

**Betriebsanleitung**  
*Elektronisches Handrad RCS07 für CAN-CNC*

---



**isel**<sup>®</sup>

**[www.isel.com](http://www.isel.com)**

Zu dieser Anleitung:

Trotz aller Sorgfalt können Druckfehler und Irrtümer nicht ausgeschlossen werden.  
Für Verbesserungsvorschläge und Hinweise auf Fehler sind wir dankbar.

**isel** Germany AG © 2017  
Alle Rechte vorbehalten

Kein Teil dieser Veröffentlichung darf ohne vorherige schriftliche Genehmigung der Firma isel Germany AG in jeglicher Weise reproduziert, in einem EDV-System gespeichert oder übertragen werden.  
Alle Angaben in diesem Handbuch erfolgen ohne Gewähr. Änderungen des Inhalts sind jederzeit ohne Vorankündigung möglich.

Hersteller: **Isel** Germany AG  
Bürgermeister-Ebert-Straße 40  
D-36124 Eichenzell

Tel.: (06659) 981-0  
Fax: (06659) 981-776  
email: [automation@isel.com](mailto:automation@isel.com)  
<http://www.isel.com>

**Stand: 06.10.2017**

# Inhaltsverzeichnis

|   |           |
|---|-----------|
| <b>Einleitung</b> .....                                       | <b>4</b>  |
| Allgemeine Bemerkung .....                                    | 4         |
| Systemvoraussetzungen .....                                   | 4         |
| Einsatzumgebung .....   | 5         |
| <b>Handradbeschreibung</b> .....                              | <b>6</b>  |
| Hardware .....  | 6         |
| Folientastatur .....  | 6         |
| Handrad-Status-LED .....                                      | 7         |
| Handrad - Drehsteller .....                                   | 7         |
| Zustimmtaster .....   | 7         |
| Not-Aus-Schalter .....  | 7         |
| Gehäuse .....   | 8         |
| Spiralkabel mit Stecker .....                                 | 8         |
| Änderung der Grundeinstellungen .....                         | 8         |
| Software .....  | 9         |
| Allgemeine Bemerkung .....                                    | 9         |
| CANOpen-Objekte .....   | 10        |
| <i>Datenobjekt des Handrad-Drehstellers</i> .....             | 10        |
| <i>Datenobjekt der Taste-Status-LEDs und der Tasten</i> ..... | 11        |
| <b>Stichwortverzeichnis</b> .....                             | <b>13</b> |

# Einleitung

## Allgemeine Bemerkung

Zum Einrichten lassen sich die Maschinenachsen manuell oder schrittweise über die Tasten einer Tastatur oder über Achsrichtungstasten des Maschinen-Bedienfeldes verfahren. Einfacher und sicherer geht es jedoch mit einem Handrad. Mit einem tragbaren Handrad ist der Maschinenbediener immer direkt am Ort des Geschehens, hat den Einrichtvorgang im Blick und steuert feinfühlig sowie exakt die Zustellung.

Neben den Endstufen der IMD-Reihe und den IO-Modulen ist das elektronische Handrad RCS07 eine weitere Komponente der CAN-CNC-Steuerung der Firma Isel Germany AG. Im Grunde genommen ist das Handrad ein IO-Modul. Mit der eigenen Intelligenz kann das Handrad Informationen intern einsammeln und verarbeiten. Über den CAN-Bus und mit Hilfe des CANOpen-Standards findet der Informationsaustausch mit dem im PC laufenden Steuerungskern. Dank der Datenübertragung über den seriellen Bus bleibt die Anzahl der elektrischen Anschlüsse auf ein absolutes Minimum begrenzt. Daher ist das Handrad auch ideal für das Nachrüsten geeignet. Nehmen Sie mit uns Verbindung auf, falls Sie es wollen.

Obwohl das Handrad eigentlich nur ein CANOpen-IO-Modul ist, können Sie unser Handrad in einer anderen Steuerung aber nicht ohne weiteres einsetzen, weil viele Softwareerweiterungen um den Steuerungskern notwendig sind, um diesem IO-Modul die Funktionalität eines Handrads zu verleihen. Um das Handrad in eine Bedienoberfläche zu integrieren, steht eine Programmierschnittstelle zur Verfügung. Der Inhalt dieses Handbuchs wird auch als Online-Hilfe angeboten. Für das Öffnen der Online-Hilfe ist es besser, daß die Online-Hilfe auf einer lokalen Festplatte steht anstatt auf einer Netzwerkfestplatte. Falls die Inhaltsseite beim Zugriff auf Hilfethemen über das Inhaltsverzeichnis leer bleibt, liegt ein Rechteproblem beim Zugriff auf die Hilfedatei vor. In diesem Fall wählen Sie die Hilfe-Datei mit der rechten Maustaste an. Durch die Auswahl von „Eigenschaft“ wird der Dialog „Eigenschaften von ...“ eingeblendet. Auf der Karte „Allgemein“ soll die Schaltfläche „Zulassen“ angeklickt werden.

## Systemvoraussetzungen

Falls Sie unser elektronisches Handrad einsetzen wollen, müssen Sie folgende Punkte beachten.

- Die CAN-CNC-Steuerung ab der Version 01.60.00.90 mit dem CAN-PCI- oder CAN-PCI-Express-Interface ist notwendig. Das Handrad kann nur eine Achse ansteuern, die auf dem gleichem CAN-Bus-Strang liegt. D. h. wenn das zweikanalige CAN-Interface im Einsatz ist, sind dann zwei Handräder notwendig, falls man jede Achse mit einem Handrad bewegen will.
- Das Handrad arbeitet momentan nur mit den Endstufen IMD20/40 für AC-Motoren ab der Firmware-Version 01.60.00.90 zusammen. Im Zusammenhang mit den Endstufen IMD10 sowie UVE8112 für DC-Motoren oder mit dem CPC12-Modul für die Ansteuerung fremder Endstufen über  $\pm 10V$ -Signal kann das Handrad nicht eingesetzt werden.
- Die CAN-Knotennummer des Handrads liegt im Bereich 64 ... 79. Im Lieferzustand ist der Wert 65 voreingestellt. Eine Änderung des voreingestellten Werts ist zwar möglich. Das Gerät muß aber aufgemacht werden.
- Verschiedene Übertragungsgeschwindigkeiten sind bei dem Handrad möglich. Standardmäßig arbeiten wir mit der höchsten Baudrate von 1MBit/s. Eine Änderung der Baudrate ist auch nur durch das Öffnen des Gehäuses möglich.
- Im Handrad ist ein CAN-Bus-Abschlußwiderstand fest bestückt. Deswegen muß das Handrad der letzte Busteilnehmer sein. Beim Bedarf kann der Widerstand deaktiviert werden.

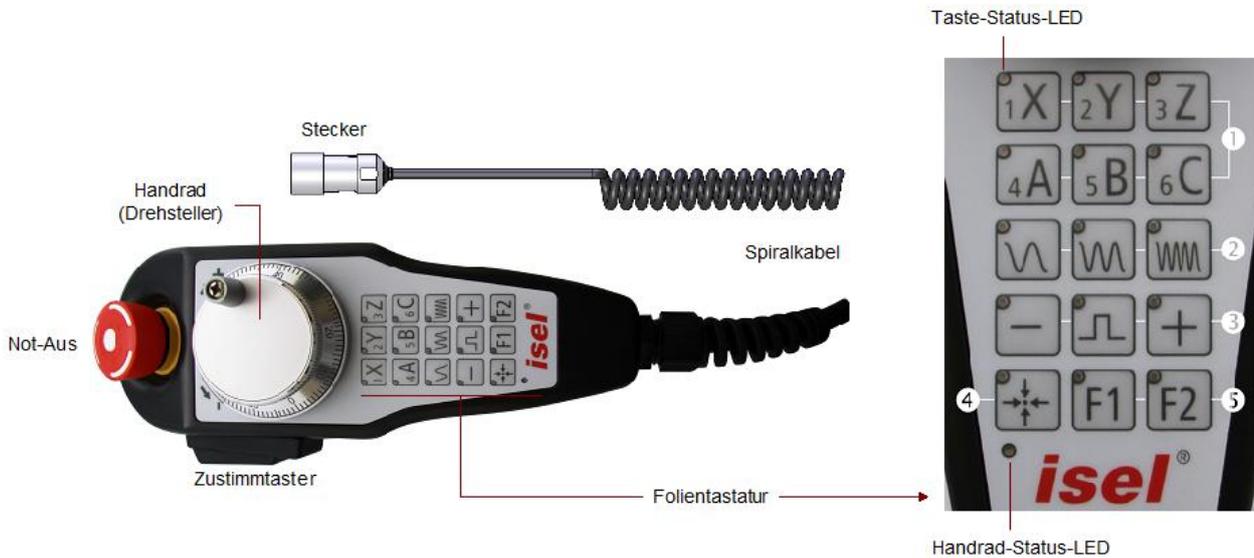
## Einsatzumgebung

Um die volle Funktionsfähigkeit des Handrads zu gewährleisten, müssen folgende Umgebungsbedingungen gewährleistet werden.

- Betriebsspannung 24V (+/-10%), Verpolschutz bis 60V
- Leistungsaufnahme: ca. 2W
- Temperaturbereich: -30 ... 70°C
- Relative Luftfeuchtigkeit: 0 ... 95% nicht kondensierend
- Gewicht: ca. 0,3kg ohne Kabel
- Schutzart: IP65

# Handradbeschreibung

## Hardware



## Folientastatur

Die Tasten sind geprägt und mit Schnappscheiben versehen, um eine gute Haptik zu garantieren. Jede der 15 Tasten der schmutzabweisenden Folientastatur ist mit einer Status-LED zum Anzeigen der Auswahl versehen. Die LED ist an bei der Auswahl. Sonst ist die LED aus. Mit der Programmierschnittstelle des Steuerungskerns kann eine Bedienoberfläche jede der Tasten still legen oder in einen der beiden Modi versetzen.

- Taster-Modus  
Die Taste ist nur aktiv beim Betätigen. Die Taste ist hier ein Taster.
- Schalter-Modus  
Bei jedem Betätigen wechselt die Taste von dem aktiven in den inaktiven Zustand oder umgekehrt. Der Zustand bleibt selbst nach dem Loslassen der Taste weiterhin erhalten. In diesem Fall ist die Taste ein Schalter.

Durch die gedruckten Symbole ist die Tastaturbelegung selbsterklärend.

1. Die 6 Schalter dienen zur Auswahl einer zu bewegendenden Achse. Eine Gantry-Achse ist nur eine Achse. Dabei ist die Master-Achse entscheidend. Als Symbol hat jede Taste einen Buchstabe und eine Nummer. Dadurch hat man die Möglichkeit, eine physikalische Achse nicht nur mit einem Buchstabe wie üblich sondern auch noch mit einer Nummer zu bezeichnen. Weiterhin kann eine Taste mit zwei Achsen belegt werden, in dem das Umschalten zwischen den beiden Achsen z. B. mit einer der frei verfügbaren Funktionstasten F1 und F2 erfolgt. In jedem Zeitpunkt darf nur eine Achse aktiv sein. Falls mehrere Tasten gleichzeitig betätigt sind, wird die erste Achse in der Reihenfolge X, Y, .... ausgewählt. Die Zuordnung einer Taste zu einer bestimmten Achse ist die Sache der Bedienoberfläche. Mit der zur Verfügung stehenden Programmierschnittstelle hat die Bedienoberfläche hier die absolute Freiheit. Aber aufgrund der

- gedruckten Symbole auf den Tasten ist es natürlich angebracht, eine Taste mit der dem Tastensymbol entsprechenden Achse zu zuordnen.
2. Durch diese 3 Schalter wählen Sie den Vorschub für die Bewegung (langsam, normal, schnell). Die Vorschübe sind Softwareeinstellungen in der Bedienoberfläche. Jederzeit kann logischerweise nur ein Vorschub aktiv sein. Falls mehrere Tasten hier gleichzeitig betätigt sind, wird der erste Vorschub in der Reihenfolge langsam-mittel-schnell aktiv.
  3. Anstatt mit dem Handrad-Drehsteller zu arbeiten, können Sie mit den beiden Tastern +/- die Achsen mit dem gewählten Vorschub konstant bewegen. Falls der mittlere Schalter  $\Pi$  aktiv ist, wird die Schrittweise-Bewegung ausgeführt. Die Schrittweite ist eine Softwareeinstellung in der Bedienoberfläche und gleich die Länge, die sich die Achse mit einem Handrad-Schritt bewegen würde. Eine Bewegung mit den Tastern +/-, die sogenannte Jog-Bewegung wird immer einer Bewegung mit dem Drehsteller bevorzugt. Falls die beiden Taster +/- gleichzeitig betätigt sind, wird nur der Taster – ausgewertet. Ohne einen aktiven Vorschub können Sie die Achsen nicht bewegen.
  4. Der Taster ist für die Übernahme der aktuellen Positionen aller Achsen gedacht. Die Positionen sind intern abgespeichert und können zurückgelesen werden. Noch nicht abgeholte Positionen werden einfach überschrieben.
  5. Die beiden Tasten F1 und F2 sind frei verfügbaren Funktionstasten. Eine Bedienoberfläche kann sie still legen oder als Taster bzw. Schalter benutzen, um weitere Funktionalität zu implementieren.

## Handrad-Status-LED

Durch diese LED signalisiert das Handrad seinen internen Betriebszustand nach außen.

-  Noch nicht initialisiert oder fehlerhaft
-  Betriebsbereit
-  Aktiv bzw. in Benutzung

## Handrad - Drehsteller

Die mechanische Rastung mit 100 Positionen pro Umdrehung ermöglicht ein sehr feinfühliges und exaktes Steuern der Bewegung. Die inkrementelle Auswertung der Positionen erfolgt intern im Handrad. Die aktuelle Position steht als 16-Bit-Wert zur Verfügung und kann von der Steuerung beim Bedarf über den CAN-Bus angefordert werden. Durch Drehen des Handrad-Drehstellers können Sie die Achsen bewegen. Die Achsgeschwindigkeit ist das Produkt aus der Drehgeschwindigkeit und dem ausgewählten Vorschub. Falls kein der Vorschub-Schalter aktiv ist, lassen sich die Achsen auch nicht bewegen.

## Zustimmtaster

Der Zustimmtaster ist 3-stufig und 2-kanalig gemäß IEC/EN 60947-5-1 ausgelegt. Im Unterschied zu einem zweistufigen Zustimmtaster garantiert ein dreistufiger die Sicherheit des Maschinenbedieners, selbst wenn er im Notfall die Zustimmungseinrichtung krampfhaft festhält. Daher ist ein dreistufiger Zustimmtaster in vielen C-Normen ausdrücklich gefordert.

## Not-Aus-Schalter

Der Not-Aus-Schalter ist 2-kanalig gemäß IEC/EN 60947-5-1 und zweipolig öffnend. Es entspricht den aktuellen Maschinenrichtlinien.

## Gehäuse

Das Gehäuse (H223mm x B91mm x T92mm) ist aus schlagfestem Kunststoff ABS (UL94V0). Auf der Rückseite sind gummierte Haftmagnete zur Befestigung an metallischen Flächen. Ein Wandhalter ist nicht notwendig bzw. kann mit einem einfachen Blechstreifen realisiert werden.

## Spiralkabel mit Stecker

Das geschirmte Kabel (18 x 0,14 mm<sup>2</sup>) hat eine Wendellänge von 1 m und daraus eine gestreckte Länge von 4 m. Der robuste und industrietaugliche Edelstahlstecker hat die folgende Pin-Belegung.

| Pin     | Signal                                 | Kabelfarbe  |
|---------|--|-------------|
| 1       | Stromversorgung +                      | Ws          |
| 2       | Stromversorgung Masse                  | Bn          |
| 3       | Zustimmtaster - Kreis 1 - Kontakt 1    | Gn          |
| 4       | Zustimmtaster - Kreis 1 - Kontakt 2    | Ge          |
| 5       | Zustimmtaster - Kreis 2 - Kontakt 1    | Gr          |
| 6       | Zustimmtaster - Kreis 2 - Kontakt 2    | Rs          |
| 7       | Not-Aus-Schalter - Kreis 1 - Kontakt 1 | Bl          |
| 8       | Not-Aus-Schalter - Kreis 1 - Kontakt 2 | Rt          |
| 9       | Not-Aus-Schalter - Kreis 2 - Kontakt 1 | Sw          |
| 10      | Not-Aus-Schalter - Kreis 2 - Kontakt 2 | Vio         |
| 11      | CAN-L                                  | Gr-Rs       |
| 12      | CAN-H                                  | Rt-Bl       |
| 13      | CAN-GND                                | Ws-Gn       |
| 14      | Diagnose Enable                        | Bn-Gn       |
| 15      | Diagnose TX                            | Ws-Ge       |
| 16      | Diagnose RX                            | Ge-Bn       |
| 17      | Frei                                   | Ws-Gr/Gr-Bn |
| Gehäuse | Schirm                                 |             |

## Änderung der Grundeinstellungen



Abschlußwiderstand      CAN-Knotennummer  
CAN-Baudrate

Im Lieferzustand hat das Handrad die CAN-Adresse 65 und eine Baudrate von 1Mbits/s. Der Abschlußwiderstand von 120 Ohm ist aktiv. Diese Voreinstellungen können Sie Ihren Bedürfnissen anpassen. Falls Sie eine Änderung hier vornehmen wollen, müssen Sie das Gerätegehäuse aufmachen. Die Adresse und die Baudrate sind über die Drehschalter einzustellen. Der Abschlußwiderstand ist einfach mit einem Seitenschneider durchzutrennen

| Schalterstellung | Gerätadresse              |
|------------------|---------------------------|
| 0                | 64=0x40                   |
| <b>1</b>         | <b>65=0x41 (Standard)</b> |
| 2                | 66=0x42                   |
| 3                | 67=0x43                   |
| 4                | 68=0x44                   |
| 5                | 69=0x45                   |
| 6                | 70=0x46                   |
| 7                | 71=0x47                   |
| 8                | 72=0x48                   |
| 9                | 73=0x49                   |
| A                | 74=0x4A                   |
| B                | 75=0x4B                   |
| C                | 76=0x4C                   |
| D                | 77=0x4D                   |
| E                | 78=0x4E                   |
| F                | 79=0x4F                   |

| Schalterstellung | Baudrate                  |
|------------------|---------------------------|
| 0                | 10kbit/s                  |
| 1                | 20kbit/s                  |
| 2                | 50kbit/s                  |
| 3                | 125kbit/s                 |
| 4                | 250kbit/s                 |
| 5                | 500kbit/s                 |
| 6                | 800kbit/s                 |
| <b>7</b>         | <b>1Mbit/s (Standard)</b> |
| 8                | unbenutzt                 |
| 9                | unbenutzt                 |
| A                | unbenutzt                 |
| B                | unbenutzt                 |
| C                | unbenutzt                 |
| D                | unbenutzt                 |
| E                | unbenutzt                 |
| F                | unbenutzt                 |

Im Bezug auf die Baudrate müssen Sie beachten, daß die gesamte Leitungslänge vom CAN-Bus bei einer Baudrate von 1 Mbits/s etwa 25 m beträgt. Dazu zählen nicht nur die Kabellänge des Handrads sondern auch noch die CAN-Busleitungen zwischen den anderen Komponenten und vom Steuerschrank bis zum Steuer-PC. Bei einer Wendellänge von 1 m und daraus einer gestreckten Länge von 4 m beträgt die Kabellänge des Handrads tatsächlich bis zu 7 m. Falls die gesamte Länge der CAN-Busleitung größer als 25 m ist, müssen Sie mit der langsameren Übertragungsrate arbeiten. Selbstverständlich müssen alle CAN-Komponenten, einschließlich des Handrads die gleiche Übertragungsrate haben. Den Zusammenhang zwischen der Übertragungsrate und der Länge der CAN-Busleitung können Sie im Handbuch ACSetup nachlesen.

## Software

### Allgemeine Bemerkung

Für das Handrad steht eine Programmierschnittstelle zur Verfügung. Mit Hilfe dieser Schnittstelle wird das Handrad in die Bedienoberfläche integriert. Die Personalisierung des Handrads wie z. B.

- Zuordnung der Tasten zu den physikalischen Achsen,
- Stilllegen einer Taste oder Belegen der Taste mit einer Funktion,
- Vorschubgrößen,
- Benutzung der Funktionstasten, ...

ist die Sache der entsprechenden Bedienoberfläche. Für die Maschinenbediener ist das Handrad ein Bestandteil der Bedienoberfläche. Um das Handrad optimal an die jeweiligen Bedürfnisse anzupassen, muß man mit der Bedienoberfläche auseinandersetzen.

Auf der Ebene des Steuerungskerns CAN-CNC gibt es ein paar Punkte zu beachten.

- Mit der Einstellsoftware CANSet können Sie über das Menü „CANSet\Extras\CAN-Bus scannen“ kontrollieren, ob das Handrad angeschlossen ist oder nicht. Falls das Handrad da ist, meldet es mit der Zeile „Isel Handwheel - RCS07 - Version. V5.0“ zurück.
- Auch in CANSet teilen Sie dem Steuerungskern die Benutzung des Handrads mit. Es geschieht im Menü „CANSet\Steuerung konfigurieren\Handrad“. Im CANSet-Handbuch können Sie sich mehr darüber informieren. Beim Vorhandensein des Handrads können Sie konfigurieren, ob Sie das Handrad benutzen wollen oder nicht. Falls das Handrad aber nicht da ist und Sie stellen hier ein, daß das Handrad benutzt werden soll, können Sie die Steuerung beim Starten nicht mehr erfolgreich initialisieren.
- Im Menü „CANSet\Steuerung konfigurieren\Koordinaten und Achsen\Koordinaten“ legen Sie mit Hilfe der Softwareendlageschalter den Bewegungsbereich der einzelnen Achse fest. Dabei sollen Sie beachten, daß die Bedienoberfläche mit Hilfe der Programmierschnittstelle jederzeit in der Lage ist, anstatt der hier gesetzten Softwareendlageschalter den eigenen Bewegungsbereich zu definieren. D. h. Ihre Einstellungen werden dann in diesem Fall vom Steuerungskern nicht benutzt.
- Durch die Diagnose-Funktionalität kann jederzeit der komplette Status aller Komponenten der CAN-CNC-Steuerung, darunter auch vom Handrad abgefragt werden. Im Menü „CANSet\Extras\Diagnose-Datei anzeigen“ können Sie mehr darüber erfahren, was für Informationen Sie sowohl während der Laufzeit als auch später mit Hilfe der Log-Datei von dem Handrad bekommen.

## CANOpen-Objekte

Das Herz des Handrads ist ein CANOpen-IO-Modul. Der Datenaustausch erfolgt über den CAN-Bus und basiert auf dem CANOpen-Standard 401. Der Handrad-Drehsteller, die Folientastatur mit ihren Tasten und den Status-LEDs werden komplett von dem IO-Modul verwaltet und ihre aktuellen Zustände sind in entsprechenden Datenobjekten abgebildet. Die Bedienung des Handrads erfolgt durch das Lesen der Daten aus diesen Datenobjekten und durch das Schreiben von Daten in diese Datenobjekte.

### Datenobjekt des Handrad-Drehstellers

|               |                                  |
|---------------|----------------------------------|
| Index:        | 0x6401                           |
| Subindex:     | Kein                             |
| Datentyp:     | INTERGER16                       |
| Zugriffrecht: | Lesen/Schreiben/PDOMapping-Lesen |

In diesem Objekt steht die Handrad-Drehsteller-Position. Pro Umdrehung liefert der Drehsteller 100 Impulse mit dem Vorzeichen für die Drehrichtung. Durch die inkrementelle Auswertung der Impulse entstehen dann die vorzeichenbehafteten 16-Bit-Positionsdaten für das Objekt. Direkt nach dem Einschalten ist die Position gleich 0.

## Datenobjekt der Taste-Status-LEDs und der Tasten

Hier sind die beiden folgenden Objekte notwendig.

### LEDs-Objekt

|               |                                      |
|---------------|--------------------------------------|
| Index:        | 0x2000                               |
| Subindex0:    | 2 - Anzahl der Subindexe             |
| Subindex1     | LEDs 1 bis 8                         |
| Datentyp:     | UNSIGNED8                            |
| Zugriffrecht: | Lesen/Schreiben/PDOMapping-Schreiben |
| Subindex2     | LEDs 9 bis 15                        |
| Datentyp:     | UNSIGNED8                            |
| Zugriffrecht: | Lesen/Schreiben/PDOMapping-Schreiben |

### Tasten-Objekt

|               |                                  |
|---------------|----------------------------------|
| Index:        | 0x2001                           |
| Subindex0:    | 2 - Anzahl der Subindexe         |
| Subindex1     | Tasten 1 bis 8                   |
| Datentyp:     | UNSIGNED8                        |
| Zugriffrecht: | Lesen/Schreiben/PDOMapping-Lesen |
| Subindex2     | Tasten 9 bis 15                  |
| Datentyp:     | UNSIGNED8                        |
| Zugriffrecht: | Lesen/Schreiben/PDOMapping-Lesen |

Jede LED wird von einem Ausgabebit des IO-Moduls angesteuert. Beim Ausgabewert 1 ist die LED an und beim Ausgabewert 0 ist die LED aus. Die LEDs sind extern angesteuert und sollen zeigen, ob die damit verbundenen Tasten aktiv sind oder nicht. Zu beachten, daß die Handrad-Status-LED ist von außen nicht ansteuerbar. Diese LED zeigt den internen Zustand des Handrads (siehe Abschnitt Handrad-Status-LED auf Seite 7).

Jede Taste ist mit einem Eingabebit des IO-Moduls verbunden. Das Eingabebit signalisiert, ob die entsprechende Taste betätigt ist oder nicht. Solange die Taste betätigt ist, ist der Bit-Wert gleich 1. Sonst ist der Bit-Wert gleich 0. Vom außen kann man durch das Lesen dieses Objekts erkennen, welche Tasten momentan betätigt sind.

Die Zuordnung der LEDs auf die Ausgabebits (Index2000\_Subindex1 und Index2000\_Subindex2) sowie die Zuordnung der Tasten auf die Eingabebits (Index2001\_Subindex1 und Index2001\_Subindex2) sind wie folgt definiert.

Subindex1:

|          |   |   |   |   |   |   |   |   |
|----------|---|---|---|---|---|---|---|---|
| LED      | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
| Taste    | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
| Position |  |  |  |  |  |  |  |  |
| Bit      | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   |

Subindex2:

|          |   |   |   |   |   |  |   |                |
|----------|---|---|---|---|---|--|---|----------------|
| LED      | 9   | 10  | 11  | 12  | 13  | 14   | 15  |                |
| Taste    | 9   | 10  | 11  | 12  | 13  | 14   | 15  |                |
| Position |  |  |  |  |  |  |  | Nicht benutzt! |
| Bit      | 0   | 1   | 2   | 3   | 4   | 5  | 6   | 7              |

Falls Sie die Absicht haben, das Handrad selbst anzusteuern, ohne die Programmierschnittstelle zu benutzen, sollen Sie das elektronische Datenblatt vom Handrad die eds-Datei handwheel.eds genauer anschauen. Hier stehen sämtliche notwendige Informationen für Sie.

# Stichwortverzeichnis

## A

Achse  
 Bewegungsbereich 10  
 Gantry 6  
 Geschwindigkeit 7  
 Master 6  
 Softwareendlageschalter 10  
 Vorschub 7, 9  
 ACSetup 9  
 Ausgabebit 11

## B

Bedienoberfläche 4, 6, 7, 9  
 Betriebsspannung 5  
 Bewegung  
 keine 7  
 konstant 7  
 schrittweise 7

## C

CAN  
 Adresse 9  
 Baudrate 4, 9  
 CNC-Steuerung 4, 10  
 gesamte Leitungslänge 9  
 Interface 4  
 Interface - zweikanalig 4  
 IO-Modul 10  
 Knotennummer 4  
 PCI-Express-Karte 4  
 PCI-Karte 4  
 CANOpen  
 Datenobjekt 10, 11  
 IO-Modul 4  
 Objekt 10  
 PDO-Mapping 10, 11  
 CANSet 10  
 CPC12-Positioniermodul 4

## D

Datenblatt  
 elektronisch 12  
 Diagnose *Siehe* Handrad

## E

eds-Datei 12  
 Eingabebit 11  
 Endstufe  
 IMD10 4  
 IMD20/40 4  
 UVE8112 4

## H

Handrad  
 Abschlußwiderstand 4, 9  
 Baudrate 4  
 Betriebsspannung 5  
 Diagnose 10  
 Drehschalter 9  
 Drehsteller 7, 10  
 Folientastatur 6, 10  
 Funktionstaste 6, 7, 9  
 Gehäuse 8  
 Haftmagnet 8  
 Hardware 6  
 Hardwareanpassung 8  
 Knotennummer 4  
 Nachrüsten 4  
 Not-Aus-Schalter 7  
 Personalisierung 9  
 Programmierschnittstelle 4, 6, 9, 12  
 Rastung 7  
 Software 9  
 Spiralkabel 8  
 Status *Siehe* LED  
 Stecker 8  
 stilllegen 10  
 Wandhalter 8  
 Zustimmungstaster 7  
 Hardware *Siehe* Handrad

## I

Impuls 10  
 inkrementell 7, 10

## K

Kabel  
 gestreckte Länge 8, 9  
 Länge 9  
 Wendellänge 8, 9

## L

LED  
 Handrad-Status 7, 11  
 Taste-Status 6, 11  
 Log-Datei 10

## P

Positionsübernahme 7  
 Programmierschnittstelle *Siehe* Handrad

## S

Software *Siehe* Handrad

Stecker 8

## **T**

Taste 11

Schalter-Modus 6

Status *Siehe* LED

stilllegen 7, 9

Taster-Modus 6

Zuordnung 9