SIEMENS

SIMATIC HMI

WinCC V8.1 Update 2 SIMATIC HMI WinCC V8.1 スタート ガイド

入門書

| ようこそ | 1 |
|-------------------------|----|
| アイコン | 2 |
| WinCC の操作 | 3 |
| プロジェクトの作成 | 4 |
| 通信の設定 | 5 |
| プロセス画面の設定 | 6 |
| フェイスプレートタイプとタ グ構造の設定 | 7 |
| 値のアーカイブと表示 | 8 |
| メッセージの設定 | 9 |
| 検索、発見、フィルタリング | 10 |

オンラインヘルプの印刷

法律上の注意

警告事項

本書には、ユーザーの安全性を確保し製品の損傷を防止するうえ守るべき注意事項が記載されています。ユーザーの 安全性に関する注意事項は、安全警告サインで強調表示されています。このサインは、物的損傷に関する注意事項に は表示されません。以下に表示された注意事項は、危険度によって等級分けされています。

<u> </u>危険

回避しなければ、直接的な死または重傷に至る危険状態を示します。

回避しなければ、死または重傷に至るおそれのある危険な状況を示します。

▲ 注意

回避しなければ、軽度または中度の人身傷害を引き起こすおそれのある危険な状況を示します。

通知

回避しなければ、物的損傷を引き起こすおそれのある危険な状況を示します。

複数の危険レベルに相当する場合は、通常、最も危険度の高い事項が表示されることになっています。安全警告サ イン付きの人身傷害に関する注意事項があれば、物的損傷に関する警告が付加されます。

有資格者

本書が対象とする製品1システムは必ず有資格者が取り扱うものとし、各操作内容に関連するドキュメント、特に安 全上の注意及び警告が遵守されなければなりません。有資格者とは、訓練内容及び経験に基づきながら当該製品1シ ステムの取り扱いに伴う危険性を認識し、発生し得る危害を事前に回避できる者をいいます。

シーメンス製品を正しくお使いいただくために

以下の事項に注意してください。

▲ 警告

シーメンス製品は、カタログおよび付属の技術説明書の指示に従ってお使いください。他社の製品または部品との 併用は、弊社の推奨もしくは許可がある場合に限ります。製品を正しく安全にご使用いただくには、適切な運搬、 保管、組み立て、据え付け、配線、始動、操作、保守を行ってください。ご使用になる場所は、許容された範囲を必 ず守ってください。付属の技術説明書に記述されている指示を遵守してください。

商標

®マークのついた称号はすべて Siemens Aktiengesellschaft の商標です。本書に記載するその他の称号は商標であり、 第三者が自己の目的において使用した場合、所有者の権利を侵害することになります。

免責事項

本書のハードウェアおよびソフトウェアに関する記述と、実際の製品内容との一致については検証済みです。しか しなお、本書の記述が実際の製品内容と異なる可能性もあり、完全な一致が保証されているわけではありません。記 載内容については定期的に検証し、訂正が必要な場合は次の版て更新いたします。

目次

| 1 | ようこそ | 7 |
|---|---|--|
| 2 | アイコン | |
| 3 | WinCC の操 | 单作13 |
| 4 | プロジェク | トの作成17 |
| | 4.1 | "Quick_Start"プロジェクトの作成18 |
| 5 | 通信の設定 | 21 |
| | 5.1 | WinCC のチャンネルと接続23 |
| | 5.2 | WinCC のタグ24 |
| | 5.3 | チャンネルの追加 |
| | 5.4 | 接続の作成 |
| | 5.5 | タググループのプロセスタグの作成35 |
| | 5.6 | WinCC でのプロセスタグのスケーリング41 |
| | 5.7 | 内部タグの作成43 |
| 6 | プロセス画 | 面の設定47 |
| | 6.1 | グラフィックシステム49 |
| | 6.2 | プロセス画面の作成 |
| | 6.3 6.3.1 6.3.2 | プロセス画面の編集 |
| | 6.4 6.4.1 6.4.2 | プロセス画面のダイナミック化 |
| | 6.5 6.5.1 6.5.2 6.5.3 | ユーザー定義メニューおよびツールバーの作成77 カスタマイズされたメニューおよびツールバーのプロシージャの作成 |
| | 6.6 6.6.1 6.6.2 6.6.3 6.6.4 | ランタイム中のプロジェクトの実行 |

| 7 | フェイスプ | プレートタイプとタグ構造の設定 | 103 |
|---|--------------|--|-----|
| | 7.1 | フェイスプレートタイプの設定 | 103 |
| | 7.2 | 構造タグの作成 | 106 |
| | 7.2.1 | 構造タイプの作成 | 106 |
| | 7.2.2 | 構造タグの作成 | 109 |
| | 7.3 | ポップアップ画面の設定 | 112 |
| | 7.3.1 | ポップアップ画面の作成とそのサイズの適合 | |
| | 7.3.2 733 | ホップ / ップ 画面に オブジェクトに 配直 | 113 |
| | 7.5.5 | | 120 |
| | 7.4 7.4.1 | フェイ ヘノレートタイノの設止 | 120 |
| | 7.4.2 | 構造タイプとフェイスプレートタイプのリンク | |
| | 7.4.3 | ポップアップ画面の設定 | 127 |
| | 7.5 | フェイスプレートインスタンスの作成 | 129 |
| | 7.6 | フェイスプレートタイプの変更 | |
| | 7.6.1 | オブジェクトプロパティのアニメーション化 | 133 |
| | 7.7 | プロジェクトのテスト(グラフィックシステム - フェイスプレートタイプ) | 137 |
| 8 | 値のアーカ | イブと表示 | 143 |
| | 8.1 | 値のアーカイブと表示 | 143 |
| | 8.2 | アーカイブシステム | 144 |
| | 8.3 | タグロギングの開始 | 146 |
| | 8.4 | タイマーの設定 | 148 |
| | 8.5 | プロセス値アーカイブの作成 | 150 |
| | 8.6 | プロセス値アーカイブの編集 | 153 |
| | 8.7 | プロセス画面の設定(アーカイブシステム) | 156 |
| | 8.7.1 | [トレンド]ウィンドウの設定 | 156 |
| | 8.7.2 | [テーブル]ウィンドウの設定 | 164 |
| | 8.8 | ユーザー定義メニューのカスタマイズ(アラームシステム) | 170 |
| | 8.9 | ランタイムプロパティの定義 (アラームシステム) | 173 |
| | 8.10 | プロジェクトの有効化とテスト(アーカイブシステム) | 176 |
| 9 | メッセージ | ^シ の設定 | 179 |
| | 9.1 | メッセージシステム | 180 |
| | 9.2 | アラームロギングの起動 | 182 |
| | 9.3 | メッセージブロックの設定 | 183 |

| 9.4 | ビットメッセージの設定 | |
|-------|---------------------------------|-----|
| 9.4.1 | ヒットメッセーンの作成 | |
| 9.5 | アナログメッセージの設定 | |
| 9.5.1 | 制限値の設定 | 191 |
| 9.5.2 | トレンドウィンドウでのメッセージの表示 | |
| 9.6 | メッセージステータスの色の定義 | |
| 9.7 | プロセス画面の設定(メッセージシステム) | |
| 9.7.1 | アラームメッセージウィンドウの設定 | |
| 9.7.2 | スライダオブジェクトの挿入とダイナミック化 | |
| 9.7.3 | スケールの挿入 | |
| 9.7.4 | I/O フィールド(メッセージシステム)の挿入とダイナミック化 | |
| 9.8 | ユーザー定義メニューのカスタマイズ(メッセージシステム) | 220 |
| 9.9 | ランタイムプロパティの定義(メッセージシステム) | |
| 9.10 | プロジェクトの有効化(アラームロギング) | 225 |
| 9.11 | プロジェクトのテスト(メッセージシステム) | 229 |
| 検索、発見 | 1、フィルタリング | 233 |
| 用語集 | | |
| あざ | | 247 |
| おり | | |

10

ようこそ

1

WinCC「スタートガイド」へようこそ

「スタートガイド」は、WinCC V8 に関するターゲットが絞られている概要ドキュメントです。

SIMATIC HMI WinCC

WinCC は、Microsoft Windows または Microsoft Windows Server で使用する強力な HMI シ ステムです。

HMIは、"ヒューマンマシンインターフェース"、つまり人間と機械の間のインターフェースを表します。

WinCC は、マシンやプラントで実行されているプロセスのオペレータコントロールおよび モニタリングを有効にします。WinCC とマシンの間の通信は、AS を通して行なわれます。



スタートガイド

このドキュメントの全てのテーマを学習して、WinCC 設定の基本を学ぶには約4時間で十分です。

個別の設定ステップの画像表示により、エディタにすばやく移動することができます。

サンプルプロジェクト「Quick_Start」

サンプルプロジェクトの作成は、「スタートガイド」で説明します。このプロジェクトを使用 して、水処理システムを制御できます。

そうするために、次のようなシステムを操作、監視するのに必要な様々なオブジェクトを 「設定」します。

- 制御装置でプロセスを表示、操作する画像
- 操作対象装置と設備の間でデータを転送するタグ

- プロセスデータを保存するアーカイブ
- HMI デバイスのプラントの動作ステータスを示すメッセージ

「スタートガイド」は、以下のステップを通じてガイダンスを行います。

- プロジェクトの作成
 ユーザーインターフェースを設定するためのベースとなる WinCC エクスプローラのプロジェクトを作成します。
- 通信の設定
 [タグ管理]エディタで接続とタグを設定し、ASとHMIデバイスの間の通信を確立します。
- プロセス画像の設定
 プロセスの表示、操作、モニタリングに使用する[グラフィックデザイナ]エディタでプロセス画像を設定します。
- フェイスプレートタイプと構造タイプの設定 プロセス画像と構造タググループのために、再利用可能なテンプレートを設定します。
- 値のアーカイブと表示
 [タグロギング]エディタでアーカイブを設定し、アーカイブサイクルを定義し、プロセス
 値をアーカイブできるようにします。
- メッセージの構成 [アラームロギング]エディタでディスクリートアラームとアナログアラームを設定し、 マシンやプラントのプロセスを監視できるようにします。

WinCC のインストール

WinCC セットアップには詳細なインストール説明書が含まれます。

WinCC セットアップには、設定ステップの実行に必要なすべてのプログラムが含まれています。

WinCC の技術情報およびトレーニング

WinCC の詳細については、次のインターネットを参照してください。

- オンラインサポートの SIMATIC WinCC V7 (<u>https://</u> <u>support.industry.siemens.com/cs/ww/en/view/93906404</u>):
 WinCC に関連するすべての重要情報を一覧表示します。
- トレーニング:
 - SITRAIN コース:WinCC V7 (基本コース/システムコース) (<u>https://</u>support.industry.siemens.com/cs/ww/en/view/109758633)
 - SITRAIN 上級コース:WinCC V7 および WinCC オプション (<u>https://</u> support.industry.siemens.com/cs/ww/en/view/109758660)
- オートメーションおよびドライブテクノロジー用互換性ツール: (https:// support.industry.siemens.com/kompatool/index.html?lang=en): ツールは、Siemens 製品の互換性に関する情報を提供します。例えば、オペレーティ ングシステムのバージョンなど、他の製品もカバーされています。

下記も参照

プロジェクトの作成 (ページ 17) 通信の設定 (ページ 21) プロセス画面の設定 (ページ 47) フェイスプレートタイプとタグ構造の設定 (ページ 103) 値のアーカイブと表示 (ページ 143) メッセージの設定 (ページ 179) アイコン

概要

このセクションでは、「スタートガイド」で使用するアイコンに関する情報を説明します。

使用するアイコン

以下のアイコンを使用して、画像により手順の個別ステップを示します。

| アイコン | 意味 |
|------|------------------------------------|
| | マウスの左ボタンでクリック |
| | マウスの右ボタンでクリック |
| | マウスの左ボタンでダブルクリック |
| A | キーボードでテキストを入力 |
| | 左マウスボタンを押し続けます(ドラッグアンドドロップ) |
| | 左マウスボタンを押し続けてドラッグします(ドラッグアンドドロップ) |
| | ドラッグした後、マウスの左ボタンを放します(ドラッグアンドドロップ) |
| 1 | 個々の操作ステップのナンバリング |

WinCC の操作

概要

WinCC はモジュラシステムです:

- WinCC を使用して、プロセスを視覚化し、グラフィカルユーザーインターフェースを設定します。
- プロセスのオペレータ制御および監視用ユーザーインターフェースを使用します。

WinCC の機能

- WinCCのプロセスをグラフィック表示します。
 この理由で、プロセスは画面上にグラフィック表示されます。
 プロセスの状態が変更されると、すぐに表示が更新されます。
- WinCC のプロセスを制御します。
 例えば、ユーザーインターフェースからセットポイントを指定したり、バルブを開いたりすることができます。
- WinCCのプロセスを監視します。
 アラームは、危機的なプロセス状態が発生すると、自動的にトリガされます。
 例えば、事前定義された値を超過した場合は、画面にメッセージが表示されます。
- WinCC のプロセスをアーカイブします。
 WinCC を使用すると、デジタルにメッセージをアーカイブできるだけでなく、値を処理したり、PDF として出力したりすることができます。
 それにより、プロセス履歴を文書化し、過去の生産データにアクセスできます。

WinCC のコンポーネント

WinCC の基本コンポーネントは、設定ソフトウェア(CS)およびランタイムソフトウェア(RT) です。



プロジェクト設計ソフトウェア(CS)

WinCC エクスプローラはプロジェクト設計ソフトウェアの核となっています。

WinCC エクスプローラでは、プロジェクト構造全体が表示され、プロジェクトが管理されます。

WinCC エクスプローラからさまざまなエディタを開くことができます。各エディタは、 WinCC のサブシステムに属しています。

| サブシステム | エディタ | ファンクション |
|-----------|-----------|------------------|
| グラフィックシステ | グラフィックデザイ | 設定画像 |
| 4 | ナ | |
| アラームロギング | アラームロギング | メッセージの設定 |
| アーカイブシステム | タグロギング | アーカイブデータ |
| レポートシステム | レポートデザイナ | レイアウトの作成 |
| ユーザー管理 | ユーザー管理者 | ユーザーおよびユーザー権限の管理 |
| 通信 | タグ管理 | 通信の設定 |

WinCC の最も重要なサブシステムは、次のとおりです:

WinCC Runtime ソフトウェア(RT)

WinCC Runtime は、プロセスモードでプロジェクトを実行します。すると、プロジェクト はランタイムになります。

WinCC Runtime は、プロセスのオペレータ制御および監視を有効にします。

ランタイムシステムには、特に以下のタスクが含まれます。

- CS データベースに保存されている設定データの読み取り
- 画面でのプロセス画像の表示
- AS との通信
- プロセス値やアラームイベントのような現在のランタイムデータのアーカイブ
- スルーセットポイント指定または電源オン/オフのようなプロセスの制御

プロジェクトの作成

概要

このセクションでは、WinCC エクスプローラのプロジェクトの作成方法について説明します。

プロジェクトは、WinCC のユーザーインターフェースの設定の基本です。プロジェクト内で、プロセスを操作、監視する必要がある全てのオブジェクトを作成、編集します。

原理

プロジェクトは WinCC エクスプローラで作成します。

WinCC エクスプローラは WinCC の設定システムです。WinCC エクスプローラを使用して、 プロジェクトを管理します。 4.1 "Quick_Start" プロジェクトの作成

4.1 "Quick_Start"プロジェクトの作成

概要

次のステップは、WinCCの起動方法や"Quick_Start"プロジェクトの作成方法を示しています。

"Quick Start"プロジェクトは、"シングルユーザープロジェクト"として作成します。

「シングルユーザープロジェクト」は、1 台の PC でのみ実行されます。他のコンピュータ はこのプロジェクトにアクセスできません。

プロジェクトは、データを処理するサーバー、およびオペレータステーションとして動作 する1台のコンピュータ上で実行されます。

WinCC エクスプローラの起動

初めて WinCC を起動すると、[WinCC エクスプローラ]が開きます。このダイアログで、プロジェクトのタイプを選択するか、既存のプロジェクトを開きます。

次に WinCC を起動するときは、前回作業を行なったプロジェクトが開きます。

WinCC を終了した時にプロジェクトが起動されていた場合、プロジェクトは有効な状態で 再起動されます。

必要条件

• WinCC の「標準」バリアントが最低限インストールされていること。

4.1 "Quick_Start"プロジェクトの作成

手順

- 1. WinCC を起動します。
 - 必要な場合、<Windows+R>を使用して入力画面を開きます。



2. [ようこそ]画面で、[シングルユーザープロジェクト]を選択し、プロジェクト情報を入力します。
 [新規サブフォルダー]と[プロジェクトパス]のフィールドでデフォルト設定を受け入れるか、
 PC で別のパスを選択します。

| 💰 WinCCExplorer - Senza titolo | | - 🗆 X |
|--|--|---|
| 属 WinCC | エクスフ゜ローラ | |
| <i>←</i> | 新規プロジェクトを作成します | |
| ☆ ホーム □ 新規プロジェクト (ご) ブロジェクトを聞く | | _ _ |
| ▶ 有効 化 □ 無効化 | >>>b y | לקער לאראין אין אין אין אין אין אין אין אין אין |
| × ブロジュウトを閉じる ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ | | × |
| License Analysis | Quick_Starq 7 αν' κήτη λ | × |
| | C:\Users\Public\Documents\Siemens\WinCCProjects 新しいアブフォルダー Quick_Start | |
| Settings | デブ フォルダ−の名前付けのデフォルトとしてブ ロジェクト名を設定する | 4 作成 1 +v>ts |

4.1 "Quick_Start" プロジェクトの作成

結果

"Quick_Start"プロジェクトが作成されています。

WinCC エクスプローラでプロジェクトが開きます。

必要なエディタのあるプロジェクト構造が WinCC エクスプローラの左ペインに表示され ます。

左ペインにでエディタが選択されると、右ペインにはエディタに属する要素が表示されま す。



通信の設定

概要

このセクションでは、WinCC とオートメーションシステムの間の通信の設定に関する情報 を説明します。

このセクションでは、タグを作成し、スケーリングする方法について説明します。

原理

「WinCC Configuration Studio」を使用して通信を設定します。

通信を設定するには、少なくとも以下のコンポーネントが必要です。

- チャンネルユニット付きのチャンネルが1つ
- 1つの接続
- 1つのプロセスタグ

実際には、通信を設定することで AS の現在のプロセス値をアーカイブできるようになります。

WinCC システムでの通信



「Quick_Start」プロジェクトの設定

"Quick_Start"プロジェクトには、AS は必要ありません。

このプロジェクトでは、WinCC 内の内部タグで値が転送されます。

WinCC と AS の間の通信の設定は、プロセスタグのリニアスケーリングと同様に、実践的な 経験を得るための練習です。

5.1 WinCC のチャンネルと接続

5.1 WinCC のチャンネルと接続

概要

チャンネルと接続によって、ASをどのようにWinCCに接続するかを指定します。

チャンネル

チャンネルは、特殊な通信ドライバです。

チャンネルによって、ASからプロセスタグにプロセス値を供給することも、その逆を行うこともできます。

WinCC は、異なるオートメーションシステムの接続用の多様なチャンネルを提供します。

チャンネルユニット

チャンネルには、異なる通信ネットワークのための異なるチャンネルユニットがあります。 さらに、このチャンネルユニットは、特定のタイプの AS にアクセスするために使用され ます。

チャンネルユニットは、1つの基礎ハードウェアドライバが装備されたインターフェース、 つまり PC の1つの通信プロセッサとのインターフェイスとして動作します。

接続

チャンネルユニットで、多様なオートメーションシステムへの接続を設定します。

それぞれの接続は、定義された個々のASへのインターフェースという意味になります。 データ交換は接続を介してランタイムで行なわれます。



5.2 WinCC のタグ

5.2 WinCC のタグ

概要

WinCC のタグは、外部値または内部値のいずれかを示します。内部値は、WinCC 内で計算 またはシミュレーションに使用されます。外部値は、AS で読み込みなどに使用されます。 WinCC は、通信接続およびタグをタグ管理で管理します。

[WinCC Configuration Studio]の[タグ管理]エディタを使って、設定および管理を行います。

プロセスタグ

外部タグは、WinCC と AS 間のデータ交換用リンクとして機能します。

WinCC のそれぞれの外部タグは、接続された AS のいずれかのメモリ中の特定のプロセス値 に対応します。そのため、外部タグはプロセスタグと呼ばれます。

ランタイムで、プロセスタグのプロセス値は WinCC によって決定、入力されます。

WinCC で、プロセスタグの値を決定することもできます。これらの値は、規定されたチャンネルを通じて AS へ転送されます。したがって、AS がプロセスを制御します。



内部タグ

WinCC で内部タグを使用して、プロセスリンクなしで WinCC 内で値を転送します。

タググループ

タググループは、[タグ管理]エディタのコンポーネントです。 タググループは、タグを明確な構造に整理するために使用します。

構造タグ

構造タグは、それぞれのケースで同じプロパティである複数のタググループを作成するために使用できるテンプレートと同じです。

5.2 WinCC のタグ

作成される構造は、比較的構造化されているプラントコンポーネントを繰り返しビジュアル 化するのに役立ちます。

このために、特定の要素とプロパティの構造タイプを定義します。構造タイプは好きなだけの数の構造タグとリンクされます。それぞれが同じ要素とプロパティとなります。

この方法で、同一のアプリケーションを複数回設定しなければならないことを回避します。 実際には、このような構造は PLC またはサードパーティのアプリケーションからよくイン ポートされます。そうすると、タグを別々に設定しないで、WinCC で構成を使用できます。 5.3 チャンネルの追加

5.3 チャンネルの追加

概要

次のステップは、WinCC でのチャンネルの作成方法を示しています。

WinCC は、このチャンネルによってオートメーションシステムと通信します。これは、プロセス値を WinCC のプロセスタグにも提供します。

必要条件

• "Quick_Start"プロジェクトが開いている。

5.3 チャンネルの追加

手順

1. [タグ管理]エディタを開きます。

| 🕻 WinCCExplorer - C:\Users\Public\Documents\Siemens\WinCCProjects\Quick_Start\Quick_Start.MCP - 🗆 🗙 | | | | | |
|---|---|----------|--|--|--|
| ファイル(F) 編集(E) ビュー(V) ツール(T) ヘルプ(H) | | | | | |
| | | ۶ ? | | | |
| 🛃 Quick_Start | $\leftarrow \rightarrow \uparrow$ \Variab | ~ 7 | | | |
| 1-9 IVL° 1-9 | | | | | |
| □ \$ 9 5 [*] 管理1 | | | | | |
| | | | | | |
| | | | | | |
| 1.0 7キストおよびク >フィックリスト 2.4 | | | | | |
| ⑦ アラームO キ ゙ンク゛ | | | | | |
| \$90°0‡*`>0* | Keine Objekte vorhanden | | | | |
| ■■↓ レホ°ートテ゛サ゛イナ | | | | | |
| ケ*ローパ*ルスクリフ*ト | | | | | |
| ₽, 7‡ストライブラリ | | | | | |
| ₽ 7 ⁺ 7 ⁺ 7 [×] | | | | | |
| 2-ザー管理 | | | | | |
| Quick_Start\%的 管理\ | 小部外*:0/ライセンス:102400 | うたいえされてい | | | |

[タグ管理]エディタは、WinCC Configuration Studio で開きます。

2. "SIMATIC S7-1200、S7-1500 チャンネル"を挿入します。

通信の設定

5.3 チャンネルの追加



5.3 チャンネルの追加

データ領域を拡大するには、
 ボタンを使用して、[プロパティ]ウィンドウのサイズを減らします。
 [プロパティ]ウィンドウを再度表示するには、
 (ボタンをクリックします。)

[フロハナイ]ワイントワを再度表示するには、 《ホタンをクリックします。

4. WinCC Configuration Studio のエディタで、データ領域にある列を非表示および表示することができます。



結果

"SIMATIC S7-1200、S7-1500 チャンネル"がタグ管理に表示されます。 チャンネルユニット「OMS+」がチャンネルの下に表示されています。 通信の設定

5.3 チャンネルの追加

| 🔜 Tag Management - WinCC Configuration S | tudio | | | 2 | | × |
|--|--------------|--------------------|--------|--------|----------------|---------------------------------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | |
| Tag Management « | 📦 Tags [SII | MATIC Find | | | ب م | « |
| 🖃 🖫 Tag Management | Name | Data type | Length | Linear | scalin ^ | σ |
| 🖶 🍄 Internal tags | 1 米 | | | | | 5 |
| 🖮 📙 SIMATIC S7-1200, S7-1500 Channel | 2 | T | | | | ĕ |
| MS+ | 3 | | | | | i i i i i i i i i i i i i i i i i i i |
| Structure tags | 4 | | | | | |
| , | 5 | | | | 1 | |
| | 6 | | | | | |
| | 7 | | | | | |
| | 8 | | | | ~ | |
| 🛄 🔛 🏹 🖬 🗊 🗶 🚳 🗣 🌸 🔀 🔹 | | nnections / Groups | т. < | | > .: | |
| Ready NUM | German (Ger | many) | 100 % | 9 | 0 | ÷., |

次のステップ

チャンネルユニット「OMS+」の下でASへの接続を作成します。

概要

次のステップは、チャンネルユニット"OMS+"の下でのオートメーションシステムへの接続の作成方法を示しています。

ランタイムには、この接続を通してデータ交換が行なわれます。

必要条件

• "SIMATIC S7-1200、S7-1500 チャンネル"がタグ管理に追加されていること。

手順

1. チャンネルユニット"OMS+"の下で、新しい接続を作成します。



2. 接続名として"PLC_1"と入力します。

| 💻 Tag Management - WinCC Configu | ration Studio | | | - 0 | × |
|--|---------------|------------------|-----------|-------|------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | |
| Tag Management « | 📦 Tags [PL | .C_1] | Find | Q | • « |
| 🖫 Tag Management | Name | Comment | Data type | Leng | ^ v |
| 🗄 🍄 Internal tags | 1 米 | _1 | | | 5 |
| 🗄 – 📙 SIMATIC S7-1200, S7-1500 Ch | 2 | 1 | | | ĕ |
| | 3 | 1 | | | fie |
| | 4 | | | | |
| Structure tags | 5 | | | | |
| AI | 6 | - | | | 0 |
| < > | 7 | | | | |
| | 8 | | | | v |
| 🛄 🚰 🗘 🚝 🗊 🕷 🚭 📣 🔺 | H + H Gro | ups Tags | | > | |
| Ready | Engli | sh (United State | s) 100 | % 🗩 🖓 | - + .: |

| Tag Management | « 📦 Tags [PLC_1] | Find | | ρ- |
|--|---|---------|-----------|-----|
| Tag Management Finternal tags SIMATIC S7-1200, S7-1500 OMS+ Structure Structure PLC Copy Paste Delete Rename Export | Name 1 ※ 2 3 up | Comment | Data type | |
| Computer Computer Create Eu Delete co AS Symb Alarm Logging | ion Parameters hable/Disable Tag entificate hols 19 20 14 4 + H Groups Tags | | | • • |

3. [接続パラメータ]ダイアログを開いて、接続を設定します。

4. 接続パラメータを入力します。

| | PLC_1 | x |
|--|-----------------------------------|---|
| Connection | | |
| S7Plus network IP address: <u>A</u> ccess point: | AI 141.73.184.60 AS_Network | |
| Product family: | s71500-connection | |
| Password | | |
| Change | | |
| Password: | | |
| <u>R</u> epeat password: | | |

5. [OK]を押してダイアログを閉じます。

結果

オートメーションシステムへの"PLC_1"接続が確立されました。

接続は、タグ管理で表示されました。



次のステップ

次のステップで、[PLC_1]接続の下にタググループおよびタグを作成します。

5.5 タググループのプロセスタグの作成

概要

次のステップは、タググループでのプロセスタグの作成方法と、そのプロパティの決定方法 について示しています。

タググループは、[タグ管理]エディタのコンポーネントです。これらのコンポーネントを 使用して、構造化タグをソートし、明確に表示することができます。

プロセスタグについて、次のプロパティを指定します。

- 名前
- データタイプ
 データタイプは、WinCC におけるデータフォーマットを決定します。
 WinCC のタグのデータタイプは、AS で使用されるデータタイプとは異なります。
- アドレス
 アドレス指定によって、ASの特定のデータ範囲をプロセスタグに割り当てます。
 アドレス指定のタイプは、通信パートナのタイプに依存します。

必要条件

• "PLC_1"接続がタグ管理で作成されていること。

5.5 タググループのプロセスタグの作成

手順

1. タググループを作成します。

| Tag Management - WinCC Config | uration Studio | | | | | × |
|--|--|-----------|--------|-------------------|----------|------------|
| Tag Management « | 🃦 Tags [PLC | _1] | Find | | <u>۶</u> | « |
| Tag Management Internal tags SIMATIC S7-1200, S7-1500 Ch: OMS+ New Group | Name I 1 ※ 2 3 4 | Data type | Length | Format adaptation | ^ | Properties |
| Copy Paste Delete Rename | | | | | | |
| Computer Connection Create Enable Delete certi AS Symbols | Parameters ole/Disable Tags ficate | | | | | |
| | 20 Group | s Tags | | | >i | |

- 2. タググループ名として"Tag_Group_1"と入力します。
- 3. タググループの下にプロセスタグを作成するには、プロセスタグ名として、 「Process_Tag_1」を[名前]列の一番上にあるフィールドに入力します。 「Process_Tag_1」プロセスタグが作成されました。
| | | | _ | 2 | | - | |
|------------------------------|----|-----------|-------|--------------------------|----------|---------|-----|
| lag Management « | | lags | ag_Gr | oup_1] (| | , Q | 1 |
| 🗐 🔜 Tag Management | | Name | | Data type | Length | Forma / | ^ . |
| 🗉 🍄 Internal tags | 1 | Process_7 | rag_1 | Unsigned 32-bit value | - | Dword | |
| 🖮 📙 SIMATIC S7-1200, S7-1500 | 2 | ** | Bina | T | | | |
| ⊡ !! OMS+ | 3 | | Sigi | value | (🍋) | 5 | |
| | 4 | | Unsi | gned o bit value | | | |
| Tag Group 1 | 5 | | Linsi | aned 16-bit value | 4 | | |
| | 6 | | Sian | ed 32-bit value | | | |
| | 7 | | Unsi | gned 32-bit value | | | |
| | 8 | | Float | ting-point number 32-bit | IEEE 754 | | |
| | 9 | | Float | ting-point number 64-bit | IEEE 754 | | |
| | 10 | | Text | tag 8-bit character set | | | |
| | 11 | | Raw | Data Type | | | |
| | 12 | | Date | /Time | | | |
| | 13 | | | | | | |
| | 14 | | | | | | |
| Computer | 15 | | | | | | |
| | 16 | | | | | _ | |
| Tag Management | 17 | | | | | | |
| 000 | 18 | | | | | | |
| | - | | | | | | |
| Alarm Logging | 19 | | | | | | |

4. "符号付き 16 ビット値"データタイプを選択します。

| 💂 Tag Management - WinCC Configuration Stu | dio | | | | 10000 | | | Х |
|--|-----------------------|----------------------|----------------------|---------|-------|---------|--------|------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | do. | | | | | | - 21 | |
| Tag Management « | | Tags [Tag_G | Find | | | ρ. | - | ~ |
| Arag Management Arag Management Arag Internal tags Arag SIMATIC S7-1200, S7-1500 Channel OMS+ OMS+ DUC_1 Arag Group 1 | 1 2 3 4 5 | Name Process_Tag_1 X | Group Tag_Group_1 | Address | | inear : | ^ 3 | Properties |
| Structure tags | 6 7 8 9 | | | | | | | |
| | 11 12 13 14 | | | | | | | |
| Computer | 15 16 | | | | | _ | | |
| Tag Management | 17 | | - | | | | | |
| Alarm Logging | 19 20 | | | | 0 | | ~ | |
| | М | Tags | [| < | | > | | |
| Ready | Engl | lish (United States) | Table: 1 Tag | 100 % 😑 | | 0 | (| ÷ |

5. [アドレス]欄の[...]をクリックして、タグのアドレス指定のプロパティを指定します。

[アドレスプロパティ]ダイアログボックスが開きます。

| 6. アド | レスの説明に情報を入力します。 |
|-------|-----------------|
|-------|-----------------|

| Address properties | × |
|---|---|
| Address | |
| Description CPU Data area Address Word MW Address MW Quality Code | |
| Specify the start address OK Cancel Help | |

7. [OK]を押してダイアログを閉じます。

結果

プロセスタグ"Process_Tag_1"が作成されます。

[タグ管理]エディタのテーブルエリアに、プロセスタグが表示されます。

[プロパティ-タグ]領域は、選択されたタグのすべてのプロパティを表示します。

タイプ変換によって、ASのデータフォーマットをWinCCフォーマットに変換できます。

プロセスタグは、WinCC と AS の間の通信を作成するのに必要な最後のコンポーネントです。

| 🔩 Tag Management - WinCC Configura | tion S | Studio | | | | | | - 🗆 X |
|------------------------------------|--------|---------|---------|-------------------------|--------|---|---------------------|----------------|
| <u>File Edit View Tools H</u> elp | | | | | | | | |
| Tag Management « | | Tag | Find | | Q | • | 🤪 Properties - Ta | g » |
| 🖃 🖳 Tag Management | | Name | | Data type | Length | ^ | Selection | ^ |
| 🗈 😌 Internal tags | 1 | Process | s_Tag_1 | Signed 16-bit value | 2 | | Object type | Tag |
| | 2 | * | | | | | Object name | Process_Tag_ |
| □I OMS+ | 3 | | | | | | 🗄 General | |
| | 4 | | | | | | Name | Process_Tag_ |
| | 5 | | | | | | Comment | |
| ag_Group_1 | 6 | | | | | | ID | 96 |
| | 7 | | | | | | Data type | Signed 16-bit |
| | 8 | | | | | | Length | 2 |
| | - | | | | | | Format adaptation | ShortToSigne |
| | 10 | | | | | | AS Length | 2 |
| | 10 | - | | | | | Address | |
| Computer | 11 | | | | | | AS tag name | |
| | 12 | - | | | | | Name space | |
| Tag Management | 13 | _ | | | | | Assignment | |
| (-) | 14 | _ | | | | | Communication drive | r SIMATIC S7-1 |
| Alarm Logging | 15 | | | | | | Channel unit | OMS+ |
| | 16 | | | | | ~ | Connection | PLC_1 |
| | 14 4 | (F F] | Tags | | > | : | Group | Tag_Group_1 ∨ |
| Ready | | | | English (United States) | | | Table: 1 Tag 100 % | |

次のステップ

WinCC でプロセス値をスケーリングするためのリニアスケーリングのプロパティを定義します。

5.6 WinCC でのプロセスタグのスケーリング

5.6 WinCC でのプロセスタグのスケーリング

概要

次のステップは、WinCC でのプロセスタグのスケール方法を示しています。

リニアスケーリングを使用して、プロセス値の値範囲をリニアに、WinCC のプロセスタグの定義されている値範囲にマップできます。プロセス値自体は変更されません。

リニアスケーリングについて、次のプロパティを指定します。

- [AS 値の範囲]列にプロセス値の範囲を設定します。
- [OS 値の範囲]列に WinCC のプロセスタグの値の範囲を設定します。

リニアスケーリングは、プロセスタグでのみ使用可能です。内部タグはスケーリングできません。

例:

温度センサはシステム内の抵抗を測定し、値を"オーム"単位に戻します。特定の抵抗値は 特定の温度に対応します。

抵抗値を温度スケールに自動変換するためにリニアスケーリング機能を使用することがで きます。これにより、測定した抵抗値が瞬時にプロジェクトでは温度として表示されるよ うになります。

タグ管理では、抵抗値は[AS 値の範囲]列の設定に対応しています。温度は"OS 値の範囲"列の設定に対応しています。

必要条件

• [Process_Tag_1]プロセスタグがタグ管理で作成されていること。

5.6 WinCC でのプロセスタグのスケーリング

手順

🔜 Tag Management - WinCC Configuration Studio X File Edit View Tools Help 0 -Tag Management Find 📦 Properties - Tag << >> AS Length 🖃 🔜 Tag Management Name Linear scaling A 2 ~ 1 Process_Tag_1 Address MWO 🛓 🍄 Internal tags AS tag name 2 🖮 📙 SIMATIC S7-1200, S7-15(Name space 3 . ∎.-. 🚹 OMS+ E Assignment 4 🗄 📂 PLC_1 Communication driver SIMATIC S7 5 🔤 Tag_Group_1 Channel unit OMS+ 6 - 🔙 Structure tags Connection PLC 1 7 Group Tag_Group_ 8 E Linear scaling 9 Linear scaling 10 AS value range from -20 11 AS value range to 20 12 OS value range from 0 13 OS value range to 100 < 14 Limit Values 15 A) Low limit Computer 16 High limit 딮 Tag Management 17 Start value 18 Specifies if linear scaling is to be Alarm Logging Δ 19 performed between an OS value range 20 and an AS value range. 🔁 🗊 🖓 😫 📣 🐈 👯 H + + H Tags / < > Table: 1 Tag NUM English (United States) 100 % -Ready 0 (+)

1. リニアスケーリングを有効にし、プロセスとタグの値範囲を定義します。

結果

"Process_Tag_1"プロセスタグのためのリニアスケーリングが有効になります。

プロセスとタグの値範囲が設定されます。プロセス値の値範囲[-20...20]が、プロセスタ グの値範囲[0...100]として表示されます。

次のステップ

内部タグを作成します。

5.7 内部タグの作成

概要

次のステップは、タググループでの内部タグの作成方法と、そのプロパティの決定方法に ついて示しています。

内部タグは、WinCC 内で値を転送するために使用します。

必要条件

• "Quick_Start"プロジェクトが開いている。

手順

- 1. タグ管理で、[内部タグ]エントリを選択します。
- 2. [グループ]タブで[QuickStart_Tags]タググループを作成します。

| 💻 Tag Management - WinCC Configura | tion Studio | | | | - 0 | × |
|--|-------------|----------------|----------------|-------------|-----------|-----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | |
| Tag Management « | 🏮 Groups | [Internal ta | a Find | | Q | • « |
| 🖃 🖫 Tag Management | Name | C | onnection | Last Change | 2 | ^ 7 |
| 🖶 🌍 Internal tags | 1 Performa | ince Ir | iternal tags | 4/19/2023 9 | 21:19 AN | 70 |
| - S Performa | 2 ProcessH | istorian Ir | iternal tags | 4/19/2023 9 | 21:19 AM | per |
| ProcessH(🍋) | 3 Script | Ir | iternal tags | 4/19/2023 9 | 21:19 AN | te |
| Script | 4 TaqLoqq | ingRt Ir | iternal tags | 4/19/2023 9 | 21:19 AM | S |
| Tagl oggingBt | 5 QuickSta | rt_Tags Ir | iternal tags | 4/19/2023 1 | 0:22:22 A | |
| - Auguoggingik | 6 💥 | | AT | | | |
| | 7 | Q | | | | |
| SIMATIC 37-1200, 37-1300 CF | 8 | | | | | |
| ······ | 9 | | | | | |
| | 10 | | | | | |
| | 11 | | | | | |
| | 12 | | | | | |
| | 13 | | | | | |
| < > | 14 | | | | | |
| Computer | 15 | | | | | |
| Computer | 16 | | | | | |
| Tag Management | 17 | | | | | |
| 655 | 18 | | | | | |
| Alarm Logging | 19 | | | | | |
| | 20 | | / | | | ~ |
| 1 ■ ■ ■ ■ ■ ★ ½ ■ · | HAPHG | roups Tags | | < | > | |
| Ready NUM | English (L | Inited States) | Table: 5 Group | is 100 % 🗩 | 0 | + |

5.7 内部タグの作成

内部タグの名前として"Tank_Level"と入力して、内部タグをタググループで作成します。
 データタイプ[符号なし16ビット値]を選択します。



結果

内部タグ[Tank_Level]を作成しました。またそのデータタイプを[QuickStart_Tags]タググ ループで定義しました。

内部タグ[Tank_Level]がタグ管理で表示されています。

「@」で始まる名前のあるタグはシステムタグです。システムタグは、プロジェクトで使用 できますが、変更はできません。

通信の設定

5.7 内部タグの作成

| 🔜 Tag Management - WinCC Configur | ation | Studio | | 1. | | > | < |
|--|-------|-------------------------|-------------------------------|--------|-----|------------|---|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | |
| Tag Management « | | Tags [Internal tags |] Find | | ۹. | - | < |
| 🖃 🚮 Tag Management | | Name | Data type | Length | Lir | <u>^</u> . | τ |
| 🖶 🍄 Internal tags | 85 | @RedundantServerState | Unsigned 16-bit value | 2 | | | 0 |
| | 86 | @SCRIPT_COUNT_ACT | Unsigned 32-bit value | 4 | | | Ē |
| - 🧃 ProcessHistorian | 87 | @SCRIPT_COUNT_REQ | Unsigned 32-bit value | 4 | | | |
| Script | 88 | @SCRIPT_COUNT_TAG | Unsigned 32-bit value | 4 | | | N |
| | 89 | @ServerName | Text tag 16-bit character set | 256 | | | |
| A Quick Start Tags | 90 | @ServerVersion | Text tag 16-bit character set | 256 | | | |
| | 91 | @TLGRT_AVERAGE_TA | Floating-point number 64-bit | 8 | | | |
| Charles the second seco | 92 | @TLGRT_SIZEOF_NLL_I | Floating-point number 64-bit | 8 | | | |
| Structure tags | 93 | @TLGRT_SIZEOF_NOTI | Floating-point number 64-bit | 8 | | | |
| | 94 | @TLGRT TAGS PER SI | Floating-point number 64-bit | 8 | | | |
| | 95 | Tank_Level | Unsigned 16-bit value | 2 | | | |
| | 96 | × | | | | | |
| | 97 | | | | | | |
| · · · · · · · · · · · · · · · · · · · | 98 | | | | | | |
| Computer | 99 | | | | | | |
| Computer | 100 | | | | | | |
| Tag Management | 101 | | | | | | |
| 000 | 102 | | | | | | |
| Alarm Logging | 103 | | | | | | |
| | 104 | | | | | v | |
| | H 4 | ► H Groups Tags | | | > | | |
| Ready NUM | | English (United States) | Table: 95 Tags 100 % (=)- | 0 | | (+ | 5 |

通信の設定

5.7 内部タグの作成

プロセス画面の設定

概要

このセクションでは、グラフィックシステムに関する情報やプロセス画像の設定方法を説明 します。

プロセス画像はプロジェクトの中心的要素です。プロセス画像はプロセスを表現し、この プロセスに関する操作や監視を可能にします。

原理

[グラフィックデザイナ]エディタを使用して、プロセス画像を設定できます。このエディ タは、WinCC のグラフィックシステムの設定コンポーネントです。

各プロセス画像は、1つ以上のオブジェクトで構成されます。

- スタティックオブジェクトは、ランタイム中に変化しません。
- ダイナミックオブジェクトは、個々のプロセス値に従って変更されます。
 ダイナミックオブジェクトの例としては、バーがあります。バーの長さは、現在の温度値に応じて変化します。
- コントロール可能なオブジェクトを使用することで、プロセスに対して能動的な影響を 与えることができます。
 これらのオブジェクトには、ボタン、スライダ、あるいは特定のプロセスパラメータ
 への入力に使用する I/O フィールド(入力/出力フィールド)などがあります。

プロジェクトが複数のプロセス画像で構成されることがよくあります。それぞれのプロセス 画像は、異なるプロセスステップを表示したり、特殊なプロセスデータを表示します。

「Quick_Start」プロジェクトの設定

このセクションでは、水処理システムを示すプロセス画像「START.Pdl」を作成します。2 番目のプロセス画像「SAMPLE.Pdl」の作成は練習として見なされます。

WinCC は、プロセス画像を設定するために必要なすべてのオブジェクトを提供します。

2 つ目のステップで、前もって作成してある内部タブ「Tank_Level」を使用してプロセス 画面「START.Pdl」をダイナミック化します。この結果、ウォータータンクの水位がタグ値 の機能としてグラフィック表示されます。

6.1 グラフィックシステム

6.1 グラフィックシステム

概要

グラフィックシステムは、WinCC のサブシステムです。このサブシステムは、プロセス画 像を設定するために使用します。

グラフィックシステムは以下のタスクを処理します。

- テキスト、グラフィックまたはボタンなどのスタティックでオペレータ操作可能なオ ブジェクトを表示します。
- プロセス値に応じて棒グラフの長さを変更するなど、ダイナミックオブジェクトを更新します。
- ボタンをクリックしたり、テキストボックスにテキストを入力するなどのオペレータ入力に対応します。

グラフィックシステムのコンポーネント

グラフィックシステムは、設定コンポーネントおよびランタイムコンポーネントで構成さ れています。

- [グラフィックデザイナ]エディタは、グラフィックシステムの設定コンポーネントです。 このエディタで、プロジェクトのプロセス画像を作成します。
- グラフィックランタイムは、グラフィックシステムのランタイムコンポーネントです。
 グラフィックランタイムはランタイム中に画像を表示し、すべての入力および出力を管理します。

プロセス画面の設定

6.1 グラフィックシステム

WinCC システムのグラフィックシステム



6.2 プロセス画面の作成

6.2 プロセス画面の作成

概要

次のステップでは、WinCC エクスプローラでのプロセス画像の作成方法、名前の変更の方法を示しています。

新しい、空のプロセス画像は、[グラフィックデザイナ]エディタまたは WinCC エクスプロ ーラを使用して作成することができます。

- すぐにプロセス画像を編集する場合は、[グラフィックデザイナ]エディタを使用して作 成します。
- 処理する前に必要なプロセス画像をすべて作成しておく場合は、WinCC エクスプロー ラを使用することを推奨します。

必要条件

• "Quick_Start"プロジェクトが開いている。

6.2 プロセス画面の作成

手順

1. 新しいプロセス画像を作成します。

| WinCCExplorer - C:\Users\Public\Docur | ments\Siemens\Win | CCProjects\Quick | _Start\Quicl | k_Start.MCP | - | - | | × |
|--|-------------------|-----------------------------------|--------------|-------------|---|---|------|-----|
| ファイル(F) 編集(E) ビュー(V) ツール(T) | ∿⊮7° (H) | | | - | | | | |
| ☆ 🕞 🗁 🔲 🖬 | ▶ % | | | | | 2 | ? | • |
| Quick_Start | 1 | $\leftarrow \rightarrow \uparrow$ | Å | \Gr | | | ~ 7 | 7 |
|])/L°1-9 | | | | | | | | |
| 三 鼎 90 [°] 管理 | | | | | | | | |
| \$\$ 0 574995 \$ | 開く | I | | | | | | |
| E Satan - MA | 画像の新規作成 新規フォルダ | 2 | | | | | | |
| 1.0 テキストおよびケッラフィ 2.4 | 画像名やフォルダ4 | | | | | | | |
| Ø 75-40‡~>7 | ActiveXコントロールを追 | 崔択 | | | | | | |
| <u>ም</u> ያሳ [*] ዐ‡* ንሳ | [表示名]列を表示 | | | | | | | |
| 正 レホ [。] -トテ [、] サ [、] イナ | お気に入りの順序の | D編集 | | | | | | |
| </td <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> | | | | | | | | |
| Quick_Start\グラフィックデザイナ\ | | | 0オブション | 71 | | Þ | 化ンスさ | れてい |

プロセス画像は、WinCC エクスプローラの右ペインに表示されます。

2. 作成したプロセス画像の名前を「START.Pdl」に名前変更します。

プロセス画面の設定

6.2 プロセス画面の作成



「START.Pdl」プロセス画像は、WinCC エクスプローラの右ペインに表示されます。

3. ステップ1およびステップ2と同様にして2番目のプロセス画像を作成し、それに「SAMPLE.Pdl」という名前を付けます。

結果

プロセス画像「START.Pdl」および「SAMPLE.Pdl」が作成されます。

これらは、WinCC エクスプローラの右ペインに表示されます。

6.2 プロセス画面の作成



次のステップ

「START.Pdl」プロセス画像に複数のオブジェクトを挿入することにより、水処理システムを 画像で示します。

概要

このセクションでは、[グラフィックデザイナ]エディタを使用したプロセス画像の編集方法 を説明します。

このセクションの最後には、自分の「SAMPLE.Pdl」画像を編集して、さらに深く学べる機 会があります。

原理

[グラフィックデザイナ]エディタを使用して、プロセス画像を設定できます。このエディ タの構造は描画プログラムと似ており、同じような方法で操作できます。

[グラフィックデザイナ]エディタは、プロセス画面を設定するためのオブジェクトとツー ルを提供します。「Quick_Start」プロジェクトに対しては、主としてオブジェクトパレッ トや[グラフィックデザイナ]エディタのライブラリを使用します。

オブジェクトパレットには、プロセス画像の処理で頻繁に必要となる様々なタイプのオブ ジェクトがあります。

オブジェクトパレットのオブジェクトを、ドラッグおよびドロップでプロセス画像に挿入 します。または、オブジェクトを選択してから、プロセス画像の作業エリアでマウスの左 ボタンを一度クリックします。

オブジェクトパレットには、画像を設定する以下のタイプのオブジェクトがあります。

- 標準オブジェクト:線、多角形、楕円、円、長方形、スタティックテキストなど
- スマートオブジェクト:アプリケーションウィンドウ、画像ウィンドウ、OLE オブジェ クト、I/O フィールド、バー、ステータス表示など
- Windows オブジェクト:ボタン、チェックボックス、オプショングループ、スライダオ ブジェクトなど
- チューブオブジェクト:例、多角形チューブ、Tピース、ドブルTピース、チューブ曲がり
- コントロール:[コントロール]タブには、最も重要な ActiveX コントロールがあります。
 コントロールオブジェクトを使用して、追加のコントロールを統合することができます。

ライブラリを使用して画像を効率よく作成できます。ライブラリには、ドラッグとドロップを使用して画像に挿入できるグラフィックオブジェクトがあります。

[グラフィックデザイナ]エディタで、外部のグラフィックプログラムのグラフィックをインポートすることも可能です。

6.3.1 ライブラリからのグラフィックオブジェクトの挿入

概要

次のステップは、SVG ライブラリからのグラフィックオブジェクトをプロセス画像「START.Pdl」に挿入する方法を示しています。

SVG ライブラリは、[グラフィックデザイナ]エディタのコンポーネントです。

このコンポーネントは、グラフィックオブジェクトを保存、管理する多用途のツールです。 ライブラリは、2つのエリアに分かれています。

- グローバルライブラリ
 [グローバル SVG ライブラリ]エリアは、ディレクトリツリーの構造です。
 マシンやシステムのパーツ、測定デバイス、操作コントロールやビルなどの様々な事前
 に作成されたグラフィックオブジェクトを提供します。
- プロジェクトライブラリ
 [プロジェクト SVG ライブラリ]エリアには、カスタムプロジェクトを保存できます。

"Quick_Start"プロジェクトでは、[グローバル SVG ライブラリ]エリアのグラフィックオブ ジェクトだけが必要です。

これらのオブジェクトを使用して、「START.Pdl」プロセス画像で水処理システムを示します。

?

VV

6.3 プロセス画面の編集

手順

K WinCCExplorer - C:\Users\Public\Documents\Siemens\WinCCProjects\Quick_Start\Quick_Start.MCP ファイル(F) 編集(E) ビュー(V) ツール(T) ヘルプ(H) ^ [⊋ [⊃ 1 X 🔏 Quick_Start $\leftarrow \rightarrow \uparrow$ Å▲ ...\Gr 1-9 J/L° 1-9 917° 名前 AS START.Pd プロセス画像 見 タグ管理 1.0 テキストおよびり、ラフィックリスト 🏠 እጋ-ዋם 🛴 90°0‡°20° EV LA® - トテ゛サ゛イナ 7) 0°0-1°1, NZ097°1 Quick_Start\0*574905*#*47\ 外部タグ: 0 / ライセンス: 102400 ライセンスされてい 2. [SVG ライブラリ]ウィンドウが表示されない場合、ビューを有効にします。 👉 Graphics Designer - START.Pdl File Edit View Arrange Tools Window DataConnector Help Toolbars Þ ✓ Alignment Palette Co. 100% - : 2 er Palette STAR Full Screen olor Palette START.Pd Grid... **Object Palette** Zoom ~ Font Palette Layers... Standard Palette ~ Language... ~ Status Bar 4 Zoom Palette Update references F5 ✓ Output Window Symbol library SVG library * ~ Controls

Dynamic Wizard

Standard

Object properties

Process Pictures

~

~

~ Styles ✓ Tags Reset

1. [グラフィックデザイナ]エディタで、「START.Pdl」プロセス画像を開きます。

[SVG ライブラリ]ウィンドウが開きます。

3. グローバル SVG ライブラリ"IndustryGraphicLibraryV2.0"を[SVG ライブラリ]ウィンドウで 選択します。



ライブラリのツールバーのがボタンを使用して、使用可能なオブジェクトのプレビューを 表示します。

とごボタンを使用して表示されたシンボルのサイズを変更することができます。



4. [Digester]オブジェクトを[Water Wastewater]フォルダから挿入します。

バイオコンバータの図が、作業エリアに表示されます。 マウスで表示されたポイントのいずれかをクリックアンドドラッグして画像を拡大します。

5. 他のオブジェクトの表示を追加します(例、パイプラインやバルブ)。 マウスまたはカーソルキーを使用して、プロセス画像上でオブジェクトの画像を移動します。



結果

水処理システムが描画されました。



次のステップ

プロセス画像で「静的テキスト」オブジェクトを挿入することにより、示されたプロセス にラベルを付けます。

6.3.2 [スタティックテキスト]グラフィックオブジェクトの挿入

概要

次のステップは、"スタティックテキスト"オブジェクトを挿入および編集する方法を示しています。 "スタティックテキスト"オブジェクトは、ランタイムで変化しないテキストフィールドです。 "Quick_Start"プロジェクトでは、表示されたプロセスにラベル付けするスタティックテキ ストが必要です。複数の画像を作成する場合は、示されたプロセスに明確なラベルを付け ることが重要です。

必要条件

• [グラフィックデザイナ]エディタで、プロセス画像「START.Pdl」が開いている。

手順

- 標準 ▼ ₽ X START.PdI X -~ ▶ 選択 ~ 📄 🔜 標準打ジェ外/線 1 _____多角形 -16 · ホッライン ● 楕円 🍊 楕円セゲメント -- いイセグメント ○ 楕円弧 の円弧 ── 矩形 長方形(角なし) A スタティックテキスト 2 ያሳትር ግራ スマートオブジェクト - アフリケーションウィンド v < > ער-מאנב xoo SVGライフ・ラリ OLE OLEオフジェクト **▼** ₽ X ·0.12 10フィールト 📸 | X 🖻 🖻 🗙 📔 📴 🐼 -"N = 📄 ว่าม-ภามSVGライフ รีป 🔝 グラフィックオブジェクト IndustryGraphicLibraryV1.0 💐 ステータス表示 industryGraphicLibraryV2.0 フ[®]ロジェクトSVGライフ・ラリ 🛔 複数行テキスト ↓ ■ コンホホシックス
- 1. [スタティックテキスト]オブジェクトを挿入するには、[標準]ウィンドウでオブジェクトを クリックしてから、プロセス画像をクリックします。

テキストフィールドが、プロセス画像に表示されます。

2. マウスを使用してテキストボックスをドラッグし、フォントタイプとサイズを設定します。



- 3. テキストフィールド内をダブルクリックし、キーボードを使用して「Wastewater Treatment Ltd」などのタイトルを入力します。
- 4. テキストに合わせて、テキストボックスのサイズを調整します。
- 5. ツールバーの J ボタンを使用して、プロセス画像「START.Pdl」を保存します。
- 6.「START.Pdl」プロセス画像を閉じます。

結果

スタティックテキストフィールドが追加され、表示されたプロセスにラベルが付きます。



次のステップ

- プロセス画像の作成と編集をもっと練習する場合、追加のプロセス画像「SAMPLE.Pdl」 を作成するオプションがあります。
- 入力フィールドを挿入し、内部タグを使用してそれをバイオコンバーターのグラフィ ックにリンクすることにより、「START.Pdl」プロセス画像でプラントをダイナミック化 します。

6.3.3 「SAMPLE.Pdl」プロセス画像の編集

概要

次のステップで、「SAMPLE.Pdl」プロセス画像を編集します。 編集中に、ライブラリのグラフィックオブジェクトを使用してプロセスを表示します。 プロセスの表示は自由練習です。"Quick_Start"プロジェクトでは、このステップは必要あ りません。

必要条件

- 「SAMPLE.Pdl」プロセス画像が作成されている。
- [グラフィックデザイナ]エディタが開いている。

手順

- 1. [グラフィックデザイナ]エディタのツールバーの デボタンを使用して、「SAMPLE.Pdl」プロ セス画像を開きます。
- 2. ライブラリのグラフィックオブジェクトを使用して、例えば、ステータスディスプレイで ポンプを示します。
- 3. ツールバーの Hボタンを使用して、プロセス画像「SAMPLE.Pdl」を保存します。
- 4. 「SAMPLE.Pdl」プロセス画像を閉じます。

結果

SAMPLE.Pdi X

「SAMPLE.Pdl」プロセス画像を編集しました。

下記も参照

ライブラリからのグラフィックオブジェクトの挿入(ページ56)

6.4 プロセス画面のダイナミック化

概要

このセクションでは、プロセス画像をダイナミック化する方法を説明します。

原理

「Quick_Start」プロジェクトで、直接タグ接続によって「START.Pdl」プロセス画像をダイ ナミック化します。

タグ接続

直接タグ接続では、1つのタグをプロセス画像のダイナミックオブジェクトに接続します。

ランタイムでタグが値を取ると、この値はダイナミックオブジェクトに直接転送されます。 オブジェクトの動的表示は、タグ値に従ってランタイムで変化します。

実際には、プロセス画像のダイナミックオブジェクトはプロセスタグに接続されます。 WinCC とオートメーションシステム(AS)の間に接続がある場合、AS はプロセスタグに値を 提供します。ダイナミックオブジェクトは、ラインタイムでプロセス値の変化を示します。

[グラフィックデザイナ]エディタで、ASに値を転送するオブジェクトを設定することもできます。ASは、転送された値を使用して、プロセスを制御します。

内部タグを使用したダイナミック化

"Quick_Start"プロジェクトには、AS は必要ありません。このプロジェクトでは、内部タグ [Tank Level]をバイオコンバータのグラフィック画像に接続します。

内部タグに値を指定するには、I/O フィールドを設定します。I/O フィールドは、タグ値を 表示、変更するために使用する入力および出力フィールドです。

ランタイムに I/O フィールドの値を入力すると、この値は内部タグ「Tank_Level」によって 受け取られます。内部タグは、入力された値をバイオコンバータを表すグラフィックオブ ジェクトに転送します。充填レベルインジケータが、タグ値に従って変化します。

WinCC Runtime の実行

プロジェクトが実行されたときに、WinCC Runtime が起動します。

WinCC Runtime は、プロジェクトをプロセスモードで実行します。すると、プロジェクト はランタイムになります。

ランタイムのプロセスのオペレータコントロールおよびモニタリングを実行します。WinCC エクスプローラでランタイムのプロパティを指定します。

6.4.1 フィルレベルインジケータのダイナミック化

概要

次のステップは、バイオコンバータの充填レベルインジケータをダイナミック化する方法を 示しています。

フィルレベルインジケータのダイナミック化には、以下のステップが含まれます。

- バイオコンバータのグラフィック画像と内部タグ[Tank_Level]の接続 内部タグ[Tank_Level]への接続によって、バイオコンバータを表すグラフィックオブジェクトにタグ値を転送できます。充填レベルインジケータが、ランタイムでタグ値に従って変化します。
 タグとオブジェクト間のリンクが、♥アイコンと太字で"オブジェクトプロパティ"ウィンドウに表示されます。
- 更新サイクルの指定
 更新サイクルは、フィルレベルインジケータを更新する時間間隔を決定します。

必要条件

- •「START.Pdl」プロセス画像が作成されている。
- 内部タグ"Tank Level"が作成されている。
- SVG オブジェクト[Digester]が「START.Pdl」プロセス画像に、バイオコンバータのグラフィックの説明として挿入されていること。

手順

- 1.「START.Pdl」プロセス画像を開き、[Digester]オブジェクトをクリックします。
- 2. [シンボルプロパティ]グループを[オブジェクトプロパティ]ウィンドウで開き、 [DisplayFillLevel]属性を[なし]から[あり]に変更します。



バーグラフが[Digester] SVG オブジェクトで、充填レベルインジケータとして表示されます。

3. [タグ]ダイアログを開き、[FillLevelValue]属性をタグとリンク付けします。

| オブジェクトプロパティ | • | џх |
|---|--|----|
| 7~111~77 イベント テキスト アニ | メーション | |
| SVGオフジェクト 形状 色 之刻イル 点滅 その他 塗りつぶし グラフィック ジンボ ¹ ルフ ¹ ロパティ 効果 | 居性 スタティック ダ*イナミック サイクルの更新 間接 BasicColor DisplayFillLevel FillLevelColor FillLevelValue | |
| ॑॓॓॓ऻ॒ ॑॑ॻॱऀジュウトフ⁰ロハ°ティ 🛥 ᠀シ | ■出力ウィントウ □ ライフ・ラリ □ SVGライン | |

4. 内部タグ"Tank_Level"を選択します。 必要な場合、"データソース"エリアで"WinCC タグ"オプションを選択します。



"FillLevelValue"の透明な電球が緑色に変わります。 "シンボルプロパティ"プロパティと"FillLevelValue"属性が太字で表示されます。

5. 充填レベル更新サイクルを"5秒"の値に設定します。



結果

内部タグ[Tank_Level]がバイオコンバータのグラフィック画像に接続されます。この接続に よって、タグ値がグラフィックオブジェクトに転送されます。

次のステップ

I/O フィールドに、値の入力と出力に使用する「START.Pdl」プロセス画像を挿入します。

6.4.2 I/O フィールド(グラフィックシステム - プロセス画像)の挿入とダイナミック化

概要

次のステップは、I/O フィールドを挿入し、ダイナミック化する方法を示しています。 I/O フィールドは、タグ値を表示、変更するために使用する入力/出力フィールドです。

I/O フィールドのダイナミック化には、以下のステップが含まれます。

- I/O フィールドと内部タグ"Tank_Level"の接続
 I/O フィールドを、"Quick_Start"プロジェクトの内部タグ"Tank_Level"と接続します。これにより、I/O フィールドとバイオコンバータのグラフィックの間に間接的接続が確立されます。
 ランタイムに I/O フィールドの値を入力すると、この値は内部タグ"Tank_Level"によって取得されます。タグは、値をバイオコンバータを表すグラフィックオブジェクトに転送します。バイオコンバータの充填レベルインジケータが、タグ値に従ってランタイムに変化します。
- 更新の定義
 更新によって、I/O フィールドの表示が更新される時間間隔を定義します。
- 属性[下限値]および[上限値]の定義
 属性[下限値]および[上限値]によって、I/Oフィールドへの入力を一定の値範囲に制限できます。設定した値範囲外の値はシステムによって拒否され、表示されません。

必要条件

- [グラフィックデザイナ]エディタで、プロセス画像「START.Pdl」が開いている。
- 内部タグ"Tank_Level"が作成されている。
手順

- [I/O フィールド]スマートオブジェクトを挿入するには、[標準]ウィンドウでオブジェクト をクリックしてから、プロセス画像をクリックします。
 I/O フィールドが、作業エリアに表示されます。[I/O フィールド設定]ダイアログが開きます。
- 2. 新しい I/O フィールドに"Tank_Level"タグをリンクします。 更新サイクルとして 500 ミリ秒、フォントサイズとして 24 を選択します。



I/O フィールドのショートカットメニューから[設定ダイアログ]を選択することで、[I/O フィールド設定]ダイアログを再度開きます。

"出力/入力"プロパティが"オブジェクトプロパティ"ウィンドウで太字で表示されます。ここで、内部タグ[Tank_Level]が I/O フィールドに接続していることを確認できます。

"オブジェクトプロパティ"ウィンドウと同様に、"I/O フィールド設定"ダイアログでタグへの 接続を作成できます。

3. [制限]グループを[オブジェクトプロパティ]ウィンドウで開き、[下限値]を0に設定します。

| オブジェクトプロパティ | | | | | | ▼ ‡ × |
|---|-----------------------|--|--------------------------|---------|--------------|-------|
| 7°ロハ°ティ イベント テキスト ア. | ニメーション | | | | | |
| 107ィールト[*] 一形状 一色 一次ケイル フォント 一点滅 一での他 1 、 、 、 、 、 、 、 、 、 、 、 、 、 | 属性 下限値 上限値 이 | 297199 1 707602- 209 97 8 OK | タドイナミック × ノセル 3 | サイクルの更新 | 間接 □ □ | |

- 4. ステップ3と同様に、[上限値]を100に設定します。
- 5.「START.Pdl」プロセス画像を保存します。
- 6. [グラフィックデザイナ]エディタを閉じます。

別の手順

- 1. [QuickStart_Tags]タググループの[タグ]ウィンドウで、[Tank_Level]タグを選択します。
 - START.PdI X ~ 0,000 3 Wastewater Treatment Ltd. < 3 わが , д X 觞 711/7: Image: A state of the state of □ 個別 WinCC かう 名前 \$17° ハッラメータ コパノ 署えまれていない16-ビット値 💼 Tank_Level 🔊 Performance ProcessHistorian TagLoggingRt - Script → ● 全構造インスタンスのり >
- 2. タグをプロセス画像にドラッグアンドドロップします。

I/O フィールドが選択されたタグに対して作成されます。I/O フィールドは、[Tank_Level]タ グと接続されています。

- 3. I/O フィールドのショートカットメニューから"設定ダイアログ"を選択することで、"I/O フィールド設定"ダイアログを開きます。
- 4. 更新サイクルとして 500 ミリ秒、フォントサイズとして 24 を選択します。
- 5. I/O フィールドを希望する位置に移動するには、マウスまたはキーボードの矢印キー(カー ソルキー)を使用します。 上記手順のステップ3に進みます。

結果

I/O フィールドを挿入し、これを内部タグ"Tank_Level"に接続します。

ランタイムに I/O フィールドの値を入力すると、この値は内部タグによってバイオコンバータのグラフィック画像に転送されます。充填レベルインジケータが、入力された値に従って変化します。

属性[下限値]および[上限値]によって、I/Oフィールドの値範囲が定義されています。

ランタイムで I/O フィールドに値"0"を入力すると、空のバイオコンバータが表示されます。 ランタイムで I/O フィールドに値"100"を入力すると、満水のバイオコンバータが表示され ます。値範囲外の値は、システムで拒否されます。



次のステップ

- ランタイムの設定済み画像間の変更を行なえるようにするため、ユーザー定義メニューを作成します。
- 「START.Pdl」プロセス画像のダイナミック化を表示するために、WinCC Runtime のプロ パティを定義し、「Quick_Start」プロジェクトを有効にします。

概要

このセクションでは、カスタマイズされたメニューおよびツールバーを使用した画像の作成方法を説明します。

原理

[メニューとツールバー]エディタでは、カスタマイズされたメニューとツールバーを設定 できます。

カスタマイズされたメニューおよびツールバーは、WinCC の"コンピュータ"エディタでプ ロジェクトに割り当てる設定ファイルに保存されます。

グローバルスクリプト手順を使って、メニュー項目とシンボルを接続します。

追加のコンフィグレーションの選択肢:

- 権限の割り当て
 ログインしたオペレータは、対応する設定エレメントを表示するには、必要な権限を持っている必要があります。
- メニューコマンドとシンボルの非表示化または無効化 異なる機能を対象とする2つの設定ファイルを作成します。 適用例:ランタイムでユーザーが変更すると、現在の設定ファイルが2番目のファイルに 置き換えられます。

6.5.1 カスタマイズされたメニューおよびツールバーのプロシージャの作成

概要

次のステップは、グローバルスクリプトのモジュールでのプロシージャの作成方法を示しています。

カスタマイズされたメニューおよびツールバーを機能させるには、2 つのプロシージャが 必要になります。

- ActivatePicture(ByVal PictureName):
 "PictureName"パラメータと一緒に転送される画像に対して、画像の変更を実行します。
- StopRuntime(ByVal Item):
 ランタイムを終了します。

手順

1. VBS-Editor を開きます。



2. [プロジェクトモジュール]タブを選択し、以下のプロシージャコードを書き込みます。 Sub ActivatePicture (Byval PictureName)

Dim objScreen Dim strScreenName

```
strScreenName = PictureName.Userdata
HMIRuntime.BaseScreenName = strScreenName
```

End Sub

プロセス画面の設定

6.5 ユーザー定義メニューおよびツールバーの作成



- 3. [ファイル]>[保存]を使用して、モジュールを「MenuToolbars_Commands.bmo」ファイル として保存します。
- 4. 新しいプロシージャを挿入します。



5. 名前"StopRuntime(ByVal 項目)"を入力します。



以下のプロシージャコードを書き込みます。
 Sub ActivatePicture (Byval PictureName)

```
Dim objScreen
Dim strScreenName
```

'Userdata contains the screen name specified in editor menus and toolbars

```
strScreenName = PictureName.Userdata
HMIRuntime.BaseScreenName = strScreenName
```

```
End Sub
Sub StopRuntime (ByVal Item)
HMIRuntime.Stop
```



モジュールを保存します。
 VBS-Editorを閉じます。

結果

プロシージャ"ActivatePicture(ByVal PictureName)"および"StopRuntime(ByVal Item)"が作成 されます。

次のステップ

- 「START.Pdl」および「SAMPLE.Pdl」の画像に画像変更を実行するカスタマイズされた メニューを作成します。
- カスタマイズされたツールバーを使用して、ランタイムを終了します。

6.5.2 画面変更のためのカスタマイズされたメニューの作成

概要

次のステップは、2つのメニュー項目[開始]と[サンプル]を持つ、カスタマイズされたメニュー[画像変更]を作成する方法を示しています。

メニューエントリ"開始"と"サンプル"をプロシージャ"ActivatePicture(ByVal PictureName)"にリンクさせます。

変更するプロセス画像の名前を、"ユーザーデータ"ボックスに入力します。

必要条件

• プロシージャ"ActivatePicture(ByVal PictureName)"が作成されている。

手順

1. [メニューとツールバー]エディタを開きます。

2. 新しいメニュー項目を作成します。



メニュー項目に"Picture Change"という名前を付けます。
 "プロパティ"ウィンドウでは、"テキスト"エリアにその他のランタイム言語の名前を入力します。

| T Menus and toolbars - V | VinCC Configuration (| Studio | | | - 0 | × |
|--------------------------|-----------------------|-------------------|-----|-----------------|------------------|-----|
| Eile Edit | <u>H</u> elp | | | | | |
| Menus a | Find | + م | | Properties | s - Menu item | * |
| Henus | Name | Text ^ | | Object type | Menu item | ^ |
| Picture Change | 1 Picture Chang | e Picture Chan | | Object name | Picture Change | - |
| | 2 💥 | | | General | are al | |
| | 3 | A12 | | Name | Picture Change | _ |
| | 4 | | | Text | Picture Change | |
| | 5 | | | Active | | _ |
| | 6 | | | Visible | | |
| | 7 | | | Separator | | |
| | 0 | | | Authorization | | |
| | • • | | | User data | | |
| | 9 | | | Script | | |
| | 10 | | | Picture | | |
| | 11 | | | Source | | |
| ***** | 12 | _ | E | Text | | |
| Computer | 13 | | | Text (DEU) | Picture Change | |
| | 14 | | | Text (ENU) | Picture Change | |
| Tag Management | 15 | | | Text (ESP) | Picture Change | _ |
| 6.N | 16 | | | Text (FRA) | Picture Change | |
| Alarm Logging | 17 | | | i she (i to i) | | |
| | 19 | ~ | | Text in Languag | e Spanish (Spain | nal |
| | Menü | | 111 | Sort) | ,p | |
| Ready | Englis | h (United States) | Tab | le: 1 Menu item | 100 % 🕞 – 🖓 – | -+ |



4. [スタート]メニューコマンドを作成します。

5. メニュー項目[開始]を設定して、「START.Pdl」画像に画像変更を実行できるようにします。

| 👕 Menus and toolbars - W | inCC Config | guration Studio | | | | | | - | | \times |
|---|--------------|-----------------|------------|------|-------|---------------|------------|------|--------------|----------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | | | | | | | | | |
| Menus and to « | Find | | Q | • | 8 | Propertie | s - Menu | item | | * |
| 📮 👕 Menus | Nam | e Text | Active | 1 | | Object type | Menu item | 1 | | ^ |
| 🖃 👕 Picture Change | 1 Start | Start | 1 | | | Object name | Start | | | |
| Start | 2 💥 | | | | Θ | General | | | | |
| | 2 | | | | | Name | Start | | | |
| Function selection | | 4 | | x | | Text | Start | | | |
| - Meil Durth an Miterial an | | -(()) | | - 1 | | Active | V | | | |
| | Name | Prot | otype | 100 | | Visible | 2 | | - | |
| Mienu looibars_u | or 🛛 🖻 Ac | tivatePict Sub | ActivatePi | | | Separator | | | 2 | |
| I I I I I I I I I I I I I I I I I I I | 🛛 🖾 Sto | pRuntime Sub | StopRunti | | | Authorization | | | `(() |) |
| | > < | | | > | | User data | START | | C | |
| | | | | _ | | Script | | A T | | |
| Apply | | ок | Cancel | | | Picture | | AI | | |
| 9. ÿ 5 0 & 4 · | | Men | > | • | | Source | Picture Ch | ange | | |
| Ready | | English (Unite | d States) | Tabl | le: 1 | Menu item | 100 % 🕞 | (| | ÷., |

6. 同じ手順を使用して、「SAMPLE.Pdl」画像に画像変更を実行するメニュー項目[サンプル]を 作成します。

7. [ファイル]>[保存]を使用して、設定に「ActivatePicture」という名前を付けて保存します。

結果

2つのメニューコマンドを持つ[画像変更]メニューが作成されます。

ランタイムでこれらメニューコマンドを使用して、「START.Pdl」と「SAMPLE.Pdl」のプロ セス画像を変更することができます。

| 👕 Menus and toolbars - W | inCC | Configuration S | itudio | | | | | - | × |
|---|-------------|-----------------|--------------|--------|----------|------|---------------|-----------------|----------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> el | р | | | | | | | |
| Menus and to « | Fin | d | | Q | • | Ï | Propertie | s - Menu item | » |
| 🖃 👕 Menus | | Name | Text | Active | > | | Object type | Menu item | ^ |
| 🖃 👕 Picture Change | 1 | Start | Start | 1 | | | Object name | Sample | |
| - Start | 2 | Sample | Sample | 1 | | Ξ | General | | |
| Sample | 3 | * | | | | | Name | Sample | |
| Teelbaw | 4 | | | | | | Text | Sample | |
| | 5 | | | | | | Active | | |
| | 6 | | | | | | Visible | | |
| | 7 | _ | | | | | Separator | | |
| | <u>_</u> | | | | | | Authorization | | |
| | 8 | | | | | | User data | SAMPLE | |
| Commuter | 9 | | | | | | Script | ActivatePicture | - |
| Computer | 10 | | | | | | Picture | | |
| | 11 | | | | × | | Source | Picture Change | |
| | H 4 | 🕩 🕨 🔤 Menu | it < | > | : | - | T | | * |
| Ready | | English | n (United St | ates) | Table | e: 2 | Menu items | 100 % 😑 🔤 | Ð., |

6.5.3 ランタイムを終了するカスタマイズされたツールバーの作成

概要

次のステップは、ランタイムを終了するアイコンがある、カスタマイズされたツールバーの 作成方法を示しています。

シンボルのグラフィックファイル

シンボルとして表示するグラフィックファイルを次のフォルダにコピーします。

• <プロジェクトフォルダ>\GraMT\mtimages

シンボルとして、BMP、PNG、JPG または GIF フォーマットのグラフィックを使用します。

必要条件

- [ActivatePicture]設定が、[メニューとツールバー]エディタで開いていること。
- "StopRuntime(ByVal 項目)"プロシージャが作成されていること。
- グラフィックファイルが"ランタイムの終了"のシンボルとして WinCC プロジェクトフ オルダにコピーされていること。例:
 - C:\Users\Public\Documents\Siemens\WinCCProjects\Quick_Start\GraMT\mtimages

手順

1. 新規ツールバーを作成し、「WinCC Runtime」を名前として入力します。

| T Menus and toolbars [ActivatePicture] - WinCC Co | nfiguration Studio | 194 | |
|--|--------------------|---------|------------------|
| <u>File E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | |
| Menus and toolbars [ActivatePictur « | Toolbar Find | d | » - ۹ |
| 🖃 📲 Menus | Name | Text | ~ . |
| 🗄 👕 Picture Change | 1 💥 | 1 | - To |
| Toolbars | 2 | | Per |
| New toolbar | 3 | | tie |
| Import graphics | 4 | | |
| | 5 | | |
| Paste | 6 | | |
| | 7 | | |
| | 8 | | ~ |
| | II I I I Toolbars | / . | > |
| Ready NUM G | erman (Germany) | 100 % 😑 | - \ + |

2. デフォルトで画像の上部端に表示されるようにツールバーを設定しますが、ユーザーはこの 位置を変えることができます。

| T Menus and toolbars [Ac | ctivatePicture] - WinCC | Configuration Stud | lio | | - 🗆 X |
|---|--------------------------|--------------------|----------------|------------------|------------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | | | | |
| Menus and to « | Find | | ب م | *** Propertie | es - Toolbar 🛛 » |
| 🗐 📲 Menus | Name | Text | Active ^ | General | ^ |
| 🐵 👕 Picture Change | 1 WinCC Runtime | WinCC Runtime | | Name | WinCC Runtime |
| Toolbars | 2 💥 | | | Text | Wip 3 ime |
| | 3 | | | Active | |
| | 4 | | | Visible | |
| | 5 | | | Fixed | |
| | 6 | | | Authorization | |
| | 7 | | | User data | 4 |
| | 0 | | | Alignment | Up (🍋) |
| | 0 | | | Mode | Left |
| Computer | 9 | | | Picture size | Up |
| | 10 | | ¥ | Masking color | Bottom |
| ポウ 42 50 40 6 0 . | Symbole | eister I < | > | 🗆 Text | V |
| Ready | G | erman (Germany) | Table: 1 To | oolbar 100 % (| + |

3. [ランタイムの停止]エレメントを作成し、ランタイムを停止するアイコンを設定します。 この例では、グラフィック「stoprt.png」がアイコンとして使用されています。



4. [ActivatePicture] 設定を保存します。

5. "メニューとツールバー"エディタを閉じます。

結果

ランタイムを終了するアイコンのあるツールバーが作成され、設定が保存されます。

次のステップ

ランタイムでメニューバーおよびツールバーを表示するには、ランタイムのプロパティを 指定しているときに設定ファイルを選択します。

そうすると、ランタイム時に、それぞれのプロセス画像にはユーザー定義のメニューとツ ールバーが表示されます。

「スタートガイド」のコースで追加のプロセス画像を生成する場合、ここに示すステップを 使用して追加の図を含むようにメニューの設定を拡張できます。

6.6 ランタイム中のプロジェクトの実行

6.6.1 ランタイムタイムプロパティの定義(グラフィックシステム - プロセス画像)

概要

次のステップは、WinCC Runtime のプロパティの定義方法を示しています。

WinCC Configuration Studio の[コンピュータ]エディタで WinCC Runtime のプロパティを 設定します。

プロジェクトを有効にするときにグラフィックランタイムが実行されるように WinCC Runtime をセットアップします。

「START.Pdl」プロセス画像を、ランタイムウィンドウの初期画像として選択します。

作成した設定ファイルをプロセス画像でユーザー定義メニューおよびツールバーを表示す るために読み込みます。

必要条件

• "Quick_Start"プロジェクトが開いている。

手順

 ショートカットメニューで[コンピュータ]と[プロパティ]のエントリのコンピュータ名を選択 します。



[コンピュータ]エディタが開かれます。

2. [スタートアップ]タブの[グラフィックスランタイム]アプリケーションを選択します。

| Computer properties | × |
|---|------|
| General Startup Parameters REST settings Graphics Runtime Runtime WinCC Runtime S Global Script R Alarm Logging Runtime Tag Logging Runtime Report Runtime Graphics Runtime Parameters REST settings Graphics Runtime Tag Logging Runtime Report Runtime Parameters REST settings Graphics Runtime Report Runtime Setting Runtime Parameters REST settings Graphics Runtime Report Runtime Setting Runtime Setting Runtime Report Runtime Setting Runtime Report Runtime Runtime Report Runtime Report Runtime Report Runtime Report Runtime Report Runtime Runtime Runtime Report Runtime | Edit |

3. [グラフィックランタイム]タブで、開始画像としてプロセス画像「START.Pdl」を定義します。

| General Startup | Parameters | REST settings | Graphics Runtime | Runtime | | |
|------------------------------------|--------------|---|--|---------------------|----------|---|
| Project File: C:\Users\Public | \Documents\S | oiemens\WinCCF | | Quick_Start.r | ncp | |
| Start Picture: Window Attribute | es: | | Start Configuration ActivatePicture.m | n Menu and T ntl | oolbars: | |
| 📧 Selection (| of picture | (| U | | ? | Х |
| B- B-B- | | | | | | |
| Hierarchy: | | | | | | |
| Hierarchy: | DOCU-01 | File Name * | | | ; | 7 |
| Hierarchy: | DOCU-01 | File Name * -†r SAMPLE -†r <mark>START.P</mark> -†r Tag_Log | E.Pdl dl agin 3 | | | 7 |

- 4. メニューおよびツールバーの設定ファイル"ActivatePicture.mtl"を選択します。.
- 5. "タイトル"、"最大化"、"最小化"および"スクロールバー"ウィンドウ属性を有効にします。

| Compute | r propert | ies | | | |) |
|-----------|------------|------------|----------------|---|----------------------|---|
| General | Startup | Parameters | REST settings | Graphics Runtime | Runtime | |
| Project | File: | | | | | |
| C:\Use | rs\Public\ | Documents\ | Siemens\WinCCF | ^p rojects\Quick_Start ^v | \Quick_Start.mcp 👔 🔪 | |
| Start Pic | cture: | | | Start Configuration | n Menu and To | |
| START | .Pdl | | | ActivatePicture.n | ntl | |
| Window | Attribute: | 3: | | Turn Off: | 75.0 | |
| 🗹 Tit | le | | ^ | Alt-F4 | ^ | |
| 🗌 🗆 Bo | rder | | | 🗌 Resize | | |
| 🛛 🗹 Ma | aximize | | | Move | | |
| 🗹 Mi | nimize | | | 🔲 Minimize | | |
| 🗌 🗆 Fu | II Screen | | | 🔲 Maximize | | 1 |
| 🗹 Sc | roll Bar | | | Close | ~ | |
| | | 2 | | | | |
| Hide Sy | stem f 🛛 🛓 | | | Independent Pict | ure Windows: | |
| Use | prefix 🗸 | 9 @ | | 🗌 Hide main win | dow | |

結果

WinCC Runtime のプロパティが定義されています。

"Quick_Start"プロジェクトが有効にされている場合、[グラフィックランタイム]コンポー ネントが実行されます。開始画像として、「START.Pdl」プロセス画像が表示されます。 ランタイムをナビゲート、終了するために、すべてのプロセス画像で、カスタマイズされ たメニューとツールバーが表示されます。

このウィンドウ属性を使用して、ランタイムウィンドウの追加機能を決定します。

次のステップ

「Quick Start」プロジェクトを有効にします。

6.6.2 プロジェクトの有効化(グラフィックシステム - プロセス画像)

概要

次のステップは、「Quick_Start」プロジェクトを有効にする方法や、ランタイムでの動的 なプロセス画像「START.Pdl」の操作方法を示しています。

プロジェクトを有効にすると、WinCC Runtime が開始します。WinCC Runtime は、プロセスモードでプロジェクトを実行します。

ランタイムで、設定済みの I/O フィールドを操作し、フィルレベルインジケータの変化を 監視します。

必要条件

ランタイムプロパティが定義されている。

手順

1. WinCC エクスプローラの >> ツールバーボタンを使って"Quick_Start"プロジェクトを有効に します。

短いロード時間の後に、ランタイムウィンドウが開きます。「START.Pdl」プロセス画像が表示されます。



- 2. I/O フィールドに 0 から 100 の間の値を入力し、入力内容を<Return>で確定します。 フィルレベルインジケータの変化を監視します。
- 3.「SAMPLE.Pdl」プロセス画像に切り替えるには、[画像変更]メニューで[サンプル]エントリ をクリックします。



- 4. 「START.Pdl」プロセス画像に戻るには、[画像変更]メニューで[開始]エントリをクリックします。
- 5. 設定したツールバーボタンをクリックして、ランタイムを終了します。 あるいは、WinCC エクスプローラツールバーの ボタンを使用して、ランタイムを停止し ます。



結果

"Quick_Start"プロジェクトが有効になり、WinCC Runtime が開始します。動的なプロセス 画像「START.Pdl」がランタイムウィンドウに表示されます。

I/O フィールドに値を入力すると、この値は内部タグ"Tank_Level"によってバイオコンバー タのグラフィック画像に転送されます。これによって、バイオコンバータの充填レベルイ ンジケータを監視することができます。

次のステップ

内部タグ「Tank_Level」をシミュレートするために、「Quick_Start」プロジェクトをテストし、[タグシミュレーション]エディタを使用します。

下記も参照

ランタイムを終了するカスタマイズされたツールバーの作成 (ページ84)

6.6.3 プロジェクトのテスト(グラフィックシステム - プロセス画像)

概要

次のステップは、WinCC タグシミュレータによって「Quick_Start」プロジェクトをテス トする方法を示しています。

WinCC タグシミュレータによって、まだ開発段階であるプロジェクトをテストできます。 テスト中に、オートメーションシステム(AS)に接続した時に、プロジェクトがどのように 動作するかを確認します。

「Quick_Start」プロジェクトで、WinCC タグシミュレータによって内部タグ「Tank_Level」 の値をシミュレーションします。WinCC Configuration Studio の[タグシミュレーション]エ ディタでシミュレーションを設定します。

WinCC タグシミュレータは、内部タグ「Tank_Level」に異なる値を割り付けます。バイオ コンバータのグラフィック描写に内部タグを接続すると、バイオコンバータの充填レベル インジケータはタグ値に従って変化します。

必要条件

ランタイムプロパティが定義されている。

手順

- [タグシミュレーション]エディタを起動します。
 [タグシミュレーション]エディタが開きます。
- 2. タグ選択ダイアログを開き、内部タグ[Tank_Level]をタググループ[QuickStart_Tags]から選 択します。

| - | Is [Simulation | 1 | | | F | Find | | | ب م | - 8 |
|------------------|-----------------|---------------|----------|-----------|----------------------|--------------|-----------|---------|----------------|-----|
| ٦ | Fag name | Data type | Function | Cycle | Active Connect | tion name (| Value set | Quality | se | - |
| | X | | | | | | | | | 4 |
| 2 | | | | | | | | | | |
| 3 | Tag selection | | | | | | × | | | |
| | Filter: | | | | | V 🗈 🕽 | | | - 11 | |
| | 🕞 👘 WinCC Ta | as | Nat | ne | Type | | Paran | | - 11 | |
| | 🛓 📙 🔒 Interna | al tags | * | inc. | al * | | | | | |
| | 🗍 🗍 Qu | iickStart_Tag | s 🛛 | | и 1. простокала и | re instancia | 1 | | - 11 | |
| | 📄 🗄 📥 🦰 🏹 | 2nce | | iank_Leve | al Unsigned | lo-bit Value | | | | |
| | | Historia | in | | | | | | | |
| | 📄 📄 🔒 🦒 | ggingRt | | | | | | | | |
|) | | | - 1 | | | | | | | |
|) L 2 | | | | | | | | | | |
|) L 2 3 | | | | Annly | ОК | Car | ncel | | | |

3. シミュレーションタイプ[インクリメント]を選択します。

| ł | ags [Simulatio | on] | | Fin | d | e | | Q | • |
|--------|-----------------|-----------------------|--|-------|--------|-----------|--------|--------|---|
| | Tag name | Data type | Function | Cycle | Active | Connectio | n name | (Value | |
| L | Tank_Level | Unsigned 16-bit value | User input | - | | | | | |
| 2 3 | * | 1 | Sine Oscillation Random | | | | | | |
| 5 | | | Increment Decrement User input Script | | 2 | | | | |

| 4. | [プロパティ |]ウィンドウを開き、 | シミュレーションタ | イプのプロパティを定義します。 |
|----|--------|------------|-----------|-----------------|
|----|--------|------------|-----------|-----------------|

| Tag | s [Simula F | ind | <mark>ب م</mark> | F | Properties - Tag | -1 |
|-----|--|-----------------------|------------------|---|--------------------------------|-------------|
| Т | ag name | Data type | FL | | Selection | |
| | Tank_Level | Unsigned 16-bit value | In | | Object type | Tag |
| | N. Contraction of the second s | 1997. | | | Object name | Tank_Level |
| | | | | | Tag | |
| | | | | | Tag name | Tank_Level |
| | | | | | Data type | Unsi 25-bit |
| | | | | | Function | In(🎒) |
| | | | | | Cycle | 1 |
| | | | | | Active | |
| | | | | | Connection name (internal tag: | |
| | | | | Œ | Sine function | |
| | | | | Ð | Oscillation | |
| | | | | Ð | Random value | |
| - | | | | | Increment | |
| 3 | | | | | Initial value Increment | 0 |
| 1 | | | | | End value Increment | 100 |

5. シミュレーションを"QuickStart_Simulation"として保存します。

6. >ボタンを使用して、WinCC エクスプローラで WinCC Runtime を有効にします。

7. [タグシミュレーション]エディタでシミュレーションを起動します。

| ① <u>F</u> ile | Tags simulation | art_Simulation] - WinCC (Simulation <u>H</u> elp | Configuration Studi | io | | | | × |
|-------------------|--------------------|---|---------------------|----------|--------|-----------------|------|-----|
| Та | gs [Runtime activ | 🕞 Start | | Find | | | ۰ م | - (|
| | Tag name | O Stop | Function | Cycle | Active | Connection name | (Val | Τ |
| 1 | Tank_Level | Unsigned 1 alue | Increment | 1 | | | | 0 |
| 2 | * | | | | | | | |
| 3 | 8 | | | | | | | |
| 4 | | | | | | | | o l |
| 5 | | | | | | | | |
| 6 | | | | | | | | |
| 7 | | | | | | | | |
| 8 | | | | | | | | |
| 14 4 | Tags | 1 | | | | - | | |
| Rea | | Engli | sh (United States) | Table: 1 | Tag | 100 % 🗐 📉 🕻 | , | |

プロセス画面の設定

6.6 ランタイム中のプロジェクトの実行

- [タグシミュレーション]エディタとランタイムウィンドウを、両方を同時に監視できるように位置調整します。 シミュレーション値の変化が、フィルレベルインジケータにどのように影響するかを監視します。 設定更新サイクルは異なる表示の遅延を引き起こします。
- 9. シミュレーションおよび WinCC Runtime を終了します。

| Tag | gs [Simulation a | Start | | Find | | | P | • | - 9 |
|-----|-------------------|-----------------------|-----------|-------|--------|-----------------|------|----|-----|
| - | Tag name | O Stop | Function | Cycle | Active | Connection name | (Val | | 1 |
| | Tank_Level | Unsigned 16-bit value | Increment | 1 | | | | | - |
| | ※ | | | | | | | | |
| | | | | | | | | | 1 |
| | | | | | | | 1 | | |
| | | | | | | | | | |
| | | | | | | | | 1 | |
| | | | | | | | | | |
| | | | | | | | | | |
| 4 | E H Tags | | - II - | | | | | .: | |



結果

WinCC タグシミュレータによって、"Quick_Start"プロジェクトがテストされます。 テストは、プロセス値を提供された時のプロジェクトの動作を示します。

6.6.4 ランタイムシステムダイアログの使用

概要

ランタイムシステムダイアログは、ランタイム中に、頻繁に必要になるアクションを実行 するのに使用できます。画像内のアクションを設定する必要はありません。

可能なアクションには、以下が含まれています。

- ランタイムシステムダイアログボックスを閉じる
- 画像の開始を表示
- 直前の画像を表示
- 次の画像を表示
- お気に入りの画像を表示

ランタイムシステムダイアログを有効にする手順は以下の通りです。 例では、2つの画像を取り替える手順を示しています。

必要条件

- •「START.Pdl」と「SAMPLE.Pdl」のプロセス画像が作成されている。
- 開始画像として、「START.Pdl」プロセス画像が定義されている。

手順

- 1. WinCC エクスプローラで、「START.Pdl」と「SAMPLE.Pdl」のプロセス画像を選択します。 両方の画像を選択するには、クリックするときに<Ctrl>ボタンを押します。
- 2. プロセス画像をお気に入りとしてマーキングします。



「START.Pdl」と「SAMPLE.Pdl」のプロセス画像が WinCC エクスプローラの右ペインにお気 に入りとして表示されます。

3. [コンピュータ]エディタのナビゲーションエリアでショートカットメニューによりコンピ ュータのプロパティを開きます。



[コンピュータ]エディタが開かれます。

[ランタイム]タブの[ランタイプオプション]エリアで、[ランタイムシステムダイアログの有効化]オプションを有効にします。





プロジェクトプロパティが開きます。

6. [ショートカット]タブで、[ランタイムシステムダイアログ]アクションに<Ctrl+R>などのキ ーボードショートカットを割り付けます。

| ϽʹʹϿϿʹϫʹͻͰϽʹʹϿͶʹʹͻィ | | × |
|--|---|---|
| 操作モード 全般 更新 ショートカットとは、 きるキーの組み 1 | ユーザーイン iサイク WinC(し 和 のアクショ 合わせ | ターフェイスとデザイソ カット オフ [®] ション いを書り付けることがで |
| アクション: ログオン ログオフ ハート当と ^{&} ー ランタイムシステムダイアログ | 2 | 割り付け STRG + RI 既に割けている: |

7. "Quick_Start"プロジェクトを有効にします。 短いロード時間の後に、ランタイムウィンドウが開きます。「START.Pdl」プロセス画像が表示されます。

8. <Ctrl+R>などの割り付けられたキーボードショートカットを使ってランタイムシステムダ イアログを起動します。



ランタイムウィンドウが開き、システムメニューが表示されます。

9. [お気に入り]および[SAMPLE]プロセス画像をクリックします。



ランタイムシステムダイアログが非表示になり、「SAMPLE.Pdl」プロセス画像が表示されます。

結果

ランタイムシステムダイアログを有効にし、テストしました。

システムダイアログを使用して作成したプロセス画像間をナビゲートしたり、WinCC Runtime の言語を変更したりすることができます。

7

フェイスプレートタイプとタグ構造の設定

7.1 フェイスプレートタイプの設定

概要

このセクションは、フェイスプレートタイプと構造タグの使用方法に関する情報を提供し、 プロセス画像のために再利用可能なカスタマイズされているテンプレートを設定できるよ うにします。

フェイスプレートタイプと構造タグを使用すると、複数の使用に対してオブジェクトとタ グを設定できます。

フェイスプレートのタイプは個別ファイルとして作成されます。WinCC でのこれらのファ イルの管理は、プロセス画像の処理に対応しています。

原理

フェイスプレートタイプ

プロセス画像とまったく同じ方法で、[グラフィックデザイナ]エディタのフェイスプレー トタイプを設定します。スタティックで動的かつ操作可能なオブジェクトを配置し、それ らをタグにリンクします。これにより、プロセス画像のフェイスプレートインスタンスと して好きな頻度で使用できるテンプレートが生成されます。このタイプインスタンスモデ ルにより、設定の労力が大幅に軽減されます。

複数の使用の場合、フェイスプレートタイプを構造タイプとリンクすることにより匹敵する 個別のタググループをリンクします。

タグ管理:構造タイプ

構造タイプは、タグの特定のコンパイレーションのスキームのためのテンプレートです。 構造タイプは別の構造タグに割り付けられます。この結果、これらの構造タグは常に、構造 タイプと同じ要素で構成されますが、スタンドアロンの構造インスタンスとして構成され ます。

構造タイプがフェイスプレートタイプとリンクされている場合、ドラッグアンドドロップで 構造インスタンスをプロセス画像へドラッグすることにより、フェイスプレートインスタ ンスを配置します。

実際には、構造タイプと関連するインスタンスは、PLC またはサードパーティのアプリケーションからインポートされ、手動で個別に設定する必要がありません。

7.1 フェイスプレートタイプの設定

データレコードを TIA Portal プロジェクトからエクスポートするには、「SIEMENS SIMATIC SCADA Export」ツールを使用します。

Industry Online Support ページで、ツールをダウンロードしたり、追加情報を取得したり できます。

- Industry Online Support: 「TIA Portal の SIMATIC SCADA Export」をダウンロード(ID 109748955) (<u>https://support.industry.siemens.com/cs/ww/en/view/109748955</u>)
- Industry Online Support: 「SIMATIC SCADA Export」ドキュメント(ID 101908495) (https://support.industry.siemens.com/cs/ww/en/view/101908495)

WinCC システムのフェイスプレートタイプとタグ構造



「Quick_Start」プロジェクトの設定

「START.Pdl」プロセス画像は、2つのバルブがあります。これらはグラフィックな表示以外は、これまで何の役割も担ってきていません。

プロジェクトが拡張され、各バルブをクリックすると、ポップアップ画面が表示されるようになりました。ポップアップ画面は、温度や流量などの追加情報を表示します。フェイ

7.1 フェイスプレートタイプの設定

スプレートタイプや構造タイプを使用して、関連のタグやポップアップ画面を複数回作成 しなければならなくなることを回避します。

このセクションでは、次のものを作成します。

- 1つの構造ラベルタイプと2つの構造タグ
- 構造タイプとリンクされている1つのフェイスプレートタイプ
- フェイスプレートタイプとリンクされている1つのポップアップ画像

作成されているフェイスプレートタイプをプロセス画像のフェイスプレートインスタンス として使用します。

フェイスプレートタイプの付加価値を示すために、バルブ状態のカラーによるビジュアル化をプロジェクトに追加します。

下記も参照

Industry Online Support: 「SIMATIC SCADA Export」ドキュメント(ID 101908495) (<u>https://</u> support.industry.siemens.com/cs/ww/en/view/101908495)

Industry Online Support:「TIA Portal の SIMATIC SCADA Export」をダウンロード(ID 109748955) (<u>https://support.industry.siemens.com/cs/ww/en/view/109748955</u>)

7.2 構造タグの作成

7.2 構造タグの作成

7.2.1 構造タイプの作成

概要

次のステップは、構造タイプと関連の構造タイプ要素を作成する方法を示しています。

必要条件

• "Quick_Start"プロジェクトが開いている。

手順

1. 新しい構造タイプをタグ管理の[構造タグ]エントリで作成します。

| ag Managemen | t | * | 📦 Structur | Find | | Q | • |
|-------------------|----------------------|-------|------------|------|---|------|---|
| 🖳 Tag Manageme | nt | | Name | | 0 | Comm | ^ |
| 🗄 🍄 Internal tags | | | 1 派 | | | | |
| 🥂 Structure tags | | | 2 | | | | |
| | S New structure type | M-0 | 3 | | | | |
| | Conu | (🐴 🗍 | 4 | | | _ | |
| | а сору | | 5 | | | | |
| | B Paste | | 7 | | | _ | |
| | | | 7 | | | | |

2. 構造タイプの名前として「Valve_Info」を入力します。

7.2 構造タグの作成

| File Edit View Tools Help Tag Management Image: Tag Management <th>🔩 Tag Managemer</th> <th>nt - WinCC Configuration Studio</th> <th>5</th> <th>- 🗆 🗙</th> | 🔩 Tag Managemer | nt - WinCC Configuration Studio | 5 | - 🗆 🗙 |
|--|---|--|----------------------------------|------------------|
| Tag Management Tag Management Tag Management Tag Management Image: Structure tags Structure tags Valve_Info New structure type element Image: Structure type element | <u>File E</u> dit <u>V</u> iew | Too <u>l</u> s <u>H</u> elp | | |
| Tag Management Internal tags Image: Structure tags <th>Tag Managem</th> <th>ent</th> <th>« ধ Structur Find</th> <th><mark>ہ ج</mark></th> | Tag Managem | ent | « ধ Structur Find | <mark>ہ ج</mark> |
| Valve_Info | □য় Tag Manage ⊡ ♥ Internal t □ ♥ Structure tag | ment ags Is | Name 1 X 2 - | Extern ^ |
| | Computer | New structure type eleme Copy Paste Delete Rename Synort | a b b Structure type elf | |

3.「Valve_Info」構造タイプの構造タイプ要素を作成します。

7.2 構造タグの作成

4. 「Name」を名前として入力して、データタイプ[テキストタグ 16 ビット文字セット]を選択 します。

| 💂 Tag Management - WinC | C Configuration Studio | · — | | |
|--|---------------------------|---|------------------|---|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | | | |
| Tag Management « | 🎕 Structure typ | e elements [Va Find | ک ج م | « |
| 📮 🔜 Tag Management | Name | Externa Data type Leng | th Forma 🔨 🔒 | |
| 🗄 🍄 Internal tags | 1 Name | Text tag 16-bit character set | | |
| 🖨 🌆 Structure tags | 2 💥 | Binary Tag | | |
| 🗄 🕂 😽 Valve_Info | 3 | Signed 8-bit value | | |
| 🧼 🥥 Name | 4 | Signed 16-bit value | | 8 |
| 125 | 5 | Unsigned 16-bit value | | |
| | 6 | Signed 32-bit value | | |
| | 7 | Unsigned 32-bit value | | |
| **** | 8 | Floating-point number 32-bit IEEE 754 | | |
| Computer | 9 | Text tag 8-bit character set | | |
| | 10 | Text tag 16-bit character set | | |
| <mark>7.</mark> († 7. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. | 11 II II II II Structu | Ire type Text reference Date/Time | > | |
| Ready NUM | English (United State | es) Table: 1 Structure type element 100 % | | |

5. ショートカットメニューにより次のプロパティの追加の構造タイプ要素を作成します。

| 名前 | データタイプ |
|-------|------------------------|
| ステータス | 2 進タグ |
| 温度 | 浮動小数点数 64 ビット IEEE 754 |
| 圧力 | 浮動小数点数 64 ビット IEEE 754 |
| 流量 | 浮動小数点数 64 ビット IEEE 754 |

あるいは、次に利用可能なフィールドをクリックして、構造タイプ要素の名前を直接入力します。

シングルアクションで複数の構造タイプを作成するには、「Name」という名前のフィールド 内をクリックして、フィールドの右下の隅をマウスで下にドラッグします。Microsoft Excel のオートフィル機能と同様に、行の列エントリは下の行に適用されます。

結果

「Valve_Info」構造タイプを作成し、5つの構造タイプ要素を定義しています。

「Valve_Info」構造タイプはタグ管理の[構造タグ]の下に表示されます。
7.2 構造タグの作成

| 💻 Tag Management - WinCC Cont | figu | ratio | n Studio | | <u></u> | | × | (|
|--|------|--------|-----------------|------------|-----------------------------------|-------|----------|----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | |
| Tag Management | « | 4 | Structure type | e eleme | Find | Q | - | ~ |
| 🗄 🚭 Structure tags | ^ | | Name | External | Data type | Lengt | ^ T | 0 |
| 🗄 🔧 Valve_Info | | 1 | Name | | Text tag 16-bit character set | 256 | 3 | 5 |
| 🛶 🥥 Name | | 2 | Status | | Binary Tag | 1 | | |
| 🛶 🥥 Status | | 3 | Temperature | | Floating-point number 64-bit IEEE | 8 | | 5 |
| 🖉 🥥 Temperature | | 4 | Pressure | | Floating-point number 64-bit IEEE | 8 | | ' |
| Pressure | | 5 | Flow | | Floating-point number 64-bit IEEE | 8 | | |
| Flow | | 6 | * | | | | | |
| • | ~ | 7 | | | | | | |
| | _ | 8 | | | | | v | |
| 🔲 🚰 🗘 🛃 🗊 🕿 🛃 📣 | * | 14 4 | Structur | e type e | elements Structure tags Str | uctur | | |
| Ready NUM (| Gern | nan ((| Sermany) Table: | 5 Structur | e type elements 100 % 😑 | 0 | | .; |

次のステップ

定義されている構造タイプに基づいて構造タグを作成します。

7.2.2 構造タグの作成

概要

次のステップは、前に作成した構造タイプとデータタイプを使用する構造タグを作成する 方法を示しています。

必要条件

• 「Valve_Info」構造タイプが作成され、定義されている。

7.2 構造タグの作成

手順

1. [構造タグ]タブで、「Valve_Info」構造タイプのために構造タグ「Valve1」を作成します。



- 利用可能なフィールドをクリックします。
 これまでに1つの構造データタイプしか作成されていないため、「Valve_Info」が[データタイプ]列に自動的に入力されます。
- 3.「Valve2」という名前の別の構造タグを作成します。

結果

「Valve_Info」構造タイプに基づいて2つの構造タグを作成しています。

| 💻 Tag Management - WinCC Config | jurati | on Studio | | | | _23 | | × |
|--|--------|-------------------|----------------|----------------|----------|--------------|------|-----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | |
| Tag Management « | | Structure ta | igs [Valve. | Find | | | ٩ | • « |
| 🖃 🛼 Tag Management 🛛 🔺 | | Name | Comment | Data type | Length | Format adapt | atio | ^ τ |
| 🗄 🥰 Internal tags | 1 | Valve1 | | Valve_Info | 0 | | | 2 |
| 🖃 🚟 Structure tags | 2 | Valve2 | | Valve_Info | 0 | | | ĕ |
| 🗄 🕂 🔧 Valve_Info | 3 | 洸 | | | | | | Ē |
| 🔷 🖉 Name | 4 | | | | | | | 0 |
| 🛛 🥥 Status | 5 | | | | | | | |
| 🖉 🥥 Temperature | 6 | | | | | | | |
| Pressure 🗸 | 7 | | | | | | | |
| | 8 | | | | | | | ~ |
| 🛄 🚰 🔁 🖪 📲 🔹 🔹 | 14 | 🕩 🕅 🛛 Struc | ture type eler | ments Str | ucture t | ags 🗸 < | > | |
| Ready NUM | En | glish (United Sta | tes) Table: 2 | Structure tags | 100 % | | | -+ |

タグのインスタンスとして、関連の構造タグ要素が自動的に作成されます。

7.2 構造タグの作成

| 💻 Tag Management - WinCC (| Conf | iguration Studio | | | | 3 <u>840</u> | | | × |
|--|----------|---|-----------------------------------|------|------------------------|--------------|-----|--------|-----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> | elp | | | | | | | | |
| Tag Management « | | Structure tag eler | ments [Valve_Info] | | Find | | Q | • | * |
| 🖃 🖬 Tag Management | | Name | Data type | Leng | th Format adaptation | Connecti | on | ^ | σ |
| 🞰 🍄 Internal tags | 1 | Valve1.Flow | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | 5 |
| 🖃 🚟 Structure tags | 2 | Valve1.Name | Text tag 16-bit character set | 256 | | Internal t | ags | | ĕ |
| 🗄 🕂 Valve_Info | 3 | Valve1.Pressure | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | te |
| 🖉 Name | 4 | Valve1.Status | Binary Tag | 1 | | Internal t | ags | | N I |
| 🖉 Status | 5 | Valve1.Temperature | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | |
| 🧳 Temperature | 6 | Valve2.Flow | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | |
| | 7 | Valve2.Name | Text tag 16-bit character set | 256 | | Internal t | ags | | |
| A Flow | 8 | Valve2.Pressure | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | |
| 11000 | 9 | Valve2.Status | Binary Tag | 1 | | Internal t | ags | | |
| | 10 | Valve2.Temperature | Floating-point number 64-bit IEEE | 8 | | Internal t | ags | | |
| 8 - <u>-</u> | 11 4 | X X X X X X X X X X X X X X X X X X X | pe elements 🖌 Structure tags 🗎 | Stru | cture tag elements | | j | ¥ : | |
| Ready NUM | | | English (United States) | Ta | ble: 10 Tags 100 % (| Э | Q | _ | ÷., |

7.3 ポップアップ画面の設定

7.3 ポップアップ画面の設定

7.3.1 ポップアップ画面の作成とそのサイズの適合

概要

次のステップは、フェイスプレートインスタンスをクリックすると開くポップアップ画面の 作成方法を示しています。

ポップアップ画面で下のコンテンツが非表示になるのを防止するために、プロセス画像の サイズを適合します。

必要条件

プロセス画像「POPUP.Pdl」が作成されていること。
 プロセス画面の作成 (ページ 51)で説明されている手順に従ってください。

手順

- 1. [グラフィックデザイナ]エディタで、「POPUP.Pdl」プロセス画像を開きます。
- 2. [ジオメトリ]グループを[オブジェクトプロパティ]ウィンドウで開き、[画像幅]を値「200」 に設定します。

| Object Properties | | | | × |
|---|--|--------------|--------|-------|
| Properties Events Texts Animation | | | | |
| Picture Object Geometry Colors Styles Miscellaneo Background Pic Effects Attribute Picture Width On Grid Width Grid Height | Static 200 1024 Yes 10 10 | Dynamic D | Update | Indir |
| | | | | > |
| 🔲 Object Properties 🛛 Tags 📄 Output Window 🕒 Library 🕒 SVG librar | y Dynamic V | Wizard | | |

3. ステップ2と同様に、[画像高さ]を値「200」に設定します。

結果

プロセス画像の寸法を 200 x 200 ピクセルに変更しています。

| 👍 Graphics Designer - POPUP.Pdl | | <u>100</u> | | × |
|--|-----------------------------|-------------------|--------------|--------------|
| File Edit View Arrange Tools Window DataConnector Help | . | 0.0 | | |
| | <u>a</u> at t | Q LQ 100 | % <u>*</u> : | 1111 100 |
| | | | | |
| POPURPdi X | | | | - |
| | | | | |
| Object Properties | | | | ▼ ₽ × |
| Properties Events Texts Animation | | | | |
| Picture Object Attribute Geometry Picture Width Colors Picture Height Styles Grid On | Static 200 200 Yes | Dynamic Q Q | Update | |
| Object Properties Image Output Window Image Image SVG I 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 15 | ibrary O Lawar | Wizard | | |
| Drücken Sie F1, um Hilfe zu erhalten. | Englisch (US | A) | | Rur |

次のステップ

「POPUP.Pdl」プロセス画像にすべての必要なオブジェクトを配置します。

7.3.2 ポップアップ画面にオブジェクトに配置

概要

次のステップは、必要なオブジェクトをポップアップ画面に挿入する方法を示しています。

7.3 ポップアップ画面の設定

スタティックテキストをラベルとして使用し、プロセス値を出力するために I/O フィール ドを使用します。

必要条件

•「POPUP.Pdl」プロセス画像が開いていること。

手順

- 1. [スタティックテキスト]タイプの5つのオブジェクトを挿入します。 プロセス画面の編集(ページ55)に説明されている手順に従ってください。
- 2. 次のテキストを入力します。
 - バルブの情報
 - 名前
 - 温度[℃]
 - 圧力[Pa]
 - 流量[l/s]

| POPUP.Pdi x | |
|-------------------|--|
| Valve Information | |
| Name | |
| Temperature [°C] | |
| Pressure [Pa] | |
| Flow [I/s] | |

3. テキストボックスでフォントサイズを定義します。

7.3 ポップアップ画面の設定

4. テキストボックスのサイズと位置を適合します。

| _ |
|---|
| |
| |
| |
| |
| |
| |
| |
| |

5. [I/O フィールド]タイプの3つのオブジェクトを挿入し、各ケースのフィールドタイプとして [出力]を選択します。[タグ]フィールドを空にしておきます。

| POPUP.PdI × | | | • |
|------------------------|-------------------------|--------|-------------------------|
| Valve Information | I/O Field Configuration | ? × | and the second |
| Name | Tag: | | |
| Temperature [°C] 0,000 | | | المحصور فللسوي المتحمين |
| Pressure (Pa) 0,000 | O Input | | |
| Flow [l/s] 0,000 | • Output I/O Field | | |
| | Font Size 12 | | |
| | Font | | |
| | Color | | |
| | ОК | 2 2 | |

7.3 ポップアップ画面の設定

結果

ポップアップ画面にすべての必要なオブジェクトを挿入しています。

| POPUP.Pdi x |
|------------------------|
| Valve Information |
| Name |
| Temperature [°C] 0.000 |
| Pressure (Pa) 0.000 |
| Flow [l/s] 0.000 |

次のステップ

挿入されているオブジェクトをダイナミック化します。

7.3.3 ポップアップ画面でオブジェクトをダイナミック化

概要

次のステップは、ポップアップ画面でオブジェクトをダイナミック化する方法を示してい ます。

レベルインジケーターをダイナミック化するのに使用したのと同じ手順に従っていると、同 じタグがすべてのインスタンスで出力されます。この場合、バルブのすべてのポップアップ 画面は同じコンテンツになります。

別のバルブにたいして異なる値を表示するには、特定のタグではなく構造タイプ要素をリ ンクします。この場合、表記「.Temperature」をタグ名として使用します。

オブジェクトは2つのステップでダイナミック化します。

- スタティックテキスト「Name」のオブジェクトプロパティ[テキスト]のダイナミック 化: それぞれのバルブの名前のダイナミック表示
- I/O フィールドのオブジェクトプロパティ[出力値]のダイナミック化:
 そバルブのそれぞれのプロセス値のダイナミック表示

7.3 ポップアップ画面の設定

必要条件

•「POPUP.Pdl」プロセス画像が開いていること。

7.3 ポップアップ画面の設定

手順

1. 「Name」ラベルのある[スタティックテキスト]オブジェクトの場合、オブジェクトプロパ ティ[テキスト]を構造タグ要素「.Name」でダイナミック化します。



ライトの電球が緑色になります。テキストがダイナミック化されています。

2. 「Temperature [℃]」ラベルの横にある[I/I/O フィールド]オブジェクトの場合、オブジェクトプロパティ[出力]値を構造タグ要素「.Temperature」でダイナミック化します。

7.3 ポップアップ画面の設定

| POPUP.PdI × | | | | - | Standard • | , † X |
|---|--|---|---|--------------|--------------------|---|
| Valve Infor Name Temperature (*C) Pressure (Pa) Flow [l/s] Object Properties | exts Animation | | | → # X | Selectio | n A gon line se le se Si Segn se A ular rang ndec c Te |
| | Attribute Field Type Input Value Output Value Data Format Output Format Vy on Full | Static Output 0 0,000000e + 00 Decimal 999,999 No | Dynamic .Temperaturi .Temperaturi | Update ^ | Con Smart C | nect Ibjec licat ure \ trol obje Field obje |

ライトの電球が緑色になります。出力値がダイナミック化されています。

- 3. 他の2つの I/O フィールドに対してステップ2を繰り返します。 ダイナミック化のために次の構造タグ要素を使用します。
 - ".Pressure"
 - ".Flow"

結果

ポップアップ画面のすべてのオブジェクトをダイナミック化しています。

7.4 フェイスプレートタイプの設定

7.4 フェイスプレートタイプの設定

7.4.1 フェイスプレートタイプの作成とオブジェクトの配置

概要

次のステップは、フェイスプレートを作成し、必要なオブジェクトを配置する方法を示しています。

必要条件

[グラフィックデザイナ]エディタが開いている。

手順

- 1. [ファイル]メニューの[新規フェイスプレートタイプ]エントリを選択します。 *.FPT 形式で新しい画像が開きます。
- 2. [ファイル]メニューの[保存]エントリを選択します。
- フェイスプレートタイプに「Valve.Fpt」という名前を付けます。
 [保存]を押して確定します。
 「Valve.Fpt」フェイスプレートタイプが「GraCS」プロジェクトフォルダに作成されています。
- グローバル SVG ライブラリ"IndustryGraphicLibraryV2.0"を[SVG ライブラリ]ウィンドウで 選択します。
- 5. [HandValve2]オブジェクトを[Valves]フォルダから挿入します。

- Standard Valve.FPT X -▼ ₽ X ~ Selection ~ 🚋 😽 Standard Objects 2/ Line A Polygon A Polyline Ellipse Circle ?????? 🕒 Ellipse Segment - Pie Segment C Ellipse Arc 💪 Circular Arc 3) > Rectangle X SVG library - Rounded Rectangle 2 A Static Text 📸 🗼 🖪 🗙 🚬 🔚 🔠 🐻 3 Connector hart Objects — 🚞 Signal_Lamps ^ ^ Application Windov ----- Status-Overlays Picture Window ControlValve HandValve OCX Control ----- Tanks OLE OLE object 👆 Valves J 0.12 I/O Field Bar - Water ter 🖳 Graphic Obiect - Wires Ca HandValve1 HandValve2 < 📳 Object Pro... 🧃 Tags 📃 Output Wi... 🎦 Library 🕒 SVG library 😤 Dynam 🔊 St... 🔳 C... 💥 St... 🎍 Pr..
- 6. [スタティックテキスト]タイプのオブジェクトを挿入し、テキスト「Valve X」をラベルとして入力します。

7. オブジェクトの位置を適合します。



- 8. 選択されているオブジェクトがない状態になるように、背景をクリックします。プロセス画像のオブジェクトプロパティが表示されます。
- 9. [画像幅]を値「160」に、[画像高さ]を値「190」に設定します。



結果

[バルブ]フェイスプレートタイプを作成し、すべての必要なオブジェクトを配置しています。

次のステップ

「Valve_Info」構造タイプを[バルブ]フェイスプレートタイプとリンクします。

7.4.2 構造タイプとフェイスプレートタイプのリンク

概要

次のステップは、フェイスプレートタイプの設定方法を示しています。 フェイスプレートタイプを作成してある構造タグとリンクし、ポップアップ画面を構成し ます。

必要条件

•「Valve.Fpt」フェイスプレートタイプが開かれていること。

7.4 フェイスプレートタイプの設定

手順

1. フェイスプレートタイプのプロパティを定義するには、[設定]ダイアログを開きます。

| File | Edit | View | Arrange | Tools | Window | DataConnector Help |
|---------|------|--------------------|--------------------------------|-------------------------|--------|--------------------|
| : D 6 | 5 | | | | Ctrl+Z | |
| : Valve | 3 | |) | | Ctrl+Y | |
| Val | X | Cut | | | Ctrl+X | - |
| | | Сору | | | Ctrl+C | |
| 1.1.1 | | Duplicat | e | | | |
| | 8 | Paste | | | Ctrl+V | |
| | | Delete | | | Del | |
| | | Select Al | I. | | Ctrl+A | |
| | | Customi | zed Object | | | • |
| | | Group | | | | • |
| • • • | | TAB Sequ | lence | | | • |
| | | Linking | | | | • |
| | | Propertie | es | | | • |
| | | Edit face | plate tags | | | |
| | | Configu | re faceplate | e type | | |
| | | Configu Configu | re text list f re graphic l | or facep list for fa | | |
| | | | | | | |

2. [タグ]タブで、「Valve_Info」構造タイプを選択されているタグとして設定します。

7.4 フェイスプレートタイプの設定



構造タイプがインターフェースタグとして定義されています。 フェイスプレートタイプのダイナミック化を行うために、インターフェースタグを使用し ます。

7.4 フェイスプレートタイプの設定

3. [テキスト]オブジェクトプロパティを「Valve Info.Name」タグとリンクします。



4. 設定を保存するには、[適用]をクリックして、[はい]で確定します。

結果

作成した構造タイプをインターフェースタグとして定義し、タグを割り付けることにより テキストをダイナミック化しています。

次のステップ

「POPUP.Pdl」プロセス画像をポップアップ画面として設定します。

7.4.3 ポップアップ画面の設定

概要

次のステップは、フェイスプレートタイプでポップアップ画面を設定する方法を示してい ます。

必要条件

• 「Valve.Fpt」フェイスプレートタイプが開かれていること。

手順

- 1. [編集]メニューの[フェイスプレートタイプの設定]エントリを選択することにより、設定ダ イアログを開きます。
- 2. [ポップアップ画面]タブで「POPUP.Pdl」プロセス画像をポップアップ画面として設定します。

| Properties | Events | Tags | Popup screen | | | | |
|-----------------------|------------------------------|---------|-------------------------|------|--------------|---------------------------|-------|
| Selected tags: | | | Picture servers: | 2 | Pictures: | | |
| E⊷ Tags Valvi « | e_Info KLOCALHOST>::PO | PUP.PdI | <localhost></localhost> | | POPUP.Pdl | B | |
| Popup windo | w parameters: | | Relative offset | X: 0 | Expanded set | tings: nultiple popups | |
| Show w | indow title Close" button | | Relative offset | Y: 0 | Event: | Mouse click | ~ |
| | | | | | | | Apply |

3. 設定を保存するには、[適用]をクリックして、[はい]で確定します。 画像がポップアップ画面として設定されています。

結果

「POPUP.Pdl」画像をポップアップ画面として設定しています。フェイスプレートインスタンスをクリックすると、ランタイムで画像が開きます。

関連するスクリプトが自動的に作成されます。

7.5 フェイスプレートインスタンスの作成

7.5 フェイスプレートインスタンスの作成

概要

次のステップは、フェイスプレートインスタンスをプロセス画像に挿入する方法を示して います。

このために、ドラッグアンドドロップで構造タグをプロセス画像にドラッグします。

必要条件

- [グラフィックデザイナ]エディタが開いている。
- •「START.Pdl」プロセス画像が開いている。

手順

1. プロセス画像の2つのSVGオブジェクト「HandValve2」のそれぞれを右クリックして、[削除]を選択します。

2つの SVG オブジェクトが削除されます。



 すべての構造インスタンスのリストを[タグ]選択ウィンドウに表示します。
 [タグ]ウィンドウが表示されない場合、[ツールバー]の[ビュー]メニューの[タグ]エントリを 有効にします。 7.5 フェイスプレートインスタンスの作成

3. ドラッグアンドドロップで「Valve1」構造インスタンスをプロセス画像にドラッグします。 「Valve.Fpt」フェイスプレートタイプを選択します。



フェイスプレートインスタンスがプロセス画像に配置されていて、「Valve1」構造インスタンスの構造タグとダイナミック化されています。

7.5 フェイスプレートインスタンスの作成



7.5 フェイスプレートインスタンスの作成

- START.Pdl X
- 4. フェイスプレートインスタンスの位置とサイズを適合します。

5.「Valve2」構造インスタンスに対してステップ 3~4 を繰り返します。



結果

構造インスタンスを挿入し、その結果としてプロセス画像で2つのフェイスプレートイン スタンスを作成しています。

7.6 フェイスプレートタイプの変更

概要

このセクションは、フェイスプレートタイプを使用することにより、設定の労力を軽減した状態で変更を実装する方法について、情報を提供します。

原理

フェイスプレートタイプを使用する主なメリットは、容易で時間を節減できる形で変更を 実装できることです。

フェイスプレートタイプを変更すると、変更はすべてのインスタンスに適用されます。こ れにより、個別オブジェクトの変更を時間をかけないで行うことができます。

「Quick Start」 プロジェクトの設定

さらに、「Valve.Fpt」フェイスプレートタイプは、カラーを使用することにより、バルブ のステータスを示します。

タグを使用して、バルブの基本カラーをアニメーション化します。

フェイスプレートタイプでアニメーションを設定するため、変更はプロセス画像の両方のバルブに適用されます。

7.6.1 オブジェクトプロパティのアニメーション化

概要

次のステップは、オブジェクトプロパティのアニメーション化の方法を示しています。 希望のプロパティをタグとリンクします。オブジェクトプロパティが、タグ値の機能とし てランタイムで変更されます。

必要条件

•「Valve.Fpt」フェイスプレートタイプが開かれていること。

7.6 フェイスプレートタイプの変更

手順

- フェイスプレートタイプの SVG ライブラリのバルブオブジェクトをクリックします。 SVG オブジェクトが選択され、[オブジェクトプロパティ]選択ウィンドウで表示されます。
- 2. パブリッシュされているインターフェースタグを使用する新しいアニメーションを作成し ます。

| Object Properties | × |
|---|---|
| Properties Events Texts Animation | |
| SVG Object | ^ |
| Add New Animation | |
| Animate Property I O Tag | |
| Published Tag 2 s Y Y Y | |
| | |
| | ~ |
| | |
| 🔝 Object Properties 🛥 Tags 🔄 Output Window 🕒 Library 🎦 SVG library 🤻 Dynamic Wizard | |

- 3.「…」を使用して、タグの選択を開きます。
- インターフェースタグ「Valve_Info.Status」を選択して、[OK]で確定します。 アニメーションは「Valve_Info」構造タイプの構造タイプ要素「.Status」でコントロールされます。

| 📧 Published tag list | × |
|--|---|
| Published tags: | |
| Tags Valve_Info Valve_Info.Flow Valve_Info.Name Valve_Info.Status Valve_Info.Status Valve_Info.Tempe | |
| OK Cancel | |

7.6 フェイスプレートタイプの変更

| Properties Events Texts Animation | |
|---|----------------------------------|
| SVG Object - Add New Animation - Animate Property 0 | Settings Data Type |
| | Bool 0 🗘 |
| < | Bool Direct Range Range |

5. [ブーリアン]データタイプを選択します。

6. [BasicColor]プロパティを追加します。

| Valve X | Authorization Background Color Background Flash Frequency Border Background Flash Frequency Border Colo Border Flas Border Style Border Style Border Style Border Weight ContrastColor Display Dynamic Filling Fill direction Fill Level Eill Pattern | | |
|---|---|------------------------------|-------|
| Object Properties | Fill Pattern Color Fillshing Background Active | | ▼ ‡ X |
| Properties Events Texts Anir SVG Object - Add New Animation - Animate Property 0 | nati Flashing Background Color Off Flashing Border Active Flashing Border Color Off Flashing Border Color Off Flashing Border Color On Global Shadow Graphic Height Object Transparency Operator-Control Enable Picture Transparent Color Picture Transparent Color On Position X | Valve_Info.Status 2 s | > |
| Object Properties 🛛 Tags | 📃 Output Window 💾 Library 💾 | SVG library 🤻 Dynamic Wizard | |

7.6 フェイスプレートタイプの変更

7. 緑色を[真]、赤色を[偽]に対して選択します。

| SVG Object | Add Propert | У | ^ |
|----------------------|-------------|------------|---|
| - Animate Property 0 | Area | BasicColor | |
| | True | | |
| | False | | |
| | < | | > |

8. プロセス画像を保存します。

結果

「Valve_Info.Status」インターフェースタグで SVG オブジェクトをアニメーション化して います。バルブのステータスに応じて、バルブは緑色あるいは赤色でランタイムで示され ます。

| Valve.FPT × | | | - |
|--|-------------------------|---------|--------------|
| Object Properties | | | → ‡ X |
| Properties Events Texts Ani | mation | | |
| Object Properties 1 Properties Events Texts Animation SVG Object - Add New Animation - Add New Animate Property 0 - Expression - Expression - Add Property - Area Basic Color | | | |
| | Area Basi True Palse | icColor | × |
| Object Properties Properties Events Texts Animation SVG Object Process Otata Type Data Type Bool Bool Bool Quality Code Image: Color True False Image: Color Image: Color < | | | |

7.7 プロジェクトのテスト(グラフィックシステム-フェイスプレートタイプ)

7.7 プロジェクトのテスト(グラフィックシステム-フェイスプレートタイ プ)

概要

次のステップは、WinCC タグシミュレータによって拡張している「Quick_Start」プロジェクトをテストする方法を示しています。

拡張している「Quick_Start」プロジェクトで、WinCC タグシミュレータによって構造タ グの値をシミュレーションします。

WinCC タグシミュレータは、構造タグエレメントに対して様々な値を割り付けます。構造 タグ要素がポップアップ画面の I/O フィールドおよびバルブの表示にリンクされています。 タグ値に従ってランタイムの I/O フィールドおよびバルブ表示の値は異なります。

必要条件

ランタイムプロパティが定義されていること。
 ランタイムタイムプロパティの定義(グラフィックシステム-プロセス画像)(ページ87)に
 説明されている手順に従ってください。

7.7 プロジェクトのテスト(グラフィックシステム-フェイスプレートタイプ)

手順

1. 「QuickStart_Simulation.sim」設定を[タグシミュレーション]エディタで開きます。



2. タグ選択ダイアログを開き、「Valve1」と「Valve2」のタグを構造インスタンスのリストから 選択します。

| gs [Simulation] | | Find | ب م |
|------------------------------------|---------------|---------------|----------------|
| Tag name | Data type | Function | Cycle 🔨 |
| 1 Tank Level Unsigned 16-bit value | | Increment | 1 |
| | | | |
| | | | ~ |
| lag | | | ^ |
| ter: | | ~ | ► = III |
| 💣 WinCC Tags | Name Type | Parameter Cor | nment L |
| 🖶 📙 Internal tags | * 7* | 7* 7* | 7 * |
| 🔃 📙 List of all structure in | stanc | Info | 0 |
| ist of all tags | Valve1 Valve_ | Info | 0 |
| | | | |
| | | \ U | <u> </u> |

7.7 プロジェクトのテスト(グラフィックシステム - フェイスプレートタイプ)

- 3. 」ボタンを使用して、構造タグ要素を拡張します。
- 4. [アクティブ]列ですべてのタグを有効にします。

| Гаg | s [Simulation] | | | (🔊 | | Q | • |
|-----|--------------------|------------|------|----------|-------------------|------------|---|
| Т | | Function | Cycl | Activ | 2 alu Quality set | Current va | ^ |
| L | 1 vel | Increment | 1 | V | 0x80 - good - ok | | |
| 2 🔺 | Valve2 | | h 🛫 | | | | |
| 3 | Valve2.Name | User input | - | V | 0x80 - good - ok | | |
| 4 | Valve2.Status | Random | 1 | V | 0x80 - good - ok | | |
| 5 | Valve2.Temperature | Random | 1 | V | 0x80 - good - ok | | |
| 6 | Valve2.Pressure | Random | 1 | V | 0x80 - good - ok | | |
| 7 | Valve2.Flow | Random | 1 | V | 0x80 - good - ok | | |
| 8 4 | Valve1 | | | | | | |
| 9 | Valve1.Name | User input | 1 | V | 0x80 - good - ok | | |
| 10 | Valve1.Status | Random | 1 | V | 0x80 - good - ok | | |
| 11 | Valve1.Temperature | Random | 1 | V | 0x80 - good - ok | | |
| 12 | Valve1.Pressure | Random | 1 | V | 0x80 - good - ok | | |
| 13 | Valve1.Flow | Random | 1 | V | 0x80 - good - ok | | |

5. タグの次のプロパティを定義します。

| タグ名 | ファンクション | サイクル | 品質セット |
|-------------------|---------|------|----------------|
| Tank_Level | インクリメント | 1 | 0x80 - 良好 - OK |
| Valve2.Name | ユーザー入力 | 1 | 0x80 - 良好 - OK |
| Valve2.Status | ランダム | 1 | 0x80 - 良好 - OK |
| Valve2.Temperatur | ランダム | 1 | 0x80 - 良好 - OK |
| е | | | |
| Valve2.Pressure | ランダム | 1 | 0x80 - 良好 - OK |
| Valve2.Flow | ランダム | 1 | 0x80 - 良好 - OK |
| Valve1.Name | ユーザー入力 | 1 | 0x80 - 良好 - OK |
| Valve1.Status | ランダム | 1 | 0x80 - 良好 - OK |
| Valve1.Temperatur | ランダム | 1 | 0x80 - 良好 - OK |
| е | | | |
| Valve1.Pressure | ランダム | 1 | 0x80 - 良好 - OK |
| Valve1.Flow | ランダム | 1 | 0x80 - 良好 - OK |

6. 設定を保存します。

7. >ボタンを使用して、WinCC エクスプローラで WinCC Runtime を有効にします。

8. [タグシミュレーション]エディタでシミュレーションを起動します。

7.7 プロジェクトのテスト(グラフィックシステム-フェイスプレートタイプ)

9. バルブの名前を表示するには、名前を構造タグ要素「Valve1.Name」と「Valve2.Name」の ための現在のタグ値として入力します。

| Ta | gs [Simulation active | e] | | Find | | P | • |
|----|------------------------|------------|-------------|------------------|------------------|------|---|
| | Tag name | Function | Cycl Active | Con Value set | Quality set | Cur | ^ |
| 1 | Tank_Level | Increment | 1 | | 0x80 - good - ok | | |
| 2 | Valve2 | | | | | | |
| 3 | Valve2.Name | User input | 1 🔽 | Valve 2 | 0x80 - good - ok | Valv | |
| 4 | Valve2.Status | Random | 1 | | 0x80 - good - ok | 0 | |
| 5 | Valve2.Temperature | Random | 1 | 97,5920896023438 | 0x80 - good - ok | 191 | |
| 6 | Valve2.Pressure | Random | 1 🔽 | 152,433851130711 | 0x80 - good - ok | 152 | |
| 7 | Valve2.Flow | Random | 1 🔽 | 137,748344370861 | 0x80 - good - ok | 118 | |
| 8 | Valve1 | | | | | | |
| 9 | Valve1.Name | User input | 1 🔽 | Valve 1 | 0x80 - good - ok | Val | |
| 10 | Valve1.Status | Random | 1 2 | | 0x80 - good - ok | 0 | |
| 11 | Valve1.Temperature | Random | 1 | 132,071901608325 | 0x80 - good - ok | 38,2 | |
| 12 | Valve1.Pressure | Random | 1 🔽 | 82,7417828913236 | 0x80 - good - ok | 82,2 | |
| 13 | Valve1.Flow | Random | 1 🔽 | 2,36823633533738 | 0x80 - good - ok | 116 | |

10.[タグシミュレーション]エディタとランタイムウィンドウを、両方を同時に監視できるように位置調整します。

2つのバルブのうち1つでクリックすることによりポップアップ画面を開き、様々なシミュレーション値がディスプレイに影響を与える方法を監視します。

「Valve(x).Status」タグの値の機能としてバルブの色が変更する方法を監視します。



11.シミュレーションおよび WinCC Runtime を終了します。

7.7 プロジェクトのテスト(グラフィックシステム - フェイスプレートタイプ)

結果

WinCC タグシミュレータを使用して「Quick_Start」プロジェクトをテストし、拡張しています。

テストは、プロセス値を提供された時のプロジェクトの動作を示します。

値のアーカイブと表示

8.1 値のアーカイブと表示

概要

このセクションは、アーカイブシステムに関する情報を提供し、プロセス値アーカイブでの 値の保存方法を説明します。

原理

プロセス値アーカイブを使用して、プロセス値の経時的な展開を、例えばダイアグラムや テーブルなどのように表示します。実際には、早期に問題を認識できる可能性があるので、 こうした暫定的な表示は非常に重要です。

プロセス値アーカイブの別の活用方法として、個別の履歴的なプロセス値へのアクセスが あります。たとえば、生産上の問題が発生した場合に、特定の値がどの程度高かったのかを 特定するために役立ちます。

「Quick_Start」プロジェクトの設定

"Quick Start"プロジェクトでは、プロセス値は必要ありません。

このプロジェクトでは、内部タグ「Tank_Level」を使用します。WinCC タグシミュレータ によって、これらのタグの値をシミュレートします。シミュレートしたタグの値は、プロ セス値アーカイブに保存されます。

保存した値のシーケンスは、トレンドダイアグラムやテーブルとして、プロセス画像に入力 されます。このために、[グラフィックデザイナ]エディタの[コントロール]選択画面にあ るコントロールを使用します。

ランタイムで設定済みコントロールのシミュレーション値の変更を監視します。

8.2 アーカイブシステム

8.2 アーカイブシステム

概要

アーカイブシステムは、WinCC のサブシステムです。このサブシステムは、プロセス値と メッセージをアーカイブするために使用します。

アーカイブシステムのコンポーネント

プロセス値のアーカイブシステムは、設定コンポーネントとランタイムコンポーネントで 構成されています。

- アーカイブシステムの設定コンポーネントは"タグロギング"エディタです。
 このエディタで、(その中の)以下のタスクを実行できます。
 - 取得サイクルとアーカイブサイクルの定義
 - プロセス値アーカイブと圧縮アーカイブの設定
 - アーカイブするプロセス値の定義
- タグロギングランタイムは、アーカイブシステムのランタイムコンポーネントです。
 タグロギングランタイムには、次のようなタスクがあります。
 - プロセス値のプロセス値アーカイブへの書き込み
 - プロセス値アーカイブからの、アーカイブ済みプロセス値の読み取り

アーカイブ

プロセス値は、コンピュータのハードディスクのアーカイブデータベース、またはタグロ ギングランタイムのメインメモリに、保存されます。

アーカイブされたプロセス値を圧縮し、データ量を減らすことができます。

アーカイブ期間

アーカイビングのサイクルとイベントによりアーカイビング時間を制御します。

ー定間隔、または定義されている量やパーセンテージでプロセス値が変更されたときのみ などに、プロセス値をアーカイブできます。

ライセンシング

WinCC 基本システムで、追加のライセンスなしで 512 のロギングタグを設定することが できます。
WinCC システムのアーカイブ



8.3 タグロギングの開始

8.3 タグロギングの開始

概要

次のステップは、"タグロギング"エディタの開始方法を示しています。

"タグロギング"エディタで、取得サイクルやアーカイブサイクルの時間と共に、プロセス値 アーカイブを設定します。

必要条件

• "Quick_Start"プロジェクトが開いている。

手順

"タグロギング"エディタを開始します。
 "タグロギング"エディタが開きます。

結果

"タグロギング"エディタが開いています。

| 🛄 ፵ክጊቶንንሶ - WinCC Configuration Studio — 🗆 🖂 | | | | | | | | |
|--|-----------------------------------|------|---------------|--|--|--|--|--|
| ファイル(E) 編集(E) ビュー(V) ツール(L) ヘルフ [*] (H) | | | | | | | | |
| ፟፟፟፟፟፟፟፟፟፟፟፟፟፟፟፟፟ | ァーカイブ・[プロセス値ァー 検索 | ۶ | > - « | | | | | |
| □ ■ タウロキシウ* | <mark>アーカイブ名</mark> コメント アーカイブ 無効 | 手動入力 | 18 ~ 🔍 | | | | | |
| | 1 💥 | | 1 2 | | | | | |
| ≝ ⊷ ₩ アーカイフ* | 2 | | শ | | | | | |
| | 3 | | | | | | | |
| 正縮アーカイフ* | 4 | | | | | | | |
| | 5 | | | | | | | |
| | 6 | | | | | | | |
| | 7 | | | | | | | |
| | 8 | | ~ | | | | | |
| U 🗹 📶 📑 🐌 🖑 🔶 🗶 🗌 | I ← → → J つロセス値アーカイフ* 圧縮アーカイフ* | < 2 | ×: | | | | | |
| 準備完了 NUM | 英語 (米国) 100 % 🤅 |) | | | | | | |

次のステップ

- [タグロギング]エディタで、取得サイクルとアーカイブサイクルの時間を設定します。
- 設定された時間で、タグ値を取得し、アーカイブする時間間隔を定義します。
- プロセス値アーカイブを作成します。

値のアーカイブと表示

8.3 タグロギングの開始

8.4 タイマーの設定

8.4 タイマーの設定

概要

"Quick_Start"プロジェクトで、取得サイクルおよびアーカイブサイクルの新しい時間を設定 します。

この時までに、タグ値を取得、アーカイブする時間間隔を決めておきます。

取得サイクル

取得サイクルは、プロセスタグのプロセス値を読み出す間隔を決定します。

取得サイクルは、WinCC Runtime が起動するとすぐに開始します。

アーカイブサイクル

アーカイブサイクルは、プロセス値がアーカイブデータベースに保存される時間間隔です。

- アーカイブサイクルは、常に設定された取得サイクルの整数倍です。
- アーカイブサイクルは、WinCC Runtimeの起動時またはユーザーが定義した時点に開始します。
 開始ポイントを指示すると、値の時間差アーカイブやアーカイブ負荷の分散が可能になります。
 システムは、取得とアーカイブの間に最大で取得サイクルの長さまでの遅延を引き起こすことがあります。

"タグロギング"エディタは、異なる標準時間を提供します。

これらの時間を変更することはできません。標準時間が十分でない場合は、新しい時間を 設定してください。

時間を設定する場合、基本時間と時間係数を定義します。基本時間と時間係数の積が、2 つのアーカイブの間の時間間隔を決定します。例えば、時間ベースを「1秒」に、時間係数 を「5」に設定すると、プロセス値は5秒毎にアーカイブされます。

必要条件

[タグロギング]エディタが開いている。

手順

- 1. 新しいサイクルタイム[Fill_Level]を作成します。
- 2. 新規タイマを作成するには、先頭の空白セルをクリックして、テーブルペインの[タイマ名] 列に名前を入力します。



結果

取得サイクルおよびアーカイブサイクルの新しい時間[Fill_Level]が設定されます。 設定した時間によって、タグ値は2秒毎に取得、アーカイブされます。

次のステップ

プロセス値アーカイブを作成し、内部タグの値をアーカイブできるようにします。

8.5 プロセス値アーカイブの作成

8.5 プロセス値アーカイブの作成

概要

次のステップは、"Quick_Start"プロジェクトのプロセス値アーカイブの作成方法を示して います。内部タグ"Tank Level"の値は、このアーカイブに保存されます。

[タグロギング]エディタでアーカイブを作成します。

プロセス値アーカイブを作成する場合、値をアーカイブするタグも定義します。

タグを定義すると、[タグロギング]はプロセス値アーカイブにアーカイブタグを作成しま す。アーカイブする値は、アーカイブタグに保存されます。

必要条件

• [Tank Level]タグがタグ管理で設定されている。

手順

- 1. [タグロギング]エディタのナビゲーションエリアで、[プロセス値アーカイブ]フォルダを選択 します。
- 2. テーブルエリアの[アーカイブ名]列で、上部の空白行をクリックして、名前「Tank_Level_Archive」を入力します。

| Tag Logging - WinCC Configuration S | itudio | | s | | × |
|--|--|--------------------|--------------------|----------|------------|
| <u>File Edit View Tools H</u> elp | | | | | |
| Tag Logging « | III Archives [Process | s Find | | ۰ م | • « |
| Tag Logging Timers Cycle times Time series Time series Time series Time series Compressed Archives Compressed Archives | Archive name 1 Tank_Level_Archive 2 2 3 2 4 5 1 Process Value | Comment | Archiving disabled | Mai | Properties |
| Ready NUM English | (United States) Table: 1 Pro | cess value archive | 100 % 🕞 🚽 | _ | - + .: |

 ナビゲーションエリアで、アーカイブのフォルダを選択します。 テーブルエリアで[タグ]タブを選択します。

8.5 プロセス値アーカイブの作成

4. テーブルで[プロセスタグ]列の先頭の空き行をクリックしてから、[...]ボタンをクリックします。

タグの選択用ダイアログが開きます。



5. 内部タグ[Tank_Level]を[QuickStart_Tags]タググループで選択します。

| Tag selection | | | | | | | × |
|--------------------------------------|------------|-----------------------|--------------|-----------|---|----------|-----------------|
| Filter: | | | | (| t | 6- 6- | \$=\$: ∰ |
| 🖃 💣 WinCC Tags | Name | Туре | | Parameter | | Cor | nment |
| 🖕 📙 Internal tags | * 7 | * | \mathbf{Y} | * | Y | * | |
| 🗄 📑 QuickStart_Tags | Tank Level | Unsigned 16-bit value | | | | | |
| 🖹 📑 Performance 🔪 📜 | | 2 | | | | | |
| 🕀 📑 ProcessHistoriar (🍋) | | | | | | | |
| 🗈 📑 TagLoggingRt 🛛 💙 | | | | | | | |
| 🗊 – 🦰 Script | | | | | | | |
| | | | | | | | |
| 🖅 📲 List of all tags | | | | | | | |
| 🗄 🕌 SIMATIC S7-1200, S7-1500 Channel | | | | | | | |
| | | | | (🍋) 🎽 | | | |
| | · | | | | | | |
| | | Apply | | ок | 0 | lance | * |

6. [OK]を押してダイアログを閉じます。 アーカイブタグが作成されました。内部タグの名前が使用されます。

結果

[Tank_Level_Archive]プロセス値アーカイブが作成されます。

内部タグ"Tank Level"の値は、このプロセス値アーカイブに記入されます。

8.5 プロセス値アーカイブの作成

| 🖶 Tag Logging - WinCC Configuration | Stuc | lio | | | | | | × |
|--|------|---------------|---------------|-------------------|-------------|----------------|----|-----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | |
| Tag Logging « | | Archives [Ta | nk_Lev | Find | | ، م | ·] | ~ |
| 🖃 📱 Tag Logging | | Process tag | Tag type | Tag name | Archive nam | e | | σ |
| Timers | 1 | Tank_Level | Analog | Tank_Level | Tank_Level_ | Arch | | 5 |
| Archives | 2 | 洸 | | | | | | ĕ |
| Process Value Archives | 3 | | | | | | | tie |
| Tank_Level_Archive | 4 | | | | | | | 0 |
| Compressed Archives | 5 | | | | | _ | | |
| | 6 | | | | | | | |
| | 7 | | | | | | | |
| | 8 | | | | | | | |
| Tags / Process-controlled ti | | | | | | | | |
| Ready NUM | | English (Unit | ted States) T | able: 1 Tag 100 | % 🖃 — (| J | (| ÷., |

次のステップ

作成したプロセス値アーカイブを編集します。

8.6 プロセス値アーカイブの編集

概要

次のステップは、プロセス値アーカイブ"Tank_Level_Archive"の編集方法を示しています。 プロセス値アーカイブの編集は、以下のステップで行なわれます。

- アーカイブタグの名前の変更
- 取得サイクルおよびアーカイブサイクルへの設定時間"Fill Level"の割り付け
- プロセス値アーカイブの保存場所の定義
- プロセス値アーカイブのサイズの定義

「Quick_Start」プロジェクトのメインメモリを、プロセス値アーカイブの保存場所として 選択します。メインメモリは、ランタイム中だけプロセス値アーカイブを提供します。

プロセス値アーカイブに保存されたデータセットの数によって、プロセス値アーカイブの サイズを定義できます。

必要条件

• プロセス値アーカイブ"Tank_Level_Archive"が作成されている。

手順

- 設定済みのアーカイブタグでラインにあるテーブルエリアをクリックし、 ベボタンを使用して[プロパティ]ウィンドウを開きます。
 必要な場合は、プロパティエリアを拡大します。
- 2. アーカイブタグ名として"Fill Level Archive"を入力します。

8.6 プロセス値アーカイブの編集

| File Edit View Tools Help Tag Logging Find Tag Logging Tag Logging Tag Logging Tag Logging Tag Logging Process tag Selection General General archive properties Archive name Tank_Level_Archives Tag tog Archives Tag tog Tank_Level_Archives General tag prope General tag prope Process tag Tank_Level_Archives General tag prope Process tag Process tag Tank_Level_Archive General tag prope Process tag Tag tope Archive name Tank_Level_Archive Second Tag tope Bereal B | 🔁 Tag Logging - WinCC Configuratio | n Studio | | | | - 0 × |
|---|--|--------------|----------|----|----------------------------|--------------------|
| Tag Logging I Tag Logging Tag Logging Tag Logging Tag Logging Tank_Level I Tank_Level 2 I Tank_Level I Tan | <u>File E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | |
| Tag Logging Timers Archives Tank_Level Tank_Level Tank_Level Tank_Level Tank_Level Tank_Level Tank_Level General archive properties Archive name Tank_Level_Archive General tag prope Tank_Level Tag type Tank_Level Tag supply System Alarm Logging Alarm Logging Alarm Logging Tag Management Tag Management<td>Tag Logging «</td><td>Find 🔎</td><td>•</td><td>11</td><td>Properties - Tag</td><td>;</td> | Tag Logging « | Find 🔎 | • | 11 | Properties - Tag | ; |
| Image: Timers Image: Archives | 🖃 🔒 Tag Logging | Process tag | | Ŧ | Selection | |
| Archives Image: Process Value Archives Image: | Timers | 1 Tank_Level | | Ð | General | |
| Image: Process Value Archives | Archives | 2 💥 | | Θ | General archive propertie | es |
| Image: Computer Image: Computer <td></td> <td>3</td> <td><i>)</i> </td> <td></td> <td>Archive name</td> <td>Tank_Level_Archi</td> | | 3 | <i>)</i> | | Archive name | Tank_Level_Archi |
| Salar Tank_Level Tag type Salar Tank_Level Tag type Salar Tank_Level Tag type Salar Tank_Level Tag name Fill_Level System Alarm Logging Salar Salar Tag Salar Salar Tag Salar Salar Tag Salar Tag type | Tank Level Archive | 4 | | Θ | General tag prope | |
| Tag type Analog 7 7 8 9 9 10 10 11 11 11 12 11 13 12 Alarm Logging 14 15 14 15 14 15 14 16 14 17 14 18 14 19 14 10 15 11 14 12 14 13 14 14 15 15 14 16 14 17 14 18 14 19 15 11 14 15 14 16 14 17 14 18 14 19 14 10 15 11 14 | 111 Canal Anthing | 5 | | | Process tag (A) | Tank_Level |
| Image: Computer | Archives | 6 | | | Tag type 🥥 | Analog |
| Image: Second state sta | | 7 | | | Tag name | Fill_Level_Archive |
| Image: Computer | <u> </u> | 8 | | | Tag supply | System |
| Computer 10 10 11 11 12 Acquisition type Cyclical, 12 13 13 14 15 15 Number of values leader 1 15 1 15 1 16 1 17 1 18 10 19 10 10 14 15 1 16 1 17 1 18 10 19 10 10 14 15 1 16 1 17 1 10 1 10 1 11 1 14 1 15 1 10 1 10 1 10 1 11 1 12 1 14 1 15 1 10 | | 0 | | | Also in tag | |
| Image: Computer | | 10 | | Ð | REST API | 4 |
| Tag Management 11 12 12 Alarm Logging 13 14 14 15 Number of values leader Number of values leader 100 15 14 16 17 17 18 18 19 19 10 10 10 11 10 12 14 15 10 14 15 15 16 16 17 17 18 18 19 19 10 10 10 11 10 12 14 15 10 18 10 19 10 10 10 11 10 12 10 14 10 15 10 10 10 10 10 10 10 10 10 <td>Computer</td> <td>11</td> <td>•</td> <td>Θ</td> <td>Archiving</td> <td></td> | Computer | 11 | • | Θ | Archiving | |
| Image: Addition cycle Fill Level Image: Addition cycle Fill Level Image: Addition cycle I day Ima | Tag Management | 12 | | | Acquisition type | Cyclical, Cyclical |
| Alarm Logging 13 Factor for archiving cycle 1 day 14 14 14 1 hour 15 15 Number of values leader 1 minute 16 16 14 1 minute 10 15 10 1 minute 10 10 1 minute 1 second 500 ms 100 100 1 minute | m | 12 | | | Acquisition cycle | Fill Level |
| Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state | Alarm Logging | 13 | | | Factor for archiving cycle | 1 day |
| Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state Image: Second state | φ | 14 | | | Archiving/display cycle | 1 minute |
| 500 ms | 🔁 🖪 🗶 🖴 🌗 🐈 🔡 🐨 🔹 | | - | | Number of values leader | 1 second |
| | | | .11 | | | 500 ms |

3. 取得サイクルおよびアーカイブサイクルに設定時間"Fill_Level"を割り付けます。

- 4. 作成されたプロセス値アーカイブ"Tank_Level_Archive"にあるナビゲーションエリアでクリ ックします。
- 5. プロセス値アーカイブ"Tank_Level_Archive"のメモリの場所とサイズを定義します。

| Tag Logging « | < Fir | nd 🚽 | • | Properties - Proce | ss value archive |
|-------------------------------|-------|-------------|-------|----------------------------|-----------------------|
| ∃] Tag Logging | | Process tag | 1 | 3 Selection | |
| Timers | 1 | Tank_Level | | Object type | Process value archive |
| Archives | 2 | × | | Object name | Tank_Level_Archive |
| Barrier Process Value Archive | ac 3 | 1989a | | 3 General | |
| 111 Track Lowel Archive | . 4 | | | Comment | |
| | ve 5 | | | Archiving disabled | |
| Archives | 5 6 | | | Manual input permitted | V |
| | 7 | | | Last change | 20.04.2023 16:02:3 |
| | - | | - 6 | General archive proper | ties |
| | 8 | | - | Archive name | Tank_Level_2 |
| | 9 | | _ | Action at archive start/er | 1a 🕻 💾 |
| Computer | 10 | | - ' E | Memory location | C C |
| | 11 | _ | _ | Memory location | Main memory |
| Tag Management | 12 | _ | | Size in data records | 100 |
| | 13 | | | Size in kB/tags | 3 |
| | 14 | | | | 3 |

6. "タグロギング"エディタを閉じます。

8.6 プロセス値アーカイブの編集

結果

プロセス値アーカイブが編集されます。

内部タグ[Tank_Level]の値が2秒毎に取得され、アーカイブタグ[Fill_Level_Archive]に保存 されます。

タグ値がメインメモリにアーカイブされ、ランタイム中だけ利用できます。

次のステップ

保存された値をランタイムのトレンドとして出力するには、[グラフィックデザイナ]エデ ィタの[トレンド]ウィンドウを設定します。

8.7 プロセス画面の設定(アーカイブシステム)

概要

このセクションでは、プロセス値アーカイブからの出力値を表示するプロセス画像の設定 方法を説明します。

原理

プロセス画像を[グラフィックデザイナ]エディタで設定します。

このため、以下のオブジェクトを使用します。

- WinCC OnlineTrendControl
 "WinCC OnlineTrendControl"オブジェクトは、[トレンド]ウィンドウを作成するために 使用します。
 [トレンド]ウィンドウでは、プロセス値アーカイブからの値はトレンドとして出力され ます。
- WinCC OnlineTableControl
 "WinCC OnlineTableControl"オブジェクトは、[テーブル]ウィンドウを作成するために使用します。
 [テーブル]ウィンドウでは、プロセス値アーカイブからの値はテーブルとして出力されます。

各出力はランタイムで行なわれます。

8.7.1 [トレンド]ウィンドウの設定

概要

次のステップは、[トレンド]ウィンドウの設定方法を示しています。

[トレンド]ウィンドウは、[グラフィックデザイナ]エディタで設定します。

このため、新しいプロセス画像を作成します。プロセス画像に、オブジェクト"WinCC OnlineTrendControl"を挿入します。

"Quick_Start"プロジェクトで、このオブジェクトをアーカイブタグ"Fill_Level_Archive"と接続させます。

これで、ランタイムでアーカイブタグに保存された値がトレンドとして出力されます。

必要条件

• プロセス値アーカイブ"Tank_Level_Archive"が作成されている。

手順

- 1.「Tag_Logging.Pdl」という名前の新しいプロセス画像を作成し、[グラフィックデザイナ]エ ディタで開きます。
- 2. [コントロール]ウィンドウから、「Tag_Logging.Pdl」プロセス画像に、「WinCC OnlineTrendControl」オブジェクトを挿入します。



[グラフィックデザイナ]エディタの作業エリアに、[トレンド]ウィンドウが表示されます。 [WinCC OnlineTrendControl のプロパティ]ダイアログが開きます。

3. [トレンド]ウィンドウのタイトルとして"Tank_Level_Trends" と入力し、デフォルト設定を適用します。



4. 時間軸の設定を定義します。

| ツールバー ステータスパー | オンラインコンフィグレーション エクスホペート |
|--|--|
| トレント* 全般 フォント | トレントウィン |
| 時間軸: 日本 時間軸 1 新規作成 削除 上へ 下へ | オフジェクト 時間軸 1 トレントヴィントウ: トレントヴィントウ 1 ラへリ: 配置: 0 - 下揃え 公 最新の情報の更新 |
| 時間範囲 | |
| 設定: | 開始時間: |
| 2-測定点の数 ~ | 19.10.2018 ■▼ 14:37:42 終了時間: 19.10.2018 ■▼ 14:38:42 |
| 測定占の数: | 時間範囲: |
| 120 | 1 X 1/2 ~ |
| 効果 時間の形式: 自動 ~ 日付の形式: 自動 ~ | 色: |

5. 値軸の設定を定義します。

| ツールバー | ステ | ータスパー | オンラインコンフィグレーション エクスホペート |
|--|------------|-----------|--|
| トレント | 全般 | フォント | トレントウィントウ 時間軸 値軸 |
| 値軸: ✓ 値軸 1 新規作成 値範囲 | 削除 | 上へ 下/ | オフジェクト名: 値軸1 トレントウィントウ: トレントウィントウ1 ラペル: 配置: 0 - 左揃え 3 ~ スワーリング: 0 - 線形 |
| 開始: 0 | <u>أ</u> | 态了: 10 | ☑自動 |
| 効果 | | | |
| 小数点均 | 、 下の桁数: | | 色: |
| 2 | ☑ 自動 | b | 4 |
| □指数詞 | 長言己 | | |
| - ユーザ・ースケ | -リンゲ | | |
| | | | 家 小 [] |

- ツールバー ステータスパー オンラインコンフィグレーション エクスホペート 12 トレント トレントウィントウ 時間軸 値軸 フォント トレンドミ わジェ外名: Tank Tank_Level 2 トレントウィントウ: トレントウィントウ 1 ~ 時間軸: 時間軸1 \sim 値軸: 値軸 1 \sim ラベル 新規作成 削除 上へ 下へ □コメントをカーブ名として使用する データ接続 データソース: 妙名<mark>:</mark> ~ □ アラームの表示 1-アーカイフラグ 効果 トレントウイフキ トレンド色: 1-点を直線で結ぶ \sim 2 □塗りつぶし 線の太さ 線のスタイル: 0-実線 1 \sim 点の幅<mark>:</mark> 点の種類: \sim 3 2-四角 点の色: 塗りつぶし色: 制限値 □拡張
- 6. トレンド名前"Tank_Level"を入力し、 📴 をクリックします。

タグの選択用ダイアログが開きます。

 [Fill_Level_Archive]アーカイブタグを[タグ]ダイアログで選択します。 タグを表示するには、[データソース]エリアで[WinCC タグ]オプションの選択が必要な場合 があります。

| Tags - Project: C:\Users\Public\Docur | nents\Siemens\Win | CCProjects\Quick_St | art\Quick | ? × |
|--|------------------------------|-------------------------------|-----------|------|
| | | | | |
| Filter: | | | | |
| WinCC Server WCC-NIGHTLY-05 Archives Tank_Level_Archive | Name * Fill_Level_Arcl | Parameter * hive Analog | Commen | nt · |
| | | ОК | Cancel | Help |

8. 時間軸と値軸を選択します。

| ツールパー | ステー | タスパー | オンラインコンフィグレージ | 7a2 | エクスホペート |
|----------------------|---------------|----------|---|----------------|---------|
| FD7F | 全般 | フォント | トレントウィントウ | 時間軸 | 値軸 |
| トレンド: I Tank_Leve | el | | オフジェクト名: Tank_Level トレントウィントウ: トレントウィントウ 1 時間軸: 時間軸 1 値軸: <u>「値軸 1</u> ラヘル: | | |
| 新規作成 | 削除 | 上へ 下へ | □ コメントをカーフ 注 | らとして使用す | 3 |
| ーデータ接続 — データソース: | | | 妙名: | | |
| 1 - アーカイフ | <u>፝</u> ም/ ~ | □ アラームの妻 | 示 Tank_Level_Ard | hive\Fill_Leve | l 属 🚰 |

9. [OK]をクリックして、ダイアログボックスを閉じ、設定を適用します。 10.「Tag Logging.Pdl」プロセス画像を保存します。

別の手順

- [タグロギング]エディタのテーブルペインにあるアーカイブタグ"Fill_Level_Archive"の付いた 行を選択します。 マウスを選択長方形の端に動かします。 カーソルが十字型ポインタ[+]に変わります。
- 2. タグの選択内容をグラフィックデザイナの画像の空白エリアにドラッグしている間、マウ スの左ボタンを押したままにします。
- 画像でマウスボタンから指を放します。 WinCC OnlineTrendControl が作成されます。 作成されたコントロールには、選択した[Fill_Level_Archive]アーカイブタグのデータ接続の あるトレンドが含まれています。
- 4. 上記手順で説明したように、ステップ3から5および8に進みます。

結果

[トレンド]ウィンドウ"Tank_Level_Trends"が設定されます。

ランタイムで、このウィンドウは、トレンドとしてアーカイブタグ[Fill_Level_Archive]に 保存される値を出力します。

| Ta | ag_Log | gging.Pdl > | ٢ | | | |
|----|-------------|----------------------|---------------------------|---------------------------|---------------------------|--------------|
| | 🛛 Tanl | k_Level_Tren | ds | @ ♣ ₿ | | x z |
| 1 | 10.0 7.5 | | | | | |
| L | 5.0 | | | | | |
| | 0.0 | | | | | |
| - | - | 19:00 AM /11/2018 | 11:19:15 AM 10/11/2018 | 11:19:30 AM 10/11/2018 | 11:19:45 AM 10/11/2018 | 4 11: 10/ |
| Re | eady | | | Ē | 11:45:44 / | AM: |

次のステップ

ロギングタグに保存されている値をランタイムのテーブルとして出力するには、[グラフ ィックデザイナ]エディタの[テーブル]ウィンドウを設定します。

8.7.2 [テーブル]ウィンドウの設定

概要

次のステップは、[テーブル]ウィンドウの設定方法を示しています。

[テーブル]ウィンドウは、"グラフィックデザイナ"エディタで設定します。このために 「Tag_Logging.Pdl」プロセス画像を使用します。

プロセス画像に、オブジェクト「WinCC OnlineTableControl」を挿入します。

"Quick_Start"プロジェクトで、このオブジェクトをアーカイブタグ"Fill_Level_Archive"と接続させます。

これで、ランタイムでアーカイブタグに保存された値がテーブルとして出力されます。

必要条件

- プロセス値アーカイブ"Tank_Level_Archive"が作成されている。
- •「Tag_Logging.Pdl」プロセス画像が開いている。

手順

1. 「Tag_Logging.Pdl」プロセス画像に、[WinCC OnlineTableControl]オブジェクトを挿入しま す。



[グラフィックデザイナ]エディタの作業エリアに、[テーブル]ウィンドウが表示されます。 [WinCC OnlineTableControl のプロパティ]ダイアログが開きます。

2. [テーブル]ウィンドウに"Tank_Level_Tables"という名前を入力します。

| 時間列 値列 オペシータメッセージ | オンラインコンフィグレーション エクスホペート |
|--|---|
| 全般 八字 効果 選択 | フォント ツールバー ステータスパー |
| ウイント ウイント ウイント ウイント 1 - 通常 ✓ テキスト: Tank_Level_Tables 図移動可能 〇 グロースで可能 〇 グリースで可能 〇 グリースで可能 〇 フロジェクト設定 ✓ 行ス加ールパー: ✓ | 精度 (係数: 時間単位: ① × 同一 ~ とやチャを開く ジ更新開始 ジアーカイフ*からのデータのロート* 列の色の使用 ジョオント色 1 背景色 |
| 1-必要時 ~ 列スカールパー: 1-必要時 ~ | 編集 🗹 編集を許可する |
| -現在の印刷シショフを表示 OnlineTableControl - Table 時間ヘ ^x ース 2 - ブロジェ外設定 v | |

3. [値]列に"Tank_Level"という名前を入力し、 🕞 をクリックします。

| | 選択 | フォント ツールパー ステータスパー |
|-----------------|--------|-------------------------|
| | メッセーシー | オンラインコンフィクレーション エクスホペート |
| 値 💭) | | オブジェクト名: |
| - Turnk_Level | | Tank_Level |
| | | 時間列: (Aǐ) |
| | | 時間列1 ~ |
| | | <u> </u> ን^ነሁ: |
| | | |
| | | 配置: 長さ(文字数): |
| | | 0-左揃え ~ 16 |
| 新規作成 削除 上へ | 下へ | |
| テジータ接続 | | |
| データソース: | | 奶治: |
| 1 - アーカイフ ፟ጶጛ | \sim | |
| | | |
| - レイアウト | | |
| ☑ テキストとしてのコンテンツ | | |
| アイコンとしてのコンテンツ | | |
| マテキストとしてのタイトル | | □ 指数表記 |
| | | = 3.1 Å |
| M 713286C091Nk | | 7374世: |
|)334 m | | 北日 会。 |
| 選択 | | 百京巴: |
| 福集 | | 4 |
| | | 限界値 |
| | | |

アーカイブタグ選択のダイアログが開きます。

 [Fill_Level_Archive]アーカイブタグを[タグ]ダイアログで選択します。 タグを表示するには、[データソース]エリアで[WinCC タグ]オプションの選択が必要な場合 があります。

| III Tags - Project: C:\Users\Public\Docur | nents\Siemens\Win Data source: 🗾 Wi | CCProjects\Quick_St | art\Quick | ? | × |
|---|--|-------------------------------|-----------|---------------|---|
| Filter: Gradient WinCC Server WinCC Server WCC-NIGHTLY-05 Frank_Level_Archive | Name * Fill_Level_Arc | Parameter * hive Analog | Comr V | 🗁 🏥 1 nent | 1 |
| | | ОК | Cancel | Help | |

- 5. [OK]をクリックして、[WinCC OnlineTableControl のプロパティ]ダイアログを閉じます。
- 6.「Tag_Logging.Pdl」プロセス画像を保存します。
- 7. グラフィックデザイナを閉じます。

結果

[テーブル]ウィンドウ"Tank_Level_Tables"が設定されます。

ランタイムで、このウィンドウは、表としてアーカイブタグ[Fill_Level_Archive]に保存される値を出力します。

| | Tag_Log | gging.Pdl | × | | | | | | | | | |
|----|----------|----------------------|-----------|-------------|-----------------|-----|---------------|---------------|---|---------------|---------------|-------|
| r, | | | | _ | _ | - | - | - | - | - | - | |
| Ľ | 🖆 Tanl | k_Level_Tre | nds | | | | | | | | | 23 |
| | 2 | | ${\rm I}$ | | 9 9 | (🚸 | ţ] | 1:1 | | | 1 | £} |
| | 10,0 | | | | | | | | | | | |
| | 7,5 - | | | | | | | | | | | |
| | 5,0 | | | | | | | | | | | |
| | 2,5 | | | | | | | | | | | |
| | 0,0 - | | _ | | | | | | | | | |
| l | | 12:08:30 11.10.20 |) 18 | 12: 11.1 | 08:45 0.2018 | : | 12:0 11.10 | 9:00 .2018 | | 12:0 11.10 | 9:15 .2018 | |
| L | Readv | | | | | | | | | el 1 | 2:45:5 | 57 .: |
| L | | | | | | | | | | | | |
| | III Tanl | k Level Tał | oles | | | | | | | | | 8 |
| ſ | 2 | | | BB I | N 2 | | | | ٢ | ¢ | | |
| | T | Time colum | nn 1 | Tank | _Level | | | | | | | |
| | 1 | | | | | | | | | | | |
| | 2 | | | | | | | | | | | |
| | 4 | | | | | | | | | | | |
| | 5 | | | | | | | | | | | |
| | 6 | | | _ | | | | | | | | |
| | 8 | | | | | | | | | | | |
| | 9 | | | | | | | | | | | |
| | 10 | | | | | | | | | | | |
| | | | | | | | | | | | | |

次のステップ

- •「Tag_Logging.Pdl」ページをユーザー定義メニューに追加します。
- WinCC Runtime のプロパティを定義します。

8.8 ユーザー定義メニューのカスタマイズ(アラームシステム)

8.8 ユーザー定義メニューのカスタマイズ(アラームシステム)

概要

次のステップは、[タグロギング]メニュー項目をカスタマイズされた[画像変更]メニューに 追加する方法を示しています。

メニューエントリ"タグロギング"をプロシージャ"ActivatePicture(ByVal PictureName)"にリ ンクさせます。

変更するプロセス画像の名前を、[ユーザーデータ]フィールドに入力します。

必要条件

- プロシージャ"ActivatePicture(ByVal PictureName)"が作成されている。
- ユーザ定義の[画像変更]メニューが「START.Pdl」および「SAMPLE.Pdl」のプロセス画 像用に作成されていること。

8.8 ユーザー定義メニューのカスタマイズ(アラームシステム)

手順

1. 保存された設定[ActivatePicture.mtl]を[メニューとツールバー]エディタで開きます。



2. [タグロギング]メニューコマンドを作成します。

| T Menus and toolbars [ActivatePicture |] - WinCC C | Configuration Stud | lio | | | - 0 | × | ŝ |
|--|-------------|--------------------|------------------|----------|---------|-----------|-----|---|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | |
| Menus and toolbars [Act « | 📑 Men | u items [Pict | ur Find | | | Q | - | < |
| | Name | e | Text | Active | Visible | Separator | ^ τ | , |
| 🖮 👕 Picture Change | 1 Start | 9 | Start | 1 | 1 | | 3 | |
| 1 Start | 2 Sam | ole | Sample | 1 | 1 | | Der | |
| (💾)) 📻 Sample | 3 Tag | Logging | Tag Logging | V | 1 | | | |
| Tag Logging | 4 💥 | | | | | | S S | 8 |
| Toolbars | 5 | | 1 | | | | | |
| | 6 | | | | | | | |
| | 7 | | | | | | | |
| | 8 | | | | | | J | |
| □ 9. 0 6 0 8 6 4 ☆ - | 14 A F F | Menu items | | < | | > | | |
| Ready NUM | English (Ur | nited States) Tal | ole: 3 Menu item | is 10 | 0% 🗩 | 0 | • | |

8.8 ユーザー定義メニューのカスタマイズ(アラームシステム)

3. [タグロギング]メニュー項目を設定して、「Tag_Logging.Pdl」画像に画像変更を実行できる ようにします。



4. [メニューとツールバー]エディタで変更内容を保存します。

5. "メニューとツールバー"エディタを閉じます。

結果

[タグロギング]メニュー項目を[画像変更]メニューに追加しました。

メニュー項目を使用して、ランタイムで「START.Pdl」、「SAMPLE.Pdl」および 「Tag_Logging.Pdl」のプロセス画像に切り替えます。

8.9 ランタイムプロパティの定義(アラームシステム)

概要

次のステップは、WinCC Runtime のプロパティの定義方法を示しています。

このセクションでは、プロジェクトの起動時にタグロギングランタイムが実行されるように WinCC Runtime をセットアップします。「Tag_Logging.Pdl」プロセス画像を、ランタイム ウィンドウの開始画像として選択します。

必要条件

• 「Tag_Logging.Pdl」プロセス画像が作成されている。

手順

1. [コンピュータ]エディタのナビゲーションエリアでショートカットメニューによりコンピ ュータのプロパティを開きます。



コンピュータのプロパティが開きます。

2. [スタートアップ]タブの[タグロギングランタイム]アプリケーションを選択します。

| ⊐2℃1−ダのフ [°] ロハ [°] ティ ™ | × |
|---|-------|
| 全般 スタートアッフ。 バラズ ゲラフィックスランタイム ランタイム | |
| WinCC Runtimeの起動命令(0): | |
| | |
| | |
| マンジャイム レーケンスレポート /SEQPROT | |
| Cloud Connector | |
| | |
| | 編集(T) |

3. [グラフィックランタイム]タブで、開始画像として「Tag_Logging.Pdl」プロセス画像を定義します。

8.9 ランタイムプロパティの定義(アラームシステム)

| 全般 スタートアッフ ハペラメータ クミ | ラフィックスランタイム マシルイ |
|--------------------------------|--|
| プロジェクトファイル: | |
| C:\Users\Public\Documents\Sier | mens\WinCCProject=rquick_Start\Quick_Start.mcp |
| 画像の開始: | コンフィクシーションメニューとツールバーの起動: |
| START.Pdl | vatePicture.mtl |
| ウィントウ属性: | |
| I b./kii | |
| ■ ■ 画像: | ? × |
| | |
| 階層: | |
| DESKTOP-NTN3TGA | 771ル名 |
| | * 7 |
| | -*r SAMPLE.PdI |
| | -fr START.Pdl |
| | Tag_Logging.Pdl |
| 3 | |
| | |
| | |
| | OK 4+1/17/L 1/1/7°(H) |
| | |

結果

WinCC Runtime のプロパティが定義されています。「Quick_Start」プロジェクトを有効に するときに、タグロギングランタイムが実行され、「Tag_Logging.Pdl」プロセス画像が表示 されます。

次のステップ

「Quick_Start」プロジェクトを有効にし、テストします。

8.10 プロジェクトの有効化とテスト(アーカイブシステム)

8.10 プロジェクトの有効化とテスト(アーカイブシステム)

概要

次のステップは、"Quick_Start"プロジェクトで設定の有効化およびテスト方法を示しています。

WinCC タグシミュレータは、ランタイムの内部タグ「Tank_Level」に値を割り付けます。 これらの値は"Quick_Start"プロジェクトで2秒毎に取得され、アーカイブタグ [Fill_Level_Archive]に保存されます。

タグロギングランタイムがアーカイブ値を読み取り、[トレンド]ウィンドウおよび[テーブル]ウィンドウに転送します。値は、トレンドおよびテーブルとして出力されます。

必要条件

ランタイムプロパティが定義されている。

手順

- 1. 「QuickStart_Simulation.sim」設定を[タグシミュレーション]エディタで開きます。
- 2. ▶ボタンを使用して、WinCC エクスプローラで WinCC Runtime を有効にします。
- 3. [シミュレーション]>[起動]を使用して、[タグシミュレーション]エディタでシミュレーションを起動します。

8.10 プロジェクトの有効化とテスト(アーカイブシステム)

4.「Tag_Logging.Pdl」プロセス画像で、シミュレーション値の出力を監視します。



- 5. [シミュレーション]>[停止]を使用して、[タグシミュレーション]エディタでシミュレーションを停止します。
- 6. ツールバーの ボタンを使用して、WinCC Runtime を終了します。

結果

"Quick_Start"プロジェクトが有効化され、内部タグ"Tank_Level"の値がシミュレーションされます。

設定済みの[トレンド]ウィンドウおよび[テーブル]ウィンドウに、値シーケンスが表示さ れます。 8.10 プロジェクトの有効化とテスト(アーカイブシステム)

メッセージの設定

概要

このセクションでは、アラームロギングに関する情報や[アラームロギング]エディタでの メッセージの設定方法を説明します。

原理

アラームロギングはプロセスを監視します。

アラームシステムの[アラームロギング]エディタで次のメッセージを設定できます。

- ディスクリートアラーム プロセスのステータスの変化を表示します。
 ディスクリートアラームは、PLCによってトリガされます。
- アナログアラーム: 制限値違反を表示します。
 アナログアラームは、設定した制限値が違反されているときにトリガされます。

「Quick Start」 プロジェクトの設定

"Quick_Start"プロジェクトで、メッセージを設定して、給水バルブやバイオコンバータの 充填レベルを監視します。

- 給水バルブをシミュレートするには、新しい内部タグを作成します。
 [アラームロギング]エディタで、給水バルブの異なるステータスに対して、ディスクリートアラームを設定します。
 タグ値で特定のビットが設定されると、ディスクリートアラームがトリガされます。
- 内部タグ[Tank_Level]の値を使用して、"Quick_Start"プロジェクトのバイオコンバータの充填レベルをシミュレーションします。 フィルレベルの下限値および上限値を設定します。
 内部タグ"Tank_Level"の値が限度に違反すると、対応するアナログメッセージがトリガされ、ランタイムで表示されます。

9.1 メッセージシステム

9.1 メッセージシステム

はじめに

メッセージシステムは、プロセスの監視に使用する WinCC のサブシステムです。

特定のステータスやプロセスにおける変化によって、メッセージシステムはメッセージを 生成し、それらをランタイムのテーブルに出力します。メッセージは、危機的な状況を初 期段階で認識し、ダウンタイムを回避するのに役立ちます。

メッセージシステムのコンポーネント

メッセージシステムは、設定コンポーネントおよびランタイムコンポーネントで構成されています。

- メッセージシステムの設定コンポーネントは"アラームロギング"エディタです。[アラ ームロギング]エディタで、主に以下のタスクを実行できます。
 - メッセージの作成
 - メッセージの準備
 - 制限値の設定
 - メッセージテキストとメッセージ状態の表示
 - メッセージの承認プロパティの定義
 - メッセージのアーカイブプロパティの定義
- アラームロギングランタイムは、メッセージシステムのランタイムコンポーネントです。アラームロギングランタイムは、主に次のタスクを実行するために使用されます:
 - 定義済みモニタリングの実行
 - メッセージ出力の制御
 - 確認応答の管理


WinCC システムでのメッセージシステムおよびメッセージアーカイブ

9.2 アラームロギングの起動

9.2 アラームロギングの起動

概要

次のステップは、[アラームロギング]エディタの開始方法を示しています。

"アラームロギング"エディタで、"Quick_Start"プロジェクトに必要な全てのビットメッセ ージおよびアナログメッセージを設定します。

必要条件

• "Quick_Start"プロジェクトが開いている。

手順

[アラームロギング]エディタを起動します。
 [アラームロギング]エディタが開きます。

結果

"アラームロギング"エディタが開いています。

次のステップで、メッセージブロックとメッセージクラスを定義します。

| 👔 Alarm Logging - WinCC Configurati | ion SI | udio | | | | - | | × | < |
|--|--------|------------|----------------|-------------|---------------|----------------------|----------|----------|------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | | |
| Alarm Logging « | | Message | es [Selectio | Find | | | Q | - | « |
| | | Number | Message tag | Message | Message class | Message [•] | Mes | ^ - | 0 |
| 🖶 🗐 Message blocks | 1 | 1 | | 0 | . U | | | <u> </u> | 5 |
| | 2 | 米 | X | | | | | <u>d</u> | É |
| | 3 | | | | | | | | , |
| 🦳 🤤 Limit monitoring | 4 | | | | | | | ľ | " |
| | 5 | | | | | | _ | | |
| | 6 | | | | | | | | |
| 8 | 7 | | | | | | | | |
| | 8 | | | | | | | ~ | |
| | 14 | i de del 🔤 | essage classes | Messag | (< | | > | : | |
| Ready NUM | | Germar | i (Germany) Ta | ble: 1 Mess | age 100 % 🤅 |)(| <u> </u> | ÷ |) ., |

9.3 メッセージブロックの設定

概要

次のステップは、「Quick_Start」プロジェクトのメッセージに対するメッセージブロックの 設定方法を示しています。

メッセージが、ランタイムのテーブルに表示されます。

メッセージブロック

それぞれのメッセージは、テーブルの列に表示される情報で構成されています。

これらの個別の情報が、メッセージブロックとして参照されます。それぞれのメッセージブロックは、テーブルの列に対応しています。

メッセージブロックは、3つのグループに細分されます。

- システムデータ(例:日付、時刻、メッセージ番号、ステータス)を有するシステムブロック。
 システムブロックは、事前定義されています。
- ユーザーテキストブロックには、説明テキスト(例:ロケーションおよび障害の原因に関する情報のあるテキスト)があります。
 テキストは自由にカスタマイズできます。
- メッセージをプロセス値(例:現在のフィルレベル、温度、速度)に接続するために使用するプロセス値ブロック。

メッセージブロックをランタイムに表示するためにプロパティを変更できます。

"Quick Start"プロジェクトに使用されるユーザーテキストブロックの長さを変更します。

必要条件

[アラームロギング]エディタが開いている。

9.3 メッセージブロックの設定

手順

- ナビゲーションエリアで、[メッセージブロック]フォルダにある[システムブロック]項目を 選択します。
- 2. ランタイムに表示するシステムブロックを有効にします。
 - "日付"
 - "時間"
 - "番号"

| Alarm Logging « | | Message | block Find | • | < |
|----------------------|---|-----------|---------------------------------|---|----|
| H-Messages | | Used | Message block | | τ |
| Hessage blocks | 1 | V | Date | | 0 |
| System blocks | 2 | | Time | | ĕ |
| User text blocks | 3 | | Duration | | E |
| Process value blocks | 2 | | Daylight Saving / Standard Time | | v, |
| Message groups | Y | | Status | | |
| System messages | 6 | | Acknowledgment Status | | |
| | 7 | J | Number | | |
| | 8 | | Class | | |
| | 9 | | Туре | | |
| | H | A P PI MO | essage blocks | : | |

- 3. [ユーザーテキストブロック]項目をナビゲーションエリアで選択し、[メッセージテキスト] および[エラーのポイント]ブロックを有効にします。
- 4. テーブルエリアまたは[プロパティ]ウィンドウのいずれかで、[文字数]フィールドのテキストの長さを変更します。
 - メッセージテキスト:30 文字
 - エラーのポイント:25 文字



9.3 メッセージブロックの設定

結果

"Quick_Start"プロジェクトのメッセージに対するメッセージブロックが定義されています。

次のステップ

3つのディスクリートアラームを設定して、給水バルブのステータスの変化を監視します。

9.4 ビットメッセージの設定

概要

このセクションでは、[アラームロギング]エディタでのディスクリートアラームの設定方法 を説明します。

概要

それぞれのディスクリートアラームは、バイオコンバータでの給水バルブのステータスに 対応します。

- Valve_open (バルブが開いている)
- Valve_closed (バルブが閉じている)
- Valve inop (バルブが故障している)

給水バルブのステータスが変化すると、対応するディスクリートアラームがトリガされ、 ランタイムに表示されます。

新しい内部タグを作成して給水バルブの様々なステータスをシミュレーションできます。次 いで、このタグを、作成したディスクリートアラームのメッセージタグとして設定します。 それぞれのディスクリートアラームに以下のプロパティを定義します。

| メッセージタグ | メッセージタグは、プロセスにおけるステータスの変化にリンクして |
|---------|---|
| | います。 |
| | プロセスでステータスの変化が発生すると、タグ値のビットが設定さ |
| | れます。タグ値に応じて、ディスクリートアラームがトリガされます。 |
| メッセージビッ | メッセージビットは、どのビットがディスクリートアラームをトリガ |
| F | するかを定義します。 |
| メッセージテキ | メッセージテキストは、例えば[バルブが開いている]など |
| スト | の、"Quick_Start"プロジェクトにおける給水バルブのステータスを説明 |
| | します。 |
| エラーの位置 | エラーの位置はステータス変更の場所を示します。 |

9.4.1 ビットメッセージの作成

概要

次のステップは、[アラームロギング]エディタでのディスクリートアラームの作成方法を示 しています。

"Quick_Start"プロジェクトでは、3つのディスクリートアラームをメッセージクラス[エラー]、メッセージタイプ[アラーム]で作成します。

必要条件

- [アラームロギング]エディタが開いている。
- 新しい内部タグをタグ管理で作成している。
 - 名前:"Inflow_Valve"
 - データタイプ:"符号なし16ビット値"

内部タグの作成(ページ43)に説明されている手順に従ってください。

手順

- ナビゲーションエリアの[メッセージ]フォルダで、[エラー]の下にある[アラーム]メッセージタイプを選択します。
 メッセージクラスとメッセージタイプの表示言語を変更できます。[表示]>[入力言語]メニューで、希望の言語を選択します。
 メッセージのプロパティを編集するには、[プロパティ]ウィンドウを開きます。
- 2. テーブルエリアで、先頭の行を選択し、次のプロパティを定義します。
 - メッセージタグ:"Inflow Valve"
 - メッセージビット:2
 - メッセージテキスト:"Valve_open"
 - エラーのポイント: "Valve"

| 🖄 Alarm Logging - WinCC (| Confi | guration S | itudio | | | | |
|---|--------------|------------|----------------|------------|---------|-----------------------|---|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elj | þ | | | | | |
| Alarm Logging | | Find | | ۶. | - | Properties - Message | • |
| 🖃 🖂 Messages 🌘 🎽 | | Number | Message tag | Message I | . E | Selection | 3 |
| 🖨 🖼 Error | 1 | 1 | Inflow_Valve | 2 | E | General | <u> (()) </u> |
| - Alarm | 2 | - 2 | ※ | | E | Tags | |
| | 3 | | | | | Message tag Inflow_ | Valve 🔤 |
| | 4 | | | | | Message bit 2 | |
| | 5 | | | | | Status tag | 4 |
| System, requires i | 6 | | | | | Status bit 0 | |
| 🗄 🔛 System, without : | 7 | | | | | Acknowledgment ta | |
| 🗄 📃 Message blocks | 0 | | | | | Acknowledgment bit 0 | |
| - 🔁 Message groups | 0 | | | | Œ | Parameter | |
| | 9 | | | | Œ | Extended | AI |
| 🔄 Limit monitoring | 10 | | | | E | User Text Blocks | |
| | 11 | | | | | Message text Valve_0 | pen |
| | 12 | | | | | Point of error Valve | |
| 📼 🛼 🚺 🗟 🗷 × | 13 • | () H I | lessages 🖉 | | | Info text | |
| Ready NUM Gen | man | (Germany) | Selection: 1 M | essage Tab | le: 1 N | lessage 100 % 😑 — 🤇 | · • • |

2 つの追加のディスクリートアラームを作成します。
 それぞれのテーブルエリアで次に利用可能な行の[番号]列をクリックします。
 メッセージの番号を入力します。

- 4. [プロパティ-メッセージ]エリアで、2番目のディスクリートアラームの次のプロパティを 定義します。
 - メッセージタグ:"Inflow_Valve"
 - メッセージビット:3
 - メッセージテキスト:"Valve_closed"
 - エラーのポイント: "Valve"
- 5.3番目のディスクリートアラームに以下のプロパティを定義します。
 - メッセージタグ:"Inflow_Valve"
 - メッセージビット:4
 - メッセージテキスト:"Valve inop"
 - エラーのポイント: "Valve"

結果

"Quick_Start"プロジェクトのプロパティを持つディスクリートアラームを定義しました。 作成されたディスクリートアラームは、[アラームロギング]エディタのテーブルウィンド ウで表示されます。

ランタイムでのディスクリートアラームの出力はメッセージの設定でコントロールされて います。例えば、内部タグ[nflow_Valve]の値で右から2番目のビットが設定されている場 合、ディスクリートアラーム"Valve_open"がトリガされます。

| 🖄 Alarm Logging - WinCC (| Confi | guration S | Studio | | | | - | | × |
|---|--------------|------------|-------------------|------------|-----------------|---------|-------------|------|-------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | 0 | | | | | | | |
| Alarm Logging « | | Messag | ges [Alarm] | | Find | | | ۰ م | • « |
| 📮 🖂 Messages | | Number | Message tag | Message | Message class | Message | 1 Message f | text | σ |
| 🖶 🖓 Error | 1 | 1 | Inflow_Valve | 2 | Error | Alarm | Valve_ope | en | 5 |
| | 2 | 2 | Inflow_Valve | 3 | Error | Alarm | Valve_clos | sed | e e |
| | 3 | 3 | Inflow_Valve | 4 | Error | Alarm | Valve_ino | p | tie |
| | 4 | 洸 | ※ | | | | | | » |
| 🖽 🖾 System, requires : | 5 | | | | | | | | |
| 🕀 🖾 System, without a | 6 | | | | | | | | |
| Hessage blocks | 7 | | | | | | | | |
| Message groups | 8 | | | | | | | | |
| Sustem messages | 9 | | | | | | | | |
| Jimit monitoring | 10 | | | | | | | | |
| | 11 | | | | | | | | |
| | 12 | | | | | | | | |
| 💷 🖫 🏠 🗟 🗷 🗷 - | 13 14 4 | | Messages | | I — | | | | |
| Ready NUM | | | English (United S | States) Ta | ble: 3 Messages | 100 % | 0 | | - + . |

次のステップ

アナログアラームを2つ設定して、内部タグ「Tank_Level」の動作を監視できるようにします。

9.5 アナログメッセージの設定

概要

この章では、"アラームロギング"エディタでアナログメッセージの設定方法を説明します。

概要

アナログアラームは、ランタイムでの制限値違反を表示します。

"Quick_Start"プロジェクトでのアナログメッセージの設定は、以下のステップから構成されています。

- 1. 監視するタグの定義 監視するタグを[アラームロギング]の[制限値モニタリング]で定義します。
- 2. 制限値の設定

"Quick_Start"プロジェクトで、内部タグ"Tank_Level"の動作を監視します。タグ値は、バイ オコンバータの充填レベルをシミュレーションします。

タグの上限値および下限値を定義します。

[アラームロギング]エディタは、それぞれの定義済み制限値に対してアナログメッセージを 作成します。

- 上限値は、バイオコンバータで許容される最大の水容量を定義します。
 上限値が違反される場合、バイオコンバータは溢れてしまいます。
 対応するアナログメッセージが、ランタイムで表示されます。
- 下限値は、バイオコンバータで必要とされる最小の水容量を定義します。
 下限値が違反される場合、バイオコンバータの充填レベルは危険なレベルに低下しています。
 対応するアナログメッセージが、ランタイムで表示されます。

9.5.1 制限値の設定

概要

次のステップは、内部タグ"Tank Level"の制限値の定義方法を示しています。

1つのタグに対して、制限値はいくつでも設定できます。"アラームロギング"エディタは、 それぞれの定義済み制限値に対してアナログメッセージを作成します。

"Quick Start"プロジェクトで、内部タグ"Tank Level"の上限値と下限値を定義します。

必要条件

- [アラームロギング]エディタが開いている。
- 内部タグ"Tank_Level"が設定されている。

手順

- 1. ナビゲーションエリアで、[制限値モニタリング]フォルダを選択します。
- 2. テーブルエリアで[タグ]列の先頭の空き行をクリックします。 内部タグ"Tank_Level"を選択します。

| 🔯 Alarm Logging - WinCC | Configuration Studio | | | - 0 | × |
|---|---------------------------------------|---------------------|-----------------------|-----------------|---------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | Help | | | | |
| Alarm Logging « | 🔄 Limit Valu |] | Find | + م | ~ |
| 🖽 🖂 Messages | Tag | Shared message | Message num | Comparison valu | υ |
| 🖶 🗐 Message blocks | 1 ▷♀ Tank_Level | | 0 | | P |
| 🛛 🔁 Message groups | 2 💥 | 3 | | | ĕ |
| - 🛃 System messages | 3 | | | | ties |
| 🔤 Limit monitoring | 4 | | | | 1 V |
| | | | | | |
| | 8 | | | | |
| 💷 🖫 🗘 🔁 🕮 🔹 | I I I I I I I I I I I I I I I I I I I | s / Messages / | 1 — | a | - |
| Ready NUM | lta | lian (Italy) Table: | 1 Limit Value 100 % | | - (±) , |

- 3. エントリを開くには、タグ名の前にある矢印をクリックします。 テーブルペインのタグの下で[上限]エントリを選択します。
- 4. 上限値のプロパティを[プロパティ 制限値]エリアまたはテーブルエリアに入力します。 新しい、未使用のメッセージ番号を使用します。例、"4"。

| 🛱 Alarm Logging - WinCC Confi | guration Studio | - O X |
|--|--------------------------------|--------------------------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | - 0 | |
| Alarm Logging « | Fin A | 🔀 Properties - Limit Value 🛛 » |
| ⊞ <u>S</u> Messages | T S Dessag | Selection |
| 🗄 🖃 Message blocks | 1 Tank Level | 🕀 Limit |
| Message groups | 2 High limit 4 | Comparison |
| System messages | 3 High limit | Comparison value 90 |
| V Limit monitoring | 4 Low limit | Comparison value tag |
| | 5 Value is same | Indirect |
| | 6 Value is different | Hysteresis ■ |
| | 7 | Message |
| | 2 | Message number 4 |
| | 0 | Consider quality code |
| | H I I I Limit Valu | |
| Ready NUM | Italian (Italy) Table: 3 Limit | /alues 100 % 😑 🗸 🖓 🕀 , |

- 5.2番目の制限値のデータを入力するには、タグの下の次の行で[下限]エントリを選択します。
- 6. 下限値のプロパティを[プロパティ 制限値]エリアまたはテーブルエリアに入力します。 新しい、未使用のメッセージ番号を使用します。例、"5"。

| 🐒 Alarm Logging - WinCC Configura | ation Studio | nineste por contractorio de la proc | – 🗆 X |
|--|---------------------------------------|-------------------------------------|---------------------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | |
| Alarm Logging « | F D | + م | 🔀 Properties - Limit Va » |
| ⊞ Messages | T | Shared 2)es Mess A | E Selection |
| 🗄 🔚 Message blocks | 1 A Tank_Level | | E Limit |
| | 2 High limit | 4 | Comparison |
| | 3 Low limit | - 5 | Comparison value 10 |
| V Limit monitoring | 4 High limit | | Comparison value tag |
| | 5 Low limit | | Indirect 5 |
| | 6 Value is same | 3 | Hysteresis (AI) |
| | Value is different | | E Message |
| | | | Message number 5 |
| | 8 | ¥ | Consider quality code |
| | I I I I I I I I I I I I I I I I I I I | es Messages 4 | |
| Ready NUM | English (United States) | Table: 4 Limit Values | 100 % |

結果

内部タグ"Tank_Level"の制限値が定義されています。

- タグ値が90より大きい場合、アナログメッセージ[上限値]がトリガされ、ランタイムで 表示されます。
 このメッセージは、バイオコンバータの充填レベルを超えたことを示しています。
- タグ値が10より小さい場合、アナログメッセージ[下限値]がトリガされ、ランタイムで表示されます。
 このメッセージは、充填レベルが許容レベルを下回っていることを示しています。

[制限値モニタリング]エントリをナビゲーションペインで選択し、[メッセージ]タブをク リックして作成済みのアナログアラームの概要を取得します。 メッセージの設定

9.5 アナログメッセージの設定

| 🖄 Alarm Logging - WinCC Config | urati | on Studio | | | | | _ | | × | |
|--|-------|-----------|------------|----------|------------------|------------|--------------|----------|-----|---|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s <u>H</u> elp | | | | | | | | | | |
| Alarm Logging « | | Messag | jes [Lii | nit moni | to Find | | | <u>م</u> | • « | < |
| ⊞ <u>M</u> essages | | Number | Message | Message | Message class | Message | Message text | P | τ | , |
| 🗄 🖃 Message blocks | 1 | 4 | | 0 | Error | Alarm | Limit @1%f@ | e | 0 | |
| | 2 | 5 | | 0 | Error | Alarm | Limit @1%f@ | U | e | |
| | 3 | | | | | | | | | |
| | 4 | | | | | | | | 0 | |
| - | 5 | | | | | | | | | |
| | 6 | | | | | | | | | |
| | 7 | | | | | | | | | |
| | 8 | | | | | | | | | |
| 🛄 🖫 🗘 🔁 🗐 🕷 📲 🔹 | H. | • • • _ | Limit Valu | es Mes | ssages | | | | : | |
| Ready NUM | | Englis | h (United | States) | Table: 2 Message | es 100 % | • <u> </u> | _ | | |

次のステップ

WinCC OnlineTrendControl での制限値違反の表示を設定します。

9.5.2 トレンドウィンドウでのメッセージの表示

概要

次のステップは、WinCC OnlineTrendControl でツールヒントとして制限値モニタリングの メッセージを表示する方法を示しています。

[タンクレベル]タグのトレンドを作成し、[アラームの表示]オプションを有効にします。 割り付けられたメッセージが、ランタイムで制限値違反のあるトレンド値に対するシンボ ルおよびツールヒントとして表示されます。

必要条件

- グラフィックデザイナで「Tag_Logging.Pdl」画像が開かれている。
- アラームロギングで制限値のモニタリングを設定している。
 次に説明されている手順に従ってください。制限値の設定(ページ 191)

手順

- 1. WinCC OnlineTrendControl [Tank_Level_Trends]を選択します。
- 2. ショートカットメニューを使用して、[WinCC オンライントレンドコントロールのプロパティ]設定ダイアログを開きます。



3. [Limit_Warning]トレンドを内部タグ[Tank_Level]に対して作成し、[アラームの表示]オプションを選択します。

タグ選択ダイアログでタグを表示するには、[データソース]エリアで[WinCC タグ]オプションの選択が必要な場合があります。

| ツールバー | ステ | ータスパー | オンラインコンフィグレージ | 242 | エクスホペート |
|-----------------|-----------------|------------|---------------|-------------|-----------|
| トレント | 全般 | フォント | トレントウィントウ | 時間軸 | 値軸 |
| NU2151 | | | わジェ外名: | | |
| ✓ Tank_L | .evel | | Limit_Warning | | |
| Limit_V | Varning | | トレントウィントウ: | | |
| | | | トレントウィントウ 1 | | ~ |
| | | | 時間軸: | | |
| | | | 時間軸 1 | | \sim |
| | | | (直車曲: | | |
| | | | (直車曲 1 | | \sim |
| (| | | ∋^າル: | | |
| | | | | | |
| 新規作成 | 削除 | 上へ一下へ | | ふとして使用する | 5 |
| データ接続 | | | | 10000000000 | |
| データソーン | Z: | | 妙名: | | 5 |
| 2 - オンラ | インタグ 、 | ✓ ☑ 75-403 | 表示 | | ia 🔗 🕺 |
| ᄷᆔᄪ | - 3 | | カニレーション | | × |
| 「別未」 「トレットなん | ъ. (Ц) | | | (| |
| 1-点を | 直線で結ぶ | | | | |
| 線のスタイ | 1. 1. | | が名 Tank_L | evel | |
| 0-実線 | 1 | | | | |
| 点の種類 | 頁: | | サイクル時間 1秒 | | ~ |
| 2 - 四角 | 3 | | -12 | —(🍋) | |
| 点の色: | 2 | 塗り: | | K(O) | |
| | | | | K(U) | 197 EN(C) |

4. 下限のトレンドの色変更を設定します。

| ツールバー ステータスパー | 限界値 |
|--|--|
| トレンド [*] トレント [*] : ✓ Tank_Level ✓ Limit_Warning 新規作成 削除 上へ 下/ | 下限値 値: 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 11 10 12 10 17 10 17 10 17 10 17 10 17 10 17 10 17 10 10 10 11 10 12 10 13 10 14 10 |
|) ーッソース: 2 - オンラインタケ V V アラームの3 | 大 Tank Level 🔀 😭 |
| | |
| 200米 トレントやイフ [*] : 1 - 点を直線で結ぶ 線のスタイル: 0 - 実線 との毎*毎。 | トレント色: 線の太さ: ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ |
| 点0/理规: 2-四角 | → 3 |
| 点の色: 塗りつぶし色: | ☆☆☆●は「「「」「「」」」は「「」」」では「「」」」では「「」」」」は「「」」」」は「「」」」」は「「」」」」」」 |

事前定義された色を適用したい場合は、[色選択]ダイアログで色パレットを選択します。

メッセージの設定

9.5 アナログメッセージの設定

| /inCC | | | |
|-------|----|-------------|--|
| | 0 | Red | |
| | 1 | DarkRed | |
| | 2 | Blue-Gray | |
| | 3 | DarkBlue | |
| | 4 | Black | |
| | 5 | Orange | |
| | 6 | Brown | |
| | 7 | Purple | |
| | 8 | LightBlue | |
| | 9 | StoneGrey | |
| | 10 | Yellow | |
| | 11 | GrassGreen | |
| | 12 | LightPurple | |
| | 13 | IceBlue | |
| | 14 | LightGrey | |
| | 15 | LemonYellow | |
| | 16 | LightGreen | |
| | 17 | OldPink | |
| | 18 | Beige | |
| | 19 | White | |
| | 20 | | |

5. 上限および不特定なステータスの値のトレンドの色変更を設定します。

結果

[Limit_Warning]トレンドを内部タグ[Tank_Level]に対して設定し、それを制限値モニタリ ングに関連付けます。

制限値の違反は、ランタイムで表示されます。

- 赤色のシンボルは、制限値が違反されていることを示します。
- ツールヒントはメッセージのメッセージテキストを表示します。
- トレンドは色を変更します。
- 制限値内に値が戻ると、シンボルは緑色になります。
 メッセージが引き続き表示されることがあります。



次のステップ

ランタイムでメッセージステータスを表示する色を定義します。

9.6 メッセージステータスの色の定義

9.6 メッセージステータスの色の定義

概要

次のステップは、異なるメッセージステータスに対して表示色を定義する方法を示してい ます。

WinCC では、メッセージステータスの基本タイプが3つに分けられています。

- メッセージの原因が存在する限り、メッセージは[着信]です。
- メッセージの原因が存在しなくなると、メッセージは[発信]になります。
- ユーザーがメッセージを確認すると、メッセージは"確認済み"になります。

それぞれのメッセージの現在のステータスは、ランタイムで異なる色で表示されます。個別 のメッセージステータスの表示色は、"アラームロギング"エディタで決定します。

"Quick_Start"プロジェクトで、3 つの基本ステータスに対して異なるテキスト色と背景色を 定義します。この定義は、メッセージクラス[エラー]のメッセージタイプ[アラーム]全体に 適用されます。これは、設定を「Quick_Start」プロジェクトの全てのメッセージに適用し ます。

必要条件

• [アラームロギング]エディタが開いている。

手順

- ナビゲーションエリアの[エラー]の下にある、[アラーム]メッセージタイプのフォルダを選択 します。
- 2. [プロパティ]エリアで、メッセージタイプの色を編集します。

| 🖄 Alarm Logging - WinCC | Configuration Studio | | - 0 X |
|---|-----------------------|-------------------------------------|---------------|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | | |
| Alarm Logging | Find P | 🛛 🖼 Properties - Message Type | » |
| 🖃 🖂 Messages 🛛 🚺 | Num Message tag Messa | Status Texts | 2 |
| 📄 🖼 Error | 1 1 Inflow_Valve 2 | ⊞ Tags | |
| | 2 2 Inflow_Valve 3 | | |
| | 3 3 Inflow_Valve 4 | Font color "came in" | 255; 255; |
| Eailure | 4 💥 💥 | Background color "came in" | 0 |
| - renore | 5 | Font color "went out" | 255; 255; 255 |
| 🖶 🖼 System, requi | 6 | Background color "went out" | 11 |
| 🕴 🖽 System, withc | 7 | Font color "acknowledged" | 255; 255; 255 |
| | 0 | Background color "acknowledged" | 3 |
| 📼 🖬 🔥 📾 🔊 🖉 | | Translation "Name" | |
| | MAPH Mes — | (ID) | 2 |
| Ready NUM | English (United St | ites) Table: 3 Messages 100 % 🕞 — | |

- 3. メッセージステータス[着信]の以下のプロパティを定義します。
 - フォント色:白色
 - 背景色:赤色
- 4. メッセージステータス[発信]の以下のプロパティを定義します。
 - フォント色:白色
 - 背景色:緑色
- 5. メッセージステータス「承認済み」の以下のプロパティを定義します。
 - フォント色:白色
 - 背景色:青色
- 6. "アラームロギング"エディタを閉じます。

結果

メッセージステータス[着信]、[発信]および[確認済み]の表示色が定義されています。

ランタイム中に、メッセージはステータスに応じてそれぞれの色で表示されます。

次のステップ

ランタイムのテーグルでメッセージを出力するには、[グラフィックデザイナ]エディタの プロセス画像を設定します。

9.7 プロセス画面の設定(メッセージシステム)

概要

このセクションでは、メッセージの出力を表示するプロセス画像の設定方法を説明します。

原理

プロセス画像を[グラフィックデザイナ]エディタで設定します。

このため、以下のオブジェクトを使用します。

WinCC AlarmControl

[WinCC AlarmControl]オブジェクトは、アラームウィンドウを作成するために使用しま す。メッセージは、アラームウィンドウのテーブルに表示されます。 出力はランタイムで行なわれます。

- スライダオブジェクト
 「Quick_Start」プロジェクトで、スライダオブジェクトを使用して、アナログ値を内部
 タグ「Tank_Level」に転送できるようにします。
 転送した値が定義済みの制限値に違反している場合、対応するアナログアラームがト
 リガされます。
- I/O フィールド [I/O フィールド]オブジェクトを、「Quick_Start」プロジェクトの内部タグ 「Inflow_Valve」とリンクします。
 I/O フィールドに2進数値を入力します。これらの値は、ランタイムでタ グ"Inflow_Valve"に割り当てられます。
 特定のビットがタグ値に設定されている場合、対応するディスクリートアラームがト リガされます。

9.7.1 アラームメッセージウィンドウの設定

概要

次のステップは、メッセージウィンドウの設定方法を示しています。 [グラフィックデザイナ]エディタでメッセージウィンドウを設定します。

このため、新しいプロセス画像を作成します。"WinCC AlarmControl"オブジェクトをプロ セス画像に挿入します。このオブジェクトは、メッセージの表示のために事前に設定され ています。

WinCC AlarmControl のプロパティを使用して、メッセージウィンドウの列としてどのメ ッセージブロックを表示するかを定義します。

ランタイムで、メッセージはこれらのメッセージブロックで構成されます。

手順

- 1. 「Alarm_Logging.Pdl」という名前の新しいプロセス画像を作成し、[グラフィックデザイナ] エディタで開きます。
- 2. "WinCC AlarmControl"オブジェクトをプロセス画像に挿入します。



[WinCC AlarmControl プロパティ]ダイアログが開きます。

- オペレータメッセージ ツールバー ステータスパー オンラインコンフィグレーション エクスホペート 全般 パラメータ 効果 選択 フォント メッセージフロック メッセーシリスト 統計リスト ウルトウー プロパティー ウィンドウタイトル: 開いたとやチャについてのアクティフリスト 1-通常 0 - メッセージリスト \sim リーバーの選択 テキスト localhost; 63 Wastewater ☑移動可能 ☑ すべてのサーバー ☑加−ズ可能 ☑ メッセージ色の表示 ☑ サイス 変更可能 ☑自動ス加ーリング スタイル: デフォルトのソート: プロジェ外設定 \sim \sim 0 - 昇順 行ス加ールバー: 表示わジョン・ 1-必要時 \sim メッセージの表示: 列スケロールバー: 1-表示メッセージのみ \sim 1-必要時 \sim 長期アーカイフリストのページ □ペーシングの有効化 現在の印刷ジョブの表示 ページ当たりのメッセージ: 50 AlarmControl - Table 2 - አረዋ- አ -タフルクリック時にアクション 2-プロジェクト設定 \sim 4-列依存 \sim
- 3. メッセージウィンドウの名前を指定します。

[メッセージブロック]タブをクリックします。
 [プロジェクト設定の適用]設定が有効になっていることを確認します。

これにより、アラームロギングからのメッセージブロックの設定が適用されます。

| オヘット | ータメッセージ | ツールハ | ř= | ステーク | れい | _ | オンライン | ルンフィ | ゲレーション | | エクスホペート |
|------------|---|-----------------------|----|------|-----|-------------|-----------------|-------------|---------------------|------------|---------|
| 全般 | パラメータ | 効果 | 選択 | 7: | 47F | 10 | セーシブロ | לפו | メッセーシリ | 자 | 統計リスト |
| 使用 | 可能なタッセー | ንን ከッ <mark>ን:</mark> | | | _ | <u>わり</u> | 在外名: | { | | | |
| | 日付 | | | | × | Dat | e | | | | |
| | 時刻 | | | - 1 | | 5^%I | | | | <u>7</u> - | キストID: |
| 티브 | 維続時間 | a + 88 | | - 1 | | 日作 | 4 | | | 1 | 14 |
| | 夏時間/標準 フテーカフ | 時間 | | | | 配置 | : | | 長さ <mark>(</mark>) | 文字 | 銰): |
| | ヘノ ノヘ 確認ステータス | | | | | 0 - 2 | 左揃え | \sim | 8 | | |
| | 番号 | | | | | | ッカーン・白、 | の表示 | | | |
| | クラス | | | | | <u>一</u> // | アビン 二 健祝可能が | いるの1 な確据 | 、 ኴንፈይወታን | | |
| | タイフ° | | | | | | ヨバー146/ 記成オン | 0.481/1 | ., I) H) | | |
| | コントローフ /CPU あか [、] | 田石 | | | | | mi)任 | | | | |
| | ~/ = | | | | | 0 - 1 +: | !≂÷= =t | | | | × |
| 30 | | | | | | е. Ц | い 間の形式 | t: | | | |
| | ゴテキストとして | のコンテンツ | | | | | | - | | | ~ |
| |]7イコンとしてく | カコンテンツ | | | | E | 付の形式 | ť: | | | |
| |] テキストとして(| ወቃイトル | | | | (| dd/MM/yy | / | | | \sim |
| Г | ารสวบรุปสด | ወቃለኮル | | | | | 1日付の | 表示 | | | |
| | _,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,, | | | | | | | | | | |
| | \frown | | | | | | | | | | |
| | | | | | | | | | | | |
| <u>ہ</u> ے | 2 | | | | | | | | | | |
| Ľ | 」プロジェクト設 | 定の適用 | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |

 [メッセージリスト]タブをクリックします。 ユーザーテキストブロック[メッセージテキスト]および[エラーのポイント]を[選択されて いるメッセージブロック]フィールドに移動します。

| オペシータメッセー | -ジー ウールパ | ご ス | テータスパー | オンラインコンフ | ィクレーション | エクスホペート |
|----------------------------|-------------|-----|---------------|--|------------------|------------|
| 全般 パラ: | ∽> 効果 | 選択 | フォント | メッセーシブ泊ック | メッセーシリスト | 統計リスト |
| 使用可能な 使用可能な I-5-のポイン | *yt->>71y7: | | 選手 | パピンクロック 沢されたメッセージ 付 刻 号 セージテキスト | <u>ንጉ</u> ጋ እንግ | |
| - yト 編 | 集 | ·選折 | > 【 編集… | < << | 上へ 固定選択 編集 | <u>۲</u> ۸ |

ユーザーテキストブロック[メッセージテキスト]および[エラーのポイント]は、メッセージ ウィンドウに表示されます。

6. [OK]を押してダイアログを閉じます。 必要な場合に、メッセージウィンドウを拡大します。

結果

[水処理]メッセージウィンドウが設定されました。

「Quick_Start」プロジェクトに対して設定したメッセージはランタイム中にこのウィンド ウで表示されます。

メッセージがトリガされるかどうかは、内部タグ"Tank_Level"と"Inflow_Valve"の値により 異なります。メッセージの表示色は、メッセージステータスに応じて変化します。

メッセージの内容は、メッセージウィンドウの以下のメッセージブロックによって構成されます。

- システムブロック:日付、時刻および番号
- ユーザーテキストブロック:メッセージテキストおよびエラーのポイント

| 🔒 w | astewater Trea | atment | | | | 8 |
|-----|----------------|------------|------------|--------------|-----------|----------|
| 0 | 3 | | <u>a</u> 🔢 | | 音 🖻 🚰 📥 | |
| | 日付 | 時刻 | 番号 | メッセージテキスト | エラーの木 | <u> </u> |
| 91 | 04/10/18 | 04:43:48 4 | 91 | TEXT | TEXT | |
| 92 | 04/10/18 | 04:43:49 年 | 92 | TEXT | TEXT | |
| 93 | 04/10/18 | 04:43:50 午 | 93 | TEXT | TEXT | |
| 94 | 04/10/18 | 04:43:51 年 | 94 | TEXT | TEXT | |
| 95 | 04/10/18 | 04:43:52 4 | 95 | TEXT | TEXT | |
| 96 | 04/10/18 | 04:43:53 年 | 96 | TEXT | TEXT | |
| 97 | 04/10/18 | 04:43:54 4 | 97 | TEXT | TEXT | |
| 98 | 04/10/18 | 04:43:55 4 | 98 | TEXT | TEXT | |
| 99 | 04/10/18 | 04:43:56 午 | 99 | TEXT | TEXT | |
| 100 | 04/10/18 | 04:43:57 4 | 100 | TEXT | TEXT | |
| < | | | | | | > |
| 準備到 | 宅 保留 - | 中:0 未確認: | 0 非表 | 示: 0 기자: 100 | 🔲 🖗 🎦 🐘 🗗 | 16:42:24 |

次のステップ

スライダオブジェクトを設定して、内部タグ「Tank_Level」にアナログ値を供給できるようにします。

9.7.2 スライダオブジェクトの挿入とダイナミック化

概要

次のステップは、スライダオブジェクトを挿入し、ダイナミック化する方法を示していま す。

スライダオブジェクトは、タグ値の表示や変更のために使用します。スライダオブジェクトのプロセスタグへの接続によって、ASを制御できます。

「Quick_Start」プロジェクトで、「Alarm_Logging.Pdl」プロセス画像にスライダオブジェ クトを挿入します。

内部タグ「Tank_Level」への接続により、スライダオブジェクトをダイナミック化します。

ランタイムにスライダオブジェクトを使用する場合、内部タグ"Tank_Level"に値が割り当て られます。割り当てられた値が定義済みの制限値のいずれかに違反している場合、対応す るアナログアラームがアラームウィンドウに表示されます。

「Quick_Start」プロジェクトに対して、スライダオブジェクトのプリセットプロパティを 変更します。[オブジェクトプロパティ]ウィンドウで、スライダオブジェクトの新しい名前 を入力し、その高さを定義します。

必要条件

- 「Alarm_Logging.Pdl」プロセス画像が開いている。
- 内部タグ"Tank_Level"が作成されている。

手順

- 標準 **→** ¤ × Alarm_Logging.Pdl X • ~ 23 → → 標準オフジェクト 🚋 🔜 スマートオフジェクト └──**─** ウィントウオフジェクト Point of error ----- ホタン TEXT 習二 チェックボックス TEXT TEXT ◎= オフ°ショングルーフ° し 丸いホタン TEXT 麦 スライダオブジェクト TEXT ーフォフシェクト TEXT 多角形チューブ TEXT - T-部品 📥 タンブルT-部品 TEXT チューフ・ヘッント > 🔬 🗐 3:57:51 PM
- 1. 「Alarm_Logging.Pdl」プロセス画像に、スライダオブジェクトを挿入します。

[スライダ設定]ダイアログが開きます。

2. スライダオブジェクトを内部タグ"Tank_Level"にリンクさせます。

| スライダコンフィグレー | ション | ? | × |
|-------------|----------------|----|----------|
| 9D°: | Tank_Level | 0- | 1 |
| 更新: | 2秒 | | |
| 限界: | | | |
| 最大値: | 100 | | |
| 最小値: | 0 | | |
| ステッフ°: | 10 | | |
| 方向: | | | |
| ○横に並べて | 表示 ● 縦に のK | | R TUN |

3. [オブジェクトプロパティ]ウィンドウに切り替えます。 スライダオブジェクトの名前として、"Water_Tank"を入力します。

| オフѷᠴクトプロハ℉ィ | | | • | џх |
|-----------------------------------|----------------|-------------|---------|----|
| フロハウィ イベント テキスト アニメーション | | | | |
| スライダオフジェント 居性 | スタティック | ダイナミック | サイクルの更新 | 間 |
| | 环入力 | | × | |
| | 7% | | | |
| | Vater_Tank | 3 | | |
| | | | | |
| → 効果 > | ОК | キャンセ | al 🔤 | |
| | | | | > |
| オフジェクトフロハディ 🛥 タグ = 出力ウィントウ 🍋 ラ | (ブラリ 🏳 フィブラリ 🍕 | く ダイナミックウィナ | ザード | |

4. スライダオブジェクトの高さとして、[400]と定義します。

| オブジュクトフ⁰ロハ⁰ティ | ▼ ₽ | × |
|------------------------------------|---|---|
| 7°µл°ティ イベント テキスト ア: | ニメーション | |
| スライダオフ*ジェクト | 属性 スタティック ダイナミック サイクルの更新 健 位置X 値の入力 X □ 位置Y 高さ □ □ 幅 高さ ● □ 部さ ● ● □ 2 ○ ● ● 100 ● ● ● 100 ● ● ● 2 ○ ● ● 2 ○ ● ● | - |
| ┃ ■ オフジェクトフロハディ ~ ■ タグ | ・ < | |

結果

スライダオブジェクト"Water_Tank"が挿入され、ダイナミック化されています。

スライダオブジェクトのダイナミック化により、内部タグ"Tank_Level"への値の転送が可能 になります。このタグに対して制限値モニタリングが設定されているので、設定した値が 違反している場合は対応するメッセージがトリガされます。

次のステップ

スライダオブジェクトで値を設定できるようにするために、スケールを挿入します。

9.7.3 スケールの挿入

概要

次のステップは、[グラフィックデザイナ]エディタのライブラリからスケールを挿入する 方法を示しています。

「Quick_Start」プロジェクトで、「Alarm_Logging.Pdl」プロセス画像にスケールを挿入します。

このスケールによって、スライダオブジェクト「Water_Tank」が想定できる値を表示しま す。スケール上のラインは、スライダオブジェクトの操作ステップに対応します。

必要条件

- 「Alarm_Logging.Pdl」プロセス画像が開いている。
- スライダオブジェクト"Water_Tank"が挿入されている。

手順

- 1. グローバルライブラリに切り替えます。
- 2. スケール"02"を挿入します。

| Ala | rm_Logging.Pdl | x | | | • |
|------------|----------------|----------------------|--------------------------|--------|-----------|
| · · | | | | | • • • • |
| 1 | Wastewater Tre | atment | 23 | | 3 |
| | | | | | 5) |
| | | | 🖗 🗹 🖾 🔁 🖽 🏪 😂 | | |
| | 日付 | 時刻 番号 | メッセージテキスト | | |
| 82 | 04/10/18 | 05:47:31 午 82 | TEXT | . / | |
| 83 | 04/10/18 | 05:47:32 午 83 | TEXT | | |
| 84 | 04/10/18 | 05:47:33 午 84 | TEXT | / | • • • • • |
| 85 | 04/10/18 | 05:47:34 午 85 | TEXT | | |
| 86 | 04/10/18 | 05:47:35 午 86 | TEXT | | |
| 87 | 04/10/18 | 05:47:36 午 87 | TEXT | | |
| 88 | 04/10/18 | 05:47:37 午 88 | TEXT | | |
| 89 | 04/10/18 | 05:47:38 午 89 | TEXT | | |
| 90 | 04/10/18 | 05:47:39 午 90 | TEXT | | |
| 91 | 04/10/18 | 05:47:40 午 91 | TEXT | | |
| 92 | 04/10/18 | 05:47:41 年 92 | TEXT | 1 1111 | |
| 93 | 04/10/18 | 05:47:42 午 93 | TEXT | | |
| 94 | 04/10/18 | 05:47:43 7 94 | TEXT | | |
| 95 | 04/10/18 | 05:47:44 年 95 | TEXT | | |
| 96 | 04/10/18 | 05:47:45 午 96 | TEXT | | |
| 9/ | 04/10/18 | 05:47:46 7 97 | | | |
| 98 | 04/10/18 | 05:47:47 7 98 | | 1 1111 | |
| 99 | 04/10/18 | 05:47:48 + 99 | | | |
| 100 | 04/10/18 | 05:47:49 + 100 | | | |
| 保留 | 留中:0 未確認 | : 0 非表示: 0 リスト: 1 | .00 🔲 🗑 🎦 强 🖻 17:46:11 🔡 | | |
| | | | | | |
| | | | | | |
| < | | | | | > |
| ライフ・ラリ | | | 3 | | ▼ ‡ X |
| | K B B X | ₽_ :- ::: ::: ::: 60 | | | |
| ÷ | Displays | _ | ▲ 名前 | サイズ | 最 ^ |
| +. | Displays | | 1 01 | 0000 | 20.0 |
| | Meters | | 1 01 / | 9882 | 29.0 |
| | Cealing | | | 10018 | 29.0 |
| H . | Scaling - | | 03 | 9984 | 29.(|
| . | I ext Fields | | | 10110 | 29.0 4 |
| | indows | | | | > |
| 1173 | νʹェクトフ°ロハ°ティ 🔫 | ■ タゲ =出力ウィンドウ | | P | |

スケールは、プロセス画像「Alarm_Logging.Pdl」に表示されます。

- Alarm_Logging.Pdl X • ~ ~ 23 100 न 占 介 Numme Meldetext Störort hrzeit 4:53:54 PM 91 TEXT TEXT 80. 4:53:55 PN 92 TEXT TEXT 4:53:57 PN 94 TEXT TEXT 4:53:58 PN 95 TEXT TEXT 4:53:59 PM 96 TEXT TEXT TEXT 60 TEX1 c 4:54:02 PN 99 4:54:03 PN 100 TEXT TEXT 40 > 未確認:0 非表示:0 リスト:100 🗊 😰 🎦 强 💷 16:52:27 v < > オフジジェクトフロハウィ **▼** ₽ × 7°D/1°71 11 テキスト アニメーション グルーフッヒ ~ 属性 スタティック ダイナミック サイクルの更新 ---形状 位置X 値の入力 × 色 位置Y スタイル AI 幅 高さ フォント 3 高さ 2 400 点滅 回転参照X その他 回転参照Y 定義されているユーザー1 4やンセル OK 1 回転角度 定義されているユーザー2 🗸 > < < ■オフジェクトフロパティ ④ ダグ 国出力ウィントウ とうインデリ といいのうインデリ ペダダイナシックウィザートご
- 3. [オブジェクトプロパティ]に切り替えます。 スケールの高さとして、[400]と定義します。

- オフジェクトフロパティ **→** ₽ × プロノパティ イベント テキスト アニメーション ケルーフット \land 属性 スタティック ダイナミック サイクルの更業 🔨 形状 フォント Arial Q 色 フォントサイズ 12 値の入力 \times スタイル 00 太字 フォント 斜体 いいき フォントサイズ 点滅 下線 いい 16 その他 テキストの方向 横(ご 定義された .#*__1 <mark>X</mark>整列 中央 定義されているユーザー2 🗸 OK キャンセル \/#∀⊼II ≪ 4 > < >
- 4. スケールのフォントサイズとして、"16"を設定します。

5. スケールとスライダオブジェクトを同じ水平線に並べます。

| 0 | 3 | 3 G F | 11 🔅 🗟 💹 😤 | 🚰 🚰 🚢 🍦 🐼 📕 | 111 |
|----|----------|----------------|------------------|----------------|----------------|
| | Date | Time Nu | mbe Message text | Point of error | 100 |
| 3 | 11/10/18 | | | TEXT | and a |
| 4 | 11/10/18 | 05:05:19 PF 84 | TEXT | TEXT | 100 |
| 15 | 11/10/18 | 05:05:20 PI 85 | TEXT | TEXT | 1.4.4.4 |
| 16 | 11/10/18 | 05:05:21 PF 86 | TEXT | TEXT | 100 |
| 17 | 11/10/18 | 05:05:22 Pt 87 | TEXT | TEXT | 0.000 |
| 18 | 11/10/18 | 05:05:23 Pf 88 | TEXT | TEXT | and the second |
| 19 | 31/10/18 | | | 1930 | 00000 |
| 0 | 11/10/18 | 05:05:25 PE90 | TEXT | TEXT | 1000 |
| 1 | 11/10/18 | 05:05:26 PF 91 | TEXT | TEXT | 1000 |
| 2 | 11/10/18 | 05:05:27 Pt 92 | TEXT | TEXT | 1000 |
| 3 | 11/10/18 | 05:05:28 Pf 93 | TEXT | TEXT | 1000 |
| 4 | 11/10/18 | 05:05:29 PF 94 | TEXT | TEXT | 1000 |
| 5 | 11/10/18 | 05:05:30 Pt 95 | TEXT | TEXT | |
| 6 | 11/10/18 | 05:05:31 PM 96 | TEXT | TEXT | 1000 |
| 7 | 11/10/18 | 05:05:32 Pt 97 | TEXT | TIXI | |
| 8 | 11/10/18 | 05:05:33 PM 98 | COM TEXTS | TEXT | 1000 |
| 9 | 11/10/18 | 05:05:34 PE 99 | TEXT | TEXT | 1000 |
| 00 | 11/10/16 | 05:05:35 Ph 10 | 0 TEXT | TEXT | Concerned in |

結果

スケールが、「Alarm_Logging.Pdl」プロセス画像に挿入されています。

ランタイムで、このスケールによってスライダオブジェクト「Water_Tank」の値を設定します。

次のステップ

I/O フィールドを挿入して、内部タグ「Inflow_Valve」を 2 進数値と共に供給できるよう にします。

概要

次のステップは、I/O フィールドを挿入し、ダイナミック化する方法を示しています。

「Quick_Start」プロジェクトで、「Alarm_Logging.Pdl」プロセス画像に I/O フィールドを挿入します。

内部タグ「Inflow_Valve」への接続により、I/O フィールドをダイナミック化します。

このタグにステータスを保存することにより、I/O フィールドの値のバイナリ出力形式を定義します。

I/O フィールドは、ランタイム中にバイナリ値を内部[Inflow_Valve]タグに転送するために 使用されます。

必要条件

- 「Alarm_Logging.Pdl」プロセス画像が開いている。
- 内部タグ"Inflow_Valve"が作成されている。

手順

- Standard ▼ ₽ X Alarm_Logging.Pdl X • Selection ~ ~ 23 🖶 😹 Standard Objects TOE 🛓 📑 Smart Objects r 🛉 1 Picture Window Message tex 80 OCX Control TEXT OLE OLE object TEXT 0,000 0.2 I/O Field TEXT 60 🚛 Bar 🔝 Graphic Object TEXT 📲 Status Display TEXT TEXT 40 Text List TEXT A Multiple row text Combo Box TEXT List Box TEXT 20 🖶 Faceplate instance > .NET Control 👌 📴 12:44:23 PM 🔡 WPF Control 0 3D-Bar < > 🔊 Stan... 🔝 Contr... 🚟 Styles 🛛 🎄 Proce...
- 1. [I/O フィールド]スマートオブジェクトを挿入するには、[標準]ウィンドウでオブジェクト をクリックしてから、プロセス画像をクリックします。

[I/O フィールド設定]ダイアログが開きます。

2. I/O フィールドを内部タグ"Inflow_Valve"にリンクします。

| IO -フィールト・コンフィ | バレーション ? × |
|------------------------|-------------------|
| \$7°: | Inflow_Valve |
| 更新: | 2秒 |
| フィールドタイフ⁰: | |
| ОДЛ | |
| ○出力 | |
| ● IO7ィールト [×] | |
| フォントサイス | 12 |
| 7ォント | Arial |
| 色 | OK THUEN |
9.7 プロセス画面の設定(メッセージシステム)

3. [オブジェクトプロパティ]ウィンドウに切り替えます。 I/O フィールドの名前として、"バルブ"と入力します。



- 4. I/O フィールドの以下のフォントプロパティを定義します。
 - フォントサイズ:28
 - 太字:はい
 - X 整列:右揃え
 - Y 整列:中央揃え



5. I/O フィールドの出力フォーマットとして、[2進数]を定義します。

| Object Properties | | | | | ▼ Į X |
|---|--|--|---------------------------------------|---------------|-------|
| Properties Events Texts | Animation | | | | |
| I/O Field Geometry Colors Styles Font Flashing Miscellaneous Limits Output/Input Effects | Attribute Field Type Input Value Output Value Data Format Output Format Apply on Full Apply on Exit Clear on New Input Clear on Invalid Input | Static I/O Field 0.000000 0 Binary Binary Decimal String Hexadecimal Date/Time Date/Time Date/Time (local Control characte | Dynamic Dynamic Dynamic alve | Update 2 s | |

9.7 プロセス画面の設定(メッセージシステム)

6. [出力フォーマット]プロパティに対して、位置の数を6から8へ増やします。"11111111"。



7. I/O フィールドを拡大し、「Alarm_Logging.Pdl」プロセス画像を保存します。
 8. [グラフィックデザイナ]を閉じます。

結果

1/0 フィールド"バルブ"が設定されています。

設定済み I/O フィールドに 2 進数値を入力します。これらの値は、内部タグ「Inflow Valve」に転送されます。

タグ値で特定のビットが設定されると、対応するディスクリートアラームがトリガされ、 メッセージウィンドウに表示されます。例えば、2番目のビットがタグ値で設定されると、 ディスクリートアラーム"Valve closed"がトリガされます。

9.7 プロセス画面の設定(メッセージシステム)

| / | Alarm_ | Logging.Pdl | × | | |
|----|--------|--------------|---------------------------------------|---------------------------|---|
| | | | | | |
| | 🔒 Wa | stewater Tre | atment | 23 | 100 |
| | | | | | |
| | | | | 🏷 🗹 🕅 🧮 🗒 🔛 🎒 🤤] | · · · · · · · · · · · · · · · · · · · |
| | | 日付 | 時刻 番号 | メッセージテキスト | |
| | 82 | 08/10/18 | 05:07:01 午 82 | TEXT | · · · · · · · · · · · · · · · · · · · |
| | 83 | 08/10/18 | 05:07:02 午 83 | TEXT | 80 |
| | 84 | 08/10/18 | 05:07:03 午 84 | TEXT | |
| | 85 | 08/10/18 | 05:07:04 午 85 | TEXT | |
| | 86 | 08/10/18 | 05:07:05 午 86 | TEXT | |
| | 87 | 08/10/18 | 05:07:06 午 87 | TEXT | 60 |
| | 88 | 08/10/18 | 05:07:07 午 88 | TEXT | |
| ١. | 89 | 08/10/18 | | TEXT | |
| Ľ | 90 | 08/10/18 | 05:07:09 午 90 | TEXT | |
| | 91 | 08/10/18 | 05:07:10 午 91 | TEXT | |
| | 92 | 08/10/18 | 05:07:11 午 92 | TEXT | 10 |
| | 93 | 08/10/18 | 05:07:12 午 93 | TEXT | · · · · · · · · · · · · · · · · · · · |
| | 94 | 08/10/18 | 05:07:13 午 94 | TEXT | |
| | 95 | 08/10/18 | 05:07:14 午 95 | TEXT | |
| | 96 | 08/10/18 | 05:07:15 午 96 | TEXT | |
| | 97 | 08/10/18 | 05:07:16 午 97 | TEXT | 20 |
| | 98 | 08/10/18 | 05:07:17 午 98 | TEXT | |
| | 99 | 08/10/18 | 05:07:18 午 99 | TEXT | |
| | 100 | 08/10/18 | 05:07:19 午 100 | TEXT | |
| | 保留由 | ・0 未確認 | ・0 非表示・0 Uzk・ | 100 🔲 🖓 🛅 🚱 🖓 17:05:44 | · · · · · · · · · · · · · · · · · · · |
| | | | · · · · · · · · · · · · · · · · · · · | | v · · · · · · · · · · · · · · · · · · · |
| 10 | | | | ······ | |

次のステップ

- 「Alarm_Logging.Pdl」ページをユーザー定義メニューに追加します。
- WinCC Runtime のプロパティを定義します。

9.8 ユーザー定義メニューのカスタマイズ(メッセージシステム)

9.8 ユーザー定義メニューのカスタマイズ(メッセージシステム)

概要

次のステップは、[アラームロギング]メニュー項目をカスタマイズされた[画像変更]メニ ューに追加する方法を示しています。

"アラームロギング"メニューエントリをプロシージャ"ActivatePicture(ByVal PictureName)"と接続させます。

変更するプロセス画像の名前を、[ユーザーデータ]フィールドに入力します。

必要条件

- プロシージャ"ActivatePicture(ByVal PictureName)"が作成されている。
- ユーザ定義の[画像変更]メニューが「START.PdI」、「SAMPLE.PdI」、「Tag_Logging.PdI」のプロセス画像用に作成されていること。

手順

- 1. 保存された設定[ActivatePicture.mtl]を[メニューとツールバー]エディタで開きます。
- 2. [アラームロギング]メニューコマンドを作成します。

| 👕 Menus and toolbars [Acti | vatel | Picture] - WinCC C | onfiguration Stu | dio | | | | 1000 | | Х |
|---|--------------|---------------------|------------------|-----------|-----|------|--------------|-----------|--------|-----|
| <u>F</u> ile <u>E</u> dit <u>V</u> iew Too <u>l</u> s | <u>H</u> elp | | | | | | | | | |
| Menus and tool « | 1 | M Find | | Q | • | Ï | Propertie | s - Men | u item | » |
| 🖃 📑 Menus 🔷 | 1 | Name | Text | Active 1 | ^ | Ξ | Selection | | | ^ |
| 🚊 📲 Picture Change | 1 | Start | Start | V | | 10 | Object type | Menu iter | m | |
| 1 Start | 2 | Sample | Sample | V | | 1.5 | Object name | Alarm_Lo | gging | _ |
| Sample | 3 | Tag Logging | Tag Logging | V | | | General | | | |
| Tag Loggin | 4 | Alarm Logging | Alarm Loggin | | | | Name | Alarm_Lo | gging | |
| | 5 | * | | | | 19 | Text | Alarm_Lo | gging | |
| | 6 | | | | | | Activo | N | | |
| loolbars V | 7 | 8 | | | | | | | | |
| | 8 | () () | <u> </u> | | | | | | | |
| 💷 🛼 🂢 🖶 💷 🗶 🔹 | 14 | 🕩 🕅 🛛 Menu ite | ms I < | > | • | | | | | |
| Ready NUM | | English (Un | ited States) Ta | able: 4 M | enu | iten | ns 100 % 🤅 |) | 0 | ÷., |

9.8 ユーザー定義メニューのカスタマイズ(メッセージシステム)

3. 「Alarm_Logging.pdl」画像に変更するために、[アラームロギング]メニューコマンドを設定 します。

| lenus and tool « | 1 | M Find | | Q | • | | Properties | - Menu item | >> |
|--------------------|----|----------------|---------------|----------|---|---|----------------------|---------------|----|
| 👕 Menus | | Name | Text | Active | 1 | Đ | Selection | | 1 |
| | 1 | Start | | V | | Ξ | General | | |
| Start | 2 | Sample | | V | | | Name | Alarm_Logging | |
| Sample | 3 | Tag Logging | | | | | Text | | |
| Tag Logging | 4 | Alarm_Logging | | V | | | Active | V | |
| Function selection | | | | | × | | Visible Separator | | |
| 🔠 Project Modules | ~ | 4 | Prototype | | | | Authorizat 1 | 1 | |
| MenuToolbars_C | om | A ctiupte Dict | Sub Activate | Dist | | | User data | Alarm_Logging | |
| | | | , SUD ALINALE | Plue | | | C main h | | |

4. [メニューとツールバー]エディタで変更内容を保存します。

5. "メニューとツールバー"エディタを閉じます。

結果

[アラームロギング]メニュー項目を[画像変更]メニューに追加しました。

ランタイムでメニューエントリを使用して、「START.Pdl」、「SAMPLE.Pdl」、

「Tag_Logging.Pdl」および「Alarm_Logging.Pdl」のプロセス画像に切り替えられるように します。 9.9 ランタイムプロパティの定義(メッセージシステム)

9.9 ランタイムプロパティの定義(メッセージシステム)

概要

次のステップは、WinCC Runtime のプロパティの定義方法を示しています。

このセクションでは、プロジェクトの起動時にアラームロギングランタイムが実行される ように WinCC Runtime をセットアップします。

「Alarm_Logging.Pdl」プロセス画像を、ランタイムウィンドウの開始画像として設定します。

必要条件

- "Quick_Start"プロジェクトが開いている。
- 「Alarm_Logging.Pdl」プロセス画像が作成されている。

9.9 ランタイムプロパティの定義(メッセージシステム)

手順

1. [コンピュータ]エディタを開きます。

| WinCCExplorer - C:\Users\Public\Documents\Siemens\Win | nCCProjects\Quic | k_Start\Quick_Sta | rt.MCP | – 🗆 X |
|--|---|-------------------|--------------|--------------|
| ファイル(F) 編集(E) ビュー(V) ツール(T) ヘルプ(H) | | | | |
| ☆ [] [] [] [] [] [] [] [] [] [] [] [] [] | | | | ≣ २ ? |
| Quick_Start | $\boldsymbol{\leftarrow} \rightarrow \boldsymbol{\uparrow}$ | \R | 9 | ~ 7 |
| יאנ 📰 אני 1-9 | 名前 | | 917° | |
| □ \$95°管理 | wcr. | DOCU-02 | <u> </u> | |
| | | | 切り取り(T) | |
| | | |]t°-(C) | |
| | | | 15月1月(h) | |
| | | | 削除(D) | |
| 1.0 テキストおよびク、ラフィックリスト 2.4 | | | ፓ° ዐ시° ティ(O) | ~ |
| ٢٩-۵0 [±] ۲۶−۵0 [±] | | | (| |
| \$90°04">20" | | | | |
| 正 レ ^ル ・トテ [、] サ [、] イナ | | | | |
| | | | | |
| Ouick Start\コンド° 1-タ\ | | 選択した1オブ | 5 101 | ライヤンスされてい |

[コンピュータ]エディタが WinCC Configuration Studio で開きます。

2. ナビゲーションエリアでコンピュータ名を選択します。

[WinCC Runtime の起動時のプロセス]タブで[アラームロギングランタイム]アプリケーションを有効にします。



3. [プロパティ-コンピュータ]エリアの[グラフィック]で、「Alarm_Logging.Pdl」プロセス画像 をランタイムウィンドウの初期画像として設定します。 

結果

WinCC Runtime のプロパティが定義されています。

「Quick_Start」プロジェクトを有効にするときに、タグロギングランタイムが実行され、 「Alarm_Logging.Pdl」プロセス画像が表示されます。

次のステップ

「Quick_Start」プロジェクトを有効にします。

9.10 プロジェクトの有効化(アラームロギング)

概要

次のステップは、「Quick_Start」プロジェクトを有効にする方法や、ランタイムでの「Alarm Logging.Pdl」プロセス画像の操作方法を示しています。

"Quick Start"プロジェクトを有効にすると、WinCC Runtime が開始します。

開始画像として、「Alarm Logging.Pdl」プロセス画像が表示されます。

ランタイムのメッセージウィンドウ

ランタイムで、メッセージウィンドウを操作、監視します。ツールバーのボタンによって メッセージウィンドウを使用します。

●ボタンによって、メッセージリストを表示できます。メッセージリストは、現在保留中のメッセージを含みます。

メッセージウィンドウのメッセージの表示色は、メッセージステータスによって変化します。

必要条件

- "Quick Start"プロジェクトが開いている。
- ランタイムプロパティが定義されている。

手順

 WinCC エクスプローラの ツールバーボタンを使って"Quick_Start"プロジェクトを有効に します。
 WinCC Runtime が起動します。

「Alarm_Logging.Pdl」プロセス画像がランタイムウィンドウに表示されます。

2. [Water Tank] スライダオブジェクトのコントローラを動かします。

| Picture Change I Wastewater Treatment I Image: Control of the state of | KinCC-Runtime - | -15 | |
|---|---|-----|---|
| 1 Wastewater Treatment Image: Contract of the state of | Picture Change | | |
| • Watewater Treatment • • • • • • • • • • • • • • • • • • • | | | |
| | Wastewater Treatment Image: Constraint of the state of the sta | | 0 |
| | < | | × |

内部タグ「Tank_Level」に、スライダ位置に対応する値が割り付けられます。 この値が設定されている下限値(10)を下回った場合、アナログメッセージ「下限値」がト

上限値(90)が違反される場合、アナログアラーム「上限値」がトリガされます。

リガされます。

| K WinCC-Runtime - | | - 0 | × |
|--|----|-----|----|
| Picture Change | | | |
| | | | |
| Wastewater Treatment | 10 | 00 | ^ |
| 日付 時刻/ 番号 メッセージデキスト 1 08/10 06:24 取界90.00000を超えました: 0.000000 2 08/10 06:24 5 取界10.00000を下回りました: 20.000000 3 08/10 06:32 1 Valve_open | 8 | 0 | |
| 4 5 6 7 8 | 6 | 0 | |
| 9 10 11 12 | | AI | 00 |
| 13 14 15 16 | 2 | 0 | |
| 17 18 く 後にに、「日本語」では、 18・3 18・3 18・3 18・3 18・3 18・3 18・3 18・3 | | | |
| | (|) | × |

3. [バルブ] I/O フィールドに値[100]を入力し、<Return>で入力内容を確定します。

タグ値の2番目のビットが設定されます。 ディスクリートアラーム"Valve open"が表示されます。

- I/O フィールドに値[1000]を入力し、<Return>で入力内容を確定します。 タグ値の3番目のビットが設定されます。 ディスクリートアラーム"Valve closed"が表示されます。
- 5. I/O フィールドに値[10000]を入力し、<Return>で入力内容を確定します。 タグ値の4番目のビットが設定されます。 ディスクリートアラーム"Valve inop"が表示されます。

- メッセージウィンドウのツールバーにある
 ボタンをクリックし、[自動スクロール]機能を 無効にします。
 このようにして、メッセージを選択することができます。
- 7. アナログアラーム[上限値]をクリックし、アナログアラームを確認します。 メッセージステータスの表示色は変化します。

| ✓ WinCC-Runtime - | 1 | | × |
|---|---|------|----------|
| Picture Change | _ | | |
| | | | |
| 1. Wastewater Treatment (*) <td< th=""><th>1</th><th>0000</th><th></th></td<> | 1 | 0000 | |
| | | | ~ |

結果

"Quick_Start"プロジェクトが有効化されています。

プロジェクトの開始画像として、「Alarm Logging.Pdl」プロセス画像が表示されます。

内部タグ"Tank_Level"および"Inflow_Valve"が、手動入力による値と共に提供されます。これらのタグは、アラームロギングによってモニタリングされます。

タグ値によって、対応するメッセージがトリガされ、アラームウィンドウに表示されます。

次のステップ

内部タグ[Tank_Level]および[Inflow_Valve]を値と共に自動的に供給するには、[タグシミ ュレーション]エディタを使用します。

概要

次のステップは、WinCC タグシミュレータによって「Quick_Start」プロジェクトをテス トする方法を示しています。

WinCC タグシミュレータは、ランタイムの内部タグ「Tank_Level」および「Inflow_Valve」 に値を割り付けます。

これらのタグに対してモニタリングが設定されているため、タグ値が定義されている制限に 到達するとメッセージがトリガされます。

メッセージが、メッセージウィンドウに表示されます。メッセージステータスは、異なる 表示色によってマークされます。

必要条件

ランタイムプロパティが定義されている。

手順

- 1. 「QuickStart_Simulation.sim」設定を[タグシミュレーション]エディタで開きます。
- タグ選択ダイアログを開き、内部タグ[Inflow_Valve]をタググループ[QuickStart_Tags]から 選択します。

| Tags [Simulation | 1] | | Find | | Q | • |
|-------------------|-----------------|----------------------|----------------|---------------|---------------|--------------|
| Tag name | Data type | Function | C | vola Activa C | onnection nau | A . |
| Tank Level | Unsigned 16-bit | Tag selection | | | | > |
| * | | Filter: | | | N 🗈 🦫 | 8-8- 8-8- |
| | | internal tags | ^ | Name | Туре | |
| | | B Berforma | t_lags nce | * | 7* | |
| | | ProcessHi | storian | Inflow_Valve | e Unsigne | d 16-l |
| | | TaqLoqqii | ngRt | 1 - 4 /el | Unsigne | d 16-b |
| | | 🛓 📑 Script | - | | | |
| | | 🚋 🎚 List of all stru | cture instar 🎽 | | | |
| 3 | | < | > | < | | |

3. シミュレーションタイプを選択し、シミュレーションタイプのプロパティを定義します。

| Ð | Fags simulation [Qu | ickStart_Simulatic | n] - WinCC Confi | guratio | on St | udio — 🗆 | × | < |
|--------------|-------------------------------|--------------------|------------------|--------------|-------|-----------------------------|-----|----|
| <u>F</u> ile | <u>E</u> dit <u>V</u> iew Too | ols Simulation | <u>H</u> elp | | | | | |
| Та | gs [Simula F | ind | | ب | F | Properties - Tag | ; | >> |
| | Tag name | Data type | Function | ^ | Ŧ | Selection | | ^ |
| 1 | Tank Level | Unsigned 16- | bit v Increment | 0 | | Tag | | |
| 2 | Inflow_Valve | signed 16- | bit v Random | | | Tag name Inflow_Valve (📉) | | |
| 3 | * | 0 | | | | Data type | | |
| 4 | () | | | | | Function | | |
| 5 | | Ý | | | | Cycle 3 | | |
| 6 | | 1 | | - 8. | | Active | _ | |
| 7 | | | | 1.1 | | Connection name (| | |
| 8 | | 12 | | | Ð | Sine function | | |
| 9 | | | | | Œ | Oscillation | _ | |
| 10 | | | | | | Random value 🖤 — (AI) | - | |
| 11 | | | | | | Random minimum 0 | - 1 | |
| 12 | | | | ~ | | Random maximum 100 | | |
| 12 | E H Tags | | | > . | Ð | Increment | | v |

- 4. メニューバーで[ファイル]>[保存]コマンドを使用して設定を保存します。
- 5. 》ボタンを使用して、WinCC エクスプローラで WinCC Runtime を有効にします。
- 6. [シミュレーション]>[起動]を使用して、[タグシミュレーション]エディタでシミュレーションを起動します。
- 7.「Alarm_Logging.Pdl」プロセス画像で、シミュレーション値の出力を監視します。 シミュレーション値の変化が、メッセージウィンドウにどのように影響するかを監視します。

| K WinCC-Runtime - | - 🗆 X |
|--|-------|
| Picture Change | |
| Start T | |
| Sample | |
| | |
| | |
| | |
| 日付時刻番号メッセージテキスト | |
| 1 08/10/1 08:04:03 午後 4 限界90.000000を超えました: 0.000000 | |
| 208/10/1 08:04:03 午後 2Valve_closed00 | |
| 3 08/10/108:04:04 午後 1 Valve_open | |
| 4 08/10/108:04:04 午後 3 Vave_mop | |
| 5 08/10/108:04:05 牛(夏) 5 段赤10:000000を下回りました: 0:000000 | |
| 7 60 | |
| 8 | 40044 |
| | 10011 |
| 10 | |
| 11 | |
| 12 40 | |
| 13 | |
| 14 | |
| 15 | |
| | |
| | |
| | |
| | |
| 準備元了 1米留中: 5 未確認: 5 非表示: 0 リスト: 5 20:12:39 | |
| | ~ |
| < | > |

8.「Tag_Logging.Pdl」プロセス画像に切り替えます。

9. WinCC OnlineTrendControl でのトレンドの拡大表示を確認します。



- 10.[シミュレーション]>[停止]を使用して、[タグシミュレーション]エディタでシミュレーションを停止します。
- 11.ツールバーの ボタンを使用して、WinCC Runtime を終了します。

結果

WinCC タグシミュレータによって、"Quick_Start"プロジェクトがテストされます。

テストは、モニタリングされているタグが値と共に継続的に提供されたときの、アラーム ロギングの動作を示します。

WinCC OnlineTrendControl は、[Inflow_Valve]タグの2番目のトレンドを表示します。感 嘆符をクリックすることで、ツールヒントの形式で、このタグのトリガされたメッセージを 表示します。

検索、発見、フィルタリング

10

概要

WinCC エクスプローラには、オブジェクトやテキストを検索するための便利なオプション が用意されています。

検索と発見

- 1. 検索機能を開くには、WinCC エクスプローラで次の操作を選択してください
 - プロジェクトエクスプローラのメニューバーで検索アイコンを選択します

Q

または

- スタートページで[検索]エントリを選択します。 検索語句を入力するテキストボックスを含むスタートページが表示されます。

| 検索 | | | | | | Q |
|-----|----|--------|-------|----------|----|------|
| すべて | 画面 | スクリフ°ト | V∜°-ŀ | <u> </u> | 機能 | ~#7° |

- 2. 検索したい文字列を入力します。 ワイルドカード(*?)はサポートされていません。
- 3. 検索シンボルをクリックします。 その文字列を含む結果が表示されます。
- 4. 表示される結果を絞り込むには、定義済みの領域のいずれかを選択します。

すべて 画面 スクリプト レポート タグ 機能 ヘルプ

Found 1 matches (70 milliseconds). Search instead for <u>sample</u>, <u>sampledsn</u> or <u>samplepdl</u>?



SAMPLE.Pdl

Source: C:\Users\Public\Documents\Siemens\WinCCProjects\Quick_Start\GraCS \SAMPLE.Pdl

さらに検索するための提案も表示されます。クリックするだけで、検索用のテキストボックスに直接転送できます。

- 5. エディタまたは結果の情報を開くには、結果をクリックします。これにより、たとえばグ ラフィックデザイナ、スクリプトエディタ、情報システムなどが開きます。
- 6. プロジェクトに戻るには、[戻る]アイコンをクリックします。

例:ダイナミック化に使用されるタグ

 \leftarrow

この例では、タグを使用してダイナミック化を行う画像を示します。 検索機能を開きます。

- テキストボックスにタグ名を入力します。
 名前の一部のみを入力すると、この文字列を含むすべてのタグの結果が表示されます。
- [画面]エリアを選択します。
 検索したタグが使われている画像がすべて表示されます。

| dyn AT |
|---|
| すべて 画面 スクリフ゜ト レホ゜ート タク゛ 機能 ヘルフ゜ |
| Found atches (74 milliseconds). Search instead for <u>dynam</u> , <u>dynamic</u> or <u>dynamicable</u> ? |
| dyn_limits Source: D:\SmokeTest \SmokeTest_SRV_V08020000_01020002_2024-12-19_09-30-42\GraCS \OPCUATags.PDL Object: IOFic*d Property: LimitMin |
| dyn_limits Source: D:\SmokeTest \SmokeTest_SRV_V08020000_01020002_2024-12-19_09-30-42\GraCS \S7-1500\PLC41_OnlineTags.pdl Object: TechValues_IOField_9 Property: LimitMin |

- 以下の情報が得られます。
- プロジェクトの名前
- 画面の名前
- タグがダイナミック化に使用されるオブジェクト
- 検索内のタグによってダイナミック化されたプロパティ。

タグが複数のオブジェクトで使用されている場合、それに応じて複数の結果が返されます。

 画像をダブルクリックします。 グラフィックデザイナで、画像が開きます。動的プロパティは、[オブジェクトプロパティ] ウィンドウで選択されます。

フィルタリングで検索

WinCC エクスプローラプロジェクトウィンドウには、画像やスクリプトなど、選択したエ ディタで作成され使用可能なオブジェクトが表示されます。異なるサイズのタイルやリスト 表示を選択できます。

特定のオブジェクトのみを表示するには、フィルタ機能を使用します。

| K WinCCExplorer - C:\Users\Public\Documents\Siemens\WinCCProjects\Quick_Start\Quick_Start.MCP | | | – 🗆 X |
|---|------------------------|----------------------------|-----------------------|
| File Edit View Tools Help | | | \frown |
| | | ? | |
| 🗆 🛃 C-Editor | ← → ↑ 【♪ …\Standard f | unctions\GRAPHICS get | ~ 7 |
| 🗉 🔛 Actions | Name | Туре | Last Change |
| 🗉 🚝 Standard functions | GetParentPictureWindow | getparentpicturewindow.fct | 12/15/2024 7:03:42 PM |
| 🚝 Alarm | GetParentPicture | getparentpicture.fct | 12/15/2024 7:03:42 PM |
| | GetLocalPicture | getlocalpicture.fct | 12/15/2024 7:03:42 PM |
| | 🔚 GetLinkedVariable | getlinkedvariable.fct | 12/15/2024 7:03:42 PM |
| GRAPHICS | | | |
| 🗉 🔚 Obsolete functions | | | |
| Dptions | | | |
| ProDiag | | | |
| Quick Start\Global Script\C-Editor\Standard functions\GRAPHI | CS\ 4 object(s) | Licensed mode | |

- エディタを選択します。 利用可能なオブジェクトが表示されます。
- フィルタのテキストボックスに文字列を入力します。 指定した文字列を含む名前を持つオブジェクトのみが表示されます。

用語集

[グラフィックデザイナ]エディタのライブラリ

[グラフィックデザイナ]エディタのライブラリは、グラフィックオブジェクトを保存、管理 する多用途のツールです。

シンボルライブラリでは、プロセス画像の作成に使用するカスタマイズオブジェクトおよびコントロールを管理します。

SVG ライブラリでは、SVG オブジェクトを管理します。

ライブラリは、2つの領域に分割されています:

- グローバルライブラリ
- プロジェクトライブラリ

Configuration Studio

[WinCC Configuration Studio]には、[タグ管理]、[アラームロギング]および[タグロギング] エディタなどの、WinCC の設定に必要なほとんどのエディタが含まれています。

10フィールド

I/O フィールドは、タグ値を表示、変更するために使用する入力/出力フィールドです。

WinCC AlarmControl

[WinCC AlarmControl]オブジェクトは、メッセージイベントの表示に使用される[メッセージ]ウィンドウです。

WinCC OnlineTableControl

"WinCC OnlineTableControl"オブジェクトは、プロセス値アーカイブからのプロセス値を表示するテーブルウィンドウとして使用します。

WinCC OnlineTrendControl

"WinCC OnlineTrendControl"オブジェクトは、プロセス値アーカイブからのプロセス値を表示するトレンドウィンドウとして使用します。

WinCC Runtime

WinCC Runtime は、プロセスモードでプロジェクトを実行します。 WinCC Runtime は、プロセスのオペレータ制御および監視を有効にします。

WinCC エクスプローラ

WinCC エクスプローラは、WinCC のプロジェクト設計ソフトウェアの核となっています。 WinCC エクスプローラでは、プロジェクト構造全体が表示され、プロジェクトが管理され ます。

WinCC タグシミュレータ

WinCC タグシミュレータによって、まだ開発段階であるプロジェクトをテストできます。 [タグシミュレーション]エディタを使用することで、シミュレーションを設定および有効化 することができます。

WinCC のプロジェクト設計ソフトウェア

プロジェクト設計ソフトウェアは、WinCCの一部です。

プロジェクト設計ソフトウェアの中核には、WinCC エクスプローラと WinCC Configuration Studio があります。

アーカイブサイクル

アーカイブサイクルは、プロセス値がアーカイブデータベースに保存される時間間隔です。 アーカイブサイクルは、常に設定された取得サイクルの整数倍です。

アーカイブサイクルは、WinCC Runtime の起動時またはユーザーが定義した時点に開始し ます。開始ポイントを指示すると、値の時間差アーカイブやアーカイブ負荷の分散が可能 になります。

アーカイブタグ

アーカイブする値は、アーカイブタグに保存されます。

アナログアラーム

アナログアラームは、制限値を超過しているか、達していないことを示します。

アナログアラームは、定義されている制限値を超過しているか、達していないときに、ト リガされます。

アニメーション

アニメーションでは、複数のオブジェクトプロパティのダイナミック化をタグまたはスクリプトと結合します。

これにより、複数のプロパティ変更をタグ値または値範囲に割り付けて、オブジェクトのサイズを拡大したり、同時に点滅させたりすることができます。

アラームロギング

[アラームロギング]エディタは、アラームロギングの設定コンポーネントです。 このエディタは、メッセージを設定するために使用します。

アラームロギングランタイム

アラームロギングランタイムは、アラームロギングのランタイムコンポーネントです。ア ラームロギングランタイムは、主に次のタスクを実行するために使用されます:

- 定義済みモニタリングの実行
- メッセージ出力の制御
- 承認の管理

グラフィックデザイナ

[グラフィックデザイナ]エディタは、グラフィックシステムの設定コンポーネントです。 このエディタは、プロセス画像を設定するために使用します。

グラフィックランタイム

グラフィックランタイムは、グラフィックシステムのランタイムコンポーネントです。 グラフィックランタイムは、ランタイム中に画像を表示し、プロジェクトの起動時にすべ ての入力と出力を管理します。

システムブロック

システムブロックは、メッセージブロックに属します。

システムブロックは、システムデータ(日付、時刻、メッセージ番号、ステータスなど)を含 みます。

システムブロックは、事前定義されています。

シングルユーザープロジェクト

「シングルユーザープロジェクト」は、1 台の PC でのみ実行されます。他のコンピュータ はこのプロジェクトにアクセスできません。

プロジェクトは、データを処理するサーバー、およびオペレータステーションとして動作 する1台のコンピュータ上で実行されます。

スタティックオブジェクト

スタティックオブジェクトは、ランタイム中に変化しません。

スタティックオブジェクトには、長方形、円、線、あるいはコネクタなどがあります。

スタティックテキスト

"スタティックテキスト"オブジェクトは、表示したプロセスまたは内容をマークするために 使用するスタティックオブジェクトです。

スライダオブジェクト

スライダオブジェクトは、タグ値の表示や変更のために使用します。

スライダオブジェクトは、タグへの接続によって動的に作成されます。スライダオブジェ クトのプロセスタグへの接続によって、ASを制御できます。

ダイナミックオブジェクト

ダイナミックオブジェクトは、個々のプロセス値に従って変更されます。

ダイナミックオブジェクトの例としては、バーがあります。バーの長さは、現在の温度値に 応じて変化します。

タググループ

タググループは、タグ管理のコンポーネントです。 タググループは、タグを明確な構造に整理するために使用します。

タグロギング

[タグロギング]エディタは、アーカイブシステムの設定コンポーネントです。 このエディタは、アーカイブを設定するために使用します。

タグロギングランタイム

タグロギングランタイムは、アーカイブシステムのランタイムコンポーネントです。タグ ロギングランタイムは、主に次のタスクを実行するために使用されます:

- プロセス値のプロセス値アーカイブへの書き込み
- プロセス値アーカイブからの、アーカイブ済みプロセス値の読み取り

タグ管理

"タグ管理"コンポーネントの下で、WinCCとASの間の通信を設定します。

"タグ管理"で、全てのタグやチャンネルを作成、管理します。

チャンネル

チャンネルは、特殊な通信ドライバです。

チャンネルによって、WinCC とオートメーションシステム(AS)の間の通信が可能になります。

チャンネル経由で、WinCCのプロセスタグにASからのプロセス値が提供されます。

チャンネルユニット

チャンネルユニットは、1つの基礎ハードウェアドライバが装備されたインターフェース、 つまり PC の1つの通信プロセッサとのインターフェイスとして動作します。

さらに、このチャンネルユニットは、特定のタイプの AS にアクセスするために使用され ます。

ディスクリートアラーム

ディスクリートアラームはプロセスのステータスの変化を示し、PLC によって起動されます。

フェイスプレートタイプ/フェイスプレートインスタンス

フェイスプレートタイプは、再利用可能なテンプレートを作成し、プロセス画像でフェイスプレートインスタンスとして使用する特別なプロセス画像です。

インターフェースタグまたは構造タグで、フェイスプレートタイプのオブジェクトをダイ ナミック化します。

フェイスプレートタイプでの変更が、すべてのフェイスプレートインスタンスに自動的に 適用されます。

フェイスプレートタイプは[グラフィックデザイナ]エディタで設定されます。

プロジェクト

プロジェクトは、WinCC のユーザーインターフェースの設定の基本です。

プロジェクト内で、プロセスを操作、監視する必要がある全てのオブジェクトを作成、編集 します。

プロセスタグ

プロセスタグは、WinCC と AS 間のデータ交換用リンクを形成します。 WinCC の各プロセスタグは、接続された AS のいずれかのメモリ中の特定のプロセス値に 対応します。

プロセス画像

プロセス画像はプロジェクトの中心的要素です。 プロセス画像はプロセスを表現し、このプロセスに関する操作や監視を可能にします。 [グラフィックデザイナ]エディタを使用して、プロセス画像を設定できます。

プロセス値

プロセス値は、AS によって提供される値です。

プロセス値アーカイブ

プロセス値アーカイブは、ASからのプロセス値を保存するアーカイブです。

プロセス値ブロック

メッセージをプロセス値(例:現在のフィルレベル、温度、速度)に接続するために使用す るプロセス値ブロック。

ページレイアウト

ページレイアウトでは、外観やレポート出力のためのデータ提供を設定します。

ボタン

ボタンは、プロセス操作を有効にします。

ボタンを使って、メッセージの確認やランタイムのナビゲーションができます。

メインメモリ

コンピュータのメインメモリは、データオブジェクトを保存し、後でそこから取り出すメ モリです。

メインメモリは、作業メモリとも呼ばれます。

じて、ディスクリートアラームがトリガされます。

メッセージクラス

メッセージクラスには、類似の動作を行なうメッセージが含まれます。 メッセージクラスによって、個別のメッセージを一元管理できます。

メッセージタグ

メッセージタグは、プロセスにおけるステータスの変化にリンクしています。 プロセスでステータスの変化が発生すると、タグ値のビットが設定されます。タグ値に応

メッセージの表示色

表示色は、メッセージの現在のステータスを示します。 個別のメッセージの表示色は、"アラームロギング"エディタで決定します。

メッセージビット

[メッセージビット]プロパティによって、ディスクリートアラームをいつトリガするかを 定義できます。

メッセージブロック

メッセージの内容はメッセージブロックで構成されています。

各メッセージブロックは、WinCC アラームコントロールの表形式の表示の中の1列に対応 します。

メッセージレポート

[メッセージ]ウィンドウの現在のメッセージリストにある全てのメッセージは、メッセージレポートで文書化されます。

ユーザーテキストブロック

ユーザーテキストブロックは、メッセージブロックに属します。

ユーザーテキストブロックには、説明テキスト(例:ロケーションおよび障害の原因に関する 情報)が含まれています。

ユーザーテキストブロック内のテキストは自由にカスタマイズできます。

ユーザーテキストブロックの長さが、このブロックに入力できる文字の数を決定します。 文字列の最大長は 255 文字です。

ランタイム

このオプションが有効な場合は、プロジェクトはランタイム中です。

リニアスケーリング

リニアスケーリングの使用時には、プロセスタグの値の範囲を WinCC のプロセスタグの値 の特定の範囲にマップできます。

プロセス値そのものは修正されません。

レポートデザイナ

[グラフィックデザイナ]エディタは、レポートシステムの設定コンポーネントです。

このエディタを使用して、ページや行のレイアウトの設定、さらに印刷ジョブの設定を行います。

レポートランタイム

レポートランタイムは、レポートシステムのランタイムコンポーネントです。

レポートランタイムは、アーカイブまたはコントロールから文書化するデータを取得し、 印刷出力をコントロールします。

印刷ジョブ

印刷ジョブは、プロジェクト文書やランタイム文書を出力します。 印刷ジョブでは、出力媒体、印刷する範囲、印刷の開始時間、その他の出力パラメータを 設定します。

開始画像

開始画像は、[ランタイム]ウィンドウでプロジェクトが起動したときに表示されるプロセス 画像です。

更新サイクル

更新サイクルは、ランタイムの表示を更新する時間間隔を決定します。

構造タグ/構造タイプ

構造タイプは、タグの特定のコンパイレーションを作成するために、WinCC で使用される スキームを示します。

構造タグは、構造タイプにリンクされたタグです。

構造タイプを使用して、複数のタグを構造タグ要素として、1つのステップで同時に作成 することができます。この操作で、内部タグとプロセスタグを作成できます。

コントロールシステムから直接タグ構造を読み込み、WinCC タグ管理にインポートすることもできます。

取得サイクル

取得サイクルは、プロセスタグのプロセス値を読み出す間隔を決定します。

取得サイクルは、WinCC Runtime が起動するとすぐに開始します。

取得サイクルおよびアーカイブサイクルの時間

取得サイクルおよびアーカイブサイクルの時間は、2つのアーカイブの間の時間間隔を決定 します。

制限値モニタリング

制限値モニタリングは、WinCC のアドインです。

制限値モニタを用いると、タグに任意数の制限値を設定できます。これらの制限値のいず れかに違反すると、対応するメッセージがトリガされ、ランタイム中に表示されます。

接続

接続は、定義された個々の AS へのインターフェースを説明します。 データ交換は接続を介してランタイムで行なわれます。 接続は、チャンネルユニットの下で設定されます。

操作可能なオブジェクト

コントロール可能な画面エレメントを使用することで、オペレータはプロセスに対して能 動的な影響を与えることができます。

そのようなオブジェクトとしては、ボタン、スライダオブジェクト、あるいは特定のプロ セスパラメータへの入力に使用する I/O フィールド(入力/出力フィールド)などがあります。

直接タグ接続

直接タグ接続では、プロセス画像のダイナミックオブジェクトと1つのタグを接続します。 ランタイムでタグが値を取ると、この値はダイナミックオブジェクトに直接転送されます。 オブジェクトの動的表示は、タグ値に従ってランタイムで変化します。

内部タグ

内部タグはプロセスリンクを持たず、WinCC 内部の値を転送するだけです。

索引

[

[テーブル]ウィンドウ, 164 WinCC OnlineTableControl, 164 設定, 164 [トレンド]ウィンドウ, 156 WinCC OnlineTrendControl, 156 設定, 156

Α

Alarm_Logging.Pdl, 202

I

I/O フィールド, 66, 71, 202 ダイナミック化, 71, 215 フォントプロパティの定義, 215 更新の定義, 71 出力フォーマットの定義, 71, 215 挿入, 71, 215 入力の制限, 71

Μ

MPI, 26, 31

Q

Quick_Start シングルユーザープロジェクト, 18

S

S7-1200、S7-1500 チャンネル, 26
SAMPLE.Pdl, 51, 64
 作成, 51
 編集, 64
START.Pdl, 51
 作成, 51
 編集, 64
SVG オブジェクト
 ダイナミック化, 67
SVG ライブラリ, 56

Т

Tag_Logging.pdl, 156

W

```
WinCC, 13
  サブシステム.13
  プロジェクト設計ソフトウェア,13
  ランタイムソフトウェア,13
WinCC AlarmControl, 202
WinCC OnlineTableControl, 156, 164
  挿入,164
WinCC OnlineTrendControl, 156
  メッセージの表示, 194
  挿入,156
WinCC Runtime, 13
  プロパティの指定, 87, 173, 222
  開始画像, 173
WinCC エクスプローラ,13
  フィルタ,235
WinCC タグシミュレータ, 93, 137, 176, 229
  シミュレーションタイプの定義,229
  開始, 93, 137, 176, 229
Windows オブジェクト, 55
```

7

```
アーカイブ, 143, 150
 プロセス値アーカイブ, 143, 150
 值, 143
アーカイブサイクル,148
 時間の割り付け,153
 時間の設定,148
アーカイブシステム,144
 タグロギング,144
  タグロギングランタイム,144
アーカイブタグ,150
 作成,150
 名前の変更,153
アイコン,11
アドレス指定,35
 プロセスタグ,35
アナログアラーム,179
 ランタイムでの表示, 194
 制限値の設定,191
 設定,191
```

アラームウィンドウ, 202
WinCC AlarmControl, 202
設定, 202
アラームロギング, 179, 180
開始, 182
アラームロギングランタイム, 180
有効, 222

ウ

ウィンドウ属性, 87 有効, 87

エ

エディタ, 13, 24, 49 アラームロギング, 13, 182 グラフィックデザイナ, 49, 55 タグロギング, 144, 146 タグ管理, 24 レポートデザイナ, 13

オ

```
オブジェクト, 47, 49, 56
 1/0 フィールド, 202, 215
 WinCC AlarmControl, 202
 WinCC OnlineTableControl, 156, 164
 WinCC OnlineTrendControl, 156
 Windows オブジェクト, 55
  スケール,211
  スタティックオブジェクト,47
  スタティックテキスト,61
  スマートオブジェクト,55
  スライダオブジェクト, 202, 207
  ダイナミックオブジェクト,47
 挿入,55
  操作可能なオブジェクト,47
  標準オブジェクト,55
オブジェクトパレット,55
 Windows オブジェクト, 55
  コントロール.55
  スマートオブジェクト,55
  標準オブジェクト,55
オブジェクトプロパティ, 67, 71, 207, 211
  アニメーション化,133
  ダイナミック化, 116, 133
```

ク

グラフィックシステム,49 グラフィックデザイナ,49 グラフィックランタイム,49 グラフィックデザイナ,49,55 オブジェクトパレット,47 プロセス画像の設定,47 ライブラリ,55 グラフィックランタイム,49 有効,87

Э

コントロール, 55 WinCC AlarmControl, 202 WinCC OnlineTableControl, 156, 164 WinCC OnlineTrendControl, 156 コンポーネント, 13 WinCC, 13 アーカイブシステム, 144 グラフィックシステム, 49 メッセージシステム, 180

シ

システムダイアログ 使用,97 システムブロック,183 シミュレーション,93,137 WinCC タグシミュレータ,93,137,176,229 シミュレーションタイプ,93,137,176,229 開始,93,137,229 終了,229 シミュレーションタイプ,229

ス

```
スクリプト
検索,233
スケール,41,211
サイズの定義,211
フォントサイズの定義,211
プロセスタグ,41
挿入,211
スタティックテキスト,61
フォントサイズを変更する,61
挿入,61
```

ステータス, 187 バルブ, 179 メッセージ, 200 スマートオブジェクト, 55 スライダオブジェクト, 202 サイズの定義, 207 ダイナミック化, 207 色の定義, 207 挿入, 207 名前の定義, 207

タ

ダイナミックオブジェクト,47 ダイナミック化,66,67 1/0 フィールド, 71, 215 スライダオブジェクト,207 タイプ変換,35 プロセスタグ.35 タイマ,148 作成.148 設定,148 タグ,24 アーカイブタグ,150 スケーリング.41 プロセスタグ, 24, 35, 41 リンク,67 検索.233 作成, 35, 43 内部タグ,24,43 タググループ,24 作成,35 タグロギング,144 開始, 146 タグロギングランタイム,144 有効,173 タグ管理,24 タググループ,35 プロセスタグ,24,35 内部タグ,24,43 タグ接続.66 直接タグ接続,66

チ

チャンネル, 23 SIMATIC S7-1200、S7-1500 チャンネル, 26 挿入, 26 チャンネルユニット, 23, 31 MPI, 26 挿入, 26

テ

ディスクリートアラーム, 186 作成, 187 データタイプ, 35, 43 プロセスタグ, 35 符号なし 16 ビット値, 43 符号付き 16 ビット値, 35 データ交換, 24 プロセスタグ, 24 テキストフィールド, 61 テスト, 93, 137 プロジェクト, 93, 137, 176, 229

フ

フィルタ,235 フェイスプレートインスタンス 作成,129 使用,129 フェイスプレートタイプ オブジェクトの挿入,103 ポップアップ画面, 112, 113, 116, 127 作成, 103, 120 設定, 103, 120, 122, 127, 133 変更,133 フォントサイズ,61 変更, 61, 71, 211 プロジェクト, 17, 90, 93, 137 WinCC の起動, 18 シングルユーザープロジェクト.18 テスト, 93, 137, 176, 229 作成,18 有効,90,176,225 プロジェクト設計ソフトウェア,13 WinCC, 13 プロセス,13 モニタリング,13,179 視覚化, 13, 47, 49 操作, 13, 21, 23, 24 プロセスタグ,24 アドレス指定,35 スケーリング,41 タイプ変換,35 データタイプ,35 作成,35 プロセス画像,47 Alarm Logging.Pdl, 202 SAMPLE.Pdl, 51 START.Pdl, 51

Tag Logging.Pdl, 156 ダイナミック化,66 作成,51 設定, 47, 156, 202 編集, 55, 64 保存, 61, 64 名前の変更,51 プロセス値, 21, 24, 143 アーカイブ, 143, 148 ウィンドウ.143 シミュレーション, 93, 137, 176, 229 入力,148 プロセス値アーカイブ,143 サイズの定義,153 メモリの場所の定義, 153 作成,150 編集, 153 保存, 153 プロセス値ブロック,183

ホ

ポップアップ画面 フェイスプレートタイプでの使用, 112, 113, 116, 127 作成, 112, 113, 116

7

マニュアル類,143

メ

メッセージ,179 アナログアラーム,179 ディスクリートアラーム,179 メッセージステータス,200 メッセージステータスの色の定義,200 確認,225 制限値モニタリング,191 設定, 179 保存, 191 メッセージクラス,183 定義, 183 メッセージシステム, 180 アラームロギング,180 アラームロギングランタイム,180 メッセージステータス, 187, 200 メッセージブロック,183 システムブロック,183 プロセス値ブロック,183

ユーザーテキストブロック,183 定義,183 表示の有効化,202 メッセージリスト,225

ユ

ユーザーテキストブロック,183

ラ

```
ライブラリ, 55, 56
グローバルライブラリ, 56
プロジェクトライブラリ, 56
開く, 56
ランタイム, 13
ランタイムウィンドウ, 87
ウィンドウ属性, 87
開始画像, 87, 173, 222
ランタイムコンポーネント, 13
アーカイブシステム, 144
グラフィックシステム, 144
グラフィックシステム, 49
メッセージシステム, 180
ランタイムシステムダイアログ, 97
ランタイムソフトウェア, 13
WinCC, 13
```

IJ

リニアスケーリング, 41 例, 41

\mathcal{V}

レポート 検索,233

画

画像 検索, 233

開

開く, 18, 51 プロジェクト, 18 プロセス画像, 51, 64 ライブラリ, 56 開始, 18 WinCC, 18 WinCC タグシミュレータ, 176 アラームロギング, 182 タグロギング, 146 開始画像, 87, 173 定義, 87, 222

確

確認, 225 メッセージ, 225

機

機能 検索, 233

更

更新サイクル I/O フィールド, 71

構

構造タイプ データタイプ, 106, 109 リンク, 122 作成, 106 構造タグ, 106, 109 作成, 109 構造タグ要素, 109 構造要素, 106

作

作成, 31, 35, 43, 156 [テーブル]ウィンドウ, 164 [トレンド]ウィンドウ, 156 アナログアラーム, 191 アラームウィンドウ, 202 タイマ, 148 タググループ, 35 ディスクリートアラーム, 187 プロジェクト, 17, 18 プロセスタグ, 35 プロセス画像, 51 プロセス値アーカイブ, 150 接続, 31 内部タグ, 43

実

実例, 56 スケール, 211

取

取得サイクル, 148 時間の割り付け, 153 時間の設定, 148

色

色, 200 メッセージステータスの色の定義, 200

制

制限値, 191 下限値, 191 上限値, 191 定義, 191 制限値モニタリング, 191 ランタイムでのメッセージの表示, 194 制限値の設定, 191

接

接続, 23, 31 作成, 31

設

設定, 21, 55, 156
[テーブル]ウィンドウ, 164
[トレンド]ウィンドウ, 156
アラームウィンドウ, 202
プロセス画像, 55, 156, 202
プロセス値アーカイブ, 150, 153
メッセージ, 179, 186, 191
取得サイクルおよびアーカイブサイクルの時間, 148
通信, 21
設定コンポーネント, 49, 144, 180
アーカイブシステム, 144
グラフィックシステム, 49
メッセージシステム, 180

値

値, 21 アーカイブ, 143 シミュレーション, 93, 137 プロセス値, 21, 24, 143 値の範囲, 41 プロセスタグのスケーリング, 41

通

通信, 21
チャンネル, 23, 26
チャンネルユニット, 23, 26
プロセスタグ, 24, 35
接続, 23, 31
設定, 21
通信ドライバ, 23
SIMATIC S7-1200、S7-1500 チャンネル, 26

貼

貼り付け, 26 I/O フィールド, 71, 215 SIMATIC S7-1200、S7-1500 チャンネル, 26 WinCC AlarmControl, 202 WinCC OnlineTableControl, 164 WinCC OnlineTrendControl, 156 スケール, 211 スタティックテキスト, 61 スライダオブジェクト, 207 チャンネル, 26 チャンネルユニット, 26 ライブラリからのオブジェクト, 56

内

内部タグ,24 データタイプ,43 構造タイプ,106,109 作成,43

発

発見, 233

標

標準オブジェクト,55 スタティックテキスト,61

表

```
表示色, 200
メッセージステータスの色の定義, 200
表示要素, 71, 143, 207
値, 71, 143, 207
```

符

符号なし 16 ビット値, 43 符号付き 16 ビット値, 35

編

```
編集,55
スタティックテキスト,61
プロセス画像,55,64
プロセス値アーカイブ,153
```

保

```
保持, 176
保存, 64
プロセス画像, 61, 64
プロセス値アーカイブ, 153
メッセージ, 191
```

名

名前の変更, 51, 153 アーカイブタグ, 153 プロセス画像, 51

有

有効,90 アラームロギングランタイム,222 ウィンドウ属性,87 グラフィックランタイム,87 タグロギングランタイム,173 プロジェクト,90,176,225 リニアスケーリング,41