

IPCOMM

phytron®

Kommunikations-Software für
Schrittmotorensteuerungen
IPP, GSP, GCD und GLD



Manual 1086-A010 D

customized solutions
in motion

IPCOMM

Kommunikations-Software für Schrittmotorsteuerungen IPP, GSP GCD und GLD

© 2005

Alle Rechte bei:

Phytron-Elektronik GmbH

Industriestraße 12

82194 Gröbenzell, Deutschland

Tel.: +49(0)8142/503-0

Fax: +49(0)8142/503-190

Alle Angaben in diesem Handbuch erfolgen nach bestem Wissen, aber ohne Gewähr. Wir behalten uns im Interesse unserer Kunden vor, Verbesserungen und Berichtigungen an Hardware, Software und Dokumentation jederzeit ohne Ankündigung vorzunehmen.

Für Anregungen und Kritik sind wir dankbar.

(E-Mail an doku@phytron.de)

Inhaltsverzeichnis

1 IPCOMM zum Kennenlernen.....4	5.3.9 RS-Monitor >> 23
1.1 Die ersten Schritte4	5.3.10 SFI >>..... 24
1.2 Die Leistungsmerkmale im Überblick ...4	6 Fehlermeldungen 25
2 Systemvoraussetzungen5	6.1 Schnittstellenfehler..... 25
3 Programminstallation.....5	7 Adapterkabel..... 26
3.1 Durchführung der Programminstallation.....6	8 Copyright und Haftungsausschluss..... 28
3.2 Dateizeichnungen8	9 Das IPCOMM-Protokoll..... 29
3.3 Update von IPCOMM.....8	9.1 Format des Sendetelegramms..... 30
4 Die Arbeitsumgebung9	9.2 Format des Antworttelegramms 31
5 Die Menü-Referenz.....10	9.3 Behandlung von Übertragungsfehlern 32
5.1 Aufrufen und Schließen von IPCOMM10	9.4 Statusinformation 33
5.2 Das Menü <i>Datei</i>10	9.4.1 Kurzstatus 33
5.2.1 Neu10	9.4.2 Erweiterter Status..... 34
5.2.2 Öffnen10	9.5 Befehlsdefinitionen..... 36
5.2.3 Speichern.....10	9.5.1 Stop-Befehle „H“ und „B“..... 36
5.2.4 Speichern unter11	9.5.2 Fahrbefehle „G“..... 37
5.2.5 Schließen.....11	9.5.3 Parameterbefehle P 38
5.2.6 Drucken11	9.5.4 I/O-, Status- und Infobefehle „I“.. 39
5.2.7 Programmende11	9.5.5 Rücksetzbefehle „C“..... 40
5.3 Das Haupt-Eingabefenster12	9.5.6 Flash-Programmierbefehle F..... 41
5.3.1 Übertragen / Schließen.....12	9.5.7 Schreibbefehle W 41
5.3.2 Fahrbefehle.....13	9.5.8 Erweiterte Befehle E..... 42
5.3.3 Parameter14	9.5.9 Retrybefehl R 42
5.3.3.1 Strom >>14	9.5.10 Sonderbefehle S..... 43
5.3.4 Streckenregister.....15	9.6 Online-Betrieb einer PSS 44
5.3.5 Kurzstatus der Achsen.....15	9.7 SPS-Betrieb einer PSS 44
5.3.5.1 Status >> 16	9.8 Zusatzinformationen..... 46
5.3.5.2 Reset>> 17	9.8.1 Tools..... 46
5.3.6 Digitaler Eingangs- / Ausgangsstatus..... 17	9.8.2 Softwarestruktur 47
5.3.7 Achse programmieren.....18	9.8.3 SPS-Sequenzen konfigurieren ... 48
5.3.7.1 SPS >> 18	9.8.4 Erweiterte SPS-Sequenzen..... 49
5.3.7.2 SPS Erw. >> 19	9.8.5 Schnittstellenkommunikation..... 51
5.3.7.3 Rampen >>21	9.8.6 Auswahl der Baudrate 53
5.3.7.4 Aktuelle Parameter22	10 Stichwortverzeichnis 54
5.3.8 Optionale Zusatzkarte >>22	

Manual IPCOMM

1 IPCOMM zum Kennenlernen

IPCOMM ist ein Windows®-Programm, das als Inbetriebnahmesoftware mit jeder PSS (Phyton **S**chrittmotor **S**teuerung) ausgeliefert wird. Es dient zur Parametereinstellung und Programmierung von SPS-Sequenzen in den PHYTRON-Schrittmotorsteuerungen IPP, GSP, GCD und GLD.

1.1 Die ersten Schritte

Anmerkung: Da Microsoft® Windows® ein für die meisten Anwender vertrautes Betriebssystem ist, wird für die Bedienung von IPCOMM die Anwendung windowsspezifischer Eingaben, Schalter, Mausclicks vorausgesetzt. Hilfreiche Informationen zu Ihrem Windows Betriebssystem finden Sie in Ihrem Windows Handbuch.

- Überprüfen Sie, ob Ihr PC den Systemvoraussetzungen auf Seite 5 (Kap. 2) entspricht.
- Installieren Sie IPCOMM nach den Anweisungen ab Seite 6 (Kap 3.1).
- Starten Sie das Programm durch Klicken der Schaltfläche **Start** auf der Taskleiste. Im Menü *Start* wählen Sie **Programme/IPComm V2/IPCommV2** durch den linken Mausclick. Oder starten Sie durch einen Doppelklick auf das gewählte IPCOMM-Icon.

Wichtig:

- Probieren Sie einfach einige Funktionen aus, die Sie interessieren. Das Programm ist sehr einfach zu bedienen und selbsterklärend.
- Für weitergehende Informationen lesen Sie bitte in diesem Manual die entsprechenden Punkte nach. Neben dem Inhaltsverzeichnis und vielen Textverweisen erleichtert Ihnen ein Stichwortverzeichnis auf den letzten Seiten den Überblick über alle Programmfunktionen.
- Programmierbefehle und sonstige Angaben über die Steuerung finden Sie in den entsprechenden Programmier- bzw. Gerätehandbüchern.
- Parameter, die in der Steuerung gespeichert sind, oder mit IPCOMM neu konfigurierte Dateien können ausgedruckt werden.

1.2 Die Leistungsmerkmale im Überblick

- Menügesteuerte und mausunterstützte Benutzeroberfläche
- Hotkey-Funktionen
- Menüdialog in 3 Sprachen (deutsch, englisch, französisch)
- IPCOMM arbeitet in zwei Modi: Offline und Online
- Einfache und schnelle Parametereinstellungen der Steuerungen

2 Systemvoraussetzungen

IPCOMM stellt an das eingesetzte Computersystem folgende Anforderungen:

- Pentium 100 MHz
- Betriebssystem Windows® 95, 98, 2000, NT oder auch XP®
- Mindestens 32 MB RAM Arbeitsspeicher
- Maus
- CD-ROM Laufwerk, wenn über CD-ROM installiert wird.
- Freie Schnittstelle RS232 zum Anschluss der Steuerung

3 Programminstallation

Auf der CD sind folgende Ordner gespeichert:

English	enthält die Dateien für das Setup in englischer Sprache
French	enthält die Dateien für das Setup in französischer Sprache
German	enthält die Dateien für das Setup in deutscher Sprache

In jedem Ordner sind folgende Dateien gespeichert

Setup.exe	Installationsdatei
Setup.ini	Konfigurationseinstellungen
Setup IPComm.msi	Windows Installer Paket
InstMsiA.exe	Gepackte Installerdatei A
InstMsiW.exe	Gepackte Installerdatei W

Manual IPCOMM

3.1 Durchführung der Programminstallation

1. Legen Sie die CD in das CD-ROM Laufwerk.
2. Wenn das Autostartprogramm nicht automatisch startet, öffnen Sie das Phyttron-Installationsprogramm [CD-ROM-Adresse]Autostart.exe durch Mausclick.
3. Das Phyttron-Installationsprogramm öffnet folgendes Menü:



Starten Sie durch Mausclick auf **IP-COMM installieren** das IPCOMM-Installationsprogramm.

4. Nach dem Programmstart kopiert das Installationsprogramm die Dateien in den Arbeitsspeicher.
5. Am Bildschirm erscheint folgende Meldung:

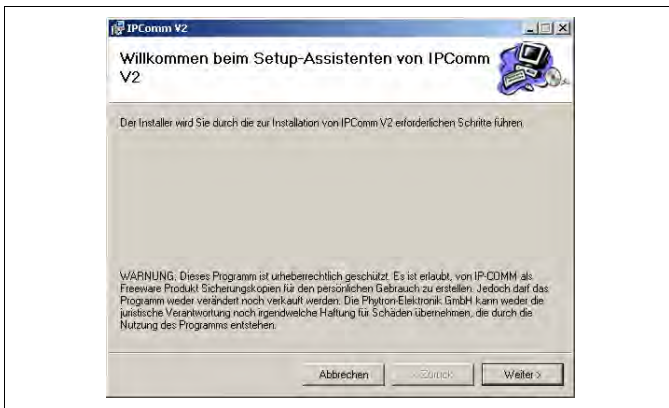


Abb. 1: Setup-Programm für IPCOMM

6. Nach Klicken auf **Weiter >** erscheint folgendes Fenster

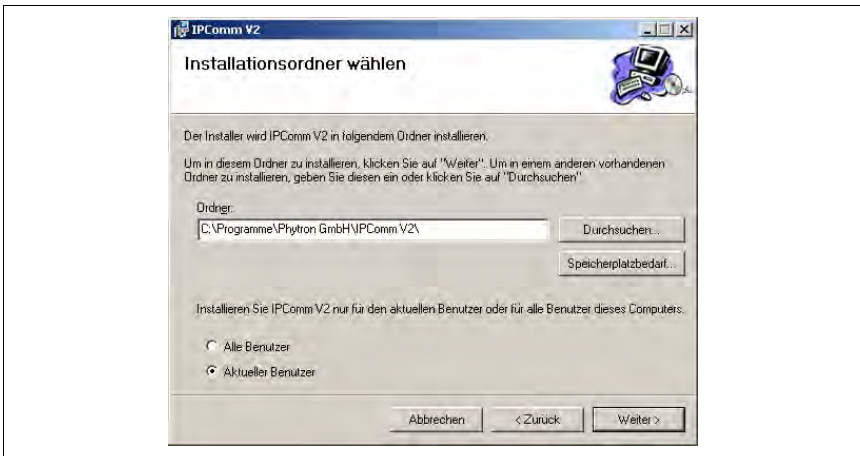


Abb. 2: Setup-Start für IPCOMM

7. Als Standardverzeichnis, in das das Programm gespeichert werden soll, wird **c:\Programme\Phytron GmbH\IPComm V2** vorgeschlagen. Mit **Durchsuchen** kann ein anderes Verzeichnis ausgewählt werden. Klicken Sie auf den Button **Weiter>**, um die Installation zu starten.
8. Es werden dann die Dateien in das angegebene Verzeichnis kopiert, einzelne Dateien werden auch in den Windows Ordner gespeichert.
9. Wenn am Bildschirm die Meldung *,IPComm V2 wurde erfolgreich installiert. Klicken Sie auf Schließen'* erscheint, ist die Installation erfolgreich beendet. Mit **Schließen** beenden.
10. Entfernen Sie die CD aus dem CD-ROM Laufwerk.
11. Zum schnellen Start des Programms empfehlen wir, das IPCOMM V2-Icon, das auf den Desktop gespeichert wurde, zu verwenden.

Hinweis:

Haben Sie die Dateien von IPCOMM auf anderem Weg (z. B. email) erhalten, so erfolgt die Programminstallation ähnlich wie in 3.1 (ab Punkt 2) beschrieben. Anstelle des CD-ROM-Laufwerks wählen Sie das entsprechende Verzeichnis, auf dem die Datei SETUP.exe gespeichert ist.

Manual IPCOMM

3.2 Dateibezeichnungen

Nach der Installation sind im gewählten Verzeichnis folgende Dateien gespeichert:

IPComm.exe	das Anwendungsprogramm IPCOMM
IPS2000.exe	das Anwendungsprogramm IPSearch (siehe Kap.9.8.1)
opm.exe	das Anwendungsprogramm opm (siehe Kap. 9.8.1)
comdlg32.dll	Programmbibliothek
oledlg.dll	Programmbibliothek
SFI	Ordner mit hinterlegten Daten für Endstufe GSP mit Schrittfehlererkennung (SFI)

3.3 Update von IPCOMM

Wollen Sie ein Update über eine bestehende IPCOMM Version installieren, wird nach Start des Installationsprogramms die bestehende Version auf Anfrage entfernt und dann die neue Version installiert.

4 Die Arbeitsumgebung

Nach dem erfolgreichen Start von IPCOMM zeigt der Bildschirm folgende Arbeitsfläche:

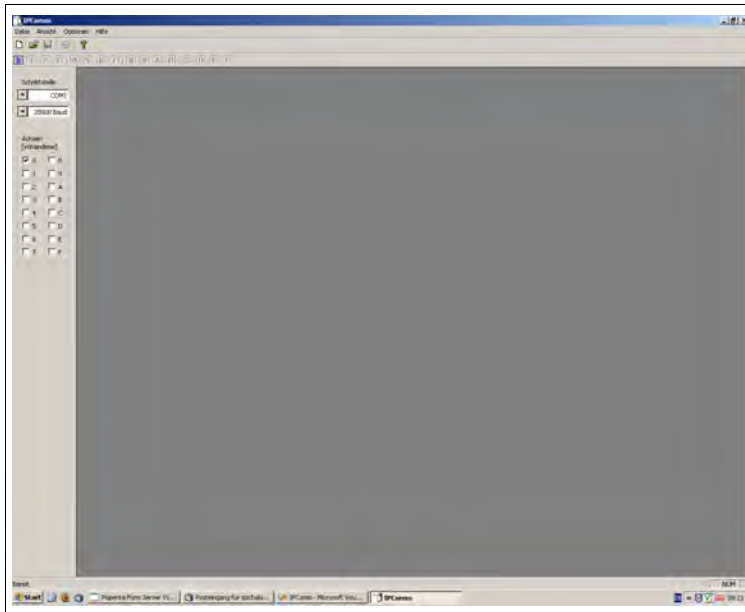







Abb. 3: Die Arbeitsoberfläche nach Programmstart

Die Arbeitsumgebung besteht aus 6 Komponenten:

1. Auswahlmnü in der obersten Zeile:
Datei, Ansicht, Optionen, Fenster, Hilfe
2. Auswahl einiger Menübefehle mit Icons (siehe Tabelle)
3. Auswahl der COM-Schnittstelle und der Baudrate
4. Auswahl der Geräteadresse: Adresse 0 bis 9 und A bis F
5. Arbeitsfläche
6. Statuszeile

Icon	Bedeutung der Icons
	Neue Parameterdatei erstellen
	Parameterdatei öffnen
	Parameterdatei speichern
	Parameterdatei drucken
	Informationen über IPCOMM

Manual IPCOMM

5 Die Menü-Referenz

Dieses Kapitel fasst zusammen, was Sie für den Start und das Verlassen von IPCOMM wissen müssen, es enthält außerdem eine detaillierte Zusammenstellung der einzelnen Menü-Befehle, Dialogfenster, Schalter usw.

5.1 Aufrufen und Schließen von IPCOMM

Wählen Sie **Start / Programme / Phyttron GmbH / IPComm V2** mit der Maus und bestätigen Sie mit der linken Maustaste den Programmstart. Oder klicken Sie – falls vorhanden – das IPCOMM-Icon auf Ihrem Desktop an.

Mit **Datei / Beenden** beenden Sie IPCOMM. Wenn Sie Änderungen nicht gespeichert haben, erscheint vor dem Verlassen von IPCOMM eine entsprechende Abfrage.

5.2 Das Menü *Datei*

Über das Dateimenü können Sie Parameterdateien öffnen, neu anlegen, Veränderungen sichern, eine Parameterdatei unter einem neuen Namen speichern, Dateien drucken und IPCOMM verlassen.

5.2.1 Neu

Im Untermenü **Datei / Neu** öffnen Sie das Menüfenster, mit dem im Offline- oder Online-Modus eine neue Parameterdatei erstellt werden kann.

IPCOMM vergibt den Namen <NAME>.IPC.

5.2.2 Öffnen

Im Untermenü **Datei / Öffnen** können Sie eine vorhandene Parameterdatei auswählen. Das Dialogfenster enthält ein Eingabefeld, eine Dateiliste und die Aktionsschalter **Öffnen** und **Abbrechen**.

Wählen Sie in der Dateiliste den Dateinamen aus. Die Datei wird geöffnet und in ein neues Editor-Fenster geladen.

5.2.3 Speichern

Datei / Speichern speichert die Datei des aktiven Eingabefensters.

5.2.4 Speichern unter

Mit **Datei / Speichern unter** können Sie die Datei im aktiven Editorfenster unter einem anderen Namen in einem anderen Verzeichnis und/oder Laufwerk speichern.

5.2.5 Schließen

Mit **Datei / Schließen** wird die Datei im aktiven Editorfenster geschlossen. Das Programm IPCOMM bleibt geöffnet.

5.2.6 Drucken

Datei / Drucken öffnet das windowsspezifische Druckuntermenü, in dem Sie auswählen können, auf welchem Drucker und mit welchen Druckeigenschaften die Parameterdatei gedruckt werden soll.

5.2.7 Programmende

Mit **Datei / Beenden** beenden Sie IPCOMM. Bevor das Programm verlassen wird, fordert IPCOMM Sie auf, noch nicht gesicherte Dateien abzuspeichern.

Manual IPCOMM

5.3 Das Haupt-Eingabefenster

Mit **Datei / Neu oder Öffnen** wird ein Eingabefenster geöffnet, mit dem Parameter, Fahrbefehle, Achsen, Streckenregister definiert oder Achsen-Status und Digital-I/O abgefragt werden können:

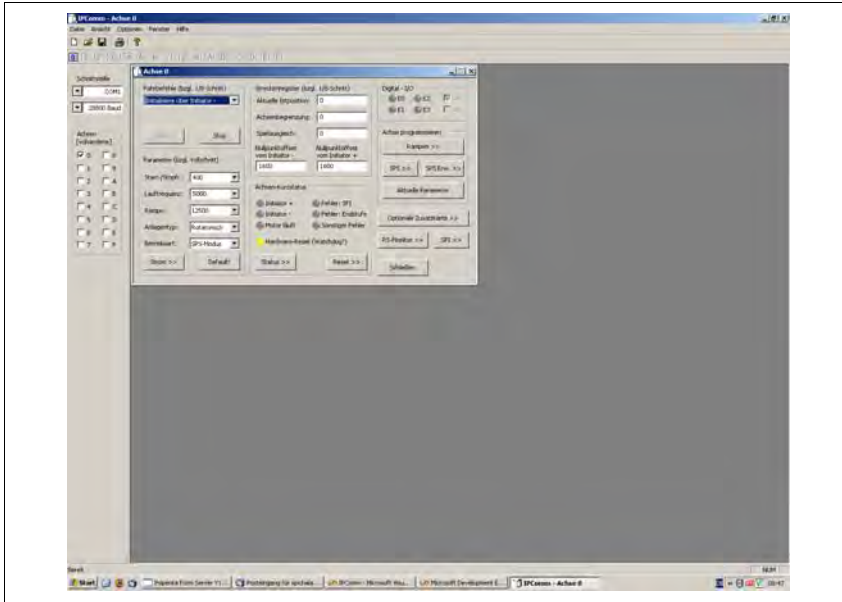


Abb. 4: Das Haupt-Eingabefenster

5.3.1 Übertragen / Schließen

Schließen

Die Parameterdatei <NAME>.ipc ohne Speichern schließen.

Übertragen

Im OFFLINE-Modus:
Die eingestellten Werte werden an die Steuerung übertragen.

Hinweis:

Erzeugen Sie eine neue Parameterdatei, so werden die Parameter für alle Achsen angelegt, die per Mausklick an der Adressleiste ausgewählt wurden.

5.3.2 Fahrbefehle

Im Eingabefenster kann durch Klicken des Drop-Down-Pfeiles gewählt werden, wie der Motor verfahren wird. Die Fahrbefehle beziehen sich auf 1/8-Schritt:

Fahrbefehle	Bedeutung
Initialisiere über Initiator +	Die Achse wird in + Richtung zu einem reproduzierbaren Anfangspunkt gefahren.
Initialisiere über Initiator –	Die Achse wird in – Richtung zu einem reproduzierbaren Anfangspunkt gefahren.
Positioniere absolut...	Bewegung auf absolute Position
Verfahre relativ...	Bewegung von der aktuellen Position um x-Zählerwerte
Freier Lauf Richtung +	freier Motorlauf in + Richtung
Freier Lauf Richtung –	freier Motorlauf in – Richtung
Ein Schritt Richtung +	1 Motorschritt in + Richtung
Ein Schritt Richtung –	1 Motorschritt in – Richtung

Start

Nur im ONLINE-Modus: Der eingestellte Fahrbefehl wird ausgeführt.

Stop

Der Motor wird angehalten.

5.3.3 Parameter

Im Eingabefenster können durch Klicken des Drop-Down Pfeils verschiedene Parameter (bezogen auf einen Vollschrift) eingestellt werden.

Parameter	Werte	Bedeutung
Start-/Stop-Frequenz	50...1000	Frequenz in Hz
Lauffrequenz	50...10000	Frequenz in Hz
Rampe	programmierbar	Lauffrequenz des Schrittmotors bei Beschleunigung und Abbremsen
Anlagentyp	Rotatorisch Linear	Motor treibt Achse an, die keine Endbegrenzung hat (Drehachse). Motor treibt Achse mit räumlicher Begrenzung an (z.B. lineare Achse oder Rundtisch mit Anschlag)
Betriebsart	PC-Anschluss SPS-Modus	Direkter PC-Anschluss an Endstufe SPS-Anschluss

5.3.3.1 Strom >>

Mit **Strom >>** startet das Untermenü zur Stromeinstellung.

Lauf-, Stop- und Booststrom können stufenweise durch Klicken des Drop-Down-Pfeils eingestellt werden: Stufe 0 bis 9, A bis F.

Die Stromüberhöhungszeit kann Werte von 10 bis 1000 ms erhalten.

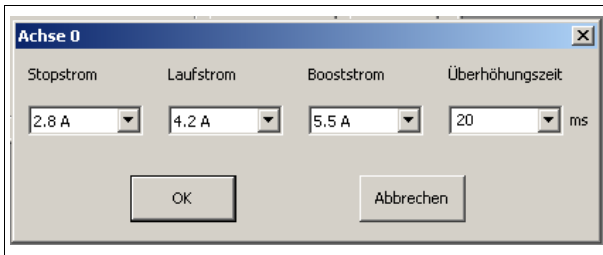


Abb. 5: Stromeinstellung

Default!


Die Parameterwerte werden auf die internen Auslieferungsdaten (Defaultwerte) gesetzt.


5.3.4 Streckenregister

Hier können verschiedene Daten des Streckenzählers eingelesen oder geändert werden. Die Werte beziehen sich auf 1/8-Schritt:

Streckenregister	Bedeutung
Aktuelle Istposition	Zeigt aktuelle Position des Streckenzählers an
Achsenbegrenzung	Wert, der im freien Lauf und bei der Initialisierung verwendet wird. Bei Erreichen dieser Schrittzahl wird der Lauf abgebrochen.
Spielausgleich	Beim Positionieren einer spielbehafteten Mechanik wird das Spiel eliminiert und mit diesem Wert definiert.
Nullpunktoffset vom Initiator –	Nach Erreichen des Referenzpunktes wird um diesen Wert in – Richtung verfahren
Nullpunktoffset vom Initiator +	Nach Erreichen des Referenzpunktes wird um diesen Wert in + Richtung verfahren

5.3.5 Kurzstatus der Achsen

Hier werden 7 Zustandswerte der Achsen mit Kontrolllampen  angezeigt:

Farbe der 	Bedeutung
grün	aktiv / in Ordnung
gelb	Warnung
rot	Fehler
grau	nicht aktiv / nicht verfügbar

Manual IPCOMM

Status	Kontrolllampe leuchtet
Initiator +	rot oder grau
Initiator –	rot oder grau
Motor läuft	grün oder grau
Fehler: SFI	rot oder grau
Fehler: Endstufe	rot oder grau
Sonstiger Fehler	rot oder grau
Hardware-Reset (Watchdog?)	gelb oder grau

5.3.5.1 Status >>

Mit **Status >>** startet das Untermenü zur Gesamtstatusabfrage.

Hier werden die Zustände der Achsen durch Kontrolllampen angezeigt:

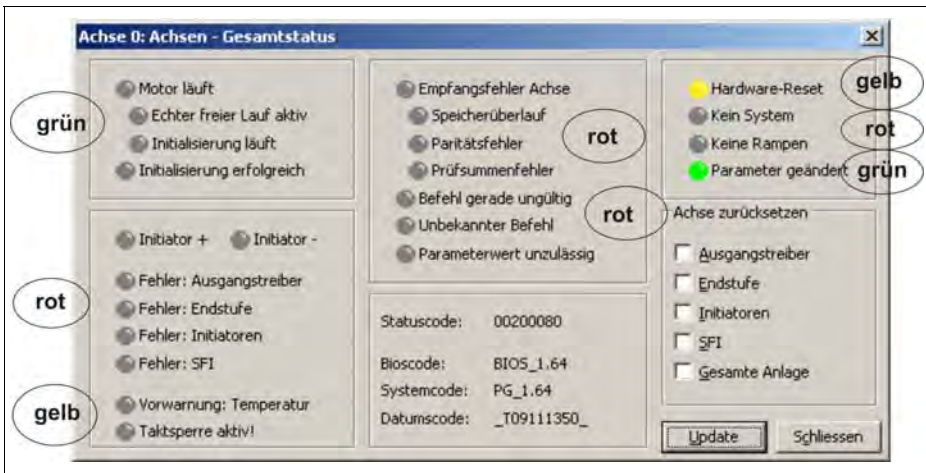


Abb. 6: Achsen Gesamtstatus

Achse zurücksetzen

Hier können die Ausgangstreiber, Endstufe, Initiatoren, SFI und/oder die Gesamte Anlage zurückgesetzt werden.

Wird 1 von den 5 Anlagenbereichen durch Mausklick mit markiert, trägt der **Update**-Button die Bezeichnung **Reset**, um das Reset durch Mausklick durchzuführen.

Update

Eine neue Statusabfrage wird gestartet.

Schließen

Beenden des Untermenüs

5.3.5.2 Reset>>

Mit **Reset >>** startet das Untermenü zum Reset der Anlage

Durch Mausklick können die Ausgangstreiber, Endstufe, Initiatoren, SFI und die Gesamte Anlage ausgewählt werden.



Abb. 7: Untermenü: Reset

Reset

Die per Mausklick ausgewählten Anlagenbereiche werden zurückgesetzt.

Schließen

Das Untermenü wird ohne Aktivierung des Reset verlassen.

5.3.6 Digitaler Eingangs- / Ausgangsstatus

In diesem Abschnitt werden die Zustände der 4 Eingänge (E0 bis E3) überwacht:

E0 bis E3 leuchten grün oder grau. Die Ausgänge A0 und A1 können per Mausklick gesetzt werden.

Manual IPCOMM

5.3.7 Achse programmieren

5.3.7.1 SPS >>

Mit **SPS >>** startet das Untermenü SPS-Funktionalität.

Hier können für den SPS-Betrieb bis zu 256 (0..255) Fahrbefehle (Sequenzen) geschrieben werden. Außerdem ist es möglich, die Sequenzen in einer SPS-Protokollsimulation zu testen und Ausgangssignale zu setzen.

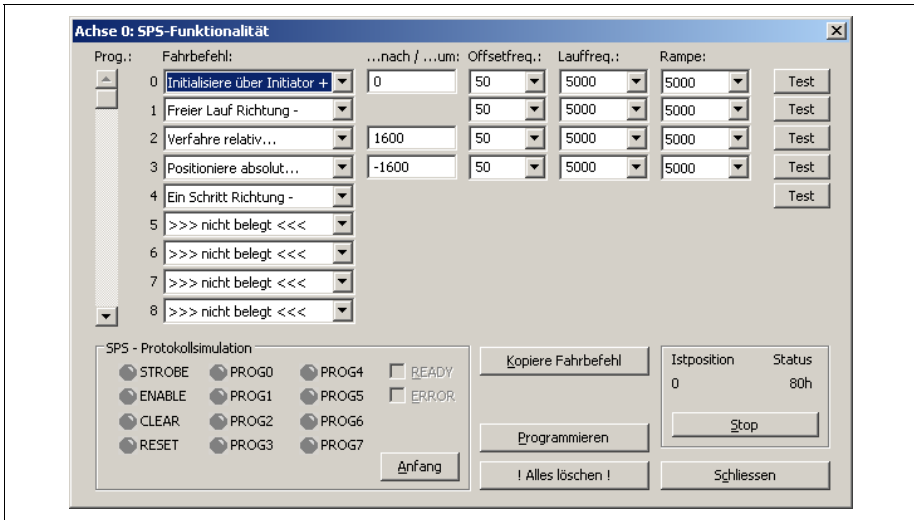


Abb. 8: Untermenü: SPS-Programmierung

Test	Sequenz testen.
Anfang	Start der SPS-Protokollsimulation, die mit Ende beendet wird.
Kopiere Fahrbefehl	Die markierte Sequenz wird auf die folgenden Sequenzen kopiert.
Programmieren	Im Online-Modus werden die ausgewählten Fahrbefehle in die Steuerung gespeichert.
Werte übernehmen	Im Offline-Modus werden die eingestellten Fahrbefehle im Zwischenspeicher abgelegt und können später in der Parameterdatei gespeichert.
! Alles löschen !	Löschen aller Fahrbefehle.

Stop	Fahrbefehl abbrechen
Schließen	Das Untermenü verlassen.

Unter **Istposition** wird der aktuelle Zählerstand, unter **Status** der Achsen-Kurzstatus in hexadezimal angezeigt.

5.3.7.2 SPS Erw. >>

Mit **SPS Erw. >>** startet das Untermenü Erweiterte SPS-Funktionalität.

Hier werden die eingegebenen Fahrbefehle (bis zu 256 (0..255)) automatisch nacheinander ausgeführt. Diese Ablaufsteuerung funktioniert solange das STROBE und ENABLE Signal anliegen. (Siehe Kap. 9.8.4)

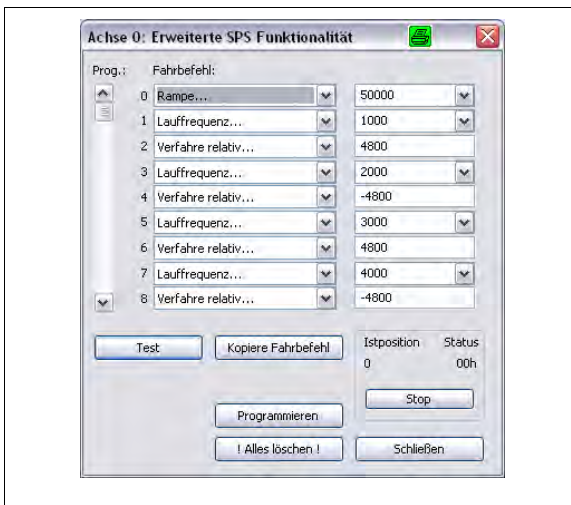


Abb. 9: Untermenü: Erweiterte SPS-Sequenzen

Manual IPCOMM

Test	Sequenz testen.
Kopiere Fahrbefehl	Die markierte Sequenz wird auf die folgenden Sequenzen kopiert.
Programmieren	Im Online-Modus werden die ausgewählten Fahrbefehle in die Steuerung gespeichert.
Werte übernehmen	Im Offline-Modus werden die eingestellten Fahrbefehle im Zwischenspeicher abgelegt und können später in der Parameterdatei gespeichert.
! Alles löschen !	Löschen aller Fahrbefehle.
Stop	Fahrbefehl abbrechen
Schließen	Das Untermenü verlassen.

5.3.7.3 Rampen >>

Mit **Rampen >>** startet das Untermenü zur Rampenprogrammierung.

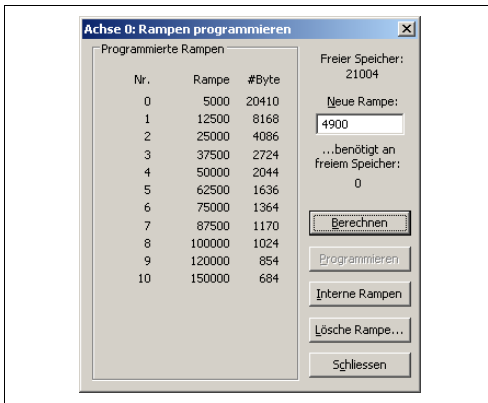


Abb. 10: Untermenü: Rampen programmieren

Im linken Abschnitt werden die bereits **programmierten Rampen (Nr.)** mit dem benötigtem **Speicher (#Byte)** angezeigt. Es sind 16 Rampen (0 bis 15) programmierbar.

Freier Speicher zeigt den noch freien Speicher für neue Rampen an.

In das Eingabefeld **Neue Rampe** wird der kleinstmögliche Rampenwert, der aus dem freien Speicherwert errechnet wird, geschrieben. Dieser Wert kann überschrieben werden.

In **....benötigt an freiem Speicher:** wird der für die Rampe berechnete Speicherwert ausgegeben.

Berechnen	Es wird der für den eingegebenen Rampenwert benötigte Speicherplatz berechnet.
Programmieren	Die Rampenwerte werden in der Steuerung gespeichert.
Interne Rampen	Geräteinterne Defaultwerten werden geladen.
Lösche Rampe...	Ein neues Fenster wird geöffnet: Durch Markieren mit <input checked="" type="checkbox"/> werden diese Rampen gelöscht und die übrigen Rampenwerte neu programmiert.
Schließen	Das Untermenü wird verlassen.

Manual IPCOMM

5.3.7.4 Aktuelle Parameter

Mit **Aktuelle Parameter** werden alle eingestellten Werte in die Steuerung gespeichert (ONLINE-Modus).

5.3.8 Optionale Zusatzkarte >>

Mit **Optionale Zusatzkarte >>** startet das Untermenü Optionale Zusatzkarte.

Hat die Steuerung mehr als 4 Eingänge und mehr als 2 Ausgänge, bietet dieses Untermenü die Möglichkeit, insgesamt 12 Eingänge und 6 Ausgänge durch Anzeige der Kontrolllampen zu überwachen.



Abb. 11: Untermenü: Zusatz-Eingänge und Ausgänge

5.3.9 RS-Monitor >>

Mit **RS Monitor >>** startet das Untermenü RS-Monitor, um Befehle direkt an die Steuerung zu senden.

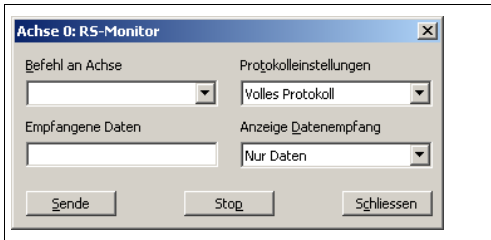


Abb. 12: Untermenü: RS-Monitor

In **Befehl an Achse** wird der Befehlscode geschrieben.

Folgende **Protokolleinstellungen** sind möglich:

Nur STX/ETX

STX/ETX und Prüfsumme

STX/ETX und Adresse

Volles Protokoll

In **Anzeige Datenempfang** wird das Datenempfangsprotokoll definiert:

Daten ohne Steuerzeichen

Kurzstatus und Daten

Alle Zeichen

Sende	Den Direktbefehl an die Steuerung schicken.
Stop	Motorlauf wird abgebrochen.
Schließen	Das Untermenü verlassen.

5.3.10 SFI >>

Falls die Steuerung eine Schrittfehlererkennung (SFI) durchführen kann, ist das Untermenü **SFI >>** freigeschalten. Hier können die Daten (Vorzugsdrehrichtung, Motorschrittzahl, Encoderstrichzahl) für die Schrittfehlererkennung (SFI) konfiguriert werden:



Abb. 13: Untermenü: SFI-Optionen

In **Status** kann SFI aktiviert oder deaktiviert werden.

Programmieren	Die eingestellten Werte werden in der Steuerung gespeichert.
Reset	SFI zurücksetzen
Schließen	Das Untermenü verlassen.

6 Fehlermeldungen

In diesem Anhang finden Sie Erläuterungen zu Fehlermeldungen von IPCOMM. Die Fehlermeldungen selbst sind durch fette und unterstrichene Schreibweise hervorgehoben. Nicht aufgeführt sind Fehlermeldungen, die selbsterklärend sind und daher keiner weiteren Erläuterung bedürfen.

Hinweis:

Fehlermeldungen im Zusammenhang mit Peripheriegeräten sind häufig auf nicht eingeschaltete Geräte, defekte Kabel und falsch verdrahtete oder nicht verbundene Anschlüsse zurückzuführen. Prüfen Sie daher immer, ob hier alles in Ordnung ist.

6.1 Schnittstellenfehler

Schnittstelle auf Timeout gelaufen

Mögliche Ursachen können in diesem Fall sein:

- Die Kabelkonfektion stimmt nicht (RD- und TD-Pin vertauscht).
- Die Steuerung ist nicht im Rechnerbetrieb (Schalter Remote/Local).
- Das Schnittstellenkabel ist defekt (Kabelbruch).
- Die Schnittstellenparameter von IPCOMM stimmen nicht mit denen der Steuerung überein.
- Keine Steuerung angeschlossen

COM-Schnittstelle already open

Mögliche Ursache kann in diesem Fall sein:

- Die COM-Schnittstelle wird schon von einem anderen Programm benützt.

Im Haupt-Eingabefenster wird keine COM-Schnittstelle angezeigt

Mögliche Ursache kann in diesem Fall sein:

- Keine seriellen Schnittstellen im Rechner vorhanden

Manual IPCOMM

7 Adapterkabel

Wichtig: Beim Anschluss vom PC zur Steuerung nur Kabel mit folgender Steckerbelegung (s.u.) verwenden! Zusatzbelegungen am Stecker können zu Übertragungsfehlern führen!

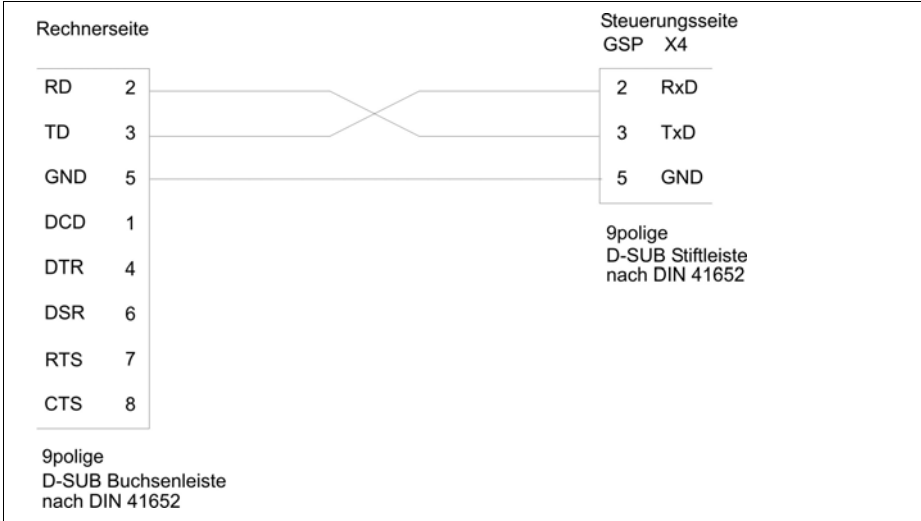


Abb. 14: PC (9polige Buchsenleiste) <-----> Steuerung GSP

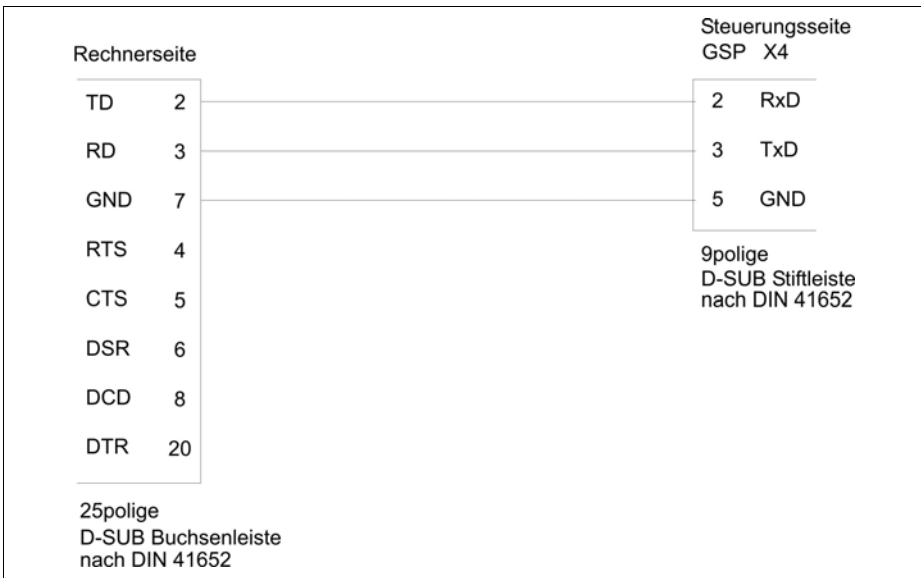


Abb. 15: PC (25polige Buchsenleiste) <-----> Steuerung GSP

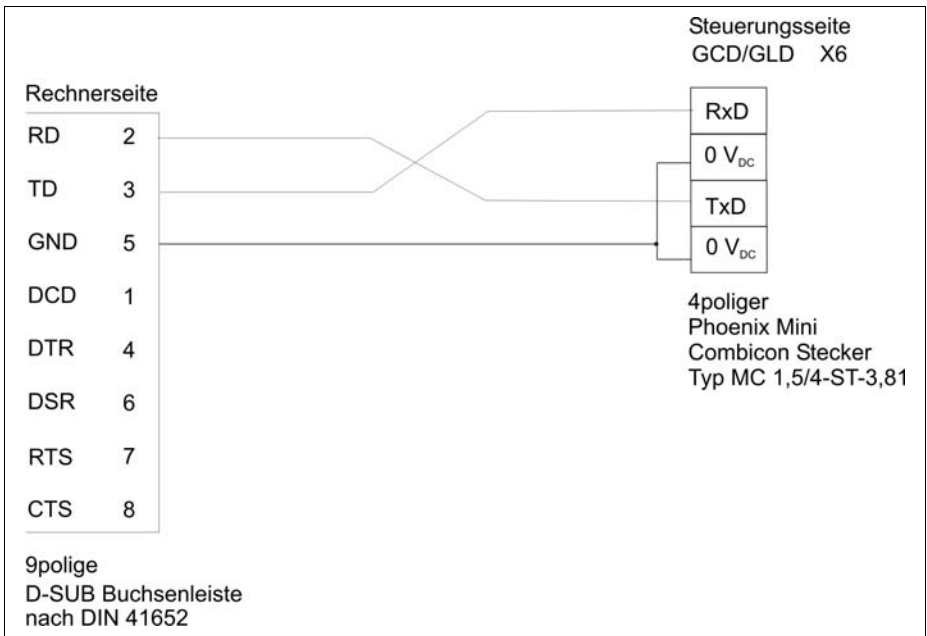


Abb. 16: PC (9polige Buchsenleiste) <-----> Steuerung GCD/GLD

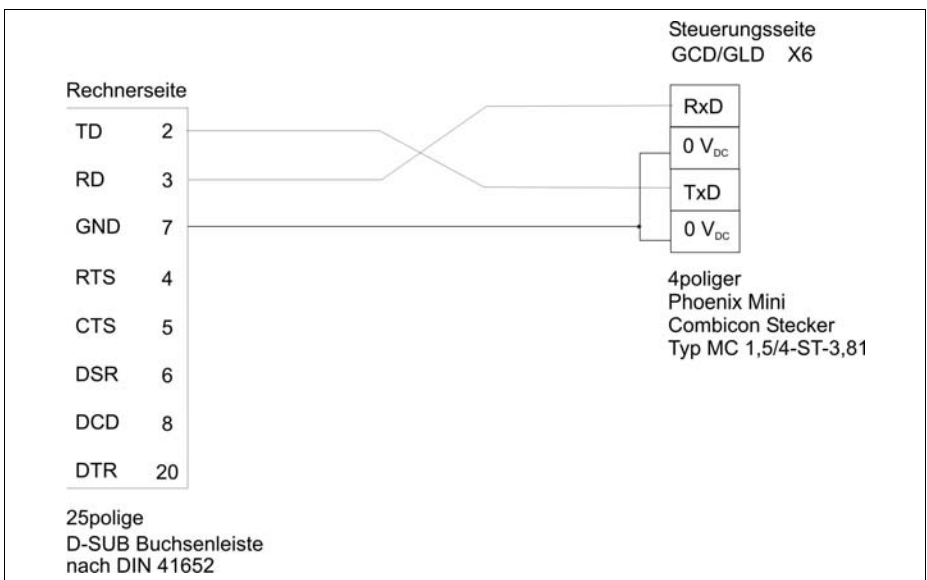


Abb. 17: PC (25polige Buchsenleiste) <-----> Steuerung GCD/GLD

8 Copyright und Haftungsausschluss

Das Softwareprogramm IPCOMM sowie die dazugehörige Dokumentation sind urheberrechtlich geschützt.

Von IPCOMM als Freeware Produkt ist es erlaubt Sicherungskopien für den persönlichen Gebrauch zu erstellen. Jedoch darf das Programm weder verändert noch verkauft werden.

Einschränkung der Gewährleistung

Das Softwareprogramm IPCOMM und das zugehörige Handbuch wurden mit größter Sorgfalt erstellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. trotzdem können Fehler existieren. Die Phytron-Elektronik GmbH kann weder die juristische Verantwortung noch irgendwelche Haftung für Schäden übernehmen, die durch die Benutzung von Informationen aus diesem Handbuch oder durch die Nutzung des in dieser Dokumentation beschriebenen Programms entstehen. Für die Mitteilung eventueller Fehler sind wir jederzeit dankbar.

Wir verweisen hier auf unsere Liefer- und Zahlungsbedingungen, insbesondere auf Punkt VII Haftung und Punkt IX Softwarenutzung.

Indem Sie das Softwareprodukt installieren, kopieren oder anderweitig verwenden, erklären Sie sich mit unseren Liefer- und Zahlungsbedingungen einverstanden. Falls Sie den Bestimmungen dieser AGBs nicht zustimmen, sind Sie nicht berechtigt, das Softwareprodukt zu installieren oder zu verwenden.

Geschützte Warenzeichen

Wir nehmen in diesem Handbuch auf geschützte Warenzeichen Bezug, die innerhalb des laufenden Textes nicht mehr explizit als solche gekennzeichnet sind. Aus dem Fehlen einer Kennzeichnung kann also nicht geschlossen werden, dass der entsprechende Produktname frei von Rechten Dritter ist:

- Microsoft ist ein eingetragenes Warenzeichen, und Windows ist eine Kennzeichnung der Microsoft Corporation in den Vereinigten Staaten und anderen Ländern.

9 Das IPCOMM-Protokoll

Dieser Abschnitt beschreibt das Protokoll, das zur Ansteuerung von **Phytron Schrittmotor Steuerungen** der Baugruppenserien IPP, GSP, GCD und GLD verwendet wird. Zur Vereinfachung werden die Steuerungen im Folgenden als **PSS** abgekürzt.

Beim IPCOMM-Protokoll handelt es sich um ein strenges Master-Slave Protokoll. Der Master, ein Leitreechner oder ein PC, sendet Befehlsdaten in Form von Telegrammen zu einem oder mehreren (maximal 16) Slaves, den PSS. Die Telegramme setzen sich aus einem Startzeichen, einem Adressbyte, mehreren Datenbytes, einer Prüfsumme und einem Stopzeichen zusammen.

Der oder die Slaves, eine oder mehrere PSS, antworten auf das Befehlstelegramm mit einem Antworttelegramm. Die Unterscheidung zwischen Sendeformat (Leitreechner an PSS) und Antwortformat (PSS an Leitreechner) ist notwendig, weil bei typischen Anwendungen regelmäßig Statusinformationen von der PSS zum Leitreechner zu übertragen sind, nicht aber vom Leitreechner zur PSS.

Bei der Definition des Protokoll wurde Wert darauf gelegt, dass der Telegramminhalt leicht lesbar ist. Alle Daten außer Start- und Stopzeichen sind lesbare ASCII-Zeichen. So können einfache Befehle auch per Hand mit einem einfachen Terminalprogramm eingegeben und zur PSS übertragen werden.

Die PSS antwortet auf jedes Telegramm, das ihre Adresse als erstes Zeichen enthält. Die Adresse entspricht der Steuerungsadresse, die mit dem Drehkodierschalter an der Vorderseite des Geräts (bei IPP und GSP) oder mit den Menüasten (bei GCD und GLD) einzustellen ist. Telegramme mit anderen Adressen werden ignoriert.

Bei Busbetrieb der Steuerungen (mehrere Achsen an einer RS485 Schnittstelle) muss die Adressierung der PSS eindeutig sein, jede Busadresse darf nur einmal vergeben werden. Der Leitreechner muss die Antwort der PSS bzw. eine gewisse Timeoutzeit abwarten, erst dann darf er das nächste Telegramm senden. So wird ein Buskonflikt durch mehrere gleichzeitige Antworten sicher vermieden.

Es ist möglich, Steuerungen unterschiedlicher Baugruppenserien am selben Bus zu betreiben.

Die vorzugsweise von IPCOMM verwendete Übertragungsgeschwindigkeit wurde darauf optimiert, Fahrbefehle möglichst schnell zur PSS zu übertragen. Deshalb wurde die Bitrate auf 28800 Baud¹ festgelegt (bei acht Bit je Byte, ein Stopbit, keine Parität).

Diese Bitrate wird von IPCOMM automatisch eingestellt. IPCOMM verwendet nicht den Wert, der von der Windows® Systemsteuerung vorgegeben wird.

Wenn Sie zur Ansteuerung einer PSS eigene Software schreiben wollen, muss auch eine Routine zur Einstellung der korrekten Bitrate implementiert werden².

¹ Optional kann die Bitrate per Software auf 9600 Baud eingestellt werden. Bei der Installation von IPCOMM wird auch das dazu erforderliche Hilfsprogramm BAUDRATE.EXE in der Phytron Programmgruppe installiert.

² Programmbeispiel vorhanden, siehe Kapitel 9.8.6.

Manual IPCOMM

9.1 Format des Sendetelegramms

Das Sendetelegramm vom Leitrechner zu einer PSS ist wie folgt definiert:

<STX> | Adresse | Daten | Separator | Prüfsumme | <ETX>

Beispiel: Übertragung des Fahrbefehls „Fahre Relativ 1000 Schritte“:

ASCII: <STX> | 1 | G | R | 1 | 0 | 0 | 0 | : | 1 | F | <ETX>

HEX: 02 | 31 | 47 | 52 | 31 | 30 | 30 | 30 | 3A | 31 | 46 | 03

Zeichen	Bedeutung
1	<STX> (Start of Text, 02 _H) als Kennzeichen für den Start eines neuen Telegramms. Da innerhalb eines Telegrammblocks dieser Code nicht verwendet wird, kann eine PSS damit auf den Telegrammbeginn synchronisieren.
2	Adresse der PSS mit den Werten „0“ bis „9“ und „A“ bis „F“ (30 _H ..39 _H bzw. 41 _H ..46 _H). Außerdem die Broadcast ³ Adresse @ (40 _H).
3 bis n-4	Nutzdaten, alle ASCII-Zeichen mit Codes größer/gleich 20 _H mit Ausnahme des als Separator benutzten Doppelpunktes : (3A _H). In der aktuellen Implementierung des Protokolls werden nur Großbuchstaben verwendet.
n-3	Separator : (3A _H) zur Trennung von Nutzdaten und Prüfsumme.
n-2	Höherwertiges Byte der Prüfsumme (Berechnung s.u.).
n-1	Niederwertiges Byte der Prüfsumme (Berechnung s.u.).
n	<ETX> (End of Text, 03 _H) als Telegrammende-Kennung.

Die Prüfsumme wird berechnet, indem – beginnend beim Adressbyte – alle Bytes einschließlich des Separators (:) mit einer Exklusiv-Oder-Verknüpfung (\oplus) aufsummiert werden.

$$CS = \text{Adresse} \oplus \text{Datenbyte1} \oplus \text{Datenbyte2} \dots \oplus \text{DatenbyteN} \oplus \text{Separator}$$

Die Prüfsumme CS wird als binärer Byte-Wert berechnet, das Ergebnis ist ein Byte im Wertebereich 00_H bis FF_H. Dieses Byte wird in zwei Hälften (Nibbles) zerlegt, jeweils mit dem Wertebereich 0_H bis F_H. Dann werden die den Nibbles entsprechenden lesbaren ASCII Zeichen ins Telegramm geschrieben, „0“ bis „9“ statt 0_H bis 9_H und „A“ bis „F“ für A_H bis F_H (rechnerisch wird auf den Nibble Wert 30_H bzw. 37_H addiert).

Die PSS berechnet beim Empfang eines Telegramms die Prüfsumme über die empfangenen Bytes und vergleicht sie mit der empfangenen Prüfsumme. Bei einer Abweichung wird das empfangene Telegramm verworfen und ein Fehlerbit im Status gesetzt.

Falls auf die Absicherung des Telegramminhaltes durch die Prüfsummenüberwachung kein Wert gelegt wird, kann diese auch ausgeschaltet werden. Die PSS akzeptiert auch Telegramme, bei denen statt der beiden Prüfsummenbytes zwei X gesendet werden, im Beispiel also <STX> | 1 | G | R | 1 | 0 | 0 | 0 | : | X | X | <ETX>.

³ Broadcast: Alle Achsen eines RS485 Bussystems empfangen das Telegramm und werten es aus. Da alle Achsen auch fast zeitgleich antworten würden und somit unweigerlich ein Buskonflikt entstünde, wird die Antwort der Steuerung bei Adressierung per „@“ unterdrückt, keine Achse antwortet.

9.2 Format des Antworttelegramms

Das Antworttelegramm von PSS zum PC ist wie folgt definiert:

<STX> | Adr. | Status | Sep. | Daten | Sep. | Prüfsumme | <ETX>

Beispiel: Antwort auf den Fahrbefehl „Fahre relative 1000 Schritte“, die PSS antwortet mit einer Statusinformation, die Antwort enthält keine Daten:

ASCII: <STX> | 1 | 0 | 1 | : | : | 0 | A | <ETX>

HEX: 02 | 31 | 30 | 31 | 3A | 3A | 30 | 41 | 03

Zeichen	Bedeutung
1	<STX> (Start of Text, 02 _H) als Startzeichen.
2	Adresse der PSS, die die Antwort sendet. Der Adressbereich ist „0“ bis „9“ und „A“ bis „F“ (30 _H bis 39 _H bzw. 41 _H bis 46 _H).
3 und 4	Kurzstatus, 8 Bit Information in ASCII codiert als 2 hexadezimale Ziffern.
5	Separator : (3A _H) zur Trennung von Status und optionalen Daten.
6 bis n-4	Optionale Nutzdaten, erlaubt sind alle Zeichen des ASCII-Zeichensatzes mit Codes größer/gleich 20 _H mit Ausnahme des als Separator benutzten Doppelpunktes : (3A _H). In der aktuellen Implementierung des Protokolls werden nur Großbuchstaben verwendet. Wenn im Antworttelegramm keine Nutzdaten enthalten sind, wird nur der Kurzstatus gefolgt von zwei Separatoren :: übertragen.
n-3	Separator : (3A _H) zur Trennung von Nutzdaten und Prüfsumme.
n-2	Höherwertiges Byte der Prüfsumme (Berechnung s.u.).
n-1	Niederwertiges Byte der Prüfsumme (Berechnung s.u.).
n	<ETX> (End of Text, 03 _H) als Telegrammende-Kennung.

Die Prüfsumme wird beim Antworttelegramm genauso wie beim Sendetelegramm berechnet. Beginnend mit dem Adressbyte werden alle Bytes einschließlich des zweiten Separators mit einer Exklusiv-Oder-Verknüpfung aufsummiert.

$$CS = \text{Adresse} \oplus \text{Statusbyte1} \oplus \text{Statusbyte2} \oplus \text{Separator} \oplus \text{Datenbyte1} \oplus \text{Datenbyte2} \dots \oplus \text{DatenbyteN} \oplus \text{Separator}.$$

9.3 Behandlung von Übertragungsfehlern

Die blockweise Übertragung von Daten als Telegramme wird im IPCOMM-Protokoll durch Auswerten der Prüfsumme für jeden Block abgesichert. Bei Auftreten einer Übertragungsstörung muss von der übergeordneten Steuerung, PC oder SPS, entschieden werden, ob ein als fehlerhaft erkannt Telegramm erneut zu senden oder zu verwerfen ist.

Die übergeordnete Steuerung sollte auch durch eine Timeout-Überwachung in der Lage sein, ausgefallene Antworttelegramme zu erkennen.

Fehlerursache	Symptom	Reaktion
Sendetelegramm vom Leit- rechner zur PSS wird ge- stört, Adresse wird aber erkannt.	PSS antwortet dem Statusbit „Protokollfehler“ oder „Befehl unbekannt“.	Sendetelegramm muss erneut gesendet werden.
Das Sendetelegramm wird bereits am Telegrammkopf (<STX> bzw. Adressbyte) gestört.	Die PSS reagiert nicht auf die übertragenen Daten.	Der Leitrechner muss den Fehler über eine Timeout-Überwachung erkennen und den Sendeversuch wieder- holen.
Das Empfangstelegramm wird schon im Telegramm- kopf gestört.	Die übergeordnete Steuerung erkennt den Telegrammkopf nicht.	Übergeordnete Steuerung muss den Fehler über eine Timeout-Überwachung er- kennen und gegebenenfalls den letzten Befehl wieder- holen. Achtung: Der Steuerungsstatus ist vorher auszulesen (z.B. bei Wiederholung einer Relativ- fahrt ist nicht sicher, ob die erste Fahrt ausgeführt wur- de. Deshalb muss der Positionszähler abgefragt werden).
Das Empfangstelegramm ist gestört.	Die übergeordnete Steuerung erkennt eine fehlerhafte Prüfsumme.	Der Leitrechner erkennt die falsche Prüfsumme und wie- derholt den letzten Befehl bzw. prüft den internen Status der PSS.

9.4 Statusinformation

Von der PPS wird als Antwort auf jeden Befehl eine Kurzstatusmeldung an den Leit-rechner zurückgeschickt (außer bei einem Broadcast-Telegramm). Auf eine explizite Statusabfrage kommt als Antwort ein Telegramm mit dem Kurzstatus vor dem ersten Separator und dem erweiterten Status nach dem ersten Separator zurück. Insgesamt sind im Gerätestatus 32 Bit codiert, die wichtigsten im Kurzstatus, alle anderen im Langstatus.

9.4.1 Kurzstatus

Der Kurzstatus enthält acht Bit Information und wird, als Zwei-Byte-Wert codiert, in zwei hexadezimalen Ziffern übertragen. Die Bedeutung der Bits ist wie folgt:

Bit / Wertigkeit	Bedeutung	Erklärung
7 / 128	Kaltstart	Das Bit wird bei Reset oder Einschalten der PSS gesetzt und durch eine Statusabfrage zurückgesetzt.
6 / 64	Any Error	Sammelmeldung für einen Fehler, der im Byte 3 des Gesamtstatus codiert ist.
5 / 32	RX Error	Sammelmeldung für einen Fehler, der im Byte 2 des Gesamtstatus codiert ist.
4 / 16	SFI Error	Schrittfehlersignal (nur bei IPP/GSP, nicht bei GCD/GLD), wird nur angezeigt, falls die optionale Zusatzkarte zur Schrittfehlerüberwachung bestückt ist.
3 / 8	Endstufe	Elektrischer Fehler der Endstufe (Kurzschluss, Über-spannung oder Übertemperatur).
2 / 4	Initiator –	Zustand Minus-Initiator
1 / 2	Initiator +	Zustand Plus-Initiator
0 / 1	Motor läuft	Motor läuft (einige Kommandos werden nicht akzeptiert).

9.4.2 Erweiterter Status

Mit der Abfrage des erweiterten Status bzw. Gesamtstatus (Kommando IS?) können interne Statusbits der PSS abgefragt werden. In der Antwort sind ein Byte Kurzstatus und drei Byte (codiert als sechs hexadezimale Ziffern) erweiterter Status enthalten. Mit dieser Abfrage werden Fehlerzustände zurückgesetzt, soweit dies möglich ist.

Statusbyte 2: Schnittstellenstatus

Bit / Wertigkeit	Bedeutung	Erklärung
7 / 128	Checksum	Beim Empfang des Telegramms wurde ein Prüfsummenfehler erkannt.
6 / 64		Zur Zeit nicht benutzt
5 / 32	Overrun	Der Empfangspuffer ist übergelaufen.
4 / 16	Not Now	Der Befehl wird momentan nicht akzeptiert, z.B. Frequenzänderung während des Motorlaufs.
3 / 8	Unknown	Die PSS kennt den Befehl nicht.
2 / 4	Bad Value	Der übertragene Wert ist nicht zulässig.
1 / 2	Parameter Limits	Der übertragene Wert liegt außerhalb der für diesen Befehl zulässigen Bereichsgrenzen.
0 / 1		Zur Zeit nicht benutzt

Statusbyte 3: Zusatzstatus

Bit / Wertigkeit	Bedeutung	Erklärung
7 / 128	No System	Der Systembereich der PSS ist nicht programmiert. Zur Systemprogrammierung liefert Phytron entsprechende Download-Utilities.
6 / 64	No Ramps	Es ist keine Rampentabelle programmiert worden. Um Rampen zu programmieren kann IPCOMM verwendet werden.
5 / 32	Parameter Changed	Ein Aktualparameter (z.B. Lauffrequenz) wurde geändert. Das Bit wird zurückgesetzt, wenn die Aktualparameter permanent gespeichert werden.
4 / 16	Just Busy	Die PSS ist intern beschäftigt (z.B. mit der Programmierung der vordefinierten Rampentabelle).
3 / 8	Programmierfehler	Bei der Programmierung des Flash-Speichers ist ein Fehler aufgetreten (Hardwarefehler).
2 / 4	Temperatur	Vorwarnung, die Endstufentemperatur ist erhöht. Bei einem Anstieg um weitere 10 °C schaltet die Endstufe ab.
1 / 2	Initiator-Fehler	Ein Initiatorfehler wurde erkannt und muss mit dem Befehl CI zurückgesetzt werden.
0 / 1	Interner Fehler	Bei der Berechnung der Fahrparameter ist ein Fehler aufgetreten.

Statusbyte 4: Zusatzinformation

Bit / Wertigkeit	Bedeutung	Erklärung
7 / 128	Treiber Fehler	Ein 24 Volt-Ausgang wurde kurzgeschlossen oder überlastet.
6 / 64		Zur Zeit nicht benutzt
5 / 32	Wait For SYNC	Bei einer Synchronstartsequenz wartet die PSS auf Fahrtfreigabe.
4 / 16	Lineare Achse	Die Bewegungsart LINEAR ist eingestellt.
3 / 8	Freilauf	Die Achse wird im freien Lauf bewegt.
2 / 4	Initialisierung erfolgreich	Die letzte Initialisierungsfahrt der Achse wurde erfolgreich beendet, der Referenzpunkt gefunden. Das Bit wird beim Start einer neuen Initialisierung zurückgesetzt.
1 / 2	HW Disable	Nur bei GSP: Der Eingang X5.6 HW Enable ist nicht mit 24 Volt beschaltet, die Verbindung zwischen Prozessor und Endstufe ist unterbrochen.
0 / 1	Initialisierung	Die Achse führt eine Initialisierungssequenz durch.

9.5 Befehlsdefinitionen

Im Sendetelegramm werden Befehle an die PSS übermittelt. Das erste Zeichen des Befehls teilt diese in Gruppen ein, die folgende Tabelle gibt eine kurze Übersicht.

Code	Kurzform für	Befehl	Siehe Kapitel
B	Break	Motorstop mit Nothaltrampe	9.5.1
H	Halt	Motorstop	9.5.1
G	Go	Fahrbehl ausführen	9.5.2
P	Parameter	Parameter ändern und abfragen	9.5.3
I	Info	I/O und Statusinformationen	9.5.4
C	Clear	Rücksetzbefehle	9.5.5
F	Flash	Speicher der PSS lesen	9.5.6
W	Write	Speicher der PSS schreiben	9.5.7
E	Erweiterung	Hardwareerweiterungen	0
R	Retry	Telegrammwiederholung anfordern	9.5.9
S	Sonder	Softwarespezifische Sonderbefehle	9.5.10

Das zweite Zeichen wird als Befehlskennung innerhalb der Gruppe verwendet, ab dem dritten Zeichen folgen optionale Parameter. Um einen eingestellten Parameter abzufragen, muss als drittes Zeichen ein „?“ gesendet werden. Um einen Parameter zu ändern, muss der neue Wert im Befehl angegeben werden.

Der Wertebereich für Positionsangaben beträgt 32 Bit binär (Long Integer Format) und wird im Telegramm stets in dezimaler Darstellung mit optionalem Vorzeichen angegeben (maximal also z.B. „GA-2147483647“).

Reine Richtungsangaben, z.B. die Richtung von Initialisierungsfahrten, freiem Lauf und Einzelschrittbetrieb, werden durch das Zeichen „+“ bzw. „-“ festgelegt (z.B. „GF+“).

Parameter, die einen binären Code enthalten, werden als hexadezimal codierte Zahlen übertragen, z.B. die Stufe des einzustellenden Stromes.

9.5.1 Stop-Befehle „H“ und „B“

Im Online Modus gibt es zwei Möglichkeiten, eine Positionierung oder einen freien Lauf zu beenden, den Befehl „H“ (wie Halt) und den Befehl „B“ (wie Break).

Der Stop-Befehl „H“ bewirkt einen Fahrtabbruch mit der eingestellten Rampe (mit dem Befehl „PN“), der Stop-Befehl „B“ bewirkt einen Fahrtabbruch mit der Nothaltrampe. Die Nothaltrampe ist ein Vielfaches der eingestellten Rampe, der Faktor wird mit dem Befehl „PH“ eingestellt. Der Fahrtabbruch mit Nothaltrampe wird mit fester Schrittweite durchgeführt, die sonst verwendete Schrittweitenanpassung findet hierbei nicht statt.

Der Fahrtabbruch erfolgt kontrolliert, das heißt der Positionszähler hat weiterhin Gültigkeit.

9.5.2 Fahrbefehle „G“

Das Zeichen nach dem „G“ legt den Typ des Fahrbefehls fest.

Code	Befehl
GA	Absolutfahrt
GR	Relativfahrt
GF	Freier Lauf
GI	Initialisierungsfahrt
GS	Einzelschritt
GW	Beginn der Synchronstartsequenz, siehe folgendes Beispiel
GX	Ausführen der Synchronstartsequenz, Start der Achse, siehe folgendes Beispiel
GB	Abbruch einer begonnenen Synchronstartsequenz

Beispiele für Befehlssequenzen

In den folgenden Beispielen werden nur Nutzdaten und Adresse im Sendetelegramm angegeben, Separator, Prüfsumme und Delimiter <STX> / <ETX> sind entsprechend zu ergänzen:

Absolutfahrt auf Position 1000	„1GA1000“
Relativfahrt um –200 Achtelschritte	„1GR–200“
Initialisierung am positiven Initiator	„1GI+“
Initialisierung am negativen Initiator	„2GI–“
Ein Achtelschritt in positiver Richtung	„1GS+“ oder „1GR+1“

Synchronstartsequenz:

Vorbereiten Achse 1	„1GW“
Positionsangabe für Achse 1	„1GR1000“
Vorbereiten Achse 2	„2GW“
Positionsangabe Achse 2	„2GR800“
...und synchrones Starten	„@GX“

Hinweis:

Das Zeichen „@“ wird als Broadcast-Adresse verwendet. Das heißt, alle Achsen empfangen dieses Telegramm und werten es aus, aber nur die Achsen, die vorher per GW/GR freigeschaltet wurden, führen eine Bewegung aus.

Da alle Achsen fast zeitgleich antworten würden und somit unweigerlich ein Buskonflikt entstünde, wird die Antwort der PSS bei Adressierung per „@“ unterdrückt.

Manual IPCOMM

9.5.3 Parameterbefehle P

Alle Parameter werden durch den Wert im Befehl neu gesetzt oder mit „?“ausgelesen.

Code	Parameter	Wertebereich	Default	Beispiel
PA	Booststrom (A cceleration C urrent)	$0^4 \dots F_H^5$ (PAx.x GCD / GLD)	0	PAC (PA6.1)
PB	BIOS-Parameter laden	Kein Parameter erforderlich		
PC	Istposition (P osition C ount)	32 bit	0	PC?
PD	Betriebsart (D igital M ode)	0 = Online Mode 1 = SPS Mode	0	PD0
PE ⁶	IPCOMM intern (Hardwaredaten)			
PF	Lauffrequenz	1 .. 10000 (f_{max})	2000	PF4700
PG	Achsenbegrenzung, wirksam bei freiem Lauf und Initialisierung, Fahrtabbruch bei Überschreitung	32 bit	1000000	PG?
PH	Nothaltrampe	0 .. 250 * PN	0	PH5
PI	SFI (Option nur für GSP und IPP) aktivieren/deaktivieren	0 = aus 1 = ein	0	PI?
PL	Bewegungsart (L inear A xis)	0 = rotatorisch 1 = linear	0	PL1
PM	Offset vom M inus Initiator	0 .. 40000	0	PM1600
PN	Rampennummer	0 .. R_{max}	0	PN2
PO	Offset- oder Start/Stop Frequenz	0 .. 1250 ($f_{max}/8$)	400	PO350
PP	Offset vom P lus Initiator	0 .. 40000	0	PP1200
PR	Laufstrom (R un C urrent)	$1 \dots F_H^5$ (PRx.x GCD / GLD)	4	PR7 (PR3.9)
PS	Stopstrom	$0 \dots F_H^5$ (PSx.x GCD / GLD)	2	PS2 (PS0.1)
PT	Laufstromüberhöhungszeit in ms	0 .. 4000	20	PT25
PU ⁶	IPCOMM intern (Benutzerdaten)			
PX ⁶	IPCOMM intern (Rampendaten)			
PW	Spielausgleich (W echselspiel)	-30000 .. 30000	0	PW0

⁴ Null deaktiviert den Boost-Mode, auch während der Rampen wird der Laufstrom eingestellt.

⁵ Die Phasenströme aller Steuerungen werden in 16 Stufen eingestellt. Die Stufen entsprechen unterschiedlichen Motorströmen, deren Werte den Geräteunterlagen zu entnehmen sind. Steuerungen vom Typ GCD und GLD können außerdem in der Form „PRx.x“ eingestellt werden, die Stromvorgabe kann hier in Stufen von 0.1 A erfolgen. Die zu IPCOMM kompatible Stromvorgabe wird aber genauso unterstützt. Abfrage des eingestellten Wertes in der Form „PR?“ liefert die Stromstufe, die Abfrage „PR??“ liefert den exakten Wert.

⁶ IPCOMM verwendet die Befehle „PE“, „PU“ und „PX“, um in der Steuerung Informationen über die zur Programmierung verwendeten DLLs zu hinterlegen.

Die Parameter existieren in drei Kopien in der PSS. Einmal im Programmspeicher (BIOS-Default im Flash-EPROM), einmal im permanenten Datenspeicher (serielles EPROM) und einmal im Arbeitsspeicher (flüchtiges RAM). In der Tabelle sind die Wertebereiche und die Standardwerte, die nach Reset aller Parameter bzw. nach Auslieferung voreingestellt sind, angegeben.

Die Daten aus dem permanenten Speicher werden beim Einschalten oder Reset der PSS in den Arbeitsspeicher kopiert. Alle Parameter sind bei Auslieferung in einem definierten Zustand, dem sog. BIOS-Default.

Während des Betriebs betreffen Änderungen der Parameter nur die Aktualparameter im flüchtigen RAM. Diese Parameter haben nur Gültigkeit bis zu einem Reset der PSS. Um Änderungen permanent zu machen, müssen die Parameter mit dem Befehl WP gespeichert werden.

Parameteränderungen sind nur zulässig, wenn die Achse steht und nach dem letzten Fahrbefehl die Laufstromüberhöhungszeit abgelaufen ist (die PSS meldet dann Stillstand in der Statusinformation).

9.5.4 I/O-, Status- und Infobefehle „I“

Diese Befehlsgruppe liefert allgemeine Informationen z.B. über interne Zustände der PSS. Das Zeichen nach dem „I“ wählt eine der Unterfunktionen aus. Alle Befehle außer den Befehlen „IO“ und „IZ“ sind reine Abfragen.

Code	Parameter
IA?	Adresse des nächsten freien Speicherplatzes im Rampenspeicher ermitteln
IB?	Versionskennung des BIOS auslesen
IC?	Datumscode der Systemsoftware
ID?	Datumscode der BIOS-Software
IE?	Nummer der Erweiterungskarte
IF?	Maximal einstellbare Frequenz f_{\max} abfragen
II?	Eingangsabfrage (Inputs), liefert den Wert der Standardeingänge
IN?	Anzahl der programmierten Rampen
IQ?	Prozessortaktfrequenz (Quarz)
IS?	Statusabfrage
IV?	Versionskennung der Systemsoftware auslesen
IO	Abfragen oder Setzen der Standardausgänge (Outputs)
IZ	Abfragen oder Setzen der Nullungsdifferenz (Delta Zero)

Anmerkung: Die Befehle „II“ und „IO“ beziehen sich nur auf die Standard-I/O, die schon im ersten Steuerungstyp, der IPP ohne Hardwareerweiterung, integriert waren. Alle anderen Ein- und Ausgänge werden über die erweiterten Befehle „E“ bedient (siehe Kap.9.5.8).

9.5.5 Rücksetzbefehle „C“

Mit dem Clearbefehl „C“ können einzelne Fehlerstatusinformationen zurückgesetzt werden. Außerdem kann per Software ein Hardwarereset der PSS durchgeführt werden. Der Buchstabe nach dem „C“ entscheidet, was zurückgesetzt werden soll.

Code	Parameter
CA	Endstufenreset (A mplifier) setzt das Fehlerbit zurück und veranlasst einen Hardwarereset der Endstufe.
CI	Ein Initiatorfehler muss auf jeden Fall durch CI bestätigt werden.
CO	Ausgangstreiberreset (24 V _{DC} digital O utput) setzt das Fehlerbit zurück und gibt ein Resetsignal an die Treiber (nicht bei IPP).
CR	Veranlasst einen Hardwarereset der gesamten PSS.
CS	Veranlasst einen Reset der optionalen Schrittfehlerüberwachung (SFI-Option nur bei IPP oder GSP verfügbar, nicht bei GCD und GLD).

Anmerkungen

Zu "Endstufenfehler":

Ein Endstufenfehler kann mehrere Ursachen haben: ein kurzzeitiger Spannungseinbruch, ein Anstieg der Zwischenkreisspannung oder ein Kurzschluss. Falls das Fehlerbit nach einem Reset immer noch gesetzt ist, liegt wahrscheinlich ein Kurzschluss vor.

Zu "Initiatorfehler":

Ein Initiatorfehler kann mehrere Ursachen haben:

- Beide Initiatoren haben gleichzeitig angesprochen (z.B. wenn die Versorgung der Initiatorschalter nach der Hauptspannung der PSS eingeschaltet wurde oder ein Initiator nicht beschaltet ist).
- Beim Positionieren in linearer Betriebsart hat ein Initiator angesprochen (in dieser Betriebsart werden die Initiatoren gleichzeitig als Endlagenschalter betrachtet).
- Während der Referenzpunktsuche im linearen Betrieb hat der falsche Initiator angesprochen.
- Der Initiator ist überfahren worden, d.h. nach Stillstand der Achse liegt das Signal nicht mehr an (die Initiatoren von Linearachsen sollten bis zum Ende des Verfahrwegs bedämpft sein).
- Der Initiator ist innerhalb der Streckenbegrenzung nicht gefunden worden.

Zu "Ausgangstreiberfehler":

Das Bit Ausgangstreiberfehler zeigt an, dass ein Kurzschluss oder eine Überhitzung den Ausgangstreiber zum Abschalten gebracht hat. Im Normalfall kann der Fehler per Hardwarereset zurückgesetzt werden, manchmal ist auch ein Aus- und Wiedereinschalten der 24 Volt Hilfsspannung nötig. Wenn die Fehlermeldung dann immer noch ansteht, liegt entweder der Kurzschluss noch an oder der Treiber wurde dauerhaft geschädigt.

9.5.6 Flash-Programmierbefehle F

Die Befehle zum Programmieren des geräteinternen Flash-Speichers setzen genaue Kenntnisse der Gerätehardware voraus. Hier wird davon ausgegangen, dass die Softwaremodule im Flash-EPROM der PSS nur mit dem Konfigurationsprogramm IPCOMM programmiert werden. Die hier aufgeführten Befehle sind somit nur der Vollständigkeit halber erläutert, genauso die Befehle des folgenden Kapitels.

Code	Parameter
FA	Stromtabelle ⁷ (A mplifier)
FC	Flash freischalten (C lear)
FR	R ampentabelle lesen
FU	SPS-Benutzerbereich (U ser) lesen im Format FUx, die Programmnummer x muss als ein hexadezimaleres Zeichen (0...F _H) angegeben werden
FF	Blockweises Lesen des seriellen Eproms
FQ	Blockweises Lesen des Flash-Speichers

9.5.7 Schreibbefehle W

Lesen Sie hierzu bitte die Erläuterungen zu den Befehlen im Kapitel 9.5.6.

Code	Parameter
WA	Die bei Auslieferung voreingestellte Rampentabelle restaurieren
WB	Programmierung abbrechen
WD	Löschen der Rampentabellen und des SPS-Benutzerbereiches
WE	Löschen des SPS-Benutzerbereiches
WF	Blockweises Schreiben des seriellen EPROMs
WP	Aktualparameter vom RAM ins serielle EPROM schreiben
WQ	Blockweises Schreiben des SPS-Benutzerbereiches im Flash
WR	Rampentabelle blockweise beschreiben
WS	Systembereich blockweise beschreiben
WU	Schreiben des SPS-Benutzerbereichs im Format WUxSTRING mit der Programmnummer x (ein hexadezimaleres Zeichen, 0...F _H) und dem SPS Befehlsstring STRING.
WX	Programmierung abschließen

⁷ Die Stromtabelle kann nur ausgelesen werden (mit FA?). Als Antwort kommt ein String mit 16 hexadezimal codierten Werten zurück, ein Wert für jede der 16 einstellbaren Stromstufen (0 .. F_H). Jeder Wert entspricht dem 10-fachen des eingestellten Effektivstromes in der jeweiligen Stufe, z.B. der letzte Wert einer GSP 92-70 wird übertragen als 3F_H = 63_D, der maximale effektive Motorstrom ist also 6,3 Ampere.

Manual IPCOMM

9.5.8 Erweiterte Befehle E

Das IPCOMM-Protokoll zum Datenaustausch zwischen der Schrittmotorsteuerung und der Inbetriebnahmesoftware IPCOMM wurde parallel zu der Steuerung IPP und IPCOMM entwickelt. Die IPP hat vier Eingänge und zwei Ausgänge, als Option gibt es eine I/O-Karte, die zusätzlich noch acht Eingänge und vier Ausgänge besitzt. Die E-Befehle basieren auf dieser Hardwareerweiterung. Die Eingänge werden als Programmauswahleingänge bezeichnet.

Bei den später entwickelten Steuerungen GSP und GCD wurde Wert darauf gelegt, dass jede Steuerung zu IPCOMM kompatibel ist, daher wurde die Befehlsgruppe unverändert beibehalten. Die GSP hat zwölf Eingänge und sechs Ausgänge, die GCD hat zehn Eingänge und vier Ausgänge. Die Ausgänge OUT0 und OUT1 entsprechen den zwei Ausgängen der IPP-Grundversion, die Ausgänge ab OUT2 den Zusatzausgängen.

Code	Parameter
EA ⁸	Abfragen und Setzen aller vier Ausgänge des GCD/GLD. EA? liefert ein hexadezimaleres Zeichen als Antwort, EAx setzt die Ausgänge neu (x = 0 .. F _H).
EI	Abfrage der Programmauswahleingänge Prog.0 .. Prog.n-1. EI? liefert 8 Bit binäre Daten, hexadezimal codiert als zwei Bytes. GCD, GLD, GSP und IPP mit Erweiterungskarte haben n = 8 Eingänge, GCD und GLD mit T-Option haben n = 4 Eingänge.
EO	Abfragen und Setzen der vier Zusatzausgänge (bzw. zwei beim GCD/GLD ⁹). EO? liefert ein hexadezimaleres Zeichen als Antwort, EOx setzt die Ausgänge neu (x = 0 .. F _H bei GSP/IPP bzw 0 .. 3 bei GCD/GLD). Die beiden Standardausgänge werden mit dem Befehl IO bedient.
ER	Lesen der SPS-Benutzersequenzen in der Form ERxx. 256 Benutzersequenzen sind möglich (bei der IPP in Grundversion nur 7), diese werden hexadezimal adressiert (zwei hexadezimale Zeichen xx = 00...FF _H).
EW	Schreiben der SPS-Sequenzen im Format EWxxSTRING mit der Programmnummer xx (zwei hexadezimale Zeichen 00...FF _H) und dem SPS Befehlsstring STRING.

9.5.9 Retrybefehl R

Mit der Retry-Anforderung R wird die PSS aufgefordert, die letzte Nachricht an den Master zu wiederholen.

⁸ Nur bei GCD und GLD.

⁹ Nicht bei eingebautem T-Modul, bei dieser Option sind die Ausgänge für den DEGAS-Vorgang reserviert.

9.5.10 Sonderbefehle S

Die Befehlsgruppe S wird für Erweiterungen verwendet, die nicht alle Steuerungen betreffen. Die folgenden Befehle gelten nur für GCD und GLD und betreffen vor allem die Bedienung des T-Moduls (optionales Temperaturmessmodul).

Code	Parameter
SA?	Abfrage der Temperatur der Leistungsendstufe (A mplifier)
SC?	Abfrage des Motorstromes, Augenblickswert wird angezeigt
SD	Abfragen oder Setzen der Nullungsdifferenz (D elta Zero)
SU?	Abfrage der Spannung des Gleichstromzwischenkreises
SW?	Abfrage des angeschlossenen Thermoelements. Die Antwort lautet „PT“, „KT“ oder „NO“.
SH ¹⁰	Start eines Ausheizzyklus
SM? ¹⁰	Abfrage der Motortemperatur, ein Messfühler muss im Motor angebracht sein
SR? ¹⁰	Abfrage der Referenztemperatur. Diese wird im Gerät in der Nähe der Eingangsklemmen gemessen.
ST ¹⁰	Abfrage und Vorgabe der Ausheizdauer in Minuten
SI? ¹¹	Abfrage der internen Referenzspannung
SV? ¹¹	Abfrage der Referenzspannung des T-Moduls
SX ¹¹	Abfrage und Vorgabe der Temperatur 1
SY ¹¹	Abfrage und Vorgabe der Temperatur 2
SZ ¹¹	Abfrage und Vorgabe der Temperatur 3

¹⁰ Nur bei eingebautem T-Modul

¹¹ Nur bei eingebautem T-Modul und angeschlossenen PT100 Messfühler

9.6 Online-Betrieb einer PSS

Unter Online-Betrieb (PD0) verstehen wir, dass die Phytron Schrittmotor-Steuerung (PSS) ihre Fahrbefehle von einem Leitrechner über die serielle Schnittstelle erhält.

Wenn Sie eine PSS mit eigener Software betreiben wollen, müssen Sie nicht alle Befehle des IPCOMM Protokolls implementieren. Für eine einfache Applikation kann sich die Ansteuerungssoftware auf wenige Befehle beschränken: Die Grundkonfiguration der Achse kann mit IPCOMM durchgeführt werden, die Achsparameter werden von der PSS gespeichert und haben dann auch nach einem Reset Gültigkeit.

Zum Fahren einer Achse benötigen Sie nur wenige Befehle:

IS?	Abfrage des Achsenstatus
GI+ und GI-	Fahrt zum Endschalter (Initialisierung)
GAXxxxx	Fahrt auf eine Absolutposition
GRxxxxx	Fahrt relativ um xxxxx Achtelschritte
PNx	Setze die Nummer der zu verwendenden Rampe
PFxxxxx	Setze Lauffrequenz
POxxxx	Setze Offsetfrequenz
CI	Lösche Fehlerbedingung Endschalter
CA	Lösche Fehlerbedingung Endstufenfehler

9.7 SPS-Betrieb einer PSS

Unter SPS-Betrieb (PD1) verstehen wir, dass die PSS ihre Fahrbefehle von einem Leitrechner über die digitalen Eingänge erhält. Der Handshake mit dem Leitrechner erfolgt in dieser Betriebsart über die digitalen Ausgänge.

Für einen gesicherten Datenaustausch muss ein eindeutiges Protokoll eingehalten werden. Die SPS muss auf das READY-Signal der PSS warten. Dann kann sie die Programminformation anlegen, die anschließend durch ein STROBE-Signal aktiviert wird.

Die PSS übernimmt die Programminformation und bearbeitet den Befehl. Während der Programmausführung wird READY passiv.

Die digitalen Eingänge haben bei SPS-Betrieb dann folgende Bedeutung:

Input 0 STROBE: Aktiviert das eingestellte Programm.

Input 1 SW-ENABLE¹²: Muss aktiv sein, damit die Achse freigegeben ist und ein Achsstart ausgeführt werden kann.

Durch Wegnehmen des Signals kann eine laufende Positionierung abgebrochen werden. Der Motor wird mit Rampe gestoppt, der Positionszähler enthält einen gültigen Wert.

Input 2 CLEAR ERROR¹²: Durch Beschalten des Eingangs und anschließendes STROBE-Signal wird ein Fehlerstatus der PSS rückgesetzt (z.B. Initiatorfehler,

¹² Nicht bei GCD/GLD und nicht bei der IPP in Grundausstattung. Bei dieser Steuerung sind insgesamt nur vier Eingänge vorhanden, der erste hat die Funktion des STROBE, die drei anderen dienen zur Auswahl des Programms (eins aus acht).

Endstufenfehler). Gleichzeitig wird auch der Fehlerausgang der PSS zurückgesetzt.

Der Eingang muss durch ein STROBE Signal aktiviert werden, er hat Priorität gegenüber den Programmwahleingängen.

Input 3 RESET¹²:-Der Eingang entspricht in der Funktion einem Resetaster. Durch Beschalten des Eingangs und anschließendes STROBE-Signal führt die PSS einen Reset aus.

Der Reseteingang hat Priorität gegenüber allen anderen Eingängen, muss aber auch mittels STROBE aktiviert werden.

Prog x Mit den Programmwahleingängen wird eins von 256 Benutzerprogrammen¹³ aus der SPS-Tabelle ausgewählt. Die Eingänge werden binär interpretiert. Wenn die Signale an den Auswahleingängen stabil anliegen, kann das selektierte Programm durch Aktivieren des STROBE-Eingangs gestartet werden.

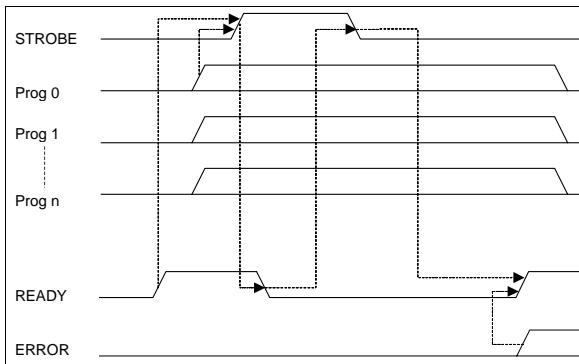
Die Ausgänge haben bei SPS-Betrieb folgende Bedeutung:

Output 0 READY: Die Steuerung ist bereit zur Ausführung des nächsten Fahrbefehls. Das Signal wird bei Motorstart weggenommen und kommt erst wieder, wenn der Motor steht UND wenn die Laufstromüberhöhungszeit abgelaufen ist UND wenn das STROBE-Signal nicht mehr anliegt.

Output 1 ERROR: Es liegt eine Fehlerbedingung vor, folgende Ursachen sind möglich:

- Das gewählte Programm ist nicht programmiert
- Ein Initiatorfehler ist aufgetreten
- Ein Endstufenfehler liegt vor
- Ein Schrittfehler ist erkannt worden (nur IPP/GSP mit Option SFI)

Der digitale Datenaustausch zwischen SPS und PSS kann grafisch so dargestellt werden:



¹³ IPP (Grundausstattung): 8, bei GCD oder GLD mit installiertem T-Modul: 16 mögliche Benutzerprogramme.

Manual IPCOMM

9.8 Zusatzinformationen

Hier finden Sie Zusatzinformationen zum IPCOMM Protokoll

9.8.1 Tools

IPS2000: Das Programm kann zur Inbetriebnahme der seriellen Verbindung zwischen PC und PSS verwendet werden.

Kurzbeschreibung der drei Befehle:

- **Suche!**
Hier erfolgt nacheinander eine Versionsabfrage über die seriellen Schnittstellen COM1 bis COM4 auf den Geräte-Adressen 0 bis F und zwar jeweils mit der Baudrate 9600 und 28800. Wird ein Gerät gefunden, so erfolgt die Ausgabe der Adresse, der Baudrate und der Geräteversion.
- **Blinke!**
Mit dieser Option können Sie feststellen, ob die Leitung zum Gerät in Ordnung und richtig verdrahtet ist. Auf den seriellen Schnittstellen Ihrer Wahl werden im Abstand von ca. 250ms mit der von Ihnen gewählten Baudrate 9600 oder 28800 alternierend die Zeichen <STX> und <ETX> gesendet. Hierdurch blinkt beim IPP z.B. die LED "Status".
Ist der Befehl "Blinke!" erfolgreich, jedoch der Befehl "Suche!" findet kein IPP, so ist wahrscheinlich der Interrupt der betreffenden Schnittstelle falsch eingestellt.
- **LoopBack!**
Mit dieser Option können Sie feststellen, ob die Schnittstelle Ihres PCs richtig konfiguriert ist. Hierzu überbrücken Sie provisorisch die Pins 2 und 3 am Stecker, z.B. mit einem Testadapter oder einer Drahtbrücke. Auf der Schnittstelle Ihrer Wahl wird dann mit der ausgewählten Baudrate permanent das Zeichen 'U' (55h) ausgegeben. Wenn alles ok ist, muss die Anzahl der empfangenen Zeichen mit der Anzahl der gesendeten Zeichen übereinstimmen.

OPM: Ist ein kleines Tool, das unabhängig von IPCOMM zur Archivierung von Projektdaten verwendet werden kann. Alle Steuerungsparameter und SPS-Sequenzen werden von der PSS gelesen und in einer Textdatei gespeichert. Die Parameter werden im Format des IPCOMM-Protokolls zeilenweise in die Textdatei geschrieben und können mit einem Texteditor bearbeitet werden. Die Parameterdatei kann per OPM wieder zur PSS übertragen werden. (Siehe auch Manual OPM)

9.8.2 Softwarestruktur

Die Software der PSS ist in C programmiert. Sie ist in zwei Teile aufgeteilt, "BIOS" und "System". Im BIOS sind die Kommunikationsroutinen, Nachladeroutinen, Bibliotheks-routinen und Konstanten hinterlegt, im System die eigentlichen Motorsteuerungsroutinen.

Die beiden Softwareteile sind in drei Blöcken des permanenten Steuerungsspeichers untergebracht (blockweise löschbares Flash-EPROM mit acht Blöcken).

In einem vierten Block werden die SPS-Sequenzen gespeichert, also benutzerdefinierte Fahrbefehle, die einmal gespeichert werden und dann im SPS Modus durch Beschalten der digitalen Eingänge aktiviert werden können.

Die restlichen vier Blöcke des Flash Speichers enthalten die Rampendaten. Das sind Tabellen für die Erzeugung des Motortaktes während Beschleunigungsphasen.

Diese Aufteilung des Speicherbereiches erlaubt es, Teile des Speichers im Betrieb der Steuerung vor Ort neu zu programmieren. Typischerweise wird diese Neuprogrammierung während der Inbetriebnahme der Steuerung vorgenommen. Zur Programmierung neuer Rampen oder SPS Sequenzen kann die zusammen mit der Steuerung ausgelieferte Inbetriebnahmesoftware IPCOMM verwendet werden.

Manchmal kann es auch erforderlich sein, das Betriebssystem der PSS zu aktualisieren oder durch eine anwendungsspezifische Variante zu ersetzen. Phytron liefert zu diesem Zweck entsprechende Downloadsoftware und Daten. Die Software wurde nach bestem Wissen und mit Sorgfalt erstellt. Bei neuen Programmversionen wegen Änderungen und Erweiterungen in der Hardware werden dann selbstverständlich auch bis dahin bekanntgewordene Fehlfunktionen entfernt (Bugfixes).

Diesem Handbuch liegen die Version 1.11 der GCD/GLD-Software und die Version 1.55 der GSP/IPP-Software zugrunde.

Die GCD/GLD-Version berücksichtigt die Erweiterungen, die mit dem optionalen Temperaturmodul verbunden sind.

In der GSP/IPP-Version ist als neues Feature die Nothaltrampe implementiert.

Alle Versionen haben eine gegenüber den Vorgängerversionen erweiterte SPS-Funktionalität.

9.8.3 SPS-Sequenzen konfigurieren

Die Standard SPS-Befehle sind ausschließlich Fahrbefehle. In eine SPS-Zeile werden die Start-Stop-Frequenz, Lauffrequenz, Rampe und die Art der Bewegung eingetragen.

Schon während der Programmierung werden die mit den Befehlen übertragenen Parameter in steuerungsinterne Parameter umgerechnet. Beim Start einer dieser SPS-Sequenzen werden die Parameter aus dem Speicher geladen. Bis zum Achsstart sind nur noch wenige Parameter zu berechnen, der Start erfolgt mit wenigen Mikrosekunden Verzögerung.

Wir empfehlen, diese Programmierung der SPS-Befehle mit IPCOMM oder OPM durchzuführen. Wenn es für Sie notwendig ist, die SPS-Befehle mit eigener Software zu ändern, so ist als Beispiel hier eine Kommandosequenz abgedruckt, mit der die SPS-Befehle geändert werden können.

Folgende Befehle werden hierfür benötigt:

- | | | |
|------|-----------------------------------|--------------------------|
| 1. | Abfrage des Freischaltcodes | Befehl IC? |
| 2. | Freigabe der Flash-Programmierung | Befehl FC |
| (3.) | Inhalt des Flash-Eproms löschen | Befehl WE |
| 4. | Neuen Code laden | Befehl EW siehe Beispiel |
| 5. | Programmierzklus abschließen | Befehl WX |

Bedingt durch die Blockstruktur des Flash-Eproms wird mit dem Befehl WE der gesamte Block gelöscht, der zur Speicherung der SPS-Sequenzen verwendet wird. Also werden alle bisher programmierten Sequenzen gelöscht. Sollen nur zusätzliche, bisher nicht verwendete Programmplätze programmiert werden, ist das Löschen nicht erforderlich.

In jedem EW Befehl müssen die spezifischen Daten eines SPS-Befehls enthalten sein. Die Information ist auf die gleiche Weise kodiert wie bei Online-Befehlen.

Die benötigten Daten sind:

1. die Nummer des SPS-Programms (Speicherplatz),
2. die Start/Stop-Frequenz,
3. die Lauffrequenz,
4. der Fahrbefehl und
5. der Rampenwert aus der Rampentabelle (Nummer der zu verwendenden Rampe).

Beispiel: Die SPS-Sequenz für Programm 4 soll umgeändert werden auf

$f_{ss} = 100 \text{ Hz}$,

$f_r = 1000 \text{ Hz}$,

Bewegungsart relativ mit der Distanz = 2300 Schritte,

Rampe Nr. 3 (das hieße bei den intern vorgegebenen Rampen: Steilheit 37500 Hz/s).

Der EW Befehl lautet: EW04 PO100 PF1000 GR2300 PN3

9.8.4 Erweiterte SPS-Sequenzen

Aufgrund von Kundenwünschen wurde die Funktionalität der SPS Befehle erweitert. Zusätzlich zu den bisherigen Befehlen, deren Parameter schon zur Programmierzeit berechnet und zusammen mit dem Befehl gespeichert werden, gibt es jetzt Sonderbefehle der Form „\$Onlinebefehl“.

Die Sonderbefehle für den erweiterten SPS Modus können einfach mit IPCOMM V2 programmiert werden. Die Programmierung dieser Befehle in der Steuerung kann auch mit OPM durchgeführt werden.

Als Befehle sind einige der Online Befehle zugelassen:

- \$PAX stellt neuen Booststrom ein (bzw auch \$PAX.x beim GCD)
- \$PB setzt alle Parameter zurück auf Auslieferungszustand
- \$PCxxx setzt den Positionszähler
- \$PFxxx stellt eine neue Lauffrequenz ein
- \$PNx ändert die Rampe
- \$POxxx verändert die Startstopfrequenz
- \$PRx stellt neuen Laufstrom ein (bzw auch \$PRx.x beim GCD)
- \$PSx stellt neuen Stopstrom ein (bzw auch \$PSx.x beim GCD)
- \$PT ändert die Laufstromüberhöhungszeit

- \$CA veranlasst Endstufenreset
- \$CI löst Initiatorreset aus
- \$CO setzt die Ausgangstreiber zurück
- \$CR veranlasst Reset der gesamten Steuerung
- \$CS setzt den SFI zurück (nur bei GSP)

- \$GAXxx führt eine Absolutfahrt aus
- \$GI startet eine Initialisierungsfahrt
- \$GRxxx startet eine Relativfahrt

- \$SH startet den Ausheizvorgang (nur bei GCD)

- \$&Nxx setzt den Zeiger neu, der Befehl in Zeile xx_H wird als nächster ausgeführt

- \$Txxx macht eine Pause von xxx ms Dauer

Außer dem Befehl in der Form „\$Onlinebefehl“ gibt es auch noch den Befehl in der Form „&Onlinebefehl“. Das & bedeutet, dass nach Abarbeiten des Befehles in der aktuellen Zeile noch der Befehl in der nächsten Zeile ausgeführt wird. Die Programmsequenz wird ausgeführt, solange das Strobesignal anliegt. Dadurch wird auf minimalistische Art und Weise eine Ablaufsteuerung implementiert.

Manual IPCOMM

Zur Veranschaulichung ist ein Teil einer OPM-Datei abgedruckt. Die Programmsequenz 8000 Schritte vor, 8000 Schritte zurück, 1600 Schritte vor, 1600 Schritte zurück, 2000 ms Pause, Sprung auf Zeile 03_H wird wiederholt, solange der Strobe Eingang mit 24 Volt beschaltet ist.

```
      ; Phytron parameter file (Date: 26.06.2000)
      ; -----
      ;
      ; [comment]
      ; You may freely edit this file to add your own
      ; comment.
      ; Any comment line must be preceded by a semicolon.
      ;
      ; [GCD]
      ;
      ; Q32 F10
      ; BIOS_1.11  _P25051845_
      ; MK_1.11   _P25051850_
      ;
      ; [parameters]
      ;
      PD1
      PA0
      PR3.4
      PS0.8
      PF2000
      PG1000000
      PH0
      PL1
      PM0
      PN0
      PO400
      PP0
      PT20
      PW0
      ;
      ; [PLC sequences]
      ;
      EW00$&PO250
      EW01$&PF1500
      EW02$&PN2
      EW03$&GR+8000
      EW04$&GR-8000
      EW05$&GR+1600
      EW06$&GR-1600
      EW07$&T2000
      EW08$&N03
```

9.8.5 Schnittstellenkommunikation

Die folgende Tabelle gibt einen Ausschnitt aus der Kommunikation zwischen PSS und dem IPCOMM wieder. Diese Tabelle ist ein Beispiel dafür, wie die PSS über die serielle Schnittstelle angesteuert werden kann.

Hinweis: Mit IPCOMM kann, durch eine Kommandozeilenoption, eine Kopie des gesamten Schnittstellenverkehrs auf eine Festplatte im PC kopiert werden. Dazu ist IPCOMM mit dem Parameter: IPCOMM /D:S einzurichten.

Die Daten werden in Dateien mit der Bezeichnung IPC_DSxxx geschrieben und im Verzeichnis TEMP (Umgebungsvariable) des PCs abgelegt.

Achtung: Die Dateien können sehr lang werden, da IPCOMM permanent den Status der PSS ausliest.

Kommando vom Leitrechner	Antwort von der PSS	Kommentar
<STX>1IB?:3F<ETX>	<STX>100:BIOS_1.04:62<ETX>	Abfrage BIOS-Type
<STX>1IC?:3E<ETX>	<STX>100:_K05051043_:7C<ETX>	Abfrage Datumcode
<STX>1IV?:2B<ETX>	<STX>100:IPP_1.04:3C<ETX>	Abfrage Versions-ID
<STX>1IF?:3B<ETX>	<STX>100:10000:00<ETX>	Abfrage Maximalfrequenz (10000 Hz)
<STX>1PG?:23<ETX>	<STX>100:10000000:30<ETX>	Abfrage Achsenbegrenzung (10000000 Schritte)
<STX>1IO?:32<ETX>	<STX>100:0:01<ETX>	Abfrage digitale Ausgänge
<STX>1PL?:28<ETX>	<STX>100:0:01<ETX>	Abfrage Bewegungsart (0 = rotatorisch)
<STX>1PD?:20<ETX>	<STX>100:0:01<ETX>	Abfrage Betriebsart (0 = Onlinebetrieb am PC)
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	Abfrage digitale Eingänge
<STX>1PF?:22<ETX>	<STX>100:5:04<ETX>	Abfrage aktuelle Lauffrequenz (hier: 5 Hz)
<STX>1PC?:27<ETX>	<STX>100:666:07<ETX>	Abfrage der Istposition
<STX>1PO?:2B<ETX>	<STX>100:400:05<ETX>	Abfrage der Start/Stopfrequenz
<STX>1PM?:29<ETX>	<STX>100:800:09<ETX>	Offset zum – Initiator nach Referenzfahrt
<STX>1PP?:34<ETX>	<STX>100:8000:39<ETX>	Offset zum + Initiator nach Referenzfahrt
<STX>1IS?:2E<ETX>	<STX>100:000008:39<ETX>	Statusabfrage
<STX>1PU?:31<ETX>	<STX>100:PSNORMAL 1.0.000:30<ETX>	(IPCOMM intern)
<STX>1IS?:2E<ETX>	<STX>100:000008:39<ETX>	Statusabfrage
<STX>1PX?:3C<ETX>	<STX>100:PRLINEAR 1.0.000:3F<ETX>	Abfrage Rampentypus (lineare Rampen erzeugt mit

Manual IPCOMM

Kommando vom Leitrechner	Antwort von der PSS	Kommentar
		der DLL PRLINEAR, Version 1.0.000)
<STX>1IN?:33<ETX>	<STX>100:10:30<ETX>	Abfrage Rampenanzahl
<STX>1FR0050:1A<ETX>	<STX>100:0032007D00FA017701F4027102EE036B:47<ETX>	Abfrage Rampeninformation (Speicheradressen der Rampentabelle)
<STX>1FR0060:19<ETX>	<STX>100:03E804B0FFFFFFFFFFFFFFFFFFFFFFFF:39<ETX>	Fortsetzung
<STX>1PN?:2A<ETX>	<STX>100:0:01<ETX>	Abfrage aktuell gewählte Rampe
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	Abfrage Eingangszustand: Eingang 1 ist aktiv, Eingänge 2,3,4 passiv
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	Fahrbefehl
<STX>1GR1234:1A<ETX>	<STX>101::30<ETX>	Relativfahrt um 1234 (dezimal) 1/8-Schritte in positive Richtung
<STX>1PC?:27<ETX>	<STX>101:670:01<ETX>	Positionsabfrage
<STX>1II?:34<ETX>	<STX>101:0:00<ETX>	Wiederholt Positions- und Eingangsabfrage
<STX>1PC?:27<ETX>	<STX>101:674:05<ETX>	
<STX>1II?:34<ETX>	<STX>101:0:00<ETX>	
<STX>1II?:34<ETX>	<STX>101:0:00<ETX>	
<STX>1II?:34<ETX>	<STX>101:0:00<ETX>	
<STX>1PC?:27<ETX>	<STX>101:1603:34<ETX>	
<STX>1II?:34<ETX>	<STX>101:0:00<ETX>	
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	
<STX>1II?:34<ETX>	<STX>100:0:01<ETX>	
<STX>1IS?:2E<ETX>	<STX>100:000000:31<ETX>	Erweiterte Statusabfrage

9.8.6 Auswahl der Baudrate

Die Kommunikation mit der PSS kann wahlweise mit 28800 oder 9600 Baud erfolgen, Voreinstellung 28800. Das IPCOMM Programm bietet im Hauptfenster eine „Checkbox“, mit deren Hilfe die Baudrate auf der Leitreechner-Seite an die Baudrate der Steuerung angepasst werden kann.

Für die Einstellung der Baudrate in der PSS ist im Lieferumfang von IPCOMM ein eigenes Hilfsprogramm „BAUDRATE.EXE“ enthalten.

Bei der Verwendung eigener PSS Software muss sichergestellt sein, dass eine Programmroutine zur Einstellung der Baudrate implementiert ist.

Siehe dazu folgendes Programmierbeispiel (TurboPascal):

```

THR          = $00;  { Transmitter Holding Register }
RBR          = $00;  { Receiver Buffer Register }
DLL          = $00;  { Divisor Latch, Least Significant }
DLM          = $01;  { Divisor Latch, Most Significant }
IER          = $01;  { Interrupt Enable Register }
IIR          = $02;  { Interrupt Identifier Register }
LCR          = $03;  { Line Control Register }
MCR          = $04;  { Modem Control Register }
LSR          = $05;  { Line Status Register }
MSR          = $06;  { Modem Status Register }
MR1          = $21;  { Interrupt-Maskenregister 1 }
MR2          = $A1;  { Interrupt-Maskenregister 2 }

DTR          = $01;
RTS          = $02;
OUT1         = $04;
OUT2         = $08;

type
PariType     = ( None, xxxx, ODD, EVEN );
const
Portnummer: byte   = 0;           { 0 für COM1 oder 1 für COM2 }
Interrupt: byte    = 4;           { oder 3 für COM2 }
COM_Port: word     = $03F8;       { oder $02F8 für COM2 }
Baudrate: longint = 28800;
Paritaet: PariType = None;
Wortlaenge: word  = 8;
Stopbits: word    = 1;

.....

procedure SetUART ( Baudrate:longint; parity:PariType; wlength, stops: byte );
{-----}
begin
  CLI;
  port[LCR+COM_Port] := $80;
  port[DLL+COM_Port] := lo(word(115200 div baudrate));
  port[DLM+COM_Port] := hi(word(115200 div baudrate));
  port[LCR+COM_Port] := (wlength-5) or ((stops-1) shl 2) or (ord(parity) shl 3);
  port[IER+COM_Port] := $01;
  port[MCR+COM_Port] := Port[MCR+COM_Port] OR $0B;
  if IRQ_Nummer < $10 then Port[MR1] := Port[MR1] AND ((IRQ_Maske) XOR $FF)
  else Port[MR2] := Port[MR2] AND ((IRQ_Maske) XOR $FF);
  Dummy := Port[LSR+COM_Port];
  Dummy := Port[RBR+COM_Port];
  STI;
end;

```

10 Stichwortverzeichnis

A

Achsenbegrenzung 15
Achsenstatus 15, 16
Adapterkabel 26
Anlagentyp 14
Arbeitsfläche 9
Arbeitsumgebung 9
Ausgangstreiber 17
Auswahlmenü 9

B

Baudrate 53
Befehle
 Ein-/Ausgangsbefehle 39
 Erweiterte Befehle 42
 Fahrbefehl 37
 Infobefehle 39
 Parameterbefehle 38
 Programmierbefehle 41
 Retrybefehl 42
 Rücksetzbefehle 40
 Schreibbefehle 41
 Sonderbefehle 43
 Statusbefehle 39
 Stop-Befehl 36
Befehlscode 23
Befehlsübersicht 36
Betriebsart 14
Betriebssystem 5
Bitrate 29
Broadcast 30
Broadcast-Adresse 37

C

CD 6
COM-Schnittstelle 25
Copyright 28

D

Default 14

F

Fahrbefehle 13, 18, 20
Fehler 15
Fehlermeldungen 25
Freeware 28

G

Geräteadresse 9

I

Installationsdateien 5
Installationsprogramm 6
IPCOMM-Icon 10
IPS2000 8
Istposition 15

K

Kabelbruch 25
Kabelkonfektion 25
Kontrolllampe 15

L

Lauffrequenz 14
Leistungsmerkmale 4
Leitrechner 29

N

Nullpunktoffset 15

O

Online-Betrieb 44
OPM 8, 46
Optionale Zusatzkarte 22

P

Parameter 14, 22
Parameterdatei 9, 10
Programminstallation 5
Prüfsumme 30

R

RAM Arbeitsspeicher 5
Rampen 14, 21
Reset 17
RS-Monitor 23

S

Schnittstellenfehler 25
Schnittstellenkommunikation 51
Separator 30
Setup.exe 6

SFI 8, 16, 17, 24
Softwarestruktur 47
Spieldausgleich 15
SPS-Befehle 48
SPS-Betrieb 44
SPS-Modus 18
SPS-Protokollsimulation 18
SPS-Sequenzen 48
Start 4
Start-/Stop-Frequenz 14
Statusinformation
 Erweiterter Status 34
 Kurzstatus 33
Statuszeile 9
Streckenregister 12, 15
Stromeinstellung 14

Stromüberhöhungszeit 14
Systemvoraussetzungen 5

T

Telegrammformat
 Antworttelegramm 31
 Sendetelegramm 30

U

Update 8

W

Warenzeichen 28
Windows 4

Phytron-Elektronik GmbH • Industriestraße 12 • 82194 Gröbenzell, Germany
Tel. +49(0)8142/503-0 • Fax +49(0)8142/503-190 • E-Mail info@phytron.de • www.phytron.de

Phytron, Inc. • 1347 Main Street • Waltham, MA, 02451 USA
Tel. +1-781-647-3581 • Fax +1-781-647-3526 • Email info@phytron.com • www.phytron.com