

isel-Schrittmotor-Controller C 116-4



Hardware-Beschreibung

B.38321x.04/99.50

Zu dieser Anleitung

In dieser Anleitung finden Sie verschiedene Symbole, die Ihnen schnell wichtige Informationen anzeigen.

Gefahr



Achtung



Hinweis



Beispiel



Zusatz-Infos



© Fa. **iselautomation** 1999
Alle Rechte vorbehalten

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



isel-Maschinen und Controller sind CE-konform und entsprechend gekennzeichnet.
Für alle sonstigen Maschinenteile und -komponenten, auf die CE-Sicherheitsrichtlinien anzuwenden sind, ist die Inbetriebnahme solange untersagt, bis alle entsprechenden Anforderungen erfüllt sind.



Die Firma **iselautomation** übernimmt keine Gewähr, sobald Sie irgendwelche Veränderungen an dem Gerät vornehmen.



Die in der Konformitätserklärung aufgeführten Grenzwerte gelten nur für die ab Werk gelieferte Originalkonfiguration.

Hersteller: Fa. **iselautomation** GmbH & Co. KG
Im Leiboldzgraben 16
D-36132 Eiterfeld

Fax: (06672) 898-888
e-mail: automation@isel.com
<http://www.isel.com>

Inhaltsverzeichnis

1	Einleitung	4
2	Sicherheitshinweise	5
3	Technische Daten	6
4	Systembeschreibung	7
4.1	Anschlüsse des C 116-4	7
4.2	Aufbau isel-Schrittmotor-Controller	8
4.3	Anschluss und Inbetriebnahme	10
4.4	Bedienelemente	16
5	Optionen	18
5.1	DC-Netzteil NT-24	18
6	Inbetriebnahme	19
6.1	Anwendungshinweise	19
7	Schaltungsunterlagen	22
7.1	Prinzipschaltbild Sicherheitskreis	22
8	Konformitätserklärung	23
	Vorschriften und Normen	Anhang 1
	isel-Interfacekartenserie	Anhang 2
	isel-Schrittmotor-Steuerkarte UMS 3.5	Anhang 3
	isel-Powerblock PB xxx-C	Anhang 4
	isel-CNC-Betriebssystem	Anhang 5

1 Einleitung

isel-Schrittmotor-Controller C 116-4 sind komplette Einheiten zur Ansteuerung von maximal drei Schrittmotoren.

Sie ermöglichen in Verbindung mit einer leistungsfähigen Anwendersoftware eine 2,5-dimensionale Positionierung der angeschlossenen Antriebsachsen. Weiterhin wird eine Kreis-Interpolation mit konstanter Bahngeschwindigkeit unterstützt.

Hierzu verfügt der Controller über eine Prozessorkarte mit einem 8-Bit-Mikro-Controller und drei Leistungsstufen mit einer Ausgangsleistung von je 44 V/3,5 A.

Das Betriebssystem der Prozessorkarte (Interfacekarte UI 4.C E/A) unterstützt neben der Schrittberechnung und Impulsausgabe an die Endstufen auch die Überwachung der Referenz-/Endlagenschalter innerhalb der numerischen Antriebsachsen.

Dabei ist eine Programmierung der Karte sowohl im CNC- als auch im DNC-Modus möglich. So können die übergebenen Datensätze entweder direkt umgesetzt oder in einem 32 kB statischen RAM gespeichert werden. Mit einem optionalen Akku bleiben die Daten im RAM auch nach Ausfall der Versorgungsspannung erhalten. Darüberhinaus unterstützt die Interfacekarte den Einsatz eines auswechselbaren Scheckkarten-Speichers (Memory-Card, 32 kB).

Zur Verbindung mit einem Steuerrechner verfügt die Interfacekarte über eine serielle Schnittstelle nach RS232.

Neben den reinen Positionierbefehlen ermöglicht die Prozessorkarte auch die Bearbeitung von acht optoisolierten Signaleingängen (+ 24 V) sowie 16 Relais-Schaltausgängen (30 V/ 300 mA). Zum Anschluss entsprechender Signale sind in der Rückseite des Controllers Schraub-Steck-Klemmen vorgesehen. Ein optional einbaubares DC-Netzteil (+ 24 V/2,6 A) ermöglicht den direkten Anschluss von Sensoren, Ventilen usw. ohne zusätzliche externe Stromversorgung.

Als Leistungsstufen werden in der C 116-4 drei Schrittmotor-Leistungskarten UMS 3.5 eingesetzt. Diese bipolaren Endstufen für 2(4)-Phasen-Schrittmotoren verfügen über D-MOS-Endstufen, die bei einer Versorgungsspannung von 44 V DC mit maximal 3,5 A belastbar sind.

Die Spannungsversorgung des Controllers übernimmt der nach VDE 0160 zertifizierte Powerblock PB 450-C. Durch Integration von Überwachungsschaltungen und Sicherheitskreis stellt der Powerblock auch die Sicherheitseinrichtungen nach EN 292 zu Verfügung.



isel-Schrittmotor-Controller C 116-4

2 Sicherheitshinweise



Der Einbau bzw. Einsatz des Betriebsmittels ist entsprechend den Normen der Konformitätserklärung auszuführen. Die vom Hersteller eingehaltenen Vorschriften und Grenzwerte schützen nicht bei unsachgemäßem Gebrauch des Gerätes.

In diesem Zusammenhang sollten Sie ...

... alle Anschluss- und Montagearbeiten an dem Betriebsmittel nur unter völliger Spannungsfreiheit vorgenommen werden, d. h. Gerät abgeschaltet und Netzzuleitung gezogen.

... alle Arbeiten ausschließlich von Fachpersonal ausgeführt werden. Hierbei berücksichtigen Sie insbesondere die Bestimmungen und Vorschriften der Elektroindustrie sowie der Unfallverhütung.

Zugrundegelegte Vorschriften des Schrittmotor-Controllers:

EN 50178 (VDE 0160)

- Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln

VDE 0551

- Bestimmungen für Sicherheitstransformatoren

EN 292 Teil 1 und 2

- Sicherheit von Maschinen

EN 55011 (VDE 0875)

- Funkentstörung, Grenzwert B

IEC 1000-4 (Teil 2-5)

- Prüf- und Messverfahren der Störfestigkeit

VDE 0113 (EN 60204) Teil 1 (Ausgabe 1992)

- Elektrische Ausrüstung von Industriemaschinen

3 Technische Daten

- Gehäuse:

- Systemträger: Stahlblech-Gehäuse, B x H x T = 475 x 155 x 410 mm
- im pulverbeschichtetes Aluminium-Halbschalengehäuse (anthrazit)
- als 19-Zoll-Einbaugeschäuse (DIN 41494), 4 HE, (optional)

- isel-Interfacekarte UI 4.C E/A

- 8-Bit-Mikrocontroller mit Schrittmotor-Betriebssystem 4.1C
- 2,5-dimensionale Linear-/Zirkularinterpolation
- Positioniergeschwindigkeit maximal 10 000 Schritte/s
- 32 kB-Datenspeicher, optional mit Akku zur Datensicherung
- 8 optoisolierte Signalein-, und 16 Relais-Schaltausgänge
- vorbereitet für Einsatz eines Scheckkarten-Speicher 32 kB
- serielle Schnittstelle nach RS 232

- isel-Schrittmotor-Leistungskarte UMS 3.5

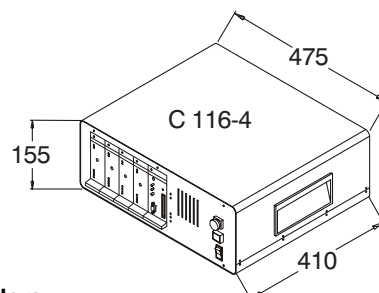
- bipolare Leistungsstufe für 2(4)-Phasen-Schrittmotoren
- Konstantstromregelung mit 18 kHz Chopperfrequenz
- Phasenstrom maximal 3,5 Ampere, kurzschlussfest
- Betriebsspannung 44 V DC

- isel-Powerblock PB 450-C

- 450 VA- Ringkerntransformator mit Temperaturüberwachung und elektronischer Einschaltstrombegrenzung
- Hilfsspannung I + 24 V/2 A, (intern, für Referenzschalter, Opto-Koppler)
- Hilfsspannung II + 24 V/1 A, (intern, für Sicherheitskreis)
- Sicherheitskreis nach EN 292 mit NOT-AUS- und EIN-Taster-Eingang
- Einbindung externer Sicherheitselemente
- geschlossenes Stahlblech-Gehäuse B x H x T = 150 x 140 x 220 mm
- Kontaktierung über Sub D-Steckverbinder
- Zertifizierung nach EN 50178 (VDE 160)
- Einhaltung der EN 55011 (VDE 0875 B)

- DC-Netzteil NT-24 (optional, Art.-Nr. 301040)

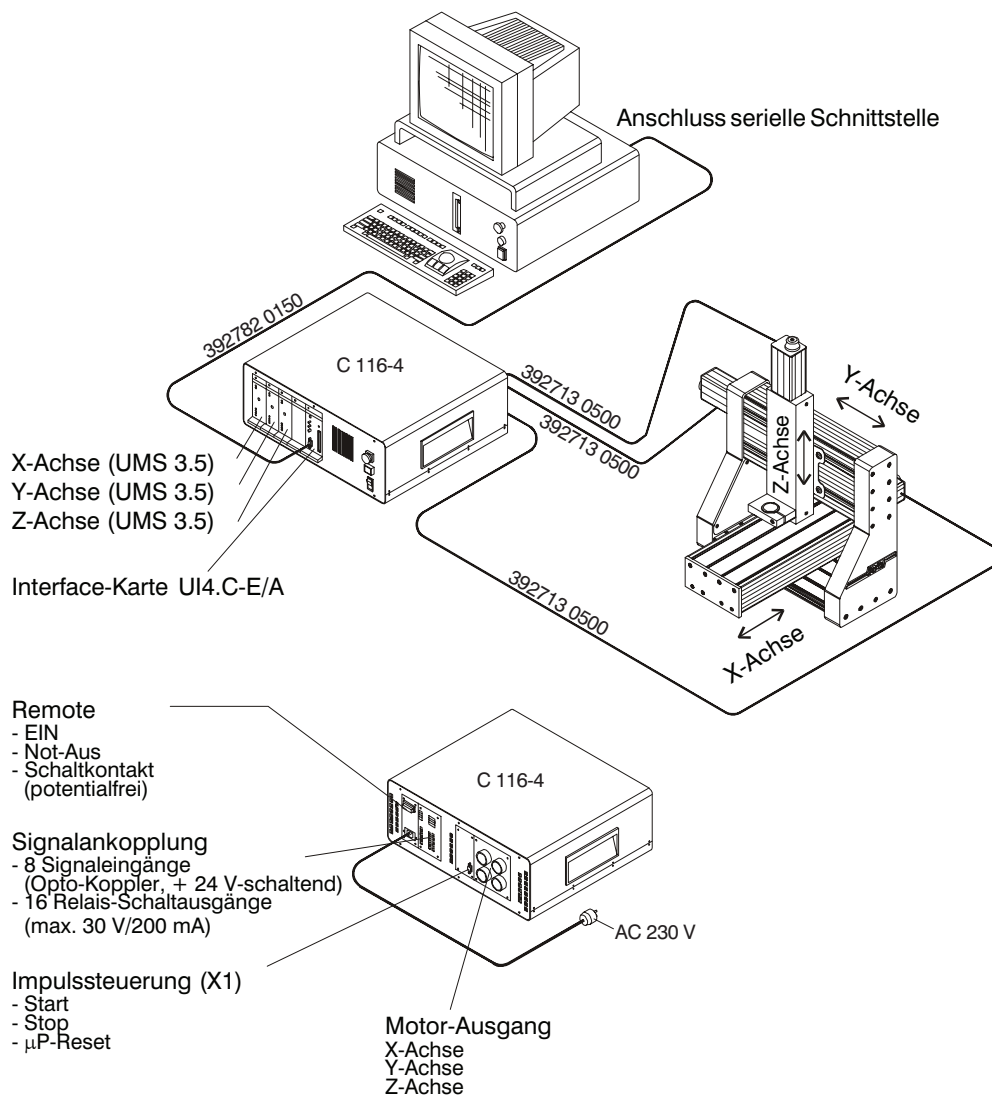
- geschlossenes Einbaunetzteil mit Ringkerntransformator
- Ausgangsleistung + 24 V/2,6 A, stabilisiert



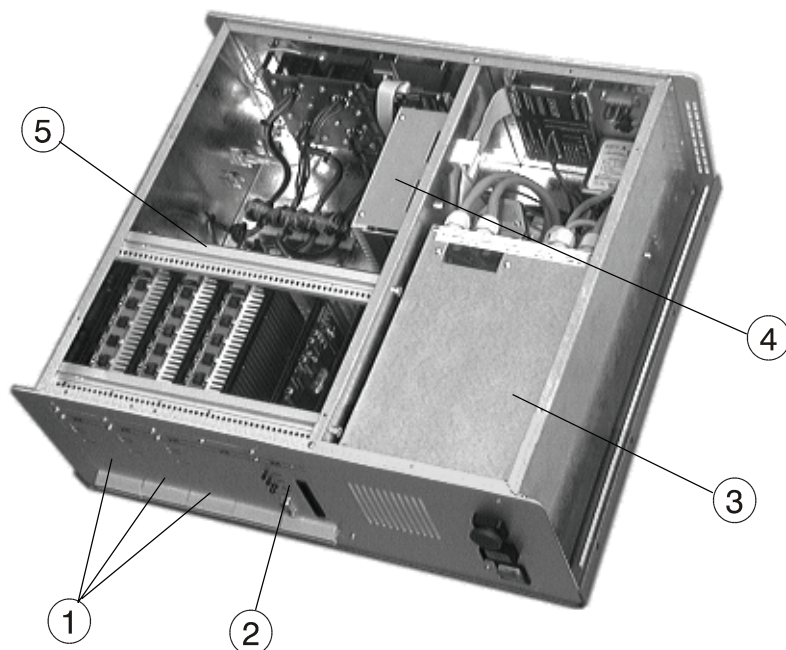
Abmessungen des Controllers

4 Systembeschreibung

4.1 Anschlüsse des C 116-4



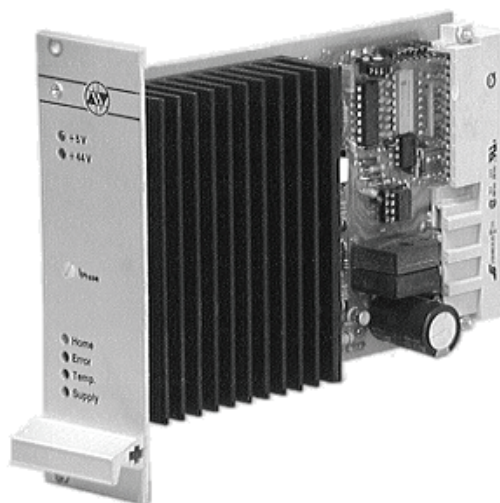
4.2 Aufbau *isel*-Schrittmotor-Controller



Schrittmotor-Controller C 116-4

- ① Schrittmotor-Leistungskarte UMS 3.5
- ② Interfacekarte UI 4.C-E/A
- ③ Powerblock PB 450-C
- ④ DC-Netzteil (optional)
- ⑤ Verbindungsplatine

① Schrittmotor-Leistungskarte UMS 3.5



Schrittmotor-Leistungskarte UMS 3.5



Zusätzliche Informationen zur UMS 3.5 sind im Anhang 2 enthalten

② Interfacekarte UI 4.C-E/A

Die Interfacekarte ist eine 8-Bit-Prozessorkarte zur Ansteuerung von maximal drei Schrittmotor-Leistungsendstufen. Das Betriebssystem berechnet aus den seriell übergebenen Positionier- und Steuerinformationen die Schrittfrequenzen der einzelnen Schrittmotoren sowie den Zustand der entsprechenden Signalein-/ausgänge. Dabei ist die Interfacekarte sowohl im DNC-Modus (on-line) als auch im CNC-Modus (Speicherbetrieb) programmierbar.

Als Datenspeicher verfügt die Karte über ein 32 kB RAM-Speicher, der durch einen nachrüstbaren Akku in einen quasipermanent-Speicher umgewandelt werden kann. Zur Einbeziehung von externen Funktionen unterstützt die Interfacekarte acht optoisolierte Signalein- und 16 Relais-Schaltausgänge.



Interfacekarte mit E/A-Modul



Zusätzliche Informationen zur UI 4.C-E/A sind im Anhang 1 enthalten

③ Powerblock PB 450-C

Der Powerblock PB 450-C ist eine kompakte Stromversorgungseinheit für Leistungseinheiten/Controller. Er umfasst neben der Bereitstellung der Zwischenkreisspannung (Betriebsspannung der Leistungsendstufen) auch die Steuer- und Überwachungsfunktionen von sicherheitsrelevanten Komponenten einer komplexen Antriebssteuerung.



Powerblock PB 450-C

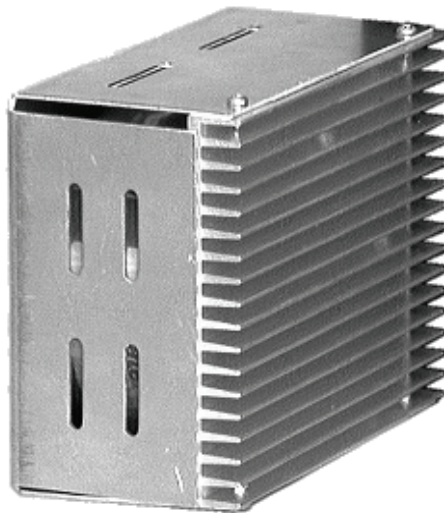


Zusätzliche Informationen zum Powerblock sind im Anhang 3 enthalten

④ DC-Netzteil (optional)

Das Einbaunetzteil dient zur Spannungsversorgung der externen Signalein- und -ausgänge des Schrittmotor-Controllers.

Das Netzteil wird innerhalb des Controllergehäuses montiert und über die Spannungsversorgung des Controllers eingeschaltet. Die Ausgangsleistung der stabilisierten Festspannung beträgt + 24 V/2,6 A. Hierdurch lassen sich kleinere Schalteinheiten (z. B. Ventile, Relais, Sensoren etc.) ohne zusätzliche Maßnahmen direkt betreiben.



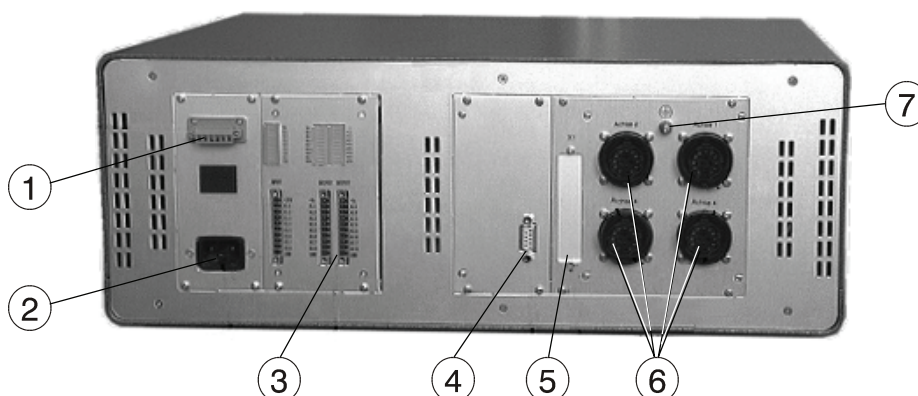
Einbaunetzteil 24 V/2,6 A

⑤ Verbindungsplatine

Die Verbindungsplatine dient zur Aufnahme der Schrittmotor-Leistungskarten UMS 3.5 sowie der Prozessorkarte. Ebenso wird hier (bei herausgezogener Leistungskarte) die Betriebsart-Einstellung der Schrittmotoren festgelegt (siehe Kapitel 2.2 Absatz 7).

4.3 Anschluss und Inbetriebnahme

Verbinden Sie den Schrittmotor-Controller C 116-4 und die mechanischen Antriebskomponenten über die Steckverbinder auf der Rückseite der Geräte.



Rückansicht C 116-4

① Remote

Der Remote-Steckverbinder dient zur Adaption des Sicherheitskreises des Schrittmotor-Controllers.

Belegung des Steckverbinders:

- 1 - 2 ——— potentialfreier Schaltkontakt (Schließer, Ausgang)
- 3 - 4 ——— NOT-AUS-Einrichtung (Öffner-Kontakt, Eingang)
- 5 - 6 ——— EIN-Taste (Schließer-Kontakt, Eingang)

Die Steuersignale im einzelnen:

potentialfreier Schaltkontakt (1 - 2)

Der potentialfreie Schaltkontakt dient zur Einbindung der Leistungseinheit in übergeordnete NOT-AUS-Systeme. Der Kontakt bleibt geschlossen, bis die Leistungsendstufen mit Spannung versorgt werden (Sicherheitsrelais und Schaltrelais sind aktiv). Eine Unterbrechung der Sicherheits-Einrichtung führt zum Öffnen des Kontaktes.

NOT-AUS-Einrichtung (3 - 4)

Dieser Eingang dient zum Anschluss von externen Sicherheits-Einrichtungen (NOT-AUS-Schalter, Sicherheitsschalter etc.). Es handelt sich hierbei um einen Schaltkontakt des internen Sicherheitskreises. Bei nicht benötigter externer NOT-AUS-Einrichtung ist das Kontakt-Paar durch eine Brücke zu schließen.



Die Klemmen führen die Spannung des Sicherheitskreises. Achten Sie unbedingt darauf, dass als Schaltelement ein potentialfreier Öffner-Kontakt verwendet wird.

EIN-Taste (5 - 6)

Dieser Schaltkontakt ist parallel zu dem frontseitigen EIN-Taster geschaltet, d. h. das Schließen des Kontakt-Paares bewirkt, wenn alle Sicherheitsbedingungen erfüllt sind, das Einschalten der Betriebsspannung der Schrittmotor-Leistungsendstufen.



Da gemäß der Maschinenschutzverordnung nur eine EIN-Taste im sicherheitsrelevanten Teil einer Steuerung vorhanden sein darf, ist der Anschluss eines externen EIN-Tasters nur unter der Maßgabe möglich, dass der frontseitige EIN-Taster durch entsprechende Maßnahmen (Einbauort des Controllers, Abdeckung des Schalters etc.) unwirksam gemacht wird.

② Netzeingang

Der Netzeingang dient zur Spannungszuführung des *isel*-Powerblockes und des DC-Netztes. Dabei wird die Versorgungsspannung über den frontseitigen, 2-poligen Netzschalter und danach über eine Einschaltstrombegrenzung zum Leistungstransformator geführt.

Die Spannungsversorgung des optionalen DC-Netztes (AC 230 V/50 Hz) wird durch ein Schaltrelais im Powerblock freigegeben.

Bei einer Betriebsspannung von AC 230 V/50 Hz ergibt sich für den Controller eine Gesamtstromaufnahme von ca. 3,0 Ampere.



Der Controller kann auch mit einer Netzspannung von AC 125 V/60 Hz geliefert werden. Hierbei erhöht sich die Nennstromaufnahme auf ca. 5,0 A

③ E/A-Ankopplung

Das Ankopplungsmodul dient zum Anschluss von externen Einheiten an die galvanisch getrennten Ein- und Ausgänge des Schrittmotor-Controllers. Bei den Signaleingängen handelt es sich um acht Opto-Koppler (+ 24 V-schaltend). Die Auswertung der Eingänge wird von der Prozessorkarte vorgenommen. Durch entsprechende Programmierung können hierdurch z. B. Sensoren überwacht oder Verzweigungen im Datenfeld realisiert werden. Als Ausgänge stellt der Controller 16 Relais-Schaltausgänge (+ 24 V-schaltend) mit einer maximalen Schaltleistung von 30 V/0,3 A zur Verfügung. Zur Spannungsversorgung des Modules kann entweder das optionale DC-Netzteil NT-24 in den Controller integriert oder eine entsprechende Spannung extern an die Klemmen + 24 V und GND angelegt werden.

④ Steckverbinder X2

Der 9-polige Sub D-Buchsenstecker ermöglicht den Anschluss von externen Schaltelementen, die in ihrer Funktion denen der Prozessorkarte entsprechen. Hierdurch können die frontseitigen Taster Start, Stop und μ P-Reset auf ein externes Bedienteil gelegt werden. Die Eingänge sind optoisoliert und arbeiten mit einer Signalspannung von + 24 V.

Anschlussbelegung des Steckverbinders:

	Signal	Pin	Pin	Signal
Prozessor-Reset	E	1	6	A + 24 V
Stop-Taste	E	2	7	A + 24 V
Start-Taste	E	3	8	A + 24 V
GND	A	4	9	A GND
+ 24 V	A	5		

μ P-Reset (Kontakt 1 - 6)

Der μ P-Reset-Taster bewirkt einen Hardware-Reset der Interfacekarte und somit den abrupten Abbruch aller Funktionen des Controllers. Gleichzeitig wird der Steuerausgang Bremse deaktiviert und somit die Steuerspannung zu einer eventuell angeschlossenen Magnetbremse abgeschaltet. (Eine spannungslose Magnetbremse blockiert die Motorwelle und verhindert dadurch eine selbsttätige Bewegung des Antriebes. Die Funktion wird durch einen Schliesser-Kontakt-Taster erreicht.

Stop (Kontakt 2 - 7)

Der Halt-Taster bewirkt eine gesteuerte Unterbrechung der angeschlossenen Schrittmotorbewegung durch Einleiten einer Bremsrampe. Die Funktion wird durch einen ÖFFNER-Kontakt-Taster erreicht.



Bei Betrieb des Controllers ist unbedingt eine Verbindung der Kontakte notwendig. Hierzu ist entweder ein entsprechender Taster anzuschließen oder die Kontakte mittels einer Brücke zu verbinden.

Start (Kontakt 3 - 8)

Der Start-Impuls bewirkt die Ausführung eines in der Interfacekarte gespeicherten Datenfeldes bzw. die Fortsetzung eines unterbrochenen Befehlsablaufes. Die Funktion wird durch einen Schließer-Kontakt-Taster erreicht.

⑤ Steckverbinder X1 (in Vorbereitung)

Der 50-polige Ribbon-Steckverbinder ermöglicht die direkte Ansteuerung des Controllers durch + 24 V SPS-kompatible Steuersignale. Hierzu sind die Schrittpuls- und Richtungseingänge der Leistungsendstufen über Opto-Koppler getrennt auf den Steckverbinder geführt. Ebenso liegen auf dem Steckverbinder die Ausgangssignale der Referenzschalter in den Antriebseinheiten sowie einige Status-Signale des Controllers.

⑥ Ausgangssteckverbinder Schrittmotor

Zum Anschluss der Schrittmotor und Referenzschalter stellt der Controller rückseitig vier Rundsteckverbinder zur Verfügung.

Diese sind wie folgt den mechanischen Antriebsachsen zugeordnet:

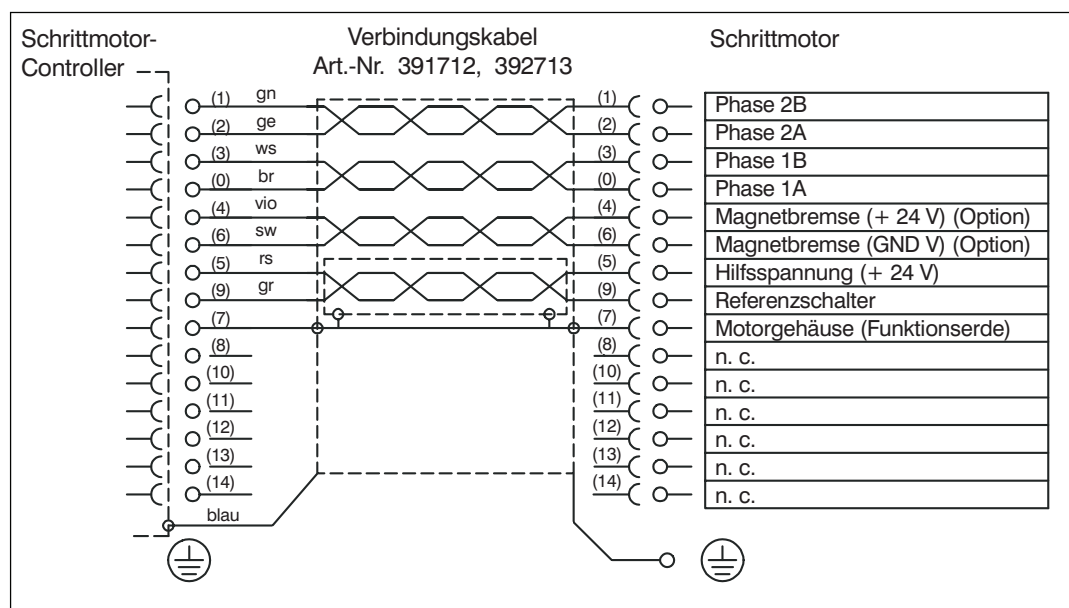
Achse 1	—	Schrittmotor X-Achse
Achse 2	—	Schrittmotor Y-Achse
Achse 3	—	Schrittmotor Z-Achse
Achse 4	—	nicht bestückt (in Vorbereitung)

Als Motoranschlussleitungen sind abgeschirmte Kabel einzusetzen, deren Schirmgeflecht sowohl auf der Controllerseite als auch auf der Motorseite auf das Gehäusepotential gelegt werden müssen.



Das beidseitig aufgelegte Schirmgeflecht der Motorleitung stellt keine Schutzleiterverbindung bzw. Potentialausgleich der Einheiten dar sondern dient lediglich der Funktionserdung.

Die Belegung des Steckverbinders ist wie folgt definiert:



Aufbau der Motoranschlussleitung (Art.-Nr. 392 712, 392 713)

Motorphasen

Die Ausgänge 1A und 1B sowie 2A und 2B sind die Leistungsausgänge des Controllers. Sie sind signalrichtig mit den Motorphasen des Schrittmotoren zu verbinden.

Auswertung Referenzschalter

Referenzschalter dienen in numerisch gesteuerten Anlagen (Bearbeitungs- oder Handlingssysteme) zur Bestimmung des Maschinennullpunktes. Nach erfolgter Referenzfahrt werden alle Positionieranweisungen im Absolutmaßsystem auf diesen Nullpunkt bezogen.

Die Signalspannung der Schalter beträgt + 24 V (plusschaltend).

Endlagenschalter (optional)

Endlagenschalter werden zur Absicherung der maximalen Verfahrswege und somit zum Schutz vor Zerstörung der mechanischen und elektrischen Komponenten eingesetzt. Sie führen bei Betätigung zum Abschalten der Leistungseinheit.



Die Endlagenschalter werden vom Schrittmotor-Controller C 116-4 nicht ausgewertet. Eine Begrenzung der Verfahrswege wird hier über den Referenzschalter realisiert.

Magnetbremse

Die Ausrüstung einer Antriebsachse mit einer Bremse ist dann sinnvoll, wenn die Kraftmomente, die auf die Antriebsachse einwirken, größer sind als deren Haltemomente. Dies kann z. B. schon bei senkrechter Montage einer Antriebsachse bei abgeschalteter Betriebsspannung des Controllers bzw. bei Netzspannungsausfall eintreten.

Bei den in isel-Antriebsachsen montierten Einheiten handelt es sich um Magnetbremsen, die im spannungslosen Zustand das 2. Wellenende des Schrittmotors und damit die Antriebsspindel bremsen. Durch Bestromen mit einer + 24 V Spannung (Leistungsaufnahme ca. 500 mA) wird die Bremswirkung aufgehoben.

Die Ansteuerung des Relais geschieht durch ein Steuersignal der Prozessorkarte, unter Berücksichtigung der Freigabe durch die Stromversorgung des Controllers.

Zum Lieferumfang der Controller gehören konfektionierte Verbindungsleitungen $l = 3 \text{ m}$, bzw. $l = 5 \text{ m}$.

Zur Vermeidung von Kontaktierungsfehlern ist die Kodierung der Steckverbinder in Schrittmotor-Controllern auf Code 6 gelegt.



Die Versorgungsspannung (+ 24 V) ist extern über den 2-poligen Steckverbinder auf der Rückseite zuzuführen. Hierzu können Sie ggf. die Spannung an der Signalkanopplung E/A abgreifen. Bitte halten Sie hierzu Rücksprache mit unserer Elektronikabteilung.

Funktionserdung

Die zusätzlich aus dem Steckverbinder herausgeführte Leitung ist im Stecker mit dem äußeren Kabelschirm verbunden. Sie dient zur Funktionserdung der Einheiten und ist rückseitig auf den mit einem Erdungszeichen versehenen Gewindebolzen zu führen.

⑦ Schutzleiter / Potentialausgleich

Neben der Funktionserdung über die Motoranschlussleitungen zum Potentialausgleich sind die einzelnen Funktionseinheiten eines Antriebssystems mit einer niederohmigen Schutzleiterverbindung auszurüsten.

Gemäß der VDE 0113 sind hierbei ... alle Körper der elektrischen Ausrüstung und der Maschine (einschließlich des Rahmens) ... mit dem Schutzleiter zu verbinden.

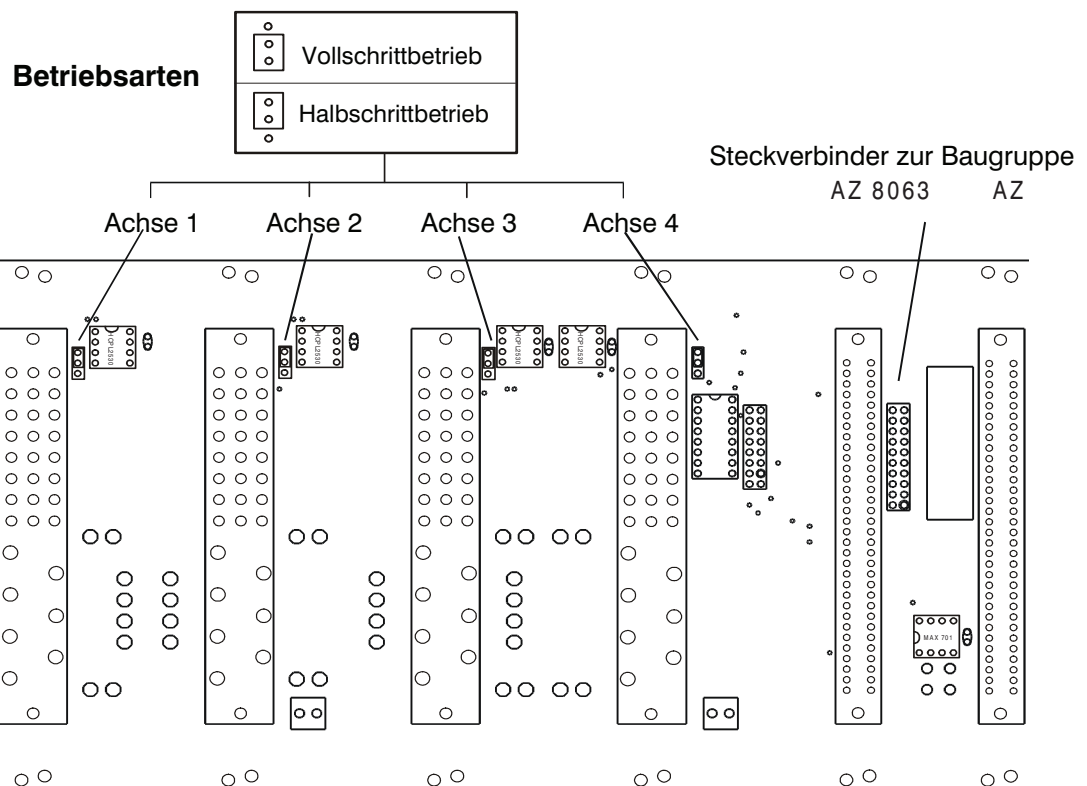
Bezogen auf den Controller ist der entsprechende Schutzleiteranschluss rückseitig als M4-Gewindebolzen herausgeführt. Antriebsseitig ist der Schutzleiter auf einen geeigneten Erdungspunkt zu führen.



Der Potentialausgleich ist zudem notwendig, um die in der Konformitätserklärung definierten Normen und deren Grenzwerte einzuhalten. Hierzu ist ein Leiterquerschnitt von 1,5 mm² notwendig.

⑧ Einstellung Schrittauflösung

isel-Schrittmotor-Leistungskarten UMS 3.5 ermöglichen die Ansteuerung eines Schrittmotors im Vollschritt bzw. Halbschritt-Betrieb. In Verbindung mit isel-Schrittmotoren werden hierdurch Schrittweiten von 1,8° (200 Schritte/Umdrehung) im Vollschritt-Betrieb und 0,9° (400 Schritte/Umdrehung) im Halbschritt-Betrieb erreicht. Die Vorgabe der Schrittweite geschieht durch eine Jumper-Brücke auf der Rückwandverdrahtung des Controllers.

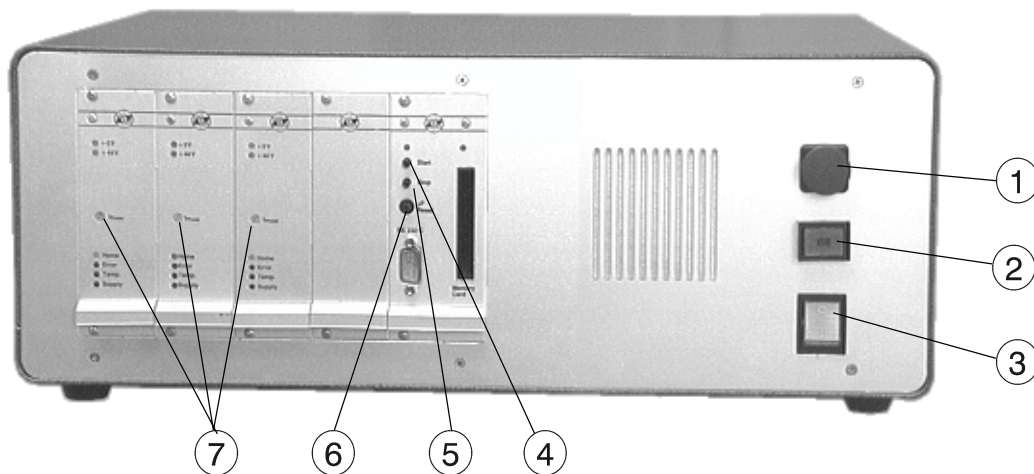


Rückwandplatine Schrittmotor-Controller (Platinenzeichnung)



Das Umstecken der Jumper ist nur bei ausgeschaltetem Controller vorzunehmen. Bei Auslieferung wird die Betriebsart auf Halbschritt-Betrieb eingestellt, um die Resonanzeigenschaften des Schrittmotor-Systemes zu reduzieren.

4.4 Bedienelemente



Frontseite *isel*-Schrittmotor-Controller

- ① NOT-AUS-Tastschalter
- ② EIN-Taste
- ③ Netzschalter
- ④ Start-Taste
- ⑤ Stop-Taste
- ⑥ Prozessor-Reset
- ⑦ Phasenstrom-Potentiometer

① **NOT-AUS-Tastschalter**

Der NOT-AUS Tastschalter ist ein Schaltelement mit zwangsgeführten Kontakten. Bei Betätigung unterbricht er den Sicherheitskreis des Controllers und schaltet die Spannungsversorgung der Leistungsendstufen ab. Gleichzeitig werden die Leistungstransistoren der Endstufen stromlos geschaltet (disabled) und ein Prozessor-Reset der Interfacekarte ausgelöst.

② **EIN-Taste**

Die EIN-Taste schaltet bei geschlossenem Sicherheitskreis (alle sicherheitsrelevanten Bedingungen sind erfüllt) das Sicherheits-Relais und damit die Spannungsversorgung der Leistungsendstufen ein. Durch Selbsthaltung der Leistungsrelais' wird ein selbsttätiges Wiedereinschalten des Controllers vermieden.

③ **Netzschalter**

Der Netzschalter dient zum zweipoligen Abschalten der Netzspannung des Schrittmotor-Controllers. Leuchtet der integrierte Kennmelder, ist der Controller betriebsbereit.

④ Start-Taste

Die Start-Taste bewirkt die Ausführung eines im Datenspeicher abgelegten CNC-Datenfeldes. In Verbindung mit dem μ P-Reset-Schalter wird ein Selbsttest des Controllers eingeleitet.

⑤ Stop-Taste

Die Stop-Taste unterbricht die Ausführung einer aktuellen Impulsausgabe an die Endstufen durch Einleiten einer Bremsrampe. Den unterbrochenen Prozess können Sie durch Betätigen der Start-Taste bzw. dem Befehl "@0S" erneut starten.

⑥ Prozessor-Reset

Der Prozessor-Reset unterbricht alle Aktivitäten der Interfacekarte. Eventuell auftretende Schrittfehler der Schrittmotoren (bedingt durch den abrupten Abbruch der Schritimpuls-Ausgabe) werden ignoriert. Durch Betätigen der μ P-Reset-Taste bei gleichzeitig betätigter Start-Taste wird ein Selbsttest des Controllers eingeleitet.



Der Selbsttest der Interfacekarte wird erst nach Abschalten der Spannungsversorgung oder durch einen nochmaligen μ P-Reset abgebrochen. Ist während des μ P-Reset die Speicherkarte gesteckt, wird ein dort gespeichertes Datenfeld in das statische RAM der Prozessorkarte kopiert.

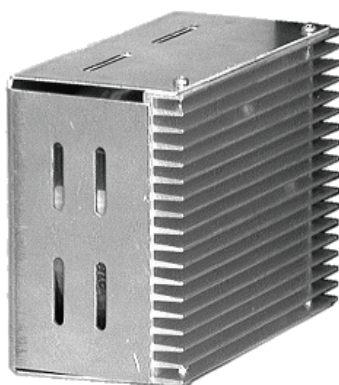
⑦ Phasenstrom-Potentiometer

Das Phasenstrom-Potentiometer der Leistungsendstufe ermöglicht die Anpassung des Treiberstromes an den benötigten Motorstrom. Der Einstellbereich liegt zwischen 0,1 A und 3,5 A.

5 Optionen

5.1 DC-Netzteil NT-24

Das DC-Netzteil ist eine kompakte Einbau-Stromversorgung mit einer Ausgangsleistung von + 24 V/2,6 A. Es besteht aus einem Ringkerntransformator und einer Spannungsstabilisation durch Festspannungsregler.



DC-Netzteil NT-24

Technische Daten

Eingangsspannung	AC 230 V/50 Hz ($\pm 10\%$)
Ausgangsspannung	DC + 24 V
Ausgangstrom 100 % ED	2,6 A
80 % ED	3,0 A
50 % ED	3,5 A
Übertempeartur bei 80 % ED	48 °K
Restwelligkeit	20 mV
Gehäuse	Alu-Blechgehäuse
Gewicht	2,0 kg

Zur Montage des Netzteiles innerhalb des Controllers sind an der Trennwand zwei Befestigungsbohrungen ($d = 6\text{ mm}$) vorbereitet. Der elektrische Anschluss findet über Steckverbinder statt.

Bestellbezeichnung: DC-Netzteil NT-24

Art.-Nr. 301040

6 Inbetriebnahme

6.1 Anwendungshinweise

- Nach Einschalten des Controllers und Freigabe der Spannungsversorgung für die Endstufen durch den EIN-Taster, bleibt die Interfacekarte noch für weitere 1-2 Sekunden im Reset-Zustand. In dieser Zeit kann die Prozessorkarte weder über die serielle Schnittstelle angesprochen noch durch Tasten bedient werden.
Ebenso ist in dieser Zeit der Steuerausgang Bremse inaktiv, d. h. die am Motor angeflanschten Magnetbremsen verhindern das Anlaufen der Motoren. Wird innerhalb dieser Zeitschleife die Start-Taste betätigt, erfolgt automatisch ein Selbsttest der Interfacekarte.
- Im Controller C 116-4 wird eine angepasste Interfacekarte eingesetzt.
Diese Interfacekarte mit der Bezeichnung UI 4.C E/A unterscheidet sich in einigen Speicheradressen von der UI 4.0-E/A, der Interfacekarte des Controller C 116.
Das Betriebssystem 4.xC bleibt jedoch unverändert, sodass die vorhandenen Anwenderprogramme uneingeschränkt angewendet werden können.
Ebenso behält auch die Hardware-Beschreibung der Interfacekarte ihr Gültigkeit.
Die Anpassung der Interfacekarte ist besonders dann zu berücksichtigen, wenn ein Schrittmotor-Controller aus Einzelkomponenten zusammengestellt wird.
Eine Umrüstung der Interfacekarten UI 4.0-E/A auf UI 4.C-E/A ist möglich.
- Das Betriebssystem 4.xC der Interfacekarte UI 4.C-E/A verfügt zusätzlich zum Betriebssystem 4.x über Befehlszyklen, die eine Programmierung durch die Standard-Software PRO-PAL, PRO-DIN sowie HP-DOWN 2.0 erlauben. Hierzu wird jedoch auch der Software-Treiber i5drv benötigt, sodass nur eine indirekte Kommunikation mit der Interfacekarte möglich ist.
- Das Standard-Betriebssystem 4.xC der Interfacekarte ermöglicht in Verbindung mit Memory-Karten die Speicherung von Daten des internen RAM-Speichers auf austauschbaren Speichermedien. Zur Programmierung der Speicherkarten steht entsprechend der Anleitung des isel-CNC-Betriebssystems der Befehl @0u zur Verfügung. Das automatische Abspeichern innerhalb des Datenfeldes (Befehlswort: save) ist nicht zu empfehlen.
- Die Signalspannung der Referenzschalter ist im Gegensatz zum C 116-Controller von GND-schaltend auf + 24 V-schaltend geändert worden.
Das hat zur Folge, dass die in 'alten' Kabel vorhandene Brücke zwischen Kontakt 5 und Kabelschirm zum Kurzschluss der + 24 V-Versorgungsspannung führt. In diesem Fall ist die Kontaktbelegung der Kabelsteckverbinder beidseitig entsprechend Kapitel 4.3 Absatz 6 anzupassen.

- Zur Einstellung des Schrittmotor-Phasenstromes verfügen die Leistungsendstufen UMS 3.5 über frontseitige Potentiometer. Der optimale Betriebsstrom ergibt sich aufgrund der technischen Daten des Motors unter Berücksichtigung des effektiven Leistungs-verbrauches.
Im Normalfall kann der Schrittmotorstrom nach akustischen Merkmalen optimiert werden. Hierbei zeugt eine ruhiges gleichförmiges Laufgeräusch des Motors von einer korrekten Einstellung. Ein weitere Stromerhöhung führt hier zu einem erhöhten Laufgeräusch (kratzen), eine Stromreduzierung zum Ausphasen, d. h. Stillstand des Schrittmotors.
Bei Auslieferung ist der Betriebsstrom der Leistungsendstufen auf ca. 2,5 A eingestellt.
- Die Umgebungstemperatur des Controllers während des Betriebes sollte 40 °C nicht überschreiten. Sie sollten darauf achten, dass die Lüftungsschlitze des Controllers im Bodenblech sowie in der Rückwand nicht abgedeckt werden. Ein eventuell hierdurch auftretender Hitzestau führt zur Abschaltung der Leistungselektronik.
- Zur Einhaltung der geforderten EMV-Grenzwerte ist ein möglichst niederohmiger Potentialausgleich von mechanischen und elektronischen Geräten notwendig.
Hierzu ist es sinnvoll, sowohl den Controller als auch die numerischen Achsen auf einen gemeinsamen Erdpunkt zu legen (Leitungsquerschnitt 2,5 mm²).
- Als Verbindungsleitung zwischen Controller und Schrittmotor sind im Lieferumfang des C 116-4 drei konfektionierte 3 bzw. 5 Meter lange Verbindungsleitungen enthalten. Sollten diese Kabel nicht ausreichend lang sein, so können Sie die Leitung bis maximal
 $l = 10 \text{ m}$ verlängern. Hierbei ist der Kabelaufbau gemäß Kapitel 4.3 Absatz 6 zu berücksichtigen.
- Die zusätzlich aus dem Kabelstecker herausgeführten Abschirmungen der Motoranschlussleitungen sind in keinem Falle als Potentialausgleich einzusetzen; sie dienen lediglich zur Funktionserdung der Einheit.
Zum Potentialausgleich ist eine zusätzliche, niederohmige Verbindung vom Controller zu den numerischen Antriebsachsen zu führen.
- Zur Programmierung der Interfacekarte verfügt diese über eine serielle Schnittstelle nach RS-232. Als Schnittstellenanschluss steht frontseitig ein 9-poliger Sub D-Stiftstecker zur Verfügung.
Die Verbindung von Interfacekarte und Steuerrechner ist durch eine 3-polige abgeschirmte Leitung zu realisieren. Die Kontaktbelegung ist der Interfacekarten-Beschreibung zu entnehmen.

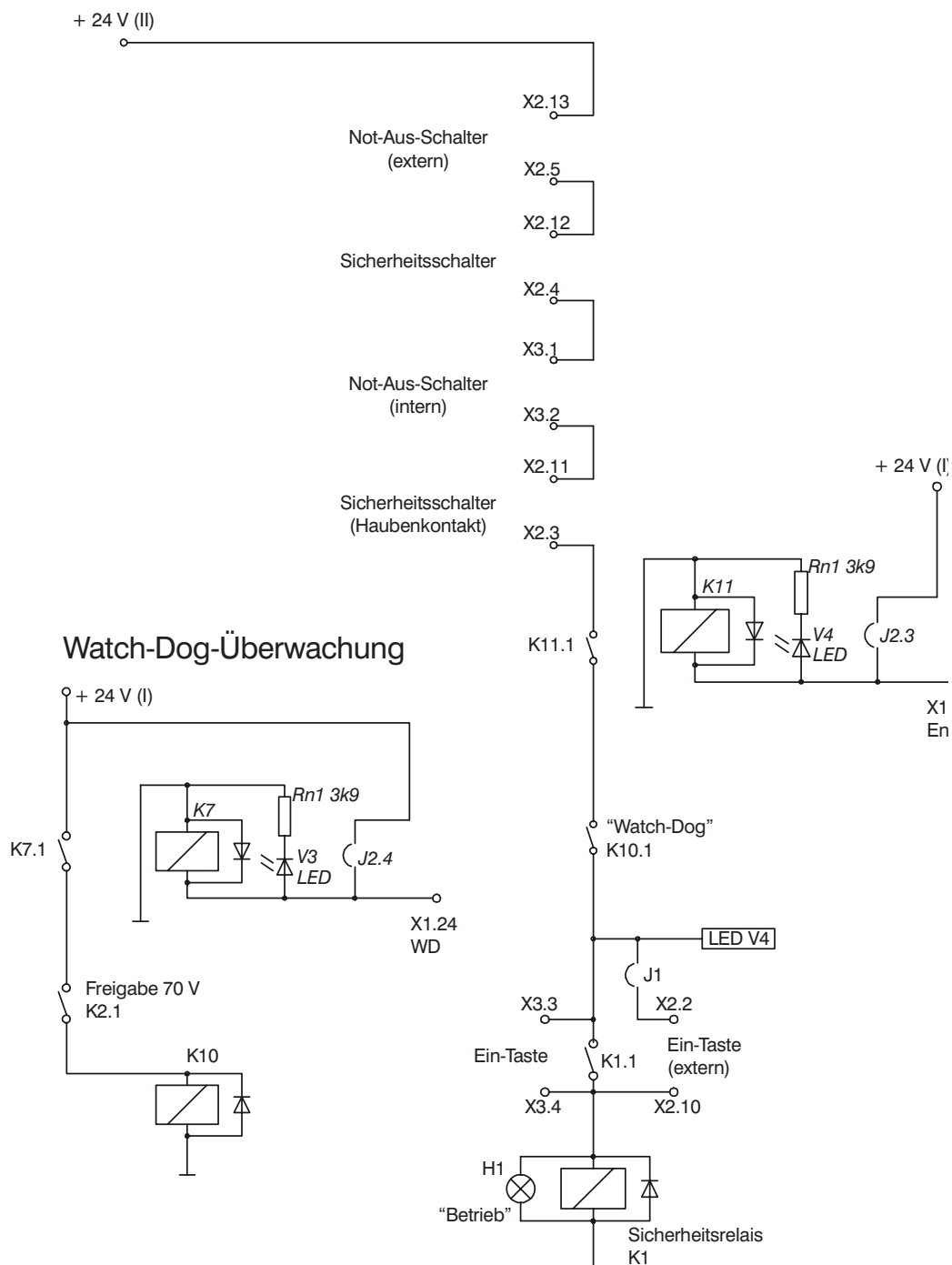
Die im Lieferumfang des Controllers enthaltene Schnittstellenleitung ist ca. 2 Meter lang und verfügt beidseitig über einen Sub D-Buchsenstecker. Da die Pin-Belegung der beiden Steckverbinder nicht identisch ist (keine 1:1-Leitung), besteht die Möglichkeit, die beiden Anschlüsse zu vertauschen. Um dies zu verhindern sind die Steckverbinder farblich gekennzeichnet.

Der rote Steckverbinder ist mit dem Steuerrechner zu verbinden, der graue mit der Interfacekarte. Zusätzlich ist die Rechner-Seite mit einem Aufkleber gekennzeichnet.

- Vor Inbetriebnahme des Controllers sind alle Verbindungsleitungen anzuschließen und die Steckverbinder Remote und X2 zu stecken. Achten Sie darauf, dass auf dem Steckverbinder Remote die Brücke 3-4 (NOT-AUS) und auf dem Steckverbinder X2 die Brücke 2-7 (Stop) geschlossen ist.
Eine fehlende Verbindung 3-4 auf dem Remote-Steckverbinder führt dazu, dass die Spannungsversorgung der Leistungsendstufen nicht eingeschaltet werden kann.
- Schrittmotorantriebe können in bestimmten Betriebszuständen zu Resonanzen neigen, die sich entweder in Schrittverlusten einzelner Achsen oder in besonderen Fällen zum Stillstand (Ausphasen) des Motors führen.
Ursache hierfür ist in Aufbau und Wirkungsweise des Schrittmotors begründet. Die Drehbewegung des Schrittmotors erfolgt durch ein schrittweises weiterschalten des Statorfeldes (Motorspulen). Der magnetisierte Rotor beschleunigt daraufhin, führt die Schrittbewegung aus, schwingt kurz in seine neue Position ein und verharrt dort bis zum nächsten Schrittpuls. Überlagern sich die Schrittpulse mit dem Ausschwingverhalten des Rotors, addieren sich die Kraftvektoren. Die Stärke und Häufigkeit dieser Resonanzerscheinungen ist unter anderem von der mechanischen und elektrischen Eigenschwingung des Motors, der mechanischen Konstruktion und der Verbindung beider Komponenten abhängig.
Da bei interpolierendem Betrieb die Achsgeschwindigkeiten gegeneinander geregelt werden, kann nicht ausgeschlossen werden, dass sich bei bestimmten Vektoren systembedingte Resonanzen ergeben. Diese können durch folgende Maßnahmen gemindert werden:
 - Höhere Beschleunigungsrampen um die Aufenthaltsdauer in einem Resonanzbereich während der Beschleunigungs- und Bremsrampe zu minimieren.
 - Einsatz von Magnet- oder Viskosedämpfer als Grundlast (auf die Antriebswelle montiert).
 - Mechanische Entkopplungen durch spezielle Kupplungen mit resonanzdämpfenden Kunststoffteilen.
 - Verwendung von Leistungsendstufen mit höherer Schrittauflösung.
 - Optimierung der Phasenstrom-Einstellung.
- Die Stop-Taste der Impulssteuerung (Steckverbinder X2) ist nur aktiv, wenn auf der Interfacekarte der DIP-Schalter S1.5 auf OFF geschaltet ist. Die Schalter S1.4 und S1.6 müssen in der Schalterstellung ON liegen.

7 Schaltungsunterlagen

7.1 Prinzipschaltbild Sicherheitskreis



8 Konformitätserklärung

gem. EG-Richtlinie Niederspannung sowie elektromagnetische Verträglichkeit.

Dok.Nr.: k301/95

Wir, Firma *isel/automation GmbH & Co. KG*
 Bürgermeister-Ebert-Str. 40
 D- 36142 Eichenzell

erklären in alleiniger Verantwortung, dass das Produkt

Artikelbez.: CNC-Steuerung C 116-4

Artikelnr.: 383 210

auf das sich diese Erklärung bezieht, mit der /den folgenden Norm(en) oder normativen Dokument(en) übereinstimmt.

1. **EN 50081-1; EN 55011 (VDE 0875)**
 - Elektromagnetische Verträglichkeit- Fachgrundnorm Störaussendung
 - Teil 1: Wohnbereich, Geschäfts- und Gewerbebereiche sowie Kleinbetriebe
 - Grenzwerte und Messverfahren für Funkentstörung von industriellen, wissenschaftlichen und medizinischen Hochfrequenzgeräten (Grenzklasse B)
2. **EN 50082-1; IEC 801 (Teil 1-4)**
 - Elektromagnetische Verträglichkeit- Fachgrundnorm Störfestigkeit
 - Teil 1: Wohnbereich, Geschäfts- und Gewerbebereiche sowie Kleinbetriebe
 - Prüf- und Messverfahren der Störfestigkeit
3. **EN 50178 (VDE 0160)**
 Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln

Wir versichern hiermit, dass das Bescheinigungsverfahren ausschließlich gemäß der Richtlinie 73/23/EWG (19.02.73), Änderung 93/86/EWG (22.07.93).

Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedsstaaten betreffend elektrische Betriebsmittel zur Verwendung innerhalb bestimmter Spannungsgrenzen

sowie

Richtlinie 89/336/EWG (03.05.89), Änderung 91/263/EWG (29.04.91), Änderung 2/31/EWG (28.04.92), Änderung 93/68EWG (22.07.93) Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedstaaten über die elektromagnetische Verträglichkeit

durchgeführt wurde und dass die Vorschriften der Norm

DIN EN 45014

Allgemeine Kriterien für Konformitätserklärungen von Anbietern bei der Ausstellung der Konformitätserklärung beachtet wurden.

Eiterfeld, den 24.10.1995



Rainer Giebel, Fertigungsleitung Elektronik

Konformitätserklärung

gem. EG-Richtlinie Niederspannung sowie elektromagnetische Verträglichkeit.

Dok.Nr.: k302/95

Wir, Firma *isel/automation KG*
 Im Leibolzgraben 16
 D- 36132 Eiterfeld

erklären in alleiniger Verantwortung, dass das Produkt

Artikelbez.: CNC-Steuerung C 116-4

Artikelnr.: 383 211

auf das sich diese Erklärung bezieht, mit der /den folgenden Norm(en) oder normativen Dokument(en) übereinstimmt.

1. **EN 50081-1; EN 55011 (VDE 0875)**
 - Elektromagnetische Verträglichkeit- Fachgrundnorm Störaussendung
 - Teil 1: Wohnbereich, Geschäfts- und Gewerbebereiche sowie Kleinbetriebe
 - Grenzwerte und Messverfahren für Funkentstörung von industriellen, wissenschaftlichen und medizinischen Hochfrequenzgeräten (Grenzklasse B)
2. **EN 50082-1; IEC 801 (Teil 1-4)**
 - Elektromagnetische Verträglichkeit- Fachgrundnorm Störfestigkeit
 - Teil 1: Wohnbereich, Geschäfts- und Gewerbebereiche sowie Kleinbetriebe
 - Prüf- und Messverfahren der Störfestigkeit
3. **EN 50178 (VDE 0160)**
 Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln

Wir versichern hiermit, dass das Bescheinigungsverfahren ausschließlich gemäß der Richtlinie

73/23/EWG (19.02.73), Änderung 93/86/EWG (22.07.93).

Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedsstaaten betreffend elektrische Betriebsmittel zur Verwendung innerhalb bestimmter Spannungsgrenzen

sowie

Richtlinie 89/336/EWG (03.05.89), Änderung 91/263/EWG (29.04.91), Änderung 2/31/EWG (28.04.92), Änderung 93/68/EWG (22.07.93) Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedstaaten über die elektromagnetische Verträglichkeit

durchgeführt wurde und dass die Vorschriften der Norm

DIN EN 45014

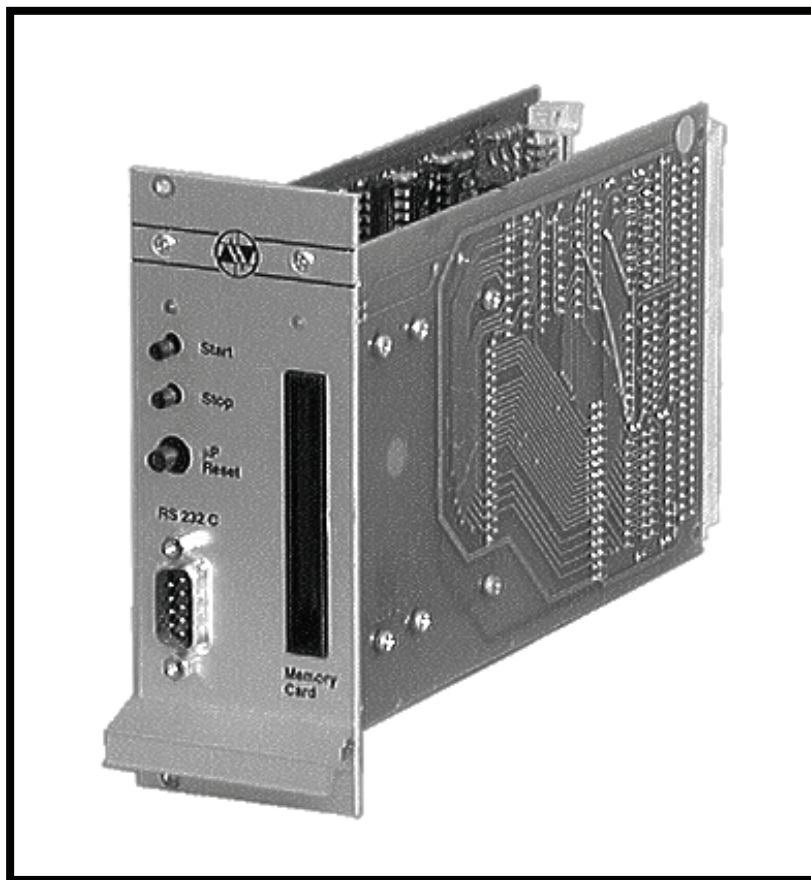
Allgemeine Kriterien für Konformitätserklärungen von Anbietern bei der Ausstellung der Konformitätserklärung beachtet wurden.

Eiterfeld, den 24.10.1995



Rainer Giebel, Fertigungsleitung Elektronik

isel-Interfacekarten-Serie



Hardware-Beschreibung

B.325xxx.03/2000.12

Diese Dokumentation gilt für folgende Baugruppen:

Art.-Nr.:	325 000	-	Interfacekarte UI 4.0
Art.-Nr.:	325 001	-	Interfacekarte UI 4.C
Art.-Nr.:	325 500	-	Interfacekarte UI 4.0-E/A
Art.-Nr.:	325 501	-	Interfacekarte UI 4.C-E/A
Art.-Nr.:	325 050	-	Interfacekarte UI 5.0
Art.-Nr.:	325 051	-	Interfacekarte UI 5.C
Art.-Nr.:	325 550	-	Interfacekarte UI 5.0-E/A
Art.-Nr.:	325 551	-	Interfacekarte UI 5.C-E/A

Unterschiede der Prozessorkarte liegen nur im eingesetzten Betriebssystem und dem Befehlsumfang der Karte sowie der Taktfrequenz des Prozessors. Eine Übersicht der jeweils nutzbaren Befehle ist in der Programmieranleitung 'CNC-Betriebssystem 5.x' enthalten.

In dieser Anleitung finden Sie verschiedene Symbole, die Ihnen schnell wichtige Informationen anzeigen.

Gefahr



Achtung



Hinweis



Beispiel



Zusatz-Infos



© Fa. **iselautomation** 1998

Alle Rechte vorbehalten

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



isel-Maschinen und Controller sind CE-konform und entsprechend gekennzeichnet. Für alle sonstigen Maschinenteile und -komponenten, auf die CE-Sicherheitsrichtlinien anzuwenden sind, ist die Inbetriebnahme solange untersagt, bis alle entsprechenden Anforderungen erfüllt sind.



Die Firma **iselautomation** übernimmt keine Gewähr, sobald Sie irgendwelche Veränderungen an dem Gerät vornehmen.



Die in der Konformitätserklärung aufgeführten Grenzwerte gelten nur für die ab Werk gelieferte Originalkonfiguration.

Hersteller: Fa. **iselautomation GmbH & Co. KG**
Bürgermeister-Ebert-Str. 40
D-36142 Eichenzell

Fax: (06672) 898-888
e-mail: automation@isel.com
<http://www.isel.com>

Inhaltsverzeichnis

1	Einleitung	4
2	Technische Daten	5
3	Systembeschreibung	6
3.1	Bedienelemente	6
3.2	Serielle Schnittstelle	7
3.3	Funktionselemente	8
3.3.1	Einstellung DIP-Schalter S1 (Baudrate)	9
3.3.2	Einstellung der Beschleunigung	9
3.3.3	Einstellung Voll-/Halbschrittbetrieb (Dip-Schalter S2) (Option)	10
3.3.4	Aktivierung Endlagen-/ Überfahrschalter (Dip-Schalter S3)	10
3.4	Programmier-Modus	11
3.5	Spannungsversorgung	11
3.6	Betriebsstörungen	11
4	Anschluss und Inbetriebnahme	12
4.1	Steckverbinder	13
4.1.1	Signaleingänge	14
4.1.1.1	Referenz-Schalter (Ref.Sw.)	14
4.1.1.2	Überfahrschalter (Stop)	15
4.1.1.3	Start (P1.0)	15
4.1.1.4	µP-Reset	15
4.1.1.5	Signalausgänge	16
4.1.1.6	Betriebsart Voll-/Halbschritt (V/H)	16
4.1.1.7	Taktabstaltung	16
4.1.1.8	Takt	16
4.1.1.9	Richtung	16
4.1.1.10	Stromabsenkung	17
4.1.1.11	Bremse	17
4.1.2	Datenspeicher	17
5	Optionen und Erweiterungen	18
5.1	Aufrüstmöglichkeiten	18
5.2	Optionen	18
6	E/A-Erweiterung	19
6.1	Steckerleiste	20
6.2	Signalankopplung	21
6.2.1	Signaleingänge	22
6.2.2	Signalausgänge	23
6.3	Externer Datenspeicher	24
7	Software-Treiber I5DRV	24

1 Einleitung

isel-Interfacekarten sind Prozessorkarten mit einem ausgereiften CNC-Betriebssystem zur Steuerung von bis zu drei Schrittmotoren. Als Euro-Einschub mit 1" Breite (5 TE) und 3 HE Höhe sind sie in allen 19"-Systemen einsetzbar.

- Die Interfacekarte basiert auf einem 8-Bit-Mikro-Controller-System mit 32 kB Betriebs-EPROM und 32 kB Datenspeicher. Eine umfangreiche, praxisorientierte CNC-Betriebssoftware garantiert die einfache Programmierbarkeit.
- Zur Programmierung von Bewegungsabläufen stehen dabei unter anderem Befehle zur relativen und absoluten Positionierung von bis zu drei Schrittmotoren, Nullpunktfahrt und virtuelle Nullpunkte zur Verfügung. Hierbei wird eine lineare 3D-Interpolation genau so unterstützt wie eine zirkulare Interpolation von zwei aus drei Achsen.
- Die maximal erreichbaren Positionier-Geschwindigkeiten liegen zwischen 30 und 10 000 Schritten/Sekunde. Der Wertebereich beträgt dabei 24 Bit, d. h. eine maximale Wegauflösung von $\pm 8\,000\,000$ Schritten. Zur Ablaufsteuerung stehen die Befehle schachtelbare Schleifen, erzwungene Verzweigungen, Zeitverzögerungen usw. zur Verfügung.
- Darüber hinaus erleichtern einige Hilfsfunktionen den Umgang mit der umfangreichen Software, so z. B. Einzelschrittausführung (Trace-Mode), Positionsrückmeldungen, Ändern der Gerätenummer und Auslesen von Speicherzellen.
- Durch Direktausführung (DNC-Betrieb) oder Speicherbetrieb (CNC-Betrieb) der Befehle sind sowohl Stand-Alone-Applikationen als auch Anwendungen mit Leitrechnern realisierbar.
- Zur Speicherung von Systemvariablen und CNC-Programmen steht ein 32 kB-Datenspeicher zur Verfügung. Durch Einbau eines optionalen Akku wird eine quasi-permanente Speicherung der CNC-Programme möglich.
- Zur Ansteuerung von Schrittmotorleistungsendstufen erzeugen isel-Interfacekarten Signale für Takt, Richtung, Stromabsenkung während Motorstillstand, Takt-Stop und Voll-/Halbschrittschaltung.
- Die Signalpegel sind TTL-kompatibel (+ 5 V-Logik). Ausgangstreiber ermöglichen den parallelen Betrieb mehrerer Leistungsendstufen. Alle Steuersignale werden an der Kartenrückseite über einen 64-poligen Steckverbinder nach DIN 41612 Bauform C geführt.
- Die Programmierung der Interfacekarte sowie die Kommunikation mit anderen Rechnersystemen ist über eine serielle Schnittstelle mit Software-Handshake und 256 Byte Pufferbereich realisiert. Sie ermöglicht eine zuverlässige 3-Draht-Verbindung zu Steuerrechnern, wobei Baudraten von 2 400 Bd bis 19 200 Bd über DIP-Schalter umschaltbar sind.
- Als Bedienelemente sind in der Frontplatte der Interfacekarten Start-, Stop- sowie Not-Aus-Taster integriert. Die Betriebsbereitschaft wird durch eine LED angezeigt.

2 Technische Daten

Abmessungen	Euro-Karte, 100 x 160 mm, Frontplatte 5 TE (1")	
Spannungsversorgung	+ 5 V, $\pm 5\%$, 300 mA (auf + 6 V bis + 12 V umrüstbar)	
Steckverbinder	DIN 41612 Bauform C, 64-polig a + c	
Eingänge	Rechner-Reset	(aktiv-low)
	Referenz-Schalter	(Schmitt-Trigger)
	Überfahrschalter	(Schmitt-Trigger)
Ausgänge	Takt	(3-State-Output)
	Stromabsenkung	(3-State-Output)
	Richtung	(3-State-Output)
	Taktabschaltung	
	Voll-/Halbschritt	
	Portausgang/-eingang	(P1.0)
Datenübertragung	RS 232 C (9-poliger Sub D-Stiftstecker)	

3 Systembeschreibung

3.1 Bedienelemente



Bild 1: Interfacekarte

Betriebs-LED

... leuchtet bei Betriebsbereitschaft der Prozessorkarte.

Start-Taste

... startet die Ausführung eines im Datenspeicher abgelegten CNC-Datenfeldes.
In Verbindung mit dem μ P-Reset-Taster wird ein Selbsttest der Prozessorkarte gestartet.

Stop-Taste

... unterbricht die Ausführung einer programmierten Bewegung durch Einleiten einer Bremsrampe. Der unterbrochene Prozess kann mit der Start-Taste bzw. dem Befehl '@0S' fortgesetzt werden.

Not-Aus (μ P-Reset)

... unterbricht, bedingt durch einen Prozessor-Reset, sofort alle Aktivitäten der Interfacekarte. Darüber hinaus werden durch einen parallelen Schaltkontakt die Signalausgänge 'Taktabschaltung' auf 0 V-Potential gelegt. Eventuell auftretende Schrittfehler der über Leistungsendstufen angeschlossenen Motoren werden ignoriert. Durch Betätigen der μ P-Reset-Taste bei gleichzeitig gedrückter Start-Taste wird ein Selbsttest der Interfacekarte eingeleitet. Bedingt durch die Ausführung der μ P-Reset-Taste als Tast-Rast-Schalter ist zum "Lösen" des Reset-Zustandes und zur Freigabe des Taktabschaltungs-Ausgangs eine zweite Betätigung des Tasters notwendig.

3.2 Serielle Schnittstelle

Zur Datenübertragung zwischen der Interfacekarte und einem Steuerrechner wird eine serielle Schnittstelle nach RS 232 eingesetzt. Die Verbindung ist über eine 3-Draht-Leitung realisiert; ein Software-Protokoll ermöglicht die fehlerfreie Übertragung der ASCII- Zeichen. Dabei ist es notwendig, dass sich beide Systeme an das im Folgenden beschriebene Übertragungsprotokoll halten.

- Der angeschlossene Steuerrechner sendet einen Befehl, der mit einem Zeilenende-Zeichen [chr(13)] abgeschlossen ist.
- Die Prozessoreinheit quittiert die Ausführung bzw. Speicherung des Befehles durch das Quittierungs-Signal '0' [chr(48)] oder meldet einen aufgetretenen Fehler mit einem ASCII-Zeichen ungleich '0' (vgl. CNC-Betriebssystem 5.0 Kapitel *Fehlermeldungen der Prozessorkarten*).

Als Datenübertragungsparameter sind auf der Prozessorkarte folgende Werte festgelegt:

9 600 Baud (einstellbar)
8 Daten-Bit
1 Stop-Bit
no Parity

Zur Überprüfung des korrekten Anschlusses bzw. der Funktion der seriellen Schnittstelle verfügt die Prozessorkarte über eine Selbsttestroutine. Sie wird ausgeführt, wenn Sie die Start-Taste festhalten und die μ P-Reset-Taste kurz betätigen.

Die Interfacekarte überprüft daraufhin ihren Speicherbereich sowie die Schalterstellung des 4-fach-DIP-Schalters. Anschließend werden zum Test des angeschlossenen Schrittmotors einige Taktimpulse ausgegeben. Abgeschlossen wird die Testroutine durch einen permanent gesendeten ASCII-Zeichensatz an der seriellen Schnittstelle.

Durch Betätigen irgendeiner Taste der Rechnertastatur wird dieser Modus abgebrochen und jedes weiterhin von der Prozessorkarte empfangene Zeichen als Echo zurückgesendet.

Der Selbsttestroutine wird durch einen μ P-Reset beendet!

Zur Inbetriebnahme der seriellen Verbindung von Steuerrechner und Interfacekarte kann folgendes Basic-Schnittstellen-Testprogramm verwendet werden.

Schnittstellen-Testprogramm z. B. in GW-Basic:

```
100   open"com1:9600,N,1,RS,CS,DS,CD" as#1
110   if loc(1)0 then print input$ (loc(1),1):
120   a$=inkey$: if a$="" then print #1,a$;;print a$;
130   goto 110
```

Die Pin-Belegung der Steckverbinder

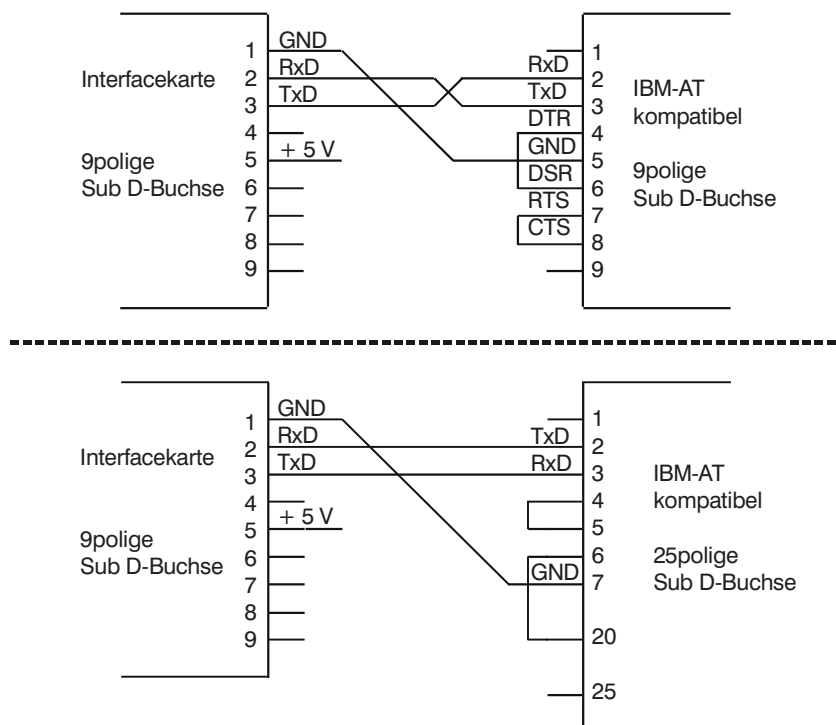


Bild 2: Anschluss serielle Schnittstelle

3.3 Funktionselemente

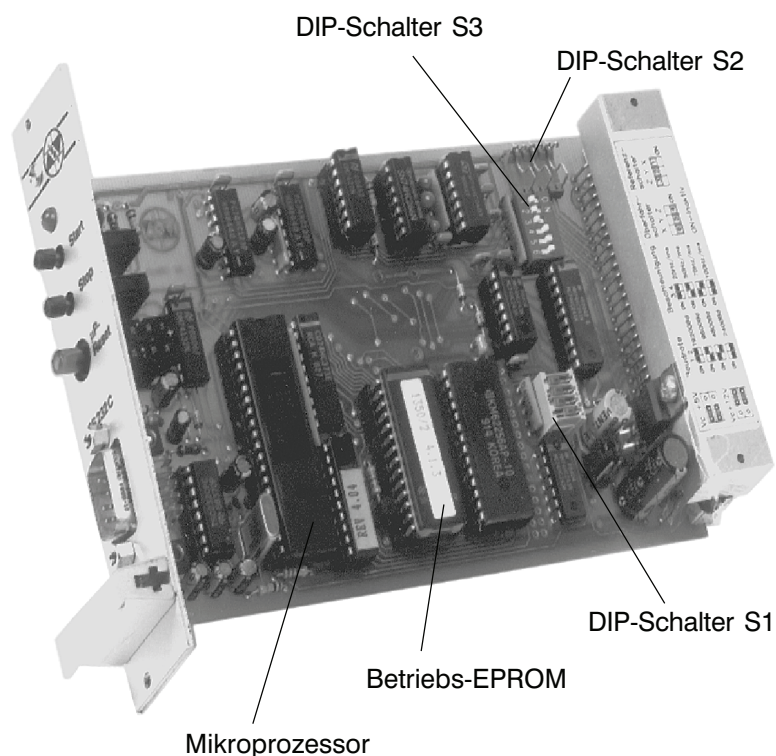


Bild 3: Interfacekarte (ohne E/A-Erweiterung)

3.3.1 Einstellung DIP-Schalter S1 (Baudrate)

Zur Festlegung der Übertragungsrate der seriellen Schnittstelle wird nach jedem Mikroprozessor-Reset die Schalterstellung des 4-poligen Schiebeschalters S1 abgefragt. Dabei ergeben sich aus den vier möglichen Schalterkonfigurationen von Schalter 1 und 2 die unterschiedlichen Baudraten.

S1.1	S1.2	Baudrate
OFF	OFF	2 400 Bd
ON	OFF	4 800 Bd
OFF	ON	9 600 Bd*
ON	ON	19 200 Bd

* Auslieferungszustand 9 600 Bd

3.3.2 Einstellung der Beschleunigung

Bei Betrieb eines Schrittmotors außerhalb des Anlaufbereiches ist eine Beschleunigungs- und Bremsrampe erforderlich. Während bei der Beschleunigungsrampe die Schrittfolgefrequenz des Motors kontinuierlich von der Startfrequenz auf die Betriebsfrequenz gesteigert wird, erfordert die Verzögerungsrampe den umgekehrten Vorgang. Durch unterschiedliche Steigungen lassen sich die Kurven in Bezug auf Beschleunigungszeit und Last optimieren.

Es stehen Ihnen standardmäßig vier verschiedene Rampen zur Verfügung.

Mit Schalter 3 und 4 des 4-poligen DIP-Schalters S1 können Sie die Rampen definieren.

S1.3	S1.4	Rampe
ON	ON	25 Hz/ms
OFF	ON	50 Hz/ms
ON	OFF	75 Hz/ms
OFF	OFF	100 Hz/ms

* Auslieferungszustand 25 Hz/ms

3.3.3 Einstellung Voll-/Halbschrittbetrieb (Dip-Schalter S2) (Option)

Dieser Schalter ermöglicht die zentrale Einstellung der Betriebsart der angeschlossenen Leistungsendstufen.

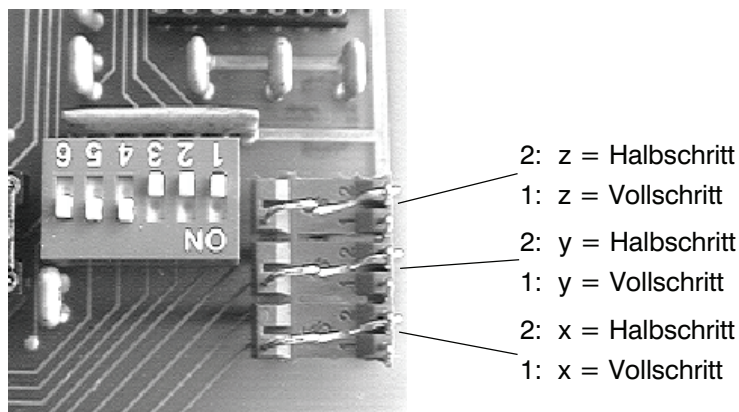


Bild 4: Interfacekarte (Platinenauszug Schalter S2)

Der Schalter S2 wird bei Einsatz der Karte in den Schrittmotor-Controller C 116-4 und C 142-4 nicht ausgewertet. Die Festlegung der Betriebsart wird dort direkt auf der Verbindungsplatine mit Jumper-Steckern vorgenommen.

3.3.4 Aktivierung Endlagen-/ Überfahrschalter (Dip-Schalter S3)

Zur Überwachung von Endlagen- und Überfahrschalter der Schrittmotorantriebs-einheiten werden die Signale der entsprechenden Achsen getrennt auf die Prozessorkarte geführt und dort verarbeitet. Zur Freigabe des Signaleinganges dient der 6-fach-DIP-Schalter S3. Jeder extern zu überwachende Schalter muss durch Umschalten auf OFF aktiviert werden, dementsprechend jeder nicht vorhandene Schalter durch Umschalten auf ON gesperrt werden. Dabei ergibt sich folgende Zuordnung:

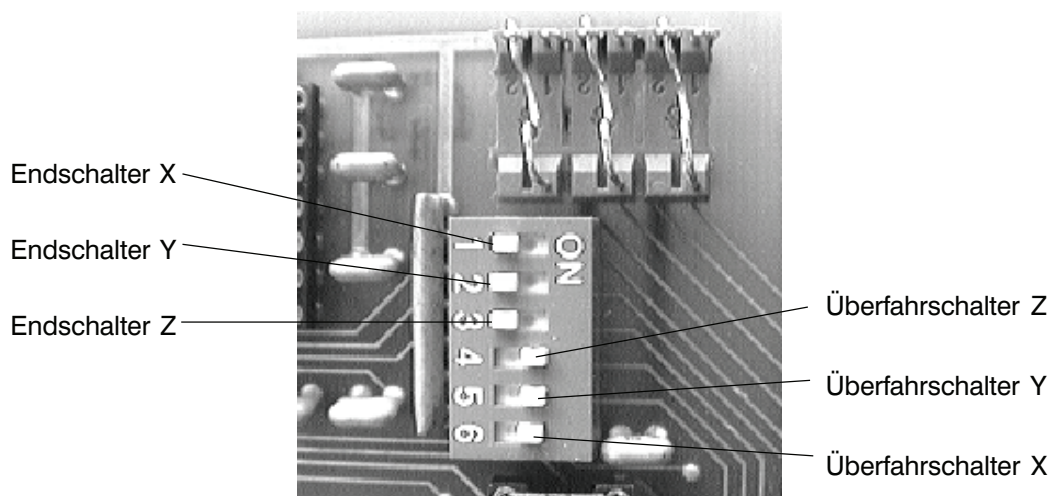


Bild 5: Interfacekarte (Platinenauszug Schalter S3)

3.4 Programmier-Modus

Für einen optimalen Einsatz ermöglicht das Betriebssystem sowohl eine Programmierung im DNC-Modus (direkte Ausführung der übergebenen Befehle) als auch im CNC-Modus (auszuführendes Programm wird im internen Datenspeicher abgelegt und später durch ein Start-Signal gestartet, vgl. CNC-Betriebssystem 5.0).

Im DNC-Modus werden dem Prozessormodul die Bearbeitungsparameter einzeln übergeben und von ihm direkt ausgeführt. Durch Auswertung der Quittierungssignale der IT 108 ist der übergeordnete Steuerrechner in der Lage, kontinuierlich und ohne Begrenzung Daten zu übergeben.

Im CNC-Modus (Speicherbetrieb) wird der Prozessoreinheit ein komplettes Datenfeld übergeben. Die Daten werden nach Erhalt vom Prozessor quittiert und in einem Datenspeicher abgelegt. Die Ausführung des Datenfeldes (ca. 1 800 Befehlssätze) erfolgt anschließend durch Betätigen der Start-Taste bzw. eines Startbefehles des Steuerrechners.

3.5 Spannungsversorgung

Als Spannungsversorgung benötigt die Interfacekarte eine Gleichspannung von + 5 V bei einem mittleren Stromverbrauch von ca. 300 mA. Sie wird über die Steckkontakte a,c30 (+ Vc) und a,c32 (GND) des rückwärtigen Steckverbinders auf die Karte geführt.

Zur Überwachung der Speisespannung befindet sich auf den Prozessorkarte (ab Version 1350/4) eine entsprechende Schaltung, die bei Unterschreiten einer Schwellenspannung den Prozessor zurücksetzt. Dies wird durch gleichzeitiges Verlöschen der Betriebs-LED angezeigt.

Ein DC/DC-Wandler auf der Interfacekarte ermöglicht die Spannungsversorgung mit + 6 V bis + 12 V. Das Umschalten des Eingangsspannung-Levels geschieht durch zwei Jumper (siehe Aufkleber auf dem Steckverbinder der Interfacekarte).

3.6 Betriebsstörungen

Zur Erkennung von Betriebsstörungen verfügt die Interfacekarte hardwaremäßig über einen Unterspannungsdetektor sowie softwaremäßig über Überwachungsmodule für End- und Überfahrschalter sowie über Kommunikations- und Speicherfehler.

Während bei Spannungsfehlern der Mikroprozessor in den Reset-Zustand geschaltet und die Kommunikation zum übergeordneten Rechner abgebrochen wird, erfasst der Prozessor alle anderen Betriebszustände durch das Betriebssystem. Hier erfolgt die Fehleranzeige über die serielle Schnittstelle (Fehlercode vgl. CNC-Betriebssystem 5.0 Kapitel 4, sowie serielle Schnittstelle S. A3).

Fehlercode	Fehlerart	Fehlerbeseitigung
Betriebs-LED leuchtet nicht	<ul style="list-style-type: none"> - keine Versorgungsspannung 7 angelegt - Versorgungsspannung 4,65 - μP-Reset-Eingang (c28) ist aktiv low 	<ul style="list-style-type: none"> - Versorgungsspannung + 5 V/300 mA an Pin 30 (+ 5 V) und Pin 32 (GND) anlegen - Signaleingang μP-Reset überprüfen
LED in μ P-Reset-Taste leuchtet	<ul style="list-style-type: none"> - Tast-Schalter ist nach μP- Reset eingerastet 	<ul style="list-style-type: none"> - durch nochmaliges Betätigen Tast-Schalter lösen
Karte antwortet nicht	<ul style="list-style-type: none"> - Verbindungsleitung der RS 232 nicht korrekt gesteckt. - Serielle Schnittstelle der Interfacekarte defekt - Serielle Schnittstelle des Steuerrechners defekt 	<ul style="list-style-type: none"> - Steckverbinder mit dem Aufkleber 'AT-Seite' mit der seriellen Schnittstelle des PC verbinden. - Schnittstellen-Testprogramm (s. S.6) starten und Selbsttest ausführen. - ggf. seriellen Schnittstellen-Baustein (MAX 232) ersetzen. - Überprüfen der Schnittstelle durch Ankopplung eines anderen Gerätes

4 Anschluss und Inbetriebnahme

Zum Einsatz in 19"-Baugruppenträgern (nach DIN 41494) verfügt die Interfacekarten-Serie über einen 64-poligen Steckverbinder DIN 41612 C. Über ihn werden zum einen alle Signaleingänge der Prozessorkarte zugeführt (z. B. Start-, Stop-, Referenz-Schalter), zum anderen von der Prozessorkarte alle Steuerausgänge zur Verfügung gestellt (z. B. Takt und Richtung).

Bedingt durch die Konzeption als Interpolator für max. drei Schrittmotorantriebe sind auf der Prozessorkarte die entsprechenden Signalein- und -ausgänge für jede Antriebsachse getrennt ausgeführt.

4.1 Steckverbinder

Zur Adaption in 19"-Systemgehäusen verfügt die Interfacekarte über eine 64-polige Stiftleiste nach DIN 41612 Bauform C.

Reihe A			Reihe C	
Signal		Pin	Pin	Signal
NC		1	1	NC
NC		2	2	NC
NC		3	3	NC
NC		4	4	NC
NC		5	5	NC
NC		6	6	NC
V/H X-Achse	A	7	7	NC
V/H Z-Achse	A	8	8	A V/H Y-Achse
Ref.Sw. Y-Achse	E	9	9	E Ref.Sw. X-Achse
NC		10	10	E Ref.Sw. Z-Achse
Taktabschaltung X	A	11	11	A Taktabschaltung Y
Taktabschaltung Z	A	12	12	NC
NC		13	13	NC
+ 5 V**		14	14	Bremse**
RxD*		15	15	TxD*
NC		16	16	A Richtung X-Achse
Takt X-Achse	A	17	17	A Richtung Z-Achse
Takt Z-Achse	A	18	18	A Richtung Y-Achse
Takt Y-Achse	A	19	19	NC
Stromabsenkung Z	A	20	20	A Stromabsenkung Y
Stromabsenkung X	A	21	21	A NC
NC		22	22	NC
NC		23	23	NC
NC		24	24	NC
Stop Z-Achse	E	25	25	E Stop Y-Achse
Stop X-Achse	E	26	26	E P1.0
P1.0	E	27	27	E P1.0
NC		28	28	μP-Reset
NC		29	29	NC
+ 5 V		30	30	+ 5 V
NC		31	31	NC
GND		32	32	GND

NC= nicht belegt

A = Signalausgang

E = Signaleingang

* ab Version AZ1350/3

** ab Version AZ1350/4

4.1.1 Signaleingänge

Als Signaleingänge verarbeitet die Interfacekarte folgende Eingänge:

- Referenz-Schalter (Ref.Sw.)
- Überfahrschalter (Stop)
- Start (Start)
- - μ P-Reset

4.1.1.1 Referenz-Schalter (Ref.Sw.)

Zur Positionsbestimmung innerhalb eines Schrittmotor-Antriebssystems besteht die Notwendigkeit eines Maschinennullpunktes bzw. Referenzpunktes.

Zur Auswertung von entsprechenden Sensoren verfügt die Interfacekarte über den Eingang *Referenz-Schalter (Ref.Sw.)*. Bei dem Eingang handelt es sich um einen aktiv-high-Eingang, der intern über einen Pull-up-Widerstand auf + 5 V gelegt ist. Die Auswertung des Signales erfolgt, wenn auf dem im Ruhezustand GND-Potential führenden Eingang ein + 5 V-Signal auftritt.

In *isel*-Lineareinheiten hat sich als Referenz-/Endlagenschalter ein Mikro-Schalter (Öffner-Schaltkontakt) durchgesetzt, der zwischen GND und Signaleingang *Ref.Sw.* geschaltet ist.

Wird während einer Verfahrbewegung der Referenzschalter betätigt, stoppt die Prozessoreinheit abrupt die Schritimpulsausgabe. Erfolgt eine Aktivierung des Schalters während der Ausführung einer Referenzfahrt, wird die Impulsausgabe ebenfalls unterbrochen, jedoch nach Ändern des Richtungsbits mit einer kleinen Schrittfrequenz wieder gestartet.

Ein erneuter Interrupt (durch Verlassen des Schalterbereiches) stoppt den Schrittmotor exakt am Maschinen-Nullpunkt. Hierbei wird eine Wiederholgenauigkeit von ± 1 Schritt erreicht. Bei Verwendung eines induktiven, kapazitiven oder optischen Näherungsschalters ist der Minus-Pol des Sensors mit dem GND-Signal der Antriebseinheit sowie der Signalausgang des Sensors (open-collector) mit dem Steuerungseingang *Ref.Sw.* zu verbinden.



- als Sensor muss ein NPN-Typ eingesetzt werden
- der Sensor muss als Öffner arbeiten (Ruhezustand Ausgang leitend)

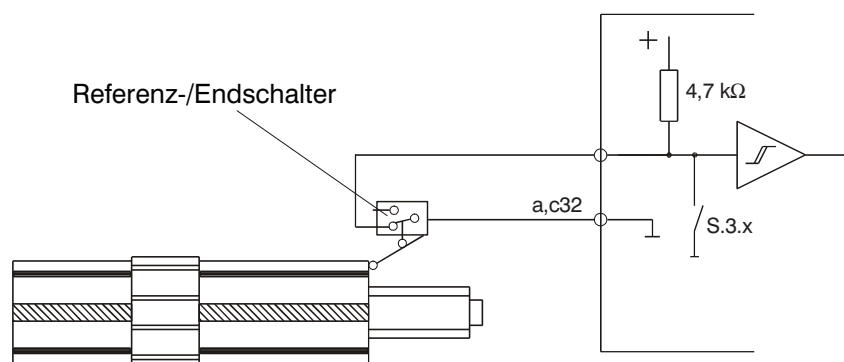


Bild 6: Anschluss Referenzschalter

Bei nicht oder nicht korrekt angeschlossenem Referenz-Schalter meldet die Interfacekarte über die serielle Schnittstelle Fehler '2'.

Bedingt durch die begrenzte Anzahl von Hardware-Interrupts werden auf der Interfacekarte die Signalquellen der drei Referenzschalter-Eingänge miteinander verknüpft. Hierzu sind die Signaleingänge an eine Impulsformungsstufe geführt, die aus jeder Flankenänderung eines Eingangssignales einen definierten Impuls mit 10 μ s Impulsbreite erzeugt.

Werden einzelne Referenzschalter nicht benötigt bzw. angeschlossen, ist der entsprechende Signaleingang direkt auf GND-Potential zu legen oder - wie in Absatz 3.4.4 beschrieben, mit Hilfe des DIP-Schalter S3 zu sperren.

4.1.1.2 Überfahrschalter (Stop)

Dieser Eingang führt, genauso wie bei Betätigung des frontseitigen Stop-Tasters, zu einem Stop-Interrupt des CNC-Betriebssystems. So veranlasst ein negativer Impuls (H-L-Signal-wechsel) am Signaleingang einen gebremsten Abbruch einer Verfahroutine.

Einsatzmöglichkeiten dieses Einganges sind z. B. in Verbindung mit Referenzschaltern geringer Schalthysterese zu sehen (mechanische Zerstörung durch Nachlaufweg des Schrittmotors bei abrupten Reset mit hoher Geschwindigkeit). Ähnlich dem Signaleingang *Ref.Sw.* werden auch die Überfahrschalter-Eingänge zu einem Interrupt zusammengefasst, sodass die Aktivierung eines Einganges den Bewegungsablauf aller aktiven Schrittmotor-achsen unterbricht.

Zu beachten ist hierbei, dass ein solchermaßen unterbrochener Bewegungsablauf mit der Start-Taste reaktiviert werden kann und ein kontinuierlich offener Signaleingang einen erneuten Interrupt verhindert. Sie sollten deshalb darauf achten, dass ein Überfahrschalter-Eingang nur durch einen kurzen negativen Impuls beschaltet wird.

Analog zum *Ref.Sw.*-Eingang sind auch beim Überfahrschalter-Eingang einzelne, nicht benötigt Signaleingänge direkt auf GND-Potential zu legen oder, wie in Absatz 3.4.4 beschrieben, mit Hilfe des DIP-Schalter S3 zu sperren.

4.1.1.3 Start (P1.0)

Der Signaleingang arbeitet parallel zur frontseitigen Start-Taste. Durch kurzzeitiges Verbinden mit dem GND-Potential wird ein in der Steuerung gespeichertes Programm gestartet.

4.1.1.4 μ P-Reset

Der Steuerungseingang μ P-Reset liegt schaltungstechnisch parallel zum frontseitigen μ P-Reset-Tast-Rast-Schalter. Durch Verbinden des Eingangs mit GND-Potential wird der Mikroprozessor gesperrt und somit alle Aktivitäten unterbrochen. Hierbei werden Positioniervorgänge der angeschlossenen Schrittmotoren abrupt beendet.

4.1.1.5 Signalausgänge

Zur Ansteuerung von Schrittmotor-Leistungsendstufen stellt die Interfacekarte zur Verfügung:

- Betriebsart Voll-/Halbschritt (V/H)
- Taktabschaltung
- Takt
- Richtung
- Stromabsenkung
- Bremse

4.1.1.6 Betriebsart Voll-/Halbschritt (V/H)

Je nach Schalterstellung des 3-poligen DIP-Fix-Schalters liegt an den entsprechenden Signalausgängen entweder + 5 V- oder 0 V-Potential.

Schalterstellung 1 (0 V)	- Vollschrittbetrieb
Schalterstellung 2 (+ 5 V)	- Halbschrittbetrieb

Zur Zuordnung der jeweiligen Schalter siehe Kapitel 3.4.3.

4.1.1.7 Taktabschaltung

Der Signalausgang stellt eine zusätzliche Sicherheit bei einem Hardware-Reset der Interfacekarte dar. Durch Betätigen der frontseitigen μ P-Reset-Taste werden neben dem Reset-Impuls für den Mikro-Controller die drei Signalausgänge auf 0 V-Potential geschaltet. In isel-CNC-Controllern ist dieser Ausgang auf den jeweiligen Takt-Stop bzw. Reset-Eingang der Schrittmotor-Leistungsendstufe gelegt und bewirkt ein zusätzliches Sperren der Taktverarbeitung.

4.1.1.8 Takt

Am Taktausgang der Interfacekarte stehen - entsprechend des im Mikro-Controller berechneten Frequenzverlaufes der einzelnen Schrittmotoren - die jeweiligen Takte für die Leistungsendstufen zur Verfügung. Als Taktimpuls ist ein positiver Impuls von ca. 10 μ s Breite definiert.

4.1.1.9 Richtung

Der Richtungsausgang gibt je nach vorgegebener Drehrichtung des Schrittmotors ein + 5 V-Signal (Drehrichtung CCW) oder ein 0 V-Signal (Drehrichtung CW) aus.

4.1.1.10 Stromabsenkung

Zur Reduzierung der Temperaturentwicklung von Schrittmotor und Leistungsendstufen verfügen Schrittmotor-Endstufen über eine integrierte Phasenstrom-Reduzierung im Stillstand. Dieses Merkmal kann jedoch zu Problemen bei der Bearbeitung im X-Y-Z-Betrieb zweier oder mehrerer Schrittmotorachsen führen.

Sind z. B. während des Fräsbetriebes einer Achse die Schneidkräfte des Werkzeuges höher als die Halte- bzw. Stillstandskräfte des zweiten nicht bewegten Schrittmotor-Achsantriebes, kann diese Achse aus ihrer Ruheposition bewegt werden und einen undefinierbaren Versatz erfahren. Diese ungewollte Eigenschaft kann umgangen werden, indem während der Bearbeitung alle Achsen den vollen Betriebsstrom zur Verfügung gestellt bekommen. Aus diesem Grunde verfügt die Interfacekarte über einen Steuerausgang zur definierten Aktivierung der Stromabsenkungslogik innerhalb der Endstufen.

4.1.1.11 Bremse

Zur Steuerung einer Haltebremse in Schrittmotor-Systemen unterstützt die Interfacekarte ab Version AZ1350/4 die Ansteuerung eines entsprechenden Steuerrelais. So können Magnetbremsen gezielt ein- und ausgeschaltet werden.

In *isel*- Antriebseinheiten werden Magnetbremsen verwendet, die im Ruhezustand aktiv sind. Diese werden nach dem Power-On-Reset der Interfacekarte über ein Steuerrelais mit + 24 V Betriebsspannung versorgt und so geöffnet (inaktiv).

Je nach Applikation kann die Bremse im Direktmodus des CNC-Betriebssystems programmiert werden.



Die Signalausgänge *Takt*, *Richtung*, *Stromabsenkung* und *Bremse* sind über einen 20 mA-Leistungstreiber geführt.

4.1.2 Datenspeicher

Zur Speicherung von systembedingten Variablen und programmierten Funktionsabläufen im CNC-Betrieb verfügen die Interfacekarten über ein 32 kB statisches RAM.

Da dieser Speicher nach Wegfall der Versorgungsspannung die gespeicherten Informationen verliert, ist ggf. in Stand-Alone-Applikationen eine Pufferung der Versorgungsspannung des RAM notwendig. Hierzu verfügt die Interfacekarte optional über eine 100 mAh Akku mit 3,6 V Ausgangsspannung. Ein spezieller Schaltkreis überwacht das Unterschreiten der Versorgungsspannung 4,75 V und sperrt ggf. den Prozessor durch einen Reset-Signal.

5 Optionen und Erweiterungen

5.1 Aufrüstmöglichkeiten

UI 4.0	—>	UI 4.0-E/A	Art.Nr. 328010
UI 4.0	—>	UI 5.0	Art.Nr. 328020
UI 4.0	—>	UI 5.0-E/A	Art.Nr. 328030
UI 4.0-E/A	—>	UI 5.0-E/A	Art.Nr. 328040
UI 5.0	—>	UI 5.0-E/A	Art.Nr. 328050
ab Version UI 4.0	—>	UI 5.C-E/A	Art.Nr. 325551

5.2 Optionen

Programmwahleinheit	Art.Nr. 318110
Akku zur RAM-Pufferung	Art.Nr. 328120
Hand-Terminal UHT1	Art.Nr. 328200
Memory-Card Datenspeicher 32 kByte	Art.Nr. 440114

6 E/A-Erweiterung

Die *isel*-E/A-Erweiterung ist ein Zusatzprodukt zur Interfacekarten-Serie und rundet mit ihren Funktionsblöcken den Bereich 'Schrittmotorantriebe in der Automatisierungstechnik' ab. Sie erweitert den Funktionsumfang der Prozessorkarte um acht Signalein- und 16 Signalausgänge sowie um einen austauschbaren Datenspeicher (Memory-Card).



Bild 7: E/A-Erweiterung (montiert auf Interfacekarte und Signalankopplung)

Die E/A-Erweiterung besteht aus einer 100 x 160 mm großen Baugruppe zur Signalverarbeitung und einem Signal-Ankopplungsmodul. Während die Signalverarbeitung direkt mit der Interfacekarte verbunden ist, verfügt die Signalankopplung über eine eigene Frontplatte.

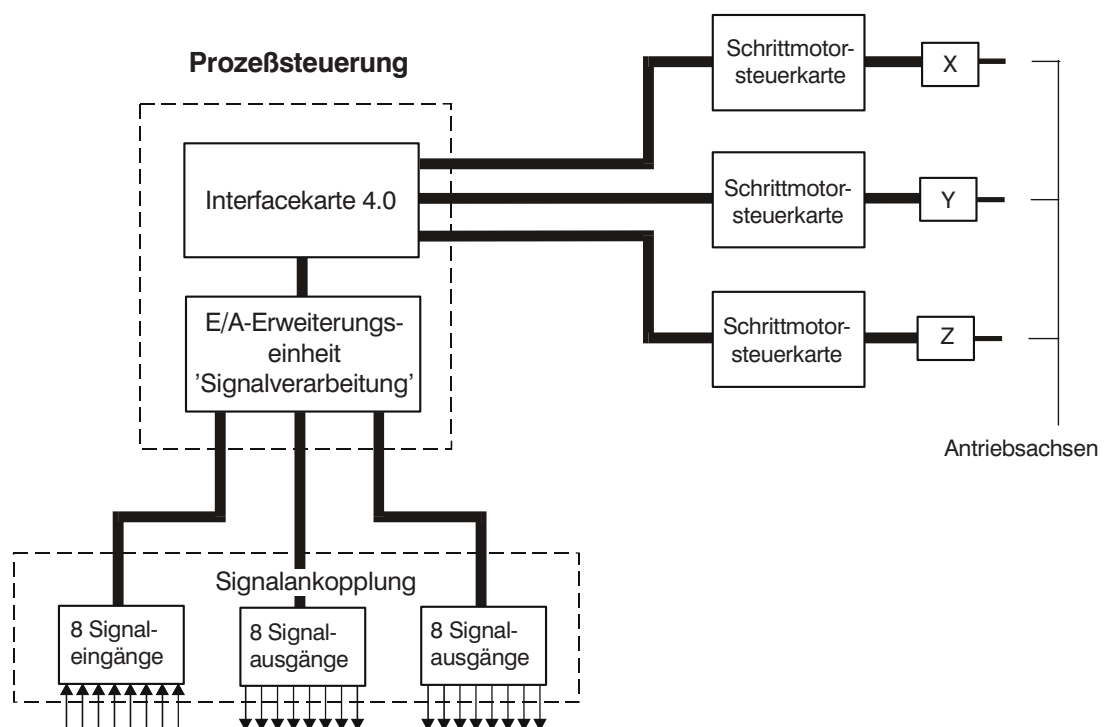


Bild 8: Funktionsblöcke der E/A-Erweiterungseinheiten

6.1 Steckerleiste

Zur Adaption in 19"-Systemgehäusen verfügt die Erweiterungseinheit über eine 64-polige Stiftleiste nach DIN 41612 Bauform C.

		Reihe A		Reihe C	
Signal	Pin	Pin		Signal	
GND	1	1	E	GND	
NC	2	2		NC	
Vcc (+ 5 V)	3	3	E	Vcc (+ 5 V)	
NC	4	4		NC	
NC	5	5	E	In 1.1	
NC	6	6	E	In 1.2	
NC	7	7	E	In 1.3	
NC	8	8	E	In 1.4	
NC	9	9	E	In 1.5	
NC	10	10	E	In 1.6	
NC	11	11	E	In 1.7	
NC	12	12	E	In 1.8	
NC	13	13		NC	
NC	14	14	A	Out 1.8	
NC	15	15	A	Out 1.7	
NC	16	16	A	Out 1.6	
NC	17	17	A	Out 1.5	
NC	18	18	A	Out 1.4	
NC	19	19	A	Out 1.3	
NC	20	20	A	Out 1.2	
NC	21	21	A	Out 1.1	
NC	22	22		NC	
NC	23	23		NC	
NC	24	24	A	Out 2.8	
NC	25	25	A	Out 2.7	
NC	26	26	A	Out 2.6	
NC	27	27	A	Out 2.5	
Reset E	28	28	A	Out 2.4	
NC	29	29	A	Out 2.3	
NC	30	30	A	Out 2.2	
NC	31	31	A	Out 2.1	
GND	32	32		GND	

NC= nicht belegt

A = Signalausgang

E = Signaleingang

6.2 Signalankopplung

Die Signalankopplung ermöglicht den einfachen Anschluss von externen Sensoren, Relais, elektromagnetischen Ventilen etc. über Schraub-Klemm-Steckverbinder. Die notwendige Versorgungsspannung von + 24 V ist extern zur Verfügung zu stellen und an den Klemmen + 24 V bzw. GND anzulegen.

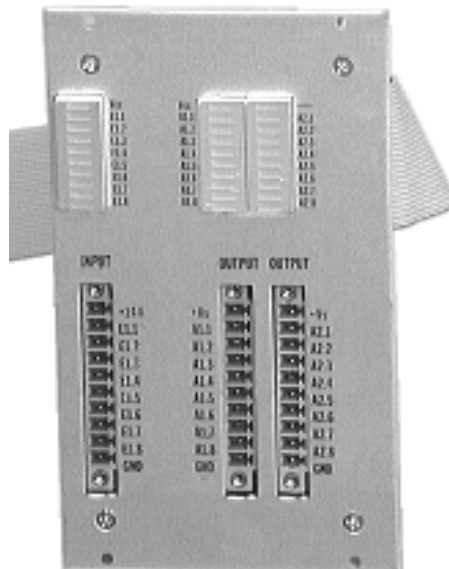


Bild 9: Signalankopplung

6.2.1 Signaleingänge

Die E/A-Erweiterung stellt dem Anwender 8 optoisolierte Signaleingänge zur Verfügung. Entsprechend nachfolgender Zeichnung sind die Eingänge mit einer 12 V-Z-Diode sowie einem Vorwiderstand beschaltet. Hieraus ergibt sich eine Signaleingangsspannung von + 24 V.

Zur optischen Kontrolle der belegten Eingänge stehen LED's zur Verfügung.

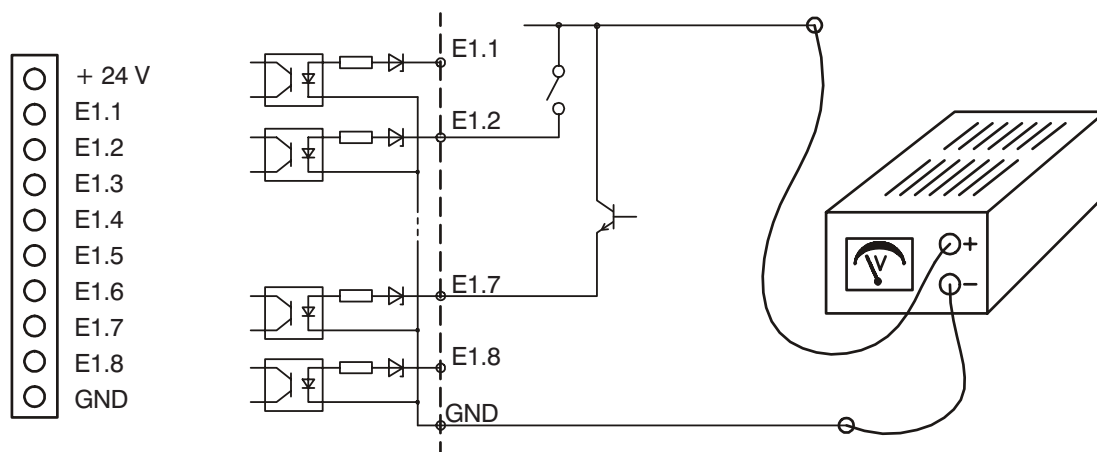


Bild 10: Signaleingänge der E/A-Erweiterung

Die Verarbeitung der Eingänge erfolgt über das Auslesen der Portadresse (65531). Hierzu stehen der Interfacekarte sowohl im DNC- als auch im CNC-Modus entsprechende Befehle zur Verfügung.

DNC-Modus @0b65531
(Auslesen des Eingangsports, byteweise)

CNC-Modus	0	65531	5	1	3	
						Vorwärtssprung um 3 Befehlszeilen (- 5 = Rückwärtssprung um 5 Zeilen)
						Abfrage ob Signaleingang aktiv (1 = Signaleingang aktiv) (0 = Signaleingang inaktiv)
						Abfrage Signaleingang 5
						Adresse des Signaleinganges
						Befehlswort 'Lesen'

6.2.2 Signalausgänge

Die Signalausgänge der E/A-Erweiterung sind als Relais-Schaltausgänge ausgeführt. Die dabei verwendeten Relais erlauben eine maximale Belastung von 50 V bei 300 mA Laststrom. Bedingt durch die 8-Bit-Speicherstruktur der Interfacekarte sind die 16 Ausgänge in zwei 8-bit-Ports unterteilt. Die jeweiligen Port-Adressen sind:

Port A1.1 ... A1.8 Adresse 65529

Port A2.1 ... A2.8 Adresse 65530

Zur optischen Kontrolle verfügt die Signalankopplung über LED-Balkenanzeigen, die bei gesetztem Ausgang leuchten.

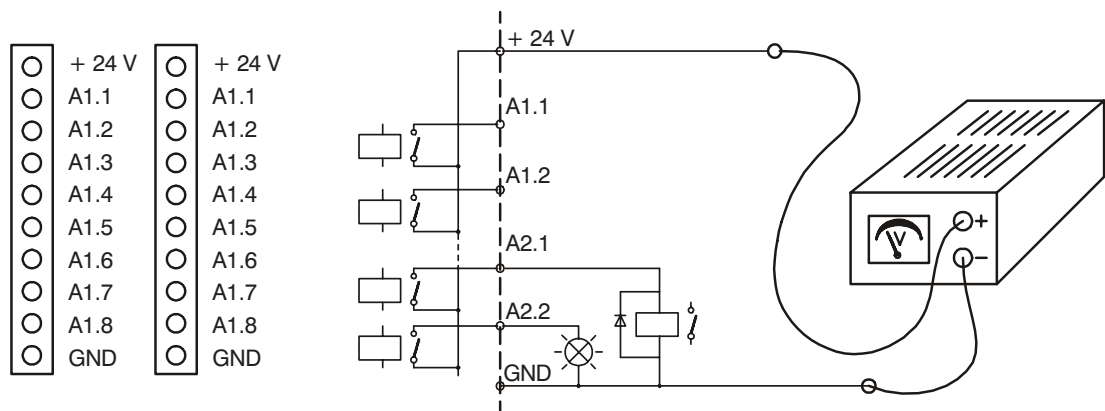


Bild 11: Signalausgänge der E/A-Erweiterung

Die Verarbeitung der Ausgänge wird von der Interfacekarte entsprechend ihrer Programmierung entweder bit- oder byteweise vorgenommen.

DNC-Modus @0b65529,16

(Setzen des Ausgangs 1 mit dem Binärwert 16)

@0b65530,128

(Setzen des Ausgangs 2 mit dem Binärwert 128)

CNC-Modus	p	65530	5	1	
					Ausgang setzen
					(1 = Signalausgang setzen)
					(0 = Signalausgang löschen)
					Ausgang Bit 5
					Adresse des Signalausganges
					Befehlswort 'Schreiben'

6.3 Externer Datenspeicher

Zur externen Speicherung eines Datenfeldes unterstützt die Interfacekarte in Verbindung mit der E/A-Erweiterung den Einsatz eines Scheckkarten-Speichers.

Die Speicherkarte (Memory-Card) mit 32 kB RAM-Speicher und integrierter Batterie wird durch den Befehl @0u mit dem kompletten Inhalt des Interfacekarten-RAM geladen und kann jederzeit durch Betätigen des frontseitigen μ P-Reset-Tasters in das RAM zurückgeschrieben werden.

7 Software-Treiber I5DRV

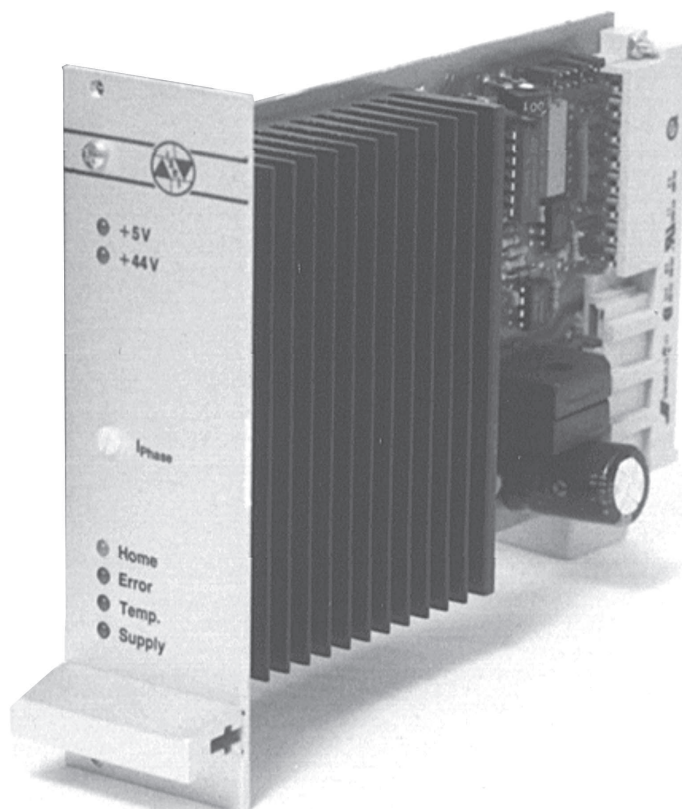
Im Lieferumfang der Interfacekarte ist ab der Version UI5.C die Diskette *isel-I5DRV* enthalten. Dieser Softwaretreiber, der nach dem Laden resident im Hauptspeicher des Steuerrechners bleibt und ab diesem Zeitpunkt für Sie solche Arbeiten wie z. B. die Interpolation und die Kommunikation der Achsenbewegungen, die Verwaltung des Systems, die Kommunikation mit der Hardware etc. übernimmt.

Die Funktionalität des Treibers wird in einer gesonderten Beschreibung 'isel-Treiber für isel-Interfacekarte' behandelt.

Die Beschreibung ist ebenfalls im Lieferumfang enthalten.



Die Programmierung der Interfacekarte mittels der Software PAL-PC ist durch die zusätzliche Funktion nicht eingeschränkt, d. h. bereits erstellte Programme für die Interfacekarte sind voll lauffähig.



Die Zweiphasen-Schrittmotor-Steuerkarte eignet sich zur bipolaren Ansteuerung von 2(4)-Phasen-Schrittmotoren. Sie ist einsetzbar in Baugruppenträgern nach DIN 41494 (19"-Einschubtechnik).

Die Karte erzeugt aus externen digitalen Takt- und Richtungssignalen eine synchrone Steuersequenz, die eine taktproportionale Drehbewegung des angeschlossenen Schrittmotors zur Folge hat.

Eine Stromregelung nach dem Konstant-Strom-Prinzip mit einer Chopperfrequenz von ca. 20 kHz ermöglicht ein optimales Laufverhalten des Schrittmotors auch bei höheren Frequenzen.

Den Phasenstrom (Betriebsstrom des Schrittmotors) können Sie mit Hilfe eines Trimm-Potentiometers (I_{Phase})

dem effektiven Leistungsverbrauch des angeschlossenen Schrittmotors angleichen.

Eine integrierte Überstromabschaltung vermeidet bei einem Kurzschluss die Zerstörung der Leistungsstufe. Zusätzlich überwacht ein Temperaturschalter die Kühlflächentemperatur.

Zum Betrieb eines Schrittmotors benötigt die Steuerkarte die Signale Takt (Clk) und Richtung (Dir).

Weiterhin stehen Signaleingänge für Taktabschaltung (ClkEna), Endstufenabschaltung (Enable), Stromabsenkung (Stdbby), Voll-/Halbschritt-Betrieb (V/H) sowie Reset (Phasenlage ABCD = 01 01) zur Verfügung.

Durch Umstecken eines Jumpers ist sowohl eine Aktiv-high-Ansteuerung (H'-Signale) als auch eine Aktiv-low-Ansteuerung (L'-Signale) möglich.

Unter Berücksichtigung einer notwendigen Start- und Stop-Rampe kann mit der Schrittmotor-Steuerkarte - in Abhängigkeit von der mechanischen Last - eine Taktrate bis zu 10.000 Schritten (25 U/s) im Halbschritt-Betrieb, bzw. bis zu 8.000 Schritten Hz (40 U/s) im Vollschritt-Betrieb gefahren werden.

Als Open-Collector-Ausgänge sind am Kartensteckverbinder nach DIN 41 612 F24 H7 (Mischpolleiste) die Signale Home (Phasenlage 0101) sowie Störung (Resout) abgreifbar.

Für die Spannungsversorgung ist eine geregelte Festspannung (+ 5 V/200 mA) und eine unregelte Motorspannung (siehe "Technische Daten") notwendig.

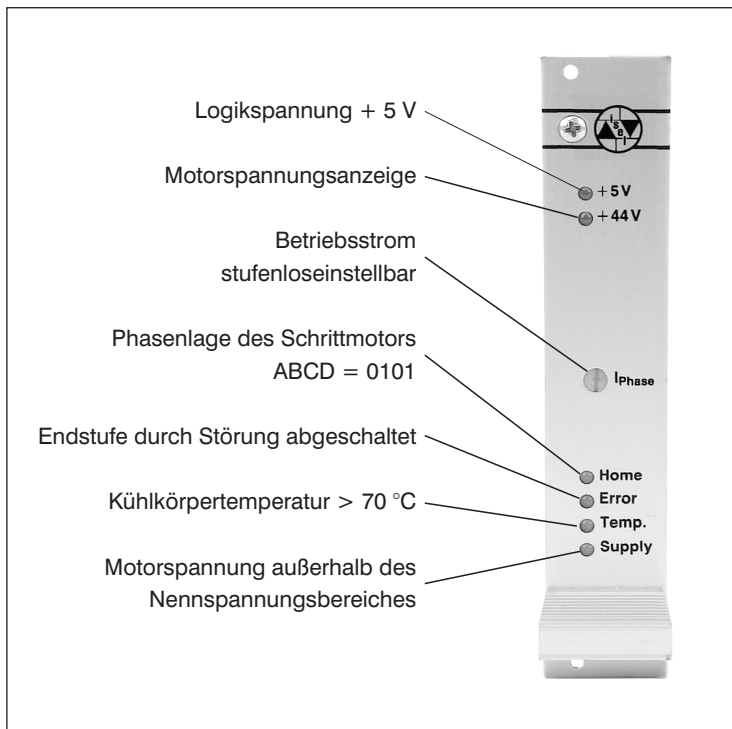


Bild 1: Frontplatte Schrittmotor-Leistungskarte

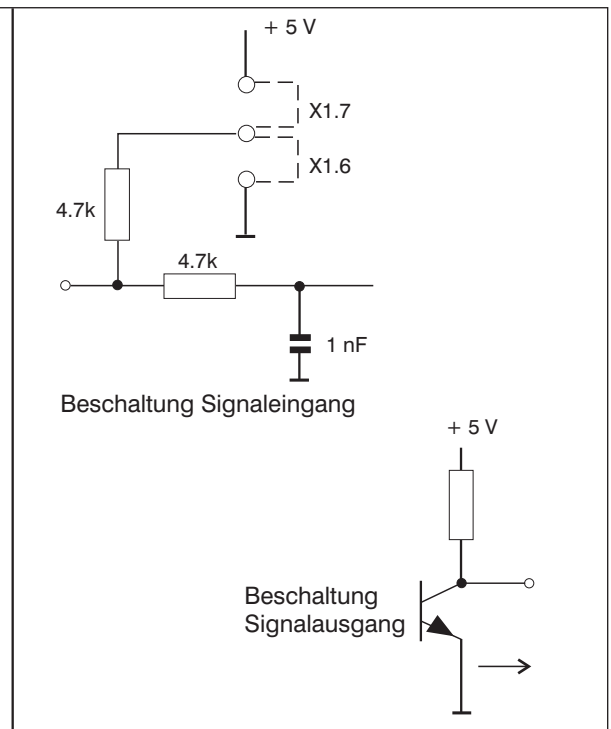


Bild 2: Signalein-/ausgänge

Technische Daten	
Abmessungen	Eurokarte 100 x 160 mm
Frontplatte	B = 8 TE (= 40 mm); H = 3 HE (= 128 mm)
Spannungsversorgung	
- Logikspannung	+ 5 V/200 mA
- Motorspannung	+ 24 V bis + 44 V
- Betriebsstrom	3,5 A
Steckverbinder	Mischpolleiste F 24 + H 7 (DIN 41 612)
Signaleingänge	<div>Takt (Clk)</div> <div>Richtung (Dir)</div> <div>Taktabschaltung (Clk Ena)</div> <div>Stromabsenkung (Stdby)</div> <div>Voll-/Halbschritt-Betrieb (V/H)</div> <div>Endstufenabschaltung (Enable)</div> <div>Reset (Reset)</div>
Bedienelemente	<div>Phasenstrom-Potentiometer (I_{Phase})</div> <div>Haltestrom-Potentiometer</div>
Signalausgänge	<div>Home (Home)</div> <div>Störung (Resout)</div>
Schrittmotorausgang	Mischpolleiste F 24 + H 7
Leuchtdioden	siehe Bild 1

Signalbeschreibung - Eingänge

Takt (Clk) d8

Abhängig von der Jumperstellung (s. Bild 2 und 4) führt jeder Taktimpuls mit einer Mindestbreite von 20 μ s am Eingang Clk (d8) zu einer definierten Schrittwinkelbewegung von 1,8° im Vollschritt- und 0,9° im Halbschritt-Betrieb. Die Schrittausführung können Sie durch den Signaleingang ClkEna (z8) sperren (siehe Taktabschaltung).

Richtung (Dir) d6

Signaleingang zur Festlegung der gewünschten Motordrehrichtung: Signal "H" - positive Drehrichtung des Schrittmotors (CCW)
Signal "L" - negative Drehrichtung des Schrittmotors (CW)

Taktabschaltung (ClkEna) z8

Eingang zum Sperren der Schrittmotorbewegung, unabhängig vom Takteingang d8.

Stromabsenkung (Stdby) z10

Signal "H"-Absenkung des Motorstromes auf einen niedrigeren, mit Haltestrom-Potentiometer einstellbaren Wert.

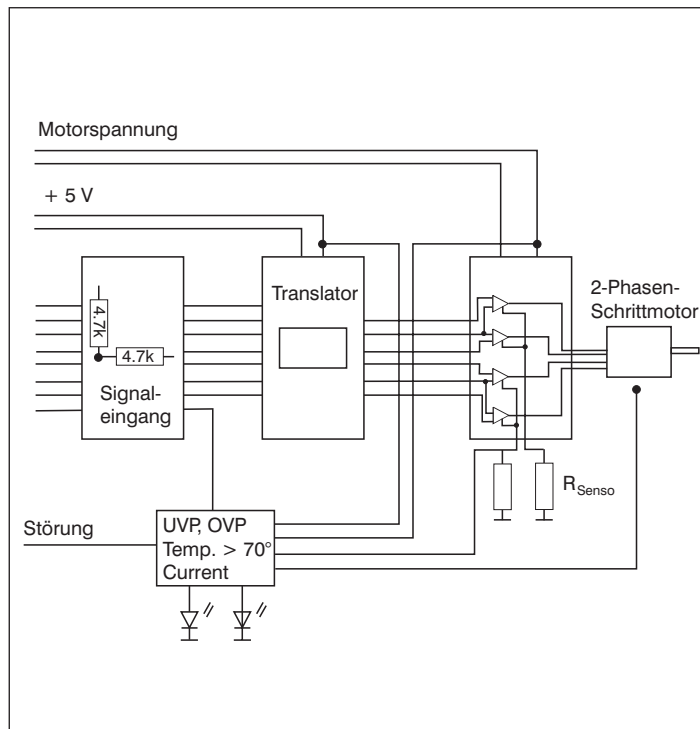


Bild 3: Blockschaltbild

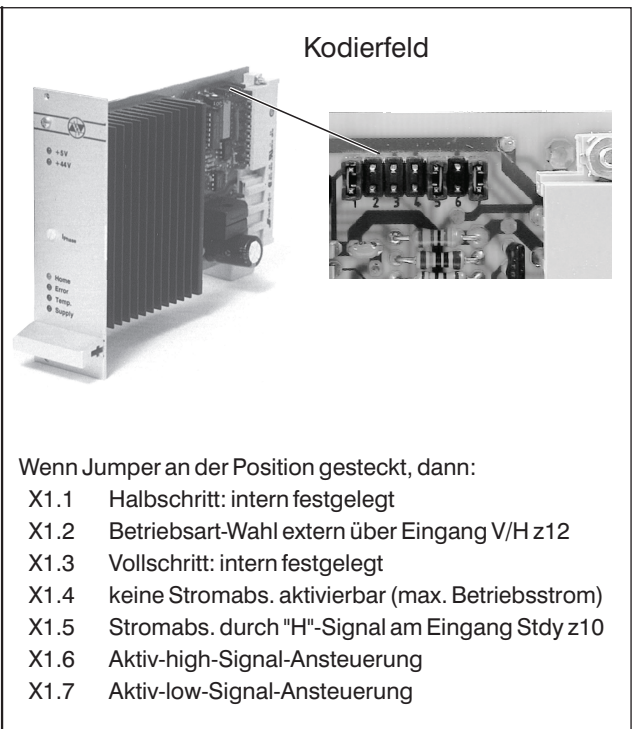


Bild 4: Kodierfeld X1

Signal "L" gibt den vollen eingestellten Motorstrom frei. Den Eingang können Sie durch Umstecken des Jumpers auf dem Kodierfeld X1 deaktivieren (siehe Bild 4).



Bei Haltestromeinstellung > 50 % des max. Betriebsstromes, Aktiv-low-Signal sowie deaktivierten Signaleinganges ist eine Fremdbelüftung der Steuerkarte notwendig.

Voll-/Halbschritt (V/H) z12
Eingang zur Bestimmung der gewünschten Betriebsart:

Signal "L" - Vollschritt

Signal "H" - Halbschritt

Das Signal bleibt unbeachtet, wenn das Kodierfeld X1 intern eine Betriebsart vorgibt (siehe Bild 4).

Endstufenabschaltung (Enable) z6
Signaleingang zum Abschalten der Leistungsendstufe.

Signal "H" - Endstufe aktiv

Signal "L" - Endstufe stromlos

Reset(PhasenlageABCD=0101) d10
Ein "L" -Signal am Signaleingang sperrt den Schrittmotorsequenzer. Nach anschließendem "H"-Signal wird die Phasenlage des Schrittmotors auf die definierte Startsequenz ABCD 0101 gelegt. Diese Situation wird durch die LED "Home" angezeigt und kann am Ausgang "Home" abgefragt werden.

- Ausgänge -

Home (Phasenlage 0101) d14
Der Open-Collector-Ausgang "Home" zeigt die stabile Schrittmotorphasenlage ABCD = 0101 an. Sie kann durch Aktivierung des Einganges Reset gesetzt werden. Der Ausgangstransistor öffnet daraufhin im Vollschritt-Betrieb nach jedem 4. und im Halbschritt-Betrieb nach jedem 8. Schritt. Es ergibt sich jeweils ein positiver Impuls, der unter anderem zur Referenzpunktbestimmung herangezogen werden kann.

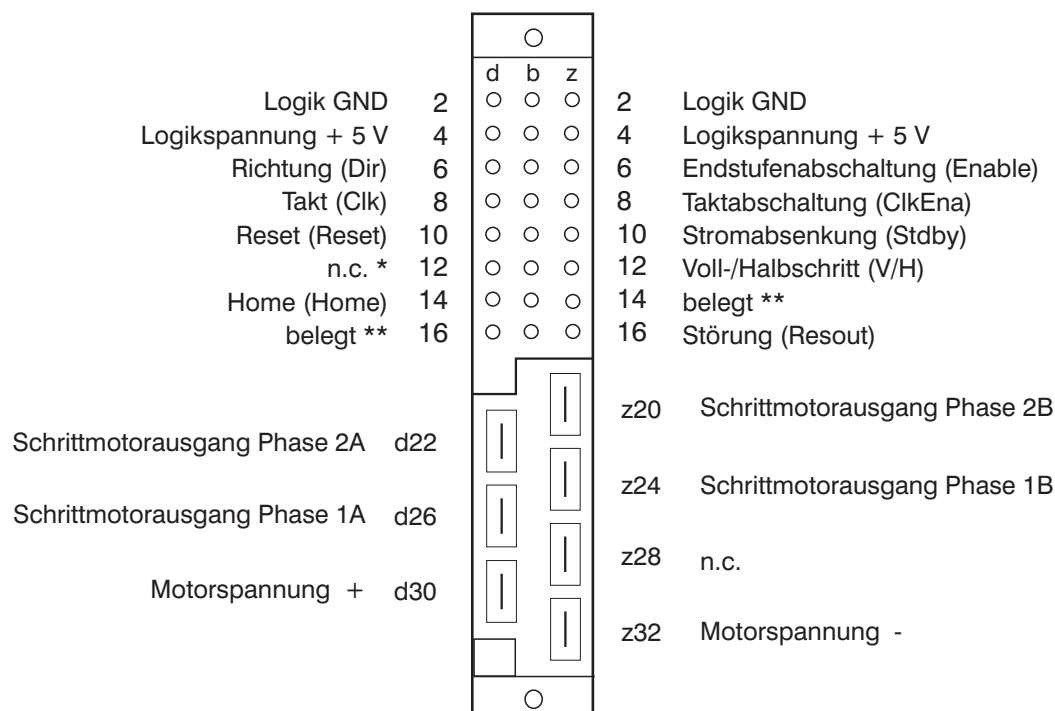
Störung (Resout) z16
Bei eingetretener Betriebsstörung der Steuerkarte mit Abschaltung der Leistungsendstufe, legt der Open-Collector-Ausgang der Überwachungsschaltung den Signalausgang "Resout" auf Masse-Potential.

Eine Betriebsstörung kann durch verschiedene Umstände erfolgen:

- Spannung am Spannungseingang + 5 V ist kleiner 4,6 V.
- Spannung am Motorspannungseingang liegt unterhalb des Spannungsbereiches (siehe "Technische Daten"). Bei Überschreiten der max. zulässigen Motorversorgungsspannung wird ein SCR aktiviert. Dieser schließt die Eingangsspannung kurz und lässt somit ein vorgeschaltetes Sicherungselement ansprechen.
- Überstrom-Schutzschaltung der Endstufe hat durch Überlast oder Kurzschluss an den Schrittmotorausgängen angesprochen.
- Kühlkörpertemperatur ist > 70°
- externer Eingang Endstufenabschaltung (Enable) liegt auf "L" Pegel.
- Optisch wird die Betriebsstörung durch die LED "Error" angezeigt.

Steckerbelegung des Kartensteckverbinder

- DIN 41 612 Mischpol -



Steckverbinder F24 H7
- Rückansicht -

* n.c. - nicht belegt - not connected

** belegt - Signalplan ist belegt, wird jedoch zur Zeit nicht ausgewertet.

Betriebsstromeinstellung (I_{phase})

Das frontseitige Potentiometer I_{phase} ermöglicht die lineare Einstellung des Phasenstromes.

Der Regelbereich liegt zwischen 0,5 A und 3,5 A im Normal-Betrieb.

Messtechnisch können Sie den Phasenstrom mit einem AC-Messinstrument bestimmen. Hierzu schalten Sie das Gerät in Reihe in eine der Schrittmotorleitungen. Bei einer programmierten Schrittfrequenz von ca. 400 Hz im Halbschritt-Betrieb zeigt das Messinstrument:

$$I_{\text{Mess}} = I_{\text{Phase}} \times 0,7 \Rightarrow I_{\text{P}} = I_{\text{M}} / 0,7$$

Zur Bestimmung des Phasenstromes mit einem Multimeter schalten

Sie das Gerät in eine Motorphase und messen den Phasenstrom im Stillstand (direkt nach dem Einschalten der Einheit; Home-LED leuchtet).

Haltestromeinstellung (I_{stdby})

Zur Verlustleistungsminimierung ist es sinnvoll und notwendig, während Motorstillstandszeiten den Betriebsstrom des Schrittmotors abzusenken. Dies ist mit Hilfe des Potentiometers "Haltestrom" stufenlos möglich. Zur Einstellung des Haltestroms ist, ähnlich wie bei der Einstellung des Betriebsstroms, ein Wechselstrommessinstrument in Reihe zu einer Motorphase zu schalten. Nachdem

der Signaleingang „Stdby“ z10 auf „H“-Potential gelegt ist, können Sie durch Drehen am Potentiometers den Haltestrom stufenlos einstellen.

- Drehrichtung links: Haltestrom <

- Drehrichtung rechts: Haltestrom >

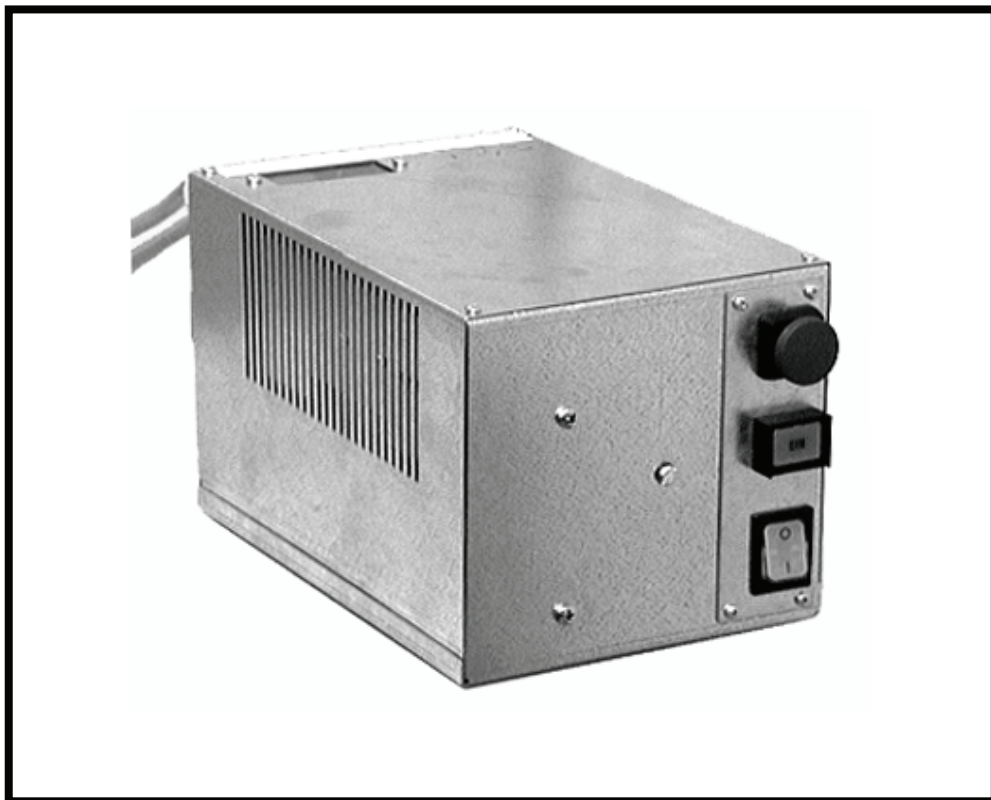


Bei einem eingestellten Betriebsstrom > 50 % des Nennstromes, bei Umgebungstemperaturen > 30 °C sowie bei nicht aktivierter Stromabsenkung muss die Leistungseinheit fremdbelüftet werden.

isel-Powerblock 300-C

isel-Powerblock 450-C

isel-Powerblock 600-C



Hardware-Beschreibung

B.308059.03/2000.11

Zu dieser Anleitung

In dieser Anleitung finden Sie verschiedene Symbole, die Ihnen schnell wichtige Informationen anzeigen.

Gefahr



Achtung



Hinweis



Beispiel



Zusatz-Infos



© Fa. **iselautomation** 1998

Alle Rechte vorbehalten

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



isel-Maschinen und Controller sind CE-konform und entsprechend gekennzeichnet.
Für alle sonstigen Maschinenteile und -komponenten, auf die CE-Sicherheitsrichtlinien anzuwenden sind, ist die Inbetriebnahme solange untersagt, bis alle entsprechenden Anforderungen erfüllt sind.



Die Firma **iselautomation** übernimmt keine Gewähr, sobald Sie irgendwelche Veränderungen an dem Gerät vornehmen.



Die in der Konformitätserklärung aufgeführten Grenzwerte gelten nur für die ab Werk gelieferte Originalkonfiguration.

Hersteller: Fa. **iselautomation GmbH & Co, KG**
Im Leibolzgraben 16
D-36132 Eiterfeld

Fax: (06672) 898-888
e-mail: automation@isel.com
<http://www.isel.com>

Inhaltsverzeichnis

1	Einleitung4
2	Lieferumfang4
3	Sicherheitshinweise5
4	Technische Daten7
4.1	Motorspannung7
4.2	Hilfsspannung I7
4.3	Hilfsspannung II7
4.4	Sicherheitseinrichtungen7
5	Systembeschreibung8
5.1	Funktionsgruppen8
5.2	Anschluss und Verkabelung8
5.2.1	Steckverbinder X18
5.2.1.1	Signalausgänge (A)9
5.2.1.2	Signaleingänge (E)10
5.2.2	Steckverbinder X210
5.2.3	Steckverbinder X312
5.3	Statusanzeigen des Powerblocks13
5.4	Kodierfeld14
5.5	Sicherungen15
5.6	Spannungsausgang AC 230 V/50 Hz16
6	Blockschaltbild Powerblock PB xxx-C17
7	Schaltungsunterlagen18
7.1	Sicherheitskreis isel-Powerblock18

1 Einleitung

isel-Powerblöcke PB xxx-C sind Einbaugeräte, die speziell zur Spannungsversorgung von isel-Leistungseinheiten (CV 4, C 142-4) entwickelt wurden. Sie integrieren innerhalb eines Stahlblechgehäuses (Abmessungen B = 150 x H = 140 x T = 220 mm) einen 650 VA-Ringkerntransformator mit Einschaltstrombegrenzung und Netzfilter sowie eine Flachbaugruppe zur Bereitstellung von Hilfsspannungen und sicherheitsrelevanten Funktionselementen.

Die Powerblöcke stehen in drei verschiedenen Ausführungen zur Verfügung, die sich nur durch die Höhe der Lastspannung (Versorgungsspannung der Leistungsstufen) unterscheiden.

PB 600-C	Spannungsausgang 68 V/7 A
PB 450-C	Spannungsausgang 43 V/8 A
PB 300-C	Spannungsausgang 30 V/8 A



Bild 1: isel-Powerblock PB xxx-C

2 Lieferumfang

Der Lieferumfang des Powerblock PB xxx-C umfasst:

- Powerblock mit Netzzuleitung (l = 0,5 m)

3 Sicherheitshinweise



- Bei Einbau bzw. Verwendung des Powerblocks beachten Sie bitte die Normen gemäß der Konformitätserklärung.
- Die vom Hersteller eingehaltenen Vorschriften und Grenzwerte schützen nicht bei unsachgemäßem Gebrauch des Gerätes.

In diesem Zusammenhang sollten ...

- ... Sie das Gerät nur dann anschließen und montieren, wenn es abgeschaltet und die Netzleitung gezogen ist.
- ... alle Arbeiten ausschließlich von Fachpersonal ausgeführt werden. Hierbei berücksichtigen Sie insbesondere die Bestimmungen und Vorschriften der Elektroindustrie sowie der Unfallverhütung.

Zugrundegelegte Vorschriften des Schrittmotor-Controllers:

EN 60204 (VDE 0113) Teil 1 (Ausgabe 1992)

- Elektrische Ausrüstung von Industriemaschinen

EN 50178 (VDE 0160)

- Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln

VDE 0551

- Bestimmungen für Sicherheitstransformatoren

EN 292 Teil 1 und 2

- Sicherheit von Maschinen

EN 55011 (VDE 0875)

- Funkentstörung, Grenzwert B

IEC 1000-4 (Teil 2-5)

- Prüf- und Messverfahren der Störfestigkeit

Zur Spannungsversorgung benötigen die Powerblöcke:

PB 600-C	AC 230 V/50 Hz, max. 8 A
PB 450-C	AC 230 V/50 Hz, max. 7,5 A
PB 300-C	AC 230 V/50 Hz, max. 7,0 A

Der Netztransformator verfügt primärseitig über einen Temperaturschalter mit einer Ansprechtemperatur von 120 °C. Setzen Sie bei Anschluss des Powerblockes zusätzliche Primärsicherungen ein. Bei direktem Anschluss an die Hausinstallation ist eine Primärabsicherung des Powerblockes durch das dortige Sicherungselement (16 A) gewährleistet.

Bei Einbau des Powerblockes in ein Steuerungssystem (z. B. Schaltschrank) muss eine zusätzliche Primärsicherung vorgesehen werden. Hierzu sind ausschließlich Sicherungen nach IEC-127 zu verwenden. Die Netzleitung wird durch eine PG 9-Verschraubung (Klemmbereich 4-8 mm) in den Powerblock geführt. Die Verbindungsleitung muss eine doppelt isolierte Leitung sein.

Bei der Montage der Powerblöcke sind weiterhin folgende Faktoren zu beachten:

- Bei dem Powerblock handelt es sich um ein Einbaugerät der Schutzklasse 1
- Die Schutzart des Powerblockes ist mit IP 20 festgelegt.
- Die Montage des Powerblockes darf nur horizontal liegend erfolgen.
- Primär- und Sekundärleitungen sind als Kabel auszuführen (keine Einzelleitungen).
- Primär- und Sekundärleitungen müssen durch 3 Lagen Isolierstoff getrennt sein.
- Der Powerblock ist für einen Betrieb bei Umgebungstemperatur von max. 40 °C ausgelegt.

4 Technische Daten

4.1 Motorspannung

Zur Versorgung von Leistungsendstufen stellen die Powerblöcke PB xxx-C eine unregelte Gleichspannung (Zwischenkreisspannung) zur Verfügung. Der Spannungsausgang wird durch eine Sicherheitseinrichtung mit nachgeschalteten Schaltrelais freigegeben.

Bei der Sicherheitseinrichtung handelt es sich um eine Reihenschaltung von Befehlseinrichtungen die über ein Sicherheitsrelais mit zwangsgeführten Kontakten und nachgeschalteten Schaltrelais die Sekundärspannung des Ringkerntransformators einschaltet. Die Ausgangsspannung des Zwischenkreises wird über vier getrennte Sicherungselemente auf WAGO-Klemmen geführt. Die dort angeschlossenen Verbindungsleitungen werden über PG-Verschraubungen aus dem Netzteil herausgeführt.

Zum Schutz vor Überspannungen der Zwischenkreisspannung (z. B. durch Energierückspeisung im Bremsbetrieb der Motoren) verfügt der Powerblock über eine entsprechende Schutzschaltung (Bremschopper). Diese schaltet bei Spannungen $> 80 \text{ V}$ automatisch einen Leistungswiderstand ein, der die Energie in Wärme umsetzt. Bei abgeschaltetem Sicherheitskreis wird die gespeicherte Energie des Zwischenkreiskondensators durch einen Lastwiderstand entladen.

4.2 Hilfsspannung I

Bei dieser $+ 24 \text{ V}$ -Hilfsspannung handelt es sich um den Ausgang eines Festspannungsreglers. Die Eingangsspannung wird von einer doppelt isolierten Sekundärwicklung des Transformators bezogen.

Die $+ 24 \text{ V}$ dienen der Spannungsversorgung der Signalein- und -ausgänge, der externen Endlagen- und Referenzschalter sowie der Steuerrelais des Sicherheitskreises. Der maximal extern nutzbare Strom beträgt $0,7 \text{ A}$.

4.3 Hilfsspannung II

Diese $+ 24 \text{ V}$ -Hilfsspannung ist zur Spannungsversorgung des Sicherheitskreises vorgesehen. Als Eingangsspannung wird ein doppelt isolierter Sekundärabgriff des Ringkerntransformators genutzt. Der eingesetzte $+ 24 \text{ V}$ -Festspannungsregler begrenzt den Ausgangsstrom auf ca. $1,0 \text{ A}$.

4.4 Sicherheitseinrichtungen

Die Realisierung des Sicherheitskreises beruht auf einer Reihenschaltung von Befehlseinrichtungen, z. B. NOT-AUS-Schalter, Sicherheitsschleifen und EIN-Taste.

Die sicherheitsrelevanten Teile wirken auf ein Relais mit zwangsgeführten Kontakten, das wiederum Lastrelais' einschaltet. Die Lastrelais werden gemäß EN 60204 durch einen Opto-Koppler überwacht und sind redundant ausgeführt.

5 Systembeschreibung

5.1 Funktionsgruppen

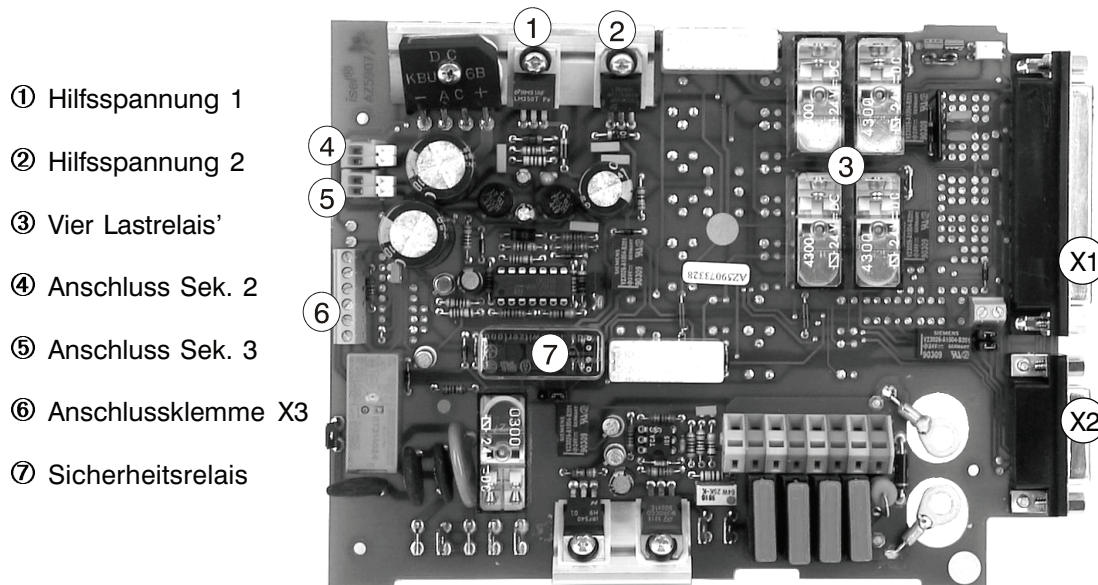


Bild 2: Funktionsgruppen des Powerblocks

5.2 Anschluss und Verkabelung

5.2.1 Steckverbinder X1

Zur Ankopplung der Leistungseinheiten verfügt der Powerblock über einen 37-poligen Sub D-Buchsenstecker.

Signal	Pin	Pin	Signal
GND	A 1	20	nicht belegt
+ 24 V	A 2	21	nicht belegt
nicht belegt	3	22	E Endlagenschalter
Endstufenabschaltung 1	A 4	23	A Freischaltung Endlagenschalter
GND	A 5	24	E Antriebsfreigabe
+ 24 V	A 6	25	nicht belegt
nicht belegt	7	26	nicht belegt
Endstufenabschaltung 2	A 8	27	A Sicherheitskreis ok
GND	A 9	28	nicht belegt
+ 24 V	A 10	29	nicht belegt
nicht belegt	11	30	nicht belegt
Endstufenabschaltung 3	A 12	31	A Überwachung Lastrelais
GND	A 13	32	nicht belegt
+ 24 V	A 14	33	nicht belegt
nicht belegt	15	34	nicht belegt
Endstufenabschaltung 4	A 16	35	nicht belegt
nicht belegt	17	36	A Opto-Koppler X1
GND	A 18	37	A GND
GND	A 19		

5.2.1.1 Signalausgänge (A)

+ 24 V, GND

Hierbei handelt es sich um den Spannungsausgang der Hilfsspannung I.

Die Spannung dient zur Versorgung der Referenzschalter der numerischen Achsen sowie von Opto-Kopplern in der Leistungselektronik.

Endstufenabschaltung

Dieser Ausgang dient zum Abschalten (stromlos schalten) der Schrittmotor-Leistungsendstufen. Das + 24 V-Ausgangssignal des Powerblockes wird bei abgeschaltetem Lastrelais aktiviert (Versorgungsspannung der Leistungsendstufen ist ausgeschaltet) und parallel auf alle angeschlossenen Endstufen-Karten geführt.

Freischaltung der Endlagenschalter

Der Signalausgang liefert bei überbrückter Endlagenschalter-Überwachungsschaltung ein + 24 V-Signal.

Eine Überbrückung dieser sicherheitsrelevanten Funktionsgruppe ist dann notwendig, wenn ein oder mehrere Endlagenschalter (Begrenzungsschalter) aktiv sind. Dies kann z. B. durch ein Fehlverhalten der Antriebseinheit (Controller) bzw. der Mechanik oder infolge eines Bedienfehlers geschehen (siehe hierzu auch Signalausgang Freischaltung).



Das Signal wird nur in Verbindung mit Servomotor-Leistungseinheiten verwendet.

Sicherheitskreis ok

Eine + 24 V-Spannung am Steckverbinder X1.27 signalisiert, dass die Spannungsversorgung der Leistungsendstufen (Zwischenkreisspannung) eingeschaltet ist.

Dieses Ausgangssignal ist die Steuerspannung des Lastrelais' und ist auf den Steckkontakt geführt; Bezugsmasse ist Kontakt X1.9 (GND).

Überwachung Lastrelais

Dies ist ein + 24 V-Ausgang (offener Emitter) eines Opto-Kopplers, der signalseitig die Schaltkontakte der redundant aufgebauten Lastrelais' überwacht. Bei ausgeschalteter Zwischenkreisspannung (Sicherheitsrelais ist abgeschaltet) und fehlerhaften Lastrelais' (z. B. Kontakt ist verschweißt) führt der Ausgang + 24 V-Potential.

Opto-Koppler X1

Der Ausgang ist die Statusanzeige des Sicherheitskreises. Der Opto-Koppler-Ausgang ist aktiv (+ 24 V geschaltet), wenn die Reihenschaltung der sicherheitsrelevanten Bedienelemente funktionsfähig ist und somit das Betätigen des EIN-Tasters zum Schalten des Sicherheitsrelais' führt.

5.2.1.2 Signaleingänge (E)

Endlagenschalter

Endlagenschalter in den numerischen Achsen dienen zur Begrenzung der maximalen Verfahrswege. Sie wirken über Relais direkt in die Sicherheitskette des Powerblockes und unterbrechen bei Betätigung die Spannungsversorgung der angeschlossenen Leistungskarten.



Zur Aktivierung des Relais' ist der Eingang des Powerblockes mit einem + 24 V-Signal zu beschalten. Das Fehlen der Steuerspannung führt zum Abfallen des Relais' und somit zur Unterbrechung der Sicherheitseinrichtung.

Der Signaleingang Endlagenschalter wird nur in Servomotor-Leistungseinheiten ausgewertet. Hierbei werden alle Endlagenschalter auf der Ankopplungsplatine des Controllers überwacht und als Summensignal zum Powerblock geführt.

Antriebsfreigabe

Zur Überwachung der Funktionsbereitschaft der angeschlossenen Leistungseinheiten bzw. eines Steuerrechners erwartet der Powerblock ein Freigabesignal.

Das + 24 V-Signal wirkt über ein Relais auf den Sicherheitskreis des Powerblocks.

5.2.2 Steckverbinder X2

Der 15-polige Sub D-Buchsenstecker X2 ist zur Kontaktierung von externen, sicherheitsrelevanten Bedienelementen vorbereitet. Entsprechend nachstehender Belegung sind hier NOT-AUS-Schalter, EIN-Taste, Sicherheitskontakte usw. anschließbar.

Signal	Pin	Pin	Signal
Schlüsselschalter (Schließer)**	1	9	Schlüsselschalter (Schließer)**
EIN-Taste (Schließer)	2	10	EIN-Taste (Schließer)
Sicherheitsschalter (Öffner)	3	11	Sicherheitsschalter (Öffner)
Sicherheitskontakt (Öffner)	4	12	Sicherheitskontakt (Öffner)
NOT-AUS-Schalter (Öffner)	5	13	NOT-AUS-Schalter (Öffner-Kontakt)
Sicherheitsrelais (GND)	6	14	Sicherheitsrelais (+ 24 V = aktiv)
potentialfreier Schaltkontakt	7	15	potentialfreier Schaltkontakt
Überwachung Lastrelais	8		

** wird nur in Verbindung mit Servomotor-Leistungseinheiten ausgewertet

Die Kontaktbelegung im einzelnen:

Schlüsselschalter

Ein geschlossener Kontakt zwischen X2.1 und X2.9 überbrückt die Endlagenschalterüberwachung. Dies hat zur Folge, dass ein betätigter Begrenzungsschalter in den numerischen Achsen nicht zum Abschalten der Betriebsspannung führt (siehe hierzu auch Kapitel 4.2.1.2: Freischaltung Endlagenschalter).



Bei Verwendung eines Schlüsselschalters achten Sie darauf, dass der Kontakt nicht länger als unbedingt notwendig eingeschaltet wird.



Die Schutzvorrichtungen der Antriebsachsen sind abgeschaltet! Achten Sie unbedingt auf die maximalen Verfahrswege ihrer Antriebsachse. Bei einer Kollision innerhalb der Mechanik sind Beeinträchtigungen der Funktionsfähigkeit nicht auszuschließen.

EIN-Taste

Ein Schließerkontakt zwischen X2.2 und X2.10 führt zum Einschalten der Zwischenkreisspannung (Spannungsversorgung der Leistungsendstufen), sofern alle Funktionselemente der Sicherheitskette aktiv sind.

Sicherheitsschalter

Das Betätigen eines zwischen den Kontakten X2.3 und X2.11 angeschlossenen Öffner-Schalters bewirkt das Abschalten der Betriebsspannung der Endstufen. Auswahl und Einsatz des Schalters sollten gemäß EN 418 erfolgen. Wird der Schaltkontakt nicht benötigt, sind die Kontakte durch eine Brücke zu schließen.

Sicherheitskontakt

Das Betätigen eines zwischen den Kontakten X2.4 und X2.12 angeschlossenen Öffner-Schaltkontaktes führt zum Abschalten der Betriebsspannung. Auswahl und Einsatz des Schalters sollte gemäß EN 418 erfolgen. Bei nicht benötigtem externen Schaltkontakt sind die Kontakte zu brücken.

NOT-AUS Schalter

Das Betätigen des zwischen den Kontakten X2.5 und X2.13 angeschlossenen Öffner-Schaltkontaktes eines NOT-AUS-Schalters führt zum Abschalten der Betriebsspannung. Auswahl und Einsatz des Schalters sollte gemäß der EN 418 erfolgen. Bei nicht angeschlossenem NOT-AUS-Schalter sind die Kontakte durch eine Brücke zu verbinden.

Sicherheitsrelais aktiv

Der Ausgang ist die Steuerspannung des im Powerblock eingebauten Lastrelais'. Somit liegt am Ausgang X2.14 bei eingeschaltetem Lastrelais eine + 24 V-Spannung an; Bezugsmasse ist der Kontakt X2.6.

Potentialfreier Schaltkontakt

Die Ausgänge X2.7 und X2.15 sind innerhalb des Powerblockes auf einen potentialfreien Relaiskontakt geführt. Der Kontakt ist bei eingeschalteter Zwischenkreisspannung (Betriebs-spannung der Leistungsendstufen) geschlossen. Er kann zur Einbindung des Powerblockes in übergeordnete Sicherheitssysteme genutzt werden.

Überwachung Lastrelais

Dieser Ausgang führt bei abgeschaltetem Sicherheitskreis aber defekten Schaltkontakten der Schaltrelais' zu einem + 24 V-Signal (pulsierende Gleichspannung). Als Bezugsmasse wird Kontakt X2.6 verwendet.

5.2.3 Steckverbinder X3

Die 8-polige Platinklemme ermöglicht den Anschluss von folgenden Bedienelementen:

- 1 - 2 NOT-AUS-Schalter
- 3 - 4 EIN-Taste
- 5 - 6 Leuchte EIN-Taste (leuchtet bei eingeschaltetem Sicherheitskreis)
- 7 - 8 Schlüsselschalter



Die Funktion der Schaltelemente ist identisch mit der des 15-poligen Sub D-Steckers X2. Bei Anschluss des EIN-Tasters achten Sie darauf, dass gemäß den Sicherheitsvorschriften nur eine EIN-Schaltfunktion vorhanden sein darf und somit ein gleichzeitiger externer Anschluss des EIN-Tasters an dem Steckverbinder X2 nicht erlaubt ist.

5.3 Statusanzeigen des Powerblocks

Zur Anzeige der Betriebszustände verfügt der Powerblock über vier LEDs. (V1 bis V4)

- ① LED V1
- ② LED V2
- ③ LED V3
- ④ LED V4

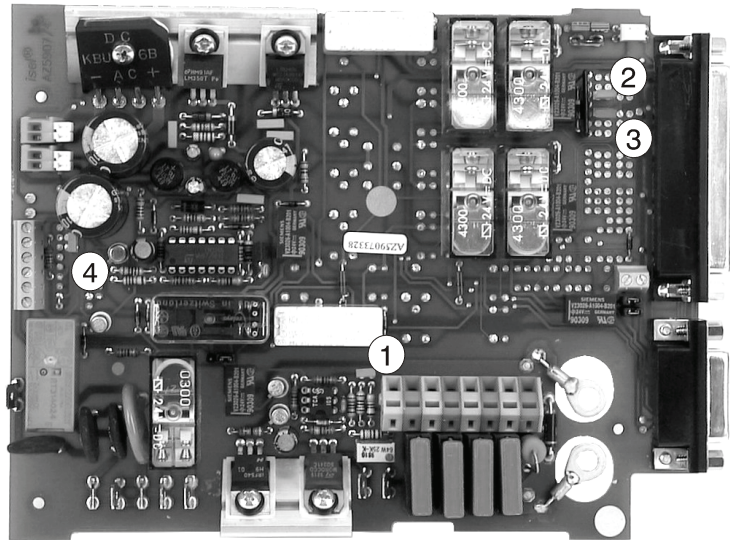


Bild 3: LED-Elemente des Powerblocks PB xxx-C

Die LED's im einzelnen:

- V1** Die LED V1 zeigt an, dass der Bremschopper aktiv ist und damit einen Lastwiderstand parallel zum Zwischenkreiskondensator schaltet. Dieser Betriebszustand kann durch zwei Ereignisse herbeigeführt werden:
- Überspannung am Kondensator z. B. durch Energierückspeisung des angeschlossenen DC-Servomotors.
 - Einschalten des Lastwiderstandes zum schnellen Abbau der Speicherenergie des Zwischenkreiskondensators nach Ausschalten des Sicherheitskreises.
- V2** Die LED V2 signalisiert, dass der Eingang Endlagenschalter aktiv (+ 24 V) ist und die Überwachung durch den Sicherheitskreis erfüllt ist.
- V3** Die LED V3 leuchtet, sobald am Signaleingang Antriebsfreigabe ein + 24 V-Signal anliegt und damit die Leistungsendstufen ihre Betriebsbereitschaft melden. Der Eingang wirkt direkt in den Sicherheitskreis des Powerblockes.
- V4** Die LED V4 leuchtet, wenn der Sicherheitskreis Betriebsbereit ist, d. h. wenn folgende sicherheitsrelevanten Bedienteile aktiv sind.
- Not-Aus-Schalter (extern) ÖFFNER
 - Sicherheits-Schalter (z. B. Haubenkontakt) ÖFFNER
 - Not-Aus-Schalter (intern)
 - Sicherheitskontakt (z. B. Trittschutzmatte, Lichtband) ÖFFNER

5.4 Kodierfeld

Die Steuerplatine des Powerblockes verfügt über drei Kodierbrücken, mit denen Sie den Powerblock auf unterschiedliche Betriebsbedingungen anpassen können.

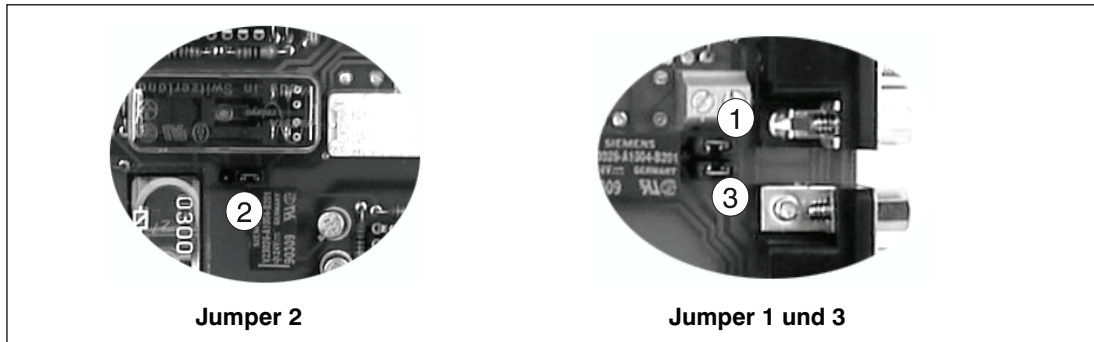


Bild 4: Kodierfelder und Sicherungen des Powerblockes PB 600-C

Kodierfelder J1

Der Kodierstecker J1 hat die Aufgabe, den Anschluss eines externen EIN-Tasters an Steckverbinder X2 vorzubereiten. Da entsprechend der Maschinenschutzverordnung nur durch eine EIN-Taste die Zwischenkreisspannung des Powerblockes eingeschaltet werden darf, achten Sie nach dem Brücken von J1 unbedingt darauf, dass der EIN-Taster nicht mehr bedient werden kann (Entfernen der Tasterleitungen aus Stecker X3, Abdecken des Betätigungsknopfes, etc.).

Kodierbrücke J2

Diese Kodierbrücke ermöglicht, den Schaltzeitpunkt des Ausgangsrelais' zu bestimmen (siehe Kapitel 4.6). Sie können zwischen zwei Betriebszuständen wählen:

- **J2.1** Das Ausgangsrelais schaltet zeitgleich mit dem Lastrelais des Powerblockes. Hierbei wird die Ausgangsspannung (AC 230 V/50 Hz) durch das Sicherheitsrelais aktiviert.
- **J2.2** Bei gesteckter Kodierung '2' schaltet das Ausgangsrelais sofort nach dem Einschalten des Transformators.

Kodierbrücke J3

Diese Kodierbrücke ist für eventuelle Erweiterungen vorgesehen und bei dem Powerblock PB xxx-C geschlossen.

5.5 Sicherungen

Sicherungen F1 - F4

Der Spannungsabgriff der Zwischenkreisspannung erfolgt über vier getrennte WAGO-Klemmblöcke. Es können maximal vier Leistungsendstufen adaptiert werden. Zum Schutz der Spannungsausganges ist jedem Klemmblock eine Sicherung (F1 - F4) vorgeschaltet. Es handelt sich um FKS-Sicherungen mit einem Nennwert von 5 A (träge).



Sicherungen F5 und F6

Die Sicherungen F5 und F6 sind in die Ausgangsleitung des Schaltrelais' eingefügt (siehe Kapitel 4.6) und schützen das Relais vor Überlastung. Es handelt sich um zwei Schmelzsicherungen nach IEC-127 mit einem Nennwert von 4 A (träge).

Temperaturschalter T1

Der Temperaturschalter T1 liegt innerhalb der Primärwicklung des Ringkerntransformators. Dieser löst bei einer Temperatur von über 120 °C aus. Nach Abkühlen des Transformators auf ca. 60 °C schaltet der Temperaturschalter wieder selbsttätig ein. Bedingt durch die Selbsthaltung des Sicherheitsrelais' wird gewährleistet, dass dabei die Motorspannung nicht freigegeben wird.



Da der Primärkreis des Netztransformators lediglich mit einem Temperaturschalter vor Überlastung geschützt ist, müssen Sie beim Einbau des Powerblockes eine zusätzliche Primärabsicherung einbauen. Bei direktem Anschluss an die Hausinstallation, ist eine Primärabsicherung des Powerblockes durch das dortige Sicherungselement (16 A) gewährleistet



Beim Einbau des Powerblockes in ein Steuerungssystem (z. B. Schaltschrank), muss eine zusätzliche Primärabsicherung vorgesehen werden.



Es sind ausschließlich Sicherungen nach IEC-127 zu verwenden. Die Sicherungen mit einem Nennwert von 8 A sollten eine träge Schaltcharakteristik aufweisen.

Temperaturschalter T2

Der Temperaturschalter T2 liegt innerhalb der Sekundärwicklung 2 (Hilfsspannung II) des Netztransformators. Die Ansprechtemperatur des Schalters liegt bei 120 °C.

Temperaturschalter T3

Der Temperaturschalter T3 liegt innerhalb der Sekundärwicklung 3 (Hilfsspannung I) des Netztransformators. Die Ansprechtemperatur des Schalters liegt bei 120 °C.

5.6 Spannungsausgang AC 230 V/50 Hz

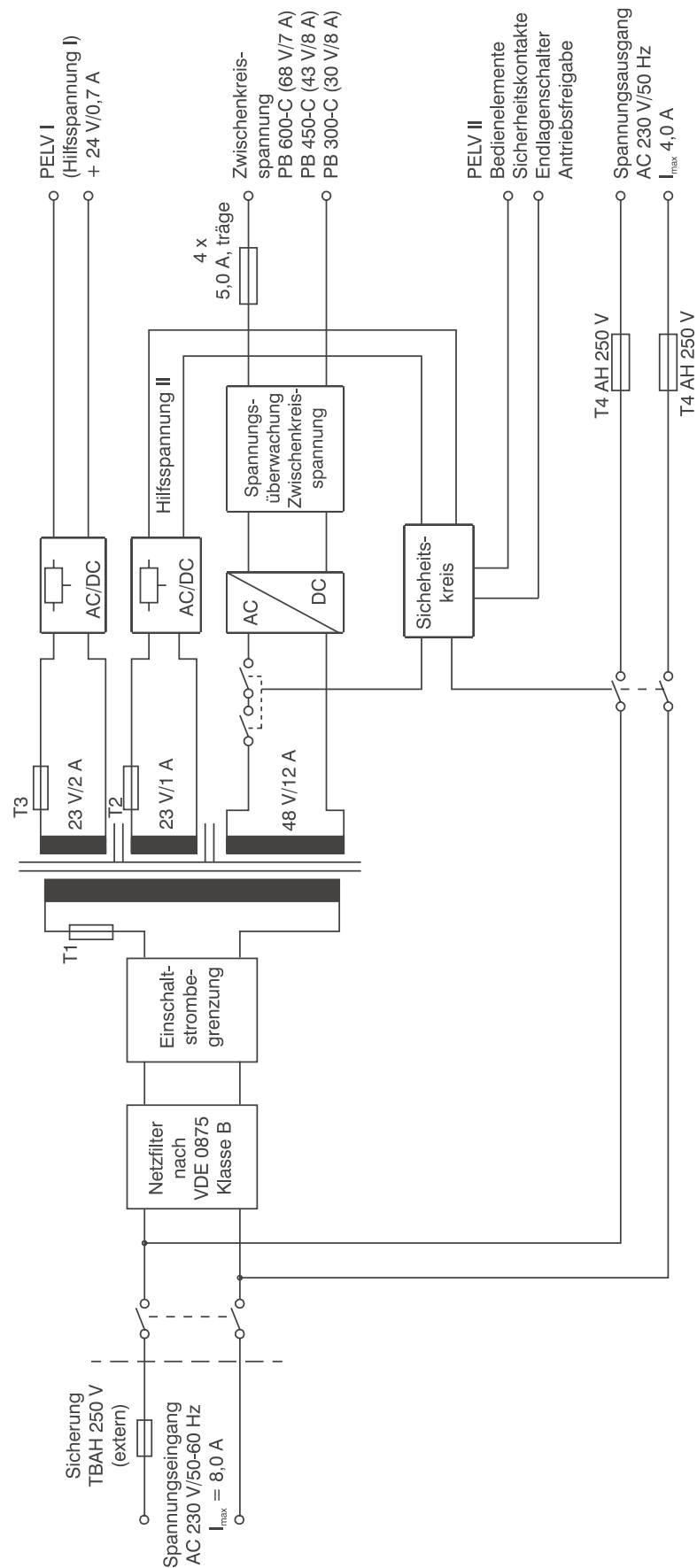
Zur Ansteuerung eines zusätzlichen externen Gerätes (Eingangsspannung AC 230 V/50 Hz max. 4 A) stellt der Powerblock einen entsprechenden Ausgang zur Verfügung.

Dieser Ausgang wird durch ein Lastrelais eingeschaltet, das elektronisch mit dem Sicherheitsrelais des Sicherheitskreises gekoppelt ist. Dadurch steht die Ausgangsspannung erst dann zur Verfügung, wenn alle sicherheitsrelevanten Teile aktiv sind.

Der Spannungsausgang ist ausgangsseitig durch zwei Sicherungen T 4.0 A H 250 V (5 x 20 mm, IEC-127) abgesichert. Als Anschluss ist eine 3-adrige PVC-Leitung vorgesehen, die über eine PG-11-Verschraubung aus dem Gehäuse herausgeführt ist.

Bei nachträglicher Montage der Ausgangsleitung ist eine doppelt isolierte Leitung (keine Einzelleitungen) mit einem Leiterquerschnitt von 1,0 mm zu verwenden. Der Kabeldurchmesser sollte bedingt durch die PG-9-Verschraubung im Bereich von 4-8 mm liegen.

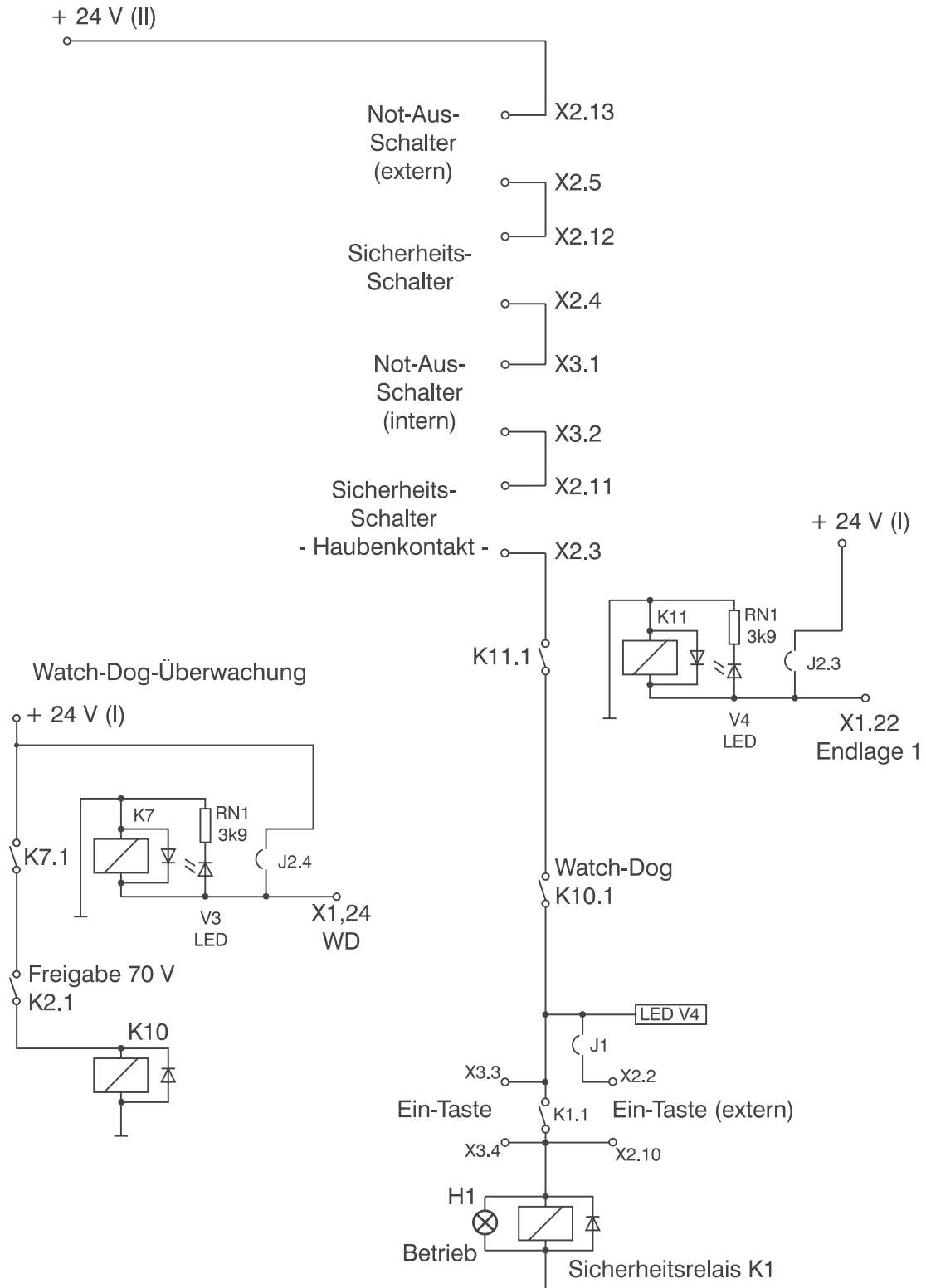
6 Blockschaltbild Powerblock PB xxx-C



Überspannungs-Festigkeitsklasse 2

7 Schaltungsunterlagen

7.1 Sicherheitskreis isel-Powerblock



isel-CNC-Betriebssystem 5.x



Programmieranleitung

970325 BD003
10/2000

Zu dieser Anleitung

In dieser Anleitung finden Sie verschiedene Symbole, die Ihnen schnell wichtige Informationen anzeigen.

Gefahr



Achtung



Hinweis



Beispiel



Zusatz-Infos



© Fa. **isel** automation KG 1998
Alle Rechte Vorbehalten

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

Kein Teil dieser Programmieranleitung darf in irgendeiner Form ohne schriftliche Genehmigung der Firma **isel** automation KG reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Hersteller: Fa. **isel** automation GmbH & Co. KG
Im Leibolzgraben 16
D-36132 Eiterfeld

Fax: (06672) 898-888
e-mail: automation@isel.com
<http://www.isel.com>

Inhaltsverzeichnis

1	Einleitung	5
2	Befehlsaufbau der DNC-Befehle	6
2.1	Grundbefehlssatz ab Prozessorkarte 4.0	7
2.1.1	Befehl: Achsenanzahlsetzen	7
2.1.2	Befehl: Referenzfahrt	8
2.1.3	Befehl: Referenzgeschwindigkeit setzen	10
2.1.4	Befehl: Bewegung relativ	11
2.1.5	Befehl: Bewegung zur Position (move to)	13
2.1.6	Befehl: Positionsabfrage	15
2.1.7	Befehl: Nullpunktverschiebung	16
2.1.8	Befehl: Ebenenwahl	17
2.1.9	Befehl: Peek (Lesen von Speicheradressen)	19
2.1.10	Befehl: Poke (Beschreiben von Speicheradressen)	20
2.1.11	Befehl: Batterie-RAM löschen	21
2.1.12	Befehl: CR/LF setzen	22
2.1.13	Befehl: Gerätenummer setzen	23
2.1.14	Befehl: TRACE (Einzelschrittausführung)	24
2.1.15	Befehl: Selbsttest	25
2.2	Ergänzender Befehlsumfang der Interface-Karte 5.0	26
2.2.1	Befehl: 3D-Linearinterpolation	26
2.2.2	Befehl: Zirkularinterpolation	28
2.3	Ergänzender Befehlsumfang bei Interface-Karten mit E/A Erweiterung	34
2.3.1	Befehl: Externe Speicherung	34
2.3.2	Befehl: Ausgangsport setzen	35
2.3.3	Befehl: Eingangsport lesen	35
2.4	Ergänzender Befehlsumfang EP1090	36
2.4.1	Befehl: Output-Modul	36
2.5	Ergänzender Befehlsumfang ab Interface-Karte Version AZ1350/5	36
2.5.1	Befehl: Magnetbremse	36
2.6	Kontroll- und Steuercodes	37
2.6.1	Befehl: Selbsttest	37
2.6.2	Befehl: STOP	38
2.6.3	Befehl: μ P-Reset	39
2.6.4	Befehl: Break	39

...

3	Befehlsaufbau CNC-Befehle	40
3.1	Grundbefehlssatz ab Prozessorkarte 4.0	41
3.1.1	Befehl: INPUT	41
3.1.2	Befehl: Referenzfahrt	42
3.1.3	Befehl: Bewegung relativ	43
3.1.4	Befehl: Bewegung zur Position (move to)	44
3.1.5	Befehl: Nullpunktverschiebung	45
3.1.6	Befehl: Ebenenwahl	46
3.1.7	Befehl: Synchronisationszeichen senden	47
3.1.8	Befehl: Warte auf Synchronisationszeichen	49
3.1.9	Befehl: Schleife / Verzweigung	50
3.1.10	Befehl: Impulssteuerung	52
3.1.11	Befehl: Zeitverzögerung	53
3.1.12	Befehl: Bewegung bis Impuls	54
3.1.13	Befehl: Starte angeschlossene Interface-Karte	55
3.2	Ergänzender Befehlsumfang Interface-Karte 5.0	56
3.2.1	Befehl: 3D-Linearinterpolation	56
3.2.2	Befehl: Zirkularinterpolation	57
3.3	Ergänzender Befehlsumfang bei Interface-Karten mit E/A-Erweiterung	59
3.3.1	Befehl: Ausgangsport setzen	59
3.3.2	Befehl: Eingangsport lesen	61
3.4	Ergänzender Befehl in Verbindung mit einer Programmwahleinheit	62
3.4.1	Befehl: Tastaturabfrage	62
4	Fehlermeldungen	64
4.1	Fehlermeldungen isel -Prozessorkarten	64
4.2	Fehlermeldungen PAL-PC	68

1 Einleitung

Die Beschreibung des CNC-Betriebssystems 5.x ist eine umfassende Dokumentation aller Befehle der **isel**-Prozessorkarten. Die beschriebenen Befehle gelten für folgende **isel**-Steuerungssysteme:

- **isel**-Interface-Karte (bis Softwarestand 5.x)
- **isel**-CNC-Controller C 116, C 142/1, C 116-4, C 142-4
- **isel**-CNC-Steuerung C 10C, C 10C-E/A
- **isel**-Integrierte Technologien IT 108, IT 116
- **isel**-Bearbeitungszentrum EP 1090
- **isel**-Bearbeitungszentrum EP 1090/4

Das CNC-Betriebssystem unterstützt die Positionierung von maximal drei Schrittmotor-Antriebsachsen. Zusätzlich zu den Positionierparametern verarbeitet das Betriebssystem unterschiedliche Steuer- und Kontrollfunktionen.

Durch die Zusammenfassung aller Steuerungssysteme (hier Prozessorkarte genannt) innerhalb eines Betriebssystems sind evtl. Einschränkungen in Bezug auf die Programmierung der einzelnen Geräte zu berücksichtigen. Diese sind in den jeweiligen Hardware-Beschreibungen vermerkt.

Die in der Beschreibung verwendeten Programmbeispiele sind auf das Maximal-System ausgelegt. Somit müssen Sie ggf. Anpassungen im Bereich der Positionierbefehle vornehmen.

Der Begriff PAL-PC wird sowohl im Zusammenhang mit der Programmiersprache PAL-PC als auch mit dem Software-Ankopplungsmodul PAL-EP verwendet.

Zur direkten Programmierung der Prozessorkarten steht Ihnen ein festgelegtes Übertragungsformat zur Verfügung. Als Beispiel ist in dieser Dokumentation eine Programmierung in der Programmiersprache BASIC aufgelistet.

2 Befehlsaufbau der DNC-Befehle

Im DNC-Modus werden die von einem Steuerrechner übergebenen Datensätze bzw. Befehle direkt ausgewertet und ausgeführt. Dafür ist zu Beginn der Datenkommunikation eine sogenannte Initialisierung notwendig. Sie besteht aus dem Dateneröffnungszeichen @, der Gerätenummer (Standard = 0) und der Anzahl der zu verfahrenen Achsen.

Anschließend werden der Prozessorkarte die Programmschritte einzeln übergeben und von ihr direkt ausgeführt.

Zur Überprüfung der Datenübertragung bzw. Meldung von aufgetretenen Fehlern werden über die Schnittstelle entsprechende ASCII-Zeichen an den Steuerrechner zurückgesendet. Dieses sogenannte Software-Handshake-Verfahren kann zu zwei unterschiedlichen Zeitpunkten realisiert werden:

1. Die Prozessorkarte setzt direkt nach Empfang des abzuarbeitenden Datensatzes das Quittierungs-/Fehlerzeichen ab.
2. Die Prozessorkarte arbeitet den übersendeten Befehlssatz ab, und meldet anschließend das Quittierungs-/Fehlerzeichen zurück.

Der gewünschten Modus wird durch Groß-/Kleinschreibung des Befehlszeichens unterschieden. Großbuchstaben führen zur Rückmeldung nach Abarbeitung der Befehle, Kleinbuchstaben zur direkten Rückmeldung.

Im folgenden wird der Befehlsumfang der Interface-Karte 4.0 beschrieben. Ergänzungen durch erweiterte Hardware (z. B. Interface-Karte 5.0) befinden sich am Ende des Kapitels.

Der in den Beispiel-Programmen benannte Terminalmode ist eine Funktion der **isel**-Software PAL-PC. Er wird im PAL-PC durch die Funktionstaste F2 eingeschaltet und stellt eine direkte Verbindung zwischen Bildschirm und Interface-Karte her.



Weitere Informationen hierzu finden Sie in der Anleitung zum PAL-PC, Kapitel X1 Kommunikationsfenster.

2.1 Grundbefehlssatz ab Prozessorkarte 4.0

2.1.1 Befehl: Achsenanzahl setzen

Zweck	Durch Übergeben der Achsenanzahl wird die Prozessorkarte neu initialisiert. Dabei wird der Datenspeicher gelöscht und zur Speicheroptimierung entsprechend der Anzahl der Achsen neu eingeteilt.										
Aufbau	<p><GN> <Achsen></p> <p><GN> = Gerätenummer, Standard = 0</p> <p><Achsen> = Achsenangabe, s. u.</p>										
Anwendung	@07										
Erläuterung	Die Karte wird durch @0 adressiert; der anschließende Zahlenwert beinhaltet die Achsenkonfiguration.										
	<table> <tr> <th>Achsenangabe</th><th>Wert</th></tr> <tr> <td>x</td><td>1</td></tr> <tr> <td>xy</td><td>3</td></tr> <tr> <td>xz</td><td>5</td></tr> <tr> <td>xyz</td><td>7</td></tr> </table>	Achsenangabe	Wert	x	1	xy	3	xz	5	xyz	7
Achsenangabe	Wert										
x	1										
xy	3										
xz	5										
xyz	7										
Beschränkung	Nicht zulässig sind die Kombinationen @00, @02, @04, @06, sowie @08 und @09.										



Programmierbeispiel

PAL-PC

#axis xyz;

GW-BASIC

```

100 open"com1:9600,N,8,1,DS,CD"as #1
110print#1,"@07":gosub 1000
120 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print Karte meldet Fehler : ;a$
1030 stop

```



Der Befehl *Achsenanzahl setzen* löscht alle im RAM vorhandenen Daten, auch wenn durch die integrierte Option *Memory Back-up* die Daten nach Wegfall der Versorgungsspannung im RAM-Speicher der Prozessorkarte gespeichert waren.

2.1.2 Befehl: Referenzfahrt

Zweck	Die Prozessorkarte verfährt alle angegebenen Achsen an ihre Nullpunkte (Referenzpunkte). Die Referenzpunkte der Achsen liegen bei isel -Systemen an der dem Motor zugewandten Seite; positive Bewegungen finden in der Richtung vom Motor weg statt.																									
Aufbau	@<GN>R<Achsen> <GN> = = Gerätenummer, Standard = 0 <Achsen> = ganzzahliger Wert zwischen 1 und 7																									
Anwendung	@0R7 oder @0r7																									
Erläuterung	<p>Die Karte wird durch @0 adressiert. R gibt an, dass eine Referenzfahrt ausgeführt werden soll. Der anschließende Zahlenwert definiert die Achsen, die eine Referenzfahrt durchführen sollen:</p> <table><tr><td>x</td><td>=</td><td>1</td><td>xy</td><td>=</td><td>3</td></tr><tr><td>y</td><td>=</td><td>2</td><td>yz</td><td>=</td><td>6</td></tr><tr><td>z</td><td>=</td><td>4</td><td>xyz</td><td>=</td><td>7</td></tr><tr><td>xz</td><td>=</td><td>5</td><td></td><td></td><td></td></tr></table> <p>Die Reihenfolge der Ausführung ist dabei wie folgt festgelegt:</p> <p>—> Z-Achse —> Y-Achse —> X-Achse</p> <p>Dies gilt auch, wenn mit dem Ebenenbefehl eine andere Achse als Werkzeugachse bestimmt wurde. Hier kann ggf. die getrennte Referenzfahrt der Achsen Kollisionen mit dem Werkstück verhindern.</p> <p>Nach erfolgter Referenzfahrt sendet die Prozessorkarte ihr Quittierungszeichen und wartet auf die nächsten Befehle. Besteht die Notwendigkeit einer sofortigen Rückmeldung, müssen Sie statt R ein r verwenden. Die Prozessorkarte kann jedoch in jedem Fall erst wieder Befehle verarbeiten, nachdem die Referenzfahrt durch die Mechanik ausgeführt worden ist.</p>		x	=	1	xy	=	3	y	=	2	yz	=	6	z	=	4	xyz	=	7	xz	=	5			
x	=	1	xy	=	3																					
y	=	2	yz	=	6																					
z	=	4	xyz	=	7																					
xz	=	5																								
Beschränkung	<p>Den Befehl können Sie verwenden, nachdem eine Initialisierung der Prozessorkarte durch den Befehl <i>Achsenanzahl setzen</i> erfolgt ist und beschränkt sich auf die dort vorgegebene Achsenkonfiguration. Bei falscher Achsangabe erfolgt die Fehlerrückmeldung 3. Befindet sich die Karte im 3D-Modus, schaltet der Befehl wieder in die 2,5D-Ausführung zurück.</p>																									



Programmierbeispiel

PAL-PC

```
#axis xyz;  
reference xyz;
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"ast#1  
110 print#1,"@07":gosub 1000  
120 print#1,"@0R7":gosub 1000  
130 stop  
1000 if loc(1)<1 then goto 1000  
1010 a$=input$(1,1)  
1015 if a$="0" then return  
1020 print "Karte meldet Fehler:";a$  
1030 stop
```



Bei nicht angeschlossenem Referenzschalter gibt die Prozessorkarte permanent Impulse aus. Durch zweimaliges Betätigen des Stopp-Tasters besteht jedoch die Möglichkeit, die Referenzfahrt der entsprechenden Achse abubrechen.

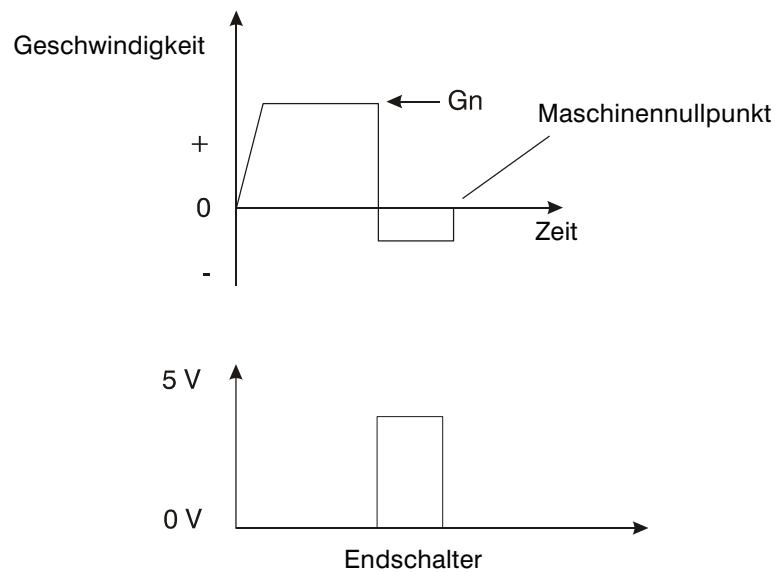


Bild 1: Verlauf einer Referenzfahrt (zum Maschinennullpunkt)

2.1.3 Befehl: Referenzgeschwindigkeit setzen

Zweck	Der Befehl definiert - getrennt für jede Achse - die Geschwindigkeit, mit der eine Referenzfahrt ausgeführt wird. Dabei handelt es sich nur um die Geschwindigkeit, mit der die Achse in negativer Richtung auf den Motor zuläuft; die Geschwindigkeit aus dem Schalter heraus lässt sich nicht beeinflussen (vgl. Referenzfahrt).		
Aufbau	@<GN>d<Gx>	(x)	
	@<GN>d<Gx>, <Gy>	(X-y)	
	@<GN>d<Gx>, <Gz>	(X-z)	
	@<GN>d<Gx>, <Gy>, <Gz>	(X-Y-z)	
	<GN>	=	Gerätenummer, Standard = 0
	<Gx>	=	Referenzgeschwindigkeit x
	<Gy>	=	Referenzgeschwindigkeit y
	<Gz>	=	Referenzgeschwindigkeit z
	<Gx>, <Gy>, <Gz>	=	ganze Zahl zwischen 30 und 10 000 Hz
Anwendung	@0d2500	(1 Achse)	
	@0d2400,3000	(2 Achsen)	
	@0d1000,3000,9000	(3 Achsen)	
Erläuterung	Werden der Prozessorkarte keine Informationen zur Referenzgeschwindigkeit übergeben, erfolgt die Ausführung mit einem Default-Wert von 2 000 Schritten/s. Ein geänderter Wert bleibt beim Ausschalten erhalten, falls die Option <i>Memory-Back-up</i> eingebaut ist.		
Beschränkung	-		



Programmierbeispiel

PAL-PC

```
#axis xy;
#ref_speed 3000,5000;
```

GW-BASIC

```
100open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0d3000,5000":gosub 1000
130 print#1,"@0R3":gosub 1000
140 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler: ";a$
1030 stop
```



Eine zu hoch gewählte Referenzgeschwindigkeit kann in Verbindung mit einer großen Spindelsteigung durch die vorhandene Massenträgheit zu einer Beschädigung der Referenzschalter führen.

Die Prozessorkarte benötigt eine Schalthysterese des angeschlossenen Nullagenschalters (bei Anschluss von elektronischen Nullsensoren zu beachten).

2.1.4 Befehl: Bewegung relativ

Zweck	Die Prozessorkarte gibt entsprechend der übergebenen Schrittzahl und Schrittgeschwindigkeit für jede Leistungsstufe eine Impulskette aus. Die Verfahrbewegung wird sofort ausgeführt oder gespeichert.	
Aufbau	<p>@<GN>A<Sx>,<Gx>,<Sy>,<Gy>,<Sz1>,<Gz1>,<Sz2>,<Gz2></p> <p><GN> = Gerätenummer, Standard = 0</p> <p><Sx> = Schrittzahl x, Wert zwischen 0 und +/- 8 388 607</p> <p><Gx> = Geschwindigkeit x, Wert zwischen 30 und 10 000</p> <p>.</p> <p>.</p> <p><Gz2> = Geschwindigkeit Z-Achse (2. Bewegung)</p>	
Anwendung	@0A 5000,900	(nur X-Achse)
	@0A 50,900,20,9000	(X- und Y-Achse)
	@0A 50,900,20,900,-20,900	(X- und Z-Achse)
	@0A 30,800,10,900,4,90,-4,30	(X-, Y- und Z-Achse)
Erläuterung	<p>Die Prozessorkarte wird durch @0 adressiert; A gibt an, dass eine Bewegung erfolgen soll. Die Prozessorkarte erwartet nun für jede Achse ein Zahlenpaar bestehend aus Schrittzahl und Geschwindigkeit.</p> <p>Die Bewegung erfolgt im Relativmaß, d. h. bezogen auf die letzte Position. Die Anzahl der Angaben muss mit der Achsenzahl übereinstimmen, d. h. ein Parameterpaar bei X-Betrieb, zwei Parameterpaare bei XY-Betrieb, drei Parameterpaare bei XZ-Betrieb und vier Parameterpaare für XYZ-Betrieb. Die einzelnen Zahlen müssen durch Kommata getrennt sein.</p> <p>Für die Z-Achse werden zwei Zahlenpaare erwartet, da für Bearbeitungsanwendungen sehr häufig die Situation "Fahren, Werkzeug absenken und anschließend anheben" vorkommt.</p> <p>Im 2,5D-Interpolationsbetrieb werden zuerst die Bewegungen der X- und Y-Achse verfahren (linear interpoliert), anschließend wird die Z-Achse zuerst um die in z1 angegebenen und dann um die in z2 angegebenen Werte verfahren. Diese Interpolationszuordnung lässt sich bei 2D-Betrieb mit dem Ebenenbefehl ändern.</p> <p>Besteht die Notwendigkeit, nur eine Achse zu bewegen, sind dennoch für alle initialisierten Achsen Werte zu übertragen. Dabei ist für die Schrittzahlen der nicht bewegten Achsen 0 und für die Geschwindigkeit ein Wert zwischen 30 und 10 000 anzugeben.</p> <p>Die Prozessorkarte meldet sich nach erfolgter Ausführung mit dem Handshake-Character (0). Durch den unterschiedlichen Befehlscode a und A können Sie zwischen einer Quittierungsmeldung direkt nach der Übertragung und einer Rückmeldung nach Ausführungsende wählen. Die Prozessorkarte kann jedoch in jedem Fall erst wieder Befehle verarbeiten, nachdem die Ausführung beendet ist.</p>	

Beschränkung Den Befehl können Sie erst verwenden, nachdem die Achsenanzahl gesetzt worden ist. Die Prozessorkarte prüft nicht, ob die Bewegung den zulässigen Bereich der angeschlossenen Mechanik verlässt.



Programmierbeispiel

PAL-PC

```
#axis xy;  
move50(500),300(900);
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as#1  
110 print#1,"@03":gosub 1000  
    120 print#1,"@0A50,500,300,900":gosub 1000  
    130 print#1,"@0A20,200,-30,900":gosub 1000  
140 stop  
1000 if loc(1)<1 then goto 1000  
1010 a$=input$(1,1)  
1015 if a$="0" then return  
1020 print "Karte meldet Fehler : ";a$  
1030 stop
```



Im 2,5D-Interpolationsbetrieb wird die Geschwindigkeitsangabe der Achse mit dem längsten Weg als Bahngeschwindigkeit übernommen, die Geschwindigkeit der anderen Achse entsprechend dem Wegeverhältnis angepasst.
Im Gegensatz dazu wird im 3D-Interpolationsbetrieb die Geschwindigkeitsangabe der X-Achse als Vorgabewert für die Bahngeschwindigkeit herangezogen.

2.1.5 Befehl: Bewegung zur Position (move to)

Zweck	Die Prozessorkarte fährt mit den angegebenen Geschwindigkeiten an die angegebene Position. Die Verfahrbewegung wird sofort ausgeführt.	
Aufbau	<p>@<GN>M<Sx>,<Gx>,<Sy>,<Gy>,<Sz1>,<Gz1>,<Sz2>,<Gz2></p> <p><GN> = Gerätenummer, Standard = 0 <Sx> = Weginformation X-Achse <Gx> = Geschwindigkeit X-Achse . . <Sz2> = bei Absolut-Positionierung = 0 <Gz2> = Geschwindigkeit Z-Achse (2. Bewegung)</p>	
Anwendung	<p>@0M 5000,900 (X-Achse) @0M 50,900,20,9000 (X- und Y-Achse) @0M 50,900,20,900,0,21 (X- und Z-Achse) @0M 30,800,10,900,4,90,0,21 (X-, Y- und Z-Achse)</p>	
Erläuterung	<p>Die Prozessorkarte wird durch @0 adressiert. M gibt an, dass eine Absolut-Position folgt. Aus Kompatibilitätsgründen zum relativen Positionierbefehl werden auch hier für die Z-Achse zwei Zahlenpaare erwartet. Die zweite Positionierungsangabe der Z-Position muss jedoch Null sein und wird ignoriert. Die Prozessorkarte meldet sich nach erfolgter Ausführung mit dem Handshake-Character. Soll die Rückmeldung sofort erfolgen, müssen Sie m statt M verwenden. Die Prozessorkarte kann jedoch in jedem Fall erst wieder Befehle empfangen, nachdem die Ausführung beendet ist.</p>	
Beschränkung	<p>Den Befehl können Sie erst verwenden, nachdem die Achsenanzahl gesetzt worden ist. Der Befehl kann nicht während der Bearbeitung gespeicherter Befehle übersandt werden. Die Prozessorkarte prüft nicht, ob die Bewegung den zulässigen Bereich der angeschlossenen Mechanik verlässt. Wollen Sie den Befehl speichern, müssen Sie die Prozessorkarte vorher in den Input-Modus setzen (vgl. Input) und den Befehlscode m benutzen.</p>	



Programmierbeispiel

PAL-PC

```
#axis xy;  
reference xy;  
moveto 50(500),300(900);  
moveto 20(200),30(900);  
moveto 0(21),00(2000);  
stop.
```

GW-BASIC

```
100 open"com1.9600,N,8,1,DS,CD"as #1  
110 print#1,"@03":gosub 1000  
    120 print#1,"@0M50,500,300,900"  
125 gosub 1000  
    130 print#1,"@0M20,200,30,900"  
135 gosub 1000  
    140 print#1,"@0M 0,21,700,2000"  
145 gosub 1000  
150 stop  
1000 if loc(1)<1 then goto 1000  
1010 a$=input$(1,1)  
1015 if a$="0" then return  
1020 print "Karte meldet Fehler : ";a$  
1030 stop
```

2.1.6 Befehl: Positionsabfrage

Zweck Die Prozessorkarte gibt die momentane Sollposition aller Achsen an den übergeordneten Rechner zurück.

Aufbau @<GN>P
 <GN> = Gerätenummer, Standard = 0

Anwendung @0P

Erläuterung Die Prozessorkarte wird durch @0 adressiert. P gibt an, dass eine Positionsanfrage erfolgt. Die Prozessorkarte bestätigt dies mit dem Handshake-Character und gibt anschließend im hexadezimalen Format die Positionswerte aller Achsen aus (insgesamt 19 Byte = 18 hexadezimale Ziffern + 1 x Handshake)

Der Aufbau der rückgemeldeten Position ist folgendermaßen:

0 000010 002000 FFFFFFE
 A B C

A Position x, hexadezimal im 2er-Komplement, im Beispiel der Wert 16 dezimal.

B Position y, hexadezimal im 2er-Komplement, im Beispiel der Wert 8096 dezimal.

C Position z, hexadezimal im 2er-Komplement, im Beispiel der Wert - 2 dezimal.

Beschränkung Der Befehl kann nur verwendet werden, wenn keine Verfahrbewegung stattfindet (wenn sich die Anlage im Stopp-Zustand befindet).
 Der Befehl kann nicht während der Bearbeitung gespeicherter Befehle übersandt werden.
 Die Prozessorkarte kann nicht prüfen, ob die Sollposition der aktuellen Position der Mechanik entspricht, da kein Regelkreis vorhanden ist.



Programmierbeispiel

PAL-PC

GW-BASIC

(Terminalmode)

@0P

-



Es werden immer die Positionen aller drei Achsen durch die Funktion zurückgegeben, unabhängig von der Anzahl der definierten Achsen.

Die Interface-Karte sendet die entsprechenden ASCII-Zeichen mit der eingestellten Übertragungsgeschwindigkeit ohne, mittels Hardware-Handshake, eine Bestätigung des empfangenden Rechners zu erwarten.

2.1.7 Befehl: Nullpunktverschiebung

Zweck	<p>Die Prozessorkarte speichert die momentane Position als virtuellen Nullpunkt für die angegebene(n) Achse(n). Die nächsten <i>Verfahre absolut</i>-Anweisungen berücksichtigen diesen virtuellen Nullpunkt als neuen Bezugspunkt.</p>																						
Aufbau	<p>@<GN>n<Achsen> <GN> = Gerätenummer, Standard = 0 <Achsen> = ganzzahliger Wert zwischen 1 und 7</p>																						
Anwendung	@0n7	@0n1																					
	.	.																					
	.	.																					
Erläuterung	<p>Die Karte wird durch @0 adressiert. n gibt an, dass eine Nullpunktverschiebung vorgenommen werden soll. Nach dem Befehl werden dem Rechner die Achsen mitgeteilt, für die eine Nullpunktverschiebung durchgeführt werden soll. Hierbei gilt die Zuordnung x = 1, y = 2, z = 4. Soll für mehrere Achsen eine Nullpunktverschiebung durchgeführt werden, müssen obige Werte addiert werden:</p> <table> <tr> <th>Achsen</th><th>Wert</th><th>Achsen</th><th>Wert</th></tr> <tr> <td>x</td><td>1</td><td>xy</td><td>5</td></tr> <tr> <td>y</td><td>2</td><td>yz</td><td>6</td></tr> <tr> <td>z</td><td>4</td><td>xyz</td><td>7</td></tr> <tr> <td>xy</td><td></td><td>3</td><td></td></tr> </table> <p>Der Rechner meldet sich nach erfolgter Ausführung mit der Rückmeldung (vgl. Software-Handshake).</p>			Achsen	Wert	Achsen	Wert	x	1	xy	5	y	2	yz	6	z	4	xyz	7	xy		3	
Achsen	Wert	Achsen	Wert																				
x	1	xy	5																				
y	2	yz	6																				
z	4	xyz	7																				
xy		3																					
Beschränkung	<p>Der virtuelle Nullpunkt hat nur für den Befehl <i>Verfahre absolut</i> eine Bedeutung. Relativpositionierung wird vom virtuellen Nullpunkt nicht beeinflusst, da hier ein Fahrvektor angegeben wird.</p>																						



Programmierbeispiel

PAL-PC

```
#axis xy;
#elev 4,4;
moveto 80(900),8(900);
null xy;
moveto 2(900),4(990);
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0r3":gosub 1000
130 print#1,"@A1000,2000,2000,2000":gosub 1000
140 print#1,"@0n3":gosub 1000
150 print#1,"@M100,2000,100,2000":gosub 1000
160 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Der virtuelle Nullpunkt wird durch Referenzfahrt wieder in den Anlagennullpunkt gelegt.

2.1.8 Befehl: Ebenenwahl

Zweck

2,5D interpolierende Prozessorkarten (z. B. Interface-Karte 4.0) können nur zwei von drei Achsen interpolieren. Im Einschaltzustand sind dies die X- und Y-Achse. Durch den *Ebenenwahl*-Befehl besteht jedoch die Möglichkeit, jede andere Ebenenkonfiguration als Hauptebene zu definieren. Die übrigbleibende dritte Achse wird als Werkzeugachse aufgefasst und anschließend, also nach der Positionierung der Hauptachsen verfahren.

Aufbau

@<GN>e<Ebene>

<GN>	=	Gerätenummer, Standard = 0
<Ebene>	=	Zahl zwischen 0 und 2
0	=	xy
1	=	xz
2	=	yz

Anwendung

@0e1	Schalte auf xz-Interpolation
@0e0	Schalte auf xy-Interpolation

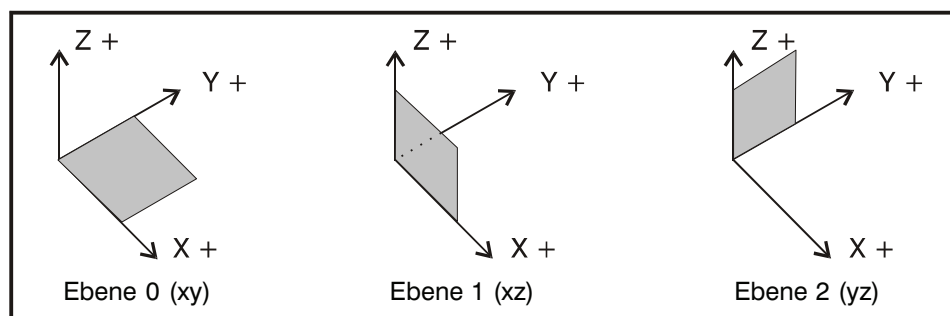


Bild 2: Umschaltbare Interpolationsebenen

- Erläuterung** Um hohe Geschwindigkeiten zu erzielen (eine maximale Achsen-
geschwindigkeit von 10 kHz entspricht bei einer 5 mm-Spindelsteigung
und einem 45° Grad Vektor einer Vektorgeschwindigkeit von 175 mm/s
im Halbschrittbetrieb), kann die Prozessorkarte innerhalb dieser Zeit
nur die Geschwindigkeitsverhältnisse von zwei Achsen gegeneinander
berechnen. Der Ebenenbefehl erlaubt eine Umschaltung zwischen den
Interpolationsebenen ohne Geschwindigkeitsverlust.
- Beschränkung** Falls eine andere Interpolationsebene als XY gewählt wird, sollte für die
zweite Bewegung der Bearbeitungsachse (die nicht interpolierte
Achse) Null als Schrittzahl übergeben werden.



Programmierbeispiel

PAL-PC

```
#axis xyz;
line yz;
move 20(1000),30(1000),
      33(1000),0(30);
stop.
```

GW-BASIC

```
100 open"com1.9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
120 print#1,"@0e2"
125 gosub 1000
130 print#1,"@0M20,200,30,900,33,900, 0,21"
135 gosub 1000
140 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Im obigen Beispiel werden die Y- und Z-Achsen interpoliert (geradlinig zum Zielpunkt
verfahren), die X-Achse nachgeführt.
Die Reihenfolge der Referenzfahrt wird durch den Ebenenwahlbefehl nicht beeinflusst.
Falls eine geänderte Referenzfahrtreihenfolge benötigt wird, dürfen keine Referenzbefehle
mit Achsenkombinationen übergeben werden.

2.1.9 Befehl: Peek (Lesen von Speicheradressen)

Zweck	Der <i>Peek</i> -Befehl ermöglicht, den Inhalt einer Speicherzelle der Prozessorkarte sowohl im Datenspeicher als auch im Festwertspeicher über die serielle Schnittstelle abzufragen.
Aufbau	<p>@<GN>c<Addr> (Festwertspeicher)</p> <p>@<GN>b<Addr> (Schreib-/Lesespeicher)</p> <p><GN> = Gerätenummer, Standard = 0</p> <p><Addr> = Adresse zwischen 0 und 65 536</p>
Anwendung	<p>@0c 2048</p> <p>@0b 4711</p>
Erläuterung	Die Karte wird durch @0 adressiert. c gibt an, dass ein Wert aus dem Festwertspeicher gelesen werden soll. 2048 gibt die Adresse des zu lesenden Wertes an. Der Rechner antwortet mit dem Software-Handshake gefolgt von zwei Zeichen, die einen Hexadezimalwert angeben, der dem Inhalt der Speicherzelle entspricht. Soll ein Wert aus dem Datenspeicher gelesen werden, ist statt c der Befehlscode b zu verwenden.
Beschränkung	-



Programmierbeispiel

PAL-PC
(Terminalmode)

@0b 65531

GW-BASIC

-



Dieser Befehl wird in erweiterter Form in Verbindung mit einer E/A-Erweiterungseinheit eingesetzt (vgl. Kapitel 2.3.3, Speicherbare Befehle: *Eingangsport lesen*).

2.1.10 Befehl: Poke (Beschreiben von Speicheradressen)

Zweck	Der <i>Poke</i> -Befehl erlaubt es, den Speicherinhalt des Datenspeichers einer Prozessorkarte zu ändern
Aufbau	<p>@<GN>B<Addr>,<Data></p> <p><GN> = Gerätenummer, Standard = 0</p> <p><Addr> = Adresse zwischen 0 und 65 535</p> <p><Data> = Wert zwischen 0 und 255</p>
Anwendung	@0B 33211,128
Erläuterung	Die Karte wird durch @0 adressiert. B gibt an, dass ein Wert in den Speicher geschrieben werden soll. 33211 gibt die Adresse des zu schreibenden Wertes an. 128 ist der neue Wert dieser Speicherzelle. Der Rechner bestätigt die Ausführung mit dem Software-Handshake.
Beschränkung	Der Befehl prüft nicht, ob ein am Datenbus angeschlossenes Gerät die Daten korrekt übernommen hat.



Programmierbeispiel

PAL-PC

GW-BASIC

(Terminalmode)

@0B33211,128

-



Der Befehl sollte nicht verwendet werden, um interne Kartenparameter zu ändern, da sich deren Adresse ohne Mitteilung ändern kann. Der Befehl sollte nicht mit Adressen kleiner 32767 benutzt werden, da diese Adressen als Datenspeicher von der Prozessorkarte genutzt werden.

2.1.11 Befehl: Batterie-RAM löschen

Zweck	Der Befehl löscht alle Daten des RAM-Speichers auch wenn sie durch die Option <i>Memory Back-up</i> quasi-permanent gespeichert sind. Damit werden auch alle eventuell vorhandenen Angaben über Referenzgeschwindigkeiten, Achsen usw. zurückgesetzt.
Aufbau	<p>@<GN>k (direkt)</p> <p><GN> = Gerätenummer, Standard = 0</p>
Anwendung	@Ok
Erläuterung	Die Karte wird durch @0 adressiert. k gibt an, dass das Batterie-RAM gelöscht werden soll. Der Rechner meldet sich nach erfolgter Ausführung mit der Rückmeldung (vgl. Software-Handshake).



Programmierbeispiel

PAL-PC

(Terminalmode)

@Ok

GW-BASIC

```

100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@Ok":gosub 1000
120 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop

```



Sollte sich das RAM auf diese Weise nicht mehr löschen lassen, ist es kurzzeitig aus seiner Fassung zu entnehmen und anschließend wieder einzusetzen.

2.1.12 Befehl: CR/LF setzen

Zweck	<p>Einige Computer müssen beim Datenempfang an der seriellen Schnittstelle ein Delimiter (Begrenzungs)-Zeichen erhalten.</p> <p>Dies ist eine betriebssystem-spezifische Eigenart einiger Prozessrechner (z. B. DEC VAX, HP-Prozessrechner).</p> <p>Das Delimiter-Zeichen wird vom Prozessrechner benötigt, um am Ende der Datenübermittlung einen Interrupt für den empfangenen Prozess auslösen zu können.</p> <p>Falls Sie einen MS-DOS-Rechner (IBM-PC, XT, AT o. ä.) verwenden, sollte dieser Befehl nicht verwendet werden.</p>
Aufbau	<p>@<GN>C<STATUS> (direkt)</p> <p><GN> = Gerätenummer, Standard = 0</p> <p><STATUS> = 0 = AUS (Standard), 1 = EIN</p>
Anwendung	@0C1
Erläuterung	<p>Die Karte wird durch @0 adressiert. C gibt an, dass der Software-Handshake geändert werden soll. Die Karte meldet sich bei allen Befehlen dann mit der Folge :</p> <p style="text-align: center;">Fehler CHR(13) CHR(10)</p>
Beschränkung	<p>Die Karte lässt sich anschließend nur noch mit diesem neuen Protokoll ansprechen. Um in das alte Protokoll zurückzuschalten, muss das neue Protokoll zur Übergabe verwendet werden.</p>

Ein Programmbeispiel ist nicht verfügbar, da weder PAL-PC noch GW-BASIC auf diesen Prozessrechnern verfügbar sind.

2.1.13 Befehl: Gerätenummer setzen

Zweck	Die Gerätenummer der Prozessorkarte (<GN>) wird geändert. Zulässig sind Nummern zwischen 0 und 9 (chr(48) ... chr(58)). Die neue Gerätenummer bleibt bis zum Ausschalten des Gerätes aktiv.	
Aufbau	@<GN>G<GNneu>	(direkt)
	<GN>	= Gerätenummer, Standard = 0
	<GNneu>	= Zeichen zwischen 0 und 9
Anwendung	@0G1	
Erläuterung	Die Karte wird durch @0 adressiert. G gibt an, dass eine neue Gerätenummer zur Adressierung der Karte verwendet werden soll. Nach G erwartet die Karte die neue Gerätenummer. Der Rechner meldet sich nach erfolgter Speicherung mit der Rückmeldung (vgl. Software-Handshake).	
Beschränkung	Die Karte lässt sich anschließend nur noch mit dieser Gerätenummer ansprechen. Die Karte prüft nicht, ob eine zulässige Gerätenummer übergeben wird.	



Programmierbeispiel

PAL-PC

(Terminalmode)

@0G1

GW-BASIC

```

100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1, "@03":gosub 1000
120 print#1,"@0G1":gosub 1000
130 print#1,"@1i":gosub 1000
140 print#1,"m 8000,900,800,900":gosub 1000
150 print#1,"n 3":gosub 1000
160 print#1,"m 200,900,400,990":gosub 1000
170 print#1,"9":gosub 1000
180 print#1,"@1S":gosub 1000
190 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop

```



Um PAL-PC eine Prozessorkarte mit geänderter Gerätenummer anzusprechen, wird der Befehl #GN verwendet. Die Prozessorkarte kann anschliessend nur noch mit dieser Gerätenummer angesprochen werden.

2.1.14 Befehl: TRACE (Einzelschrittausführung)

Zweck	Die Prozessorkarte führt die gespeicherten Befehle einzeln aus. Nach jedem Befehl wird auf ein Zeichen an der seriellen Schnittstelle gewartet und der Befehlszählerstand mit allen zugehörigen relevanten Parametern ausgegeben.
Aufbau	@<GN>t <GN> = Gerätenummer, Standard = 0
Anwendung	@0t
Erläuterung	Die Prozessorkarte führt die Befehle wie bisher aus; vor jedem Befehl wird jedoch der Befehlszählerstand als Integer-Zahl ausgegeben. Nach dem Befehlszählerstand wird die Befehlsnummer und die Operationskonstante mit zugehörigen Daten ausgegeben. Die Zeile wird mit CR abgeschlossen. Der Rechner wartet dann auf ein Zeichen an der Schnittstelle. Anschließend wird der Befehl ausgeführt. Die Funktion verhält sich bei jedem auszuführenden Befehl wie folgt: (A): Der Trace-String wird ausgegeben (s. u.). Es wird auf ein Zeichen gewartet. Falls Zeichen = 127 dann wird μ P-Reset ausgelöst. Befehl wird ausgeführt. Falls Befehl = Datenfeldende, dann Schluss. Sonst nächster Befehl, nach (A).

Der bei jedem Befehl übergebene Trace-String hat folgenden Aufbau

Aufbau	<u>01234</u> <u>00001</u> <u>30</u> <u>000001</u> <u>FE87</u> ... <u>FFFF01</u> <u>FE01</u>
	A B C D E F G

- A Speicherzeiger - Er gibt an, wo der Befehl im Speicher der Prozessorkarte abgelegt ist.
- B Befehlszähler - Er gibt die Nummer des aktuellen NC-Befehles an.
- C NC-Befehlscode - Er gibt den auszuführenden Befehl an.
Die Angabe ist in Hexadezimal und bezieht sich auf den ASCII-Wert des Befehlscodes. Im Beispiel oben ist der Befehl
0 = Bewegung relativ gespeichert.
- D Befehlsparameter der X-Achse - Im obigen Beispiel ist die 24-Bit-Hexadezimal-Darstellung des Verfahrensweges in der 2er-Komplement-Darstellung.
- E Geschwindigkeitswert der X-Achse - Um die Geschwindigkeit rückzuwandeln, kann der Bruch $921600/(HI-Byte \cdot (256-LO-Byte))$ gebildet werden.
- F Befehlsparameter wie D, jedoch für die z2-Angabe.
- G Geschwindigkeitsangabe für die z2-Angabe.

Bei anderen Befehlen werden die Parameter in der übergebenen Reihenfolge entweder als Zeichen oder als 2er-Komplement abgelegt.



Bei den Synchronisations-Befehlen ist zuerst das Zeichen für die Trace-Funktion und anschließend das Synchronisations-Zeichen zu übergeben. Die Zeichen <Leertaste>, <TAB> und <Linefeed> können nicht verwendet werden, um die Einzelschrittausführung fortzuschalten.

Um den Trace-Modus zu beenden, müssen Sie die Prozessorkarte ausschalten oder bei der Anfrage auf ein Zeichen DEL (char(127)) übergeben.

Die Parameter-Reihenfolge und -Speicherung kann sich im Rahmen der technischen Weiterentwicklung ohne Mitteilung bei Software-Versionen ändern.

Die Anzahl der rückgesendeten Befehlsparameter entspricht der gewählten Achsenzahl.

2.1.15 Befehl: Selbsttest

Zweck	Diese Anweisung löst einen Selbsttest der Prozessorkarte aus. Im Gegensatz zum Selbsttest der durch Betätigen des Starttasters eingeleitet wird, bewirkt der Befehl nur die Ausführung des ersten Teiles der Testroutine, nicht den Verfah- und Schnittstellentest.	
Aufbau	@<GN>?	(direkt)
	<GN>	= Gerätenummer, Standard = 0
Anwendung	@0?	
Erläuterung	Die Karte wird durch @0 adressiert. ? gibt an, dass ein Selbsttest durchgeführt werden soll. Die Karte testet den Speicherbereich, Prozessor und Prozessorregister sowie interne Speicherbereiche. Anschließend werden einige Prozessorkartenbezogene Variablen und eine Prüfsumme ausgegeben. Um den erweiterten Selbsttest auszuführen, müssen Sie den Starttaster festhalten und das Gerät einschalten. Um den Selbsttest zu beenden, müssen Sie das Gerät ausschalten, vorher können keine anderen Befehle übergeben werden!	



Programmierbeispiel

PAL-PC

(Terminalmode)

@0?

GW-BASIC

-



Zur weiteren Befehlsübergabe an die Prozessorkarte muss der Selbsttest abgeschlossen sein. Anderenfalls wird ein Listen-Zeichen (Fehler @) als Fehlermeldung gesendet. PAL-PC wird in diesem Fall Fehler 164 rückgemeldet.

2.2 Ergänzender Befehlsumfang der Interface-Karte 5.0

2.2.1 Befehl: 3D-Linearinterpolation

Zweck	Die Interface-Karte 5.0 erweitert die 2,5D-Interpolation des Standard-Betriebssystems auf eine 3D-Interpolation (räumlich). Mit diesem Befehl können Sie die Interpolation gezielt ein- und ausschalten.
Aufbau	@<GN>z<STATUS> <GN> = Gerätenummer, Standard = 0 <STATUS> = 0 - 3D-Interpolation aus = 1 - 3D-Interpolation ein
Anwendung	@0z1
Erläuterung	Die Prozessorkarte wird mit dem Dateneröffnungsteil @0 auf einen neuen Befehl vorbereitet. z1 ändert die Interpolation von 2-Achs- auf 3-Achs-Betrieb. Die Anweisung wirkt modal, d. h. alle move- und moveto-Anweisungen werden dreidimensional ausgeführt. Die Angabe von z2-Parametern in diesen Verfahrbewegungen wird ignoriert. Die Geschwindigkeitsangabe der Interpolation muss bei der X-Angabe erfolgen.



Programmierbeispiel

PAL-PC

```
#axis xyz;
reference xyz;
set3don;
move 10(700),15(800),3(400),
      0(30);
set3doff;
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
120 print#1,"@0r7":gosub1000
130 print#1,"@0z1":gosub 1000
140 print#1,"@0A100,700,150,800,30,400,0,30"
145 gosub 1000
150 print#@1,"@0z0":gosub1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Die Einleitung einer Referenzfahrt schaltet automatisch auf 2,5D-Interpolation zurück. Die korrekte Bearbeitung einer 3D-Interpolation setzt als Bezugsebene eine XY-Ebene voraus (vgl. Ebenenwahl).

Die maximale Geschwindigkeit für eine 3D-Interpolation beträgt 10 000 Schritte/s.

Die auf der Mechanik erreichbare Geschwindigkeit hängt von den angeschlossenen Motoren und Leistungsteilen ab.

Für Eilgang-Bewegungen sollten Sie kurzzeitig auf eine 2D-Interpolation umschalten und bei angehobener Z-Achse die Positionierung ausführen, da während einer 3D-Interpolation keine Kollisionsprüfung stattfindet.

Bei einer 3D-Interpolation ist nach einem Stopp-Befehl die aktuelle Position nicht korrekt. Die Positionsabfrage liefert erst nach Ende der Ausführung die korrekten Werte zurück.

2.2.2 Befehl: Zirkularinterpolation

Zweck Bearbeiten von Kreisen und Kreisbögen mit konstanter Bahngeschwindigkeit. Die Kreisinterpolation wird durch zwei aufeinanderfolgende Befehle ausgelöst. Der erste Befehl legt die Kreisrichtung fest, im zweiten werden die Interpolationsparameter übergeben.

Aufbau Kreisrichtung: @<GN>f-1 Gegen den Uhrzeigersinn
@<GN>f0 Uhrzeigersinn

Kreisbogen: @<GN>y B,V,D,Xs,Ys,Rx,Ry

B = Bogenlänge in Schritten
V = Geschwindigkeit (30 bis 10 000)
D = Interpolationsparameter
Xs = Startpunkt x
Ys = Startpunkt y
Rx = Richtung x
Ry = Richtung y

Berechnung der Parameter

Bogenlänge B Die Bogenlänge gibt die Länge des Bogens zwischen dem Start- und Endpunkt der Kreisinterpolation in Schritten an. Zur Berechnung des Parameters können Sie auf nachstehende Programmteile zurückgreifen. Hierbei gilt:

A - Anfangswinkel des Bogen- oder Kreissegmentes
 $A = \pi \cdot \text{Anfangswinkel} / 180$
E - Endpunkt der Bewegung
 $E = \pi \cdot \text{Endwinkel} / 180$
B - die resultierende Bogenlänge



Zur Berechnung der Bogenlänge dürfen nur Winkel im Bogenmaß verwendet werden.

1. Approximierende Formel (nur bei Viertel-, Halb- und Vollkreisen)

$$B = 4 \cdot \text{Radius} \cdot (E - A) / \pi$$

2. Berechnung der Bogenlänge mittels Software-Routine

```

if (Kreisrichtung=Gegenuhrzeigersinn) then
begin
  while(A<0) do A:=A+2.0*pi;
  while(E<0) do E:=E+2.0*pi; {Winkel in den positiven Bereich normieren}
  while (A>=pi/2.0) do
  begin
    A:=A-pi/2;
    E:=E-pi/2;
  end;
  B:=0.0;
  while (E-A>=pi/2.0) do
  begin
    E:=E-pi/2.0;
    B:=B+2.0*Radius;
  end;
  B:=B+Radius*(cos (A) -cos (E) +sin (E) -sin (A));
end;
else {Kreisrichtung = Uhrzeigersinn}
begin
  while (A>0) do A:=A-2.0*pi;
  while (E>0) do E:=E-2.0*pi; {Winkel in den negativen Bereich normieren}
  while (A<=-pi/2.0) do
  begin
    A:=A+pi/2;
    E:=E+pi/2;
  end;
  B:=0.0;
  while (A-E>=pi/2.0) do
  begin
    E:=E+pi/2.0;
    B:=B+2.0*Radius;
  end;
  B:=B+Radius*(cos (A) -cos (E) +sin (A) -sin (E));
end;
if (B<0) then B:= -B;

```



Die berechnete Bogenlänge muss auf den nächsten, ganzzahligen Wert gerundet übergeben werden. Hierbei sind Werte im Bereich von 3 bis 8 000 000 Schritten zulässig.

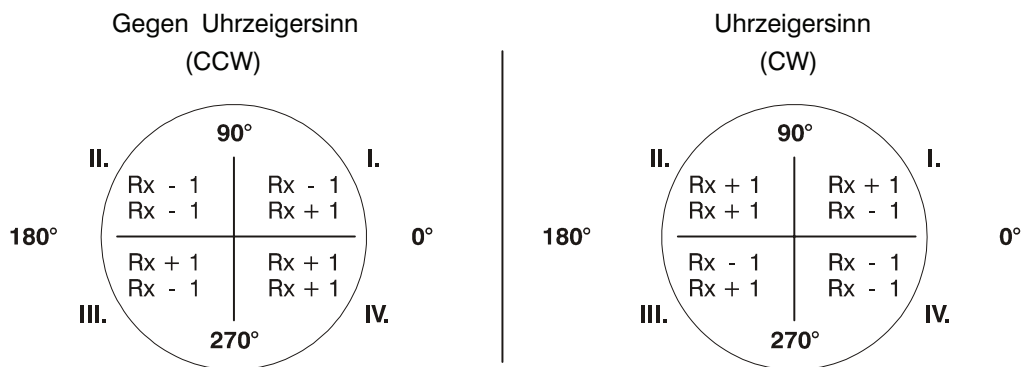
Geschwindigkeit V

Als Geschwindigkeitsangaben sind alle ganzzahligen Werte von 30 bis 10 000 Schritte/s zulässig.

Ob und mit welchen Geschwindigkeiten die Interpolation ausgeführt werden kann, hängt von den verwendeten Leistungsteilen und der angeschlossenen Mechanik ab.

Richtung Rx und Ry

Die Parameter Rx und Ry geben der Prozessorkarte an, in welchem Quadranten des Kreises die Interpolation startet.



Startpunkt Xs und Ys

Diese Parameter geben die Startpunkte Xs und Ys relativ auf den Kreismittelpunkt an.

Zur Berechnung gelten folgende Formeln:

$$X_s = \text{Radius} * \cos(A)$$

$$Y_s = \text{Radius} * \sin(A)$$

Interpolationsparameter D

D muss übergeben werden, da die Prozessorkarte aufgrund ihrer Speicherkapazität nicht in der Lage ist, diesen Parameter zu berechnen.

Zur Berechnung können Sie folgenden Programmausschnitt heranziehen.

```
function Summe(xx:real):real;
begin
  if(xx>0) then
    Summe:=xx*(xx+1)
  else
    Summe:= -xx*(xx-1)
  end;

function formel:real;
begin
  if (Kreisrichtung=Gegenuhrzeigersinn) then
    Formel:= (Rx*Ry*Radius+Rx*Ry*Summe(Radius-1.0)
              -Rx*Summe(Xs+(Rx-Ry)/2.0)+Ry*Summe(Ys+(Rx+Ry)/2.0))/2;
  else
    {Richtung = Uhrzeigersinn}
    Formel:= (-Rx*Ry*Radius-Rx*Ry*Summe(Radius-1.0)
              - Rx*Summe(Xs+(Rx+Ry)/2.0) + Ry*Summe(Ys+(Ry-Rx)/2))/2;
  end;
D:=Formel;
```



Der berechnete Parameter muss als gerundet und als ganzzahliger Wert übergeben werden.



Programmierbeispiel zur Parameter-Berechnung

Es soll ein Viertelkreis (90°) entgegen dem Uhrzeigersinn mit einem Radius von 200 Schritten und einer Verfahrgeschwindigkeit von 1 500 Schritten/s ausgeführt werden. Als Startwinkel ist 135° festgelegt.

1. Winkel im Bogenmaß:

$$\begin{aligned} A &= \pi \cdot 135 / 180 \\ &= 3 \cdot \pi / 4 \\ B &= \pi \cdot (135 + 90) / 180 \\ &= 5 \cdot \pi / 4 \end{aligned}$$
2. Bogenlänge: (nach Näherungsformel)

$$\begin{aligned} B &= 4 \cdot \text{Radius} \cdot (E - A) / 180 \\ &= 2 \cdot 200 \cdot (5 \cdot \pi / 4 - 3 \cdot \pi / 4) / \pi \\ &= 400 \end{aligned}$$
3. Richtung Rx, Ry:

Anfangswinkel = 135°, Gegenuhrzeigersinn

$$\begin{aligned} R_x &= -1 \\ R_y &= -1 \end{aligned}$$
4. Startpunkt Xs, Ys:

$$\begin{aligned} X_s &= \text{Radius} \cdot \cos(\text{Anfangswinkel}) \\ &= 200 \cdot \cos(135) \\ &= -141 \\ Y_s &= \text{Radius} \cdot \sin(\text{Anfangswinkel}) \\ &= 200 \cdot \sin(135) \\ &= 141 \end{aligned}$$

5. Interpolationsparameter D: (Gegenuhrzeigersinn)

$$D = (R_x * R_y * \text{Radius} + R_x * R_y * \text{Summe}(\text{Radius} - 1.0) - R_x * \text{Summe}(X_s + (R_x - R_y) / 2.0) + R_y * \text{Summe}(Y_s + (R_x + R_y) / 2.0)) / 2$$

$$\begin{aligned} \text{Summe}(\text{Radius}-1) &= \text{Summe } (200-1) \\ &= \text{Summe } (199) \\ &= 199 * (199+1) \\ &= 39\,800 \end{aligned}$$

$$\begin{aligned} \text{Summe}(X_s + (R_x - R_y) / 2.0) &= \text{Summe } (-141 + (-1 - (-1)) / 2.0) \\ &= \text{Summe } (-141) \\ &= -(-141) * ((-141) - 1) \\ &= -20\,022 \end{aligned}$$

$$\begin{aligned} \text{Summe } (Y_s + (R_x + R_y) / 2.0) &= \text{Summe } (141 + (-1 + (-1)) / 2.0) \\ &= \text{Summe } (141-1) \\ &= 140 * (140+1) \\ &= 19\,740 \end{aligned}$$

$$\begin{aligned} D &= (R_x * R_y * \text{Radius} + R_x * R_y * 39800 - R_x * (-20022) + R_y * 19740) / 2 \\ &= ((-1) * (-1) * 200 + (-1) * (-1) * 39800 - (-1) * (-20022) + (-1) * 19740) / 2 \\ &= (200 + 39800 - 20022 - 19740) / 2 \\ &= 119 \end{aligned}$$

Der Programmabschnitt muss also folgendermaßen aussehen:

```
...
...
@Of-1
@Oy400,1500,119,-141,141,-1,-1
...
...
```

bzw. im Direktformat:

```
...
...
f-1
y400,1500,119,-141,141,-1,-1
...
...
9
@0s
```

2.3 Ergänzender Befehlsumfang bei Interface-Karten mit E/A Erweiterung

2.3.1 Befehl: Externe Speicherung

Zweck	Speichern eines CNC-Programmes auf ein externes Speichermedium.
Aufbau	@<GN>u <GN> = Gerätenummer, Standard = 0
Erläuterung	<p>Dieser Befehl ermöglicht, ein im Datenspeicher der Prozessorkarte befindliches CNC-Programm auf einem Datenspeicher im Scheckkartenformat (Memory-Card) abzuspeichern oder von einer Speicherkarte zurück in die Prozessorkarte zu lesen.</p> <p>Dabei ist folgender Ablauf einzuhalten:</p> <ol style="list-style-type: none">1. Übertragen von Prozessorkarte zur Memory-Card<ol style="list-style-type: none">a) Programm wie gewohnt in Prozessorkarte übertragenb) Memory-Card einsteckenc) Befehl @0u übergebend) Memory-Card entnehmen2. Übertragen von Memory-Card nach Prozessorkarte<ol style="list-style-type: none">a) Controller einschaltenb) Memory-Card einsteckenc) μP-Reset drückend) Memory-Card entnehmen



Die Memory-Card sollte beim Einschalten der Steuerung nicht gesteckt sein. Bei den Memory-Card-Typen 8 k x 8 und 16 k x 8 wird nicht die vorhandene Speicherkapazität geprüft, d. h. dass bei komplexen Programmen die Speichergrenze ohne Fehlermeldung überschritten wird.

2.3.2 Befehl: Ausgangsport setzen

Zweck	Die Prozessorkarte setzt am definierten Ausgangsport der E/A-Erweiterungseinheit ein gewünschtes Ausgangsmuster.
Aufbau	<p>@<GN>B<ADRESSE>, <WERT></p> <p><GN> = Gerätenummer, Standard = 0 <ADRESSE> = Ausgangsport 1 —> 65 529 = Ausgangsport 2 —> 65 530 <WERT> = 0 ... 255</p>
Erläuterung	<p>Dieser Befehl entspricht im wesentlichen dem Poke-Befehl des Standard-Betriebssystems 5.x. Im CNC-Betrieb (Speicher-Modus) wird zusätzlich zur byteweisen Bearbeitung eine bitweise Bearbeitung der Ausgangsports möglich.</p> <p>So können Sie gezielt einzelne Bit setzen oder löschen.</p>

2.3.3 Befehl: Eingangsport lesen

Zweck	Die Prozessorkarte liest das anstehende Bitmuster am Eingangsport der E/A-Erweiterung ein.
Aufbau	<p>@<GN>b<ADRESSE></p> <p><GN> = Gerätenummer, Standard = 0 <ADRESSE> = Eingangsport 65 531</p>
Erläuterung	<p>Der Befehlsaufbau ist identisch mit dem <i>Peek</i>-Befehl des Standard-Betriebssystems 5.x. Im Speicher-Modus (CNC-Betrieb) besteht darüberhinaus die Möglichkeit, in Abhängigkeit eines Bitmusters Verzweigungen nach Vorwärts und Rückwärts zu realisieren.</p>

2.4 Ergänzender Befehlsumfang EP1090

2.4.1 Befehl: Output-Modul

Zweck	Hauptspindel zu- oder abschalten Die Interface-Karte schaltet die Hauptspindel ein bzw. aus.		
Aufbau	@<GN>h<Parameter>	(DNC)	
	h<Parameter>	(speicherbar)	
	<GN>	= Gerätenummer, Standard =0	
	<Parameter>	= 1 Hauptspindel einschalten	
		= 0 Hauptspindel abschalten	
Anwendung	@0h1		
Erläuterung	Die Karte wird durch @0 adressiert. h gibt an, dass das Output-Modul geschaltet werden soll. Der Parameter 1 bewirkt das Einschalten des integrierten Relais. Sollte sich die Spindel bereits im Ein-Zustand befinden, hat der Befehl keine Auswirkungen. Der Zustand der Hauptspindel wird durch den Dezimalpunkt der 7-Segment-Anzeige angezeigt. Der Rechner meldet sich nach erfolgter Ausführung mit der Rückmeldung 0.		

2.5 Ergänzender Befehlsumfang ab Interface-Karte Version AZ1350/5

2.5.1 Befehl: Magnetbremse

Zweck	Der Befehl ermöglicht eine softwaremäßige Aktivierung bzw. Deaktivierung eines speziellen Ausgangs, der über Relais verstärkt, Magnetbremsen in Antriebseinheiten steuert.		
Aufbau	@<GN> g <Status>		
	<GN>	= Gerätenummer	
	<Status>	= 1 Bremse magnetisiert	
		= 0 Bremse inaktiv	
Anwendung	@0g1		
Erläuterung	Die Karte wird durch @0 adressiert, g gibt an, dass das Brems-Relais geschaltet werden soll. Status 1 bewirkt das Einschalten des Relais. Dadurch wird eine angeschlossene Magnet-Bremse aktiviert und gibt die Antriebsachse des Motors frei. Bei Status 0 wird die Bremse nicht bestromt, d. h. dass die Antriebsachse des Motors gebremst wird. Der Rechner meldet sich nach erfolgter Ausführung mit der Rückmeldung 0.		

2.6 Kontroll- und Steuercodes

Kontroll- und Steuercodes ermöglichen über die serielle Schnittstelle den direkten Eingriff in den Funktionsablauf der Interface-Karte. Dabei werden die jeweils gesendeten Kommandos ohne Verzögerung ausgeführt.

2.6.1 Befehl: Selbsttest

Zweck	Die Prozessorkarte überprüft die Funktionsbereitschaft ihrer Funktionsbaugruppen.
Aufbau	chr(252)
Erläuterung	Die Interface-Karte überprüft die Kapazität ihres Datenspeichers, die Checksumme ihres Betriebs-Eproms sowie die Schalterstellung des DIP-Schalters. Anschließend werden zum Test der angeschlossenen Schrittmotoren einige Taktimpulse an die Leistungselektronik der X- und Y-Achse ausgegeben. Abgeschlossen wird die Test-Routine durch die permanente Ausgabe eines ASCII-Zeichensatz über die serielle Schnittstelle.

Beschränkung -



Programmierbeispiel

PAL-PC

GW-BASIC

(Terminalmode)

chr(252)

-



Den Selbsttest können Sie nur beenden, wenn Sie die Versorgungsspannung ausschalten oder einen μ P-Reset durchführen.

2.6.2 Befehl: STOPP

Zweck	Die Prozessorkarte bricht eine momentane Verfahrbewegung ab.
Aufbau	chr(253)
Erläuterung	Eine Positionierbewegung im DNC-Modus (relativ oder absolut) kann durch einen STOPP-Befehl unterbrochen werden, ohne dass Schrittverluste auftreten. Ein danach ausgeführter START-Impuls beendet den unterbrochenen Funktionsablauf. Außerdem können Sie nach einem STOPP-Befehl mit Hilfe des Befehles <i>Positionsabfrage</i> die aktuelle erreichte Position rücklesen.
Beschränkung	Der Befehl können Sie nur verwenden, wenn eine Positionierbewegung stattfindet.



Programmierbeispiel

PAL-PC

(Terminalmode)

chr(253)

GW-BASIC

-



Die Prozessorkarte meldet als Quittierung den Stopp-Fehler zurück.

Da der Befehl ohne Adressierung arbeitet, werden die Verfahrbewegungen aller angeschlossenen Prozessorkarten unterbrochen.

Der übergeordnete Rechner muss die letzte anzufahrende Position im DNC-Modus erneut senden.

2.6.3 Befehl: μ P-Reset

Zweck	Die Prozessorkarte unterbricht sofort alle Aktivitäten und geht in den Reset-Zustand
Aufbau	chr(254)
Erläuterung	Durch einen μ P-Reset wird der integrierte Mikro-Controller abrupt in seinen Einschaltzustand zurückgeschaltet. Die Ausgänge liegen während des Reset-Zustandes auf Vcc-Potential und schalten nach Aufheben auf GND-Potential.
Beschränkung	-



Programmierbeispiel

PAL-PC
(Terminalmode)

GW-BASIC

chr(254)

-

2.6.4 Befehl: Break

Zweck	Der Break-Befehl dient zum Abbruch des aktuellen Positioniervorganges
Aufbau	chr(255)
Erläuterung	Durch Senden eines Break-Befehles an der seriellen Schnittstelle wird der aktuelle Positioniervorgang der Interface-Karte abgebrochen, ohne dass eine Stopp-Rampe eingeleitet wurde. Dadurch auftretende Schritt-fehler werden ignoriert. Im Gegensatz zum ähnlich wirkenden μ P-Reset, kann nach dem <i>Break</i> -Befehl normal weitergearbeitet werden ohne eine Neuinitialisierung der Prozessorkarte vorzunehmen.
Beschränkung	-



Programmierbeispiel

PAL-PC
(Terminalmode)

GW-BASIC

chr(255)

-

3 Befehlsaufbau CNC-Befehle

Im CNC-Modus speichert die Prozessorkarte alle übersendeten Befehle im internen Datenspeicher. Zur Aktivierung ist nach der Standard-Initialisierung @<Gerätenummer> <Achszahl> der Befehl @<Gerätenummer>i zu übertragen. Anschließend wird das Datenfeld übergeben und mit dem Befehl 9 als Datenfeldende-Zeichen abgeschlossen. Jetzt können Sie das Programm erneut durch einen externen Start-Befehl aktivieren. Als Start können Sie sowohl eine Start-Taste (z. B. in der Frontplatte der Prozessorkarte) als auch den Befehl @<Geräte- nummer>S verwenden.

Durch den physikalisch begrenzten RAM-Speicher der Prozessorkarten ist die Anzahl der speicherbaren Befehle auf ca. 1 200 im 3-Achs-Betrieb, ca. 1 800 im 2-Achs-Betrieb und ca. 2 400 im 1-Achs-Betrieb eingeschränkt.

Um bei Ausfall der Versorgungsspannung (z. B. Abschalten der Versorgungsspannung) keinen Datenverlust des RAM-Speichers zu erleiden, können Sie durch einen optionalen Akku bzw. einer Primärzelle ein sogenanntes Memory-back-up erzeugen.

Im folgenden werden die speicherbaren Befehle der Prozessorkarte aufgelistet und kurz erläutert. Eine Detaillierung kann unter dem entsprechendem Befehl des Direkt-Modus nachgeschlagen werden.

3.1 Grundbefehlssatz ab Prozessorkarte 4.0

3.1.1 Befehl: INPUT

Zweck	Die Prozessorkarte wird in den Speicher-Modus gesetzt. Alle nachfolgenden Befehle werden im internen Datenspeicher abgelegt. Die Ausführung der gespeicherten Befehle wird mit dem Befehl Start oder der Start-Taste veranlasst.
Aufbau	@<GN>i <GN> = Gerätenummer, Standard = 0
Anwendung	@0i
Erläuterung	Die Karte wird durch @0 initialisiert. i gibt an, dass zu speichernde Befehle folgen. Nach Empfang des Befehles erwartet die Prozessorkarte ein komplettes NC-Programm - bestehend aus speicherbaren Befehlen. Dieses muss mit dem Datenfeld-Ende (9) abgeschlossen sein. Das Datenfeld kann alle speicherbaren Befehle enthalten.
Beschränkung	Den Befehl können Sie erst verwenden, nachdem die Achsenanzahl gesetzt, d. h. die Prozessorkarte initialisiert ist.



Programmierbeispiel

PAL-PC

```
#axis xy;
move 50(100),40(100);
stop.
#start
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 50,100,40,100":gosub 1000
140 print#1,"9":gosub 1000
150 print#1,"@0S":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler:";a$
1030 stop
```



Der *Input*-Befehl löscht die bisher gespeicherten Anweisungen. Anschließend erwartet die Prozessorkarte ein komplettes Datenfeld - abgeschlossen mit dem Datenfeld-Ende-Kennzeichen (9).

Tritt während der Übertragung der zu speichernden Befehle ein Fehler auf, verlässt die Prozessorkarte den Input-Modus, das bis dahin übergebene NC-Programm geht verloren.

3.1.2 Befehl: Referenzfahrt

Zweck	Die Prozessorkarte verfährt alle angegebenen Achsen an ihre Nullpunkte (Referenzpunkte)								
Aufbau	7<Achsen> <Achsen> = ganzzahliger Wert zwischen 1 und 7								
Erläuterung	7 gibt an, dass eine Referenzfahrt ausgeführt werden soll. Der anschließende Zahlenwert definiert die Achsen, die eine Referenzfahrt durchführen sollen.								
	<table> <tr> <td>x = 1</td><td>xz = 5</td></tr> <tr> <td>y = 2</td><td>yz = 6</td></tr> <tr> <td>xy = 3</td><td>xyz = 7</td></tr> <tr> <td>z = 4</td><td></td></tr> </table>	x = 1	xz = 5	y = 2	yz = 6	xy = 3	xyz = 7	z = 4	
x = 1	xz = 5								
y = 2	yz = 6								
xy = 3	xyz = 7								
z = 4									

Die Reihenfolge der Ausführung ist dabei wie folgt festgelegt:

—> Z-Achse —> Y-Achse —> X-Achse

Dies gilt auch, wenn mit dem Ebenenbefehl eine andere Achse als Werkzeugachse bestimmt worden ist. Hier kann gegebenenfalls die getrennte Referenzfahrt der Achsen Kollisionen mit dem Werkstück verhindern.



Programmierbeispiel

PAL-PC

```
#axis xyz;
reference xyz;
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000stop.
120 print#1,"@0i":gosub 1000
130 print#1,"77":gosub 1000
140 print#1,"9":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler:";a$
1030 stop
```



(siehe Befehl *Referenzfahrt* Kapitel 2.1.2)

3.1.3 Befehl: Bewegung relativ

Zweck	Die Prozessorkarte gibt entsprechend der übergebenen Schrittzahl und Schrittgeschwindigkeit für jede Leistungsstufe eine Impulskette aus. Die Verfahrbewegung wird sofort ausgeführt oder gespeichert.
Aufbau	<p>0 <Sx>,<Gx>,<Sy>,<Gy>,<Sz1>,<Gz1>,<Sz2>,<Gz2></p> <p>0 = Bewegung relativ</p> <p><Sx> = Schrittzahl x, Wert zwischen 0 und $\pm 8\,388\,607$</p> <p><Gx> = Geschwindigkeit x, Wert zwischen 30 und 10 000</p> <p>.</p> <p>.</p> <p><Gz2> = Geschwindigkeit Z-Achse (2. Bewegung)</p>
Erläuterung	0 gibt an, dass eine relative Bewegung erfolgen soll. Die Prozessorkarte erwartet nun für jede Achse ein Zahlenpaar bestehend aus Schrittzahl und Geschwindigkeit.



Programmierbeispiel

PAL-PC

```
#axis xyz;
move 50(500),300(900),
      20(200),-20(900);
move 20(300),300(3000),
      0(21),0(21);
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 35,800,250,2000":gosub 1000
140 print#1,"0 20,2000,-25,1000":gosub 1000
150 print#1,"9":gosub 1000
160 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```

(siehe Befehl *Bewegung relativ* Kapitel 2.1.4)



3.1.4 Befehl: Bewegung zur Position (move to)

Zweck	Die Prozessorkarte fährt mit den angegebenen Geschwindigkeiten an die angegebene Position. Die Verfahrbewegung wird sofort ausgeführt.
Aufbau	m <Sx>, <Gx>, <Sy>, <Gy>, <Sz1>, <Gz1>, <Sz2>, <Gz2>
Erläuterung	<p>m gibt an, dass eine Absolut-Position folgt.</p> <p>Aus Kompatibilitätsgründen zum relativen Positionierbefehl werden auch hier für die Z-Achse zwei Zahlenpaare erwartet. Die zweite Positionsangabe der Z-Position muss Null sein. Diese Zahl wird ignoriert, muss aber vorhanden sein.</p>

Programmierbeispiel



PAL-PC

```
#axis xy;
moveto 50(500),300(900);
moveto 50(500),300(900);
moveto 20(200),30(900);
stop.
```

GW-BASIC

```
100 open"com1.9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"m 500,800,200,31":gosub 1000
140 print#1,"m31,500,40,500":gosub 1000
150 print#1,"9":gosub 1000
160 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```

(siehe Befehl *Bewegung zur Position* Kapitel 2.1.5)



3.1.5 Befehl: Nullpunktverschiebung

Zweck	Die Prozessorkarte speichert die momentane Position als virtuellen Nullpunktes für die angegebene(n) Achse(n). Die nächsten <i>Verfahre absolut</i> -Anweisungen berücksichtigen diesen virtuellen Nullpunkt als neuen Bezugspunkt.
Aufbau	n<Achsen> <Achsen> = ganzzahliger Wert zwischen 1 und 7
Erläuterung	n gibt an, dass eine Nullpunktverschiebung vorgenommen werden soll. Nach dem Befehl werden müssen Sie dem Rechner die Achsen mitteilen, für die eine Nullpunktverschiebung durchgeführt werden soll. Hierbei gilt die Zuordnung x = 1, y = 2, z = 4. Soll für mehrere Achsen eine Nullpunktverschiebung durchgeführt werden, müssen obige Werte addiert werden.

Programmierbeispiel



PAL-PC

```
#axis xy:
move 350(800),200(800);
null xy;
move 20(500),30(300);
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 8000,900,800,900":gosub 1000
140 print#1,"n 3":gosub 1000
150 print#1,"m 200,900,400,990":gosub 1000
160 print#1,"9":gosub 1000
170 print#1,"@0S":gosub 1000
180 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```

(siehe Befehl *Nullpunktverschiebung* Kapitel 2.1.7)



Befehl: Ebenenwahl

Zweck	2,5-D interpolierende Prozessorkarten (z. B. Interface-Karte 4.0) können nur zwei von drei Achsen interpolieren. Im Einschaltzustand sind dies die X- und Y-Achse. Durch den <i>Ebenenwahl</i> -Befehl haben Sie die Möglichkeit, jede andere Ebenenkonfiguration als Hauptebene zu definieren. Die übrigbleibende dritte Achse wird als Werkzeugachse aufgefasst und anschließend an die Positionierung der Hauptachsen verfahren.	
Aufbau	e<Ebene>	
	<Ebene>	= Zahl zwischen 0 und 2
	0	= xy
	1	= xz
	2	= yz
Erläuterung	(siehe Kapitel 2.1.8)	

Programmierbeispiel

PAL-PC



```
#axis xyz;
line yz;
move 20(200),33(500),
      40(1000),0(21);
stop.
```

GW-BASIC

```
100 open"com1.9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"e2":gosub 1000
140 print#1,"m20,200,30,900,33,900, 0,21"
150 gosub 1000
160 print#1,"9":gosub 1000
170 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```

(siehe Befehl *Ebenenwahl* Kapitel 2.1.8)

3.1.7 Befehl: Synchronisationszeichen senden



Zweck Die Prozessorkarte teilt einer zweiten Prozessorkarte oder einem

übergeordneten Rechner mit, dass ein bestimmter Punkt im Ablaufschema (NC-Programm) erreicht ist. Der Befehl dient dazu, die Prozessorkarte mit einer externen Einheit zu synchronisieren bzw. eine externe Einheit aufzufordern, eine Tätigkeit vorzunehmen.

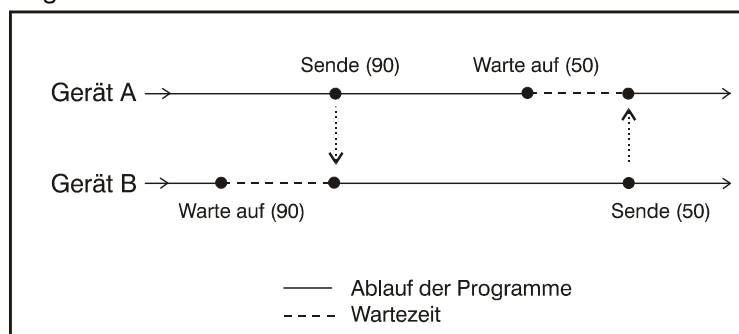
Aufbau

1 <SyncChar>

<SyncChar> = Synchronisationszeichen zwischen 33 und 125

Erläuterung

Die Prozessorkarte sendet an die serielle Schnittstelle ein definiertes ASCII-Zeichen. Die Empfangsstation wartet - bedingt durch den Befehl *Warte auf Synchronisationszeichen* - auf das entsprechende Zeichen und fährt nach Erhalt mit dem programmierten CNC-Ablauf fort. Nachfolgendes Schema zeigt einen kurzen Überblick über den Funktionsablauf.


Beschränkung

Durch die Befehle *Synchronisationszeichen senden* und *Warte auf Synchronisationszeichen* können ohne übergeordneten Rechner nur zwei Prozessorkarten synchronisiert werden. Die übergebene Nummer des Synchronisationszeichens muss ein druckbares Zeichen im Bereich von 33 - 125 sein, da andere Zeichen von der Prozessorkarte gefiltert werden. Das Zeichen 64 sollte nicht verwendet werden, da dieses Zeichen den Datenverkehr wartender Prozessorkarten eröffnet. Die seriellen Schnittstellen der Geräte sind über das Verbindungskabel Interface - Interface zu verbinden.

Programmierbeispiel
PAL-PC

```
#axis xyz;
#input
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
```



send 90;

.
.
.

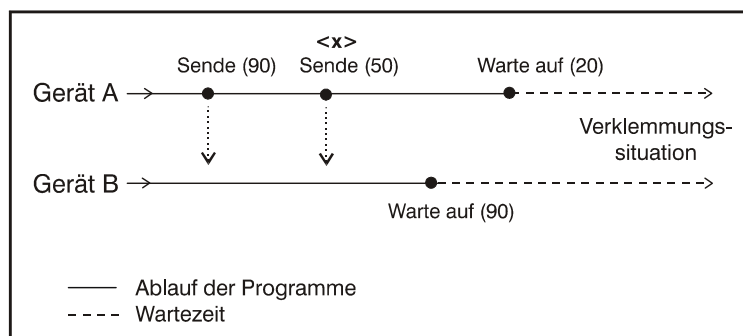
```

120 print#1,"@0i":gosub 1000
130 print#1,"1 90":gosub 1000
140 print#1,"9":gosub 1000
150 print#1,"@0s":gosub 1000
160 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop

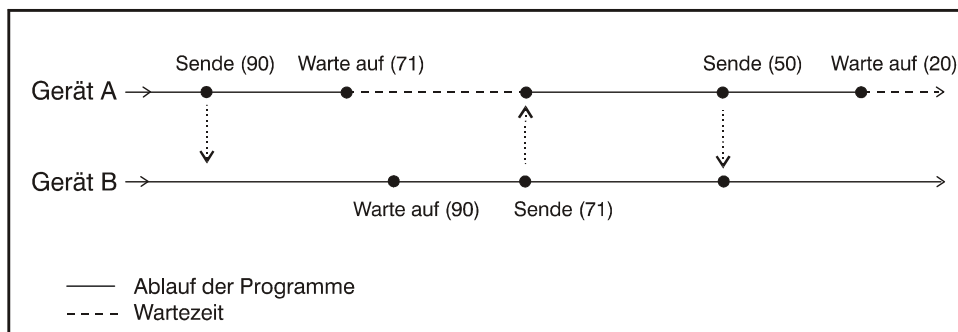
```

Zur Prüfung des PAL-PC-Programmes können Sie die Funktion *Kommunikation* verwenden, zur Prüfung des BASIC-Programmes können Sie das bei dem Schnittstellentest angegebene Programm benutzen.

Die Interface-Karte kann während des Ablaufes einer gespeicherten Befehlsfolge nur ein Zeichen empfangen und zwischenspeichern. Folgende Situation führt daher zwangsweise zu einer Verklemmung des Gesamtsystemes:



Bei <x> steht im Eingangspuffer des Gerätes B ein Zeichen char(50) (vorher gesendetes Zeichen wurde überschrieben). Der Prozess wird somit "ewig" auf das notwendige Zeichen char(90) warten. Aus diesem Grund sollte das sendende Gerät auf eine Bestätigung des empfangenden Gerätes warten, bevor ein erneutes Synchronisationszeichen gesendet wird.



3.1.8 Befehl: Warte auf Synchronisationszeichen

Zweck

Die Prozessorkarte wartet auf den Empfang des angegebenen

Zeichens an der seriellen Schnittstelle. In Verbindung mit einem übergeordneten Rechner kann der Befehl für Verzweigungen im gespeicherten Ablauf genutzt werden.

Aufbau	<p>2 <SyncChar>, <Offset></p> <p><SyncChar> = Synchronisationszeichen von 33 bis 125</p> <p><Offset> = Verzweigung bei Empfang <SyncChar> + 1 Zahl zwischen - 32 767 und + 32 767</p>
Anwendung	<p>2 50,0 Warte auf Synchronisationszeichen</p> <p>502 55,7 Warte auf 55, verzweige 7 Befehle vorwärts bei Empfang von 56</p>
Erläuterung	<p>Verwendung des Befehles vgl. "Sende Synchronisationszeichen". In Verbindung mit einem übergeordneten Rechner können Sie diesen Befehl auch für logische Entscheidungen im Prozessablauf nutzen:</p>

Programmschritt

- 1
- 2
- 3 wait 50,-1 <— ext. Rechner sendet 50 oder 51
- 4
- 5

Im Ablauf des Datenfeldes wird bei Befehl 3 gestoppt. Sendet der übergeordnete Rechner ein char(50), wird als nächster Befehl Nr. 4 ausgeführt; sendet der Rechner ein char(51), wird als nächster Befehl der Befehl Nr. 2 ausgeführt.

Allgemein: Wird das auf das zu wartende Zeichen folgende Zeichen von der Prozessorkarte empfangen, findet die angegebene Verzweigung statt, anderenfalls wird der dem Wartebefehl folgende Befehl ausgeführt.

Programmierbeispiel

PAL-PC

GW-BASIC



```

#axis x;
input
label: move 3(1000);
wait 50,label;
stop.
#start

100 open"com1:9600,N,8,1,DS,CD"as #1#
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 500,5000":gosub 1000
140 print#1,"2 50,-1":gosub 1000
150 print#1,"9":gosub 1000
160 print#1,"@0s":gosub 1000
170 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
    
```

Dies Programm wird nach der Übertragung zur Prozessorkarte zunächst mit @0s gestartet. Im Programmablauf folgt eine relative Bewegung der X-Achse. Anschließend wartet die Prozessorkarte an der seriellen Schnittstelle auf den Empfang von char(50) oder char(51). Bei einem gesendeten char(51), verzweigt die Karte zurück, es wird dann erneut die relative Bewegung ausgeführt. Bei Empfang des char(50) wird das Programm beendet.

Bitte beachten Sie, dass eine Verzweigung vor oder hinter das Ende des Datenfeldes zu nicht vorhersehbaren Ergebnissen führen kann.

3.1.9 Befehl: Schleife / Verzweigung

Zweck Programmschleifen dienen dazu, gleichartige Bewegungsabläufe zusammenzufassen. Hierdurch wird der zur Verfügung stehende Speicherplatz der Prozessorkarte besser genutzt. Durch Verzweigungen kann nach einer logischen Entscheidung an einen bestimmten Punkt des Prozesses zurückverzweigt werden.

Aufbau

```

3 <Anzahl>,<Offset>
  <Anzahl>          =   Schleifen:      0 < Zahl < 32 767
                       Verzweigung:    0
    
```


<p><Offset> =</p> <p>Anwendung: 3 25,-1</p> <p> 3 0,-5</p> <p> 3 0,5</p> <p> 3 6,-5</p>	<p>Anzahl der zu wiederholenden Befehle bzw. Verzweigungsziel relativ Schleifen: - 1 > Zahl > - 3 000</p> <p>Verzweigung: - 3 000 < Zahl < 3 000</p> <p>Wiederhole letzten Befehl 25 mal</p> <p>Verzweige immer 5 Schritte zurück</p> <p>Überspringe die nächsten 4 Befehle</p> <p>Wiederhole die letzten 5 Befehle 6 mal</p>
--	--

Erläuterung Trifft die Prozessorkarte innerhalb des CNC-Programmablaufes auf den Befehl 3, wird ein Schleifenzähler eingerichtet, vorbesetzt und der Befehlszähler um den angegebenen Offset korrigiert. Die Befehle bis zum Schleifenzähler werden so oft wiederholt, bis der Schleifenzähler Null erreicht. Anschließend wird mit der Ausführung des ersten Befehles nach der Schleife fortgefahren. Falls die Angabe der Schleifenanzahl 0 ist, wird eine erzwungene Verzweigung ausgelöst.

Programmierbeispiel

PAL-PC



```
#axis x;
#input
repeat
  repeat
    move 2(1000);
  until 5;
  move -10(2000);
until 10;
stop.
#start
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 200,2000":gosub 1000
140 print#1,"3 5,-1":gosub 1000
150 print#1,"0 -1000,1000":gosub 1000
160 print#1,"3 10,-3":gosub 1000
170 print#1,"9":gosub 1000
180 print#1,"@0S":gosub 1000
190 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```

Es darf nicht vor dem Anfang des Datenfeldes verzweigt werden.

Vorwärtsschleifen (3 10,10) sind unzulässig.

Eine Schleife wiederholt immer die letzten n-Befehle.



Es muss mindestens ein Befehl wiederholt werden; 3 10,0 ist unzulässig.

Schleifen dürfen geschachtelt sein, die maximale Schachtelungstiefe beträgt 15.

Eine Schleife darf nicht durch eine Verzweigung vorwärts verlassen werden.

3.1.10 Befehl: Impulssteuerung

Zweck	Die Hardware-Option <i>Impulsausgabe</i> erweitert die Signalaus- und -eingänge an der Prozessorkarte um einen gesonderten Port. Sie können ihn sowohl als Eingang als auch als Ausgang verwenden.
Aufbau	<p>4 <Option> <Option> = ganze Zahl zwischen 1 und 6.</p> <p>1 = Ausgang auf EIN setzen 2 = Ausgang auf AUS setzen 3 = Impuls für 0,5 s 4 = Auf Impuls warten 5 = Impuls ausgeben und auf Quittierung warten Wiederhole nach 0,5 s 6 = Auf Impuls warten und Quittierung ausgeben</p>
Anwendung	<p>4 1 4 5</p>
Erläuterung	Die Impulsausgabe dient dazu, externe Geräte mit der Prozessorkarte zu verbinden. Die einzelnen Optionen (1 ... 6) bewirken eine reibungslose Steuerung mit geringem externen Hardware-Aufwand. Die Impulsausgabe-Option stellt einen über ein Reed-Relais gepufferten potentialfreien Ausgang zur Verfügung.

Programmierbeispiel



PAL-PC

```
#axis x;
#input
pulse wait;
move 2(9000)
.
.
.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"4 4":gosub 1000
140 print#1,"0 200,9000":gosub 1000
150 print#1,"9":gosub 1000
160 print#1,"@0S":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Beachten Sie, dass der Impulsausgang vor dem Ende des Programmes zurückgesetzt werden muss, da sonst die Prozessorkarte einen Startbefehl am Startastereingang erkennt und das gespeicherte Programm sofort erneut ausführt. Sollte dies geschehen, muss die Prozessorkarte mit dem Not-Aus-Taster gestoppt oder ausgeschaltet werden. Der Stopp-Taster wird ignoriert solange der Impulsausgang gesetzt ist.

3.1.11 Befehl: Zeitverzögerung

Zweck	Die Prozessorkarte wartet die angegebene Zeit, bevor der nächste Programmschritt ausgeführt wird.	
Aufbau	5 <time> <time> - Zahl zwischen 0 ... 32 767 (Angabe in 1/10 s)	
Anwendung	5 40	(warte 4 s)
Erläuterung	-	



Programmierbeispiel

PAL-PC

```
#axis x;
#input
move 2(1000);
wait 100;
move -2(9000);
stop.
#start
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 200,1000":gosub 1000
140 print#1,"5 100":gosub 1000
150 print#1,"0 -200,9000":gosub 1000
160 print#1,"9":gosub 1000
170 print#1,"@0S":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ",a$
1030 stop
```



Eine Zeitverzögerung können Sie nicht durch Betätigen des Stopp-Tasters abbrechen.
Bei fehlerhafter Programmierung bricht die μ P-Taste den Prozess ab.

3.1.12 Befehl: Bewegung bis Impuls

Zweck	Dieser Befehl ermöglicht, die relative Positionierung ohne eine exakte Weginformation vorzugeben (siehe 1. Hinweis unten). Dabei wird der Prozessorkarte ein maximaler Verfahrensweg vorgegeben, der durch einen externen Stopp-Impuls abgebrochen wird (z. B. Betätigen der Stopp-Taste). Anschließend wird der nächste Befehl des Datenfeldes abgearbeitet (siehe 2. Hinweis unten).
Aufbau	6 <Sx>, <Gx>, <Sy>, <Gy>, <Sz1>, <Gx1>, <Sz2>, <Gz2> <Sx> = Schrittzahl X-Achse . <Gz2> = Geschwindigkeit 2. Z-Achse
Erläuterung	Der Befehlsaufbau ist identisch mit dem Befehl <i>Bewegung relativ</i> (siehe 3.1.3 und 2.1.4)



Programmierbeispiel

PAL-PC

```
#axis x;
#input
movep 20(1000);
movep -20(1000);
stop.
#start
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"6 200,1000":gosub 1000
140 print#1,"6 -200,9000":gosub 1000
150 print#1,"9":gosub 1000
160 print#1,"@0S":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Die Prozessorkarte prüft nicht, ob die Bewegung den zulässigen Verfahrensbereich der angeschlossenen Mechanik verlässt (Endschalter könnten aktiviert werden).



Die Impulslänge des externen Tasters darf maximal 40 µs betragen. Falls diese Impulslänge nicht eingehalten werden kann, muss im Programmablauf als nachfolgender Befehl eine Zeitverzögerung programmiert werden, da sonst der anschließende Befehl übergangen wird.

3.1.13 Befehl: Starte angeschlossene Interface-Karte

Zweck	Dieser Befehl ergänzt die durch Prozesssynchronisation mögliche gegenseitige Aktivierung von zwei Prozessorkarten um vier Optionen.
Aufbau	<p>8 <GN> <Optionen> [<Achsen>]</p> <p><GN> = Gerätenummer der anzusprechenden Karte</p> <p><Option> = R Referenzfahrt auslösen, Ende abwarten S Starte zweite Karte, Ende abwarten r Referenzfahrt auslösen, eigene Ausführung fortsetzen s Starte zweite Karte, eigene Ausführung fortsetzen</p> <p><Achsen> = Achsangabe für Referenzfahrt</p>
Anwendung	<p>8 OS</p> <p>8 OR1</p>
Erläuterung	-



Programmierbeispiel

PAL-PC

```
#axis xy;
#input
repeat
  move 20(100),20(100);
  tell 0 reference x;
  move 20(100),20(100);
until 0;
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@03":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"0 20,100,20,100":gosub 1000
140 print#1,"8 0R1":gosub 1000
150 print#1,"0 -20,100,-10,100":gosub 1000
160 print#1,"3 0,-4":gosub 1000
170 print#1"9":gosub 1000
1000 ...
```



Bei Verwendung der Optionen r und s muss sichergestellt werden, dass ein erneuter Befehl erst nach Bearbeitungsende des momentanen Befehls in der angesprochenen Prozessorkarte, gesendet wird.

3.2 Ergänzender Befehlsumfang Interface-Karte 5.0

3.2.1 Befehl: 3D-Linearinterpolation

Zweck	Die Interface-Karte 5.0 erweitert die 2,5D-Interpolation des Standard-Betriebssystems auf eine 3D-Interpolation (räumlich). Mit dem Befehl können Sie diese Interpolation gezielt ein- und ausschalten.
Aufbau	<p>z<STATUS></p> <p><STATUS> = 0 - 3D-Interpolation aus = 1 - 3D-Interpolation ein</p>
Erläuterung	Die Anweisung wirkt modal, d. h. alle move- und moveto-Anweisungen werden dreidimensional ausgeführt. Die Angabe von z2-Parametern in diesen Verfahrbewegungen wird ignoriert. Als Geschwindigkeitsangabe der Interpolation wird der Wert der X-Achse herangezogen.



Programmierbeispiel

PAL-PC

```
#axis xyz;
reference xyz;
set3don;
move 10(700),15(800),3(400),
    0(30);
set3doff;
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
120 print#1,"@0i":gosub1000
130 print#1,"z1":gosub 1000
140 print#1,"0100,700,150,800,30,400,0,30"
145 gosub 1000
150 print#@1,"z0":gosub1000
160 print#1,"9":gosub 1000
1000 ...
```



(nähere Erläuterungen siehe Befehl *3D-Linearinterpolation* Kapitel 2.2.1).

3.2.2 Befehl: Zirkularinterpolation

Zweck	<p>Bearbeiten von Kreisen und Kreisbögen mit konstanter Bahngeschwindigkeit.</p> <p>Die Kreisinterpolation wird durch zwei aufeinanderfolgende Befehle ausgelöst. Der erste Befehl legt die Kreisrichtung fest, im zweiten werden die Interpolationsparameter übergeben.</p>
Aufbau	<p>Kreisrichtung f1 Gegenuhreigersinn f0 Uhrzeigersinn</p> <p>Kreisbogen y B,V,D,Xs,Ys,Rx,Ry</p> <p>B Bogenlänge - sie gibt die Länge des Bogens zwischen Start- und Endwinkel des Kreissegmentes in Schritten an.</p> <p>V Geschwindigkeit - sie legt die Positioniergeschwindigkeit während der Bearbeitung fest ($30 < V < 10\,000$).</p> <p>Rx Richtung X - die Parameter Rx u. Ry zeigen der Prozessorkarte an, in welchem Quadranten des Kreises die Interpolationen startet.</p> <p>Xs Startpunkt X - Xs und Ys geben den Startpunkt der Interpolation</p> <p>Ys Startpunkt Y bezogen auf den Kreismittelpunkt an.</p> <p>D Interpolationsparameter - Aufgrund der geringen Speicherkapazität erwartet die Prozessorkarte die Angabe des Quadranten, in dem der Kreisbogen beginnt und die tangentielle Richtung des zu beschreibenden Kreises.</p>
Erläuterung	(siehe Befehl <i>Zirkularinterpolation</i> Kapitel 2.2.2)



Programmbeispiel

Nach einer relativen Positionierung von 150 mm (600 Schritte) soll ein Viertelkreis entgegen dem Uhrzeigersinn ausgeführt werden. Der Radius des Kreises ist mit 50 mm (200 Schritte) vorgegeben, der Startwinkel beträgt 0° der Endwinkel 90°. Die Geschwindigkeit über den gesamten Verlauf soll 200 Schritte/s betragen.

PAL-PC

```
#axis xyz;
reference xyz;
move 150(200),150(200),
      15(800),0(21);
circle_ccw50(300),0,90;
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@07":gosub 1000
120 print#1,"@0r7":gosub 1000
130 print#1,"@0i":gosub 1000
140 print#1,"0 600,200,600,200,15,800,0,21"
145 gosub 1000
150 print#1,"f1":gosub 1000
160 print#1,"y6400,300,-400,800,-1,1,":gosub1000
170 print#1,"9":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



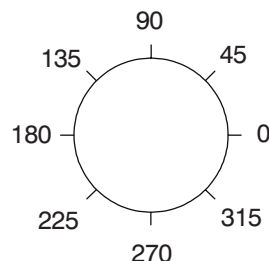
Bei Verwendung von PAL-PC wird die Berechnung der Parameter von der PC-Software übernommen. Somit beschränkt sich die Programmierung auf die Angabe von Radius, Verfahrgeschwindigkeit des Kreissegm. sowie den Start- und Endwinkel der Kreisbahn.

Die Unterscheidung der Bewegungsrichtung geschieht durch den Befehl circle_cw —> Bewegung im Uhrzeigersinn und circle_ccw —> Bewegung entgegen dem Uhrzeigersinn (siehe auch Beschreibung PAL-PC).



Beispiel: Kreisinterpolation in PAL-PC

Es sei ein Kreis mit einem Radius von 20 mm gegeben, als Arbeitsgeschwindigkeit werden 5 000 Hz angenommen. Nachfolgende Befehlszeilen zeigen die Programmierung bei unterschiedlichen Start- und Stoppwinkeln in positiver Richtung (gegen den Uhrzeigersinn).



circle_ccw 20(5000),0,360;	Vollkreis	Beginn und Ende bei 0°
circle_ccw 20(5000),0,45;	Kreisabschnitt	Beginn bei 0° und Ende bei 45°
circle_ccw 20(5000),45,225;	Kreisabschnitt	Beginn bei 45° und Ende bei 225°
circle_ccw 20(5000),225,585; Vollkreis		Beginn und Ende bei 225°

Bei Bewegung in negativer Richtung (im Uhrzeigersinn) achten Sie bitte darauf, dass der Startwinkel immer größer als der Stoppwinkel ist. Gegebenenfalls müssen Sie zum Startwinkel den Wert 360° (Vollkreis) addieren.

circle_cw 20(5000),360,0;	Vollkreis	Beginn und Ende bei 0°
circle_cw 20(5000),360,45;	Kreisabschnitt	Beginn bei 0° und Ende bei 45°
circle_cw 20(5000),405,225;	Kreisabschnitt	Beginn bei 45° und Ende bei 225°
circle_cw 20(5000),585,225;	Vollkreis	Beginn und Ende bei 225°

3.3 Ergänzender Befehlsumfang bei Interface-Karten mit E/A-Erweiterung

3.3.1 Befehl: Ausgangsport setzen

Zweck Die Prozessorkarte setzt am definierten Ausgangsport der E/A-Erweiterungseinheit ein gewünschtes Ausgangsmuster.

Aufbau p<Adresse>, <BITNR>, <Wert>

<Adresse> = Ausgangsport, 1 —> 65 529
 = Ausgangsport, 2 —> 65 530

<BITNR> = bitweise setzen, 1 <[BITNR]> 8
 = byteweise setzen, 128

<Wert> = 0 ... 255

Erläuterung Als <WERT> ist ein Zahlenwert einzugeben, der je nach <BITNR> die entsprechenden Ausgänge einzeln beschreibt oder bei der byteweisen Verarbeitung das Ausgangsmuster des kompletten Ports setzt.

1. bitweise setzen

Die Bitnummer bestimmt, welches Ausgangsbit bearbeitet wird; der Wert legt den Betriebszustand des Bits fest.

Befehl	Ausgangsport	Bit	Zustand
p65529,5,0	Port I	5	aus
p65529,4,1	Port I	4	ein
p65530,1,1	Port II	1	ein

2. byteweise setzen

Bei der byteweisen Bearbeitung des Ausgangsports legt der <WERT> das Bitmuster des kompletten Ausganges fest.

Befehl	Ausgangsport	Dualmuster
p65529,128,0	Port I	00000000
P65529,128,27	Port I	00011011
p65530,128,205	Port II	11001101
p65530,128,255	Port II	11111111



Programmierbeispiel

PAL-PC

```
#axis x;
reference x;
set_port 65529,5=0;
set_port 65530,128=27;
stop.
#start
.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"p 65529,5,0":gosub 1000
140 print#1,"p 65530,128,27":gosub 1000
150 print#1,"9":gosub 1000
160 print#1,"@0S":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Die Verarbeitung der Signalausgänge wird innerhalb der Prozessorkarte ablaufgesteuert durchgeführt. Somit ist ein Setzen bzw. Löschen von Ausgängen während einer Befehlsbearbeitung z. B. einer Positionierbewegung nicht möglich.

Bei Ausfall der Versorgungsspannung der Prozessorkarte schalten alle Ausgänge auf inaktiv. In Verbindung mit den open-Kollektor-Ausgängen der E/A-Erweiterungseinheit ergeben sich somit folgende Signalzustände:

Ausgang 1

Alle Ausgänge sind aktiv low

Durch Opto-Isolierung der Ausgänge und Einsatz einer externen Spannungsversorgung sind die Endstufen-Transistoren bei fehlender Ansteuerung durchgeschaltet, das Spannungspotential am Kollektor-Ausgang beträgt 1,0 V (VCEsat). Eine zwischen einem Ausgang und +Vs ext. angeschlossene Lampe leuchtet.

Ausgang 2

Alle Ausgänge sind inaktiv

Bei fehlender Steuerspannung am Eingang der Endstufen-Transistoren sind die Ausgänge offen, d. h. eine zwischen einem Ausgang und +Vs angeschlossene Lampe leuchtet nicht.

3.3.2 Befehl: Eingangsport lesen

Zweck Die Prozessorkarte liest das anstehende Bitmuster am Eingangsport der E/A-Erweiterung ein.

Aufbau o<Adresse>,<BITNR>,<Wert>,<Offset>
 <GN> = Gerätenummer, Standard = 0
 <Adresse> = Eingangsport 65 531
 <BITNR> = bitweise lesen, 1 < [BITNR] > 8
 = byteweise lesen, 128
 <Wert> = zu vergleichender Wert
 <Offset> = ist die Anzahl der Programmschritte, um die eine Verzweigung vor- oder rückwärts erfolgen soll.
 Dabei wird der Eingangsport auf das im Parameter <WERT> festgelegte Bitmuster überprüft und bei Erfüllung der Bedingung die Verzweigung durchgeführt.

Erläuterung Durch Angabe des Parameters <BITNR> unterscheidet das Betriebssystem zwischen einer bit- oder byteweisen Verarbeitung des Eingangsports.

1. bitweise lesen

Die Bitnummer bestimmt, welches Eingangsbit abgefragt wird.

Befehl	Abfragekriterium	Verzweigung
o65531,2,0,3	Bit 2 = aus	3 Zeilen vorwärts
o65531,8,1,-2	Bit 8 = ein	2 Zeilen zurück

2. byteweise lesen

Bei der byteweisen Bearbeitung der Signaleingänge wird das Bitmuster des kompletten Ports abgefragt.

Befehl	Abfragekriterium	Verzweigung
o65531,128,10,3	Dual 00001010	3 Zeilen vorwärts
o65531,128,0,-2	Dual 00000000	2 Zeilen rückwärts



Programmierbeispiel

PAL-PC

```
#axis x;
reference x;
on_port 65531,2=0,3;
on_port 65531,8=1,-2;
set_port 65530,1=1;
move 100(2000);
set_port 65530,2=1;
move -100(2000);
stop.
```

GW-BASIC

```
100 open"com1:9600,N,8,1,DS,CD"as #1
110 print#1,"@01":gosub 1000
120 print#1,"@0i":gosub 1000
130 print#1,"71":gosub 1000
140 print#1,"o 65531,2,0,3":gosub 1000
150 print#1,"o 65531,8,1,-2":gosub 1000
160 print#1,"p 65530,1,1":gosub 1000
170 print#1,"0 400,2000":gosub 1000
180 print#1,"p 65530,2,1":gosub 1000
190 print#1,"0 -400,2000":gosub 1000
200 print#1,"9":gosub 1000
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler : ";a$
1030 stop
```



Die Signaleingänge sind optoisolierte Eingänge. Durch integrierte Vorwiderstände an den Anoden der Opto-Koppler ist eine einfache Masse-Beschaltung der Kathoden-Eingänge ausreichend, um das Eingangsbit zu setzen.

Die Information der Signaleingänge wird auf der E/A-Erweiterungseinheit nicht zwischengespeichert. Somit gehen impulsartige Eingangssignale, die während der internen Verarbeitung eines Datensatzes anfallen, verloren.

3.4 Ergänzender Befehl in Verbindung mit einer Programmwahleinheit

3.4.1 Befehl: Tastaturabfrage

Zweck

Die Prozessorkarte fragt an der seriellen Schnittstelle den Code einer betätigten Taste der Programmwahleinheit ab. Entsprechend der empfangenen Information wird ein im Programm definierter Offset durchgeführt.

Aufbau	k <Tastenummer>, <Offset>
	<Tastenummer> = Zahl zwischen 1 und 20 (255)
	<Offset> = Anzahl der Programmschritte, um die eine Verzweigung erfolgen soll
Anwendung	k13,0 Warte, bis eine Taste gedrückt ist k1,2 überspringe nächsten Befehl, falls Taste 1 gedrückt
Erläuterung	Der Befehl k veranlasst den Prozessor, an der seriellen Schnittstelle eine Impulskette auszugeben und auf eine Rückinformation der angekoppelten Programmwahleinheit zu warten. Der empfangene Code bewirkt eine im Datenfeld vorgesehene Verzweigung des Programmablaufes. Wird keine Taste betätigt, ergibt sich als rückgesendete Information <Gesamtanzahl Tasten> + 1 (Programmwaheinheit mit 20 Taste —> 21).



Programmierbeispiel

PAL-PC

```
#axis x;
Anfang:
  repeat
    on_key 1, do_move;
    on_key 2, do_reference;
  goto Anfang;
do_move:
  move 100(2000);
  move -100(2000);
goto Anfang;
do_reference:
  reference x;
goto Anfang;
stop.
```

GW-BASIC

```
100 open "com1:9600,N,8,1,DS,CD" as #1
110 print#1, "@01":gosub 1000
120 print#1, "@0i":gosub 1000
130 print#1, "k1,4":gosub 1000
140 print#1, "k2,5":gosub 1000
150 print#1, "k3,6":gosub 1000
160 print#1, "3 0,-3":gosub 1000
170 print#1, "0 1000,1000":gosub 1000
180 print#1, "3 0,-5":gosub 1000
190 print#1, "71":gosub 1000
200 print#1, "3 0,-7":gosub 1000
210 print#1, "9":gosub 1000
220 stop
1000 if loc(1)<1 then goto 1000
1010 a$=input$(1,1)
1015 if a$="0" then return
1020 print "Karte meldet Fehler: ";a$
1030 stop
```



Die Programmwaheinheit ermöglicht keine direkte Programmierung der Prozessorkarte, sie kann nur eine vorher definierte Verzweigung im Programmablauf bewirken. In Verbindung mit der Software PAL-PC ist eine Programmierung durch "Labelzuweisung" wesentlich vereinfacht. Sie können einzelne Programmteile durch ein Label kennzeichnen und durch Betätigen der Taste mit demselben Label aktivieren.

4 Fehlermeldungen

4.1 Fehlermeldungen der Prozessorkarten

Fehlerbeschreibung	Ursachen	Beseitigung
0 Kein Fehler	Befehl wurde korrekt ausgeführt oder gespeichert	
1 Fehler in übergeordneter Zahl. Die Interface-Karte hat eine Zahlenangabe empfangen, die nicht korrekt interpretiert werden konnte.	<p>A Der übergebene Zahlenwert ist außerhalb des zulässigen Bereiches. Für 8-Bit-Angaben beträgt der zulässige Bereich - 128 bis + 127, für 16-Bit-Angaben - 32 768 bis + 32 767, für 24-Bit-Angaben - 8 388 608 bis + 8 388 607.</p> <p>B Der übergebene Zahlenwert enthält unzulässige Zeichen.</p>	Lassen Sie alle Ausgaben, die zur Interface-Karte gehen, protokollieren. Prüfen Sie anschließend an der Stelle, an der der Fehler auftritt, ob alle Wertangaben für den übertragenen Befehl korrekt sind.
2 Endschalter-Fehler	<p>Durch die Verfahrbewegung wurde ein Endschalter angesprochen. Die Schrittausgabe wurde gestoppt. Die Interface-Karte hat keine korrekte Sollposition mehr (Schrittverlust). Falls ein Programm ausgeführt wurde, wird dieses gestoppt.</p> <p>Die Referenzfahrt einer Schrittmotorachse wurde nicht korrekt ausgeführt.</p>	Die Achse, die im Endschalter steht, müssen Sie von Hand aus dem Endschalterbereich bewegen. Anschließend sollten Sie die Fehlerursache prüfen (falsche Verfahrwege, Überlastung der Anlage mit resultierendem Schrittverlust, Schwergängigkeit der Mechanik o. ä.) und nach deren Beseitigung das Programm neu starten. Neue Referenzfahrt durchführen.
3 Unzulässige Achsangabe	Der Interface-Karte wurde eine Achsangabe für einen auszuführenden oder einen zu speichernden Befehl Übermittelt, die eine nicht definierte Achse enthält.	Verwenden Sie in den Befehlen nur Werte gemäß Kapitel 1.1.1 Befehl <i>Achsenanzahl</i> setzen.

Fehlerbeschreibung	Ursachen	Beseitigung
4 Keine Achsen definiert	Bevor der Interface-Karte speicherbare Befehle, Bewegungen oder allgemeine Befehle, die eine von der Achsanzahl abhängige Anzahl v. Parametern haben, übergeben werden, muss der Befehl 'Achsenanzahl setzen' übergeben werden, um die internen Kartenparameter korrekt zu setzen. Die Achsenzahl bleibt nach dem Ausschalten erhalten, falls die Option "Batterie-Backup" verwendet wird. Sollte dennoch ein Fehler 4 auftreten, kann dies auf Batterieprobleme hinweisen.	
5 Syntax-Fehler	<p>A Ein Befehl wurde in Großschreibung verwendet, obwohl dieser Befehl nur in Kleinschreibung existiert.</p> <p>B Während der Übertragung eines Datenfeldes wurde versucht, einen speicherbaren Befehl zu verwenden.</p> <p>C Der verwendete Befehl existiert nicht.</p>	Lassen Sie alle Ausgaben, die zur Interface-Karte gehen, protokollieren. Prüfen Sie anschließend an der Stelle, an der der Fehler auftritt, ob alle übertragenen Befehle korrekt sind.
6 Speicherende	Es wurde versucht, mehr Befehle zu übertragen, als die Interface-Karte speichern kann.	Teilen Sie das Programm in kleinere Abschnitte auf, übertragen Sie jeweils einen Abschnitt, führen Sie diesen aus und übertragen Sie dann den nächsten Abschnitt.
7 Unzulässige Parameterzahl	Die Interface-Karte hat mehr oder weniger Parameter für den Befehl erhalten, als benötigt werden.	Prüfen Sie, ob die Anzahl der Parameter für den Befehl in Verbindung mit der Anzahl der Achsen korrekt ist. Berücksichtigen Sie hierbei die z2-Bewegung.
8 Zu speichernder Befehl nicht korrekt	Der Interface-Karte wurde ein Befehl zur Speicherung übergeben, der in dieser Form nicht vorhanden ist.	Prüfen Sie den übertragenen Befehl. Existiert der Befehlscode? Ist Groß-/Kleinschreibung berücksichtigt worden?

Fehlerbeschreibung	Ursachen	
A Impulsfehler	Die übergebene Option für den Befehl "Impuls" ist nicht im zulässigen Bereich zwischen 1 und 6.	
B Tell-Fehler	Die Tell-Funktion (starte zweite Interface-Karte) hat kein Endezeichen nach der Höchstzahl der zu sendenden Zeichen gefunden. Dieser Fehler deutet auf Speicherprobleme hin, da die Eingabe des Tell-Befehles immer ein Endezeichen anfügt.	
C (CR) erwartet	Die Interface-Karte hat auf das (CR)-Zeichen als Befehlsende gewartet. Sie haben jedoch ein anderes Zeichen übergeben. Dies ist hauptsächlich ein Problem der Parameteranzahl. Sie versuchen mehr Parameter zu übertragen, als für die Funktion notwendig sind.	
D Unzulässige Geschwindigkeit	Beachten Sie, dass auch für Achsen ohne Bewegung eine zulässige Geschwindigkeit benötigt wird, d. h. 0,0 als Wertepaar ist nicht zulässig.	
E Schleifenfehler	Es wurde versucht, eine Vorwärtsschleife auszuführen. Beachten Sie, dass bei Schleifen immer die letzten n-Befehle wiederholt werden, d. h. 34,4 ist unzulässig.	

Fehlerbeschreibung	Ursachen	
F BenutzerStopp	<p>Sie haben Stopp an der Interface-Karte gedrückt. Die Befehlsausführung können Sie mit der Starttaste oder dem Befehl @0s wieder aufnehmen. Nach einem Stopp sind die Befehle @0P und @0Q zulässig! (@0P, um die erreichte Position abzufragen und @0Q, um die Ausführung endgültig abzuberechnen).</p> <p>Fehler F fällt aus dem üblichen Software-Handshake-Protokoll. Ein zusätzliches F kann jederzeit auftreten, falls Sie an der Interface-Karte während einer Verfahrbewegung den Stopp-Taster drücken. Um diesem Verhalten Rechnung zu tragen, sollte die Unterroutine der Software, die die Rückmeldung verarbeitet, mit der Behandlung dieses Sonderfalles ergänzt werden.</p>	
= (cr)-Fehler	Die Interface-Karte hat ein (CR) empfangen, obwohl noch weitere Parameter für den aktuellen Befehl erwartet wurden.	

In Ergänzung zur Fehlerart "Referenzschalter angefahren" stellt das Betriebssystem der Prozessoreinheiten ab Produktionsdatum 10/93 eine zusätzliche Fehlerart mit diesem Syntax (Fehler 2) dar.

Es handelt sich hierbei um eine inkorrekt ausgeführte Referenzfahrt, d. h. eine Referenzfahrt-Bewegung, die nicht durch einen Interrupt des Referenzschalters abgebrochen wurde (der Referenzschalter wurde beim An- oder Freifahren nicht erkannt), vgl. "Ausführung einer Referenzfahrt", siehe Befehl Referenzfahrt Seite 7 bzw. 41.

In den verschiedenen Betriebsmodi der Prozessorkarte führt der Fehler zu folgenden Funktionen:

Mode	Start durch	Ergebnis
DNC-Mode	@0R7	als Rückmeldung wird Fehlercode 2 ausgegeben
	@0r7	zusätzlicher Fehlercode 2 am Ende der Bewegung
CNC-Mode	@0S	als Rückmeldung wird Fehlercode 2 ausgegeben
	@0s	Die Ausführung des Programmes wird abgebrochen
	Start-Taste	zusätzlicher Fehlercode 2 am Ende der Bewegung Die Ausführung des Programmes wird abgebrochen

4.2 Fehlermeldungen PAL-PC

1	unexpected End of File	unerwartetes Fileende
2	' ; ' expected	' ; ' erwartet
3	illegal axis-entry	falsche Achsangabe
4	'x', 'xy', 'xz' or 'xyz' expected	'x', 'xy', 'xz' oder 'xyz' erwartet
5	axis already defined	Achsen schon definiert
6	'mm', 'cm', 'zoll', 'zoll/10' or 'zoll/20' expected	'mm', 'cm', 'zoll', 'zoll/10' oder 'zoll/20' erwartet
7	missing 'stop.', stop assumed	'stop.' fehlt, stop. ergänzt
8	input already active	Inputbefehl schon aktiv
9	too much nested repeats (Limit is 20)	zu viele geschachtelte Wiederholungen (maximal 20)
10	repeat without until detected	repeat ohne until gefunden
11	#-command not recognized	#-Befehl ist unbekannt
12	duplicate axis entry in command	doppelte Achsangabe in Befehl
13	'x', 'y' or 'z' expected	'x', 'y' oder 'z' erwartet
14	integer expected	Integer erwartet
15	' , ' expected	' , ' erwartet
16	positive integer expected	Positive Integergröße erwartet
17	until without repeat	until ohne repeat
18	real number expected	Realgröße erwartet
19	positive real number expected	Positive Realgröße erwartet
20	missing '#input'	vermisse '#input'
21	' (' expected	' (' erwartet
22	') ' expected	') ' erwartet
23	' . ' expected	' . ' erwartet
24	too much definitions	zu viele Definitionen (maximal 50)
25	definition name expected	Name für Definition erwartet
26	illegal character for send or wait (number between 1..126 expected)	illegales Zeichen für send oder wait (Zeichen zwischen 1 ... 126 erwartet)
27	' " ' or unit number expected	' " ' oder Gerätenummer erwartet
28	' " ' expected	' " ' erwartet
29	' wait ' expeted	' wait ' erwartet
30	unit entry expected	Gerätenummer erwartet
31	command not recognized	Befehl wird nicht unterstützt
32	too much label definitions	zu viele Marken definiert (maximal 50)

33	positive integer between 1 and 126 expected	Positive Integergröße zwischen 1 und 126 erwartet
34	label not found	Marke nicht gefunden
35	no label definition in text	keine Markendefinition im Text
36	' , ' or 'times' expected	' , ' oder 'times' erwartet
37	'in' or 'out' expected	'in' oder 'out' erwartet
38	'on', 'off', 'in', 'out' or 'sync' expected	'on', 'off', 'in', 'out' or 'sync' erwartet
39	end of remark missing	Ende einer Bemerkung nicht gefunden
40	serial transmission error (time out in receive)	Übertragungsfehler (time out beim Empfangen)
41	elevation must be > 0.001	Steigung muss größer 0,001 sein
42	file not found	File nicht gefunden
43	letter or '_' expected	Buchstabe oder '_' erwartet
44	replace text exceeds 250 chars	Textersatz zu lang (max. 250 Zeichen)
45	line exceeds 250 chars after replace of definition	Zeile ist nach Textersatz länger als 250 Zeichen
46	illegal definition occurred	nicht erlaubte Definition
47	' " ' or ' < ' expected	' " ' oder ' < ' erwartet
48	' " ' expected	' " ' erwartet
49	' > ' expected	' > ' erwartet
50	include file not found or i/o error	Includefile nicht gefunden o. E/A-Fehler
51	i/o error on read	E/A-Fehler beim Lesen
53	illegal unit-no	unerlaubte Gerätenummer
54	'xy', 'xz' or 'yz' expected	'xy', 'xz' oder 'yz' erwartet
55	positiv real number expected	Positive Realgröße erwartet
56	no matching definition for redefine	keine gültige Definition für Redefinition
57	'*' expected	'*' erwartet
58	forward loop not allowed	Schleife mit positiv. Offset nicht erlaubt
59	'=' expected	'=' erwartet
60	GUZ or UZ expected	GUZ oder UZ erwartet
61	starting angle must be less ending angle	Startwinkel muss < als Endwinkel sein
62	starting angle must be greater ending angle	Startwinkel muss > als Endwinkel sein
63	Zero circle not allowed	Bögen mit Längen 0 nicht erlaubt
149	invalid number (Interface)	Fehler in übergebener Zahl (Interface)
150	reference switch (Interface)	Endschalter (Interface)

151	invalid axis (Interface)	unzulässige Achsangabe (Interface)
152	no axis information (Interface)	keine Achsen definiert (Interface)
153	syntax error (Interface)	Syntax-Fehler (Interface)
154	out of memory (Interface)	Speicherende (Interface)
155	invalid number of parameters (Interface)	unzulässige Parameterzahl (Interface)
156	incorrect command (Interface)	Befehl inkorrekt (Interface)
161	(cr) error	(cr)-Fehler (Interface)
164	self test not terminated or cable error	Selbsttest nicht abgeschlossen oder Übertragungsfehler (Interface)
165	pulse error (Interface)	Impulsfehler (Interface)
166	tell error (Interface)	Tellfehler (Interface)
167	(cr) expected (Interface)	(cr) erwartet (Interface)
168	invalid speed (Interface)	unzulässige Geschwindigkeit (Interface)
169	loop error (Interface)	Schleifenfehler (Interface)
170	user stop (Interface)	BenutzerStopp (Interface)
100 ... 199	Interface-Card error (100+Error)	Fehlermeldungen der Interface-Karte 100+Fehler)