

RcWare SoftPLC HMI Editor

User manual



1 Contents

1	Contents	2
1	Introduction	3
1.1	Touch screen	3
1.2	Web panel	3
1.3	LCD display menu of the MiniPLC controller	3
2	Touch screen project	3
2.1	New project	3
2.2	New Panel	4
2.3	Retrieve variables	5
2.4	Inserting objects into panel	7
2.5	Copying of the objects	7
2.6	Object properties	7
2.7	Save project	16
2.8	Running and configuration of the project	16
3	Web access	20
3.1	Web server installation	21
3.2	Web server configuration	21
3.3	Web panel Service Manager	22
4	LCD display project	23
4.1	New project	23
4.2	Menu item types	24
4.3	Save project	29
4.4	Save project as	29
5	Alarms, SMS, E-mails	29
5.1	Touchscreen alarm table	29
5.2	Alarm SMS messages	31
5.3	Alarm e-mails	39
5.4	Alarm sounds	40

Any comments, questions and improvement suggestions to this manual are welcome at support@domat.cz. Thank you for your feedback.

www.domat.cz

1 Introduction

The HMI (Human Machine Interface) Editor is a software tool for designing and creating of HMI platforms for the SoftPLC runtimes. It is used for editing of those platforms:

1.1 Touch screen

Application which runs at the same or another process station as the SoftPLC runtime (used for I/O communication and process control). The Touchscreen application communicates with the runtime over a TCP port (12345 by default) and reads and writes values. One Touch screen services one or more runtimes, a runtime can be controlled and viewed by one or more runtimes. Therefore the system topology is very flexible.

1.2 Web panel

Web Panel is a web server that reads pages (panels) created by HMI Editor and exported to Web Panel as .svgz files, fills them with current values, and sends the resulting web pages to the browser. Web Panel is used for easy access to the process values with a thin client (web browser).

1.3 LCD display menu of the MiniPLC controller

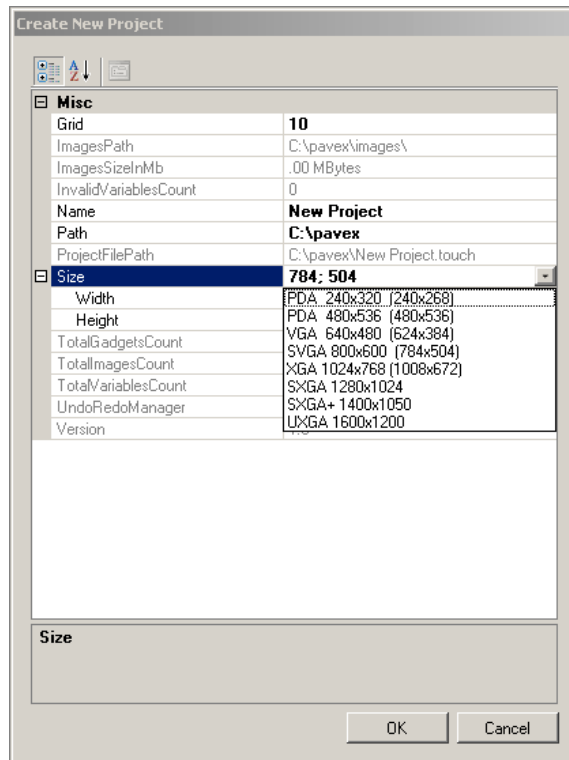
The LCD display menu for a MiniPLC controller can be freely defined in the HMI Editor and saved as a .lcdmenu project. The project is compiled and uploaded into the MiniPLC controller at a later stage in IDE – Platform Config menu, Upload LCD Menu files.

There are more types of objects (value display, value setter, time schedule, alarm, PIN protection etc.) which can be incorporated in the tree menu structure.

2 Touch screen project

2.1 New project

Create new project in the menu File – New – Project or Ctrl-N.



When a new project is created, there are some parameters to be specified:

2.1.1 Parameters of a new project

AlarmsConfigFilePath – complete path to the alarms.config file where predefined configuration of the alarms is stored (texts for pop-up windows, e-mail and SMS definitions etc.)

Grid – grid size for snapping of the objects in the editor, in pixels.

Name – project name; there will be created a directory of this name in the folder defined in the *Path* parameter.

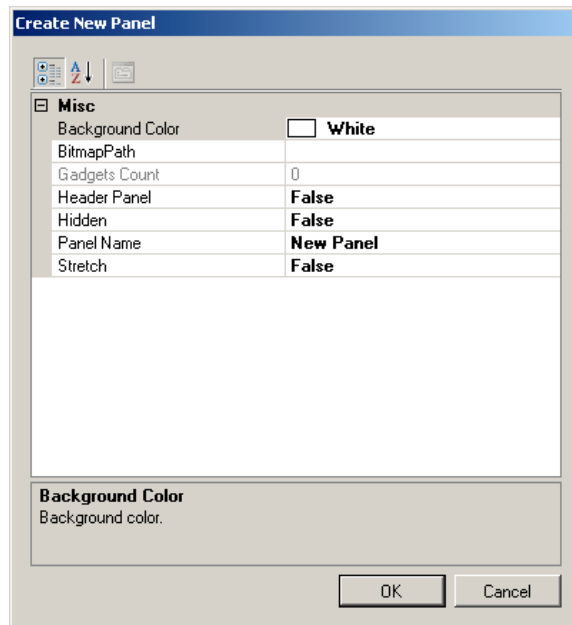
Path – file path to the Touch screen project, freely definable.

Size – dimensions of individual panels (screens). They are always a bit smaller than the screen resolution for which they are defined because there is another row of generic buttons below the panels. The background bitmaps (if used) should be sized according to the panel dimensions.

2.2 New Panel

After the project parameters are entered, a window with configuration of the first panel opens – each Touch screen application must contain at least one panel.

Another panels can be added in the *File – New – Panel* dialogue or by pressing Ctrl-P.



This dialogue contains following parameters:

2.2.1 New panel parameters

Background Color – background colour of the panel. Select one from the predefined colours in the selection boxes, or enter a specific colour as RGB triplet (three numbers 0...255 separated by a „;“).

Bitmap Path – full path and file name of the background image. The image is placed over the colour set in the Background Color parameter. Supported picture formats are .bmp, .png, .gif, and .jpg.

Header Panel – a parameter which specifies if this panel is a header panel. Header Panel may be only one in the project, it is located in the upper part of the screen and it is displayed along with the other panels. It may contain data which should be visible if any panel is displayed, e.g. outside temperature, system date and time, communication status, alarm summary etc.

Hidden – there is no button at the bottom bar generated for the panel. This panel can be displayed with the Button object which refers to the hidden panel („jump“). It is used if more panels than e.g. 10 are defined and the bottom bar would be overfilled with hardly readable panel names.

Panel Name – the name of the panel which is displayed in the panel list and at the bottom bar in the runtime. It usually corresponds with the displayed technology, e.g. „Boiler room“, „AHU Hall“ etc.

Stretch – stretches the background image to the whole size of the panel. Suitable for photographs etc., not so for technology schemas.

After a new panel is created in the working area, its edges are framed by a red line. Tabs at the upper part of the working area switch between the panels.

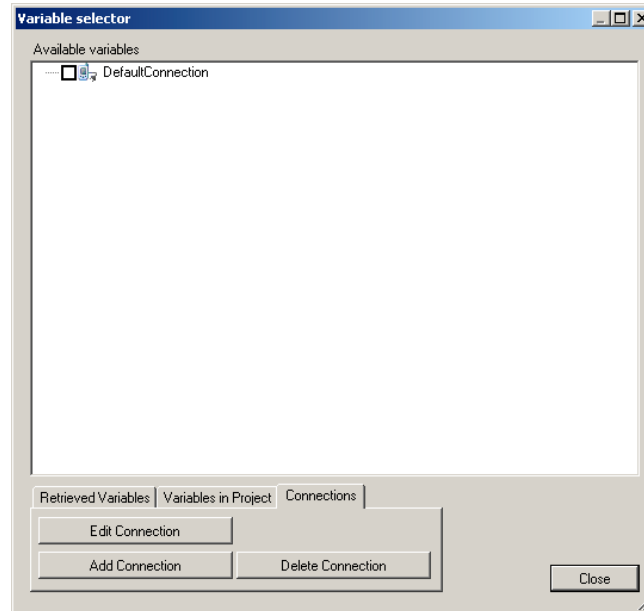
2.3 Retrieve variables

As the Touchscreen application works over a particular runtime (or runtimes), it is necessary to read in the HMI editor the variables which the runtime provides. From this list we select values to attach to objects (Attached Value, see below).

If the variables are not retrieved yet, a dialog for retrieving variables opens at an attempt to attach variable. The variables must be also re-retrieved after the application has been changed (adding of function blocks), so that the new variables are available to be attached to objects in the Touchscreen editor.

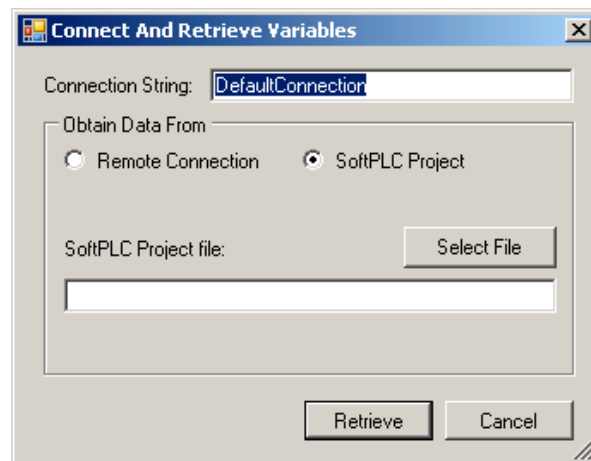
The variables are retrieved in the menu *File – Variables manager*.

Window with connection overview to the runtimes



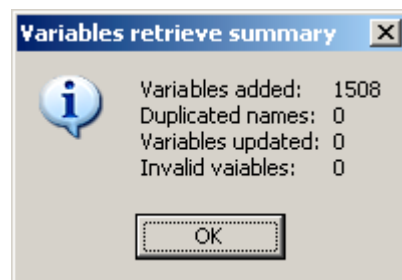
1. Select *DefaultConnection*.
2. In the *Connections* tab select *Edit connection*.
3. Rename connection, e.g. according to panel name or function group name („RMA“, „AHUroom“ etc.)

Select project with variables



4. Now there are two options: in case we want to retrieve data from the running runtime choose *Remote Connection*. If we have a source project created in IDE and want to import it as a file select *SoftPLC Project*. The latter is most used option.
5. If file is selected (*SoftPLC Project*) open the *Select File* dialog and choose a project file (.splcproj).
6. Import the variables with the *Retrieve* button. A status window appears which tells how many variables have been imported.

Imported variables info box



Variables added: amount of newly added variables

Duplicated names: number of variables with duplicate names. This is no error,

duplicate naming is allowed as the variables' unique identifiers are GUIDs and not variable names.

Variables updated: the variable changed its name but its unique ID (GUID) is unchanged. If this variable has been attached to an object in the HMI editor project, it is necessary to reassign it. The project identifies the variables according to their names.

Invalid variables: number of import errors.

7. After closing this window, a variable tree opens and variables can be selected and attached to the objects.

8. Close the window using the *Close* button.

This is how more connections can be attached so that from one touchscreen or web panel more process stations (runtimes) can be controlled. (This does not apply to LCD menu editor, where we only work with variables from one particular MiniPLC.)

2.4 Inserting objects into panel

Into the panels then objects are inserted which display or set values. Each object has properties which are visible at the panes at the right: the properties are both specific for the particular object type (attached value, state description of the buttons), and general visual attributes (background colour of the indicator, font type etc.).

It is also possible to change the properties of more objects at the same time, even if the objects are of different types. Only properties common for all the selected objects are displayed.

The objects are inserted by dragging & dropping from the left panel with the objects. Each object or group of selected objects can be moved and copied within the panel or between the panels.

2.5 Copying of the objects

1. Select the object by left-click of the mouse.
2. Copy the objects with Ctrl-C, Ctrl-V or in the menu Edit – Copy, Edit – Paste within the same panel. The position of the new instance of the object is slightly shifted so that the elements do not overlap.
3. Press Ctrl-C, switch between panels and Ctrl-V (or in the menu) to copy objects between panels. Positions of new elements are the same as they were in the source panel.

2.6 Object properties

The objects have properties (attributes), which can be edited in the right pane of the working area. Most of them are default with no need to change, of course the most important one is variable assignment.

2.6.1 Visual attributes

Common attributes for most of the objects, regarding especially background and text colour.

BackgroundColor – background colour of the object

Alpha – background transparency, 0: fully transparent, background colour does not apply, 255: full colour, not transparent.

Color – colour of the object.

BorderVisibility – if True, the object is framed by a thin black line.

Font – summary of properties of the font type

Name – font type

Size – font size in units specified below

Unit – units for font size, for IPCT.1 it is recommended to use pixels, other units are useful e.g. for touch panels for PDA, where the font size is often optimized for each device type in a different way and it may be useful to use e.g. Points.

Bold – attribute for bold typeface

Italic - attribute for italics typeface

Strikeout – striked-through letters

Underline – underlined letters

Layer – sequence of the object in the panel: the bigger this value is, the more „upstairs“ the object is located and overlays objects with lower Layer values. Can be changed by right click on the object and selecting *Bring to front / Send to back*.

TextAlignment – text alignment within the object shapes

TextColor – color of the text

Alpha – transparency, 0: fully transparent, text colour does not apply, 255: full colour, not transparent.

Color – color of the text.

2.6.2 Analog Indicator

Displays an analog value (integer, double).

AttachedValue – the value which is displayed, subvalues **ConnectionString** and **VariableName** are filled automatically after the variable has been assigned

DefinedFormatString – format of the value, i.e. number of decimals, if the variable is of Date/time type then local format (dd.MM.yyyy etc.) must be selected

FormatString – preview of the formatted string incl. unit

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

IsDateTime – if True, the value is formatted as Date and Time according to the **FormatString** entry

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

Prefix – the unit is displayed in front of the value and not behind it.

Unit – physical unit, can either be chosen from the predefined ones or entered manually.

Value – any value can be entered here to check the look (e.g. if negative or large values are displayed correctly).

2.6.3 AnalogPlusMinus

AttachedValue – the value which is set, subvalues **ConnectionString** and **VariableName** are filled automatically after the variable has been assigned

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

MaxValue – maximum value which can be set by repeated pressing of the “+” button

MinValue - maximum value which can be set by repeated pressing of the “-” button

Defaults

If both values are set to 0 (default setting), there is no limitation for user setting (no maximum nor minimum limits apply).

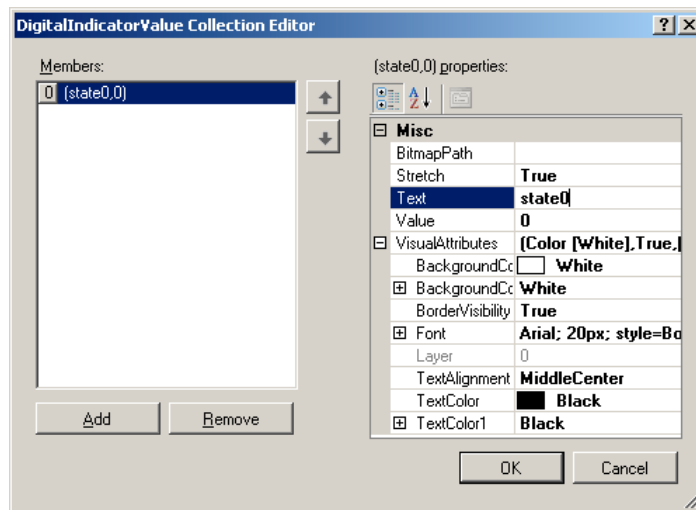
StepValue – value which is added or subtracted to the variable by pressing the “+” or “-” button

2.6.4 DigitalIndicator

AttachedValue – the value which is displayed, subvalues **ConnectionString** and **VariableName** are filled automatically after the variable has been assigned

DigitalIndicatorValues – this is the first object which has a predefined set of states (Collection) attached to various values of the variable.

Dialogue for states (Collection) definition



Click the button at the *Collection* line. A dialog appears, which contains status set in the left part and field for editing of properties of the selected state. Typical use can be found for indication of the On / Off status with „On“ / „Off“ texts or bitmaps with symbols for those states.

BitmapPath – if a bitmap should be displayed for this state, here is the path to it.

Stretch – stretches the bitmap to the size of the object (given by **Height, Width**). If False, only part of the bitmap beginning with upper left corner is displayed (if the object is smaller than the bitmap) or the rest of the object will be filled in by BackgroundColor (if the object is larger than the bitmap).

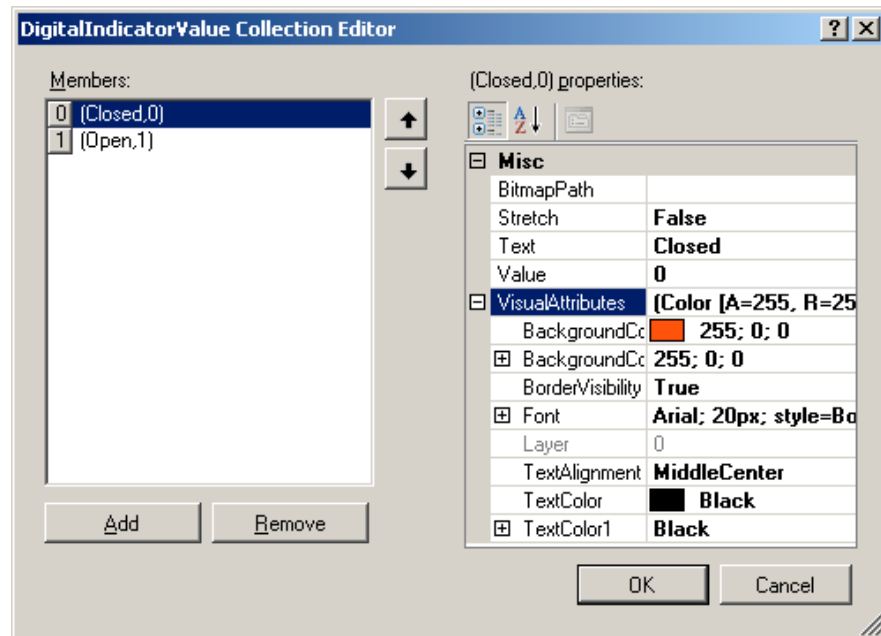
Text – if no bitmap is entered, this text will be displayed (e.g. “Comfort”).

Value – value for which this state is active. For Boolean variables use 0 = False, 1 = True.

VisualAttributes – attributes of text – see above (Object properties).

A typical two-state digital indicator may look like this:

Example of two-state digital indicator



If the attached variable is False, a black text „Closed“ on red background appears. If it is True, the text changes to „Open“, and the background goes green (definition not visible here).

The Add button inserts another item, the Remove button deletes active (selected) item.

The amount of items (Members) is not limited, the up and down arrows change their order.

After the collection is edited press OK to go back to the digital indicator dialogue.

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped.

However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

Note

Font parameters for all the items of a collection can be changed together using *Visual Attributes* of the complete digital indicator object.

2.6.5 DigitalSetter

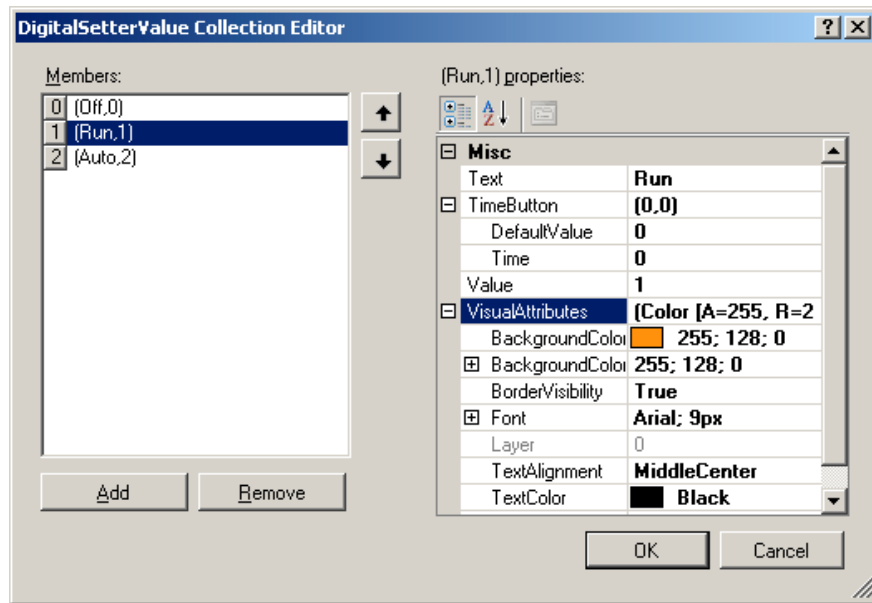
Setting element for digital (binary or multistate) values. Typically a switch *On / Off, Auto / 0 / I / II / III, Day / Standby / Night* etc. Can not be used for Double type variables.

The element displays as row or column (see **Layout**) of buttons, which are assigned to the individual states. The „not pressed“ buttons may be of different colours, the active (pressed) button always has the same colour (**HighlightedBackColor**).

AttachedValue – the value to be controlled, subvalues **ConnectionString** and **VariableName** are filled automatically after the variable has been assigned.

DigitalSetterValues – set (Collection) of different states (values) of this variable.

Example of a 3-state control element



The states (members) are added, removed, and edited in a similar way to Digital Indicator. However, it is not possible to use bitmaps and there is an extra property - **TimeButton**.

Text – Text for a particular state (e.g. “Active”).

TimeButton – button with time functionality: after pressing the timed button, the attached variable is set to **Value** for a time interval (**Time**) and then is set to **DefaultValue**. Typical use of this function is the acknowledge button: being pushed, it goes to True for about 5 seconds and then sets automatically back to False. Then the collection may contain only one state called (**Text**) „Acknowledge alarms“.

Value – value which is set to the variable if this state is active. For Boolean variables: 0 = False, 1 = True.

VisualAttributes – properties of the text (Object properties).

The Add button adds another item, Remove deletes the item selected.

Number of items (Members) is not limited, up and down arrows change their order.

After the editing is done, click OK to return to the digital element setting dialogue.

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

HighlightedBackColor – background color of the activated button

HighlightedForeColor – text color of the activated button

Layout – direction of the button ordering (horizontally / vertically)

Locked – if true, the object is locked so that it cannot be dragged and dropped.

However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

Note

Font parameters for all the items of a collection can be changed together using *Visual Attributes* of the complete digital indicator object.

2.6.6 Text

A very simple object for text displaying – usually description of the displayed or set value.

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

Text – displayed text string (such as “Room setpoint”). Text parameters are set in the Visual Attributes section.

2.6.7 Image

Displays a fixed image (bitmap) in the panel on a given position.

BitmapPath – file path to the bitmap

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

Stretch – stretches the bitmap to the size of the object (given by **Height, Width**). If False, only part of the bitmap beginning with upper left corner is displayed (if the object is smaller than the bitmap) or the rest of the object will be filled in by BackgroundColor (if the object is larger than the bitmap).

2.6.8 Button

Button to jump between panels.

If the **Hidden** parameter is set to True in the panel properties, there is no generic button created in the bottom bar of the touchscreen. This panel can be displayed using the Button object which refers to a hidden panel („jump“). It is mostly used if there are more than 10 panels in the touchscreen and names of the panels would be small and illegible.

BitmapPath – file path to the bitmap displayed on the button. If no image is defined the **Text** is displayed.

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing.

ReferredPanel – the panel to be switched to if the button is pressed.

Stretch – stretches the bitmap to the size of the object (given by **Height, Width**). If False, only part of the bitmap beginning with upper left corner is displayed (if the object is smaller than the bitmap) or the rest of the object will be filled in by BackgroundColor (if the object is larger than the bitmap).

Text – displayed text string (such as “Boilers”). Text parameters are set in the Visual Attributes section.

2.6.9 TimeProgram

Opens a dialogue for weekly schedule settings. The time schedule opens over the whole screen in a new window, after editing or viewing the original panel is displayed again.

AttachedValue – the value to be controlled, must be of TPG type (typically named ...*TPG_default*), subvalues **ConnectionString** and **VariableName** are filled automatically after the variable has been assigned.

BitmapPath – file path to the bitmap displayed on the button. If no image is defined the **Text** is displayed.

Geometry – dimensions of the object on the screen, **Height** – object height, **Width** – object width; **X, Y** – upper left corner of the object. Coordinate base (0,0) is in the upper left corner of the panel.

Locked – if true, the object is locked so that it cannot be dragged and dropped. However, the locked elements are subject to multiple selection and are copied and pasted in the same way as unlocked objects. Only relevant for editing..

Stretch – stretches the bitmap to the size of the object (given by **Height, Width**). If False, only part of the bitmap beginning with upper left corner is displayed (if the object is smaller than the bitmap) or the rest of the object will be filled in by BackgroundColor (if the object is larger than the bitmap).

Text – displayed text string (such as "Schedule AHU1"). Text parameters are set in the Visual Attributes section.

Binary scheduler table

	0 h.	1 h.	2 h.	3 h.	4 h.	5 h.	6 h.	7 h.	8 h.	9 h.	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Monday	VYP	VYP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP	VYP	VYP
Tuesday	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP	VYP	VYP
Wednesday	VYP	VYP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP	VYP
Thursday	VYP	VYP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP	VYP
Friday	ZAP	ZAP	VYP	VYP	VYP	VYP	VYP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP	VYP
Saturday	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP
Sunday	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	ZAP	VYP

Legend: False True

Buttons: Text mode, OK, Cancel

2.6.10 AlarmIndicator

Status indication of alarms. An alarm block (BD49, BD50) may come into those states:

Status	Description	Possible actions	K	A	M	alr_status value
OK	Normal state	-				0
Alarm Unacknowledged	Alarm active, unacknowledged	Acknowledge		×	×	1280
Alarm Acked	Alarm active, acknowledged	-	×	×	×	1792
Unreset	Alarm inactive, acknowledged	Reset	×		×	1536
Unacked Unreset	Alarm inactive, unacknowledged	Reset			×	1024

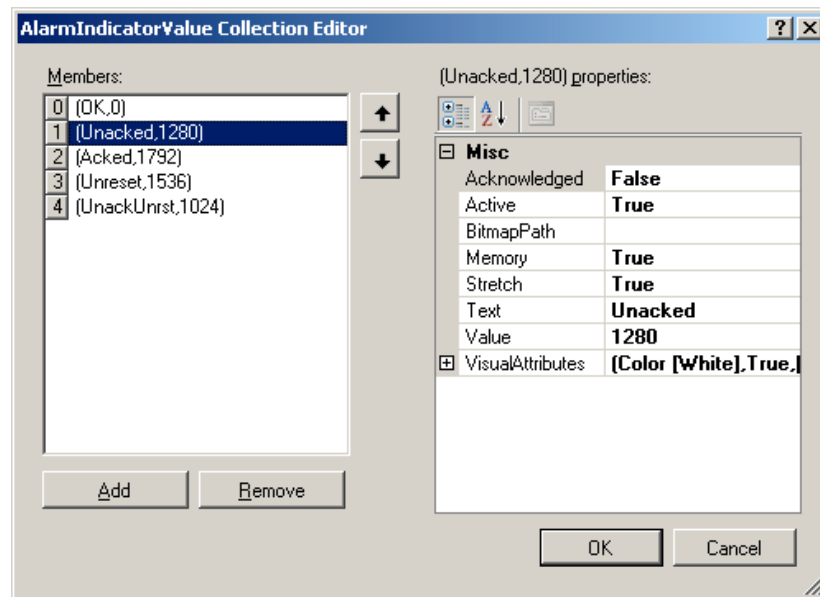
K – Acknowledged, attribute indicating that the alarm has been acknowledged (by a 0->1 signal at the *backn* input or from HMI)

A – Active, alarm signal True (at least one of the inputs *bx...* is True)

M – Memory, alarm not reset yet (resets by a 0->1 signal at the *brst* input or from HMI)

In the HMI editor it is possible to create a multistate object which displays those states. They can be indicated by texts or bitmaps.

Example of an alarm indicator definition



Object parameters are similar as for the digital indicator. In the status set (Collection) there are following parameters for each status:

Acknowledged – alarm was acknowledged, bit **K**

Active – alarm active, bit **A**

BitmapPath – file path to the bitmap if the states are indicated by bitmaps

Memory – memory output *bmem* is active, bit **M**

Stretch – stretches the bitmap to the size of the object (given by **Height**, **Width**). If False, only part of the bitmap beginning with upper left corner is displayed (if the object is smaller than the bitmap) or the rest of the object will be filled in by BackgroundColor (if the object is larger than the bitmap).

Text – displayed text string in case no bitmaps are defined

Value – not relevant here

VisualAttributes – see above (Object properties).

The main difference between Alarm indicator and Digital Indicator is that the Alarm Indicator after activating (click or touch) opens a dialog where alarm can be acknowledged or reset. In order to display the dialog properly, the alarm texts must be configured in the *Project alarm configuration* menu – see below.

2.6.11 Graph Gadget

Object to display set of analogue or digital value(s) in time – trend recorder. May be a memory eater, so use it reasonably. It has two basic forms :

- continuous flow in time and
- sampling over a given time span

– see **Type**.

Parameters:

AxisMaximum - maximum of the Y axis (applies only if FixedYAxis = true)

AxisMinimum - minimum Y of the Y axis (applies only if FixedYAxis = true)

BufferFileName – file name of the file for data storage (applies only if SaveBufferToFile = true)

ButtonText - text which is written at the button which opens the full-page graph (applies only if Kind = Button)

Counter – applies only if Type = QuarterHourMax, the variable containing seconds from the beginning of the interval of the current 15 min. (or another interval). Its subvalues **ConnectionString** and **VariableName** are filled in automatically after attaching a variable in the Counter dialogue

DataHoldLength – specifies how long (in seconds) the data are stored (applies to memory allocation, and if SaveBufferToFile = true, also for disk file). Values older than this interval are deleted

EstimatedValue - applies only if Type = QuarterHourMax - "contracted" value, is plotted in the graph as a comparison with the measured value. The subvalues **ConnectionString** and **VariableName** are filled in automatically after attaching a variable in the Counter dialogue

FixedYAxis - fixed Y axis, its range is specified by the AxisMinimum and AxisMaximum values. (If FixedYAxis = False, the Y axis scaling is automatically adapted to display the complete range of sampled data.)

Geometry – dimensions of the object at the screen, Height, Width; X, Y – upper left corner of the object. The (0,0) position is the upper left corner of the panel.

GraphValues – list of variables to display in the graph. Each value has those properties in the (Collection) table:

- **AttachedValue** – the value to be displayed, **ConnectionString** and **VariableName** are filled in automatically after attaching a variable in the dialogue
- **Color** – color of the curve
- **Description** – text which displays in the graph legend (only if LegendVisible = True)
- **HighlightMeasuredValues** – if True, the measured values are marked with dots
- **Width** – line width in pixels

Kind – Graph or Button. Graph displays the curves directly in the panel while Button displays only a button with text defined in the ButtonText parameter. If this button is

pressed, a new graph opens which covers the whole screen, and which also contains zoom and shift buttons.

LegendPlacement – choose one of the four corners. Only if LegendVisible = True.

LegendVisible – if True, the curve description is displayed, defined in Description and LegendPlacement.

Locked – only for editing mode, if Locked = True the object can not be moved. However, it is subject to copy operations.

MinimalSaveInterval – Interval to save the data on the disk, if SaveBufferToFile = True. “0” saves data at each value change (COV).

SaveBufferToFile – if True, the data is stored on a disk as often as defined in MinimalSaveInterval. When the application starts, the saved values are read in the graph.

Attention!

Do not use this function for flash memories, the media can be destroyed by frequent writing.

ShownDataLength – length of the displayed interval in seconds. However, there may be longer interval stored in memory – see DataHoldLength. The old, currently not displayed data can be invoked by zoom and shift functions.

Type – TimeGraph: normal current online trend, QuarterHourMax: there is a new graph each 15 mins, and parameters EstimatedValue and Counter apply.

NB.: You can define any interval for QuarterHourMax, not only 15 mins. The graph is reset as soon as the Counter value is less than its previous value.

2.7 Save project

Save the project by clicking the Disk icon, Ctrl-S or in the menu *File – Save*.

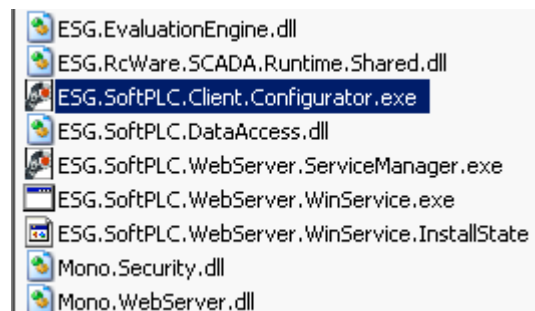
Project can also be saved under another name (*Save as...*), another menu item is saving all the open projects at the same time (*Save all*).

2.8 Running and configuration of the project

The saved project must be copied into the target process station. In the station, the Touchscreen runtime must be installed (the “viewer”).

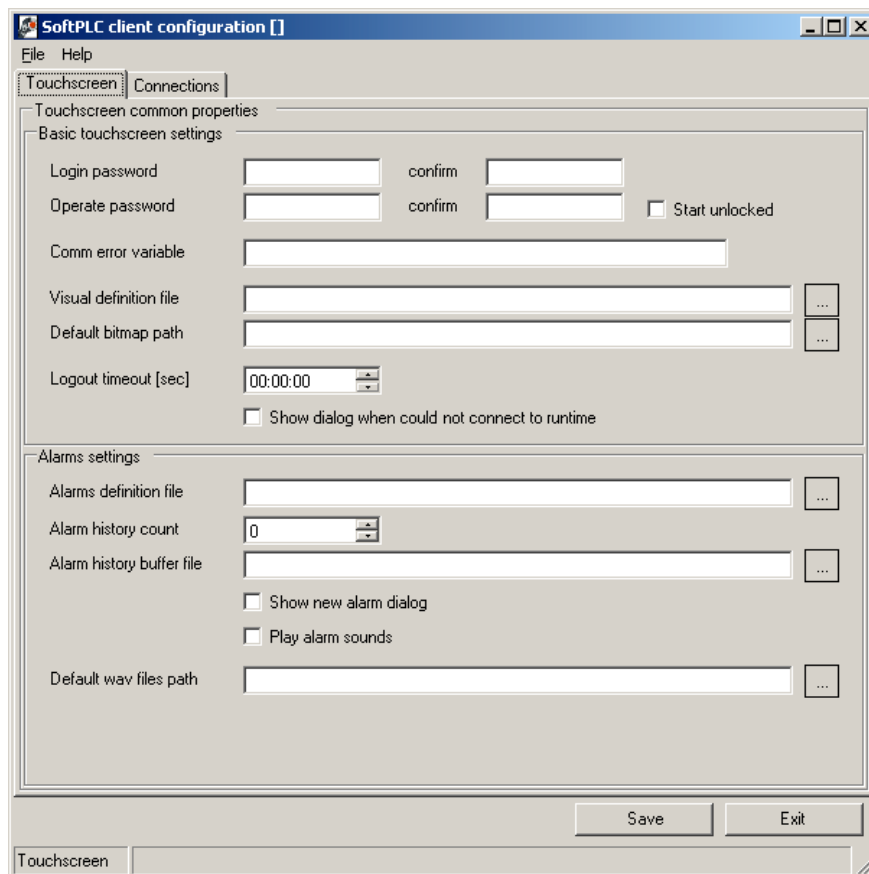
To configure all the clients of the SoftPLC runtime (the clients may be touchscreen runtimes, web panel, OPC server, and optionally other programs) there is an universal configurator called SoftPLC Client Configurator:

Client configurator



Run the Client configurator and select *File – New – Touchscreen*.

Set the path to the .touch file and other data:



2.8.1 The Touchscreen tab

Login password – max. 4 digit PIN for the service technician to log in and be able to shut down the Touchscreen application and return to the operating system HMI.

confirm – enter the PIN again to check.

Operate password – max. 4 digit PIN for the user to log in and be able to set values, time programs etc.

confirm – enter the PIN again to check.

Start unlocked – if checked, the *Operate password* is automatically logged in at startup so that the values may be changed even without entering the PIN. The **Logout timeout** is disabled. May be used where no unauthorised access to the HMI is expected.

Comm error variable – optional parameter: enter a variable which indicates communication failure if True. This failure is indicated on the Header Panel of the touchscreen by a special icon.

Visual definition file – path to the graphics definition file (.touch) which was designed in the Touch screen editor.

Default bitmap path – path to the bitmaps. In case they are not used in the graphics, leave blank. Bitmaps are used for panel backgrounds, images, icons in the animations in Collections, icons on the buttons etc.

Logout timeout – in case of user inactivity (not touching the screen) the display locks after elapsing this time and the **Data access password** or **Login password** must be entered again.

Show dialog when could not connect... – if the connection with the runtime is lost or can not be established, a warning pop-up appears. Usually not used.

Alarms definition file – path to the alarms.config file, where additional definitions of the alarms are stored (explaining text etc.). Must be entered for proper alarm display in the generic alarm table and pop-ups.

Alarm history count – number of records to be stored in the alarm history table. Older records over this amount will be deleted.

Alarm history buffer file – file path and name for alarm history storage. If blank, after touchscreen restart the alarm history is not available.

Show new alarm dialog – if ticked, alarm pop-up window appears at each new alarm appearance. It is possible to acknowledge and reset the alarm in the pop-up window.

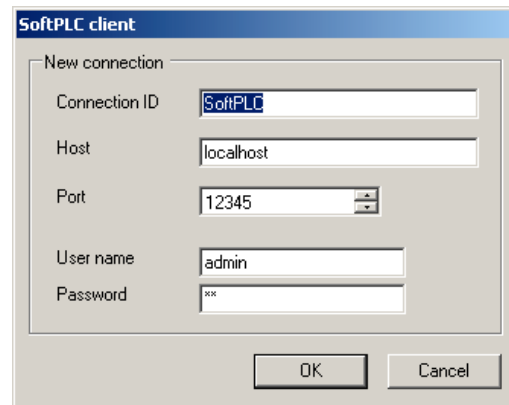
Play alarm sounds – plays alarm sounds according to settings in the *Alarms* menu in the IDE.

Default wav files path – path to the .wav files which are played as alarm sounds.

After this, the connection(s) to runtime(s) must be defined:

2.8.2 The Connections tab

Define a new connection clicking the *Add connection* button.



Connection ID – connection name, must be same as the connection name entered at the retrieving variables dialogue in the touch screen editor (*Connection String*).

Host – IP address of the machine where the SoftPLC runtime is running, i.e. where the data are read from. In case of IPCT.1 and everywhere the HMI runs at the same machine as the runtime it is „localhost“.

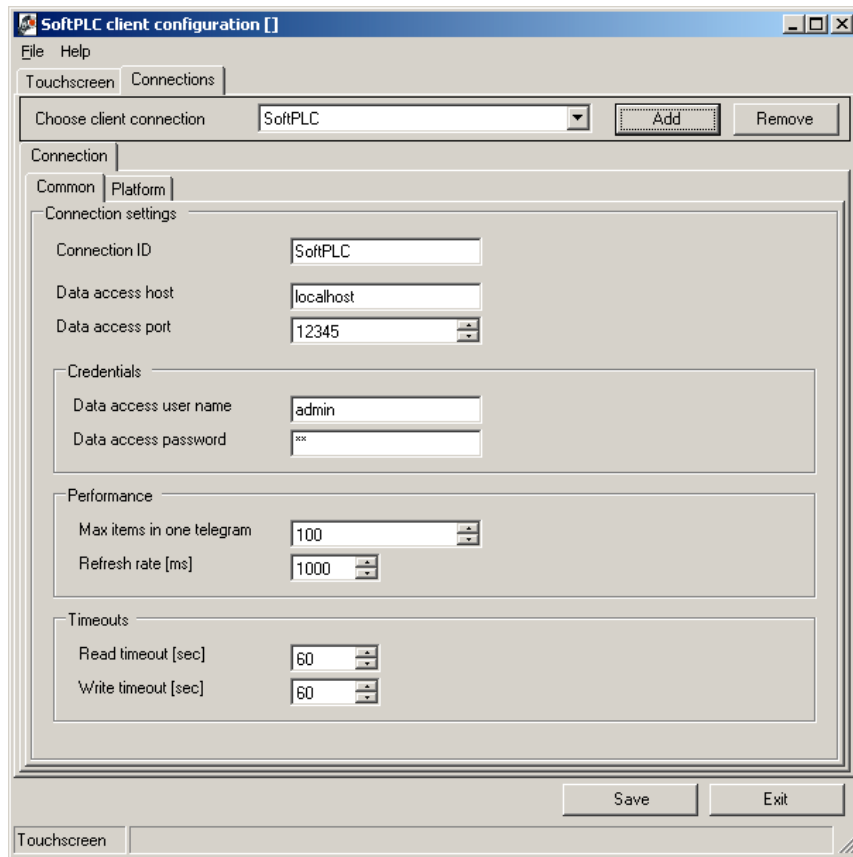
Port – TCP port where the runtime accepts connections from clients (touch screens, OPC servers etc.). Default value is „12345“, can be set at the servers (runtimes) for SoftPLC in the SoftPLC runtime settings (*Options*), for MiniPLC in the IDE, *Platform Config* dialogue.

User name – name under which the touch screen logs in to the runtime. Default value is „admin“. Settings are similar to the Port parameter.

Password – password under which the touch screen logs in to the runtime. Default value is „rw“. Settings are similar to the Port parameter.

Save the configuration clicking the *OK* button. A new window with the connection parameters opens. Some of the parameters are taken from the previous dialogue:

General connection parameters



Normally, it is not necessary to change those parameters.

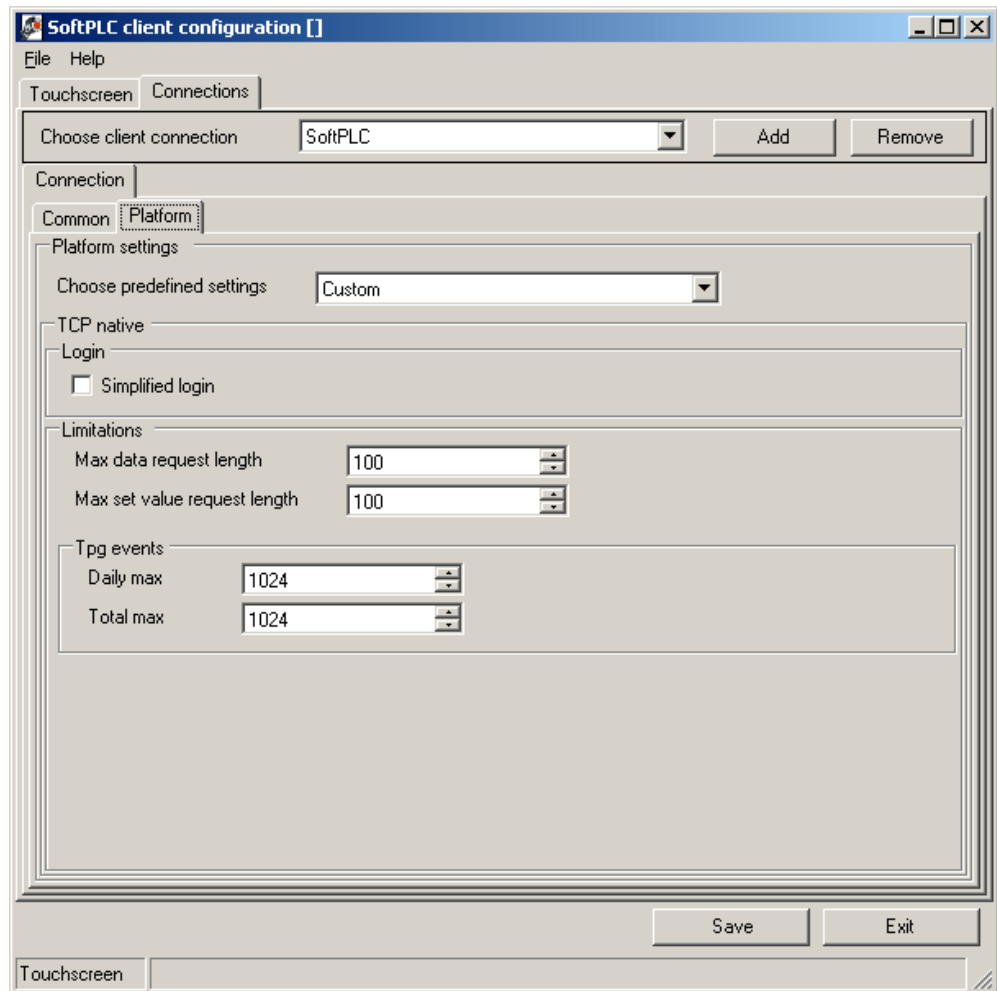
Max items in one telegram – system parameter for optimizing communication in case of remote access over low throughput and / or high latency networks.

Refresh rate – update interval, increase in low throughput networks.

Read timeout – read timeout insensitivity, after a packet has not been received until this time the TCP connection is declared as lost and it is attempted to reopen it again.

Write timeout – write timeout insensitivity, after a packet has not been received until this time the TCP connection is declared as lost and it is attempted to reopen it again.

Another tab is the *Platform* tab:



In the *Choose predefined settings* box select among

Full, i.e. an OS Windows XP / XP Embedded or Linux based runtime, and

MiniPLC, a hardware platform with limited computing power and thus different communication properties.

Custom is not used, it is an option how to fine tune some of the communication parameters.

The settings are saved into a config file named *soft_plc_touchscreen.config* (do not change the file name) by clicking the *Save* button.

After the configuration has been finished, run the *SoftPLC.Touchscreen* which runs the application (.touch), establishes communication with the runtime and displays the graphics on the screen.

For IPCT.1, add a shortcut to this file to the Startup directory, so that after the operating system starts the *SoftPLC.Touchscreen* starts too and users do not have access to the operating system HMI and menu.

3 Web access

To enable web access to the runtime data, a service (web server) must be installed. It connects to the runtime, reads data and interprets them as web pages. The definition of the web pages is stored in files named *panelX.svgz*. Those files have to be created in the Touch Screen Editor in the same way as the touchscreen graphics.

If the process station has to have web access enabled, there is no need of engineering an extra web graphics: the touch screen graphics is only exported.

After saving of the .touch project, export data in the *File – Export to Web Panel* menu.

The same way applies for MiniPLC. To transfer the files to the MiniPLC controller, use button in the *Platform Config* dialogue in the IDE.

3.1 Web server installation

The web server is installed either to the process station or to another computer which may be a web server for more runtimes. The installer is *SoftPLC.WebPanel.WinService.Install.msi*. The server runs as a Windows service which can be configured by the *SoftPLC.Client.Configurator*.

3.2 Web server configuration

Run the *SoftPLC.Client.Configurator*. Select *File – New – Web server*.

The screenshot shows the 'SoftPLC client configuration' dialog box. It has a menu bar with 'File' and 'Help'. There are two tabs: 'Web server' (selected) and 'Connections'. The 'Web server' tab contains two sections: 'Web Server Settings' and 'SoftPLC Web Panel properties'. 'Web Server Settings' includes a 'Protocol' dropdown set to 'HTTP', a 'Port' dropdown set to '8080', and two empty text boxes for 'Key file (for HTTPS certification)' and 'Key password (for HTTPS certification)'. 'SoftPLC Web Panel properties' includes a 'Login' section with 'Name' and 'Password' text boxes, and a 'WriteAccess' section with 'PIN' and 'Timeout' (set to '00:00:00') text boxes. At the bottom right are 'Save' and 'Exit' buttons. A status bar at the bottom left shows 'Web server'.

Set following parameters:

Protocol – HTTP or HTTPS, according to security demands etc.

Port – TCP port, at which the server responds.

Key file (for HTTPS certification) – in case of HTTPS protocol, enter path and file name to the key file

Key password (for HTTPS certification) – password for HTTPS protocol

Login Name – set user name which has to be entered at the user login prompt

Login Password – set password for web users

This Name / Password login enables data viewing only, it is not possible to change values after login.

Write Access PIN – 1 to 4 digits PIN which enables changing values over the web interface.

Write Access Timeout – if the user is inactive for this time, it is necessary to re-enter the PIN for write access. Web (TCP) connection is not terminated after timeout expiration.

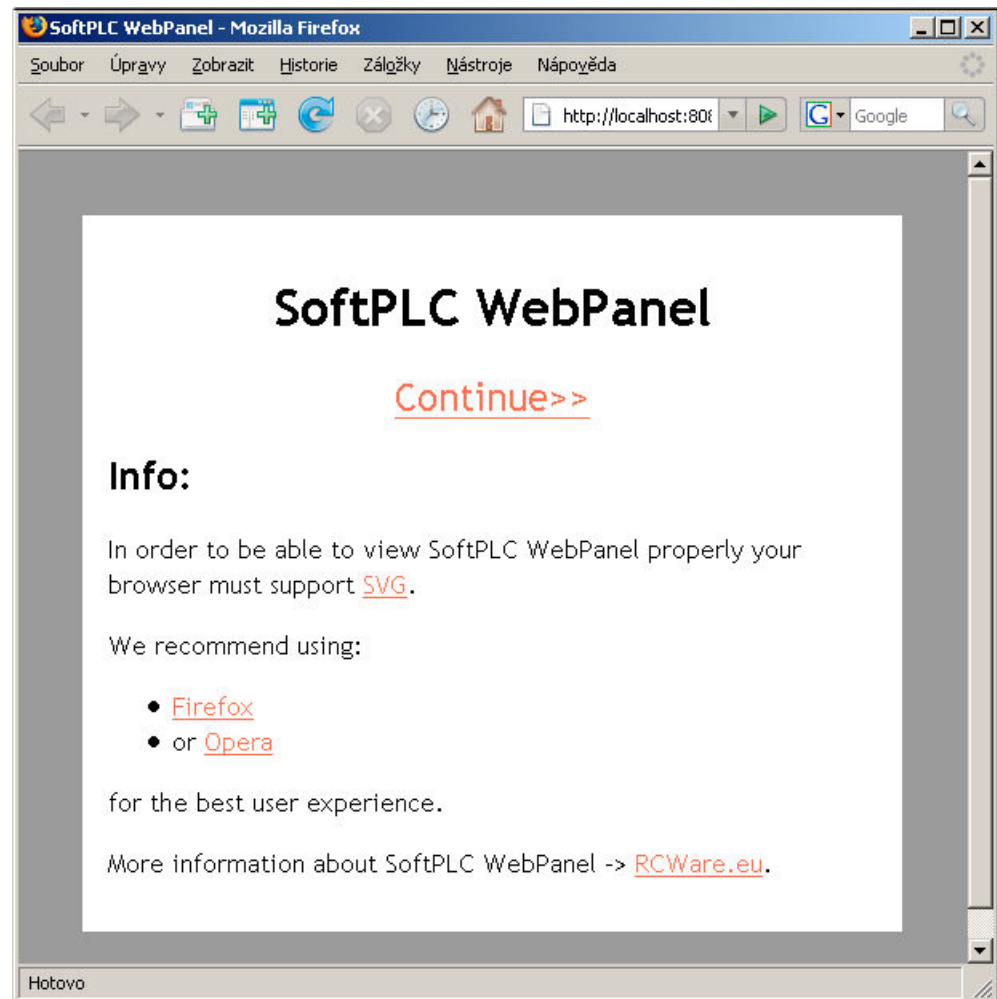
In the *Connections* tab, set the connection from web server and runtime in the same way as for the touch screen – see above.

Save the settings clicking the *Save* button into the file named *C:\Program Files\RcWare\SoftPLC.WebPanel.WinService\webpanel\soft_plc_ws.config*.

The exported web pages and scripts from the touch screen editor (see above) must be copied into the process station (or machine which works as the web server, which may be any computer in the network) into the web access directory: *C:\Program Files\RcWare\SoftPLC.WebPanel.WinService\webpanel*.

In this directory, *index.html* is the default web page which can be edited by any html editor – you can localize it, add your company logo, links to other panels, etc. The *Continue* link refers to the first panel.

Default page:
index.html

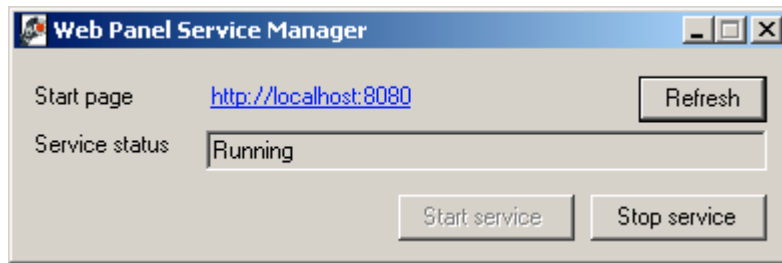


3.3 Web panel Service Manager

To start and stop the web server service, and to acquire information about the service status, there is Web panel Service Manager. It is installed in *C:\Program Files\RcWare\SoftPLC.WebPanel.WinService* and after it is launched, an icon indicating service status appears on the Windows toolbar:

green arrow – server is running
red square – server is stopped.

Click the icon to open Service Manager window:



Here the default URL and server status are displayed. Start or stop the service by clicking the buttons below.

4 LCD display project

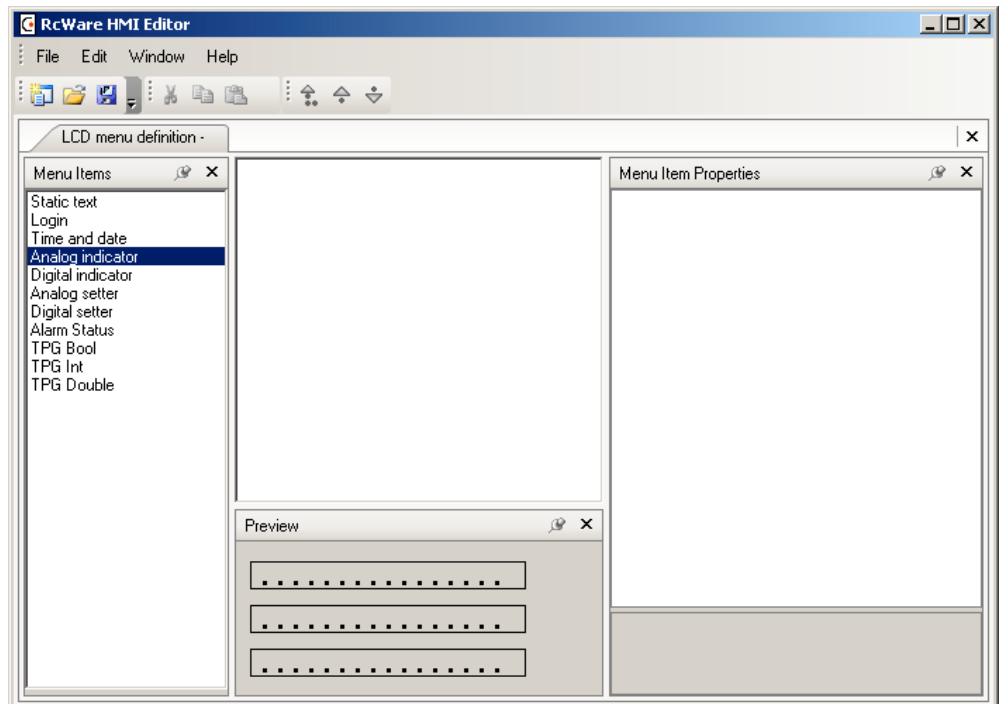
This project type defines LCD display menu structure for the MiniPLC controllers. MiniPLC has a 3 rows x 16 characters display and 6 buttons to control the menu.

A prerequisite for menu creation is the completed application software created for the MiniPLC platform, namely the *.sproj file. From this file, variable names are imported.

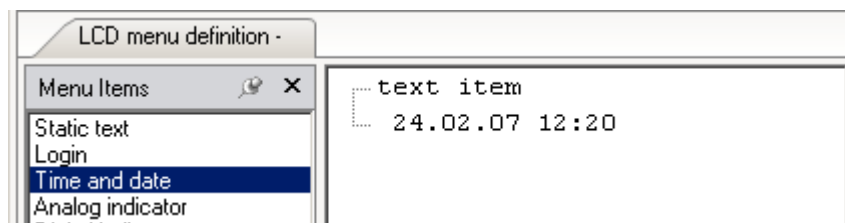
4.1 New project

Create a new project in the *File – New – MiniPLC LCD Menu definition* menu. A working area appears:

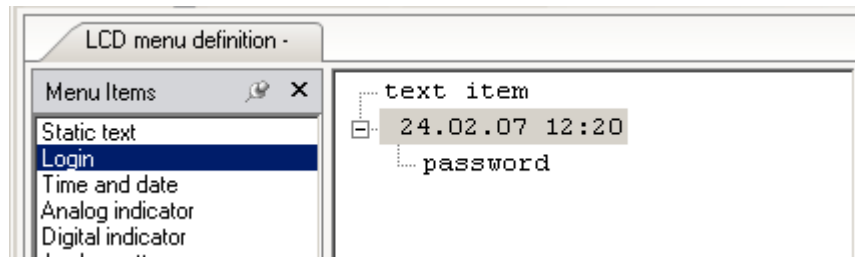
LCD menu editor



In the left part, there is a list of available menu item types (Menu Items) which may be dragged & dropped into the middle panel. The item is dropped either under existing items:



or into any of the existing items:



which is how to create the tree menu structure. Number of items in one branch is limited to 100.

If any of the menu items contains subitems (another branch of the menu tree or a generic dialogue) the last character in the line reads „>“. Enter the submenu by positioning this item on the second (middle) display row and pressing the Enter button.

Items or item trees may be moved in the menu by dragging & dropping or using the *Move to Parent*, *Move Up*, *Move Down* buttons at the toolbar.

In the middle pane there is a working area where the menu tree is created. Under the tree, there is a preview to display how the item will look at the MiniPLC display. This is useful for checking of decimal places definition, length of texts etc.

In the right pane are the properties of the active item. There is where the object (menu item) is configured.

4.2 Menu item types

4.2.1 Static text

A fixed text is displayed. Use for welcome menu (commission name), menu branch names (Setpoints, Settings, Heating, Lights etc.), or for a submenu with contact to the installers who commission and service the plant, so that the data is always available and can be updated by the service personnel.

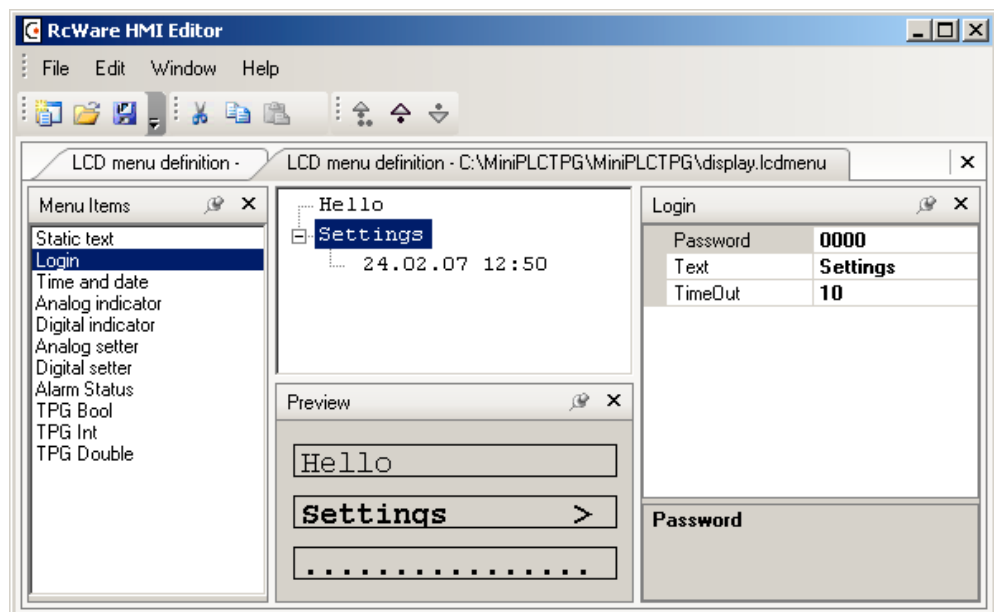
The Static text item has only a single parameter:

Text – text with max. length of 16 characters.

4.2.2 Login

Login enables to lock a menu branch by a 4-digit PIN.

Login: Example of use



If this item is selected, there is a prompt to enter the PIN which unlocks the submenu branches. It is not possible to access those subitems without entering the PIN. At the above example, controller date and time setting is protected.

The LCD menu may contain any number of branches with identical or different PINs.

Parameters:

Password – 4-digits PIN for access to submenu branches. It is not possible to change the PIN for the user.

Text – text with max. length of 15 characters.

TimeOut – inactivity timeout (minutes), after which the submenu is locked and PIN must be entered again.

4.2.3 Time and Date

The MiniPLC contains real-time clock. Its status is displayed and set with the *Time and Date* item.

DateTime – displays current date and time of the controller.

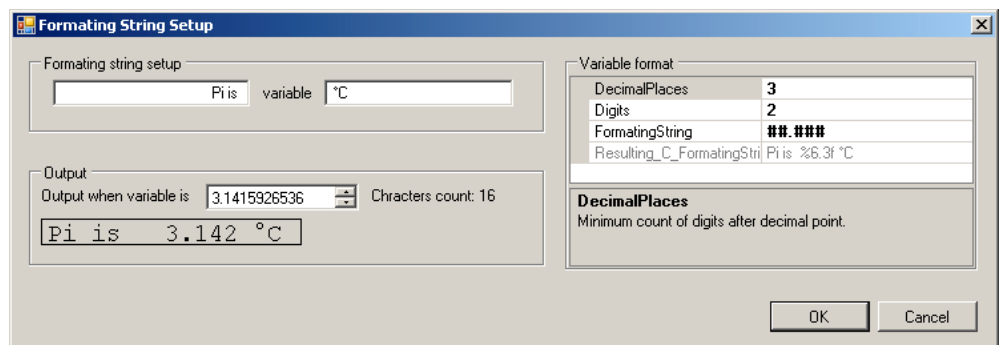
Editable – if True, date and time may be set by a generic dialogue which is open by entering the menu row with date and time indication. If False, the Date and Time is read-only.

4.2.4 Analog Indicator

Displays value of an analogue (double, integer) variable.

FormattingString – string with the text and value format. Click to the small button in the row to enter the dialogue with detailed settings:

Dialogue for analogue value format setting



DecimalPlaces – number of decimal places (right from the decimal separator)

Digits – number of figures on the left side of the decimal separator. This number also includes the minus sign, take into account negative values if relevant (e.g. relevant for outside temperature $-30...50^{\circ}\text{C}$, not relevant for the damper position $0..100\%$).

FormattingString – resulting formatting string

Resulting_C_FormatingString – just for check, displays the formatting in the C language.

In the Formatting string setup text fields, the explaining text and unit are entered. Set any value in the *Output* field and watch the resulting format in the preview.

Click OK to close the window and return back to the Analog Indicator properties.

Variable – výběr proměnné, která se má zobrazit. Postupuje se stejně jako v Touch screen editoru, viz výše – **Načtení proměnných**, pouze s tím rozdílem, že lze importovat proměnné pouze z jediného runtime; displej zobrazuje proměnné z jednoho regulátoru MiniPLC.

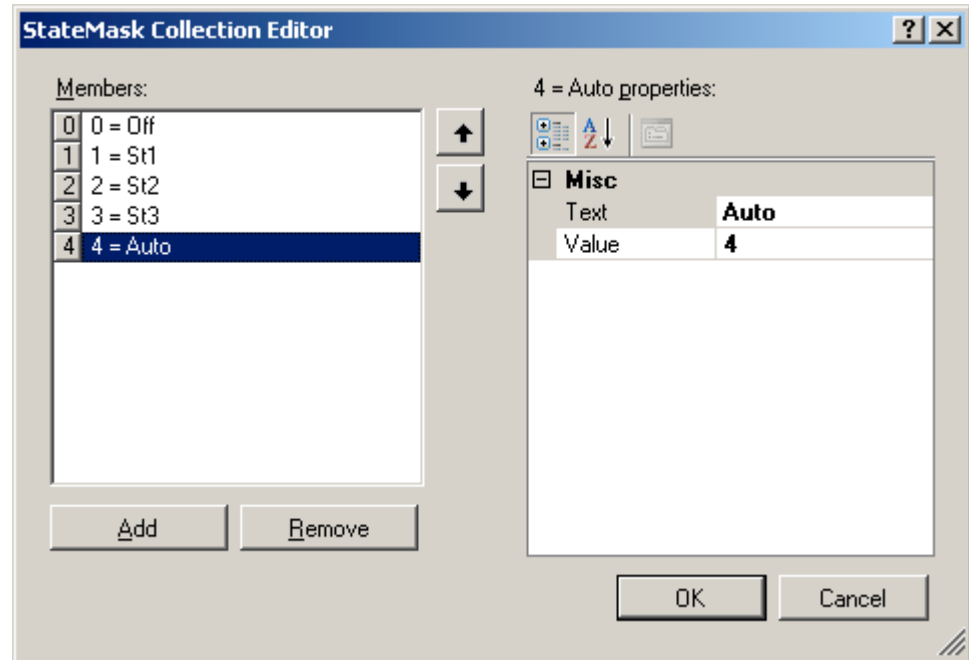
4.2.5 Digital Indicator

Displays a multistate value, i.e. boolean or integer which has assigned states to its values. There may be maximum 8 states (collection members) in the Indicator.

FormattingString – resulting formatting string. Enter the text here, „#” stands for a text derived from the Collection.

StatesMapping – set of possible states of the variable and their descriptions.

*Definition example:
multistate value –
fancoil status*



Click *Add* and *Remove* to add and remove status lines (collection members).

Text – state text

Value – variable value for this state. For Integer, the value is any number from the Integer range (not necessarily 0, 1, 2, ...).

Typical Boolean settings is

Off = 0

On = 1.

Click OK to close the window and return back to the Digital Indicator properties.

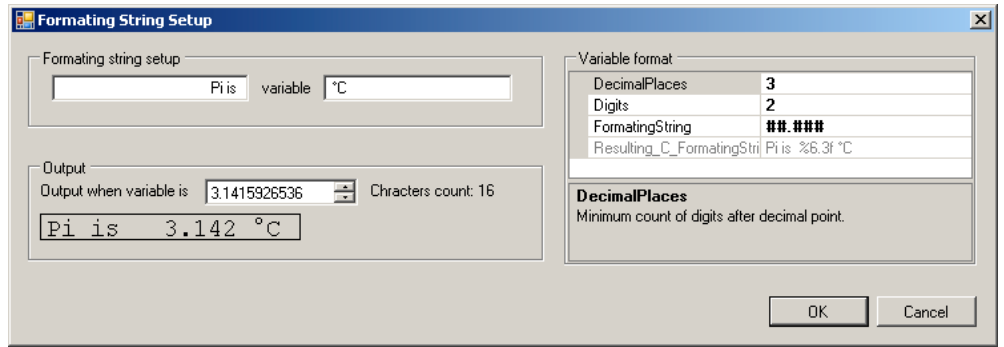
Variable – select the variable to be displayed.

4.2.6 Analog Setter

Item for analogue value setting (double or integer type variable).

FormattingString – string with the text and value format. Click to the small button in the row to enter the dialogue with detailed settings:

*Dialogue for analogue
value format setting*



DecimalPlaces – number of decimal places (right from the decimal separator)

Digits – number of figures on the left side of the decimal separator. This number also includes the minus sign, take into account negative values if relevant.

FormattingString – resulting formatting string

Resulting_C_FormattingString – just for check, displays the formatting in the C language.

In the Formatting string setup text fields, the explaining text and unit are entered. Set any value in the *Output* field and watch the resulting format in the preview.

Take into account that the last character in the row is „>“, indicating that this value can be changed in the generic submenu.

Click OK to close the window and return back to the Analog Setter properties.

Max – maximum value, upper limit that can be set

Min - minimum value, lower limit that can be set

Step – increase or decrease of the value on pushing the „arrow up“, „arrow down“ buttons.

Variable – select the variable to be displayed and set.

4.2.7 Digital Setter

Item for multistate value setting (boolean or integer).

DefaultValue – only if **TimeFunction** = True, value which the variable is set to after **Duration** time

Duration – only if **TimeFunction** = True, time in seconds for which the variable is set to the value entered in Collection. After this time the variable is set to **DefaultValue**.

FormattingString – resulting formatting string

StatesMapping – set of possible variable states and their textual interpretation. Setting is the same as for *Digital Indicator*.

Maximum number of states

Note that the maximum available number of states is 8.

TimeFunction – if True, the item works as a pushbutton which sets temporarily the variable to a certain value and after **Duration** it automatically sets to **DefaultValue**. Use as alarm acknowledge, pulse source etc.

Variable – select the variable to be displayed and set.

4.2.8 Alarm Status

Alarm status indicator. An alarm block (BD49, BD50) may come into those states:

Alarm indicator states

Status	Display	Description	Possible actions	alr_status value

OK	OK	Normal state	-	0
Alarm Unacknowledged	*	Alarm active, unacknowledged	Acknowledge	1280
Alarm Acked	*√	Alarm active, acknowledged	-	1792
Unreset	-X	Alarm inactive, acknowledged	Reset	1536
Unacked Unreset	-	Alarm inactive, unacknowledged	Reset	1024

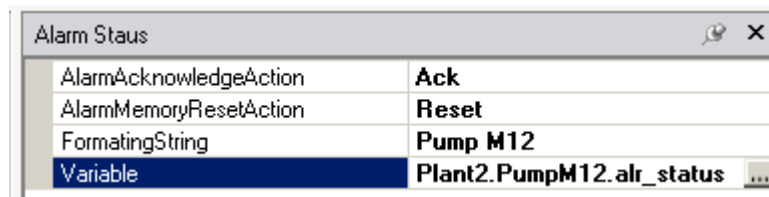
AlarmAcknowledgeAction – text to display in the alarm acknowledge dialogue

AlarmMemoryResetAction – text to display in the alarm reset dialogue

FormatingString – description of the alarm on the display, e.g. „FrostAHU1“, „Leakage“, „PumpM4“ etc.

Variable – attached variable *.alr_status*, which is the output of an alarm block.

Typical alarm definition



Use localized Ack and Reset strings if necessary.

4.2.9 TPG Bool

Boolean time scheduler.

This item displays text (schedule name), actual status and a reference to the table with switching events – status changes. The events may be edited in a generic dialogue.

Text – scheduler name, the „#“ sign is substituted by a text according to the actual scheduler state taken over from the IDE. It is not possible to change the status texts here.

Variable – attached variable. *TPG_default*, time scheduler.

4.2.10 TPG Int

Multistate time scheduler.

This item displays text (schedule name), actual status and a reference to the table with switching events – status changes. The events may be edited in a generic dialogue.

Text – scheduler name, the „#“ sign is substituted by a text according to the actual scheduler state taken over from the IDE. It is not possible to change the status texts here.

Variable – attached variable. *TPG_default*, time scheduler.

4.2.11 TPG double

Analogue time scheduler.

This item displays text (schedule name), actual and a reference to the table with switching events –changes of the value. The events may be edited in a generic dialogue.

Step – increase or decrease of the value on pushing the „arrow up“, „arrow down“ buttons. The limits are fixed, set in the IDE.

Text – scheduler name, the „##.#“ signs are substituted by the value according to the actual scheduler state taken over from the IDE. It is not possible to change the status texts here.

Variable – attached variable. *TPG_default*, time scheduler.

4.3 Save project

The project is saved as a *.lcdmenu* file. The menu tree is compiled and downloaded into the MiniPLC in the Platform Config dialogue in the IDE.

4.4 Save project as

Save project under a different name in the menu *File – Save As*.

5 Alarms, SMS, E-mails

5.1 Touchscreen alarm table

The SoftPLC Touchscreen application provides generic alarm table with overviews of active alarms and alarm history: touch the bell icon at the header of the screen. Its background colour is

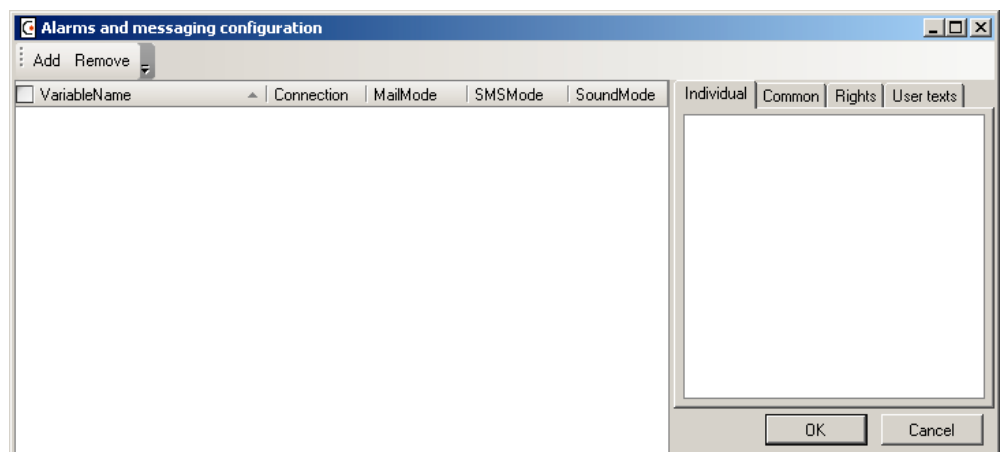
red when there is at least one unacknowledged alarm in the system

blue when there is at least one unreset alarm in the system

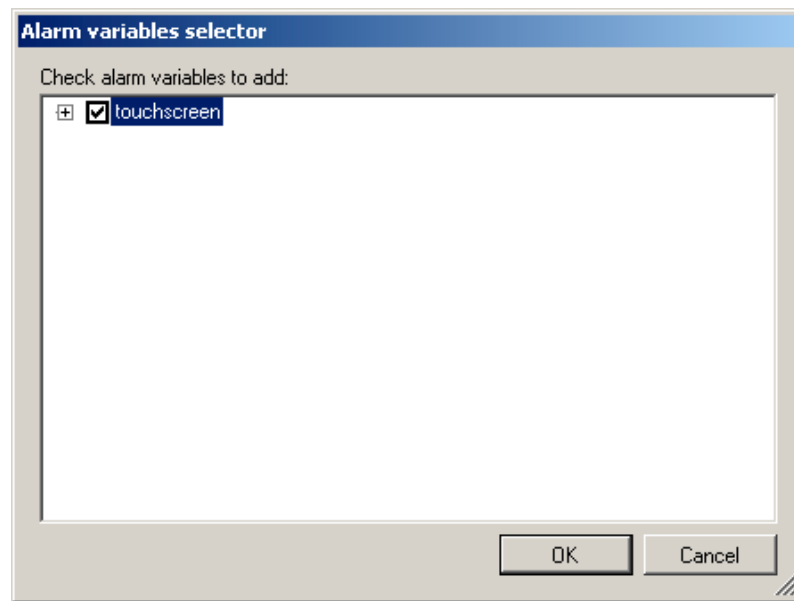
grey when there are no alarms in the system.

In order to display the alarms correctly, the alarm descriptions must be entered in the *Project alarm configuration* menu. The alarms configuration file is created, which contains textual alarm description which displays in the alarms overview table in the touchscreen rather than just the variable name.

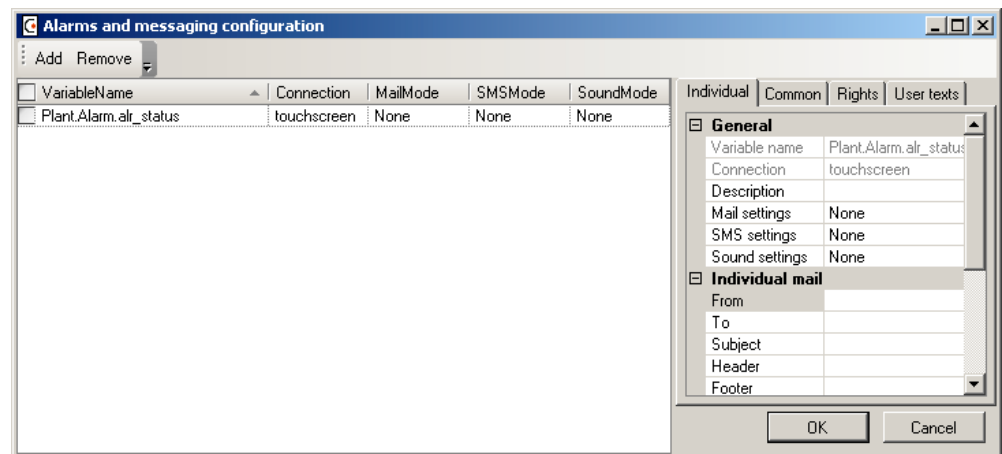
1. Check in the *Project Properties* that a proper path and file name is entered in the *AlarmsConfigFilePath* item, eg. **D:\Projects\MyProject\alarms.config**. The file name and extension is freely definable, however, *alarms.config* is a kind of standard for this file. (The same path and file name must be entered in the Touchscreen configurator at the target SoftPLC process station.)
2. Go to *File – Project alarms configuration*.



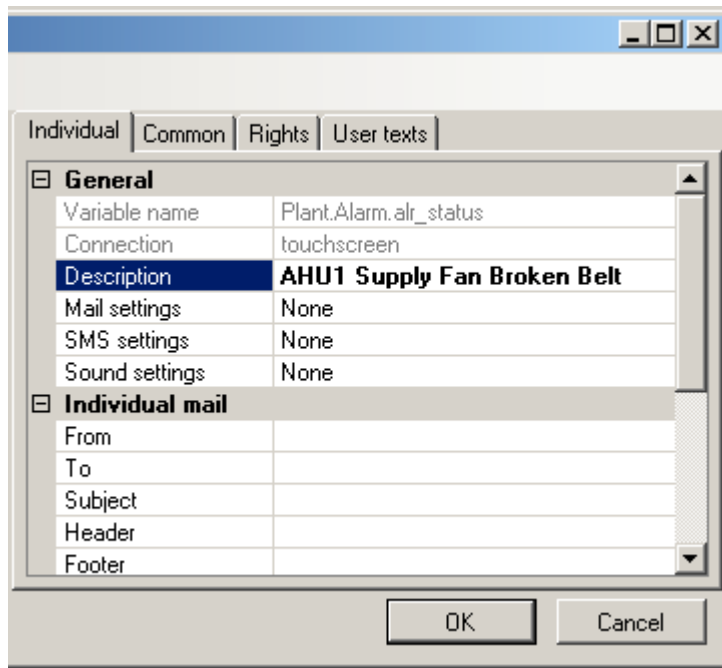
3. Click *Add* and check the connection(s) you want to import alarms from – usually those are all connections defined for your touchscreen project. It is also possible to check selected alarms only (deeper in the tree) in case there are alarms that should not appear in the alarm table. However, the most usual choice is to select all alarms in a project.



4. Click OK. The alarms will be imported into the alarm table.



5. In the *General* tab, enter the *Description* text for each alarm. This is the text to display in the alarm table in the Touchscreen application.



6. Click OK to save the alarms configuration file and return to the Touch Screen Editor.

The above proceedings are enough to configure the Touchscreen alarm table. Make sure that the path and file name to the file is properly defined in the touchscreen configuration at the target device; see 2.8.1, *The Touchscreen Tab*.

5.2 Alarm SMS messages

It is possible to **send and acknowledge alarm messages over SMS** at both SoftPLC (Full, PC-based) platform and MiniPLC. The prerequisites are:

- a free (RS232) serial port on the process station to connect GSM modem
- a functional GSM modem with RS232 serial port (preferably Siemens TC35i, MC35i, Wavecom; for other modems please contact Domat Control System technical support: most probably they will require customized AT commands settings)
- GSM signal availability at the location where the modem is installed (check in advance)
- SIM card configured for sending and receiving SMS messages. The SMS messaging centre (provided by your network operator) must be defined at the SIM card. Usually it is predefined by the operator.

Suitable MiniPLC controllers are those with RS232 port(s), e.g. IPCL300. Remember to install a power socket for the modem power adapter in the panel where the SMS messaging is installed.

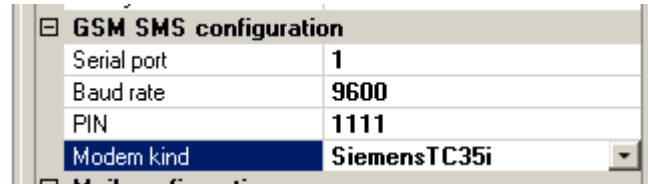
The alarm is set as soon as the corresponding alarm block goes to the active state. In case there should be a switch-on delay, it has to be configured in the PLC logic prior to bringing the signal to the alarm block input. The SMS is send at each rising edge of the alarm input signal; if it is desirable to limit the frequency or amount of the alarm SMS messages, create a corresponding structure in the SoftPLC (e.g. with a Digital Filter, BD15).

It is possible to process alarms from multiple process stations in a network on a Full platform SoftPLC process station. Simply define multiple connections in the touch screen editor.

5.2.1 Modem settings

All parameters concerning SMS messaging are defined in the *File – Project alarms configuration* menu.

1. In the *File – Project alarms configuration* menu, import alarms into the alarm table as described above.
2. Define the GSM modem properties in the *Common* tab, GSM SMS configuration:



Serial port: COM port number of the process station where the GSM modem is connected. For the MiniPLC, it will be 1 or 2.

Baud rate: recommended baud rate is 9600 bps which is also the default baud rate of most modems. Some modems may have autodetecting function. If necessary, check the AT command list of your modem.

PIN: the PIN code to enable access to the SIM card. To avoid commissioning difficulties, it is highly recommended to unblock the PIN in a mobile phone prior to inserting the SIM card into the GSM modem. If done so, this item is not relevant.

Modem kind: type of the modem. Use *General* in case you have a modem which does not fit any of those in the list.

Please check that your modem is in the default factory setting and it responds with text commands (not numerical commands). Using a Windows Hyperterminal or any other terminal application, set the modem as follows:

at&f (Enter) – go to factory settings

atv1 (Enter) – verbose codes (answers like „OK“ rather than „0“)

ate0 (Enter) – no echo (modem does not send back the characters sent from PLC)

at&w (Enter) – write the settings to permanent memory.

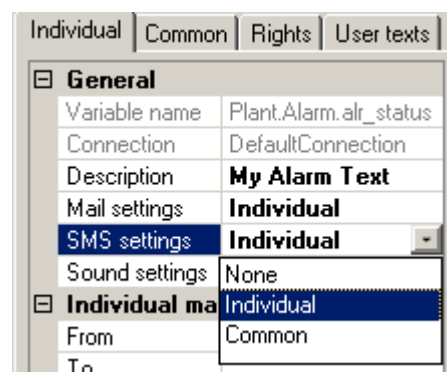
Most of available modems do support those commands.

5.2.2 Alarm SMS definition

Each alarm SMS message is defined either as

Common, or **Individual**.

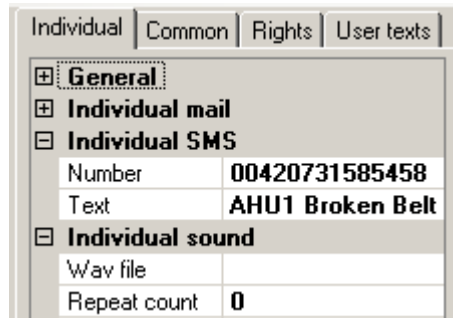
The mode is selected in the Individual alarm settings:



None: No SMS is sent when this alarm occurs.

Common: The SMS is sent using parameters defined in the *Common* settings. All SMS messages using Common settings have the same text and are sent to the same number.

Individual: The SMS is sent using parameters defined in the *Individual settings*. Each SMS will have its own text defined in the *Individual SMS* parameters and may be sent to a separate number.



Individual	
General	
Individual mail	
Individual SMS	
Number	00420731585458
Text	AHU1 Broken Belt
Individual sound	
Wav file	
Repeat count	0

Number: Enter the recipient's number in the international format (00.....).

Text: A short text describing the alarm nature.

Note:

When entering the text to be transmitted over SMS,
– keep the text short as in the SMS there are about 20 characters indicating Message ID, and date and time of alarm occurrence. Total SMS length must not be more than 160 characters
– avoid using national characters and special symbols, such as °. They may not be recognized by some operators. The SMS message then would not be delivered.

The outgoing SMS format is as follows:

MessageID Date Time Text

where

MessageID: a number consisting of hour when the SMS was sent and a current number. Eg. the first SMS sent after 15:00 has a MessageID of 150, the second then 151, etc. The MessageID is used for acknowledging the alarms, see below.

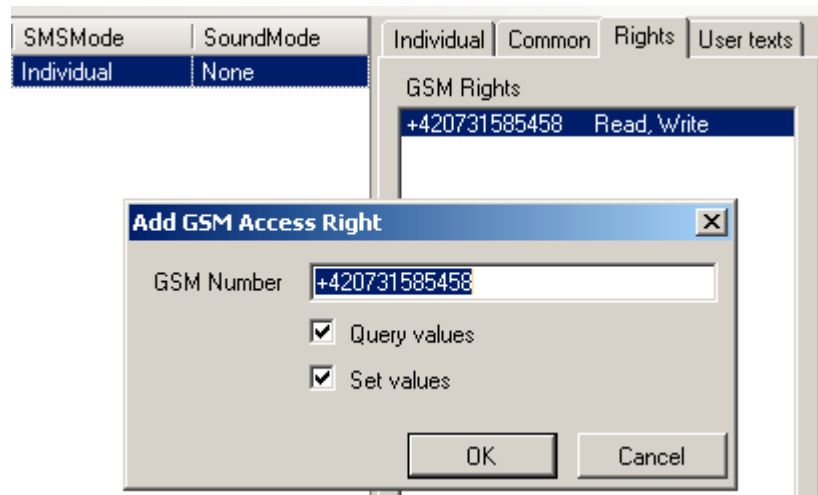
Date: d.m.yyyy; e.g. June 21, 2009 displays as 21.7.2009

Time: hh:mm; e.g. 10:16

Text: the text entered in the Common/Individual SMS – Text field; e.g. AHU1 Broken Belt.

5.2.3 Acknowledging alarms

It is possible to acknowledge one alarm or all alarms in the system. The phone number from which the acknowledge SMS is sent must be entered in the list of authorised users in the *Rights* tab.



Query values: asking for the report SMS through *User text* – see below

Set values: acknowledging alarm, set values and states through *User texts*.

To acknowledge alarm, send an SMS back to the PLC with following text:

a MessageID [MessageID] [MessageID] . . .

to acknowledge one or more alarms, or

aa

to acknowledge all alarms in the system which are defined as SMS alarms (SMS settings is different from *None*, i.e. it is *Individual* or *Common*).

Example 1:

Incoming alarm:

145 20.11.2008 14:31 Chiller B High Pressure

SMS to acknowledge the alarm:

a 145

Example 2:

Incoming alarms:

145 20.11.2008 14:31 Chiller B High Pressure

146 20.11.2008 14:37 Pump M34.2 Overload

SMS to acknowledge those two alarms:

a 145 146

Example 3:

Incoming alarms:

145 20.11.2008 14:31 Chiller B High Pressure

146 20.11.2008 14:37 Pump M34.2 Overload

SMS to acknowledge all alarms in the system, i.e. those two alarms:

aa

Note:

Uppercase and lowercase characters are considered as equal. Therefore e.g. the „Acknowledge all alarms“ message may have any of the following appearances:

aa

AA

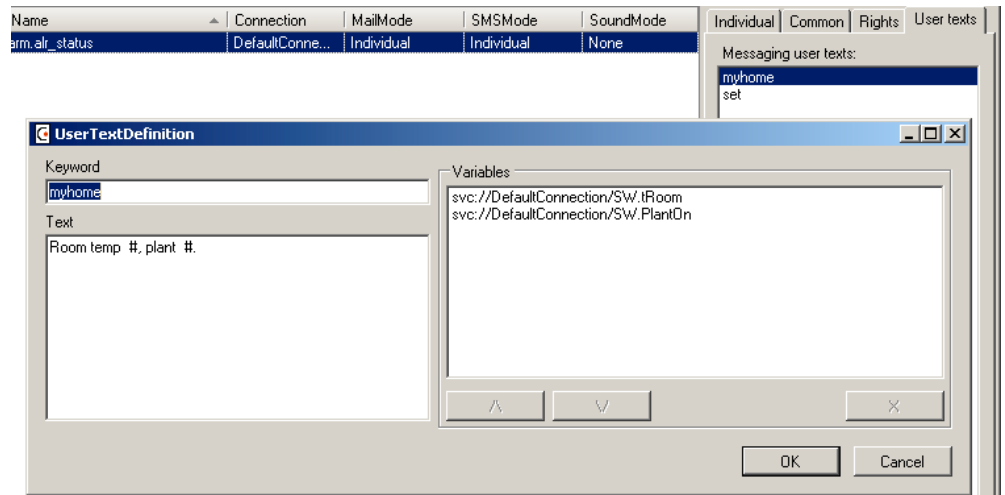
Aa

aA

5.2.4 Querying values over SMS

There may be more predefined sets of values and texts called *User texts* which are identified by a keyword. The keyword shall be easy to type: the user sends the keyword to the PLC and the PLC responds with the corresponding *User text* which contains actual values and states.

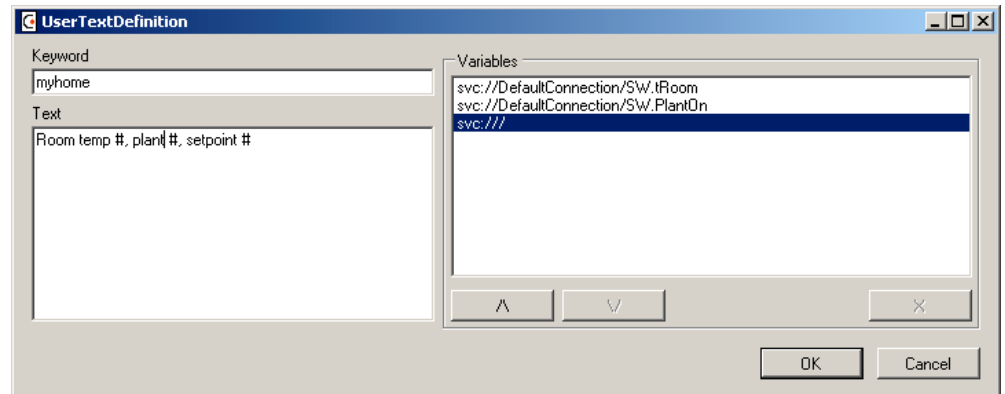
The user texts, or SMS formatting rules, are defined in the *User texts* tab.



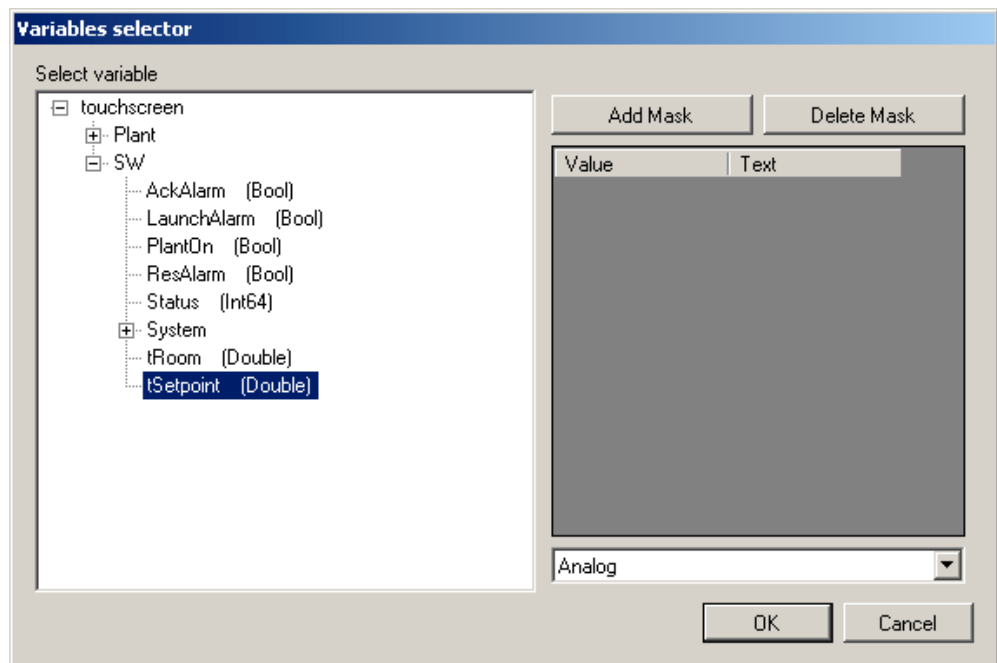
At the example above, there are two user texts defined: their keywords are *myhome* and *set*.

In the *myhome* user text, there are two values, each substituted by the hash (#) sign. The first of them is room temperature, the second is plant status.

As soon as a hash is entered in the *Text* field, a new item in the *Variables* table appears. Doubleclick the item to assign it a variable from the list of imported runtime variables.



At the screenshot above, another item had been added with the text *setpoint*. Doubleclick the empty item in the *Variables* table – the variable selection dialog appears:



Select the variable which shall be reported in the query. At the lower right corner, there is an Analog / Digital selector:

Analog: the variable is represented as a number (e.g. 20 or 0).

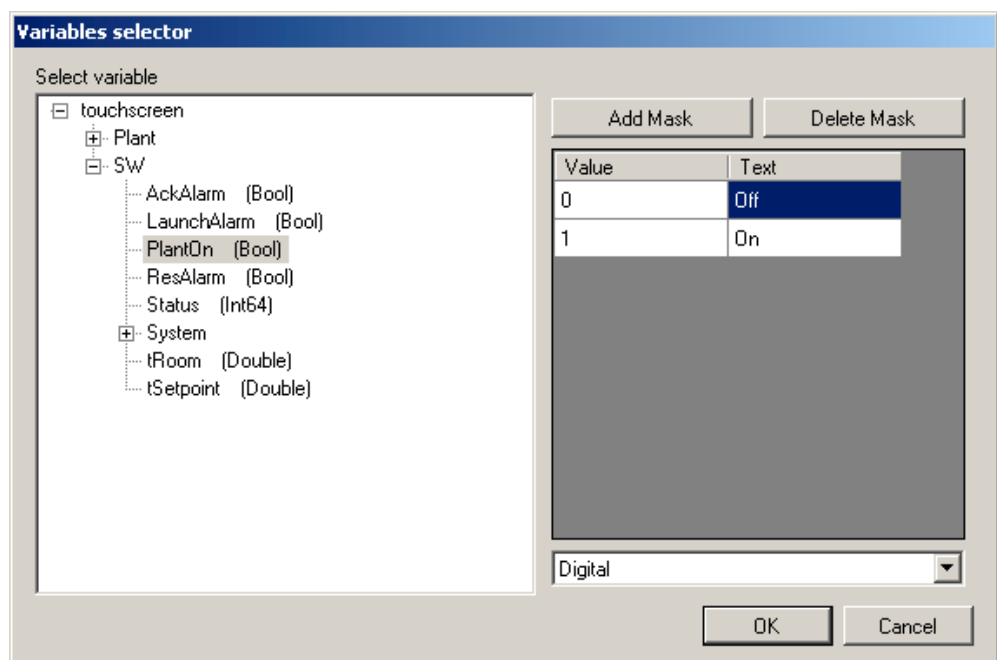
Decimal places for analog values

For analog values, it is possible to define then number of decimal places to be displayed. Simply write **###** instead of **#** to display a value as e.g. **20.45**.

Digital: the variable is binary or multistate and a mask must be added to transform the states into strings (texts), such as Off for 0, and On for 1.

Find a typical mask at the screenshot below. Note that the variable is defined as Digital. If it would be defined as Analog, there would be the variable's value (0, 1, 2, ...) rather than its state text (On, Off, Auto,...) in the response. This is similar to the Touchscreen *Collection* definition; in the SMS definition there is a separate table for masks as users may want the SMS texts to be abbreviated for comfortable typing. Moreover, national and special characters are not allowed in the SMS messages.

The phone number from which the query SMS is sent must be entered in the list of authorised users in the *Rights* tab, see *Acknowledging alarms*.



Again, uppercase and lowercase characters are considered as equal.

Example:

For the configuration above:

SMS query to PLC:

```
Myhome
```

PLC response:

```
Room temp 19, plant off, setpoint 20
```

5.2.5 Setting values over SMS

To set value, just send a query SMS with one or more parameters. The parameters are the new values of the variables, as they follow in the response SMS.

Example:

Keyword: set

Text: plant # setpoint #

Each variable to be changed must be settable (SW variable, setpoint). It is, of course, possible also just to query the setpoints by sending just the keyword without parameters.

SMS to PLC to query (read) values:

```
Set
```

Response from PLC:

```
plant off setpoint 21
```

SMS to PLC to set new plant status (to switch the plant to On):

```
Set on
```

Response from PLC (the PLC confirms with the complete *User text* that the new values have been accepted):

```
plant on setpoint 21
```

It is also possible to set more parameters at a time:

SMS to PLC to set new plant status and setpoint:

```
Set on 23
```

Response from PLC:

```
plant on setpoint 23
```

The parameters must be entered in the same order as they are in the definition text; it is not possible to gap them. If a parameter should not be changed or is not settable at all (measured value), a dummy parameter must be entered instead, e.g. any character.

Example:

For the configuration above:

SMS query to PLC:

```
Myhome x on 23
```

PLC response:

```
Room temp 19, plant on, setpoint 23
```

An „x“ was used as the first parameter, which is measured value (room temperature). Therefore it did not have any influence on its settings and was used only as a placeholder.

When compiling the texts, it makes sense to put the most frequently changed parameters first, and actual (measured) values as the last in the text, so that it is not necessary to use the dummy parameters.

In case the value is not understood, such as **On** at the position where analog setpoint is expected, no changes are made.

Note:

When entering the text to be transmitted over SMS,

- keep the text short as in the SMS there are about 20 characters indicating Message ID, and date and time of alarm occurrence. Total SMS length must not be more than 160 characters
- avoid using national characters and special symbols, such as °. They may not be recognized by some operators. The SMS message then would not be delivered.

5.2.6 Alarm message containing user text

An advanced feature is including a user text into the alarm message. Then it is possible to send an SMS containing both alarm information and a set (or more sets) of actual values defined in the *User texts*.

The user text contents with actual values is inserted into the alarm message by defining the **\$(keyword)** string in the *Individual/Common SMS – Text field*, see Alarm SMS definition.

Example:

In the *User texts*, there is a User text defined:

Keyword: Boilers

Text: Outside # C, Boiler # C, Setpoint # C, DHW # C

In the SMS definition, *Text* field, there is an alarm text:

```
General boiler alarm! $Boilers
```

When there is an alarm, the alarm SMS may look like this:

```
129 16.06.2009 12:55 General boiler alarm! Outside -10 C, Boiler  
92 C, Setpoint 85 C, DHW 48 C
```

It is possible to enter more User texts in one alarm message:

Example:

In the SMS definition, *Text* field, there is an alarm text:

```
High pressure alarm! $Boilers $AHU $Heating1
```

NB. Maximum total SMS length is 160 characters. Longer messages will probably be cut off.

5.2.7 Troubleshooting

If there are no outgoing SMS messages, check the following:

1. SMS message center number (provided by your mobile operator) must be stored on the SIM card
2. the COM port with the GSM modem is not occupied neither by any of the channels, nor by another process – e.g. by a Modbus server definition (delete the Modbus server definition in the Platform config dialog (MiniPLC) or Communication server configuration (SoftPLC Full platform)
3. the target phone number is entered in the international format: 00..... or +.....
4. in the SMS there are no special / national characters (°, ñ, õ etc.)
5. the modem has been set to factory settings and configured according to the description above
6. to diagnose the PLC to modem communication, check if the PLC sends the query commands to the modem (connect the PLC to a terminal). Check with a serial spy cable if the modem responds. Most troubles originate in improperly configured modem.

If the outgoing alarms are working but it is not possible to query / set values:

Check if your phone number is in the list of allowed recipients (the *Rights* tab) and if the *Read*, *Write* options are checked.

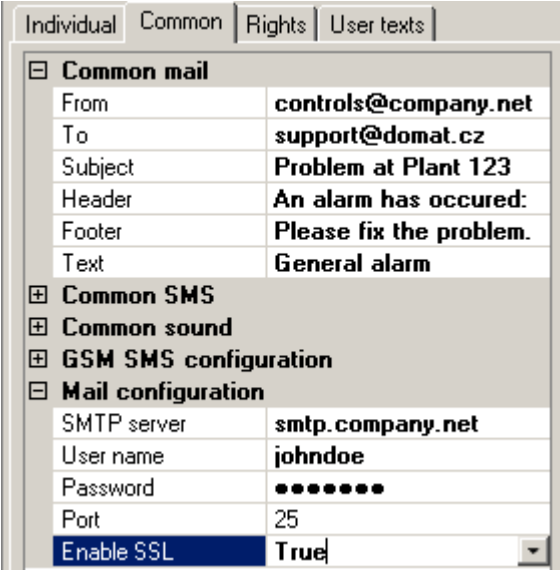
5.3 Alarm e-mails

It is also possible to send alarms by e-mail. The prerequisites are:

- the process station is connected to the Internet
- the network properties of the process station are properly defined, especially Default Gateway and DNS servers
- there is a SMTP server available (with or without SSL) with an account that can be used for alarm messaging.

5.3.1 General e-mail settings

In the *Mail configuration*, set the general e-mail properties:



Individual		Common		Rights		User texts	
Common mail							
From	controls@company.net						
To	support@domat.cz						
Subject	Problem at Plant 123						
Header	An alarm has occurred:						
Footer	Please fix the problem.						
Text	General alarm						
Common SMS							
Common sound							
GSM SMS configuration							
Mail configuration							
SMTP server	smtp.company.net						
User name	johndoe						
Password	●●●●●●●●						
Port	25						
Enable SSL	True						

SMTP server: the server which the PLC uses for sending e-mails. Contact your network administrator to obtain this address.

User name: an e-mail user which is allowed to communicate with the SMTP server. Most servers require user authorisation.

Password: Password for the user to access the SMTP server.

Port: TCP port on the SMTP server which accepts incoming mail.

Enable SSL: select True if the SMTP server uses Secure Socket Layer.

5.3.2 Alarm-specific e-mail settings

Same as with SMS messages, there are *Common* or *Individual* mail settings.

The items are the same both for Common and Individual settings:

From: the address which appears in the From: field of the incoming mail; sender.

To: the address where the mail has to be sent; addressee.

Subject: text which appears in the *Subject:* field of the message

Header, Footer, Text: the body of the mail has following format:

[Header]

[MessageID] [Date] [Time] [Text]

[Footer]

Example

Example of the mail body:

An alarm has occurred:

120 29.7.2009 12:57 General alarm

Please fix the problem.

5.3.3 User texts in e-mails

In the Text field it is possible to use the predefined User texts. The user text contents with actual values is inserted into the e-mail message by inserting the **\$(keyword)** string. Referred to the above examples (see *Querying values*), if the *Common mail – Text* field contains e.g. the string

General alarm \$set

then the e-mail body looks like this:

An alarm has occurred:

120 29.7.2009 12:57 General alarm plant off setpt 20

Please fix the problem.

This feature is often used for automated metering readouts: a MiniPLC sends automatically an e-mail once a day / week / month with values communicated from M-Bus integrated meters. The „alarm“ is then triggered by a time block in SoftPLC, e.g. B99, rather than by a field alarm event.

It is not possible for the PLC to accept incoming e-mails.

5.3.4 Troubleshooting

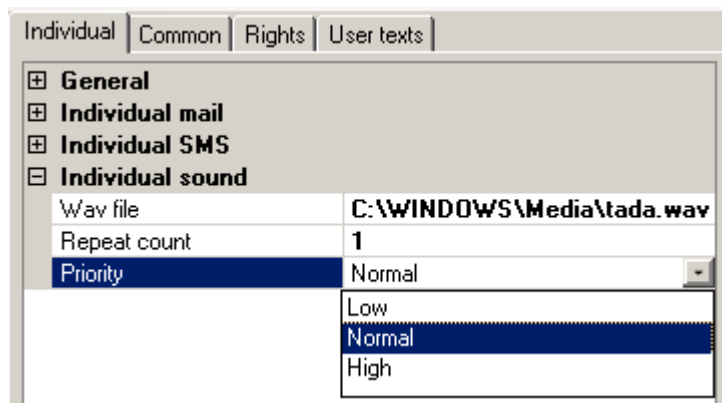
1. Check your internet connectivity.
2. Check the network settings of the process station, especially the default gateway and DNS server addresses.
3. Try to send an e-mail with the same SMTP server / user / password / other settings... from another mail client (e.g. Microsoft Outlook Express). If this fails, note the error message that the client pops up in its error message box.
4. Check if your mail mode (*Common / Individual*) is configured correctly.
5. Try Wireshark or another network analyser to follow the TCP stream and observe the messages which the SMTP server sends back to the process station.

5.4 Alarm sounds

Alarms may notify users by sounds – playing **.wav** files. This is only possible with process stations containing audio output, e.g. PCs. The sounds are played to the default audio output device.

Same as with SMS messages, there are *Common* or *Individual* sound settings.

5.4.1 Alarm sound configuration



Wav file: full path to the sound file which has to be played when alarm occurs.

Repeat count: how many times the sound is repeated at each rising edge of the alarm. To play the sound only once, set the value to 1. If the *Repeat count* is 0, the sound will not be played.

Priority: higher priority alarm sounds are queued ahead of other alarm sounds. This applies in case of more alarms rising at (virtually) the same time.

The playing is stopped as soon as the falling edge of the alarm signal comes; to be sure to play longer messages up to the end, extend the alarm input signal e.g. by the Switch-off delay block (BD13).

The sound is not played with following rising edges of the alarm signal unless the alarm had been reset.

Note

Remember to enable alarm sounds in *Touchscreen configurator*, the *Touchscreen* tab, *Alarm settings*.

5.4.2 Troubleshooting

1. Check if the alarm Sound Mode is configured as *Common* or *Individual* according to your alarming concept.
2. Check if the .wav path and file name is valid.
3. Make sure the *Repeat count* parameter is at least 1.
4. Check the Touchscreen configurator settings, see Note above.
5. Try to launch the .wav file from media player or another application. If this does not work, check your sound hardware (speakers) and driver configuration / installation.