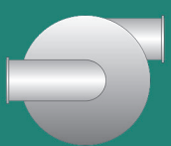
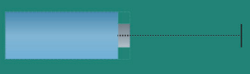
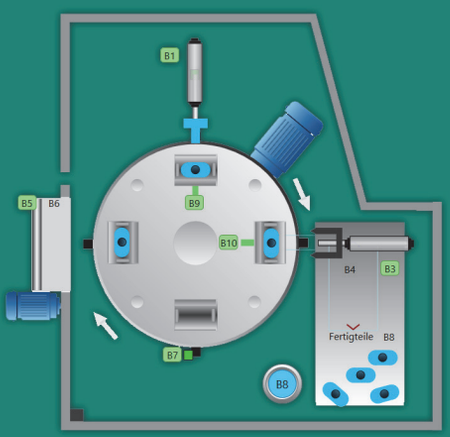
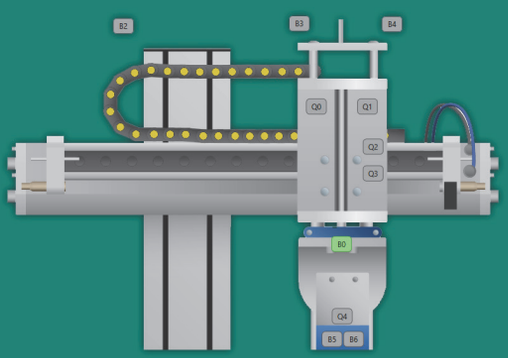
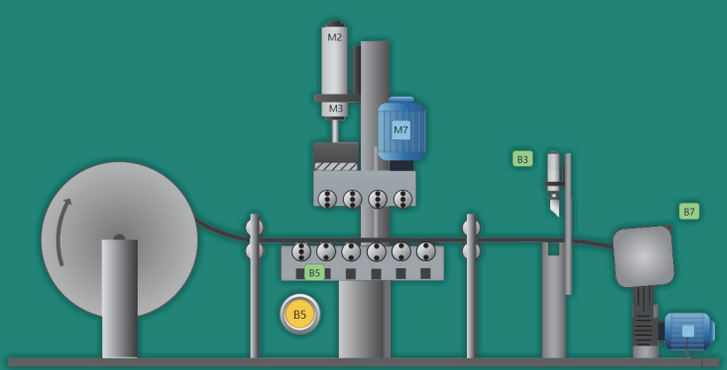




Aufbau von virtuellen Anlagen und mechatronischen Systemen mit PLC-Lab







MHJ-Software
GmbH & Co. KG
Albert-Einstein-Str. 101
75015 Bretten
www.mhj.de

Die PLC-Lab Beispiele können Sie hier herunterladen:

http://www.mhj-download.de/plclab/plc-lab-handbuch-beispiele.zip

oder

https://bit.ly/2RJ2MvE

© 2020 MHJ-Software GmbH & Co. KG • Albert-Einstein-Str. 101 • D-75015 Bretten • www.mhj.de

Kein Teil dieses Dokumentes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung von MHJ-Software reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Warenzeichen: STEP®, SIMATIC®, TIA-Portal®, S7-1200®, S7-1500®, S7-300® und S7-400® sind eingetragene Warenzeichen der SIEMENS Aktiengesellschaft.

Stand: 28.09.2020

Inhalt

1 PLC-Lab Oberfläche 4
2 Zeichnen einer Anlage 9
3 Devices in PLC-Lab 31
4 Verwendung von Symbolen in PLC-Lab 67
5 Die Schalter-Objekte 84
6 Die Lampen-Objekte 88
7 Der Endschalter bzw. Sensor in PLC-Lab 90
8 Geometrische Formen 93
9 Rohrleitungen 97
10 Zylinder 101
11 Förderelemente 113
12 Das Flüssigkeits-Objekt 122
13 Verbindungselemente bzw. Gelenke 128
14 Creator und Destroyer 161
15 Anzeige-Objekte 165
16 Eingabe-Objekte 172
17 Rechner-Objekt 176
18 Kollisionsgruppen 179
19 Farbverläufe und Transparenz 184
20 Verwendung von Grafiken in den Objekten 185
21 Anlagen vor Veränderungen schützen 188

1 PLC-Lab Oberfläche

1.1 Tools

Innerhalb der **Tools** sind in einer Baumstruktur die Objekte von PLC-Lab gelistet. Diese sind dabei nach Kategorien sortiert.

So befinden sich z. B. in der Kategorie „Lampen“ alle lampenähnlichen Objekte, beispielsweise Leuchtmelder, LEDs, Signal-Säulen usw. Alle in der Liste vorhandenen Objekte können für den Aufbau einer virtuellen Anlage in PLC-Lab verwendet werden.

Nach der Platzierung eines Objekts auf der Zeichenfläche kann man die Eigenschaften des Objekts verändern und so das Verhalten den eigenen Bedürfnissen anpassen.

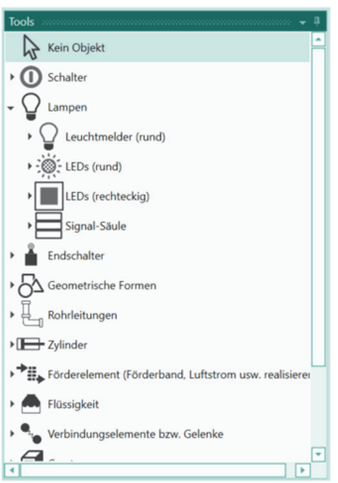


Bild: Tools

1.2 Aktionen

In der **Aktions-Liste** finden sich Funktionen und Aktionen, die beim Zeichnen einer Anordnung benötigt werden. Auf die Verwendung der einzelnen Aktionen wird noch explizit eingegangen, z. B. bei der Erläuterung von Zeichnen-Gruppen, Körper-Gruppen oder auch bei der Verwendung der intelligenten Führungslinien.

Auch die Aktions-Liste hat eine Baumstruktur. Die Unterpunkte einer Kategorie werden sichtbar, sobald man einen Knoten mit der Maus anklickt.

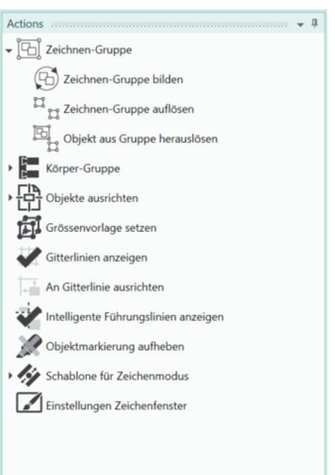


Bild: Aktions-Liste mit geöffneter Kategorie „Zeichnen-Gruppe“

1.3 Symboltabelle

In der **Symboltabelle** können die im Anlagenprojekt verwendeten Operanden mit Namen bzw. Bezeichnungen versehen werden. Die verwendete Bezeichnung sollte die Aufgabe des Operanden beschreiben. Das Symbol eines Operanden sollte aussagekräftig sein. Da ein vergebenes Symbol in der Regel nicht nochmals von Hand eingetippt werden muss, kann es auch etwas länger ausfallen (max. 80 Zeichen). Die Symbole werden entweder per Drag-and-Drop oder mit Hilfe der **Autovervollständigung** in den Eigenschaften der Objekte angegeben. Dies wird im weiteren Verlauf noch detailliert erläutert.

Jeder Operand und dessen Symbol sind einem Device zugeordnet. Innerhalb des Device müssen die Operanden und Symbole eindeutig sein. Ein Operand (bzw. Symbol) kann dagegen mehrmals vorhanden sein, sofern er unterschiedlichen Devices zugeordnet ist. Nachfolgend ist dies zu sehen:

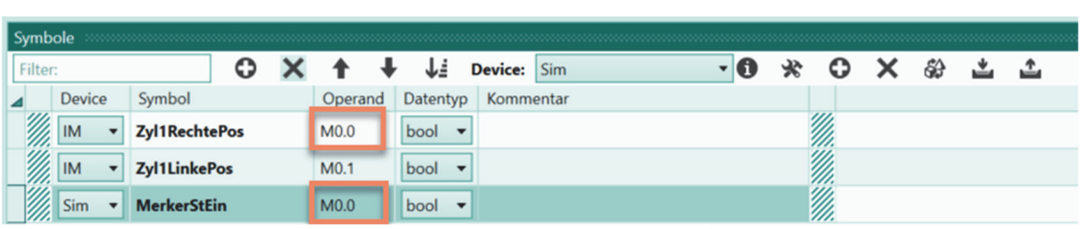


Bild: Symboltabelle mit doppelter Angabe für M0.0

Dabei wurde dem Merker M0.0 des **IM-Device** das Symbol „Zyl1RechtePos“ zugewiesen. Der Merker M0.0 des Sim-Device hat das Symbol „MerkerStEin“.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Möchte man Symbole aus dem TIA-Portal, dem Simatic-Manager, WinSPS-S7 oder GRAFCET-Studio importieren, dann stehen innerhalb der Symboltabelle entsprechende **Importfunktionen** zur Verfügung.

Auch der **Export** einer Symboltabelle von PLC-Lab ist möglich. Somit können die Symbole in die oben genannten Tools importiert werden.

Auf die genaue Verwendung von Symbolen und deren Vorteile wird in einem gesonderten Kapitel eingegangen.

1.4 Zeichenfläche

Die Zeichenfläche in PLC-Lab hat eine Größe von **5000 x 5000 Pixel**. Darauf können die Objekte der Anlage platziert und mit Hilfe von Verbindungselementen gekoppelt werden.

Rechts unten von der Zeichenfläche befindet sich eine **Miniaturansicht**. Diese bietet eine Gesamtansicht der Zeichenfläche und der Objekte. Über ein transparentes farbiges Rechteck wird der momentan sichtbare Bereich der Zeichenfläche symbolisiert. Mit Hilfe der Maus ist der Bereich verschiebbar, womit man den sichtbaren Ausschnitt der Zeichenfläche verändert.

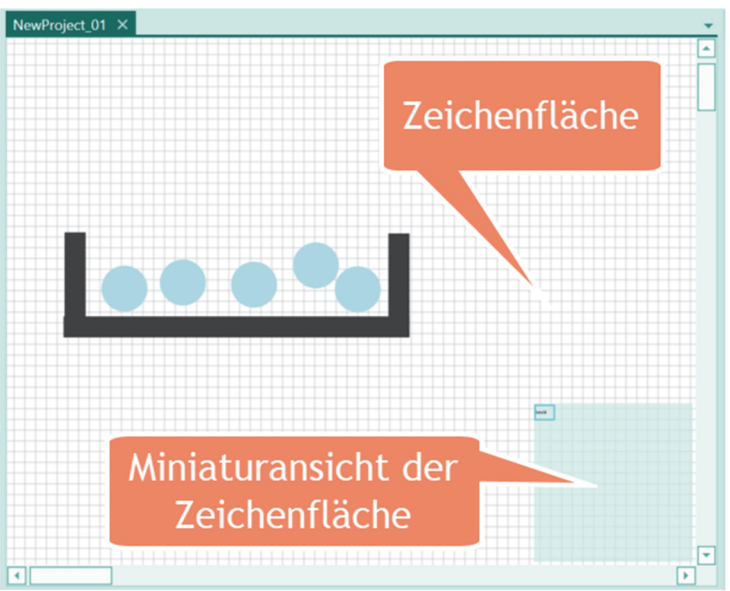


Bild: Zeichenfläche mit Miniaturansicht

Die Anzeige der Miniaturansicht kann mit Hilfe des Maus-Buttons „**Übersicht ein/aus**“ ein- bzw. abgeschaltet werden.

Standardmäßig sind Gitterlinien als Hintergrund innerhalb der Zeichenfläche dargestellt. Diese sollen das Platzieren der Objekte erleichtern. Ihr Abstand ist variabel, die Einstellung dazu kann im Dialog „Einstellungen Zeichenfenster“ verändert werden.

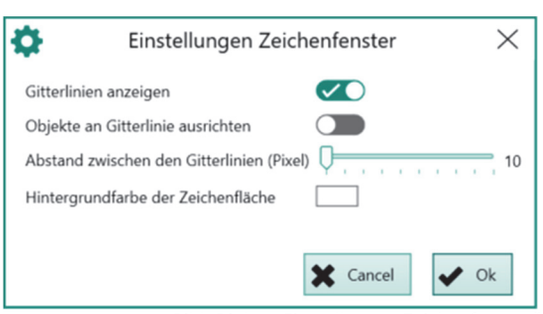


Bild: Dialog „Einstellungen Zeichenfenster“

Die Einstellung hat die Bezeichnung „**Abstand zwischen den Gitterlinien**“. Des Weiteren kann die Hintergrundfarbe der Zeichenfläche auf dem Dialog selektiert werden.

Tipps zum effektiven Zeichnen einer Anlage werden im weiteren Verlauf in einem eigenen Kapitel vorgestellt.

1.5 Eigenschaften der Objekte

Die meisten Objekte besitzen zahlreiche Einstellungsmöglichkeiten. Diese werden sichtbar, sobald ein Objekt auf der Zeichenfläche selektiert wurde.

In einem solchen Fall sind die Eigenschaften des selektierten Objekts innerhalb der Tabelle „**Eigenschaften Objekt**“ aufgeführt.

Je nach Art der Eigenschaft kann der Zahlenwert, die Farbe, die anzuzeigende Grafik oder der Operand angegeben werden. Dies wird im weiteren Verlauf der Beschreibung noch oft zur Anwendung kommen.

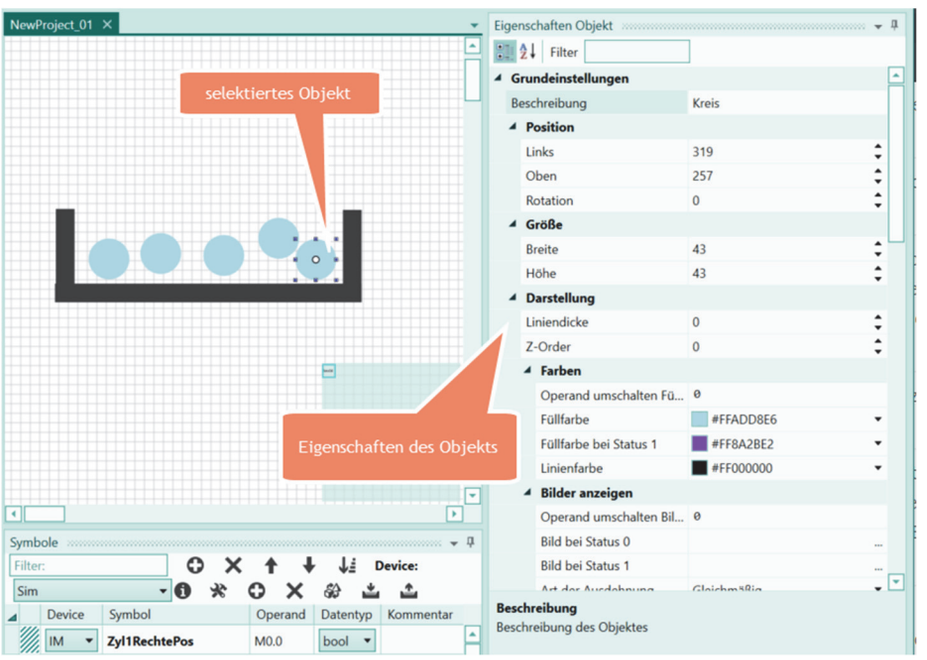


Bild: Selektiertes Objekt und dessen Eigenschaften

Wählt man eine Eigenschaft aus, dann erscheint im unteren Bereich der Tabelle ein Hilfetext.

Im Bild wurde dabei eine Eigenschaft für einen Operanden selektiert. Der hier angegebene Operand schaltet die Füllfarbe des Objekts um. Im Hilfetext ist auch ein Beispiel-Operand mit angegeben.

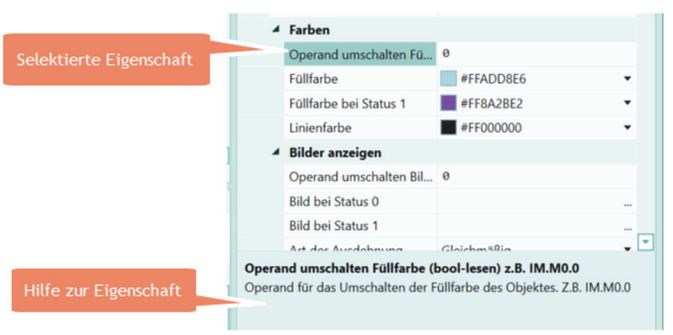


Bild: Selektierte Eigenschaft und der angezeigte Hilfetext

© MHJ-Software GmbH & Co. KG - PLC-Lab

1.5.1 Eigenschaften bei der Selektion von mehr als einem Objekt

Ein Sonderfall ist die Selektion von mehr als einem Objekt auf der Zeichenfläche. In diesem Fall sind ebenfalls Eigenschaften innerhalb der Tabelle zu sehen. Dabei handelt es sich um die Eigenschaften des zuerst selektierten Objekts. Allerdings werden dabei nicht alle Eigenschaften veränderbar sein.

Es sind nur die Eigenschaften zugänglich, welche in allen selektierten Objekten vorhanden **und** deren Werte identisch sind. Nachfolgend ein Beispiel: Zwei Schalter-Objekte sind selektiert. Die „Summe“ ihrer Eigenschaften ist im Bild zu sehen.

Im Bild sind diejenigen Eigenschaften mit einem Pfeil gekennzeichnet, welche veränderbar sind. Dazu gehört beispielsweise die Eigenschaft „Button ist ein Taster“. Wird diese nun verändert, dann hat dies Auswirkungen auf die Eigenschaft **beider** selektierten Objekte, d. h. beide Schalter-Objekte sind dann als Taster ausgelegt.

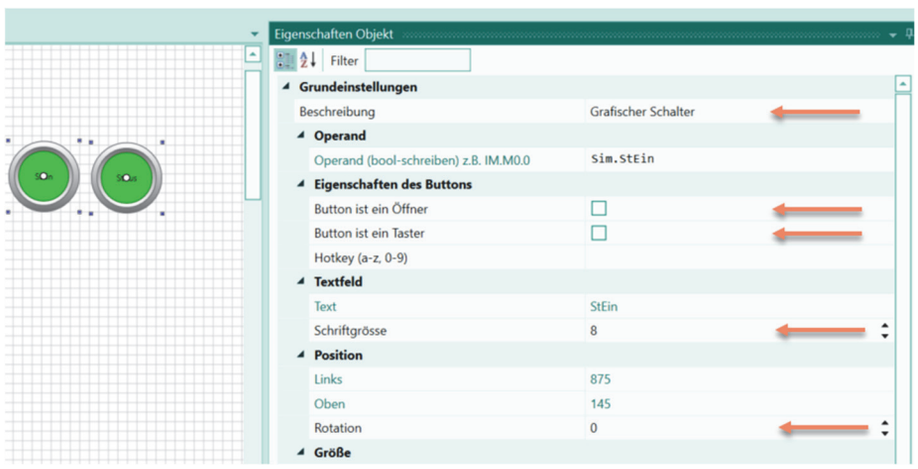


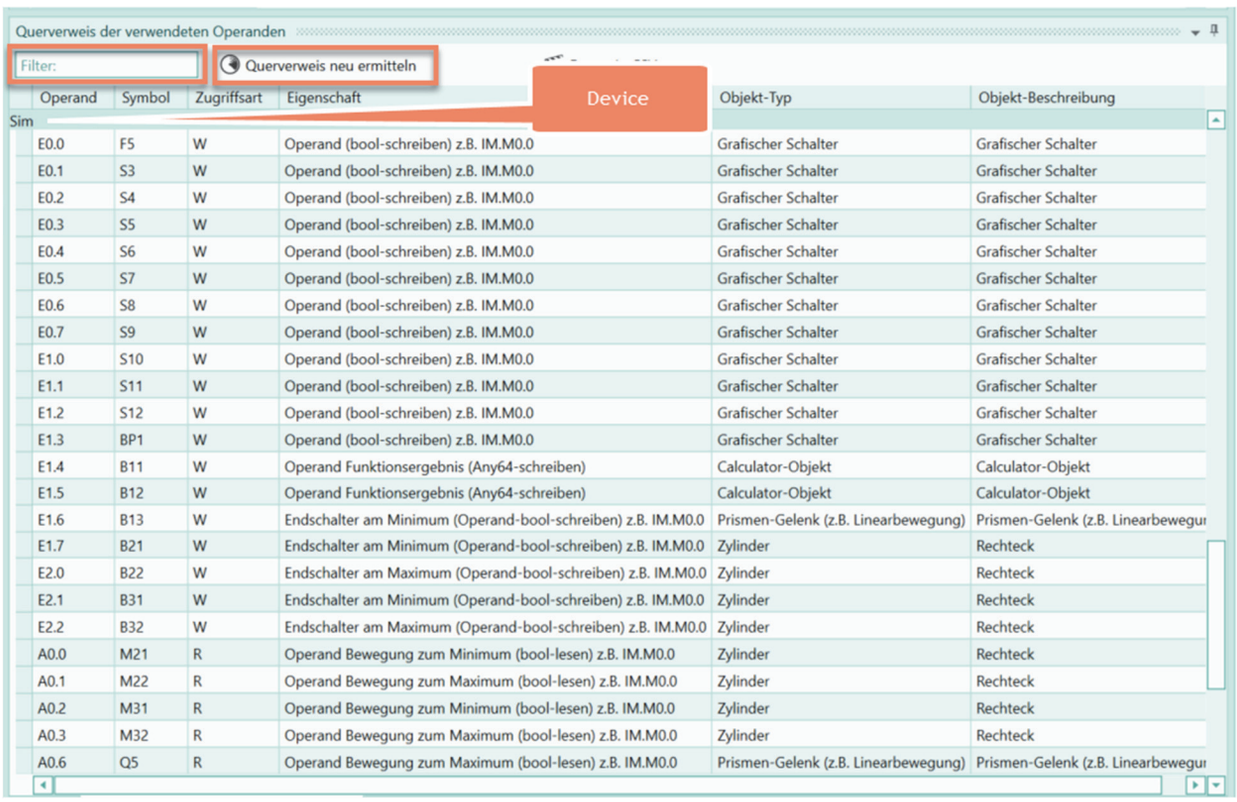
Bild: Anzeige der Eigenschaften der beiden selektierten Schalter-Objekte

Die Eigenschaft „Text“ ist **nicht veränderbar**, denn der Inhalt der Eigenschaft ist bei beiden Objekten **unterschiedlich:** Der eine Schalter ist mit „StEin“ der andere mit „StAus“ beschriftet.

Möchte man die Eigenschaften von mehreren gleichartigen Objekten auf die gleiche Weise verändern, dann selektiert man sie alle und ändert die Eigenschaft ab. Die Änderung hat dann Auswirkungen auf alle selektierten Objekte.

1.6 Querverweisliste

Die **Querverweisliste** ist unterhalb der Zeichenfläche von PLC-Lab angeordnet. Sie bietet eine Übersicht der in den Objekten verwendeten Operanden. Des Weiteren wird die **Zugriffsart** angezeigt. Dabei wird zwischen lesendem (Kürzel „R“ für read) und schreibendem Zugriff (Kürzel „W“ für write) unterschieden.
In der Spalte „Eigenschaft“ wird die Operanden-Eigenschaft benannt, in welcher der Operand angegeben ist.



Operand	Symbol	Zugriffsart	Eigenschaft	Objekt-Typ	Objekt-Beschreibung
Sim					
E0.0	F5	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.1	S3	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.2	S4	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.3	S5	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.4	S6	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.5	S7	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.6	S8	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E0.7	S9	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E1.0	S10	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E1.1	S11	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E1.2	S12	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E1.3	BP1	W	Operand (bool-schreiben) z.B. IM.M0.0	Grafischer Schalter	Grafischer Schalter
E1.4	B11	W	Operand Funktionsergebnis (Any64-schreiben)	Calculator-Objekt	Calculator-Objekt
E1.5	B12	W	Operand Funktionsergebnis (Any64-schreiben)	Calculator-Objekt	Calculator-Objekt
E1.6	B13	W	Endschalter am Minimum (Operand-bool-schreiben) z.B. IM.M0.0	Prismen-Gelenk (z.B. Linearbewegung)	Prismen-Gelenk (z.B. Linearbewegu
E1.7	B21	W	Endschalter am Minimum (Operand-bool-schreiben) z.B. IM.M0.0	Zylinder	Rechteck
E2.0	B22	W	Endschalter am Maximum (Operand-bool-schreiben) z.B. IM.M0.0	Zylinder	Rechteck
E2.1	B31	W	Endschalter am Minimum (Operand-bool-schreiben) z.B. IM.M0.0	Zylinder	Rechteck
E2.2	B32	W	Endschalter am Maximum (Operand-bool-schreiben) z.B. IM.M0.0	Zylinder	Rechteck
A0.0	M21	R	Operand Bewegung zum Minimum (bool-lesen) z.B. IM.M0.0	Zylinder	Rechteck
A0.1	M22	R	Operand Bewegung zum Maximum (bool-lesen) z.B. IM.M0.0	Zylinder	Rechteck
A0.2	M31	R	Operand Bewegung zum Minimum (bool-lesen) z.B. IM.M0.0	Zylinder	Rechteck
A0.3	M32	R	Operand Bewegung zum Maximum (bool-lesen) z.B. IM.M0.0	Zylinder	Rechteck
A0.6	Q5	R	Operand Bewegung zum Maximum (bool-lesen) z.B. IM.M0.0	Prismen-Gelenk (z.B. Linearbewegung)	Prismen-Gelenk (z.B. Linearbewegu

Bild: Beispiel einer Querverweisliste

Die Operanden werden zum Device zugehörig aufgelistet. Im Bild sind die Operanden des Sim-Device zu sehen.

Oberhalb der Tabelle kann man einen Filter angeben. Es werden dann nur die Operanden angezeigt, welche die Filterzeichen im Device-Namen, der Operanden-Kennzeichnung, dem Symbol, der Objektbeschreibung oder dem Eigenschaftsnamen beinhalten.

Ist der Querverweis nicht aktuell, so wird dies ebenfalls oberhalb der Tabelle angezeigt. In diesem Fall kann durch Betätigung der Schaltfläche „Querverweis neu ermitteln“ das Ermitteln angestoßen werden.

Durch einen **Doppelklick** auf eine Tabellenzeile wird in der Zeichenfläche das Objekt hervorgehoben, in welchem der Operand der Zeile verwendet wird.

Im folgenden Bild wurde auf der Zeile des Operanden „E0.1“ ein Doppelklick ausgeführt. Daraufhin wird in der Anlage das Schalter-Objekt mit dem Operanden hervorgehoben.

Bild: Das Schalter-Objekt mit dem Operanden E0.1 wird hervorgehoben.

Möchte man die Markierung wieder entfernen, dann führt man die Aktion „Objektmarkierung aufheben“ innerhalb der Aktionen aus.

© MHJ-Software GmbH & Co. KG - PLC-Lab

2 Zeichnen einer Anlage

In diesem Abschnitt sollen die Bedienschritte beim Zeichnen einer Anlage vorgestellt werden.

2.1 Objekt auf der Zeichenfläche platzieren und verändern

2.1.1 Objekt aufziehen bzw. platzieren

Um ein bestimmtes Objekt auf der Zeichenfläche zu platzieren, ist zunächst die zu zeichnende Objekt-Art innerhalb der Tools zu selektieren. Im Beispiel ist dies ein Schalter ohne Leuchteinsatz mit einer grauen Farbe.

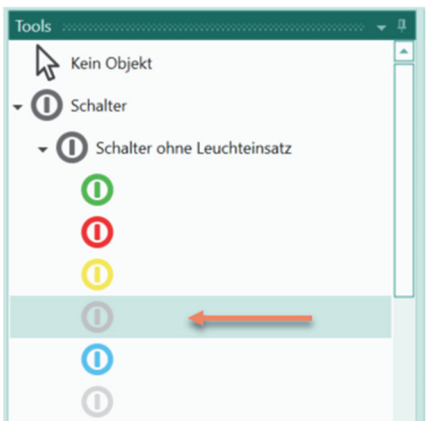


Bild: Selektion der Objekt-Art „Schalter ohne Leuchteinsatz“

Bewegt man nach der Auswahl der Objekt-Art die Maus in die Zeichenfläche, dann verändert sich die Form des Mauszeigers zu einem Kreuz. Dieses platziert man an der Stelle der Zeichenfläche, an welcher sich die linke obere Ecke des Objekts befinden soll. Dann wird die linke Maustaste betätigt, gehalten und das Objekt nach rechts unten aufgezogen.

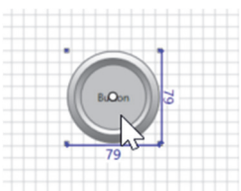


Bild: Platzierter Schalter auf der Zeichenfläche

Lässt man die linke Maustaste los, dann ist der Vorgang abgeschlossen und das Objekt behält die bis dahin eingestellte Größe bei.

2.1.2 Objekt mit der Tastatur verschieben

Möchte man ein Objekt auf der Zeichenfläche verschieben, dann kann man dies zum einen mit der Maus erledigen. Soll das Objekt aber nur in der Horizontalen oder Vertikalen verschoben werden, dann erreicht man dies am besten mit Hilfe der Pfeiltasten auf der Tastatur. Wird dabei eine der Pfeiltasten betätigt, dann verschiebt sich das selektierte Objekt jeweils um ein Pixel. Die Richtung ist vom verwendeten Pfeil auf der Tastatur abhängig.

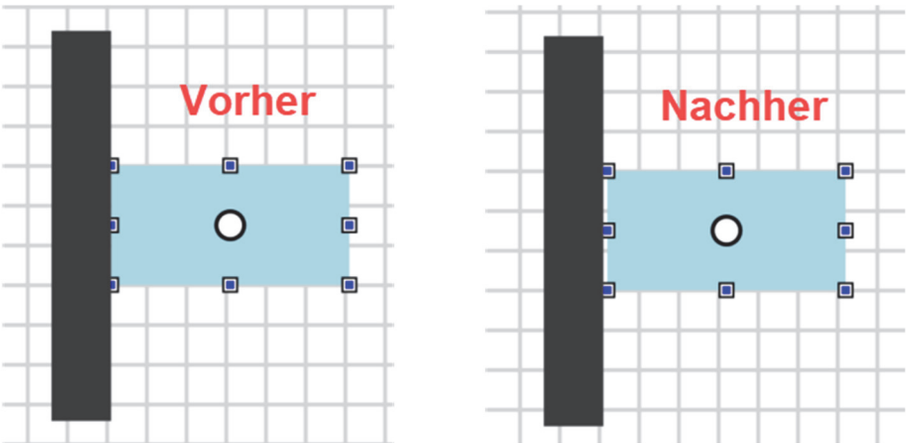


Bild: Selektiertes Objekt mit der Pfeil-Taste nach rechts verschoben

Im obigen Bild ist zu erkennen, dass sich das blaue Rechteck geringfügig, genauer gesagt um ein Pixel, nach rechts bewegt hat. Zwischen dem schwarzen Rechteck und dem blauen Rechteck ist somit eine kleine Lücke vorhanden.

Wird zusätzlich zur Pfeiltaste die [STRG]-Taste betätigt und gehalten, dann wird das Objekt um die Anzahl der Pixel des Zeichenrasters bewegt.

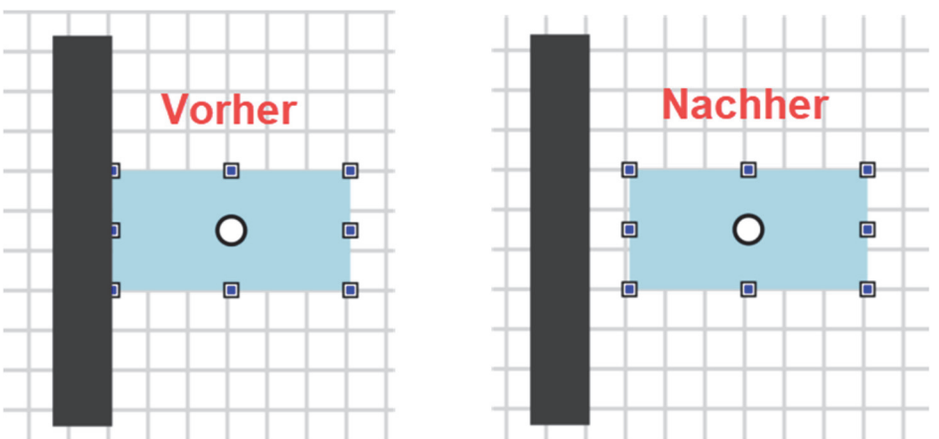


Bild: Selektiertes Objekt mit der Pfeil-Taste nach rechts verschoben, bei gleichzeitiger Betätigung der [STRG]-Taste

Nun ist der Abstand zwischen den beiden Rechtecken deutlich größer, denn durch die zusätzliche Betätigung der [STRG]-Taste wird das blaue Rechteck um eine Raster-Einheit (im Beispiel 10 Pixel) verschoben.

Hinweis: Sind mehrere Objekte selektiert, dann werden alle selektierten Objekte verschoben.

2.1.3 Objekt mit der Maus in der Größe verändern

Bevor ein Objekt mit der Maus in seiner Größe verändert werden kann, ist das Objekt zu selektieren. Denn nur im selektierten Zustand werden die für die Größenänderung notwendigen **Anfasser** angezeigt.

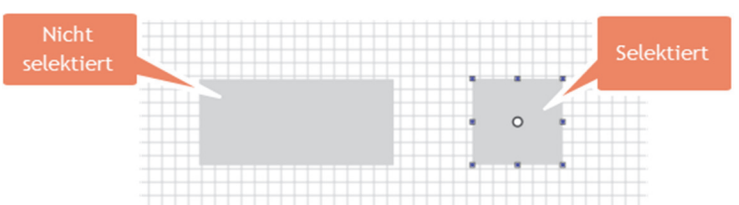


Bild: Linkes Objekt ist nicht selektiert, rechtes Objekt ist selektiert.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Diese **Anfasser** sind der „Angreif-Punkt“ für die Maus. Dabei ändert der Mauszeiger sein Aussehen.

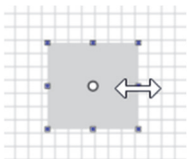


Bild: Mauszeiger über Anfasser für die Größenveränderung

Betätigt man nun zusätzlich die linke Maustaste, dann wird das Objekt in seiner Größe verändert. Die angefasste Seite folgt dann der Bewegung der Maus.

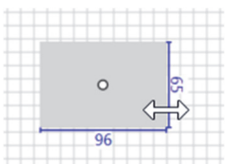


Bild: Objekt wird in der Breite verändert.

Gleichzeitig erscheint auch eine Bemaßung, damit man erkennt, welche Breite und Höhe das Objekt momentan besitzt.

Wird die linke Maustaste losgelassen, dann bleibt das Objekt in seiner Größe erhalten und die Bemaßungslinien verschwinden. Die Größenänderung ist somit abgeschlossen.

2.1.3.1 Verändern der Größe eines Objekts mit der Maus, bei gleichbleibenden Seitenverhältnissen

Möchte man die Größe eines Objekts verändern, wobei aber die Seitenverhältnisse konstant bleiben sollen, so erreicht man dies, indem beim Vorgang die [Shift]-Taste auf der Tastatur betätigt und gehalten wird.

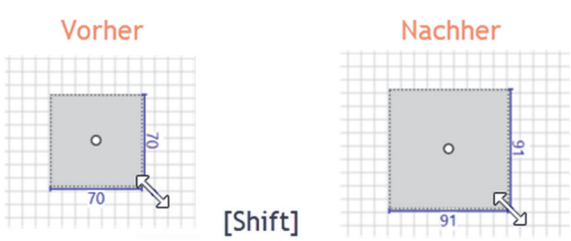


Bild: Größenänderung mit Beibehaltung der Seitenverhältnisse

Im Bild ist zu sehen, dass das Objekt zunächst eine Höhe und Breite von je 70 Pixeln besitzt. Höhe und Breite sind also identisch. Da bei der Größenänderung die [Shift]-Taste betätigt wurde, ist dies auch nach der Aktion der Fall.

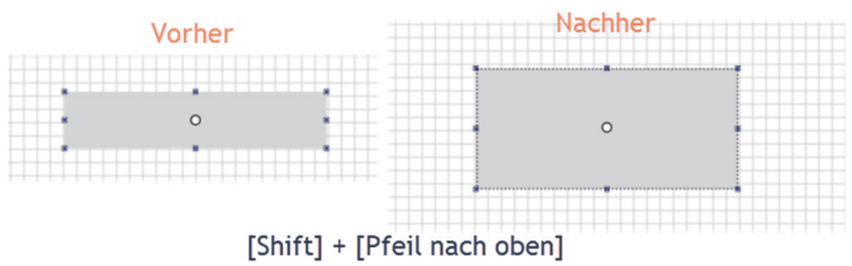
2.1.4 Objekt mit Tastatur in der Größe verändern

Selektierte Objekte können auch mit Hilfe der **Tastatur** in der Höhe oder Breite verändert werden. Dabei kommen die [Shift]-Taste und die Pfeiltasten zum Einsatz.

2.1.4.1 Die Höhe eines Objekts gleichmäßig mit der Tastatur verändern

Wird ein Objekt selektiert und danach die [Shift]-Taste sowie die **Pfeil-Taste** nach oben bzw. unten betätigt, dann wird das Objekt in der **Höhe** verändert. Die Änderung erfolgt dabei gleichmäßig an der oberen und unteren Seite des Objekts.

Bild: Verändern der Höhe eines Objekts



Im Bild wurde die [Shift]-Taste gehalten und mehrmals die Pfeil-Taste nach oben betätigt. Als Folge wird das Objekt gleichmäßig an der oberen und unteren Seite in der Höhe verändert. Die Lage des Mittelpunkts des Objekts verändert sich somit nicht.

Würde man die [Shift]-Taste und die Pfeil-Taste nach unten betätigen, dann würde die Höhe des Objekts gleichmäßig verringert werden.

2.1.4.2 Die Breite eines Objekts gleichmäßig mit der Tastatur verändern

Das Verändern der Breite eines Objekts mit Hilfe der Tastatur ist in ähnlicher Weise durchführbar.

Dabei kommen die beiden Pfeil-Tasten nach rechts und links zum Einsatz. Die Pfeil-Taste nach rechts erhöht die **Breite** eines Objekts, die Pfeil-Taste nach links verringert diese. Dabei ist immer die [Shift]-Taste gedrückt zu halten. Das Verändern der Breite geschieht gleichmäßig auf der rechten und linken Seite des Objekts. Die Lage des Mittelpunkts wird dabei nicht verändert.

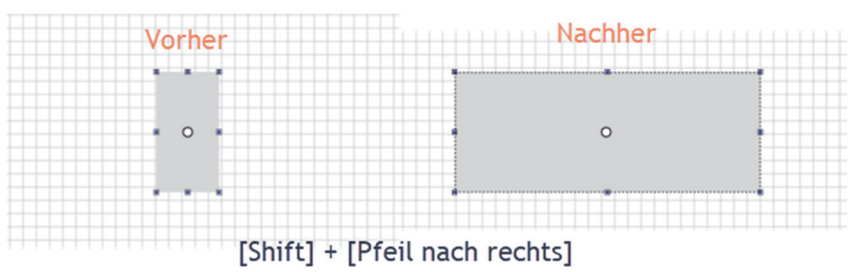


Bild: Verändern der Breite eines Objekts

2.1.5 Größenvorlagen für Objekt-Arten verwenden

Die Größenvorlage für eine Objekt-Art kommt immer dann zum Einsatz, wenn man Objekte in gleicher Größe zeichnen möchte. Am häufigsten ist dies bei Schalter- oder Lampen-Objekten der Fall.

Die Vorgehensweise ist dabei wie folgt:

1. Ein Objekt dieser Objekt-Art in der gewünschten Größe auf der Zeichenfläche zeichnen und das Objekt selektieren.
2. Die Aktion „Größenvorlage setzen“ in der Aktions-Liste anklicken.
3. Das Objekt wird als Größenvorlage für diese Objekt-Art gesetzt und eine entsprechende Meldung in der Status-Zeile ausgegeben.

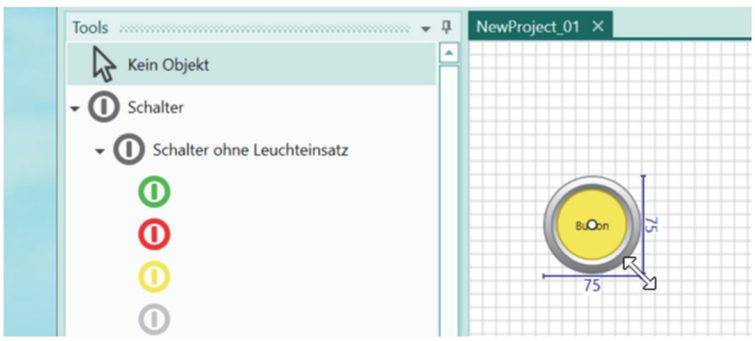
© MHJ-Software GmbH & Co. KG - PLC-Lab

2.1.5.1 Beispiel zum Setzen einer Größenvorlage

Im Beispiel soll ein Schalter-Objekt platziert und als Größenvorlage für andere Schalter verwendet werden.

Im ersten Schritt ist das Schalter-Objekt in den Tools zu selektieren, wobei es sich um einen Schalter ohne Leuchteinsatz in gelber Farbe handelt. Dieser wird auf der Zeichenfläche aufgezogen, bis er eine Breite und Höhe von 75 Pixeln besitzt.

Bild: Schalter wird auf der Zeichenfläche platziert.



Hat der Schalter die passende Größe, dann wird die linke Maustaste losgelassen. Der Schalter ist noch selektiert.

Nun wird innerhalb der Aktions-Liste der Eintrag „Größenvorlage setzen“ angewählt.

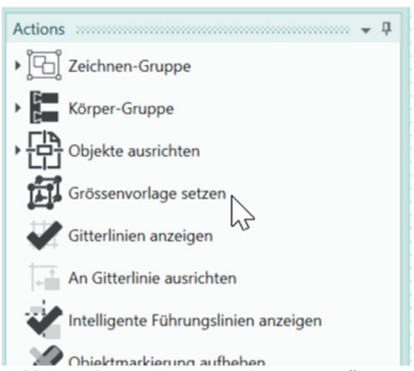


Bild: Die Aktion „Grössenvorlage setzen“ auswählen

Als Folge erscheint im Statusfeld die Meldung, dass die Größenvorlage für die Schalter-Objekte gesetzt wurde.

Bild: Bestätigung der Aktion

Somit werden alle Schalter-Objekte mit dieser Größe auf der Zeichenfläche platziert. Dabei muss der Schalter nicht mehr mit der Maus aufgezogen werden. Es genügt nach der Selektion des zu zeichnenden Schalters ein Mausklick auf der Zeichenfläche. Dies wird im nachfolgenden Bild gezeigt.

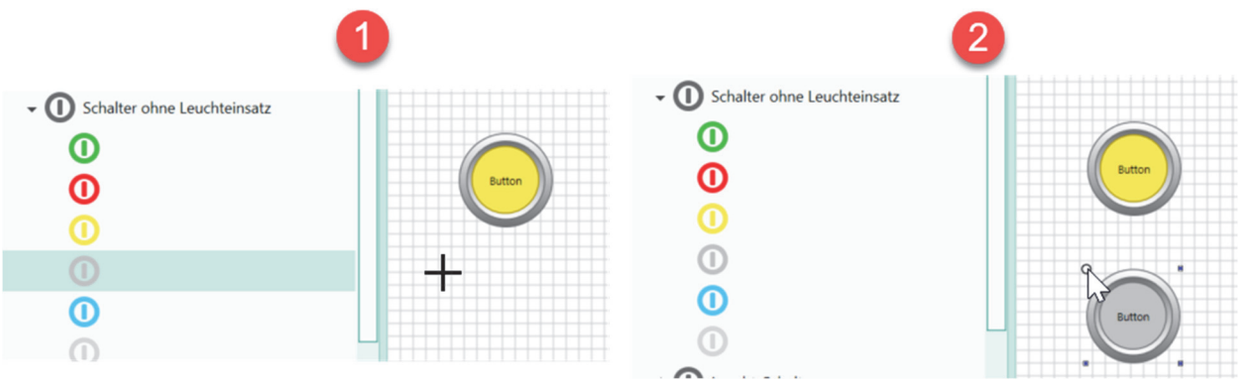


Bild: Auswahl und Platzierung eines weiteren Schalter-Objekts

Links im Bild (Punkt 1) wird der graue Schalter innerhalb der **Tools** selektiert. Auf der rechten Seite (Punkt 2) wurde der Schalter durch einen Klick auf der Zeichenfläche platziert. Dabei wurde mit der Maus auf die gewünschte linke obere Ecke geklickt. Das Einstellen der Größe erfolgte automatisch, so wie in der Größenvorlage für die Objekt-Art festgelegt.

2.1.6 Objekte rotieren mit der Maus (frei und in 15°-Schritten)

Wie schon erwähnt, erkennt man ein selektiertes Objekt anhand der Anfasser, die am Rand des Objekts angeordnet sind.

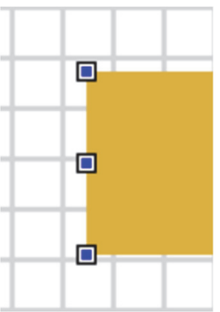


Bild: Aussehen der Anfasser auf der linken Seite eines selektierten Objekts

Haben die Anfasser das im obigen Bild dargestellte Aussehen, dann kann man sie mit der Maus anklicken und so das Objekt in der Größe verändern.

Wird auf einem Objekt im selektierten oder nicht-selektierten Zustand ein Doppelklick ausgeführt, dann werden Anfasser mit einem veränderten Aussehen am äußeren Rand des Objekts dargestellt:

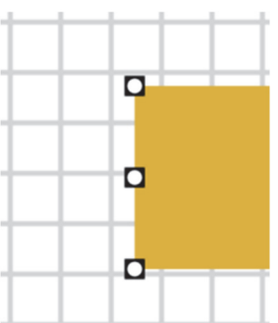


Bild: Aussehen der Anfasser nach einem Doppelklick.

Diese Anfasser signalisieren, dass sich das Objekt im **Rotations-Modus** befindet.

Bewegt man die Maus über einen Anfasser, dann verändert dieser sein Aussehen wie folgt:

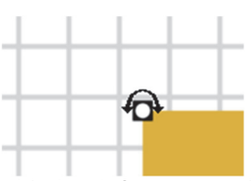


Bild: Mauszeiger im Bereich eines Anfassers, wenn der Rotationsmodus aktiv ist

Wird in diesem Zustand die **linke Maustaste betätigt und gehalten**, dann kann man das Objekt rotieren. Als zusätzliche Information ist die aktuelle Rotation des Objekts zu sehen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

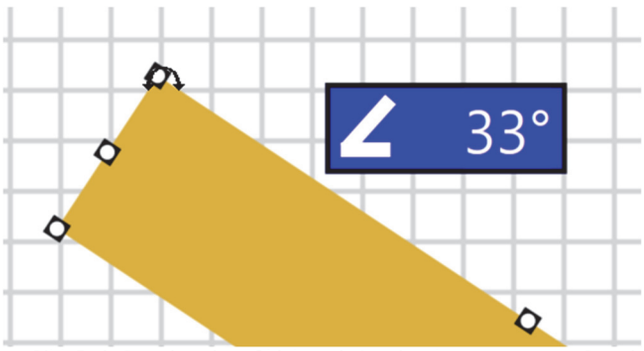


Bild: Freie Rotation eines Objekts mit Anzeige der momentanen Rotation

Lässt man die linke Maustaste los, dann bleibt das Objekt in dieser Lage und auch die Anzeige des Rotations-Grads verschwindet. Klickt man auf eine freie Stelle der Zeichenfläche, dann wird das Objekt deselektiert und der Rotationsmodus verlassen. Ähnliches geschieht bei einem Doppelklick auf das Objekt: Auch hier wird der Rotationsmodus verlassen, allerdings bleibt das Objekt selektiert.

2.1.6.1 Rotation in 15°-Schritten

Die zuletzt durchgeführte Rotation wird als freie Rotation bezeichnet. Dabei kann man ein Objekt in 1°-Schritten rotieren. Möchte man das Objekt in 15°-Schritten rotieren, dann wird dies durch Betätigung der [Shift]-Taste erreicht.

Ist die [Shift]-Taste betätigt, dann ändert das Objekt erst seine Rotation, wenn mit der Maus eine Rotations-Bewegung von mehr als 15° ausgeführt wurde. Dabei wird dann das Objekt um 15° rotiert.

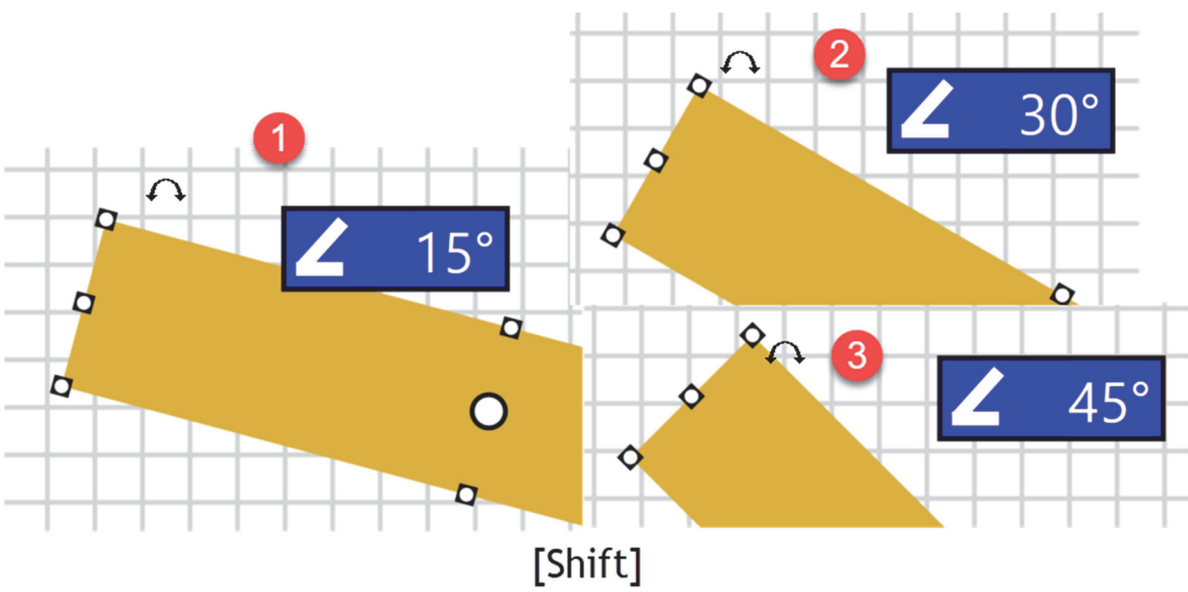


Bild: Rotieren in 15°-Schritten bei betätigter [Shift]-Taste

Im Bild wird die Rotation in 15°-Schritten bei betätigter [Shift]-Taste durchgeführt.

2.1.7 Mehrere Objekte selektieren mit der Maus

Hat man ein Objekt selektiert und ein weiteres soll zusätzlich selektiert werden, dann erreicht man dies durch **Halten der [Shift]-Taste** beim Anklicken des zweiten und jedes weiteren Objekts.

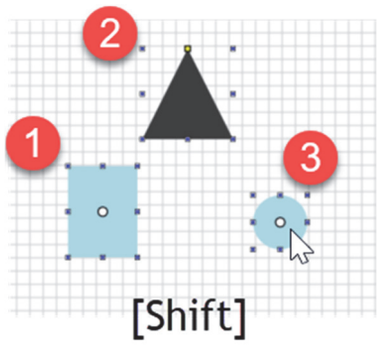


Bild: Selektion von mehreren Objekten durch Anklicken mit der Maus bei betätigter [Shift]-Taste

Im Bild wurde zunächst das erste Objekt (blaues Rechteck) selektiert, anschließend bei betätigter [Shift]-Taste das Dreieck angeklickt. Und schließlich der Kreis, ebenfalls bei betätigter [Shift]-Taste. Als Folge werden alle angeklickten Objekte als selektiert dargestellt.

2.1.8 Mehrere Objekte selektieren über das Maus-Gummiband

Mit Hilfe des Maus-Gummibands ist es ebenfalls möglich, mehrere Objekte zu selektieren. Das Maus-Gummiband wird mit Hilfe der betätigten [Shift]-Taste aufgezogen. Dabei platziert man die Maus am linken oberen Startpunkt, betätigt die [Shift]-Taste und hält sie betätigt. Nun wird die linke Maustaste gedrückt und die Maus nach rechts unten bewegt. Dabei wird ein „Gummiband“ sichtbar, welches einen Bereich auf der Zeichenfläche symbolisiert.

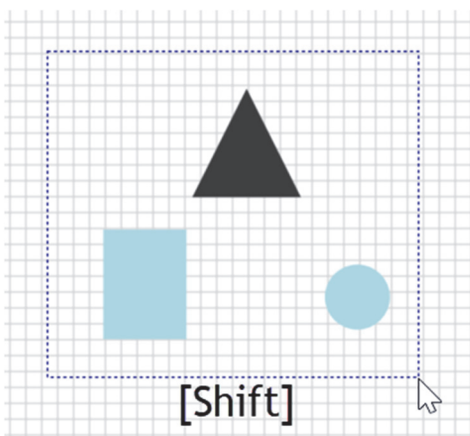


Bild: Aufziehen des Maus-Gummibands bei betätigter [Shift]-Taste

Alle Objekte innerhalb dieses Bereichs werden selektiert, sobald die linke Maustaste losgelassen wird.

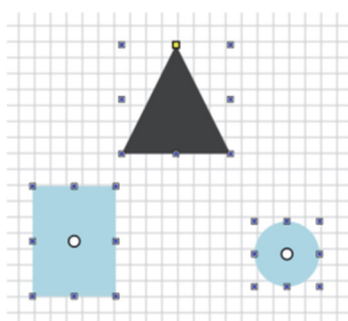


Bild: Linke Maustaste wurde losgelassen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

2.1.9 Selektieren aller Objekte

Gelegentlich ist es von Nutzen, alle Objekte einer Anlage zu selektieren. Beispielsweise dann, wenn die Position der gesamten Anlage auf der Zeichenfläche verschoben werden soll, um weitere Anlagenteile anzubringen.

Die im letzten Abschnitt beschriebene Methode mit Hilfe des Maus-Gummibands wäre eine Möglichkeit. Die schnellste Variante ist allerdings das Betätigen der **Tastenkombination [STRG] + [A].** Nach Ausführung sind **alle** Objekte der aktuellen Anlage **selektiert**.

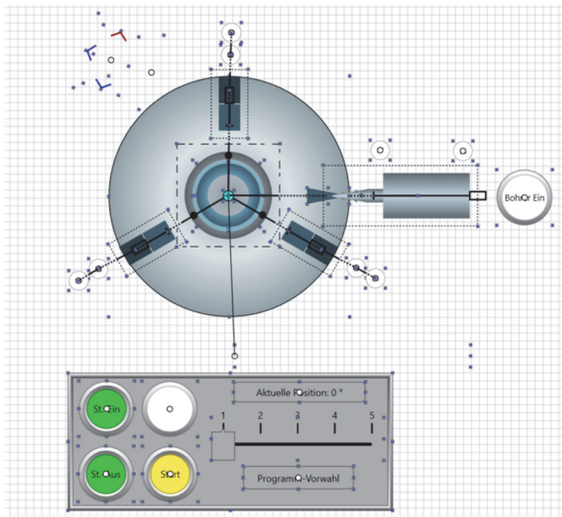


Bild: Alle Objekte wurden über [STRG] + [A] selektiert.

2.1.10 Löschen einzelner oder mehrerer Objekte

Das **Löschen** von Objekten gestaltet sich recht einfach: Man selektiert die zu löschenden Objekte und betätigt die **[Entf]-Taste**.

2.1.11 Z-Order der Objekte verändern

Die **Z-Order** eines Objekts kann die Wert -20 bis 20 annehmen. Sie ist ein Maß dafür, in welcher Ebene sich das Objekt befindet. Beim Erzeugen eines Objekts hat es den Z-Order-Wert 0. Damit würde es von allen Objekten mit einer höheren Z-Order (1 bis 20) verdeckt werden. Hat ein anderes Objekt ebenfalls den Z-Order Wert 0, dann überdeckt das jüngere Objekt das ältere Objekt.

Die Z-Order befindet sich innerhalb der Eigenschaften in der Rubrik „Grundeinstellungen->Darstellung.“

2.1.11.1 Beispiel zur Z-Order

Im folgenden Bild hat der Kreis eine Z-Order mit dem Wert 1, das Dreieck besitzt die Z-Order mit dem Wert 0.

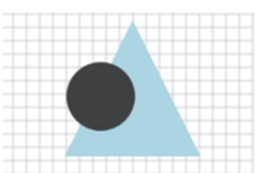


Bild: Anordnung mit Kreis und Dreieck

Da der Kreis die höhere Z-Order besitzt, verdeckt er Teile des Dreiecks.

2.2 Zoomen der Zeichenfläche

Die Objekte von PLC-Lab sind **vektorbasiert**. Dies hat den großen Vorteil, dass die Objekte beliebig zoombar sind, ohne dass ihre Darstellungsqualität darunter leidet.

Die Möglichkeit des Zoomens kann man sich auch während des Erstellens von Anlagen zu Nutze machen. Zum einen hat man die Möglichkeit, beim Zeichnen von relativ kleinen Objekten den Zeichenbereich so stark zu vergrößern, dass sie komfortabel gehandhabt werden können. Zum anderen kann man sich durch starkes Verkleinern der Zeichenfläche einen Überblick über die gesamte Anlage verschaffen.

Nachfolgend werden die Bedienmöglichkeiten innerhalb von PLC-Lab gezeigt, mit denen die unterschiedlichen Zoom-Grade einstellbar sind.

2.2.1 Zoomen mit dem Slider

Am rechten unteren Fensterrand von PLC-Lab befindet sich ein Slider, mit dem die Ansicht der Zeichenfläche vergrößert bzw. verkleinert werden kann. Der aktuelle Zoom-Faktor wird links neben dem Slider angezeigt.

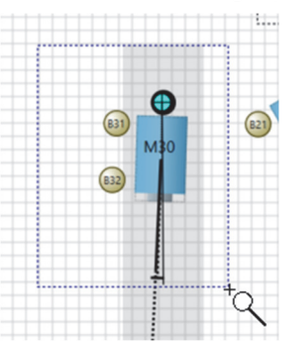
Bild: Slider für die vergrößerte bzw. verkleinerte Darstellung der Zeichenfläche

Wird der Slider verwendet, dann liegt der Ursprung für die Zoom-Aktion im linken oberen Ecke des momentan dargestellten Zeichenbereichs.

2.2.2 Zoomen mit dem Maus-Gummiband

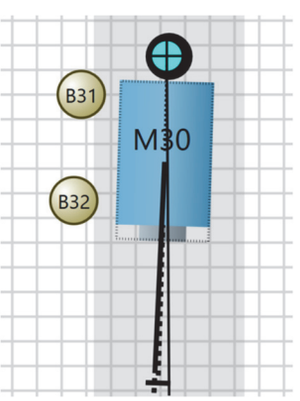
Das Maus-Gummiband wurde schon im Zusammenhang mit dem Selektieren von Objekten erwähnt. Wird beim Aufziehen anstatt der [Shift]-Taste die [STRG]-Taste betätigt, dann verändert der Mauszeiger sein Aussehen zu einer **Lupe**.

Bild: Maus-Gummiband bei betätigter [STRG]-Taste



Wenn man nun das Maus-Gummiband aufzieht und die linke Maustaste loslässt, dann wird der Bereich des Gummibands max. vergrößert.

Bild: Bereich des Maus-Gummiband wird vergrößert



© MHJ-Software GmbH & Co. KG - PLC-Lab

2.2.3 Zoomen mit dem Maus-Rad

Eine weitere Möglichkeit, den **Zoom-Faktor** zu beeinflussen, besteht über das **Maus-Rad**. Platziert man die Maus an einer freien Stelle der Zeichenfläche, betätigt und hält die [STRG]-Taste und bewegt das Maus-Rad, dann wird je nach Drehrichtung der Zoom-Faktor verringert oder erhöht.

Die Position der Maus markiert dabei den linken oberen Ursprungspunkt.

2.3 Objekte duplizieren und kopieren

In PLC-Lab sind zahlreiche Varianten vorhanden, um eine oder mehrere Objekte zu duplizieren.

2.3.1 Kopieren von einem oder mehreren Objekten mit Hilfe der Maus

Möchte man eines oder mehrere Objekte mit der Maus kopieren, dann selektiert man zunächst die zu kopierenden Objekte. Danach betätigt man die [STRG]-Taste und hält sie gedrückt. Anschließend bewegt man die Maus über eines der zu kopierenden Objekte, betätigt die linke Maustaste und verschiebt das Objekt. Als Folge werden alle selektierten Objekte kopiert und bis zum Loslassen der linken Maustaste verschoben.

Bild: Die drei Ursprungs-Objekte werden mit der Maus kopiert.

2.3.2 Kopieren mit Hilfe der Zwischenablage (Copy/Paste)

Eine weitere Möglichkeit, die selektierten Objekte zu kopieren, ist die Variante Copy/Paste, welche über die Standard-Tastenkombination [STRG] + [C] / [STRG] + [V] ausgelöst wird. Die Tastenkombination [STRG] + [C] kopiert dabei die selektierten Objekte in die Zwischenablage. Mit [STRG] + [V] werden die in der Zwischenablage befindlichen Objekte in die Zeichenfläche eingefügt.

Hinweis: Mit dieser Variante kann man auch Objekte aus einem anderen PLC-Lab-Projekt kopieren. Werden dabei Operanden eines Device in den Objekten verwendet, welches in dem neuen Projekt noch nicht vorhanden ist, dann wird auch das Device dem neuen Projekt hinzugefügt. Gleiches gilt auch für Symbole: Werden in den kopierten Objekten Symbole verwendet, welche im neuen Projekt noch nicht vorhanden sind, dann werden diese Symbole in der Symboltabelle eingefügt.

2.3.3 Duplizieren von Objekten mit Hilfe von [STRG] + [D]

Diese Art des **Duplizierens** ist insbesondere beim Platzieren von Schalter- und Lampen-Objekten interessant. Ein neu erzeugtes Objekt wird dabei rechts neben dem Ursprungs-Objekt angeordnet.

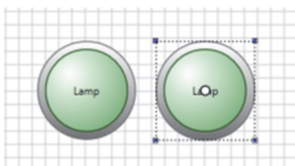


Bild: Duplizieren über [STRG] + [D], links das Original-Objekt, rechts das Duplikat

Betätigt man mehrmals die Tasten [STRG] + [D], so hat man relativ zügig eine Reihe von Objekten platziert, welche alle in einer Reihe mit dem gleichen Abstand zueinander angeordnet werden. Möchte man das Duplikat unterhalb des Original-Objekts platzieren, dann ist dies ebenfalls möglich. Dabei wird zusätzlich zu [STRG] + [D] auch die [Shift]-Taste gehalten.

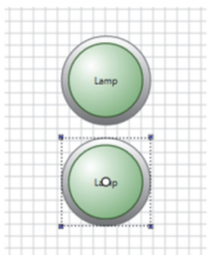


Bild: Duplizieren über [Shift] + [STRG] + [D], oben das Original-Objekt, unten das Duplikat

Mit diesen beiden Tastenkombinationen können somit auch mehrzeilige Anordnungen von Objekten relativ zügig umgesetzt werden.

2.3.4 Verbindungselemente kopieren

Die **Verbindungselemente** (Verbindung für lineare Bewegung, Verbindung für Drehbewegung usw.) nehmen beim Kopieren eine Sonderstellung ein.

Diese Objekte können nicht einzeln kopiert werden. Sie werden automatisch kopiert, sobald beide Anker-Objekte am Kopiervorgang beteiligt sind.

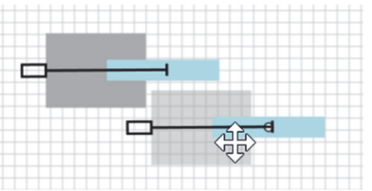


Bild: Kopiervorgang mit Verbindung für lineare Bewegung

Im obigen Bild sind zwei Rechtecke über eine Verbindung für lineare Bewegung miteinander verbunden. Beim Kopiervorgang mit Hilfe der Maus waren beide Objekte selektiert. Somit wird beim Kopieren auch die Verbindung mitkopiert.

2.4 Verwendung der intelligenten Führungslinien

Die **Führungslinien** können über die Aktion „Intelligente Führungslinien anzeigen“ ein- bzw. ausgeschaltet werden. Ist ein Haken an dem Symbol innerhalb der Aktions-Liste vorhanden, dann ist die Funktion eingeschaltet.

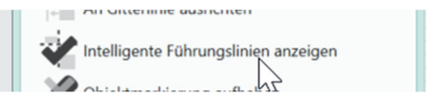


Bild: Symbol für die Aktion „Intelligente Führungslinien anzeigen“

Ist die Aktion eingeschaltet, dann werden beim Verschieben von Objekten mit der Maus immer dann Hilfslinien sichtbar, sobald die Eckpunkte des verschobenen Objekts mit umliegenden Objekten identisch sind.

Beispiel:

Im folgenden Bild soll das blaue Rechteck so platziert werden, dass dessen linke Seite mit der rechten Seite des schwarzen Rechtecks eine Linie bildet.

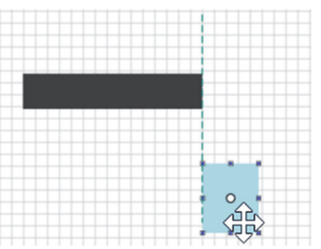


Bild: Hilfslinie für das Beispiel

Wird die linke Maustaste in der obigen Situation losgelassen, dann befindet sich das blaue Rechteck an der gewünschten Position.

Beispiel:

Im nächsten Beispiel soll das blaue Rechteck so angeordnet sein, dass dessen linke Seite eine Linie mit der rechten Seite des schwarzen Rechtecks bildet. Zusätzlich soll sich der Mittelpunkt des blauen Rechtecks auf gleicher Höhe mit der Oberseite des grünen Rechtecks befinden.

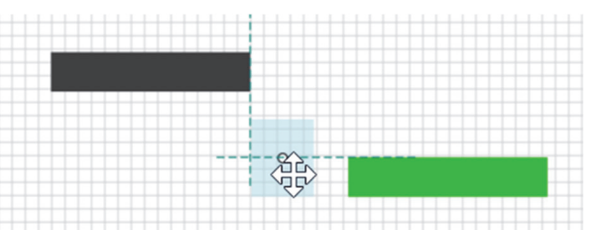


Bild: Platzierung für die genannte Aufgabenstellung

Im Bild ist zu erkennen, dass eine horizontale Führungslinie erscheint, sobald der Mittelpunkt des blauen Rechtecks die gleiche Höhe wie die Oberseite des grünen Rechtecks besitzt. In der gezeigten Darstellung kann die linke Maustaste losgelassen und so das Verschieben beendet werden.

Hinweis: Die intelligenten Führungslinien werden in einem gewissen Toleranzbereich angezeigt, damit die Platzierung einfacher ist. Ist eine Führungslinie sichtbar und das Verschieben wird beendet, dann wird das Objekt noch in die genaue Position gebracht.

© MHJ-Software GmbH & Co. KG - PLC-Lab

2.4.1 Vorteil der Führungslinien

Die intelligenten **Führungslinien** unterstützen bei der genauen Platzierung eines Objekts, wenn es zu Eckpunkten oder Seiten eines anderen Objekts ausgerichtet werden soll. Die genaue Positionierung des Objekts erfolgt, wenn die linke Maustaste losgelassen wird. Das Ganze funktioniert auch, wenn die Objekte rotiert dargestellt werden.

Die Führungslinien werden nur zu Objekten angezeigt, die in der momentanen Ansicht sichtbar sind. Es wird immer nur eine senkrechte und eine waagerechte Linie angezeigt. Sind mehrere senkrechte oder waagerechte Linien möglich, dann werden nur die Linien zu dem am nächsten platzierten Objekt angezeigt.

2.4.2 Wann sollten die Führungslinien abgeschaltet werden?

Möchte man Objekte **leicht versetzt** zu Eckpunkten anderer Objekte platzieren, dann können die Führungslinien störend sein. Denn in diesem Fall würden die Linien erscheinen und dann das Objekt bei Beendigung des Verschieb-Vorgangs an ihnen ausgerichtet werden. Um dies zu verhindern, sollten man bei einem solchen Fall die Funktion „Führungslinien anzeigen“ innerhalb der Aktionen abschalten.

2.5 Objekte ausrichten

Die Rubrik „Objekte ausrichten“ findet sich in den Aktionen von PLC-Lab.

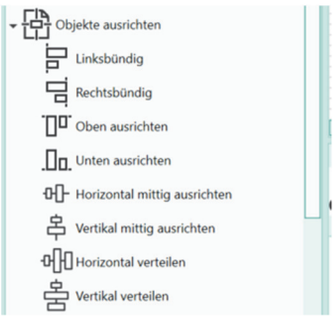


Bild: Aktionen innerhalb der Rubrik „Objekte ausrichten“

Einige Aktionen sind selbsterklärend, wie z. B. „Linksbündig“, „Rechtsbündig“ usw.

2.5.1 Verwendung der Funktionen zum Platzieren von Bedienelementen

Die Ausrichtungsfunktionen finden häufig Anwendung beim Platzieren von Bedienelementen, wie z. B. Schaltern oder Lampen.

2.5.1.1 Erstes Beispiel zum Ausrichten von Objekten

Eine Anordnung von Lampen soll horizontal angeordnet werden. Nach den Aktionen sollen die Lampen in einer Höhe und mit gleichem Abstand platziert sein. Folgendes Bild zeigt die Ausgangssituation:

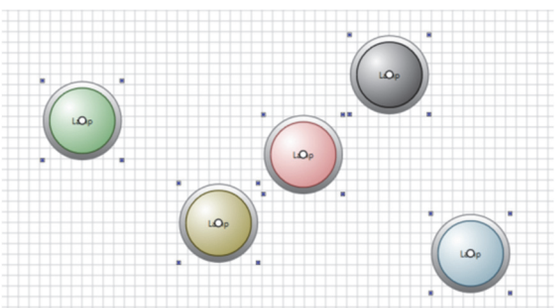


Bild: Ausgangssituation

Schritt 1: Aktion „Horizontal mittig ausrichten“ ausführen.

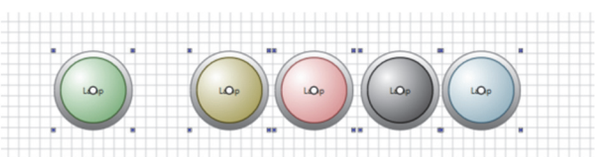


Bild: „Horizontal mittig ausrichten“ wurde ausgeführt.

Schritt 2: Aktion „Horizontal verteilen“ ausführen.

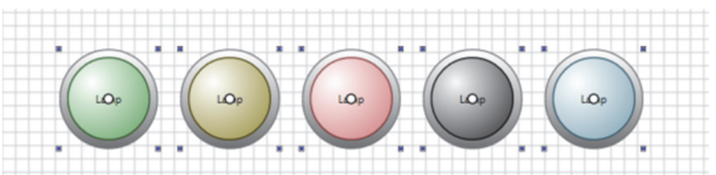


Bild: „Horizontal verteilen“ wurde ausgeführt.

Nach diesen beiden Schritten sind die Lampen so angeordnet wie in der Aufgabenstellung verlangt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

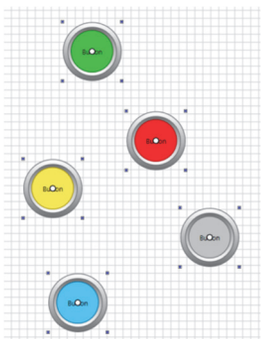
2.5.1.2 Zweites Beispiel zum Ausrichten von Objekten

Eine Anordnung von Schaltern soll vertikal angeordnet werden. Nach den Aktionen sollen die Schalter in einer vertikalen Linie und mit gleichem Abstand platziert sein. Das linke Bild in der Tabelle zeigt die Ausgangssituation.

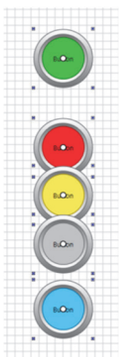
Schritt 1: Aktion „Vertikal mittig ausrichten“ ausführen.

Schritt 2: Aktion „Vertikal verteilen“ ausführen.

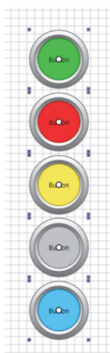
Nach diesen beiden Schritten sind die Schalter so angeordnet wie in der Aufgabenstellung verlangt.



Die Ausgangssituation



„Vertikal mittig ausrichten“ wurde ausgeführt.



„Vertikal verteilen“ wurde ausgeführt.

2.5.2 Objekte an Gitterlinien ausrichten

Ist diese Aktion aktiv (ist der Fall, wenn ein Haken am Symbol der Aktion angezeigt wird), dann wird ein Objekt beim Verschieben mit der Maus jeweils an der nächsten Gitterlinie ausgerichtet. Die Aktion ist unabhängig davon, ob die Gitterlinien angezeigt werden oder nicht.

Ist die Aktion inaktiv, dann ist eine freie Platzierung möglich.

Das Ausrichten passiert nur beim Verschieben mit der Maus; wird das Objekt über die Tastatur verschoben, dann ist immer eine freie Platzierung möglich.

2.6 Die Undo-Funktion

Mit der **Undo-Funktion** können visuelle Veränderungen an einer Anlage rückgängig gemacht werden. Dazu gehören z. B. das Verschieben und Rotieren von Objekten oder auch das Löschen.

Die Undo-Schaltfläche ist oberhalb der Zeichenfläche von PLC-Lab angeordnet.

Bild: Undo-Schaltfläche

Die Zahl im unteren Bereich zeigt an, wie viele Undo-Schritte möglich sind. Beim Speichern einer Anlage werden die Undo-Schritte zurückgesetzt.

2.7 Zeichnen-Gruppen

Zeichnen-Gruppen **erhöhen** den **Komfort** beim Erstellen von Anlagen in PLC-Lab. Sie sind unter anderem für folgende Aufgaben optimal geeignet:

- Vergrößern und verkleinern von mehreren Objekten gleichzeitig, wobei die Relationen zwischen den Elementen der Zeichnen-Gruppe erhalten bleiben.
- Zusammenfassen einer Anordnung von Objekten, um diese gemeinsam zu verschieben und zu kopieren.
- Zusammenfassen einer Anordnung, um die enthaltenen Objekte gemeinsam zu rotieren. Der Rotationspunkt liegt dabei in der Mitte der Zeichnen-Gruppe.

Innerhalb der Aktionsliste von PLC-Lab befindet sich die Rubrik „Zeichnen-Gruppe“. In dieser Rubrik sind Aktionen aufgelistet, welche im Zusammenhang mit Zeichnen-Gruppen benötigt werden. Dazu gehört beispielsweise das Bilden einer Zeichnen-Gruppe, das Auflösen einer Zeichnen-Gruppe und das Herauslösen einzelner Objekte aus einer solchen Gruppe.

2.7.1 Bilden von Zeichnen-Gruppen

Zwei Schritte sind notwendig, um eine Zeichnen-Gruppe zu bilden.

- **Schritt 1:** Selektion der Objekte, welche zu einer Zeichnen-Gruppe zusammengefasst werden sollen.
- **Schritt 2:** Ausführung der Aktion „Zeichnen-Gruppe bilden“ innerhalb der Rubrik „Zeichnen-Gruppe“.

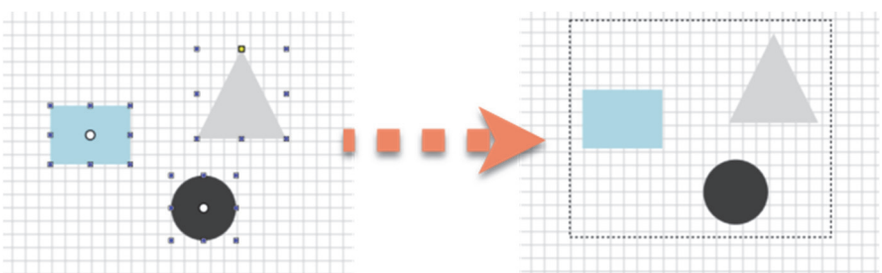


Bild: Bilden einer Zeichnen-Gruppe

2.7.2 Bestandteile einer Zeichnen-Gruppe

Im folgenden Bild sind die Bestandteile einer Zeichnen-Gruppe benannt.

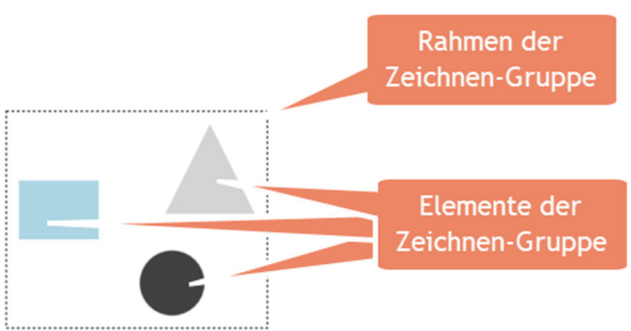


Bild: Elemente einer Zeichnen-Gruppe

Die Elemente einer Zeichnen-Gruppe werden von einem Rahmen umgeben. Dieser umschließt alle Elemente der Gruppe. Wird ein Element der Gruppe verschoben, dann verschiebt sich (falls notwendig) auch die Umrandung der Gruppe.

© MHJ-Software GmbH & Co. KG - PLC-Lab

2.7.3 Selektion einer Zeichnen-Gruppe

Möchte man die Zeichnen-Gruppe selektieren, dann klickt man im nicht-selektierten Zustand der Gruppe auf ein Element der Gruppe. Als Folge wird die Umrandung mit Anfasser versehen.

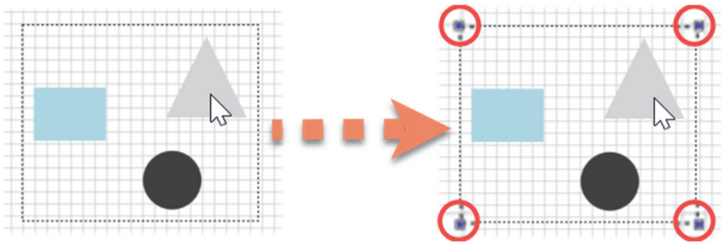


Bild: Gruppe wurde selektiert.

2.7.4 Verschieben einer Zeichnen-Gruppe

Möchte man eine Zeichnen-Gruppe verschieben, dann klickt man bei einer nicht selektierten Gruppe auf ein Element, hält die linke Maustaste betätigt und verschiebt die Maus. Daraufhin wird die **gesamte Gruppe** mit der Maus verschoben.

2.7.5 Selektion eines Objekts in einer Zeichnen-Gruppe

Soll ein Element innerhalb einer Zeichnen-Gruppe selektiert werden, dann ist zunächst die Gruppe zu selektieren und anschließend das Element selbst. Als Resultat sind sowohl die Umrandung der Gruppe als auch das selektierte Objekt mit Anfasser versehen.

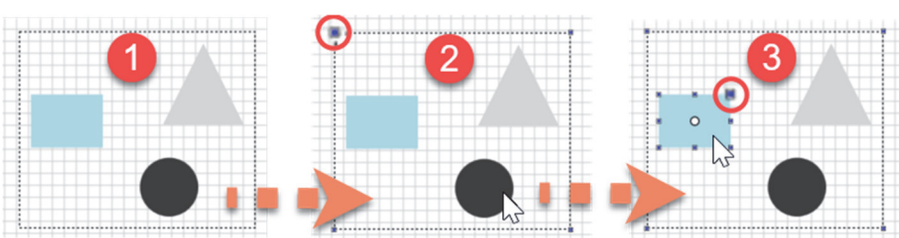


Bild: Schritte zur Selektion eines Elements einer Zeichnen-Gruppe

Sollen weitere Elemente der Zeichnen-Gruppe selektiert werden, dann betätigt man die [Shift]-Taste und klickt die weiteren Elemente mit der Maus an. Diese werden dann ebenfalls mit Anfasser angezeigt.

2.7.6 Verschieben bzw. Rotieren von Objekten in einer Zeichnen-Gruppe

Ist ein Element einer Zeichnen-Gruppe selektiert, dann kann man es verschieben.

Im Bild wird das blaue Rechteck nach links verschoben. Dabei erkennt man, dass sich die Umrandung ebenfalls verändert, damit weiterhin alle Elemente der Gruppe von ihr umschlossen werden. Selektiert man mehrere Objekte innerhalb der Gruppe, dann verschieben sich alle selektierten Objekte.

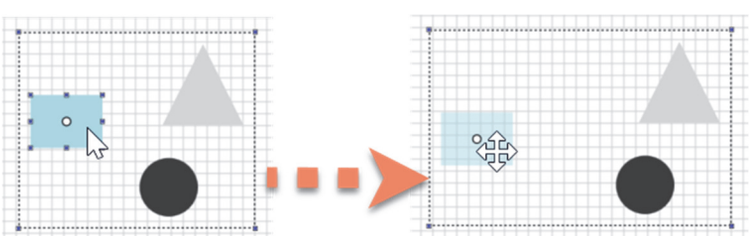


Bild: Verschieben eines Elements einer Zeichnen-Gruppe

Nun soll ein Element einer Zeichnen-Gruppe rotiert werden.

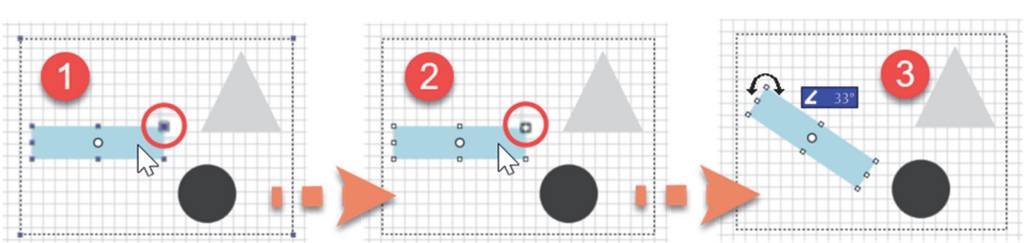


Bild: Rotieren eines Elements einer Zeichnen-Gruppe

Möchte man ein Element einer Zeichnen-Gruppe rotieren, dann ist es zunächst zu selektieren (Punkt 1). Anschließend wird ein Doppelklick auf dem Objekt ausgeführt, damit es sich im Rotationsmodus befindet (Punkt 2). Der Modus wird, wie bereits bekannt, durch die veränderten Anfasser symbolisiert. Nun kann das Objekt wie gewohnt mit der Maus rotiert werden (Punkt 3). Bei gleichzeitiger Betätigung der [Shift]-Taste erfolgt die Rotation jeweils in 15°-Schritten.

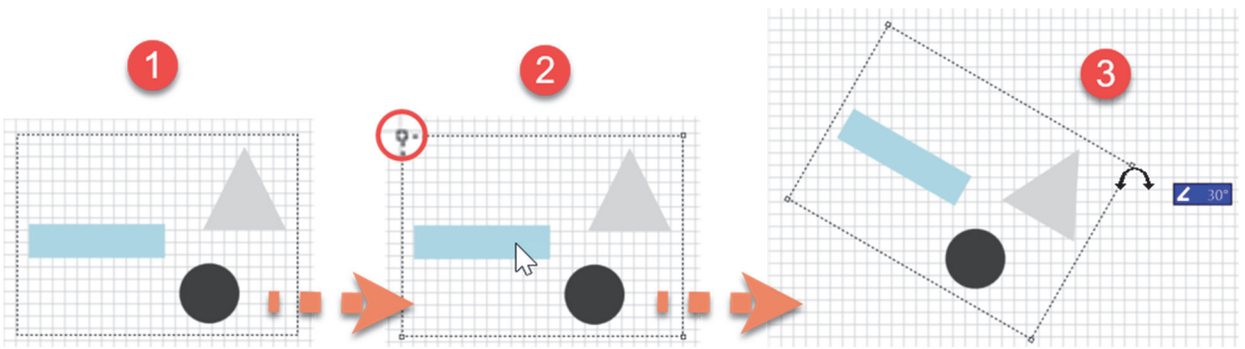
Sollen mehrere Elemente gleichzeitig rotiert werden, dann werden sie zunächst alle selektiert und anschließend auf einem Element ein Doppelklick ausgeführt. Als Folge werden alle selektierten Objekte in den Rotationsmodus versetzt und man kann sie gleichzeitig rotieren.

2.7.7 Rotieren der gesamten Zeichnen-Gruppe

Wie einzelne Elemente einer Zeichnen-Gruppe rotiert werden können, wurde im letzten Abschnitt gezeigt. Nun soll die Rotation der **gesamten Gruppe** erfolgen.

Das Interessante bei dieser Rotation ist, dass der Ursprung der Rotation im Mittelpunkt der Zeichnen-Gruppe liegt.

Bild: Rotieren der gesamten Gruppe



Im nicht-selektierten Zustand (Punkt 1) wird ein Doppelklick auf einem Element der Gruppe ausgeführt. Dadurch wird die gesamte Gruppe in den Rotationsmodus versetzt (Punkt 2). Jetzt kann man die Gruppe rotieren, bei betätigter [Shift]-Taste erfolgt die Rotation in 15°-Schritten.

Klickt man mit der Maus auf eine freie Stelle der Zeichenfläche, dann wird der Rotationsmodus verlassen.

2.7.8 Größenänderung der gesamten Zeichnen-Gruppe

Das Ändern der **Größe** einer Zeichnen-Gruppe ist nur unter **Beibehaltung** der **Seitenverhältnisse** möglich. Bei der Größenänderung wird versucht die Proportionen innerhalb der Gruppe beizubehalten. Dazu gehören auch die Abstände zwischen den Elementen der Gruppe.

Das Verändern der Größe einer Gruppe erfolgt analog zur Änderung der Größe eines einzelnen Objekts. Zunächst wird die Gruppe selektiert, damit die Anfasser angezeigt werden. Danach wird mit der Maus ein Anfasser angeklickt und in der Lage verändert.

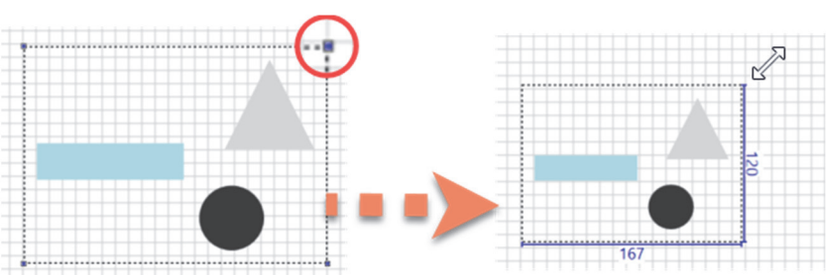


Bild: Größenänderung einer Zeichnen-Gruppe

Im Bild wurde die Größe der Gruppe verringert, man erkennt, dass innerhalb der Gruppe die relative Lage der Elemente **beibehalten** wurde.

© MHJ-Software GmbH & Co. KG - PLC-Lab

2.7.9 Auflösen von gesamten Zeichnen-Gruppen oder Herauslösen einzelner Objekte

Soll ein einzelnes oder mehrere Objekte aus einer Zeichnen-Gruppe herausgelöst werden, dann sind dazu zwei Schritte notwendig:

1. Selektion der herauszulösenden Elemente der Gruppe.
2. Ausführen der Aktion „**Objekt aus Gruppe herauslösen**“ innerhalb der Rubrik „Zeichnen-Gruppe“.

Danach sind die Objekte nicht mehr Teil der Gruppe und werden somit bei der Umrandung nicht mehr berücksichtigt. Im folgenden Bild wurden diese Schritte für die Rechtecke ausgeführt.

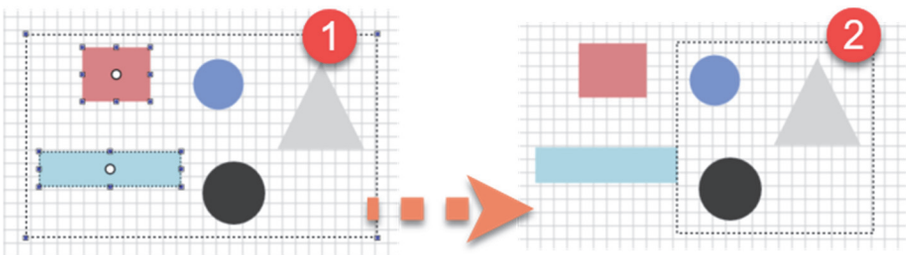


Bild: Herauslösen von Objekten aus einer Gruppe

Bei Punkt 2 ist zu erkennen, dass die Umrandung der Gruppe nur noch die drei verbliebenen Elemente der Gruppe einschließt.

2.7.10 Objekte zu einer bestehenden Zeichnen-Gruppe hinzufügen

Es ist ebenso möglich, ein oder mehrere Objekte in eine bestehende Zeichnen-Gruppe **einzufügen**. Die Vorgehensweise dazu ist recht einfach.

1. Selektion der Gruppe, zu welcher die Objekte hinzugefügt werden sollen.
2. Selektion der Objekte, welche der Gruppe hinzu zufügen sind. Bei Selektion über die Maus ist dabei die [Shift]-Taste zu betätigen.
3. Ausführen der Aktion „Zeichnen-Gruppe bilden“ innerhalb der Rubrik „Zeichnen-Gruppen“.

2.7.11 Einzelne Objekte in einer Zeichnen-Gruppe kopieren

Sollen Objekte, welche bereits Elemente einer Gruppe sind, kopiert werden und sollen die Duplikate ebenfalls Elemente der Gruppe sein, **dann muss das Duplizieren mit Hilfe der Maus und der [STRG]-Taste erfolgen**. Folgende Schritte sind dabei notwendig:

1. Selektion der zu duplizierenden Objekte.
2. Kopieren der Objekte mit Hilfe der Maus bei **betätigter** [STRG]-Taste.

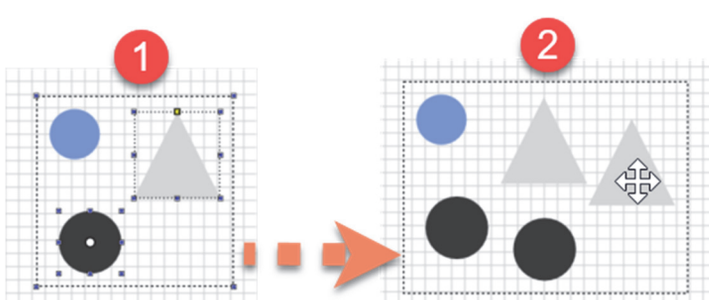


Bild: Elemente der Gruppe werden dupliziert und sind ebenfalls Elemente der Gruppe.

Im Bild sind die Schritte vollzogen worden. Man erkennt in Punkt 2, dass die als Kopie entstandenen Objekte ebenfalls der Gruppe zugehörig sind, denn die Umrandung der Gruppe hat sich beim Verschieben der Objekte ebenfalls angepasst.

Werden Objekte einer Gruppe mit Hilfe von Copy/Paste dupliziert, dann sind die Duplikate **nicht** Elemente der Gruppe.

2.7.12 Gesamte Zeichnen-Gruppe kopieren

Eine gesamte Zeichnen-Gruppe kann ähnlich wie ein einzelnes Objekt kopiert werden. Die Gruppe wird selektiert (es darf kein Element der Gruppe zusätzlich selektiert sein) und der Kopiervorgang wird z. B. mit Hilfe der Maus durchgeführt.

2.7.13 Vorteile einer Zeichnen-Gruppe

Beispiel (ZeichnenGruppenBeispiel):

In einem Beispiel wird demonstriert, wie man sich die Eigenschaften einer Zeichnen-Gruppe zu Nutze machen kann. Im Beispiel soll eine **Fahrstuhltür** simuliert werden. Diese Tür besteht aus zwei Schiebetür-Hälften. Begonnen wird mit dem linken Teil der Tür. Diese besteht aus einem feststehenden Teil und dem beweglichen Teil. Diese beiden Teile sind über eine Verbindung für lineare Bewegung miteinander verbunden.

Hinweis: Da zu diesem Zeitpunkt die verwendeten Objekt-Arten noch nicht bekannt sind, wird das Beispiel vorgeführt.

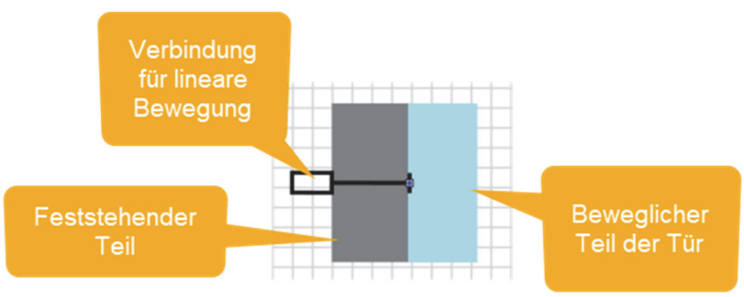


Bild: Linker Teil der Tür

Die Simulation soll ohne SPS-Programm oder GRAFCET funktionieren. Deshalb werden die Operanden, welche die Tür bewegen, direkt über Schalter beeinflusst.

Dazu werden in der Symboltabelle zwei Operanden des Sim-Device angelegt.

Um die Operanden in die Verbindung für lineare Bewegung einzutragen, wird das Symbol der Verbindung selektiert. Danach sind dessen Eigenschaften zu sehen.

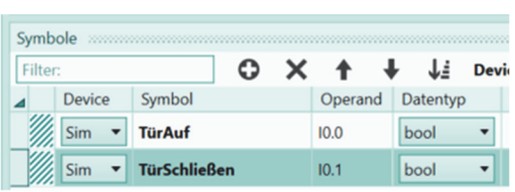


Bild: Operanden und Symbole für das Beispiel

Für das Beispiel sind die beiden Eigenschaften „Operand Bewegung zum Minimum“ und „Operand Bewegung zum Maximum“ zu befüllen, und zwar mit den Symbolen „TürAuf“ sowie „TürSchließen“. Das Eintragen erfolgt per **Drag-and-Drop**, beginnend mit der schraffierten Fläche in der Zeile des Operanden in der **Symbol-Tabelle**:

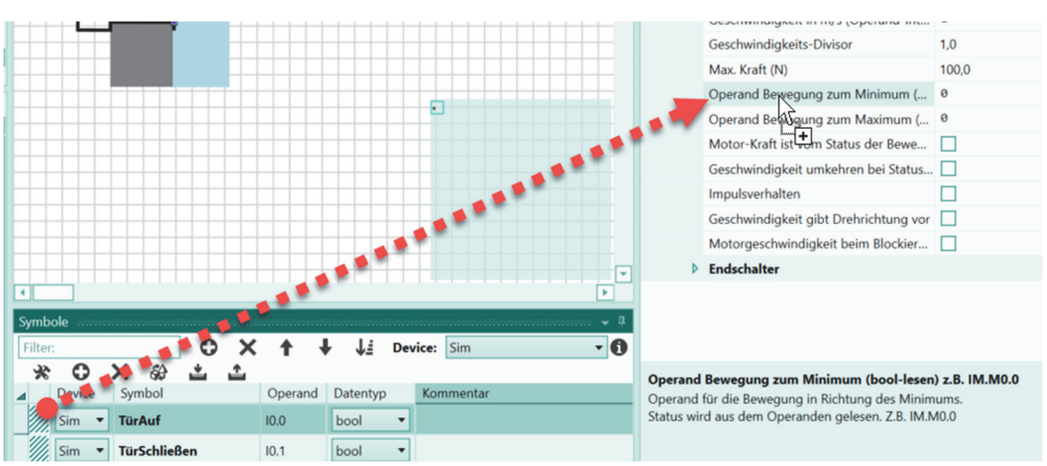


Bild: Operand wird per Drag-and-Drop in die Eigenschaft eingefügt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Das Ziel der Drag-and-Drop-Aktion kann wahlweise die Beschreibung der Eigenschaft oder aber der Editor der Eigenschaft sein.

Auf die gleiche Weise wird das Symbol „TürSchließen“ in die Eigenschaft „**Operand Bewegung zum Maximum**“ eingetragen. Somit sind die beiden Eigenschaften wie folgt gefüllt:

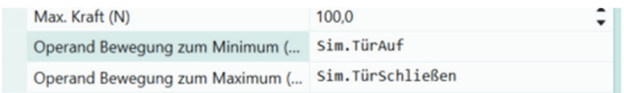


Bild: Eigenschaften mit Operanden

Nun werden die beiden benötigten Schalter-Objekte platziert. Nach der Platzierung sind sie zu selektieren, anschließend wird die Eigenschaft „Button ist Taster“ angewählt.

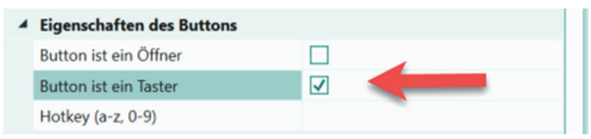


Bild: Eigenschaft „Button ist ein Taster“ wird ausgewählt

Da beide Schalter selektiert sind, ändert sich die Eigenschaft ebenfalls bei beiden.

Jetzt müssen die Schalter mit den Operanden versehen werden. Dies ist bei Schaltern sehr einfach möglich, indem man per Drag-and-Drop den Operanden aus der Symbol-Tabelle auf das Schalter-Objekt zieht. Die Aktion wird zunächst mit dem Symbol „TürAuf“ durchgeführt.

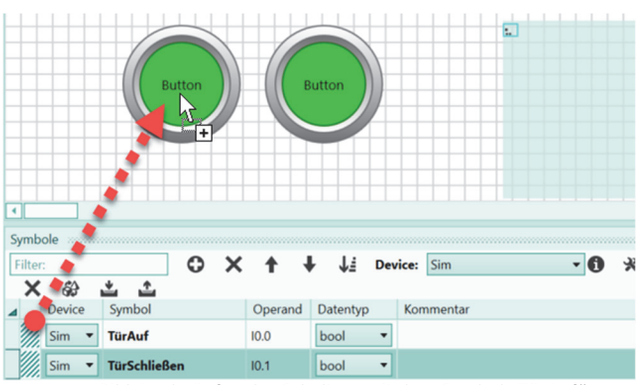


Bild: Verknüpfen des Schalters mit dem Symbol „TürAuf“

Bei diesem Schritt wird zum einen der Operand als zu beeinflussender Operand im Schalter eingetragen und zum anderen das **Symbol** als Beschriftung des Schalters verwendet.

Anschließend führt man die gleiche Aktion mit dem Symbol „TürSchließen“ und dem zweiten Schalter-Objekt aus. Danach haben die beiden Schalter folgendes Aussehen:

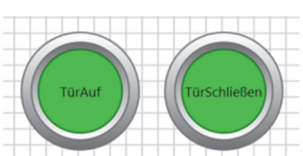


Bild: Schalter-Objekte nach diesen Aktionen

Im nächsten Schritt soll der linke Teil der Tür in einer Zeichnen-Gruppe zusammengefasst werden. Ziel ist es, dass aus diesem auch der rechte Teil der Tür entsteht.

Zu diesem Zwecke werden die beiden Rechtecke selektiert und die Aktion „**Zeichnen-Gruppen bilden**“ ausgeführt.

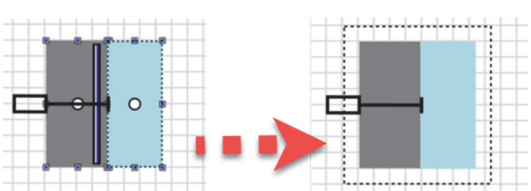


Bild: Bilden der Zeichnen-Gruppe

Somit werden die beiden Rechtecke von einem Rahmen umschlossen als Zeichen dafür, dass sie eine Gruppe bilden. Die Gruppe wird nun dupliziert.

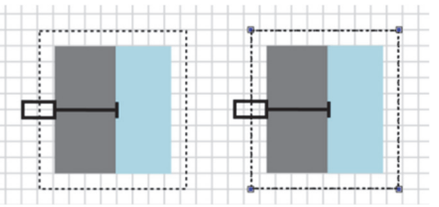


Bild: Die erste Gruppe wird dupliziert.

Die Zeichnen-Gruppe auf der rechten Seite soll für den rechten Teil der Schiebetür verwendet werden. Sie wird jetzt um 180° rotiert. Deshalb wird die Gruppe in den Rotationsmodus versetzt und rotiert.

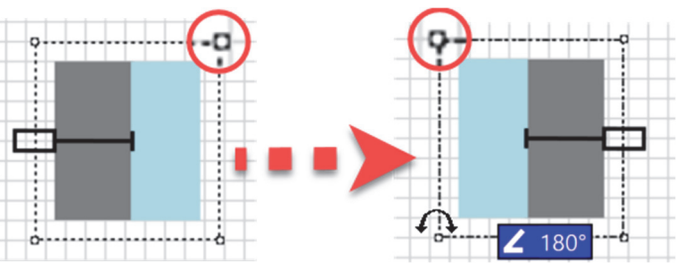


Bild: Duplizierte Zeichnen-Gruppe wird um 180° rotiert.

Damit ist die Anordnung komplett, denn die rechte Tür ist einfach durch ein rotiertes Duplikat der linken Tür entstanden. Die Simulation kann auf Run geschaltet und die Anordnung getestet werden.

Fazit: Das Beispiel hat einen von vielen Anwendungsfällen für Zeichnen-Gruppen gezeigt. Hier wurde der Vorteil ausgenutzt, dass bei der Rotation einer Zeichnen-Gruppe der Rotations-Ursprung in der Mitte der Gruppe liegt.

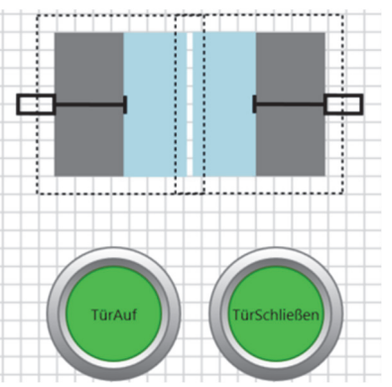


Bild: Fertige Fahrstuhltür

© MHJ-Software GmbH & Co. KG - PLC-Lab

3 Devices in PLC-Lab

Die Objekte von PLC-Lab können mit Operanden verknüpft werden. Je nach Objekt bzw. je nach Eigenschaft wird dann der Status aus dem Operanden **gelesen** oder ein Wert in ihn **geschrieben**.

Die Operanden wiederum kommen aus einem Device, also einem Gerät, das z. B. eine reale SPS (S7-300/400, S7-1200/1500 usw.) sein kann, eine Simulations-SPS (PLCSIM der S7-300, PLCSIM der S7-1200/1500, PLCSIM Advanced, SoftSPS von WinSPS-S7) oder die GRAFCET-Engine.

Innerhalb eines Anlagenprojekts können Operanden von unterschiedlichen Devices verwendet werden. Beim Start der Simulation wird dann die Kommunikation zu den Devices aufgebaut und der Datenaustausch gestartet.

PLC-Lab unterstützt in der **Pro-Edition** folgende Devices:

- PLCSIM S7-1200 ab TIA V13 oder höher
- PLCSIM S7-1500 ab TIA V13 oder höher
- PLCSIM S7-300 ab TIA V13 oder höher
- PLCSIM S7-400 ab TIA V13 oder höher
- PLCSIM S7-300 ab Simatic-Manager V5.5 oder höher
- PLCSIM S7-400 ab Simatic-Manager V5.5 oder höher
- PLCSIM ADVANCED V2.0 SP1 oder höher (ab PLC-Lab V1.5.3)
- CodeSys V3 basierte Steuerungen über OPC UA (ab PLC-Lab V1.6.0)
- GRAFCET-Studio
- WinSPS-S7 V6
- Reale S7-1200 von Siemens über Ethernet/Profinet
- Reale S7-1500 von Siemens über Ethernet/Profinet
- Reale S7-1200/1500 von Siemens über OPC UA (ab PLC-Lab V1.6.0)
- Reale S7-300 von Siemens oder Kompatible über Ethernet/Profinet, NetLink PRO TCP/IP, Simatic-Net
- Reale S7-400 von Siemens über Ethernet/Profinet, NetLink PRO TCP/IP, Simatic-Net
- Reale LOGO! von Siemens (0BA7, 0BA8 oder höher) über Ethernet

In den folgenden Kapiteln werden die einzelnen Devices vorgestellt und die verwendbaren Operanden benannt. Des Weiteren wird auf die Besonderheiten hingewiesen, die für den korrekten Aufbau der Kommunikation zu beachten sind.

Hinweis: In PLC-Lab ist eine Umverdrahten-Funktion vorhanden. Mit dieser können sehr schnell Operanden eines Device auf ein anderes Device umverdrahtet werden, sofern die Operanden im Ziel-Device ebenfalls vorhanden sind. Hat man also beispielsweise eine Anlage für das PLCSIM der S7-1200/1500 erstellt, dann können die Operanden auf das PLCSIM der S7-300 umverdrahtet werden. Die Anlage muss dafür nicht neu erstellt werden.

3.1 IM-Device

Das **IM-Device** hat eine **Sonderstellung** in PLC-Lab. Es handelt sich dabei um den internen Speicher (Intern Memory) von PLC-Lab. Dieser Speicher ist immer vorhanden und hat eine Größe von 64 KByte. Der Zugriff auf den Speicher erfolgt über den Operandentyp „M“ (Merker bzw. Flag). Die Merker können mit Hilfe von Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Der Speicher ist Byte-orientiert aufgebaut. Auf die Merker-Operanden sind sowohl lesende als auch schreibende Zugriffe erlaubt.

3.1.1 Zugriff auf Merker-Operanden

Operand	Beispiel	Kommentar
Bit-Operand	IM.M21.4	An der Byte-Adresse 21 wird das Bit 4 adressiert.
Byte-Operand	IM.MB20	Adressierung des Bytes mit der Adresse 20
Wort-Operand	IM.MW20	Adressierung des Worts mit der Adresse 20. Dieses besteht aus den Bytes MB20 und MB21. Das MB20 ist das HiByte.
Doppelwort-Operand	IM.MD20	Adressierung des Doppelworts 20. Dieses besteht aus den beiden Wörtern MW20 und MW22 und somit den Bytes MB20 bis MB23. Das MB20 ist das Hi-Byte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über MB20). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.1.2 Wann werden Operanden des IM-Device verwendet?

Die Merker des IM-Device kommen zur Anwendung, wenn in einem Projekt von PLC-Lab Zwischenergebnisse abzulegen sind. Des Weiteren können Daten in diesem Speicher abgelegt werden, welche nur zur Simulation der Anlage notwendig sind. Beispiele sind Umrechnungen von analogen Werten, Erfassen von bestimmten Anlagenzuständen zur verbesserten Simulation usw.

3.1.3 Das Taktmerkerbyte IM.MB65534

In vielen Anlagen werden Blink-Takte benötigt, um Zustände zu triggern oder auch Anzeigeelemente, z. B. Lampen, blinkend darzustellen.

Aus diesem Grund wurde das Merker-Byte 65534 im IM-Device als sog. Taktmerkerbyte ausgelegt. Dies bedeutet, dass im Run-Zustand von PLC-Lab jedes einzelne Bit des Bytes in einer bestimmten Frequenz toggelt. In der folgenden Tabelle sind die Frequenzen angegeben.

Bit	0	1	2	3	4	5	6	7
Frequenz in Hz	10	5	2.5	2	1.25	1	0.625	0.5

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.2 Das Device PLCSim

Im TIA-Portal von Siemens ist das **PLCSIM** für die SPS-Familien **S7-1200** und **S7-1500** vorhanden. Um dieses PLCSIM anzusprechen, sind in PLC-Lab Operanden für das Device mit der Bezeichnung „PLCSim“ anzulegen. Dazu muss das TIA-Portal die Version 13 SP1 oder höher besitzen.

Beim Device PLCSim können Eingänge beschrieben und Ausgänge gelesen werden.

Damit PLC-Lab mit dem PLCSIM des TIAs zusammenarbeiten kann, müssen die beiden Programme über die gleichen Rechte verfügen. Wird also das TIA mit Admin-Rechten gestartet, dann ist auch PLC-Lab als Admin zu starten; notwendig ist dies allerdings nicht.

3.2.1 Erläuterungen zum Device PLCSim

Das Device PLCSim ist in der **Pro-Edition** von PLC-Lab vorhanden. Ein explizites Erzeugen ist nicht notwendig.

Um Operanden des Device in der Symboltabelle anzulegen, muss das Device als aktives Device selektiert werden.

Bild: Auswahl des Device „PLCSim“

Nun kann man mit Hilfe des linken Plus-Symbols Operanden einfügen. Es ist dabei **unerheblich**, ob die englische oder deutsche Syntax für die Operanden verwendet wird. Im folgenden Beispiel wurde der Operand „I0.0“ mit dem Symbol „StartTaster“ eingefügt.

Bild: Einfügen eines Operanden für das Device „PLCSim“

Im Bild wird zunächst die Schaltfläche für das Einfügen eines Operanden betätigt (Punkt 1). Dann kann in der Zeile des Operanden das Symbol, der Operand, der Datentyp und der Kommentar angegeben werden.

Möchte man anstatt der englischen Syntax, die deutsche verwenden, dann ändert man einmalig den Operanden von I0.0 zu E0.0. Nachfolgend eingefügte Operanden werden dann in der deutschen Syntax eingefügt.

Tipp: Über die Plus-Schaltfläche (Punkt 1) lassen sich so **sehr schnell fortlaufende Symbole** wie S1, S2, S3, S4 usw. erstellen: „S1“ als Symbol einfügen, dann die Zeile mit „S1“ selektieren und wiederholt die Plus-Schaltfläche drücken.

3.2.1.1 Voreinstellungen für das PLCSim-Device

Ist das Device **PLCSim** als aktives Device selektiert, dann kann über die Schaltfläche „Einstellungen des Device verändern“ der Einstellungsdialog aufgerufen werden.

Bild: Einstellungen von PLCSim aufrufen

Als Folge wird der Dialog „Einstellungen PLCSim S7-1200/1500“ angezeigt.

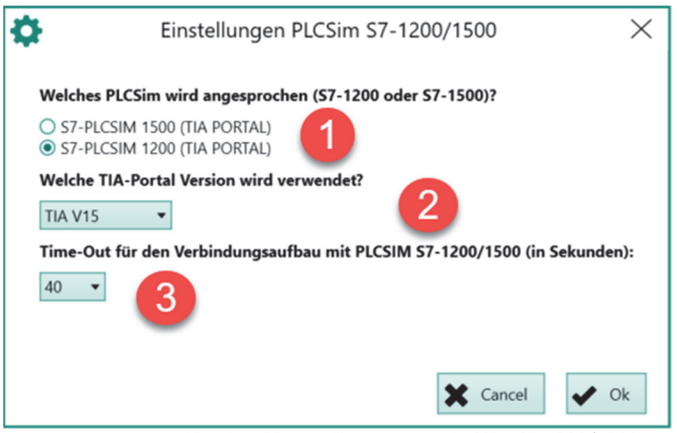


Bild: Der Dialog „Einstellungen PLCSim S7-1200/1500“

Anbei die Erläuterung der Dialogelemente:

1. Hier ist einzustellen, welche SPS-Familie vom PLCSIM des TIAs simuliert wird. Also entweder die S7-1200 oder die S7-1500.
2. Leider unterscheiden sich die TIA-Versionen **V13** bis **V16** recht stark, was man auch immer dann bemerkt, wenn man ein Projekt einer älteren Version mit einer neueren öffnen möchte. Dies ist auch beim Zugriff auf PLCSim der Fall. Hier ist ebenfalls die TIA-Version anzugeben, mit der man momentan arbeitet. Die Einstellung „Universal“ ist zu wählen, wenn der Kommunikationsaufbau mit der genauen Version fehlgeschlagen ist.
3. Insbesondere der erste Verbindungsaufbau zu PLCSim kann einige Sekunden dauern. Hier kann das max. Timeout (in Sekunden) für den Verbindungsaufbau angegeben werden.

3.2.1.2 Systemvoraussetzungen des TIA-Portals

Die **Hardwarevoraussetzungen** für das TIA-Portal sind recht hoch. Siemens gibt folgende Eckdaten vor:

- Core i5-6440EQ, 3,4 GHz
- RAM 16 GB
- SSD mit mindestens 50GB freiem Speicherplatz

Diese gelten nur für das TIA-Portal. Wird zusätzlich das PLCSIM gestartet, dann vereinnahmt es zusätzlich einen gesamten Prozessorkern.

Die stromsparenden Prozessoren von Notebooks (z. B. die U- und Y-Prozessoren der Core-i-Serie) sind für den Betrieb des TIA-Portals zu langsam. Bei Notebooks sollten z. B. die H-Prozessoren der Core-i-Serie zum Einsatz kommen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.2.2 Die Operanden von PLCSim

Wie schon erwähnt, werden **Eingänge** und **Ausgänge** vom Device PLCSim unterstützt. Dabei sollten die Eingänge in einer Anlage von PLC-Lab beschrieben und die Ausgänge gelesen werden.

3.2.2.1 Unterstützte Operanden-Bytes bei Verwendung von PLCSIM S7-1500

Bei PLCSIM S7-1500 werden die Eingangsbytes **EB0 bis EB462** unterstützt. Als Ausgangsbytes können die Bytes **AB0 bis AB510** verwendet werden.

3.2.2.2 Verbotene Operanden-Bytes bei Verwendung von PLCSIM S7-1500

Folgende Operanden dürfen nicht im SPS-Programm verwendet werden, wenn das PLCSIM S7-1500 zum Einsatz kommt:

- Alle Eingangsbyte **ab EB463** und höher.
- Alle Ausgangsbytes **ab AB511** und höher.

3.2.2.3 Unterstützte Operanden-Bytes bei Verwendung von PLCSIM S7-1200

Bei PLCSIM S7-1200 werden die Eingangsbytes **EB0 bis EB510** unterstützt. Als Ausgangsbytes können die Bytes **AB0 bis AB510** verwendet werden.

3.2.2.4 Verbotene Operanden-Bytes bei Verwendung von PLCSIM S7-1200

Folgende Operanden dürfen nicht im SPS-Programm verwendet werden, wenn das PLCSIM S7-1200 zum Einsatz kommt:

- Alle Eingangsbyte **ab EB511** und höher.
- Alle Ausgangsbytes **ab AB511** und höher.

3.2.2.5 Adressierung der Eingänge und Ausgänge

Die Eingänge und Ausgänge können über Bit-, Byte-, Wort- oder Doppelwort-Operanden angesprochen werden. Ob die englische oder deutsche Syntax verwendet wird ist unerheblich.

Operand	**Beispiel**	**Kommentar**
Bit-Operand	PLCSim.A5.4	An der Ausgangs-Byteadresse 5 wird das Bit 4 adressiert.
Byte-Operand	PLCSim.EB10	Adressierung des Eingangs-Bytes mit der Adresse 10
Wort-Operand	PLCSim.AW12	Adressierung des Ausgangs-Worts mit der Adresse 12. Dieses besteht aus den Bytes AB12 und AB13. Das AB12 ist das HiByte.
Doppelwort-Operand	PLCSim.ED30	Adressierung des Eingangs-Doppelworts 30. Dieses besteht aus den beiden Wörtern EW30 und EW32 und somit den Bytes EB30 bis EB33. Das EB30 ist das HiByte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über EB10). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

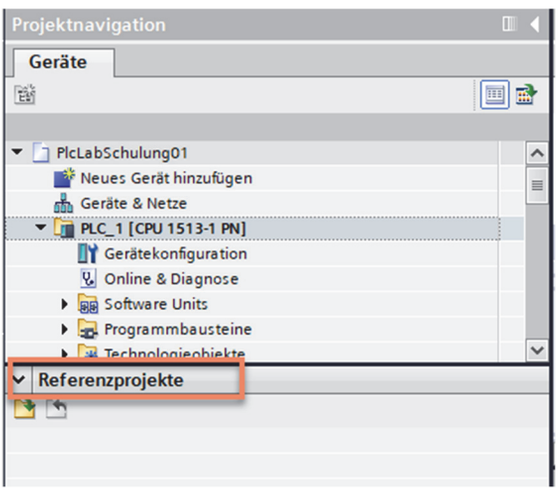
3.2.3 Die Vorlageprojekte für das TIA-Portal

Damit der Datenaustausch zwischen PLC-Lab und dem PLCSIM des TIA-Portals funktioniert, sind Konnektierungsbausteine im Netzwerk 1 des OB1 aufzurufen.

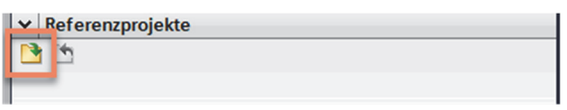
Diese Bausteine werden mit PLC-Lab installiert und befinden sich im Verzeichnis „…\Automatisierung\PLC-Lab-Editor\“ innerhalb der eigenen Dateien.

Damit die Bausteine im TIA-Portal von Siemens projektübergreifend zur Verfügung stehen, sollten die Projekte in der Task-Card „Referenzprojekte“ geöffnet werden. Dabei geht man wie folgt vor:

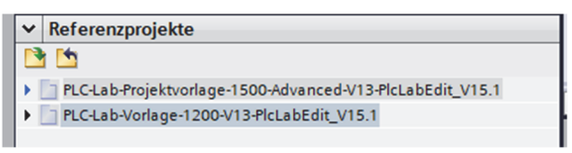
1. Öffnen eines beliebigen Projekts im TIA-Portal.
2. Ausführen des Menüpunkts „Ansicht->Referenzprojekte“. Daraufhin ist die Task-Card „Referenzprojekte“ zu sehen.



3. Innerhalb der Referenzprojekte wird der Button „Referenzprojekt öffnen“ betätigt.



4. Nun werden nacheinander die Vorlageprojekte für die S7-1500 und S7-1200 aus dem oben genannten Verzeichnis geöffnet. Diese sind dann innerhalb der Referenzprojekte offen und die Konnektierungsbausteine können jederzeit in das momentane Projekt eingefügt werden.



Diese stehen auch nach dem erneuten Öffnen des TIA-Portals wieder zur Verfügung.

3.2.4 Vorgehensweise, wenn ein vorhandenes oder neues S7-1200/1500-Projekt simuliert werden soll

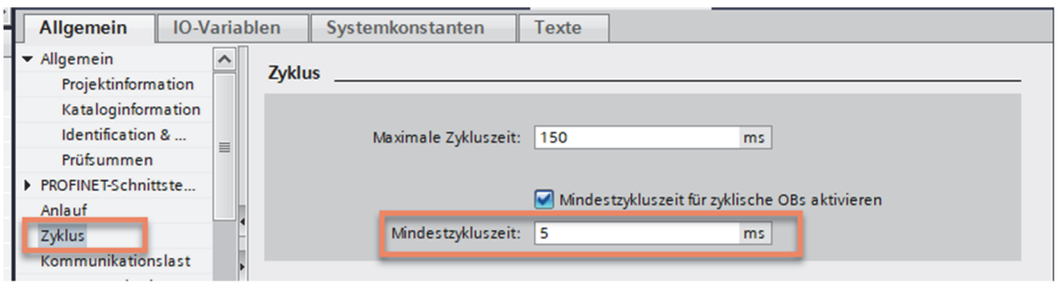
Mit dem Öffnen der Vorlageprojekte innerhalb der Referenzprojekte des TIA-Portals wurde die Voraussetzung für einen schnellen Zugriff auf den jeweils benötigten Konnektierungsbaustein geschaffen.

Nun soll gezeigt werden, wie ein TIA-Portal Projekt vorzubereiten ist, damit das SPS-Programm mit Hilfe von PLC-Lab und PLCSIM der S7-1200/1500 simuliert werden kann. Die Schritte sind auch bei einem bereits vorhandenen TIA-Projekt durchführbar.

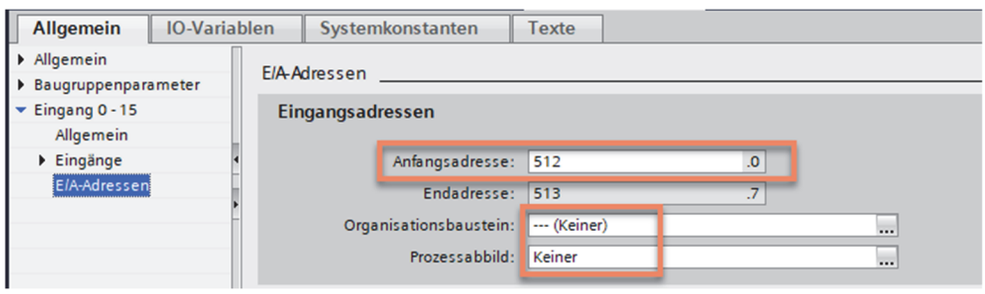
© MHJ-Software GmbH & Co. KG - PLC-Lab

3.2.4.1 Vorgehensweise bei einer S7-1500 und der Simulation mit PLCSIM S7-1500

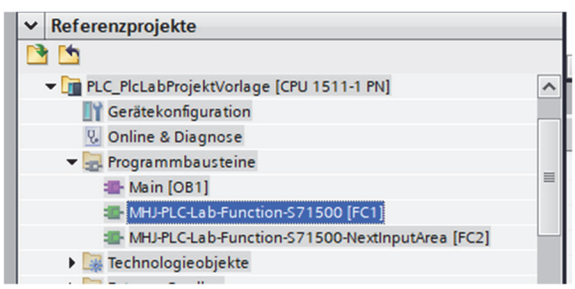
1. Öffnen oder Erzeugen des TIA-Projekts mit einer S7-1500er CPU und Auswahl der Projektansicht. Anschließend wird ein Doppelklick auf der Gerätekonfiguration innerhalb der vorhandenen SPS-Station ausgeführt. Somit ist das Rack der CPU zu sehen.
2. Einstellen der Mindestzykluszeit der CPU auf 5 ms. Diese Einstellung ist nicht wegen PLC-Lab vorzunehmen, sondern damit das PLCSIM die PC-CPU nicht noch mehr belastet.



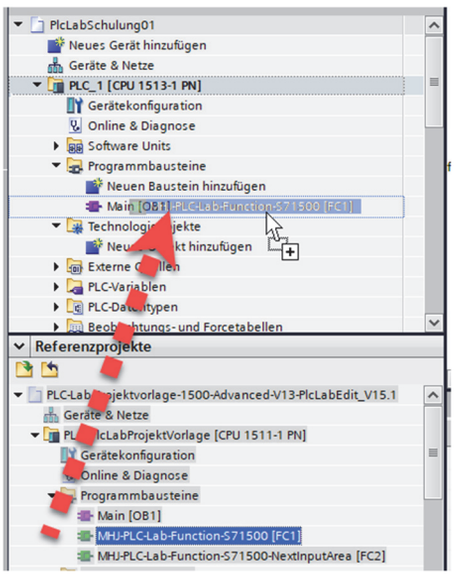
3. Nun wird eine DI16-Baugruppe auf einem freien Steckplatz eingefügt und dieser Baugruppe die Start-Adresse 512 zugeteilt. Anschließend ist die Aktualisierung des Prozessabbildes abzuschalten.



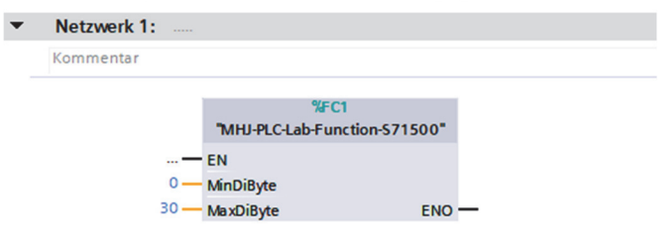
4. Im nächsten Schritt ist in der Task Card „Referenzprojekte“ das PLC-Lab S7-1500 Advanced Vorlageprojekt aus dem Verzeichnis „…\Automatisierung\PLC-Lab-Editor\“ innerhalb der eigenen Dateien zu öffnen.



5. Aus dem PLC-Lab Vorlageprojekt für die S7-1500 ist die Funktion „MHJ-PLC-Lab-Function-S71500“ in das Projekt zu kopieren. Dazu wird im Referenzprojekt innerhalb der Programmbausteine die Funktion per Drag-and-Drop in die Programmbausteine des vorhandenen TIA-Projekts kopiert.



6. Die so eingefügte Funktion wird im Netzwerk 1 des OB1 aufgerufen. Dabei ist an den Parametern „MinDiByte“ und „MaxDiByte“ die minimale und maximale Eingangsbyteadresse anzugeben, welche von PLC-Lab aus beeinflusst werden soll. Im Beispiel wurde EB0 bis EB30 angegeben.



Wichtig ist, dass die hier angegebenen Eingänge als Baugruppe in der Hardwarekonfiguration vorhanden sind und das Aktualisieren (Refreshen) des Prozessabbilds dieser Baugruppen abgeschaltet ist!

Damit kann das TIA-Projekt mit PLCSIM der S7-1500 und der Anlage von PLC-Lab simuliert werden.

3.2.4.1.1 Angabe von mehreren Eingangs-Adressbereichen bei einer S7-1500

Bei einigen Anlagen, insbesondere beim Einsatz von dezentraler Peripherie, werden die Eingangsadressbereiche gestückelt. So werden beispielsweise die Adressen EB0 bis EB40, EB100 bis EB130 und EB200 bis EB250 verwendet.

In einem solchen Fall kommt neben der Funktion „MHJ-PLC-Lab-Function-S71500“ auch die Funktion „MHJ-PLC-Lab-Function-S71500-NextInputArea“ zur Anwendung. Die zweite Funktion befindet sich ebenfalls im Vorlageprojekt und kann, wie oben gezeigt, per Drag-and-Drop aus dem Referenzprojekt in das momentane Projekt kopiert werden. Der Aufruf für die oben genannten Adressbereiche würde dann wie folgt aussehen.

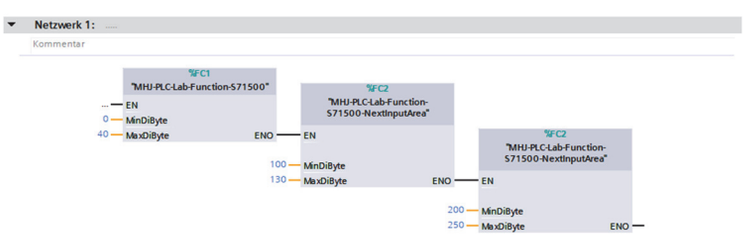


Bild: Aufruf der Funktionen bei gestückelten Eingangsbereichen

© MHJ-Software GmbH & Co. KG - PLC-Lab

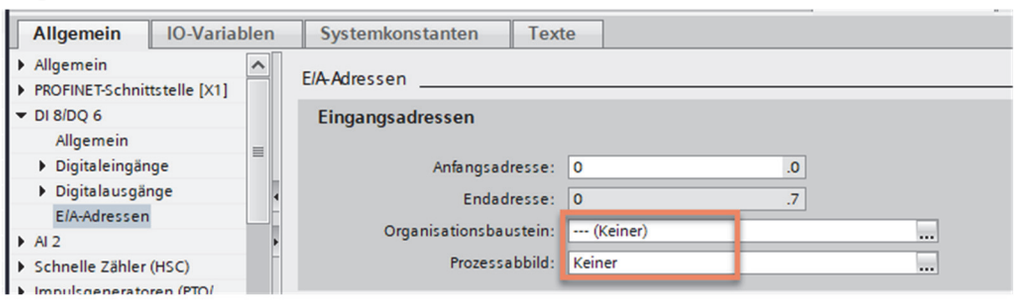
3.2.4.1.2 Besonderheiten bei dezentralen Eingangsmodulen

Im Gegensatz zu den Eingangsmodulen der **zentralen** Peripherie muss das Aktualisieren des Prozessabbilds der Eingangsmodule, die **dezentral** (also z. B. auf Profinet-IO-Devices) vorhanden sind, nicht abgeschaltet werden. Dies zumindest bei PLCSim der TIA-Versionen V13 bis V16.

3.2.4.2 Vorgehensweise bei einer S7-1200 und der Simulation mit PLCSIM S7-1200

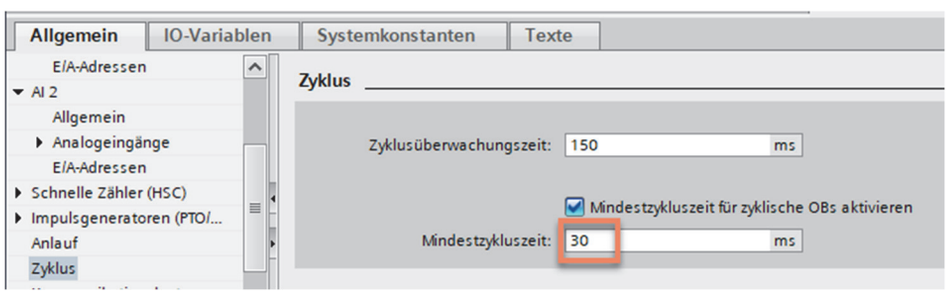
Jetzt die Schritte, um ein vorhandenes oder neues Projekt einer S7-1200 mit PLC-Lab und PLCSIM der S7-1200 zu simulieren:

1. Zunächst ist ein vorhandenes Projekt zu öffnen oder ein neues Projekt zu erzeugen. Anschließend wird zur Projektansicht gewechselt und ein Doppelklick auf der Gerätekonfiguration innerhalb der SPS-Station ausgeführt. Jetzt ist das Rack der CPU zu sehen.
2. Nun wird die CPU selektiert und die digitalen Eingänge ausgewählt. Bei diesen ist für das **Prozessabbild** die Einstellung „**Keiner**“ auszuwählen.

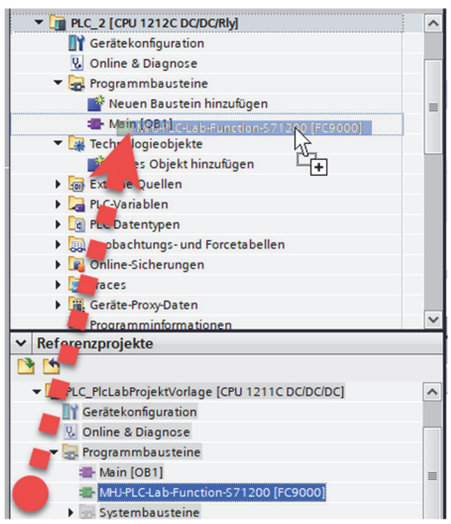


Diese Einstellung ist auch bei den analogen Eingängen vorzunehmen. Sollte die CPU über zusätzliche Eingangsmodule verfügen, dann ist diese Einstellung auch bei ihnen zu tätigen.

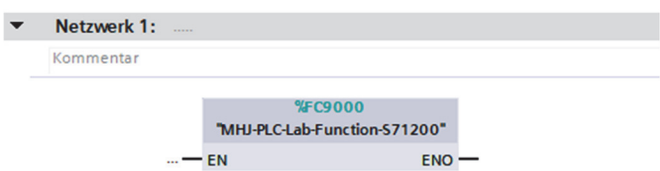
3. Im nächsten Schritt wird die **Mindestzykluszeit** der CPU auf den Wert 30 ms gesetzt. Diese Einstellung muss nicht für PLC-Lab vorgenommen werden. Diese minimiert die PC-CPU-Last durch PLCSIM.



4. Aus dem PLC-Lab Vorlageprojekt für die S7-1200 ist die Funktion „MHJ-PLC-Lab-Function-S71200“ in das Projekt zu kopieren. Dazu wird im Referenzprojekt innerhalb der Programmbausteine die Funktion per Drag-and-Drop in die Programmbausteine des vorhandenen TIA-Projekts kopiert.



5. Die Funktion wird dann im Netzwerk 1 des **OB1** aufgerufen.



6. Jetzt kann das SPS-Programm in PLCSIM der S7-1200 simuliert und an der Anlage von PLC-Lab getestet werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.3 Das Device PLCSim300

Das Device **PLCSim300** ist zu verwenden, wenn das SPS-Programm einer **S7-300/400** im PLCSIM für die S7-300/400 des TIA-Portals ab V13 oder dem Simatic-Manager ab V5.5 bearbeitet wird und die PLC-Lab Anlage zum Test verwendet wird.

3.3.1 Die Operanden von PLCSim300

Das Device **PLCSim300** unterstützt **Eingänge, Ausgänge, Merker und Daten aus Datenbausteinen**. Dabei sollten Eingänge innerhalb der PLC-Lab Anlage beschrieben werden. Ausgänge werden im Normalfall gelesen. Bei Merkern und Daten aus Datenbausteinen gibt es keine klassische Zugriffsart, diese können sowohl gelesen als auch beschrieben werden.

3.3.1.1 Verwendung von Eingängen, Ausgängen und Merkern

Die Operanden-Bereiche Eingänge, Ausgänge und Merker können als Bit-, Byte-, Wort- oder Doppelwort-Operanden angesprochen werden.

Operand	Beispiel	Kommentar
Bit-Operand	PLCSim300.M12.1	An der Merker-Byteadresse 12 wird das Bit 1 adressiert.
Byte-Operand	PLCSim300.AB14	Adressierung des Ausgangs-Bytes mit der Adresse 14
Wort-Operand	PLCSim300.EW16	Adressierung des Eingangs-Worts mit der Adresse 16. Dieses besteht aus den Bytes EB16 und EB17. Das EB16 ist das HiByte.
Doppelwort-Operand	PLCSim.ED34	Adressierung des Eingangs-Doppelworts 34. Dieses besteht aus den beiden Wörtern EW34 und EW36 und somit den Bytes EB34 bis EB37. Das EB34 ist das HiByte.

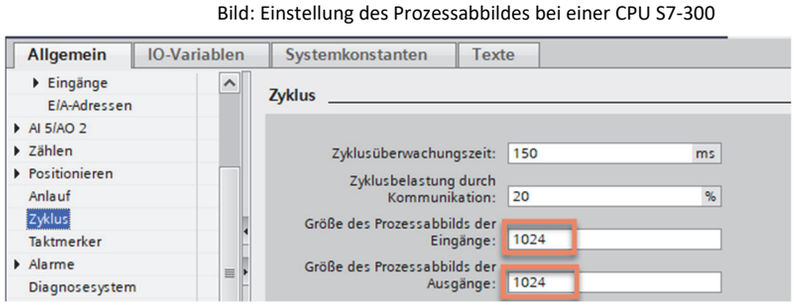
Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über AB14). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.3.2 Besonderheit: E/A-Operanden müssen innerhalb des Prozessabbildes liegen

Bei der Verwendung von Eingängen und Ausgängen mit dem Device PLCSim300 ist darauf zu achten, dass sie **sich innerhalb der Prozessabbildes der CPU** befinden. Dies gilt auch für die analogen Eingänge und Ausgänge.

Wird beispielsweise eine CPU 314PN/DP verwendet, dann liegen deren Standard-Adressen für die Onboard-Peripherie auf der Adresse 800 für die analogen E/As. Hier muss die Größe des PAE/PAA in der Hardwarekonfiguration der CPU auf **1024 Bytes** eingestellt werden. Diese Einstellung findet man im Register „Zyklus“ der Hardwarekonfiguration der CPU.

Bild: Einstellung des Prozessabbildes bei einer CPU S7-300



3.3.3 Verwendung von Daten aus Datenbausteinen

Auch die Daten von Datenbausteinen können über Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Voraussetzung ist, dass sich der angesprochene Datenbaustein in der CPU (also in PLCSIM der S7-300/400) befindet und dass er über die angesprochenen Daten verfügt.

Operand	Beispiel	Kommentar
Bit-Operand	PLCSim300.DB10.DBX23.5	Im DB10 wird an der Byte-Adresse 23 das Bit 5 angesprochen.
Byte-Operand	PLCSim300.DB1.DBB21	Im DB1 wird das Byte mit der Adresse 21 angesprochen.
Wort-Operand	PLCSim300.DB120.DBW2	Adressierung des Daten-Worts mit der Adresse 2. Dieses besteht aus den Bytes DBB2 und DBB3. Das DBB2 ist das HiByte. Das Daten-Wort befindet sich im Datenbaustein 120.
Doppelwort-Operand	PLCSim300.DB12.DBD40	Adressierung des Daten-Doppelworts 40. Dieses besteht aus den beiden Wörtern DBW40 und DBW42 und somit den Bytes DBB40 bis DBB43. Das DBB40 ist das HiByte. Das Datendoppelwort befindet sich im Datenbaustein 12.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über DB10.DBB4). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.4 Das Device PLCSIM Advanced

Das **PLCSIM Advanced** muss separat zum TIA-Portal erworben werden. Es erweitert die Möglichkeiten der Simulation gegenüber dem im TIA-Portal integrierten PLCSIM für die S7-1500.

PLCSIM Advanced kann **nur** für die Systemfamilie **S7-1500** verwendet werden.

Einige der **Vorteile** von PLCSIM Advanced sind:

- Es können mehrere Instanzen auf einem PC gestartet und auch die Kommunikation zwischen den Instanzen simuliert werden.
- Eine Instanz ist von außen erreichbar, was bedeutet, es kann z. B. ein reales HMI oder eine Visualisierung auf die Instanz zugreifen (über Ethernet).
- Der Zugriff über OPC UA kann simuliert werden.

PLC-Lab unterstützt den Zugriff auf PLCSIM Advanced; dabei können auch mehrere Instanzen von PLCSIM Advanced innerhalb einer PLC-Lab Anlage angesprochen werden. Die Identifikation der einzelnen Instanzen geschieht über den Instanz-Namen, welcher im **Control-Panel** von PLCSIM Advanced angegeben ist.

Im folgenden Bild ist das Control-Panel zu sehen; dabei sind die beiden Instanzen mit den Bezeichnungen „Instanz1“ und „Instanz2“ gestartet.

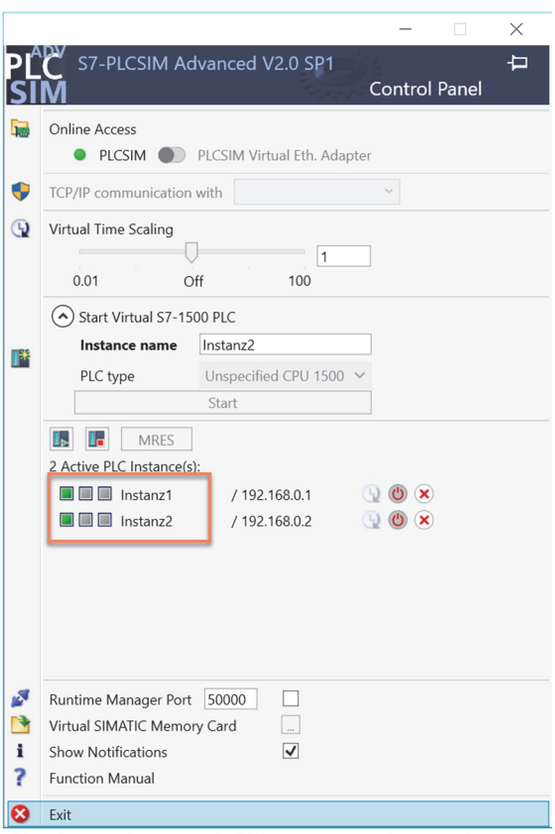


Bild: Control-Panel von PLCSIM Advanced mit den beiden Instanzen

3.4.1 Ansprechen der Operanden von PLCSIM Advanced mit PLC-Lab

Wie schon erwähnt, wird eine Instanz von PLCSIM Advanced in PLC-Lab über den Instanz-Namen angesprochen. Dabei geht man wie folgt vor.

Im PLC-Lab Projekt wird innerhalb der Symboltabelle ein neues Device erzeugt.

Bild: Erzeugen eines neuen Device in PLC-Lab

Daraufhin erscheint der Dialog „**Neues Gerät bzw. Verbindung erzeugen**“. Auf diesem ist als Device Typ „**PLCSIM Advanced**“ zu selektieren. Der Name für das Device ist frei wählbar, im Beispiel wird „PlcSimAdv01“ angegeben.

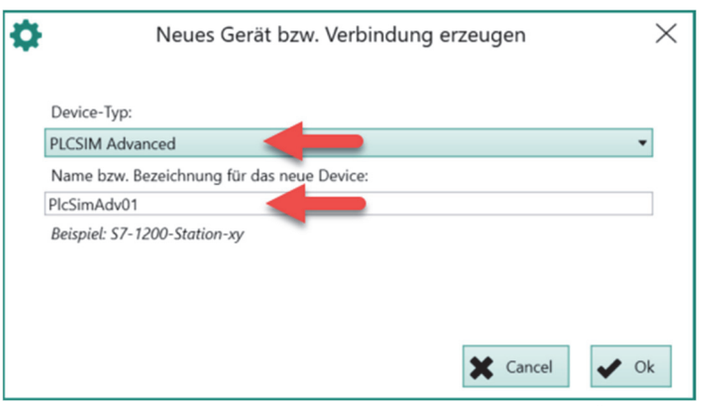


Bild: Dialog „Neues Gerät bzw. Verbindung erzeugen“

Anschließend kann man den Dialog über „OK“ bestätigen. Als Folge ist der Dialog „Einstellungen PLCSIM Advanced“ zu sehen. Auf diesem ist lediglich die Bezeichnung für die anzusprechende Instanz anzugeben.

Die Bezeichnung der Instanz muss korrekt sein, da andernfalls keine Kommunikation mit PLCSIM Advanced zustande kommen wird.

Im Beispiel hat die Instanz die Bezeichnung „**Instanz1**“.

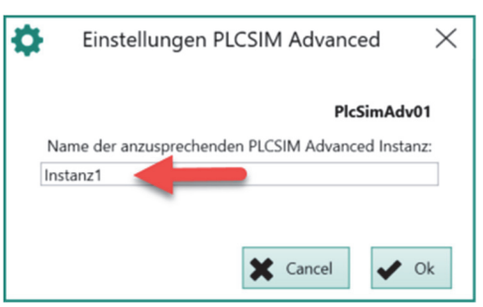


Bild: Angabe des Instanz-Namens von PLCSIM Advanced

Nach Bestätigung wird das neue Device erzeugt und in der Symboltabelle aktiviert.

Bild: Das neue Device ist aktiv

© MHJ-Software GmbH & Co. KG - PLC-Lab

Somit können Operanden für das Device angelegt werden.

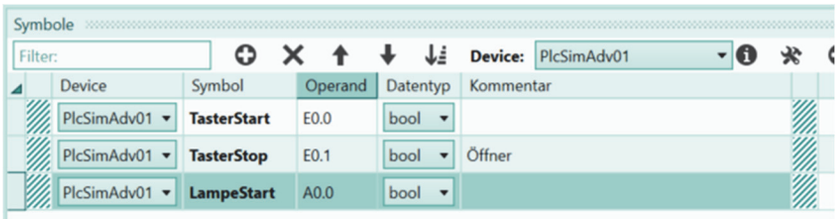


Bild: Symboltabelle mit angelegten Operanden des Device „PlcSimAdv01“

Wird nun die Simulation gestartet, dann versucht PLC-Lab den Kommunikationsaufbau zur „Instanz1“ von PLCSIM Advanced. Gelingt dies, dann werden die Daten ausgetauscht, d. h. die oben angegebenen **Eingänge beschrieben und der Ausgang gelesen** – sofern sie in Objekten der PLC-Lab-Anlage in Verwendung sind.

Die in PLC-Lab verwendeten Eingänge und Ausgänge müssen als Module in der Hardwarekonfiguration der CPU der PLCSIM Advanced-Instanz vorhanden sein. Ansonsten werden sie von PLCSIM Advanced nicht korrekt mit dem Prozessabbild synchronisiert.

3.4.2 Die Operanden von PLCSIM Advanced

Das PLCSIM Advanced Device unterstützt die Operanden Eingänge, Ausgänge und Merker. Normalerweise werden die Eingänge in PLC-Lab beschrieben und die Ausgänge gelesen. Merker können in der Anlage von PLC-Lab sowohl beschrieben als auch gelesen werden.

Alle drei Operanden-Arten sind als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressierbar. In der nachfolgenden Tabelle sind Beispiele zu sehen, dabei hat das PLCSIM Advanced Device die Bezeichnung „**PlcSimAdv**“.

Operand	Beispiel	Kommentar
Bit-Operand	PlcSimAdv.M22.3	An der Merker-Byteadresse 22 wird das Bit 3 adressiert.
Byte-Operand	PlcSimAdv.AB54	Adressierung des Ausgangs-Bytes mit der Adresse 54
Wort-Operand	PlcSimAdv.EW160	Adressierung des Eingangs-Worts mit der Adresse 160. Dieses besteht aus den Bytes EB160 und EB161. Das EB160 ist das HiByte.
Doppelwort-Operand	PlcSimAdv.ED40	Adressierung des Eingangs-Doppelworts 40. Dieses besteht aus den beiden Wörtern EW40 und EW42 und somit den Bytes EB40 bis EB43. Das EB40 ist das HiByte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über AB54). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.5 Das Device „OPC UA"

Das Device mit der Bezeichnung „OPC UA" ist zu verwenden, wenn Operanden einer CodeSys V3-basierten SPS anzusprechen sind.

3.5.1 Unterstützte Operanden

Als Operanden werden Eingänge und Ausgänge unterstützt. Dabei sollten die Eingänge in der virtuellen Anlage von PLC-Lab beschrieben und die Ausgänge gelesen werden. Sowohl bei den Eingängen als auch bei den Ausgängen können die Bytes 0 bis 1023 von PLC-Lab angesprochen werden.

3.5.2 Adressierungsarten

Neben den CodeSys V3-Steuerungen können auch S7-Steuerungen von Siemens über OPC UA angesprochen werden, sofern die Steuerungen dies unterstützen. Dies ist z.B. bei den S7-1500 ab der Firmwareversion 2.X möglich. Ebenso bei PLCSIM Advanced ab V2.X. Da sich die Adressierung der Operanden bei CodeSys-Systemen und S7-Systemen unterscheidet, muss beim Erzeugen des OPC UA-Device in PLC-Lab entschieden werden, ob eine CodeSys-Steuerung oder eine S7-Steuerung von Siemens anzusprechen ist.

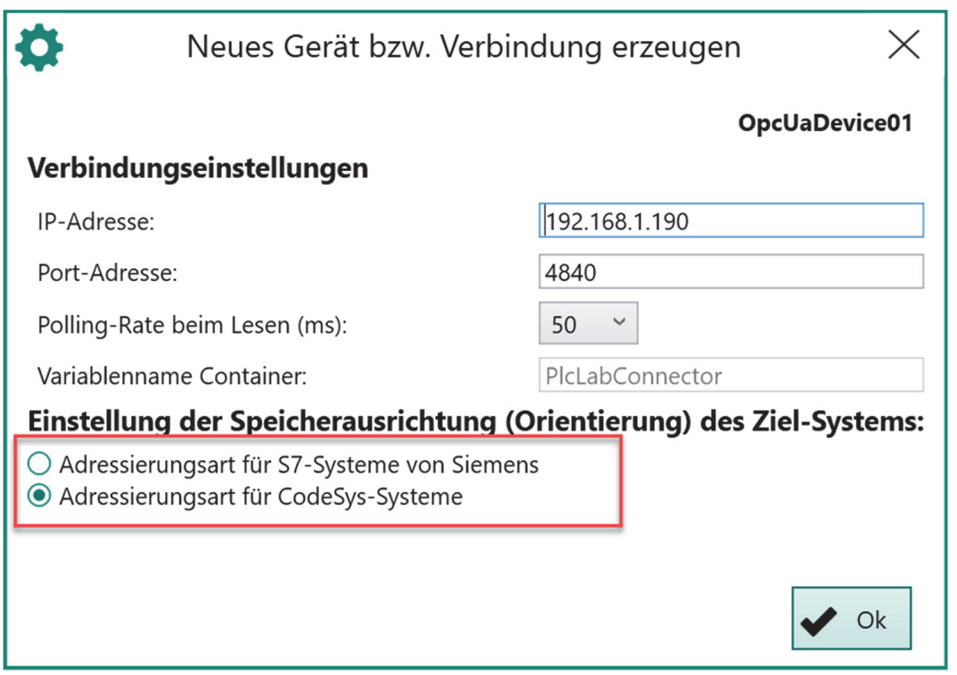


Bild: Einstellung der Speicherausrichtung bei eine OPC-UA Device.

Hier muss zwingend die korrekte Adressierungsart eingestellt werden. Nach dem Erzeugen des Device ist diese nicht mehr veränderbar.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.5.3 Adressierung der Eingänge und Ausgänge bei selektierter Adressierungsart für CodeSys-Systeme

Die Eingänge und Ausgänge können als Bit-, Byte-, Wort- oder Doppelwortoperanden angesprochen werden. In der Tabelle sind die Operanden dem Device mit der Bezeichnung "OpcUaCodeSys" zugeordnet.

Operand	Beispiel	Kommentar
Bit-Operand	OpcUaCodeSys.M10.3	An der Merker-Byteadresse 10 wird das Bit 3 adressiert.
Byte-Operand	OpcUaCodeSys.EB100	Die Byte-Adresse 100 wird adressiert.
Wort-Operand	OpcUaCodeSys.EW4	Das Wort 4 wird adressiert und somit die Bytes EB8 und EB9. EB9 ist das HiByte.
Doppelwort-Operand	OpcUaCodeSys.ED1	Das Doppelwort 1 wird adressiert und somit die Bytes EB4, EB5, EB6 und EB7. Das EB7 ist das HiByte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über EB100). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.5.4 Adressierung der Eingänge und Ausgänge bei selektierter Adressierungsart für S7-Systeme von Siemens

Die Eingänge und Ausgänge können als Bit-, Byte-, Wort- oder Doppelwortoperanden angesprochen werden. In der Tabelle sind die Operanden dem Device mit der Bezeichnung "OpcUaS7" zugeordnet.

Operand	Beispiel	Kommentar
Bit-Operand	OpcUaS7.M10.3	An der Merker-Byteadresse 10 wird das Bit 3 adressiert.
Byte-Operand	OpcUaS7.EB100	Die Byte-Adresse 100 wird adressiert.
Wort-Operand	OpcUaS7.EW4	Das Wort 4 wird adressiert und somit die Bytes EB4 und EB5. EB4 ist das HiByte.
Doppelwort-Operand	OpcUaS7.ED2	Das Doppelwort 2 wird adressiert und somit die Bytes EB2, EB3, EB4 und EB5. Das EB2 ist das HiByte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über EB100). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.5.5 Voraussetzungen für die Kommunikation zwischen PLC-Lab und einer CodeSys V3-basierten Steuerung über OPC UA

Damit PLC-Lab mit einer CodeSys V3-basierten Steuerung über OPC UA kommunizieren kann, sind ein paar Voraussetzungen auf Seiten der CodeSys-Steuerung zu schaffen.

Diese werden in einem kleinen Beispiel innerhalb des Wikis von PLC-Lab erläutert. Rufen Sie dazu den nachfolgenden Link auf:

https://www.mhj-wiki.de/de/plc-lab/device-opcua-codesys/

3.5.6 Voraussetzungen für die Kommunikation zwischen PLC-Lab und einer S7-1200/1500 über OPC UA

Damit PLC-Lab mit einer S7-1200/1500 (oder PLCSIM Advanced) über OPC UA kommunizieren kann, sind ein paar Voraussetzungen auf Seiten der Siemens-Steuerung zu schaffen.

Diese werden in einem kleinen Beispiel innerhalb des Wikis von PLC-Lab erläutert. Rufen Sie dazu den nachfolgenden Link auf:

https://www.mhj-wiki.de/de/plc-lab/device-opcua-siemens/

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.6 Das Device Sim

Das Sim-Device ist für die Zusammenarbeit mit **WinSPS-S7 V6** und **GRAFCET-Studio** zuständig.

Hinweis: Die Operanden des Sim-Device sollte man ebenfalls dann verwenden, wenn man eine neue Anlage entwickelt und den Ablauf ohne SPS-Programm oder GRAFCET testen möchte.

Eine mit Operanden des Sim-Device erstellte Anlage kann immer in RUN versetzt werden. Hier muss keine Hardware-konfiguration oder Ähnliches vorhanden sein.

Nach der Entwicklung der Anlagen können die Operanden dann zum Ziel-Device (z. B. PLCSim) **umverdrahtet** werden.

3.6.1 Die Operanden von Sim

Das Sim-Device unterstützt als Operanden Eingänge, Ausgänge, Merker und Daten von Datenbausteinen. Diese Operanden sind als Bit-, Byte-, Wort- oder Doppelwort-Operand adressierbar. Alle Operanden sind Byte-orientiert.

3.6.1.1 Adressierung von Eingängen, Ausgängen und Merkern

In der folgenden Tabelle sind Beispiele zur Adressierung von Operanden der obigen Bereiche angegeben.

Operand	Beispiel	Kommentar
Bit-Operand	Sim.M31.7	An der Merker-Byteadresse 31 wird das Bit 7 adressiert.
Byte-Operand	Sim.AB21	Adressierung des Ausgangs-Bytes mit der Adresse 21
Wort-Operand	Sim.EW100	Adressierung des Eingangs-Worts mit der Adresse 100. Dieses besteht aus den Bytes EB100 und EB101. Das EB100 ist das HiByte.
Doppelwort-Operand	Sim.ED62	Adressierung des Eingangs-Doppelworts 62. Dieses besteht aus den beiden Wörtern EW62 und EW64 und somit den Bytes EB62 bis EB65. Das EB62 ist das HiByte.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über AB21). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

3.6.1.2 Adressierung von Daten in Datenbausteinen

Ähnlich wie die anderen Operanden können auch die Daten von Datenbausteinen über Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Voraussetzung ist, dass sich der angesprochene Datenbaustein in der CPU (also in der S7-300-SoftSPS von Sim) befindet und dass der DB über die angesprochenen Daten verfügt.

Operand	Beispiel	Kommentar
Bit-Operand	Sim.DB23.DBX41.2	Im DB23 wird an der Byte-Adresse 41 das Bit 2 angesprochen.
Byte-Operand	Sim.DB1.DBB21	Im DB1 wird das Byte mit der Adresse 21 angesprochen.
Wort-Operand	Sim.DB50.DBW2	Adressierung des Daten-Worts mit der Adresse 2. Dieses besteht aus den Bytes DBB2 und DBB3. Das DBB2 ist das HiByte. Das Daten-Wort befindet sich im Datenbaustein 50.
Doppelwort-Operand	Sim.DB50.DBD40	Adressierung des Daten-Doppelworts 40. Dieses besteht aus den beiden Wörtern DBW40 und DBW42 und somit den Bytes DBB40 bis DBB43. Das DBB40 ist das HiByte. Das Datendoppelwort befindet sich im Datenbaustein 50.

Hinweis: Innerhalb der Symboltabelle werden die Operanden ohne den Device-Namen angegeben (also z. B. nur über DB1.DBB21). Die Beispiele in der obigen Tabelle sind die Operandenangaben innerhalb der Eigenschaften eines Objektes (z. B. Schalter). Normalerweise sollten Operanden mit Symbolen versehen werden, die dann innerhalb der Eigenschaften der Objekte verwendet werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.7 Das Device S7-300 und S7-400

Die **Pro-Edition** von PLC-Lab unterstützt die Kommunikation zu den CPUs der Systemfamilie **S7-300** und **S7-400** von Siemens. Ebenso können die **kompatiblen Steuerungen** der VIPA-Systeme SPEED7, SLIO und Micro angebunden werden.

Das SPS-Programm wird dabei von den realen CPUs verarbeitet und steuert die virtuelle Anlage von PLC-Lab. Parallel kann mit einer S7-Programmiersoftware das SPS-Programm beobachtet werden.

3.7.1 Verbindung über Ethernet bzw. Profinet

Verfügt die anzusprechende CPU über eine Ethernet- oder Profinet-Schnittstelle, dann ist kein Adapterkabel (MPI-Kabel) notwendig. Die Verbindung kann in einem solchen Fall direkt über die Netzwerkkarte des PCs hergestellt werden.

Nachfolgend ist der Einstellungsdialog zu sehen.

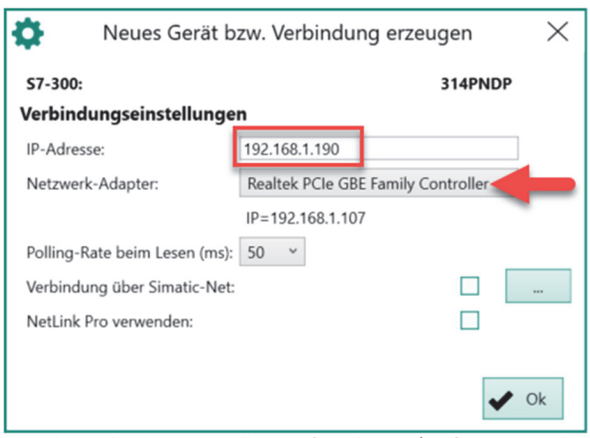


Bild: Verbindungseinstellungen für Ethernet/Profinet-Verbindung

Als IP-Adresse ist die IP-Adresse der anzusprechenden CPU anzugeben, im Beispiel 192.168.1.190. Des Weiteren ist der Netzwerk-Adapter zu selektieren, über den der PC mit der CPU verbunden ist. Kommt dabei kein Router zum Einsatz, dann müssen sich der PC und die CPU im gleichen Subnetz befinden.

3.7.2 Verbindung über NetLink Pro oder NetLink Pro Compact

Ist an der CPU keine Ethernet-Schnittstelle vorhanden, dann kann die Anbindung mit Hilfe eines NetLink Pro Compact oder NetLink Pro erfolgen. Dabei handelt es sich um Ethernet- nach MPI/DP-Umsetzer, mit denen man ebenfalls eine sehr schnelle Anbindung realisieren kann.

Soll diese Verbindung zum Einsatz kommen, dann ist im Einstellungsdialog die Option „NetLink Pro verwenden“ zu selektieren.

Bild: NetLink Pro soll verwendet werden.

Nach Betätigung der Option sind zusätzliche Einstellungen sichtbar. Diese werden nachfolgend benannt.

Einstellung	Bedeutung
IP-Adresse	Die IP-Adresse, auf welche der NetLink Pro bzw. der NetLink Pro Compact eingestellt ist.
Netzwerkadapter	Der Netzwerkadapter, über den der PC mit dem NetLink Pro verbunden ist.
PG-Adresse	Die MPI- bzw. Profibus-DP-Adresse des NetLink Pro. Diese Adresse muss eindeutig sein. Die Standard-PG-Adresse im MPI-Netz ist 0.
MPI/DP-Adresse CPU	Die MPI- bzw. DP-Adresse der anzusprechenden CPU.
Höchste MPI-Adresse	Die höchste im MPI/DP-Bus erlaubte Adresse. Mögliche Werte sind 15, 31, 63 und 126.
Baudrate MPI/DP	Die im MPI- bzw. Profibus-DP eingestellte Baudrate. Diese wird normalerweise automatisch vom NetLink Pro ermittelt. Ist es in Ausnahmefällen notwendig, die automatische Baudratenerkennung abzuschalten, dann ist die in der Bus-Konfiguration verwendete Baudrate anzugeben. Bei MPI ist dies normalerweise 187.5 kBaud.
Baudrate automatisch ermitteln	Wenn selektiert, dann versucht der NetLink Pro die Baudrate des MPI/DP-Netzwerks automatisch zu ermitteln (Standardeinstellung). Ist dies nicht möglich, dann wird die Option deselektiert und die Baudrate am Bus muss von Hand eingestellt werden.

3.7.3 Verbindung über Simatic Net

Eine Verbindung über **Simatic Net** steht zur Verfügung, wenn der Treiber dazu auf dem PC installiert ist. Dies ist dann der Fall, wenn sich eines der beiden Programme TIA-Portal oder Simatic-Manager auf dem PC befindet. Dann können auch die Siemens-MPI-Adapter (CP5711 oder USB-MPI-Adapter) zum Einsatz kommen.

Auf dem Einstellungsdialog ist dabei die Option „Verbindung über Simatic Net“ zu selektieren.

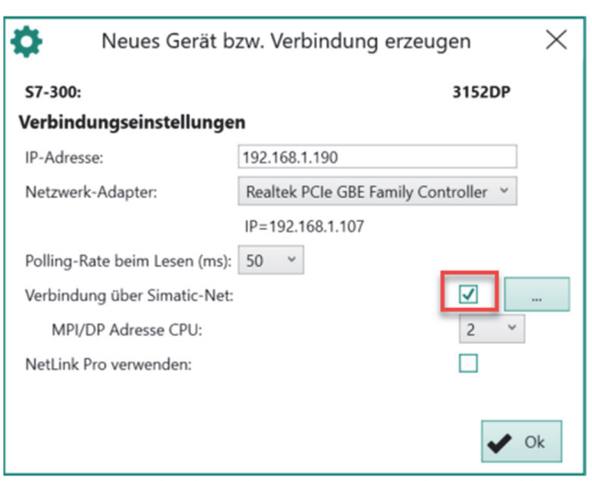


Bild: Verbindung über Simatic Net wurde ausgewählt.

Zusätzlich ist nur noch die MPI- bzw. DP-Adresse der anzusprechenden CPU anzugeben.

Über den Button […] ist der Dialog „PG/PC-Schnittstelle einstellen“ von Siemens erreichbar. Auf diesem Dialog sind die Einstellungen für das verwendete Interface von Siemens zu tätigen. Es handelt sich dabei um den gleichen Dialog, der auch innerhalb des Simatic-Managers zum Einsatz kommt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.7.4 Die Operanden bei Verbindungen zu S7-300/400

3.7.5 Verwendung von Eingängen

Eingänge können in der PLC-Lab Anlage sowohl gelesen als auch beschrieben werden. Wird ein Eingang beschrieben, dann schreibt PLC-Lab den neuen Status des Eingangs in die CPU (dies ist der Normalfall).

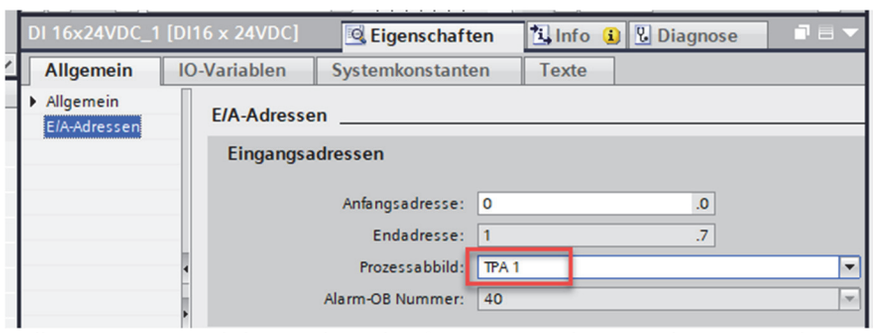
Ist der Eingang in der CPU als Modul vorhanden, dann wird der von PLC-Lab geschriebene Status sofort vom realen Status des Eingangs am Modul überschrieben. Ist keine Eingangsbaugruppe mit dieser Adresse vorhanden, dann passiert dies nicht und der Status des von PLC-Lab geschriebenen Eingangs hat Bestand.

Meist ist es so, dass ein Eingang real als Baugruppe vorhanden ist. Das Problem des Überschreibens kann dann wie folgt gelöst werden.

3.7.5.1 Aktualisierung des Prozessabbildes der Eingangsmodule abschalten

Der Status eines von PLC-Lab geschriebenen Eingangs wird deshalb überschrieben, weil das Betriebssystem der CPU zyklisch die Daten aus der realen Baugruppe lädt und in das Prozessabbild der Eingänge schreibt. Dieser Vorgang kann für jedes Eingangsmodul separat abgeschaltet werden. Im folgenden Bild ist die betreffende Einstellung zu sehen.

Bild: Abschalten des Prozessabbilds



Wurde diese Einstellung getätigt und die Hardwarekonfiguration in die CPU übertragen, dann wird der Wert des von PLC-Lab geschriebenen Eingangs im Adressbereich E0.0 bis E1.7 **nicht** mehr überschrieben.

Nach der Simulation mit PLC-Lab kann dann die Einstellung wieder zurückgeführt werden.

3.7.5.2 Abändern der Adressen der realen Eingangsbaugruppen

Sind für die Eingangsadressbereiche E0.0 bis E10.7 Module vorhanden und sollen diese Eingänge von PLC-Lab beschrieben werden, dann könnten für die Dauer der Simulation die Adressen der realen Module auf einen **freien** Eingangs-Adressbereich gelegt werden. Beispielsweise auf die Adressen E100.0 bis E110.7.

Somit würden die von PLC-Lab beschriebenen Eingänge nicht mehr vom Status der realen Module überschrieben werden.

Nach der Simulation mit PLC-Lab werden die Module wieder auf ihre alten Adressbereiche umgestellt.

Es ist darauf zu achten, dass die Änderungen in der Hardwarekonfiguration der CPU erst Gültigkeit haben, wenn sie auch in die CPU **übertragen** wurden.

3.7.5.3 Überschreiben der Eingänge mit dem Inhalt von Merkern oder Daten aus DBs

Kommen die beiden letzten Varianten nicht in Betracht, dann besteht noch die Möglichkeit, dass PLC-Lab anstatt der Eingänge in **Merker** schreibt. Diese Merker können dann dazu verwendet werden, im Netzwerk 1 des OB1 die gewünschten Eingänge zu überschreiben.

Beispiel:

In einer virtuellen Anlage von PLC-Lab wird das Merker-Byte MB20 beschrieben. Dieses soll das Eingangsbyte EB100 in der SPS beeinflussen bzw. überschreiben. Für dieses Eingangsbyte ist ein reales Modul vorhanden.

Bei dieser Aufgabenstellung wird im Netzwerk 1 des OB1 ein **MOVE-Block** programmiert. Somit wird der Inhalt des MB20 in das EB100 geschrieben. Im nachfolgenden Bild sind die SPS-Operationen zu sehen.

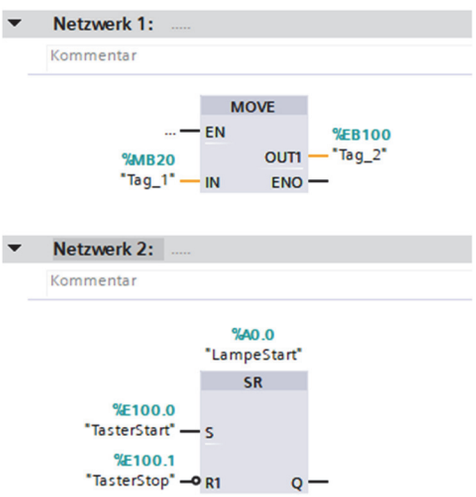


Bild: SPS-Programm für das Beispiel

Im Bild ist ebenfalls zu erkennen, dass die nachfolgenden Zugriffe auf die Eingänge (E100.0 und E100.1) so erhalten bleiben können. Im weiteren SPS-Programm müssen also keine Änderungen vorgenommen werden.

Nach dem Test mit PLC-Lab wird einfach das Netzwerk 1 gelöscht, und damit haben wieder die realen Eingänge am Modul ihren Einfluss zurück.

3.7.5.4 Adressierung von Eingängen

Eingänge können als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Es ist dabei **unerheblich,** ob die **deutsche** oder **englische Syntax** verwendet wird.

Beispiele sind:

- Bit-Zugriff: E10.1, I21.7
- Byte-Zugriff: EB21, IB33
- Wort-Zugriff: EW120, IW30
- Doppelwort-Zugriff: ED34, ID20

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 314PNDP und es soll das EB24 angesprochen werden, dann lautet die Angabe „314PNDP.EB24“.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.7.6 Verwendung von Ausgängen und Merkern

Ausgänge können in den Anlagen von PLC-Lab beschrieben oder gelesen werden. Normalerweise werden sie gelesen und Aktoren innerhalb der virtuellen Anlagen von PLC-Lab angesteuert. Es ist unerheblich, ob zu den Ausgängen auch reale Module vorhanden sind.

Merker können ebenfalls gelesen und beschrieben werden. Hier ist kein Standardzugriff vorhanden.

3.7.6.1 Adressierung von Ausgängen und Merkern

Ausgänge und Merker können als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Es ist dabei unerheblich, ob die deutsche oder englische Syntax verwendet wird. Bei Merkern wird das **Operandenkennzeichen** „**M**“ sowohl bei der deutschen als auch bei der englischen Syntax verwendet.

Beispiele sind:

- Bit-Zugriff: A10.1, M12.5, Q21.7
- Byte-Zugriff: AB21, MB45, QB33
- Wort-Zugriff: AW120, MW30, QW32
- Doppelwort-Zugriff: AD34, MD120, QD20

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 314PNDP und es soll das AB24 angesprochen werden, dann lautet die Angabe „314PNDP.AB24“.

3.7.7 Verwendung von Daten in Datenbausteinen

Die Inhalte von Datenbausteinen können in PLC-Lab gelesen und beschrieben werden. **Voraussetzung ist, dass sich der DB in der CPU befindet und die angesprochenen Daten (Adressen) im DB vorhanden sind**.

3.7.7.1 Adressierung von Daten in Datenbausteinen

DB-Daten können als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Die deutsche und englische Syntax ist identisch.

Beispiele sind:

- Bit-Zugriff: DB5.DBX13.7
- Byte-Zugriff: DB30.DBB24
- Wort-Zugriff: DB2.DBW10
- Doppelwort-Zugriff: DB33.DBD10

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 314PNDP und es soll das DBB24 im Datenbaustein 30 angesprochen werden, dann lautet die Angabe „314PNDP.DB30.DBB24“.

3.8 Das Device S7-1200 und S7-1500

Die **Micro-Edition** von PLC-Lab kann mit einer **S7-1200** von Siemens zusammenarbeiten, die **Pro-Edition** sowohl mit einer **S7-1200** als auch mit einer **S7-1500** von Siemens.

Die Verbindung zwischen dem PC und der CPU wird bei beiden SPS-Familien mit Hilfe der **Profinet-Schnittstelle** auf der CPU hergestellt. Dabei wird die Netzwerkkarte des PCs direkt mit der Schnittstelle auf der CPU verbunden.

Das SPS-Programm befindet sich in der realen CPU und steuert die virtuelle Anlage von PLC-Lab. Parallel kann mit dem TIA-Portal das SPS-Programm beobachtet werden.

3.8.1 Notwendige Einstellungen in einer S7-1200 oder S7-1500 für den Betrieb mit PLC-Lab

Die nachfolgenden Einstellungen innerhalb der Rubrik „Schutz & Security“ der CPU sind notwendig, damit PLC-Lab mit einer S7-1200/1500 kommunizieren kann.

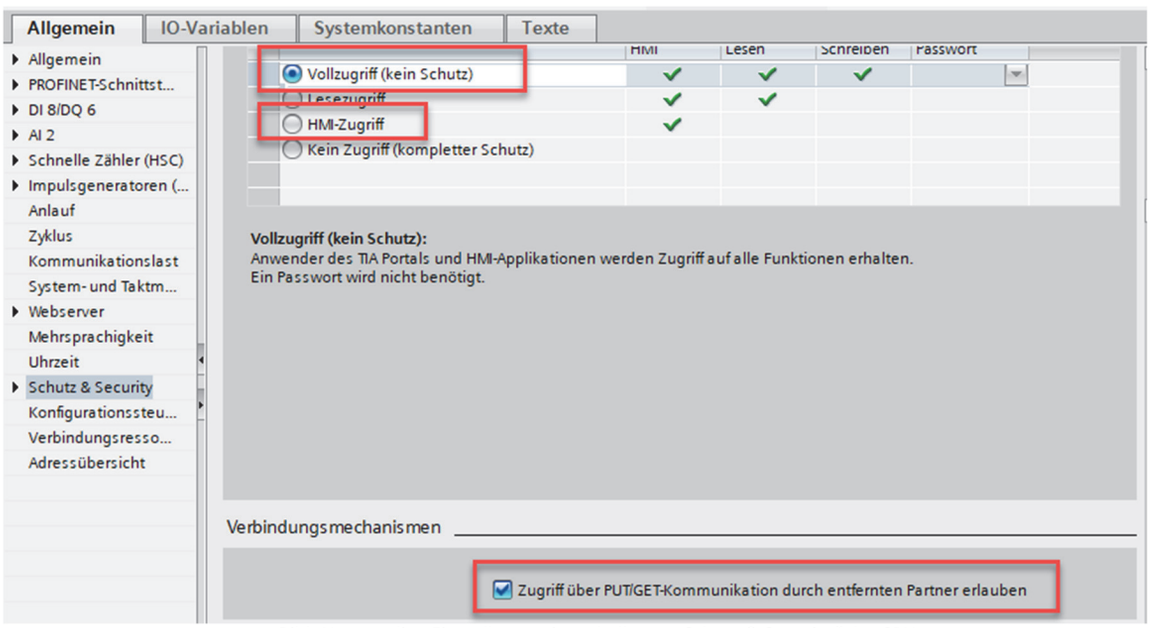


Bild: Notwendige Einstellungen innerhalb von „Schutz & Security“ der CPU

Es ist sicherzustellen, dass entweder Vollzugriff (kein Schutz) oder HMI-Zugriff ausgewählt ist. **Zusätzlich** muss die Option „Zugriff über PUT/GET-Kommunikation … erlauben“ selektiert sein.

Nachdem die Einstellungen vorgenommen wurden, ist die Hardwarekonfiguration in die CPU zu übertragen. Erst danach sind die Einstellungen in der SPS auch wirklich wirksam.

3.8.2 Erzeugen eines neuen Device in PLC-Lab für den Zugriff auf eine S7-1200 oder S7-1500 von Siemens

Sollen Daten zwischen PLC-Lab und einer S7-1200/1500 von Siemens ausgetauscht werden, dann ist zunächst ein neues Device vom Typ „S7-1200“ bzw. „S7-1500“ anzulegen. Dazu wird in der Symboltabelle die Schaltfläche „Ein neues Device erzeugen“ betätigt.

Bild: Neues Device in PLC-Lab erzeugen

Als Folge ist der Dialog „Neues Gerät bzw. Verbindung erzeugen“ zu sehen. Auf diesem ist als Device-Typ „S7-1200“ oder „S7-1500“ zu selektieren. Weiterhin ist eine Bezeichnung für das neue Device anzugeben.

© MHJ-Software GmbH & Co. KG - PLC-Lab

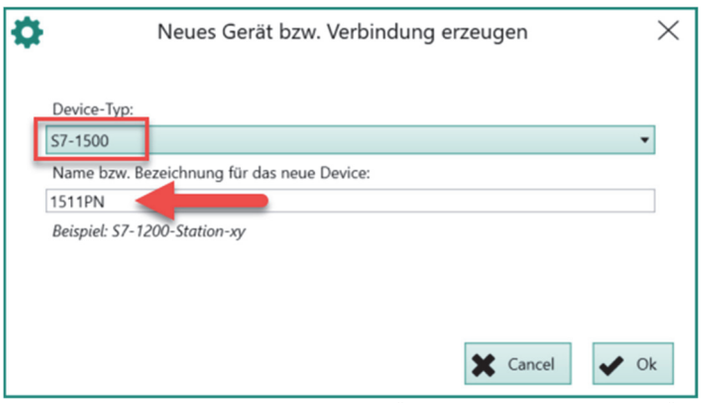


Bild: Neues Device vom Typ „S7-1500“ mit der Bezeichnung „1511PN“

Über „OK“ wird die Eingabe bestätigt und die Verbindungseinstellungen des Device erscheinen.

Hier ist die IP-Adresse der CPU und die zu verwendende Netzwerkkarte anzugeben.

Bild: Verbindungseinstellungen der S7-1200/1500

Eine erneute Betätigung des „OK“-Buttons bestätigt auch diese Einstellungen und schließt das Erzeugen des neuen Device ab. Das Device wird in der Symboltabelle voreingestellt, damit sofort mit dem Erzeugen der Operanden für das Device begonnen werden kann.

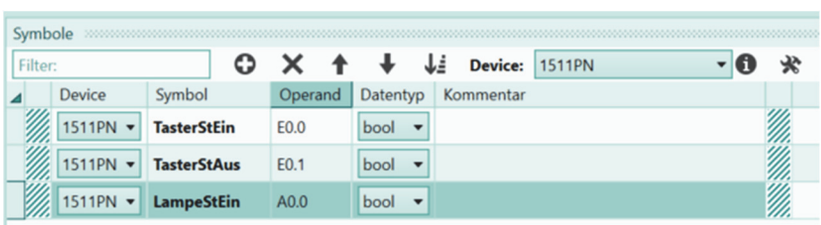


Bild: Symboltabelle mit den neu erzeugten Operanden des Device „1511PN“

3.8.3 Verwendung von Eingängen

Eingänge können in der PLC-Lab Anlage sowohl gelesen als auch beschrieben werden. Wird ein Eingang beschrieben, dann schreibt PLC-Lab den neuen Wert des Eingangs in die CPU.

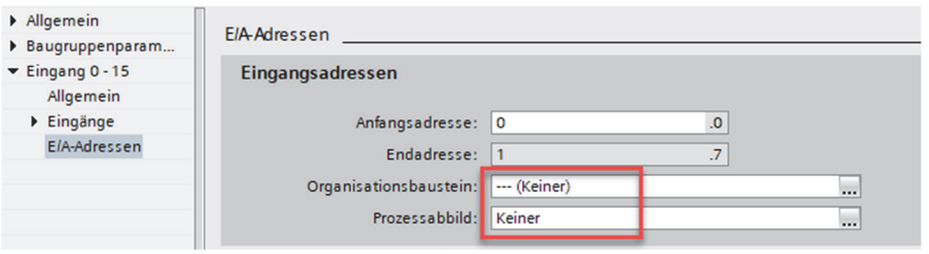
Ist der Eingang in der CPU als Modul vorhanden, dann wird der von PLC-Lab geschriebene Status sofort vom realen Status des Eingangs am Modul **überschrieben**. Ist keine Eingangsbaugruppe mit dieser Adresse vorhanden, dann passiert dies nicht und der Wert des von PLC-Lab geschriebenen Eingangs bleibt erhalten.

Meist ist es so, dass ein Eingang real als Baugruppe vorhanden ist. **Das „Überschreiben“-Problem kann wie folgt gelöst werden.**

3.8.3.1 Aktualisierung des Prozessabbildes der Eingangsmodule abschalten

Der Status eines von PLC-Lab geschriebenen Eingangs wird deshalb überschrieben, weil das Betriebssystem der CPU zyklisch die Daten aus der realen Baugruppe lädt und in das **Prozessabbild der Eingänge** schreibt. Dieser Vorgang kann für jedes Eingangsmodul separat abgeschaltet werden. Im folgenden Bild ist die betreffende Einstellung zu sehen.

Bild: Abschalten des Prozessabbilds



Wurde diese Einstellung getätigt und die Hardwarekonfiguration in die CPU übertragen, dann wird der Status des von PLC-Lab geschriebenen Eingangs im Adressbereich E0.0 bis E1.7 nicht mehr überschrieben.

Nach der Simulation mit PLC-Lab kann dann die Einstellung wieder zurückgeführt werden.

3.8.3.2 Abändern der Adressen der realen Eingangsbaugruppen

Sind für die Eingangsadressbereiche E0.0 bis E10.7 Module vorhanden und sollen diese Eingänge von PLC-Lab beschrieben werden, dann können für die Dauer der Simulation die Adressen der realen Module auf einen **freien Eingangs-Adressbereich** gelegt werden. Beispielsweise auf die Adressen E100.0 bis E110.7.

Somit würden die von PLC-Lab beschriebenen Eingänge nicht mehr vom Status der realen Module überschrieben werden.

Nach der Simulation mit PLC-Lab werden die Module wieder auf ihre alten Adressbereiche umgestellt.

Es ist darauf zu achten, dass die Änderungen in der Hardwarekonfiguration der CPU erst Gültigkeit haben, wenn sie auch in die CPU übertragen wurden.

3.8.3.3 Überschreiben der Eingänge mit dem Inhalt von Merkern oder Daten aus DBs

Kommen die beiden letzten Varianten nicht in Betracht, dann besteht noch die Möglichkeit, dass PLC-Lab anstatt der Eingänge in Merker schreibt. Diese Merker können dann dazu verwendet werden, im Netzwerk 1 des OB1 die gewünschten Eingänge zu überschreiben.

Beispiel:

In einer virtuellen Anlage von PLC-Lab wird das Merker-Byte MB20 beschrieben. Damit ist das Eingangsbyte EB100 in der SPS zu beeinflussen bzw. zu überschreiben. Zum Eingangsbyte ist ein reales Modul vorhanden.

Für diese Aufgabenstellung wird im Netzwerk 1 des OB1 ein **MOVE-Block** programmiert. Damit wird der Inhalt des MB20 in das EB100 geschrieben. Im nachfolgenden Bild sind die SPS-Operationen zu sehen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

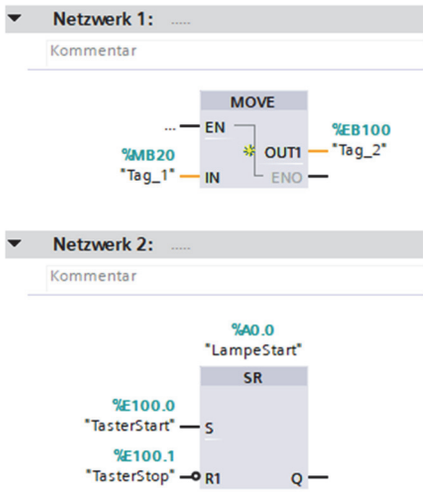


Bild: SPS-Programm für das Beispiel

Im Bild ist ebenfalls zu erkennen, dass die nachfolgenden Zugriffe auf die Eingänge (E100.0 und E100.1) so **erhalten** bleiben können. Im weiteren SPS-Programm müssen also **keine Änderungen** vorgenommen werden.

Nach dem Test mit PLC-Lab wird einfach das Netzwerk 1 gelöscht und damit haben wieder die realen Eingänge am Modul ihren Einfluss zurück.

3.8.3.4 Adressierung von Eingängen

Eingänge können als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Es ist dabei unerheblich, ob die deutsche oder englische Syntax verwendet wird.

Beispiele sind:

- Bit-Zugriff: E10.1, I21.7
- Byte-Zugriff: EB21, IB33
- Wort-Zugriff: EW120, IW30
- Doppelwort-Zugriff: ED34, ID20

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 1511PN und es soll das EB21 angesprochen werden, dann lautet die Angabe „1511PN.EB21“.

3.8.4 Verwendung von Ausgängen und Merkern

Ausgänge können in den Anlagen von PLC-Lab beschrieben oder gelesen werden. Normalerweise werden sie gelesen und Aktoren innerhalb der virtuellen Anlagen von PLC-Lab angesteuert. Es ist unerheblich, ob zu den Ausgängen auch reale Module vorhanden sind.

Merker können ebenfalls gelesen und beschrieben werden. Hier ist kein Standardzugriff vorhanden.

3.8.4.1 Adressierung von Ausgängen und Merkern

Ausgänge und Merker kann man als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressieren. Es ist dabei unerheblich, ob die deutsche oder englische Syntax verwendet wird. Bei Merkern wird das Operandenkennzeichen „M“ sowohl bei der deutschen als auch bei der englischen Syntax verwendet.

Beispiele sind:

- Bit-Zugriff: A10.1, M12.5, Q21.7
- Byte-Zugriff: AB24, MB45, QB33
- Wort-Zugriff: AW120, MW30, QW32
- Doppelwort-Zugriff: AD34, MD120, QD20

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 1511PN und es soll das AB24 angesprochen werden, dann lautet die Angabe „1511PN.AB24“.

3.8.5 Verwendung von Daten in Datenbausteinen

Die Inhalte von Datenbausteinen können in PLC-Lab gelesen und beschrieben werden. Voraussetzung ist, dass sich der DB in der CPU befindet und die angesprochenen Daten im DB vorhanden sind.

Hinweis: Der Zugriff auf optimierte DBs wird nicht unterstützt.

3.8.5.1 Adressierung von Daten in Datenbausteinen

DB-Daten können als Bit-, Byte-, Wort- oder Doppelwort-Operanden adressiert werden. Die deutsche und englische Syntax ist identisch.

Beispiele sind:

- Bit-Zugriff: DB5.DBX13.7
- Byte-Zugriff: DB30.DBB2
- Wort-Zugriff: DB2.DBW10
- Doppelwort-Zugriff: DB33.DBD10

Hinweis: Wird ein Operand innerhalb der Eigenschaft eines Objekts verwendet, dann ist zusätzlich noch der Device-Name mit anzugeben. Hat das Device z. B. die Bezeichnung 1511PN und es soll das DBB24 im Datenbaustein 30 angesprochen werden, dann lautet die Angabe „1511PN.DB30.DBB24“.

© MHJ-Software GmbH & Co. KG - PLC-Lab

3.9 Das Device Siemens-LOGO!

PLC-Lab kann zum Test des Programms in einer **LOGO!-Steuerung** von Siemens verwendet werden. Dabei läuft das Programm in einer realen LOGO! und PLC-Lab ist über Ethernet mit der LOGO! von Siemens verbunden.

Parallel dazu ist das **Beobachten** des Programms der LOGO! über die Programmiersoftware LOGO!Soft-Comfort von Siemens **möglich**.

3.9.1 Welche LOGO!-Steuerungen werden unterstützt?

Da PLC-Lab über Ethernet mit einer LOGO! von Siemens verbunden wird, werden alle LOGO!s mit Ethernet-Schnittstelle unterstützt. Dazu gehören:

- 0BA7
- 0BA8 oder höher.

3.9.2 Erzeugen eines neuen Device in PLC-Lab für den Zugriff auf eine LOGO! von Siemens

Sollen Daten zwischen PLC-Lab und einer LOGO! von Siemens ausgetauscht werden, dann ist zunächst ein neues Device vom Typ „Siemens-Logo“ anzulegen. Dazu wird in der Symboltabelle die Schaltfläche „Ein neues Device erzeugen“ betätigt.

Bild: Neues Device in PLC-Lab erzeugen

Als Folge ist der Dialog „Neues Gerät bzw. Verbindung erzeugen“ zu sehen. Auf diesem ist als Device-Typ „Siemens-Logo“ zu selektieren. Weiterhin ist eine Bezeichnung für das neue Device anzugeben.

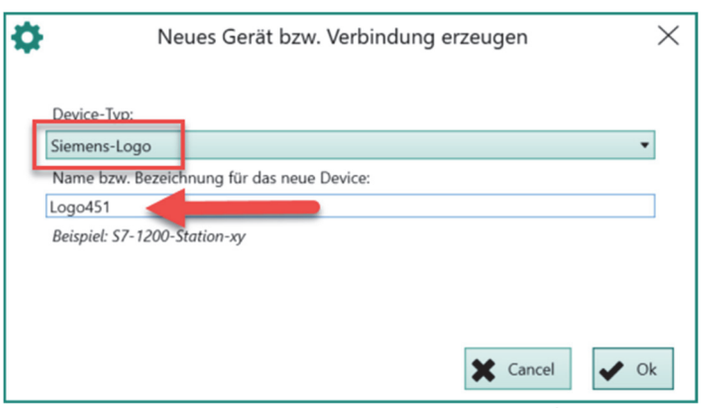


Bild: Selektion des Device-Typs und Angabe der Bezeichnung für das neue Device

Über „OK“ werden die Angaben bestätigt und man gelangt zum nächsten Schritt. Hier kann die IP-Adresse der LOGO! und die Netzwerkkarte, an welcher sie angeschlossen ist, selektiert werden.

Bild: Angabe der IP-Adresse der anzusprechenden LOGO!-Steuerung

Der „OK“-Button bestätigt auch diese Einstellungen.

In der Symboltabelle ist das neu erzeugte Device aktiv, es können sogleich Operanden für das Device angelegt werden.

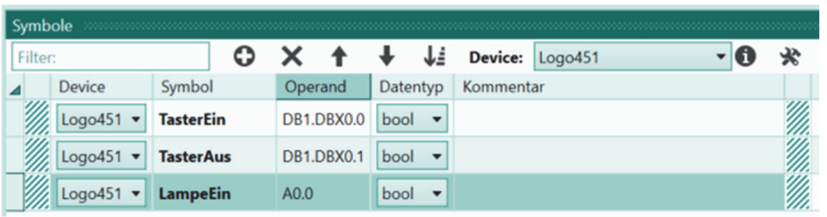


Bild: Operanden des neuen Device

3.9.3 Verwendung von digitalen Eingängen

Eingänge kann man bei einer LOGO! von Siemens **nicht** beschreiben. Dies bedeutet, würde man Eingänge über PLC-Lab schreibend beeinflussen, dann hätte dies auf den Eingang der LOGO! **keine** Auswirkungen.

Anstatt der Eingänge muss man innerhalb von PLC-Lab sog. Netzwerkeingänge der LOGO! verwenden. Diese Netzwerkeingänge werden in der LOGO! über den **V-Bereich** adressiert. Die Adresse des V-Bereichs beginnt bei V0.0. **Innerhalb von PLC-Lab wird der DB1 verwendet, um den V-Bereich der LOGO! zu adressieren.** Möchte man das Bit V0.0 im V-Bereich der LOGO! über PLC-Lab ansprechen, dann gibt man in PLC-Lab den Operanden DB1.DBX0.0 an.

Beispiel:

In einer Anlage ist ein Taster an dem Eingang I1 der LOGO! angeschlossen. Die Funktion des Tasters soll mit PLC-Lab simuliert werden.

Dazu wird im LOGO!-Programm zunächst der I1 auf einen OR-Block geführt. Zusätzlich wird ein Netzwerk-Eingang eingefügt und ebenfalls am OR-Block angeschlossen. Der Netzwerk-Eingang hat die Adresse V0.0, ist also in PLC-Lab über die Adresse DB1.DBX0.0 anzusprechen.

Somit kann im LOGO!-Programm das Signal vom realen Taster ausgelöst werden oder auch vom virtuellen Taster in PLC-Lab.

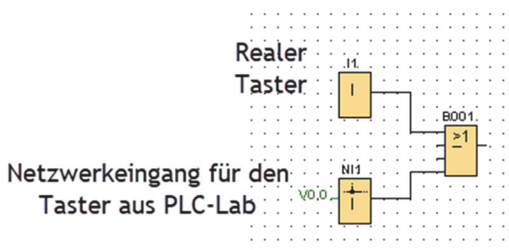


Bild: Verwendung des Netzwerk-Eingangs

Fazit: Da man den Status von digitalen Eingängen einer LOGO! von Siemens nicht direkt überschreiben kann, muss deren Funktion durch Netzwerkeingänge übernommen werden. Die bevorzugte Vorgehensweise ist, den Eingang und den Netzwerkeingang auf einen OR-Block zu verschalten und den Ausgang des OR-Blocks weiter zu verwenden. Dadurch kann das Schaltprogramm nach der Simulation unverändert in der realen Anlage verwendet werden. Eine spätere Simulation des Programms ist jederzeit möglich. Die Netzwerkeingänge werden mit Bit-Adressen des V-Bereichs in der LOGO! verbunden. Diese V-Operanden werden in PLC-Lab über den DB1 angesprochen. Nachfolgende Tabelle zeigt den Zusammenhang zwischen V-Operanden und dem DB1 in PLC-Lab.

© MHJ-Software GmbH & Co. KG - PLC-Lab

V-Bereich der LOGO!	DB1-Bereich in PLC-Lab
V0.0	DB1.DBX0.0
V0.1	DB1.DBX0.1
V0.2	DB1.DBX0.2
V0.3	DB1.DBX0.3
V0.4	DB1.DBX0.4
V0.5	DB1.DBX0.5

3.9.4 Verwendung von analogen Eingängen

Ähnlich wie bei den digitalen Eingängen ist es auch bei den **analogen Eingängen** einer LOGO! von Siemens nicht möglich, sie direkt zu beschreiben. Die Lösung des Problems ist auch hier die Verwendung von Netzwerkeingängen, genauer gesagt von **analogen Netzwerkeingängen**.

Die analogen Netzwerkeingänge werden in der LOGO! über den V-Bereich adressiert, wobei sie bei VW0 beginnen und die Breite eines Worts (16 Bit) haben. In PLC-Lab werden sie über den DB1 adressiert, will man also das VW10 in PLC-Lab ansprechen, dann wird das DB1.DBW10 angesprochen.

Beispiel:

Im Schaltprogramm einer LOGO! von Siemens wird der analoge Eingang AI1 verwendet. Dessen Signal soll für den Test des Programms von PLC-Lab geliefert werden.

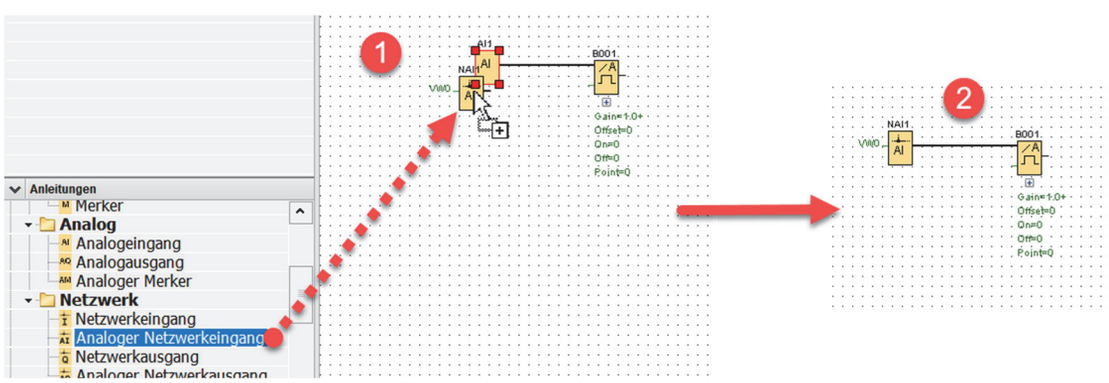


Bild: Der analoge Eingang wird durch einen analogen Netzwerkeingang ersetzt.

Um dies zu erreichen, ist der AI1 durch einen analogen Netzwerkeingang zu **ersetzen**. Dabei wird einfach ein analoger Netzwerkeingang aus dem Katalog entnommen und per Drag-and-Drop über dem analogen Eingang fallen gelassen (Punkt 1). Der analoge Eingang wird dann durch den analogen Netzwerkeingang ersetzt (Punkt 2).

Führt man einen Doppelklick auf dem analogen Netzwerkeingang aus, dann ist die Adresse veränderbar. Im Beispiel hat der analoge Netzwerkeingang die Adresse VW0.

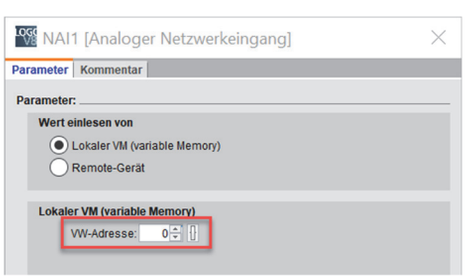


Bild: Dialog des analogen Netzwerkeingangs

Somit ist der analoge Netzwerkeingang in PLC-Lab über die Adresse DB1.DBW0 anzusprechen.

Fazit: Da man analoge Eingänge einer LOGO! von Siemens nicht direkt überschreiben kann, muss deren Funktion durch analoge Netzwerkeingänge übernommen werden. Die bevorzugte Vorgehensweise ist, den Eingang durch einen Netzwerkeingang zu ersetzen. Nach der Simulation kann dann der analoge Netzwerkeingang wieder durch den analogen Eingang ersetzt werden. Dieser Vorgang ist in der Programmiersoftware sehr schnell zu erledigen (siehe obiges Beispiel). Die analogen Netzwerkeingänge sind mit Wort-Adressen des V-Bereichs in der LOGO! verbunden. Diese V-Operanden sind in PLC-Lab über den DB1 anzusprechen. Nachfolgende Tabelle zeigt den Zusammenhang zwischen V-Operanden und dem DB1 in PLC-Lab.

Hinweis: Bei der Verwendung ist darauf zu achten, dass man die Wörter im V-Bereich nur geradzahlig adressiert, da diese Byte-orientiert sind und sich ansonsten Überschneidungen ergeben können.

V-Bereich der LOGO!	DB1-Bereich in PLC-Lab
VW0	DB1.DBW0
VW2	DB1.DBW2
VW4	DB1.DBW4
VW6	DB1.DBW6
VW8	DB1.DBW8
VW10	DB1.DBW10

3.9.5 Verwendung von digitalen Ausgängen

Digitale Ausgänge einer LOGO! von Siemens können in PLC-Lab beschrieben und gelesen werden. Normalerweise werden sie gelesen und mit ihrem Status die Aktoren in PLC-Lab beeinflusst. Als Beispiel sei die Fahrt eines Zylinders zum Maximum genannt.

In der folgenden Tabelle ist angegeben, wie die digitalen Ausgänge einer LOGO! in PLC-Lab adressiert werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Ausgang der LOGO!	Adresse in PLC-Lab
Q1	A0.0
Q2	A0.1
Q3	A0.2
Q4	A0.3
Q5	A0.4
Q6	A0.5
Q7	A0.6
Q8	A0.7
Q9	A1.0

Hinweis: Die verwendeten digitalen Ausgänge müssen nicht real auf der LOGO! vorhanden sein.

3.9.6 Verwendung von analogen Ausgängen

Analoge Ausgänge einer LOGO! von Siemens können in PLC-Lab nur gelesen werden. Man kann also beispielsweise den Zahlenwert anzeigen lassen oder sie als Geschwindigkeitsvorgabe für eine Linearbewegung verwenden.

Die analogen Ausgänge einer LOGO! sind über den V-Bereich anzusprechen. Dabei ist die Adresse der Ausgänge von LOGO! zu LOGO! unterschiedlich.

Hinweis: In der „LOGO!Soft Comfort“ Online-Hilfe erhalten Sie eine Übersicht der V-Adressbelegungen innerhalb des Hilfethemas „Extras->Parameter->VM-Zuordnung (nur 0BA7 und höher)“.

In der folgenden Tabelle sind die Adressen für eine **0BA7** angegeben:

Analoger Ausgang der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
AQ1	VW944	DB1.DBW944
AQ2	VW946	DB1.DBW946

Nachfolgend die Adressen einer **0BA8**:

Analoger Ausgang der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
AQ1	VW1072	DB1.DBW1072
AQ2	VW1074	DB1.DBW1074
…	…	…
AQ16	VW1102	DB1.DBW1102

3.9.7 Verwendung von digitalen Merkern

Digitale Merker erlauben sowohl einen lesenden als auch einen schreibenden Zugriff. Die Adressierung erfolgt über den V-Bereich der LOGO! von Siemens, was bedeutet, dass sie in PLC-Lab über den DB1 angesprochen werden.

Die Lage der Merker innerhalb des V-Bereichs ist je nach verwendetem Gerät unterschiedlich. Hier zunächst die Adressen einer **0BA7**:

Digitaler Merker der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
M1	V948.0	DB1.DBX948.0
M2	V948.1	DB1.DBX948.1
…	…	…
M15	V949.6	DB1.DBX949.6
M16	V949.7	DB1.DBX949.7

Nun die Adressen einer **0BA8**:

Digitaler Merker der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
M1	V1104.0	DB1.DBX1104.0
M2	V1104.1	DB1.DBX1104.1
…	…	…
M…	V1117.6	DB1.DBX117.6

3.9.8 Verwendung von analogen Merkern

Analoge Merker können in PLC-Lab gelesen oder beschrieben werden. Die Adressierung erfolgt über den V-Bereich der LOGO!. Die Adresse im V-Bereich ist wiederum von der verwendeten LOGO! abhängig. Anbei die Adressen für eine **0BA7**:

Analoger Merker der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
AM1	VW952	DB1.DBW952
AM2	VW954	DB1.DBW954
…	…	…
AM16	VW982	DB1.DBW982

Und für eine **0BA8**:

Analoger Merker der LOGO!	Adresse im V-Bereich	Adresse in PLC-Lab
AM1	VW1118	DB1.DBW1118
AM2	VW1120	DB1.DBW1120
…	…	…
AM64	VW1244	DB1.DBW1244

© MHJ-Software GmbH & Co. KG - PLC-Lab

4 Verwendung von Symbolen in PLC-Lab

Ähnlich wie beim Erstellen eines SPS-Programms sollten auch beim Erstellen eines neuen PLC-Lab Anlagenprojekts zunächst die Symbole für die Operanden festgelegt werden. Die Bezeichnungen sind dabei aussagekräftig zu wählen, auch wenn sie dadurch etwas länger sind. Denn im Normalfall muss ein Symbol nicht nochmals von Hand eingetippt werden.

Soll ein Symbol einer Objekt-Eigenschaft zugeordnet werden, dann wird dies per Drag-and-Drop oder mit Hilfe von „Intellisense“ (Autovervollständigung) ausgeführt. Bei beiden Varianten muss man das Symbol nicht mehr von Hand eingeben. Wie dabei vorzugehen ist, wird im weiteren Verlauf erläutert.

Oftmals wird eine PLC-Lab Testumgebung erstellt und ein SPS-Programm bzw. ein GRAFCET ist bereits vorhanden. In diesem Fall wurden beispielsweise im TIA-Portal oder in GRAFCET-Studio Symbole für Operanden festgelegt. Diese können dann in PLC-Lab **importiert** und verwendet werden.

Gleiches gilt, wenn die Anlage als erstes in PLC-Lab erstellt wird und bereits Symbole festgelegt wurden. In diesem Fall würde man die Symbole in PLC-Lab **exportieren** und im Ziel-Programmiersystem (TIA, WinSPS-S7, Simatic-Manager, GRAFCET-Studio usw.) **importieren**.

4.1 Aus welchen Teilen besteht ein Symboleintrag?

Zunächst sollen die Bestandteile eines Symboleintrags benannt werden.

4.1.1 Device

Im letzten Abschnitt wurden die verschiedenen **Devices** vorgestellt, mit denen PLC-Lab zusammenarbeiten kann. Des Weiteren wurde erwähnt, dass in einem PLC-Lab Anlagenprojekt unterschiedliche Devices verwendet werden können. Zu diesen Devices wird dann beim Start der Simulation eine Verbindung aufgebaut und der Datenaustausch gestartet.

Nachfolgendes Bild zeigt eine Symboltabelle, in der Symbole für Operanden von drei unterschiedlichen Devices vorhanden sind.

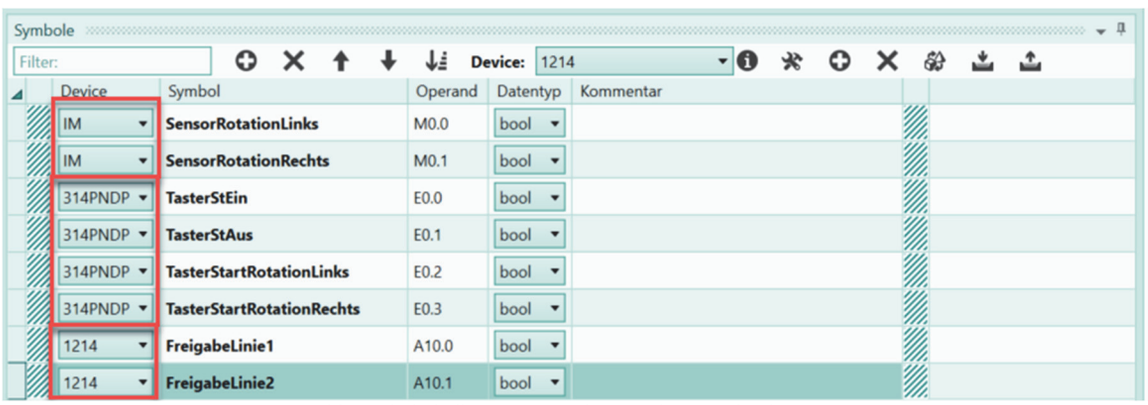


Bild: Symboltabelle mit Operanden von drei verschiedenen Devices

Beim Device „314PNDP“ stammen die Operanden von einer S7-314PN/DP, beim Device „1214“ von einer S7-1200 CPU 1214. Das IM-Device ist ja bekanntlich das interne Device von PLC-Lab.

Das Device eines Operanden bzw. Symbols kann verändert werden. Voraussetzung ist, dass der Operand im neuen Device erlaubt ist und das Symbol bzw. der Operand im neuen Device noch nicht definiert sind. Dieses **Umverdrahten** kann von Hand erfolgen oder auch über die Funktion „Umverdrahten von Operanden auf ein anderes Device“. Auch dies wird im weiteren Verlauf noch explizit gezeigt.

4.1.2 Symbol

Das Symbol ist die symbolische Bezeichnung des Operanden. Im Idealfall beschreibt das Symbol die Aufgabe bzw. die Funktion des Operanden innerhalb des Anlagenprojekts.

Dafür stehen pro Symbol 80 Zeichen zur Verfügung. Erlaubt sind a–z, A–Z, Umlaute, der Unterstrich sowie die Zahlen 0-9. Das Symbol muss mit einem Buchstaben beginnen.

Das Symbol muss innerhalb des Device eindeutig sein. Groß- und Kleinschreibung wird dabei nicht unterschieden.

4.1.3 Operand

Der Operand legt die Adresse fest, die dem Symbol zugeordnet ist. Welche Operanden (Adressen) zur Verfügung stehen, ist vom Device abhängig.

4.1.4 Datentyp

Die Liste der **Datentypen** wird aufgebaut, sobald der Operand angegeben wurde. Der Datentyp entscheidet beispielsweise bei digitalen Operanden (Byte-, Wort-, Doppelwort-Operanden) darüber, ob sie auch negative Zahlen beinhalten. Werden negative Zahlen benötigt, dann muss ein **bipolarer Datentyp** selektiert werden (z. B. Int16).

Beinhaltet der digitale Operand analoge Werte im S7-Umfeld, dann sollte der Datentyp „S7Analog“ eingestellt werden. Dieser hat dann den Wertebereich -27648 bis +27648, was dem analogen Wertebereich einer Simatic-Steuerung von Siemens entspricht. Dies ist insbesondere dann von Bedeutung, wenn PLC-Lab bei der Auswertung oder dem Beschreiben von Operanden auf den Wertebereich skaliert.

4.1.5 Kommentar

Hier kann zusätzlich ein Kommentar für das Symbol bzw. dem Operanden angegeben werden.

4.2 Handhabung der Symboltabelle

4.2.1 Benennung der Elemente der Symboltabelle

Im nachfolgenden Bild sind die Bedienelemente der Symboltabelle gekennzeichnet. Diese sollen anschließend benannt werden.

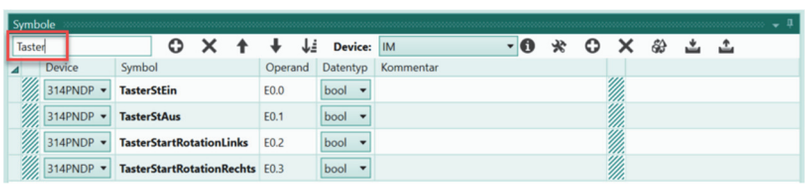
Bild: Elemente der Symboltabelle

Element	Beschreibung
1	**Filter für die anzuzeigenden Symbole in der Tabelle:** Mit Hilfe des Filters können nur bestimmte, dem angegebenen Filter entsprechende Symbole oder Operanden angezeigt werden. Dies ist insbesondere dann hilfreich, wenn man nach bestimmten Symbolen sucht.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Beispiel:

Es sollen nur die Symbole angezeigt werden, deren Symbol den Inhalt „Taster“ haben.



Die im Filter angegebenen Zeichen werden innerhalb des Device-Namens, des Symbols und des Operanden gesucht. Wenn ein Filter aktiv ist, dann können keine neuen Operanden der Tabelle hinzugefügt werden. Auch das Löschen ist nicht möglich.

2 **Neues Symbol einfügen:** Bei Betätigung des linken Plus-Symbols wird ein neues Symbol für das im Element 6 selektierte Device hinzugefügt.

Gehört die momentan in der Symboltabelle aktive Zeile ebenfalls zu dem selektierten Device, dann wird der Operand aus dieser Zeile mit einer „logisch erhöhten“ Adresse eingefügt: Bei Bit-Operanden wird somit die Bit-Adresse inkrementiert, bei Byte-Operanden die Byte-Adresse. Handelt es sich um Wort- oder Doppelwort-Operanden, dann wird die Adresse auf die nächste Wort- bzw. Doppelwort-Adresse gesetzt. Sind die bei der Erhöhung ermittelten Adressen schon vorhanden, dann wird bis zur nächsten freien Adresse inkrementiert.

Ist ein anderes Device aktiv, dann wird ein Standard-Operand des Device eingefügt mit einer noch nicht belegten Adresse. Es ist unerheblich, welche Syntaxsprache für die Operanden verwendet wird. Ändert man einen Operanden aktiv auf eine bestimmte Syntax, dann wird diese im weiteren Verlauf verwendet.

3 **Symbol löschen:** Die momentan selektierten Zeilen in der Tabelle werden gelöscht.

Ist das Symbol in der Eigenschaft eines Objekts vorhanden, dann wird das Symbol beseitigt und durch den Absolut-Operanden ersetzt.

4 **Symbole verschieben:** Die momentan selektierten Zeilen in der Tabelle werden nach oben oder unten verschoben.

Für einzelne Zeilen ist dies auch per Drag-and-Drop möglich. Dabei sind die schraffierten Flächen in den Zeilen zu verwenden.

5 **Tabelle sortieren:** Bei Betätigung werden alle Operanden innerhalb der Tabelle anhand ihrer Adressen aufsteigend sortiert. Die Sortierung erfolgt Device-intern.

Die Aktion kann nicht rückgängig gemacht werden.

6 **Aktives Device:** Alle im Projekt vorhandenen Devices sind in dieser Liste vorhanden. Das in der Liste selektierte Device wird für das Einfügen der Symbole in der Tabelle verwendet, d. h. die eingefügten Operanden sind diesem Device zugeordnet. Die rechts neben der Liste befindlichen Elemente 7, 8, 10 beziehen sich ebenfalls auf das selektierte Device.

7 **Info zu Device:**

Wird die Maus über dieses Element bewegt, dann erscheint eine Info zu dem momentan aktiven Device als Tool-Tipp.

8 **Einstellungen des aktiven Device:**

Können beim momentan aktiven Device Einstellungen getätigt werden, dann wird der entsprechende Dialog bei Betätigung angezeigt.

9 **Neues Device erzeugen:** Unterstützt die Edition von PLC-Lab unterschiedliche Devices, dann erscheint ein Dialog, um ein neues Device zu erzeugen. Auf diesem ist zunächst der neue Device-Typ zu selektieren. Mögliche Device-Typen sind „S7-300“, „S7-1500“, „PLCSIM Advanced“ usw.

Wurde der Typ selektiert und eine in diesem Projekt noch nicht vorhandene Bezeichnung für das Device angegeben, dann kann man den Dialog bestätigen. Der nachfolgende Dialog zeigt die Device-Typ-spezifischen Einstellungen.

Nach Abschluss wird das neue Device in der Liste der vorhandenen Devices eingefügt (Element 6) und als aktives Device selektiert.

Hinweis:

Die Simulations-Devices IM, Sim, PLCSim und PLCSim300 können **nicht** erzeugt werden. Diese sind in den jeweiligen Editionen in jedem Projekt **vorhanden**.

10 **Device löschen:** Das momentan im Element 6 selektierte Device wird auf Nachfrage gelöscht, sofern kein Operand des Device in der Eigenschaft eines Objekts verwendet wird. Ist dies der Fall, dann folgt ein Hinweis und das Objekt mit dem Operanden wird auf der Zeichenfläche hervorgehoben.

Das interne Device „IM“ sowie die standardmäßig vorhandenen Simulations-Devices können nicht gelöscht werden.

11 **Umverdrahten von Operanden auf ein anderes Device:** Mit dieser Funktion können Operanden eines Device auf ein anderes Device umverdrahtet werden, sofern die Operanden im Ziel-Device verfügbar sind. Es erscheint ein Dialog, auf dem das Quell-Device und das Ziel-Device auswählbar sind.

Diese Funktion wird im weiteren Verlauf noch ausführlich erläutert.

12 **Import von Symbolen für eingestelltes Device:** Mit Hilfe dieser Funktion können Symbole aus einer Datei importiert und dem momentan aktiven Device zugewiesen werden. Die Datei muss zuvor als Export vom TIA-Portal, Simatic-Manager, WinSPS-S7 oder GRAFCET-Studio erzeugt worden sein. Als Formate sind .XML (TIA, GRAFCET-Studio), .CSV (WinSPS-S7, GRAFCET-Studio) oder .ASC (Simatic-Manager, GRAFCET-Studio) möglich.

13 **Export von Symbolen für eingestelltes Device:** Bei Betätigung erscheint ein Dialog, auf dem die exportierenden Operanden-Arten des aktuellen Devices selektierbar sind. Als Ziel-Datei kann eine Datei mit der Endung .XML oder .ASC erzeugt werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

4.3 Verwendung von Symbolen in den Objekten

Grundsätzlich können die Operanden (Adressen) in den Eigenschaften der Objekte **absolut** angegeben werden. Dabei wird dann der Name des Device gefolgt von einem Punkt und dann die Absolutadresse des Operanden eingetragen.

Beispiel: „PLCSim300.E12.4“

Dies ist allerdings nicht die empfohlene Vorgehensweise. Man sollte einen Operanden vor seiner Verwendung in der Symboltabelle angeben und dabei einen symbolischen Namen für den Operanden vergeben. Dieser symbolische Name beschreibt am besten die Funktion des Operanden. Als Beispiel sei genannt „TasterSteuerungEin“ für den Operanden, welcher vom Taster „Steuerung Ein“ umgeschaltet wird. Man erkennt dann auf einen Blick, ob der korrekte Operand in der Eigenschaft des entsprechenden Taster-Objekts angegeben ist. Des Weiteren wird eine Mehrfachverwendung vermieden, ein häufiger **Fehler** bei der Benutzung von absoluten Operanden.

Sind alle verwendeten Operanden in der Symboltabelle eingetragen, dann hat man eine zentrale Verwaltungsstelle. Hier kann das **Umverdrahten** auf ein anderes Device oder das Ändern der absoluten Adresse eines Symbols vorgenommen werden.

Die in der Symboltabelle vorhandenen Operanden müssen bei den Eigenschaften der Objekte nicht mehr von Hand eingefügt werden. Dabei stehen mehrere Möglichkeiten zur Auswahl. Diese werden nachfolgend beschrieben.

4.3.1 Per Drag-and-Drop den Operanden in die Objekt-Eigenschaft eintragen

Ein Operand kann direkt aus der Symbol-Tabelle per Drag-and-Drop in die Eigenschaft eines Objekts eingetragen werden. Startpunkt ist dabei eine der beiden schraffierten Flächen in jeder Zeile eines Symboleintrags in der Tabelle. Zielpunkt ist der **Editor** der Eigenschaft oder ihr **Beschreibungstext**.

Beispiel:

Im Beispiel ist ein Symbol mit der Bezeichnung „TasterStEin“ vorhanden. Es handelt sich dabei um das Symbol des Eingangs E0.0, welcher dem Device „314PNDP“ zugeordnet ist. Der Operand ist über einen Schalter zu beeinflussen, welcher als Taster konfiguriert ist. Im folgenden Bild ist dies zu sehen:

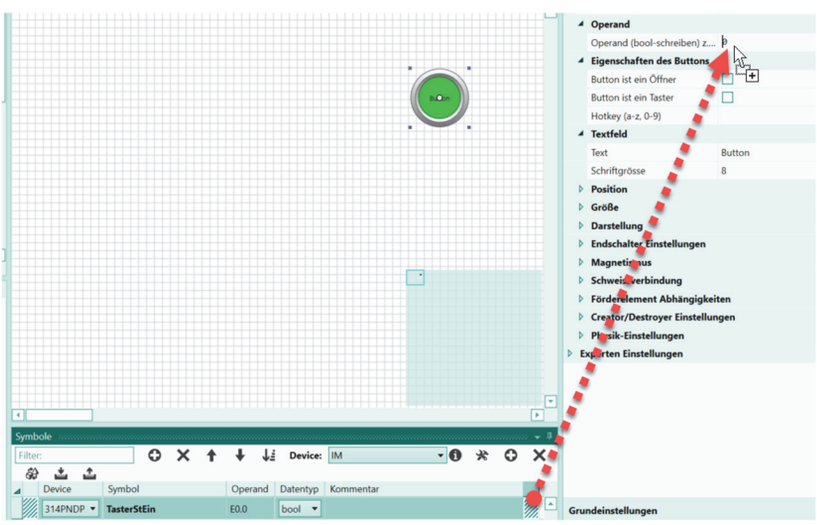


Bild: Eintragen des Operanden per Drag-and-Drop

Nach dieser Aktion ist der Operand mit dem Device in der Eigenschaft eingetragen.

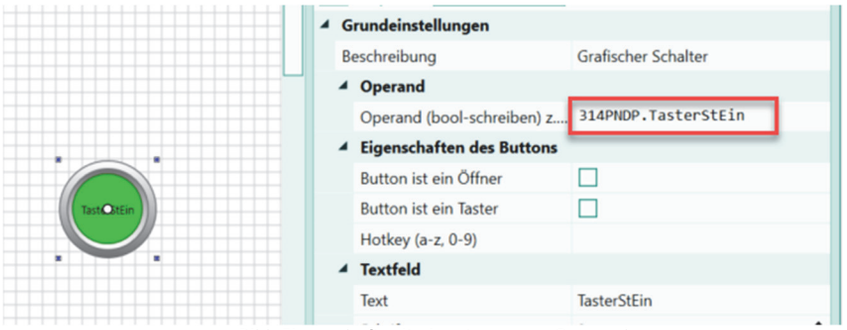


Bild: Eigenschaft mit eingetragenem Operanden

Zusätzlich wurde das Symbol als Text im Schalter-Objekt übernommen.

Auf diese Weise können alle in der Symbol-Tabelle vorhandenen Operanden mit den gewünschten Eigenschaften der Objekte verknüpft werden. Dabei fällt keine Schreibarbeit an, denn es wird sowohl der Operand als auch das Device automatisch übernommen.

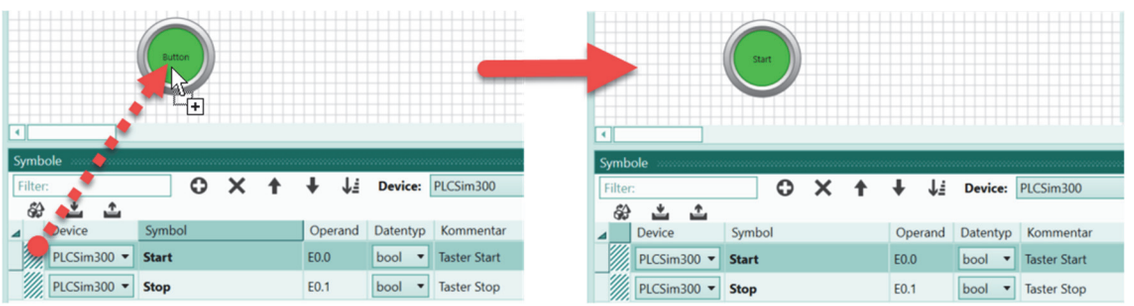
4.3.2 Per Drag-and-Drop mit dem Objekt (z. B. Schalter) als Ziel

Um das Eintragen eines Operanden in die Eigenschaft eines Objekts nochmals zu beschleunigen, wurde bei einigen Objekten eine Sonderfunktion ermöglicht. Hier kann das Objekt selbst das Ziel der Drag-and-Drop-Aktion sein. Der Operand wird dann automatisch in die „Haupteigenschaft“ des Objekts eingetragen. Bei einem Schalter-Objekt bedeutet dies, dass das Objekt in der Eigenschaft eingefügt wird, welche vom Schalter beeinflusst wird.

Beispiel:

Der Operand E10.0 mit dem Symbol „Start“ ist mit einem Schalter-Objekt zu verknüpfen. Dabei wird die Zeile des Operanden in der Symbol-Tabelle als Ausgangspunkt verwendet. Ziel ist das Schalter-Objekt, das nicht selektiert sein muss.

Bild: Per Drag-and-Drop mit dem Objekt als Ziel



Linkes Bild: Das Symbol „Start“ wird über den Drag-and-Drop-Vorgang auf den Schalter gezogen. In der rechten Darstellung ist zu sehen, dass im Schalter-Objekt das Symbol des Operanden als Beschriftung übernommen wurde. Wenn man den Schalter selbst selektiert, dann erkennt man, dass in der Eigenschaft „Operand“ der eingefügte Operand mit seinem Symbol eingefügt wurde.

Während der Aktion wird in der Statuszeile von PLC-Lab angezeigt, welcher Eigenschaft der Operand zugewiesen wurde.

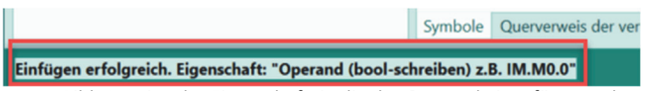


Bild: Anzeige der Eigenschaft, in die der Operand eingefügt wurde

© MHJ-Software GmbH & Co. KG - PLC-Lab

Beispiel:

Nun ein weiteres Beispiel, diesmal mit einem **Text-Label-Objekt**. Ein solches Objekt kann für die Anzeige von Text, aber auch zur Anzeige von Statuswerten verwendet werden. Das Text-Label-Objekt ist ebenfalls als direktes Ziel der Drag-and-Drop-Aktion geeignet.

Im Beispiel soll der Wert des Symbols „SensorTempWert1“ im Text-Label-Objekt angezeigt werden.

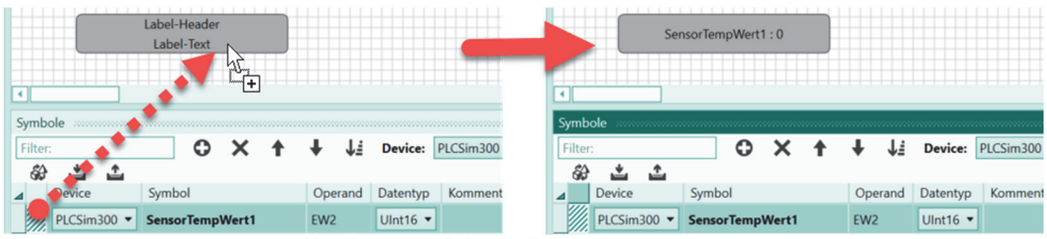


Bild: Text-Label-Objekt als Ziel der Drag-and-Drop-Aktion

Links wird das Symbol per Drag-and-Drop auf das Text-Label gezogen. Als Folge wird der Operand in der Eigenschaft „Operand“ übernommen und als Anzeigetext **„SensorTempWert1: {0}“** in die Eigenschaft „Text für Überschrift“ eingetragen. Damit wird beim Start der Simulation der Inhalt des Operanden im Label angezeigt.

4.3.2.1 Welche Objekte sind als direktes Drag-and-Drop-Ziel geeignet?

Diese Objekte sind als Ziel einer direkten **Drag-and-Drop**-Aktion **geeignet**:

- Schalter-Objekte: Der Operand wird als Schalter-Operand übernommen und das Symbol als Text des Schalters.
- Lampen-Objekte: Der Operand wird als Lampen-Operand übernommen und das Symbol als Text der Lampe.
- Endschalter-Objekt: Der Operand wird als Endschalter-Operand übernommen.
- Geometrie-Objekte: Der Operand wird als Endschalter-Operand übernommen.
- Rohrleitungen: Der Operand wird als Operand für Farbveränderungen übernommen.
- Creator-Objekte: Der Operand wird als Create-Trigger-Operand übernommen.
- Text-Label: Der Operand wird als Operand übernommen und als Text dessen Symbol gefolgt vom Wert des Operanden angegeben.
- Tacho-Anzeige: Der Operand wird als Operand für den momentanen Wert übernommen.
- Digital-Anzeige: Der Operand wird als Operand für den momentanen Wert übernommen.
- Schieberegler: Der Operand wird als Operanden für den momentanen Wert übernommen.
- Digitaler Ziffernsteller: Der Operand wird als Operand für den momentanen Wert übernommen.

Hinweis: Ist ein Objekt als Ziel geeignet, dann verändert der Mauszeiger sein Symbol.

Er hat dann folgendes Aussehen:

4.3.3 Mit Hilfe von Intellisense (Autovervollständigung)

Sollte es trotz der Möglichkeiten, per Drag-and-Drop einen Operanden in eine Eigenschaft einzutragen, notwendig sein, ihn von Hand einzutippen, dann kommt die Intellisense-Funktion zu Hilfe.

Dabei kann nach der Eingabe des Device-Namens gefolgt vom Punkt-Operator die Tastenkombination [STRG] + [Leerzeichen] betätigt werden. Als Folge erscheint eine Auswahl aller Symbole des Device. Durch Angabe eines Anfangsbuchstabens kann man die Liste verkürzen.

Hat man das gewünschte Symbol gefunden, dann wird es per Maus oder Tastatur in der Liste selektiert und eingefügt.

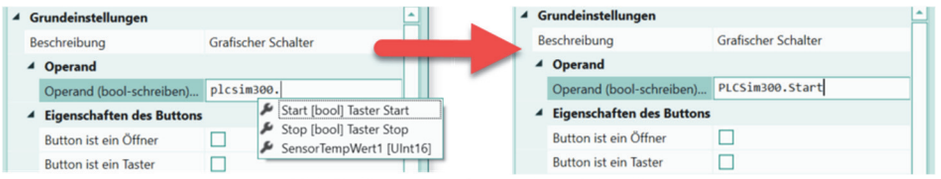


Bild: Einfügen eines Operanden mit Intellisense

4.4 Symbole umverdrahten

4.4.1 Umverdrahten über die Funktion „Umverdrahten“

Häufig kommt es vor, dass man ein Anlagenprojekt für ein bestimmtes Device erstellt hat, nun aber auch für ein anderes Device benötigt. Beispielsweise hat man die Anlage mit Operanden für PLCSim300 erstellt und möchte nun die Anlage zum Test an einer realen S7-1500 verwenden.

Hier kommt die Funktion „Umverdrahten“ zur Anwendung. Die Vorgehensweise ist wie folgt:

- Speichern des Anlagenprojekts unter einem anderen Namen. Dies ist nur notwendig, wenn die bisherige Anlage weiterhin mit dem alten Device verwendet werden soll.
- Erzeugen des neuen Device.
- Ausführen der Funktion „Umverdrahten von Operanden auf ein anderes Device“ in der Symbol-Tabelle.
- Es erscheint der Dialog „Operanden auf ein anderes Device umverdrahten“. In der oberen Liste wird das Quell-Device ausgewählt, darunter das Ziel-Device. Der Button „OK“ startet die Aktion.

Anschließend sind alle Operanden des alten Devices auf das neue Device umverdrahtet worden, sofern dies möglich war. Sollte auch nur ein Operand im neuen Device nicht vorhanden sein, dann wird die Aktion nicht ausgeführt und ein entsprechender Fehler ausgegeben. Die Aktion kann rückgängig gemacht werden, indem man sie nochmals ausführt und die beiden Devices vertauscht angibt.

Hinweis: Das Umverdrahten wird **nicht nur** bei den Operanden innerhalb der Symboltabelle durchgeführt, sondern auch bei den absolut adressierten Operanden in den Eigenschaften der Objekte.

© MHJ-Software GmbH & Co. KG - PLC-Lab

4.4.2 Manuelles Umverdrahten

Sind nur einige wenige Operanden auf ein anderes Device umzuverdrahten und befinden sich alle diese Operanden in der Symbol-Tabelle, dann kann man dies auch sehr schnell per Hand erledigen.

Dazu selektiert man die gewünschten Zeilen innerhalb der Symbol-Tabelle und verändert die Device-Einstellung in einer der selektierten Zeilen. Als Folge sind nun alle selektierten Operanden dem neuen Device zugeordnet.

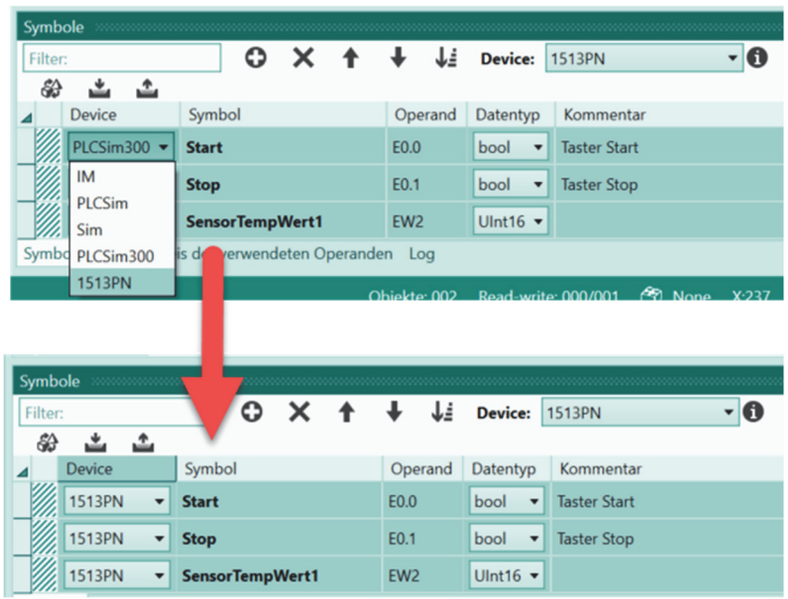


Bild: Verändern des Device der selektierten Operanden

Hinweis: Die Aktion kann nur ausgeführt werden, wenn alle Operanden des Quell-Devices auch im Ziel-Device vorhanden sind und noch keine Symbole für diese Operanden im Ziel-Device definiert wurden.

4.5 Import und Export von Symbolen

Wurde ein Projekt in einer anderen Software als PLC-Lab begonnen, z. B. mit der Erstellung des SPS-Programms im TIA-Portal, dann sind dort Symbole bereits definiert worden. Diese Symbole können mit Hilfe des Imports von Symbolen in **PLC-Lab übernommen** und genutzt werden.

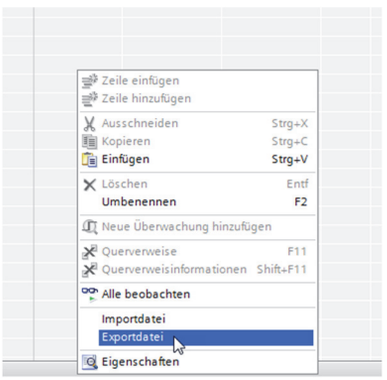
Auch der umgekehrte Fall ist möglich: Wurden für das Anlagenprojekt bereits Symbole in PLC-Lab definiert, dann können diese zur weiteren Verwendung **exportiert** werden.

4.5.1 Import von Symbolen aus dem TIA-Portal von Siemens

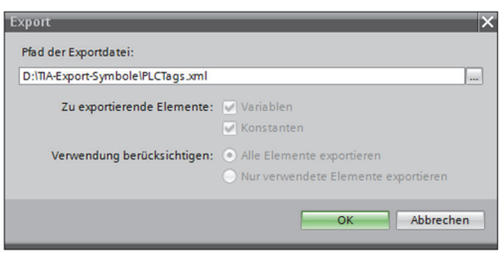
Möchte man in PLC-Lab Symbole aus dem TIA-Portal importieren, dann müssen diese zuvor vom TIA-Portal im XML-Format exportiert werden. Die so erzeugte XML-Datei wird dann in PLC-Lab für den Import ausgewählt.

Vorgehensweise:

1. Öffnen des TIA-Projekts und dann der **PLC-Variablen**.
2. Aufruf des Kontextmenüs innerhalb der PLC-Variablen durch Betätigung der rechten Maustaste. Selektion des Menüpunkts „**Exportdatei**“.



3. Anschließend ist der Dialog „Export“ zu sehen, auf dem die Export-Datei und der Ablageort zu selektieren sind. Es ist darauf zu achten, dass die Datei die **Endung „.XML“** besitzt.

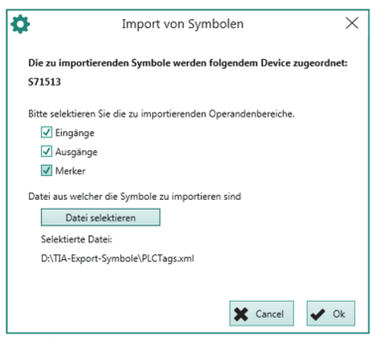


Durch Betätigung des „OK“-Buttons wird die Aktion ausgelöst und die Variablen in die XML-Datei exportiert.

4. Nun wechselt man zu PLC-Lab und öffnet das gewünschte Anlagenprojekt. Anschließend wird in der Symboltabelle das Device selektiert, welchem die zu importierenden Symbole zugeordnet werden sollen. Existiert das Device noch nicht, dann ist dieses zunächst zu erzeugen.
   Sind diese Voraussetzungen erfüllt, dann betätigt man die Schaltfläche „Import von Symbolen für eingestelltes Device“.

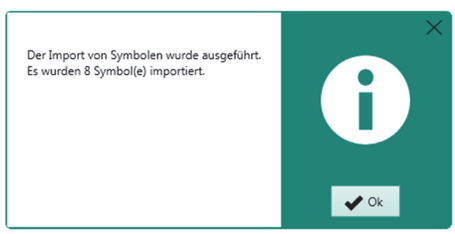
© MHJ-Software GmbH & Co. KG - PLC-Lab

5. Jetzt ist der Dialog „Import von Symbolen“ zu sehen. Hier können die zu importierenden Operanden-Bereiche selektiert werden. Es stehen nur die Operanden-Bereiche zur Auswahl, welche das Ziel-Device unterstützt. Im Beispiel ist das Ziel-Device ein reale S7-1500.

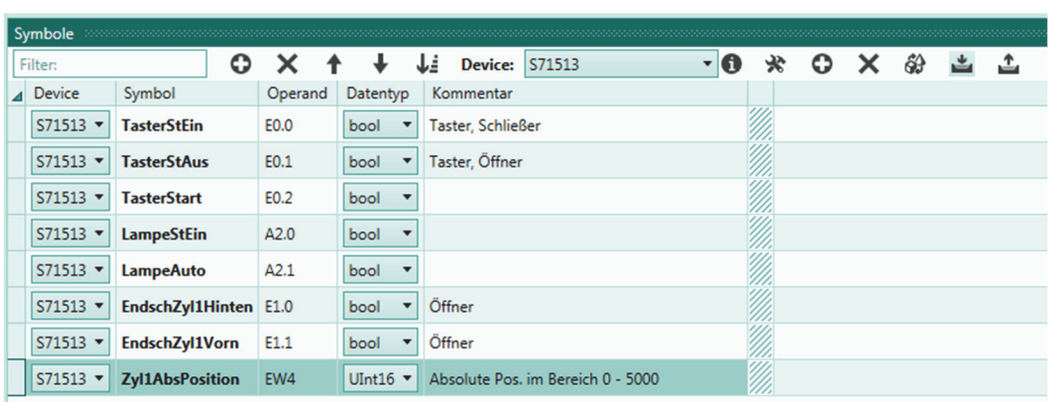


Über den Button „Datei selektieren“ ist die zuvor vom TIA erzeugte XML-Datei auszuwählen. Danach startet man die Aktion über die „OK“-Schaltfläche.

6. Nach Beendigung der Aktion erscheint eine Erfolgsmeldung. Etwaige Fehler beim Import sind in einer Protokolldatei verzeichnet, welche bei Bedarf ebenfalls aufgerufen wird.



Damit ist die Aktion abgeschlossen. Die importierten Symbole befinden sich in der Symbol-Tabelle und wurden dem zuvor eingestellten Device zugeordnet.



Device	Symbol	Operand	Datentyp	Kommentar
S71513	TasterStEin	E0.0	bool	Taster, Schließer
S71513	TasterStAus	E0.1	bool	Taster, Öffner
S71513	TasterStart	E0.2	bool	
S71513	LampeStEin	A2.0	bool	
S71513	LampeAuto	A2.1	bool	
S71513	EndschZyl1Hinten	E1.0	bool	Öffner
S71513	EndschZyl1Vorn	E1.1	bool	Öffner
S71513	Zyl1AbsPosition	EW4	UInt16	Absolute Pos. im Bereich 0 - 5000

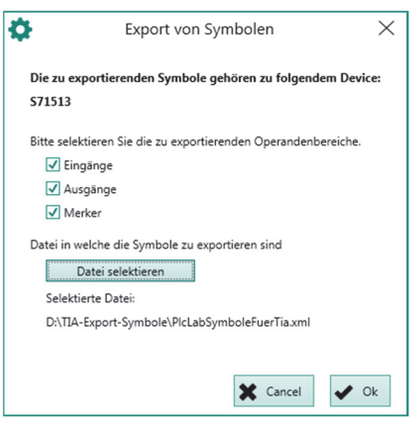
Bild: Symbol-Tabelle mit den importierten Symbolen

4.5.2 Export von Symbolen für das TIA-Portal von Siemens

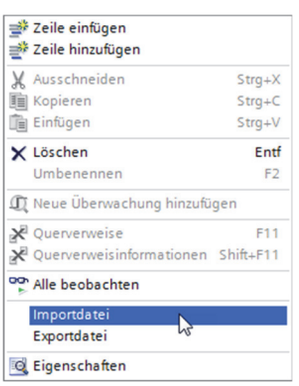
Auch beim **Export** der Symbole aus PLC-Lab und dem Import im TIA-Portal wird das **XML-Format** verwendet. Nachfolgend die Schritte für den Export der Symbole aus PLC-Lab in eine XML-Datei:

1. Zunächst öffnet man das PLC-Lab Projekt und selektiert das Device innerhalb der Symbol-Tabelle, von welchem die Symbole zu exportieren sind. Danach wird die Schaltfläche „Export von Symbolen für eingestelltes Device“ betätigt.

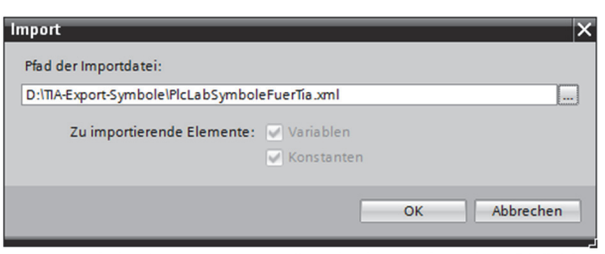
2. Es erscheint der Dialog „Export von Symbolen“, auf dem die zu exportierenden Operanden-Bereiche auszuwählen sind. Des Weiteren ist über den Button „Datei selektieren“ die Datei und deren Ablageort für die zu exportierenden Symbole anzugeben. Dabei ist zu beachten, dass als **Endung .XML** angegeben wird. Dann wird eine XML-Datei erzeugt, welche vom TIA importiert werden kann.



3. Der Button „OK“ auf dem Dialog startet die Aktion. Nach deren Abschluss wird über ein Meldefenster angezeigt, wie viele Symbole exportiert wurden.
4. Im nächsten Schritt wird das TIA-Portal mit dem Projekt geöffnet. Anschließend ist die Tabelle mit den PLC-Variablen aufzurufen und über die rechte Maustaste wird das Kontextmenü zur Ansicht gebracht. Dabei ist der Menüpunkt „Importdatei“ zu selektieren.



5. Auf dem erscheinenden Dialog „Import“ wählt man die zuvor von PLC-Lab erzeugte XML-Datei aus und startet den Import.



Damit ist die Aktion abgeschlossen und die importierten Symbole sind in den PLC-Variablen der TIA-Station vorhanden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

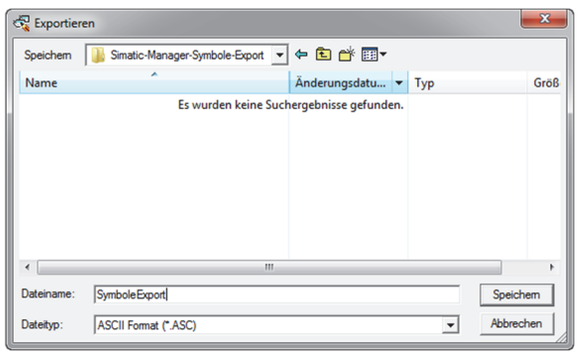
Hinweis: Die oben gezeigten Schritte für den Import und Export von Symbolen innerhalb des TIA-Portals können **geräteübergreifend** (S7-300/400, S7-1200/1500) zur Anwendung kommen.

4.5.3 Import von Symbolen aus dem Simatic-Manager von Siemens

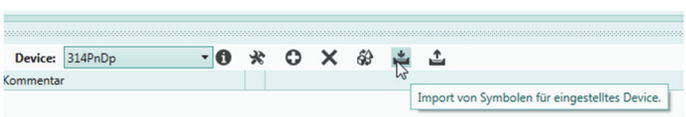
Der Simatic-Manager von Siemens kann die Symbole eines Projekts in eine sog. ASC-Datei exportieren. PLC-Lab kann diese Datei öffnen und die darin enthaltenen Symbole importieren.

Dabei geht man wie folgt vor:

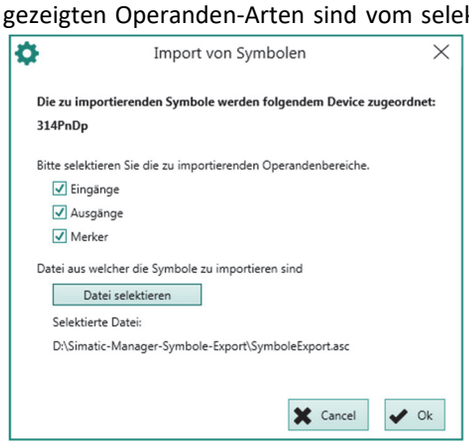
1. Im Simatic-Manager wird das gewünschte SPS-Projekt geöffnet und der Symbolik-Editor aufgerufen. Anschließend führt man im Editor den Menüpunkt „Tabelle->Exportieren“ aus.
2. Daraufhin ist der Dialog „Exportieren“ zu sehen, mit dem die Export-Datei und deren Ablageort ausgewählt wird. Als Dateityp ist **„ASCII Format (ASC)“** einzustellen.



3. Wurde die Datei erzeugt, dann wechselt man zu PLC-Lab und öffnet das Anlagenprojekt. In der Symbol-Tabelle wird das Device selektiert, welchem die zu importierenden Symbole zugeordnet werden sollen.

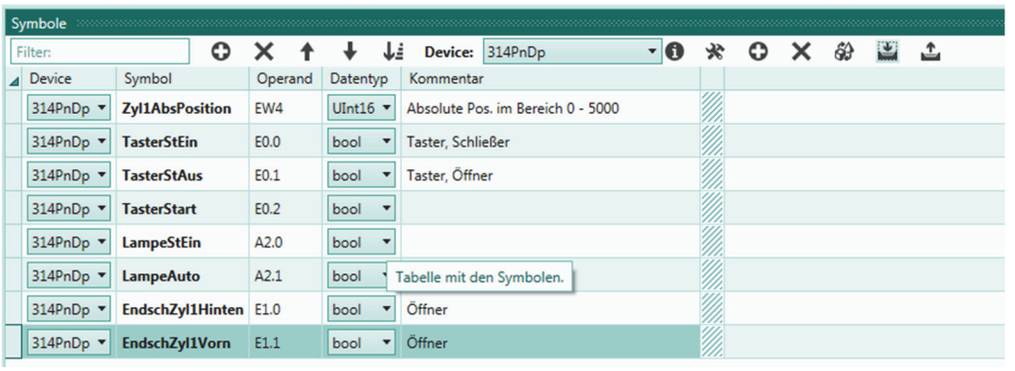


   Die Schaltfläche „Import von Symbolen für eingestelltes Device“ startet den Import.
4. Es erscheint der Dialog „Import von Symbolen“, über den die zu importierende Datei zu selektieren ist. Des Weiteren können die Operanden-Arten ausgewählt werden, die beim Import zu berücksichtigen sind. Die angezeigten Operanden-Arten sind vom selektierten Device-Typ abhängig. Der Button „OK“ startet die Aktion.



5. Nach Durchführung der Aktion erscheint eine Meldung, welche darüber informiert, wie viele Symbole importiert wurden. Etwaige Fehler beim Import werden in einer Protokolldatei angezeigt.

6. Die importierten Symbole sind daraufhin in der Symbol-Tabelle vorhanden und dem Device zugeordnet.



4.5.4 Export von Symbolen für den Simatic-Manager von Siemens

Sollen die Symbole von PLC-Lab für den Simatic-Manager von Siemens **exportiert** werden, dann ist das ASC-Format für die Datei auszuwählen.

Vorgehensweise:

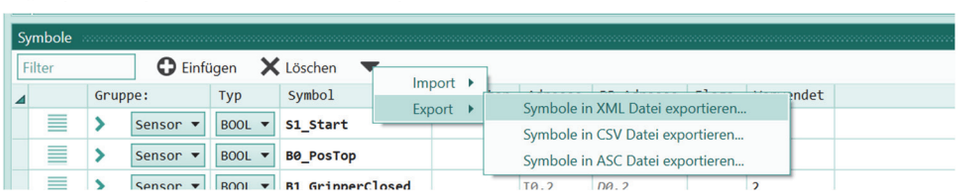
1. In PLC-Lab wird in der Symboltabelle das Device selektiert, dessen Symbole man exportieren möchte. Anschließend wird die Schaltfläche „Export von Symbolen für eingestelltes Device“ betätigt.
2. Auf dem nun erscheinenden Dialog „Export von Symbolen“ können die zu exportierenden Operanden-Bereiche und die Export-Datei eingestellt werden. Als Dateiendung ist **„.ASC“** zu selektieren. Der Button „OK“ startet den Export.
3. Im nächsten Schritt ist der Simatic-Manager zu starten, das gewünschte Projekt und der Symbolik-Editor zu öffnen. Im Symbolik-Editor ist der Menüpunkt „Tabelle importieren“ auszuführen. Auf dem erscheinenden Dialog wird die zuvor erzeugte ASC-Datei ausgewählt. Über die Schaltfläche „OK“ wird der Import ausgelöst.
4. Nach Abschluss erscheint ein Meldefenster, welches das Import-Ergebnis anzeigt.

4.5.5 Import von Symbolen aus GRAFCET-Studio

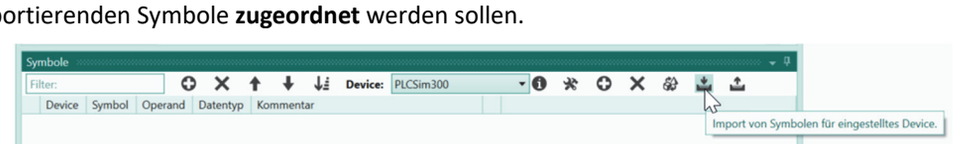
GRAFCET-Studio kann den Inhalt der Symbol-Tabelle unter anderem in eine XML-Datei exportieren. Diese Datei wird dann in PLC-Lab importiert. Damit stehen die Symbole im Anlagenprojekt von PLC-Lab zur Verfügung.

Dazu geht man wie folgt vor:

1. Starten von GRAFCET-Studio und öffnen des gewünschten Projekts. In der Symbol-Tabelle wird der Menüpunkt „Export->Symbole in XML-Datei exportieren“ selektiert.



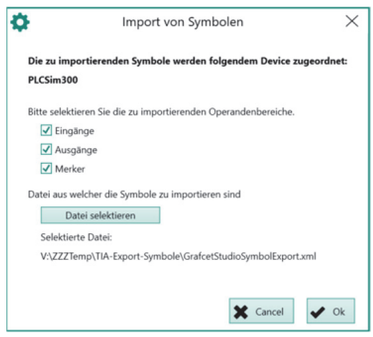
2. Im erscheinenden Datei-Auswahl-Dialog werden der Ablageort und der Dateiname der zu erzeugenden XML-Datei angegeben. Die Bestätigung des Dialogs startet die Aktion und die XML-Datei wird mit den Symbolen gefüllt.
3. In PLC-Lab öffnet man das Anlagenprojekt und selektiert in der Symbol-Tabelle das **Device**, welchem die zu importierenden Symbole **zugeordnet** werden sollen.



Anschließend ist die Schaltfläche „Import von Symbolen für eingestelltes Device“ zu betätigen.

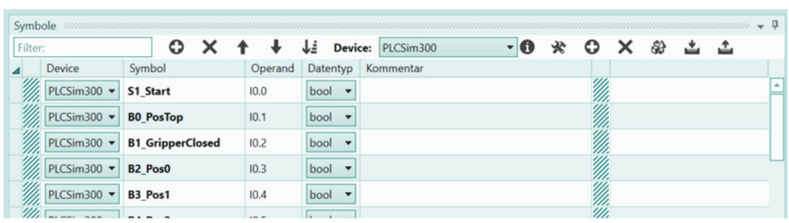
© MHJ-Software GmbH & Co. KG - PLC-Lab

4. Daraufhin ist der Dialog „Import von Symbolen“ sichtbar. Auf diesem sind die zu importierenden Operanden-Bereiche sowie die zuvor erzeugte XML-Datei auszuwählen. Die Operanden-Bereiche sind vom Device abhängig, welchem die Symbole zugeordnet werden.



Der Button „OK“ startet den Import.

5. Nach der Aktion wird ein Meldefenster angezeigt. Es gibt Auskunft darüber, wie viele Symbole importiert wurden. Bei etwaigen Fehlern ist zusätzlich eine Protokolldatei zu sehen.
6. In der Symbol-Tabelle sind die importierten Symbole vorhanden, zugeordnet zum zuvor selektierten Device.



4.5.6 Export von Symbolen für GRAFCET-Studio

GRAFCET-Studio kann Symbole unter anderem aus einer XML-Datei importieren. Möchte man Symbole aus PLC-Lab exportieren, dann erzeugt man eine XML-Datei beim Export. Dazu geht man wie folgt vor:

1. Starten von PLC-Lab und Öffnen des Anlagenprojekts. In der Symbol-Tabelle wird das Device selektiert, dessen Symbole zu exportieren sind. Danach wird in der Symbol-Tabelle die Schaltfläche „Export von Symbolen für eingestelltes Device“ selektiert.
2. Es erscheint der Dialog „Export von Symbolen“, auf dem die zu exportierenden Operanden-Arten und die Datei für den Export auszuwählen sind. Die Datei muss die Endung **„.XML“** besitzen. Der Button „OK“ auf dem Dialog startet die Aktion.
3. Nun wechselt man zu GRAFCET-Studio und öffnet das gewünschte Projekt. In der Symbol-Tabelle wird der Menüpunkt „Import->Symbole aus XML-Datei importieren“ ausgewählt.

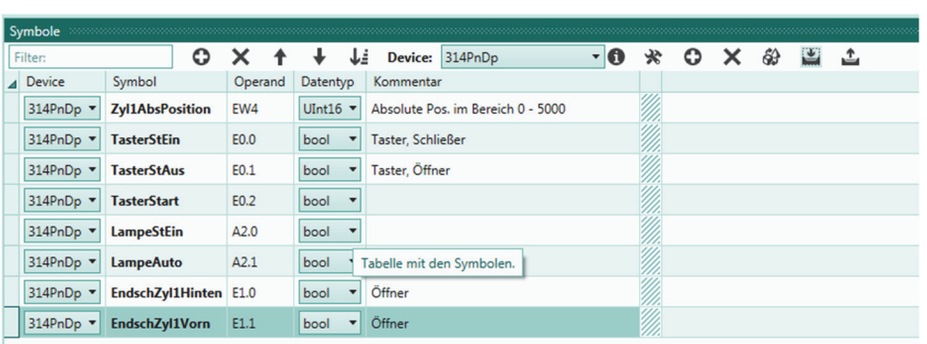
4. Über den Datei-Auswahl-Dialog wird die zuvor von PLC-Lab erzeugte XML-Datei selektiert und die Aktion gestartet.
5. Nun stehen die Symbole im Projekt von GRAFCET-Studio zur Verfügung.

4.5.7 Import von Symbolen aus WinSPS-S7

Wurde ein Projekt in WinSPS-S7 begonnen und möchte man die Symbole daraus übernehmen, dann wird das ASC-Format für den Datenaustausch verwendet. Dabei geht man folgendermaßen vor:

1. Man startet WinSPS-S7 V6, öffnet die Projektmappe und das Projekt. Anschließend wird der Menüpunkt „*Datei->Exportieren->Symbolikdatei als ASC/SDF-Datei exportieren*“ ausgewählt.
2. Es erscheint ein Dialog, auf dem das Format „ASC“ zu selektieren ist. Anschließend wird er bestätigt.
3. Auf dem nachfolgenden Dialog können der Dateiname und der Ablagepfad der ASC-Datei ausgewählt werden. „OK“ bestätigt auch diesen Dialog und startet den Export.
4. Nun wechselt man zu PLC-Lab, öffnet das Anlagenprojekt und selektiert in der Symbol-Tabelle das Device, welchem die zu importierenden Symbole zugeordnet werden sollen. Anschließend wird die Schaltfläche „Importieren von Symbolen für eingestelltes Device“ betätigt.

5. Als Folge ist der Dialog „Import von Symbolen“ zu sehen. Auf diesem können die Operanden-Arten selektiert werden, welche beim Import zu berücksichtigen sind. Des Weiteren kann über den Button „Datei selektieren“ die zuvor von WinSPS-S7 V6 erzeugte ASC-Datei ausgewählt werden. Danach startet man den Import über „OK“.
6. Ein Meldefenster gibt am Ende des Imports Auskunft darüber, wie viele Symbole übernommen wurden. Etwaige Fehler werden in einer Protokolldatei abgelegt und diese wird bei Bedarf angezeigt.
7. Die importierten Symbole befinden sich in der Symbol-Tabelle und sind dem selektierten Device zugeordnet.

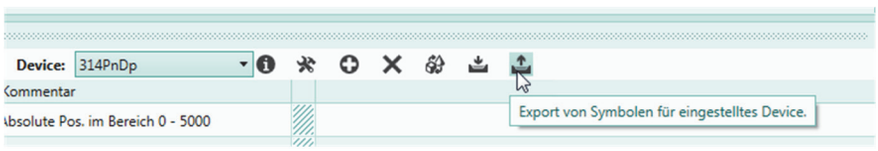


© MHJ-Software GmbH & Co. KG - PLC-Lab

4.5.8 Export von Symbolen für WinSPS-S7

Möchte man in WinSPS-S7 V6 Symbole importieren, so ist dies über das ASC-Format möglich. Somit sind die Symbole aus PLC-Lab in dieses Format zu exportieren. Die Vorgehensweise ist nachfolgend beschrieben.

1. Zunächst startet man PLC-Lab und öffnet das gewünschte Anlagenprojekt. In der Symbol-Tabelle wird das Device selektiert, von dem die Symbole in eine ASC-Datei zu exportieren sind. Danach wird die Schaltfläche „Export von Symbolen für eingestelltes Device“ betätigt.



2. Als Folge ist der Dialog „Export von Symbolen“ sichtbar. Auf diesem werden die Operanden-Arten ausgewählt, deren Symbole zu exportieren sind. Des Weiteren ist über den Button „Datei selektieren“ der Ablageort und der Name der Datei für den Export anzugeben. Wichtig dabei ist, dass die Datei die Endung „.ASC“ besitzt, damit dieses Export-Format verwendet wird.
   Der Dialog ist über „OK“ zu bestätigen, gleichzeitig startet dies die Aktion.
3. Wurde der Export ausgeführt, dann zeigt ein Meldefenster, wie viele Symbole in die Datei geschrieben wurden.
4. Im nächsten Schritt wird WinSPS-S7 V6 gestartet, die Projektmappe und das Projekt geöffnet. Anschließend führt man den Menüpunkt „Datei->Importieren->Symbolikdatei aus ASC/SDF-Datei importieren“ aus. Auf dem nun erscheinenden Datei-Auswahl-Dialog wird die zuvor von PLC-Lab erzeugte Datei ausgewählt und die Auswahl über „OK“ bestätigt.
5. Die Symbolikdatei ist nun über einen Doppelklick innerhalb des Fensters „Projektinhalt“ als die im Projekt aktive Symbolikdatei selektierbar.

Hinweis: In WinSPS-S7 V6 ist die max. Anzahl der Zeichen für Symbole auf 24 Zeichen begrenzt. Die Kommentare dürfen nicht länger als 80 Zeichen sein. Beim Export in ASC werden die Symbole und Kommentare entsprechend gekürzt.

5 Die Schalter-Objekte

Schalter-Objekte sind Objekte, die **Bit-Operanden** bei Betätigung beeinflussen. In PLC-Lab sind sie in unterschiedlichen Ausprägungen vorkonfiguriert. Vorkonfiguriert bedeutet, die Objekte sind beispielsweise in unterschiedlichen Farben im Katalog enthalten und können so direkt verwendet werden. Diese Farben sind nachträglich über die Eigenschaft „Füllfarbe“ veränderbar.

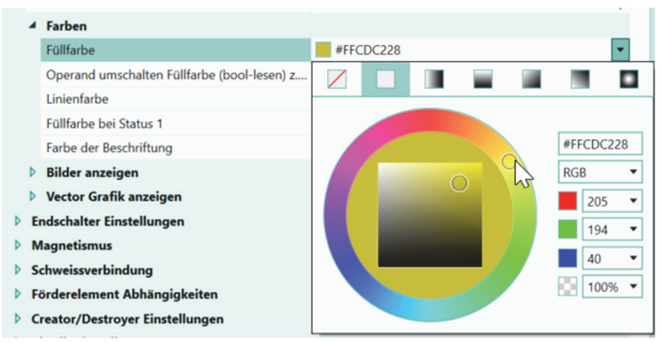


Bild: Verändern der Füllfarbe eines Schalter-Objekts

Alle Objekte innerhalb der Rubrik „Schalter“ stammen vom gleichen Objekt ab. Dies bedeutet, dass jedes Schalter-Objekt durch Veränderung seiner Eigenschaften umkonfiguriert werden kann.

5.1.1 Schalter, Taster, Schließer, Öffner

Alle Schalter-Objekte können als **Schalter** oder **Taster** eingestellt werden. Ebenso kann man selektieren, ob das Schalter-Objekt als **Schließer** oder **Öffner** ausgelegt sein soll.

Die dazu notwendigen Eigenschaften finden sich in der Rubrik „Eigenschaften des Buttons“.

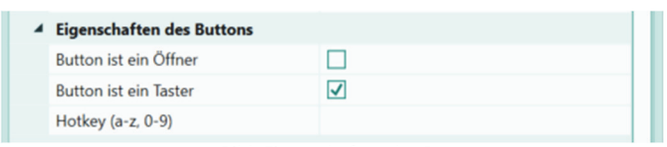


Bild: Eigenschaften des Buttons

In dieser Rubrik kann auch ein **Hotkey** für den Schalter angegeben werden. Dabei sind die Buchstaben a–z und die Zahlen 0-9 auf der Tastatur als Hotkeys erlaubt. Wird die Simulation gestartet und der Hotkey betätigt, dann wird auch das Schalter-Objekt betätigt. Auf diese Weise lassen sich beispielsweise Zweihand-Auslösungen simulieren, wenn man für zwei Schalter-Objekte je einen Hotkey angibt und sie dann zeitgleich betätigt.

5.1.2 Physik-Eigenschaften von Schalter-Objekten

In der Standard-Einstellung sind Schalter-Objekt für die Physik-Engine in PLC-Lab nicht sichtbar. Dies kann allerdings bei Bedarf geändert werden.

Dies ist z. B. notwendig, wenn sich ein Bedienfeld mit anderen Objekten mitbewegen soll. Als Beispiel sei die Simulation einer Hebebühne genannt, wenn die Bedienelemente auf der Hebebühne die Bewegung mitvollziehen sollen. In diesem Fall sind die Schalter-Objekte als dynamische Objekte einzustellen und die Eigenschaft „Ist Physik-Objekt“ ist ebenfalls zu selektieren.

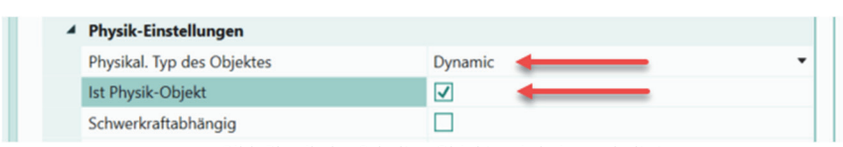


Bild: Physik des Schalter-Objekts wird eingeschaltet.

Ob das Schalter-Objekt auch schwerkraftabhängig sein soll, muss je nach Einsatzgebiet entschieden werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Beispiel (SchalterMitPhysikEigenschaften):

Im folgenden Bild ist eine Hebebühne zu sehen. Die beiden Schalter-Objekte „Auf“ und „Ab“ sind dabei mit dem Korb der Hebebühne verbunden (über eine Körper-Gruppe) und vollziehen somit die gleiche vertikale Bewegung.

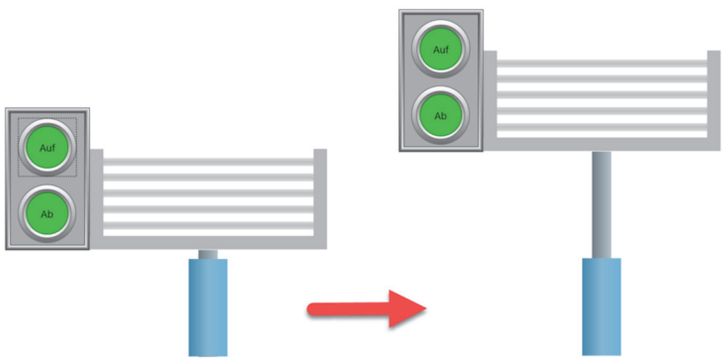


Bild: Beispiel Hebebühne mit Schaltern

Die beiden Schalter-Objekte sind als Taster konfiguriert. Der Schalter „Auf“ hat als Hotkey den Buchstaben „A“ eingetragen, bei Schalter „Ab“ wurde „B“ eingestellt. Somit kann man die Hebebühne mit diesen beiden Tasten auf der Tastatur bedienen.

5.1.3 Der Operand eines Schalter-Objekts

Wie schon erwähnt, ist der Operand eines Schalter-Objekts meist ein Bit-Operand. Er wird schreibend vom Schalter beeinflusst und je nach Konfiguration wird bei Betätigung der Wert 1 oder 0 (bzw. true oder false) in den Operanden geschrieben.

Handelt es sich bei dem Operanden um einen Byte-, Wort- oder Doppelwort-Operanden, dann wird ebenfalls der Wert 0 oder 1 als Zahlenwert in den Operanden eingetragen.

Es besteht allerdings auch die Möglichkeit, bei digitalen Operanden einen Bereich anzugeben.

Beispiel (SchalterDigitalenWertSchreiben):

Bei Betätigung eines Tasters soll der Wert 2000 in das Eingangswort EW20 des Sim-Device geschrieben werden. Ist der Taster nicht betätigt, dann ist der Wert 0 in den Operanden zu schreiben.

Um dies zu erreichen, wird an der Eigenschaft „Operand“ des Schalter-Objekts der Bereich „0-2000“ zusätzlich zum Operanden angegeben. Die vollständige Angabe für das Beispiel lautet: „0-2000 Sim.EW20“.

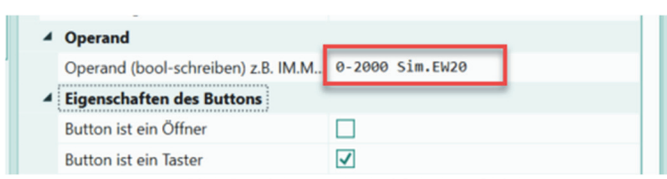


Bild: Bereichsangabe bei Schalter-Objekt

Dies hat zur Folge, dass bei Betätigung des Schalters der Wert 2000 in das EW20 geschrieben wird. In der Ruhestellung des Schalters der Wert 0.

Nachfolgend die Anordnung bei aktiver Simulation, wobei der Inhalt des EW20 in einem Text-Label angezeigt wird.

Bild: Test-Anordnung für das Beispiel

Links im Bild ist der Schalter nicht betätigt, somit hat das EW20 den Inhalt 0. Rechts ist der Schalter im betätigten Zustand zu sehen, folglich wird der Wert 2000 in das EW20 geschrieben.

Hinweis: Es ist durchaus möglich, mit Schalter-Objekten andere Wert als 0 und 1 in einen Operanden zu schreiben. Die zu schreibenden Werte werden dabei in Form eines Bereichs angegeben. Der untere Wert des Bereichs muss nicht unbedingt 0 sein. Natürlich muss der angegebene Operand in der Lage sein, die Werte des Bereichs aufzunehmen.

5.2 Schalter Bezeichnungen

In der Rubrik „Schalter“ sind einige Unter-Rubriken vorhanden. Hierbei handelt es sich um Schalter-Objekte, welche für bestimmte Einsatzgebiete vorbereitet sind.

5.2.1 Schalter ohne Leuchteinsatz

Wie der Name es vermuten lässt, wird dieser Schalter verwendet, wenn man keinen Leuchteinsatz benötigt, z. B. weil neben dem Schalter ein Lampen-Objekt platziert wird, welches den Status entsprechend anzeigt.

5.2.2 Schalter mit Leuchteinsatz

Diese Schalter-Art besitzt einen Leuchteinsatz. Dies bedeutet, man kann zusätzlich zu dem vom Schalter beeinflussten Operanden auch einen Operanden für den Leuchteinsatz angeben.

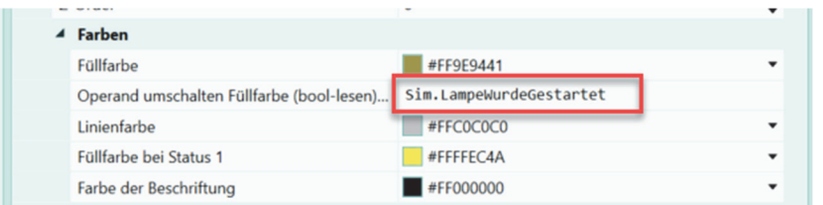


Bild: Operand für die Umschaltung der Füllfarbe

Der an der Eigenschaft „Operand umschalten Füllfarbe“ angegebene Operand ist für das Leuchten des Leuchteinsatzes verantwortlich.

Die Farben des Schalters mit Leuchteinsatz sind so aufeinander abgestimmt, dass ein Leuchten des Schalters gut zu erkennen ist.

5.2.3 Drehschalter

Wird ein Drehschalter für Ein/Aus benötigt, so kann ein Schalter aus dieser Rubrik verwendet werden. Die Funktion des Drehschalters wird durch die beiden Eigenschaften „Bild bei Status 0“ und „Bild bei Status 1“ erreicht. Hier können auch beliebige eigene Grafiken verwendet werden.

Möchte man den Ruhezustand gegen den betätigten Zustand austauschen, so tauscht man einfach die beiden Grafiken gegeneinander aus. Dies bedeutet, die Grafik in „Bild bei Status 0“ wird in „Bild bei Status 1“ übernommen und umgekehrt.

5.2.4 Horizontaler Schalter

Auch diese Schalter-Art wird über zwei Grafiken realisiert, welche in der Eigenschaft „Bild bei Status 0“ und „Bild bei Status 1“ festgelegt sind. Möchte man den Ruhezustand gegen den betätigten Zustand austauschen, so tauscht man einfach die beiden Grafiken gegeneinander aus. Dies bedeutet, die Grafik in „Bild bei Status 0“ wird in „Bild bei Status 1“ übernommen und umgekehrt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

5.3 Schalter mit eigenen Grafiken verwenden

Bei der Erklärung der einzelnen Schalter-Arten wurde bei den Drehschaltern und horizontalen Schaltern erwähnt, dass deren Aussehen durch die verwendeten Grafiken zustande kommt.

Man kann die dabei verwendeten Eigenschaften „Bild bei Status 0“ und „Bild bei Status 1“ auch mit eigenen Grafiken versehen und so das Aussehen der Schalter den eigenen Bedürfnissen anpassen.

Beispiel (SchalterMitEigenerGrafik):

In einem Schalter-Objekt soll ein Pfeil nach Links mit angezeigt werden. Bei der verwendeten Grafik handelt es sich um eine PNG-Grafik mit transparentem Hintergrund. Dies bedeutet, der ursprüngliche Schalter ist weiterhin sichtbar.

Die Grafik wird innerhalb der Eigenschaft „Bild bei Status 0“ mit Hilfe der Schaltfläche am rechten Rand des Eingabefelds der Eigenschaft selektiert. Bei Betätigung der über drei Punkte gekennzeichneten Schaltfläche wird der Datei-Auswahl-Dialog sichtbar, über den die Grafik selektiert werden kann.

Bild: Aufruf des Dialogs zur Auswahl der Grafik-Datei

Wurde die Grafik selektiert, dann ist anschließend der Dateiname in der Eigenschaft eingetragen.

Bild: Eigenschaft mit selektierter Grafik

Des Weiteren ist die Grafik innerhalb des Objekts zu sehen.

Hinweis: Soll der ursprüngliche Schalter beim Einsatz einer eigenen Grafik nicht mehr zu sehen sein, dann muss man die **Füllfarbe** und die **Linienfarbe** des Schalters vollständig transparent einstellen. Dies erreicht man dadurch, dass man die Deckkraft der jeweiligen Farbe auf 0% setzt.

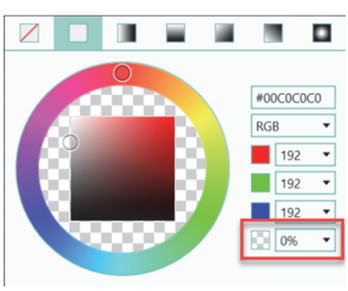


Bild: Deckkraft einer Farbe auf 0% einstellen. Die Farbe ist somit transparent.

Hinweis: Die in einer Anlage verwendeten Grafikdateien werden als Kopie im Verzeichnis des Anlagenprojekts mit abgelegt.

6 Die Lampen-Objekte

Lampen-Objekte werden normalerweise dazu verwendet, den Status von **Bit-Operanden** anzuzeigen. Dazu sind in der Rubrik „Lampen“ Lampen-Objekte in unterschiedlicher Ausprägung vorhanden.

Diese vorkonfigurierten Lampen entstammen alle der gleichen Objekt-Art. Die Objekte wurden nur entsprechend angepasst, d. h. es wurden ihre Eigenschaften verändert.

6.1.1 Die Operanden einer Lampe

In den Eigenschaften eines Lampen-Objekts innerhalb der Rubrik „Operand“ können drei verschiedene Operanden angegeben werden.

In der Eigenschaft „Operand“ wird der Operand eingetragen, dessen Status für das Leuchten der Lampe verantwortlich sein soll. Dies ist der eigentliche Operand der Lampe.

An der Eigenschaft „**Operand blinkende Darstellung**“ kann man darüber hinaus einen Operanden angeben. Hat er den Status 1 und besitzt der Operand der Lampe ebenfalls den Status 1, dann wird die Lampe **blinkend** dargestellt.

Wurde an der Eigenschaft „**Operand pulsende Darstellung**“ ebenfalls ein Operand eingetragen, dann wird die Lampe **pulsierend** dargestellt, sobald der Lampen-Operand den Status 1 besitzt und der hier angegebene Operand den Status 1 hat.

Somit ist es möglich, verschiedene Eskalationsstufen mit einer Lampe darzustellen.

6.1.1.1 Lampe mit digitalem Operanden

Klassisch wird eine Lampe mit einem Bit-Operanden versehen. In PLC-Lab besteht aber auch die Möglichkeit, einen **digitalen Operanden** mit einem Bereich als Operanden einzutragen. Der Bereich wird dann von der Lampe ausgewertet und diese leuchtet, wenn der obere Wert des Bereichs erreicht bzw. überschritten ist.

Beispiel (LampeBereichsangabe):

Eine Lampe soll leuchten, sobald der Wert im EB2 des Sim-Device größer oder gleich 30 ist. Dazu wird am Operanden folgendes angegeben:

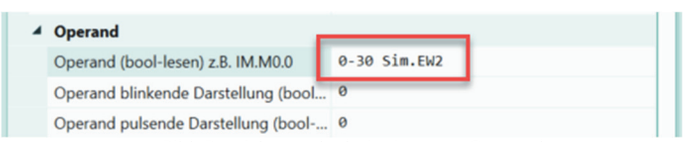


Bild: Bereichsangabe beim Lampen-Operanden

Für das Beispiel wird der Wert im EW2 über einen Schieberegler verändert. Anbei die Anordnung mit verschiedenen Werten:

Bild: Test der Anordnung

Auf der linken Seite des Bilds hat das EW2 einen Wert von unter 30, somit ist die Lampe dunkel. Auf der rechten Seite ist ein Wert von über 30 eingestellt, somit leuchtet die Lampe.

Hinweis: Lampen können auch zur Auswertung von digitalen Operanden verwendet werden. Dazu wird am Operanden zusätzlich eine Bereichsangabe mit angegeben. Sobald der Wert des oberen Bereichs erreicht bzw. überschritten ist, leuchtet die Lampe.

© MHJ-Software GmbH & Co. KG - PLC-Lab

6.2 Benennung der einzelnen Lampen-Arten

Nun zu den einzelnen Lampen-Arten innerhalb der Rubrik „Lampen“.

6.2.1 Leuchtmelder

Leuchtmelder kommen zum Einsatz, wenn man Lampen an Schalttafeln nachahmen möchte. Diese sind in verschiedenen Farben vorkonfiguriert. Ist die passende Farbe nicht enthalten, dann können sie angepasst werden.

6.2.2 LEDs (rund)

Hier handelt es sich um runde LEDs in verschiedenen Farben. Diese können z. B. zum Einsatz kommen, wenn man den Status von Endschaltern visuell hervorheben möchte.

6.2.3 LEDs (rechteckig)

Hierbei handelt es sich um die rechteckige Ausführung von LEDs.

6.2.4 Signal-Säule

Diese Lampen werden in Anlagen verwendet, um bestimmte Zustände an Ort und Stelle zu signalisieren. Beispielsweise ein Material-Stau, fehlendes Material usw.

Neben einzelnen Säulen können auch Säulen mit mehreren Segmenten zusammengestellt werden. Dabei stehen unterschiedliche Farben zur Auswahl, die auch gemischt werden können.

Da auch Signal-Säulen die blinkende und pulsende Darstellung unterstützen, sind sehr viele verschiedene Situationen visualisierbar.

7 Der Endschalter bzw. Sensor in PLC-Lab

In jeder Anlage werden **Sensoren** benötigt, um z. B. Endlagen zu erfassen oder vorhandenes Material zu detektieren. Aus diesem Grund sind in PLC-Lab zahlreiche Möglichkeiten vorhanden, Endschalter oder Sensoren zu konfigurieren.

7.1 Welche Objekte können als Endschalter konfiguriert werden?

Zwar ist in PLC-Lab eine Rubrik „Endschalter“ vorhanden, es können aber auch **alle** geometrischen Physik-Objekte als Endschalter verwendet werden.

Des Weiteren besitzen Zylinder-Objekte bereits Endlagen-Sensoren. Gleiches gilt für lineare und rotierende Verbindungselemente: Auch sie sind bereits mit zahlreichen Sensoren ausgestattet.

Das in der Rubrik „Endschalter“ vorhandene Endschalter-Objekt ist somit nur eine Art Platzhalter für Sensoren oder Endschalter, da viele andere Objekte ebenfalls als Sensoren bzw. Endschalter verwendet werden können.

7.2 Die Endschalter-Einstellungen

Platziert man ein Endschalter-Objekt, bzw. ist ein Endschalter selektiert, dann ist in dessen Eigenschaften die Rubrik „Endschalter Einstellungen“ vorhanden. Dort finden sich die Grundeinstellungen für einen Sensor.

Dazu gehört z. B. der **Endschalter-Operand**. Dieser wird schreibend vom Endschalter beeinflusst, je nachdem, ob dieser betätigt ist oder nicht. Normalerweise handelt es sich bei dem Operanden um einen Bit-Operanden, es können aber auch digitale Operanden mit einem Bereich angegeben werden.

Mit Hilfe der Eigenschaft „**Ist ein Öffner**“ ist selektierbar, ob der Endschalter als Schließer oder Öffner ausgelegt sein soll. Ist er als Öffner ausgelegt, dann wird im Ruhezustand der Wert 1 (True) in einen Bit-Operanden geschrieben.

Die Eigenschaft „**Neutral bei Kollision**“ ist standardmäßig selektiert. Dies bedeutet, kollidiert ein Objekt mit dem Endschalter-Objekt, dann wird diese Kollision zwar vom Endschalter bemerkt, es wird aber keine Kraft auf das kollidierende Objekt ausgeübt. Das andere Objekt „spürt“ den Endschalter somit nicht.

Die Eigenschaft „**Endschalter nicht beeinflussen**“ ist keine typische Endschalter-Eigenschaft. Sie wird eher für andere Objekte benötigt, welche im Einflussbereich eines Endschalters liegen, diesen aber nicht betätigen sollen. Ein typisches Beispiel ist ein Band-Objekt in der Draufsicht, auf dem Lichtschranken angebracht sind. Hier ist beim Band-Objekt die Eigenschaft „Endschalter nicht beeinflussen“ zu selektieren, damit es die Lichtschranken nicht auslöst.

Die beiden weiteren Eigenschaften „Endschalter verändert Kollisiongruppe“ und „Neue Kollisiongruppe bei Kollision“ wird im weiteren Verlauf noch explizit erläutert. Hierzu muss zunächst der Begriff „Kollisionsgruppe“ geklärt werden.

Alle in dieser Rubrik vorhandenen Eigenschaften sind auch bei den geometrischen Physik-Objekten vorhanden. Somit können auch diese die Funktion eines Endschalters übernehmen.

7.3 Wie realisiert man kapazitive, induktive oder auch Farbsensoren: Stichwort „Kollisionsgruppen“

In Anlagen sind häufig Sensoren vorhanden, welche nur auf bestimmte Objekte reagieren. Dies sind z. B. induktive Sensoren, kapazitive Sensoren oder auch Sensoren, welche nur auf bestimmte Farben oder Oberflächen-Eigenschaften der zu detektierenden Objekte reagieren.

Diese Arten von Sensoren sind in PLC-Lab mit Hilfe von **Kollisionsgruppen** realisierbar.

Jedes dynamische oder feststehende Physik-Objekt kann einer bestimmten Kollisionsgruppe zugeordnet werden. Standardmäßig sind die Objekte in der Kollisionsgruppe 0. Dies bedeutet, die Objekte kollidieren mit allen anderen Objekten, unabhängig davon, in welcher Kollisionsgruppe sich die anderen Objekte befinden.

Ist ein Objekt einer Kollisionsgruppe > 0 zugehörig, dann kollidiert es zunächst nur mit anderen Objekten, die ebenfalls dieser Kollisionsgruppe angehören. Alle anderen Objekte werden nicht bemerkt.

7.3.1 Beispiel zu Kollisionsgruppen (EndschalterKollisionsgruppen01)

© MHJ-Software GmbH & Co. KG - PLC-Lab

Im folgenden Beispiel kommen Kollisionsgruppen zum Einsatz. Ein Sensor soll nur betätigt werden, wenn er von einer roten Kugel berührt wird. Wird der Sensor von einer blauen Kugel berührt, dann soll die Berührung nicht zu einer Betätigung führen. Im Bild ist die Anordnung zu sehen:

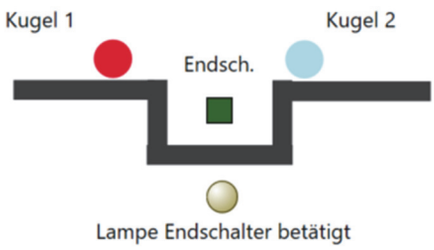


Bild: Anordnung mit Kollisionsgruppen

Das Verhalten des Endschalters erreicht man dadurch, dass er der Kollisionsgruppe 1 zugeordnet wird. Dazu ist die Eigenschaft „Kollisionsgruppe“ innerhalb der Rubrik „Kollisions-Einstellungen“ auf den Wert 1 zu setzen.

Bild: Einstellung der Kollisionsgruppe des Endschalters

Die rote Kugel wird ebenfalls auf die Kollisionsgruppe 1 eingestellt, denn sie soll ja den Endschalter betätigen. Die blaue Kugel ist der Kollisionsgruppe 2 zugehörig.

Bei der Simulation haben diese Einstellungen folgende Wirkung. (Mit der Maus kann man die Kugel im Simulationsmodus bewegen.)

Bild: Zwei Szenarien innerhalb der Anordnung

Links im Bild berührt die blaue Kugel den Endschalter. Man erkennt, dass die Lampe nicht leuchtet, der Endschalter wird somit nicht betätigt. Anders auf der rechten Seite im Bild: Dort berührt die rote Kugel den Endschalter und betätigt ihn, als Folge leuchtet die Lampe.

Hinweis: Das Beispiel hat gezeigt, wie man mit Hilfe von Kollisionsgruppen verschiedenartige Sensoren realisieren kann. Soll ein Sensor nur bei bestimmten Objekten ausgelöst werden, dann wird im Endschalter und in den Objekten die gleiche Kollisionsgruppe eingestellt. Auf diese Weise sind Farbsensoren, Materialsensoren usw. realisierbar.

Auf Kollisionsgruppen wird nochmals in einem eigenen Kapitel explizit eingegangen. Dort wird dann auch die Anwendung der Eigenschaft „Zusätzliche Kollisionsgruppen“ gezeigt.

7.4 Best Practice: Füllfarbe des Endschalters umschalten

Einem Endschalter sieht man zunächst nicht an, ob er betätigt ist oder nicht bzw. wie der Status des Operanden ist, den der Endschalter beeinflusst. Man kann neben dem Endschalter eine LED platzieren und sie vom Status des Operanden des Endschalters abhängig machen. Dann leuchtet die LED, sobald der Operand den Status 1 besitzt.

Man kann allerdings auch auf die zusätzliche Lampe verzichten und stattdessen die Farbe des Endschalters vom Endschalter-Operanden abhängig machen. Dazu wird der Operand zusätzlich in der Eigenschaft „**Operand umschalten Füllfarbe**“ angegeben.

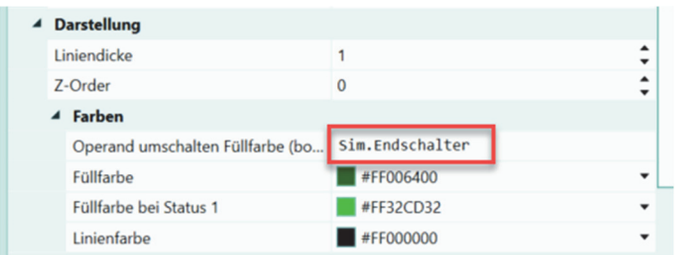


Bild: Endschalter-Operand schaltet auch die Füllfarbe des Endschalters um.

Im Beispiel für die Kollsionsgruppen bedeutet dies:

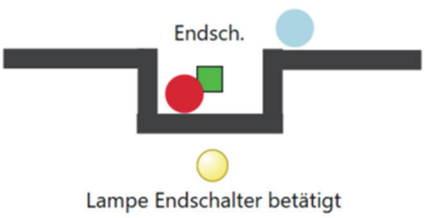
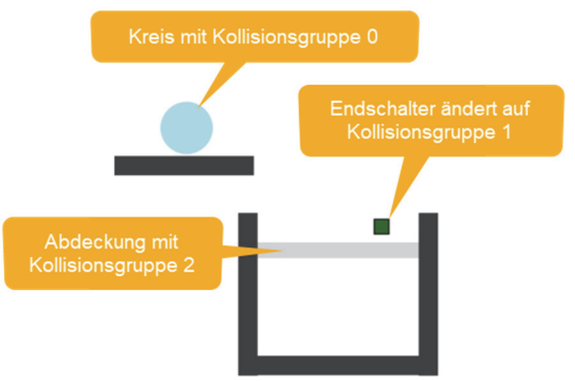


Bild: Bei Betätigung wird auch die Füllfarbe umgeschaltet.

Dass eigentlich die Lampe nicht mehr benötigt wird, da ja die Farbe des Endschalters schon den Zustand des Endschalter-Operanden signalisiert.

7.5 Verändern der Kollisionsgruppe durch einen Endschalter

Dass ein Endschalter die Kollisionsgruppe eines Objekts verändern kann, wurde schon erwähnt. Nun soll ein Beispiel folgen, welches diese Möglichkeit verdeutlicht.

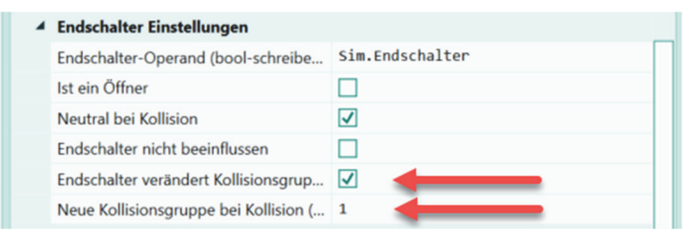


<u>Beispiel (EndschalterKollisionsgruppen02):</u>

Die Anordnung des Beispiels hat folgenden Aufbau (Bild rechts):

Der blaue Kreis besitzt die Kollisionsgruppe 0, kollidiert also mit Objekten aller anderen Kollisionsgruppen. Auf dem Behälter ist eine Abdeckung, welche der Kollisionsgruppe 2 zugeordnet ist. Der Endschalter ist so eingestellt, dass er bei Berührung ein Objekt in die Kollisionsgruppe 1 überführt.

Bild: Die Einstellungen des Endschalters



Wird die Simulation gestartet und das Kreis-Objekt auf die Abdeckung fallen gelassen, dann zeigen sich folgende Reaktionen:

© MHJ-Software GmbH & Co. KG - PLC-Lab

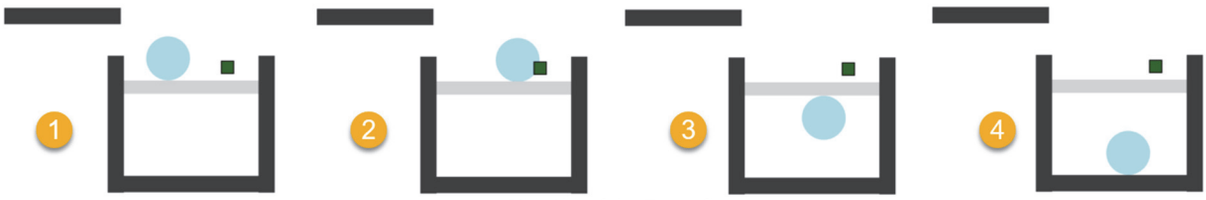


Bild: Ablauf des Testaufbaus

1. Der blaue Kreis liegt zunächst auf der Abdeckung. Denn der Kreis hat die Kollisionsgruppe 0 und kollidiert somit mit allen anderen Objekten – auch mit der Abdeckung, welche der Kollisionsgruppe 2 angehört.
2. Nun berührt der Kreis den Endschalter. Somit wird der Kreis in die Kollisionsgruppe 1 überführt. Als Folge kollidiert er nicht mehr mit der Abdeckung.
3. Der Kreis fällt durch die Abdeckung hindurch …
4. … und liegt nun am Boden des Behälters.

Das Beispiel hat gezeigt, wie man mit Hilfe eines Endschalters die **Kollisionsgruppe** von Objekten **verändern** kann.

7.6 Angabe eines digitalen Operanden am Endschalter

Meist wird bei einem Endschalter ein **Bit-Operand** angegeben. Allerdings ist auch die Angabe eines digitalen Operanden möglich. Dies macht insbesondere dann Sinn, wenn man zusätzlich einen Bereich mit angibt. Ist der Endschalter beispielsweise als Schließer ausgelegt und wurde für den digitalen Operanden der Bereich „0-500“ angegeben, dann wird bei Betätigung des Endschalters der Wert 500 in den Operanden geschrieben.

Grundsätzlich ist also auch die Angabe von Byte-, Wort- oder Doppelwort-Operanden als Endschalter-Operand möglich.

7.7 Fazit

In diesem Abschnitt wurde der Endschalter vorgestellt. Es wurde gezeigt, wie man mit Hilfe von Kollisionsgruppen Endschalter realisiert, die auf **unterschiedliche Materialien, Farben oder Formen reagieren**.

Weiterhin wurde die Möglichkeit vorgestellt, dass ein Endschalter die Kollisionsgruppe anderer Objekte verändert und damit das Verhalten innerhalb eines Anlagen-Aufbaus beeinflusst.

Das Endschalter-Objekt ist im Prinzip ein Platzhalter, denn alle geometrischen Physik-Objekte besitzen ebenfalls die Endschalter-Eigenschaften und können ebenso als Endschalter verwendet werden. Diese werden im weiteren Verlauf vorgestellt.

Daneben besitzen Objekte wie Zylinder, Flüssigkeits-Objekte, Verbindungselemente bereits eingebaute Sensoren. Hier sind keine zusätzlichen Endschalter notwendig. Auch dies wird im weiteren Verlauf noch gezeigt.

8 Geometrische Formen

Die geometrischen Formen sind die Grundbausteine von PLC-Lab. Diese Objekte wurden bereits in den Beispielen zuvor verwendet, ohne sie explizit zu erwähnen. Dies wird im folgenden Kapitel nachgeholt.

Mit den geometrischen Formen werden die meisten Anlagenteile abgebildet. Sie sind mit den später gezeigten Verbindungselementen kombinierbar und somit dynamisierbar. Man kann mit Hilfe von Körper-Verbindungen beliebige Körper kreieren und in den Anlagenprozess einbinden.

Innerhalb der geometrischen Formen stehen drei Grundformen zur Auswahl: Rechtecke, Ellipsen und Dreiecke. Diese Formen sind wiederum in drei Kategorien unterteilt. Im ersten Schritt sollen diese Kategorien erläutert werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

8.1 Charakter der geometrischen Formen: Dynamisch, Feststehend, Dekoration

8.1.1 Die Kategorie „Dynamisch (Physik)“

Die Objekte dieser Kategorie bilden den **beweglichen Teil** einer Anlage. Sie sind der Physik-Engine unterworfen, können schwerkraftabhängig sein und unterschiedliche Massen besitzen. Ihre Oberflächenreibung ist einstellbar, sie können magnetisch sein und das Verhalten eines Metallkörpers oder eines Gummiballs aufweisen.

Diese Objekte werden von Förderelementen bewegt und bilden den dynamischen Teil einer linearen Bewegung bzw. einer Drehbewegung.

8.1.2 Die Kategorie „Feststehend (Physik)“

Auch diese Objekte sind der Physik unterworfen. Allerdings sind die Objekte **nicht beweglich**. Sie bilden in Anlagen Endbegrenzungen, Wände, Führungsschienen und feststehende Anschläge.

Kollidieren dynamische Objekte mit feststehenden Objekten, dann wird ihre Bewegung abrupt unterbrochen bzw. werden sie abgelenkt.

Stellt man sich einen Flipperautomaten vor, dann würde die Kugel als dynamisches Objekt ausgelegt werden, die Seitenbegrenzungen und Führungen wären feststehende Physik-Objekte.

8.1.3 Die Kategorie „Dekoration (Keine Physik)“

Die Bezeichnung „**Dekoration**“ sagt im Prinzip schon, welche Funktion diese Objekte übernehmen: Sie sind für die Physik-Engine nicht vorhanden und haben somit keine Auswirkung auf die dynamischen oder feststehenden Physik-Objekte.

Die Objekte dieser Kategorie werden beispielsweise als Abdeckungen verwendet oder um einen Anlagenaufbau optisch realistischer zu gestalten.

8.1.4 Exemplarische Anordnung mit allen drei Charakteren

Im Bild ist ein Billardtisch zu sehen, dabei sind exemplarisch die Kategorien der geometrischen Formen benannt, würde man den Tisch in PLC-Lab umsetzen.

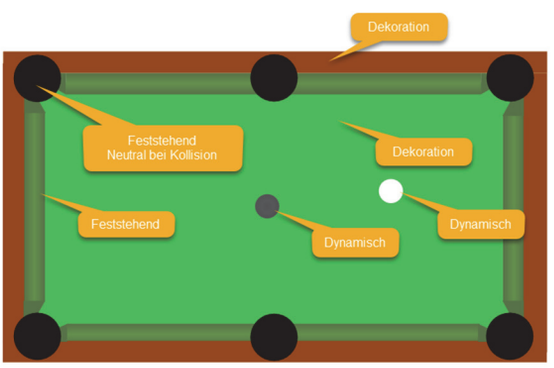


Bild: Billardtisch als Beispielaufbau

Die beiden Kugeln sind als dynamische Objekte ausgelegt, wobei die Schwerkraftabhängigkeit abzuschalten ist. Die grünen Seitenbegrenzungen sind klassische feststehende Objekte. Ebenso die Löcher, in denen die Kugeln verschwinden sollen. Allerdings müsste bei ihnen zusätzlich die Eigenschaft „Neutral bei Kollision“ eingestellt sein, denn die Kugeln sollen ja nicht an ihnen abprallen.

Die Umrandung des Tischs und der grüne Tisch selbst sind Dekorations-Objekte.

© MHJ-Software GmbH & Co. KG - PLC-Lab

8.2 Verwendung von Dekorations-Objekten als Lampen mit eigenen Grafiken

Bei der Erläuterung von Lampen-Objekten wurde erwähnt, dass die Lampen nicht mit eigenen Grafiken versehen werden können. Für diesen Zweck kann man Dekorations-Objekte verwenden.

Beispiel (GeometrFormenLampe01):

Es soll eine kreisförmige Lampe mit eigener Grafik realisiert werden. Die Grafik ist eine PNG-Grafik mit transparenten Bereichen. Die Lampe soll gelb leuchten, sobald der mit ihr verknüpfte Operand den Status 1 besitzt.

Zu diesem Zweck wird ein Dekorations-Objekt in Form einer Ellipse selektiert und gleichförmig auf der Zeichenfläche aufgezogen. Als Füllfarbe wird eine weiße Farbe eingestellt, die Füllfarbe bei Status 1 ist gelb.

Der Operand Sim.E0.0 soll den Status umschalten und wird somit in der Eigenschaft „Operand umschalten Füllfarbe“ angegeben. Innerhalb der Eigenschaft „Operand umschalten Bild“ wird die anzuzeigende Grafik ausgewählt bzw. eingetragen. Die Umschaltung des E0.0 wird mit Hilfe eines Schalter-Objekts vorgenommen.

Nachfolgendes Bild zeigt die Simulation.

Bild: Anordnung für dieses Beispiel

Auf der linken Seite des Bilds hat der E0.0 den Status 0, somit wird das Kreis-Objekt mit einer weißen Farbe dargestellt. Rechts wurde der Schalter betätigt, als Folge hat der Operand E0.0 den Status 1. Der Kreis wird mit einer gelben Farbe gefüllt, die Farbe ist in den transparenten Bereichen der Grafik zu sehen.

Auf diese Weise können sehr einfach lampenähnliche Objekte mit eigenen Grafiken realisiert werden.

8.3 Dynamische Objekte ohne Schwerkraft für Anlagen in der Draufsicht

Möchte man Anlagen in der Draufsicht zeichnen, dann muss meist die Schwerkraftabhängigkeit der dynamischen Objekte abgeschaltet werden. Dazu wird die Eigenschaft „Schwerkraftabhängig“ innerhalb der „Physik-Einstellungen“ ausgeschaltet.

Dies hat zur Folge, dass sich das Objekt beim Start der Simulation wie in einem schwerelosen Raum verhält. Wird eine Kraft auf das Objekt ausgeübt, dann bewegt es sich ungebremst. Dieses Verhalten ist oft nicht gewünscht, aus diesem Grund kann bei den Objekten eine Dämpfung eingestellt werden.

8.3.1 Einsatz der Dämpfung für Objekte ohne Schwerkraft

Bei geometrischen Objekten ist sowohl für die lineare Bewegung als auch für die Rotationsbewegung eine Dämpfung einstellbar. Die Eigenschaften dazu befinden sich innerhalb der Rubrik „Experten-Einstellungen->Physik-Einstellungen“ und haben die Bezeichnung „Dämpfung für lineare Bewegungen“ sowie „Dämpfung für Rotationsbewegungen“.

Beispiel (GeometrFormenDaempfung01):

Gegeben sei folgende Anordnung:

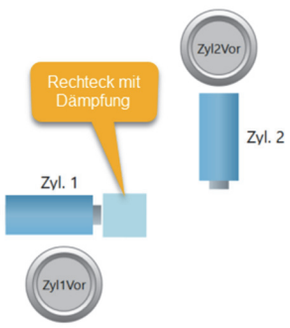


Bild: Anordnung für das Beispiel

Das Rechteck soll von den beiden Zylindern bewegt werden. Die Anordnung ist aus der Vogelperspektive aufgebaut. Das blaue Rechteck ist ein dynamisches Physik-Objekt, dessen Schwerkraft abgeschaltet ist. Die Dämpfungseinstellungen wurden wie folgt vorgenommen:

Bild: Dämpfungs-Einstellungen für das dynamische Objekt

Wäre bei dem Objekt keine Dämpfung eingestellt, dann würde es vom Zylinder 1 angestoßen und sich anschließend schwerelos durch den Raum bewegen.

Mit der Dämpfung zeigt sich folgendes Verhalten:

Bei Betätigung des Schalters „Zyl1Vor“ fährt der Zylinder 1 aus und bewegt das blaue Rechteck nach rechts. Ist der Zylinder 1 komplett ausgefahren, dann befindet sich das Rechteck im Einflussbereich von Zylinder 2. Bei Betätigung des Schalters „Zyl2Vor“ fährt Zylinder 2 aus und bewegt das Rechteck nach unten. Die Dämpfung kann man sich wie eine Art Reibung vorstellen, welche die Krafteinwirkung auf das dynamische Objekt bremst.

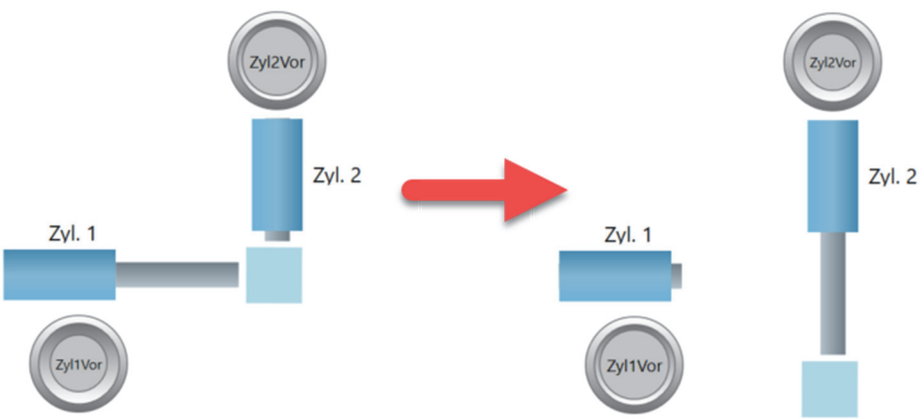


Bild: Verhalten der Anordnung bei eingestellter Dämpfung des dynamischen Objekts

Fazit: Werden Anlagen in der Draufsicht erstellt und kommen dynamische Objekte zum Einsatz, deren Schwerkraft abgeschaltet wurde, dann ist es meist notwendig eine Dämpfung für die linearen Bewegungen und die Rotationsbewegungen einzustellen. Damit wird verhindert, dass sich die dynamischen Objekte nach einer Krafteinwirkung frei und ungebremst im Raum bewegen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

9 Rohrleitungen

Die in dieser Rubrik befindlichen Objekte werden zum Bau von Rohrleitungssystemen verwendet. Typische Anwendungen sind die Versorgungs- bzw. Abfluss-Leitungen von Flüssigkeitsbehältern.

Mit Hilfe der Eigenschaft „Operand für Farbveränderung“ kann die Farbe des Objekts verändert und so der Eindruck des Durchflusses eines Mediums vermittelt werden. Die Farbe ist dabei frei konfigurierbar.

Die Objekte in der Rubrik „Rohrleitungen“ sind als Dekorations-Objekte ausgelegt, sie sind also in der Grundeinstellung keine Physik-Objekte. Dies kann aber bei Bedarf geändert werden, indem man einfach die Option „Ist Physik-Objekt“ selektiert und den physikalischen Typ des Objektes entsprechend anpasst.

9.1 Verwendung der Eigenschaft zum Verändern der Füllfarbe

Beispiel (Rohrleitung01):

Im Beispiel sollen einige Rohrleitungs-Objekte zum Einsatz kommen, unter anderem der 90° Winkel. Da alle Objekte beliebig rotierbar sind, ist ihr Einsatz sehr variabel. So wird der 90° Winkel im Beispiel zunächst „normal“ verwendet und dann in einer um 180° rotierten Form. Nachfolgend ist die Anordnung zu sehen:

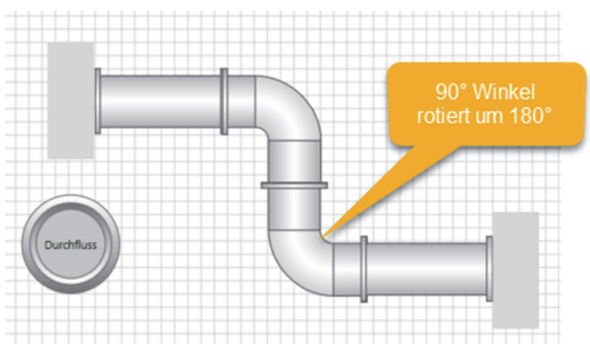


Bild: Anordnung des Beispiels

Der Durchfluss durch das Leitungssystem ist vom Operanden E0.0 „Durchfluss“ des Sim-Device abhängig. Sobald der E0.0 den Status 1 besitzt, sollen die Rohrleitungen den Durchfluss optisch signalisieren. Dazu wird der Operand in der Eigenschaft „Operand für Farbveränderung“ eingetragen. Ist der Operand in der Symboltabelle vorhanden, dann ist dies einfach per Drag-and-Drop auf die Rohrleitung möglich.

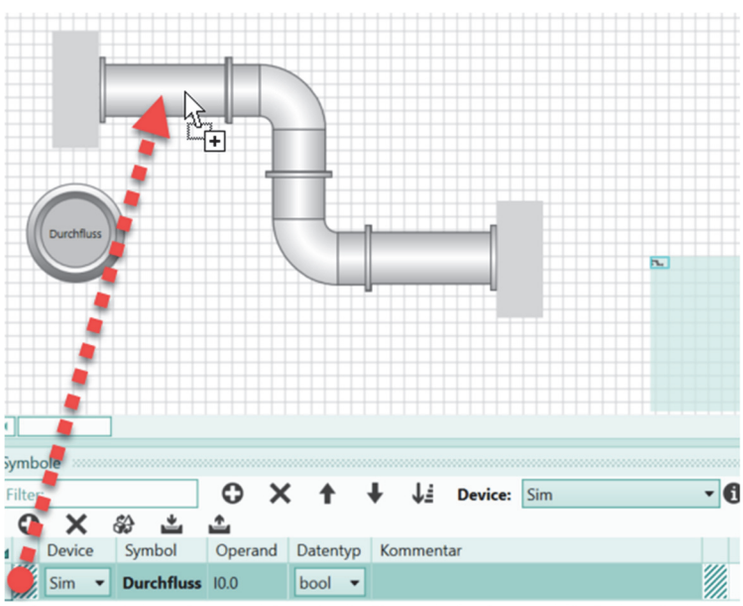


Bild: Der Operand wird per Drag-and-Drop in die Eigenschaft „Operand für Farbveränderung“ eingefügt.

Auf diese Weise hat man sehr schnell die einzelnen Rohrleitungen mit dem Operanden verknüpft. Für das Beispiel soll die Umschaltung des Eingangs über einen Schalter erfolgen.

Beim Start der Simulation zeigt sich somit folgendes Verhalten:

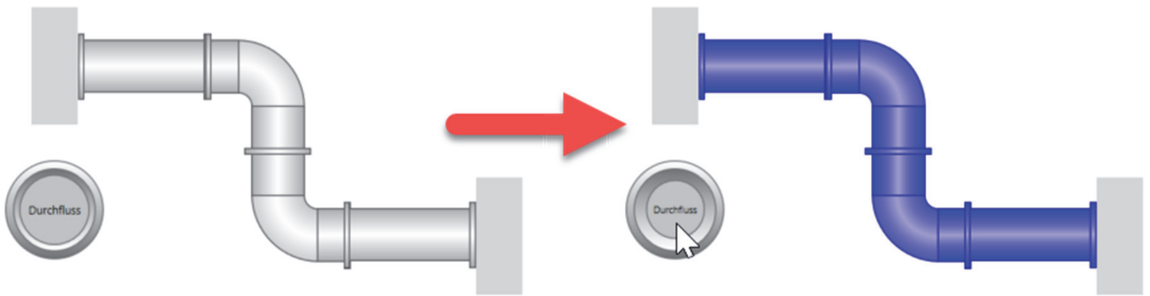


Bild: Simulation der Anordnung

Auf der linken Seite des Bilds ist noch kein Durchfluss vorhanden, die Rohrleitungen werden ohne Färbung dargestellt. Rechts wurde der Schalter betätigt und somit der E0.0 auf den Status 1 gesetzt. Dies hat zur Folge, dass die Rohrleitungen mit einer blauen Färbung versehen werden, was den Durchfluss des Mediums symbolisiert.

9.2 Animation einer Pumpe mit Hilfe der Verbindung für Drehbewegungen

Beispiel (Rohrleitung02):

Innerhalb der Rohrleitungen befindet sich auch ein Pumpen-Symbol.

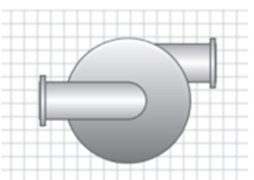


Bild: Pumpe innerhalb der Rohrleitungen

Im Beispiel soll es als Ausgangspunkt für die Animation einer Pumpe verwendet werden. Dabei wird das Pumpen-Symbol zunächst in die im letzten Beispiel verwendete Anordnung eingebaut. Dabei sei angemerkt, dass sich der Rohrdurchmesser der Pumpe dynamisch verändert, wenn man das Pumpen-Symbol in der Höhe verändert. Man kann aber auch den Durchmesser direkt angeben, indem man den Wert der Eigenschaft „Rohr-Durchmesser bei Pumpe“ durch einen positiven Wert ersetzt.

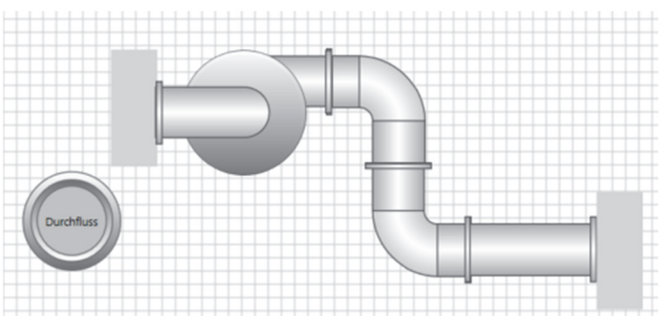


Bild: Eingebaute Pumpe

Im nächsten Schritt sollen einige Veränderungen an der Pumpe vorgenommen werden. Zunächst wird die Pumpe als Physik-Objekt eingestellt mit dem Typ „Static“.

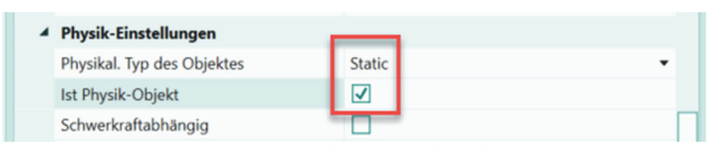


Bild: Pumpe als Physik-Objekt vom Typ „Static“ einstellen

© MHJ-Software GmbH & Co. KG - PLC-Lab

Dann wird die Deckkraft (Transparenz) der beiden Füllfarben „Füllfarbe“ und „Füllfarbe bei Status 1“ auf ca. 50% eingestellt.

In der Pumpe wird in der Eigenschaft „Operand für Farbveränderung“ der Operand „Durchfluss“ eingetragen, dabei geht man genauso vor wie oben beschrieben: Der Operand wird per Drag-and-Drop auf das Objekt gezogen und fallen gelassen.

Im nächsten Schritt wird eine dynamische Ellipse platziert und mit einer grauen Füllfarbe versehen.

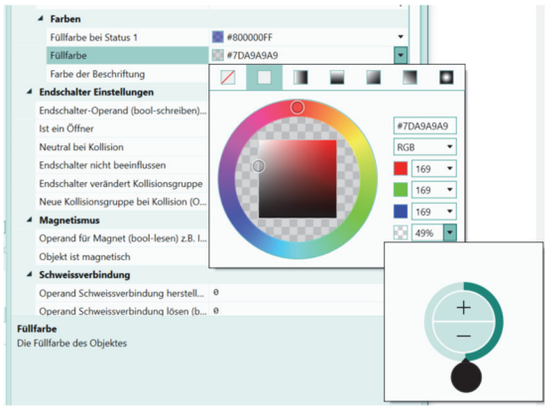


Bild: Deckkraft der beiden Farben wird auf ca. 50% eingestellt.

Die Ellipse hat folgendes Aussehen:

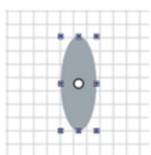


Bild: Neu erzeugte dynamische Ellipse

Die Höhe der Ellipse entspricht ungefähr der Höhe der Pumpe.

Diese Ellipse soll **animiert** werden und so anzeigen, dass die Pumpe in Betrieb ist. Zu diesem Zwecke soll die Ellipse rotieren, sobald der Operand „Durchfluss“ den Status 1 besitzt. Hierfür ist eine Verbindung für Drehbewegung notwendig, welche sich in der Rubrik „Verbindungselemente bzw. Gelenke“ befindet.

Diese Verbindung wird selektiert. Anschließend bewegt man die Maus über das Pumpen-Symbol. Über der Pumpe betätigt man die linke Maustaste, hält sie betätigt und zieht die Maus über die Ellipse. Dort wird die linke Maustaste losgelassen.

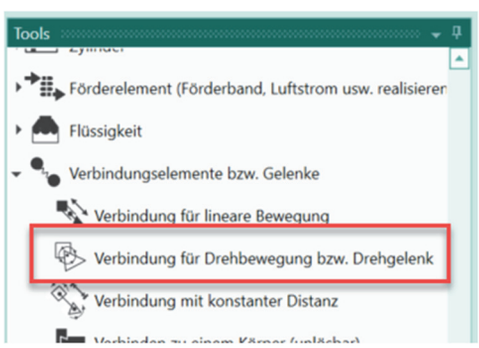


Bild: Verbindung für Drehbewegung

Als Folge wird zwischen den beiden Objekten eine Verbindung für Drehbewegungen eingezeichnet.

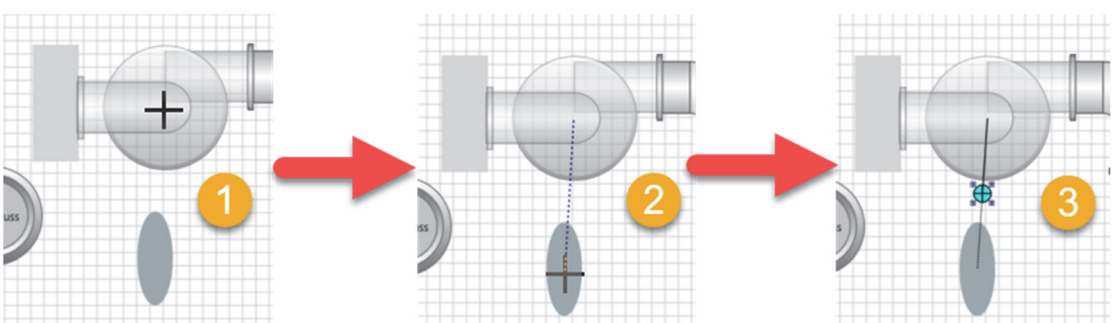


Bild: Zeichnen der Verbindung für Drehbewegung

Wichtig ist, dass das Zeichnen der Verbindung bei der Pumpe begonnen wird! Denn das zweite Objekt ist das Objekt, welches rotiert, und das soll im Beispiel die Ellipse sein.

Im obigen Bild (Punkt 3) ist die Verbindung erzeugt worden, der Kreis mit der bläulichen Farbe symbolisiert den Drehpunkt der Rotation. Dieser wird in die Mitte der Ellipse gelegt und beide Objekte anschließend in die Mitte der Pumpe verschoben. Hierzu sind sowohl die Ellipse als auch der Drehpunkt zu selektieren.

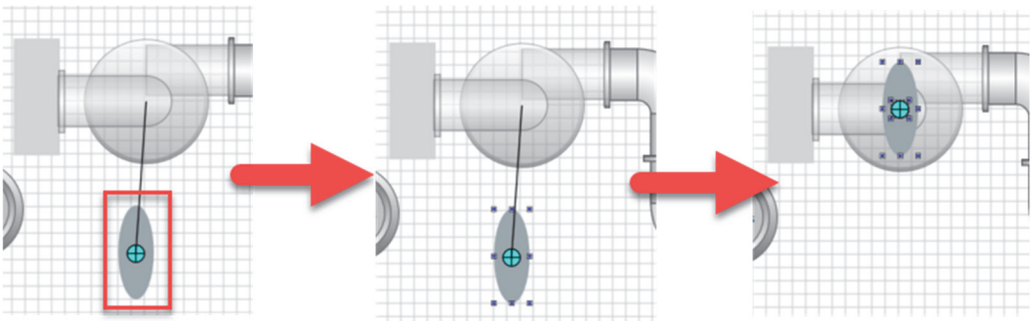


Bild: Verschieben der Ellipse und des Rotations-Punkts in die Mitte der Pumpe

Nun muss die Rotationsbewegung noch mit dem Operanden „Durchfluss“ verbunden werden. Dazu wird der Drehpunkt selektiert und in die Eigenschaft „Operand für Rechtsdrehung“ das Symbol des Operanden eingetragen.

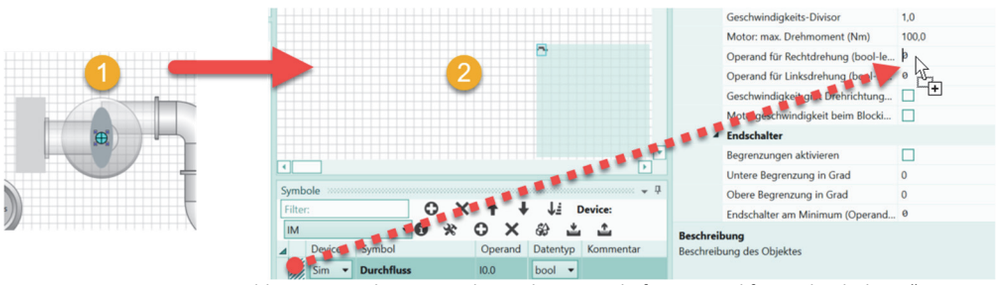


Bild: Eintragen des Operanden in die Eigenschaft „Operand für Rechtsdrehung“

Nach der Aktion ist der Operand in der Eigenschaft eingetragen.

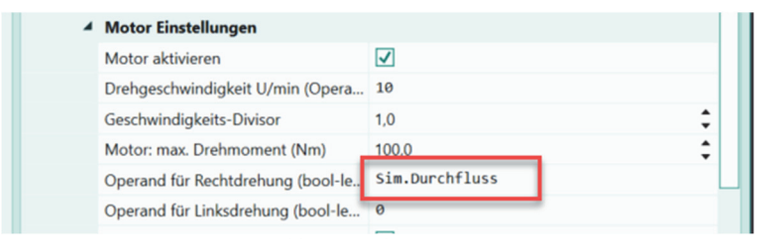


Bild: Operand ist eingetragen.

Schaltet man nun die Simulation ein und betätigt den Schalter für den Durchfluss, dann werden die Rohrleitungen eingefärbt und zusätzlich rotiert die Ellipse innerhalb der Pumpe als Zeichen dafür, dass die Pumpe in Betrieb ist.

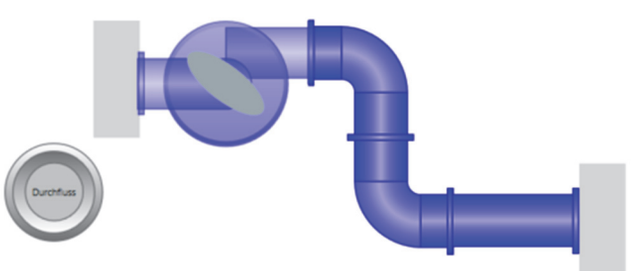


Bild: Neue Anordnung mit animierter Pumpe

Hinweis: Die im Beispiel verwendete Verbindung für Drehbewegung wird in einem eigenständigen Kapitel noch genauer beschrieben. So viel vorweg: Man kann die Geschwindigkeit der Rotation über einen analogen Wert vorgeben und so auch die Fördergeschwindigkeit der Pumpe animiert wiedergeben.

© MHJ-Software GmbH & Co. KG - PLC-Lab

10 Zylinder

In dieser Rubrik sind Zylinder vorhanden, welche über bestimmte **Grundkonfigurationen** verfügen. Alle Zylinder entstammen dem gleichen „Grund-Objekt“ und können somit beliebig umkonfiguriert werden.

Folgende Zylinderarten sind vorkonfiguriert:

- Zylinder **mit Federrückstellung**
- Zylinder **ohne Feststelleinheit**
- Zylinder **mit Feststelleinheit**

Diese Zylinder-Arten werden nachfolgend mit Beispielen vorgestellt.

10.1 Zylinder mit Federrückstellung

Der Zylinder mit Federrückstellung ist ein einfachwirkender Zylinder. Hat der Operand, der für die Bewegung des Zylinders verantwortlich ist, den Status 1, dann bewegt sich die Kolbenstange in die angegebene Richtung. Wechselt der Status des Operanden auf 0, dann bewegt sich die Kolbenstange zurück.

Der Zylinder hat die Eigenschaften „Operand Zylinder einfahren“ und „Operand Zylinder ausfahren“. **Bei dem Zylinder mit Federrückstellung wird an einer der beiden Eigenschaften ein Operand angegeben.**

Beispiel (Zylinder01):

Ein Zylinder soll ausfahren, wenn der Operand E0.0 des Sim-Device den Status 1 besitzt. Bei Status 0 fährt der Zylinder wieder zurück.

Auf der Zeichenfläche wird ein Zylinder mit Federrückstellung platziert. In dessen Eigenschaft „Operand Zylinder ausfahren“ wird der Operand E0.0 des Sim-Device eingetragen.

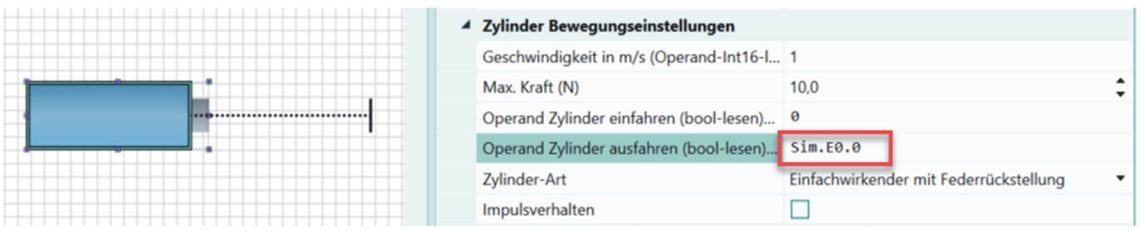


Bild: Angabe des Operanden an der Eigenschaft „Operand Zylinder ausfahren“

Zusätzlich ist ein Schalter-Objekt zu platzieren, welches den Operanden E0.0 beeinflusst.

Jetzt kann die Simulation gestartet werden. Sobald man den Schalter betätigt, fährt die Kolbenstange des Zylinders aus. Hat die Kolbenstange die Endposition erreicht, dann stoppt diese. Wenn der Schalter nicht mehr betätigt ist und somit der E0.0 den Status 0 besitzt, fährt die Kolbenstange zurück. Dies geschieht auch, wenn die Kolbenstange noch nicht ihre vordere Endlage erreicht hatte.

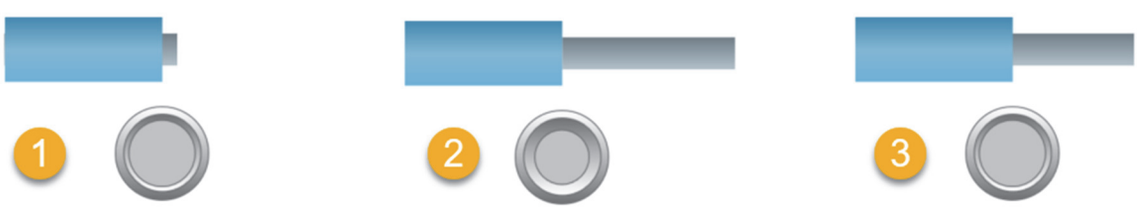


Bild: Verhalten des Zylinders mit Federrückstellung

Im obigen Bild wird im Punkt 1 die Ausgangssituation dargestellt. In Punkt 2 wurde der Schalter betätigt und die Kolbenstange ist ausgefahren. Punkt 3 zeigt die Situation, in welcher der Schalter unbetätigt ist, somit der E0.0 den Status 0 besitzt und die Kolbenstange zurückfährt.

10.2 Zylinder ohne Feststelleinheit

Bei dieser Zylinder-Art ist in **beiden** Eigenschaften „Operand Zylinder einfahren“ und „Operand Zylinder ausfahren“ ein Operand einzutragen. Da der Zylinder über keine Feststelleinheit verfügt, ist die Kraft des Zylinders nur so lange vorhanden, wie der jeweilige Operand den Status 1 besitzt.

Dies hat zur Folge, dass der Zylinder eine Last nur so lange halten kann, wie der Status eines der Operanden den Wert 1 hat. Dies muss insbesondere dann bedacht werden, wenn der Zylinder vertikal angeordnet ist.

Beispiel (Zylinder02A):

Die Kolbenstange eines Zylinders soll sich nach rechts bewegen, wenn der Status des E0.0 den Wert 1 besitzt. Die Bewegung nach links wird über den Operanden E0.1 gesteuert. Beide Operanden gehören zum Sim-Device und werden über Taster beeinflusst.

Es wird ein Zylinder ohne Feststelleinheit horizontal auf der Zeichenfläche platziert. In den Eigenschaften „Operand Zylinder einfahren“ und „Operand Zylinder ausfahren“ sind die entsprechenden Operanden anzugeben.

Bild: Zylinder mit eingestellten Operanden

Die beiden Operanden sind mit Hilfe von als Taster konfigurierten Schalter-Objekten zu beeinflussen. Damit hat die Anordnung folgendes Aussehen:

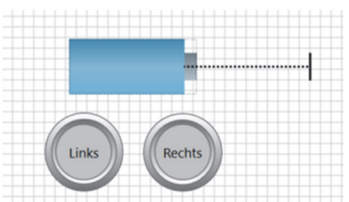


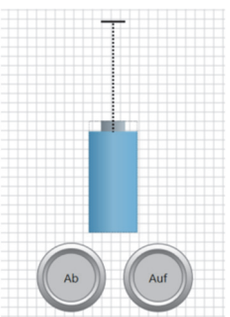
Bild: Zylinder und Taster

Startet man die Simulation und betätigt den Taster „Rechts“, dann wird die Kolbenstange ausgefahren. Die Betätigung des Tasters „Links“ bewirkt das Einfahren der Kolbenstange. Tippt man einen Taster nur kurzzeitig an, dann fällt auf, dass die Kolbenstange etwas „nachläuft“. Dies liegt daran, dass der Zylinder über keine Feststelleinheit verfügt, somit wirkt keine Kraft mehr auf die Kolbenstange, wenn der Status der Bewegungs-Operanden 0 ist. Durch die Massenträgheit bewegt sich die Kolbenstange noch etwas, falls ansonsten kein Widerstand vorhanden ist.

Beispiel (Zylinder02B):

Richtig deutlich wird dieses Verhalten, wenn man den Zylinder vertikal anordnet (durch Rotation des Zylinders).

Bild: Vertikal angeordneter Zylinder



© MHJ-Software GmbH & Co. KG - PLC-Lab

Betätigt man den Taster für die Aufwärtsbewegung der Kolbenstange und lässt diesen wieder los, dann senkt sich die Kolbenstange wieder nach unten ab, da keine Kraft vorhanden ist, welche die Kolbenstange an der Stelle hält.

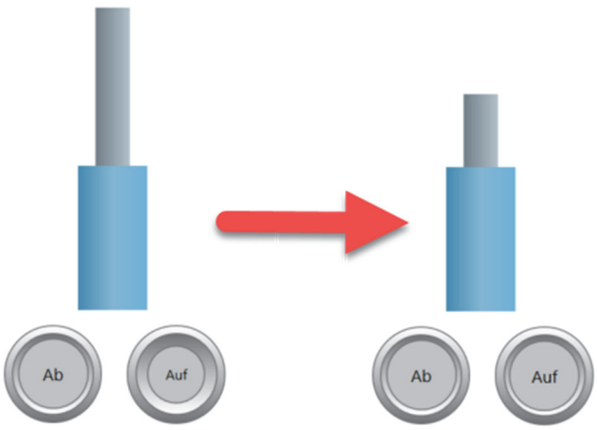


Bild: Simulation des vertikalen Zylinders ohne Feststelleinheit

Auf der linken Seite des obigen Bilds ist der Taster „Auf“ betätigt, somit bewegt sich die Kolbenstange nach oben und behält ihre Position. Rechts ist der Taster nicht mehr betätigt, Der Status des Operanden ist 0 und die Kolbenstange sinkt wegen der Schwerkraft nach unten ohne, dass die Bewegung nach unten über den Taster „Ab“ ausgelöst wird.

10.3 Zylinder mit Feststelleinheit

Auch diese Zylinder-Art unterstützt die Angabe von zwei Operanden, d. h. für jede Bewegung der Kolbenstange kann ein Operand angegeben werden. Hat einer der Operanden den Status 1, dann bewegt sich die Kolbenstange in die entsprechende Richtung. Wird der Status des Operand 0 bevor die Endlage erreicht ist, dann behält die Kolbenstange ihre Position und wird mit der in der Eigenschaft „Max. Kraft“ angegebenen Kraft gehalten.

Beispiel (Zylinder03):

Die Kolbenstange eines Zylinders soll sich nach oben bewegen, wenn der Status des E0.0 den Wert 1 hat. Die Bewegung nach unten wird über den Operanden E0.1 gesteuert. Beide Operanden gehören zum Sim-Device und werden über Taster beeinflusst.

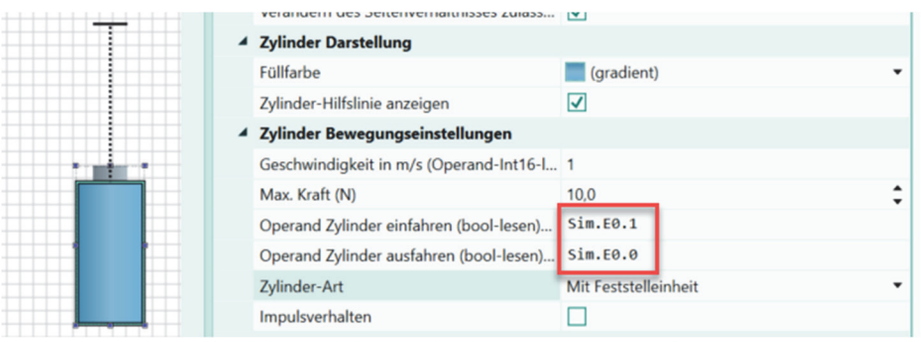


Bild: Zylinder mit Feststelleinheit und den Bewegungs-Operanden

Damit hat die Anordnung folgendes Aussehen:

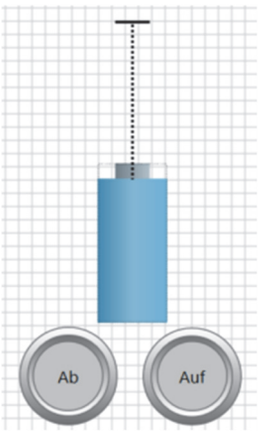


Bild: Vertikal angeordneter Zylinder mit Tastern

Startet man die Simulation der Anordnung und betätigt den Taster „Auf“, dann bewegt sich die Kolbenstange nach oben. Lässt man den Taster kurz darauf wieder los, dann bleibt die Kolbenstange an deren Position. Über ein erneutes Betätigen von „Auf“ fährt diese weiter nach oben, bis die obere Endlage erreicht ist.

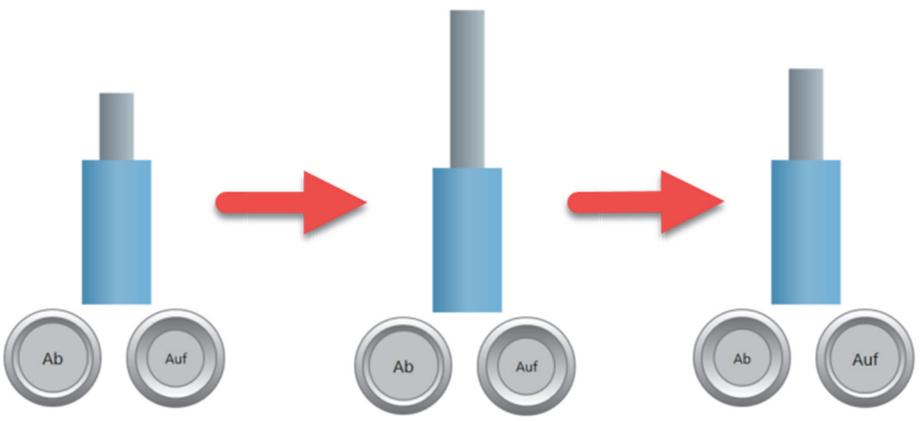


Bild: Simulation des Anordnung

Bei Betätigung des Tasters „Ab“ wird die Kolbenstange nach unten bewegt. Auch hier kann vor dem Erreichen der unteren Endlage der Taster losgelassen werden. Die Kolbenstange verharrt daraufhin an der gestoppten Position.

Oben wurde schon erwähnt, dass die Feststelleinheit des Zylinders über die in der Eigenschaft „Max. Kraft“ eingestellte Kraft verfügt. Solange also die einwirkende Kraft geringer als diese max. Kraft ist, bleibt die Position der Kolbenstange erhalten. Ist die einwirkende Kraft größer, dann bewegt sich die Kolbenstange in Richtung der einwirkenden Kraft, bis die Endlage erreicht ist.

Im folgenden Bild wird die Kolbenstange mit dynamischen Objekten „belastet“. Die beiden ersten Objekte können noch gehalten werden. Beim dritten Objekt wird die Last zu hoch und die Kolbenstange wird nach unten gedrückt.

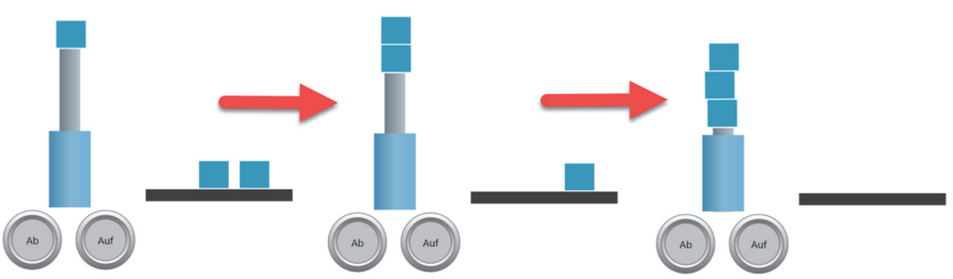


Bild: Die Kolbenstange wird mit dynamischen Objekten belastet.

Der Zylinder wurde dabei auf eine max. Kraft von 100 N eingestellt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

10.4 Wechsel der Zylinder-Art

Hat man einen Zylinder platziert und stellt dann fest, dass man eine andere Zylinder-Art benötigt, dann kann man dies nachträglich mit der Einstellung „Zylinder-Art“ ändern:

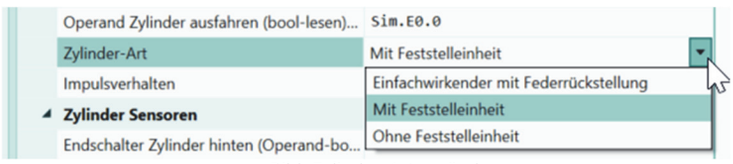


Bild: Zylinder-Art verändern

Hier stehen alle drei Zylinder-Arten zur Auswahl und können bei gezeichneten Zylindern verändert werden.

10.5 Zylinder mit Impulsverhalten

Bei den bisherigen Beispielen war es notwendig, dass der Operand für eine Fahrbewegung dauerhaft den Status 1 hatte. Wechselte der Status zu 0, dann stoppte auch die Bewegung.

In manchen Anlagen geben die Operanden für die Fahrbewegung eines Zylinders nur einen kurzen Impuls ab, d. h. sie wechseln den Status kurzzeitig von 0 nach 1 und wieder nach 0. Dies soll dann ausreichen, damit die Kolbenstange vollständig ausfährt.

Möchte man ein solches Verhalten erreichen, dann ist die Eigenschaft „Impulsverhalten“ des Zylinders zu selektieren.

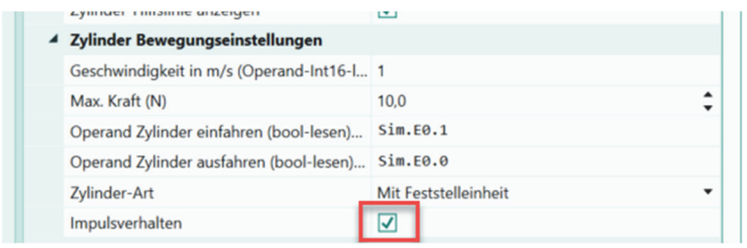


Bild: Impulsverhalten wurde eingeschaltet.

Für das letzte Beispiel mit dem Zylinder inkl. Feststelleinheit bedeutet dies: Wird der Taster „Auf“ nur kurz betätigt, dann bewegt sich die Kolbenstange bis zum oberen Endanschlag. Gleiches gilt für abwärts: Wird der Taster „Ab“ kurzzeitig betätigt, dann fährt die Kolbenstange in die untere Endlage.

Hinweis: Bei der Zylinder-Art „Einfachwirkender mit Federrückstellung“ macht die Einstellung keinen Sinn.

10.6 Geschwindigkeit des Zylinders vorgeben bzw. einstellen

Die Geschwindigkeit, mit der die Kolbenstange des Zylinders sich bewegt, ist einstellbar. Dabei stehen mehrere Vorgabe-Varianten für die Eigenschaft „Geschwindigkeit in m/s“ zur Auswahl:

- Angabe eines konstanten Werts (z. B. 2).
- Angabe eines Bit-Operanden mit einer Bereichsangabe (z. B. 1-3 SimE0.2).
- Angabe eines digitalen Operanden, dessen Inhalt als Geschwindigkeitswert verwendet wird (z. B. Sim.EW2).

Beispiel (Zylinder05):

Im ersten Beispiel ist ein Zylinder mit zwei Geschwindigkeiten zu realisieren: einer normalen, niedrigeren Geschwindigkeit (1 m/s) und einer schnelleren Geschwindigkeit (5 m/s). Die Umschaltung soll über den E0.2 „Schnell“ erfolgen. Der Zylinder ist horizontal angeordnet, die Kolbenstange wird über den E0.0 „Rechts“ nach rechts bewegt, E0.1 „Links“ hat die Bewegung nach links zur Folge.

Die Bewegungseinstellungen des Zylinders sind im folgenden Bild zu sehen:

© MHJ-Software GmbH & Co. KG - PLC-Lab

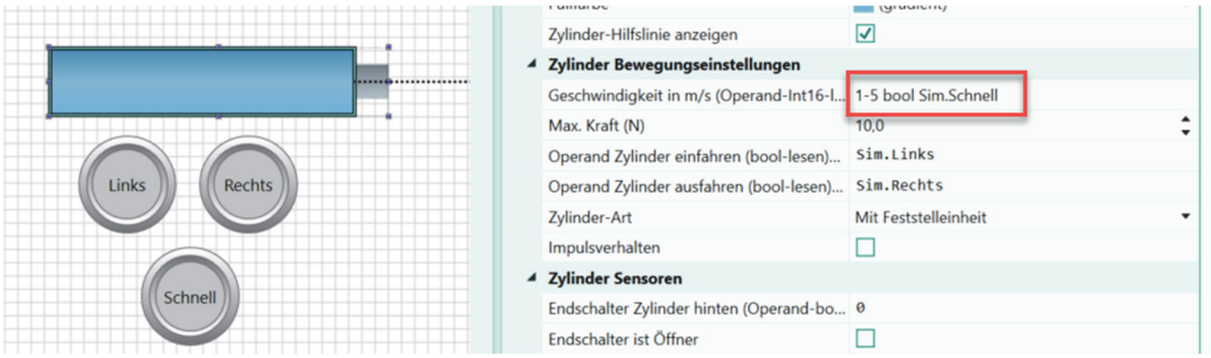


Bild: Geschwindigkeitsvorgabe über Bit-Operand

An der Eigenschaft „Geschwindigkeit in m/s“ ist der Bit-Operand mit dem Symbol „Schnell“ des Sim-Device eingetragen. Zusätzlich ist der Bereich „1-5“ angegeben. Dies hat zur Folge, dass beim Status 0 des Bit-Operanden die Geschwindigkeit 1 m/s vorgegeben wird. Hat der Bit-Operand den Status 1, dann ist die Vorgabe 5 m/s. Die Geschwindigkeit wird dabei mit Hilfe des Schalters „Schnell“ umgeschaltet: Ist er betätigt, dann ist die höhere Geschwindigkeit von 5 m/s vorgegeben.

Beispiel (Zylinder06):

Im zweiten Beispiel wird die Geschwindigkeit mit Hilfe des Wort-Operanden EW2 „ZylGeschw“ vorgegeben. Das EW2 ist dem Sim-Device zugeordnet. Ein Schieberegler verändert den Wert des EW2 im Bereich von 1 bis 10 m/s.

Der horizontale Schieberegler wird aus der Rubrik „Eingabe-Objekte“ entnommen und auf den Bereich 1 bis 10 gesetzt. Als Operand ist „Sim.ZylGeschw“ anzugeben.

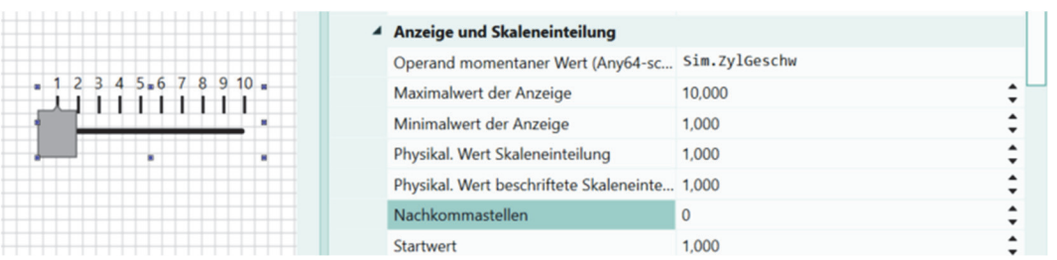


Bild: Einstellungen des Schiebereglers

Am Zylinder ist das Symbol „ZylGeschw“ in der Eigenschaft „Geschwindigkeit in m/s“ einzutragen.

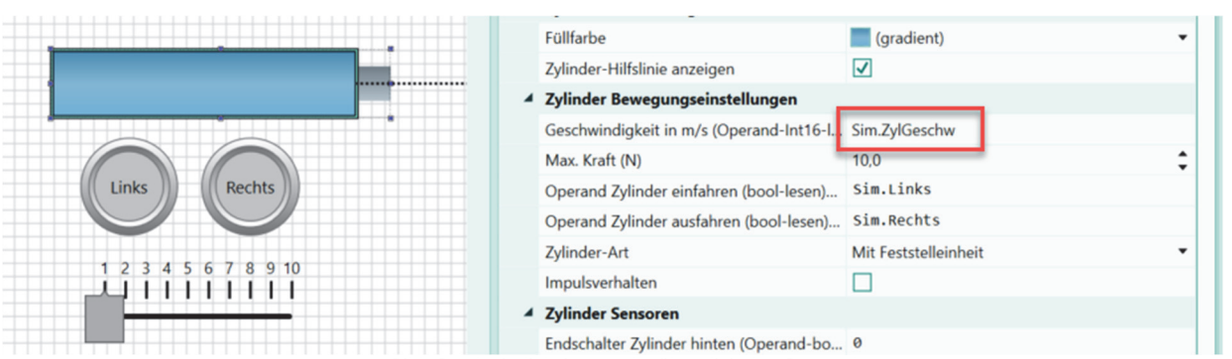


Bild: Geschwindigkeitsvorgabe über digitalen Operanden

Startet man die Simulation, dann kann die Geschwindigkeit der Kolbenstange über den Schieberegler im Bereich 1 bis 10 m/s eingestellt werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

10.7 Die Sensoren des Zylinders

Innerhalb der Rubrik „Zylinder Sensoren“ finden sich zahlreiche Sensoren, welche mit Operanden belegt werden können.

10.7.1 Endlagen-Sensoren des Zylinders

An den Eigenschaften „Endschalter Zylinder hinten“ und „Endschalter Zylinder vorn“ werden meist Bit-Operanden angegeben. Sind diese als Schließer konfiguriert, dann wird der Status 1 in den Operanden geschrieben, sobald die Kolbenstange die jeweilige Endlage erreicht hat. Jeder Endschalter ist als Öffner konfigurierbar, dann wird bei Betätigung der Wert 0 geschrieben.

10.7.2 Sensor der Kolbenstangenposition

Mit Hilfe der Eigenschaft „Sensor Kolbenstangenposition in Pixel“ kann die momentane Position der Kolbenstange in einen digitalen Operanden geschrieben werden. Der Operand muss dabei die notwendige Bit-Breite besitzen, um den Zahlenwert aufnehmen zu können.

10.7.2.1 Verwendung der Kolbenstangenposition ohne Skalierung

Beispiel (Zylinder07):

Von einem horizontal angeordneten Zylinder mit Feststelleinheit soll die Position der Kolbenstange erfasst werden. Die aktuelle Position ist in das EW2 mit dem Symbol „ZylPosition“ zu schreiben. Die Kolbenstange des Zylinders wird über die beiden Operanden E0.0 „Rechts“ und E0.1 „Links“ gesteuert. Diese Operanden sind über Taster beeinflussbar. Der Positionswert soll mit Hilfe eines Text-Labels angezeigt werden.

Für das Beispiel wird ein Text-Label aus der Rubrik „Anzeige-Objekte“ platziert und der Operand EW2 mit dem Symbol „ZylPosition“ per Drag-and-Drop aus der Symboltabelle in das Label eingetragen.

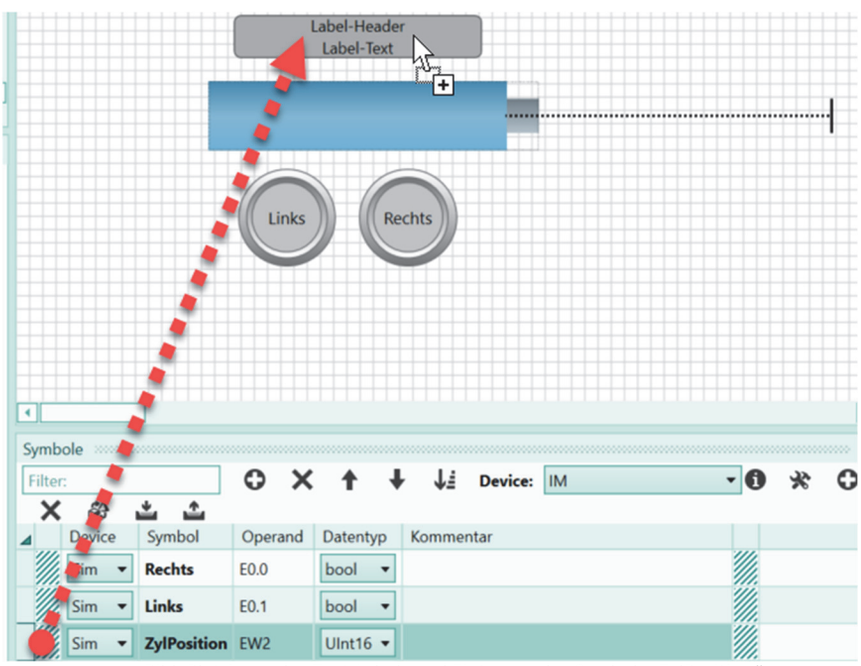


Bild: Platzieren des Text-Labels und Eintragen des Symbols „ZylPosition“

Danach wird der Inhalt des Operanden im Label angezeigt.

Bild: Text-Label nach der Aktion

Im nächsten Schritt ist der Operand „ZylPosition“ in der Eigenschaft „Sensor Kolbenstangenposition in Pixel“ einzutragen, damit die Kolbenstangenposition in ihn geschrieben wird.

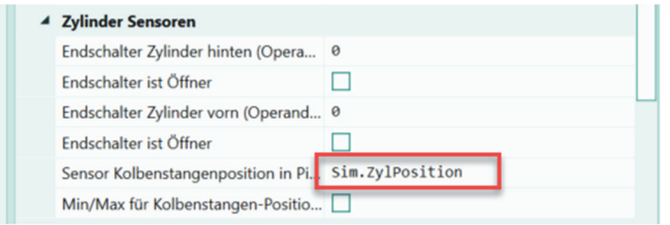


Bild: Angabe des Operanden in der Eigenschaft „Sensor Kolbenstangenposition in Pixel“

Damit ist die Anordnung komplett und die Simulation kann gestartet werden.

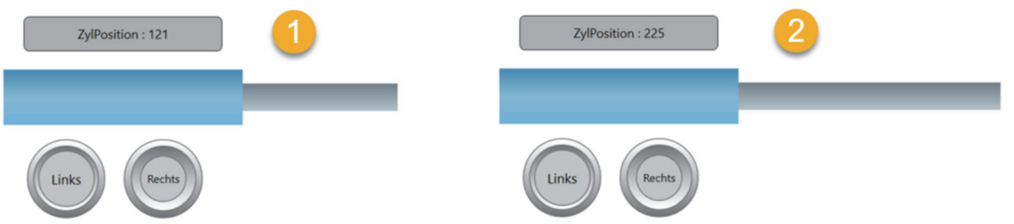


Bild: Simulation der Anordnung

Fährt man die Kolbenstange nach rechts, dann verändert sich der Inhalt von „ZylPosition“, bis die rechte Endlage erreicht ist. Dort wird dann der Wert 225 in den Sensor-Operanden geschrieben. Bewegt man die Kolbenstange nach links, dann verringert sich der Wert, bis an der linken Endlage der Wert 0 erreicht ist.

10.7.2.2 Verwendung der Kolbenstangenposition mit Skalierung

Beispiel (Zylinder08):

Im letzten Beispiel wurde die Zylinderposition direkt in das EW2 geschrieben. Dabei entsprachen die geschriebenen Werte der Pixelposition der Kolbenstange, beginnend mit Position 0 in der hinteren Endlage.

Bei der Simulation einer Anlage werden die Werte von der realen Anlage vorgegeben. Die Werte in der Simulation sollten ihnen möglichst entsprechen, damit die Algorithmen identisch sein können. Dies würde beispielsweise bedeuten: Wenn in einer realen Anlage ein Zylinder die Positionsdaten 0 bis 3000 liefert, dann müsste der Zylinder in der Simulationsanordnung mehr als 3000 Pixel breit sein. Dies ist nicht praktikabel.

Aus diesem Grund ist an den meisten Eigenschaften für digitale Operanden die Angabe eines **Skalierungsbereichs** möglich.

Das letzte Beispiel soll so abgeändert werden, dass in den Operanden für die Kolbenstangenposition die Wert 0 bis 3000 geschrieben werden, da dies dem Wert des Zylinders aus der realen Anlage entspricht.

Diese Änderung ist sehr einfach: An der Eigenschaft „Sensor Kolbenstangenposition in Pixel“ wird vor dem Operanden zusätzlich der Skalierungsbereich „0-3000“ angegeben.

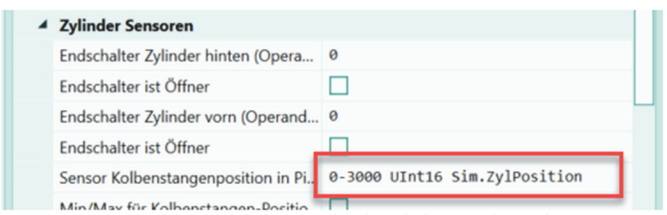


Bild: Zusätzliche Angabe des Skalierungsbereichs

Diese Angabe bewirkt, dass die Kolbenstangenposition in Pixel (im Beispiel 0-225) auf den Bereich 0 bis 3000 skaliert und danach erst in den Operanden geschrieben wird. Damit ist also die Pixel-Länge des Zylinders in der Simulation nicht mehr relevant für die gelieferten Positionswerte.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Damit sind schon alle notwendigen Änderungen vorgenommen. Bei Start der Simulation zeigt sich nun folgendes Bild:

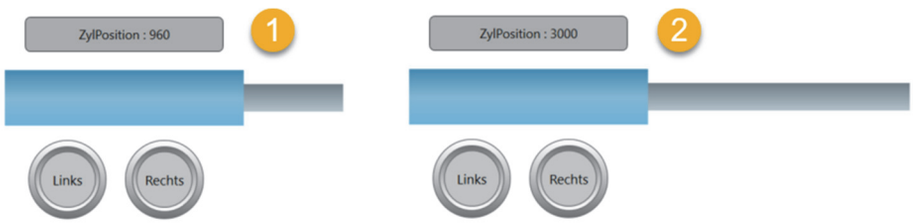


Bild: Simulation der Anordnung mit Skalierung des Positionswerts

Man erkennt, dass die Positionswerte deutlich höher sind, denn diese werden jetzt im Bereich 0 bis 3000 ausgegeben. Hat also die Kolbenstange die rechte Endlage erreicht (Punkt 2), dann wird der Wert 3000 in den Sensor-Operanden „ZylPosition“ geschrieben.

10.8 Objekte an der Kolbenstange eines Zylinders befestigen

In vielen Anordnungen ist es notwendig, dass ein oder mehrere Objekte fest an der Kolbenstange eines Zylinders angebracht werden. In PLC-Lab wird dies mit Hilfe einer Verbindung zu einem Körper realisiert.

Beispiel (Zylinder09):

An die Kolbenstange eines horizontal angeordneten Zylinders mit Feststelleinheit soll ein **Behälter** angebracht werden. Der Behälter besteht aus drei dynamischen Rechtecken. Wie in den letzten Beispielen wird der Zylinder über die beiden Operanden E0.0 „Rechts“ und E0.1 „Links“ gesteuert. Beide Operanden sind dem Sim-Device zugeordnet und ihre Beeinflussung erfolgt über Taster.

Die Anordnung des Zylinders und der Taster ist identisch mit der Anordnung aus den letzten Beispielen. Neu ist der Behälter, bestehend aus drei dynamischen Rechtecken. Die Rechtecke werden aus den geometrischen Formen entnommen und wie folgt auf der Zeichenfläche platziert.

Bild: Form des Behälters, zusammengesetzt aus dynamischen Rechtecken

Würde man die Simulation in diesem Zustand starten, dann würde der Behälter auseinanderfallen, denn er besteht ja aus drei dynamischen Rechtecken. Damit dies nicht passiert, müssen die drei Rechtecke zu einem **Körper** zusammengefasst werden. Dies wird folgendermaßen erreicht:

1. Selektion aller drei Rechtecke.
2. Ausführen der Aktion „Körper bilden“ innerhalb der Aktionslisten-Rubrik „Körper-Gruppe“.
3. Die drei Rechtecke sind zu einer Körper-Gruppe zusammengefasst und werden von einem Rahmen umgeben.

In der folgenden Darstellung sind die Schritte zu sehen.

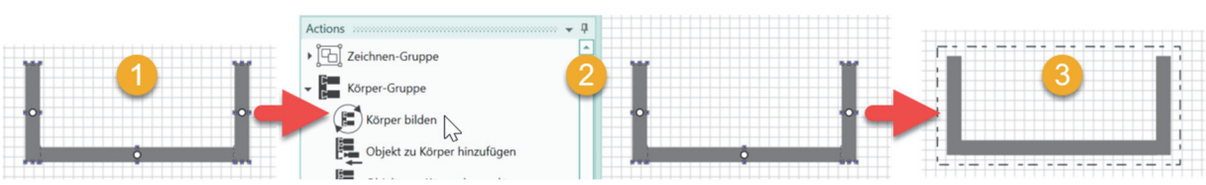


Bild: Schritte zum Bilden der Körper-Gruppe

Unter Punkt 3 wird die gebildete Körper-Gruppe dargestellt. Die Körper-Gruppe ist über einen Rahmen symbolisiert, ähnlich einer Zeichnen-Gruppe. Allerdings unterscheidet sich der Rahmen-Typ, damit die beiden Gruppen-Arten voneinander unterschieden werden können. Das Verhalten der beiden Gruppen-Arten im Zeichnen-Modus ist identisch, was bedeutet, die Körper-Gruppe wird genauso gehandhabt wie eine Zeichnen-Gruppe.

Durch das Bilden der Körper-Gruppe hat sich das Verhalten der drei dynamischen Rechtecke in der Simulation verändert. Sie sind nun unlösbar miteinander verbunden, der Behälter würde sich also nicht mehr in seine Bestandteile zerlegen.

Im nächsten Schritt muss nun der Behälter an der Kolbenstange des Zylinders befestigt werden, dazu wird die Körper-Gruppe zunächst neben dem Zylinder platziert.

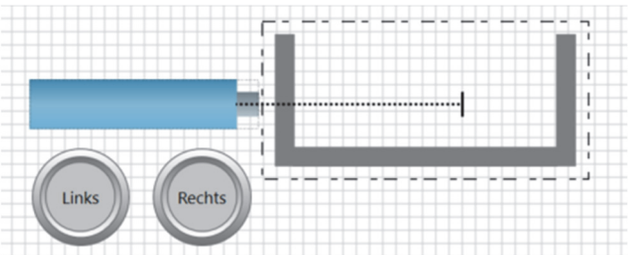


Bild: Behälter wird neben dem Zylinder platziert.

Als nächstes sind die Kolbenstange und der Behälter fest (also unlösbar) miteinander zu verbinden. Dazu ist eine Verbindung zu einem Körper notwendig, welche innerhalb der Verbindungselemente zu finden ist. Dabei sind folgende Schritte notwendig:

1. Selektion der „Verbindung zu einem Körper“ innerhalb der Rubrik „Verbindungselemente bzw. Gelenke“.
2. Platzierung der Maus über der Kolbenstange des Zylinders, Betätigen und Halten der linken Maustaste.
3. Bewegen der Maus über das linke vertikal angeordnete Rechteck des Behälters und loslassen der linken Maustaste.
4. Es wird eine Verbindung zu einem Körper zwischen der Kolbenstange und dem Rechteck gebildet.

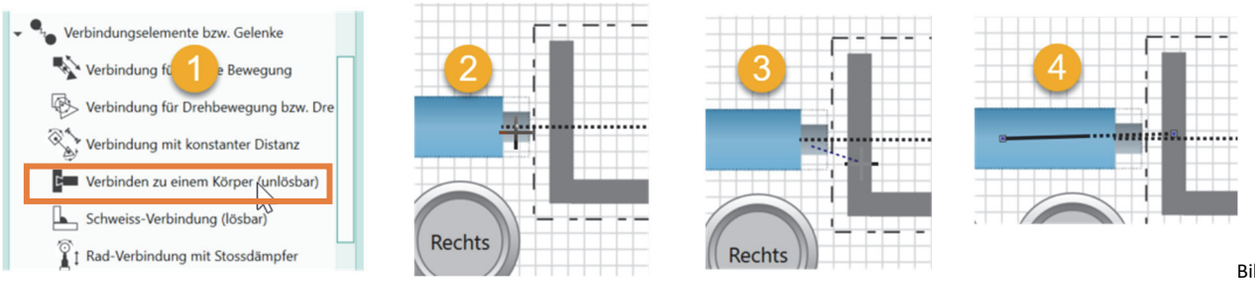


Bild: Schritte für die Verbindung zu einem Körper

Damit ist die Anordnung fertiggestellt und kann simuliert werden.

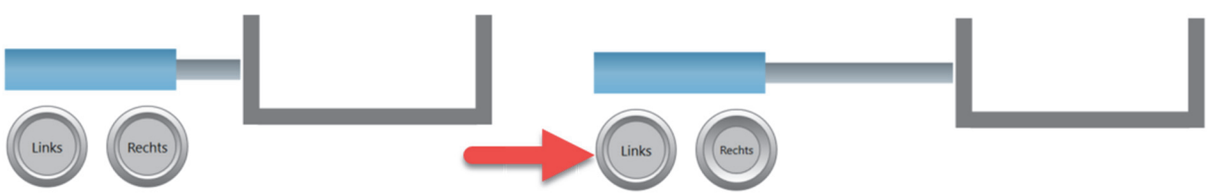


Bild: Simulation der Anordnung

Bei Bewegung der Kolbenstange des Zylinders mit Hilfe der Taster wird der Behälter ebenfalls bewegt, denn er ist ja fest mit der Kolbenstange verbunden.

Fazit: Das Beispiel hat gezeigt, wie man Objekte fest an die Kolbenstange eines Zylinders anbringen kann. Dabei kommt die Verbindung zu einem Körper zum Einsatz. Beim Ziehen dieser Verbindung ist genau darauf zu achten, dass die Verbindung bei der Kolbenstange des Zylinders begonnen wird und bei dem Objekt endet, welches angebunden werden soll. Auf diese Art der Verbindung wird noch in einem eigenen Abschnitt explizit eingegangen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

10.9 Zylinder hintereinander anbringen mit Hilfe von Verbindungen zu einem Körper

In einigen Anordnungen ist es notwendig Zylinder zu kaskadieren, sie also hintereinander anzuordnen. Da die Zylinderkörper standardmäßig statisch sind, muss dies bei dem Zylinder, welcher an der Kolbenstange eines anderen befestigt wird, geändert werden.

Dies ist mit der Eigenschaft „Physikal. Typ des Objekts“ innerhalb der Rubrik „Zylinderkörper Physikeinstellungen“ möglich. Hier ist als Typ „Dynamic“ zu selektieren.

Beispiel (Zylinder10):

An einen horizontal angeordneten Zylinder ist ein vertikal angeordneter zweiter Zylinder zu befestigen. Beide Zylinder verfügen über eine Feststelleinheit. Der Zylinder 1 wird über die beiden Operanden E0.0 „Links“ und E0.1 „Rechts“ aus- bzw. eingefahren. Zylinder 2 ist mit Hilfe der beiden Operanden E0.2 „Auf“ und E0.3 „Ab“ beeinflussbar. Die Operanden sind dem Sim-Device zugeordnet.

Im ersten Schritt wird der horizontale Zylinder platziert und die beiden Operanden eingetragen. Des Weiteren sind die beiden Schalter zu zeichnen und mit den Operanden zu verknüpfen.

Bild: Zylinder 1 mit den beiden Schalter-Objekten

Nun folgt der zweite Zylinder, dieser wird an der Kolbenstange des ersten angeordnet (linkes Bild):

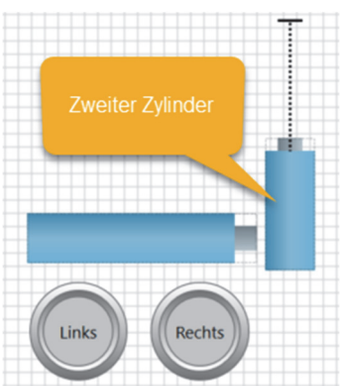


Bild: Anordnung des vertikalen Zylinders

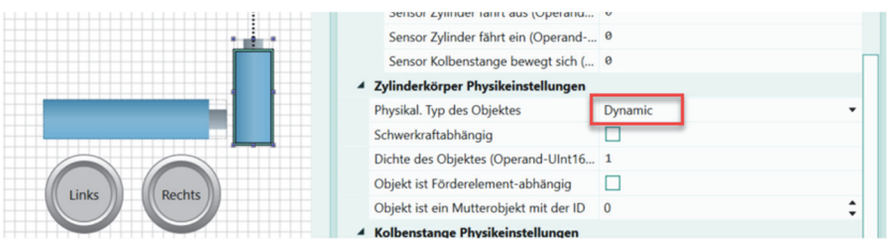


Bild: Einstellung des physikalischen Typs des Zylinderkörpers

Beim zweiten Zylinder ist der physikalische Typ anzupassen: Es muss der Typ „Dynamic“ eingestellt werden (rechtes Bild). Nur damit ist es möglich, den Zylinderkörper in der Simulation zu verschieben.

Im zweiten Schritt sind die Operanden „Auf“ und „Ab“ in den vertikalen Zylinder einzutragen und die beiden Schalter-Objekt für deren Umschaltung zu platzieren. Die Anordnung hat somit folgendes Aussehen:

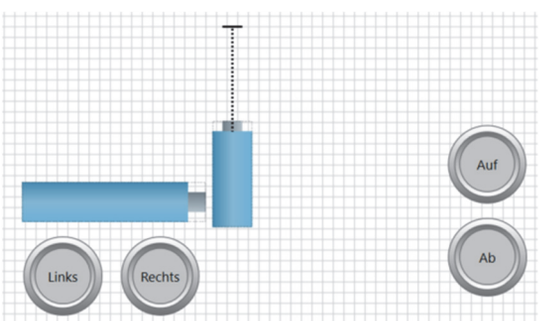


Bild: Anordnung nach dem zweiten Schritt

Im letzten Schritt muss der vertikale Zylinder an der Kolbenstange des horizontalen Zylinders befestigt werden. Dazu wird die im letzten Kapitel bereits verwendete „Verbindung zu einem Körper“ benötigt. Ausgangspunkt der Verbindung ist die Kolbenstange des ersten Zylinders.

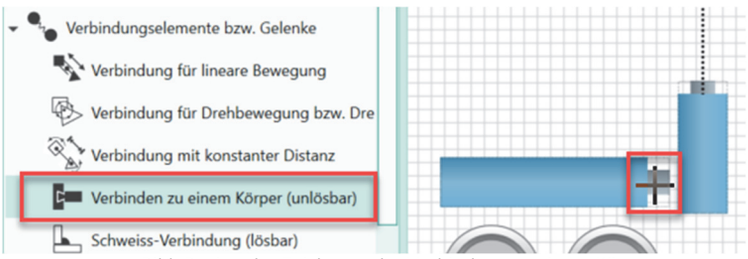


Bild: Beginn des Zeichnens der Verbindung zu einem Körper

Der Endpunkt muss sich auf dem Zylinderkörper des zweiten befinden.

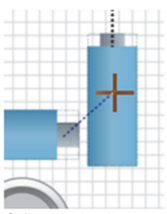


Bild: Der Endpunkt auf dem Zylinderkörper des vertikalen Zylinders

Nach dieser Aktion ist der zweite Zylinder fest an der Kolbenstange des ersten angebracht.

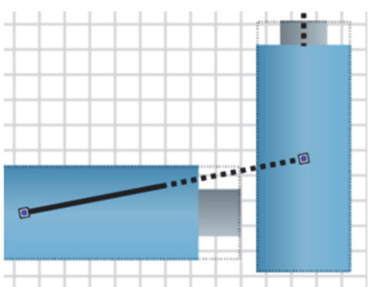


Bild: Verbindung zu einem Körper ist fertiggestellt.

Nun kann die Anordnung simuliert werden.

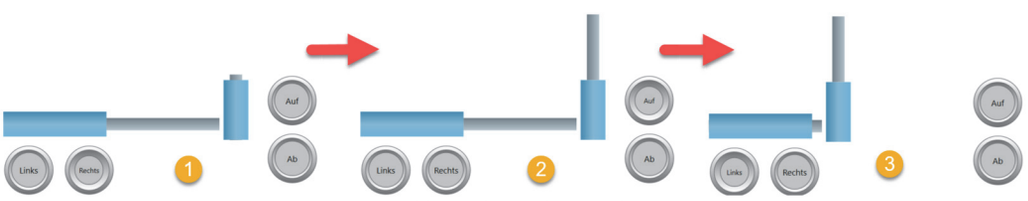


Bild: Simulation der Anordnung

Im Bild (Punkt 1) wurde die Kolbenstange des horizontalen Zylinders ausgefahren. Dabei wird der vertikale Zylinder ebenfalls nach rechts bewegt. Bei Punkt 2 bewegt sich die Kolbenstange des vertikalen Zylinders nach oben, initiiert durch den Schalter „Auf“. In Punkt 3 wird die Kolbenstange des horizontalen Zylinders eingefahren. Der fest an der Kolbenstange angebrachte Zylinder bewegt sich somit ebenfalls nach links.

An die Kolbenstange des vertikalen Zylinders könnte nun ein weiterer Zylinder angebracht werden. Auch dessen physikalischer Typ des Zylinderkörpers ist dabei entsprechend anzupassen.

Fazit: Das Beispiel hat gezeigt, wie man Zylinder kaskadieren kann. Dabei werden die Zylinderkörper mit Hilfe der Verbindung zu einem Körper an die Kolbenstange des jeweils anderen Zylinders angebracht. Beim Ziehen der Verbindung ist darauf zu achten, dass als Startpunkt die Kolbenstange verwendet wird und als Endpunkt der Zylinderkörper des angebundenen Zylinders. Des Weiteren muss der physikalische Typ des angebundenen Zylinders auf den Typ „Dynamic“ gesetzt werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

11 Förderelemente

Sollen in einer Anordnung Objekte von links nach rechts bzw. von oben nach unten **bewegt** werden, dann können Förderelemente aus der gleichnamigen Rubrik zum Einsatz kommen.

Die Förderelemente beeinflussen förderelementabhängige Objekte, sobald sich diese in ihrem **Einflussbereich** befinden. Dabei stehen **drei** verschiedene Förderelemente zur Auswahl:

- **Förderband**:
  Dieses Förderelement kann sowohl in der Seitenansicht als auch in der Draufsicht verwendet werden. Sobald sich ein förderbandabhängiges Objekt im Einflussbereich des Förderbands befindet, wird das Objekt in seiner Eigendynamik eingeschränkt. Dies ist vergleichbar mit Reibung des Bands mit dem Objekt. Verantwortlich dafür sind die beiden Eigenschaften „Dämpfung für lineare Bewegung“ und „Dämpfung für Rotations-Bewegung“. Möchten man den Einfluss des Bands auf das beförderte Objekt verstärken, dann kann man den Wert dieser beiden Eigenschaften erhöhen.
- **Förderband mit Führungsschiene**:
  Das Förderband mit Führungsschiene wird meist in der Draufsicht verwendet. Es besitzt eine in der Mitte des Objekts angebrachte Führungsschiene (bzw. Führungsrille), in der die beförderten Objekte gehalten werden. Ein zu beförderndes Objekt wird erst beeinflusst, wenn sich sein Mittelpunkt auf der Führungsschiene befindet. Diese Art des Förderbands ist besonders geeignet für schienenähnliche Systeme bzw. für Anordnungen, bei denen die Objekte sich auf einer Art Kreisbahn bewegen.
- **Luftstrom**:
  Wie der Name es vermuten lässt, wird der Luftstrom eingesetzt, wenn der Einfluss des Förderelements auf das zu befördernde Objekt nicht so stark ausfallen soll. Beispiele sind das Ablenken von Objekten, das Abblasen von Spänen oder ähnliches.

Alle drei Förderelement-Arten entstammen dem gleichen Basis-Typ. Diese wurden nur über die Voreinstellung der Eigenschaften auf spezielle Einsatzgebiete ausgelegt. Dies bedeutet, durch Verändern dieser Eigenschaften (Rubrik „Förderelement Einstellungen“) kann man das Verhalten den eigenen Wünschen entsprechend anpassen.

11.1 Verwendung des Förderelements „Förderband“ in der Seitenansicht

Beispiel (Förderelemente01):

Zunächst soll das Förderband in der Seitenansicht vorgestellt werden. Dabei muss das Förderband einen Unterbau erhalten, damit die zu befördernden Objekte nicht hindurchfallen. Im folgenden Bild ist dies zu sehen:

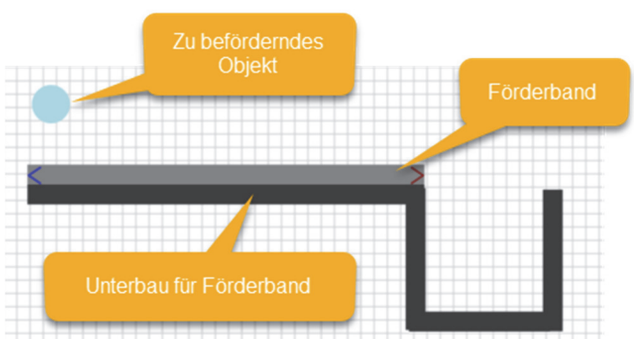


Bild: Förderband in der Seitenansicht

Der Unterbau ist ein feststehendes Rechteck. Das zu befördernde Objekt ist ein dynamischer Kreis, der beim Einschalten der Simulation nach unten fällt (wg. der Schwerkraftabhängigkeit). Beim Förderband sind zwei **Pfeile** eingezeichnet. Der rote Pfeil gibt die Richtung nach rechts an, der blaue Pfeil entsprechend die Bewegung nach links. Im Förderband ist in der Eigenschaft „Operand Bewegung nach rechts“ die Konstante 1 eingetragen, was bedeutet, dass das Band dauerhaft eingeschaltet ist und die zu befördernden Objekte nach rechts bewegt werden.

Somit zeigt sich beim Start der Simulation folgende Situation.

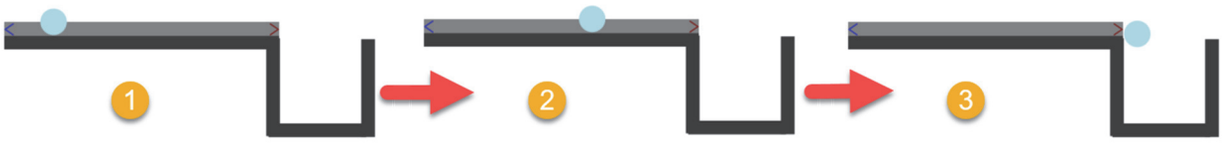


Bild: Start der Simulation

Zunächst fällt der Kreis nach unten auf das Förderband und wird dabei vom Unterbau des Förderbands aufgehalten. Dann wird der Kreis vom Förderband nach rechts bewegt, bis das Ende des Förderbands erreicht ist. Danach fällt das Objekt in den Behälter.

11.2 Verwendung des Förderelements „Förderband“ in der Draufsicht

Beispiel (Förderelemente02):

Im folgenden Beispiel wird ein Förderband aus der Vogelperspektive angeordnet. Bei dem zu befördernden dynamischen Rechteck wird die Schwerkraftabhängigkeit abgeschaltet und eine Dämpfung für die lineare Bewegung sowie für die Rotationsbewegung gesetzt.

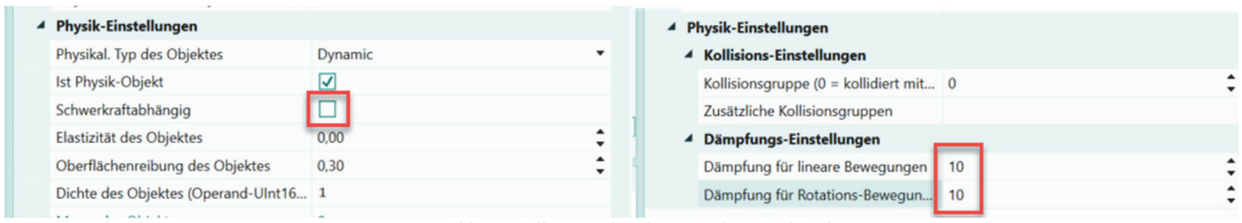


Bild: Einstellungen des dynamischen Rechtecks

Durch die Dämpfungs-Einstellung wird erreicht, dass sich das Objekt außerhalb des Einflussbereichs des Förderbands **nicht** wie in einem schwerelosen Raum bewegt.

Am Band ist ein Zylinder mit Federrückstellung angeordnet, mit dem das Rechteck vom Hauptband auf ein darüber angeordnetes Nebenband verschoben werden kann.

Der Operand E0.0 „BandNachRechts“ schaltet die Förderbewegung des Hauptbands nach rechts. Der Zylinder wird über den E0.1 „ZylinderVor“ gesteuert. Das Nebenband ist immer eingeschaltet, hier ist in der Eigenschaft „Operand Bewegung nach rechts“ die Konstante 1 eingetragen. Die Operanden sind dem Sim-Device zugeordnet und werden über Taster beeinflusst.

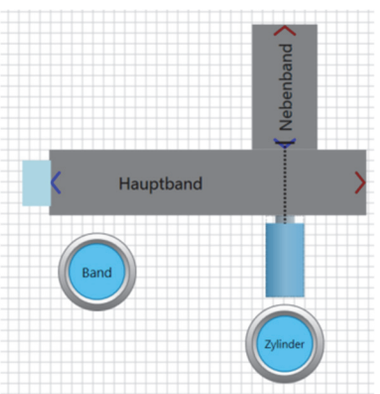


Bild: Anordnung für dieses Beispiel

Nun wird die Simulation gestartet und mit Hilfe des Band-Tasters das Hauptband eingeschaltet. Dies hat zur Folge, dass das Rechteck nach rechts befördert wird. Sobald das Rechteck den Zylinder erreicht hat, wird der Taster losgelassen, das Förderband stoppt und das Rechteck bleibt an der Stelle liegen. Daraufhin wird der Taster „Zylinder“ betätigt. Somit schiebt der Zylinder das Objekt in Richtung des Nebenbands. Sobald das Rechteck in den Einflussbereich des Nebenbands kommt, wird es nach oben transportiert.

Nachfolgend sind diese Situationen zu sehen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

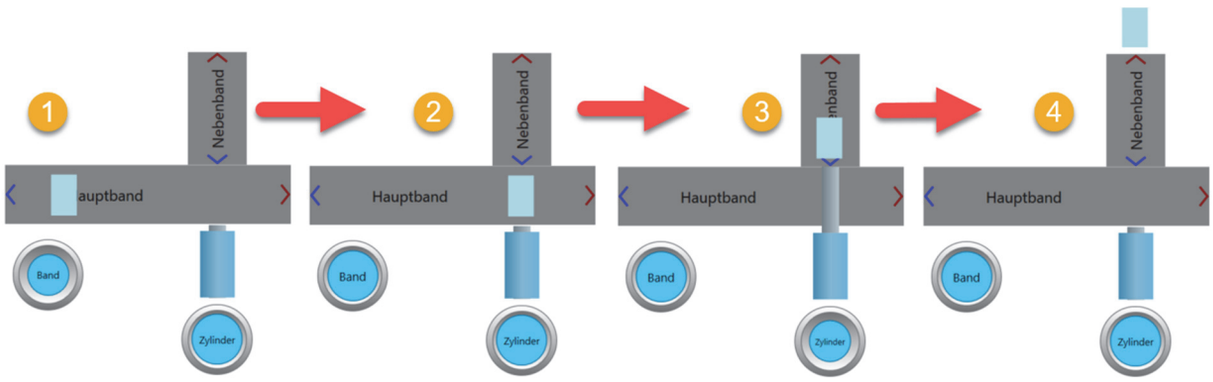


Bild: Stationen bei der Simulation der Anordnung

Fazit: Wird ein Förderband in der **Draufsicht** verwendet, dann ist meist die Schwerkraftabhängigkeit der zu befördernden Objekte **abzuschalten**. Des Weiteren sollte man bei den zu befördernden Objekten eine **Dämpfung** für die lineare Bewegung und die Rotationsbewegung einstellen. Der Wert 10 als Dämpfungsgrad ist dabei eine gute Ausgangsbasis.

11.3 Bewegen eines Förderelements in der Seitenansicht

Förderbänder sind in der Standardeinstellung statische Physik-Objekte. Möchte man ein Förderband dynamisieren, dann ist der Physik-Typ in „Dynamic“ abzuändern. Dies ist beispielsweise notwendig, wenn man ein Förderband in der vertikalen bewegen möchte, wie dies bei einem Hubtisch der Fall ist.

Beispiel (Förderelemente03):

Objekte werden von Band 1 auf das Band 2 transportiert. Band 2 wird mit Hilfe eines Zylinders (mit Federrückstellung) **angehoben** und die zu befördernden Objekte in der **zweiten Ebene** über das Band 3 abtransportiert. Folgende Operanden werden verwendet:

Operand	Symbol	Funktion
E0.0	Band1Ein	Band 1: Operand für die Bewegung nach rechts.
E0.1	Band2Ein	Band 2: Operand für die Bewegung nach rechts.
E0.2	ZylinderNachOben	Zylinder ausfahren.
E0.3	Band3Ein	Band 3: Operand für die Bewegung nach rechts.

Alle Operanden sind dem Sim-Device zugehörig.

Die Anordnung wird in der Seitenansicht gezeichnet. Die beiden Bänder 1 und 3 erhalten einen Unterbau, wie in den Beispielen zuvor. Die Operanden „Band1Ein“ bzw. „Band3Ein“ werden jeweils in der Eigenschaft „Operand Bewegung nach rechts“ von Band 1 und 3 eingetragen.

Das zweite Band ist ebenfalls mit einem Unterbau zu versehen. Dieser muss allerdings (ebenso wie das Förderband selbst) die Physik-Art „Dynamic“ erhalten, denn das Band 2 und dessen Unterbau sollen ja mit Hilfe eines Zylinders bewegt werden.

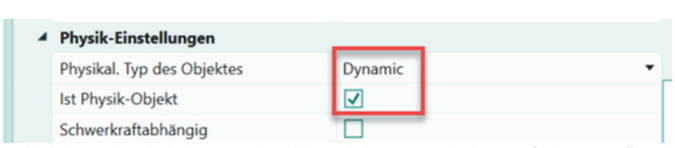


Bild: Den physikal. Typ von Band 2 und dessen Unterbau auf „Dynamic“ ändern

In der Eigenschaft „Operand Bewegung nach rechts“ von Band 2 ist der Operand mit dem Symbol „Band2Ein“ einzutragen. Der Zylinder fährt aus, sobald „ZylinderNachOben“ den Status 1 besitzt, weshalb der Operand in der entsprechenden Eigenschaft des Zylinders anzugeben ist.

Damit der Zylinder das Band 2 und dessen Unterbau nach **oben** und **unten** bewegt, sind **zwei Schritte** erforderlich.

1. Bilden einer Körper-Gruppe aus Band 2 und dessen Unterbau: Dabei werden die beiden Objekte selektiert und die Aktion „Körper bilden“ innerhalb der Rubrik „Körper-Gruppe“ ausgeführt. Als Folge wird ein Rahmen um die beiden Objekte angezeigt.
2. Herstellen einer Verbindung zu einem Körper zwischen der Kolbenstange des Zylinders und dem Unterbau des Bands. Dazu wird das Verbindungselement „Verbinden zu einem Körper“ selektiert und dann die Verbindung beginnend an der Kolbenstange des Zylinders zum Unterbau des Bands gezogen.

Nach diesen beiden Aktionen haben der Zylinder und das Band mit seinem Unterbau folgendes Aussehen:

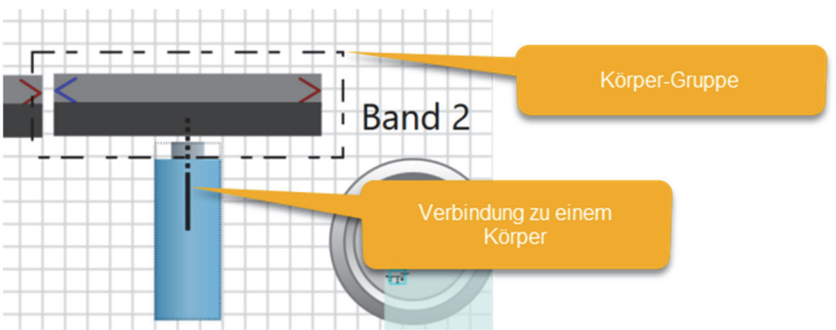


Bild: Zylinder, Band 2 und dessen Unterbau nach den beiden Aktionen

Alle Operanden werden über Schalter-Objekte beeinflusst, wobei nur der Schalter für „ZylinderNachOben“ ein Schalter ist, die anderen werden als Taster konfiguriert.

Das zu befördernde Objekte ist ein dynamisches Rechteck, welches auf der linken Seite von Band 1 zu platzieren ist. Damit präsentiert sich die Anordnung wie folgt:

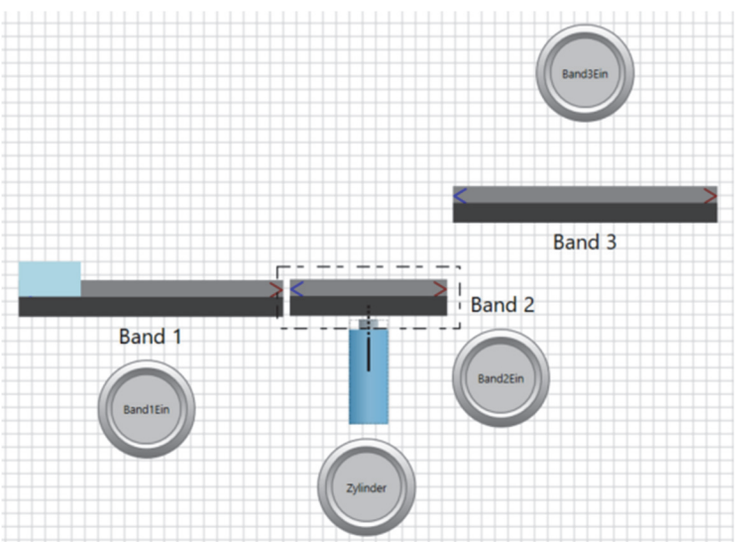


Bild: Anordnung für dieses Beispiel

Nach dem Start der Simulation wird der Taster für die Bewegung nach rechts von **Band 1** betätigt. Damit bewegt sich das zu befördernde Rechteck nach rechts in Richtung von **Band 2**. Hat das Objekt das zweite Band erreicht, dann soll es bis zur Mitte des Bands befördert werden. Dies erreicht man durch Betätigung des Tasters für den Operanden „Band2Ein“. Nun kann der Schalter für den Zylinder betätigt werden. Als Folge fährt der Zylinder aus und das Band 2, dessen Unterbau und das zu befördernde Objekt werden zusammen nach oben bewegt. Hat der Zylinder die obere Endlage erreicht, dann wird Band 2 wieder eingeschaltet und somit das Rechteck in Richtung von **Band 3** transportiert. Dieses kann dann das Rechteck weiter nach rechts befördern.

Wird der Schalter des Zylinders wieder in seine Ruhestellung gebracht, dann bewegt sich der Zylinder mitsamt dem zweiten Band nach unten in seine Grundstellung. Im folgenden Bild sind diese Situationen dargestellt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

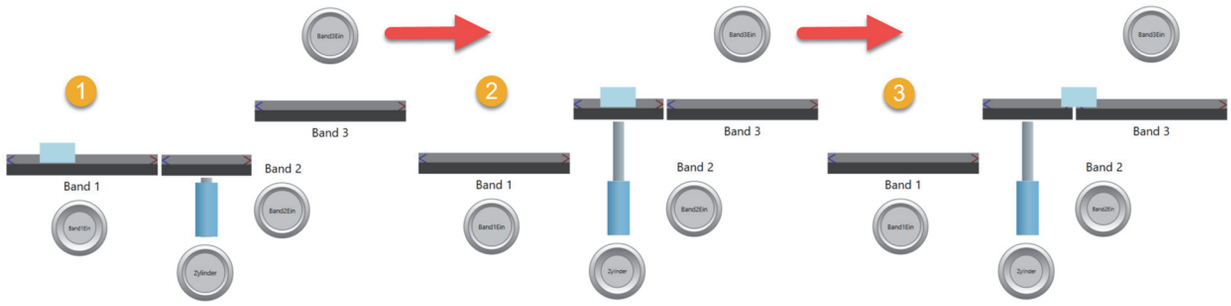


Bild: Darstellung der oben beschriebenen Situationen

Fazit: Soll ein Förderband bewegt werden, dann muss man dessen physikalischen Typ von „**Static**“ auf „**Dynamic**“ abändern. Danach hat das Förderband ein Verhalten ähnlich dem eines dynamischen Geometrie-Objekts.

11.4 Förderband mit Nebenband am Ende

Beispiel (Förderelemente04):

Nun eine Darstellung, bei der zwei Förderbänder in der Draufsicht verwendet werden. Das erste Band ist horizontal angebracht, das zweite am rechten Ende in der Vertikalen. Dabei überlappen sie sich.

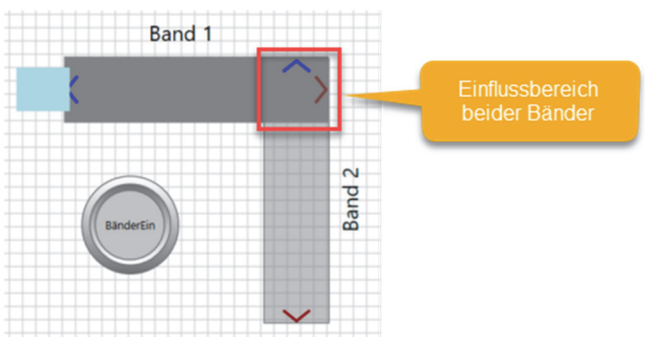


Bild: Anordnung für dieses Beispiel

Beide Bänder werden in der Bewegung nach rechts eingeschaltet, sobald der Operand E0.0 „BänderEin“ des Sim-Device den Status 1 besitzt. Der Operand wird über ein als Taster konfiguriertes Schalter-Objekt beeinflusst. Das zu befördernde Rechteck ist **dynamisch**, allerdings wurde die **Schwerkraftabhängigkeit abgeschaltet** und jeweils ein **Dämpfungswert** von 10 für die linearen und rotierenden Bewegungen gesetzt.

Im oberen Bild ist der Bereich gekennzeichnet, in welchem beide Bänder das zu befördernde Objekt beeinflussen.

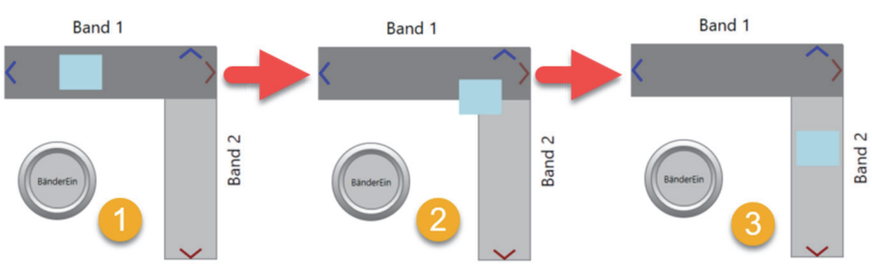


Bild: Darstellung der versch. Teilbereiche bei der Simulation

Sobald das Objekt diesen Bereich erreicht (Punkt 2), wird es sowohl von Band 1 als auch von Band 2 befördert.
Dies hat zur Folge, dass das Objekt sowohl in horizontaler als auch in vertikaler Richtung bewegt wird.

Erst wenn sich das Objekt nur noch im Bereich des vertikal angeordneten Förderbands befindet (Punkt 3), bewegt es sich auch nur noch vertikal nach unten.

11.5 Die Eigenschaft „Ist exklusives Förderelement“

Beispiel (Förderelemente05):

Ist ein Förderband als „exklusiv“ gekennzeichnet, dann ist es das einzige Förderband, das ein von ihm befördertes Objekt beeinflusst. Wird in der letzten Anordnung das horizontale Förderband exklusiv gesetzt, dann kann das beförderte Objekt nicht vom vertikalen Band erfasst und abgelenkt werden.

Allerdings gilt die Exklusivität nur so lange, wie einer der Operanden für die Bewegung (hier: nach rechts oder links) den Status 1 besitzt. Wird also das exklusive Band abgeschaltet, dann kann ein anderes Band ebenfalls das Objekt befördern.

Um dies zu demonstrieren, wird das letzte Beispiel etwas abgeändert. Im **Band 2** wird der Operand E0.2 „Band2Ein“ des Sim-Device in die Eigenschaft „Operand Bewegung nach rechts“ eingetragen.

Bild: **Förderband 1** wird als exklusives Förderelement gesetzt.

Des Weiteren ist ein zweiter Schalter zu platzieren, der diesen Operanden verändert. Der bereits vorhandene Taster für „Band1Ein“ ist ebenfalls als Schalter zu konfigurieren. Zuletzt wird das **Band 1** als **exklusives** Förderelement eingestellt.

Nun zeigt sich bei der Simulation folgendes Verhalten:

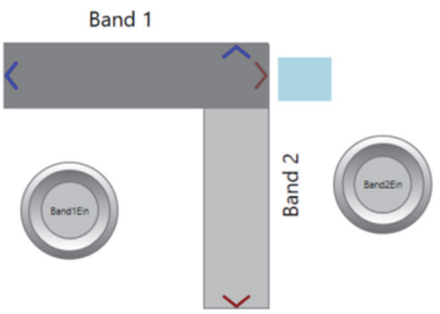
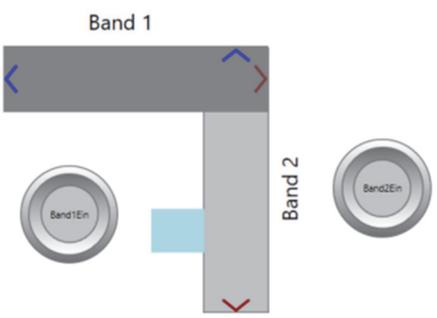


Bild: Das Verhalten, wenn Band 1 als exklusives Förderelement eingestellt ist

Das **Band 2** hat keinen Einfluss mehr auf das zu befördernde Objekt, da Band 1 exklusiv die Beförderung übernimmt, solange dessen Operand für die Bewegung den Status 1 besitzt. Aus diesem Grund wird das Rechteck bis an das Ende von Band 1 bewegt, ohne dass es durch Band 2 nach unten befördert wird.

Im nächsten Schritt wird das Band 2 ebenfalls als exklusiv eingestellt (Bild rechts):



Sobald das zu befördernde Objekt den Bereich des vertikalen Bands erreicht, wird es von ihm erfasst und ohne weitere Ablenkung des horizontalen Bands nach unten bewegt.

Dies ist darin begründet, dass das letzte exklusive Band, mit dem ein Objekt kollidiert (und dessen Bewegungsoperand den Status 1 besitzt), die höchste Priorität hat.

Fazit: Mit der Eigenschaft „Ist exklusives Förderelement“ lässt sich das Verhalten von aneinander gereihten Förderelementen zusätzlich beeinflussen. Sobald einer der Bewegungsoperanden eines exklusiven Förderelements den Status 1 besitzt, übt es seine Exklusivität aus. Sind zwei exklusive Förderelemente hintereinander angebracht, dann hat das zuletzt von dem zu beförderten Objekt berührte Förderelement die höchste Priorität.

© MHJ-Software GmbH & Co. KG - PLC-Lab

11.6 Förderbänder mit Führungsschiene

Förderbänder dieser Art bewegen die Objekte in einer mittigen **Führungsschiene** oder Rille. Damit kann auch erreicht werden, dass sich die zu befördernden Objekte auf einer Kreisbahn bewegen.

Die Objekte werden erst von diesem Förderelement bewegt, wenn sich der **Mittelpunkt** des Objekts auf der Führungsschiene befindet.

Die Förderbänder können **aneinandergereiht** werden, wobei sich die Führungsschienen an den Enden berühren sollten, damit eine Übergabe stattfinden kann.

Beispiel (Förderelemente06):

Mit Hilfe von Förderelementen des Typs „Förderband mit Führungsschiene“ soll ein Halbkreis gebildet werden. Die Bewegung nach rechts soll über den Operanden E0.0 „BandEin“ des Sim-Device erfolgen. Die Förderelemente sind jeweils um **30° zueinander** zu rotieren und aneinanderzureihen. Damit ergibt sich folgende Anordnung:

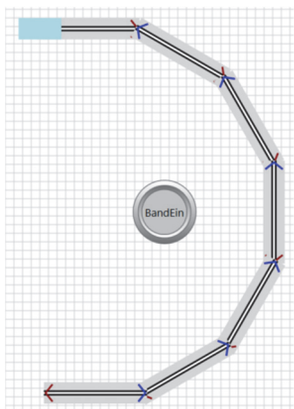


Bild: Anordnung für dieses Beispiel

Die Führungsschienen gehen direkt ineinander über. Sollten die intelligenten Führungslinien bei der Platzierung störend sein, dann können sie mit Hilfe der Aktion „Intelligente Führungslinien anzeigen“ abgeschaltet werden. Am oberen Förderelement wird das zu befördernde Rechteck platziert. Dabei ist zu beachten, dass der Mittelpunkt des Objekts die Führungsschiene berührt, denn erst dann wird das Objekt befördert. Beim Start der Simulation zeigt sich folgendes Verhalten:

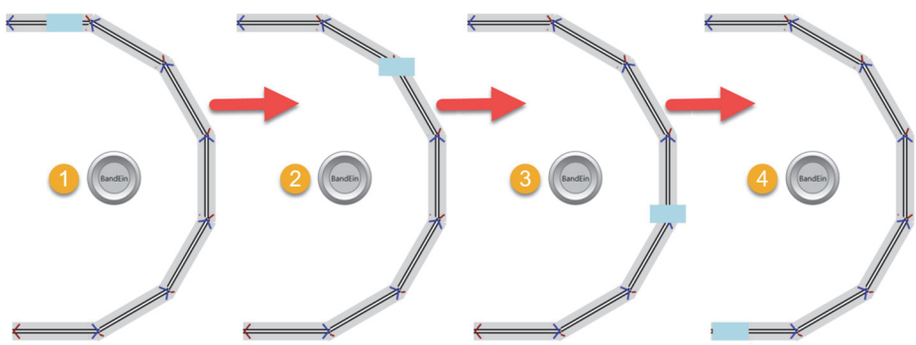


Bild: Stationen bei der Simulation

Man erkennt, dass sich der Mittelpunkt des Rechtecks immer auf der Führungsschiene befindet. Das jeweilige Förderelement versucht permanent diesen Zustand herzustellen. Allerdings ist die Kraft, um den Mittelpunkt des zu befördernden Objekts in der Führungsschiene zu halten, nicht sehr hoch. Es kann somit durchaus passieren, dass durch Fliehkräfte oder anderes das Objekt aus der Führungsschiene gerissen und somit nicht weiterbewegt wird.

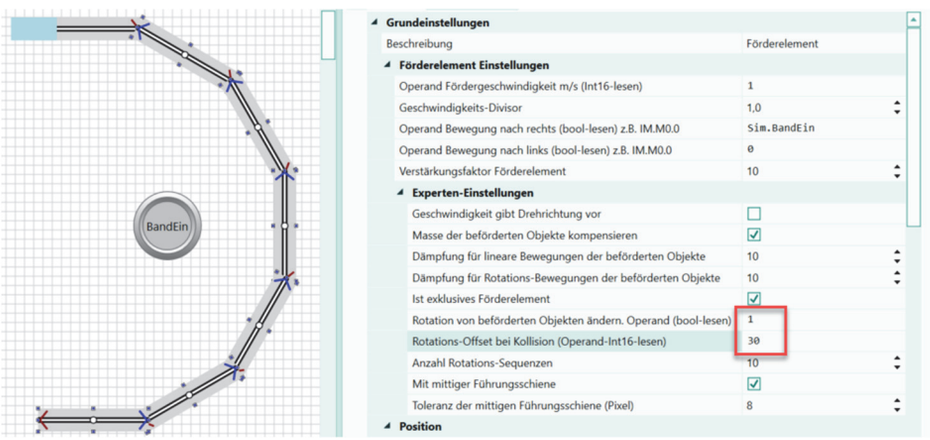
11.6.1 Rotation der beförderten Objekte ändern

Beispiel (Förderelemente07):

Beim letzten Beispiel wurde ein Rechteck von Förderelementen bewegt, welche zusammen einen Halbkreis bildeten. Dabei wurde die **Rotation** des beförderten Objekts **nicht** verändert. Dieses Verhalten ist bei vielen Anordnungen so gewünscht. Es kann aber auch sein, dass das beförderte Objekt seine Rotationslage verändern soll, sobald es von einem Förderelement bewegt wird.

Um das Verhalten in einem solchen Fall zu demonstrieren, werden die Förderelement, welche gegenüber dem vorhergehenden Förderelement um 30° verdreht sind, weiter geändert. Und zwar dahingehend, dass die beiden Eigenschaften „**Rotation von beförderten Objekten ändern**“ und „**Rotations-Offset bei Kollision**“ wie folgt eingestellt werden:

Bild: Ändern der beiden Eigenschaften bei allen rotierten Förderelementen



In der Eigenschaft „Rotation von beförderten Objekten ändern“ wird die Konstante 1 angegeben. Somit ist die Rotation des zu befördernden Objekts immer aktiv. Die Eigenschaft „Rotations-Offset bei Kollision“ erhält den Wert 30, somit werden alle beförderten Objekte beim ersten Kontakt mit dem Förderelement um 30° im Uhrzeigersinn rotiert.

Diese Änderungen haben folgende **Auswirkungen**:

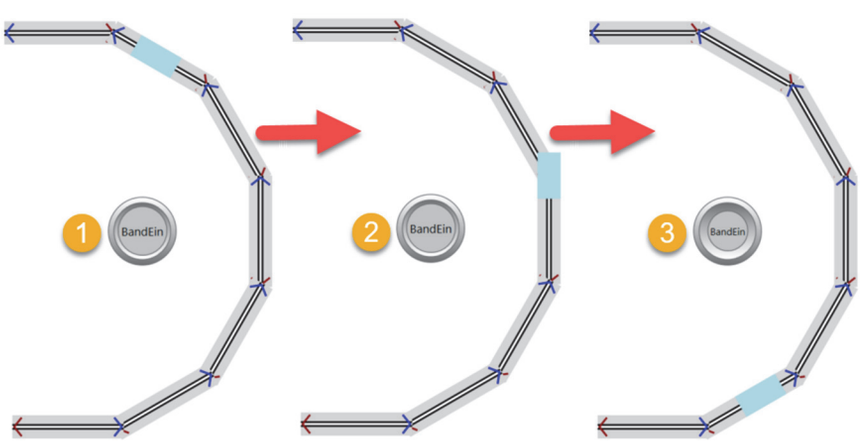


Bild: Verhalten nach den Änderungen

Im Bild ist zu sehen, dass das beförderte Objekt entsprechend dem aktiven Förderelement rotiert wird. Die Rotation erfolgt dabei in dem Moment, wo der Mittelpunkt des zu befördernden Objekts die Führungsschiene zum ersten Mal berührt.

Die hier verwendeten Eigenschaften stehen auch bei den anderen Förderelement-Typen zur Verfügung.

© MHJ-Software GmbH & Co. KG - PLC-Lab

11.7 Der Luftstrom

Wie der Name es andeutet, verwendet man dieses Förderelement, um luftstrom-ähnliches Verhalten zu simulieren.

Beispiel (LuftStrom01):

Ein Förderelement vom Typ „Luftstrom“ wird horizontal platziert. Die Eigenschaft „Operand Bewegung nach rechts“ ist mit Hilfe der Konstanten 1 dauerhaft eingeschaltet. Über dem Luftstrom befindet sich eine Plattform, auf der zwei Rechtecke abgelegt sind. Das blaue Rechteck verfügt dabei über eine Dichte von 1, im grauen Rechteck wurde eine Dichte von 5 eingestellt. Somit ist die Masse des grauen Rechtecks 5-mal höher. Im Luftstrom ist ein Verstärkungsfaktor von 1 eingestellt. Die Anordnung hat das im folgenden Bild dargestellte Aussehen:

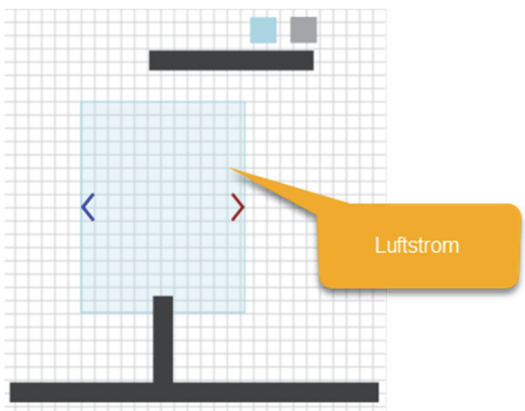


Bild: Anordnung für dieses Beispiel

Die Simulation wird gestartet und die beiden Rechtecke mit Hilfe der Maus von der Plattform gestoßen.

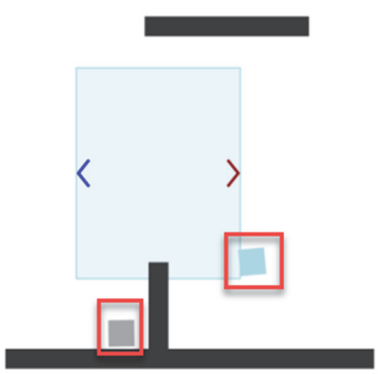


Bild: Simulation der Anordnung

Da das graue Rechteck über eine relativ **hohe Masse** verfügt, wird es vom Luftstrom nur leicht abgelenkt. Das Rechteck fällt in den linken Bereich unterhalb des Luftstroms. Anders das blaue Rechteck: Es wird wegen seiner **geringen Masse** sehr stark vom Luftstrom nach rechts abgelenkt. Es landet im rechten Bereich unterhalb des Luftstroms.

11.8 Fazit

In der Rubrik „Förderelemente“ stehen drei spezialisierte Förderelemente zur Verfügung. Je nach Anwendungsfall kann ein bestimmter Typ direkt verwendet werden. Es ist aber ebenso möglich, Anpassungen an den Eigenschaften vorzunehmen und so das Verhalten des Förderelements weiter den eigenen Bedürfnissen bzw. den Gegebenheiten der realen Anlage anzupassen.

12 Das Flüssigkeits-Objekt

Ein Flüssigkeits-Objekt verfügt über **drei Zuflüsse** und ebenso viele **Abflüsse**. Die Zu- und Abflüsse können sowohl mit Bit-Operanden als auch mit digitalen Operanden angesteuert werden.

12.1 Binäre Zu- und Abläufe

Beispiel (Fluessigkeit01):

Im ersten Beispiel sind an dem Flüssigkeits-Objekt **binäre** Operanden an den Zu- und Abläufen angegeben. Dabei werden die beiden Operanden E0.0 „Zulauf1“ und E0.1 „Zulauf2“ für die Zuläufe ausgewertet. Die Operanden E0.2 „Ablauf1“ und E0.3 „Ablauf2“ kommen an den Abläufen zum Einsatz. Alle Operanden des Beispiels entstammen dem Sim-Device. Als max. Füllstand sind 1000 Inkremente an der Eigenschaft „Max. Füllstand in Inkrementen“ eingestellt.

Da ein Bit-Operand nur den Wert 0 oder 1 annehmen kann und ein Zulauf bzw. Abfluss von einem Inkrement zu gering wäre, besteht bei Bit-Operanden die Möglichkeit, einen Bereich anzugeben. Über diesen Bereich ist es möglich, die Anzahl der Inkremente vorzugeben, welche beim Status 1 des Bit-Operanden für den Zu- bzw. Ablauf zu verwenden sind. Im Beispiel sollen die beiden Zuläufe unterschiedlich stark sein. Während der „Zulauf1“ mit 20 Inkrementen gewichtet wird, soll der „Zulauf2“ eine Gewichtung von 40 Inkrementen aufweisen.

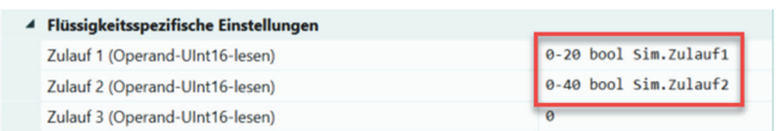


Bild: Einstellung der Zuläufe

Ähnliches bei den Abflüssen: Auch hier wird der „Abfluss2“ stärker gewichtet. Bei Status 1 fließen 40 Inkremente aus dem Flüssigkeits-Objekt ab, während „Abfluss1“ den Füllstand nur um 20 Inkremente verringert.

Bild: Einstellung für die Abflüsse

Alle Bit-Operanden sind mit Hilfe von Schalter-Objekten zu beeinflussen. Somit werden die vier notwendigen Schalter platziert und mit den Operanden versorgt.

Der momentane Stand des Flüssigkeitsbehälters ist mit einer digitalen Anzeige aus der Rubrik „Anzeige-Objekte“ zu visualisieren. Allerdings ist zunächst der Operand zu definieren, in welchen der aktuelle Füllstand des Flüssigkeits-Objekts zu schreiben ist. Hier soll das EW2 „Füllstand“ zum Einsatz kommen, somit ist dieser Operand an der Eigenschaft „Füllstands-Sensor“ einzutragen.

Das EW2 wird nach dem Platzieren der digitalen Anzeige auch an der Eigenschaft „Operand“ angegeben. Des Weiteren ist an der Eigenschaft „Verwendetes Zahlenformat“ die Einstellung „Dezimal“ zu selektieren.

Die Anordnung hat nach diesen Aktionen das nachfolgend dargestellte Aussehen.

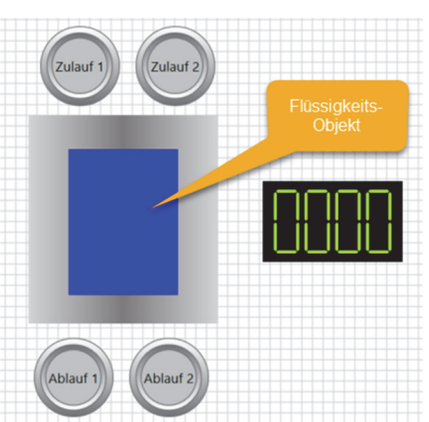


Bild: Anordnung für dieses Beispiel

© MHJ-Software GmbH & Co. KG - PLC-Lab

Damit sind alle Einstellungen getätigt und die Simulation kann beginnen.

Beim Start der Simulation hat das Flüssigkeits-Objekt den in der Eigenschaft „Anfangsfüllstand in Prozent“ eingestellten Füllstand. Diese Eigenschaft ist standardmäßig auf 10% gesetzt, somit wird ein Füllstand von 100 Inkrementen beim Start eingestellt.

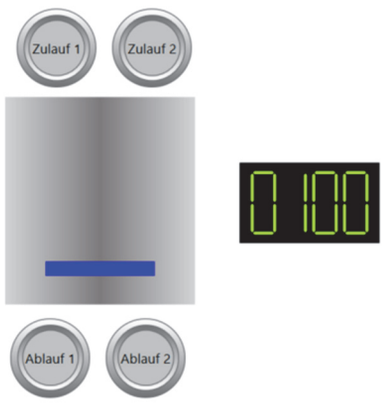


Bild: Situation beim Start der Simulation

Die digitale Anzeige zeigt den Wert 100 an und das Flüssigkeits-Objekt visualisiert über die blaue Fläche den Füllstand.

Nun können mit Hilfe der Schalter-Objekte die Zu- und Abflüsse gesteuert werden. Je nachdem, ob die Inkremente der Zu- oder der Abflüsse höher sind, steigt bzw. sinkt der Füllstand. Im folgenden Bild ist der Zulauf 1 offen sowie der Ablauf 2. Somit erfolgt ein Zulauf von 20 Inkrementen und ein Abfluss von 40 Inkrementen. In der Summe verringert sich also der Füllstand um 20 Inkremente pro Zeiteinheit.

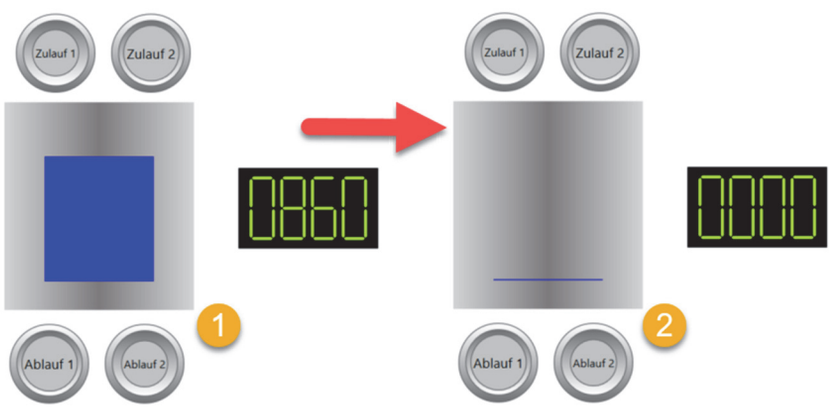


Bild: Simulation der Zu- und Abflüsse

12.2 Digitale Zu- und Abläufe

Neben binären Operanden ist auch die Angabe von **digitalen Operanden** an den Zu- und Abflüssen möglich. Dabei wird dann direkt der Wert des Operanden als Maß für den Zu- bzw. Abfluss verwendet.
Zum Verändern der Werte können beispielsweise Schieberegler aus der Rubrik „Eingabe-Objekte“ zum Einsatz kommen.

Beispiel (Fluessigkeit02):

Im Beispiel soll der Behälter aus dem vorherigen Beispiel zum Einsatz kommen. Allerdings werden digitale Operanden beim Zu- und Abfluss verwendet. Beim Zufluss kommt das EW4 „Zulauf1“ zum Einsatz, für den Abfluss das EW6 „Abfluss1“. Das EW2 „Füllstand“ wird für den aktuellen Füllstand des Flüssigkeits-Objekts verwendet. Dieser wird wie gehabt mit Hilfe einer digitalen Anzeige visualisiert.

Für die Steuerung des Zu- und Abflusses kommt jeweils ein **Schieberegler-Objekt** aus der Rubrik „Eingabe-Objekte“ zum Einsatz. Beide haben hier einen Bereich von 0 bis 20, weshalb bei beiden der Wert 0 in der Eigenschaft „Minimalwert der Anzeige“ und der Wert 20 in „Maximalwert der Anzeige“ anzugeben ist. Nachfolgend ist die Einstellung für den Schieberegler des Ablaufs zu sehen.

Bild: Einstellungen des Schiebereglers für den Abfluss

Im Flüssigkeits-Objekt wird der Operand „Zulauf1“ an der Eigenschaft „Zulauf 1“ eingetragen. Gleiches gilt für den Abfluss, hier wird der Operand „Abfluss1“ an der Eigenschaft „Abfluss 1“ angegeben.

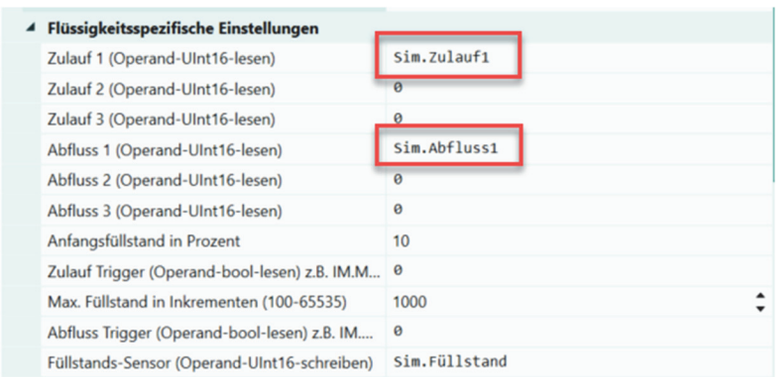


Bild: Einstellungen innerhalb des Flüssigkeits-Objekts

Die Anordnung hat somit das im folgenden Bild dargestellte Aussehen:

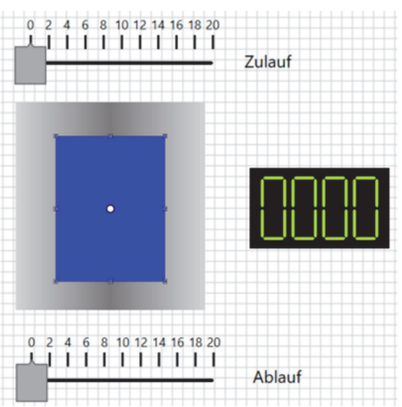


Bild: Anordnung für dieses Beispiel

Wird die Simulation gestartet, dann präsentiert sich ein ähnliches Bild, wie beim letzten Beispiel. Der Anfangs-Füllstand beträgt 100 Inkremente und es findet kein Zu- oder Abfluss statt, da beide Schieberegler auf den Wert 0 gesetzt sind.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Verändert man die Einstellungen der Schieberegler, dann steigt bzw. sinkt der Füllstand des Flüssigkeits-Objekts, je nachdem, ob der Zufluss oder der Abfluss überwiegt.

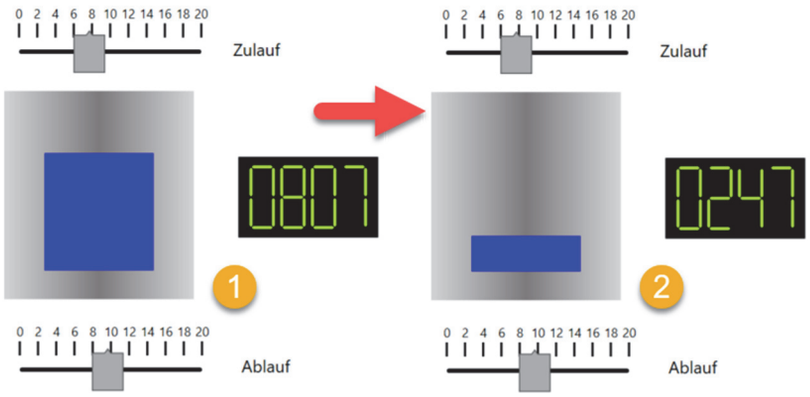


Bild: Simulation der Anordnung

In obiger Darstellung überwiegt der Abfluss, weshalb mit zunehmender Simulationsdauer der Füllstand abnimmt.

12.3 Auswertung des Füllstands über binäre Sensoren

In den bisherigen Beispielen wurde der **Füllstand** des Flüssigkeits-Objekts mit einer digitalen Anzeige visualisiert. Allerdings erfolgte keine direkte Auswertung des Füllstands mit Sensoren, wie dies in der Praxis meist der Fall ist. Die Vorgehensweise dafür wird nun erläutert.

Eine Möglichkeit, den Füllstand des Flüssigkeits-Objekts auszuwerten, besteht darin, die beiden binären Sensoren zu verwenden. Dabei kann angegeben werden, in welchem Bereich der Sensor auslösen soll. Gibt man z. B. den Bereich 0-200 an, dann würde der Sensor betätigt werden, sobald sich der Füllstand des Flüssigkeits-Objekts in diesem Bereich befindet.

Beispiel (Fluessigkeit03):

Es wird die Anordnung des letzten Beispiels mit der Steuerung der Zu- und Abflüsse über digitale Operanden und Schieberegler verwendet. Das Beispiel wird um die beiden Operanden E0.0 „MinSensor“ und E0.1 „MaxSensor“ erweitert. Der Operand „MinSensor“ soll den Status 1 besitzen, sobald sich der Füllstand im Bereich 0 bis 200 befindet. Der max. Füllstand im Bereich 800 bis 1000 soll den Operanden „MaxSensor“ betätigen.

Somit wird „MinSensor“ an der Eigenschaft „Binärer Füllstands-Sensor 1“ eingetragen und als Bereich ist „0-200“ anzugeben. Der Operand „MaxSensor“ ist an der Eigenschaft „Binärer Füllstands-Sensor 2“ anzugeben, als Bereich ist „800-1000“ notwendig.

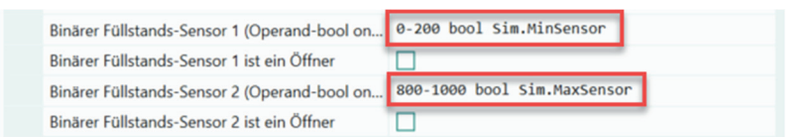


Bild: Angaben an den Sensor-Eigenschaften des Flüssigkeits-Objekts

Die Sensoren könnten als **Öffner** ausgelegt werden, wenn man die jeweilige Eigenschaft „Binärer Füllstands-Sensor X ist ein Öffner“ selektiert. Für das Beispiel sind sie aber als **Schließer** konfiguriert, was bedeutet: Befindet sich der Füllstand innerhalb des angegebenen Bereichs, dann hat der Operand den Status 1.

Mit den beiden Sensor-Operanden sollen **gelbe LEDs** angesteuert werden, um optisch anzuzeigen, welchen Wert sie besitzen. Dazu werden zwei Lampen aus der Unter-Rubrik „LEDs“ entnommen und platziert. Nach diesen Aktionen hat die Anordnung folgendes Aussehen:

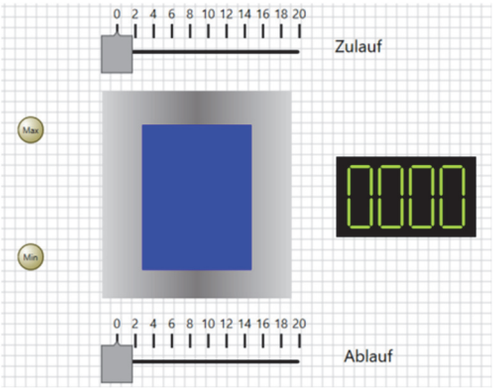


Bild: Anordnung für das Beispiel

Beim Start der Simulation hat der Anfangs-Füllstand den Wert 100, somit wird der Sensor mit dem Operanden „Min-Sensor“ ausgelöst, denn der Füllstand befindet sich ja innerhalb des Bereichs 0 bis 200. Der Operand besitzt den Status 1 und die entsprechenden LED leuchtet.

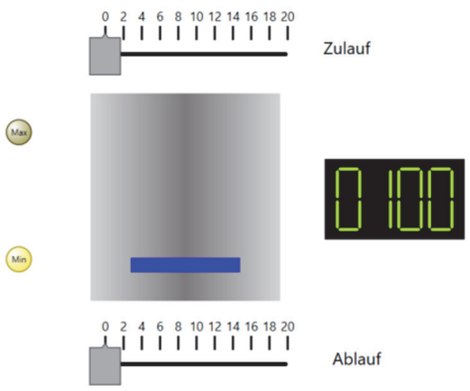


Bild: Start der Simulation mit Füllstand 100

Am Zulauf ist nun der Wert 10 einzustellen. Damit erhöht sich der Füllstand, und sobald der Wert höher als 200 ist, wird die LED von „MinSensor“ wieder dunkel. Denn dann besitzt der Operand wieder den Status 0.

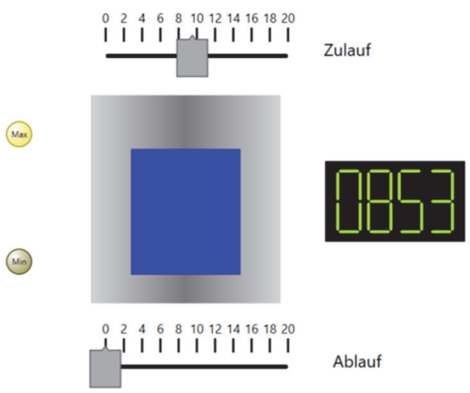


Bild: Ein Füllstand von über 800 wird erreicht

In der obigen Darstellung ist ein Füllstand von über 800 Inkrementen vorhanden. Dies hat zur Folge, dass der Sensor mit dem Operanden „MaxSensor“ betätigt ist und der Operand den Status 1 besitzt. Somit leuchtet die LED „Max“. Würde man den Zulauf stoppen und den Ablauf erhöhen, dann verringert sich der Füllstand, und sobald der Füllstand von 800 Inkrementen unterschritten wird, hat der Operand wieder den Status 0.

© MHJ-Software GmbH & Co. KG - PLC-Lab

12.4 Flüssigkeits-Objekt mit mehr als zwei Sensoren realisieren

Gelegentlich sind mehr als zwei binäre Sensoren an einem Flüssigkeits-Objekt notwendig. Ist dies der Fall, dann können die Sensoren sehr einfach „nachgerüstet“ werden. Hierbei kommen sog. **Rechner-Objekte** aus der Rubrik „Rechner-Objekt“ zum Einsatz.

Beispiel (Fluessigkeit04):

Die Anordnung aus dem vorausgegangenen Beispiel wird verwendet und entsprechend erweitert.

Im Beispiel soll ein weiterer Sensor nachgerüstet werden. Dieser zeigt an, wann sich der Füllstand im Norm-Bereich befindet. Dieser Norm-Bereich ist bei einem Füllstand im Bereich 400 bis 600 vorhanden. In diesem Fall soll der Operand E0.2 „NormSensor“ den Status 1 besitzen.

Dazu wird zunächst ein Rechner-Objekt aus der gleichnamigen Rubrik platziert. In der Eigenschaft „Anzuwendende Funktionsart“ ist „In Bereich“ auszuwählen. Dies bedeutet, es wird überprüft, ob ein an der Eigenschaft „Operand **Eingang 2**“ eingetragener Operand sich **innerhalb** des über „Operand **Eingang 1**“ und „Operand **Eingang 3**“ definierten **Bereichs** befindet. Somit ist an „Operand Eingang 1“ die Konstante 400 anzugeben, bei „Operand Eingang 3“ ist der Wert 600 einzutragen. An der Eigenschaft „Operand Eingang 2“ wird der Operand „Füllstand“ angegeben, denn er enthält ja den aktuellen Füllstand des Flüssigkeits-Objekts.

Das Ergebnis wird in den Operanden „NormSensor“ geschrieben.

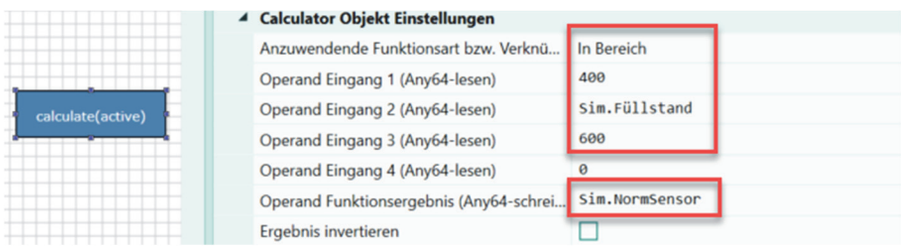


Bild: Einstellungen des Rechner-Objekts

Auch für den Operand „NormSensor“ ist eine LED zu platzieren, um dessen Status anzuzeigen. Die Anordnung hat somit folgendes Aussehen:

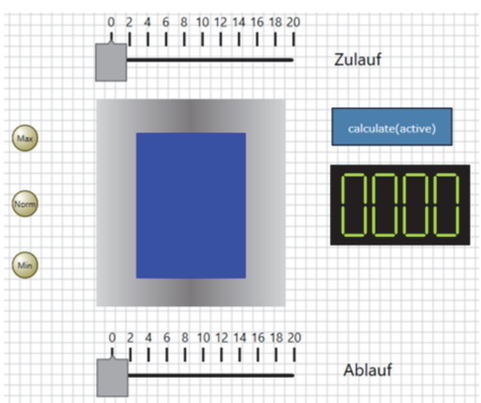


Bild: Anordnung für das Beispiel

Startet man die Simulation, dann hat der Operand „NormSensor“ den Status 1, sobald sich der Füllstand im Bereich 400 bis 600 befindet. In diesem Fall leuchtet dann auch die entsprechende LED.

12.5 Weitere Einstellungen

Die beiden Eigenschaften „Zulauf Trigger“ und „Abfluss Trigger“ ermöglichen es, den Zeitpunkt der Auswertung der Zu- und Abflüsse zu bestimmen. Sobald eine positive Flanke am Zulauf-Trigger ansteht, werden die Zuläufe erfasst und entsprechend dem Füllstand hinzuaddiert. Gleiches gilt für den Abfluss-Trigger.

So kann ein Flüssigkeits-Objekt auch beispielsweise zum **Zählen** verwendet werden.

13 Verbindungselemente bzw. Gelenke

Verbindungselemente bzw. Gelenke werden zur Dynamisierung der Anlagen verwendet. Damit können beispielsweise **lineare Bewegungen** wie bei Zylindern oder **Rotationsbewegungen** realisiert werden. Ebenso sind Verbindungen vorhanden, um dynamische Objekte unlösbar miteinander zu verbinden.

Verbindungselemente definieren die Beziehung zwischen zwei Physikobjekten. Die beiden Physik-Objekte werden dann als Anker-Objekte bezeichnet. Bei einigen Verbindungen ist eines der beiden Anker-Objekte dominant und wird als Master-Objekt bezeichnet. Wird eine solche Verbindung erzeugt, dann ist das Objekt, bei dem das Verbindungselement **begonnen** wird, das **Master-Objekt**.

13.1 Verbindung für lineare Bewegung

Diese Verbindung kommt beispielsweise bei **Zylindern** zum Einsatz, weitere Anwendungsfälle sind:
Aufzüge, Hebebühnen, Schiebetüren, Vorschübe, Greifer.

Bild: Prinzip-Symbol der Verbindung für lineare Bewegung

13.1.1 Erstes Beispiel zur Demonstration der Verbindung für lineare Bewegung

Im ersten Beispiel wird zunächst ein feststehendes Rechteck und anschließend ein dynamisches Rechteck platziert. Danach selektiert man eine Verbindung für lineare Bewegung innerhalb der Tools.

Da die Verbindung für lineare Bewegungen über ein Master-Objekt sowie ein Slave-Objekt verfügt, **muss das Zeichnen beim Master-Objekt begonnen werden**. Dies ist das feststehende Rechteck. Daraufhin wird die Maus bei betätigter linker Maustaste in Richtung des zweiten Objekts bewegt und dort die Maustaste losgelassen. Als Folge ist das Symbol der Verbindung zu erkennen.

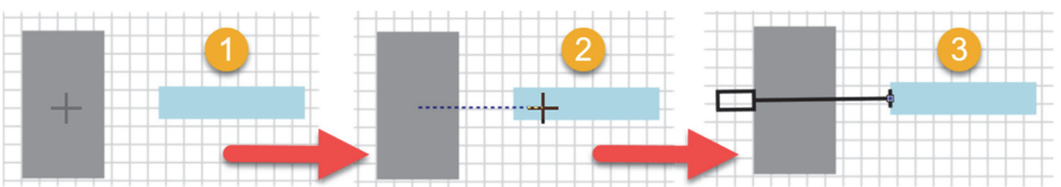


Bild: Zeichnen der Verbindung für lineare Bewegung

Im folgenden Bild ist die Verbindung dargestellt, des Weiteren sind die Elemente der Verbindung benannt:

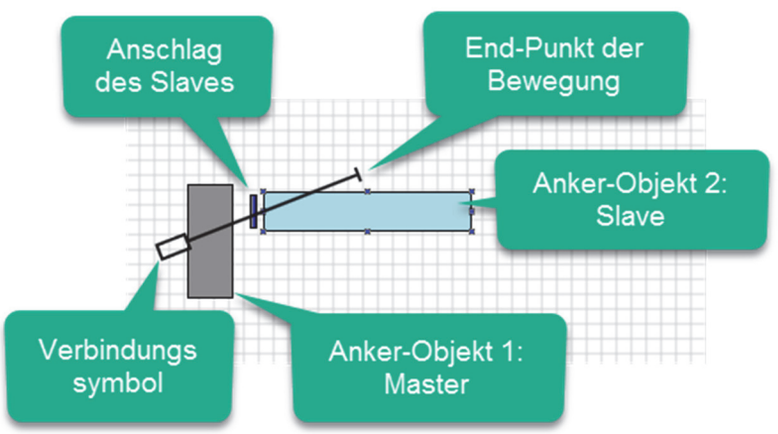


Bild: Elemente einer linearen Verbindung

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.1.2 Länge der Verbindung anpassen

Es gibt mehrere Möglichkeiten, die **Länge** der Verbindung zu verändern.

13.1.2.1 Verschieben der Anker-Objekte

Wird das Slave-Objekt verschoben, dann wird die Länge der Verbindung angepasst, falls dies notwendig ist. Gleiches gilt für das Verschieben des Master-Objekts.

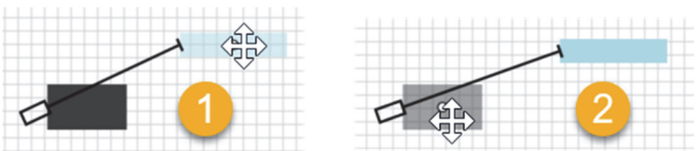


Bild: Die Länge der linearen Verbindung wird beim Verschieben der Anker-Objekte angepasst.

Im Bild (Punkt 1) wird das Slave-Objekt verschoben und bei Bedarf die Länge der linearen Verbindung automatisch angepasst. Bei Punkt 2 ist die gleiche Reaktion zu sehen, allerdings wird das Master-Objekt in seiner Position verändert.

13.1.2.2 Selektion der Verbindung und Verändern der Länge mit der Maus

Eine weitere Variante besteht darin, das Symbol der Verbindung zu selektieren …

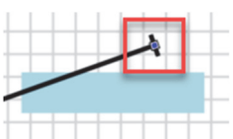


Bild: Anfasser der linearen Verbindung

… und danach den Anfasser am Ende des Symbols mit der Maus zu verschieben.

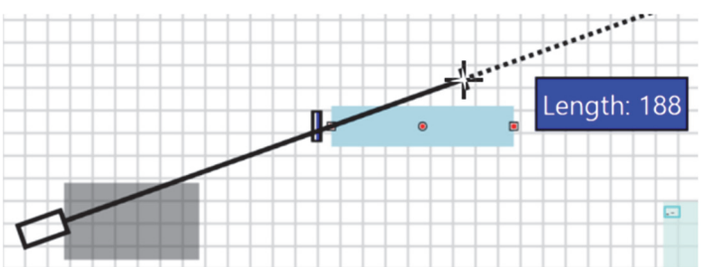


Bild: Verändern der Länge durch Verschieben des Anfassers

Das Verkürzen ist dabei nur bis zu der Stelle möglich, wo sich der Anschlag des Slave-Objekts befindet.

13.1.2.3 Selektion der Verbindung und Verändern der Länge über die Eigenschaft

Zuletzt kann man das Symbol der Verbindung selektieren und danach in der Tabelle der Eigenschaften den Eintrag „Länge Linearbewegung“ verändern. Auch hier gilt: Die Länge kann nur so weit verkürzt werden, bis der Anschlag des Slave-Objekts erreicht ist.

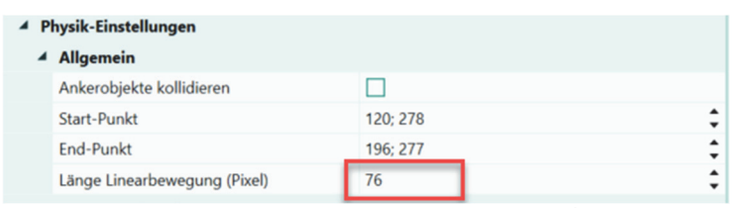


Bild: Länge verändern innerhalb der Eigenschaften

13.1.3 Steigung und Winkel verändern

Auch für das Verändern der **Steigung** bzw. des **Winkels** der linearen Bewegung stehen mehrere Möglichkeiten zur Verfügung.

13.1.3.1 Verschieben der Anker-Objekte

Wird eins der Anker-Objekte verschoben, dann passt sich die lineare Verbindung der neuen Position des Objekts entsprechend an. Im folgenden Bild ist das Verschieben des Slave-Objekts zu sehen.

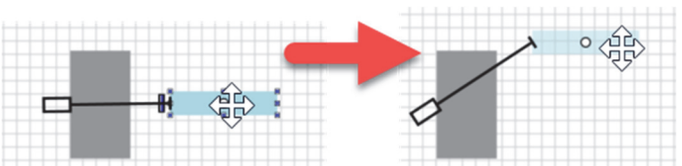


Bild: Verschieben eines Anker-Objekts

13.1.3.2 Einstellung des Winkels mit Hilfe der Maus

Führt man einen Doppelklick auf dem Symbol der Verbindung aus, dann wird der **Rotationsmodus** der Verbindung gestartet. Dies ist auch am Aussehen der Anfasser zu erkennen. Nun kann mit der Maus der Anfasser am Ende angeklickt und in der Ausrichtung geändert werden.

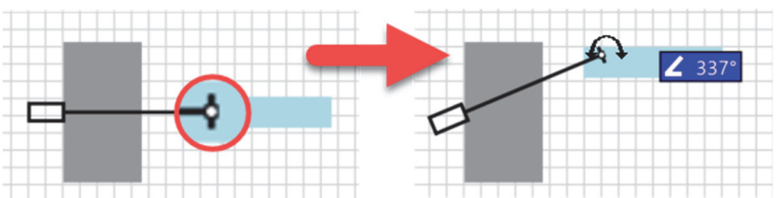


Bild: Verändern des Winkels über den Rotationsmodus

Beim Verändern des Winkels wird auch das Slave-Objekt nachgeführt.

13.1.3.3 Direkte Einstellung des Winkels mit Hilfe der Eigenschaft

Selektiert man das Symbol der Verbindung, dann sind die **Eigenschaften** in der Tabelle sichtbar. Hier kann nun mit Hilfe der Eigenschaft „**Rotation**“ innerhalb der Rubrik „**Größe**“ die **Ausrichtung** direkt verändert werden.

13.1.4 Motor der Verbindung verwenden

Die Verbindung für lineare Bewegung besitzt einen **Motor**, mit dem das Slave-Objekt zum Minimum oder Maximum bewegt werden kann. In den Eigenschaften der Verbindung ist dieser Motor aktivierbar. Des Weiteren finden sich dort die Eigenschaften für die Bewegung zum Minimum bzw. zum Maximum. Diese können sowohl mit Operanden als auch mit Konstanten (0 oder 1) belegt werden.

Beispiel (LinearеVerbindung02):

Mit Hilfe der Operanden E0.0 „ZumMaximum“ und E0.1 „ZumMinimum“ soll das Slave-Objekt einer Verbindung für lineare Bewegungen bewegt werden. Beide Operanden gehören zum Sim-Device. Die Beeinflussung der beiden Operanden erfolgt mit Hilfe von als Tastern konfigurierten Schalter-Objekten.

Die Anordnung hat folgendes Aussehen:

© MHJ-Software GmbH & Co. KG - PLC-Lab

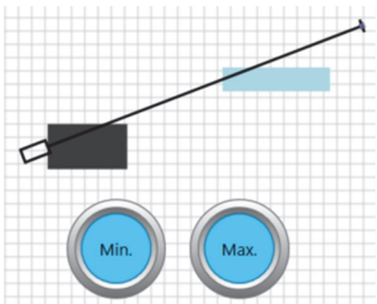


Bild: Anordnung für dieses Beispiel

Damit die beiden Operanden die Bewegung der Verbindung auslösen, sind die Motor-Eigenschaften entsprechend zu belegen.

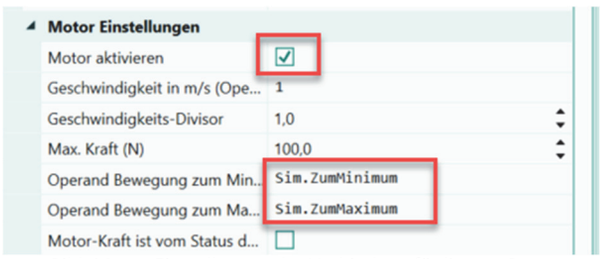


Bild: Motor-Einstellungen der Verbindung für lineare Bewegung

Die beiden Operanden werden in die Eigenschaft „Operand Bewegung zum Minimum“ und „Operand Bewegung zum Maximum“ eingetragen. Des Weiteren ist der Motor zu aktivieren. Die anderen Einstellungen, wie z. B. die Geschwindigkeit und die Kraft des Motors, können für diese Beispiel auf den Standardwerten belassen werden.

Damit ist die Anordnung komplett und die Simulation kann beginnen.

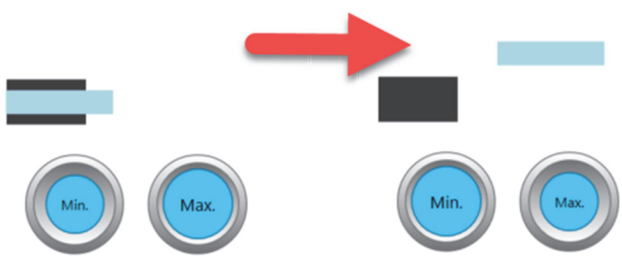


Bild: Simulation des Beispiels

Wird der Taster „Min.“ betätigt, dann bewegt sich das Slave-Objekt nach links unten in Richtung des Master-Objekts, bis das Minimum der Verbindung erreicht ist. Bei Betätigung des Tasters „Max.“ erfolgt die Bewegung in die entgegengesetzte Richtung, hin zum Maximum.

13.1.4.1 Geschwindigkeit des Motors

An der Eigenschaft für die **Geschwindigkeit des Motors** für die lineare Verbindung kann sowohl eine Konstante als auch ein Operand angegeben werden. Im letzten Beispiel wurde die Standardeinstellung mit der Konstanten 1 belassen, somit bewegte sich das Slave-Objekt mit einer Geschwindigkeit von 1 m/s. Möchte man die Geschwindigkeit von einem Operanden abhängig machen, dann wird meist ein digitaler Operand angegeben. Dessen Inhalt wird dann als Geschwindigkeitsvorgabe in m/s verwendet.

Ist die Geschwindigkeit in der Realität zu hoch, so dass sie in der Simulation nicht sinnvoll darstellbar ist, dann kann man mit Hilfe der Eigenschaft „Geschwindigkeits-Divisor“ den Wert für die Simulation verringern. Dabei wird die Geschwindigkeit durch den dort angegebenen Wert dividiert.

Benötigt man nur zwei verschiedene Geschwindigkeiten, dann kann auch ein Bit-Operand mit Angabe eines Bereichs zur Anwendung kommen. So würde die Angabe „1-3 Sim.E0.2“ bewirken, dass beim Status 0 des Operanden die Geschwindigkeit **1 m/s** eingestellt ist und **3 m/s** sobald der Operand den Status 1 besitzt. Auf diese Weise ist es möglich, mit einem Bit-Operanden eine normale Geschwindigkeit und einen schnellen Vorschub zu realisieren.

Beispiel (LineareVerbindung03):

Eine Verbindung für lineare Bewegung wird mit den Operanden E0.0 „ZumMaximum“ und E0.1 „ZumMinimum“ gesteuert. Über den Operanden E0.2 „GeschwDoppelt“ soll die Geschwindigkeit der Bewegung verdoppelt werden. Die normale Geschwindigkeit beträgt 1 m/s. Alle Operanden gehören zum Sim-Device.

Die Umschaltung der Operanden erfolgt mit Schalter-Objekten, dabei sind die Schalter der Richtungs-Operanden als Taster auszulegen. Nachfolgend ist die Anordnung des Beispiels zu sehen.

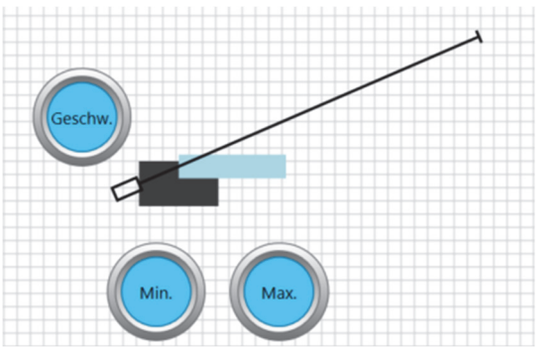


Bild: Anordnung für dieses Beispiel

Die Motor-Einstellungen der Verbindung für lineare Bewegung sind wie folgt vorzunehmen:

Bild: Einstellungen des Motors der Verbindung

Besonders hervorzuheben ist dabei die Einstellung für die Geschwindigkeit. Hier wurde neben dem Bit-Operanden auch der Bereich „1-2“ angegeben. Dies hat zur Folge, dass beim Status 0 des Operanden die Geschwindigkeit **1 m/s** verwendet wird, beim Status 1 ist die Geschwindigkeit **2 m/s** eingestellt.

Mit diesen Einstellungen kann die Simulation gestartet werden. Solange der Schalter „Geschw.“ nicht betätigt ist, bewegt sich das Slave-Objekt mit normaler Geschwindigkeit. Bei betätigtem Schalter „Geschw.“ ist die doppelte Geschwindigkeit eingestellt, weshalb die Bewegung auch deutlich schneller ist.

Fazit: Die **Geschwindigkeit** der Verbindung für lineare Bewegung kann **analog** vorgegeben werden. Damit sind auch beispielsweise **Regelungen** möglich, insbesondere in Kombination mit den noch im weiteren Verlauf beschriebenen Sensoren. Häufig werden in der Praxis lediglich zwei Geschwindigkeiten benötigt, wobei die Umschaltung mit einem Bit-Operanden erfolgt. Auch dies ist realisierbar: Dabei wird zusätzlich zum Bit-Operanden der Bereich mit den beiden Geschwindigkeitswerten angegeben.

13.1.4.2 Weitere Motor-Einstellungen

Bei der Erläuterung der Zylinder-Objekte wurde bereits erwähnt, dass die Verbindung für lineare Bewegung auch verwendet wurde, um die verschiedenen Zylinder-Arten zu realisieren. Die dazu notwendigen Eigenschaften stehen alle zur Verfügung. So wird z. B. mit der Eigenschaft „Geschwindigkeit umkehren bei Status 0 der Bewegungsoperanden“ die Federrückstellung realisiert. Ebenso ist die Eigenschaft „Impulsverhalten“ selektierbar.

Sollte also eine Zylinder-Art nicht vorhanden sein, dann kann das Verhalten mit der Verbindung für lineare Bewegung selbst aufgebaut werden.

Dies gilt auch dann, wenn man Verhalten von Zylindern benötigt, welches über die vorkonfigurierten Zylinder-Objekte nicht einstellbar ist. Als Beispiel sei ein Zylinder genannt, dessen Kolbenstange in Ruhestellung irgendeine Mittelposition einnehmen soll. Dies ist sehr einfach mit Hilfe der Verbindung für lineare Bewegung zu realisieren.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.1.5 Endschalter und Sensoren der Verbindung für lineare Bewegung

Die Verbindung ist mit zahlreichen Sensoren ausgestattet. Zu den üblichen gehören die **Sensoren** am **Minimum** und am **Maximum**, welche wahlweise als Schließer oder Öffner konfigurierbar sind.

13.1.5.1 Sensor für die Position des Slave-Objekts

An diesem Sensor kann ein **digitaler Operand** angegeben werden. Hier wird die **aktuelle** Position des Slave-Objekts hineingeschrieben. Zusätzlich zum Operanden ist auch die Angabe eines Bereichs möglich. Dies erweitert die Flexibilität des Sensors enorm, da somit vom Sensor die Werte geliefert werden können, welche auch in der realen Anlage vorkommen.

Beispiel (LineareVerbindung04):

Im Beispiel besitzt die Verbindung für lineare Bewegung eine Länge von 500 Pixeln. Somit werden die Werte 0 bis 500 vom Sensor geliefert. In der realen Anlage liefert der Sensor Werte im Bereich von 0 bis 3000.

Damit diese Werte auch in der Simulation zur Verfügung stehen, ist an der Eigenschaft zusätzlich der Bereich „0-3000“ mit anzugeben. Mit Hilfe dieser Bereichsangabe werden die Werte 0-500 auf den Bereich 0-3000 skaliert. Damit ist die Länge der Verbindung in der Simulation unerheblich.

Die Verbindung für lineare Bewegung wird über die Operanden E0.0 „ZumMaximum“ und E0.1 „ZumMinimum“ gesteuert. Die Positionsdaten sind im EW2 „Position“ abzulegen. Alle Operanden sind dem Sim-Device zugeordnet.

Die aktuelle Position ist mit Hilfe einer **digitalen Anzeige** aus der Rubrik „Anzeige-Objekte“ zu visualisieren. Als verwendetes Zahlenformat ist „**Dezimal**“ einzustellen. Als Operand wird das EW2 „Position“ eingetragen.

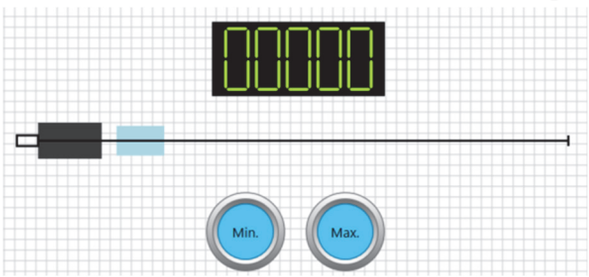


Bild: Anordnung des Beispiels

Nachfolgend die Belegung des Sensors für die Position.

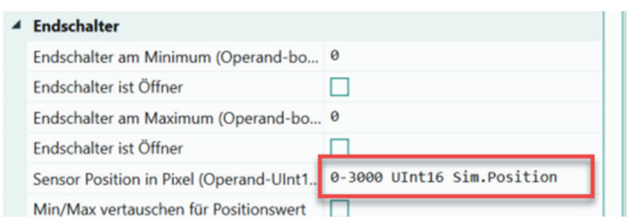


Bild: Belegung der Eigenschaft „Sensor Position in Pixel“

Damit sind alle Einstellungen getätigt und die Simulation kann gestartet werden.

Wird das Slave-Objekt zum Minimum auf der linken Seite bewegt, dann zeigt die digitale Anzeige den Wert 0 an. Wird das Slave-Objekt daraufhin nach rechts bewegt, dann erhöht sich der Positionswert, bis der maximale Wert 3000 erreicht ist.

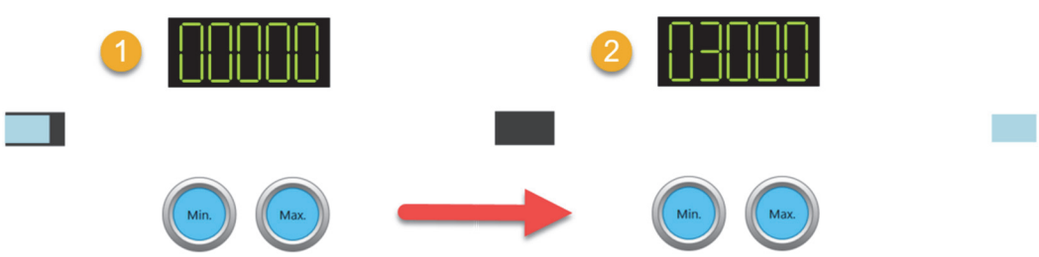


Bild: Simulation des Beispiels

Fazit: Mit Hilfe des Positions-Sensors kann die Pixel-Position des Slave-Objekts einer Verbindung für lineare Bewegung erfasst und in einen digitalen Operanden geschrieben werden. Damit die gelieferten Werte den Werten der realen

Anlage entsprechen, ist zusätzlich die Angabe eines Bereichs möglich. Die gelieferten Werte werden dann auf diesen Bereich skaliert.

13.1.5.2 Kraft-Sensor

Über den Kraft-Sensor ist beispielsweise eine **Waage** realisierbar. In den Sensor wird die Kraft in N geschrieben, welche die Verbindung benötigt, um die momentane Position zu **halten**.

Beispiel (LineareVerbindung05):

Drei identisch große, dynamische Rechtecke mit unterschiedlicher blauer Färbung haben eine Dichte von 1, 2 und 3. Somit besitzen sie jeweils unterschiedliche Massen. Die Massen sollen mit Hilfe einer Waage erfasst werden. Diese Waage wird mit Hilfe einer Verbindung für lineare Bewegung realisiert, wobei die beiden Anker-Objekte vertikal angeordnet sind. Das Slave-Objekt befindet sich ca. in der Mitte des Verfahrwegs.

Wichtig dabei ist, dass die beiden Eigenschaften „Operand Bewegung zum Maximum“ und „Operand Bewegung zum Minimum“ den Wert 0 besitzen.

Die aktuell von der Verbindung benötigte Kraft wird im Operand EW2 „Kraft“ abgelegt, der aus dem Sim-Device stammt. Die Visualisierung des Werts erfolgt mit einer digitalen Anzeige im Zahlenformat „Dezimal“, welcher dieser Operand zugeordnet ist.

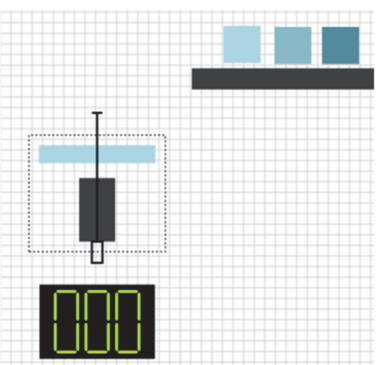


Bild: Anordnung für dieses Beispiel

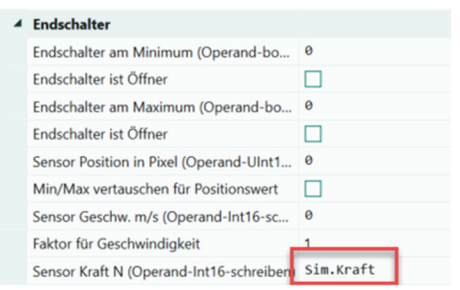


Bild: Belegung des „Sensor Kraft“

Startet man die Simulation, dann wird die Kraft angezeigt, welche zum Halten der Position notwendig ist. Wird ein Rechteck auf dem Slave-Objekt abgelegt, dann erhöht sich die notwendige Kraft. Beim Wechsel der Rechtecke kann anhand der Kraft ermittelt werden, welches Objekt die höchste Masse besitzt.

Bild: Simulation des Beispiels

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.1.6 Die Eigenschaft „Ankerobjekte kollidieren“

Wird diese Eigenschaft selektiert, dann kollidieren die beiden Anker-Objekte miteinander. Dies bedeutet, dass das Slave-Objekt nicht mehr in das Master-Objekt eindringen kann. Somit kann auch oftmals das Minimum der linearen Bewegung nicht mehr erreicht werden.

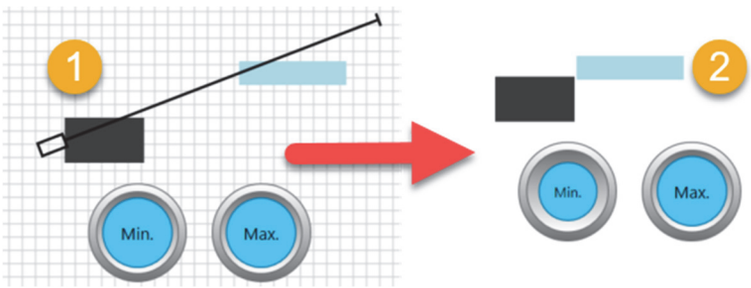


Bild: Simulation der linken Anordnung, wenn „Ankerobjekte kollidieren“ selektiert ist

Im rechten Bild (Punkt 2) kann sich das Slave-Objekt nicht weiter nach links bewegen, da es mit dem Master-Objekt kollidiert.

Würde man in diesem Fall die Position des Slave-Objekts über den Sensor auswerten, dann würde die Null-Position nicht erreicht werden. Möchte man dies ändern, dann kann man das Master-Objekt um **180° rotieren**, damit sich das Minimum auf der rechten Seite befindet.

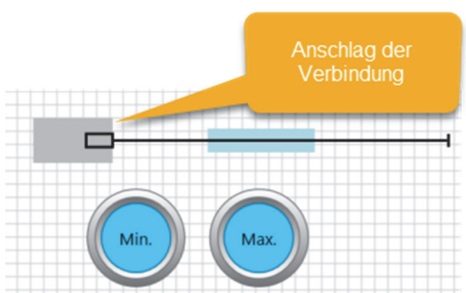


Bild: Das Master-Objekt wurde um 180° **rotiert** und der Anschlag befindet sich nun auf der rechten Seite

Fazit: Wird die Eigenschaft „Ankerobjekte kollidieren“ selektiert, dann kann das Slave-Objekt nicht mehr in das Master-Objekt eindringen, das Slave-Objekt wird vom Master-Objekt blockiert. Wird bei der Verbindung die Position des Slave-Objekts ausgewertet und durch Aktivierung der Eigenschaft die Null-Position nicht mehr erreicht, dann kann durch Rotation des Master-Objekts um 180° der Anschlag verändert werden.

13.1.7 Platzierung des Slave-Objekts

Bei einer Verbindung für lineare Bewegung kann man das Slave-Objekt innerhalb des Verfahrwegs beliebig platzieren. Die so eingestellte Position ist dann auch die **Grundstellung** beim **Start** der Simulation.

Ist die lineare Bewegung senkrecht oder waagerecht angeordnet und ist das Slave-Objekt selektiert, dann ist das Verschieben mit Hilfe der Pfeiltasten auf der Tastatur möglich.

13.1.7.1 Platzierung des Slave-Objekts am Minimum

Möchte man die **Grundstellung** des Slave-Objekts direkt am **Minimum einstellen**, dann bewegt man es so lange in Richtung des Minimums, bis sich die **Richtung** der linearen Bewegung **ändert**. Von dieser Stelle aus geht man eine Position zurück und hat damit die Anfangsposition eingestellt.

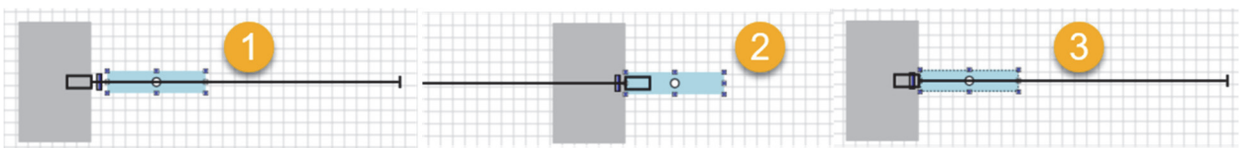


Bild: Darstellung der Bedienhandlungen

1. Das Slave-Objekt wird über die Pfeil-Taste nach links bewegt.
2. Das Slave-Objekt hat die linke Endposition um ein Pixel überschritten, weshalb sich die Bewegungsrichtung der Verbindung um 180° dreht.
3. Einmalig wird die Pfeil-Taste nach rechts betätigt, damit die ursprüngliche Richtung wieder vorhanden ist. Somit befindet sich das Slave-Objekt am Minimum der Verbindung.

13.1.7.2 Verwendung einer Schablone zur Platzierung des Slave-Objekts

Ist die lineare Bewegung nicht senkrecht oder waagerecht, dann ist die Platzierung des Slave-Objekts bei gleichzeitiger Beibehaltung des Winkels schwierig. Hier kommt dann eine sog. **Schablone** zum Einsatz. Folgende Schritte sind notwendig, um Objekte mit einer solchen Schablone zu versehen:

1. Selektion aller Objekte, welche mit der Schablone belegt werden sollen. Um mehrere Objekte zu selektieren, betätigt man die Umschalt-Taste und hält sie betätigt. Dann werden die einzelnen Objekte mit der Maus angeklickt. **Zuletzt wird das Symbol der Verbindung selektiert**.
2. Nun kann die Umschalt-Taste losgelassen werden. Innerhalb der Liste der Aktionen wird die Aktion „Schablone setzen“ ausgewählt.
3. Im Statusfeld von PLC-Lab erscheint die Meldung, dass die Schablone gesetzt wurde.
4. Nun sind die Objekte nur noch entlang einer Linie in der Position veränderbar. Dies gilt sowohl für Änderungen mit der
   Maus als auch bei Änderungen mit den Pfeil-Tasten auf der Tastatur.

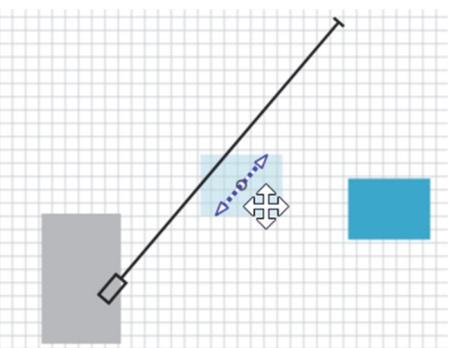


Bild oben: Für die beiden blauen
Rechtecke sind Schablonen gesetzt

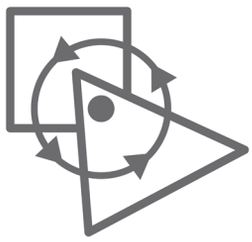
Im Bild ist zu erkennen, dass beim Verschieben des mit einer **Schablone** versehenen Objekts ein Pfeil angezeigt wird, welcher die Freiheitsgrade des Objekts anzeigt. Dies bedeutet eine Verschiebung kann nur in Pfeil-Richtung erfolgen.

Hinweis: Es können nicht nur Slave-Objekte mit einer Schablone versehen werden, dies ist auch für andere Objekte möglich. Allerdings ist **immer** eine lineare Verbindung notwendig, deren Ausrichtung die Schablone bestimmt. Soll ein Objekt wieder alle Freiheitsgrade beim Verschieben besitzen, also von der Schablone gelöst werden, dann selektiert man das Objekt und führt die Aktion „Schablone rücksetzen“ aus. Die Schablone hat nur Auswirkungen auf das Zeichnen, in der Simulation ist sie nicht aktiv.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.2 Verbindung für Drehbewegung

Bild: Prinzipdarstellung der Verbindung für Drehbewegung



Wie die Bezeichnung es vermuten lässt, können mit der Verbindung für Drehbewegung, **Drehbewegungen** jeglicher Art realisiert werden. Dazu gehören:

- Drehgelenke
- Ventilatoren mit variabler Geschwindigkeit
- Schranken
- Kippvorrichtungen
- Bewegliche Klappen
- Usw.

Auch diese Verbindung besitzt **ein Master-Objekt und ein Slave-Objekt**. Das Master-Objekt ist meist der feststehende Teil. Von diesem Master-Objekt aus führt eine durchgezogene Linie zum Drehpunkt der Verbindung, der durch einen Kreis mit Fadenkreuz symbolisiert wird. Von diesem ausgehend führt eine gestrichelte Linie zum Mittelpunkt des Slave-Objekts.

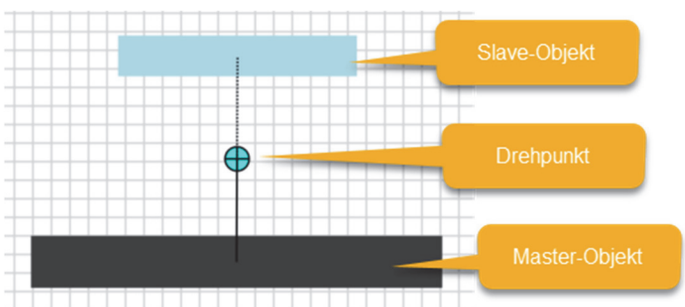


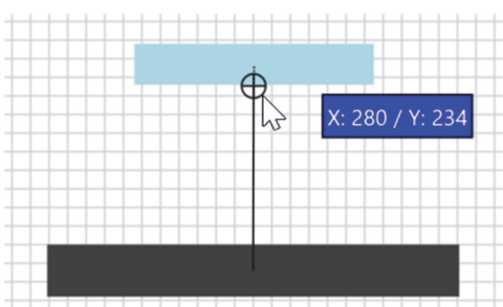
Bild: Benennung der Elemente einer Verbindung für Drehbewegung

Das Slave-Objekt rotiert um den Drehpunkt, welcher frei positioniert werden kann.

13.2.1 Lage des Drehpunkts verändern

Die Lage des Drehpunkts ist mit Hilfe der Maus veränderbar. Somit lassen sich verschiedenartige Drehungen realisieren.

Bild: Drehpunkt wird über die Maus verlagert.



Des Weiteren kann der Drehpunkt bei den Eigenschaften der Verbindung über die Pixelposition angegeben werden.

13.2.2 Motor der Verbindung für Drehbewegung

Die Drehbewegung besitzt einen **Antrieb**, dessen Geschwindigkeit und Drehrichtung über Konstanten oder Operanden vorzugeben sind.

Soll die Drehgeschwindigkeit (Angabe in U/min) konstant sein, dann kann an dieser Eigenschaft eine Konstante angegeben werden. Für eine variable Geschwindigkeit ist die Angabe eines digitalen Operanden möglich. Wird ein **Bit-Operand** mit einem **Bereich** eingetragen, dann wird zwischen den beiden Geschwindigkeiten umgeschaltet, je nach Status des Bit-Operanden (z. B. „10-30 Sim.E10.0“).

Beispiel (VerbindungDrehbewegung02):

Über die beiden Operanden E0.0 „Rechts“ und E0.1 „Links“ kann die Drehrichtung einer Verbindung für Drehbewegung gesteuert werden. Zusätzlich soll die Geschwindigkeitsvorgabe über den Operanden EW2 „Drehgeschw“ erfolgen. Alle Operanden gehören zum Sim-Device.

Die Beeinflussung der Bit-Operanden erfolgt über Schalter-Objekte. Der Operand EW2 „Drehgeschw“ soll mit Hilfe eines horizontalen Schiebereglers im Bereich von 10 (Minimalwert der Anzeige) bis 50 (Maximalwert der Anzeige) einstellbar sein.

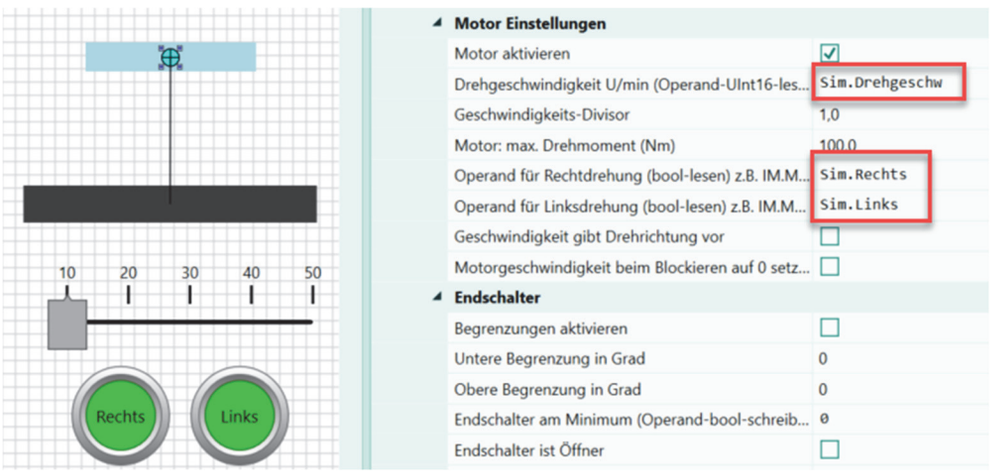


Bild: Anordnung für dieses Beispiel mit den Motor-Einstellungen der Drehbewegung

Im Bild ist zu erkennen, dass sich der Drehpunkt in der Mitte des Slave-Objekts befindet.

Wird die Simulation gestartet, dann kann die Drehbewegung über die Schalter „Rechts“ bzw. „Links“ ausgelöst werden. Über den Schieberegler ist die Geschwindigkeit einstellbar.

Mit Hilfe der Eigenschaft „Geschwindigkeits-Divisor“ kann die Geschwindigkeitsvorgabe für die Simulation angepasst werden, ohne dass das Signal selbst für die Simulation verändert werden muss. Wird hier z. B. der Wert 10 angegeben, dann wird der Geschwindigkeitswert des Operanden zunächst durch 10 dividiert und das Ergebnis als Geschwindigkeitswert für die Simulation verwendet.

Fazit: Über die Eigenschaft „Drehgeschwindigkeit“ kann die **Geschwindigkeit** der Rotationsbewegung in U/min vorgegeben werden. Im Normalfall wird dazu ein digitaler Operand verwendet und dessen Wert ausgewertet. Es ist aber auch die Angabe eines Bit-Operanden mit einem Bereich möglich. Eine solche Konstellation kommt dann in Frage, wenn nur zwei Geschwindigkeiten vorzugeben sind. Diese werden dann über den Bereich definiert.

Kommt in der Realität eine Drehgeschwindigkeit zum Einsatz, welche in der Simulation nicht mehr sinnvoll ist (z. B. 10000 U/min), dann kann mit Hilfe des „Geschwindigkeits-Divisor“ der eigentliche Vorgabewert identisch bleiben. Die in der Simulation verwendete Umdrehungs-Geschwindigkeit befindet sich dann in einem Bereich, in dem Änderungen visuell wahrgenommen werden können.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.2.2.1 Drehrichtung über Geschwindigkeitswert vorgeben

Im letzten Beispiel wurde die Drehrichtung mit Hilfe von Operanden vorgegeben. Dies bedeutet, bevor eine Drehung stattfindet, muss zunächst eine Geschwindigkeit > 0 vorhanden sein und einer der beiden Drehrichtungs-Operanden den Status 1 besitzen.

Es gibt allerdings auch Anwendungsfälle, bei denen das Vorzeichen der Geschwindigkeit die Drehrichtung bestimmt. Möchte man einen solchen Fall simulieren, dann ist die Eigenschaft „Geschwindigkeit gibt Drehrichtung vor“ zu selektieren. Der Motor wird in diesem Fall über den Operanden für die Rechtsdrehung eingeschaltet, hier kann dann wahlweise ein Operand oder die Konstante 1 angegeben werden.

Beispiel (VerbindungDrehbewegung03):

Eine Verbindung für Drehbewegung ist über den Operanden E0.0 „MotorEin“ einzuschalten. Die Drehrichtung und die Drehgeschwindigkeit wird über den Operanden EW2 „Drehgeschw“ vorgegeben. Das EW2 hat den Datentyp „Int16“, kann also auch negative Werte annehmen. Ein **negativer** Wert hat eine Drehbewegung **gegen** den **Uhrzeigersinn** zu Folge. Beide Operanden sind dem Sim-Device zugeordnet.

Der Operand E0.0 „MotorEin“ ist über ein Schalter-Objekt beeinflussbar; der digitale Operand kann über einen Schieberegler im Bereich -50 bis 50 verändert werden.

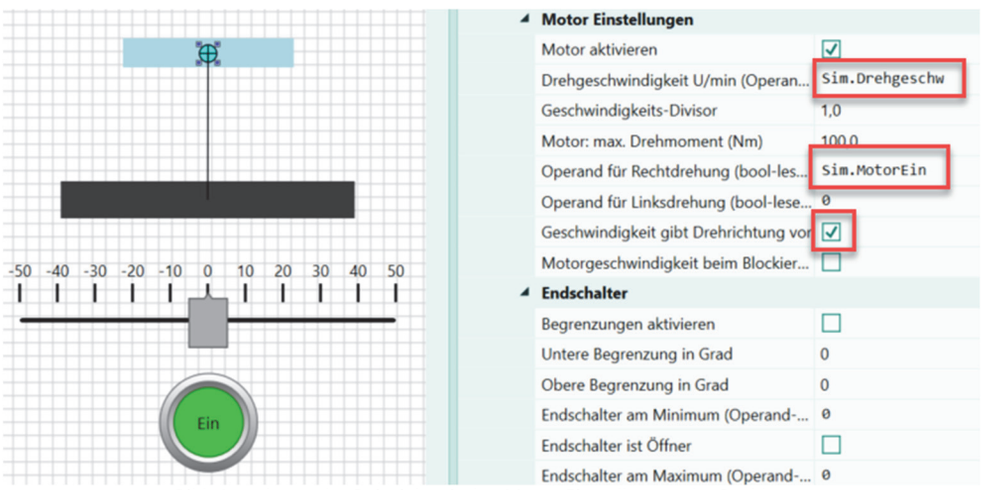


Bild: Anordnung des Beispiels mit den Motor-Einstellungen der Drehbewegung

Startet man die Simulation und betätigt den Schalter „Ein“, dann wird durch die Selektion der Drehgeschwindigkeit zusätzlich die Drehrichtung vorgegeben. Ist die Drehgeschwindigkeit positiv, dann hat dies eine Drehung im Uhrzeigersinn zur Folge (Drehung nach rechts). Bei einem negativen Wert ergibt sich eine Drehung entgegen dem Uhrzeigersinn (Drehung nach links).

13.2.3 Sensoren der Drehbewegung

13.2.3.1 Endschalter der Drehbewegung

Die Endschalter der Drehbewegung sind nur verwendbar, wenn die Bewegung begrenzt ist!

Es kann eine untere und eine obere Begrenzung in Grad angegeben werden. Dabei sind folgende Werte möglich:

- Untere Begrenzung: -180° bis 0°
- Obere Begrenzung: 0° bis 180°

Die Begrenzungen sind immer **relativ zur Lage** des Slave-Objekts und des Drehpunkts. Dabei ist die Drehbewegung im Uhrzeigersinn positiv, entgegen dem Uhrzeigersinn negativ.

Beispiel (VerbindungDrehbewegung04):

Eine Schranke soll über den Operanden E0.0 „Auf“ geöffnet und mit dem Operanden E0.1 „Ab“ geschlossen werden. Die beiden Endlagen werden über die Operanden E0.2 „EndschOffen“ und E0.3 „EndschGeschl“ erfasst. Alle Operanden sind dem Sim-Device zugeordnet.

Das Steuern der Operanden zum Öffnen und Schließen der Schranke erfolgt mit Schalter-Objekten, die als Taster konfiguriert sind. Die beiden Endlagen sind über LEDs visualisiert. Diese sind somit von den Endlagen-Operanden abhängig zu machen.

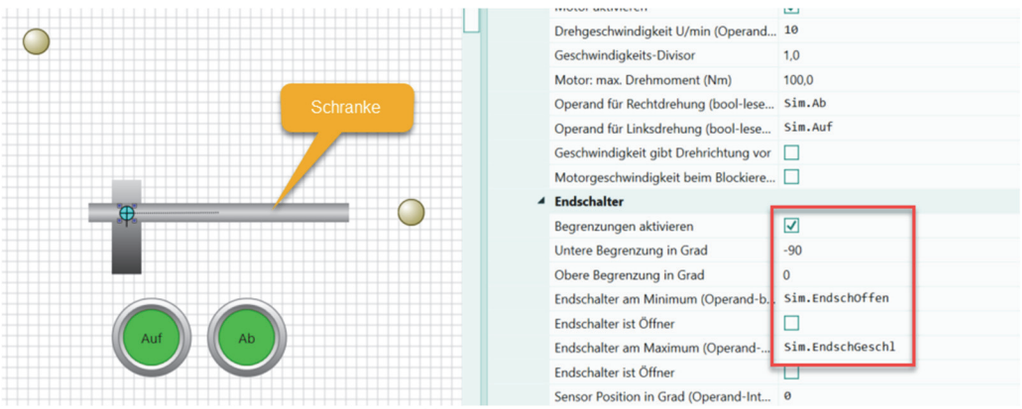


Bild: Anordnung des Beispiels

Im Bild ist die Verbindung für Drehbewegung selektiert. Innerhalb der Endschalter wurde die untere Begrenzung auf -90° eingestellt, die obere Begrenzung auf 0°. Somit kann sich die Schranke vom Drehpunkt aus um 90° nach links drehen. Die aktuelle Position ist die Begrenzung bei einer Rechtsdrehung. An den Eigenschaften „Endschalter am Minimum“ und „Endschalter am Maximum“ sind die beiden Endschalter-Operanden eingetragen. Diese könnten auch als Öffner konfiguriert werden.

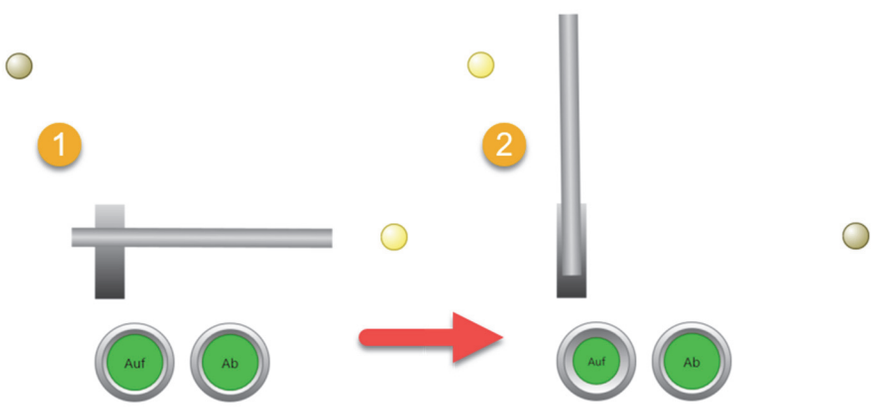


Bild: Simulation der Anordnung

Wird die Simulation gestartet, dann befindet sich die Schranke in der **unteren Endlage**, somit hat der Operand „EndschGeschl“ den Status 1 und die entsprechende LED leuchtet. Würde man in dieser Position den Taster „Ab“ betätigen, dann würde sich die Schranke nicht bewegen, da ja die untere Endlage bereits erreicht ist. Bei Betätigung von „Auf“ vollzieht die Schranke eine Drehung nach links und öffnet sich, bis die **obere Endlage** erreicht ist. Hier wird nun der entsprechende Operand auf den Status 1 gesetzt und die obere LED leuchtet.

Fazit: Mit Hilfe der Begrenzung kann bei einer Verbindung für Drehbewegung angegeben werden, in welchem Bereich sie sich bewegt. Bei der Einstellung ist zu beachten, dass zunächst die obere und untere Begrenzung in Grad anzugeben ist, dann erst kann die Option „Begrenzung aktivieren“ selektiert werden. Ist eine Begrenzung vorhanden, dann sind auch die Endschalter am Minimum und am Maximum aktiv und können mit Operanden belegt werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.2.3.2 Der Sensor Position in Grad

Die Verbindung für Drehbewegung verfügt über einen Sensor, mit dem die aktuelle Position des Slave-Objekts in Grad ausgelesen werden kann. Dies bedeutet, die Grad-Zahl wird in den angegebenen Operanden geschrieben. Der Operand muss dabei einen Zahlenbereich von 0 bis 360 aufnehmen können.

Beispiel (VerbindungDrehbewegung05):

Zwei dynamische Rechtecke werden zu einem Körper zusammengefügt und als Kreuz angeordnet. Dieses Kreuz soll rotieren. Dazu wird einer Verbindung für Drehbewegung zwischen einem der beiden dynamischen Rechtecke und einem darunter platzierten feststehenden Rechteck erzeugt.

Die Steuerung der Drehbewegung erfolgt über die beiden Operanden E0.0 „Rechts“ und E0.1 „Links“, die beide dem Sim-Device zugehörig sind. Die Umschaltung der Operanden erfolgt mit Hilfe von Schalter-Objekten, welche als Taster ausgelegt werden. Die Position der Drehbewegung wird in den Operanden EW2 „Pos“ geschrieben und mit Hilfe einer digitalen Anzeige visualisiert.

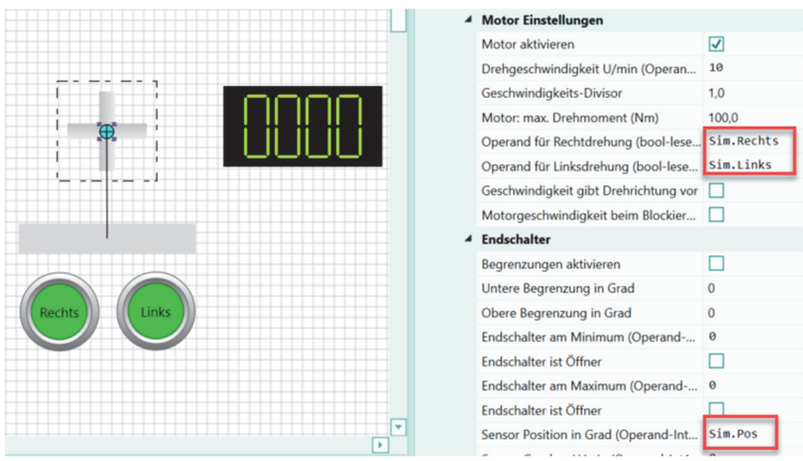


Bild: Anordnung und Einstellung der Verbindung für Drehbewegung

Startet man die Simulation und betätigt einen der beiden Taster, dann wird die Drehung nach links oder rechts ausgeführt. Des Weiteren ist die aktuelle Position in der digitalen Anzeige zu sehen. Der Positionswert variiert dabei im Bereich 0 bis 360.

Bild: Simulation der Anordnung

13.2.3.2.1 Skalierung des Positions-Sensors

Im letzten Beispiel wurde der Sensor „Position in Grad“ der Drehbewegung verwendet. Die Drehbewegung lieferte dabei Werte von 0° bis 360°. Oftmals werden in realen Anlagen Positionswerte in einem anderen Werte-Bereich geliefert. Auch diese Fälle können mit PLC-Lab simuliert werden.

Beispiel (VerbindungDrehbewegung06):

Es ist die Anordnung des letzten Beispiels als Grundlage zu verwenden. Allerdings soll der Positions-Sensor die Werte 0 bis 2000 in den Operanden EW2 „Pos“ schreiben.

Die Änderung gegenüber dem letzten Beispiel ist sehr einfach. Es ist lediglich die Eigenschaft „Sensor Position in Grad“ um den Bereich 0 bis 2000 zu erweitern.

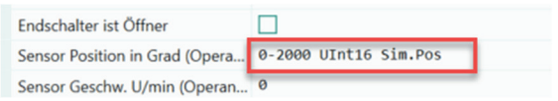


Bild: Erweiterung des Beispiels

Die Angabe des Bereichs bewirkt die Skalierung auf den neuen Bereich: In den Operanden werden die Werte 0 bis 2000 geschrieben.

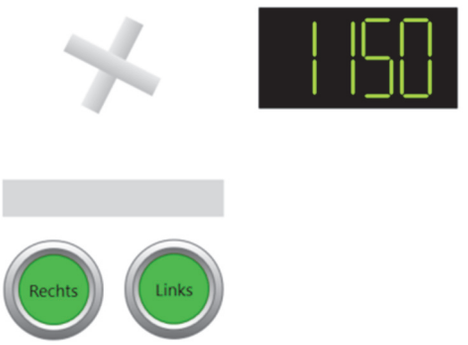


Bild: Simulation der Anordnung mit Angabe des Skalierungs-Bereichs

Somit ist es sehr einfach die Werte zu erzeugen, welche auch in der realen Anlage vom Sensor geliefert werden.

13.2.3.2.2 Realisierung von Endschaltern an verschiedenen Positionen

Ist es notwendig, die Position einer Drehbewegung mit einem Endschalter zu erfassen, so kann natürlich ein Endschalter an der entsprechenden Stelle platziert werden, damit er „mechanisch“ ausgelöst wird.

Einfacher ist es allerdings, den Wert des Positions-Sensors mit Hilfe eines **Rechner-Objekts** auszuwerten und damit einen Bit-Operanden zu beeinflussen.

Beispiel (VerbindungDrehbewegung07):

Der Positions-Sensor einer Drehbewegung schreibt die Wert 0 bis 2000 in den Operanden EW2 „Pos“. Bei einer Position von 1500 bis 1600 Einheiten soll der Operand E0.2 „Endsch1500Bis1600“ den Status 1 besitzen. Der Status des Operanden ist mit Hilfe eines LED-Lampen-Objekts zu visualisieren.

Für die Umsetzung des Beispiels wird ein Rechner-Objekt verwendet. Dieses Objekt befindet sich in der gleichnamigen Rubrik. Die Eigenschaft „Anzuwendende Funktionsart“ ist auf „In Bereich“ einzustellen. Die weiteren Einstellungen sind nachfolgend zu sehen:

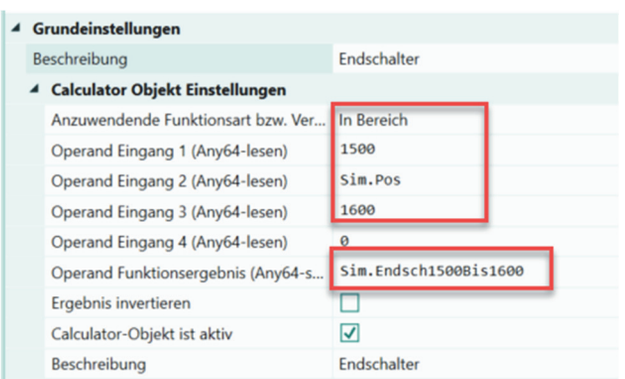


Bild: Einstellungen für das Rechner-Objekt

Am „Operand Eingang 1“ ist für diese Funktionsart der untere Wert des Bereichs anzugeben, an „Operand Eingang 2“ der Operand mit dem Vergleichswert und schließlich an „Operand Eingang 3“ der obere Wert des Bereichs.

Das Ergebnis der Funktion wird in „Operand Funktionsergebnis“ geschrieben. Hier ist im Beispiel E0.2 „Endsch1500Bis1600“ anzugeben.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Sobald sich die Drehbewegung an einer Position zwischen 1500 bis 1600 befindet, wird der Wert 1 in den Bit-Operanden geschrieben, andernfalls der Wert 0.

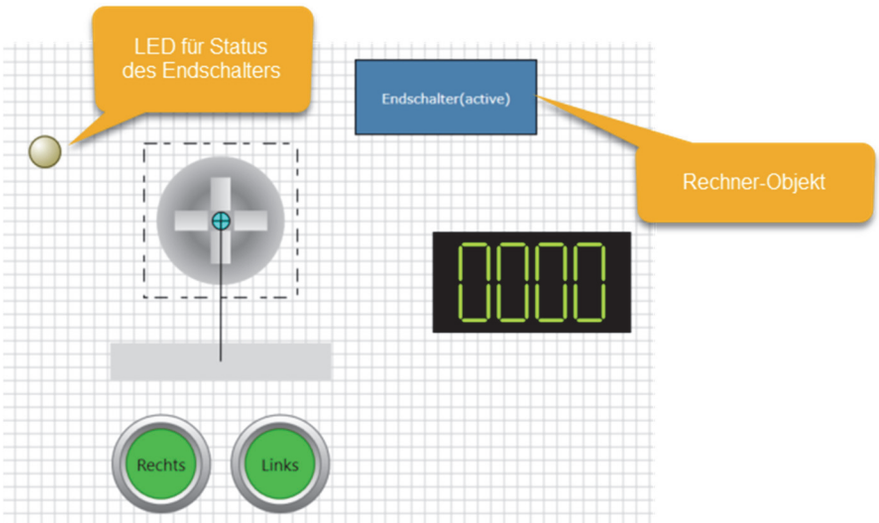


Bild: Anordnung für das Beispiel

Bei der Simulation zeigt sich folgendes Verhalten:

Bild: Simulation der Anordnung

Links ist eine Situation zu sehen, bei der sich die Drehbewegung in der Position **122** befindet. Somit hat der Operand E0.2 den Status 0 und die LED leuchtet nicht. Auf der rechten Seite des Bilds wurde die Position **1578** erreicht. Somit schreibt das Rechner-Objekt den Status 1 in den E0.2 und die LED leuchtet.

Auf diese Weise können an verschiedenen Positionen Endschalter realisiert werden.

13.2.4 Animierte Pumpe realisieren

Bei der Beschreibung der Rohrleitungs-Objekte wurde erwähnt, dass hierbei auch eine **Pumpe** enthalten ist. Diese wurde auch bereits in einem Beispiel in animierter Form verwendet. Dabei kam eine dynamische Ellipse zum Einsatz, welche mit Hilfe der Verbindung für Drehbewegung in Abhängigkeit vom Pumpen-Operanden rotiert wurde.

Somit ist zu erkennen, ob die Pumpe eingeschaltet ist und welche Drehrichtung aktiv ist. Nun soll auch noch die Drehgeschwindigkeit variiert werden.

Beispiel (VerbindungDrehbewegung08):

Eine Pumpe wird über den Operanden E0.0 „PumpeRechts“ im Rechtslauf eingeschaltet. Der Operand E0.1 „PumpeLinks“ ist für den Linkslauf verantwortlich. Die Geschwindigkeit der Pumpe wird über den Operanden EW2 „PumpeGeschw“ variiert. Alle Operanden sind dem Sim-Device zugeordnet.

Die beiden Bit-Operanden werden mit Hilfe von Schalter-Objekten beeinflusst, die dabei als Taster konfiguriert sind. Der Wert des EW2 soll mit Hilfe eines Schiebereglers im Bereich 10 bis 50 verändert werden.

Für das Beispiel ist zunächst die Pumpe aus der Rubrik „Rohrleitungen“ zu entnehmen und die Eigenschaft „Ist Physik-Objekt“ zu selektieren. Danach wird die Füllfarbe und die Füllfarbe bei Status 1 mit einem Deckungsgrad von 50% eingestellt.

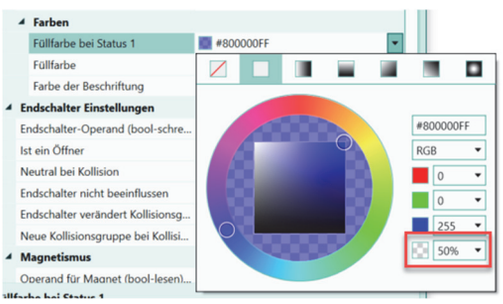


Bild: Deckungsgrad der Füllfarbe und der Füllfarbe bei Status 1 auf ca. 50% einstellen

Im nächsten Schritt wird eine dynamische Ellipse platziert und in der Mitte der Pumpe angebracht. Anschließend verbindet man die Pumpe und die Ellipse mit einer Verbindung für Drehbewegung und legt den Rotationspunkt in die Mitte der Pumpe.

Wichtig ist dabei, dass die Verbindung für Drehbewegung am Pumpengehäuse begonnen wird und an der Ellipse endet. Denn die Ellipse ist das rotierende Slave-Objekt.

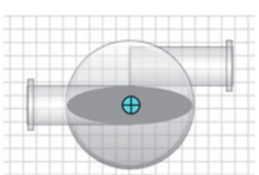


Bild: Pumpe und dynamische Ellipse über Verbindung für Drehbewegung verbunden

Danach sind noch die beiden Operanden zum Einschalten der Pumpe und ihrer **Drehgeschwindigkeit** anzulegen und zu verknüpfen.

Bild: Einstellung der Eigenschaften der Verbindung für Drehbewegung

© MHJ-Software GmbH & Co. KG - PLC-Lab

Da das Pumpen-Gehäuse sowohl bei der Rechts- als auch Links-Drehung seine Farbe verändern soll, sind die beiden Operanden über eine ODER-Funktion zu verknüpfen und das Ergebnis wird im Operanden M0.0 „PumpeIstEin“ des IM-Device abgelegt. Zu diesem Zweck wird ein Rechner-Objekt platziert und wie folgt eingestellt:

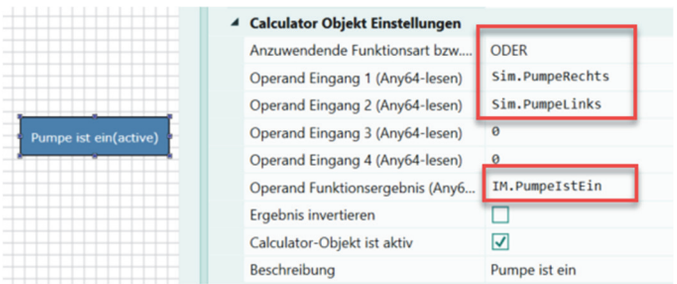
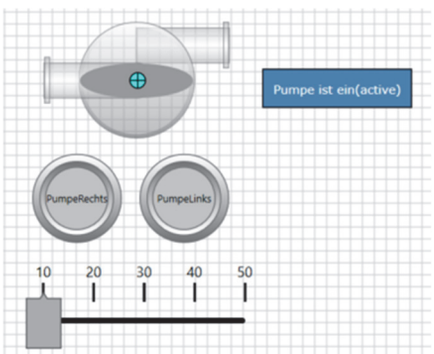


Bild: Einstellung des Rechner-Objekts

Der Operand M0.0 „PumpeIstEin“ ist an der Eigenschaft „Operand für Farbveränderung“ der Pumpe anzugeben.

Zuletzt wird ein **Schieberegler** platziert, um die **Drehgeschwindigkeit** zu verändern und die beiden Taster zum Einschalten der Pumpe für die jeweilige Drehrichtung. Damit ist die Anordnung komplett und hat folgendes Aussehen:

Bild: Komplette Anordnung des Beispiels



Nun kann man die Simulation starten, die Pumpe einschalten und die Drehzahl verändern. Alle Veränderungen werden über die animierte Pumpe dargestellt.

13.3 Verbindung mit konstanter Distanz

Diese Verbindung zwischen zwei Objekten ist vergleichbar mit einer **Stange**, wobei die Haltepunkte drehbar gelagert sind.

Bild: Prinzipdarstellung der Verbindung mit konstanter Distanz

Über die Eigenschaften „Steifigkeit“ und „Dämpfung“ kann der Charakter der Verbindung verändert werden.

Steifigkeit

Hier sind die Werte 1 bis 30 zulässig. Beim Wert 1 (soft) bekommt die Verbindung eher die Eigenschaften einer Kette mit relativ großen Kettengliedern. Der Standardwert ist 29.

Dämpfung

Der Wertebereich ist von 0.0 bis 1.0, womit das Verhalten von Hartgummi bis Metall variiert werden kann. Als Standard ist der Wert 0.9 eingestellt.

Diese Art der Verbindung kommt oft zum Einsatz, wenn man Mutter-Objekte (wird noch erläutert in Kap. 14) an der momentanen Stelle halten möchte oder auch zum Realisieren von Pendeln.

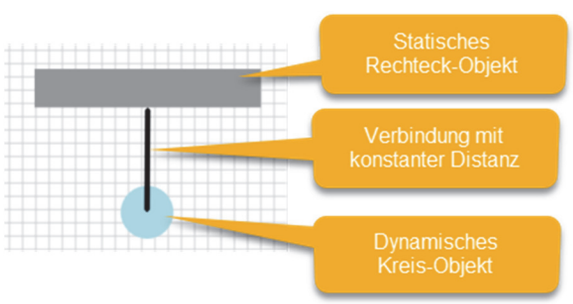


Bild: Beispiel einer Verbindung mit konstanter Distanz

Im Bild ist ein dynamisches Kreis-Objekt über eine Verbindung mit konstanter Distanz an ein statisches Rechteck angebunden. In den Standard-Einstellungen der Verbindung zeigt sich beim Start der Simulation folgendes Verhalten:

Bild: Simulation der Anordnung

1. Ruhestellung beim Start der Simulation.
2. Das Kreis-Objekt wird mit der **Maus** nach links oben gebracht und losgelassen.
3. Da das dynamische Kreis-Objekt schwerkraftabhängig ist, fällt es nach unten und wird aber von der Verbindung mit konstanter Distanz auf einer Kreisbahn gehalten. Es schwingt wie bei einem Pendel nach rechts (Punkt 6), um sich dann wieder zurückzubewegen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.3.1 Halte- bzw. Ankerpunkte verändern

Selektiert man die Linie, welche die Verbindung symbolisiert und klickt mit der Maus an eins der Enden, dann werden beim zugehörigen **Anker-Objekt** die Standard-Anker-Punkte hervorgehoben.

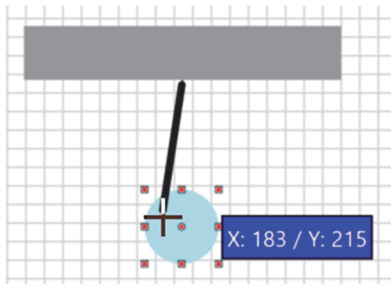


Bild: Selektion eines Ankerpunkts für das Kreis-Objekt

Man kann dann den Anker-Punkt an einem solchen Punkt andocken. Ebenso sind Anker-Punkte an einer anderen Stelle möglich, die sich aber innerhalb des jeweiligen Anker-Objekts befinden sollten.

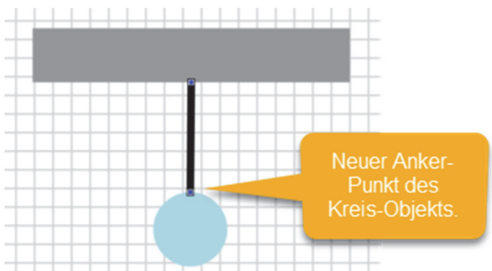


Bild: Es wurde ein neuer Anker-Punkt für das Kreis-Objekt selektiert.

13.4 Verbindung zu einem Körper

Immer wenn zwei dynamische Objekte **unlösbar** miteinander zu verbinden sind, kommt die Verbindung zu einem Körper zum Einsatz.

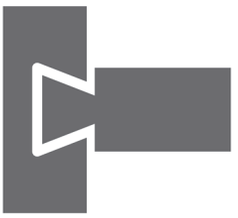


Bild: Prinzipdarstellung der Verbindung zu einem Körper

Bei dieser Art der Verbindung ist ein **Master-Objekt** sowie ein **Slave-Objekt** vorhanden. Beim Zeichnen sollte die Verbindung immer ausgehend vom „dominanteren" Objekt begonnen werden. Das Slave-Objekt „**erbt**" dabei einige Eigenschaften, darunter die Schwerkraftabhängigkeit vom Master-Objekt. **Diese Eigenschaft ist fortan nur noch über das Master-Objekt umschaltbar.**

Am **Verbindungsymbol** ist zu erkennen, welches Objekt das Master-Objekt ist, denn es befindet sich auf der Seite mit der durchgezogenen Linie.

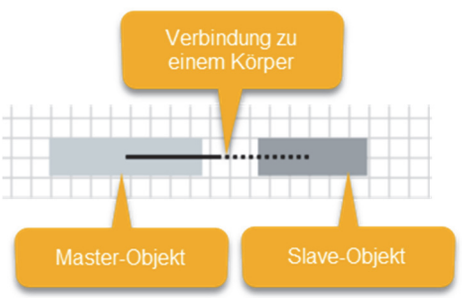


Bild: Verbindung zu einem Körper mit zwei dynamischen Rechtecken

13.4.1 Lücken zwischen den Objekten

Objekte, welche über eine Verbindung zu einem Körper miteinander verbunden sind, müssen sich nicht direkt berühren. Es können also auch Lücken zwischen den Objekten vorhanden sein. Diese Lücken können von anderen Objekten durchdrungen werden, wie dies z. B. bei einem **Sieb** der Fall ist.

Sinnvoll kann dies beispielweise sein, wenn man einen Endschalter weit außerhalb einer Linear- oder Dreh-Bewegung auslösen möchte.

Beispiel (VerbindungKoerper02):

Ein Zylinder mit Federrückstellung befindet sich innerhalb einer Röhre. An der Kolbenstange des Zylinders ist ein Nocken befestigt, welcher sich außerhalb der Röhre befindet und dort einen Endschalter betätigen soll.

Der Zylinder fährt aus, sobald der Operand E0.0 „Zyl" den Status 1 besitzt, die Umschaltung erfolgt mit Hilfe eines Schalter-Objekts, welches als Taster konfiguriert ist. Der Endschalter schaltet den E0.1 „Endschalter". Dieser Operand schaltet auch die Füllfarbe des Endschalters um. Alle Operanden sind dem Sim-Device zugeordnet.

Bei der Umsetzung wird zunächst der Zylinder mit Federrückstellung horizontal platziert. An dessen Eigenschaft „Operand Zylinder ausfahren" ist der Operand E0.0 „Zyl" anzugeben. Oberhalb und unterhalb des Zylinders wird mit zwei Dekorations-Rechtecken die Röhre angedeutet.

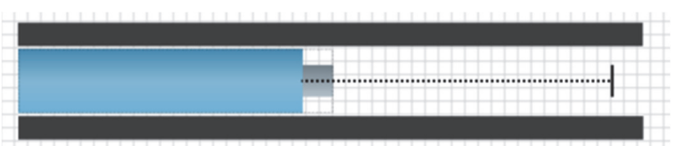


Bild: Zylinder innerhalb der Röhre

© MHJ-Software GmbH & Co. KG - PLC-Lab

Über der Röhre, etwa in der Mitte des Zylinder-Körpers, wird ein dynamisches Rechteck platziert. Dies ist der Nocken, welcher den Endschalter betätigen soll.

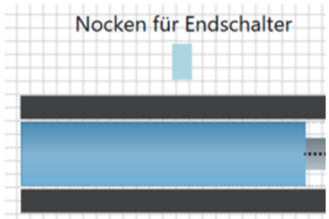


Bild: Platzierung des Nockens für den Endschalter

Im nächsten Schritt ist der Nocken mit der Kolbenstange des Zylinders über eine Verbindung zu einem Körper zu verbinden. **Dabei ist darauf zu achten, dass die Verbindung an der Kolbenstange begonnen wird, damit sie das Master-Objekt ist.**

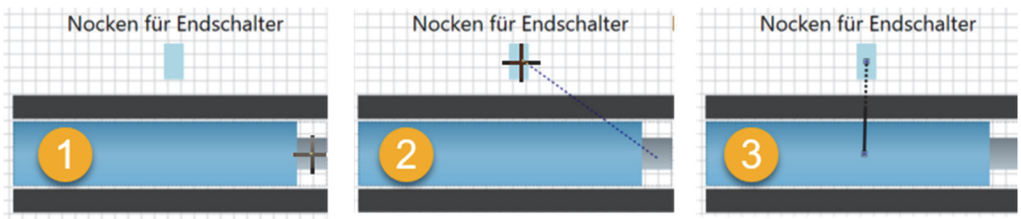


Bild: Verbindung zu einem Körper erzeugen

Danach wird der Endschalter platziert und ihm der Operand E0.1 „Endschalter“ zugeordnet. Der Operand wird auch in der Eigenschaft „Operand umschalten Füllfarbe“ angegeben.

Fehlt nur noch der Taster für die Beeinflussung des E0.0 „Zyl“, dann ist die Anordnung komplett.

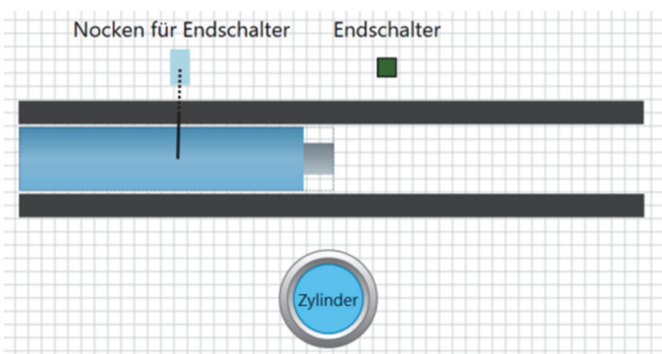


Bild: Komplette Anordnung

Nach dem Start der Simulation kann der Zylinder mit Hilfe des Tasters ausgefahren werden.

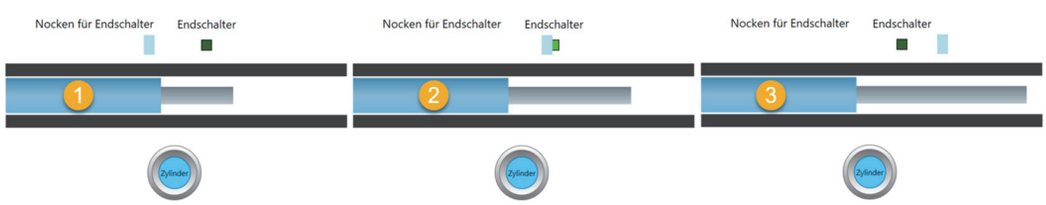


Bild: Simulation der Anordnung

Mit der Kolbenstange wird auch der Nocken bewegt. Dieser betätigt dann den Endschalter außerhalb der Röhre.

Fazit: Das Beispiel hat gezeigt, dass bei einer Verbindung zu einem Körper auch eine Lücke zwischen den beiden Anker-Objekten vorhanden sein kann. Dies ist insbesondere dann sinnvoll, wenn man das Slave-Objekt weit außerhalb des Einflussbereichs einer etwaigen Bewegung des Master-Objekts platzieren möchte, um beispielsweise Endschalter auszulösen.

13.4.2 „Verbindung zu einem Körper“ vs. „Körper-Gruppen“

Es gibt zwei verschiedene Varianten, Körper in PLC-Lab zu bilden. Zum einen die gezeigte Variante mit Hilfe der **Verbindung zu einem Körper**, zum anderen mit der sog. **Körper-Gruppe** innerhalb der Aktionen.

Beide Varianten führen zum gleichen Ergebnis. Die so miteinander verbundenen Objekte bilden einen Körper und sind in der Simulation **nicht mehr voneinander zu trennen**.

Körper-Gruppe:

Mit der Körper-Gruppe ist es einfacher und schneller eine größere Anzahl an Objekten miteinander zu verbinden. Dabei wird die Gruppe mit einem Rahmen versehen und sie verhält sich wie eine Zeichnen-Gruppe, kann also z. B. als Ganzes rotiert werden.

Verbindung zu einem Körper:

Sind nur zwei, drei Objekte miteinander zu verbinden, dann kann die Verbindung zu einem Körper zum Einsatz kommen. Ebenso, wenn man eine Körper-Gruppe beispielsweise an eine Zylinderstange anheften möchte. Dann wird von der Zylinderstange aus die Verbindung zu einem Element der Körper-Gruppe gezogen.

Ein weiterer Vorteil der Verbindung zu einem Körper besteht darin, dass man explizit festlegen kann, welches Objekt das Master-Objekt des Körpers sein soll.

Beispiel (VerbindungKoerper03):

Im Beispiel soll ein **Korb** aus fünf dynamischen Rechtecken gebildet werden. Dieser Korb ist dann fest mit der Kolbenstange eines Zylinders mit Federrückstellung zu verbinden. Der Zylinder wird über den Operanden E0.0 „ZylAusf“ gesteuert, welcher wiederum mit Hilfe eines Schalter-Objekts zu beeinflussen ist. Der Schalter ist als Taster konfiguriert.

Im ersten Schritt wird ein Rechteck in der Horizontalen aufgezogen, zweimal dupliziert und die Duplikate über dem Original platziert.

Bild: Die platzierten dynamischen Rechtecke

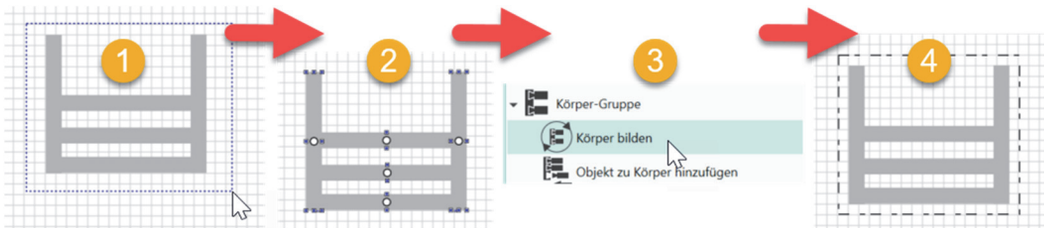
Nun wird ein weiteres Rechteck dupliziert und um 90° **rotiert**. Es ist dann für die linke Seitenbegrenzung zu verwenden. Eine Kopie des rotierten Rechtecks ist rechts anzubringen.

Bild: Platzierung der rotierten Rechtecke

Damit ist der Korb vollständig. Er muss nun zu einem Körper zusammengefügt werden. Dazu sind zunächst alle Objekte des Korbs mit dem Maus-Gummiband zu **selektieren**. Anschließend wird die Aktion „**Körper bilden**“ in der Rubrik „Körper-Gruppe“ selektiert.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Bild: Objekte des Korbs zu einem Körper zusammenfassen durch „Körper bilden“



In Punkt 4 des obigen Bilds ist die gebildete **Körper-Gruppe** zu sehen. Die Objekte werden von einem Rahmen umgeben, ähnlich einer Zeichnen-Gruppe. Die Handhabung der Körper-Gruppe ist identisch mit einer Zeichnen-Gruppe. Man kann die Gruppe selektieren und dann als Ganzes verschieben, rotieren usw.

Nun wird der Zylinder mit Federrückstellung erzeugt und horizontal platziert. Dann kann man den Korb mit Hilfe einer Verbindung zu einem Körper an der Kolbenstange des Zylinders anbringen.

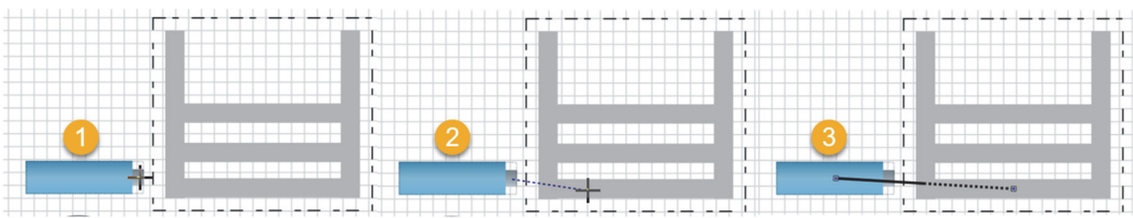


Bild: Verbindung zu einem Körper zwischen der Kolbenstange und dem Boden des Korbs

Im Bild ist zu erkennen, dass die Verbindung an der Kolbenstange begonnen wird, **damit sie das Master-Objekt ist**. Dann wird die Linie zum Boden des Korbs gezogen und die linke Maustaste losgelassen. Anschließend ist die Verbindung zu einem Körper zu sehen.

Damit ist der Korb fest an die Kolbenstange angebunden. Es fehlt nur noch der Taster für die Bedienung des Zylinders. Dann ist die Anordnung vollständig.

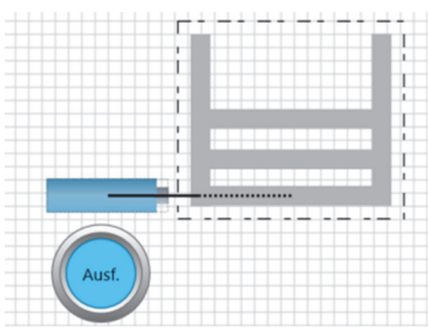


Bild: Anordnung für das Beispiel

Startet man die Simulation und betätigt den Taster, dann bewegt sich die Kolbenstange samt Korb nach rechts. Gleiches gilt, wenn der Taster losgelassen wird und die Kolbenstange einfährt.

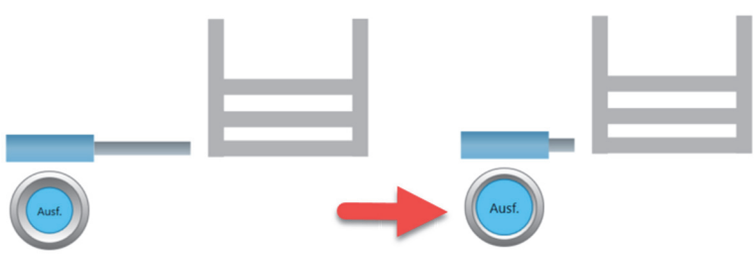


Bild: Simulation der Anordnung

Fazit des Beispiels: Das Beispiel hat gezeigt, wann es von Vorteil ist, eine Körper-Gruppe zu bilden und wann die Verbindung zu einem Körper eingesetzt werden kann. Die fünf dynamischen Rechtecke aus dem letzten Beispiel sind schneller mit einer Körper-Gruppe zusammenzufassen, zumal sie dann auch wie eine Zeichnen-Gruppe bearbeitet werden können.

Die Anbindung des Korbs an die Kolbenstange ist am einfachsten mit Hilfe der Verbindung zu einem Körper zu realisieren. Hier kann auch klar festgelegt werden, dass die Kolbenstange das Master-Objekt ist.

13.5 Schweiß-Verbindung (mit endlicher Festigkeit)

Im Gegensatz zur Verbindung zu einem Körper können Schweiß-Verbindungen brechen. Eine Schweiß-Verbindung verfügt also über eine **endliche Festigkeit**. Die max. Kraft, bis zu der die Schweiß-Verbindung widerstehen kann, ist über eine Konstante einstellbar.

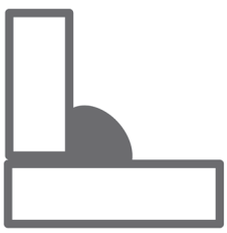


Bild: Prinzip-Bild der Schweiß-Verbindung

Eine Schweiß-Verbindung kann immer dann zum Einsatz kommen, wenn die Verbindung in der Simulation gelöst werden soll. Anderenfalls sollte die Verbindung zu einem Körper verwendet werden.

Werden zwei physikalische Objekte über eine Schweiß-Verbindung verbunden, dann sind beide Anker-Objekte gleichwertig. Es ist also kein Master- oder Slave-Objekt vorhanden.

Beispiel (SchweissVerbindung01):

Ein dynamisches Rechteck und ein dynamisches Dreieck sind über eine Schweiß-Verbindung miteinander verbunden. Unterhalb der beiden Objekte befindet sich ein statisches Rechteck, welches als Unterlage dient.

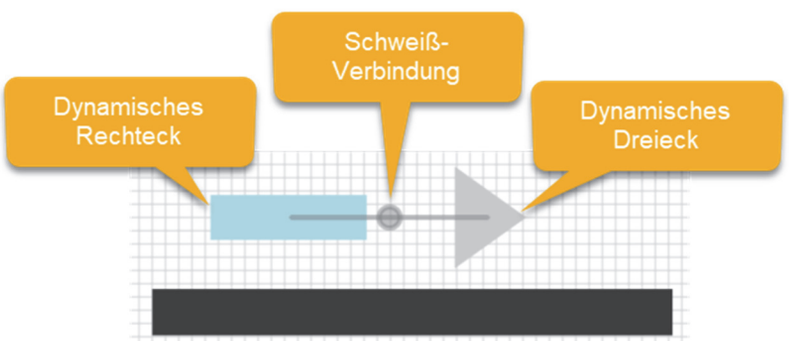


Bild: Anordnung des Beispiels

Nach dem Zeichnen sind die beiden Mittelpunkte der dynamischen Objekte miteinander verbunden. In der Mitte der **Schweiß-Verbindung** ist ein **Kreis** zu sehen. Beim Start der Simulation versucht die Schweiß-Verbindung die Anker-Punkte im Kreis miteinander zu vereinen. Dies wird in diesem Fall nur gelingen, wenn die beiden Anker-Objekte nicht miteinander kollidieren.

Bild: Start der Simulation

Ist in der Verbindung eingestellt, dass die Anker-Objekte kollidieren, …

Bild: Einstellung, dass die Anker-Objekte miteinander kollidieren

… dann wird die Verbindung brechen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

Bild: Beim Start der Simulation bricht die Schweiß-Verbindung.

Der Grund für das Brechen der Schweiß-Verbindung ist, dass die Objekte miteinander **kollidieren** und die dabei auftretenden **Kräfte** zum **Brechen** der Verbindung führen.

Beispiel (SchweissVerbindung03):

Die Anker-Punkte der Schweiß-Verbindung können mit der Maus verändert werden. Dazu wird die Verbindung selektiert, dann der gewünschte Anker-Punkt ausgewählt und mit der Maus verschoben. Im folgenden Bild ist dies am Beispiel des Anker-Punkts für das Dreieck zu sehen.

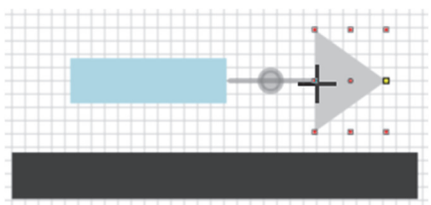


Bild: Verschieben des Anker-Punkts für das dynamische Dreieck

Als Hilfe werden die markanten Punkte des **Anker-Objekts** hervorgehoben. Man kann aber auch andere Punkte am Objekt auswählen.

Im Beispiel wurden die Anker-Punkte so gesetzt, dass die Objekte trotz Kollision direkt aneinanderhaften.

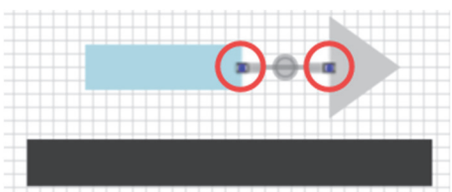


Bild: Neue Platzierung der Anker-Punkte

In der Anordnung wurde auch eine zweite Ebene mit einem statischen Dreieck und statischen Rechteck platziert.

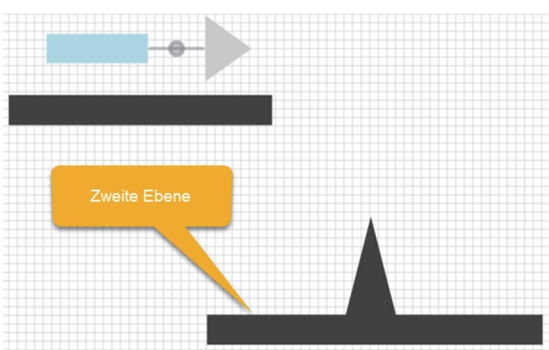


Bild: Erzeugen einer zweiten Ebene mit statischen Objekten

Des Weiteren wird die max. Kraft der Schweiß-Verbindung auf 30 N eingestellt.

Bild: Neue Einstellung für die Eigenschaft „Max. Kraft, bevor die Verbindung bricht“

Startet man die Simulation und lässt das Gebilde aus Rechteck und Dreieck auf die zweite Ebene herabfallen, dann **bricht** die **Schweiß-Verbindung** und die beiden Objekte werden voneinander **getrennt**.

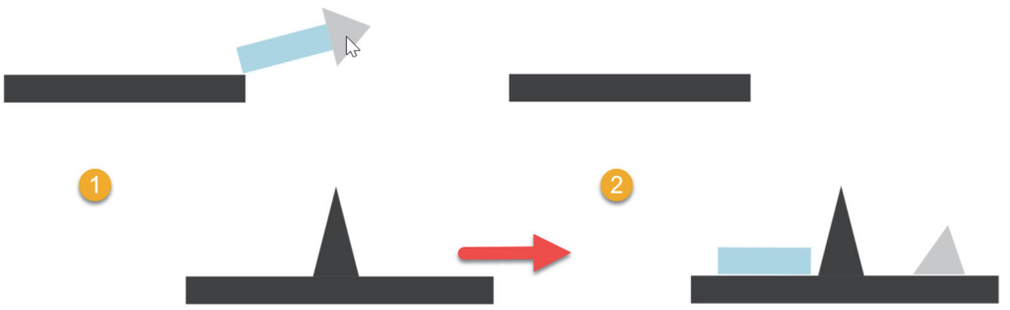


Bild: Simulation der Anordnung

Im obigen Bild ist dies zu sehen: Zunächst wird das pfeilähnliche Gebilde mit der Maus von der ersten Ebene nach rechts über den Abgrund gezogen und anschließend so fallen gelassen, dass die Verbindung der beiden Objekte auf das statische Dreieck fällt. Nach dem Aufschlag werden die beiden Objekte voneinander **getrennt**, da die Schweiß-Verbindung der auftretenden **Kraft** nicht standhalten kann.

13.5.1 Beispiel: Zerteilen einer Stange

Ein typisches Beispiel für die Anwendung von Schweiß-Verbindungen ist das Zerteilen einer Stange.

Beispiel (SchweissVerbindung04):

Eine Metallstange soll in einzelne Teile zerteilt werden. Dazu wird die Stange von einem horizontal angeordneten Zylinder mit Federrückstellung über einem Behälter platziert. Das Ausfahren des Zylinders ist mit der Konstanten 1 belegt, der Zylinder fährt somit sofort beim Start der Simulation aus.

Ein zweiter vertikal angeordneter Zylinder dient als Abscherer. Dieser **zerteilt** die Stange, indem er sehr schnell nach unten fährt. Dieser Vorgang wird über den Operand E0.0 „Zyl2“ des Sim-Device ausgelöst. Bei dem zweiten Zylinder handelt es sich ebenfalls um einen Zylinder mit Federrückstellung. Die Ausfahrgeschwindigkeit wurde dabei auf 10 m/s eingestellt. Die Umschaltung des E0.0 „Zyl2“ erfolgt über einen Taster.

Die Metallstange besteht aus Teilstücken, deren Länge der Lücke über dem Behälter entspricht. Die einzelnen Teilstücke sind über Schweiß-Verbindungen miteinander verbunden. Die max. Kraft der Verbindung wurde auf 30 N eingestellt. Des Weiteren ist einzustellen, dass die Anker-Objekte miteinander **kollidieren**, andernfalls würden die Teilstücke nach dem Abtrennen im Behälter nicht kollidieren.

Im folgenden Bild ist der Aufbau der **Metallstange** zu sehen.

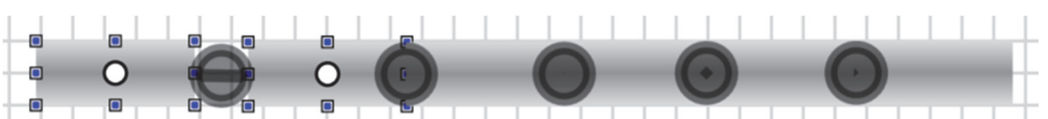


Bild: Aufbau der Metallstange aus den Teilstücken

Im Bild wurde das linke Teilstück etwas nach links versetzt, um dessen Länge zu zeigen.

Die gesamte Anordnung ist im nächsten Bild dargestellt.

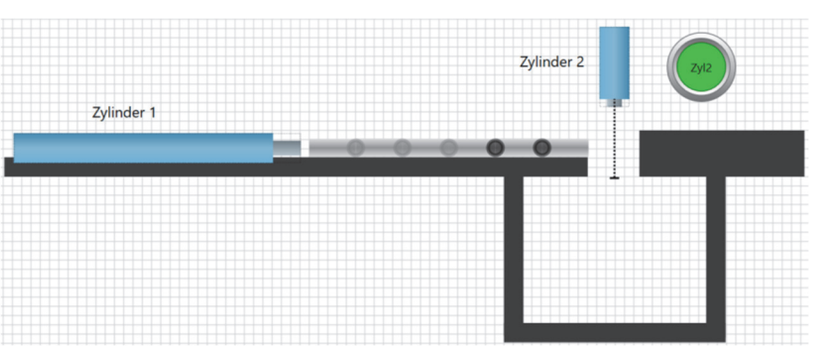


Bild: Anordnung für dieses Beispiel

© MHJ-Software GmbH & Co. KG - PLC-Lab

Beim Start der Simulation zeigt sich folgendes Bild:

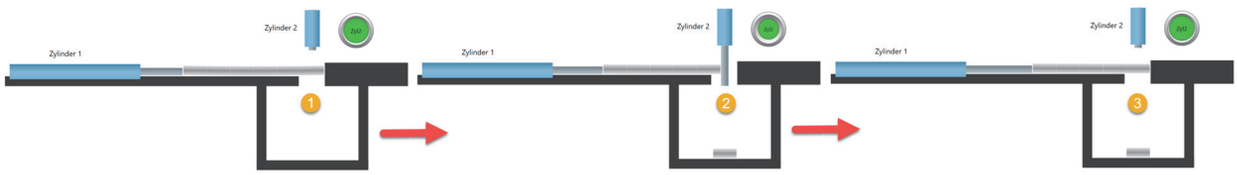


Bild: Simulation der Anordnung

Beim Einschalten (Punkt 1) wird die Metallstange durch den Zylinder 1 in Position gebracht, denn der Zylinder fährt sofort aus. Wird der Taster „Zyl2“ betätigt, dann fährt der Zylinder 2 nach unten und schert ein Teilstück der Metallstange ab. Lässt man den Taster los, dann bewegt sich der Zylinder 2 nach oben und der Zylinder 1 kann wieder etwas mehr ausfahren, bis die Metallstange am Block anliegt.

13.5.2 Schweißverbindung im Simulationsmodus erzeugen und (optional) wieder entfernen

Geometrische Formen besitzen die beiden Eigenschaften „Operand Schweißverbindung herstellen“ und „Operand Schweißverbindung lösen“. Damit kann während der Simulation eine **neue Schweiß-Verbindung** zwischen zwei kollidierenden Objekten **hergestellt** und auch wieder gelöst werden.

Wichtig: Nur eine im Simulationsmodus hergestellte Schweiß-Verbindung ist auch wieder lösbar!

13.5.2.1 Herstellen und Lösen einer Schweiß-Verbindung

Beispiel (SchweissVerbindung05):

Im folgenden Beispiel werden zwei dynamische Rechtecke **miteinander verschweißt**. Hierzu wird ein Dreieck mit Hilfe einer Verbindung für lineare Bewegungen an der Stelle positioniert, an der die beiden Rechtecke zu verbinden sind. Dazu wurde im Dreieck die Eigenschaft „Operand Schweißverbindung herstellen“ mit der Konstanten 1 belegt. Somit wird sofort bei Kollision mit den beiden Rechtecken die Schweiß-Verbindung hergestellt.

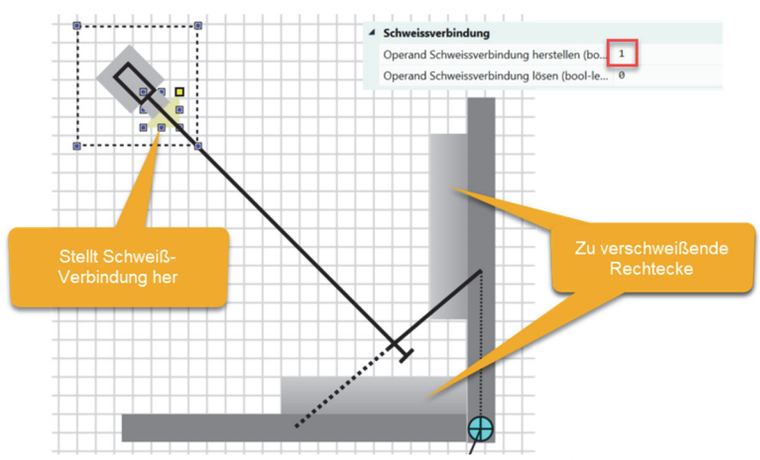


Bild: Anordnung zum Verschweißen der beiden Rechtecke

Die Vorrichtung kann anschließend um 120° nach rechts gedreht werden, damit die miteinander verbundenen Rechtecke auf eine darunter befindliche Ebene herabfallen.

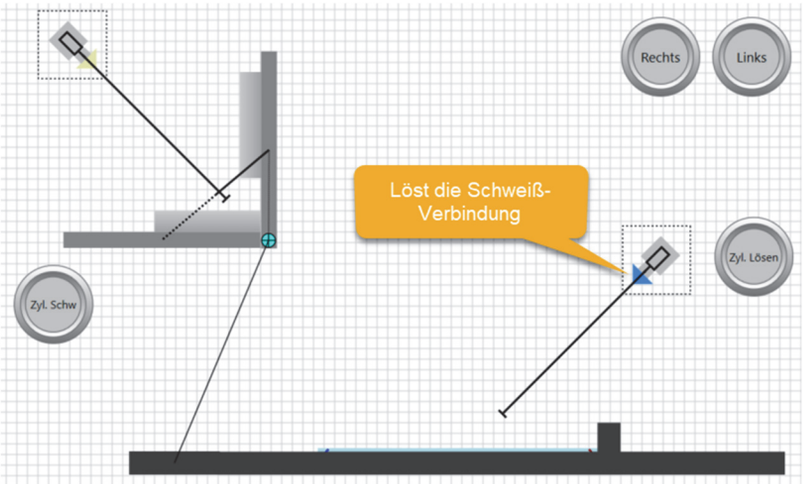


Bild: Anordnung des Beispiels

Auf der unteren Ebene werden die verschweißten Objekte mit Hilfe eines **Luftstroms** an einem Anschlag positioniert. An dieser Stelle wird ein weiteres Dreieck, ebenfalls mit Hilfe einer Verbindung für lineare Bewegungen, an der Verbindungsstelle der beiden Rechtecke platziert. An diesem Dreieck ist die Eigenschaft „Operand Schweißverbindung lösen“ mit der Konstanten 1 belegt. Somit wird bei Kollision des Dreiecks mit den beiden Rechtecken die Schweiß-Verbindung wieder gelöst.

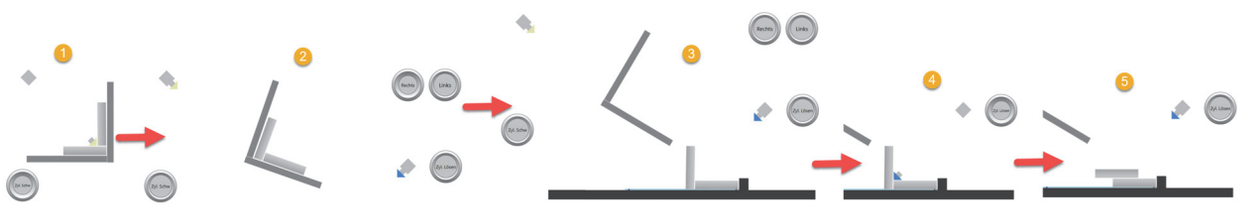


Bild: Simulation der Anordnung

Folgende Stationen der Simulation sind im obigen Bild dargestellt:

1. Über den Taster „Zyl. Schw“ wird das Dreieck für die Schweißverbindung zwischen die beiden dynamischen Rechtecken platziert. Diese sind somit verbunden.
2. Die Dreh-Vorrichtung mit den verbundenen Rechtecken wird nach rechts rotiert, damit die Rechtecke auf die zweite Ebene fallen.
3. Über den Luftstrom werden die verbundenen Rechtecke am Anschlag positioniert.
4. Über den Taster „Zyl. Lösen“ wird das Dreieck zum Lösen der Schweiß-Verbindung nach vorne bewegt, bis es die beiden Rechtecke berührt. Damit wird die Schweiß-Verbindung wieder gelöst.
5. Die beiden Rechtecke sind nicht mehr miteinander verbunden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.5.2.2 Weiteres Beispiel zum Herstellen und Lösen einer Schweiß-Verbindung

Beispiel (SchweissVerbindung06):

In diesem Beispiel wurde an einem Zylinder ein Rechteck angebracht (mit einer Verbindung zu einem Körper), bei dem die Eigenschaften „Operand Schweißverbindung herstellen“ und „Operand Schweißverbindung lösen“ jeweils mit Operanden belegt wurden. Die Operanden E0.0 „Schweißen“ und E0.1 „Lösen“ werden jeweils über Schalter beeinflusst. So kann selektiert werden, ob eine Schweißverbindung herzustellen oder zu lösen ist.

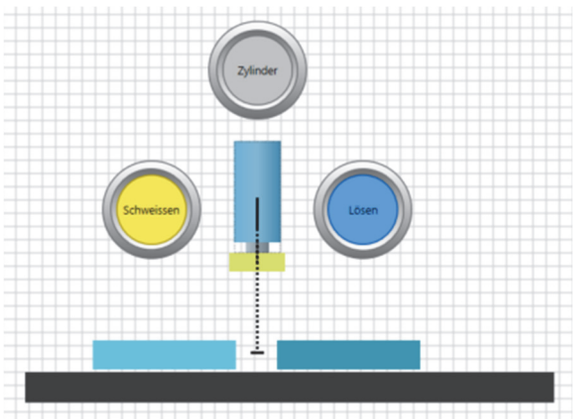


Bild: Anordnung für dieses Beispiel

Bei dem vertikal angeordneten Zylinder handelt es sich um einen Zylinder mit Federrückstellung. Er wird über den Operanden E0.2 „Zylinder“ gesteuert. Die Umschaltung des E0.2 erfolgt mit Hilfe eines Tasters.

Nachfolgend sind verschiedene Situationen der Simulation zu sehen:

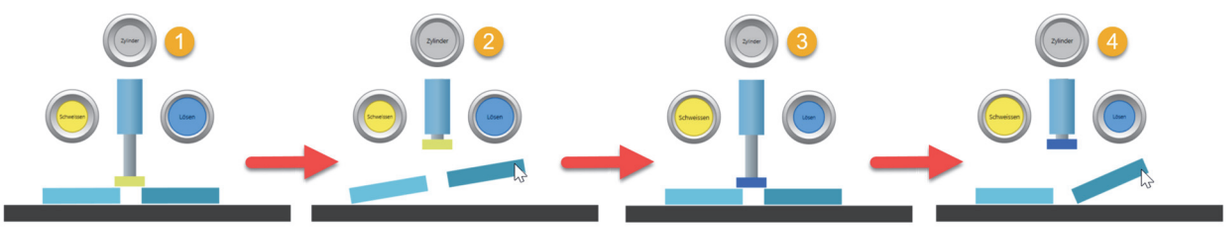


Bild: Simulation des Beispiels

1. Es wurde der Modus „Schweißen“ über den Schalter ausgewählt und der Zylinder ausgefahren. Somit werden die beiden Rechtecke **verschweißt**.
2. Das rechte Rechteck wird mit der Maus angehoben. Dabei wird das linke Rechteck ebenfalls mit bewegt, da die beiden miteinander verbunden sind.
3. Nun wird der Modus „Lösen“ selektiert und der Zylinder ist wiederum ausgefahren. Damit wird die Verbindung zwischen den beiden Rechtecken **gelöst**.
4. Als Folge hat das Anheben des rechten Rechtecks keinen Einfluss auf das linke.

Fazit: Die letzten Beispiele haben gezeigt, wie eine im Simulationsmodus hergestellte Schweißverbindung wieder gelöst werden kann. Beim letzten Beispiel wurde für das Verbinden und Lösen jeweils das gleiche Objekt verwendet, wobei die Umschaltung des Modus mit Hilfe von Bit-Operanden vorgenommen wurde. Auch diese Variante ist somit möglich.

13.6 Rad-Verbindung

Die Rad-Verbindung hat Ähnlichkeit mit der Verbindung für Drehbewegung. Dabei sollte die Rad-Verbindung verwendet werden, wenn man eine Drehbewegung mit **Stoßdämpfer-Eigenschaften** benötigt.

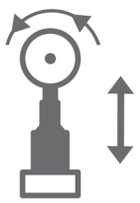


Bild: Prinzipdarstellung der Rad-Verbindung

Die Rad-Verbindung verbindet zwei Physik-Objekte miteinander, wobei ein Objekt das Master-Objekt darstellt. Dies ist im Normalfall der feststehende Teil der Verbindung, mit dem das Zeichnen begonnen werden muss.

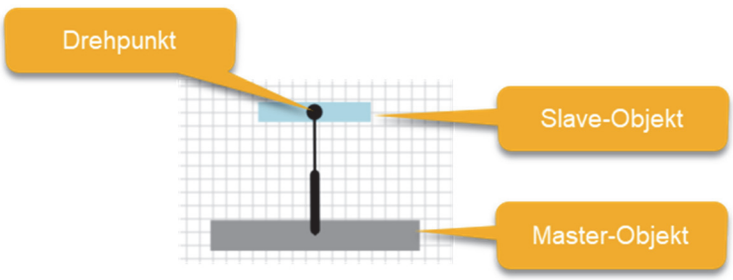


Bild: Elemente einer Rad-Verbindung

Das Symbol der Rad-Verbindung hat Ähnlichkeit mit einem Stoßdämpfer. Der dickere Teil ist am Master-Objekt angebracht, der dünnere am rotierenden Slave-Objekt. Der Drehpunkt der Verbindung wird über einen Kreis symbolisiert, welcher mit Hilfe der Maus verschoben werden kann.

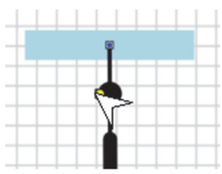


Bild: Verschiebung des Drehpunkts der Rad-Verbindung

13.6.1 Anker-Punkte verändern

Nach dem Zeichnen der Rad-Verbindung befinden sich die Anker-Punkte in der jeweiligen Mitte der Anker-Objekte. Selektiert man die Verbindung und verschiebt einen Anker-Punkt mit der Maus, dann werden im zugehörigen Anker-Objekt die Standard-Anker-Punkte hervorgehoben.

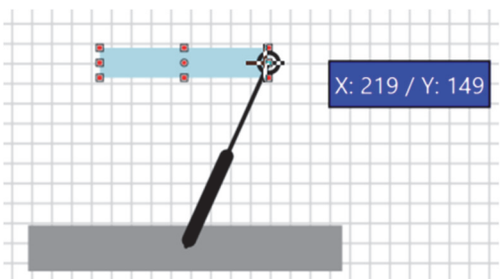


Bild: Standard Anker-Punkte des Slave-Objekts

Man kann diese vorgeschlagenen Anker-Punkte verwenden, es können aber auch andere Positionen innerhalb des Objekts selektiert werden. Theoretisch sind auch Anker-Punkte **außerhalb** der Anker-Objekte möglich, das Verhalten der Verbindung ist in diesem Fall aber nicht immer eindeutig.

© MHJ-Software GmbH & Co. KG - PLC-Lab

13.6.2 Motor der Rad-Verbindung

Eine Rad-Verbindung besitzt einen **Motor**, bei dem die Drehgeschwindigkeit mit Operanden oder auch über eine Konstante angegeben werden kann. Mit Hilfe des Geschwindigkeits-Divisors kann die Geschwindigkeits-Vorgabe wie bei der realen Anlage eingestellt werden, auch wenn sie in der Simulation nicht vernünftig darstellbar ist. In einem solchen Fall wird am Divisor eine Konstante angegeben, welche die Geschwindigkeits-Vorgabe in einen für die Simulation sinnvollen Bereich wandelt, ohne das Signal selbst für die Simulation speziell anpassen zu müssen.

Wie bereits von anderen Verbindungen mit Motor bekannt, kann mit Operanden die Drehrichtung vorgegeben werden oder man wählt die Option „Geschwindigkeit gibt Drehrichtung vor“. In diesem Fall wird der Motor über den Operanden für die Rechts-Drehung eingeschaltet und die **Polarität** der Geschwindigkeit bestimmt die **Drehrichtung** (negativ nach links, positiv nach rechts).

13.6.3 Sensoren der Verbindung

Die Rad-Verbindung verfügt über deutlich weniger Sensoren als die Verbindung für Drehbewegung.

Eine Besonderheit ist allerdings vorhanden: Die Rad-Verbindung kann die Amplitude des Stoßdämpfers sensorisch erfassen. Dabei werden die Werte von -100% bis +100% geliefert. Bei einem Wert von -100% ist der Stoßdämpfer vollständig ausgedehnt, bei +100% vollständig komprimiert. Mit Hilfe einer Bereichsangabe kann man diese Werte auf einen beliebigen Bereich skalieren. Dieser Sensor wird beispielsweise dazu verwendet, die **Unwucht** einer Drehbewegung auszuwerten. Hier kommt meist die Möglichkeit zum Einsatz, die Dichte von Objekten über digitale Operanden zu verändern und so eine Unwucht zu erzeugen. Zu diesem Zweck kann auch die **Steifigkeit** und die **Dämpfung** der Verbindung verändert werden, um z. B. die **Sensibilität** des Stoßdämpfers zu erhöhen.

Beispiel (RadVerbindung02):

Ein Rotor besitzt eine Unwucht. Der Rotor setzt sich dabei aus einem dynamischen Rechteck und einem dynamischen Kreis zusammen, welche über eine Verbindung zu einem Körper verbunden sind.

Bild: Rotor des Beispiels

Der Rotor bildet das Slave-Objekt der Rad-Verbindung. Für das Master-Objekt wird ein statisches Rechteck verwendet. Dabei sitzt der **Drehpunkt** in der **Mitte** des Rotors.

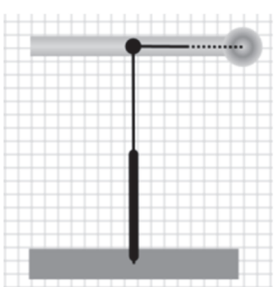


Bild: Die Rad-Verbindung

Die Drehzahl der Rad-Verbindung ist vom Operanden EW2 „Drehgeschw“ abhängig. Der Wert des EW2 wird über einen Schieberegler im Bereich 0 bis 200 verändert. In der Rad-Verbindung wird somit der Operand EW2 „Drehgeschw“ an der Eigenschaft „Drehgeschwindigkeit U/min“ eingetragen. An der Eigenschaft „Operand für Rechtsdrehung“ ist die Konstante 1 angegeben, somit vollzieht die Rad-Verbindung immer eine Drehung nach rechts, wenn die Drehgeschwindigkeit größer als 0 ist.

Innerhalb der Sensoren der Rad-Verbindung soll die **Amplitude** des Stoßdämpfers ausgewertet werden. Der Wert des Sensors ist in den Operanden EW4 „Amplitude“ zu schreiben. Dabei ist ein Bereich von -10 bis 10 vorzugeben.

Damit die Unwucht in der Lage ist, eine sichtbare Amplitude zu erzeugen, erhält die Steifigkeit der Rad-Verbindung den Wert 8 und es wird eine Dämpfung von 0,6 eingestellt. Nachfolgend sind die eingestellten Eigenschaften der Rad-Verbindung für das Beispiel zu sehen.

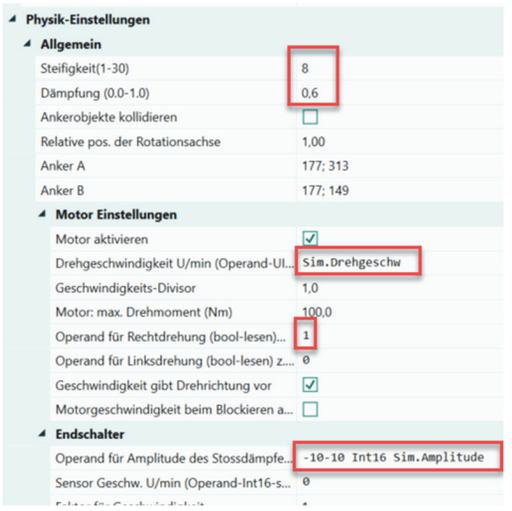


Bild: Eigenschaften der Rad-Verbindung für dieses Beispiel

Wie schon erwähnt, wird die **Unwucht** der Drehbewegung durch die Veränderung der **Dichte** des Kreis-Objekts am Rotor erzeugt. Die Dichte soll über den Operanden EW6 „DichteUnwucht“ vorgegeben werden. Die Beeinflussung des Operanden erfolgt über einen **Schieberegler** im Bereich von 0 bis 50, wobei der Startwert auf 1 zu setzen ist.

Der Operand EW6 „DichteUnwucht“ wird somit in die Eigenschaft „Dichte des Objekts“ des Kreis-Objekts eingetragen.

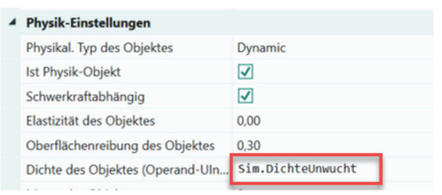


Bild: Variable Dichte des Kreis-Objekts

Die Visualisierung der **Amplitude** erfolgt über ein **Tacho-Objekt** aus der Rubrik „Anzeige-Objekte“. Dessen Bereich ist auf -10 bis 10 einzustellen und als Operand wird EW4 „Amplitude“ angegeben.

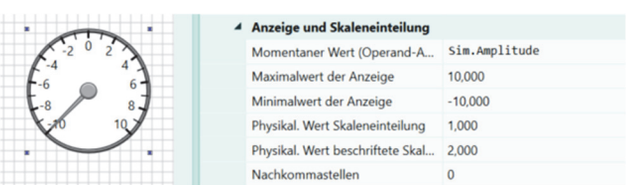


Bild: Tacho-Objekt zur Anzeige der Amplitude

Damit ist die Anordnung komplett.

Startet man die Simulation, erhöht man die Drehgeschwindigkeit und die Dichte der Unwucht, dann kann im Tacho-Objekt die Amplitude abgelesen werden. Ist die Unwucht relativ groß, dann ist dies auch der Drehbewegung selbst anzusehen.

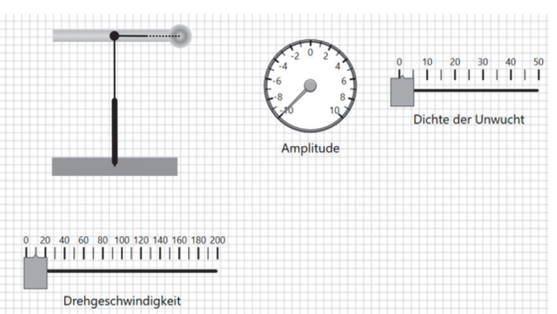


Bild: Gesamte Anordnung des Beispiels

Fazit: Die Rad-Verbindung kommt immer dann zum Einsatz, wenn die Stoßdämpfer-Eigenschaften einer Drehbewegung benötigt werden. Anderenfalls ist die Verbindung für Drehbewegungen zu verwenden, da diese Art der Verbindung mehr Sensoren bietet und flexibler in den Einstellungen ist.

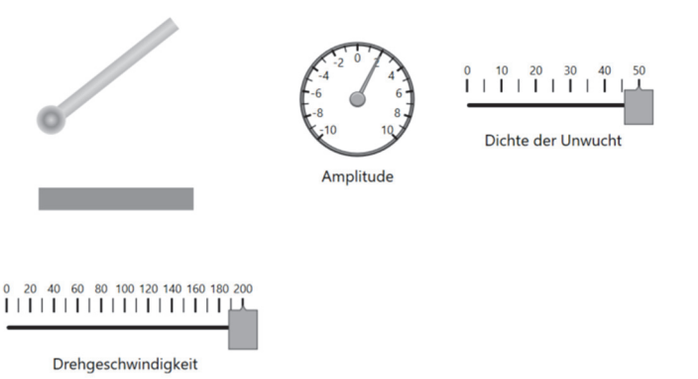


Bild: Simulation des Beispiels

© MHJ-Software GmbH & Co. KG - PLC-Lab

14 Creator und Destroyer

14.1 Creator

In vielen Anlagen besteht die Notwendigkeit **Objekte** zu „**produzieren**“, um beispielsweise Förderelemente mit zu befördernden Objekten zu versorgen. Es wird also eine **Quelle** benötigt, welche stetig Objekte mit bestimmten Eigenschaften **produziert**.

In PLC-Lab wird dies mit Hilfe der **Creator-Objekte** realisiert. Die Mutter-Objekte erhalten eine sog. **Creator-ID**, welche auch dem Creator-Objekt zugeordnet wird. Am Creator-Objekt wird dann definiert, wann die Kind-Objekte zu erzeugen sind, also welches Ereignis dazu notwendig ist. Tritt das Ereignis ein, dann wird eine Generation an Kind-Objekten erzeugt, welche genau die **Eigenschaften der Mutter-Objekte** besitzen.

Folgende Schritte sind notwendig, um ein dynamisches Objekt von einem Creator erzeugen zu lassen:

1. Erzeugen des dynamischen Objekts (z. B. einer dynamischen Ellipse).
2. Innerhalb der Rubrik „Grundeinstellungen->Creator/Destroyer Einstellungen“ in der Eigenschaft „Objekt ist ein **Mutterobjekt** mit der ID“ eine im Projekt eindeutige ID angeben. Diese ID muss im Bereich 1 bis 30 liegen. Sind mehrere verschiedene Mutter-Objekte gleichzeitig zu erzeugen, dann erhalte sie alle die gleiche ID.
3. Das Mutter-Objekt an der Stelle **fixieren**, an denen die Kind-Objekte erzeugt werden sollen. Hierzu wird oft ein feststehendes Objekt verwendet, an dem das Mutter-Objekt über eine **Verbindung mit konstanter Distanz** angebunden wird.
4. Creator-Objekt auf der Zeichenfläche erzeugen und an der Eigenschaft „Creator hat die ID“ die am Mutter-Objekt eingestellte ID eintragen.
5. An der Eigenschaft „Create-Trigger“ einen Operanden (oder eine Konstante) angeben, welcher als Trigger das Erzeugen der Mutter-Objekte steuert.
6. An der Eigenschaft „Trigger-Typ“ ist einzustellen, wann das Erzeugen erfolgen soll. Beispielsweise bei einer positiven Flanke des Operanden, der an „Create-Trigger“ angegeben wurde. Ist bei „Create-Trigger“ die Konstante '1' angegeben, dann muss als Typ „Bei Status 1 mit Pausezeit“ selektiert sein. In diesem Fall ist dann auch eine „Trigger-Pausezeit“ anzugeben.

Optional kann die max. Anzahl der **Create-Zyklen** (also die max. Anzahl der vorhandenen Generationen) in der Eigenschaft „Max. Create-Zyklen“ festgelegt werden. Ist diese max. Anzahl erreicht, dann werden keine neuen Kind-Objekt mehr erzeugt, auch wenn die Bedingung dazu erfüllt ist. Erst wenn die Anzahl der vorhandenen Generationen durch deren Zerstörung mit Hilfe von **Destroy-Objekten** unter der **maximalen Anzahl** liegt, können wieder neue Kind-Objekte erzeugt werden.

Die Mutter-Objekte können über die gesamte Anlage verteilt sein, sie müssen sich also nicht an einer Stelle befinden. Des Weiteren kann man die Mutter-Objekte über Verbindungselemente miteinander verbinden. Die Verbindungselemente werden automatisch erzeugt, wenn die Anker-Objekte die gleiche Creator-ID besitzen.

14.2 Destroyer

Der **Destroyer** ist das **Gegenstück** zum **Creator**: Er **entfernt** von einem **Creator** erzeugte Objekte. Dabei genügt die Berührung mit einem der Kind-Objekte für das Beseitigen der gesamten Generation. Dies bedeutet, werden beispielsweise drei Kind-Objekte gleichzeitig von einem Creator erzeugt, gehören sie also einer Generation an Kind-Objekten an, dann beseitigt ein Destroyer alle drei Objekte dieser Generation, wenn er mit einem der Kind-Objekte kollidiert.

Jedes Physik-Objekt kann ein Destroyer sein. Dazu ist lediglich in den Eigenschaften „**Creator/Destroyer Einstellungen**“ die Option „**Objekt ist ein Destroyer**“ zu selektieren. Zusätzlich kann eingestellt werden, welche ID die zu beseitigenden Kind-Objekte haben sollen. Wird hier die Zahl 0 angegeben, dann werden sämtliche Kind-Objekte unabhängig von ihrer ID zerstört.

14.3 Beispiel zu Creator und Destroyer

Beispiel (Creator01):

Im folgenden Beispiel sind zwei dynamische Rechtecke über eine Verbindung für lineare Bewegung miteinander zu verbinden. Beide Rechtecke erhalten die Mutter-ID 1.

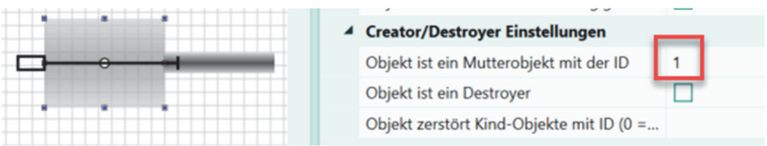


Bild: Rechtecke mit Verbindung und Mutter-ID

Der Verbindung für lineare Bewegung muss keine Mutter-ID gegeben werden, denn sind zwei Mutter-Objekte mit identischer ID über eine Verbindung gekoppelt, dann wird diese automatisch mit erzeugt.

Damit die beiden Mutter-Objekte an ihrer Position verbleiben, sind sie über eine Verbindung mit konstanter Distanz an einem statischen Rechteck anzubringen.

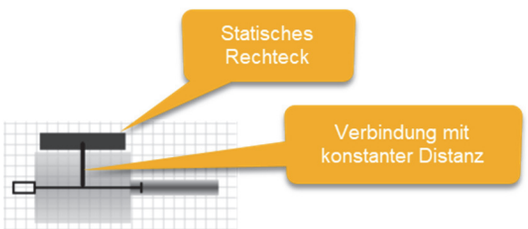


Bild: Die Mutter-Objekte werde an einer Stelle fixiert.

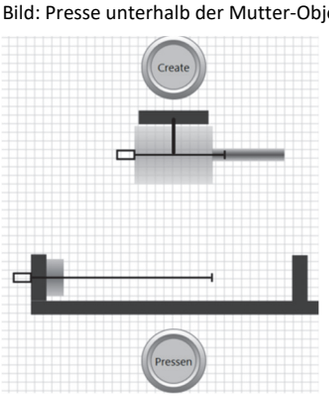
Im nächsten Schritt ist ein Creator-Objekt aus der gleichnamigen Rubrik zu erzeugen. Dem Creator-Objekt wird die ID 1 zugweisen, somit ist dieser für die beiden Mutter-Objekte zuständig. Das Erzeugen wird über den Operanden E0.0 „Create“ ausgelöst, sobald dieser eine positive Flanke erzeugt.

In der Anlage soll immer nur eine Generation von Kind-Objekten vorhanden sein. Aus diesem Grund ist in der Eigenschaft „Max. Create-Zyklen“ des Creator-Objekts die Zahl 1 anzugeben.

Bild: Einstellungen im Creator-Objekt

Wurden die Kind-Objekte erzeugt, dann fallen sie auf eine darunterliegende Ebene. Dort befindet sich eine Presse, welche über eine Verbindung für lineare Bewegung realisiert ist. Dabei bewegt sich ein dynamisches Rechteck nach rechts, sobald der Operand E0.1 „Pressen“ den Status 1 besitzt. Dessen Umschaltung erfolgt über einen Taster.

Bild: Presse unterhalb der Mutter-Objekte



In der Verbindung für lineare Bewegung ist die Eigenschaft „Geschwindigkeit umkehren bei Status 0 der Bewegungs-Operanden“ zu selektieren. Somit wird das Verhalten wie bei einem Zylinder mit Federrückstellung erreicht.

Nachdem der Press-Vorgang vollzogen wurde, sollen die verpressten Kind-Objekte aus der Anordnung **entfernt** werden können. Dieser Vorgang erfolgt in der realen Anlage manuell.

© MHJ-Software GmbH & Co. KG - PLC-Lab

In PLC-Lab soll dazu ein dynamisches Rechteck mit einem der beiden Kind-Objekte kollidieren und damit die gesamte Generation entfernen. Dazu wird im **Rechteck** die Eigenschaft „Objekt ist ein **Destroyer**“ ausgewählt und in der Eigenschaft „Objekt zerstört Kind-Objekte mit ID“ die Konstante 0 eingetragen. Somit beseitigt das Rechteck bei Kollision eine Generation an Kind-Objekten unabhängig von deren ID.

Damit das Entfernen der Kind-Objekte auf Knopfdruck geschieht, wird der Operand M0.0 „WerkstückEntfernen“ des IM-Device angelegt. Dieser Operand soll über eine Verbindung für lineare Bewegung das Rechteck mit einem Kind-Objekt kollidieren lassen. Dazu wird ein zusätzliches statisches Rechteck platziert, welches als Master-Objekt der Verbindung dient. Die Position zeigt folgendes Bild, in dem auch der Taster „Entfernen“ bereits vorhanden ist:

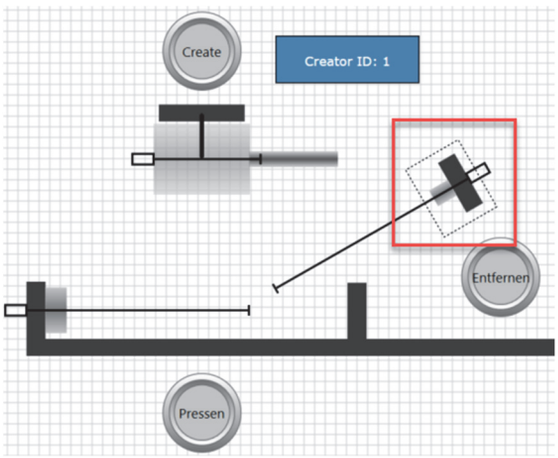


Bild: Vorrichtung zum Entfernen des Werkstücks

Die Verbindung für lineare Bewegung erhält die Geschwindigkeit 10 m/s, damit das Entfernen recht schnell erledigt wird. Bei der Verbindung ist die Eigenschaft „Geschwindigkeit umkehren bei Status 0 der Bewegungs-Operanden“ ausgewählt, somit fährt das Rechteck zurück, sobald der Taster nicht mehr betätigt ist.

Damit ist die Anordnung komplett und kann simuliert werden.

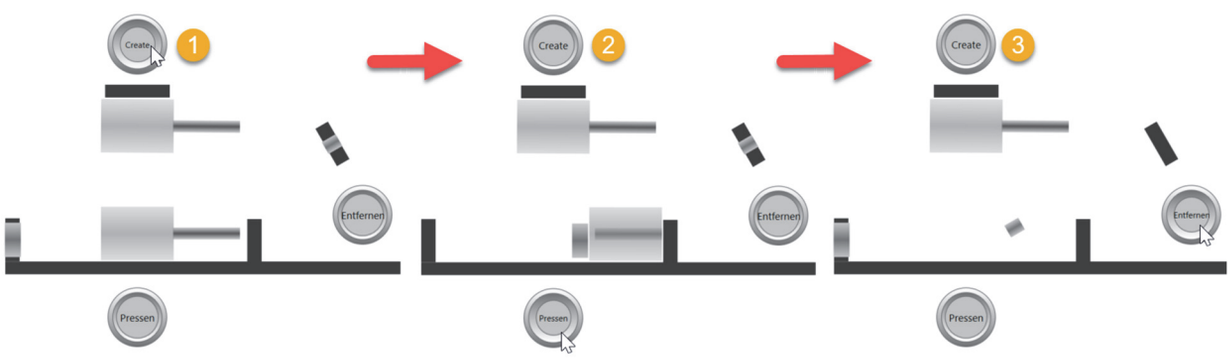


Bild: Simulation des Beispiels

1. Bei Betätigung von „Create“ wird eine **neue** Generation von Kind-Objekten erzeugt. Diese **fallen** nach unten. Ein erneutes Drücken von Create hat keine Auswirkung, da ja nur eine Generation von Kind-Objekten vorhanden sein darf. Dies wurde im Creator-Objekt über die Eigenschaft „Max. Create-Zyklen“ so eingestellt.
2. Der Taster „Pressen“ initiiert das Verpressen des Bolzens.
3. Bei Betätigung von „Entfernen“ fährt das Destroyer-Rechteck nach vorne und **entfernt** bei Berührung mit einem Kind-Objekt die gesamte Generation. Somit kann wieder eine neue Generation erzeugt werden.

14.4 Weiteres Beispiel zu Creator und Destroyer

Beispiel (Creator02):

Im nächsten Beispiel wird eine Beregnungsvorrichtung simuliert. Dabei kommt ein Luftstrom zum Einsatz für die Beschleunigung der Wassertropfen. Im Luftstrom ist ein Verstärkungsfaktor von 3 und eine Geschwindigkeit von 5 m/s einzustellen. Als Operand für die Bewegung nach rechts ist die Konstante 1 eingetragen. Somit ist der Luftstrom immer aktiv.

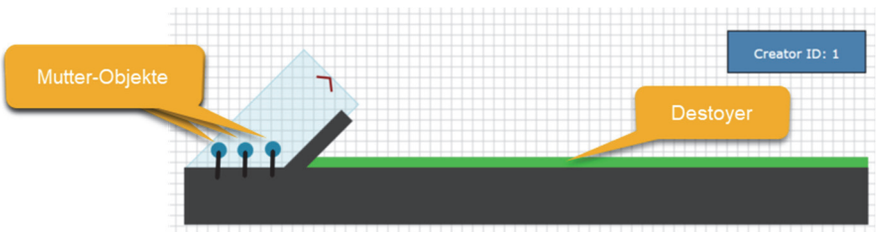


Bild: Anordnung des Beispiels

Die Mutter-Objekte besitzen die ID 1, es handelt sich dabei um dynamische Kreis-Objekte. Die grüne Fläche soll den Rasen symbolisieren und besteht aus einem statischen Rechteck, welches als Destroyer konfiguriert ist.

Bild: Einstellung des grünen Rechtecks; dies ist der Destroyer.

Zuletzt noch die Einstellungen des Creator-Objekts:

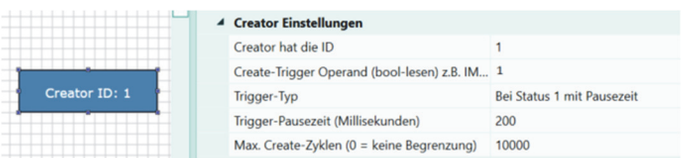


Bild: Einstellungen im Creator-Objekt

Beim Start der Simulation zeigt sich folgendes Bild:

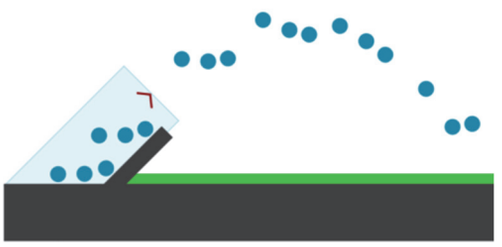


Bild: Simulation des Beispiels

Im 200ms-Takt werden neue Kreis-Objekte erzeugt und über den Luftstrom nach oben beschleunigt. Außerhalb des Luftstroms fallen sie dann auf die „Rasenfläche“. Bei der Kollision mit dem als **Destroyer** konfigurierten grünen Rechteck werden die Kind-Objekte wieder **zerstört**.

© MHJ-Software GmbH & Co. KG - PLC-Lab

15 Anzeige-Objekte

In dieser Rubrik finden sich Objekte zum **Anzeigen** des Inhalts von digitalen Operanden oder auch, um visuell auf besondere Sachverhalte innerhalb der Anordnung aufmerksam zu machen.

15.1 Text-Label

Das **Text-Label** kann zur einfachen Darstellung von **statischem Text** verwendet werden. Dazu stehen zwei Textzeilen zur Verfügung.

Die Füllfarbe des Labels ist mit Hilfe von Operanden umschaltbar, ebenso kann der Hintergrund **blinkend** oder **pulsend** eingestellt werden. Die Umschaltung der Anzeige-Modi ist ebenfalls über Operanden möglich, welche an den Eigenschaften „Operand blinkende Darstellung“ bzw. „Operand pulsende Darstellung“ angegeben werden können.

15.1.1 Text-Label für Darstellung eines Operanden-Inhalts verwenden

Neben einem statischen Text ist auch die Darstellung des Inhalts eines Operanden möglich. Dazu wird der Operand an der Eigenschaft „Operand“ eingetragen.

Innerhalb des Texts ist ein Platzhalter für den Wert des Operanden anzugeben, Beispiel „Messwert: {0}“. Dabei ist „{0}“ der **Platzhalter** für den **Wert** des **Operanden**.

Beispiel (TextLabel01):

Die Position der Kolbenstange eines Zylinders mit Federrückstellung wird in den Operanden EW2 „ZylinderPos“ geschrieben. Die Position ist dabei auf den Bereich 0 bis 1000 zu skalieren. Der Zylinder fährt aus, wenn der Operand E0.0 „ZylAusf“ den Status 1 besitzt. Die Umschaltung erfolgt mit Hilfe eines Tasters.

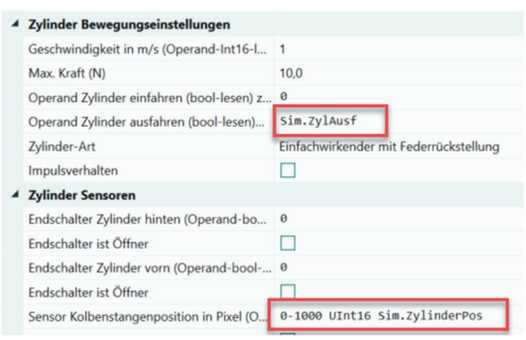


Bild: Einstellungen im Zylinder-Objekt

Der Wert des Operanden EW2 „ZylinderPos“ ist in einem Text-Label anzuzeigen. Es wird platziert und das Symbol „ZylinderPos“ per Drag-and-Drop auf das Text-Label gezogen. Damit ist es bereits fertig konfiguriert.

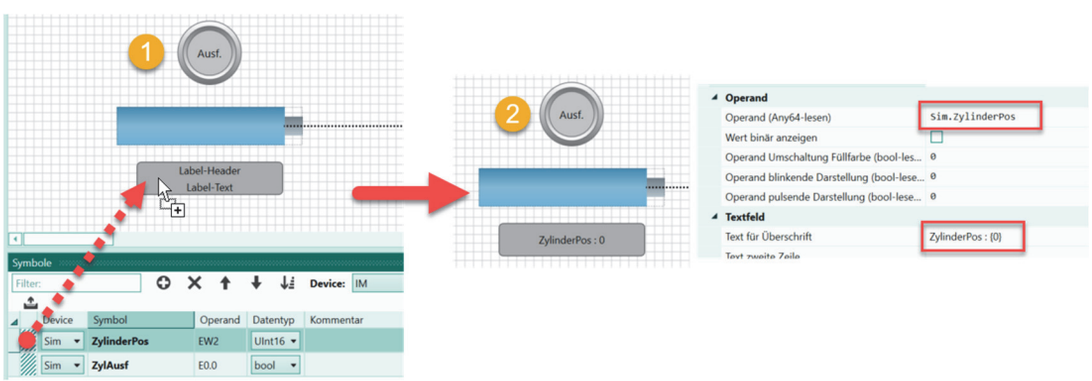


Bild: Eintragen des Operanden in das Text-Label per Drag-and-Drop

Nach der Aktion ist das EW2 „ZylinderPos“ im Text-Label als Operand eingetragen. Die Eigenschaft „Text für Überschrift“ wurde ebenfalls so angepasst, dass das Symbol des Operanden gefolgt von dessen Wert angezeigt wird.

Beim Start der Simulation und Steuern des Zylinders zeigt sich somit folgendes Bild:

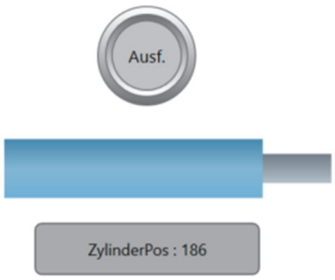


Bild: Simulation der Anordnung

Innerhalb des Text-Labels wird der aktuelle Wert des Operanden angezeigt.

15.1.1.1 Formatierung des anzuzeigenden Werts

Am Platzhalter innerhalb des Texts können auch **Formatierungsanweisungen** mit angegeben werden. Die Syntax lautet dabei „{0[:formatString]}“.

Beispiele dafür sind:

- Ausgabe des Werts in Hex: {0:X}
- Ausgabe des Werts in Hex mit mind. zwei Stellen: {0:X2}
- Ausgabe des Werts in dezimaler Darstellung mit mind. zwei Stellen: {0:D2}
- Ausgabe des Werts als Komma-Zahl mit einer Dezimalstelle: {0:f1}
- Ausgabe des Werts in exponentieller Darstellung: {0:e}

15.1.2 Text-Label ohne Rahmen

Möchte man das Text-Label ohne Rahmen darstellen, dann setzt man den Deckungsgrade der Füllfarbe auf 0%.

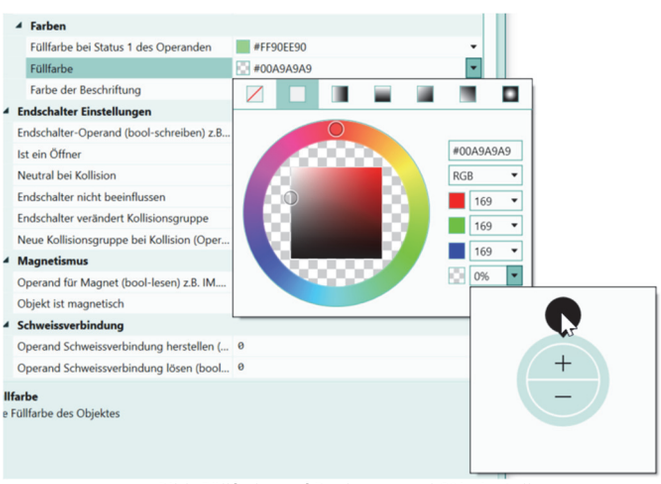


Bild: Füllfarbe auf Deckungsgrad 0% einstellen

Ebenso kann das Objekt „Text“ aus der Rubrik „Anzeige-Objekte“ verwendet werden. Hierbei handelt es sich um das Text-Label mit dieser Einstellung.

© MHJ-Software GmbH & Co. KG - PLC-Lab

15.2 Tacho-Anzeige

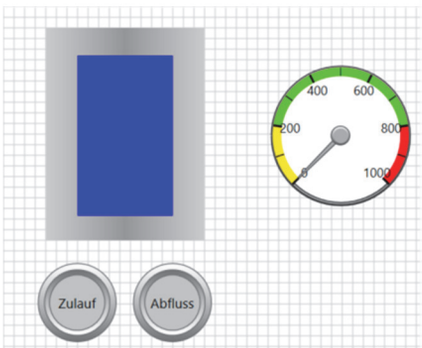
Die **Tacho-Anzeige** ist zur Darstellung von Zahlenwerten gedacht. Dazu wird der Anzeige ein Operand übergeben, dessen Inhalt gelesen und auf dem Tacho angezeigt wird.

Neben der Beschriftung der Skalenbereiche ist es auch möglich, Bereiche farbig zu kennzeichnen, um beispielsweise zu visualisieren, dass sich ein Wert im Normbereich oder aber darüber bzw. darunter befindet.

Beispiel (Tacho01):

Im Beispiel wird der Füllstand eines Tanks im Operand EW2 „Füllstand“ abgelegt und über ein Tacho-Objekt angezeigt.

Bild: Beispiel eines Tacho-Objekts mit farbigen Skalenbereichen



Anzeige und Skaleneinteilung	
Momentaner Wert (Operand-Any64-les...	Sim.Füllstand
Maximalwert der Anzeige	1000,000
Minimalwert der Anzeige	0,000
Physikal. Wert Skaleneinteilung	100,000
Physikal. Wert beschriftete Skaleneintei...	200,000
Nachkommastellen	0
Einfärben von Skalenbereichen	
Unterer Wert Farbbereich 1	0
Oberer Wert Farbbereich 1	200
Farbe für Farbbereich 1	#FFFFE900
Unterer Wert Farbbereich 2	200
Oberer Wert Farbbereich 2	800
Farbe für Farbbereich 2	#FF51D211
Unterer Wert Farbbereich 3	800
Oberer Wert Farbbereich 3	1000
Farbe für Farbbereich 3	#FFFF0000
Unterer Wert Farbbereich 4	0
Oberer Wert Farbbereich 4	0
Farbe für Farbbereich 4	

Der **Füllstand** kann Werte im Bereich von 0 bis 1000 Liter annehmen, somit wird die Skala des Tachos auf diese Werte gesetzt. Der Normalbereich befindet sich zwischen 200 bis 800 Litern, dieser soll im Tacho mit einer grünen Farbe versehen werden. Der Bereich darunter (0 bis 200) ist mit einer gelben Farbe zu kennzeichnen. Der Bereich darüber (800 bis 1000) wird mit einer roten Farbe hervorgehoben.

Der Zufluss bzw. Abfluss des Tanks ist von den Bit-Operanden E0.0 „Zulauf“ und E01. „Abfluss“ abhängig. Dabei soll sowohl der Zulauf als auch der Abfluss mit jeweils 20 Einheiten bei Status 1 der Operanden erfolgen.

15.3 Digitale Anzeige

Sind Zahlenwerte im Format **BCD, HEX** oder auch **dezimal** darzustellen, dann ist dies mit Hilfe einer **digitalen Anzeige** möglich. Dabei kann der Inhalt von digitalen Operanden der Formate Byte, UInt16 oder UInt32 angezeigt werden.

Wurde BCD als Anzeige-Format selektiert und der Inhalt des Operanden ist in diesem Format nicht darstellbar, dann wird in der Anzeige für jede nicht darstellbare Ziffer das Zeichen „-“ ausgegeben.

Beispiel (DigitalAnzeige01):

Der Füllstand eines Tanks soll über eine digitale Anzeige visualisiert werden. Der Füllstand liegt dezimal im Operanden EW2 „Füllstand“ vor. Der Normalbereich des Füllstands liegt zwischen 200 bis 800 Liter. Darunter soll die Füllfarbe der Anzeige auf ein dunkles Rot wechseln. Bei Werten darüber (von 800 bis 1000) soll ein zusätzliches Blinken der Anzeige erfolgen. Die Farbe der Segmente der digitalen Anzeige wird von grün auf weiß verändert.

Um das Unterschreiten bzw. Überschreiten des Bereichs zu erkennen, kommen Rechner-Objekte zum Einsatz. Für die Unterschreitung wird im Rechner-Objekt die Funktion „In Bereich“ eingestellt. Am Eingang 1 ist dabei der Wert 0, am Eingang 2 der Füllstand und schließlich am Eingang 3 der Wert 200 eingestellt.

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	In Bereich
Operand Eingang 1 (Any64-lesen)	0
Operand Eingang 2 (Any64-lesen)	Sim.Füllstand
Operand Eingang 3 (Any64-lesen)	200
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.UnterNorm

Bild: Einstellungen des Rechner-Objekts „Unter Normal“

Das Funktionsergebnis wird im Operanden E0.2 „Unter-Norm“ abgespeichert.

Ähnlich wird das Überschreiten des normalen Füllstands mit einem weiteren Rechner-Objekt ermittelt. Dieses Funktionsergebnis ist im Operanden E0.3 „ÜberNorm“ abzulegen.

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	In Bereich
Operand Eingang 1 (Any64-lesen)	800
Operand Eingang 2 (Any64-lesen)	Sim.Füllstand
Operand Eingang 3 (Any64-lesen)	1000
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.ÜberNorm

Bild: Einstellungen des Rechner-Objekts „Über Normal“

Sobald einer der beiden Bit-Operanden den Status 1 besitzt, soll die Hintergrundfarbe **gewechselt** werden. Aus diesem Grund kommt ein drittes Rechner-Objekt zum Einsatz, mit einer Oder-Funktion.

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	ODER
Operand Eingang 1 (Any64-lesen)	Sim.UnterNorm
Operand Eingang 2 (Any64-lesen)	Sim.ÜberNorm
Operand Eingang 3 (Any64-lesen)	0
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.FarbUmschaltung

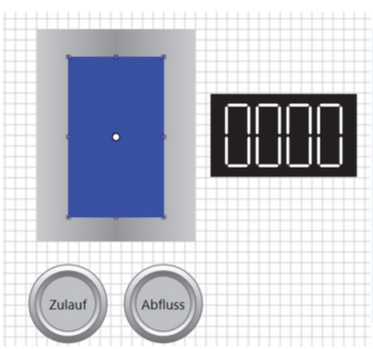
Bild: Einstellungen des Rechner-Objekts „Farbe umschalten“

Das Ergebnis wird dem Operanden E0.4 „FarbUmschaltung“ zugewiesen, den man in der digitalen Anzeige an der Eigenschaft „Operand Umschaltung Füllfarbe“ angibt. Die blinkende Darstellung hat nur zu erfolgen, wenn der Bereich überschritten wurde. Somit kann der Operand „ÜberNorm“ direkt in die Eigenschaft „Operand blinkende Darstellung“ eingetragen werden. Anbei die Einstellungen der digitalen Anzeige (siehe rechts).

Grundeinstellungen	
Beschreibung	Digital-Anzeige
Operand	
Operand (Any64)	Sim.Füllstand
Operand Umschaltung Füllfarbe (bool-les...	Sim.FarbUmschaltung
Operand blinkende Darstellung (bool-lese...	Sim.ÜberNorm

Bild: Operanden-Einstellungen innerhalb der digitalen Anzeige

Zuletzt die Gesamt-Ansicht mit dem Flüssigkeits-Objekt, welches über die Operanden E0.0 „Zulauf“ und E0.1 „Abfluss“ gesteuert wird.



Flüssigkeitsspezifische Einstellungen	
Zulauf 1 (Operand-UInt16-lesen)	0-20 bool Sim.Zulauf
Zulauf 2 (Operand-UInt16-lesen)	0
Zulauf 3 (Operand-UInt16-lesen)	0
Abfluss 1 (Operand-UInt16-lesen)	0-20 bool Sim.Abfluss
Abfluss 2 (Operand-UInt16-lesen)	0
Abfluss 3 (Operand-UInt16-lesen)	0
Anfangsfüllstand in Prozent	10
Zulauf Trigger (Operand-bool-lesen) z.B. I...	0
Max. Füllstand in Inkrementen (100-65535)	1000
Abfluss Trigger (Operand-bool-lesen) z.B....	0
Füllstands-Sensor (Operand-UInt16-schre...	Sim.Füllstand

Bild: Anordnung des Beispiels ohne die Rechner-Objekte

Damit ist die Anordnung komplett und die Simulation kann gestartet werden.

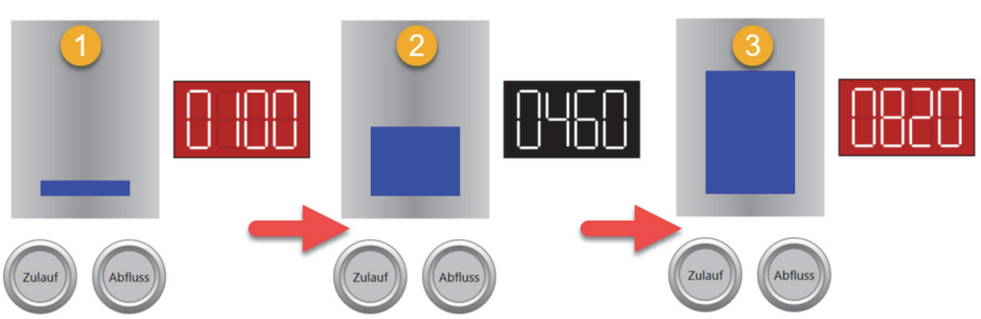


Bild: Simulation des Beispiels

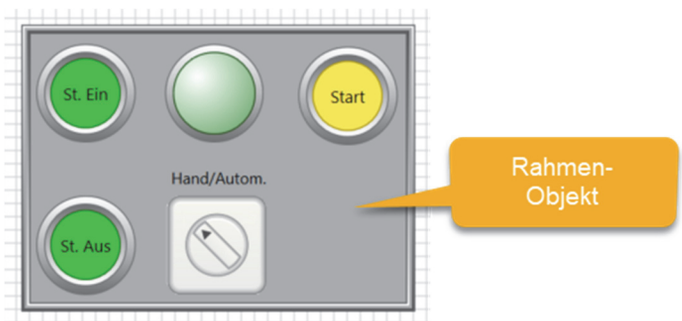
© MHJ-Software GmbH & Co. KG - PLC-Lab

Beim Start hat der Behälter den Füllstand 100, weshalb die digitale Anzeige eine **rote Hintergrundfarbe** besitzt. Wird der Füllstand auf einen Stand > 200 gebracht, dann ist der Normbereich erreicht (Punkt 2 mit Füllstand 460) und die Hintergrundfarbe wechselt auf schwarz. Bei einem Füllstand von über 800 wird wieder auf Rot gewechselt, zusätzlich blinkt die Anzeige.

15.4 Rahmen-Objekte

Rahmen-Objekte sind in rechteckiger oder kreisförmiger Darstellung vorhanden. Sie werden zur optischen Hervorhebung von anderen Objekten verwendet. Als Beispiel sei die Hervorhebung von Bedienelementen einer Anlage genannt.

Bild: Verwendung eines rechteckigen Rahmen-Objekts



Dabei wird der Rahmen meist am Ende platziert und seine **Z-Order** so eingestellt, dass er von allen anderen Objekten verdeckt wird.

Allerdings können Rahmen auch zur Bedienerführung verwendet werden. Durch die Möglichkeit, die Füllfarbe in Abhängigkeit von einem Operanden zu verändern, auch mit blinkender und pulsender Darstellung, können verschiedene „Eskalationsstufen“ visualisiert werden.

Beispiel (Rahmen02):

Ein Fahrkorb kann in vertikaler Richtung bewegt werden. Die Bedienung erfolgt mit einem Taster, welcher sich innerhalb eines kreisförmigen Rahmens befindet. Sobald eine kritische Höhe erreicht ist, ist dies zunächst durch Umschalten der Füllfarbe des Rahmens anzuzeigen. Wird der Korb weiter nach oben bewegt, dann soll durch Blinken die erhöhte Gefahr signalisiert werden. Die höchste „Eskalationsstufe“ ist dann erreicht, wenn die pulsende Darstellung erfolgt. Das Erreichen der einzelnen Höhen wird durch Rechner-Objekte ermittelt und damit werden dann die entsprechenden Bit-Operanden gesetzt.

Anbei zunächst die verwendeten Operanden und Symbole:

	Device	Symbol	Operand	Datentyp	Kommentar
	Sim	**NachOben**	E0.0	bool	Taster nach oben
	Sim	**ZuHochPhase1**	E0.1	bool	Höher als 7 m
	Sim	**ZuHochPhase2**	E0.2	bool	Höher als 10 m
	Sim	**ZuHochPhase3**	E0.3	bool	Höher als 13 m
	Sim	**FahrWeg**	EW2	UInt16	Position in Meter 0 bis 20

Bild: Symbole und Operanden des Beispiels

Nachfolgend die Anordnung für das Beispiel:

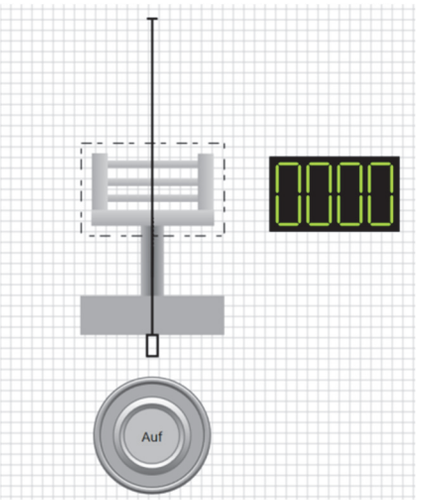


Bild: Anordnung des Beispiels ohne Rechner-Objekte

Der **Fahrkorb** besteht aus dynamischen Rechtecken, welche zu einer Körper-Gruppe zusammengefasst sind. Unterhalb des Fahrkorbs befindet sich ein weiteres dynamisches Rechteck, das als Slave-Objekt über eine Verbindung für lineare Bewegung mit einem statischen Rechteck verbunden ist. Das Slave-Objekt wiederum besitzt eine Verbindung zu einem Körper mit dem „Boden“-Rechteck des Fahrkorbs.

Die Eigenschaften der Verbindung für lineare Bewegung sind wie folgt gesetzt:

Motor Einstellungen	
Motor aktivieren	☑
Geschwindigkeit in m/s (Operand-In...	1
Geschwindigkeits-Divisor	3,0
Max. Kraft (N)	100,0
Operand Bewegung zum Minimum (...	0
Operand Bewegung zum Maximum...	Sim.NachOben
Motor-Kraft ist vom Status der Bewe...	☐
Geschwindigkeit umkehren bei Statu...	☑
Endschalter	
Endschalter am Minimum (Operand-...	0
Endschalter ist Öffner	☐
Endschalter am Maximum (Operand...	0
Endschalter ist Öffner	☐
Sensor Position in Pixel (Operand-UI...	0-20 UInt16 Sim.FahrWeg
Min/Max vertauschen für Positionsw...	☐

Bild: Eigenschaften der Verbindung für lineare Bewegung

Die digitale Anzeige zeigt den Inhalt des Operanden EW2 „FahrWeg“ als dezimalen Wert an. Für die Erkennung der einzelnen Höhen-Phasen kommen Rechner-Objekte mit den Funktionen „Größer als“ zum Einsatz.

Rechner-Objekt Phase 1

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	Größer als
Operand Eingang 1 (Any64-lesen)	Sim.FahrWeg
Operand Eingang 2 (Any64-lesen)	7
Operand Eingang 3 (Any64-lesen)	0
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.ZuHochPhase1

Rechner-Objekt Phase 2

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	Größer als
Operand Eingang 1 (Any64-lesen)	Sim.FahrWeg
Operand Eingang 2 (Any64-lesen)	10
Operand Eingang 3 (Any64-lesen)	0
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.ZuHochPhase2

Rechner-Objekt Phase 3

Calculator Objekt Einstellungen	
Anzuwendende Funktionsart bzw. Verknüpf...	Größer als
Operand Eingang 1 (Any64-lesen)	Sim.FahrWeg
Operand Eingang 2 (Any64-lesen)	13
Operand Eingang 3 (Any64-lesen)	0
Operand Eingang 4 (Any64-lesen)	0
Operand Funktionsergebnis (Any64-schreib...	Sim.ZuHochPhase3

Bild: Eigenschaften der drei Rechner-Objekte

© MHJ-Software GmbH & Co. KG - PLC-Lab

Fehlt nur noch die Belegung des Rahmen-Objekts.

Operand	
Operand Umschaltung der Füllfarbe (bool...	Sim.ZuHochPhase1
Operand blinkende Darstellung (bool-lese...	Sim.ZuHochPhase2
Operand pulsende Darstellung (bool-lese...	Sim.ZuHochPhase3
Farben	
Füllfarbe bei Status 1 des Operanden	#FFEA3110
Füllfarbe	#FFA9A9A9
Linienfarbe	#FFA9A9A9

Bild: Eigenschaften des Rahmen-Objekts

Damit ist die Anordnung komplett und die Simulation kann gestartet werden.

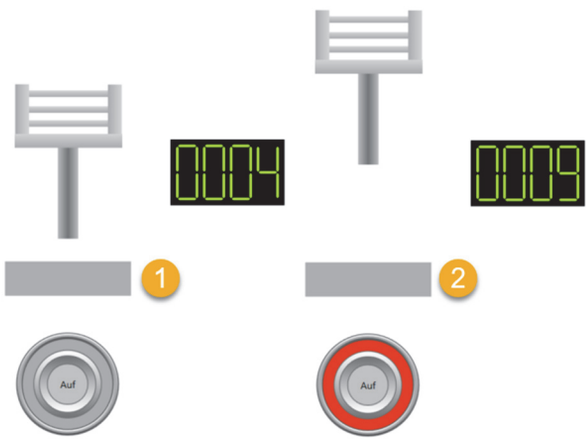


Bild: Simulation der Anordnung

Im obigen Bild ist unter Punkt 1 die Situation dargestellt, wo sich der Korb in einer normalen Höhe befindet. Der Rahmen um den Taster ist noch in der normalen Farbe dargestellt. Rechts daneben (Punkt 2) hat der Korb eine Höhe von 9 m erreicht. Dies wird über die rote Färbung des Rahmens signalisiert. Ab 11 m würde der Rahmen blinkend und ab 14 m pulsend dargestellt werden.

16 Eingabe-Objekte

16.1 Schieberegler

Schieberegler-Objekte können **horizontal** oder **vertikal** angeordnet werden. Sie werden verwendet, um Zahlenwerte in Operanden einzustellen.

Ähnlich wie beim Tacho-Objekt besteht die Möglichkeit, **Skalenbereiche** einzufärben und so z. B. kenntlich zu machen, ob der Wert im Normalbereich verändert wird oder ihn über- bzw. unterschreitet.

Beispiel (Schieberegler01):

Die Geschwindigkeit einer Verbindung für lineare Bewegung ist mit Hilfe eines horizontalen Schiebereglers vorzugeben. Der Wert ist dabei von **-10 bis +10** veränderbar. Über die Polarität des Werts wird auch die Richtung der Bewegung vorgegeben. Die Geschwindigkeit ist in den Operanden EW2 „Geschwindigkeit“ zu schreiben.

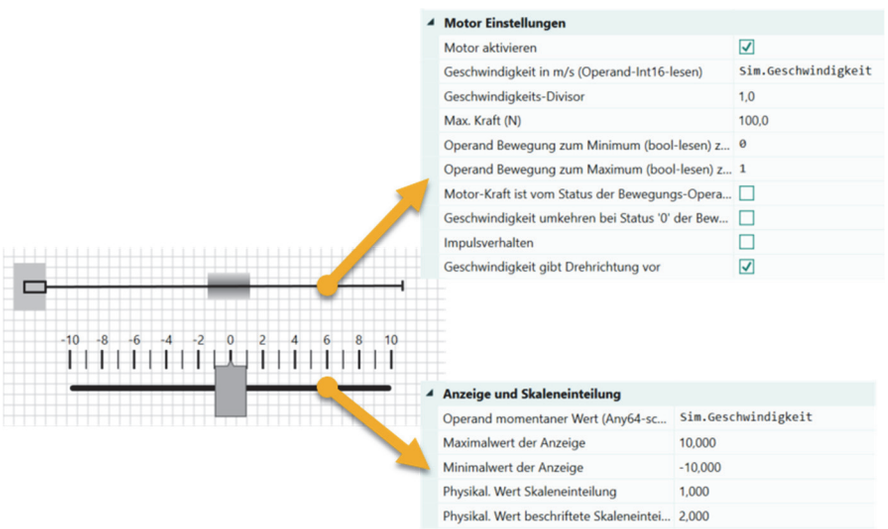


Bild: Anordnung des Beispiels mit den Einstellungen des Schiebereglers und der Verbindung für lineare Bewegung

Beim Start der Simulation bewegt sich das dynamische Rechteck nach rechts, wenn eine Geschwindigkeit > 0 im Schieberegler eingestellt ist. Bei einem negativen Wert erfolgt eine Bewegung nach links.

16.1.1 Startwert vorgeben

Über die Eigenschaft „**Startwert**“ kann der Wert des Schiebereglers vorgegeben werden, der beim **Start der Simulation** am Schieberegler eingestellt sein soll.

16.1.2 Letzten Wert im Run als Startwert vorgeben

Wird diese Option selektiert, dann wird der letzte Wert des Schiebereglers in der Simulation beim Beenden der Simulation als neuer Startwert eingetragen.

16.1.3 Analogwert über Schieberegler vorgeben

Wie man den analogen Wert eines Sensors mit Hilfe eines Schiebereglers simuliert, soll an zwei Beispielen gezeigt werden.

16.1.3.1 Analogwert über Schieberegler vorgeben

Beispiel (Schieberegler02):

In einer realen Anlage liefert ein Sensor das Signal **4 bis 20 mA**. Als SPS kommt dabei eine S7-Steuerung von Siemens zum Einsatz. Diese digitalisiert den analogen Wert und liefert ihn im Bereich **0 bis 27648.**

In PLC-Lab wird ein vertikaler Schieberegler verwendet. Dieser ist auf den **Bereich 4 bis 20** eingestellt. Am Operanden des Schiebereglers wird der Operand EW2 „AnalogerWertInDigits“ eingetragen. Dabei handelt es sich um einen **Wort-**

© MHJ-Software GmbH & Co. KG - PLC-Lab

Operanden mit dem Datentyp **Int16**. Zusätzlich wird ein Skalierungsbereich von **0 bis 27648** angegeben. Die vollständige Angabe an der Operanden-Eigenschaft des Schiebereglers lautet somit:

0-27648 Int16 Sim.AnalogerWertInDigits

Dies hat zur Folge, dass der Bereich 4 bis 20 des Schiebereglers auf den Bereich 0 bis 27648 skaliert und danach in den Wort-Operanden geschrieben wird.

Stellt man also beispielsweise den Wert 16 mA am Schieberegler ein, dann wird in den Operanden der Wert 20736 geschrieben – **der gleiche Wert wie beim realen analogen Eingangskanal mit diesem Messbereich**.

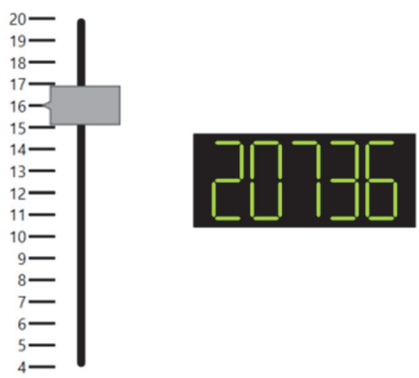


Bild: Simulation eines analogen Kanals

16.1.3.2 Weiteres Beispiel zu Analogwert über Schieberegler vorgeben

Beispiel (Schieberegler03):

Im nächsten Beispiel ist ein **Pt100-850°** Messbereich zu simulieren. Mit diesem können im Nennbereich Temperaturen von -200 °C bis +850 °C erfasst werden. Da in der realen Anlage nur Temperaturen von 10 °C bis 65 °C auftreten können, soll am Schieberegler für die Simulation auch nur dieser Bereich einstellbar sein.

Es kommt eine S7-SPS von Siemens zum Einsatz. Diese liefert bei 10 °C den digitalisierten Wert 100 und bei 65 °C den Wert 650.

In PLC-Lab wird ein vertikaler Schieberegler verwendet. Dieser ist auf den Bereich 10 bis 65 eingestellt. Am Operanden des Schiebereglers wird der Operand EW2 „AnalogerWertInDigits“ eingetragen. Dabei handelt es sich um einen **Wort-Operanden** mit dem Datentyp **Int16**. Zusätzlich wird ein Skalierungsbereich von 100 bis 650 angegeben. Die vollständige Angabe an der Operanden-Eigenschaft des Schiebereglers lautet:

100-650 Int16 Sim.AnalogerWertInDigits

Somit ergibt sich bei der Simulation folgendes Bild:

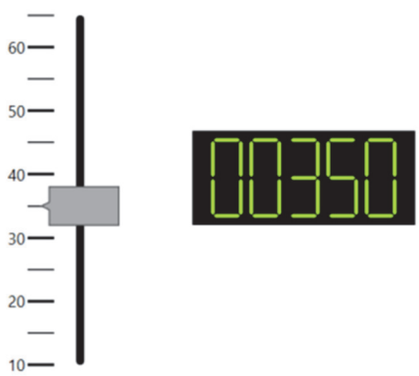


Bild: Simulation des Pt100-850°

Im Bild sind am Schieberegler 35 °C eingestellt, somit wird in den Operanden EW2 der Wert 350 geschrieben, **genau wie in der realen Anlage**.

16.1.3.3 Fazit

Die beiden Beispiele zeigen, dass durch die **Bereichsangabe** am Operanden für den Schieberegler eine **hohe Flexibilität** erreicht wird. So können beispielsweise analoge Messwerte ohne Probleme skaliert werden. Damit entspricht der Wert im Operanden genau dem Wert in der realen Anlage.

In den Beispielen wurde im Schieberegler der analoge Messwert eingestellt, also z. B. 4 bis 20 mA. Genauso ist es aber auch möglich, die reale Messgröße einzustellen, beispielsweise das Gewicht ermittelt durch eine Wäge-Zelle. In diesem Fall muss nur bekannt sein, welchen analogen Wert die Wäge-Zelle bei bestimmten Gewichten an den Eingangskanal liefert.

16.2 Digitaler Ziffernsteller

Mit einem **digitalen Ziffernsteller** kann ein Zahlenwert in den Formaten **BCD, HEX oder dezimal** in einen Byte-, Wort- oder Doppelwort-Operanden geschrieben werden. Der Ziffernsteller kommt beispielsweise zum Einsatz, um **Sollwerte** für Zähler, Zeiten oder Füllstände vorzugeben.

Im Simulationsmodus ist jede Ziffer getrennt mit der Maus einstellbar. Dabei wird auf den oberen bzw. unteren Bereich der Ziffer geklickt und somit der Wert erhöht bzw. verringert. Ist der niedrigste Wert einer Stelle eingestellt und wird er nochmals verringert, dann erfolgt ein Umbruch auf die höchste Wertigkeit (z. B. von 0 nach 9 bei dezimaler Einstellung). Umgekehrt wird bei der höchsten Wertigkeit und einer weiteren Erhöhung der Umbruch zur geringsten Wertigkeit ausgeführt (z. B. von 9 nach 0 bei dezimaler Einstellung).

Beispiel (Ziffernsteller01):

Es soll ein **Ziffernsteller** dazu verwendet werden, die Soll-Position einer vertikalen linearen Bewegung vorzugeben. Dazu wird der Ziffernsteller auf das Format „Dezimal“ eingestellt und mit dem Operand EW4 „PosVorgabe“ verknüpft. Der Wort-Operanden hat den Datentyp UInt16.

Anbei die im Beispiel zu verwendenden Operanden:

	Device	Symbol	Operand	Datentyp	Kommentar
	Sim	**ZylPos**	EW2	UInt16	
	Sim	**PosVorgabe**	EW4	UInt16	
	Sim	**Auf**	E0.0	bool	
	Sim	**Ab**	E0.1	bool	

Bild: Symboltabelle des Beispiels

An der Verbindung für lineare Bewegung werden folgende Einstellungen vorgenommen:

- Die Länge wird auf 325 Pixel eingestellt.
- Die Geschwindigkeit ist auf 1 m/s einzustellen.
- Der Geschwindigkeits-Divisor wird auf 5 gesetzt, damit die Bewegung relativ langsam ist.
- Am Operanden für die Bewegung zum Minimum wird das Symbol „Ab“ angegeben.
- Am Operanden für die Bewegung zum Maximum wird das Symbol „Auf“ angegeben.
- Am Sensor für die Position in Pixel wird das Symbol „ZylPos“ angegeben.

Die Ist-Position soll mit Hilfe einer digitalen Zifferanzeige dargestellt werden, diese ist mit dem Operanden „ZylPos“ zu verknüpfen und auf „Dezimal“ einzustellen.

Im folgenden Bild ist die Anordnung für das Beispiel zu sehen.

© MHJ-Software GmbH & Co. KG - PLC-Lab

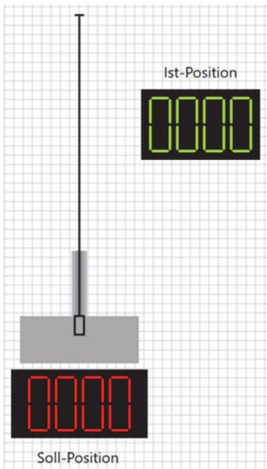
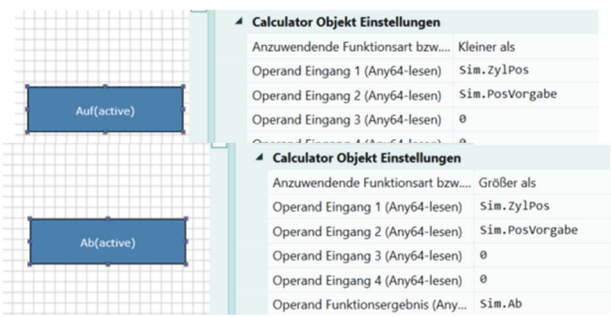


Bild: Anordnung für dieses Beispiel

Im nächsten Schritt soll erreicht werden, dass sich das Slave-Objekt nach oben bewegt, wenn die Soll-Position oberhalb der momentanen Position liegt. Dazu wird ein **Rechner-Objekt** verwendet, welches auf die Funktion „Kleiner als“ einzustellen ist. Am Eingang 1 wird „ZylPos“ angegeben, also die momentane Position des Slave-Objekts. Am Eingang 2 folgt die Angabe „PosVorgabe“. Das Funktionsergebnis wird in den Bit-Operanden mit dem Symbol „Auf“ geschrieben.

Bild: Eigenschaften des Rechner-Objekts für die Aufwärtsbewegung des Slave-Objekts



Ein weiteres Rechner-Objekt ist für die Bewegung nach unten zuständig. Dabei wird als Funktion „Größer als“ selektiert. Eingang 1 ist mit dem Symbol „ZylPos“ belegt und Eingang 2 mit „PosVorgabe“. Das Funktionsergebnis wird dem Bit-Operanden mit dem Symbol „Ab“ übergeben.

Bild: Eigenschaften des Rechner-Objekts für die Abwärtsbewegung des Slave-Objekts

Durch die beiden Rechner-Objekte wird erreicht, dass eine über den Ziffernsteller eingestellte Position angefahren wird. Da die Geschwindigkeit der linearen Bewegung relativ gering ist, muss für die Positionierung keine Regelung zum Einsatz kommen.

Damit ist die Anordnung für das Beispiel komplett und die Simulation kann gestartet werden.

Im Bild rechts (Punkt 1) ist die Situation zu sehen, bei der mit der Maus auf die obere Hälfte der dritten Ziffer des Ziffernstellers geklickt wird. Somit ändert sich der Wert des Operanden von 0 auf 100. **Als Folge fährt das Slave-Objekt auf diese Position und stoppt, sobald sie erreicht ist (Punkt 2).**

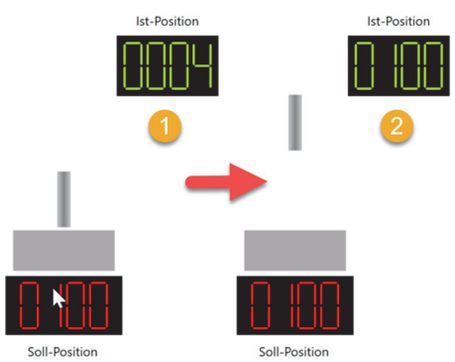


Bild: Simulation der Anordnung

17 Rechner-Objekt

Rechner-Objekte wurden bereits in zahlreichen Beispielen verwendet. Meist werden damit **Auswertungen** oder **Konvertierungen** vorgenommen, welche für die Umsetzung des Anlagenvorgangs notwendig sind.

Ein Rechner-Objekt verfügt über **vier Eingänge** und **einen Ausgang**, an den das **Funktionsergebnis** weitergegeben wird. Je nach eingestellter Funktion können einzelne Eingänge mit den für die Funktion neutralen Konstanten belegt werden, beispielsweise wenn nicht genügend Eingänge für die Verknüpfung vorhanden sind.

Beispiel: Es sollen zwei Eingänge miteinander UND-Verknüpft und das Ergebnis einem Ausgang zugewiesen werden. In diesem Fall sind die Eingänge 2 und 3 mit der Konstanten 1 zu belegen, damit für die UND-Verknüpfung nur die beiden ersten Eingänge **relevant** sind.

Folgende Funktionen sind mit dem Rechner-Objekt möglich:

- UND-, ODER-, Exklusiv-Oder-Verknüpfungen mit Bit-Operanden
- Addition und Multiplikation
- Modulo-Rechnung (Rest einer Division von ganzen Zahlen)
- Dezimal-zu-BCD für Byte- und Wort-Operanden
- BCD-zu-Dezimal für Byte- und Wort-Operanden
- Vergleiche: größer als, größer oder gleich, kleiner als, kleiner oder gleich, gleich
- Bereiche: in Bereich, außerhalb Bereich

17.1 UND-, ODER-, Exklusiv-Oder-Verknüpfungen mit Bit-Operanden

Es können alle vier Eingangs-Operanden belegt werden. Werden weniger Operanden angegeben, dann sind die nicht verwendeten Eingangs-Operanden mit den Konstanten 1 oder 0 zu belegen, je nachdem, welche Konstante für die Operation neutral ist. Am Operanden für das Funktionsergebnis ist ein Bit-Operand anzugeben. Er kann auch über die Eigenschaft „Ergebnis invertieren“ invertiert werden.

17.2 Addition, Multiplikation

Es können alle vier Eingangs-Operanden belegt werden. Werden weniger Operanden angegeben, dann sind die nicht verwendeten Eingangs-Operanden mit den Konstanten zu belegen, je nachdem, welche Konstante für die Operation neutral ist. Am Funktionsergebnis muss ein Operand angegeben sein, der in der Lage ist, den Wert des Ergebnisses aufzunehmen. Anderenfalls findet kein Schreibvorgang in den Operanden statt.

17.3 Modulo

Bei dieser Operation werden nur die beiden ersten Eingangs-Operanden verwendet. Das Funktionsergebnis wird ermittelt aus: Operand 1 [Modulo] Operand 2

Am Funktionsergebnis muss ein Operand angegeben sein, der in der Lage ist, den Wert des Ergebnisses aufzunehmen. Anderenfalls findet kein Schreibvorgang in den Operanden statt.

17.4 Dezimal-zu-BCD für Byte- und Wort-Operanden

Bei dieser Operation ist nur der erste Eingangs-Operand relevant. Es wird nur das niederwertigste Byte bzw. Wort des Operanden ausgewertet. Der ermittelte Wert wird danach in BCD gewandelt und das Ergebnis in den Operanden am Funktionsergebnis geschrieben.

17.5 BCD-zu-Dezimal für Byte- und Wort-Operanden

Bei dieser Operation ist nur der erste Eingangs-Operand zu belegen. Es wird nur das niederwertigste Byte bzw. Wort des Operanden ausgewertet. Der ermittelte Wert wird von BCD nach dezimal gewandelt und das Ergebnis in den Operanden am Funktionsergebnis geschrieben.

© MHJ-Software GmbH & Co. KG - PLC-Lab

17.6 Vergleicher: Größer als, größer oder gleich, kleiner als, kleiner oder gleich, gleich

Bei den Vergleichsoperationen werden nur die Eingangs-Operanden 1 und 2 verwendet. Die Operanden müssen nicht den gleichen Datentyp haben, d. h. es kann beispielsweise der Inhalt eines Byte-Operanden mit dem Inhalt eines Wort-Operanden verglichen werden.

Die Operationen werden wie folgt ausgeführt:

- Größer als: Wert-Operand 1 > Wert-Operand 2
- Größer oder gleich: Wert-Operand 1 >= Wert-Operand 2
- Kleiner als: Wert-Operand 1 < Wert-Operand 2
- Kleiner oder gleich: Wert-Operand 1 <= Wert-Operand 2
- Gleich: Wert-Operand 1 == Wert-Operand 2

Das Funktionsergebnis ist normalerweise in einen Bit-Operanden zu schreiben. Wird ein Byte-, Wort- oder Doppelwort-Operand angegeben, dann werden die Werte 0 oder 1 geschrieben.

17.7 Bereiche: In Bereich, außerhalb Bereich

Bei den Bereichsoperationen werden nur die Eingangs-Operanden 1, 2 und 3 verwendet. Die Operanden können dabei einen unterschiedlichen Datentyp haben, d. h. es kann beispielsweise der Inhalt eines Byte-Operanden mit dem Inhalt eines Wort-Operanden verglichen werden.

Am Eingangs-Operanden 2 ist der Operand anzugeben, dessen Wert zu prüfen ist. Am Eingangs-Operand 1 muss der min. Wert des Bereichs angegeben sein, entweder als Operand oder als Konstante. Am Eingangs-Operand 3 muss der max. Wert des Bereichs angegeben sein, entweder als Operand oder als Konstante.

Die Auswertung sieht folgendermaßen aus:

- In Bereich: Wert-Eingangs-Operand 1 <= Wert-Eingangs-Operand 2 <= Wert-Eingangs-Operand 3
- Außerhalb Bereich: Wert-Eingangs-Operand 1 > Wert-Eingangs-Operand 2 oder
  Wert-Eingangs-Operand 2 > Wert-Eingangs-Operand 3

Das Funktionsergebnis ist normalerweise in einen Bit-Operanden zu schreiben. Wird ein Byte-, Wort- oder Doppelwort-Operand angegeben, dann werden die Werte 0 oder 1 geschrieben.

17.8 Anwendung von Rechner-Objekten

Beispiel (RechnerObjekt01):

Das Beispiel besteht aus einer animierten Pumpe, Rohrleitungen und einem Ventil. Dabei sollen die Rohrleitungen eingefärbt werden, sobald das Medium durch sie hindurchfließt.

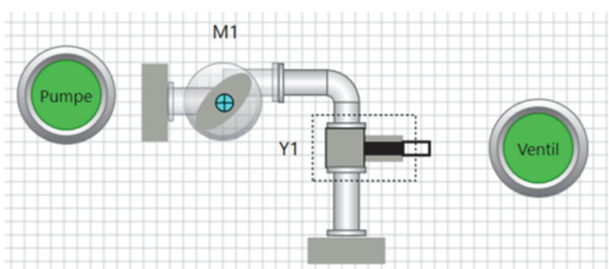


Bild: Anordnung für dieses Beispiel

	Device	Symbol	Operand	Datentyp	Kommentar
	Sim	**Ventil**	E0.0	bool	
	Sim	**PumpeM1**	E0.1	bool	
	Sim	**EndschalterVentilHinten**	E0.2	bool	
	IM	**PumpeAnUndVentilOffen**	M100.0	bool	

Bild: Symboltabelle des Beispiels

In der Anordnung werden die im Bild rechts zu sehende Operanden verwendet.

Die Rohrleitung hinter der Pumpe soll eingefärbt werden, sobald die Pumpe eingeschaltet ist. Somit kann der Operand „PumpeM1“ **direkt** zum Einfärben verwendet werden. Dann folgen das Ventil und eine weitere Rohrleitung.

Die Rohrleitung hinter dem Ventil darf nur farbig dargestellt werden, wenn die Pumpe eingeschaltet und das Ventil offen ist. Für diese **UND-Bedingung** wird ein Rechner-Objekt verwendet.

Zunächst ist am Ventil ein Endschalter für die hintere Endlage notwendig. Das Ventil wird über eine Verbindung für lineare Bewegung angetrieben, somit kann dort an der Eigenschaft „Endschalter am Minimum“ der Operand „EndschalterVentilHinten“ angegeben werden.

Bild: Endschalter-Einstellung an der Verbindung für lineare Bewegung des Ventils

Das Rechner-Objekt:

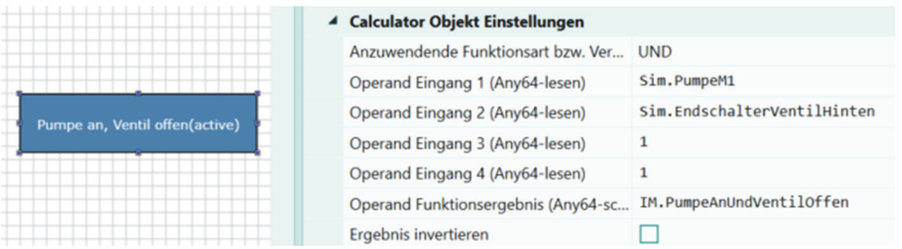


Bild: Die Eigenschaften des Rechner-Objekts

Als Funktionsart ist „UND“ einzustellen. Am Eingang 1 wird der Bit-Operand „PumpeM1“ und am Eingang 2 der Bit-Operand „EndschalterVentilHinten“ eingetragen. Weitere Operanden sind nicht zu verknüpfen, somit kann man die Eingänge 2 und 3 jeweils mit der Konstanten 1 belegen (diese ist bei der UND-Verknüpfung neutral). Das Funktionsergebnis wird in den Operanden „PumpeAnUndVentilOffen“ geschrieben.

Somit ist der Operand „PumpeAnUndVentilOffen“ für das Einfärben der Rohrleitung hinter dem Ventil verwendbar. Die Anordnung ist nun komplett und die Simulation kann gestartet werden.

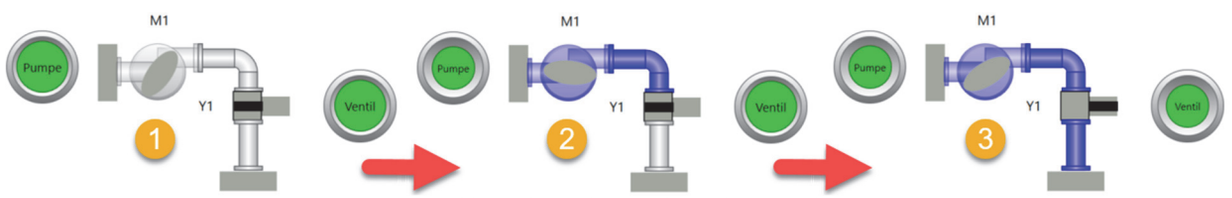


Bild: Simulation der Anordnung des Beispiels

Beim Start der Simulation (Punkt 1) ist die Pumpe noch nicht eingeschaltet, somit sind die Rohrleitungen nicht eingefärbt. Dann wird die Pumpe über den Schalter **aktiviert** (Punkt 2). Dies hat zur Folge, dass die Rohrleitung bis zum Ventil farbig dargestellt wird, denn das Medium fließt bis zum Ventil, welches aber noch geschlossen ist. Bei Punkt 3 wird das Ventil **geöffnet**, somit gelangt das Medium auch in die dahinterliegende Rohrleitung. Diese wird nun ebenfalls farbig dargestellt.

© MHJ-Software GmbH & Co. KG - PLC-Lab

18 Kollisionsgruppen

Kollisionsgruppen wurden bereits im Kapitel für Endschalter und Sensoren erläutert. Dort wurde gezeigt, wie mit Hilfe von Kollisionsgruppen **induktive**, **kapazitive** oder auch **Farbsensoren** realisierbar sind.

Des Weiteren wurde anhand eines Beispiels erläutert, wie ein Endschalter die Kollisionsgruppe eines mit ihm kollidierenden Objekts verändern kann.

In diesem Kapitel folgt ein weiteres Beispiel zu Kollisionsgruppen, wobei auch die Eigenschaft „Zusätzliche Kollisionsgruppen“ benötigt wird.

18.1 Anwendung von Kollisionsgruppen

Beispiel (Kollisionsgruppen01):

Im folgenden Beispiel sollen aus einem **Bunker** Teile mit Hilfe einer **Schaufel** entnommen und über eine Rampe abtransportiert werden.

Im Bunker befinden sich neben den von der Schaufel abzutransportierenden Teilen noch weitere Teile, welche aber von der Schaufel nicht beeinflusst werden sollen. Allerdings sollen jene Teile mit den zu transportierenden Teilen **interagieren**, d. h. diese sollen **kollidieren**.

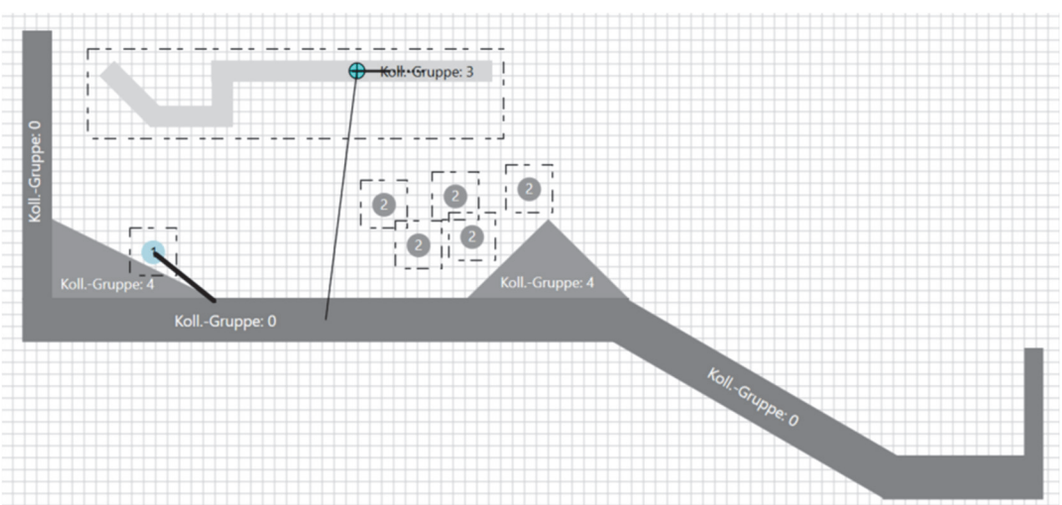


Bild: Anordnung des Beispiels

Bei der Anordnung stellt sich das Problem, dass die **rotierende Schaufel** Teile des Bunkers berührt und so eigentlich an der Rotation gehindert wird. Dies lässt sich nur dadurch verhindern, dass man die einzelnen Objekte **verschiedenen Kollisionsgruppen** zuordnet, womit sie sich nicht ins „Gehege“ kommen.

18.1.1 Zuordnung der Objekte zu den Kollisionsgruppen

Im ersten Schritt werden die Objekte der Anordnung den einzelnen Kollisionsgruppen zugeordnet.

- Die **blauen** Kreis-Objekte sollen **abtransportiert** werden.
  Diese werden der Kollisionsgruppe 1 zugeordnet.
- Die **grauen** Kreis-Objekte sind die Teile, welche im Bunker **verbleiben**,
  und erhalten die Kollisionsgruppe 2.
- Die rotierende Schaufel erhält die Kollisionsgruppe 3.
- Die Dreiecke, welche an den Seiten des Bunkers angeordnet sind, um die Teile in die Mitte des Bunkers zu leiten, erhalten die Kollisionsgruppe 4
- Alle anderen Seitenbegrenzungen verbleiben in der Kollisionsgruppe 0, kollidieren also mit allen Gruppen.

18.1.2 Eintragen der zusätzlichen Kollisionsgruppen

Die Objekte einer **Kollisionsgruppe** kollidieren zunächst einmal nur mit anderen Objekten der gleichen Kollisionsgruppe und mit den Objekten der Kollisionsgruppe 0.

Sollen Objekte auch mit Objekten anderer Kollisionsgruppen kollidieren, dann sind diese Gruppen in der Eigenschaft „Zusätzliche Kollisionsgruppen“ anzugeben.

Im Beispiel bedeutet dies:

- **Blaue** Kreise: Diese sollen mit den anderen Kreisen im Bunker kollidieren, also mit der Kollisionsgruppe 2. Zusätzlich müssen sie mit der Schaufel kollidieren, also der Gruppe 3. Zuletzt kollidieren sie auch mit den Dreiecken am Rand des Bunkers, somit der Gruppe 4. Die Angabe lautet also „2,3,4“.
- **Graue** Kreise: Diese kollidieren mit den blauen Kreisen, also der Gruppe 1, zusätzlich mit den Dreiecken am Rand des Bunkers (Gruppe 4). Somit lautet die Angabe „1,4“.
- **Schaufel**: Diese kollidiert mit den blauen Kreisen der Gruppe 1.
- **Dreiecke** am Rand des Bunkers: Die Seitenbegrenzungen müssen mit den beiden Kreisobjekten kollidieren, somit lautet die Angabe „1,2“.

Damit sind alle notwendigen Kollisionsgruppen angegeben.

Stellt sich noch die Frage: Warum kollidiert die rotierende Schaufel nicht mit dem Boden des Bunkers?

Dies liegt daran, dass der Boden das Master-Objekt der Verbindung für Drehbewegung darstellt, bei der eingestellt ist, dass die **Anker-Objekte nicht kollidieren**. Deshalb verhalten sich die beiden Anker-Objekte neutral zueinander.

18.1.3 Simulation der Anordnung

Beim Start der Simulation zeigt sich folgendes Bild:

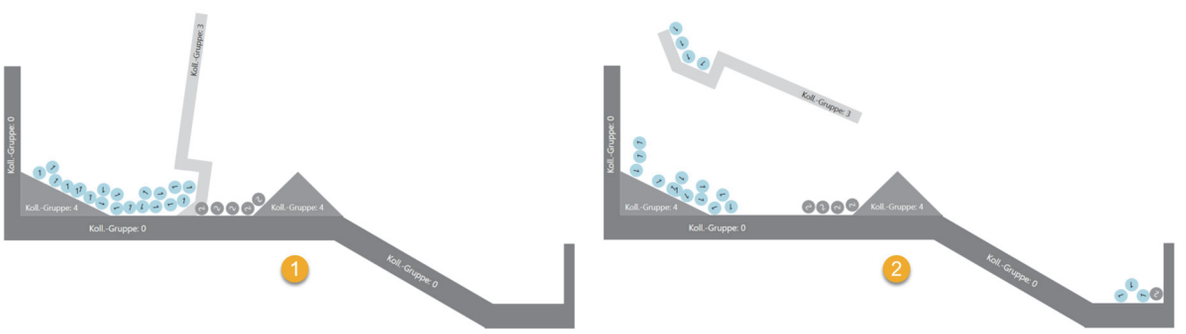


Bild: Simulation der Anordnung

Auf der linken Seite (Punkt 1) taucht die Schaufel in den Bunker ein. Man erkennt, dass die grauen Kreis-Objekte **nicht** von der Schaufel beeinflusst werden. Anders die blauen Kreis-Objekte: Sie werden von der Schaufel erfasst und nach rechts abtransportiert.

18.1.4 Fazit des Beispiels

Das Beispiel zeigt, wie durch die geschickte Zuordnung von Objekten zu einzelnen Kollisionsgruppen eine Art 3D-Anordnung erzeugt werden kann. Objekte, welche in der Realität hintereinander angeordnet sind, können so auch in 2D angeordnet werden.

© MHJ-Software GmbH & Co. KG - PLC-Lab

18.2 Induktive, kapazitive oder Farbsensoren

Bei der Erläuterung der Endschalter und Sensoren wurde bereits gezeigt, wie man in PLC-Lab Sensoren realisiert, welche auf unterschiedliche **Materialien**, **Farben** oder **Oberflächen** reagieren. Der Schlüssel für die Realisierung war dabei die Verwendung von **Kollisionsgruppen**. Hier folgt nun ein weiteres Beispiel zu diesem Thema, welches die Umsetzung einer solchen Anwendung zeigt.

18.2.1 Beispiel einer Sortier-Anlage

Beispiel (Kollisionsgruppen02):

In diesem Beispiel sind Würfel aus unterschiedlichen Materialien zu sortieren. Die Würfel bewegen sich dabei eine Rampe hinab. Das **Material** der Würfel soll im weiteren Verlauf von **Sensoren** erfasst werden. Je nachdem, welches Material erkannt wird, sind die Würfel dann entsprechend auf die Nebenbänder abzuleiten. Dazu werden Zylinder mit Hilfe von Sensoren ausgefahren. Die Zylinder leiten dann die Würfel auf verschiedene Nebenbänder.

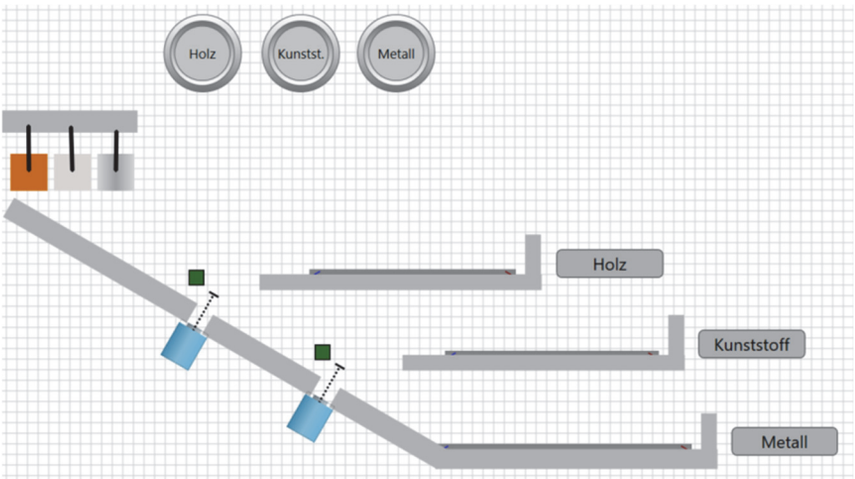


Bild: Anordnung des Beispiels

Nachfolgend die im Beispiel verwendeten Operanden.

	Device	Symbol	Operand	Datentyp	Kommentar
	Sim	**Holz**	I0.0	bool	
	Sim	**CreateHolz**	I0.1	bool	
	Sim	**Kunststoff**	I0.2	bool	
	Sim	**CreateKunststoff**	I0.3	bool	
	Sim	**CreateMetall**	I0.4	bool	

Bild: Operanden des Beispiels

Für die Umsetzung der Aufgabenstellung wird der braune Würfel, welcher die Würfel aus Holz repräsentiert, der Kollisionsgruppe 1 zugeordnet. Der hellgraue Würfel besteht aus Kunststoff, er erhält die Kollisionsgruppe 2. Und schließlich ist der Würfel aus Metall der Kollisionsgruppe 3 zugehörig.

Die Würfel aus Holz sollen auf der ersten Ebene aussortiert werden. Dazu ist an dieser Stelle ein Endschalter zu platzieren, welcher der Kollisionsgruppe 1 zugeordnet ist. Dieser löst somit nur bei den Holzwürfeln aus. Der Endschalter initiiert das Ausfahren des Zylinders, welcher dann den Holzwürfel auf die erste Ebene ablenkt.

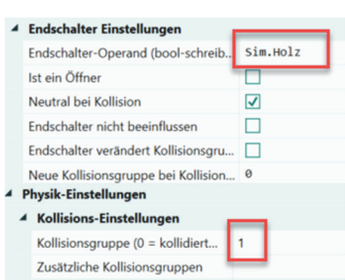


Bild: Einstellungen des Sensors für Holz

In ähnlicher Weise erfolgt das Aussortieren der Würfel aus Kunststoff. Hierbei wird ein Endschalter platziert, welcher die Kollisionsgruppe 2 besitzt und somit nur durch den Kunststoffwürfel ausgelöst wird. Damit wird dann ebenfalls der Zylinder ausgefahren und der Würfel auf die zweite Ebene abgelenkt.

Einzig der Würfel aus Metall löst keinen der beiden Endschalter aus, denn er gehört ja der Kollisionsgruppe 3 an. Er wird somit von den Endschaltern nicht „bemerkt“.

Damit ist die Anordnung fertiggestellt und die Simulation kann beginnen.

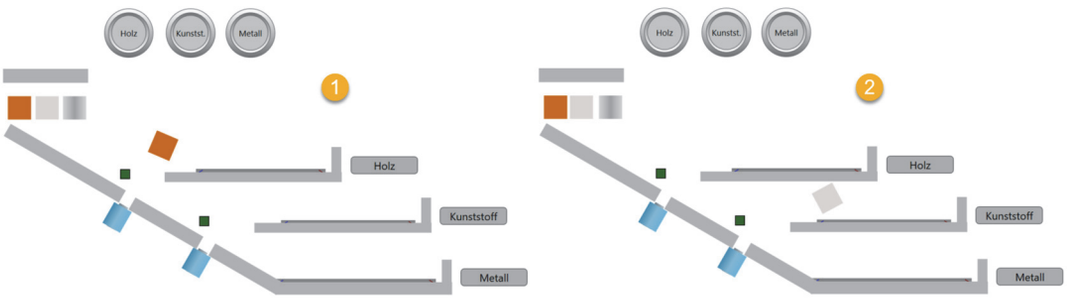


Bild: Simulation der Anordnung

Im obigen Bild ist links (Punkt 1) zu sehen, wie gerade ein Holzwürfel vom Sensor erfasst und somit vom Zylinder auf das Nebenband abtransportiert wird. Rechts daneben (Punkt 2) ist die Situation mit einem Kunststoffwürfel abgebildet. Auch hier hat der Sensor den Zylinder ausgelöst, der wiederum den Würfel nach rechts abgelenkt hat.

Fazit: Das Beispiel hat gezeigt, wie mit Hilfe von Kollisionsgruppen verschiedenartige Sensoren realisiert werden können.

18.3 Kollisionsgruppe über Endschalter beeinflussen

Mit Hilfe eines Endschalters oder auch mit anderen Physik-Objekten (welche ja bekanntlich ebenfalls die Endschalter-Eigenschaften besitzen) ist die Kollisionsgruppe eines Objekts veränderbar. Die neue Kollisionsgruppe kann mit einer Konstanten oder auch einem Operanden vorgegeben werden.

Beispiel (Kollisionsgruppen03):

Im Beispiel kann man **blaue** Rechtecke erzeugen. Diese gehören der Kollisionsgruppe 1 an. Die Objekte werden über eine schiefe Ebene nach unten befördert und treffen dann auf einen Endschalter.

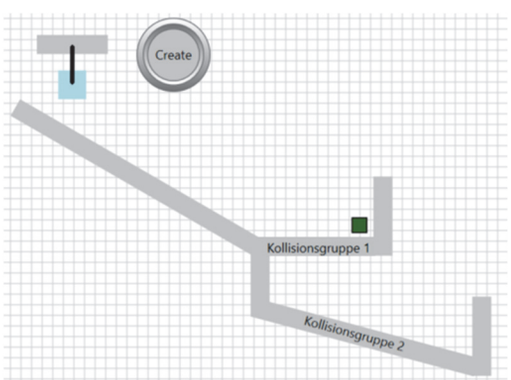


Bild: Anordnung für dieses Beispiel

Bild: Einstellung des Endschalters

Am Endschalter ist die Eigenschaft „Endschalter verändert Kollisionsgruppe“ eingeschaltet. Des Weiteren wurde als neue Kollisionsgruppe die Konstante 2 angegeben.

Gleitet nun ein Objekt die schiefe Ebene herab und berührt den Endschalter, dann wird seine Kollisionsgruppe auf 2 abgeändert.

© MHJ-Software GmbH & Co. KG - PLC-Lab

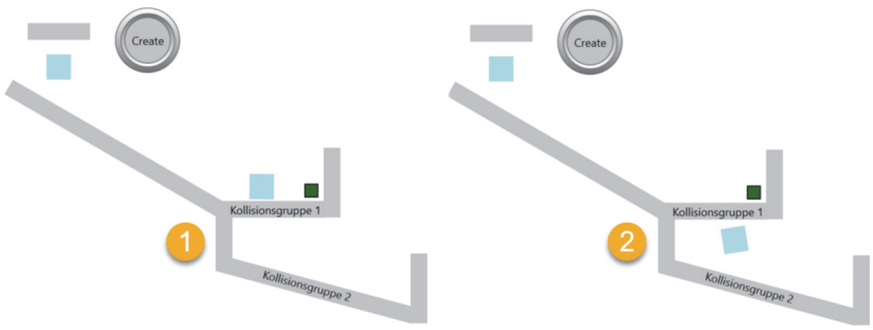


Bild: Simulation

Damit wird der Untergrund des Objekts, welcher der Kollisionsgruppe 1 angehört, durchlässig (Bild Punkt 2). Das Objekt fällt somit herab und wird von der schrägen Ebene mit der Kollisionsgruppe 2 aufgehalten.

18.4 Neutral bei Kollision

Innerhalb der Endschalter Einstellungen ist unter anderem die Eigenschaft „**Neutral bei Kollision**“ vorhanden. Diese ist bei Endschalter-Objekten automatisch vorselektiert. Die Eigenschaft bewirkt, dass ein Objekt zwar den Endschalter berührt und ihn auch auslöst, das Objekt aber in seiner Bewegung nicht beeinflusst wird. Der Endschalter ist also **rückwirkungsfrei**, was die Bewegung des ihn auslösenden Objekts anbelangt. Der Endschalter detektiert aber die Berührung und reagiert seiner Einstellung entsprechend.

Anders verhält sich das Ganze, wenn der Endschalter nicht in der gleichen Kollisionsgruppe wie das kollidierende Objekt ist (und selbst nicht der Kollisionsgruppe 0 angehört). Dann „bemerkt“ der Endschalter die Kollision nicht und löst somit auch nicht aus.

Ist die Eigenschaft „Neutral bei Kollision“ nicht selektiert, dann wird ein Objekt vom Endschalter aufgehalten und kann ihn beispielsweise nicht überfahren.

Beispiel (Kollisionsgruppen04):

Im Beispiel ist dies veranschaulicht:

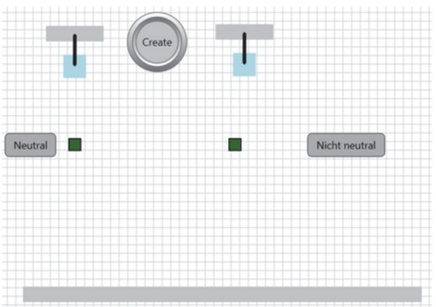


Bild: Anordnung für dieses Beispiel

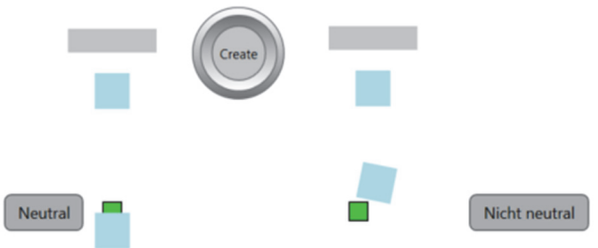


Bild: Simulation des Beispiels

Der linke Endschalter ist neutral, der rechte nicht. Dies hat zur Folge, dass der rechte Endschalter die erzeugten Objekte in ihrer Flugbahn beeinflusst, während sie durch den linken Endschalter hindurchfallen. Ausgelöst werden aber beide Endschalter. Im Bild „Simulation des Beispiels“ ist dies gut zu erkennen.

Dabei ist die Situation zu sehen, bei der sich das blaue Rechteck auf der linken Seite durch den Endschalter hindurchbewegt. Das blaue Rechteck auf der rechten Seite kollidiert mit dem Endschalter und wird von dessen Konturen abgelenkt.

19 Farbverläufe und Transparenz

19.1 Farbverläufe in Objekten verwenden

In den vorausgegangenen Beispielen wurden schon mehrfach **Farbverläufe** verwendet, um Objekten ein plastisches Aussehen zu geben.

Möchte man beispielsweise das Aussehen eines Rechtecks so verändern, dass es wie eine **Metallstange** wirkt, dann geht man nach der Selektion des Objekts wie folgt vor:

1. Bei der Eigenschaft „Füllfarbe“ den Dialog für die Farbe aufrufen. In der einfarbigen Einstellung das Grau für die Seitenbereiche auswählen.
2. Nun den passenden Gradientenverlauf selektieren. Als Folge wird ein zweiter Farbverlaufstopp für die rechte Seite einfügen. Dabei wird automatisch das zuvor selektierte Grau eingestellt.
3. Einen weiteren Farbverlaufstopp einfügen und in der Mitte platzieren. Für diesen Farbverlaufstopp eine etwas dunklere Farbe auswählen.

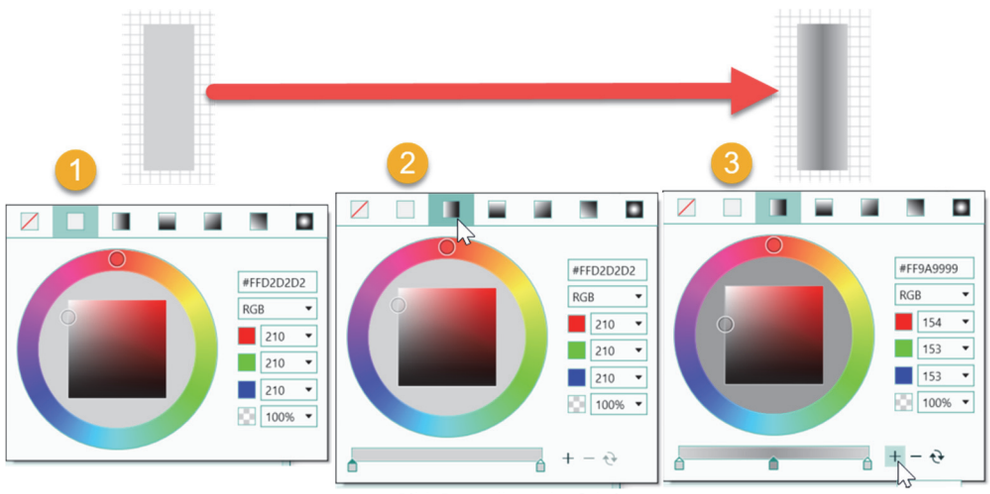


Bild: Darstellung der Schritte

19.2 Transparenz

Möchte man **glasähnliche** Effekte erzielen, so ist dies durch **Transparenz** möglich.

Beispiel (Transparenz01):

Im Beispiel wird eine Metallbearbeitungsmaschine mit einer Schutztür versehen. Diese besteht aus einer Sicherheits-Glasscheibe und einem Rahmen. Damit die Glasscheibe auch in der Simulation zu erkennen ist, wird der Deckungsgrad der Füllfarbe über den Farb-Dialog auf 50% eingestellt. Dadurch sind die Objekte hinter der Schutztür weiterhin zu erkennen, so wie dies auch in der Realität der Fall wäre.

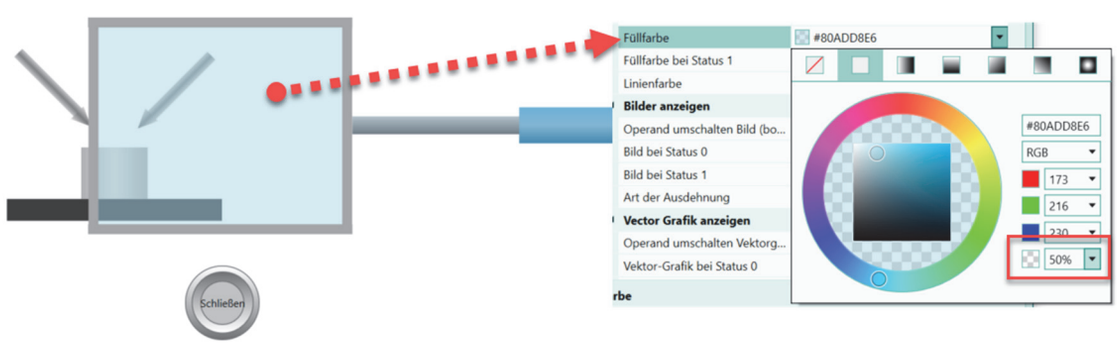


Bild: Einstellung des Deckungsgrads einer Farbe

© MHJ-Software GmbH & Co. KG - PLC-Lab

20 Verwendung von Grafiken in den Objekten

In Schaltern und geometrischen Objekten können verschiedene Grafiken angezeigt werden. Sie können im Format **JPEG, PNG, BMP usw.** vorliegen. Des Weiteren sind Vektor-Grafiken im **XAML-Format** verwendbar. Deren Vorteil besteht darin, dass sie **ohne Qualitätsverlust** vergrößert oder verkleinert werden können.

Möchte man ein Bild innerhalb eines Objekts anzeigen, so selektiert man die Bild-Datei innerhalb der Eigenschaft „Bild bei Status 0“ bzw. „Bild bei Status 1“. Soll kein Bildwechsel in Abhängigkeit von einem Operanden erfolgen, dann genügt die Auswahl einer Grafik für die Eigenschaft „Bild bei Status 0“.

20.1 Beispiel zur Verwendung von Grafiken

Beispiel (Bilder01):

Im folgenden Beispiel soll der **Bohrer** einer Bohrmaschine mit Hilfe einer Grafik dargestellt werden. Zusätzlich soll die eingeschaltete Bohrmaschine erkennbar sein, was über einen Wechsel der beiden Bilder erreicht wird.

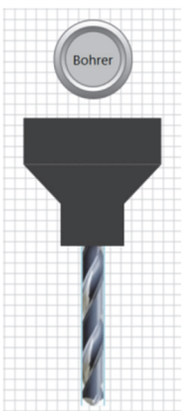


Bild: Anordnung des Beispiels

Im ersten Schritt ist ein feststehendes Rechteck zu platzieren und in der Rubrik „Darstellung->Bilder anzeigen“ an der Eigenschaft „Bild bei Status 0“ der Dialog zur Auswahl der Datei aufzurufen. **Zu beachten ist, dass dies nur möglich ist, wenn das Anlagenprojekt bereits abgespeichert wurde, da die in der Anordnung verwendeten Bilder in das Verzeichnis des Projekts kopiert werden.** Das zu verwendende Bild befindet sich in den Dokumenten des aktuellen Benutzers im Unterverzeichnis „PLC-Lab-Editor\Items“ und hat den Namen „Bohrer2_01.jpg“.

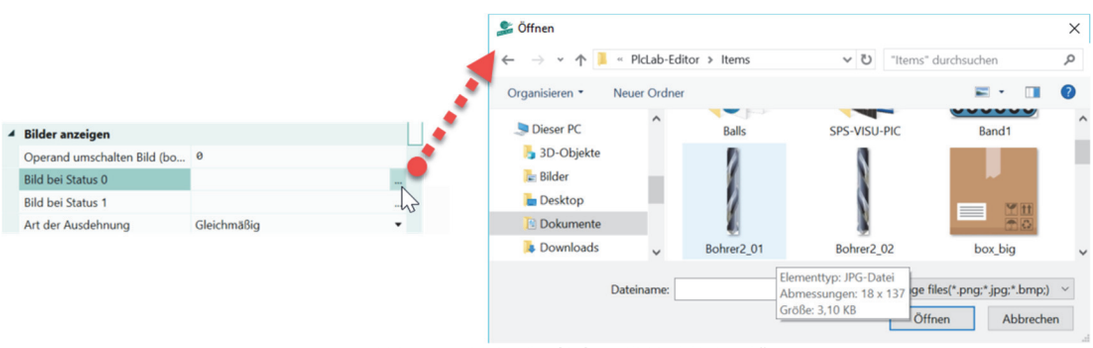


Bild: Auswahl der Grafik für „Bild bei Status 0“

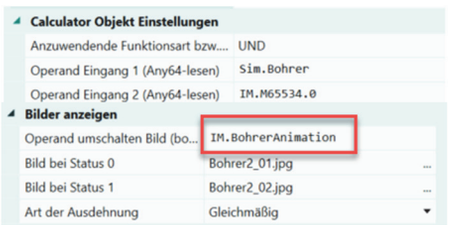
Anschließend wird das Bild für die Eigenschaft „Bild bei Status 1“ ausgewählt, das sich auch im obigen Verzeichnis befindet und die Bezeichnung „Bohrer2_02.jpg“ hat. Nach den Aktionen haben die Eigenschaften des feststehenden Rechtecks folgendes Aussehen:

Bild: Einstellung der beiden Eigenschaften zum Anzeigen der Grafiken

Wenn die Bohrmaschine **eingeschaltet** ist, dann soll ein **Wechsel** zwischen den beiden Bildern stattfinden. Damit wird der Eindruck eines sich drehenden Bohrers erzeugt. Für diesen Wechsel ist ein Operand notwendig, dessen Status stetig von 0 nach 1 wechselt, sobald der Bohrer eingeschaltet ist.

Bei der Vorstellung des IM-Device wurde erwähnt, dass das Merker-Byte mit der Adresse 65534 als Taktmerkerbyte ausgelegt ist. Somit wechseln die Bits des Bytes mit einer bestimmten Frequenz ihren Status. Im Beispiel ist das Merker-Bit mit der Adresse 65534.0 für die Animation zu verwenden. Damit die Animation nur bei eingeschaltetem Bohrer läuft, wird mit Hilfe eines **Rechner-Objekts** eine UND-Verknüpfung zwischen dem Taktmerker-Bit und dem Operanden E0.0 „Bohrer“ hergestellt. Die anderen beiden Eingänge des Rechner-Objekts sind auf die Konstanten 1 zu setzen.

Bild: Einstellungen des Rechner-Objekts



Das Funktionsergebnis wird dem M0.0 „BohrerAnimation“ aus dem IM-Device übertragen. Dieses ist dann für den Wechsel der Bilder in der Eigenschaft „Operand umschalten Bild“ anzugeben.

Bild: Verwendung des toggelnden Operanden für den Wechsel der Grafiken

Nach dem Start der Simulation und dem Einschalten des Bohrers mit Hilfe des Schalters wird innerhalb des Rechtecks immer wieder eine der beiden Grafiken abwechselnd angezeigt. Durch den Wechsel entsteht der **Eindruck** eines sich **drehenden Bohrers**.

© MHJ-Software GmbH & Co. KG - PLC-Lab

20.2 Weiteres Beispiel zur Verwendung von Grafiken

Beispiel (Bilder02):

Im nächsten Beispiel sollen **Vektorgrafiken** verwendet werden. Diese besitzen die Endung **XAML** und haben den Vorteil, dass sie beliebig vergrößert oder verkleinert werden können, ohne dass Qualitätsverluste bei der Darstellung auftreten.

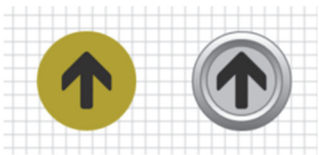


Bild: Anordnung des Beispiels

In der Anordnung wurde ein Dekoration-Kreis-Objekt platziert und darin ein Pfeil nach oben dargestellt. Dieser Pfeil ist eine Vektorgrafik, und somit wird diese Grafik mit der Endung XAML an der Eigenschaft „Vektor-Grafik bei Status 0“ des Objekts ausgewählt. Sobald der Operand E0.0 „PfeilNachOben“ den Status 1 besitzt, soll die Füllfarbe des Objekts auf eine gelbe Farbe wechseln. Somit wird der Operand in die Eigenschaft „Operand umschalten Füllfarbe“ eingetragen und die gelbe Farbe an der Eigenschaft „Füllfarbe bei Status 1“ selektiert.

Bild: Einstellungen innerhalb des Dekorations-Kreis-Objekts

Neben dem Kreis-Objekt wird ein grauer Schalter ohne Leuchteinsatz angeordnet. Bei diesem wird der Operand E0.0 „PfeilNachOben“ an der Eigenschaft „Operand“ angegeben, damit die Betätigung des Schalters dessen Status verändert. Innerhalb der Rubrik „Vektor Grafik anzeigen“ ist an der Eigenschaft „Vektor-Grafik bei Status 0“ ebenfalls der Pfeil nach oben zu selektieren. Damit wird die Grafik auch innerhalb des Schalters angezeigt.

Bild: Simulation des Beispiels

Startet man die Simulation und wird der Schalter betätigt, dann wechselt das Kreis-Objekt seine Füllfarbe (siehe Bild oben).

20.3 Fazit

Die Beispiele haben gezeigt, wie einfach es ist, Grafiken in geometrischen Objekten und Schaltern anzuzeigen. Haben die Grafiken transparente Bereiche, dann können diese durch den Wechsel der Füllfarbe hervorgehoben werden. Alternativ wechselt man die Grafik in Abhängigkeit vom Status eines Bit-Operanden.

Vektorgrafiken bzw. Symbole können z. B. hier geladen werden: http://materialdesignicons.com/

21 Anlagen vor Veränderungen schützen

Die Anordnungen in PLC-Lab können mit Hilfe eines **Passworts** vor Veränderungen geschützt werden.

Wird ein Anlagenprojekt mit Passwortschutz geöffnet, dann ist es zunächst vor Änderungen geschützt (Read-only-Modus). Die Eigenschaften von Objekten sind nicht veränderbar. Auch das Verändern der Position von Objekten auf der Zeichenfläche ist nicht möglich. Des Weiteren können keine Objekte der Zeichenfläche hinzugefügt werden.

Ein solcher Passwortschutz wird über das folgende Symbol innerhalb der Maus-Icons kenntlich gemacht:

Bild: Symbol für gesperrte bzw. mit Passwort geschützte Anlage in PLC-Lab

21.1 Ein Anlagenprojekt mit einem Passwortschutz versehen

Möchte man eine Anlage mit einem Passwortschutz versehen, dann geht man wie folgt vor:

1. Öffnen des Anlagenprojekts.
2. Ausführen des Menüpunkts „Mehr…->Passwort setzen“.
3. Auf dem erscheinenden Dialog „Passwort setzen“ gibt man das gewünschte Passwort (max. 10 Zeichen) und dessen Wiederholung an.
4. Nach Bestätigung der Eingabe über „OK“, erscheint die Meldung, dass das Setzen des Passwortes erfolgreich war.
5. Nun speichert man das Anlagenprojekt über [STRG] + [S].
6. Wird das Anlagenprojekt geschlossen und neu geöffnet, dann ist der Schreibschutz aktiv, d. h. das Projekt kann nicht verändert werden.

21.2 Ein Anlagenprojekt mit Passwortschutz entsperren

Wurde ein Anlagenprojekt mit Passwortschutz geöffnet, dann ist der Schreibschutz aktiv und die Anlage kann nicht verändert werden. Ist man im Besitz des Passwortes, dann kann die Anlage entsperrt und somit der Schreibschutz **aufgehoben** werden. Dazu geht man wie folgt vor:

1. Innerhalb der Maus-Icons befindet sich ein Icon mit einem Schloss-Symbol, welches mit dem Text „Projekt entsperren“ beschrieben ist. Ein Klick auf dieses Symbol lässt den Dialog „Passwortabfrage“ erscheinen. Ebenso kann der Menüpunkt „Mehr…->Projekt mit Passwort entsperren“ selektiert werden.
2. Wurde über eine der beiden Aktionen der Dialog „Passwortabfrage“ zur Ansicht gebracht, dann kann auf ihm das gültige Passwort eingetragen und der Dialog bestätigt werden.
3. War die Passwortabfrage erfolgreich, dann wird dies über eine Meldung angezeigt und das Anlagenprojekt kann editiert werden.

21.3 Das Passwort eines Anlagenprojekts ändern

Nachfolgend ist die Vorgehensweise beschrieben, wenn man ein vorhandenes Passwort ändern möchte.

1. Öffnen des Projekts.
2. Betätigung des Menüpunkts „Mehr…->Passwort setzen“.
3. Es erscheint der Dialog „Passwortabfrage“, auf dem das momentan gültige Passwort anzugeben ist.
4. Nach gültiger Passwortabfrage erscheint der Dialog „Passwort setzen“. Auf ihm kann nun das neue Passwort und dessen Wiederholung eingegeben werden. Bei Bestätigung des Dialogs wird das neue Passwort gesetzt und ist nach dem Speichern des Projekts gültig.

© MHJ-Software GmbH & Co. KG - PLC-Lab

21.4 Das Passwort eines Anlagenprojekts dauerhaft entfernen bzw. rücksetzen

Möchte man bei einem Anlagenprojekt mit Passwortschutz das Passwort dauerhaft entfernen, dann geht man wie folgt vor:

1. Öffnen des Projekts.
2. Betätigung des Menüpunkts „Mehr…->Passwort setzen“.
3. Es erscheint der Dialog „Passwortabfrage“, auf dem das gültige Passwort anzugeben ist.
4. Nach gültiger Passwortabfrage erscheint der Dialog „Passwort setzen“. Auf ihm werden beide Eingabefelder leer belassen und der Dialog über „OK“ bestätigt. Daraufhin erscheint eine Abfrage, ob das Passwort rückzusetzen ist. Bei Bestätigung mit „Ja“ wird das Rücksetzen ausgeführt. Nach dem Speichern des Projekts ist kein Passwort mehr vorhanden.

Notizen:

© MHJ-Software GmbH & Co. KG - PLC-Lab