

# ***DENSO ROBOTER***

Horizontal ausgerichtet

**H\*-D-SERIE**

**EINFÜHRUNGSHANDBUCH**

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 Roboterсерien und/oder -modelle werden in diesem Handbuch behandelt:**

Kleine und mittlere, horizontal ausgerichtete Roboter der H★-D-SERIE

---

## **Wichtig!**

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

---

# **Der Dokumentationssatz\*-D ist wie folgt aufgebaut**

Der H\*-D-Dokumentationssatz besteht aus folgenden sechs Handbüchern. Falls Sie mit dieser Roboterbaureihe nicht vertraut sind, lesen Sie bitte alle sechs Handbücher vor der Arbeit mit dem Roboter durch.

## **EINFÜHRUNGSHANDBUCH - vorliegendes Handbuch:**

Stellt Ihnen den 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.

## **EINSTELLUNGSHANDBUCH**

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

## **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.

## **PROGRAMMIERUNGSHANDBUCH**

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

## **FEHLERCODETABELLEN**

Enthält eine Auflistung der Fehlercodes, die am Programmiergerät, Bedienfeld oder PC-Bildschirm angezeigt werden, wenn ein Fehler in der Roboterbaureihe oder in WINCAPSII auftritt.

# Das vorliegende Handbuch ist wie folgt aufgebaut:

Dieses Handbuch ist nur ein Teil des H\*-D-Dokumentationssatzes. Es besteht aus den SICHERHEITS- UND VORSICHTSMASSNAHMEN, den Kapiteln 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 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, um den Roboter entsprechend der Programme zu steuern und beschreibt die Palettierung, eine der Hauptanwendungen bei Robotern mit 4 Achsen. Ebenso wird beschrieben, wie PAC-Bibliotheken eingesetzt werden können, um die Effizienz der Task-Programmentwicklung besonders zu verbessern.

### **Kapitel 5 Funktionen von Denso Robotern**

Beschreibt die Stromgrenzwert-Funktion, die direkte Programmierfunktion und weitere Funktionen des Denso Roboters.

## **Anhänge**

Anhang-1 Glossar

Anhang-2 Bezeichnung der Teile des Roboter-Controller

Anhang-3 Bezeichnung der Teile des Programmiergeräts

Anhang-4 Menü-Baumstruktur am Programmiergerät



# SICHERHEITS- UND VORSICHTSMASSNAHMEN

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.

|                                                                                                   |                                                                                                                                                                |
|---------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  <b>WARNUNG</b>  | Warnung vor Situationen, die zu schweren körperlichen Schäden oder Tod führen können, wenn die Anleitungen nicht korrekt befolgt werden.                       |
|  <b>VORSICHT</b> | Warnung vor Situationen, die zu geringerem körperlichen Schaden oder erheblichem Sachschaden führen können, 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:** Bezieht sich auf den Teil des Sicherheitsraums, innerhalb dessen ein Roboter durch Software-Bewegungsbegrenzungen eingeschränkt 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 Aufgabenprogrammes 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.

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 installiert werden.

### 2.2 Servicebereich

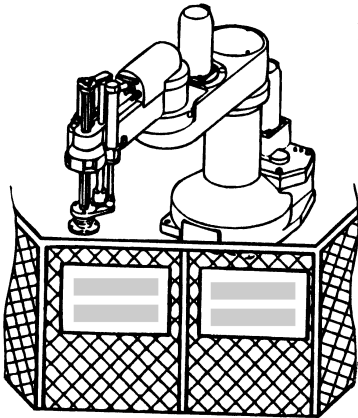
Der Roboter und die Peripheriegeräte sind so zu installieren, daß ein ausreichender Servicebereich für das sichere Programmieren, die Wartung und die Prüfung zu Verfügung steht.

## **SICHERHEITS- UND VORSICHTSMASSNAHMEN**

---

- |                                                                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>2.3 Steuergeräte außerhalb des Sicherheitsraums des Roboters</b>                   | 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.                                                                                                                                                                                    |
| <b>2.4 Anordnung von Meß- und Anzeigegeräten</b>                                      | Druckmesser, Öldruckmesser und andere Anzeiger sind an einem leicht zu kontrollierenden Ort zu montieren.                                                                                                                                                                                                                                                                                                                                                                            |
| <b>2.5 Schutz der Elektroinstallation und der Hydraulik-/ Druckluft-Rohrleitungen</b> | Die Elektroinstallationen und Hydraulik-/Druckluft-Rohrleitungen sind durch eine Abdeckung oder eine entsprechende Maßnahme zu schützen, wenn die Möglichkeit einer Beschädigung besteht.                                                                                                                                                                                                                                                                                            |
| <b>2.6 Anordnung der NOT-AUS-Schalter</b>                                             | <p>Die NOT-AUS-Schalter sind an einer leicht erreichbaren Stelle anzubringen, um den Roboter ggf. unverzüglich anhalten zu können.</p> <ul style="list-style-type: none"><li>(1) Die NOT-AUS-Schalter müssen rot sein.</li><li>(2) Die NOT-AUS-Schalter müssen so konstruiert sein, daß sie nach dem Auslösen weder automatisch noch versehentlich durch eine Bedienperson freigegeben werden können.</li><li>(3) NOT-AUS-Schalter müssen vom Hauptschalter getrennt sein.</li></ul> |
| <b>2.7 Anordnung der Betriebszustandsanzeigen</b>                                     | 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. Sollte dies nicht möglich sein, verwenden Sie andere Schutz Einrichtungen 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 konstruieren, 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 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 Sie 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 Mitarbeiter den Sicherheitsraum des Roboters betreten, und er 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 gut sichtbaren 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 Benutzerhandbuches "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 empfohlen den Reinigungsvorgang automatisch durchführen zu lassen.

### 2.13 Beleuchtung

Für den sicheren Roboterbetrieb muß ausreichende Beleuchtung vorhanden sein.

### 2.14 Schutz vor Gegenständen, die vom Endwerkzeug weggeschleudert werden

Für die Kalkulation der Verletzungsgefahr für das Personal 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 ff 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 eingestellt ist, daß die Spannung des Roboterantriebs beim Betreten des Sicherheitsraums des Roboters abgeschaltet wird.
- 3) Wenn es erforderlich ist, den Sicherheitsraum des Roboters für die Durchführung von Programmierungs- oder Wartungsarbeiten zu betreten, während der Roboter aktiv ist, so stellen Sie sicher, daß die in Abschnitt 3.3 "Gewährleistung der Sicherheit des Personals bei der Durchführung von Arbeiten im Sicherheitsraum des Roboters" beschriebenen Schritte befolgt werden.

#### 3.1 Erstellen von Arbeitsrichtlinien und Sicherstellen der Einhaltung durch das Personal

Für den Fall, daß es erforderlich wird, den Sicherheitsraum des Roboters zu betreten, um die Programmierung oder Wartungsinspektionen durchzuführen, legen Sie "Arbeitsrichtlinien" für die folgenden Punkte fest und stellen Sie sicher, daß diese vom Personal befolgt 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) 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 sicher, daß diese "Arbeitsrichtlinien" im Einklang mit dem Robotertyp, dem Aufstellungsort und den Arbeitsaufgaben stehen.

Sorgen Sie dafür, daß beim Ausarbeiten dieser "Arbeitsrichtlinien" die Meinung des betroffenen Personals, der Techniker des Herstellers der Ausrüstungen und eines Arbeitsschutzbeauftragten berücksichtigt wird.

### **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 der Programmierung, sind evtl. Reparaturen sofort nach dem Erkennen einer Störung sowie weitere 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 des 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 Pneumatikbauteilen 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 Automatikbetrieb**

#### **(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 Automatikbetrieb 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 Sicherheitsraum des Roboters zur Durchführung der Notfall-Wartung zu betreten. 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-Controller zum Auswechseln der Batterie oder aus anderen Gründen ist die Stromversorgung des Roboter-Controller 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

|                                                                                                |            |
|------------------------------------------------------------------------------------------------|------------|
| <b>Vorwort .....</b>                                                                           | <b>i</b>   |
| <b>Der Dokumentationssatz*-D ist wie folgt aufgebaut .....</b>                                 | <b>ii</b>  |
| <b>Das vorliegende Handbuch ist wie folgt aufgebaut: .....</b>                                 | <b>iii</b> |
| <b>SICHERHEITS- UND VORSICHTSMASSNAHMEN .....</b>                                              | <b>1</b>   |
| <br>                                                                                           |            |
| <b>Kapitel 1 Steuern des Roboters über das Programmiergerät .....</b>                          | <b>1</b>   |
| 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         |
| <br>                                                                                           |            |
| <b>Kapitel 2 Erstellen eines Programmes auf einem PC mit WINCAPSII .....</b>                   | <b>39</b>  |
| Lektion 3 Einstellen des Roboter-Controller mit dem Programmiergerät .....                     | 40         |
| 3.1 Ausführen der Kalibrierung (CAL) .....                                                     | 40         |
| 3.2 Aktivieren der Maschinenverriegelung des Roboter-Controller .....                          | 43         |
| 3.3 Einstellen des Kommunikationsanschlusses des Roboter-Controller .....                      | 44         |
| Lektion 4 Starten von WINCAPSII und Erstellen eines Systemprojektes .....                      | 48         |
| 4.1 Starten des System Manager .....                                                           | 48         |
| 4.2 Registrieren eines neuen Systemprojektes .....                                             | 50         |
| 4.3 Einstellen des Kommunikationsanschlusses des PC .....                                      | 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 von Programmcodes .....                                                           | 57         |
| 6.4 Einsetzen des Befehlseditors .....                                                         | 58         |
| 6.5 Speichern des Programmes .....                                                             | 60         |
| Lektion 7 Kompilieren des Programmes in ein Laufzeitformat .....                               | 61         |
| Lektion 8 Hochladen des Programmes (PC → Roboter-Controller) .....                             | 62         |
| <br>                                                                                           |            |
| <b>Kapitel 3 Simulieren der Roboterbewegung auf einem PC mit dem erstellten Programm .....</b> | <b>65</b>  |
| Lektion 9 Vorbereiten des PC auf die Simulation .....                                          | 67         |
| 9.1 Starten des Arm Manager .....                                                              | 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 Zuweisen der aktuellen Position zur Positionsvariablen .....                              | 69         |
| 10.3 Bearbeiten der Positionsvariablen .....                                                   | 71         |
| Lektion 11 Testlauf des Programmes .....                                                       | 73         |
| 11.1 Laden des Programmes .....                                                                | 73         |
| 11.2 Starten des Programmes .....                                                              | 75         |

|                                                                                                              |            |
|--------------------------------------------------------------------------------------------------------------|------------|
| Lektion 12 Überwachen und Manipulieren der E/As .....                                                        | 76         |
| 12.1 Starten des DIO Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller .....       | 76         |
| 12.2 Überwachen der E/As .....                                                                               | 77         |
| 12.3 Ein-/Ausschalten der E/A-Dummy-Schalter .....                                                           | 78         |
| Lektion 13 Überwachen und Manipulieren der Variablen .....                                                   | 79         |
| 13.1 Starten des Variablen Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller ..... | 79         |
| 13.2 Überwachen der Variablen .....                                                                          | 80         |
| Lektion 14 Kontinuierlicher Testlauf .....                                                                   | 81         |
| 14.1 Kontinuierliche Ausführung .....                                                                        | 81         |
| 14.2 Kontinuierliche Überwachung der E/As .....                                                              | 82         |
| 14.3 Sofortiger Stop .....                                                                                   | 84         |
| <b>Kapitel 4 Steuern des Roboters über Programme .....</b>                                                   | <b>85</b>  |
| Lektion 15 Steuern des Roboters in der Praxis .....                                                          | 86         |
| 15.1 Lösen der Maschinenverriegelung .....                                                                   | 86         |
| 15.2 Einstellen der Roboter Geschwindigkeit und -beschleunigung .....                                        | 87         |
| 15.3 Einschalten des Antriebsmotors .....                                                                    | 88         |
| 15.4 Programmierung .....                                                                                    | 89         |
| 15.5 Starten von Programmen .....                                                                            | 92         |
| 15.6 Wechseln der Roboter Geschwindigkeit .....                                                              | 93         |
| 15.7 Stoppen des Roboters bei kontinuierlicher Ausführung .....                                              | 94         |
| Lektion 16 Palettierung .....                                                                                | 98         |
| Lektion 17 Effektiver Einsatz von Bibliotheken .....                                                         | 106        |
| 17.1 Programmbank .....                                                                                      | 107        |
| 17.2 Importieren von Programmen .....                                                                        | 109        |
| Lektion 18 Beenden der Sitzung .....                                                                         | 111        |
| 18.1 Beenden von WINCAPSII und Herunterfahren des PC .....                                                   | 111        |
| 18.2 Ausschalten des Roboter-Controller .....                                                                | 111        |
| <b>Kapitel 5 Funktionen des Denso Roboters .....</b>                                                         | <b>113</b> |
| Lektion 19 Strombegrenzung .....                                                                             | 114        |
| Lektion 20 Direkte Programmierung .....                                                                      | 116        |
| 20.1 Auswählen des direkten Modus .....                                                                      | 116        |
| 20.2 Programmieren der Stellungsvariablen .....                                                              | 119        |
| Lektion 21 Sonstige Funktionen .....                                                                         | 121        |
| <b>Anhänge .....</b>                                                                                         | <b>123</b> |
| Appendix-1 Glossar .....                                                                                     | 124        |
| Appendix-2 Bezeichnungen der Roboter-Controller-Teile .....                                                  | 134        |
| Appendix-3 Bezeichnungen der Komponenten des Programmiergeräts .....                                         | 136        |
| Appendix-4 Menüstruktur des Programmiergeräts .....                                                          | 138        |

# 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 ausführen:
  - 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 Sicherheitsprogrammierung
  - Durchführen einer Sicherheitsprogrammierprüfung
  - Sicheres Starten und Stoppen von Programmen

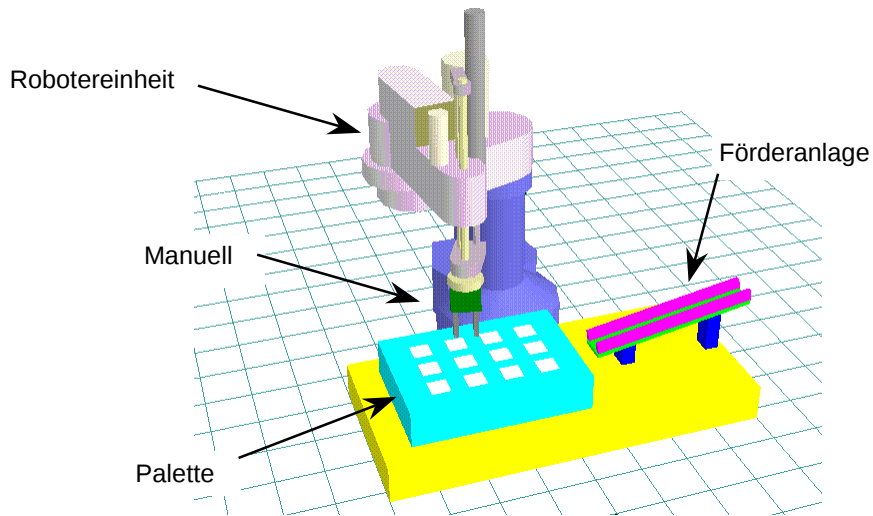
---

|   |                                                                   |    |
|---|-------------------------------------------------------------------|----|
| ► | 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 Praktische Übung                                        | 38 |

---

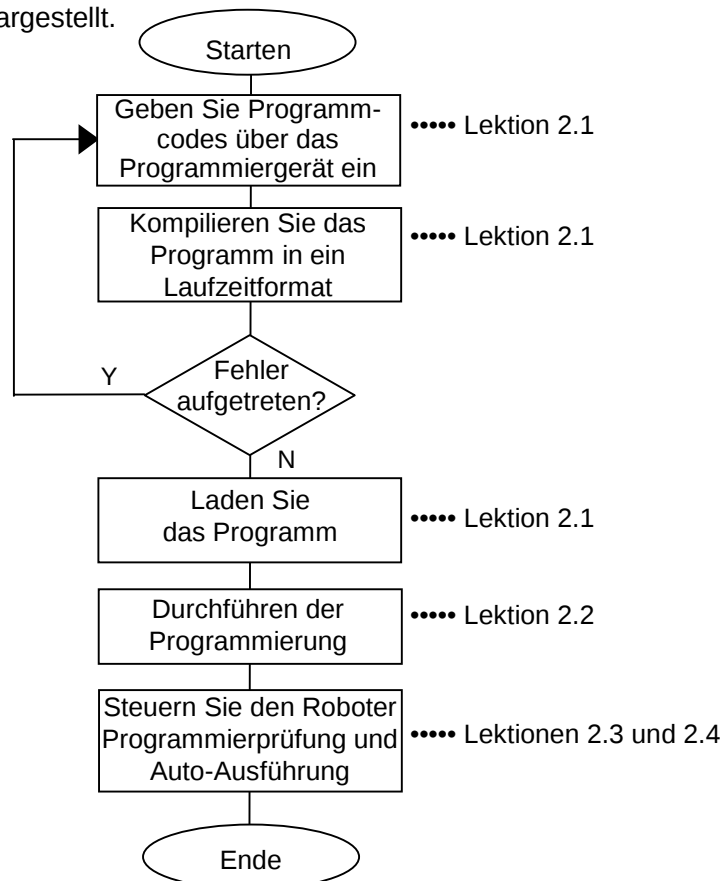
## Beispiel für eine Geräteeinstellung

Auf der Abbildung unten ist ein Beispiel für eine Geräteeinstellung 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

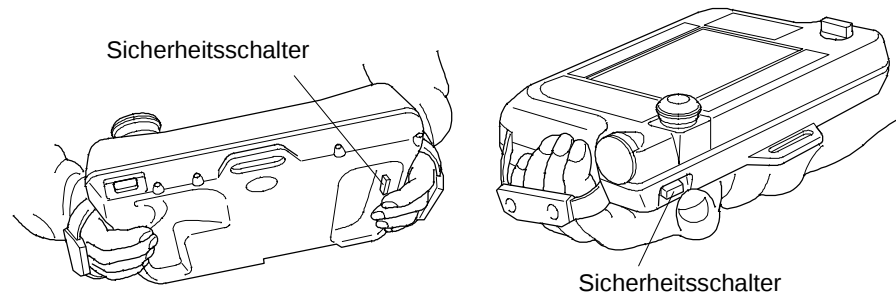
---

## 1.1 Grundfunktionen des Programmiergeräts

### Halten des Programmiergeräts und des Sicherheitsschalters

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

Das Programmiergerät verfügt über zwei Sicherheitsschalter. Daher 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 während 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:

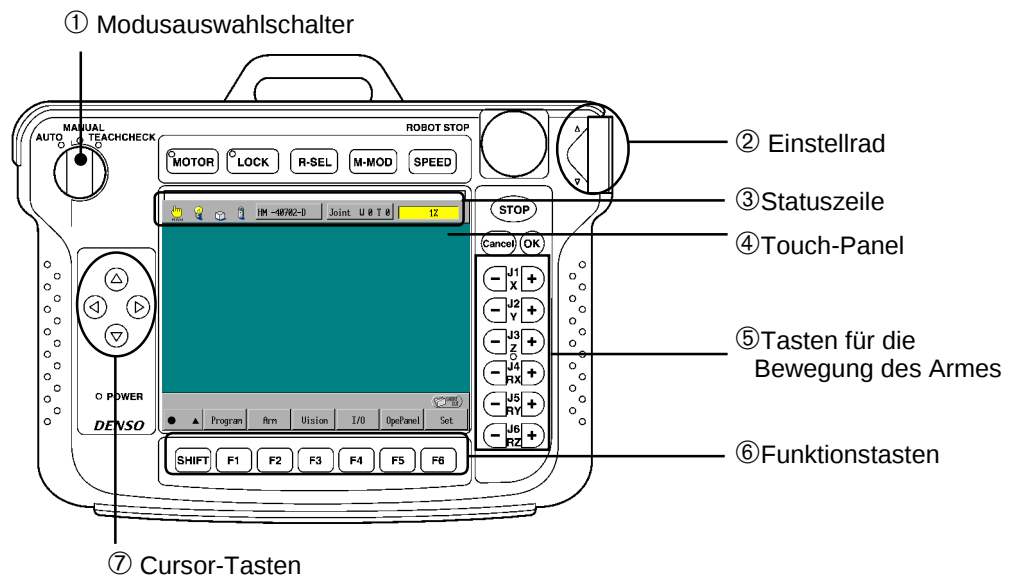
- 1) Wenn der Schalter nicht oder nur leicht gedrückt wird  
→ Schalter: AUS
- 2) Wenn der Schalter mit korrektem Druck gedrückt wird  
→ Schalter: EIN
- 3) Wenn der Schalter zu stark gedrückt wird  
→ Schalter: AUS

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

Mit diesem Schalter kann zwischen den Betriebsmodi "Auto, Manuell und Programmierprüfung" hin- und hergeschaltet werden.

#### ② Einstellrad

Damit 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.

**Vorsicht:** 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 erforderlich, den Sicherheitsschalter gleichzeitig gedrückt zu halten.

#### ⑥ Funktionstasten

Normalerweise werden F1 bis F6 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.

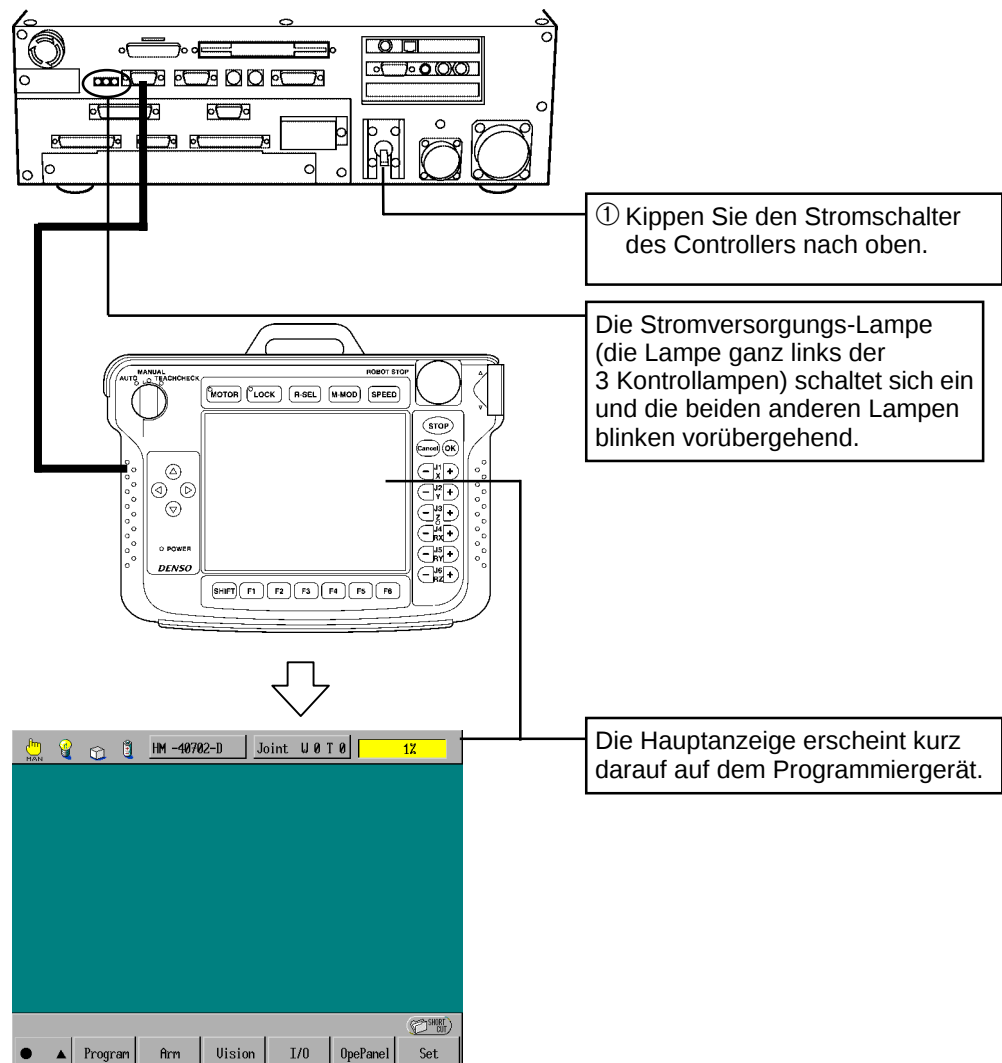
## 1.2 Manuelles Steuern des Roboters über das Programmiergerät

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

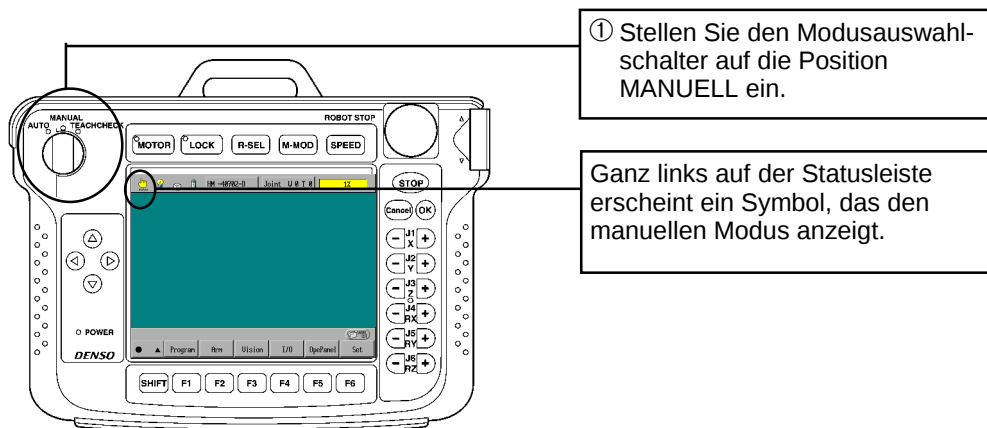
### SCHRITT 1 Vor dem Fortfahren prüfen, ob alles sicher ist

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

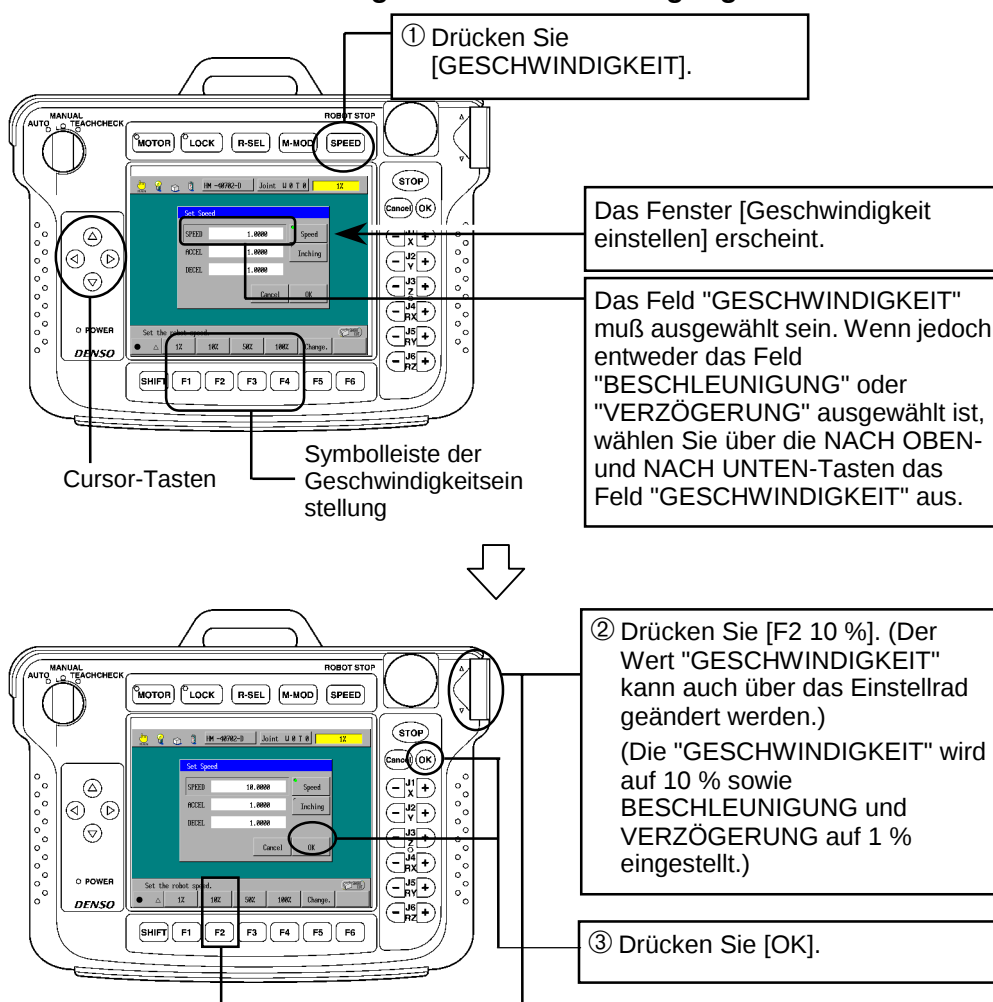
### SCHRITT 2 EINschalten des Roboter-Controller



### SCHRITT 3 Auswählen des manuellen Modus des Roboters

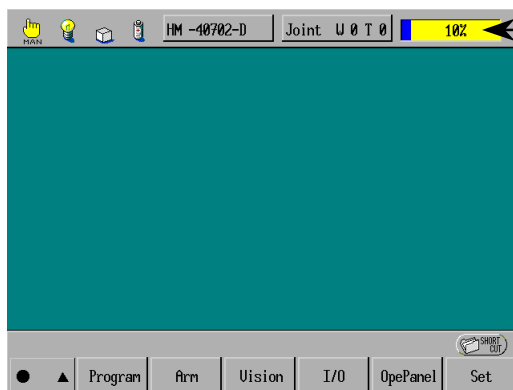


### SCHRITT 4 Einstellen von Geschwindigkeit und Beschleunigung



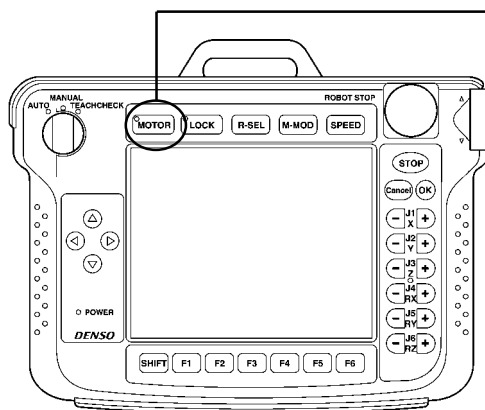
#### ★Bemerkungen★

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



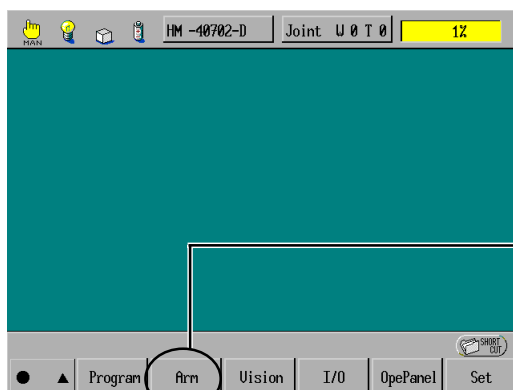
① Drücken Sie [MOTOR].  
Die Motorspannung wird eingeschaltet und die Lampe [MOTOR] leuchtet auf.

## SCHRITT 6 Manuelles Bewegen jedes Roboterarms



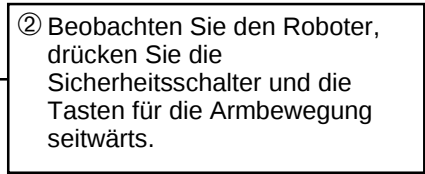
**Vorsicht**

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



① Drücken Sie [F2 Arm].



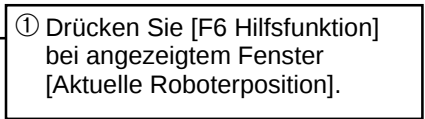


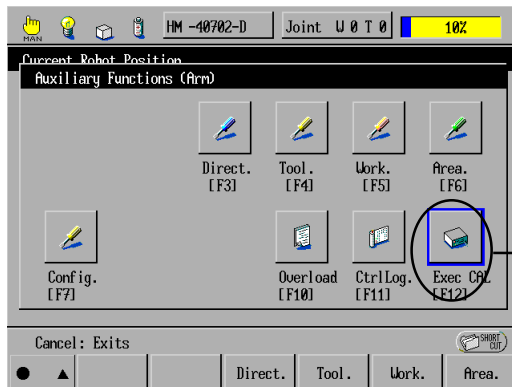
**Vorsicht:** Bis zur Durchführung der Kalibrierung (CAL) kann der Roboter nur im Gelenk-Modus gesteuert werden. Darüberhinaus sind die für die Koordinaten jeder Achse angezeigten Werte nicht korrekt.

## SCHRITT 7 Ausführen der Kalibrierung (CAL)

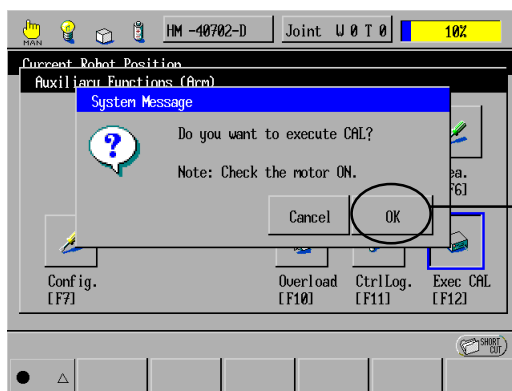
Die CAL-Prozedur wird im folgenden näher beschrieben.

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.

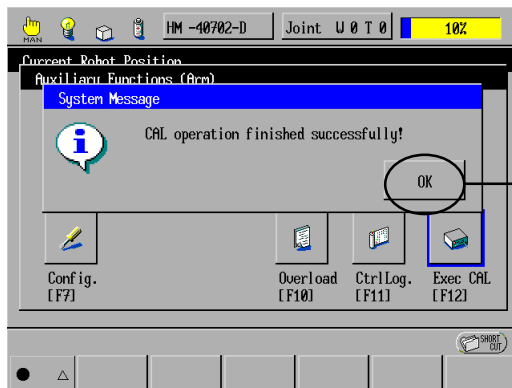




② Drücken Sie [F12 CAL ausf.].

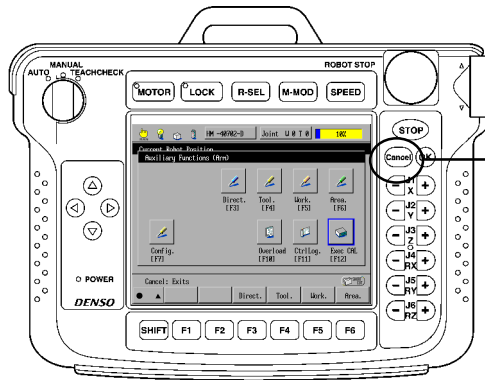


③ Prüfen Sie die Systemmeldung, und drücken Sie [OK].



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].

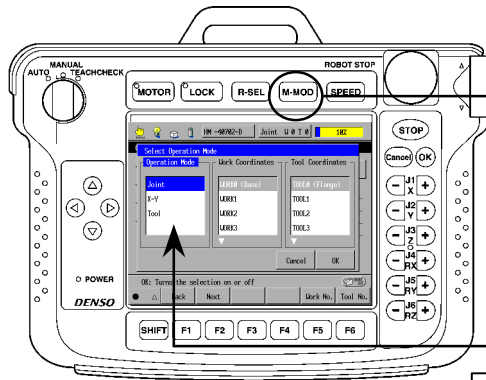


Das Fenster [Aktuelle Roboterposition] wird erneut angezeigt. Nun ist die Kalibrierung (CAL) abgeschlossen und der Roboter kann gesteuert werden.



## SCHRITT 8

### Auswählen des manuellen Modus und manuelles Steuern des Roboters



① Drücken Sie [M-MOD].

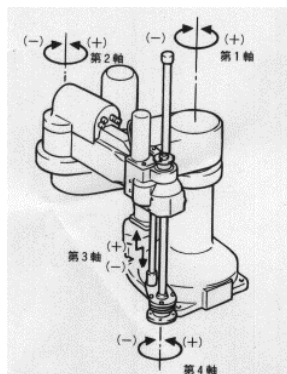
Das Fenster [Betriebsmodus] erscheint.

#### ★Punkt★

Folgende 3 Modi können für manuelle Roboter-Operationen ausgewählt werden: Gelenk-, X-Y- und Werkzeug-Modus.

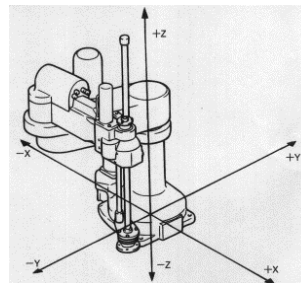
#### <Gelenk-Modus>

Ermöglicht es, jedes der 4 Gelenke unabhängig voneinander zu bewegen.



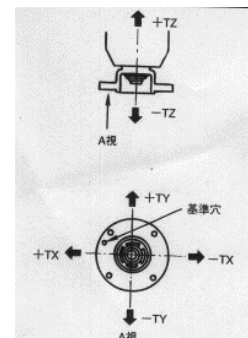
#### <X-Y-Modus>

Ermöglicht es, den Roboter innerhalb der Basiskoordinatenachsen (deren Ursprung im Zentrum des Robotersockels definiert ist) linear zu bewegen. Die 4. Achse behält die gleiche Ausrichtung bei wie unmittelbar vor dem Bewegen des Roboters.

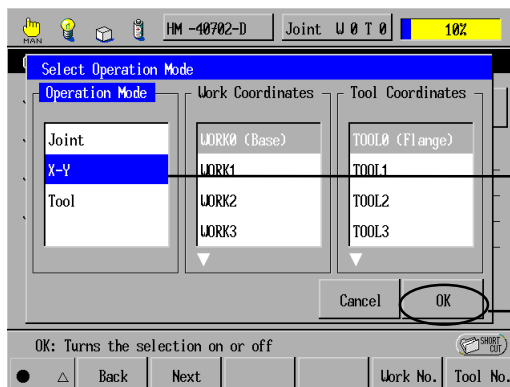


#### <Werkzeug-Modus>

Ermöglicht es, den Roboterarm linear entlang der Arbeitskoordinaten der 4. Achse zu bewegen. Die 4. Achse behält ihre Ausrichtung bei.



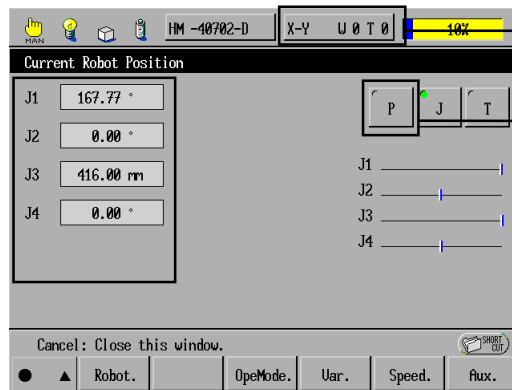
In dieser Lektion üben Sie, den Roboter im X-Y-Modus zu steuern.



② Wählen Sie im Dialogfenster [Betriebsmodus auswählen] "X-Y" (über die NACH OBEN- und NACH UNTEN-Tasten oder das Einstellrad).

③ Drücken Sie [OK].

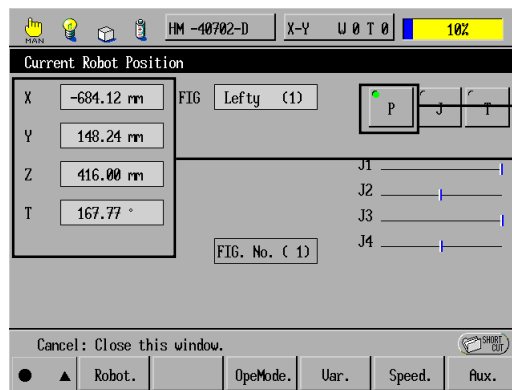
Das Hauptanzeige erscheint, wenn Sie [F2 Arm] drücken.



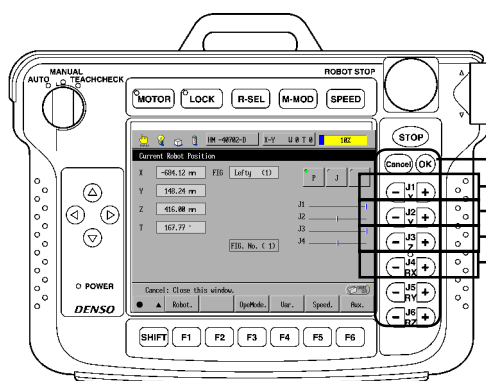
X-Y wird auf der Statusleiste angezeigt.

④ Drücken Sie die Schaltfläche "P" (Positionsvariable), um die aktuelle Position 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 steuern.)



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



⑤ Steuern Sie den Roboter, indem Sie bei gedrücktem Sicherheitsschalter die Tasten für die Arm-bewegung seitwärts drücken.

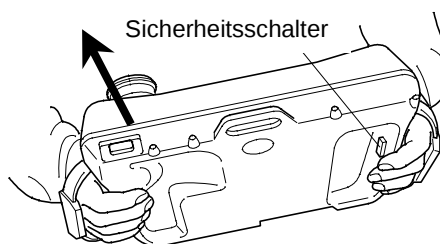
Tasten für die Bewegung des

① Bewegung in X-Richtung

② Bewegung in Y-Richtung

③ Bewegung in Z-Richtung

④ Drehung um die T-Achse



③ Bewegung entlang der Z-Achse.

④ Drehung um die T-Achse

① Bewegung entlang der X-Achse

② Bewegung entlang der Y-Achse

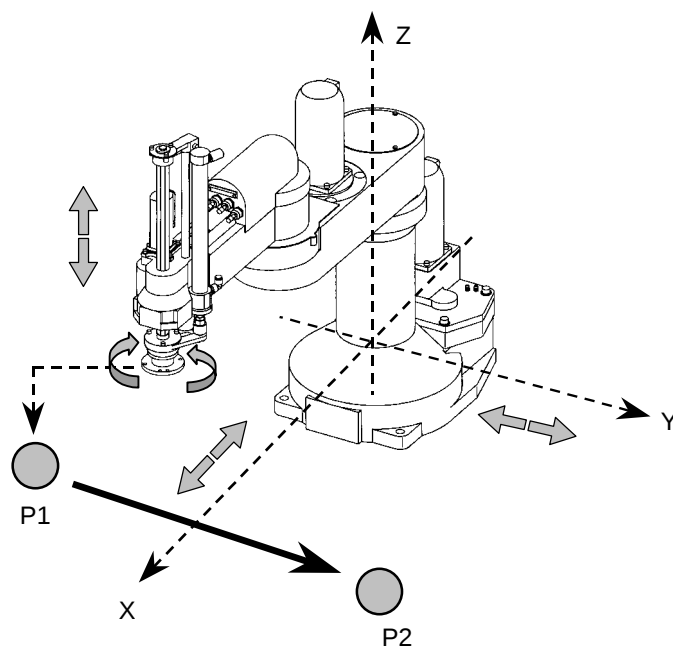
## Lektion 2 Steuern des Roboters über ein einfaches Programm

---

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 im Auto-Modus



## 2.1 Erstellen eines einfachen Programmes mit dem Programmiergerät

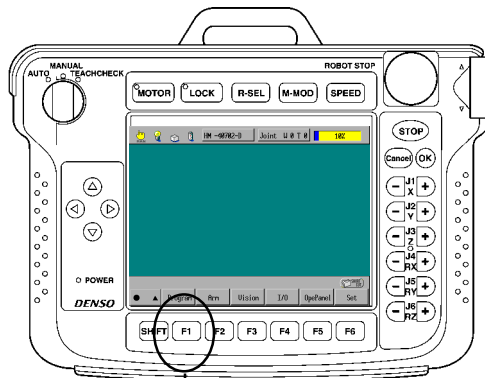
Zuerst geben Sie Codes eines einfachen Programmes über das Programmiergerät ein.

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 manuellen Modus auswählen, bevor Sie fortfahren.

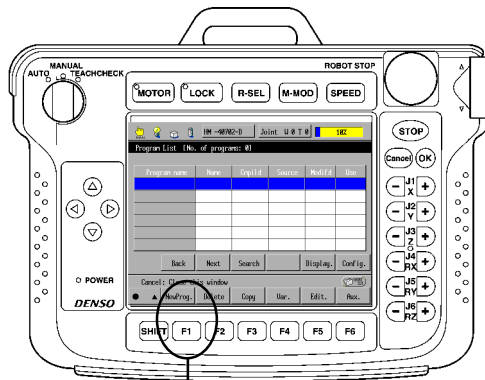
Eine ausführliche Beschreibung zum Schreiben von Programmen, siehe PROGRAMMIERUNGSHANDBUCH.

### SCHRITT 1 Öffnen eines Programmbearbeitungsfensters

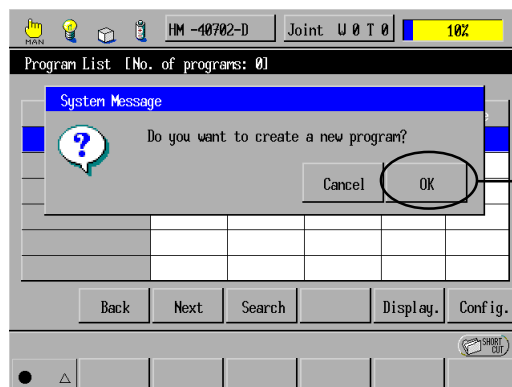
Zum Erstellen eines Programmes müssen Sie auf der Anzeige des Programmiergeräts das Fenster zum Bearbeiten von Programmen öffnen.



① Drücken Sie [F1 Programm] in der Hauptanzeige.

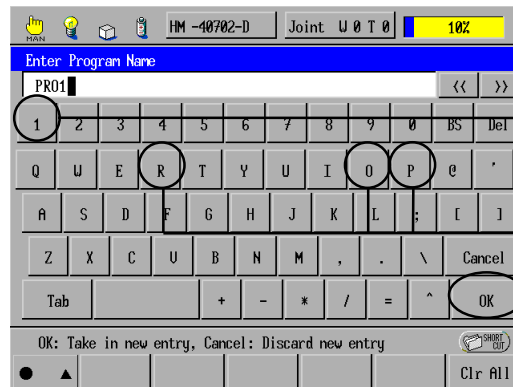


② Drücken Sie [F1 NeuesProg].



③ Drücken Sie [OK].

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

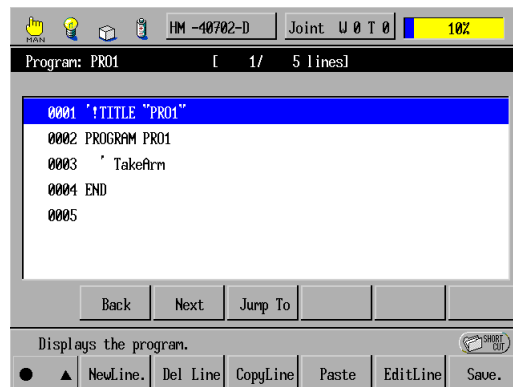


④ Geben Sie PRO1 durch Betätigen der Buchstaben- und numerischen Tasten ein.

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



Damit ist die Vorbereitung für die Programmbearbeitung abgeschlossen.



Die voreingestellten Programmcodes werden angezeigt.

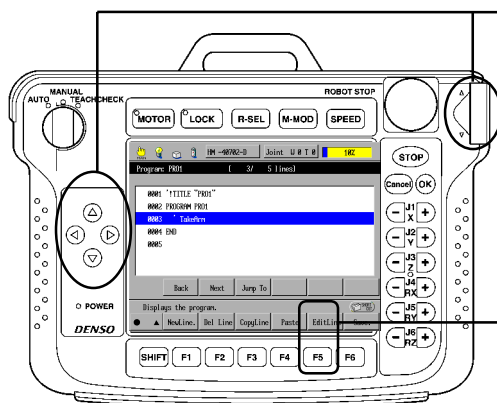
## SCHRITT 2 Eingeben von Programmcodes

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"

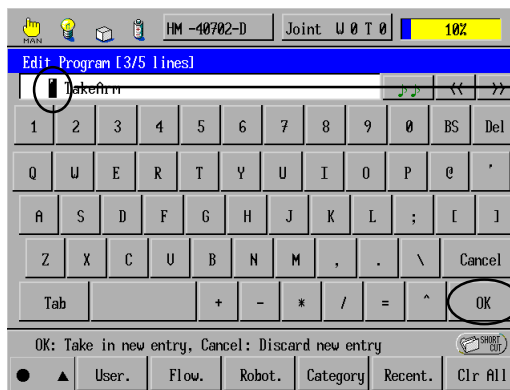
#### PROGRAM PRO1

|            |                                                    |
|------------|----------------------------------------------------|
| 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        |                                                    |



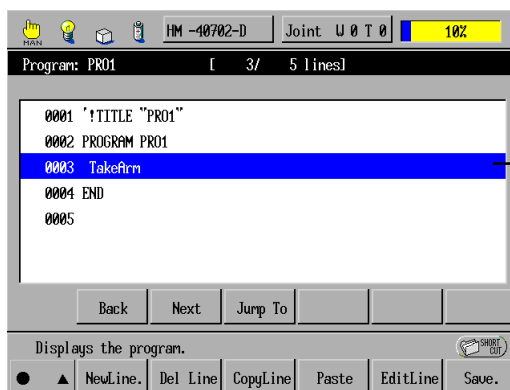
① Bewegen Sie im Fenster "Programm: PRO1" den Cursor über die Cursor-Tasten oder das Einstellrad zur 3. Zeile.

② Drücken Sie [F5 Zeile Bearbeiten].



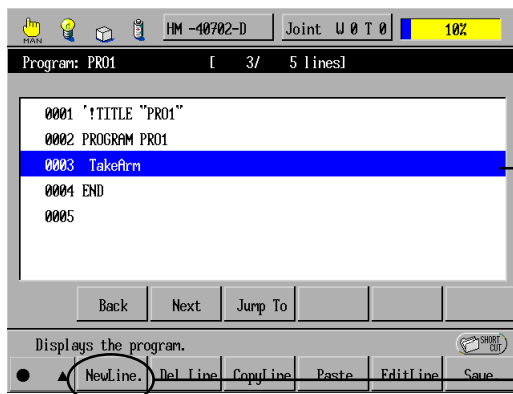
③ Löschen Sie den Apostroph (') oberhalb der Zeile mit den Cursor- und der [Del-Taste].

④ Drücken Sie [OK].

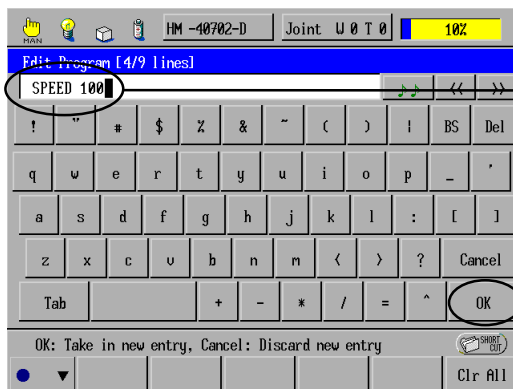


Auf dem Bildschirm erscheint wieder das Programmbearbeitungsfenster [Programm: PRO1], wobei die 3. Zeile geändert wurde.



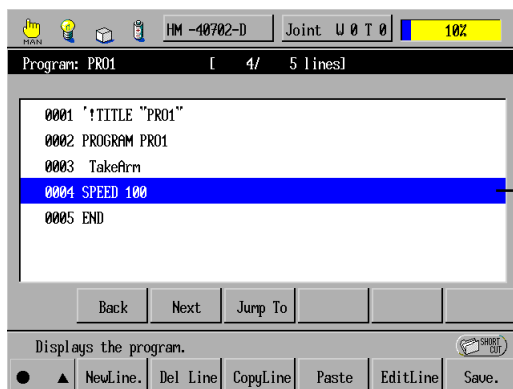


⑤ Bewegen Sie den Cursor in die dritte Zeile, und drücken Sie [F1 NeueZeil].



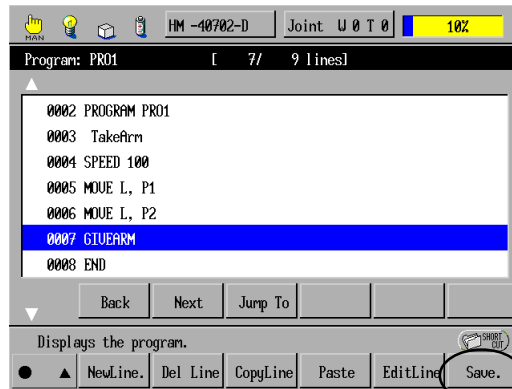
⑥ Geben Sie "GESCHWINDIGKEIT 100" über die Tastatur ein. Dies wird in diesem Fenster angezeigt.

⑦ Drücken Sie [OK].



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





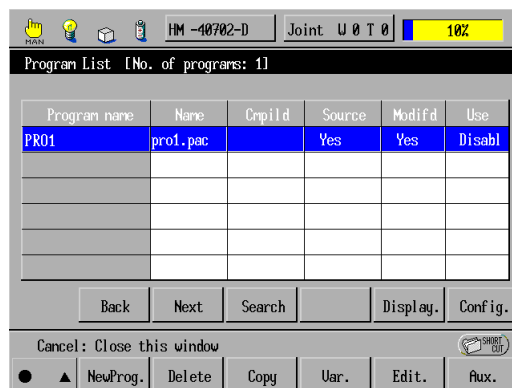
⑧ Geben Sie alle auf Seite 16 angegebenen Programmcodes genauso ein wie die "GESCHWINDIGKEIT 100".

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".



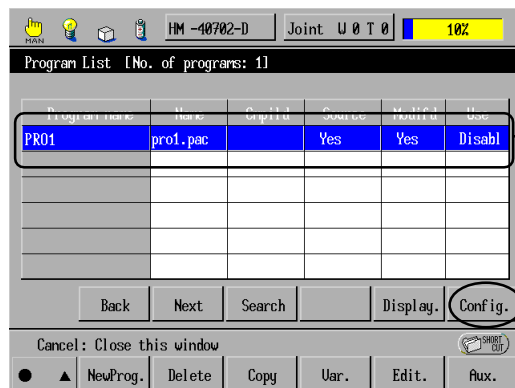
★Vorsicht★

- 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 Zum Erstellen eines neuen Programmes kehren Sie zu Schritt 1 zurück.

### SCHRITT 3 Kompilieren des Programmes in ein Laufzeitformat

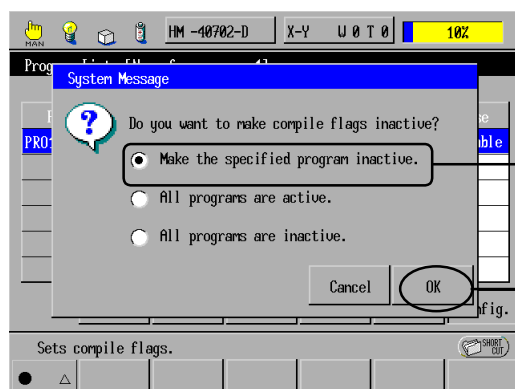
Nach dem Bearbeiten eines Programmes müssen Sie dieses kompilieren, also in ein Laufzeitformat umwandeln, das vom Roboter-Controller ausgeführt werden kann.

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.



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.].



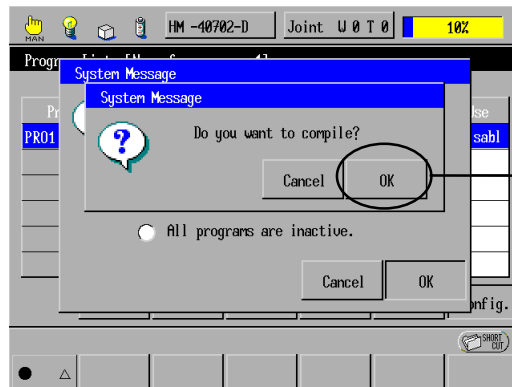
③ Wählen Sie "Angegebenes Programm inaktiv machen".

④ Drücken Sie [OK].

★**Vorsicht**★

Die gleiche Operation kann über die Option [Konfig.] ausgeführt werden, die sich unten rechts auf dem Fenster "Programmliste" befindet.





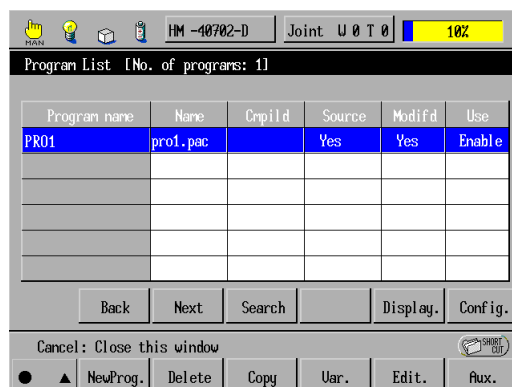
⑤ Drücken Sie [OK].  
Das Kompilieren beginnt.



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

★Vorsicht★

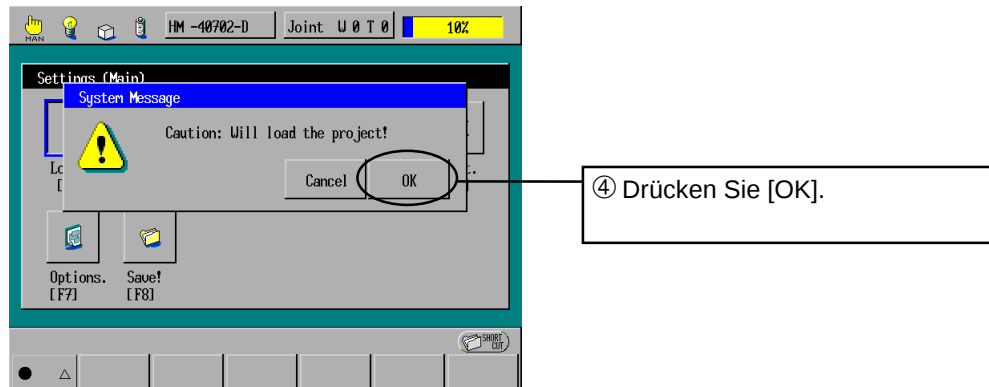
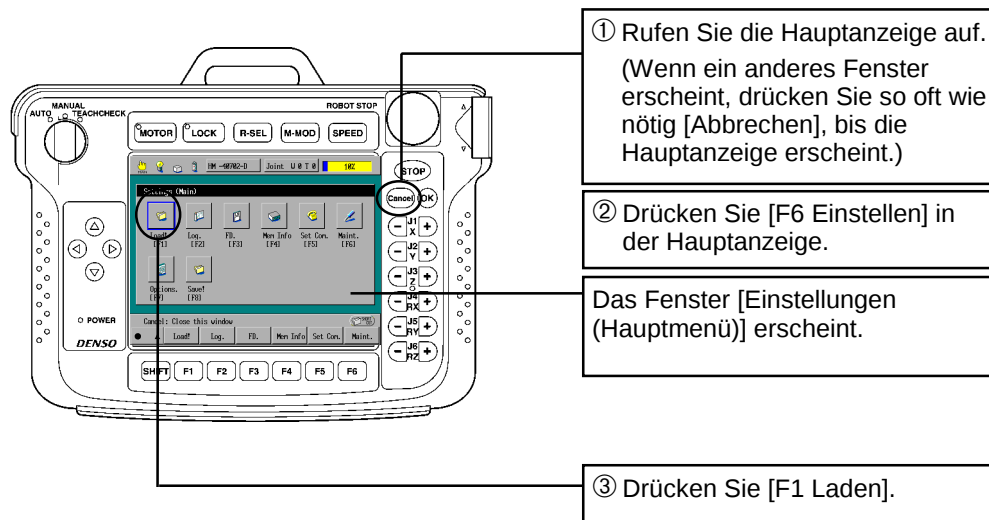
- ① Wenn Sie an dieser Stelle statt [OK] auf [Abbrechen] drücken, erscheint wieder das Fenster [Programmliste], ohne daß die Kompilierung ausgeführt wurde.
- ② Es gibt noch eine andere Möglichkeit, Programme in ein Laufzeitformat zu kompilieren.  
Drücken Sie [F6 Hilfs.] im Fenster [Programmliste], um 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.

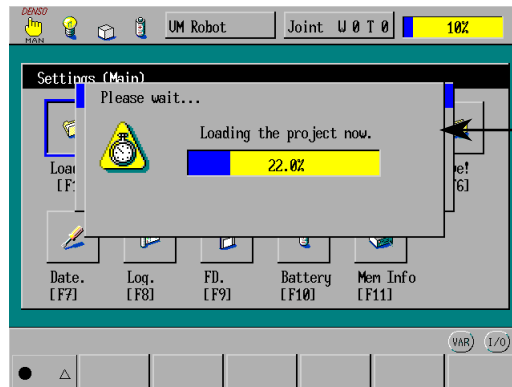


## SCHRITT 4 Laden des Programmes

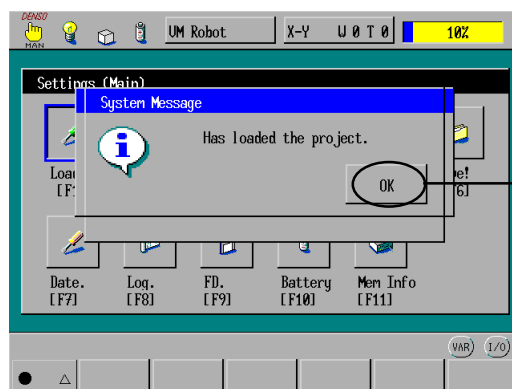
Sie müssen das kompilierte Programm laden, damit der Roboter es ausführen kann.

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





Die Meldung "Bitte warten ... Projekt wird gerade geladen." erscheint.

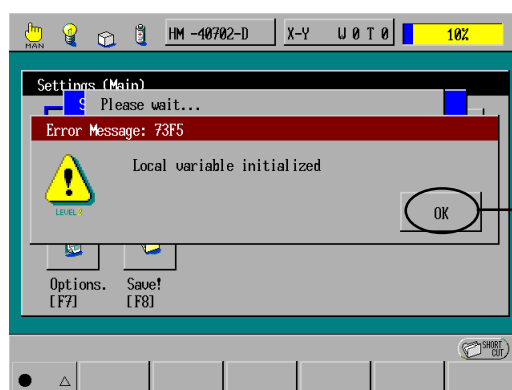


⑤ Sobald die Systemmeldung "Projekt geladen." angezeigt wird, drücken Sie [OK].

### ★Vorsicht★

Wenn Sie ein Projekt mit lokalen Variablen laden, die sich von denen im vorigen Projekt unterscheiden, erscheint die Fehlermeldung: "Lokale Variable initialisiert".

Drücken Sie [OK], um fortzufahren.



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

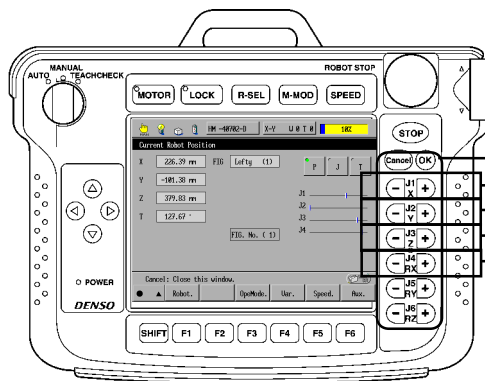
Unter "Programmierung" verstehen wir in diesem Zusammenhang eine Programmierung, bei der Sie über das Programmiergerät den Roboter anweisen, welche Bewegungen ausgeführt werden sollen. Über die Programmierung können dem Roboter Bewegungen beigebracht werden.

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.

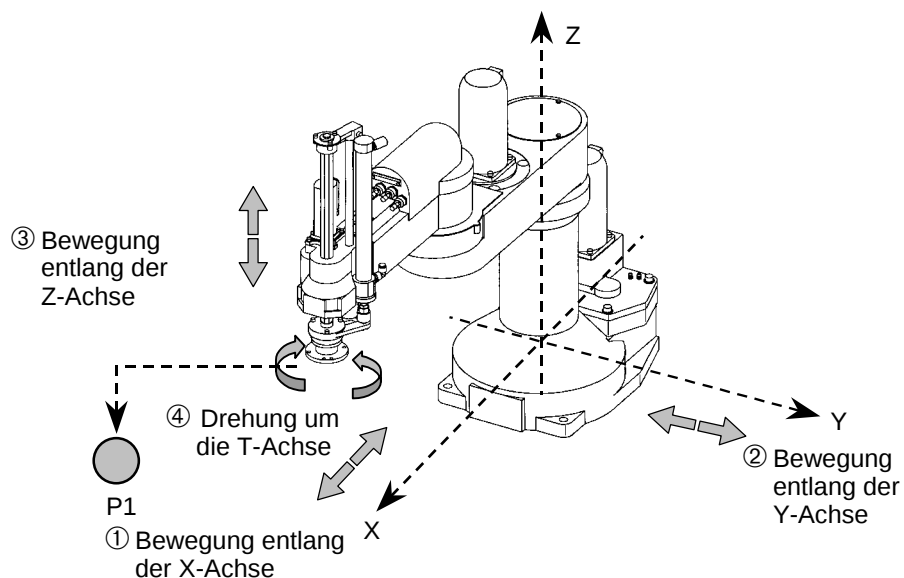
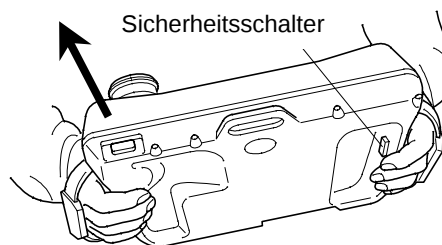
Das in Lektion 2.1 erstellte Programm enthält die zwei Positionsvariablen P1 und P2. In diesem Abschnitt erfahren Sie, wie Sie die Roboterwerte für P1 und P2 programmieren.

### SCHRITT 1 Programmieren der Roboterposition P1

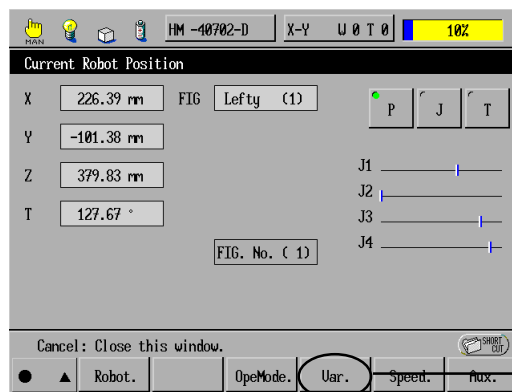
① Halten Sie den Sicherheitsschalter gedrückt, und drücken Sie die entsprechenden Tasten für die Armbewegung seitwärts, um den Roboterarm an die gewünschte Position zu bringen, die Sie P1



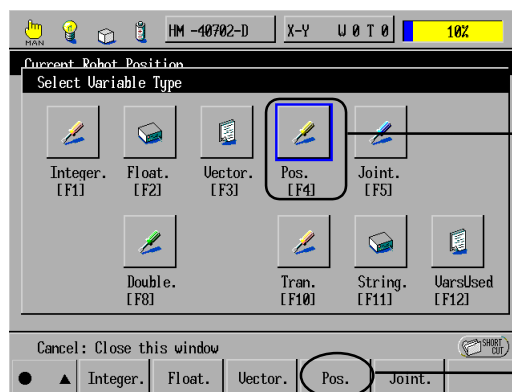
Tasten für die Bewegung des  
① Bewegung in X-Richtung  
② Bewegung in Y-Richtung  
③ Bewegung in Z-Richtung  
④ Drehung um die T-Achse



## SCHRITT 2 Zuordnen des programmierten Wertes zur "Variable P1"



① Drücken Sie [F4 Var.].

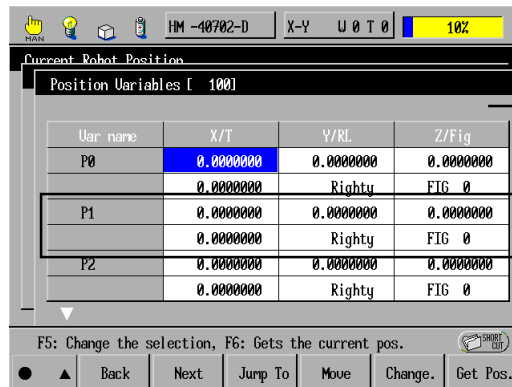


② Wählen Sie im Fenster "Variablentyp auswählen" den entsprechenden Variablentyp aus.  
Drücken Sie an dieser Stelle [F4 Pos.], um einer Positionsvariablen einen Wert zuzuordnen.  
(Es besteht auch die Möglichkeit, [Pos. 1] auf dem Bildschirm zu

### ★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<br>(Wertebereich: -2147483648 bis +2147483647)                              |
| F. (Gleiten):      | Gleitpunktvariable<br>(Wertebereich: -3,402823E+383,402823E+38)                                  |
| D. (Doppelt):      | Doppelpräzisionsvariable<br>(Wertebereich: -1,7976931348623157D+308 bis 1,7976931348623157D+308) |
| V. (Vektor):       | Vektorvariable (X, Y, Z)                                                                         |
| P. (Pos):          | Positionsvariable (X, Y, Z, T, ARMSTELL)                                                         |
| J. (Gelenk):       | Gelenkvariable (J1, J2, J3, J4)                                                                  |
| T. (Trans):        | Homogene Transformationsvariable                                                                 |
| S. (Zeichenfolge): | Zeichenfolgenvariable (die aus einer Zeichenfolge von bis zu 247 Zeichen bestehen kann)          |

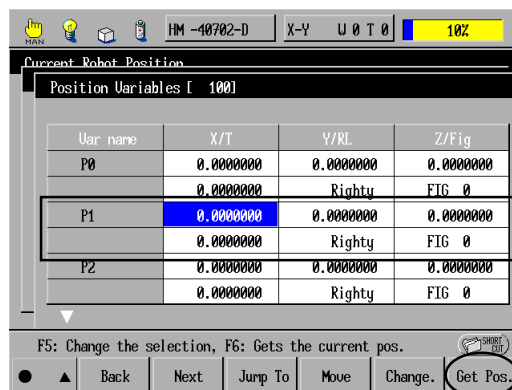


Das Fenster [Positionsvariablen] erscheint.

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

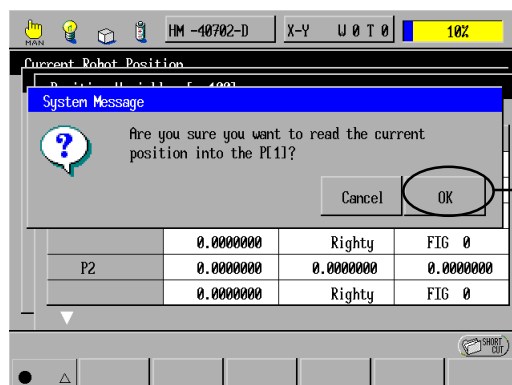
Im Fenster [Positionsvariablen] werden 5 Datentypen für jeden Variablennamen angezeigt.

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



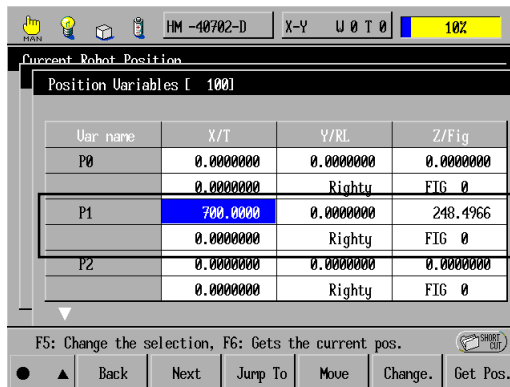
④ Prüfen Sie, ob der [Variablenname P1] ausgewählt ist.

⑤ Drücken Sie [F6 Pos. abbr.].



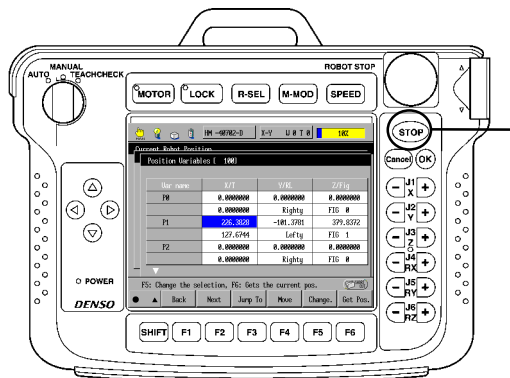
⑥ Prüfen Sie die Systemmeldung und wenn alles korrekt ist, drücken Sie [OK].



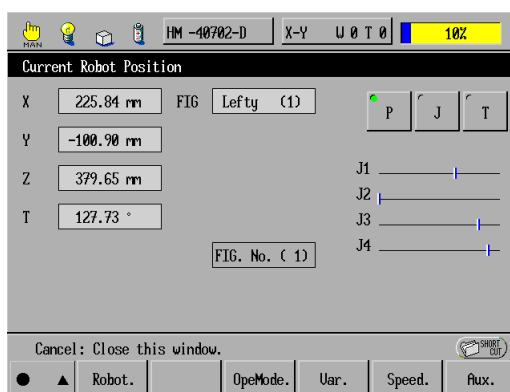


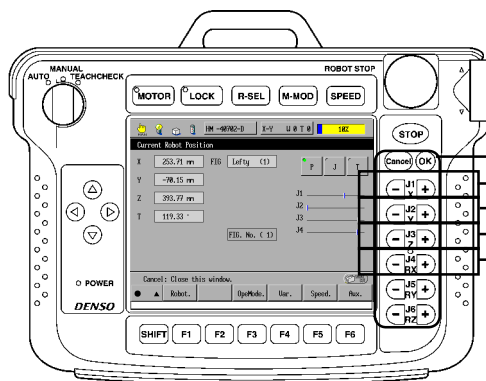
Die aktuelle Position wird in Variable P1 eingelesen.

### SCHRITT 3 Programmieren der Roboterposition P2 und Zuweisung von [Var Name P2]



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

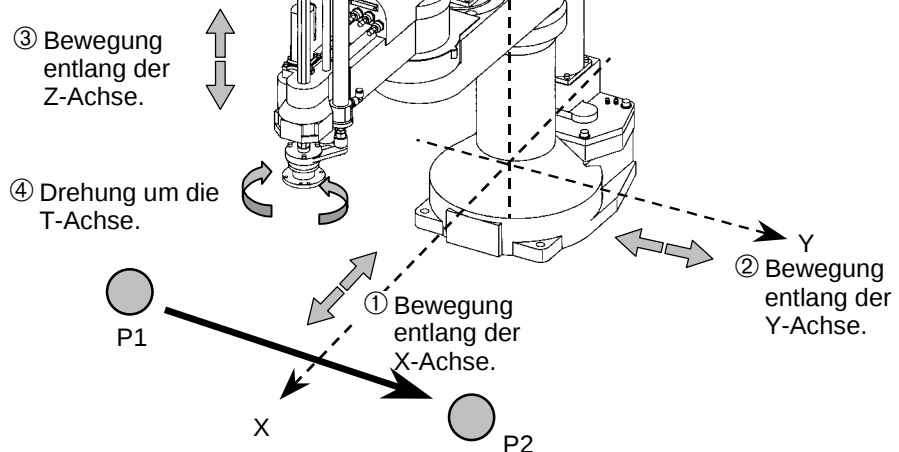
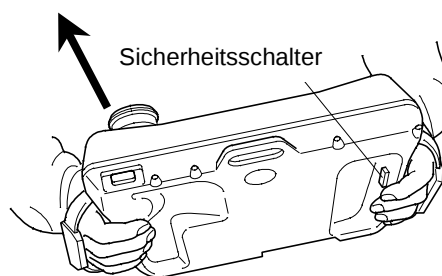




② Während Sie den Sicherheitsschalter halten, drücken Sie die Tasten zur Armbewegung seitwärts, um den Roboterarm zu der P2 zugewiesenen Position zu bewegen.

Tasten für die Bewegung des

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



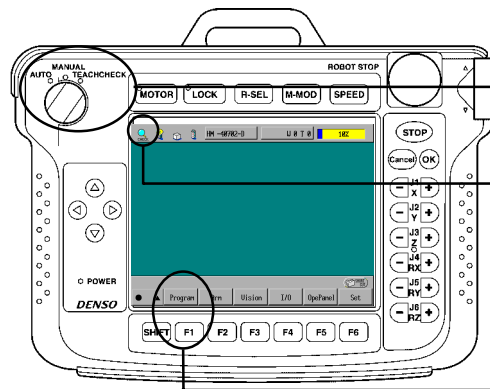
③ Weisen Sie den für P2 [Variablenname P2] programmierten Wert genauso zu wie in Schritt 2, "Zuweisen des programmierten Wertes zur [Variable P1]" beschrieben.

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 Auswählen des Programmierprüfmodus des Roboters



① Stellen Sie den Modusauswahlschalter auf die Position PROGRAMMIERPRÜFUNG ein.

Ganz links auf der Statusleiste erscheint ein Symbol, das den PROGRAMMIERPRÜFMODEUS anzeigt.

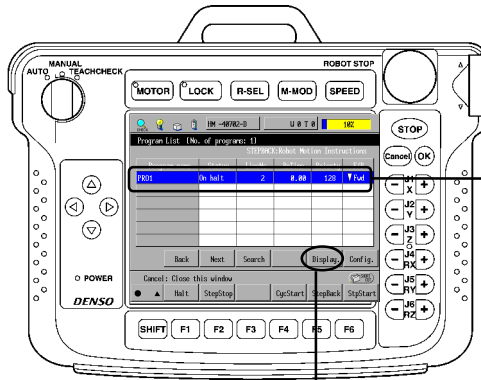
② Drücken Sie in der Hauptanzeige [F1 Programm].



Das Fenster "Programmliste" erscheint.

| <div> </div> <div> <div>HM -40702-D</div> <div>U 0 T 0</div> <div>10%</div> </div>                  |         |          |          |          |          |
|-----------------------------------------------------------------------------------------------------|---------|----------|----------|----------|----------|
| Program List [No. of programs: 1]                                                                   |         |          |          |          |          |
| STEPBACK: Robot Motion Instructions                                                                 |         |          |          |          |          |
| Program name                                                                                        | Status  | LineNo   | RnTime   | Priority | F/B      |
| PR01                                                                                                | On halt | 2        | 0.00     | 128      | ▼ Fud    |
|                                                                                                     |         |          |          |          |          |
|                                                                                                     |         |          |          |          |          |
|                                                                                                     |         |          |          |          |          |
| <div> <div>Back</div> <div>Next</div> <div>Search</div> <div>Display</div> <div>Config</div> </div> |         |          |          |          |          |
| Cancel: Close this window                                                                           |         |          |          |          |          |
| ● ▲                                                                                                 | Halt    | StepStop | CycStart | StepBack | StpStart |

## SCHRITT 2 Schrittprüfung

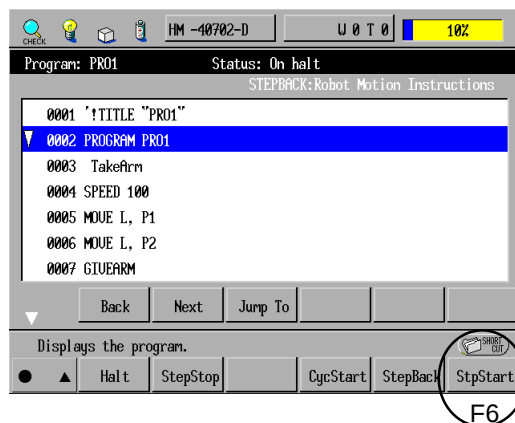


① Wählen Sie im Fenster "Programmliste" "PRO1".  
(Die Auswahl kann über die Cursor-Tasten oder das Einstellrad oder durch direktes Berühren des Bildschirms erfolgen.)

② Drücken Sie [F11 Anzeige], um die PRO1-Programmcodes anzuzeigen.



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



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



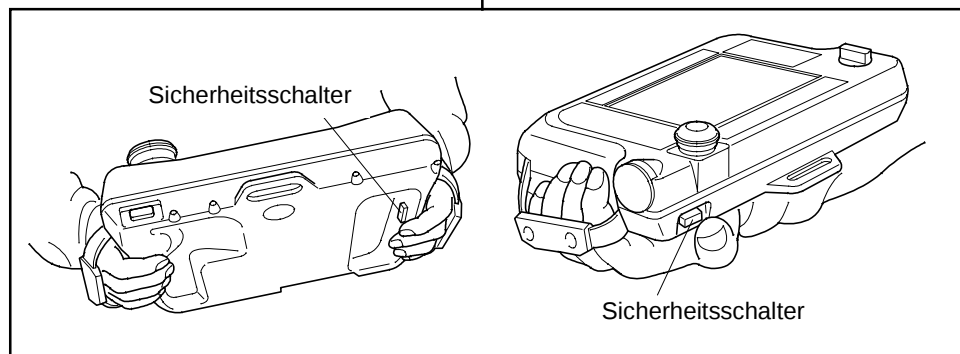
Diese Systemmeldung erscheint.

### ★Vorsicht★

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



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



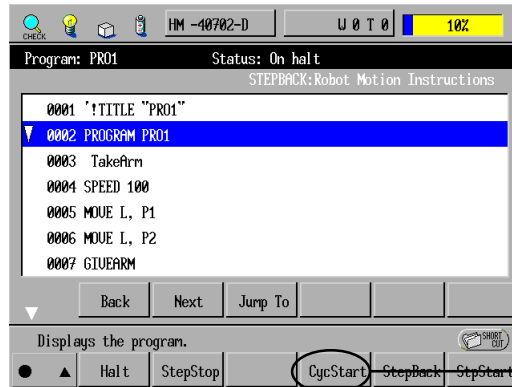
Halten Sie im Programmierprüfmodus Sicherheitsschalter und OK-Taste gedrückt, bis die Ausführung abgeschlossen ist. Wenn eine von beiden losgelassen wird, wird der Roboter sofort angehalten.



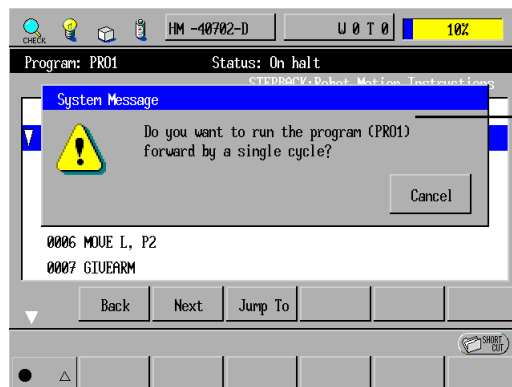
Führen Sie die oben beschriebene Prozedur wiederholt durch, um alle Codes in PRO1 auszuführen und zu prüfen, ob jede Bewegung sicher ist.

## 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.



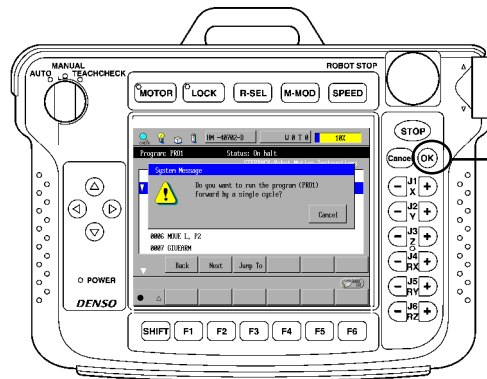
① Drücken Sie [F4 ZykStart].



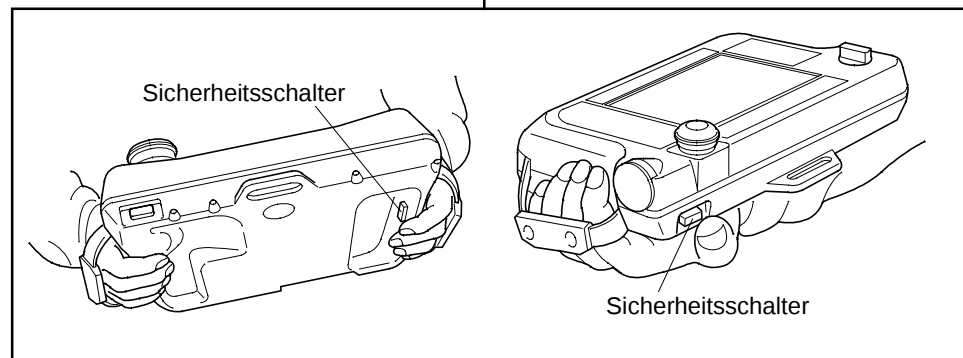
Diese Systemmeldung erscheint.

★**Vorsicht**★

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



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



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

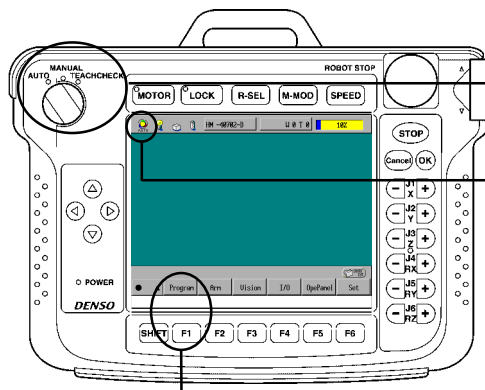
Wenn das Programm bis zum Ende ausgeführt worden ist, erfolgt ein Stop.

## 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.

**Vorsicht:** 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 Auswählen des Auto-Modus des Roboters



① Stellen Sie den Modusauswahl-schalter auf die Position AUTO ein.

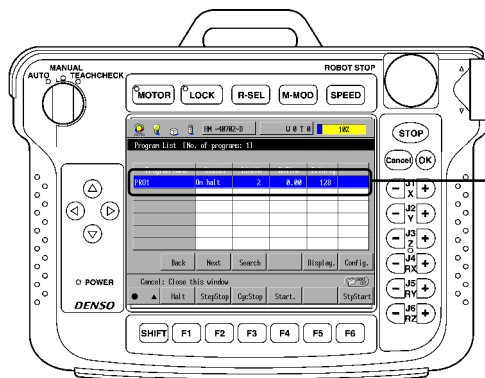
Ganz links auf der Statusleiste erscheint ein Symbol, das den Auto-Modus anzeigt.

② Drücken Sie [F1 Programm].



### SCHRITT 2 Auswählen des auszuführenden Programmes

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



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

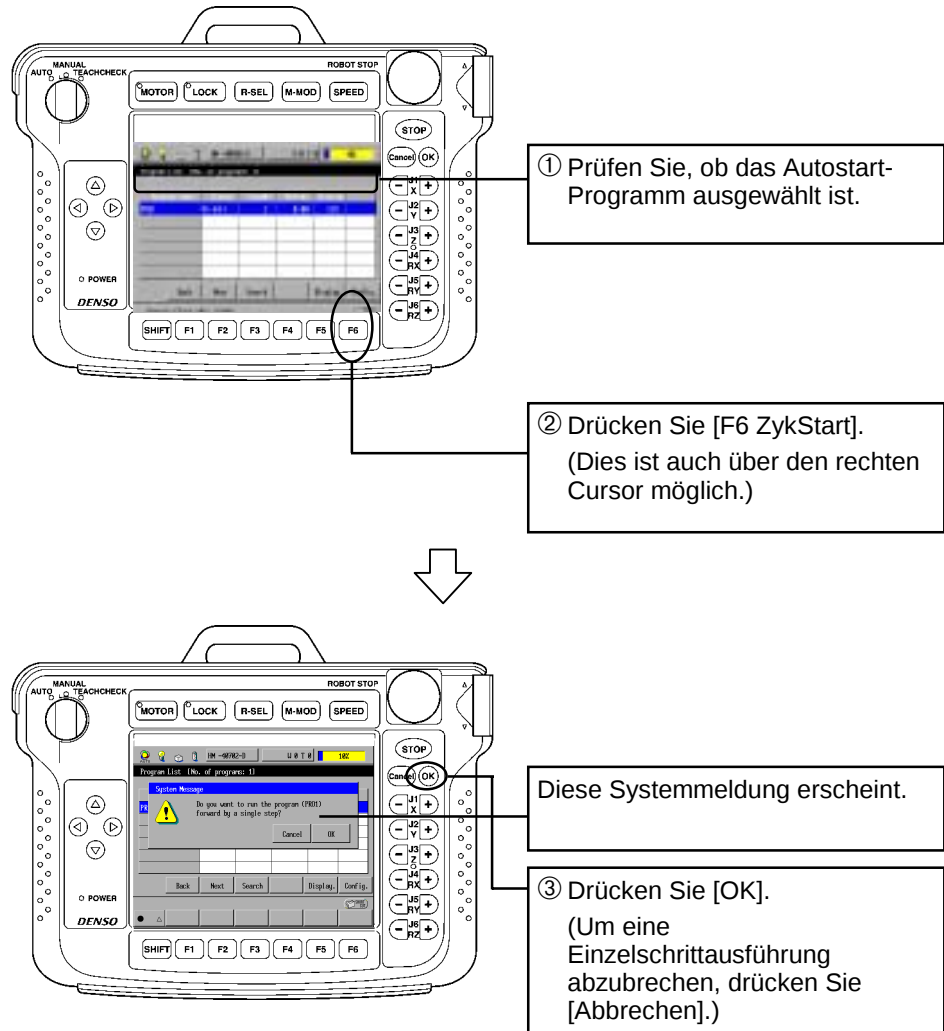


### SCHRITT 3

### Einzelschrittausführung

#### ★Anmerkung★

Wenn Sie das Programm bei einer Einzelschrittausführung anzeigen lassen möchten, drücken Sie vorher "F11 Anzeige".



#### ★Vorsicht★

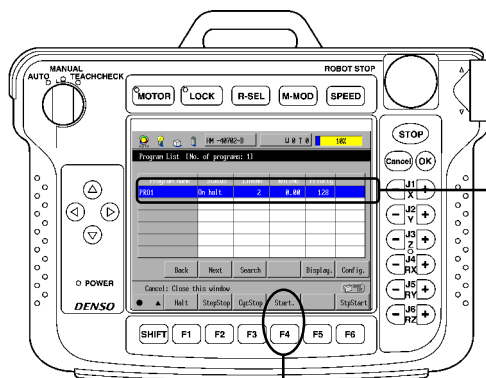
Halten Sie während der Programmausführung stets eine Hand frei und bereit, die STOP-Taste zu drücken.

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

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

## SCHRITT 4 Einzelzyklusausführung

Nach Ausführung einer Einzelschrittausführung starten Sie eine Einzelzyklusausführung.

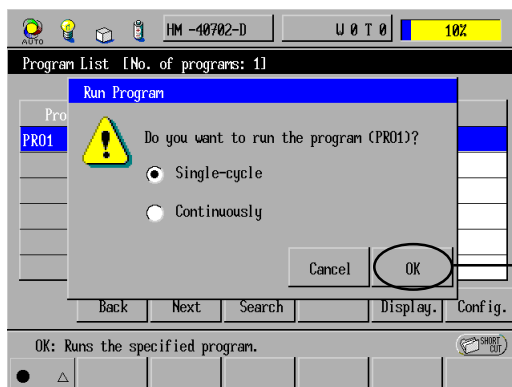


① Prüfen Sie, ob das Start-Programm ausgewählt ist.

② Drücken Sie [F4 Start.].

### ★Vorsicht★

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



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

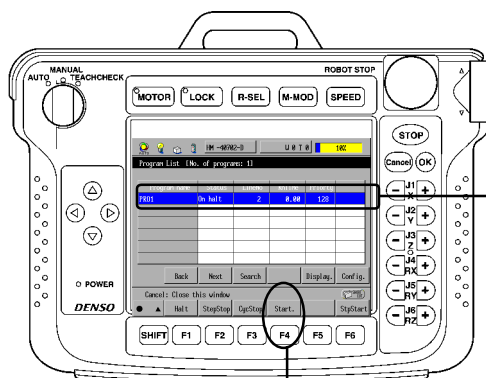
Wenn das Programm bis zum Ende ausgeführt worden ist, erfolgt ein Stop.

### ★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.

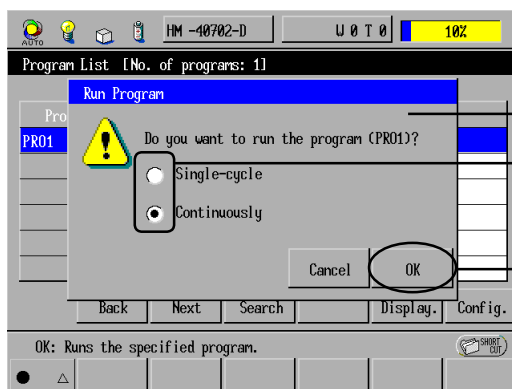
## SCHRITT 5 Kontinuierliche Ausführung

Starten Sie eine kontinuierliche Ausführung des Programmes.



① Prüfen Sie, ob das Start-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].  
Das Programm "PRO1" wird kontinuierlich ausgeführt.  
(Sie können die kontinuierliche Ausführung durch Anhalten (Stop) oder Schrittstop stoppen.)

### ★Vorsicht★

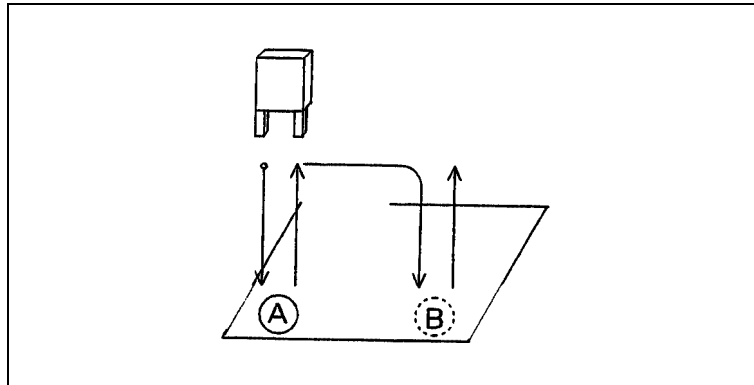
Halten Sie während der Programmausführung stets eine Hand frei und bereit, die STOP-Taste zu drücken.



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

## Kapitel 1 Praktische Übung

**Übung:** Erstellen Sie ein Programm, um ein Werkstück von Punkt A zu Punkt B zu bewegen und dann eine Betriebsprüfung durchzuführen.



Folgendes gilt:

- E/A-Zuordnung      Nr. 64 (Systemausgabe) ... geschlossene Hand  
                             Nr. 65 (Systemausgabe) ... offene Hand
- Die Geschwindigkeit beim Aufnehmen und Ablegen eines Werkstückes: 30%  
  Geschwindigkeit bei anderen Bewegungen: 80%
- Jede der Annäherungs- und Abweichungsdistanzen: 50 mm
- Interpolationssteuerung: PTP (Punkt-zu-Punkt)

**Lösung:**

```
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 des PC-Programmiersystems WINCAPSII auf einem PC, und Erstellen bzw. Kompilieren eines Programmes. Sie werden das kompilierte Programm dann zum Roboter-Controller hochladen.

Außerdem werden Sie in Kapitel 2 auch die Maschinenverriegelung des Roboter-Controller 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 | Einstellen des Roboter-Controller mit dem Programmiergerät      | 40 |
| 3.1         | Ausführen der Kalibrierung (CAL)                                | 40 |
| 3.2         | Aktivieren der Maschinenverriegelung des Roboter-Controller     | 43 |
| 3.3         | Einstellen des Kommunikationsanschlusses des Roboter-Controller | 44 |
| ► Lektion 4 | Starten von WINCAPSII und Erstellen eines Systemprojektes       | 48 |
| 4.1         | Starten des System-Manager                                      | 48 |
| 4.2         | Registrieren eines neuen Systemprojektes                        | 50 |
| 4.3         | Einstellen des Kommunikationsanschlusses des PC                 | 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 von Programmcodes                                      | 57 |
| 6.4         | Einsetzen des Befehlseditors                                    | 58 |
| 6.5         | Speichern des Programmes                                        | 60 |
| ► Lektion 7 | Kompilieren des Programmes in ein Laufzeitformat                | 61 |
| ► Lektion 8 | Hochladen des Programmes (PC → Roboter-Controller)              | 62 |

---

## Lektion 3      Einstellen des Roboter-Controller mit dem Programmiergerät

### 3.1      Ausführen der Kalibrierung (CAL)

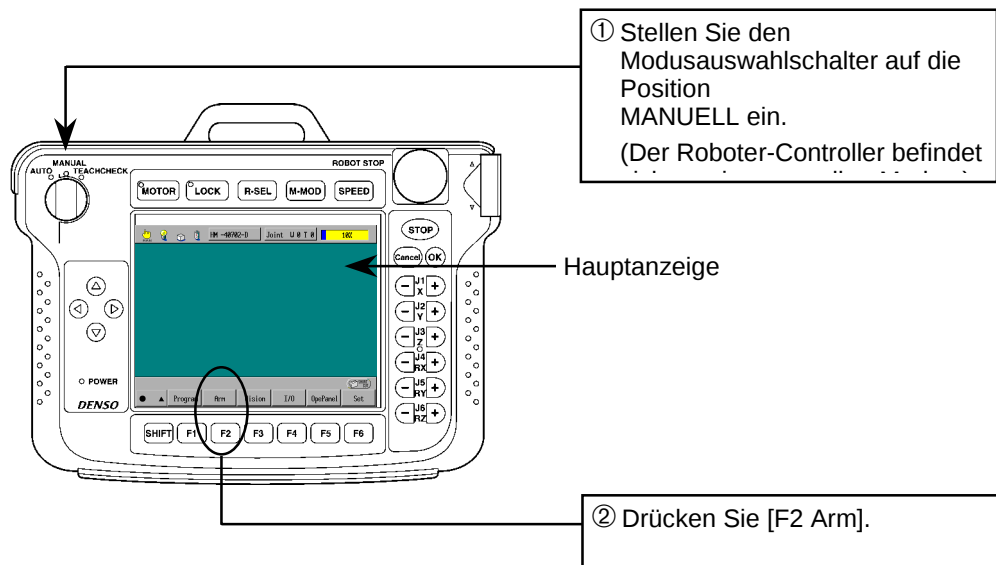
Die Kalibrierung (CAL), bei der alle Roboterachsen betätigt werden, ist erforderlich, um die aktuelle Position des Roboters zu bestätigen.



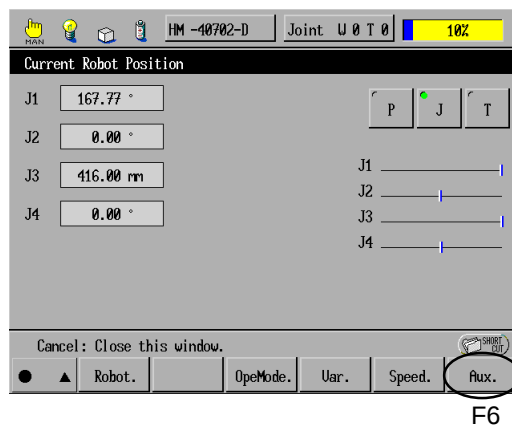
#### Warnung

Durch die 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.

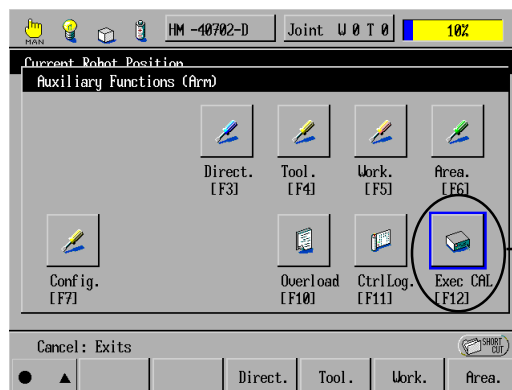
#### SCHRITT 1      Auswählen des manuellen Modus des Roboters



## SCHRITT 2 Starten der CAL-Operation

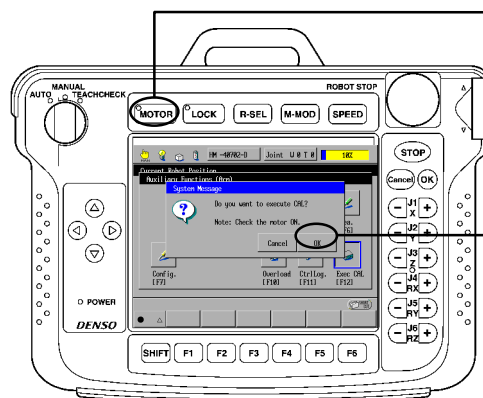


① Drücken Sie [F6 Hilfsfunktionen].



② Drücken Sie [F12 CAL ausf].

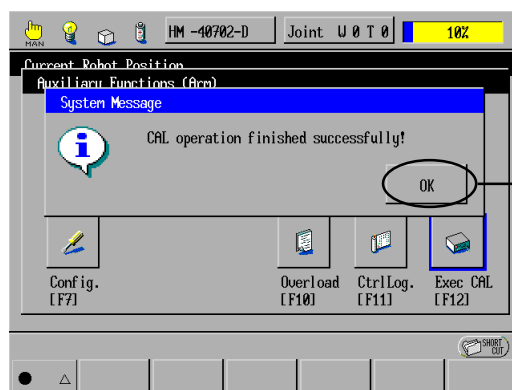
## SCHRITT 3 Einschalten des Motors



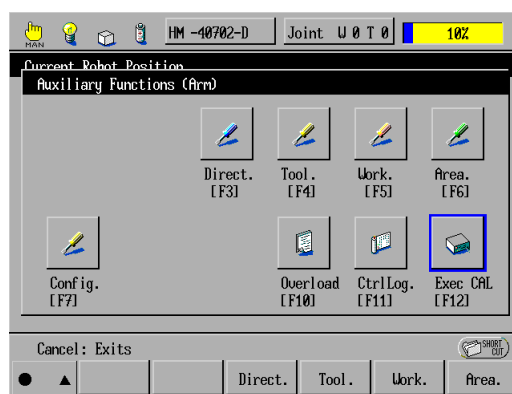
① Drücken Sie [MOTOR].  
Der Motor wird eingeschaltet und die Lampe [MOTOR] leuchtet.

② Drücken Sie [OK].

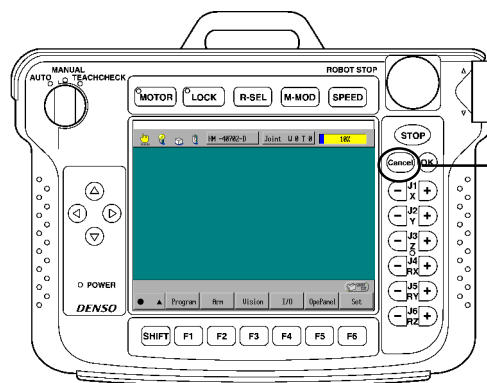
## SCHRITT 4 Beenden der CAL-Operation



① Drücken Sie [OK].



Hiermit ist die Kalibrierung abgeschlossen.



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

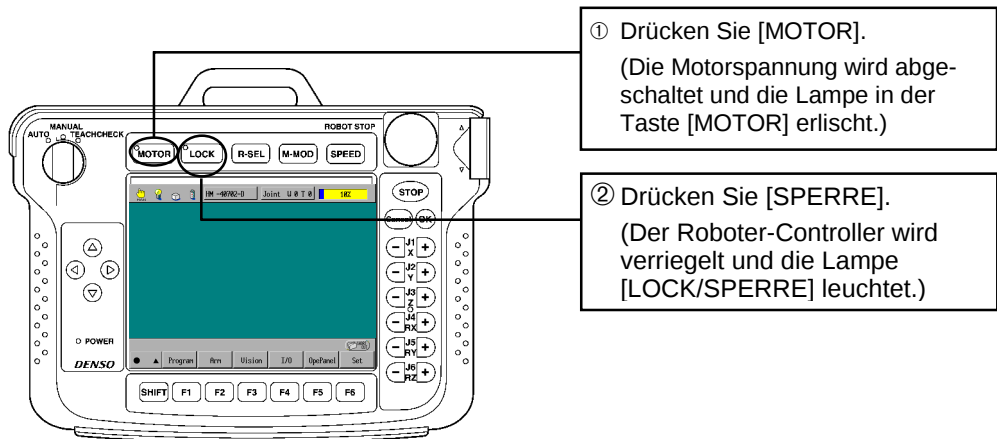
③ Drücken Sie [Abbrechen].  
(Der Benutzer kehrt zur Hauptanzeige zurück.)

## 3.2 Aktivieren der Maschinenverriegelung des Roboter-Controller

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

### SCHRITT 1 Ausschalten des Motors

### SCHRITT 2 Aktivieren der Maschinenverriegelung des Roboters



### ★Vorsicht★

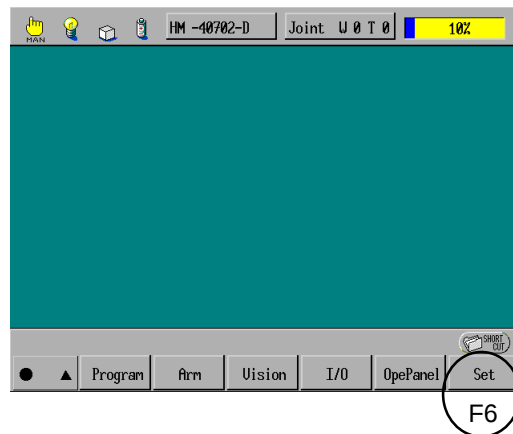
Bevor Sie die Maschinenverriegelung des Roboter-Controller aktivieren, stellen Sie sicher, daß der Motor ausgeschaltet ist, d. h. daß die "MOTOR"-Lampe aus ist.

### 3.3 Einstellen des Kommunikationsanschlusses des Roboter-Controller

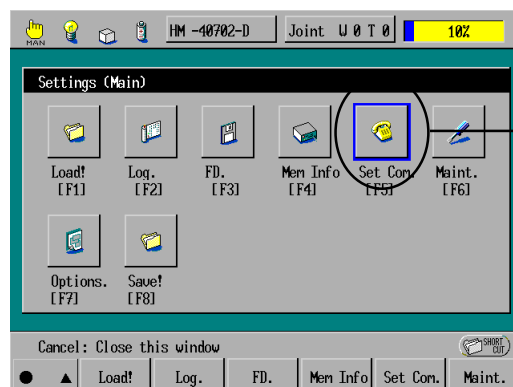
Damit der Roboter-Controller mit dem PC Daten austauschen kann, müssen Sie den Kommunikationsanschluß einstellen.

Dieser Teilabschnitt beschreibt die am häufigsten verwendete Anschlußmethode mit RS232C.

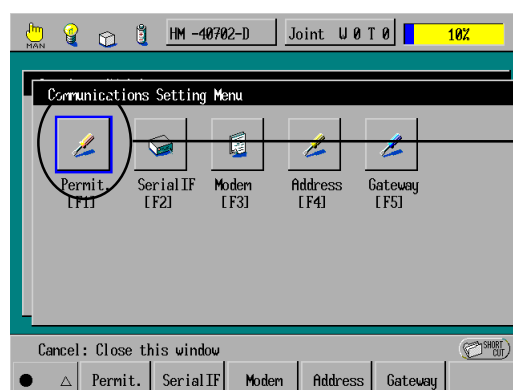
#### SCHRITT 1 Einstellen der Kommunikationsberechtigung



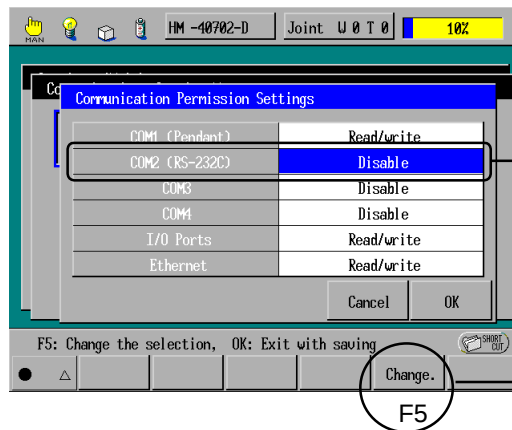
① Drücken Sie [F6 Einstellen].



② Drücken Sie [F5 Einst. KOMM].

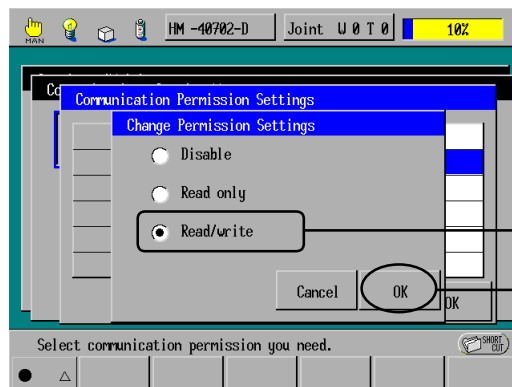


③ Drücken Sie [F1 Berecht.].



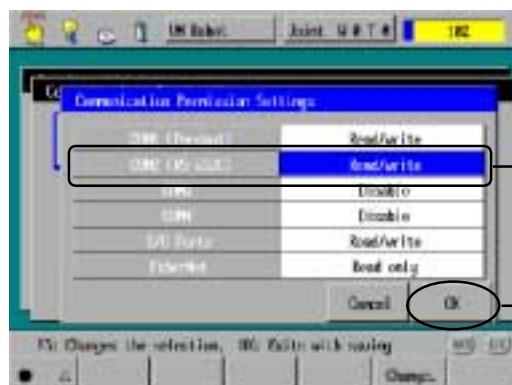
④ Wählen Sie die Zeile [COM2 (RS232C)].

⑤ Drücken Sie [F5 Ändern].



⑥ Drücken Sie [Lesen/Schreiben].

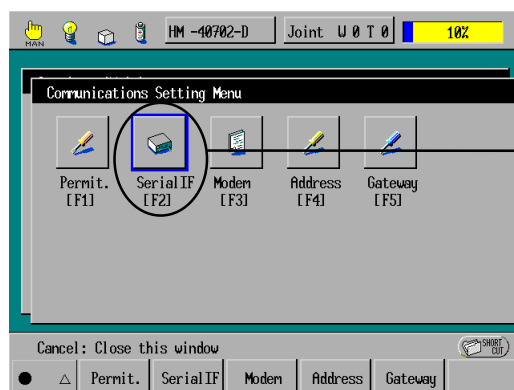
⑦ Drücken Sie [OK].



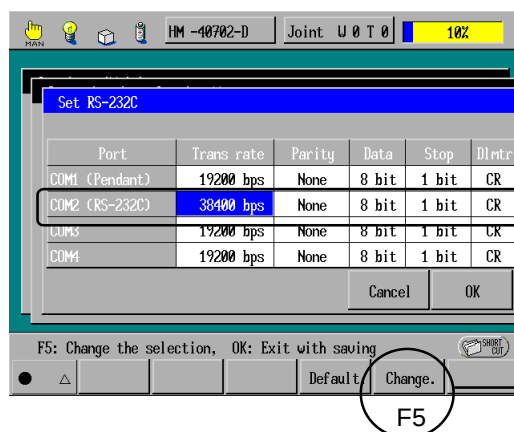
Die Spalte [COM2 (RS232C)] wird auf [Lesen/Schreiben] geändert.

⑧ Drücken Sie [OK].

## SCHRITT 2 Einstellen der Übertragungsrate

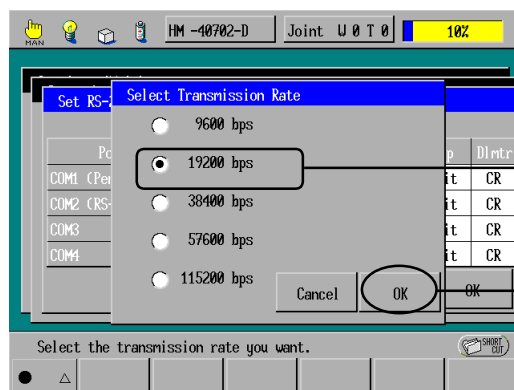


① Drücken Sie [F2 Serial IF].



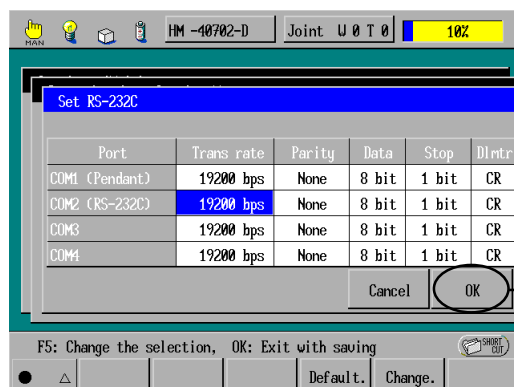
② Wählen Sie die Zeile [COM2 RS232C].

③ Drücken Sie [F5 Ändern].

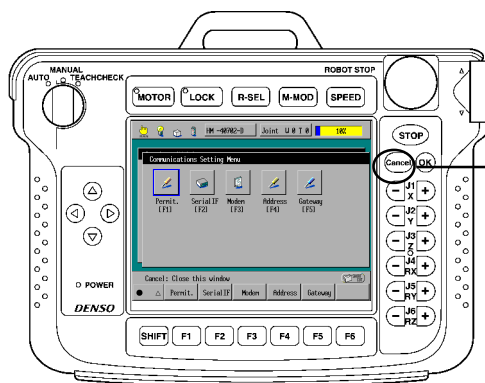


④ Wählen Sie [19200 BPS] aus.

⑤ Drücken Sie [OK].



⑥ Drücken Sie [OK].



⑦ Drücken Sie [Abbrechen].  
(Die Anzeige kehrt zum  
[★Kommunikationseinstellungs  
menü] 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 PC vornehmen.

### 4.1 Starten des System Manager

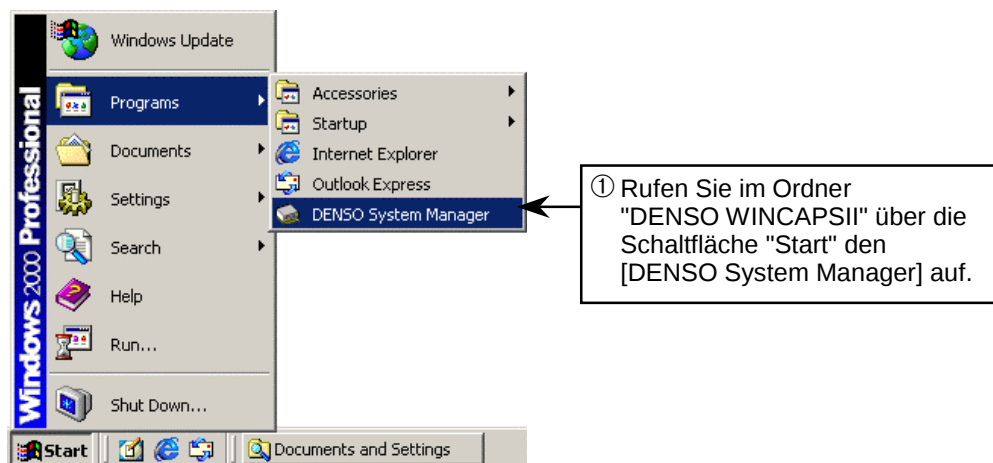
WINCAPSII besteht aus folgenden Funktionsmodulen:

- PAC-Programm-Manager
- Variable Manager
- DIO Manager
- Arm Manager
- Vision Manager
- Log Manager
- Kommunikationseinstellungsmanager

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:

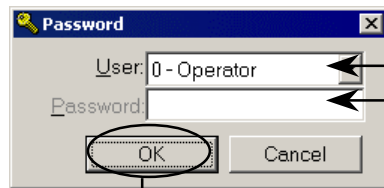
#### SCHRITT 1 Auswählen des DENSO System Managers



#### ★Vorsicht★

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.

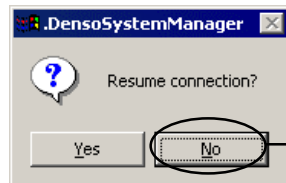
## SCHRITT 2 Auswählen der Benutzerebene



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★

Unten sehen Sie die Start-Schaltflächen der Funktionsmodule auf der Symbolleiste.



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

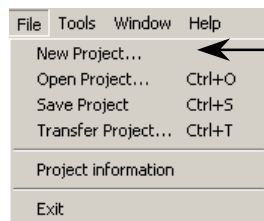
## 4.2 Registrieren eines neuen Systemprojektes

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 Gruppe in einem Projekt zu verarbeiten.

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

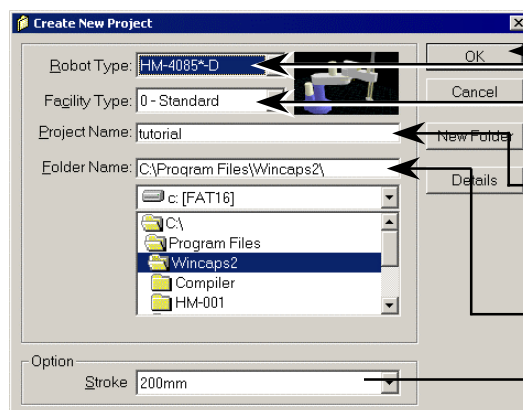
**Vorsicht:** Wenn Sie den System Manager zum ersten Mal starten, erscheint das Dialogfeld [Neues Projekt erstellen]. Führen Sie zuerst "Neue Projektregistrierung" aus, bevor Sie mit Schritt 2 in 4.1 "Starten des System Managers" fortfahren.

### SCHRITT 1 Auswählen von "Neues Projekt" aus dem Menü "Datei"



① Hier klicken.

### SCHRITT 2 Registrieren eines neuen Projektes



① Wählen Sie HM-4085\*-D.

② Wählen Sie 0-Standard.

③ Geben Sie den gewünschten Projektnamen ein. (Geben Sie bei diesem Beispiel "Tutorial/Lernprogramm" ein.)

④ Geben Sie den Ordernamen an.

⑤ Wählen Sie 200 mm.

⑥ Klicken Sie [OK] an.

## 4.3 Einstellen des Kommunikationsanschlusses des PC

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

### SCHRITT 1 Aufrufen des Dialogfeldes "ROBOTalk Manager"

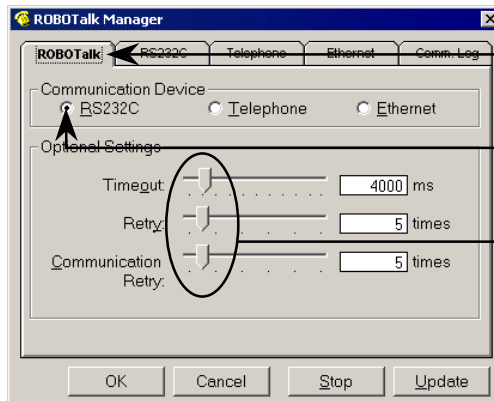


① Klicken Sie die Schaltfläche [Kommunikationseinstellung] an.

★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 Einstellen des Kommunikationsgeräts und der optionalen Einstellungen



② Klicken Sie das Register [ROBOTalk] an.

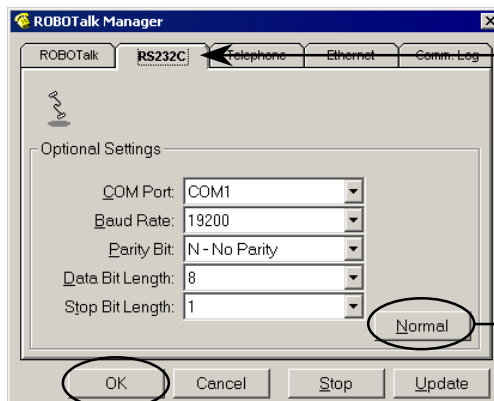
③ Hier klicken, um RS232C auszuwählen.

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 Einstellen der RS232C-Kommunikationsoptionen



⑤ Klicken Sie das Register [RS232C] an.

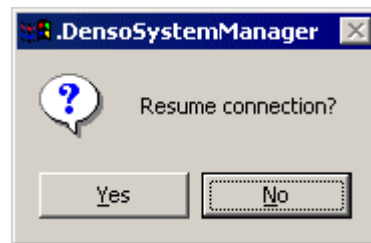
⑥ Klicken Sie [Normal] an.  
Das Feld "Optionale Einstellungen" wird auf "Normale Einstellungen" geändert.

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 PC oder Roboter-Controller 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.



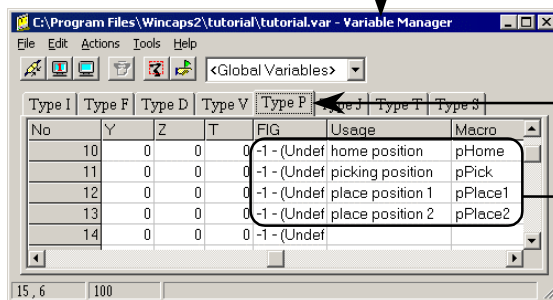
# Lektion 5 Definieren von Makros

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

## SCHRITT 1 Erstellen einer Variablen-Makrodefinitionsdatei



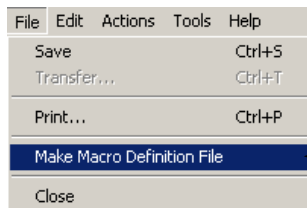
① Klicken Sie die Schaltfläche "Variable Manager" an, um den "Variable Manager" zu starten.



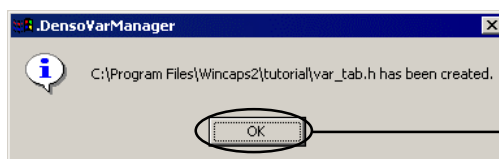
Das Fenster [Variable Manager] erscheint.

② Klicken Sie das Register [Typ P] an.

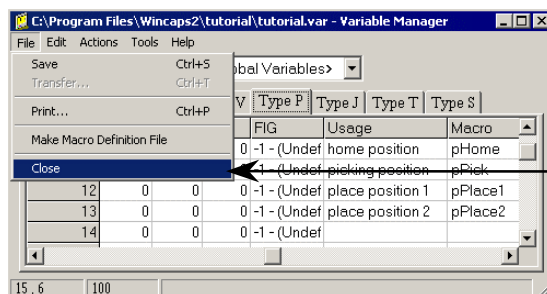
③ Doppelklicken Sie auf jedes Feld, und geben Sie die Verwendung und den Makronamen der Variablen an. Geben Sie bei diesem Beispiel die Positionsvariablen P 10 bis P 13 an.



④ Klicken Sie [Makrodefinitionsdatei erstellen] an.



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

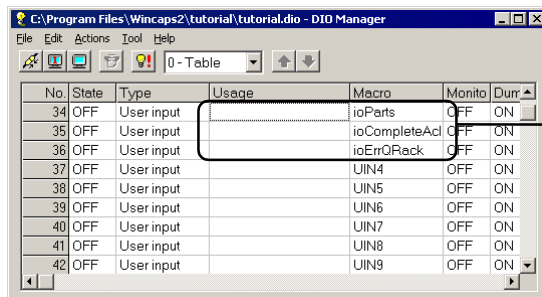


⑥ Hier klicken, um den Variable Manager zu verlassen.

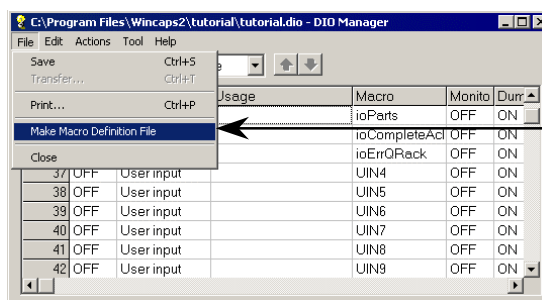
## SCHRITT 2 Erstellen einer E/A-Makrodefinitionsdatei



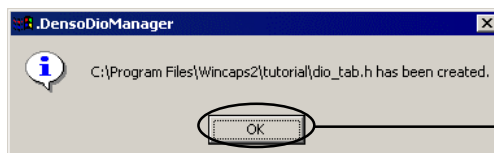
① Klicken Sie die Schaltfläche "DIO Manager" an, um den "DIO Manager" zu starten.



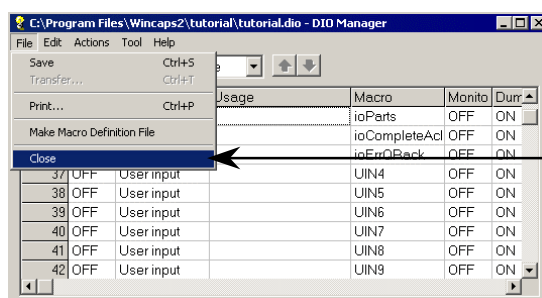
② Doppelklicken Sie auf jedes Feld, und geben Sie die Verwendung und den Makronamen der E/As an.



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 das Codierungslistenmuster 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 des Programmbearbeitungsfensters

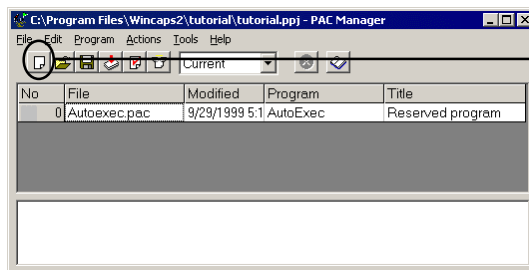
Zur Eingabe und Bearbeitung von Task-Programmen verwenden Sie den Programm-Manager, der über den System Manager aufgerufen wird.

### SCHRITT 1 Starten des PAC-Programm-Managers

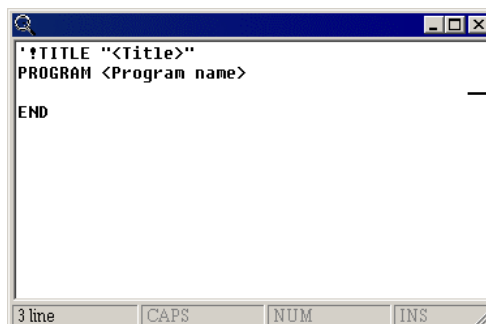


① Klicken Sie die Schaltfläche "PAC-Programm-Manager" an, um den PAC-Programm-Manager zu starten.

### SCHRITT 2 Öffnen eines Programmbearbeitungsfensters



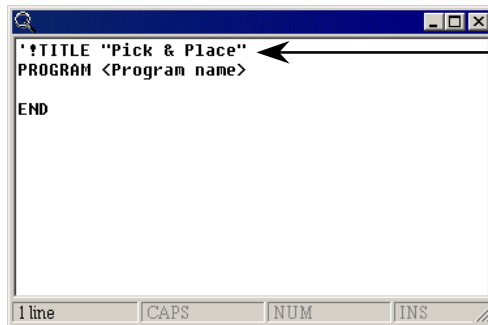
② Klicken Sie die Schaltfläche [Neues Programm] an.



Ein neues Bearbeitungsfenster wird geöffnet.

## 6.3 Eingeben von Programmcodes

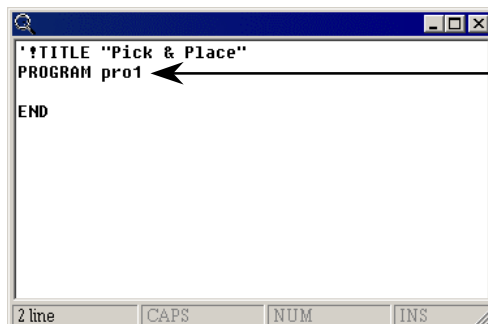
### SCHRITT 1 Eingeben des Programmtitels



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

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

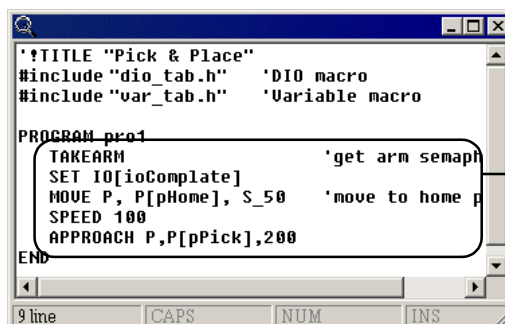
### SCHRITT 2 Eingeben des Programmnamens



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

② Geben Sie den Programmnamen ein. (Geben Sie bei diesem Beispiel "Pro1" ein.)

### SCHRITT 3 Eingeben der Programmcodes



```
'!TITLE "Pick & Place"
#include "dio_tab.h" 'DIO macro
#include "var_tab.h" 'Variable macro

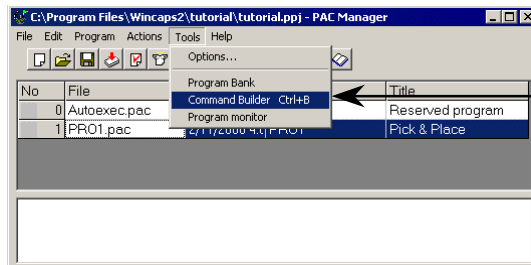
PROGRAM pro1
TAKEARM
SET IO[ioComplete]
MOVE P, P[pHome], S_50
SPEED 100
APPROACH P,P[pPick],200
END
```

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

## 6.4 Einsetzen des Befehlseditors

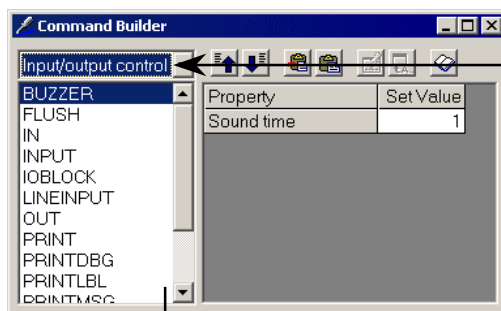
Sie können wie bei einer Textverarbeitung Programmcodes direkt über die Tastatur in das Programmbearbeitungsfenster eingeben. Der Programm-Manager ist jedoch mit einer Befehlseditorfunktion ausgestattet, die eine einfache Befehlseingabe ermöglicht. In diesem Abschnitt wird beschrieben, wie Sie mit dem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

### SCHRITT 1 Auswählen des Befehlseditors



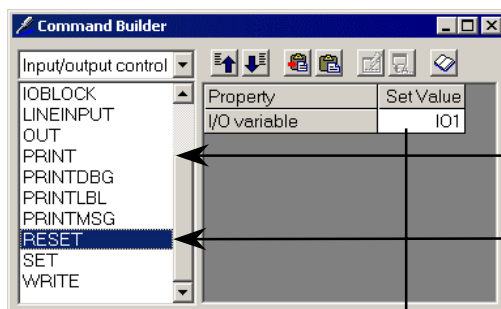
① Wählen Sie aus dem Menü [Extras] den [Befehlseditor] aus.

### SCHRITT 2 Einsetzen des Befehlseditors



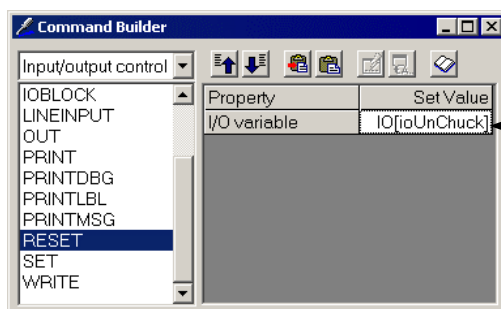
② Klicken Sie die [Klassenauswahl]-Liste im Fenster [Befehlseditor] an, und wählen Sie aus der Liste "E/A-Steuer" anweisungen.

Befehlsliste



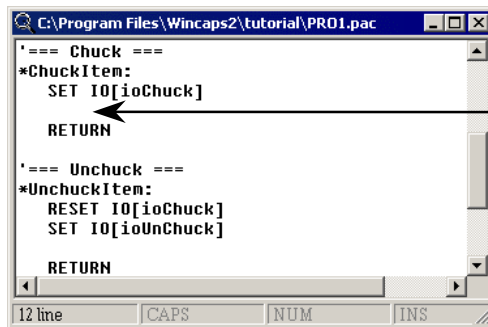
③ Blättern Sie über die Bildlaufleiste der E/A-Steueranweisungen nach unten, um den Befehl [Rücksetzung] anzuzeigen.

④ Klicken Sie den Befehl [Rücksetzung] an.

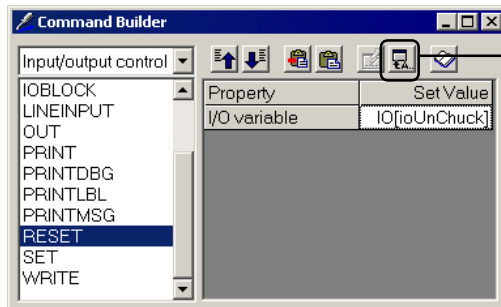


⑤ Doppelklicken Sie auf das Feld [E/A-Variable, Wert setzen].

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

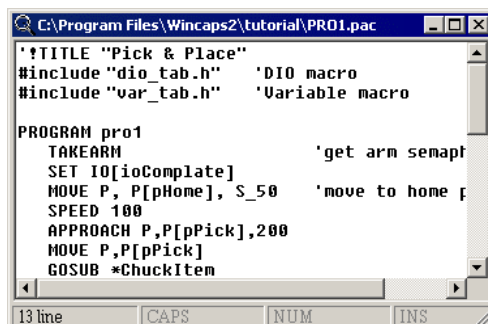


⑦ Klicken Sie im Programmbearbeitungsfenster die Position an, von wo aus Sie Befehle eingeben wollen.



⑧ Klicken Sie die Schaltfläche [Einfügen] an.

### SCHRITT 3 Bearbeiten von Programmcodes zur besseren Übersicht



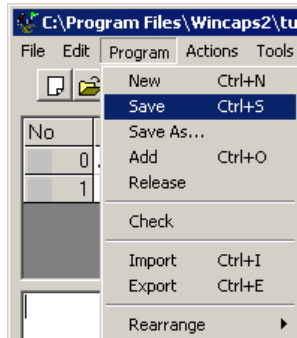
★**Vorsicht**★

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

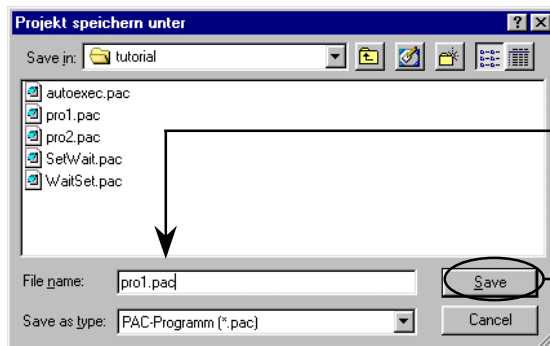
In diesem Abschnitt werden Sie PRO1, das im vorherigen Abschnitt 6.4 "Einsetzen des Befehlseditors" erstellte Programm, speichern.

### SCHRITT 1 Auswählen von "Speichern"



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 ein.  
(Die Dateierweiterung wird automatisch angehängt.)

③ Klicken Sie [Speichern] an. Die Programm Quelldatei "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 ein Laufzeitformat

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

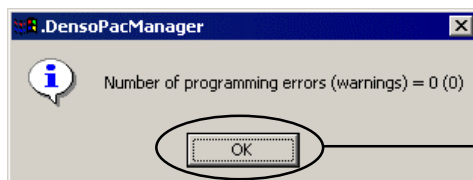
### SCHRITT 1 Kompilieren des Programmes in ein Laufzeitformat



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 Meldungsteilfenster des Fensters [Denso PAC-Manager] angezeigt.

### SCHRITT 2 Sicherstellen, 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 überprüfen Sie das Programm 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

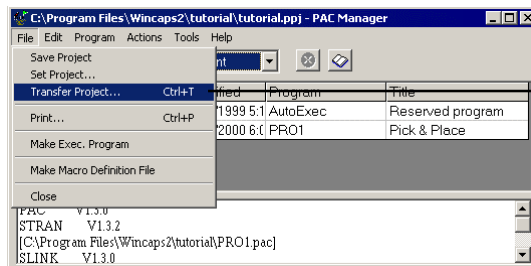


① Klicken Sie die Schaltfläche [Verbindungen] an.

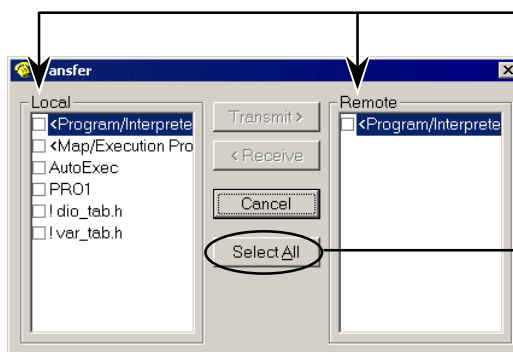


Im System Manager wird dementsprechend dann das Symbol der Schaltfläche [PAC-Program-Manager] geändert.

### SCHRITT 2 Auswählen des hochzuladenden Programmes

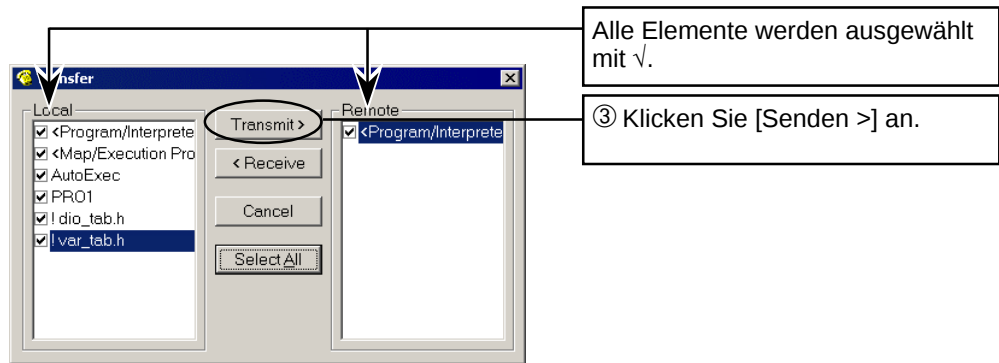


① Wählen Sie [Projekt übertragen...].

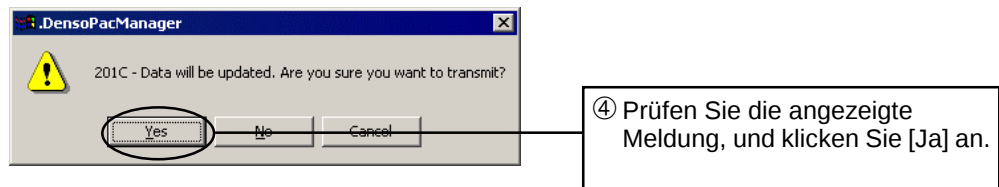


"Lokal" bezieht sich auf den PC und "Fern" auf den Roboter-Controller.

② Klicken Sie [Alle auswählen] an.



### SCHRITT 3 Hochladen des ausgewählten Programmes



Die Programmdatei wird nun zum Roboter-Controller hochgeladen.



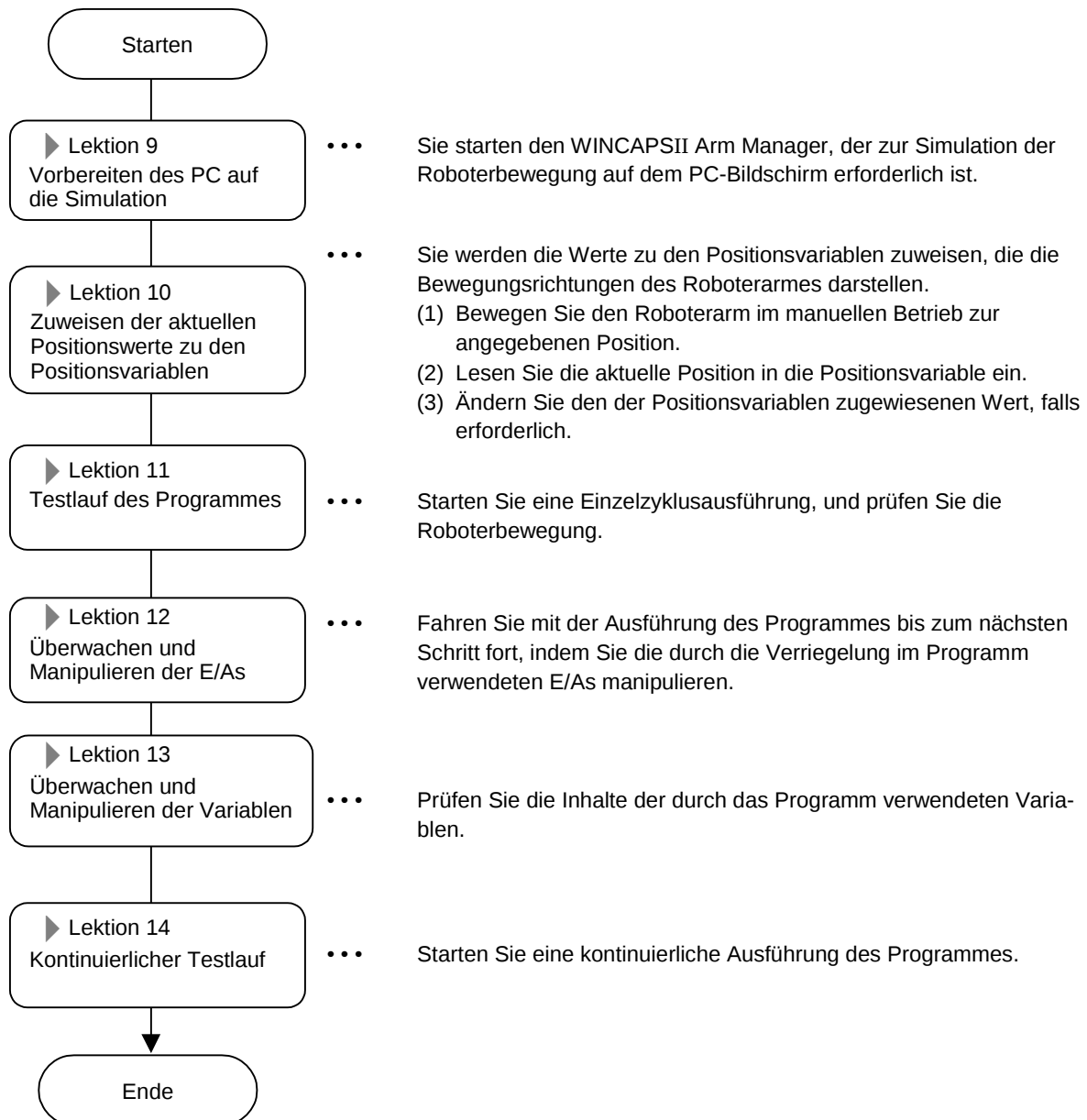
# Kapitel 3 Simulieren der Roboterbewegung auf einem PC mit dem erstellten Programm

In Kapitel 3 lernen Sie folgendes:

Ausführen des Programmes, das Sie auf einem PC erstellt und zum Roboter-Controller hochgeladen haben, mit aktivierter Maschinenverriegelung, 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 Leistungsfähigkeit des entwickelten Programmes erhöhen.

|              |                                                                                                   |    |
|--------------|---------------------------------------------------------------------------------------------------|----|
| ► Lektion 9  | Vorbereiten des PC auf die Simulation                                                             | 67 |
| 9.1          | Starten des Arm Manager                                                                           | 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         | Zuweisen der aktuellen Position zur Positionsvariablen                                            | 69 |
| 10.3         | Bearbeiten der Positionsvariablen                                                                 | 71 |
| ► Lektion 11 | Testlauf des Programmes                                                                           | 73 |
| 11.1         | Laden des Programmes                                                                              | 73 |
| 11.2         | Starten des Programmes                                                                            | 75 |
| ► Lektion 12 | Überwachen und Manipulieren der E/As                                                              | 76 |
| 12.1         | Starten des DIO Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller       | 76 |
| 12.2         | Überwachen der E/As                                                                               | 77 |
| 12.3         | Ein-/Ausschalten der E/A-Dummy-Schalter                                                           | 78 |
| ► Lektion 13 | Überwachen und Manipulieren der Variablen                                                         | 79 |
| 13.1         | Starten des Variablen Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller | 79 |
| 13.2         | Überwachen der Variablen                                                                          | 80 |
| ► Lektion 14 | Kontinuierlicher Testlauf                                                                         | 81 |
| 14.1         | Kontinuierliche Ausführung                                                                        |    |
| 14.2         | Kontinuierliche Überwachung der E/As                                                              | 82 |
| 14.3         | Sofortiger Stop                                                                                   | 84 |



## Lektion 9 Vorbereiten des PC auf die Simulation

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

### 9.1 Starten des Arm Manager

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

#### SCHRITT 1

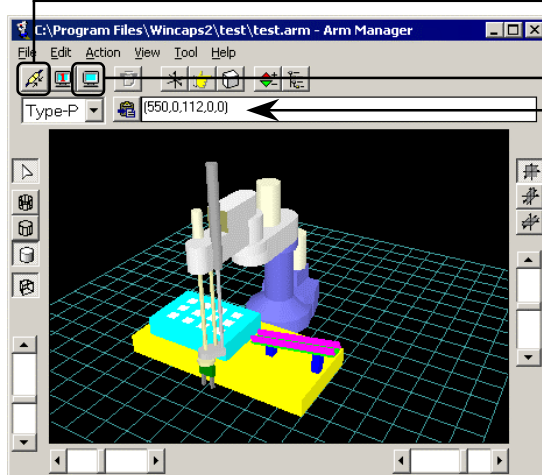


① Klicken Sie die Schaltfläche [Arm] an.

### 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.)

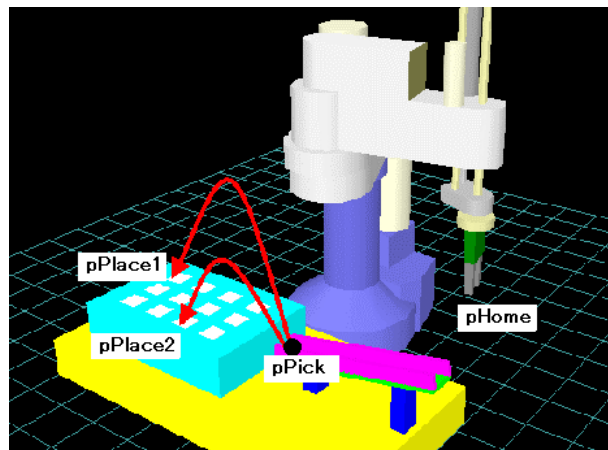
Die Kommunikationsverknüpfung zwischen Arm Manager und Roboter-Controller ist hergestellt, um den Datenaustausch zwischen beiden zu ermöglichen.

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

## Lektion 10 Zuweisen der aktuellen Positionswerte zu den Positionsvariablen

Vor Ausführung des Programmes ist es erforderlich, die den jeweiligen Positionsvariablen "pHome", "pPick", "pPlace1" und "pPlace2" des in "Lektion 6 "Eingeben und Bearbeiten von Programmen" von Kapitel 2 "Erstellen eines Programmes auf einem PC mit WINCAPSII" erstellten Programmes zugewiesenen Werte festzulegen.

Bei dieser Lektion werden Sie die Werte des Programmiergeräts über "Punktprogrammierung" eingeben.

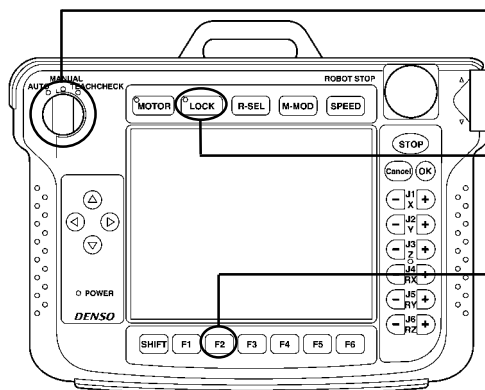


★**Achtung**★ Siehe Lektion 5 "Definieren von Makros" für jede Positionsvariablennummer für "pHome", "pPick", "pPlace1" und "pPlace2".

### 10.1 Manuelles Simulieren der Roboterbewegung

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.

#### SCHRITT 1 Auswählen des manuellen Modus des Roboters und Anzeigen der aktuellen Roboterposition

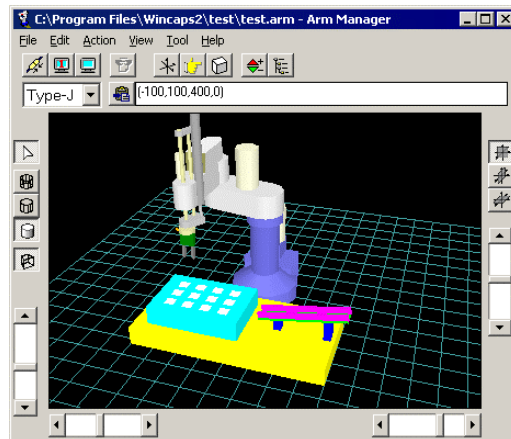


① Stellen Sie den Moduswahlschalter auf MANUELLE Position, um in den "Manuellen Modus" zu schalten.

② Drücken Sie [SPERRE].

③ Drücken Sie [F2 Arm] in der Hauptanzeige.  
Das Fenster "Aktuelle Roboterposition" erscheint. Dort müssen Sie zum Fenster "Gelenkvariablen" wechseln.

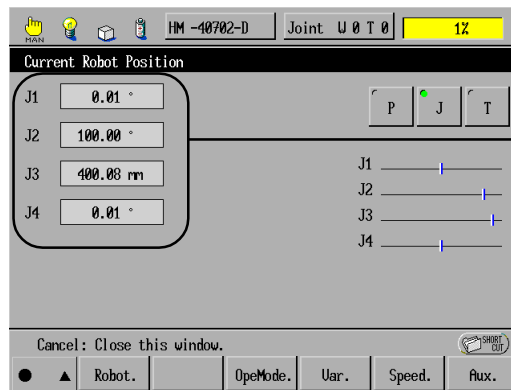
## SCHRITT 2 Bewegen des Roboterarmes



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

- J1: ca. 0 G.
- J2: ca. 100 G.
- J3: ca. 400 mm
- J4: 0,00 G. (keine Bewegung)

## SCHRITT 3 Prüfen der aktuellen Position

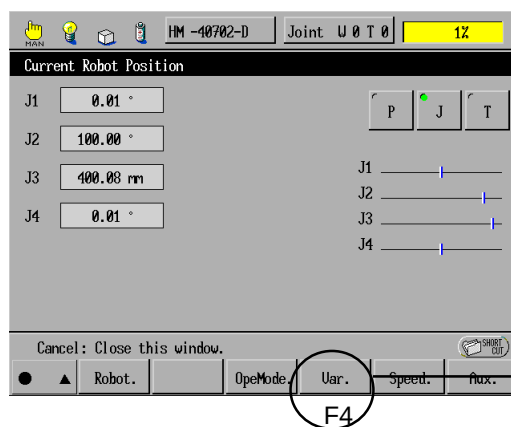


⑤ Prüfen Sie im Fenster [Aktuelle Roboterposition] die Position jeder Achse.

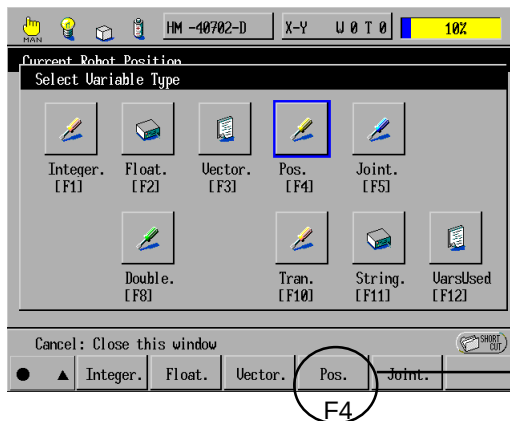
## 10.2 Zuweisen der aktuellen Position zur Positionsvariablen

Weisen Sie die Werte der Position zu, zu der der Roboterarm im Arm Manager bewegt wurde.

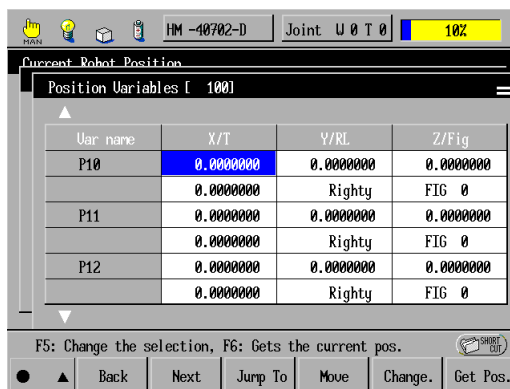
### SCHRITT 1 Anzeigen der Positionsvariablen



① Drücken Sie [F4 Var.], wenn das Fenster [Aktuelle Roboterposition] angezeigt wird.

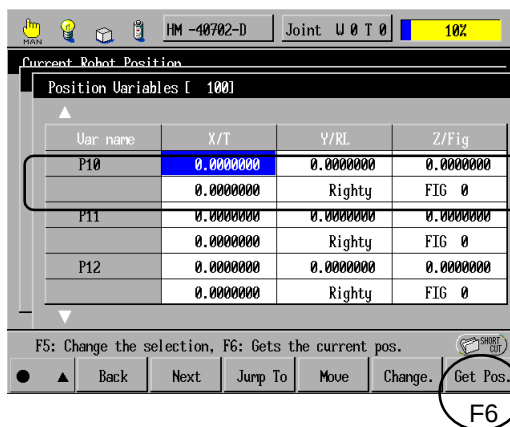


② Drücken Sie im Fenster [Variablentyp auswählen] [F4 Pos.].



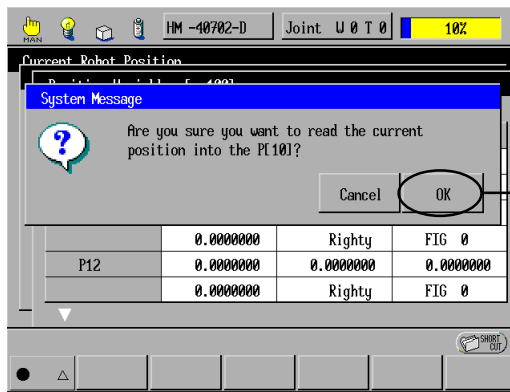
Das Fenster [Positionsvariablen] erscheint.

## SCHRITT 2 Zuweisen des aktuellen Positionswertes zur Positionsvariablen



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

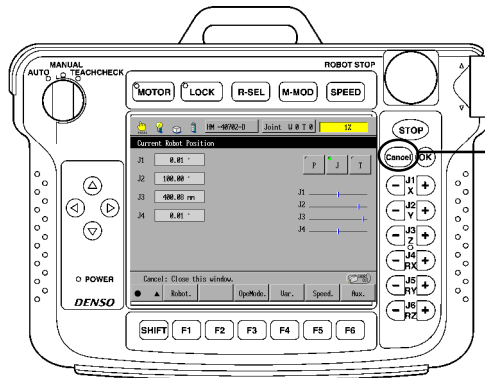
④ Drücken Sie [F6 Pos. abbr.].



⑤ Drücken Sie [OK].

Der aktuelle Positionswert des Roboterarmes wird der Positionsvariablen P[10] zugewiesen.

### SCHRITT 3 Zuweisen sonstiger Positionswerte zu Positionsvariablen



⑥ Drücken Sie zweimal [Abbrechen].

Es erscheint wieder das Fenster [Aktuelle Roboterposition].

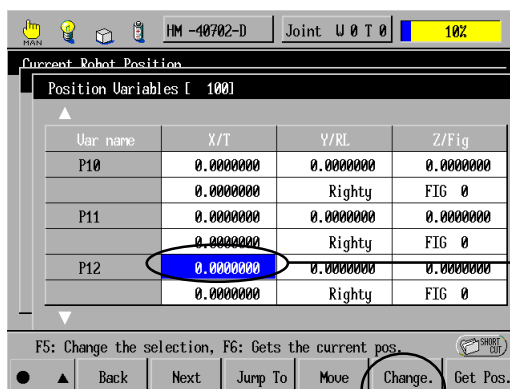
⑦ Gehen Sie zurück zu Schritt 2 "Bewegen des Roboterarmes" in Lektion 10.1.

## 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" wie unten dargestellt angezeigt.

### SCHRITT 1 Auswählen der Positionsvariablen

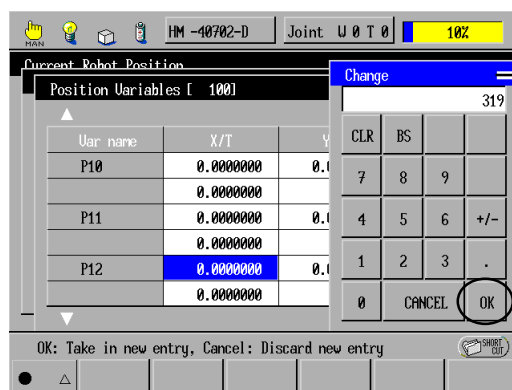


① Über das Einstellrad oder die Cursor-Tasten wählen Sie die Spalte [X/T] der Positionsvariablen "P12".

② Drücken Sie [F5 Ändern].

Die Zehnertastatur erscheint wie auf der nächsten Seite dargestellt.

## SCHRITT 2 Ändern des "P12" zugewiesenen Wertes



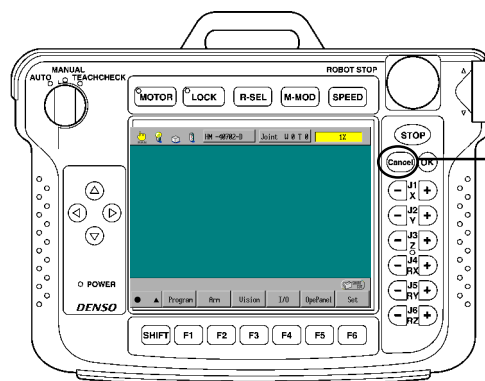
① Geben Sie einen neuen Wert über die Zifferntasten der Zehner-tastatur ein.

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 "P12" eingegeben.

## Zurück zur Hauptanzeige



③ Drücken Sie dreimal die Taste [Abbrechen], um zur Hauptanzeige zurückzukehren.

Damit ist die Bearbeitung von Positionsvariablen abgeschlossen.

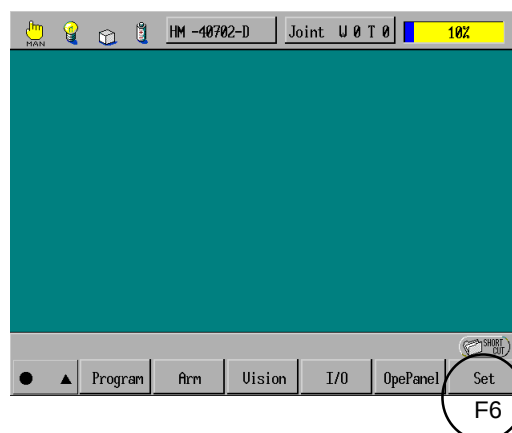
# Lektion 11 Testlauf des Programmes

Sie werden in der Einzelzyklusausführung einen Testlauf des Programmes "PRO1" starten.

## 11.1 Laden des Programmes

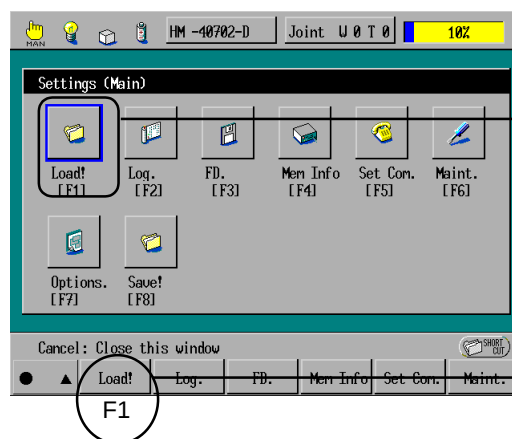
Selbst wenn ein kompiliertes Programm vom PC zum Roboter-Controller hochgeladen wird, kann es der Controller nicht ausführen. Das Programm muß in den Speicherbereich geladen werden, von wo aus es ausgeführt werden kann.

### SCHRITT 1 Anzeigen des Ladebildschirmes

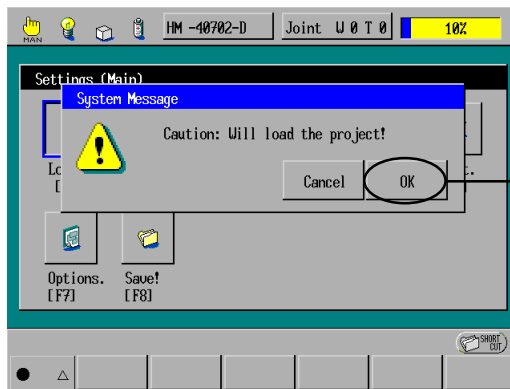


① Drücken Sie in der Hauptanzeige [F6 Einstellen].

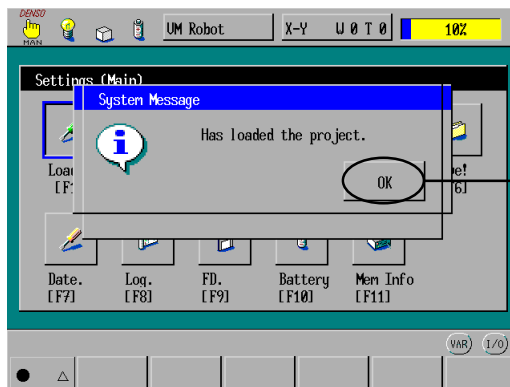
### SCHRITT 2 Laden des Projektes



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

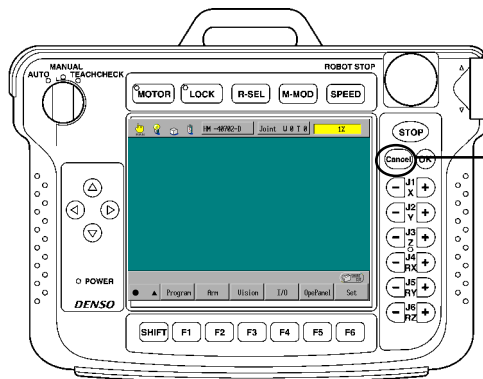


③ Prüfen Sie die Systemmeldung, und drücken Sie [OK].



④ Prüfen Sie die Systemmeldung, und drücken Sie [OK].

### Zurück zur Hauptanzeige



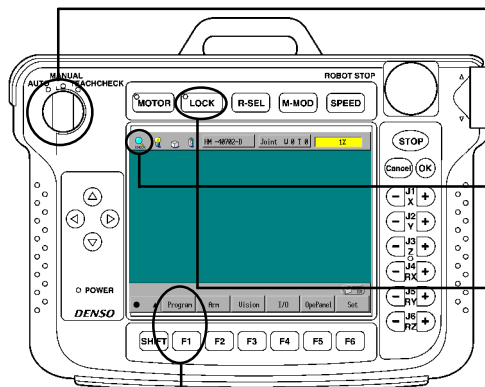
⑤ Drücken Sie [Abbrechen], um zur Hauptanzeige zurückzukehren.

## 11.2 Starten des Programmes

Jetzt aktivieren Sie die Maschinenverriegelung des Roboter-Controller und starten das geladene Programm in der Einzelzyklusausführung, 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, da Sie in Lektion 12 E/As manipulieren werden.

### SCHRITT 1 Anzeigen des Fensters "Programmliste" im Programmierprüfmodus



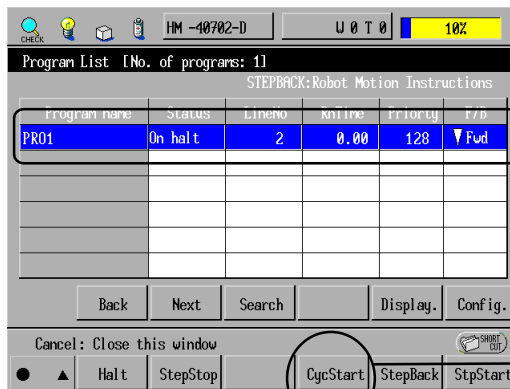
① Stellen Sie den Modusauswahlschalter auf die Position PROGRAMMIERPRÜFUNG ein. (Der Roboter-Controller wechselt in den Programmierprüfmodus.)

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

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 Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller

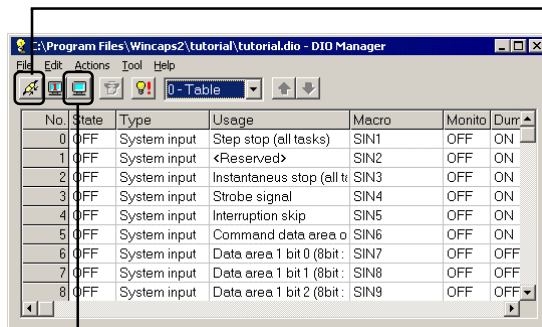
Sie starten den DIO Manager und stellen die Kommunikationsverknüpfung zum Roboter-Controller her, um den Datenaustausch zwischen PC und Roboter-Controller zu ermöglichen.

#### SCHRITT 1 Starten des DIO Manager



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

#### SCHRITT 2 Verbinden mit dem Roboter-Controller zum Starten der kontinuierliche Überwachung



- 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.

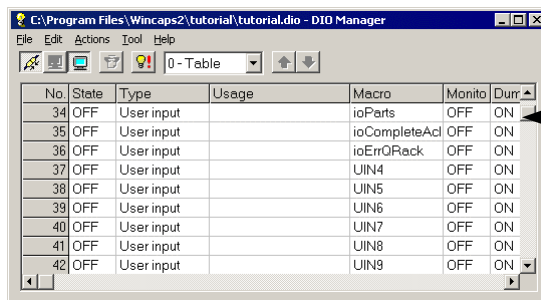
## 12.2 Überwachen der E/As

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 Tabellentypanzeige, um "ioParts" an E/A Nr. 34 zu prüfen.

Die Tabellentypanzeige 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

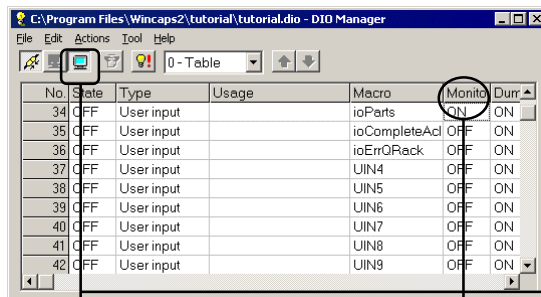


The screenshot shows the 'DIO Manager' window with a menu bar (File, Edit, Actions, Tool, Help) and a toolbar. A dropdown menu is set to '0 - Table'. Below the toolbar is a table with the following data:

| No. | State | Type       | Usage | Macro         | Monitor | Durr |
|-----|-------|------------|-------|---------------|---------|------|
| 34  | OFF   | User input |       | ioParts       | OFF     | ON   |
| 35  | OFF   | User input |       | ioCompleteAck | OFF     | ON   |
| 36  | OFF   | User input |       | ioErrQRack    | OFF     | ON   |
| 37  | OFF   | User input |       | UIN4          | OFF     | ON   |
| 38  | OFF   | User input |       | UIN5          | OFF     | ON   |
| 39  | OFF   | User input |       | UIN6          | OFF     | ON   |
| 40  | OFF   | User input |       | UIN7          | OFF     | ON   |
| 41  | OFF   | User input |       | UIN8          | OFF     | ON   |
| 42  | OFF   | User input |       | UIN9          | OFF     | ON   |

- 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 Einstellen des Monitortyps



The screenshot is similar to the previous one, but with two annotations: a box around the 'Monitor' column header and a circle around the 'ON' value in the 'Monitor' column for E/A Nr. 34.

| No. | State | Type       | Usage | Macro         | Monitor | Durr |
|-----|-------|------------|-------|---------------|---------|------|
| 34  | OFF   | User input |       | ioParts       | ON      | ON   |
| 35  | OFF   | User input |       | ioCompleteAck | OFF     | ON   |
| 36  | OFF   | User input |       | ioErrQRack    | OFF     | ON   |
| 37  | OFF   | User input |       | UIN4          | OFF     | ON   |
| 38  | OFF   | User input |       | UIN5          | OFF     | ON   |
| 39  | OFF   | User input |       | UIN6          | OFF     | ON   |
| 40  | OFF   | User input |       | UIN7          | OFF     | ON   |
| 41  | OFF   | User input |       | UIN8          | OFF     | ON   |
| 42  | OFF   | User input |       | UIN9          | OFF     | ON   |

- 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.

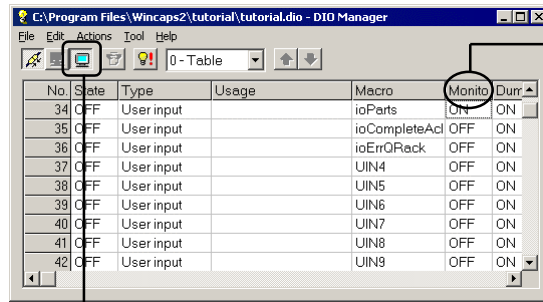
- ④ Klicken Sie die Schaltfläche [Kontinuierlicher Monitor] an, um den Monitor einzuschalten.  
Entsprechend der an E/A vorgenommenen Änderungen wird der "Status" von E/A Nr. 34 als "EIN" oder "AUS" angezeigt.  
In diesem Beispiel werden die "ioParts" von E/A Nr. 34 als "AUS" angezeigt.

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

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

Das Programm PRO1 "PICK & PLACE" wartet auf E/A Nr. 34 "ioParts", um sich einzuschalten und fährt in diesem Status nicht mit dem nächsten Schritt fort. Durch die künstliche Verwaltung der E/A, können Sie einen Betriebstest durchführen.

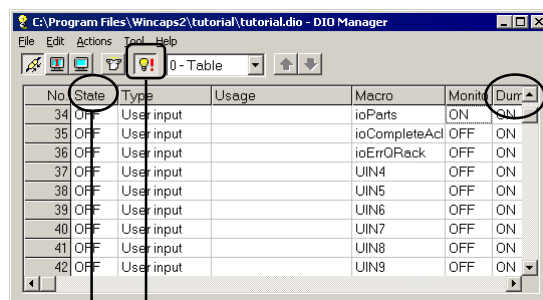
### SCHRITT 1 Stoppen der Überwachung



① Prüfen Sie im Fenster [DIO Manager], ob das Feld [Monitor] von E/A (Nr. 34) auf "EIN" gesetzt ist.

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

### SCHRITT 2 Einstellen des Dummy-Schalters



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 den 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 Manager und Einrichten der Kommunikationsverknüpfung zum Roboter-Controller

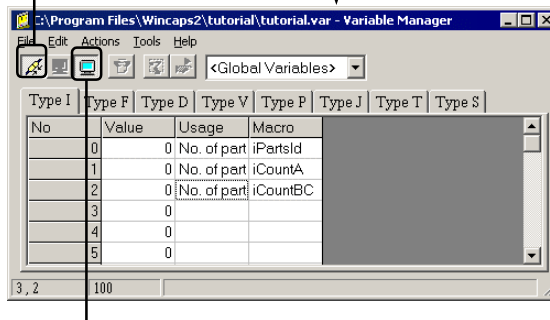
Starten Sie den Variablen Manager und stellen Sie die Kommunikationsverbindung her, damit ein konstanter Datenaustausch mit dem Roboter-Controller gewährleistet ist.

#### SCHRITT 1 Starten des 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.

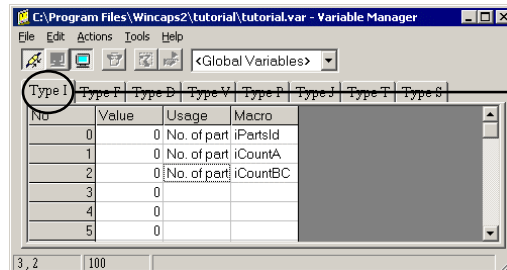
③ Klicken Sie [Kontinuierlicher Monitor] an, um den Monitor zu starten.  
Die Schaltfläche [Kontinuierlicher Monitor] wird gedrückt dargestellt.

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

## 13.2 Überwachen der Variablen

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

### SCHRITT 1 Anzeigen der Variablen



Die Spalte [Wert] für die Ganzzahl-Variable [0] zeigt die an der Variablen vorgenommene Änderung an.

## Lektion 14 Kontinuierlicher Testlauf

Bis zur vorherigen Lektion haben Sie PRO1 mit der Einzelzyklusausführung 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.

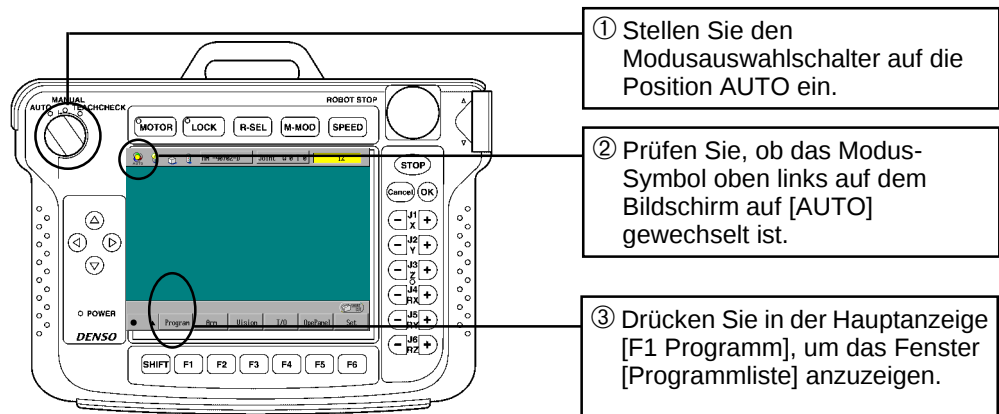
Bevor Sie den Roboter tatsächlich steuern, aktivieren Sie die Maschinenverriegelung des Roboter-Controller und führen einen kontinuierlichen Testlauf durch. Überwachen Sie dabei im Arm Manager die simulierten Roboterbilder.

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

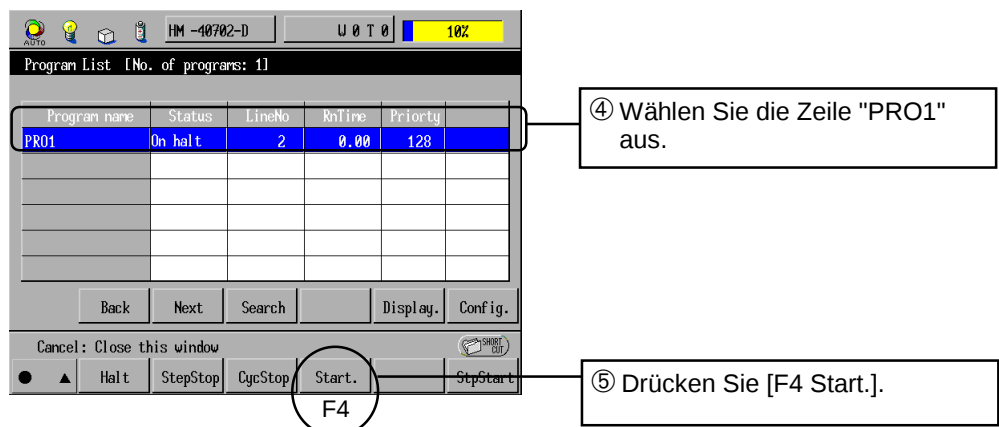
### 14.1 Kontinuierliche Ausführung

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.

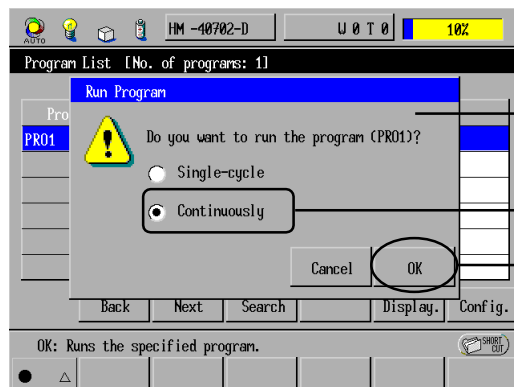
#### SCHRITT 1 Anzeigen des Fensters "Programmliste" im Auto-Modus



#### 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 bei "EIN" geschaltetem [Monitor]-Feld 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 Manager zunächst erscheint.

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

| No. | State | Type       | Usage | Macro         | Monitor | Dum |
|-----|-------|------------|-------|---------------|---------|-----|
| 34  | OFF   | User input |       | ioParts       | ON      | ON  |
| 35  | OFF   | User input |       | ioCompleteAck | ON      | ON  |
| 36  | OFF   | User input |       | ioErrORack    | ON      | ON  |
| 37  | OFF   | User input |       | UIN4          | OFF     | ON  |
| 38  | OFF   | User input |       | UIN5          | OFF     | ON  |
| 39  | OFF   | User input |       | UIN6          | OFF     | ON  |
| 40  | OFF   | User input |       | UIN7          | OFF     | ON  |
| 41  | OFF   | User input |       | UIN8          | OFF     | ON  |
| 42  | OFF   | User input |       | UIN9          | OFF     | ON  |

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.)

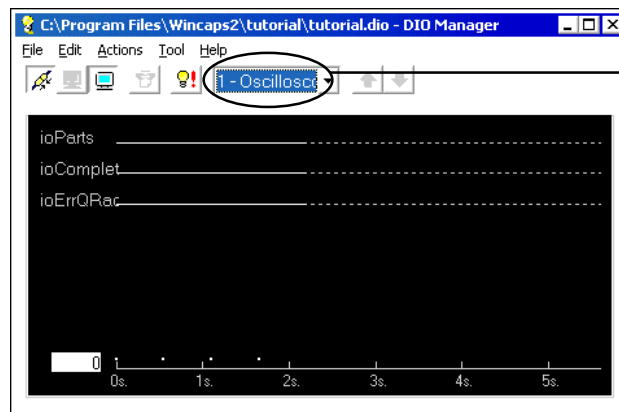
② Doppelklicken Sie im Fenster [DIO Manager] auf die angezeigten Felder [Monitor] der E/As Nr. 34-36, um diese auf "EIN" zu setzen.

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

## Oszilloskoptyp

In der Tabellentypanzeige des Fensters [DIO Manager] können Sie den E/A-Status mit "EIN" geschalteter [Monitor]-Software wie auf einem Oszilloskop beobachten.

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



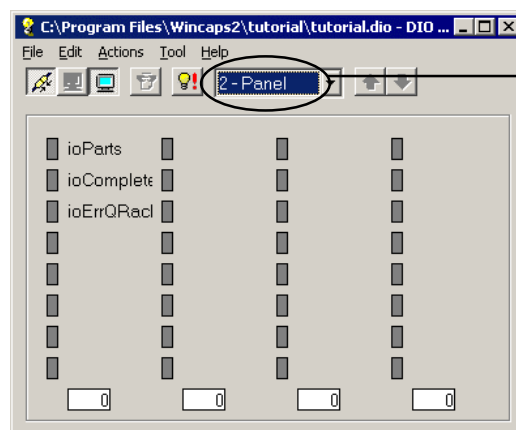
① Klicken Sie die Schaltfläche  an, und wählen Sie [1 – Oszilloskop] aus der angezeigten Liste.  
Die Zustände der E/A Nr. 34 bis 36 werden wie bei einem Oszilloskop angezeigt.

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

## 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 "EIN"geschaltetem [Monitor] im Tabellentypformat anzeigen.

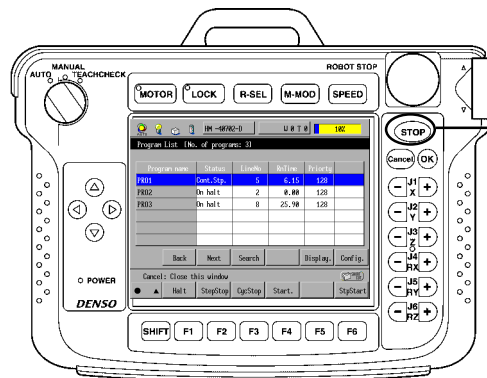


① Klicken Sie auf der Liste [Anzeigeschalter] die Schaltfläche  an, und wählen Sie [2 – Anzeige] aus der angezeigten Liste.  
Die Zustände der E/A Nr. 34 bis 36 werden wie bei einem Oszilloskop angezeigt.

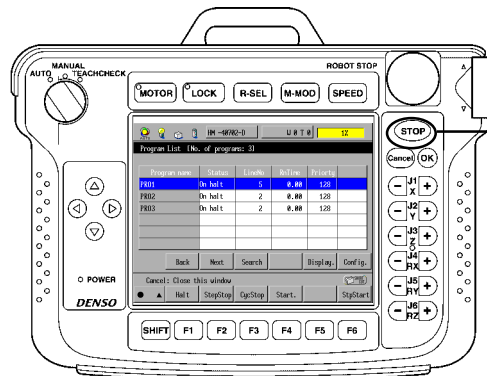
② Die Anzeigentypdarstellung ist geeignet zur gleichzeitigen Beobachtung der Zustände und Änderungen mehrerer E/A-Signale.  
Überprüfen Sie, ob jeder E/A entsprechend dem Programm ein- oder ausgeschaltet ist.

## 14.3 Sofortiger Stop

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 [STOP].  
Das Programm wird sofort unterbrochen und das Feld [Status] zeigt "Fort.Stp." an.



② Drücken Sie noch einmal [Stop].  
Das Feld [Status] zeigt "Gestoppt" an und die programmierte Operation wird unterbrochen.

# Kapitel 4

## Steuern des Roboters über Programme

In Kapitel 4 werden Sie folgendes tun:

- Steuern des Roboters über ein Programm.
- Kennenlernen der Palettierung, die zu den Hauptanwendungen bei Robotern mit 4 Achsen gehört, und der zugehörigen Programmierung.
- Lernen, wie Sie die einfachste Möglichkeit zum Entwickeln von Roboterprogrammen, die sog. PAC-Bibliothek, effektiv einsetzen können.

---

|             |                                                                     |     |
|-------------|---------------------------------------------------------------------|-----|
| ► Lektion15 | Steuern des Roboters in der Praxis                                  | 86  |
| 15.1        | Lösen der Maschinenverriegelung                                     | 86  |
| 15.2        | Einstellen der Robotergeschwindigkeit und -beschleunigung           | 87  |
| 15.3        | Einschalten des Antriebsmotors                                      | 88  |
| 15.4        | Programmierung                                                      | 89  |
| 15.5        | Starten von Programmen                                              | 92  |
| 15.6        | Wechseln der Robotergeschwindigkeit                                 | 93  |
| 15.7        | Stoppen des Roboters bei kontinuierlicher Ausführung                | 94  |
| ■           | Erstellen des Programmes [Fortsetzungsstop] und [Fortsetzungsstart] | 94  |
| ■           | Vollständiges Stoppen des Programmes ([Programmrücksetzung])        | 96  |
| ► Lektion16 | Palettierung                                                        | 98  |
|             | Was versteht man unter Palettierung?                                | 98  |
| ► Lektion17 | Effektiver Einsatz von Bibliotheken                                 | 106 |
| 17.1        | Programmbank                                                        | 107 |
| 17.2        | Importieren von Programmen                                          | 109 |
| ► Lektion18 | Beenden der Sitzung                                                 | 111 |
| 18.1        | Beenden von WINCAPSII und Herunterfahren des PC                     | 111 |
| 18.2        | Ausschalten des Roboter-Controller                                  | 111 |

---

## Lektion 15 Steuern des Roboters in der Praxis

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

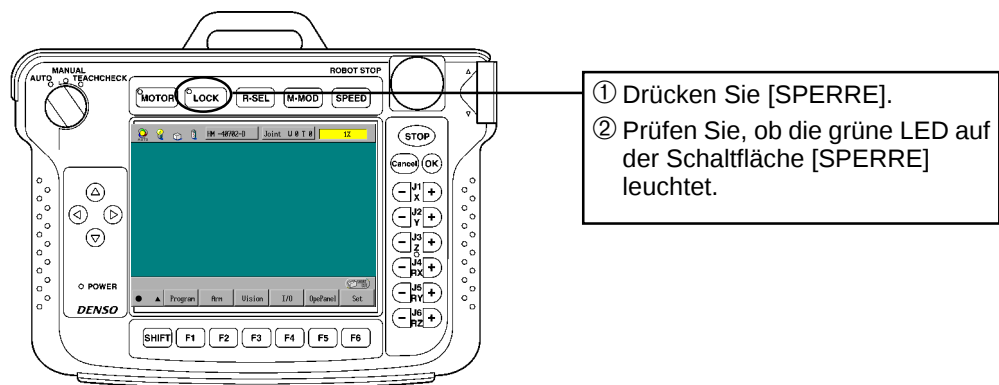
Zuerst müssen Sie eine Sicherheitsprüfung durchführen. Dann stellen Sie den Roboter auf eine niedrige Geschwindigkeit ein und prüfen die Roboterbewegung im langsamen Betrieb.

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.

Weitere Details zur Hardware-Konfiguration, siehe INSTALLATIONS- UND WARTUNGSHANDBUCH.

### 15.1 Lösen der Maschinenverriegelung

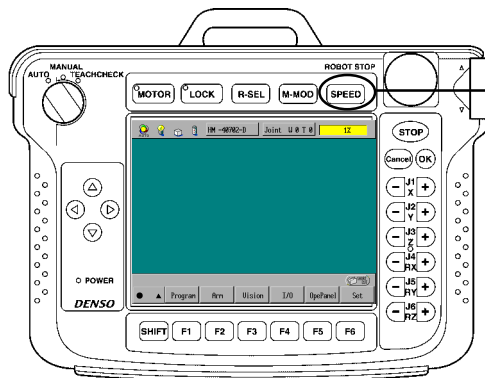
#### SCHRITT 1



Dadurch wird die Maschinenverriegelung gelöst.

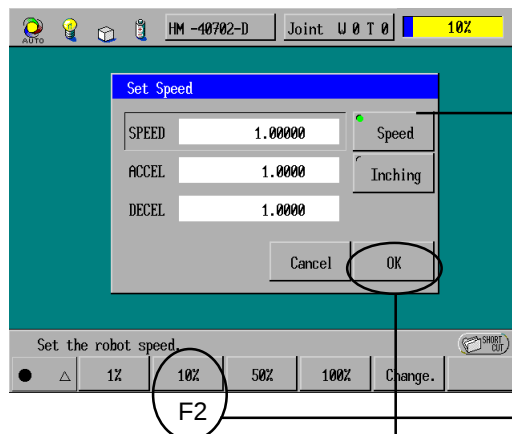
## 15.2 Einstellen der Robotergeschwindigkeit und -beschleunigung

### SCHRITT 1



① Drücken Sie [GESCHWINDIGKEIT].

### SCHRITT 2



Das Fenster [Geschwindigkeit einstellen] erscheint.

② Drücken Sie [F2 10%].

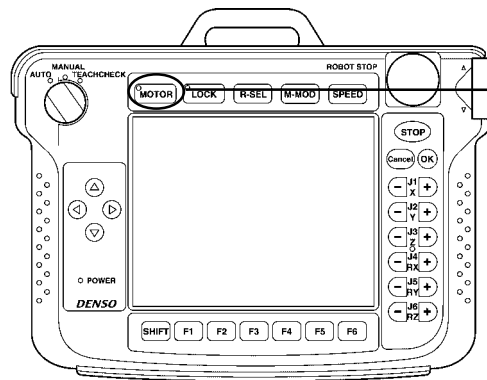
Die eingestellten Werte erscheinen in den Feldern [GESCHW], [BESCHL] und [VERZÖG].

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 Antriebmotors

### SCHRITT 1



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

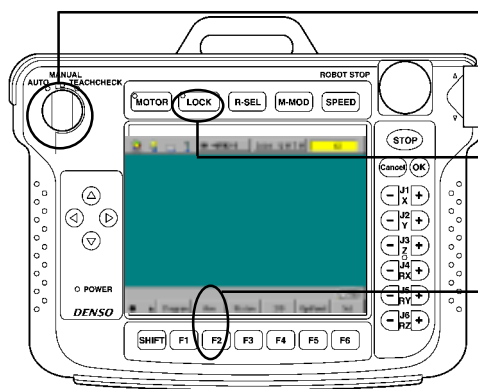
Die Motorspannung ist jetzt eingeschaltet.

## 15.4 Programmierung

Sie haben in Lektion 10 "Zuweisen der aktuellen Positionswerte zu den Positionsvariablen" Werte für die Positionsvariablen 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 den Positionsvariablen genaue Werte zuweisen.

Sie müssen praktisch die gleichen Anweisungen befolgen wie in Lektion 10 "Zuweisen der aktuellen Positionswerte zu den 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

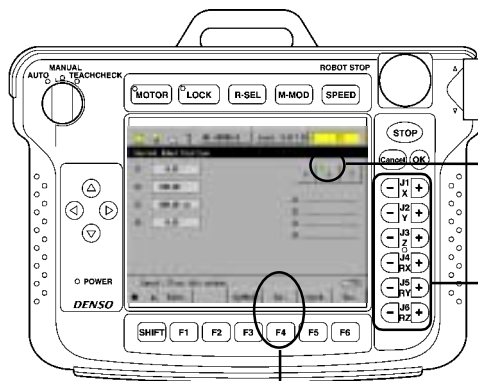


① Stellen Sie den Moduswahlschalter auf MANUELLE Position, um den Roboter in den "Manuellen Modus" zu schalten.

② Drücken Sie [SPERRE].

③ Drücken Sie in der Hauptanzeige [F2 Arm].

### SCHRITT 2 Bewegen des Roboterarmes

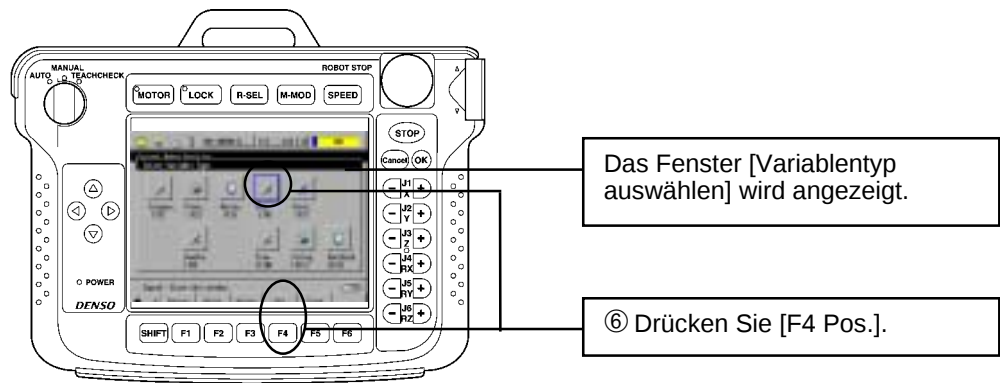


Das Fenster "Aktuelle Roboterposition" wird angezeigt. Lassen Sie die Positionsanzeige im Gelenk-Modus.

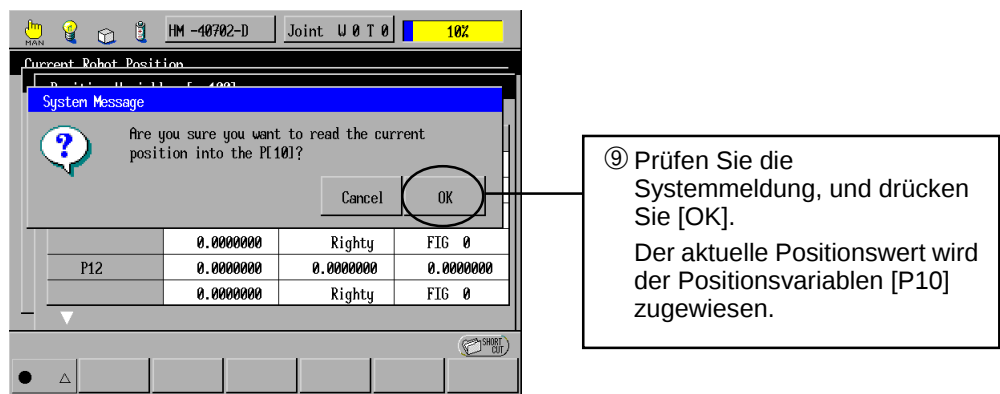
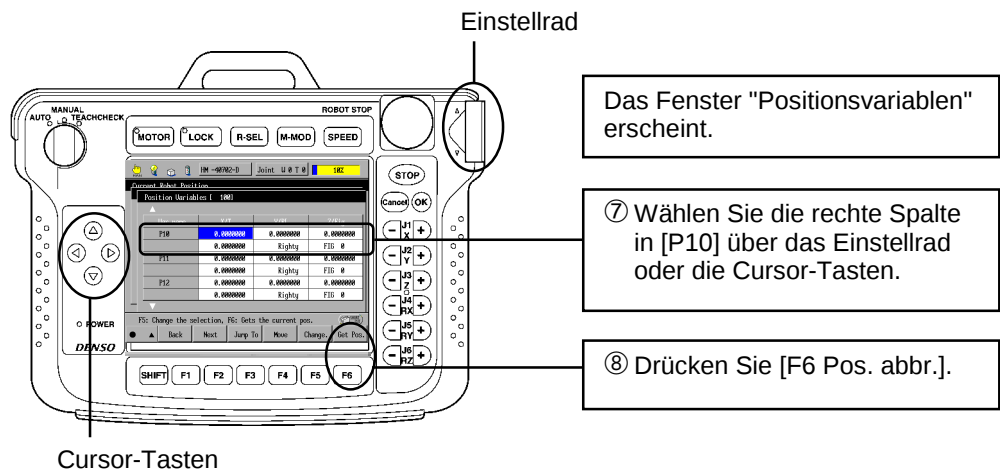
④ Bewegen Sie den Roboterarm mit den Tasten für die Armbewegung seitwärts zur Position [pHome], und beobachten Sie dabei die Roboterbewegung.

⑤ Drücken Sie [F4 Var.].

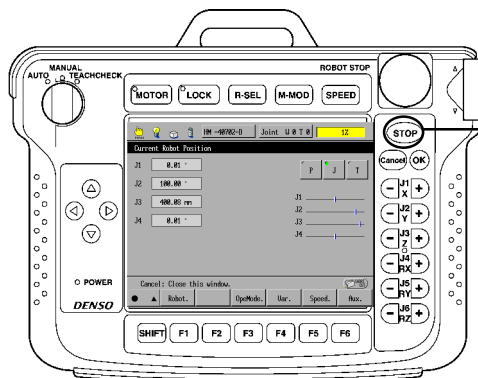
SCHRITT 3     Anzeigen der Positionsvariablen



SCHRITT 4     Zuweisen der aktuellen Roboterposition zur Positionsvariablen



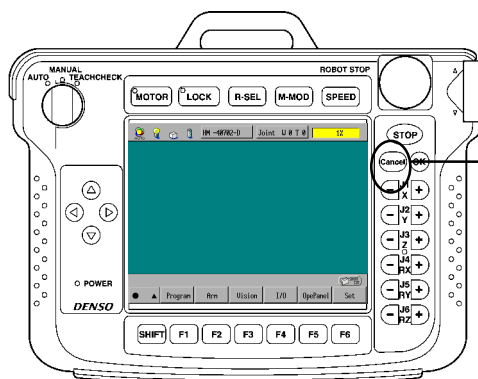
## SCHRITT 5 Zuweisen sonstiger Positionswerte zu Positionsvariablen



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

⑪ Wiederholen Sie die Schritte 2 bis 4, um für jeweils drei andere Variablen [pPlace1], [pPlace2] und [P13] die Positionsdaten [P11], [P12] und [P13] zuzuweisen.

## Schritt 6 Zurück zur Hauptanzeige



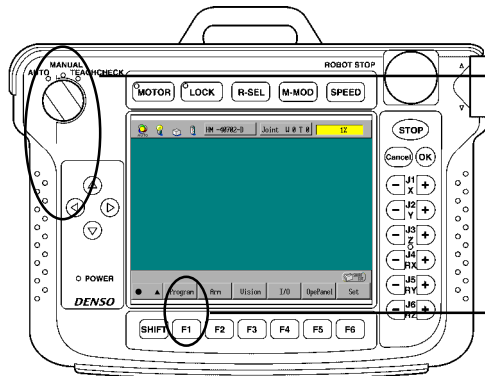
⑫ Drücken Sie [Abbrechen], um nach Beenden aller oben genannten Prozeduren zur Hauptanzeige zurückzukehren.

## 15.5 Starten von Programmen

Sie werden den Roboter jetzt tatsächlich über ein Programm steuern.

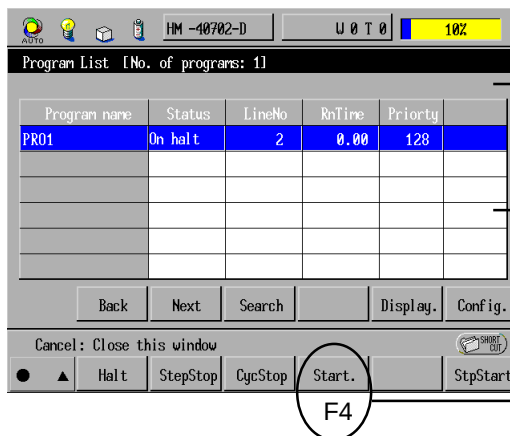
**Achtung:** Halten Sie stets eine Hand frei und bereit, die STOP-Taste zu drücken.

### SCHRITT 1 Auswählen des Auto-Modus zum Anzeigen des Fensters "Programmliste"



① Stellen Sie den Moduswahl-schalter auf AUTO-Position, um den Roboter in den "Auto-Modus" zu schalten.

② Drücken Sie [F1 Programm].

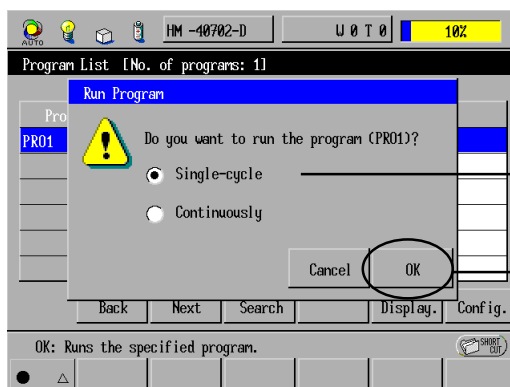


Das Fenster [Programmliste] wird angezeigt.

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.

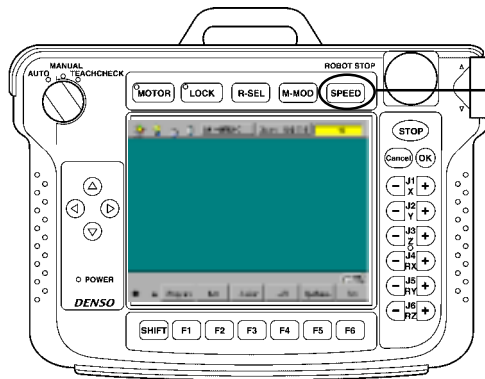
**SCHRITT 3** Wenn Sie [Kontinuierlich] anstatt [Einzelzyklus] auswählen, startet der Roboter im kontinuierlichen Modus.

Wenn der Betrieb im Modus "Einzelzyklus" funktioniert, probieren Sie den Modus "Kontinuierlicher Start".

## 15.6 Wechseln der Robotergeschwindigkeit

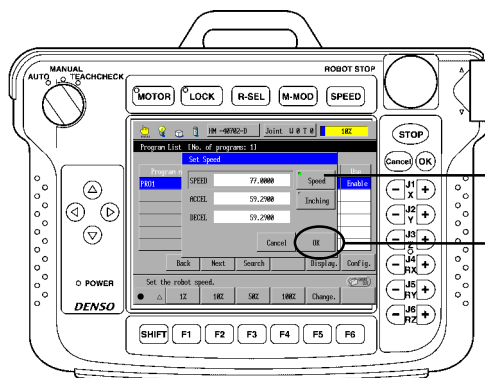
Lassen Sie den Roboter schneller laufen, indem Sie langsam die Geschwindigkeitseinstellungen ändern, wenn sowohl Einzelzyklus- als auch kontinuierliche Ausführung funktioniert haben.

### SCHRITT 1



① Drücken Sie [GESCHWINDIGKEIT].

### SCHRITT 2



Das Fenster [Geschwindigkeit einstellen] erscheint.

② Drehen Sie das Einstellrad, um die gewünschte Geschwindigkeit auszuwählen.

Wenn in der Spalte [Geschwindigkeit] ein Wert eingestellt ist, werden in den Spalten [BESCHL] und [VERZÖG] automatisch

③ Drücken Sie [OK].

Robotergeschwindigkeit, Beschleunigung und Verzögerung sind jetzt eingestellt.

## 15.7 Stoppen des Roboters bei kontinuierlicher Ausführung

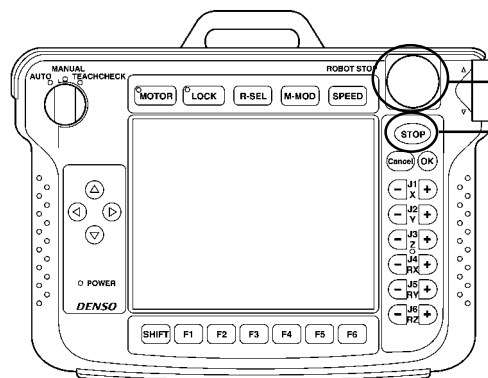
Wenn Sie den Roboter bei kontinuierlicher Ausführung stoppen, indem Sie die Schaltfläche [Stop] drücken, wechselt der Roboter in den Status [Fortsetzungsstop]. In diesem Status können Sie das Programm von der Stelle, an der es gestoppt wurde, neu starten.

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

**Vorsicht:** Im Notfall drücken Sie die Roboter-Stoptaste.

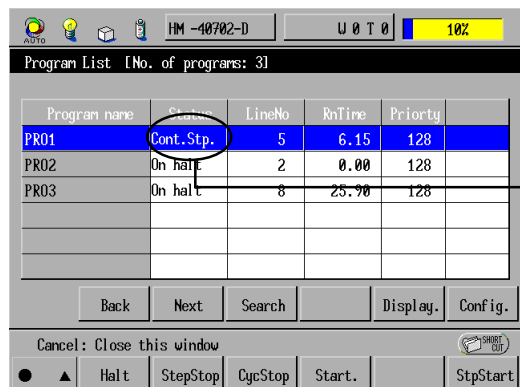
### ■ Erstellen des Programmes [Fortsetzungsstop] und [Fortsetzungsstart]

#### SCHRITT 1 Erstellen des Task-Programmes [Fortsetzungsstop]



Roboter-Stoptaste

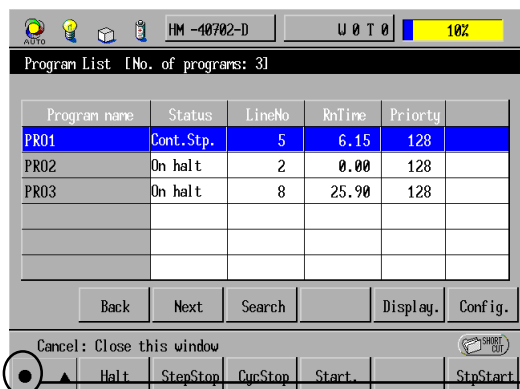
① Drücken Sie [STOP].



Das Programm stoppt sofort und [Cont. Stp.] wird in der Spalte [Status] angezeigt.

| Program name | Status    | LineNo | RnTime | Priority |
|--------------|-----------|--------|--------|----------|
| PR01         | Cont.Stp. | 5      | 6.15   | 128      |
| PR02         | On halt   | 2      | 0.00   | 128      |
| PR03         | On halt   | 8      | 25.90  | 128      |

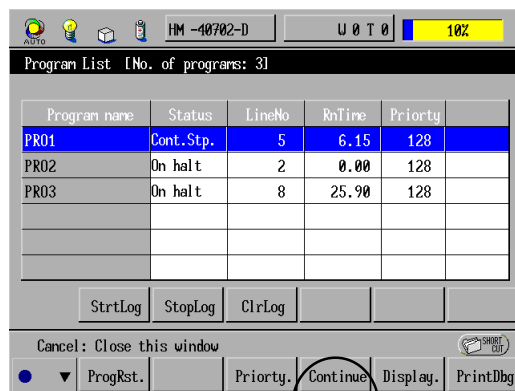
#### SCHRITT 2 Erstellen des Programmes [Fortsetzungsstart]



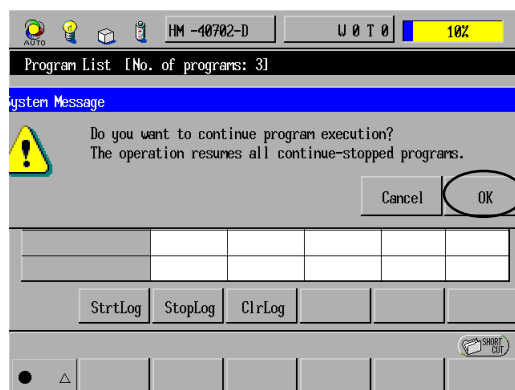
② Drücken Sie die [Umschalttaste].

UMSCHALTTASTE

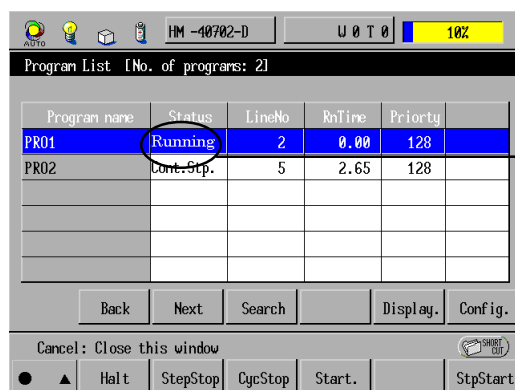
| Program name | Status    | LineNo | RnTime | Priority |
|--------------|-----------|--------|--------|----------|
| PR01         | Cont.Stp. | 5      | 6.15   | 128      |
| PR02         | On halt   | 2      | 0.00   | 128      |
| PR03         | On halt   | 8      | 25.90  | 128      |



③ Drücken Sie [F10 Weiter].



④ Drücken Sie [OK].

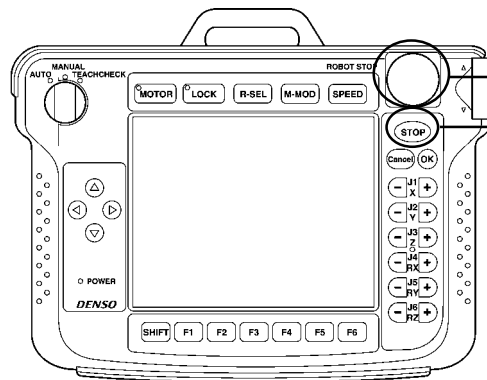


Das Programm wird neu gestartet und in der Spalte [Status] wird [Aktiv] angezeigt.

Jetzt können Sie das Programm neu starten.

## ■ Vollständiges Stoppen des Programmes [Programmrücksetzung]

### SCHRITT 1 Erstellen des Programmes [Fortsetzungsstop]



Roboter-Stoptaste

① Drücken Sie [STOP].

The screenshot shows a 'Program List' window with the following data:

| Program name | Status    | LineNo | RnTime | Priority |
|--------------|-----------|--------|--------|----------|
| PR01         | Cont.Stp. | 5      | 6.15   | 128      |
| PR02         | On halt   | 2      | 0.00   | 128      |
| PR03         | On halt   | 8      | 25.90  | 128      |

The 'Status' column for PR01 is circled, with a line pointing to it from the text 'Das Programm stoppt sofort und [Cont. Stp.] wird in der Spalte [Status] angezeigt.'

Das Programm stoppt sofort und [Cont. Stp.] wird in der Spalte [Status] angezeigt.

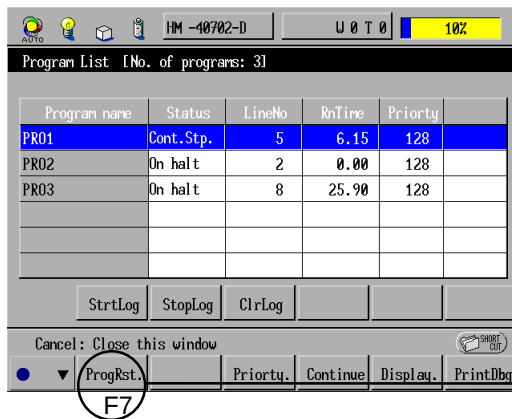
### SCHRITT 2 Rücksetzen des Programmes ([Programmrücksetzung])

The screenshot shows the same 'Program List' window as in Step 1. At the bottom of the screen, there is a row of buttons: 'Halt', 'StepStop', 'CycStop', 'Start.', and 'StpStart'. The 'StepStop' button is highlighted with a circle, and a line points from it to the text '② Drücken Sie die [Umschalttaste].'

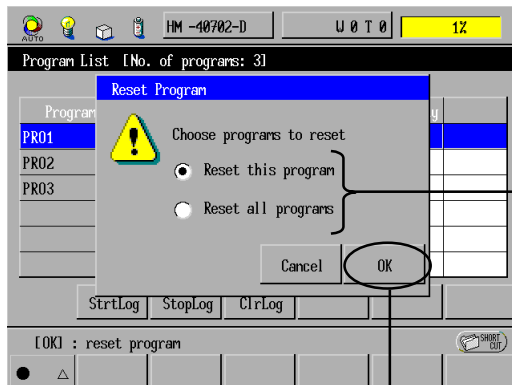
② Drücken Sie die [Umschalttaste].

UMSCHALTASTE



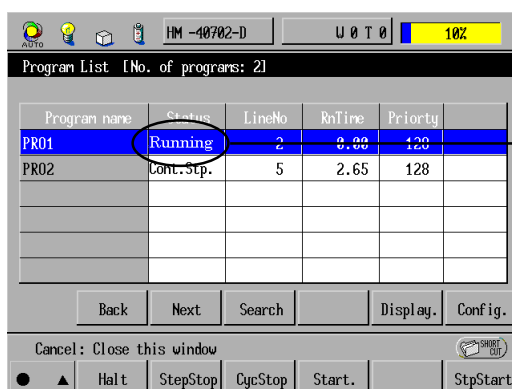


③ Drücken Sie [F7 ProgRst.].



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

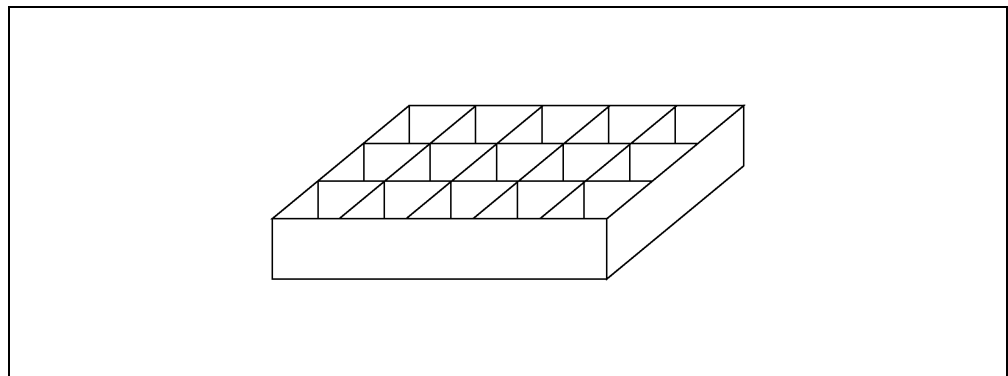
---

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

Die Palettierungsprogramme aktualisieren die Informationen zur Unterteilung während jede Position aufgerufen wird, damit der Roboter weiß, an welcher Position das nächste Teil abgelegt oder von wo es entnommen werden muß.



**Abbildung 16-1 Partitionierte Palette**

## Palettierungsprogramm "PRO3"

Die übliche Prozedur zum Entwickeln eines Programmes unter Einbeziehung der Bibliothek steht im Programm "PRO3" mit dem Titel "Palettierungsvorlage" zur Verfügung, obwohl es je nach Anwendungen und Einsatzzweck eine Vielzahl unterschiedlicher Programme zur Palettierung gibt.

Setzen Sie diese "Palettierungsvorlage" in effektiver Weise als Vorlage ein, indem Sie die für Ihre Anwendungen erforderlichen Elemente hinzufügen/löschen.

Im folgenden ist ein Mustervorlagen-Programm mit der Bezeichnung "PRO3" aufgeführt.

Folgendes ist bei dieser Mustervorlage festgelegt:

- die Palettierungspunkte müssen bei P50 bis P55 liegen und
- das zu entwickelnde Programm sollte den Roboter so steuern, daß er zur Arbeitsentnahmeposition P50 bewegt wird, das Palettierungsprogramm "0" ausführt, 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 zur Palettierung erforderlichen Parameter sind in den Abbildungen 16-3, 16-4, 16-5 und der Tabelle 16-1 dargestellt.

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

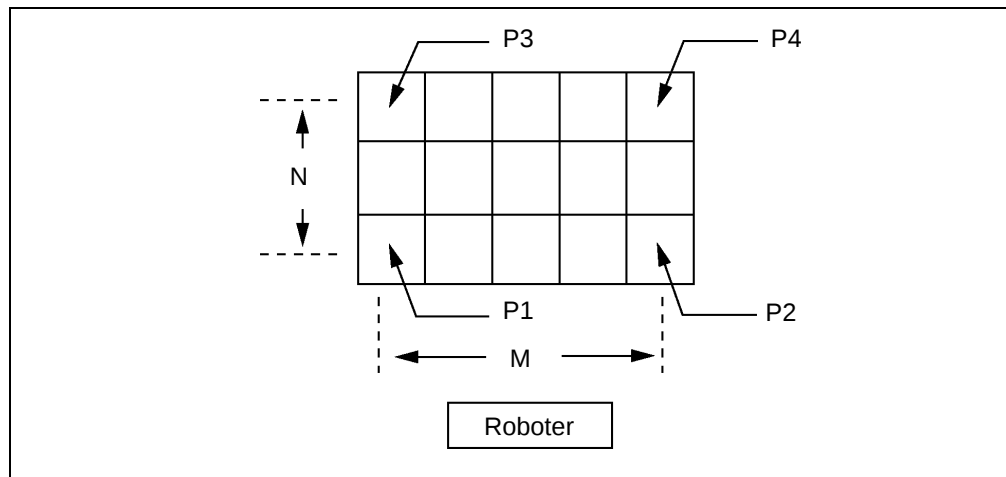


Abbildung 16-3 Palettendraufsicht

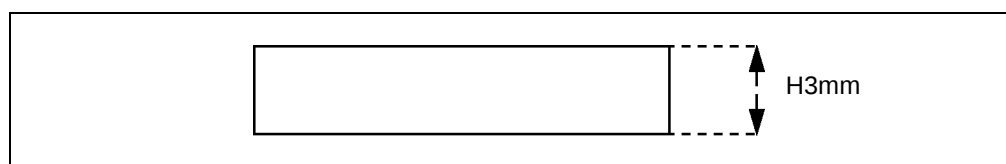


Abbildung 16-4 Seitenansicht der Palette

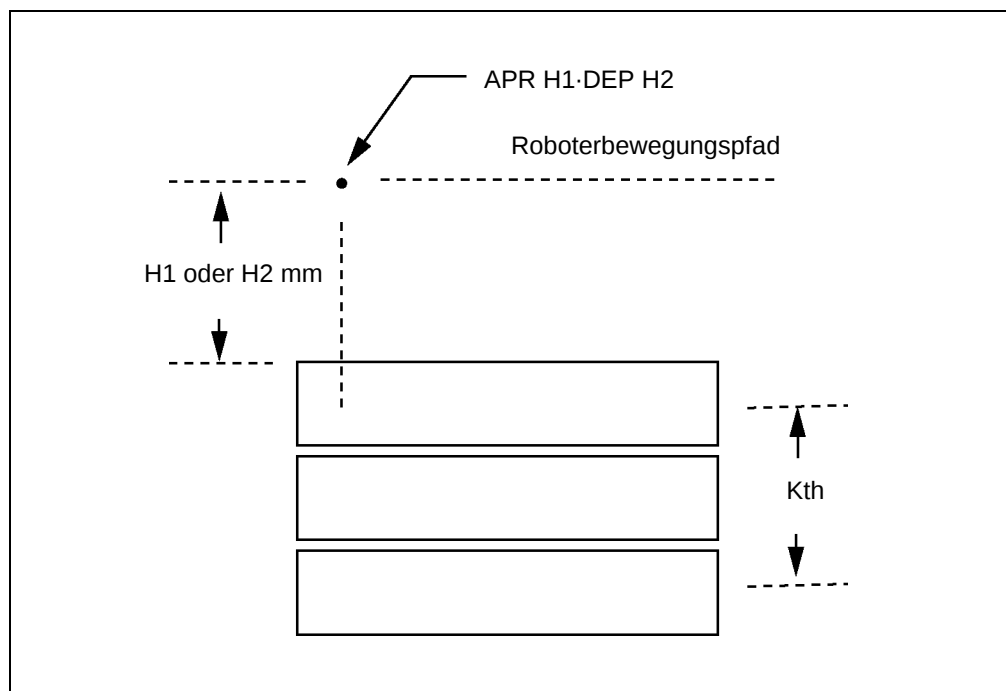


Abbildung 16-5 Gestapelte Paletten

**Tabelle 16-1 Für die Palettierung erforderliche Parameter**

| Symbol               | Name                                                                                                                                                                                                                                                                                      | Beschreibung                                                            | Einheit                       |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------|
|                      | Palettierungsnummer                                                                                                                                                                                                                                                                       | Palettierungsindex                                                      | Keine (Ganzzahl)              |
| N                    | Anzahl der Reihenteile                                                                                                                                                                                                                                                                    | Anzahl der Unterteilungen von P1 zu P3                                  | Zählung (Ganzzahl)            |
| M                    | Anzahl der Spaltenteile                                                                                                                                                                                                                                                                   | Anzahl der Unterteilungen von P1 zu P2                                  | Zählung (Ganzzahl)            |
| K                    | Anzahl der gestapelten Paletten                                                                                                                                                                                                                                                           | Anzahl der gestapelten Paletten                                         | Zählung (Ganzzahl)            |
| H1                   | Annäherungsabstand                                                                                                                                                                                                                                                                        | Annäherungsabstand, mit dem sich der Roboter einer Palette annähert     | mm (einfache Genauigkeit FPT) |
| H2                   | Abweichungsabstand                                                                                                                                                                                                                                                                        | Abweichungsabstand, mit dem sich der Roboter von einer Palette entfernt | mm (einfache Genauigkeit FPT) |
| H3                   | Höhe einer Palette                                                                                                                                                                                                                                                                        | Höhe einer Palette                                                      | mm (einfache Genauigkeit FPT) |
|                      | Wo H1 und H2 die unten genannten Bedingungen erfüllen.<br>$H1 > \{H3 \times K - 1\} + 5$<br>$H2 > \{H3 \times K - 1\} + 5$                                                                                                                                                                |                                                                         |                               |
| P1<br>P2<br>P3<br>P4 | Positionen der 4 Ecken der Palette wie in Abbildung 16-3 dargestellt.<br>Es ist nicht möglich, die relative Positionierung einer der Ecken auszutauschen.<br>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, auf dem sich der Roboter einer Palette annähert.  
 Ein Programm wendet die einfache Annäherungspfadlänge bei jedem Aufruf des gleichen Palettierungsprogrammes an.

H2 Abweichungspfadabstand  
 Gibt die Länge des Abweichungspfads an, auf dem sich der Roboter von einer Palette entfernt.  
 Ein Programm wendet die einfache Abweichungspfadlänge bei jedem Aufruf des gleichen Palettierungsprogrammes an.

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.

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

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

Ist dies nicht der Fall, 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.

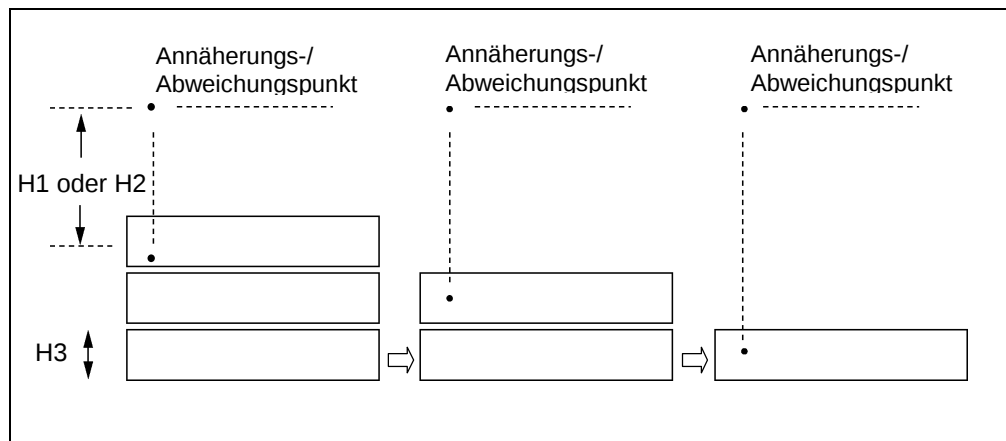


Abbildung 16-6 Beziehung zwischen Stapelhöhe und Annäherungs-/Abweichungspunkten

#### 4 Eckpunkte P1, P2, P3 und P4

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

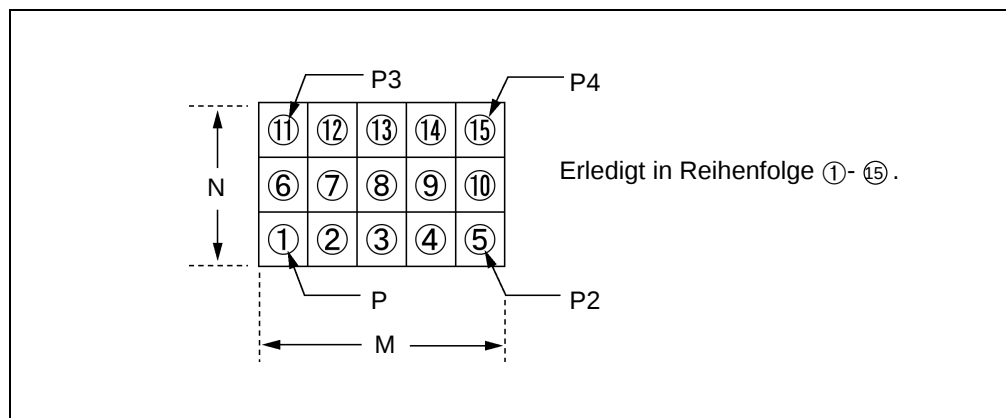


Abbildung 16-7 Palettierungsreihenfolge

## Einstellen der Palettierungsparameter

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

Wenn Sie bei der Erstellung eines "Neuen Systemprojektes" im Dialogfeld [Neues Projekt] unter [Gerätetyp] [1-Palettierung] auswählen, 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 Reihenunterteilungen (3)
  3. Anzahl der Spaltenunterteilungen (5)
  4. Anzahl der gestapelten Paletten (3)
  5. Annäherungsabstand (50 mm)
  6. Abweichungsabstand (50 mm)
  7. Palettenhöhe (10 mm)
  8. P1 Position (P52)
  9. P2 Position (P53)
  10. P3 Position (P54)
  11. P4 Position (P55)
- } Geben Sie nur die Indexe der Positionsvariablen an.

**Abbildung 16-9 Parameter von "pltInitialize"**

Beschreibungen zu den oben genannten Parametern finden Sie auch 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

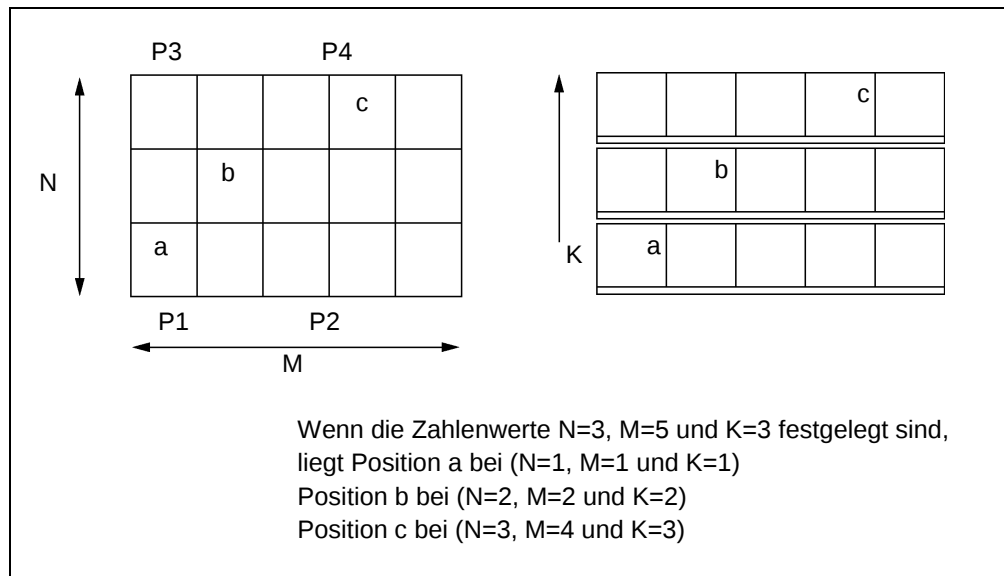
Der Palettierungszähler addiert zum Gesamtzähler 1 hinzu, jedesmal wenn "pltMove" ausgeführt wird, und paßt die Zahlenwerte entsprechend den anderen an, damit die nächste Palettierungsposition sichergestellt ist.

Wenn 1 zum Gesamtzähler addiert wird, bewegt sich die durch den Spaltenzähler (M) angezeigte Position der Palettenspalte zur nächsten Spalte. Wenn die durch den Spaltenzähler (M) angezeigte Position der Palettenspalte das Ende bzw. den maximalen Zahlenwert erreicht hat, dann zählt der Reihenzähler (N) um 1 weiter und der Spaltenzähler (M) wird zu seinem Minimalwert. Wenn die Position des Reihenzählers (N) das Ende der Reihenunterteilung der Palette erreicht hat und zum Maximalwert wird, zählt der gestapelte Reihenzähler (K) um 1 weiter und der Reihenzähler (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.

**Vorsicht:** Wenn Sie ein neues Task-Programm kompilieren und sein Laufzeitmodul 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)
```

**Vorsicht:** 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 Flag-Status "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:

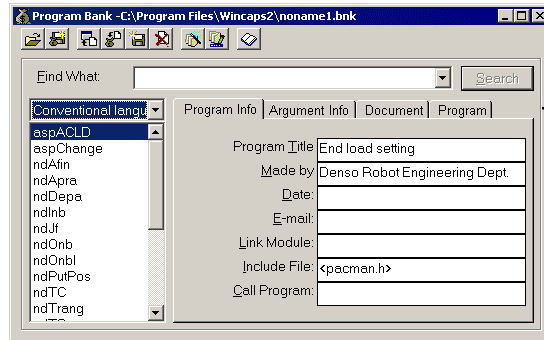
Sie können bereits für andere Programmprojekte erstellte Programme in Ihrem neuen Projekt registrieren. In diesem Fall kann das registrierte Projekt mit anderen Programmen genutzt werden.

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

## 17.1 Programmbank

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

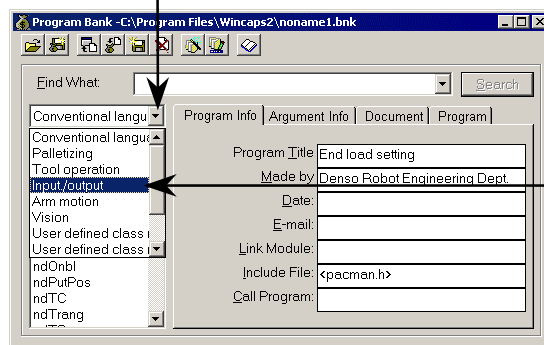
### SCHRITT 1 Durchsuchen der Programmbank



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

Das Fenster [Programmbank] wird angezeigt.

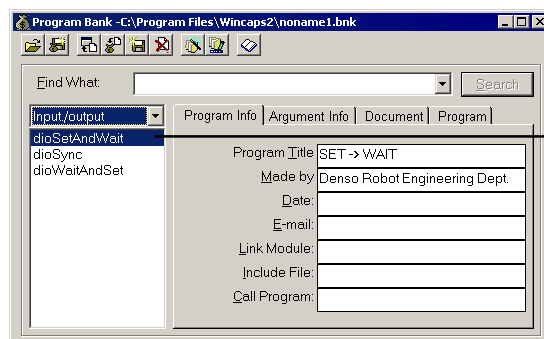
### SCHRITT 2 Auswählen der Klasse



② Klicken Sie das Klassenauswahlfeld an.  
Das Klassenauswahl-Pulldown-Menü wird angezeigt.

③ Wählen Sie [Eingabe/Ausgabe].

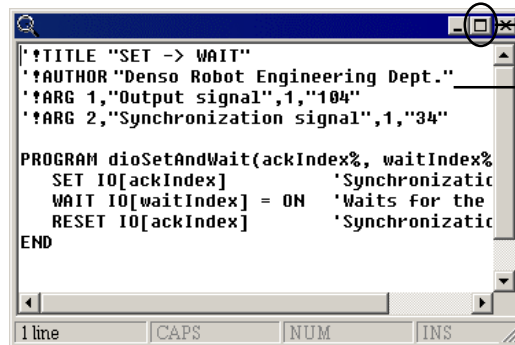
### SCHRITT 3 Auswählen des hinzuzufügenden Programmes



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]      'Synchronizatio
  WAIT IO[waitIndex] = ON 'Waits for the
  RESET IO[ackIndex]    'Synchronizatio
END
```

⑤ Klicken Sie [Zu Projekt hinzufügen] an.

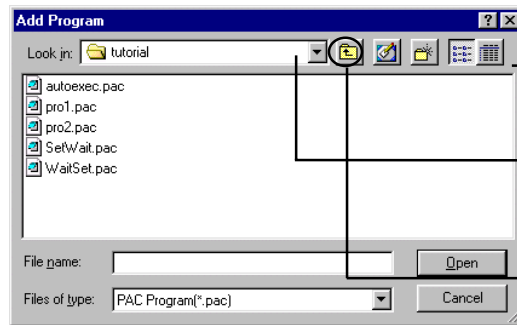
Die Option [dioSetAndWait] wird im neuen Projekt registriert und das Fenster [Bearbeiten] erscheint.

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



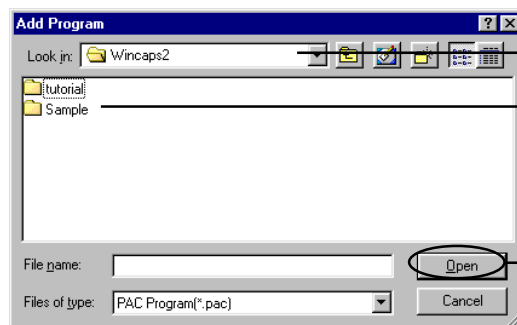
① Wählen Sie aus dem Menü "Extras" im Fenster [PAC-Manager] die Option [Importieren].

Das Dialogfenster [Programm hinzufügen] wird angezeigt.

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

② Klicken Sie die Schaltfläche [Vorherig] an .

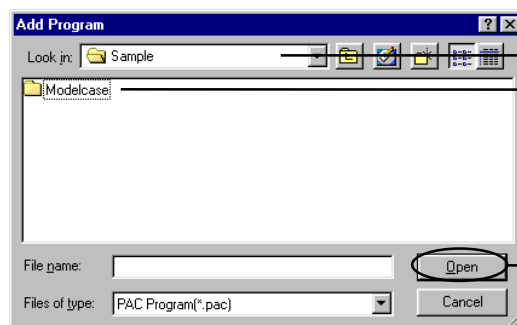
### SCHRITT 2



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

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

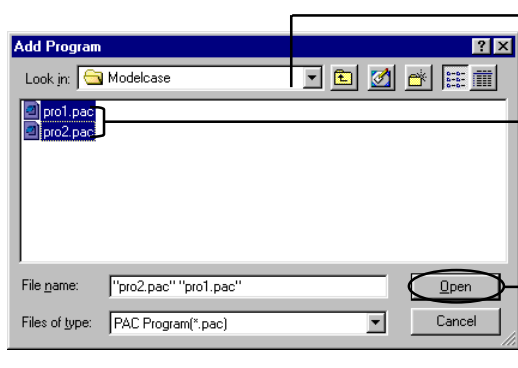
### SCHRITT 3



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

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

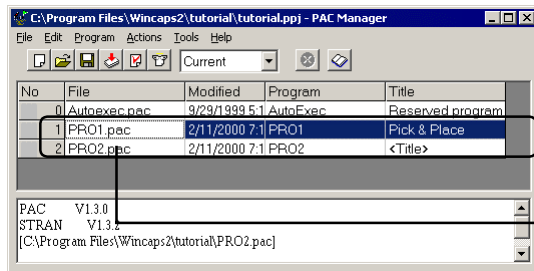
## SCHRITT 4



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



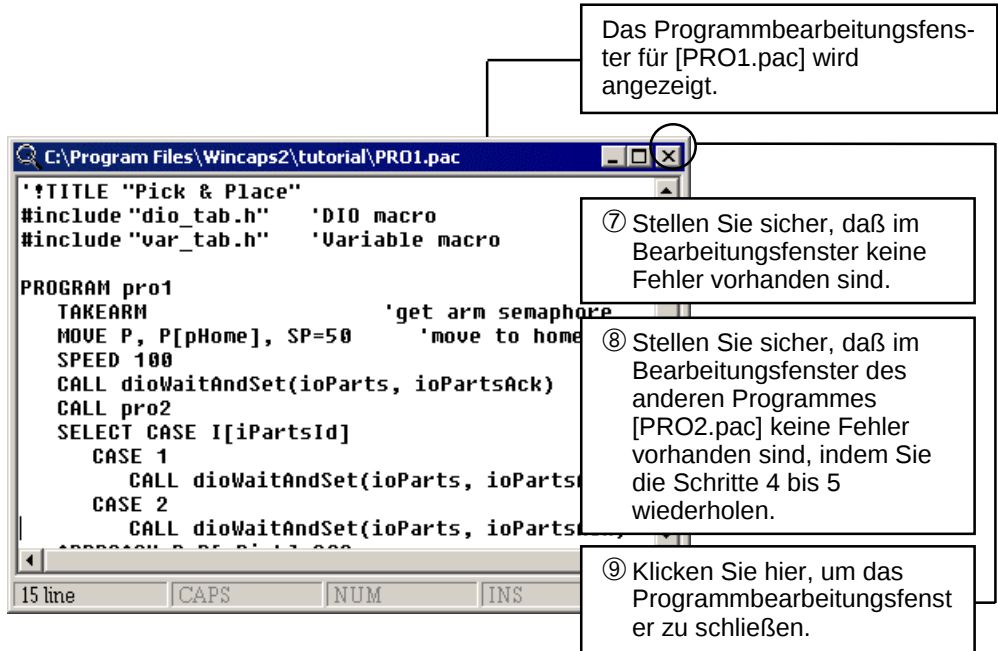
Die beiden Programme wurden zur Liste im Fenster [PAC-Manager] neu hinzugefügt.

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 Programmbearbeitungsfenster 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.

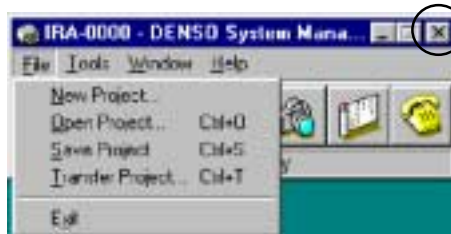
## Lektion 18 Beenden der Sitzung

---

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

### 18.1 Beenden von WINCAPSII und Herunterfahren des PC

#### SCHRITT 1



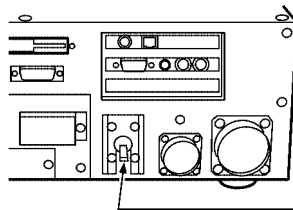
① Klicken Sie in der Ecke rechts oben des Fensters [DENSO System Manager] das Feld "Schließen" an.  
Jede WINCAPSII-Software auf dem PC wird geschlossen.

#### SCHRITT 2

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

### 18.2 Ausschalten des Roboter-Controller

#### SCHRITT 1



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



# Kapitel 5

## Funktionen des Denso Roboters

---

In Kapitel 5 werden Sie folgendes tun:

Die nützlichen Funktionen des Denso Roboters kennen lernen, wie z. B. Strombegrenzung und direkte Programmierung. Diese Funktionen dienen zur Senkung der Installationskosten und zur Verbesserung der Arbeitseffizienz in Vorbereitung und Praxis.

---

|                                           |     |
|-------------------------------------------|-----|
| ► Lektion 19 Strombegrenzung              | 114 |
| Strombegrenzwertbefehle                   | 115 |
| ► Lektion 20 Direkte Programmierung       | 116 |
| 20.1 Auswählen des direkten Modus         | 116 |
| 20.2 Programmieren der Stellungsvariablen | 119 |
| ► Lektion 21 Sonstige Funktionen          | 121 |

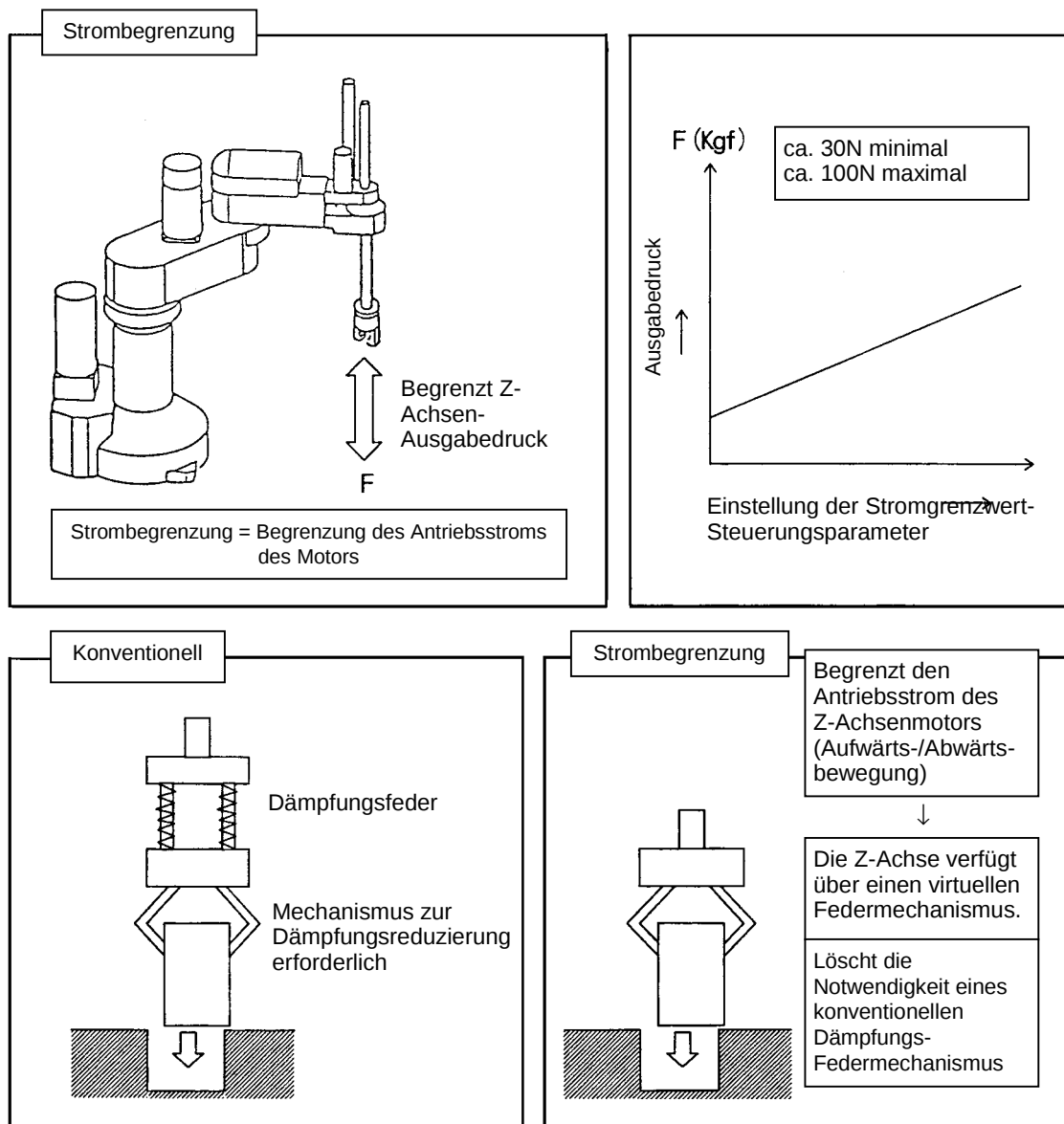
---

## Lektion 19 Strombegrenzung

Durch die Strombegrenzung wird der normale Betriebsstrom des Servomotors auf einen Wert unterhalb des eingestellten Wertes begrenzt. Da das Drehmoment des Servomotors proportional zum geflossenen Strom ist, kann sein Drehmoment auch durch die Begrenzung des Motorstroms beschränkt werden.

Der Ausgabedruck von Robotern kann auf einer direkten Achse wie der Z-Achse (vertikale Achse) des horizontal ausgerichteten Roboters beschränkt werden. Das Ein-/Ausschalten oder der eingestellte Stromgrenzwert kann durch das Programm flexibel festgelegt werden.

Wenn der Stromgrenzwert auf die vertikale Achse (Z-Achse) angewendet wird, kann wie auf folgenden Abbildungen dargestellt eine Beschädigung des Werkstückes durch Einsatz- und Überstromfehler vermieden werden, ohne daß der konventionelle Dämpfungsfedermechanismus verwendet wird.



## Stromgrenzwertbefehle

Die Stromgrenzwertbefehle sind in den PAC-Bibliotheken integriert und in zwei Bibliotheken gruppiert: Stromgrenzwert-Funktionsbibliothek und Bibliothek zur Einstellung des Abweichungstoleranzwertes. Ausführliche Hinweise finden Sie im "Programmierungshandbuch".

Im folgenden sind diese Bibliotheken und einige Beispiele für Anwendungsprogramme dargestellt.

### ■ Stromgrenzwert-Funktionsbibliothek

|             |                                                                                                       |
|-------------|-------------------------------------------------------------------------------------------------------|
| SetCurLmt   | Festlegen des Stromgrenzwertes                                                                        |
| ResetCurLmt | Zurücksetzen des Stromgrenzwertes                                                                     |
| SetForce_HM | Festlegen des Stromgrenzwertes zur<br>Ausgabedrehmoment-Festlegung für Z-Achsen von<br>HM/HS-Robotern |
| SetForce_HC | Festlegen des Stromgrenzwertes zur<br>Ausgabedrehmoment-Festlegung von HC-Robotern                    |

### ■ Bibliothek zur Einstellung des Abweichungstoleranzwertes

|            |                                            |
|------------|--------------------------------------------|
| SetEralw   | Festlegen des Abweichungstoleranzwertes    |
| ResetEralw | Zurücksetzen des Abweichungstoleranzwertes |

### Programmbeispiele zur Einstellung des Stromgrenzwertes

|                            |                                                                          |
|----------------------------|--------------------------------------------------------------------------|
| TAKEARM                    | Erfassen des Armsemaphors                                                |
| CALL SetEralw (3, 50)      | Festlegen des<br>Abweichungstoleranzwertes der Z (3)-<br>Achse auf 50 mm |
| CALL SetCurLmt (3, 30)     | Festlegen des Stromgrenzwertes der<br>Z(3)-Achse auf 30 %                |
| (CALL SetForce_HM (50, 0)) | Festlegen der Kraft an der Z-Achse des<br>HM-Roboters auf 50N            |

### Programmbeispiele zur Neueinstellung des Stromgrenzwertes

|                      |                                                                                                |
|----------------------|------------------------------------------------------------------------------------------------|
| TAKEARM              | Erfassen des Roboterarmsemaphors                                                               |
| CALL ResetEralw (3)  | Zurücksetzen des<br>Abweichungstoleranzwertes der Z(3)-<br>Achse auf den Anfangswert           |
| CALL ResetCurLmt (3) | Zurücksetzen des Stromgrenzwertes<br>der Z(3)-Achse (die Abweichung ist auch<br>ausgeschaltet) |

**Anmerkung:** Denken Sie immer daran, daß der eingestellte Stromgrenzwert der RoboterAusgabedefinition in den oben genannten Beispielen nur als Referenz dient und kein garantierter Wert ist.

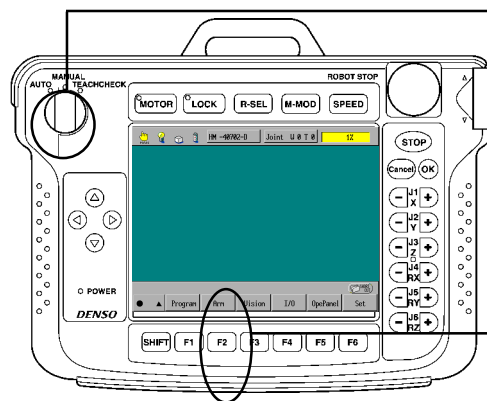
## Lektion 20 Direkte Programmierung

Direkte Programmierung bedeutet, den Roboterarm *bei ausgeschaltetem Motor* direkt mit der Hand zu bewegen (ohne Einsatz des Programmiergerätes) und die aktuelle Position auf eine Gelenk-, Positionsvariable oder eine homogene Transformationsvariable zu programmieren.

### 20.1 Auswählen des direkten Modus

Führen Sie vorher die Kalibrierung (CAL) durch, und fahren Sie mit folgender Prozedur fort.

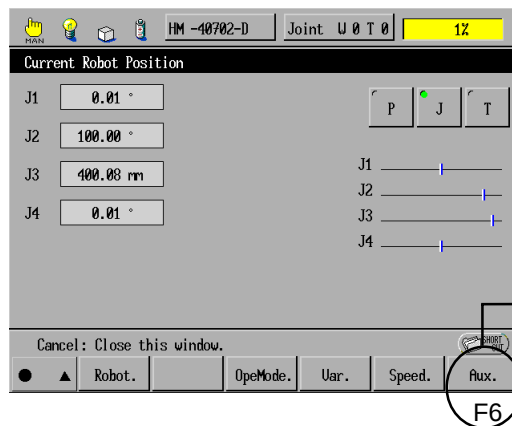
#### SCHRITT 1



① Stellen Sie den Moduswahl-schalter auf [MANUELL], um in den "Manuellen Modus" zu schalten.

② Drücken Sie [F2 Arm] in der Hauptanzeige.

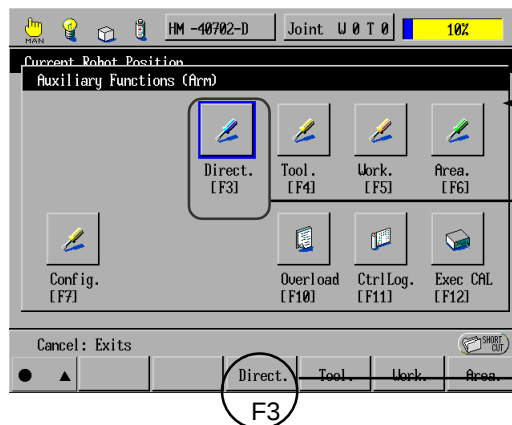
#### SCHRITT 2



Das Fenster [Aktuelle Roboterposition] erscheint.

③ Drücken Sie [F6 Hilfsfunktionen].

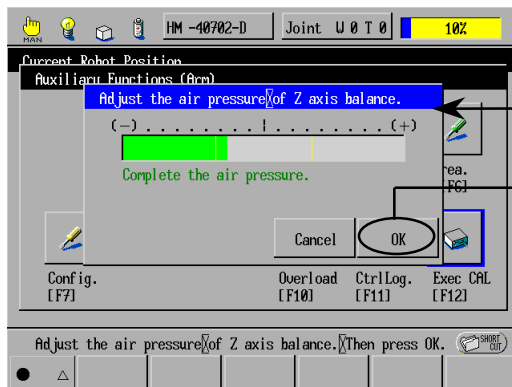
#### SCHRITT 3



Das Fenster [Hilfsfunktionen (Arm)] erscheint.

④ Drücken Sie [F3 Direkt].

## SCHRITT 4



Das Fenster [Luftdruck des Z-Achsen-Ausgleichs einstellen] erscheint.

- ⑤ Stellen Sie den Luftdruck entsprechend den auf dem Bildschirm angezeigten Anleitungen ein.  
Wenn "Luftdruck abschließen" links angezeigt wird, drücken Sie die Schaltfläche "OK".

## ★PUNKT★

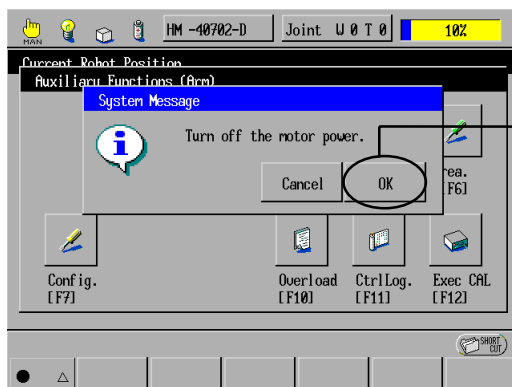
Die Luftdruckeinstellung des Z-Achsen-Ausgleichs ist nur erforderlich, wenn Sie den Roboter nach Einschalten des Roboter-Controller erstmals in den direkten Programmiermodus setzen.

## SCHRITT 5



- ⑥ Drücken Sie [OK], um die Luft-Ausgleichseinstellung der Z-Achse zu beenden.

## SCHRITT 6



- ⑦ Schalten Sie den Motor aus, und dann drücken Sie [OK].

## SCHRITT 7



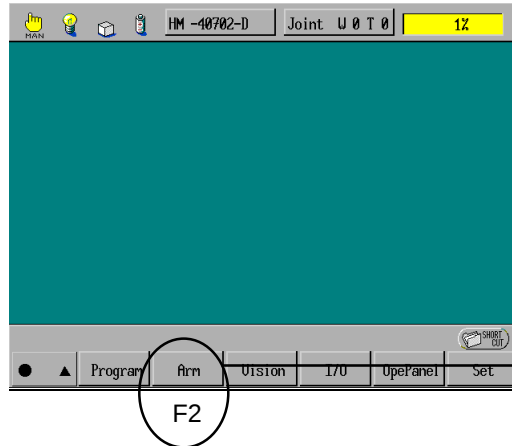
⑧ Prüfen Sie die Meldung. Wenn sie akzeptabel ist, dann drücken Sie [OK].

Der Roboter wird in den direkten Programmiermodus gesetzt.

## 20.2 Programmieren der Stellungsvariablen

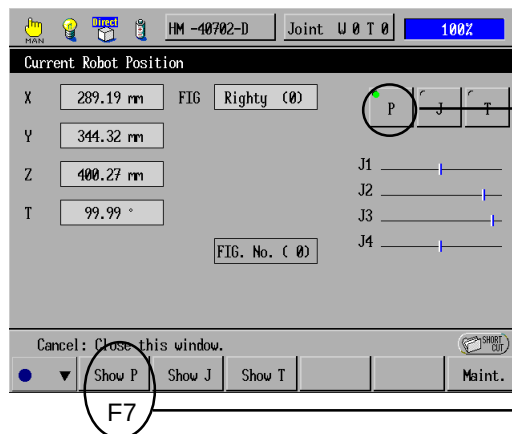
Sie können bei gelöster Z-Achsen-Bremse den Roboter einfach von Hand bewegen, wenn er sich im direkten Modus befindet. Bewegen Sie das obere Ende des Roboters von Hand zur gewünschten Position und programmieren diese ein.

### SCHRITT 1



① Drücken Sie [F2 Arm] in der Hauptanzeige.

### SCHRITT 2

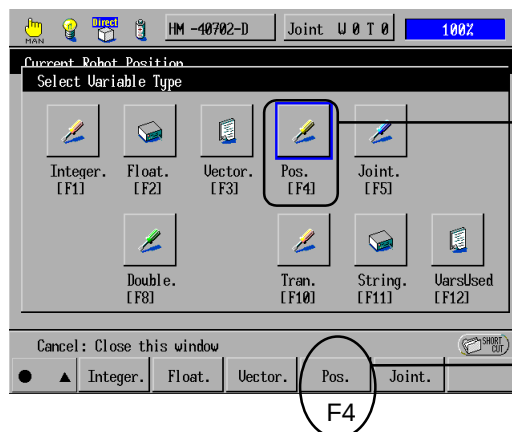


② Drücken Sie [F7 Anzeige], um den Positionstyp anzuzeigen. Dies kann auch ausgeführt werden, indem Sie die [UMSCHALTASTE] drücken, um [F7 Anzeige P] auf der F1-Tastenposition anzuzeigen, und dann F7 drücken.

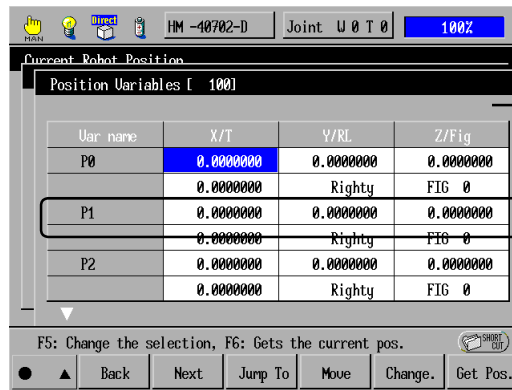
### SCHRITT 3

③ Bewegen Sie das obere Ende des Roboterarmes von Hand zur gewünschten Position.

### SCHRITT 4



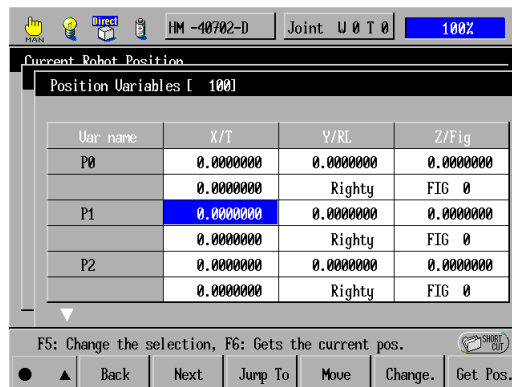
④ Drücken Sie F4 [Pos.] (Oder drücken Sie im Fenster [Variablentyp auswählen] [P Position].)



Das Fenster [Positionsvariablen] erscheint.

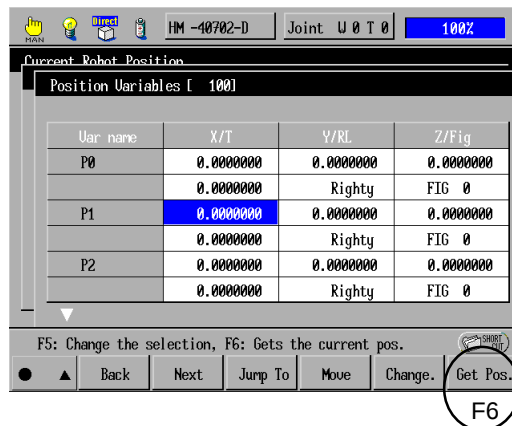
⑤ Wählen Sie die Zeile der Positionsvariablennummer, die Sie programmieren möchten. Wählen Sie jetzt Variable P1. Wählen Sie im Fenster [Positionsvariablen] die Zeile für Variable 1.

## SCHRITT 5



Im Fenster [Positionsvariablen] werden sechs Datenwerte für jede Variable angezeigt. Wenn Sie einen der sechs Datenwerte markieren, dann bedeutet dies, daß der Variablenname (P1 bei diesem Beispiel) ausgewählt ist.

## SCHRITT 6



⑥ Drücken Sie [F6 Pos. abbr.]. Eine Systemmeldung erscheint.

## SCHRITT 7

Prüfen Sie die Systemmeldung, und drücken Sie [OK], wenn alles korrekt ist. Die aktuelle Position wird als Wert der Variable P1 eingelesen.

## Lektion 21 Sonstige Funktionen

Neben Strombegrenzung und direkter Programmierung verfügen Denso Roboter über folgende Funktionen. Ausführliche Hinweise finden Sie im "Programmierungshandbuch".

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



# Anhänge

---

---

|                                                       |     |
|-------------------------------------------------------|-----|
| ▶ Anhang-1 Glossar                                    | 124 |
| ▶ Anhang-2 Bezeichnungen der Roboter-Controller-Teile | 134 |
| ▶ Anhang-3 Bezeichnungen der Programmiergeräts        | 136 |
| ▶ Anhang-4 Menüstruktur des Programmiergeräts         | 138 |

---

# Anhang-1 Glossar



## **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 Stopmethode, 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.

## **ARMDATEI**

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

## **ARMKONFIGURATIONSMAKRO-DEFINITIONSDATEI**

Die Datei, die die Makrodefinitionsinformationen der Armeinstellungsdaten enthält.

## **ARM MANAGER**

Die Software, welche die Roboterbewegung simuliert.

## **ARM-SEMAPHOR**

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

## **ARMSTELL**

Die Zahl, welche die Robotearmstellung anzeigt.

## **ARMSTELLUNG**

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

## **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.

## **ARMSTELLUNGSKOMPONENTE**

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**

Zu Ausschalten des Motor des Roboters.

## **AUSSCHALTEN DES ROBOTER-CONTROLLER**

Zum Ausschalten des Roboter-Controller.

## **AUTO AKTIVIEREN**

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

## **AUTOMATISCHE ROBOTER-AUSFÜHRUNG**

Die Bewegung des Roboters durch die Ausführung eines Programmes.



## **BALKENTABELLE**

Die Häufigkeitsrate des Helligkeitswertes in einem Fenster (Bildverarbeitungsterminus).

## **BASIS**

Der Teil zum Installieren der 1. Achse des Roboters.

## **BASISBEFESTIGUNGSFLÄCHE**

Die Verbindungsfläche des Basis- und Installationsrahmens.

## **BASISKOORDINATEN**

Das dreidimensionale, orthogonale Koordinatensystem mit Ursprung an der Roboterbasis.

## **BEDIENER**

Eine der Benutzerebenen von WINCAPSII. Wichtige Parameter können nicht geändert werden. Kennworteingabe ist nicht erforderlich.

## **BEDIENFELD**

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

## **BEDIENFELD**

Zum Schalten des Betriebs des internen E/A über die Anzeige des Programmiergeräts EIN/AUS.

## **BEFEHL**

In ein Programm geschriebene Anweisung. Der Controller liest Befehle in der Reihenfolge, wie sie in ein Programm geschrieben wurden, interpretiert und führt sie 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 Roboterstandardposition und mechanischer Bewegungsgrenze bestimmt.

## **BEREICH**

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

## **BETRIEBSMODUS**

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

## **BETRIEBSPROTOKOLL**

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

## **BEWEGUNGSRAUM**

Der Bereich, innerhalb dessen der Roboter gesteuert werden kann.

## **BEZEICHNUNG**

Zur Numerierung des binärisierten weißen und schwarzen Bereiches (Bildverarbeitungsterminus).

## **BIBLIOTHEK**

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

## **BILDVERARBEITUNG**

Visuelles Gerät, das von Denso hergestellt ist.

## **BINÄRISIERUNG**

Änderung der Helligkeit jedes Pixels anhand des Schwellwerts (Binärisierungs-Level) entweder auf weiß (0) oder schwarz (1).

## **BINÄRISIERUNGS-LEVEL**

Der Schwellwert der Binärisierung. (Bildverarbeitungsterminus)

#### **BREMSE AUS (Lösen der Bremsen)**

Lösung der Bremse jeder Achse.

#### **BREMSE EIN (Verriegeln der Bremsen)**

Betätigung der Bremse jeder Achse.



#### **CAL**

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

#### **CALSET**

Kalibrierung des Verhältnisses zwischen aktueller Roboter-Position und den Positionangaben des Controller.

#### **CALSET EINER EINZELNEN ACHSE**

Ausführung des CALSET an der angegebenen Achse.

#### **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**

Zur Definition des Interferenzbereiches. Er wird entweder über das Programmiergerät, in WINCAPSII oder über den Programmbefehl festgelegt.

#### **DEFINIEREN VON WERKZEUGKOORDINATEN**

Zur Definition der 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.

#### **DISKRIMINATIONSANALYSEMETHODE**

Die Methode zum Festlegen des Binärisierungsniveaus einer Balkentabelle anhand der statistischen Methode (Bildverarbeitungsterminus).

#### **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)

#### **DURCHLAUFBEWEGUNG**

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

#### **D-VARIABLE (Doppelpräzisionsvariable)**

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



#### **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-SYSTEMSIGNALE**

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

#### **EINFACH**

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

#### **EINFACH4**

Eine der Armstellungen der 4. Achse eines Roboters mit 6 Achsen. (⇔ DOPPELT4)

#### **EINSCHALTEN DES MOTORS**

Zum Einschalten des Motor des Roboters.

#### **EINSCHALTEN DES ROBOTER-CONTROLLER**

Zum Einschalten des Roboter-Controller.

#### **EINSTELLRAD**

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

#### **EINZELSCHRITTSTART**

Die Startmethode, um ein Programm in einem Schritt auszuführen. Das Programm stoppt nach einer Schrittausführung.

#### **EINZELZYKLUSSTART**

Die Startmethode, um ein Programm in einem Zyklus auszuführen. Das Programm stoppt nach einer Zyklusausführung (vor dem letzten Schritt des Programmes).

#### **ELBBOGENARMSTELLUNG**

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

#### **ENDBEWEGUNG**

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

#### **ERSTER ARM**

Der Roboterarm, welcher der Basis am nächsten ist.

#### **ETHERNET-KARTE**

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

#### **EXTERNE AUTOMATISCHE AUSFÜHRUNG**

Zur Ausführung eines Programmes 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.



#### **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 (Bildverarbeitungsterminus).

#### **FERNSTEUERUNGSBETRIEB**

Zum Steuern des Roboterarmes, 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 (Gleitpunktvariable)**

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 Mindestträgheitsmoment bei einer Umdrehung des Objekts auf einer Ebene angibt (Bildverarbeitungsterminus).

### **HAUPTACHSE - WINKEL**

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

### **HELLIGKEIT**

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



### **INITIALISIERUNGSDISKETTE**

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 HELLIGKEITSWERT**

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

### **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**

Zur Ausführung eines Programmes über das Bedienfeld oder das Programmiergerät.

### **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 (Bildverarbeitungsterminus).

#### **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**

Zum Festlegen der Einsatzbedingungen (Kommunikationsgeschwindigkeit etc.) jedes Kommunikationsanschlusses des Roboter-Controller.

#### **KOMMUNIKATIONSERLAUBNIS EINSTELLEN**

Zum Festlegen Sie der Einsatzerlaubnis jedes Kommunikationsanschlusses des Roboter-Controller fest:

#### **KOMMUNIKATIONS PROTOKOLL**

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 Robotersprache.



#### **LADEN**

Zum Einlesen von Programmen, Armdaten etc. von Diskette in den Roboter-Controller.

#### **LAUFENDER START**

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

#### **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.

#### **LINKS**

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



#### **MAKRO**

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

#### **MAKRODEFINITIONS DATEI**

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.

### **MECHANISCHER STOPPER**

Der Mechanismus zur physikalischen Begrenzung der Bewegung der Roboterachsen.

### **MECHANISCHE SCHNITTSTELLE**

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

### **MECHANISCHE SCHNITTSTELLENKOORDINATEN**

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

### **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.

### **MODUSSCHALTER**

Der Schalter am Programmiergerät. Damit kann der Roboterausführungsmodus aktiviert werden.

### **MOMENTAUFNAHME**

Die Funktion zur Aufzeichnung des aktuellen Status des Roboters.

### **MONITOR**

Zum Anzeigen des aktuellen Roboterstatus.

### **MULTITASKING**

Der Zustand, in dem mehrere Programme praktisch gleichzeitig ausgeführt werden können. Dies wird möglich, indem die CPU des Roboter-Controller 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.

**O BEN**

Eine der Ellbogenarmstellungen 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**

Zum Auflegen/Entnehmen von Teilen auf/von Palette mit Unterteilungen.

**PIXEL**

Der Punkt, aus dem sich die Bildschirmanzeige aufbaut (Bildverarbeitungstermini).

**PLIM**

Der positive, direktionale Endwert der Softwarebegrenzung. ( $\Leftrightarrow$  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.

**PROGRAMMIERER**

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

**PROGRAMMIERPRÜFUNG**

Zur Prüfung der Bewegung durch das Programm.

**PROGRAMMIERUNG**

Zur Eingabe der notwendigen Bedienungsinformationen in den Roboter über das Programmiergerät.

**PROGRAMMRÜCKSETZUNG**

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

**PROGRAMMÜBERTRAGUNG**

Zum Senden/Empfangen von Roboterprogrammen zwischen Roboter-Controller und WINCAPSII (PC).

**PROGRAMMSTART**

Das Eingabesignal zum Starten eines Programmes. Wenn es sich um einen Schrittop handelt, beginnt die Ausführung mit

dem nächsten Schritt. Beim Anhalten beginnt sie mit dem folgenden des gleichen Schrittes.

### **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. ( $\Leftrightarrow$  CP-Steuerung)

### **P-TYPE-METHODE**

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

### **P-VARIABLE (Positionvariable)**

Die durch Position, Stellung und Armstellung gekennzeichnete Variable.



### **RECHTS (RECHTS)**

Eine der Armstellungen eines Roboters mit 6 Achsen. ( $\Leftrightarrow$  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 Servofehler, Programmfehler etc. meldet.

### **ROBOTERSTOP**

Die Stopmethode, um Programme sofort zu stoppen und den Motor auszuschalten.

### **ROBOTERWARNUNG**

Das Ausgabesignal, das einen beim E/A-Befehl oder bei der Servoverarbeitung 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 Stopmethode, um ein Programm nach einer Schrittausführung zu stoppen.

### **SCHWERPUNKT**

Der Gleichgewichtspunkt, an dem das Objektgewicht auf einer Ebene balanciert (Bildverarbeitungsterminus).

### **SEMAPHOR**

Das Task-Ausführungsprivileg, 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, solange 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. (⇔ mechanische Bewegungsgrenze)

### **SPEICHERKAPAZITÄT**

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

### **SPEICHERN**

Zum Speichern von Programmen, Armdaten etc. des Roboter-Controller auf Diskette.

### **STATUS BEI NICHT ANGESCHLOSSENEM PROGRAMMIERGERÄT**

Zur Steuerung Sie den Roboters 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 Werkzeugrichtung, 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.

### **STOPTASTE**

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**

Zum Suchen eines Abstandes, der zu standardisierten Bilddaten paßt (Suchmodell) (Bildverarbeitungsterminus).

### **SYSTEM-MANAGER**

Die Software, die im allgemeinen alle Informationen von WINCAPSII verwaltet.

### **SYSTEMPROJEKT**

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

### **SYSTEMVARIABLE**

Die Variable zur Prüfung der Systembedingungen eines Programmes.



### **TAKTSIGNAL**

Das Eingangssignal, 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**

Zum Deklarieren eines Variablentyps in einem Programm.



### **UNTEN**

Eine der Ellbogenarmstellungen eines Roboters mit 6 Achsen. ( $\Leftrightarrow$  OBEN)

### **UNTERBRECHUNG ÜBERSPRINGEN**

Das Eingangssignal, 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 ausgewertet werden.

### **VISUELLES GERÄT**

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



### **WERKZEUG**

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

### **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.

### **WERKZEUGMODUS**

Der manuelle Betriebsmodus an den Werkzeugkoordinaten.



### **X-Y-MODUS**

Der manuelle Betriebsmodus an den Basiskoordinaten.



### **ZWEITER ARM**

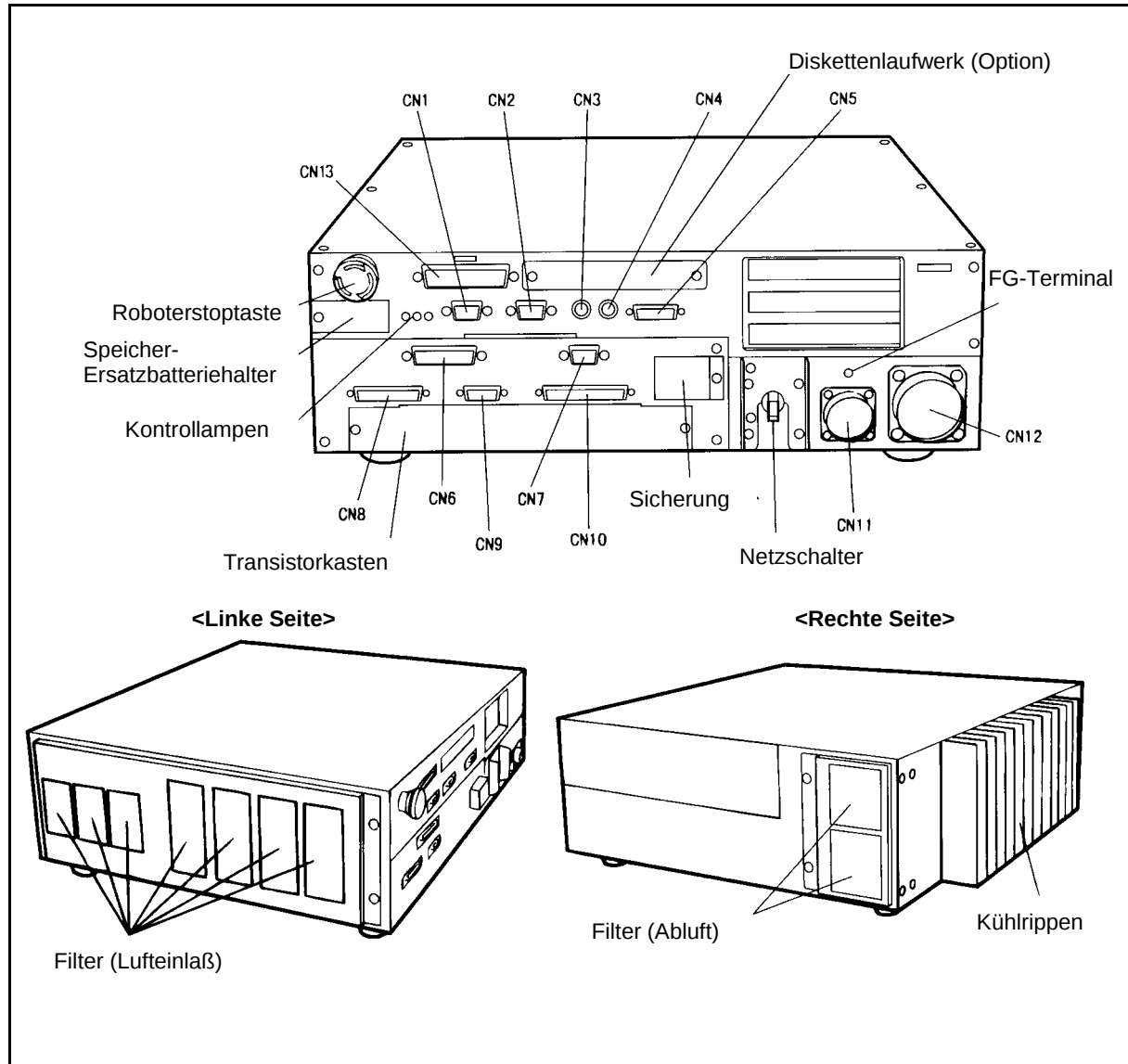
Der von der Basis gemessen weiter entfernte Arm der Roboterarme.

### **ZYKLUSSTOP**

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

## Anhang-2 Bezeichnungen der Roboter-Controller-Teile

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



Bezeichnungen der Komponenten des Roboter-Controller

### Bezeichnungen des Anschlusses

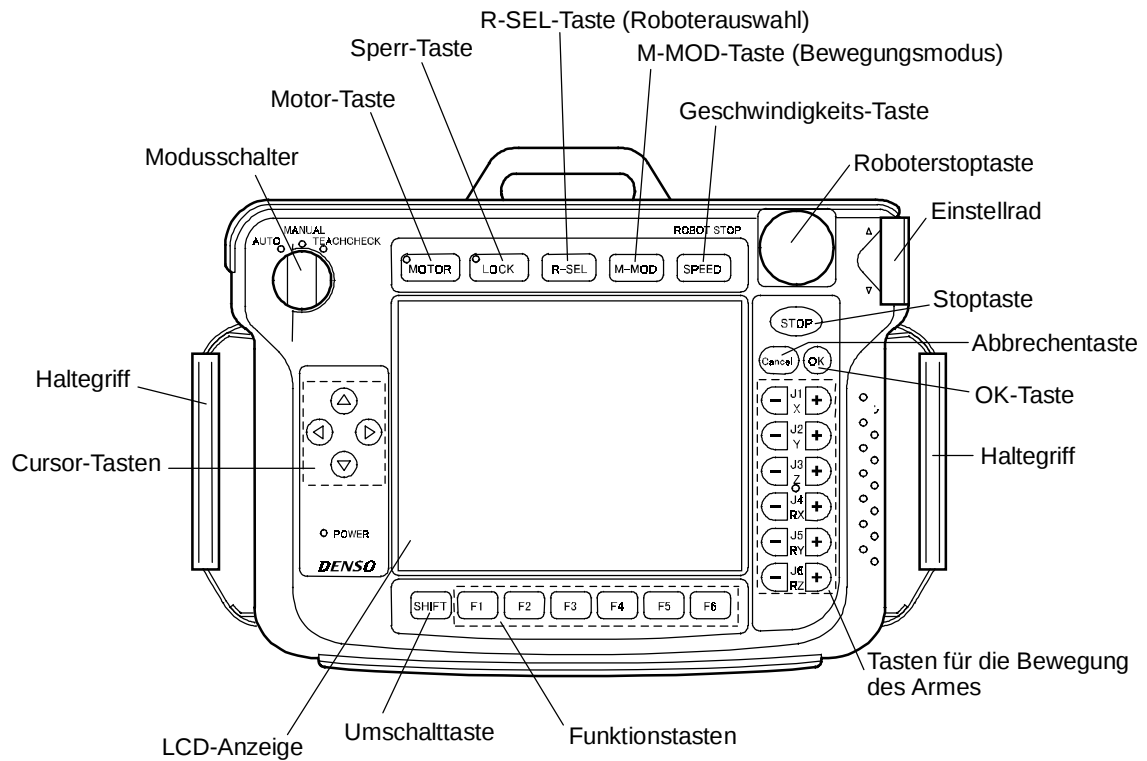
| Anschluß Nr. | Kennzeichnung    | Name                                | Anschluß Nr. | Kennzeichnung | Name                                            |
|--------------|------------------|-------------------------------------|--------------|---------------|-------------------------------------------------|
| 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 Programmiergerät       | CN12         | MOTOR         | Anschluß für Motor                              |
| CN6          | DRUCKER          | Anschluß für Drucker                | CN13         | CODIERER      | Anschluß für Codierer                           |
| CN7          | E/A-SPANNUNG     | Stromanschluß für E/A               |              |               |                                                 |



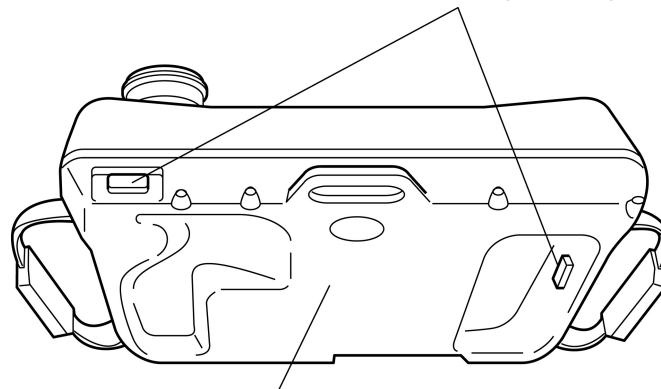
**Vorsicht:** Die Roboter-Controller-Anschlüsse haben eine Schraubsicherung oder eine Ringsicherung. Sichern Sie die Stecker richtig. Ist nur ein Anschluß nicht richtig eingerastet, kann durch unzureichenden Kontakt ein Fehler auftreten.

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

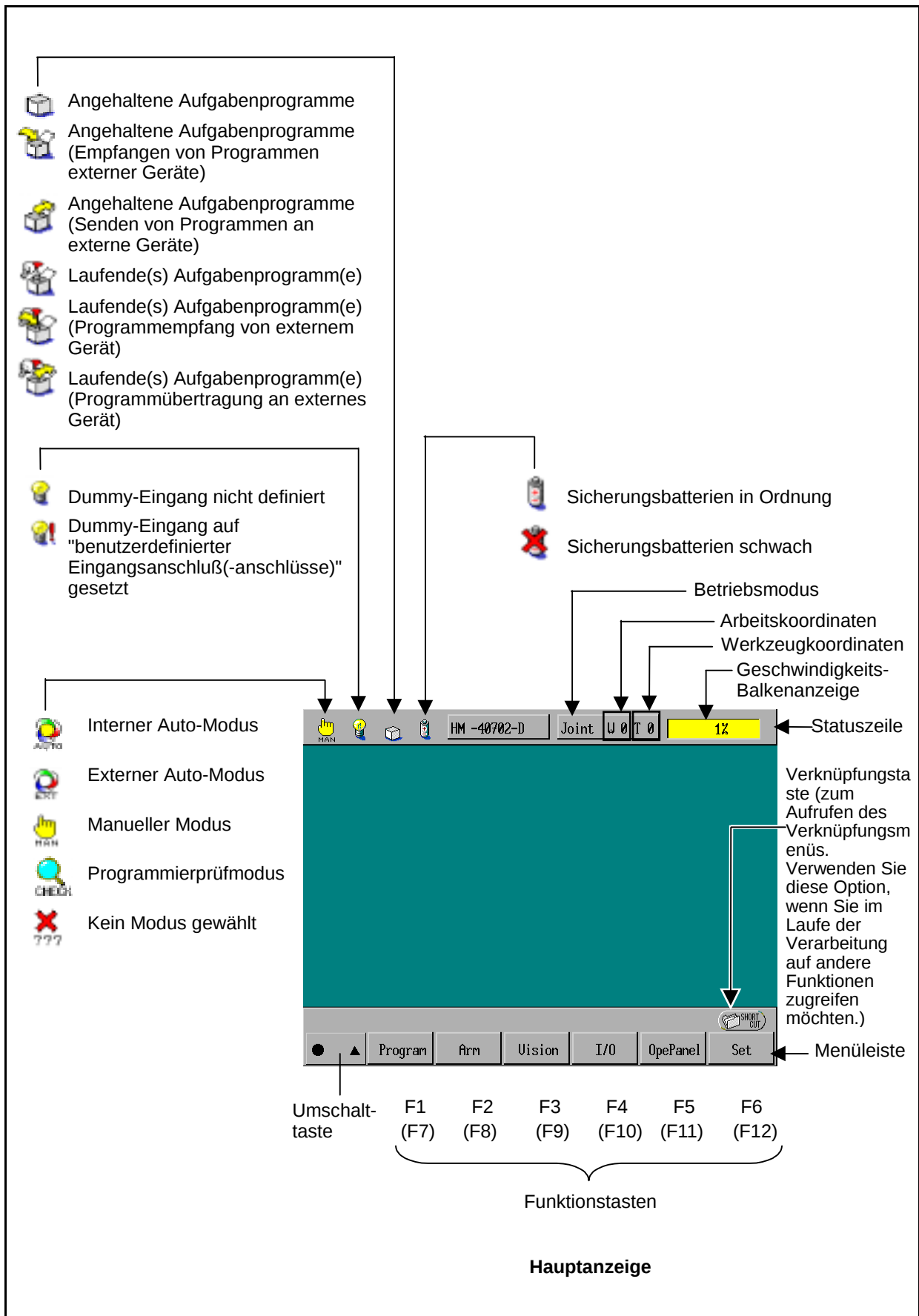
## Anhang-3      Bezeichnungen der Komponenten des Programmiergeräts



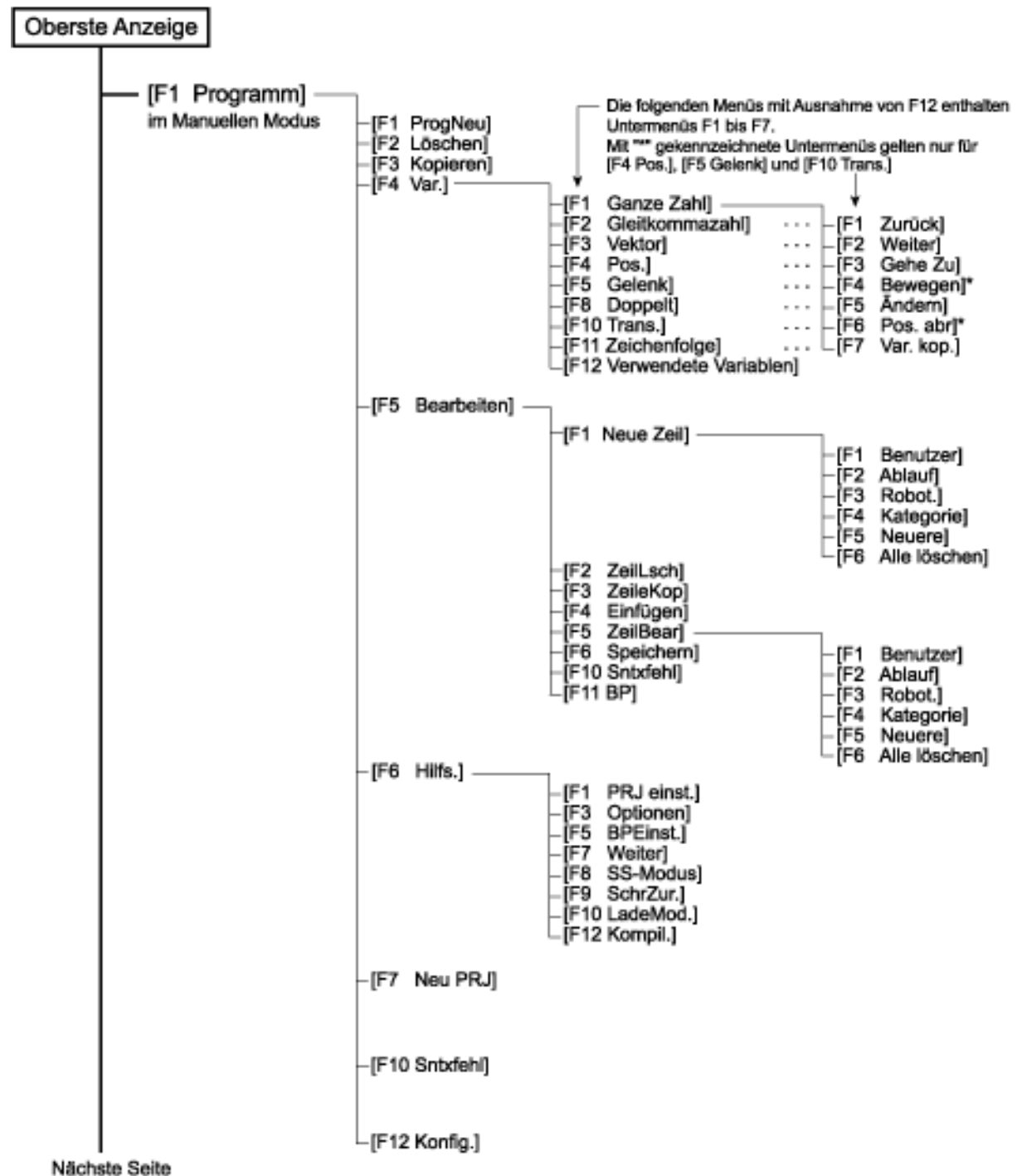
**Sicherheitsschalter**  
(Zur Steuerung des Roboters halten Sie einen dieser Schalter gedrückt, wenn Sie eine der Tasten für die Armbewegung seitwärts im manuellen Modus oder die OK-Taste im Programmierprüfmodus drücken.

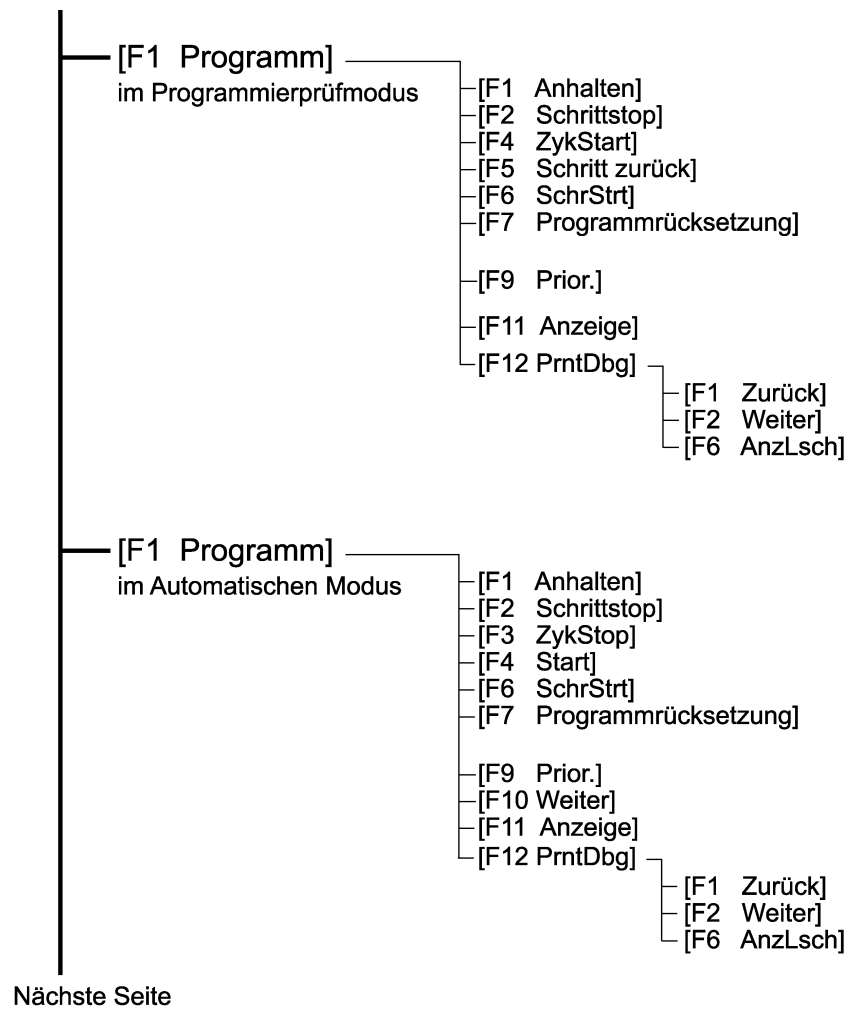


Unterseite des Programmiergeräts



## Anhang-4 Menüstruktur des Programmiergeräts





# **Horizontal ausgerichteter Roboter H\*-D-SERIE**

---

## **EINFÜHRUNGSHANDBUCH**

Erste Auflage

März 2000

DENSO CORPORATION  
Industrial Systems Product Division

---

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 willkommen.

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

