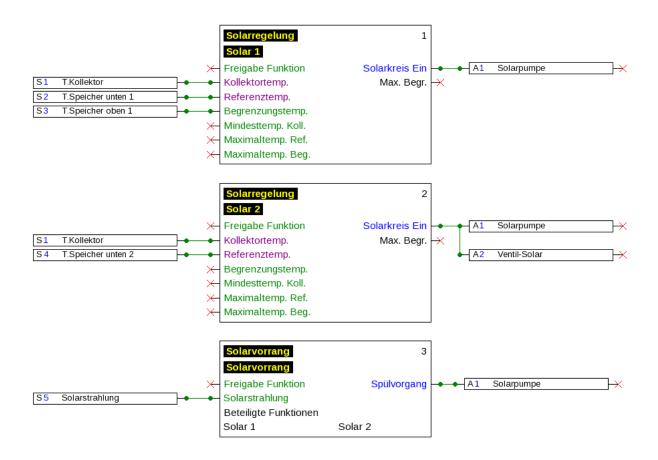


# TAPPS2 PROGRAMMING SOFTWARE Version 1.20



# **Table of contents**

Description	
Menu overview	. 6
File menu	6
Edit menu	6
View menu	7
Object menu	7
Extras menu	7
Help menu	7
Language	
Tool bar	
Tool bar on the right	
Creating, opening, saving files	
Generating files	
Opening existing files	
Saving files	
Drawing Interface	
View menu	
Zooming	
Zooming with the scroll wheel	
Zooming with tools from the tool bar	
Moving the drawing area	
Importing function data (*.dat)	
Programming	14
Procedure of programming	14
Function data objects	
General information	
Pasting obejcts into the drawing	
Selecting	
Selecting several objects	
Moving objects	
Undo / Redo	
Signal transfer and acceptance	
Moving object to the back or the front	
Aligning objects in the drawing	
Function data objects for UVR1611	
Application of user-defined designations	
Double designations	
Outputs	
Inputs	
Sensor status	
Network outputs	
Network inputs	
Network status	
Functions	
Function data objects for devices with x2 technology	
Managing designations, creating user defined designations	
Double designations	
Global user defined designations (multilingual)	
Applying the translated designations	
Creation of translations for non-defined languages	
Inputs	
CAN outputs	
CAN outputs	
DL outputs	
	00

# Table of contents

DL inputs	
Fixed values	
System values	
Functions	
CORA devices	
Links	
Direct links between 2 objects	
Changing the line shape	
Editing lines	
Moving corner points	
Changing a line	
Moving a line	
Branches	
Deleting objects and lines	
Command "Select links"	
Invalid links	
Network input – output	
Linking two output variables	
Single network output on multiple functions	
Several network inputs refer to one network output of another CAN-Bus device $\ldots$ .	
Texts	
Text insertions	
Global text formatting	
Graphics/Images	
Copying objects and texts	
Copying within a drawing	
Copying between two drawings	
Paste as CAN-Inputs	
Deleting objects	
Cutting objects	
Inserting simple drawing objects	
Finding objects	
Personalised libraries	
Shared libraries	
Settings	
Device settings	
Sort functions and messages	
Sorting CORA devices	
Sorting inputs/outputs	
Function overview UVR1611	
Function overview UVR16x2	
CAN data logging	
x2 devices	
Datalogging on the SD card of the UVR16x2	
UVR1611	
Simulation	72
Example	
Changing values	
Analogue values	
Digital values	
Pulse	
Temporal mean value calculation	
CAN bus simulation	
Date/time/Mean value calculation	
Buttons	
Generating function data and documentation	
	70

# **Table of contents**

Function data	78
Exporting a screenshot	
Drawing function under "Hydraulics"	80
Libraries	80
Standard libraries	80
Personalised libraries in hydraulic drawings	
Shared libraries in hydraulic drawings	
Creating and editing library elements	82
Icon editor	82
Selection mode	
Editor mode	84
Duplicating elements	85
Creating a hydraulic drawing	86
Example of a simple hydraulic drawing	88

# Description

**TAPPS2** is a vector-based drawing program which was developed for planning and programming of the UVR1611 and UVR16x2 controllers, the RSM610 control module, CAN-I/O45 and CAN-EZ2.

A configuration created with **TAPPS2** can be converted into a data format that can be loaded to the controller via the C.M.I. interface or the SD card of the UVR16x2 or the CAN-MTx2 CAN monitor.

This manual only describes the necessary tools and procedures that are required for the graphical creation of programming or hydraulic drawings.

Neither the principles of control technology nor a precise description of the individual function modules are covered by this manual. Please refer to the controller operating instructions for information on function modules and their operation.

# Menu overview

# File menu

File Edit View Obje	ct Extras H	
New	Ctrl+N	Creating a new file
Open	Ctrl+O	<ul> <li>Opening an existing file</li> </ul>
Close		<ul> <li>Closing one or several files</li> </ul>
Close all		<ul> <li>Saving one or several files</li> </ul>
Save	Ctrl+S	Settings
Save as		<ul> <li>Page set-up, page view, print</li> </ul>
Save all		
Settings	>	Importing of small graphics/images
Deservation		<ul> <li>Importing function data</li> </ul>
Page setup		<ul> <li>Exporting function data and documentation</li> </ul>
Page view		<ul> <li>Display of projects opened last</li> </ul>
Print	Ctrl+P	
Import	>	
Export	>	
Files last opened	>	
Exit	Alt+F4	

# Edit menu

Edit	View	Object Extras	
	Undo	Ctrl+Z	Undo / Redo processing steps
	Redo	Ctrl+Y	Cut / Copy / Paste / Paste CAN outputs as correspondingly converted
	Cut	Ctrl+X	CAN inputs / Delete selected objects
	Сору	Ctrl+C	<ul> <li>Finding objects / Select all links of the selected object</li> </ul>
	Paste	Ctrl+V	
	Delete		
	Find	Ctrl+F	
	Select al	I Ctrl+A	

# View menu

View	V Object Extras Help
~	Page number
~	Page grid
~	Drawing grid

## **Object menu**

Object	Extras Help					
An	range	>				
Ali	Align >					
Lin	Line and fill					
Fo	nt					

- Selection of display of page number, page grid and drawing grid
- Arrange and align objects
- Line type and fill selection (global selection for the entire **hydraulic** drawing and for drawing elements under programming)
- Font selection (global text formatting)

# Extras menu

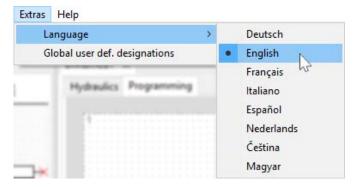
Extras Help		
Language Global user def. designations	>	<ul><li>Language selection</li><li>Generating multilingual user defined designations</li></ul>

# Help menu

Hel	р		
	Manual	F1	<ul> <li>Display of the manual</li> </ul>
	Info about Tapps2		Information about TAPPS2 version

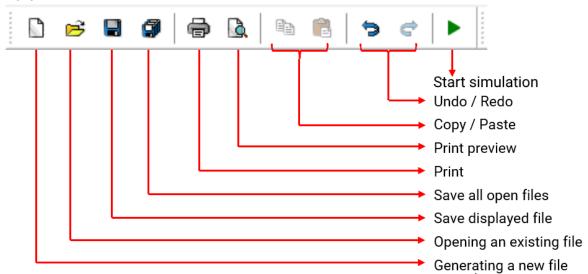
# Language

A number of languages are available for selection. Select **Extras \ Sprache** (Extras\Language) and click on the required language. *TAPPS2* must be restarted for the language choice to come into effect.

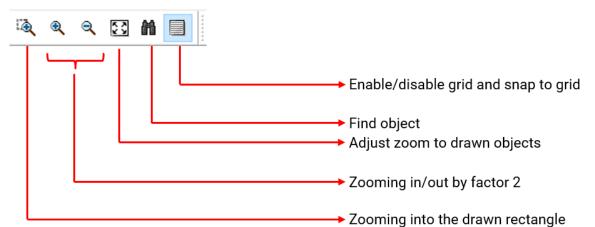


# Tool bar

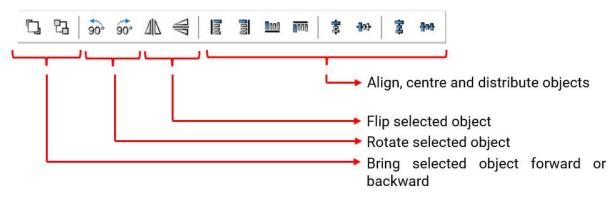
Frequently used actions can be started with a single mouse click on one of these icons. **Part 1:** 



#### Part 2:



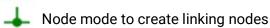




# Tool bar on the right



Selection mode for inserting objects, setting object parameters and creation of link lines



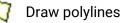
Editing mode to edit lines



Text mode to paste and edit texts



Draw polygons

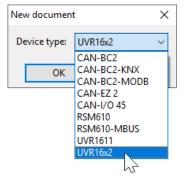


Draw rectangles

# Creating, opening, saving files

## **Generating files**

A new file can be generated with icon 🗋 from the tool bar or **File / New...**. The device type is defined in the following window:



# **Opening existing files**

An existing file (\*.tdw) can be opened with icon 🥩 from the tool bar or File / Open...

Several files can be opened simultaneously. The opened files are displayed in the tab, at the top above the drawing area. The drawing area which is currently visible is highlighted.

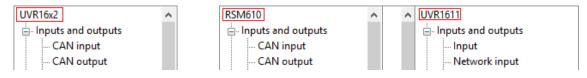
Example: Programming Solar X is currently displayed

Unnamed1	Test	Solar X	х	
Hydraulics	Progra	mming		

Below this, you can switch between the hydraulic drawing and programming for the file displayed.

With **File / Files opened last**, the required file can be selected from a list of the files most recently opened.

The associated controller type can be seen in the search tree when "Programming" is selected:



# **Saving files**

The displayed file can be saved with 🔚 from the tool bar or **File / Save**.

If no name has been allocated to the file yet, a name will be allocated during the first save. All opened files can be saved with **(a)** or **File / Save all**.

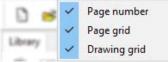
Saving changes should also be carried out **regularly** during work in order to avoid a greater data loss in the case of malfunctions (computer crash, power failure).

File / Save as... can be used to save an opened file with another name and edit it further.

# **Drawing Interface**

# View menu

File Edit View Object Extras Helf The settings in the **View** menu can be used to structure the drawing interface.



**Page grid** and **page number**: This distribution and numbering facilitates a clear print of the program. You can thus take into consideration the limits of the individual pages while drawing the program and prevent overlapping of objects across page margins.

**Drawing grid**: The drawing grid achieves a clear arrangement of objects and safe linking of the objects with the link lines. The objects and lines are aligned along the grid. Snapping the lines to the linking points is facilitated by the automatic snap mode.

The grid can also be switched on or off in the tool bar:

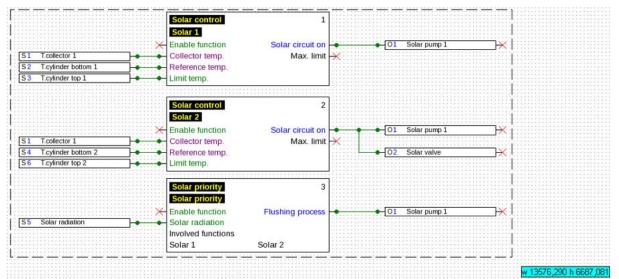
# Zooming

## Zooming with the scroll wheel

The scroll wheel enables fast and easy zooming whereby the position of the cursor is the fixed point of the zoom.

## Zooming with tools from the tool bar

Clicking on this symbol changes the cursor. You can now draw a rectangle across a group of objects which are then zoomed to the size of the drawing interface.

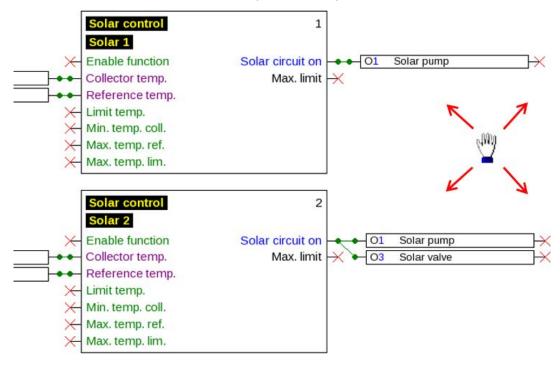


< If you click on one of these symbols, the drawing is zoomed in or out by a factor of 2. If a drawing area is selected, the fixed point is the centre of the selected section; without selection, the centre of the display area.

With the help of this tool, you can adjust the view to the size of the drawn program.

# Moving the drawing area

The drawing area can be moved in any direction by holding down the **right mouse button**.



# **Importing function data (\*.dat)**

File E	dit View Object	Extras H	Help				
	ew	Ctrl+N	2.1	11	•	1	
	pen lose	Ctrl+0	Unname	di Ur	named	ж	
C	lose all		Hydrau	fics P	ograma	ning	
Sa	ave	Ctrl+S	1				
	ave as						
	ave all						
Se	ettings	>					
	age setup						
	age view						
Pi	rint	Ctrl+P					File / Import / Function data allows function
In	nport	>	Fu	nction o	lata		data (*.dat files) to be pasted into a drawing.
Ex	nort	>			-	hr	

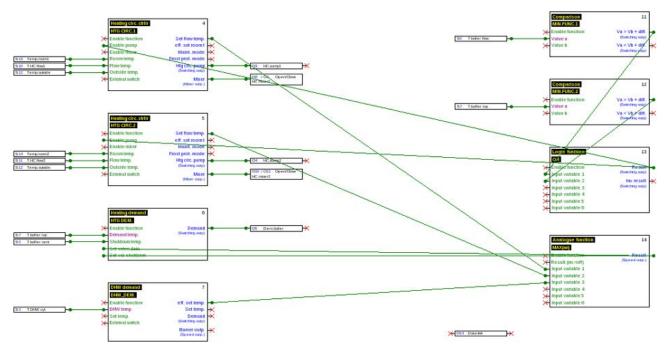
**Instructions for function data for UVR1611:** In order to be able to import function data into TAPPS2, these must have been created with TAPPS V1.25 or higher. To be able to import older programs (\*.eng/\*.par) into TAPPS2, their function data (\*.dat) must have been re-created with TAPPS 1.29.

Any **default settings** (function overview UVR1611, CAN data logging, device settings, etc.) are overwritten. The functions are embedded in the new drawing **page by page**. The **sequence** of functions is sorted in numerical order.

When copying old TAPPS programs to an UVR 1611, it is therefore sensible to check the sequence. With **Sort functions...** sequences can be changed in TAPPS 1.xx in order to optimize the arrangement in TAPPS 2.

If the original program used **signal transmissions** and **transfers**, then they will now be displayed as link lines.

#### **Example** of a function data import:



# Programming

Function data is created under "**Programming**", which is displayed under the file name. **Example:** 

System 🗙	
Hydraulics	Programming

# Procedure of programming

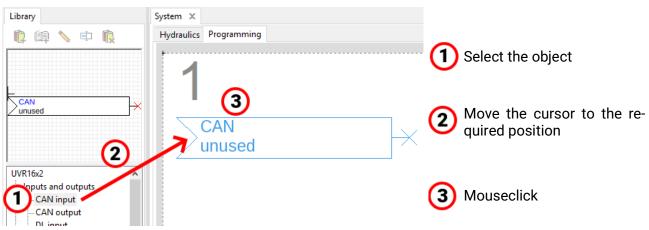
- 1. Generating a new file 🗅 opening an existing file 🖻 for further editing.
- 2. Paste required elements (inputs/outputs, functions, etc.) into the drawing and position accordingly.
- 3. Set input and output parameters.
- 4. If available: set network input and output parameters.
- 5. Link these elements graphically with lines.
- 6. Set function and message parameters.
- 7. Make settings (depending on controller type: device settings, sort functions and messages, create function overview, adjust CAN data logging)
- 8. With **Export** from the logical circuit diagram, create the \*.dat file (function data) for the controller.
- 9. Transfer the function data via interface (BL-NET or C.M.I.) or SD card to the controller.

# Function data objects General information

# Pasting obejcts into the drawing

Objects can be pasted into a drawing in two ways:

1. Directly from the search tree



#### 2. From the icon preview (with drag & drop)

Library	System X	Calact the object
🛱 🕼 📏 🗉 🕵	Hydraulics Programming	1 Select the object
CAN unused	1	O Mouse-click into the pre- view field
UVR16x2 ^ UVR16x2 ^ Outputs and outputs -CAN input -CAN output	CAN unused	Move the cursor to the re- quired position while hold- ing down the mouse button
DL input DL output Fixed value		<b>4</b> Release the mouse button

All objects, including functions, can be embedded in the drawing in the same way.

## A double click on an object opens its parameter menu.

## Selecting

A **simple click** on the object in the drawing selects it. The object is then coloured red. It can now be moved or aligned.

Example: Output



#### Selecting several objects

- 1. Selecting several objects in sequence whilst holding down the Shift key.
- 2. Dragging a frame whilst holding down the left mouse button can select an entire group of objects:

51 Tcolledor 1 52 Tcyfindir bolton 1 53 Tcyfindir top 1	Solar control Solar 1 Collector temp. Collector temp. Limit temp.	Solar circut on + - 03 3 Max.limit +:		S1 Fasilesia 1 S2 Cycleate actions ) S3 Concerning 1	Solar control Solar 3 Heatle function Collector temp.	1 Solar chock on Max.lime →:
S1         T collector 1           54         Explinitier boltom 2           56         T cylinitier top 2	Solar control Solar 2 Chable function Collector temp. Reference temp.	Max. limit 兴	istiur pump 1	51 [raiedar1 [64 [cyclet totim7]	Solar control solar 2 Enable function Collector temp. Reference temp.	2 Solar cacut on Max. limit +( <u>01</u> Solar pump 1 +( <u>01</u> Solar pump 1 +( <u>02</u> Solar strim
55 Solernediation	Solar peterny Solar peterny Solar radiation Involved functions Solar 1	3 Flushing process Solar 2	islari purno 1	[51 Soar ratation	Solar actions Solar actions Solar relation Solar radiation Involved functions Solar 1	3 Flucting process Flucting Flu

### **Moving objects**

**Selected** objects or object groups can be moved with the mouse button held down or with the arrow keys of the keyboard. With an active grid, the objects are aligned on the grid after moving them.

#### Undo / Redo

The tools 🦻 🗢 in the toolbar make it possible to undo or redo programming steps.

### Signal transfer and acceptance

A logical connection can thus be realised without having to draw the appropriate line of a link across the entire page of the drawing.

#### Example: Signal transfer



Window for entering the signal name following a **double click** 

Signal name		×	N	Signal name		×
Signal name:	Signal ID_xy	~		Signal name:	Set room temp.	~
	ОК	Cancel			ОК	Cancel

Overwriting the predefined signal name "Signal ID\_xy" with the required name and complete with **OK**.

Set room temp.

View of signal transfer after entry of the name

#### Signal acceptance



Entry window for signal name after double clicking

Signal name	×	- K	Signal name		×
Signal name:	Signal ID_xy 🗸	$\square$	Signal name:	Set room temp.	~
	Signal ID_xy			ОК	Cancel

Selection of the saved name and completion with **OK**.



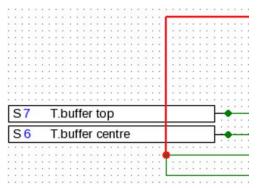
View after selecting the name

If multiple objects in the drawing are selected, the signal name for all selected signal objects is defined in the dialogue.

### Moving object to the back or the front

This tool in the tool bar makes it possible to move objects in front or behind each other. **Example**: A line crosses inputs and should be behind the inputs after drawing.

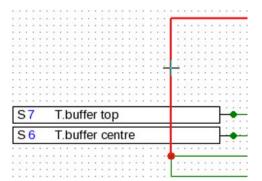
1. Select the line



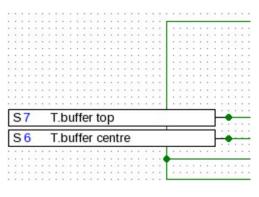
2. Select the tool in the tool bar (Selected objects to the back)



3. Click on the line with the changed cursor (+)



4. The view has changed:



Tapps2

File Edit View	Object Extras He	elp	
0 ≤ 🖬	Arrange	>	To the front
Library	Align	>	To the back
10 ED N	Line and fill Font		s Programming

This action can also be carried out with **Object / Ar**range / To the back.

## Aligning objects in the drawing

With the help of these functions, is it possible to align objects according to specific criteria. **Example**: Left adjust 3 inputs

1. Select the group of objects to be aligned (hold down Shift or with a selection frame)

60			_					
32	T.cylinder b	ottom	1					
linder	top 1		H	•	 •	•	-	
		linder top 1				· · · · · · · · · · · · · · · · · · ·		

The selection of the object that is **align left** (S3) is selected automatically. The group is aligned on this specially selected object.

If the alignment is to be on **another** object, then this object is selected with two **individual** mouse-clicks **whilst holding down Shift**.

Example: Alignment on sensor 2:

1		1.11				S	1	T.c	ollector 1				]	•	•		8	8 14	* *
-	•	1 1 1 1	19 19 19	10	10 - 10 - 10 - 10 - 10 - 10 - 10 - 10 -	)		<b>S</b> 2	T.cylinder botto	m 1							}		¢
[	S	; ;	3			Τ.	C	ylinde	r top 1		•	зя Isa	54 54	э 2	э х	а а	•	э Э	а ж

The sensor S2 is now highlighted.

2. Selection of the alignment method (in this example: "Align left on the object selected last") with a mouse-click



3. The sensors were aligned on sensor S2:

<b>S</b> 1	T.collector 1	
S2	T.cylinder bottom 1	
<b>S</b> 3	T.cylinder top 1	

This action can also be carried out with "Object / Arrange / Align Left".

# Function data objects for <u>UVR1611</u> Application of user-defined designations

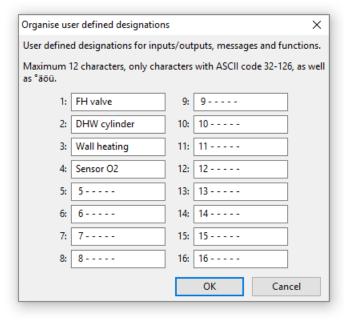
Dutputs	- Output 13	×
	Drawing	object: Output 13 🗸 🗸
Links	Parameters An	ti-blocking protection
	Des. group:	User def. v
	Designation:	~ ~
	Туре	FH valve
	Ounused	DHW cylinder Wall heating Sensor Q2 0 sec
	Switching ou	5
	<ul> <li>Speed contro</li> <li>Mixer</li> </ul>	7
	O Data link	8 9
		11
		12 13 14
	ОК	OK, without allocation Cancel

A click on this field calls up management of user-defined designations.

Up to 16 designations can be defined, which can be allocated to inputs, outputs, messages or functions as required.

These can be selected from the list following the definition of the required designations.

Example: Management of user-defined designations



The designation must **not exceed 12 characters** and must not contain special characters or umlauts.

However, for **functions** displayed on the UVR1611 **only 9** of the maximum 12 possible characters can be displayed.

#### **Double designations**



If a designation is selected for an object that has already been allocated before, a **warning triangle** will be displayed.

### Outputs

A **double click** on the object takes you to the parameter menu. The drawing object will be initially deemed to be **unused**. After assigning an output number, the standard parameters are offered for selection.

#### Example: Output 1, Solar pump 1

Outputs	- Output 1 - So	lar pump1			×			
	Draw	ing object: (	Dutput 1	$\sim$				
Links	Parameters	Parameters Anti-blocking protection						
	Des. grou	ip: General	× .					
	Designat	ion: Solar pu	imp ~ 1	$\sim$				
	Туре		Mode:		$\sim$			
	Ounused		Delay:	0 sec	~			
	Switching	g output	Delay.	U SEC	×			
	○ Speed co	ntrol outp.	Run-on:	0 sec	$\sim$			
	<ul> <li>Mixer</li> </ul>		Runtime:		$\sim$			
	🔿 Data link		UVR1611E:		$\sim$			
			Netw.in.=>DL:		$\sim$			
		ОК	OK, without alloca	tion	Cancel			

Outputs -	Output 1 - So	lar pump1			×
	Draw	ing object:	Output 1	~	
Links	Parameters	Anti-block	ing protecti	on	
Funct	ion	Output v	ariable		
SOLA	R	Solar circ	uit		
SOL. F	PRI.	Flushing	process		
		OK	01/ 11		Const
		ОК	OK, witho	ut allocation	Cancel

**OK** saves the settings of all outputs and the drawing object is assigned to the selected output.

OK, without allocation saves the settings of all outputs.

The **Links** comply with the **Output status** in the controller.

Here, all output links are displayed with their functions.

Likewise, the above also applies to output 14 (data link), analogue outputs 15 and 16, as well as the mixer output pairs 3/4, 8/9, 10/11 and 12/13.

## Inputs

As with all drawing objects, a **double click** takes you to the parameter menu. **Example**: Input 1 collector sensor

Inputs - Input 1 - T.collector			×						
	Drawing object: In	put 1 🛛 🗸 🗸							
Parameters									
De	es. group: General	×							
Designation: T.collector V									
Туре	Measured variable	Process variable	Sensor						
Ounused	Temperature	🔾 dim.less	KTY10						
◯ Digital	◯ Solar rad.	◯ Temperature	O Pt1000						
Analogue	◯ Voltage	🔾 Solar rad.	⊖ RAS						
O Pulse	Current	◯ Voltage	⊖ RAS PT						
	○ Resistance	◯ Current	GBS01						
Average: 1,0 sec 🗸	○ Flow rate	○ Resistance	Check: No 🗸						
Quot.: 🗸 🗸	O Pulse	◯ Flow rate	Corr.: 0.0 K 🗸						
Divisor: V	○ Wind speed	OPressure	Con.: 0,0 K V						
Scaling · · · · · · · · · · · · · · · · · · ·									
	ОК	OK, without al	location Cancel						

#### **Sensor status**

Setting parameters following a **double click**. **Example**: Display of the sensor status for sensor input 1

Sensor status S1 T.Collector

### **Network outputs**

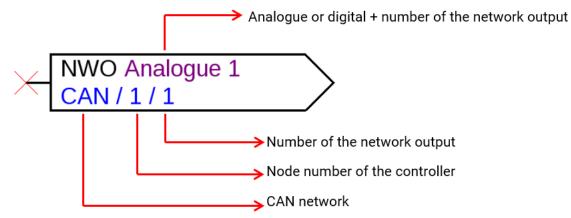
Setting parameters following a **double click**.

Under **Controller**, the network settings of the controller are defined (node number, network enable, auto operation).

The link can be seen under Parameter.

Under the 3rd tab, the Transmission conditions are defined.

After setting parameters, the network output is displayed as follows:



#### **Network inputs**

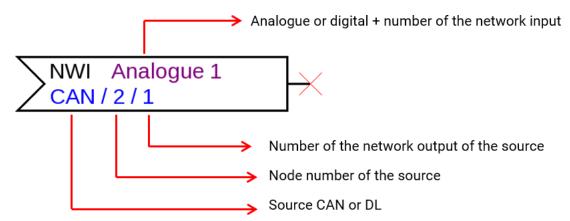
Setting parameters following a double click.

In the tab "**Controller**", the network settings of the controller are defined (node number, network release, auto operating).

In "Parameter", the source for the network input is defined.

In the 3rd tab, the "Timeouts" are defined.

After parameterisation, the network output is displayed as follows:



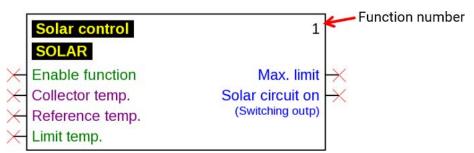
#### **Network status**

Example: Network status display for network input analogue 1



#### **Functions**

Setting parameters following a double click. **Example**: Solar function



The input variables highlighted in **purple** are **mandatory variables** that must be linked without fail. Setting parameters following a double click on the drawing object is not possible immediately unless these links have been made. If this is attempted anyhow, an error message will appear which will have to be confirmed with OK.

"Mandatory!" will appear in the parameters of the respective input variables that were not linked.

The input variables highlighted in **green** can be used but do not have to be.

Collector temperature
Source: Mandatory!

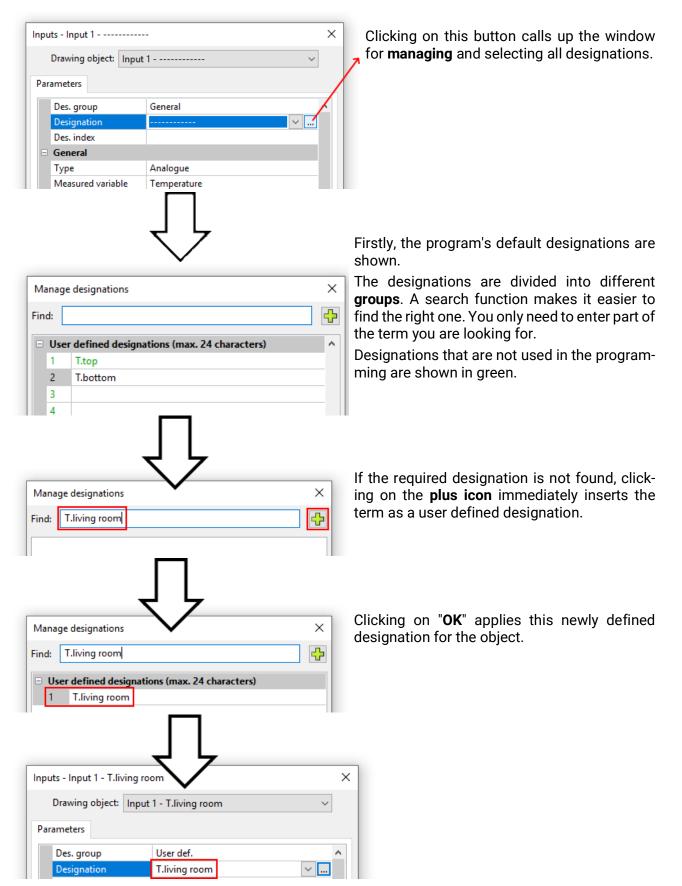
- Reference temperature Source: Mandatory!

Example: Heating circuit controller

ating circ. ctrllr - HTG CIRC.1			>
nput variables Parameters Outpu	it variables		
	Des. group: General Designation: HTG CIRC. Operation: Tir	✓ … ✓ 1 ✓	
Room temperature		Shutdown conditions	
T.room setback: 16 °C ~ T.room std: 20 °C ~	Time programs Derivative time: 0 min ~	If T.room Act. > set? No ~	If T.flow Set < min.? No ~
-		Hysteresis: 1,0 K $\checkmark$	Hysteresis: 2,0 K $\checkmark$
Flow temp heat curve Control O Fixed value O Outside temp.	Room influence: 0 % ~ Start excess: 0 % ~	If T.outside Aver.off > max.? No ~ T.outside max.: 20 °C ~	If setback mode and T.outside act. > min.? No ~ T.outside min.: 5 °C ~
Heat curve	T.flow+10°C: 35 °C ∨ T.flow-20°C: 60 °C ∨ Slope: ∨	Hysteresis: 2,0 K $\checkmark$	Hysteresis: 2,0 K ∨
	T.flow max.: 65 °C V T.flow min.: 20 °C V	Act. > max.? No <>	er: Close V
Outside temp averages		Frost protection	
Flow ctrl: 10 min ~	Shutdown: 30 min $\checkmark$	T.outside av.c. < 5 °C ~	T.room frost: 5 °C $$
			OK Cancel

## Function data objects for devices with x2 technology

(UVR16x2, RSM610, UVR610, CAN-I/O45, CAN-BC2, CAN-EZ2) Managing designations, creating user defined designations



-	User	defined designations (max. 24 characters)
	1	T.living room
	2	T.bottom
1	3	

A new designation can also be entered directly in the list of user defined designations.

Clicking on "**OK**" allocates the designation to the object, whereas clicking on "**OK**, without allocation" only saves the new designation in the list without applying the designation to the object.

🗆 La:	st used user def. de	signations		
	T.living room			
	Oxygen content			~
	ОК	OK, without allocation	Cancel	

All **recently** used user defined designations (including those from earlier programs) are listed at the bottom of the window and can be selected directly or using the search function.

For direct selection, click on the designation (becomes highlighted) and then "OK".

As soon as a designation from the list of recently used designations is allocated to an object, it is also shown in the list of user defined designations in the current program.

**Up to 100** designations can be defined (**250** for UVR16x2, UVR610, CAN-EZ3 and CAN-MTx2) with a maximum of 24 characters, which can be allocated to **all** function data objects as required.

Previously defined designations can be edited (changed) directly in the list of user defined designations. The modified designation appears in **addition** to the original designation in the list of **recently** used designations.

Entries in the list of recently used designations can be deleted with the "**Delete**" key. The program's default designations **cannot** be edited or deleted.

#### **Double designations**

Inputs - Input 2 - T.bottom				
Drawing object: Input	2 - T.bottom	$\sim$		
Parameters				
Des. group User def.				
Designation	T.bottom			
Des. index				

If a designation that has already been assigned is selected for on object, the areas "Designation" and "Des. index" will be shaded in **yellow**.

### Global user defined designations (multilingual)

Using the "**global user defined designations**", it is possible to create translations of these designations and apply them in the required language.

These designations include the "recently used user defined designations".

For this purpose, a **translation memory** is created, which is saved on the user's PC irrespective of the relevant program, and which can be called up by TAPPS2.

The example below explains how to create a **translation memory** in the available program languages.

File	Edit	view O	oject	EXITAS	пе	ip			
	B	8 9	1   4	La	angu	iage			
:				G	loba	l user d	ef. desi	gnations	N
Libra					ĺ		7		-6 <u>-</u>
	G	obal user d	lefined	designa	ation	s		×	
		Use multi	lingua	l user de	fine	d desig	nations		
	_	- Translation	-					_	
				-	. ,				
		Source lan	guage	: Englis	h (	8 Desig	gnation	s)	
		Languag	e Tra	nslated					
		German	0%						
		French	0%						
		Italian	0%						
		Spanish	0%						
		Dutch	0%						
		Czech	0%						
		Hungaria	in 0%						
		Romania	n 0%						
		Danish	0%						
		Swedish	0%						
		Export	. <b></b>	Im	port	t			
							Close		

Eight designations are currently saved. After ticking "Use multilingual user defined designations", click on "Export" to create a csv file for the translator.

Export		×		
Individual language for translator				
Languag	e: German	$\sim$		
🗌 incl. d	designations a	already translated		
O Entire translation memory				
	ОК	Cancel		

Here, the **language** for the csv file is selected along with whether previously translated designations should be included in the csv file as well.

If the **entire** translation memory (all languages with all existing translations) is exported to the csv file, the translation memory can be imported into TAPPS2 on **other** computers.

Click on  $"\mbox{OK}"$  and then specify the folder and file name for the csv file.

#### **Example** of a csv file after export and translation:

1	A	В	С	D
1	en	Comments	Max. length	de
2	T.bottom		24	T.unten
3	T.centre		24	T.Mitte
4	T.top		24	T.oben
5				

The programmer can enter explanatory comments on the terms for the translator in the csv file.

The translator puts the translated designations in the relevant language column. In this example, the csv file was created for German (column "de" = German).

Translation m Source langu	iemory age: English (8Designations)
Language	Translated
German	0%
French	0%
Italian	0%
Spanish	0%
Dutch	0%
Czech	0%
Hungarian	0%
Romanian	0%
Danish	0%
Swedish	0%
Export	Import

After translation, this table can be reimported into TAPPS2.

Import X		
Add translations		
OK Cancel		

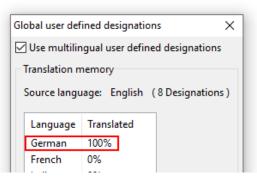
Import	×	
Merge translation memories     Replace translation memory		
O Replace translation memory OK Cancel		

Tick these boxes as required to insert the translation and/or the comments.

If comments have been included once, in future they will automatically be included (even in other languages) when exporting to the csv file. Press "**OK**" to finish.

When exporting an "**entire translation memory**", you have to decide whether to **combine** the file with the existing translations in the translation memory, or whether to completely **replace** the translation memory. Press "**OK**" to finish.

In the "User defined designations" menu, you can see whether all the designations in the translation memory are translated (= 100 %).



#### Applying the translated designations

**Example:** In a program that has been created in **English**, the user defined designations also need to be shown in **German** on the x2 device for an **German-speaking** user.

Requirement: All user defined designations within the program must be translated.

vice set	tings	
General	CAN bus DL bus	
🗆 Gen	eral settings	
Brig	htness	100,0 %
Disp	olay timeout	00:30 [mm:ss]
Cur	rency	Euro
🗆 Use	r	
Tec	hnician password	64
Expe	ert password	128
Acc	ess to menu	User
🗆 Tim	e / location	
auto	om. time changeover	Yes
Tim	e zone	01:00 [hh:mm]
GPS	latitude	48,836500 °
GPS	longitude	15,080000 °
🗆 Use	r defined designation	ns
Lan	guage	English 🗸
		Undefined
		German
		English
		French <sup>1</sup> /5"
		Italian
		Spanish
		Dutch
		Czech
		Hungarian
		Romanian
		Danish
		Swedish

The original language was **English**. Before the change, **English** user defined designations are therefore shown. The language for the designations needs to be **changed** to **German**.

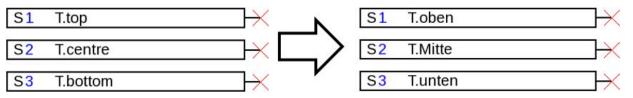


The language has to be **changed** for this application scenario.

If not all designations in the list of "**User defined designations**" are translated, the language cannot be changed and the following message appears:

Tap	pps2	×
	•	The language cannot be changed as translations are not available for all designations.
		OK

Once the language has been changed from English to German, designations are shown in German.



A **dat-**file has to be generated for the x2 device for every language so that the right user defined designations can be shown on the device.

#### Creation of translations for non-defined languages

Procedure if the language of the user defined names has not yet been defined:

- 1. Open the program with TAPPS2 version  $\geq$ 1.11.
- 2. Set the language to **English** for the user defined designations in the menu File/Settings/Device settings.



3. Add the designations to the translation memory (menu "Extras/Global user defined designations").

Clabel was def			$\overline{}$						
Global user defined designations X									
Use multilingual user defined designations									
Translation m	Translation memory								
Source langu	Source language: English (8 Designations)								
A 2 User of memory	2 User defined designations of drawing not yet transferred to translation Accept								
memo	y.		-						
Language	Translated								
German	100%								
French	0%								
Italian	0%								
Spanish	0%								
Dutch	0%								
Czech	0%								
Hungarian	0%								
Romanian	0%								
Danish	0%								
Swedish	0%								
Export Import									
Exportin	in portai								
		Close							
		Close							

If the language has not yet been defined in the device settings (language "**undefined**"), the following message appears:

Global user defined designations					
Use multilingual user defined designations					
Translation memory					
Source language: English (8 Designations)					
The language of the user defined designations in the drawing is undefined.	n				

- 4. Click "Export" to create a csv file for the translator.
- 5. After translation, **import** the csv file.
- 6. In future, the "old" user defined designations can now also be used in other programs (see previous chapter "Applying the translated designations").

#### Outputs

A **double click** on the object takes you to the parameter menu. The drawing object will be initially deemed to be **unused**. After assigning an output number, parameters are displayed for selection.

Example: Output 1, Solar pump

utp	uts - Output 1 - Solar j	pump 1 >				
	Drawing object: 0	utput 1 - Solar pump 1 🗸 🗸 🗸 🗸 🗸 🗸				
ink		king protection				
	5					
	Des. group	General				
	Designation	Solar pump				
	Des. index	1				
	General					
	Туре	Switching output				
	Mode					
	Delay	00:00 [mm:ss]				
Run-on		00:00 [mm:ss]				
	Runtime					
	Runtime limit					
	Output value digital /	manual mode				
	Dominant off					
	Digital on					
	Scaling					
	Input value 1					
	Target value 1					
	Input value 2					
	Target value 2					
	Output status					
	ON if					
	Threshold					
	Manual mode					
	Can be changed throu	igh User				

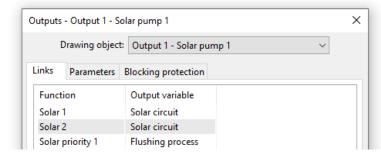
Nach Auswahl der Ausgangsnummer muss der Typ festgelegt werden.

Danach wird die Bezeichnung ausgewählt und werden alle weiteren Einstellungen vorgenommen.

**OK** saves the settings of all outputs and the drawing object is assigned to the selected output.

**OK, without allocation** saves the settings of all outputs.

#### Links



Here,	all	conr	nected	links	of	the	output
are di	spla	ayed	with th	neir fu	nct	ions	•

#### **Blocking protection**

A time for all outputs that are to receive blocking protection (see controller manuals) can be entered here.

#### Inputs

Setting parameters following a double click. **Example**: Input 1 Sensor collector 1

puts - Input 1 - T.collector 1				
Drawing object: Inpu	ut 1 - T.collector 1 ~			
rameters				
ameters				
Des. group	General			
Designation	T.collector			
Des. index	1 🗸 🗸			
General				
Туре	Analogue			
Measured variable	Temperature			
Process variable				
Sensor	PT 1000			
Sensor correction	0,0 K			
Quotient				
Unit				
Time unit				
Average	1,0 sec			
Scaling				
Input value 1				
Target value 1				
Input value 2				
Target value 2				
Sensor check				
Sensor check	Yes			
Short circuit thresho	ld Standard			

After selecting the input number, type and variable must be defined. The designation is then selected and all additional settings are made.

**OK** saves the settings of all inputs and the drawing object is assigned to the selected input.

Save the settings of all inputs with **OK**, without allocation.

### **CAN outputs**

Setting parameters following a double click

Under the **Controller** tab, the CAN network settings of the controller are defined (node number, BUS rate, designation).

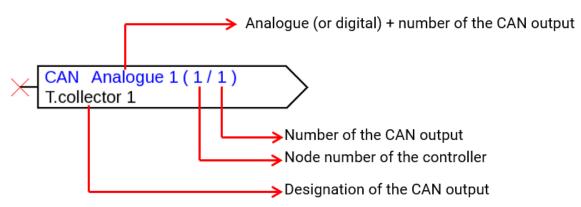
The link can be seen under **Parameter**. The designation of the CAN output and the transmission condition are defined here.

Example: Linking of analogue CAN output analogue 1 with the actual value of input 1

CAN	CAN outputs - Analogue 1 - T.collector 1 X							
D	Drawing object: Analogue V 1 - T.collector 1 V							
Devi	Device Parameters							
	Des. group	Temperature actual value						
	Designation	T.collector						
	Des. index	1						
	Input variable							
	Source type	Input						
	Source	1: T.collector						
	Variable	Measurement						
-	condition							
	lf change >	10						
	Blocking time	00:10 [mm:ss]						
Interval time 5 min								
		OK OK, without allocation Cancel						

After selecting the type and the CAN output number, the designation is selected and all additional settings are made.

After setting parameters, the CAN output is displayed as follows:



## **CAN** inputs

Setting parameters following a double click

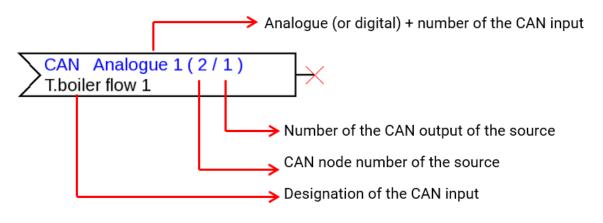
Under the **Controller** tab, the CAN network settings of the controller are defined (node number, BUS rate, designation).

In Parameter, the source and the time-out for the CAN input are defined.

**Example:** Analogue CAN input analogue 1 of CAN Bus device with node number 2 and its CAN output 1

Drawing object: Ar	alogue V 1 - T.boiler flow 1 V	After selecting the type and the CAN input number, the designa-
Device Parameters		tion is selected and all addition- al settings are made.
Des. group	Temperature actual value	ar settings are made.
Designation	T.boiler flow	
Des. index	1	
General		
Node number	2	
Output number	1	
CAN BUS timeout	00:20 [hh:mm]	
🗉 Unit		
Measured variable	User def.	
Unit	Temperature °C	
Sensor correction	0,0 K	With entry of the User def. vari-
Value at timeout	Unchanged	able User def., the unit, a sensor
Output value		correction and user-defined
Sensor check		monitoring of the sensor value
Sensor check	Yes	can be defined.
Short circuit thresh	old Standard	
Threshold value	2	
Short circuit value	Standard	
Output value		
	ld Standard	
Lead break thresho		
Lead break thresho Threshold value	2	
	Standard	

After setting parameters, the CAN input is displayed as follows:



#### **DL outputs**

Setting parameters following a double click

Analogue as well as digital values can be transmitted via DL output.

The link can be seen under Parameter. The designation of the DL output is defined here.

Example: Linking the DL output 1 with the result of the OR function for target address 1 index 1

DL-A	Ausgänge - J	Ausgang 1 - O2 Senso	ŗ	×	After selecting the DL output num-
	Zeichnun	gsobjekt: Ausgang 1	- O2 Sensor	~	ber, the designation is selected and
Par	ameter				all additional settings are made.
	BezGrupp	e Benutzerdef.			
	Bezeichnur	ng O2 Sensor			
	BezIndex				
Ξ	Eingangsva	ariable			
	Quelletyp	Funktion			
	Quelle	Oder			
	Variable	Ergebnis			
	Ziel				
	Adresse	1			
	Index	1			
te	r setting	OK parameters, the	OK, ohne Zuweisen DL output is displayed DL output + nu		DL output
		Output 1 ( 1	1/1)	$\mathbf{\mathbf{N}}$	
\$	Senso	r O2		/	
			Target in	ndex	
			Target a	ddress	

Designation of the DL output

### **DL inputs**

L

Setting parameters following a double click In Parameter, the type and the source of the DL input are defined. **Example:** Analogue DL input 1 of DL sensor with address 1 and its index 1

DL-	Eingänge - Eingang 1 - T.R	aum X	After selecting the DL input num-					
	Zeichnungsobjekt: Ein	gang 1 - T.Raum 🗸 🗸	ber, the designation is selected and					
Par	rameter		all additional settings are made.					
	BezGruppe	Temperatur Istwert						
	Bezeichnung	T.Raum						
BezIndex								
Ξ								
Typ Analog								
	Adresse	1						
	Index	1						
	Einheit							
	Messgröße	Benutzerdef.						
	Einheit	Temperatur °C						
	Sensorkorrektur	0,0 K	With entry of the <b>User def.</b> variable,					
Ξ	Wert bei Timeout	Unverändert	the unit, a sensor correction and					
	Ausgabewert		user-defined monitoring of the sen-					
=	Sensorcheck		sor value can be defined.					
	Sensorcheck	Ja						
=	Kurzschlussschwelle Standard							
	Schwellwert							
Ξ	E Kurzschlusswert Standard							
	Ausgabewert							
0	Unterbrechungsschwelle Standard							
	Schwellwert							
Ξ	Unterbrechungswert	Standard						
	Ausgabewert							
	O	K OK, ohne Zuweisen Abbrechen						
∆ft∆	r setting narameters	s, the DL input is displayed as follows:						
	r setting parameters							
	Г	DL input + number of the second se	he DL input					
$\overline{}$	DI lucit	a (a 1 a)						
	DL Input	1 (1 / 1)	/					
/	T.Room		×					
		Index of the actual v	alue of the source					
		Address of the source						

#### **Fixed values**

Setting parameters following a **double click** 

Example: Fixed value 1 with designation "Nominal temperature 1", with setting limits

xed valu	ies - Fixed va	lue 1 - Se	t value	:1	
D	rawing objec	t: Fixed	value	1 - Set value 1	~
aramet	ers				
Des	group		Gene	ral	
Desi	gnation		Set va	lue	
Des.	index		1		
General					
Тур	e		Analo	gue	
Fun	ction quantit	у	dime	nsionless	
Cha	ngeover				
Min	imum		18		
Max	imum		24		
🗆 Fixe	d value				
Valu	e		20		
Can	be changed	through	User		

After selecting the fixed value number, type and function size must be defined. Then the designation can be entered and all additional settings made.

#### **System values**

Setting parameters following a double click

Example: System value time / hour

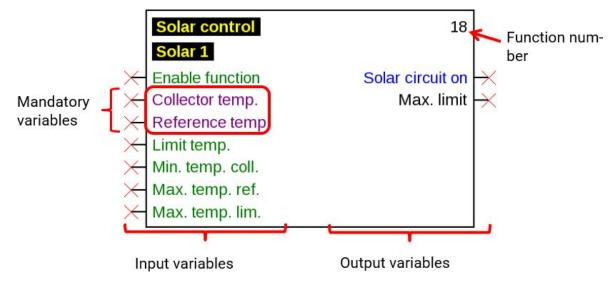
· · · · · · · · · · · · · · · · · · ·
---------------------------------------

System valu	ie - unused X
Parameter	5
Group	unused 🗸
Value	unused
	General
	Time
	Date 😼
	Sun

The required value is defined after selecting the group.

### **Functions**

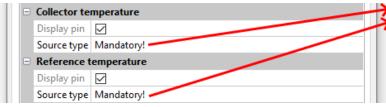
Setting parameters following a **double click Example**: Solar function



The input variables highlighted in **purple** are **mandatory variables** that must be linked without fail. Setting parameters following a double click on the drawing object is not possible immediately unless these links have been linked. If this is attempted anyhow, an error message will first appear which can be confirmed with **OK**.

Example: Solar function

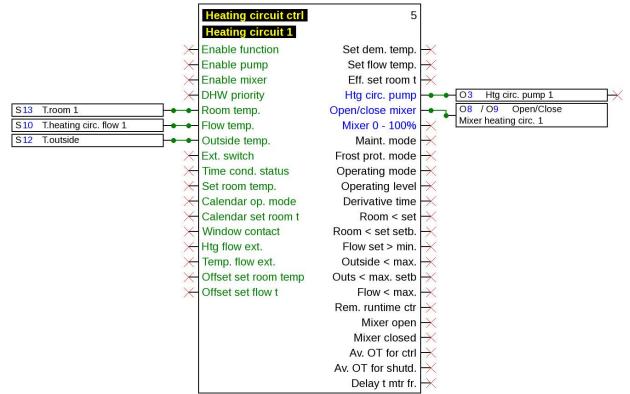
Error	×
8	Mandatory input variables must be combined with exactly one programmed source: Target: Solar control - Solar 1 - Collector temperature Mandatory input variables must be combined with exactly one programmed source: Target: Solar control - Solar 1 - Reference temperature Result: 2 Errors, 0 Warnings
	OK Abbrechen



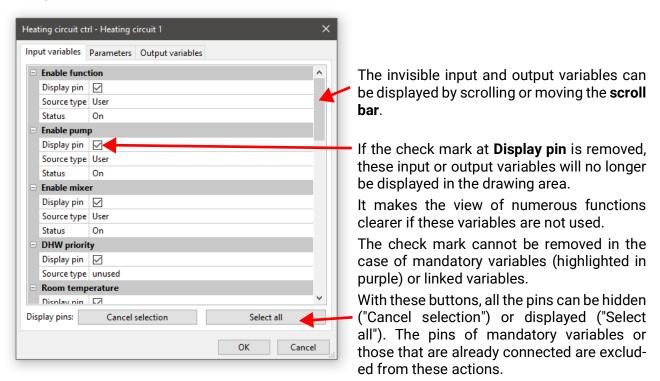
"Mandatory!" will appear in the parameters of the respective input variables that were not linked.

The input variables highlighted in green can be used but do not have to be.

#### Example: Heating circuit controller with all input and output variables



Display / hide input and output variables



A selection for the input and output variables displayed **immediately** was made at the factory to make the display of functions clearer. These hidden variables can be re-displayed at any time with **Display pin**.

**Example:** Unused input and output variables were hidden.

	Heating circuit ctrl	5	
	Heating circuit 2		
×	Enable pump	Set flow temp.	×
×	Room temp.	Eff. set room t	×
×	Flow temp.	Htg circ. pump	×
$\times$	Outside temp.	Open/close mixer	$\times$
$\times$	Status time cond		
×	Calendar op. mode		
X	Calendar set room t		

#### Setting the parameters

nput variables	Parameters	Output variables		Tho	hiddon parameters can be
Des. group		General	^		hidden parameters can be blayed by scrolling or mov-
Designation		Heating circuit			the scroll bar.
Des. index		1	K	g	
•					
Operation		Time/auto			
B Room tempe	erature				
T.room setba	ck	16,0 °C			
T.room stand	lard	20,0 °C			
Outside tem	perature				
Derivative tin	ne	00:00 [hh:mm]			
Average	time				
for flow c		00:10:00 [hh:mm:ss]			
for shutd	own	00:30:00 [hh:mm:ss]			
■ Flow temp	heat curve				
Control		Outside temp.			
Heat curve		Temperature			
D : fl		0.0%			

After linking an input, output, DL input or CAN input with a function, the information of the variables to be transmitted to the function can be defined in the input variables of the function. **Example:** Analogue function, DL input analogue

np	ut variables	Parameters Output variables	
•	Input variab	le 1	٨
	Display pin	$\checkmark$	
	Source type	Input	
	Source	6: T.cylinder 1	
	Variable	Measurement	
	Input variab	le 2	
	Display pin	$\checkmark$	
	Source type	DL input	
	Source	1: T.room	
	Variable	Measurement 🗸	
-	Input variab	Measurement	
	Display pin	RAS mode	
	Source type		
	Input variab	Network error	Υ.

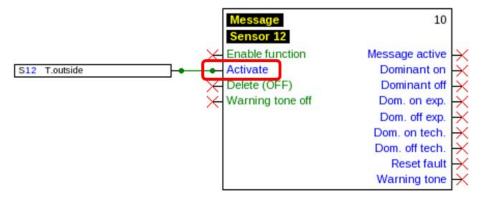
- Measurement the value captured by the sensor
- **RAS mode** the following analogue values are issue depending on the switch position on the room sensor (RAS):
  - Automatic 0
  - Normal 1
  - Setback
  - Standby
- Sensor error digital value, ON with sensor errors

2

3

- Network error digital value, ON there has been a time-out (= error)
- Input variables deviating from the factory settings will be displayed in **blue**.

Example: The Sensor error variable of the sensor was selected.



#### Output variables

	Anforderung Warmwasse	<b>r</b> 3	
	Anforderung Warmwasse	r	The output variables highlighted in
×	Freigabe Funktion	Eff. Solltemp.	$\rightarrow$ <b>black</b> can be linked <b>only</b> with input var- iables of other functions or CAN and DL
X	T.WW.oben	Solltemp.	$\rightarrow$ outputs.
X	T.WW.unten	Anforderung	$\rightarrow$ The output variables highlighted in <b>blue</b>
X	Status Zeitbed.	Erzeugerleistung	$\rightarrow$ can be linked with outputs. But links to input variables of other functions or
X	T.Soll.oben		CAN and DL outputs are also permit-
X	T.Soll.unten		ted.
$\times$	Ext. Schalter		

An error message is displayed if a forbidden link is made. **Example:** 

Error	×
$\otimes$	Output variable must not be combined with any output: Source: DHW demand - DHW demand - Effective set temperature Target: Output 1 - DHW charging pump
	Result: 1 Error, 0 Warnings
	OK Abbrechen

### **CORA devices**

Shown only when programming CORA devices (e.g. CAN-EZ3) The "CORA devices" item is located at the end of the selection tree.

CORA devices				

#### **Example: EHS** immersion heater Programming after **double clicking**.

EHS immers Immersion h	Timeout	Number of the CORA device in the program Output variables (sent by the immersion heat- er)
EHS immersion heater - Imm	nersion heater 1 X	
Des. group	General	
Designation	Immersion heater	
Des. index	1	
•		
CORA ID	00000001	x2 wireless ID of the CORA device
HOP1 ID	00000000	
HOP2 ID	00000000	
HOP3 ID	0000000	
Connect automatically	Yes	
	OK Cancel	

The drawing objects for CORA devices are also designed for input variables (= values sent to the CORA device), but there is no use for them as yet.

More detailed information on CORA devices and how they work, as well as general information about the wireless system, can be found in the instructions for the respective devices.

# Links

## **Direct links between 2 objects**

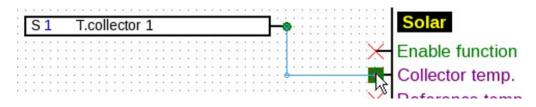
1. If you hold the cursor over a link, it will turn into a green square and the cursor turns into a pencil.

S1 T.collector 1

2. A mouse-click changes the shape to a green circle.

S1 T.collector 1

3. Now drag the cursor to the required target point to generate a line.

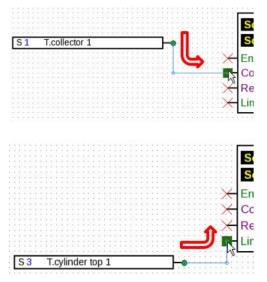


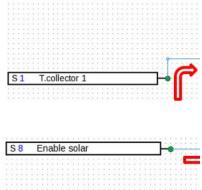
If the target point is not on the same level, the line is adjusted at right angles. 4. A click on the target point fixes the line and thus the link in place.

-	S	1			٦		CC	oll	e	C	to	r	1												Ľ				****					Solar
_	-	-		_				_	_	-		-	-	_	-	-	-	_	-	_	_	-	-	_		T ·	+		+	•		+	• •	
						• •							*	• •				1						· • •		1.1						1	• ;	
122	5		10	32	÷.	101	0		10.5		8÷.	53	•	•	63	13	35	13			35	100		200	1.5	1.	. *	 316	100	<			<del>~</del>	Enable func
	έ.		1.14			• •	i s					4		• •	÷.,		+	÷				4	• •			•		 	*		. 4	./	. \	Encore runo
	5.	0.2	578	100		1.1	0.12		0.08			12			0.8	12	1		10.1		0.8	100			5.52	•	1.78.7	 0.04	*					O 11
		6.9		+	÷.							+				- 20					+					_		 				-	<u> </u>	Collector te
		2.			2				12						1.2																			Concetor te

The green, circular link points show that a link exists.

The direction in which the cursor is dragged from the starting point determines the course of the link line:

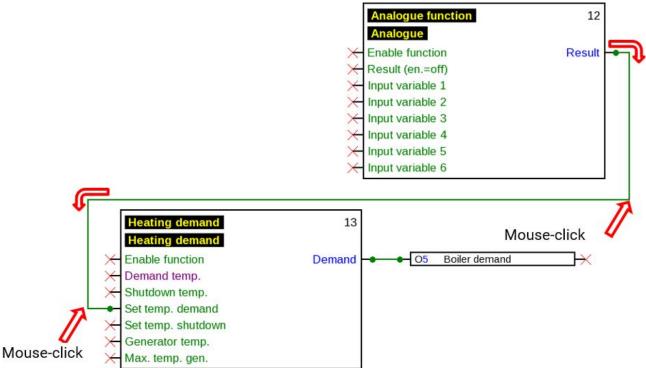






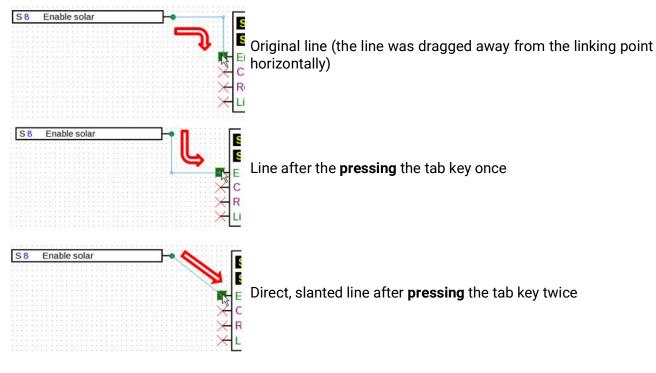
A mouse-click whilst dragging of the line generates a corner point. Lines can thus be taken around other objects and thus do not disturb the view of the object.





### Changing the line shape

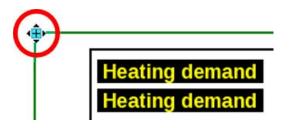
As long as the link is not completed, the line shape can be changed by pressing the **tab key**:



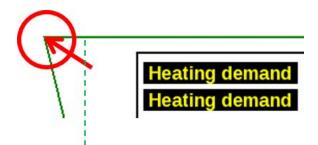
# **Editing lines**

### Moving corner points

1. Place the cursor over the required corner point which changes its shape.

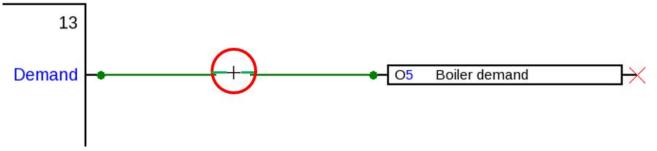


2. With the mouse button held down, the corner point can now be moved.

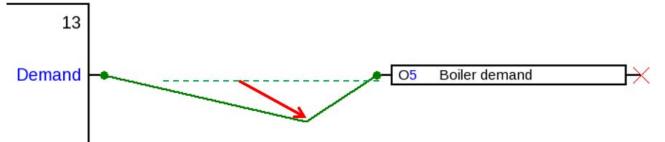


#### Changing a line

If you place the cursor directly over a point on the line, its shape will turn into a "+" sign.

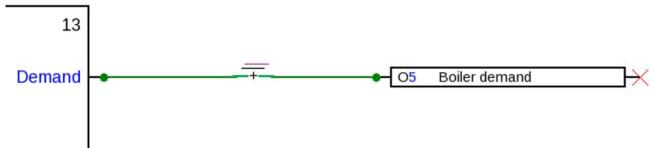


With the left mouse button held down, the line can now be dragged to this point.

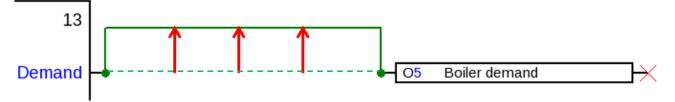


### Moving a line

The shape of the cursor changes if you place the cursor over a line whilst holding down Alt.

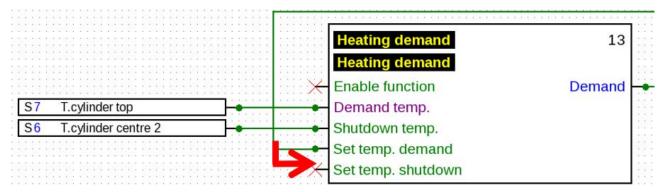


With the left-hand mouse button held down, the line can now be dragged in parallel.



## **Branches**

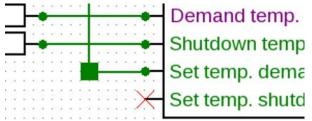
**Example**: Heating demand; the set value demand is to be linked with the set temperature shutdown by means of a branch.



The branch can be generated in two ways:

- 1. Press **Ctrl** on the keyboard and take the cursor to the required node point. The node point is displayed as a green square; the cursor turns into a "pencil".
- 2. Click on the **Node mode** in the right-hand tool bar.

The cursor is taken to the required node point. The node point is displayed as a green square; the cursor turns into a "pencil".





A mouse-click changes the shape to a green circle.

Now drag the cursor to the required target point to generate a line.

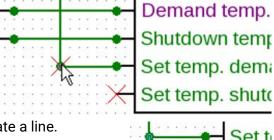
A click on the target point fixes the line and thus the link in place.

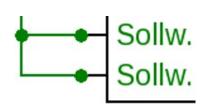
# Deleting objects and lines

- 1. Select an object, object group or line
- 2. Delete with the help of **Del** or with the **Edit / Delete** parameter.

# Command "Select links"

The commands "Select links" and "Edit / Select links" (shortcut: Ctrl+Shift+F) select all links of the currently selected drawing object. In addition, if signal transfer is selected, all associated signal acceptances are selected and vice versa.





# Invalid links

If links are created that are not permitted, an **error message** will be displayed when attempting to set parameters.

Setting parameters can continue if you click on **OK** in spite of the message.

We recommend removing the error though **before** continuing with setting parameters.

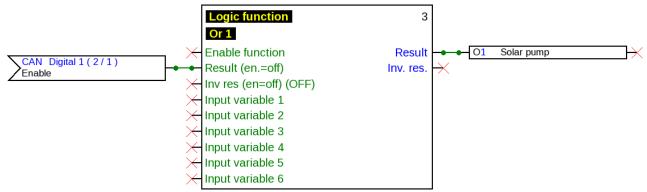
Function data for the controller can be generated only after all errors have been removed.

#### Network input – output

An output cannot be switched directly from a digital network input. This requires the appropriate logic or analogue function.

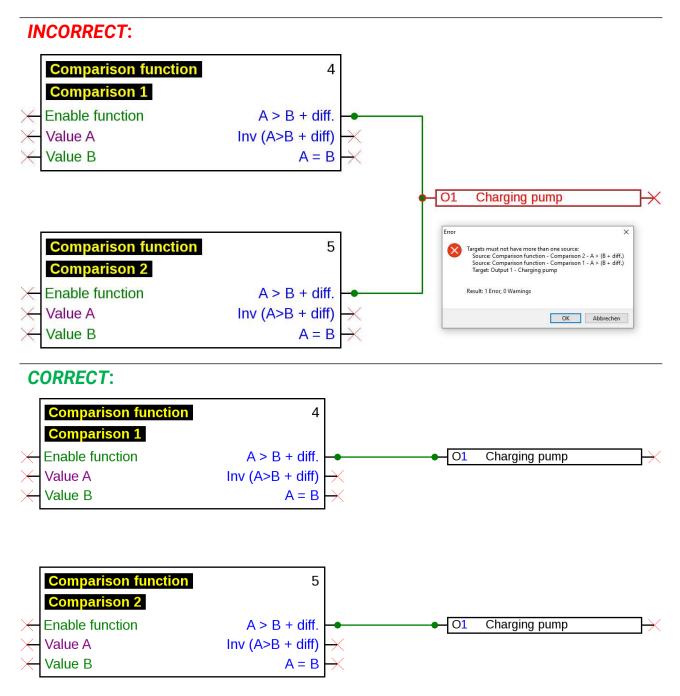
CORREC	Т:								
	CAN Digital 1 ( 2/1) Enable O1 Solar pump								
Error	×								
8	Inputs, bus inputs, fixed values, system values and outputs of outputs must not be directly combined with outputs: Source: CAN input Digital 1 - Enable Target: Output 1 - Solar pump								
	Result: 1 Error, 0 Warnings								
	OK Abbrechen								

# CORRECT:



#### Linking two output variables

Output variables may not be linked.

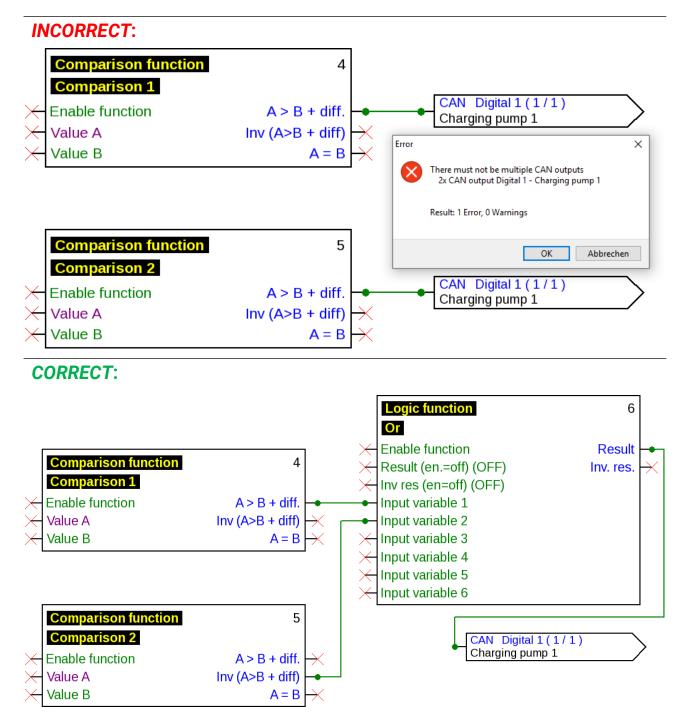


For this, identical outputs are linked with the logical operator OR (if one switches to ON, so do all the rest).

### Single network output on multiple functions

Every network output must appear only once in the drawing.

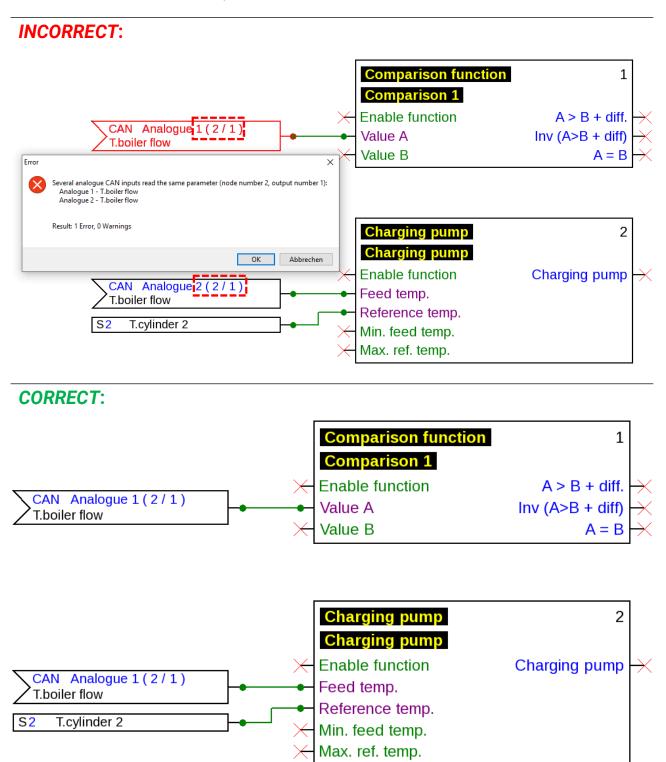
If the value of a network output is formed with multiple functions, then this must be realised accordingly with logic or analogue functions.



#### Several network inputs refer to one network output of another CAN-Bus device

For every parameter which is called up from the network, only one network input may be defined on a single device.

But it is possible to use several symbols for the same network input in the drawing.



#### x2 devices only

If a **CAN input** with identical parameters is defined for the node number and the output number (of the sender) of an **existing** CAN input, then this error is highlighted with a yellow background.

The same error display is issued for a DL input with the same address and index of an existing DL input.

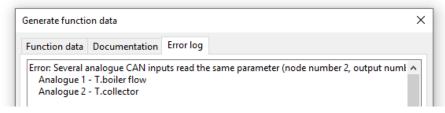
#### Example: CAN input

CAN inputs - Analogue 1 - T.boiler flow									
Drawing object: Anal	Drawing object: Analogue V 1 - T.boiler flow V								
Device Parameters									
Des. group	Temperature actual value								
Designation	T.boiler flow								
Des. index									
General									
Node number	2								
Output number	1								
CAN BUS timeout	00:20 [hh:mm]								

If this error is not corrected and the CAN input is opened again, an error message will be displayed:

Error	×
8	Several analogue CAN inputs read the same parameter (node number 2, output number 1): Analogue 1 - T.boiler flow Analogue 2 - T.collector
	Result: 1 Error, 0 Warnings
	OK Abbrechen

If the error is still not corrected, exporting faulty function data will be prevented and the error log will highlight the cause:



# **Texts**

# **Text insertions**

- 1. A mouse-click on the text symbol A in the right-hand tool bar activates text mode. The cursor in the diagram changes its shape to a "pencil".

2. Positioning the text field the drawing with a mouse-click.

- 3. Entry of the text
- Texteingabe 4. Clicking in a free drawing area stops the entry.

# Texteingabe

- 5. The cursor is still a "pencil" and additional text can therefore be entered. Only when Selection mode (cursor:  $\mathbf{k}$ ) in the tool bar is selected will the text mode terminate.
- 6. If necessary, the text format (in Selection mode) can now be changed with a double click on the text. Tard

	ie)	XL			
ont					×
Font: Arial Comic Sans MS Georgia Impact Liberation Mono Liberation Sans Liberation Serif Palatino Linotype Tahoma Times New Roman Trebuchet MS Vordana	review:	Font style: Black Bold Italic Italic Narrow Standard	C	Size: 7 7 8 9 10 11 12 14 16 18 20	2
Font colour:		Background c	olour		
	$\sim$	Tran	nspare	ent	$\sim$



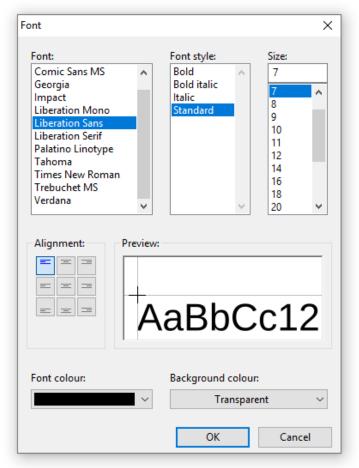
Subsequent editing of the text is possible with shift + double click on the text or by selecting the tool A (insert Text) and a mouse-click on the text.

Texts can be moved, selected and aligned like all others objects.

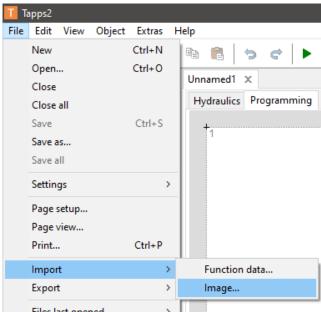
### **Global text formatting**

By selecting Object / Font, the font can be formatted for all subsequently inserted texts if no text object in the drawing is selected.

If a text object is selected, the formatting can be changed for this object.GG



### **Graphics/Images**



Files last opened

Small graphics can be placed in the hydraulic system drawing via File > Import > Image.

Images must be in PNG format and should not exceed a file size of 100 kB.

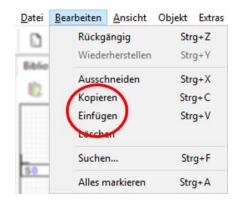
# **Copying objects and texts**

Before copying, the required object or the required object group must be selected.

- Copying can be done in three different ways:
- 1. Using the symbols in the top tool bar:

T Kopieren P

- 2. Using shortcuts: Copy: Ctrl + c Paste: Ctrl + v
- 3. Using the commands under Edit:



### Copying within a drawing

An example explains the individual steps. **Example**: Copying an output

- 1. Selecting the object 🗡 O1 Solar pump 1
- 2. Applying one of the three copying methods
- 3. Trigger the associated **Paste**, the object is now indicated at the cursor.



Solar pur

01

4. A mouse-click fixes the position of the object. Pasting several times is also possible.

#### All copied objects adopt the parameters of the original object.

**Exception:** Functions in which **involved functions** are listed (e.g. solar priority).

If <u>only</u> the function is copied, then the **involved** functions of the copied function will be **maintained**. If <u>at least one</u> of the involved functions is also copied, then the involved functions also copied will be listed **as new functions** but with the same designation. The functions that are not also copied return to "not selected". Their parameters must subsequently be set again.

If functions are copied, then the new functions will receive numbers that continue from the last function.

### Copying between two drawings

If inputs, outputs and/or fixed values are copied between two drawings, the following window appears.

No.	Paste		No.	Allocated	Drawing
1	T.collector		1		unused
2	T.DHW		2		unused
			3		unused
			4		unused
			5		unused
			6		unused
			7		unused
			8		unused
			9		unused
		Allocate >	10		unused
		Allocate >	11		unused
		< Back	12		unused

This is used to determine the number with which copied inputs/outputs etc. are to be inserted in the programming. Objects which have already been allocated can be rearranged in the "Allocated" column using drag & drop.

When copying between two drawings, a check is performed to determine whether the elements being copied are already in use elsewhere in the target file.



Clicking "Next" starts conflict resolution for the first object type (e.g. inputs).

este - F	Resolve conflicts			>
[npi	uts (3 not allocated)			
No.	Paste	No.	Allocated	Drawing
1	T.collector	1		T.boiler flow
3	T.DHW cyl. bottom	2		T.DHW
4	T.buffer bottom 1	3		T.outside
		4		T.room 1
		5		T.heating circ. flow 1
		6		T.room 2
		7		T.heating circ. flow 2
		8		unused
		9		unused

The entries on the left are those which could not be allocated. It is now possible to use drag & drop to insert objects anywhere in the target drawing (= right-hand side). Dragging onto existing entries overwrites these entries. The number of unallocated entries is shown in red at the top, next to the object type (e.g. inputs).

When a value in the target drawing is being replaced, an arrow is shown in the right-hand list.

Inpu	uts (2 not allocated)				
No.	Paste	No.	Allocated		Drawing
3	T.DHW cyl. bottom	1	T.collector	∍	T.boiler flow
4	T.buffer bottom 1	2			T.DHW

In the example, the arrow indicates that the name and parameter settings of the left-hand value are overwriting those of the right-hand value. Clicking on the arrow reverses this function. Click "Continue" to display the conflict resolution task for the next value type.

After completing the last conflict resolution task, a summary is shown.

Paste - Resolve conflicts	-	×
All conflicts resolved, insertion now possible.		

If there are no further conflicts, the summary looks like the one shown above. Select "Finish" to accept the changes and exit conflict resolution. The copied objects can now be inserted by left-clicking. If any unresolved conflicts remain, these are shown in the last window. In this case, selecting "Finish" will discard the values in the list and insert unused values instead.

Paste - Resolve conflicts	-	×
Unassigned objects are inserted as unused:		
Outputs: 1 - Solar pump		^

Conflicts may also arise when copying objects between drawings on different appliances.

**Example:** The source file (UVR16x2 programming) has more inputs assigned to it than the target file (RSM610 programming) permits (7 inputs programmed, RSM610 only has 6 inputs).

iste - F	Resolve conflicts				– 🗆 X
Inp	uts (7 not allocated)				
No.	Paste	No.	Allocated		Drawing
7	T.buffer top	1	T.collector	=>	unused
9	T.boiler flow	2	T.DHW	=>	unused
10	T.heating circ. flow 1	3	T.DHW cyl. bottom	=>	unused
11	T.heating circ. flow 2	4	T.buffer bottom 1	=>	unused
12	T.outside	5	T.buffer bottom 2	=>	unused
13	T.room 1	6	T.buffer centre	=>	unused
14	T.room 2				

User defined designations are transferred automatically. However, if the maximum number of user defined designations in the target file is reached, this will also result in a conflict page being displayed.

You may also be prevented from inserting anything at all due to differences between the devices (in which case neither a dialogue box nor an error message is displayed). This can occur, for example, when inserting inputs into the program for a CAN-BC2.

## Paste as CAN-Inputs

If there are CAN outputs among the copied objects, they can be converted to corresponding CAN inputs when pasting. You can do this with the command "**Edit > Paste as CAN inputs**" (shortcut: Ctrl+Alt+V); the CAN node numbers, output numbers and descriptions are also adopted.

If CAN inputs are pasted in this way, none of the other elements that you initially copied are inserted. The CAN inputs are grouped according to analogue and digital inputs, and sorted by input number.

# **Deleting objects**

Highlighted objects are deleted if the **Del** key is pressed, or via Edit > Delete.

If an object (input, output, CAN input, etc.) is deleted, its parameters nevertheless continue to be stored in the programming.

If an object is deleted with **Shift+Del**, the object (if applicable, e.g. input) is deleted and its programming set to **unused**.

# **Cutting objects**

The menu command **Edit / Cutting** or the shortcut **Ctrl + x** can be used to cut out a **selected** object or **selected** object group. They are thus deleted from the drawing but remain in the clipboard so they can be pasted again.

With the **Paste** command or the shortcut **Ctrl + v**, this object can be pasted in the same or any other drawing of **the same controller type**. Pasting several times is also possible.

As with **Copy**, the same conditions for maintaining parameter settings apply.

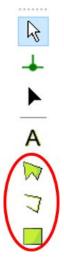
If functions are cut, the subsequent numbered functions move up in sequence.

The cut function that was pasted into a drawing will receive the number following that of the last function.

### Inserting simple drawing objects

Polygons, polylines and rectangles can be inserted in the drawing area:

- 1. Click one of the drawing icons in the right-hand toolbar to activate drawing mode. In the drawing, the cursor changes to a "pencil".
- 2. Click to position the drawing object in the drawing.
- 3. Additional editing (line type/fill) is carried out as in the "hydraulic" drawing area and is described in chapter "Hydraulics".



### Finding objects 🛔

The find function can be used, for example, in extensive programming to search for elements using their full name or type, or just parts thereof.

Example: Find "Solar pump"

M Search	×
Find: Solar pump	
⊡. Output 	
	Find and select Cancel

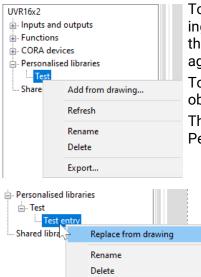
**Result**: The output with the designation "Solar pump" appears 10 times in the drawing. Clicking "**Find and select**" highlights the 10 solar pumps in the drawing in red so they can be found easily.

# **Personalised libraries**

It is possible to create personalised libraries.

	and outputs ions	<ul> <li>At the end of the selections, there is an entry "Personalised libraries".</li> <li>Right-clicking on this opens a context menu.</li> <li>•Add a new Personalised Library</li> <li>•Select the directory in which Personalised Libraries are saved</li> </ul>
Shai	Add new library	
	Select directory Refresh	•Update Personalised Libraries if they have changed •Import and export Personalised Libraries
	Import Export	

Clicking on "Add new library..." opens a dialogue box where the library can be given a name.



To add an entry to the library, select any number of objects in the drawing. Then right-click on the name of the relevant personalised library in the list and select "Add from drawing". The naming prompt appears again.

To paste the entry into a drawing, select it in the same way as any other object and then position it in the drawing.

The parameter settings of all objects added to the library are adopted. Personalised libraries affect all drawings and files.

An entry can be changed by taking it from the library and placing it in the drawing, editing it there, highlighting it and then replacing it in the library with the new version (right-click > "Replace from drawing").

Personalised libraries can be imported and exported as .lib files. Right clicking on "Personalised libraries" and "Export" opens a pop-up window, where you can choose which personalised libraries to export. You then select a folder in which to save the selected personalised library as a separate file. If you click on "Import", you can follow the same principle to import .lib files.

You can also right-click directly on a personalised library to export this file only.

Personalised libraries are saved as .lib files under *Documents*\*Technische Alternative*\*Tapps2*\*libraries*. This directory can be amended (right-click > "Select directory").

## **Shared libraries**

UVR16x2

Inputs and outputs

CORA devices

Personalised libraries

Shared libraries

Libraries can be shared between multiple users.

There is a separate additional entry for this, "Shared libraries".

To be able to use "Shared libraries", first of all a directory needs to be specified (right-click > "Select directory...").

Note that all computers with the same network protocol (SMB or NFS) can access the "Shared libraries" directory on the fileserver.

TAPPS2 automatically combines changes from multiple users of the same library.

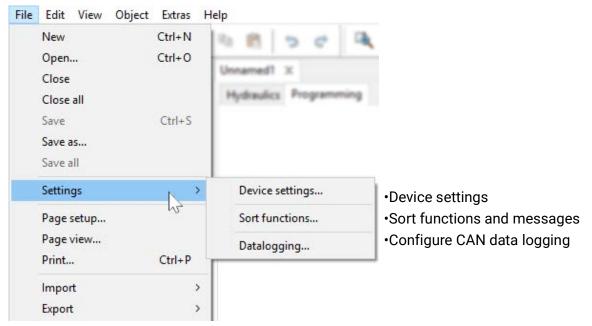
If the same element is being edited simultaneously by multiple users outside the library and is then replaced in the library, however, it is not possible to combine these changes.

If changes have been made to a library outside the program, e.g. by another user, the library is automatically reloaded before your own changes are made.

A library can also be reloaded manually with right-click > "Refresh".

# Settings

In the Settings menu, the following general settings can be made for the controller:



### **Device settings**

Depending on controller type, the default settings for the controller, the network settings and the welcome screen can be set here.

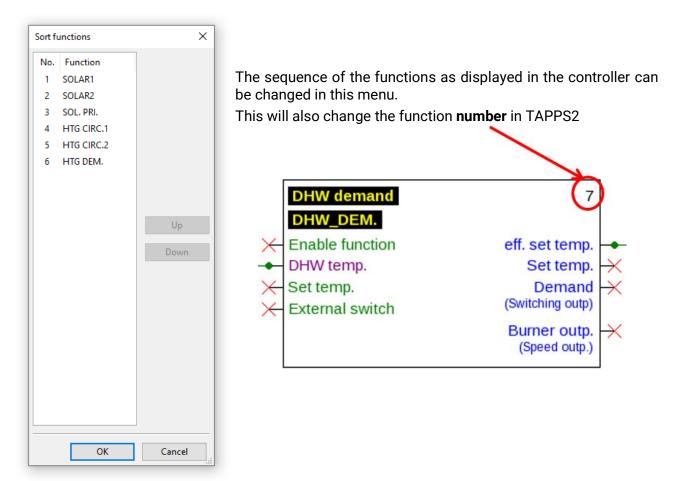
#### UVR1611

Device settings	×
Operation Network Welcome Display	
Contrast: 41 v Brightness: 10 v Illumination off after: v Automatic switching to function overview	
Date/time Automatic changeover standard/summertime	
User block Parameters Outputs Menu	
Expert code	
OK Cancel	

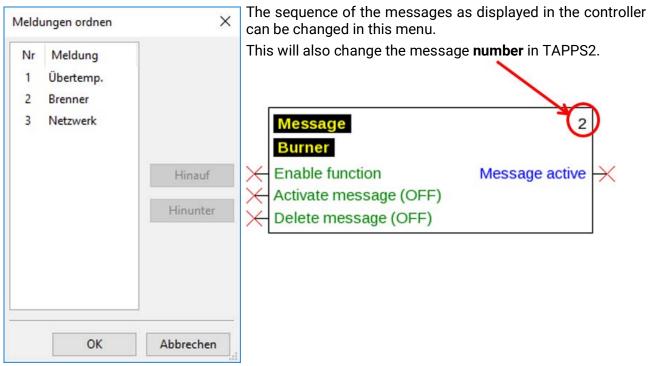
#### Devices with x2 technology (e.g. UVR16x2)

Devi	ice sett	ings				Х
Ge	eneral	CAN bus	DL bus			
E	Gen	eral setting	js			
	Brig	htness		100,0 %		
	Disp	lay timeout	t	00:30 [mm:ss]		
	Curr	ency		Euro		
E	User	r				
	Tech	nnician pass	sword	64		
	Expe	rt password	ł	128		
	Acce	ess to menu	1	User		
E	Time	e / location				
	auto	m. time ch	angeover	Yes		
	Tim	e zone		01:00 [hh:mm]		
	GPS	latitude		48,836500 °		
	GPS	longitude		15,080000 °		
E	User	defined d	esignatio	ns		
	Lang	juage		English		
				ОК	Cancel	

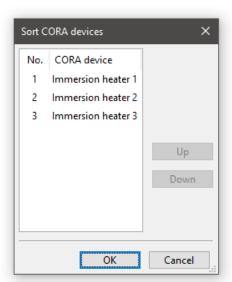
### Sort functions and messages



#### Only for UVR1611:

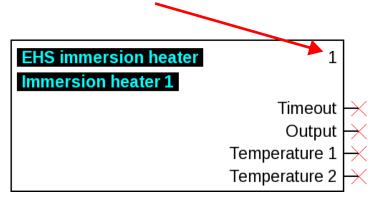


### **Sorting CORA devices**



In this menu, the order of the functions as they are displayed in the controller can be changed.

This also changes the number of the function in TAPPS2.



### Sorting inputs/outputs

Eingänge	Nr	Zeichnung
Ausgänge	1 INF	T.Kollektor
Ausgangspaare	· ·	T.Warmwasser
Fixwerte	-	T.Boiler unten
CAN-Eingänge Analog		T.Puffer unten 1
CAN-Eingänge Digital	2 T 3 T 4 T 5 T 6 T 7 T 8 T 9 T 10 T 11 T 12 T 13 T 14 T 15 u	
CAN-Ausgänge Analog CAN-Ausgänge Digital		T.Puffer unten 2
DL-Eingänge	-	T.Puffer mitte
DL-Ausgänge	· .	T.Puffer oben
5 5	-	T.Puver oben
	-	T.Kessel VL
	T.Heizkreis VL 1	
		T.Außen
		T.Raum 1
		T.Raum 2
	15	unbenutzt
	16	unbenutzt

In this menu, the assignment of inputs and outputs can be changed.

Inputs/outputs can be assigned to empty spaces or swapped with each other. These changes apply to all existing drawing objects.

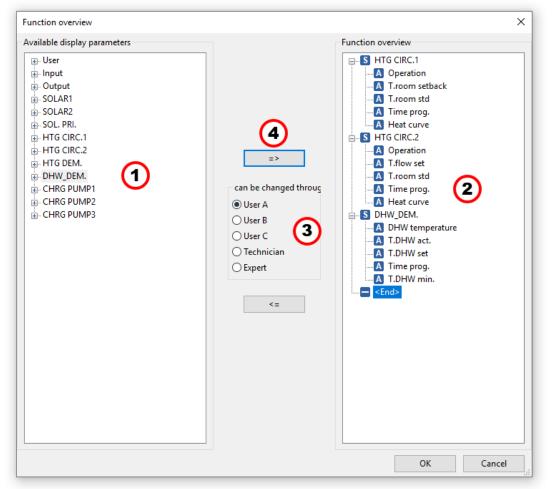
When rearranging CAN outputs, the corresponding CAN input at the receiver must be adjusted.

Likewise, an existing visualisation (TA-Designer) must be updated after rearranging inputs/ outputs. If a .dat file is exchanged there, the TA-Designer does not recognise rearranged inputs/outputs. Manual adjustments have to be made.

# **Function overview UVR1611**

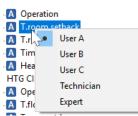
The function overview is a menu screen in the controller that serves to display only the information that is important to the customer.

The parameters displayed there can also be allocated with an authorisation for changing set values.



#### Procedure for inserting a display parameter into the function overview:

- 1. Select the parameter that is to be displayed in the function overview from the list of available parameters.
- 2. Select the position above which the display parameter is to be inserted from the list of function overview parameters.
- 3. Select the authorisation for changing the parameter.
- 4. Insert the selected display parameter in the function overview.



A click with the right-hand mouse button on a selected parameter opens a selection menu where the authorisation level can be changed.

Elements are deleted in the function overview within the same manner:



### **Function overview UVR16x2**

The function overview for UVR16x2 is created with the **TA-Designer** program.

# CAN data logging

#### x2 devices

#### Minimum versions:

#### C.M.I. 1.25

#### Winsol 2.06

Up to 64 analogue and 64 digital values can be defined for CAN datalogging for x2 devices.

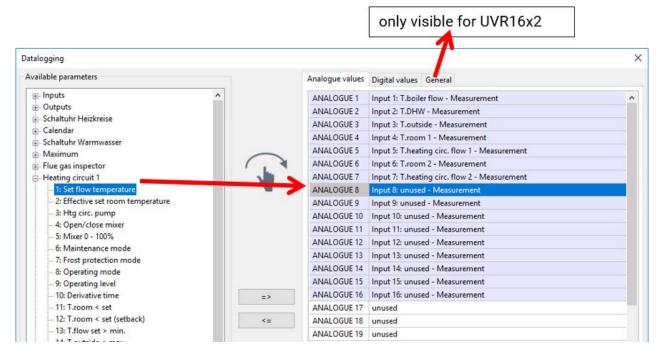
The left column shows all the available parameters that can be added to the right column. The right column has the tabs "Analogue values" and "Digital values". Therefore, when new logging values are inserted, it is important to check whether the value is an analogue value (numerical value) or a digital value (ON/OFF).

#### Procedure for linking a parameter into data logging:

#### There are 2 options:

1. Selection of the value to be logged from **Available parameters** on the left and dragging it to the logging value where it should be displayed (drag & drop).

**Example**: The set flow temperature for heating circuit 1 is to be logged as analogue value 8.



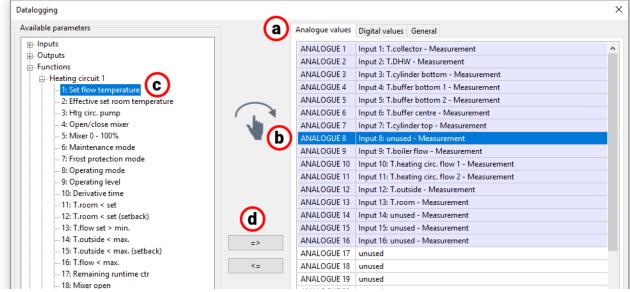
It is possible to select several values simultaneously using the Shift or Ctrl key.

ANALOGUE 7 Input 7: T.cylinder top - To remove values, select them and press the "Delete" key on the PC key-ANALOGUE 8 unused board to set them to "unused".

ANALOGUE 9 Input 9: T.boiler flow - 1

- 2. Using the arrows to overwrite or delete the entries in the list on the right
  - a) Select the data type in the right column (analogue/digital).
  - b) Highlight the **position** where a new parameter is to be inserted into the list.
  - c) Select the parameter which is to be incorporated from the list of "Available parameters".
  - d) Insert the selected parameter into the list on the right by clicking =>. After the parameter has been inserted, the following parameter is automatically selected.

Example: Inserting the heating circuit set flow temperature into "Analogue values" as "Analogue 8"



It is possible to select several values simultaneously using the Shift or Ctrl key.

Selected values can be reset to "unused" by clicking the back arrow (<=).

#### Datalogging on the SD card of the UVR16x2

#### Minimum controller version: V1.24

Under the "General" tab, you can define whether the logging values are stored on the controller SD card and if so, at what intervals.

1	Analogue values	Digital val	ues	General
	ANALOGUE 1	Input 1: ur	nuse	d - Mussurer
	Analogue values	Digital val	ues	General
	Datalogging	on SD card	Yes	
	Interval time		00:0	00:30 [hh:mm:ss]

**Example**: datalogging on the controller SD card has been selected. The interval time is 30 seconds.

### UVR1611

For CAN data logging, two data records can be freely defined. One data record comprises 16 analogue and 13 digital parameters.

The left-hand column shows all available parameters that can be added to the data record in the right-hand column. In this respect the following details must be observed:

#### Speed stages of outputs 1, 2, 6 and 7:

To record the speed stage of an output, the digital parameter with the same number must be allocated to the output in data record 1.

#### Heat meter functions:

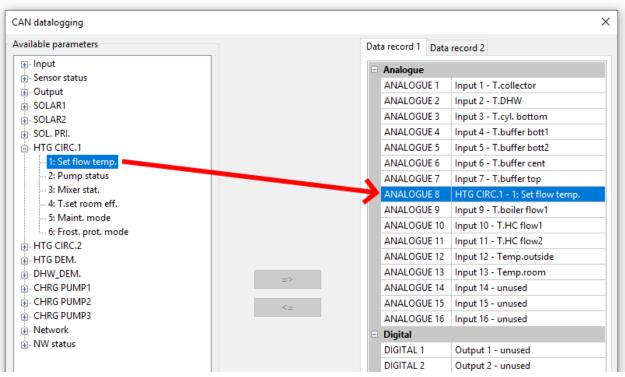
The output variables of the heat meter functions are, according to the sequence in the function list, **automatically** linked to the two data records (heat meters 1 and 2 in data record 1, heat meters 3 and 4 in data record 2). Although Winsol can log the output variables in a data record, they are displayed with the incorrect unit (°C). When correspondingly high values are reached, these can no longer be displayed correctly in the diagram and are no longer meaningful.

#### Procedure for linking a parameter into data logging:

#### There are 2 options:

1. Selection of the value to be logged from **Available parameters** on the left and dragging it to the logging value where it should be displayed.

**Example**: The set flow temperature of heating circuit 1 should be logged as analogue value 7 of data record 1

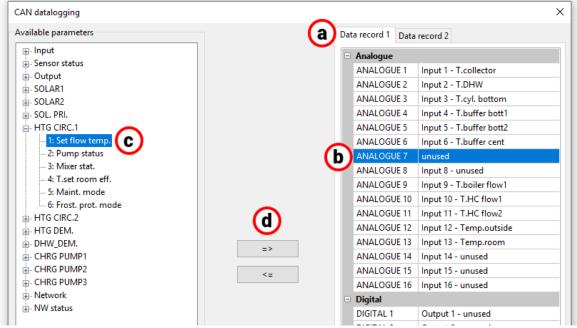


To remove a value from the data record, select it and use **Del** on the PC to set it to **unused**.

ANALOGUE 6	Input 6 - T.buffer cent
ANALOGUE 7	unused
ANALOGUE 8	HTG CIRC.1 - 1: Set flow temp.

- 2. Use of the arrows to overwrite or delete the entry in the data record
- a) Select the **data record** to be defined from the right-hand column.
- b) Select the **position** in which a new parameter is to be inserted in the data record.
- c) Select the parameter that is to be inserted in the data record from the list of **available parame-ters**.
- d) Use => to insert the selected parameter into the data record. After insertion of the parameter, the data record automatically selects the following parameter.

Example: Inserting the set flow temperature of heating circuit in data record 1 as Analogue 7



The back arrow (<=) can be used to reset a selected value in the data record to **unused**. The **Master node** is the node number of the C.M.I. or BL-NET.

# Simulation

As of version 1.16, function data can be simulated directly in TAPPS2.

Clicking the "Start simulation" button (on the taskbar at the top) simulates the current programming.

The x2 simulator (minimum version 1.38) must be installed on the PC. (<u>ta.co.at/download/soft-</u> <u>ware/</u>)

### Example

1 2	Solar control Solar 1	1 Solar circuit on		)ff
S1 T.collector	Collector temp.			
S2 T.cylinder bottom	Reference temp.		· · · · · · · · · · · · · ·	
S3 T.cylinder top	Limit temp.			

- 1. Value of the sensor input. Click on the numerical value to change it.
- 2. Digital input variable. The status of digital values is displayed here (e.g. On/Off).
- **3. Status of the output variable.** The output variable of the function is displayed. It cannot be changed manually as it reflects the actual result of the function calculations.
- 4. Status of the output. The status of the output is displayed.

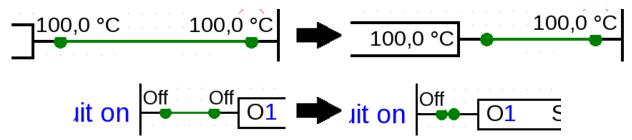
The programming cannot be changed once the simulation has been started. If, for example, a new function needs to be created, you first have to exit the simulation. This does **not** apply to simulation values of input variables, fixed values, etc.

Double-click on a function to view the values of all output variables, even if their pins are hidden in the programming:

np	ut variables Pa	rameters	Output variables	
•	Solar circuit			^
	Aktueller Wert	Off		
	Display pin	$\checkmark$		
	Output	1		
	Maximum limi	it		
	Aktueller Wert	Off		
	Display pin			
	T.coll. < T.coll.	max.		
	Aktueller Wert	On		
	Display pin			

Parameters of functions, fixed values, inputs and outputs can be changed, but you have to exit the simulation to apply the changes.

Values that do not have enough space to be displayed are added to the corresponding drawing object:

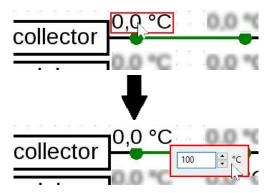


#### **Changing values**

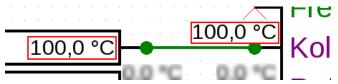
During the simulation, values such as sensor inputs and similar can be adjusted to simulate the control effect of the programming under various circumstances. In addition to sensor inputs, input variables, etc., values appear that you can click on.

#### Analogue values

Example: Sensor input



The value can be changed with the arrow buttons, scroll wheel or by entering numbers directly. The input variable to which the sensor input is connected adopts this value:



#### **Digital values**

Example: Digital fixed value F1 Enable function If you click on the field with the status of the digital value (e.g.: "Off"), it is changed (e.g. to "On"): F1 Enable function Pulse Example: Resetting a heat meter using a fixed pulse value F1 Reset off Meter reset The field changes briefly to the pulse that is issued and then returns to the normal display.



Pulse inputs for e.g. wind speed are set in the same way as analogue inputs.

### **Temporal mean value calculation**

In simulations, temporal mean value calculation is disabled. This means:

- No temporal mean value calculation of the outside temperature in heating circuit control, cooling circuit control or individual room control.
- Analogue function, filter mode: simulation with "filtering time" = 0.
- No temporal mean value calculation for inputs.

### **CAN bus simulation**

CAN bus inputs and outputs can also be simulated for all devices.

For this purpose, multiple programming functions (up to 62) are opened in the same TAPPPS2 window.



The CAN bus inputs and outputs are created in the usual way in the respective programming functions. If the node numbers and output numbers match, the program automatically recognises the relationship.

# CAN output from node 32 and output number 11

	· · · · · · · · · · · · · · · · · · ·
RSM610 ×	UVR16x2
Hydraulics	Programming
· · · · ·	AN Analogue 11 ( 32 / 11 ) O2 content

The value is read in again as a CAN input (node 32, output number 11)

	· · ·		
RSM610	UVR16x2 ×		
Hydraul	ics Programming		
	CAN Analo	gue 33 ( 32 / 11	.)
/0	CO2 conten	t	

The node number of the device which sent the value to the CAN bus (on the left in the example) is changed via **File > Settings > Device Settings > CAN Bus**.

If the simulation is now activated in **both** projects with **b**, the value is synchronised between them.

RSM610 🗴 UVR16x2	RSM610 UVR16x2 ×
Hydraulics Programming	Hydraulics Programming
CAN Analogue 11 ( 32 / 11 ) CO2 content	CAN Analogue 33 ( 32 / 11 ) 1000 ppm CO2 content

The simulation of the CAN bus inputs and outputs does not take into account their send conditions.

#### Date/time/Mean value calculation

The date and time can either be taken from the PC or set by the user.

If the simulation is active, a bar appears at the bottom:



In this example, the values are greyed out and cannot be selected. The date and time are therefore taken from the PC.

Click on the clock symbol 🙆 to change the date and time and use them for the simulation.

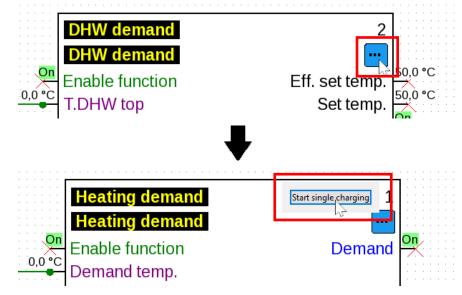
Changing the time only affects functionalities that actually relate to the time. Values such as remaining runtimes or timers are not influenced by the time.

### **Buttons**

Many functions have buttons which can be pressed on the controller itself in the menu or in the function overview, e.g. "**Start single charging**" for the **DHW demand** function.

As of version 1.19, these buttons can also be actuated in simulation mode.

Example: Start single charging



A single click triggers the charging process.

#### Example: Meter reset



A single click resets the meter.

Example: Blind control

· · · · ·	Blind control	5	
	Blind control 1		
On	Enable function	Switch to automatic mode	Off 67%,
Off	Enable auto mode Open blind	Close blind	70%,
Off	Close blind	Fully open blind	On
X	Fully open blind	Fully close blind	0 sec
X	Fully cls blind	Set blind horizontally	· · ·

The "Open blind" and "Close blind" buttons are actuated by holding down the mouse button. The other buttons are clicked once.

# **Generating function data and documentation** Function data

New	Ctrl+N	B B 5 0 4
Open	Ctrl+0	MARKA IN CONTRACT
Close		10x2.00w X 1011.00w
Close all		Hydraulics Programming
Save	Ctrl+S	
Save as		
Save all		
Settings	>	
Page setup		
Page view		
Print	Ctrl+P	
Import	>	
Export	>	Function data
Files last opened	>	Documentation
Exit	Alt+F4	Screenshot

With **File / Export / Function data** or **File / Export / Documentation**, the function data (\*.dat file) and the documentation (\*.txt file) can be generated.

If the programming is faulty, an error log listing the errors is generated before generating the function data.

Function data and the documentation can be generated only after all errors have been removed. Example of an error log:

Function data Error log	
Error: Several DL inputs read the same parameter (address 1, index 1): Input 1 - T.collector 1 Input 2 - T.collector 2	^
Functions: 0 of 128	
Result: 1 Error, 0 Warnings	

Before saving the function data, the minimum requirements for the operating system and the serial number must be checked and observed in accordance with the specification of the controller type. **Example UVR16x2:** 

ienerate functi	on data		×
Function data	Error log		
Requirements:			
Operating sys Serial number	em: V1.38 or higher every device		^
<			~
-		ОК	Cancel

### Documentation

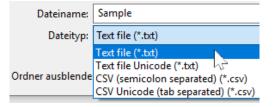
New	Ctrl+N	B B 5 0 B
Open	Ctrl+0	
Close		10x2.tdw × 1611.tdw
Close all		Hydraulics Programming
Save	Ctrl+S	
Save as		
Save all		
Settings	>	
Page setup		
Page view		
Page view Print	Ctrl+P	
	Ctrl+P	
Print		Function data
Print Import	>	Function data Documentation

Unused parameters can be filtered out before saving the documentation.

#### Example UVR16x2

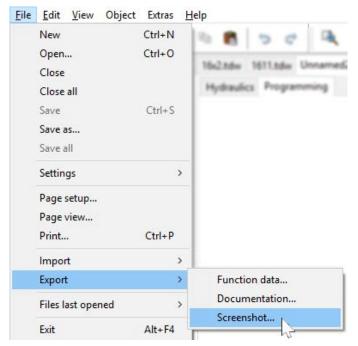
/UC	umentation Error log		
	Parameters to be listed		^
	Requirements	$\square$	
•	Inputs	$\square$	
	Without unused inputs		
•	Outputs		
	Without unused outputs		
	Functions		
	Without unused variables		
•	Fixed values		
	Without unused fixed values		
-	DL bus		
	Without unused DL variables		
	CAN bus		~

The documentation can be created in different file formats as required:



### **Exporting a screenshot**

This functionality makes it possible to export the **visible section** of the drawing area as a PNG or JPEG file.

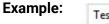


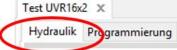
Export scre	enshot	×
Size:		
Width:	1024	Pixels
Height:	540	Pixels
ОК		Cancel

The required image size is entered to fit the display screen. The height/width proportions are kept the same.

# **Drawing function under "Hydraulics"**

A hydraulic drawing is created under "Hydraulics", which is displayed under the file name.





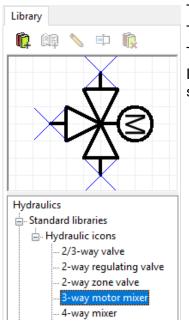
**Navigating** within the drawing area (selecting, zooming, moving, bringing forward/backward, aligning, copying) and dealing with **links** are exactly the same as under "**Programming**" and are described there.

Note on lines:

If the Shift key is pressed before a **diagonal** line is completed, it changes to an **orthogonal** line (horizontal or vertical).

# Libraries

### **Standard libraries**



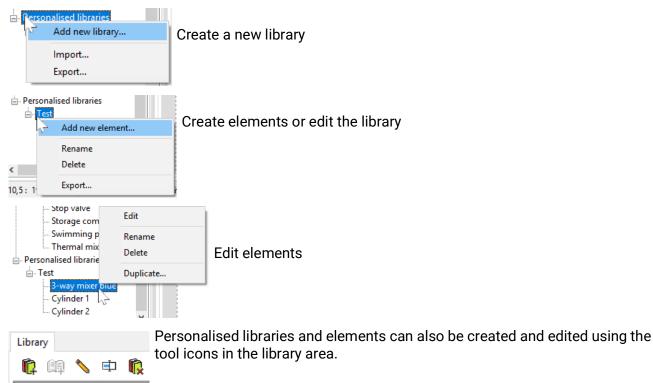
The program provides various hydraulic icons in a **standard library**. These icons are available on the left-hand side in a search tree. The selected element is displayed in the preview window above. Icons are inserted into the drawing in the same way as objects are inserted in the programming area.

### Personalised libraries in hydraulic drawings

> --- Cylinder 1 --- Cylinder 2

It is also possible to create **personalised libraries** with personally designed elements. In the example, the personalised library "**Test**" and 3 personalised elements have already been created. The selected element is displayed in the preview window above.

Editing is possible by right-clicking the relevant element:



### Shared libraries in hydraulic drawings

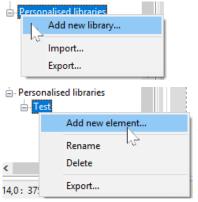
The principles of shared libraries are described on page 63.

While a hydraulic symbol is being edited in a library, that library is unavailable to all other users.

During this period, no other users can make changes to the library.

### **Creating and editing library elements**

The **standard library** elements cannot be changed in the library. Standard elements **in the drawing area** can be changed using the lcon editor.



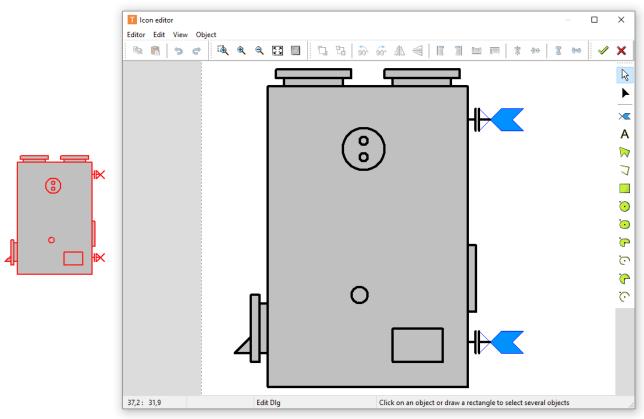
**Personalised** library elements can only be set up once a personalised library has been created.

The new element can then be set up. This must firstly be given a **name**, then the **Icon editor** opens.

#### **Icon editor**

The "**Icon editor**" opens when new personalised elements are created or an element in the drawing area is **double clicked**.

Example - "Solid fuel boiler":



If you want to edit a **standard element** and save it as a **personalised element**, right-click the standard icon in the list to *duplicate* it.

Alternatively, you can first position the standard element and then double-click on it to open the editor shown above. An edited icon can be saved to a personalised library under Editor  $\rightarrow$  Save as + Exit.

#### **Selection mode**

$\sim$	N
	N2
$\sim$	2
	-12
Select selection n	node

**Single, selected** elements can be brought forward or backward, rotated, flipped or aligned using the **toolbar at the top**.

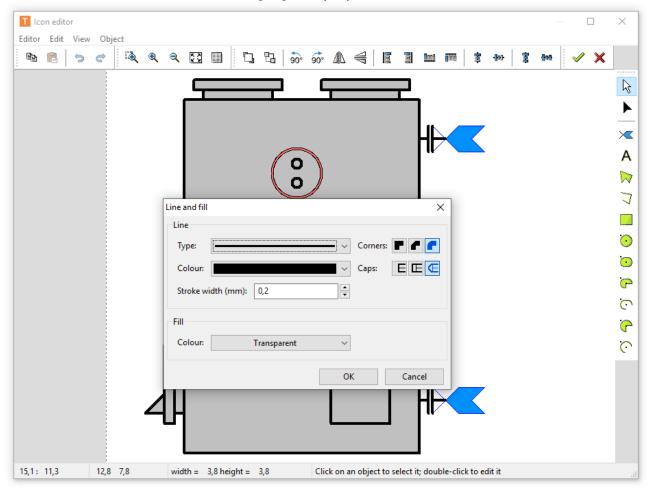


Further drawing elements (lines, rectangles, circles, arcs, ellipses, texts) can also be inserted using the **toolbar on the right**.

In **selection mode**, **double clicking** a line or an area element (e.g. circle) enables you to change the type, colour and shape of the line or the fill colour of the area element:

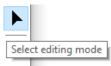
Example: Circle in the boiler

The element that has been clicked is highlighted (red)

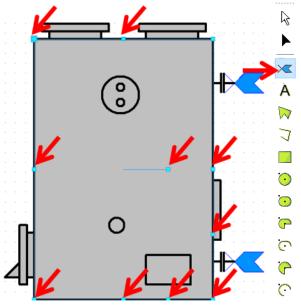


#### **Editor mode**

In **editor mode**, the **shape** of individual elements of the icon can be edited.



Example: Solid fuel boiler

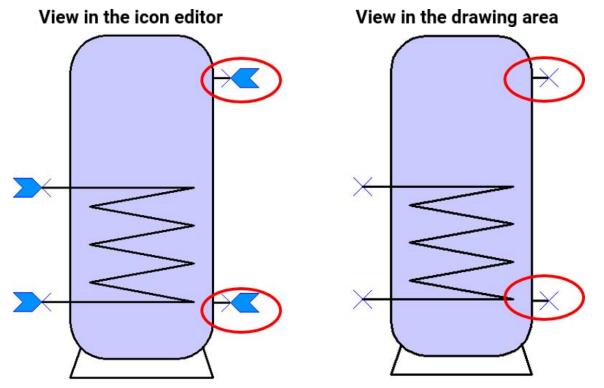


Click the rectangle in **editor mode**. Now the shape can be changed by dragging the displayed marker points.



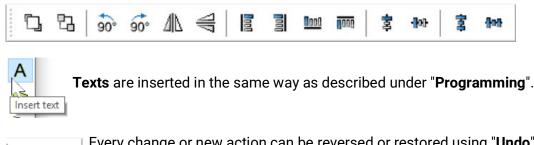
"**Pins**" are a special feature. These are connecting points for lines, which enable a precise link to the elements.

**Example**: Extending the cylinder icon using 2 pins



This amended icon could now be copied and saved as a new library element in a **personalised library**.

Selected elements can be brought forward or backward, rotated, flipped or aligned using the toolbar at the top.



Every change or new action can be reversed or restored using "Undo" or "Redo".

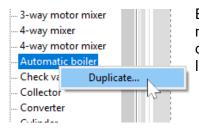
Every change in the **Icon editor** must be completed with the tick 🛷 .

If the changes should not be saved, click  $\mathbf{X}$  .

#### **Duplicating elements**

5

¢



Elements can be duplicated by clicking the right mouse button. Elements from standard libraries and personalised libraries can be duplicated, but the duplicated element is always saved in a personalised library.

### Creating a hydraulic drawing

In addition to the hydraulic icons, **further drawing elements** (lines, rectangles, polygons, texts) <sup>–</sup> can be inserted and edited using the **toolbar on the right**.

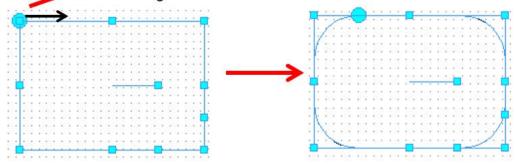
#### Example: Rectangle

Once the rectangle has been inserted and "drawn", you are in **editor mode.** Editor mode is recognisable by the shape of the cursor: +

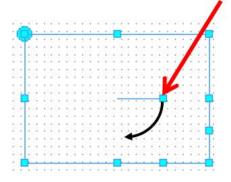
An object which has already been inserted can be edited further by starting editor mode  $\mathbf{k}$  in the toolbar on the right and clicking the object.

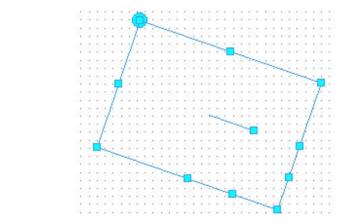
The shape of the rectangle can be changed by dragging the marker points

By dragging the marker circle towards the inside, the corners of the rectangle can be rounded.

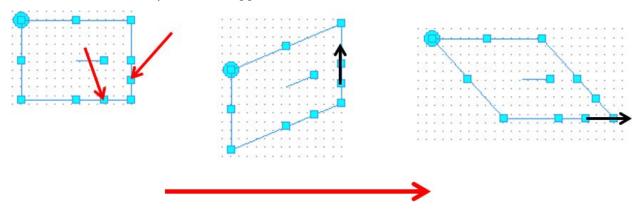


The rectangle can be rotated using the marker point **inside** it.





If one of these marker points is dragged, rhombuses are created.



The line type, colour and thickness, as well as the shape and fill colour can be changed by double clicking the rectangle in **selection mode**.

Example.			
	·····		
Line and fill	· · · · · · · · · · · · · · · · · · ·	<	
Line			
Туре:			
Colour:	V Caps: E 🗉 🧲		
Stroke width (mm):	<b>1</b> ,5 ▼	$\longrightarrow$	
Fill			
Colour:	~		
· · · · · · · · · · · · · · · · · · ·	OK Cancel		

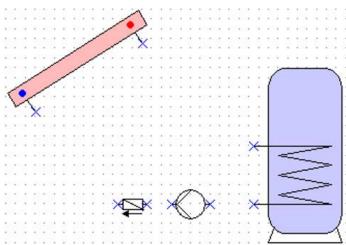
Object	Extras	Help								
An	Arrange									
Ali	Align									
Line and fill										
Fo	nt	13								

A global setting for these elements can be selected for **all hydraulic drawings** via the menu item **"Object/line and fill...".** These settings are retained, even if the program is restarted.

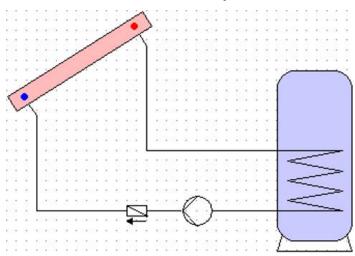
### Example of a simple hydraulic drawing

As an example, the individual steps for creating a simple hydraulic drawing (solar thermal system) are described below.

1. Place the **hydraulic icons** "Collector", "Cylinder", "Pump" and "Check valve" in the drawing area by dragging them from the search tree and aligning them on the grid.



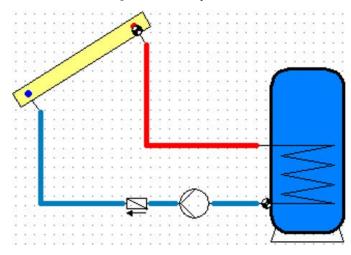
2. Create the link lines between the pins as described under "Programming".



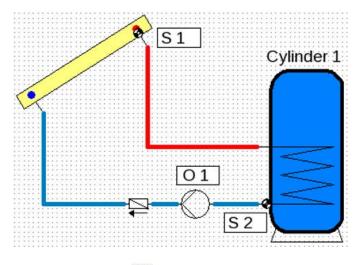
3. **Optional:** Edit the lines and areas (colour/line thickness) by double clicking the element or going to the lcon editor.

· · ·	· · ·			· ·	· · ·	  	· ·	• •	•	j -	•		• • • • •			4	<	-	
Line	and	fill								 		 							×
	ine Type:											 ~	C	Corn	ers:	F	1	•	•
1	Colou	r:										~	C	aps	:	E	E		
	Stroke	wi	dth	(m	nm)	: [	0,2					•							

View after editing lines and hydraulic elements:



4. Adding sensors and designations



5. Finalise by saving 📳

#### Legal notice

These operating instructions are protected by copyright.

Use outside the copyright requires the consent of the company Technische Alternative RT GmbH. This applies in particular to reproductions, translations and electronic media.

# **Technische Alternative RT GmbH**

A-3872 Amaliendorf, Langestraße 124

Tel.: +43 (0)2862 53635

Fax +43 (0)2862 53635 7

E-Mail: mail@ta.co.at

--- <u>www.ta.co.at</u> ---

©2023

C E