

DENSO ROBOTER

Vertikal beweglich

V*-D-SERIE

EINFÜHRUNGS-HANDBUCH

Copyright © DENSO, 2000

Alle Rechte vorbehalten. Kein Teil dieses Handbuches darf in irgendeiner Form oder auf irgendeine Weise ohne die schriftliche Genehmigung des Herausgebers reproduziert werden.

Änderungen der Spezifikationen ohne vorherige Ankündigung vorbehalten.

Alle erwähnten Produkt- und Firmennamen sind Warenzeichen oder eingetragene Warenzeichen ihrer entsprechenden Inhaber.

Vorwort

Vielen Dank für den Kauf dieses Hochgeschwindigkeits- und Hochpräzisions-Montageroboters.

Bevor Sie den Roboter in Betrieb nehmen, lesen Sie bitte sorgfältig dieses Handbuch durch, damit Sie aus dem Roboter den größtmöglichen Nutzen für Ihre Montagearbeiten erhalten.

Folgende Roboterserien und/oder -modelle werden in diesem Handbuch behandelt:

Vertikal beweglich, V*-D-SERIE

Wichtig!

Zur Gewährleistung der Sicherheit muß das Bedienpersonal mit den Vorsichtsmaßnahmen und Anleitungen im Abschnitt "SICHERHEITS- UND VORSICHTSMASSNAHMEN" vertraut sein.

V*-D ist wie folgt aufgebaut

Der V*-D-Dokumentationssatz besteht aus folgenden sechs Handbüchern. Wenn Sie nicht mit dieser Roboter-Serie vertraut sind, arbeiten Sie bitte alle sechs Handbücher genau durch, bevor Sie den Roboter in Betrieb nehmen.

EINFÜHRUNGS-HANDBUCH - vorliegendes Handbuch:

Stellt Ihnen DENSO Roboter vor. Anhand eines Gerätekonfigurationsbeispiels erfahren Sie in diesem Handbuch, wie ein Roboter mit dem Programmiergerät gesteuert werden kann. Dazu wird ein Programm in WINCAPSII erstellt, um den Roboter automatisch zu steuern.

INSTALLATIONS- UND WARTUNGSHANDBUCH

Beschreibt den Roboteraufbau, gibt die Anleitungen für die Installation der Roboterelemente sowie die Wartungs- und Prüfungsarbeiten.

EINSTELLUNGSCHANDBUCH

Beschreibt das Einstellen oder Programmieren des Roboters mit Hilfe des Programmiergeräts oder des Bedienfeldes.

WINCAPSII HANDBUCH (ausgeliefert mit WINCAPSII)

Enthält Anleitungen zum Einsatz des auf dem PC installierten und mit dem Roboter und dem Roboter-Controller verbundenen Programmiersystems für die Programmentwicklung und -verwaltung.

PROGRAMMIERHANDBUCH

Beschreibt die Programmiersprache PAC, die Schritte der Programmentwicklung in PAC und Befehlsspezifikationen.

TABELLEN DER FEHLERCODES

Hier finden Sie Fehlercodes, die auf dem Programmiergerät, Bedienfeld oder PC-Bildschirm angezeigt werden, wenn ein Fehler innerhalb der Roboter-Serie oder von WINCAPSII auftritt. Diese Tabellen enthalten detaillierte Beschreibungen von Fehlern und Anleitungen zu deren Beseitigung.

Das vorliegende Handbuch ist wie folgt aufgebaut:

Bei diesem Handbuch handelt es sich nur um einen Teil des V*-D-Dokumentationssatzes. Es besteht aus den SICHERHEITS- UND VORSICHTSMASSNAHMEN, Kapitel 1-5 und den Anhängen.

SICHERHEITS- UND VORSICHTSMASSNAHMEN

Erläutert sicherheitsrelevante Begriffe und Symbole und enthält zu beachtende Vorsichtsmaßnahmen. Bitte lesen Sie diesen Abschnitt unbedingt, bevor Sie anfangen, mit dem Roboter zu arbeiten.

Kapitel 1 Steuerung des Roboters mit dem Programmiergerät

Beschreibt wie der Roboter mit dem Programmiergerät im manuellen Modus gesteuert werden kann, wie damit ein einfaches Programm erstellt und wie der Roboter programmiert werden kann.

Kapitel 2 Erstellen eines Programmes auf einem PC mit WINCAPSII

Enthält Anleitungen zum Setup von WINCAPSII auf einem PC, zum Erstellen und Kompilieren eines Programmes und zum Hochladen des kompilierten Programmes zum Roboter-Controller.

Ebenso wird die Maschinenverriegelung beschrieben, die für in Kapitel 3 auszuführende Simulationen erforderlich ist.

Kapitel 3 Simulieren einer Roboterbewegung entsprechend des Programmes auf einem PC

Beschreibt wie die programmierte Operation über einen Simulator am PC geprüft werden kann.

Kapitel 4 Steuern des Roboters über Programme

Enthält Prozeduren zum Steuern des Roboters über Programme und beschreibt mit der Palettierung eine der Hauptanwendungen von Robotern. Ebenso wird beschrieben, wie PAC-Bibliotheken eingesetzt werden können, um die Effizienz der Task-Programmentwicklung besonders zu verbessern.

Kapitel 5 Funktionen der Denso Roboter

Beschreibt Konformitätskontrollen und sonstige Funktionen der Denso Roboter.

Anhänge

Anhang-1 Glossar

Anhang-2 Bezeichnung der Teile des Roboter-Controllers

Anhang-3 Bezeichnung der Teile des Programmiergeräts

Anhang-4 Menü-Baumstruktur am Programmiergerät

SICHERHEITS- UND VORSICHTS- MASSNAHMEN

Bitte stellen Sie die Einhaltung der folgenden Sicherheits- und Vorsichtsmaßnahmen sicher.

Die strikte Einhaltung der Warn- und Vorsichtshinweise ist ein MUSS zur Verhütung von Unfällen, die zu körperlichen Schäden und erheblichem Sachschaden führen können. Stellen Sie sicher, daß Sie die unten aufgeführten Definitionen dieser Begriffe und die dazugehörigen, weiter unten dargestellten Symbole verstehen, bevor Sie mit dem Text fortfahren.

 WARNUNG	Warnung vor Situationen, die zu schweren körperlichen Schäden oder Tod führen können, wenn die Anleitungen nicht korrekt befolgt werden.
 VORSICHT	Warnung vor Situationen, die zu geringerem körperlichen Schaden oder erheblichem Sachschaden führen könnten, wenn die Anleitungen nicht korrekt befolgt werden.

Terminologie und Definitionen

Maximaler Roboterraum: Bezieht sich auf die Ausdehnung des Raums, der von den maximal möglichen Bewegungen aller Roboterbauteile, einschließlich Endwerkzeug, Werkstück und Anbaueinheiten, eingeschlossen wird. (Zitiert nach "RIA* Committee Draft".)

Sicherheitsraum: Bezieht sich auf den Teil des maximalen Roboterraums, auf den ein Roboter durch Begrenzungsvorrichtungen (z.B. mechanische Anschläge) eingeschränkt wird. Der maximale Weg, den der Roboter, das Endwerkzeug und das Werkstück nach dem Ansprechen der Begrenzungsvorrichtung zurücklegen können, definiert die Grenzen des Sicherheitsraums des Roboters. (Zitiert nach "RIA* Committee Draft".)

Bewegungsraum: Verweist auf einen bestimmten Sicherheitsraum, innerhalb dessen der Roboter durch Software-Bewegungsbegrenzungen gesteuert wird. Der maximale Weg, den der Roboter, das Endwerkzeug und das Werkstück nach dem Einstellen der Software-Bewegungsgrenzen zurücklegen können, definiert die Grenzen des Bewegungsraums des Roboters. (Der Begriff "Bewegungsraum" ist eine spezifische Terminologie von Denso.)

Arbeitsraum: Bezieht sich auf den Teil des Sicherheitsraums (oder Bewegungsraums in der Terminologie von Denso), der vom Roboter während des Ausführens seines Aufgabenprogramms tatsächlich genutzt wird. (Zitiert nach "RIA* Committee Draft".)

Aufgabenprogramm: Bezieht sich auf die Instruktionen für die Bewegung und die Hilfsfunktionen, die die jeweilige Aufgabe des Robotersystems definieren. (Zitiert nach "RIA* Committee Draft".)

(*RIA: Robotic Industries Association/Vereinigung der Roboterindustrie)

1. Einführung

In diesem Abschnitt sind die Sicherheits- und Vorsichtsmaßnahmen beschrieben, die während der Installation, des Programmierens, der Prüfung, der Anpassung und der Wartung des Roboters einzuhalten sind.

2. Vorsichtsmaßnahmen während der Installation

2.1 Sicherstellen der geeigneten Installationsumgebung

2.1.1 Standardausführung

Die Standardausführung ist nicht explosionsgeschützt, staubdicht oder spritzwassergeschützt ausgelegt. Demzufolge darf sie nicht in Umgebungen mit:

- (1) entzündlichen Gasen oder Flüssigkeiten,
- (2) herumfliegenden Spänen der Metallbearbeitung oder sonstigem elektrisch leitenden Material,
- (3) säurehaltigen, alkalihaltigen oder sonstigen korrosiven Gasen,
- (4) Ölnebeln vom Schneiden oder Schleifen,
- (5) der Gefahr des Eintauchens in Flüssigkeiten,
- (6) schwefelhaltigen Nebeln von Schneidöl oder Schleiföl oder
- (7) großen Umrichtern, Transmittern hoher Ausgangsleistung und hoher Frequenz, großen Schützen, Schweißgeräten oder sonstigen Quellen elektrischer Störungen installiert werden.

2.1.2 Staubdichte, spritzwassergeschützte Ausführung

Die staubdichte, spritzwassergeschützte Ausführung ist eine dem Schutzgrad IP54 entsprechende staubdichte und spritzwassergeschützte Konstruktion, die jedoch nicht explosionsgeschützt ausgelegt ist. (Beim Handgelenk des VM-D-W handelt es sich um eine IP65-entsprechende, staubdichte und spritzwassergeschützte Roboterkomponente.)

Beachten Sie, daß der Roboter-Controller keine staubdichte oder spritzwassergeschützte Roboterkomponente ist. Aus diesem Grund ist der Roboter-Controller in einer durch Ölnebel belasteten Umgebung in einem optionalen Schutzgehäuse unterzubringen.

Die staubdichte, spritzwassergeschützte Ausführung darf nicht in einer Umgebung mit:

- (1) entzündlichen Gasen oder Flüssigkeiten,
- (2) säurehaltigen, alkalihaltigen oder sonstigen korrosiven Gasen,
- (3) großen Umrichtern, Transmittern hoher Ausgangsleistung und hoher Frequenz, großen Schützen, Schweißgeräten oder sonstigen Quellen elektrischer Störungen,
- (4) der Gefahr des Eintauchens in Flüssigkeiten,
- (5) Partikeln oder Spänen vom Schleifen oder Zerspanen,
- (6) der Verwendung eines von DENSO nicht zugelassenen Öls oder

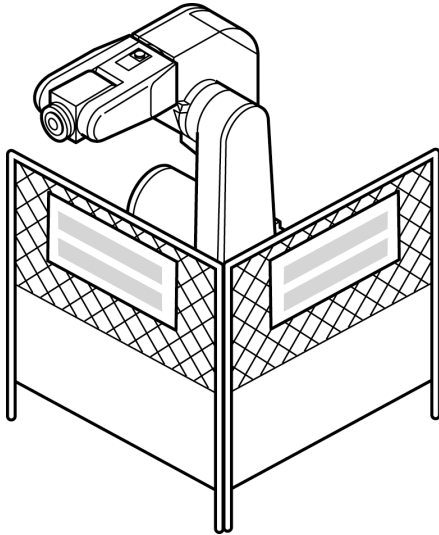
Anmerkung: DENSO-zugelassenes Öl: Yushiron-Öl Nr. 4C (unlöslich)

- (7) schwefelhaltigen Nebeln von Schneidöl oder Schleiföl.

SICHERHEITS- UND VORSICHTSMASSNAHMEN

- 2.2 Servicebereich** Der Roboter und die Peripheriegeräte sind so zu installieren, daß ein ausreichender Servicebereich für das sichere Programmieren, die Wartung und die Prüfung zu Verfügung steht.
- 2.3 Steuergeräte außerhalb des Sicherheitsraums des Roboters** Der Roboter-Controller, das Programmiergerät und das Bedienfeld sind außerhalb des Sicherheitsraums des Roboters und an einem Ort zu installieren, von dem aus beim Arbeiten mit dem Roboter-Controller, dem Programmiergerät oder dem Bedienfeld alle Bewegungen des Roboters überschaubar sind.
- 2.4 Anordnung von Meß- und Anzeigegegeräten** Druckmesser, Öldruckmesser und andere Anzeiger sind an einem leicht zu kontrollierenden Ort zu montieren.
- 2.5 Schutz der Elektroinstallation und der Hydraulik-/Druckluft-Rohrleitungen** Die Elektroinstallationen und Hydraulik-/Druckluft-Rohrleitungen sind durch eine Abdeckung oder eine entsprechende Maßnahme zu schützen, wenn die Möglichkeit einer Beschädigung besteht.
- 2.6 Anordnung der NOT-AUS-Schalter** Die NOT-AUS-Schalter sind an einer leicht erreichbaren Stelle anzubringen, um den Roboter ggf. unverzüglich anzuhalten.
- (1) Die NOT-AUS-Schalter müssen rot sein.
 - (2) Die NOT-AUS-Schalter müssen so konstruiert sein, daß sie nach dem Auslösen weder automatisch noch versehentlich durch eine Bedienperson freigegeben werden können.
 - (3) NOT-AUS-Schalter müssen vom Hauptschalter getrennt sein.
- 2.7 Anordnung der Betriebszustandsanzeigen** Die Betriebszustandsanzeigen sind so anzuordnen, daß das Bedienpersonal leicht erkennt, ob der Roboter vorübergehend, durch einen NOT-AUS oder durch eine Fehlfunktion stillsteht.

2.8 Aufstellen der Sicherheitsumzäunung oder -verkleidung



Eine Sicherheitsumzäunung oder -verkleidung ist aufzustellen, so daß niemand den Sicherheitsraum des Roboters betreten kann. Falls möglich, verwenden Sie weitere Schutzeinrichtungen wie in Abschnitt 2.9 beschrieben.

- (1) Die Umzäunung oder Verkleidung muß so aufgebaut sein, daß man sie nicht ohne weiteres bewegen oder entfernen kann.
- (2) Die Umzäunung oder Verkleidung muß so aufgebaut sein, daß man sie durch äußere Krafteinwirkung nicht ohne weiteres beschädigen oder verformen kann.
- (3) In der Umzäunung oder Verkleidung ist ein Ein-/Ausgang einzurichten. Errichten Sie die Umzäunung oder Verkleidung so, daß niemand sie ohne weiteres durch Klettern über die Umzäunung oder Verkleidung überwinden kann.
- (4) Die Umzäunung oder Verkleidung ist so zu errichten, daß man Hände oder andere Körperteile nicht hindurchstecken kann.
- (5) Ergreifen Sie eine der folgenden Schutzmaßnahmen für den Ein-/Ausgang der Umzäunung oder Verkleidung:
 - 1) Bringen Sie eine Tür, ein Seil oder eine Kette mitten im Ein-/Ausgang der Umzäunung oder Verkleidung an. Montieren Sie eine Sperre, die das automatische Ansprechen des NOT-AUS-Gerätes sicherstellt, wenn die Sperre geöffnet oder entfernt wird.
 - 2) Bringen Sie einen Warnhinweis am Ein-/Ausgang der Umzäunung oder Verkleidung mit der Aufschrift "In Betrieb! Zutritt verboten!" oder "Es wird gearbeitet! Nicht bedienen!" an und sorgen dafür, daß diese Anweisungen vom Personal zu jeder Zeit befolgt werden.

Beim Ausführen eines Testlaufs und vor dem Aufstellen einer Sicherheitsumzäunung oder -verkleidung sollte sich eine Aufsichtsperson neben dem Sicherheitsraum des Roboters positionieren und eine andere dort, wo er/sie alle Bewegungen des Roboters sehen kann. Die Aufsichtsperson soll verhindern, daß Personal den Sicherheitsraum des Roboters betritt, und soll sich ausschließlich dieser Aufgabe widmen.

2.9 Anbringen eines Seils oder einer Kette

Wenn es nicht möglich ist, die in Abschnitt 2.8 beschriebene Sicherheitsumzäunung oder -verkleidung aufzustellen, bringen Sie um den Sicherheitsraum des Roboters ein Seil oder eine Kette an, um sicherzustellen, daß niemand den Sicherheitsraum betritt.

- (1) Sorgen Sie dafür, daß die Stützpfeiler nicht leicht verschoben werden können.
- (2) Stellen Sie sicher, daß die Farbe oder das Material des Seils oder der Kette mühelos von der Umgebung zu unterscheiden ist.
- (3) Bringen Sie einen Warnhinweis an einer übersichtlichen Stelle mit der Aufschrift "In Betrieb! Zutritt verboten!" oder "Es wird gearbeitet! Nicht bedienen!" an und sorgen Sie dafür, daß diese Anweisungen vom Personal zu jeder Zeit befolgt werden.
- (4) Legen Sie den Ein-/Ausgang fest und befolgen Sie die Anweisungen (3) bis (5) von Abschnitt 2.8.

2.10 Einstellen des Bewegungsraums des Roboters

Das für den Betrieb des Roboters erforderliche Volumen wird als Arbeitsraum bezeichnet.

Wenn der Bewegungsraum des Roboters größer als der Arbeitsraum ist, wird empfohlen, einen kleineren Bewegungsraum einzustellen, um Störungen oder Unterbrechungen anderer Geräte durch den Roboter zu verhindern.

Siehe Kapitel 4 des Benutzerhandbuchs "INSTALLATION und WARTUNG".

2.11 Änderungen am Roboter sind nicht erlaubt

Verändern Sie niemals den Roboteraufbau, den Roboter-Controller, das Programmiergerät oder andere Geräte.

2.12 Reinigen der Werkzeuge

Wenn der Roboter mit Schweißpistolen, Farbspritzdüsen oder anderen Endwerkzeugen arbeitet, die gereinigt werden müssen, wird die automatische Durchführung des Reinigungsvorganges empfohlen.

2.13 Beleuchtung

Für den sicheren Roboterbetrieb muß ausreichende Beleuchtung vorhanden sein.

2.14 Schutz vor Gegenständen, die vom Endwerkzeug weggeschleudert werden

Bei Verletzungsgefahr des Personals durch vom Endwerkzeug fallengelassene oder weggeschleuderte Objekte sind die Abmessungen, das Gewicht, die Temperatur und die chemischen Eigenschaften der Objekte zu berücksichtigen und geeignete Sicherheitsvorkehrungen zu treffen, um die Sicherheit des Personals zu garantieren.

2.15 Anbringen des Warnschildes

Bringen Sie das mit dem Roboter mitgelieferte Warnschild am Ein-/Ausgang der Sicherheitsumzäunung oder an einer anderen gut sichtbaren Stelle an.



3. Vorsichtsmaßnahmen während der Roboter aktiv ist



Warnung

Das Berühren des Roboters während seines Betriebs kann zu schweren Verletzungen führen. Stellen Sie sicher, daß die folgenden Vorschriften eingehalten und die in Abschnitt 3.1 aufgezählten Vorsichtsmaßnahmen bei der Durchführung aller Arbeiten befolgt werden.



- 1) Betreten Sie den Sicherheitsraum des Roboters nicht, wenn der Roboter in Betrieb oder der Antriebsmotor eingeschaltet ist.
- 2) Als Vorsichtsmaßnahme gegen Fehlfunktionen sorgen Sie dafür, daß das NOT-AUS-Gerät so aktiviert ist, daß die Spannung des Roboterantriebs beim Betreten des Sicherheitsraums des Roboters abgeschaltet wird.
- 3) Wenn es während des Roboter-Betriebs erforderlich ist, den Sicherheitsraum zu Programmier- oder Wartungszwecken zu betreten, stellen Sie sicher, daß die in Abschnitt 3.3 "Gewährleisten der Sicherheit von Personen, die Arbeiten im Sicherheitsraum des Roboters ausführen" genannten Schritte befolgt werden.

3.1 Erstellen von Arbeitsrichtlinien und Sicherstellen der Einhaltung durch das Personal

Beim Betreten des Sicherheitsraums zu Programmier- oder Wartungszwecken müssen Arbeitsbestimmungen für folgende Punkte festgelegt sein bzw. von den Arbeitern eingehalten werden.

- (1) Notwendige Bedienungsabläufe für den Roboterbetrieb.
- (2) Robotergeschwindigkeit während des Lernvorganges.
- (3) Verwendete Signalisierungsverfahren, wenn die Arbeiten von mehr als einem Arbeiter durchgeführt werden.
- (4) Vorgehensweise des Personals im Störfall je nach Art der Störung.
- (5) Notwendige Schritte zur Überprüfung der Freigabe und der Sicherheit während des Störungszustandes, um den Roboter nach dem Anhalten der Roboterbewegung infolge der Betätigung des NOT-AUS-Gerätes neu zu starten.
- (6) Nachfolgende Schritte sind unabhängig von den vorher genannten Punkten notwendig, um Gefahren infolge unerwarteter Roboterbewegung oder Störung des Roboters zu verhindern.
 - 1) Anzeige der Steuertafel (siehe Abschnitt 3.2 auf der nächsten Seite)
 - 2) Gewährleistung der Sicherheit des Personals, das Arbeiten im Sicherheitsraum des Roboters durchführt (siehe Abschnitt 3.3 auf der nächsten Seite)
 - 3) Die vom Personal einzunehmende Position und Haltung

Die Position und Haltung, die dem Personal ermöglicht, den normalen Roboterbetrieb zu bestätigen und bei Auftreten einer Störung sofort Schutz zu suchen.
 - 4) Realisierung von Maßnahmen für den Lärmschutz
 - 5) Signalisierungsverfahren für Personal an angeschlossenen Geräten
 - 6) Arten der Störungen und wie diese zu unterscheiden sind

Stellen Sie bitte sicher, daß die Arbeitsbestimmungen dem jeweiligen Robotertyp, dem Installationsbereich und der Art der Arbeit entsprechen.

Holen Sie beim Erstellen der Arbeitsbestimmungen die Meinung der betreffenden Arbeiter, der Entwickler des Geräteherstellers und eines Beauftragten für Arbeitssicherheit ein.

3.2 Anzeige des Bedienfeldes

Erstellen Sie eine Anzeige, die auf dem Bedienfeld oder Programmiergerät erscheint, um darauf hinzuweisen, daß der Roboter in Betrieb ist und um zu verhindern, daß der EIN-Schalter oder Umschalter während des Betriebs durch andere Personen als durch den Bediener zufällig betätigt werden. Ergreifen Sie zusätzliche geeignete Maßnahmen wie das Verriegeln der Maschinenverkleidung.

3.3 Gewährleistung der Sicherheit des Personals bei der Durchführung von Arbeiten im Sicherheitsraum des Roboters

Bei der Durchführung von Arbeiten im Sicherheitsraum des Roboters sind folgende Maßnahmen einzuhalten, um sicherzustellen, daß der Roboterbetrieb bei einer Störung sofort angehalten werden kann.

- (1) Sorgen Sie dafür, daß sich eine Aufsichtsperson an einem Ort außerhalb des Sicherheitsraums des Roboters befindet, von dem diese Aufsichtsperson sämtliche Roboterbewegungen überblicken kann, und daß sie sich ausschließlich dieser Aufgabe widmet.
 - ① Bei einer Störung ist sofort ein Not-Aus-Gerät zu aktivieren.
 - ② Erlauben Sie niemandem außer dem zuständigen Arbeiter das Betreten des Sicherheitsraums des Roboters.
- (2) Stellen Sie sicher, daß ein Fachmann innerhalb des Sicherheitsraums des Roboters einen tragbaren NOT-AUS-Schalter (Schaltfläche "Roboterstop" auf dem Programmiergerät) bei sich hat, den er im Notfall sofort betätigen kann.

3.4 Prüfungen vor Beginn von Arbeiten wie z. B. das Programmieren

Vor dem Beginn von Arbeiten wie die Programmierung sind evtl. Reparaturen sofort nach dem Erkennen einer Störung und sonstige notwendige Maßnahmen durchzuführen.

- (1) Überprüfen Sie die Umhüllung oder Abdeckung der externen Verdrahtung oder der externen Geräte auf evtl. Beschädigung.
- (2) Überprüfen Sie die normale Funktionsweise des Roboters (außergewöhnlicher Lärm oder Vibrationen während des Betriebs).
- (3) Überprüfen Sie die Funktionstüchtigkeit der NOT-AUS-Gerätes.
- (4) Überprüfen Sie die Druckluft oder Öl führenden Leitungen auf Leckage.
- (5) Überprüfen Sie, ob sich im Sicherheitsraum des Roboters oder in seiner Nähe Hindernisse befinden.

3.5 Entlüftung

Vor der Demontage oder dem Auswechseln von Pneumatik-Bauteilen ist der Antriebszylinder zu entlüften.

3.6 Vorsichtsmaßnahmen für Testläufe

Während der Durchführung von Testläufen sollte sich das Personal nach Möglichkeit außerhalb des Sicherheitsraums des Roboters aufhalten.

3.7 Vorsichtsmaßnahmen für den Automatik-Betrieb

(1) Bei Inbetriebnahme

Vor der Inbetriebnahme des Roboters sind zuerst die nachstehenden Punkte zu überprüfen. Außerdem müssen die zu benutzenden Signale festgelegt, und die Signalisierung mit dem gesamten betroffenen Personal überprüft werden.

- 1) Überprüfen Sie, ob sich jemand im Sicherheitsraum des Roboters aufhält.
 - 2) Überprüfen Sie, ob das Programmiergerät und die Werkzeuge an ihren vorgesehenen Plätzen sind.
 - 3) Überprüfen Sie, ob Störmeldeleuchten am Roboter oder dazugehörigen Ausrüstungen leuchten.
- (2) Überprüfen Sie, ob die Anzeigelampe für automatischen Betrieb im Automatik-Betrieb leuchtet.
- (3) Handlungen im Störfall

Stoppen Sie den Betrieb des Roboters durch Betätigen des NOT-AUS-Gerätes, wenn eine Störung des Roboters oder der angeschlossenen Geräte auftritt und es notwendig ist, den Sicherheitsraums des Roboters zur Durchführung der Notfall-Wartung einzugeben. Ergreifen Sie alle notwendigen Maßnahmen wie z. B. das Anbringen eines Warnschildes am Start-Schalter, um auf die laufenden Arbeiten hinzuweisen, und zu verhindern, daß jemand Zugang zum Roboter bekommt.

3.8 Vorsichtsmaßnahmen bei Reparaturen

- (1) Reparaturen dürfen nicht außerhalb des dafür vorgesehenen Bereichs durchgeführt werden.
- (2) Unter keinen Umständen dürfen Sperrvorrichtungen entfernt werden.
- (3) Beim Öffnen der Abdeckung des Roboter-Controllers zum Auswechseln der Batterie oder aus anderen Gründen ist die Stromversorgung des Roboter-Controllers abzuschalten und das Stromkabel zu ziehen.
- (4) Verwenden Sie nur von DENSO zugelassene Ersatzwerkzeuge.

4. Tägliche und regelmäßige Prüfungen

- (1) Stellen Sie sicher, daß die täglichen und regelmäßigen Prüfungen durchgeführt werden. Überprüfen Sie immer vor Arbeitsbeginn, daß keine Probleme mit dem Roboter und der dazugehörigen Ausrüstungen vorliegen. Bei Problemen sind alle notwendigen Maßnahmen zu ergreifen, um diese zu beheben.
- (2) Die Durchführung regelmäßiger Prüfungen oder evtl. Reparaturen sind zu protokollieren. Die Protokolle sind mindestens 3 Jahre aufzubewahren.

5. Verwalten von Disketten

- (1) Die mit dem Roboter mitgelieferten Disketten mit den "Systemeinstellungen", auf denen spezielle, ausschließlich für Ihren Roboter vorbereitete Daten enthalten sind, müssen sorgfältig behandelt und aufbewahrt werden.
- (2) Nach beendetem Programmiervorgang oder etwaigen Änderungen sind die Programme und die Daten stets auf Disketten zu sichern.

Datensicherungen helfen Ihnen, die im Roboter-Controller gespeicherten Daten wiederherzustellen, wenn diese aufgrund einer leeren Backup-Batterie verlorengegangen sind.
- (3) Beschriften Sie alle für das Speichern von Aufgabenprogrammen verwendeten Disketten, um das Laden falscher Disketten im Roboter-Controller auszuschließen.
- (4) Bewahren Sie die Disketten an einem von Staub, Luftfeuchtigkeit und Magnetfeldern freien Ort auf. Diese Einflüsse könnten die Disketten oder die auf den Disketten gespeicherten Daten zerstören.

INHALT

Vorwort	i
V*-D ist wie folgt aufgebaut	ii
Das vorliegende Handbuch ist wie folgt aufgebaut:	iii
SICHERHEITS- UND VORSICHTSMASSNAHMEN	1
Kapitel 1 Steuern des Roboters über das Programmiergerät	1
Lektion 1 Steuern des Roboters im manuellen Modus	3
1.1 Grundfunktionen des Programmiergeräts	3
1.2 Manuelles Steuern des Roboters mit dem Programmiergerät	5
Lektion 2 Steuern des Roboters über ein einfaches Programm	14
2.1 Erstellen eines einfachen Programmes mit dem Programmiergerät	15
2.2 Programmierung	24
2.3 Programmierprüfung	29
2.4 Steuern des Roboters im Auto-Modus	34
Kapitel 2 Erstellen eines Programmes auf einem PC mit WINCAPSII	39
Lektion 3 Einstellung des Roboter-Controllers mit dem Programmiergerät	40
3.1 Ausführen der Kalibrierung (CAL)	40
3.2 Aktivieren der Maschinenverriegelung des Roboter-Controllers	43
3.3 Einstellung des Kommunikationsanschlusses des Roboter-Controllers	44
Lektion 4 Starten von WINCAPSII und Erstellen eines Systemprojektes	48
4.1 Starten des System Managers	48
4.2 Registrieren eines neuen Systemprojektes	50
4.3 Einstellung des Kommunikationsanschlusses eines PCs	51
Lektion 5 Definieren von Makros	53
Lektion 6 Eingeben und Bearbeiten von Programmen	55
6.1 Musterprogramm	55
6.2 Öffnen eines Programmbearbeitungsfensters	56
6.3 Eingeben von Programmcodes	57
6.4 Einsetzen des Befehlseditors	58
6.5 Speichern des Programmes	60
Lektion 7 Kompilieren des Programmes in das Laufzeit-Format	61
Lektion 8 Hochladen des Programmes (PC → Roboter-Controller)	62
Kapitel 3 Simulieren der Roboterbewegung auf einem PC mit dem erstellten Programm	65
Lektion 9 Vorbereiten des PCs auf die Simulation	67
9.1 Starten des Arm Managers	67
9.2 Einrichten der Kommunikationsverknüpfung zum Roboter-Controller im Arm Manager	67
Lektion 10 Zuweisen der aktuellen Positionswerte zur Positionsvariablen	68
10.1 Manuelles Simulieren der Roboterbewegung	68
10.2 Abrufen der aktuellen Roboterposition in der Positionsvariablen	70
10.3 Bearbeiten der Positionsvariablen	76
Lektion 11 Testlauf des Programmes	77
11.1 Laden des Programmes	77
11.2 Starten des Programmes	79

Lektion 12 Überwachen und Manipulieren der E/As	80
12.1 Starten des DIO Managers und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller	80
12.2 Überwachen der E/As	81
12.3 Ein-/Ausschalten der E/A-Dummy-Schalter	82
Lektion 13 Überwachen und Manipulieren der Variablen	83
13.1 Starten des Variablen Managers und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller	83
13.2 Überwachen von Variablen	84
Lektion 14 Kontinuierlicher Testlauf	85
14.1 Kontinuierliche Ausführung	85
14.2 Kontinuierliche Überwachung der E/As	86
14.3 Sofortiger Stop (Anhalten)	88
Kapitel 4 Steuern des Roboters über Programme	89
Lektion 15 Steuern des Roboters in der Praxis	90
15.1 Lösen der Maschinenverriegelung	90
15.2 Einstellung der Roboter-Geschwindigkeit und -Beschleunigung	91
15.3 Einschalten des Antriebsmotors	92
15.4 Programmierung	93
15.5 Starten von Programmen	96
15.6 Wechseln der Robotergeschwindigkeit	97
15.7 Stoppen des Roboters bei kontinuierlicher Ausführung	98
Lektion 16 Palettierung	102
Lektion 17 Effektiver Einsatz von Bibliotheken	110
17.1 Programmbank	111
17.2 Importieren von Programmen	113
Lektion 18 Beenden der Sitzung	115
18.1 Beenden von WINCAPSII und Herunterfahren des PCs	115
18.2 Ausschalten des Roboter-Controllers	115
Kapitel 5 Funktionen des DENSO Roboters	117
Lektion 19 Konformitätskontrolle	118
19.1 Stromgrenzwert-Funktion für einzelne Achsen	118
19.2 Tip-Konformitäts-Funktion	119
Lektion 20 Sonstige Funktionen	120
Anhänge	121
Appendix-1 Glossar	122
Appendix-2 Bezeichnungen der Roboter-Controller-Teile	132
Appendix-3 Bezeichnungen der Komponenten des Programmiergeräts	134
Appendix-4 Menüstruktur des Programmiergeräts	136

Kapitel 1

Steuern des Roboters über das Programmiergerät

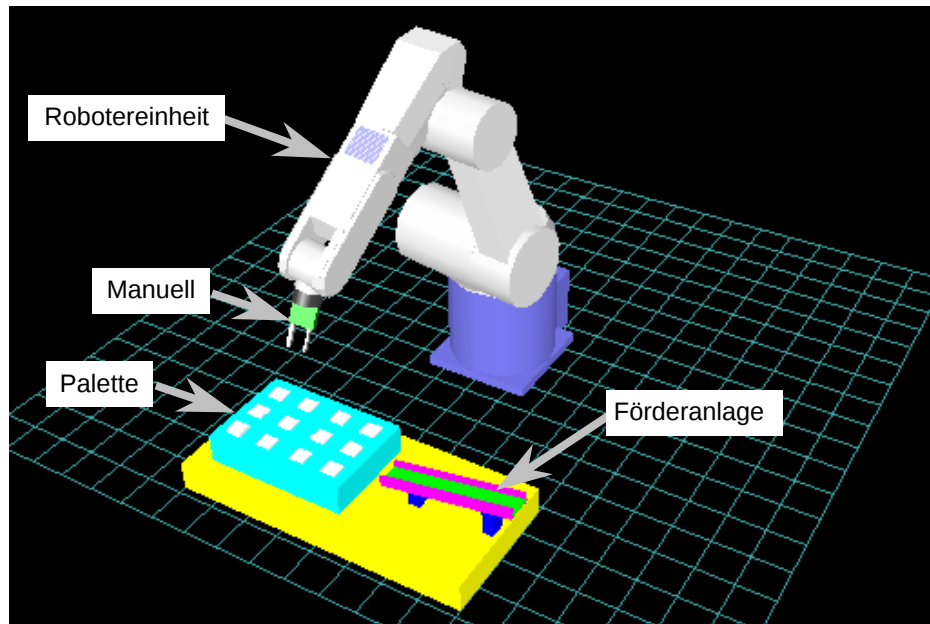
In Kapitel 1 werden Sie folgendes tun:

- Lernen, wie das Programmiergerät zu handhaben und einzusetzen ist.
- Folgendes mit dem Programmiergerät ausprobieren:
 - Durchführen einer sicheren und präzisen manuellen Operation (in den Gelenk-, X-Y- und Werkzeug-Modi)
 - Kalibrieren (CAL) des Roboters
 - Erstellen und Bearbeiten von Programmen
 - Durchführen einer Sicherheits-Programmierung
 - Durchführen einer Sicherheits-Programmierprüfung
 - Sicheres Starten und Stoppen von Programmen

► Lektion 1	Steuern des Roboters im manuellen Modus	3
1.1	Grundfunktionen des Programmiergeräts	3
1.2	Manuelles Steuern des Roboters über das Programmiergerät	5
► Lektion 2	Steuern des Roboters über ein einfaches Programm	14
2.1	Erstellen eines einfachen Programmes mit dem Programmiergerät	15
2.2	Programmierung	24
2.3	Programmierprüfung	29
2.4	Steuern des Roboters im Auto-Modus	34
■ Kapitel 1	Probleme bei der praktischen Ausführung	38

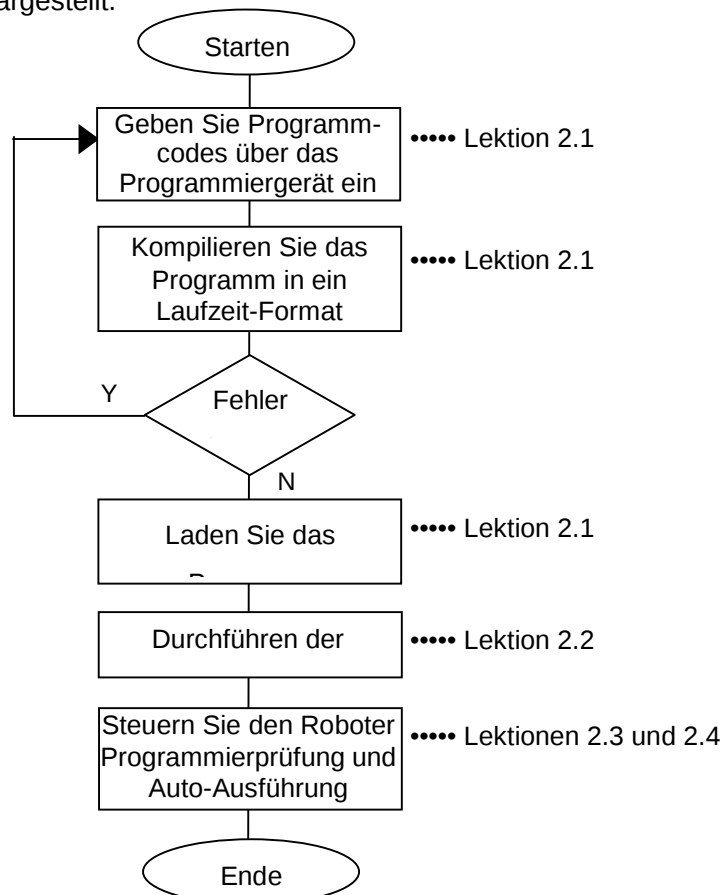
Beispiel für eine Geräte-Einstellung

Auf der Abbildung unten ist ein Beispiel für eine Geräte-Einstellung mit dem Roboter als Teil der Produktionslinie dargestellt. Dieser Roboter führt Palettierungsaufgaben aus.



Prozeßfluß von der Programmerstellung bis zum Prüfen der Roboterbewegung

Im folgenden ist der Prozeßfluß beginnend mit der Programmerstellung bis hin zum Prüfen der Roboterbewegung dargestellt.



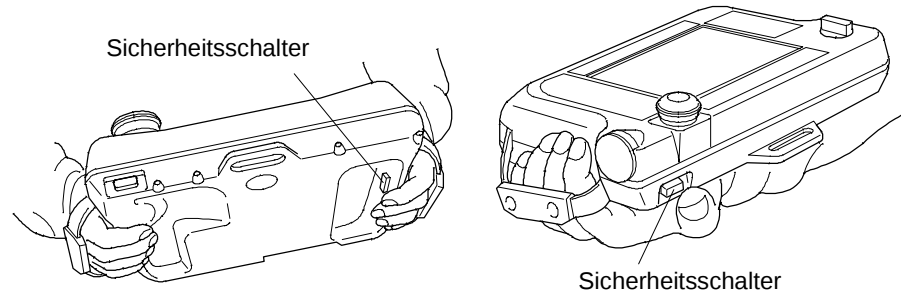
Lektion 1 Steuern des Roboters im manuellen Modus

1.1 Grundfunktionen des Programmiergeräts

Halten des Programmiergeräts und des Sicherheitsschalters

Bei Verwendung des Programmiergeräts ist dieses wie unten gezeigt zu greifen.

Das Programmiergerät verfügt über zwei Sicherheitsschalter. Somit kann es auf zwei verschiedene Arten gehalten werden:



Halten des Programmiergeräts

★Tip★

Der Sicherheitsschalter dient dazu, den Roboter automatisch und sicher zu stoppen, wenn der Bediener aufgrund unvorhergesehener Umstände, z. B. der Bediener erleidet einen Blackout oder stirbt bei der manuellen Steuerung des Roboters mit dem Programmiergerät, den Roboter nicht mehr länger korrekt steuern kann. Wenn eine ähnliche Situation auftritt, wird der Bediener den Sicherheitsschalter deutlich stärker oder schwächer drücken. Beim Sicherheitsschalter handelt es sich um einen Schalter mit 3 Positionen, der folgende 3 Betriebszustände erkennen und entsprechend darauf reagieren kann:

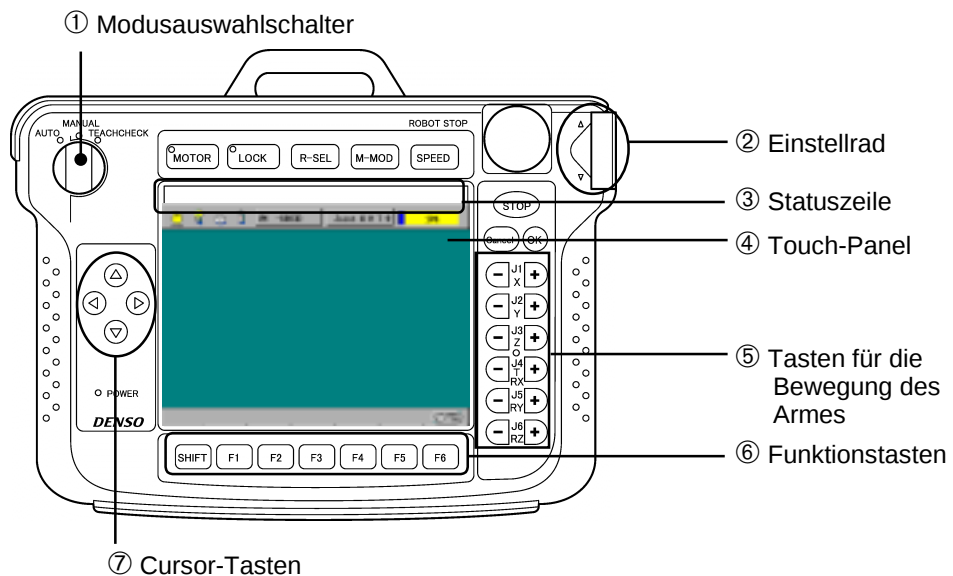
- 1) Wenn der Schalter nicht oder nur leicht gedrückt wird
→ Schalter: AUS
- 2) Wenn der Schalter mit korrektem Druck gedrückt wird
→ Schalter: EIN
- 3) Wenn der Schalter zu stark gedrückt wird
→ Schalter: AUS

Wenn der Schalter nicht auf EIN steht, kann der Roboter weder gesteuert noch angetrieben werden.

Aus Sicherheitsgründen ist der Roboter darauf ausgerichtet, daß der Sicherheitsschalter im manuellen Modus gedrückt gehalten werden muß, wenn der Bediener z. B. eine der Tasten zur Armbewegung seitwärts drückt.

Grundeinstellung des Programmiergeräts

Wenn der Controller eingeschaltet ist, erscheint die unten dargestellte Hauptanzeige auf dem Programmiergerät.



Programmiergerät – Hauptanzeige

① Modusauswahlschalter

Dadurch kann zwischen den Betriebsmodi "Auto, Manuell und Programmierprüfung" hin- und hergeschaltet werden.

② Einstellrad

Dadurch können Werte leichter angepaßt werden.

③ Statuszeile

Diese zeigt immer den aktuellen Betriebsmodus und den Roboterstatus an.

④ Touch-Panel

Der LCD-Bildschirm des Programmiergeräts ist ein Touch-Panel: Indem Sie die auf dem Bildschirm angezeigten Schaltflächen oder Dateneintragsbereiche berühren, können Sie Operationen ausführen und eine Auswahl vornehmen.

Achtung: Berühren Sie das Programmiergerät nur mit den Fingern, niemals mit der Spitze eines Bleistifts oder anderen spitzen Gegenständen. Andernfalls könnte die LCD-Anzeige beschädigt werden.

⑤ Tasten für die Bewegung des Armes

Über diese Tasten kann der Roboterarm manuell in die gewünschte Richtung bewegt werden. Es ist ebenso erforderlich, den Sicherheitsschalter gleichzeitig gedrückt zu halten.

⑥ Funktionstasten

F1 bis F6 werden normalerweise am Bildschirm angezeigt. Dies kann, falls erforderlich, durch Drücken der Umschalttaste auf eine Anzeige von F7 bis F12 umgeschaltet werden.

⑦ Cursor-Tasten

Diese dienen dazu, den Cursor auf der Bildschirmanzeige oder der Eingabemaske hin- und herzubewegen.

Weitere Einzelheiten zu jedem Abschnitt des Programmiergeräts, siehe Anhang 3.

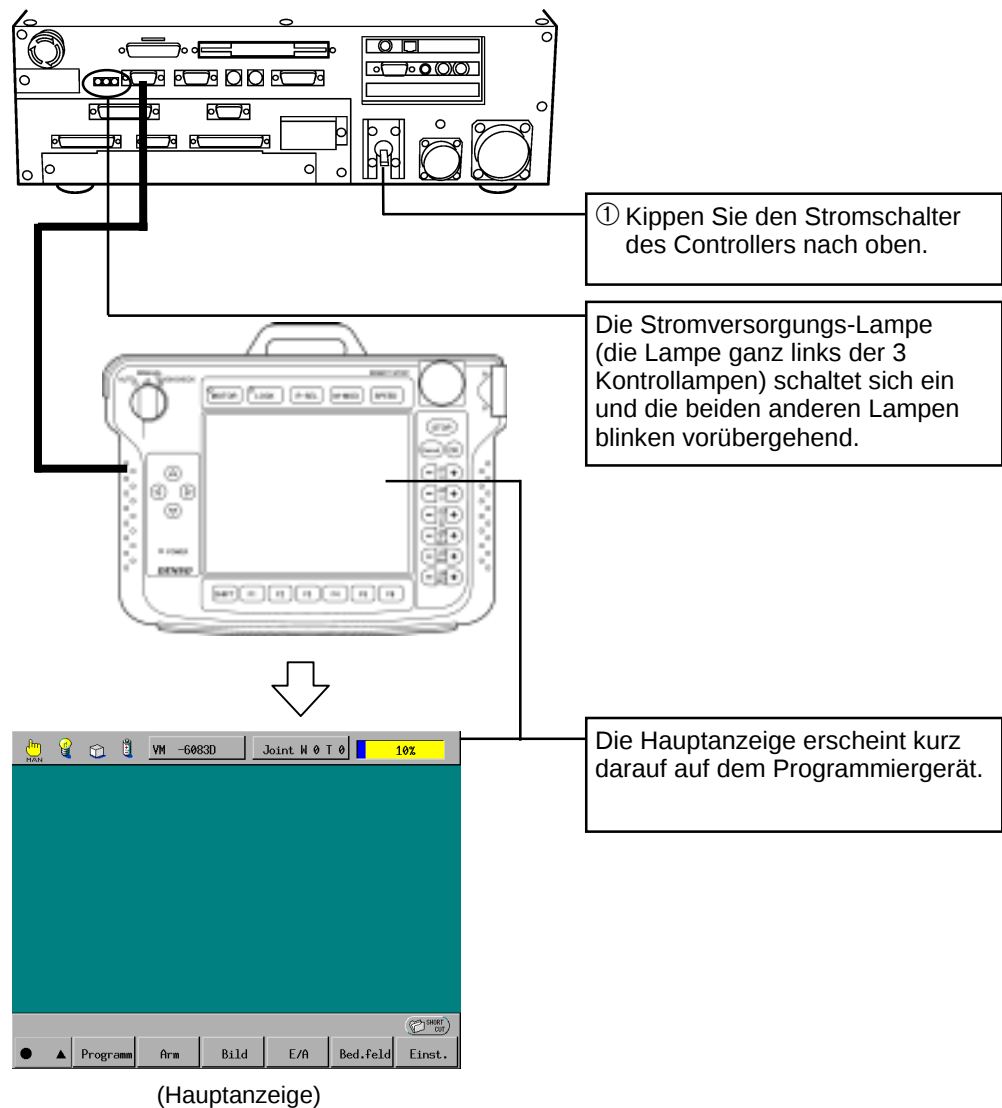
1.2 Manuelles Steuern des Roboters mit dem Programmiergerät

Sie beginnen Ihre Übung zuallererst damit, den Roboter-Controller und Motor einzuschalten und den Roboter manuell mit dem Programmiergerät zu steuern.

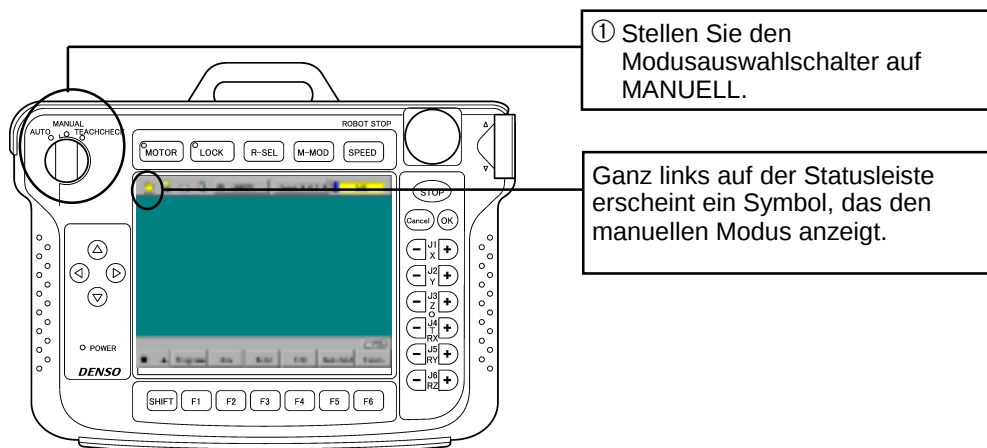
SCHRITT 1 Prüfen Sie, ob alles sicher ist, bevor Sie fortfahren

- Stellen Sie sicher, daß der Roboter korrekt installiert ist.
- Stellen Sie sicher, daß sich niemand im Sicherheitsraum des Roboters befindet.

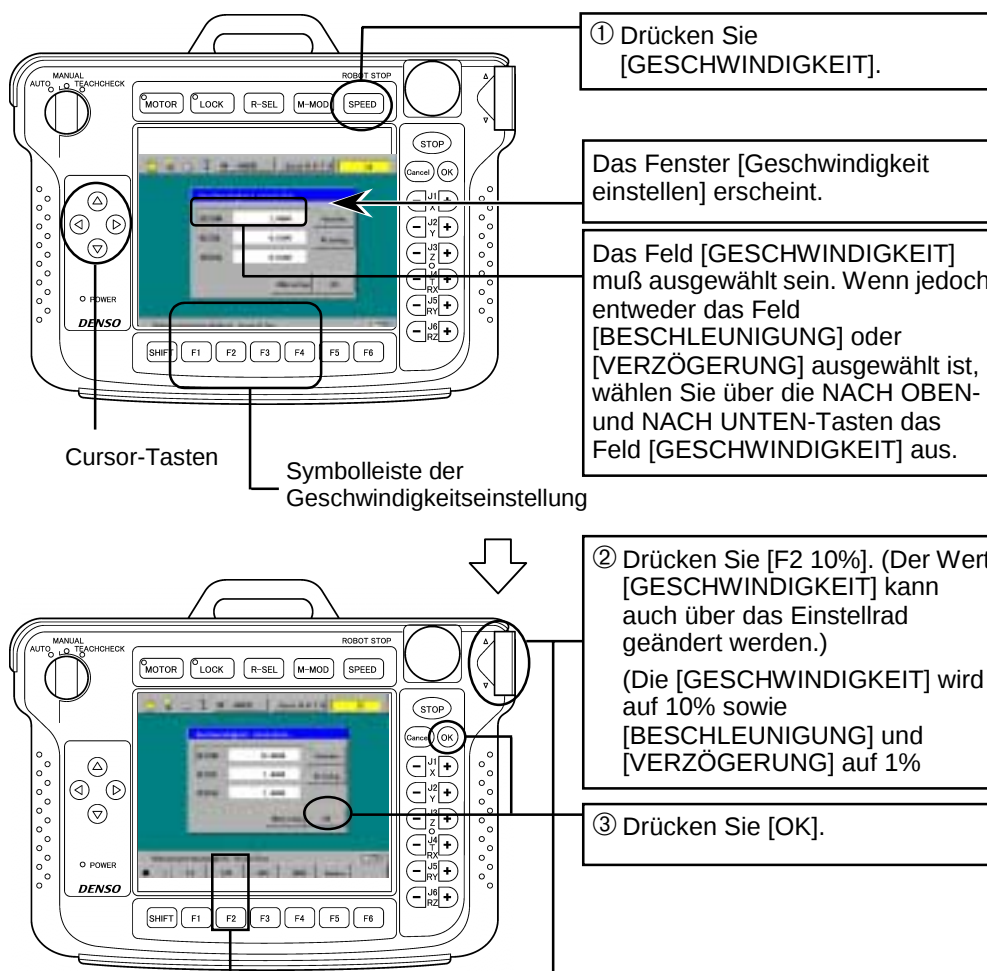
SCHRITT 2 Schalten Sie den Roboter-Controller EIN



SCHRITT 3 Setzen Sie den Roboter in den manuellen Modus

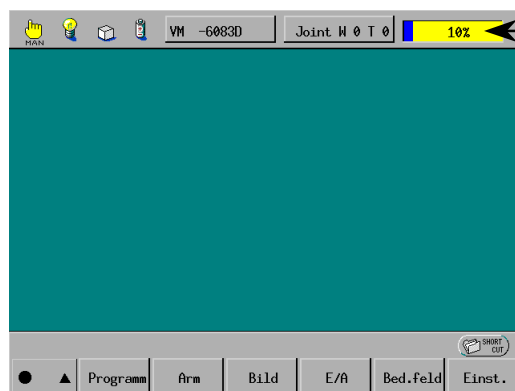


SCHRITT 4 Stellen Sie Geschwindigkeit und Beschleunigung ein



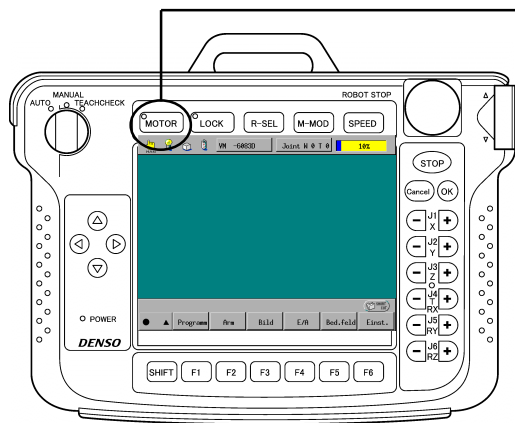
★Bemerkungen★

Lassen Sie zu Anfang diese Einstellungen unverändert, da Sie den Roboter aus Sicherheitsgründen zunächst nur langsam steuern. Die Einstellungen können Sie später ändern, wenn Sie sich an die Steuerung des Roboters mit dem Programmiergerät gewöhnt haben.



Die Anzeige [GESCHWINDIGKEIT] wechselt zu 10%.

SCHRITT 5 Einschalten des Motors



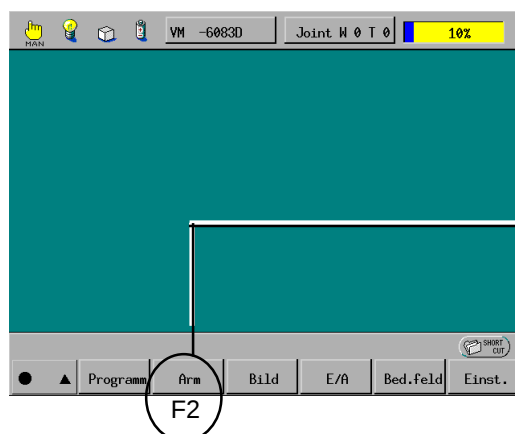
① Drücken Sie [MOTOR].
Der Motor wird eingeschaltet und die Lampe in der Taste [MOTOR] leuchtet auf.

SCHRITT 6 Manuelles Bewegen jedes Roboterarms



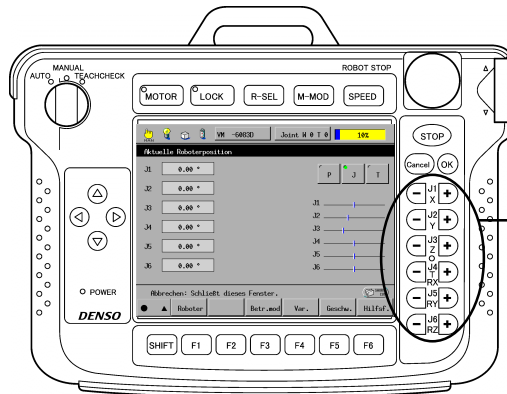
Vorsicht

Sobald diese Operation ausgeführt ist, wird sich der Roboterarm bewegen. Kein Arbeiter darf sich dann mehr im Sicherheitsraum des Roboters befinden.



① Drücken Sie [F2 Arm].





② Beobachten Sie den Roboter, halten Sie den Sicherheitsschalter gedrückt, und drücken Sie nacheinander die Tasten für die Armbewegung

Derjenige Arm, der mit dem Betrieb der Tasten J1 bis J6 zur Armbewegung seitwärts in Verbindung steht, wird bewegt. Im Fenster "Aktuelle Roboterposition" wird der Winkel jeder Achse angezeigt.

Achtung: Bis zur Durchführung der Kalibrierung (CAL) kann der Roboter nur im Gelenk-Modus gesteuert werden. Darüberhinaus sind die für die Koordinaten jeder Achse angezeigten Werte nicht korrekt.

★**Vorsicht**★

CAL ist erforderlich, um den Roboter mit exakten Werten zu steuern.

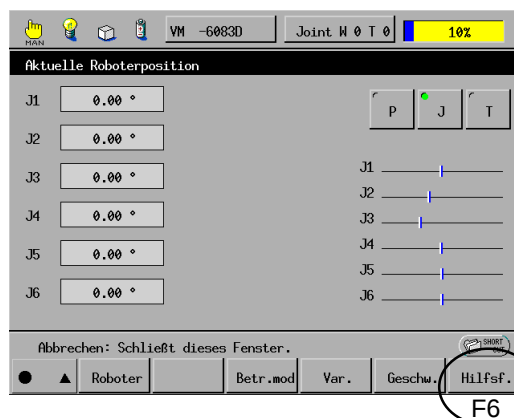
SCHRITT 7 Ausführen der Kalibrierung (CAL)

CAL steht für Kalibrierung, wodurch alle Roboterachsen betätigt werden, um den Roboterarm jeweils nur leicht zu bewegen und die aktuelle Armposition zu bestätigen, nachdem die Stromversorgung des Controllers eingeschaltet ist.

Die CAL-Prozedur wird im folgenden näher beschrieben.

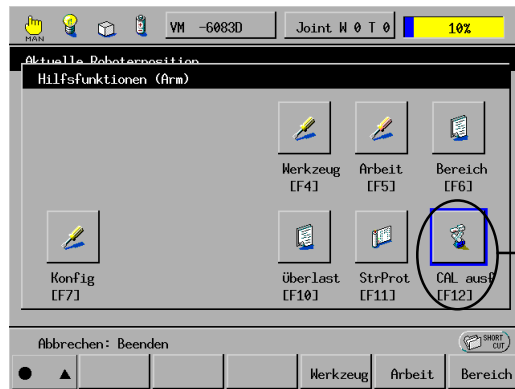
⚠**Vorsicht**

Durch die Ausführung der Kalibrierung wird der Roboterarm bewegt. Bevor Sie fortfahren, stellen Sie sicher, daß alle Arbeiter den Sicherheitsraum des Roboters verlassen haben und sich dort auch keine Hindernisse befinden.

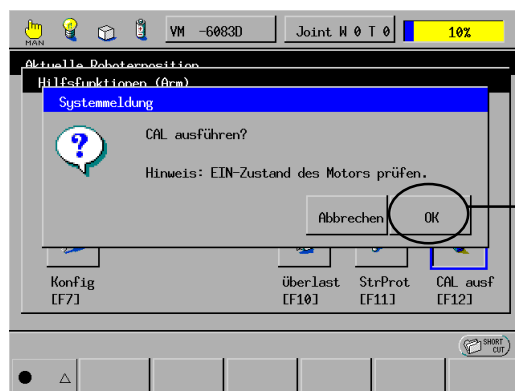


① Drücken Sie [F6 Hilfsf.] bei angezeigtem Fenster [Aktuelle Roboterposition].

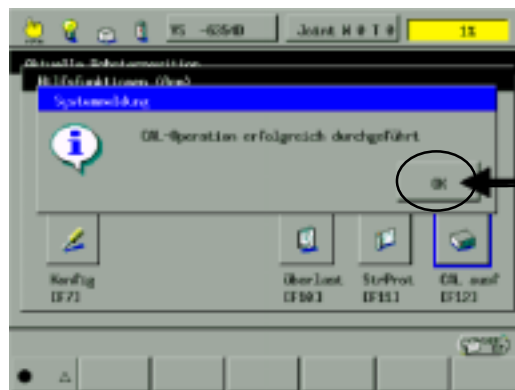




② Drücken Sie [F12 CAL ausf].

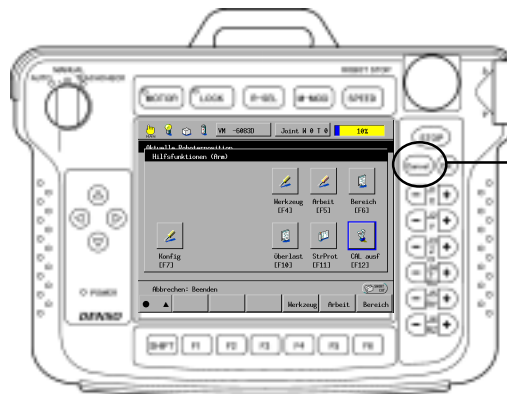


③ Prüfen Sie die Systemmeldung und drücken Sie dann [OK].



④ Prüfen Sie, ob die Systemmeldung "Erfolgreich beendete "CAL-Operation!" erscheint, und drücken Sie [OK].

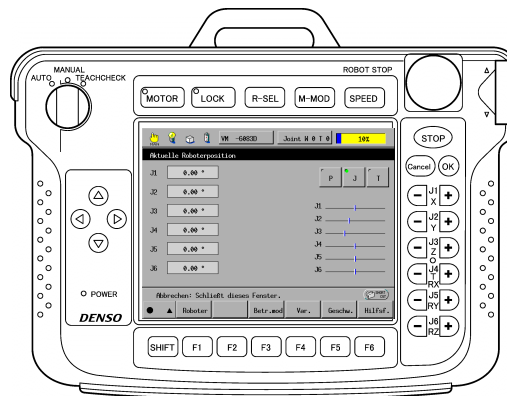




⑤ Nach der Rückkehr zum Fenster [Hilfsfunktionen (Armstellung)] drücken Sie [Abbrechen].

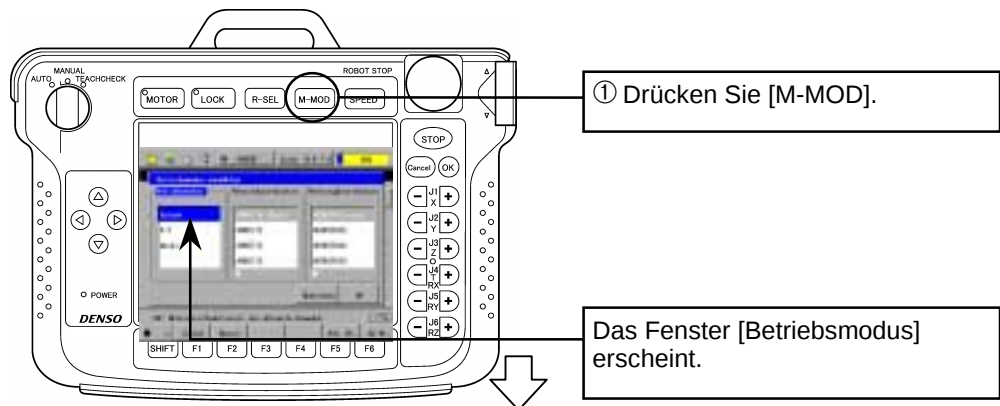


Das Fenster [Aktuelle Roboterposition] wird erneut angezeigt. Nun ist die Kalibrierung (CAL) abgeschlossen und der Roboter kann gesteuert werden.



SCHRITT 8

Auswählen des manuellen Modus und manuelles Steuern des Roboters

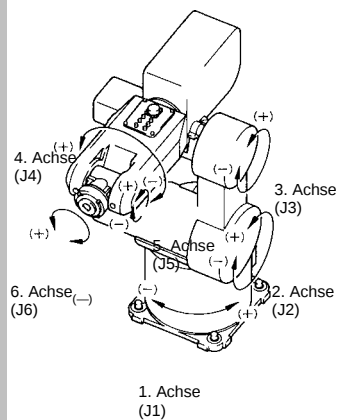


★Punkt★

Im manuellen Modus können Sie einen der 3 folgenden Betriebsmodi auswählen: Gelenk-, X-Y- und Werkzeug-Modus.

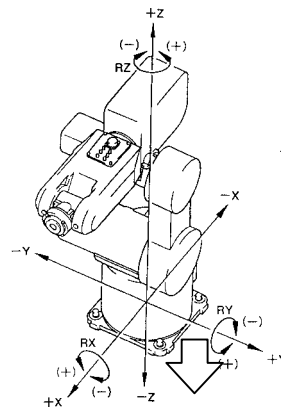
<Gelenk-Modus>

Ermöglicht es, jedes der sechs Gelenke unabhängig voneinander anzusteuern.



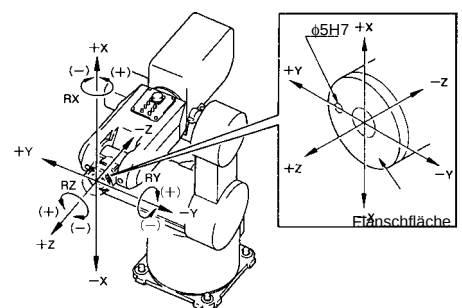
<X-Y-Modus>

Ermöglicht es Ihnen, den Roboterflansch jeweils linear zur X-, Y- oder Z-Achse anzusteuern. Falls Sie die Tasten RX, RY oder RZ betätigen, dreht sich der Roboterarm um jede Achse der virtuellen Arbeitskoordinaten, die im Zentrum der Flanschoberfläche definiert sind, ohne das Zentrum der Flanschoberfläche zu ändern.

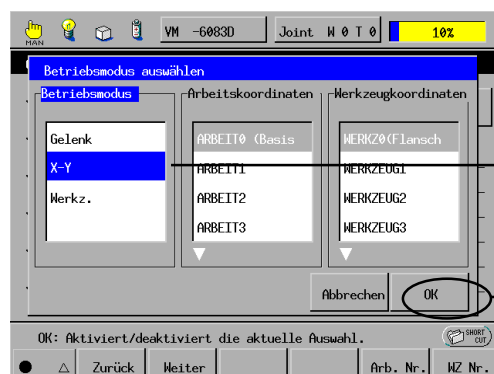


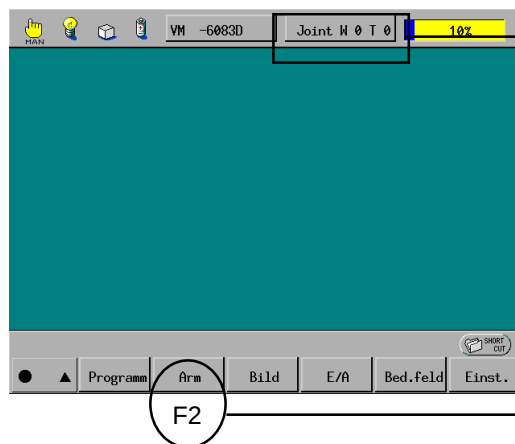
<Werkzeug-Modus>

Ermöglicht es Ihnen, den Roboterflansch jeweils linear zur X-, Y- oder Z-Achse anzusteuern. Falls Sie die Tasten RX, RY oder RZ betätigen, dreht sich der Roboterarm um jede Achse der Werkzeugkoordinaten.



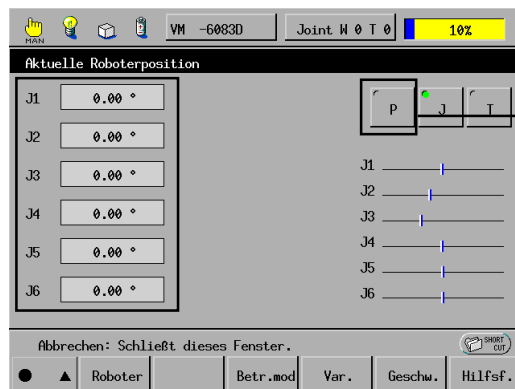
In dieser Lektion üben Sie, den Roboter im X-Y-Modus zu steuern.





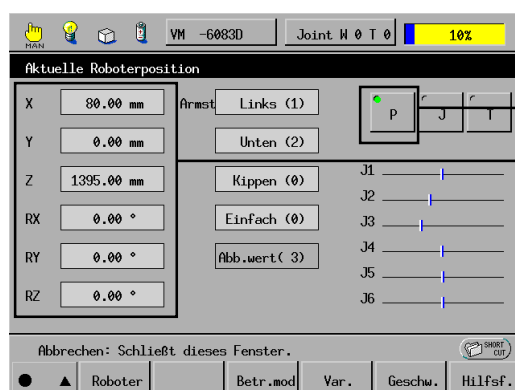
X-Y erscheint auf der Statusleiste.

④ Drücken Sie [F2 Arm].

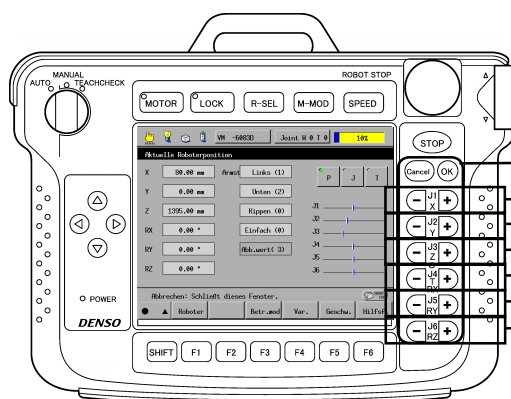


Das Fenster "Aktuelle Roboterposition" erscheint.

⑤ Drücken Sie die Schaltfläche P (Positionsvariable), um die aktuelle Roboterposition anzuzeigen. Sie können statt der Schaltfläche "P" auch die Umschalttaste und [F7 P anz] in der Menüleiste drücken.
(Dies ist erforderlich, um den Roboter im X-Y-Modus zu



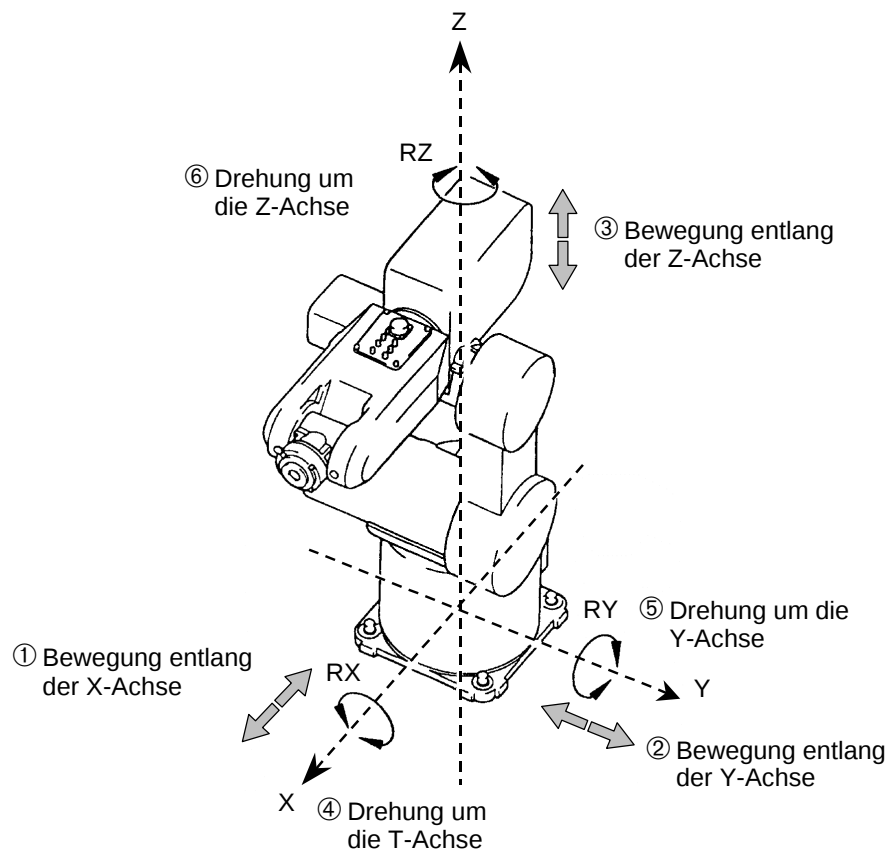
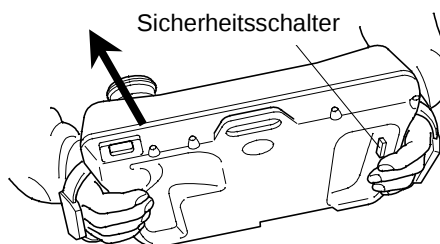
Die P-Lampe schaltet sich ein und der Bildschirm ändert sich entsprechend, damit die aktuelle Roboterposition als Positionsvariable angezeigt werden



⑥ Steuern Sie den Roboter, indem Sie bei gedrücktem Sicherheitsschalter die Tasten für die Armbewegung seitwärts

Tasten für die Bewegung des Armes

- ① Bewegung in X-Richtung
- ② Bewegung in Y-Richtung
- ③ Bewegung in Z-Richtung
- ④ Drehung um die X-Achse
- ⑤ Drehung um die Y-Achse
- ⑥ Drehung um die Z-Achse

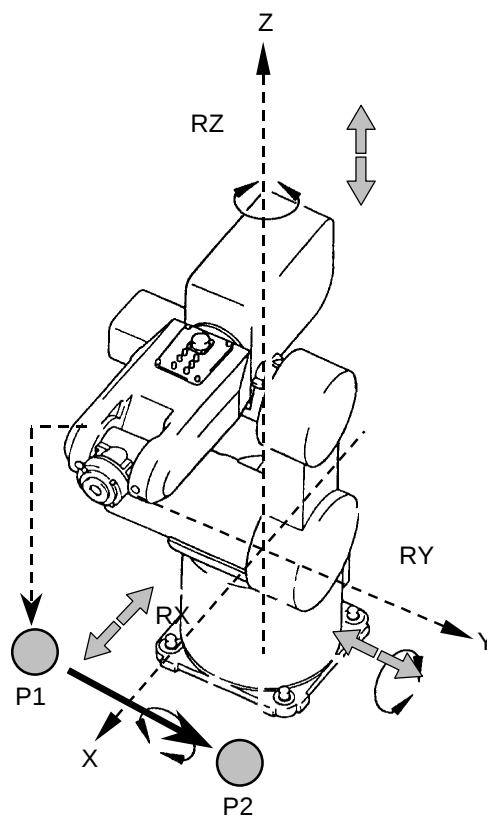


Lektion 2 Steuern des Roboters über ein einfaches Programm

Um den Roboter auf eine vorgegebene Weise steuern zu können, müssen Sie ein Programm erstellen und die Positionen programmieren, zu denen sich der Roboterarm hinbewegen soll.

In dieser Lektion üben Sie, den Roboterarm wie unten dargestellt von P1 zu P2 zu bewegen. Dies wird in der folgenden Reihenfolge beschrieben.

- 2.1 Erstellen eines einfachen Programmes mit dem Programmiergerät
- 2.2 Programmierung (Programmierung von P1 und P2)
- 2.3 Programmierprüfung
- 2.4 Steuern des Roboters bei automatischem Betrieb



2.1 Erstellen eines einfachen Programmes mit dem Programmiergerät

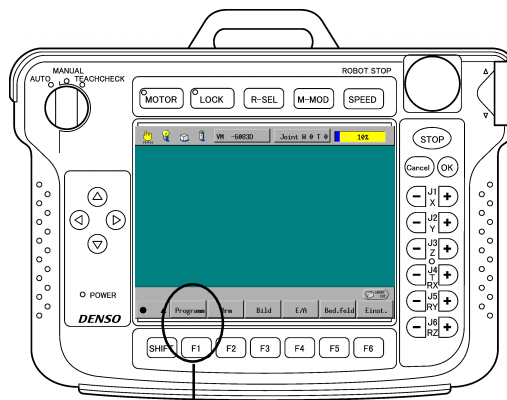
Zuerst geben Sie Codes eines einfachen Programmes über das Programmiergerät ein.

Das Erstellen und Bearbeiten von Programmen muß im manuellen Modus erfolgen. Wenn Sie die folgende Vorgehensweise unmittelbar im Anschluß an die vorherige Lektion ausführen, ist der Roboter bereits im manuellen Modus, und Sie können sofort fortfahren. Falls nicht, müssen Sie den Roboter in den manuellen Modus setzen, bevor Sie fortfahren.

Siehe PROGRAMMIERHANDBUCH für eine detaillierte Beschreibung zum Schreiben von Programmen.

SCHRITT 1 Öffnen eines Programmbearbeitungsfensters

Zum Erstellen eines Programmes müssen Sie auf der Anzeige des Programmiergeräts das Fenster zum Bearbeiten von Programmen öffnen.



① Drücken Sie [F1 Programm] in der Hauptanzeige.

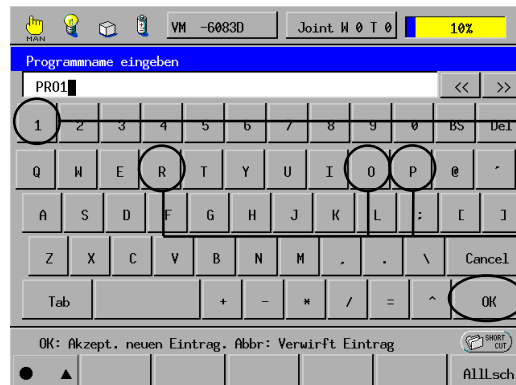


② Drücken Sie [F1 NeuesProg].



③ Drücken Sie [OK].

Dann geben Sie den Dateinamen des zu erstellenden Programmes (in diesem Fall PRO1) ein.

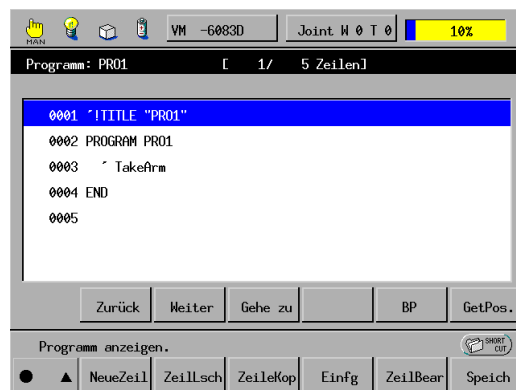


④ Geben Sie PRO1 durch Betätigen der Buchstaben- und numerischen Tasten ein.

⑤ Nach korrekter Eingabe von PRO1 drücken Sie [OK].



Damit ist die Vorbereitung der Programmbearbeitung abgeschlossen.



Die voreingestellten Programmcodes werden angezeigt.

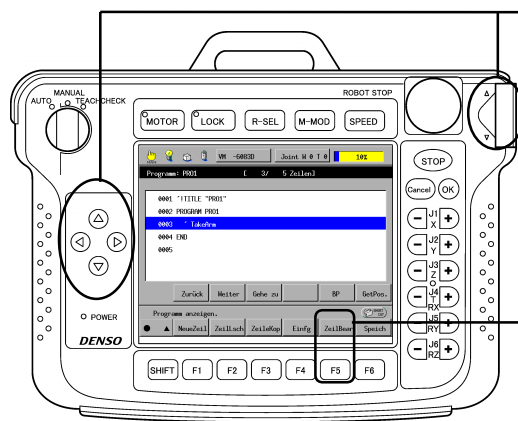
SCHRITT 2 Eingeben von Programmcodes

Bei diesem Schritt erstellen Sie ein Programm, das eine Bewegung von P1 zu P2 ermöglicht. Geben Sie die in der Tabelle unten aufgeführten Programmcodes ein.

Codierungen für "PRO1"

PROGRAM PRO1

TAKEARM	'Erfasst den Armsemaphor
SPEED 100	'Legt die interne Geschwindigkeit fest
MOVE L, P1	'Bewegt sich zu den für P1 angegebenen Koordinaten
MOVE L, P2	'Bewegt sich zu den für P2 angegebenen Koordinaten
GIVEARM	'Läßt den Armsemaphor los
END	



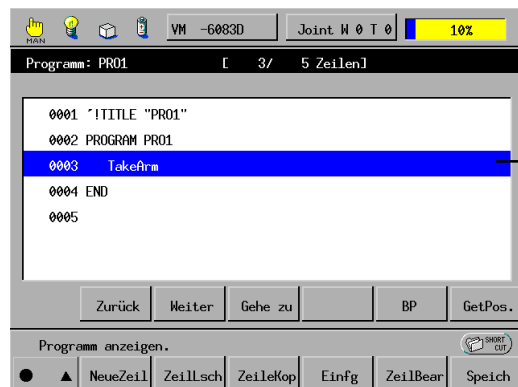
① Bewegen Sie im Fenster "Programm: PRO1" den Cursor über die Cursor-Tasten oder das Einstellrad zur 3. Zeile.

② Drücken Sie [F5 Zeile Bearbeiten].



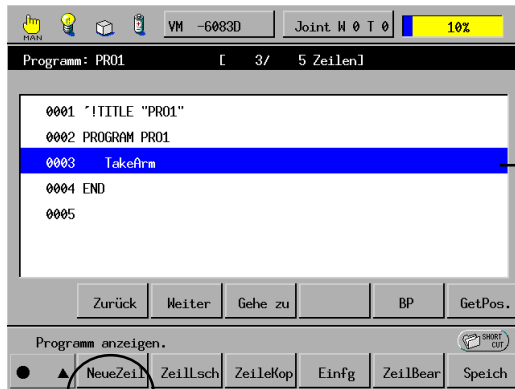
③ Löschen Sie den Apostroph (') oberhalb der Zeile mit den Cursor- und der [Del-Taste].

④ Drücken Sie [OK].



Auf dem Bildschirm erscheint wieder das Programmbearbeitungsfenster [Programm: PRO1], wobei die 3. Zeile geändert wurde.



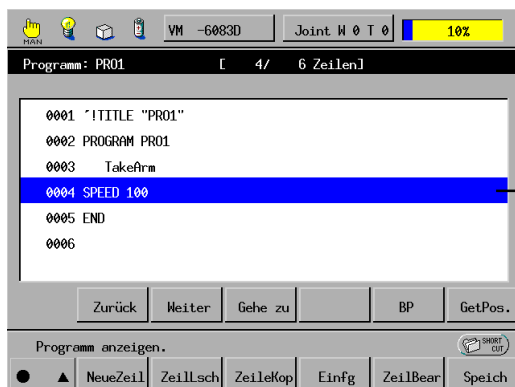


⑤ Bewegen Sie den Cursor in die dritte Zeile, und drücken Sie [F1 NeueZeil].



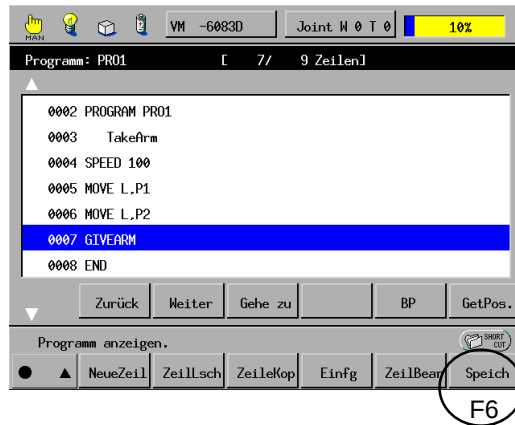
⑥ Geben Sie "GESCHWINDIGKEIT 100" über die Tastatur ein. Dies wird in diesem Fenster angezeigt.

⑦ Drücken Sie [OK].



Das Programmbearbeitungsfenster "Programm: PRO1" erscheint dort, wo "GESCHWINDIGKEIT 100" in der 4. Zeile angezeigt wird.





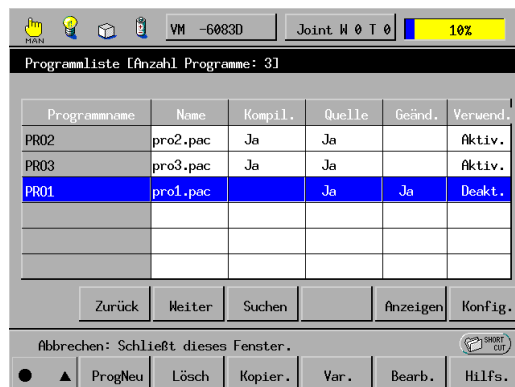
⑧ Geben Sie alle auf Seite 16 angegebenen Programmcodes genauso ein wie die [GESCHWINDIGKEIT 100].

⑨ Nachdem alle Codes erfaßt sind, drücken Sie [F6 Speichern].



⑩ Drücken Sie [OK], um das neu eingetragene Programm zu speichern.

Es erscheint wieder die Anzeige des Fensters "Programmliste".



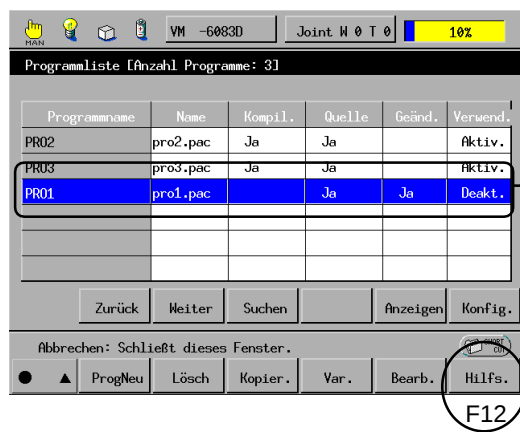
★Vorsicht★

- ① Wenn Sie die vorgenommenen Änderungen nicht speichern möchten, drücken Sie statt [OK] auf [Abbrechen], damit wieder die Anzeige des Programmbearbeitungsfensters erscheint, ohne daß die Änderungen gespeichert wurden.
- ② Kehren Sie zum Erstellen eines neuen Programms zurück zu Schritt 1.

SCHRITT 3 Kompilieren des Programmes in das Laufzeitformat

Nach dem Bearbeiten eines Programms müssen Sie dieses kompilieren, also in ein Laufzeitformat umwandeln, das vom Roboter-Controller ausgeführt werden kann.

Falls das bearbeitete Programm Syntaxfehler enthält, werden diese bei der Kompilierung erkannt. Sie müssen alle Syntaxfehler beheben, da Programme mit Syntaxfehlern nicht geladen oder ausgeführt werden können.



Programmliste [Anzahl Programme: 3]

Programmname	Name	Kompil.	Quelle	Geänd.	Verwend.
PRO2	pro2.pac	Ja	Ja		Aktiv.
PRO3	pro3.pac	Ja	Ja		Aktiv.
PRO1	pro1.pac		Ja	Ja	Deakt.

Zurück Weiter Suchen Anzeigen Konfig.

Abbrechen: Schließt dieses Fenster.

ProgNeu Lösch Kopier. Var. Bearb. Hilfs. F12

① Wählen Sie im Fenster "Programmliste" "PRO1".
(Sie können es über die Cursor-Tasten oder das Einstellrad oder durch direktes Berühren des Bildschirms auswählen.)

② Drücken Sie [F12 Konfig.].

↓



Systemmeldung

ⓘ Kompilierkennungen aktiv machen?

☒ Angegeb. Prog. aktiv machen.

☐ Alle Programme sind aktiv.

☐ Alle Programme sind inaktiv.

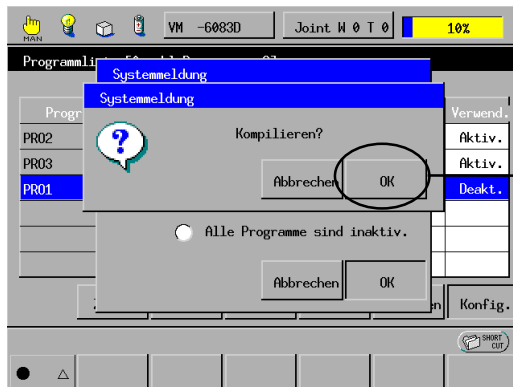
Abbrechen OK

Setzt Kompilierkennungen.

③ Wählen Sie "Aktivieren des angegebenen Programmes".

④ Drücken Sie [OK].

↓



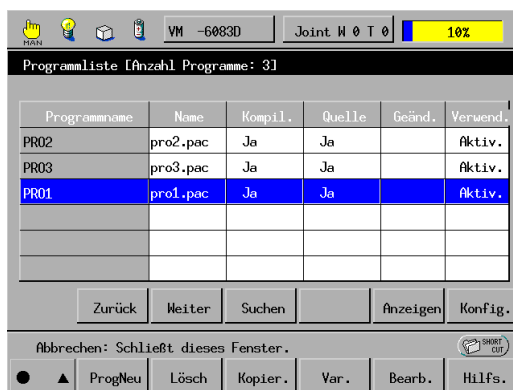
⑤ Drücken Sie [OK].
Das Kompilieren beginnt.



Sobald das Kompilieren abgeschlossen ist, erscheint wieder das Fenster [Programmliste].

★Vorsicht★

- ① Wenn Sie an dieser Stelle statt [OK] auf [Abbrechen] drücken, erscheint wieder das Fenster [Programmliste], ohne daß die Kompilierungs-Operation ausgeführt wurde.
- ② Es gibt noch eine andere Möglichkeit, Programme in ein Laufzeitformat zu kompilieren.
Drücken Sie [F6 Hilfs.] im Fenster [Programmliste], um das Fenster [Hilfsfunktionen (Programme)] aufzurufen. Drücken Sie dort dann [F12 Kompil.]. Mit dieser Methode können Sie dann weiter fortfahren, um Programme nach dem Kompilieren zu laden.

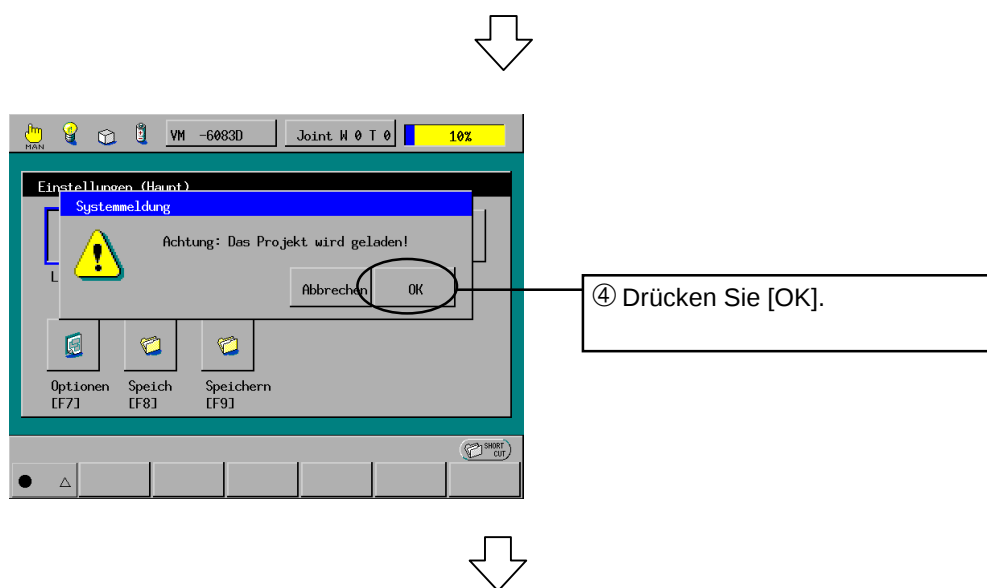
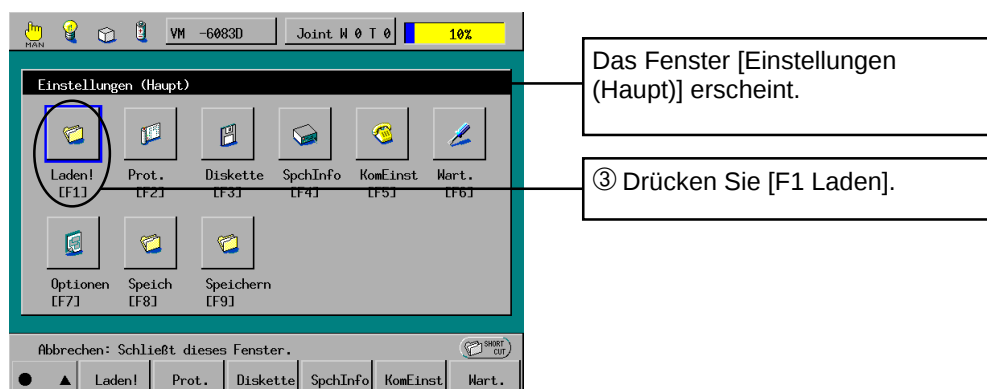
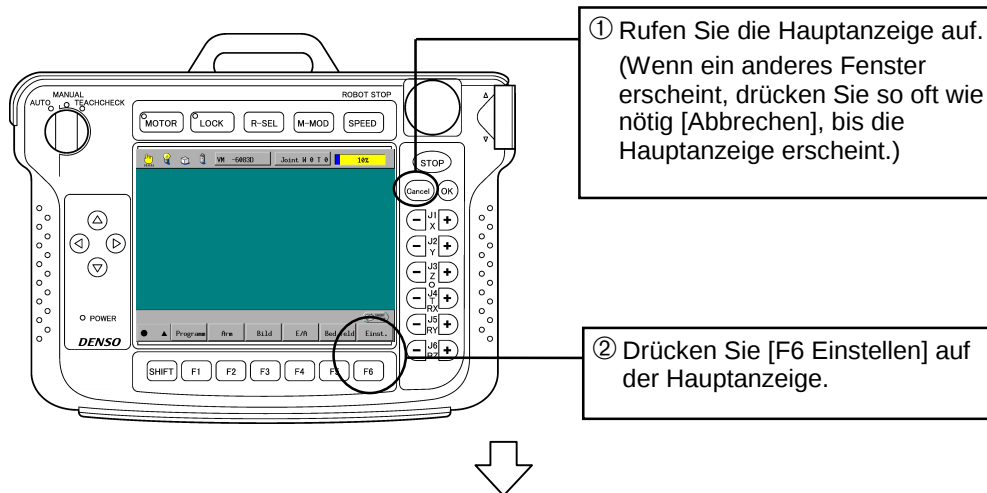


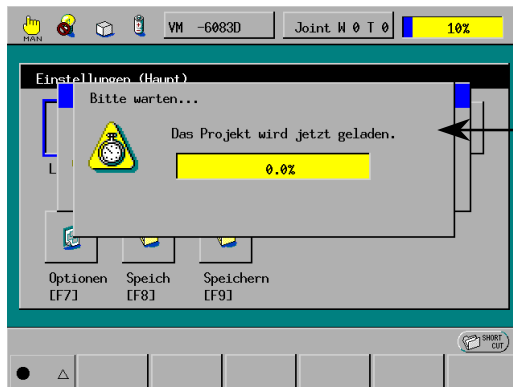
F6

SCHRITT 4 Laden des Programmes

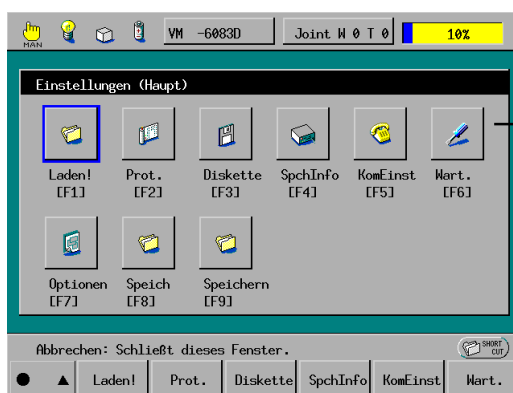
Sie müssen das kompilierte Programm laden, damit der Roboter es ausführen kann.

Selbst wenn kompilierte Programme vom an den Roboter-Controller angeschlossenen PC übertragen werden, können sie aber nicht ausgeführt werden. Sie müssen in den Speicherbereich geladen werden, von wo aus das Programm ausgeführt werden kann.





Die Meldung "Bitte warten... Projekt wird geladen." wird angezeigt.



Nach Abschluß des Ladevorgangs kehrt die Bildschirmanzeige wieder zum Fenster [Einstellung (Haupt)] zurück.

★Vorsicht★

Wenn Sie ein Projekt mit anderen als den im vorherigen Projekt verwendeten Variablen laden, erscheint die Fehlermeldung "Lokale Variable initialisiert".

Drücken Sie [OK], um fortzufahren.



⑤ Drücken Sie [OK].

Jetzt ist das Programm zur Ausführung bereit.
Drücken Sie [Abbrechen], um zur Hauptanzeige zurückzukehren.

Damit ist die Erstellung des Programmes zur Steuerung des Roboters abgeschlossen.

2.2 Programmierung

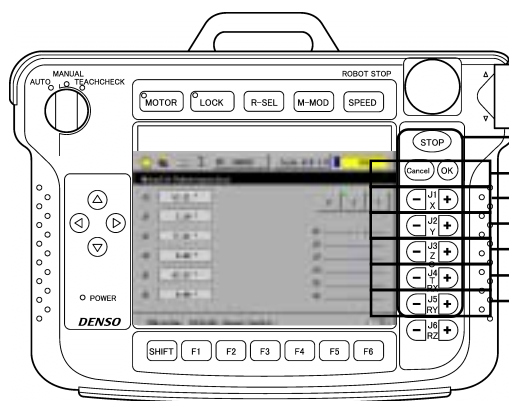
Unter "Programmierung" verstehen wir in diesem Zusammenhang eine Programmierung, bei der Sie über ein Programmiergerät den Roboter anweisen, welche Bewegungen ausgeführt werden sollen. Über die Programmierung können dem Roboter Bewegungen beigebracht werden.

Bei der Programmierung können Sie Positionen als Konstanten festlegen. Damit der Roboter jedoch die relative Positionsbeziehung zwischen sich selbst und dem Objektpunkt "lernt", müssen Sie den Roboter vor Ort entsprechend bewegen. Sie beschreiben bei der Programmierung daher Positionen als Datenvariablen und ordnen diesen Variablen bei der Programmierung vor Ort den tatsächlichen Werten zu.

Das in Lektion 2.1 erstellte Programm enthält die zwei Positionsvariablen P1 und P2. In diesem Abschnitt erfahren Sie, wie Sie die Roboterwerte für P1 und P2 programmieren.

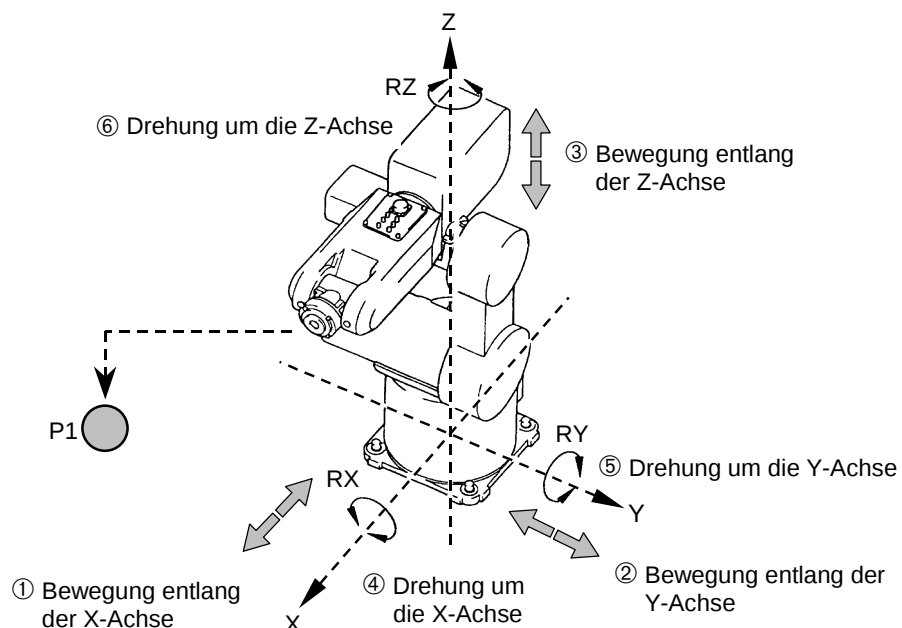
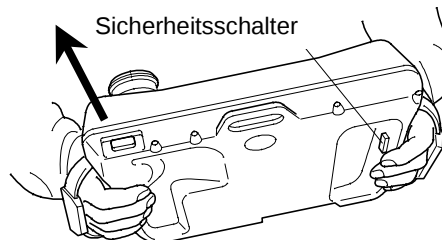
SCHRITT 1 Programmieren der Roboterposition P1

① Halten Sie den Sicherheitsschalter gedrückt, und drücken Sie die entsprechenden Tasten für die Armbewegung seitwärts, um den Roboterarm an die gewünschte Position zu bringen, die Sie P1 zuordnen wollen.

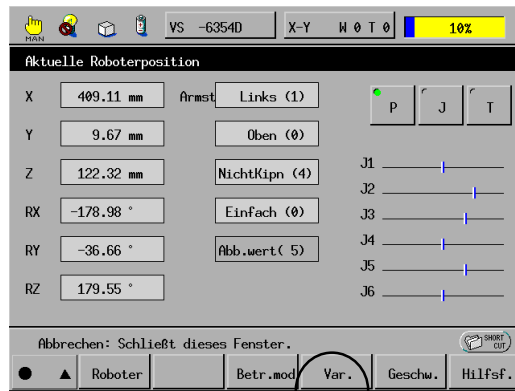


Tasten für die Bewegung des Armes

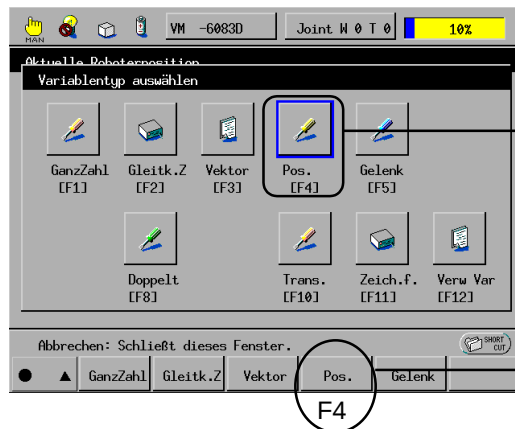
- ① Bewegung in X-Richtung
- ② Bewegung in Y-Richtung
- ③ Bewegung in Z-Richtung
- ④ Drehung um die X-Achse
- ⑤ Drehung um die Y-Achse
- ⑥ Drehung um die Z-Achse



SCHRITT 2 Zuordnen des programmierten Wertes zur [Variable P1]



① Drücken Sie [F4 Var.].

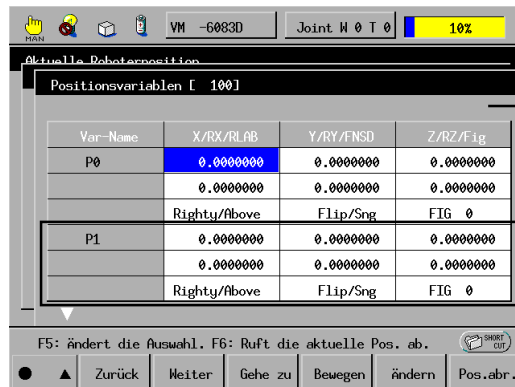


② Wählen Sie im Fenster [Variablentyp auswählen] den Variablentyp aus.
Drücken Sie an dieser Stelle [F4 Pos.], um einer Positionsvariablen einen Wert zuzuordnen.
(Es besteht auch die Möglichkeit, [Pos.] auf dem Bildschirm zu ...)

★Tip★

Eine Variable verweist auf eine Programmkennung eines Speicherortes, die aus beliebigen Zahlen oder Buchstaben besteht und sich während des Programmes ändern kann. Folgende Typen von Variablen werden unterstützt:

- I. (Ganzzahl): Ganzzahlige Variable (Wertebereich: -2147483648 bis +2147483647)
- F. (Gleiten): Gleitpunkt-Variablen (Wertebereich: -3,402823E+383,402823E+38)
- D. (Doppelt): Doppelpräzisions-Variablen (Wertebereich: -1,7976931348623157D+308 bis 1,7976931348623157D+308)
- V. (Vektor): Vektor-Variablen (X, Y, Z)
- P. (Pos): Positionsvariablen (X, Y, Z, RX, RY, RZ, FIG)
- J. (Gelenk): Gelenkvariablen (J1, J2, J3, J5, J6)
- T. (Trans): Homogene Transformationsvariablen (Px, Py, Pz, Ox, Oy, Oz, Ax, Ay, Az, FIG)
- S. (Zeichenfolge): Zeichenfolgen-Variablen (die aus einer Zeichenfolge von bis zu 247 Zeichen bestehen kann)

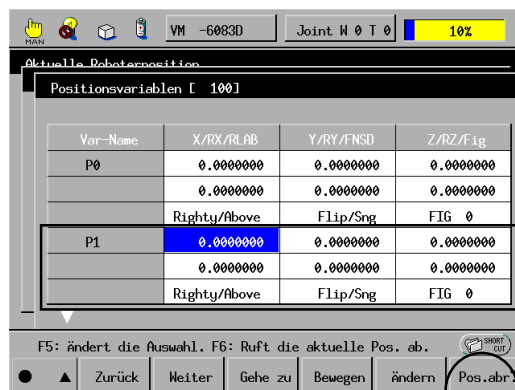


Das Fenster [Positionenvariablen] erscheint.

③ Wählen Sie mit den Cursor-Tasten oder dem Einstellrad das Feld [P1].

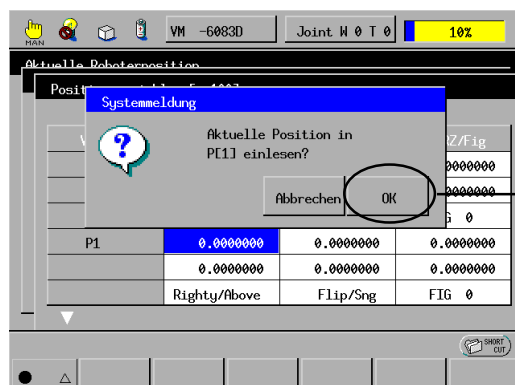
Das Fenster [Positionenvariablen] zeigt sieben Datentypen für jeden Variablennamen an.

Wenn Sie einen davon auswählen und markieren, z. B. im Feld [Var Name P1], dann bedeutet dies, daß [Var Name P1] ausgewählt ist.



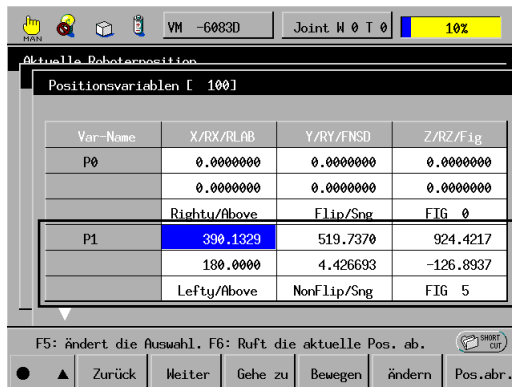
④ Prüfen Sie, ob der [Variablenname P1] ausgewählt ist.

⑤ Drücken Sie [F6 Pos. abbr.].



⑥ Prüfen Sie die Systemmeldung und wenn alles korrekt ist, drücken Sie [OK].



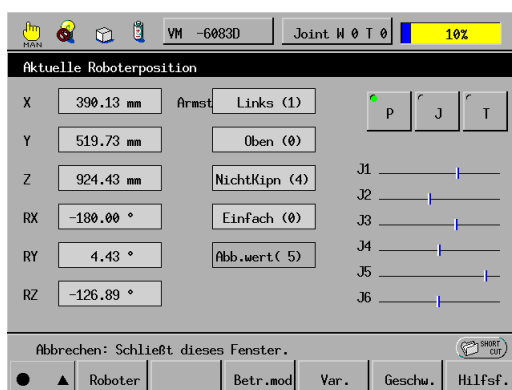


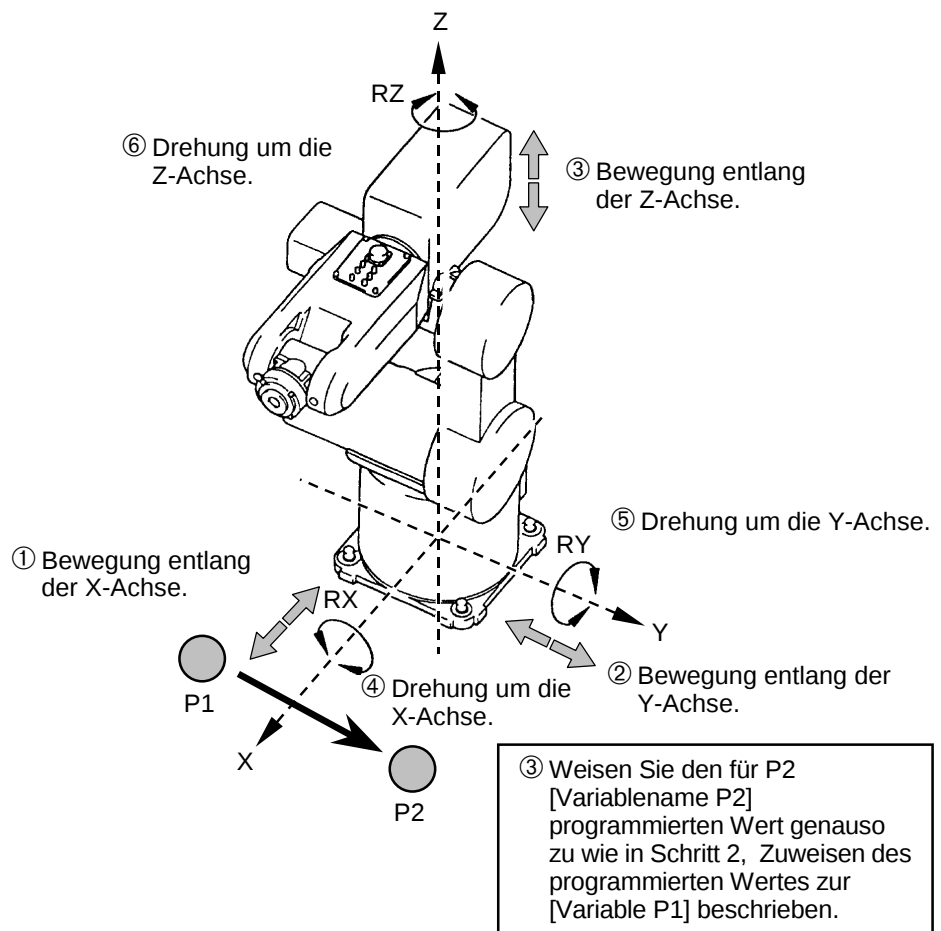
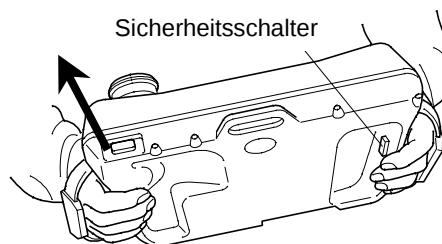
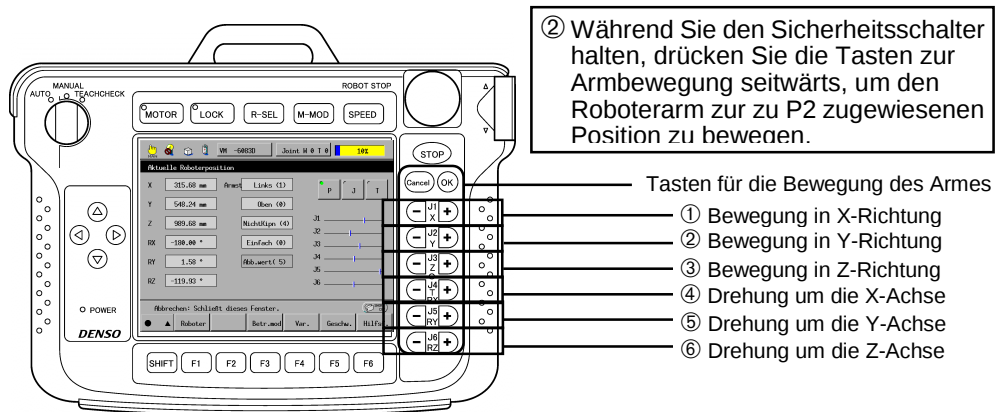
Die aktuelle Position wird in Variable P1 eingelesen.

SCHRITT 3 Programmieren der Roboterposition P2 und diese [Var Name P2] zuweisen



① Drücken Sie zweimal [Abbrechen], um zum Fenster [Aktuelle Roboterposition] zurückzukehren.



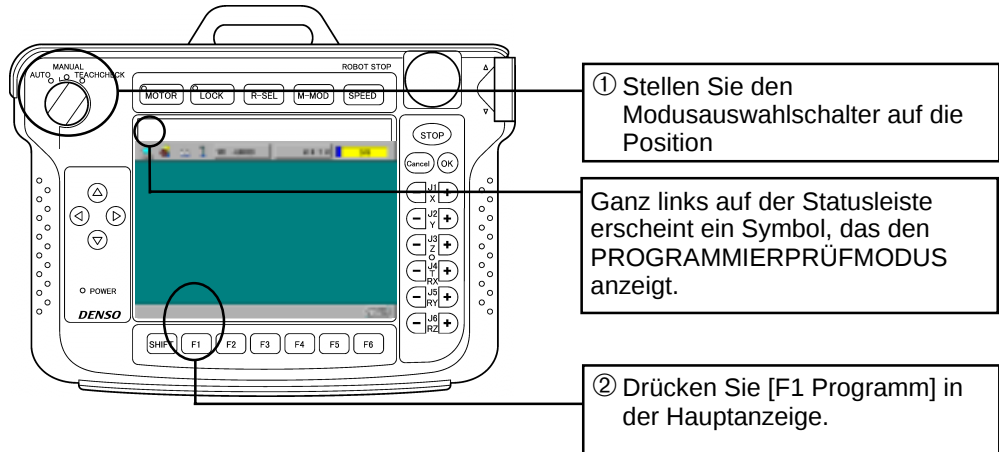


Damit ist die Programmierung von P1 und P2 abgeschlossen.

2.3 Programmierprüfung

“Programmierprüfung” verweist auf die Prüfung der Programmierergebnisse durch manuelles Ausführen des Programmes. Sie können die Programmierprüfung im Programmierprüfmodus durchführen.

SCHRITT 1 Setzen Sie den Roboter in den Programmierprüfmodus



Das Fenster “Programmliste” erscheint.

CHECK

VM -6083D

W 0 T 0

10%

Programmliste [Anzahl Programme: 5]

RÜCKSCHRITT: Rob.bew.anweisungen

Programmname	Status	Zeile	Laufz.	Prior.	F/B
AUTOEXEC	Angehalt	4	0.00	128	▽ Vorw
PR04	Angehalt	5	0.00	128	▽ Vorw
PR01	Angehalt	2	0.00	128	▽ Vorw
PR02	Angehalt	2	0.00	128	▽ Vorw
PR03	Angehalt	2	0.00	128	▽ Vorw

Zurück

Weiter

Suchen

Anzeigen

Konfig.

Abbrechen: Schließt dieses Fenster.

Anhalten

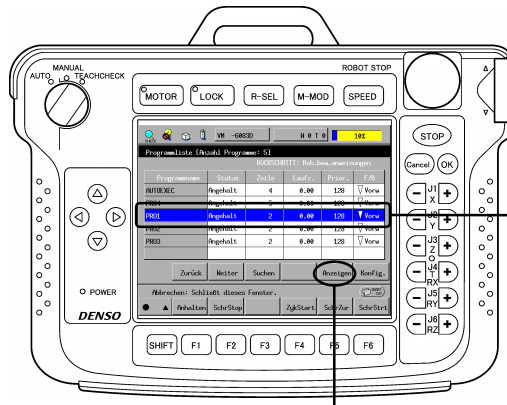
SchrStop

ZykStart

SchrZur

SchrStrt

SCHRITT 2 Schrittprüfung

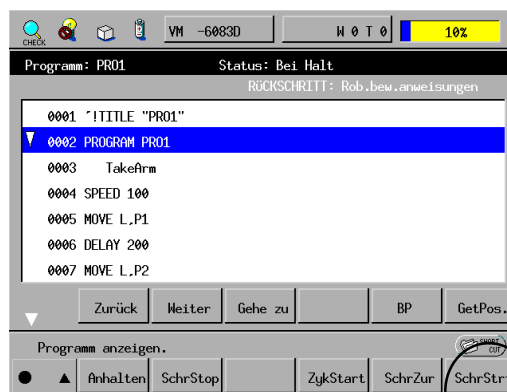


① Wählen Sie im Fenster "Programmliste" "PRO1".
(Die Auswahl kann über die Cursor-Tasten oder das Einstellrad oder durch direktes Berühren des Bildschirms)

② Drücken Sie [Anzeige], um die PRO1-Programmcodes aufzurufen.



Die Codierungsliste PRO1 erscheint im Programmbearbeitungsfenster "Programm: PRO1".



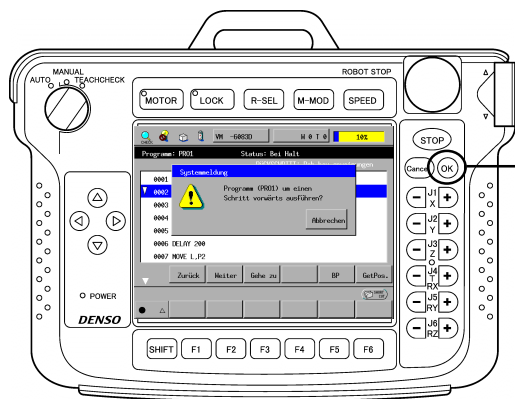
③ Drücken Sie [F6 ZyklusStart].
(Dies ist auch über den rechten Cursor möglich.)



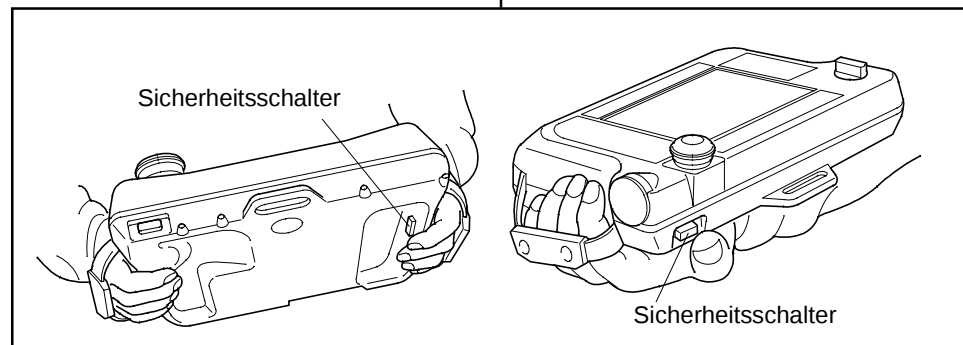
Diese Systemmeldung erscheint.

★Vorsicht★

Halten Sie während der Programmierprüfung stets eine Hand bereit, die STOP-Taste zu drücken.



④ Halten Sie den Sicherheitsschalter gedrückt, und drücken Sie [OK].
(Um eine Schritt-Operation abubrechen, drücken Sie [Abbrechen].)



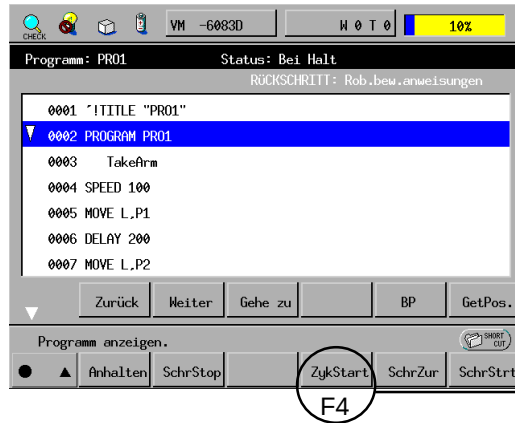
Halten Sie im Programmierprüfmodus Sicherheitsschalter und OK-Taste gedrückt, bis die Ausführung abgeschlossen ist. Wenn eine von beiden losgelassen wird, wird der Roboter sofort angehalten.



Führen Sie die oben beschriebene Prozedur wiederholt durch, um alle Codes in PRO1 auszuführen und zu prüfen, ob jede Bewegung sicher ist.

SCHRITT 2 Zyklusprüfung

Als nächstes prüfen Sie das gerade mit der Schrittprüfung geprüfte Programm, dieses Mal jedoch mit der Zyklusprüfung. Die Zyklusprüfung führt das ausgewählte Programm von der aktuellen Programmzeile bis zum Ende als Einzelzyklus aus.



① Drücken Sie [F4 ZykStart].



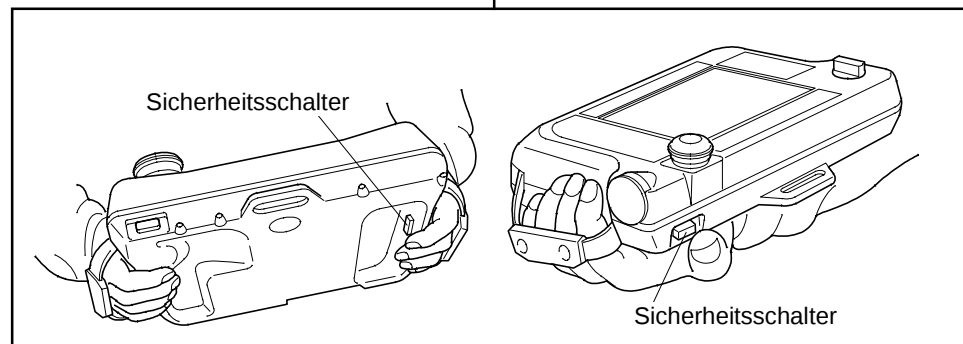
Diese Systemmeldung erscheint.

★**Vorsicht**★

Halten Sie während der Programmierprüfung stets eine Hand bereit, die STOP-Taste zu drücken.



② Halten Sie den Sicherheitsschalter gedrückt, und drücken Sie [OK].
(Um die Zyklusprüfung abzubrechen, drücken Sie



Halten Sie im Programmierprüfmodus Sicherheitsschalter und OK-Taste gedrückt, bis die Ausführung abgeschlossen ist. Wenn eine von beiden losgelassen wird, wird der Roboter sofort angehalten.



Da das Programm mit der Ausführung der Zyklusprüfung zur Steuerung des Roboters beginnt, wird der hervorgehobene Abschnitt im Fenster "Codierungsliste" entsprechend fortgesetzt.

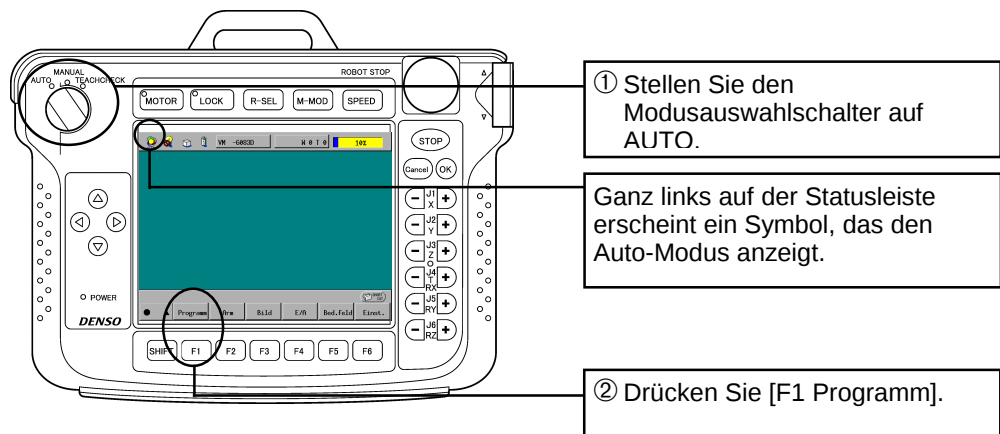
Wenn das Programm bis zum Ende ausgeführt worden ist, erfolgt ein Stop.

2.4 Steuern des Roboters im Auto-Modus

Nach der Programmierprüfung führen Sie das Programm im Auto-Modus entsprechend des im letzten Abschnitt bearbeiteten Programmes PRO1 aus.

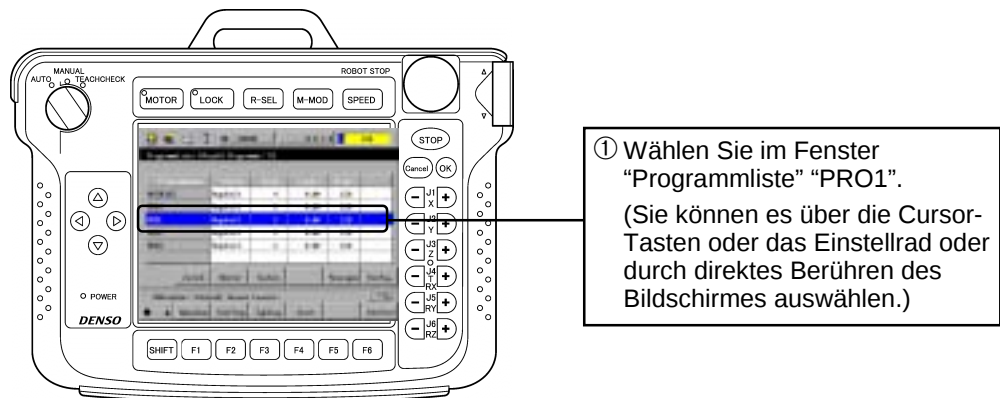
Achtung: Für Programme, die zum ersten Mal im Auto-Modus ausgeführt werden, stellen Sie das Reduzierungsverhältnis für die programmierte Geschwindigkeit auf 10% oder weniger ein. Im Auto-Modus kann der Roboter bei voller Geschwindigkeit laufen, während er im Manuellen Modus oder im Programmierprüfmodus auf 10% der vollen Geschwindigkeit reduziert wird.

SCHRITT 1 Setzen Sie den Roboter in den Auto-Modus



SCHRITT 2 Auswählen des auszuführenden Programmes

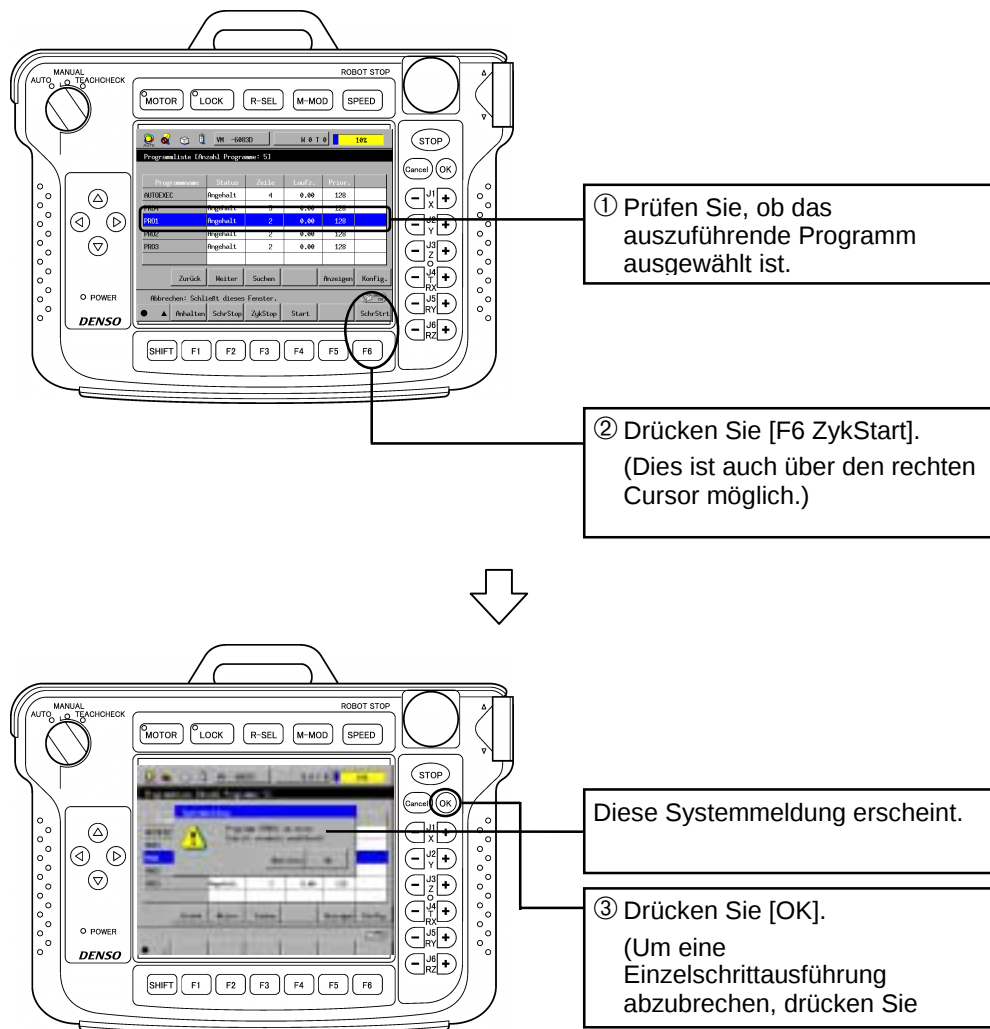
Wählen Sie im Fenster [Programmliste] das im Auto-Modus auszuführende Programm aus.



SCHRITT 3 Einzelschrittausführung

★Anmerkung★

Wenn Sie das Programm bei einer Einzelschrittausführung anzeigen möchten, drücken Sie vorher [F11 Anzeige].



★Vorsicht★

Halten Sie während der Programmausführung stets eine Hand bereit, die STOP-Taste zu drücken.



Das PRO1-Programm startet im Auto-Modus eine Einzelschrittausführung.

Führen Sie die oben beschriebene Prozedur wiederholt bis zum Ende des Programmes durch, und prüfen Sie, ob jede Bewegung sicher ist.

SCHRITT 4 Einzelzyklus-Ausführung

Nach Ausführung einer Einzelschritt-Ausführung starten Sie eine Einzelzyklus-Ausführung.

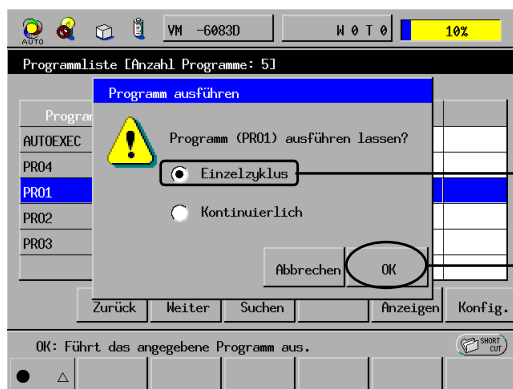


① Prüfen Sie, ob das auszuführende Programm ausgewählt ist.

② Drücken Sie [F4 Start.].

★Vorsicht★

Halten Sie während der Programmausführung stets eine Hand bereit, die STOP-Taste zu drücken.



③ Wählen Sie [Einzelzyklus], und drücken Sie dann [OK]. Programm PRO1 wird ausgeführt.

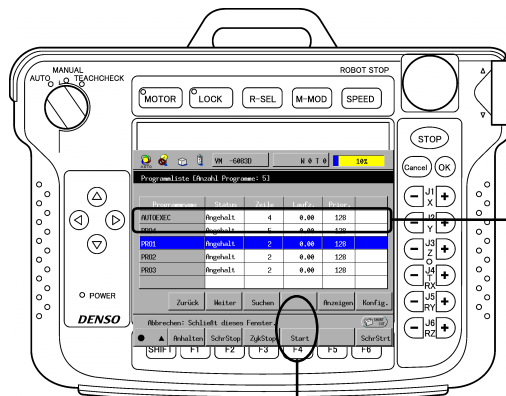
Wenn das Programm bis zum Ende ausgeführt worden ist, erfolgt ein Stop.

★Vorsicht★

Die auf der Anzeige angegebene abgelaufene Zeit bezieht sich auf die Dauer vom Start bis zum Ende des Programmes einschließlich vorübergehender Stopzeiten, die durch einen Schrittstop oder ein Anhalten verursacht wurden.

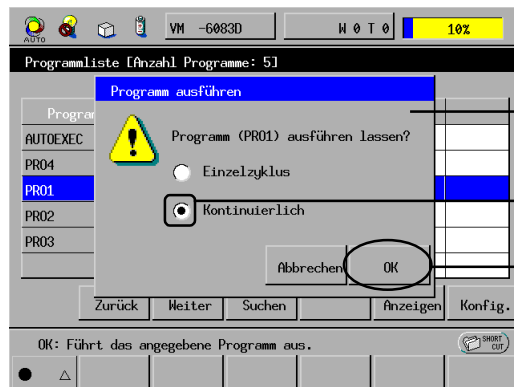
SCHRITT 5 Kontinuierliche Ausführung

Starten Sie eine kontinuierliche Ausführung des Programmes.



① Prüfen Sie, ob das auszuführende Programm ausgewählt ist.

② Drücken Sie [F4 Start.].



Der Auswahl-Bildschirm für [Einzelzyklus] und [Kontinuierlich] wird angezeigt.

③ Wählen Sie [Kontinuierlich].

④ Drücken Sie [OK].
Programm PRO1 wird kontinuierlich ausgeführt.
(Sie können die kontinuierliche Ausführung durch Anhalten (Stop) oder Schrittstop stoppen.)

★Vorsicht★

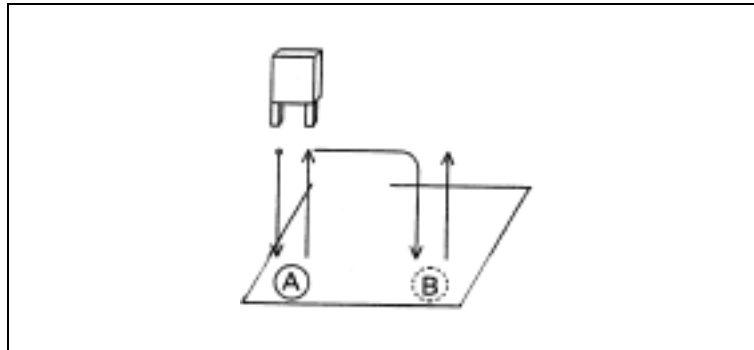
Halten Sie während der Programmausführung stets eine Hand bereit, die STOP-Taste zu drücken.



Damit ist die Prozedur zur Steuerung eines Roboters mit dem Programmiergerät abgeschlossen.

Kapitel 1 Probleme bei der praktischen Ausführung

Übung: Erstellen Sie ein Programm zum Verschieben eines Werkstückes von Punkt (A) zu Punkt (B), dann führen Sie eine Betriebsprüfung durch.



Folgendes gilt:

- E/A-Zuordnung Nr. 64 (Systemausgabe) ... Hand schließen
 Nr. 65 (Systemausgabe) ... Hand öffnen
- Geschwindigkeitsfaktor beim Aufnehmen und Ablegen eines Werkstückes: 30%
 Geschwindigkeitsfaktor für sonstige Bewegungen: 80%
- Alle Annäherungs- und Abweichungsdistanzen: 50 mm
- Interpolationssteuerung: PTP (Punkt-zu-Punkt)

Antwort:

```
0001  '!TITLE "P&P"
0002  PROGRAM PR010
0003      TAKEARM
0004      APPROACH P , P5 , 50 , S=80
0005      MOVE P , P5 , S=30
0006      DELAY 500
0007      RESET IO[65]
0008      SET IO[64]
0009      DEPART P , 50 , S=80
0010      APPROACH P , P6 , 50 , S=80
0011      MOVE P , P6 , S=30
0012      DELAY 500
0013      RESET IO[64]
0014      SET IO[65]
0015      DEPART P , 50 , S=80
0016      GIVEARM
0017  END
```

Kapitel 2

Erstellen eines Programmes auf einem PC mit WINCAPSII

In Kapitel 2 lernen Sie folgendes:

Starten Sie das PC-Programmiersystem WINCAPSII auf einem PC, und erstellen bzw. kompilieren Sie ein Programm. Sie werden das kompilierte Programm dann zum Roboter-Controller hochladen.

Außerdem werden Sie in Kapitel 2 auch die Maschinenverriegelung des Roboter-Controllers aktivieren. Dies dient als Vorbereitung auf Kapitel 3, wo Sie die programmierte Roboterbewegung auf einem PC-Bildschirm simulieren, ohne den Roboter tatsächlich zu steuern.

► Lektion 3 Einstellung des Roboter-Controllers mit dem Programmiergerät	40
3.1 Durchführen der Kalibrierung (CAL)	40
3.2 Aktivieren der Maschinenverriegelung des Roboter-Controllers	43
3.3 Einstellung des Kommunikationsanschlusses des Roboter-Controllers	44
► Lektion 4 Starten von WINCAPSII und Erstellen eines Systemprojektes	48
4.1 Starten des System Managers	48
4.2 Registrieren eines neuen Systemprojektes	50
4.3 Einstellung des Kommunikationsanschlusses des PCs	51
► Lektion 5 Definieren von Makros	53
► Lektion 6 Eingeben und Bearbeiten von Programmen	55
6.1 Musterprogramm	55
6.2 Öffnen des Programmbearbeitungsfensters	56
6.3 Eingeben des Programmcodes	57
6.4 Einsetzen des Befehlseditors	58
6.5 Speichern des Programmes	60
► Lektion 7 Kompilieren des Programmes in das Laufzeit-Format	61
► Lektion 8 Hochladen des Programmes (PC → Roboter-Controller)	62

Lektion 3 Einstellung des Roboter-Controllers mit dem Programmiergerät

3.1 Ausführen der Kalibrierung (CAL)

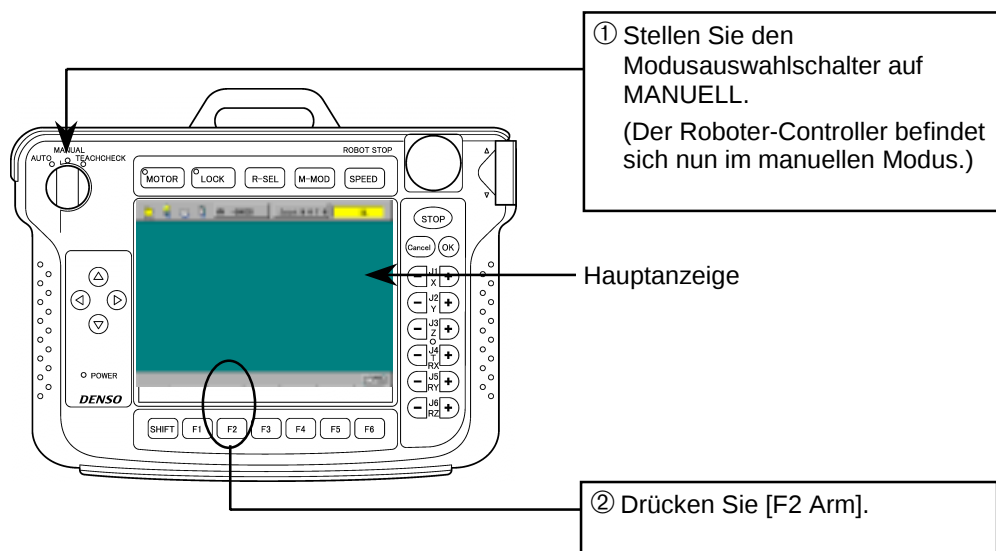
Die Kalibrierung (CAL), bei der alle Roboterachsen betätigt werden, ist erforderlich, um die aktuelle Position des Roboters zu bestätigen.



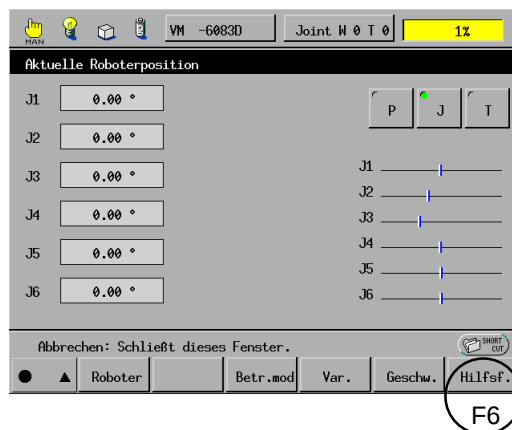
Warnung

Bei Ausführung der Kalibrierung wird der Roboterarm bewegt. Stellen Sie vor der Fortsetzung der CAL-Prozedur sicher, daß sich keine Personen innerhalb des Sicherheitsraums des Roboters aufhalten und daß dieser Bereich nicht durch irgendwelche Hindernisse blockiert wird.

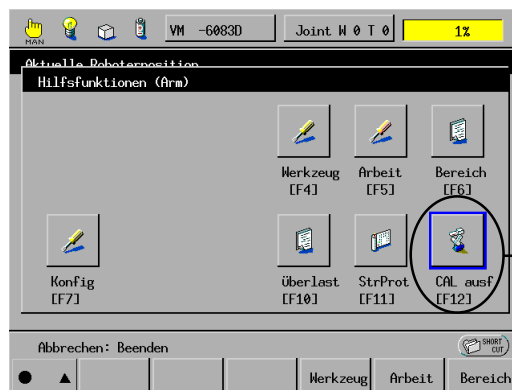
SCHRITT 1 Setzen Sie den Roboter in den manuellen Modus



SCHRITT 2 Starten der CAL-Operation



① Drücken Sie [F6 Hilfsfunktionen].



② Drücken Sie [F12 CAL ausf].

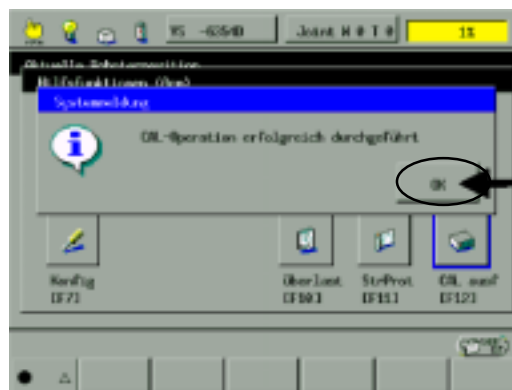
SCHRITT 3 Einschalten des Motors



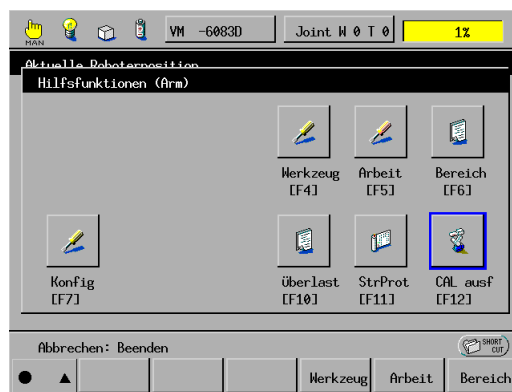
① Drücken Sie [MOTOR].
Der Motor wird eingeschaltet und die Lampe [MOTOR] leuchtet.

② Drücken Sie [OK].

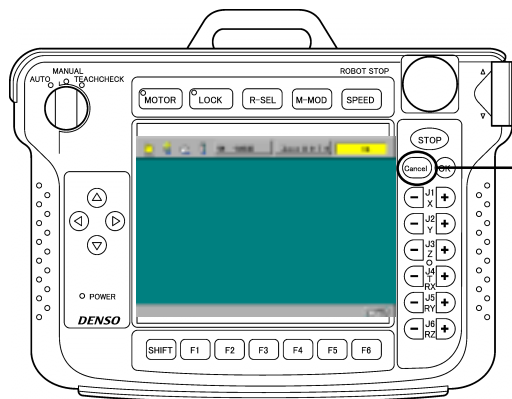
SCHRITT 4 Beenden der CAL-Operation



① Drücken Sie [OK].



Hiermit ist die Kalibrierung abgeschlossen.



② Drücken Sie [Abbrechen].
(Es erscheint wieder das Fenster [Aktuelle Roboterposition].)

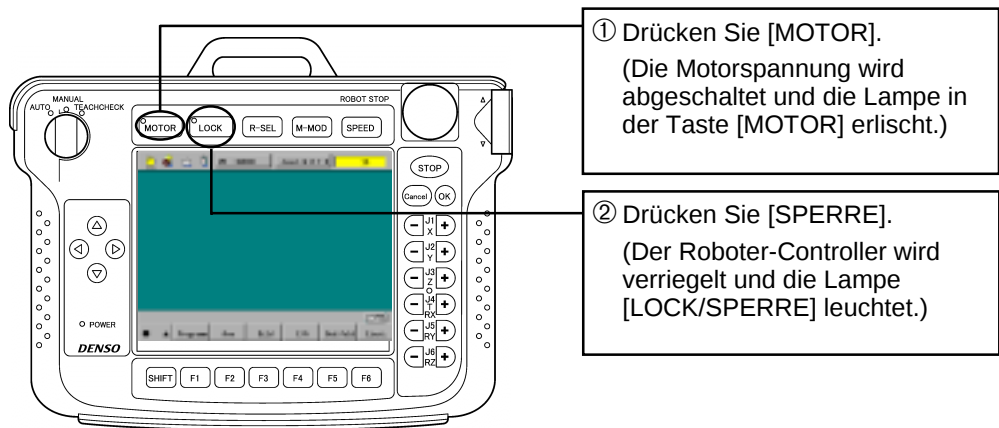
③ Drücken Sie [Abbrechen].
Der Benutzer kehrt zur Haupt-
anzeige zurück.)

3.2 Aktivieren der Maschinenverriegelung des Roboter-Controllers

Sie aktivieren nun die Maschinenverriegelung des Roboter-Controllers. Dadurch können Sie die programmierte Roboterbewegung am PC-Bildschirm simulieren, ohne den Roboter in Kapitel 3 tatsächlich zu steuern.

SCHRITT 1 Ausschalten des Motors

SCHRITT 2 Aktivieren der Maschinenverriegelung des Roboters



★**Vorsicht**★ Bevor Sie die Maschinenverriegelung des Roboter-Controllers aktivieren, stellen Sie sicher, daß der Motor ausgeschaltet ist, d. h. daß die [MOTOR]-Lampe aus ist.

★**Tip**★ [Version 1.4 oder höher]

Wenn die Maschine verriegelt ist, können Sie die E/A-Ausgabe einschränken. Weitere Details, siehe EINSTELLUNGSHANDBUCH, Abschnitt 5.5 "Anzeigen von E/A-Signalen und Simulieren der Roboterbewegung."

Das Dummy-Eingabesymbol auf der Statusleiste ändert sich je nach E/A-Ausgabebeschränkung.



: Keine E/A-Ausgabe eingeschränkt



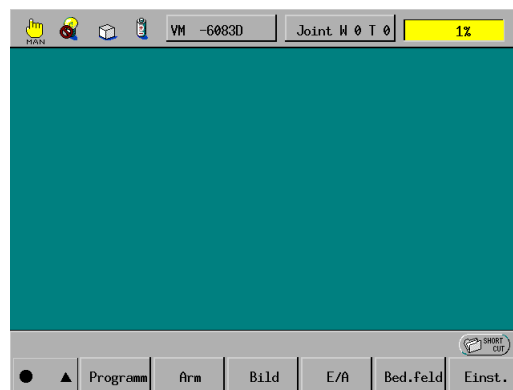
: E/A-Ausgabe eingeschränkt

3.3 Einstellung des Kommunikationsanschlusses des Roboter-Controllers

Damit der Roboter-Controller mit dem PC Daten austauschen kann, müssen Sie den Kommunikationsanschluß einstellen.

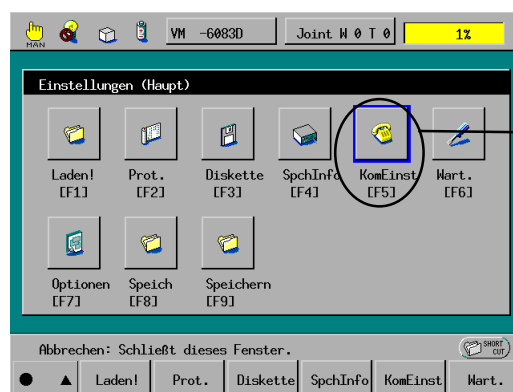
In diesem Teilabschnitt wird die am häufigsten verwendete Anschlußmethode mit RS-232C beschrieben.

SCHRITT 1 Einstellen der Kommunikationsberechtigungen



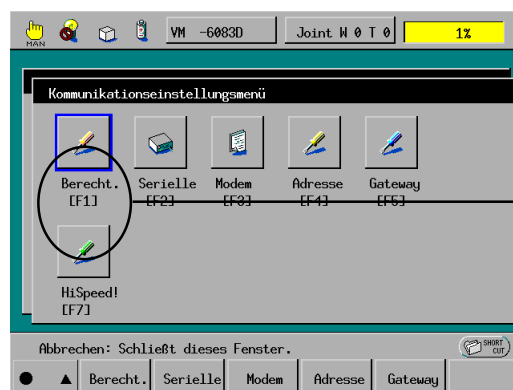
F6

① Drücken Sie [F6 Einstellen].



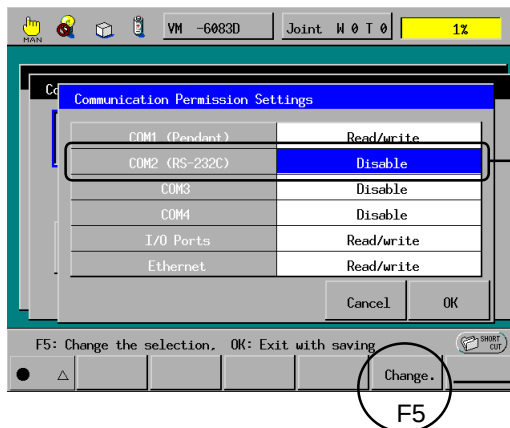
F5

② Drücken Sie [F5 Einst. Komm].



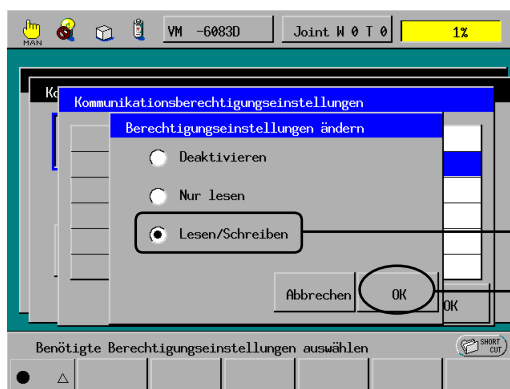
F1

③ Drücken Sie [F1 Berecht.].



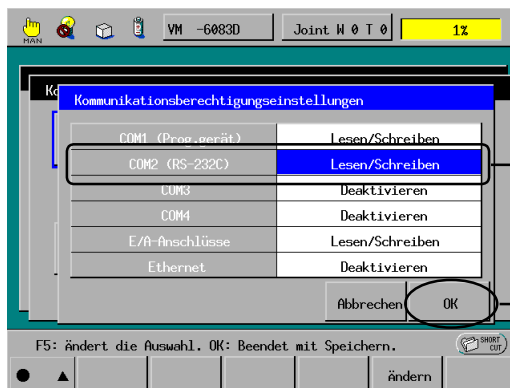
④ Wählen Sie die Zeile [COM2 (RS232C)].

⑤ Drücken Sie [F5 Ändern.]



⑥ Wählen Sie [Lesen/Schreiben].

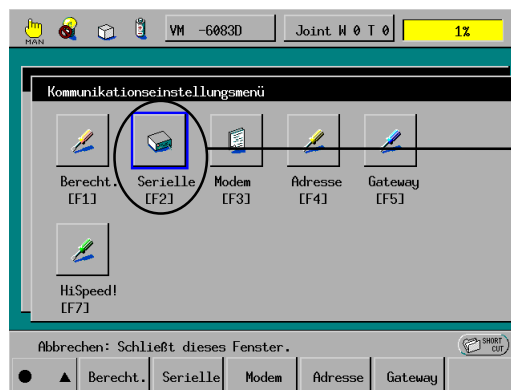
⑦ Drücken Sie [OK].



Die Spalte [COM2 (RS232C)] wird auf [Lesen/Schreiben] geändert.

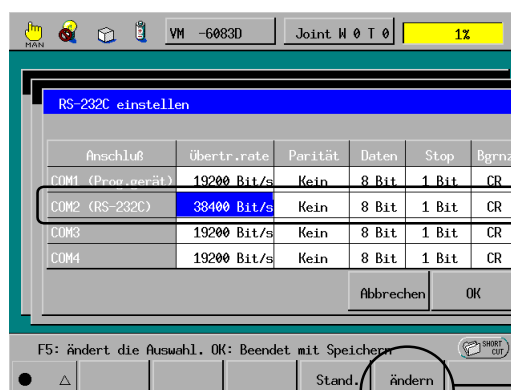
⑧ Drücken Sie [OK].

SCHRITT 2 Einstellung der Übertragungsrate



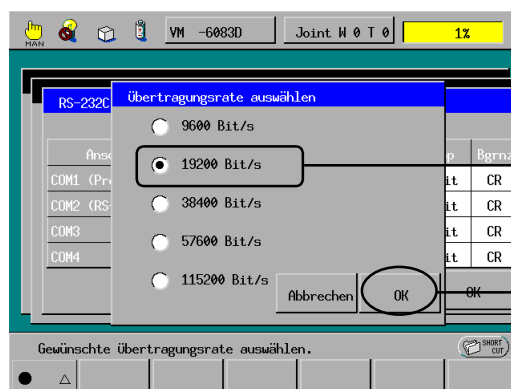
① Drücken Sie [F2 Serielle].

F2



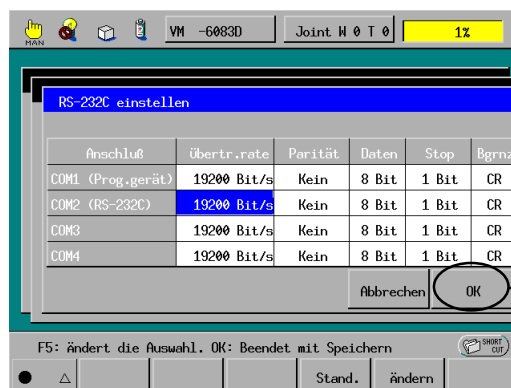
② Wählen Sie die Zeile [COM2 (RS232C)].

③ Drücken Sie [F5 Ändern.]



④ Wählen Sie [19200 BPS] aus.

⑤ Drücken Sie [OK].



⑥ Drücken Sie [OK].



⑦ Drücken Sie [Abbrechen].
(Die Anzeige kehrt zum
[Kommunikationseinstellungsme
nü] zurück.)

⑧ Drücken Sie [Abbrechen].
Der Benutzer kehrt zur
Hauptanzeige zurück.)

Lektion 4 Starten von WINCAPSII und Erstellen eines Systemprojektes

In dieser Lektion werden Sie WINCAPSII auf einem PC starten und ein neues Systemprojekt registrieren. Dies ist erforderlich, um das Programm einzugeben, zu bearbeiten und zu überprüfen. Sie werden auch Einstellungen am Kommunikationsanschluß des PCs vornehmen.

4.1 Starten des System Managers

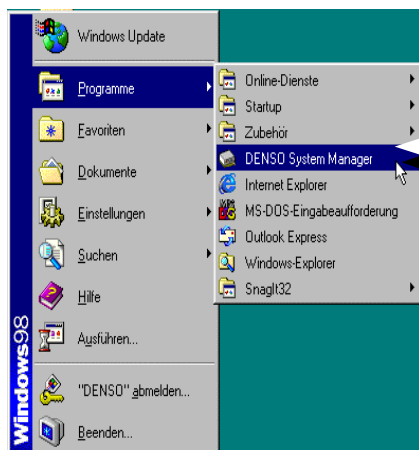
WINCAPSII besteht aus folgenden Funktionsmodulen:

- PAC-PROGRAMM-MANAGER
- Variable Manager
- DIO Manager
- Arm Manager
- Vision Manager
- Log Manager
- Kommunikationseinstellungs-Manager

Der System Manager ermöglicht die Gesamtkontrolle über diese Funktionsmodule. Alle Funktionen von WINCAPSII können vom System Manager aus aufgerufen werden.

Zum Einsetzen des PC-Programmierungssystems müssen Sie zunächst den System Manager mit folgenden Schritten starten:

SCHRITT 1 Auswählen des DENSO System Managers

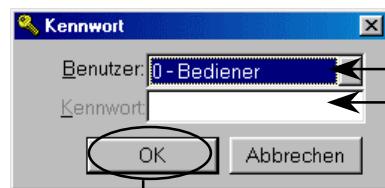


① Rufen Sie im Ordner "DENSO WINCAPSII" über die Schaltfläche "Start" den [DENSO System Manager] auf.

★Vorsicht★

- Wenn Sie den System Manager zum ersten Mal nach der Installation starten, müssen Sie einen Dateinamen für die Programmbank angeben. Das Dialogfeld "Erstellen einer neuen Programmbank" kann ggf. erscheinen. Siehe WINCAPSII-Handbuch, Kapitel 5, Teilabschnitt 5.6.2.3 "Aktualisieren der Programmbank."
- Wird der System Manager zum ersten Mal gestartet, erscheint das Dialogfeld "Neues Projekt erstellen", weil bisher noch kein Systemprojekt definiert wurde. In diesem Fall müssen Sie zuerst "4.2 Registrieren eines neuen Systemprojektes" durchführen, bevor Sie mit Schritt 2 fortfahren.

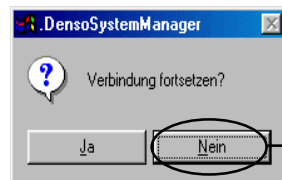
SCHRITT 2 Auswählen der Benutzerebene



① Wählen Sie die Benutzerebene aus dem Popup-Menü aus.

② Geben Sie ein Kennwort ein, falls erforderlich.

③ Drücken Sie [OK].



④ Klicken Sie [Nein] an.

★Punkt★

Weiter unten sehen Sie die Start-Schaltflächen der Funktionsmodule auf der Symbolleiste.



Der System Manager startet und das Fenster [DENSO System Manager] erscheint. Über die Schaltflächen mit Symbolen können Funktionsmodule gestartet werden.

Schaltfläche "Kommunikationseinstellung"

Schaltfläche "Log Manager"

Schaltfläche "Vision Manager"

Schaltfläche "Arm Manager"

Schaltfläche "DIO Manager"

Schaltfläche "Variable Manager"

Schaltfläche "PAC-Programm-Manager"

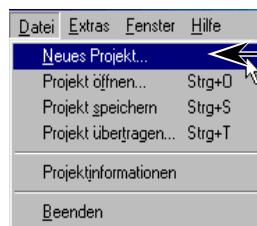
4.2 Registrieren eines neuen Systemprojektes

WINCAPSII steuert in den jeweiligen Projekteinheiten mehr als ein Roboterprogramm. Zur Steuerung eines einzelnen Roboters wird eine Kombination von Programmen verwendet. Es ist daher zweckmäßig, diese Programme als einen Satz für ein Projekt zu steuern.

Zum Erstellen eines Roboterprogrammes müssen Sie zuerst ein neues Projekt registrieren.

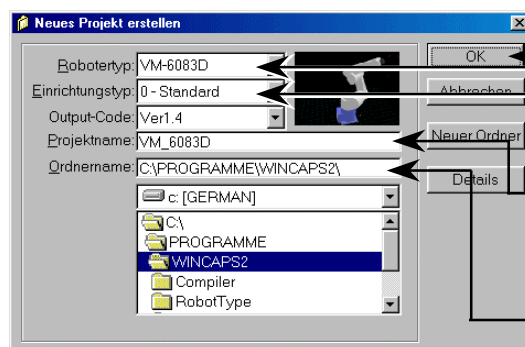
Vorsicht: Wenn Sie den System Manager zum ersten Mal starten, erscheint das Dialogfeld [Neues Projekt erstellen]. Führen Sie zuerst "Neue Projektregistrierung" aus, bevor Sie mit Schritt 2 in "4.1 Starten des System Managers" fortfahren.

SCHRITT 1 Aus dem Menü [Datei] "Neues Projekt" auswählen



① Hier klicken.

SCHRITT 2 Registrieren eines neuen Projektes



① Wählen Sie VM-60B1D.

② Wählen Sie 0-Standard.

③ Geben Sie den gewünschten Projektnamen ein.
(Geben Sie "IRA-000" bei diesem Beispiel ein.)

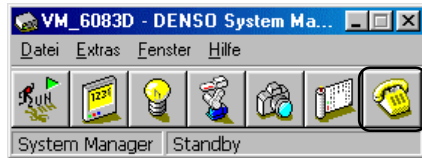
④ Geben Sie den Ordnernamen an.

⑤ Klicken Sie [OK] an.

4.3 Einstellung des Kommunikationsanschlusses eines PCs

Nehmen Sie in WINCAPSII die gleichen Kommunikationseinstellungen vor wie beim Roboter-Controller.

SCHRITT 1 Aufrufen des Dialogfeldes [ROBOTalk Manager]

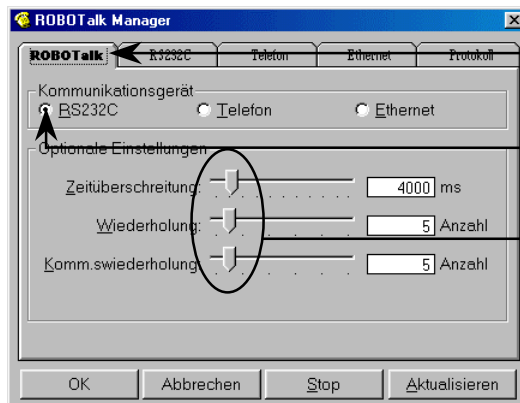


① Klicken Sie die Schaltfläche [Kommunikationseinstellung] an.

★Vorsicht★

Wenn Sie noch kein Kennwort eingegeben haben, erscheint die Eingabemaske [Kennwort]. Sie müssen die Benutzerebene auswählen und ein Kennwort eingeben. (Siehe "4.1 Starten des System Managers".)

SCHRITT 2 Einstellung des Kommunikationsgeräts und der optionalen Einstellungen



② Klicken Sie das Register [ROBOTalk] an.

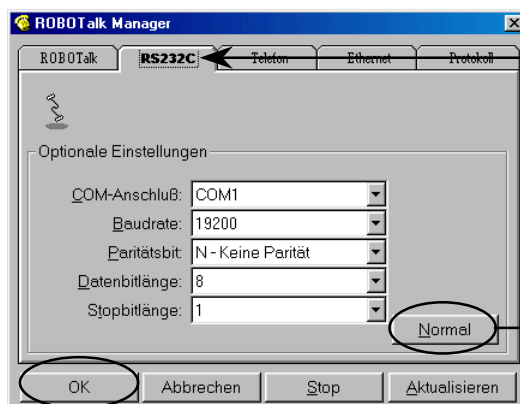
③ Hier klicken, um RS232C auszuwählen.

④ Nehmen Sie optionale Einstellungen über diese Schalter vor.
Legen Sie bei diesem Beispiel die Zeitüberschreitung auf 4000 msec, die Wiederholung auf 5 Mal und die Kommunikationswiederholung

★Vorsicht★

Wenn es zu einer Zeitüberschreitung bei der Datenübertragung kommt, stellen Sie die Zeitüberschreitung auf einen längeren Zeitraum ein.

SCHRITT 3 Einstellung der RS232C-Kommunikationsoptionen



⑤ Klicken Sie das Register [RS232C] an.

⑥ Klicken Sie [Normal] an.
Das Feld "Optionale Einstellungen" wird auf "Normale Einstellungen" geändert.

⑦ Klicken Sie [OK] an.
Die Einstellungen werden aktiv und der [ROBOTalk Manager] wird geschlossen.

★**Vorsicht**★ Wenn Sie andere als die normalen Einstellungen vornehmen, stellen Sie sicher, daß diese den Spezifikationen des verwendeten PCs oder Roboter-Controllers entsprechen.

★**Vorsicht**★ Wenn [OK] deaktiviert ist, setzen Sie die Schaltfläche [Verbinden]  für alle Manager auf AUS. Wenn einer der Manager angeschlossen ist, können die Kommunikationseinstellungen nicht geändert werden.

Sobald sie einmal eingestellt wurden, bleiben die Kommunikationseinstellungen in WINCAPSII aktiv, bis Sie diese erneut ändern. Sie müssen nicht bei jedem Start von WINCAPSII diese Einstellungen vornehmen. Klicken Sie als Antwort auf die Dialogmeldung [Verbindung fortsetzen], die beim Start von WINCAPSII erscheint, einfach [Ja] an. Wenn Sie [Nein] anklicken, werden keine automatischen Einstellungen vorgenommen.



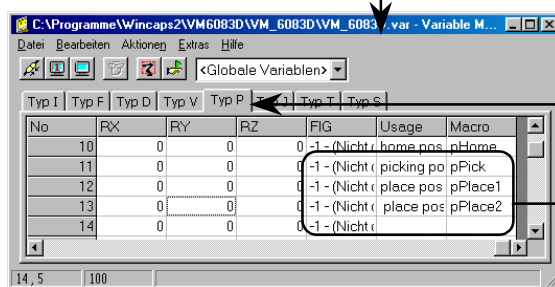
Lektion 5 Definieren von Makros

In dieser Lektion erstellen Sie Makrodefinitionsdateien, indem Sie Namen, Anwendungen von Variablen und E/As definieren.

SCHRITT 1 Erstellen einer Variablen-Makrodefinitionsdatei



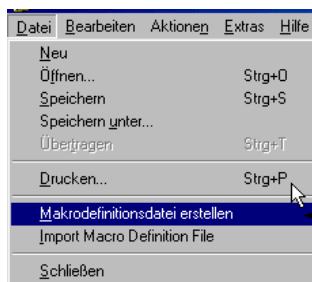
- ① Klicken Sie die Schaltfläche "Variable Manager" an, um den "Variable Manager" zu starten.



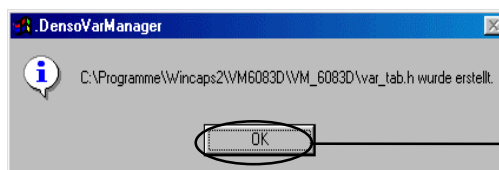
Das Fenster [Variable Manager] erscheint.

- ② Klicken Sie das Register [Typ P] an.

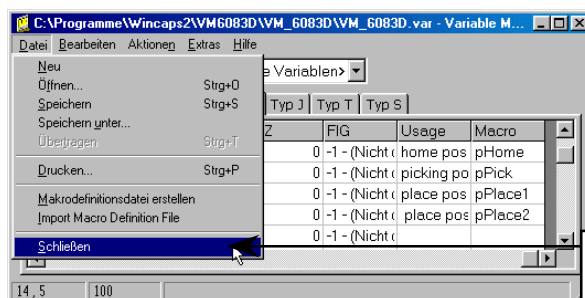
- ③ Doppelklicken Sie auf jedes Feld, und geben Sie die Verwendung und den Makronamen der Variablen an. Geben Sie bei diesem Beispiel die Positionsvariablen P10 bis P13 an.



- ④ Klicken Sie [Makrodefinitionsdatei erstellen] an.

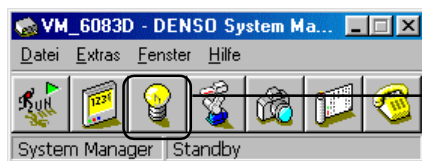


- ⑤ Drücken Sie [OK].
Die Makrodefinitionsdatei ist jetzt erstellt.

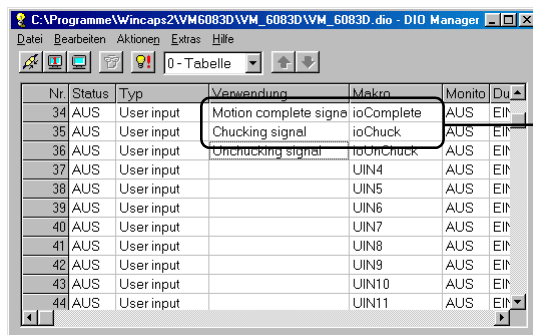


- ⑥ Hier klicken, um den Variable Manager zu verlassen.

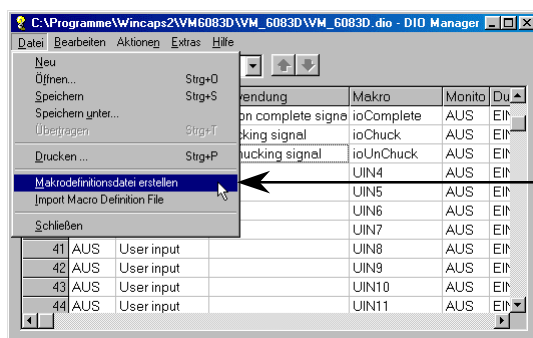
SCHRITT 2 Erstellen einer E/A-Makrodefinitionsdatei



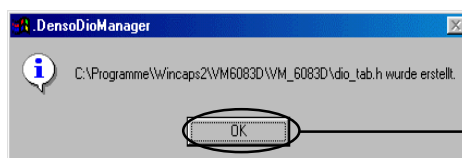
① Klicken Sie die Schaltfläche "DIO Manager" an, um den "DIO Manager" zu starten.



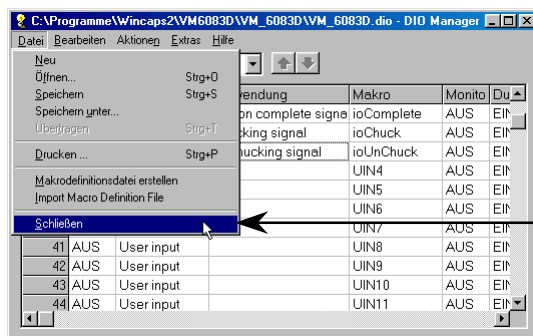
② Doppelklicken Sie auf jedes Feld, und geben Sie die Verwendung und den Makronamen der E/As an.



③ Klicken Sie [Makrodefinitionsdatei erstellen] an.



④ Drücken Sie [OK].
Die E/A-Makrodefinitionsdatei ist jetzt erstellt.



⑤ Klicken Sie [Schließen] an, um den DIO Manager zu verlassen.

Lektion 6 Eingeben und Bearbeiten von Programmen

6.1 Musterprogramm

Prüfen Sie vor dem Start der Programmeingabe-Prozedur das Codierungslisten-Muster unten.

Zum besseren Verständnis der Bewegung lesen Sie die Angaben zum Prozeß und beachten dabei die Kommentare rechts.

Codierungsliste für "PRO1"

```
'!TITLE "Pick & Place"
#INCLUDE "dio_tab.h"           'Liest die DIO-Makrodefinitionsdatei.
#INCLUDE "var_tab.h"          'Liest die Variablen-Makrodefinitionsdatei.

PROGRAM pro1
    TAKEARM                    'Erfäßt den Armsemaphor.
    SET IO[ioComplate]
    MOVE P, P[pHome], S=50     'Punkt-zu-Punkt-Bewegung zur Ausgangsposition mit
                                '50% der internen Geschwindigkeit
    SPEED 100                  'Wechseln zu 100% Geschwindigkeit.
    APPROACH P,P[pPick],200
    MOVE P,P[pPick]
    GOSUB *ChuckItem
    DEPART P,200
    APPROACH P,P[pPlace1],200
    MOVE P,P[pPlace1]
    GOSUB *UnchuckItem
    DEPART P,200
    SET IO[ioComplate]         'Gibt ein Signal "Bewegung abgeschlossen" aus.
    GIVEARM                    'Gibt den Armsemaphor frei.
    END

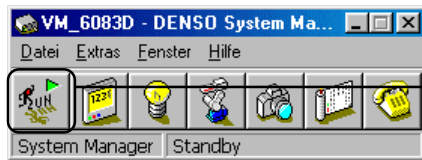
' ===== Parts chuck =====
*ChuckItem:
SET IO[ioChuck]
RESET IO[ioUnChuck]
RETURN

' ===== Parts chuck =====
*UnchuckItem:
RESET IO[ioChuck]
SET IO[ioUnChuck]
RETURN
```

6.2 Öffnen eines Programmbearbeitungsfensters

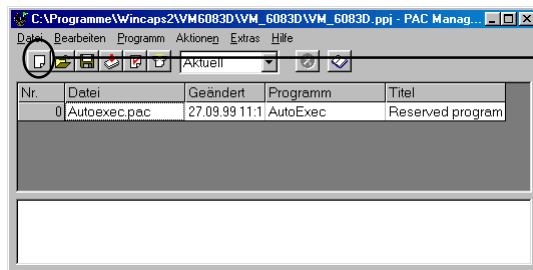
Zur Eingabe und Bearbeitung von Task-Programmen verwenden Sie den Programm-Manager, der über den System Manager aufgerufen wird.

SCHRITT 1 Starten des PAC-Programm-Managers

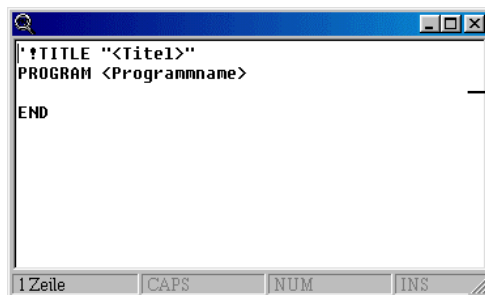


① Klicken Sie die Schaltfläche "PAC-Programm-Manager" an, um den PAC-Programm-Manager zu starten.

SCHRITT 2 Öffnen eines Programmbearbeitungsfensters



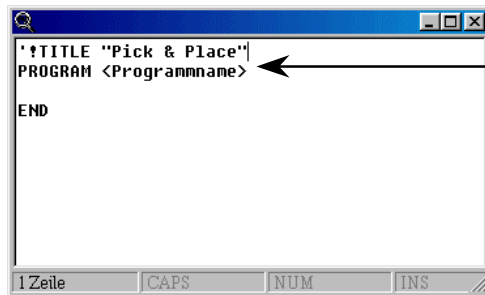
② Klicken Sie die Schaltfläche [Neues Programm] an.



Ein neues Bearbeitungsfenster wird geöffnet.

6.3 Eingeben von Programmcodes

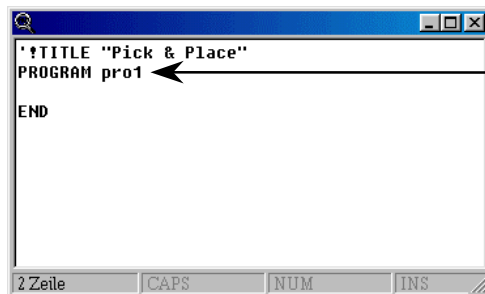
SCHRITT 1 Geben Sie den Programmtitel ein



```
'!TITLE "Pick & Place"
PROGRAM <Programmname>
END
```

① Geben Sie den Programmtitel ein.
(Geben Sie bei diesem Beispiel
"Aufnehmen & Ablegen" ein.)

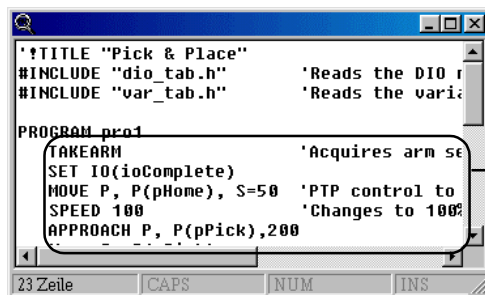
SCHRITT 2 Geben Sie den Programmnamen ein



```
'!TITLE "Pick & Place"
PROGRAM pro1
END
```

② Geben Sie den Programmnamen
ein. (Geben Sie "pro1" bei diesem
Beispiel ein.)

SCHRITT 3 Eingeben der Programmcodes



```
'!TITLE "Pick & Place"
#include "dio_tab.h"      'Reads the DIO i
#include "var_tab.h"      'Reads the varia

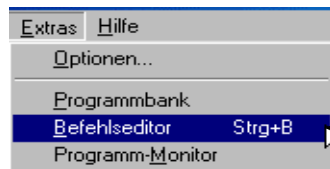
PROGRAM pro1
  TAKEARM                'Acquires arm se
  SET IO(ioComplete)
  MOVE P, P(pHome), S=50 'PTP control to
  SPEED 100               'Changes to 100%
  APPROACH P, P(pPick),200
```

③ Geben Sie die Programmcodes
für "Aufnehmen & Ablegen" ein.

6.4 Einsetzen des Befehlseditors

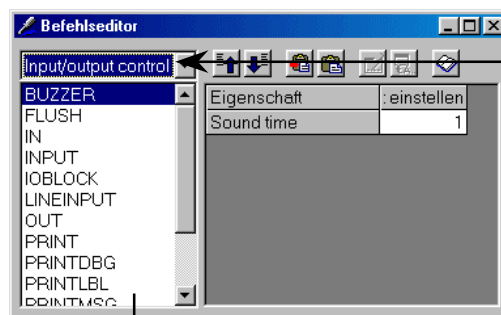
Sie können wie bei einer Textverarbeitung Programmcodes direkt über die Tastatur in das Programmbearbeitungsfenster eingeben. Der Programm-Manager ist jedoch mit einer Befehlseditorfunktion ausgestattet, die eine einfache Befehlseingabe ermöglicht. In diesem Abschnitt wird beschrieben, wie Sie mit dem Befehlseditor Befehle eingeben können.

SCHRITT 1 Auswählen des Befehlseditors



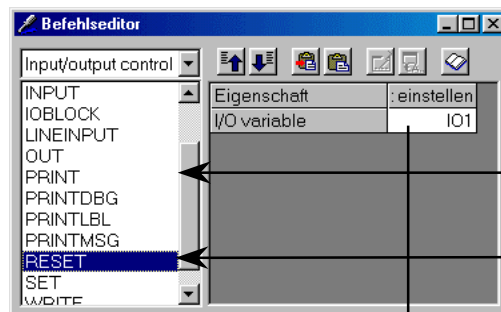
① Wählen Sie aus dem Menü [Extras] den [Befehlseditor] aus.

SCHRITT 2 Einsetzen des Befehlseditors



② Klicken Sie die [Klassenauswahl]-Liste im Fenster [Befehlseditor] an, und wählen Sie aus der Liste "E/A-Steuer" anweisungen.

Befehlsliste



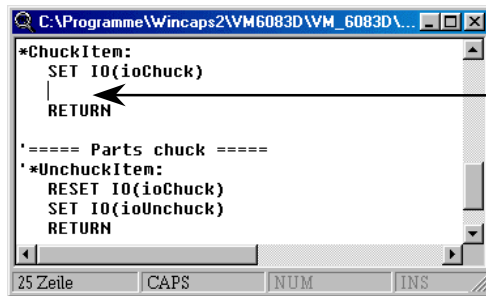
③ Blättern Sie über die Bildlaufleiste der E/A-Steueranweisungen nach unten, um den Befehl [RÜCKSETZUNG] anzuzeigen.

④ Klicken Sie den Befehl [RÜCKSETZUNG] an.

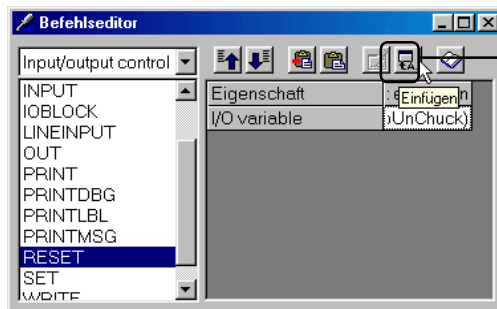
⑤ Doppelklicken Sie auf das Feld [E/A-Variable, Wert setzen].



⑥ Geben Sie über die Tastatur [E/A-Variable, Wert setzen] "ioUnchuck" ein.

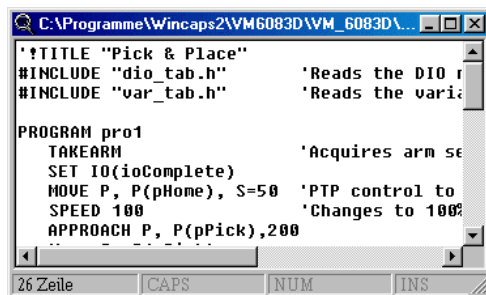


⑦ Klicken Sie im Programmbearbeitungsfenster die Position an, von wo aus Sie Befehle eingeben wollen.



⑧ Klicken Sie die Schaltfläche [Einfügen] an .

SCHRITT 3 Einfaches und klares Bearbeiten von Programmcodes



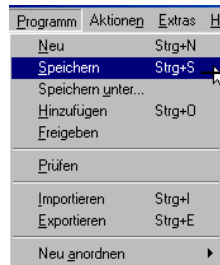
★**Vorsicht**★

Um das bearbeitete Programm zu speichern, fahren Sie mit der im folgenden Abschnitt "6.5 Speichern des Programmes" beschriebenen Prozedur fort.

6.5 Speichern des Programmes

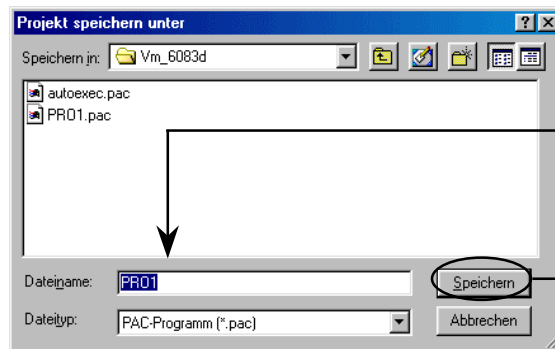
In diesem Abschnitt werden Sie PRO1, das im vorherigen Abschnitt "6.4 Einsetzen des Befehlseditors" erstellte Programm, speichern.

SCHRITT 1 Wählen Sie [Speichern] aus



① Wählen Sie aus dem Menü [Programm] die Option [Speichern].

SCHRITT 2 Eingeben des Dateinamens



② Geben Sie im Feld [Dateiname] einen Dateinamen ein.
Geben Sie bei diesem Beispiel "pro1" als Dateinamen an.
(Die Dateierweiterung wird automatisch angehängt.)

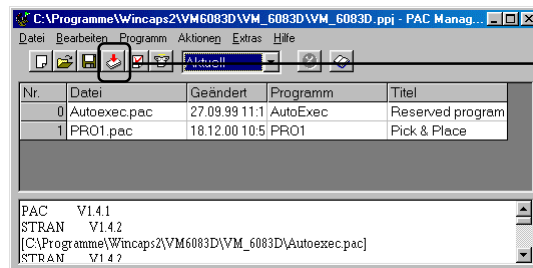
③ Klicken Sie [Speichern] an. Die Programmquelldatei "pro1.pac" wird gespeichert.

Der Dateiname des Fensters [PAC-Manager] wird auf "pro1.pac" geändert. Damit ist die Eingabe- und Speicher-Prozedur des Programmes "pro1" abgeschlossen.

Lektion 7 Kompilieren des Programmes in das Laufzeit-Format

Zum Ausführen eines in der PAC-Sprache geschriebenen Programmes müssen Sie es in das Laufzeit-Format umwandeln (kompilieren), damit es durch den Roboter-Controller ausgeführt werden kann. Das kompilierte Programm wird als Ausführungsprogramm bezeichnet.

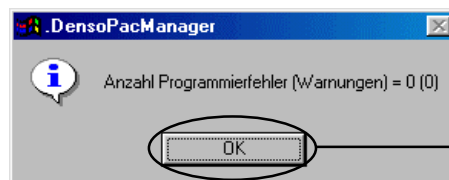
SCHRITT 1 Kompilieren des Programmes in das Laufzeitformat



① Klicken Sie die Schaltfläche [Ausführungsprogramm erstellen] an.

Alle im aktuell ausgewählten Projekt enthaltenen Programme werden in Ausführungsprogramme umgewandelt. Die Aufzeichnung des Kompilierungsprozesses wird im Meldungs-Teilfenster des Fensters [Denso PAC-Manager] angezeigt.

SCHRITT 2 Stellen Sie sicher, daß kein Fehler aufgetreten ist



② Prüfen Sie, ob "Anzahl Programmierfehler (Warnungen) = 0 (0)" angezeigt wird, und klicken Sie [OK] an.

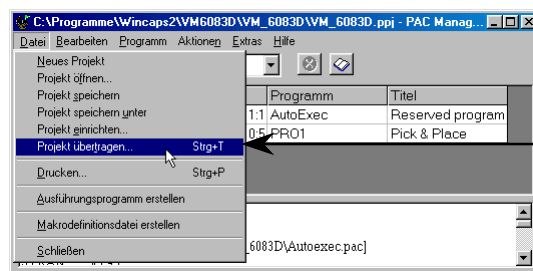
Wenn ein Fehler angezeigt wird, gehen Sie zu "Lektion 6 Eingeben und Bearbeiten von Programmen" zurück, und prüfen Sie auf Syntaxfehler.

Lektion 8 Hochladen des Programmes (PC → Roboter-Controller)

Gegenwärtig befindet sich das in Lektion 7 genannte Ausführungsprogramm immer noch auf dem PC. Um das Programm auszuführen, müssen Sie es zum Roboter-Controller übertragen (hochladen).

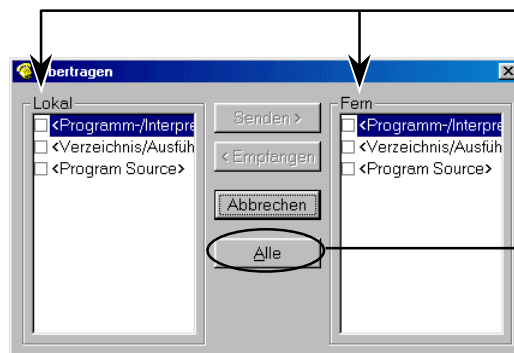
Da Sie bereits in Lektion 3.3 und 4.3 Kommunikationseinstellungen an Roboter-Controller und PC vorgenommen haben, können Sie nun das Programm zum Roboter-Controller hochladen.

SCHRITT 1 Einrichten der Kommunikationsverknüpfung zwischen Programm-Manager und Roboter-Controller



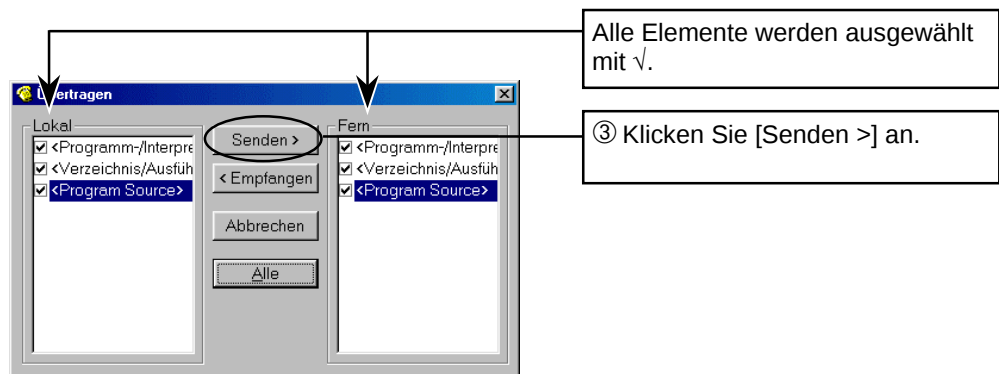
① Wählen Sie aus dem Menü "Datei" die Option [Projekt übertragen.]

SCHRITT 2 Auswählen des hochzuladenden Programmes

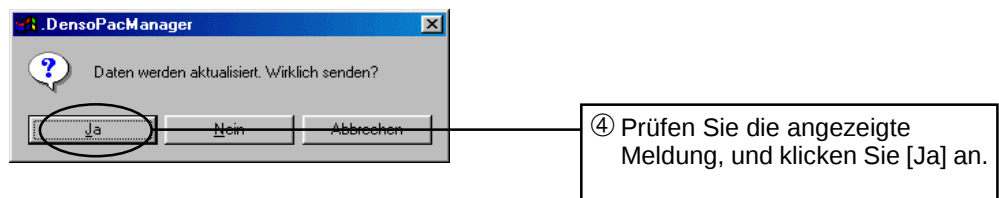


"Lokal" bezieht sich auf den PC und "Fern" auf den Roboter-Controller.

② Klicken Sie [Alle auswählen] an.



SCHRITT 3 Hochladen des ausgewählten Programmes



Die Programmdatei wird nun zum Roboter-Controller hochgeladen.

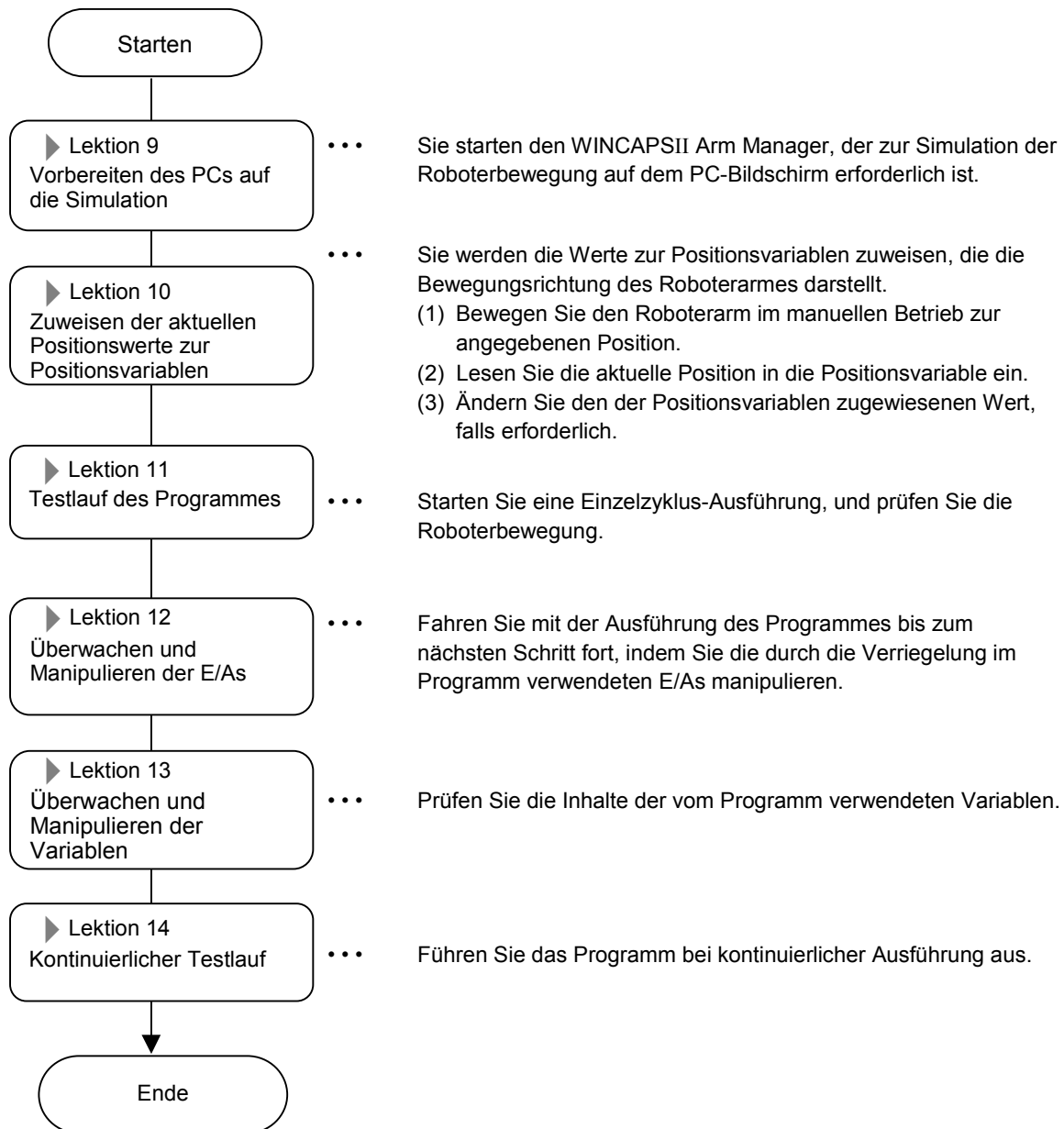
Kapitel 3 Simulieren der Roboterbewegung auf einem PC mit dem erstellten Programm

In Kapitel 3 lernen Sie folgendes:

Führen Sie das Programm, das Sie auf einem PC erstellt und zum Roboter-Controller hochgeladen haben, in der Maschinenverriegelung aus, um die Roboterbewegung auf dem PC-Bildschirm zu simulieren.

Durch die Simulation können Sie das Programm vor der tatsächlichen Steuerung des Roboters überprüfen und dadurch die Sicherheit und Wirksamkeit des entwickelten Programmes erhöhen.

► Lektion 9	Vorbereiten des PCs auf die Simulation	67
9.1	Starten des Arm Managers	67
9.2	Einrichten der Kommunikationsverknüpfung zum Roboter-Controller im Arm Manager	67
► Lektion 10	Zuweisen der aktuellen Positionswerte zu den Positionsvariablen	68
10.1	Manuelles Simulieren der Roboterbewegung	68
10.2	Abrufen der aktuellen Position in der Positionsvariablen	70
10.3	Bearbeiten der Positionsvariablen	76
► Lektion 11	Testlauf des Programmes	77
11.1	Laden des Programmes	77
11.2	Starten des Programmes	79
► Lektion 12	Überwachen und Manipulieren der E/As	80
12.1	Starten des DIO Managers und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller	80
12.2	Überwachen der E/As	81
12.3	Ein-/Ausschalten der E/A-Dummy-Schalter	82
► Lektion 13	Überwachen und Manipulieren der Variablen	83
13.1	Starten des Variablen Managers und Herstellen der Kommunikation	83
13.2	Überwachen der Variablen	84
► Lektion 14	Kontinuierlicher Testlauf	85
14.1	Kontinuierliche Ausführung	85
14.2	Kontinuierliche Überwachung der E/As	86
14.3	Sofortiger Stop (Anhalten)	88



Lektion 9 Vorbereiten des PCs auf die Simulation

Sie starten den Arm Manager und stellen die Kommunikationsverknüpfung zum Roboter-Controller her.

9.1 Starten des Arm Managers

Sie müssen den Arm Manager starten, um die simulierten Roboter-Bilder anzuzeigen. Der Arm Manager wird über den System Manager aufgerufen.

SCHRITT 1

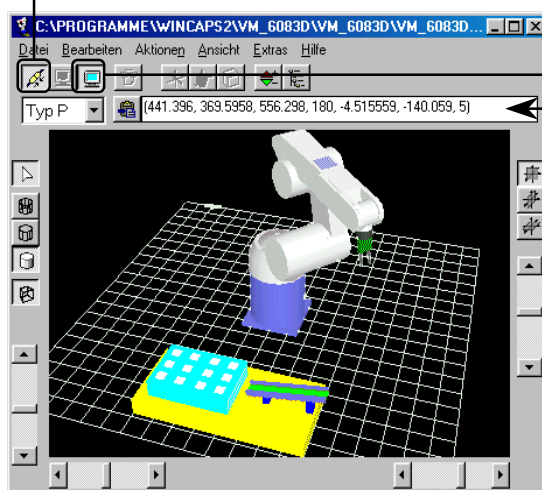


① Klicken Sie die Schaltfläche [Arm] an.

9.2 Einrichten der Kommunikationsverknüpfung zum Roboter-Controller im Arm Manager

Sie stellen die Kommunikationsverknüpfung zum Roboter-Controller her, um den Datenaustausch zwischen PC und Roboter-Controller zu ermöglichen.

SCHRITT 1



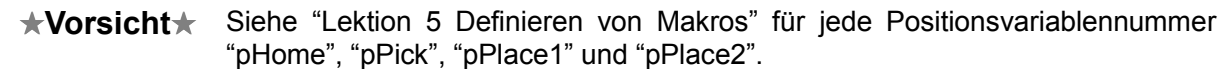
① Klicken Sie die Schaltfläche [Verbinden] an.
(Die Schaltfläche [Verbinden] wird gedrückt dargestellt.)

② Klicken Sie die Schaltfläche [Monitor] an.
(Die Schaltfläche [Monitor] wird gedrückt dargestellt.)

Die Kommunikationsverknüpfung zwischen Arm Manager und Roboter-Controller ist hergestellt, um den Datenaustausch zwischen beiden zu ermöglichen.

Im Fenster [Aktuelle Roboterposition] wird die aktuelle Roboterposition angezeigt.

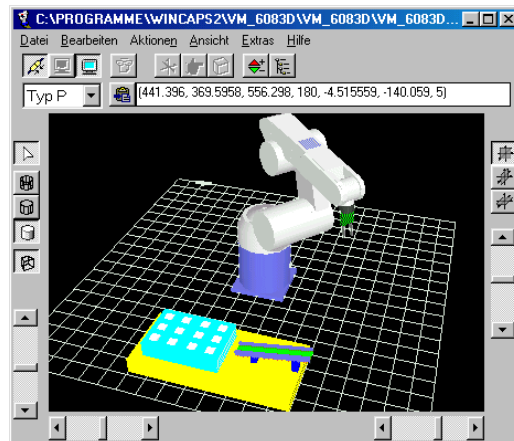
Bei dieser Lektion werden Sie die Werte des Programmiergeräts über “Punkt-Programmierung” eingeben.



Während Sie die im Arm Manager angezeigten Simulationsbilder überwachen, bewegen Sie entsprechend der unten beschriebenen Prozedur den Roboterarm manuell zu den der Positionsvariablen "pHome" zugewiesenen Positionswerten.

- 68

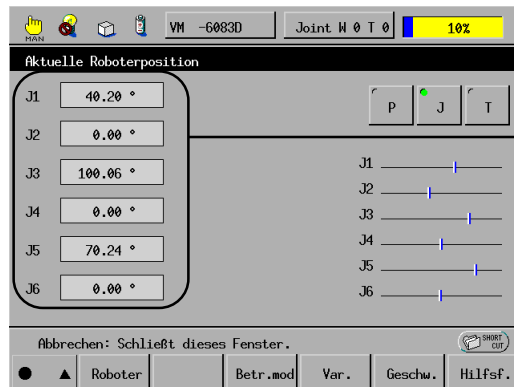
SCHRITT 2 Bewegen des Roboterarmes



④ Während Sie das Roboter-Simulationsbild überwachen, bewegen Sie den Roboterarm über den Sicherheitsschalter und die Tasten der Armbewegung seitwärts so zu pHome, damit folgende Werte gelten:

- J1: Ca. 40°
- J2: 0° (keine Bewegung)
- J3: Ca. 100°
- J4: 0° (keine Bewegung)
- J5: Ca. 70°
- J6: 0° (keine Bewegung)

SCHRITT 3 Prüfen der aktuellen Position



⑤ Prüfen Sie im Fenster [Aktuelle Roboterposition] die Position jeder Achse.

10.2 Abrufen der aktuellen Roboterposition in der Positionsvariablen

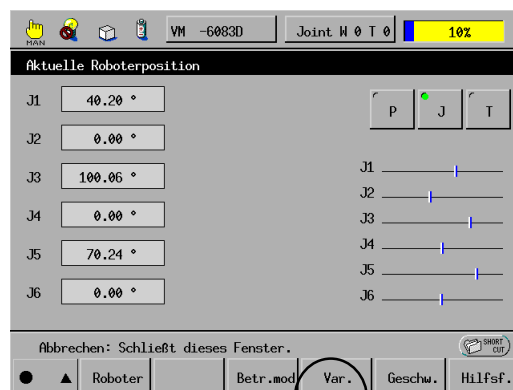
Rufen Sie in der Positionsvariablen die aktuelle Position ab, zu der der Roboterarm im Arm Manager bewegt wurde.

Es gibt folgende 3 Möglichkeiten, um die aktuelle Position in der Positionsvariablen abzurufen:

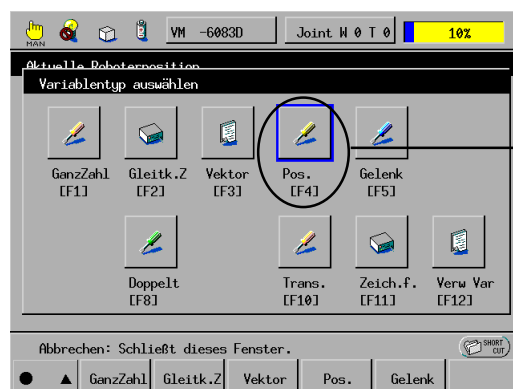
- [1] Über das Fenster [Positionsvariablen] auf dem Programmiergerät
- [2] Über das Programmbearbeitungsfenster (Codierungsliste) auf dem Programmiergerät, wo Sie die aktuelle Position in der im angegebenen Programm festgelegten Positionsvariablen abrufen können
- [3] Über das Bedienfeld

[1] Über das Fenster [Positionsvariablen] auf dem Programmiergerät

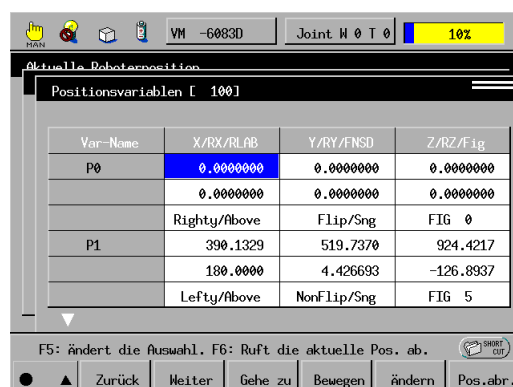
SCHRITT 1 Aufrufen des Fensters [Positionsvariablen]



① Drücken Sie [F4 Var.] bei angezeigtem Fenster [Aktuelle Roboterposition].

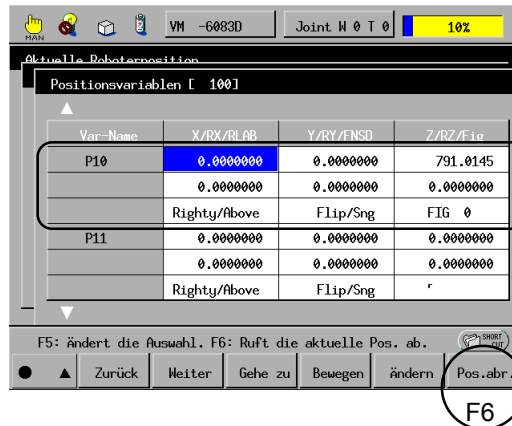


② Drücken Sie im Fenster [Variablentyp auswählen] [F4 Pos.].



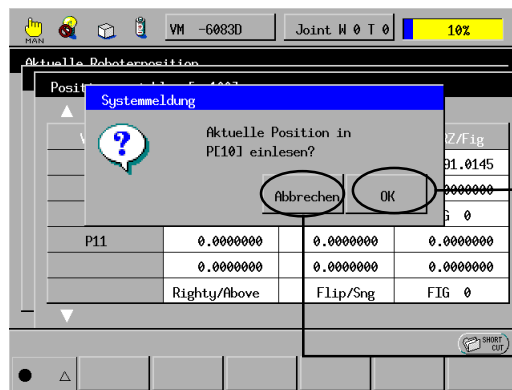
Das Fenster [Positionsvariablen] erscheint.

SCHRITT 2 Abrufen der aktuellen Position in der Positionsvariable



③ Über die Cursor-Tasten oder das Einstellrad wählen Sie das Feld P10 aus.

④ Drücken Sie [F6 Pos. abbr.].

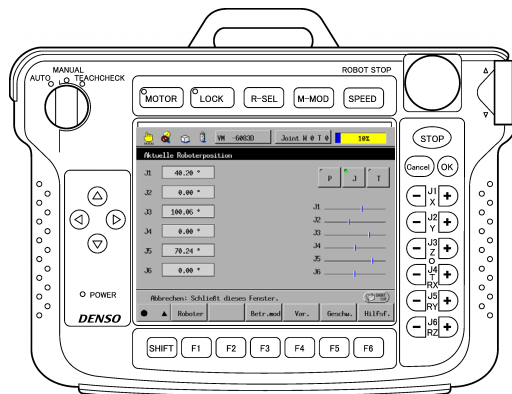


⑤ Drücken Sie [OK]. Dadurch wird die aktuelle Position des Roboterarmes in der Positionsvariablen P[10] abgerufen.

⑥ Drücken Sie zweimal [Abbrechen]. Es erscheint wieder das Fenster [Aktuelle Roboterposition].

⑦ Gehen Sie zurück zu Schritt 2 Bewegen des Roboterarmes in Lektion 10.1.

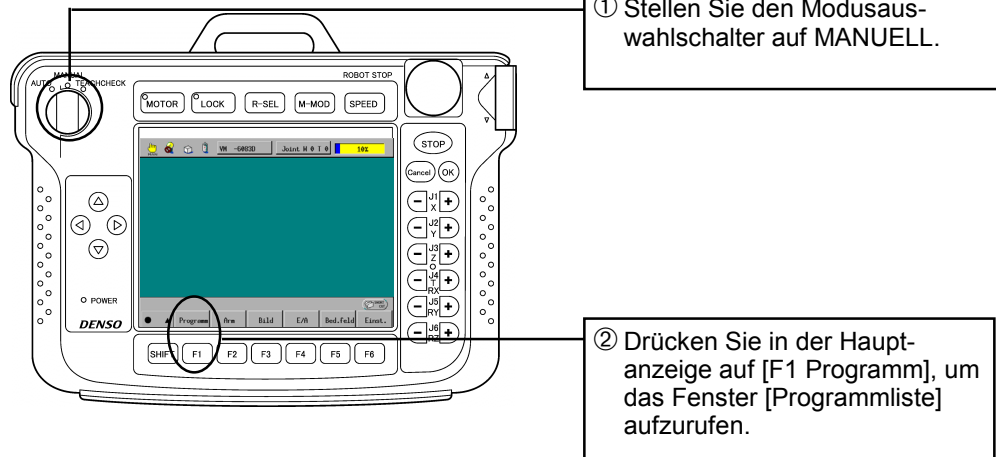
SCHRITT 3 Abrufen sonstiger Armstellungen in den Positionsvariablen



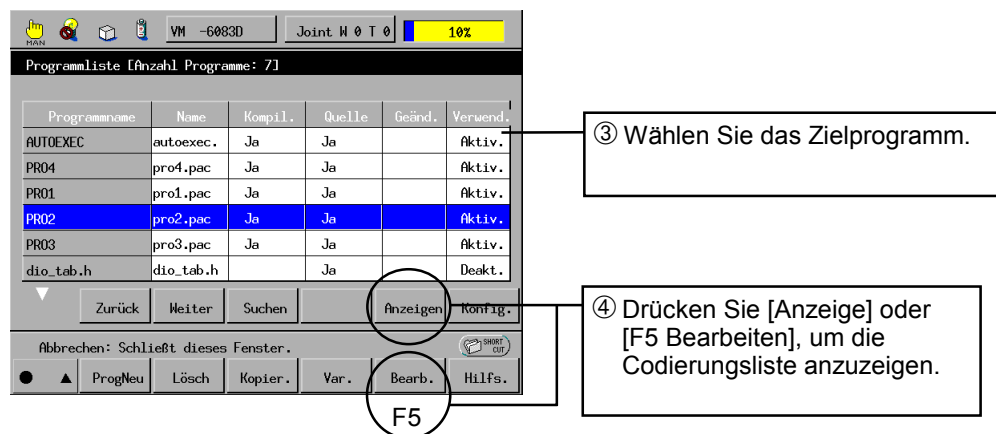
Rufen Sie sonstige Armstellungen ab wie in Schritt 1 und 2 beschrieben.

[2] Über das Programmbearbeitungsfenster (Codierungsliste) auf dem Programmiergerät, wo Sie die aktuelle Position in der im angegebenen Programm festgelegten Positionsvariablen abrufen können

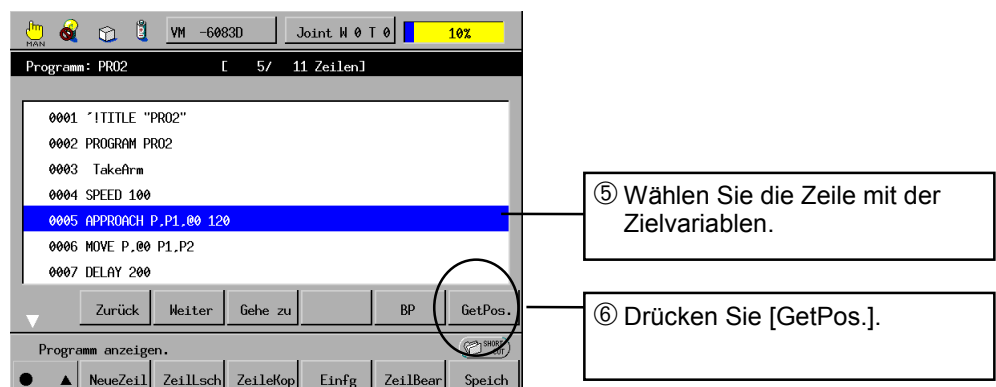
SCHRITT 1 Aufrufen des Fensters "Programmliste"



SCHRITT 2 Auswählen des Zielprogrammes

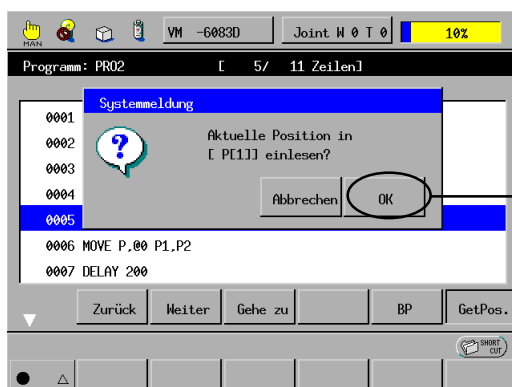


SCHRITT 3 Auswählen der Programmzeile mit der Zielvariablen



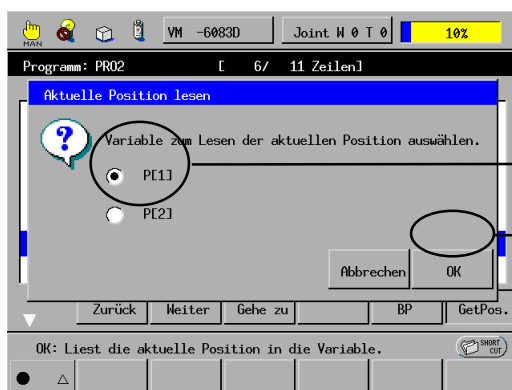
SCHRITT 4 Abfragen der aktuellen Position in der Variablen in der Zeile

Wenn die Programmzeile eine einzige Variable enthält, erscheint die Systemmeldung wie unten dargestellt.



⑦ Drücken Sie [OK]. Dadurch wird die aktuelle Position in der Variablen abgerufen.

Wenn die Programmzeile mehr als eine Variable enthält, erscheint die Systemmeldung wie unten dargestellt. Wählen Sie die gewünschte Variable aus.



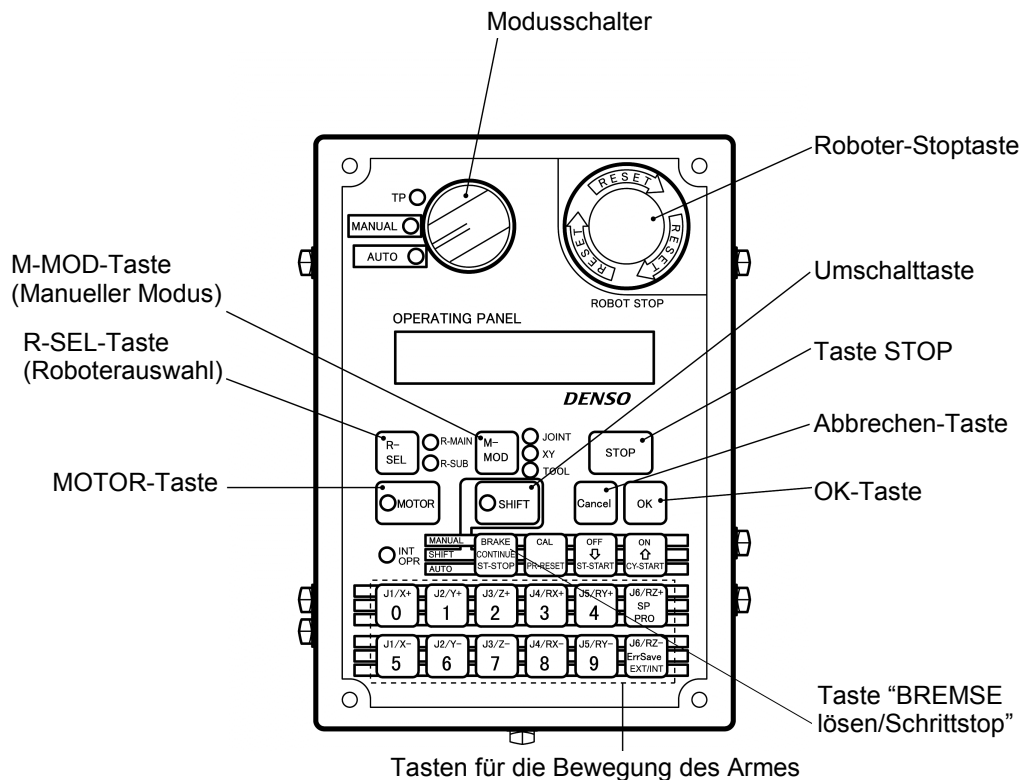
⑧ Wählen Sie mit den Cursor-Tasten oder dem Einstellrad die gewünschte Variable.

⑨ Drücken Sie [OK]. Dadurch wird die aktuelle Position in der ausgewählten Variablen abgerufen.

Wenn die in der ausgewählten Programmzeile stehende(n) Variable(n) nicht korrekt sind, werden keine Variablen für die Programmierung angezeigt.

Nur die ersten 3 Variablen der Programmzeile werden angezeigt.

[3] Über das Bedienfeld



SCHRITT 1 Wechseln zum manuellen Modus

Stellen Sie den Modus-Schalter auf manuelle Position, und prüfen Sie, ob die LED-Anzeige aufleuchtet.

SCHRITT 2 Aufrufen des Funktionsauswahl-Modus

Drücken Sie die Umschalt-Taste und prüfen Sie, ob die LED-Anzeige aufleuchtet.

Dann drücken Sie die Taste "Bremse lösen/Schrittstop", damit das Bedienfeld in den Funktionsauswahlmodus wechselt.

SCHRITT 3 Auswählen der Funktion "Position abrufen"

Wählen Sie die gewünschte Funktion über die Cursor-Tasten $\uparrow$ und $\downarrow$.

So wird die aktuelle Position abgerufen: Wählen Sie die Option:

Positionsvariable	[F7: Set CurPos P]
Gelenkvariable	[F8: Set CurPos J]
Transformationsvariable	[F9: Set CurPos T]

Nach Auswählen der gewünschten Funktion drücken Sie die OK-Taste.

SCHRITT 4 Auswählen einer Zielvariablennummer

Verwenden Sie die Zifferntasten zum Eingeben einer Variablennummer, über die Sie die aktuelle Position abrufen möchten.

Drücken Sie die Taste "OK". (Um die eingegebene Zahl rückgängig zu machen, drücken Sie die Taste "Abbrechen". Die LCD-Anzeige kehrt zu Schritt 3 zurück.)

SCHRITT 5 Abrufen der aktuellen Position

Die LCD-Anzeige zeigt [SetCurrentPos?] an.

Drücken Sie die Taste "OK". (Drücken Sie die Taste "Abbrechen", um die Operation abzubrechen. Die LCD-Anzeige kehrt zu Schritt 4 zurück.)

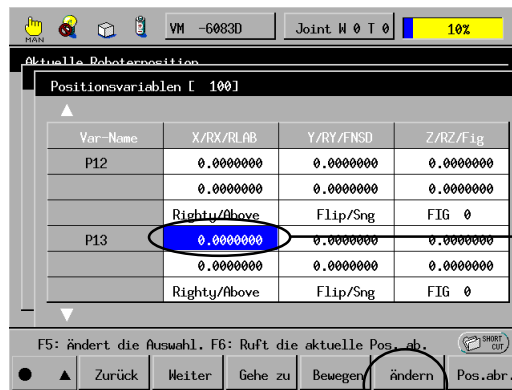
Ist das Abrufen der aktuellen Position abgeschlossen, zeigt die LCD-Anzeige [Set VarVal OK] an.

10.3 Bearbeiten der Positionsvariablen

Sie können Positionsvariablen bei Bedarf bearbeiten. Dazu ein einfaches Beispiel: Sie werden den einer Positionsvariablen zugewiesenen Wert ändern.

Auf dem "Programmiergerät" wird das Fenster "Positionsvariable" angezeigt wie unten dargestellt.

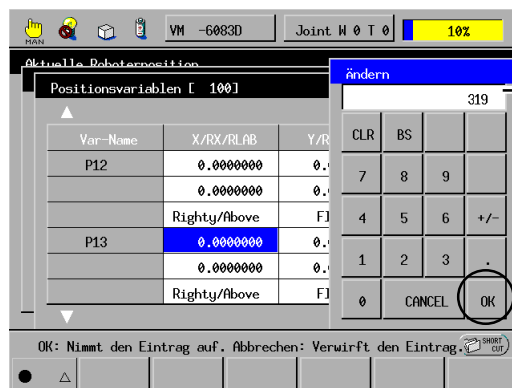
SCHRITT 1 Auswählen der Positionsvariablen



① Wählen Sie über das Einstellrad oder die Cursor-Tasten die Spalte [X] der Positionsvariablen P13.

② Drücken Sie [F5 Ändern]. Die nachstehend gezeigte Zehnerkastatur wird angezeigt.

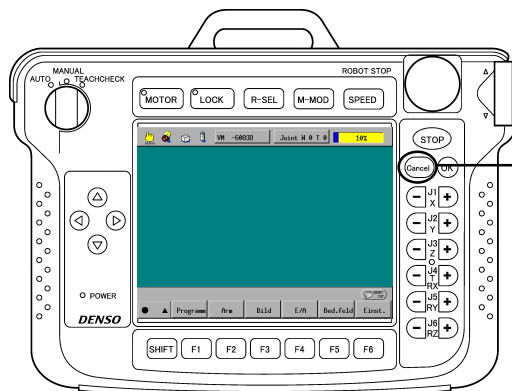
SCHRITT 2 Ändern des "P12" zugewiesenen Wertes



① Geben Sie einen neuen Wert über die Zifferntasten der Zehnerkastatur ein. Wenn nach Eingabe eines zu sehr abweichenden Wertes ein Fehler angezeigt wird, geben Sie den Wert 319 ein, der dem Originalwert sehr nahe kommt.

② Drücken Sie [OK]. 319 wird in der Spalte [X/T] der Positionsvariablen P13 eingegeben.

Zurück zur Hauptanzeige



③ Drücken Sie dreimal die Taste [Abbrechen], um zur Hauptanzeige zurückzukehren. Damit ist die Bearbeitung von Positionsvariablen abgeschlossen.

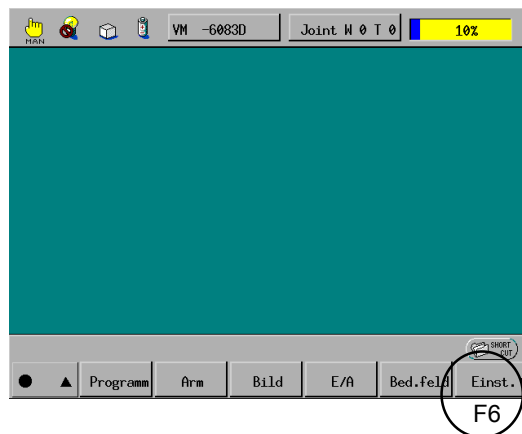
Lektion 11 Testlauf des Programmes

Sie führen einen Testlauf des Programmes "PRO1" in der Einzelzyklusausführung durch.

11.1 Laden des Programmes

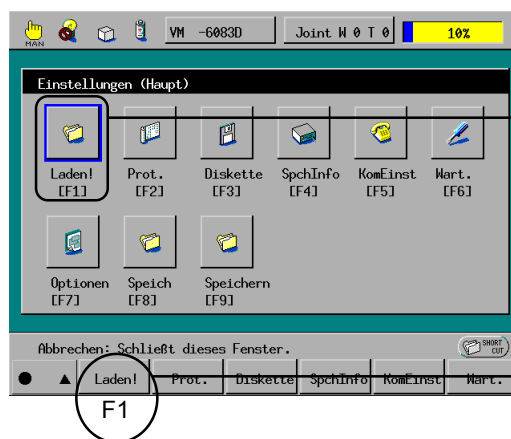
Selbst wenn ein kompiliertes Programm vom PC zum Roboter-Controller hochgeladen wird, kann es der Controller nicht ausführen. Das Programm muß in den Speicherbereich geladen werden, von wo aus es ausgeführt werden kann.

SCHRITT 1 Anzeigen des Lade-Bildschirmes

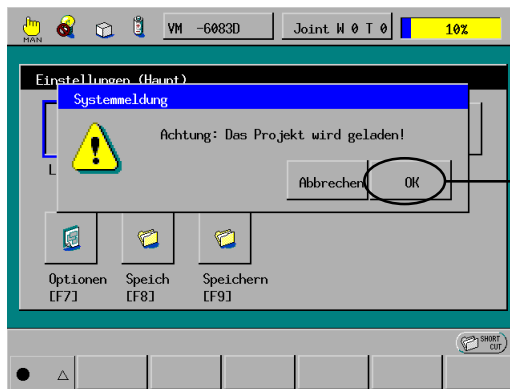


① Drücken Sie [F6 Einstellen] in der Hauptanzeige.

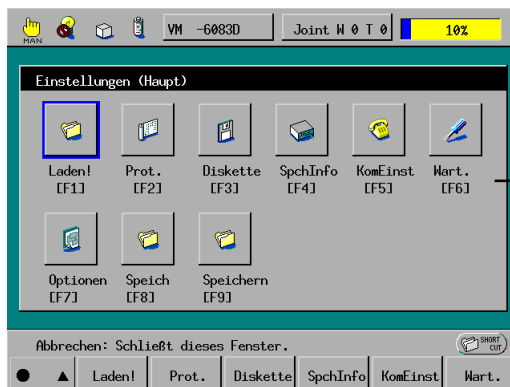
SCHRITT 2 Laden des Projektes



② Drücken Sie entweder die Schaltfläche [F1 Laden!] oder [Laden!].

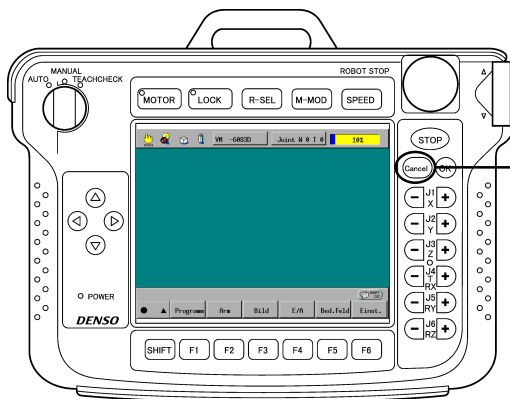


③ Prüfen Sie die Systemmeldung, und drücken Sie [OK].



Nach Abschluß des Ladevorganges kehrt die Bildschirmanzeige zum Fenster [Einstellung (Haupt)] zurück.

Zurück zur Hauptanzeige



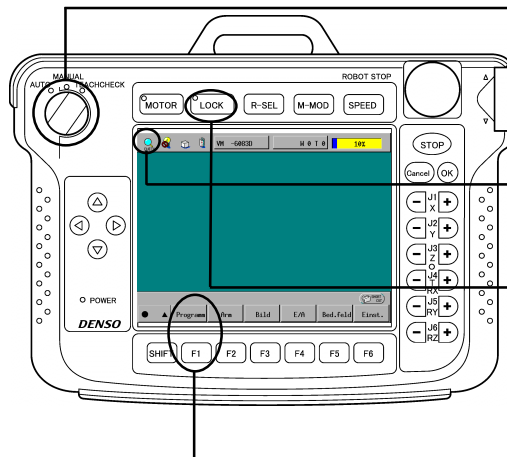
⑤ Drücken Sie [Abbrechen], um zur Hauptanzeige zurückzukehren.

11.2 Starten des Programmes

Jetzt aktivieren Sie die Maschinenverriegelung des Roboter-Controllers und starten das geladene Programm in der Einzelzyklus-Ausführung, um die Roboterbewegung auf dem PC-Bildschirm zu simulieren.

Das Programm wird unterbrochen und wartet dann auf Teilversorgung bestätigende E/A-Signale. Fahren Sie jedoch mit den folgenden Schritten fort, wenn Sie in Lektion 12 E/As manipulieren wollen.

SCHRITT 1 Anzeigen des Fensters "Programmliste" im Programmierprüfmodus



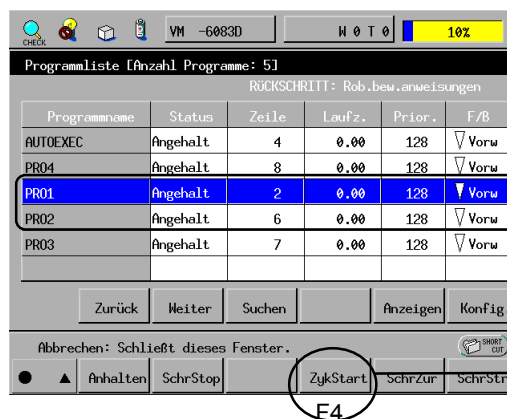
① Stellen Sie den Modusauswahlschalter auf PROGRAMMIERPRÜFUNG. (Der Roboter-Controller wechselt in den Programmierprüfmodus.)

② Prüfen Sie, ob das Modus-Symbol oben links auf dem Bildschirm auf [CHECK] gewechselt

③ Drücken Sie [SPERRE], um die Maschinenverriegelung des Roboters zu aktivieren. Prüfen Sie, ob die LED [SPERRE] eingeschaltet ist.

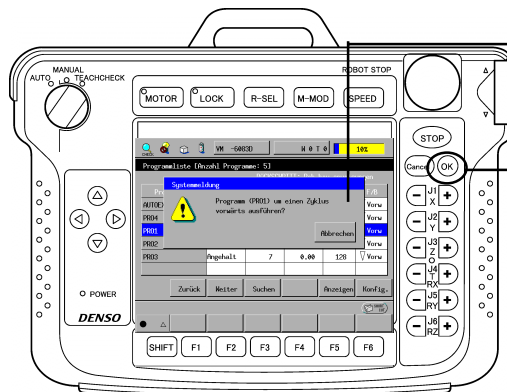
④ Drücken Sie [F1 Programm], um das Fenster [Programmliste] anzuzeigen.

SCHRITT 2 Auswählen des Start-Programmes



⑤ Wählen Sie die Zeile PRO1 aus.

⑥ Drücken Sie [F4 ZykStart].



⑦ Prüfen Sie die Systemmeldung.

⑧ Drücken Sie [OK]. PRO1 führt einen Einzelzyklus aus.

Lektion 12 Überwachen und Manipulieren der E/As

Das im Zyklus gestartete Programm "PRO1" wird nun angehalten und wartet auf Makro-E/A "ioParts" zur Bestätigung der Teileversorgung.

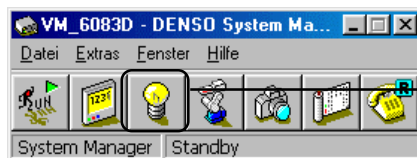
Bei dieser Lektion werden Sie zum Testen des Programmes den E/A-Manager zum Überwachen und Manipulieren von E/As verwenden, um die Programmausführung fortzusetzen.

Anmerkung: Siehe "Lektion 5 Definieren von Makros" für E/A-Nummern von "ioParts".

12.1 Starten des DIO Managers und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller

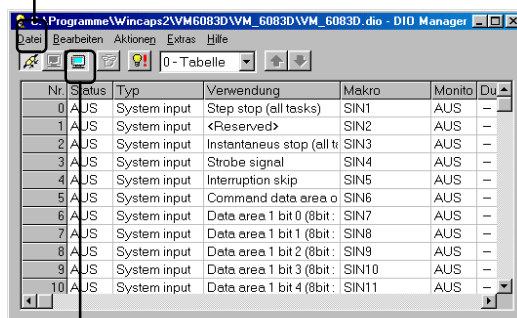
Sie starten den DIO Manager und stellen die Kommunikationsverknüpfung zum Roboter-Controller her, um den Datenaustausch zwischen PC und Roboter-Controller zu ermöglichen.

SCHRITT 1 Starten des DIO Managers



- ① Klicken Sie im Fenster System Manager die Schaltfläche [E/A] an.
Das Fenster [DIO Manager] erscheint.

SCHRITT 2 Verbinden mit dem Roboter-Controller, um die kontinuierliche Überwachung zu starten



- ② Klicken Sie im Fenster DIO Manager die Schaltfläche [Verbinden] an.
Die Schaltfläche [Verbinden] wird gedrückt dargestellt.

- ③ Klicken Sie die Schaltfläche [Monitor] an.
Die Kommunikationsverknüpfung zwischen Arm Manager und Roboter-Controller ist hergestellt, um den Datenaustausch zwischen beiden zu ermöglichen.
Die Schaltfläche [Monitor] wird gedrückt dargestellt.

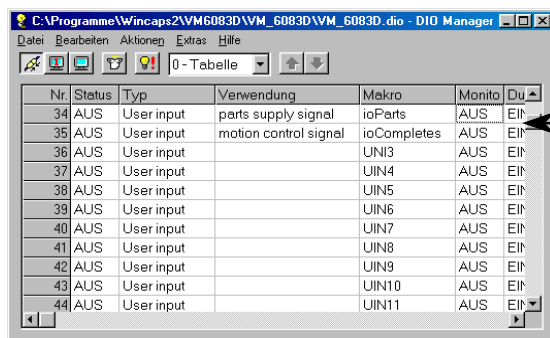
12.2 Überwachen der E/As

Sie werden die E/As mit dem DIO Manager überwachen.

Der DIO Manager kann drei Typen von Anzeigeformaten des E/A-Status darstellen: Tabellentyp, Oszilloskoptyp und Anzeigentyp. Bei dieser Lektion verwenden Sie die Tabellentyp-Anzeige, um "ioParts" an E/A Nr. 34 zu prüfen.

Die Tabellentyp-Anzeige ähnelt einer Liste und erscheint, wenn Sie das Fenster [Dio Manager] öffnen. Wenn ein anderer Anzeigentyp angezeigt wird, wählen Sie aus der Liste [Variablentyp] den Tabellentyp aus.

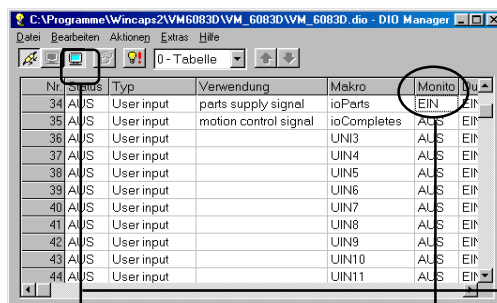
SCHRITT 1 Anzeigen der E/As



Nr.	Status	Typ	Verwendung	Makro	Monitor	Du
34	AUS	User input	parts supply signal	ioParts	AUS	EIN
35	AUS	User input	motion control signal	ioCompletes	AUS	EIN
36	AUS	User input		UIN3	AUS	EIN
37	AUS	User input		UIN4	AUS	EIN
38	AUS	User input		UIN5	AUS	EIN
39	AUS	User input		UIN6	AUS	EIN
40	AUS	User input		UIN7	AUS	EIN
41	AUS	User input		UIN8	AUS	EIN
42	AUS	User input		UIN9	AUS	EIN
43	AUS	User input		UIN10	AUS	EIN
44	AUS	User input		UIN11	AUS	EIN

- ① Blättern Sie im Fenster [DIO Manager] bis zur Anzeige der E/A Nr. 34.
(Verwenden Sie zum Blättern die Bildlaufleiste auf der rechten Seite des Fensters.)

SCHRITT 2 Einstellung des Monitortyps



Nr.	Status	Typ	Verwendung	Makro	Monitor	Du
34	AUS	User input	parts supply signal	ioParts	EIN	EIN
35	AUS	User input	motion control signal	ioCompletes	AUS	EIN
36	AUS	User input		UIN3	AUS	EIN
37	AUS	User input		UIN4	AUS	EIN
38	AUS	User input		UIN5	AUS	EIN
39	AUS	User input		UIN6	AUS	EIN
40	AUS	User input		UIN7	AUS	EIN
41	AUS	User input		UIN8	AUS	EIN
42	AUS	User input		UIN9	AUS	EIN
43	AUS	User input		UIN10	AUS	EIN
44	AUS	User input		UIN11	AUS	EIN

- ② Klicken Sie im Fenster [DIO Manager] die Schaltfläche [Kontinuierlicher Monitor] an, um den Monitor abzuschalten. (Die Satzinhalte können nicht geändert werden, während der Monitor eingeschaltet ist.)

- ③ Doppelklicken Sie das Feld [Monitor] von E/A Nr. 34, das im Fenster [DIO Manager] angezeigt wird.
Die Einstellung im Feld [Monitor] wechselt von AUS auf EIN und ermöglicht nun die Überwachung von E/A Nr. 34.

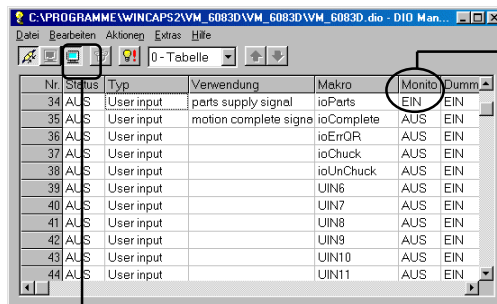
- ④ Klicken Sie die Schaltfläche [Kontinuierlicher Monitor] an, um den Monitor einzuschalten.
Entsprechend der an E/A vorgenommenen Änderungen wird der Status von E/A Nr. 34 als EIN oder AUS angezeigt.
In diesem Beispiel werden die ioParts von E/A Nr. 34 als AUS angezeigt.

12.3 Ein-/Ausschalten der E/A-Dummy-Schalter

Im DIO Manager können Sie die E/As fälschlicherweise ein- oder ausschalten.

Das Programm PRO1 "PICK & PLACE" wartet auf E/A Nr. 34 "ioParts", um sich einzuschalten und fährt bei diesem Schritt nicht in diesem Status fort. Falls Sie E/A nicht korrekt verwaltet haben, können Sie einen Betriebstest durchführen.

SCHRITT 1 Stoppen der Überwachung

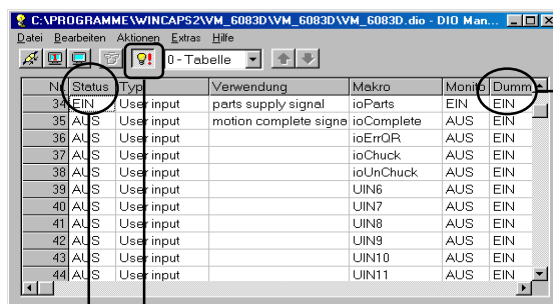


Nr.	Status	Type	Verwendung	Makro	Monitor	Dummy
34	AUS	User input	parts supply signal	ioParts	EIN	EIN
35	AUS	User input	motion complete signal	ioComplete	AUS	EIN
36	AUS	User input		ioErrQR	AUS	EIN
37	AUS	User input		ioChuck	AUS	EIN
38	AUS	User input		ioUnChuck	AUS	EIN
39	AUS	User input		UIN6	AUS	EIN
40	AUS	User input		UIN7	AUS	EIN
41	AUS	User input		UIN8	AUS	EIN
42	AUS	User input		UIN9	AUS	EIN
43	AUS	User input		UIN10	AUS	EIN
44	AUS	User input		UIN11	AUS	EIN

① Prüfen Sie im Fenster [DIO Manager], ob im Feld [Monitor] für E/A Nr. 34 EIN eingestellt ist.

② Klicken Sie hier, um den [Kontinuierlichen Monitor] auf AUS zu setzen.

SCHRITT 2 Einstellung des Dummy-Schalters



Nr.	Status	Type	Verwendung	Makro	Monitor	Dummy
34	EIN	User input	parts supply signal	ioParts	EIN	EIN
35	AUS	User input	motion complete signal	ioComplete	AUS	EIN
36	AUS	User input		ioErrQR	AUS	EIN
37	AUS	User input		ioChuck	AUS	EIN
38	AUS	User input		ioUnChuck	AUS	EIN
39	AUS	User input		UIN6	AUS	EIN
40	AUS	User input		UIN7	AUS	EIN
41	AUS	User input		UIN8	AUS	EIN
42	AUS	User input		UIN9	AUS	EIN
43	AUS	User input		UIN10	AUS	EIN
44	AUS	User input		UIN11	AUS	EIN

③ Doppelklicken Sie im Fenster [DIO Manager] das Feld [Dummy SW] von E/A (Nr. 34), um es auf EIN zu setzen.

④ Doppelklicken Sie auf das Feld [Status] der E/A Nr. 34, um sie auf EIN zu setzen.

⑤ So aktivieren Sie den E/A-Dummy:
Klicken Sie im Fenster [DIO Manager] die Schaltfläche [E/A-Dummy] an.
Die Schaltfläche [E/A-Dummy] erscheint gedrückt.

Das Programm erkennt die E/A Nr. 34 "ioParts" als auf "EIN" gesetzt und fährt mit dem nächsten Schritt fort. Wenn kein Problem erkannt wird, fährt das Programm bis zum Ende fort und stoppt dann.

Lektion 13 Überwachen und Manipulieren der Variablen

Über den Variablen Manager können Sie Variablen überwachen.

Bei dieser Lektion werden Sie die Ganzzahl-Variable [0] von Makro "iParts" prüfen, die bei der Betriebszählung verwendet wird.

Anmerkung: Siehe "Lektion 5 Definieren von Makros" für die Variablennummer 0 von "ioParts".

13.1 Starten des Variablen Managers und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller

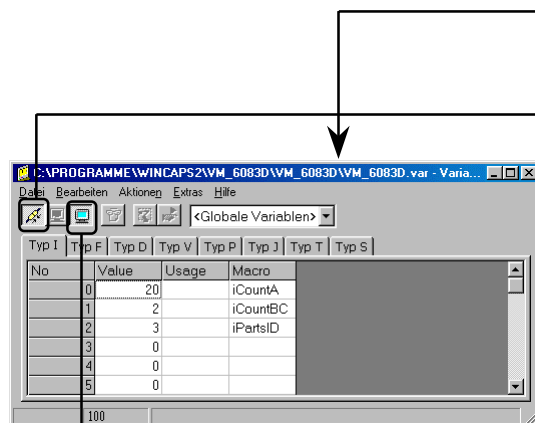
Starten Sie den Variablen Manager und stellen Sie die Kommunikationsverbindung her, damit ein konstanter Datenaustausch mit dem Roboter-Controller gewährleistet ist.

SCHRITT 1 Starten Sie den Variablen Manager



① Klicken Sie im Fenster [DENSO System Manager] die Schaltfläche Variable an.

SCHRITT 2



Das Fenster [Variable Manager] erscheint.

② Klicken Sie die Schaltfläche [Verbinden] an, um eine Kommunikationsverknüpfung zum Roboter-Controller herzustellen.
Die Schaltfläche [Verbinden] wird gedrückt dargestellt.

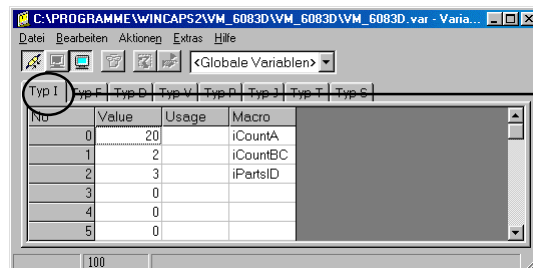
③ Klicken Sie die Schaltfläche [Kontinuierlicher Monitor] an, um den kontinuierlichen Monitor einzuschalten.
Die Schaltfläche [Kontinuierlicher Monitor] wird gedrückt dargestellt.

Durch den oben beschriebenen Prozeß wurde die Kommunikationsverknüpfung hergestellt, um den Datenaustausch zwischen PC und Roboter-Controller zu ermöglichen.

13.2 Überwachen von Variablen

Durch Einrichten der Kommunikationsverknüpfung zum Roboter-Controller können Sie die im Roboter-Controller verwendeten Variablen überwachen.

SCHRITT 1 Anzeigen der Variablen



① Blättern Sie im Fenster [Variable Manager] weiter, bis die Ganzzahl-Variable [0] angezeigt wird. Verwenden Sie dazu die Bildlaufleiste auf der rechten Seite des Fensters.

Die Spalte [Wert] für Ganzzahl-Variablen [0] zeigt die an den Variablen vorgenommenen Änderungen an.

Lektion 14 Kontinuierlicher Testlauf

Bis zur vorherigen Lektion haben Sie PRO1 mit der Einzelzyklus-Ausführung geprüft.

Bei der kontinuierlichen Ausführung läßt der Roboter die programmierte Bewegung wiederholt ausführen. Wird E/A Nr. 34 "ioParts" über den Dummy-Schalter auf EIN gehalten, ist die kontinuierliche Ausführung sogar möglich, wenn kein aktueller Eintrag bei E/A Nr. 34 vorliegt.

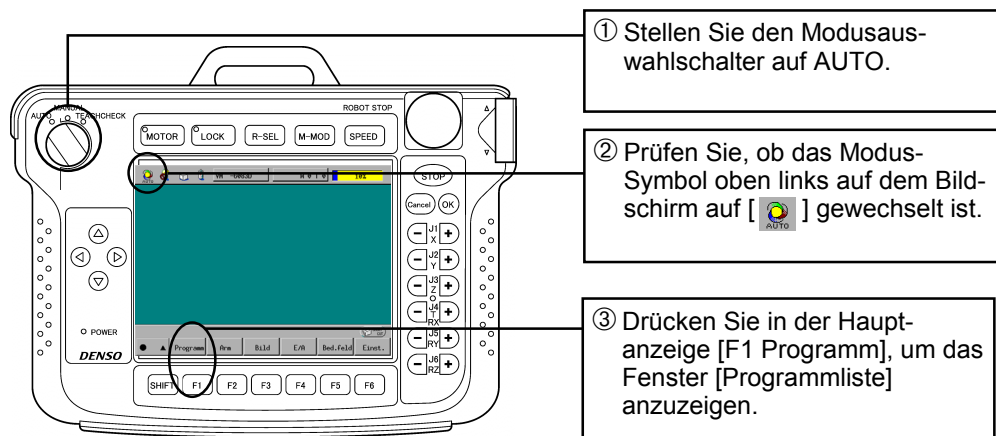
Bevor Sie den Roboter tatsächlich steuern, aktivieren Sie die Maschinenverriegelung des Roboter-Controllers und führen einen kontinuierlichen Testlauf durch. Dabei überwachen Sie im Arm Manager die simulierten Roboterbilder.

Im folgenden wird angenommen, daß die E/A Nr. 34 "ioParts" des vorhergehenden Abschnittes auf EIN gesetzt bleibt.

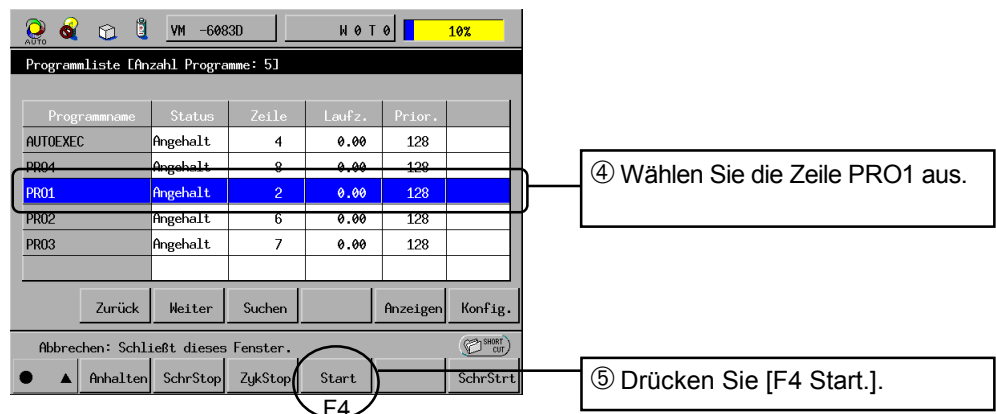
14.1 Kontinuierliche Ausführung

Starten Sie den Variablen Manager, und stellen Sie die Kommunikationsverknüpfung zum Roboter-Controller her, um den Datenaustausch zwischen Variablen Manager und Roboter-Controller zu ermöglichen.

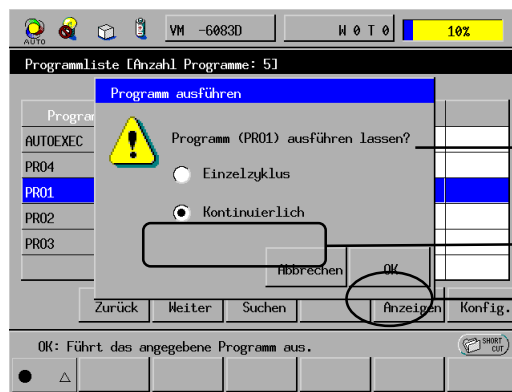
SCHRITT 1 Anzeigen des Fensters "Programmliste" im Auto-Modus



SCHRITT 2 Auswählen des auszuführenden Programmes



SCHRITT 3 Kontinuierliche Ausführung



Der Auswahl-Bildschirm für Einzelzyklus und Kontinuierlich wird angezeigt.

⑥ Wählen Sie [Kontinuierlich], und drücken Sie dann [OK].
PRO1 startet eine kontinuierliche Ausführung.

14.2 Kontinuierliche Überwachung der E/As

In diesem Abschnitt werden Sie den E/A-Status bei kontinuierlicher Ausführung mit dem DIO Manager überwachen.

Der DIO Manager kennt drei Typen von E/A-Anzeigeformaten: Tabellentyp, Oszilloskoptyp und Anzeigentyp.

Ist die Schaltfläche [Monitor] gedrückt, dann wird der E/A-Status des Feldes [Monitor] "EIN" kontinuierlich in Echtzeit angezeigt.

Sie werden die im Programm verwendeten E/As überwachen.

Der in Lektion 13 geöffnete DIO Manager muß nach wie vor geöffnet sein. Ist er geschlossen, klicken Sie im System Manager die Schaltfläche [Variable] an, um ihn zu starten.

Tabellentyp

Das E/A-Anzeigeformat des Tabellentyps ist ähnlich der Liste, die beim Öffnen des DIO Managers zunächst erscheint.

Wird nicht der Tabellentyp angezeigt, dann wählen Sie aus der Liste [Anzeigeauswahl] den Tabellentyp.

Nr.	Status	Typ	Verwendung	Makro	Monitor	Dumm
34	AUS	User input	parts supply signal	ioParts	EIN	EIN
35	AUS	User input	motion complete signal	ioComplete	EIN	EIN
36	AUS	User input		ioErrOR	EIN	EIN
37	AUS	User input		ioChuck	AUS	EIN
38	AUS	User input		ioUnChuck	AUS	EIN
39	AUS	User input		UIN6	AUS	EIN
40	AUS	User input		UIN7	AUS	EIN
41	AUS	User input		UIN8	AUS	EIN
42	AUS	User input		UIN9	AUS	EIN
43	AUS	User input		UIN10	AUS	EIN
44	AUS	User input		UIN11	AUS	EIN

① Blättern Sie im Fenster [DIO Manager] bis zur Anzeige der E/A Nr. 34 - 36.

(Verwenden Sie zum Blättern die Bildlaufleiste auf der rechten Seite des Fensters.)

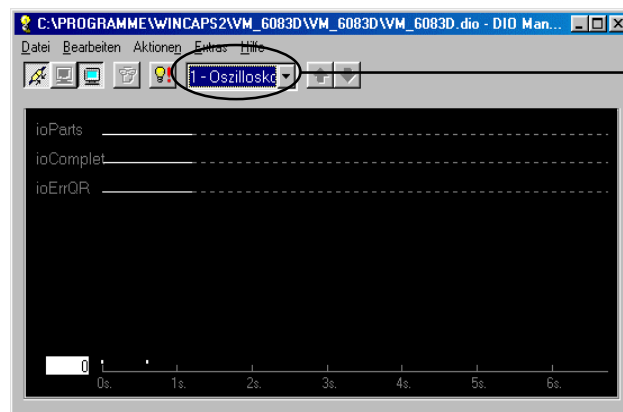
② Doppelklicken Sie auf die im Fenster [DIO Manager] angezeigten Felder [Monitor] von E/A Nr. 34 bis 36, um diese auf "EIN" zu setzen.

Entsprechend der an E/A vorgenommenen Änderungen wechselt die Anzeige des Feldes [Status] auf EIN oder AUS.

Oszilloskoptyp

In der Tabellentyp-Anzeige des Fensters [DIO Manager] können Sie den E/A-Status mit eingeschalteter Monitor-Software wie auf einem Oszilloskop beobachten.

Fahren Sie in Fortsetzung der im vorhergehenden Abschnitt beschriebenen Operation wie folgt fort:



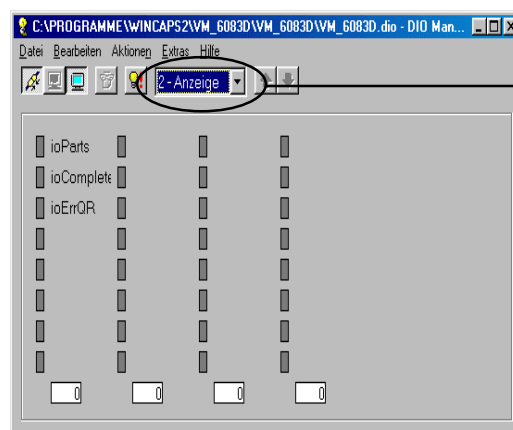
① Klicken Sie die Schaltfläche  an, und wählen Sie [1 – Oszilloskop] aus der angezeigten Liste.
Die Zustände der E/A Nr. 34 bis 36 werden wie bei einem Oszilloskop angezeigt.

② Die Oszilloskoptypanzeige eignet sich zur Untersuchung der Statusveränderungen von E/A-Signalen über der Zeit.
Überprüfen Sie, ob jeder E/A entsprechend dem Programm ein- oder ausgeschaltet ist.

Anzeigentyp

Über den DIO Manager kann der E/A-Status auch wie bei einer Anzeigentypdarstellung angezeigt werden.

Hier können Sie den E/A-Status bei eingeschaltetem Monitor im Tabellentypformat anzeigen.

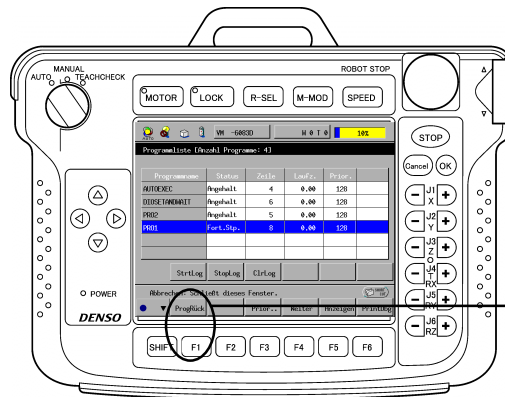


① Klicken Sie die Schaltfläche  der Liste [Anzeigeschalter] an, und wählen Sie [2 – Anzeige] aus der angezeigten Liste.
Die Zustände der E/A Nr. 34 bis 36 werden wie bei einem Oszilloskop angezeigt.

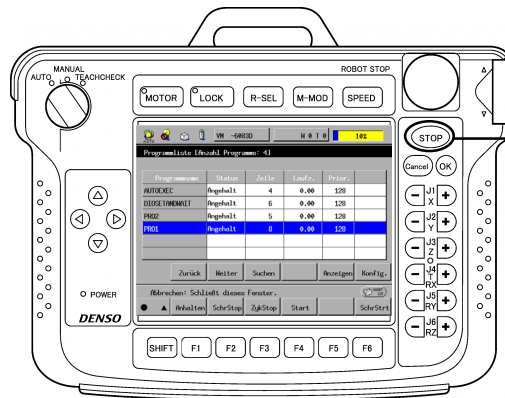
② Die Anzeigentypdarstellung ist geeignet zur gleichzeitigen Beobachtung der Zustände und Änderungen mehrerer E/A-Signale.
Überprüfen Sie, ob jeder E/A entsprechend dem

14.3 Sofortiger Stop (Anhalten)

Bis zu dieser Stelle haben Sie die kontinuierliche Ausführung von "PRO1" geprüft. Bevor Sie mit Kapitel 4 fortfahren, stoppen Sie die kontinuierliche Ausführung des Programmes.



① Drücken Sie [F1 Anhalten]
Das Programm wird sofort unterbrochen und das Feld [Status] zeigt Fort.Stp. an.



② Drücken Sie [STOP].
Das Feld [Status] zeigt Gestoppt an und die programmierte Operation wird unterbrochen.

Kapitel 4

Steuern des Roboters über Programme

In Kapitel 4 lernen Sie folgendes:

- Steuern Sie den Roboter über Programme.
- Lernen Sie die Palettierung, die zu den Hauptanwendungen bei Robotern mit 4 Achsen gehört, und die zugehörige Programmierung kennen.
- Lernen Sie, wie Sie die einfachste Möglichkeit zum Entwickeln von Roboterprogrammen, die sog. PAC-Bibliothek, effektiv einsetzen können.

► Lektion 15 Steuern des Roboters in der Praxis	90
15.1 Lösen der Maschinenverriegelung	90
15.2 Einstellung der Roboter-Geschwindigkeit und -Beschleunigung	91
15.3 Einschalten des Antriebsmotors	92
15.4 Programmierung	93
15.5 Starten von Programmen	96
15.6 Wechseln der Roboter-Geschwindigkeit	97
15.7 Stoppen des Roboters bei kontinuierlicher Ausführung	98
■ Erstellen des Programmes [Fortsetzungsstop] und [Fortsetzungsstart]	98
■ Vollständiges Stoppen des Programmes ([Programmrücksetzung])	100
► Lektion 16 Palettierung	102
Lektion 17 Effektiver Einsatz von Bibliotheken	110
► 17.1 Programmbank	111
17.2 Importieren von Programmen	113
Lektion 18 Beenden der Sitzung	115
► 18.1 Beenden von WINCAPSII und Herunterfahren des PCs	115
18.2 Ausschalten des Roboter-Controllers	115

Lektion 15 Steuern des Roboters in der Praxis

Sie haben in Kapitel 1 bis 3 erfahren, wie der Programmbetrieb zu prüfen ist. In Lektion 15 üben Sie das Steuern von Robotern über Programme'.

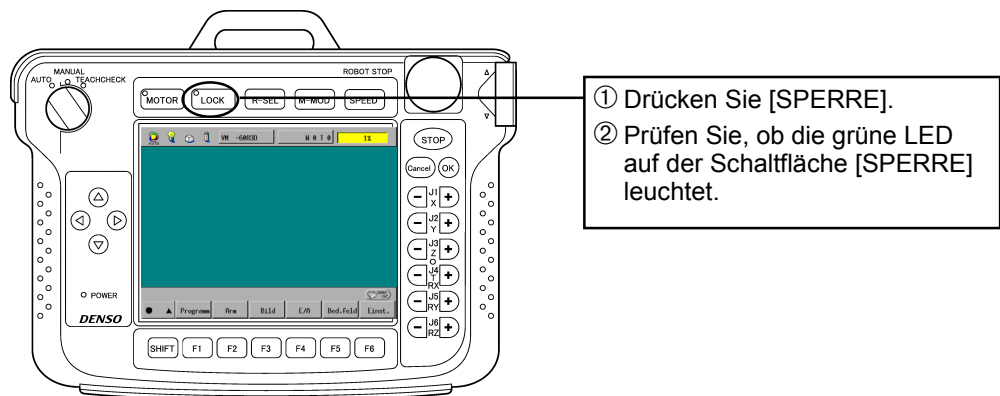
Zuerst müssen Sie eine Sicherheitsprüfung durchführen. Dann stellen Sie den Roboter auf eine niedrige Geschwindigkeit ein, und prüfen Sie die Roboterbewegung im langsamen Betrieb.

Sie müssen auch einen Testlauf des Roboters über E/A-Dummy-Signale ausführen. In dieser Lektion werden Sie die aktuelle Hardware an den Roboter-Controller anschließen und dessen Bewegung prüfen.

Siehe INSTALLATIONS- UND WARTUNGSHANDBUCH für Einzelheiten zum Einstellen der Hardware.

15.1 Lösen der Maschinenverriegelung

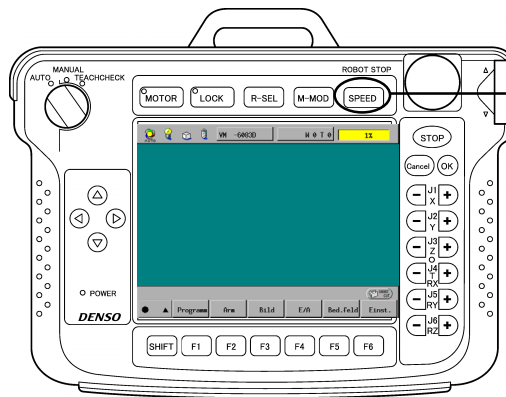
SCHRITT 1



Dadurch wird die Maschinenverriegelung gelöst.

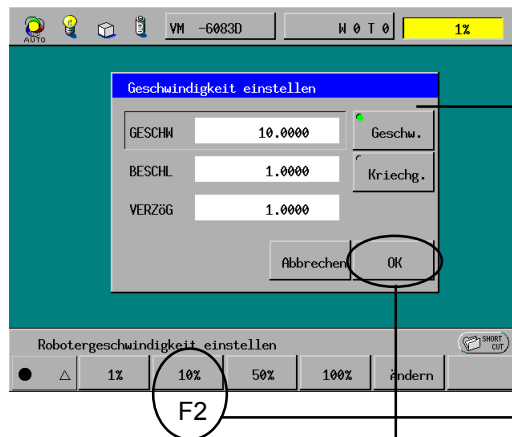
15.2 Einstellung der Roboter-Geschwindigkeit und -Beschleunigung

SCHRITT 1



① Drücken Sie [GESCHWINDIGKEIT].

SCHRITT 2



Das Fenster [Geschwindigkeit einstellen] erscheint.

② Drücken Sie [F2 10%].

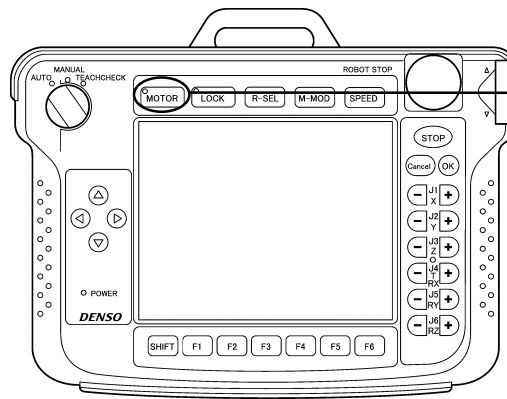
Die eingestellten Werte erscheinen in den Feldern [GESCHW], [BESCHL] und [VERZÖG].

③ Drücken Sie [OK], um diese Einstellungen zu übernehmen.

Damit ist die Prozedur zur Einstellung von Geschwindigkeit und Beschleunigung des Roboters abgeschlossen.

15.3 Einschalten des Antriebsmotors

SCHRITT 1



- ① Drücken Sie [MOTOR].
- ② Prüfen Sie, ob die LED auf der Schaltfläche [MOTOR] leuchtet.

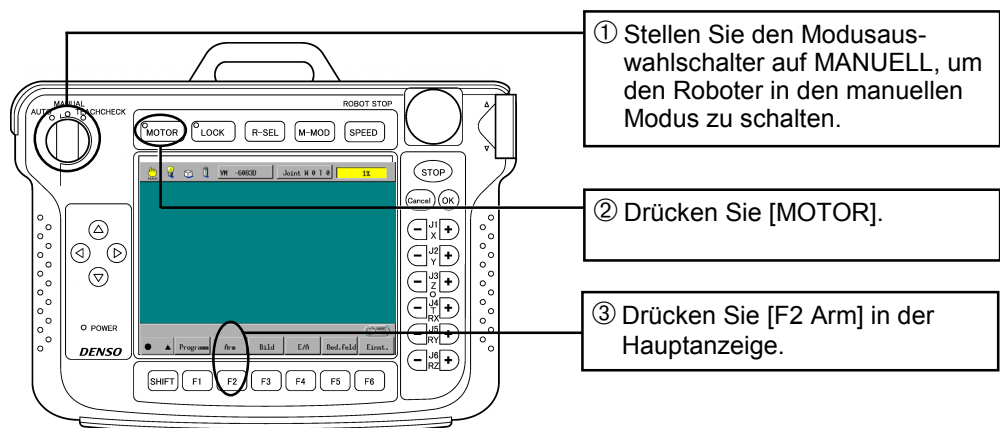
Die Motorspannung ist jetzt eingeschaltet.

15.4 Programmierung

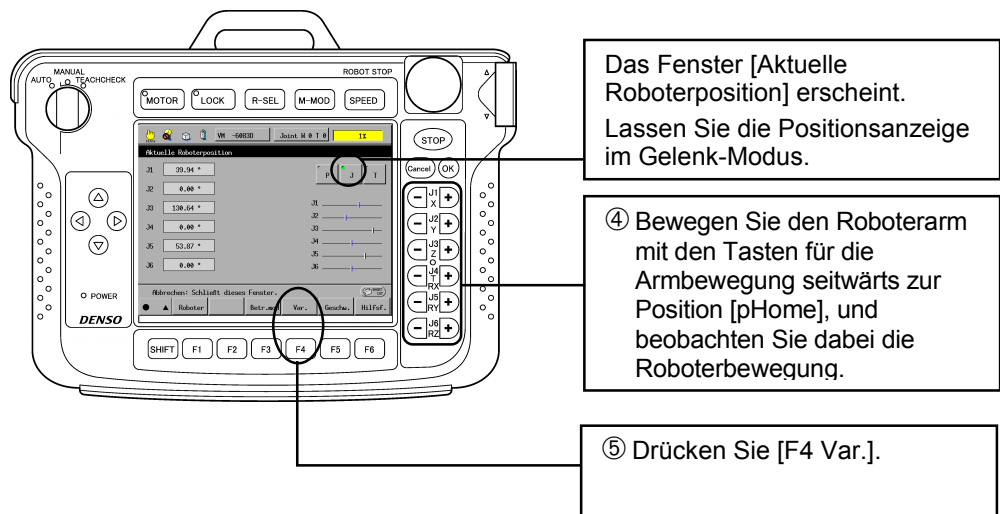
Sie haben in Lektion 10 "Eingeben von Positionsvariablen" Werte für die Positionsvariable eingegeben, aber die Positionierung im Arm Manager ist nicht unbedingt exakt. Sie müssen hier eine Neupositionierung durchführen, indem Sie den Roboter in der Praxis einsetzen und genaue Werte der Positionsvariablen zuweisen.

Sie müssen praktisch die gleichen Anweisungen befolgen wie in Lektion 10 "Eingeben von Werten für Positionsvariablen" beschrieben. Hierbei müssen Sie jedoch die aktuellen Werte eingeben und dabei die aktuelle Roboterposition und Armstellung beobachten, ohne die durch Zielwerte dargestellte Position zu suchen.

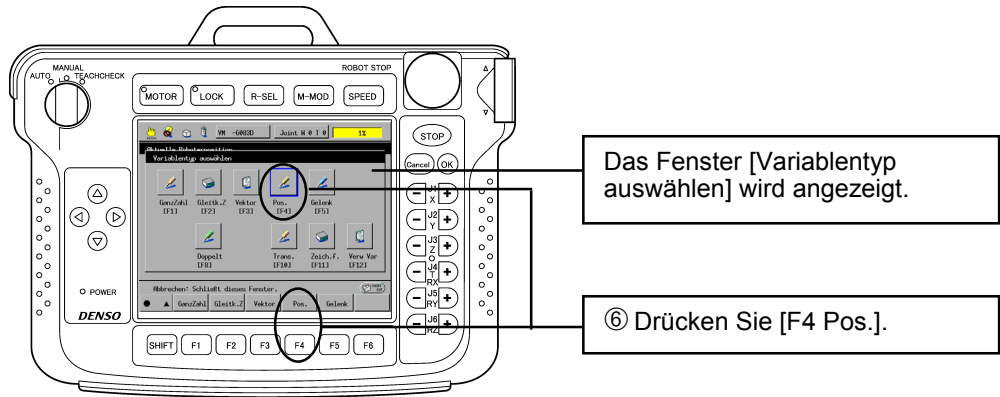
SCHRITT 1 Auswählen des manuellen Modus und Anzeigen der aktuellen Roboterposition



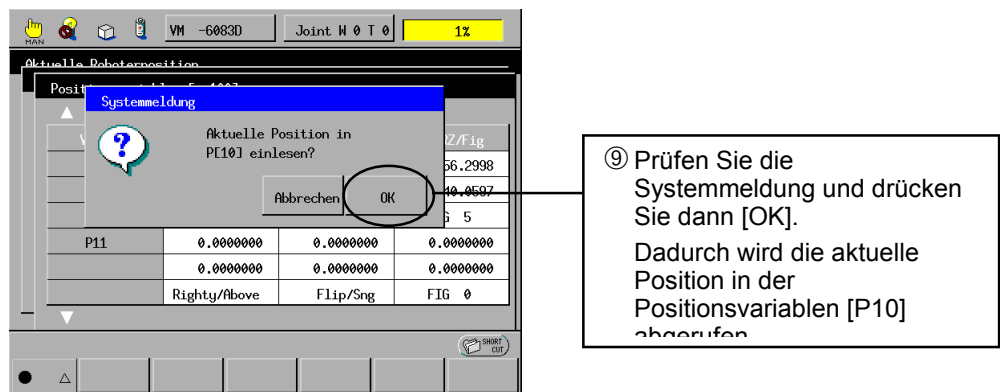
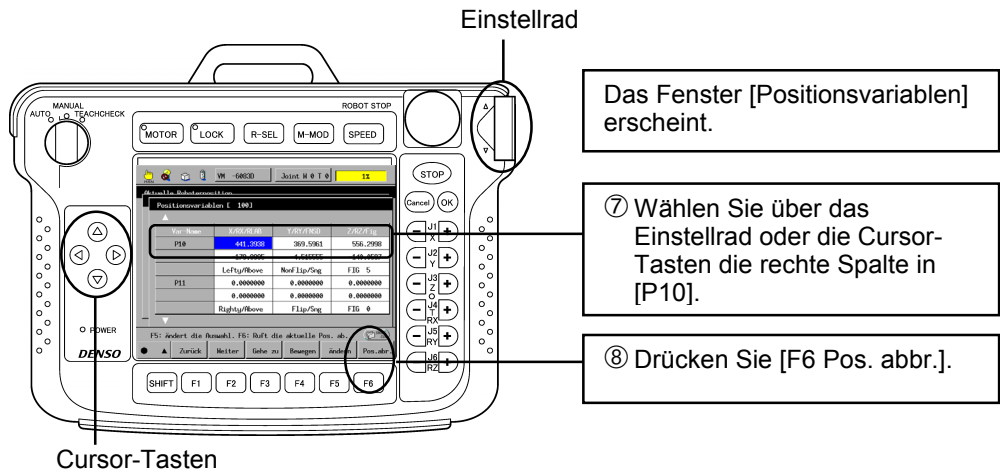
SCHRITT 2 Bewegen des Roboterarmes



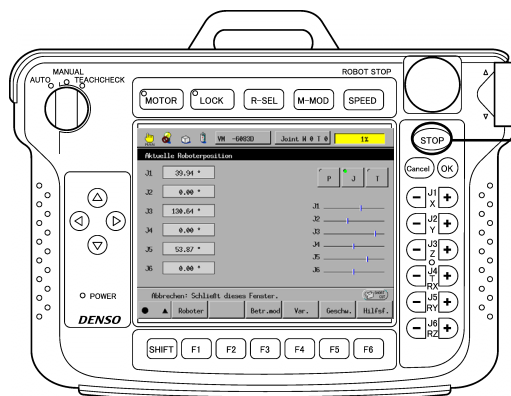
SCHRITT 3 Anzeigen der Positionsvariablen



SCHRITT 4 Abrufen der aktuellen Roboterposition in den Positionsvariablen



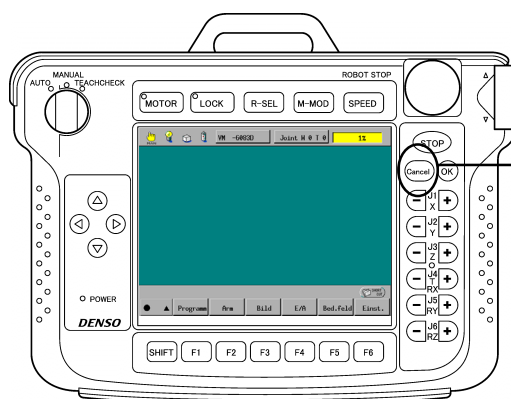
SCHRITT 5 Abrufen sonstiger Positionswerte in den Positionsvariablen



⑩ Drücken Sie zweimal [Abbrechen].
Der Benutzer kehrt zum Fenster [Aktuelle Roboterposition] zurück.

⑪ Wiederholen Sie die Schritte 2 bis 4, um die Positionsdaten zu [P11], [P12] und [P13] in den jeweiligen 3 Variablen [pPlace1], [pPlace2] und [P13] abzurufen.

SCHRITT 6 Zurück zur Hauptanzeige



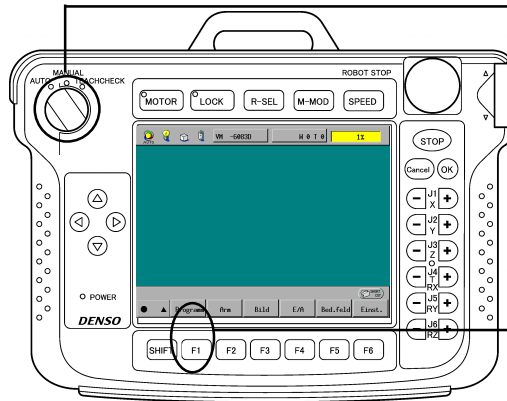
⑫ Nach Abschluß aller oben genannten Prozeduren drücken Sie [Abbrechen], um zur Hauptanzeige zurückzukehren.

15.5 Starten von Programmen

Sie werden den Roboter jetzt tatsächlich über ein Programm steuern.

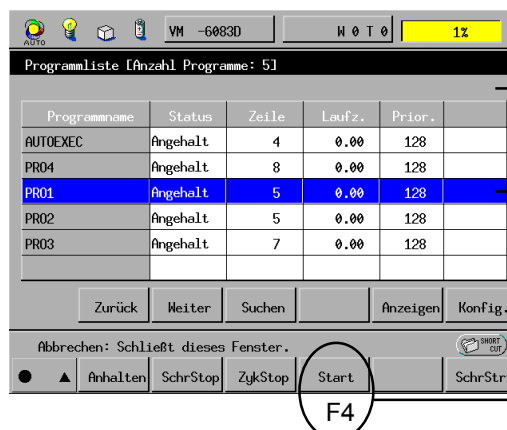
Achtung: Halten Sie stets eine Hand bereit, die STOP-Taste zu drücken.

SCHRITT 1 Wechseln zum Auto-Modus, um das Fenster "Programmliste" anzuzeigen



① Stellen Sie den Modusauswahlschalter auf AUTO, um den Roboter in den Auto-Modus zu schalten.

② Drücken Sie [F1 Programm].



Das Fenster [Programmliste] erscheint.

③ Wählen Sie die Zeile [PRO1] aus.

④ Drücken Sie [F4 Start].

SCHRITT 2



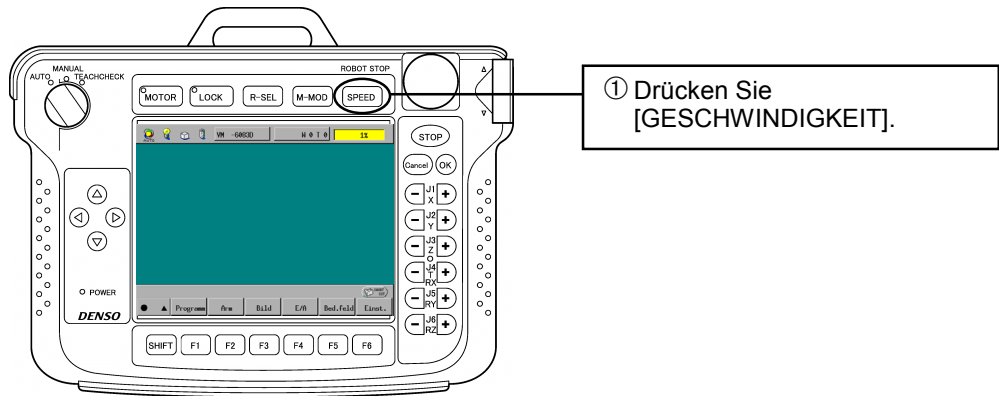
⑤ Wählen Sie [Einzelzyklus], und drücken Sie dann [OK]. Das Task-Programm [PRO1] startet im Einzelzyklus.

SCHRITT 3 Wenn Sie [Kontinuierlich] anstatt [Einzelzyklus] auswählen, startet der Roboter im kontinuierlichen Modus.
Wenn der Betrieb im Modus [Einzelzyklus] funktioniert, probieren Sie den Modus [Laufender Start].

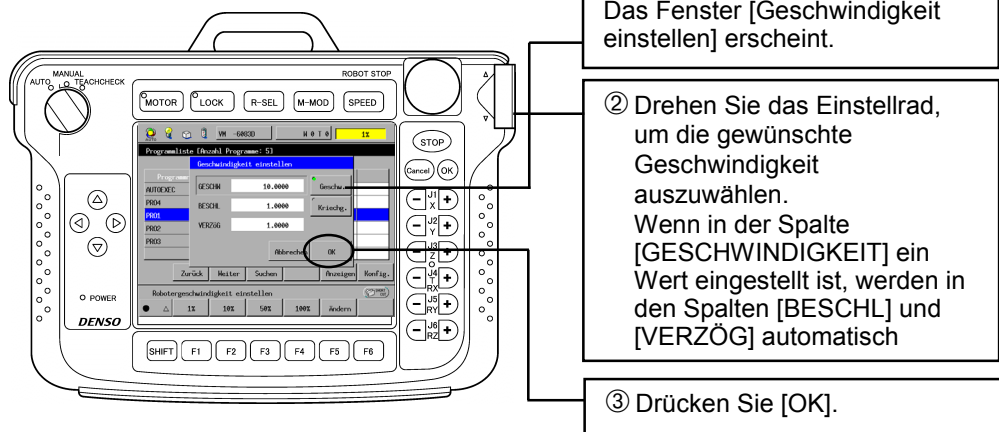
15.6 Wechseln der Robotergeschwindigkeit

Lassen Sie den Roboter schneller laufen, indem Sie langsam die Geschwindigkeitseinstellungen ändern, wenn sowohl Einzelzyklus- als auch kontinuierliche Ausführung funktioniert haben.

SCHRITT 1



SCHRITT 2



Roboter-Geschwindigkeit, -Beschleunigung und Verzögerung sind jetzt eingestellt.

15.7 Stoppen des Roboters bei kontinuierlicher Ausführung

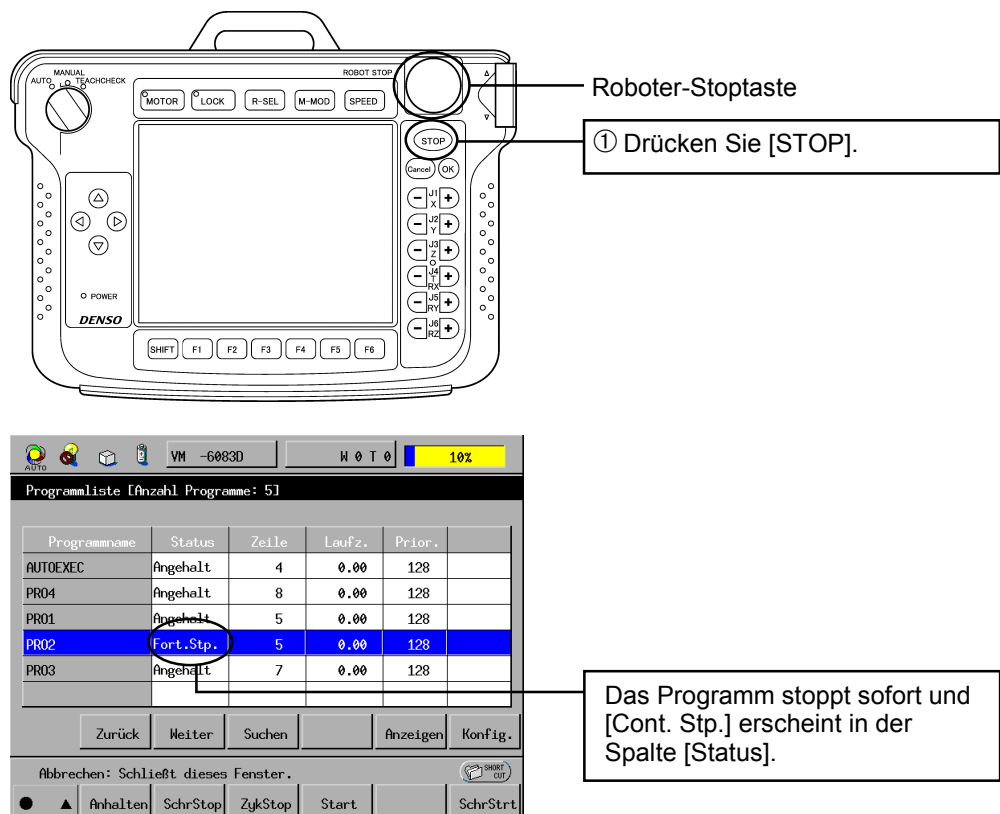
Wenn Sie den Roboter bei kontinuierlicher Ausführung stoppen, indem Sie die Schaltfläche [Stop] drücken, wechselt der Roboter in den Status [Fortsetzungsstop]. In diesem Status können Sie von der Stelle aus, wo es gestoppt wurde, das Programm neu starten.

Wenn Sie das Programm vollständig abbrechen möchten, müssen Sie die Option [Programmrücksetzung] wählen.

Achtung: Im Notfall drücken Sie die Roboter-Stopptaste.

■ Erstellen des Programmes [Fortsetzungsstop] und [Fortsetzungs start]

SCHRITT 1 Erstellen des Task-Programmes [Fortsetzungsstop]

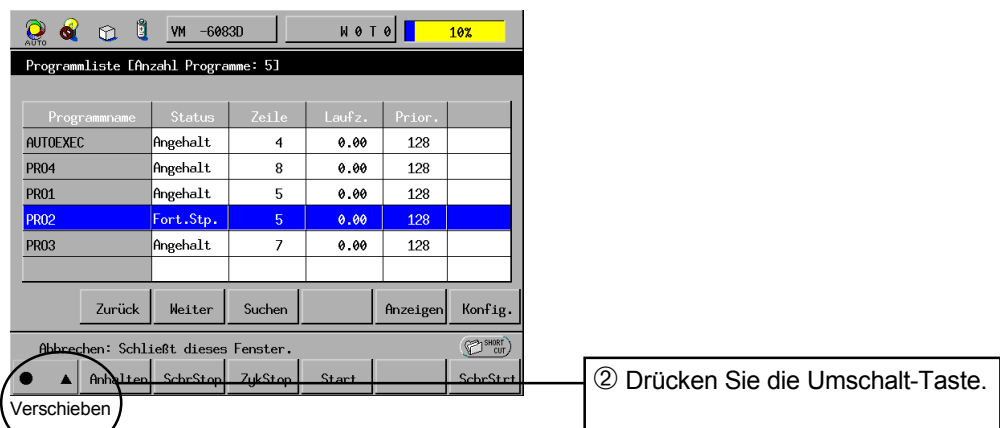


The diagram shows the physical control panel of the robot. A callout points to the 'STOP' button, labeled 'Roboter-Stopptaste'. Another callout points to the 'STOP' button with the instruction '① Drücken Sie [STOP].'. Below the diagram is a screenshot of the 'Programmliste' (Program List) interface. The table shows the following data:

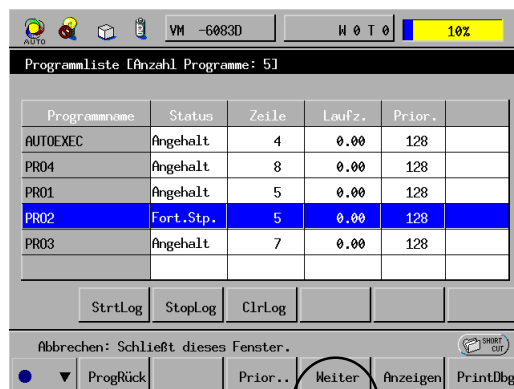
Programmname	Status	Zeile	Laufz.	Prior.
AUTOEXEC	Angehalt	4	0.00	128
PR04	Angehalt	8	0.00	128
PR01	Angehalt	5	0.00	128
PR02	Fort.Stp.	5	0.00	128
PR03	Angehalt	7	0.00	128

A callout points to the 'Fort.Stp.' status in the PR02 row, stating: 'Das Programm stoppt sofort und [Cont. Stp.] erscheint in der Spalte [Status].'. At the bottom of the interface, there are buttons for 'Anhalten', 'SchrStop', 'ZykStop', 'Start', and 'SchrStrt'.

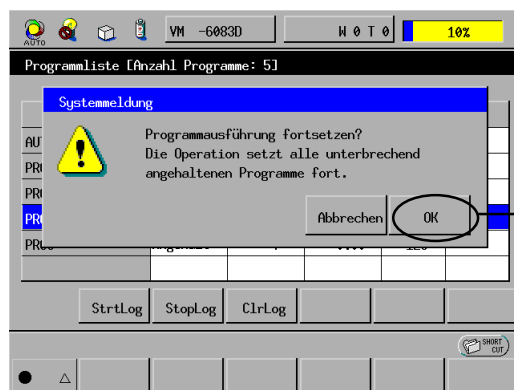
SCHRITT 2 Erstellen des Programmes [Fortsetzungsstart]



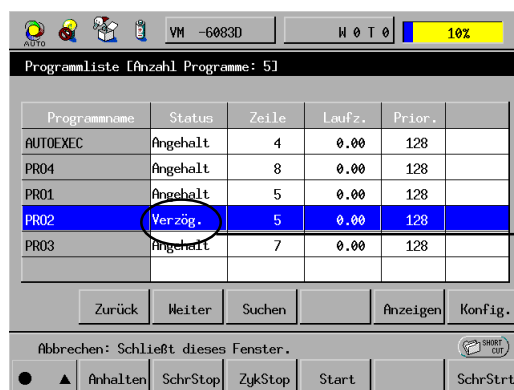
The screenshot shows the same 'Programmliste' interface as in the previous step. The 'PR02' row is still highlighted with 'Fort.Stp.' status. A callout points to the 'Verschieben' button (represented by a circle with a right-pointing arrow) at the bottom left of the interface, labeled '② Drücken Sie die Umschalt-Taste.'.



③ Drücken Sie [F10 Weiter].



④ Drücken Sie [OK].

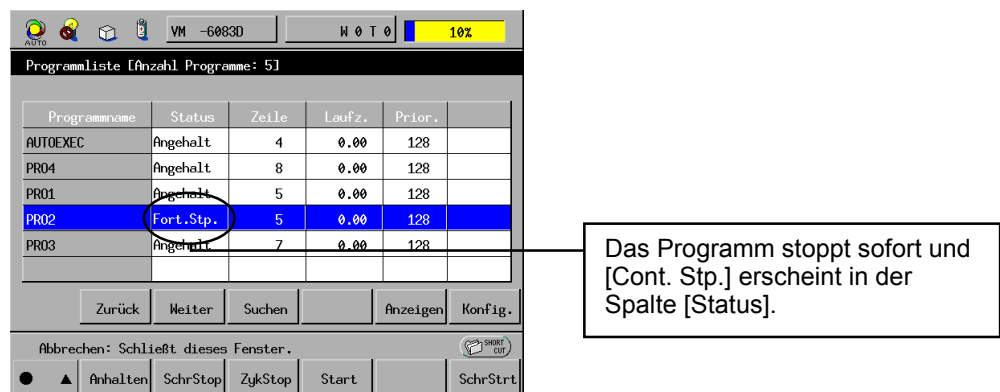
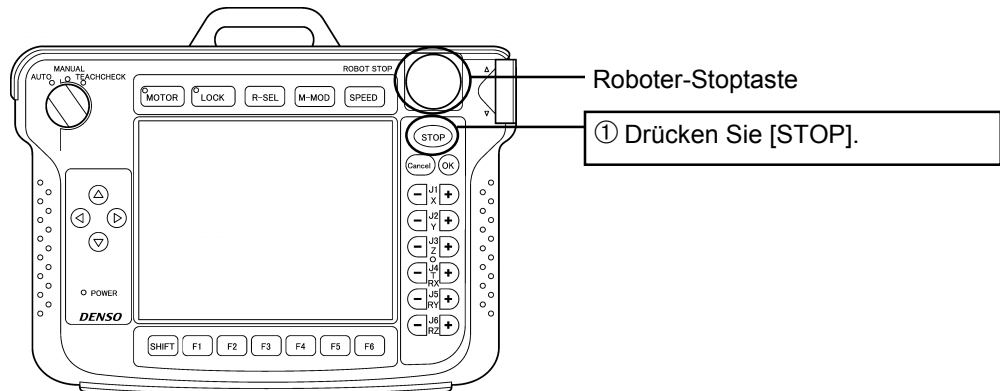


Das Programm wird erneut gestartet und [Aktiv] erscheint in der Spalte [Status].

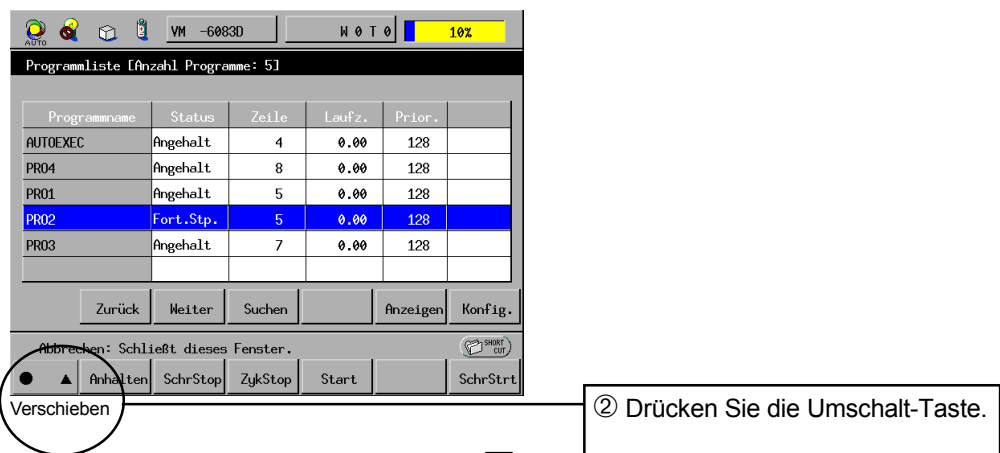
Jetzt können Sie das Programm neu starten.

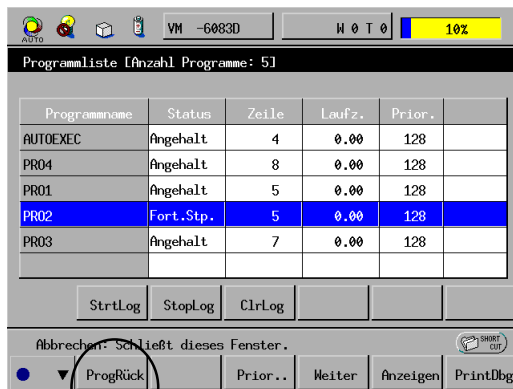
■ Vollständiges Stoppen des Programmes ([Programmrücksetzung])

SCHRITT 1 Erstellen des Programmes [Fortsetzungsstop]



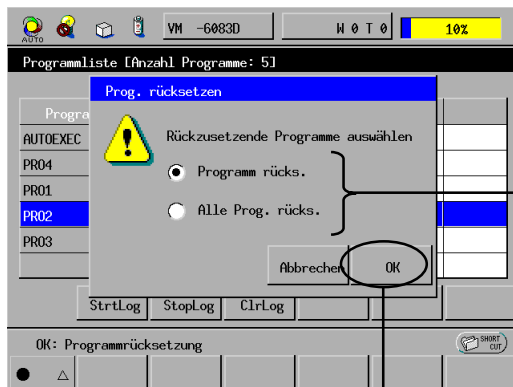
SCHRITT 2 Rücksetzen des Programmes (Programmrücksetzung)





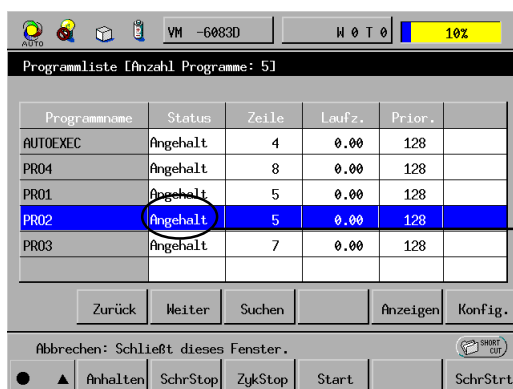
F7

③ Drücken Sie [F7 ProgRst.].



④ Wählen Sie das Programm, das Sie abbrechen möchten.
Wenn Sie die Option [Dieses Programm rücksetzen] wählen, wird nur das aktuelle Programm abgebrochen.
Wenn Sie die Option [Alle Programme rücksetzen] wählen, werden alle aktuellen Programme abgebrochen.

⑤ Drücken Sie [OK].



Das (die) Programm(e) wird (werden) abgebrochen und in der Spalte [Status] wird [Gestoppt] angezeigt.

Das (die) Programm(e) wird (werden) jetzt abgebrochen.

Lektion 16 Palettierung

Was versteht man unter Palettierung?

Palettieren bedeutet, daß in einer programmierten Reihenfolge Teile abgelegt oder von einer eingeteilten Palette (siehe unten) entnommen werden.

Sie können Bibliotheksprogramme einfach zum Palettieren einsetzen. Zum Einsetzen dieser Programme müssen Sie nur die Zahl der auf der Palette vorgesehenen Unterteilungen und deren jeweiligen Positionen an den 4 Ecken der Palette kennen und diese Informationen für den Roboter programmieren.

Die Palettierungsprogramme aktualisieren die Informationen zur Unterteilung, während jede Position zum Aktivieren des Roboters aufgerufen wird, um zu erfahren, an welcher Position das nächste Teil abgelegt oder von wo es entnommen werden muß.

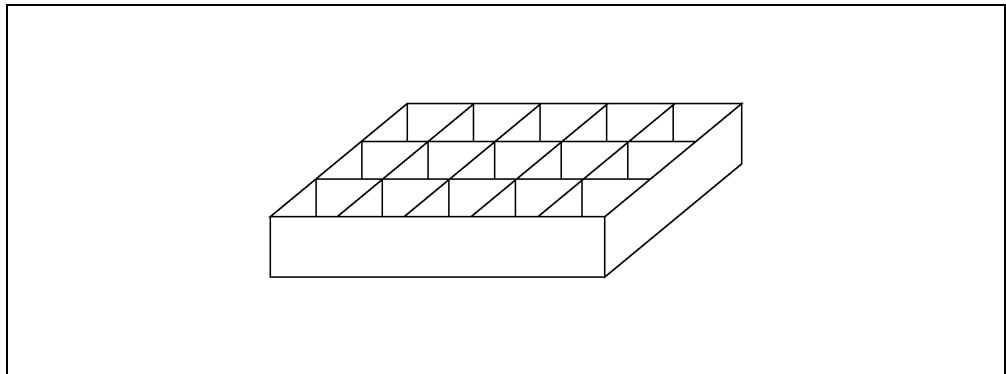


Abbildung 16-1 Partitionierte Palette

Palettierungsprogramm “PRO3”

Unter Einbeziehung der Bibliothek steht die übliche Prozedur zum Entwickeln eines Programmes im Programm “PRO3” mit dem Titel “Palettierungsvorlage,” zur Verfügung, obwohl es je nach Anwendungen und Einsatzzweck wirklich viele unterschiedliche Programme zur Palettierung gibt.

Setzen Sie diese “Palettierungsvorlage” effektiv als Vorlage ein, indem Sie die für diese Anwendungen erforderlichen Elemente hinzufügen/löschen.

Im folgenden ist ein Mustervorlagen-Programm mit der Bezeichnung “PRO3.” aufgelistet.

Folgendes ist bei dieser Mustervorlage festgelegt:

- die Palettierungspunkte müssen bei P50 bis P55 liegen und
- das zu entwickelnde Programm müßte den Roboter so steuern, daß er zur Arbeits-Entnahmeposition P50 bewegt wird, das Palettierungsprogramm “0” steuert, daß er zur Position P51 (der Position zur Befestigung des Werkstückes) bewegt wird, dieses Werkstück entspannt, das Ende der Palette prüft, die Palette ersetzt, wenn ein Signal für das Arbeitsende erkannt wurde, und ggf. das Programm beendet, wenn kein Werkstück mehr übrig ist.

Wenn Sie ein “Neues Projekt” erstellen, indem Sie “Palettierung” im “Gerätetyp,” auswählen, wird der System Manager die “Palettierungsvorlage” mit dem Programmnamen “PRO3” und die “Vorlage zur Palettierungsinitialisierung” mit dem Programmnamen “PRO4 automatisch registrieren.”

' !TITLE “Palettierungsvorlage”	
' !AUTHOR “DENSO Robot Engineering”	
#DEFINE pltIndex 0	' Index des Palettierungsprogrammes
	' Es kann eine beliebige Nummer zwischen 0
	' und 30 ausgewählt werden.
#DEFINE ChuckNG 40	' Nummer der E/A-Entnahmeprüfung
	' Es kann eine positive Ganzzahl ausgewählt
	' werden.
PROGRAM PRO1	' Benennen Sie PRO1 nach Wunsch um.
MOVE P, P50	' Bewegung zur Position P50.
	' Bewegung zur Palettierungsposition P50.
IF IO[ChuckNG] = ON THEN	' Prüfen Sie den Status der vorherigen Entnahme.
CALL pltDecCnt(pltIndex)	' Ziehen Sie 1 für die Gesamtsumme ab.
END IF	
CALL pltMove(pltIndex)	' Führen Sie Palettierung 0 aus.
MOVE P, P51	' Bewegung zur Befestigungsposition P51.
	' Bewegung zur Palettierungsposition P51.
'<--- Fügen Sie hier das Entspannen oder sonstige Operationen hinzu --->	
CALL pltGetPLT1END(pltIndex,0)	' Erfasst das Ende des 1. Palettensignals an I[0].
	' Zweiter Index “0” ist die Spalten/Zeilennummer
	' von “I” geprüft in der nächsten Zeile.
IF I[0] THEN	' Wenn EIN,
'<--- Fügen Sie hier das Ersetzen der Palette oder sonstige Operationen hinzu --->	
CALL pltResetPLT1END(pltIndex)	' Löscht das Ende des 1. Palettensignals.
END IF	
END	

Abbildung 16-2 Programm [PRO3], “Palettierungsvorlage”

Palettierungsparameter

Die Abbildungen 16-3, 16-4, 16-5 und Tabelle 16-1 enthalten die zur Palettierung erforderlichen Parameter.

Die PAC-Sprache speichert diese Parameter als festgelegte Werte von Variablen.

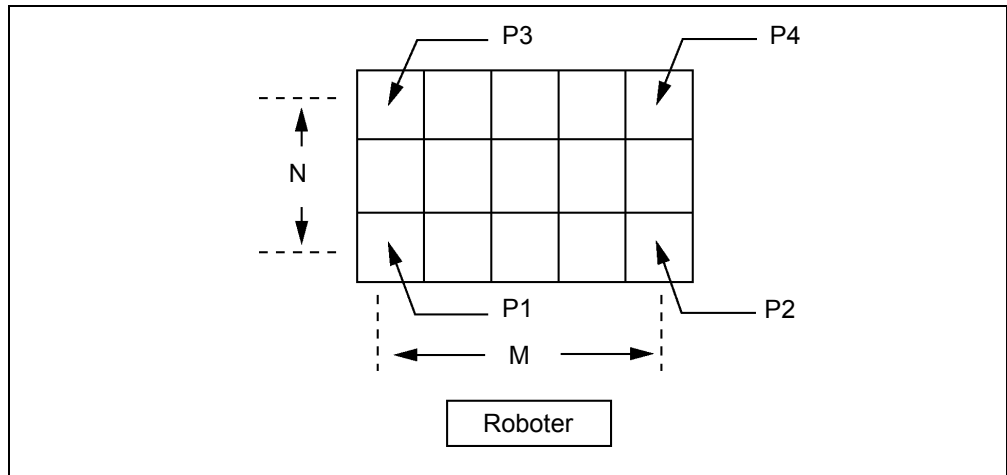


Abbildung 16-3 Paletten-Draufsicht

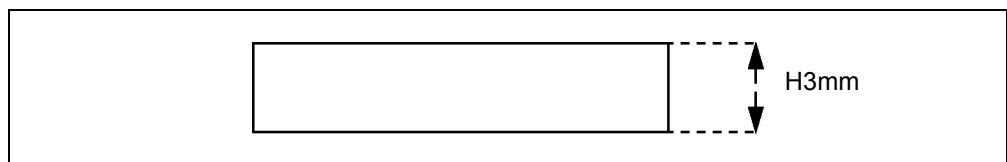


Abbildung 16-4 Seitenansicht der Palette

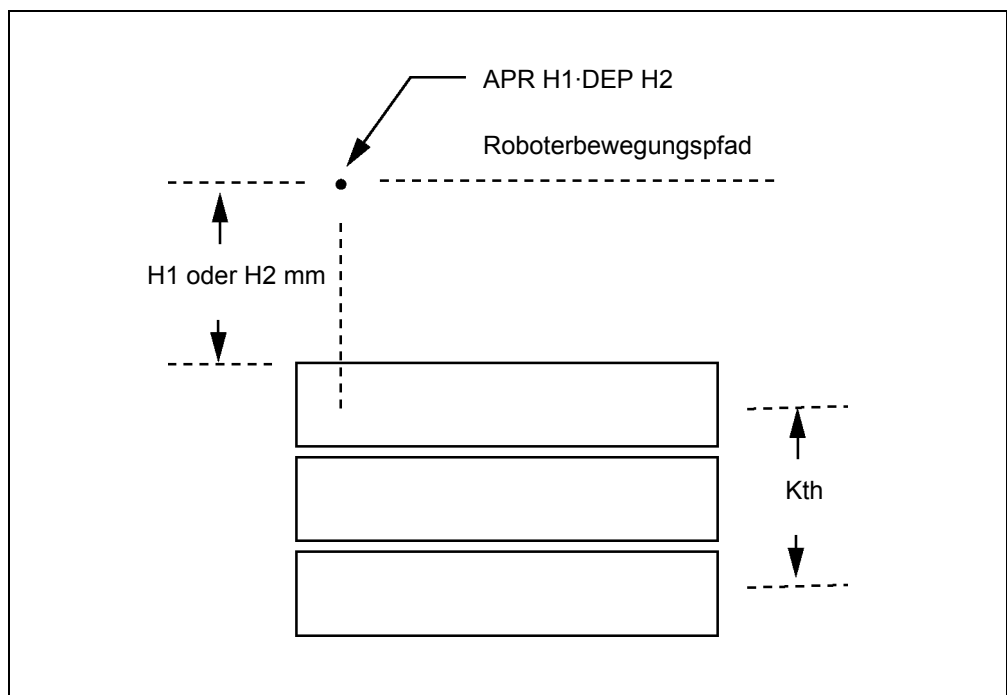


Abbildung 16-5 Gestapelte Paletten

Tabelle 16-1 Für die Palettierung erforderliche Parameter

Symbol	Name	Beschreibung	Einheit
	Palettierungsnummer	Palettierungsindex	Keine (ganze Zahl)
N	Anzahl der Reihen- Teile	Anzahl der Unterteilungen von P1 zu P3	Zählung (Ganzzahl)
M	Anzahl der Spalten- Teile	Anzahl der Unterteilungen von P1 zu P2	Zählung (Ganzzahl)
K	Anzahl der gestapelten Paletten	Anzahl der gestapelten Paletten	Zählung (Ganzzahl)
H1	Annäherungsabstand	Annäherungsabstand, wo sich der Roboter einer Palette annähert	mm (einfache Genauigkeit FPT)
H2	Abweichungsabstand	Abweichungsabstand, wo der Roboter von einer Palette abweicht	mm (einfache Genauigkeit FPT)
H3	Höhe einer Palette	Höhe einer Palette	mm (einfache Genauigkeit FPT)
	Wo H1 und H2 die unten genannten Bedingungen erfüllen. $H1 > \{H3 \times K - 1\} + 5$ $H2 > \{H3 \times K - 1\} + 5$		
P1 P2 P3 P4	Positionen der 4 Ecken der Palette wie in Abbildung 16-3 dargestellt. Es ist nicht möglich, die relative Positionierung einer der Ecken auszutauschen. Für alle Punkte im Programm behält der Roboter seine Ausrichtung dort bei, von wo die Position P1 vorher programmiert wurde.		

N Anzahl der Unterteilungen pro Reihe

Gibt die Zahl der Unterteilungen in jeder Reihe der Palette an.

Wenn hier 3 steht, werden 3 Reihen dargestellt wie im Beispiel auf Abbildung 16-3.

M Anzahl der Unterteilungen pro Spalte

Damit wird die Zahl der Unterteilungen in jeder Spalte der Palette angegeben.

Wenn hier 5 steht, werden 5 Reihen dargestellt wie im Beispiel auf Abbildung 16-4.

K Anzahl der gestapelten Paletten

Damit wird die Anzahl der Paletten in einem Palettenstapel angegeben.

Wenn hier 3 steht, werden 3 gestapelte Paletten dargestellt wie im Beispiel auf Abbildung 16-5.

H1 Annäherungsabstand

Gibt die Länge des Annäherungspfads an, wenn sich der Roboter einer Palette annähert.

Ein Programm wendet die einfache Annäherungspfadlänge bei jedem Aufruf des gleichen Palettierungsprogrammes auf.

H2 Abweichungspfadabstand

Gibt die Länge des Abweichungspfads an, wenn der Roboter von einer Palette abweicht.

Ein Programm wendet die einfache Abweichungspfadlänge bei jedem Aufruf des gleichen Palettierungsprogrammes auf.

H3 Höhe der Paletteneinheit

Gibt die Höhe jeder Palette an.

Für jede einem Stapel hinzugefügte Palette wird ein Plus-Einheitswert hinzugefügt.

Für jede von einem Stapel entfernte Palette wird ein Minus-Einheitswert hinzugefügt.

Wenn der Stapel unverändert ist, wird 0 hinzugefügt.

Achtung: H1 und H2 müssen die unten genannten Bedingungen erfüllen.

$$H1 > \{H3 \times (K-1)\} + 5$$

$$H2 > \{H3 \times (K-1)\} + 5$$

Wenn nicht, tritt ein Fehler beim Initialisieren auf. Durch diese Einschränkungen ist sichergestellt, daß der Roboter die Palette im Betrieb nicht zerdrückt, weil die Annäherungen oder Abweichungen des Roboters von den gestapelten Paletten 5 mm über der obersten Palette des Stapels liegen.

Wie auf Abbildung 16-6 dargestellt, beeinflusst eine Änderung der Stapelhöhe nicht die Abweichungs-/Annäherungspunkte des Roboters im gleichen Palettierungsprogramm.

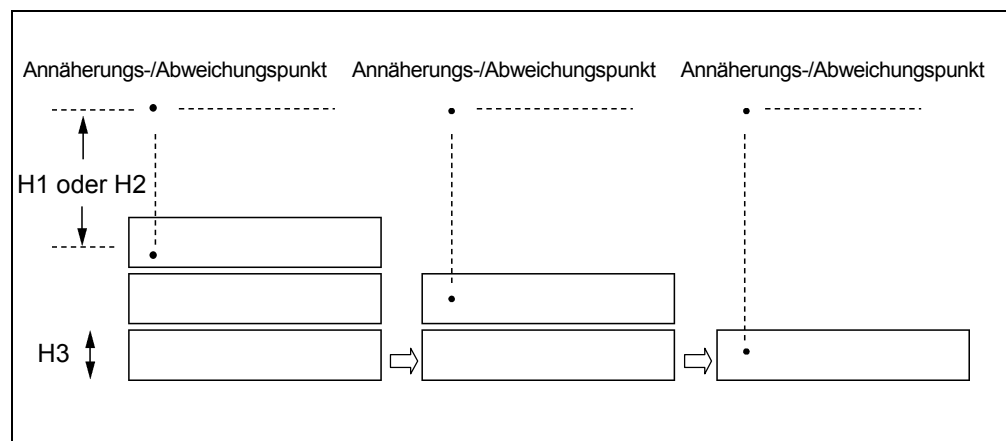


Abbildung 16-6 Beziehung zwischen Stapelhöhe und Annäherungs-/Abweichungspunkten

4 Eckpunkte P1, P2, P3 und P4

Diese Punkte stehen für die Teile-Position jeder der Palettenunterteilungen an allen 4 Ecken. Auf Abbildung 16-7 ist dargestellt, in welcher Reihenfolge der Roboter diese Teile palettiert.

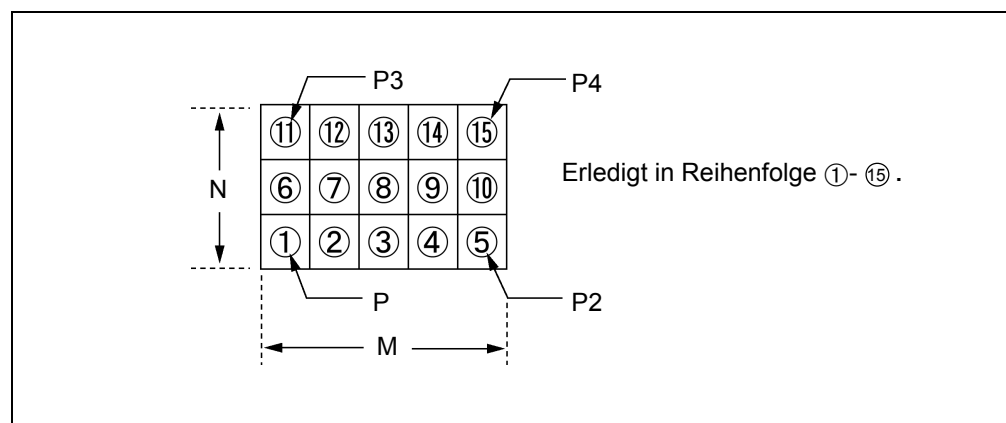


Abbildung 16-7 Palettierungsreihenfolge

Einstellung der Palettierungsparameter

Um die Parameterwerte wie Anzahl der Reihen- und Spalten-Unterteilungen und Anzahl der Paletten in einem Stapel festzulegen, rufen Sie zunächst das Modul "pltInitialize" von der Programmbibliothek aus auf.

Wenn Sie im Dialogfeld [Neues Projekt] unter [Gerätetyp] [1-Palettierung] auswählen, wenn Sie ein "Neues Systemprojekt," erstellen möchten, wird der System Manager das Programm "PRO2" mit der Bezeichnung "Vorlage zur Paletteninitialisierung 1" in der Bibliothek automatisch registrieren. Da dieses Programm zum Aufrufen des Moduls "pltInitialize" dient, ändern Sie die Indexwerte in der CALL-Anweisung in jene um, die Sie verwenden möchten.

```
' !TITLE "Vorlage 1 zur Paletteninitialisierung"
#DEFINE pltIndex 0

PROGRAMM PRO4
  CALL pltInitialize(pltIndex,4,3,1,50,50,50,52,53,54,55)  'Initialisieren der Palettierungsnr. 0.
END
```

Abbildung 16-8 Bibliotheksprogramm "Vorlage 1 der Paletteninitialisierung"

Beschreibungen zu den CALL-Parametern pltInitialize (pltIndex, 3, 5, 3, 50, 50, 10, 52, 53, 54, 55)

- | | |
|---|---|
| 1. ... Index des Palettierungsprogrammes (pltIndex) | |
| 2. ... Anzahl der Reihen-Unterteilungen (3) | |
| 3. ... Anzahl der Spalten-Unterteilungen (5) | |
| 4. ... Anzahl der gestapelten Paletten (3) | |
| 5. ... Annäherungsabstand (50 mm) | |
| 6. ... Abweichungsabstand (50 mm) | |
| 7. ... Palettenhöhe (10 mm) | |
| 8. ... P1 Position (P52) | } Geben Sie nur die Indexe der Positionsvariablen an. |
| 9. ... P2 Position (P53) | |
| 10. ... P3 Position (P54) | |
| 11. ... P4 Position (P55) | |

Abbildung 16-9 Parameter von "pltInitialize"

Sie finden auch Beschreibungen zu den oben genannten Parametern im Werkzeug "Befehlseditor", das im PAC-Manager von WINCAPSII verfügbar ist.

Palettierungszähler

Beim Palettieren zählt der Roboter die Anzahl der wechselnden Unterteilungen und speichert die Zählungen in den Variablen.

Es gibt vier Typen von Zahlenwerten: Anzahl der Unterteilungen pro Reihe (N), Anzahl der Unterteilungen pro Spalte (M), Anzahl der gestapelten Paletten (K) und Gesamt (Zählung).

Diese Zahlenwerte sind festgelegt in "pltKernl", dem Hauptprogramm zur Steuerung des Palettierungsbetriebs.

Das Bibliotheksprogramm "pltMove" addiert jedesmal nach Beenden des Palettierungsbetriebs den Wert 1 des Gesamtzählers hinzu und paßt die Werte der anderen Zähler an.

Das Bibliotheksprogramm "pltDecCnt" subtrahiert bei jedem Programmaufruf den Wert 1 von der Gesamtzahl und paßt jeden Zahlenwert an.

Bis zu 30 Palettierungsprogramme können bei der Anfangseinstellung erstellt werden. Deshalb kann das System auch über 31 Sätze von Palettierungszählern verfügen.

Zählregeln

Der Palettierungszähler addiert zum Gesamtzähler 1 hinzu, jedesmal wenn "pltMove" ausgeführt wird, und paßt die Zahlenwerte entsprechend den anderen an, damit die nächste Palettierungsposition sichergestellt ist.

Wenn 1 zum Gesamtzähler addiert wird, bewegt sich die durch den Spalten-Zähler (M) angezeigte Position der Paletten-Spalte zur nächsten Spalte. Wenn die durch den Spalten-Zähler (M) angezeigte Position der Palettenspalte das Ende bzw. den maximalen Zahlenwert erreicht hat, dann zählt der Reihen-Zähler (N) um 1 weiter und der Spalten-Zähler (M) wird zu seinem Minimalwert. Wenn die Position des Reihen-Zählers (N) das Ende der Reihen-Unterteilung der Palette erreicht hat und zum Maximalwert wird, zählt der gestapelte Reihen-Zähler (K) um 1 weiter und der Reihen-Zähler (N) wird zum festgelegten Minimalwert.

Wird ein Palettierungsprogramm mitten in der Bearbeitung unterbrochen und dann wieder neu gestartet, bewegt sich der Roboter zur nächsten Unterteilung, weil der Wert der Zählervariable hinzuaddiert wird.

Das System speichert den Wert des Palettierungszählers, selbst wenn die Stromzufuhr abgeschaltet wird. Wenn das System nach dem erneuten Starten nicht initialisiert ist, fährt der Roboter mit der Palettierung anhand des letzten Zählerwertes fort.

Achtung: Wenn Sie ein neues Task-Programm kompilieren und sein Laufzeit-Modul laden, initialisiert das System automatisch die Werte aller Variablen.

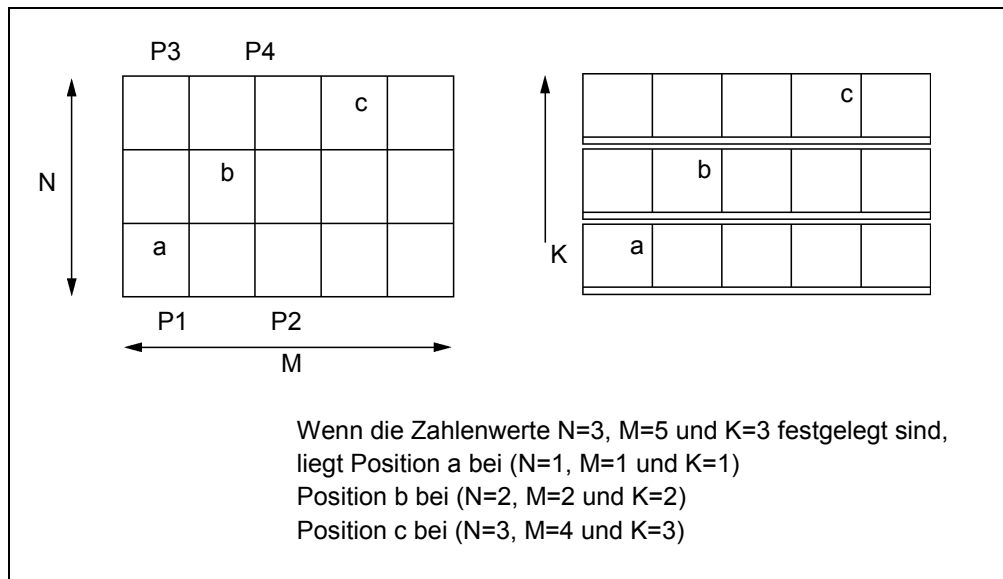


Abbildung 16-10 Verhältnis zwischen Palettierungsposition und Zählerwerten

Initialisieren der Zählerwerte

Wenn Sie Paletten ersetzen oder keine Unterteilungen verwenden möchten, müssen Sie alle Zählerwerte initialisieren.

Die Systeme ersetzen 1 bei allen Zählerwerten, um diese zu initialisieren.

Bei Verwendung des Bibliotheksprogrammes "pltResetAll" können Sie alle Palettierungszählerwerte auf einmal initialisieren.

Wenn Sie beispielsweise alle Zähler der Palette Nummer 1 initialisieren möchten, sieht die Beschreibung wie folgt aus:

```
CALL pltResetAll(1)
```

Wenn Sie die Palettierungszählerwerte unabhängig voneinander initialisieren möchten, verwenden Sie die Bibliotheksprogramme "pltLetN1," "pltLetM1," "pltLetK1" und "pltLetCnt."

Wenn Sie beispielsweise den N-Zähler der Palette Nummer 1 initialisieren möchten, sieht die Beschreibung wie folgt aus:

```
CALL pltLetN1(1,1)
```

Achtung: Beim zweiten Argument handelt es sich um den im N-Zähler zu ersetzenden Wert. Sie können auch jede Nummer außer 1 wählen.

Ende des Palettierungsprogrammes

Beim Beenden der Palettierung eines der gestapelten Paletten oder des gesamten Palettenstapels legt das Palettierungsprogramm ein End-Flag für eine gestapelte Palette oder für einen gesamten Palettenstapel fest.

Um den Status "gestapelte Palette fertig" zu erlangen, verwenden Sie das Bibliotheksprogramm "pltGetPLT1END". Um den Status "gesamter Palettenstapel fertig" zu erlangen, verwenden Sie das Bibliotheksprogramm "pltGetPLT1END".

Um den Flag-Status "gestapelte Palette fertig" auf (0) zurückzusetzen, verwenden Sie das Bibliotheksprogramm "pltResetPLT1END." Um den Status-Flag "gesamter Palettenstapel fertig" auf (0) zurückzusetzen, verwenden Sie das Bibliotheksprogramm "pltResetPLT1END".

Lektion 17 Effektiver Einsatz von Bibliotheken

Wenn Sie bestehende Programme in ein Projekt einfügen wollen, dann stehen Ihnen hierfür drei Möglichkeiten zur Verfügung:

- Programmbank:

In der Programmbank registrierte Programme können zum neuen Projekt hinzugefügt werden.

- Importieren von Programmen:

Sie können Programme in Ihrem neuen Projektordner registrieren, indem Sie bereits für ein anderes Projekt erstellte Programme kopieren.

- Hinzufügen von Programmen:

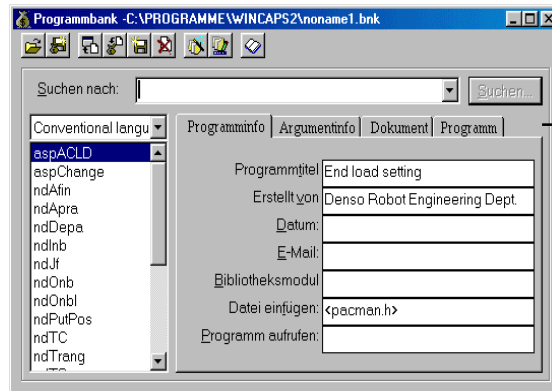
Sie können bereits für andere Programmprojekte in Ihrem neuen Projekt erstellte Programme registrieren. In diesem Fall kann das registrierte Projekt mit anderen Programmen genutzt werden.

Sie können mit einer der oben genannten Methoden bereits vorhandene Programme effektiv für Ihr neues Projekt verwenden. Es folgen noch Beispiele, um dies zu veranschaulichen.

17.1 Programmbank

Fügen Sie das Programm "dioSetAndWait" aus der Programmbank Ihrem neuen Projekt hinzu.

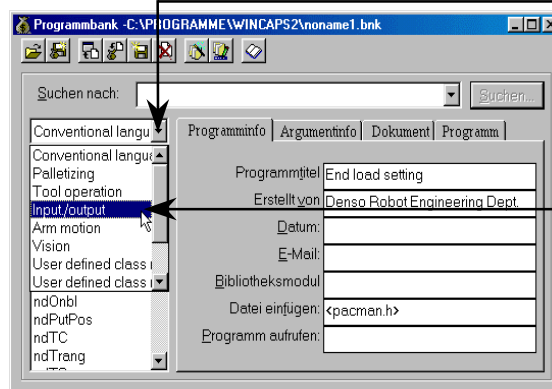
SCHRITT 1 Durchsuchen der Programmbank



① Wählen Sie aus dem Menü "Extras" im Fenster [PAC-Programm] die Option [Programmbank].

Das Fenster [Programmbank] erscheint.

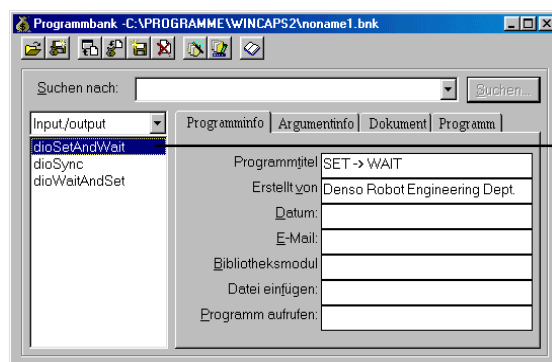
SCHRITT 2 Auswählen der Klasse



② Klicken Sie das Klassen-auswahlfeld an.
Das Pulldown-Menü "Klassen-auswahl" erscheint.

③ Wählen Sie [Eingabe/Ausgabe].

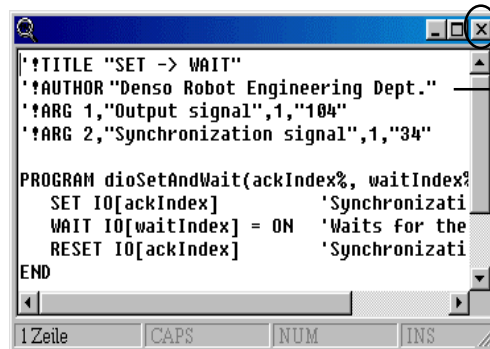
SCHRITT 3 Auswählen des hinzuzufügenden Programmes



Das Listenfeld "Programmname" zeigt die in der Klasse [Eingabe/Ausgabe] registrierten Programmtitel an.

④ Wählen Sie die in der Liste angezeigte Option [dioSetAndWait].

SCHRITT 4 Hinzufügen eines Programmes zu einem neuen Projekt



```
!TITLE "SET -> WAIT"
!AUTHOR "Denso Robot Engineering Dept."
!ARG 1,"Output signal",1,"104"
!ARG 2,"Synchronization signal",1,"34"

PROGRAM dioSetAndWait(ackIndex%, waitIndex%
SET IO[ackIndex]      'Synchronizati
WAIT IO[waitIndex] = ON 'Waits for the
RESET IO[ackIndex]    'Synchronizati
END
```

⑤ Klicken Sie [Zu Projekt hinzufügen] an.

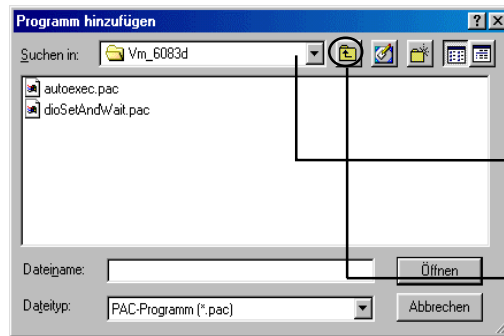
[dioSetAndWait] wird im neuen Projekt registriert und das Fenster [Bearbeiten] erscheint.

⑥ Klicken Sie hier, um das Fenster [Programmbank] zu schließen.
Das Fenster wird geschlossen.

17.2 Importieren von Programmen

Importieren Sie die beiden Programme, "pro1" und "pro2" in Ihr neues Projekt'. Wenn Sie WINCAPSII "Standard," installiert haben, befinden sich diese Programme im Verzeichnis
C:\Programmdateien\Wincaps2\Muster\Modellfall\.

SCHRITT 1



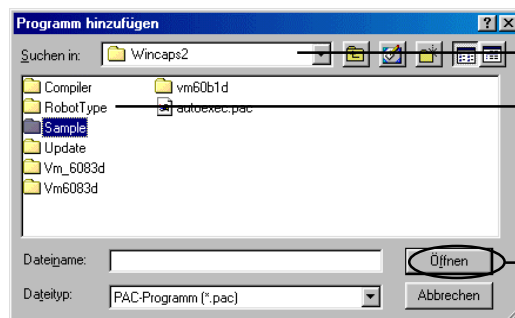
① Wählen Sie [Importieren] aus dem Menü "Programm" im Fenster [PAC Manager].

Das Dialogfenster [Programm hinzufügen] wird angezeigt.

Der Ordner [Lernprogramm] des aktuellen Systemordners wird am Bildschirm angezeigt.

② Klicken Sie die Schaltfläche  [Vorherig] an.

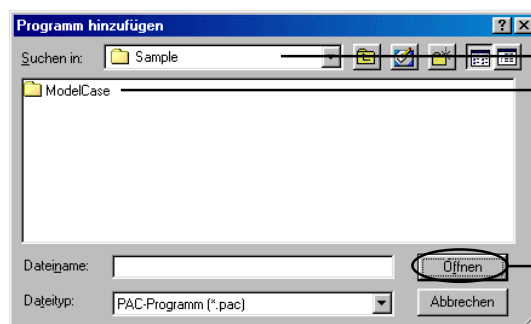
SCHRITT 2



Der Ordner [Wincaps2] wird im Fenster [Einsehen] angezeigt, das das gesuchte Programmverzeichnis enthält.

③ Wählen Sie den Ordner [Muster], und klicken Sie [Öffnen] an.

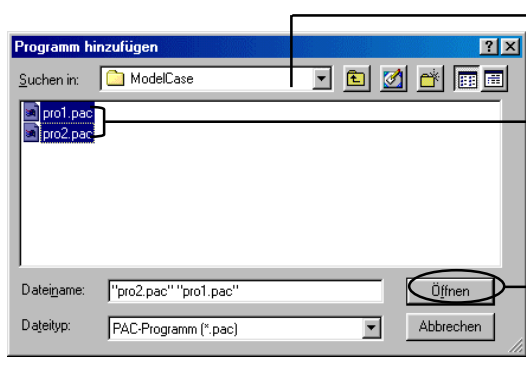
SCHRITT 3



Die Fensteranzeige [Einsehen] wechselt zum Ordner [Muster].

④ Wählen Sie den Ordner [Modellfall], und klicken Sie [Öffnen] an.

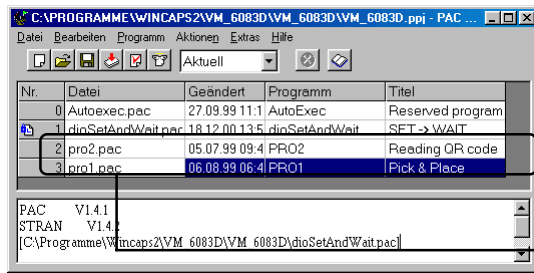
SCHRITT 4



Die Fensteranzeige [Einsehen] wechselt zum Ordner [Modellfall].

⑤ Wählen Sie die Symbole für die Dateien "PRO1.pa" und "PRO2.pac", und klicken Sie [Öffnen] an.

SCHRITT 5



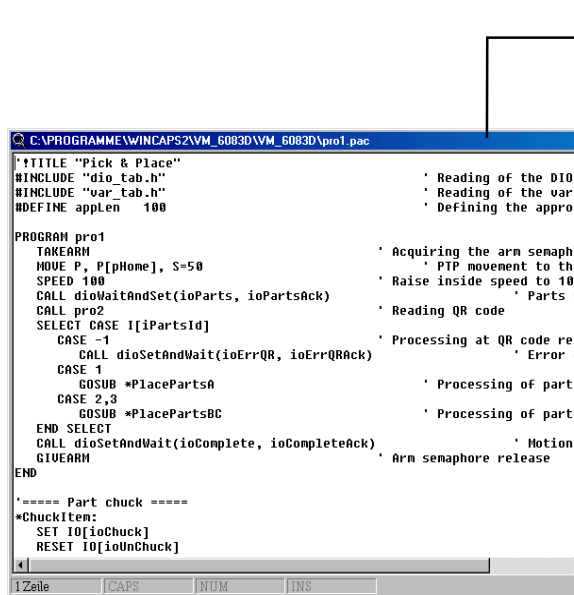
Die beiden Programme wurden zur Liste im Fenster [PAC-Manager] neu hinzugefügt.

⑥ Doppelklicken Sie in der Liste im Fenster [PAC Manager] auf [PRO1.pac].

★Tip★

Sie können auch das Programmbearbeitungsfenster öffnen, indem Sie im Menü [Aktion] [Einblenden] auswählen, nachdem Sie zum Markieren [PRO1.pac] angeklickt haben.

SCHRITT 6



Das Programm-bearbeitungsfenster für [PRO1.pac] wird angezeigt.

⑦ Stellen Sie sicher, daß im Bearbeitungsfenster keine Fehler vorhanden sind.

⑧ Stellen Sie sicher, daß im Bearbeitungsfenster des anderen Programmes [PRO2.pac] keine Fehler vorhanden sind, indem Sie die Schritte 4 bis 5 wiederholen.

⑨ Klicken Sie hier, um das Programmbearbeitungsfenster zu schließen.

Lektion 18 Beenden der Sitzung

Schließen Sie diese Sitzung von Kapitel 4 im PC und im Roboter-Controller'.

18.1 Beenden von WINCAPSII und Herunterfahren des PCs

SCHRITT 1



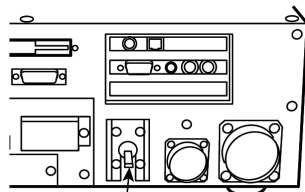
① Klicken Sie in der Ecke rechts oben des Fensters [DENSO System Manager] das Feld "Schließen" an.
Jede WINCAPSII-Software auf dem PC wird geschlossen.

SCHRITT 2

Falls erforderlich, beenden Sie das Betriebssystem (wie etwa Windows 95), und fahren Sie den PC herunter.

18.2 Ausschalten des Roboter-Controllers

SCHRITT 1



① Kippen Sie den Stromschalter nach unten, um den Roboter-Controller auszuschalten.

Kapitel 5

Funktionen des DENSO Roboters

In Kapitel 5 lernen Sie folgendes:

Lernen Sie die nützlichen Funktionen der DENSO Roboter, wie z. B. Konformitätskontrolle, kennen. Diese Funktionen dienen zur Senkung der Installationskosten und zur Verbesserung der Arbeitseffizienz in Vorbereitung und Praxis.

► Lektion 19 Konformitätskontrolle	118
19.1 Stromgrenzwert-Funktion für einzelne Achsen	118
19.2 Tip-Konformitäts-Funktion	119
► Lektion 20 Sonstige Funktionen	120

Lektion 19 Konformitätskontrolle

Durch die Konformitätskontrolle können Sie die Konformität von Robotern über die Software steuern. Diese Funktion kann während des Montagebetriebs beim Verarbeiten oder Verladen von Teilen auftretende Ausrichtungsfehler ausgleichen oder verhindern, daß an Roboter oder Werkstück zu viel Kraft eingesetzt wird.

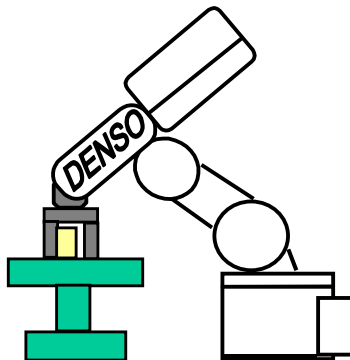
Zwei Arten von Konformitätskontrolle stehen zur Verfügung: Zum einen die Stromgrenzwert-Funktion, die die Konformität für einzelne Gelenke festlegt. Zum andern die Tip-Konformitäts-Funktion, die die Konformität für einzelne Elemente der Koordinaten am Ende des Roboterflansches (die mechanischen Schnittstellenkoordinaten) festlegt.

ANMERKUNG: Die Stromgrenzwert-Funktion ist ab Version 1.2 oder höher verfügbar. Die Tip-Konformitäts-Funktion ist bei der V*-D-Serie (mit Ausnahme von VM-6070D) ab Version 1.4 oder höher verfügbar.

19.1 Stromgrenzwert-Funktion für einzelne Achsen

Diese Funktion sorgt für Konformität an einzelnen Achsen, indem das Antriebsdrehmoment (Strom) jedes Achsenmotors begrenzt wird. Somit wird eine übermäßige Krafteinwirkung an Robotern oder Werkstücken sowie ein durch Überlast oder Überstromfehler ausgelöster Roboterstop verhindert.

Ausgleichen von Ausrichtungsfehlern beim Verarbeiten von Teilen



Prinzip der Stromgrenzwert-Funktion

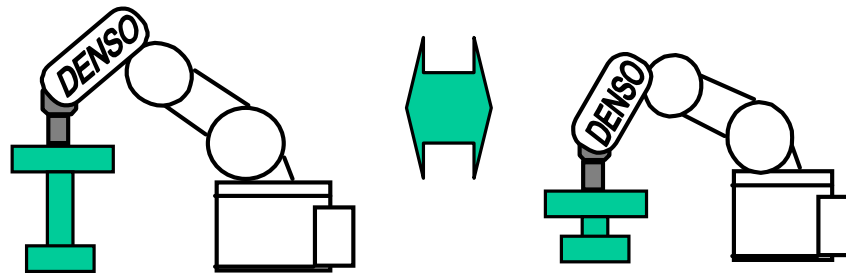
■ Aktivieren/Deaktivieren der Stromgrenzwert-Funktion

Sie können die Stromgrenzwert-Funktion aktivieren oder deaktivieren, indem Sie die Stromgrenzwert-Bibliothek, SetCurLmt bzw. ResetCurLmt ausführen. Ausführliche Hinweise zur Programmierung finden Sie im Handbuch "PROGRAMMIERUNG".

19.2 Tip-Konformitäts-Funktion

Durch Steuern des Antriebsdrehmoments (Strom) jedes Achsenmotors auf Basis der Kraftbegrenzung an der Spitze wird bei dieser Funktion die Konformität für die einzelnen Koordinatenelemente am Ende des Roboterflansches (Tip) festgelegt. Sie können die anzuwendenden Basis-, Werkzeug- oder Arbeitskoordinaten auswählen. Diese Funktion lässt den Roboter eine externe Kraft in angegebener Richtung berücksichtigen oder ein Objekt zur Höhenprüfung berühren.

Beachten einer externen Kraft oder Berühren eines Objektes zur Höhenprüfung



Prinzip der Tip-Kontrolle

■ Aktivieren der Tip-Konformitäts-Funktion über das Programmiergerät

Dies ist eine der erweiterten Funktionen. Sie müssen erweiterte Funktionen über das Programmiergerät aktivieren. Sobald Funktionen aktiviert sind, bleiben die Einstellungen nach Ausschalten des Controllers gespeichert.

■ Aktivieren/Deaktivieren der Tip-Konformitäts-Funktion

Sie können die Tip-Konformitäts-Funktion aktivieren oder deaktivieren, indem Sie die Konformitätskontroll-Bibliothek, SetCompControl bzw. ResetCompControl ausführen. Weitere Hinweise, siehe PROGRAMMIERHANDBUCH.

Lektion 20 Sonstige Funktionen

Neben der Konformitätskontrolle verfügen Denso Roboter über folgende Funktionen: Siehe PROGRAMMIERHANDBUCH für weitere Hinweise.

Name	Überblick über Funktionen und Befehle	Verwandte Befehle
Interferenzprüfungsbereich	Ein Ausgabesignal wird über den angegebenen E/A-Anschluß ausgegeben, wenn das Ende des Roboterarmes innerhalb des anvisierten Bereiches liegt. Diese Befehle sind wichtig, um die ursprüngliche Position und Arbeitsposition zu bestätigen.	BEREICH BEREICH EINSTELLEN BEREICH ZURÜCKSETZEN
Unterbrechungs-Stop-Funktion	Wenn ein Unterbrechungs-Signal des E/A-Anschlusses des Systems eingeschaltet ist, wird der aktuelle Operationsbefehl beendet und der nächste bearbeitet. Dieser Befehl ist wichtig, wenn Sie die Roboter-Stopposition über eine externe Eingabe ändern möchten. Die Distanz bis zum Stoppen des Roboters hängt jedoch von der Geschwindigkeitseinstellung des Operationsbefehles und der tatsächlichen Geschwindigkeit des Roboters ab.	UNTERBRECHUNG EIN/AUS
Parallele Ausführungsfunktion des Operations-/Nicht-Operationsbefehles	Der Nicht-Operationsbefehl kann während der Ausführung eines Operationsbefehles ausgeführt werden. Dieser Befehl ist wichtig, um die Zykluszeit zu verkürzen.	E/A-BLOCK EIN/AUS
Multitasking-Funktion	Mehr als ein Programm kann gleichzeitig ausgeführt werden. Es besteht die Möglichkeit, die Zykluszeit durch Prüfen der externen Sperre und Ausführen von Bildverarbeitungsbefehlen in anderen Programmen zu verkürzen.	AUSFÜHREN von KILL etc.
Programm-Debug-Funktionen	Es besteht die Möglichkeit, Meldungen anzuzeigen und ein akustisches Signal ertönen zu lassen. Diese Funktion ist hilfreich beim Debugging von Anwenderprogrammen.	DRUCKEN MSG DRUCKEN DBG SUMMER
Optimale Nutzlast-Einstellungsfunktion	Dadurch kann die Beschleunigungsgeschwindigkeit entsprechend der Masse des Werkstückes und des Endwerkzeugs eingestellt werden. Dadurch kann die Zykluszeit verkürzt werden.	aspACLD aspChange

Anhänge

▶	Anhang-1	Glossar	122
▶	Anhang-2	Bezeichnungen der Roboter-Controller-Teile	132
▶	Anhang-3	Bezeichnungen der Programmiergerät-Teile	134
▶	Anhang-4	Menüstruktur des Programmiergeräts	136

Appendix-1 Glossar

A

ABSOLUTBEWEGUNG

Die Bewegung zur vom Programmiergerät festgelegten Bewegungszielposition hin. ($\Leftrightarrow$ Relativbewegung)

ADRESSEINSTELLUNG (IP-Adresse)

So legen Sie die Controller-IP-Adresse fest. Dies ist bei der Ethernet-Kommunikation erforderlich.

AKTUELLE POSITION

Die aktuelle Position des Ursprungs der Werkzeugkoordinaten.

ANFAHRVEKTOR

Positiver Richtungsvektor der Z-Achse an mechanischen Schnittstellenkoordinaten.

ANHALTEN

Die Stop-Methode, um das Programm sofort zu stoppen. Der Motor ist nicht ausgeschaltet.

ANSTELLWINKEL

Der Drehwinkel um die Y-Achse.

ARBEITSKOORDINATEN

Das dreidimensionale, orthogonale Koordinatensystem, das den Ursprung für die durch den Roboter zu verarbeitende Arbeit festlegt.

ARM-DATEI

Die Datei, in der die spezifischen Informationen zu diesem Roboter erfaßt sind. Der Arm Manager verwendet die Datei.

ARM MANAGER

Die Software, welche die Roboterbewegung simuliert.

ARMKONFIGURATIONS- MAKRODEFINITIONSDATEI

Die Datei, die die Makrodefinitionsinformationen der Armeinstellungsdaten enthält.

ARM-SEMAPHOR

Das Privileg der Robotersteuerung. Der privilegierte Task kann den Roboter steuern.

ARMSTELL

Die Zahl, welche die Roboter-Armstellung anzeigt.

ARMSTELLUNG

Die Armstellung, die durch den Wert der ersten bis dritten Achsen eines Roboters mit 6 Achsen bestimmt wird. Es gibt zwei verschiedene Armstellungen: RECHTS und LINKS.

ARMSTELLUNG

Der mögliche Status jeder Achse (Gelenk) des Roboters. Mehrfach-Armstellungen sind für die gleiche Position und Stellung möglich.

ARMSTELLUNGS-KOMPONENTE

Die Komponente, welche die Armstellung bestimmt. Es gibt folgende 5 Komponenten bei Robotern mit 6 Achsen: Arm, Ellbogen, Handgelenk, die 6. und 4. Achse.

AUSFÜHRUNGSPROGRAMM

Das in ein für den Roboter verständliches Datenformat umgewandelte Programm.

AUSSCHALTEN DES MOTORS

So schalten Sie den Motor des Roboters aus.

AUSSCHALTEN DES ROBOTER- CONTROLLERS

So schalten Sie den Roboter-Controller aus.

AUTO AKTIVIEREN

Das Signal zum Aktivieren des Auto-Modus im Zustand EIN. Manueller Modus und Programmierprüfmodus sind im Zustand AUS möglich.

AUTOMATISCHE ROBOTER-AUSFÜHRUNG

So bewegen Sie den Roboter, indem Sie ein Programm ausführen.

B

BALKENTABELLE

Die Häufigkeitsrate des Helligkeitswertes in einem Fenster. (Bildverarbeitungs-Terminus)

BASIS

Der Teil zum Installieren der 1. Achse des Roboters.

BASIS-BEFESTIGUNGSFLÄCHE

Die Verbindungsfläche des Basis- und Installationsrahmens.

BASISKOORDINATEN

Das dreidimensionale, orthogonale Koordinatensystem mit dem Ursprung an der Roboterbasis.

Bediener

Eine der Benutzerebenen von WINCAPSII. Wichtige Parameter können nicht geändert werden. Kennwort-Eingabe ist nicht erforderlich.

BEDIENFELD

Das an den Controller fest angeschlossene Bedienfeld. Es hat keine Programmierfunktion.

BEDIENFELD

So schalten Sie den Betrieb des internen E/A über die Anzeige des Programmiergeräts EIN/AUS.

BEFEHL

Die in ein Programm geschriebene Anweisung. Der Controller liest Befehle in der Reihenfolge, wie sie in ein Programm geschrieben wurden, interpretiert und führt diese aus.

BEFEHLSBEREICH

Eine Gruppe von E/A-Anschlüssen, die den E/A-Befehlstyp angeben.

BEFEHLSVERARBEITUNG ABGESCHLOSSEN

Das Ausgabesignal meldet dem externen Gerät, daß die Verarbeitung des E/A-Befehls abgeschlossen ist.

BENUTZEREbene

Die für Benutzer vorgesehene Klasse, um die Sicherheit der Datenverwaltung zu gewährleisten. Der Zugriff auf Informationen oder Operationen ist durch jede Klasse eingeschränkt.

BENUTZERKOORDINATEN

Das Koordinatensystem, das Benutzer definieren können.

BER

Der Winkel, welcher das Verhältnis zwischen Roboter-Standardposition und mechanischer Bewegungsgrenze bestimmt.

BEREICH

Die Anzahl der weißen und schwarzen Pixel auf dem Bildschirm, wenn Bilddaten binär dargestellt werden. (Bildverarbeitungs-Terminus)

BETRIEBSMODUS

Der Modus, in dem der Roboter manuell bedient werden kann. Es gibt drei Modi: Modus "Jede Achse", X-Y-Modus und Werkzeug-Modus.

BETRIEBSPROTOKOLL

Die Aufzeichnung von Bedienfunktionen zur Verwendung des Programmiergeräts oder des Bedienfeldes.

BEWEGUNGSRAUM

Der Bereich, innerhalb dessen der Roboter gesteuert werden kann.

BEZEICHNUNG

So numerieren Sie den binärisierten weißen und schwarzen Bereich: (Bildverarbeitungs-Terminus)

BIBLIOTHEK

Die Sammlung von Programmen zwecks Wiederverwendung. Sie werden registriert und wiederverwendet über die Programmbank von WINCAPSII.

BINÄRISIERUNG

So ändern Sie die Helligkeit jedes Pixels anhand des Schwellwerts (Binärisierungs-Level) entweder auf weiß (0) oder schwarz (1).

BINÄRISIERUNGS-LEVEL

Der Schwellwert der Binärisierung. (Bildverarbeitungs-Terminus)

BREMSE AUS (Lösen der Bremsen)

So lösen Sie die Bremse jeder Achse.

BREMSE EIN (Verriegeln der Bremsen)

So betätigen Sie die Bremse jeder Achse.



CAL

Eine leichte Bewegung aller Achsen des Roboters, damit dieser die aktuelle Position nach dem Einschalten des Roboter-Controllers bestätigt.

CALSET

Kalibrierung des Verhältnisses zwischen aktueller Roboter-Position und den Positionsangaben des Controllers.

CALSET EINER EINZELNEN ACHSE

So führen Sie CALSET nur an der angegebenen Achse aus.

CODIERERWERT-PRÜFBEWEGUNG

Die Bewegung, die bestimmt, ob die Zielposition erreicht ist, wenn der Codiererwert innerhalb des angegebenen Impulsbereiches im Verhältnis zur durch die Programmierung festgelegten Bewegungszielposition liegt.

CP-STEUERUNG

Die Ausgleichssteuerung, um aus dem Pfad von der aktuellen Position zur Bewegungszielposition eine gerade Linie oder einen Kreis zu machen. (⇔ PTP-Steuerung)



DATENBEREICH

Eine Gruppe von E/A-Anschlüssen, um die notwendigen Daten des E/A-Befehls anzugeben.

DEFINIEREN DES INTERFERENZBEREICHES

So definieren Sie den Interferenzbereich: Er wird entweder über das Programmiergerät, in WINCAPSII oder über den Programmbefehl festgelegt.

DEFINIEREN VON WERKZEUGKOORDINATEN

So definieren Sie Werkzeugkoordinaten: Die Ursprungs-Offset- und Drehwinkelsummen an jeder Achse werden im Verhältnis zu den mechanischen Schnittstellenkoordinaten festgelegt. WERKZEUG 1 bis 63 kann festgelegt werden.

DIO MANAGER

Die Software, welche den E/A-Zustand überwacht und die E/A-Zuordnung verwaltet.

DISKRIMINATIONS-ANALYSEMETHODE

Die Methode zum Festlegen des Binärisierungs-Levels einer Balkentabelle anhand der statistischen Methode. (Bildverarbeitungs-Terminus).

DOPPELT

Eine der Armstellungen der 6. Achse eines Roboters mit 6 Achsen. (⇔ EINFACH)

DOPPELT4

Eine der Armstellungen der 4. Achse eines Roboters mit 6 Achsen. (⇔ EINFACH4)

D-VARIABLE (Doppelpräzisionsvariable)

Die Variable, die den Wert einer wirklichen Doppelpräzisionsnummer (15 Ziffern der effektiven Präzision) aufweist.

DURCHLAUFBEWEGUNG

Die Bewegung zum Durchlauf nahe der durch das Programmiergerät festgelegten Bewegungszielposition.



E/A

Das Eingabe- und/oder Ausgabesignal.

E/A-BEFEHL

Der Prozeßbefehl, der durch das externe Gerät über den E/A-Anschluß erteilt wird. Die entsprechend diesem Befehl ausgeführten Roboter-Controller-Prozesse.

E/A-BEFEHLSAUSFÜHRUNGSSIGNAL

Das Eingabe/Ausgabe-Signal des Systems meldet Peripherie/externen Geräte die Ausführung des E/A-Befehls und den Ausführungsstatus.

E/A-BENUTZERSIGNALE

Die Eingabe-/Ausgabesignale, die durch das Benutzerprogramm gesteuert werden können.

E/A-SYSTEMSIGNALS

Die Eingabe-/Ausgabesignale des Systems, um die Ausführungssteuerung oder -bedingung an externe Geräte zu melden.

EINSCHALTEN DES MOTORS

So schalten Sie den Motor des Roboters ein.

EINSCHALTEN DES ROBOTER-CONTROLLERS

So schalten Sie den Roboter-Controller ein.

EINSTELLRAD

Die Anzeige auf dem Programmiergerät, die zum Bewegen des Cursors oder zum Auswählen eines Pfads auf dem Eingabebildschirm dient.

ELLBOGENARMSTELLUNG

Die Armstellung, die durch den 2. und 3. Achsenwert eines Roboters mit 6 Achsen bestimmt wird. Es gibt zwei Arten von Ellbogen-Armstellungen: Oben und Unten.

ENDBEWEGUNG

Die Bewegung, die bestimmt, ob die Zielposition erreicht ist, wenn die angegebene Servo-Position mit der durch die Programmierung festgelegten Bewegungszielposition übereinstimmt.

ETHERNET-KARTE

Eine der optionalen Controller-Karten. Sie wird zur Kommunikation mit WINCAPSII über ein TCP/IP-Protokoll verwendet.

EXTERNE AUTOMATISCHE AUSFÜHRUNG

So führen Sie ein Programm über ein externes Gerät aus.

EXTERNE BESCHLEUNIGUNG

Der Beschleunigungswert wird über das Programmiergerät festgelegt. Der Prozentwert für die Maximalbeschleunigung wird eingegeben.

EXTERNE GESCHWINDIGKEIT

Die Geschwindigkeit wird über das Programmiergerät festgelegt. Der Prozentwert für die Maximalgeschwindigkeit wird eingegeben.

EXTERNER MODUS

Der Modus, in dem Roboter-Operationen über ein externes Gerät möglich sind.

EXTERNE VERZÖGERUNG

Der Verzögerungswert wird über das Programmiergerät festgelegt. Der Prozentwert für die Maximalbeschleunigung wird eingegeben.

ERSTER ARM

Der Roboterarm, welcher der Basis am nächsten ist.



FEHLERCODE

Ein 4-ziffriger Hexadezimalcode, der Fehlerursachen/-bedingungen beschreibt, die im Roboter oder in WINCAPII aufgetreten sind. Weitere Informationen zur Bedeutung jedes Fehlercodes entnehmen Sie der Fehlercodetabelle.

FEHLERPROTOKOLL

Aufzeichnung des Fehlers und der Zeit, zu der er aufgetreten ist.

FENSTER

Der verfügbare Speicher zur Verarbeitung von Bildern. (Bildverarbeitungs-Terminus)

FERNSTEUERUNGSBETRIEB

So steuern Sie den Roboterarm, der auf dem Arm Manager angezeigt wird.

FUNKTIONSTASTEN

Die Schaltflächen unter dem Bildschirm des Programmiergeräts. Funktionsnamen werden im unteren Bereich des Bildschirms angezeigt und die Funktion wird durch Betätigen der Schaltfläche ausgeführt.

FUNKTION ZUR OPTIMALEN LADEKAPAZITÄTSEINSTELLUNG

Die Funktion, die die optimale Geschwindigkeit und Beschleunigung zur Ladebedingung und Stellung des Roboters festlegt.

F-VARIABLE (Gleitpunkt-Variable)

Die Variable, die den Wert einer wirklichen Einfachpräzisionsnummer (7 Ziffern der effektiven Präzision) aufweist



GELENKMODUS

Der Modus, in dem der Roboter an jeder Achse manuell bedient werden kann.

GLOBALE VARIABLE

Die für jeden Task zur Verfügung stehende Variable.



HAND (Endwerkzeug)

Der Teil zur Unterbrechung der Arbeit. Das gleiche als Werkzeug.

HANDGELENK-ARMSTELLUNG

Die Armstellung, die durch den Wert der 4. und 5. Achse eines Roboters mit 6 Achsen festgelegt wird. Es gibt zwei Arten von Handgelenk-Armstellungen: KIPPEN und NICHT KIPPEN.

HAUPTACHSE

Die Achse, die das Mindest-Trägheitsmoment bei einer Umdrehung des Objekts auf einer Ebene angibt. (Bildverarbeitungs-Terminus)

HAUPTACHSE - WINKEL

Der durch die Horizontal- und Hauptachse gebildete Winkel. (Bildverarbeitungs-Terminus)

HELLIGKEIT

Ein numerischer Wert (0-255), der die Helligkeit jedes Pixels anzeigt. (Bildverarbeitungs-Terminus)



INITIALISIERUNGS-DISKETTE

Die Diskette, auf der die Starteinstellung des Roboters ab Werk gespeichert ist. Sie wird zur Wiederherstellung der Anfangseinstellungen

verwendet, sobald ein Fehler im Controller-Speicher auftritt.

INSTALLATIONS-FRAME

Die Plattform zur Installation des Roboters.

INTEGRIERTER HELBIGKEITSWERT

Der Wert, welcher sich aus dem Gesamtwert der Helligkeit aller Pixel auf einem Bildschirm ergibt. (Bildverarbeitungs-Terminus)

INTERFERENZBEREICH

Der vom Benutzer bereitgestellte Bereich, um zu prüfen, ob die Installation vom Werkzeug beeinträchtigt wird. Wenn der Ursprung der Werkzeugkoordinaten in diesem Bereich liegt, wird das Ausgabesignal über den angegebenen E/A-Anschluß ausgegeben.

INTERNE AUTOMATISCHE AUSFÜHRUNG

So führen Sie ein Programm über das Bedienfeld oder das Programmiergerät aus.

INTERNE BESCHLEUNIGUNG

Die in einem Programm festgelegte Beschleunigung.

INTERNE GESCHWINDIGKEIT

Die in einem Programm festgelegte Geschwindigkeit.

INTERNER MODUS

Der Modus, in dem Ausführung und Programmierung des Roboters über das Bedienfeld oder das Programmiergerät möglich sind.

INTERNE VERZÖGERUNG

Die in einem Programm festgelegte Verzögerung.

I-VARIABLE (Ganzzahlige Variable)

Die Variable, die einen Ganzzahlwert aufweist.



J-VARIABLE (Gelenk-Variable)

Die durch den Wert jeder Achse angegebene Variable.



KANTE

Übergangspunkt der Helligkeit. (Bildverarbeitungs-Terminus)

KIPPEN

Eine der Handgelenk-Armstellungen eines Roboters mit 6 Achsen. (⇔ NICHT KIPPEN)

KOMMENTAR

Erklärende Anmerkungen zu einem Programm, damit es leichter verständlich ist. Der Controller führt keinen Kommentar aus.

KOMMUNIKATION EINSTELLEN

So legen Sie die Einsatzbedingungen (Kommunikationsgeschwindigkeit etc.) jedes Kommunikationsanschlusses des Roboter-Controllers fest.

KOMMUNIKATIONSERLAUBNIS EINSTELLEN

So legen Sie die Einsatzerlaubnis jedes Kommunikationsanschlusses des Roboter-Controllers fest.

KOMMUNIKATIONSPROTOKOLL

Die Aufzeichnung der Kommunikation zwischen PC und Roboter.

KOMPATIBLER MODUS

Der Modus, in dem die E/A-Zuordnung so festgelegt wird, um kompatibel zur konventionellen Serie von Robotern zu sein. Dies wird über die Software gesteuert.

KONVENTIONELLE SPRACHE

Bei der für den Denso Roboter verwendeten Sprache handelt es sich um eine konventionelle Roboter-Sprache.



LADEN

So lesen Sie Programme, Armdaten etc. von Diskette in den Roboter-Controller ein.

LAUFENDER START

Die Startmethode zur Ausführung eines Programmes bei der Iteration. Die Operation geht weiter, bis sie gestoppt wird.

LINKS

Eine der Armstellungen eines Roboters mit 6 Achsen. (⇔ RECHTS)

LOG MANAGER

Die Software, die die Aufzeichnung von Operationen, Fehlern etc. des Roboters auf einem PC kopiert und verwaltet.

LOKALE VARIABLE

Die Variable, die bei einem Task verwendet wird.



MAKRO

Die Definition von Namen mit 12 Buchstaben in bezug auf Variablen- und Anschlußnummern. Namen werden bei der Programmausführung durch Nummern ersetzt.

MAKRODEFINITIONSDATEI

Die Datei, die ein Makro definiert.

MANUELLER ROBOTERBETRIEB

Der Roboterbetrieb durch den Benutzer über Bedientasten des Programmiergeräts oder des Bedienfeldes.

MASCHINENVERRIEGELUNG

Der Zustand einer durch den Roboter-Controller simulierten Bewegung, ohne daß eine aktuelle Roboterbewegung vorliegt.

MECHANISCHE BEWEGUNGSGRENZE

Die durch den mechanischen Stopper gesetzte mechanische Bewegungsbegrenzung. ($\Leftrightarrow$ Softwarebegrenzung)

MECHANISCHE FLACHSCHNITTSTELLE

Der Anteil zur Installation von Werkzeugen am oberen Ende des Roboterarmes.

MECHANISCHE SCHNITTSTELLE

Die Verbindungsfläche des Flansches und Werkzeugs. Mechanische Schnittstelle (JIS)

MECHANISCHE SCHNITTSTELLENKOORDINATEN

Das dreidimensionale, orthogonale Koordinatensystem mit dem Ursprung in der Mitte des Flansches.

MECHANISCHER STOPPER

Der Mechanismus zur physikalischen Begrenzung der Bewegung der Roboterachsen.

MENÜSTRUKTUR

Die Beschreibung des Funktionsmenüs der Funktionstasten in Form einer Baumstruktur, aufgeführt im Bedienungshandbuch.

MODUSMETHODE

Die Methode zur Festlegung des Binärisierungs-Levels am Tiefpunkt, wenn es sich um eine Balkentabelle mit zweigipfliger Verteilung handelt.

MODUS-SCHALTER

Der Schalter am Programmiergerät. Damit kann der Roboter-Ausführungsmodus aktiviert werden.

MOMENTAUFNAHME

Die Funktion zur Aufzeichnung des aktuellen Status des Roboters.

MONITOR

So zeigen Sie den aktuellen Roboterstatus an.

MULTITASKING

Der Zustand, in dem mehrere Programme praktisch gleichzeitig ausgeführt werden können. Dies wird möglich, indem die CPU des Roboter-Controllers jedes Programm in kurzen Intervallen abwechselnd ausführt.



NICHT KIPPEN

Eine der Handgelenk-Armstellungen eines Roboters mit 6 Achsen. ($\Leftrightarrow$ KIPPEN)

NICKWINKEL

Der Drehwinkel um die Z-Achse.

NLIM

Der negative, direktionale Endwert der Softwarebegrenzung. ($\Leftrightarrow$ PLIM)

NORMALMODUS

Der Standard-E/A-Zuordnungsmodus.

NORMALVEKTOR

Positiver Richtungsvektor der X-Achse an mechanischen Schnittstellenkoordinaten.



OBEN

Eine der Ellbogen-Armstellungen eines Roboters mit 6 Achsen. ($\Leftrightarrow$ UNTEN)

ORIENTIERUNGSVEKTOR

Positiver Richtungsvektor der Y-Achse an mechanischen Schnittstellenkoordinaten.

OVERHEAD-VERSION

Der angegebene Roboter ist entsprechend zu installieren, da er von der Maximaleinstellung der Basis oben und der Armstellung unten abhängt. Da auf dieser Arbeitsplattform kein Installationsabstand erforderlich ist, kann der Arbeitsabstand größer sein.



PAC (PAC)

Neue, für den Denso Roboter verwendete Roboter-Sprache. Sie ist aufwärtskompatibel zu SLIM. (Industrie-Robotersprache JIS)

PAC-PROGRAMM-MANAGER

Die Software, die die Entwicklung des PAC-Programmes unterstützt. Editor-, Befehlseditor- und Programmbank-Funktionen sind integriert.

PALETTIERUNG

So legen Sie Teile auf eine Palette mit Unterteilungen oder entnehmen diese.

PIXEL

Der Punkt, aus dem sich die Bildschirmanzeige aufbaut. (Bildverarbeitungs-Termini)

PLIM

Der negative Richtungs-Endwert der Softwarebegrenzung. (⇔ NLIM)

POSITIONSDATEN

Die Daten der Basiskoordinaten, welche gleichzeitig die Position der Roboterflanschmitte (das Werkzeug am oberen Ende, wenn die Werkzeugdefinition wirksam ist) und der Roboterstellung beschreiben.

PRIORITÄT

Die Reihenfolge der Task-Ausführung nach Wichtigkeit. Das Programm mit höherer Priorität wird zuerst ausgeführt.^{1`}

Programmierer

Eine der Benutzerebenen von WINCAPSII. Alle gemeinsamen Operationen sind möglich. Kennwort-Eingabe ist erforderlich, um in diesen Modus zu gelangen.

PROGRAMMIERPRÜFUNG

So prüfen Sie die Bewegung durch das Programm.

PROGRAMMIERUNG

So erfolgt über das Programmiergerät die Eingabe der notwendigen Bedienungsinformationen in den Roboter.

PROGRAMMRÜCKSETZUNG

Das Eingabesignal, um die Programmausführung oben im Programm zu aktivieren.

PROGRAMSTART

Das Eingabesignal zum Starten eines Programmes. Wenn es sich um einen

Schrittstop handelt, beginnt die Ausführung mit dem nächsten Schritt. Beim Anhalten beginnt sie mit dem folgenden des gleichen Schrittes.

PROGRAMMÜBERTRAGUNG

So senden/empfangen Sie Roboter-Programme zwischen Roboter-Controller und WINCAPSII (PC).

PROTOKOLL

Die Aufzeichnung von Operationen, Bewegungen etc. des Roboters. Es gibt vier Arten von Protokollen: Fehler-, Betriebs-, Steuer- und Kommunikationsprotokoll.

PTP-STEUERUNG

Die Steuerung, die den Roboterarm ohne Ausgleich zur Zielposition bewegt. Beim Pfad muß es sich nicht unbedingt um eine gerade Linie handeln. (⇔ CP-Steuerung)

P-TYLE-METHODE

Die Methode zur Einstellung des Binärisierungs-Levels, um den Bereich des Objekts und des schwarzen (oder weißen) Anteils identisch anzugleichen. (Bildverarbeitungs-Terminus)

P-VARIABLE (Positionsvariable)

Die durch Position, Stellung und Armstellung gekennzeichnete Variable.



RECHTS (RECHTS)

Eine der Armstellungen eines Roboters mit 6 Achsen. (⇔ LINKS)

RELATIVBEWEGUNG

Die Bewegung von der aktuellen Position weg, entsprechend der durch das Programmiergerät festgelegten Bewegungssumme.

ROBOTERFEHLER

Das Ausgabesignal, das im Roboter aufgetretene Fehler wie Servo-Fehler, Programm-Fehler etc. meldet.

ROBOTERSTOP

Die Stop-Methode, um Programme sofort zu stoppen und den Motor auszuschalten.

ROBOTERWARNUNG

Das Ausgabesignal, das einen beim E/A-Befehl oder bei der Servo-Verarbeitung aufgetretenen leichten Fehler meldet.

ROLLWINKEL

Der Drehwinkel um die X-Achse.

RX-KOMPONENTE

Der Wert des Drehwinkels um die X-Koordinatenachse.

RY-KOMPONENTE

Der Wert des Drehwinkels um die Y-Koordinatenachse.

RZ-KOMPONENTE

Der Wert des Drehwinkels um die Z-Koordinatenachse.



SCHRITTPRÜFUNG

Eine Schrittausführung des Programmes im Programmierprüfmodus.

SCHRITTSTOP

Die Stop-Methode, um ein Programm nach einer Schrittausführung zu stoppen.

SCHWERPUNKT

Der Gleichgewichtspunkt, an dem das Objektgewicht auf einer Ebene balanciert. (Bildverarbeitungs-Terminus)

SEMAPHOR

Das Task-Ausführungs-Privileg, das zum Synchronisieren von Tasks dient oder zur Ausführung exklusiver Steuerungen zwischen Tasks, die nicht gleichzeitig ausgeführt werden dürfen.

SERVO EIN

Das Signal, das externen Geräten meldet, daß der Motor eingeschaltet ist.

SICHERHEITSSCHALTER

Der Schalter, über den der Roboter so lange bewegt werden kann, bis eine der Tasten für die Armbewegung seitwärts aus Sicherheitsgründen gleichzeitig gedrückt wird. Der Roboter stoppt sofort, wenn entweder die Taste für die Armbewegung seitwärts oder der Sicherheitsschalter gelöst wird.

SINGULÄRER PUNKT

Die Position an der Grenze der beiden Armstellungen.

SOFTWAREBEGRENZUNG

Die Begrenzung des durch die Software festgelegten Roboterbewegungsbereiches. ($\Leftrightarrow$ mechanische Bewegungsgrenze)

SPEICHERN

So speichern Sie Programme, Armdaten etc. des Roboter-Controllers auf Diskette.

SPEICHERKAPAZITÄT

Der Gesamtspeicherwert des Werkzeugs und der Arbeit, die der Roboter speichern kann.

STATUS BEI NICHT ANGESCHLOSSENEM PROGRAMMIERGERÄT

So steuern Sie den Roboter über ein externes Gerät, wenn Bedienfeld oder Programmiergerät nicht angeschlossen sind.

STATUSBEREICH

Eine Gruppe von Ausgabesignalen, um das Ergebnis der E/A-Befehlsverarbeitung zu melden. Der Status wird entsprechend des E/A-Befehls festgelegt.

STELLUNG

Die Neigung des Werkzeuges, die bei Robotern mit 6 Achsen durch Nick-, Anstell- und Rollwinkel festgelegt ist. Die Werkzeug-Richtung, die bei Robotern mit 4 Achsen durch den Winkel um die Z-Achse festgelegt ist.

STEUERPROTOKOLL

Die Aufzeichnung des angegebenen Wertes, Codierungswertes, aktuellen Wertes und des Ladeverhältnisses. Sie werden durch jede Bewegungsachse aufgezeichnet.

STOP-TASTE

Eine der Tasten des Programmiergeräts. Durch Betätigen der Taste werden alle Programme sofort angehalten.

SUBROUTINE

Das Programm, das eine spezifische Bewegung beschreibt und von einem Teil des Hauptprogrammes aufgerufen wird.

SUCHEN

So suchen Sie den Abstand, der zu standardisierten Bilddaten paßt (Suchmodell): Bildverarbeitungs-Terminus

SYSTEM-MANAGER

Die Software, die im allgemeinen alle Informationen von WINCAPSII.

SYSTEMPROJEKT

Programme und verwandte Datengruppen, die durch den System-Manager verwaltet werden.

SYSTEMVARIABLE

Die Variable zur Prüfung der Systembedingungen eines Programmes.



TAKTSIGNAL

Das Eingabesignal, um den Start der E/A-Befehlsverarbeitung anzuweisen.

TASK

Der durch jedes Programm sich ergebende Bewegungsprozeß, wenn mehrere Programme ihre gleichzeitige Ausführung verwalten.

TÄGLICHE PRÜFUNG

Die Prüfung vor der täglichen Arbeit.

TOOL

Der Teil des Roboters, der unmittelbar für die Arbeit zuständig ist. Es handelt sich um ein Synonym des Endwerkzeugs (JIS).

T-VARIABLE (homogene Transformationsvariable)

Die durch den Positions-, Richtungs-, Annäherungsvektor und die Armstellung angezeigte Variable.

TYPDEKLARATION

So deklarieren Sie den Variablentyp in einem Programm.



UNTEN

Eine der Ellbogen-Armstellungen eines Roboters mit 6 Achsen. (⇔ OBEN)

UNTERBRECHUNG ÜBERSPRINGEN

Das Eingabesignal, das die Operation des aktuellen Schrittes unterbricht, wenn es bei Ausführung eines Roboterbefehles eingeschaltet ist, und die Ausführung des nächsten Schrittes beginnt.



VARIABLENTABELLE

Eine Gruppe von Daten, die zu jeder durch den Controller gespeicherten Anschlußnummer bzw. zu jedem Anschlußwert passen.

VISUELLE FUNKTION

Das Gerät, das die Robotersteuerungsfunktion mit notwendigen Daten versorgt, indem die von einer Kamera gelieferten Bilder verarbeitet werden.

VISUELLES GERÄT

Das Gerät, das den Roboter mit notwendigen Daten versorgt, indem die von einer Kamera gelieferten Bilder verarbeitet werden.



WERKZEUGKOORDINATEN

Das Koordinatensystem, das den Ursprung im Werkzeug festlegt und den Ursprung der mechanischen Schnittstellenkoordinaten zu einem beliebigen Punkt ausgleicht und um jede Achse dreht.

WERKZEUG-MODUS

Der manuelle Betriebsmodus an den Werkzeugkoordinaten.

WERKZEUG0

Eine Sonderform der Werkzeugdefinition mit Ursprungs-Offset 0, d. h. die mechanischen Schnittstellenkoordinaten werden impliziert.



X-Y-MODUS

Der manuelle Betriebsmodus an den Basiskoordinaten.

**ZWEITER ARM**

Der von der Basis gemessen weiter entfernte Arm der Roboterarme.

ZYKLUSSTOP

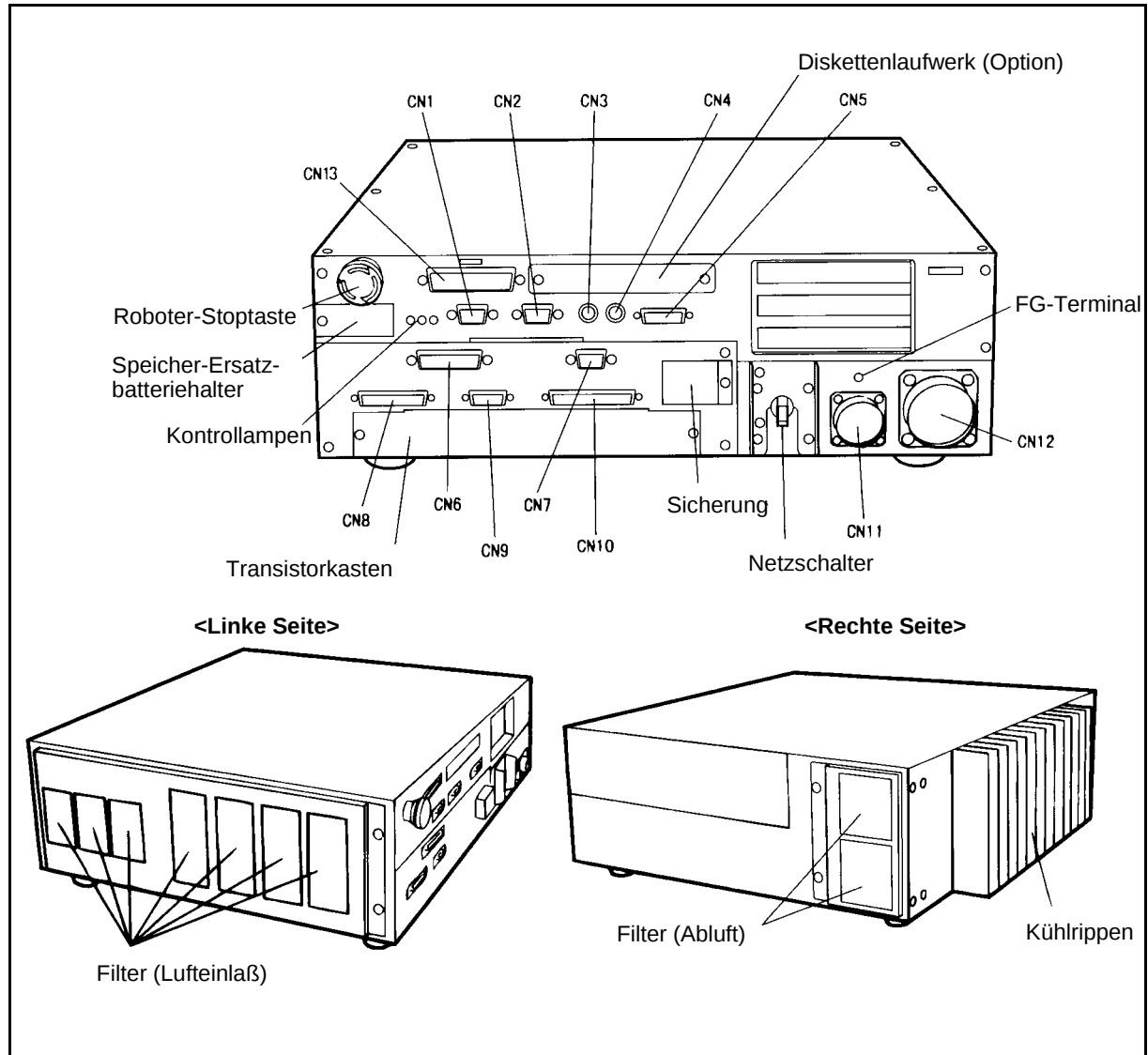
Die Stop-Methode, um ein Programm nach einer Zyklusausführung zu stoppen.

SYMBOLE**μBildverarbeitung**

Visuelles von DENSO hergestelltes Gerät.

Appendix-2 Bezeichnungen der Roboter-Controller-Teile

Auf der Abbildung und Tabelle unten sehen Sie die Bezeichnungen der Komponenten des Roboter-Controllers.



Bezeichnungen der Roboter-Controller-Teile (VS-D-Serie)

Anschlußnamen (VS-D-Serie)

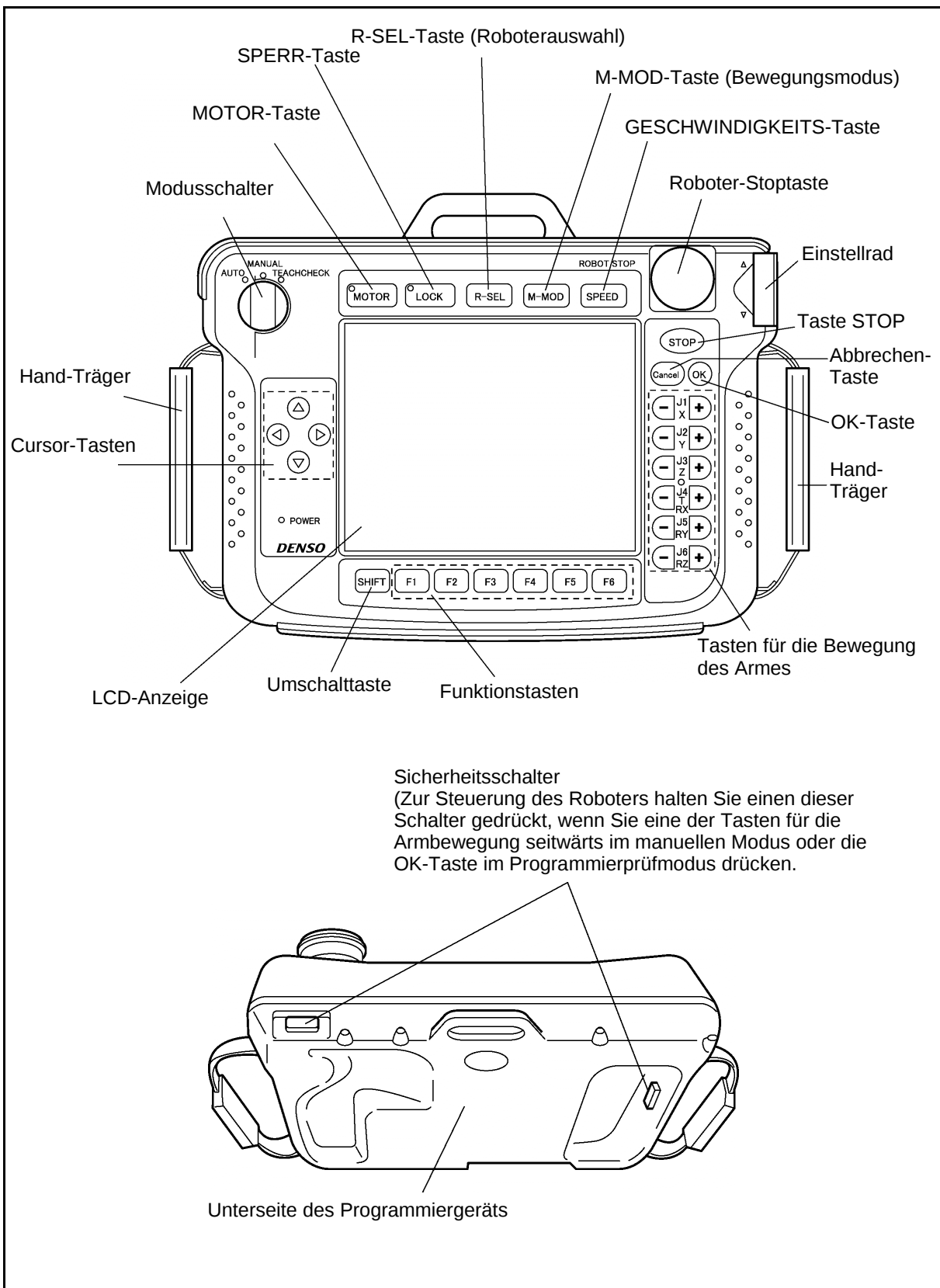
Anschluß Nr.	Kennzeichnung	Name	Anschluß Nr.	Kennzeichnung	Name
CN1	RS232C	Anschluß für serielle Schnittstelle	CN8	Eingabe	Anschluß für Benutzereingabe oder Systemeingabe
CN2	CRT	Anschluß für CRT	CN9	HAND E/A	Anschluß für Endwerkzeug E/A
CN3	TASTAT	Anschluß für Tastatur	CN10	AUSGABE	Anschluß für Benutzerausgabe oder Systemausgabe
CN4	MAUS	Anschluß für PS/2-Maus	CN11	EINGANG AC	Stromanschluß
CN5	PROGRAMMIERGERÄT	Anschluß für Programmierger	CN12	MOTOR	Anschluß für Motor
CN6	PRINTER	Anschluß für Drucker	CN13	CODIERER	Anschluß für Codierer
CN7	E/A-POWER	Stromanschluß für E/A			

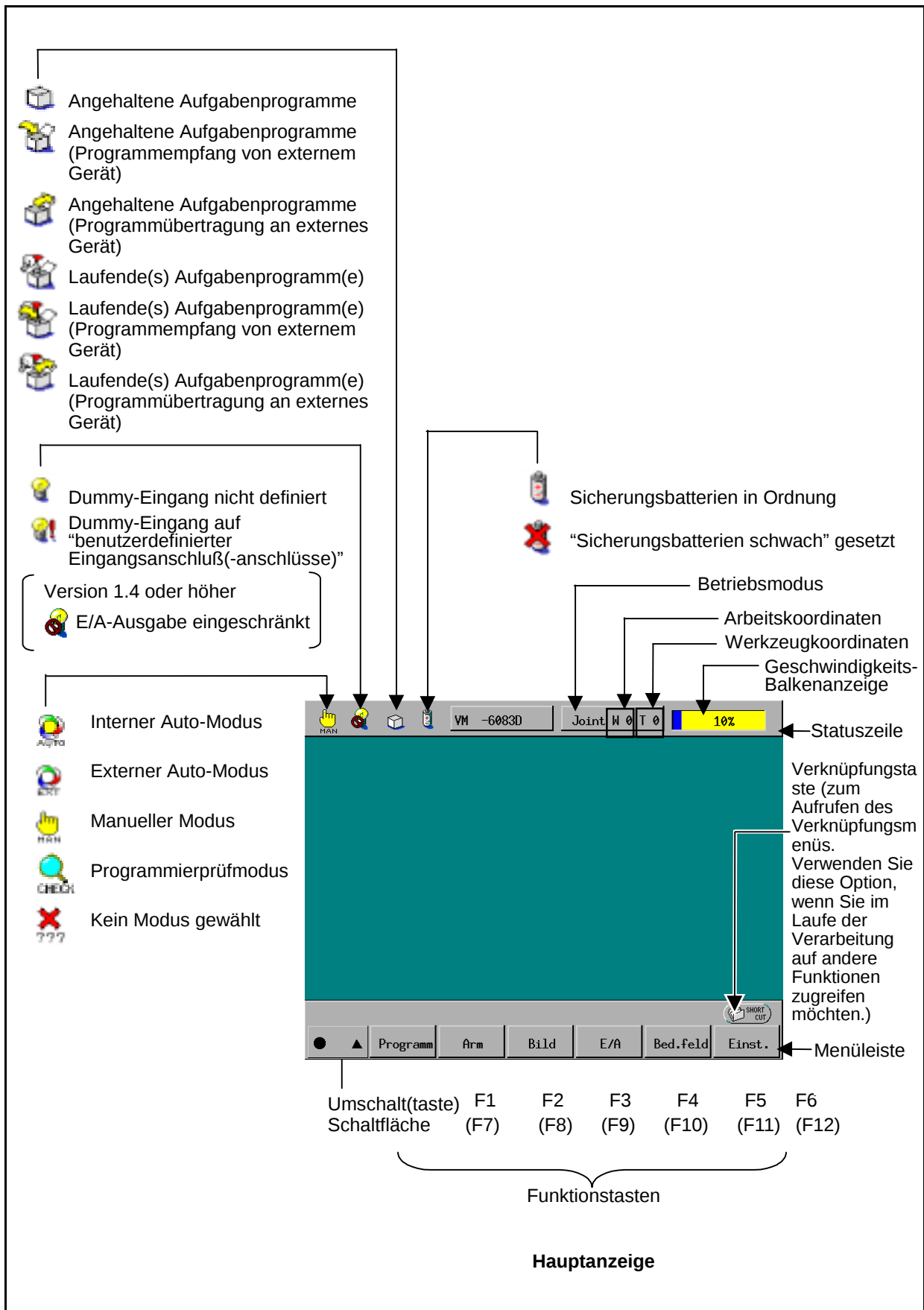


Vorsicht: Die Roboter-Controller-Anschlüsse haben eine Schraubsicherung oder eine Ringsicherung. Sichern Sie die Stecker richtig. Ist nur ein Anschluß nicht richtig eingerastet, kann durch unzureichenden Kontakt ein Fehler auftreten.

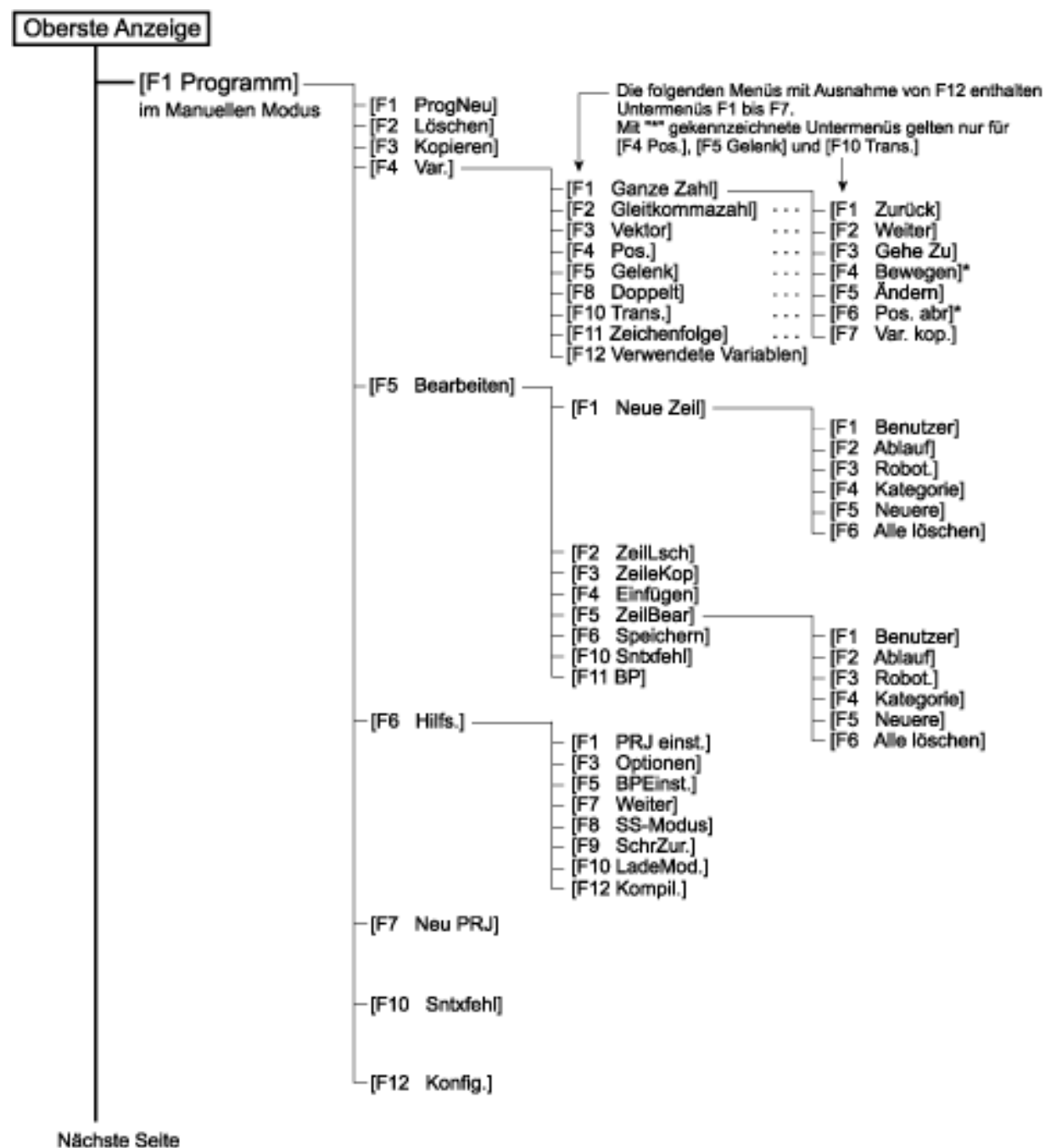
Das Anschließen oder Entfernen des Stromanschlusses bzw. des Motoranschlusses während die Stromversorgung des Roboter-Controllers eingeschaltet ist, kann die inneren Schaltkreise des Roboter-Controllers beschädigen. Stellen Sie den Netzschalter auf AUS bevor sie diese Anschlüsse einstecken oder trennen.

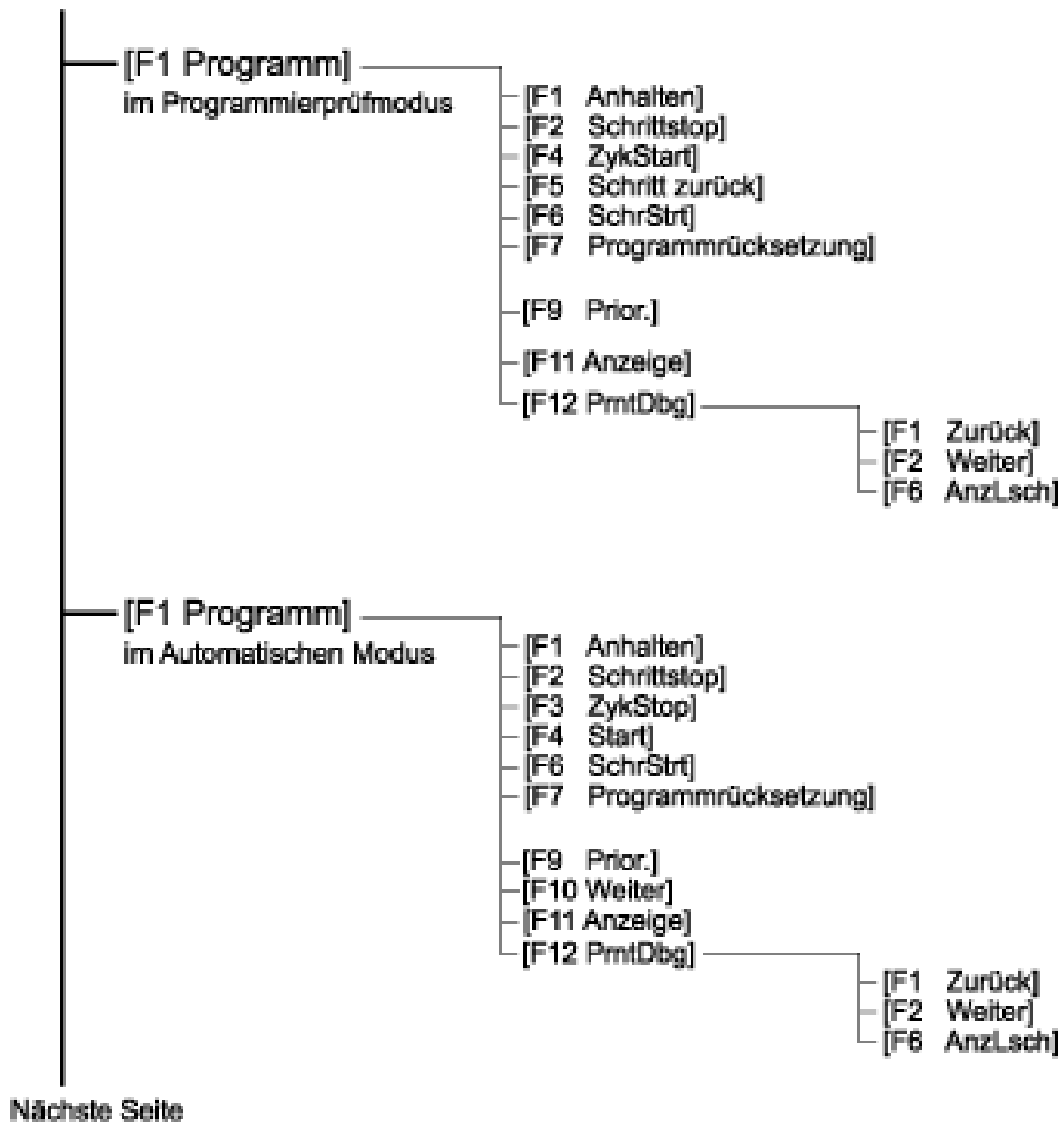
Appendix-3 Bezeichnungen der Komponenten des Programmiergeräts

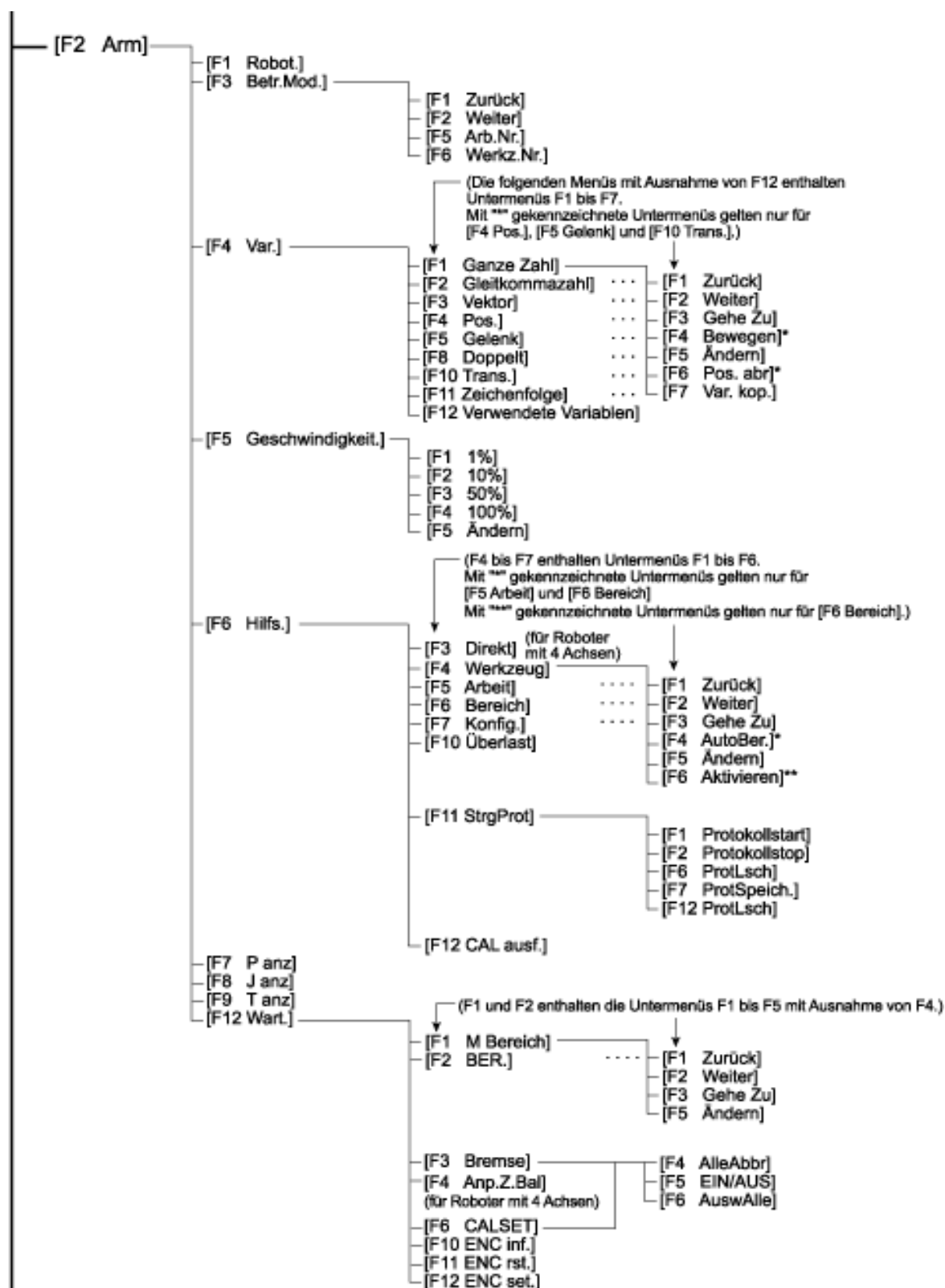




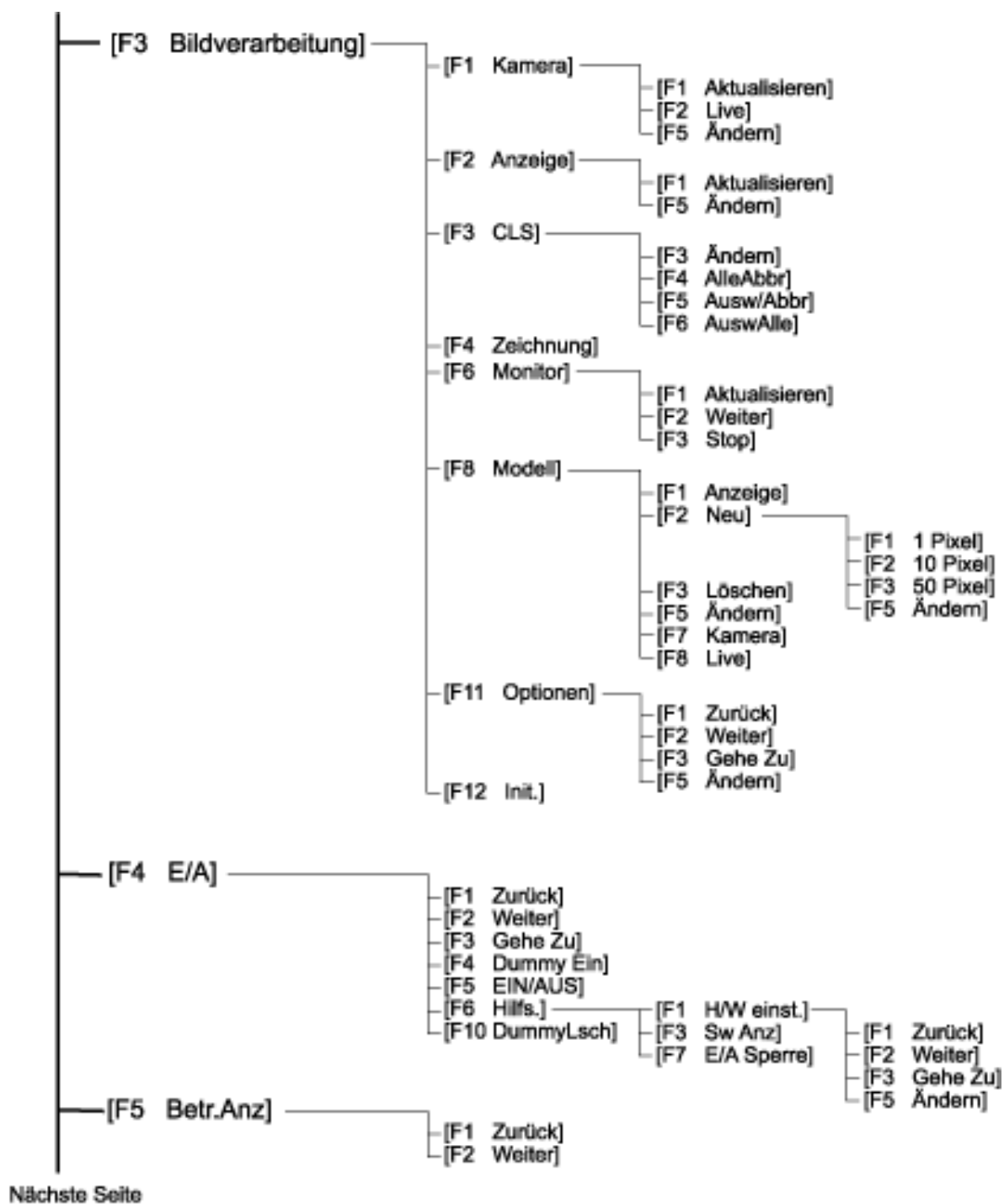
Appendix-4 Menüstruktur des Programmiergeräts

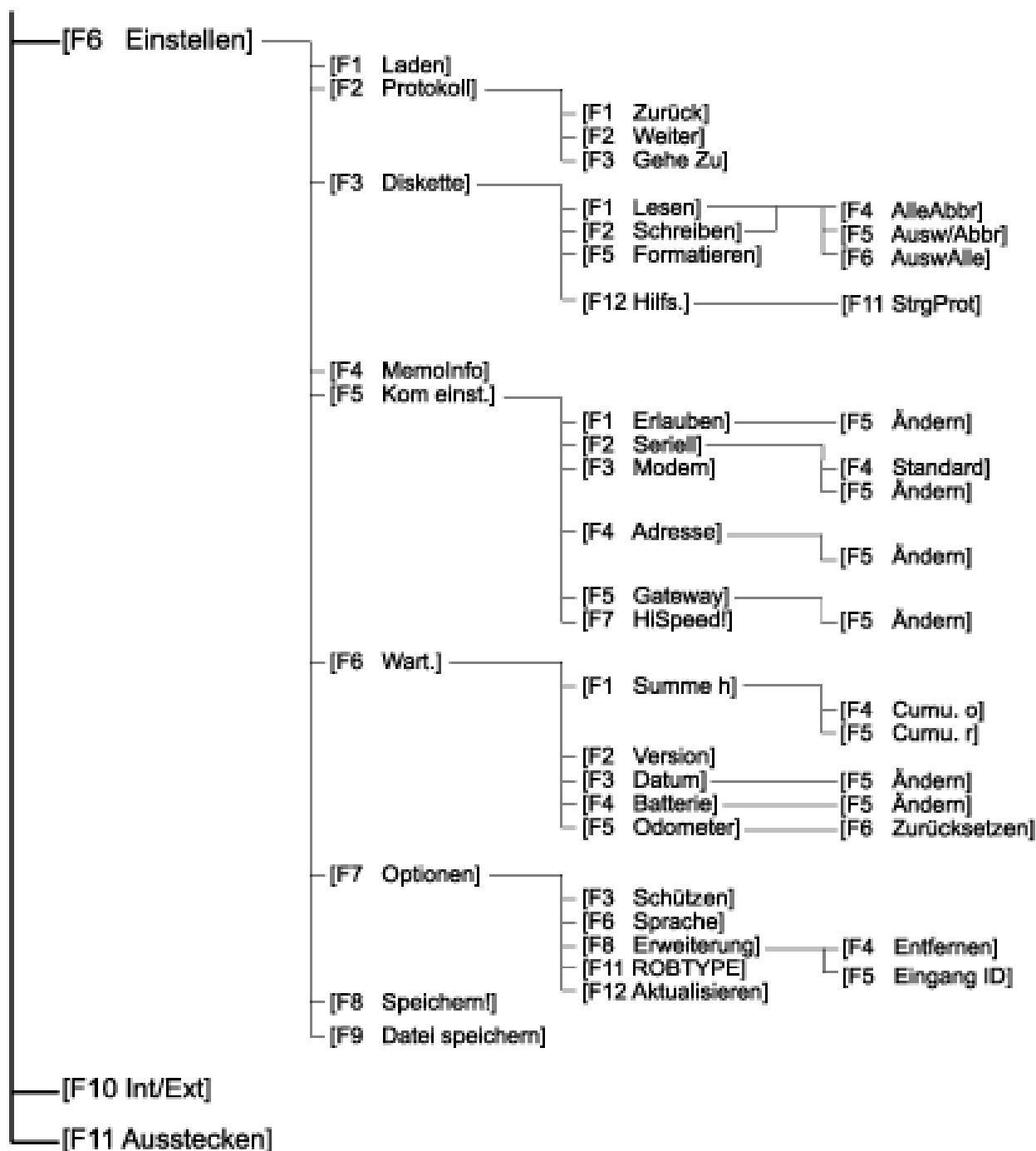






Nächste Seite





Vertikal beweglicher Roboter V*-D-SERIE

EINFÜHRUNGS-HANDBUCH

Erste Auflage

Juni 2000

DENSO CORPORATION
Industrial Systems Product Division

6B20C

Zweck dieses Handbuchs ist es, genaue Informationen für die Handhabung und Bedienung des Roboters bereitzustellen. Falls Sie Fehler oder Auslassungen feststellen, teilen Sie uns dies bitte mit; ebenso sind Vorschläge zur allgemeinen Verbesserung dieses Handbuchs gerne willkommen.

DENSO übernimmt keinerlei Haftung für direkte oder indirekte Schäden, die durch die Anwendung der Informationen in diesem Handbuch entstehen.

