Kollmorgen Visualization Builder
Quick Start Guide

Edition F, April 2017
Kollmorgen Visualization Builder

Keep all manuals as a product component during the life span of the product.
Pass all manuals to future users/owners of the product.
Record of Document Revisions

<table>
<thead>
<tr>
<th>Revision</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>A, 08/2010</td>
<td>Preliminary edition</td>
</tr>
<tr>
<td>B, 06/2013</td>
<td>Updated for software revision 2.0-SP1</td>
</tr>
<tr>
<td>C, 04/2014</td>
<td>Updated for software revision 2.10</td>
</tr>
<tr>
<td>D, 11/2015</td>
<td>Updated for software revision 2.20</td>
</tr>
<tr>
<td>F, 04/2017</td>
<td>Updated for software revision 2.20 SP2</td>
</tr>
</tbody>
</table>

IMPORTANT NOTICE

Technical changes which improve the performance of the device may be made without prior notice!

Printed in the United States of America. This document is the intellectual property of Kollmorgen™. All rights reserved. No part of this work may be reproduced in any form (by photocopying, microfilm or any other method) or stored, processed, copied or distributed by electronic means without the written permission of Kollmorgen™.
# Contents

1 Introduction

1.1 Controller ................................. 6
  1.1.1 Tags ................................. 6

1.2 Manual Structure ......................... 6

2 Installation and Start

2.1 System Requirements ..................... 7
  2.1.1 Kollmorgen Visualization Builder 7
  2.1.2 Kollmorgen Visualizer RT .......... 7
  2.1.3 Special Requirements for Some Objects 9
  2.1.4 Starting Kollmorgen Visualization Builder from the Kollmorgen Automation Suite 9
  2.1.5 Help .................................. 9

3 New Project

3.1 Creating a New Project (KVB Standalone) 10
3.2 Creating a New Project (KAS IDE) ........... 10

3.3 Desktop Area ............................... 12
  3.3.1 Project Explorer ..................... 13
  3.3.2 Ribbon Groups and Controls ........ 13
  3.3.3 Quick Access Toolbar ................. 13
  3.3.4 Mini Toolbar and Context Menu ...... 13

4 Controller Tags

4.1 Importing Tags ............................. 14

4.2 Adding Tags ............................... 14

4.3 Saving the Project ....................... 15

5 Editing Objects

5.1 Adding Objects ............................. 16
  5.1.1 Meter ................................. 16
  5.1.2 Slider ................................. 16
  5.1.3 Align ................................. 17
  5.1.4 Resize ................................. 17
  5.1.5 Changing Appearance ............... 17

5.2 Running Project Test ...................... 17

6 Navigation and Screen Jumps

6.1 Screen Navigation ....................... 18
  6.1.1 Navigation Manager .................. 18
  6.1.2 Adding a Screen ..................... 18
  6.1.3 Screen Jump .......................... 19

6.2 Background and Foreground Screen ....... 19
  6.2.1 Adding a Screen ..................... 19
  6.2.2 Background and Foreground Screen 19

6.3 Running Screen Navigation Test ........... 21

7 Trend Viewer

7.1 Adding a Real-Time Trend Viewer ........... 22
  7.1.1 Curves ............................... 22

7.2 Running Real-Time Trend Viewer Test ...... 23

7.3 Trend Viewer History ..................... 23
  7.3.1 Actions .............................. 23

7.4 Running Historical Trend Viewer Test ...... 23

8 Alarm Management

8.1 Alarm Indicator ............................ 24

8.2 Alarm Server .............................. 25
  8.2.1 Alarm Groups ....................... 25
  8.2.2 Alarm Items ......................... 25
8.3 Alarm Viewer ................................................................. 27
8.4 Running Alarm Test ...................................................... 27

9 Recipes
9.1 Creating Recipe Tags .................................................... 28
  9.1.1 Adding a Screen .................................................... 28
  9.1.2 Adjusting Navigation Buttons ................................. 29
9.2 New Objects ................................................................. 30
  9.2.1 Show Info ............................................................... 30
9.3 Recipe Items ................................................................. 30
9.4 Saving a Recipe ............................................................ 30
9.5 Loading a Recipe .......................................................... 31
9.6 Recipe Data ................................................................. 31
9.7 Running Recipe Test ...................................................... 32

10 Dynamics
10.1 Creating an Object ....................................................... 33
10.2 Resizing an Object ....................................................... 34
10.3 Coloring an Object ..................................................... 35
10.4 Running Dynamics Test ............................................... 35

11 Script
11.1 Adding Objects .......................................................... 36
11.2 Script Tab ................................................................. 36
11.3 Running Script .......................................................... 37

12 Internal Tags
12.1 Adding Internal Tags ................................................... 38
  12.1.1 Area Tag ............................................................. 38
  12.1.2 Calculation Tag .................................................. 38
12.2 Creating an Analog Numeric ....................................... 39
  12.2.1 Calculation Trigger ............................................. 39
12.3 Running Internal Tags Test .......................................... 40

13 Object Browser
13.1 Adding a Graphical Element ....................................... 41
13.2 Using the Object Browser ........................................... 41

14 Multiple Texts
14.1 Configuring Texts ....................................................... 42
14.2 Running Multiple Texts Test ........................................ 43

15 Security
15.1 Security Configuration ............................................... 44
  15.1.1 Security Groups .................................................. 44
  15.1.2 Users ............................................................... 44
15.2 Login Behavior on Access Denied ............................... 45
15.3 Creating a Logout Button .......................................... 45
15.4 Object Security ........................................................ 46
  15.4.1 Administrators .................................................. 46
  15.4.2 Users ............................................................... 46
15.5 Running Security Test ............................................... 46

16 Function Keys
16.1 Defining Function Key Actions .................................... 47
  16.1.1 Show Screen ..................................................... 47
  16.1.2 Security ........................................................... 48
  16.1.3 Setting Controller Tag Values .............................. 48
  16.1.4 Recipe ............................................................. 48
  16.1.5 Setting Time Zone, Region and Daylight Saving .... 48
16.2 Defining Function Key Scripts .................................... 49
16.2.1 Area Calculation .................................................................49
16.2.2 Acknowledge All Alarms ....................................................49
16.3 Running Function Key Test ....................................................49

17 Cross Reference
17.1 Using the Cross Reference Tool ..........................................51

18 Troubleshooting ........................................................................52

19 Appendix ..................................................................................64
1 INTRODUCTION

The Kollmorgen Visualization Builder software is used to configure Advanced Kollmorgen Interfaces (AKI) and PC operated control applications, including applications for IPCs (Industrial PCs) from Kollmorgen. Kollmorgen Visualization Builder contains all basic functions needed in an application. The functions are tested and developed with customer needs and preferences in focus. Pre-defined objects in Kollmorgen Visualization Builder can be used to create complete process images, providing an overview of a complex application. You can customize the pre-defined objects or create objects of your own.

Note: When developing applications for a panel-based controller or an independent Kollmorgen HMI panel, communication channels are automatically configured when KVB is launched through the Kollmorgen Automation Suite™ Integrated Development Environment (IDE).

1.1 Controller

Kollmorgen AKI series of HMI panels are optimized for use with Kollmorgen controllers. Ongoing use of the term controller implies any Kollmorgen controller or variant.

1.1.1 Tags

Data values in a controller are referred to as tags. Tags may also belong to the system or be internal. A tag has a symbolic name and can be of different data types. Objects connected to tags can change values in the controller, and tag values can be reflected by changing object appearance in various ways. Objects in a screen will remain static until connected to a tag.

1.2 Manual Structure

The Quick Start Guide is based on an example project that makes it easier to start using Kollmorgen Visualization Builder. If the instructions in the example are followed carefully, this should result in a functional project that can be further developed, or used for inspiration. The target of the example is a PC, but all functions works similarly for all supported operator panels. Detailed information about Kollmorgen Visualization Builder is available in the help file, displayed by pressing F1 while using the software. The instructions in the Quick Start Guide are more detailed in the beginning. As the example progresses and you become familiar with Kollmorgen Visualization Builder, instructions for tasks that are of a repetitive nature may be brief or omitted.
2 INSTALLATION AND START

Kollmorgen Visualization Builder is installed on a development PC, where projects are developed, designed and compiled. The project is then run in an HMI panel or controller to observe and control a controller (or a group of controllers).

2.1 System Requirements

2.1.1 Kollmorgen Visualization Builder

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>2 GB</td>
</tr>
<tr>
<td>Processor</td>
<td>2 GHz or higher</td>
</tr>
<tr>
<td>Operating system</td>
<td>Microsoft Windows 10</td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows 8.1</td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows 7 SP1</td>
</tr>
<tr>
<td>Graphics card</td>
<td>Tier 2:</td>
</tr>
<tr>
<td>DirectX version</td>
<td>Must be greater than or equal to 9.0.</td>
</tr>
<tr>
<td>Video RAM</td>
<td>Must be greater than or equal to 120 MB.</td>
</tr>
<tr>
<td>Pixel shader</td>
<td>Version level must be greater than or equal to 2.0.</td>
</tr>
<tr>
<td>Vertex shader</td>
<td>Version level must be greater than or equal to 2.0.</td>
</tr>
<tr>
<td>Multitexture units</td>
<td>Number of units must be greater than or equal to 4.</td>
</tr>
</tbody>
</table>

2.1.2 Kollmorgen Visualizer RT

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>1 GB</td>
</tr>
<tr>
<td>Processor</td>
<td>1.3 GHz or higher</td>
</tr>
<tr>
<td>Operating system</td>
<td>Microsoft Windows 10</td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows 8.1</td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows 7 SP1</td>
</tr>
<tr>
<td>Graphics card</td>
<td>Tier 2:</td>
</tr>
<tr>
<td>DirectX version</td>
<td>Must be greater than or equal to 9.0.</td>
</tr>
<tr>
<td>Video RAM</td>
<td>Must be greater than or equal to 120 MB.</td>
</tr>
<tr>
<td>Pixel shader</td>
<td>Version level must be greater than or equal to 2.0.</td>
</tr>
<tr>
<td>Vertex shader</td>
<td>Version level must be greater than or equal to 2.0.</td>
</tr>
<tr>
<td>Multitexture units</td>
<td>Number of units must be greater than or equal to 4.</td>
</tr>
</tbody>
</table>
Note:
When user interface applications are running on a Kollmorgen panel-based controller, ensure the USB hardware key, AKC-HMI-RK-xxx, is installed.
2.1.3 Special Requirements for Some Objects

For some objects to be included in the Kollmorgen Visualization Builder project, specific software versions are required. Simulation of the project on the development PC may also be limited for some targets.

<table>
<thead>
<tr>
<th>Object</th>
<th>Minimum requirement</th>
<th>Simulation on PC target</th>
<th>Simulation on panel target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Media Player</td>
<td>Windows Media Player 10</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td>PDF Viewer</td>
<td>Acrobat Reader 9</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td>Web Browser</td>
<td>Internet Explorer 7</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
</tbody>
</table>

2.1.4 Starting Kollmorgen Visualization Builder from the Kollmorgen Automation Suite

To start Kollmorgen Visualization Builder (KVB):

1. Right-click on System in the Kollmorgen Automation Suite IDE Project Explorer.
2. Select Add HMI Device.
3. Right-click on the HMI Device to be added and select Add KVB Panel.
4. Double-click on the newly added Panel to start Kollmorgen Visualization Builder.

For more information, see the Kollmorgen Automation Suite IDE Reference Manual.

2.1.5 Help

Help topics are provided when pressing F1 while Kollmorgen Visualization Builder is running.
3 NEW PROJECT

Objective:
• Creating a new project.
• Getting familiar with the tool windows and the layout of the desktop area.

3.1 Creating a New Project (KVB Standalone)

1. Start Kollmorgen Visualization Builder and select Create New Project.
   This starts a wizard that will guide you through creating the project.
2. Choose a PC with a 1024 × 768 resolution as target for the application. Click Next.
3. Select DEMO in the list of controllers. Click Next.
   The demo controller, including regular tags (data containers) and counters, is used to design and
test a project directly on the development PC without connection to an external controller.
4. Give the project a name. For this tutorial use DEMO_TEST. Check that the suggested location
   is appropriate. If not, click Browse to select another location.
5. Click Finish.
   Project files can be stored anywhere in the computer environment where you have write access rights.
The project opens automatically.

Note:
When a separate Kollmorgen HMI panel is used in conjunction with your controller, the Modbus/TCP address is
loaded automatically if Kollmorgen Visualization Builder is launched from the Kollmorgen Automation Suite IDE.

3.2 Creating a New Project (KAS IDE)

To create a new Visualization Builder project from within the KAS IDE:

1. Launch the KAS IDE from the Windows Start menu:
   Start/Programs/Kollmorgen/Kollmorgen Automation Suite/IDE.
2. Click the Create a New Project button or press Ctrl+N.
3. From the Controller Creation wizard, select the target you would like to create a Visualization
   Builder project for and click Next.
4. Select the controller by selecting brand and protocol) you would like to create a Visualization
   Builder project for and click Next.
5. Select the location of your project and click Finish.
   The Kollmorgen Visualization Builder application launches and the controller panel is now ready
   for design.

To create a Visualization Builder project for an HMI:

1. From the KAS IDE Project Explorer, right-click on System and select Add HMI Device.
2. From the New HMI Device dialog, choose the HMI device you would like to create and click OK.
   The HMI device is added in the Project Explorer.
3. The KVB panel is added for the HMI device in the Project Explorer.
4. Double-click on **KVBProject** from the Project Explorer.
5. Select **Yes** to open with Kollmorgen Visualization Builder.
   The Kollmorgen Visualization Builder application launches and the HMI panel is now ready for design.
### 3.3 Desktop Area

The desktop area displays screens and configuration pages for project components such as controllers and functions. The desktop area shows only one screen or component at a time. When multiple screens or components are opened a row of tabs are shown in the upper part of the desktop area. Clicking on a tab activates its contents for editing.

If there are more tabs open than can be displayed, navigation arrows in the upper part of the desktop area can be used to scroll between them.

<table>
<thead>
<tr>
<th>Indication in picture</th>
<th>Desktop area component</th>
<th>Described in section</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Project Explorer</td>
<td>Project Explorer</td>
</tr>
<tr>
<td>B</td>
<td>Ribbon tabs</td>
<td>Ribbon Groups and Controls</td>
</tr>
<tr>
<td>C</td>
<td>Control groups</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Controls</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Quick Access Toolbar</td>
<td>Quick Access Toolbar</td>
</tr>
<tr>
<td>F</td>
<td>Mini toolbar</td>
<td>Mini Toolbar and Context Menu</td>
</tr>
<tr>
<td>G</td>
<td>ContextMenu</td>
<td></td>
</tr>
</tbody>
</table>
3.3.1 Project Explorer

When a new project is opened, an empty screen is active in the desktop area. The Project Explorer is docked to the left.

3.3.2 Ribbon Groups and Controls

Ribbon tabs are located in the top section of the tool window. Each ribbon tab holds one or several control groups. Each group contains a set of related controls. You use the controls to design screens, and to make settings for objects and controls in the project.

If you are not used to software with ribbon tabs, please spend a minute to get familiar with the ribbon concept.

3.3.3 Quick Access Toolbar

The Quick Access toolbar is always visible at the top of the desktop area. It contains the commands Save, Undo, Redo, Run and Simulate as Kollmorgen Visualization Builder is started.

3.3.4 Mini Toolbar and Context Menu

When right-clicking objects in Kollmorgen Visualization Builder, a mini toolbar and a context menu are displayed. The mini toolbar contains commands that are specific for Kollmorgen Visualization Builder, for example connect objects to controller tags. The context menu contains regular Microsoft application commands such as Copy, Paste etc.
4 CONTROLLER TAGS

Objective:

• Defining a tag list for the project.
• Saving the project.

4.1 Importing Tags

When Kollmorgen Visualization Builder is launched through the Kollmorgen Automation Suite IDE, all selected PLC variables are automatically imported.

4.2 Adding Tags

Objects connected to tags can change values in the controller, and tag values can be reflected by changing object appearance in various ways. Objects in a screen will remain static until connected to a tag.

1. Click on Tags in the Project Explorer.
   The tags configuration page opens on the desktop. There is one element in the tags list by default.

2. Click on the first field (Name) in the first row.
   A cursor is placed next to the text Tag1.

   The selection moves to the next field (Data Type under Tags). You do not need to change the data type.

Note:
The data type for the tag can be used as a presentation format in some situations; for example to show the correct engineering unit when using scaling. DEFAULT follows the selection made for the Data Type under Controllers.

   The selection moves to the next field (Access Right). You do not need to change the access rights now.

5. Press [TAB] on the keyboard again.
   The selection moves to next field (Controller Data Type under Controllers). You do not need to change the data type now.

   The selection moves to next field (Controller 1).

7. Type D0 in the Controller 1 field.
   The entries in the Controllers column correspond to tags in the selected controller. There are predefined tags in the DEMO controller that can be accessed by their respective tag address, e.g. D0 denotes an integer tag field.

8. Press [TAB] until the two first rows are completely filled in. Type D1 for the second controller tag.
   Some fields are automatically filled and when needed, incremented. The data type is automatically changed depending on what you type in the Controllers column.
9. Rename Tag1 to MeterAndSlider.

---

**Note:**

The **Name** of a tag is an identifier for the tag and can be any alphanumeric string, beginning with a letter (a-z, A-Z).

---

The D0 tag will be used in the next section to control and observe a controller tag value in a screen.

### 4.3 Saving the Project

1. Click on the **Save** symbol in the **Quick Access Toolbar**.
   
The project will be saved in the location that you selected when creating the project.
5 EDITING OBJECTS

Objective:

• Inserting a slider and a meter.
• Learning how to format and align objects.
• Testing the project; controlling and observing a controller tag value with objects in a screen.

5.1 Adding Objects

5.1.1 Meter

1. Click on the Screen1 tab in the desktop area and make sure that the Home ribbon tab in the ribbon area is selected.
2. Select a circular meter from the Objects group. Place it somewhere in the upper left section of the screen.
3. Drag a corner handle to resize the meter to a suitable size where the meter needle and the scale are clearly visible.
4. Right-click on the meter and expand the Select Tag... dropdown list.
5. Select MeterAndSlider and then click OK.

5.1.2 Slider

1. Select a slider from the Objects group. You may need to expand the Objects group by clicking the small arrow at the lower right in order to select the slider.
2. Place the slider just below the circular meter on the screen.
3. Right-click on the slider and expand the Select Tag... dropdown list.
4. Select MeterAndSlider and then click OK.
5.1.3 Align

An object that is dragged snaps into position relative to other objects.

1. Slowly drag the slider up and down.
   Notice that the slider snaps into position at a short distance below the meter.
2. Slowly drag the slider to the left and to the right.
   Notice that the slider snaps into position and that snap lines appear when the slider is aligned with the meter.
3. Arrange the slider in a position closely below the meter and with its left edge aligned with the left edge of the meter.

5.1.4 Resize

1. Make a multiple selection of the two objects (point at an empty area in the screen and drag diagonally across the objects).
   A multiple selection (group) has one primary object. The primary object has an orange frame; the other objects have blue frames. When enforcing format commands on the group the primary object is used as a template.
   If the meter is not the primary object then:
2. Click on it to change the primary selection of the group to the meter.
   Now adjust the width of the objects in the group:
3. Click on the Arrange control, located in the Format group of the Home ribbon tab, and select Make Same Width.

5.1.5 Changing Appearance

1. Select the slider on Screen1.
2. Click on the Quick Styles control in the Format group and select a new color style.
3. Click the small arrow at the lower right of the Format group in order to make additional settings for outline, shadow/fill effects etc.
4. Select the meter on Screen1.
5. Select the General ribbon tab and locate the Style group. Try the different pre-defined styles and evaluate which style suits your preferences the best.

5.2 Running Project Test

The project can be compiled and run at almost any time. This allows you to test your design continuously in an iterative manner.

1. Click on the Run icon in the Quick Access Toolbar.
   The project is now validated, and when no errors are found, the project will be compiled and executes in the development environment.
2. Drag the slider handle back and forth.
   Since both objects are connected to the same tag, the meter needle follows as you change the value of the slider control.
3. Close the Run window.
6 NAVIGATION AND SCREEN JUMPS

An Kollmorgen Visualization Builder project consists of screens with objects, usually connected to controller tags. All screens have the same basic functions. A screen can be given specific properties to specialize its behavior in the project:

- **Startup Screen**: The startup screen is the first screen that is displayed in runtime. By default, Screen1 is used as startup screen, but any screen can be designated startup screen by right-clicking on the screen and selecting Set as Startup Screen.
- **Background and Foreground Screen**: Any screen, except screens with aliases, can be used as a background or foreground screen by the other screens in the project. For more information about aliases, please refer to the Kollmorgen Visualization Builder help file.
- **Screen Template**: A screen that is saved as a screen template can be used not only in the current Kollmorgen Visualization Builder project, but also in all future projects.

Screen jumps are made with actions that can be assigned to e.g. buttons. When using the Navigation Manager to add screens and create links between screens, buttons are added automatically in the upper left corner of the screen that the link originates from.

**Objective:**
- Adding new screens and setting up screen jumps with buttons.

6.1 Screen Navigation

6.1.1 Navigation Manager

- Click on the View ribbon tab in the ribbon area. Click on Navigation Manager. The Navigation Manager opens in the desktop.

6.1.2 Adding a Screen

- Point at Screen1. Click and drag a connection from Screen1 to anywhere in the Navigation Manager area. A new screen appears (Screen2). A button labeled Screen2 appears in the upper left corner of Screen1.
6.1.3 Screen Jump

- Click and drag a connection from **Screen2** to **Screen1** in the **Navigation Manager**. A button labeled **Screen1** appears in the upper left corner of **Screen2**.

6.2 Background and Foreground Screen

6.2.1 Adding a Screen

- Click on **Screen** on the **Insert** ribbon tab. A new screen (**Screen3**) is created in the project, and opens for editing.

6.2.2 Background and Foreground Screen

1. Make sure that **Screen3** is open for editing on the desktop.
2. Select a **Button** from the **Objects** group (located on the **Home** ribbon tab), and place it in the lower left area of **Screen3**. Label the button **Start Screen**.
3. Keep the button selected and click on the **Actions** ribbon tab. Select **Show Start Screen**, located in the **Screen** group, from the drop-down list for the **Click** action.

4. Open **Screen2** for editing by clicking on it in the **Project Explorer**.
5. Select the **Home** ribbon tab.
6. Open the **Parent Screen**... dropdown list.
7. Click **Screen3**.
8. Select **Background**.
9. Click **OK**.
10. Try to change location of the Start Screen button in **Screen2**. This is not possible. Notice that changes made to **Screen3** are reflected in **Screen2**.

There are now two navigation facilities from **Screen2** to **Screen1** (the start screen).

Setting a foreground screen is done by the same procedure, except that you select **Foreground** instead at step 8.
6.3 Running Screen Navigation Test

1. Run the project.
2. Verify that each of the buttons in Screen2 performs a jump to Screen1.
   Since no screen has been set to be startup screen, Screen1 remains startup screen for this project.
3. Close the Run window.
7 TREND VIEWER

The trend viewer function stores register information from the controller in the operator panel. Real-time trend viewer as well as historical trend viewer is available.

Objective:
- Adding a trend viewer object with two curves.

7.1 Adding a Real-Time Trend Viewer

1. Open Screen1 for editing in the desktop.
2. Select the Trend Viewer object from the Objects group and place it on the screen.

7.1.1 Curves

1. Click on Tags in the Project Explorer.
2. Click Add to add a new tag.
3. Type Counter in the Name field and connect it to C0 under Controller1. C0 is a counter that counts from 0 to 100 and back to 0 with a frequency of 1 Hz.
4. Open Screen1, make sure that the trend viewer object is selected, and click Edit Curves on the General ribbon tab.
5. Click Add to add a new curve.
6. Select MeterAndSlider in the Tag field, to connect the curve to the same tag that you used for the slider in Screen1.
7. Click Add to add a second curve.
8. Select Counter in the Tag field, and select another color for this curve.

9. Click OK.
7.2 Running Real-Time Trend Viewer Test

- Run the project and check that both curves are visible in the trend viewer. Test that Curve 1 changes with the slider.

7.3 Trend Viewer History

7.3.1 Actions

1. Open Screen1 for editing in the desktop.
2. Click the Home ribbon tab.
3. Place a button to the left of the trend viewer object. Label the button Trend Viewer History.
4. Keep the button selected and click on the Actions ribbon tab.
5. Click the small arrow at the bottom right of the Click group.
6. Select Click in the Trigger dropdown list.
7. Click the Select Action... dropdown list, expand Trend Viewer and select Trend Viewer History.
8. Click the Select Trend Viewer... dropdown list and select TrendViewer.
9. Select On in the last dropdown list.
10. Click OK.
11. Select the trend viewer object and click on the Actions ribbon tab.
12. Click the small arrow at the bottom right of the Mouse Down group.
13. Select Mouse Down in the Trigger dropdown list.
14. Click the Select Action... dropdown list, expand Trend Viewer and select Trend Viewer History.
15. Click the Select Trend Viewer... dropdown list and select TrendViewer.
16. Select Off in the last dropdown list.
17. Click OK.

7.4 Running Historical Trend Viewer Test

1. Run the project.

2. Test that you can switch to the historical trend viewer with the Trend Viewer History button.
3. Go back to real-time trend viewer by clicking on the trend viewer object.
8 ALARM MANAGEMENT

Alarms are used to make the operator aware of events that require immediate action. An alarm is triggered when a certain condition is met. An alarm condition is designed as a logical evaluation of a tag value. Alarms can be divided into groups to create an order of priority.

Objective:

• Configuring the alarm list and designing an alarm object.

8.1 Alarm Indicator

When an alarm goes active, the alarm indicator becomes visible to alert the operator, regardless of which screen is active.

The appearance of the alarm indicator depends on the current alarm status.

Select General settings in alarm server properties page to decide for which alarm statuses to show the alarm indicator.

The alarm indicator will show the most severe alarm in the alarm list with the following indications:

• Flashing red when there is any active, unacknowledged alarm.
• Flashing green when no active alarms exist, but inactive unacknowledged alarms exist.
• Flashing green when there are only active acknowledged alarms.

The alarm indicator disappears when all alarms are both acknowledged and have returned to inactive status.
8.2 Alarm Server

- Click on Alarm Server in the Project Explorer to open the Alarm Server configuration page.

8.2.1 Alarm Groups

The Alarm Groups tab is used to set up multiple alarm groups, e.g. when a project needs separate management of alarms for independent functions. In this example, the default alarm group is used.

8.2.2 Alarm Items

1. Click the Alarm Items tab.
2. Click Add. A new tag named AlarmItem0 is added to the list.
3. Click the empty Text field and write Slider max value.
4. Select MeterAndSlider in the Tag field.
5. Select GreaterThan in the Condition field.
6. Change the value in the Trigger Value field to 99.
7. Click Add again. A new tag named AlarmItem1 is added to the list.
   This tag will be internal unless connected to a controller on the Tags configuration page, and using an internal tag works just fine for this example project. See chapter Internal Tags for information.
8. Click the empty Text field and write Boolean tag.
9. Click the Tag field and write BoolAlarmTag.
10. Select EqualTo in the Condition field.
11. Change the value in the Trigger Value field to 1.
12. Add two more alarm items according to the image below.

13. Leave the columns for History and Acknowledge Required checked. Leave the other fields as they are.
14. Make sure that all alarm tags can be controlled from the project screens or that they will be triggered by other mechanisms (the counter will trigger AlarmItem2 and AlarmItem3 after 10 and 20 seconds respectively).
15. Place a button to the left of the trend viewer object. Label the button Set Alarm.
16. Keep the button selected. On the **Actions** ribbon tab, select **Toggle Tag**, located in the **Tag** group, from the drop-down list for the **Click** action.

17. Select **BoolAlarmTag** in the **Select Tag** field.
8.3 Alarm Viewer

1. Open Screen2 for editing in the desktop.
2. Click the small arrow at the bottom left of the Objects group and select Alarm Viewer under HMI Controls.
3. Click on the screen to place an alarm object on it.
   The columns and button placement can be customized in an alarm object.
4. Select the alarm object on the screen, and click on the General ribbon tab.
5. In the Buttons group, select Top in the Position dropdown list.
6. Adjust the size so that all button controls in the alarm object are visible.
7. Click on Configure Columns in the Display Settings group to customize the alarm information and the order of the columns in the alarm object.

8.4 Running Alarm Test

1. Run the project.
2. Test to trigger the alarms.
3. Click the Ack All button and observe the alarm indicator.
4. Press Clear to remove all alarms of normal status (acknowledged and inactive).
9 **RECIPES**

Recipes are used to set or save a predefined group of tags in one operation.

**Objective:**
- Creating and using recipes to change multiple values.

### 9.1 Creating Recipe Tags

- Create a group of controller tags that the recipe should affect. Create three integer values according to the image below, for setting weight, length and width of an imaginary item.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Data Type</th>
<th>Access Right</th>
<th>Data Type</th>
<th>Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>MeterAndSlider</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>D0</td>
</tr>
<tr>
<td>Tag2</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>D1</td>
</tr>
<tr>
<td>Counter</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>C0</td>
</tr>
<tr>
<td>BoolAlarmTag</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>DEFAULT</td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>D2</td>
</tr>
<tr>
<td>Length</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>D3</td>
</tr>
<tr>
<td>Width</td>
<td>DEFAULT</td>
<td>ReadWrite</td>
<td>INT16</td>
<td>D4</td>
</tr>
</tbody>
</table>

#### 9.1.1 Adding a Screen

1. Click on the **View** ribbon ftab in the ribbon area. Click on **Navigation Manager**.
2. In the **Navigation Manager**, point to **Screen1** and drag a connection to an empty spot in the screen navigation area.

A new screen (**Screen4**) is created in the project.

3. Open **Screen4** and select the **Home** ribbon tab.
4. In the **Screen** group, click the **Paren Screen** dropdown list.
5. Select **Screen3** and then select **Background**.
6. Click **OK**.
   
   This enables navigation from **Screen4** to **Screen1**.

### 9.1.2 Adjusting Navigation Buttons

- Open **Screen1**.
- Select the button labeled **Screen4** (in the upper left corner) and move it so the button placed under it (**Screen2**) becomes fully visible.
9.2 New Objects

1. Open Screen4 for editing in the desktop.
2. Add a slider, found in the Objects group on the Home ribbon tab.
3. Right-click on the slider and open the Select Tag... dropdown list. to open the controller drop-down list.
4. Select Weight and click OK.
   This connects the Weight tag to the object.
5. Press Ctrl, then drag the slider across the screen to make a copy of it. Position the new slider and connect it to Length.
6. Repeat step 5, but connect the slider to Width.

9.2.1 Show Info

It is possible to show information about which tag each object is connected to, and if dynamics or actions are configured for the object. This is done by clicking on the Show/Hide Info button in the desktop area, or by using the keyboard shortcut Ctrl + D.

- Press Ctrl + D on the keyboard to check that the tags are correctly bound to the sliders in the screen.

9.3 Recipe Items

1. Click on Recipe on the Insert ribbon tab.
   The recipe configuration page opens in the desktop. The new recipe is also available from the Project Explorer.
2. On the Tag Configuration tab, click Add three times to add three new recipes.
3. Connect the items to tags according to the image below.

<table>
<thead>
<tr>
<th>Name</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecipeItem1</td>
<td>Weight</td>
</tr>
<tr>
<td>RecipeItem2</td>
<td>Length</td>
</tr>
<tr>
<td>RecipeItem3</td>
<td>Width</td>
</tr>
</tbody>
</table>

### 9.4 Saving a Recipe

1. Open **Screen4** for editing in the desktop.
2. Place a button next to the set of sliders for the recipe tags and label the button **Save Recipe**.
3. Keep the button selected and click on the **Actions** ribbon tab.
4. Open the **Select Action...** dropdown list for the **Click** action.
5. Select **Save Recipe**, located in the **Recipe** group.
6. Make sure that **Recipe1** is selected in the next dropdown list.
7. Leave **Select Recipe data (optional)...** empty.

### 9.5 Loading a Recipe

1. Open **Screen4** for editing in the desktop.
2. Press **Ctrl**, then drag the **Save Recipe** button across the screen to make a copy of it.
3. Change the label to **Load Recipe**.
4. Keep the button selected and click on the **Actions** ribbon tab.
5. Open the **Select Action...** dropdown list for the **Click** action.
6. Select **Load Recipe**, located in the **Recipe** group.
7. Leave **Select Recipe data (optional)...** empty.

### 9.6 Recipe Data

Create a predefined recipe by defining values on the **Runtime Data** tab of the recipe configuration page.

1. Open the recipe configuration page by clicking on **Recipe1** in the **Project Explorer**.
2. Click on the **Runtime Data** tab.
3. Click the **Add** button twice.
4. Change the values of the recipe items according to the image below.

<table>
<thead>
<tr>
<th>Runtime Recipe Title</th>
<th>RecipeItem1</th>
<th>RecipeItem2</th>
<th>RecipeItem3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book</td>
<td>2</td>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td>TV</td>
<td>30</td>
<td>45</td>
<td>60</td>
</tr>
</tbody>
</table>

5. Open **Screen4** for editing.
6. Place a new button next to the set of sliders. Label the button **Load Book**.
7. Keep the button selected and click on the Actions ribbon tab
8. Open the Select Action... dropdown list for the Click action.
9. Select Load Recipe, located in the Recipe group.
10. Open the Select Recipe data (optional)... dropdown list and select Book.

9.7 Running Recipe Test

1. Run the project.
2. Test to set the sliders to various values and save the values in recipes.
3. Test to load the recipes.
   Check that the sliders change according to the recipe values.
Dynamic object properties are used to move and resize objects based on controller tag values.

Objective:
- Changing size and color of an object based on tag value changes.

10.1 Creating an Object

1. Open Screen4 for editing.
2. On the Home ribbon tab, select the red rectangle which is located in the Objects group.
3. Draw a small square below the set of sliders.
10.2 Resizing an Object

1. Select the square.
2. On the Dynamics ribbon tab, click on Size which is located in the Layout group.
3. Select the Width tag for Width and the Length tag for Height.
4. Click OK.
5. Adjust the enlarged size of the square directly on the screen and note the change of values in the Edit Size Dynamics window.
10.3 Coloring an Object

1. Select the square.
2. On the Dynamics ribbon tab, click on Fill which is located in the Color group.
3. Assign the Weight tag in the Select Tag... drop-down list.
4. Click Add four times.
5. Adjust the tag values to change the color of the square depending on the Weight tag value. The example in the picture below uses fill color in combination with a gradient.

6. Click OK.

10.4 Running Dynamics Test

1. Run the project.
2. Test to change the tag values with the sliders and by loading recipes. Observe what happens with the size and color of the small square.
11 SCRIPT

Scripts are used to manage functionality for objects. Scripts are written in C#.

Objective:
• Inserting a button and text box.
• Writing a script for the button to affect the text in the text box.

11.1 Adding Objects
1. Open Screen2 for editing.
2. Click the Home ribbon tab.
3. Click the small arrow at the bottom right of the Objects group.
4. Select TextBox under Windows Controls.
5. Draw a TextBox on the screen.
6. Place a button on the screen and label it Write Test.

11.2 Script Tab
1. Select the Write Test button.
2. Click on the Script tab located in the lower left the desktop area.
   The view mode changes from Layout to Script.
3. Expand the Button2 node.
4. Double-click on the Click node.
   You can now start typing script code for the Click event for Button2.
   A context sensitive name completion feature (IntelliSense) can be activated during typing with Ctrl + [Spacebar] and it triggers automatically when a period (‘.’) is typed after a code element.
5. Type the following as the click event code:

```
TextBox1.Text = "Test";
```

C# code on the Script tab:

```csharp
public partial class Screen2
{
    void Button2_Click(System.Object sender, System.EventArgs e)
    {
        TextBox1.Text = "Test";
    }
}
```

11.3 Running Script

1. Run the project.
2. Click on the **Write Test** button and check that the text string assigned with the script code now appears in the text box.
12 INTERNAL TAGS

Internal tags can be used to calculate values that need not be represented in the controller, for example information only for the operator.

Objective:

• Writing a script to perform a calculation of the area using the length and width tags.
• Showing the result with an internal tag.

12.1 Adding Internal Tags

• Click on Tags in the Project Explorer.
The tags configuration page opens on the desktop.

12.1.1 Area Tag

1. Add a tag and label it Area.
2. Change the data type to FLOAT.

12.1.2 Calculation Tag

1. Add a tag and label it Calc.
2. Change the data type to BIT.

Leaving the Controller1 column empty keeps the tag internal, not connected to a controller.

3. Switch to Script view mode.
4. Expand the Calc node.
5. Double-click ValueChange.

To access data and methods in C# control code the keyword “Globals” is used. The example uses explicit type casting (“(double)”), which is necessary for an overloaded operator (multiplication).
6. Calculate the area in the ValueChange node:

\[
\text{Area} = (\text{double} \times \text{Width}) / 100;
\]

12.2 Creating an Analog Numeric

1. Open Screen4 for editing.
2. Click the Home ribbon tab.
3. Select the Analog Numeric object in the Objects group.
4. Draw an analog numeric object below the set of sliders, clear from the rectangle object.
5. Right-click on the analog numeric object and connect it to the Area tag.
6. Keep the analog numeric object selected and click the General ribbon tab.
7. In the Display Format group, select Decimal in the dropdown list.
8. Set Number of Decimals to 2.
9. Select a Text object from the Objects group, and place a text object on the screen.
10. Place a text object on the screen in connection to the analog numeric object, and name it Area:

12.2.1 Calculation Trigger

1. Open Screen4 for editing.
2. Select the rectangle object.
3. Go to the Actions ribbon tab.
4. In the Mouse Down group, open the Select Action... dropdown list.
5. Expand the Tag node and select Toggle Tag.
6. Open the Select Tag... dropdown list and select the Calc tag.
12.3  Running Internal Tags Test

1. Run the project.
2. Test to set the sliders to various values. Click on the dynamic rectangle area and observe the change of the analog numeric control.

![Sliders and Area](image-url)
13 OBJECT BROWSER

An overview of all objects included in a screen can be displayed in the **Object Browser**.

**Objective:**

- Positioning objects in depth
- Locking objects
- Making objects invisible

13.1 Adding a Graphical Element

1. Open **Screen4** for editing.
2. Click the **Home** ribbon tab.
3. Select the **Rectangle** from the **Objects** group.
4. Draw a rectangle to fit as background of the group of sliders and buttons.
   The rectangle now totally obscures the other objects.

13.2 Using the Object Browser

1. Select **Object Browser** from the **View** ribbon tab.
2. Select the rectangle on the screen.
3. Click the **Send object backward** arrow button until all the buttons and sliders are visible.
4. Click the **Lock** icon next to the selected rectangle object in the **Object Browser**.
5. Try to move the rectangle on the screen by dragging it.
   The object is locked and cannot be moved. It cannot be selected in any way.
6. Select one of the buttons and click the **Visibility** button.
   The button is hidden. But if the project is run in Kollmorgen Visualizer RT, the object will show as normal.
7. Click the **Lock** button for the rectangle again and delete the rectangle.
14 MULTIPLE TEXTS

Text objects can be used to display textual information, and can also reflect changes in controller tags.

Objective:
- Presenting a variant text message that reflects the changes of the calculated area.

14.1 Configuring Texts

1. Open Screen4 for editing.
2. Select the text object labeled Area.
3. Click on the General ribbon tab.
4. Click on Configure Texts in the Text group.
5. Connect the text to the Area tag.
6. Add text strings and edit the intervals according to the example below.

With the default setting for the text object, Autosize, there is no need to adjust the object in order to make the longest string fit in runtime.
14.2 Running Multiple Texts Test

1. Run the project.
2. Test to set the sliders to various values. Click on the dynamic rectangle and observe the change of the analog numeric control. Check that the text is updated as well.
15 SECURITY

Access to objects and actions in the project can be limited using security groups and user passwords.

Objective:
- Adding user names and setting passwords.
- Setting up login and logout control.
- Restricting access of recipe handling.

15.1 Security Configuration

- Click on Security in the Project Explorer to open the configuration page.

15.1.1 Security Groups

Security is handled by dividing users into security groups. These are configured on the Groups tab. In this example, the two default security groups, Administrators and Operators, are used.

15.1.2 Users

1. Select the Users tab on the Security configuration page.
2. Add a user and name it SuperUser.
3. Connect SuperUser to both Administrators and Operators under Groups.
4. Add another user, name it User1 and connect it to the Operators group.
5. Enter passwords for the users.

<table>
<thead>
<tr>
<th>Username</th>
<th>Password</th>
<th>Description</th>
<th>Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admin</td>
<td>Password</td>
<td></td>
<td>Administrators</td>
</tr>
<tr>
<td>SuperUser</td>
<td>Password</td>
<td></td>
<td>Administrators, Operators</td>
</tr>
<tr>
<td>User1</td>
<td>Password</td>
<td>copy</td>
<td>Operators</td>
</tr>
</tbody>
</table>

Note:
The password is converted to asterisks as you leave the password input cell.
15.2 Login Behavior on Access Denied

A login dialog can be displayed whenever a user tries to access an object that is restricted for the user group that the user belongs to.
1. Select the Users tab, and click the Settings button.
2. Select Show Login Dialog for action on access denied, and Normal for visibility.
3. Click OK.

15.3 Creating a Logout Button

1. Open Screen3 (the background screen) for editing.
2. Place a button next to the Start Screen button. Label the button Logout.
3. Keeping the button selected and click the Actions ribbon tab.
4. Open the Select Action... dropdown list in the Click group.
5. Expand the Security node and select Logout.
15.4 Object Security

1. Open Screen4 for editing.

15.4.1 Administrators

1. Right-click on the Save Recipe button and select Administrators for Select Security Groups.

15.4.2 Users

1. Right-click on the Load Recipe button and select Operators for Select Security Groups.

15.5 Running Security Test

1. Run the project.
2. Click the Screen4 button.
3. Test to make sure that it is no longer possible to load or save recipes without logging in, and that the login dialog opens when any of the buttons are pressed.
4. Login as Administrator and save a recipe.
5. Test to load a recipe.
   The login dialog opens.
6. Login as User1 and load a recipe.
7. Test to save a recipe.
   The login dialog opens.
8. Login as SuperUser. Test to save and load recipes.
9. Log out.
10. Test that it is no longer possible to load or save recipes.
16 FUNCTION KEYS

Function keys can be used to perform actions and execute scripts. This allows operator control of data and screen functionality independent of which screen is active.

Function keys can also be configured as local function keys, which means that they are applicable individual screens. In this example, global function keys are used.

Objective:

- Programming actions for function keys to change screen, set controller tag values, recipe management and display the login dialog.
- Programming function keys to execute scripts.

16.1 Defining Function Key Actions

1. Click on Function Keys in the Insert ribbon tab.

16.1.1 Show Screen

1. Click on ... under Action on the row for function key F1 to open the Properties dialog.
2. Click Select Trigger... and select Key Down.
3. Expand the Screen node and select Show Start Screen.
4. Click OK.
5. Click on the row for function key F2.
6. Click ... under Actions.
7. Click Select Trigger... and select Key Down.
8. Expand the Screen node and select Show Screen.
9. Select Screen2 in the next dropdown list.
10. Click OK.
11. Click on the row for function key F3.
12. Click ... under Actions.
13. Click Select Trigger... and select Key Down.
14. Expand the Screen node and select Show Screen.
15. Select Screen4 in the next dropdown list.
16. Click OK.
16.1.2 Security
1. Click on the row for function key F4.
2. Click ... under Actions.
3. Click Select Trigger... and select Key Down.
4. Expand the Security node and select Login.
5. Click OK.

16.1.3 Setting Controller Tag Values
1. Click on the row for function key F5.
2. Click ... under Actions.
3. Click Select Trigger... and select Key Down.
4. Expand the Tag node and select Set Analog.
5. Select the Weight tag from the Select Tag... dropdown list.
6. Specify the analog value 50.
7. Click OK.
8. Repeat the steps above to set up function keys F6 and F7 to control the Length and Width tags.

16.1.4 Recipe
- Set up function key F8 to load Recipe1, and function key F9 to save Recipe1. Leave Select Recipe data (optional)... empty.

16.1.5 Setting Time Zone, Region and Daylight Saving
- Set up function keys F10 to set time zone, region and daylight saving. The action is located in the Other group.
16.2 Defining Function Key Scripts

Function keys can also be used to trigger scripts.

16.2.1 Area Calculation

Program a function key with the area calculation for the rectangle object:

1. Click on the row for function key F11.
2. Select Script view mode by clicking on Script at the bottom of the screen.
3. Expand the F11 node, and double-click on its KeyDown node.
4. Calculate the area on the KeyDown event with this code:

```
Globals.Tags.Area.Value =
    (double) Globals.Tags.Length.Value *
    (double) Globals.Tags.Width.Value / 100;
```

Using a function key script eliminates the need for the separate trigger tag (Calc).

16.2.2 Acknowledge All Alarms

Program a function key that acknowledges all alarms:

1. Expand the F12 node, and double-click on its KeyDown node.
2. Type the following KeyDown event code:

```
Globals.AlarmServer.Acknowledge();
```

C# code in the script tab:

```csharp
public partial class FunctionKeys
{
    void F11_KeyDown(System.Object sender, System.EventArgs e)
    {
        Globals.Tags.Area.Value =
            (double) Globals.Tags.Length.Value *
            (double) Globals.Tags.Width.Value / 100;
    }

    void F12_KeyDown(System.Object sender, System.EventArgs e)
    {
        Globals.AlarmServer.Acknowledge();
    }
}
```

16.3 Running Function Key Test

1. Run the project.
2. Test that the defined functions keys (on the PC keyboard) perform the defined actions.
17 CROSS REFERENCE

The Cross Reference tool provides an overview of where a specific tag is used in the current project.

Objective:
- Locating all occurrences of a specific tag quickly.

17.1 Using the Cross Reference Tool

1. Click Tags in the Project Explorer.
2. If you are in script view mode, then click Design at the bottom.
3. Select the MeterAndSlider row.
4. Click the Cross Reference button.
   The Cross Reference tool is displayed.

5. Double-click on AlarmServer.Default.AlarmItem0 (value) in the Cross Reference list.
   The Alarm Server configuration page opens on the desktop.
6. Double-click on Screen1.Slider (Value) in the Cross Reference list.
   Screen1 opens on the desktop, and Slider1 is selected.
18 TROUBLESHOOTING

18.1 Troubleshooting KVB

This section describes tools to use in case of unexpected behavior in Kollmorgen Visualization Builder.

18.1.1 Trouble Installing Kollmorgen Visualization Builder on Windows 8.1

If you get an error message about .NET Framework 4.5.X when installing Kollmorgen Visualization Builder on Windows 8.1, then run Windows Update and try to install Kollmorgen Visualization Builder again.

18.1.2 Project Build Failure

If project build fails, check that the absolute path to the project, including its components, does not exceed 260 characters.

18.1.3 Back Up Project Issues

Make sure that project that is fetched using the Back Up command is identical to the current Kollmorgen Visualization Builder project; i.e. has the same name and is of the same size.

<table>
<thead>
<tr>
<th>Related Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information see KVB User Guide, Upload Database</td>
</tr>
</tbody>
</table>

18.1.4 Invalid Names

If using system device names when naming objects such as controllers, screens, recipe managers, data loggers and script modules, an error message will be displayed, stating that this is an invalid name. Examples of system device names: AUX, COM1, CON, LPT3, NUL, PRN.

<table>
<thead>
<tr>
<th>Related Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information see KVB User Guide, Name Group</td>
</tr>
</tbody>
</table>

18.1.5 Adobe Reader not Recognized

For the PDF Viewer object to recognize that Adobe Reader is installed in the development PC as well as in the target, the setting Display PDF in browser must be enabled in Adobe Reader.

1. Open Adobe Reader and select Edit/Preferences/Internet.
2. Check the Display PDF in browser box.

18.1.6 Expressions Cannot Be Found

If an error indicating that an expression cannot be found is generated at project validation, make sure that objects inserted from the Component library uses global expressions only.

<table>
<thead>
<tr>
<th>Related Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information see KVB User Guide, Upload Database</td>
</tr>
</tbody>
</table>
18.1.7 Performance Issues Related to Graphics

When designing Kollmorgen Visualization Builder projects, the CPU load may suddenly increase when performing actions such as moving objects on the screen or hovering over buttons, depending on which graphics card is used.

This problem may be solved by turning off the hardware acceleration of the graphics card. In Windows this is done under the advanced display settings on the Troubleshooting tab.

Note:
Should you experience problems with the graphic rendering, make sure the latest graphical driver is used. If the latest graphical driver is used and rendering issues still occur, upgrade to a better graphics card.

<table>
<thead>
<tr>
<th>Related Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information see KVB User Guide, System Requirements</td>
</tr>
</tbody>
</table>

18.1.8 Error Related to Third Party Controls

When opening a project including third party controls, the third party controls must be included in the Kollmorgen Visualization Builder third party control folder. Otherwise the project may terminate, or unexpected error messages may be displayed.

When referencing third-party assemblies (.dll files), make sure not to place two files with the same fully qualified assembly name in the same folder, otherwise errors may occur.

18.1.9 Performance Issue When Navigating in Script Tree View

If you are working with a large Kollmorgen Visualization Builder project, especially a project with many tags and alarms, you may find that navigating the script tree view is slow.

The performance can be improved by saving the project.

<table>
<thead>
<tr>
<th>Related Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information see KVB User Guide, Script</td>
</tr>
</tbody>
</table>

18.1.10 Performance Issue When Navigating in Script Tree View

If you are working with a large Kollmorgen Visualization Builder project, especially a project with many tags and alarms, you may find that navigating the script tree view is slow.

The performance can be improved by saving the project.

18.2 Troubleshooting Common System Issues Using an AKI Terminal with a PxMM

This section describes troubleshooting common problems with KVB and the AKI terminal with PxMM.
18.2.1 How Do I Get Started with KVB When It Is Used in A KAS System with A PxMM?

The KVB software has a separate install file, which is available from Kollmorgen.com. After installation on a PC, KVB can be opened from the KAS IDE (Integrated Development Environment) by double clicking on the KVBProject item in the Project tree once the KVB Project item has been added.

18.2.2 Should I Create A KVB Project at The Controller Level or At the System Level in The KAS IDE?

The difference between the options is that at the controller level, the brief setup in the IDE does not select a specific AKI panel. The AKI panel selection is done inside the KVB environment. When added at the System level you are prompted to select a particular AKI HMI model. The AKI model information is transferred to KVB when it is opened.

Note: Some New Project templates already include KVB projects in the controller section.

At the system level:

At the controller level:
18.2.3 To Make Variables Accessible to KVB I Selected the KVB Box in The IDE Dictionary, But It Never Made It Into KVB. Why?

Variables are transferred to KVB from the IDE at the point when KVB is opened from inside the IDE. The variables transferred are those selected (box checked) prior to the last .KAS project compiled by the IDE. Changes to the list, including additions, will transfer after the next compilation and opening the KVBProject from the IDE.

18.2.4 I Cannot Download the Program to The AKI HMI

1. If the .KVB project will not compile, review the error log at the bottom of KVB and try to correct the issues.
2. Check the AKI terminal to verify the address in the Target field is the same as set in the Terminal.
18.2.5 I Cannot Run a Project from Inside KVB With the KAS Simulator

This capability allows testing both the KVB screens and the KAS application code without hardware. Without the PxMM not all functionality may be able to be tested, yet for many applications basic code flow and operations can be tested.

To use this capability, make sure the KAS project has a controller address of 127.0.0.1 in the IDE. Clicking on the Simulator button in the IDE will adjust the IP address to this. The IP address is sent when KVB is opened (inside the IDE). If KVB is open when setting this address, close KVB and reopen it.

18.2.6 How Do I Check the IP Address of The AKI Terminal?

During power-up and before the loaded project is started, the IP address will appear on the bottom left corner of the AKI screen.

18.2.7 How Do I Set Up or Change the IP Address of The AKI Terminal?

Perform the following steps to enter the service menu:

1. Apply power to the panel.
2. When the hourglass appears, press a finger on the screen and hold for approximately 20 seconds.
3. If the service menu is password protected, you will be prompted for a pin code. Enter pin code.
4. The touch calibration screen will display the following message: “Tap anywhere on screen or touch calibrate will start in 10 seconds.”
5. Press finger once again to enter the service menu.

18.2.8 How Can I Find More Information on How to Add C# Coding to the KVB Project?

- Read the KVB Users Manual.
- Do a search for “C#” on KDN in the PxMM sections. This will give you information to start.
- If you need more detail, contact your Kollmorgen representative.

18.2.9 How Do I Set the PxMM Address In KVB?

The IP address set in the IDE is sent to KVB when it is opened.
The IP address is stored here in the KVB project:

18.2.10 What Are System Variables and How Do They Get Added To a .KVB Project?

System variables are local variables (or internal tags) in KVB that can be used in the C# language to get information that can then be sent back to the PxMM or used to trigger or change operation in the AKI.
18.2.11 How Can I Change the Modbus Update Rate for Communications Between the PxMM And AKI Terminal?

In KVB, go into the Poll Groups tab within the Tags section. Here the update rate of tags can be set by Groups with each Group settable down to 100 msec.

In the Tags tab of the Tag section variables can be linked to a polled group.

18.2.12 Can an AKI Terminal Be Connected to Multiple PxMMs?

Yes, additional addresses can be added

Then click on the Settings button, then the Stations tab
18.2.13 Can Modbus TCP/IP (To the PxMM) And Modbus RTU (To A Third-Party Device) Be Used and Be Functional at the Same Time?

Yes. The Modbus TCP/IP connects to the PxMM and the Modbus RTU connects to the third-party device. For setup of Modbus RTU communications:
http://kdn.kollmorgen.com/content/kas-modbus-rtu-communications-through-aki-terminal

18.2.14 What Are Common Ways to Load A KVB Project into the AKI Terminal?

- Use the Download button to load via a URL address
- Use the Export button to export to a memory stick, to then be inserted into the AKI

18.2.15 How Do I Setup Multiple HMI Panels to Communicate with One PDMM at the Same Time?

Multiple AKI HMI panels (Modbus Master) can talk to a PDMM (Modbus slave) at the same time. Each Modbus master via the .KVB project is set up to communicate to the same PxMM address.

18.2.16 In KVB, How Do I Check or Setup the .KVB Program to Download to A Particular AKI Model?

Go to Settings, then Display/Target.
19 APPENDIX

19.1 How Do I Start a KVB Project from Within KAS?

From within the KAS IDE, double-click on KVBProject in Project Tree. The project is found at the System or Controller level, depending on where you created it.

![KVBProject in Project Tree](image)

19.2 What is Imported from the KAS IDE When I Open KVB?

When you open KVB from within the KAS IDE, vital information is transferred. This includes:
- The IP Address
- Variables (Tags) that have been flagged as applicable to KVB. Please note that only those items which were selected at the last compile command will be sent to KVB.
- The selected AKI panel type if KVBProject was created at the system level.

19.3 How do I Designate the Variables in the KAS IDE That Will Be Mapped to KVB?

Transferring variables from KAS IDE to KVB is a simple process in the dictionary. This process assumes you have created all necessary variables.

1. Open the Variables list editor in the Dictionary toolbox.
2. For each required variable, select the KVB checkbox to tab that item.
Once the project is compiled, these tagged variables will automatically be mapped and transferred when you create the KVB project. This may be done at any time; should you add a new variable which needs to be transferred, simply tag it as “KVB”, re-compile your KAS project, and open the KVBProject.

19.4 How to Download a Project to an AKI Terminal

- Use the Download button to load via a URL address and an Ethernet connection.
- Use the Export button to export to a USB memory stick, to then be inserted into the AKI

19.5 Simulating or Testing Your Project with KVB and the KAS Simulator

Just as you can simulate your project in the KAS IDE and run it on the KAS Simulator, you can test your KVB-created HMI screens in KVB (not downloaded to a panel) connected to either the KAS Simulator or an actual PxMM controller. Following are instructions on how to achieve this. You can replicate this same example by creating a new KAS project using a 2 Axes FFLD PLCopen template. This template is one of several that have pre-defined KVB projects and HMI.

Following are two important points which must be set before you can use KVB to simulate a project. The template we selected already meets these criteria:

- The IP address is set to 127.0.0.1 (“home”); this is required to run with the KAS Simulator.
- This project type, in addition to having a pre-defined KVB project, has had the necessary variables selected to be transferred to KVB.

To proceed, use the following directions.

1. Click the Simulated Device button.
2. Click the Compile button.
3. Launch KAS Simulator.
4. Click the Connect button.
   At this point you can simulate within KAS and use the built-in control panel.
5. Click the Run button. This will load the axes into the KAS Simulator, they are light blue which means they are Simulated, Power Off.
6. Double-click the **KVBProject** item in the Project Tree to launch KVB, and click **Yes** to open the project.

7. From KVB’s **Project** menu, click **Run**. This will open a virtual HMI panel.

   **Note:** The **Simulate** button renders the KVB screens, but it cannot be used to connect to a simulation or actual controller.

8. Click the green **Enable** button. This will activate the Ready light and the axes in the Simulator are light green, which means they are **Simulated, Power On**.

9. Click the **Show Geared Move Commands** button to switch screens.
10. Click the **Run/Stop** toggle switch to spin the axes. From here you may adjust the axes speeds and increments as you wish. You are simulating your project, including the HMI.
About KOLLMORGEN

Kollmorgen is a leading provider of motion systems and components for machine builders. Through worldclass knowledge in motion, industry-leading quality and deep expertise in linking and integrating standard and custom products, Kollmorgen delivers breakthrough solutions that are unmatched in performance, reliability and ease-of-use, giving machine builders an irrefutable marketplace advantage.

Join the Kollmorgen Development Network for product support. Ask the community questions, search the knowledge base for answers, get downloads, and suggest improvements.

North America
KOLLMORGEN
203A West Rock Road
Radford, VA 24141 USA
Web: www.kollmorgen.com
Mail: support@kollmorgen.com
Tel.: +1-540-633-3545
Fax: +1-540-639-4162

Europe
KOLLMORGEN Europe GmbH
Pempelfurtstraße 1
40880 Ratingen, Germany
Web: www.kollmorgen.com
Mail: technik@kollmorgen.com
Tel.: +49-2102-9394-0
Fax: +49-2102-9394-3155

South America
KOLLMORGEN
Avenida Tamboré - 1077 Tamboré - Barueri — SP Brasil
CEP: 06460–000, Brazil
Web: www.kollmorgen.com
Mail: support@kollmorgen.com

China and SEA
KOLLMORGEN
Room 202, Building 3, Lane 168
Lin Hong Road, Changning District
Shanghai, China
Web: www.kollmorgen.cn
Mail: sales.china@kollmorgen.com