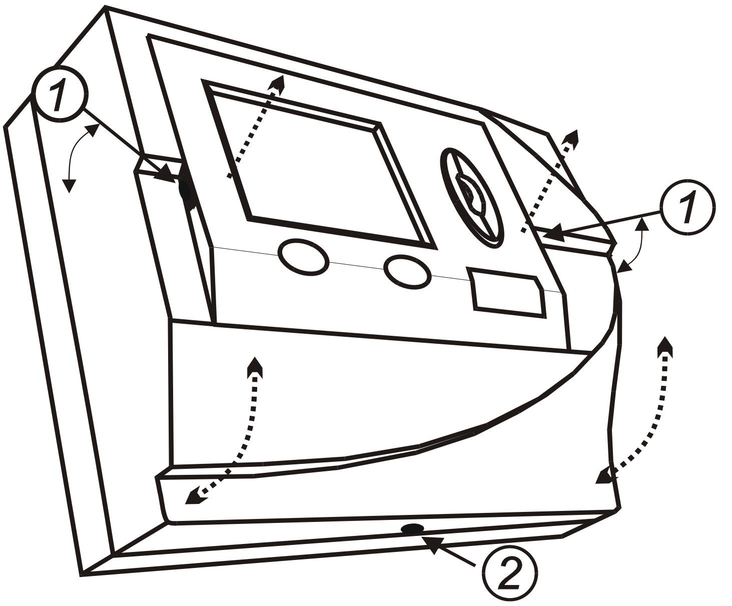
All of the sensor lines with a cross-section of 0.5mm2 can be extended up to 50m. With this length of line and a Pt1000 temperature sensor, the measurement error is approx. +1K. Longer lines or a lower measurement error require an appropriately larger cross-section. The sensor and the probe can be connected by putting the heat-shrinkable sleeve truncated to 4 cm over a wire and twisting the bare ends. If one of the wire ends is tinned then the connection must be made through soldering. Then the heat-shrinkable sleeve is put over the bare, twisted ends and carefully heated (such as with a lighter) until it has wrapped the connection tightly.



In order to prevent measurement fluctuations, the sensor cables must not be subject to negative external influences to ensure fault-free signal transmission. When using non-screened cables, sensor cables and 230V network cables must be laid in separate cable channels and at a minimum distance of 5 cm. If screened cables are used, the screen must be connected to the sensor earth (GND).

Installing the device

WARNING! Always pull the mains plug before opening the casing!

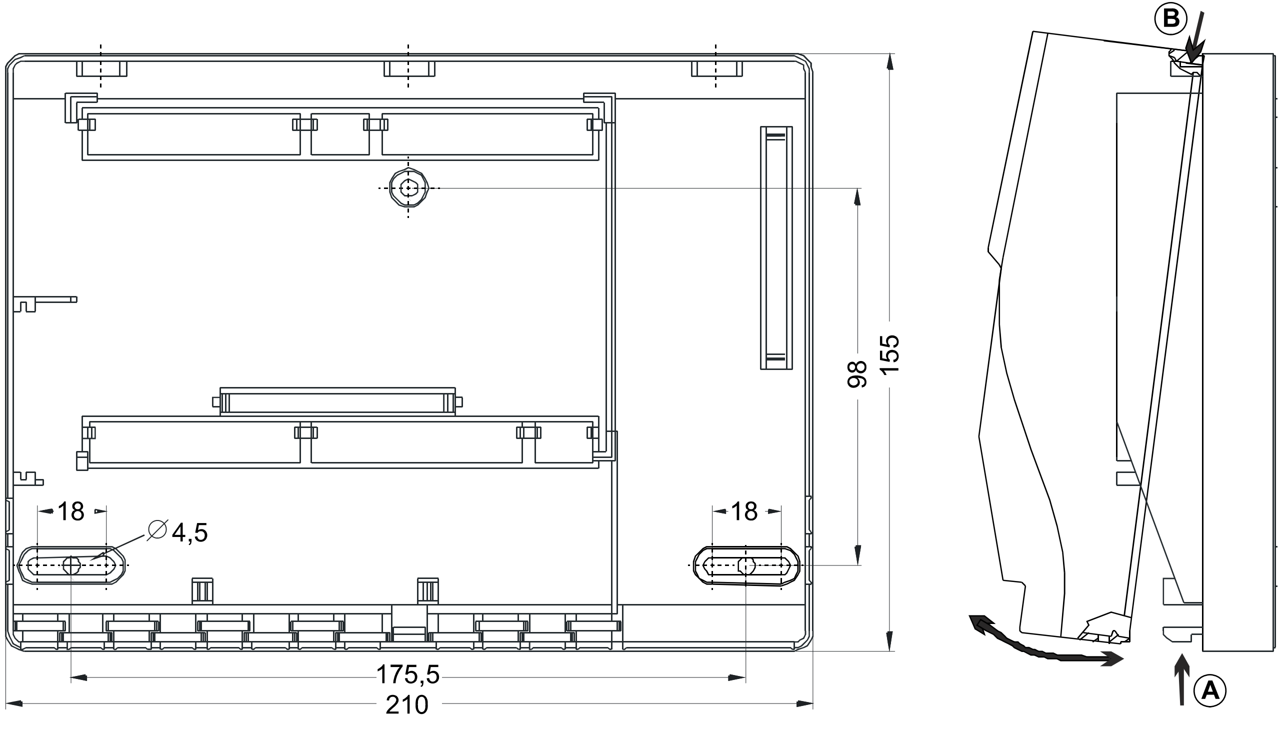


To open the console the control unit has to be separated from it as follows:

Use two big screwdrivers to remove the clamps (labeled 1 in the sketch to the left) and list the device out of the console with the screwdrivers. Once the control device has been removed, unlock the clip by pushing with a small screwdriver (point 2 sketch left), flip the lid up and to the back, and remove it.

Attach the console at eye-level (approximately 1.6 meters high) to the wall using the installation material provided so that the cable outputs are at the bottom. The console has a separate lead for each power supply cable. The very fine separation bridges sometimes break when the openings are popped out. As each power cable will later have its own strain relief, this is not a problem.

Drawing:

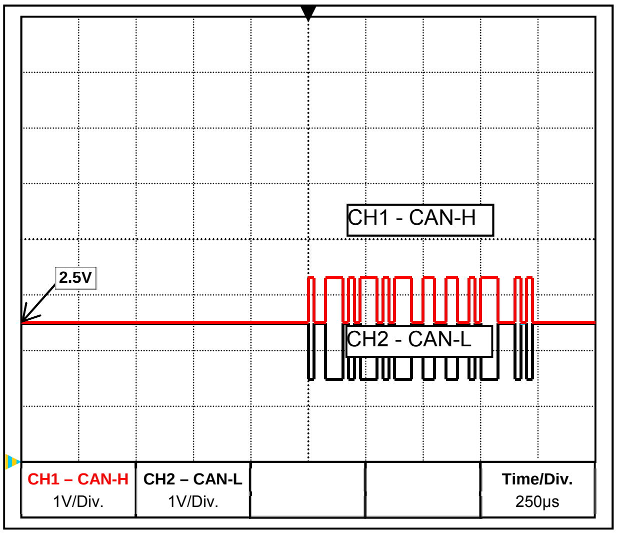


Control cabinet version UVR1611S:
The opening in the cabinet must have a size of at least 138x91 mm; the insertion depth is 70 mm including the power strips.

CAN BUS network

Guidelines for the topology of a CAN network

Technical principles



CAN-H and CAN-L data signals

The CAN BUS comprises the cables CAN-High, CAN-Low, GND and one +12 V supply cable for BUS components without their own power supply. The combined total load of all devices with 12 V and 24 V supply must not exceed 6 W.

Design CAN networks in a linear fashion and set a terminator at each network termination. This is ensured by the termination of the end devices.

In the case of larger networks (covering several buildings), problems can occur through electromagnetic interference and potential differences.
To avoid or to the greatest extent manage such problems, take the following measures:

- **Cable screening**
  The BUS cable screen must be connected well at every node to provide continuity. For larger networks we recommend including the screen in the equipotential bonding, in line with the examples shown.

- **Equipotential bonding**
  As low an ohm connection as possible to the earth potential is particularly important. Where cables enter buildings, ensure that the cable entries are in the same location where possible and that all are connected to the same equipotential bonding system ($S_{ingle}E_{entry}P_{oint}$ principle). The purpose is to create potentials that are as similar as possible, in order to achieve the smallest possible potential difference to adjacent lines in case one line suffers a voltage surge (lightning strike). Also ensure a corresponding clearance between the cable and lightning protection systems.
  The equipotential bonding also has positive properties to counteract interferences emitted from linked cables.

- **Avoiding earth loops**
  Where a BUS cable is routed between several buildings, ensure that earth loops are avoided. The reason for this is that buildings actually have different potentials compared to the earth potential. An earth loop is created when connecting one cable screen in each building **directly** with the equipotential bonding system. In other words, a current flows from the higher to the lower potential.
  For example, if lightning strikes near one of the buildings, the potential of that building will briefly be raised by several kV.
  In this case, the equalising current flows to earth via the BUS screen and causes an extreme electromagnetic input which can result in the destruction of the BUS components.

Lighting protection

Efficient lighting protection is highly dependent on good building earthing that meets the relevant regulations.
An external lightning protection system offers protection against a **direct** lightning strike.
In order to protect against voltage surges in the 230 V mains supply cable (**indirect** lightning strike), appropriate lightning conductors and surge arresters compliant with local regulations must be fitted in the upstream distribution systems.
In order to protect the individual components of a CAN network against **indirect** lightning strike, we recommend the use of surge arresters specifically developed for BUS systems (special accessories: **CAN-UES**).

Example: Gas discharge arrester for indirect earthing EPCOS N81-A90X

Examples of different network versions

Key to symbols:

□ … device with its own power supply (UVR1611K, UVR1611S, UVR1611E, UVR16x2)

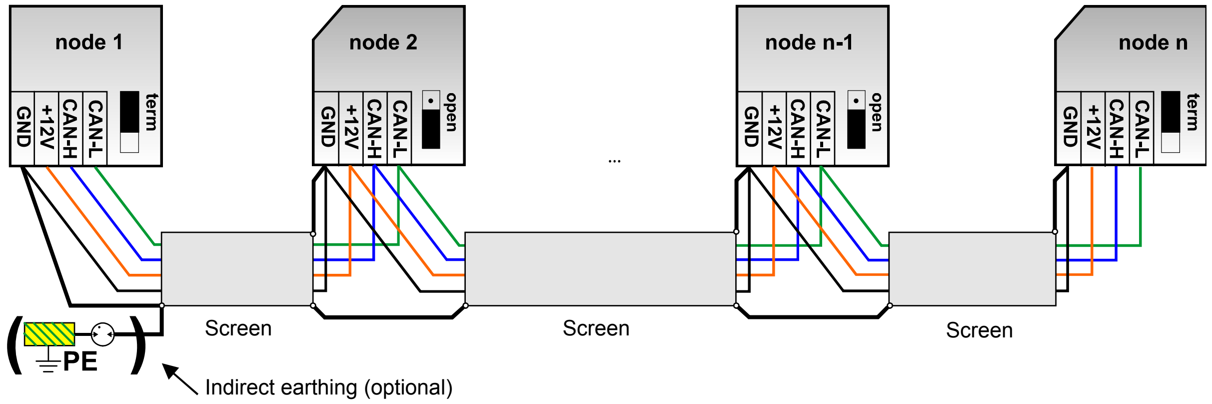
⬠ … device is supplied by the BUS (CAN I/O, CAN-MT, …)

⊟ … CAN BUS converter (CAN-BC/C)

▮ term … terminated (end devices) ▮ open. open termination

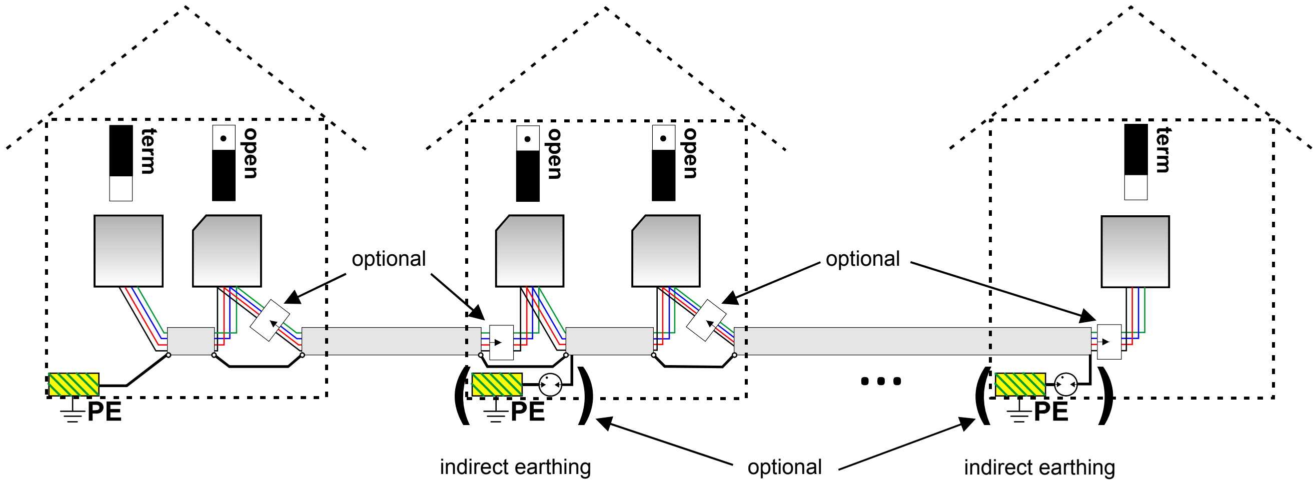
▸ … CAN BUS surge arrester ⊙ .. gas discharge arrester for indirect earthing

<u>"Small" network (within one building):</u>



Maximum cable length: 1000 m with corresponding cross-section
The screen must be continued at each network node and be connected to the device earth (GND).
The screen earthing or GND must only be implemented **<u>indirectly</u>** via a gas discharge arrester.
Ensure that no unintentional **direct** connection of earth or screen and the earth potential is created (e.g. via sensors and the earthed pipework).

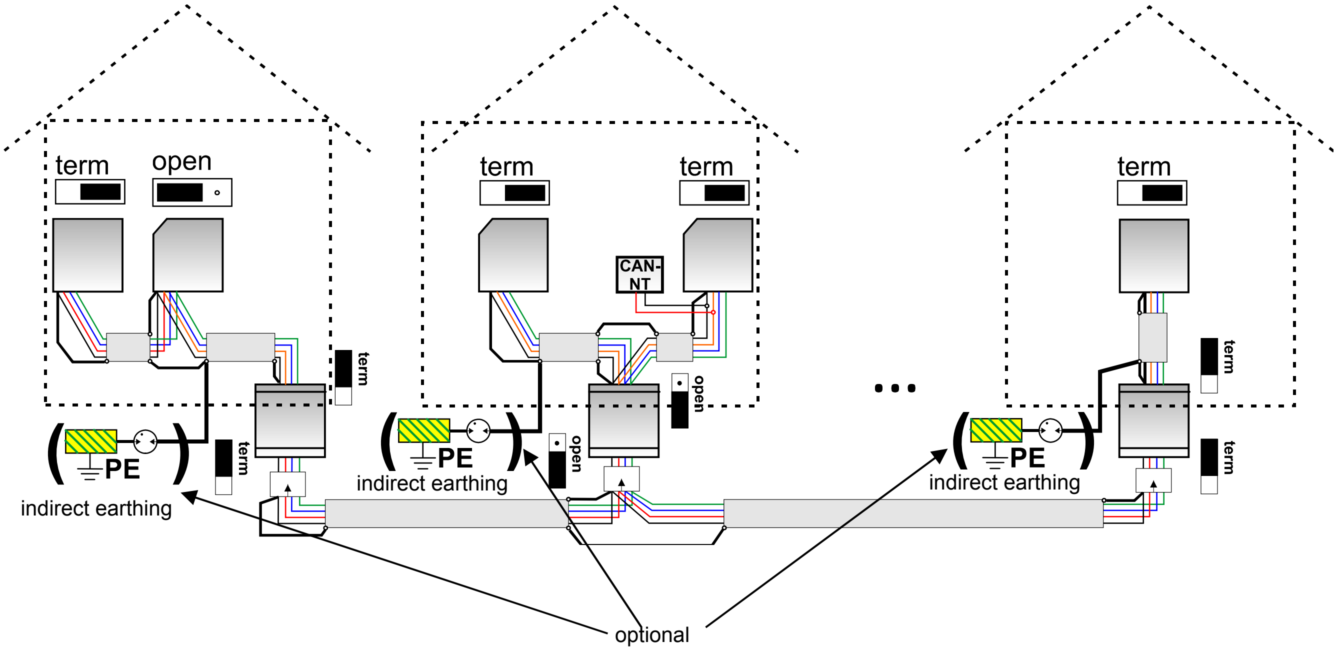
<u>Network (across several buildings) without CAN-BC:</u>



Maximum cable length: 1000 m with corresponding cross-section
The screen must be continued at every network node and be earthed at a **<u>single</u>** point. We recommend earthing the screen **<u>indirectly</u>** in the other buildings using a gas discharge arrester. The screen is **<u>not</u>** connected with the earth (GND) of the devices.

CAN BUS network

Network (across several buildings) with CAN bus converter CAN-BC/C:



Maximum cable length: subject to the set Baud rate CAN-BC/C

The screen of the **disconnected** network is connected at each BUS converter to CAN BUS earth (GND). This screen must **not** be **directly** earthed.
Without CAN BUS surge arresters, this version only protects against potential differences **up to 1 kV**, therefore cannot be considered as lightning protection.
The best solution is the use of CAN BUS converters CAN-BC/L and connection via a fibre-optic conductor, as these neither create electromagnetic interference nor surges.

Routing BUS cables underground

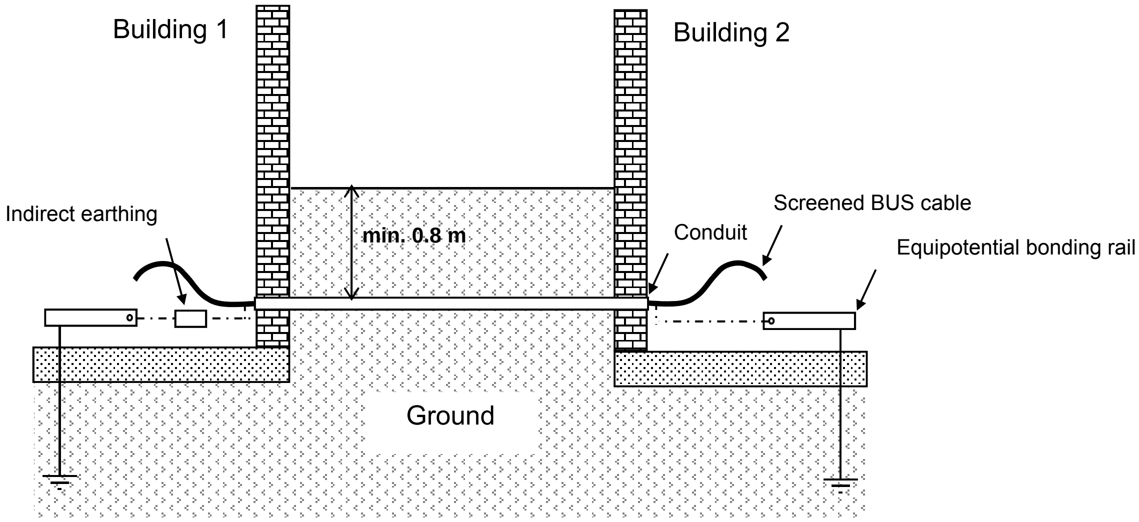
The cable should be routed at a depth of at least 0.8 m (below the frost level) and at least 30 cm clear of other cables, preferably inside a conduit.

According to the SEP principle, all cables (power, data, etc.) should enter the building at a single point in order to avoid potential differences.

No equipotential bonding currents must flow across the BUS screen.

Consequently the screen must only be linked into the equipotential bonding of a **single house** (example: network (across several buildings **without** CAN-BC).

In other houses this must be done **indirectly** via gas discharge arresters.



Cable selection and network topology

Screened **twisted pairs** have proven useful in CANopen networks. These are cables with twisted pairs of conductors and a shared external screen. Such cables are relatively resistant to EMC interference and can still carry 50 kbit/s for up to 1000 m. The CANopen recommendations (CiA DR 303-1) for cable cross-sections are given in the table below.

BUS length [m]	Resistance in terms of length [mΩ/m]	Cross-section [mm²]
0…40	70	0.25…0.34
40…300	< 60	0.34…0.60
300…600	< 40	0.50…0.60
600…1000	< 26	0.75…0.80

The maximum cable length also depends on the number of nodes [n] linked with the BUS cable and the cable cross-section [mm²].

Cable cross-section [mm²]	Maximum length [m]	
	n=32	n=63
0.25	200	170
0.50	360	310
0.75	550	470

Recommendations

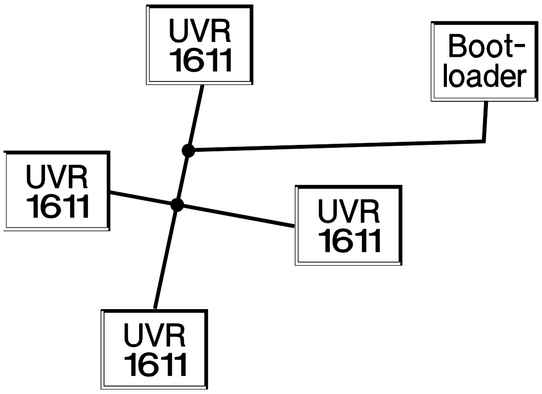
A 2x2-pole, screened twisted pair (twist CAN-L with CAN-H or +12 V with GND) with a cable cross-section of at least 0.5 mm² and a conductor-to-conductor capacity of no more than 60 pF/m and a set impedance of 120 ohms. The standard BUS speed of the UVR16x2 is 50 kbit/s. Theoretically this would enable a BUS length of approx. 500 m to guarantee reliable transmission. This recommendation corresponds, for example, to cable type **Unitronic®-BUS CAN 2x2x0.5** supplied by **Lapp Kabel** for permanent installation **in buildings or conduits.**

For **direct** routing **underground**, earth cable **2x2x0.5** mm² supplied by **HELUKABEL**, part no. 804269, or earth cable **2x2x0.75** mm² supplied by **Faber Kabel**, part no. 101465, would be suitable.

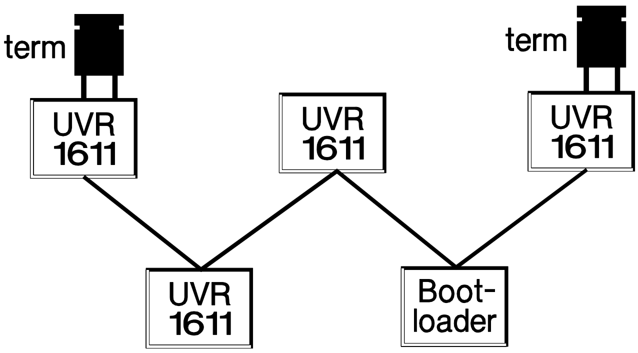
Wiring

A CAN BUS network should never have a **star topology**. Rather, the correct topology is a line from the first device (with terminator) to the second, third and so forth. The final BUS connection again has the termination jumper.

INCORRECT

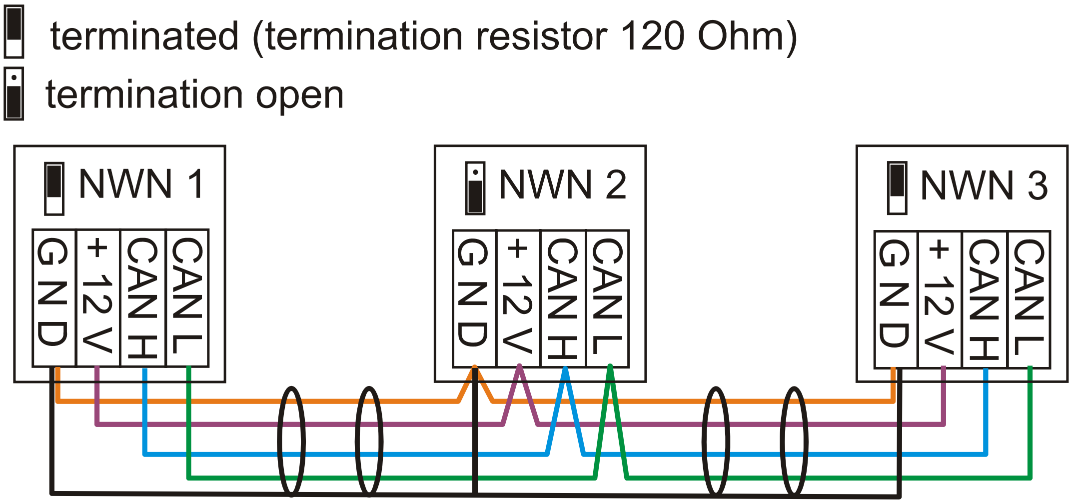


CORRECT



CAN BUS network

Example: Connection of three network nodes (NK) with a 2x2-pole cable and **termination** of the terminal network nodes (network inside one building)



Each CAN network is to be provided with a 120 ohm BUS terminator at the first and last network subscriber (= **termination**). This is achieved with a plug-in jumper **at the back of the controller**. Each CAN network therefore always has two terminators (one at each end). Branch cables or a star topology are not permissible for CAN wiring.

As you can see from the tables, reliable transmission depends on a number of factors (cable type, cross-section, length, number of nodes, etc.). All of this information can be considered relatively conservative so that no problem should occur if you are prudent when dimensioning.

Practical help

As can be seen from the tables, reliable transmission depends on a number of factors (cable type, cross-section, length, number of nodes, etc.).

Trials at the factory have shown the following:

1) Branches with a star topology of up to 10 m will not impede transmission.

2) Up to a BUS length of 150 m and with only a few nodes, the cable **CAT 5 24AWG** (or higher category; typical Ethernet cable in PC networks) can be used. This can therefore be used in many applications.

However, such networks do not comply with the recommended specification and should be tested before the network is created with cables 50 % longer to be on the safe side.

Data link (DL bus)

Any cable with a cross section of 0.75 mm² can be used for the **data link** having a max. length of 30 m. For longer cables, we recommend the use of shielded cable. Cable channels for power and data lines may cause a disturbance in the data lines if they lie too close to each other over long stretches. Therefore a minimum distance of 20 cm between the two cable ducts or the use of shielded cabling is recommended. Where the processing of two control inputs by the data converter are concerned separate screened cables must be used. If screened cables are used, the screen must be connected to the GND. Equally the DL must never be put through the same cable with the CAN.
The data line is connected to clamp DL (A14) and sensor earth.

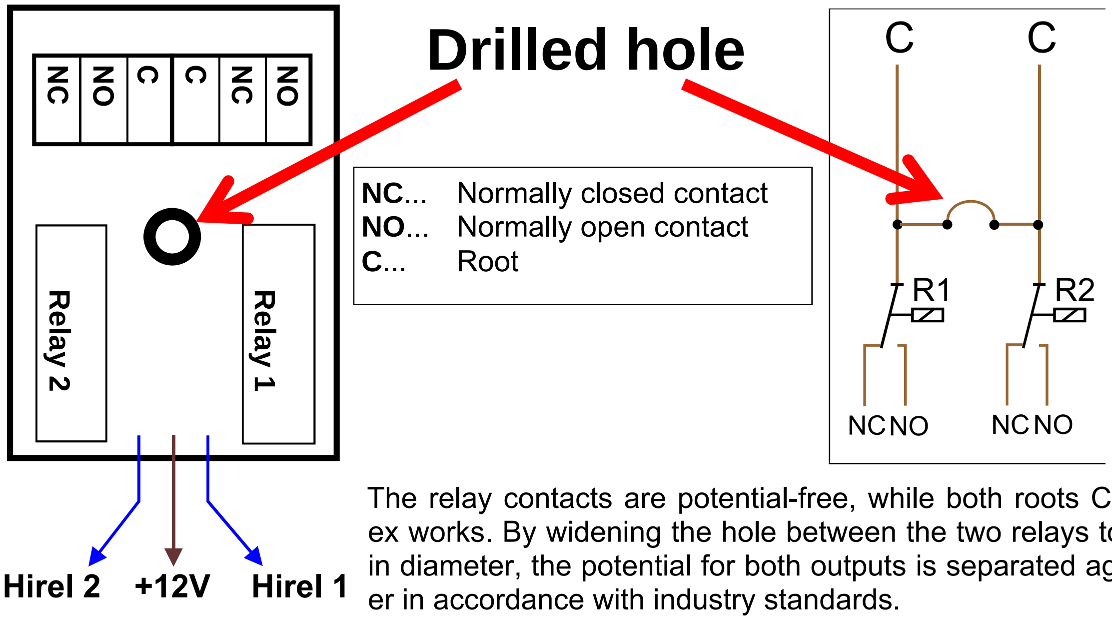
Electrical connection

It should only be made by a professional electrician in accordance with the relevant local guidelines. The sensor lines must not be laid in the same cable as the supply voltage (standard, regulations). In a jointly used cable channel, appropriate shielding and separation have to be provided.

Notice: The system has to be grounded properly and furnished with surge arresters to protect it from damage due to lightening. Sensor failures due to storms and static electricity are usually the result of faulty construction.
Cable channels for power and sensor lines may cause a disturbance in the sensor lines if they lie too close to each other over long stretches. If no fast signals (such as ultrafast sensors) are transmitted, these disturbances can be filtered out by averaging the sensor inputs. However a minimum distance of 20 cm between the two cable ducts or the use of shielded sensor cabling is recommended.

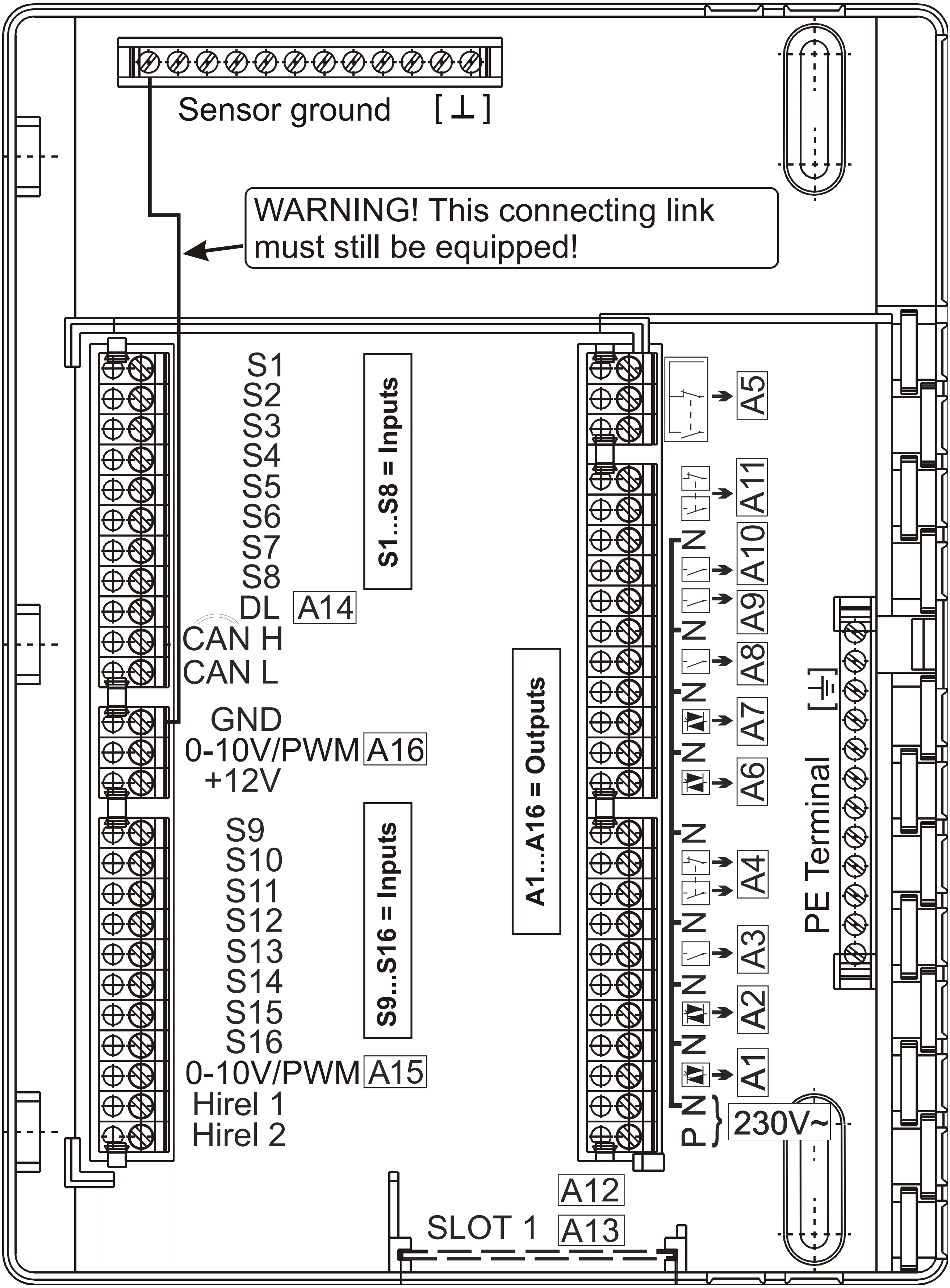
Caution: Only work inside the console with the power cable disconnected. If you assemble the device with the power connected, the device may be damaged.
All sensors and pumps/valves must be connected as they are numbered in the design selected. For grade power, cross sections of 1 - 1.5² fine-strand are recommended except for the feed line. A strip terminal above the inlets is available for the protective conductors. During assembly, this strip terminal can also be removed to make it easier. All cables can be fixed with a clamp (= strain relief) immediately after being connected. A side cutter is needed to remove the clamps; therefore, more parts than necessary are provided. Once all of the power connections have been made (without the protective conductor), the protective conductor terminal is inserted, and the remaining ground wires are connected.
There is only one connection to ground (GND) for all of the sensors on the protective low voltage side. There is thus a mass terminal at the top right of the console, which must be connected to the GND clamp before the sensors are clamped.
All of the sensor lines with a cross-section of 0.5mm^2 can be extended up to 50m. With this length of line and a Pt1000 temperature sensor, the measurement error is approx. +1K. Longer lines or a lower measurement error require an appropriately larger cross-section. One pole for these lines is put through the cable channel the right side of the console and the bridge for the particular terminal, while the second pole is connected to the ground line at the top right.

The universal controller UVR1611 can be upgraded with the HiRel1611. This allows an ex-pansion from 11 to 13 outputs. For this purpose, sliding grooves are intended on the left side of the controller’s console (designated as slot 1 in the controller’s mounting instructions).



The relay contacts are potential-free, while both roots C are connected ex works. By widening the hole between the two relays to at least 6 mm in diameter, the potential for both outputs is separated against each other in accordance with industry standards.

Electrical connection



Caution: Output A5 is potential free – thus not connected with the supply voltage.
Slot 1 is intended for the relay module for two other outputs (A12, 13).

Technical data UVR1611

All sensor inputs	For temperature sensors of type KTY (2 kΩ/25°C), PT1000 and room sensors RAS or RASPT, radiation sensors, voltages up to 5V DC, as well as digital input
Sensor input 8	Additional for current loop (4 -20 mA), voltage (0-10 V DC) or resistance (0-12.50kΩ)
Sensor input 15, 16	Additional pulse input, such as for volume flow encoder
Output 1, 2, 6, 7	Speed-controlled for standard circulating pumps
Output 3, 4, 8-11	Relay outputs, partially with opener and closer
Output 5	Change-over contact relay – **potential free**
Outputs 12, 13	Accommodation for later expansions by a double auxiliary relay module
Output 14	Data link (DL bus) to capture suitable sensors and log data (in special cases configurable with 12V relay as switch output)
Max. bus load (DL bus)	100 %
Outputs 15,16	Analogue outputs 0-10V/20mA or PWM (10V/2kHz)
CAN- Bus	Data 50 kb/sec., power supply for external units 12V= / 100mA
Differential temperatures	Equipped with separate switch-on/off differential
Threshold values	Partially set up with an adjustable hysteresis or as an alternative with separate switch-on/off threshold
Speed control	30 speed stages constitute a quantity change of max. 10 Regulation by absolute value, difference and absolute value to event
Temperature display	-50 until +199°C with a dissolution of 0.1K
Accuracy	Typ. 0.4 and max. +-1K for the range of 0 - 100°C
Max. breaking capacity	**A1: 230V/0,7A**, A2, 6, 7: each 230V/1A relay outputs max. each 230/ 3A
Connection	230V, 50- 60Hz, (outputs and units are fused along with 6.3A quick-acting)
Supply cable	3x 1mm² H05VV-F conforming to EN 60730-1 (corresponding cable with safety plug contained in the basic package)
Power draw	Max. 4.2 W (without accessory equipment)
Protection	IP40
Permissible ambient temperature	+5 to +45°C

Standard delivery

UVR1611K: The UVR1611, console including all terminals, while attachment material, 2 ground terminals, 16 strain reliefs, operating instructions

UVR1611S: Device with rear panel shaped as bushings, 2 ground terminals, 2 3-pin and 4 11-pin plug-in terminal screws, operating instructions.

Accessories

TAPPS (Technische Alternative planning and programming system):
Software to make programming the UVR1611 easy on a PC (graphic interface via function modules).
The software is available at our homepage http://www.ta.co.at as a free download.
The C.M.I. is required to transfer the data from the PC to the controller!

Hirel 1611
Expansion of the universal controller to include two potential-free outputs (A12, A13).
Order designation: 01/HIREL1611

CAN I/O module 44 and CAN I/O 35
Expansion of the universal control by three relay outputs, one analogue output (0-10V) and four inputs (CAN I/O 44) or three relay outputs, two analogue outputs and three inputs (CAN-I/O 35).
Order designation: 01/ CAN-I/O 44 and 01/CAN-I/O 35

CAN Monitor
UVR1611 units for room sensor, display, and operation.
Same operating concept as the control, communication via CAN bus.
More than one CAN Monitor can access a controller just as one CAN monitor can access several controllers in the network.
Order designation: 01/CAN-MT

CAN Touch
10" LCD screen with touch-sensitive surface.
Display and operating device for UVR1611 and CAN-I/O modules. Incorporation of temperature and humidity sensors is possible. Programming with software ***TA-Designer***.
Order designation: O1/CAN-TOUCH

Interface C.M.I.
To back up data, update operating systems, and log data
1) Back up all of the function data for the UVR1611 on a PC and second backup.
2) Update of the UVR1611's operating system
3) Data logging of temperatures and output conditions via DL and CAN bus
4) Ethernet interface for direct access to CAN bus subscribers via a browser
Order designation: 01/CMI

Simulation board
In combination with a UVR1611K for programming and simulations (every input can be simulated from -10°C to +125°C, and digital simulation is also possible for input 15 and 16).
Order designation: 01/SIM-BOARD1611

CAN bus converter
Two CAN bus interfaces, optionally available in optical waveguide model
EIB or KNX interface; M bus interface.
Order designation: 01/CAN-BC/C, 01/CAN-BC/E or 01/CAN-BC/L

The manuals for the products are available for download at our homepage: **http://www.ta.co.at**.

Tips on troubleshooting

If there is **no display**, there has been a power outage. First check the fuse (6.3A, quick-blow) that protects the device and the outputs (pumps, valves, etc.) from short circuits and the outputs in connection with the integrated overvoltage protection. The glass tube fuse is located on the rear side of the controller behind a screw cap.

Realistic temperature values but outputs not operating properly indicate false settings or connections. If it is possible to switch the outputs on and off in the manual mode, the device is in order, and all of the settings and the terminal should be checked.
Do endurance runs and standstills lead to the same reaction at the output? In other words, does this pump really run if the solar pump is activated manually, or does the heating circuit pump go into operation instead of the solar pump?
Are all of the sensors connected with the right terminal (heat up the sensor using a cigarette lighter and check the temperature display)?

If you still cannot find any error in the system, install a data logger (Bootloader or D-LOGG) in the system and record the temperature curve and switching statuses. **For data logging, output 14 must be set to "Data line" using the DL bus**.

Improper temperatures can have the following causes:

- If a value such as -999 is displayed when a sensor short-circuits or 9999 if there is a sensor interruption, the cause may not be a material or terminal error. Is the proper sensor type selected in the entry menu (KTY, PT1000, SPS, GBS, etc.)?
- The sensor can also be checked without a measuring device simply by changing the sensor that is probably defective with one that works at the strip terminal and checking the temperature display. If the error is reproduced, the problem is probably the sensor. However, if the problem remains on the same input of the device, the error is either due to the settings for sensor type, or the input itself is defective (such as defective overvoltage protection).

The following values should be found if the sensors are checked with an ohmmeter.

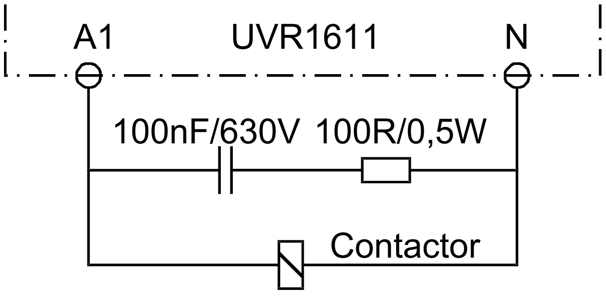
Temp. [°C]	0	10	20	25	30	40	50	60	70	80	90	100
R(KTY)[Ω]	1630	1772	1922	2000	2080	2245	2417	2597	2785	2980	3182	3392
R(PT1000) [Ω]	1000	1039	1078	1097	1117	1155	1194	1232	1271	1309	1347	1385

If the sensor is defective, pay attention to the type of sensor you exchange it with. While it is possible to use a different type of sensor, the parameters for that input also have to be set to the type used.

Not possible to manually switch an output:

- If the output has speed control (A1, A2, A6 or A7) and is actually set to speed control, pay attention to the speed stage when using manual mode. Set to stage 30 when testing the pump's basic function.
- **Electronic pumps cannot** be operated speed-controlled due to their internal structure. Connection to one of the outputs A1, A2, A6 or A7 as a **switch output** is possible. However it is recommended to connect these pumps to one of the relay outputs (A3 - A5, A8 - A11).
- If a valve or contactor is to be triggered using a speed-controlled output (also along with a pump), the parameters for this output have to be set as a switch output because the speed control cannot work on such a consumer!

- Speed-controlled outputs may not be able to switch **small loads** reliably (< 5 W, such as valves, contactor, etc.). This is especially true for output A1 with its integrated power line filter, which can only be operated with a minimum load ≥ 20 W. If the speed-controlled outputs (A2, A6, A7) **only** control small loads, an additional parallel load or the subsequent RC component is necessary to make switching reliable (available as a special accessory)



- Make sure that outputs 5, 12, and 13, are potential-free and do not have power. It is thus only possible to directly switch a 230V consumer when the proper wiring has been made.
- If it is not possible to switch an output on or off in manual mode because the cursor could not be positioned next to the proper parameter, there are the following two options:
  - A message is currently active and switches the relative output dominant ON or OFF (display of message in the function overview). In this case, manual mode is not possible.
  - An expert has set the setting User interlock (outputs) to yes. By this, only experts can manually operate the outputs.

Troubleshooting - hardware

If the cause of the error is clearly defective hardware, please send the equipment to your retailer or manufacturer for repairs. Please do not forget to describe the error when sending back the device (simply saying "the device does not work, please repair" is insufficient). Only then can the control system be repaired quickly and inexpensively.

Troubleshooting - programming

The manufacturer can help you find a remedy if the proper documentation and data are provided. The following are indispensable:

- A hydraulic diagram by fax (best option) or e-mail (WMF, JPG, ENG)
- Complete programming by means of TAPPS files or, at least, the function data by e-mail
- Operating system version for the control system
- All existing LOG files or at least the (temperature) values of the inputs at the time the system malfunctioned
- A telephone call to describe the problem - a written description of the error does often not suffice, and the manufacturer cannot accept it.

Troubleshooting in the CAN network

To localize the error, switching off of a part of the network at a time in order to see when the error disappears is recommended.

General tests:

- Node numbers - no node number may be allocated twice
- Power supply to the bus members (if necessary use the power pack CAN-NT)
- Setting of the baud rate (only if using the CAN bus converter CAN-BC/C)

Cabling tests:

For these tests all nodes must be switched off!

- Resistance between CAH-H and CAN-L
  - If this is above 70Ω, it indicates a missing terminating resistor.
  - If the resistance is below 60Ω, then search for extra terminating resistors or short circuits between wires.
- Check for a short circuit between GND or the shielding and the signal wires.
- Checking for earth loops - to do this the shielding is disconnected at the relevant node point and the connecting current measured. If a current exists, then an earth loop exists courtesy of an unwanted earth connection.

Information on the Eco-design Directive 2009/125/EC

Product	Class[1, 2]	Energy efficiency[3]	Standby max. [W]	Typ. power consumption [W][4]	Max. power consumption [W][4]
UVR1611[5]	max. 8	max. 5	2.4	2.03 / 3.73	2.4 / 4.2

[1] Definitions according to Official Journal of the European Union C 207 dated 03/07/2014

[2] The classification applied is based on optimum utilisation and correct application of the products. The actual applicable class may differ from the classification applied.

[3] Contribution of the temperature controller to seasonal central heating efficiency in percent, rounded to one decimal place

[4] No output active = standby / all outputs and the display active

[5] The class is defined on the basis of the programming of the heating circuit controller, in accordance with the Ecodesign Directive.

We reserve the right to make technical changes.

EC- DECLARATION OF CONFORMITY

Document- Nr. / Date: TA12006 / 19.11.2012

Company / Manufacturer: Technische Alternative elektronische SteuerungsgerätegesmbH.

Address: A- 3872 Amaliendorf, Langestraße 124

This declaration of conformity is issued under the sole responsibility of the manufacturer.

Product name: UVR1611K, UVR1611S, UVR1611E-NM, UVR1611E-DE, UVR1611E-NP

Product brand: Technische Alternative GmbH.

Product description: Freely programmable universal controller

The object of the declaration described above is in conformity with Directives:

2006/95/EG	Low voltage standard
2004/108/EG	Electromagnetic compatibility
2011/65/EU	RoHS Restriction of the use of certain hazardous substances

Employed standards:

EN 60730-1: 2011	Automatic electrical controls for household and similar use – Part 1: General requirements
EN 61000-6-3: 2007 +A1: 2011	Electromagnetic compatibility (EMC) - Part 6-3: Generic standards - Emission standard for residential, commercial and light-industrial environments
EN 61000-6-2: 2005	Electromagnetic compatibility (EMC) - Part 6-2: Generic standards - Immunity for industrial environments

Position of CE - label: On packaging, manual and type label

CE

Issuer: Technische Alternative elektronische SteuerungsgerätegesmbH.
A- 3872 Amaliendorf, Langestraße 124

This declaration is submitted by

TECHNISCHE
ALTERNATIVE
ELEKTRONISCHE STEUERUNGSGERÄTEGESELLSCHAFT M. B. H.
A- 3872 AMALIENDORF, LANGESTRASSE 124
TELEFON 0 28 62 / 536 35, FAX 0 28 62 / 536 35-7

Kurt Fichtenbauer, General manager,
19.11.2012

This declaration certifies the agreement with the named standards, contains however no warranty of characteristics.

The security advices of included product documents are to be considered.

Guarantee conditions

Note: The following guarantee conditions do not in any way limit the legal right to a guarantee, rather expand your rights as a consumer.

1. The company Technische Alternative elektronische Steuerungsgerätegesellschaft m. b. H. provides a two-year guarantee from the date of purchase by the end consumer for all the devices and parts which it sells. Defects must be reported immediately upon detection and within the guarantee period. Technical support knows the correct solution for nearly all problems. In this respect, contacting us immediately will help to avoid unnecessary expense or effort in troubleshooting.

2. The guarantee includes the free of charge repair (but not the cost of on site fault-finding, removal, refitting and shipping) of operational and material defects which impair operation. In the event that a repair is not, for reasons of cost, worthwhile according to the assessment of Technische Alternative, the goods will be replaced.

3. Not included is damage resulting from the effects of overvoltage or abnormal ambient conditions. Likewise, no guarantee liability can be accepted if the device defect is due to: transport damage for which we are not responsible, incorrect installation and assembly, incorrect use, non-observance of operating and installation instructions or incorrect maintenance.

4. The guarantee claim will expire if repairs or actions are carried out by persons who are not authorised to do so or have not been so authorised by us or if our devices are operated with spare, supplementary or accessory parts which are not considered to be original parts.

5. The defective parts must be sent to our factory with an enclosed copy of the proof of purchase and a precise description of the defect. Processing is accelerated if an RMA number is applied for via our home page www.ta.co.at. A prior clarification of the defect with our technical support is necessary.

6. Services provided under guarantee result neither in an extension of the guarantee period nor in a resetting of the guarantee period. The guarantee period for fitted parts ends with the guarantee period of the whole device.

7. Extended or other claims, especially those for compensation for damage other than to the device itself are, insofar as a liability is not legally required, excluded.

Legal notice

TECHNISCHE ALTERNATIVE
elektronische Steuerungsgerätegesellschaft m. b. H.
A-3872 Amaliendorf Langestraße 124
Tel +43 (0)2862 53635 Fax +43 (0)2862 53635 7
E-Mail: mail@ta.co.at --- www.ta.co.at ---