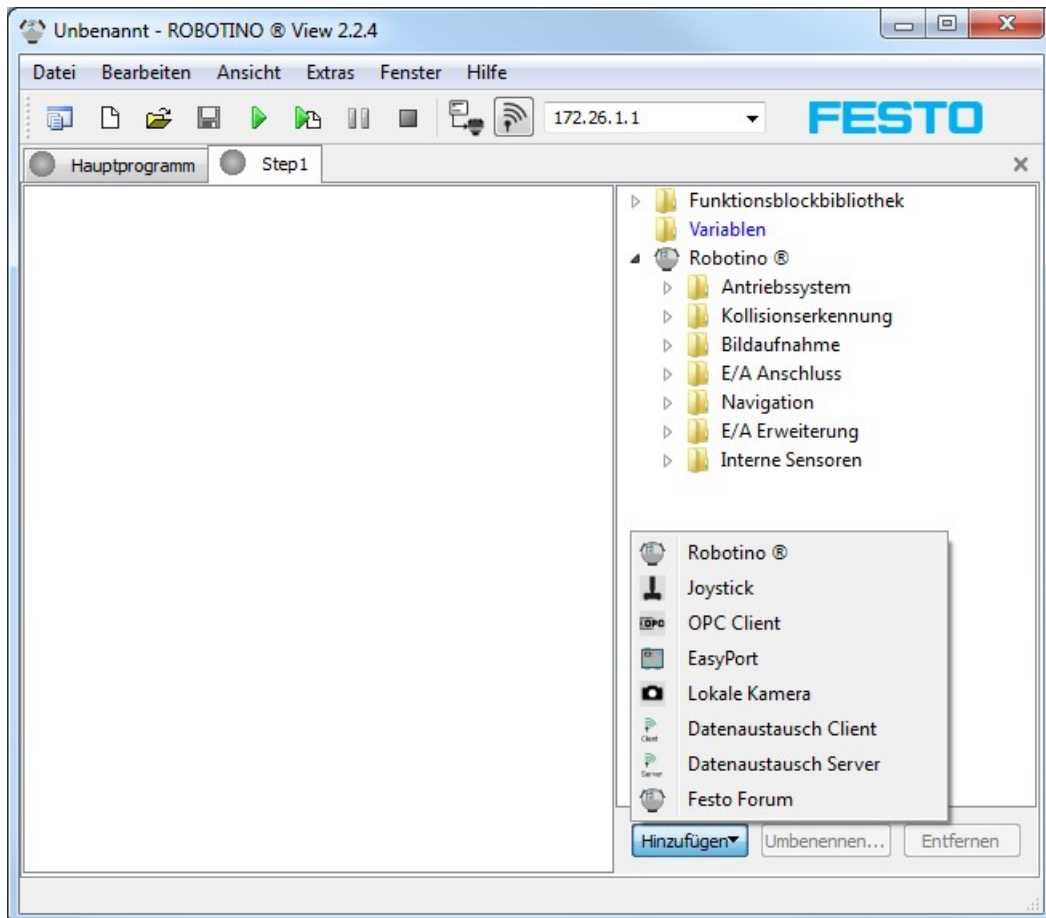


Geräte

Geräte stellen die Verbindung zwischen Robotino View und der Umwelt her. Das Gerät "Robotino" kann mit einem echten oder simulierten Robotino kommunizieren. Das Gerät "Joystick" wiederum kann die Stellungen der Achsen eines an den Rechner angeschlossenen Joysticks auslesen.

Anlegen und bearbeiten

In einem neuen Projekt ist immer das Gerät "Robotino" vorhanden. Um weitere Geräte anzulegen, wechselt man in ein Unterprogramm.



Unterhalb der Funktionsblockbibliothek können neue Geräte über den Knopf "Hinzufügen" eingefügt werden. Das gewählte Gerät erscheint dann unterhalb von Robotino in der Funktionsblockbibliothek.

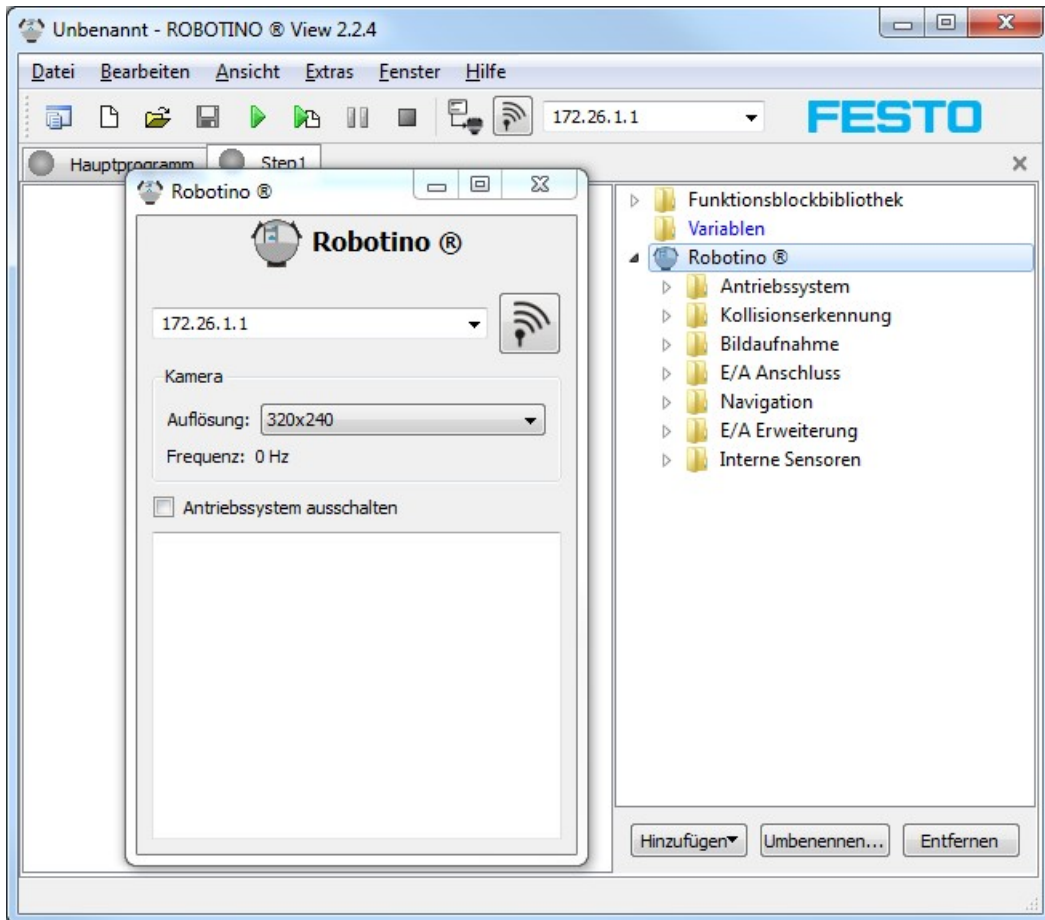
Neuen Geräten wird ein eindeutiger Name zugewiesen. Dieser Name kann über den Knopf "Umbenennen" geändert werden, wenn zuvor ein Gerät in der Funktionsblockbibliothek markiert wurde.

Mit dem "Entfernen" Knopf können Geräte aus der Funktionsblockbibliothek gelöscht werden. Diese Funktion ist nur dann verfügbar, wenn keine Funktionsblöcke des Gerätes innerhalb des gesamten Projektes verwendet werden.

Dialog aufrufen

Zu jedem Gerät gibt es einen Dialog, über welchen weitere Einstellungen vorgenommen werden können.

Die Dialoge werden durch einen Doppelklick auf das Gerät in der Funktionsblockbibliothek angezeigt.



FESTO

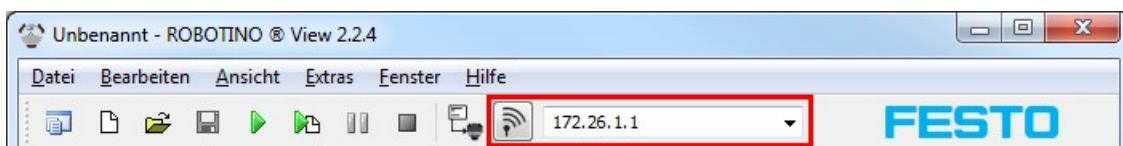
Robotino



Das Gerät "Robotino" ermöglicht den Zugriff auf die Sensoren und Aktoren eines Robotino® Robotersystems.

FESTO

Werkzeugleiste

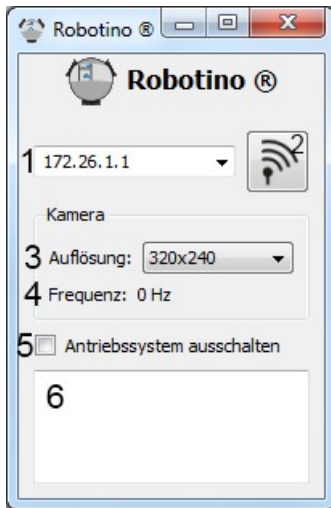


In der Werkzeugleiste von Robotino® View wird die IP-Adresse und der Verbindungsknopf des in der Geräteverwaltung an oberster Stelle geführten Robotino® Geräts angezeigt (rot umrahmter Bereich). Die Funktionalität der IP-Adressen-Eingabe und des Verbindungsknopfes sind identisch zu der im Dialog der Geräteverwaltung.

FESTO

Hauptdialog





Der Hauptdialog des Robotino Gerätes wird nach einem Doppelklick auf das Robotino Gerät in der Funktionsblockbibliothek angezeigt.

1	IP-Adressen Eingabe	Bei Auslieferung hat Robotino die IP-Adresse 172.26.1.1. Eine lokal laufende Simulation von Robotino hat die IP-Adresse 127.0.0.1:8080, wobei 8080 die Portnummer ist. Bei mehreren Robotinos in der Simulation kann die Portnummer auch höher sein.
2	Verbindungsknopf	Durch drücken dieses Schalters wird eine Verbindung zu Robotino auf- bzw. abgebaut.
3	Auflösung	Die Auflösung der von Robotino aufgenommenen Bilder.
4	Frequenz	Die Frequenz, mit welcher Bilder empfangen werden.
5	Antriebssystem ausschalten	Durch setzen dieses Schalters können die Motoren von Robotino deaktiviert werden.
6	Nachrichtenfenster	Textfeld für Nachrichten.

FESTO



Manipulatordialog



Über den Manipulatordialog wird ein optional an Robotino angeschlossener Manipulator verwaltet.

1	Strang 1-3 aktiviert: Schaltet die Stromversorgung der drei Servoanschlüsse. Im Falle eines Fehlers (Blinken der Servo LED) kann der Fehler durch Abschalten der Stromversorgung gelöscht werden. Drehmoment schalten: Schaltet den Antrieb aller Servos an bzw. ab. Bei eingeschaltetem Antrieb leuchtet der rote Taster an Robotinos Arm. Die Antriebe können auch über den roten Taster an Robotinos Arm geschaltet werden.
2	Hier werden die gespeicherten Positionen aufgelistet. Eine Position kann durch Betätigen des grünen Tasters an Robotinos Arm

gespeichert werden. Alternativ kann auch das über die rechte Maustaste erreichbare Kontextmenü genutzt werden.

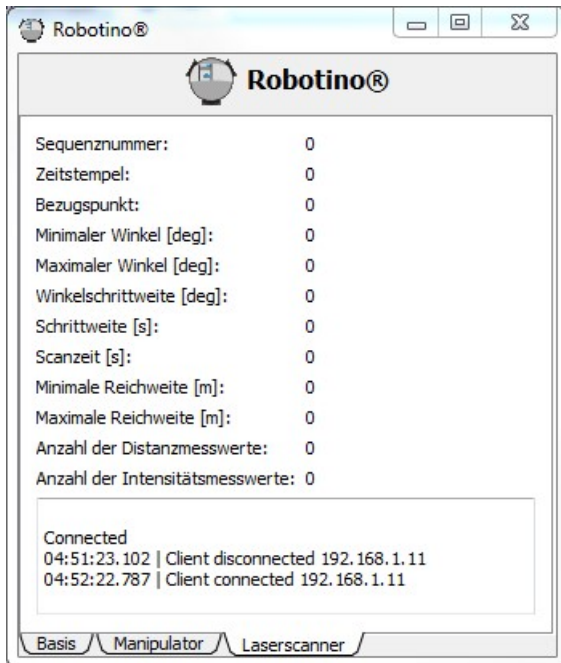
- 3 Das Nachrichtenfenster. Um die Nachrichten zu sehen, die direkt nach dem Verbinden mit Robotino empfangen werden, muss die Verbindung bei geöffnetem Gerätedialog hergestellt werden.

Siehe auch das [Beispiel](#) zur Verwendung des elektrischen Greifarms.

FESTO



Laserscannerdialog

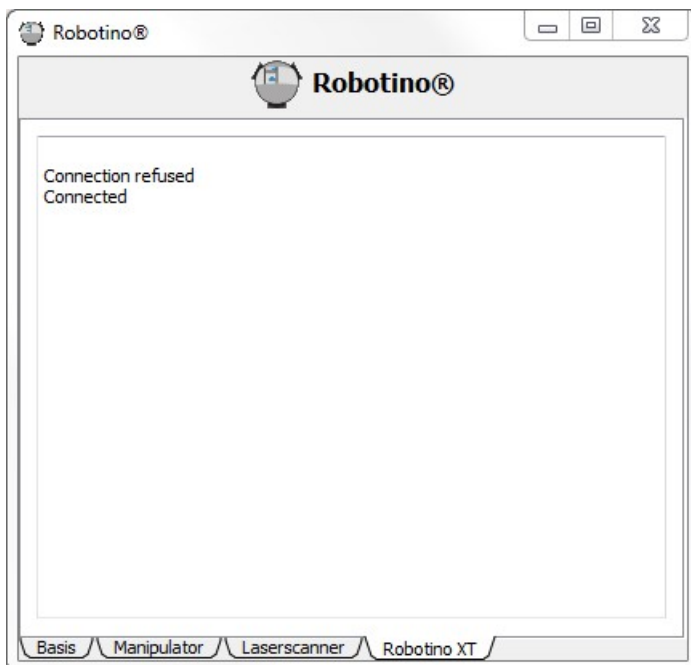


Der Laserscannerdialog zeigt Statusinformationen zu einem optional angeschlossenen Laserscanner.

FESTO



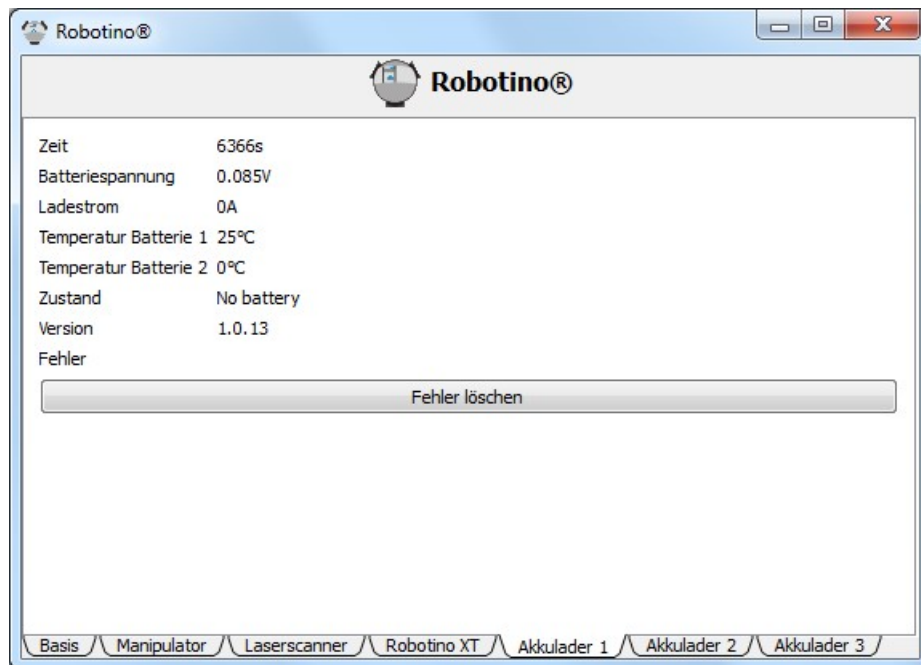
Robotino XT



Der Dialog zeigt Statusinformationen zu Robotino XT.

FESTO

Akkulader



Der Dialog zeigt den Status von Akkulader 1 in Robotino. Bei der optionalen Bestückung von Robotino mit NiMH-Akkus kommen mehrere Akkulader zum Einsatz. Deren Status kann in den Fenstern Akkulader 2 und 3 eingesehen werden.

FESTO

Funktionsblöcke



Die Funktionsblöcke ermöglichen die Verwendung des Gerätes Robotino in einem Unterprogramm.

FESTO

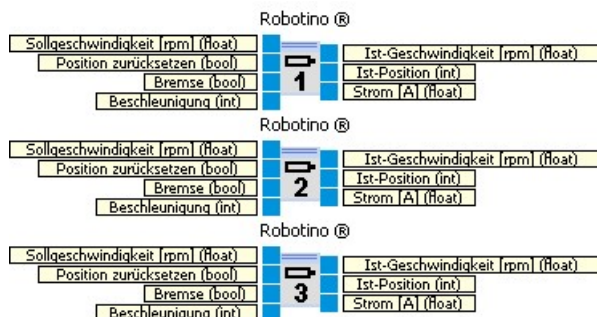
Antriebssystem



Dieser Ordner enthält Funktionsblöcke zur Antriebssteuerung von Robotino.

FESTO

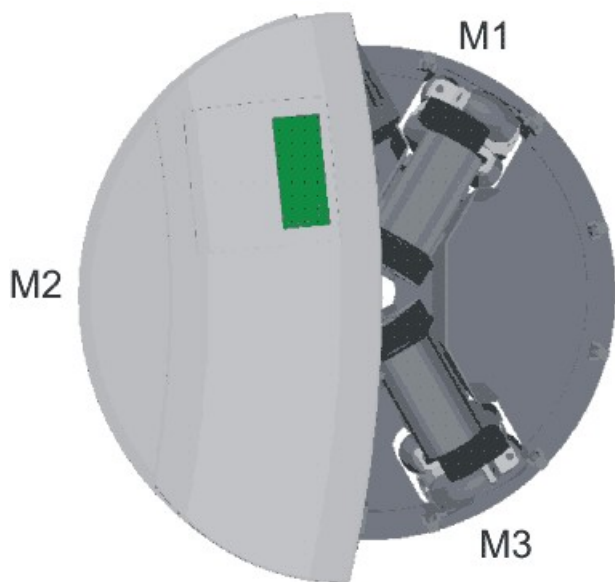
Motor



Dieser Funktionsblock repräsentiert einen der drei Antriebsmotoren von Robotino. Die Nummer des Motors ist aus der Grafik des Funktionsblocks ersichtlich.

Eingänge	Typ	Einheit	Standard	Beschreibung
Sollgeschwindigkeit	float	rpm	0	Die Sollgeschwindigkeit des Motors in Umdrehungen pro Minute (rounds per minute, rpm). Zu beachten ist, dass sich zwischen Motor und Rad ein 16:1 Getriebe

				befindet.
Position zurücksetzen	bool		false	Wenn wahr (true) wird der Zähler der Impulse des Drehgebers des Motors auf 0 gesetzt.
Bremse	bool		false	Wenn wahr (true), wird der Motor gestoppt.
Beschleunigung	int		100	Kopplung zwischen Sollgeschwindigkeit am Eingang und tatsächlich übertragener Sollgeschwindigkeit (siehe auch ► Dialog).
Ausgänge				
Geschwindigkeit	float	rpm		Die Istgeschwindigkeit des Motors.
Position	int			Die seit Einschalten von Robotino oder seit einem vorangegangenen Übergang am "Position zurücksetzen" Eingang von wahr (true) nach unwahr (false) gezählten Impulse des auf der Motorachse angebrachten Drehgebers. Der Drehgeber erzeugt 2000 Impuls pro Motorumdrehung. Kann zur Ist-Positionierung des angetriebenen Rads benutzt werden.
Strom	float	A		Der vom Motor aufgenommene Strom in A.



FESTO



Dialog

Motor #1

Motor

Beschleunigung:

kp:

ki:

kd:

☒ Standardparameter verwenden

☐ Bei Start zurücksetzen

Parameter	Beschreibung

Beschleunigung	Beschleunigungs- bzw. Verzögerungsfaktor. Mit dem Maximalwert 100 wird eine Sollgeschwindigkeit direkt an den Motor weitergegeben. Bei kleineren Werten wird eine Rampe gefahren. Dadurch kann eine weichere Bewegung des Roboters erreicht werden.
kp	Proportionalanteil des dem Motor vorgeschalteten PID Reglers
ki	Integralanteil des dem Motor vorgeschalteten PID Reglers
kd	Differentialanteil des dem Motor vorgeschalteten PID Reglers
Standardparameter verwenden	Die in der Firmware von Robotinos EA Karte hinterlegten Werte für kp, ki und kd verwenden. Die Standardparameter werden ebenfalls verwendet, wenn kp=ki=kd=255 gesetzt wird.
Bei Start zurücksetzen	Initialisierung des Positionswertes mit 0 bei jedem Programmstart

Die Geschwindigkeitsregelung jedes Motors wird von einem PID Regler übernommen.

$$u(t) = K_p \left(e(t) + \frac{1}{T_N} \int_0^t e(t') dt' \right) + K_d \dot{e}(t)$$

Die Parameter sind:

 K_p
 $K_i = 1/T_n$
 K_d

Die im Dialog einstellbaren Werte werden auf die vom Regler verwendeten Parameter umgerechnet:

 $K_p = kp / 2$
 $K_i = ki / 1024$
 $K_d = kd / 2$

Die Standardwerte sind

 $kp = 25$
 $ki = 25$
 $kd = 25$

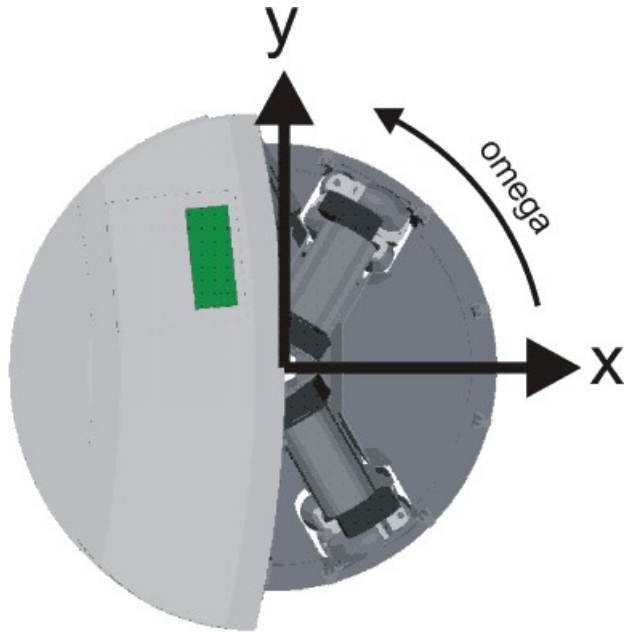

Antriebssystem



Dieser Funktionsblock ermöglicht den direkten Zugriff auf das Antriebssystem von Robotino.

Eingänge	Typ	Einheit	Standard	Beschreibung
vx	float	mm/s	0	Sollgeschwindigkeit in x-Richtung.
vy	float	mm/s	0	Sollgeschwindigkeit in y-Richtung.
omega	float	deg/s	0	Solldrehgeschwindigkeit

Ausgänge				
vx	float	mm/s		Istgeschwindigkeit in x-Richtung.
vy	float	mm/s		Istgeschwindigkeit in y-Richtung.
omega	float	deg/s		Istdrehgeschwindigkeit



Die Abbildung zeigt Robotinos lokales Koordinatensystem. Ein positiver Wert für die Drehgeschwindigkeit omega erzeugt von oben betrachtet eine Rotation gegen den Uhrzeigersinn.

FESTO



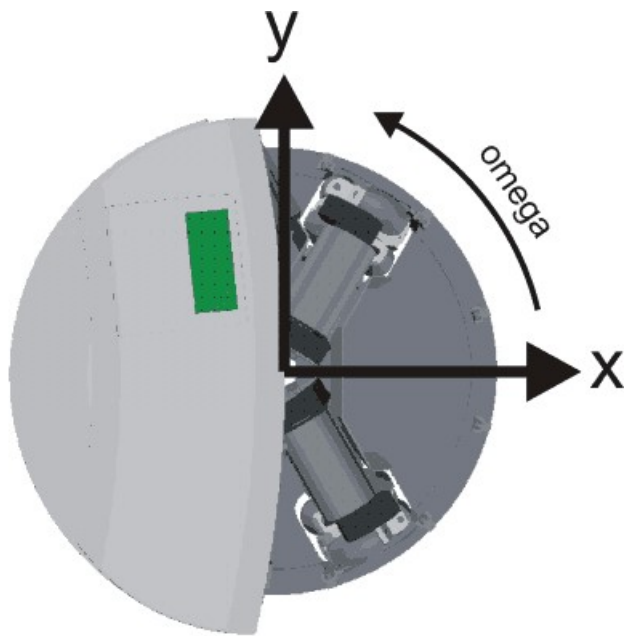
Omniantrieb

Robotino ®					
vx [mm/s] (float)				m1 [rpm] (float)	
vy [mm/s] (float)				m2 [rpm] (float)	
omega [deg/s] (float)				m3 [rpm] (float)	

Berechnet die Sollgeschwindigkeiten der Motoren 1,2 und 3 aufgrund der Vorgabe einer Sollgeschwindigkeit in x und y-Richtung und einer Söldrehgeschwindigkeit.

Eingänge	Typ	Einheit	Standard	Beschreibung
vx	float	mm/s	0	Sollgeschwindigkeit in x-Richtung.
vy	float	mm/s	0	Sollgeschwindigkeit in y-Richtung.
omega	float	deg/s	0	Söldrehgeschwindigkeit
Ausgänge				
m1	float	rpm		Sollgeschwindigkeit Motor 1
m2	float	rpm		Sollgeschwindigkeit Motor 2
m3	float	rpm		Sollgeschwindigkeit Motor 3

Der Funktionsblock "Omniantrieb (invers)" berechnet vx, vy und omega aus den Drehgeschwindigkeiten der Motoren.



Die Abbildung zeigt Robotinos lokales Koordinatensystem. Ein positiver Wert für die Drehgeschwindigkeit ω erzeugt von oben betrachtet eine Rotation gegen den Uhrzeigersinn.

FESTO

Dialog



Omniantrieb (1)

Robotino®

Omniantrieb

Basisradius: 0,187m

Radradius: 0,062m

Übersetzungsverhältnis: 32,00

FESTO

Omniantrieb Invers



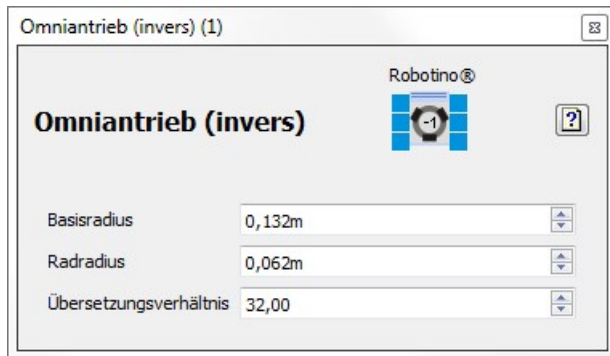
Robotino®

m1 (rpm) (float)		vx (mm/s) (float)
m2 (rpm) (float)		vy (mm/s) (float)
m3 (rpm) (float)		omega (deg/s) (float)

Berechnet die Istgeschwindigkeit und Ist Drehgeschwindigkeit aufgrund der Vorgabe von einzelnen Motorgeschwindigkeiten.

Eingänge			
m1	float	rpm	Sollgeschwindigkeit Motor 1
m2	float	rpm	Sollgeschwindigkeit Motor 2
m3	float	rpm	Sollgeschwindigkeit Motor 3
Ausgänge			
vx	float	mm/s 0	Geschwindigkeit in x-Richtung.
vy	float	mm/s 0	Geschwindigkeit in x-Richtung.
omega	float	deg/s 0	Drehgeschwindigkeit

Dialog



Kollisionserkennung

Dieser Ordner enthält Funktionsblöcke mit denen Objekte detektiert werden können.

Stoßleiste

Robotino®



In der Stoßleiste ist ein Tastsensor integriert. Bei Berührung liefert der Sensor ein Ausgangssignal.

Eingänge	Typ	Standard	Beschreibung
Ausgänge			
Wert	bool		Wahr (true), wenn ein Objekt gegen die Stoßleiste drückt. Ansonsten unwahr (false)

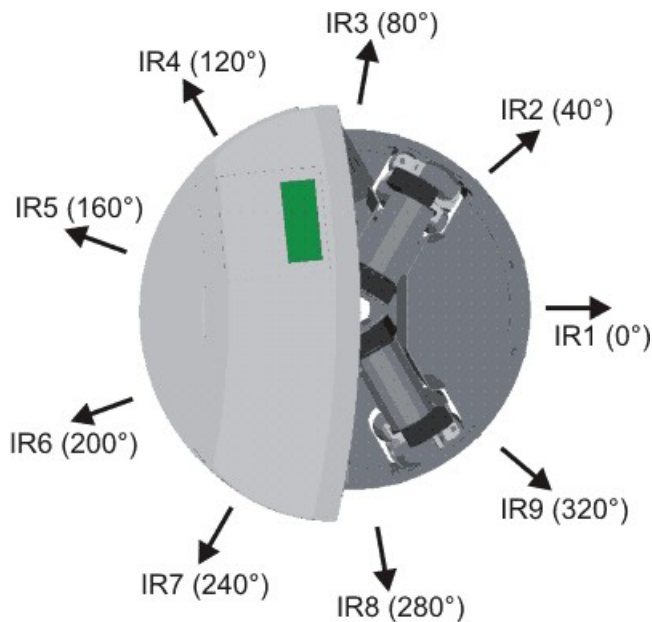
Abstand



Liefert den Wert eines Abstandssensors.

Eingänge	Typ	Einheit	Standard	Beschreibung
Ausgänge				
Wert	float	Volt		Gibt die analoge vom Sensor gelieferte Spannung in V an. Die Skalierung und Umrechnung in einen Abstandswert mit der Dimension einer Länge muss durch den Anwender erfolgen.

Orientierung	float	Grad		Die Orientierung des Sensors in Robotinos lokalem Koordinatensystem. Der Wert ist berechnet sich aus der Nummer des Abstandssensors zu Orientierung = $40^\circ \times (\text{Nummer} - 1)$
--------------	-------	------	--	--

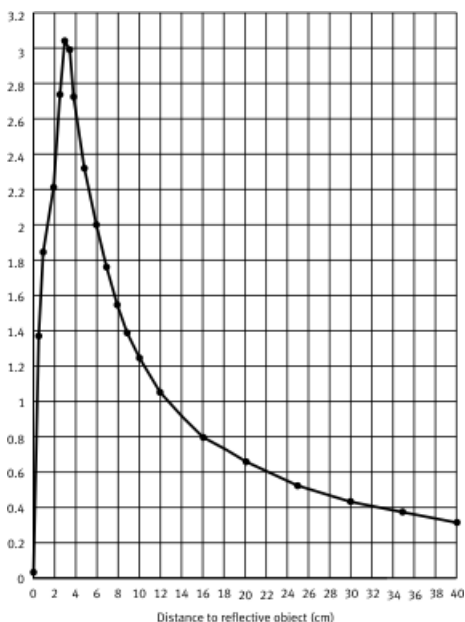


FESTO



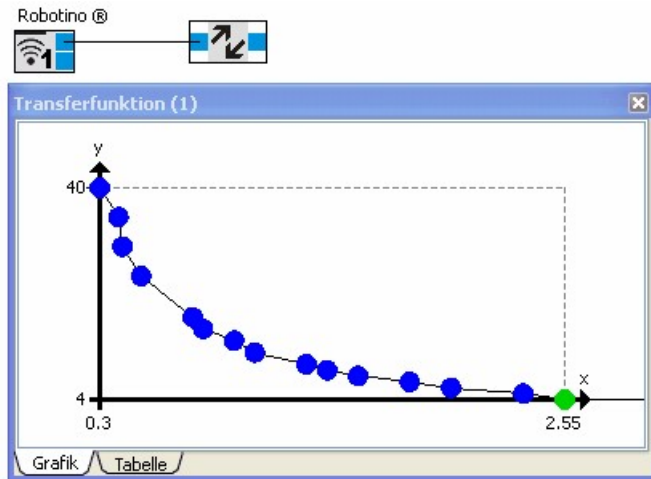
Beispiel

Dem Datenblatt zum Abstandssensor (ein Sharp GP2D120) entnimmt man folgende Kurve für die Beziehung zwischen dem Abstand zum Objekt in cm und dem analogen Ausgangssignal in Volt.



Diese Kurve kann man leicht mit einer [Transferfunktion](#) nachbilden und erhält damit die gewünschte Umrechnung der analogen Spannung in einen Abstand zum Objekt in cm. Dabei ist zu beachten, dass die Transferfunktion die Umkehrfunktion darstellt und analoge Spannung auf der x-Achse gegen Abstand in cm auf der y-Achse aufrägt. Damit die Funktion eindeutig ist, werden nur Distanzwerte größer 4cm betrachtet. Für kleinere Abstände ist anhand der analogen Spannung nicht feststellbar, ob das Objekt sehr nah ist (Abstand <4cm) oder weiter entfernt.

Ferner ist zu beachten, dass der AD-Wandler, an welchen der Sensor angeschlossen ist nur Spannungen bis 2,55V misst. Da sich die Beziehung zwischen analoger Spannung und Distanz auch mit dem Material des reflektierenden Objekts ändert, misst man am besten selbst nach und verwendet die selbst ermittelten Werte für die Umrechnung.



Die Werte der [Transferfunktion](#) sind unten angegeben. Sie können sie kopieren und in Ihre eigene [Transferfunktion](#) einfügen.

0.3	40
0.39	35
0.41	30
0.5	25
0.75	18
0.8	16
0.95	14
1.05	12
1.3	10
1.4	9
1.55	8
1.8	7
2	6
2.35	5
2.55	4

FESTO



Bildaufnahme

Dieser Ordner enthält Funktionsblöcke zur Verwendung von Robotinos Kamera.

FESTO



Kamera

Robotino ®



Liefert ein Live-Bild der auf Robotino montierten Kamera.

Eingänge	Typ	Standard	Beschreibung
Ausgänge			

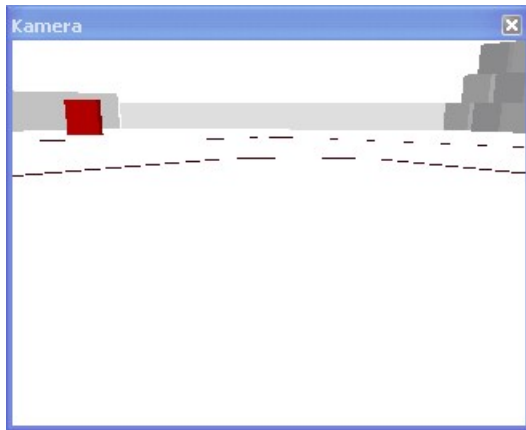
Bild	image	Das Live-Bild der Kamera auf Robotino.
------	-------	--

Einstellungen zu Farbraum und Auflösung werden im Robotino [Gerätedialog](#) vorgenommen.

FESTO



Dialog



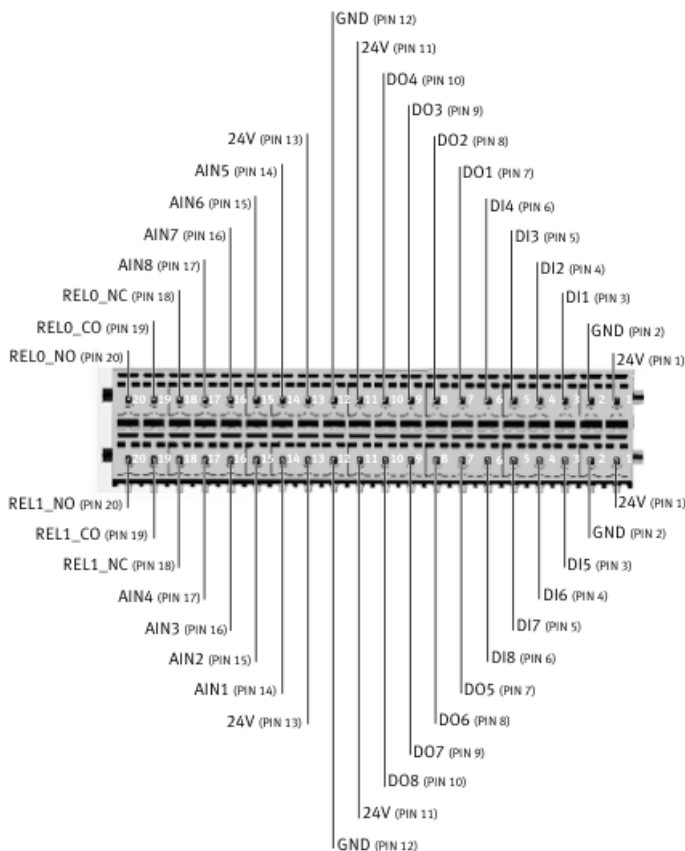
Der Kameradialog zeigt das aktuelle Kamerabild. Zur Änderung der Bildeinstellungen siehe [Gerätedialog](#).

FESTO



E/A Anschluss

Dieser Ordner enthält Funktionsblöcke zur Verwendung von Robotinos E/A Anschluss.



FESTO

Relais

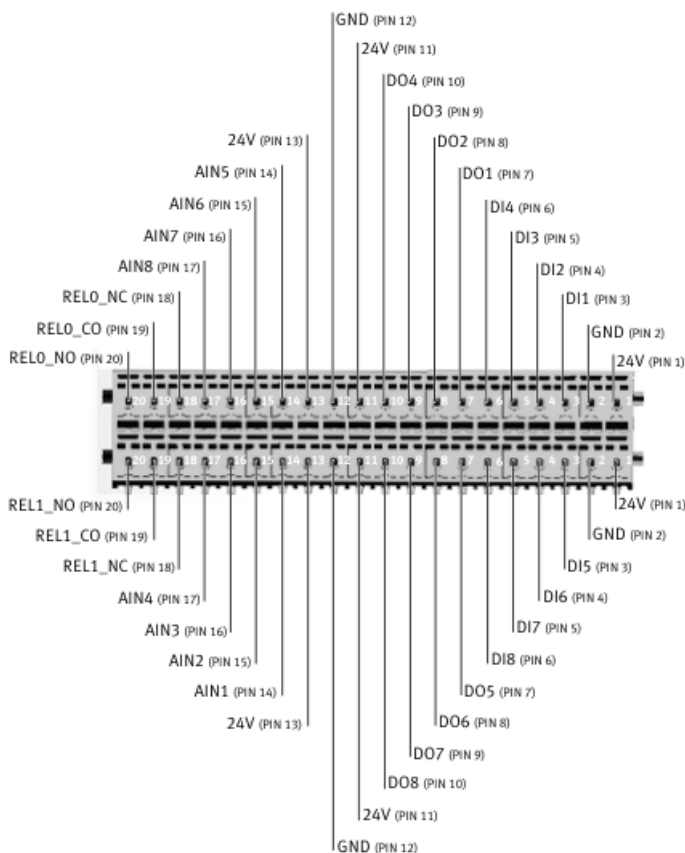


Schaltet die Relais 1 und 2.

Eingänge	Typ	Standard	Beschreibung
Wert	bool	false	Schaltet das gewählte Relais aus wenn unwahr (false), andernfalls ein.

Die Anschlüsse für Relais 1 sind REL1_NO, REL1_CO und REL1_NC.

Die Anschlüsse für Relais 2 sind REL2_NO, REL2_CO und REL2_NC.



FESTO

Digitaler Ausgang



Erlaubt das Setzen eines digitalen Ausgangs.

Eingänge	Typ	Standard	Beschreibung
----------	-----	----------	--------------

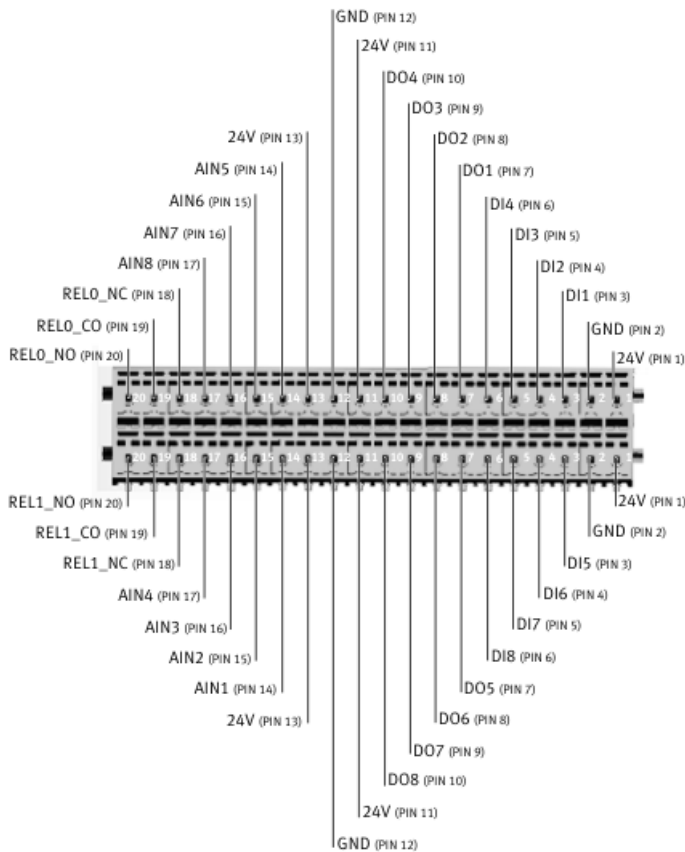
Wert	bool	false	Wenn wahr (true), wird der Ausgang an Robotinos E/A Anschluss auf +10V gesetzt. Wenn unwahr (false) liegen 0V am E/A Anschluss an.
------	------	-------	--

Der Anschluss für den digitalen Ausgang 1 ist DO1.

Der Anschluss für den digitalen Ausgang 2 ist DO2.

usw. bis

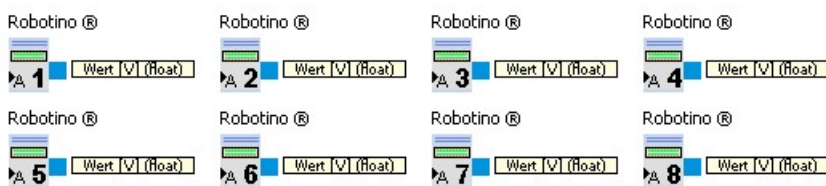
Der Anschluss für den digitalen Ausgang 8 ist DO8.



FESTO



Analoger Eingang



Liefert den Wert eines analogen Eingangs.

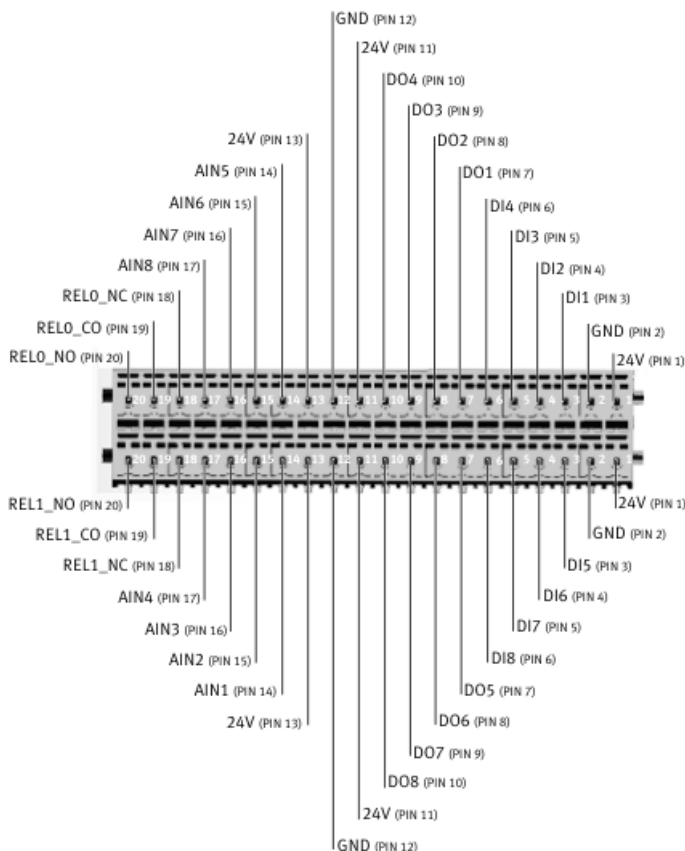
Ausgänge	Typ	Einheit	Beschreibung
Wert	float	Volt	Die an Robotinos E/A Anschluss gemessene Spannung zwischen 0V und 10V.

Der Anschluss für den analogen Eingang 1 ist AIN1.

Der Anschluss für den analogen Eingang 2 ist AIN2.

usw. bis

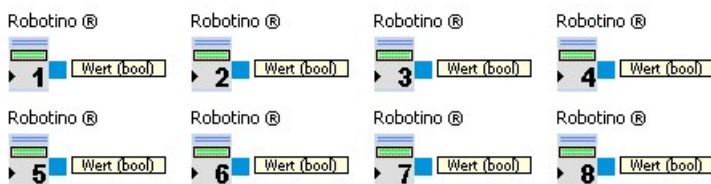
Der Anschluss für den analogen Eingang 8 ist AIN8.



FESTO



Digitaler Eingang



Liefert den Wert eines digitalen Eingangs.

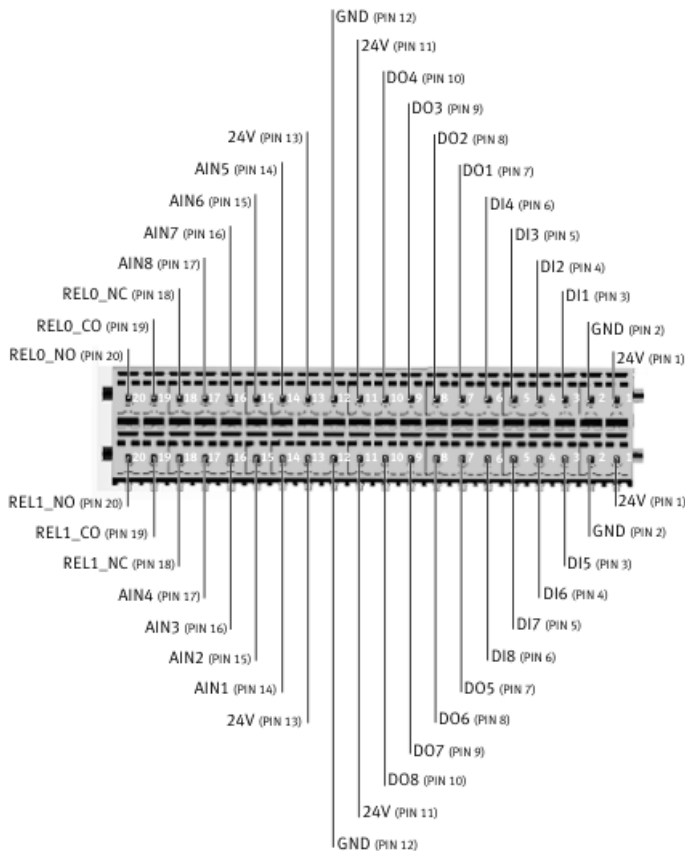
Ausgänge	Typ	Beschreibung
Wert	bool	Der an Robotinos E/A Anschluss angelegte Wert. Spannungen kleiner 5.75V ergeben unwahr (false). Spannungen größer 8.6V ergeben wahr (true). Liegt die Spannung am Eingang zwischen 5.75V und 8.6V bleibt der Wert unverändert.

Der Anschluss für den digitalen Eingang 1 ist DI1.

Der Anschluss für den digitalen Eingang 2 ist DI2.

usw. bis

Der Anschluss für den digitalen Eingang 8 ist DI8.



FESTO



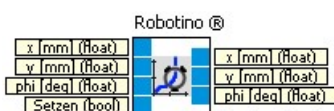
Navigation

Dieser Ordner enthält Funktionsblöcke zur Ortung von Robotino.

FESTO



Odometrie



Diese Funktion benötigt die 1GB Compact-Flash Speicherkarte von Robotino (V 1.7 oder höher).

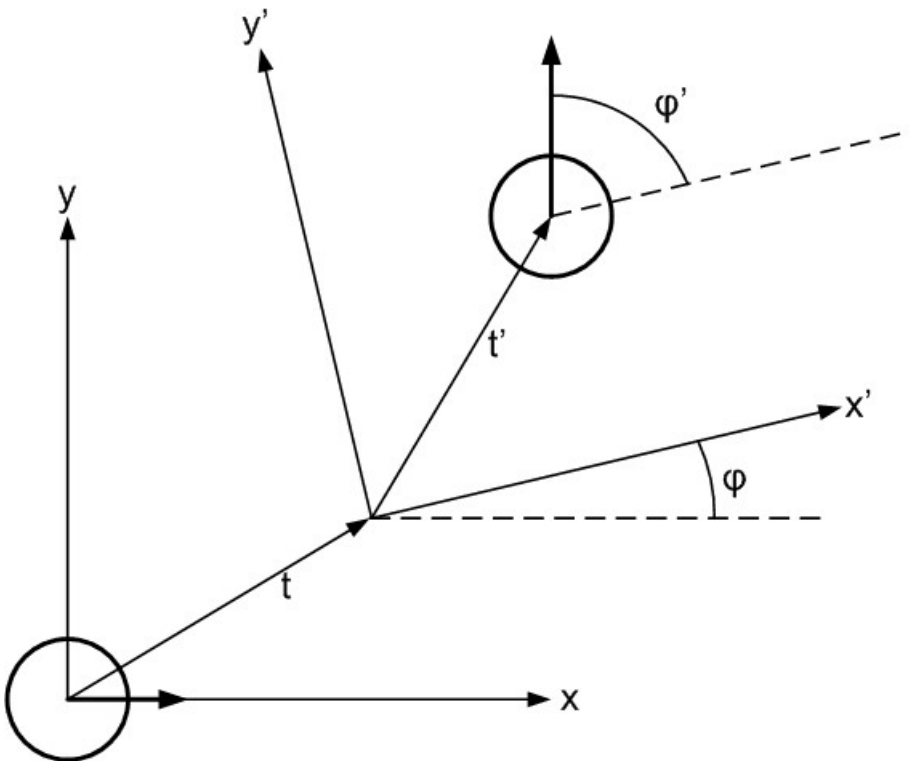
(Keine Funktion bei 256MB Speicherkarte, Version <=1.6)

Odometrie bezeichnet die Berechnung der aktuellen Position eines Fahrzeuges aufgrund der vorangegangenen Drehungen der einzelnen Räder. Das Wort Odometrie setzt sich aus den griechischen Wörtern "hodos" (Übersetzung: "Reise") und "metron" (Übersetzung: "Messung") zusammen. Siehe auch <http://de.wikipedia.org/wiki/Odometrie>.

Die Drehung der Räder wird in möglichst hoher zeitlicher Auflösung erfasst. Zu jedem Zeitschritt wird dann aufgrund der Radgeschwindigkeiten die zurückgelegte Wegstrecke des Fahrzeuges berechnet. Die (sehr kleinen) Wegstrecken der einzelnen Zeitschritte werden addiert, so dass man die aktuelle Position bezogen auf den Startpunkt erhält. Dieses Verfahren liefert lokal gute Ergebnisse. Auf langen Strecken oder unter widrigen Umständen (Schlupf der Räder durch Staub auf dem Boden, Drift des Roboters aufgrund einer Vorzugsrichtung des Teppichs usw.) kommt es bei diesem Verfahren zu großen Abweichungen. Aus diesem Grund wird Odometrie fast immer mit anderen Verfahren kombiniert, um die auftretenden Fehler zu kompensieren.

Eingänge	Typ	Einheit	Standard	Beschreibung
x	float	mm	0	Die neue x-Position. Der Wert wird übernommen, wenn am "Setzen" Eingang wahr (true) anliegt.

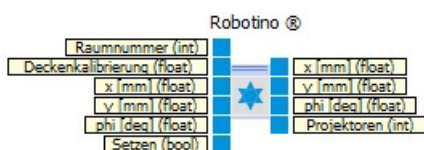
y	float	mm	0	Die neue y-Position. Der Wert wird übernommen, wenn am "Setzen" Eingang wahr (true) anliegt.
phi	float	Grad	0	Die neue Orientierung. Der Wert wird übernommen, wenn am "Setzen" Eingang wahr (true) anliegt.
Setzen	bool		false	Wenn der Eingang wahr (true) ist, wird Robotinos Odometrie auf die an den Eingängen x, y, phi anliegenden Werte gesetzt. Um die Odometrie auf (0,0,0) zu setzen reicht es daher, den Eingang für einen Zeitschritt auf wahr (true) zu setzen. Liegt unwahr (false) an, so ändert sich die Odometrie nur durch Bewegung der Räder von Robotino.
Ausgänge				
x	float	mm		Die aktuelle x-Position.
y	float	mm		Die aktuelle y-Position.
phi	float	Grad		Die aktuelle Orientierung.



FESTO



North Star



North Star® ist ein Sensor, der mit Hilfe von Projektoren die absolute Position von Robotino® ermittelt.

Eingänge	Typ	Einheit	Standard	Beschreibung
Raumnummer	int		3	Die Nummer des Raums, in welchem sich Robotino befindet. Räume werden mit 0 beginnend gezählt.
Deckenkalibrierung	float	mm	2800	Abstand zwischen Detektor und Decke. Bei einer Deckenhöhe von 3m beträgt der Abstand ca. 2800mm.
x	float	mm	0	x-Position des über den "Setzen" Eingang gesetzten Ursprungs.

y	float	mm	0	y-Position des über den "Setzen" Eingang gesetzten Ursprungs.
phi	float	Grad	0	Orientierung des über den "Setzen" Eingang gesetzten Ursprungs.
Setzen	bool		false	Wenn wahr (true) wird die Pose (x,y,phi) als Ursprung verwendet.
Ausgänge				
x	float	mm		Die aktuelle x-Position.
y	float	mm		Die aktuelle y-Position.
phi	float	Grad		Die aktuelle Orientierung.
Projektoren	int			Anzahl der sichtbaren Projektoren.

Room ID	Projector type	Switch-Setting	Frequency [Hz]
0	ProjectorKit NS1	Spot 1 SW2 = 0 Spot 2 SW2 = 8	1040 2350
1	ProjectorKit NS1	Spot 1 SW2 = 1 Spot 2 SW2 = 9	1150 2450
2	ProjectorKit NS1	Spot 1 SW2 = 2 Spot 2 SW2 = A	1250 2560
3	Projector NS2-50Hz	1	Spot A 3025 Spot B 3925
4	Projector NS2-50Hz	2	Spot A 3125 Spot B 4025
5	Projector NS2-50Hz	3	Spot A 3225 Spot B 4125



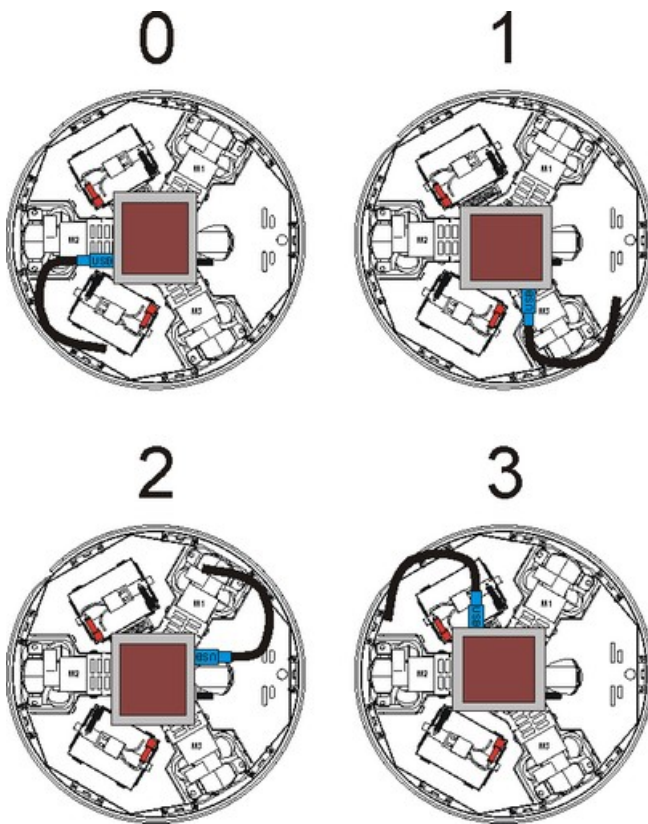
ProjectorKit NS1



Projector NS2

Der Northstar Detektor kann unterschiedlich auf Robotino montiert werden. Je nach Aufbau muss die Datei /etc/robotino/robotino.xml auf Robotino mit einem Editor angepasst werden. Der Wert für die Orientierung muss entsprechend der unten gezeigten Grafik eingestellt werden.

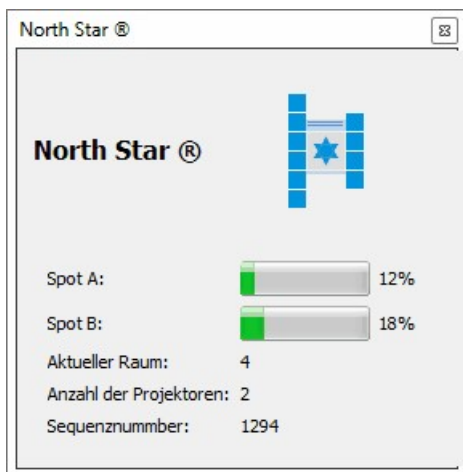
```
<NorthStar>
  <!--The orientation of the northstar sensor. See www.openrobotino.org-->
  <Orientation value="1" />
</NorthStar>
```



FESTO



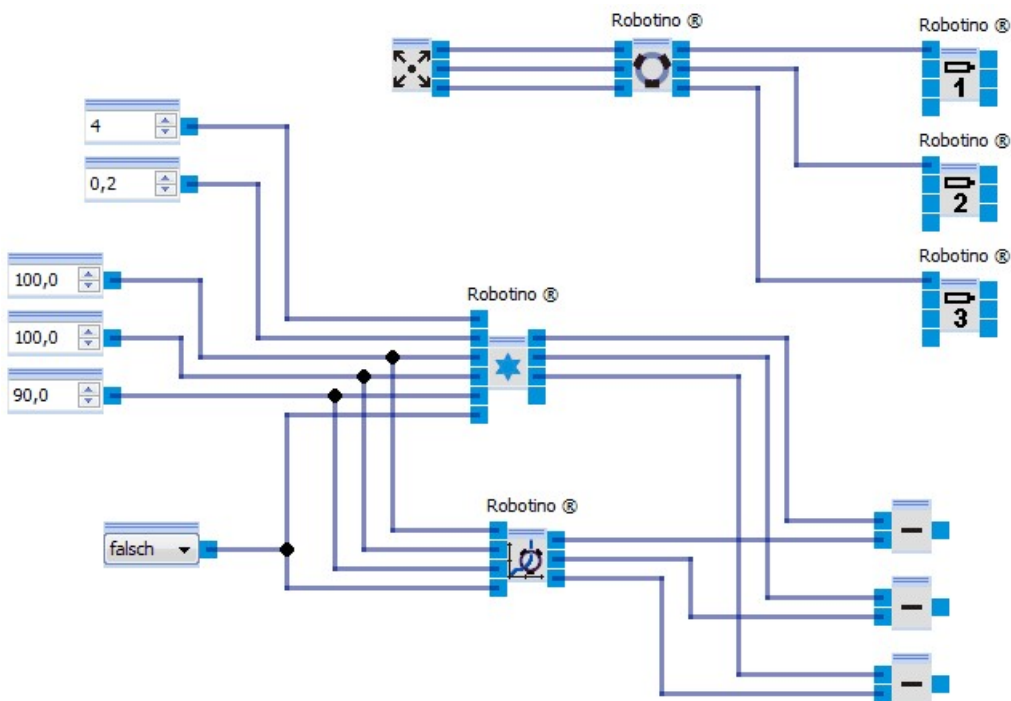
Dialog



Spot A	Intensität des vom Projektor emittierten ersten Lichtpunktes.
Spot B	Intensität des vom Projektor emittierten zweiten Lichtpunktes.
Aktueller Raum	Der vom NorthStar Sensor detektierte Raum.
Anzahl der Projektoren	Die Anzahl der für den NorthStar Sensor sichtbaren Projektoren.
Sequenznummer	Die Sequenznummer erhöht sich um 1 jedesmal wenn der NorthStar Sensor eine neue Positionsschätzung übermittelt.

FESTO

Beispiel



Das Steuerungsfeld wird benutzt, um Robotino zu verfahren.

Der Northstar soll den zu Raum 4 gehörigen Projektor detektieren. Der neue NorthStar Projektor muss dazu auf Raum 1 eingestellt sein.



Über die boolsche Konstante (wahr/falsch) können die Koordinatensysteme des Northstar und der Odometrie so transformiert werden, dass die aktuelle Northstar-Pose gleich der aktuellen Odometrie-Pose gleich (100,100,90) ist.

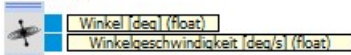
Durch Subtraktion der Northstar und Odometrie-Komponenten, sieht man den zwischen Odometrie und Northstar auftretenden Fehler.

FESTO

Gyroskop



Robotino®



Liefert die Messwerte von Robotinos internem Gyroskop.

Ausgänge	Typ	Einheit	Beschreibung
Winkel	float	deg (Grad)	Robotinos Orientierung in Weltkoordinaten
Winkelgeschwindigkeit	float	deg/s (Grad/Sekunde)	Robotinos Winkelgeschwindigkeit

Falls das Gyroskop keine Werte liefert, muss das Gyroskop über das Web-Interface von Robotino aktiviert werden

FESTO



E/A Erweiterung

Dieser Ordner enthält Funktionsblöcke zu weiteren Hardware-Schnittstellen von Robotino.

FESTO



Drehgebereingang



Mit diesem Baustein wird der auf Robotino zur Verfügung stehende freie Drehgebereingang angesprochen. Der Drehgebereingang wertet das Signal eines digitalen Drehgebers (A,B-Kanal Gray-Code) aus. Dabei werden sowohl steigende wie fallende Flanken berücksichtigt, so dass es zu einer 4fachen effektiven Drehgeberauflösung gegenüber der nominalen Drehgeberauflösung kommt.

Beispiel: Die Drehgeber der Motoren von Robotino sind mit einer Auflösung von 500 pro Umdrehung gekennzeichnet. Tatsächlich registriert man aber 2000 Impulse des Drehgebers pro Umdrehung.

Eingänge	Typ	Einheit	Standard	Beschreibung
Position zurücksetzen	bool		0	Wahr (true) = Position auf 0 setzen Unwahr (false) = Position nicht verändern
Ausgänge				
Geschwindigkeit	int	Impulse pro Sekunde		Die Geschwindigkeit des Drehgebers.
Position	int	Impulse		Das Integral über alle gemessenen Impulse seit Start von Robotino oder seit "Position zurücksetzen" = wahr.

FESTO



Leistungsausgang



Mit diesem Baustein kann der Leistungsausgang auf Robotino (ehemals Motor 4) angesprochen werden. Der Leistungsausgang kann nur verwendet werden, wenn das Unterprogramm keinen [Greifer](#) enthält.

Technisch realisiert wird dieser Ausgang durch eine H-Brücke, welche bis zu 5A Strom abgeben kann. Die H-Brücke wird mit einem hochfrequenten PWM Signal und einem Richtungsbit angesteuert. Der über den Eingang vorgegebene Sollwert setzt aufgrund seines Vorzeichens das Richtungsbit. Der Betrag des Sollwertes beeinflusst das PWM Signal. Ein Sollwert von 0 erzeugt kein PWM Signal, d.h. die H-Brücke liefert keinen Strom. Ein Sollwertbetrag von 50 stellt das Verhältnis von high zu low innerhalb des PWM Signals auf 50% ein. Ein Sollwertbetrag von 100 führt zu einem konstanten high, d.h. die H-Brücke liefert den maximalen Strom.

--	--	--	--	--

Eingänge	Typ	Einheit	Standard	Beschreibung
Sollwert	int		0	Setzt das Richtungsbit und das PWM Signal. Wertebereich -100 bis 100. Werte kleiner -100 werden auf -100 gesetzt. Werte größer 100 werden auf 100 gesetzt.
Ausgänge				
Strom	float	A		Der durch die H-Brücke fließende Strom.

Der am Leistungsausgang abgegebene Strom wird standardmäßig begrenzt. Um diese Begrenzung anzupassen oder abzuschalten muss auf Robotino die Datei /etc/robotino/robotino.xml editiert werden. Die neuen Werte werden nach 2s von Robotino eingelesen und berücksichtigt.



Greifer



Verwenden Sie diesen Baustein, um einen Festo Robotino Greifer zu bedienen. Der Greifer kann nur verwendet werden, wenn das Unterprogramm noch keinen [Leistungsausgang](#) enthält

Eingänge	Typ	Standard	Beschreibung
Öffnen	bool	false	Wahr (true) = Greifer öffnen Unwahr (false) = Greifer schließen
Ausgänge			
Geöffnet	bool		Wahr (true) = Greifer ist geöffnet
Geschlossen	bool		Wahr (true) = Greifer ist geschlossen

Der Greifer muss an Anschluss X15 auf der Platine hinter dem Akku angeschlossen werden:
links braunes Kabel (+) / rechts blaues Kabel (-)



Interne Sensoren

Enter topic text here.



Stromüberwachung

Robotino ®



Das Stromüberwachungsmodul für Robotino.

Eingänge	Typ	Einheit	Standard	Beschreibung
Ausgänge				
Stromverbrauch	float	A		Der aus den Batterien momentan entnommene Strom.

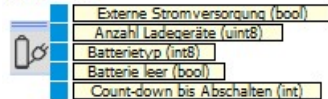
Batterie	float	Volt		Zellenspannung von Batterie 1 + 2
----------	-------	------	--	-----------------------------------

FESTO



Batterie

Robotino®



Das Stromüberwachungsmodul für Robotino.

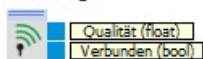
Eingänge	Typ	Einheit	Standard	Beschreibung
Ausgänge				
Externe Stromversorgung	bool		false	Wahr wenn Robotino an das externe Netzteil angeschlossen ist. Ansonsten falsch.
Anzahl Ladegeräte	uint8		0	Anzahl der Ladegeräte in Robotino.
Batterietyp	int8		0	0: Bleiakku 1: NiMH Akku -1: Unbekannter Akkutyp
Batterie leer	bool		false	Wahr wenn die Batteriespannung unter einen kritischen Wert gesunken ist. Ansonsten falsch.
Count-down bis Abschalten	int		0	Wenn die Batterie leer ist startet ein Count-down bei dessen Ablauf Robotino ausgeschaltet wird.

FESTO



Verbindung

Robotino®



Ausgänge				
Qualität	float		0	Zeigt die Qualität der Verbindung. 0: Sehr schlechte Qualität. Ein Verbindungsabbruch ist zu erwarten. 100: Maximale Qualität
Verbunden	bool		false	Ist wahr, wenn die Verbindung zu Robotino aufgebaut ist.

FESTO



Manipulator

FESTO



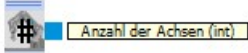
Leser

FESTO

Anzahl der Achsen



Robotino®



Gibt die Anzahl der angeschlossenen Achsen aus.

Ausgänge	Typ	Standard	Beschreibung
Anzahl der Achsen	int		Die Anzahl der angeschlossenen Achsen.

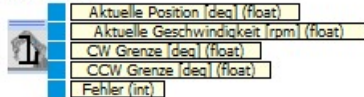
Siehe auch das [Beispiel](#) zur Verwendung des elektrischen Greifarms.

FESTO



Achse

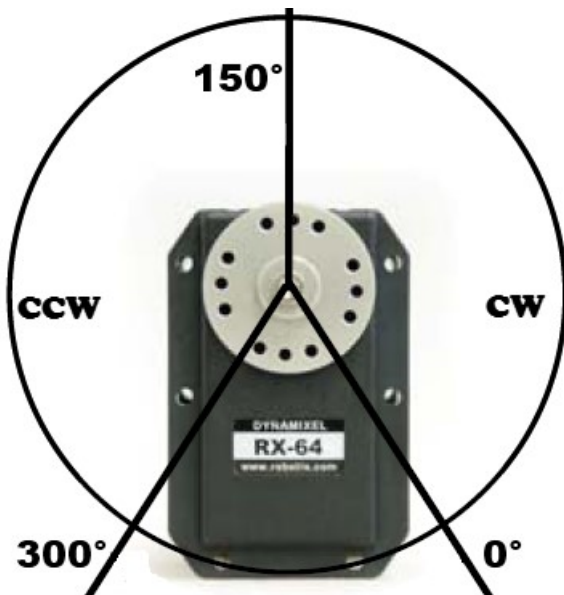
Robotino®



Gibt die Sensorwerte einer Achse aus.

Ausgänge	Typ	Einheit	Standard	Beschreibung
Aktuelle Position	float	Grad		Die aktuelle Achsenstellung.
Aktuelle Geschwindigkeit	float	rpm		Die aktuelle Drehgeschwindigkeit der Achse.
CW Grenze	float	Grad		Die Grenze bei Drehung im Uhrzeigersinn.
CCW Grenze	float	Grad		Die Grenze bei Drehung gegen den Uhrzeigersinn.
Fehler	int			<p>Anzeige eines Fehlers. Der Wert 0 zeigt an, dass kein Fehler vorliegt.</p> <p>Bit Beschreibung</p> <ul style="list-style-type: none"> 0 Falsche Eingangsspannung 1 Die Sollposition ist außerhalb der Winkelgrenzen. 2 Der Servo ist zu heiß. 3 Kommando spricht einen ungültigen Bereich an. 4 Prüfsummenfehler 5 Das für eine Bewegung erforderliche Drehmoment kann nicht aufgebracht werden. 6 Unbekanntes Kommando. 7 USB Fehler.

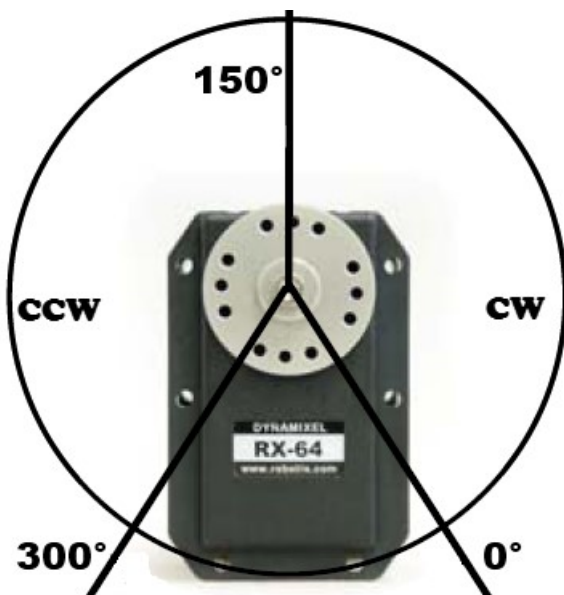
Siehe auch das [Beispiel](#) zur Verwendung des elektrischen Greifarms.



FESTO



Schreiber



FESTO



Positionsauswahl

Robotino®



Mit diesem Funktionsblock werden die gespeicherten Positionen angefahren.

Eingänge	Typ	Einheit	Standard	Beschreibung
Index	int		0	Die anzufahrende Positionsnummer. Die erste mögliche Position ist Position 1. Wenn der Index kleiner 1 oder größer der Anzahl der gespeicherten Positionen ist, bewegt sich der Arm nicht.
Name	string			Der Name der anzufahrenden Position. Wenn der Name keine leere Zeichenkette ist, wird nach einer Position mit gleichem Namen gesucht. Im Falle einer Übereinstimmung, wird die entsprechende Position angefahren. Falls keine Position mit dem angegebenen Namen gefunden wird, bewegt sich der Arm nicht.
Ausgänge				

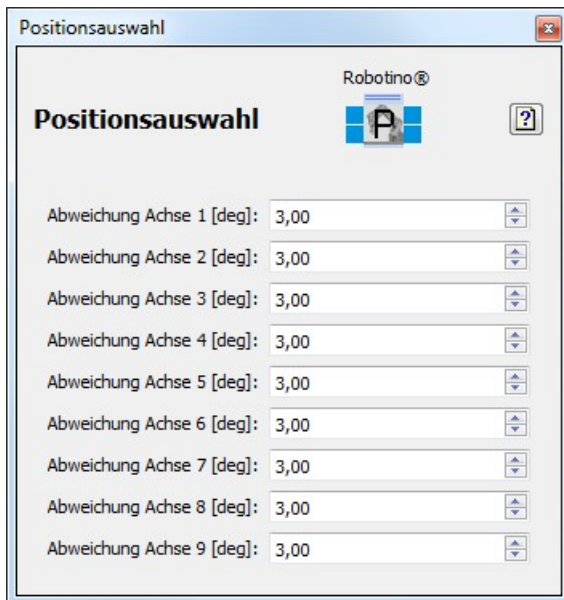
Position erreicht	bool			Ist wahr, wenn die Sollposition erreicht ist.
Anschläge überschritten	bool			Ist wahr, wenn die Sollposition aufgrund der Achsenbeschränkungen nicht angefahren werden kann.

Siehe auch das [Beispiel](#) zur Verwendung des elektrischen Greifarms.

FESTO



Dialog

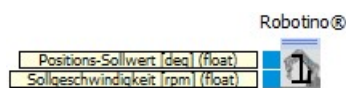


Über den Dialog kann die maximal zulässige Abweichung zur Sollposition pro Achse angegeben werden. Wenn alle Achspositionen innerhalb des Toleranzbereichs um die Sollposition sind, so wird der Ausgang "Position erreicht" wahr.

FESTO



Achse



Ansteuerung einer einzelnen Achse.

Eingänge	Typ	Einheit	Standard	Beschreibung
Positions-Sollwert	float	Grad	0	Der Sollwert der Achsenposition.
Sollgeschwindigkeit	float	Grad	0	Die Geschwindigkeit, mit der die Sollposition angefahren werden soll.

Siehe auch das [Beispiel](#) zur Verwendung des elektrischen Greifarms.

FESTO

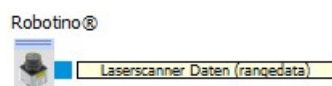


Laserscanner

FESTO



Laserscanner



Ausgabe der Laserscanner Daten.

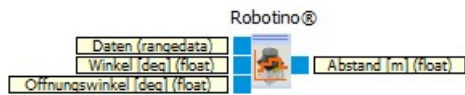
--	--	--	--

Ausgänge	Typ	Standard	Beschreibung
Laserscanner Daten	rangedata		Die Daten des Laserscanners.

FESTO



Laserscanner Auswertung



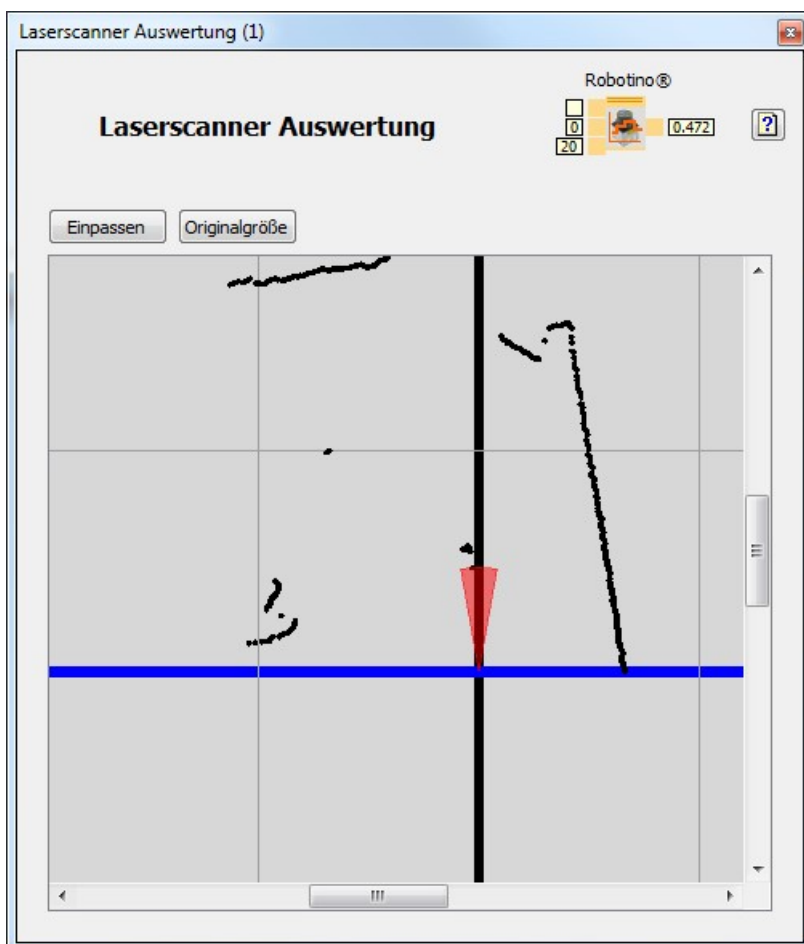
Auswertung der Laserscanner Daten.

Eingänge	Typ	Einheit	Standard	Beschreibung
Daten	rangedata			Die Daten des Laserscanners.
Winkel	Grad		0	Der Winkel, um den herum ausgewertet werden soll.
Öffnungswinkel	Grad		20	Der für die Auswertung herangezogenen Winkelbereich beginnt bei "Winkel - Öffnungswinkel / 2" und endet bei "Winkel + Öffnungswinkel / 2"
Ausgänge				
Abstand	float	m		Der minimale im Winkelbereich gemessene Abstand.

FESTO



Dialog



Im Dialog der Auswertung werden die vom Laserscanner gelieferten Abstandswerte dargestellt. Die rote Keule markiert den

ausgewerteten Winkelbereich. Die Länge der Keule richtet sich nach dem minimalen im Winkelbereich gemessenen Abstand.

FESTO



Laserscanner Information

Robotino®

Sequenznummer (uint32)
Zeitstempel (uint32)
Bezugspunkt (string)
Minimaler Winkel (deg) (float)
Maximaler Winkel (deg) (float)
Winkelschrittweite (deg) (float)
Schrittweite (s) (float)
Scanzeit (s) (float)
Minimale Reichweite (m) (float)
Maximale Reichweite (m) (float)
Anzahl der Distanzmesswerte (uint32)
Anzahl der Intensitätsmesswerte (uint32)

Informationen zum Laserscanner.

Ausgänge	Typ	Einheit	Beschreibung
Sequenznummer	int		Wird mit jedem empfangenen Datensatz um 1 erhöht.
Zeitstempel	int		Die Zeit in Millisekunden seit Start der Messungen.
Bezugspunkt	string		Der Bezugspunkt des Laserscanners.
Minimaler Winkel	float	Grad	Der Startwinkel der Messung.
Maximaler Winkel	float	Grad	Der Stopwinkel der Messung.
Winkelschrittweite	float	Grad	Der Winkelabstand der Strahlen.
Schrittweite	float	s	Zeitlicher Abstand zwischen zwei Messungen.
Scanzeit	float	s	Die für eine Messung benötigte Zeit.
Minimale Reichweite	float	m	Der Mindestabstand zwischen Laserscanner und Objekt.
Maximale Reichweite	float	m	Der maximale detektierbare Abstand zwischen Laserscanner und Objekt.
Anzahl der Distanzwerte	int		Anzahl der Distanz-Messwerte in einer Messung.
Anzahl der Intensitätswerte	int		Anzahl der Intensitäts-Messwerte in einer Messung.

FESTO



Robotino XT

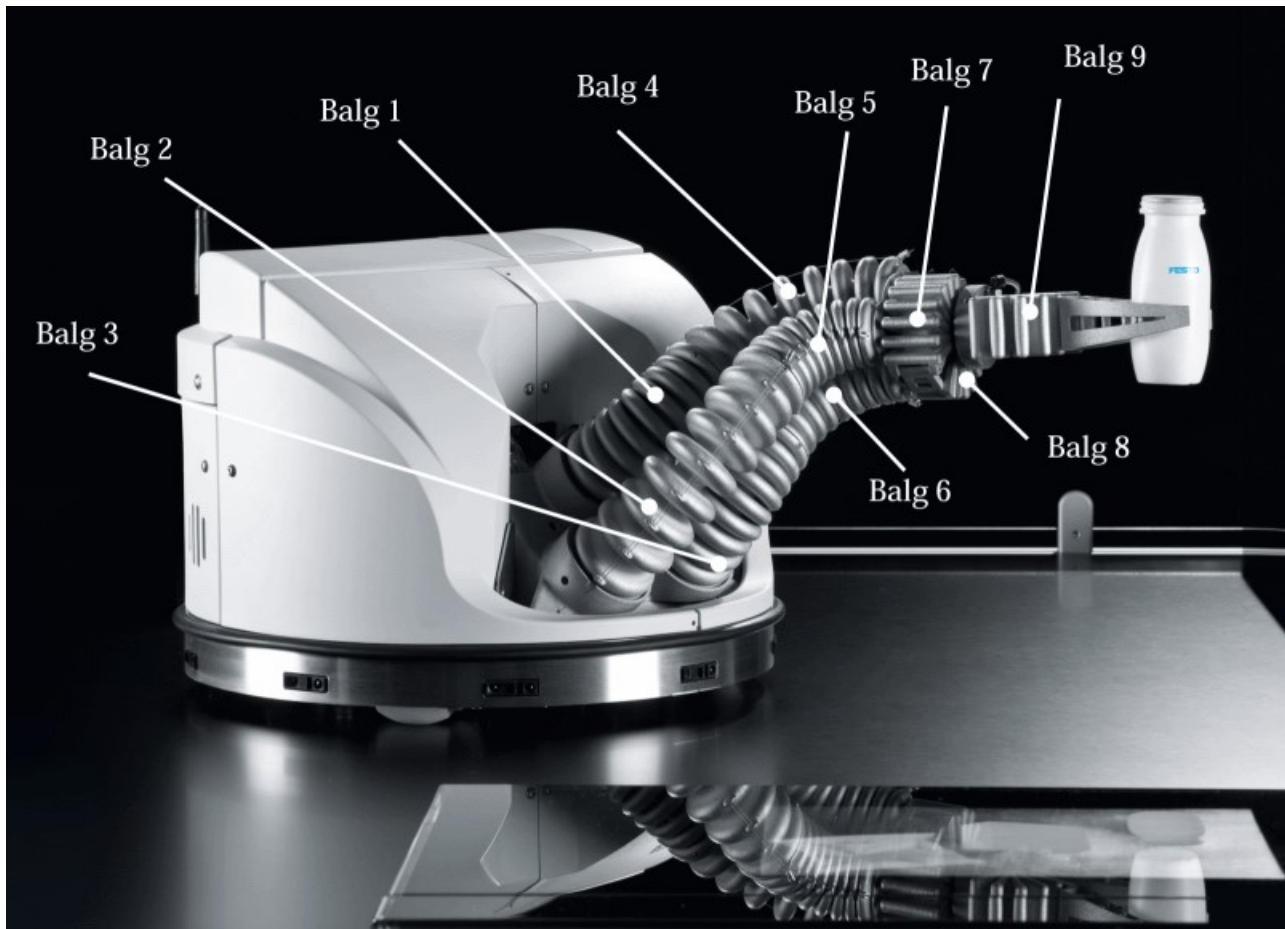


siehe http://www.festo.com/cms/en_corp/11367.htm

FESTO



Nummerierung der Bälge



FESTO



Sensoren

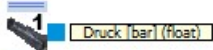
Als Sensoren stehen Druckmesssysteme für jede einzelne Kammer zur Verfügung.

FESTO



Balg

Robotino®



Liefert den Druck des entsprechenden Balgs.

Ausgänge	Typ	Einheit	Standard	Beschreibung
Druck	float	bar	0	Druck in der jeweiligen Kammer.

siehe [Nummerierung der Bälge](#)

FESTO



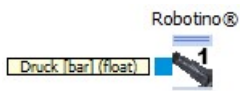
Aktoren

Die kompakte Version des bionischen Handhabungsassistenten wird über acht Kammern plus den Greifer bewegt. Es ist möglich den Druck für jede Kammer einzeln vorzugeben. Der Greifer wird über die Funktionsblöcke für die [Greiferventile](#) gesteuert. Es sind drei Funktionsblöcke vorhanden, mit denen die Bewegung des Arms vereinfacht wird.

- Segment XY: Umrechnung von xy-Koordinaten auf Drücke für ein Segment
- Greifer rotieren: Übersetzt einen Eingang in Drücke für die entsprechenden Bälge
- Greifer öffnen: Übersetzt einen Eingang in den passenden Zustand der Greiferventile

FESTO

Balg



Setzt den Druck in dem zugeordneten Balg.

Eingänge	Typ	Einheit	Standard	Beschreibung
Druck	float	bar	0	Solldruck in der jeweiligen Kammer.

siehe [Nummerierung der Bälge](#)

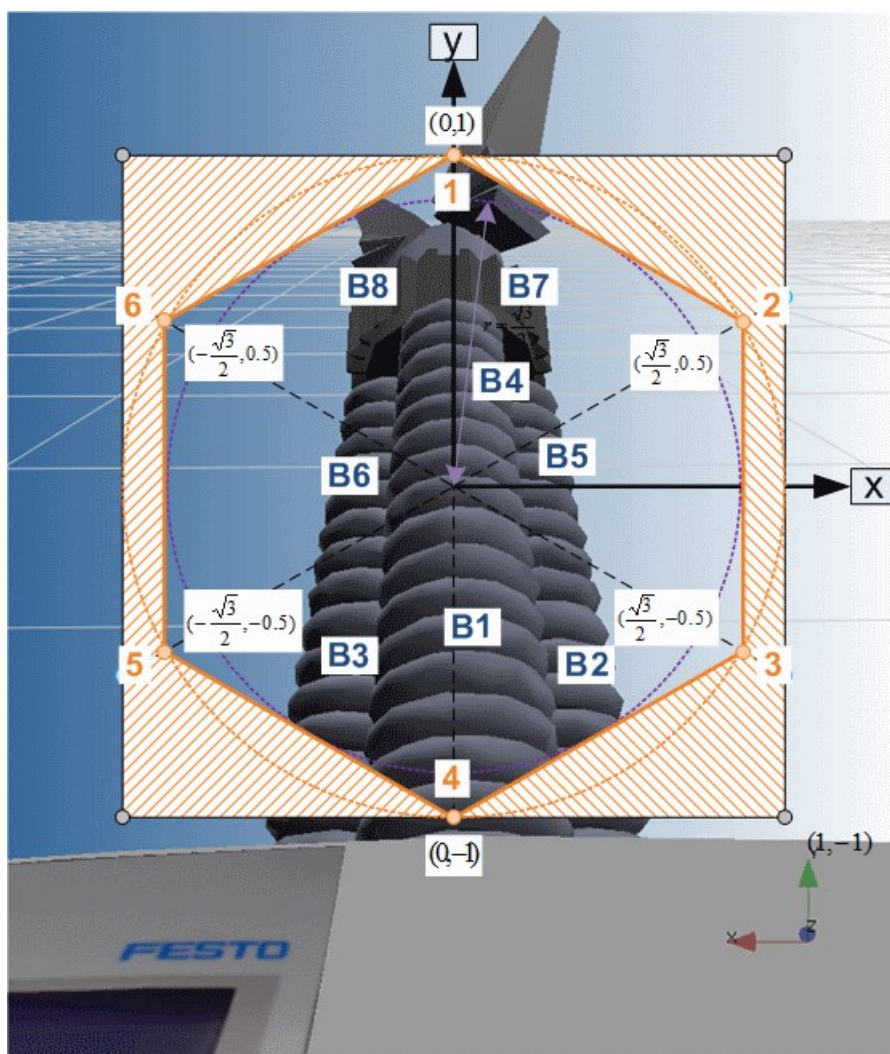
FESTO

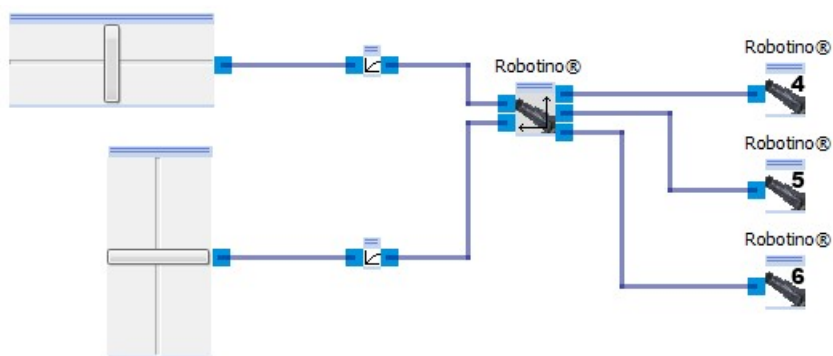
Segment XY



Rechnet xy-Koordinaten auf Drücke für die Bälge eines Segment um.

Eingänge	Typ	Einheit	Standard	Beschreibung
x	float		0	x-Auslenkung des Segments. Wertebereich [-1;1]
y	float		0	y-Auslenkung des Segments. Wertebereich [-1;1]
Ausgänge				
Balg #1	float	bar		Druck in der ersten Kammer des Segments. Die erste Kammer des ersten Segments ist Balg #1. Die erste Kammer des zweiten Segments ist Balg #4.
Balg #2	float	bar		Druck in der zweiten Kammer des Segments
Balg #3	float	bar		Druck in der dritten Kammer des Segments





← 🏠 →



Eingänge	Typ	Einheit	Standard	Beschreibung

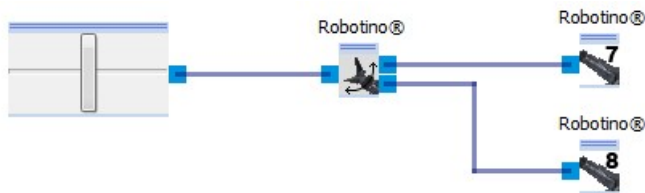
Rotieren	int		0	Wenn <0, dann rotiert der Greifer gegen den Uhrzeigersinn (gesehen von Robotino aus) Wenn 0, dann ist der Greifer nicht rotiert Wenn >0, dann rotiert der Greifer im Uhrzeigersinn (gesehen von Robotino aus)
Ausgänge				
Balg #7	float	bar		Druck in Balg #7
Balg #8	float	bar		Druck in Balg #8

siehe [Nummerierung der Bälge](#)

FESTO



Beispiel



FESTO



Greifer öffnen / schließen



Eingänge	Typ	Einheit	Standard	Beschreibung
Öffnen	bool		falsch	Wenn wahr, dann ist der Greifer geöffnet.
Schließen	bool		falsch	Wenn wahr und "Öffnen" falsch ist, dann ist der Greifer geschlossen.
Ausgänge				
Greiferventil #1	bool			Zustand für Greiferventil #1.
Greiferventil #2	bool			Zustand für Greiferventil #2.

FESTO



E/A

An die Ein- und Ausgänge der Steuerplatine sind weitere Sensoren und Aktoren angeschlossen:

Seilzugpotentiometer und Folienpotentiometer geben Auskunft über die Auslenkung der Segmente. Der Drucksensor zeigt an, ob der maximale Druck erreicht worden ist. Die in Robotino integrierten Kompressoren können ein- und ausgeschaltet werden. Über die Greiferventile wird die Bewegung des Greifers gesteuert, und durch das Wasserablassventil kann Wasser aus dem System entfernt werden.

FESTO



Seilzugpotentiometer



Liefert die Spannungswerte der Seilzugpotentiometer.

Ausgänge	Typ	Standard	Beschreibung
Poti #1 Wert	float	0	Spannung an Potentiometer 1.
...			
Poti #6 Wert	float	0	Spannung an Potentiometer 6.

FESTO



Folienpotentiometer

Robotino®



Liefert den Spannungswert des Folienpotentiometers.

Ausgänge	Typ	Standard	Beschreibung
Wert	float	0	Spannung am Potentiometer.

FESTO



Drucksensor

Robotino®



Liefert **true** zurück, falls der maximale Druck erreicht ist.

Ausgänge	Typ	Standard	Beschreibung
Wert	bool	false	true , falls der maximale Druck erreicht ist, sonst false .

FESTO



Kompressoren

Robotino®



Schaltet die Kompressoren ein oder aus. Falls die Kompressoren eingeschaltet sind, laufen sie nur, bis der maximale Druck erreicht ist, dann werden sie abgeschaltet, bis der Druck abgefallen ist.

Eingänge	Typ	Standard	Beschreibung
Wert	bool	false	Schaltet die Kompressoren ein oder aus.

FESTO



Greiferventil

Robotino®



Steuert das jeweilige Greiferventil an.

Eingänge	Typ	Standard	Beschreibung
Wert	bool	false	Steuert das jeweilige Ventil an.

Ansteuerung des Greifers:

Greiferventil #1	Greiferventil #2	Wirkung
false	false	Greifer öffnet sich.

false	true	Greifer hält die Position.
true	false	Undefiniertes Verhalten.
true	true	Greifer schließt sich.

FESTO



Wasserablassventil



Öffnet das Wasserablassventil, um angesammeltes Wasser aus dem System zu entfernen.

Eingänge	Typ	Standard	Beschreibung
Wert	bool	false	Öffnet das Ventil.

FESTO



Joystick

Das Gerät "Joystick" ermöglicht den Zugriff auf einen lokal angeschlossenen Joystick.

FESTO



Dialog



1	Liste der Joysticks	Hier werden alle am System angeschlossenen Joysticks angezeigt. Die Liste wird aktualisiert, wenn ein neuer Joystick an den Computer angeschlossen oder ein Joystick abgesteckt wird. Durch Auswahl eines Joysticks werden dessen Achsen und Knöpfe mit den entsprechenden Funktionsblöcken abrufbar.
2	Anzahl der Achsen	Die Anzahl der Achsen des Joysticks
3	Anzahl der Knöpfe	Die Anzahl der Knöpfe des Joysticks

FESTO



Funktionsblöcke

Die Funktionsblöcke ermöglichen die Verwendung des Gerätes "Joystick" in einem Unterprogramm.

FESTO



Knopf



Mit dem Knopf-Funktionsblock kann der Zustand eines Joystick-Knopfes abgefragt werden. Die Knopf-Nummer ist aus der Funktionsblock-Grafik ersichtlich.

Ausgänge	Typ	Beschreibung
Wert	bool	Wahr (true), wenn der Knopf gedrückt ist. Ansonsten unwahr (false).

FESTO



Achse



Mit dem Achsen-Funktionsblock kann der Zustand einer Joystick-Achse abgefragt werden. Die Achsen-Nummer ist aus der Funktionsblock-Grafik ersichtlich.

Ausgänge	Typ	Beschreibung
Wert	int	Wertebereich -1000 bis 1000.

FESTO



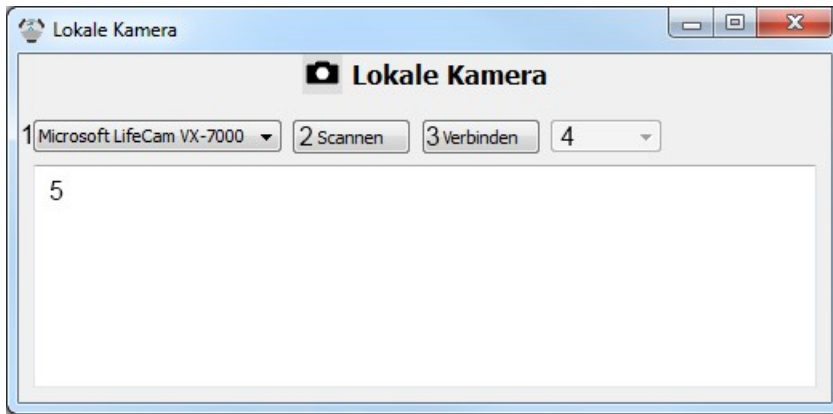
Lokale Kamera

Das Gerät "Lokale Kamera" ermöglicht den Zugriff auf eine lokal an den Rechner angeschlossene Kamera (z.B. eine Webcam).

FESTO

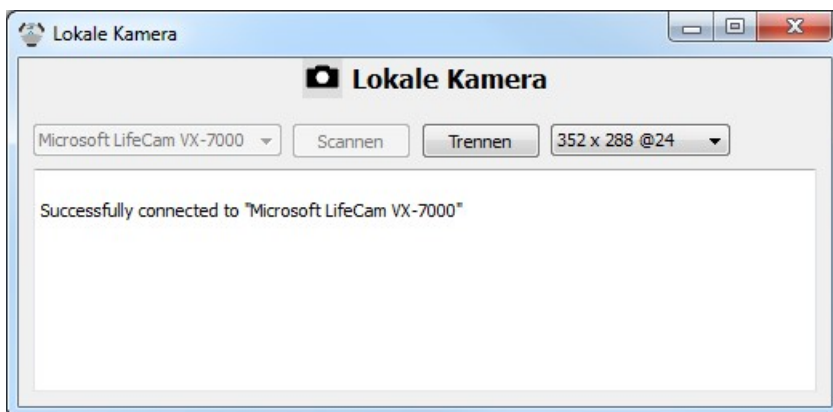


Dialog



1	Liste der Kameras	Hier werden alle am System angeschlossenen Kameras angezeigt. Die Liste wird aktualisiert, wenn eine neue Kamera an den Computer angeschlossen oder eine Kamera abgesteckt wird.
2	Scannen	Aktualisieren der Liste der Kameras.
3	Verbinden/Trennen	Herstellen/Trennen einer Verbindung zur ausgewählten Kamera.
4	Auflösung/Farbtiefe	Auswahl der Bildauflösung und Farbtiefe. Es stehen alle von der Kamera unterstützten Auflösungen zur Verfügung.
5	Nachrichtenfenster	Anzeigen von Statusmeldungen.

Nach dem Auswählen einer Kamera muss zunächst eine Verbindung hergestellt werden. Dann kann die gewünschte Auflösung eingestellt werden.



FESTO



Funktionsblöcke

Die Funktionsblöcke ermöglichen die Verwendung des Gerätes "Lokale Kamera" in einem Unterprogramm.

FESTO



Kamera

Lokale Kamera



Liefert ein Live-Bild der lokalen Kamera.

Eingänge	Typ	Standard	Beschreibung

Ausgänge			
Bild	image		Das Live-Bild der lokalen Kamera.

Einstellungen zu Farbtiefe und Auflösung werden im [Gerätedialog](#) vorgenommen.

FESTO



Dialog



Der Kameradialog zeigt das aktuelle Kamerabild. Zur Änderung der Bildeinstellungen siehe [Gerätedialog](#).

FESTO



OPC Client

OPC ist eine Standardschnittstelle zwischen verschiedenen Anwendungsprogrammen und Baugruppentreibern (z.B. SPS). Mehrere OPC-Clients können sich an einem OPC-Server anmelden. Oftmals werden (spezielle) OPC-Server von SPS-Herstellern zur Verfügung gestellt.

Im unteren Beispiel wird ein Festo EasyPort über den kostenlosen Festo EzOPC-Server mit RobotinoView verbunden.

Der EzOPC-Server stellt die Ein-/Ausgänge von bis zu 4 EasyPorts dabei in sogenannten "Groups" und "Tags" zur Verfügung:

- Gerät 1 bildet Group "EasyPort1"
- Ausgang 1 hat somit das Tag "EasyPort1.OutputPort1"

Gehen Sie wie folgt vor

1. EzOPC-Server installieren.
2. EzOPC-Server starten und "Prozess-Simulation ..." und "SPS über EasyPort" auswählen.
3. RobotinoView starten.
4. Gerät "OPC Client" hinzufügen.
5. Im Kontextmenü des Gerätedialogs Vordefinierte Einstellungen ► "Festo EzOPC EasyPort" auswählen. Die Standardwerte für den EasyPort werden geladen.
6. "FestoDidactic.EzOPC.1" auswählen, falls notwendig.
7. Verbindung starten.
8. Verwenden Sie die OPC-Bausteine um die Ein-/Ausgangswerte des OPC-Server zu verarbeiten.

TIPP: möchten Sie eine SPS eines anderen Herstellers anbinden benötigen Sie von diesem Hersteller entweder einen OPC-Server oder OPC-Client. Verwenden Sie einen OPC Client um zu sehen, welche OPC-Server mit welchen Tags auf Ihrem PC zur Verfügung stehen.

Downloads und weitere Informationen erhalten Sie z.B. unter <http://www.opcconnect.com/>



Dialog

1 FestoDidactic.EzOPC.2

	Tag	
DO_Port_1	EasyPort1.OutputPort1	4
DO_Port_2	EasyPort1.OutputPort2	
DO_Port_3		
DO_Port_4		
DI_Port_1	EasyPort1.InputPort1	
DI_Port_2	EasyPort1.InputPort2	
DI_Port_3		
DI_Port_4		
AO_Port_1	EasyPort1.AnalogOut0	
AO_Port_2	EasyPort1.AnalogOut1	
AO_Port_3		
AO_Port_4		
AI_Port_1	EasyPort1.AnalogIn0	
AI_Port_2	EasyPort1.AnalogIn1	
AI_Port_3	EasyPort1.AnalogIn2	
AI_Port_4	EasyPort1.AnalogIn3	

1	Auswahl OPC Server	Die ComboBox zeigt alle z.Zt. aktiven lokalen OPC Server an.
2	Verbinden	Durch drücken dieses Schalters wird eine Verbindung zu dem Ausgewählten OPC Server hergestellt.
3	Suche	Durch drücken dieses Schalters wird die Liste der OPC Server aktualisiert.
4	Zuordnungsliste	In der Tabelle wird die Zuordnung zwischen den Funktionsblöcken und den OPC "Tags" festgelegt.

Zeilenname	Funktionsblock
DO_Port_1	Digitaler Ausgang 1
DO_Port_2	Digitaler Ausgang 2
DO_Port_3	Digitaler Ausgang 3
DO_Port_4	Digitaler Ausgang 4
DI_Port_1	Digitaler Eingang 1
DI_Port_2	Digitaler Eingang 2
DI_Port_3	Digitaler Eingang 3
DI_Port_4	Digitaler Eingang 4
AO_Port_1	Analoger Ausgang 1
AO_Port_2	Analoger Ausgang 2

AO_Port_3	Analoger Ausgang 3
AO_Port_4	Analoger Ausgang 4
AI_Port_1	Analoger Eingang 1
AI_Port_2	Analoger Eingang 2
AI_Port_3	Analoger Eingang 3
AI_Port_4	Analoger Eingang 4

Über das Kontextmenü des Dialogs stehen folgende Funktionen zur Verfügung:

Vordefinierte Einstellungen ▶	Festo EzOPC VirtualPLC
Laden	Festo EzOPC EasyPort
Speichern	
Help	

Vordefinierte Einstellungen ▶ Festo EzOPC VirtualPLC	Lädt eine zur VirtualPLC passende Zuordnung zwischen Funktionsblöcken und Tag.
Vordefinierte Einstellungen ▶ Festo EzOPC EasyPort	Lädt eine zum EasyPort passende Zuordnung zwischen Funktionsblöcken und Tag.
Laden	Lädt eine Zuordnung von der Festplatte.
Speichern	Speichert die aktuelle Zuordnung auf der Festplatte.
Help	Anzeige der Online-Hilfe.

FESTO



Funktionsblöcke

Die Funktionsblöcke ermöglichen die Verwendung des Gerätes "OPC Client" in einem Unterprogramm.

FESTO



Eingänge

FESTO



Analoger Eingang



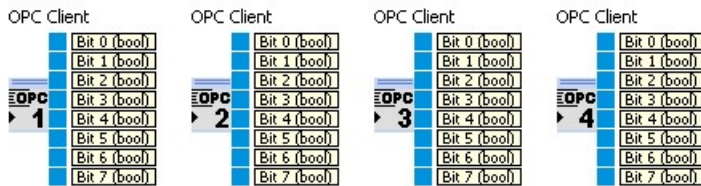
Liefert den Wert des der Zeile AI_Port_x zugeordneten "Tag".

Ausgänge	Typ	Beschreibung
Wert	int	Wertebereich 0 bis 65535

FESTO



Digitaler Eingang



Liefert die Bitwerte des der Zeile DI_Port_x zugeordneten "Tag".

Ausgänge	Typ	Beschreibung
Bit 0	bool	Wahr (true), wenn das Bit 0 in dem vom OPC Server gelieferten Wert gesetzt ist.
...		
Bit 7	bool	Wahr (true), wenn das Bit 7 in dem vom OPC Server gelieferten Wert gesetzt ist.

FESTO



Ausgänge

FESTO



Analoger Ausgang



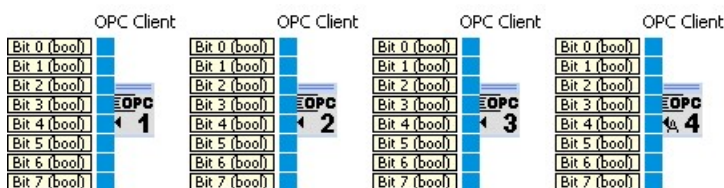
Schreibt den Eingangswert in das der Zeile AO_Port_x zugeordneten "Tag".

Eingänge	Typ	Beschreibung
Wert	int	Wertebereich 0 bis 65535

FESTO



Digitaler Ausgang



Setzt die einzelnen Bitwerte zu einem Wert zusammen und schreibt diesen in das der Zeile DO_Port_x zugeordneten "Tag".

Eingänge	Typ	Beschreibung
Bit 0	bool	Wenn wahr (true), wird der an den OPC Server gelieferte Wert um $2^0 = 1$ erhöht.
...		

Bit 7	bool	Wenn wahr (true), wird der an den OPC Server gelieferte Wert um $2^7 = 128$ erhöht.
-------	------	---



Datenaustausch

Die Geräte dieser Gruppe ermöglichen den Austausch von Daten über das Netzwerk zwischen verschiedenen Robotino View Instanzen.

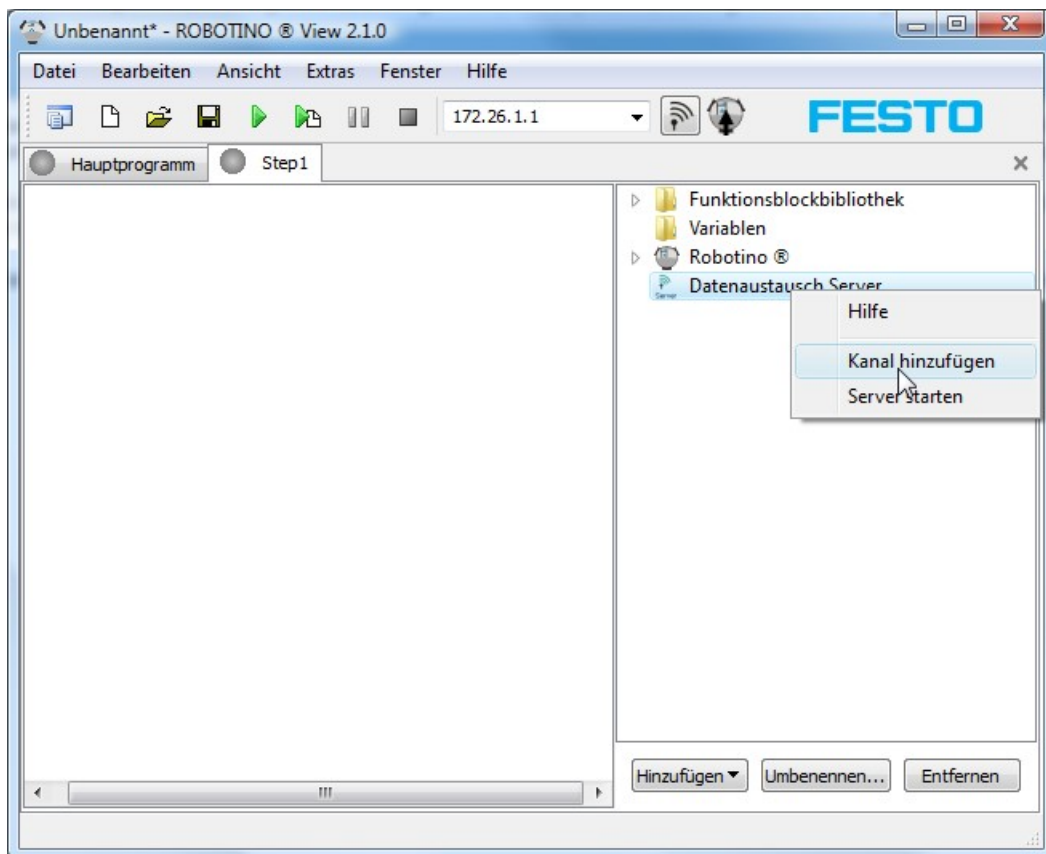


Server

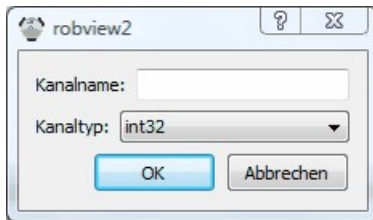
Der Datenaustausch Server kann von beliebig vielen Klienten benutzt werden, um Daten über eine beliebige Anzahl von Kommunikationskanäle auszutauschen. Auf dem Server werden die zur Verfügung stehenden Kommunikationskanäle angelegt. Die zur Verfügung stehenden Kommunikationskanäle werden den Klienten mitgeteilt. Die Klienten können dann wählen, über welche Kanäle sie mit dem Server kommunizieren wollen.

Server und Klienten sind beim Austausch der Daten gleichberechtigt. Schreibt ein Klient in einen Kommunikationskanal, so wird das Datum zunächst an den Server und von dort an alle anderen Klienten gesendet. Schreiben mehrere Teilnehmer gleichzeitig in einen Kanal, kann nicht vorhergesagt werden, welches Datum schlussendlich bei allen Teilnehmern ankommt.

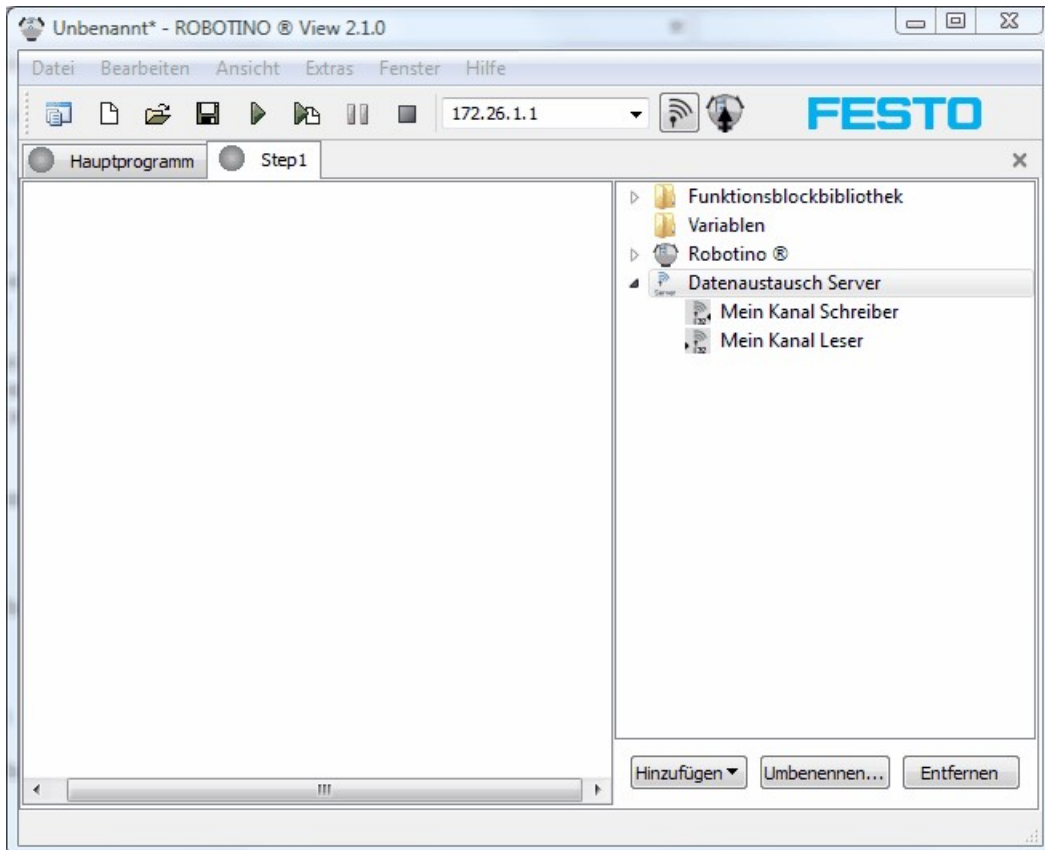
Nachdem das Datenaustausch Server Gerät zur Funktionsblockbibliothek hinzugefügt worden ist, können Kommunikationskanäle über das Kontextmenü des Datenaustausch Servers hinzugefügt werden.



Es erscheint ein Dialog, in welchem der Name und der Typ des neuen Kanals festgelegt wird.

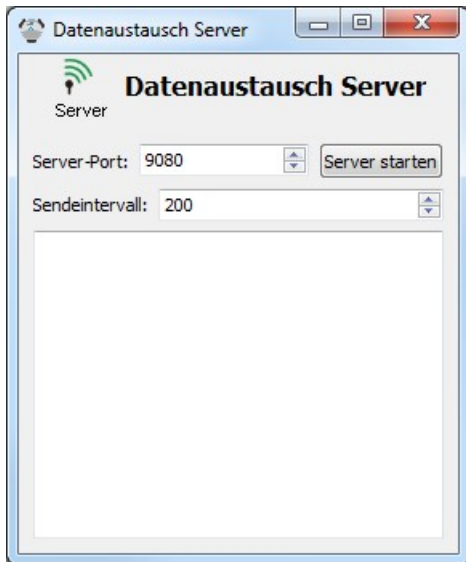


Der Kanalname muss eindeutig sein und darf nur ASCII Zeichen ohne "/" enthalten. Mit Ok wird der Kanal angelegt. In der Funktionsblockbibliothek erscheinen die Funktionsblöcke "Kanalname Schreiber" und "Kanalname Leser" zum schreiben in bzw. lesen aus dem Kanal.

**FESTO**

Dialog





Der Dialog des Datenaustausch Servers wird mit einem Doppelklick auf den Eintrag des Gerätes in der Funktionsblockbibliothek geöffnet.

"Server-Port" legt die TCP-Portnummer fest, auf welcher der Server Verbindungsanfragen von Klienten entgegennimmt.

"Sendeintervall" legt das Zeitintervall nach einer Datenübertragung fest, nach deren Ablauf erst wieder gesendet werden darf.

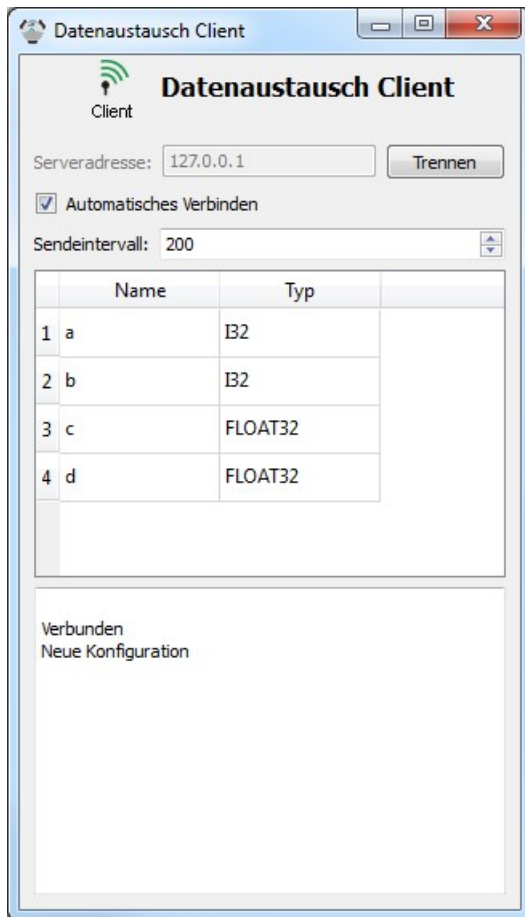
Mit "Server starten" wird der Server gestartet. Ab jetzt können Klienten eine Verbindung zum Server aufbauen.

FESTO

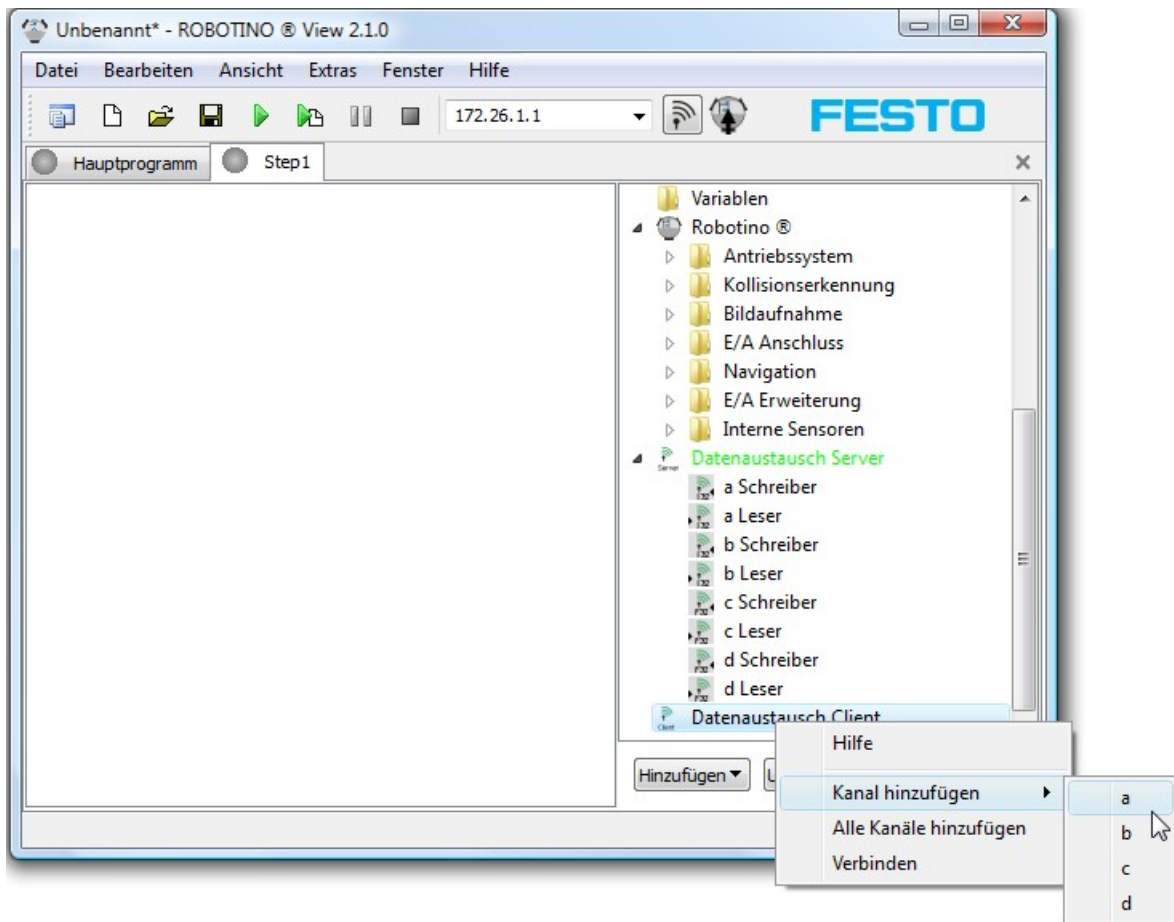
Client

Der Datenaustausch Client verbindet sich zu einem [Datenaustausch Server](#) und kann mit diesem Daten über vom Server festgelegte Kommunikationskanäle austauschen.

Nachdem sich der Datenaustausch Client erfolgreich mit einem [Datenaustausch Server](#) verbunden hat, steht die Liste der Kommunikationskanäle zur Verfügung.



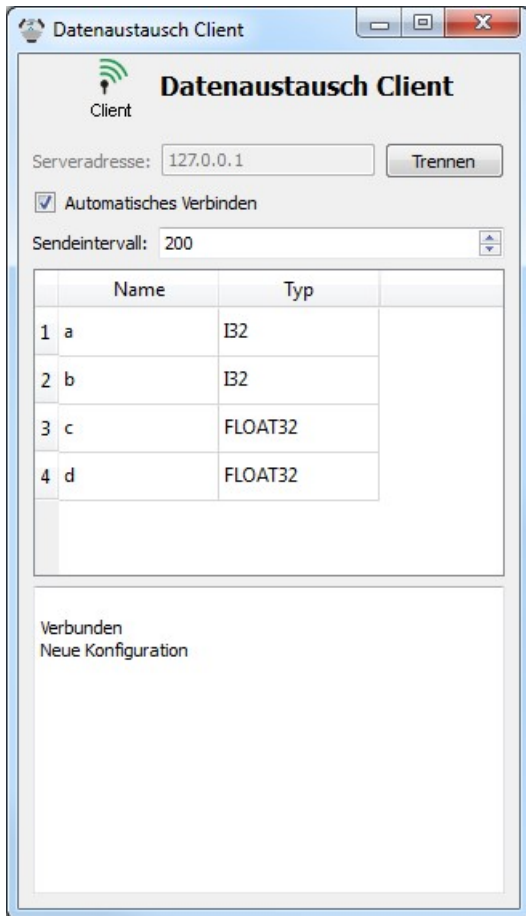
Auf dem Server wurden die Kommunikationskanäle a,b vom Typ I32 (Integer mit 32 Bit) und c,d (Fließkommazahl mit 32 Bit) angelegt. Diese Kanäle können jetzt dem Client in der Funktionsblockbibliothek hinzugefügt werden.



Wie beim [Datenaustausch Server](#) erscheinen nach dem hinzufügen eines Kanals ein Leser und Schreiber Funktionsblock. Über das Kontextmenü lassen sich die Kanäle einzeln und alle auf einmal hinzufügen. Über "Verbinden" kann die Verbindung zum Server aufgebaut werden, ohne dass der Dialog des Klienten geöffnet werden muss.

FESTO

Dialog



Serveradresse bezeichnet die IP-Adresse des Servers, zu dem eine Verbindung aufgebaut werden soll. Wird nur die IP-Adresse angegeben, so wird versucht eine Verbindung über den Standardport 9080 aufzubauen. Lauscht der Server auf einem anderen Port, so kann dieser hinter der IP-Adresse mit einem ":" getrennt angegeben werden.

Ist z.B. ein Server auf dem eigenen Rechner auf Port 8000 aktiv, so lautet die Serveradresse 127.0.0.1:8000.

Mit "Automatisches Verbinden" wird festgelegt, ob eine unterbrochene Verbindung vom Klienten automatisch neu aufgebaut werden soll.

"Sendeintervall" legt das Zeitintervall nach einer Datenübertragung fest, nach deren Ablauf erst wieder gesendet werden darf.

FESTO



Funktionsblöcke

Über die Funktionsblöcke werden Daten mit den Geräten ausgetauscht.

FESTO



Leser



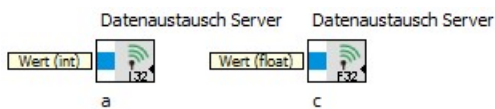
Der Leser liest Daten aus einem Kommunikationskanal.

Ausgänge	Typ	Beschreibung

Wert	int, float, float array, laser range data	Der Wert des Kommunikationskanals
------	--	-----------------------------------



Schreiber



Der Schreiber schreibt Daten in einen Kommunikationskanal.

Eingänge	Typ	Standard	Beschreibung
Wert	int, float, float array, laser range data	0	Der Wert wird an den Server gesendet und von dort an alle Klienten verteilt.



UDP Datenaustausch

Mit dem UDP-Datenaustausch-Gerät werden Daten zwischen Robotino View und externen Anwendungen per UDP ausgetauscht.



Protokoll

Spezifikation der Datenstruktur

Byte	Funktion
0	Nachrichten ID
1-2	Anzahl Bytes der gesamten Nachricht N vom Typ UINT16
3	Prüfsumme (bei der Paketerstellung mit 0 initialisieren, siehe Prüfsumme)
N-1	Letztes Byte der Nachricht



Prüfsumme

Ist die Nachricht kürzer als 100 Byte, so wird aus den einzelnen Bytes des gesamten Paketes die Byte-Summe s0 gebildet. Ist die Nachricht 100 Byte lang oder mehr, so berechnet sich s0 aus den ersten und den letzten 50 Byte der Nachricht.

In beiden Fällen muss das Byte für die Prüfsumme mit 0 initialisiert sein. Die Prüfsumme berechnet sich zu

Prüfsumme = 0xff - s0

```
unsigned char checksum( const unsigned char* Nutzdaten, unsigned int NutzdatenLänge ) const
{
    unsigned char s0 = 0;

    if( NutzdatenLänge < 100 )
    {
        for( int i = 0; i < NutzdatenLänge; ++i )
        {
            s0 += Nutzdaten[i];
        }
    }
}
```

```

    }
}
else
{
    for( int i = 0; i < 50; ++i )
    {
        s0 += Nutzdaten[i];
    }
    for( int i = NutzdatenLänge-1; i >= NutzdatenLänge - 50; --i )
    {
        s0 += Nutzdaten[i];
    }
}

return ( 0xFF - s0 );
}

```

Um zu überprüfen, ob das Paket korrekt übertragen wurde, werden für Nachrichten kürzer als 100 Byte die einzelnen Bytes des gesamten Paketes zur Byte-Summe s1 aufsummiert. Ist die Nachricht 100 Byte lang oder mehr, so berechnet sich s1 aus den ersten und den letzten 50 Byte der Nachricht.

Das Paket ist korrekt, wenn gilt

s1 = 0xFF



Datentypen



Typ	Breite in Bytes	Beschreibung
UINT16	2	<p>Byte0: low Byte1: high</p> <p>Auf einem little endian System kann ein UINT16 Datenwert direkt in die Nutzdaten kopiert werden</p> <p>Beispiel:</p> <pre>//encoding uint16 value = 9873; char Nutzdaten[2]; uint16* p = reinterpret_cast<uint16*>(Nutzdaten); *p = value; //decoding value = *(reinterpret_cast<const uint16*>(Nutzdaten));</pre>
INT32	4	<p>Byte0: low Byte3: high</p> <p>Auf einem little endian System kann ein INT32 Datenwert direkt in die Nutzdaten kopiert werden</p> <p>Beispiel:</p> <pre>//encoding int32 value = -3459873; char Nutzdaten[4]; int32* p = reinterpret_cast<int32*>(Nutzdaten); *p = value; //decoding value = *(reinterpret_cast<const int32*>(Nutzdaten));</pre>
UINT32	4	<p>Byte0: low Byte3: high</p> <p>Auf einem little endian System kann ein UINT32 Datenwert direkt in die Nutzdaten kopiert werden</p> <p>Beispiel:</p> <pre>//encoding uint32 value = 3459873; char Nutzdaten[4]; uint32* p = reinterpret_cast<uint32*>(Nutzdaten); *p = value; //decoding value = *(reinterpret_cast<const uint32*>(Nutzdaten));</pre>



Nachricht 0



Byte	Funktion
0	0
1	36
2	0
3	Prüfsumme
4-7	INT0 vom Typ INT32
8-11	INT1 vom Typ INT32
12-15	INT2 vom Typ INT32
16-19	INT3 vom Typ INT32
20-23	INT4 vom Typ INT32
24-27	INT5 vom Typ INT32
28-31	INT6 vom Typ INT32
32-35	INT7 vom Typ INT32

FESTO

Nachricht 1

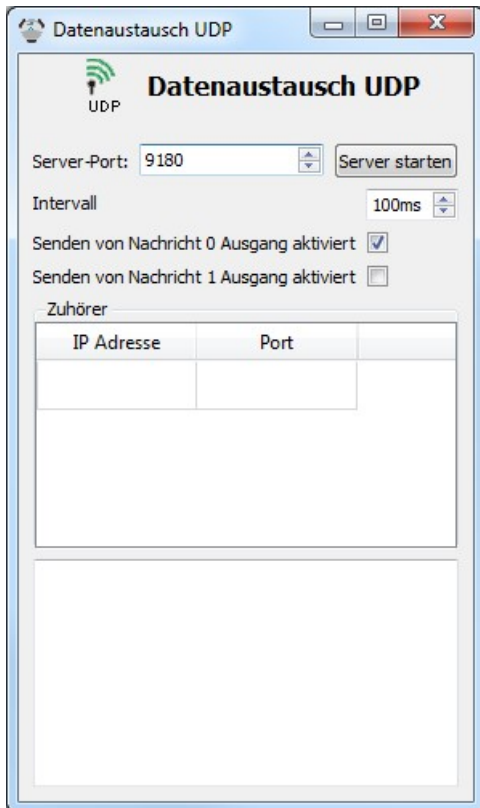


Byte	Funktion
0	1
1	36
2	0
3	Prüfsumme
4-7	INT0 vom Typ INT32
8-11	INT1 vom Typ INT32
12-15	INT2 vom Typ INT32
16-19	INT3 vom Typ INT32
20-23	INT4 vom Typ INT32
24-27	INT5 vom Typ INT32
28-31	INT6 vom Typ INT32
32-35	INT7 vom Typ INT32

FESTO

Dialog





Der Dialog des UDP-Datenaustausch-Geräts wird mit einem Doppelklick auf den Eintrag des Gerätes in der Funktionsblockbibliothek geöffnet.

Im Dialog wird das Senden sowie das Empfangen von UDP-Datenpaketen konfiguriert:

"Server-Port" legt die UDP-Portnummer fest, auf welcher UDP-Datagramme empfangen werden und von welcher UDP-Datagramme gesendet werden.

Mit "Server starten" wird der Server gestartet. Ab jetzt werden UDP-Datenpakete entgegengenommen und ausgewertet sowie gesendet.

"Intervall" legt das Zeitintervall nach einer Datenübertragung fest, nach deren Ablauf erst wieder gesendet werden darf.

Das Senden von Nachricht 0 oder Nachricht 1 kann individuell ein- oder ausgeschaltet werden.

Unter "Zuhörer" können IP-Adressen und Ports der Empfänger eingetragen werden. Wird kein Port eingetragen, so wird Port 9180 als Standard verwendet.

FESTO



Funktionsblöcke

Über die Funktionsblöcke werden Daten mit den Geräten ausgetauscht.

FESTO



Nachricht 0

Die Funktionsblöcke in der Kategorie Nachricht 0 erlauben das senden und empfangen von Daten.

FESTO



Eingang

Die Eingänge der Nachricht 0 stellen empfangene Werte bereit.

FESTO

Leser



Der Leser liest Daten stellt empfangene Daten bereit. Es gibt je einen Leser für INT0 bis INT7.

Ausgänge	Typ	Beschreibung
Wert	int	Der empfangene Wert

FESTO

Ausgang



Über die Ausgänge können Werte versendet werden.

FESTO

Schreiber



Der Schreiber nimmt die zu sendenden Daten entgegen und übergibt sie dem Gerät zur Weiterleitung an die Empfänger per UDP. Es gibt je einen Schreiber für INT0 bis INT7.

Eingänge	Typ	Beschreibung
Wert	int	Der zu sendende Wert

FESTO

Nachricht 1



Nachricht 1 ist identisch zu [Nachricht 0](#).

FESTO

Beispiel



