Rockwell Automation
Process HMI Style Guide

White Paper

Rockwell Automation

Table of Contents

1 Introduction

The Connected Enterprise revolves, in part, around analyzing manufacturing data and combining it with business data to create enterprise intelligence. This starts with focusing on the plant floor by presenting machine operators with the information they need to effectively and efficiently keep production moving.

Providing the right information and context to operations can aid in the ability to detect and respond to abnormal situations as well as simplify common tasks. A properly designed HMI can decrease downtime and scrap and improve product quality and productivity. Providing operators the information they need in the right context enables the best decision-making. For example:

- Displays designed using a specific color palette help the operator identify the most important information that may require immediate attention, such as alarms.
- Additional context on how critical parameters are changing, and whether they are within a desirable range, helps operators make better decisions.

This white paper provides guidelines for HMI design and implementation that are aligned with the industry standard; and, while it applies to general HMI development, it was written with FactoryTalk View SE and PlantPAx System applications in mind. This complements publication PROCES-WP016 *(Human Machine Interfaces for Distributed Control Systems)* which covers important principles for designing HMI based on the industry standard ANSI/ISA-101.01-2015 *(Human Machine Interfaces for Process Automation).*

ISA 101.01 defines specifics of the HMI design process: an HMI philosophy, HMI style guide, and HMI toolkit.

- The HMI philosophy provides independent or platform-specific guiding principles for HMI design at your plant.
- The HMI style guide uses the guiding principles and concepts that are defined by the HMI philosophy to provide implementation and guidance.
- The HMI toolkit includes platform-specific graphical systems and HMI elements that can be used to implement the HMI style guide.

This white paper can assist you in the implementation of ISA 101.01 in your application by providing reusable guidelines that follow standards as a starting point for your own HMI Style Guide. This can be further simplified by leveraging the Rockwell Automation Library of Process Objects as your HMI toolkit for implementation.

An editable version of this document is available on the Rockwell Automation KnowledgeBase (answer ID 1086840). From there, the document can be downloaded and edited to incorporate specifications of your own HMI application.

1.1 Before You Begin

Document your operational needs and goals in an HMI philosophy document and familiarize yourself with the following HMI and Alarm Management standards:

- ANSI/ISA-101.01-2015 Human Machine Interfaces for Process Automation
- ANSI/ISA-18.2-2016 Management of Alarm Systems for the Process Industries

1.2 Additional Resources

The following documents contain additional information on this topic or related products.

Resource	Description
Human Machine Interfaces for Distributed Control Systems, publication PROCES-WP016	Provides an overview of the concepts of good HMI design as defined by ANSI/ISA-101.01
Rockwell Automation Library of Process Objects, publication PROCES-RM002	Provides information on the Library of Process Objects, which can be used as an HMI toolkit to assist with deployment of the style guidelines provided by this document
Rockwell Automation Library of Process Objects: Display Elements, publication PROCES-RM014	Provides descriptions of the HMI visualization files provided with the Library of Process Objects
PlantPAx Distributed Control System Application Configuration, publication PROCES-UM003	Provides the steps necessary to start development of a PlantPAx Distributed Control system, including steps for deploying an HMI application template aligned with this style guideline in FactoryTalk View SE
FactoryTalk View Site Edition User Guide, publication VIEWSE-UM006	Provides details of how to use the FactoryTalk View SE software for developing and running HMI applications
ISO 9241-210:2010—Ergonomics of human-system interaction: Human-centered design for interactive systems	ISO standard that provides requirements and recommendations for human-centered design principles and activities throughout lifecycle of computer-based interactive systems
EEMUA Publication 201: Process plant control desks utilizing human-computer interfaces	EEMUA (Engineering Equipment and Materials Users Association) publication that provides guidance on designing Human Computer Interface systems for people operating industrial processes and activities
Effective Console Operator HMI Design, ASM Consortium Guidelines.	ASM (Abnormal Situation Management) guideline on HMI design process
The High Performance HMI Handbook, by Hollifield, Oliver, Nimmo, and Habibi	Book that contains useful information on HMI design, implementation, and maintenance

2 Functional Description of HMI Components

An HMI consists of many different components that come together to provide an interface for users to monitor and manipulate a process or machine:

- Display Hierarchy: How data is organized across displays and different levels of displays.
- Display Layout: How information and visuals are laid out on the display.
- Display Navigation: Method of navigating between displays.
- Display Content: Static and dynamic visuals contained in an HMI such as numerical values, pump, valves, tanks, etc.
- Alarm Depiction and Management: Presentation of alarms and how users can manage alarms.
- Security: Access control and providing the right content to the right people.
- Display Performance: How quickly display responds to initial call-up, data change, and users' interaction.

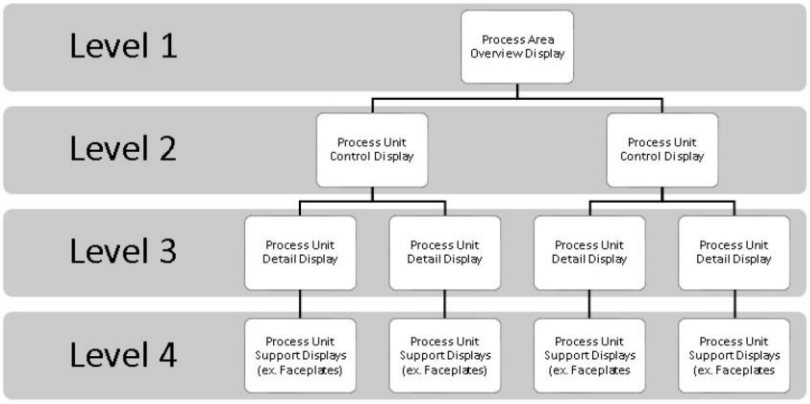
The rest of this document provides details relating to each of these components of an HMI.

3 Display Levels

Understanding the users' goals, tasks, and mental model is crucial to determining the organization of the displays in an HMI project. They should be organized for the primary user as identified in user research. In most cases this is the operator. Secondary users need to be considered as well, but the information they need can be provided on separate displays or workstations.

The hierarchy and organization of the displays should be created to provide progressive disclosure of information. A clean, simple display with an overview of the operations should lead to other displays that contain more complexity and detail as users navigate deeper into the hierarchy. Using this methodology provides a quick look and allows the user to initiate the action of diving deeper for more information rather than having it clutter the initial display.

There are four levels that are recommended for the display hierarchy, each level providing more detail than the previous level.



Level 1: Overview Display

Provides an overview of the operator's entire span of responsibility.

Level 2: Process Unit Control Display

Operator's primary operating display. Used during normal operations, routine changes, and monitoring.

Level 3: Process Unit Detail Display

Non-routine operations. Provides sufficient information to facilitate process diagnostics.

Level 4: Process Unit Support Display

Interlocks, Diagnostics, Help, and Documentation; delivered on faceplates or popup displays.

3.1 Level 1 Displays: Process Overview

Level 1 displays contain high-level overview information that can be assimilated quickly, provide clear indication of current performance, and highlight anything that requires immediate attention. Control should not be performed from this display.

Example Level 1 display

Level 1 displays contain the following types of elements:

- High-level Key Performance Indicators (KPI)
- Alarms of top 2 or 3 highest priorities
- Important calculated parameters and conditions
- Important information from upstream and downstream units
- Advanced control mechanisms performance and status
- Major equipment status
- Appropriate trends of important parameters
- Indications of abnormal situations, denoting severity

There should only be one overview display for a specific operating position; however, there may be different ones for different modes or process changes such as batch.

Level 1 displays are crucial as they provide contextual information; however, they may not contain all information users need to perform their jobs. Instead, they provide current state of operations, indications of abnormal situations that may be occurring, and quick and easy access to additional information.

Level 1 displays should be designed secondarily to Level 2 displays.

3.2 Level 2 Displays: Process Unit Operating Graphics

Level 2 displays are the primary displays used for operators to perform their tasks and should be designed first. Level 2 displays should match the users' mental model of the machine and operation and provide easy access to related displays in the display hierarchy. There may be multiple Level 2 displays for the same equipment to cover specific situations such as startup, normal operation, state or product transitions, and shutdown.

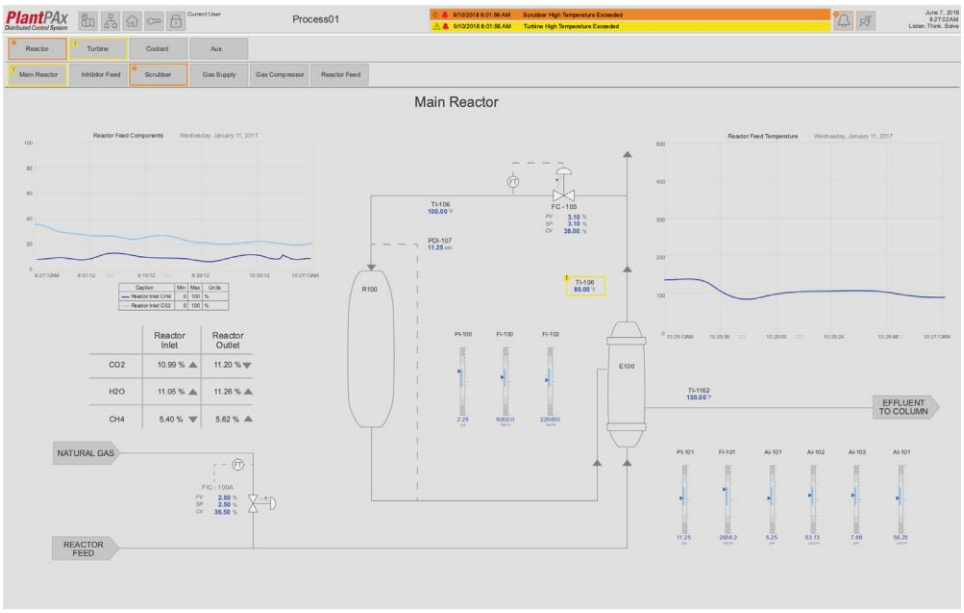
Example Level 2 display

Level 2 displays contain the following types of elements that are relevant to the tasks to be accomplished by that display:

- Key Performance Indicators (KPI)
- All alarms relevant to this display (if constrained by space, then alarms of top 2 or 3 highest priorities with indication there are additional alarms not being displayed)
- Controls needed to accomplish tasks (or access to controls, such as easy access to faceplates that contain controls)
- Indicators needed to accomplish tasks
- Navigation to related displays
- Navigation to overview display
- Navigation for continuation of flow lines
- Indications of abnormal situations, denoting severity

3.3 Level 3 Displays: Process Detail Displays

Level 3 displays contain much more detail and controls. They contain detailed view of sub-units, individual equipment items, components, and their related controls and indications. They are used for detailed investigations and interventions and are intended primarily for troubleshooting or manipulating items not accessible from Level 2 displays.



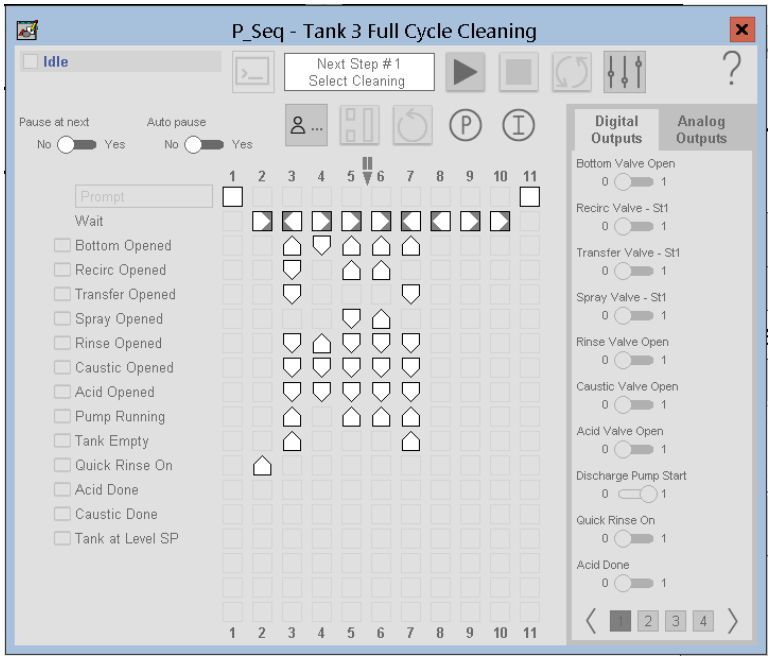
Example Level 3 display

Level 3 displays contain the following types of elements that are relevant to the tasks to be accomplished by that display:

- Alarms of all priorities relevant to that display
- Controls
- Indicators
- Detail view of equipment

3.4 Level 4 Displays: Process Support and Diagnostics Displays

Level 4 displays provide the most detail of subsystems, individual sensors, or components.



Example Level 4 display

Examples include:

- Alarm displays with details of individual sensor status
- Detailed info about equipment and instrumentation
- Detailed status of Advanced Process Control functionality
- System-supplied displays such as point detail, system diagnostics, alarm summary, etc.
- Help displays
- Operating procedures
- Alarm documentation and response guidance

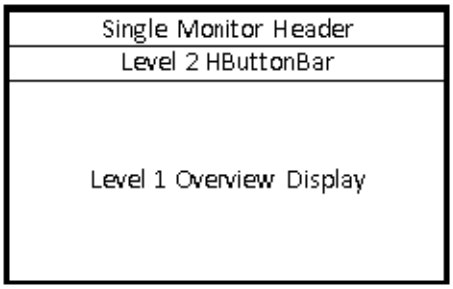
4 Standard HMI Template

Rockwell Automation provides a configurable PlantPAx HMI template that can serve as a starting point for developing a new project. Templates are applied in Studio 5000 Architect and used in FactoryTalk View SE to build an HMI application. Multiple monitors are supported in the template, allowing you to implement displays on a single- or four-monitor workstation.

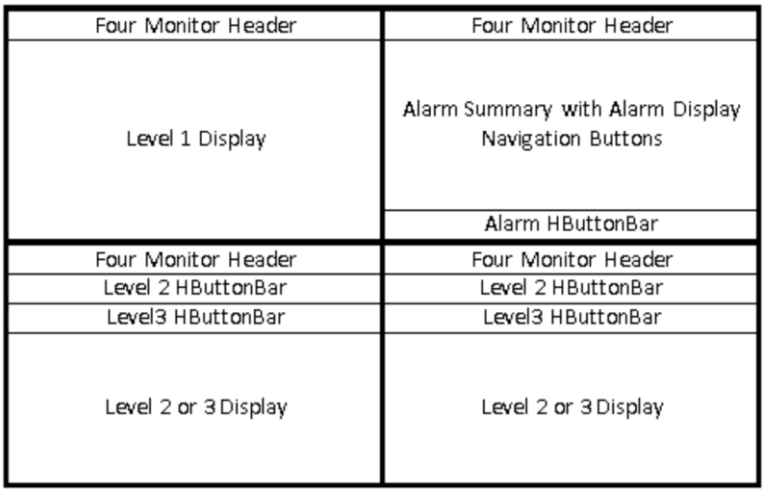
For more information on this template, visit publication PROCES-UM003 *(PlantPAx Distributed Control System Application Configuration)* or download from the Rockwell Automation Product Compatibility and Download Center (PCDC).

4.1 Layout

The single monitor HMI template configuration is below. Users view, control, and navigate from one display.



The following image is of a four monitor HMI template configuration.

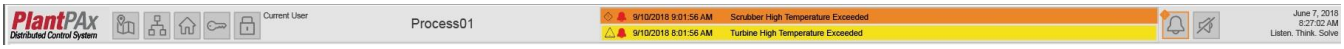


A four monitor widescreen layout with 1920x1080 screen resolution is utilized in the standard template. Level 1, Level 2, and Level 3 displays are provided as a part of the template along with navigation objects that promote display invocation from one monitor to another. Alarms and trends can be filtered as a part of the display yoking when a process area is changed from the navigation on the Level 1 display.

The HMI template requires that displays and alarms be organized using the Display Levels outlined in Section 3 of this document. For each display level, the HMI template provides not only the display framework, but also global objects that can be used. Navigation menus and headers for both one- and four-monitor configurations are also provided.

4.1.1 Header

Headers contain functionality that provides access to information. The HMI template includes headers for both a single monitor client and a four-monitor client.



Single Monitor Header

This header includes the following components:

- Display Navigation Map
- System Status
- Return to Home Screen Button
- Client Login/Logout Buttons
- Alarm and Event Banner
- Alarm Access
- Alarm Silence

Four Monitor Header

The four monitor header includes the same buttons as the single monitor header except for the alarm access button. Another difference is in the lack of an alarm and event banner. Instead, the alarm summary is continuously displayed on monitor 2. Lastly, the four monitor header includes a "Refresh all Monitors" button instead of the "Return to Home Screen" button.

4.1.2 Button Bars

The level 2 Button Bar is used for navigating through level 2 displays.

The level 3 Button Bar is used for navigating through level 3 displays.

The button bars change based on context. For example:

- If users change the Process Area, then Level 2 buttons will display the Level 2 displays for that Process Area
- If users change a Level 2 display, it will display buttons for Level 3 displays relevant to that Level 2

For a single-monitor configuration:

- Level 2 button bar appears on Level 1, Level 2, and Level 3 displays
- Level 3 button bar appears on Level 2 and Level 3 displays.

For a four-monitor configuration:

- Level 2 button bar only appears on Level 2 and Level 3 displays
- Level 3 button bar also appears on Level 2 and Level 3 displays.

4.1.3 Level 1 Display

The Level 1 display in the HMI template is used as an Overview Display as described in 3.1 of this document:

- For a single monitor configuration, the Level 1 display contains a header and Level 2 buttons
- For a four-monitor configuration, the Level 1 display contains only a header.

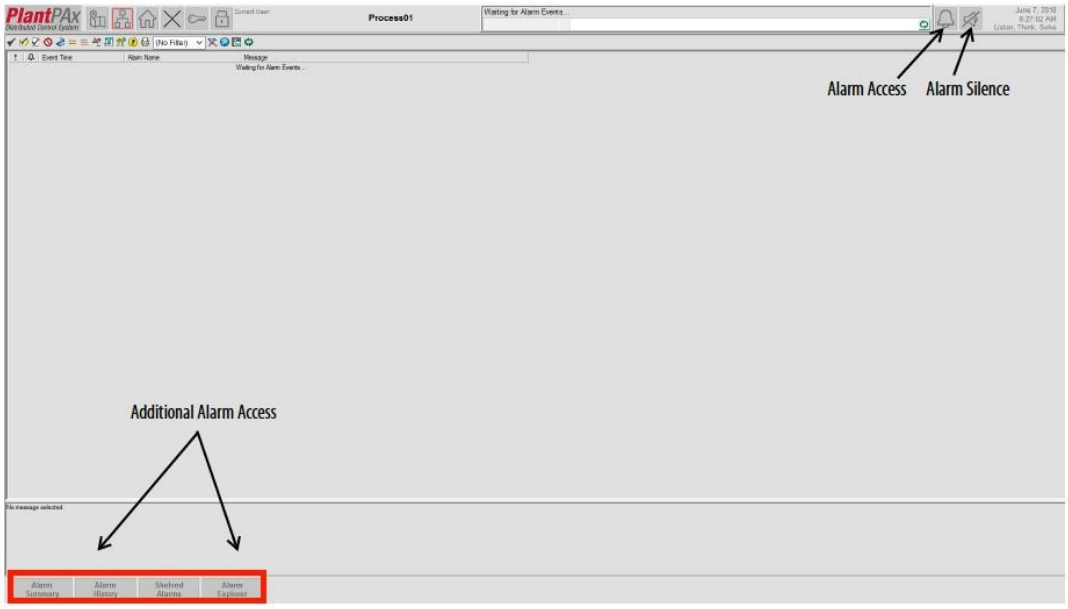
4.1.4 Level 2 and 3 Displays

The HMI template display can be used for Level 2 and Level 3 displays. Level 2 displays are the main displays to perform tasks as described in 3.2 of this document. Level 3 displays are used when additional detail or controls are required as described in 3.3 of this document.

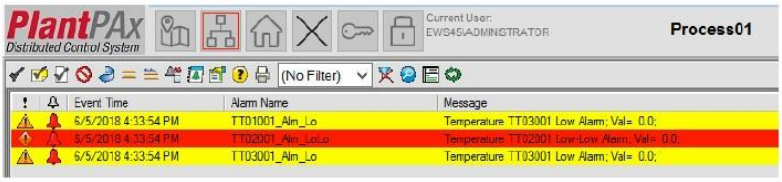
- For a four-monitor configuration, monitors 3 and 4 display the Level 2 and Level 3 displays.
- For both single monitor and four-monitor configurations, Level 2 and Level 3 displays contain a header, Level 2 button bar, and Level 3 button bar for navigation.

4.1.5 Alarm Summary

The alarm summary displays alarms for the selected Process Area. If users change the Process Area using the Display Navigation Map, the alarm summary will update to show relevant alarms.



Below is an example of alarms in the Alarm Summary.



At the bottom of the alarm summary are navigation buttons to additional alarm screens.

- For single-monitor configuration, the Alarm Summary is accessed using the bell icon button in the header
- For four-monitor configuration, the alarm summary is continuously displayed on monitor 2

4.2 HMI Instance Configuration

Each HMI instance should have its own client file. A client file defines the characteristics of how the client runs including startup displays, macros that will run at startup, whether a Windows title bar is displayed, and other configuration properties. Each instance of the HMI template deployed in a distributed system operates independently of all other template deployments.

4.3 Standard System Functionality

The design of the template provides the operator with an effective and efficient interface to control and monitor the process facility or area. Alarm grouping and naming, which facilitates effective communication of alarms in a logical manner, is described later in this document. Navigation objects are included that provide the yoking functionality that minimizes the number of clicks to perform operations and simplifies the navigation workflow for operators.

5 Operator Interaction Methodologies

When designing an HMI, it is important to consider the method in which users interact with the application:

- Touch (touch tool such as touch pen, hand, gloved hand, etc.)
- Keyboard
- Mouse or trackball

The Library of Process Objects natively supports all of these interaction methods. Key factors to keep in mind are the interaction methods that affect the size of the hit zones. For example:

- Touch requires a larger hit zone and spacing between hit zones so there is not accidental activation of objects.
- For keyboard and mouse, hit zones can be smaller due to accuracy and size of the mouse pointer.

Refer to **Section 8.15 Input Controls** for more details about appropriate sizing.

6 Display Layout and Design Considerations

The information on HMI displays should:

- Meet the needs of the users
- Match users' understanding of operations and how the information is related
- Show only the necessary information for the task
- Allow users to quickly understand current and future status and system response to actions
- Reduce visual clutter

To meet these goals, first determine:

- Users' goals, tasks, and mental model
- Information and controls that are necessary for the task
- The most appropriate way of representing the information and controls
- Contextual information that is of value for each element of information
- How to group information
- How to highlight key information
- How to assist with situation awareness
- How to physically layout and align information

IMPORTANT: Displays should be designed for the primary (typically the operator) users while still supporting secondary users. The latter will still use the HMI, but the information they need can be provided on secondary displays or pop-up screens.

6.1 Users' Goals, Tasks, and Mental Model

Understanding the users' goals, tasks, workflow, and mental model is a critical first step in HMI design and should be completed by observing and interviewing users to document:

- Their job goals and related tasks
- Their understanding of how operations work
- How the information and controls of the operations are related
- Their understanding of their tasks

Prototyping new displays with users is another way to gather data on whether the displays meet users' goals, tasks, and mental model. The way users understand operations and their tasks does not always match the way the system is mechanically designed or how the system is laid out in a Piping and Instrument Diagram (P&ID).

A prototype does not need to be drawn with the HMI package that will be utilized, for instance a prototype can be drawn in any drawing package, although if there is already an HMI toolkit available containing the necessary objects that will be used then this would be the quickest way to generate a prototype.

6.2 Necessary Information and Controls

Determining what information and controls are necessary in an HMI requires understanding the users' goals, tasks, and perception of the tasks and operations. Information that is unnecessary clutters the screen and should be eliminated, leaving only information the user needs to successfully execute their job function. Keeping focus on the most critical information can be accomplished using methods such as sizing and placement on display.

HMI displays contain data pixels and non-data pixels. Non-data pixels are any pixels that are not used to display data, including static labels, icons, and lines. All pixels (non-data and data) consume users' attention and memory, so it's important that every pixel has a purpose. Some non-data pixels provide understanding and context and cannot be removed without causing loss of meaning and clarity to the display. These should be visually muted so they do their job without attracting attention.

6.3 Data Presentation

There are different ways to present data on an HMI display. The choice of format is dependent on the type of information, intended message, and users' needs. Only information that is meaningful for the task the user is performing should be presented. It is critical to consistently display the same type of information on the same display, across displays, and throughout the system.

6.4 Contextual Information

For each element on the display, determine what contextual information would help turn that data into useful information. For example, a bar graph can show more than just current state; it can also indicate high point, low point, alarm points, and set points. This provides context to the users of whether that element is in a good state, bad state, or moving towards a good or bad state.

Contextual information can also be provided by placing different data elements next to each other. For example, placing multiple, related bar graphs next to each other inside a tank vessel or showing a trend next to a bar graph provides additional context for that data.

6.5 Grouping

Grouping related information provides additional context, visual cues, and value. For example, when placing several bar graphs of related information next to each other, they should be scaled appropriately so users can easily determine that everything is good if they see a straight line. If that line is not straight, users should be able to identify that there is an issue without having to look for more information or wait for an alarm.

Encourage meaningful comparisons by doing the following:

- Combining items in a single table or graph
- Placing items close to one another
- Linking items in different groups using a common color
- Including comparative values for clarity (i.e. ratios, percentages, or variances)

Grouping can be done by spatially placing related items near each other, using lines, or using background shading. The goal is to reduce visual clutter using the least visible means. Not all information must be compared or linked to everything else. Discourage meaningless comparisons by separating items from one another spatially.

6.6 Highlighting Key Information

Key information and controls should have more prominence than the less important information. This includes information that is always important or information that is only important at the moment. This keeps users from being distracted by less important information.

Highlighting should be done by placement, size, enclosure, and shading/coloring.

- Placement: Top-left region of display is area of greatest emphasis, all else being equal
- Size: Larger items receive more emphasis
- Enclosure: Anything enclosed by borders or surrounded by a fill color can stand out

Keep highlighting to a minimum since displays should only show necessary information and be organized to match the users' mental model and task workflow.

6.7 Situation Awareness

Users need to be aware of the current and future status of the operations and task. While performing a task, it is easy to lose this awareness. The layout of the display should be designed to help users quickly recognize the status of operations, status of the task, and where the operations and task is heading.

Using alarm state icons is helpful for situational awareness. Placing these icons on items that are in alarm and providing a list of the highest priority alarms occurring helps users quickly assess the current situation. Using specific colors to indicate alarms (and not using those colors elsewhere) allows the alarms to stand out against the rest of the information on the screen.

Trends can also give an indication of what the process has been doing and where it may be heading. For example, a temperature may be in the normal range, but if it has been steadily rising over a period of time, it may eventually rise beyond the high limit. A trend display can quickly show this increase over time while a bar graph or numeric display cannot.

6.8 Physical Layout and Alignment

After determining what information needs to be on the display, the designer needs to determine how to physically locate each piece of information on the screen. As previously mentioned, related information should be grouped and key information highlighted. Additionally, if the information is part of the users' mental model (important for accomplishing the task), it may be laid out to match operations. For example, a pump that is vertically higher than a tank can have its information appear higher than the tank on the display. This may be important so users are aware of the physical relation between the two components. Make sure this is necessary information so it does not create visual clutter and restrict the layout of the display.

Basic alignment guidelines:

- Labels for data are to be left justified, with the data to the right or underneath the label
- Numeric data that is related and meant to be compared are to be justified alike, with the decimal points aligning. Usually this means data needs to be right justified
- If the entire display contains unrelated data, then the data is to be left justified
- Engineering units appearing to the right of the numeric values shall be left justified
- Data is not to be center justified unless it appears in a button, table or diagram where center justification is needed
- Align to a grid with attention paid to margins, spacing, and padding

Items on a display should be spaced far enough apart from each other so they do not overlap and are visibly distinct objects, but not spaced so far apart that there is unnecessary unused space on the display. Objects close together are subconsciously grouped together, so place related objects close to each other and unrelated objects farther apart to ensure users perceive the correct relationship of data on a display.

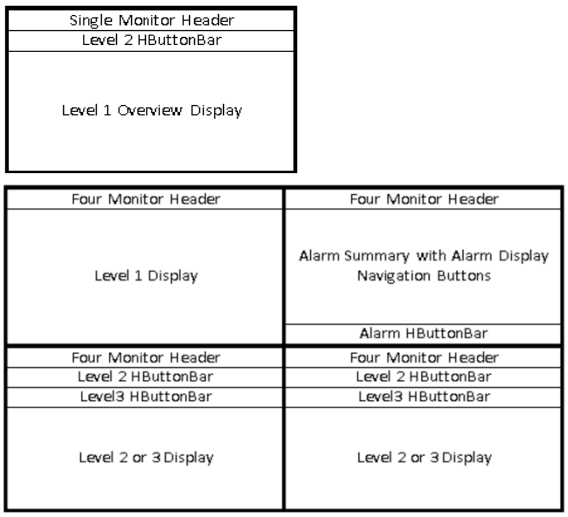
Minimum spacing is 4 pixels to ensure there is no overlapping. For touch interface, spacing is 10 pixels between command touch objects to prevent accidental activation. For navigation touch objects, there should be no additional padding between objects. Additional information on size and spacing is included in **Section 8.14 Icons** and **Section 8.15 Input Controls**.

7 Navigation Methods

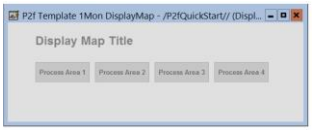
There should be multiple methods of navigation to move through the display hierarchy including main navigation on every display and additional navigation buttons to related displays, such as at the top or bottom of screen or end of a process line. Users should be able to easily navigate without being familiar with the hierarchy. This is particularly useful for maintenance and engineering.

Consideration should also be given to navigation within displays such as tabs or paging. Navigational elements should appear in consistent locations and use consistent buttons, icons, and/or text across faceplates.

The PlantPAx HMI template provides several different forms of navigation, as shown in the following template.



Navigation is available from the header, navigation bar, and display itself. In the top header, there is a button to the Display Navigation Map as shown below. This allows users to switch between process areas.



Each process area contains an overview display, Level 2 displays, and Level 3 displays.

For the single-monitor setup:

- Users navigate to the Level 2 and 3 displays from the overview display
- The top header also contains links to the Alarm Summary

For the four-monitor setup:

- Screens 3 and 4 are dedicated to Level 2 and 3 displays
- Select a process area to invoke a screen update to show information from that process area on all four monitors

On the Level 2 navigation bar, there are buttons that link to the Level 2 displays. This navigation appears on Level 2 and Level 3 displays. For the single-monitor setup, this also appears on Level 1 displays.

On the Level 3 Navigation Bar, there are buttons that link to level 3 displays. This navigation appears on Level 2 and Level 3 displays. Faceplates also have navigation as part of the faceplate object. Refer to Section 8.16 Faceplates and Popups for more details.

Refer to **Section 8.15.3 Navigation Buttons** for more details about navigation buttons.

8 Design Application Standards

This section contains application standards that are used in the Library of Process Objects and should be used to ensure consistency and effectiveness in the HMI design.

Refer to publication PROCES-RM002 (Rockwell Automation Library of Process Objects: Configuration and Usage) and publication PROCES-RM014 (Rockwell Automation Library of Process Objects: Display Elements) for additional details not included in this document.

8.1 Color Conventions

Color is an attribute that people process quickly and unconsciously, causing it to be noticed quicker and stand out more than other object attributes. Color is such a powerful characteristic that people often attempt to derive meaning from it, especially when multiple objects are shown in the same color.

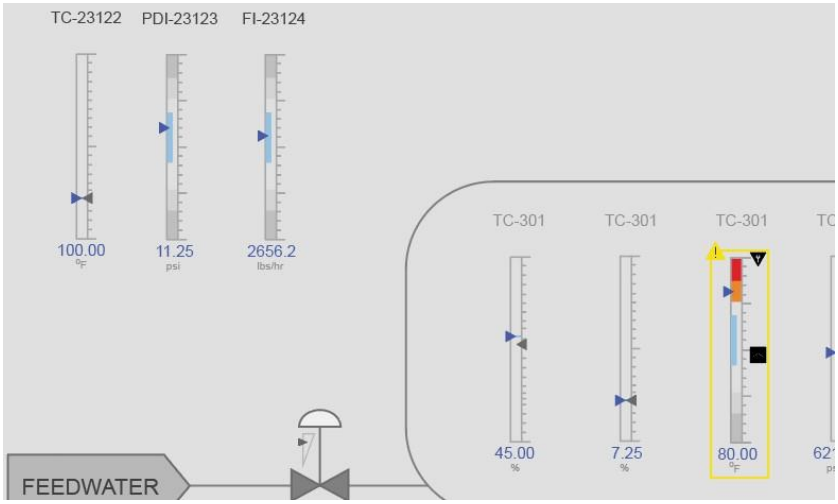
Color is also the most overused and abused attribute in display design. There are many reasons why people use color inappropriately: to make the display 'prettier,' to keep the display from being 'dull,' to make the display more 'realistic' or more like a phone 'app.' Because color subconsciously draws people's attention, be mindful when using it in your HMI designs.

When a display is designed using information that aligns with the users' mental model and task workflow (and using appropriate formats and information grouping), the users should be able to find information and controls they need quickly with minimal need for color. As always, the goal is to reduce visual clutter.

Used in a limited manner, color does have value and should be earmarked for abnormal situations, such as alarms, and to differentiate between live data, static text, and input fields. Follow these color guidelines when creating HMI displays:

- Alarms: Use bright, intense colors. Do not use these colors for anything else. For example, if using red for Priority 1 alarms, then do not use red to represent a running pump.
- Live data: Use less intense and cool colors such as dark blue or dark green. These colors are less distracting but still clearly differentiate data from static information.
- Display background: Use a non-saturated background color such as light gray as it will have minimum interference with other color choices.
- Foreground colors should be minimized. Colors used for alarms and live data should not be used for other objects. Use line thickness for emphasis rather than color.
- Gradient colors should not be used.
- Interior of static equipment: Use same color as the background display color. Process lines should be dark gray or black.
- State depiction cannot depend on color only. Additional features such as fill, shape, or simple text may be used to identify the current state.
- Use of color should be standardized, consistent, rigorously followed, and documented in the HMI Style Guide.
- Where color is used, make sure it provides enough contrast with the background, but not too much contrast as that may cause eye strain and fatigue.
- Other considerations: Some users are color impaired (red-green, green-yellow, white-cyan), lighting in the environment can affect discernibility of color, and some colors cause visual illusions and fatigue when placed near each other. As a result, color cannot be used as a sole differentiator of important factors and, therefore, it should be combined with other distinguishing characteristics such as shape.

Below is an example image of appropriate use of color. Color is reserved for live data and alarms, with the alarms having more noticeable colors. All other objects use shades of gray so as not to distract from more important information.



Example Appropriate Use of Color for Data

Below are the colors used in the PlantPAx library. Follow these colors when creating your own HMI elements to ensure consistency in the HMI.

Display Color Use	Color Name	Definition
Display Background (no tabs)	Light Gray 224	R224 G224 B224 #E0E0E0
Tab Panel Background	Light Gray 224	R224 G224 B224 #E0E0E0
Display Background behind tabs (with tabs)	Silver 192	R192 G192 B192 #C0C0C0
Static Object Color Use	**Color Name**	**Definition**
Title Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
Group Heading Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
Column Heading Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
Separator Line Color	Light Gray 216	R216 G216 B216 #D8D8D8
Process and Connector lines	Gray 160	R160 G160 B164 #A0A0A4
Equipment Border	Gray 160	R160 G160 B164 #A0A0A4
Grouping Box	Light Gray 232	R232 G232 B232 #E8E8E8

Notification Color Use	Color Name	Definition
Low Priority Alarm	Magenta	R145 G106 B173 #916AAD
Low Priority Alarm Foreground	White	R255 G255 B255 #FFFFFF
Medium Priority Alarm	Yellow	R245 G225 B027 #F5E11B
Medium Priority Alarm Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
High Priority Alarm	Orange	R236 G134 B041 #EC8629
High Priority Alarm Foreground	White	R255 G255 B255 #FFFFFF
Urgent Priority Alarm	Red	R226 G032 B040 #E22028
Urgent Priority Alarm Foreground	White	R255 G255 B255 #FFFFFF
Program Error/Bad Configuration	Black	R000 G000 B000 #000000
Program Error/Bad Configuration Foreground	White	R255 G255 B255 #FFFFFF
Fault Condition Background	Black	R000 G000 B000 #000000
Fault Condition Foreground	White	R255 G255 B255 #FFFFFF
Warning Condition Background	Dark Gray 63	R063 G063 B063 #3F3F3F
Warning Condition Foreground	White	R255 G255 B255 #FFFFFF
Prompts and Attention Background	Light Gray 224	R224 G224 B224 #E0E0E0
Prompts and Attention Foreground	Black	R000 G000 B000 #000000
Testing or Simulation Background	Light Gray 224	R224 G224 B224 #E0E0E0
Testing or Simulation Foreground	Black	R000 G000 B000 #000000
Other Abnormal State Background	Light Gray 224	R224 G224 B224 #E0E0E0
Other Abnormal State Foreground	Black	R000 G000 B000 #000000
If the normal state can appear in a control showing a notification, then the colors used for the normal state shall follow either State or Dynamic Data color uses standards		
Element State Color	**Color Name**	**Definition**
Off/De-energized/Idle/Stopped/Closed		
Off/De-energized/Idle/Stopped/Closed	Gray	R128 G128 B128#808080
On/Energized/Running/Closed	Off White	R240 G240 B240 #F0F0F0
Disabled/Out of Service	Gray	R128 G128 B128 #808080
Manual Operations (Jogging)	Light Blue	R147 G194 B228 #93C2E4
Transition (Starting, Stopping, Accelerating, Decelerating, Opening, Closing)	Light Blue	R147 G194 B228 #93C2E4

Data Entry Color	Color Name	Definition
Label Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
Engineering Unit Foreground	Light Gray 91	R145 G145 B145 #919191
Input Field Foreground (edits allowed)	Dark Gray 63	R063 G063 B063 #3F3F3F
Checkbox Foreground (edits allowed)	Dark Gray 63	R063 G063 B063 #3F3F3F
Radio Button Foreground (edits allowed)	Dark Gray 63	R063 G063 B063 #3F3F3F
Input Field Foreground (edits prohibited)	Gray	R192 G192 B192 #C0C0C0
Checkbox Foreground (edits prohibited)	Gray	R192 G192 B192 #C0C0C0
Radio Button Foreground (edits prohibited)	Gray	R192 G192 B192 #C0C0C0
Input Field, Checkbox, Radio Button Background (edits allowed)	White	R255 G255 B255 #FFFFFF
Input Field, Checkbox, Radio Button Background (edits prohibited)	Light Gray	R224 G224 B224 #E0E0E0
Dynamic Data Display Color	**Color Name**	**Definition**
Label Foreground	Dark Gray 63	R063 G063 B063 #3F3F3F
Engineering Unit Foreground	Light Gray 91	R145 G145 B145 #919191
Data Foreground	Blue	R071 G092 B167 #475CA7
Data Border (for diagrams only)	Light Gray	R192 G192 B192 # C0C0C0
Primary State Indicator Foreground	Blue	R071 G092 B167 #475CA7
Primary State Indicator Background	Light Gray	R212 G212 B212 #D4D4D4
Navigation Button	**Color Name**	**Definition**
Foreground (Fill)	Light Gray 198	R198 G198 B198 #C6C6C6
Border (Outline)	Gray 170	R170 G170 B170 #AAAAAA
Label	Dark Gray 63	R063 G063 B063 #3F3F3F

8.2 Animation

Blinking or flashing animation should only be used to annunciate a condition that requires the operator's immediate attention, and should accentuate the object or text; legibility is important in both states. Text itself should not blink in order to maintain readability; instead, using a blinking border or object next to text. When blinking animation is used, the system should provide a way for the operator to stop the blinking once he is aware of the issue.

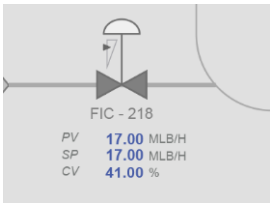
8.3 Visibility

Users should be able to view all configuration data and commands available for an object. If the users do not have access to modify data or execute the command (or if the object is in an un-editable state), the field or button should be disabled or 'greyed-out', but not made invisible as this could create confusion when users are unable to find the control.

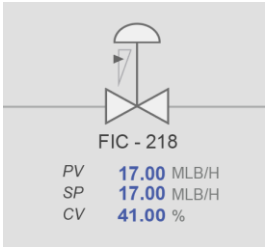
If the object's engineering configuration prevents users from entering configuration data or executing a command (or makes the configuration or command unnecessary), then it is acceptable to make these controls invisible. In this case, the object can behave in this manner because the engineering configuration usually does not change once the system is commissioned.

8.4 Depiction of Lines

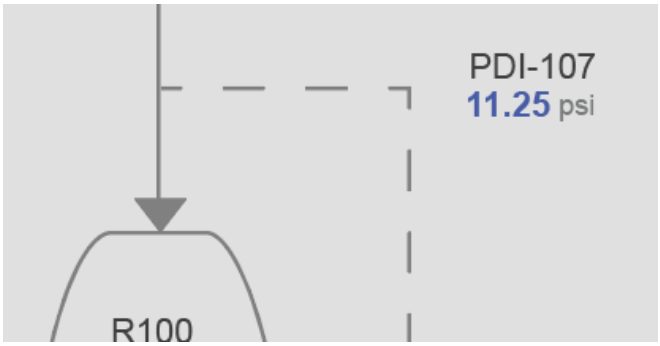
Primary process lines consist of the main flow that is occurring on the display. Height of 3 pixels and color using Gray 160 (#A0A0A4).



Secondary process lines consist of secondary flows that are occurring on the display. Height of 1 pixel and color using Gray 160 (#A0A0A4).



Connector lines are used to connect data points to objects on a display. They do not represent flow. Height of 1 pixels, dashed line, and color using Gray 160 (#A0A0A4).



8.5 Grouping Box

A grouping box should be used to group related items together and reduce visual clutter on displays. If necessary, a header can be added to the box to clarify what the group is representing.

Color for grouping box is Light Gray 232 (#e8e8e8)

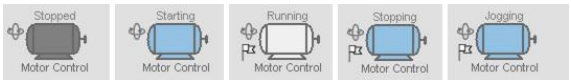
8.6 Depiction of Process Equipment

Process equipment can appear dynamically, displaying status information such as Running, Stopped, Open, or Closed. It can also be static to provide context for the display. Process equipment is based on the ANSI/ISA-5.1 and ISA-5.5 standard for process equipment presentation.

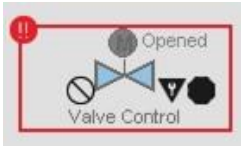
8.7 Dynamic Process Objects

Dynamic process objects display the state of the object. State is dependent on object capabilities. For example:

- the valve below is showing the following states: Opened, Closing, Closed, Opening, Stopped
- the motor below is showing the states: Stopped, Starting, Running, Stopping, and Jogging



Dynamic objects also display notifications meant to inform users of abnormal situations such as an object in maintenance mode or in an alarm condition. These notifications appear around the object, typically to the left and right to allow room for process flow lines. The following example is for a valve.

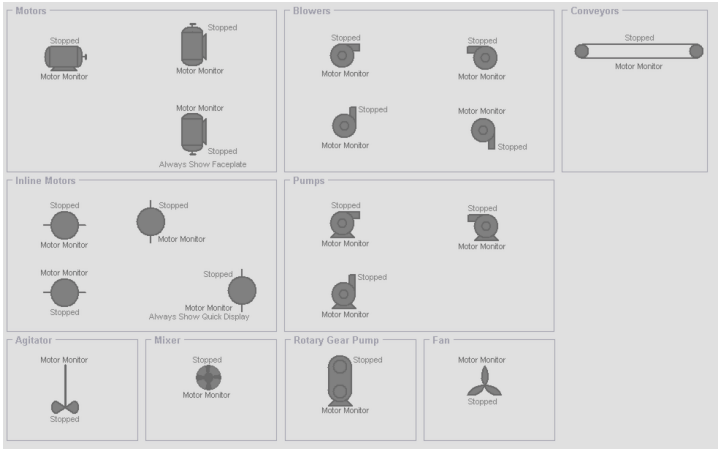


Refer to **Section 8.14 Icons** for a complete list of notification icons.

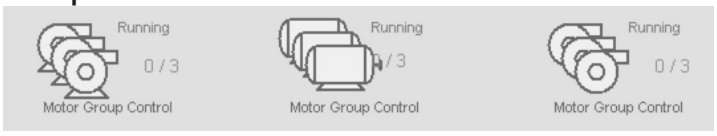
Users can click on the dynamic process object to open the faceplate for that object. Note that this capability can be turned on or off.

The Library of Process Objects contains dynamic process equipment objects based on ISA standards such as the following:

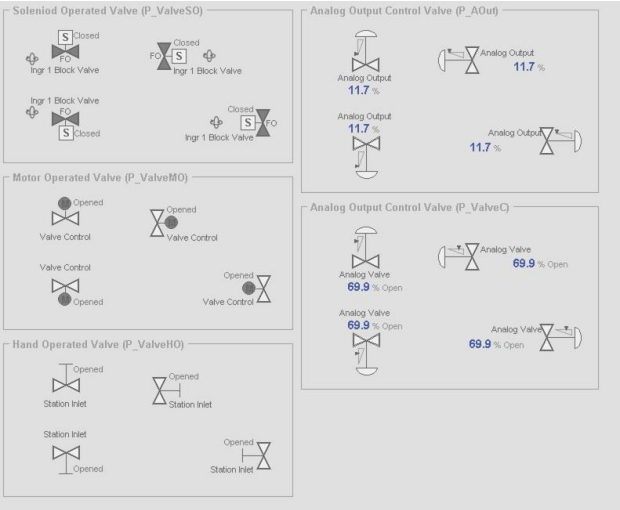
Motors



Group Control Motors



Valves



Dynamic Motor objects use the following color scheme:

Element State Color	Color Name	Definition
Off/De-energized/Idle/Stopped	Gray	R128 G128 B128 #808080
On/Energized/Running	Off White	R240 G240 B240 #F0F0F0
Disabled/Out of Service	Gray	R128 G128 B128 #808080
Manual Operations (Jogging)	Light Blue	R147 G194 B228 #93C2E4
Transition: Starting, Stopping, Accelerating, Decelerating	Light Blue	R147 G194 B228 #93C2E4

Dynamic Valve objects use the following color scheme:

Element State Color	Color Name	Definition
Closed	Gray	R128 G128 B128 #808080
Open	Off White	R240 G240 B240 #F0F0F0
Stopped	Light Blue	R147 G194 B228 #93C2E4
Closing/Opening	Part Off White Part Gray	R240 G240 B240 #F0F0F0 R128 G128 B128 #808080

8.8 Static Process Objects

Static objects are objects on a display that do not contain data themselves, but are used to enhance the users' understanding of the data presented. Examples include tanks, process lines, or equipment. For these objects, use dark gray (Gray 160 #A0A0A4) for outlining the object and the same color as the background for the interior. Some other recommendations:

- Avoid using color, gradients, three dimensional images, cutaways, and animations.
- Avoid using detailed drawings of static devices as they result in visual clutter.
- Develop standard shapes and sizes for vessels, instrumentation, pumps, etc.
- Use object size as a way to display relative importance.
- Depict flow from left to right; vapors go up and liquid down.
- Ensure lines enter and leave the screen in consistent manner; entry and exit points (that are also navigation targets) should be consistently presented and differentiated from non-navigation link labels.

Refer to ISA-5.5 for standard graphic symbols to use.

8.9 Font Types and Sizes

Text on HMI displays should be clear and easy to read as it is important to the users' understanding of navigation. Key recommendations for font types and sizes:

- Use Sans Serif font to improve readability.
- Use a consistent font size for data, captions, title, and headers (typically 10pt font). Labels and annotation text can be smaller.
- Primary data should be larger and use a bold typeface.
- Use a larger font size than their corresponding labels and units to make data stand out.

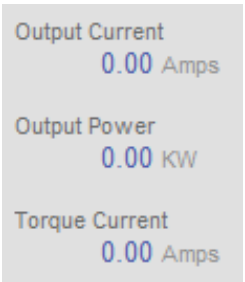
Font sizes are dependent upon screen size and how the users view the monitor. For example, a large monitor mounted high on a wall or ceiling that is meant to show data at a distance from users may need a larger font than a monitor that is mounted on a control console in front of users.

Below are recommended fonts in the Library of Process Objects for use on desktop or cabinet monitors.

Element	Font Size (pt)	Font Style
Live Data – primary (main information on display or faceplate)	11	Arial, Bold
Live Data – secondary (supporting information on display or faceplate; less common)	10	Arial, Regular
Units (engineering unit next to data)	8	Arial, Regular
Labels of data or screen elements	8	Arial, Regular
Control labels (buttons, radio, numeric input, string input, checkbox, textbox, toggle)	10	Arial, Regular
Text in button	10	Arial, Regular
On/Off label for Toggle	8	Arial, Regular
Trend label	10	Arial, Regular
Motor State (faceplate)	10	Arial, Bold
Communication Status Display (faceplate)	10	Arial
Group label 1	10	Arial, Bold
Group label 2 (used in HMI template for area labels on Overview display)	18	Arial, Regular
Display Title	24	Arial, Bold
Process Area Title in Header (single/multi-monitor display)	12/16	Arial, Bold

8.10 Data Alignment

Numeric data that is related and meant to be compared should be like-justified with the decimal points aligning. Usually this means data needs to be right justified. When engineering units are used, numeric data should be right justified and the engineering units should be to the right of the numeric values and be left justified (as shown below).



Left justify data only if it does not have engineering units and is unrelated to other data. Center justify only when the data appears in a button, table or diagram.

8.11 Static Text

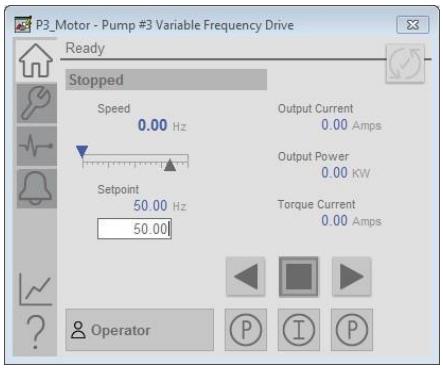
Static text refers to text that does not change during operation. This includes text that can be modified by users with appropriate privileges (for example, changing label of a control) but is not intended to be changed often.

General guidelines for static text data:

- Minimize abbreviations as they may be difficult to translate to other languages
- Size according to hierarchy and size of display
- Reserve space for translation to other languages

8.11.1 Title Bar Text

Popup displays should have a title bar containing text (sometimes referred to as a caption) that describes the content of the display. Colors and fonts for title bars are not configurable via the HMI.



Title Bar Text as it appears in the caption of a pop-up faceplate

8.11.2 Titles

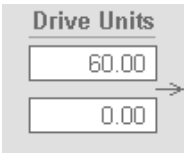
Titles appear on top of a display or page and identify or describe the content within the display or page. Follow these guidelines for titles:

- Use bold font
- Use large font size
- Place in same location on every display to help users identify the title

See **Section 8.1 Color Conventions** for title color standards and Section 8.9 Font Types and Sizes for font standards.

8.11.3 Column Headings

Column headings appear at the top of a column of data or controls and describe the content underneath.



Example of column heading

Guidelines for column headings are:

- A bold font shall be used
- A single underline is to be used as appropriate to delineate the heading from the data
- The first column is to have a left justified heading
- Wide columns are to have left justified headings
- Narrow columns are to have center justified headings
- The second and subsequent column headings are to use the same justification

See **Section 8.1 Color Conventions** for column heading color standards and **Section 8.9 Font Types and Sizes** for font standards (use group label standard).

8.11.4 Group Headings

Group headings are used to describe and delineate a set of data or controls on a display. Group headings should be left justified and be in bold font as shown below.

Example of group heading

See **Section 8.1 Color Conventions** for group heading color standards and **Section 8.9 Font Types and Sizes** for font standards.

8.11.5 Labels

Labels refer to any static, descriptive text on a display for elements such as data, groups, display title, navigation, etc.

Labels for data should be left justified, with the data to the right or underneath the label as shown below.

Example of label for numeric display

Unnecessary labels can visually clutter the screen and distract users. Minimize labels and provide only necessary information on the display. Often, identification by context is adequate. When allocating space for labels, be sure to leave enough space for the label to grow in size when translated to different languages.

To ensure labels are used appropriately, follow these guidelines:

- Use dark gray, not black text
- Use Sans Serif font
- Use legible font and font size
- Use a consistent font throughout the design
- Use uppercase for isolated words, titles, short labels, and equipment designations
- Use mixed-case for all other users; it is much more legible
- Distinguish text by varying sizes or using different colors (different shades of gray)

Ensure that labels match the users' understanding of the machine or operations (their mental model). The users may not refer to the tank as FWS-MTR-001B even though that is the technical label for the tank. Instead, they may refer to it as 'Filtering Tank B'. In that case, use 'Filtering Tank B' as the label on the display, since it matches the users' concept of the operations.

Make sure to consider the surroundings when determining the font, size, and colors used in labels as the design environment may be very different than the operational environment. Designs should be tested in the actual environment by having users go through some realistic tasks and check that the labels are legible and match the users' mental model.

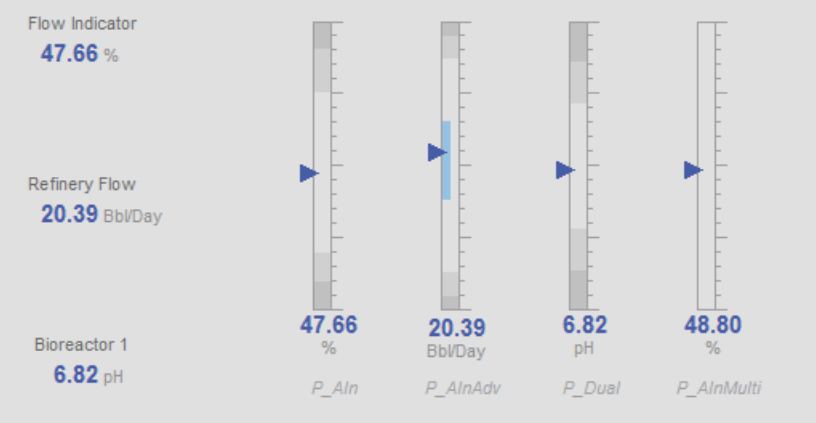
See **Section 8.9 Font Types and Sizes** for font standards.

8.12 Dynamic Data

Data by itself is not information – information is data in context that can be used to make decisions. Context is necessary for providing meaning. For example, showing limits on a bar graph allows users to know if the current value is at a good state, bad state, or approaching a dangerous situation.

It is important to present data directly, clearly, and without any form of distraction. This means providing only what is necessary and removing anything else. For example, when displaying numbers, only show the level of precision that is necessary for the operator to do their job effectively (i.e. if 2 decimal points are required for precision, don't show 3 decimal points).

Data also needs to be presented in a manner that allows for gathering information at a quick glance. Graphical presentation of data should be used when possible because it can be absorbed with a quick glance when provided with contextual information on an uncluttered display.



Display data numerically versus graphically

The information needs to be presented as rich and meaningful data along with the required level of precision that can be efficiently perceived. It is also important to be consistent when showing the same type of information on the same display, across displays, and throughout the HMI.

General Guidelines for dynamic data:

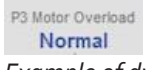
- Base the data sample rate on the rate at which the data may change and the speed at which the users must react to the change (usually 1-2 seconds). If instantaneous feedback is required, a faster rate can be used.
- Size dynamic data according to hierarchy and size of display
- Leave enough space for text translation
- Provide context for the data by placing it on top of static object or near a static object or text label
- Use a smaller font for labels and engineering units than the dynamic data
- Background color for read-only data should be the same color as the display background with no border around the data, with the following exceptions:
  - Status/State enumerations
  - Data displayed on a diagram may need a border as part of the diagram standard
  - Notifications: follow guidelines outlined in **Section 8.1 Color Conventions**

See **Section 8.1 Color Conventions** for dynamic data color standards.

8.12.1 Dynamic Text Data

Dynamic text data can be hard to ascertain from visual presentation and requires an elevated level of focus, so only use as needed.

Example of dynamic text data representing multi-state data



Example of dynamic text-label and Engineering Units can be configured but are considered more like static text

Guidelines for dynamic text data:

- Labels may appear to the left of or on top of the dynamic data
- Minimize abbreviations as they may be difficult to translate to other languages

Special cases:

- Engineering Units—Engineering units may be dynamic if they are configurable; however, when displayed with dynamic numeric data, they are to follow the same standard as static engineering units.
- Configured Names—Configured names include (but are not limited to) Object Names, Recipes Names, Step Names, Product Names. When configured names are used as a Title, Heading, or Label, then they follow the guidelines for those object types.

8.12.2 Dynamic Numeric Data

Dynamic text data can be hard to ascertain from visual presentation and requires an elevated level of focus, so only use as needed.

Example of dynamic numeric data-label on top, numeric data below

Guidelines for dynamic numeric data:

- Labels may appear to the left of or on top of the dynamic data
- Numeric data should have engineering units unless the display context eliminates the need for engineering units (or the value has no engineering units)
- Engineering units should appear to the right of (or under) the dynamic data
- Numeric data should be sized to provide accuracy of measurement. The number of significant figures and decimal places should correspond to the range, accuracy, and precision of the data.
- Integer values should not include a decimal point
- Numeric displays should be used for floating point exceptions, overflows, and underflows
- Numeric data that is related and meant to be compared should be like justified (with the decimal points aligning). Usually this means data needs to be right justified.

8.12.3 Enumerated Data (Multi-State Indicators)

Enumerated displays are used to show two or more states for an object. They may contain icons, text, or both. State and status should follow the general guidelines for Dynamic Data.

Example of multi-state indicator

Multi-state indicator with icon

Additional guidelines:

- Text displayed in enumeration tables should be left justified
- If icons and text are used together, both should be left justified with the icon appearing to the left of the text
- Leave enough space to account for language translations
- Use consistent padding around the text when a contrasting background is used

There are cases where the enumeration has only two states (binary). Information that is important, but only needs to be shown in its non-normal state, appears only when the bit is in the abnormal state.

See **Section 8.1 Color Conventions** for enumerated data color standards.

8.12.4 Binary State Indicators

A binary state indicator shows the on/off state of a Boolean value. Binary state indicators should follow the general guidelines for Dynamic Data.

Example of binary indicator. Rectangle fills when true, is empty when false. Text remains same.

Additional guidelines:

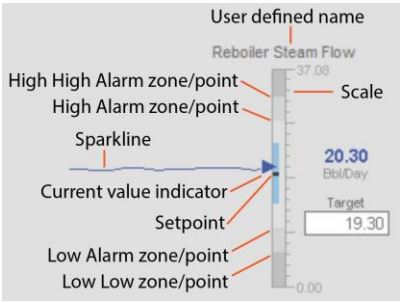
- Binary indicators should depict the On/Off state with a static label located to the right of the icon
- Text displayed with binary indicators should be left justified
- Binary indicators used to display abnormal conditions may show an icon in the abnormal state and no icon in the normal state

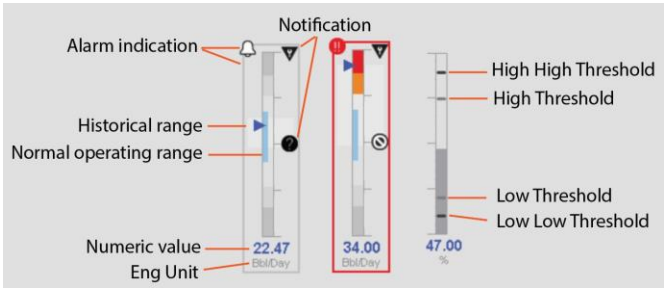
Binary information may also be displayed as enumerated data. In this case, follow the guidelines in **Section 8.12.3 Enumerated Data (multi-state indicators).**

8.12.5 Bar Graphs

Use bar graphs to show a single quantity in relation to its limit, range, or control and alarm thresholds. Multiple bar graphs can be used together to show quick relationship and comparison of data.

Bar graphs with moving indicators and limits should be used for analog data to give users an immediate picture of the data and the status.





Pros of the bar graph are that it provides contextual information in a small space allowing users to see the current data value in comparison to limits. The bar graph also allows for easy comparison with other data elements so users can quickly notice deviations. For example, multiple bar graphs can be placed next to each other and scaled appropriately such that the current value indicator arrows form a straight horizontal line when everything is good. If there are any issues, the line will no longer be straight and horizontal. Bar graphs are also useful for displaying measures that are associated with discrete items in a category.

Cons of the bar graph are that is does not provide historical data and does not have the numerical precision of a numerical display.

Required bar graph elements are as follows:

- Bar container/Background shape
- Current value indicator

Optional bar graph components:

- User defined name
- Tag Name
- High High Alarm zone/point
- High Alarm zone/point
- Low Alarm zone/point
- Low Low Alarm zone/point
- Non-alarm high high threshold
- Non-alarm high threshold
- Non-alarm low threshold

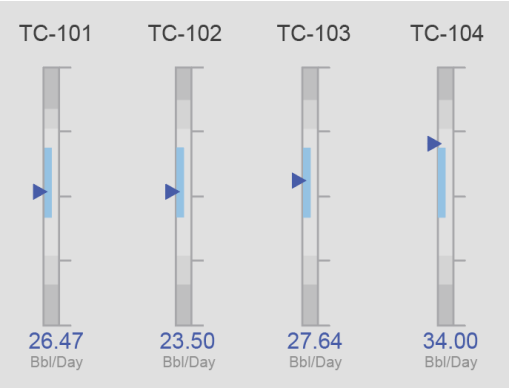
- Non-alarm low low threshold
- Scale
    - Numeric values
    - Tick Marks for scale
        - Ability to configure number of tick marks (only display as many tick marks as necessary for context, otherwise keep simple to reduce visual clutter)
- Normal Operating Range
- Set Point
- Historical range
- Numeric Value of current value
- Engineering Units
- Alarm indication - Alarm indication displays priority and alarm state. Unacknowledged alarms should blink between alarm color and gray border, while Acknowledge alarms should continually display alarm color border
- Notifications - inform users of abnormal situations that are not necessarily alarming situations
- Sparkline - historical trend of current value

Guidelines:

- Current numeric value should be displayed next to or near the bar graph (use consistent location throughout the HMI). Place the numeric value below the bar graph when possible. If there are display constraints that prevent this, numeric value should be placed to the side of the bar graph.
- Alarm thresholds should change color to indicate alarm severity. If using color, remain consistent with alarm colors throughout the HMI.
- Bar graphs may be vertical or horizontal.
- An identity label should be used either directly above or below the bar graph unless the graph is part of a group that has a label. Place the label above the bar graph when possible, but if there are display constraints that prevent this, label may be placed below bar graph

Multiple bar graphs can be used to compare related data. Follow these guidelines when using multiple bar graphs:

- Multiple bar graphs are to have a group label
- Group only related data together
- A single unit of measure shall be displayed when all bar graphs share the same units
- The scales for related bar graphs are to be normalized
- Bar graphs within the group shall have identification to differentiate one bar graph from another
- Bar graphs may show individual scales if they do not share the same range



Multiple bar graphs

8.12.6 Bar Chart

Bar charts contain two or more bars and are used for comparison of similar data. The library of process objects uses a minimized bar chart meant to give users a quick view and allow them to launch a separate dialog window with more details. This minimized bar chart contains the following components:

- Bars: Each bar chart should have a minimum of 2 bars and all bars should be the same color.
- Scale

Components can be manually added to the bar chart for additional context. These components are not attributes of the bar chart in the Library of Process Objects and would have to be added as separate components to the HMI.

- Title of the bar chart
- Subtitle with engineering unit in gray and parentheses
- Bar Labels
- Y Label
- Scale
- Total
- High High Limit
- High Limit
- Low Limit
- Low Low Limit
- Desired Range

Guidelines:

- Bars within the chart shall use the same limits
- The chart shall have an x-y Axis

8.12.7 Trends

Trends display values over time and are often used to compare similar or related values. This information allows users to predict future states to make control action decisions and review historical information to help with troubleshooting.

The pros of trends are that they provide historical information and can help users understand the current and future state of that element. The con of trends is that they can take up a lot of space.

Required trend elements are as follows:

- Title of the trend
- Current value indicator Pens/traces. Each pen/trace is differentiated by different color and icon symbol

Optional trend components:

- Subtitle with engineering unit
- Y-axis
- Y-axis min/max limit
- X-axis
- Scale for X-axis and Y-axis

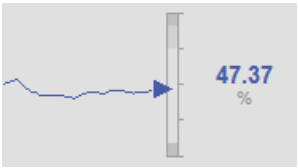
    - Tick marks
    - Numeric values
- Legend
- High High Alarm zone/point
- High Alarm zone/point
- Low Alarm zone/point
- Low Low Alarm zone/point
- Non-alarm high threshold
- Non-alarm low threshold
- Historical range indication
- Setpoint
- Numeric value
- Engineering units
- Trend navigation
- Reset historical range
- Pan/zoom functionality
- Alarm indication - Alarm indication displays priority and alarm state. Unacknowledged alarms should blink between alarm color and gray border, while Acknowledged alarms should continually display alarm color border
- Notifications - inform users of abnormal situations that are not necessarily alarming situations

Guidelines:

- Trends may have real-time and / or historical data
- Y axis is to indicate a min / max limit
- Scale range is to change based on pen selection
- Legends may be represented by text or symbol or both
- Position of legend may be based on available space

8.12.8 Sparkline

A sparkline is a very small trend typically drawn without axes or coordinates that displays a single value over time within context of limits. It is often used to allow operators to predict future states to make control action decisions.



The pros of sparklines are that they take up little space but still provide a quick view of historical information about a data point, which helps in understanding the current and future state of that data point.

The cons of sparklines are that they do not provide as much information as a typical trend and can result in visual clutter if not used appropriately.

Sparklines can be shown independently on a display or be shown as a component of a bar graph. The sparkline in the Library of Process Objects only displays the line itself.

Components can be manually added to the bar chart for additional contextual information. These components are not attributes of the bar chart in the Library of Process Objects and would have to be added as separate components to the HMI.

- Y-axis
- Y-axis min/max limit
- X-axis
- Historical range indication
- High High Alarm zone/point

- High Alarm zone/point
- Low Alarm zone/point
- Low Low Alarm zone/point
- Non-alarm high threshold
- Non-alarm low threshold
- Historical range indication
- Setpoint

Guidelines:

- Axes are not required, but should be used to clarify limits
- Use sparklines for real time trend without pan / zoom features
- Use a single pen or trace
- Provide ability to launch detailed trend if necessary

8.13 Diagrams

When used correctly, diagrams can be a very powerful tool to help users understand the context and importance of the data. Useful diagrams include (but are not limited to):

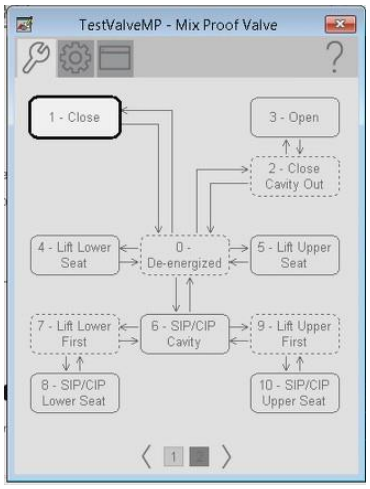
- State Diagrams
- Logical Diagrams
- Sequential Function Charts
- Limited P&ID (diagrams showing the key parts or subsets of the system)
- Equipment Layout

It is important to present the information in a manner that matches the users' mental model and does not visually clutter the display. Two common mistakes made by display developers are the use of P&ID diagrams or pictures of the actual equipment controlled by the system. P&IDs are a process design tool and should not be used as the basis for an HMI display as they can clutter the display. Furthermore, an overly detailed image of a machine with superimposed data does not help users understand the operation or system's health. Pictorials should not be used on displays as they:

- may not match the users' mental model of operations or system
- contain unnecessary information
- make it difficult to group related information
- make it difficult to highlight most important information
- take up a lot of visual space which gives more prominence to the pictorial aspect than actual data
- reduce space available for necessary data

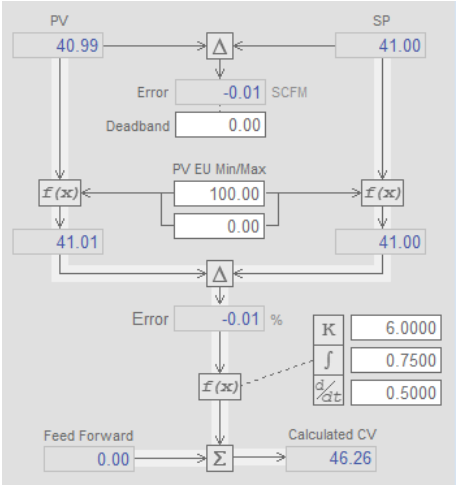
8.13.1 State Diagrams

State diagrams are a useful tool to show the behavior of a system. A state diagram includes all of the possible states a system may be in, the triggers that cause the system to change state, and the changes that occur to a system as it moves from one state to another. The state diagram clearly shows users the current state of the system and what conditions are needed to change the current state.



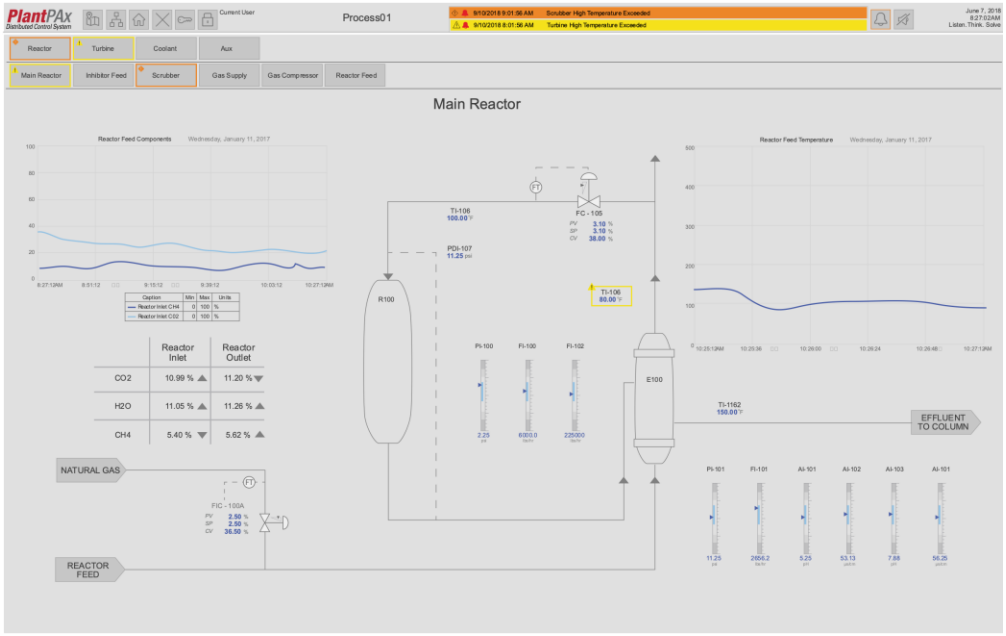
8.13.2 Logical Diagrams

Logical diagrams provide a visual flow and provide context to help users better understand a complex operation. The information includes the current state, object configuration, and logical flow.



8.13.3 Limited P&ID

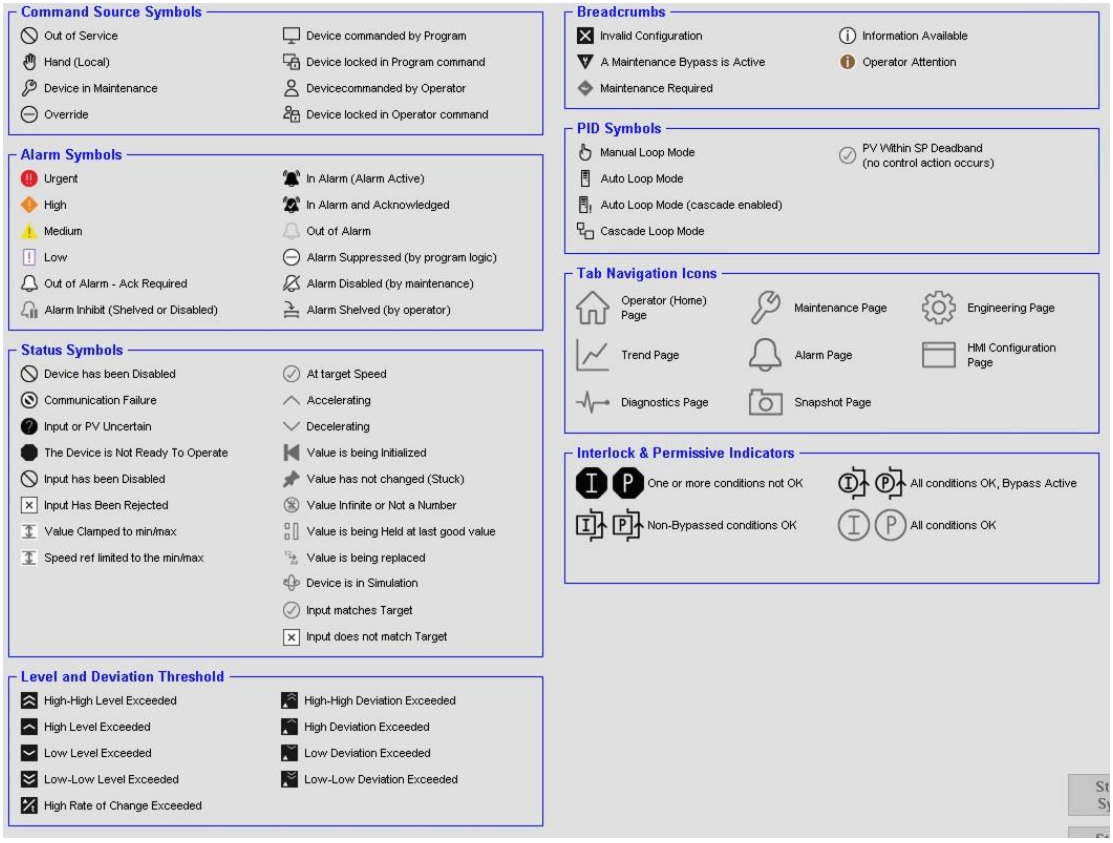
A limited P&ID can be useful when troubleshooting or manually operating equipment. They are not meant to show the entire process, but a subset of the process where users may be working. In the following image, users would have navigated to this display from a larger process display. From here, users can select the individual device faceplates.



Example of a Level 3 display

8.14 Icons

The Library of Process Objects uses a standard set of icons as shown below:



8.14.1 Icon Sizing

The recommended button icon size is 32 x 32 pixels.

The recommended information icon size is 16 x 16 pixels.

8.14.2 Icon Color

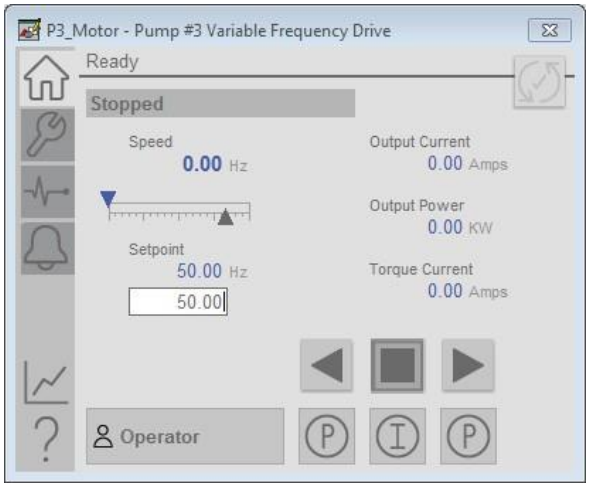
Icon color usage should be consistent with standard color definitions. For example, alarm icons should use the same colors as alarm severity levels defined in the library.

Only icons that indicate alarm conditions should use colors in the red and yellow spectrum.

Icons used for navigation should not contain colors (unless the color is indicating a warning or problem).

8.15 Input Controls

It is important to understand the users' goals and tasks to determine what controls are needed. Do not provide unnecessary capabilities—only what is necessary for achieving the tasks.



Example of controls on faceplate

Ensure consistency across controls.

- Similar controls are to use same method for entering commands or information
- Use consistent wording
- Use consistent location of entry area and information
- Use consistent location of interface management controls (e.g., scroll bar, navigation). Minimize the demands upon users.
- Do not require users to remember special codes or sequence or perform translations or conversions.
- Reduce what is required of the users and try to automate as much as possible. Beware when removing controls, as this can cause users to lose some aspects of situation awareness. This can be overcome by providing appropriate data and feedback.
- Ensure that the process of entering commands is logical.
- Support single method for input, as it is easier for users to stick with one method rather than jumping back and forth.
- Entry fields and controls need to be distinguishable from static text, so users do not have to hunt or question what is or is not a control.
- Consider providing default values where it makes sense.
- Do not require measurement of units to be entered.

Minimize potential errors.

- Allow ability to edit input before entering
- Important actions with significant consequence are to have confirmation mechanisms to avoid inadvertent activation
- Allow ability to have approval or confirmation

Feedback is an important component of controls. Users need to know that the system recognized their command and state of the system. The system response to entry should be compatible with users' expectations and understanding of the system. For lengthy processing, provide feedback so users are aware the system is still operating and won't continue to click on the command or entry thinking something is broken. Indicate to users when the process is completed.

Information update rate should be aligned with the speed at which the process is changing. Avoid excessively high update rates as they can be distracting and consume unnecessary bandwidth. The user research can help you understand what update rate is needed to accomplish tasks and provide understanding of the system state.

When a button is pressed by users, the display should provide feedback that the button was pressed. This can be done by the button itself or a control near the button indicating the state of the object.

Controls that can be disabled should have a clear enabled state where users can enter data or execute commands. They should also have a clear disabled state where users cannot enter data or execute commands due to security or device state.

Two-state buttons for equipment control should be avoided. For example, a motor should have a 'Start' button and a 'Stop' button in different positions, rather than a single button that changes meaning (or the two buttons stacked on one another). This prevents unintended operation by users that press the control more than once.

General Guidelines:

- A 'mouse over' action should indicate that the object under the mouse is an actionable control
- A control should indicate it has focus and have the ability to be disabled based on security
- If a control can be disabled, it should change appearance to reflect an enabled versus disabled state
- Controls should allow for electronic signatures as well as traceability of control usage

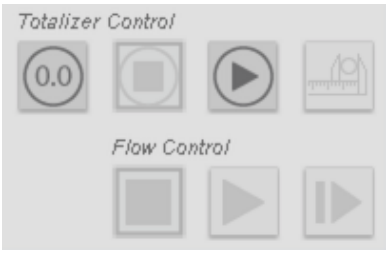
8.15.1 Input Sizing

The interaction method with the HMI (keyboard, mouse, touch) affects the size of hit zones and spacing between objects. Touch requires a larger hit zone and spacing compared to keyboard and mouse. The majority of the input controls in the Library of Process Objects have been sized to be usable in a touch environment and follow these recommended guidelines:

- Command buttons: hit zone (i.e. visible portion of button) for operator touch should be a minimum 40 height x 40 width pixels. Width may be larger if text is larger in button
- Navigation buttons: minimum 35 x 35 pixels (help and trend navigation are 32 x 32 pixels)
- Pagination buttons: 36 height x 26 width in pixels
- Input fields: 20 pixels high and width is dependent on length of input
- Checkboxes and radio buttons: hit zone minimum 20 pixels in height (for mouse use) and 30 pixels in height for touch. Minimum width is 20 pixels for mouse use with no label, 30 pixels in width for touch with no label
- Controls with labels: include control and label in the hit zone
- Toggle: height is 28 pixels and width includes the toggle label
- Minimum for touch area on screen object (for example motor on Level 2 display): 30 x 30 pixels is absolute minimum (unique case); default is 40 x 40 pixels, includes the entire object so could be larger.
- Command buttons should have 10 pixels spacing between
- Navigation buttons should have 2 pixels spacing

8.15.2 Command Buttons

Boolean operator entry that results in a change of state or operating mode should be controlled by command buttons. If an object has two or more states that can be requested (example, 'Run' and 'Stop'), then each state should have a separate command button. This prevents unintended operation by users that press the control more than once.



Example of command button in different states (enabled and not selected, disabled and selected, disabled and not selected)

Additional example of command buttons in different states (enabled and selected, enabled and not selected)

Guidelines for command buttons:

- minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- touch zone can be bigger than the button, but buffers are to be maintained.
- place a gap between command buttons to prevent unintended operation
- appearance should be different than navigation buttons to prevent users from issuing a command when they intend to navigate to a different display
- appearance should change to reflect 'Selected' or 'Pressed' state
- Text or icon on button:
    - should reflect the command to be initiated, not the current state
    - should be center justified; if the button contains enumeration, the icon and / or text should follow the enumeration guidelines for justification

Recipe commands such as Save, Restore, Download, Create, Delete or Rename are to be treated as command buttons.

8.15.3 Navigation Buttons

Navigation buttons direct users to another displays or faceplates, including pages within the faceplate.

Example of global navigation buttons

Example of navigation on faceplate in tab format

Guidelines for navigation buttons:

- Minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- Touch zone can be bigger than the button, but buffers are to be maintained.
- place a gap between buttons to prevent unintended navigation
- appearance should be different than command buttons to prevent users from issuing a command when they intended to go to a different display
- Primary navigation within a faceplate should be in tab format
- Secondary navigation should be in a page icon format
- Tab and page navigation buttons should change appearance to reflect which tab or page has focus
- Text on tab, page, or button should reflect the navigation destination
- Icons and text should be center justified; if the button contains enumeration, the icon and / or text should follow the enumeration guidelines for justification

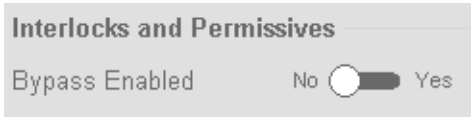
8.15.4 Security (Login / Logout)

Login / Logout buttons shall follow Navigation button guidelines.

Icons and text are to be center justified. If a button contains enumeration, the icon and / or text may follow the enumeration guidelines for justification.

8.15.5 Toggle Switches

Toggle switches should be used for On/Off or Enable/Disable functions.



Example of toggle switch for bypassing interlocks and permissives

General guidelines for toggle switches:

- minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- touch zone can be bigger than the button, but buffers are to be maintained; exception: for toggle switches that appear on advanced user displays (which would most likely be used in an office), the touch area can be smaller as users are likely to be using a mouse or similar pointing device.

- toggle switches should have a label unless the display context eliminates the need for a label
- toggle switch labels may appear to the right, left, above or below the toggle switch
- toggle switches should use position and color to indicate selection
- Toggle switches should have state labels (for example 'on/off')

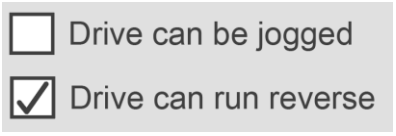
See **Section 8.1 Color Conventions** for toggle switch color standards and **Section 8.9 Font Types and Sizes** for font standards

8.15.6 Checkboxes

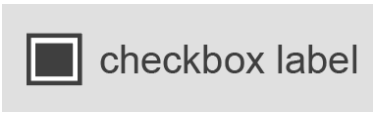
Boolean configuration data that has an 'enabled' ('Has', 'Use', 'On', etc.) and 'disabled' ('Does not Have', 'Does Not Use', 'Off', etc.) state should use checkboxes.

Checkbox labels should reflect the 'Positive' state. For example, the label for the configuration for 'Run Feedback' on a motor should be 'This Motor has Run Feedback'. In this case, checking the checkbox configures the motor to have Run Feedback.

If the configuration is TRUE in the 'Disabled' state, then the checkbox logic should be reversed, and the checkbox label revised. For example, if the configuration tag is zero when a motor has Run Feedback, then it is better to use the label 'This Motor has Run Feedback' and show the check when the tag is zero rather than using the label 'This Motor does not have Run Feedback".



Example of unchecked and checked checkbox



Example of checkbox that has children with different values, displaying indeterminate state

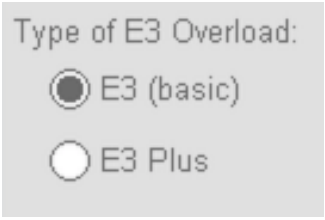
Guidelines: for using checkboxes:

- Minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- Place a gap between controls to prevent unintended changes with the following exception: for checkboxes that appear on advanced user displays (which would most likely be used in an office), the touch area can be smaller as users are likely to be using a mouse or similar pointing device.
- The checkbox should have a label unless the display context eliminates the need for a label.
- Checkbox labels should appear to the right of the checkbox and be left justified.
- If a checkbox includes a descriptive label (dedicated to a single checkbox instance), the touch area of the checkbox should include the label.
- Radio buttons and checkboxes should be the same size.
- Use a check to indicate selection; however, other indications are acceptable (for example '0' or '1')
- If the checkbox has children with different values, an indeterminate state should be displayed

See **Section 8.1 Color Conventions** for checkbox color standards and **Section 8.9 Font Types and Sizes** for font standards.

8.15.7 Radio Buttons

Configuration that has two or more selections (i.e. states or enumerations) shall use radio buttons. Radio buttons are to be used with Boolean configuration when both the 'On' and 'Off' state of the Boolean tag have meaning that are to be displayed to the users.



Example of selected and unselected radio button

Guidelines for radio buttons:

- Minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- Place a gap between controls to prevent unintended changes with the following exception: for radio buttons that appear on advanced user displays (which would most likely be used in an office), the touch area can be smaller as users are likely to be using a mouse or similar pointing device.
- Use radio buttons when there are eight or fewer options
- Radio buttons should have a label unless the display context eliminates the need for one
- Radio button labels should appear to the right of the radio button and be left justified
- If a radio button includes a descriptive label (dedicated to a single radio button instance), the touch area of the radio button should include the label
- Separate sets of radio buttons on the display label groupings accordingly
- Radio button lists should not change dynamically with the following exception: If the list is dependent on device configuration, then the list may change
- Radio buttons and checkboxes should be the same size
- Grouped radio buttons do not require a buffer between touch zones

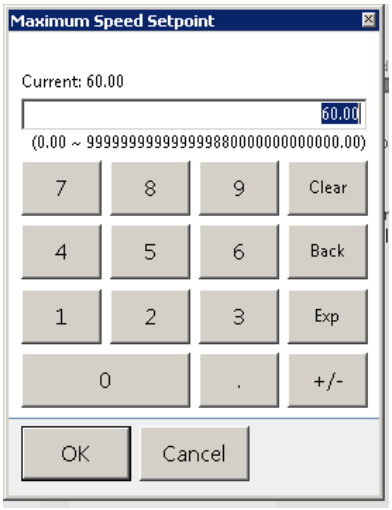
See **Section 8.1 Color Conventions** for radio button color standards and **Section 8.9 Font Types and Sizes** for font standards.

8.15.8 Analog Data Entry

There are multiple ways analog data may be changed. This section discusses direct entry, sliders, and bump buttons.

Direct Data Entry
Direct data entry uses a field on the display for users to enter the data or a popup launched by clicking on the value where the data may be entered.

Example of input field



Example of pop-up to enter value for input field

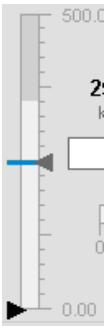
General direct data entry guidelines:

- Data entry should prevent users from entering data that is out of range.
- Entry fields should be sized according to hierarchy and size of display.
- Entry fields must have context. This can be done by placing it on top of a static object, close to, or near a static object or label.
- Analog display labels should appear to the left or above the analog input field when they are used.
- Analog data entry shall present users with engineering unit (if applicable) of the tag being modified
- Analog data entry shall present users with the minimum and maximum value of the tag being modified
- Tag description is to be provided
- If engineering units are provided, they should appear as part of the label.
- Labels and engineering units may use a smaller font than the data to make the data stand out.
- Numeric data should be sized appropriately to provide accuracy of measurement. The number of significant digits should correspond to the range, accuracy, and precision of the data.
- Integer values should not include a decimal point.
- Numeric displays should manage floating point exceptions, overflows, and underflows.

See **Section 8.1 Color Conventions** for numeric entry color standards and **Section 8.9 Font Types and Sizes** for font standards.

Slider Data Entry
Sliders are another way users can change analog data. There are two types of sliders:

- Immediate impact
- Impact after release and/or verification



Example of a PIDE control with slider (on right) on a bar graph.

General slider guidelines:

- Sliders may be used to modify analog data but should be used in addition to (rather than instead of) an analog data entry field
- An indicator should show the slider's current position as well as thresholds and/or limits.
- The control should indicate if thresholds or limits are exceeded.
- The minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- Place a gap between controls to prevent unintended changes with the following exception: for sliders that appear on advanced user displays (which would most likely be used in an office), the touch area can be smaller as users are likely to be using a mouse or similar pointing device.
- The slider control should be visually different from a bar graph.
- The slider may show tick marks if they are labeled.

Ramp Buttons Data Entry
Ramp buttons increment or decrement a current value based on a specified rate. A bump is a single increment, while a ramp button is a continuous increase / decrease while depressed.

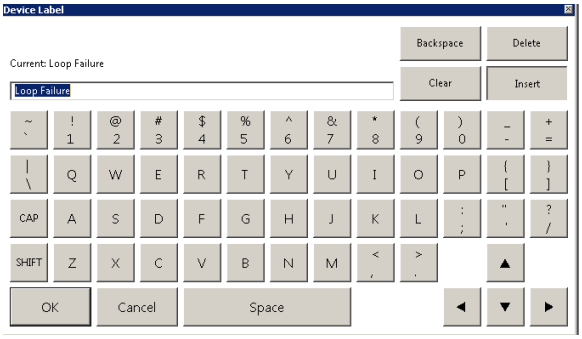
General guidelines for ramp/bump buttons:

- The minimum size of touch area should be at least as big as a fingertip to allow an operator to easily press the button without missing.
- The Touch zone can be bigger than the button, but buffers should be maintained.
- Place a gap between command buttons to prevent unintended operation.
- Text on a button is to reflect the command to be initiated
- Buttons may have an increment / decrement limit

8.15.9 Text Entry

Text entries should include appropriate descriptions, validations, and labels.

Example of text entry field



Popup to provide text input

General Analog Data Entry Guidelines:

- Entry fields should be sized according to hierarchy and size of display.
- Entry fields should have context. This can be accomplished by placing it on top of or near a static object or label.
- Text labels should appear to the left or above the text entry input field when they are used.

See **Section 8.1 Color Conventions** for color standards

8.16 Faceplates and Popups

This section contains design details for faceplates and popups to help ensure consistent and effective design.

8.16.1 Size and Orientation

Faceplate size should be an even multiple of the standard control size plus control spacing (including an extra space on the end). This allows for maximum space utilization and an easy-to-read display format. For example, if the standard control size is 40 pixels and the standard spacing is 10 pixels, then the display's width and height calculation are to be:

Size = (Maximum Number of Controls * (40 + 10)) + 10

Display orientation depends on where and how the display is used. Libraries meant for very small resolution monitors (below 800 x 600 pixels, for example) should use landscape orientation to maximize the available space.

Confirmation popups containing a few lines of text and a couple of buttons should use landscape orientation.

8.16.2 Task-Based Organization

Library faceplates should be designed with a variety of users/roles in mind. For example, a faceplate may be used by an operator running the equipment or an engineer designing the process. Users should be able to see the information they need without visual clutter. Library faceplates are divided into a series of tabs for the type of users who are using the faceplate. These tabs include (but are not limited to):

- Home: the first tab that appears when the faceplate opens. It contains data and controls that primary users would normally need to control and configure the object and should be the top or left most tab on the display.
- Maintenance: this tab contains control and configuration information that a maintenance technician would need to troubleshoot and adjust the object.
- Engineering/Configuration: this tab contains information the system developer would use when developing the control system. Most of the configuration cannot change once the object has been configured, but is available for troubleshooting and adjustments.
- Diagnostics: this tab contains data to help users determine the health of an object as well as indications of why the

object may not be operating as expected.

- Trends: the faceplate trends are not meant to replace trends on Process Graphics; instead, they are provided as an aid to troubleshoot and monitor an object's behavior.
- Alarms: this tab contains a list of all the alarms along with indications for their current status. This allows users to quickly see the alarm state of the object. Alarm acknowledgement and reset should be available on this tab. The alarm tab is to be the right-most tab on the display so it can be found quickly.
- Faults: this tab contains a list of faults to aid in troubleshooting faults or warnings from the device (Automation Device Object faceplates).

In landscape displays tabs are to be on the left side of the display. In portrait displays the tabs are to be on the top.

8.17 Security Configuration

Guidelines:

- Controls should provide a visual indication when users are restricted due to security. This visual indication may be the same as the indication used when users are restricted due to other reasons.
- Security may restrict users by role and/or qualification as well as HMI device location.

Access to controls and functions should incorporate the following security scenarios:

- Acknowledge/Reset Alarms
- Alarm Configuration
- Disable Alarms
- Shelve Alarms
- Can Bypass Feedback
- Bypass Permissives and Interlocks
- Acquire/Release Equipment Maintenance Command Source
- Acquire/Lock and Release Equipment Operator Command Source
- Can put device in/out of service
- Change Security for the device
- Change the setup of the device (Advanced)
- Configure device diagnostics
- Modify Alarm Delay Times
- Change the configuration of the device's HMI interface
- Configure device limits
- Modify Limits and Deadbands
- Configure device timers
- Change Tuning or configuration parameters
- Put Device in Simulation
- Enter Setpoints and Control Variables
- Command Equipment in Operator Command Source
- Override Inputs
- Override Outputs
- Exception Processing (Step Change; Parameter Change; Acquire; Reorder; Activate)
- Override Downloaded Phase Parameters

- Override Downloaded Setpoints
- Select; run; hold; and restart Procedures; Sequences; and Batches
- Manual Supervisory EP/EM Control
- Exception Processing (Resume; Manual; Auto; Semi-Auto; Pause; Disconnect; Release)
- Force Steps/States
- Manual Procedure; Sequence; and Batch Processing (Stop; Abort; Reset)
- Reset Run Time Accumulators
- Respond to Prompts
- Navigate to full faceplate

There are three types of security to consider: role based, user qualifications, and workstation location.

8.17.1 Role-Based Security

Role-based security gives users access to controls and data based on their job role. Examples of job roles include operator, maintenance technician, and engineer; however, each plant is different and may have different job roles.

In role-based security, a Maintenance Tech may have access to different controls and data that an Operator has.

Controls that are disabled because of security should have a consistent 'disabled' appearance that differs from the corresponding 'enabled' appearance. The disabled control should be grayscale and have a lighter foreground color than the enabled control.

8.17.2 User Qualifications

In larger plants, users may only have access to the systems in the areas where they work. For example, operators may be allowed to operate equipment in one area of the plant, but not in a different area of the plant.

8.17.3 Workstation Location

In larger systems with multiple HMI workstations, equipment operation may be restricted to certain workstations. An example of this is 'Line of Sight Control' where the operator's workstation is within site of the equipment he is controlling.

8.18 Help

Ideally, users won't need help in using the displays, but it may be necessary. It is useful to provide help documentation that is easy to find, straightforward, and focused on the user's task. For example, a Help Display may be necessary to explain the icons that are used on a faceplate for new users.

Help may be in the form of displays created using the HMI software or external documents such as PDF files.

8.19 Version Identification

Each display should have a method for storing its current version and a way to identify the version at runtime. This aids in troubleshooting and diagnosing problems. At a minimum, this information should include a version number along with the filename or storage identification so a display viewed at runtime can be found in the design environment. A brief description of display function may also be useful.

8.20 Localization

Displays may be used in other parts of the world and may require translation to different languages. Users should be able to easily change the language of all the static strings in the HMI elements of the HMI project or library. Sentences, phrases, and words in English may be longer when translated to other languages. Displays should be organized so that titles, headers, labels, and captions have room to grow when translated.

Icons may also help with localization, especially if the icons are commonly used in the industry or globally understood.

8.21 Naming Conventions

While most of the design considerations in this section have reflected on the external appearance and behavior of the HMI, the internal organization of the display elements is also important for its design and maintenance.

Just like tags and programs in the control system, design elements in the HMI should have a logical and consistent naming convention. These elements may include filenames, display elements, and collections of display elements. Each HMI development environment is different and a naming convention for one HMI platform may be different from a naming convention in another platform.

A consistent naming convention provides the developer with insight as to what the control does and how it may interact with users. For example, 'rectangle1' indicates nothing about the use of the display element, but 'NumericInputEnabledBackground' describes exactly how the display element is used.

A naming convention for filenames used in a system may aid in sorting and organization of files in a folder or dialog window.

For naming of labels that appear on a display, refer to **Section 8.11.5 Labels**.

9 Alarm Functionality

Alarms should attract attention as they are a notification of an abnormal condition that requires a response. The ANSI/ISA 18.2 standard defines an alarm as 'an audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a response.'

Present alarms in an alarm banner to see latest and highest priority alarms. Users can see, interact with, and filter all alarms from the alarm summary. Alarms should also appear next to the item causing an alarm to provide context. Do not rely on color alone to differentiate the alarm priorities; use shape as method of coding too.

Unacknowledged alarms should use blinking animation until they are acknowledged by the user.

Alarm prioritization is critical for data states that are designated as alarms, but are not in an alarm state. This is because important alarms can be hard to identify amongst the clutter of unimportant alarms. Reference the following material on how to manage alarms, determine priority, and identify what should (and should not) be classified as an alarm:

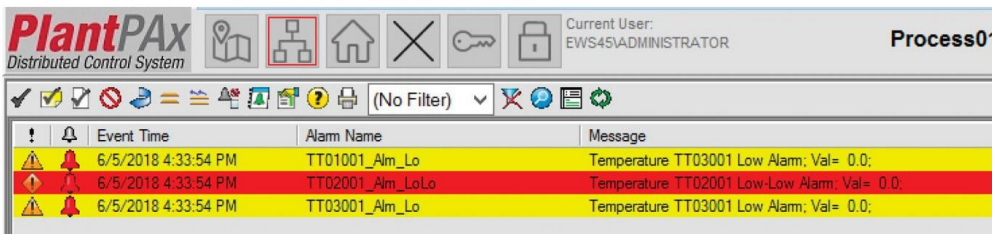
- ANSI/ISA-18.2-2016 'Management of Alarm Systems for the Process Industries' and associated Technical Reports
- PROCES-WP013B-EN-P 'Management of Alarm Systems Whitepaper— Monitoring and Assessment'
- PROCES-WP014B-EN-P 'Management of Alarm Systems Whitepaper— Performance Benchmarking and Philosophy'
- PROCES-WP015B-EN-P 'Management of Alarm Systems Whitepaper— Alarm Rationalization and Implementation'
- ASM's 'Effective Alarm Management Practices'
- The Alarm Management Handbook, by Hollifield and Habibi

9.1 Proper Configuration of the Alarm Summary

The alarm summary allows users to view and interact with a summary of all the current alarms and events in the process area. The users can acknowledge, suppress, disable, shelve, or unshelve alarms.

The alarm summary should be filtered to display only the alarms for that process area (Level 1). If the users change the process area for displays, the alarm summary should automatically update.

The standard PlantPAx HMI template includes an Alarm Summary template that can be scoped to a process area and automatically update when the process area is changed by users.

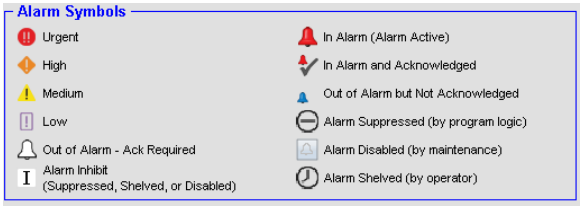


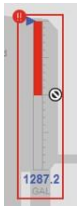
In addition to the Alarm Summary, the PlantPAx HMI template includes Alarm History, Shelved Alarms, and Alarm Explorer, accessible from navigation buttons at the bottom of the alarm screen shown in the following image.

- Alarm History displays historical data for alarms.
- Shelved Alarms display the list of currently shelved alarms and provides the capability to unshelve alarms.
- Alarm Explorer displays all existing alarms, their state and ability to enable or disable alarms, suppress or unsuppress alarms, shelve or unshelve alarms, and view operator comments.

9.2 Proper Depiction of Alarms

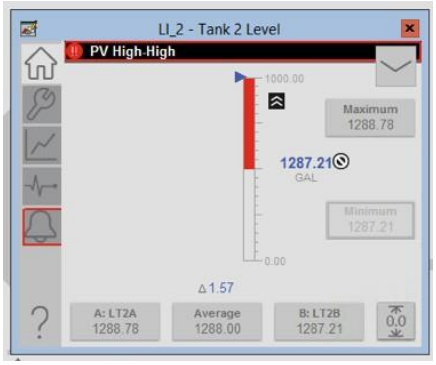
Each alarm priority should be clearly differentiated using color, shape, and alarm icon and should be consistent throughout the HMI. In addition to icons, additional visual components can be used to draw attention to the alarms. For components on an HMI display, a border is placed around component like below.





The border color should match the color of the alarm priority, with the alarm icon shown at the top left corner of border. If the alarm is unacknowledged, the border should blink between a gray border and colored border. If alarm is acknowledged, the border should remain at the alarm color until the alarm is resolved.

Alarming on a faceplate is shown below.



The top area of the faceplate displays alarms that are occurring. If there are multiple alarms, users can click on the dropdown arrow to see a full list of the alarms occurring. The highest priority alarm should be displayed on top.

The border around the top area blinks with the same color as the alarm priority. When acknowledged, the border goes away and no longer blinks. The background is black with white text. This is used to indicate an abnormal condition, whether an alarm or not. The background and text don't blink so as not to cause distraction; instead alarm color and blinking is reserved for the border to gain attention without reducing the legibility of the alarm message.

There are also breadcrumbs around the alarm tab on faceplates. When an alarm is active, the border displays the highest priority alarm color. In the screenshot above, the highest priority is Urgent, so the border is red.

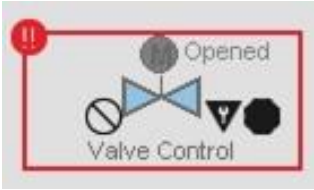
9.3 Alarm and Graphic Association

Alarms should appear where they provide useful context and attract the user's attention. This includes displaying alarms in the following areas:

- On components on display (for example, on a bar graph in alarm range)
- On components in faceplate
- Alarm tab in faceplate
- Alarm banner present on every display
- Alarm Summary
- Navigation

9.3.1 Components on Display

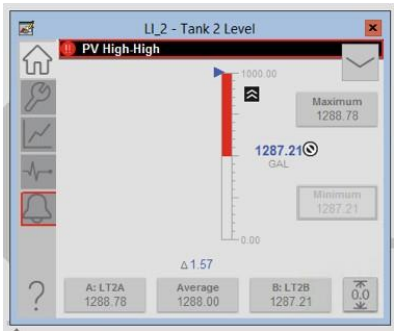
Process objects on a display have the capability to show whether they are in alarm. If in alarm, the alarm icon and border appear as shown below:

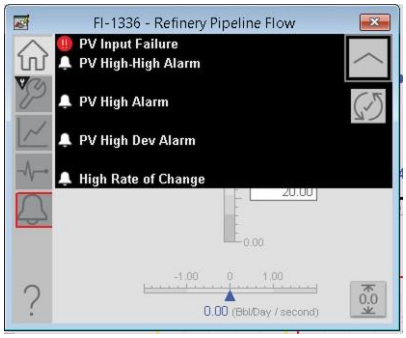


If an alarm is unacknowledged, the border blinks between gray border and alarm color. Once the alarm is acknowledged, the border remains the alarm color. This alarm capability exists for all the different process objects.

9.3.2 Components in Faceplates

Process objects on a faceplate have the capability to display whether they are in alarm. If in alarm, the alarm icon and border appear as follows:



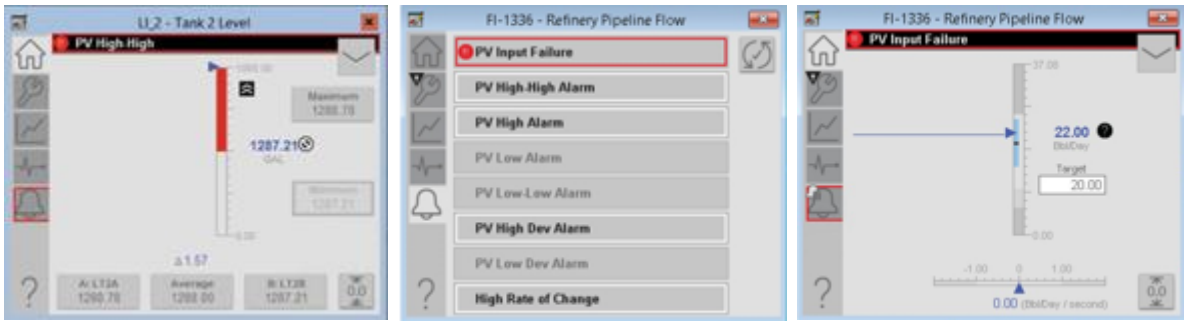


On a faceplate, the alarm icon and border does not necessarily appear around the project object. Instead, the alarm information is presented in the faceplate state area at the top of the faceplate. Users can view multiple alarms occurring on a faceplate by clicking on the dropdown. Users can also acknowledge all alarms for this object from this faceplate.

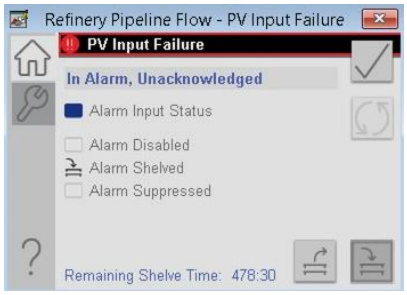
9.3.3 Alarm Tab in Faceplate

The Alarm Tab displays the highest priority alarm. An urgent alarm is shown in the example below, resulting in a red border around the alarm tab. The tab also displays alarm state; for example, if users shelve an alarm, this is indicated on the alarm tab.

The alarm page in a faceplate lists all alarms. If an alarm is occurring, the display alarm icon and border become alarm color. If the alarm goes unacknowledged, then the border blinks. If acknowledged, the border is priority color. Users can acknowledge all alarms for this object.



Users can click on alarm from here to view more details and actions as seen below.



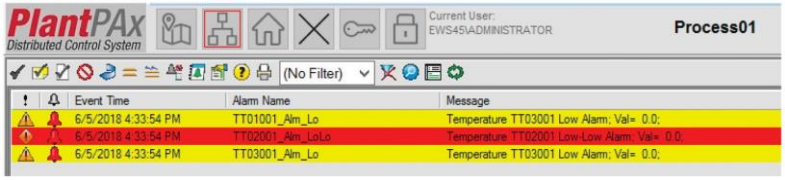
9.3.4 Alarm Banner

The alarm banner can be docked to every display so that users can be made aware if an alarm is occurring, no matter the display they are on.

The single monitor template uses an alarm banner in the header, displaying the latest alarm. The four-monitor template does not use the alarm banner since there is a monitor dedicated to alarms.

9.3.5 Alarm Summary

The alarm summary allows users to view and interact with a summary of all the current alarms and events in the process area. Users can acknowledge, suppress, disable, shelve, or unshelve alarms.



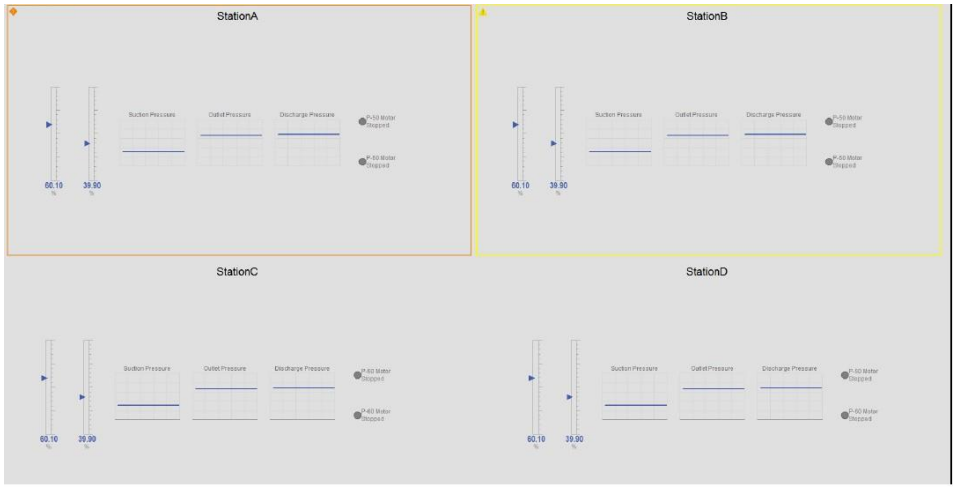
In addition to the Alarm Summary, the PlantPAx HMI template includes Alarm History, Shelved Alarms and Alarm Explorer, all accessible from navigation buttons at bottom of the alarm screen. In the single-monitor template, users can access the Alarm Summary from the alarm navigation button in the header. In the four-monitor template, the Alarm Summary is accessible from the dedicated alarm monitor.

9.3.6 Navigation

Alarm indication appears on navigation as a breadcrumb to help users find details about the alarm and to provide awareness that alarms are occurring.

Level 1 Alarm Group Annunciation
Alarm group annunciation is used on the process overview display to frame Level 1 objects associated with a unit or area and annunciate any alarm in the level 2 alarm sub-group represented by the framed area.

Alarm group annunciation includes presenting an alarm icon and border that flashes when there is an alarm in that group that is unacknowledged. When all alarms in that group are acknowledged the border remains solid.



Level 1 Alarm Group Annunciation highest priority unacknowledged alarm

Level 2 Alarm Group Annunciation
This is used with a navigation button to annunciate any alarm in a level 2 alarm sub-group that is represented by the display connected to the navigation button (for example in the header navigation).

This includes presenting alarm icon and alarm border. The border flashes when there is an alarm in that group that is unacknowledged and remains solid when all alarms are acknowledged in that group.

Level 2 Alarm Group Annunciation highest priority unacknowledged alarm

Level 3 Alarm Group Annunciation
Used with navigation button to annunciate any alarm in a level 3 alarm sub-group that is represented by the display connected to the navigation button (for example in the header navigation).

This includes presenting alarm icon and alarm border. The border flashes when there is an alarm in that group that is unacknowledged and remains solid when all alarms are acknowledged in that group. Highest priority unacknowledged alarm.

Level 3 Alarm Group Annunciation highest priority unacknowledged alarm

Alarm Navigation
For the single monitor template, the header includes a button to navigate to the alarm summary. This button shows the highest priority alarm that is in alarm, as shown in the following image.

9.4 Proper Settings for Audible Alarm Tones

Alarm tones should be clearly distinctive between each alarm priority. In a control room with multiple consoles, differentiate the alarm tones between consoles associated with each alarm condition. In scenarios where it may be difficult to differentiate by console, use distinctive tones for priority and additional tones or melodies that represents the workstations.

Alarm tone distinction can be achieved by varying the pitch (frequency) within the range of 500 to 3000 Hz, rhythm, envelope (example: rising, woop woop; constant beep beep), and timbre (example: horn versus a flute).

Guidelines for audible alarms:

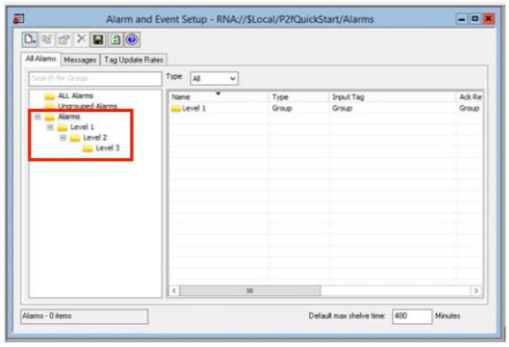
- Alarms should be audible above background ambient noise without exceeding dangerous hearing levels over 85 dB.
- Avoid startling or abrupt alarms as this can make it difficult to concentrate on addressing the issue; instead, use bursts of sounds that grab attention but are not overwhelming and tune the rise time of the alarm pulse to gradual when possible.
- Avoid disruption of other signals or background speech communication that may be important to deal with alarm.

9.5 Alarm Management Functionality

Alarm management is specified by ANSI/ISA 18.2. HMIs should follow the standards and recommended practices outlined in this document, along with PROCES-WP013B-EN-P, PROCES-WP014B-EN-P, and PROCES-WP015B-EN-P.

9.6 Alarm Organization

Alarms should be organized hierarchically to appear with the appropriate displays and coincide with the display hierarchy. This can be done by creating alarm groups that follow the same hierarchy as displays.

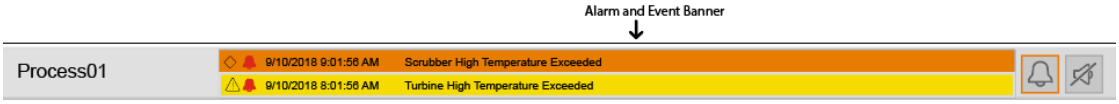


By grouping alarms in the same way as display hierarchy, easy filtration of the alarm summary and tools to display alarms related to the display of interest is possible.

In addition, this grouping is used in the four-monitor layout to filter the alarm summary to display alarms for the Level 1 display shown below. When a process area is changed from the navigation on the Level 1 display, the alarms update automatically to display alarms related to that process area.

<table>
<tr><td>Four Monitor Header</td><td>Four Monitor Header</td></tr>
<tr><td rowspan="2">Level 1 Display</td><td>Alarm Summary with Alarm Display Navigation Buttons</td></tr>
<tr><td>Alarm HButtonBar</td></tr>
<tr><td>Four Monitor Header</td><td>Four Monitor Header</td></tr>
<tr><td>Level 2 HButtonBar</td><td>Level 2 HButtonBar</td></tr>
<tr><td>Level3 HButtonBar</td><td>Level3 HButtonBar</td></tr>
<tr><td>Level 2 or 3 Display</td><td>Level 2 or 3 Display</td></tr>
</table>

For the single monitor layout, the alarm banner is filtered to display alarms for the process area. If a different process area is selected, the alarm banner updates automatically to only show alarms for the changed process area.

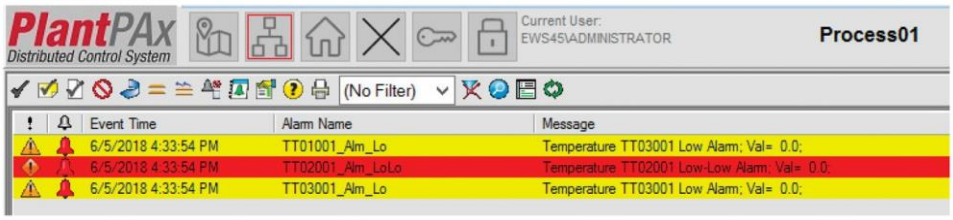


9.7 Alarm Shelving Depiction and Functionality

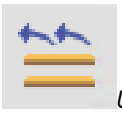
Shelving an alarm temporarily suppresses it so as to avoid clutter in the active alarm displays. It is a manual operation, where the operator specifies duration for the alarm to remain shelved. The alarm is automatically unshelved when the duration runs out.

A shelved alarm is removed from the Alarm Summary and appears in the Shelved Alarm display.

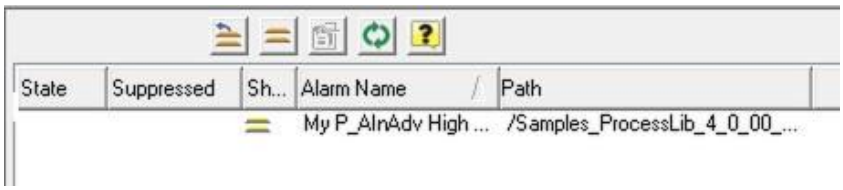
Alarm Summary
On the Alarm Summary, users can shelve an alarm using the shelve icon.



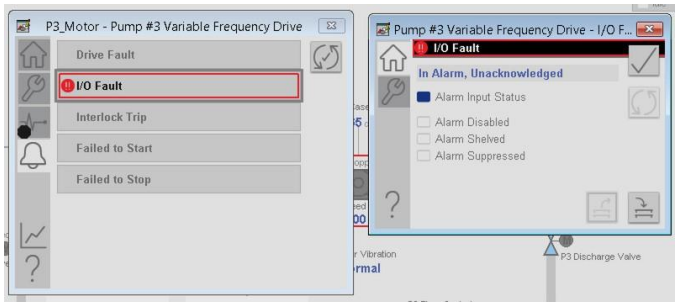

Shelve alarm icon


Unshelve alarm icon

Shelved Alarm
Users can view, shelve, and unshelve an alarm from the Shelved Alarm view.



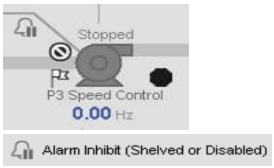
Alarm Faceplate
Users can view alarms associated with a faceplate by navigating to the settings for that alarm and shelve or unshelve the alarm.




Shelve alarm


Unshelve alarm

HMI Display
If an alarm is shelved, the shelved alarm icon appears where appropriate on HMI displays.



9.8 Non-Alarm Notifications Requiring Response

It is important to also consider notifications that do not meet the criteria for alarms. This includes alerts and events.

9.8.1 Alerts

ISA 18.2 defines an alert as "An audible and/or visible means of indicating to the operator an equipment or process condition that requires awareness, that is indicated separately from alarm indications, and which does not meet the criteria for an alarm."

Warnings and prompts are two different types of alerts. A warning is a notification of an abnormal condition that does not require a response. A prompt is a notification of a normal condition that requires a response.

Warnings and prompts should not use alarm colors. Differentiate warnings and prompts from normal conditions so it's clear to the user. This can be done in the Library of Process Objects through icons and the use of the color black. Warnings do not typically blink since they do not require a response. If a process or sequence requires the operator to respond to a prompt before continuing, blinking may be used to draw the users' attention to the prompt.

9.8.2 Events

An event is not a notification since it is logged, but not notified. Events are usually not shown on an operator display. If an event is displayed, it is for reference only and should not be highlighted.

10 Display Performance

There are three components of display performance that are important to consider: refresh rate, display call-up time, and response to users' interaction.

10.1 Display Refresh Rate

Display refresh rate should be at least twice as fast as the cycle time of the system process. For example, if the rate of change of process is four seconds, then display refreshes should happen at least every two seconds. This helps users to assess the system state and predict its next state by understanding how fast the process is going. Avoid refresh rates that are faster than twice a second as that exceeds users' abilities to process information and may cause the illusion of blinking.

The display refresh rate does not have to be the same for all parameters. Information that experiences ongoing updates (such as alarms and process values) should be refreshed frequently; but, non-changing parameters such as engineering units or alarm limits can be refreshed on a periodic basis on display call-up.

10.2 Display Call Up Time

Display call up time is the time from navigation click event to the time of display open and presentation of data. Display call-up times that are too slow can cause users to be confused or behave inappropriately (like leaving the display). To avoid this, call-up time for Level 1 and Level 4 displays should be no more than five seconds, while Level 2 and Level 3 displays should be no more than two seconds.

10.3 Response to Users' Interaction

Feedback to users' interactions on the HMI should be presented in a timely manner so users quickly get confirmation that their command was received.

Feedback is the mechanism that communicates the action taken was recognized and processed by the system. Feedback for data entry and control actions are not to be greater than two seconds or less than half a second. Feedback greater than two seconds may cause users to think that the action was not received and that they need to act again. Feedback less than a half second may be too quick to perceive the difference between enacting an action and response.

References

American Fuel & Petrochemical Manufacturers AM-15-07. Operator Situation Awareness & Delivering ASM-Compliant High Performance HMIs: Failure Modes and Success Factors.

ANSI/ISA-5.1-2009. Instrumentation Symbols and Identification. 2009. Print.

ANSI/ISA-18.2-2016. Management of Alarm Systems for the Process Industries. 2016. Print.

ANSI/ISA-101.01-2015. Human Machine Interfaces for Process Automation Systems. 2015. Print.

ASM Consortium Guidelines. Effective Console Operator HMI Design. ISBN 978-1440431647

ASM Consortium Guidelines. Effective Alarm Management Practices. ISBN 978-1442184251

EEMUA Publication 201. Process plant control desks utilizing human-computer interfaces. 2010. Print

Few, Stephen. Information Dashboard Design. Burlingame: Analytics Press, 2013. Print.

Hollifield, Bill et al. The High Performance HMI Handbook. Houston: PAS, 2008. Print.

Hollifield, Bill and Habibi, Eddie. The Alarm Management Handbook: A Comprehensive Guide. Houston: PAS, 2010. Print

ISA-5.5-1985. Graphic Symbols for Process Displays. 1985. Print.

ISO 9241-210:2010. Ergonomics of human-system interaction: human-centered design for interactive systems. 2010. Print.

PROCES-WP013B-EN-P. Management of Alarm Systems Whitepaper – Monitoring and Assessment

PROCES-WP014B-EN-P. Management of Alarm Systems Whitepaper – Performance Benchmarking and Philosophy

PROCES-WP015B-EN-P. Management of Alarm Systems Whitepaper – Alarm Rationalization and Implementation

PROCES-WP016A-EN-P. Human Machine Interfaces for Distributed Control Systems: Creating and Maintaining an Effective User Environment with Efficient Graphics.

Connect with us.

rockwellautomation.com

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication PROCES-WP023A-EN-P May 2019