

Intelligente servoelektrische Greifmodule der WSG-Serie
Betrieb an Robotersystemen von Mitsubishi

Firmware Version 4.0.0
Mai 2015



www.weiss-robotics.com

Inhalt

1	Einleitung	3
1.1	Allgemeines	3
1.2	Systemvoraussetzungen	3
1.3	Einschränkungen	3
2	Vorbereitung	4
2.1	Konfiguration des Greifmoduls	4
2.2	Konfiguration des Roboters	5
2.2.1	Netzwerkeinstellungen der Roboterschnittstelle	6
3	Inbetriebnahme.....	9
4	Beschreibung der Roboterprogramme	12
4.1	Hauptprogramm „GripCycle“	12
4.2	Hauptprogramm „Test“	12
4.3	Unterprogramme zur Abstraktion der GCL-Befehle.....	12
4.3.1	WSGBye.....	13
4.3.2	WSGVerbose	13
4.3.3	WSGDevType.....	13
4.3.4	WSGHome	14
4.3.5	WSGMove	14
4.3.6	WSGGrip.....	14
4.3.7	WSGSysFlag	15
4.3.8	WSGGripState	15
4.3.9	WSGPosition.....	16
4.3.10	WSGSpeed.....	16
4.3.11	WSGForce.....	16
4.3.12	WSGTemp	17
4.3.13	WSGFastStop.....	17
4.3.14	WSGFSack	17
4.3.15	WSGFTYPE	18
4.3.16	WSGTare	18
4.4	Hilfsprogramme	18
4.4.1	ParseErr	18

1 Einleitung

1.1 Allgemeines

Dieses Dokument beschreibt die Einrichtung und den Betrieb von Greifmodulen der WSG-Serie an Mitsubishi-Robotern mit Steuerungen des Typs *CRnD-7xx/CR75x-D* in der Programmiersprache MELFA-BASIC V unter der Programmierumgebung *RT ToolBox2* über die TCP/IP-Netzwerkschnittstelle.

Es wird vorausgesetzt, dass der Roboter, wie in der Betriebsanleitung beschrieben, fertig eingerichtet und in Betrieb genommen wurde. Zudem sind Grundkenntnisse der Roboterprogrammierung mit MELFA-BASIC V erforderlich. Die Informationen aus der Montage- und Betriebsanleitung des Greifmoduls werden ebenfalls als bekannt vorausgesetzt.

Die Verbindung zwischen Roboter und Greifmodul erfolgt über die TCP/IP-Netzwerkschnittstelle mit Hilfe des textbasierten Befehlsprotokolls „Gripper Control Language“ (GCL). Dieses Protokoll ist verfügbar ab Firmware-Version 4.0.0. Weiterführende Informationen befinden sich im Dokument „WSG GCL Reference Manual“ (engl.), das im Lieferumfang des Greifmoduls enthalten ist.

1.2 Systemvoraussetzungen

Das vorliegende Beispielprogramm wurde auf einem Roboter des Typs *Mitsubishi RV-4FL-D* mit Steuerung *CR750-D* in der Programmiersprache MELFA-BASIC V unter der Programmierumgebung *RT ToolBox2* implementiert und getestet. Eine Portierung auf andere, auf MELFA-BASIC V basierende Systeme sollte in der Regel mit geringem Aufwand durchführbar sein. Die uneingeschränkte Lauffähigkeit des Programms auf anderen Steuerungen kann jedoch nicht garantiert werden.

1.3 Einschränkungen

Bei dem vorliegenden Roboterprogramm handelt es sich um ein Programmierbeispiel, das die Kommunikation zwischen Robotersteuerung und Greifmodul exemplarisch demonstrieren soll.

Trotz sorgfältiger Prüfung kann Weiss Robotics keine Garantie für die Stabilität und Fehlerfreiheit des Programms übernehmen.

Vor der Verwendung des Programms im Produktiveinsatz sind geeignete Tests durchzuführen, die die ordnungsgemäße Funktion im Betrieb gewährleisten. Dies gilt auch und insbesondere bei Verwendung von einzelnen Teilen des Programms. Die Verantwortung hierfür liegt beim Anlagenprogrammierer.

2 Vorbereitung

Schließen Sie das Greifmodul über ein entsprechendes Kabel an die Stromversorgung an. Verbinden Sie das Netzkabel über einen geeigneten Switch mit dem selben Netzwerk, mit dem auch der Roboter verbunden ist. Über diesen Switch können auch weitere Komponenten wie z. B. ein Laptop oder eine Verbindung mit dem Hausnetz angeschlossen werden.

Beachten Sie die entsprechenden Hinweise in der Montage- und Betriebsanleitung des Greifmoduls.

2.1 Konfiguration des Greifmoduls

Zum Betrieb an einem Mitsubishi-Roboter muss am Greifmodul über die Web-Oberfläche im Menüpunkt „Settings -> Command Interface“ (vgl. Abbildung 1) die Schnittstelle „TCP/IP“ ausgewählt werden (Standardeinstellung). Darüber hinaus muss unter dem selben Menüpunkt die Verwendung des textbasierten Kommunikationsprotokolls „Gripper Control Language“ (GCL) aktiviert werden (Einstellung „Use text based Interface“, vgl. Abbildung 2).

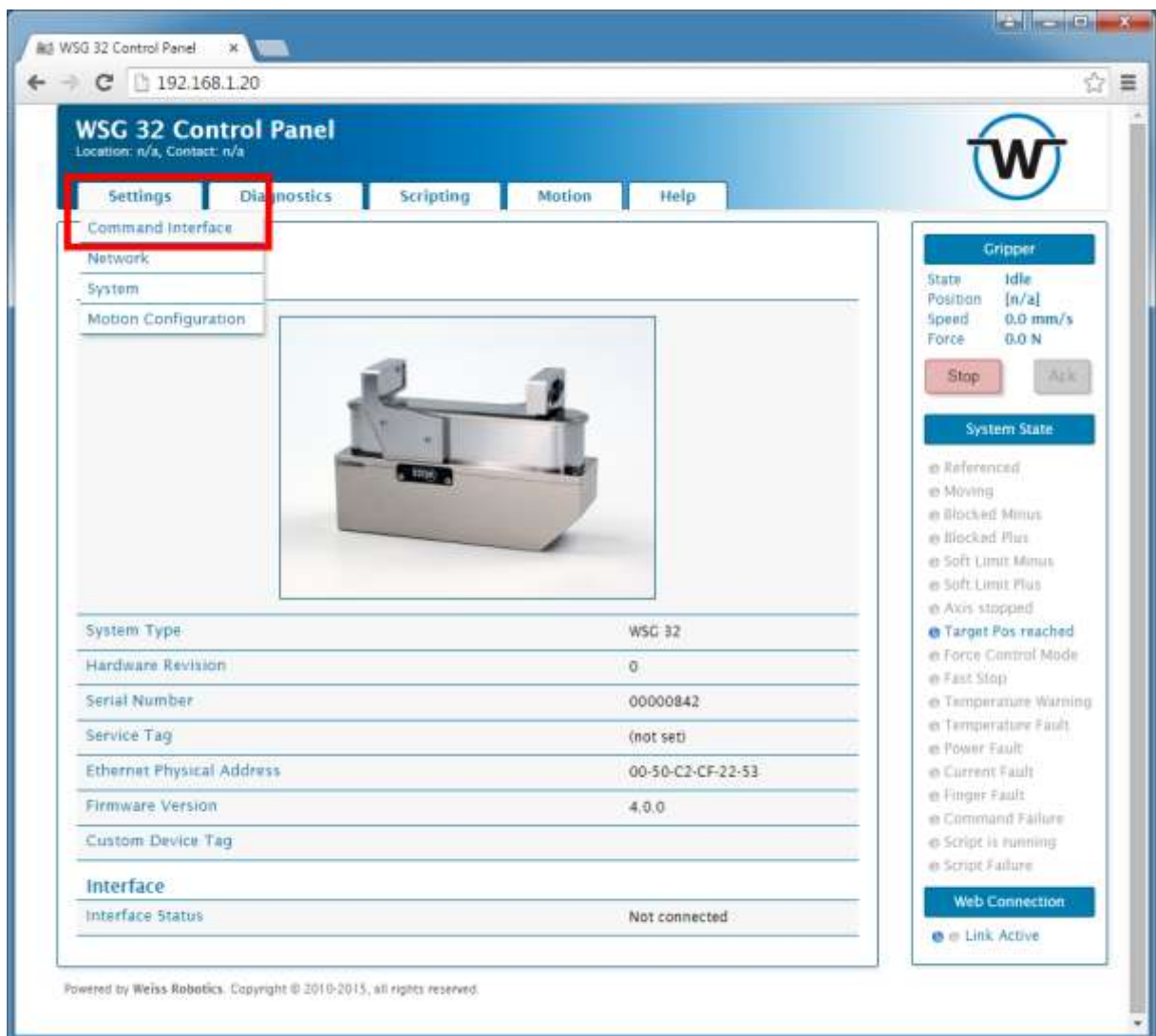


Abbildung 1: Menüpunkt zum Konfigurieren der Befehlsschnittstelle auf der Web-Oberfläche

Optional können über die Web-Oberfläche auch weitere Einstellungen, beispielsweise zur IP-Adresse, vorgenommen werden. Das vorliegende Beispiel bezieht sich jedoch auf die Standardeinstellungen. Weitere Informationen entnehmen Sie bitte der Montage- und Betriebsanleitung des Greifmoduls.

i Im Auslieferungszustand verfügt das Greifmodul über die IP-Adresse 192.168.1.20 und die Subnetzmaske 255.255.255.0.

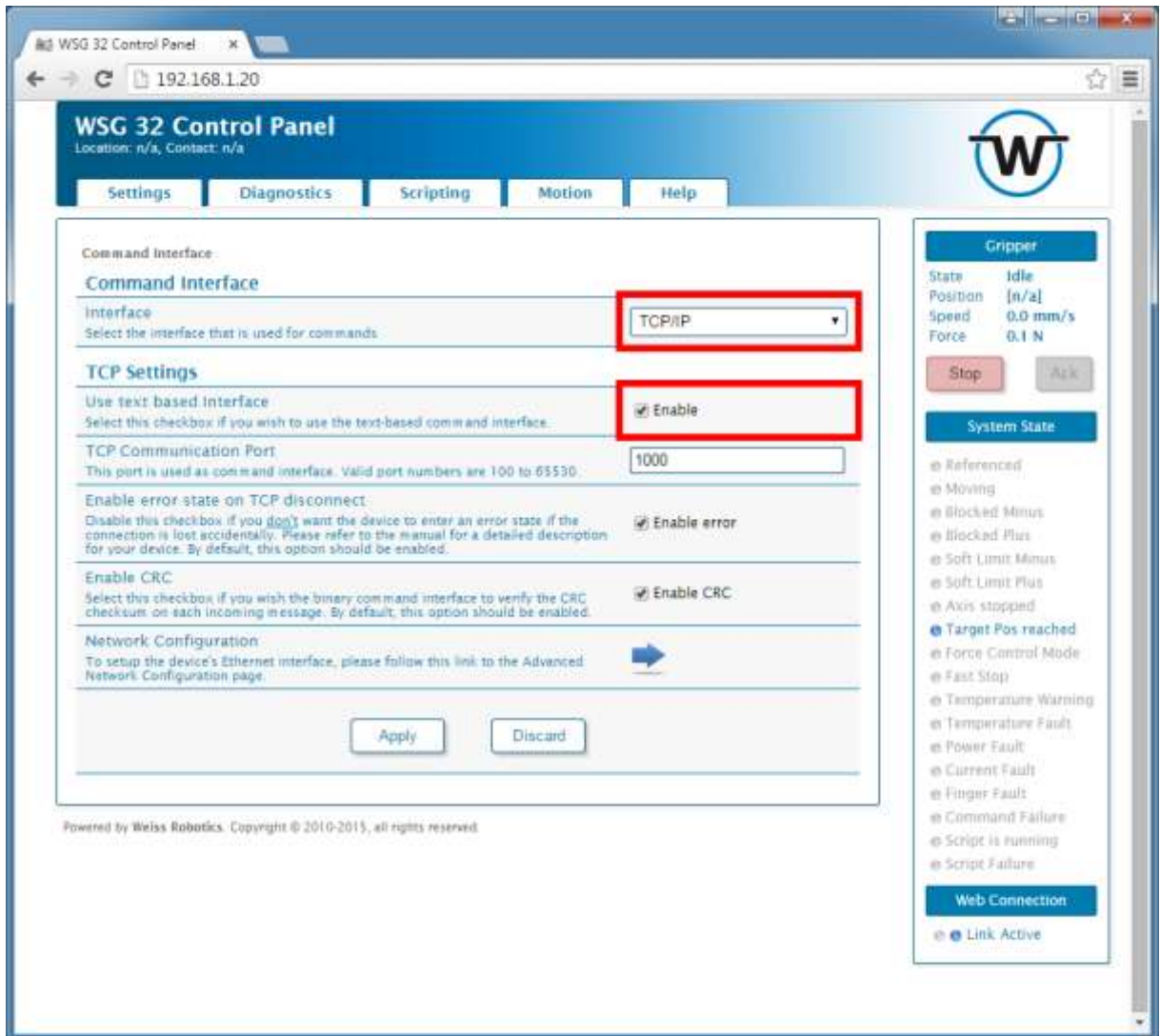


Abbildung 2: Schnittstellenkonfiguration auf der Web-Oberfläche des Greifmoduls

2.2 Konfiguration des Roboters

Es wird vorausgesetzt, dass der Roboter, wie in der Betriebsanleitung beschrieben, fertig eingerichtet und in Betrieb genommen wurde.

Für die Verbindung mit dem Greifmodul müssen zunächst die Parameter für die Netzwerkeinstellungen der Robotersteuerung angepasst werden. Anschließend muss eine Kommunikationsverbindung zum Greifmodul konfiguriert werden.

2.2.1 Netzwerkeinstellungen der Roboterschnittstelle

Die Netzwerkeinstellungen der Robotersteuerung können über das Handbedienpanel modifiziert werden. Wählen Sie dazu im Hauptmenü zunächst den Punkt „Parameter“, dann den Button „Parameter menu“ und schließlich „Communication Parameter“ -> „Ethernet“ (vgl. Abbildung 3, Abbildung 4 und Abbildung 5).



Abbildung 3: Hauptmenü des Handbedienpanels

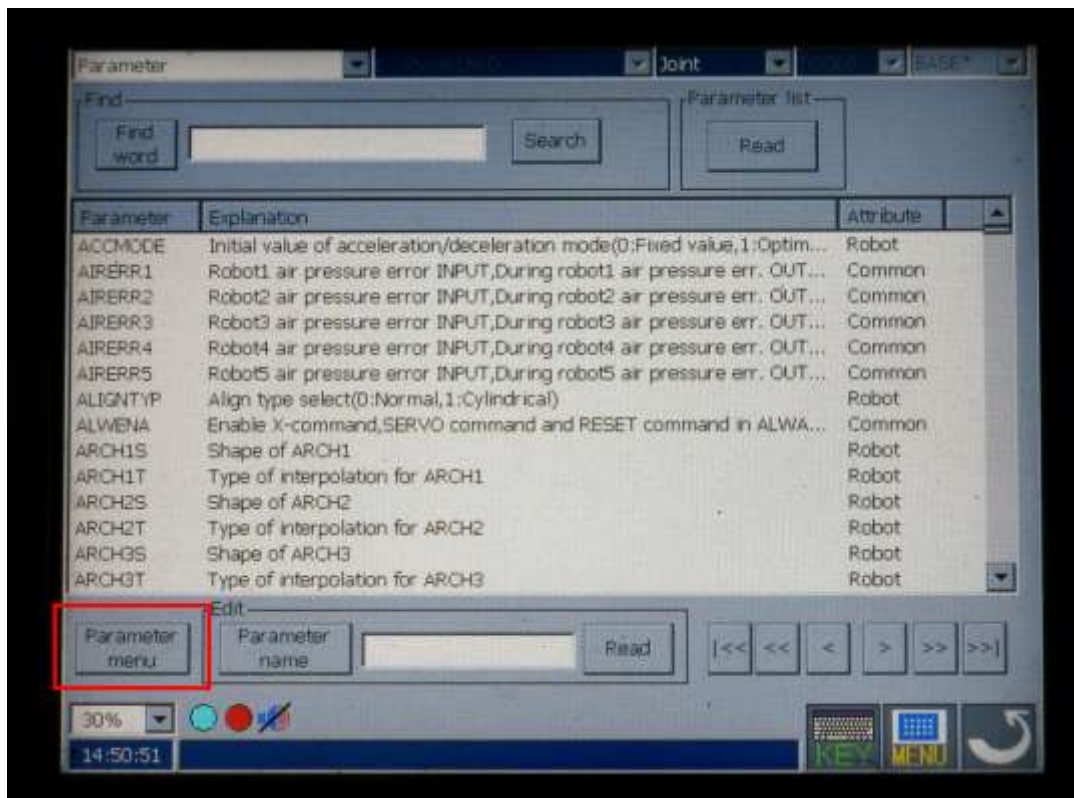


Abbildung 4: Übersicht über die Konfigurationsparameter

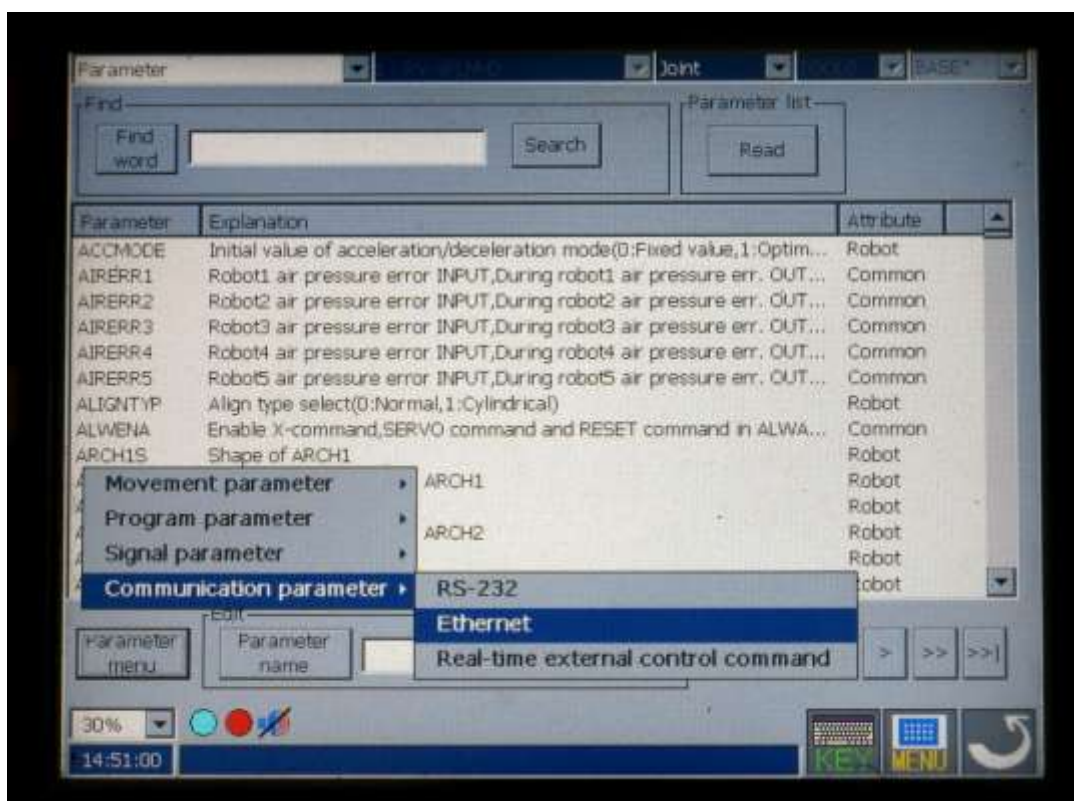


Abbildung 5: Kommunikationsparameter der Ethernet-Schnittstelle

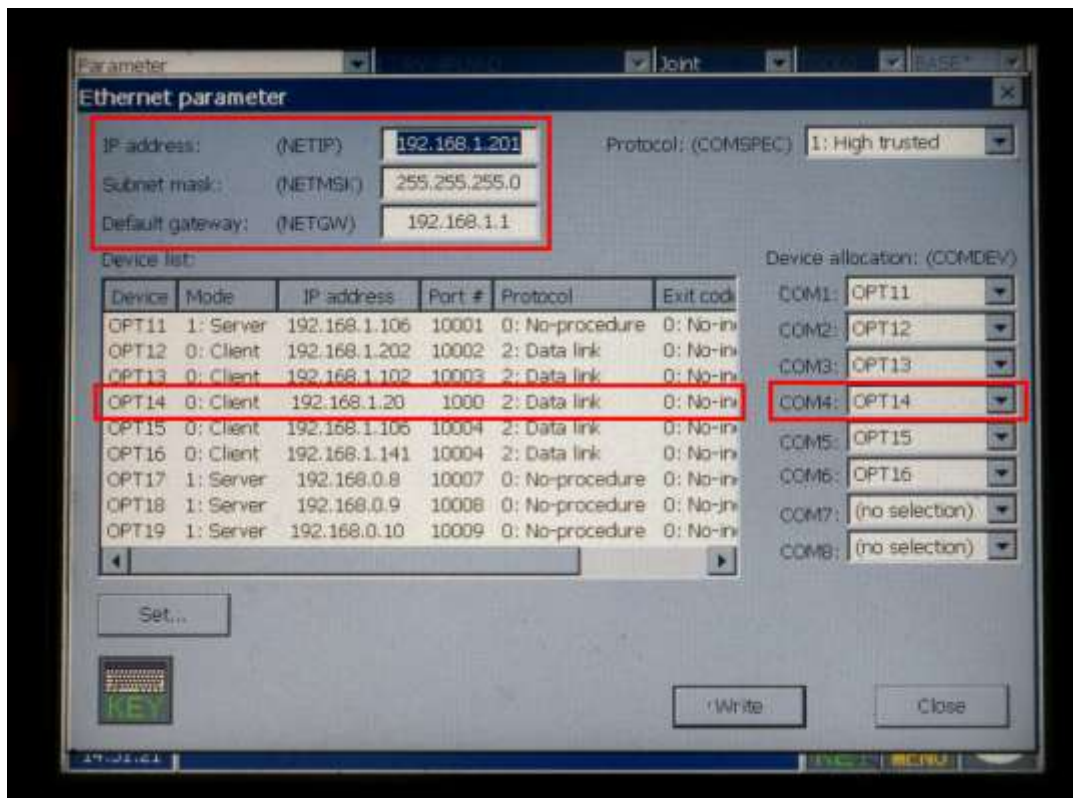


Abbildung 6: Übersicht über die Netzwerkeinstellungen

Stellen Sie unter „IP Address“ (Parameter NETIP) als IP-Adresse für den Roboter den Wert 192.168.1.201 ein. Die „Subnet Mask“ (Parameter NETMSK) muss auf 255.255.255.0 eingestellt werden. Die Einstellung für den Parameter „Default Gateway“ ist unerheblich, sofern die Robotersteuerung nicht über einen Router mit einem entfernten Netzwerk oder dem Internet verbunden werden soll und wurde in diesem Fall auf 192.168.1.1 gesetzt (vgl. Abbildung 6). Im Zweifel wenden Sie sich bitte an Ihren Netzwerkadministrator.

Darüber hinaus muss ein Device für die Verbindung zum Greifmodul konfiguriert werden. Wählen Sie dazu aus der Liste das Device *OPT14* und weisen ihm den Mode „Client“, die IP-Adresse 192.168.1.20 (Standard-Adresse des Greifmoduls), Portnummer 1000, das Protocol „Data Link“ und den Packet Type „CR+LF“ zu. Ordnen Sie *OPT14* dem Device *COM4* zu.

- ❗ Selbstverständlich können auch andere, vom Nutzer frei wählbare Netzwerkadressen verwendet werden. Beachten Sie jedoch, dass das Beispielprogramm das Device *COM4* für den Zugriff auf das Greifmodul verwendet. Bei Verwendung eines anderen Device (z. B. *COM1*) muss der Quelltext des Beispielprogramms entsprechend geändert werden.
- ❗ Sofern die IP-Adresse des Greifmoduls modifiziert wurde, müssen die IP-Adressen auf der Robotersteuerung entsprechend angepasst werden.

3 Inbetriebnahme

Öffnen Sie das Projekt *WSGExample* in der Programmierumgebung *RT ToolBox2*. Entpacken Sie dazu zunächst das beiliegende ZIP-Archiv an einen Ort Ihrer Wahl. Starten Sie anschließend die *RT ToolBox2*. Wählen Sie im Menü „Workspace“ -> „Open“ und wählen das Unterverzeichnis „WSGExample“ im Zielordner, in den Sie zuvor das ZIP-Archiv entpackt haben.

Stellen Sie durch Klick auf den Button „Online“ eine Verbindung zum Roboter her und kopieren die Roboterprogramme aus dem Abschnitt „Offline“ -> „Program“ mit Hilfe des *Program Manager* (zu erreichen via Rechtsklick auf „Program“, siehe Abbildung 9) auf den Roboter.

Laden und Starten Sie nun auf dem Roboter das Programm „GripCycle“ mit Hilfe des „Operation panel“ (Abbildung 10).

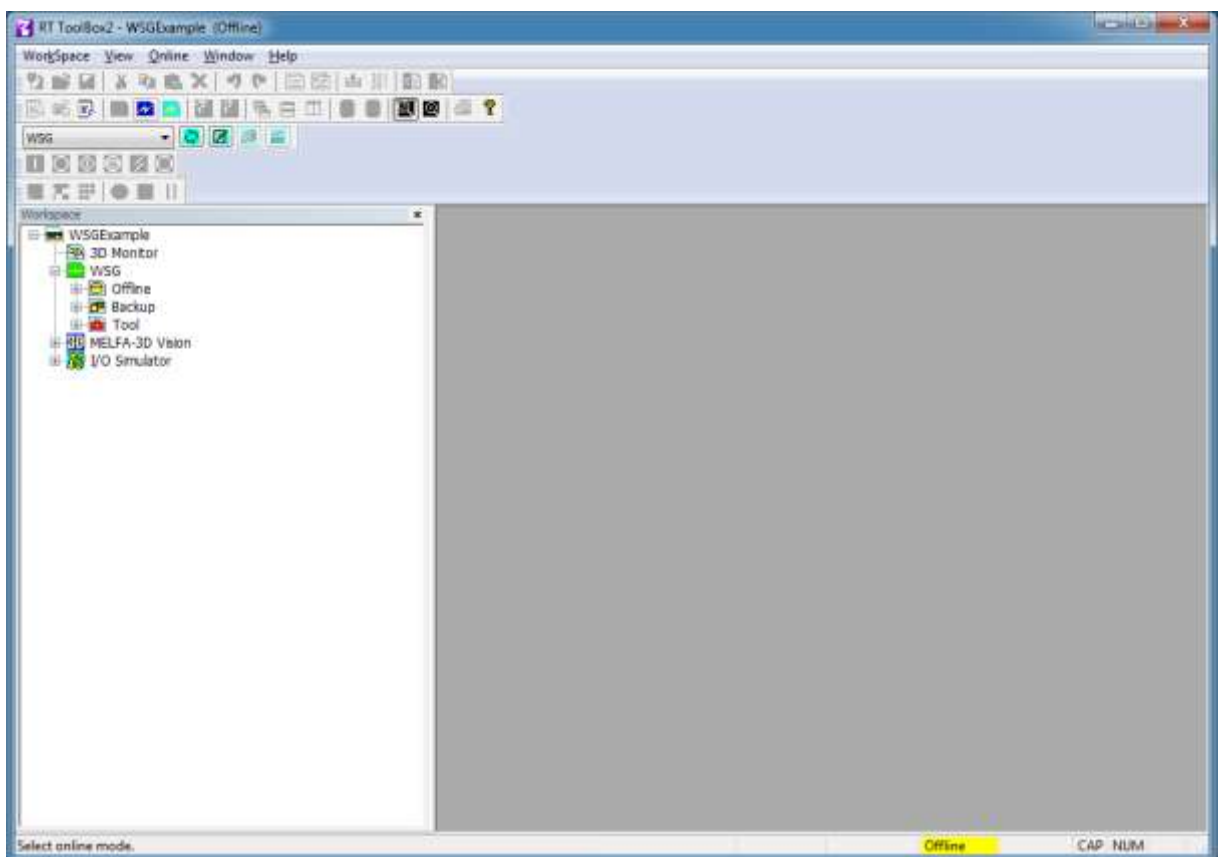


Abbildung 7: Workspace „WSGExample“ in der Programmierumgebung *RT ToolBox2*

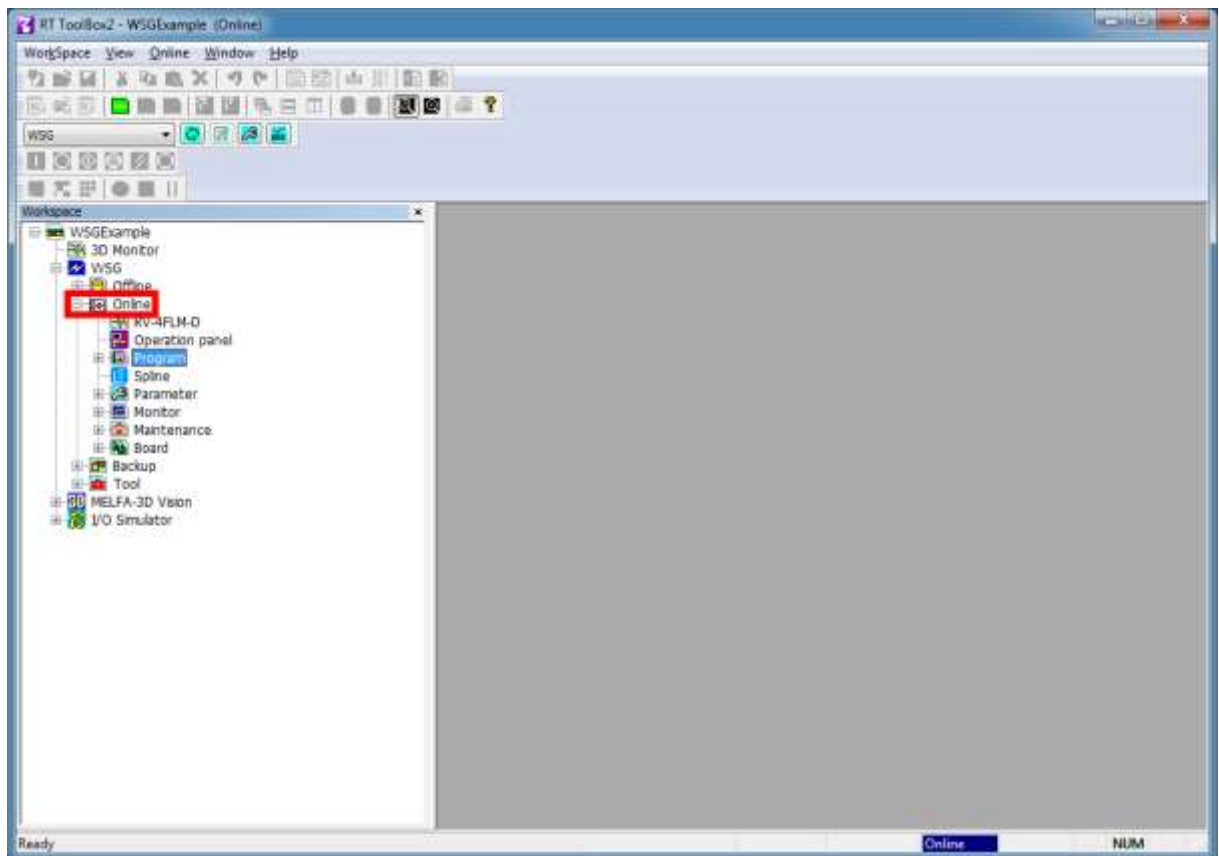


Abbildung 8: Workspace „WSGExample“ mit Verbindung zum Roboter

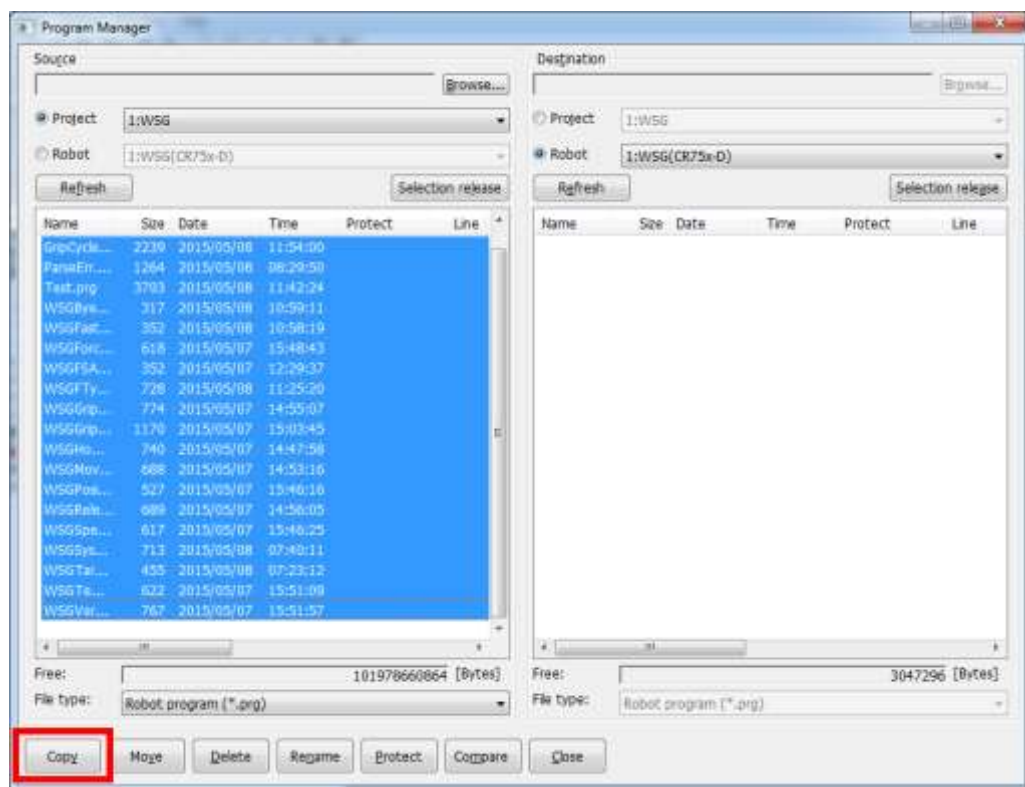


Abbildung 9: RT ToolBox2 Program Manager



Abbildung 10: „Operation panel“

4 Beschreibung der Roboterprogramme

4.1 Hauptprogramm „GripCycle“

Das Programm „GripCycle“ implementiert einen einfachen Greifzyklus aus Vorpositionieren, Greifen, Loslassen. Es verbindet sich zunächst mit dem Greifmodul unter Verwendung der eingestellten Verbindungsparametern und fragt den Typ des angeschlossenen Greifmoduls ab. Sofern das Greifmodul über eine Schnittstelle für aktive Sensorfinger („Sensor Port“) verfügt, so wird abgefragt, ob Sensorfinger des Typs „WSG-FMF“ angeschlossen sind. Wenn dies der Fall ist, werden die Sensorfinger tar-
tiert.

Nach Überprüfung des Greifzustands („Gripper State“) wird eine Referenzierungsfahrt („Homing“) an den äußeren mechanischen Anschlag durchgeführt. Ist die Referenzierungsfahrt erfolgreich abgeschlossen, wird ein einfacher Greifzyklus gefahren: Vorpositionieren auf 40 mm, Greifen auf 20 mm bei einer Greifkraft von 20 N, Loslassen/Öffnen der Finger relativ zur aktuellen Fingerposition um 20 mm. Für alle Bewegungsbefehle wird eine Geschwindigkeit von 100 mm/s vorgegeben.

Nach Beendigung des Greifzyklus wird die Verbindung zum Greifmodul wieder geschlossen.

Sofern sich die Robotersteuerung im Automatikmodus befindet, startet das Programm wieder von vorn.

4.2 Hauptprogramm „Test“

Dieses Programm führt alle Unterprogramme zur Kommunikation mit dem Greifmodul einmal testweise aus und überprüft die zurückgelieferten Parameter. Details zum Ablauf können dem kommentierten Quelltext entnommen werden.

4.3 Unterprogramme zur Abstraktion der GCL-Befehle

Die in diesem Abschnitt beschriebenen Unterprogramme bilden die wichtigsten Steuerungsbefehle der „Gripper Control Language“ (GCL) auf die Robotersteuerung ab. Zusammengenommen bilden sie eine einfach zu verwendende Schnittstelle, über die das Greifmodul von der Robotersteuerung aus gesteuert werden kann.

Weitere Informationen zu Syntax und Semantik der GCL-Befehle befinden sich im Dokument „WSG GCL Reference Manual“ (engl.), welches im Lieferumfang des Greifmoduls enthalten ist.



Die in diesem Abschnitt beschriebenen Unterprogramme stellen ausschließlich Funktionen zur transparenten Kommunikation mit dem Greifmodul bereit. Dies bedeutet insbesondere, dass vom Greifmodul zurück gelieferte Fehlerzustände nicht automatisch erkannt und verarbeitet, sondern in entsprechenden Variablen zur Abfrage bereit gestellt werden. Die Erkennung, Verarbeitung und Behebung dieser Fehlerzustände (z. B. durch Abfrage dieser Variablen) obliegt dem Nutzer.

4.3.1 WSGBye

Dieses Unterprogramm bildet den GCL-Befehl *BYE* ab und dient dazu, dem Greifmodul, eine bevorstehende Trennung der Kommunikationsverbindung anzukündigen. Dieses Unterprogramm muss vor dem Abbau der Verbindung gesendet werden, um zu verhindern, dass das im Greifmodul ein quittierungspflichtiger Fast Stop ausgelöst wird.

Parameter

keine

Rückgabewert

keiner

4.3.2 WSGVerbose

Dieses Unterprogramm bildet den GCL-Befehl *VERBOSE* ab und dient dazu, die erweiterten mit dem übergebenen Eingabeparameter an das Greifmodul. Dieser Befehl

Parameter

MEnable% *Gibt an, ob der Verbose-Modus eingeschaltet (Wert 1) oder ausgeschaltet (Wert 0) werden soll*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.3 WSGDevType

Dieses Unterprogramm bildet den GCL-Befehl *DEVTYPE* ab und dient dazu, den Typ des angeschlossenen WSG-Greifmoduls abzufragen. Der Befehl kann verwendet werden, um während des Betriebs zwischen unterschiedlichen Greifmodulen der WSG-Serie zu unterscheiden.

Parameter

Keine

Rückgabewert

Der abgefragte Typstring wird in der globalen Variablen *C_DevTypeStr* gespeichert. Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *C_DevType* ist undefiniert.

4.3.4 WSGHome

Dieses Unterprogramm bildet den GCL-Befehl *HOME* ab und dient dazu, die Position der Greiffinger bei Betriebsbeginn zu referenzieren. Eine Referenzierungsfahrt sollte immer in die Richtung durchgeführt werden, in die später auch gegriffen werden soll. Dies gewährleistet die höchstmögliche Genauigkeit bei der Positionierung der Greiffinger.

Parameter

MDirection% *Gibt an, in welche Richtung die Referenzierungsfahrt durchgeführt werden soll.
0: Referenzfahrt in negative Richtung, 1: Referenzfahrt in positive Richtung*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.5 WSGMove

Dieses Unterprogramm bildet den GCL-Befehl *MOVE* ab und dient dazu, die Greiffinger vor dem Greifen eines Teils vorzupositionieren. Das Unterprogramm darf ausschließlich zum freien Positionieren der Greiffinger verwendet werden. Das Greifen und Loslassen von Teilen muss immer durch Aufruf der GCL-Befehle *GRIP* bzw. *RELEASE* erfolgen (vgl. Unterprogramm WSGGrip, siehe Kapitel 4.3.6 sowie WSGRelease, siehe Kapitel 4.3.7).

Parameter

MWidth *Zielposition (Öffnungsweite) in mm*
MSpeed *Verfahrgeschwindigkeit in mm/s*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.6 WSGGrip

Dieses Unterprogramm bildet den GCL-Befehl *GRIP* ab und dient dazu, ein Greifteil der vorgegebenen Größe mit der vorgegebenen Kraft und Geschwindigkeit zu greifen.

Es ist zu beachten, dass der Befehl einen Fehler liefert (Statuscode 18 – *E_CMD_FAILED*), wenn kein Greifteil gefunden wird.

Parameter

MForce *Greifkraft in N*
MWidth *Nominale Größe des zu greifenden Teils in mm*
MSpeed *Greifgeschwindigkeit in mm/s*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.7 WSGRelease

Dieses Unterprogramm bildet den GCL-Befehl *RELEASE* ab und dient dazu, ein Greifteil nach dem Greifen durch Öffnen der Greiffinger wieder loszulassen.

Parameter

MDist *Öffnungsdistanz relativ zur aktuellen Position in mm*
MSpeed *Öffnungsgeschwindigkeit in mm*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.8 WSGSysFlag

Dieses Unterprogramm bildet eine Variante des GCL-Befehls *SYSFLAGS* ab und dient dazu, das Systemzustandsflag mit dem angegebenen Index abzufragen. Eine Übersicht über die zur Verfügung stehenden Systemzustandsflags und ihre Bedeutung befindet sich im Anhang des Dokumentes „WSG GCL Reference Manual“ (engl.), welches im Lieferumfang des Greifmoduls enthalten ist.

Parameter

MIndex% *Index des abzufragenden Systemzustandsflags*

Rückgabewert

Der Zustand des abgefragten Systemzustandsflags wird in der globalen Variablen *M_SysFlag* gespeichert. Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *M_SysFlag* auf 0 gesetzt.

4.3.9 WSGGripState

Dieses Unterprogramm bildet den GCL-Befehl *GRIPSTATE* ab und dient dazu, den Greifzustand („Gripper State“) abzufragen. Eine Übersicht über die möglichen Werte und ihre Bedeutung befindet sich im Anhang des Dokumentes „WSG GCL Reference Manual“ (engl.), welches im Lieferumfang des Greifmoduls enthalten ist.

Parameter

Keine

Rückgabewert

Der abgefragte Greifzustand wird in der globalen Variablen *M_GripStateVal* als numerischer Wert abgelegt sowie in der globalen Variablen *C_GripStateStr* als Zeichenkette (String) gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.10 WSGPosition

Dieses Unterprogramm bildet den GCL-Befehl *POSITION* ab und dient dazu, die aktuelle Öffnungsweite der Greiffinger in mm abzufragen.

Parameter

Keine

Rückgabewert

Die abgefragte Öffnungsposition wird in der globalen Variablen *M_FingerPos* gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *M_FingerPos* auf 0 gesetzt.

4.3.11 WSGSpeed

Dieses Unterprogramm bildet den GCL-Befehl *SPEED* ab und dient dazu, die aktuelle Geschwindigkeit der Greiffinger in mm/s abzufragen.

Parameter

Keine

Rückgabewert

Der abgefragte Geschwindigkeitswert wird in der globalen Variablen *M_FingerSpeed* gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *M_FingerSpeed* auf 0 gesetzt.

4.3.12 WSGForce

Dieses Unterprogramm bildet den GCL-Befehl *FORCE* ab und dient dazu, die aktuelle Greifkraft in N abzufragen.

Parameter

Keine

Rückgabewert

Der abgefragte Kraftwert wird in der globalen Variablen *M_FingerForce* gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *M_FingerForce* auf 0 gesetzt.

4.3.13 WSGTemp

Dieses Unterprogramm bildet den GCL-Befehl *TEMP* ab und dient dazu, den aktuellen Wert des Temperatursensors auf der Steuerungsplatine im Inneren des Greifmoduls in °C abzufragen.

Parameter

Keine

Rückgabewert

Der abgefragte Temperaturwert wird in der globalen Variablen *M_Temperature* gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *M_Temperature* auf 0 gesetzt.

4.3.14 WSGFastStop

Dieses Unterprogramm bildet den GCL-Befehl *FASTSTOP* ab und dient dazu, auf dem Greifmodul einen quittierungspflichtigen Fehlerzustand auszulösen.

Parameter

Keine

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.15 WSGFSack

Dieses Unterprogramm bildet den GCL-Befehl *FSACK* ab und dient dazu, einen vom Greifmodul selbst oder über das Unterprogramm WSGFastStop (siehe Kapitel) ausgelösten Fehlerzustand zu quittieren.

Parameter

Keine

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.3.16 WSGFType

Dieses Unterprogramm bildet den GCL-Befehl *FTYPE* ab und dient dazu, die Art der auf dem Greifmodul montierten, optionalen Sensorfinger zu bestimmen. Insbesondere kann der Befehl verwendet werden, um die Montage von Kraftmessfingern (WSG-FMF) zu erkennen.

 **Dieser Befehl ist nicht auf allen Greifmodulen der WSG-Serie verfügbar, da nicht alle Module über eine Sensorschnittstelle („Sensor Port“) zum Anschluss von Sensorfingern verfügen.**

Parameter

MIndex% *Finger-Index (0 oder 1).*

Rückgabewert

Der abgefragte Wert wird in der globalen Variablen *C_FingerTypeStr* gespeichert.

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt und der Wert von *C_FingerTypeStr* ist undefiniert.

4.3.17 WSGTare

Dieses Unterprogramm bildet den GCL-Befehl *TARE* ab und dient dazu, optional erhältliche Kraftmessfinger des Typs WSG-FMF zu tarieren.

 **Dieser Befehl ist nicht auf allen Greifmodulen der WSG-Serie verfügbar, da nicht alle Module über eine Sensorschnittstelle („Sensor Port“) zum Anschluss von Sensorfingern verfügen.**

Parameter

MIndex% *Finger-Index (0 zum Trieren von Finger 1, 1 zum Trieren von Finger 2).*

Rückgabewert

Im Fehlerfall wird der vom Greifmodul zurück gelieferte Statuscode in der globalen Variable *M_StatusCode* abgelegt.

4.4 Hilfsprogramme

4.4.1 ParseErr

Dieses Unterprogramm wird von den WSG-Unterprogrammen aufgerufen, um Fehlermeldungen zu parsen und den vom Greifmodul zurück gelieferten Statuscode zu bestimmen.



www.weiss-robotics.com

© Weiss Robotics GmbH & Co. KG. Alle Rechte vorbehalten.

Die in diesem Dokument angegebenen technischen Daten können zum Zwecke der Produktverbesserung ohne Vorankündigung geändert werden. Warenzeichen sind Eigentum des jeweiligen Eigentümers. Unsere Produkte sind nicht für den Einsatz in lebenserhaltenden Systemen oder für Systeme, bei denen ein Fehlverhalten zu Personenschäden führen könnte, vorgesehen.