ASiV alarms, trends and diagnostics

TM640

Perfection in Automation

www.br-automation.com

Requirements

Training modules:	TM610 – The Basics of ASiV
Software:	Automation Studio 3.0.80
	Automation Runtime 3.00
Hardware:	[optional] Power Panel 4PP420.1043-75

Table of contents

1. INTRODUCTION

A visualization project also contains components for diagnosing data, process sequences or the status of the machine.

Visual Components allows you to record the machine status or data changes over time, and to display them in the form of event messages or curves.

The Visual Components alarm and trend system can record this data and display it on a screen.

Fig. 1: Visual Components - alarms, trends and diagnostics

With Automation Runtime V3.00 and higher, the **S**ystem **D**iagnostics **M**anager integrated in the operating system can be used to perform web-based diagnostics of the controller. These diagnostics pages can be implemented easily in any Visual Components application.

1.1 Objectives

Selected examples will help you learn how to create pages for evaluating and analyzing data and system states in Visual Components.

You will learn how to...

... Configure the alarm system

... Display messages and alarms

... Configure the trend system

... Display online trend data

... Integrate the System Diagnostics Manager into a visualization

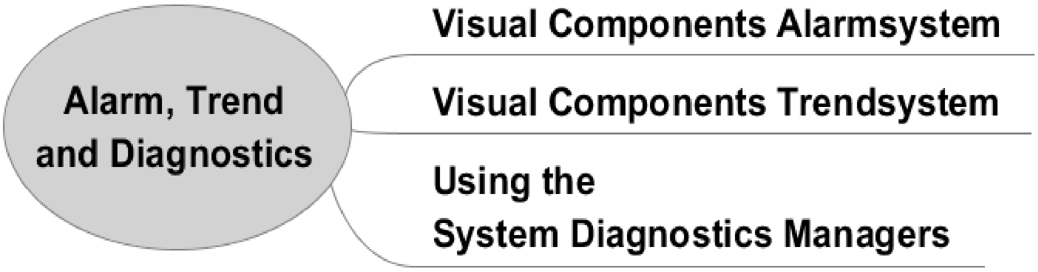

Alarm, Trend and Diagnostics
Visual Components Alarmsystem
Visual Components Trendsystem
Using the System Diagnostics Managers

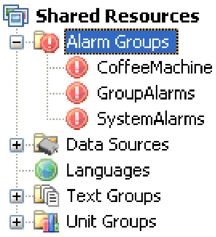
Fig. 2: Overview

2. THE VISUAL COMPONENTS ALARM SYSTEM

Alarms are used to record and respond to certain system states.

System states can be divided into in messages, warnings, and alarms by dividing them into alarm groups.

The alarm system is a resource used by all visualizations in a project, and is configured under the "**Alarm Groups**" node.



Shared Resources
- Alarm Groups
  - CoffeeMachine
  - GroupAlarms
  - SystemAlarms
- Data Sources
- Languages
- Text Groups
- Unit Groups

Features of the alarm system:

- Active alarms and historical alarms displayed differently
- Manage alarm texts and their configuration in alarm groups
- Color-coded identification of alarm priorities
- Use touch screen or keys to scroll through alarm lists, acknowledge and bypass alarm states
- Manage all alarms in the system in the alarm overview
- Language switching for all alarm system display modes
- API functions for processing alarm data (TM670)

2.1 Alarm system terminology

This chapter describes the terminology used in the alarm system.

Alarm texts show an event in the form of a message, warning or alarm. These events, referred to below as alarms, are displayed when the corresponding process variable switches from 0 to 1.

- **Messages**
  are non-critical system or machine messages. Acknowledgement is not required.
- **Warnings**
  are error states that do not require any action from the operator, but must be acknowledged.
- **Alarms**
  are critical error states in the system that require action by the user and must be acknowledged.

Alarm groups are containers for events of the same type, priority and appearance. Alarm states are formed in the application program with a freely definable array with the type "Bit".

Acknowledge means that the user must confirm an alarm that has occurred. This action can be evaluated in the application program, so that e.g. a bell linked with the alarm state can be switched off.

Bypassing means that the alarm can be shut off intentionally by the user regardless of the alarm state. Bypassed alarms can be monitored in the application program.

Latched alarms are given the date and time of the first time they occur. If the alarm is acknowledged (or bypassed) and then triggered again, the date and time is not updated.

System alarms indicate errors and status information from the Visual Components runtime environment. These include, for example, the switch on message and the battery status.

Story involving an alarm system

In 1994 an explosion and subsequent fires at the Texaco refinery in Milford Haven, Wales, injured twenty-six people and caused material damages of around £48 million and considerable production downtime.

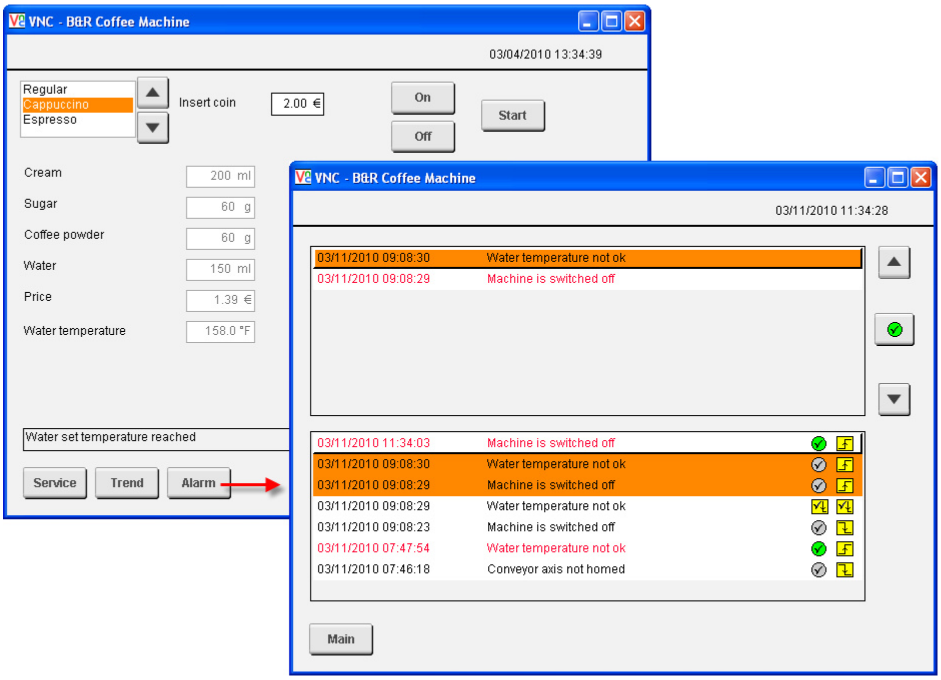
In the course of investigations by the British Health and Safety Executive (HSE) the following factors were found to have contributed to the catastrophe:

- There were too many alarms, and they were poorly prioritized.
- The displays in the control room didn't help the operators understand what was happening.

In the 11 minutes leading up to the explosion, the two operators were faced with 274 alarms that had to be detected, acknowledged and responded to appropriately.

2.2 Task

The task in this training module is based on the visualization application created in TM610.



03/04/2010 13:34:39

Regular
Cappuccino
Espresso

Insert coin 2.00 €

Cream 200 ml
Sugar 60 g
Coffee powder 60 g
Water 150 ml
Price 1.39 €
Water temperature 158.0 °F

Water set temperature reached

03/11/2010 11:34:28

```
03/11/2010 09:08:30    Water temperature not ok
03/11/2010 09:08:29    Machine is switched off
```

```
03/11/2010 11:34:03    Machine is switched off
03/11/2010 09:08:30    Water temperature not ok
03/11/2010 09:08:29    Machine is switched off
03/11/2010 09:08:29    Water temperature not ok
03/11/2010 09:08:23    Machine is switched off
03/11/2010 07:47:54    Water temperature not ok
03/11/2010 07:46:18    Conveyor axis not homed
```

Fig. 3: "AlarmSystem“ sample program

In this sample, the application program already generates alarm states, which are configured in the "CoffeeMachine" alarm group.

The following tasks will be performed in this training manual:

- Create a new alarm page
- Understand the "CoffeeMachine" alarm group
- Display active alarms
- Display historical alarms
- Acknowledging alarms
- Display bit and value alarms

> The procedure for creating a new page for displaying alarms and for changing from the main page to this new page are described in the TM610 training manual.

Task: Create a new page called "AlarmPage"

Under the "**Pages**" node, create a new page and name it "**AlarmPage**". Then connect it to the global image layer "**globalArea**".

The "**Alarm**" button on the "**MainPage**" takes you to the alarm page using the "**ChangePage**" key action.

To get from the alarm page back to the main page, a button must be created there to change the page back to "**MainPage**".

Result

Pressing the "**Alarm**" button opens the alarm page. From there it is also possible to change back to the "**MainPage**".

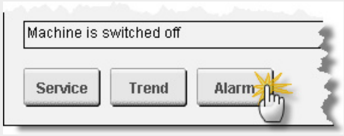

Machine is switched off

Fig. 4: Changing to the "Alarm" page

Before we begin configuring the alarm system, let's consider the following issues with respect to the handling of an alarm system:

The experiences of users and safety experts raise the following questions:

- What can be done to prevent floods of alarms?
- Are false alarms triggered frequently?
- Are alarms "acknowledged" without being responded to accordingly?
- Is the operator's response clearly defined and easy to trace?
- Are alarm texts informative and easy to understand?
- How can help be provided for alarm texts?
- How easy is it to navigate the alarm pages?

2.3 Configuring the alarm system

In Visual Components, alarms are managed and configured under the "**AlarmGroups**" node.

A new alarm group can be inserted by pressing the <**Insert**> button, or by selecting <**Add Alarm Group**> from the shortcut menu.

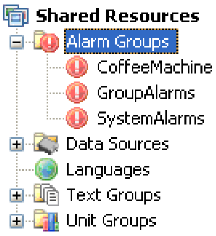


Fig. 5: Alarm groups

Visual Components Help:
Shared resources / Alarm groups / Alarm groups

Events of the same type (alarm, warning, message) and the same priority are grouped in alarm groups. As a result, all the alarms in a group have the same appearance during runtime.

The alarm system contains two predefined alarm groups, which contain the group alarms and system alarms.

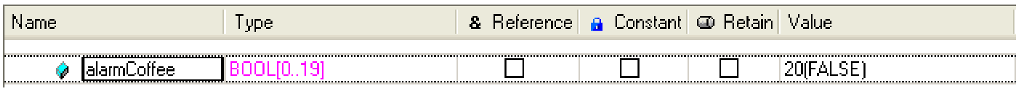
> The group alarms and system alarms cannot be deleted. Only the alarm texts can be changed and translated when using language switching.

Visual Components Help:
Shared resources / Alarm groups / Alarm groups / Predefined alarm groups

2.3.1 Management of alarms in the user task

All alarms that belong to a group must be arranged into a BOOL array in the user task.

In the sample, the application program maps the "CoffeeMachine" alarms to the local process variable "**visAlarm:alarmCoffee**".



Name	Type	Reference	Constant	Retain	Value
alarmCoffee	BOOL[0..19]	□	□	□	20(FALSE)

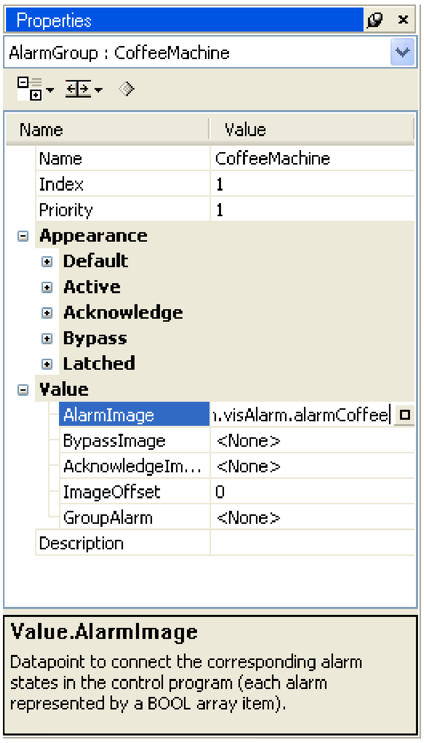
Fig. 6: Alarm states mapped to process variable "alarmCoffee"

2.3.2 "CoffeeMachine" alarm group

The process variable used to form the alarms of an alarm group is connected to the property "**Value / AlarmImage**".

This data point must be connected, but the other data points are optional.

The properties of the "**Appearance**" group can be used to define the colors that indicate the various alarm states in this group.



Properties

AlarmGroup : CoffeeMachine

Name	Value
Name	CoffeeMachine
Index	1
Priority	1
Appearance	
Default	
Active	
Acknowledge	
Bypass	
Latched	
Value	
AlarmImage	ı.visAlarm.alarmCoffee
BypassImage	<None>
AcknowledgeIm...	<None>
ImageOffset	0
GroupAlarm	<None>
Description	

Value.AlarmImage

Datapoint to connect the corresponding alarm states in the control program (each alarm represented by a BOOL array item).

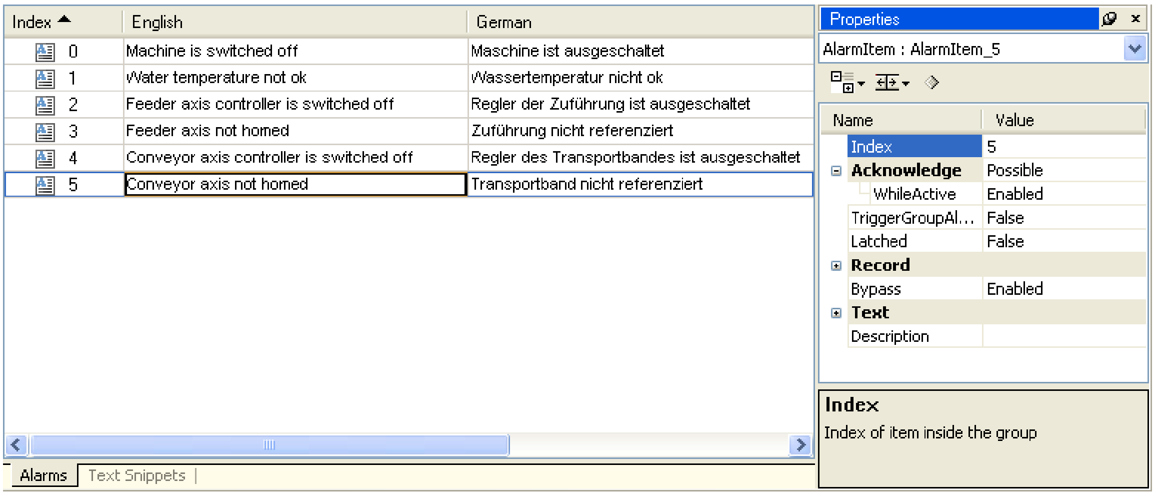
Fig. 7: Connecting the alarm image

Visual Components Help:
Shared resources / Alarm groups / Alarm groups / Configuring alarm groups

2.3.3 Alarms in the "CoffeeMachine" alarm group

The **alarms** in an **alarm group** are created like the texts in a text group.

The properties of an alarm text determine its behavior during runtime.



Index	English	German
0	Machine is switched off	Maschine ist ausgeschaltet
1	Water temperature not ok	Wassertemperatur nicht ok
2	Feeder axis controller is switched off	Regler der Zuführung ist ausgeschaltet
3	Feeder axis not homed	Zuführung nicht referenziert
4	Conveyor axis controller is switched off	Regler des Transportbandes ist ausgeschaltet
5	Conveyor axis not homed	Transportband nicht referenziert

Properties

AlarmItem : AlarmItem_5

Name	Value
Index	5
Acknowledge	Possible
WhileActive	Enabled
TriggerGroupAl...	False
Latched	False
Record	
Bypass	Enabled
Text	
Description	

Index
Index of item inside the group

Fig. 8: Alarms in the "CoffeeMachine" alarm group

The "**Acknowledge**" property defines whether or not an alarm requires acknowledgement.

- **Acknowledge**
  Defines whether this alarm needs to be acknowledged when it occurs during runtime (warning, alarm) or not (message). If an alarm requires acknowledgement, it must be acknowledged by the user during runtime so that it can be removed from display once the alarm state becomes "**inactive**" again.
- **WhileActive**
  This property specifies whether an acknowledgeable alarm is allowed to be acknowledged while it is still pending.

Visual Components Help:
Shared resources / Alarm groups / Alarms / Configuring alarms

The alarm bit field connected to the alarm group must be at least as large as the maximum index of the alarm text.

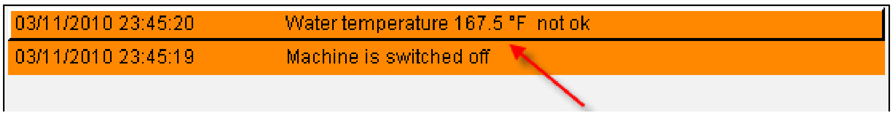
2.3.4 Static and dynamic alarm text

The most basic alarm text contains only text.

"**TextSnippets**" can be used to display a value in an alarm text.

This allows you to display the cause of an alarm when it occurs (e.g. Overtemperature at 140.00°C).

This type of alarm text will be used later on (see 2.7 Bit and value alarms).



03/11/2010 23:45:20 Water temperature 167.5 °F not ok

03/11/2010 23:45:19 Machine is switched off

Fig. 9: Displaying an alarm with a value

Visual Components Help:
Shared resources / Alarm groups / Alarms / Text snippets

Facts about the alarm system

When a process is disrupted, the cause of the disruption can trigger a large number of related alarm states.

Such a flood of alarms puts the operator under a huge amount of stress. To prevent this, the number of alarms generated after a disturbance should be minimized.

The number of alarms can be reduced by filtering alarms (messages) during normal processing using different priorities.

2.4 Viewing alarms during runtime

Alarms can be displayed in different lists during runtime according to the requirements.

An alarm field is created by dragging the "**Alarm**" control to the desired page.

2.4.1 "Current" alarm list

The "**Current**" alarm list shows all **active alarms** and **inactive alarms that are awaiting acknowledgement**. This view is used the most frequently.

This image is compiled from the respective alarm bits while starting up the visualization.

This view is selected from the "**Value** / **Source**" property.

Task: Insert alarm control to display the "Current" alarm list

On the "**AlarmPage**", have the "**Current**" alarm list appear in the top half of the page.

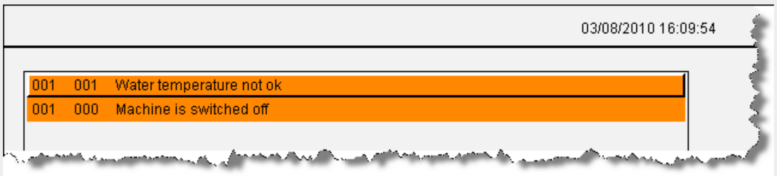
Draw the alarm list in the top half of the screen. Leave about 50 pixels free to the right of the alarm list.

Procedure:

- Insert an "**Alarm**" control on the "**AlarmPage**"
- Adjust the size
- Change the "**Value** / **Source**" property

Result

All pending alarms are displayed in the alarm list. Once the CoffeeMachine is switched on and the water reaches the defined temperature, alarms are automatically removed from this list (acknowledgement not required).



03/08/2010 16:09:54

```
001   001   Water temperature not ok
001   000   Machine is switched off
```

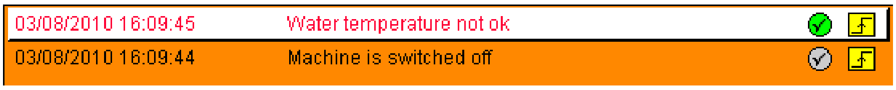
Fig. 10: "Current" alarm list

Section 2.5 "Interaction with the alarm list" describes how to use the alarm list.

When an alarm list is inserted it has a simplified column arrangement by default.

<Number of alarm group> <Index of alarm text> <Alarm text>

Simply configuring the arrangement of columns allows you to design the alarm list according to you individual needs.



```
03/08/2010 16:09:45   Water temperature not ok
03/08/2010 16:09:44   Machine is switched off
```

Fig. 11: Adjust the columns of the alarm list

2.4.2 Configure the columns of the alarm list

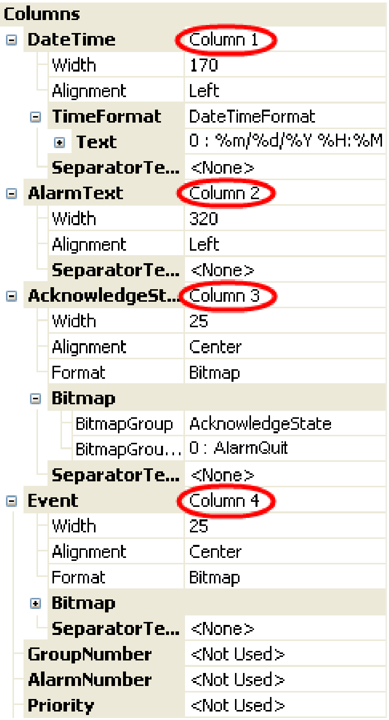
The **"Columns"** properties define the order of the alarm information in the corresponding columns and how they are displayed.

Selecting one of the columns "**Column <n>**" assigns the event to the respective position.

The width of each column [pixels] can be adjusted using the "**Width**" property.

Avaliable alarm information:

- Alarm status
- Time with time format
- Alarm text
- Alarm event
- Group number
- Alarm number
- Priority
- Acknowledgement status
- Bypass status



Columns	
DateTime	Column 1
Width	170
Alignment	Left
TimeFormat	DateTimeFormat
Text	0 : %m/%d/%Y %H:%M
SeparatorTe...	<None>
AlarmText	Column 2
Width	320
Alignment	Left
SeparatorTe...	<None>
AcknowledgeSt...	Column 3
Width	25
Alignment	Center
Format	Bitmap
Bitmap	
BitmapGroup	AcknowledgeState
BitmapGrou...	0 : AlarmQuit
SeparatorTe...	<None>
Event	Column 4
Width	25
Alignment	Center
Format	Bitmap
Bitmap	
SeparatorTe...	<None>
GroupNumber	<Not Used>
AlarmNumber	<Not Used>
Priority	<Not Used>

Fig. 12: Column configuration

Task: Define the columns for the "Current" alarm list

Your task is to configure the columns of the "Current" alarm list. Remove the columns for the group and alarm numbers.

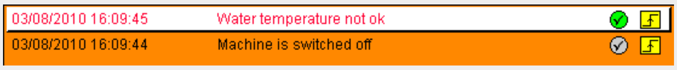
- Column 1: **Time**
- Column 2: **Alarm text**
- Column 3: Acknowledgement status as bitmap
- Column 4: Alarm event as bitmap

Visual Components Help: Control reference / Alarm / Columns

Result

An event can be displayed in any form.

Using the bitmap groups for the alarm system that are contained in the default project makes it easy to display graphic information in the alarm line.



03/08/2010 16:09:45	Water temperature not ok	✓
03/08/2010 16:09:44	Machine is switched off	✓

Fig. 13: Displaying the "Current" alarm list

The colors used for the alarm texts to display the different alarm states can be defined in the properties of the alarm group.

Each status can be assigned a different foreground and background color.

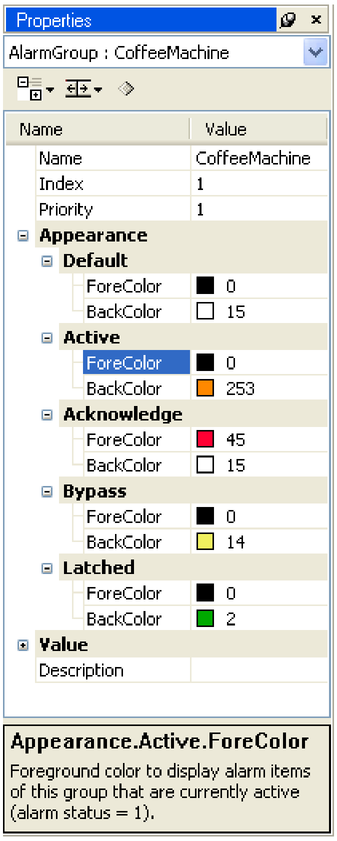
Default: Appearance of an unacknowledged inactive alarm.

Active: Appearance of an active alarm (Alarm = 1).

Acknowledge: Appearance of an acknowledged alarm.

Bypass: Appearance of a bypassed alarm.

Latched: Display of the latched alarm.



AlarmGroup : CoffeeMachine

Name	Value
Name	CoffeeMachine
Index	1
Priority	1
Appearance	
Default	
ForeColor	■ 0
BackColor	□ 15
Active	
ForeColor	■ 0
BackColor	🟧 253
Acknowledge	
ForeColor	🟥 45
BackColor	□ 15
Bypass	
ForeColor	■ 0
BackColor	🟨 14
Latched	
ForeColor	■ 0
BackColor	🟩 2
Value	
Description	

Appearance.Active.ForeColor
Foreground color to display alarm items of this group that are currently active (alarm status = 1).

Fig. 14: Color design

As can be seen in the image above, the first alarm has been acknowledged, and the second one is still active.

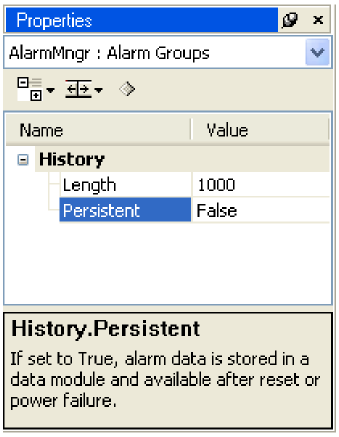
The acknowledge bitmap is also displayed accordingly. The alarm event bitmap indicates that the alarm is active.

2.4.3 "History" alarm list

The "**History**" alarm list records all changes to the alarm status and user actions that occur from the time the visualization application is started in a configurable ring buffer.

This image is compiled from the historical alarm list while starting up the visualization.

The historical alarm list can be configured in the global properties of the "**Alarm Groups**" node.



Name	Value
History	
Length	1000
Persistent	False

History.Persistent
If set to True, alarm data is stored in a data module and available after reset or power failure.

Fig. 15: Configuring the alarm history

Visual Components Help:
Shared resources / Alarm groups / Alarm history

This view is selected from the "**Value / Source**" property.

Task: Insert alarm control to display the "History" alarm list

On the "**AlarmPage**", have the "**History**" alarm list appear in the bottom half of the page.

Draw the alarm list in the bottom half of the page. Leave about 50 pixels free to the right of the alarm list.

Set up the columns so they are the same as the "**Current**" alarm list.

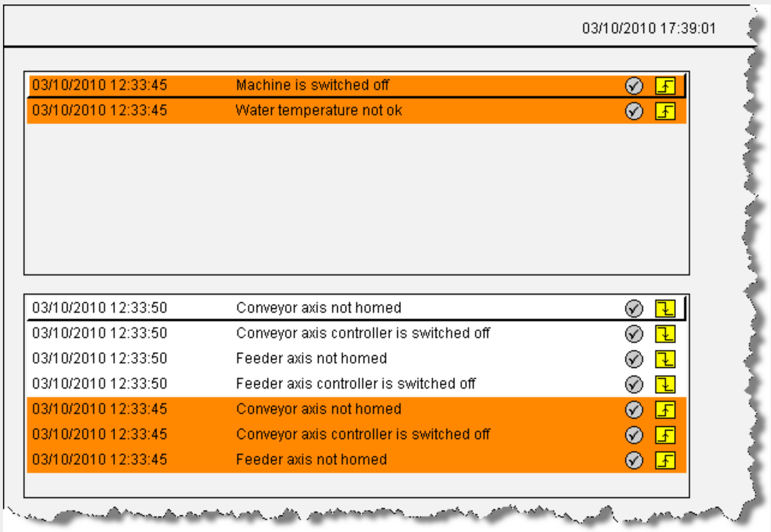
Procedure:

- Insert an "**Alarm**" control on the "**AlarmPage**"
- Adjust the size and **align** with the "**Current**" alarm list
- Change the "**Value** / **Source**" property
- Change the "**Columns**" properties

Use the Automation Studio **Copy** / **Paste** function to copy the "**Current**" alarm list. All that needs to be changed for this task is the "**Value** / **Source**" property.

Result

The pending alarms are displayed in the top alarm list. The bottom alarm list shows all alarm status changes that have occurred since the alarm server was started by the Visual Components runtime system.



03/10/2010 17:39:01

03/10/2010 12:33:45	Machine is switched off
03/10/2010 12:33:45	Water temperature not ok

03/10/2010 12:33:50	Conveyor axis not homed
03/10/2010 12:33:50	Conveyor axis controller is switched off
03/10/2010 12:33:50	Feeder axis not homed
03/10/2010 12:33:50	Feeder axis controller is switched off
03/10/2010 12:33:45	Conveyor axis not homed
03/10/2010 12:33:45	Conveyor axis controller is switched off
03/10/2010 12:33:45	Feeder axis not homed

Fig. 16: "History" alarm list

If the alarm text doesn't fit on one line, it will be displayed on multiple lines.
To fit the text on one line, you can adjust the width of the "**Alarm text**" column.

2.5 Interaction with the alarm system

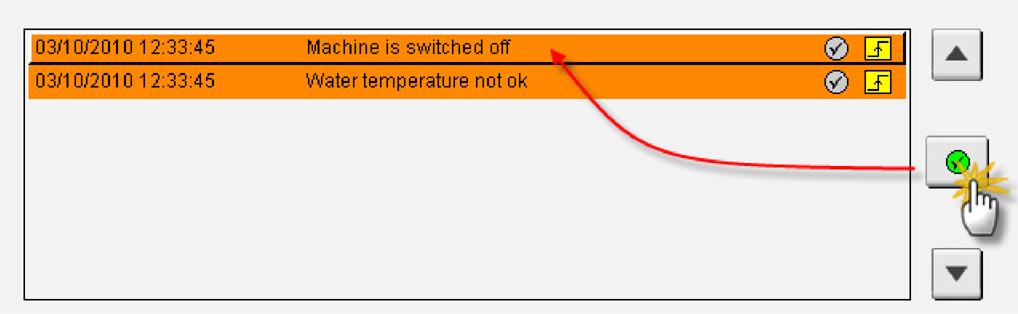
In the previous exercises, we have focused on simply displaying the alarms.

Alarm !
What do I do ?

The alarm system can be operated using a configurable touchpad, buttons or the keyboard.

You can interact with the alarm system in the following ways:

- Scroll in the alarm list
- Acknowledge an alarm that requires acknowledgement
- Acknowledge all alarms
- Bypass an alarm



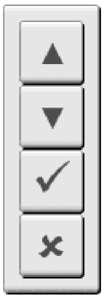
03/10/2010 12:33:45 Machine is switched off

03/10/2010 12:33:45 Water temperature not ok

Fig. 17: Example of acknowledging an alarm

In our example, buttons placed next to the "**Current**" alarm list are used to interact with it.

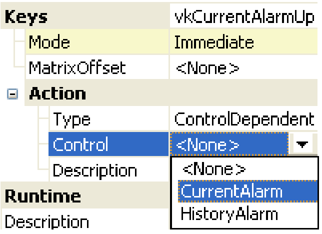
The integrated touchpad is used to interact with the "**History**" alarm list.



2.5.1 Interaction using buttons / keys

In order to be able to operate the alarm list, you must set the property "**Input = True**".

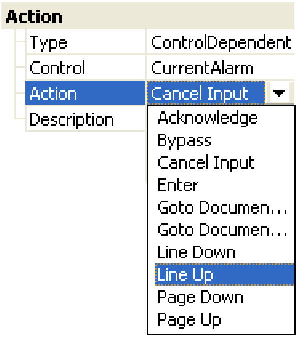
Key actions that operate another control must have the type "**ControlDependent**". For this type of key action, the "**Control**" property is set to the control that the action itself is connected.



Keys	vkCurrentAlarmUp
Mode	Immediate
MatrixOffset	<None>
Action	
Type	ControlDependent
Control	<None>
Description	
Runtime	
Description	

Fig. 18: Selecting the alarm list

After the alarm list is selected, the "**Action**" property appears, showing the possible actions for the control.



Action	
Type	ControlDependent
Control	CurrentAlarm
Action	Cancel Input
Description	

Fig. 19: Alarm list actions

Task: Draw three buttons for the alarm system scroll up, scroll down and acknowledge

Draw three buttons to the right of the "**Current**" alarm list.

- Scroll up ("**Line Up**" action)
- Scroll down ("**Line Down**" action)
- Acknowledge selected alarm ("**Acknowledge**" action)

For the acknowledge button, the "**AlarmQuit**" bitmap should be shown instead of a text.

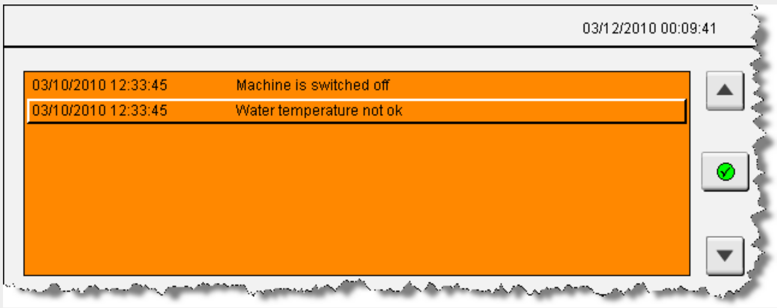
The "**ControlDependent**" task action only affects controls with the property setting "**Input = True**".

Procedure:

- For the current alarm list, set the "**Input**" property to "**True**"
- Insert three buttons as shown in the template for scrolling and acknowledging
- Create a new virtual task for each button
- Select the key action "**ControlDependent**" for each button
- Select the control for the "**Current**" alarm list
- Configure the respective actions

Result

When the "**Current**" alarm list contains more than one alarm, it is possible to scroll through the list.



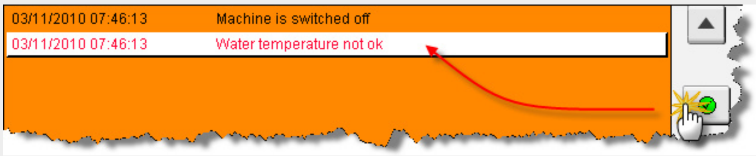
03/12/2010 00:09:41

03/10/2010 12:33:45 Machine is switched off

03/10/2010 12:33:45 Water temperature not ok

Fig. 21: Scroll in the "Current" alarm list

Pressing the acknowledge button acknowledges the selected alarm. An acknowledged alarm is displayed with the colors defined for the "**Appearance**" property.



03/11/2010 07:46:13 Machine is switched off

03/11/2010 07:46:13 Water temperature not ok

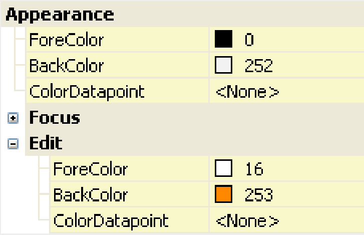
Fig. 21: Acknowledge the selected alarm

Visual Components Help:
Key action reference / ControlDependent
Control reference / Alarm / Input

Buttons are used to adapt operation of the alarm lists to a GUI (Graphical User Interface) template.

Another possibility is to use predesigned touchpads, which allow a wide range of interface designs.

When a key action is performed on the alarm list, it changes to "**Edit**" mode. This sets the pages colors to the colors defined for this mode.



Appearance	
ForeColor	■ 0
BackColor	□ 252
ColorDatapoint	<None>
⊞ **Focus**	
⊟ **Edit**	
ForeColor	□ 16
BackColor	🟧 253
ColorDatapoint	<None>

Fig. 22: Page properties

Facts about the alarm system

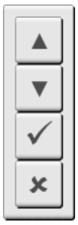
The many configuration options available for the alarm list allow it to be adapted to the requirements of a particular situation.

Messages that don't require any action from the operator should not be displayed in the list of active alarms.

The challenge of ensuring efficient alarm management begins with the planning and design of the user applications.

An alarm system changes over the course of a machine's life cycle, and should be subject to continuous improvement along the way.

2.5.2 Interaction using the touchpad



An alarm list can also be operated using the default touchpad contained in the project, or a one you create yourself.

In this example, the touchpad named "**NavigationPad_ver**" is used to operate the historical alarm list.

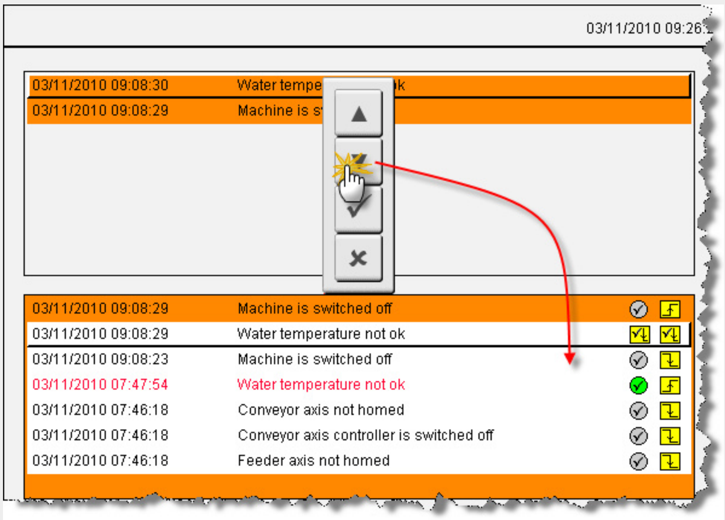
Task: Configure the historical alarm list for operation using a touchpad

The historical alarm list is operated by the "**NavigationPad_ver**" touchpad.

This touchpad can be selected from the "**Input / Touchpad**" property.

Result

The touchpad opens automatically when the historical alarm list is selected. The touchpad allows you to scroll up and down.



03/11/2010 09:26:

03/11/2010 09:08:30 Water tempe k
03/11/2010 09:08:29 Machine is s

03/11/2010 09:08:29 Machine is switched off
03/11/2010 09:08:29 Water temperature not ok
03/11/2010 09:08:23 Machine is switched off
03/11/2010 07:47:54 Water temperature not ok
03/11/2010 07:46:18 Conveyor axis not homed
03/11/2010 07:46:18 Conveyor axis controller is switched off
03/11/2010 07:46:18 Feeder axis not homed

Fig. 23: Scrolling up in the historic alarm list

The only action possible in the historical alarm list is scrolling. It is not possible to acknowledge or bypass an alarm in this view.

Facts about the alarm system

Efficient operation of an alarm system permits a maximum of 10 alarms per minute.

Control room operators are often confronted with up to 2000 messages and alarms per day. Filtering and prioritizing messages and alarms helps ensure that only the most important alarms are displayed.

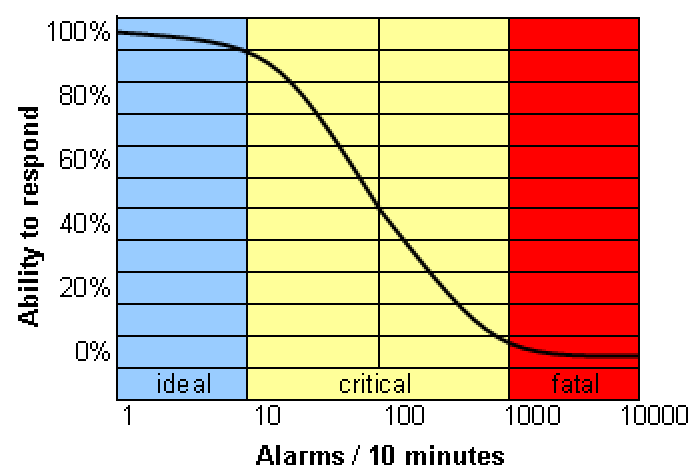


Fig. 24: Ability to react to alarms

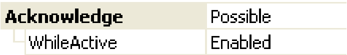
Alarm group and priority filters can be set in the alarm list using the "**Value** / **Filter**" property.

Visual Components Help:
Control reference / Alarm / Value / Filter
Shared resources / Alarm groups / Alarm groups / Configuration / Priority

2.6 Acknowledging alarms

An alarm normally requires an action from the operator.

This action can be "**forced**" using the alarm's acknowledgement settings. This means that the alarm remains in the alarm list - even after the event that triggered it is no longer active - until the system operator confirms (acknowledges) it.

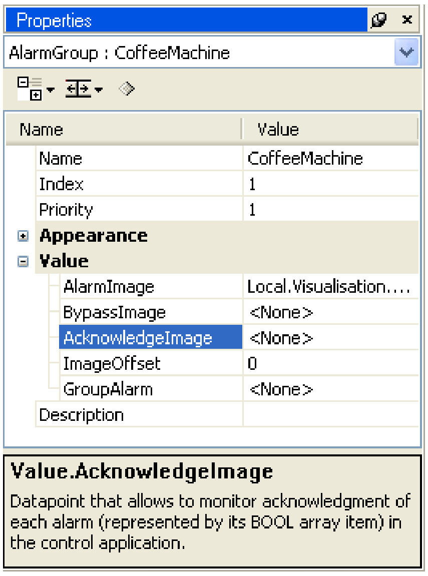


Acknowledge	Possible
WhileActive	Enabled

Fig. 25: Defining acknowledgement behavior

The optional acknowledgement image can be used to evaluate the operator's response in the application program.

An **acknowledgement image** is connected to the "**Value / AcknowledgeImage**" property of an alarm group as a bit array.



Properties

AlarmGroup : CoffeeMachine

Name	Value
Name	CoffeeMachine
Index	1
Priority	1
Appearance	
Value	
AlarmImage	Local.Visualisation....
BypassImage	<None>
AcknowledgeImage	<None>
ImageOffset	0
GroupAlarm	<None>
Description	

Value.AcknowledgeImage
Datapoint that allows to monitor acknowledgment of each alarm (represented by its BOOL array item) in the control application.

Fig. 26: Acknowledgement image for an alarm group

The combination of the acknowledgement behavior and the acknowledgement image results in the following runtime behavior:

Acknowledge	Behavior	Image
Disabled	Acknowledgement not possible or not required (e.g. message)	No effect on the acknowledgement image.
Possible	Acknowledgement is only made possible by the "WhileActive" property, so it is only possible when the alarm is active.	Alarm = Acknowledge bit set in image. The acknowledge bit is cleared when the alarm is acknowledged or no longer active.
Required	Acknowledgement is absolutely required. An inactive alarm remains in the alarm list until it is acknowledged.	Alarm = Acknowledge bit set in image. The bit is only cleared when the alarm is acknowledged.

Task: Change the acknowledgement behavior of an alarm

In the "**CoffeeMachine**" alarm group, change the acknowledgement image for the alarm with **Index [0]**, so that the alarm text remains in the alarm list until the user acknowledges the alarm independent of the alarm status.

Procedure:

- Open the "**CoffeeMachine**" alarm group
- Change the acknowledgement behavior for the alarm text with Index [0]

Result

When the visualization application is started, the "**CoffeeMachine**" is still off. This causes an alarm to be entered in the "**Current**" alarm list.

If the acknowledgement behavior is not changed, the alarm text would be automatically removed from the list when the CoffeeMachine is turned on. Changing the acknowledgement behavior keeps the entry in the list until the user acknowledges it.

Visual Components Help:
Shared resources / Alarm groups / Alarms / Configuring alarms / Acknowledge

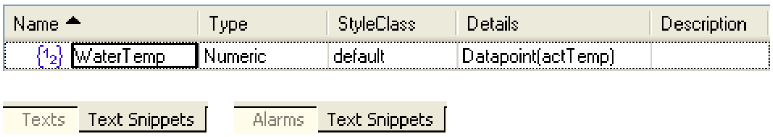
2.7 Bit and value alarms

An event or alarm is generated in the Visual Components alarm system by setting a bit in a bit array.

If an alarm should be triggered when an upper or lower limit is exceeded, the alarm line can display the corresponding value of a variable in addition to the bit event.

An alarm text is made up of a static text and an optional dynamic text and/or value. The dynamic parts of the alarm text are created as "**TextSnippets**" and inserted anywhere in the alarm text.

TextSnippets can be inserted in the alarm text editor under the "**TextSnippets**" tab.



Name ▲	Type	StyleClass	Details	Description
WaterTemp	Numeric	default	Datapoint(actTemp)	

Fig. 27: Inserting TextSnippets

In this example, an alarm is generated when the water temperature drops below a certain value. In the following task, you will add the temperature limit to the alarm text.

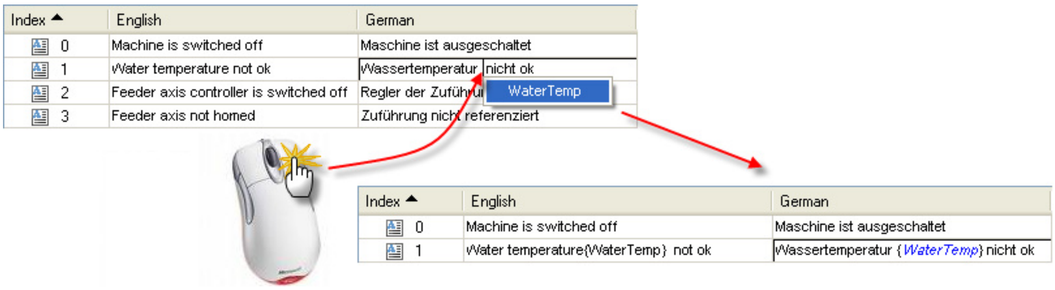
Task: Display the temperature limit in the alarm text

In the "**CoffeeMachine**" alarm group, edit the alarm text with **Index [1]** so that it displays the temperature from the process variable "**gHeating.status.actTemp**".

Instead of the text "**Water temperature not OK**", the temperature value at the time of the alarm should be displayed.

Procedure:

- Insert a new TextSnippet with the name "**WaterTemp**" and the type "**Numeric**"
- Connect the process variable "**gHeating.status.actTemp**" to the "Datapoint" property
- Under the "Alarms" tab, add the TextSnippet to the respective alarm text using the shortcut menu (check edit mode of the text)



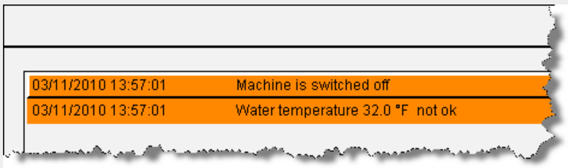
Index ▲	English	German
0	Machine is switched off	Maschine ist ausgeschaltet
1	Water temperature not ok	Wassertemperatur nicht ok
2	Feeder axis controller is switched off	Regler der Zuführu
3	Feeder axis not homed	Zuführung nicht referenziert

Index ▲	English	German
0	Machine is switched off	Maschine ist ausgeschaltet
1	Water temperature{WaterTemp} not ok	Wassertemperatur {*WaterTemp*} nicht ok

Fig. 28: Insert TextSnippet at the desired position in the alarm text

Result

When the visualization is started, an alarm is generated showing the current water temperature. This alarm starts by displaying 0°C or 32°F (initial value), and shows the limit value when the alarm is turned off.



```
03/11/2010 13:57:01    Machine is switched off
03/11/2010 13:57:01    Water temperature 32.0 °F  not ok
```

Fig. 29: Alarm with limit value

The "**Persistent**" property of the TextSnippet "freezes" the value when the alarm is entered in the historical alarm list.

Visual Components Help:
Shared resources / Alarm groups / Alarms / Text snippets

3. THE VISUAL COMPONENTS TREND SYSTEM

The trend system is a tool for displaying a two dimensional graph of one or more values over time – similar to an oscilloscope.

The trend system displays a graph in a "**Trend**" control. The horizontal x-axis (abscissa) represents time, and the process values are displayed on the vertical y-axis (ordinate).

The trend system is used to observe not only the size of a process value, but more importantly how it changes over time.

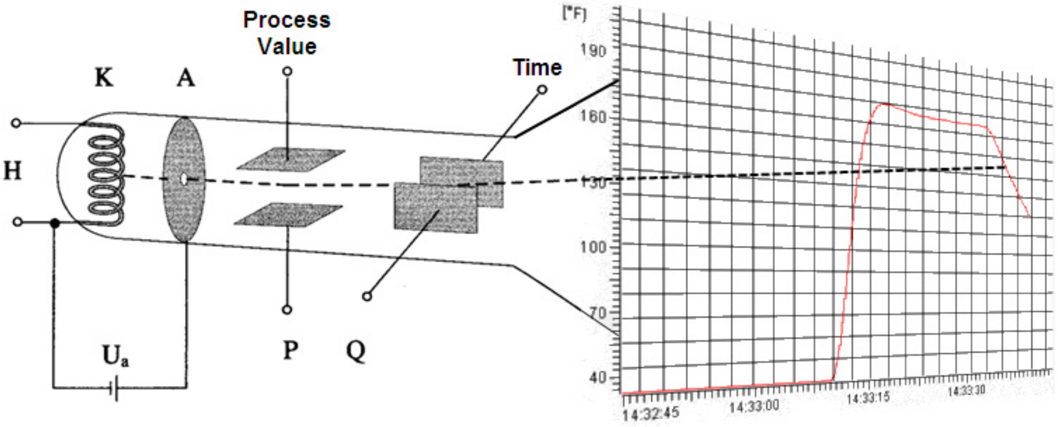


Fig. 30: Virtual oscilloscope - Trend system

Possibilities of the trend system:

- Display process values over time
- Display values recorded by the application program
- View up to 16 graphs with color coding
- Up to two measurement cursors to view values
- Flexible configuration of value and time axes
- Zoom and scroll through the trend data
- Show and hide curves

3.1 Trend system terminology

This chapter describes the terminology used in the trend system.

A **trend control** is used to display a trend window (trend container) on a page

Trend data defines which process variables should be recorded and displayed. There are two types of trend data:

- **Online trend**
  Process variable(s) are recorded at a defined sample rate for a defined interval (number of samples).
- **User trend**
  Displays data from a defined number of process values in an array from the application program.

Trend curves show the trend data in the trend container.

Trend scales are used to label and graduate the time and value axes in the trend container.

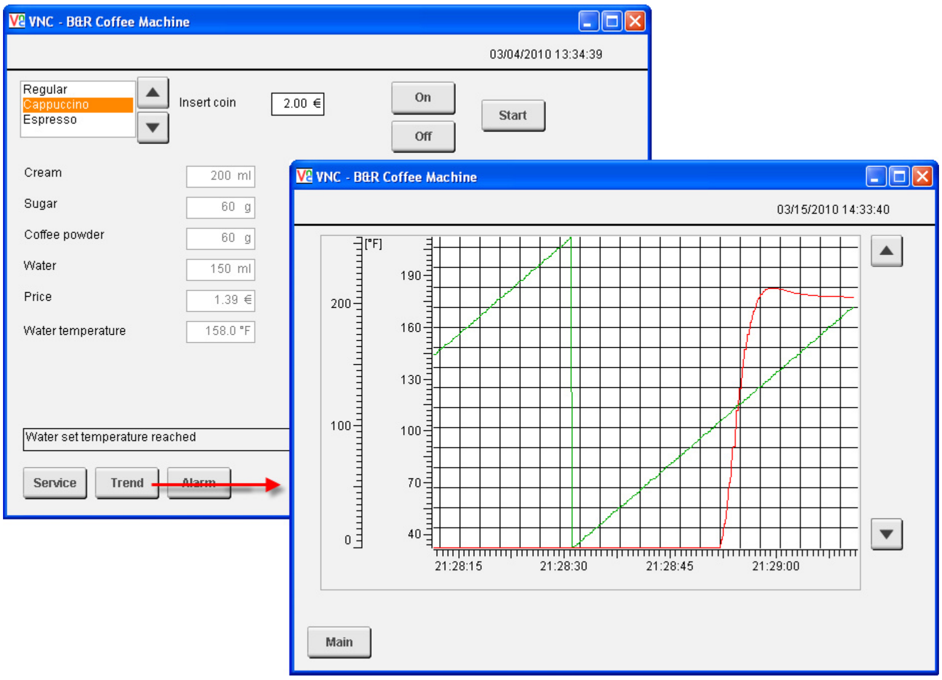
Trend containers define the appearance of the trend control, and include one or more trend curves, the grid in which the curves are displayed and the appearance of the value and time axes.

- Up to 16 curves
- One time axis with variable scaling and labeling
- Multiple value axes with different scaling and labeling functions

> The trend system is composed of various controls and configurations, which can be created using the **Trend Wizard**.

3.2 Task

The task in this training module is based on the visualization application created in TM610.


VNC - B&R Coffee Machine

03/04/2010 13:34:39

Insert coin 2.00 €

Cream 200 ml

Sugar 60 g

Coffee powder 60 g

Water 150 ml

Price 1.39 €

Water temperature 158.0 °F

Water set temperature reached

VNC - B&R Coffee Machine

03/15/2010 14:33:40

Fig. 31: "Trend system" sample program

In this sample program, a function block in the application program controls the water temperature after the CoffeeMachine is turned on.

The closed-loop control depends on the type of coffee, and is started after the CoffeeMachine is switched on.

The following tasks will be performed in this training manual:

- Create a new trend page
- Develop a trend page with the "**Trend Wizard**"
- Display a temperature curve in the trend system
- Add a second curve with a different scale
- Interact with the trend system

The procedure for creating a new page for displaying a trend and for changing from the main page to this new page are described in the TM610 training manual.

Task: Create a new page called "TrendPage"

Under the "**Pages**" node, create a new page and name it "**TrendPage**". Then connect it to the global image layer "**globalArea**".

The "**Trend**" button on the "**MainPage**" takes you to the alarm page using the "**ChangePage**" key action.

To get from the trend page back to the main page, a button must be created there to change the page back to "**MainPage**".

Result

Pressing the "**Trend**" button opens the trend page. From there it is also possible to change back to the "MainPage".

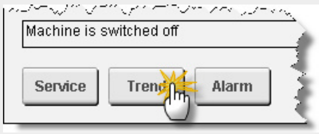


Fig. 32: Changing to the "Trend" page

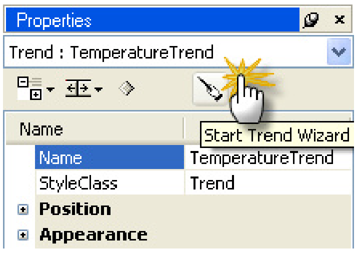
3.3 Configuring the trend system

There are two ways to configure the trend system:

- Express configuration using the Trend Wizard
- Manual configuration of trend objects (user-defined)

The "**Trend Wizard**" is used to create a new trend curve.

The Trend Wizard can be opened by selecting the Wizard icon in the property editor toolbar for the "**Trend**" control.

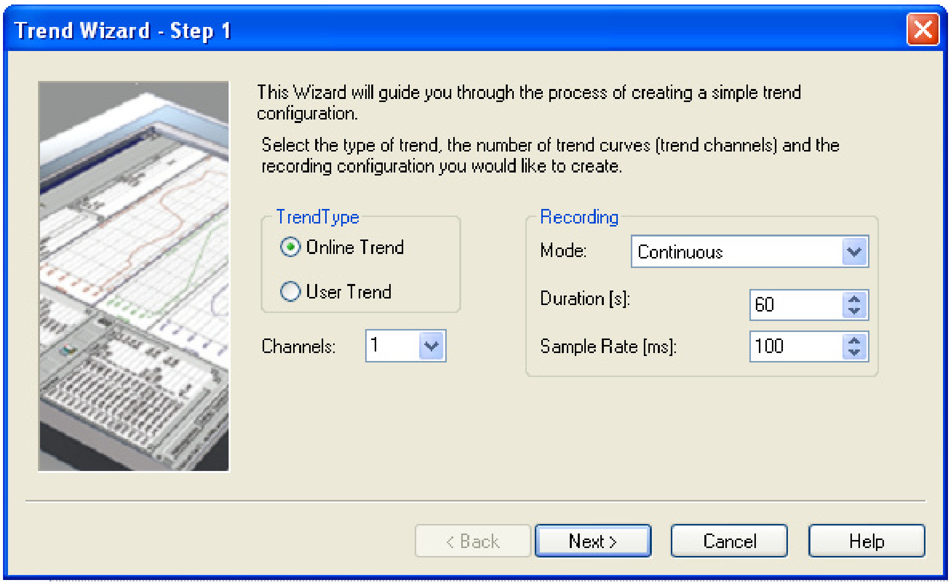

Properties

Trend : TemperatureTrend

Start Trend Wizard

Name	
Name	TemperatureTrend
StyleClass	Trend
Position	
Appearance	

Visual Components Help:
Visualization resources / Trends
Control reference / Trend / Trend Wizard


Trend Wizard - Step 1

This Wizard will guide you through the process of creating a simple trend configuration.

Select the type of trend, the number of trend curves (trend channels) and the recording configuration you would like to create.

TrendType
● Online Trend
○ User Trend

Channels: 1

Recording
Mode: Continuous
Duration [s]: 60
Sample Rate [ms]: 100

Fig. 33: Creating a trend configuration using the "Trend Wizard"

3.4 Displaying an online trend

In this example, the change in water temperature over time after the **CoffeeMachine** is turned on is displayed in the trend control as an "**online trend**".

Task: Create a trend configuration for displaying the change in water temperature over time

Insert a "**Trend**" control on the "**TrendPage**" as shown in the tasks.

Use the "**Trend Wizard**" to configure a trend with the type "**Online**", which records the process variable "**gHeating.status.actTemp**" for one minute, with a sample rate of 100 milliseconds.

Configuration in the pages of the Trend Wizard:

Page 1:

• Trend Type:	**Online Trend**
• Channels:	**1**
• Recording:	**Continuous**
• Duration [s]:	**60**
• Sample Rate [ms]:	**100**

Page 2:

• Value data point:	**gHeating.status.actTemp**
• ForeColor	**red**
• MinValue (>= AS 3.0.81)	**0**
• MaxValue (>= AS 3.0.81)	**100**

Page 3: No changes necessary

Procedure:

- Insert a "**Trend**" control on the "**TrendPage**"
- Start the Trend Wizard from the properties editor
- Make the configurations on the three pages of the Trend Wizard

Result:

A trend configuration is inserted on the page.

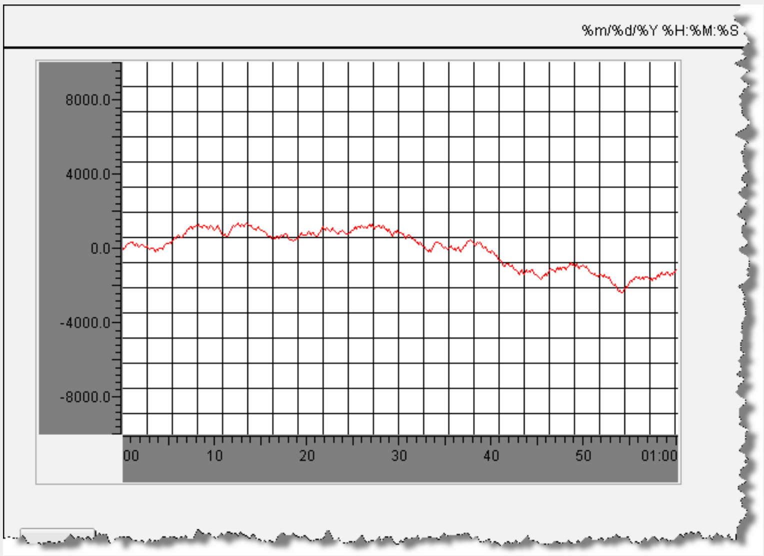


Fig. 34: Result after finishing the "Trend Wizard"

In order for the trend to be displayed correctly during runtime, the value range for the y-axis must be configured manually.

With Automation Studio 3.0.81 and higher, this setting can be made in the Trend Wizard.

Facts about the trend system

The quality of a trend system is not determined by how many colors are used or how many curves are displayed.

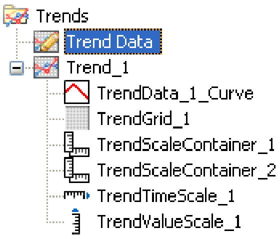
To optimize the analysis of measurement data, a maximum of 2 to 4 curves should be shown at a time.

Each line color should have a corresponding description in a legend next to or below the graph.

3.4.1 Components created using the Trend Wizard

The entries and configurations you make in the Trend Wizard create various trend components.

These components can be edited later on, or they can be created or edited manually.



```
Trends
  Trend Data
  Trend_1
    TrendData_1_Curve
    TrendGrid_1
    TrendScaleContainer_1
    TrendScaleContainer_2
    TrendTimeScale_1
    TrendValueScale_1
```

Fig. 35: Trend components

The components created using the Trend Wizard are inserted in the project explorer under the "**Trends**" node.

Trend Data includes all process variables that are recorded by the trend server.

- Process variables and duration of recording
- Type of trend - "**Online**" or "**User**"
- Minimum and maximum values to be displayed
- Optional data points for controlling the recording

One or more entries from this trend data are connected to a curve in the trend container.

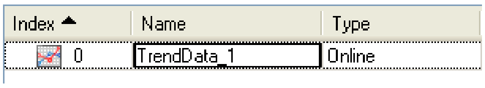
Trend_1 contains all the objects for displaying a trend curve (trend container)

- Curve(s) connected to a trend data value
- Scaling of x and y axes
- Trend container grid

The trend container is connected to the trend control to display the curves and scales it contains.

3.4.2 Trend data

Double clicking on "**Trend Data**" opens the trend data editor.



Index ▲	Name	Type
0	TrendData_1	Online

Fig. 36: Trend data generated by the Trend Wizard

In this example, the temperature value displayed on the y-axis should be limited to values between 0 and 100 degrees Celsius.

Task: Configure value limits for y-axis

In the trend data of the entry created by the Trend Wizard, set the minimum and maximum limits for the "**Value**" property to 0 and 100 degrees Celsius.

Then, rename the entry from its default name to "**WaterTemperature**".

Facts about Visual Components

All objects a user creates in a visualization project are given a default name, which is derived from the base object.

If multiple objects of the same type are inserted, a consecutive number is added to the end of the default name.

It is therefore recommended to give each object in a visualization project a descriptive name so that it can be easily identified.

Result

The limits are applied immediately to the "**Trend**" control on the "**TrendPage**".

This ensures that the temperature curve will be displayed correctly during runtime.

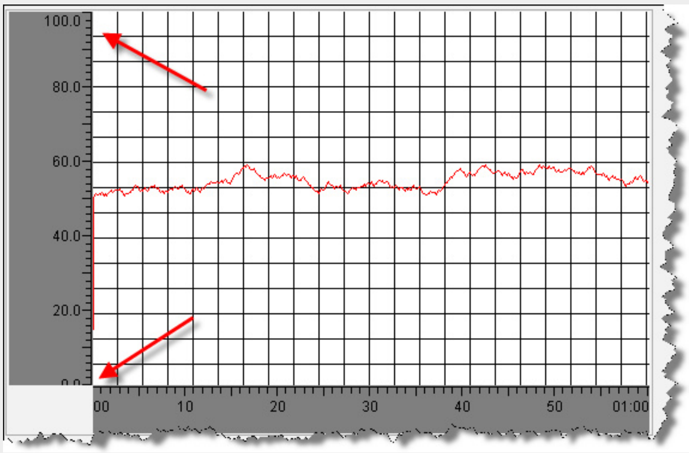


Fig. 37: Correct configuration of limits

This change ensures the basic functionality of the temperature displayed in the "**Trend**" control. The following steps will configure the design of the trend container.

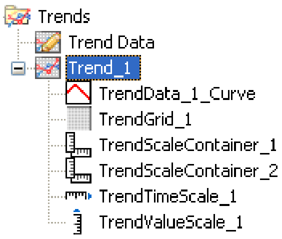
Visual Components Help:
Visualization resources / Trends / Configuring trend data

3.4.3 Trend container

The trend container contains the elements that are needed to display a trend curve.

A trend container can be compared to a process image - controls used to display the trend curves are placed here and configured using their properties.

A trend container is edited on the node of the trend object. The controls of the trend object are listed under this node..



- Trends
  - Trend Data
  - Trend_1
    - TrendData_1_Curve
    - TrendGrid_1
    - TrendScaleContainer_1
    - TrendScaleContainer_2
    - TrendTimeScale_1
    - TrendValueScale_1

Fig. 39: Trend container

Double clicking on a trend object opens an editor where it can be configured.

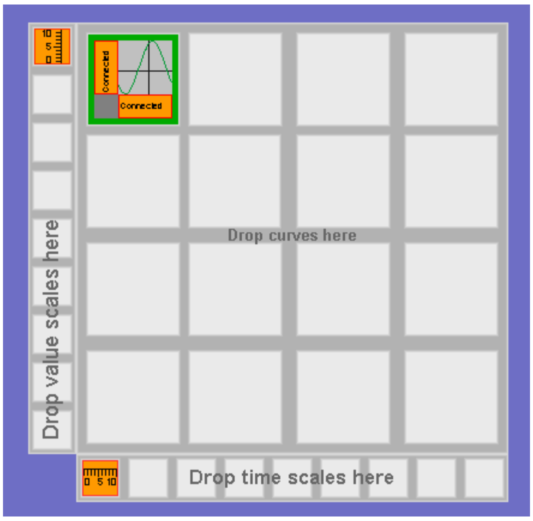


Fig. 38: Trend object editor = Trend container

Up to 16 trend curves can be inserted in this trend container. Under the trend curves there is a container for the time axis, and to the left of the trend curves there is a container for the value axis.

The properties of an object can be configured in the properties editor when the object is selected.

In the next step, the label axes for the two scales will be configured.

Task: Change the background color of the x and y axes to match the design of the page

The default background color for the label axes is dark gray. To make the axes match the background color of the process image, select a light gray with color index 252.

Make this change for the "**TrendValueScale**" and "**TrendTimeScale**" axes.

Procedure:

- Open the trend container by double clicking on the trend object
- Select the "TrendValueScale" and "TrendTimeScale" objects
- Change the **Backcolor** property in the appearance properties

The changes can be checked in the trend container's preview (Layout tab).

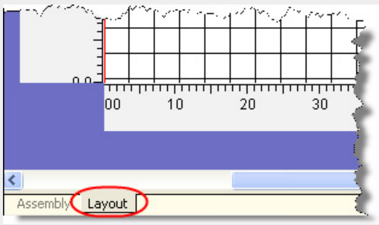


Fig. 40: Trend container preview

Visual Components Help:
Visualization resources / Trends / Configuring trend objects
Visualization resources / Trends / Editing trend objects

Result

The trend axes in the trend control now have the same background color as the process image.

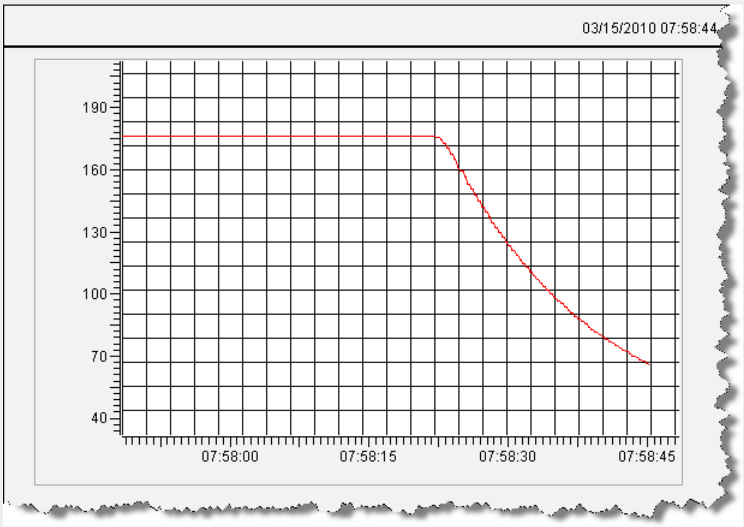


Fig. 41: Background color of the scale axes has been changed

To conclude this task, we will add the temperature units (°C or °F) to the value axis.

Task: Display the unit text on the value axis

In the properties of the "**TrendValueScale**" control, configure the unit text (short text).

Procedure:

- For the "**TrendValueScale**" control, change the "**Format / UnitText**" property.

Result

The unit text is displayed in the top left edge of the value axis according to the language setting (process variable "**gMainLogic.status.curLanguage**").

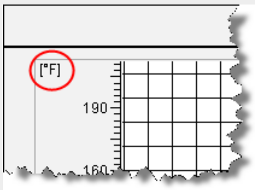


Fig. 42: Display units text

Facts about the trend system

The process variables to be recorded are added to the **trend data**.

The **trend object** is a container for the trend curves and the scale axes. Values from the trend data are connected to a trend curve.

The trend object itself is connected to the **trend control**.

The **Trend Wizard** is used to perform the basic configuration of these controls. The appearance of the trend object is configured using the properties of the controls.

3.4.4 Displaying a second curve

So far the trend control only shows one curve.

In the next example we will add a second curve, whose process variable generates a sawtooth curve. The objects for the trend configuration will be created manually to provide a closer look at the relationships between them.

The code changes needed for this task are made int he program "**visCtrl**".

Task: Create a sawtooth curve in the program "visCtrl"

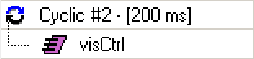
Insert the following line in the cyclic part of the Structured Text program "**visCtrl.st**":

```
trendJigSaw := trendJigSaw + 1;
```

The "**trendJigSaw**" process variable must have the data type "**USINT**".

Result:

The "**visCtrl**" program runs in a 200ms task class. The value of the process variable is increased by 1 every cycle. Once the process variable reaches the value 255 it is reset to 0 in the next cycle.



Cyclic #2 - [200 ms]
visCtrl

If a program is edited while the Visual Components editor is open and a new process variable is inserted in the program, the data points in the "**Data Sources / Local**" node must be refreshed by pressing the <**F5**> key.

From this point on, the new process variable can then be used in the visualization application.

In order for the process variable to be recorded by the trend server, it must be added to the trend data ("**TrendData**" node).

Task: Add the process variable to the trend data

Under the "**Trend Data**" node, insert a new object and name it "**JigSaw**".

- The duration of the recording will be 60 seconds with a sample rate of 100 ms.
- Insert the "**trendJigSaw**" process variable for the "**Value**" property.

Procedure:

- Open the editor for the "**Trend Data**" node
- Insert a new trend object with the name "**JigSaw**"
- The type of the trend object must be "**Online**", which is recorded continuously.
- The duration of the recording is determined by the "**SampleRate**" and "**SampleCount**" settings.
- Connect the process variable "**trendJigSaw**"

The new trend object must be connected to a trend curve in the trend container where the curve is displayed.

When the trend container is opened, the controls that can be inserted in the trend container are shown in the toolbar.

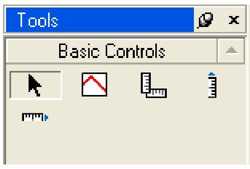


Fig. 43: Trend control

A process variable should always be sampled twice as often as a value change could possibly occur.

To insert a new curve in the trend container, drag a "**TrendCurve**" control to an open space.

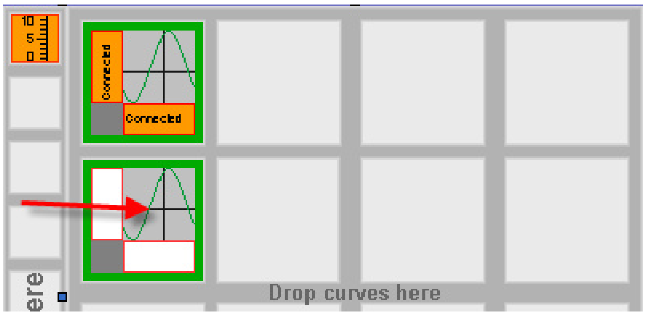
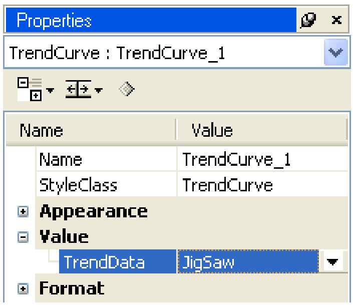


Fig. 44: Insert a trend curve in the trend container

The connection to the process variable is established by linking the trend objects found under the "**Trend Data**" node.



TrendCurve : TrendCurve_1

Name	Value
Name	TrendCurve_1
StyleClass	TrendCurve
Appearance	
Value	
TrendData	JigSaw
Format	

Fig. 45: Connecting trend data

For the trend curve to have reference to a scaling axis during runtime, the curve must be assigned a new or existing scaling axis.

The first step is to assign an existing scaling axis to the curve's value and time axes.

This is done by dragging and dropping the scaling axis into the trend curve.

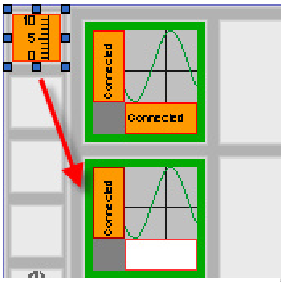


Fig. 46: Assign scaling axis

Perform this step for both axis labels.

Task: Configure the trend curve in the trend container

Insert a new curve in the trend container and connect the "**JigSaw**" trend object you just created in the trend data.

Set the curve to color index = 2 (green).

Connect the existing scaling axes to the x and y axes.

Result

The two curves are displayed in the "**Trend**" control. The value limits of the data type connected to the trend data (USINT = 0 to 255) is automatically applied to the scaling of the value axis (y).

The process variable "**gHeating.status.actTemp**" has the data type **REAL**, which doesn't permit automatic scaling.

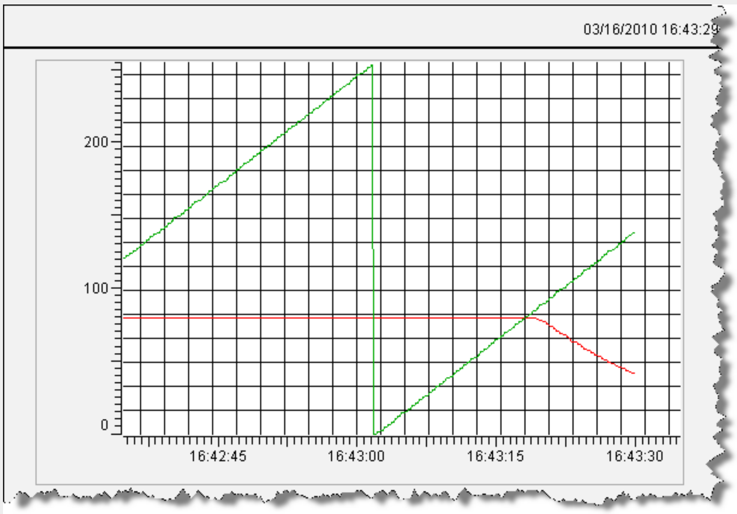


Fig. 47: Displaying two curves in the trend control

Visual Components Help: Control reference / TrendCurve

3.4.5 Multiple scales for the value axis

In the last example, the trend container that displays the temperature in a range between 0 and 100°C was expanded to include a second curve with a range of 0 to 255.

Since the two curves use the same scale for the y-axis (TrendValueScale), the limit values for the largest range are used automatically.

If the curves should be displayed with different scales over the entire value range, a second y-axis must be used.

A new scaling axis must be inserted for the y-axis. To do this, drag and drop the "**TrendScaleContainer**" control to the left of the existing container for the value scales.

In this container, insert the "**TrendValueScale**" control and connect it to the second trend curve via drag and drop.

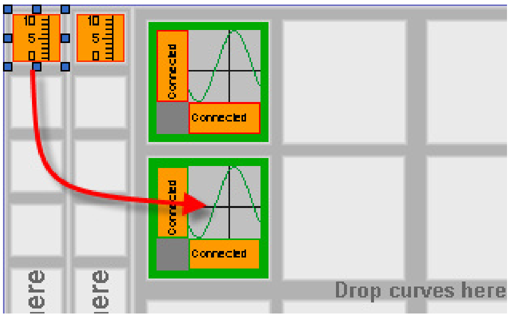
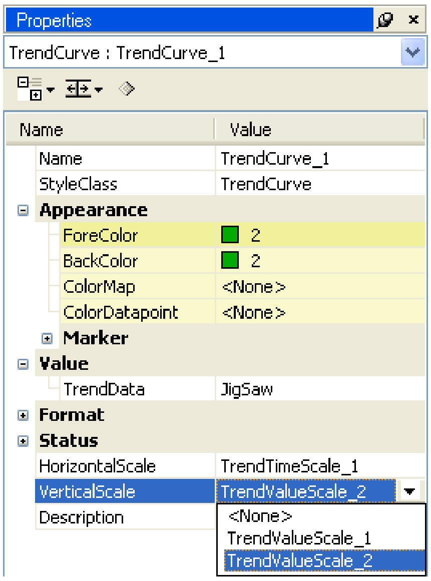


Fig. 49: Connecting the TrendValueScale control

This can also be done in the property editor of the second curve by selecting the "**TrendValueScale**" control for the "**VerticalScale**" property.



Properties

TrendCurve : TrendCurve_1

Name	Value
Name	TrendCurve_1
StyleClass	TrendCurve
⊟ Appearance	
ForeColor	🟩 2
BackColor	🟩 2
ColorMap	<None>
ColorDatapoint	<None>
⊞ Marker	
⊟ Value	
TrendData	JigSaw
⊞ Format	
⊞ Status	
HorizontalScale	TrendTimeScale_1
VerticalScale	TrendValueScale_2
Description	<None> TrendValueScale_1 TrendValueScale_2

Fig. 48: Changing trend scales

Task: Add a second y scaling axis

Insert a second scaling axis for the values in the trend container.

In this scaling axis, insert the "**TrendValueScale**" control and connect it to the second trend curve.

Result

Two scales with different ranges are shown for the y-axis.

This allows curves with different ranges to be displayed in their full height.

The temperature is displayed with the scale of the process variable and the unit text. The sawtooth curve is displayed unscaled with the limits defined by the data type "USINT".

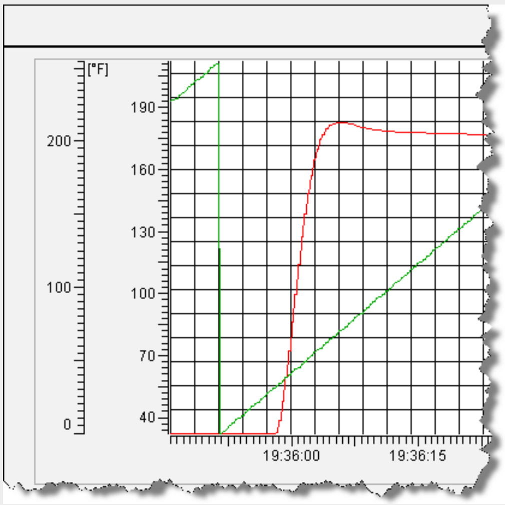


Fig. 50: Curve with second scale

Visual Components Help: Control reference / TrendScaleContainer

3.5 Interact with the trend system

The trend system is operated using key actions or using process variables connected to the properties of the trend objects.

In these exercises, we will be changing the range displayed on the time axis. This is done with buttons that perform the "**Zoom**" key action.

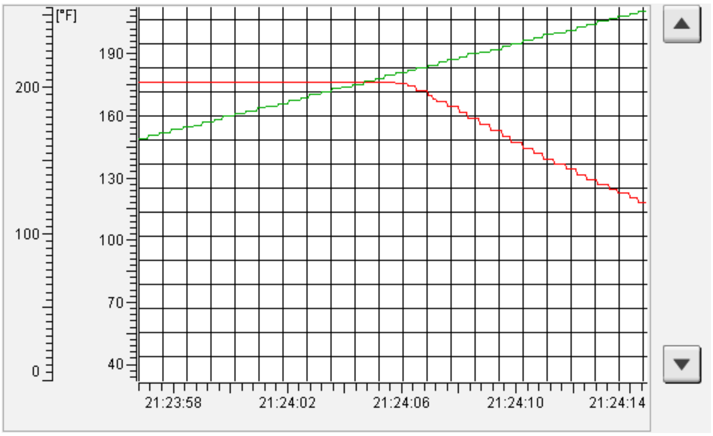


Fig. 51: Changing the time axis

In order to be able to operate the "**Trend**" control, you must set the property "**Input = True**".

Task: Use two buttons to zoom the time range

Create two buttons to the right of the trend control that can be used to zoom the time range.

One button increases the displayed time range and the other decreases it.

This is done using the "**Zoom**" key action, and the displayed time range (current) is changed horizontally.

Example of the "Action" property for decreasing the size of the time range:

• Keys / Action / Type	**Zoom**
• Keys / Action / Base	**Current**
• Keys / Action / Direction	**Horizontal**
• Keys / Action / Factor	**0.5**

Procedure:

- Set the "**Input**" property of the trend control to "**True**"
- Insert two buttons to the right of the trend control
- Create a new virtual key on the "**Keys**" property of the button
- Name the virtual keys "**vkZoomIn**" and "**vkZoomOut**"
- Select and configure the "**Zoom**" key action
- Use a zoom factor of 0.5 for decreasing and 2.0 for increasing the time range.

Result

A factor smaller than 1 reduces the displayed range, while a factor greater than 1 increases it.

If the displayed time range larger than the duration of recording for the trend data, **(SampleRate * SampleCount**) no data is displayed outside recorded duration.

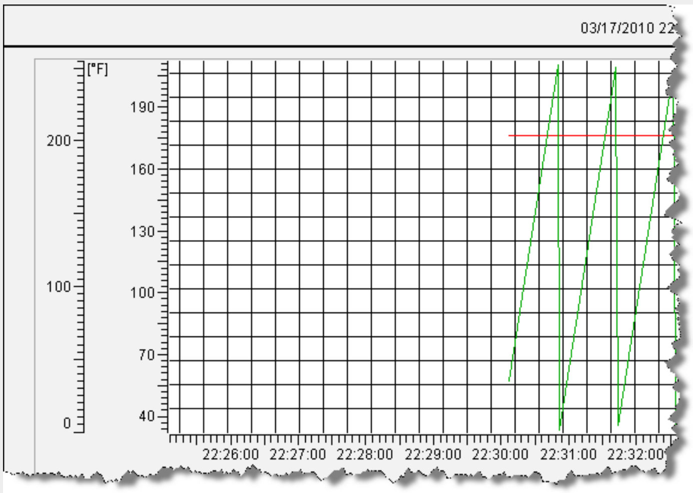
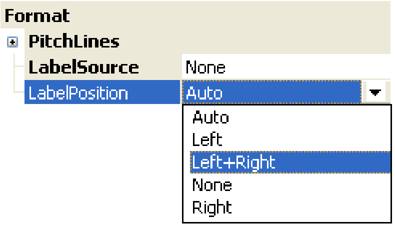


Fig. 52: Zooming the time range

The format of the time axis is set automatically (default setting).

To change how the time axis is displayed, change the "**LabelPosition**" property for the "**TrendTimeScale**" control in the trend container.



Format

PitchLines

LabelSource None

LabelPosition Auto

Fig. 53: Configure the time format for the time axis

For more information on the extensive options for operating the trend system, consult the Visual Components Help system.

Visual Components Help:
FAQ / Trend System
Key action reference / Zoom
Control reference / Trend
Control reference / TrendValueScale
Control reference / TrendTimeScale
Control reference / TrendCurve

4. USING THE SYSTEM DIAGNOSTICS MANAGER

With Automation Runtime 3.0 and higher, the controller can be diagnosed using the **S**ystem **D**iagnostics **M**anager (SDM) integrated in the operating system.

The HTML-based SDM pages can be easily displayed in Visual Components using the "**HTMLView**" control.

The full SDM functionality is automatically available on the Automation Runtime system without making any changes to the visualization application.

Automation Studio Help system
Diagnostics and Service / Diagnostics Tool / System Diagnostics Manager

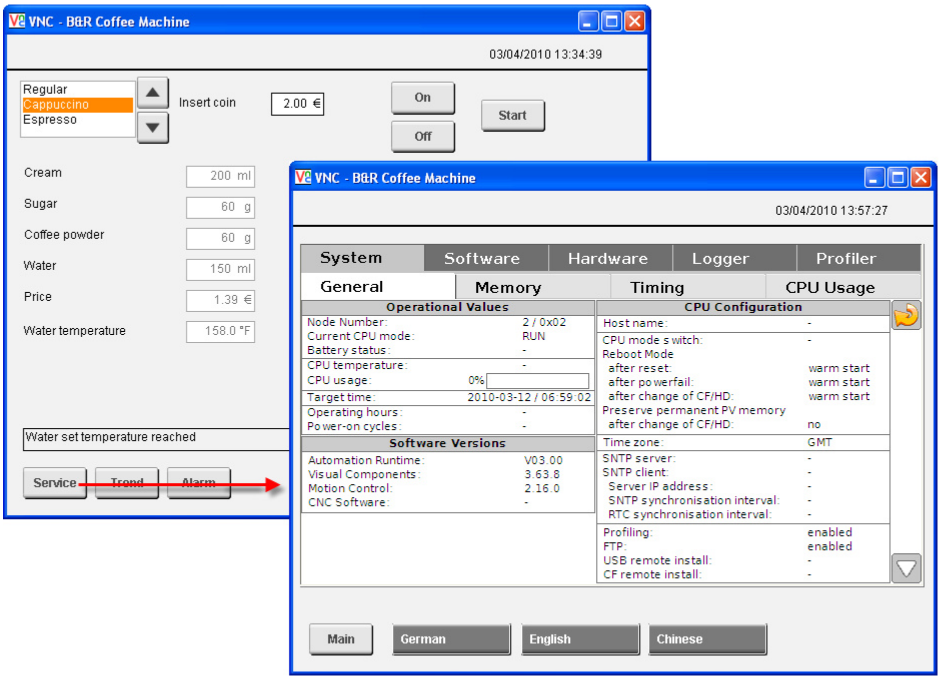
The System Diagnostics Manager can be used in any Visual Components application that is based on an SG4 target system (e.g. PowerPanel, APC, X20 CPU, etc.).

On the "PC based Runtime Simulation (AR000)" the SDM pages can be accessed **via Visual Components** starting with Automation Studio 3.0.81.

This training module will only show the procedure for configuring SDM in Visual Components. With Automation Studio 3.0.80 these tasks can be completed using an SG4 target system (e.g. PowerPanel PP420).

4.1 Task

The task in this training module is based on the visualization application created in TM610.



03/04/2010 13:34:39

Insert coin 2.00 €

Cream 200 ml
Sugar 60 g
Coffee powder 60 g
Water 150 ml
Price 1.39 €
Water temperature 158.0 °F

Water set temperature reached

03/04/2010 13:57:27

```
Operational Values
Node Number:                 2 / 0x02
Current CPU mode:            RUN
Battery status:              -
CPU temperature:             -
CPU usage:                0%
Target time:       2010-03-12 / 06:59:02
Operating hours:             -
Power-on cycles:             -
Software Versions
Automation Runtime:          V03.00
Visual Components:           3.63.8
Motion Control:              2.16.0
CNC Software:                -
```

```
CPU Configuration
Host name:                               -
CPU mode switch:                         -
Reboot Mode
 after reset:                            warm start
 after powerfail:                        warm start
 after change of CF/HD:                  warm start
Preserve permanent PV memory
 after change of CF/HD:                  no
Time zone:                               GMT
SNTP server:                             -
SNTP client:                             -
 Server IP address:                      -
 SNTP synchronisation interval:          -
 RTC synchronisation interval:           -
Profiling:                               enabled
FTP:                                     enabled
USB remote install:                      -
CF remote install:                       -
```

Fig. 54: Sample program "System Diagnostics Manager"

In this example, the **S**ystem **D**iagnostics **M**anager will be enabled in Automation Runtime and displayed on the "**ServicePage**".

The following tasks will be performed in this training manual:

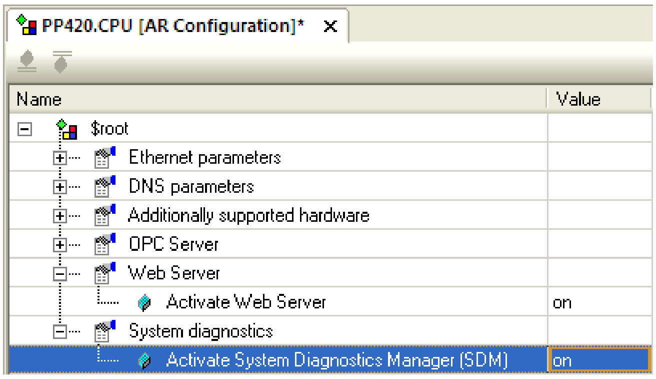
- Enable the System Diagnostics Manager in the Automation Runtime configuration
- Display the SDM on the "**ServicePage**"

The "**ServicePage**" and the buttons for switching to and from this page were created in TM610.

4.2 Enabling the System Diagnostics Manager

The SDM is enabled in the **physical view** by right clicking on the CPU node and selecting **<Open AR Configuration>** from the shortcut menu.

In the configuration editor that opens, you must enable the **WebServer** and the **SDM**(Property = On).



PP420.CPU [AR Configuration]*

Name	Value
$root	
Ethernet parameters	
DNS parameters	
Additionally supported hardware	
OPC Server	
Web Server	
Activate Web Server	on
System diagnostics	
Activate System Diagnostics Manager (SDM)	on

Fig. 55: Enable SDM in the current configuration

From this point on, the connection to the SDM can be tested using a **web browser**. This connection is also possible on the "PC based Automation Runtime Simulation (AR000)".

URL = **http://localhost/sdm-vga/vga.cgi** (access to HTML pages)
URL = **http://localhost/sdm** (access to SVG pages)

If your browser uses a proxy server, the IP addresses for SDM access to the proxy server must be circumvented.

When accessing the "PC based Automation Runtime Simulation (AR000)" the IP addresses "**127.0.0.1**" and "**Localhost**" need to be added for the proxy server.

Automation Studio Help system
Diagnostics and Service / Diagnostics Tool / System Diagnostics Manager

4.3 Displaying the SDM HTML pages

The HTML pages of the SDM are displayed using an "**HTMLView**" control.

The URL for displaying the HTML page can be configured either in a **text group** or on a **URL data point**.

Advantage of using text group:

- Language-dependent URL possible

Advantage of using URL data point:

- Easier to set up in Visual Components
- Can be configured dynamically from the application program

Requirement for displaying the SDM:

- The size of the HTMLView control on the page must be configured for the respective resolution.
- There must be no border (Border = <None>)

Supported display sizes:

Panel size	HTMLView size	URL
QVGA	316 x 191	http://IP-Adresse/sdm-qvga/qvga.cgi
VGA	620 x 339	http://IP-Adresse/sdm-vga/vga.cgi
XGA	1004 x 567	http://IP-Adresse/sdm-xga/xga.cgi

In the visualization, the page resolution or smaller can be used (e.g. QVGA pages displayed in a VGA visualization).

The text for the URL "**http://localhost/sdm-vga**" should be defined in a text group for all available languages.

Task: Enable the SDM and display the HTML page

Your task is to enable the SDM in the Automation Runtime configuration.

Enter the URL in a new text group named "**URL_SDM**" for the configured languages.

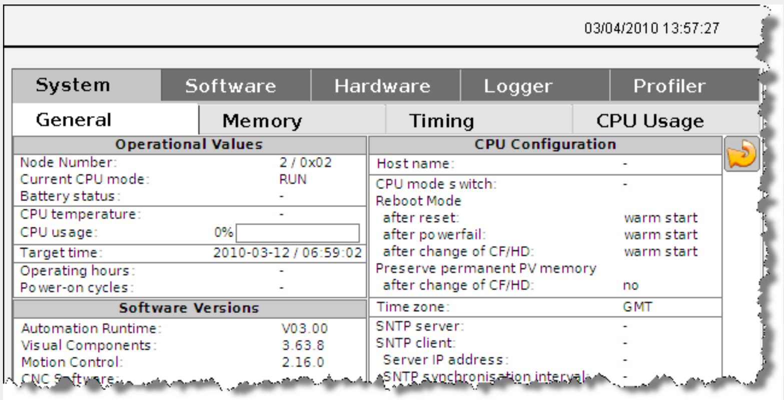
Insert an HTMLView control on the "**ServicePage**" with the correct size (620 x 339 pixels) and connect the text group "**URL_SDM**" to the "**Value**" property.

Procedure:

- Create a new text group and name it "**URL_SDM**"
- Insert text with the URL
- Insert an HTMLView control on the "**ServicePage**" with the necessary size
- Connect the newly created text group to "**Value** / **DefaultURLs**" property

Result

The pages of the System Diagnostics Manager are displayed on the "**ServicePage**".



03/04/2010 13:57:27

Operational Values	
Node Number:	2 / 0x02
Current CPU mode:	RUN
Battery status:	-
CPU temperature:	-
CPU usage:	0%
Target time:	2010-03-12 / 06:59:02
Operating hours:	-
Power-on cycles:	-
Software Versions	
Automation Runtime:	V03.00
Visual Components:	3.63.8
Motion Control:	2.16.0
CNC Software:	

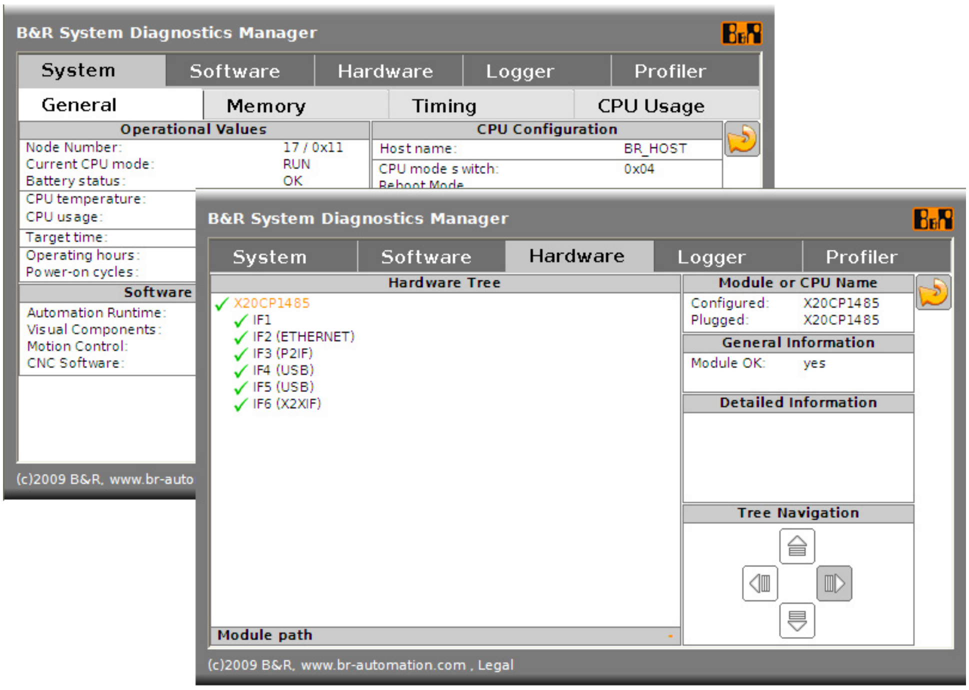
CPU Configuration	
Host name:	-
CPU mode switch:	-
Reboot Mode	
after reset:	warm start
after powerfail:	warm start
after change of CF/HD:	warm start
Preserve permanent PV memory	
after change of CF/HD:	no
Time zone:	GMT
SNTP server:	-
SNTP client:	-
Server IP address:	-
SNTP synchronisation interval:	-

Fig. 56: Displaying the SDM pages in the "HTMLView" control

4.4 Interaction with the SDM

All interaction with the SDM takes place using the HTML pages.

The screen can only be refreshed using the Refresh button on the HTML pages. You can navigate between the SDM functions using the tabs in the header.



Node Number: 17 / 0x11
Current CPU mode: RUN
Battery status: OK
CPU temperature:
CPU usage:
Target time:
Operating hours:
Power-on cycles:

Host name: BR_HOST
CPU mode switch: 0x04
Reboot Mode

Automation Runtime:
Visual Components:
Motion Control:
CNC Software:

✓ X20CP1485
✓ IF1
✓ IF2 (ETHERNET)
✓ IF3 (P2IF)
✓ IF4 (USB)
✓ IF5 (USB)
✓ IF6 (X2XIF)

Configured: X20CP1485
Plugged: X20CP1485

Module OK: yes

Fig. 57: Image navigation in the System Diagnostics Manager

5. SUMMARY

The alarm and trend system records events in the process sequence and the changes to process data over time. This data is then displayed in Visual Components using corresponding controls.

Fig. 58: Visual Components - Alarm system, Trend system and Diagnostics

With the freely configurable alarm properties and the multiple possibilities for display, the alarm system is an important Visual Components function that can be used in all of your visualization applications.

The trend system also provides extensive options for displaying curves. Wizards assist the user in creating a basic trend configuration.

The SDM integrated in Automation Runtime provides basic system diagnostics functions for every visualization application.

Notes

Overview of training modules

TM210 – The Basics of Automation Studio
TM211 – Automation Studio Online Communication
TM213 – Automation Runtime
TM220 – The Service Technician on the Job
TM223 – Automation Studio Diagnostics
TM230 – Structured Software Generation
TM240 – Ladder Diagram (LAD)
TM241 – Function Block Diagram (FBD)
TM246 – Structured Text (ST)
TM250 – Memory Management and Data Storage
TM261 – Closed Loop Control with LOOPCONR

TM400 – The Basics of Motion Control
TM410 – The Basics of ASiM
TM440 – ASiM Basic Functions
TM441 – ASiM Multi-Axis Functions
TM445 – ACOPOS ACP10 Software
TM446 – ACOPOS Smart Process Technology
TM450 – ACOPOS Control Concept and Adjustment
TM460 – Starting up Motors
TM480 – Hydraulic Drive Control

TM500 – The Basics of Integrated Safety Technology
TM510 – ASiST SafeDESIGNER
TM540 – ASiST SafeMC

TM600 – The Basics of Visualization
TM610 – The Basics of ASiV
TM630 – Visualization Programming Guide
TM640 – ASiV Alarm System, Trend and Diagnostic
TM670 – ASiV Advanced

TM700 – Automation Net PVI
TM710 – PVI Communication
TM711 – PVI DLL Programming
TM712 – PVIServices
TM730 – PVI OPC

TM800 – APROL System Concept
TM810 – APROL Setup, Configuration and Recovery
TM811 – APROL Runtime System
TM812 – APROL Operator Management
TM813 – APROL XML Queries and Audit Trail
TM830 – APROL Project Engineering
TM840 – APROL Parameter Management and Recipes
TM850 – APROL Controller Configuration and INA
TM860 – APROL Library Engineering
TM865 – APROL Library Guide Book
TM870 – APROL Python Programming
TM890 – The Basics of LINUX

KONZERNZENTRALE

Bernecker + Rainer Industrie-Elektronik Ges.m.b.H.
B&R Strasse 1
5142 Eggelsberg
Austria
Tel.: +43 (0) 77 48 / 65 86 - 0
Fax: +43 (0) 77 48 / 65 86 - 26
info@br-automation.com
www.br-automation.com

TM640TRE.30-GER 0310

155 Büros in 60 Ländern - **www.br-automation.com/contact**

Argentina • Australia • Austria • Belarus • Belgium • Botswana • Brazil • Bulgaria • Canada • Chile • China • Colombia • Croatia • Cyprus
Czech Republic • Denmark • Egypt • Emirates • Finland • France • Germany • Greece • Hungary • India • Indonesia • Ireland
Israel • Italy • Japan • Korea • Luxemburg • Kyrgyzstan • Malaysia • Mexico • Mozambique • Namibia• The Netherlands
New Zealand • Norway • Pakistan • Peru• Poland • Portugal • Romania • Russia • Serbia • Singapore • Slovakia • Slovenia • South Africa
Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • USA • Venezuela • Vietnam • Zimbabwe