

TM640

Alarms, Trends and Diagnostics



Prerequisites and requirements

Training modules	TM610 – Working with Integrated Visualization
Software	Automation Studio 4.1
Hardware	None

TABLE OF CONTENTS

1 INTRODUCTION.....	4
1.1 Learning objectives.....	4
2 THE VISUAL COMPONENTS ALARM SYSTEM.....	5
2.1 Alarm system terminology.....	5
2.2 Task definition.....	7
2.3 Configuring the alarm system.....	8
2.4 Viewing alarms at runtime.....	10
2.5 Interacting with the alarm system.....	15
2.6 Acknowledging alarms.....	19
2.7 Bit and value alarms.....	20
2.8 mapp technology centralized alarm management.....	21
3 THE VISUAL COMPONENTS TREND SYSTEM.....	24
3.1 Trend system terminology.....	24
3.2 Task definition.....	26
3.3 Configuring the trend system.....	27
3.4 Displaying an online trend.....	28
3.5 Interacting with the trend system.....	38
4 DIAGNOSTICS USING THE SYSTEM DIAGNOSTICS MANAGER (SDM).....	40
4.1 Task definition.....	41
4.2 Enabling the System Diagnostics Manager.....	42
4.3 Displaying the SDM HTML pages.....	42
4.4 Interacting with the SDM.....	44
5 SUMMARY.....	45

Introduction

1 INTRODUCTION

A visualization project can also include components for diagnosing data, process sequences and the status of the machine. Visual Components makes it possible to record the machine status or data changes over time and to display them in the form of event messages or curves. The alarm and trend systems integrated into Visual Components can record this data for on-screen display.



Figure 1: Visual Components - Alarms, trends and diagnostics

Web-based diagnostics of the controller can be performed using the System Diagnostics Manager that is integrated into the operating system. These diagnostic pages can be implemented easily in any Visual Components application.



The Help links in this document are relative Help paths under:
"Visualization \ Visual Components VC4"

1.1 Learning objectives

This training module uses selected examples to help you learn how to create pages for visualizing application data, alarm states and diagnostic data in Visual Components.

- You will learn how the alarm system integrated in Visual Components works.
- You will learn how to configure alarm groups and display them using the Alarm control.
- You will learn how to use filtering to optimize the display of alarms and to view historical alarms.
- You will learn the various alarm states and how they are acknowledged.
- You will learn how the Trend control works.
- You will learn how to configure the Trend control using the Trend wizard and display multiple trend curves.
- You will learn how to adapt trend scaling to the data being displayed and set the time and units.
- You will learn about the ability of the Trend control to zoom in and out of a trend curve.
- You will learn the functions of the System Diagnostics Manager.
- You will learn how to embed the System Diagnostics Manager into the visualization application and navigate to specific pages.

2 THE VISUAL COMPONENTS ALARM SYSTEM

Alarms are used to record and respond to certain system states. Creating alarm groups allows system states to be organized as messages, warnings and alarms. The alarm system is a resource used by all visualization pages in a project and is configured under "Alarm Groups".



Figure 2: Managing the alarm system

Features of the alarm system:

- Displays both active and historical alarms
- Possible to use alarm groups to manage alarm texts and their configurations
- Color-coded identification of alarm priorities
- Possible to use a touch screen or keys to scroll through alarm lists as well as to 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)



Shared resources \ Alarm system

2.1 Alarm system terminology

This section will describe the terminology used in the alarm system.

Alarm texts

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

- **Messages** are non-critical system or machine messages. Acknowledgment 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

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

Acknowledge

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.

The Visual Components alarm system

Bypassing

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

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

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 an investigation 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, two operators were faced with 274 alarms that had to be acknowledged and responded to appropriately.

2.2 Task definition

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

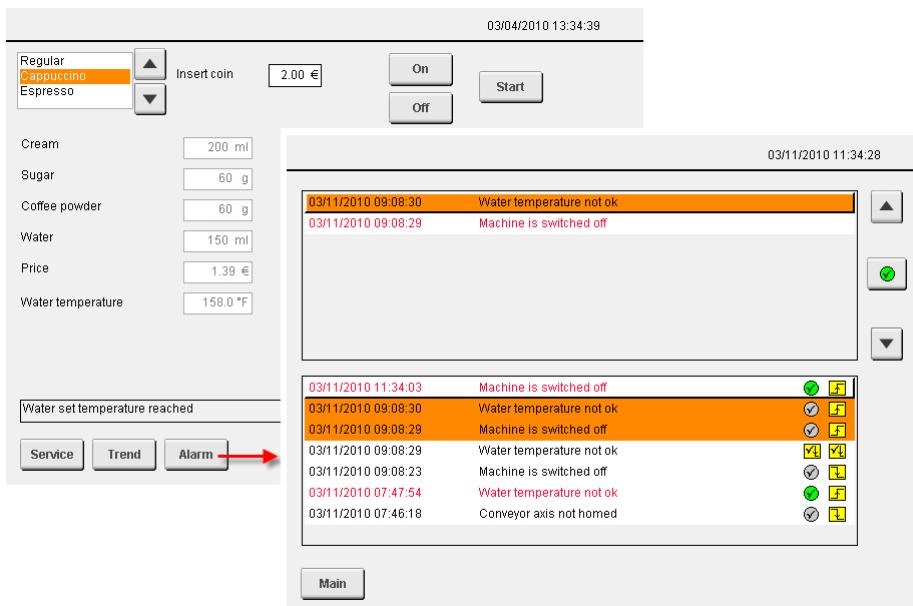


Figure 3: "AlarmSystem" example program

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

The following tasks will be performed in this training module:

- Creating a new alarm page
- Understanding the "CoffeeMachine" alarm group
- Displaying active alarms
- Displaying historical alarms
- Acknowledging alarms
- Displaying 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 module.

Exercise: Create a new page called "AlarmPage"

Under "Pages", 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".

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

The Visual Components alarm system



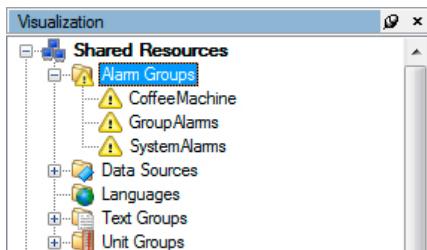
Figure 4: Changing to the "Alarm" page

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 through the alarm pages?

2.3 Configuring the alarm system



In Visual Components, alarms are managed and configured under **"AlarmGroups"**. A new alarm group can be inserted by pressing the **"Insert"** button or by selecting **<Add Alarm Group>** from the shortcut menu.

Figure 5: Alarm groups

Events of the same type (alarm, warning, message) and the same priority are grouped in alarm groups. As a result, all of 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.



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



Shared resources \ Alarm system \ Alarm groups

Shared resources \ Alarm system \ Alarm groups \ Predefined alarm groups

2.3.1 Managing alarms in the application program

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

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

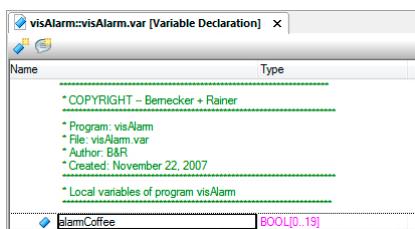


Figure 6: Mapping alarm states to the process variable "alarmCoffee"

2.3.2 "CoffeeMachine" alarm group

The process variable used to map 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.

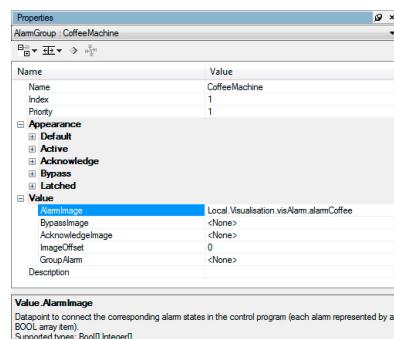


Figure 7: Connecting the alarm image

? Shared resources \ Alarm system \ Alarm groups \ Configuring alarm groups

2.3.3 Alarms in the "CoffeeMachine" alarm group

The alarms in an alarm group are created like texts in a text group. The properties of an alarm text determine its behavior during runtime.



Figure 8: Alarms in the "CoffeeMachine" alarm group

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

Acknowledge

"Acknowledge" defines whether this alarm needs to be acknowledged when it occurs during runtime (warning, alarm) or not (message). If an alarm requires acknowledgment, it must be acknowledged by the user during runtime so that it can be removed from the display once the alarm state becomes "inactive" again.

The Visual Components alarm system

WhileActive

This property specifies whether an acknowledgeable alarm is allowed to be acknowledged while it is still pending.



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



[Shared resources \ Alarm system \ Alarm groups \ Configuring alarm groups](#)

2.3.4 Static and dynamic alarm text

The most basic alarm text contains only text. "Text snippets" 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 will be used ([2.7 "Bit and value alarms" on page 20](#)) later on.

03/11/2010 23:45:20	Water temperature 167.5 °F not ok
03/11/2010 23:45:19	Machine is switched off

Figure 9: Displaying an alarm with a value



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.



[Shared resources \ Alarm system \ Alarm groups \ Alarms \ Text snippet](#)

2.4 Viewing alarms at runtime

Alarms can be displayed in different lists during runtime according to the necessary 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 acknowledgment. This view is used most frequently.

This image is compiled from the respective alarm bits when the visualization is started..

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

Exercise: Insert an alarm control to display the "Current" alarm list

On the "AlarmPage", have the "Current" alarm list appear in the top half of the page. The alarm list should therefore be placed in the top half of the screen. Leave about 50 pixels free to the right of the alarm list.

- 1) Insert an "Alarm" control on the "AlarmPage".
- 2) Adjust its size.
- 3) Change the "Value / Source" property.



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 (acknowledgment not required).

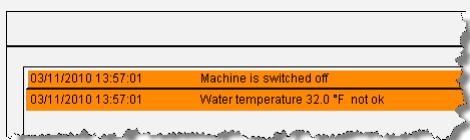


Figure 10: "Current" alarm list



Handling of the alarm list is described under [2.5 "Interacting with the alarm system" on page 15](#).

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

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

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



Figure 11: Adjusting the columns of the alarm list

The Visual Components alarm system

2.4.2 Configuring the columns of the alarm list

The "Column" 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.

Available alarm information:

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

Name	Value
Name	Alarm_1
StyleClass	AlarmCurrentEdit
Position	
Appearance	
Value	True
Input	
Format	
Cursor	
Columns	
Date Time	Column 1
AlarmText	Column 2
AcknowledgeState	Column 3
AlarmState	Column 4
GroupNumber	<Not Used>
AlarmNumber	<Not Used>
Priority	<Not Used>
BypassState	<Not Used>
Event	<Not Used>
Runtime	Description

Figure 12: Column configuration

Exercise: 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	Column 2	Column 3	Column 4
Time	Alarm text	Acknowledgment status as a bitmap	Alarm event as a bitmap

Table 1: Description of the columns in the alarm list



Control reference \ Alarm \ Columns



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.

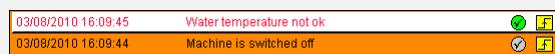


Figure 13: Displaying the "Current" alarm list

The Visual Components alarm system

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: Appearance of a latched alarm

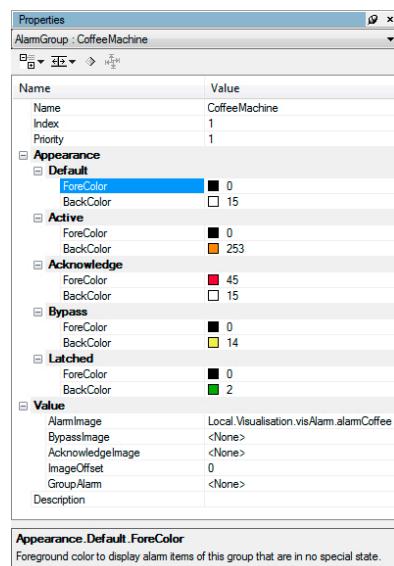


Figure 14: Color arrangement

As can be seen in the image above, the first alarm has been acknowledged while 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 to a configurable ring buffer.

This image is compiled from the historical alarm list when the visualization project is started.

The historical alarm list can be configured in the global properties for "Alarm Groups".

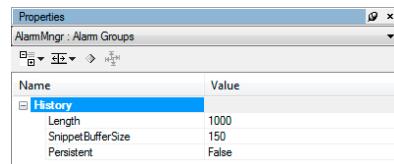


Figure 15: Configuring the alarm history



Shared resources \ Alarm system \ Historical alarm list

The view for the alarm list is selected using the "Value / Source" property.

Instructions: Insert an alarm control to display the "History" alarm list

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

To do so, 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.

- 1) Insert an "Alarm" control on the "AlarmPage".

The Visual Components alarm system

- 2) Adjust its size and align it with the "Current" alarm list.
- 3) Change the "Value / Source" property.
- 4) Change the "Columns" properties.



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



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	<input checked="" type="checkbox"/>
03/10/2010 12:33:45	Water temperature not ok	<input checked="" type="checkbox"/>
<hr/>		
03/10/2010 12:33:50	Conveyor axis not homed	<input checked="" type="checkbox"/>
03/10/2010 12:33:50	Conveyor axis controller is switched off	<input checked="" type="checkbox"/>
03/10/2010 12:33:50	Feeder axis not homed	<input checked="" type="checkbox"/>
03/10/2010 12:33:50	Feeder axis controller is switched off	<input checked="" type="checkbox"/>
03/10/2010 12:33:45	Conveyor axis not homed	<input checked="" type="checkbox"/>
03/10/2010 12:33:45	Conveyor axis controller is switched off	<input checked="" type="checkbox"/>
03/10/2010 12:33:45	Feeder axis not homed	<input checked="" type="checkbox"/>

Figure 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 Interacting with the alarm system

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

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:

- Scrolling in the alarm list
- Acknowledging an alarm that requires acknowledgment
- Acknowledging all alarms
- Bypassing an alarm



Figure 17: Example of acknowledging an alarm

Figure 18: Vertical touchpad

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 "Input" property to "True".

Key actions that operate another control must have the type "ControlDependent". For this type of key action, the control element for which the action should be undertaken is selected using the "Control" property.

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

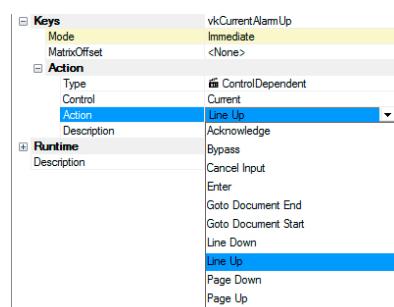


Figure 19: Selecting the alarm list

Exercise: Draw three buttons for the alarm system

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.
- 1) For the current alarm list, set the "Input" property to "True".
 - 2) Insert three buttons as shown in the template for scrolling and acknowledging.
 - 3) Create a new virtual task for each button.
 - 4) Select the key action "ControlDependent" for each button.

The Visual Components alarm system

- 5) Select the control for the "Current" alarm list.
- 6) Configure the respective actions.



The "ControlDependent" task action only affects controls whose "Input" property is set to "True".



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

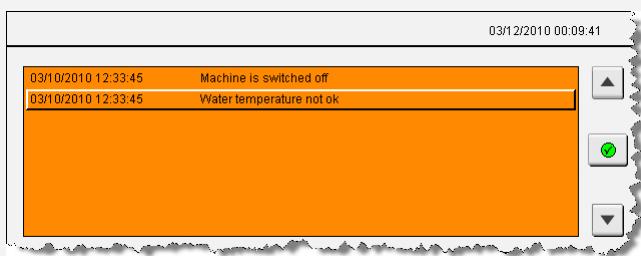


Figure 20: Scrolling in the "Current" alarm list

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



Figure 21: Acknowledging the selected alarm



Key action reference \ Control Dependent

Control reference \ Alarm \ Input

Buttons are used to adapt the operation of the alarm lists to a GUI (graphical user interface) template.

Another possibility is to use pre-designed 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.

Page : AlarmPage	
Name	Value
Name	AlarmPage
Index	2
StyleClass	default
Position	
Appearance	
ForeColor	■ 0
BackColor	□ 252
ColorDatapoint	<None>
Focus	
Edit	
ForeColor	□ 16
BackColor	■ 253
ColorDatapoint	<None>
Locked	
Help	
Runtime	
Description	

Figure 22: Page properties



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 application.

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 one you create yourself. In this example, the touchpad named "NavigationPad_ver" is used to operate the historical alarm list.



Figure 23: Vertical touchpad

Exercise: Touchpad operation in the historical alarm list

The historical alarm list is operated with the "NavigationPad_ver" touchpad. This touchpad can be selected from the "Input / Touchpad" property.

The Visual Components alarm system



When the historical alarm list is selected, the touchpad opens automatically. The touchpad makes it possible for you to scroll up and down.

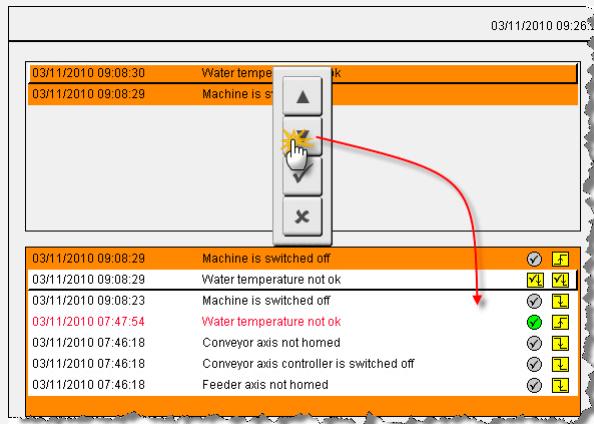


Figure 24: Scrolling down in the historic alarm list



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



The alarm system operates most efficiently if only 10 alarms per minute occur.

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.

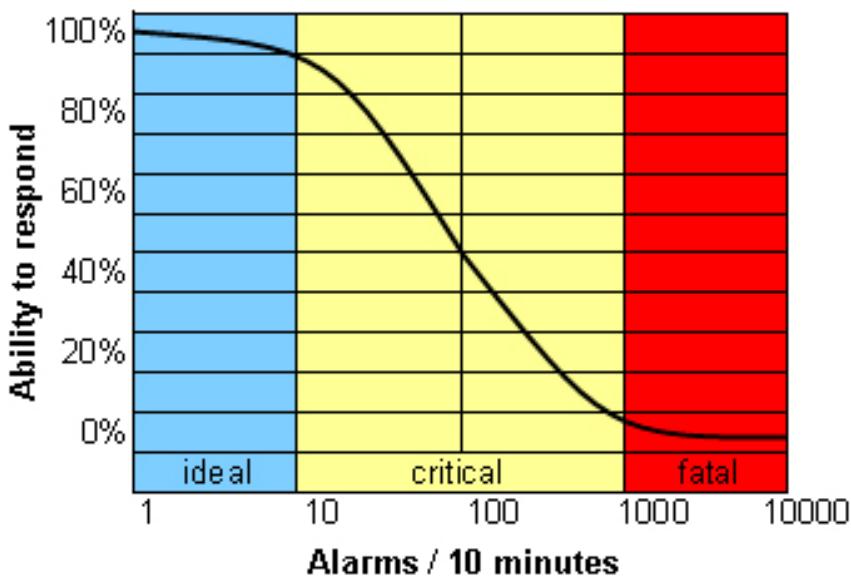


Figure 25: Ability to react to alarms



Control reference \ Alarm \ Value \ Filter

Shared resources \ Alarm system \ Alarm groups \ Configuring alarm groups

2.6 Acknowledging alarms

An alarm normally requires an action from the operator. This action can be "forced" using the alarm's acknowledgment 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.



Figure 26: Defining acknowledgment behavior

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

An acknowledgment image is connected to the "Value / AcknowledgementImage" property of an alarm group as a bit array.

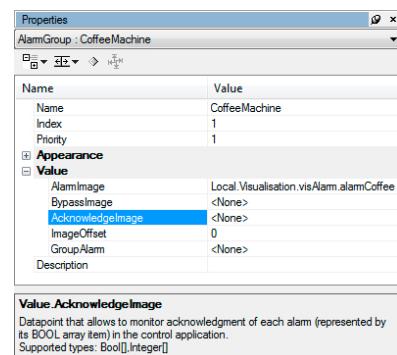


Figure 27: Acknowledgment image for an alarm group

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

Acknowledge	Behavior	Image
Disabled	Acknowledgment not possible or not required (e.g. message)	No effect on the acknowledgment image
Possible	Acknowledgment is only made possible by the "WhileActive" property, i.e. 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	Acknowledgment 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.

Table 2: Specifying acknowledgment behavior

Exercise: Change the acknowledgment behavior of an alarm

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

The Visual Components alarm system

- 1) Open the "CoffeeMachine" alarm group.
- 2) Change the acknowledgment behavior for the alarm text with Index [0].



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



Shared resources \ Alarm system \ Alarms \ Configuring alarms

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 "text snippets" and inserted anywhere in the alarm text.

Text snippets can be inserted in the alarm text editor under the "Text snippets" tab.

Name	Type	StyleClass	Details	Description
{1} WaterTemp	Numeric	default	Datapoint(actTemp)	

Texts Text Snippets Alarms Text Snippets

Figure 28: Inserting text snippets

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.

Exercise: 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.

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

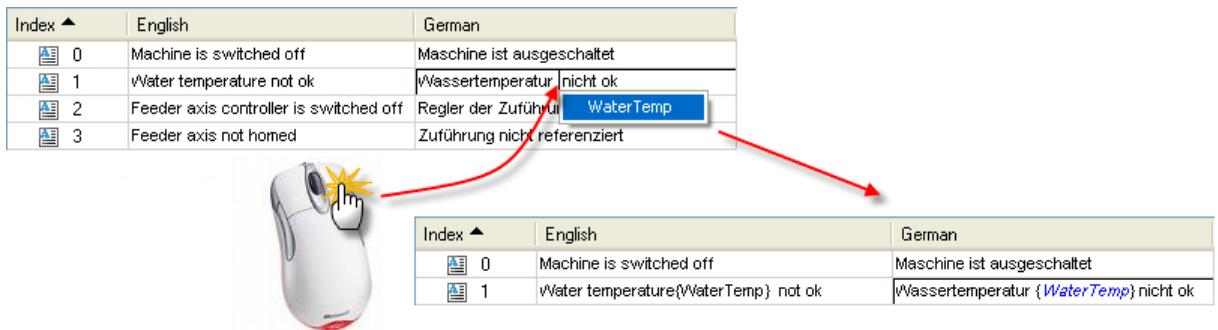


Figure 29: Inserting the text snippet at the desired position in the alarm text



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.

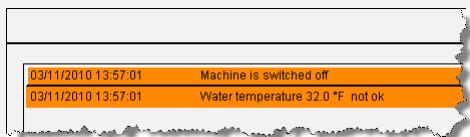


Figure 30: Alarm with a limit value



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



Shared resources \ Alarm system \ Alarms \ Text snippet



Figure 31: mapp technology logo

2.8 mapp technology centralized alarm management

mapp technology

With mapp technology, we offer users an easy-to-use interface for implementing advanced functionality. Many complex operations, such as loading and saving recipe data, controlling a drive axis and recording process values, are represented by easy-to-use mapp technology components.

mapp technology unites configuration and programming. Functionality is implemented in the application program using standard libraries. In addition, mapp provides configuration interfaces that allow the functionality of mapp components to be influenced without affecting their implementation in the application software.

The Visual Components alarm system

Centralized alarm management using mapp technology

The MpAlarm library provides centralized alarm management. It has been designed for use with Visual Components. Alarms are recorded and evaluated at a central location.

The function block MpAlarmBasic can be used to activate centralized alarm management. User alarms and alarms from mapp components are collected and managed. The MpAlarmBasicUI function block allows user alarms and alarms from mapp components to be transferred to the Visual Components application via an alarm and acknowledgment map. The MpAlarmSet and MpAlarmReset functions allow user alarms to be set and reset.

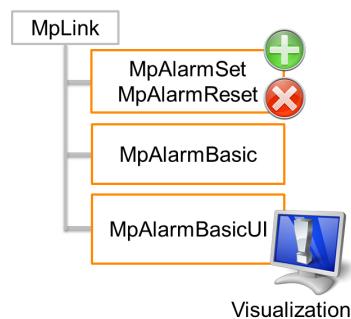


Figure 32: User alarms are entered in centralized alarm management system using MpAlarmSet and MpAlarmReset (MpAlarmBasic); Visualization is handled using MpAlarmBasicUI

Communication between the individual components takes place using a global variable that has a name which must be unique in the system. Visualization of alarms for a certain mapp component should be handled using the MpLink for the corresponding component.

Centralized alarm management supports configuration of description texts for user alarms and reactions to all alarms defined in the system via WebXs.



Solutions \ Technology Solutions \ mapp
Programming \ Libraries \ mapp technology \ MpAlarm

Exercise: Enable the centralized alarm system, display user alarms

The centralized alarm system can be used to manage mapp alarms and user alarms. The visualization component establishes the connection between the alarm and acknowledgment mapping data. It is therefore possible to easily display all alarms occurring in the system with the Visual Components alarm system.

- 1) Create the state machine.
- 2) Call the MpAlarmBasic component.
- 3) Wait until the "Active" output = TRUE.
- 4) Create the alarm and acknowledgment map and the string array for additional information.
- 5) Complete the configuration structure for the MpAlarmBasicUI component.
- 6) Call the MpAlarmBasicUI component , use mpALARM_UI_MODE_USER mode.
- 7) Enter user alarms in the centralized alarm management system using MpAlarmSet.
- 8) Check the alarm bit field and additional information.

The alarm and acknowledgment mappings can now be connected with a new alarm group in the Visual Components HMI application. The additional information texts can be added to the alarm texts using text snippets, for example.



When calling the MpAlarmBasicUI component, a warning with ID mpALARM_WRN_IMAGE_SIZE occurs if the alarm and acknowledgment mappings are too small. The size of the arrays must be large enough to contain 100 elements.

Exercise: Display mapp component alarms

In addition to the status information on the function block output, many mapp components also return alarms directly. These alarms can be loaded using the visualization component for the centralized alarm system. In this case, it is necessary for the centralized alarm system to have already been selected. Alarms from a specific mapp component are accessed using the MpLink for the component. An attempt should be made to display the recipe component alarms. An alarm is triggered when trying to load a recipe file that does not exist, for example.

- 1) Declare the alarm and acknowledgment bit field as well as the string array for additional information.
- 2) Complete the configuration structure for the MpAlarmBasicUI component.
- 3) Enable the MpAlarmBasicUI component, configure mpALARM_BASIC_UI_MODE_MAPP mode.
- 4) Try to load a recipe file that does not exist.
- 5) Check the status ID of the MpRecipeXml component.
- 6) Check the alarm bit field and additional information.

Exercise: Alarm configuration via WebXs

The automatically generated WebXs can now be used to configure the description texts for user alarms as well as reactions for user and mapp alarms.

- 1) Configure the reactions for user alarms.
- 2) Configure the reactions for recipe component alarms.
- 3) Save the configuration and reinitialize the component for the centralized alarm system.
- 4) Check the "Reaction" output for the MpAlarmBasic component.



The MpAlarmBasic component must be reactivated in order to apply the modified reactions to the WebXs. This is done by setting the enable input of the component to FALSE and then to TRUE.

The Visual Components trend system

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.

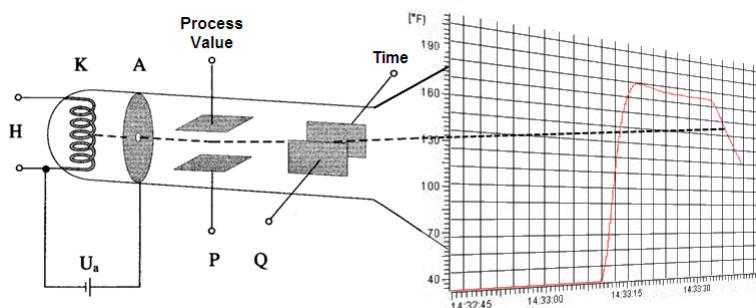


Figure 33: Virtual oscilloscope - Trend system

Possibilities of the trend system:

- Displaying process values over time
- Displaying the recorded values from the application program
- Viewing up to 16 graphs with color coding
- Up to two measurement cursors to view values
- Flexible configuration of value and time axes
- Zooming and scrolling through the trend data
- Showing and hiding curves



Visualization resources \ Trends

3.1 Trend system terminology

This chapter describes the terminology used in the trend system.

Trend control

The trend control element is used to display a trend window (trend container) on a page.

Trend data

Trend data defines which process variables should be recorded and displayed.

There are two types of trend data:

- **Online trends** record the process variable(s) at a defined sample rate for a defined interval (number of samples).
- **User trends** display data from a defined number of process values in an array from the application program.

Trend curves

Trend curves show the trend data in the trend container.

Trend scaling

Trend scaling is used to label and subdivide the time and value axes in the trend container.

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
- Several value axes with different scaling and labeling functions



The trend system is composed of various controls and configurations that can be created using the Trend Wizard.

The Visual Components trend system

3.2 Task definition

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

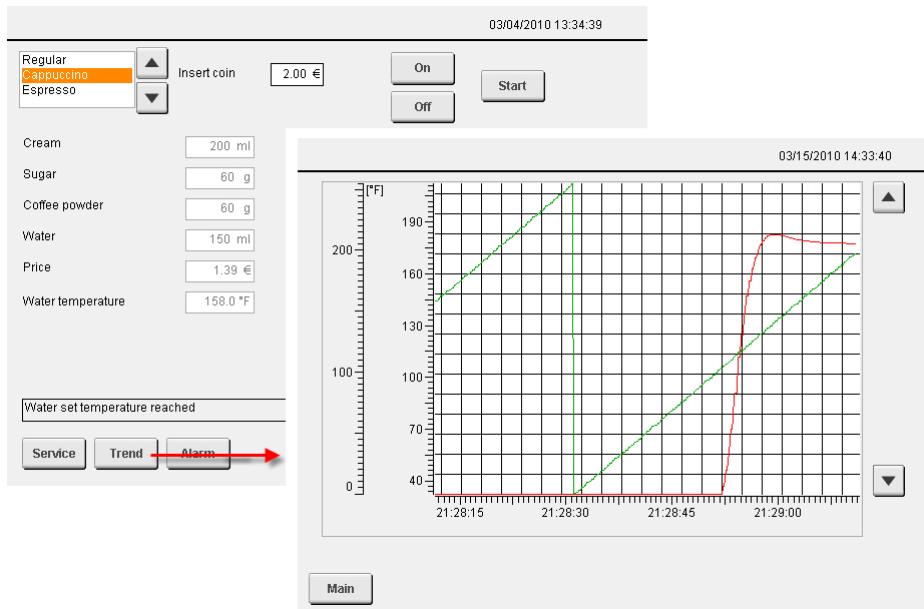


Figure 34: "Trend system" example program

In this example, 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 module:

- Creating a new trend page
- Developing a trend page with the "Trend Wizard"
- Displaying a temperature curve in the trend system
- Adding a second curve with a different scale
- Interacting with the trend system



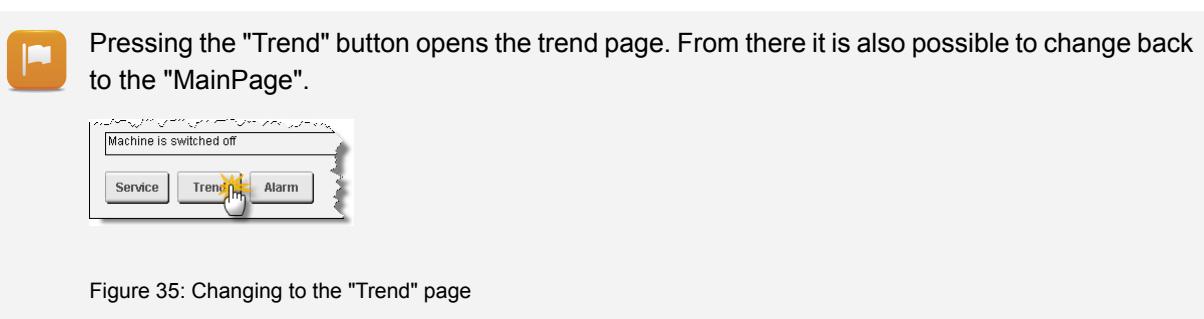
The procedure for creating a new page for displaying a trend and for changing from the main page to this new page are described in the TM610 training module.

Exercise: Create a new page called "TrendPage"

Under "Pages", 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".



3.3 Configuring the trend system

There are two ways to configure the trend system:

- Trend Wizard (express configuration)
- 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.

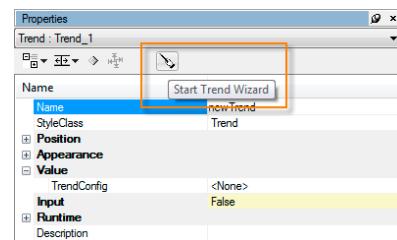


Figure 36: Starting the "Trend Wizard"

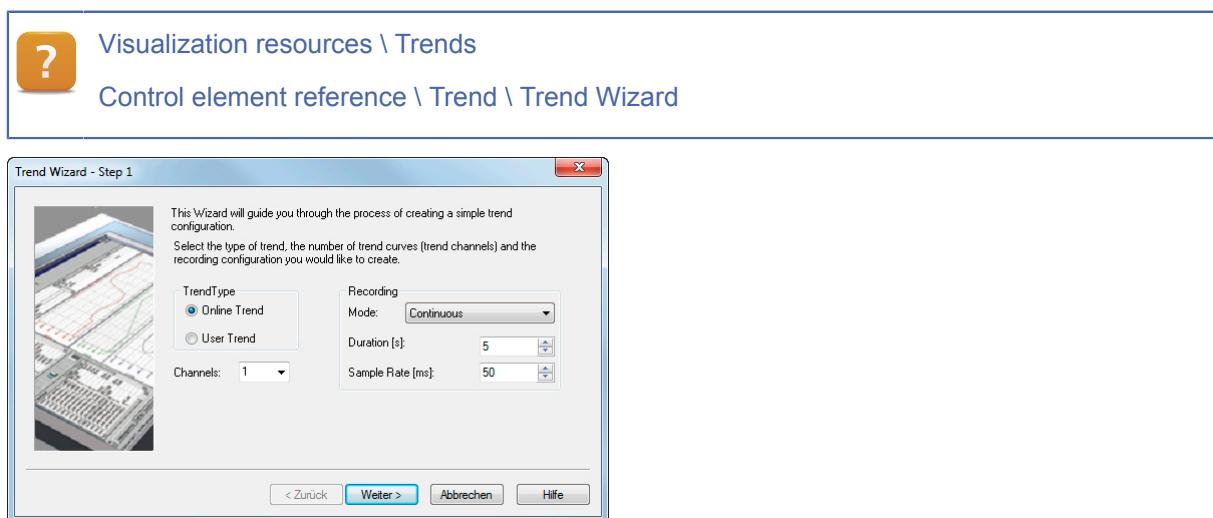


Figure 37: Creating a trend configuration using the "Trend Wizard"

The Visual Components trend system

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".

Exercise: Online trend configuration for displaying the water temperature

A trend configuration should be created to display the water temperature over time.

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

Use the "Trend Wizard" to configure an online trend that records the process variable "gHeating.status.actTemp" for one minute with a sample rate of 100 milliseconds.

Trend Wizard Page1:

	Trend Type:	Online trend
	Channels:	1
	Mode:	Continuous
	Duration [s]:	60
	Sampling rate [ms]	100

Trend Wizard Page 2:

	Value data point:	gHeating.status.actTemp
	Foreground color:	Red
	MinValue:	0
	MaxValue:	100

Trend Wizard Page 3:

No changes necessary

Table 3: Configuration in the pages of the Trend Wizard

- 1) Insert a "Trend" control on the "TrendPage".
- 2) Start the Trend Wizard from the properties editor.
- 3) Make the configurations on the three pages of the Trend Wizard.



A trend configuration is inserted on the page.

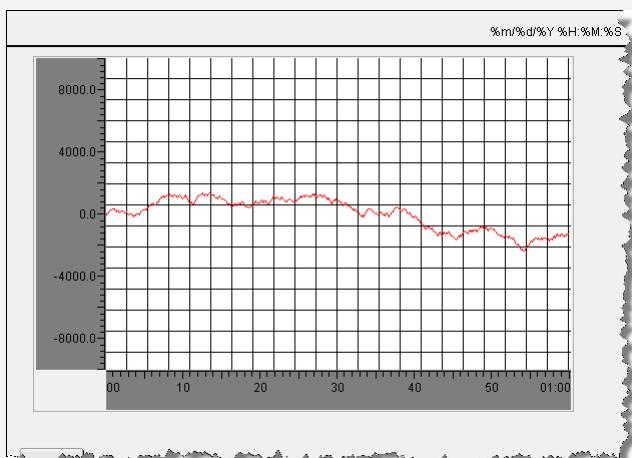


Figure 38: Result after finishing the "Trend Wizard"

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



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.

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

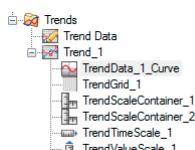


Figure 39: Trend components

Trend data

Trend data includes all of the process variables that are recorded by the trend server:

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

The Visual Components trend system

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

newTrend

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

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

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

3.4.2 Trend data

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

Index	Name	Typ
0	TrendData_1	Online

Figure 40: 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.

Exercise: Configure value limits for the 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".



All objects a user creates in a visualization project are given a default name that 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.



The limits are applied immediately to the "Trend" control on the "TrendPage". This ensures that the temperature curve will be displayed correctly during runtime.

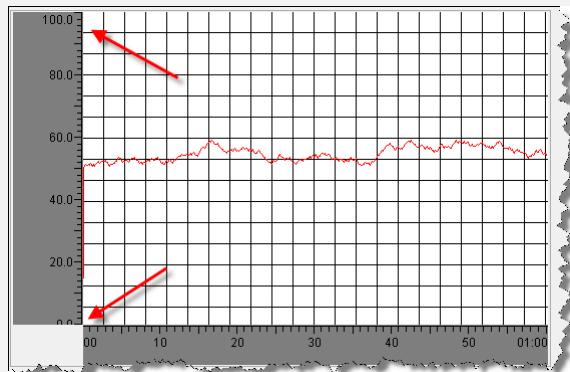


Figure 41: Correct configuration of limits

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.



[Visualization resources \ Trends \ Configuring trend data](#)

3.4.3 Trend container

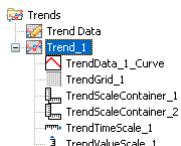


Figure 42: Trend container

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 from the trend object heading. The trend object's controls are listed under this heading.

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

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.

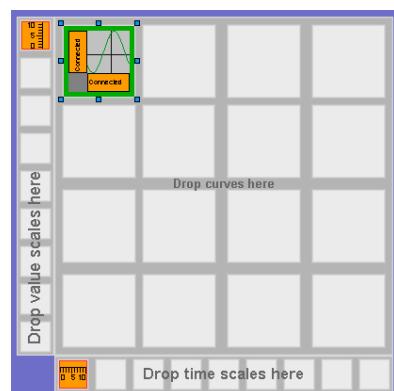


Figure 43: Trend object editor = Trend container

Labels for the axes for the two scales are configured in the next step.

The Visual Components trend system

Exercise: Adjust the background color for the x and y axis

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.

- 1) Open the trend container by double clicking on the trend object.
- 2) Select the "TrendValueScale" and "TrendTimeScale" objects.
- 3) Change the Backcolor property in the appearance properties.



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

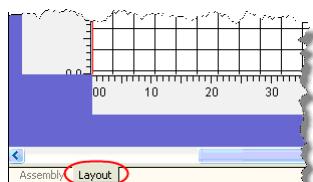


Figure 44: Trend container preview



[Visualization resources \ Trends \ Configuring trend objects](#)

[Visualization resources \ Trends \ Configuring trend objects](#)



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

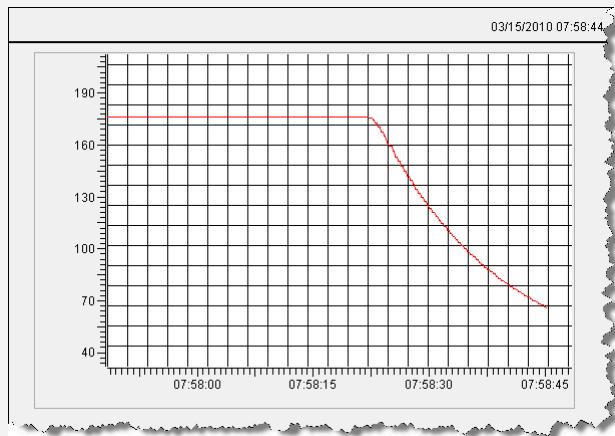


Figure 45: Modified background color of the scale axes

To conclude this task, we will add the temperature units ($^{\circ}\text{C}$ or $^{\circ}\text{F}$) to the value axis.

Exercise: Display the unit text on the value axis

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

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



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

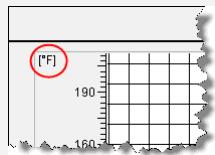


Figure 46: Display units text



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 in the program "visCtrl".

Exercise: 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".



The "visCtrl" program runs in a 200 ms 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.



Figure 47: Program "visCtrl" in the software configuration

The Visual Components trend system



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 under "Data Sources / Local" 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 (under "TrendData").

Exercise: Add the process variable to the trend data

Under "Trend Data", 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.

- 1) Open the editor for "Trend Data".
- 2) Insert a new trend object with the name "JigSaw".
- 3) The type of the trend object must be "Online", which is recorded continuously (Continuous mode).
- 4) The duration of the recording is determined by the "SampleRate" and "SampleCount" settings.
- 5) 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.



Figure 48: 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.

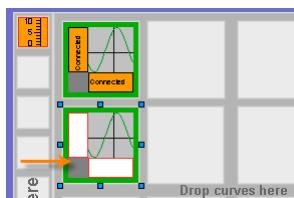


Figure 49: Insert a trend curve in the trend container

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

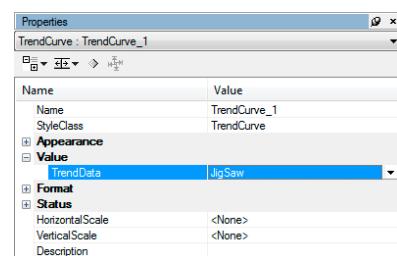


Figure 50: Connecting trend data

For the trend curve to be referenced 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.

Perform this step for both axis labels.

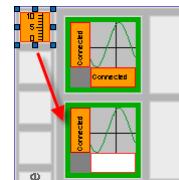


Figure 51: Assign scaling axis

Exercise: 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.



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 data type REAL, which doesn't permit automatic scaling.



Figure 52: Displaying two curves in the trend control

The Visual Components trend system



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.

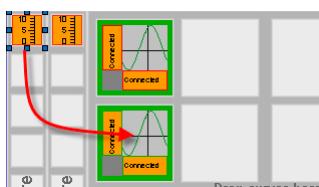


Figure 53: Connecting the TrendValueScale control

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

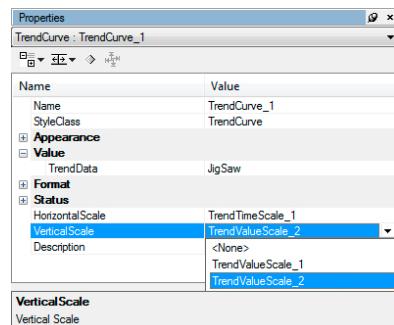


Figure 54: Changing trend scales

Exercise: 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.



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".

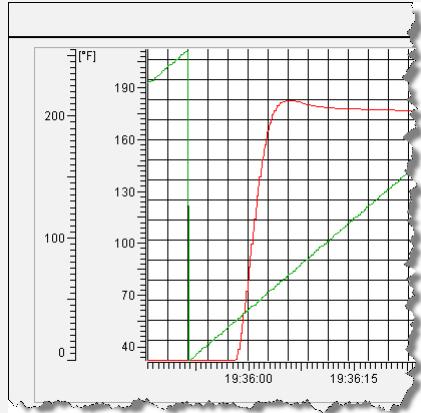


Figure 55: Curve with second scale



Control reference \ TrendScaleContainer

The Visual Components trend system

3.5 Interacting with the trend system

The trend system is operated using key actions or 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.

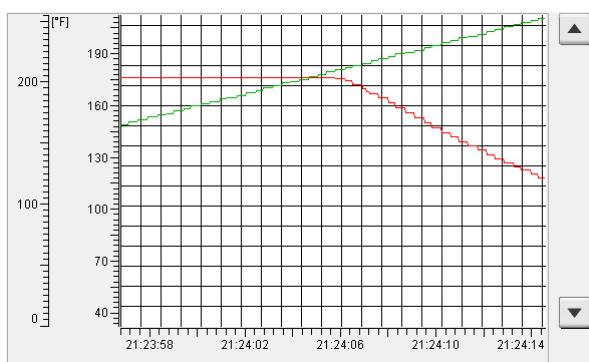


Figure 56: Changing the time axis

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

Exercise: 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, changing the displayed time range (current) horizontally.

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

Table 4: Example of the "Action" property for decreasing the size of the time range

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



A factor smaller than 1 reduces the displayed range, while a factor greater than 1 increases it. If the displayed time range is larger than the duration of recording for the trend data (SampleRate * SampleCount), no data is displayed outside the recorded duration.

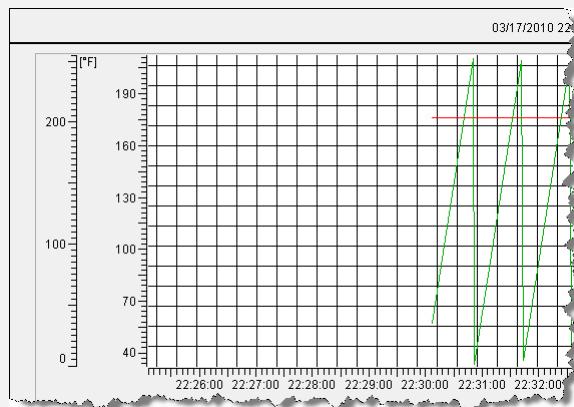


Figure 57: Zooming the time range

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.

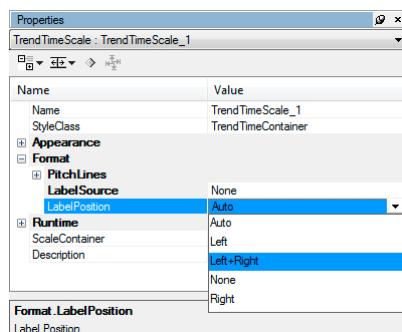


Figure 58: Configuring the time format for the time axis

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



FAQ \ Trend system

- [Key action reference \ Zoom](#)
- [Control reference \ Trend](#)
- [Control reference \ TrendValueScale](#)
- [Control reference \ TrendTimeScale](#)
- [Control reference \ TrendCurve](#)

Diagnostics using the System Diagnostics Manager (SDM)

4 DIAGNOSTICS USING THE SYSTEM DIAGNOSTICS MANAGER (SDM)

Automation Runtime can diagnose the controller using the **System Diagnostics Manager (SDM)** integrated in the operating system.

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

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

The System Diagnostics Manager can be used in any Visual Components application that is based on an SG4 target system.



On the "PC-based Simulation Runtime" system, SDM pages can be accessed via Visual Components.

This training module will only show the procedure for configuring SDM in Visual Components. With Automation Studio, these tasks can be completed using an SG4 target system.



Diagnostics and service \ Diagnostics tools \ System Diagnostics Manager

Diagnostics using the System Diagnostics Manager (SDM)

4.1 Task definition

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

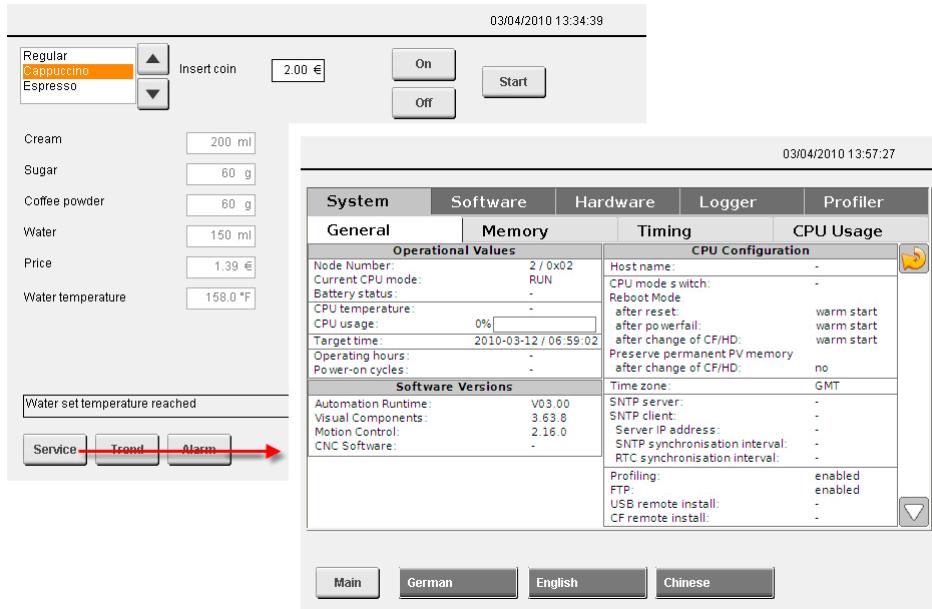


Figure 59: Example program "System Diagnostics Manager"

In this example, the System Diagnostics Manager will be enabled in Automation Runtime and displayed on the "ServicePage".

The following tasks will be performed in this training module:

- Enabling the System Diagnostics Manager in the Automation Runtime configuration
- Displaying the SDM on the "ServicePage"



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

Diagnostics using the System Diagnostics Manager (SDM)

4.2 Enabling the System Diagnostics Manager

The SDM is enabled in the Physical View by right-clicking on the CPU and selecting <Open AR Configuration> from the shortcut menu.

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

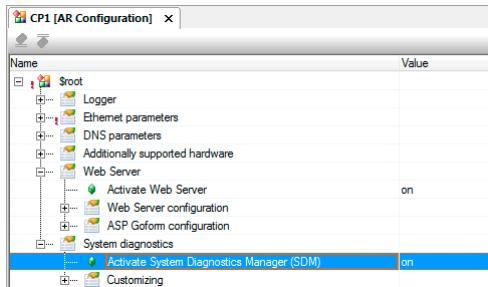


Figure 60: Enabling SDM in the current configuration

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 ARsim Runtime Simulation" system.

URL	Description
http://localhost/sdm-vga	Accesses HTML pages in VGA resolution
http://localhost/SDM	Accesses SVG pages

Table 5: URLs for accessing the HTML pages



If the browser uses a proxy server, the IP addresses for accessing SDM must be bypassed.

When accessing the "PC-based ARsim Runtime Simulation" system, the IP addresses "127.0.0.1" and "Localhost" need to be set as exceptions for the proxy server.



Diagnostics and service \ Diagnostics tools \ 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 a text group:

- Language-dependent URL possible

Advantage of using a 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.
- Defining a border is not permitted (Border = <None>).

Diagnostics using the System Diagnostics Manager (SDM)

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

Table 6: Supported display sizes

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

Exercise: 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.

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



It is important to ensure that Index 0 in the text group is reserved for specifying a file device if one exists.

If a file device is not being used, then Index 0 is ignored and the URL is entered at Index 1.



The HTML-based pages of the System Diagnostics Manager are displayed on the "ServicePage".

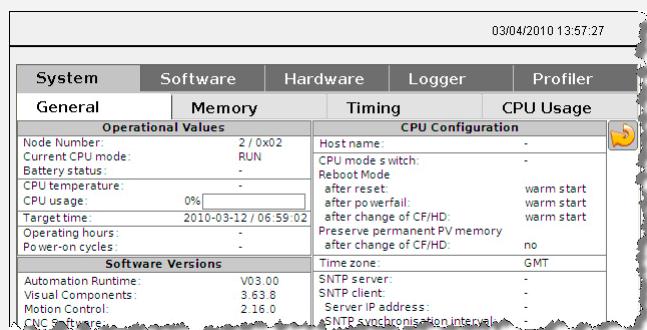


Figure 61: Displaying the SDM pages in the "HTMLView" control

Diagnostics using the System Diagnostics Manager (SDM)

4.4 Interacting 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.

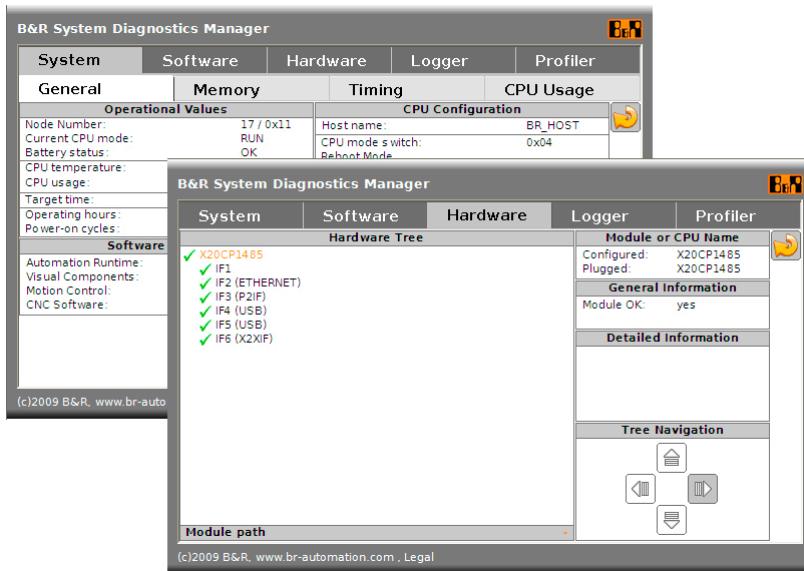


Figure 62: Image navigation in the System Diagnostics Manager

5 SUMMARY

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



Figure 63: Visual Components - Alarms, trends and diagnostics

With the freely configurable alarm properties and the various 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 diagnostic functions for every visualization application.

Training modules

TRAINING MODULES

- TM210 – Working with Automation Studio
- TM213 – Automation Runtime
- TM223 – Automation Studio Diagnostics
- TM230 – Structured Software Development
- TM240 – Ladder Diagram (LD)
- TM241 – Function Block Diagram (FBD)
- TM242 – Sequential Function Chart (SFC)
- TM246 – Structured Text (ST)
- TM250 – Memory Management and Data Storage
- TM400 – Introduction to Motion Control
- TM410 – Working with Integrated Motion Control
- TM440 – Motion Control: Basic Functions
- TM441 – Motion Control: Multi-axis Functions
- TM450 – ACOPOS Control Concept and Adjustment
- TM460 – Initial Commissioning of Motors
- TM500 – Introduction to Integrated Safety
- TM510 – Working with SafeDESIGNER
- TM540 – Integrated Safe Motion Control
- TM600 – Introduction to Visualization
- TM610 – Working with Integrated Visualization
- TM630 – Visualization Programming Guide
- TM640 – Alarm System, Trends and Diagnostics
- TM670 – Advanced Visual Components
- TM800 – APROL System Concept
- TM811 – APROL Runtime System
- TM812 – APROL Operator Management
- TM813 – APROL XML Queries and Audit Trail
- TM830 – APROL Project Engineering
- TM890 – The Basics of LINUX
- TM920 – Diagnostics and service
- TM923 – Diagnostics and Service with Automation Studio
- TM950 – POWERLINK Configuration and Diagnostics

- TM1010 – B&R CNC System (ARNC0)
- TM1110 – Integrated Motion Control (Axis Groups)
- TM1111 – Integrated Motion Control (Path Controlled Movements)
- TM261 – Closed Loop Control with LOOPCONR
- TM280 – Condition Monitoring for Vibration Measurement
- TM480 – The Basics of Hydraulics
- TM481 – Valve-based Hydraulic Drives
- TM482 – Hydraulic Servo Pump Drives

V1.2.1.1 ©2015/02/02 by B&R. All rights reserved.
All registered trademarks are the property of their respective owners.
We reserve the right to make technical changes.



TM640TRE.30-ENG