

Functions of the S-Web-Editor

Workshop days 1 and 2

Step-by-step tutorial

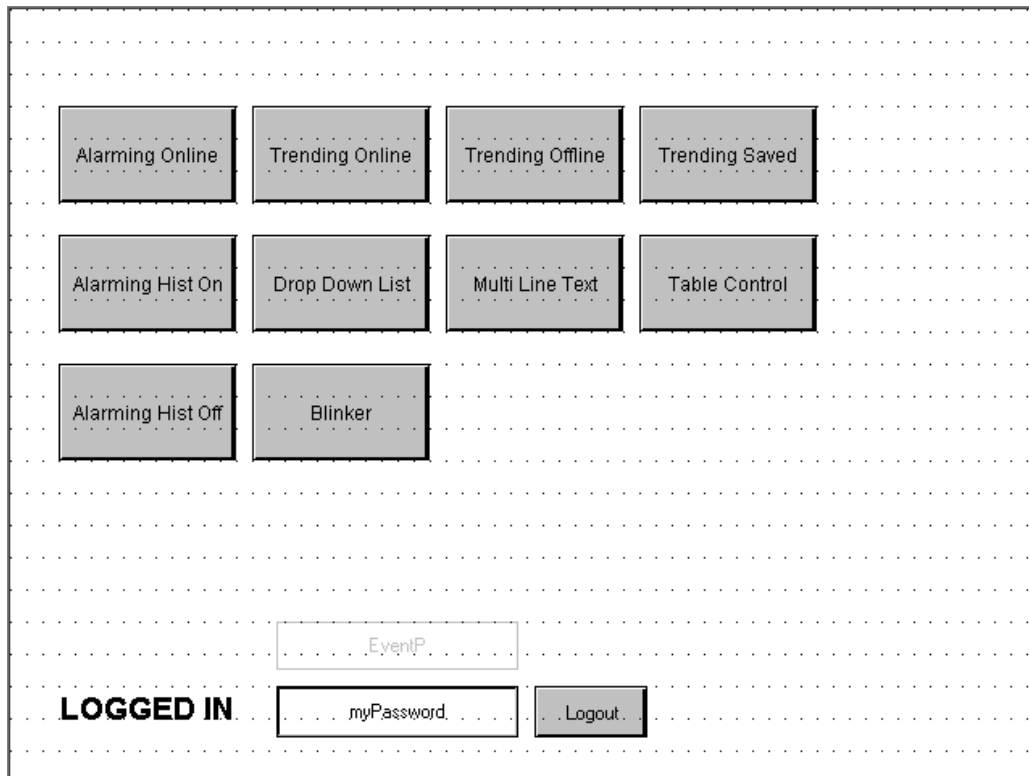
Project history

Date	Author	Modification
10.04.2008	TCS / pc	Initial Version
16.01.2009	TCS / pc	Update to S-Web Editor 5.14
15.09.2009	TCS / pc	Update to S-Web Editor 5.14.27
16.01.2010	TCS / pc	Added „Scalable S-Web Editor projects“ (V5)

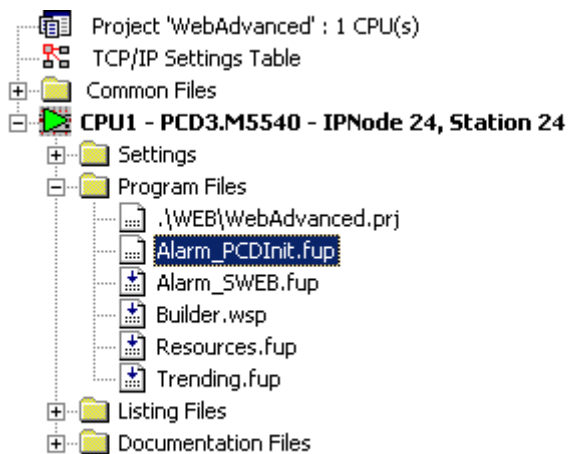
Contents

1 Start page with links to the individual subpages.....	3
2 Fupla projects for variable resources, alarming, trending, WebEditor project and WebBuilder	3
3 Overview of FUPLA projects.....	4
4 Structure of macro libraries.....	5
5 Conference room with date, time and temperature	6
6 Make a macro out of the conference room	8
7 Include multilingual feature	12
8 Implement language switching.....	13
9 Define the start project page	14
10 Generate the start.htm page	14
11 Using the WebBuilder	15
12 Alarming.....	16
13 Alarm macro online list	18
14 Alarm macro online history	20
15 Scalability of a web application	21
16 Trending.....	22
17 Online trend.....	23
18 Offline trend in DBs.....	25
19 Offline trend into flash file system.....	31
20 Login.....	36
21 URL jump with transfer of password level.....	38
22 HEAVAC macros (SWeb templates)	40
23 Drop-down list.....	43
26 Forced page change - jump event.....	45
27 Automatic URL jump for inactivity	46
28 Blinker	47
29 'Table Control' macro	48
30 Moving painters in x and y axis.....	50
31 Rotation of polygons	51
32 Tachometer macro	52
33 Button for URL jump with address in argument	53
34 Foreground.....	54
35 Special links, actions and jumps with WebEditor buttons	57

1 Start page with links to the individual subpages



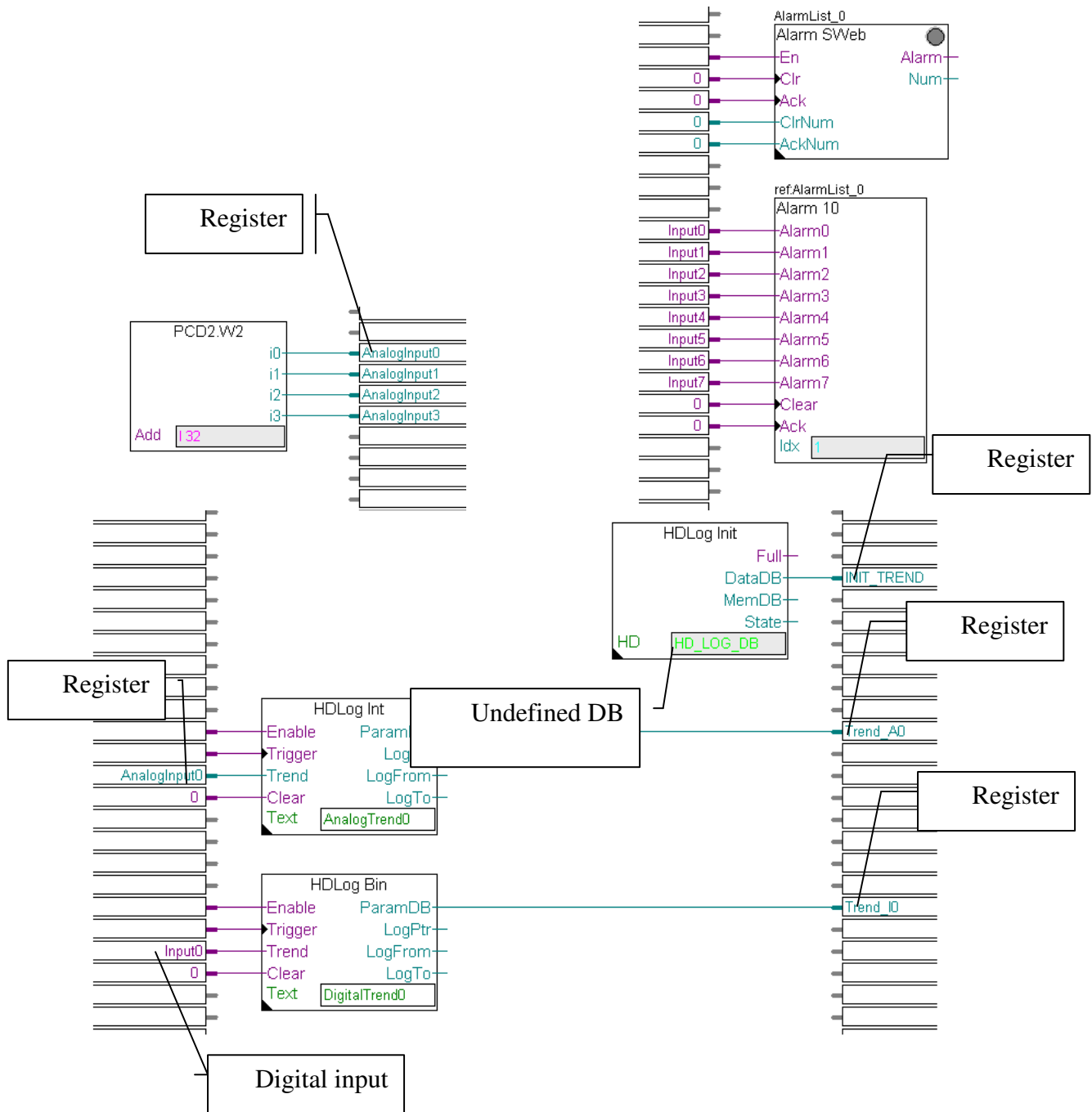
2 Fupla projects for variable resources, alarming, trending, WebEditor project and WebBuilder



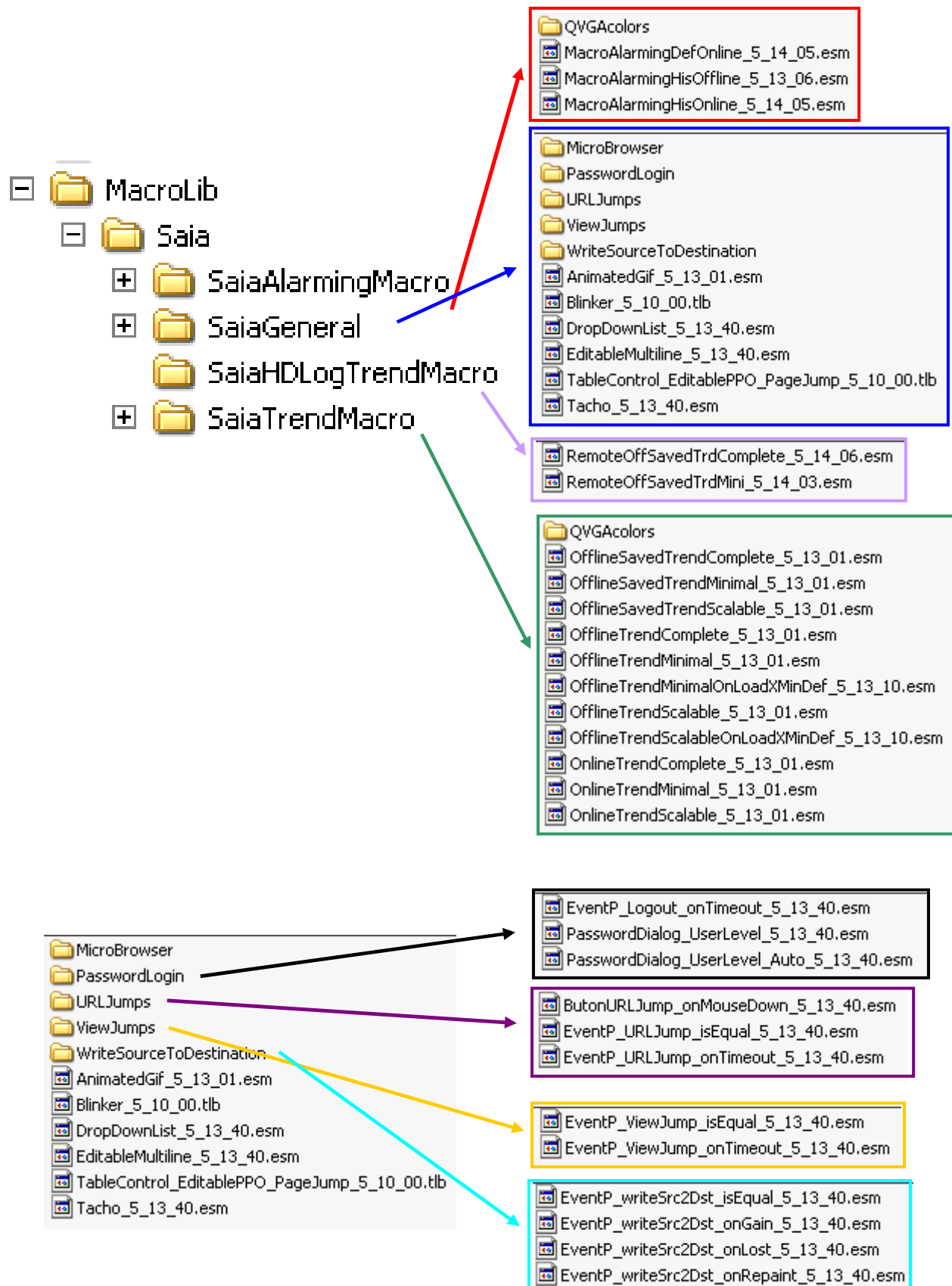
Notation to generate resources:

Digital input 0..7 I 0
Analogue input 0..3 R

3 Overview of FUPLA projects

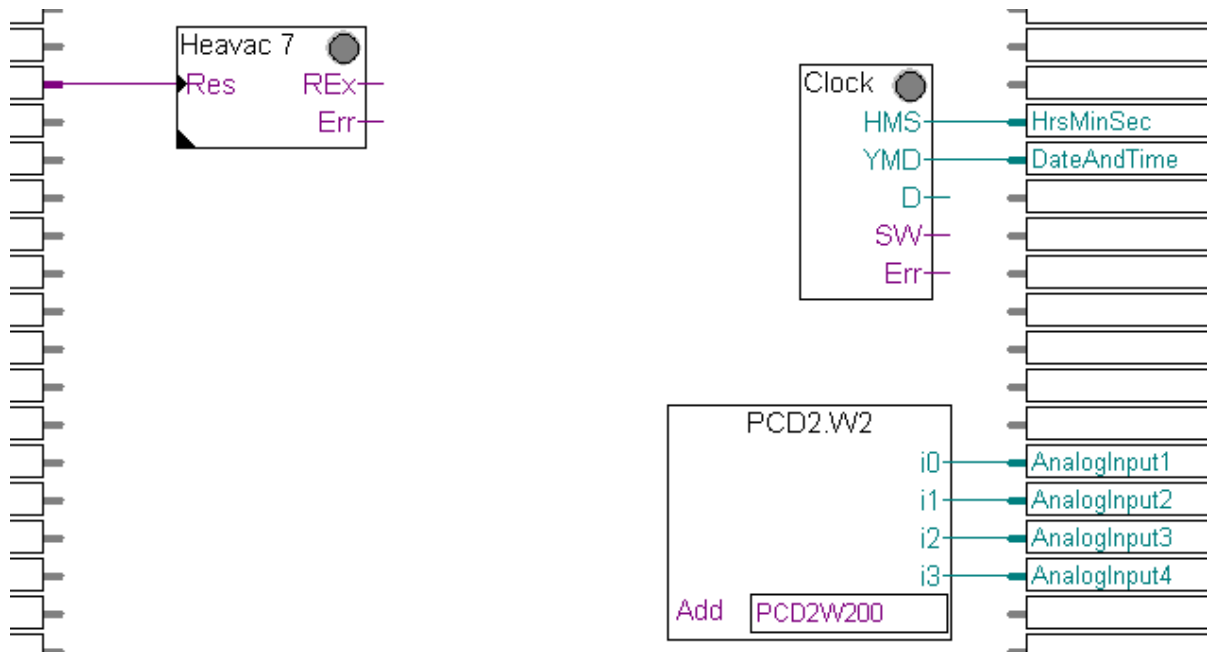


4 Structure of macro libraries



5 Conference room with date, time and temperature

FUPLA



Group the variables of the conference-room

Conference	GROUP	
AnalogInput0	R	
Clock	R	
Date	R	

WebEditor:

- Define start page Main.teq
- Define background page Background.teq with Home button
- Define conference room page with picture of conference room light/dark and on/off button for light.



LIGHT ON/OFF.

Temp. AnalogInput1

Home

DateAndTime HrsMinSec

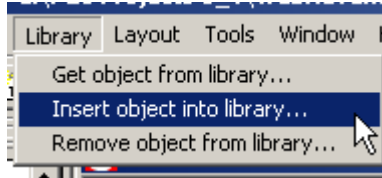
Select inclusion of room temperature (under PPO format, choose DEC.1 and under unit specify °C) and select date and time (under PPO format DD.MM.YY and HH.MM.SS)

PPO Name	Min	Max	Format	Unit
Conference.AnalogInput0			DEC.1	°C
Conference.Clock			HH:MM:SS	
Conference.Date			DD.MM.YY	

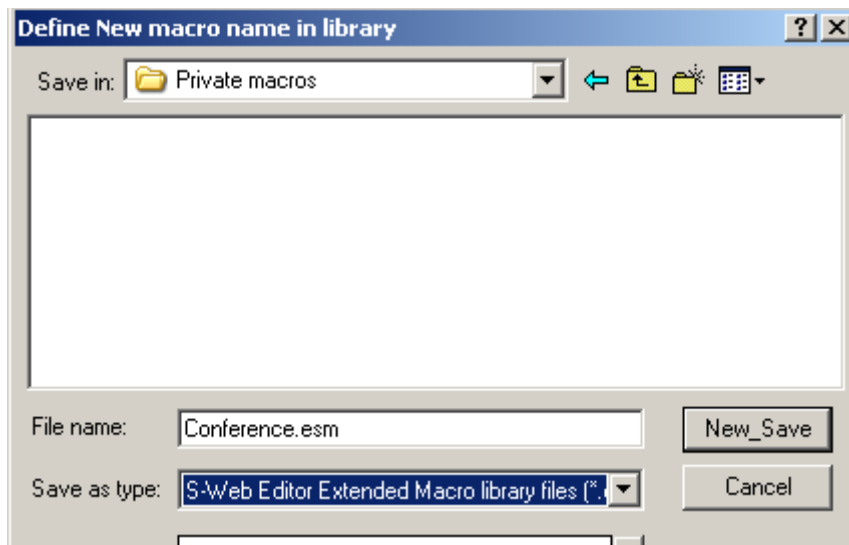
- Link in Main.teq to conference room.

6 Make a macro out of the conference room

Select the animated picture (*.gif), the on/off light switch, and the indicator of date and time. Now, with the right-hand mouse button, 'Group' these elements together and via the library function :



Save the group as a macro in a personal directory.

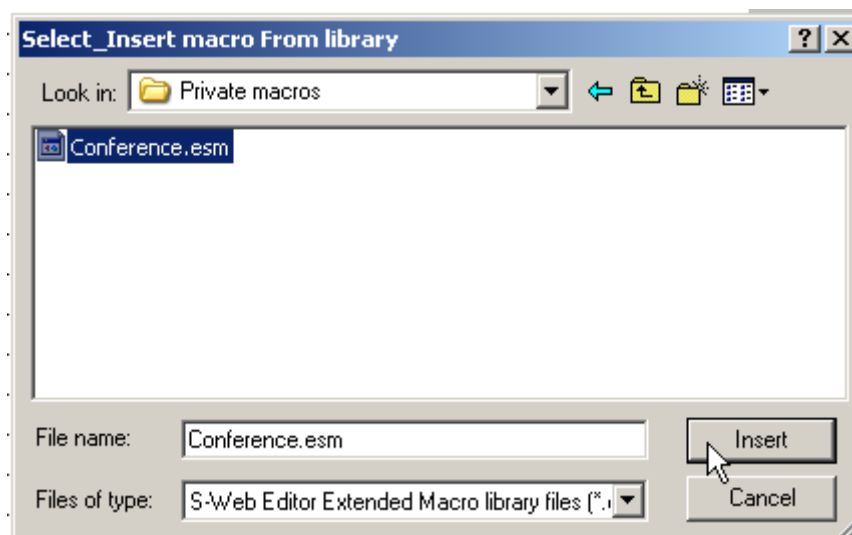
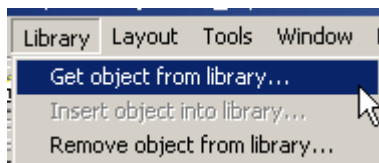


Import conference room macro and apply resources to it

Load the conference room macro onto an empty page, using either the insert macro shortcut



or the 'Get object from library' option

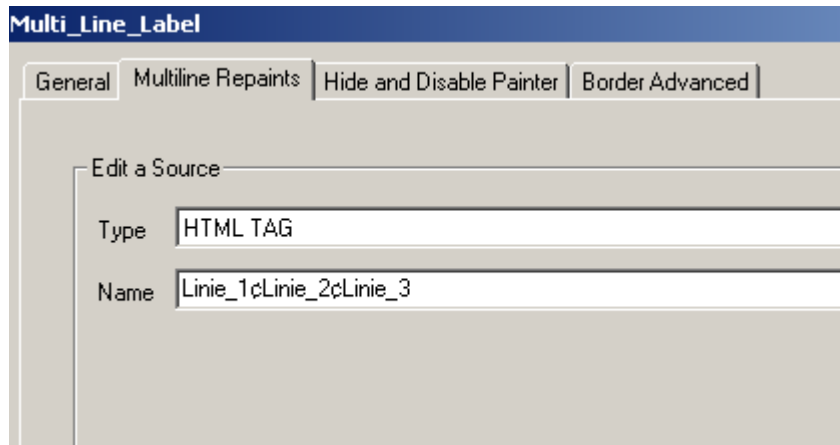


Double-click on the macro to enter the cross-reference list. It is now possible to adjust each variable individually. However, it is much easier, more reliable and efficient if the PG5 symbol editor is used in advance for the proper, rigorous definition of variables as a group.

Click on 'Macro Find/Replace'

Multiline text

Select Multiline Painter and draw box.



If defined as HTML TAG, the lines for Line_1 and Line_2 are inserted. The individual lines are entered in the WebEditor, with '¢' as the separator. Generated by the shortcut 'Alt Gr' and '8'. 'Line_1', 'Line_2', etc. are just identifiers. The text to be displayed is stored in a *.csv file.

Texts are read and displayed from whichever *.csv file corresponds to the chosen country code.

Press the 'CSV' button and specify a *.csv file. This is the default csv file.

*.csv file:

Pressing the 'CSV' button generates a *.csv file that collects all the HTML TAGs of any project, including those of its macros. In this way, the first *.csv file is defined. The default *.csv file to be used when loading a project is specified under 'Project -> Project configurations ... -> Project Applet Advanced'. Throughout engineering, new HTML TAGs will be continuously added to this *.csv file whenever the 'CSV' button is pressed.

Project Configurations

Project - Teq Configurations | Project - Applet Advanced | Project - Build Advanced | Project - ...

applet params

☐ Debug ☐ Date/Time Display Mode PM AM

☒ Order per view ☒ Encode Special Chars

period (ms) : Decimal Symbole :

main TEQ :

☒ Message Box

Message box TEQ

Nbr Errors / Warn :

Warn display time : s

☒ HTML Parameters in csv file

csv file :

static csv file :

Action mode :

Browsing mode :

Mirror PPO Name:

Loading mode :

Default Loading mode that loads all views from server at applet's init phase

7 Include multilingual feature

Other languages can be added to the project by selecting 'Add' under 'Project -> Project configurations -> Project – Teq Configuration'. Translations are carried out in the newly generated *.csv files. The method of declaration is always the same:

Identifier1;Alarm1

Identifier2;Alarm2

...

Whereby 'Identifier1' is replaced with 'Alarm1' while displaying the website.

The newly generated HTML TAGs are now continuously added to the registered and subscribed *.csv file list.

Project Configurations

Project - Teq Configurations | Project - Applet Advanced | Project - Build Advanced | Project - Advan

Teq Configurations

Width : 640 Pixels

Height : 480 Pixels

Background Color

Foreground Color

Font

Outline Width: 1 Pixels

Outline Style: Solid

background TEQ: Background.teq

foreground TEQ :

project settings

csv files (update): english.csv, french.csv, german.csv **updated list**

scalable html :

Saia HDlog Files : C:\PG5 Projects\1_4\WebAdvancedWorkshop\CPU1\Logging.HDF

Saia Project type: Saia Standard

8 Implement language switching

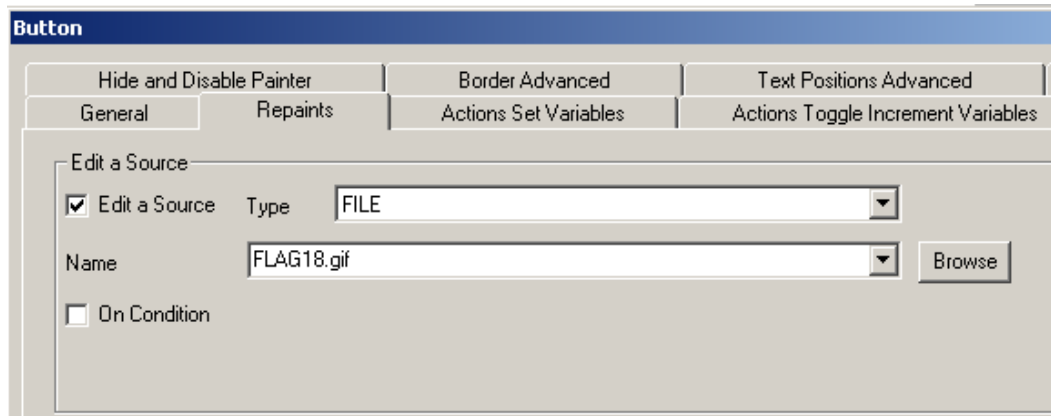
Define and position the flags as buttons on the background page.

German = FLAG18.gif

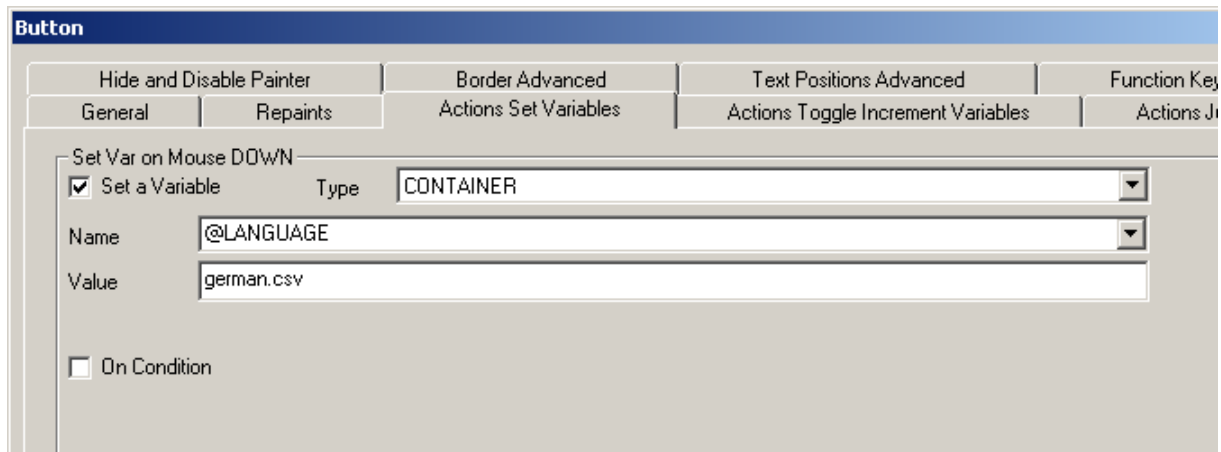
English = FLAG02.gif

French = FLAG17.gif

Button looks like a flag (image)

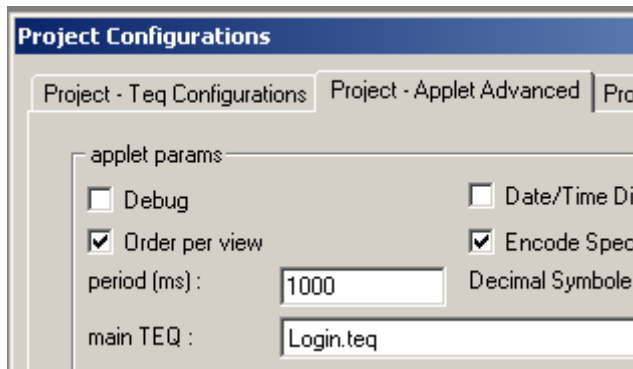


Button produces an action and sets the *.csv file to be used.



9 Define the start project page

Go to 'Project Configurations'. Select here: 'Login.teq'.



and

10 Generate the start.htm page

The entire project can be browsed via http with the  button.

If 'Start.htm' is selected, a browser entry of http:// followed by the PCD's IP address is sufficient. 'Start.htm' is loaded by default.
e.g.: http://192.168.12.105

WebEditor project build

In this way, project files are copied from the 'Web' folder to the 'Html' folder and the project can be locally tested on the computer.

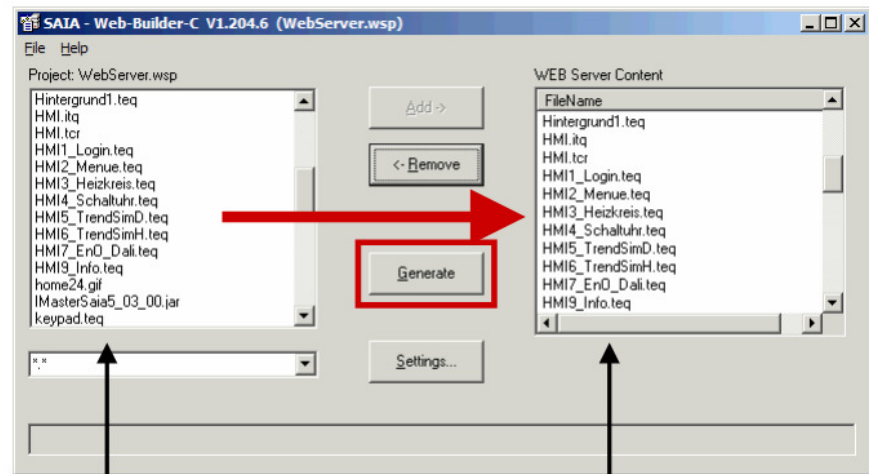
11 Using the WebBuilder

To enable the project to run autonomously on the PCD, all necessary files must now be stored on the PCD (unless WebConnect is being used). Project files can now be saved either as DBs in the PCD's user program (files selected using WebBuilder) or manually on a flash card via an FTP connection.

SCHRITT 1:
Web Server Content,
 d.h. alle Dateien des
 Web-Projektes
 auswählen, die im
 RAM der SPS
 gespeichert werden
 sollen

SCHRITT 2:
Generate ausführen,
 um die Dateien in
 einen DBX-Datenblock
 zu konvertieren

Anstatt **Generate** kann
 alternativ in der PG5
 ein **Rebuild All**
 ausgeführt werden



Alle Dateien im HTML-
 Ordner der CPU (im
 PG5-Projektordner)

Alle Dateien, die im
 RAM der SPS
 abgelegt werden

Muss in WebBuilder: *.tcr-Datei

STEP 1:
 Select all web server content (i.e. all web project files) to be stored in PLC RAM.

STEP 2:
 Press 'Generate' to convert files into a DBX data block
 Alternatively, instead of 'Generate', execute a 'Rebuild All' in the PG5.

Column left
 All files in the CPU's HTML folder (in the PG5 project folder)

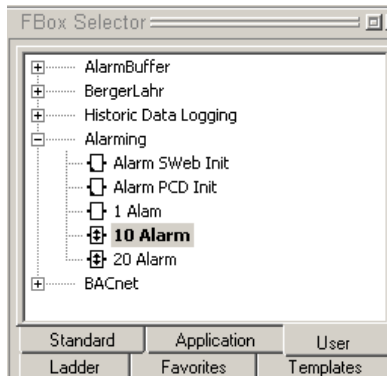
Column right
 All files to be stored in PLC RAM.

Mandatory in WebBuilder: *.tcr file

The *.tcr file must be loaded into the user program.

12 Alarming

The following FBoxes are available to support alarming.
Collective alarm and initialization FBoxes.



Alarming can be initialized with either the 'SWeb Init' or 'PCD Init' FBox.

'SWeb Init' works with alarm texts from a *.csv file. Alarm flags (open alarms, ACKs, ...) have fixed addresses as system variables.

'PCD Init' reserves alarm texts in the PCD or in a *.csv file. Alarm flags can be addressed manually.

Execute Clr and Ack

Select alarm number

myAlarmlist

Alarm SWeb

En

Clr

Ack

ClrNum

AckNum

Alarm Num

Assign right mouse button and name
System variables are reserved

Adjust: Alarm SWeb init

Read All Write all Set Defaults Info Help OK Cancel

[---- current alarm list

Number of alarms > 10

Remove alarms auto. when > Never

System clear acceptance > All

Clear the entire list Clear

Acknowledge the entire list Acknowledge

Usage of Clear/Ack flags > No

[---- Alarm history list

List is > Ring buffer

Numbers of history entries > 250

Max. alarm entries for history

Firmware

User

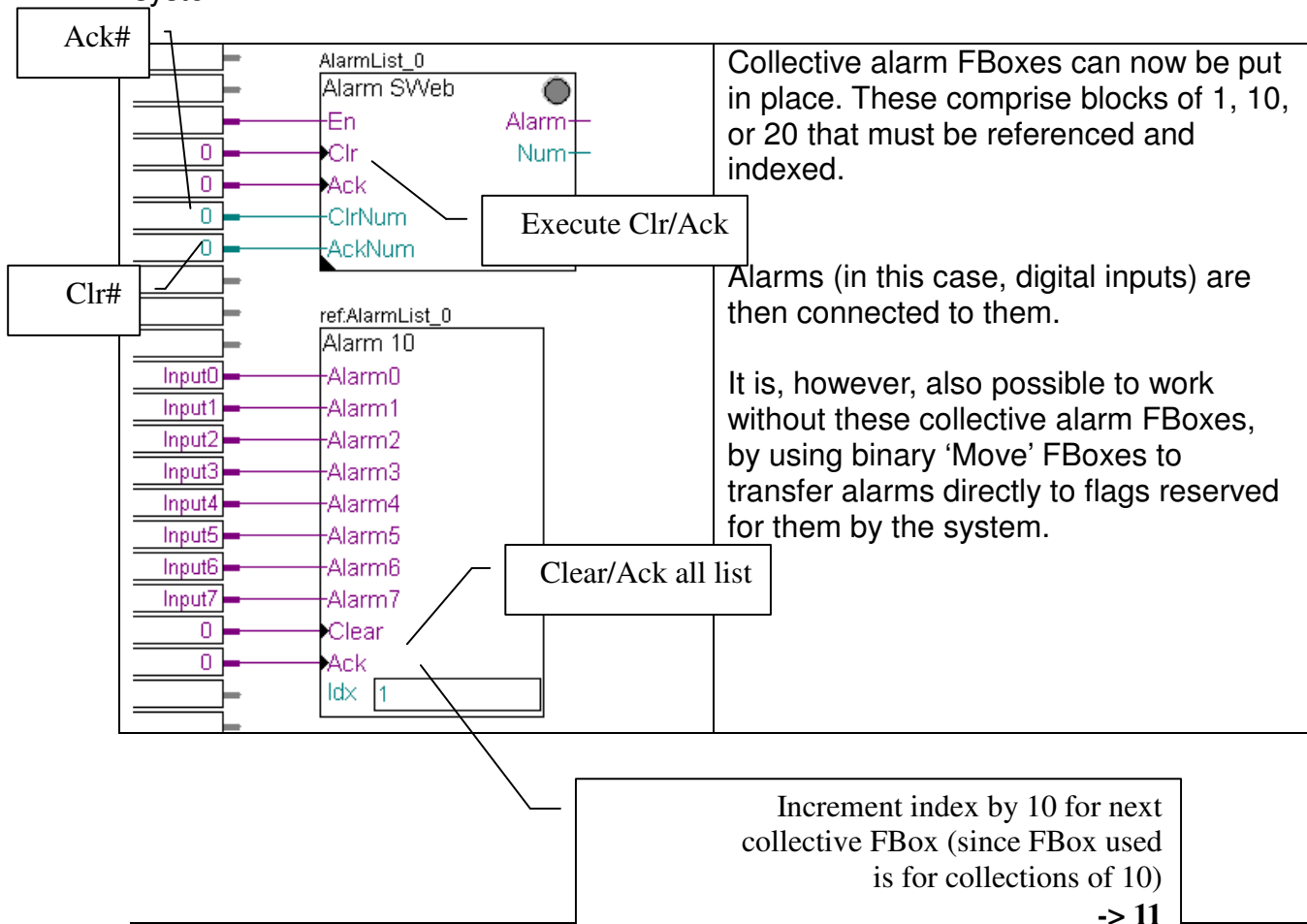
The 'Alarm SWeb Init' FBox is required to initialize the alarm buffer.
With this FBox, alarm texts displayed in the web browser are stored in *.csv files.

Number of alarms one flag is reserved for each alarm
Number of history entries max. count for alarm history

Explain how alarms are deleted by the system or user. Transparency!!

Group/Symbol	Type	Address/Value	Comment
S	GROUP		
A	GROUP		
Alarm	GROUP		
AlarmList_0	GROUP		
Flags	F	7549 [100]	Alarmflags
FlagsAck	F	7649 [100]	Alarmflags
FlagsDel	F	7749 [100]	Alarmflags
MaxAlarms		100	max. numbers of alarms
MaxHistory		250	max. alarms in history
Handle	R	2185	Handle-ID
MyName	Text	3002	Name of list

After the FUPLA program build, system resources are available for naming the collective alarm and addressing alarm flags. Addressing is therefore fixed by the system.



13 Alarm macro online list

The online alarm list displays the current status of each alarm.
This means that each alarm occupies one line in the online alarm list.

- Alarm ID
- Alarm text
- Time and date when alarm was last switched on
- Time and date when alarm was last switched off
- Alarm status, i.e. whether the alarm has been acknowledged or not
- Counter indicating how often the alarm has switched on

Users can **acknowledge** alarms (one/selection/all) and **clear** them (one/selection/all) from the list.

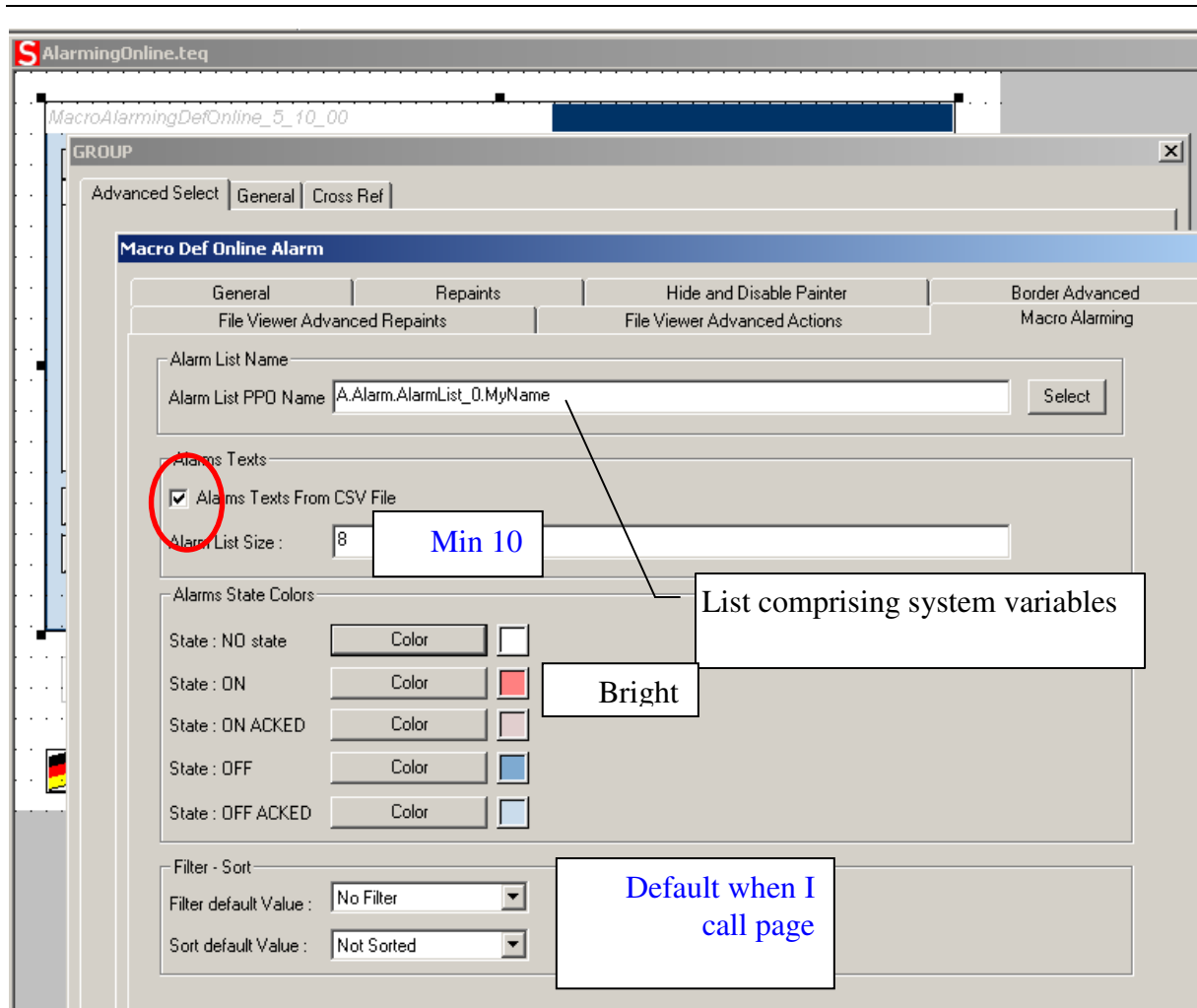
It is also possible to **filter** alarms (no filter, active alarms or alarm counter)

Sorting is also possible (unsorted or chronologically sorted)

These functionalities can be password protected. Ungroup the macro and modify properties of individual buttons.

HTML TAGs are listed in the *.csv file and can be modified there.

Under 'General', the entire macro can as a group be 'Disabled' or 'Hidden', e.g. through password protection.



Enter:

- Name of alarm group from system variables: 'A.Alarm.AlarmList_0.MyName'
- Number of alarms.
- Click on 'Alarm Texts from CSV File'. Alarm texts will be loaded from the *.csv file

Generate PPOs and declare 'A.Alarm.AlarmList_0.MyName' as a STRING.

Press *.csv button and adapt alarms into all languages.

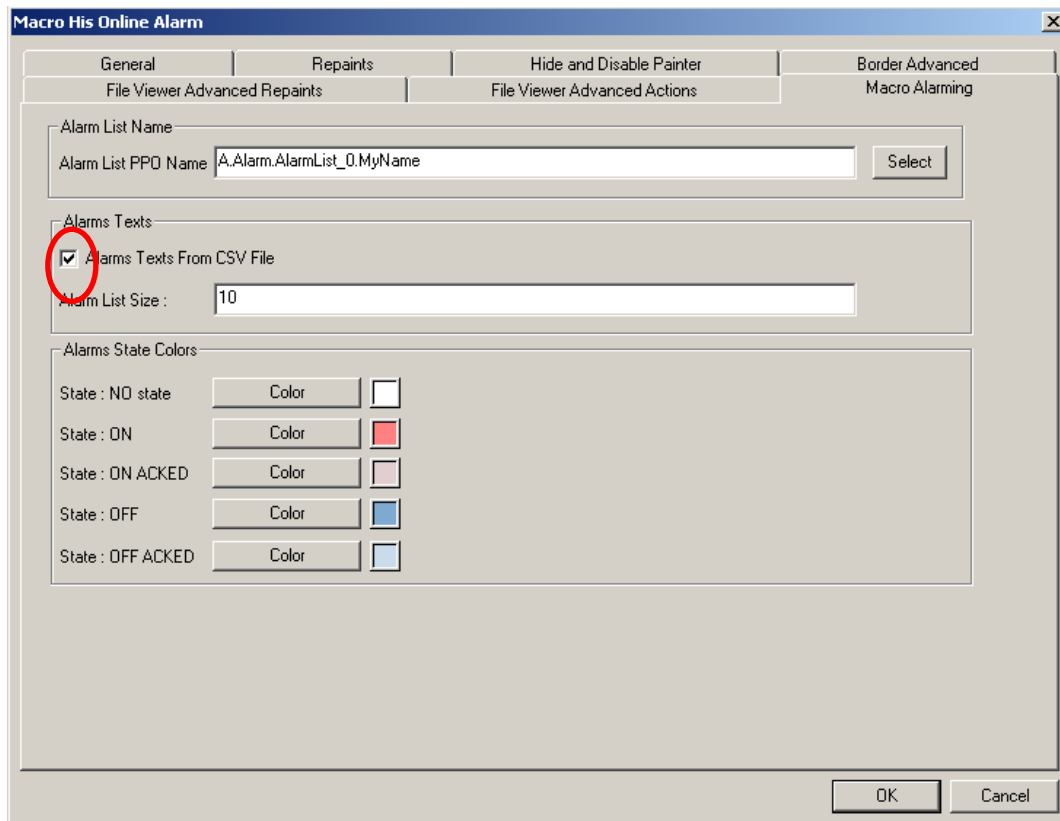
14 Alarm macro online history

The online alarm history shows the chronological sequence of alarms.

Each time an alarm is switched on, a new line is added to the list with that alarm.

- Alarm ID
- Alarm text
- Time and date when alarm was switched on
- Time and date when alarm was switched off
- Alarm status, i.e. whether the alarm has been acknowledged or not

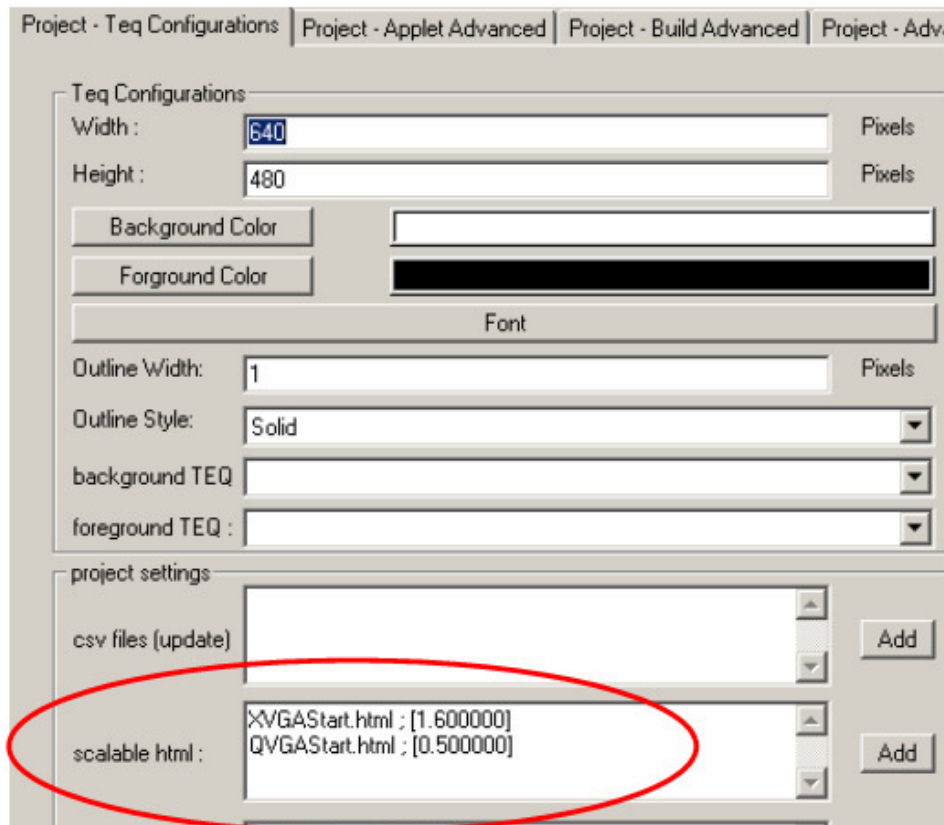
Users can acknowledge alarms (one/selection/all) and clear them (one/selection/all) from the list.



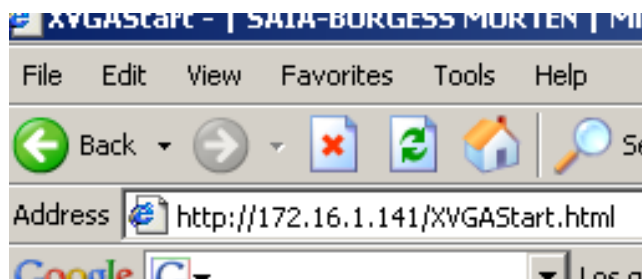
Generate PPOs and declare 'A.Alarm.AlarmList_0.MyName' as a STRING.
 Translate alarm texts in *.csv file.
 (unless done already)

15 Scalability of a web application

Project porting and scaling are now very easy. The entire project is programmed for a particular resolution. The example here shows VGA resolution. Under 'Project configurations', it is now possible to define various entry HTML pages with a scaling factor. For each new resolution, a new HTML page is generated, without duplicating the project pages.



A website programmed in VGA can be displayed in XVGA resolution simply by entering the XVGASStart.html page in the browser. The size of the relevant project pages will be scaled in runtime. Size can be increased or diminished.



But anyway, be careful when diminishing the project in runtime, especially in use with MicroBrowser panels. There are memory-limitations, as for example for the maximum size of a *.gif picture (150 kB) or the number of objects placed on one *.teq view (512 for VGA, 256 for QVGA). The best way's always to upscale the projects view with this scalable feature.

16 Trending

The S-Web-Editor allows PCD data to be displayed in the form of trend graphs. Two types of trend are available: online and offline.

The following macros have been provided:

OfflineSavedTrendComplete (load a saved trend – complete functionality)

OfflineSavedTrendMinimal (load a saved trend – reduced functionality)

OfflineSavedTrendScalable (load a saved trend – scalable display)

OfflineTrendComplete (log a trend in the background and display it later - complete functionality)

OfflineTrendMinimal (log a trend in the background and display it later - reduced functionality)

OfflineTrendScalable (log a trend in the background and display it later – scalable display)

These trends are used in association with logging in DBs (PCD user program). Since DB memory is limited on a PCD, we have introduced trend logging to the flash file system.

For this purpose, the following macro family is used in the WebEditor:

SaiaHDLLogTrendMacro

OnlineTrendComplete (display trend in real time, no registration - complete functionality)

OnlineTrendMinimal (display trend in real time, no registration - reduced functionality)

OnlineTrendScalable (display trend in real time, no registration – scalable display)

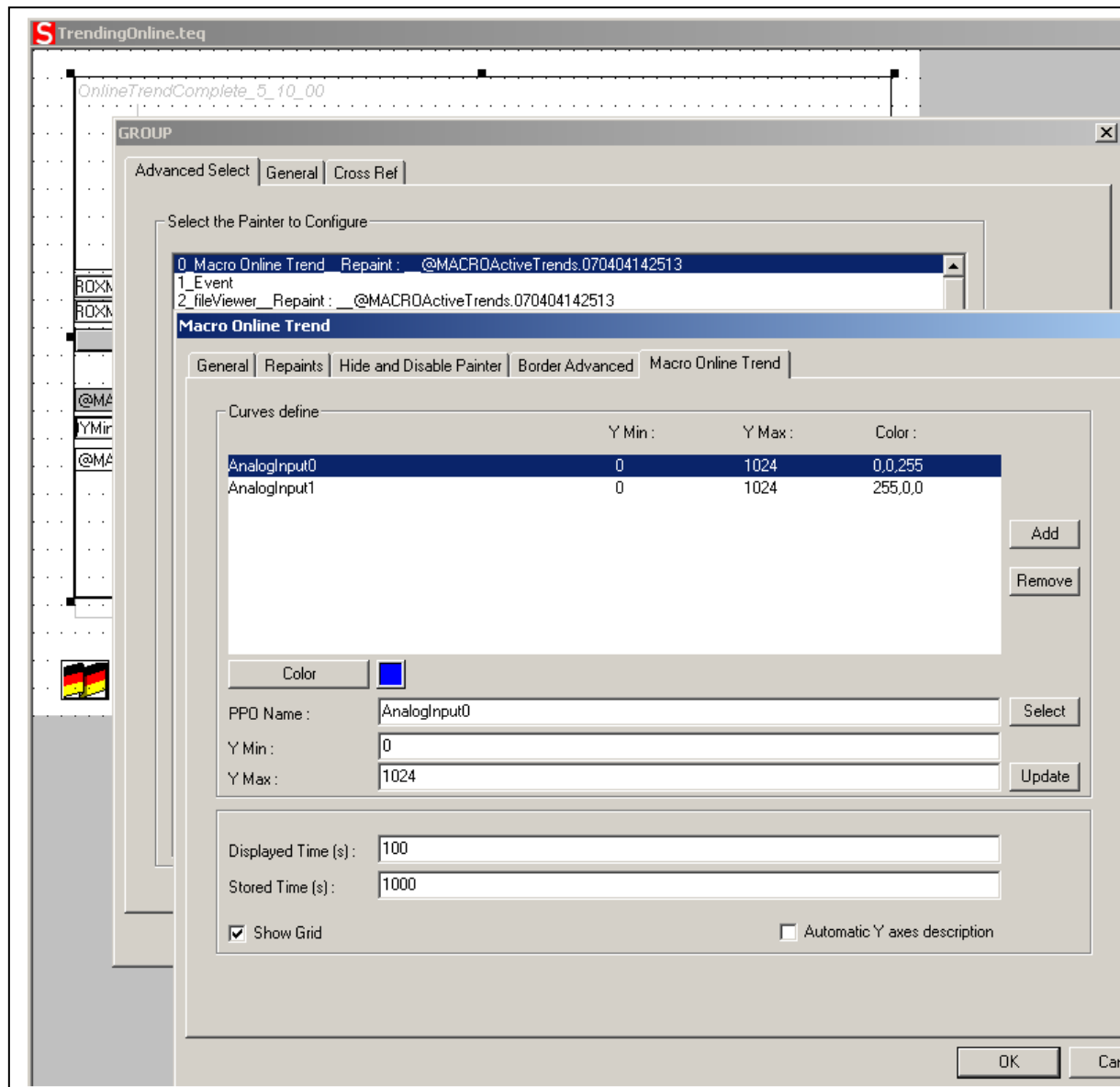
17 Online trend

OnlineTrendComplete_5_10_00.tlb

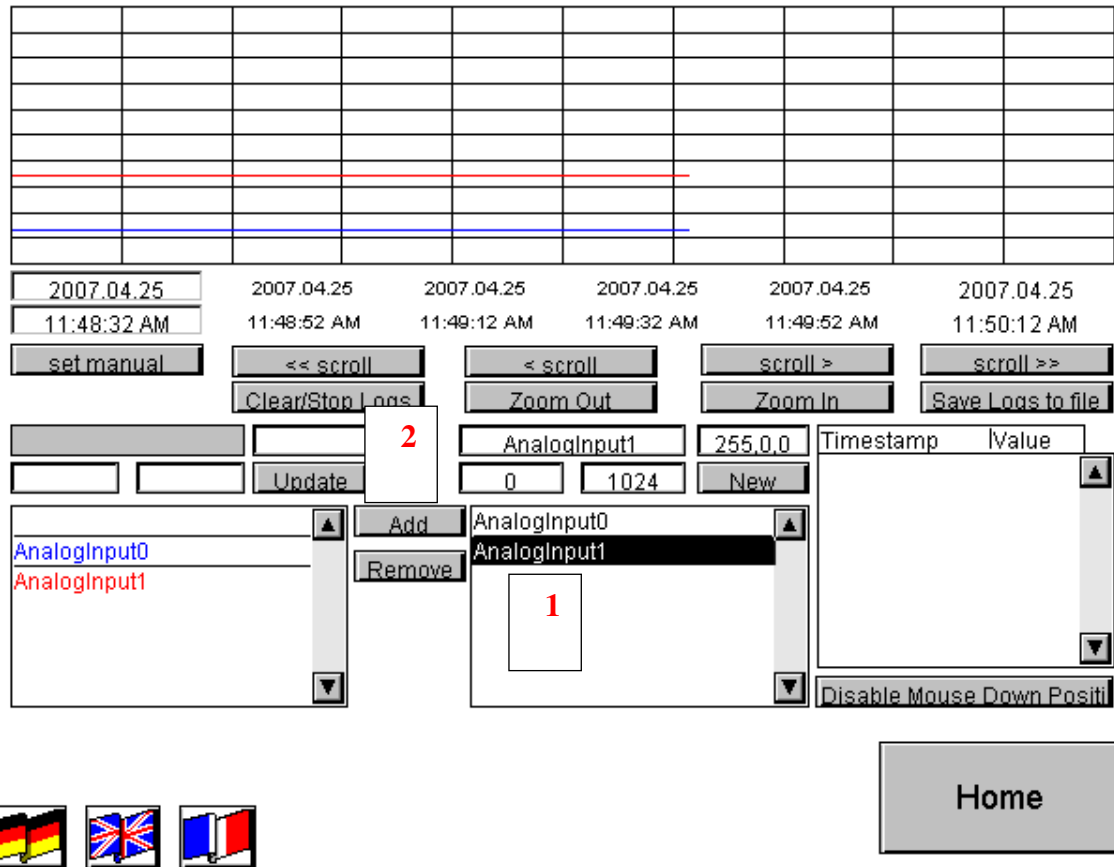
Online trending will only display PCD data in trend curve form if the web browser has an online connection to the PCD.

In this case, the online trend macro will read PCD data points **cyclically** (refresh time for Java applet is, by default, once every second).and display values in the form of trend curves.

No FBoxes needed. Just fill in 'PPO Name', 'Y Min', 'Y Max' and select colour. Click 'Update' each time. 'Displayed Time' is the value for the display time window, 'Stored Time' is the value for the ring buffer. Displayed clock time refers to the clock on the PC with the web browser.



In the browser, just click on 'Add' to include variables for display.
 It is then also possible to scroll or zoom.
 Or grab the measuring point by placing the cursor.



The number of trends that can be displayed in one macro (applicable for both online and offline macros) is approx. 20 trends

Each macro has 2kB of configuration memory.

1 trend = 90 Bytes + trend name (approx. 10Bytes) -> 100 Bytes

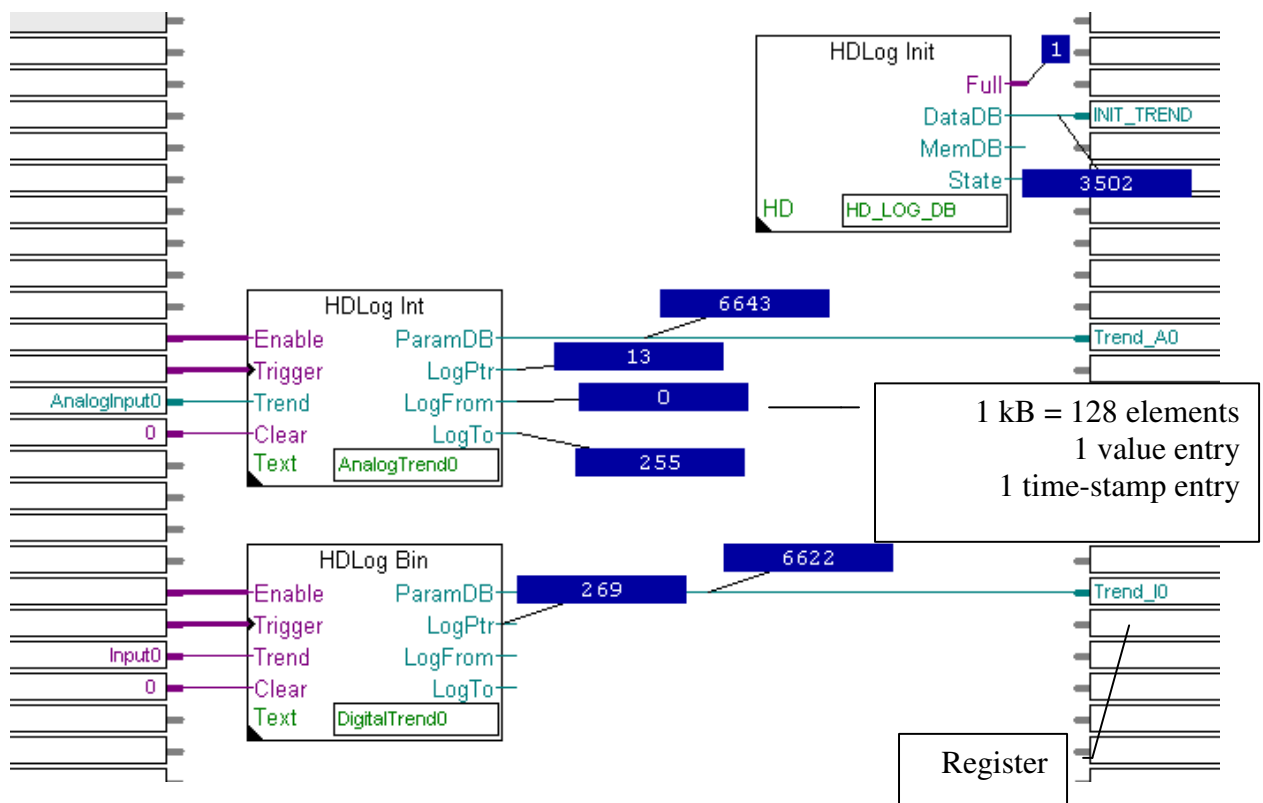
18 Offline trend in DBs

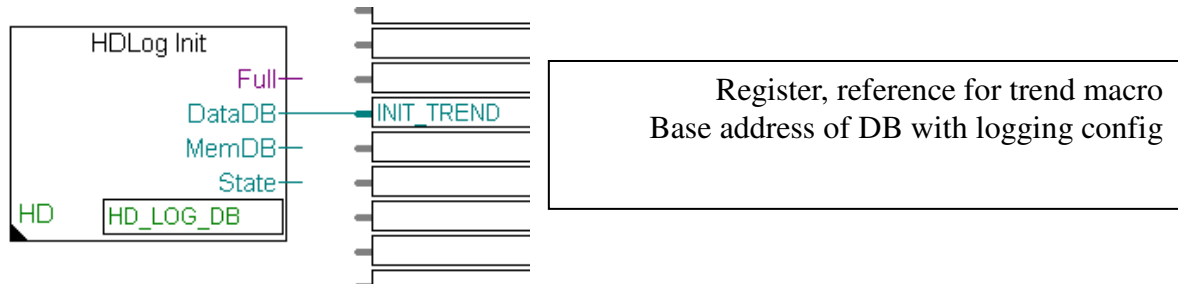
OfflineTrendComplete_5_10_00.tlb

If offline trend macros are used, PCD data points must be stored using special FBoxes in the PCD's user memory. Data points are stored DBs belonging to the user program.

FBoxes

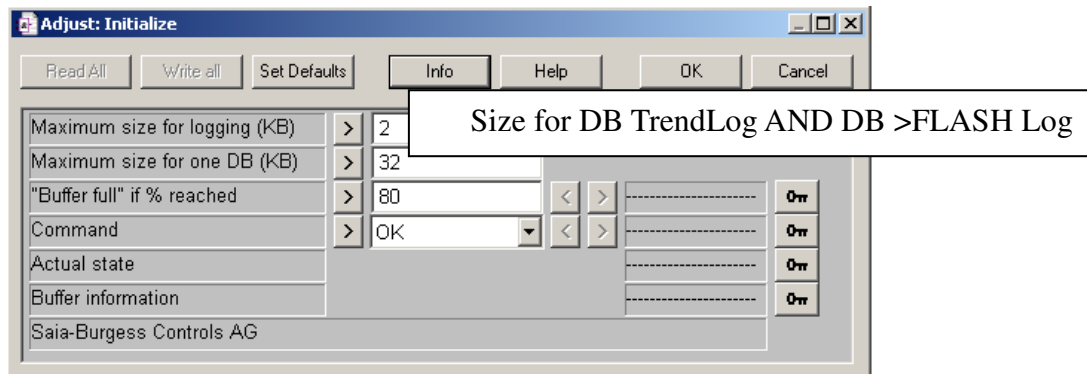
FBoxes are to be found under 'User'. They allow binary and integer values to be saved..





HD_LOG_DB DB containing all configuration data. Local variable, DB **without** definition. **Un-check 'Definition'**.

DataDB Base address of DB containing the logging configuration. This value is stored in a register and used in the offline trend macro 'Base PPO Name'. Global variable.
Contains pointers to all other trend DBs.



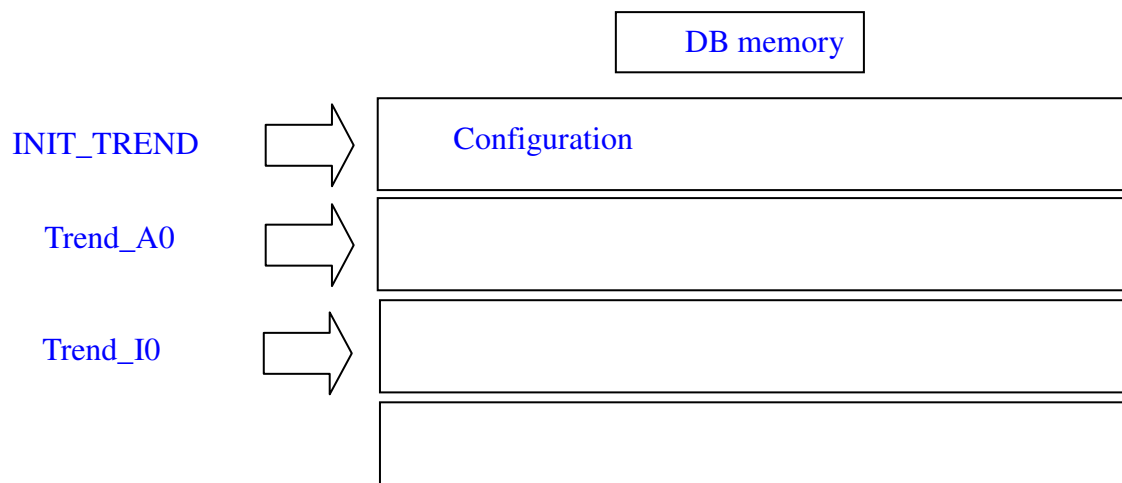
'Maximum size for logging' = Sum of the memory of all logging FBoxes.

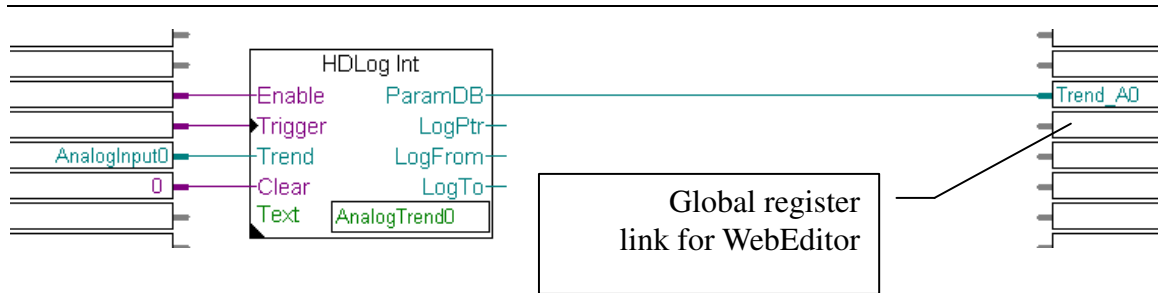
Note that overall reserved logging memory should be shared among a maximum of 20 DBs, otherwise there will be compilation errors. .

Logs are entered in DBs.

1 measurement = 8 Bytes (4 Bytes for time stamp + 4 Bytes for value) -> 128 logs in 1 kByte

This applies for both integer and binary logs.





Enable should be 1 to enable logging
 Trigger to force a log
 Trend PCD media to be logged. Global register of PCD.
 Clear clears the log file

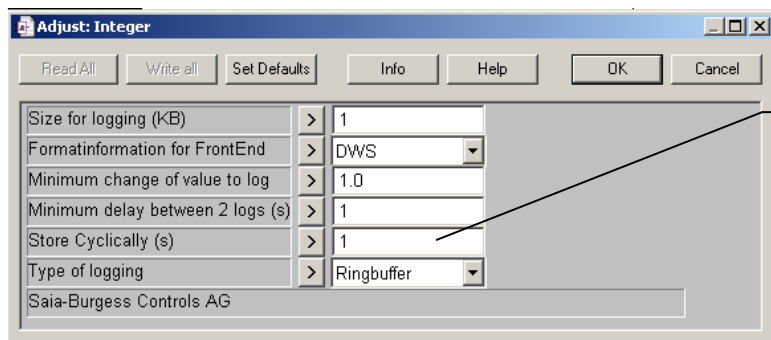
ParamDB Pointer to the configuration DB address of a trend. Global register also used in the trend macro.

AnalogInput0 analogue input to be logged
 AnalogTrend0 local text

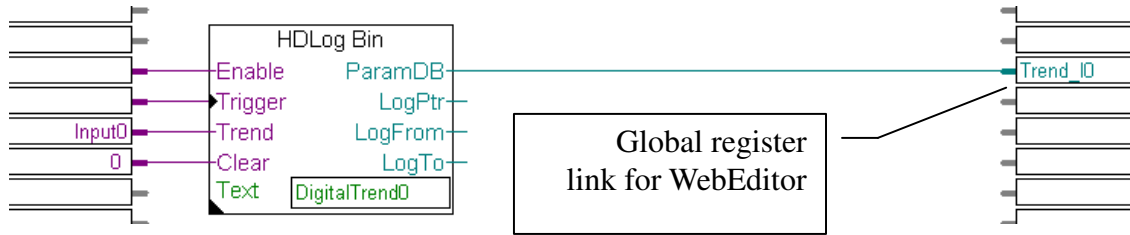
Values in the FBox can be saved cyclically, or whenever the value changes.

128 logging entries will require 1 KB

1 measurement = 8 Bytes (4 Bytes for time stamp + 4 Bytes for value)



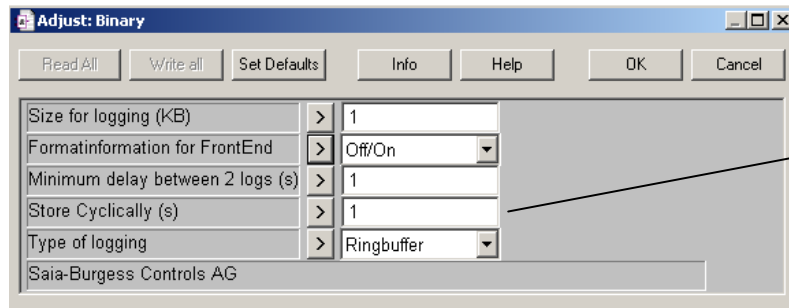
'0' if there is NO
cyclical log



Enable should be 1 to enable logging
 Trigger to force a log
 Trend PCD media to be logged. Global register of PCD.
 Clear clears the log file

ParamDB Pointer to the log DB address of a trend. Global register also used in the trend macro.

Input0 digital variable to be logged (input / flag)
 DigitalTrend0 local text



128 logging entries will require 1 KB
 1 measurement = 8 Bytes (4 Bytes for time stamp + 4 Bytes for value)

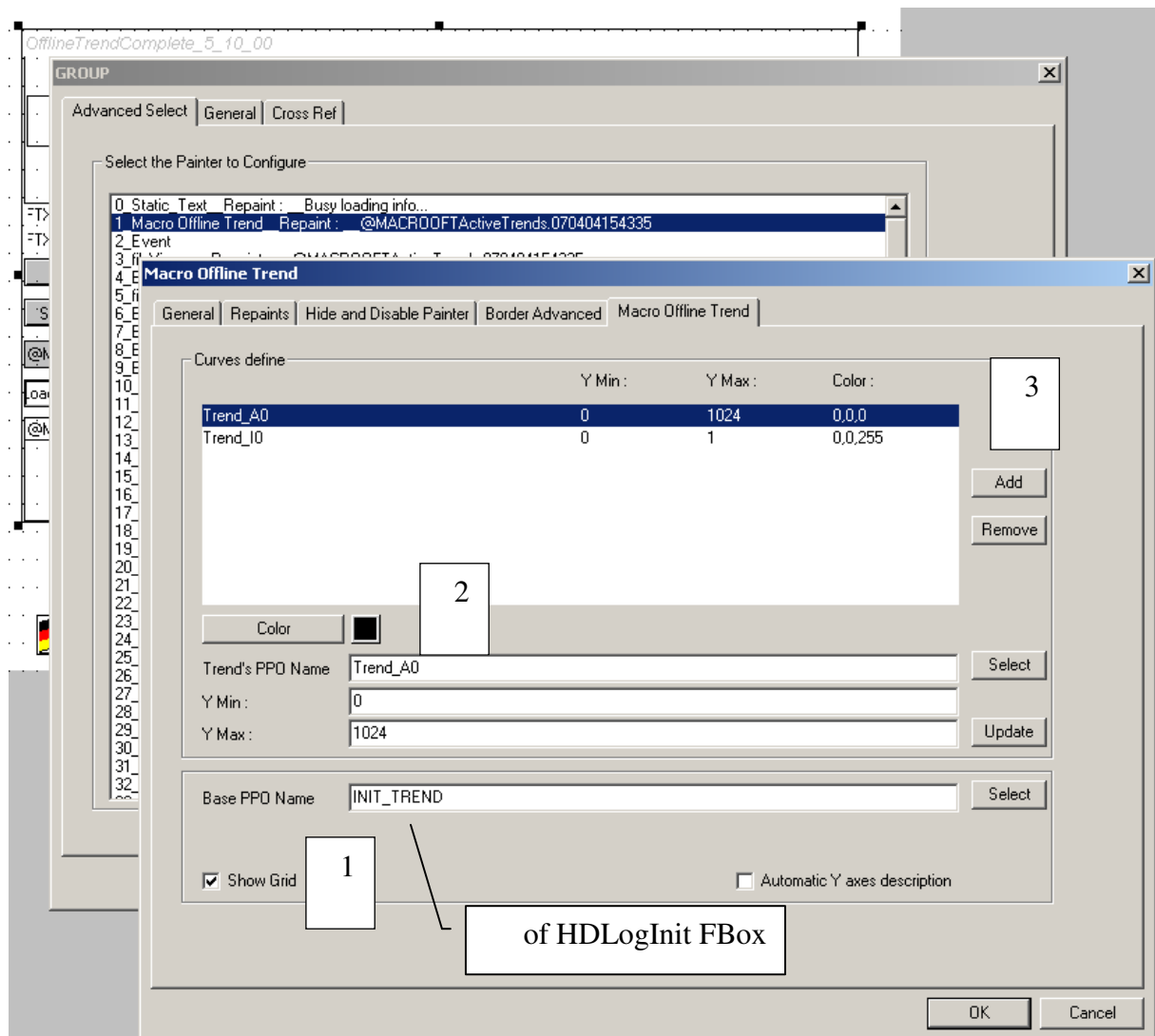
How many HDLogInit/Bin FBoxes for offline trends can be used in the Fupla?

Unlimited: as many as the PCD's memory size will allow.

HDLogInit can, in theory, take up the PCD's entire DB memory. However, for overall memory association, no more than 20 DBs should be used. This means that the maximum size for logging (kB) / maximum size for one DB <= 20.

**Approx. 20 trends
per macro**

OfflineTrendComplete_5_10_00.tlb



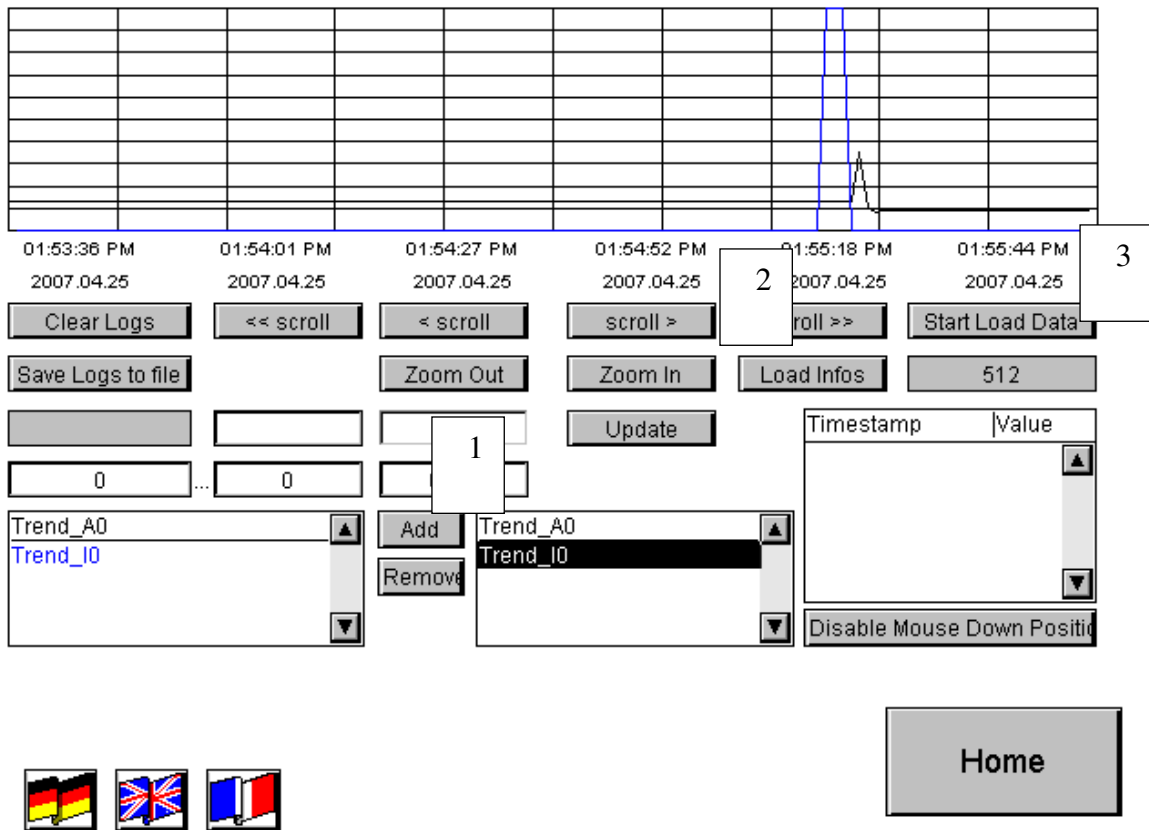
Select 'Trend's PPO Name', define a colour and limits, then add.

'Base PPO Name' is the name of the initialization DB for the Init FBox.

Total trends displayable with one online or offline macro: approx. 20

There are 2kB of configuration memory for one macro.

1 trend = 90 Bytes + trend name (approx. 10Bytes) -> 100 Bytes



Trend curves to be logged must be added with the 'Add' button.

'Load Info' indicates the total number of pending values. 'Start Load Data' loads the curve.

'Save Logs to file' will save logs to a file on the local PC. Open in Excel. A *.csv file is best. Display time by right-clicking on 'Format Cells' in Customs. Seconds are added with ':ss'.

19 Offline trend into flash file system

RemoteOffSavedTrdComplete_5_14_06.esm

FBoxes

Before the 'HD Log File' FBox can be used, an initialization FBox for the FLASH file system must be put in place. It MUST be named as follows: 'M1_FLASH', 'M2_FLASH', 'SL0FLASH', etc.

The diagram illustrates the configuration of the M2_FLASH FBox. It shows a block labeled 'M2_FLASH' with inputs 'Enable' (set to 0), 'Format' (set to 0), 'Comp', and 'AutoCmp'. The block has outputs 'Memory', 'Busy', 'Used', 'Free', 'Relsd', 'Status', and 'Error'. An arrow points from this block to the 'HDLog File' FBox, which has inputs 'LogEnable', 'Conference.Anal...', 'AnalogInput1', 'AnalogInput2', 'AnalogInput3', 'Store', 'LogWriteFile', and 'DelFile'. The 'HDLog File' block has outputs 'En', 'Val0', 'Val1', 'Val2', 'Val3', 'Store', 'WrFile', 'DelFile', 'Busy', 'WrOK', 'Error', 'Buffer', and 'DelRdy'. A 'Path' input is also shown.

The 'Adjust: Memory Management' dialog is shown with the following settings:

- PCD Memory: M2 On board
- Block size: Default
- Auto compression: By system
- Auto compress, user level...: 50%
- Total memory size [kb]:
- Used memory [kb]:
- Free memory [kb]:
- Released memory [kb]:
- Memory status: Update
- Update memory+status: Select
- Format memory: Execute
- Format memory: Compress
- Compress memory: Compress
- Common file properties:
 - Group ID: Web
 - Group Access: All
 - Index length: No index
 - File extension: CSV
 - Value separator...: Semicolon

The 'General settings' dialog is shown with the following settings:

- Records in buffer...: 120
- Create file: per day
- Decimal sign: (dot)
- Write from buffer into file: automatically
- Mailbox: OK
- Logging:
 - Record: cyclically
 - cyclically interval: 1.000
 - COV hysteresis: 5.0
- Group 1:
 - Format: DEC
 - Unit: °C
 - Y-axis minimum: 0.0
 - Y-axis maximum: 1023.0
- Group 2:
 - Format: DEC.1
 - Unit: °C
 - Y-axis minimum: 0.0
 - Y-axis maximum: 100.0
- Group 3:
 - Format: DEC.1
 - Unit: °C
 - Y-axis minimum: 0.0
 - Y-axis maximum: 1000.0
- Group 4:
 - Format: DEC.1
 - Unit: %
 - Y-axis minimum: 0.0
 - Y-axis maximum: 100.0

The 'Allocation' dialog is shown with the following settings:

- Value 0: group 1
- Value 1: group 2
- Value 2: group 2
- Value 3: group 1
- Value 4: group 2
- Value 5: group 2
- Value 6: group 3
- Value 7: group 3
- Value 8: group 4
- Value 9: group 4

The 'Colours' dialog is shown with the following settings:

- Value 0: red
- Value 1: grey
- Value 2: green
- Value 3: blue
- Value 4: yellow
- Value 5: light yellow
- Value 6: red
- Value 7: light red
- Value 8: green
- Value 9: light green

The 'HD Log File' FBox can log up to 10 variables. These variables are then logged to the PCD's RAM DBs. The 'HD Log Init' FBox reserves the necessary DB size. If this memory is 50% full, logs will automatically be copied to a *.csv file on the flash file system. This may take a few seconds. To avoid losing data during copying time, the FBox will log provisionally to the remaining half of RAM DB memory.

Records in buffer .. In the FBox, the number of log entries can be defined at which DB content should be saved to a file on the flash card.

Create File It is also possible to define whenever a new file should be generated in the flash.

Write from buffer into file One can also choose whether DB content should be copied to flash either automatically or manually, using the 'WrFile' trigger

Record Logging is possible either cyclically, or when a variable changes, or in response to a trigger signal. There is AND linkage between 'cyclically interval' and 'COV hysteresis'

Group Up to 4 formatting groups can be defined. The trend format, unit and limits of the Y axis are defined here. Each trend can be assigned to one of these groups.

All variables are logged with the same time stamp.

En 'Enable' is needed for any logging whatsoever. During course, set to a default of '0'.

Val0..Val9 Variables to be logged

Store Triggers when logging is to take place. Dynamized. Used if neither cyclically timed logging is required, nor logging after a change of value.

LogWriteFile Will be triggered if variables logged in DBs are to be saved to a file on the file system.

DelFile Delete the file according FAQ #101'276

Path GLOBAL Text containing a declaration of the directory path to which log files are to be saved. i.e.: "/Webpages" or „/Webpages/data“ if the directory "data" was before created with the file system FBoxes.

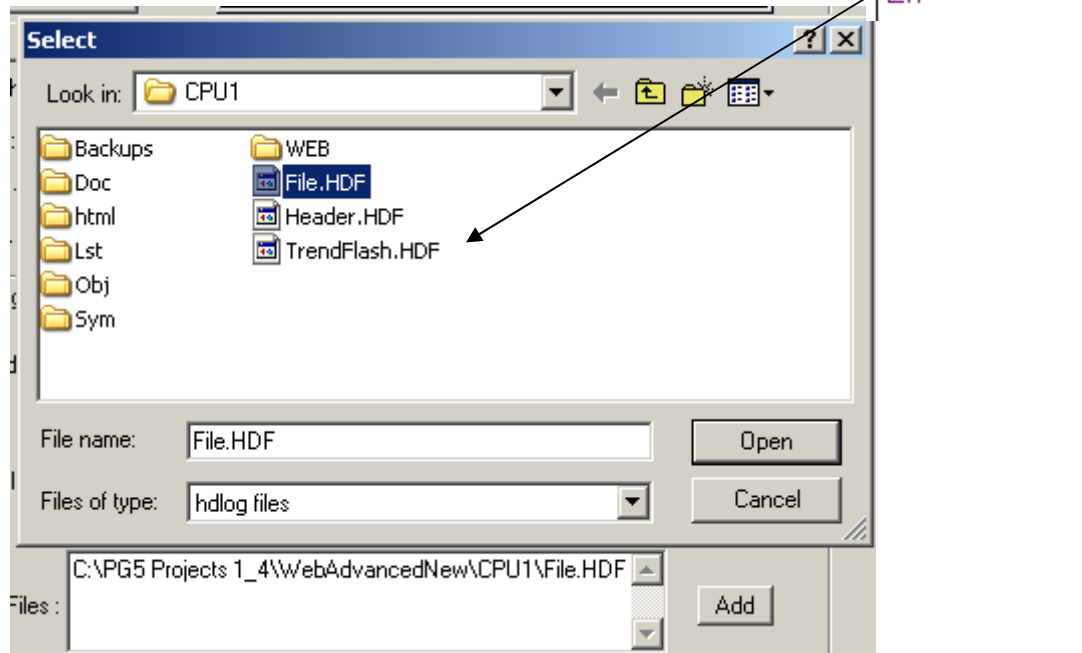
Build the project and show the *.hdf-file. Download the projekt, log values and show the logged *.csv-file.

WebEditor

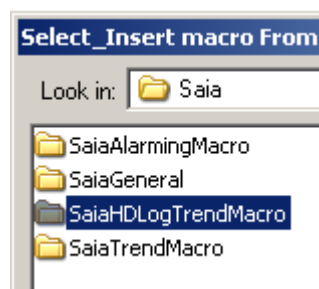
Under



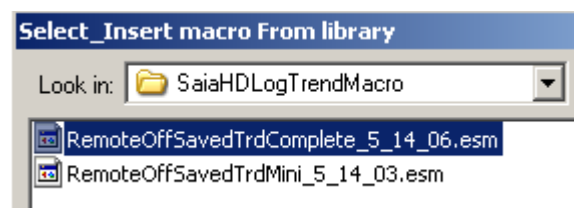
add the reference 'File.HDF' at SaiaHDlog Files.



Insert the HDLog offline trend macro in the WebEditor



As with the traditional offline trend, there is a slimmed-down minimal version and a complete version.



RemoteOffSavedTrdComplete_5_14_06

Curve Name	YMin	YMax	Color

Displayed Time [s]

Stored Time [s]

Number of Curves

☒ Show Grid

☒ Automatic Y Axes description

'Displayed Time' defines the display time window (3600 seconds for 1 hour) and 'Stored Time' shows the total time buffer in seconds.

'Add' will display a selection list of available, defined trend variables.

Select the trends and click on to insert in macro and save.



Available Curves

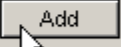

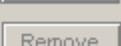
Please define the HDLog files in the menu Project->Project configurations...

TrendFlash	AnalogInput0	0.0	1023.0
TrendFlash	AnalogInput1	0.0	100.0
TrendFlash	AnalogInput2	0.0	100.0
TrendFlash	AnalogInput3	0.0	1023.0

Insert the other trends here with .

RemoteOffSavedTrdComplete_5_14_06

	Curve Name	YMin	YMax	Color
TrendFlash	AnalogInput0	0.0	1023.0	
TrendFlash	AnalogInput1	0.0	100.0	

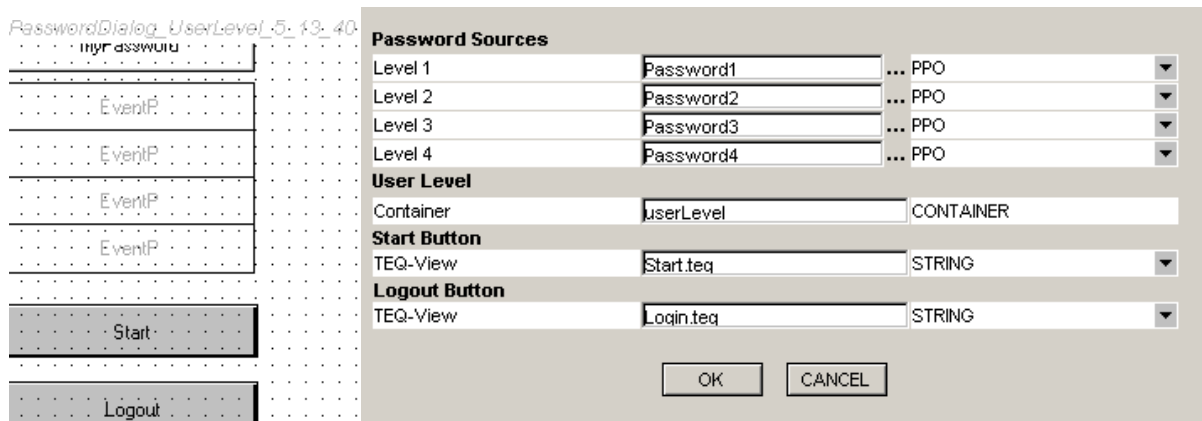
  

20 Login

The login macro comes by default with 4 login levels. There are two login mechanisms: one using just a password, and the other using a password and a user name.

Both macros will be found under 'MacroLib' -> 'Saia' -> 'SaiaGeneral' -> 'PasswordLogin'

PasswordDialog_UserLevel_5_13_40.esm



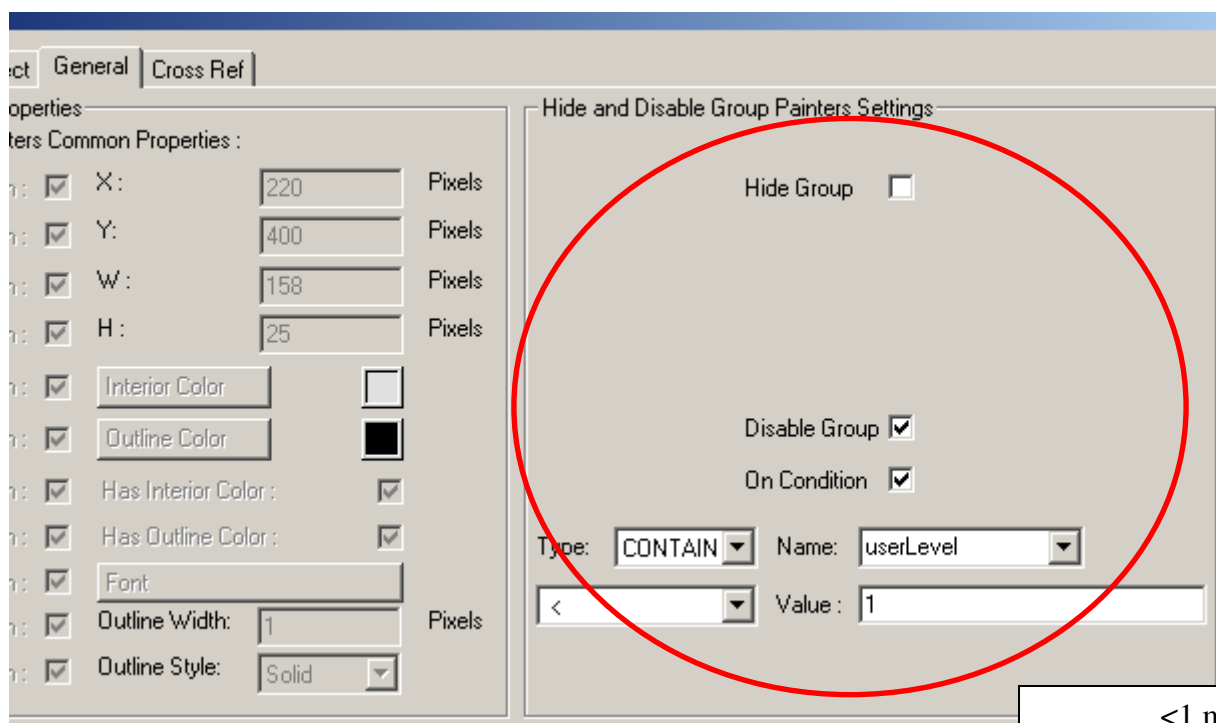
For example, a user can only change to the 'Start.teq' page when a correct password has been entered. Up to 4 passwords can be defined in the macro. These passwords are PCD texts, containers, HTML tags or strings: 'Password1', 'Password2', 'Password3', 'Password4' for the four login levels. This PPO must be defined in the WebEditor's PPO list as a 'STRING'.

The macro has 4 invisible event objects that cyclically compare the passwords stored in the PCD with the password entered in the entry field.

The entry field does not display the characters keyed in during password entry, instead it shows '*'. Text entered is stored in the container variable 'myPassword'. If one of the 4 passwords stored in the PCD matches the text entered, a value between 1 and 4 (1 for login level 1, 2 for login level 2, ...) is written to the container variable 'userLevel'. This makes it possible to execute conditional page calls or display of elements. In the absence of a login, the container has a 'userLevel' value of '0'.

Pressing the logout button executes a jump to the 'login.teq' page.

Block the activity of a button (e.g. 'Delete Alarm List' in Alarming Online) when not logged in.



<1 means:
not logged in

In addition to the simple login, there is a similar macro with user name and password:
PasswordDialog_UserLevel_Auto_5_13_40.esm

Auto Jump view:	where to jump to if login OK
TEQ-View:	where to jump to if logout
userLevel:	CONTAINER holding the password level
Username1..4:	coded user name in PPO, Container, String, HTML tag
Password1..4:	coded password in PPO, Container, String, HTML tag

User Level 1		
Username	User1	... PPO
Password	Password1	... PPO
Auto Jump View	Entry.teq	STRING
User Level 2		
Username	User2	... PPO
Password	Password2	... PPO
Auto Jump View	Entry.teq	STRING
User Level 3		
Username	User3	... PPO
Password	Password3	... PPO
Auto Jump View	Entry.teq	STRING
User Level 4		
Username	User4	... PPO
Password	Password4	... PPO
Auto Jump View	Entry.teq	STRING
User Level Container		
Container	userLevel	CONTAINER
Logout Button		
TEQ-View	Login.teq	STRING

Macros related to the login: EventP_Logout_onTimeout_5_13_40.esm

User Level	
Container	UserLevel
TEQ-View	
Name	Login.teq
Max. User Inactivity	
Type	STRING
Name	20
Event	
Name	On Timeout

After 20 seconds of inactivity, a jump will be made to 'Login.teq' and the 'userLevel' variable will be reset to '0'. The timeout can be programmed either as a STRING, in a CONTAINER, in a PPO or as an HTML tag.

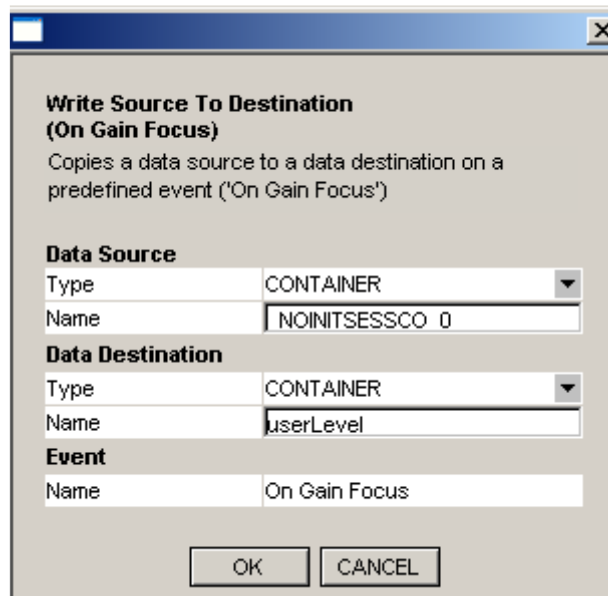
21 URL jump with transfer of password level

When jumping from one PCD to another, it is often useful to carry over a previously logged-in password level. This happens if, when leaving the local PCD, the password level (container variable) is copied to the clipboard on the browser used. On arrival at the remote PCD, the password level can be written again from the browser clipboard back to the container variable. Session containers are used for this purpose.

After logging in and arriving at the start page, I copy the 'userLevel' container onto the session container '_NOINITSESSCO_0'. In this way, the current user level is saved to the browser's clipboard.

Write Source To Destination (On Gain Focus)	
Copies a data source to a data destination on a predefined event ('On Gain Focus')	
Data Source	
Type	CONTAINER
Name	UserLevel
Data Destination	
Type	CONTAINER
Name	_NOINITSESSCO_0
Event	
Name	On Gain Focus
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>	

The URL jump is now made and, after arriving at the start page of the remote PCD, the user level is fetched back from the clipboard.



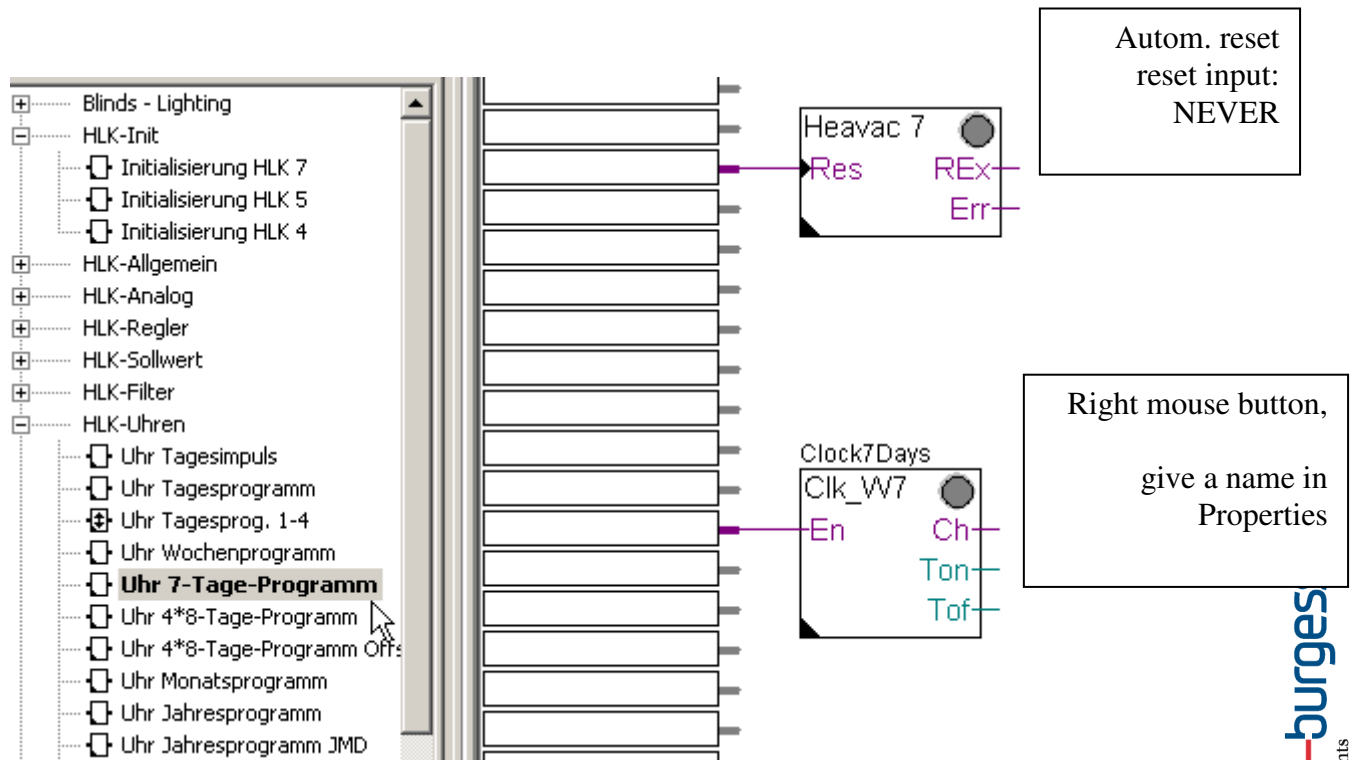
The dialog box is titled "Write Source To Destination (On Gain Focus)". It contains the following fields:

- Data Source:**
 - Type: CONTAINER (dropdown menu)
 - Name: NOINITSESSCO_0 (text field)
- Data Destination:**
 - Type: CONTAINER (dropdown menu)
 - Name: userLevel (text field)
- Event:**
 - Name: On Gain Focus (text field)

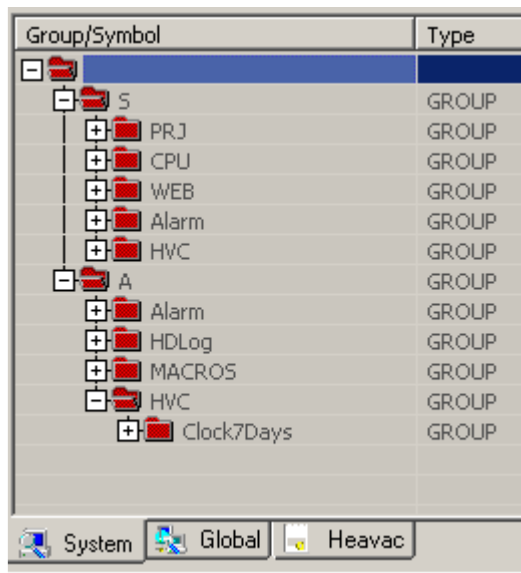
At the bottom, there are two buttons: OK and CANCEL.

22 HEAVAC macros (SWeb templates)

For each HEAVAC FBox there is now a macro in the WebEditor.

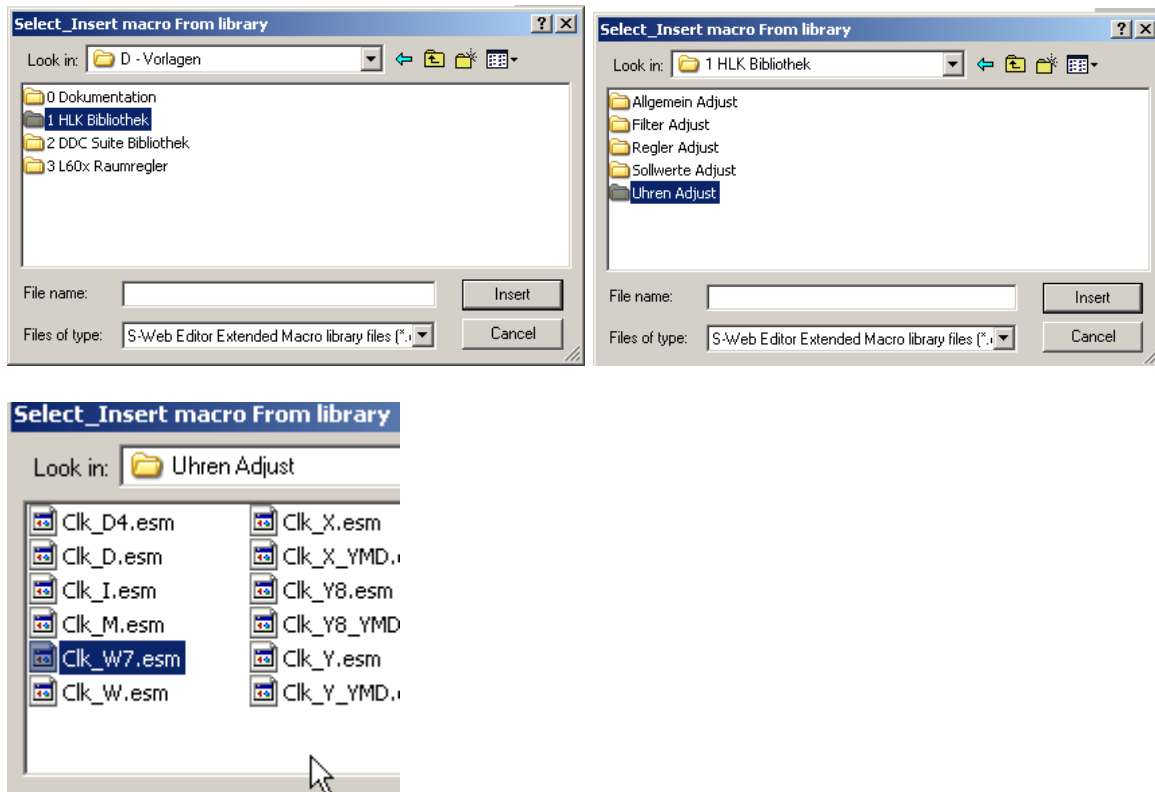


Adapt properties of the FBox as a reference -> 'Clock7Days'. Build in FUPLA. The relevant variables will then be reserved among system variables in a folder.

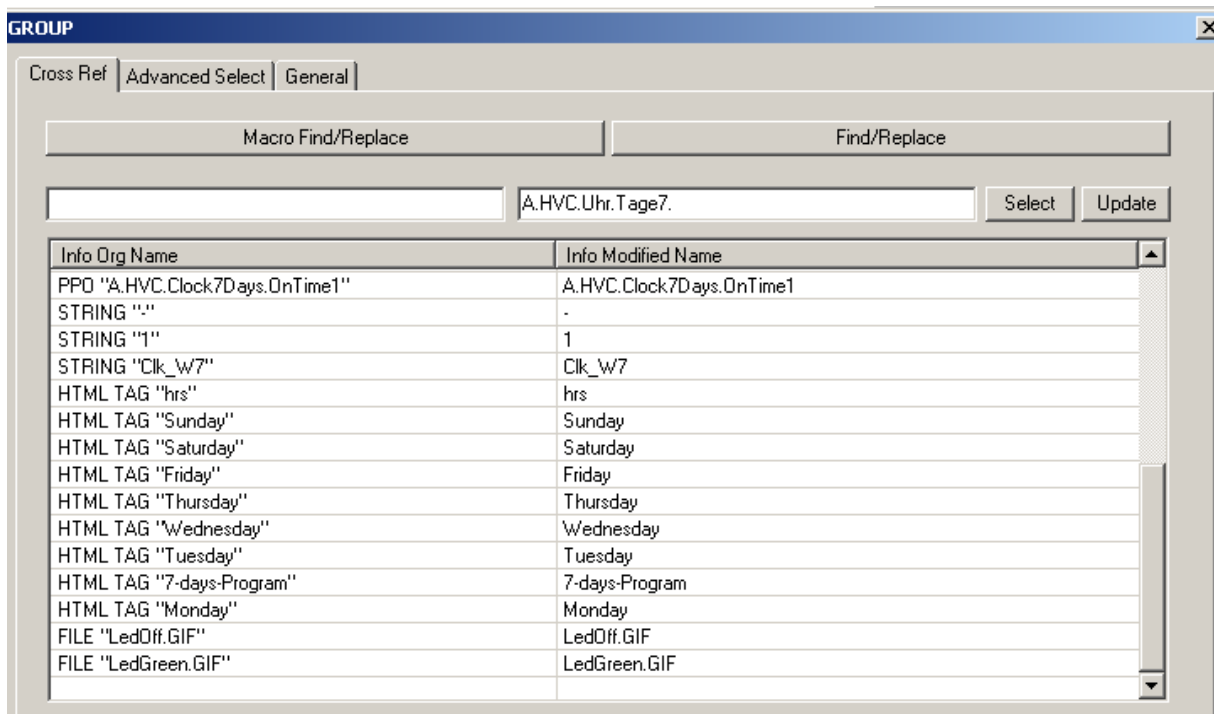


Group in system variables is generated automatically with all associated variables

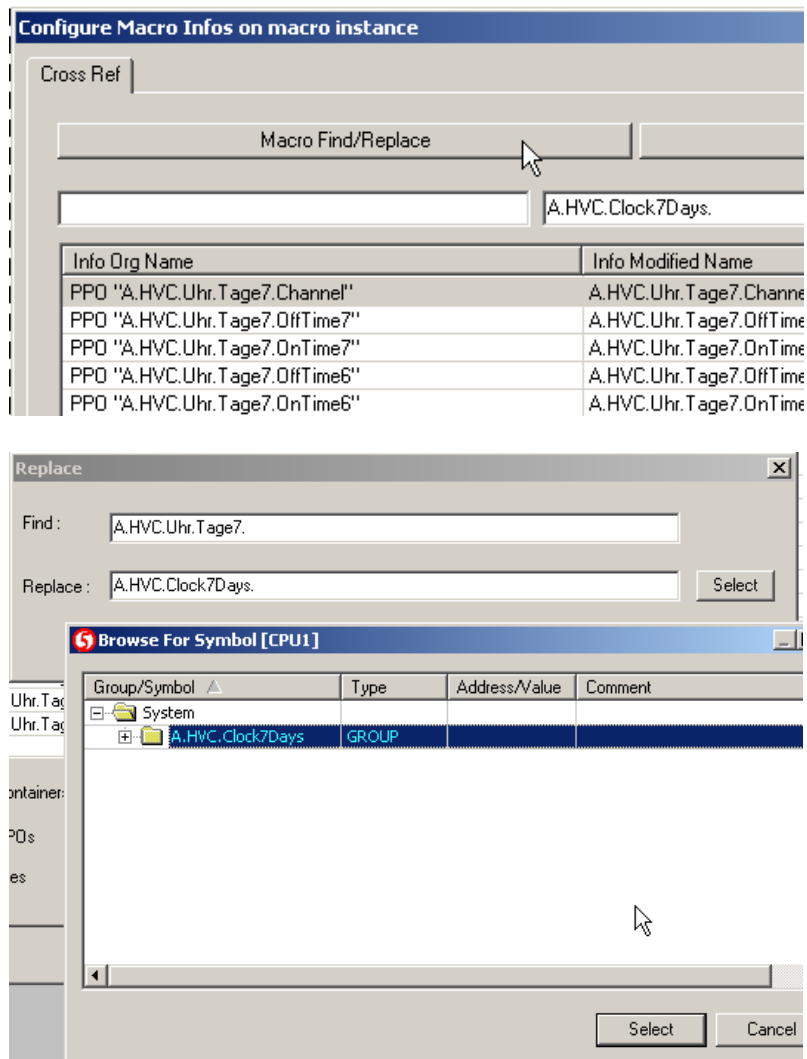
Import macro ... SWeb Templates ...



All elements (PPOs, strings, HTML tags, GIFs) are contained in the macro.

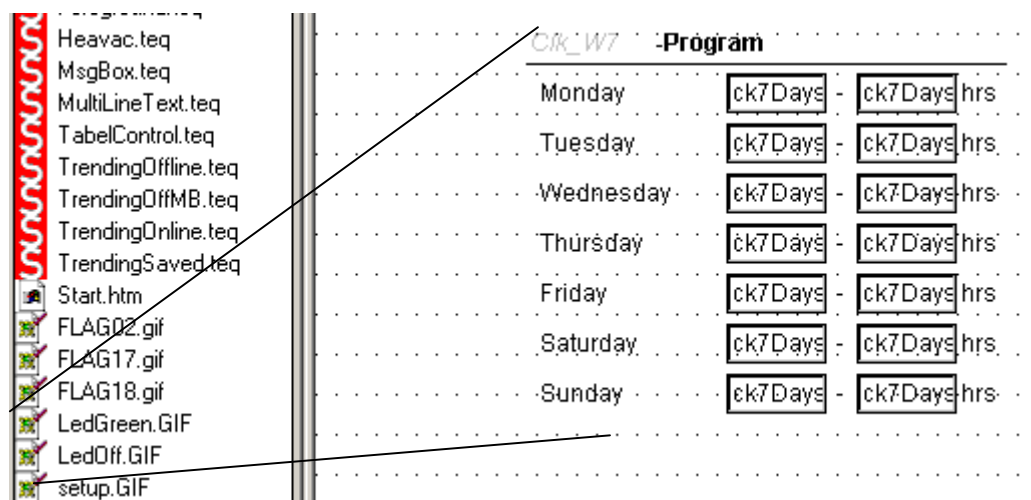


'Macro Find/Replace' is used to adapt the default text of PPO variables to the FBox we have referenced.



All possible groups
are displayed
(filter)

GIFs also imported at same time.



Generate CSV and translate added HTML tags

23 Drop-down list

A drop-down list is an object allowing a selection of elements from an existing list.

24 A PPO variable, in which the value read or written (as an integer) will depend on a selected element. Selecting 'Action0' will produce a value of 0 in the PPO 'DropDownReg'; 'Action1' will produce a value of 1, etc.

25 A text list that describes the text of an element (Action0, Action1, ..) and which can be modified by the user as required

The text list can be defined as a string variable. Separate lines of text with '\n'

The text list can be defined as an HTML tag.

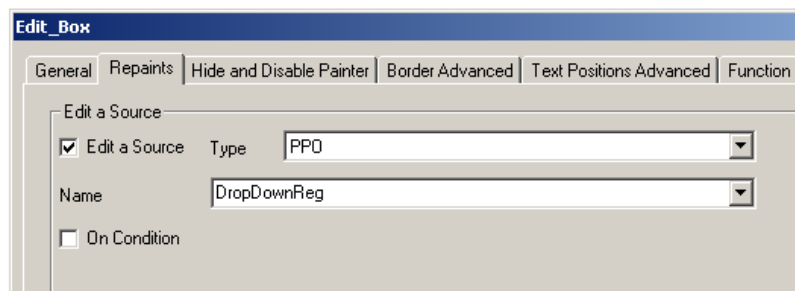
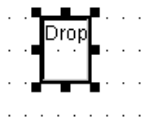
With an HTML tag variable, the text can be integrated into a language-dependent *.csv file.

In the *.csv file, one will find:

Text1\nText2\nText3\nText4;Line1\nLine2\nLine3\nLine4

Here, 'Text1', 'Text2', ... are identifiers. To the right of the ';' will be found a selection of line elements. They are separated by the new-line character '\n'.

The drop-down register can be displayed in editable form using an Edit box. This will also result in modification of the drop-down menu selection.



It is possible to use the drop-down menu selection and its associated description of a PPO variable for jumps to other web pages.

For example:

Room A	0
Room B	1
Room C	2
Room D	3
Home	4

26 Forced page change - jump event

EventP_ViewJump_isEqual_5_13_40.esm

A page change can be generated in the S-Web-Editor by either the user or the PCD. For example, the PCD may use a jump event in case of error, in order to show the user the 'correct' page directly. Alternatively, an input can control an automatic jump to a specific page.

The changeover condition will only be executed if the jump event macro is located on the page being displayed.

This macro will often be included on the background page, which is part of all other pages.

TEQ-View

Type: STRING
Name: Entry.teq

X Data Source

Type: PPO
Name: Input7 ...

Y Data Source

Type: STRING
Name: 1

Condition

Name: Is Equal

OK CANCEL

The following condition applies here: Jump to 'Entry.teq' page when input 7 == 1
Or as in previous example: Jump to 'Entry.teq' page when PPO == 4.

Caution: When exiting the page with the 'onLostFocus' macro, remember to write value '0' to the PPO, otherwise you will program an endless loop.

27 Automatic URL jump for inactivity

URL jumps let users navigate from one PCD to another. They can also be used to ensure that, when there is inactivity on a remote PCD, an automatic jump will be made to the original PCD. This function is provided by the macro URL jump for inactivity 'EventP_URLJump_onTimeout_5_13_40.esm'.

URL Jump Event-Painter (On Timeout)
Performs a URL Jump after a specific time of user inactivity in seconds (uses the Container MB_InactivityCounter)

URL Address
Type: STRING
Name: http://172.16.1.141

X (Max. Inactivity)
Type: STRING
Name: 10

Condition
Description: Inactivity Counter > X

OK CANCEL

The address of the original PCD can, as illustrated, be directly defined as a STRING. Alternatively however, it may also be defined as a CONTAINER, HTML tag or PPO variable of the PCD.

URL Jump Event-Painter (On Timeout)
Performs a URL Jump after a specific time of user inactivity in seconds (uses the Container MB_InactivityCounter)

URL Address
Type: PPO
Name: HomeAddress

X (Max. Inactivity)
Type: STRING
Name: 10

Condition
Description: Inactivity Counter > X

OK CANCEL

i.e. 10 seconds of inactivity

HomeAddress

Text

Edit Text - HomeAddress

☒ Definition

Text Size

☒ Default

☐ Fixed

Text:

"http://172.16.1.141"

28 Blinker

The blinker macro does not even have to be put in place. Just use the special CONTAINER variable @BLINKCO, which is switched at a refresh frequency between 0 and 1. Quasi-subscription to the container variable @BLINKCO

Button

Hide and Disable Painter	Border Advanced	Text Positions Advanced	Function
General	Repaints	Actions Set Variables	Actions Toggle Increment Variables

☒ Edit a Source Type:

Name:

☒ On Condition

Type: Name: == Value:

☒ Edit a Source2 Type:

Name:

☒ On Condition

Type: Name: == Value:

Button

General	Repaints	Actions Set Variables	Actions Toggle Increment Variables	Actions Jur
Hide and Disable Painter	Border Advanced	Text Positions Advanced	Function Keys	

☒ 2 Colors on conditions

☒ Interior Color1 (ColorVar == 0)

☒ Interior Color2 (ColorVar == 1)

Color Conditions

ColorVar Type:

ColorVar Name:

29 'Table Control' macro

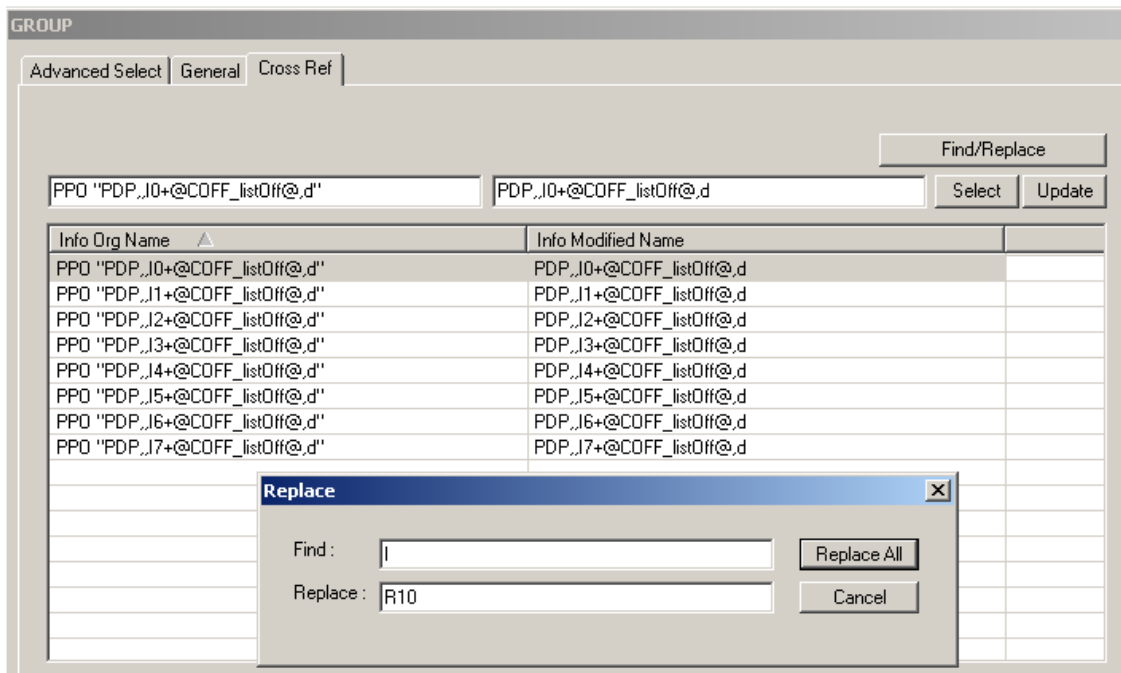
Each element of the table corresponds to a PCD variable. PCD variables can be read and written to.

This macro supports page-by-page browsing through the list. It does not support line-by-line scrolling down the table, unless one programs a jump from '1' unit in the table.

Addressing is only possible with static addresses, i.e. not the PG5's global symbols.

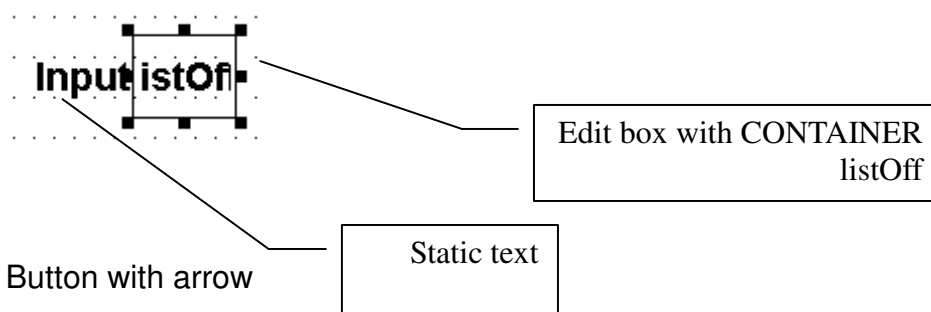
Take the macro, open it up, add the required number of entries to it, adjust increment of container variable 'listOff'. '>' '<' arrow. Remember limits. Otherwise, adjust them via the cross reference.


Execute grouping. Then use Find/Replace in the cross reference to enter/adjust the required medium and base addresses.



In course: Find R100 -> Replace I

Place a small Edit box displaying the CONTAINER 'listOff' as the base address of this list. It serves as a reference to indicate one's location on the list.



Increment +8 for 

Button

Hide and Disable Painter	Border Advanced	Text Positions Advanced	Function Keys
General	Repaints	Actions Set Variables	Actions Toggle Increment Variables

Toggle Button

☐ Toggle

Increment

☒ Increment a Variable Type:

Name:

Value:

Adjust limits

Button

General	Repaints	Actions Set Variables	Actions Toggle Increment Variables	Actions Jump
Hide and Disable Painter	Border Advanced	Text Positions Advanced	Function Keys	

Hide Painter

☒ Hide Painter

☒ On Condition


Type: Name:

Disable Painter's Actions

☒ Disable Actions

☒ On Condition

Type: Name:

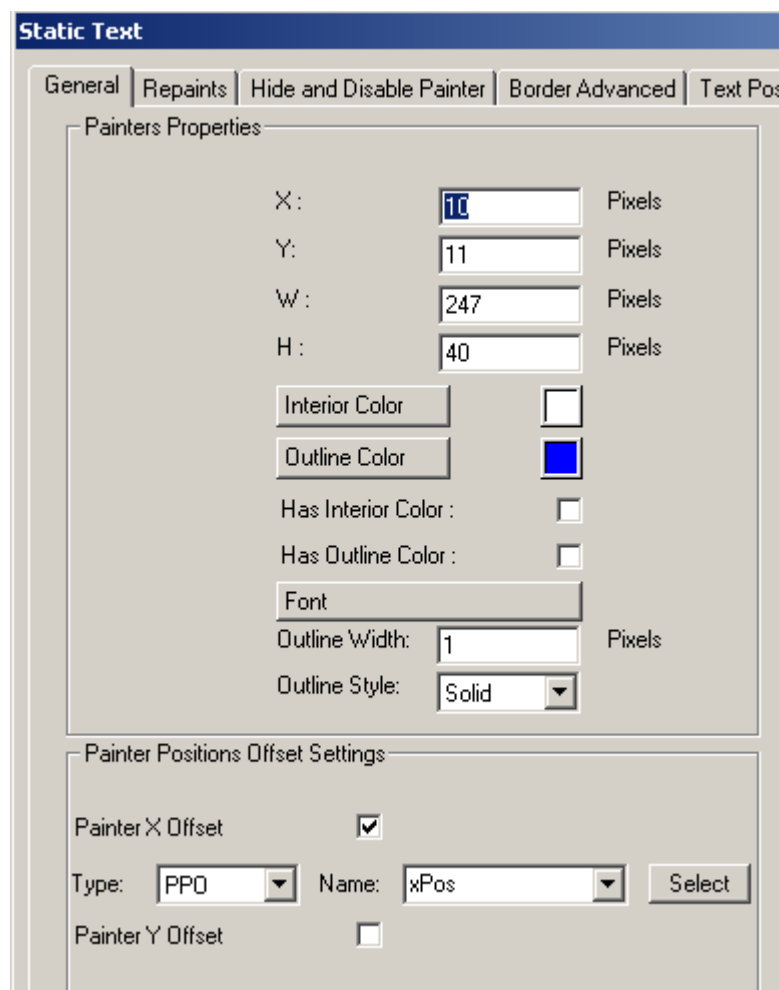
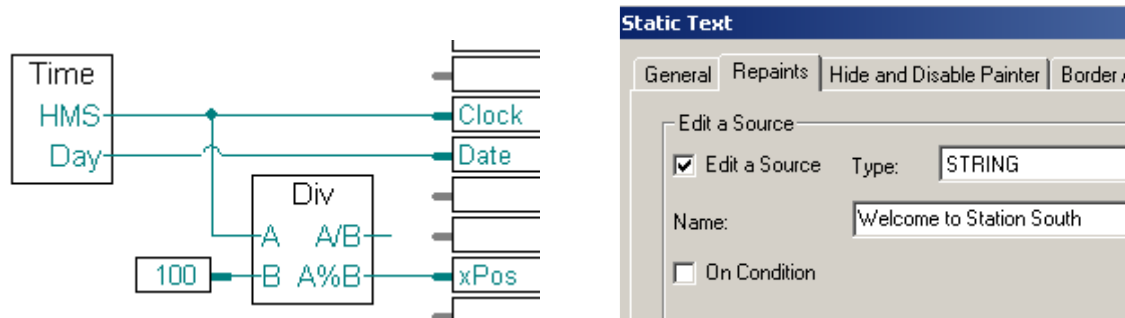
The same for 

Decrement -8

30 Moving painters in x and y axis

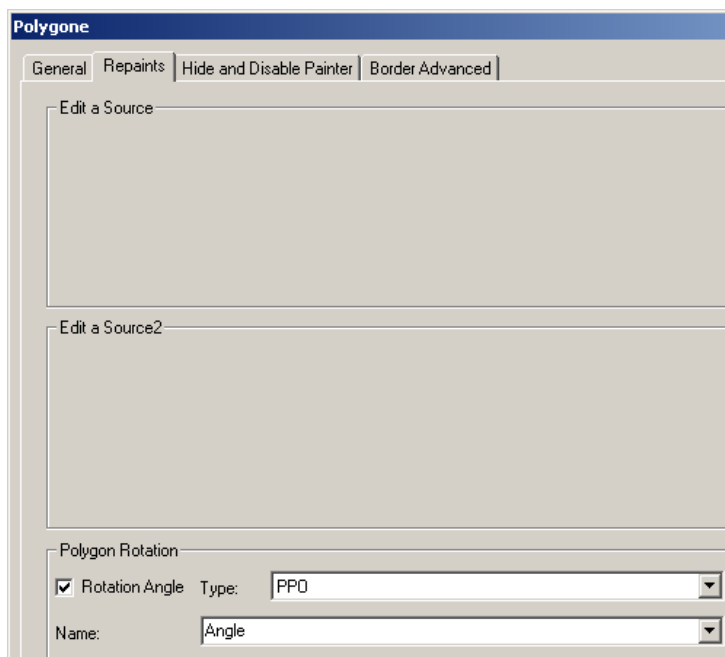
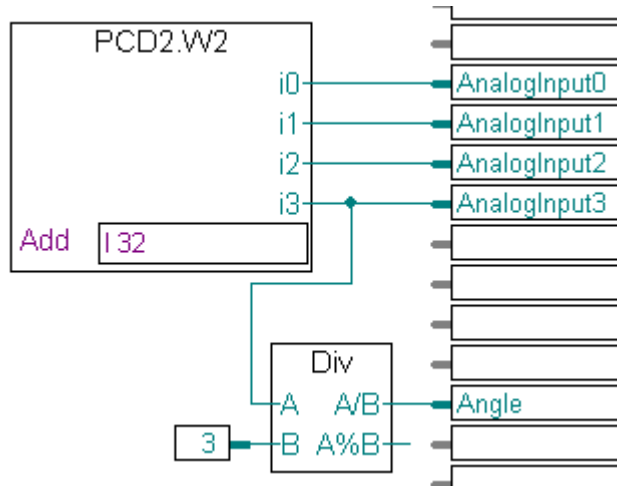
Each painter element can be moved along the x or y axis. The offset is entered as dx or dy.

In this example, a static text 'Welcome to Station South' is made to move along the x axis at one-second intervals. The PCD clock was used for the time base.

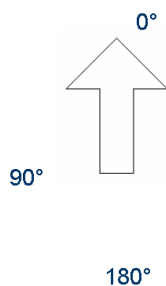


31 Rotation of polygons

All polygons can be rotated about their centre-points via the 'Repaint' tab.

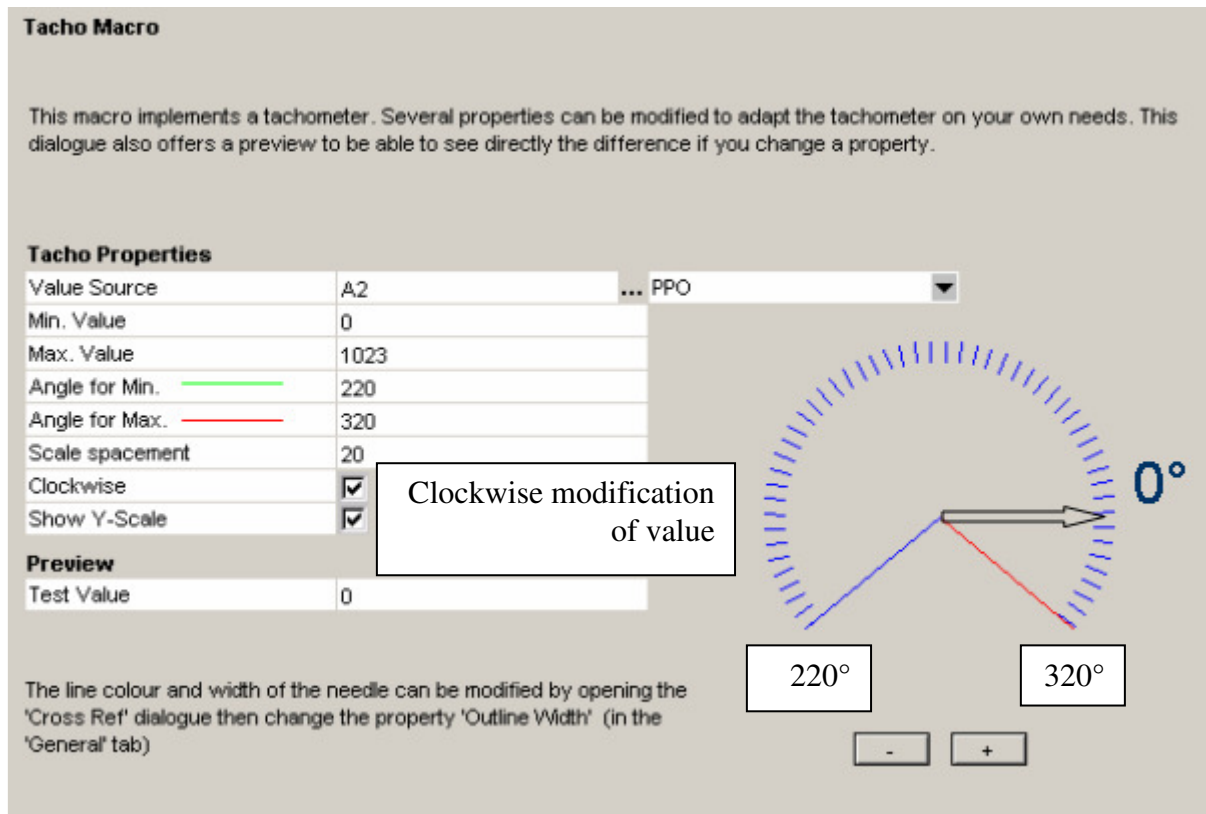


Note the appropriate angle definition as follows:



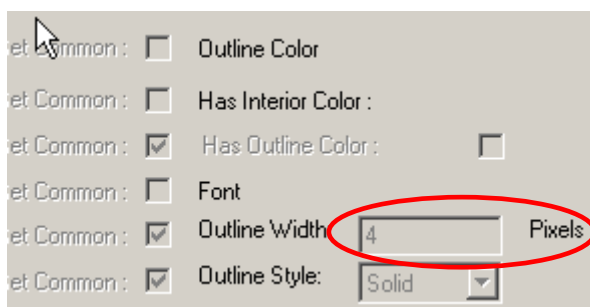
32 Tachometer macro

Variable values can now also be displayed with an analogue 'needle' pointer:
Tacho_5_13_40.esm

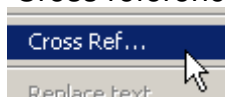


Note the somewhat peculiar definition of angular degrees. 0° is at 3 o'clock and the angle increases counter-clockwise.

The thickness of the needle pointer can be modified via the 'General' tab under 'Cross reference'.



'Cross reference' can be reached by right-clicking on the macro and selecting



33 Button for URL jump with address in argument

It is now possible, with the macro 'ButtonURLJump_onMouseDown_5_13_40.esm' to indicate which website to jump to when the button is pressed by including it as a STRING, PPO PCD variable, CONTAINER or HTML tag.

URL Jump Button (Mouse Down)

Button to jump to a URL specified by a PPO or a Container variable

Button Label

Type: STRING
Name: ButtonLabel

URL Address

Type: STRING
Name: STRING
Event: PPO
Name: CONTAINER
HTML TAG

OK CANCEL

For example, a navigation address bar can be programmed as follows.

your WebEditor!! address bar
with URL - jump on Button Mouse Down

http://www.meteoschweiz.ch



34 Foreground

The Basic Workshop demonstrates background layers. There is also a foreground layer on which, for example, superposed alarms are displayed.

It is best to enter any alarm that must be displayed over every web page in 'Project -> Project configurations ...'. It will then be superposed over any subsequently generated page.

Display with 'Static Text' box

PPO global register

Alarms are now displayed when IO == 1

Click on "CSV" button

In *.csv files, make additions to the various translation files

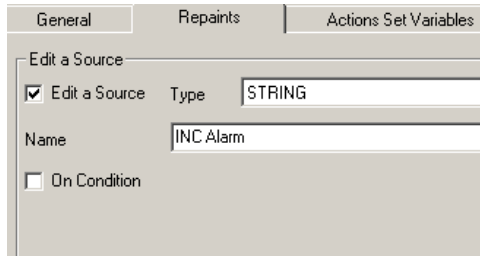
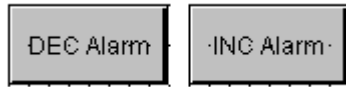
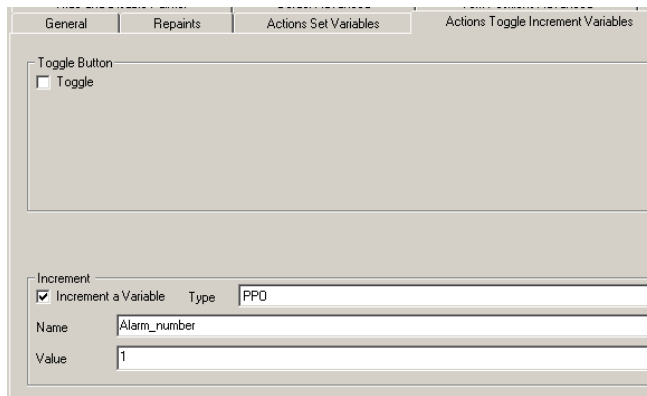
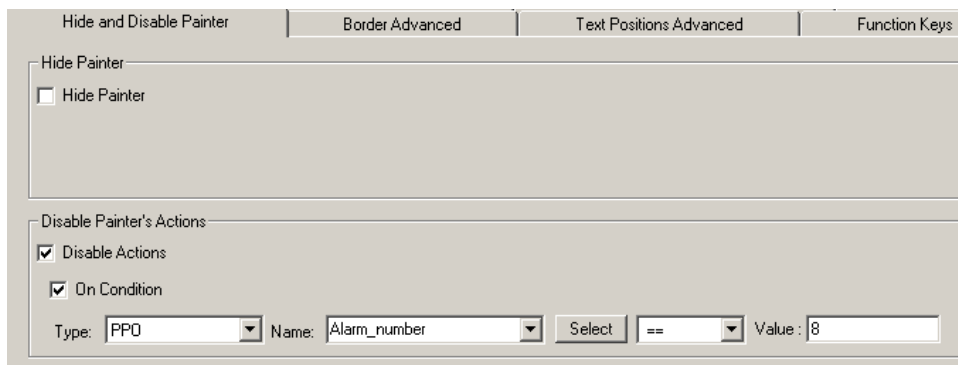
Alarm_text_0;No alarm
 Alarm_text_1;Potential drop
 Alarm_text_2;Low oil
 Alarm_text_3;Fault in heating system

Notation: Alarm_text_@PPO_Alarm_number@

PPO global register

Special syntax

Buttons to generate alarms on Entry.teq

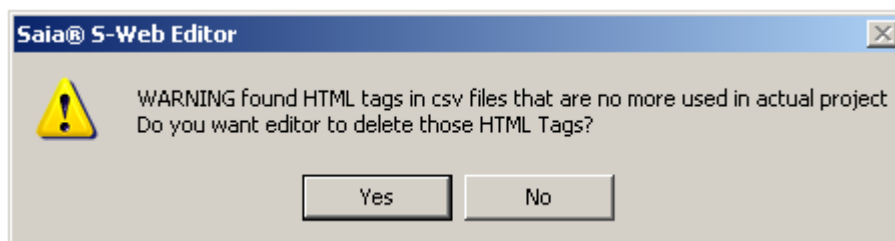
A dialog box with three tabs: 'General', 'Repaints', and 'Actions Set Variables'. The 'General' tab is active. It contains a section 'Edit a Source' with a checked checkbox 'Edit a Source', a 'Type' dropdown set to 'STRING', a 'Name' text field containing 'INC Alarm', and an unchecked checkbox 'On Condition'.A dialog box with four tabs: 'General', 'Repaints', 'Actions Set Variables', and 'Actions Toggle Increment Variables'. The 'Actions Toggle Increment Variables' tab is active. It contains a section 'Toggle Button' with an unchecked checkbox 'Toggle'. Below it is a section 'Increment' with a checked checkbox 'Increment a Variable', a 'Type' dropdown set to 'PPD', a 'Name' text field containing 'Alarm_number', and a 'Value' text field containing '1'.A dialog box with four tabs: 'Hide and Disable Painter', 'Border Advanced', 'Text Positions Advanced', and 'Function Keys'. The 'Hide and Disable Painter' tab is active. It contains a section 'Hide Painter' with an unchecked checkbox 'Hide Painter'. Below it is a section 'Disable Painter's Actions' with a checked checkbox 'Disable Actions' and a checked checkbox 'On Condition'. At the bottom, there are dropdowns for 'Type' (set to 'PPD') and 'Name' (set to 'Alarm_number'), followed by a 'Select' button, a dropdown for the operator (set to '=='), and a 'Value' text field containing '8'.

The screenshot shows the 'Edit a Source' configuration window. It has three tabs: 'General', 'Repaints', and 'Actions Set Variables'. The 'General' tab is active. It contains a section 'Edit a Source' with a checked checkbox 'Edit a Source', a 'Type' dropdown set to 'STRING', a 'Name' text field containing 'DEC Alarm', and an unchecked checkbox 'On Condition'.

The screenshot shows the 'Increment' configuration window. It has four tabs: 'General', 'Repaints', 'Actions Set Variables', and 'Actions Toggle Increment Variables'. The 'General' tab is active. It contains a section 'Increment' with a checked checkbox 'Increment a Variable', a 'Type' dropdown set to 'PPO', a 'Name' text field containing 'Alarm_number', and a 'Value' text field containing '-1'.

The screenshot shows the 'Disable Painter's Actions' configuration window. It has four tabs: 'Hide and Disable Painter', 'Border Advanced', 'Text Positions Advanced', and 'Function Keys'. The 'Hide and Disable Painter' tab is active. It contains a section 'Disable Painter's Actions' with a checked checkbox 'Disable Actions' and a checked checkbox 'On Condition'. Below these, there is a 'Type' dropdown set to 'PPO', a 'Name' dropdown set to 'Alarm_number', a 'Select' button, an '==' operator dropdown, and a 'Value' text field containing '0'.

Caution: Once the alarms have been added manually and the *.csv files updated by pressing the *.csv button, the system will ask whether unused entries should be deleted. SELECT NO. Otherwise, the manual entries will immediately be deleted.



35 Special links, actions and jumps with WebEditor buttons

FAQ #100791

A number of interesting features that can be executed with the WebEditor buttons

URL jump to another PCD and open this page in a new browser tab by using the "_new" argument.

The start page of the WebEditor project in the other PCD is called "start.htm".

The screenshot shows the 'Actions Jump' tab in the WebEditor interface. The 'Jump' section is expanded, showing options for 'View Jump', 'Back Button', and 'URL Jump'. 'URL Jump' is selected. The 'URL' field contains 'http://172.23.14.26' and the 'Frame' dropdown is set to '_new'. A 'Browse' button is next to the URL field.

URL jump to a www-address and open this page in a new browser tab by using the "_new" argument

The screenshot shows the 'Actions Jump' tab in the WebEditor interface. The 'Jump' section is expanded, showing options for 'View Jump', 'Back Button', and 'URL Jump'. 'URL Jump' is selected. The 'URL' field contains 'http://www.sbc-support.ch' and the 'Frame' dropdown is set to '_new'. A 'Browse' button is next to the URL field.

Open a *.PDF-File from the local flash card with a non-Micro-Browser in a new tab of the browser

Open a *.PDF-File from the local storage card with a Micro-Browser. Start the PDF-Viewer in a new Micro-Browser process

From the CE-Panel in the Micro-Browser, open Internet Explorer in a new window and specify the URL to open directly as an argument / parameter. Start Internet Explorer in a new process

Check if the user browses with a Micro-Browser or with a non-Micro-Browser to implement respective actions in the WebEditor project. Display / hide buttons, actions of the button, etc

Check the container variable "@MB_OS_VERSION". It contains the value "0" if you browse with a non-Micro-Browser or the respective version number of the Micro-Browser.