

SIEMENS

Ingenuity for life

24/7

Industry Online Support

Home

Grundlagenwissen zu HMI-Bildbausteinen

SIMATIC Comfort Panels, Runtime Advanced und
WinCC (TIA Portal)

<https://support.industry.siemens.com/cs/ww/DE/view/68014632>

Siemens
Industry
Online
Support



Gewährleistung und Haftung

Hinweis

Die Anwendungsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Anwendungsbeispiele stellen keine kundenspezifischen Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Anwendungsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Anwendungsbeispiele erkennen Sie an, dass wir über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden können. Wir behalten uns das Recht vor, Änderungen an diesen Anwendungsbeispiele jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesem Anwendungsbeispiel und anderen Siemens Publikationen, wie z. B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr.

Unsere Haftung, gleich aus welchem Rechtsgrund, für durch die Verwendung der in diesem Applikationsbeispiel beschriebenen Beispiele, Hinweise, Programme, Projektierungs- und Leistungsdaten usw. verursachte Schäden ist ausgeschlossen, soweit nicht z. B. nach dem Produkthaftungsgesetz in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird. Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden.

Weitergabe oder Vervielfältigung dieser Anwendungsbeispiele oder Auszüge daraus sind nicht gestattet, soweit nicht ausdrücklich von der Siemens AG zugestanden.

Security-hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Lösungen, Maschinen, Geräten und/oder Netzwerken unterstützen. Sie sind wichtige Komponenten in einem ganzheitlichen Industrial Security-Konzept. Die Produkte und Lösungen von Siemens werden unter diesem Gesichtspunkt ständig weiterentwickelt. Siemens empfiehlt, sich unbedingt regelmäßig über Produkt-Updates zu informieren.

Für den sicheren Betrieb von Produkten und Lösungen von Siemens ist es erforderlich, geeignete Schutzmaßnahmen (z. B. Zellschutzkonzept) zu ergreifen und jede Komponente in ein ganzheitliches Industrial Security-Konzept zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen.

Weitergehende Informationen über Industrial Security finden Sie unter <http://www.siemens.com/industrialsecurity>.

Um stets über Produkt-Updates informiert zu sein, melden Sie sich für unseren produktspezifischen Newsletter an. Weitere Informationen hierzu finden Sie unter <http://support.industry.siemens.com>.

Inhaltsverzeichnis

Gewährleistung und Haftung	1
Vorwort	5
1 Grundlagen	6
1.1 Was ist ein Bildbaustein?	6
1.2 Wie wird ein Bildbaustein erstellt und geöffnet?	6
1.3 Was ist mit dem Editor "Bilder" gemeint?	6
1.4 Was ist ein Bildobjekt?	7
1.5 Unterschied zwischen Bildobjekt und Bildbaustein?	7
1.5.1 Bedienfeld als Bildobjekt erstellt	8
1.5.2 Bedienfeld als "Bildbaustein" erstellt	10
1.6 Was ist ein "HMI UDT" (Anwenderdatentyp)?	12
1.7 Was ist ein PLC-Datentyp ("PLC-UDT")?	13
1.8 Unterschied "HMI UDT" und "PLC-UDT"	13
1.9 Projektbibliothek und Globale Bibliotheken	13
2 Funktionen und Eigenschaften des Bildbausteins	14
2.1 Bildbaustein erstellen	14
2.1.1 Bildbaustein über den Editor "Bilder" erstellen	14
2.1.2 Bildbaustein über den Bildbaustein Editor erstellen	16
2.2 Bildbaustein öffnen	18
2.2.1 Bildbaustein im Editor "Bilder" öffnen	18
2.2.2 Bildbaustein über die Projektbibliothek öffnen	19
2.3 Bildbaustein freigeben bzw. wiederherstellen	20
2.3.1 Bildbaustein freigeben	21
2.3.2 Bildbaustein wiederherstellen	22
2.3.3 Versionsbezeichnung eines Bildbausteins	23
2.4 Bildbaustein aufrufen bzw. in ein Bild einfügen	24
2.5 Bildbaustein Editor	25
2.5.1 Arbeitsbereich	26
2.5.2 Konfigurationsbereich	27
2.5.3 Inspektorfenster	28
2.6 Parameter im Arbeitsbereich	30
2.7 Parameter im Konfigurationsbereich	31
2.7.1 Registerkarte "Eigenschaften"	31
2.7.2 Registerkarte "Ereignisse"	43
2.7.3 Registerkarte "Variablen"	49
2.7.4 Registerkarte "Skripte"	50
2.7.5 Registerkarte "Textlisten"	51
2.7.6 Registerkarte "Grafiklisten"	52
2.7.7 Registerkarte "Texte"	53
2.7.8 Registerkarte "Sprachen"	54
2.8 Parameter im Inspektorfenster	55
2.9 Bildbausteine unter "Globale Bibliotheken" speichern	56
3 Funktionen und Eigenschaften eines HMI UDT	59
3.1 Anwenderdatentyp erstellen	60
3.2 Anwenderdatentyp bearbeiten/öffnen	62
3.3 Anwenderdatentyp Konfigurieren	63
3.4 Anwenderdatentyp	65
3.4.1 Anwenderdatentyp freigeben	66
3.4.2 Anwenderdatentyp wiederherstellen	67
3.5 HMI UDT einer HMI-Variablen zuweisen	68
4 Funktionen und Eigenschaften eines PLC-Datentypen	70

4.1	Einleitung.....	70
4.2	PLC-Datentypen erstellen	71
4.3	PLC-Datentyp in einen Datenbaustein einfügen.....	72
4.4	PLC-Datentypen in einen Bildbaustein einbinden.....	72
4.5	PLC-Datentypen bearbeiten/öffnen.....	74
4.6	Tipps zur Projektierung	75
4.6.1	Reduzierung von Powertags	75
4.6.2	Umstieg von HMI UDT nach PLC-Datentypen.....	75
5	Literaturhinweise	77
6	Historie.....	77

Vorwort

Ziel des Dokuments

Ziel des Dokuments ist es, Ihnen alles Wissenswerte zum Thema "Erstellen von HMI-Bildbausteinen" zu vermitteln und wie Sie HMI-Bildbausteine als Anwender nutzen können.

Gültigkeit

- Softwareversion ab WinCC Comfort V14
- Für alle SIMATIC HMI Bediengeräte, die Bildbausteine unterstützen.
Informationen zum Funktionsumfang der SIMATIC HMI Bediengeräte finden Sie in der Online Hilfe von WinCC (TIA Portal) bzw. im FAQ <http://support.automation.siemens.com/WW/view/de/40227286.13>

1 Grundlagen

In dem Kapitel Grundlagen werden einzelne, applikationsbezogene Begriffe, beschrieben.

1.1 Was ist ein Bildbaustein?

Ein Bildbaustein ist eine konfigurierte Gruppe von Anzeige- und Bedienobjekten. Über eine Schnittstelle am erstellten Bildbaustein erfolgt der "Datenaustausch" mit den im Bildbaustein enthaltenen Anzeige- und Bedienobjekten.

Die Eigenschaften der verwendeten Anzeige- und Bedienobjekte werden im "Bildbaustein Editor" den einzelnen Objekten zugewiesen.

Bildbausteine können zentral in einer Bibliothek verwaltet und geändert werden.

1.2 Wie wird ein Bildbaustein erstellt und geöffnet?

Bildbausteine können auf zwei Arten erstellt und geöffnet werden.

- Erstellen- und öffnen im Editor "Bilder".
- Erstellen- und öffnen über die Projektbibliothek.

Details hierzu werden im Kapitel 2.1 beschrieben.

1.3 Was ist mit dem Editor "Bilder" gemeint?

Wenn in dieser Applikation vom Editor "Bilder" gesprochen wird, dann handelt es sich um den Editor, indem Sie Ihre HMI Bilder erstellen und bearbeiten.

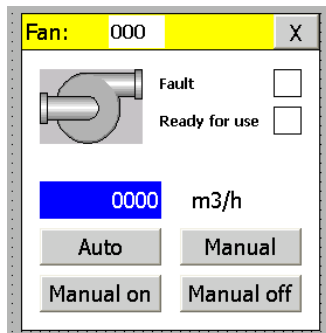
Im Editor "Bilder" stehen Ihnen z. B. folgende Ihnen bekannte Task Cards zur Verfügung:

- Werkzeuge:
 - Anzeige- und Bedienobjekte
- Animationen:
 - Vorlagen zur dynamischen Projektierung.
- Layout:
 - Hilfsmittel zur Anpassung der Darstellung
- Bibliotheken:
 - Verwaltung der Projektbibliothek und der Globalen Bibliotheken.

1.4 Was ist ein Bildobjekt?

Wenn in dieser Applikation von einem Bildobjekt gesprochen wird, dann handelt es sich hierbei um eine Zusammenstellung von mehreren einzelnen Anzeige- und Bedienobjekten, wie z. B. EA-Felder und Schaltflächen, deren Eigenschaften **einzel**n bearbeitet werden können. Als Beispiel wird in dieser Applikation das in der Abbildung zu sehende "Bedienfeld" verwendet.

Abbildung 1-1



Bedienfeld bestehend aus:

- Schaltflächen
- EA-Feldern
- Textfeldern

1.5 Unterschied zwischen Bildobjekt und Bildbaustein?

Wenn Sie sich ein "Bedienfeld" von zusammengestellten Objekten ansehen, dann können Sie zunächst nicht erkennen, ob dieses "Bedienfeld" als "herkömmliches" Bildobjekt oder als Bildbaustein erstellt wurde.

Erst wenn Sie sich die Eigenschaften des Bildobjektes anschauen, werden Sie die einzelnen Unterschiede erkennen.

An dem nachfolgend beschriebenen Beispiel wird der prinzipielle Unterschied zwischen einem Bedienfeld gezeigt, das

- als "herkömmliches" Bildobjekt erstellt wurde.
- als Bildbaustein erstellt wurde.

Anhand der beschriebenen Aufgabe, werden die Vorteile eines Bildbausteins besonders deutlich.

1.5.1 Bedienfeld als Bildobjekt erstellt

Sie haben die Aufgabe für ca. 20 Lüfterantriebe jeweils ein einheitliches Bedienfeld zu erstellen. Als Vorlage dient Ihnen das in der Abbildung 1-2 vorliegende **Bildobjekt**.

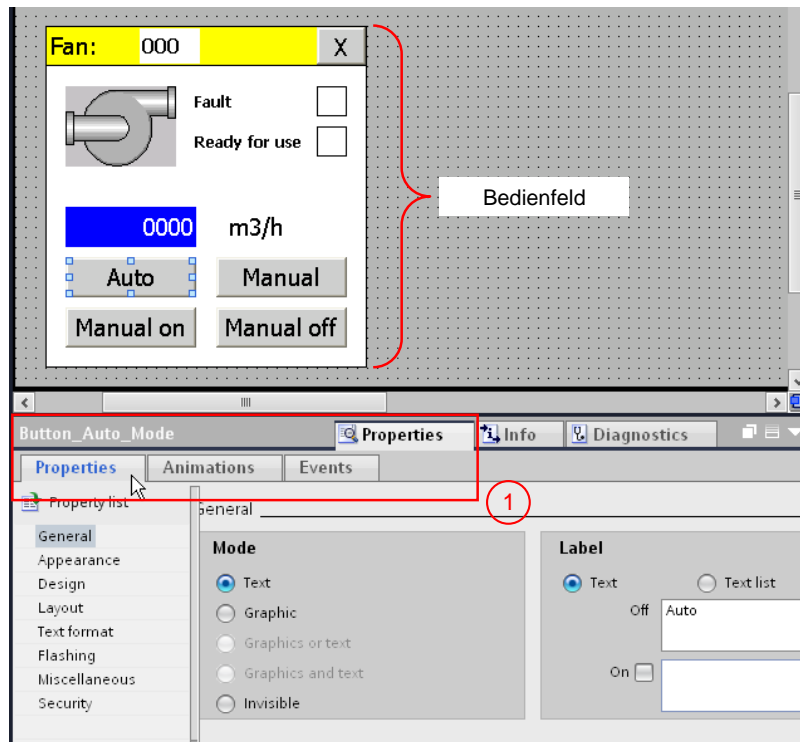
Das Bildobjekt besteht aus einer Vielzahl von einzelnen Objekten. Für jedes verwendete Objekt können Sie "Eigenschaften", "Animationen" und "Ereignisse" festlegen (1).

Die Eigenschaften wie z. B. die "Beschriftung", "Layout" usw. haben Sie für jedes Objekt vorgegeben.

Unter dem Register "Ereignisse" bzw. "Animationen" sind an den projektierten Objekten verschiedene Systemfunktionen, wie z. B. "SetzeBit" usw., projektiert.

Abhängig von der verwendeten Systemfunktion (z. B. "SetzeBit"), müssen diese mit einer HMI-Variablen verschaltet werden.

Abbildung 1-2



Problem

- Bei jedem der 20 Bedienfelder müssen Sie an den Objekten, an denen Sie eine Systemfunktion projektiert haben, die HMI-Variablen für den zugehörigen Antrieb anpassen.
 - Hier kann es schnell zu Fehlern bei der Auswahl der Variablen für die einzelnen Antriebe kommen.
 - Die Zuweisung ist sehr zeitintensiv.

1.5 Unterschied zwischen Bildobjekt und Bildbaustein?

- Sie haben alle Bedienfelder im Projekt parametrieren. Ihr Kunde benötigt nachträglich eine weitere Schaltfläche in dem Bedienfeld.
 - Sie müssen jedes der 20 Bedienfelder einzeln aufrufen und anpassen. Hoher zeitlicher Aufwand.
- Sie haben alle Bedienfelder im Projekt parametrieren. Ihr Kunde möchte nachträglich eine weitere Funktion bzw. Animation hinzugefügt haben.
 - Sie müssen jedes der 20 Bedienfelder einzeln aufrufen und anpassen. Hoher zeitlicher Aufwand.
- Ihre Kunden verwenden HMI Bediengeräte mit unterschiedlichen Displaygrößen.
 - Sie müssen Bedienfelder in verschiedenen Größen erstellen, damit diese für die unterschiedlichen Displaygrößen verwendet werden können. Hierzu müssen Sie die Schriftgröße und die Abmessungen der einzelnen Objekte anpassen. Hoher zeitlicher Aufwand.

1.5.2 Bedienfeld als "Bildbaustein" erstellt

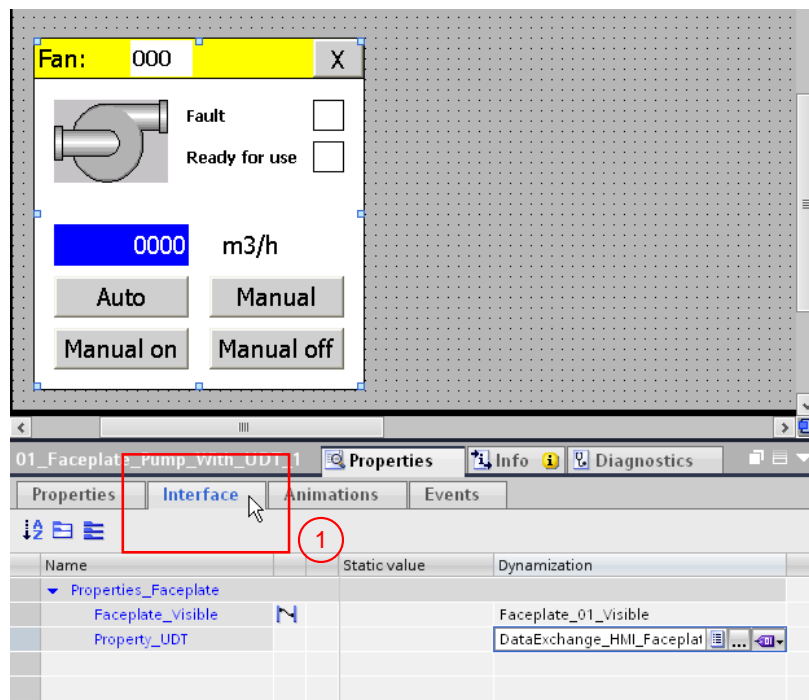
Sie haben die Aufgabe für ca. 20 Lüfterantriebe jeweils ein einheitliches Bedienfeld zu erstellen. Als Vorlage dient Ihnen der in der Abbildung 1-3 gezeigte **Bildbaustein**.

Alle Eigenschaften und Ereignisse sind im Bildbaustein Editor festgelegt und einem Anwenderdatentyp (HMI UDT) zugewiesen worden. Mit einem Anwenderdatentyp (HMI UDT) fassen Sie eine Anzahl verschiedener Variablen zusammen.

Über das Register "Schnittstelle" erfolgt der Datenaustausch (1).

Hinweis Durch das zusätzliche Register "Schnittstelle" ist das "Bildobjekt" als Bildbaustein ersichtlich.

Abbildung 1-3



Vorteile durch den Einsatz eines Bildbausteins

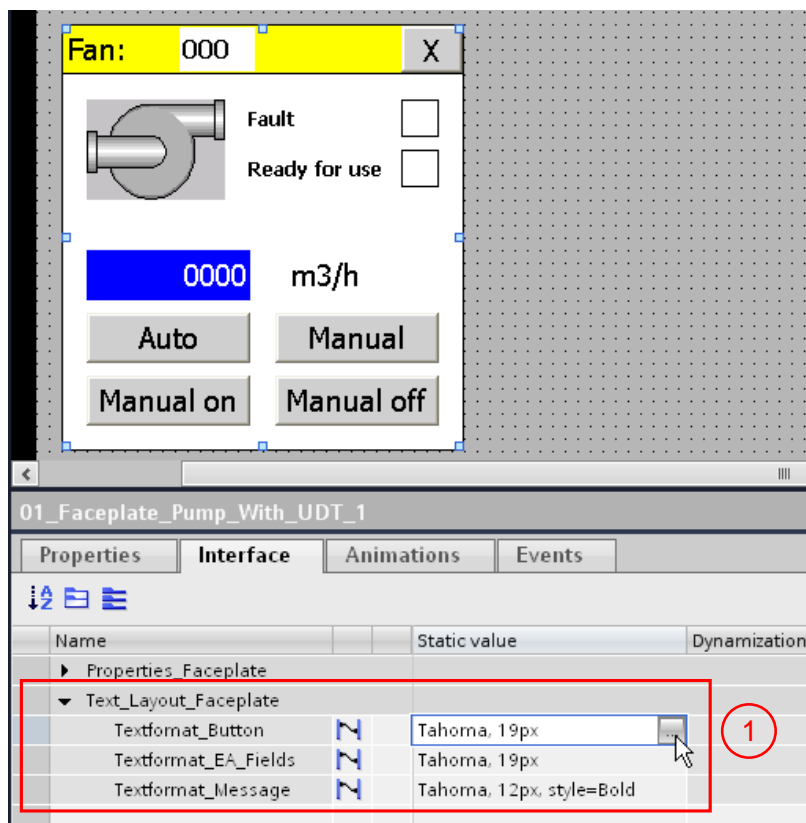
- Anstelle vieler einzelner HMI-Variablen wird in diesem Fall nur eine Variable pro Antrieb benötigt. Die verwendete Variable besitzt hierzu eine Struktur.
 - Durch den einfachen Aufbau kann es nicht so schnell zu einem Fehler bei der Variablenzuweisung kommen. Der zeitliche Aufwand verringert sich.
- Sie haben alle Bedienfelder im Projekt parametrisiert. Ihr Kunde benötigt nachträglich eine weitere Schaltfläche in dem Bedienfeld.
 - Sie können den verwendeten Bildbaustein zentral ändern und um die Schaltfläche erweitern. Die Änderungen werden anschließend für alle zugehörigen Bildbausteine übernommen. Der Projektierungsaufwand verringert sich.

1 Grundlagen

1.5 Unterschied zwischen Bildobjekt und Bildbaustein?

- Sie haben alle Bedienfelder im Projekt parametriert. Ihr Kunde möchte nachträglich eine weitere Funktion bzw. Animation hinzugefügt haben.
 - Sie können den verwendeten Bildbaustein zentral ändern und um die Funktion bzw. Animation erweitern. Die Änderungen werden anschließend für alle zugehörigen Bildbausteine übernommen. Der Projektierungsaufwand verringert sich.
- Ihre Kunden verwenden HMI Bediengeräte mit unterschiedlichen Displaygrößen.
 - Sie haben bei der Erstellung von Bildbausteinen die Möglichkeit, die Größe z. B. des "Textformates" oder das des "Layouts" als "Parameter" (statische Variable) zu deklarieren. Sie können somit später über das Register "Schnittstelle" die Schriftgröße individuell für alle Objekte vorgeben (1). Der Bildbaustein kann somit für unterschiedliche Displaygrößen verwendet werden. Der Projektierungsaufwand verringert sich.

Abbildung 1-4



Hinweis

Details zur Projektierung finden Sie im Dokument "Projektierungsbeispiele für HMI-Bildbausteine". (68014632_Faceplates_EngineeringExamples_de).

1.6 Was ist ein "HMI UDT" (Anwenderdatentyp)?

Hinweis Ab WinCC (TIA Portal) V13 SP1 und bei der Verwendung einer S7-1200 bzw. S7-1500 Steuerung benötigen Sie keinen "HMI UDT" mehr.
Sehen Sie sich hierzu das Kapitel [1.7](#) bzw. [1.8](#) an.

Mit Anwenderdatentypen, kurz "HMI UDT"; fassen Sie eine Anzahl verschiedener Variablen zusammen, die eine logische Einheit bilden (unter WinCC flexible 2008 vormals "Struktur" genannt).

Der Vorteil liegt in der einmaligen Erstellung von einer "Gruppe" von Variablen, die Sie in mehreren Bildbausteinen anschließend verwenden können.

Beispiel:

Die unterschiedlichen Zustände eines Motors lassen sich z. B. mit 5 einzelnen Variablen beschreiben.

- Zustand "Aus"
- Zustand "Ein"
- Zustand "Störung"
- Zustand "Handbetrieb"
- Zustand "Automatikbetrieb"

Diese Variablen (Anwenderdatentyp-Elemente) werden zu einer "Gruppe" zusammengefasst. Die Funktionalität ist die gleiche, wie Sie es aus der STEP 7 Programmierung her kennen, wenn Sie eine Struktur in einem Datenbaustein anlegen.

Details zur Erstellung eines Anwenderdatentyps werden im Kapitel 3 beschrieben.

Wie wird ein Anwenderdatentyp erstellt und geöffnet?

Erstellt- und geöffnet werden Anwenderdatentypen über die Projektbibliothek.

Details hierzu werden im Kapitel 3 beschrieben.

1.7 Was ist ein PLC-Datentyp ("PLC-UDT")?

PLC-Datentypen, kurz "PLC-UDT", sind von Ihnen definierte Datenstrukturen, die Sie im Programm mehrmals verwenden können. Die Struktur eines PLC-Datentyps setzt sich aus mehreren Komponenten zusammen. Die Art der Komponenten legen Sie bei der Deklaration des PLC-Datentyps fest.

Der Vorteil liegt in der einmaligen Erstellung von einer "Gruppe" von Variablen, die Sie in mehreren Bildbausteinen anschließend verwenden können.

Beispiel:

Die unterschiedlichen Zustände eines Motors lassen sich z. B. mit 5 einzelnen Variablen beschreiben.

- Zustand "Aus"
- Zustand "Ein"
- Zustand "Störung"
- Zustand "Handbetrieb"
- Zustand "Automatikbetrieb"

Diese 5 Variablen werden zu einer "Gruppe" zusammengefasst. Die Funktionalität ist die gleiche, wie Sie es aus der STEP 7 Programmierung her kennen, wenn Sie eine Struktur in einem Datenbaustein anlegen.

Details zur Erstellung eines PLC-Datentypen werden im Kapitel 4 beschrieben.

1.8 Unterschied "HMI UDT" und "PLC-UDT"

Der wesentlichste Unterschied zwischen einem "HMI UDT" und einem "PLC-UDT" ist der, dass

- ein "HMI UDT" **nur** in Bildbausteinen verwendet werden kann.
- die Datenstruktur eines "PLC-UDTs" nur einmal angelegt werden muss. Im Gegensatz zum "HMI UDT". Hier müssen Sie exakt zweimal die gleiche Datenstruktur im Programm hinterlegen.
- bei Änderungen im zugehörigen Datenbaustein, diese anschließend manuell im "HMI UDT" nachgetragen werden müssen. Dieses kann zu Fehlern bei der Namen/Variablentyp Zuweisung führen.
- das Handling von "PLC-UDT" in Verbindung mit Bildbausteinen einfacher ist.

1.9 Projektbibliothek und Globale Bibliotheken

Die erstellten Bildbausteine werden in der Projektbibliothek angelegt und bearbeitet. Wenn Sie die erstellten Bildbausteine projektübergreifend wiederverwenden möchten, dann speichern Sie die fertig gestellten Bildbausteine unter "Globale Bibliotheken" ab.

Details hierzu werden im Kapitel 2.9 beschrieben.

2 Funktionen und Eigenschaften des Bildbausteins

In diesem Kapitel werden die Funktionen und Eigenschaften eines Bildbausteins und die einzelnen Parameter des Bildbaustein Editors beschrieben.

Wenn Sie mit den Funktionen und Eigenschaften wie z. B. den Parametern im "Bildbaustein Editor" noch nicht vertraut sind, dann lesen Sie sich dieses Kapitel unbedingt durch.

2.1 Bildbaustein erstellen

Um einen Bildbaustein zu erstellen gibt es zwei Möglichkeiten.

- Bildbaustein über den Editor "Bilder" erstellen.
- Bildbaustein über den "Bildbaustein Editor" erstellen.

Nachfolgend werden die einzelnen Punkte an einem Beispiel erklärt.

Wenn Sie das Kapitel nachstellen wollen, dann genügt es, wenn Sie zunächst z. B. nur eine Schaltfläche als Objekt verwenden.

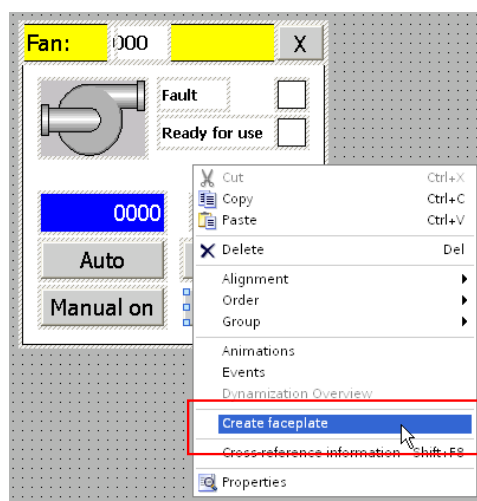
2.1.1 Bildbaustein über den Editor "Bilder" erstellen

Einen Bildbaustein über den Editor "Bilder" zu erstellen bietet sich immer dann an, wenn Sie z. B. bereits ein fertig projektiertes "Bildobjekt" erstellt haben. Sie können auf bereits bestehende "Bildobjekte" zurückgreifen.

Um einen Bildbaustein über den Editor "Bilder" zu erstellen gehen Sie wie folgt vor.

- Erstellen Sie ein Bild und fügen Sie in dieses Bild alle benötigten Objekte ein bzw. rufen Sie ein bestehendes "Bildobjekt" auf. .
- Markieren Sie alle Objekte in dem Bild, die Sie für den Bildbaustein benötigen.
- Klicken Sie mit der rechten Maustaste auf eines der markierten Objekte. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem geöffneten Kontextmenü den Punkt "Erzeuge Bildbaustein-Typ".
Die nachfolgende Abbildung zeigt das Kontextmenü.

Abbildung 2-1

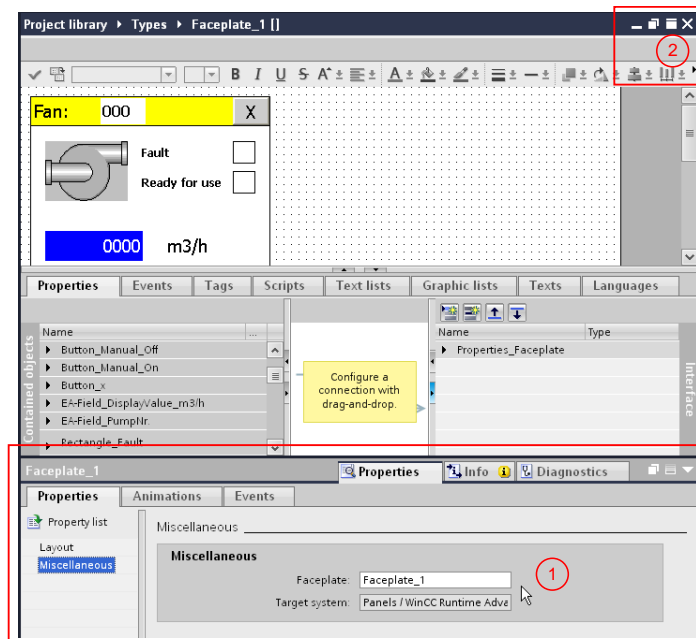


2 Funktionen und Eigenschaften des Bildbausteins

2.1 Bildbaustein erstellen

- Der Bildbaustein Editor zum Erstellen / Bearbeiten des Bildbausteins wird geöffnet.
Im Inspektorfenster (1) unter "Eigenschaften > Eigenschaften > Verschiedenes" können Sie den Namen des Bildbausteins anpassen. Unter "Bediengerät Klasse" wird angezeigt, für welche Runtime der Bildbaustein geeignet ist.

Abbildung 2-2



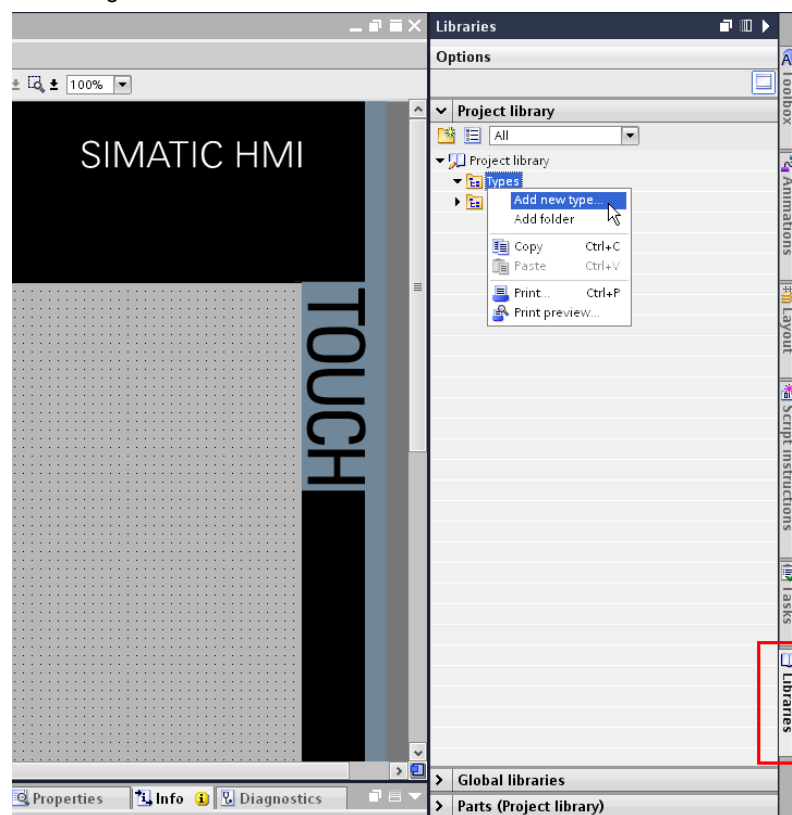
- Schließen Sie zunächst den erstellten Bildbaustein über die Schaltfläche "Schließen" "X" (2).

2.1.2 Bildbaustein über den Bildbaustein Editor erstellen

Um einen Bildbaustein über den Bildbaustein Editor zu erstellen gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie mit der rechten Maustaste den Ordner "Typen". Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Neuen Typen hinzufügen..." an. Das Fenster "Neuen Typ hinzufügen" wird geöffnet.

Abbildung 2-3

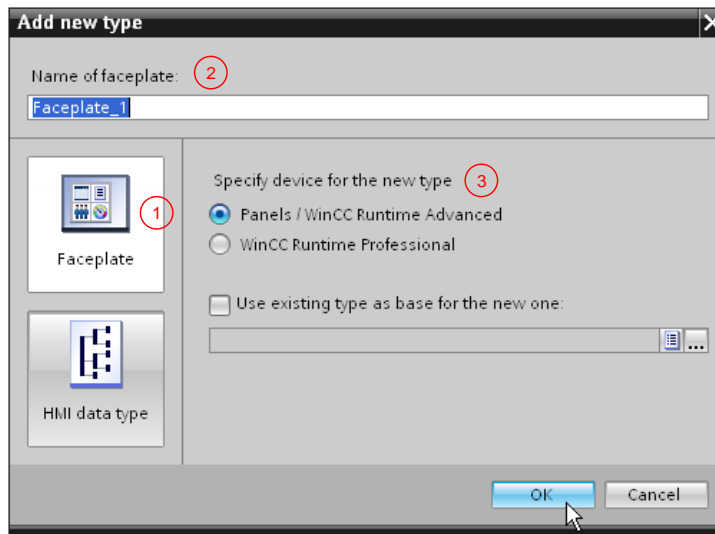


2 Funktionen und Eigenschaften des Bildbausteins

2.1 Bildbaustein erstellen

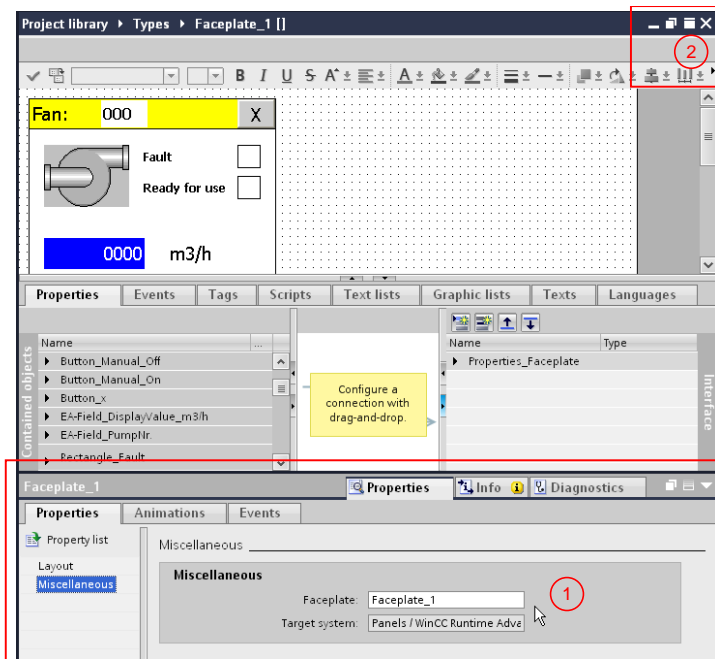
Ansicht des Fensters "Neuen Typ hinzufügen".

Abbildung 2-4



- Wählen Sie auf der linken Seite den Typ "HMI Bildbaustein" an (1).
- Vergeben Sie dem Bildbaustein einen eindeutigen Namen (2).
- Wählen Sie unter "Gerät für den neuen Typ festlegen" das Optionsfeld "Panels / WinCC Runtime Advanced" an (3).
- Bestätigen Sie die Eingaben mit "OK".
Der Bildbaustein Editor zum Erstellen / Bearbeiten des Bildbausteins wird geöffnet.
- Fügen Sie nun alle Objekte die Sie benötigen in den "Arbeitsbereich" des Bildes ein. Die Vorgehensweise ist die gleiche, wie Sie es vom Editor "Bilder" kennen.

Abbildung 2-5



2.2 Bildbaustein öffnen

- Schließen Sie zunächst den erstellten Bildbaustein über die Schaltfläche "Schließen" "X" (2).

2.2 Bildbaustein öffnen

Um einen Bildbaustein zu öffnen gibt es zwei Möglichkeiten.

- Bildbaustein über den Editor "Bilder" öffnen
- Bildbaustein über die "Projektbibliothek" öffnen.

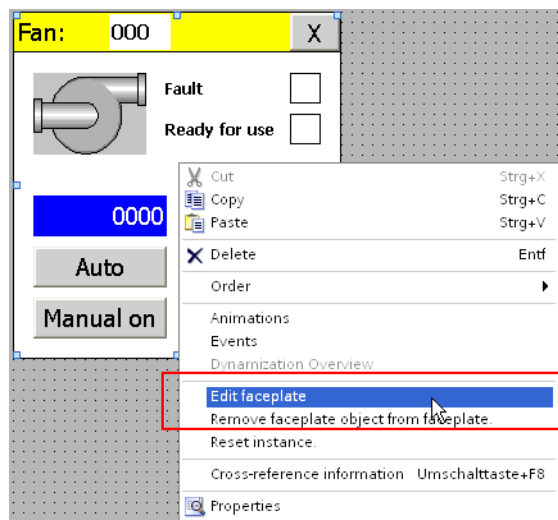
Nachfolgend werden die einzelnen Punkte an einem Beispiel erklärt.

2.2.1 Bildbaustein im Editor "Bilder" öffnen

Um einen bestehenden Bildbaustein über den Editor "Bilder" zu öffnen gehen Sie wie folgt vor:

- Rufen Sie das Bild auf, in dem sich der Bildbaustein befindet.
- Klicken Sie mit der rechten Maustaste auf den Bildbaustein. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem geöffneten Kontextmenü den Punkt "Bildbaustein bearbeiten". Der Bildbaustein Editor zum Erstellen / Bearbeiten des Bildbausteins wird geöffnet.

Abbildung 2-6

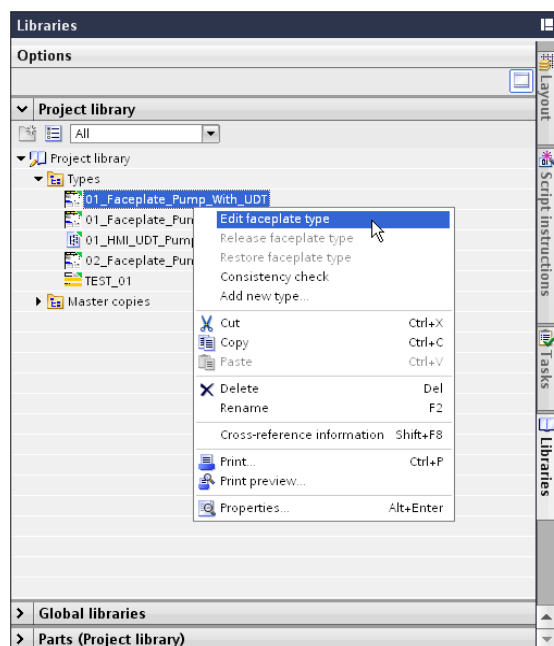


2.2.2 Bildbaustein über die Projektbibliothek öffnen

Um einen vorhandenen Bildbaustein aus der Projektbibliothek zu öffnen, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Bildbaustein den Sie öffnen möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Bildbaustein-Typen bearbeiten" an. Der Bildbaustein Editor zum Erstellen / Bearbeiten des Bildbausteins wird geöffnet.

Abbildung 2-7



2.3 Bildbaustein freigeben bzw. wiederherstellen

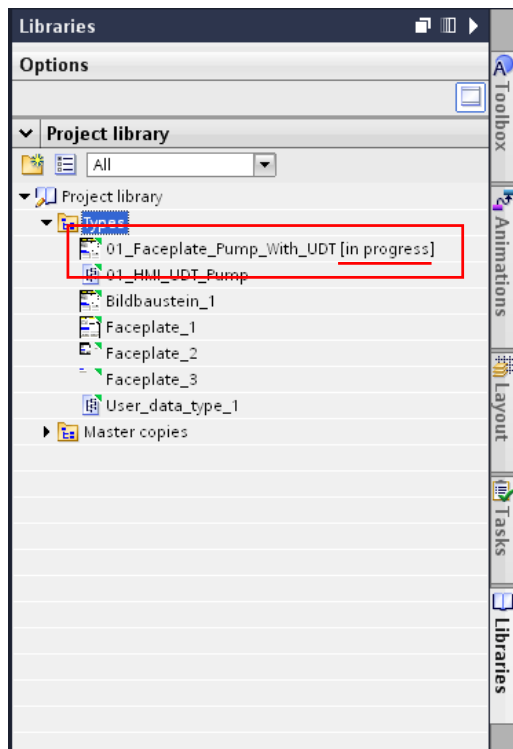
Status Bildbaustein

Wenn Sie einen Bildbaustein bearbeiten, dann erscheint neben dem Namen des Bildbausteins der folgende Eintrag "Name des Bildbausteins [in Arbeit]".

In der Kopfzeile des Bildbaustein Editors können Sie ebenfalls den aktuellen Status des geöffneten Bildbausteins sehen.

Über die "Statusangabe" [in Arbeit] können Sie erkennen, dass der vorliegende Bildbaustein bearbeitet wurde und diese Änderungen noch nicht freigegeben worden sind. Sehen Sie sich hierzu das Kapitel 2.3.2 an.

Abbildung 2-8



Was passiert mit den im Projekt verwendeten Bildbausteinen, wenn der zugehörige Bildbaustein aktuell bearbeitet wird?

Wenn Sie einen im Projekt verwendeten Bildbaustein bearbeiten, dann werden durchgeführte Änderungen an diesem Bildbaustein erst wirksam, wenn Sie den zugehörigen Bildbaustein explizit freigeben (siehe Kapitel 2.3.1).

Sie können z. B. die Bearbeitung eines Bildbausteins unterbrechen und die Projektierung speichern und schließen. Nach dem Öffnen der Projektierung können Sie die Bearbeitung des Bildbausteins fortsetzen. Der Status des bearbeiteten Bildbausteins steht dann immer noch auf "[in Arbeit]".

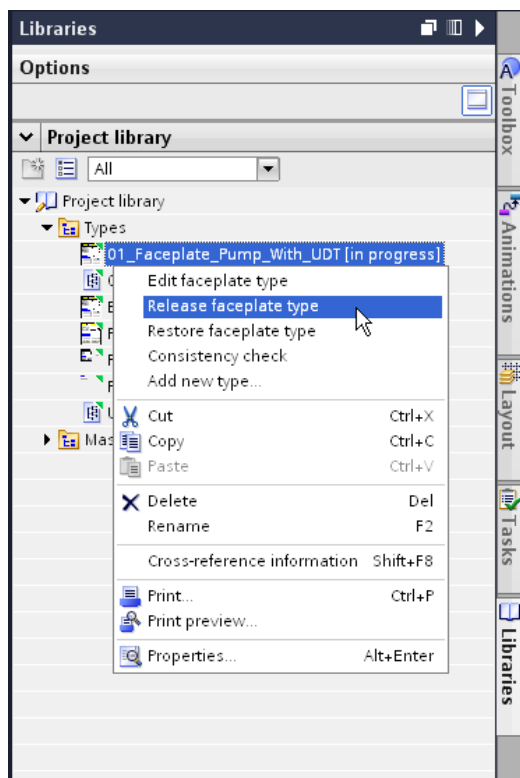
2.3.1 Bildbaustein freigeben

Mit dem Freigeben des Bildbausteins erhält ein Bildbaustein eine Versionsbezeichnung (siehe Kapitel 2.3.3). Nach der Freigabe werden alle Bildbausteine dieses Typs aktualisiert und verweisen auf die neue Version des Bildbausteins. Die alte Version wird gelöscht.

Um einen Bildbaustein freizugeben gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Bildbaustein, den Sie freigeben möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Bildbaustein-Typ freigeben" an. Der Bildbaustein wird aktualisiert.

Abbildung 2-9



Beispiel:

Sie verwenden einen Bildbaustein in mehreren Bildern. Nachträglich bearbeiten Sie diesen Bildbaustein und fügen eine weitere Schaltfläche hinzu. Solange Sie den bearbeiteten Bildbaustein nicht "freigeben" ändert sich an den vorhandenen Bildbausteinen in den Bildern nichts. Erst wenn Sie den bearbeiteten Bildbaustein "freigeben", wird die durchgeführte Änderung wirksam. Der in den Bildern verwendete Bildbaustein hat jetzt die hinzugefügte Schaltfläche.

2.3.2 Bildbaustein wiederherstellen

Sie können z. B. die Bearbeitung eines Bildbausteins unterbrechen und die Projektierung speichern und schließen. Nach dem Öffnen der Projektierung können Sie die Bearbeitung des Bildbausteins fortsetzen.

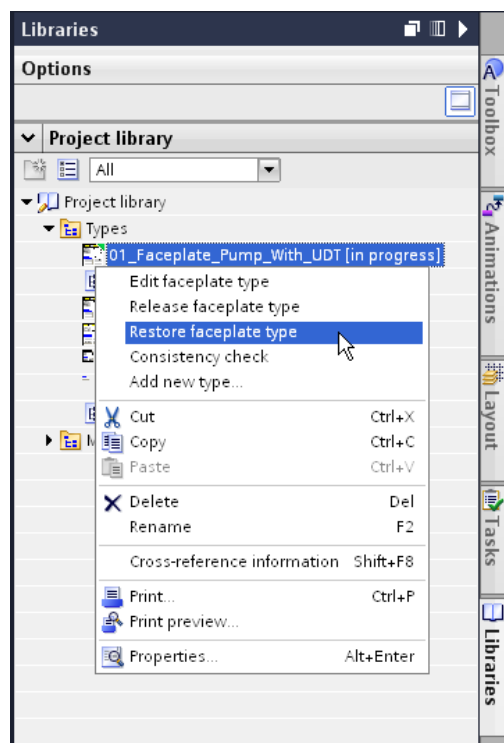
Wenn Sie jetzt den "alten" Stand des Bildbausteins wieder herstellen möchten, dann verwenden Sie die Funktion "Bildbaustein-Typ wiederherstellen".

Alle Änderungen, die seit der letzten Freigabe am Bildbaustein vorgenommen worden sind, werden verworfen. Der Bildbaustein ist anschließend wieder freigegeben.

Um einen Bildbaustein wiederherzustellen gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Bildbaustein, den Sie wiederherstellen möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Bildbaustein-Typ wiederherstellen" an. Der Änderungen werden verworfen. Der Bildbaustein ist wieder freigegeben.

Abbildung 2-10



Beispiel:

Sie führen Änderungen an einem Bildbaustein durch. Sie stellen fest, dass Sie den "falschen" Bildbaustein bearbeitet haben. Um den "alten" Stand wieder herzustellen verwenden Sie jetzt die Funktion "Bildbaustein-Typ wiederherstellen". Die Änderungen werden verworfen. Der in den Bildern zugehörige Bildbaustein wird nicht geändert.

2.3.3 Versionsbezeichnung eines Bildbausteins

Mit dem Freigeben eines Bildbausteins erhält der Bildbaustein eine Versionsbezeichnung. Nach der Freigabe werden alle Bildbausteine dieses Typs aktualisiert und verweisen auf die neue Version des Bildbausteins. Die alte Version wird gelöscht.

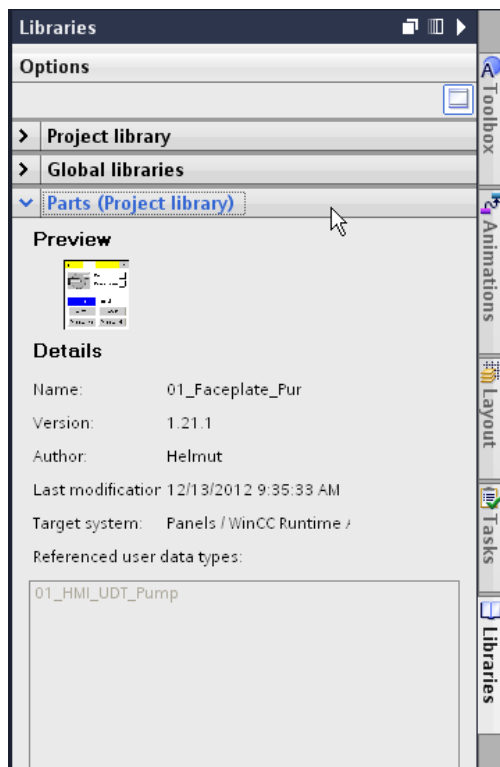
Die aktuelle Version eines Bildbausteins können Sie sich wie folgt anzeigen lassen.

Voraussetzung:

Ein Bildbaustein ist im Bildbaustein Editor geöffnet.

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Teile (Projektbibliothek)".
- Die Daten des aktuell in der Bearbeitung befindlichen Bildbausteins werden angezeigt.

Abbildung 2-11



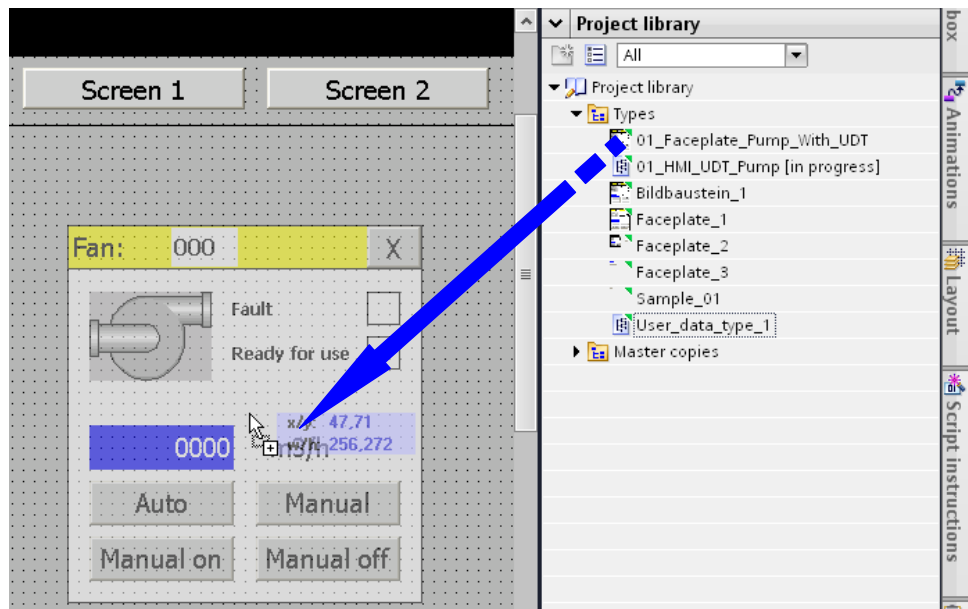
2.4 Bildbaustein aufrufen bzw. in ein Bild einfügen

Bildbausteine können über die Task Card "Bibliotheken" aufgerufen und in ein Bild eingefügt werden.

Um einen Bildbaustein in ein Bild einzufügen gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Öffnen Sie den Ordner "Typen".
In dem Ordner "Typen" werden Ihnen die verfügbaren Bildbausteine angezeigt.
- Markieren Sie den gewünschten Bildbaustein und ziehen Sie diesen per "Drag&Drop" in das HMI Bild hinein.

Abbildung 2-12



2.5 Bildbaustein Editor

Die nachfolgende Abbildung zeigt den Bildbaustein Editor, in dem Sie den Bildbaustein erstellen und bearbeiten.

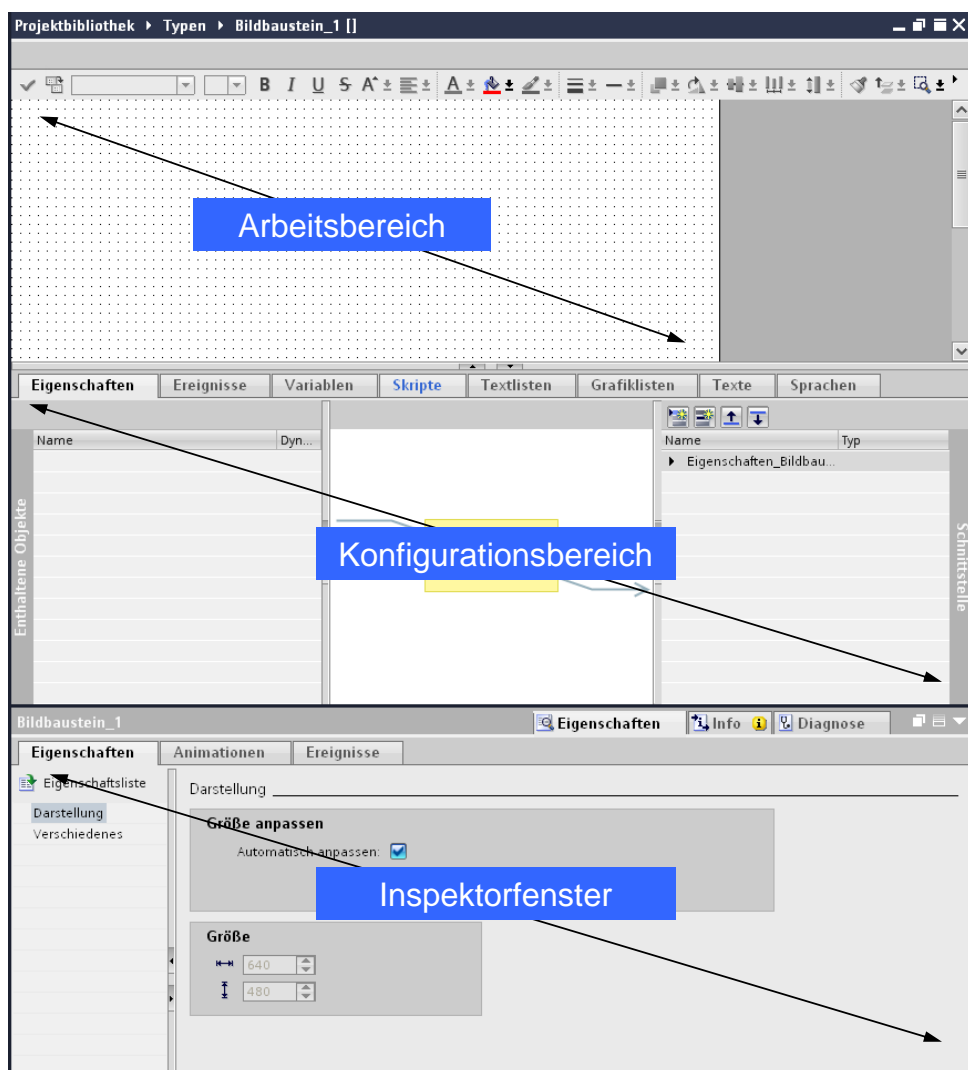
Im weiteren Verlauf der Applikation werden die einzelnen Bereiche wie

- der Arbeitsbereich
- der Konfigurationsbereich
- das Inspektorfenster

näher beschrieben.

An mehreren Beispielen wird gezeigt, welche Auswirkung die projektierte Funktion auf den erstellten Bildbaustein hat.

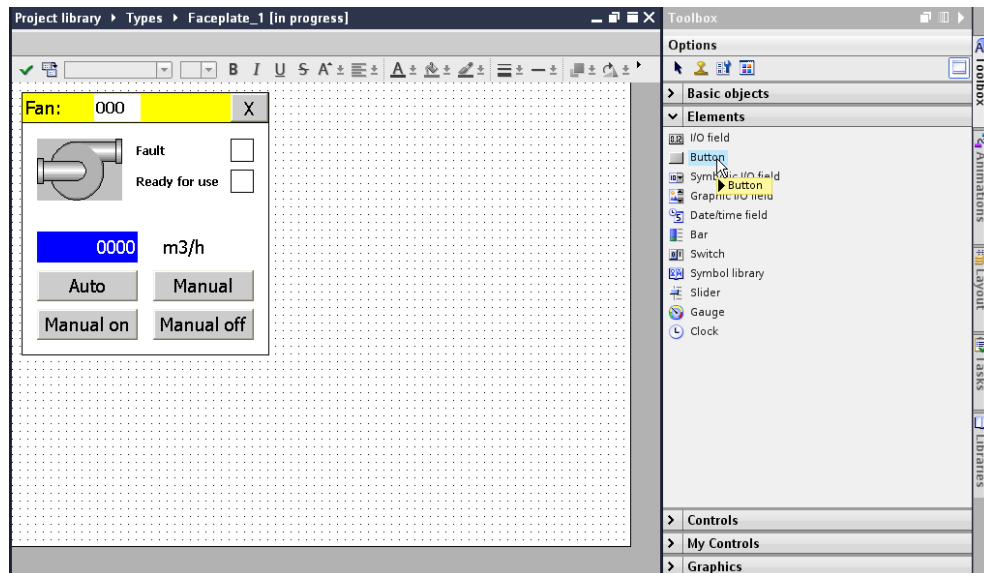
Abbildung 2-13



2.5.1 Arbeitsbereich

Im Arbeitsbereich platzieren Sie die Objekte für den Bildbaustein, wie sie es vom Editor "Bilder" gewohnt sind. Sie entfernen Objekte oder fügen, aus der Task Card "Werkzeuge", neue Objekte in den Arbeitsbereich ein.

Abbildung 2-14



Tipp:

Wenn Sie ein Objekt eingefügt haben, dann vergeben Sie dem Objekt gleich einen Namen, über den Sie das Objekt leicht identifizieren können. Diese Vorgehensweise erleichtert die Arbeit bei der nachfolgenden Parametrierung.

Beispiel:

Sie haben eine Schaltfläche eingefügt. Das System vergibt dieser Schaltfläche automatisch einen Namen (z. B. "Button_1").

Ändern Sie den Namen z. B. von "Button_1" auf "Button_Automatic_Mode" ab.

Öffnen Sie hierzu im "Inspektorfenster" die Eigenschaften der Schaltfläche. Unter dem Register "Eigenschaften > Eigenschaftsliste > Verschiedenes" können Sie den Namen des Objektes anpassen.

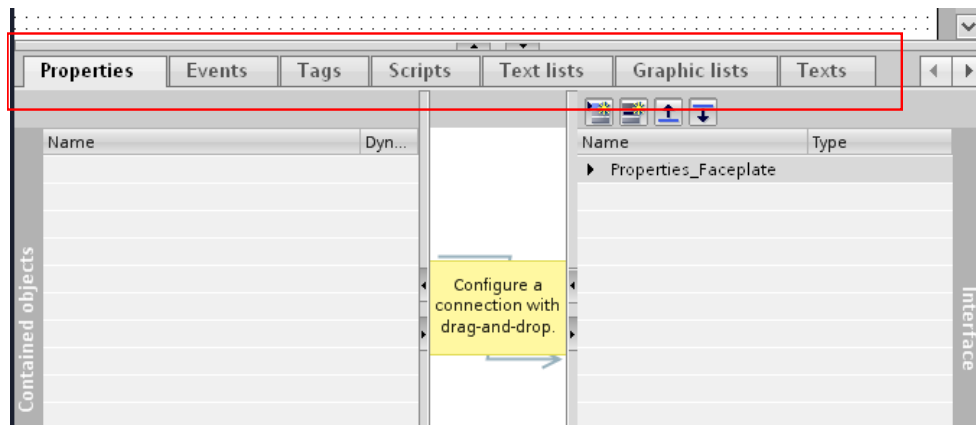
Im weiteren Verlauf dieser Applikation werden Sie erkennen, wofür diese Vorgehensweise sinnvoll ist.

2.5.2 Konfigurationsbereich

Im Konfigurationsbereich befinden sich verschiedene Registerkarten.

Details zu den einzelnen Registerkarten werden im Kapitel 2.7 beschrieben.

Abbildung 2-15



2.5.3 Inspektorfenster

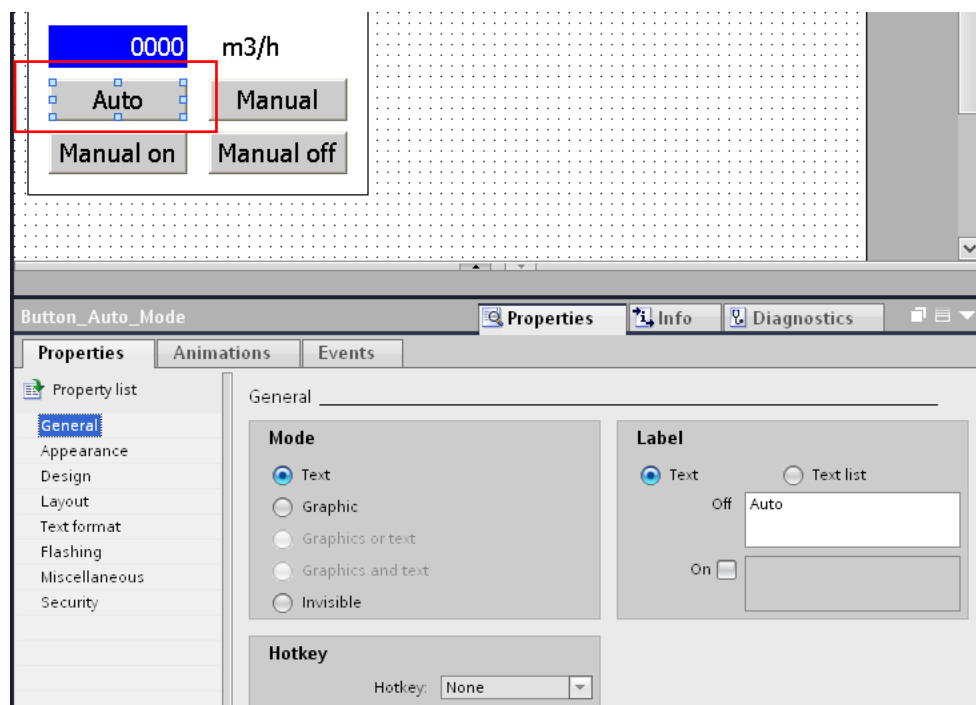
Im Inspektorfenster werden die Eigenschaften des selektierten Objekts angezeigt. Es entspricht derselben Ansicht, wie Sie es vom Editor "Bilder" bereits kennen.

Eigenschaften, Animationen und Ereignisse, die Sie hier im Bildbaustein an den Objekten projektieren, können im Editor "Bilder" nicht bearbeitet werden.

Im Gegensatz zum "Konfigurationsbereich" (siehe vorheriges Kapitel), bei dem vorgegebene Eigenschaften, Animationen und Ereignisse später dem Bildbaustein als "Parameter" übergeben bzw. geändert werden konnten.

Einstellungen im Inspektorfenster bieten sich immer dann an, wenn Sie einem Objekt z. B. einen festgelegten Text, Layout oder Funktion usw. zuweisen möchten.

Abbildung 2-16



Beispiel 1 (Text zuweisen):

Sie haben eine Schaltfläche für den "Automatikbetrieb" projiziert. Der angezeigte Text der Schaltfläche soll immer "Auto" heißen.

In diesem Fall geben Sie den Namen der Schaltfläche im Inspektorfenster unter dem Register "Eigenschaften > Eigenschaftsliste > Verschiedenes" vor.

- **Ergebnis:**
Der Text "Auto" ist der Schaltfläche fest zugeordnet und kann später am Bildbaustein unter dem Register "Schnittstelle" nicht geändert werden.
- **Alternative:**
Alternativ haben Sie die Möglichkeit, im Konfigurationsbereich den Namen der Schaltfläche als "statische" Eigenschaft zu parametrieren. Am Bildbaustein unter dem Register "Schnittstelle" könnten Sie dann später den Namen individuell anpassen.

Beispiel 2 (Funktion zuweisen):

Sie haben eine Schaltfläche für den "Automatikbetrieb" projiziert.

Unter der Eigenschaft "Drücken" haben Sie im Inspektorfenster die Funktion "SetzeBit" zugewiesen.

Die zugehörige Variable haben Sie als "Parameter" im Konfigurationsfenster projiziert.

- **Ergebnis:**

Beim "Drücken" der Schaltfläche wird die Funktion "SetzeBit" ausgeführt.

Die Funktion kann später am Bildbaustein unter dem Register

"Schnittstelle" nicht geändert werden.

Die zugehörige Variable kann der verwendeten Funktion jedoch individuell vorgegeben werden.

- **Alternative:**

Alternativ haben Sie z. B. die Möglichkeit, der Schaltfläche im Konfigurationsbereich das Ereignis "Drücken" zu parametrieren.

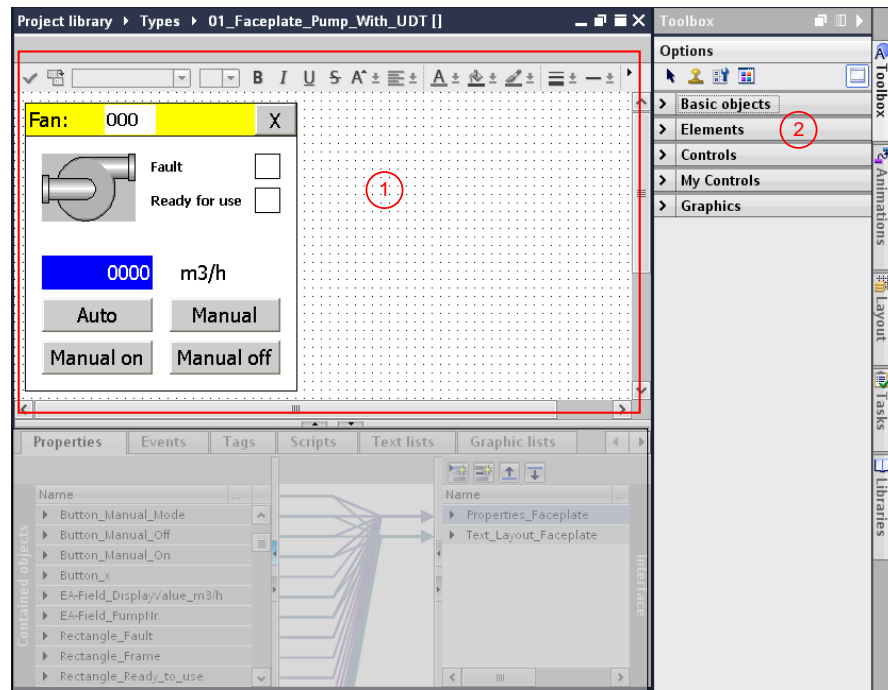
Am Bildbaustein unter dem Register "Ereignisse" können Sie dann der Schaltfläche an dem Ereignis "Drücken" individuell eine Systemfunktion zuweisen. Sie sind also nicht an eine feste Funktion wie z. B. "SetzeBit" gebunden.

2.6 Parameter im Arbeitsbereich

Der Arbeitsbereich des Bildbaustein Editors beinhaltet die gleichen Parameter und Einstellungsmöglichkeiten, wie Sie es vom Editor "Bilder" her kennen (1).

In erster Linie benötigen Sie hier die Task Card "Werkzeuge" und die darin enthaltenen Paletten "Basisobjekte" und "Elemente" (2).

Abbildung 2-17

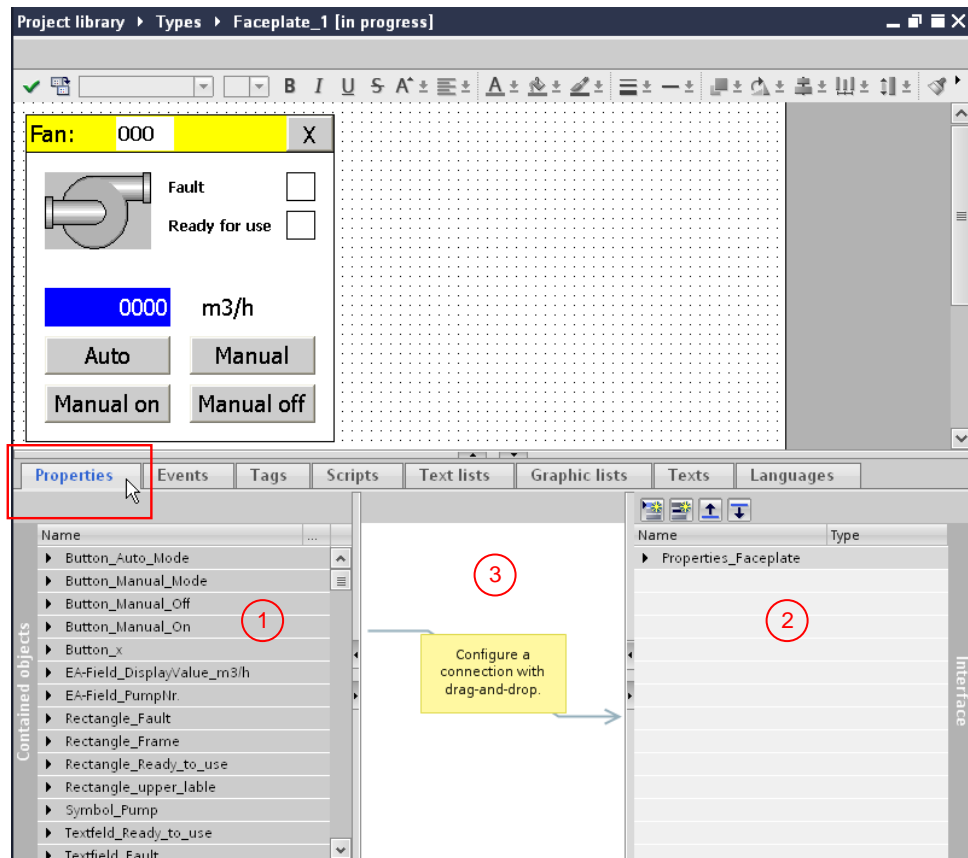


2.7 Parameter im Konfigurationsbereich

Nachfolgend werden die einzelnen Registerkarten im Konfigurationsbereich beschrieben.

2.7.1 Registerkarte "Eigenschaften"

Abbildung 2-18



Die Registerkarte "Eigenschaften" enthält zwei Listen sowie einen "grafischen Bereich".

- Liste "Enthaltene Objekte" (1)
- Liste "Schnittstelle" (2)
- "Grafischer" Bereich (3)

Liste "Enthaltene Objekte" (1)

Die Liste "Enthaltene Objekte" enthält alle im Arbeitsbereich enthaltenen Objekte. Jedes dieser Objekte besitzt objektspezifische Eigenschaften, welche Sie bereits aus dem Editor "Bilder" kennen (eine Schaltfläche hat andere Eigenschaften als z. B. ein Rechteck).

Über das Symbol neben den Objekten zum "Öffnen/Schließen" der Eigenschaft können Sie durch die Eigenschaften der Objekte navigieren.

Eigenschaften der Objekte

In der Spalte "Dynamisierungen" (2) können Sie an den Symbolen erkennen, ob die verfügbare Eigenschaft eine "statische" bzw. eine "dynamische" Eigenschaft besitzt.



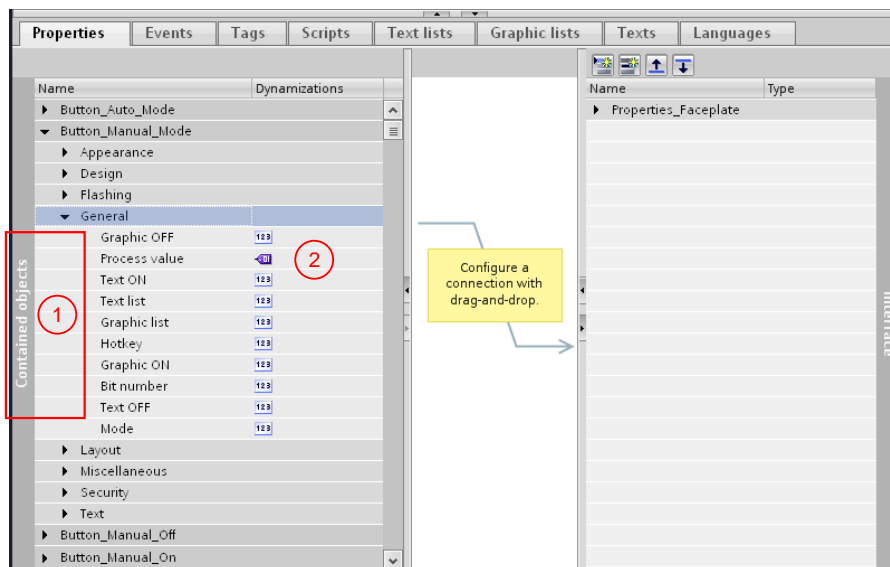
- Dynamische Eigenschaften (zu erkennen an dem Symbol ):
Werte können dem Bildbaustein später als Variable bzw. über eine Text- bzw. Grafikliste vorgegeben werden.
- Statische Eigenschaften (zu erkennen an dem Symbol ):
Werte werden dem Bildbaustein später als konstanter Wert vorgegeben.
Ausnahme:
Bei einer Kurvenanzeige werden hierüber die zu parametrierenden Kurven vorgegeben.

Abbildung 2-19



Das nachfolgende Beispiel soll dies verdeutlichen.

2 Funktionen und Eigenschaften des Bildbausteins

2.7 Parameter im Konfigurationsbereich

Beispiel: EA-Feld in einem Bildbaustein

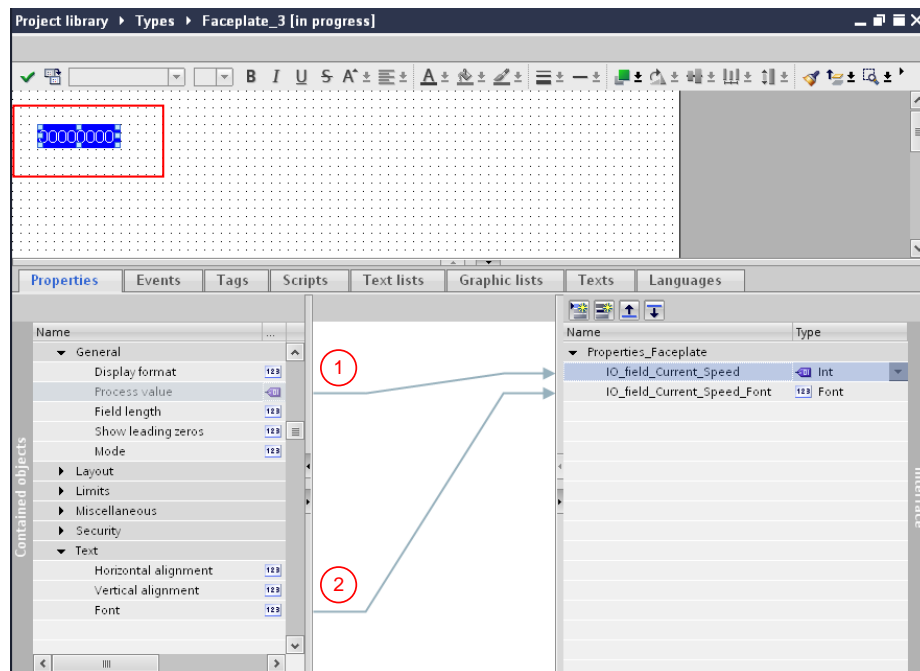
Die Eigenschaft "Prozesswert" hat eine "dynamische" Eigenschaft (1).

Durch die Parametrierung kann dem EA-Feld am Bildbaustein unter dem Register "Schnittstelle" individuell eine Variable zugewiesen werden.

Die Eigenschaft "Schriftart" hat eine "statische" Eigenschaft (2).

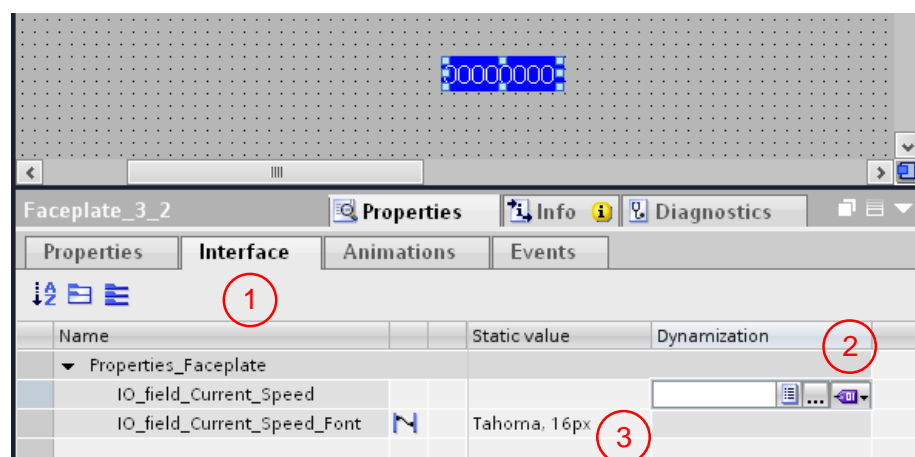
Durch die Parametrierung kann dem EA-Feld am Bildbaustein unter dem Register "Schnittstelle" die Schriftgröße individuell zugewiesen werden.

Abbildung 2-20



Die nachfolgende Abbildung zeigt den erstellten Bildbaustein (in diesem Fall nur ein EA-Feld), der im Editor "Bilder" aufgerufen wurde.

Abbildung 2-21



Unter dem Register "Schnittstelle" (1) sehen Sie die zwei parametrierten Eigenschaften des Bildbausteins. Der "dynamischen" Schnittstelle kann eine Variable zugewiesen werden (2). Der "statischen" Schnittstelle kann nur ein fester Wert zugewiesen werden (3).

Liste "Schnittstelle"

Ein Bildbaustein ist zunächst ein in sich geschlossenes "Bild-Objekt". Von "außen" hat man keinen Zugriff auf die im Bildbaustein verwendeten Anzeige- und Bedienobjekte.

In der Liste "Schnittstelle" werden ausgewählte Eigenschaften, Animationen und Ereignisse von den verwendeten Anzeige- und Bedienobjekten aufgeführt. Über diese festgelegten Eigenschaften und Variablen erfolgt dann im Editor "Bilder" die Parametrierung des Bildbausteins.

Sie können **eine** Eigenschaft **einer** erstellten Eigenschaft bzw. Variablen in der Liste "Schnittstelle" zuweisen.

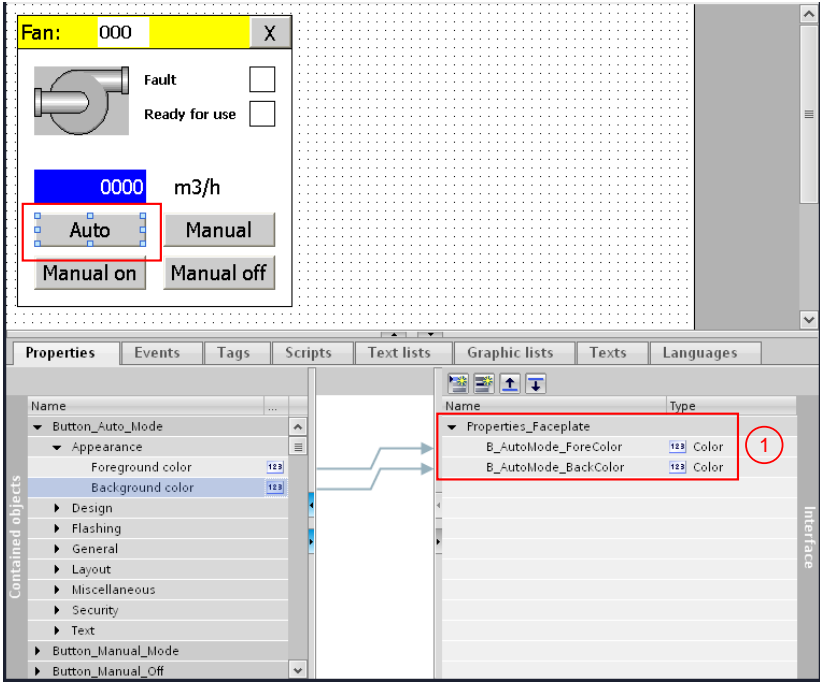
Es ist aber auch möglich, **mehrere** Eigenschaften **einer** erstellten Eigenschaft bzw. Variablen in der Liste "Schnittstelle" zuzuweisen.

Die nachfolgenden Beispiele sollen die Schnittstellenfunktionalität verdeutlichen.

Beispiel 1: Schaltfläche, Farbe parametrierbar gestalten

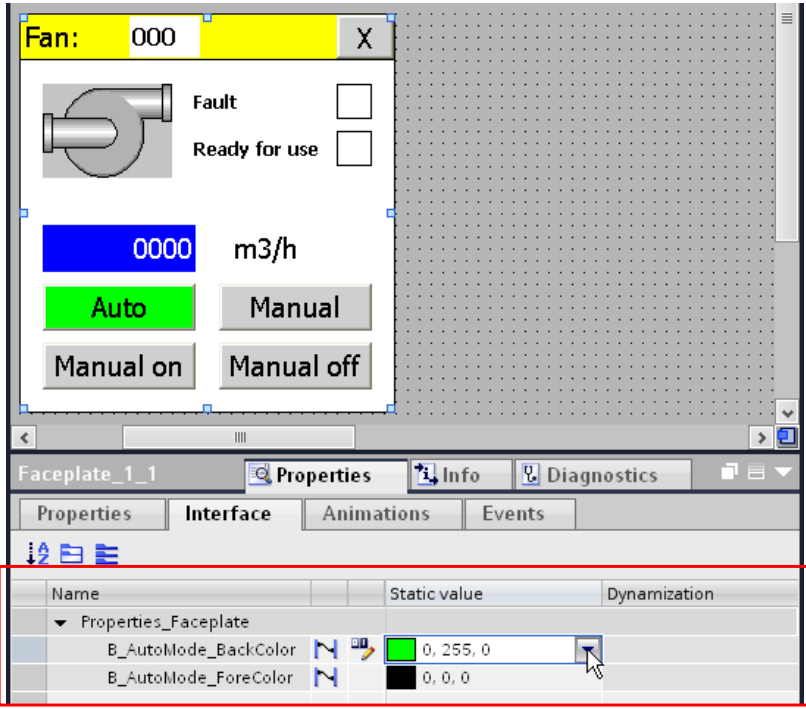
Die Vordergrund- und Hintergrundfarbe der Schaltfläche "Auto" soll parametrierbar sein.

Tabelle 2-1

Nr.	Aktion
1.	<p>Projektieransicht:</p> <p>Aus der Liste "Enthaltene Objekte" werden von der Schaltfläche "Button_Auto_Mode" die ausgewählten Eigenschaften via "Drag&Drop" mit den erstellten "Eigenschaften" in der Liste "Schnittstelle" verbunden (1). Der Datentyp wird automatisch vorgegeben.</p> <p>Die "Eigenschaften" unter der Liste "Schnittstelle" wurden zuvor angelegt.</p> 

2 Funktionen und Eigenschaften des Bildbausteins

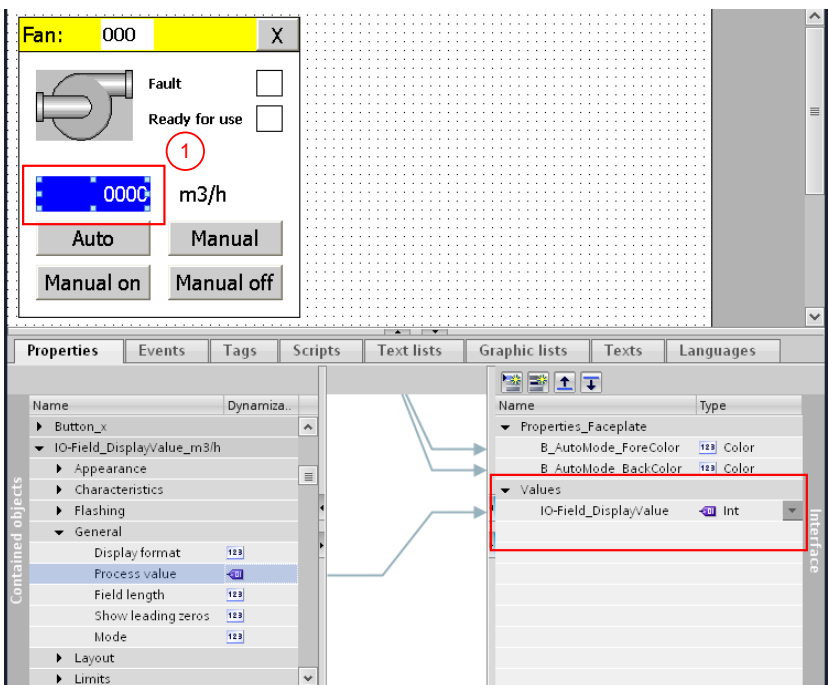
2.7 Parameter im Konfigurationsbereich

Nr.	Aktion												
2.	<p data-bbox="316 309 836 338">Ansicht des Bildbausteins im Editor "Bilder":</p> <p data-bbox="316 371 1267 432">Über das Register "Schnittstelle" kann der Schaltfläche "Auto" über die zwei projizierten Eigenschaften die Farbe vorgegeben werden.</p>  <table border="1" data-bbox="331 1037 1139 1178"><thead><tr><th>Name</th><th>Static value</th><th>Dynamization</th></tr></thead><tbody><tr><td>Properties_Faceplate</td><td></td><td></td></tr><tr><td> B_AutoMode_BackColor</td><td>0, 255, 0</td><td></td></tr><tr><td> B_AutoMode_ForeColor</td><td>0, 0, 0</td><td></td></tr></tbody></table>	Name	Static value	Dynamization	Properties_Faceplate			B_AutoMode_BackColor	0, 255, 0		B_AutoMode_ForeColor	0, 0, 0	
Name	Static value	Dynamization											
Properties_Faceplate													
B_AutoMode_BackColor	0, 255, 0												
B_AutoMode_ForeColor	0, 0, 0												

Beispiel 2: A-Feld, Variablenzuweisung

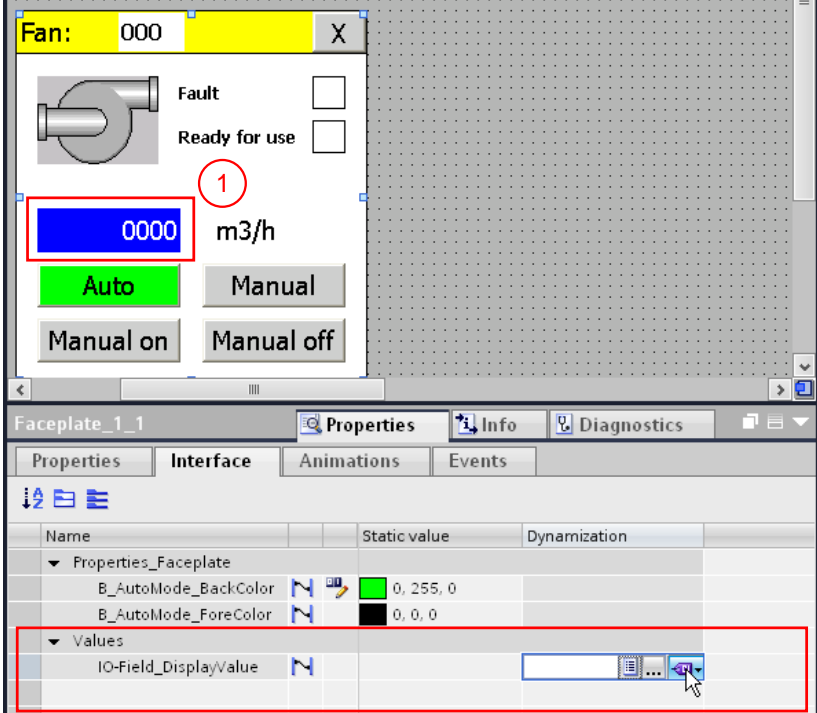
Die Variable des EA-Feldes für die Ausgabe des aktuellen Wertes für den Durchfluss soll parametrierbar sein (1).

Tabelle 2-2

Nr.	Aktion
1.	<p>Projektieransicht:</p> <p>Aus der Liste "Enthaltene Objekte" wird vom EA-Feld "IO-Field_DisplayValue_m3/h" die ausgewählte Eigenschaft via "Drag&Drop" mit der in der Liste "Schnittstelle" erstellten "Variablen" verbunden. Der Datentyp muss entsprechend vorgegeben / angepasst werden. Die "Variable" unter der Liste "Schnittstelle" wurde zuvor angelegt.</p> 

2 Funktionen und Eigenschaften des Bildbausteins

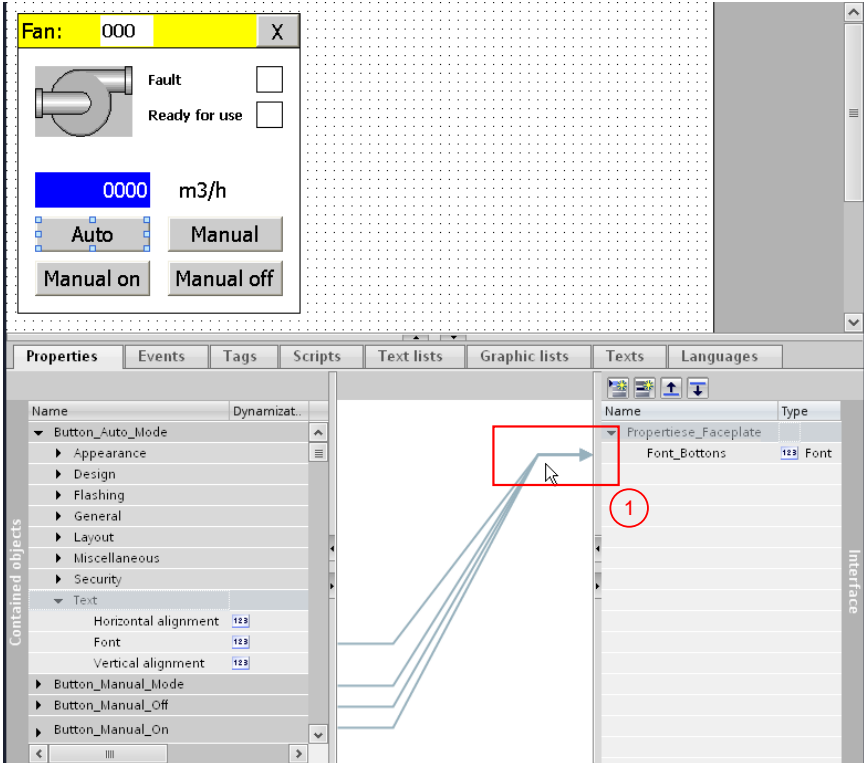
2.7 Parameter im Konfigurationsbereich

Nr.	Aktion
2.	<p data-bbox="316 309 837 338">Ansicht des Bildbausteins im Editor "Bilder":</p> <p data-bbox="316 371 1345 432">Über das Register "Schnittstelle" kann dem EA-Feld für die Ausgabe des aktuellen Durchflusses über die Variable "IO-Field_DisplayValue" eine HMI-Variable zugewiesen werden (1).</p> 

Beispiel 3: Mehrfachzuweisung

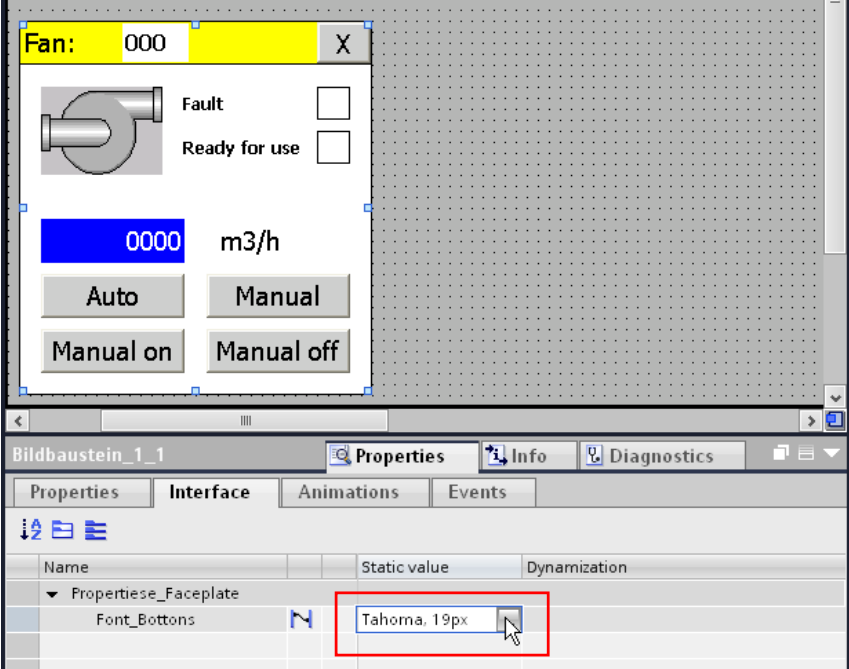
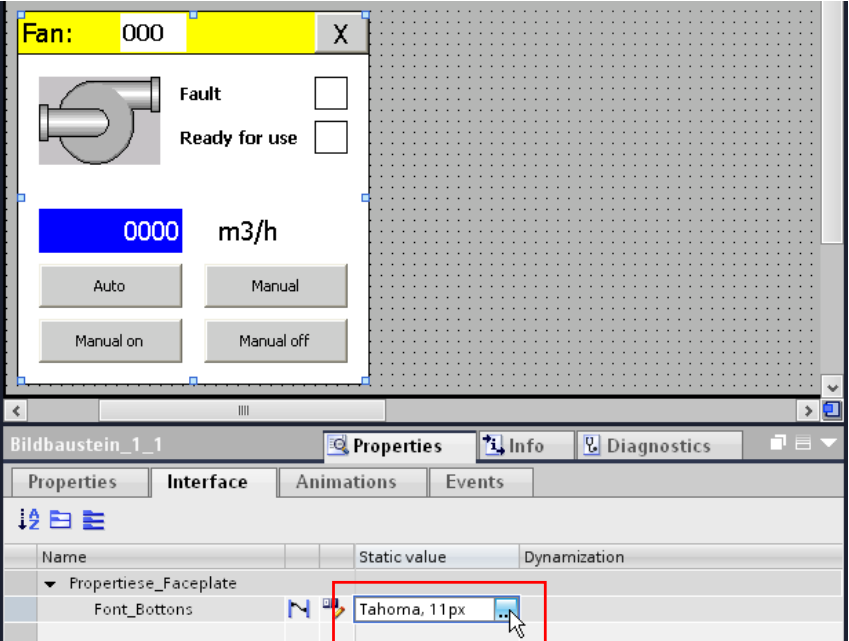
Das Schriftformat der vier Schaltflächen soll parametrierbar sein.

Tabelle 2-3

Nr.	Aktion
1.	<p>Projektieransicht:</p> <p>Aus der Liste "Enthaltene Objekte" wird die ausgewählte Eigenschaft der vier Schaltflächen via "Drag&Drop" mit einer erstellten "Eigenschaft" in der Liste "Schnittstelle" verbunden (1). Der Datentyp wird automatisch vorgegeben.</p> <p>Die "Eigenschaft" unter der Liste "Schnittstelle" wurde zuvor angelegt.</p> 

2 Funktionen und Eigenschaften des Bildbausteins

2.7 Parameter im Konfigurationsbereich

Nr.	Aktion
2.	<p>Ansicht des Bildbausteins im Editor "Bilder":</p> <p>Über das Register "Schnittstelle" kann den Schaltflächen über die "Eigenschaft" "Font_Bottoms" das Schriftformat vorgegeben werden. In diesem Beispiel: "Tahoma 19px"</p>  <p>The screenshot shows a graphical user interface for a fan control. At the top, there is a yellow bar with the text 'Fan: 000' and a close button 'X'. Below this, there is a fan icon, two checkboxes labeled 'Fault' and 'Ready for use', a digital display showing '0000' and 'm3/h', and four buttons: 'Auto', 'Manual', 'Manual on', and 'Manual off'. The configuration window below the main interface shows the 'Font_Bottoms' property set to 'Tahoma, 19px'.</p>
3.	<p>Zum Vergleich.</p> <p>In diesem Fall wurde das Schriftformat "Tahoma 11px" gewählt.</p>  <p>The screenshot shows the same graphical user interface as in step 2. However, the configuration window below the main interface shows the 'Font_Bottoms' property set to 'Tahoma, 11px'.</p>

Eigenschaften in der Liste "Schnittstelle" festlegen:

Die nachfolgende Abbildung zeigt die Liste "Schnittstelle" und die zugehörigen Parameter.

Allgemein:

Abhängig von dem hinzugefügten Ereignis und dem daraus resultierenden "Typ" spricht man in der Liste "Schnittstelle" von einer



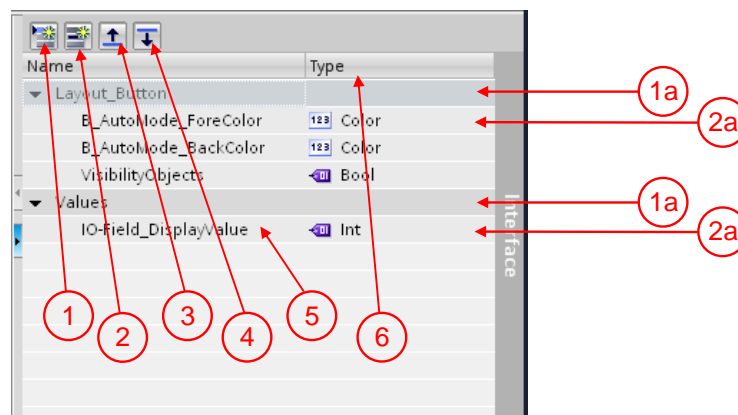
- Eigenschaft
 - Eigenschaft (zu erkennen an dem Symbol ).
 - Werte werden dem Bildbaustein später als fester Wert vorgegeben.
- Variablen
 - Variable (zu erkennen an dem Symbol ).
 - Werte können dem Bildbaustein später als Variable bzw. über eine Text- bzw. Grafikliste vorgegeben werden.

Abbildung 2-22



Namen anpassen:

Wenn Sie den Namen einer Kategorie, Eigenschaft bzw. Variablen bearbeiten möchten, dann markieren Sie mit der rechten Maustaste den Namen, der geändert werden soll. Es öffnet sich ein Dialogfeld. Wählen Sie in dem Dialogfeld die Option aus, die Sie durchführen möchten.

"Kategorie hinzufügen" (1)

- Um eine neue Kategorie anzulegen klicken Sie auf die Schaltfläche "Kategorie hinzufügen" (1). Sie haben mit dieser Funktion die Möglichkeit die verwendeten Eigenschaften bzw. Variablen "thematisch" zu gliedern (1a). Passen Sie gegebenenfalls hierzu den Namen an. Der Name wird am Bildbaustein unter dem Register "Schnittstelle" angezeigt.

"Eigenschaft bzw. Variable hinzufügen" (2)

- Um eine neue Eigenschaft bzw. Variable anzulegen klicken Sie auf die Schaltfläche "Eigenschaft hinzufügen" (2). Der Name wird am Bildbaustein unter dem Register "Schnittstelle" angezeigt.

Tip: Vergeben Sie einen Namen über den Sie die vorgegebene Eigenschaft bzw. Variable leicht wiedererkennen können (2a).

Beispiel: Anstelle von "Button_01" besser "Button_AutoMode".

2.7 Parameter im Konfigurationsbereich

Reihenfolge anpassen (3), (4)

- Über die Schaltflächen können Sie die Reihenfolge der vorgegebenen Kategorien, Eigenschaften und Variablen nachträglich anpassen.

Name (5)

- Name der hinzugefügten Kategorie, Eigenschaft bzw. Variablen.

Type (6)

- Wählen Sie hierüber den Datentyp aus.

Hinweis:

Wenn Sie "statische Werte" aus der Liste "Enthaltene Objekte" mit der Liste "Schnittstelle" per "Drag&Drop" verbinden, wird der Datentyp automatisch zugewiesen. Wenn Sie "Eigenschaften" aus dem Inspektorfenster der Liste "Schnittstelle" zuweisen, dann müssen Sie den zugehörigen Datentyp vorgeben bzw. an Ihre Vorgabe anpassen.

"Grafischer" Bereich

Der grafische Bereich zeigt die projektierten Verbindungen zwischen der Liste "Enthaltene Objekte" und der Liste "Schnittstelle" an. Des Weiteren werden Funktionszuordnungen, die zwischen dem Inspektorfenster und der Liste "Schnittstelle" vorgenommen worden sind, angezeigt.

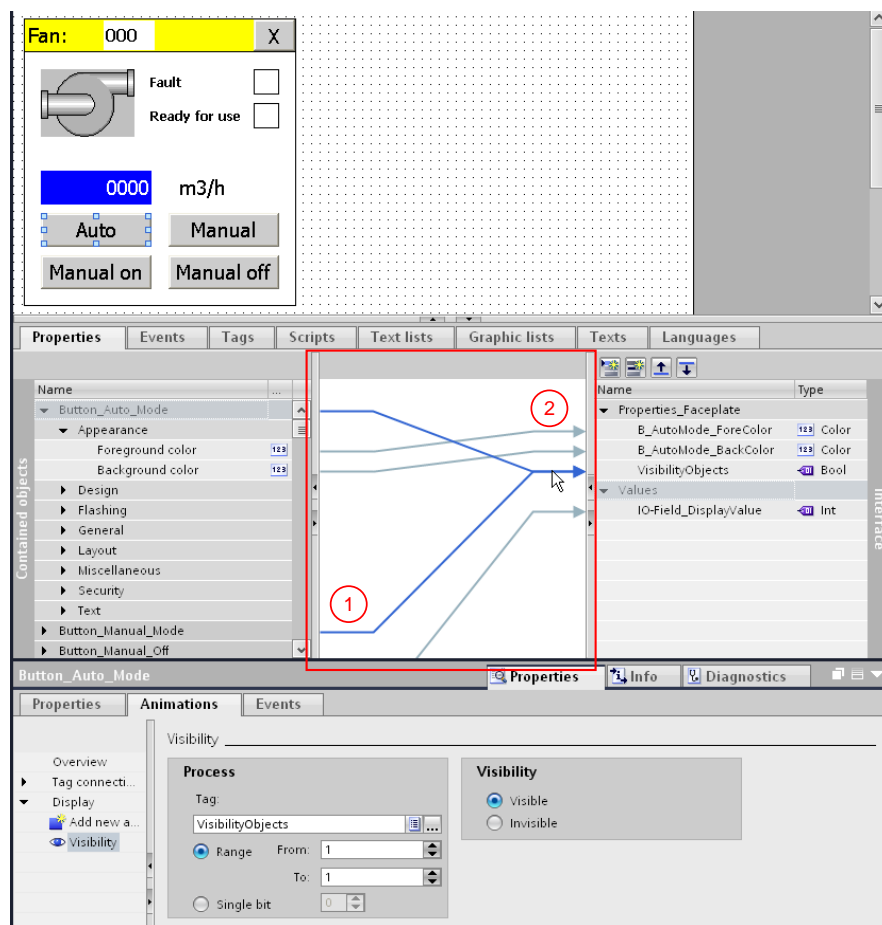
Durch Anklicken einer Linie können Sie erkennen, welche Eigenschaften miteinander verbunden sind. Dies erleichtert die Zuordnung untereinander, insbesondere bei umfangreichen Zuweisungen.

Beispiel:

An den Schaltflächen "Hand" und "Auto" ist im Inspektorfenster die Funktion "Sichtbarkeit" projektiert. Die zugehörige Variable ist in der Liste "Schnittstelle" hinterlegt. Durch diese Zuweisung erfolgt automatisch die grafische Linienzuordnung direkt an den "Namen" der Schaltfläche (1).

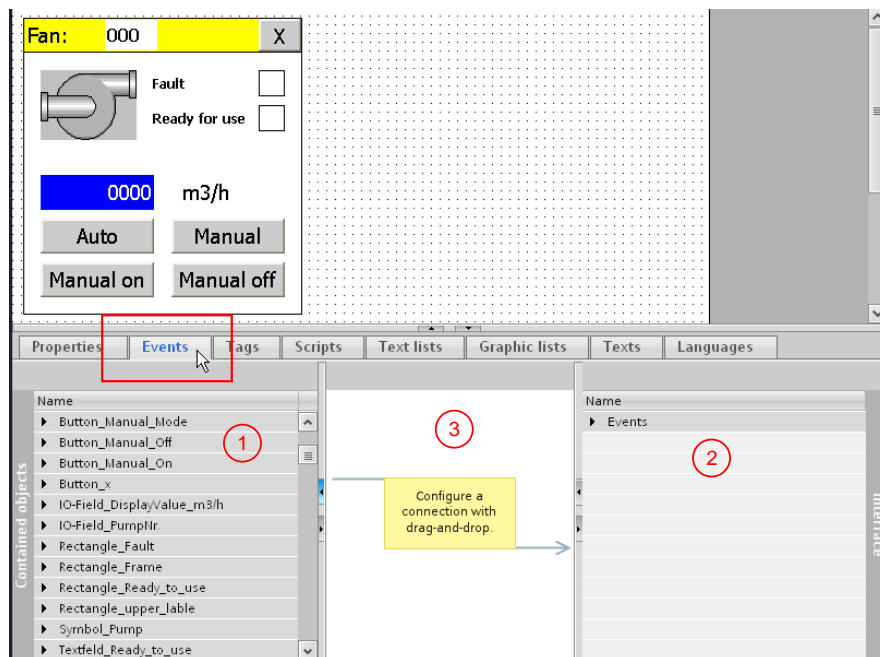
Im Gegensatz zur grafischen Linienzuordnung zwischen der Liste "Enthaltene Objekte" und der Liste "Schnittstelle". Hier erfolgt eine direkte grafische Linienzuordnung zwischen der verwendeten "Eigenschaft" aus der Liste "Enthaltene Objekte" und der zugehörigen "Eigenschaft" aus der Liste "Schnittstelle" (2).

Abbildung 2-23



2.7.2 Registerkarte "Ereignisse"

Abbildung 2-24



Die Registerkarte "Ereignisse" enthält, wie zuvor die Registerkarte "Eigenschaften", zwei Listen und einen "grafischen" Bereich.

- Liste "Enthaltene Objekte" (1)
- Liste "Schnittstelle" (2)
- "Grafischer" Bereich (3)

Liste "Enthaltene Objekte"

Mit Aufruf der Registerkarte "Ereignisse" ändern sich in der Liste "Enthaltene Objekte" die Eigenschaften der Objekte.

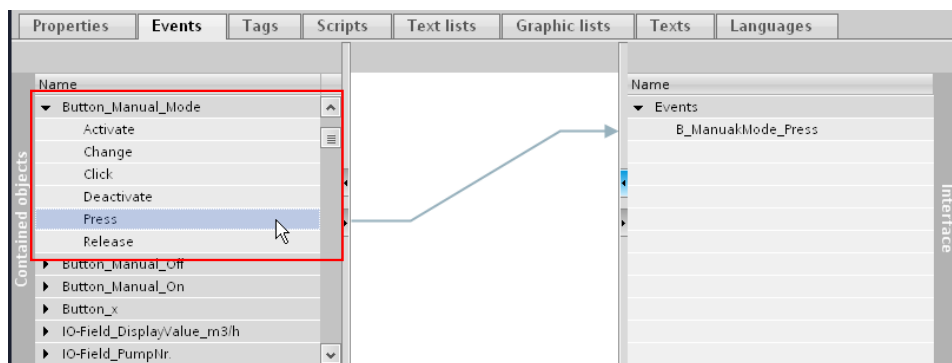
An Objekten wie einer Schaltfläche" stehen Ihnen die bekannten Ereignisse wie "Drücken", "Klicken", "Loslassen" usw. zur Verfügung. Des Weiteren ändert sich die Ansicht in der Liste "Schnittstelle".

Über das Symbol zum "öffnen/schließen" der Objekte können Sie durch die Eigenschaften der Objekte navigieren.

In dem gezeigten Beispiel sind die Namen der einzelnen Objekte zuvor individuell angepasst / vergeben worden. Durch diese angepasste Namensvergabe können die verwendeten Objekte leicht identifiziert werden.

Die nachfolgende Abbildung zeigt einen Detailausschnitt von den möglichen "Ereignissen" der Schaltfläche "Button_Manual_Mode".

Abbildung 2-25



Beispiel:

Als Ereignis wurde an die Schaltfläche "Button_Manual_Mode" die Funktion "Drücken" gewählt und der Liste "Schnittstelle" zugewiesen.

Am Bildbaustein kann im Editor "Bilder" unter dem Register "Ereignisse" individuell eine Funktion zugewiesen werden. Die Funktion wird beim "Drücken" der Schaltfläche "Auto" durchgeführt. Sehen Sie sich hierzu die nachfolgende Beschreibung der Liste "Schnittstelle" an.

Liste "Schnittstelle"

Die Liste "Schnittstelle" dient dazu, ausgewählte Ereignisse aus der Liste "Enthaltene Objekte" wie "Drücken", "Klicken", "Loslassen" usw. über eine "Schnittstelle" nach "außen" zu führen.

Das heißt.

Sie haben einen Bildbaustein im Editor "Bilder" aufgerufen. Die in dem Bildbaustein enthaltenen Bedienobjekte sollen Funktionen zugewiesen werden. Damit die Funktionen den im Bildbaustein enthaltenen Bedienobjekten zugewiesen werden können, dienen die in der Liste "Schnittstelle" parametrisierten Eigenschaften.

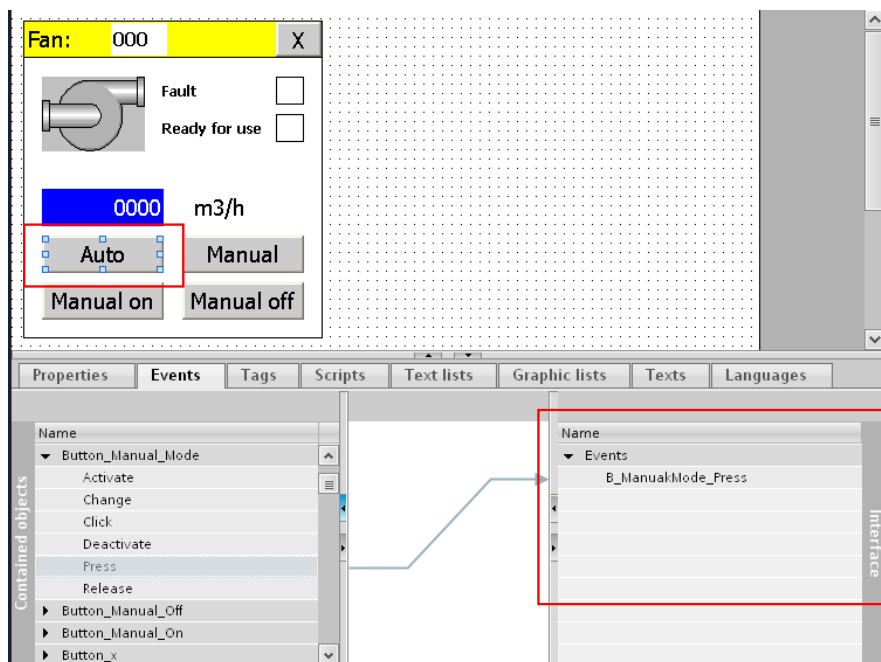
Beispiel:

An die Schaltfläche "Button_Manual_Mode" soll beim Ereignis "Drücken" eine Funktion parametrierbar sein.

Projektieransicht:

Aus der Liste "Enthaltene Objekte" wird das ausgewählte Ereignis "Drücken" via "Drag&Drop" der Liste "Schnittstelle" zugewiesen.

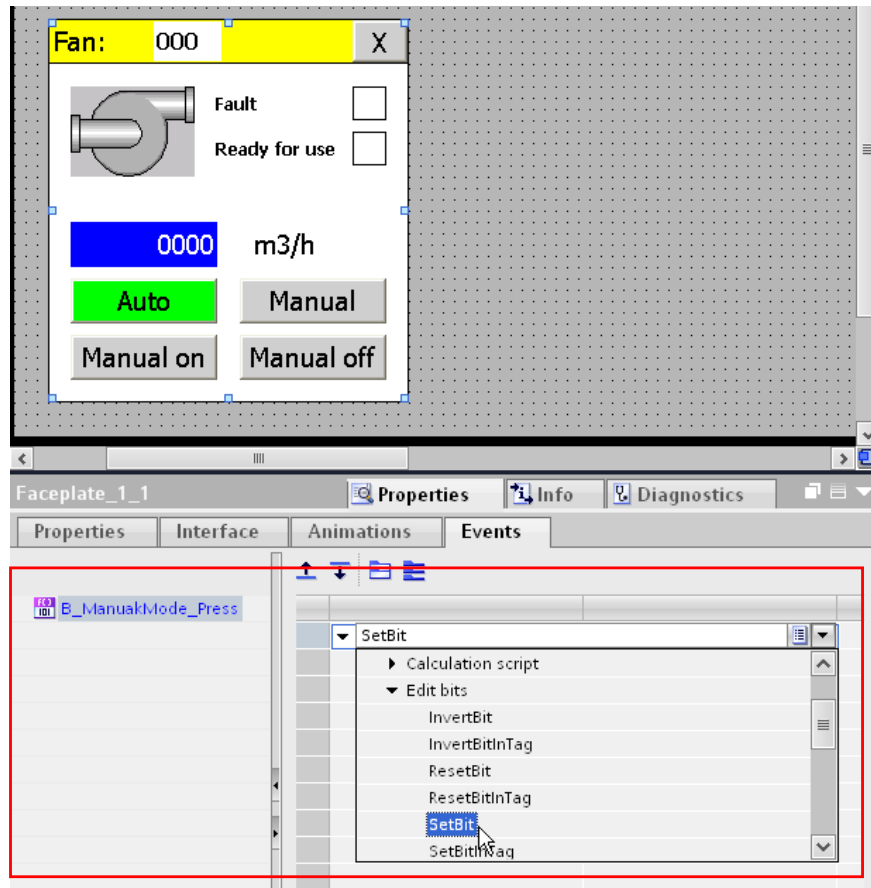
Abbildung 2-26



Ansicht des Bildbausteins im Editor "Bilder":

Über die Register "Ereignisse" kann eine beliebige Funktion an die Schaltfläche "Hand" vergeben werden. In diesem Fall "SetzeBit".

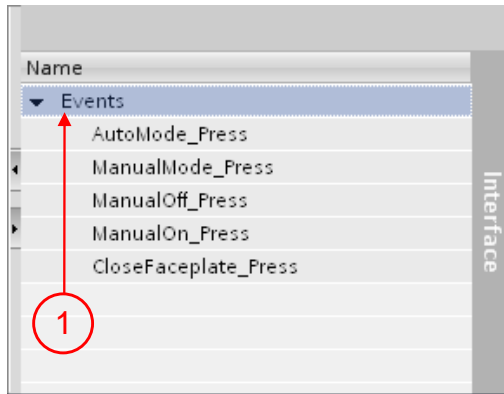
Abbildung 2-27



Eigenschaften in der Liste "Schnittstelle" festlegen:

Die nachfolgende Abbildung zeigt die Liste "Schnittstelle". Die Liste "Schnittstelle" besitzt keine spezifischen Eigenschaften.

Abbildung 2-28



Ereignisse (1)

- Hier wird der Name des hinzugefügten Ereignisses aus der Liste "Enthaltene Objekte" angezeigt. Die Ereignisse aus der Liste "Enthaltene Objekte" werden direkt per "Drag&Drop" der Liste "Schnittstelle" zugewiesen.

Den Namen, den Sie hier vorgeben, wird am Bildbaustein unter dem Register "Ereignisse" angezeigt.

Tipp: Vergeben Sie einen Namen, über den Sie die vorgegebene Eigenschaft leicht wiedererkennen können.

Beispiel: Anstelle von "Press" besser "AutoMode_Press".

Hinweis:

Die Reihenfolge der aufgeführten Eigenschaften kann nicht nachträglich angepasst werden.

"Grafischer" Bereich

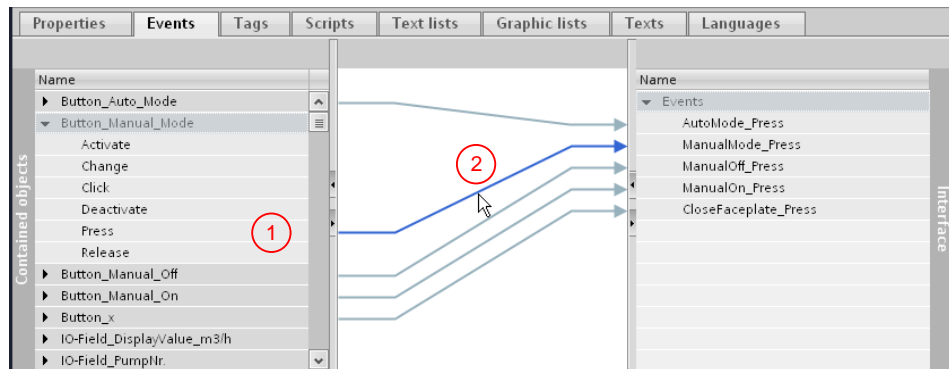
Der grafische Bereich zeigt die projektierten Verbindungen zwischen der Liste "Enthaltene Objekte" und der Liste "Schnittstelle" an.

Durch Anklicken einer Linie können Sie erkennen welche Eigenschaften miteinander verbunden sind. Dies erleichtert die Zuordnung untereinander, insbesondere bei umfangreichen Zuweisungen.

Beispiel:

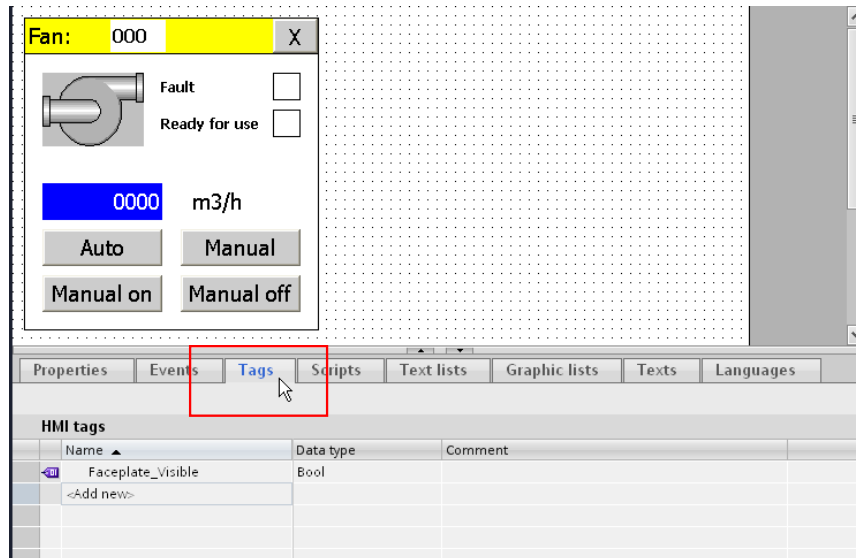
An den Schaltflächen ist das Ereignis "Drücken" projektiert (1). Die Namen in der Liste "Schnittstelle" sind entsprechend der zugewiesenen Eigenschaft angepasst worden. Durch Anklicken einer der Linien (2) können Sie die zugehörigen Zuweisungen untereinander gut erkennen.

Abbildung 2-29



2.7.3 Registerkarte "Variablen"

Abbildung 2-30

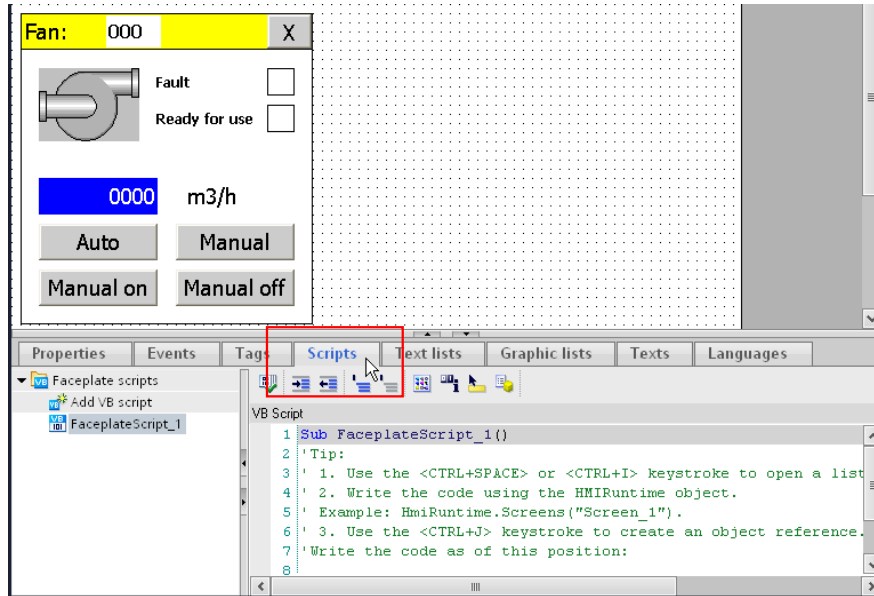


In der Registerkarte "Variablen" befinden sich Variablen, die nur innerhalb des Bildbausteins verfügbar sind. Die Variablen werden direkt mit den im Bildbaustein enthaltenen Objekten verbunden.

Hinweis Sie haben über den Bildbaustein Editor keinen Zugriff auf die Variablen tabellen, die sich im Projektbaum Ihres Projektes befinden.

2.7.4 Registerkarte "Skripte"

Abbildung 2-31



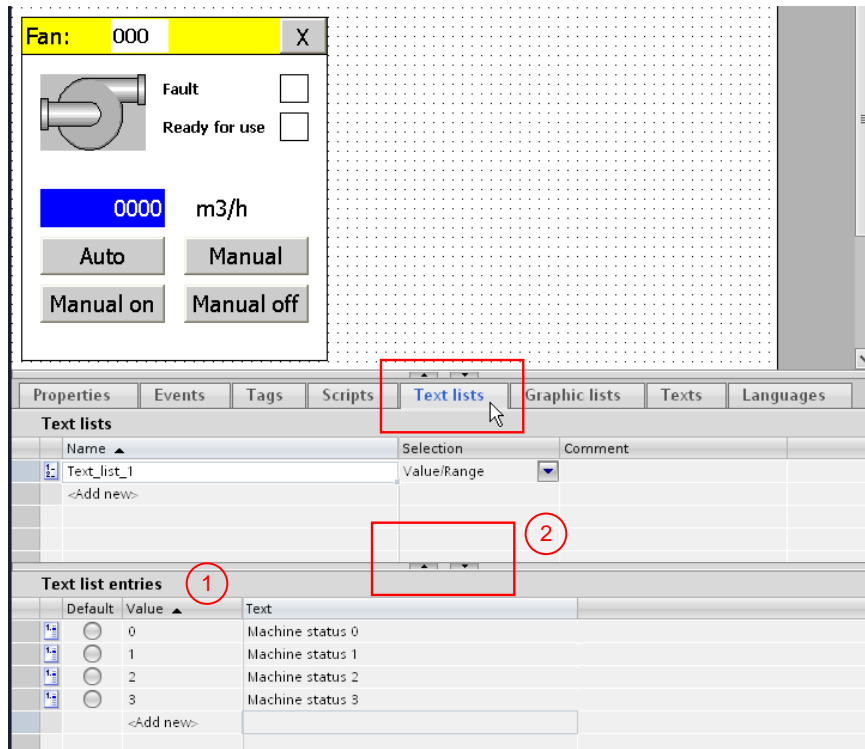
In der Registerkarte "Skripte" befinden sich Skripte, die nur innerhalb des Bildbausteins verfügbar sind. Im Skript rufen Sie Systemfunktionen auf oder programmieren neue Funktionen, z. B. zum Umrechnen von Werten. Die Programmierung erfolgt im VB-Script-Code.

Hinweis

Sie haben über den Bildbaustein Editor keinen Zugriff auf die Skripte, die sich im Projektbaum Ihres Projektes befinden.

2.7.5 Registerkarte "Textlisten"

Abbildung 2-32



In der Registerkarte "Textlisten" legen Sie Textlisten für den Bildbaustein an und bearbeiten sie. Die erstellten Textlisten sind nur innerhalb des Bildbausteins verfügbar. Die Textlisten des Bildbausteins verschalten Sie direkt an ein symbolisches E/A-Feld.

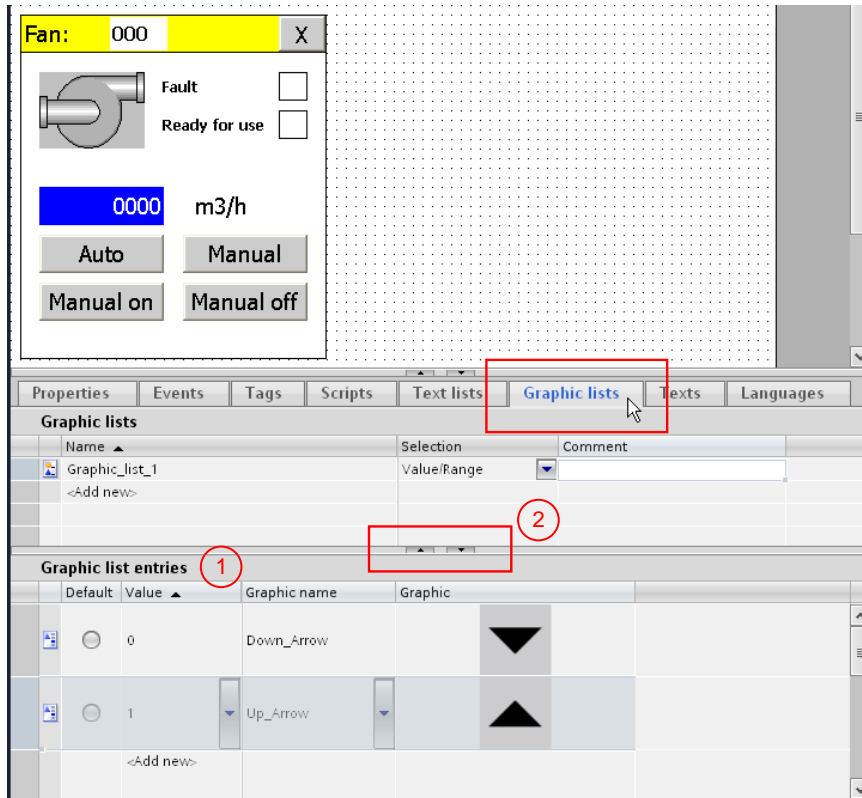
Die Einträge für die Textlisten erfolgen in dem separaten Fenster "Einträge in Textliste" (1).

Sollte das Fenster "Einträge in Textliste" nicht angezeigt werden, dann öffnen Sie das Fenster über die Pfeiltaste (2).

Hinweis Sie haben über den Bildbaustein Editor keinen Zugriff auf die Textlisten, die sich im Projektbaum Ihres Projektes befinden.

2.7.6 Registerkarte "Grafiklisten"

Abbildung 2-33



In der Registerkarte "Grafiklisten" legen Sie bei Bedarf Grafiklisten für den Bildbaustein an. Diese Grafiklisten sind nur innerhalb des Bildbausteins verfügbar. Die Grafiklisten des Bildbausteins verschalten Sie direkt an ein grafisches E/A-Feld.

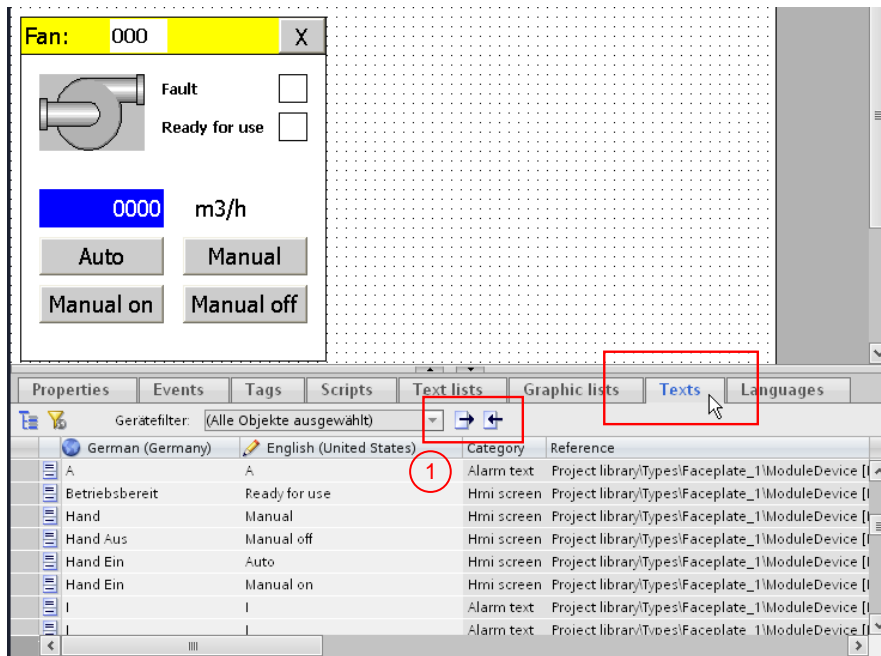
Die Einträge für die Grafiklisten erfolgen in dem separaten Fenster "Einträge in Grafiklisten" (1).

Sollte das Fenster "Einträge in Grafiklisten" nicht angezeigt werden, dann öffnen Sie das Fenster über die Pfeiltaste (2).

Hinweis Sie haben über den Bildbaustein Editor keinen Zugriff auf die Grafiklisten, die sich im Projektbaum Ihres Projektes befinden.

2.7.7 Registerkarte "Texte"

Abbildung 2-34

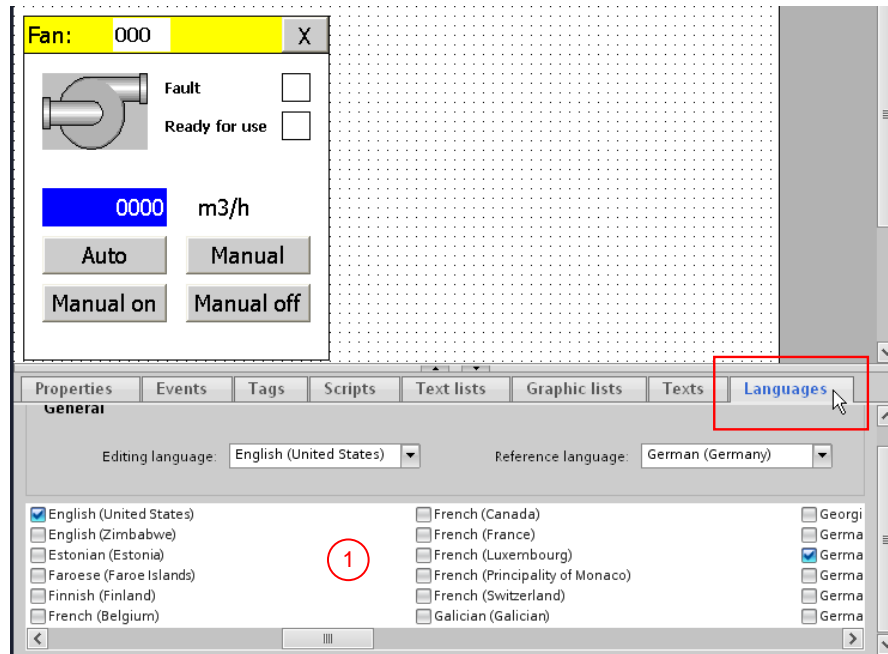


In der Registerkarte "Texte" befinden sich nur die Texte, die im Bildbaustein verwendet werden. Für mehrsprachige Projektierungen stehen Ihnen die bekannten Funktionen wie "Projekttexte importieren" und "Projekttexte exportieren" zur Verfügung (1). Alternativ tragen Sie die mehrsprachigen Texte in dem vorhandenen Editor direkt ein.

Hinweis Sie haben über den Bildbaustein Editor keinen Zugriff auf die Textliste, die sich im Projektbaum Ihres Projektes befinden.

2.7.8 Registerkarte "Sprachen"

Abbildung 2-35



In der Registerkarte "Sprachen" geben Sie die Sprachen vor, die für den erstellten Bildbaustein zur Verfügung stehen sollen. Die Sprachen sind unabhängig von der im Projekt gewählten Sprache.

Durch diese Funktionalität können Sie einen Bildbaustein auch in Projekten mit unterschiedlich erstellten Sprachen verwenden.

Bei der Verwendung des Bildbausteins ist dabei zu beachten, dass die hinterlegten Sprachen des Bildbausteins, auch in dem Projekt verwendet werden, in dem der Bildbaustein aufgerufen wird.

Beispiel:

Sie haben in Ihrem Projekt die Sprachen "Deutsch(Deutschland)" und "English(**United States**)" ausgewählt.

Den Bildbaustein haben Sie in den Sprachen "Deutsch(Deutschland)" und "English(**Great Britain**)" erstellt.

Wird der Bildbaustein in dem Projekt mit den Sprachen Deutsch(Deutschland)" und "English(**United States**)" verwendet und die Sprache "English(**United States**)" angewählt, dann werden am Bildbaustein keine Texte angezeigt, da am Bildbaustein "nur" die Sprache "English(**Great Britain**)" hinterlegt ist.

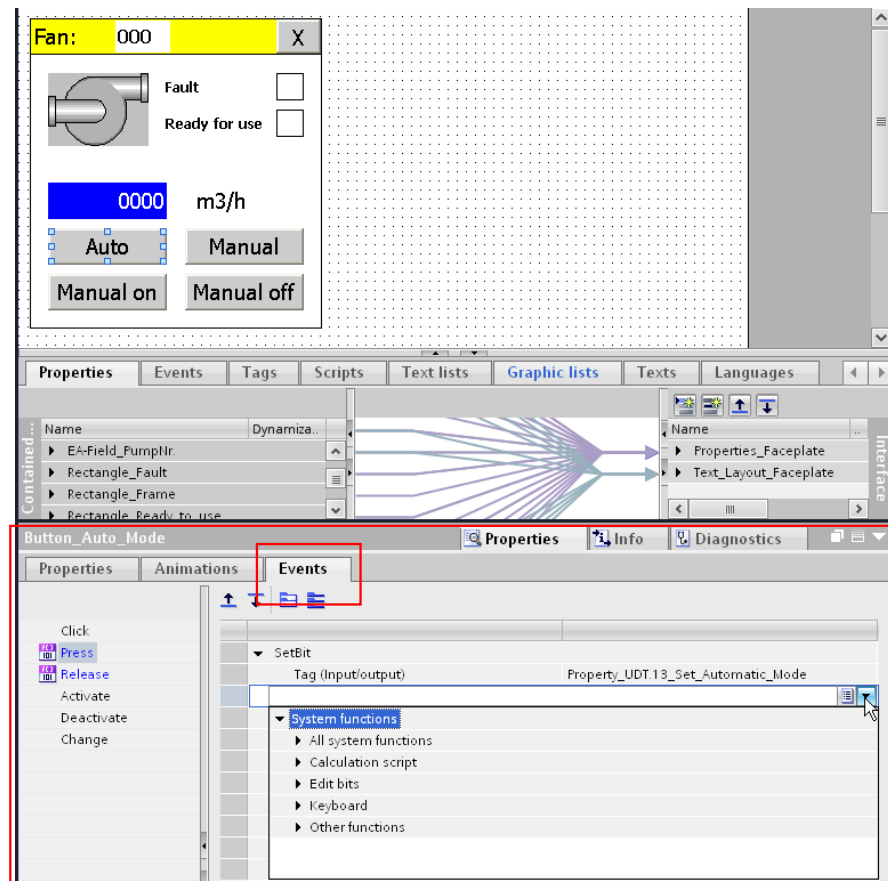
Hinweis Einstellungen unter den Projektsprachen im Bildbaustein Editor sind unabhängig von den Projektsprachen des Anwenderprojektes.

2.8 Parameter im Inspektorfenster

Das Inspektorfenster im Bildbaustein Editor beinhaltet im Wesentlichen die gleichen Parameter und Einstellungsmöglichkeiten, wie Sie es vom Editor "Bilder" her kennen.

Abweichungen gibt es im Register "Eigenschaften > Ereignisse". Funktionen wie z. B. "AktiviereBild" stehen Ihnen hier nicht zur Verfügung.

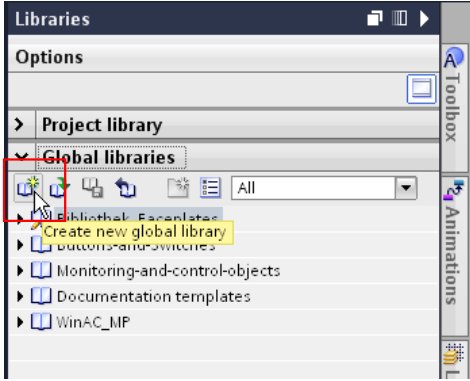
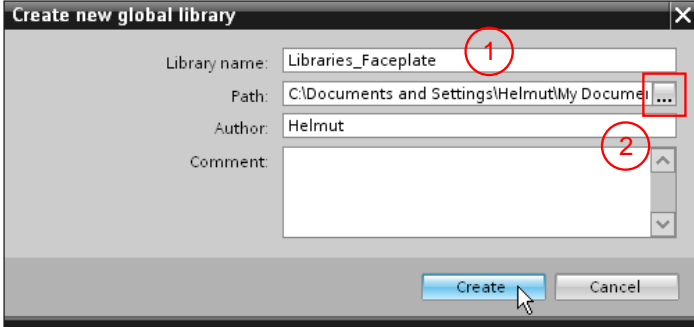
Abbildung 2-36



2.9 Bildbausteine unter "Globale Bibliotheken" speichern

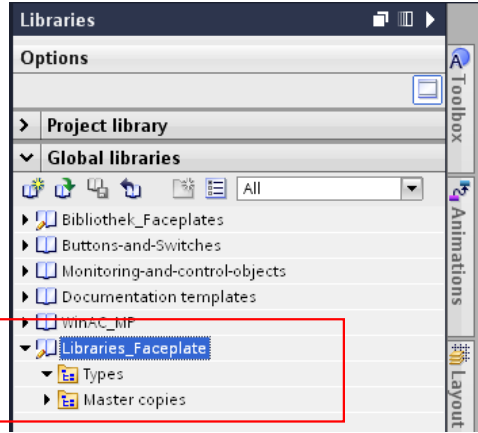
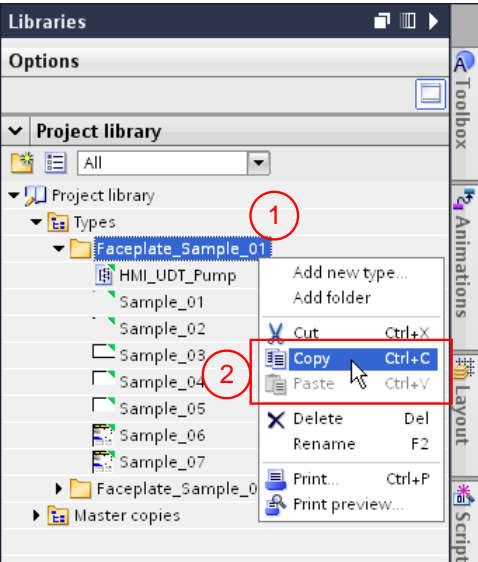
Die erstellten Bildbausteine werden in der Projektbibliothek angelegt und bearbeitet. Wenn Sie die erstellten Bildbausteine projektübergreifend wiederverwenden möchten, dann speichern Sie die fertig gestellten Bildbausteine unter der Task Card "Bibliotheken" und dort unter der Palette "Globale Bibliotheken" ab.

Tabelle 2-4

Nr.	Aktion
1.	<p>Neue "Globale Bibliothek" erstellen</p> <ul style="list-style-type: none"> • Öffnen Sie die Task Card "Bibliotheken" und dort die Palette "Globale Bibliotheken". • Klicken Sie auf das erste Icon. Über dieses Icon wird eine neue globale Bibliothek erstellt.  <ul style="list-style-type: none"> • Es öffnet sich ein Fenster, in dem Sie den Namen (1) sowie den Ablagepfad (2) usw. vorgeben. Schließen Sie die Eingaben über die Schaltfläche "Erstellen" ab. 

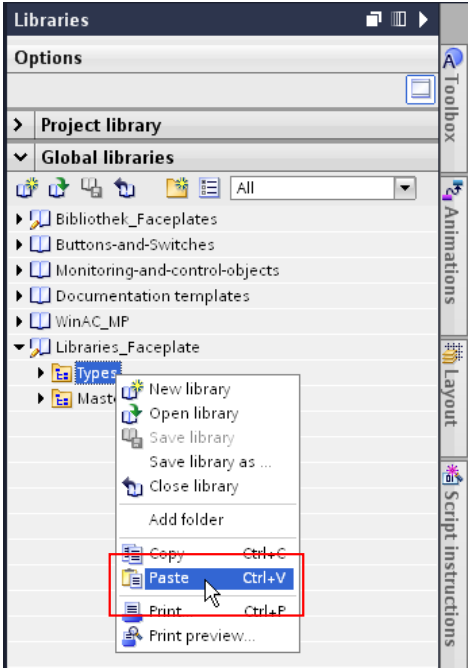
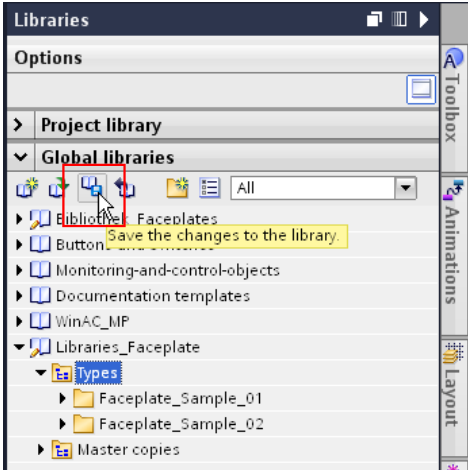
2 Funktionen und Eigenschaften des Bildbausteins

2.9 Bildbausteine unter "Globale Bibliotheken" speichern

Nr.	Aktion
	<p>Ansicht der neu erstellten "Globalen Bibliothek"</p> <p>Das nachfolgende Bild zeigt die Ansicht der neu erstellten globalen Bibliothek "Libraries_Faceplate" mit den Unterordnern "Typen" und "Kopiervorlagen".</p>  <p>The screenshot shows the 'Libraries' panel in a software interface. Under the 'Global libraries' section, the 'Libraries_Faceplate' folder is selected and expanded, showing subfolders 'Types' and 'Master copies'. A red rectangular box highlights the 'Libraries_Faceplate' folder and its immediate subfolders.</p>
2.	<p>Bildbausteine aus der Projektbibliothek kopieren</p> <ul style="list-style-type: none">• Wechseln Sie in die Palette "Projektbibliothek".• Markieren Sie unter dem Ordner "Typen" einen einzelnen Bildbaustein bzw. den ganzen Ordner mit den enthaltenen Bildbausteinen (1).• Klicken Sie mit der rechten Maustaste auf den markierten Eintrag. Es öffnet sich ein Kontextmenü. Wählen Sie hier die Option "Kopieren" (2).  <p>The screenshot shows the 'Libraries' panel with the 'Project library' section expanded. The 'Types' folder is selected, and its subfolder 'Faceplate_Sample_01' is highlighted with a red circle labeled '1'. A right-click context menu is open over 'Faceplate_Sample_01', and the 'Copy' option is highlighted with a red box and a red circle labeled '2'. The context menu includes options like 'Add new type...', 'Add folder', 'Cut', 'Copy', 'Paste', 'Delete', 'Rename', 'Print...', and 'Print preview...'.</p>

2 Funktionen und Eigenschaften des Bildbausteins

2.9 Bildbausteine unter "Globale Bibliotheken" speichern

Nr.	Aktion
3.	<p>Kopierte Bildbausteine in die globale Bibliothek "Libraries_Faceplate" einfügen</p> <ul style="list-style-type: none">• Wechseln Sie wieder in die Palette "Globale Bibliotheken" und öffnen Sie die globale Bibliothek "Libraries_Faceplate".• Markieren Sie über die rechte Maustaste den Ordner "Typen". Es öffnet sich ein Kontextmenü. Wählen Sie hier die Option "Einfügen". Der zuvor kopierte Ordner mit den Bildbausteinen wird in den Ordner "Typen" kopiert.  <p>The screenshot shows the 'Libraries' window with the 'Global libraries' section expanded. The 'Libraries_Faceplate' folder is selected, and its context menu is open. The 'Paste' option is highlighted with a red box, and a mouse cursor is pointing at it. Other options in the menu include 'New library', 'Open library', 'Save library', 'Save library as ...', 'Close library', 'Add folder', 'Copy', 'Print', and 'Print preview...'. The 'Copy' and 'Print' options are also highlighted with a red box.</p>
4.	<p>Änderungen unter "Globale Bibliotheken" speichern</p> <ul style="list-style-type: none">• Klicken Sie auf das dritte Icon. Über dieses Icon werden die Änderungen in der Bibliothek gespeichert.  <p>The screenshot shows the 'Libraries' window with the 'Global libraries' section expanded. The 'Libraries_Faceplate' folder is selected, and its context menu is open. The 'Save the changes to the library' icon is highlighted with a red box. The context menu also shows 'Copy', 'Paste', 'Print', and 'Print preview...' options.</p> <p>Weitergehende Informationen zum Thema "Bibliotheken" finden Sie in der Onlinehilfe.</p>

3 Funktionen und Eigenschaften eines HMI UDT

In diesem Kapitel werden die Schritte zum Erstellen / Bearbeiten eines HMI UDT Anwenderdatentyps beschrieben.

Hinweis Ab WinCC (TIA Portal) V13 SP1 und bei der Verwendung einer S7-1200 bzw. S7-1500 Steuerung benötigen Sie keinen "HMI UDT" mehr. Sehen Sie sich hierzu das Kapitel [4](#) an.

Wenn Sie mit den Eigenschaften eines Anwenderdatentyps noch nicht vertraut sind, dann lesen Sie sich dieses Kapitel unbedingt durch.

Einleitung

Jeder projektierten Funktion eines Bildbausteins muss später über die Schnittstelle eine Variable zugewiesen werden. Dieses kann je nach Umfang des Bildbausteins recht umfangreich sein.

Durch die Verwendung eines HMI UDTs, wird dem Bildbaustein über die Schnittstelle nur noch **eine** Variable zugewiesen.

Allgemein

Zu einem Anwenderdatentyp (HMI UDT) gehört immer

- ein Datenbaustein, der dieselbe Struktur aufweist wie der Anwenderdatentyp.
- eine HMI Variable, bei der als "Datentyp" der erstellte Anwenderdatentyp ausgewählt worden ist. Über diese Variable erfolgt dann der Datenaustausch zwischen der SPS und dem im Bildbaustein verwendeten Anwenderdatentyp.

Für die Projektierung eines Anwenderdatentyps ist es sinnvoll, zunächst den Datenbaustein mit der erforderlichen Struktur zu erstellen.

Sie können einem Anwenderdatentyp jeweils einen Datenbaustein zuweisen. Sie können aber auch einen Datenbaustein verwenden, der mehrere Strukturen beinhaltet und diese den einzelnen Anwenderdatentypen zuordnen.

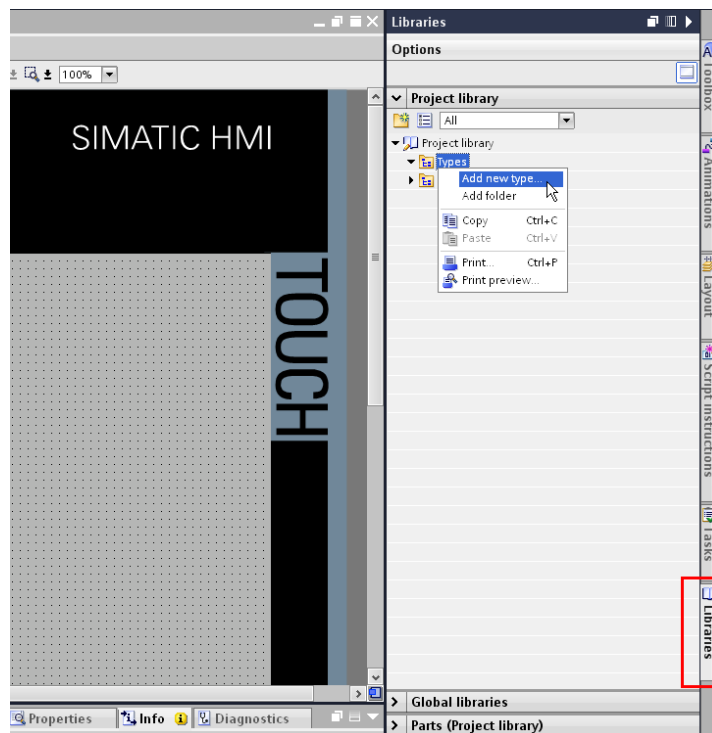
Hinweis Einen HMI UDT können sie nur in Verbindung mit einem Bildbaustein verwenden.

3.1 Anwenderdatentyp erstellen

Um einen neuen Anwenderdatentyp zu erstellen, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie mit der rechten Maustaste den Ordner "Typen". Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Neuen Typen hinzufügen..." an. Das Fenster "Neuen Typ hinzufügen" wird geöffnet.

Abbildung 3-1

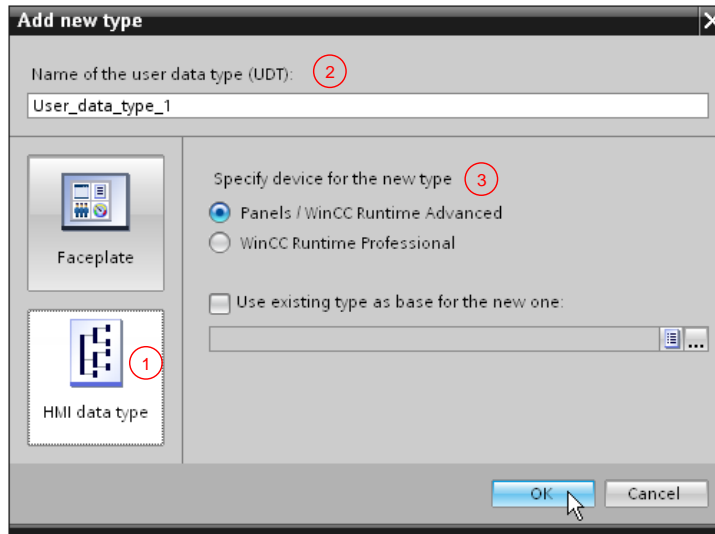


3 Funktionen und Eigenschaften eines HMI UDT

3.1 Anwenderdatentyp erstellen

Ansicht des Fensters "Neuen Typ hinzufügen".

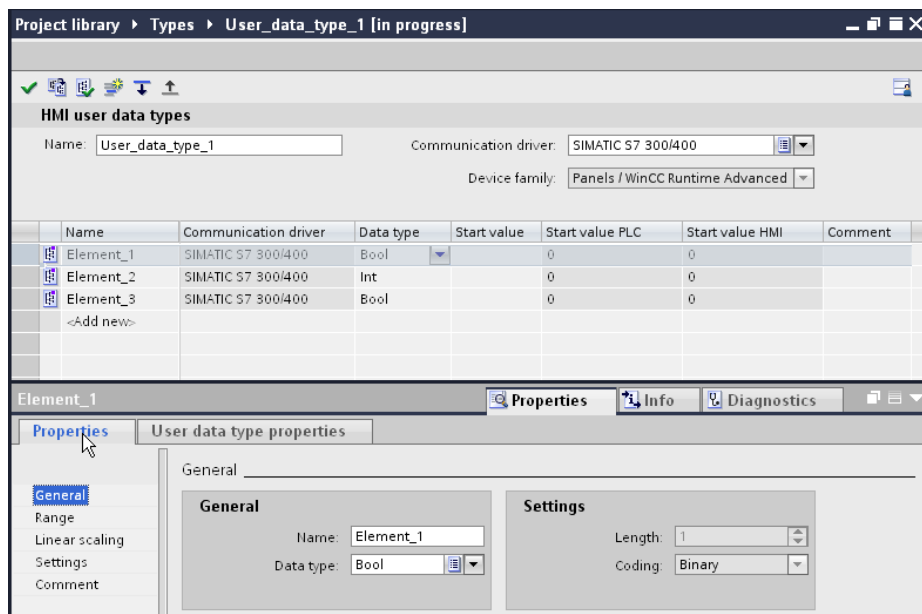
Abbildung 3-2



- Wählen Sie auf der linken Seite den Typ "HMI UDT" an (1).
- Vergeben Sie dem Anwenderdatentyp einen eindeutigen Namen (2).
- Wählen Sie unter "Gerät für den neuen Typ festlegen" das Optionsfeld "Panels / WinCC Runtime Advanced" an (3).
- Bestätigen Sie die Eingaben mit "OK".
Der Editor zum Erstellen / Bearbeiten des Anwenderdatentyps wird geöffnet.

Ansicht des geöffneten Editors.

Abbildung 3-3

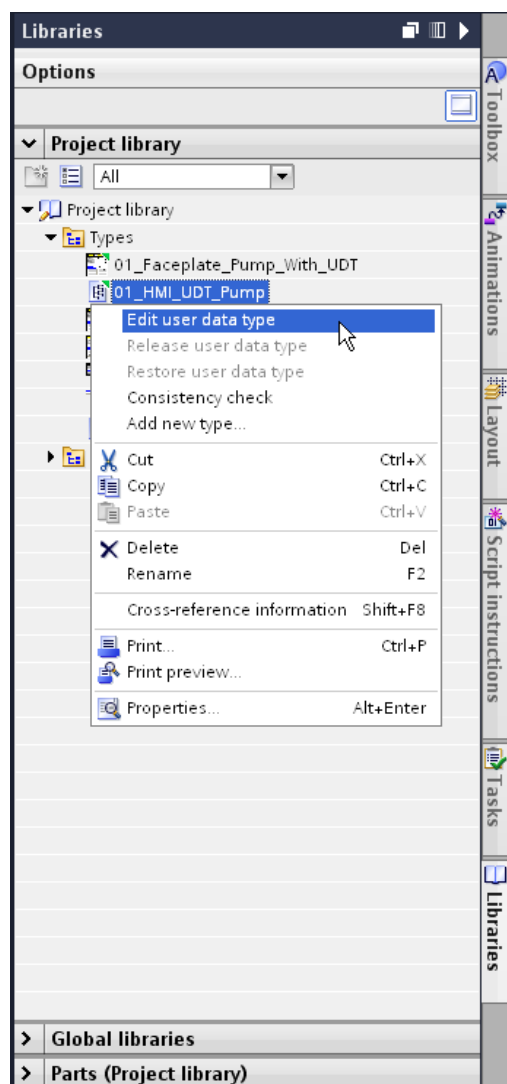


3.2 Anwenderdatentyp bearbeiten/öffnen

Um einen vorhandenen Anwenderdatentyp zu bearbeiten, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Anwenderdatentyp, den Sie öffnen möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Anwenderdatentyp bearbeiten" an. Der Editor zum Erstellen / Bearbeiten des Anwenderdatenbausteins wird geöffnet.

Abbildung 3-4



3.3 Anwenderdatentyp Konfigurieren

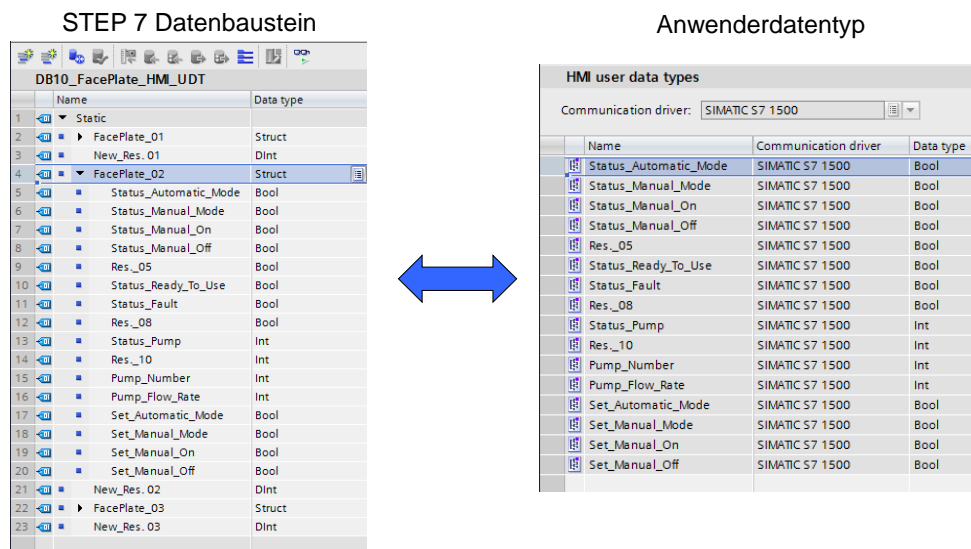
Allgemein

Was Sie bei der Erstellung beachten sollten:

- Einem Anwenderdatentyp können Sie nur **einen** Kommunikationstreiber zuweisen.
 - **Beispiel:**
 Sie haben eine Projektierung mit einem Bildbaustein. Dieser Bildbaustein verwendet einen Anwenderdatentyp. Als Kommunikationstreiber ist in dem Anwenderdatentypen eine S7-300/S7-400 Steuerung projektiert. Wenn Sie diesen Bildbaustein in einem Projekt mit einer S7-1500 Steuerung verwenden möchten, dann müssen Sie im Anwenderdatentyp den Kommunikationstreiber anpassen.
- Bevor Sie einen Anwenderdatentyp konfigurieren, sollten vorher alle "Funktionen" die ein Bildbaustein haben soll, bekannt sein. Es erleichtert später die Zuordnung zu den ausgewählten Eigenschaften im Bildbaustein. Sie können einen Anwenderdatentyp aber auch jederzeit später anpassen.
- Der Datenbaustein in der STEP 7 Programmierung muss dieselbe "Struktur" haben, wie der erstellte Anwenderdatentyp.
- Die verwendeten "Datentypen" im Datenbaustein der STEP 7 Programmierung muss mit den Datentypen im Anwenderdatentyp übereinstimmen.

Beispielansicht zwischen einem DB mit dem Datentyp "Struct" und einem Anwenderdatentyp. Das Bild verdeutlicht den Aufbau (Struktur) beider Dateien.

Abbildung 3-5



Hinweis:

Für jeden Bildbaustein, den Sie im Projekt verwenden, müssen Sie einen Datenbereich (Struktur) vorsehen. Sie können für jeden Bildbaustein einen eigenen Datenbaustein anlegen oder, wie

3.3 Anwenderdatentyp Konfigurieren

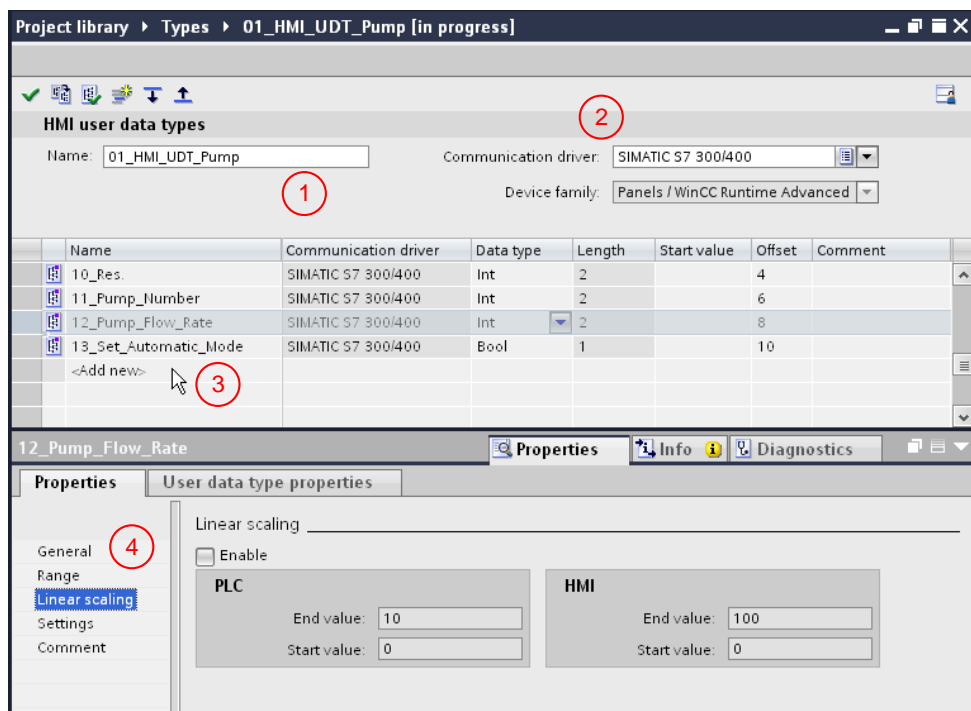
in der Abbildung zu sehen ist, einen Datenbaustein für mehrere Bildbausteine vorsehen.

Der Anwenderdatentyp muss nur einmal erstellt werden. Dieser kann für jeden Bildbaustein verwendet werden, der die gleiche Funktionalität hat.

Variablen anlegen

Die Vorgehensweise um Variablen in den Anwenderdatentyp Editor einzufügen ist ähnlich, wie Sie es vom Einfügen von Variablen in eine "Variablen-tabelle" her kennen.

Abbildung 3-6



- (1) Hier können Sie den Namen des Anwenderdatentyps anpassen.
- (2) Wählen Sie hier den Kommunikationstreiber aus (z. B. S7 300/400). Der eingestellte Kommunikationstyp gilt für alle Elemente. Wenn Sie einen Kommunikationstreiber abgewählt haben, dann können Sie keine internen Variablen als Datentyp verwenden.
- (3) Doppelklicken Sie in der Spalte "Name" auf "<Hinzufügen>" und geben Sie den Datentyp usw. vor. Ersetzen Sie in der Spalte "Name" den Standardeintrag durch einen aussagekräftigen Namen. Der Name sollte einen Bezug zu dem im Datenbaustein verwendeten Variablennamen haben.
- (4) Hier können Sie spezielle Eigenschaften wie z. B. "Lineare Skalierung" für die Variable vorgeben. Die möglichen Eigenschaften sind im Vergleich zur "Variablen-tabelle" eingeschränkt.

Nachdem Sie alle Variablen eingefügt haben, schließen Sie den Editor über die Schaltfläche "Schließen" "X".

Damit der erstellte Anwenderdatentyp im Projekt verfügbar ist, muss dieser zunächst freigegeben werden. Schauen Sie sich hierzu das Kapitel 3.4 an.

3.4 Anwenderdatentyp

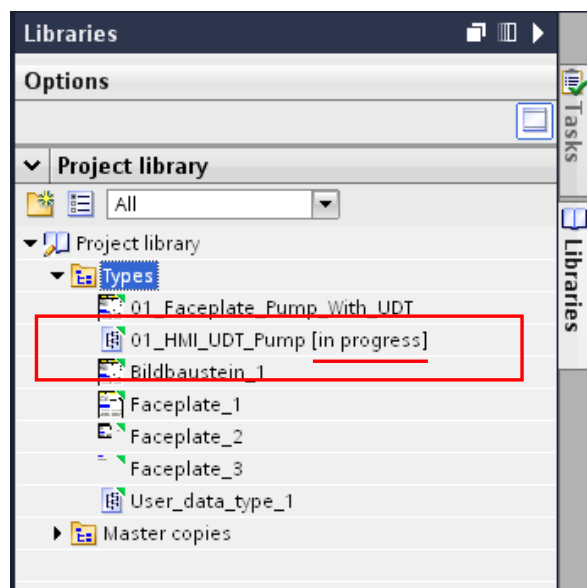
Status Anwenderdatentyp

Wenn Sie einen Anwenderdatentyp bearbeiten, dann erscheint neben dem Namen des Anwenderdatentyps der folgende Eintrag "Name des Anwenderdatentyps **[in Arbeit]**".

In der Kopfzeile des Anwenderdatentyp Editors können Sie ebenfalls den aktuellen Status des geöffneten Anwenderdatentyps sehen.

Über die Statusangabe "[in Arbeit]" können Sie erkennen, dass an dem vorliegenden Anwenderdatentyp Änderungen durchgeführt worden sind.

Abbildung 3-7



Was passiert mit einem im Projekt verwendeten Anwenderdatentyp, wenn der zugehörige Anwenderdatentyp aktuell bearbeitet wird?

Wenn Sie einen Anwenderdatentyp bearbeiten, dann werden Änderungen an dem im Projekt verwendeten Anwenderdatentyp erst wirksam, wenn Sie den zugehörigen Anwenderdatentyp explizit freigeben (siehe Kapitel 3.4.1).

Sie können z. B. die Bearbeitung eines Anwenderdatentyps unterbrechen und die Projektierung speichern und schließen. Nach dem Öffnen der Projektierung können Sie die Bearbeitung des Anwenderdatentyps fortsetzen. Der Status des bearbeiteten Anwenderdatentyps steht dann immer noch auf "[in Arbeit]".

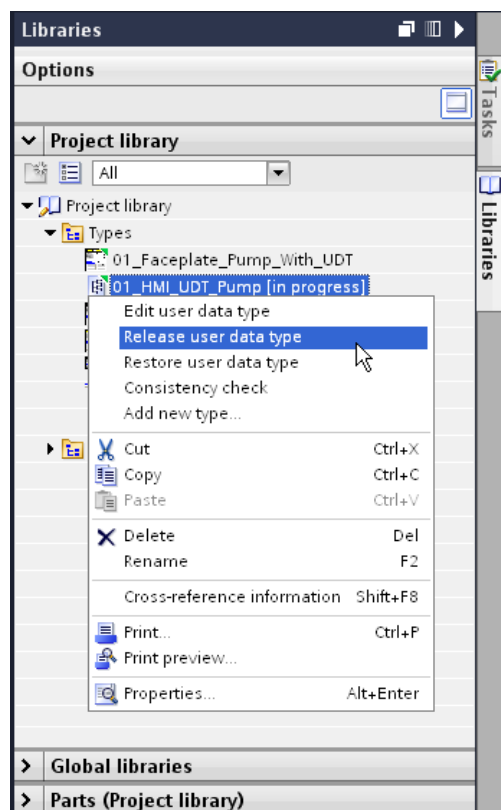
3.4.1 Anwenderdatentyp freigeben

Wenn Sie einen Anwenderdatentyp erstellt bzw. bearbeitet haben, dann ist dieser im Projekt zunächst nicht verwendbar. Einen Anwenderdatentyp müssen Sie zuerst freigeben, damit er im Projekt verfügbar ist.

Um einen vorhandenen Anwenderdatentyp freizugeben, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Anwenderdatentyp, den Sie freigeben möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Anwenderdatentyp freigeben" an. Der Anwenderdatentyp wird aktualisiert und kann jetzt im Projekt verwendet werden.

Abbildung 3-8



3.4.2 Anwenderdatentyp wiederherstellen

Sie können z. B. die Bearbeitung eines Anwenderdatentyps unterbrechen und die Projektierung speichern und schließen. Nach dem Öffnen der Projektierung können Sie die Bearbeitung des Anwenderdatentyps fortsetzen.

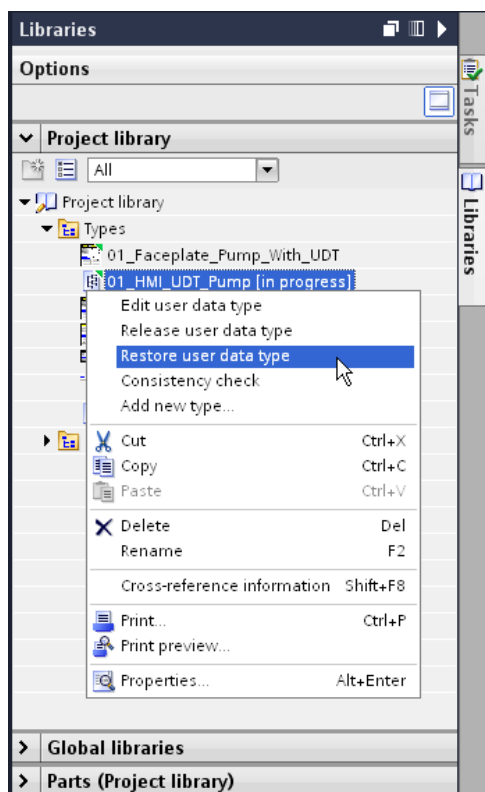
Wenn Sie jetzt den "alten" Stand des Anwenderdatentyps wieder herstellen möchten, dann verwenden Sie die Funktion "Anwenderdatentyp wiederherstellen".

Alle Änderungen, die seit der letzten Freigabe am Anwenderdatentyp vorgenommen worden sind, werden verworfen. Der Anwenderdatentyp ist anschließend wieder freigegeben.

Um einen Anwenderdatentyp wiederherzustellen, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Markieren Sie in dem Ordner "Typen" mit der rechten Maustaste den Anwenderdatentyp, den Sie wiederherstellen möchten. Es öffnet sich ein Kontextmenü.
- Wählen Sie in dem Kontextmenü "Anwenderdatentyp wiederherstellen" an. Die Änderungen werden verworfen. Der Anwenderdatentyp ist wieder freigegeben.

Abbildung 3-9

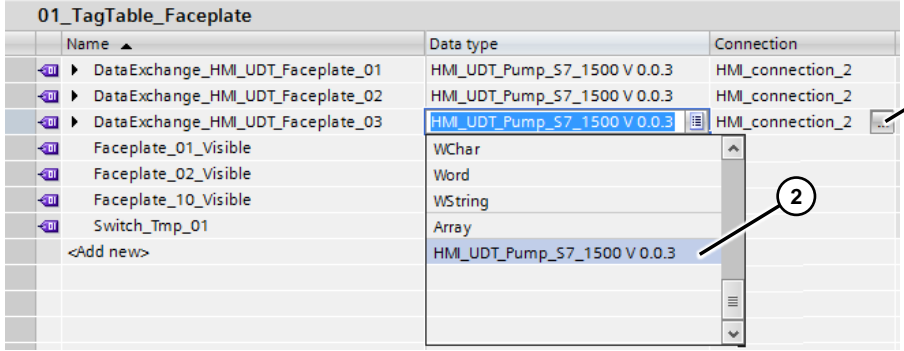


3.5 HMI UDT einer HMI-Variablen zuweisen

Für den Datenaustausch zwischen der SPS und dem im Bildbaustein verwendeten Anwenderdatentyp ist eine HMI-Variable erforderlich.

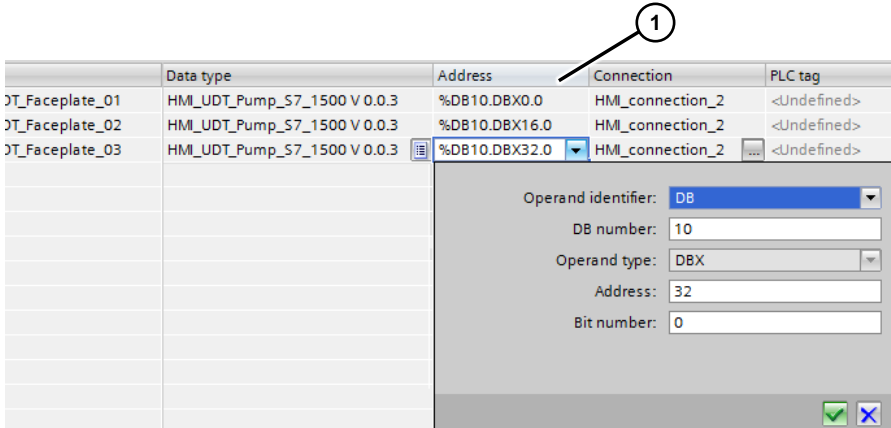
Sehen Sie sich hierzu die nachfolgende Beschreibung an.

Tabelle 3-1

Nr.	Aktion
1.	<p>HMI Variable anlegen</p> <ul style="list-style-type: none">• Öffnen Sie in der HMI-Projektierung den Variablen-Ordner und legen Sie in der Variablen-tabelle eine neue Variable an.• Weisen Sie der Variablen eine Steuerungsanbindung zu, über die der Bildbaustein mit der SPS kommuniziert (1).• Wählen Sie über die Klappliste in der Spalte "Datentyp" den im Bildbaustein verwendeten Anwenderdatentyp (HMI UDT) aus (2). 

3 Funktionen und Eigenschaften eines HMI UDT

3.5 HMI UDT einer HMI-Variablen zuweisen

Nr.	Aktion																				
2.	<p>Anfangsadresse der HMI Variablen zuweisen</p> <p>Für jeden Bildbaustein haben Sie in einem Datenbaustein einen Datenbereich ("Struktur") vorgesehen. Sehen Sie sich hierzu das Kapitel 3.3 an. Für den Datenaustausch müssen Sie jeweils die Anfangsadresse vorgeben.</p> <ul style="list-style-type: none"> Geben Sie in der Spalte "Adresse" die Anfangsadresse von dem Datenbereich an (1). Der im Kapitel 3.3 gezeigte Datenbaustein "DB10" beinhaltet die Struktur für drei Bildbausteine. <ul style="list-style-type: none"> Die Anfangsadresse für den ersten Bildbaustein lautet "DB10.DBX0.0" Die Anfangsadresse für den zweiten Bildbaustein lautet "DB10.DBX16.0" Die Anfangsadresse für den dritten Bildbaustein lautet "DB10.DBX32.0"  <table border="1" data-bbox="470 806 1364 907"> <thead> <tr> <th></th> <th>Data type</th> <th>Address</th> <th>Connection</th> <th>PLC tag</th> </tr> </thead> <tbody> <tr> <td>DT_Faceplate_01</td> <td>HMI_UDT_Pump_S7_1500 V 0.0.3</td> <td>%DB10.DBX0.0</td> <td>HMI_connection_2</td> <td><Undefined></td> </tr> <tr> <td>DT_Faceplate_02</td> <td>HMI_UDT_Pump_S7_1500 V 0.0.3</td> <td>%DB10.DBX16.0</td> <td>HMI_connection_2</td> <td><Undefined></td> </tr> <tr> <td>DT_Faceplate_03</td> <td>HMI_UDT_Pump_S7_1500 V 0.0.3</td> <td>%DB10.DBX32.0</td> <td>HMI_connection_2</td> <td><Undefined></td> </tr> </tbody> </table> <div data-bbox="901 907 1364 1176"> <p>Operand identifier: DB</p> <p>DB number: 10</p> <p>Operand type: DBX</p> <p>Address: 32</p> <p>Bit number: 0</p> </div>		Data type	Address	Connection	PLC tag	DT_Faceplate_01	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX0.0	HMI_connection_2	<Undefined>	DT_Faceplate_02	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX16.0	HMI_connection_2	<Undefined>	DT_Faceplate_03	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX32.0	HMI_connection_2	<Undefined>
	Data type	Address	Connection	PLC tag																	
DT_Faceplate_01	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX0.0	HMI_connection_2	<Undefined>																	
DT_Faceplate_02	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX16.0	HMI_connection_2	<Undefined>																	
DT_Faceplate_03	HMI_UDT_Pump_S7_1500 V 0.0.3	%DB10.DBX32.0	HMI_connection_2	<Undefined>																	

4 Funktionen und Eigenschaften eines PLC-Datentypen

In diesem Kapitel werden die Schritte zum Erstellen / Bearbeiten eines PLC-Datentypen ("PLC-UDT") beschrieben.

Hinweis

Ab WinCC (TIA Portal) V13 SP1 und bei der Verwendung einer S7-1200 bzw. S7-1500 Steuerung benötigen Sie keinen "HMI UDT" mehr.

4.1 Einleitung

Jeder projektierten Funktion eines Bildbausteins muss später über die Schnittstelle eine Variable zugewiesen werden. Dieses kann je nach Umfang des Bildbausteins recht umfangreich sein.

Durch die Verwendung eines PLC-Datentypen, wird dem Bildbaustein über die Schnittstelle nur noch **eine** Variable zugewiesen.

Im Kapitel [3](#) wird beschrieben, wie Sie einen Anwenderdatentyp (HMI UDT) verwenden. Dieses ist z. B. der Fall, wenn Sie eine SIMATIC S7 300/400 Steuerung verwenden.

Der Nachteil eines HMI UDTs ist:

- Der HMI UDT kann nur für Bildbausteine verwendet werden.
- Werden Änderungen im zugehörigen Datenbaustein vorgenommen, dann müssen diese anschließend manuell im HMI UDT nachgetragen werden. Dieses kann zu Fehlern bei der Namen/Variablentyp Zuweisung führen.

Ab WinCC (TIA Portal) V13 SP1 und in Verbindung mit einer SIMATIC S7-1200 bzw. S7-1500 Steuerung benötigen Sie **keinen** Anwenderdatentyp mehr. Sie können ab dieser Version einen **PLC-Datentypen** verwenden.

Der Vorteil eines PLC-Datentypen ist:

- Der Datentyp muss nur einmal erstellt werden.
- Der erstellte Datentyp kann auch in weiteren Anwendungen – nicht nur in Bildbausteinen – verwendet werden.
- Änderungen in der Datenstruktur, werden automatisch in allen Verwendungsstellen nachgezogen.

4.2 PLC-Datentypen erstellen

Allgemein

Was Sie bei der Erstellung beachten sollten:

Bevor Sie einen PLC-Datentypen konfigurieren, sollten vorher alle "Funktionen" die ein Bildbaustein haben soll, bekannt sein. Es erleichtert später die Zuordnung zu den ausgewählten Eigenschaften im Bildbaustein. Sie können einen PLC-Datentypen aber auch jederzeit später anpassen.

PLC-Datentyp erstellen

- Wählen Sie in der Projektnavigation die SPS-Steuerung an, über die später der Datenaustausch mit dem Bildbaustein erfolgt.
- Wählen Sie den Ordner PLC-Datentypen an und fügen Sie einen neuen "Datentypen" hinzu.
- Tragen Sie alle benötigten Variablen in die Tabelle ein.

Hinweis

Wenn in einem bestehenden Datenbaustein bereits eine "Datenstruktur" vorliegt, dann können Sie diese kopieren und direkt in den PLC-Datentypen einfügen.

Abbildung 4-1

Name		UserDataTyp_Faceplate	
		Name	Data type
PLC_2_HMI_2 [CPU 1515-2 PN]	1	Status_Automatic_Mode	Bool
Device configuration	2	Status_Manual_Mode	Bool
Online & diagnostics	3	Status_Manual_On	Bool
Program blocks	4	Status_Manual_Off	Bool
Technology objects	5	O5_Res.	Bool
External source files	6	Status_Ready_To_Use	Bool
PLC tags	7	Status_Fault	Bool
PLC data types	8	O8_Res.	Bool
Add new data type	9	Status_Pump	Int
UserDataTyp_Faceplate	10	10_Res.	Int
Watch and force tables	11	Pump_Number	Int
Online backups	12	Pump_Flow_Rate	Int
Traces	13	Set_Automatic_Mode	Bool
Program info	14	Set_Manual_Mode	Bool
Device proxy data	15	Set_Manual_On	Bool
PLC alarms	16	Set_Manual_Off	Bool
Text lists	17	<Add new>	
Local modules			

4.3 PLC-Datentyp in einen Datenbaustein einfügen

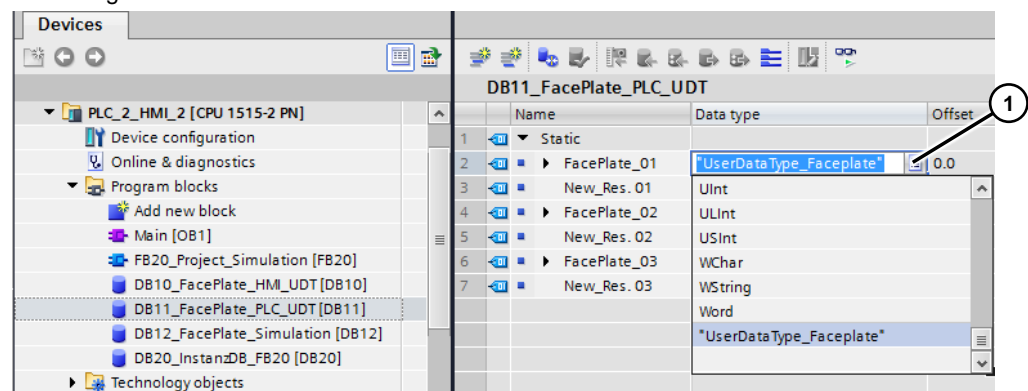
Vorausgesetzt wird, dass bereits ein "PLC-Datentyp" vorhanden ist.

- Öffnen Sie einen bestehenden Datenbaustein bzw. erstellen Sie einen neuen Datenbaustein.
- Legen Sie eine neue Variable an. Wählen Sie in der Spalte "Datentyp" den erstellten "PLC-Datentyp" aus (1). In diesem Beispiel "UserDataTyp_Faceplate".
- Fügen Sie für jede Bedienstation eine weitere Variable ein.

Hinweis

Sie können den erstellten "PLC-Datentyp" auch für "Bildbausteine" verwenden, die bis jetzt "HMI-UDTs" verwenden. Voraussetzung ist, dass beide Datentypen die gleiche Struktur/Aufbau haben.

Abbildung 4-2



4.4 PLC-Datentypen in einen Bildbaustein einbinden.

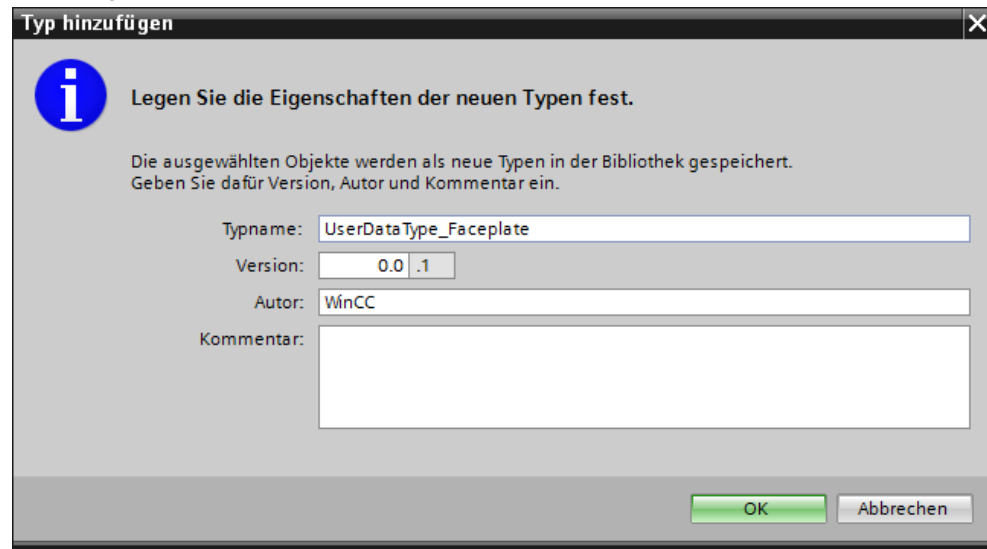
Um einen PLC-Datentypen in einem Bildbaustein zu verwenden, gehen Sie wie folgt vor:

- Wählen Sie die Task Card "Bibliotheken" an.
- Öffnen Sie die Palette "Projektbibliothek".
- Öffnen Sie in dem Ordner "Typen".
- Wählen Sie in der Projektnavigation die SPS-Steuerung an, über die später der Datenaustausch mit dem Bildbaustein erfolgt.
- Wählen Sie den Ordner PLC-Datentypen an und markieren Sie den bestehenden PLC-Datentypen. In diesem Beispiel "UserDataTyp_Faceplate".
- Ziehen Sie den markierten PLC-Datentypen per "Drag&Drop" in die geöffnete "Projektbibliothek > Typen". Es öffnet sich ein Fenster, indem Sie den Namen des hinzugefügten PLC-Datentypen verändern können. Bestätigen Sie die Eingaben mit "OK".

4 Funktionen und Eigenschaften eines PLC-Datentypen

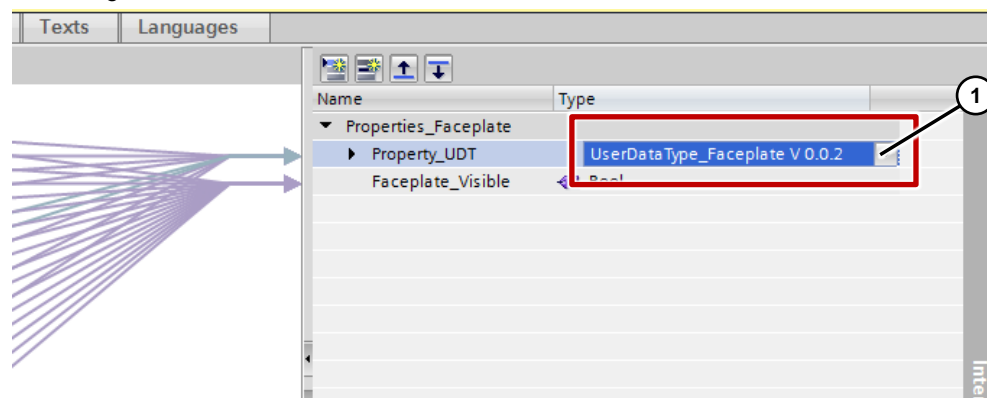
4.4 PLC-Datentypen in einen Bildbaustein einbinden.

Abbildung 4-3



- Öffnen Sie einen bestehenden – bzw. erstellen Sie einen neuen Bildbaustein.
- Fügen Sie im Konfigurationsbereich im Register "Eigenschaften > Schnittstelle" eine Variable hinzu und wählen Sie unter "Typ" über die Klappliste den angelegten PLC-Datentypen (1) aus. In diesem Beispiel "UserDataTyp_Faceplate".

Abbildung 4-4



4.5 PLC-Datentypen bearbeiten/öffnen

Um einen vorhandenen PLC-Datentypen zu bearbeiten, gehen Sie wie folgt vor:

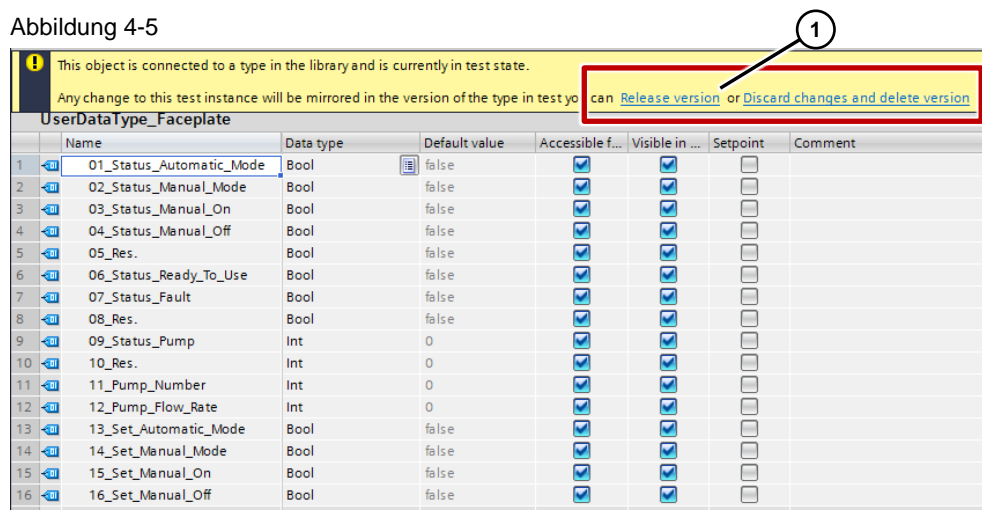
"PLC-Datentyp" wurde noch nicht in eine Bibliothek eingefügt

- Markieren Sie den zu bearbeitenden PLC-Datentypen mit der rechten Maustaste.
- Wählen Sie in dem Kontextmenü den Punkt "Öffnen".
- Führen Sie die erforderlichen Änderungen durch und schließen Sie den PLC-Datentypen wieder.

"PLC-Datentyp" wurde in eine Bibliothek eingefügt

- Markieren Sie den zu bearbeitenden PLC-Datentypen mit der rechten Maustaste.
- Wählen Sie in dem Kontextmenü den Punkt "Typ bearbeiten".
- Führen Sie die erforderlichen Änderungen durch.
- Wählen Sie vor dem Schließen des Editors eine der folgenden Optionen.
 - Version freigeben
 - Änderungen verwerfen

Abbildung 4-5



4.6 Tipps zur Projektierung

4.6.1 Reduzierung von Powertags

Wenn Sie einen PLC-Datentypen erstellen, dann müssen nicht zwangsläufig alle erstellten Variablen in der HMI-Projektierung verwendet werden.

Nutzen Sie die Option nicht benötigter Variablen für das HMI Bediengerät zu deaktivieren. Dadurch können Sie die Anzahl an verwendeten Powertags deutlich reduzieren.

Abbildung 4-6

	Name	Data type	Default value	Accessible from HMI	Visible in HMI	Setpoint
1	Status_Automatic_Mode	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Status_Manual_Mode	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Status_Manual_On	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Status_Manual_Off	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	05_Res.	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	Status_Ready_To_Use	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	Status_Fault	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	08_Res.	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	Status_Pump	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	10_Res.	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	Pump_Number	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	Pump_Flow_Rate	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	Set_Automatic_Mode	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14	Set_Manual_Mode	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
15	Set_Manual_On	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
16	Set_Manual_Off	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

4.6.2 Umstieg von HMI UDT nach PLC-Datentypen

Sie haben eine bestehende Projektierung mit einem Bildbaustein, indem ein "HMI UDT" verwendet wird. Dieser "HMI UDT" soll durch einen PLC-Datentypen ersetzt werden.

Den geringsten Aufwand haben Sie, wenn der "PLC-Datentyp" die gleiche Struktur hat wie der HMI UDT. In diesem Fall können Sie den HMI UDT 1:1 durch den PLC-Datentypen ersetzen.

Wird der "HMI-UDT" durch einen PLC-Datentypen ersetzt, bei dem die Struktur im Vergleich zum "HMI UDT" unterschiedlich ist, dann müssen im Bildbaustein-Editor die Verbindungen zu den einzelnen Objekten neu erstellt werden.

Hinweis Ab WinCC (TIA Portal) V13 SP1 können PLC-Datentypen in Verbindung mit einer S7-1200 bzw. S7-1500 Steuerung verwendet werden.

Vorgehensweise

- Öffnen Sie den Datenbaustein, der die Struktur des "HMI UDTs" hat und kopieren Sie die zugehörige Struktur.
- Öffnen bzw. erstellen Sie einen PLC-Datentypen.
- Fügen Sie die zuvor kopierte "Daten-Struktur" in den PLC-Datentypen ein.

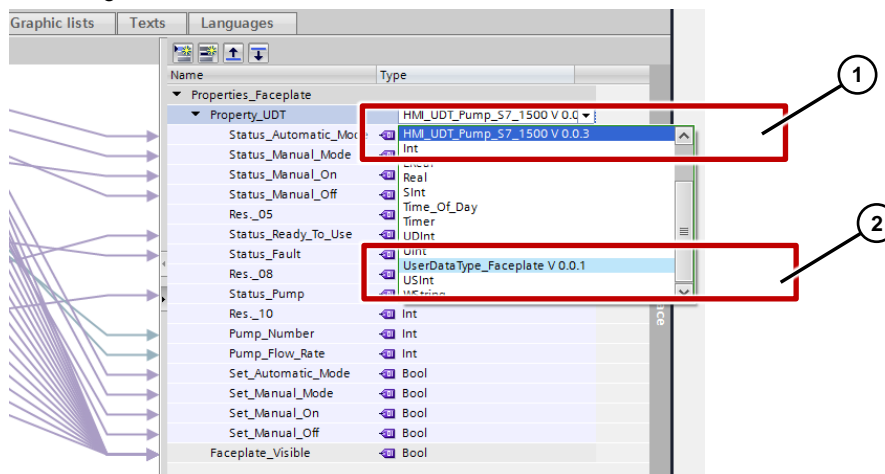
4 Funktionen und Eigenschaften eines PLC-Datentypen

4.6 Tipps zur Projektierung

- Ziehen Sie den erstellten PLC-Datentypen per Drag&Drop in Projektbibliothek.
- Öffnen Sie den Bildbaustein, bei dem der "HMI UDT" durch den PLC-Datentypen ersetzt werden soll.
- Öffnen Sie im Konfigurationsbereich im Register "Eigenschaften > Schnittstelle" die Variable mit dem "HMI UDT" (1).
- Wählen Sie unter "Typ" über die Klappliste den angelegten PLC-Datentypen aus (2). In diesem Beispiel "UserDataTyp_Faceplate".
Wenn alle Daten übereinstimmen, werden alle zugeordneten "Verbindungen" übernommen.

Hinweis Die Namen der Variablen dürfen nicht mit einer Ziffer beginnen.

Abbildung 4-7



5 Literaturhinweise

Internet-Link-Angaben

Diese Liste ist keinesfalls vollständig und spiegelt nur eine Auswahl an geeigneten Informationen wieder.

Tabelle 5-1

	Themengebiet	Titel
\1\	Referenz auf den Beitrag	https://support.industry.siemens.com/cs/ww/de/view/68014632
\2\	Siemens Industry Online Support	Industry Online Support https://support.industry.siemens.com
\3\	FAQ	Welche funktionalen Unterschiede gibt es zwischen den verschiedenen SIMATIC Panels? https://support.industry.siemens.com/cs/ww/de/view/40227286
\4\	FAQ	Wie können in WinCC (TIA Portal) Bildbausteine mit Benutzerberechtigungen erstellt werden? https://support.industry.siemens.com/cs/ww/de/view/57434982
\5\	Applikation	Applikation Beispielbausteine für STEP 7 und WinCC flexible. https://support.industry.siemens.com/cs/ww/de/view/36435784
\6\	Applikation	Bildbausteine zur Visualisierung von Sentron PAC Energie-Messgeräten https://support.industry.siemens.com/cs/ww/de/view/67318600

6 Historie

Tabelle 6-1

Version	Datum	Änderung
V1.0	01/2013	Erste Ausgabe
V2.0	06/2015	Update auf WinCC V13 SP1 (neue Funktion – PLC-Datentypen)
V2.0	08/2017	Hochrüstung V14