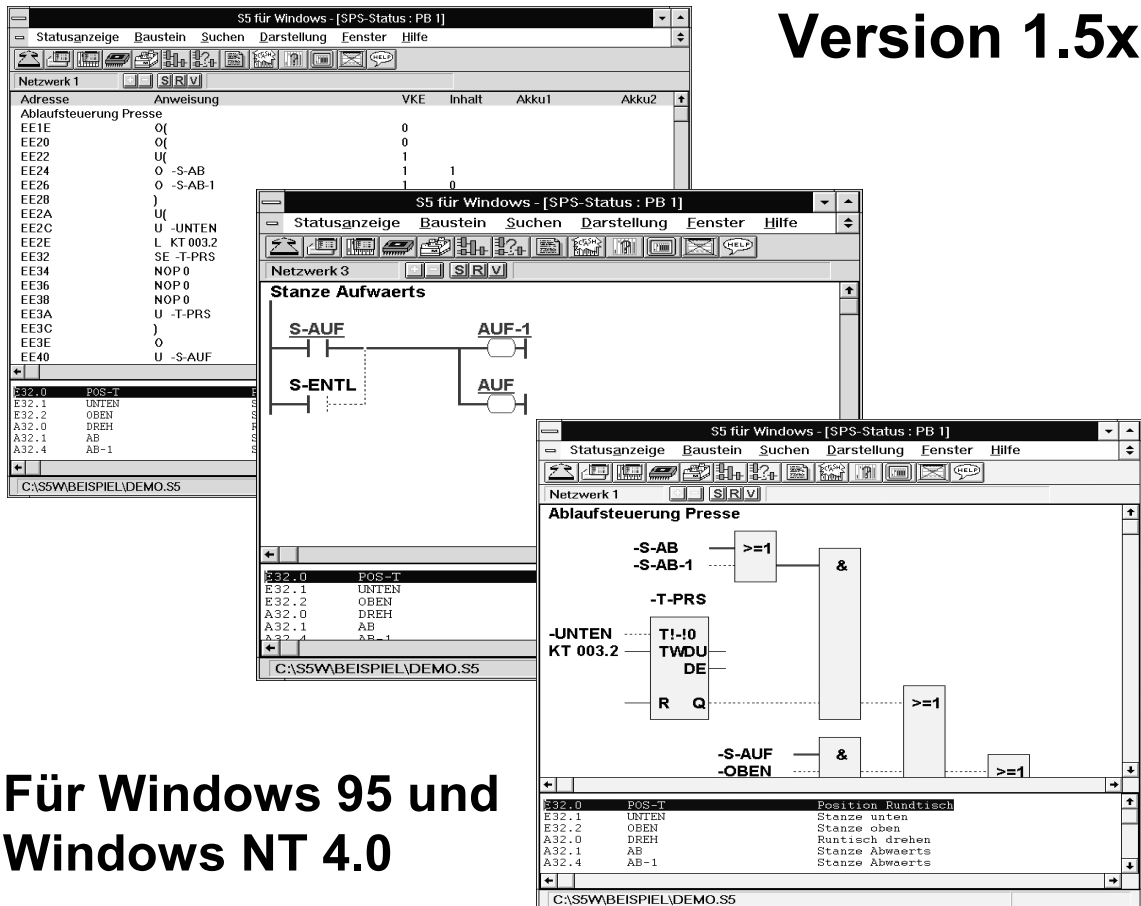


# S5 für Windows

## Erweiterte Simulations SPS PLC45

Version 1.5x



The image displays three overlapping screenshots of the S5 für Windows software interface, showing different views of a PLC program.

**Top-left screenshot:** Shows a variable declaration table for 'Netzwerk 1'.

Adresse	Anweisung	VKE	Inhalt	Akku1	Akku2
EE1E	O(	0			
EE20	O(	0			
EE22	U(	1			
EE24	O -S-AB	1	1		
EE26	O -S-AB-1	0			
EE28	)				
EE2A	U(	1			
EE2C	U -UNTEN				
EE2E	L KT 003.2				
EE32	SE -T-PRS				
EE34	NOP 0				
EE36	NOP 0				
EE38	NOP 0				
EE3A	U -T-PRS				
EE3C	)				
EE3E	)				
EE40	U -S-AUF				

**Top-right screenshot:** Shows a ladder logic diagram for 'Netzwerk 3' titled 'Stanze Aufwärts'. It features a normally open contact 'S-AUF' in series with a normally closed contact 'S-ENTL'. This combination is connected to a coil 'AUF-1'. A parallel branch with a normally open contact 'AUF' is also connected to coil 'AUF-1'.

**Bottom-right screenshot:** Shows a ladder logic diagram for 'Netzwerk 1' titled 'Ablaufsteuerung Presse'. It includes a timer 'T1-T0' and a timer 'KT 003.2'. The logic involves several normally open and normally closed contacts, including '-S-AB', '-S-AB-1', '-T-PRS', '-UNTEN', 'KT 003.2', 'R', 'Q', '-S-AUF', and '-OBEN'. There are also set coils '>=1' and an AND gate '&'.

**Bottom-left screenshot:** Shows a variable declaration table for 'Netzwerk 1'.

Adresse	Anweisung	Position	Rundtisch
E32.0	POS-T		
E32.1	UNTEN	Stanze unten	
E32.2	OBEN	Stanze oben	
A32.0	DREH	Rundtisch drehen	
A32.1	AB	Stanze Abwärts	
A32.4	AB-1	Stanze Abwärts	

Für Windows 95 und  
Windows NT 4.0

---

*S5 für Windows*  
Benutzerhandbuch  
***Simulations SPS PLC45***  
Version 1.5x  
( Windows 95 und Windows NT 4.0 )

---

Die in diesem Handbuch enthaltene Information kann ohne Vorankündigung geändert werden und stellt keine Verpflichtung seitens der IBH softec GmbH dar. Die Software, die in diesem Handbuch beschrieben ist, wird unter einer Lizenzvereinbarung und einer Geheimhaltungsvereinbarung zur Verfügung gestellt. Die Software darf nur nach Maßgabe der Bedingungen der Vereinbarung benutzt und kopiert werden. Es ist rechtswidrig, die Software auf ein anderes Medium zu kopieren, soweit das nicht ausdrücklich in der Lizenzvereinbarung und Geheimhaltungsvereinbarung erlaubt wird. Der Käufer darf eine Kopie zu Sicherheitszwecken machen. Ohne ausdrückliche schriftliche Erlaubnis von IBH softec GmbH dürfen für andere Zwecke als dem persönlichen Gebrauch durch den Käufer dieses Handbuchs nicht in irgendeiner Form mit irgendwelchen Mitteln, elektronisch oder mechanisch, mittels Fotokopie, durch Aufzeichnung oder mit Informationsspeicherungsgeräten reproduziert und übertragen werden.

© Copyright 1993 - 1997 IBH softec GmbH. Alle Rechte vorbehalten.



Gesellschaft für Automatisierungstechnik mbH

Turmstraße 77

D - 64743 Beerfelden / Odw.

Telefon 0 60 68 / 30 02

Telefax 0 60 68 / 30 74

Hotline: 06068 3001 BBS Mailbox: 06068 912013

Internet: [www.IBH Softec.de](http://www.IBH Softec.de) E-Mail: [INFO@IBHSOFTEC.DE](mailto:INFO@IBHSOFTEC.DE)

Windows, Windows NT sind eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Simatic® S5, Step® 5 und GRAPH® 5 sind eingetragene Warenzeichen der Siemens Aktiengesellschaft, Berlin und München.

**S5 für Windows Simulations SPS PLC45**

---

## Teil 1

<b>1</b>	<b>Installation <i>Simulations SPS PLC45</i></b> .....	3
1.1	<b>Systemanforderungen</b> .....	3
1.2	<b><i>PLC45</i> Installationsprogramm</b> .....	3
1.2.1	Installation starten .....	4
1.2.2	<i>PLC45</i> Start-Up Textdatei .....	5
1.2.3	<i>PLC45</i> Installationsmenü .....	6
1.2.3.1	Symbol einrichten .....	7
1.2.4	Installation erfolgreich abgeschlossen .....	8
1.3	<b>Starten der <i>PLC45</i> das erste Mal</b> .....	8
<b>2</b>	<b><i>PLC45</i> Online - Funktionen</b> .....	11
2.1	<b>Mit <i>S5 für Windows</i> direkt auf die <i>PLC45</i> zugreifen</b> .....	11
2.2	<b>Mit <i>S5 für Windows</i> über eine serielle Schnittstelle auf die <i>PLC45</i> zugreifen</b> .....	12
2.2.1	Schnittstellentreiber öffnen .....	13
2.2.2	Schnittstellentreiber schließen .....	14
2.3	<b><i>SPS</i> Programme in die <i>Simulations SPS</i> laden</b> .....	15
<b>3</b>	<b>Einführung <i>PLC45</i></b> .....	17
3.1	<b>Zeitverhalten der <i>PLC45</i></b> .....	17
3.1.1	Jitter .....	17
3.2	<b>Dialogfeld <i>PLC45</i></b> .....	18
3.2.1	Aufbau des <i>PLC45</i> Dialogfeldes .....	18
3.2.1.1	<i>SPS</i> Programm .....	19

3.2.1.2	Umläufe.....	19
3.2.1.3	Durchlaufzeit .....	20
3.2.1.4	Jitter .....	20
3.2.1.5	Serialnummer / Version .....	21
3.2.1.6	Einstellungen .....	21
3.2.1.7	SPS-Programm und Datenbausteine .....	21
3.2.1.8	I/O-Ports.....	23
3.2.1.9	SPS - Speicher .....	23
<b>4</b>	<b>Operationsvorrat <i>PLC45</i></b> .....	<b>25</b>
<b>4.1</b>	<b>Erweiterungen der <i>PLC45</i></b> .....	<b>25</b>
<b>4.2</b>	<b>Unterschiede zwischen der <i>PLC45</i> und der Siemens CPU945</b> .....	<b>26</b>
<b>4.3</b>	<b>Funktionen, die in der <i>PLC45</i> nicht vorhanden sind</b> .....	<b>26</b>
<b>4.4</b>	<b>Operationsvorrat</b> .....	<b>28</b>
4.4.1	Verknüpfungsoperationen Binär.....	28
4.4.1.1	UND Verknüpfungen mit Abfrage auf Signalzustand 1 .....	28
4.4.1.2	UND Verknüpfungen mit Abfrage auf Signalzustand 0 .....	28
4.4.1.3	ODER Verknüpfungen mit Abfrage auf Signalzustand 1 .....	29
4.4.1.4	ODER Verknüpfungen mit Abfrage auf Signalzustand 0 .....	29
4.4.1.5	Sonstige Verknüpfungen .....	30
4.4.2	Speicheroperationen Binär.....	30
4.4.3	Ladeoperationen .....	31
4.4.3.1	Laden von Variablen.....	31
4.4.3.2	Laden von Konstanten.....	32
4.4.4	Transferoperationen.....	32
4.4.5	Zeitoperationen .....	33
4.4.6	Zähloperationen .....	34

---

4.4.7	Arithmetische Operationen .....	35
4.4.8	Vergleichsoperationen .....	35
4.4.9	Bausteinaufrufoperationen.....	36
4.4.9.1	Absolute Aufrufe .....	36
4.4.9.2	Bedingte Aufrufe (wenn VKE=1) .....	36
4.4.9.3	Datenbausteinoperationen .....	36
4.4.10	Bausteinrücksprungoperationen .....	37
4.4.10.1	Nulloperationen .....	37
4.4.11	Stoppoperationen.....	37
4.4.12	Bitoperationen .....	37
4.4.12.1	Abfrage auf Bitzustand = 1 .....	37
4.4.12.2	Abfrage auf Bitzustand = 0 .....	38
4.4.12.3	Setzen eines Bits .....	38
4.4.12.4	Rücksetzen eines Bits .....	38
4.4.13	Umwandlungsoperationen .....	39
4.4.14	Schiebe- und Rotieroperationen .....	39
4.4.15	Sprungoperationen .....	39
4.4.16	Sonstige Operationen .....	40
4.4.17	Gleitpunktzahloperationen .....	40
4.4.17.1	Umwandeloperationen.....	40
4.4.17.2	Ladeoperationen.....	40
4.4.17.3	Vergleichsoperationen.....	41
4.4.17.4	Arithmetische Operationen.....	41
4.4.17.5	Lade- und Transferoperationen.....	42
4.4.18	Vier Akkumulatoren (CPU 948 kompatibel).....	44
4.4.18.1	Arithmetische Operationen (vier Akkumulatoren Operationen).....	45
4.4.18.2	Registeroperationen (vier Akkumulatoren Operationen).....	45
<b>4.5</b>	<b>Aufbau des SPS-Speichers der PLC45.....</b>	<b>46</b>
<b>4.6</b>	<b>Aufbau der Systemdaten PLC45.....</b>	<b>47</b>
<b>4.7</b>	<b>In der PLC45 integrierte Organisationsbausteine (OB).....</b>	<b>50</b>

---

4.7.1	OB 19 Reaktion bei Aufruf eines nicht geladenen Bausteins .....	50
4.7.2	OB 27 Reaktion bei einem Substitutionsfehler .....	50
4.7.3	OB 31 Zyklusüberwachungszeit neu starten .....	51
4.7.4	OB 32 Reaktion bei einem Transfer- / Ladebefehlfehler.....	51
4.7.5	OB 250 Betriebssystemdienste.....	51
4.7.6	OB 251 PID - Regelalgorithmus.....	53
<b>5</b>	<b>DDE Manager.....</b>	<b>55</b>
5.1	Beispiel.....	57
<b>6</b>	<b>Visualisierungsprogramme .....</b>	<b>59</b>
6.1	Schneller Direktzugriff für Visual Basic .....	59

## Anhang

1	Null Modem Kabel.....	63
---	-----------------------	----

---

---

**Teil 1**

***Simulations SPS PLC45***

**Technische Details**

---

# 1 Installation *Simulations SPS PLC45*

---

Die *Simulations SPS PLC45* ist lauffähig unter den Betriebssystemen Windows 95 und Windows NT Version 4.0 von Microsoft.

---

## 1.1 Systemanforderungen

Die Installation der *PLC45* kann auf jedem Rechner erfolgen, auf dem Microsoft Windows 95 oder Windows NT 4.0 installiert ist.

Um die Betriebssysteme voll zu nutzen, sollte der Rechner bei Windows 95 über mindestens 16 MB RAM Speicher und bei Windows NT über mindestens 32 MB RAM Speicher verfügen.

Für die Installation der *PLC45* wird ein Floppylaufwerk mit dem Diskettenformat 1,44 MB 3,5 Zoll benötigt.

Zum Online Betrieb mit einem externen Programmiersystem ist eine freie serielle Schnittstelle (Com1 - Com4) notwendig.

---

## 1.2 *PLC45* Installationsprogramm

Das Installationsprogramm führt für Sie alle notwendigen Schritte zum Installieren der *PLC45* auf der Festplatte durch. Es wird Ihnen ein Verzeichnis für die Installation vorgeschlagen. Die Datei README.TXT, die die letzten Änderungen enthält, wird angezeigt. Die Symbole und Einträge zum Starten der *PLC45* und der dazugehörigen Funktionen werden in die Programmverwaltung eingetragen. Unterverzeichnisse mit Beispielen werden ebenfalls erstellt. Falls Sie den vorgeschlagenen Ordner (Verzeichnis, Laufwerk oder Pfad) nicht nehmen wollen, können Sie das Verzeichnis nach Ihren Wünschen ändern.

- ◆ Falls Windows (95/NT) noch nicht auf Ihrem Rechner läuft, starten Sie Windows.

## 1.2.1 Installation starten

- ◆ Start, Ausführen anklicken.

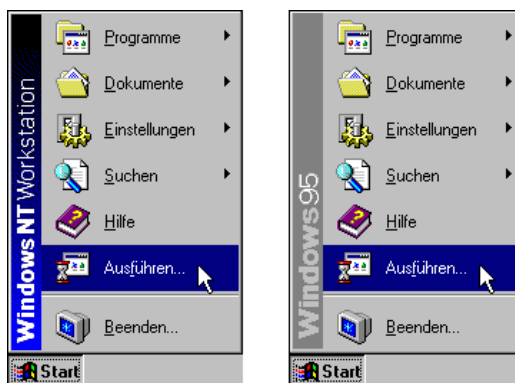


Abbildung 1-1

Das Dialogfeld Ausführen wird geöffnet.

### Dialogfeld Ausführen:



Abbildung 1-2

In das Textfeld Öffnen geben Sie bitte den Laufwerksnamen (z.B. **a:\** oder **b:\**) und **install.exe** ein. Als Laufwerksnamen nehmen Sie die Bezeichnung des Diskettenlaufwerks, in dem sich die **PLC45** Installationsdiskette Nr. 1 befindet. Über das Schaltfeld Durchsuchen können Sie auch das Laufwerk und die Datei **install.exe** auswählen.

- ◆ Schaltfläche **OK** anklicken.

- ◆ Mit der **EINGABE** Taste bestätigen.

Das Installationsprogramm ist jetzt gestartet. Folgen Sie jetzt bitte den Anweisungen auf Ihrem Bildschirm.

### 1.2.2 *PLC45* Start-Up Textdatei

Als erstes werden Ihnen in einer Textdatei wichtige Informationen zur Installation angezeigt. Bitte beachten Sie diese Hinweise.

Beispiel einer Start-Up Textdatei

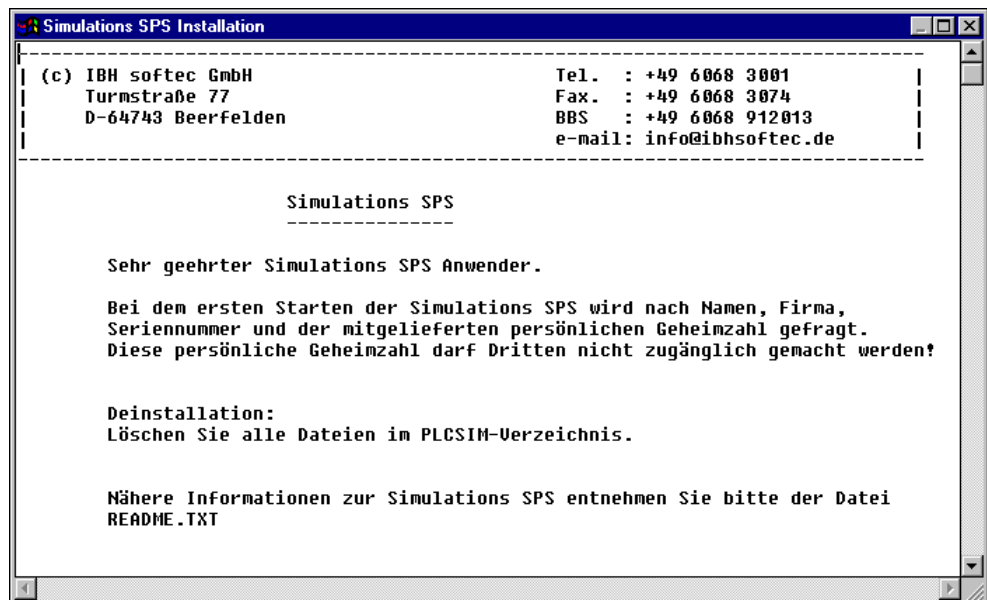


Abbildung 1-3

- ◆ Schließen Sie die Textdatei, nachdem Sie sie gelesen haben.



- ◆ Klicken Sie hierzu die Schaltfläche in der rechten oberen Ecke des Textfensters an.



- ◆ Tasten **ALT+F4** betätigen.

### 1.2.3 *PLC45* Installationsmenü

Sollten Sie mit dem Vorschlag, daß die *PLC45* auf der Festplatte **C:** im Ordner **plcsim** installiert wird, nicht einverstanden sein, geben Sie die neue Laufwerkbezeichnung und den neuen Ordner ein.

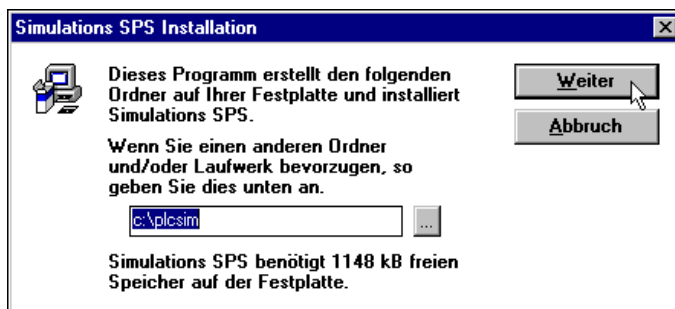






Abbildung 1-4

- ◆  Schaltfläche **Weiter** anklicken.
- ◆  Bestätigen Sie das Laufwerk und Verzeichnis mit der **EINGABE** Taste.

Das Installationsprogramm erstellt das Verzeichnis Ihrer Wahl auf dem angegebenen Laufwerk und kopiert die Dateien dorthin. Zusätzlich werden Unterverzeichnisse für Beispiel erzeugt.

Das Installationsprogramm können Sie wie folgt abbrechen:

- ◆  Die Schaltfläche **Abbruch** anklicken.
- ◆  Tasten **ALT+A** betätigen.

Das Installationsprogramm zeigt Ihnen (Abbildung 1-5), welche Dateien im Moment kopiert werden und den Fortschritt der Installation an.

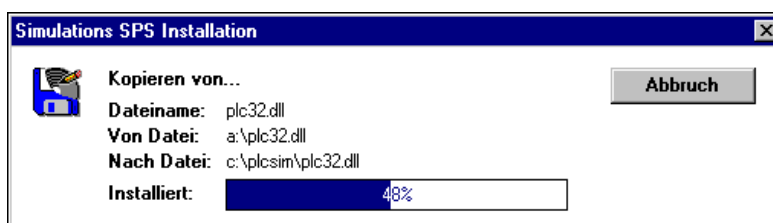


Abbildung 1-5

Bei der Anzeige von 100% sind alle für die *PLC45* erforderlichen Dateien auf die Festplatte kopiert worden. Der Kopiervorgang benötigt einige wenige Minuten.

### 1.2.3.1 Symbol einrichten

Nach Beendigung des Kopiervorganges werden Sie gefragt, ob eine Gruppe in dem Programm Manager erstellt werden soll. Bei Windows NT handelt es sich um den Programmnamen und das Symbol, um die *PLC45* zu starten. Es wird vorgeschlagen, das Programmsymbol *S5 für Windows* zu nennen. Möchten Sie eine andere Bezeichnung benutzen, tragen Sie diesen Namen in das Eingabefeld ein.

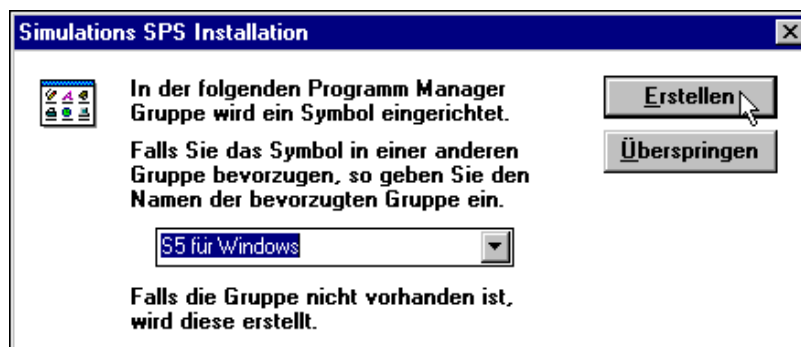


Abbildung 1-6

 ◆ **Erstellen** anklicken.

 ◆ Mit der Taste **EINGABE** bestätigen.

Möchten Sie, daß kein Symbol zum Starten der *PLC45* erstellt wird, sollten Sie diesen Schritt überspringen.

 ◆ **Überspringen** anklicken.

 ◆ Tasten **ALT+Ü** bestätigen.

### 1.2.4 Installation erfolgreich abgeschlossen

Die erfolgreich abgeschlossene Installation wird mit dem Einrichten der Gruppe *S5 für Windows* und den dazugehörigen Symbolen angezeigt.

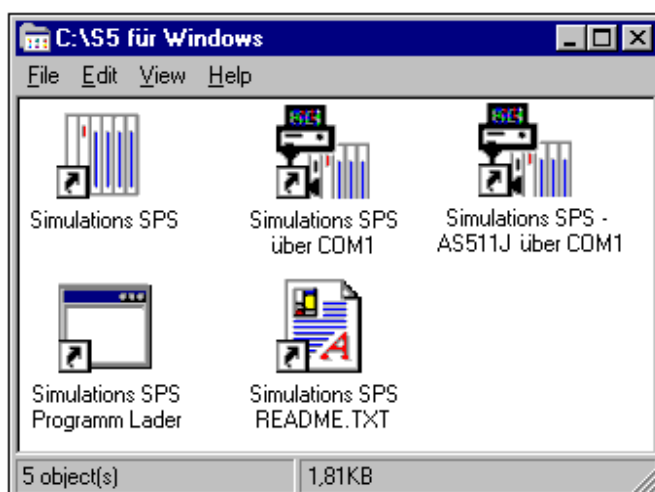


Abbildung 1-7

Durch Anklicken des Symbols *Simulations SPS README.TXT* sollten Sie die Textdatei, die Ihnen die neuesten Informationen zur *PLC45* gibt, öffnen.



Simulations SPS  
README.TXT

---

## 1.3 Starten der *PLC45* das erste Mal

Um die *PLC45* zu starten, muß die Software mit Ihrer persönlichen Geheimzahl freigeschaltet werden. Ihre persönliche Geheimzahl und die Seriennummer der *PLC45* finden Sie auf der Freischaltkarte, die zusammen mit der Software-Registrierkarte (Registration Card) den Installationsdisketten beigelegt ist.

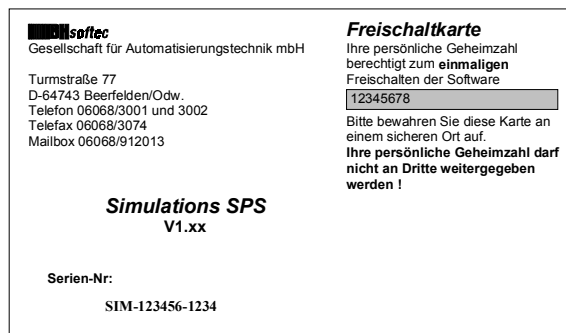


Abbildung 1-8

Wurde wie bei der Installation vorgeschlagen, die Gruppe *S5 für Windows* eingerichtet, können Sie die *PLC45* wie folgt starten:

- ◆ **Start, Programme, S5 für Windows , PLC45** markieren und mit der linken Maustaste anklicken.

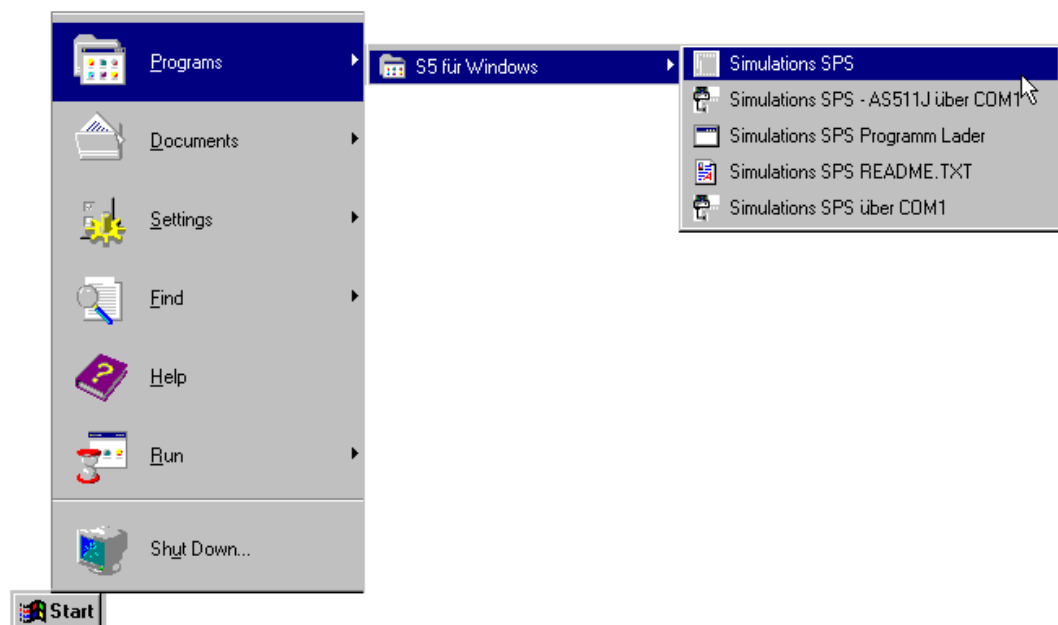


Abbildung 1-9

Das Dialogfeld zur Eingabe der Seriennummer und Ihrer persönlichen Geheimzahl zum Freischalten der Software wird geöffnet.

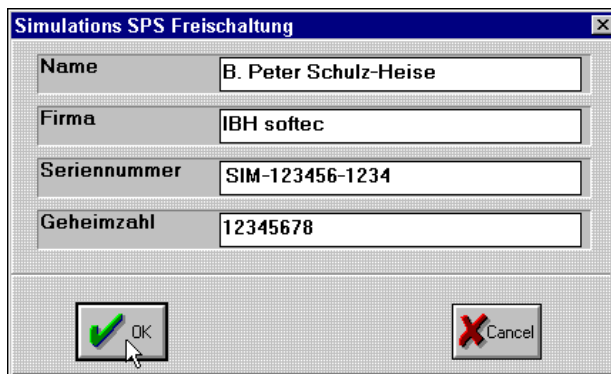


Abbildung 1-10

Geben Sie Ihren Namen und den der Firma ein. Die Seriennummer und die Geheimzahl müssen exakt wie auf der Freischaltkarte angegeben, eingegeben werden (Großbuchstaben, Bindestriche usw.).

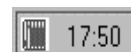


- ◆ Klicken Sie **OK**, um die **PLC45** Software freizuschalten.



- ◆ Taste **EINGABE** betätigen, um die **PLC45** Software freizuschalten.

In der rechten unteren Ecke (neben der Uhrzeit) wird das Symbol der **PLC45** angezeigt.



**Anmerkung:**

Die Farbe der Umrandung des **PLC45** Symbols zeigt den Status der **PLC45** an.

Umrandung:

- |             |  |
|-------------|--|
| <b>Gelb</b> | Kein SPS Programm in der <b>PLC45</b> vorhanden. |
| <b>Rot</b>  | <b>PLC45</b> in STOP                             |
| <b>Grün</b> | <b>PLC45</b> in RUN                              |

## 2 PLC45 Online - Funktionen

---

Der Datenaustausch mit der **PLC45** kann direkt aus *S5 für Windows* oder über eine serielle Schnittstelle erfolgen.

---

### 2.1 Mit *S5 für Windows* direkt auf die **PLC45** zugreifen

*S5 für Windows* bietet Ihnen die Möglichkeit, direkt auf die **PLC45** zuzugreifen, um Programme zu übertragen. Alle *S5 für Windows* Online Funktionen stehen bei dem direkten Zugriff auf die **PLC45** zur Verfügung. Der Datenaustausch erfolgt extrem schnell.

Um direkt auf die **PLC45** zuzugreifen, müssen die folgenden Dateien in das *S5 für Windows* Verzeichnis kopiert werden.

**Ab *S5 für Windows* Version 3.10:**

PLCCMNT.DLL

S5WPGI.DLL

**Ab *S5 für Windows* Version 3.00:**

PLCCMNT.DLL

S5WPGI30.DLL            diese Datei ist dann in S5PGI.DLL umzubenennen

Falls Sie eine ältere *S5 für Windows* Version (z.B. V 2.1x) besitzen, bestellen Sie bitte ein Update.

**Anmerkung:**

Die oben aufgeführten Dateien müssen in dem *S5 für Windows* Verzeichnis **und** dem **PLC45** Verzeichnis(C:\PLCSIM) vorhanden sein.

Die Dateien müssen kopiert werden und nicht verschoben werden.

## 2.2 Mit *S5 für Windows* über eine serielle Schnittstelle auf die *PLC45* zugreifen

Bei der Installation wird der Schnittstellentreiber (PLCCOM.EXE) für die Initialisierung der COM 1 Schnittstelle auf eine Baudrate von 38 400 Baud vorbereitet.

Wollen Sie für den Datenaustausch mit der *PLC45* die COM 1 Schnittstelle benutzen, so brauchen Sie nur das Programm PLC43 über COM1 aus der *S5 für Windows* Gruppe zu starten.



Als Verbindungskabel zwischen der COM Schnittstelle der *PLC45* und der COM Schnittstelle des externen *S5 für Windows* Rechners wird ein Nullmodem- Kabel benötigt (siehe Anhang).

Um eine andere COM Schnittstelle und/oder eine andere Baudrate einzustellen, ist das Schnittstellentreiberprogramm (PLCCOM.EXE) mit entsprechenden Parametern zu starten.

COM1	PLCCOM.EXE 0	38 400 Baud	PLCCOM.EXE n 38800
COM2	PLCCOM.EXE 1	19 200 Baud	PLCCOM.EXE n 19200
COM3	PLCCOM.EXE 2	14 400 Baud	PLCCOM.EXE n 14400
COM4	PLCCOM.EXE 3	9 600 Baud	PLCCOM.EXE n 9600

### Beispiel:

**PLCCOM.EXE 1 14400**

COM 2, Baudrate 14 400

Sollte die Möglichkeit geschaffen werden, ein Symbol zum Aufruf einer anderen COM Schnittstelle und/oder einer anderen Baudrate zu generieren, so gehen Sie wie folgt vor:

1. Im Explorer markieren Sie die Datei PLCCOM.EXE.
2. Klicken Sie den Befehl **Verknüpfung erstellen** im Menü Datei an. Im Explorer (Inhalt von C:\PLCSIM) wird die Verknüpfung angezeigt.
3. Klicken Sie den Befehl **Eigenschaften** im Menü Datei an. Das Dialogfeld Eigenschaften von Verknüpfungen mit *plccom.exe* wird geöffnet.

4. Unter dem Reiter **Verknüpfung** werden im Textfeld **Ziel** die Parameter für das Programm *plccom.exe* eingegeben.
5. Bestätigen Sie die Eingabe im Textfeld *Ziel* mit **OK**.

#### Dialogfeld Eigenschaften von Verknüpfungen mit plccom.exe

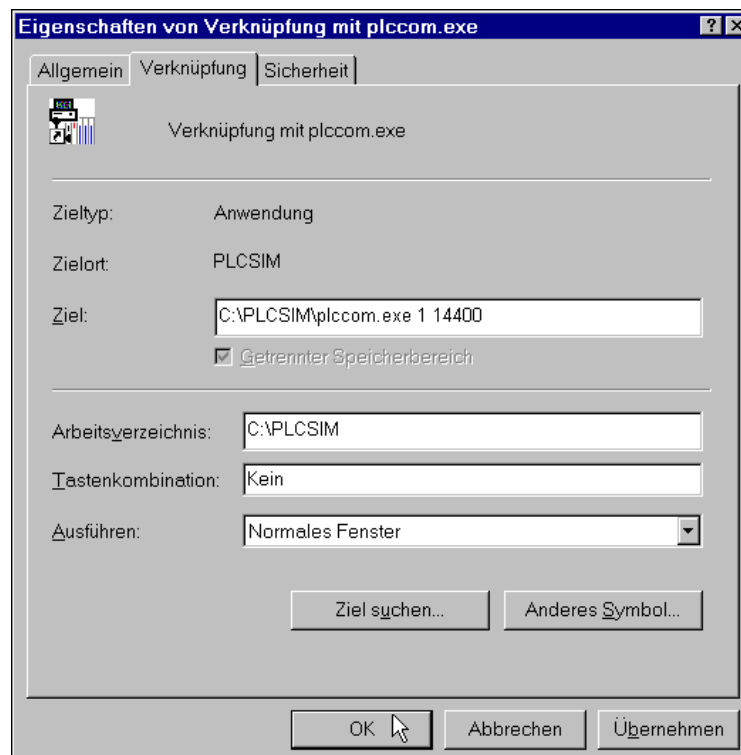


Abbildung 2-1

### 2.2.1 Schnittstellentreiber öffnen

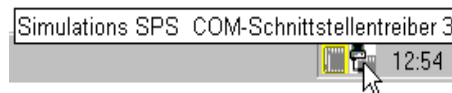
Der Schnittstellentreiber (PLCCOM.EXE) kann wie unter Abschnitt 2.2 beschrieben, gestartet werden. Außerdem kann das Schnittstellentreiberprogramm (PLCCOM.EXE) wie folgt gestartet werden: Öffnen Sie das Dialogfeld **Ausführen** (Abbildung 1-2). In die Befehlszeile geben Sie bitte den Pfad (Laufwerk, Ordner) und den Programmnamen mit den entsprechenden Parametern ein. Über das

Schaltfeld **Durchsuchen** können Sie auch das Laufwerk und die Datei **plccom.exe** auswählen. Die Parameter sind entsprechend der obigen Tabelle hinzuzufügen.

Ist eine Schnittstelle durch Aufruf des Programms PLCCOM.EXE definiert, wird in der rechten unteren Ecke (neben der Uhrzeit) das Symbol für die Schnittstelle angezeigt.



Wenn sich der Mauszeiger auf dem Symbol befindet, wird die initialisierte COM Schnittstelle angezeigt.



## 2.2.2 Schnittstellentreiber schließen

Falls Sie eine Änderung der Schnittstellenparameter oder einen anderen Schnittstellentreiber einsetzen wollen, ist es erforderlich, die Schnittstelle vorher zu schließen.

- ◆ Symbol in der Task-Leiste mit der linken Maustaste anklicken.

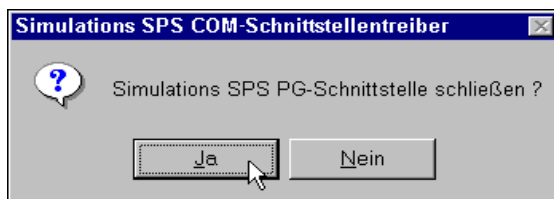
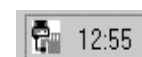


Abbildung 2-2

Bestätigen Sie das Dialogfeld. Das Schnittstellen-Treiberprogramm ist damit beendet.

### Anmerkung:

Wird Windows erneut gestartet, so wird **kein** Schnittstellentreiber automatisch gestartet.

Soll beim Hochlaufen von Windows ein Schnittstellentreiber automatisch geladen werden, so ist das entsprechende Programm in die Auto-Start-Gruppe einzutragen.

## 2.3 SPS Programme in die Simulations SPS laden.

In die *PLC45* können Programme auch ganz ohne Programmiersystem geladen werden.

Bei der Installation wurde das Programm **PLCLDR.EXE** in das Verzeichnis *PLCSIM* geladen. Mit diesem Programm können SPS Programme im *S5 für Windows* (\*.S5) und im Siemens Step® 5 Format (\*.S5D) direkt in die *PLC45* geladen werden.

Das Programm **SPS Direkt Laden (PLCLDR.EXE)** kann wie folgt gestartet werden: Öffnen Sie das Dialogfeld **Ausführen** (Abbildung 2-3). In die Befehlszeile geben Sie bitte den Pfad (Laufwerk, Ordner) und den Programmnamen mit den entsprechenden Parametern ein. Über das Schaltfeld **Durchsuchen** können Sie auch das Laufwerk und die Datei **plcldr.exe** auswählen.

Als Parameter ist das zu ladende SPS Programm mit dem dazugehörige Pfad einzugeben.

Beispiel SPS Programm laden.



Abbildung 2-3

Das Dialogfeld **Ausführen** zur Eingabe des Programms wird geöffnet.

Dialogfeld **Ausführen**:

Der Eintrag in der Befehlszeile (Abbildung 2-4) führt zum direkten Übertragen eines SPS Programmes im *S5 für Windows* Format (\*.S5) in die *PLC45*.

Dialogfeld **Ausführen**.

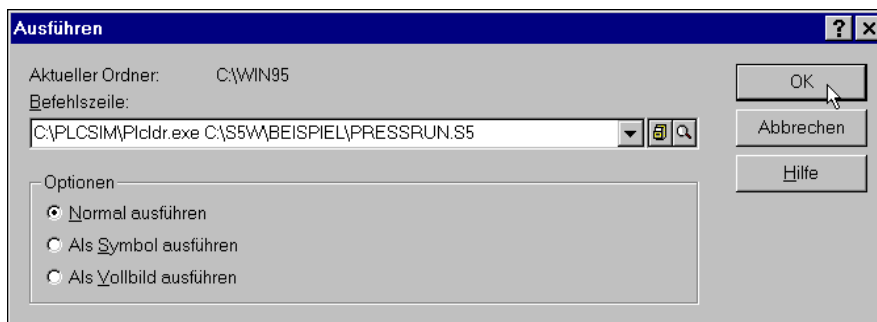


Abbildung 2-4

Der Eintrag in der Befehlszeile führt zum direkten Übertragen eines SPS Programmes im Siemens Step@ 5 Format (\*.S5D) in die PLC45.

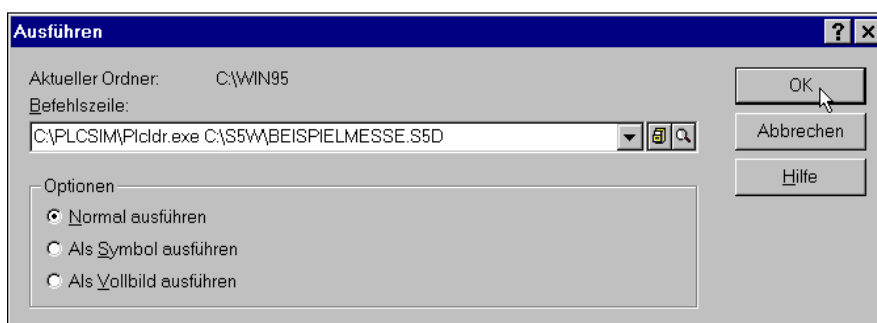


Abbildung 2-5

## 3 Einführung *PLC45*

---

Die *PLC45* ist eine Software SPS, die als Interpreter ausgelegt ist. Das SPS Programm wird in nicht kompilierter Form (wie bei einer Hardware SPS) abgearbeitet. Dies bietet den Vorteil, daß der Status direkt, ohne daß ein Recompile durchgeführt werden muß, angezeigt werden kann. Das Austesten eines SPS Programms wird dadurch sehr erleichtert, da jederzeit festgestellt werden kann, welcher Befehl gerade abgearbeitet wurde.

---

### 3.1 Zeitverhalten der *PLC45*

Das Zeitverhalten der *PLC45* ist fest vorgegeben. Das Zeitverhalten der *PLC45* wird in dem *PLC45* Dialogfeld angezeigt (siehe Abschnitt 4.1.3, 4.1.4).

Untersuchungen haben ergeben, daß SCSI- Festplatten (Controller als Bus-Master) das Echtzeitverhalten der *PLC45* weitaus weniger beeinflussen als AT-Bus Festplatten (IDE). Die Kommunikation mit IDE-Devices, egal ob Festplatten oder CD-ROM-Laufwerke, muß die Rechner-CPU interruptgesteuert selbst abwickeln. Bei SCSI hingegen erledigt dies ein eigener Prozessor auf dem Hostadapter. Auch bei der Auswahl von Netzwerkkarten sollten Sie eine auswählen, die die CPU der Mutterplatine möglichst wenig belastet. Dies gilt für alle Komponenten der *PLC45* Rechner.

#### 3.1.1 Jitter

Mit *Jitter* bezeichnet man unregelmäßige zeitlichen Schwankungen von Vorgängen. Der *Jitter* bei einer SPS Steuerung ist die zeitliche Schwankung der Zykluszeit, die vom System verursacht wird.

Wird bei einer SPS in einem Zeit-OB (z.B. OB 10) alle 10 Millisekunden ein Impuls (Ausgang) ausgegeben, so ist der *Jitter* die Schwankung der Zeit, die von den 10 Millisekunden abweicht. Zeitschwankungen sind sowohl bei Hardware-SPS-Steuerungen wie auch bei Software-SPS-Steuerungen vorhanden.

Neben programmtechnischen Zeitverschiebungen (programmierte Programmunterbrechungen wie z.B. Prozeßalarme) können bei einer Software-SPS-Steuerung Zeitschwankungen durch Festplattenzugriffe, Netzwerkzugriffe usw. verursacht werden.

## 3.2 Dialogfeld *PLC45*

Nachdem Sie die *PLC45* gestartet haben (siehe Abschnitt 1.3) können Sie durch Anklicken das *PLC45* Dialogfeld öffnen.



### 3.2.1 Aufbau des *PLC45* Dialogfeldes

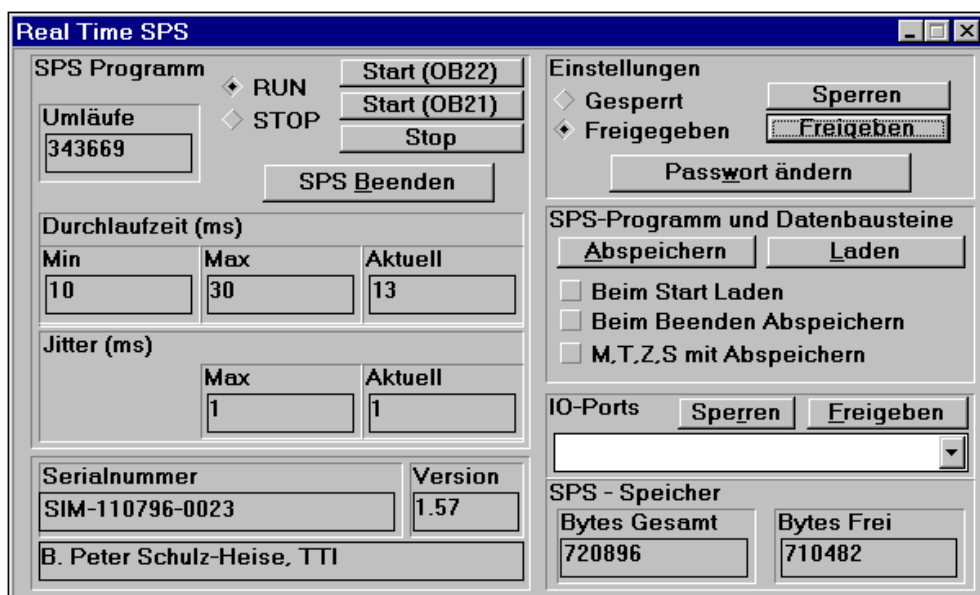
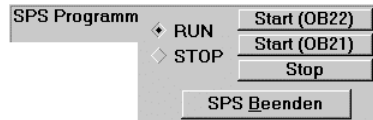


Abbildung 3-1

Das Zeitverhalten der *PLC45* wird kontinuierlich im Dialogfeld angezeigt.

### 3.2.1.1 SPS Programm

Im SPS Programm Feld wird der Zustand der CPU angezeigt:



◆ **RUN** Das SPS Programm wird abgearbeitet

◆ **STOP** Das SPS Programm wird nicht abgearbeitet

Mit Betätigen der Schaltfläche **Start (OB22)** wird das SPS Programm gestartet. Als erstes wird der Organisationsbaustein OB22 abgearbeitet. Erst danach läuft die zyklische Programmbearbeitung mit OB1 an. Dies entspricht dem Anlauf einer Hardware SPS nach Spannungswiederkehr.

Mit Betätigen der Schaltfläche **Start (OB21)** wird das SPS Programm gestartet. Als erstes wird der Organisationsbaustein OB21 abgearbeitet. Erst danach läuft die zyklische Programmbearbeitung mit OB1 an. Dies entspricht dem Anlauf einer Hardware SPS nach manuellen Einschalten (STOP → RUN).

Mit betätigen der Schaltfläche **Stop** wird die zyklische Programmbearbeitung unterbrochen. In der Betriebsart STOP wird das SPS Programm nicht abgearbeitet. Der Zustand der Prozeßabbilder, der Merker, Zähler und Zeiten, die beim Eintritt des STOP – Zustand aktuell waren, bleiben erhalten.

Mit Betätigen der Schaltfläche **SPS Beenden** wird das PLC45 Programm beendet. Ein erneutes Starten des PLC45 Programms ist jederzeit möglich (siehe Abschnitt 1.3).

### 3.2.1.2 Umläufe

Die Anzahl der zyklischen Programmdurchläufe wird angezeigt. Mit Betätigen der Schaltfläche **STOP** wird der Umlaufzähler angehalten. Ein erneuter START setzt den Zähler zurück und die Umläufe werden wieder gezählt.



Die Anzahl der zyklischen Umläufe (OB1 Umläufe) können mit dem SPS Programm ausgelesen werden.

- BS 20: niederwertiges WORT
- BS 21: höherwertiges WORT

### 3.2.1.3 Durchlaufzeit

Die Zeit, die ein zyklischer Programm-durchlauf benötigt, wird von der *PLC45* kontinuierlich gemessen. Die minimale, maximale und die aktuelle Zykluszeit des momentan abzuarbeitenden SPS

Durchlaufzeit (ms)		
Min	Max	Aktuell
10	30	13

Programms werden angezeigt. Die Auflösung der Zykluszeitmessung ist eine Millisekunde.

Die Zykluszeit einer SPS kann je nach SPS Programmaufbau erheblich schwanken.

Am Ende eines Zyklus speichert die *PLC45* das Ergebnis der Zykluszeitmessung. Dies ist die Zeit, die zwischen einem OB1 Aufruf bis zum nächsten OB1 Aufruf vergangen ist.

Die Zeit der zyklischen Durchläufe (OB1 Durchlaufzeit) kann mit dem SPS Programm ausgelesen werden.

- BS 121: aktuelle Durchlaufzeit in ms
- BS 122: maximale Durchlaufzeit in ms
- BS 121: minimale Durchlaufzeit in ms

### 3.2.1.4 Jitter

Mit *Jitter* bezeichnet man unregelmäßige zeitlichen Schwankungen von Vorgängen. Der *Jitter* bei einer SPS Steuerung ist die zeitliche Schwankung der Zykluszeit, die vom System verursacht wird.

Wird bei einer SPS in einem Zeit-OB (z.B. OB 10) alle 10 Millisekunden ein Impuls (Ausgang) ausgegeben, so ist der *Jitter* die Schwankung der Zeit, die von den 10 Millisekunden abweicht. Zeitschwankungen sind sowohl bei Hardware-SPS-Steuerungen wie auch bei Software-SPS-Steuerungen vorhanden.

Neben programmtechnischen Zeitverschiebungen (programmierte Programmunterbrechungen wie z.B. Prozeßalarme können bei einer Software-SPS-Steuerung Zeitschwankungen durch Festplattenzugriffe, Netzwerkzugriffe usw. verursacht werden.

Der *Jitter* der *PLC45* wird kontinuierlich gemessen. Die maximale und die aktuelle Zeitabweichung von einem imaginären, festen Zeittakt werden angezeigt. Die Auflösung der Zeitmessung ist eine Millisekunde. Die angezeigte Zeitschwankung addiert sich zur Reaktionszeit der *PLC45* und muß entsprechend berücksichtigt werden.

Jitter (ms)	
Max	Aktuell
1	1

Die Jitter-Zeiten können mit dem SPS Programm ausgelesen werden

- Maximale Jitter-Zeit in ms
- Aktuelle Jitter-Zeit in ms

### 3.2.1.5 Seriennummer / Version

Die Seriennummer und die Versionsnummer der installierten *PLC45* und der Benutzername mit Firma werden in einem Textfeld angezeigt.

Serialnummer	Version
SIM-110796-0023	1.57
B. Peter Schulz-Heise, TTI	

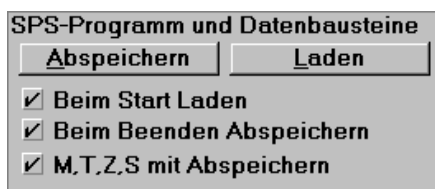
### 3.2.1.6 Einstellungen

Bei der *PLC45* sind die Einstellungen nicht relevant und haben keinen Einfluß.

Einstellungen	
<input type="checkbox"/> Gesperrt	Sperren
<input checked="" type="checkbox"/> Freigegeben	Freigegeben
Passwort ändern	

### 3.2.1.7 SPS-Programm und Datenbausteine

Schaltflächen zum Speichern und zum Laden eines SPS-Programms sind vorhanden.



**Abspeichern** Das momentan in der *PLC45* befindliche SPS-Programm wird unter **C:\PLCSIM\PLC.BIN** auf der Festplatte gespeichert. Ob außer dem SPS-Programm auch die momentan gültigen Merker (M), Zeiten (T), Zähler (Z) und S-Merker (S) mit gespeichert werden sollen, kann mit der entsprechenden Schaltfläche festgelegt werden.

Die *PLC45* zeigt Ihnen folgende Warnung vor dem eigentlichen Abspeichern an.

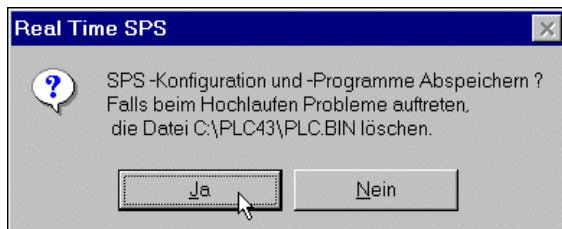


Abbildung 3-2

Sollte bei dem nächsten Hochlaufen des Rechners Probleme auftreten, so löschen Sie die Datei **C:\PLCSIM\PLC.BIN**. Solche Probleme können auftreten, wenn ein defektes SPS-Programm abgespeichert wurde oder der eigentliche Speicherprozeß gestört wurde. In einem solchen Fall muß das SPS-Programm mit Hilfe des Programmiersystems erneut geladen werden. Die Merker (M), Zeiten (T), Zähler (Z) und S-Merker (S) sind in einem solchen Fall in ihrem Ausgangszustand.

**Laden** Das momentan auf der Festplatte unter **C:\PLCSIM\PLC.BIN** gespeicherte SPS-Programm wird in die *PLC45* geladen. Wurden Merker (M), Zeiten (T), Zähler (Z) und S-Merker (S) mit gespeichert, so werden diese mit in die *PLC45* übernommen. Die Anzeige des *Jitters* wird zurückgesetzt.

**Beim Start Laden** Ist dieses Schaltfeld markiert, so wird das unter **C:\PLCSIM\PLC.BIN** gespeicherte SPS-Programm beim Hochlaufen des Rechners in die *PLC45* geladen. Das Hochlaufen des Rechners entspricht dem Anlauf einer Hardware SPS nach Spannungswiederkehr. Somit wird als erstes der

Organisationsbaustein OB22 abgearbeitet. Erst danach läuft die zyklische Programmbearbeitung mit OB1 an. Wurden Merker (M), Zeiten (T), Zähler (Z) und S-Merker (S) mit gespeichert, so werden diese mit in die PLC45 übernommen.

**Beim Beenden Abspeichern** Ist dieses Schaltfeld markiert, so wird mit dem Herunterfahren der PLC45 das SPS-Programm unter **C:\PLCSIM\PLC.BIN** auf die Festplatte gespeichert.

**M.T.Z.S mit Abspeichern** Ist dieses Schaltfeld markiert, so werden die momentan gültigen Merker (M), Zeiten (T), Zähler (Z) und S-Merker (S) mit dem SPS-Programm unter **C:\PLCSIM\PLC.BIN** auf die Festplatte gespeichert.

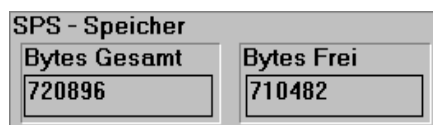
### 3.2.1.8 I/O-Ports

Bei der PLC45 sind die Einstellungen von I/O Ports nicht relevant und haben keinen Einfluß.



### 3.2.1.9 SPS - Speicher

Der für das SPS-Programm vorgesehene Speicher und der noch freie Speicher werden in der Anzahl der Bytes angezeigt.



Die PLC45 hat einen SPS-Programmspeicher von 720 KByte

Das SPS-Programm kann Informationen über den Speicher lesen. Dies ist wichtig für die Funktion der Graph® 5 II Diagnose.

- BS 36 - 37 = Füllstandsanzeiger SPS-Speicher
- BS 32 - 33 = Anfangsadresse des internen RAMs
- BS 34 - 35 = Endadresse des internen RAMs



## 4 Operationsvorrat *PLC45*

---

In den folgenden Abschnitten werden die Unterschiede zwischen der *PLC45* und der Siemens CPU945 beschrieben. Außerdem sind die Befehle der *PLC45* und der interne Speicheraufbau aufgelistet.

---

### 4.1 Erweiterungen der *PLC45*

Folgende zusätzliche Möglichkeiten sind in der *PLC45* realisiert:

- Der PID-Baustein OB251 ist in seiner Ausführung wesentlich schneller (ca. 100  $\mu$ s bei 486-66 Rechner).
- Doppelwort Befehle
- 32 Bit Akkus
- 32768 S-Merker
- FX
- DX
- Multiplikation / Division
- BCD-Wandlung ( DEF,DUF,DED,DUD)
- Schieben mit Vorzeichen (SVW,SVD)
- Sprung bei speicherndem Overflow (SPS=)
- 256 Zeiten und Zähler
- OB19, OB27, OB32, OB250 werden unterstützt bzw. sind integriert.

---

## 4.2 Unterschiede zwischen der *PLC45* und der Siemens CPU945

- „**Alle Bausteine löschen**“ im SPS-Bausteinverzeichnis bewirkt ein Urlöschen.
- **AS, AF**: Die Alarmsperre sperrt alle Interrupts im PC. Während der Alarmsperre findet auch keine Zykluszeitüberwachung statt. Deshalb sollte man die Alarmsperre mit äußerster Vorsicht anwenden.

### Anmerkung:

Eine Endlosschleife innerhalb der Alarmsperre führt zum Hängenbleiben des Rechners !

- Falls die Alarmfreigabe nicht programmiert wurde, wird der Alarm am Ende des OB's automatisch wieder freigegeben.

---

## 4.3 Funktionen, die in der *PLC45* nicht vorhanden sind

Da es physikalische Unterschiede zwischen einer Hardware-SPS und einem Windows-Rechner gibt, können nicht alle Funktionen der CPU945 in der *PLC45* realisiert werden. Es sollten jedoch Ihre vorhandenen Programme mit geringen Änderungen auf der *PLC45* funktionieren.

Die folgenden Funktionen sind nicht in der *PLC45* realisiert:

- Kein Remanenzverhalten von Merkern.
- Es besitzen lediglich folgende Systemdatenwörter eine Funktion :
  - SD 8..12 (Integrierte Uhr)

- SD 96 (Zykluszeitüberwachung)
- SD 97..100 (Zeitintervall für OB10..OB13)
- SD 101 (Zeit bis zum Aufruf des OB6)
- SD 121..123 (Zykluszeit)
- SD 203..238 (Unterbrechungsstack)

Die übrigen Systemdatenwörter werden nicht ausgewertet.

- Die folgenden Organisationsbausteine werden nicht unterstützt bzw. sind nicht integriert: OB160 (Variable Zeitschleife), OB254, OB255.
- Kein OB160
- Zeit-OB's und Interrupt-OB's werden nicht durch Windows unterbrochen, d.h. diese OB's dürfen keine langen Durchlaufzeiten haben (<1ms).
- Der OB6 kann die Zeit-OB's nicht unterbrechen.
- Die folgenden integrierten Funktionsbausteine sind nicht integriert: FB244..FB249 (Hantierungsbausteine), FB 250, FB251.
- Die integrierte Uhr besitzt keine Weck- und Alarmfunktion, keinen Betriebsstundenzähler und der Korrekturwert wird nicht ausgewertet. Der Uhrendatenbereich ist wie in der CPU945 aufgebaut, die Datenbereiche der nicht vorhandenen Funktionen werden nicht ausgewertet.
- Keine Parametrisierung der internen Funktionen über DB1.

## 4.4 Operationsvorrat

### Hinweis:

Bei der *Simulations SPS PLC45* besitzen alle Befehle, die mit absoluten Adressen arbeiten, eine Verarbeitungsbreite von 20 Bit.

Es ist beim Benutzen dieser Anweisungen darauf zu achten, daß die oberen 12 Bit der betroffenen AKKUs gelöscht sind.

### 4.4.1 Verknüpfungsoperationen Binär

#### 4.4.1.1 UND Verknüpfungen mit Abfrage auf Signalzustand 1

Operation	Operanden	Funktionsbeschreibung
<b>U</b>	E 0.0 .. 127.7	Abfrage eines Einganges
<b>U</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>U</b>	M 0.0..255.7	Abfrage eines Merkers
<b>U</b>	S 0.0..4095.7	Abfrage eines S-Merkers
<b>U</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>U</b>	Z 0..255	Abfrage eines Zählers
<b>U</b>	T 0..255	Abfrage einer Zeit
<b>U=</b>	<Parameter>	Abfrage eines FB-Parameters (Typ: BI)

#### 4.4.1.2 UND Verknüpfungen mit Abfrage auf Signalzustand 0

Operation	Operanden	Funktionsbeschreibung
<b>UN</b>	E 0.0 .. 127.7	Abfrage eines Einganges
<b>UN</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>UN</b>	M 0.0..255.7	Abfrage eines Merkers
<b>UN</b>	S 0.0..4095.7	Abfrage eines S-Merkers

UND Verknüpfungen mit Abfrage auf Signalzustand 0

Operation	Operanden	Funktionsbeschreibung
<b>UN</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>UN</b>	Z 0..255	Abfrage eines Zählers
<b>UN</b>	T 0..255	Abfrage einer Zeit
<b>UN=</b>	<Parameter>	Abfrage eines FB-Parameters (Typ: BI)

#### 4.4.1.3 ODER Verknüpfungen mit Abfrage auf Signalzustand 1

Operation	Operanden	Funktionsbeschreibung
<b>O</b>	E 0.0 .. 127.7	Abfrage eines Einganges
<b>O</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>O</b>	M 0.0..255.7	Abfrage eines Merkers
<b>O</b>	S 0.0..4095.7	Abfrage eines S-Merkers
<b>O</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>O</b>	Z 0..255	Abfrage eines Zählers
<b>O</b>	T 0..255	Abfrage einer Zeit
<b>O=</b>	<Parameter>	Abfrage eines FB-Parameters (Typ: BI)

#### 4.4.1.4 ODER Verknüpfungen mit Abfrage auf Signalzustand 0

Operation	Operanden	Funktionsbeschreibung
<b>ON</b>	E 0.0 .. 127.7	Abfrage eines Einganges
<b>ON</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>ON</b>	M 0.0..255.7	Abfrage eines Merkers
<b>ON</b>	S 0.0..4095.7	Abfrage eines S-Merkers
<b>ON</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>ON</b>	Z 0..255	Abfrage eines Zählers
<b>ON</b>	T 0..255	Abfrage einer Zeit
<b>ON=</b>	<Parameter>	Abfrage eines FB-Parameters (Typ: BI)

#### 4.4.1.5 Sonstige Verknüpfungen

Operation	Operanden	Funktionsbeschreibung
<b>O</b>	-	ODER -Verknüpfung von UND-Funktionen
<b>O(</b>	-	UND -Verknüpfung von Klammersausdrücken (8 Klammerebenen möglich)
<b>O(</b>	-	ODER -Verknüpfung von Klammersausdrücken (8 Klammerebenen möglich)
<b>)</b>	-	Abschluß eines Klammersausdruckes
<b>UW</b>	-	AKKU1=AKKU1 UND AKKU2 (16 Bit)
<b>OW</b>	-	AKKU1=AKKU1 ODER AKKU2 (16 Bit)
<b>XOW</b>	-	AKKU1=AKKU1 XOR AKKU2 (16 Bit)

#### 4.4.2 Speicheroperationen Binär

Operation	Operanden	Funktionsbeschreibung
<b>S</b>	E 0.0 .. 127.7	Setzen eines Einganges
<b>S</b>	A 0.0..127.7	Setzen eines Ausganges
<b>S</b>	M 0.0..255.7	Setzen eines Merkers
<b>S</b>	S 0.0..4095.7	Setzen eines S-Merkers
<b>S</b>	D 0.0..255.15	Setzen eines Datenwortes
<b>S=</b>	<Parameter>	Setzen eines FB-Parameters (Typ: BI)
<b>R</b>	E 0.0 .. 127.7	Rücksetzen eines Einganges
<b>R</b>	A 0.0..127.7	Rücksetzen eines Ausganges
<b>R</b>	M 0.0..255.7	Rücksetzen eines Merkers
<b>R</b>	S 0.0..4095.7	Rücksetzen eines S-Merkers
<b>R</b>	D 0.0..255.15	Rücksetzen eines Datenwortes
<b>RB=</b>	<Parameter>	Rücksetzen eines FB-Parameters (Typ: BI)
<b>=</b>	E 0.0 .. 127.7	Zuweisen eines Einganges
<b>=</b>	A 0.0..127.7	Zuweisen eines Ausganges
<b>=</b>	M 0.0..255.7	Zuweisen eines Merkers
<b>=</b>	S 0.0..4095.7	Zuweisen eines S-Merkers
<b>=</b>	D 0.0..255.15	Zuweisen eines Datenwortes
<b>==</b>	<Parameter>	Zuweisen eines FB-Parameters (Typ: BI)

### 4.4.3 Ladeoperationen

#### 4.4.3.1 Laden von Variablen

Operation	Operanden	Funktionsbeschreibung
L	EB 0 .. 127	Laden eines Eingangsbytes
L	AB 0..127	Laden eines Ausgangsbytes
L	MB 0..255	Laden eines Merkerbytes
L	SY 0..4095	Laden eines S-Merkerbytes
L	DL 0..255	Laden eines linken Datenbytes
L	DR 0..255	Laden eines rechten Datenbytes
L	PY 0 ..255	Laden eines Peripheriebytes
L	EW 0 .. 126	Laden eines Eingangswortes
L	AW 0..126	Laden eines Ausgangswortes
L	MW 0..254	Laden eines Merkerwortes
L	SW 0..4094	Laden eines S-Merkerwortes
L	DW 0..254	Laden eines Datenwortes
L	PW 0 ..255	Laden eines Peripheriewortes
L	ED 0 .. 124	Laden eines Eingangsdoublewortes
L	AD 0..124	Laden eines Ausgangsdoublewortes
L	MD 0..252	Laden eines Merkerdoublewortes
L	SD 0..4092	Laden eines S-Merkerdoublewortes
L	DD 0..252	Laden eines Datendoublewortes
L	T 0 ..255	Laden eines Zeitwertes (Dual Codiert)
L	Z 0..255	Laden eines Zählwertes (Dual Codiert)
LC	T 0 ..255	Laden eines Zeitwertes (BCD Codiert)
LC	Z 0..255	Laden eines Zählwertes (BCD Codiert)
L	BS 0 ..255	Laden eines Systemdatenwortes
LIR	0 (AKKU1) 2 (AKKU2)	Akku mit dem Inhalt eines Speicherwortes (durch AKKU1 adressiert) indirekt laden
LDI	A1 (AKKU1) A2 (AKKU2)	Akku mit dem Inhalt eines Speicherwortes (durch 32-Bit-AKKU1 adressiert) indirekt laden

Laden von Variablen

Operation	Operanden	Funktionsbeschreibung
<b>L=</b>	<Parameter>	Laden eines FB-Parameters (Typ: BY,W,D)
<b>LC=</b>	<Parameter>	Laden eines FB-Parameters (BCD Codiert) (Typ: T,Z)
<b>LW=</b>	<Parameter>	Laden eines FB-Parameters (Type: KF,KH,KM,KY,KC,KT,KZ)

#### 4.4.3.2 Laden von Konstanten

Operation	Operanden	Funktionsbeschreibung
<b>L</b>	KB 0 .. 255	Laden der Konstanten in AKKU1 (Bit 0..7) und 0 in Bit 8..15
<b>L</b>	KC 'AA'	Laden von 2 ASCII-Zeichen
<b>L</b>	KF -32768 .. +32767	Laden einer Festpunktzahl
<b>L</b>	KH 0..FFFF	Laden einer Hexadezimalzahl
<b>L</b>	DH 0 .. FFFFFFFF	Laden einer Hexadezimalzahl als Doppelwort
<b>L</b>	KM <Bitmuster 16 Bit>	Laden eines Bitmusters
<b>L</b>	KT 0.0 .. 999.3	Laden einer Zeitkonstanten
<b>L</b>	KY 0 .. 255, 0..255	Laden einer 2-Byte Zahl
<b>L</b>	KZ 0 .. 999	Laden eines Zählwertes

#### 4.4.4 Transferoperationen

Operation	Operanden	Funktionsbeschreibung
<b>T</b>	EB 0 .. 127	Transferieren eines Eingangsbytes
<b>T</b>	AB 0..127	Transferieren eines Ausgangsbytes
<b>T</b>	MB 0..255	Transferieren eines Merkerbytes
<b>T</b>	SY 0..4095	Transferieren eines S-Merkerbytes
<b>T</b>	DL 0..255	Transferieren eines linken Datenbytes

Transferoperationen

Operation	Operanden	Funktionsbeschreibung
<b>T</b>	DR 0..255	Transferieren eines rechten Datenbytes
<b>T</b>	PY 0 ..255	Transferieren eines Peripheriebytes
<b>T</b>	EW 0 .. 126	Transferieren eines Eingangswortes
<b>T</b>	AW 0..126	Transferieren eines Ausgangswortes
<b>T</b>	MW 0..254	Transferieren eines Merkerwortes
<b>T</b>	SW 0..4094	Transferieren eines S-Merkerwortes
<b>T</b>	DW 0..254	Transferieren eines Datenwortes
<b>T</b>	PW 0 ..255	Transferieren eines Peripheriewortes
<b>T</b>	ED 0 .. 124	Transferieren eines Eingangsdoppelwortes
<b>T</b>	AD 0..124	Transferieren eines Ausgangsdoppelwortes
<b>T</b>	MD 0..252	Transferieren eines Merkerdoppelwortes
<b>T</b>	SD 0..4092	Transferieren eines S-Merkerdoppelwortes
<b>T</b>	DD 0..252	Transferieren eines Datendoppelwortes
<b>TIR</b>	0 (AKKU1) 2 (AKKU2)	Akku in ein Speicherwort (durch AKKU1 adressiert) indirekt transferieren
<b>TDI</b>	A1 (AKKU1) A2 (AKKU2)	Akku in ein Speicherwort (durch 32Bit-AKKU1 adressiert) indirekt transferieren
<b>TNB</b>	0..255	Byteweiser Blocktransfer (Anzahl 0..255) Endadresse Quelle: AKKU2 (16 Bit) Endadresse Ziel : AKKU1 (16 Bit)
<b>T</b>	BS 0..255	Transferieren eines Systemdatenwortes
<b>T=</b>	<Parameter>	Transferieren eines FB-Parameters (Typ: BY,W,D)

#### 4.4.5 Zeitoperationen

Operation	Operanden	Funktionsbeschreibung
<b>SI</b>	T 0..255	Eine Zeit als Impuls starten
<b>SV</b>	T 0 ..255	Eine Zeit als verlängerten Impuls starten
<b>SE</b>	T 0..255	Eine Zeit als Einschaltverzögerung starten
<b>SS</b>	T 0 ..255	Eine Zeit als speichernde Einschaltverzögerung starten

Zeitoperationen

Operation	Operanden	Funktionsbeschreibung
-----------	-----------	-----------------------

<b>SA</b>	T 0..255	Eine Zeit als Ausschaltverzögerung starten
<b>R</b>	T 0 ..255	Eine Zeit zurücksetzen
<b>FR</b>	T 0..255	Eine Zeit freigeben
<b>SI=</b>	<Parameter>	Eine Zeit als Impuls starten (FB - Parameter: T)
<b>SVZ=</b>	< Parameter >	Eine Zeit als verlängerten Impuls starten (FB - Parameter: T)
<b>SE=</b>	< Parameter >	Eine Zeit als Einschaltverzögerung starten (FB - Parameter: T)
<b>SSV=</b>	< Parameter >	Eine Zeit als speichernde Einschaltverzögerung starten (FB - Parameter: T)
<b>SAR=</b>	< Parameter >	Eine Zeit als Ausschaltverzögerung starten (FB - Parameter: T)
<b>FR=</b>	< Parameter >	Eine Zeit (Parameter) freigeben (FB - Parameter: T)
<b>RD=</b>	< Parameter >	Rücksetzen eines Parameters für Zeiten und Zähler (FB - Parameter: T)

#### 4.4.6 Zähloperationen

Operation	Operanden	Funktionsbeschreibung
<b>ZV</b>	Z 0..255	Zähler um 1 vorwärts zählen
<b>ZR</b>	Z 0 ..255	Zähler um 1 rückwärts zählen
<b>S</b>	Z 0..255	Zähler setzen
<b>R</b>	Z 0 ..255	Zähler rücksetzen
<b>FR</b>	Z 0 ..255	Einen Zähler freigeben
<b>FR=</b>	< Parameter >	Eine Zeit (Parameter) freigeben (FB - Parameter: Z)
<b>RD=</b>	< Parameter >	Rücksetzen eines Parameters für Zeiten und Zähler (FB - Parameter: T,Z)

#### 4.4.7 Arithmetische Operationen

Operation	Operanden	Funktionsbeschreibung
-----------	-----------	-----------------------

<b>+F</b>	-	AKKU1 = AKKU2 + AKKU1 (16 Bit)
<b>-F</b>	-	AKKU1 = AKKU2 - AKKU1 (16 Bit)
<b>xF</b>	-	AKKU1 = AKKU2 * AKKU1 (16 Bit)
<b>:F</b>	-	AKKU1 = AKKU2 / AKKU1 (16 Bit)
<b>+D</b>	-	AKKU1 = AKKU2 + AKKU1 (32 Bit)
<b>-D</b>	-	AKKU1 = AKKU2 - AKKU1 (32 Bit)
<b>D</b>	0 .. 255	AKKU1 = AKKU2 - Wert (8 Bit)
<b>I</b>	0 .. 255	AKKU1 = AKKU2 + Wert (8 Bit)
<b>ADD</b>	BF -128 .. +127	AKKU1 = AKKU2 + BF Wert (8 Bit) (vorzeichenexpandiert)
<b>ADD</b>	KF -32768..+32767	AKKU1 = AKKU2 +/- Wert (16 Bit)
<b>ADD</b>	DH 0.. FFFFFFFF	AKKU1 = AKKU2 + Wert (32 Bit)

#### 4.4.8 Vergleichsoperationen

Operation	Operanden	Funktionsbeschreibung
<b>!=F</b>	-	VKE = AKKU2 = AKKU1 (16 Bit)
<b>&gt;&lt;F</b>	-	VKE = AKKU2 >< AKKU1 (16 Bit)
<b>&gt;F</b>	-	VKE = AKKU2 > AKKU1 (16 Bit)
<b>&gt;=F</b>	-	VKE = AKKU2 >= AKKU1 (16 Bit)
<b>&lt;F</b>	-	VKE = AKKU2 < AKKU1 (16 Bit)
<b>&lt;=F</b>	-	VKE = AKKU2 <= AKKU1 (16 Bit)
<b>!=D</b>	-	VKE = AKKU2 = AKKU1 (32 Bit)
<b>&gt;&lt;D</b>	-	VKE = AKKU2 >< AKKU1 (32 Bit)
<b>&gt;D</b>	-	VKE = AKKU2 > AKKU1 (32 Bit)
<b>&gt;=D</b>	-	VKE = AKKU2 >= AKKU1 (32 Bit)
<b>&lt;D</b>	-	VKE = AKKU2 < AKKU1 (32 Bit)
<b>&lt;=D</b>	-	VKE = AKKU2 <= AKKU1 (32 Bit)

#### 4.4.9 Bausteinaufrufoperationen

#### 4.4.9.1 Absolute Aufrufe

Operation	Operanden	Funktionsbeschreibung
<b>SPA</b>	PB 0 .. 255	Programmbaustein aufrufen
<b>SPA</b>	FB 0 ..255	Funktionsbaustein aufrufen
<b>BA</b>	FX 0 .. 255	Erweiterten Funktionsbaustein aufrufen
<b>SPA</b>	SB 0 ..255	Schrittbaustein aufrufen
<b>SPA</b>	OB 1 .. 255	Organisationsbaustein aufrufen
<b>B=</b>	<Parameter>	Baustein im Parameter aufrufen

#### 4.4.9.2 Bedingte Aufrufe (wenn VKE=1)

Operation	Operanden	Funktionsbeschreibung
<b>SPB</b>	PB 0 .. 255	Programmbaustein aufrufen
<b>SPB</b>	FB 0 ..255	Funktionsbaustein aufrufen
<b>BAB</b>	FX 0 .. 255	Erweiterten Funktionsbaustein aufrufen
<b>SPB</b>	SB 0 ..255	Schrittbaustein aufrufen
<b>SPB</b>	OB 1 .. 255	Organisationsbaustein aufrufen

#### 4.4.9.3 Datenbausteinoperationen

Operation	Operanden	Funktionsbeschreibung
<b>A</b>	DB 1 .. 255	Datenbaustein aufschlagen
<b>AX</b>	DX 0 ..255	Erweiterten Datenbaustein aufschlagen
<b>E</b>	DB 1 .. 255	Datenbaustein erzeugen
<b>EX</b>	DX 0 ..255	Erweiterten Datenbaustein erzeugen

#### 4.4.10 Bausteinrücksprungoperationen

Operation	Operanden	Funktionsbeschreibung
<b>BE</b>	-	Baustein beenden (Abschließen des Bausteines)

<b>BEB</b>	-	Baustein beenden wenn VKE=1
<b>BEA</b>	-	Baustein absolut beenden

#### 4.4.10.1 Nulloperationen

Operation	Operanden	Funktionsbeschreibung
<b>NOP</b>	0 .. 1	
<b>BLD</b>	0 .. 255	

#### 4.4.11 Stoppoperationen

Operation	Operanden	Funktionsbeschreibung
<b>STP</b>	-	Stopp am Programmende
<b>STS</b>	-	Sofort Stopp

#### 4.4.12 Bitoperationen

##### 4.4.12.1 Abfrage auf Bitzustand = 1

Operation	Operanden	Funktionsbeschreibung
<b>P</b>	E 0.0 .. 127.7	Abfrage eines Einganges
<b>P</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>P</b>	M 0.0..255.7	Abfrage eines Merkers
<b>P</b>	T 0.0..255.15	Abfrage eines Zeitwortes
<b>P</b>	Z 0.0..255.15	Abfrage eines Zählerwortes
<b>P</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>P</b>	BS 0.0..255.15	Abfrage eines Systemdatenwortes

##### 4.4.12.2 Abfrage auf Bitzustand = 0

Operation	Operanden	Funktionsbeschreibung
<b>PN</b>	E 0.0 .. 127.7	Abfrage eines Einganges

<b>PN</b>	A 0.0..127.7	Abfrage eines Ausganges
<b>PN</b>	M 0.0..255.7	Abfrage eines Merkers
<b>PN</b>	T 0.0..255.15	Abfrage eines Zeitwortes
<b>PN</b>	Z 0.0..255.15	Abfrage eines Zählerwortes
<b>PN</b>	D 0.0..255.15	Abfrage eines Datenwortes
<b>PN</b>	BS 0.0..255.15	Abfrage eines Systemdatenwortes

#### 4.4.12.3 Setzen eines Bits

Operation	Operanden	Funktionsbeschreibung
<b>SU</b>	E 0.0 .. 127.7	Setzen eines Einganges
<b>SU</b>	A 0.0..127.7	Setzen eines Ausganges
<b>SU</b>	M 0.0..255.7	Setzen eines Merkers
<b>SU</b>	T 0.0..255.15	Setzen eines Zeitwortes
<b>SU</b>	Z 0.0..255.15	Setzen eines Zählerwortes
<b>SU</b>	D 0.0..255.15	Setzen eines Datenwortes
<b>SU</b>	BS 0.0..255.15	Setzen eines Systemdatenwortes

#### 4.4.12.4 Rücksetzen eines Bits

Operation	Operanden	Funktionsbeschreibung
<b>RU</b>	E 0.0 .. 127.7	Rücksetzen eines Einganges
<b>RU</b>	A 0.0..127.7	Rücksetzen eines Ausganges
<b>RU</b>	M 0.0..255.7	Rücksetzen eines Merkers
<b>RU</b>	T 0.0..255.15	Rücksetzen eines Zeitwortes
<b>RU</b>	Z 0.0..255.15	Rücksetzen eines Zählerwortes
<b>RU</b>	D 0.0..255.15	Rücksetzen eines Datenwortes
<b>RU</b>	BS 0.0..255.15	Rücksetzen eines Systemdatenwortes

#### 4.4.13 Umwandlungsoperationen

Operation	Operanden	Funktionsbeschreibung
-----------	-----------	-----------------------

<b>KEW</b>	-	1er-Komplement (16 Bit)
<b>KZW</b>	-	2er-Komplement (16 Bit)
<b>KZD</b>	-	2er-Komplement (32 Bit)
<b>DEF</b>	-	Festpunkt in BCD wandeln (16 Bit)
<b>DUF</b>	-	BCD in Festpunkt wandeln (16 Bit)
<b>DED</b>	-	Festpunkt in BCD wandeln (32 Bit)
<b>DUD</b>	-	BCD in Festpunkt wandeln (32 Bit)

#### 4.4.14 Schiebe- und Rotieroperationen

Operation	Operanden	Funktionsbeschreibung
<b>SLW</b>	0..15	Nach links schieben (16 Bit)
<b>SRW</b>	0..15	Nach rechts schieben (16 Bit)
<b>SLD</b>	0..32	Nach links schieben (32 Bit)
<b>SVW</b>	0..15	Nach rechts mit Vorzeichen schieben (16 Bit)
<b>SVD</b>	0..32	Nach rechts mit Vorzeichen schieben (32 Bit)
<b>RLD</b>	0..32	Nach rechts rotieren (32 Bit)
<b>RRD</b>	0..32	Nach links rotieren (32 Bit)

#### 4.4.15 Sprungoperationen

Operation	Operanden	Funktionsbeschreibung
<b>SPA=</b>	<Symboladresse>	Unbedingt zur Symboladresse springen
<b>SPB=</b>	<Symboladresse>	Zur Symboladresse springen, wenn VKE=1
<b>SPZ=</b>	<Symboladresse>	Zur Symboladresse springen, wenn Ergebnis=0
<b>SPN=</b>	<Symboladresse>	Zur Symboladresse springen, wenn Ergebnis<>0
<b>SPP=</b>	<Symboladresse>	Zur Symboladresse springen, wenn Ergebnis>0
<b>SPM=</b>	<Symboladresse>	Zur Symboladresse springen, wenn Ergebnis<0
<b>SPO=</b>	<Symboladresse>	Zur Symboladresse springen, wenn Overflow

Sprungoperationen

Operation	Operanden	Funktionsbeschreibung
<b>SPS=</b>	<Symboladresse>	Zur Symboladresse springen, wenn

		speichernder Overflow
<b>SPR</b>	-32768..+32767	Relativer Sprung innerhalb eines Funktionsbausteines

#### 4.4.16 Sonstige Operationen

Operation	Operanden	Funktionsbeschreibung
<b>AS</b>	-	Interrupts sperren
<b>AF</b>	-	Interrupts freigeben
<b>B</b>	DW 0 ..255	Datenwort bearbeiten
<b>B</b>	MW 0 ..255	Merkerwort bearbeiten
<b>BI</b>	-	Operation ausführen, deren Operationscode in einem Parameter abgelegt ist. Die Nummer des Parameters steht im AKKU1
<b>TAK</b>	-	Inhalt von AKKU1 und AKKU2 tauschen

#### 4.4.17 Gleitpunktzahloperationen

Die folgenden CPU 945 kompatiblen Gleitpunktzahloperationen sind in der *PLC45* vorhanden.

##### 4.4.17.1 Umwandeloperationen

FDG	Wandlung Festpunktzahl nach Gleitpunktzahl
GFD	Wandlung Gleitpunktzahl nach Festpunktzahl

##### 4.4.17.2 Ladeoperationen

L KG	Lade Konstante (Gleitpunktzahl) in AKKU1 laden
------	--

##### 4.4.17.3 Vergleichsoperationen

! = G	Vergleich zweier Gleitpunktzahlen auf gleich	AKKU2 = AKKU1 - >VKE = '1'
> < G	Vergleich zweier Gleitpunktzahlen auf ungleich	AKKU2 >< AKKU1 - >VKE = '1'
> G	Vergleich zweier Gleitpunktzahlen auf größer	AKKU2 > AKKU1 - >VKE = '1'
> = G	Vergleich zweier Gleitpunktzahlen auf größer oder gleich	AKKU2 > = AKKU1 - >VKE = '1'
< = G	Vergleich zweier Gleitpunktzahlen auf kleiner oder gleich	AKKU2 < = AKKU1 - >VKE = '1'
< G	Vergleich zweier Gleitpunktzahlen auf kleiner	AKKU2 < AKKU1 - >VKE = '1'

#### 4.4.17.4 Arithmetische Operationen

+ G	Addition zweier Gleitpunktzahlen (32 Bit)	AKKU2 + AKKU1 Ergebnis in AKKU1
- G	Subtraktion zweier Gleitpunktzahlen (32 Bit)	AKKU2 - AKKU1 Ergebnis in AKKU1
* G	Multiplikation zweier Gleitpunktzahlen (32 Bit)	AKKU2 * AKKU1 Ergebnis in AKKU1
/ G	Division zweier Gleitpunktzahlen (32 Bit)	AKKU2 / AKKU1 Ergebnis in AKKU1

Die arithmetischen Funktionen verknüpfen den Inhalt von AKKU1 und AKKU2. Das Ergebnis steht danach in AKKU1.

#### 4.4.17.5 Lade- und Transferoperationen

Operation	Operand	Funktion
<b>TNW</b>	BS 0 ... 255 Anzahl der zu transferierenden Worte	<b>Transferiere einen Datenblock</b> (wortweise) (Nummer 0 .. 255) Endadresse Zielbereich: AKKU1 Endadresse Quellbereich: AKKU2
<b>MBR</b>	= ...F FFFF	<b>Lade BR-Register (Basisadreßregister)</b> Das BR-Register wird mit einer 20 Bit Konstanten geladen
<b>ABR</b>		<b>Addiere Offset zum BR-Register</b> Der Offset wird mit 2 multipliziert (Wort-Offset) und das Ergebnis zum Inhalt des BR-Registers addiert. Das Ergebnis steht im BR-Register.
<b>LRW</b>		<b>Lade Wort, das durch das BR-Register + Offset adressiert ist</b> Der Offset wird mit 2 multipliziert (Wort-Offset) und das Ergebnis zum Inhalt des BR-Registers addiert. Das damit adressierte Wort steht dann im AKKU1-L. Das BR-Register wird nicht verändert. Der ursprüngliche Wert aus AKKU1 wird in AKKU2 geladen.
<b>LRD</b>	-32768 .. 32767	<b>Lade Doppelwort, das durch das BR-Register + Offset adressiert ist</b> Der Offset wird mit 2 multipliziert (Wort-Offset) und das Ergebnis zum Inhalt des BR-Registers addiert. Das damit adressierte Doppelwort steht dann im AKKU1-. Das BR-Register wird nicht verändert. Der ursprüngliche Wert aus AKKU1 wird in AKKU2 geladen.

Lade- und Transferoperationen

<b>TRW</b>		<p><b>Transferiere den Inhalt von AKKU1-L in das Wort, das mit dem BR-Register plus Offset adressiert ist.</b> Der Offset wird mit 2 multipliziert (Wort-Offset) und das Ergebnis zum Inhalt des BR-Registers addiert. Der Inhalt von AKKU1-L wird in das damit adressierte Wort transferiert.</p> <p>Das BR-Register wird nicht verändert.</p>
<b>TRD</b>	-32768 ... 32767	<p><b>Transferiere den Inhalt von AKKU1 in das Doppelwort, das mit dem BR-Register plus Offset adressiert ist.</b> Der Offset wird mit 2 multipliziert (Wort-Offset) und das Ergebnis zum Inhalt des BR-Registers addiert. Der Inhalt von AKKU1 wird in das damit adressierte Doppelwort transferiert.</p> <p>Das BR-Register wird nicht verändert.</p>
<b>MAS</b>		<p><b>Transferiere den Inhalt von AKKU1 in den SAZ (Stepadreßzähler)</b></p> <p>Der Inhalt von AKKU1 (Bit 31 bis 20 von AKKU1 müssen Null sein) wird in den Stepadreßzähler (SAZ) transferiert.</p>
<b>MAB</b>		<p><b>Transferiere den Inhalt von AKKU1 in das BR-Register</b></p> <p>Der Inhalt von AKKU1 (Bit 31 bis 20 von AKKU1 müssen Null sein) wird in das Basisadreßregister (BR) transferiert.</p>
<b>MSA</b>		<p><b>Transferiere den Inhalt des SAZ (Stepadreßzähler) in den AKKU1</b></p> <p>Der Inhalt des Stepadreßzählers (SAZ) wird in den AKKU1 transferiert (Adresse des nächsten Befehls). Dabei werden Bit 31 bis 20 von AKKU1 auf Null gesetzt.</p>
<b>MSB</b>		<p><b>Transferiere den Inhalt des SAZ (Stepadreßzähler) in das BR-Register</b></p> <p>Der Inhalt des Stepadreßzählers (SAZ) wird in das BR-Register transferiert (Adresse des nächsten Befehls).</p>

<b>MBA</b>		<b>Transferiere den Inhalt des BR-Register in den AKKU1</b> Der Inhalt des Basisadreßregisters (BR) wird in den AKKU1 transferiert. Dabei werden Bit 31 bis 20 von AKKU1 auf Null gesetzt.
<b>MBS</b>		<b>Transferiere den Inhalt des BR-Register in den SAZ (Stepadreßzähler)</b> Der Inhalt des Basisadreßregisters (BR) wird in den Stepadreßzähler (SAZ) transferiert.

**Anmerkung:**

Die Lade- und Transferoperationen verändern nicht die nicht benutzten Register.

Werden die Befehle MAS oder MBS genutzt, so wird als nächster Befehl der ausgeführt, dessen Adresse im AKKU bzw. im Basisadreßregister (BR) steht.

#### 4.4.18 Vier Akkumulatoren (CPU 948 kompatibel)

Mit der PLC45 besteht die Möglichkeit vier (4) Akkumulatoren zu benutzen. Die Operationen mit den vier Akkumulatoren sind zu der Siemens CPU 948 kompatibel. Mit dem Systemdatenwort BS 60 wird zwischen der Nutzung von zwei bzw. vier Akkumulatoren umgeschaltet.

◆ Vier (4) Akkumulatoren Operationen

L KF1 ;Lade Konstante (1)

T BS60 ;Transfer in das Systemdatenwort BS 60

◆ Zwei (2) Akkumulatoren Operationen

L KF0 ;Lade Konstante (0)

T BS60 ;Transfer in das Systemdatenwort BS 60

#### 4.4.18.1 Arithmetische Operationen (vier Akkumulatoren Operationen)

+ G	Addition zweier Gleitpunktzahlen (32 Bit)	AKKU2 + AKKU1 Ergebnis in AKKU1
- G	Subtraktion zweier Gleitpunktzahlen (32 Bit)	AKKU2 - AKKU1 Ergebnis in AKKU1
* G	Multiplikation zweier Gleitpunktzahlen (32 Bit)	AKKU2 * AKKU1 Ergebnis in AKKU1
/ G	Division zweier Gleitpunktzahlen (32 Bit)	AKKU2 / AKKU1 Ergebnis in AKKU1

Die arithmetischen Funktionen verknüpfen den Inhalt von AKKU1 und AKKU2. Das Ergebnis steht danach in AKKU1.

Der Inhalt der Akkumulatoren ändert sich wie folgt:

Inhalt	[AKKU1]	[AKKU2]	[AKKU3]	[AKKU4]
Vor der Operation	<AKKU1>	<AKKU2>	<AKKU3>	<AKKU4>
Nach der Operation	<Ergebnis>	<AKKU3>	<AKKU4>	<AKKU4>

#### 4.4.18.2 Registeroperationen (vier Akkumulatoren Operationen)

Die Akkumulatoren AKKU3 und AKKU4 können als Zwischenspeicher für komplexe arithmetische Operationen genutzt werden. Der Befehl ENT (Eingabe in arithmetischen Zwischenspeicher) wird zum Laden der Akkumulatoren AKKU3 und AKKU4 genutzt.

ENT      Eingabe in arithmetischen Zwischenspeicher AKKU3 und AKKU4

Der Inhalt der Akkumulatoren ändert sich wie folgt:

Inhalt	[AKKU1]	[AKKU2]	[AKKU3]	[AKKU4]
Vor ENT Operation	<AKKU1>	<AKKU2>	<AKKU3>	<AKKU4>
Nach ENT Operation	<AKKU1>	<AKKU2>	<AKKU2>	<AKKU3>

Der Inhalt von AKKU1 und AKKU2 wird nicht verändert. Der ursprüngliche Inhalt von AKKU4 geht verloren..

Arithmetische Operationen werden benutzt, um den Inhalt der Akkumulatoren AKKU4 und AKKU3 zu speichern.

## 4.5 Aufbau des SPS-Speichers der *PLC45*

Der Speicher der *PLC45* gleicht im wesentlichen dem Speicher der 115U CPU945.

Beachten Sie, daß bei einem Zugriff von Windows-Programmen auf diesen Speicher das niederwertige mit dem höherwertigen Byte getauscht werden muß.

### Aufbau SPS-Speicher *PLC45*:

Adresse	Länge	Beschreibung
0000	60416	reserviert
EC00	256	Zeiten 0 ... 127 *)
ED00	256	Zähler 0 ... 127 *)
EE00	256	Merker *)
EF00	128	Eingangsabbild *)
EF80	128	Ausgangsabbild *)
F000	256	Zeiten 128 ... 255 *)
F100	256	Zähler 128 ... 256 *)
F200	3584	reserviert
10000	720896	Bausteinspeicher

Aufbau SPS-Speicher *PLC45*:

C0000	131072	reserviert
E0000	4096	S-Merker
E1000	512	Systemdaten
E1200	7680	reserviert
E3000	512	OB - Adreßliste
E3200	512	FB - Adreßliste
E3400	512	PB - Adreßliste
E3600	512	SB - Adreßliste
E3800	512	DB - Adreßliste
E3A00	512	FX - Adreßliste
E3C00	512	DX - Adreßliste
E3E00	115200	reserviert

**\*) Anmerkung:**

Um bei der *PLC45* einen Programmspeicher von 720kByte zu erreichen, war es notwendig die Speicherbelegung der *PLC45* gegenüber der Simatic® CPU945 geringfügig zu ändern. Bei der CPU945 befinden sich die Zähler ab Adresse 20000, die Zeiten ab Adresse 20200, die Merker ab Adresse 20400 und das Prozessabbild ab Adresse 20500.

---

## 4.6 Aufbau der Systemdaten *PLC45*

Der Aufbau der Systemdaten gleicht im wesentlichen dem Aufbau der 115U CPU945.

Die nicht aufgeführten Systemdaten sind nicht belegt und für zukünftige Versionen reserviert.

**Aufbau der Systemdaten *PLC45*:**

Adresse	System - Datenwort	Bit	Beschreibung
EA08	4	8	Stopp am Ende der Programmbearbeitung
EA0A	5	10 11 12 13	Komprimieren abgebrochen Adreßlistenaufbau Bausteinschieben aktiv Bausteinschieben angefordert
EA0C	6	2 13 14 15	Alarmfreigabe SPS im Neustart Stopp - Anzeige Stopp-Zustand über PG
EA0E	7	5 6 13	Neustart nicht möglich Synchronisierfehler (Bausteine sind nicht in Ordnung) Bausteinkopf ist nicht interpretierbar
EA10.. EA16	8.. 11		Echtzeituhr
EAC0	96		Zykluszeitüberwachung (Vielfaches von 10ms)
EAC2	97		Zeitintervall für OB13 (Vielfaches von 10ms)
EAC4	98		Zeitintervall für OB12 (Vielfaches von 10ms)
EAC6	99		Zeitintervall für OB11 (Vielfaches von 10ms)
EAC8	100		Zeitintervall für OB10 (Vielfaches von 10ms)
EACA	101		Zeit (in ms) bis zum Aufruf des OB6
EAF2	121		aktuelle Zykluszeit (in ms)
EAF4	122		maximale Zykluszeit (in ms)
EAF6	123		minimale Zykluszeit (in ms)
EBF0.. EBFF	248.. 255		für Anwender reserviert

Die folgenden Systemdaten sind nur gültig, wenn die SPS sich in Stopp befindet:

**Aufbau der Systemdaten PLC45:**

Adresse	System -	Bit	Beschreibung
---------	----------	-----	--------------

	Datenwort		
EB00	128		AKKU1 (Höherwertiges Wort)
EB02	129		AKKU2 (Höherwertiges Wort)
EB96	203		AKKU1 (Niederwertiges Wort)
EB98	204		AKKU2 (Niederwertiges Wort)
EB9A	205		Step-Adreßzähler (Höherwertiges Wort)
EB9C	206		Step-Adreßzähler (Niederwertiges Wort)
EB9E	207		Bausteinstack
EBA0	208		Anfangsadresse Datenbaustein
EBA2.. EBA8	209.. 212		Klammerstack
EBAA	213	0 1 3 4 5 6 7	SPS Status: Erstabfrage VKE Oder-Aktiv Speichernder Overflow Overflow Anz 0 Anz 1
EBAC	214	2 3 4 8 9 10 11 12 13	Unterbrechungsanzeigewort: Peripherie unklar Kein Neustart möglich (Urlöschen erforderlich) Zykluszeitüberschreitung Fehler in Selbsttestroutine Bausteinstacküberlauf Softwarestopp durch Anweisung Nicht dekodierter Befehl Transferfehler Substitutionsfehler

## 4.7 In der PLC45 integrierte Organisationsbausteine (OB)

OB 19	OB19 wird ausgeführt, wenn ein nicht geladener Baustein aufgerufen wird.
OB 27	OB27 wird ausgeführt, wenn ein Fehler bei einer Substitutionsanweisung auftritt.
OB 31	Zyklusüberwachungszeit neu starten.
OB 32	OB32 wird ausgeführt, wenn ein Fehler bei einer Transferanweisung auftritt.
OB 250	Betriebssystemdienste
OB 251	PID - Regelalgorithmus

#### 4.7.1 OB 19 Reaktion bei Aufruf eines nicht geladenen Bausteins

In dem Organisationsbaustein OB19 kann programmiert werden, wie die *PLC45* sich verhalten soll, wenn ein nicht geladener Baustein vom SPS Programm aufgerufen wird. Ist OB19 nicht programmiert, so wird bei dem Aufruf eines nicht vorhandenen Bausteins das SPS Programm direkt nach der Sprunganweisung fortgesetzt (keine Reaktion der *PLC45*).

OB19 kann z.B. bei entsprechender Programmierung zum Stoppen der *PLC45* genutzt werden.

#### 4.7.2 OB 27 Reaktion bei einem Substitutionsfehler

Ein Substitutionsfehler (SUF) kann auftreten, wenn nach dem Aufruf eines Funktionsbausteins (SPA FBx, SPB FBx) seine Formalparameter geändert werden.

Tritt ein Substitutionsfehler auf, so geht die *PLC45* normalerweise in den Stoppzustand.

Ist der Organisationsbaustein OB27 vorhanden, so wird OB27 abgearbeitet und die *PLC45* geht nicht in den Stoppzustand.

#### 4.7.3 OB 31 Zyklusüberwachungszeit neu starten

Die *PLC45* hat eine integrierte Zykluszeitüberwachung. Wird die eingestellte maximale Zykluszeit (Systemdatenwort BS96) überschritten, so geht die *PLC45* in den Stoppzustand.

Durch Aufruf des Organisationsbausteins OB32 (SPA OB31) kann an jeder beliebigen Stelle im SPS Programm die Zyklusüberwachungszeit neu gestartet werden.

#### **4.7.4 OB 32 Reaktion bei einem Transfer- / Ladebefehlfehler**

Tritt ein Transfer- / Ladefehler (TRAF) auf, so geht die *PLC45* normalerweise in den Stoppzustand.

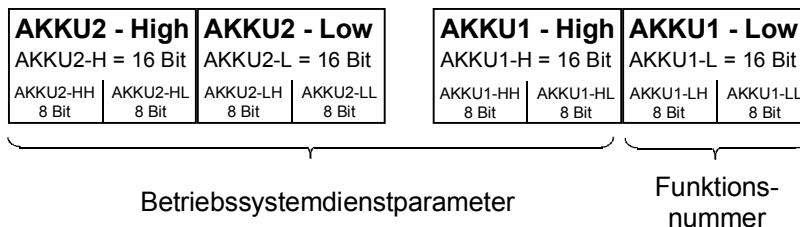
Ist der Organisationsbaustein OB32 vorhanden, so wird OB32 abgearbeitet und die *PLC45* geht nicht in den Stoppzustand.

#### **4.7.5 OB 250 Betriebssystemdienste**

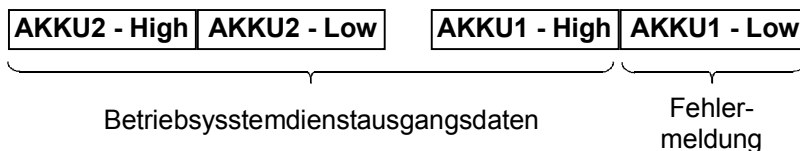
Wird der Organisationsbaustein OB250 (Betriebssystemdienste) aufgerufen, so können verschiedene Betriebssystemfunktionen aktiviert werden oder es können bestimmte Parameter während des zyklischen SPS Programmablaufs geändert werden.

Eine Funktionsnummer ist jeder Betriebssystemdienstefunktion zugeordnet. Vor dem Aufruf des OB250 muß die Funktionsnummer in den AKKU1 (niederwertiges Byte) geladen werden.

Die Betriebssystemdienstparameter sind in den AKKU2 und /oder AKKU1 (höherwertiges Byte) zu laden.



Fehlermeldungen werden im AKKU1 (niederwertiges Byte) geladen. Gleichzeitig sind die Betriebssystemdienstaussgangsdaten in den AKKU2 und /oder AKKU1 (höherwertiges Byte) vorhanden.



Die folgenden Funktionen sind in dem OB250 der *PLC45* integriert:

Betriebssystemdienst	Funktionsnummer	Parameter	Fehlermeldung
Aktivieren von OB 6	1	AKKU2-L: 0: Aktivierung abbrechen 1 ... 65535: Zeitwert (msec)	AKKU1-L: 0: kein Fehler
Neues Zeitintervall für OB 10	2	AKKU2-L: 0: Kein OB 10 Aufruf 1 ... 65535: Zeitintervall (msec)	AKKU1-L: 0: kein Fehler
Neues Zeitintervall für OB 11	3	AKKU2-L: 0: Kein OB 11 Aufruf 1 ... 65535: Zeitintervall (msec)	AKKU1-L: 0: kein Fehler
Neues Zeitintervall für OB 12	4	AKKU2-L: 0: Kein OB 12 Aufruf 1 ... 65535: Zeitintervall (msec)	AKKU1-L: 0: kein Fehler

Betriebssystemdienst	Funktionsnummer	Parameter	Fehlermeldung
Neues Zeitintervall für OB 13	5	AKKU2-L: 0: Kein OB 13 Aufruf 1 ... 65535 Zeitintervall (msec)	AKKU1-L: 0: kein Fehler
Erzeugung eines Datenbausteins (DBx) ohne TRAF	10	AKKU2-LL: 0 ... 255: DB Nummer AKKU2-H: 0: DB löschen 1 ... FFF9: DB Länge bis einschließlich DW	AKKU1-L: 0: kein Fehler 5: TRF (identisch mit EDB)
Erzeugung eines Datenbausteins (DXx) ohne TRAF	11	AKKU2-LL: 0 ... 255: DX Nummer AKKU2-H: 0: DX löschen 1 ... FFF9: DX Länge bis einschließlich DW	AKKU1-L: 0: kein Fehler 5: TRF (identisch mit EXDX)

#### 4.7.6 OB 251 PID - Regelalgorithmus

In der *PLC45* ist ein PID – Regelalgorithmus integriert. Die Bearbeitungszeit des integrierten PID – Regelalgorithmus ist in etwa 100 mal schneller als in der vergleichbaren Hardware SPS. Um die Genauigkeit des Regelalgorithmus zu erhöhen, benutzt die *PLC45* für die Berechnungen Zahlenwerte im Format **REAL**.

Vor dem Aufruf von OB251 muß ein Datenbaustein (Regler-DB), der die Reglerparameter und die Reglerdaten enthält, geöffnet werden. Der PID – Regelalgorithmus wird zyklisch aufgerufen (zeitgleiches Intervall), um die neue Stellgröße zu berechnen.



## 5 DDE Manager

---

Bei der Installation der *PLC45* werden der *PLC43\_45 DDE Manager* und der *PLC43\_45 NetDDE Manager* eingerichtet. Diese Programme können wahlweise im Hintergrund mitlaufen.

Die *DDE Manager* arbeitet völlig unabhängig von *S5 für Windows*. Sie sind jedoch in der Funktion kompatibel.

Über den *DDE Manager* können Sie von anderen Programmen beliebiger Hersteller auf die *PLC45* zugreifen.

Sollen Daten zwischen der *PLC45* und Windowsapplikationen, die auf dem gleichen Rechner laufen, ausgetauscht werden, so ist das Programm *PLC43\_45 DDE Manager* zu starten.

Sollen Daten von Windowsapplikationen, die auf einem über ein Netzwerk verbundenen Rechner vorhanden sind, mit dem Rechner, auf dem die *PLC45* läuft, ausgetauscht werden, so ist das Programm *PLC43\_45 NetDDE Manager* zu starten.

Die Methode des DDE Zugriffes hängt vom jeweiligen Fremdprogramm ab. Die Beschreibung hierzu entnehmen Sie bitte dem Handbuch der entsprechenden Software, die auf die *PLC45* zugreift

Bei dem *DDE Manager* handelt es sich um einen DDE Server mit den folgenden Funktionen:

Request ()

Poke ()

Advise ()

Unadvise ()

Coldlink und Hotlink

Mit der Funktion 'Request' können Sie einen Wert von der *PLC45* lesen oder über 'Poke' einen Wert in die *PLC45* schreiben. Mit 'Advise' bauen Sie eine Hotlink-Verbindung auf und mit 'Unadvise' wird die Hotlink-Verbindung getrennt. Bei einer

Hotlink-Verbindung werden die Daten bei Änderung automatisch von der *PLC45* übertragen.

Der Servername des DDE Managers ist 'PLC43'. Der Topicname ist 'PLC'.

#### Eingabeformate :

	Eingabeformat	Beispiele
Eingangsbits	En.n	E1.1
Eingangsbytes	En oder Ebn	E1 oder EB3
Eingangsworte	Ewn	EW123
Ausgangsbits	An.n	A1.1
Ausgangsbytes	An oder Abn	A1 oder AB3
Ausgangsworte	Awn	AW123
Merkerbits	Mn.n	M1.1
Merkerbytes	Mn oder MBn	M1 oder MB3
Merkerworte	MWn	MW123
S-Merkerbits	Sn.n	S1.1
S-Merkerbytes	Sn oder SYn	S1 oder SY3
S-Merkerworte	SWn	SW123
Datenworte	DWn,dbn	DW3,12 (DW3 von DB12)
Datenbytes links	DLn,dbn	DL3,12 (DL3 von DB12)
Datenbytes rechts	DRn,dbn	DR3,12 (DR3 von DB12)
Datenworte aus DX	DWn,Xdbn	DW3,X12 (DW3 von DX12)
Datenbytes links aus DX	DLn,Xdbn	DL3,X12 (DL3 von DX12)
Datenbytes rechts aus DX	DRn,Xdbn	DR3,X12 (DR3 von DX12)
Zeiten	Tn	T1
Zähler	Zn	Z5

## 5.1 Beispiel

Lesen des Merkerbytes MB 10 und Darstellen in Microsoft EXCEL:

Schreiben Sie hierzu in eine Zelle folgende Anweisung:

```
=PLC43|PLC!MB10
```

Bei der Anweisung handelt es sich um eine Fernbezugsformel. Diese baut automatisch zur *PLC45* eine Hotlink-Verbindung auf. Jedesmal, wenn sich in der *PLC45* der Wert von MB10 ändert, wird dieser auch in EXCEL angezeigt. Manchmal kann es vorkommen, daß EXCEL die Anweisung nicht annimmt. In diesem Fall geben Sie die Teile der Formel mit einem einfachen Hochkomma ein :

```
='PLC43'|'PLC!'MB10'
```

Beschreiben des Merkerbits M 3.2 über Microsoft EXCEL:

#### **Erstellen eines Makros über EXCEL:**

- Menü 'Einfügen', 'Makro', 'Visual Basic Modul'

#### **Schreiben Sie dann folgendes VB-Makro:**

```
Sub SchreibeDatenIntern(Operand; Wert)
```

```
    Dim SendWert Als Objekt
```

```
    Setze SendWert = ZelleListe(11; 2)
```

```
    SendWert = "S5VAL=" + ZuZnF(Wert)
```

```
    DDEKanal = DDEKanalÖffnen("PLC43"; "PLC")
```

```
    DDESenden DDEKanal; Operand; SendWert
```

```
    DDEKanalSchliessen (DDEKanal)
```

```
Ende Sub
```

```
Sub Hand()
```

```
    SchreibeDatenIntern "M3.2"; 1
```

```
Ende Sub
```

#### **Das gleiche Makro in Visual Basic (English):**

```
Sub WriteToPlc(Operand, Value)
```

```
    DIM SendValue As Object
```

```
Set SendValue = Cells(11, 2)
SendValue = "S5VAL" + CStr(Value)
DDEChannel = DDEInitiate("PLC43"; "PLC")
DDEPoke DDEChannel, Operand, SendValue
DDETerminate (DDEChannel)
```

```
EndSub
```

```
Sub Manual()
```

```
WriteToPlc "M3.2"; 1
```

```
End Sub
```

Um das Makro 'Hand' zu starten, müssen Sie eine Befehlsschaltfläche in Tabelle 1 einrichten.

Öffnen Sie das Fenster Tabelle 1.

Klicken Sie das *Zeichnen* Symbol an. Die *Zeichnen* Werkzeuge werden angezeigt.

Klicken Sie das Symbol *Befehlsschaltfläche* an.

Zeichnen Sie eine *Schaltfläche* (Rechteck) mit der Maus.

In dem geöffneten Dialogfeld *Zuweisen* geben Sie als Makroname/Bezug *Hand* ein.

Schließen Sie das Dialogfeld mit *OK* und speichern Sie die Mappe. Mit der Befehlsschaltfläche können Sie nun das Makro 'Hand' starten.

Durch Starten des Makros 'Hand' wird in den Merker M3.2 der *PLC45* eine 1 geschrieben.

Dem SendWert wurde der String "S5VAL=" vorangestellt. Dies bewirkt speziell bei EXCEL eine schnellere Bearbeitung. Bei anderen Programmen muß dieser String nicht vorangestellt werden.

## 6 Visualisierungsprogramme

---

Von mehreren Visualisierungsprogramm-Herstellern sind Treiber für einen schnellen Direktzugriff von den Visualisierungsprogrammen aus erstellt worden.

---

### 6.1 Schneller Direktzugriff für Visual Basic

Von Visual Basic Version 4.0 aus kann direkt auf die *PLC45* zugegriffen werden. Die dazu benötigte DLL wird mitgeliefert (PLC43.DLL). Es handelt sich hierbei um eine 16 Bit Version.

Zum Aufruf der Funktionen stehen im Unterverzeichnis \DLL\VB4 einige Beispiele. Die Erklärung der Beispiele finden Sie in der Datei README.TXT in diesem Verzeichnis.

---



---

# Anhang

## 1 Null Modem Kabel

---

Sub.-D Gehäuse			Sub.-D Gehäuse	
Buchse			Buchse	
9 Pin	25 Pin		9 Pin	25 Pin
1	-----	----- keine Verbindung -----	1	-----
2	3	-----	3	2
3	2	-----	2	3
4	20	-----	6	6
5	7	-----	5	7
6	6	-----	4	20
7	4	-----	8	5
8	5	-----	7	4
9	-----	----- keine Verbindung -----	9	-----
Gehäuse	Gehäuse	----- Schirm -----	Gehäuse	Gehäuse



# SPS im PC      Software SPS

Die zur Verfügung stehende Software SPS Reihe läßt sich in 3 Zweige aufteilen:

- Kompatibel zur Step® 5 Programmierung
- Kompatibel zur Step® 7 Programmierung
- Gemischte Programmierung in S5 und S7.

Die angegebenen Eigenschaften sowohl was S5 als auch S7 angeht, sind in der kombinierten SPS im PC verwirklicht.

Die Unterschiede werden in der folgenden Übersicht herausgestellt.

Die Bearbeitungszeiten hängen natürlich in erheblichem Umfang vom verwendeten PC-Ausbau ab. Deshalb werden im folgenden zwei verschiedene Prozessorsysteme als Referenz (Einfach- und Doppelprozessor-Board) aufgeführt.

	Windows NT 4.x				Windows 3.1x / 95 /98	
	Kompatibel zu Step® 5		Kompatibel zu Step® 7		Kompatibel zu Step® 5	
	PLC S5-943	PLC S5-945	PLC S7-315	PLC S7-416	PLC S5-943	PLC S5-945
<b>Ladespeicher</b>	-----	-----	100 kByte	einstellbar		
<b>Arbeitsspeicher</b>	48 kByte	720 kByte	100 kByte	einstellbar	48 KByte	720 KByte
<b>Merker</b>	2 048	2 048	2 048	16 348	2 048	2 048
<b>S-Merker</b>	32 768	32 768	-----	-----	32 768	32 768
<b>Zeiten</b>	256	256	128	512	256	256
<b>Zähler</b>	256	256	64	512	256	256
<b>Digital E/A</b>	1024	1024	1024	32 768	1024	1024
<b>Analog E/A</b>	256	256	64	2048	256	256
<b>Bearbeitungszeiten *)</b>						
<b>Pentium II 300 **)</b>	560 µs	560 µs	300 µs	300 µs	560 µs	560 µs
<b>2 x Pentium II 300 MHz Doppelprozessor</b>	180 µs	180 µs	90 µs	90 µs	560 µs	560 µs

\*) für eine Binäranweisung

\*\*\*) Zeiteinstellung : 50% SPS-Zeit, 50% Windows-Zeit. 1 024 gemischte Anweisungen (50% Binär, 50% Digital)