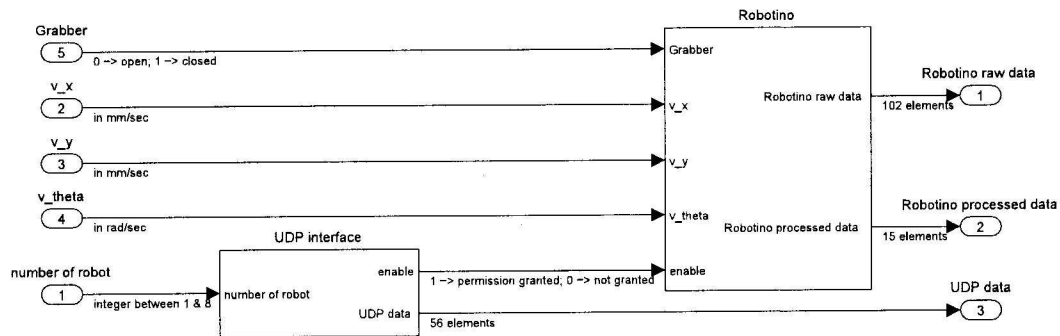


Starterkit Robotino

Simulink-Modell



Inhaltsverzeichnis

1 Subsystem „Robotino“	1
1.1 Eingänge	1
1.2 Ausgänge	2
2 Subsystem „UDP interface“	4
2.1 Eingang	4
2.2 Ausgänge	4

1 Subsystem „Robotino“

Der Block „Robotino“ ermöglicht die Kommunikation mit dem Robotersystem von Festo Didactic. Über den Eingang *Grabber* kann ein elektrischer Greifer geöffnet und geschlossen werden. Zum Ansteuern des omnidirektionalen Antriebes sind die drei Eingänge v_x , v_y und v_{θ} vorhanden. Der fünfte Eingang *enable* aktiviert bzw. deaktiviert die Ansteuerung des Roboters.

Der Ausgang *Robotino raw data* ist ein Array, der den gesamten Datensatz, mit dem das Robotersystem antwortet beinhaltet. Die Elemente von *Robotino processed data* geben aufbereitete Daten der Robotino-Antwort wieder.

1.1 Eingänge

Grabber: Öffnen bzw. Schließen des elektrischen Greifers. Ein Wert von 0 bewirkt ein öffnen des Greifers, ein Wert von 1 schließt ihn.

v_x: Längsgeschwindigkeit in $\frac{mm}{s}$. Ein positiver Wert lässt den Roboter in Richtung des zweitgrößten Reflektors (Richtungsreflektor) fahren, ein negativer Wert in die entgegengesetzte.

v_y: Quergeschwindigkeit in $\frac{mm}{s}$. Ein positiver Wert lässt den Roboter (in Fahrtrichtung blickend) nach links fahren, ein negativer Wert nach rechts.

v_theta: Rotationsgeschwindigkeit in $\frac{rad}{s}$. Ein positiver Wert lässt den Roboter in mathematisch positiver Richtung (links herum) drehen, ein negativer Wert in mathematisch negativer (rechts herum).

enable: Freischalten bzw. Blockieren der übrigen Eingänge des Robotino-Blocks. Ein Wert von 1 aktiviert die Eingänge und ermöglicht, dass der Roboter angesteuert werden kann. Ein Wert von 0 zieht die Eingänge auf 0 und verhindert ein Ansteuerung des Roboters.

1.2 Ausgänge

Robotino processed data: Array mit 15 Elementen aufbereiteter Daten der Robotino-Antwort.

- Element 1: Lichtschranke am elektrischen Greifer.
0: kein Werkstück zwischen den Zangen.
1: Werkstück erkannt.
- Element 2: Schieber am elektrischen Greifer.
0: kein Signal am Schieber.
1: Werkstückvorrichtung erreicht.
- Element 3 - 5: Rückgabewerte der Längs-, Quer- und Rotationsgeschwindigkeit von Robotino. Einheiten und Vorzeichen verhalten sich wie bei den Eingängen s. o.
- Element 6 - 14: Daten der neun Abstandssensoren des Robotinos in *mm*.
- Element 15: Zeigt eine Kollision des Robotinos an. Die Bumper-Daten wurden umgerechnet, so dass ein Wert von 1 eine kollisionsfreie Fahrt und 0 eine Kollision anzeigt. Bei 0 bleibt der Roboter stehen und kann erst wieder bewegt werden, wenn die Simulation gestoppt und erneut gestartet wurde.

Robotino raw data: Array mit 102 Elementen. Beinhaltet die Rohdaten der Robotino-Antwort (siehe Tabelle 1).

Element	Bit	Beschreibung
1	0 - 7	Anzahl der folgenden Elemente: 101
2	0 - 7	Start-Byte 0, das Zeichen „R“
3	0 - 7	Start byte 1, das Zeichen „E“
4	0 - 7	Start byte 2, das Zeichen „C“
5	0 - 7	Master ADU 0 Bits 2 - 10
6	0 - 7	Master ADU 1 Bits 2 - 10
7	0 - 7	Master ADU 2 Bits 2 - 10
8	0 - 7	Master ADU 3 Bits 2 - 10
9	0 - 7	Master ADU 4 Bits 2 - 10
10	0 - 7	Master ADU 5 Bits 2 - 10
11	0 - 7	Master ADU 6 Bits 2 - 10
12	0 - 7	Master ADU 7 Bits 2 - 10
13	0 - 1	Master ADU 0 Bits 0 - 1
	2 - 3	Master ADU 1 Bits 0 - 1
	4 - 5	Master ADU 2 Bits 0 - 1
	6 - 7	Master ADU 3 Bits 0 - 1
14	0 - 1	Master ADU 4 Bits 0 - 1
	2 - 3	Master ADU 5 Bits 0 - 1
	4 - 5	Master ADU 6 Bits 0 - 1
	6 - 7	Master ADU 7 Bits 0 - 1
15	0 - 7	Master time
16	0 - 7	Slave 0 ADU 0 Bits 2 - 10
17	0 - 7	Slave 0 ADU 1 Bits 2 - 10
18	0 - 7	Slave 0 ADU 2 Bits 2 - 10
19	0 - 7	Slave 0 ADU 3 Bits 2 - 10
20	0 - 7	Slave 0 ADU 4 Bits 2 - 10
21	0 - 7	Slave 0 ADU 5 Bits 2 - 10
22	0 - 7	Slave 0 ADU 6 Bits 2 - 10
23	0 - 7	Slave 0 ADU 7 Bits 2 - 10
24	0 - 1	Slave 0 ADU 0 Bits 0 - 1
	2 - 3	Slave 0 ADU 1 Bits 0 - 1
	4 - 5	Slave 0 ADU 2 Bits 0 - 1
	6 - 7	Slave 0 ADU 3 Bits 0 - 1
25	0 - 1	Slave 0 ADU 4 Bits 0 - 1
	2 - 3	Slave 0 ADU 5 Bits 0 - 1
	4 - 5	Slave 0 ADU 6 Bits 0 - 1
	6 - 7	Slave 0 ADU 7 Bits 0 - 1
26	0	Drehrichtung Motor 0, 0: negativ, 1: positiv
27	0 - 7	Geschwindigkeit Motor 0
28	0 - 7	Position Motor 0 Bits 0 - 7
29	0 - 7	Position Motor 0 Bits 8 - 15
30	0 - 7	Position Motor 0 Bits 16 - 23
31	0 - 7	Position Motor 0 Bits 24 - 31
32	0	Digitaler Eingang 0
	1	Digitaler Eingang 1
	2	Digitaler Eingang 2
	3	Digitaler Eingang 3
	4	Bumper
33	0 - 7	Motorlaufzeit Bits 0 - 7
34	0 - 7	Motorlaufzeit Bits 8 - 15
35	0 - 7	Motorlaufzeit Bits 16 - 23
36	0 - 7	Motorlaufzeit Bits 24 - 31
37 - 57		Motor 1: wie Bytes 14 - 34
58 - 78		Motor 2: wie Bytes 14 - 34
79 - 99		Motor 3 (optional): wie Bytes 14 - 34
100	0 - 7	Stopp-Byte 0, das Zeichen „r“
101	0 - 7	Stopp-Byte 1, das Zeichen „e“
102	0 - 7	Stopp-Byte 2, das Zeichen „c“

Tabelle 1: Robotino raw data

2 Subsystem „UDP interface“

Der Block „UDP interface“ dient zum Empfangen der Kameradaten. Der erste Ausgang *enable* liefert das Signal für die Ansteuerbarkeit des Robotinos, während der zweite *udp* die kompletten Kameradaten in einem Array mit einer Länge von 56 Elementen liefert. Sind während eines Taktes keine „neuen“ Kameradaten vorhanden, werden die letzten ausgegeben.

Über den Eingang *number of robot* ist die Nummer des Roboters einzugeben, um unkontrolliertes Verhalten von Robotino zu verhindern, wenn eine fehlerhafte Ortung auftritt.

2.1 Eingang

number of robot: Auf Grund der Reflektoranordnung auf dem Roboter besitzt dieser eine Ziffer, die auf der Kommandobrücke festgehalten ist. Wird diese an den Eingang *number of robot* geleitet, stoppt der Roboter, so lange er von den Kameras nicht geortet werden kann. Dies kann beispielsweise geschehen, wenn er den Kamerabereich verlässt oder die Reflektoren verdeckt werden.

2.2 Ausgänge

enable: Dient zum Freischalten bzw. Blockieren des omnidirektionalen Antriebs des Robotino-Blocks (s. o.).

UDP data: Array mit einer Länge von 56 Elementen.

- Elemente 1 bis 24 geben die Position in *mm* und die Ausrichtung im Bogenmaß von bis zu acht Robotern an. Jeweils drei Elemente sind für einen Roboter reserviert. Diese folgen direkt aufeinander. Als Beispiel hat Roboter 2 die Elemente 4 bis 6. Der Ausschnitt der drei Elemente sieht folgendermaßen aus: [x-Position, y-Position, Ausrichtung].
- Elemente 25 bis 27 dienen als Zeitstempel. Der Ausschnitt der drei Elemente sieht wie folgt aus: [Stunde, Minute, Sekunde].
- Element 28 beinhaltet eine Fahrterlaubnis. Hat dieses Element den Wert 0, werden sich die Roboter nicht bewegen.
- Elemente 29 bis 36 dienen als Variablen, die von der *camera.mdl* aus veränderbar sind. So kann beispielsweise eine Geschwindigkeitsänderung während der Fahrt realisiert werden.
- Elemente 37 bis 56 dienen zur Übermittlung der Koordinaten von maximal zehn Checkpoints. Dabei benötigt jeder Checkpoint zwei aufeinander folgende Elemente. Im ersten Element ist die x-Position in Millimetern und im zweiten Element die y-Position in Millimetern angegeben. So geben die Elemente 39 und 40 die Position des zweiten Checkpoints wieder. Der jeweilige Ausschnitt sieht folgendermaßen aus: [x-Position, y-Position].