

SIMATIC

PG 702 Programming Device

Manual

This manual has the order number:

6ES7702-0AA01-8BA0

Preface, Contents

Introduction to the PG 702	1
Installing the PG 702	2
Creating a Sample Program	3
Working with the PG 702	4
Editing STL Programs	5
Debugging and Testing the Program	6
Overview of Menus and Instruction Sets	A
ASCII Characters	B
Error Codes	C
Technical Specifications and Hardware	D
PG 702 Troubleshooting Guide	E

Index

C79000-G7076-C709

Release 01

Safety Guidelines

This manual contains notices which you should observe to ensure your own personal safety, as well as to protect the product and connected equipment. These notices are highlighted in the manual by a warning triangle and are marked as follows according to the level of danger:



Danger

indicates that death, severe personal injury or substantial property damage will result if proper precautions are not taken.



Warning

indicates that death, severe personal injury or substantial property damage can result if proper precautions are not taken.



Caution

indicates that minor personal injury or property damage can result if proper precautions are not taken.

Qualified Personnel

The device/system may only be set up and operated in conjunction with this manual. Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:



Warning

This device and its components may only be used for the applications described in the catalog or the technical description, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Trademarks

SIEMENS® and SIMATIC® are registered trademarks of SIEMENS AG.

PG 702, STEP™ 7, and S7™ are trademarks of Siemens AG.

Some of the other designations used in these documents are also registered trademarks; the owners' rights may be violated if they are used by third parties for their own purposes.

Copyright Siemens AG 1998 All rights reserved

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Siemens AG
Bereich Automatisierungs- und Antriebstechnik
Geschäftsgebiet Industrie Automatisierungssysteme
P.O. Box 4848, D- 90327 Nuernberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

© Siemens AG 1998
Technical data subject to change.

Preface

Purpose

The PG 702 is designed for editing on-line programs and must be connected to an S7-200 CPU in order to perform its functions. It can transfer an entire program (for example, a statement list program, CPU configuration, and the data block) from one CPU to another CPU by transferring programs to and from its internal non-volatile flash memory. The PG 702 can edit all three types of memory blocks: program block, data block, and system data block (CPU configuration).

This *PG 702 Programming Device User Manual* provides information about the software features of the PG 702 Statement List version of the hand-held programmer for S7-200 CPUs. It emphasizes the features required to program, edit, and monitor new and existing programs. This manual also specifies how to use diagnostic features and PG menu functions, such as CPU mode changes and system configuration.

Audience

This manual contains entry-level information on programming SIMATIC S7 programmable controllers (S7-200 family) and is also intended for the experienced control engineer.

Scope of the Manual

This manual describes version 2.0 of the PG 702 programming device (order number 6ES7 702 0AA01 0YA0) using the international mnemonics and keyboard overlay.

How to Use This Manual

If you are a first-time user of hand-held programmers, you should read the entire manual. If you are an experienced user, refer to the table of contents or index to find specific information.

The manual includes installation and basic setup information as well as specific guidelines for working with the following three main CPU programming areas:

- CPU Program Block (statement list program)
- CPU Configuration (port configuration, input filters, retentive ranges)
- CPU Data Block (data block)

Additional information (such as technical information and a quick reference list of all menu functions and error codes) is provided in the appendices.

Related Information

Refer to the following documentation for more detailed information about selected topics:

SIMATIC S7-200 Programmable Controller System Manual: provides information about installing and programming the S7-200 Micro PLCs, including the following topics:

- Installing and wiring the S7-200 CPU and expansion I/O modules, and installing the STEP 7–Micro/WIN software
- Designing and entering a program
- Understanding features of the CPU, such as data types and addressing modes, the CPU scan cycle, password-protection, and network communication

This manual also includes descriptions and examples for the programming instructions, typical execution times for the instructions, and the data sheets for the S7-200 equipment.

Additional Assistance

For assistance in answering technical questions, for training on this product, or for ordering, contact your Siemens distributor or sales office.

Contents

1	Introduction to the PG 702	1-1
1.1	Using the PG 702	1-2
1.2	Programming with the PG 702	1-3
2	Installing the PG 702	2-1
2.1	Connecting the PG 702	2-2
2.2	Changing the Keyboard Overlay	2-3
2.3	Mounting the PG 702 in a Panel or Cabinet Door	2-4
3	Creating a Sample Program	3-1
3.1	Example for Powering Up and Setting the Programming Device	3-2
3.2	Uploading a Program from the CPU to the PG 702	3-4
3.3	Sample Task	3-7
4	Working with the PG 702	4-1
4.1	Working with the Keypad	4-2
4.2	Starting Up the PG 702 / Password Protection	4-9
4.3	Working with Menus	4-10
4.4	Operating Modes of the PG 702	4-12
4.5	CPU Program Block	4-14
4.6	CPU Configuration	4-16
4.7	CPU Data Block	4-26
4.8	Flash Operations	4-30
4.9	CPU Functions	4-35
4.10	PG Configuration	4-41
4.11	Passwords	4-47
4.12	Debug Read/Write Variables	4-50

5	Editing STL Programs	5-1
5.1	Using the Edit Functions	5-2
5.2	Setting the Edit Mode and Loading the Program into the CPU	5-3
5.3	Entering Instructions	5-4
5.4	Entering Operands	5-8
5.5	Using the Display for Editing	5-12
5.6	Using Editing Aids	5-15
5.7	GOTO Network or Line Number	5-16
5.8	Find Operand or Instruction	5-17
6	Debugging and Testing the Program	6-1
6.1	Debug Functions	6-2
6.2	Status Chart	6-3
6.3	Forcing and Unforcing Variables and I/O	6-5
6.4	Single Scan Debug Function	6-8
A	Overview of Menus and Instruction Sets	A-1
A.1	Logic Operations	A-2
A.2	Instruction Set Menus	A-6
A.3	Operands	A-13
A.4	PG Menu Functions	A-16
B	ASCII Characters	B-1
C	Error Codes	C-1
D	Technical Specifications and Hardware	D-1
D.1	PG 702 Specifications	D-2
D.2	Programmer Cable (PPI Interface for the PG 702)	D-4
D.3	Handling Electrostatic Sensitive Devices (ESD)	D-5
D.4	Approbation	D-7
E	PG 702 Troubleshooting Guide	E-1
	Index	Index-1

List of Figures

2-1	PG 702 Connections	2-2
2-2	PG 702 Keyboard Overlay for International Mnemonics	2-3
2-3	PG 702 Keyboard Overlay for SIMATIC Mnemonics	2-3
2-4	Mounting the PG 702 in a Panel	2-4
3-1	Display at Start Up	3-2
4-1	PG 702 Keypad and Display	4-2
4-2	Example of a Menu Sequence	4-10
4-3	Defining the Beginning of a Retentive Range	4-17
4-4	Defining the End of a Retentive Range	4-17
4-5	Viewing the Default Operand	4-18
4-6	Sample of I/O Groups	4-20
4-7	DP Status Parameter Screens	4-25
4-8	View of a Data Block Value	4-27
4-9	Sample View of the Read I/O Configuration Display	4-37
4-10	Write Time-of-Day Screens	4-39
5-1	Keys for Entering Instructions	5-4
5-2	Keys for Entering Operand Identifiers and Operand Offsets	5-8
5-3	Sample Instruction Line with Character Spacing	5-12
5-4	Sample Information Line with Supplementary Information	5-13
B-1	Sample ASCII Character Displays	B-3
B-2	Scrolling through the Second ASCII Character Menus	B-3
D-1	PPI Interface	D-4

List of Tables

3-1	Uploading a Program from a CPU	3-4
3-2	Clearing an Existing Program Block and Uploading the Program	3-5
3-3	Using the Function Key to Download a Program to the CPU	3-12
3-4	Using the PG Menu to Download a Program to the CPU	3-12
4-1	Four Ways to Enter Instructions	4-4
4-2	Key Colors and Functions	4-7
4-3	PG 702 Operating Modes	4-12
4-4	CPU Configuration Menu	4-16
4-5	Input Delay Time Values	4-21
4-6	Port 0 Parameter Functions	4-22
4-7	Port 1 Parameter Functions	4-23
4-8	CPU Data Block Functions	4-26
4-9	Key Definitions in the Data Block Editor	4-28
4-10	Flash Operation Functions	4-30
4-11	Compatibility Between CPU Models	4-34
4-12	CPU Functions	4-35
4-13	Valid Keys for Modifying the Time-of-Day Clock	4-38
4-14	PG Configuration Functions	4-41
4-15	PG Menu Functions Available Under Restricted Capability	4-45
4-16	Passwords Function	4-47
4-17	Debug Read/Write Variables	4-50
5-1	Explanations for Positioning the Cursor in the Instruction Line	5-14
5-2	Editing Aids	5-15
5-3	GOTO Results	5-16
6-1	Debug Functions Menu	6-2
6-2	PG Functions Available in Status Chart Setup	6-3
6-3	Valid Keys for Modifying the Force Information	6-5
A-1	AND Instructions	A-2
A-2	OR Instructions	A-3
A-3	Assignment, Set/Reset Instructions	A-3
A-4	Load Instructions	A-4
A-5	Timer/Counter Instructions	A-5
A-6	Label Instructions	A-6
A-7	Return Instructions	A-6
A-8	Interrupt Instructions	A-7
A-9	Miscellaneous Control Instructions	A-7
A-10	Rotate and Shift Instructions	A-8
A-11	Instructions	A-8
A-12	Conversion Instructions	A-9
A-13	Word Logic Instructions	A-9
A-14	Data Move Instructions	A-10
A-15	Stack Instructions	A-10
A-16	Mathematical Instructions	A-11
A-17	Miscellaneous Instructions	A-12
A-18	Operands	A-14
A-19	PG Menu Functions	A-16
B-1	ASCII Characters	B-1

Introduction to the PG 702

This chapter provides you with a general overview of the PG 702. It explains the tasks, functions, and function range of the PG 702.

Chapter Overview

Section	Description	Page
1.1	Using the PG 702	1-2
1.2	Programming with the PG 702	1-3

The PG 702 is a programming device for the SIMATIC S7-200 family of CPUs. It has a two-line display. You can use the PG 702 to modify the program in the CPU, change the CPU configuration, edit the data block of the CPU, store the entire CPU program in the non-volatile flash memory of the PG 702, and change the mode of the CPU. In addition, it performs several auxiliary functions that are detailed in this manual.

1.1 Using the PG 702

The main application areas are as follows:

- Create/modify the statement list program
- Modify the CPU configuration
- Create/modify the CPU data block
- Test your program in an S7-200
- Transfer the entire CPU program to or from the PG 702 non-volatile memory
- Change the mode of the CPU

Mounting the PG 702 permanently in a panel or cabinet door makes it possible for you to install a stationary programming device that is permanently connected to an S7-200 and easily accessible to perform necessary service.

PG 702 Language

You can set the PG 702 to any of the following languages: German, English, French, Italian, and Spanish.

You can use both SIMATIC and international mnemonics.

Functionality

Use the PG 702 to perform the following tasks:

- Create a new program or change an existing program in an S7-200
- Test an existing program in the S7-200
- Manage your programmable controller, such as setting the CPU node address, clearing the CPU, and configuring your programmable controller
- Transfer the CPU program, data, and configuration blocks as a whole from one CPU to another by means of non-volatile memory in the PG 702

1.2 Programming with the PG 702

Creating a User Program

You can use the PG 702 to change an existing program in an S7-200 or to create a new one. If your program is a ladder diagram block, the program is automatically converted to statement list (STL).

Note

All ladder programs can be represented in STL, but not all STL programs can be represented as ladder diagrams. If you want to write or change a program with the PG 702, and you want to view this program later with a ladder editor such as STEP 7-MicroWIN, be sure to construct networks that can be displayed as ladder diagrams. To view the program in ladder, you must divide segments of code with the keyword NETWORK.

Editing an On-Line Program

The PG 702 can edit all three types of memory blocks: CPU program blocks, system data blocks (CPU configuration), and data blocks.

To edit the program block, you must first upload the program block from the CPU. You can then edit the program and download it to the CPU. The download function is disabled unless you have uploaded a program block. If there is no program in the CPU, the PG 702 creates a program block.

Use the data block editor to edit the data block. When you enter the data block editor, the PG 702 determines if the CPU has a data block present. If there is a data block available, it is uploaded. If one is not present, the PG 702 creates one.

The CPU configuration is modified and updated one field at a time.

For more information on the editing functions, see Chapter 5.

Installing the PG 702

2

This chapter shows you how to connect the PG 702 and how to install it in a panel or cabinet door.

Chapter Overview

Section	Description	Page
2.1	Connecting the PG 702	2-2
2.2	Changing the Keyboard Overlay	2-3
2.3	Mounting the PG 702 in a Panel or Cabinet Door	2-4

2.1 Connecting the PG 702

Programmable Controller Connection

Connect the PG 702 to the PPI interface on the CPU using the supplied programmer cable at the programmer cable port (see Figure 2-1). This cable supplies the programming device with the necessary power. The programming device can be operated only when it is connected to an S7-200 CPU.

External Power Supply Connection

To connect the PG 702 with a cable longer than the one provided with the programming device (greater than 2.5 m or 8.2 ft.), you must operate the PG 702 with a separate power supply unit (24 VDC, safety extra-low voltage = SELV).

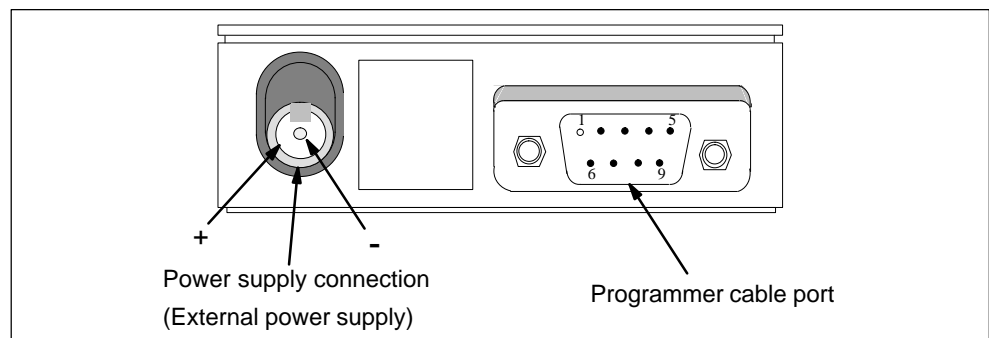


Figure 2-1 PG 702 Connections

2.2 Changing the Keyboard Overlay

The two removable keyboard template overlays shown in Figures 2-2 and 2-3 are provided with the PG 702. Place the mnemonic keyboard overlay you prefer on your PG 702.

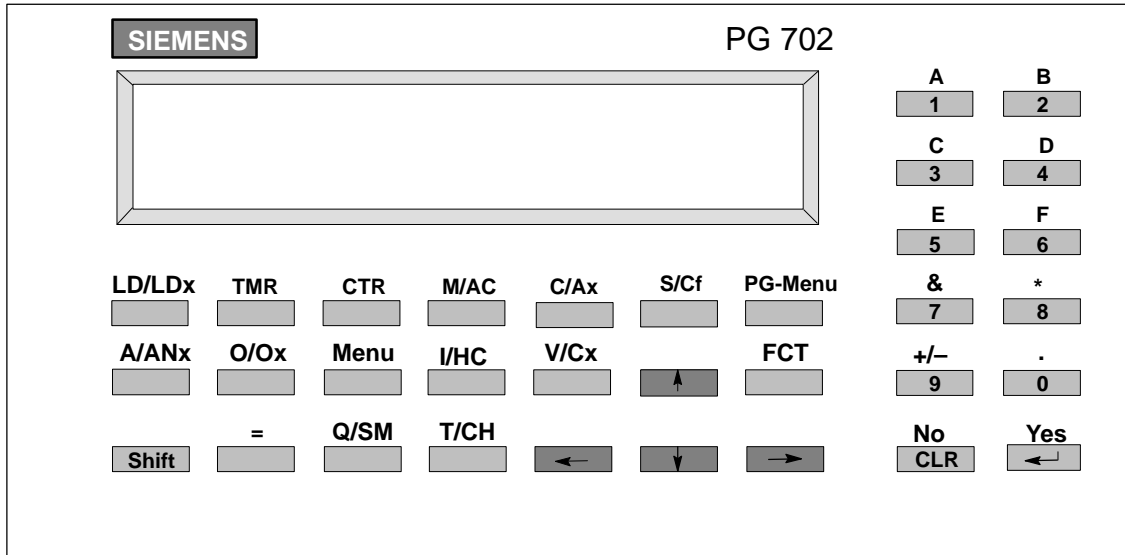


Figure 2-2 PG 702 Keyboard Overlay for International Mnemonics

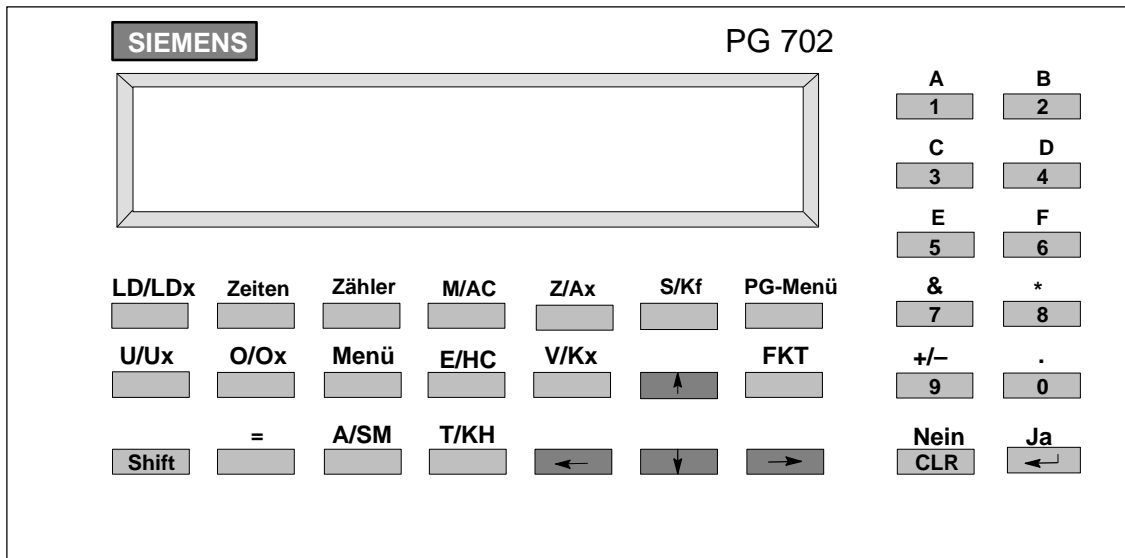


Figure 2-3 PG 702 Keyboard Overlay for SIMATIC Mnemonics

2.3 Mounting the PG 702 in a Panel or Cabinet Door



Warning

Failure to disable all power to the PG 702 and related equipment during installation or removal procedures may result in electrical shock. Electrical shock can cause death or serious personal injury, and/or damage to equipment. Disconnect all power to the PG 702 and related equipment before installation or removal.

The PG 702 is prepared for permanent mounting in a panel or cabinet door. To do this, proceed as follows:

1. Make a standard cutout to DIN 43700 (138 mm x 68 mm/5.2 in. x 2.7 in.) in the panel.
2. Remove the three screws on the rear of the PG 702 (see Figure 2-4).
3. Detach the rear panel.
4. Remove the center piece from the self-adhesive spacer rings supplied and stick the rings over the drilled holes for the screws on the rear of the casing as required for your mounting:
 - No ring for metal thicknesses up to 0.3 mm (0.012 in.)
 - One ring for metal thicknesses of 0.3 mm to 1.5 mm (0.012 in. to 0.06 in.)
 - Two rings for metal thicknesses of 1.5 mm to 4 mm (0.06 in. to 0.16 in.)
5. Guide the front part of the casing into the cutout in the panel.
6. Push the rear panel onto the casing.
7. Fix the PG 702 in position with the screws, and slightly tighten the screws.

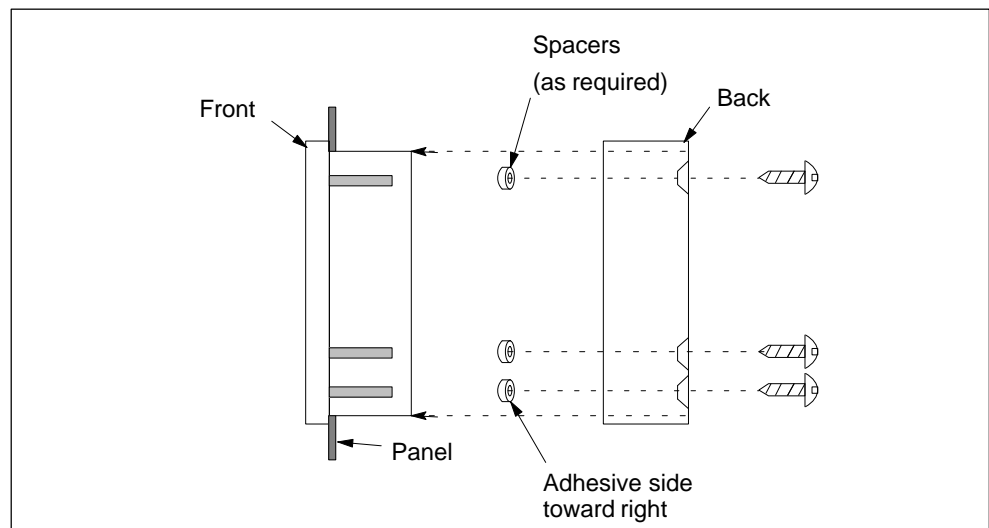


Figure 2-4 Mounting the PG 702 in a Panel

When the PG 702 case is open, take all precautions related to handling electrostatic sensitive devices as described in Section D.3.

3

Creating a Sample Program

This chapter describes how to use the PG 702 to perform the following tasks:

- Powering up and setting the programming device
- Uploading a program to the PG 702
- Performing a sample task

No particular knowledge of special key functions is required to carry out these examples since every operator input is explained.

Chapter Overview

Section	Description	Page
3.1	Example for Powering Up and Setting the Programming Device	3-2
3.2	Uploading a Program from the CPU to the PG 702	3-4
3.3	Sample Task	3-7

3.1 Example for Powering Up and Setting the Programming Device

In this example you are shown how to perform the following tasks:

- Power up the PG 702
- Set the dialog language (English, German, French, Italian, or Spanish)
- Set the instruction mnemonic language (SIMATIC or International)


Connecting the Programming Device to the CPU

Use the cable supplied with your PG 702 to connect the PG 702 to the programmer port on the programmable controller. This cable supplies the programming device with the necessary power.

First Power Up

The PG 702 is ready to operate as soon as you connect it to the programmable controller. The first item displayed is the version of the firmware. This banner is displayed for approximately five seconds and is then replaced by the main PG menu (see Figure 3-1). If the PG 702 reports COMM TIMEOUT, the CPU port address and/or baud rate may not match that expected by the PG 702. Use the FIND CPU ADDRESS feature (see Section 4.10) to locate the correct CPU port address and baud rate.

Confirming Messages

The programming device displays system messages in the lower display line. If an error message is displayed, you must confirm these messages by pressing  before continuing with your inputs.

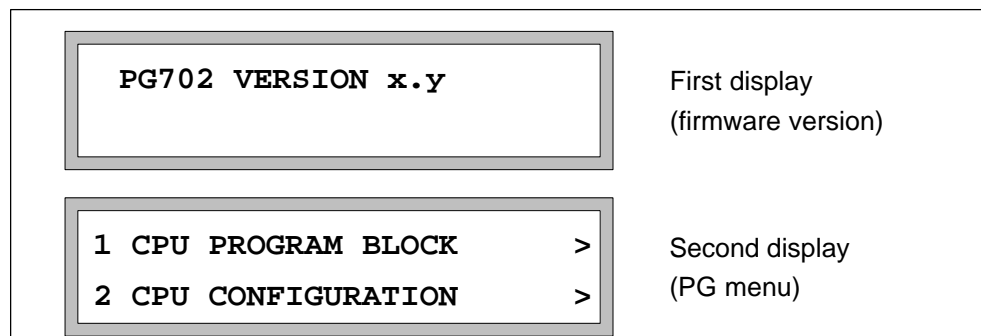


Figure 3-1 Display at Start Up

Note

The PG 702 establishes communication with the CPU before the PG menu is displayed. In a multi-master network, this process can take up to one minute.

Setting the Language

English is set at the factory as the default language. If you wish to change the language, use the following procedure from within the PG menu.

1. Press PG CONFIGURATION
2. Press SET LANGUAGE
3. Press one of the following keys to make your language selection:
 - GERMAN
 - ENGLISH
 - FRENCH
 - ITALIAN
 - SPANISH
4. Press one of the following keys to select the mnemonic (the choice appears after you select the menu language):
 - SIMATIC
 - INTERNATIONAL

The PG 702 stores its configuration in non-volatile memory. Whatever language is chosen now becomes the default.

Useful Keys When Entering the Examples

If you make an error or encounter a problem while you are working on the examples on the following pages, these keys are helpful:

- is the “Yes” key. Press this key to acknowledge an error message and clear the display.
- Press to return to a higher-level menu on the PG 702.
- Press to undo a selection or cancel an operation.
- Press to escape.

3.2 Uploading a Program from the CPU to the PG 702

In this example you are shown how to perform the following tasks:

- Upload a program to the PG 702
- Create a new program in the PG 702

Uploading an Existing Program

From the main PG menu, use the steps shown in Table 3-1 to upload a program from the CPU.

Table 3-1 Uploading a Program from a CPU

	Key	Function	Display
1.	PG-Menu	Access the main PG menu if it is not already displayed	<pre> 1 CPU PROGRAM BLOCK > 2 CPU CONFIGURATION > </pre>
2.	1	Activate CPU PROGRAM BLOCK menu	<pre> 1 UPLOAD FROM CPU 2 DOWNLOAD TO CPU </pre>
3.	1	Upload the program from the CPU	<pre> █ ED: 1:1 </pre>

If the program contained in the CPU is empty, the display matches that shown in Step 3 in Table 3-1. If you upload a program that already contains instructions, the first instruction line is displayed.

If the PG 702 already contains a program, the message “REPLACE PROGRAM?” is displayed. Press **←** for “yes” or **CLR** for “no.”

Creating a New Program

If you upload a program that contains instructions, the PG 702 shows the first instruction of your program in the first line of the display. For example:

```

LD    I0.0
ED:           1:1
                    
```

To create a new program, you must clear the program block in the CPU and upload the empty program block to the PG 702. Use the steps shown in Table 3-2 to clear the program block and upload it to the PG 702.

Table 3-2 Clearing an Existing Program Block and Uploading the Program


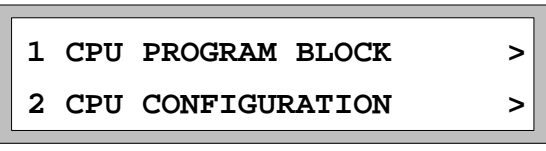

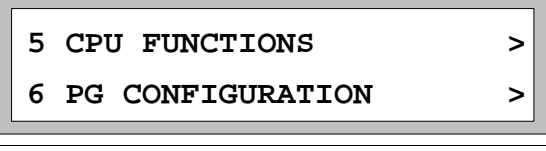

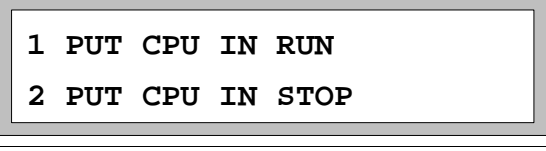

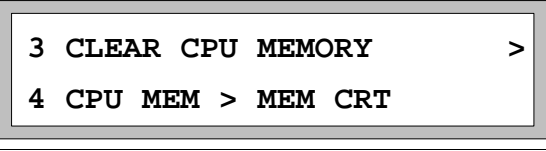

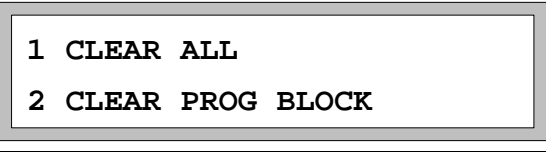

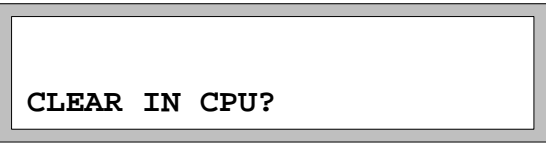

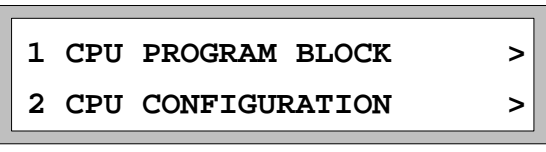

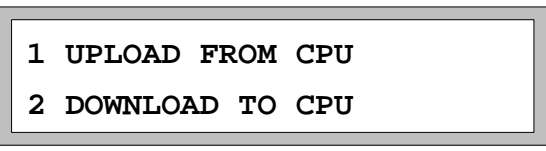



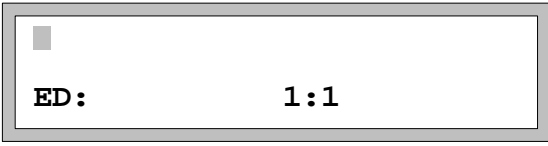
	Key	Function	Display
1.		Return to the PG menu	
2.	Repeatedly 	Display CPU FUNCTIONS	
3.		Select the CPU FUNCTIONS menu	
4.		Scroll through the CPU functions	
5.		Select CLEAR CPU memory	
6.		Select CLEAR PROG BLOCK	
7.		Confirm your selection to clear the CPU program block memory and return to the PG menu.	
8.		Activate CPU PROGRAM BLOCK menu	

Table 3-2 Clearing an Existing Program Block and Uploading the Program

	Key	Function	Display
9.		Upload the program from the CPU. If your PG 702 contains a program, the "REPLACE PROGRAM?" message appears.	
10.		Confirm your selection to upload the empty program from the CPU to the PG 702.	

You can now create a new program as shown in the example on the following pages.

3.3 Sample Task

This sample program cyclically increments a memory word (MW1). Only the contents of the higher-order memory byte (MB1) are forwarded to the outputs of the programmable controller. The incrementing of MB2 (lower-order byte) is not relevant at that point. As soon as the maximum value is reached, the procedure starts at the beginning again.

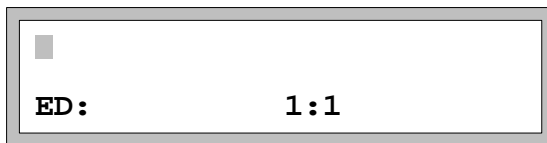
Sample Instruction Sequence

The entire, executable program consists of the following instructions:

```
NETWORK 1
LD  SM0.0      //Activate the top of stack
+l  #1,MW1     //Count up
MOVB MB1,QB0   //Write to output
NETWORK 2
MEND           //Program end
```

The Edit Display

Once you have uploaded an empty program from the CPU, the PG 702 displays the following:



Entering the First Instruction Line

Use the following steps to enter the first instruction.

	Key	Function	Display
1.	LD/LDx	Select LD instruction	The cursor jumps to the entry field for operands.
2.	Shift Q/SM	Select SM operand identifier	The cursor jumps to the entry field for the operand offset.
3.	0 → 0	Input operand offset 0.0: 0 Use the right arrow key to insert the "." as a delimiter 0	<p>Network no. Instruction no.</p> <p>LD SM0.0</p> <p>ED: 1:1</p>
4.	←	Terminate the input and move the cursor to the next line	<p>ED: 1:2</p>

Entering the Second Instruction Line

The cursor is at the entry field for the instruction. Now enter the second instruction.



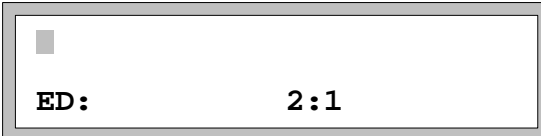
	Key	Function	Display
1.	Menu	Activate the menu for instructions	<div style="border: 1px solid gray; padding: 5px;"> <p>1 PROGRAM CONTROL ></p> <p>2 ROTATE & SHIFT ></p> </div>
2.	Repeatedly ↓	Scroll through the instructions menu	<div style="border: 1px solid gray; padding: 5px;"> <p>7 STACK INSTRUCTNS ></p> <p>8 MATH INSTRUCTNS ></p> </div>
	8	Select the MATH INSTRUCTIONS menu	<div style="border: 1px solid gray; padding: 5px;"> <p>1 + ></p> <p>2 - ></p> </div>
3.	1	Select + instructions	<div style="border: 1px solid gray; padding: 5px;"> <p>1 +I</p> <p>2 +D</p> </div>
	1	Select +I instruction	<div style="border: 1px solid gray; padding: 5px;"> <p>+I █</p> <p>ED: 1:2</p> </div>
4.	Shift V/Cx	Select first constant identifier #	<div style="border: 1px solid gray; padding: 5px;"> <p>+I #1</p> <p>ED: 1:2</p> </div>
	1	Enter 1 as the value of the constant	
5.	→ M/AC	Enter second operand MW. The comma is generated automatically	<div style="border: 1px solid gray; padding: 5px;"> <p>+I #1,MW1</p> <p>ED: 1:2</p> </div>
	1	Enter 1 as the operand offset	
6.	←	Terminate the input and move the cursor to the next line	<div style="border: 1px solid gray; padding: 5px;"> <p>█</p> <p>ED: 1:3</p> </div>

Entering the Third Instruction Line

The cursor is at the entry field for the instruction. Now enter the third instruction.


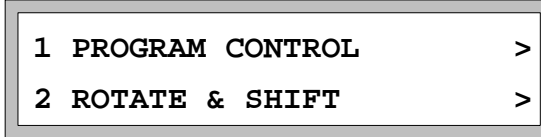


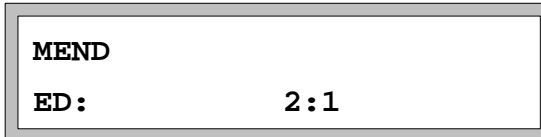



	Key	Function	Display
1.	Menu	Activate the menu for instructions	<div style="border: 1px solid gray; padding: 5px;"> <p>1 PROGRAM CONTROL ></p> <p>2 ROTATE & SHIFT ></p> </div>
2.	Repeatedly ↓ 6	Scroll through the instructions menu. Select the DATA MOVES menu	<div style="border: 1px solid gray; padding: 5px;"> <p>5 WORD WIDE LOGIC ></p> <p>6 DATA MOVES ></p> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> <p>1 MOVB</p> <p>2 MOVW</p> </div>
3.	1	Select MOVB	<div style="border: 1px solid gray; padding: 5px;"> <p>MOVB</p> <p>ED: 1:3</p> </div>
4.	M/AC 1	Enter the first operand MB Enter 1 as the operand offset	<div style="border: 1px solid gray; padding: 5px;"> <p>MOVB MB1</p> <p>ED: 1:3</p> </div>
5.	→ Q/SM 0	Enter second operand QB. The comma is generated automatically Enter 0 as the operand offset	<div style="border: 1px solid gray; padding: 5px;"> <p>MOVB MB1, QB0</p> <p>ED: 1:3</p> </div>
6.	←	Terminate the input and move the cursor to the next line	<div style="border: 1px solid gray; padding: 5px;"> <p>█</p> <p>ED: 1:4</p> </div>

Changing to Network 2

	Key	Function	Display
1.	 	Begin next network	

Entering the Fourth Instruction Line

The cursor is at the entry field for the instruction. Now enter the fourth instruction.

	Key	Function	Display
1.		Activate the menu	
		Select PROGRAM CONTROL	
	Repeatedly 	Scroll through the program control options	
		Select MEND	
2.		Terminate the input	

Downloading Your Program to the CPU

Now you must download the completed program into CPU memory. You can either use the **FCT** key (see Table 3-3) or you can use the PG menu to save your program to the CPU (see Table 3-4).

Table 3-3 Using the Function Key to Download a Program to the CPU



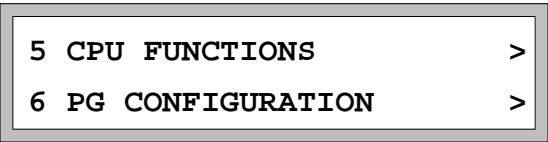
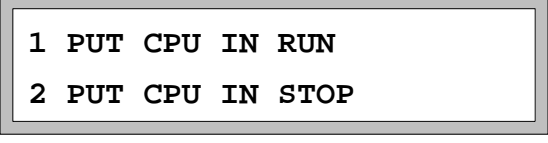


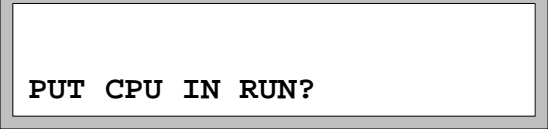
	Key	Function	Display
1.	FCT Repeatedly ↓ 5	Access the editing aids menu Scroll through the editing aids Select DOWNLOAD TO CPU	<div style="border: 1px solid black; padding: 5px;"> <p>1 INSERT ></p> <p>2 DELETE ></p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>4 FIND ></p> <p>5 DOWNLOAD TO CPU</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>LOAD TO CPU?</p> </div>
2.	←	Download program.	Program is stored in the CPU.

Table 3-4 Using the PG Menu to Download a Program to the CPU

	Key	Function	Display
1.	PG-Menu	Select the PG menu	<div style="border: 1px solid black; padding: 5px;"> <p>1 CPU PROGRAM BLOCK ></p> <p>2 CPU CONFIGURATION ></p> </div>
2.	1	Activate CPU PROGRAM BLOCK menu	<div style="border: 1px solid black; padding: 5px;"> <p>1 UPLOAD FROM CPU</p> <p>2 DOWNLOAD TO CPU</p> </div>
3.	2	Download the program to the CPU	<div style="border: 1px solid black; padding: 5px;"> <p>LOADING TO CPU</p> </div>

Starting Your Program

Be sure the switch on your CPU is in the TERM position. Using the PG 702, put the CPU in the RUN mode to start the completed program.

	Key	Function	Display
1.	PG-Menu Repeatedly  	Activate menu for changing the CPU operating mode	 
2.	 	Select RUN mode Confirm your choice	 Now the program should run. If you have problems, make sure that the operating mode switch on the CPU is in the "TERM" position.

Examine the output LEDs on the CPU for QB0. They should be cycling in a binary progression.

4

Working with the PG 702

This chapter shows you how to operate and configure the PG 702. It also gives you general notes about the display format, and how to use the operator interface (menus).

Chapter Overview

Section	Description	Page
4.1	Working with the Keypad	4-2
4.2	Starting Up the PG 702 / Password Protection	4-9
4.3	Working with Menus	4-10
4.4	Operating Modes of the PG 702	4-12
4.5	CPU Program Block (PG-Menu Function 1)	4-14
4.6	CPU Configuration (PG-Menu Function 2)	4-16
4.7	CPU Data Block (PG-Menu Function 3)	4-26
4.8	Flash Operations (PG-Menu Function 4)	4-30
4.9	CPU Functions (PG-Menu Function 5)	4-35
4.10	PG Configuration (PG-Menu Function 6)	4-41
4.11	Passwords (PG-Menu Function 7)	4-47
4.12	Debug Read/Write Variables (PG-Menu Function 8)	4-50

4.1 Working with the Keypad

Keypad Layout

The PG 702 keypad layout, shown in Figure 4-1, is ergonomically designed in its physical arrangement and color. The design is tailored to the operator's requirements in the various operating modes.

There are four major groups of keys on the PG 702 keyboard:

- one for entering instructions
- one for entering operand types
- one for selecting functions
- one for entering operand offsets and digits

Note

The keyboard overlay shown in Figure 4-1 is the international version of the PG 702 and uses international mnemonics. See Figure 2-3 on page 2-3 for the SIMATIC version of the keyboard overlay.

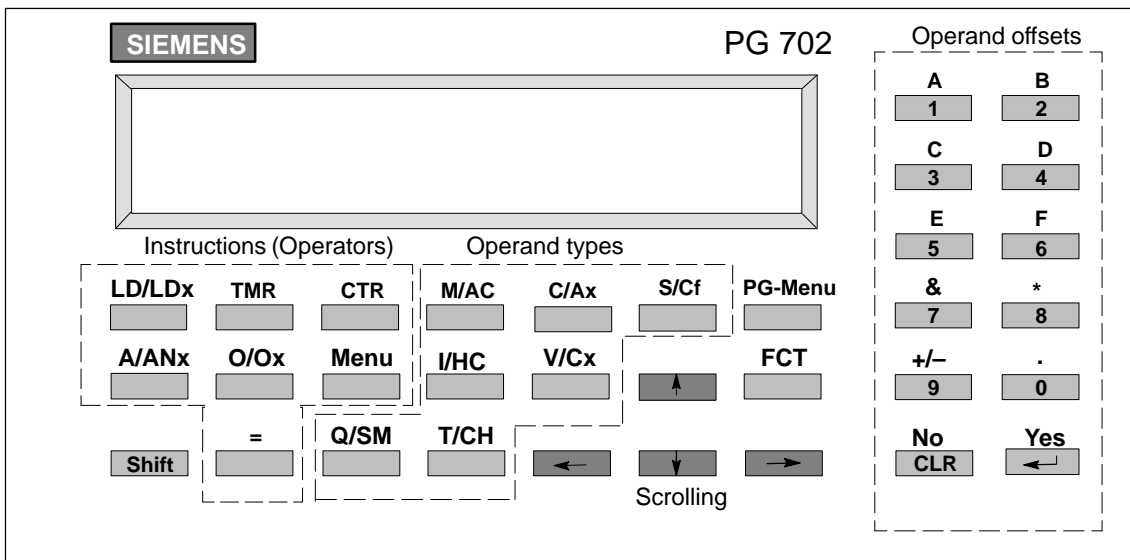


Figure 4-1 PG 702 Keypad and Display


Key Labeling and Key Functions

The key functions depend on the current mode of the programming device and on the cursor position in the entry fields.

Note

All keypad entries are checked for validity. The PG 702 beeps or displays an error message if you make an invalid entry.

Shift Key

Use the **Shift** key to change the function of keys that have both a blue and a white label. The blue (left) label or the label on the key, shows the default meaning of the key. The white (right) label shows the meaning of the key when you press and release the **Shift** key first. For other keys, the **Shift** key also affects the function even if there is no additional label on the key. For example, **Shift**  in the user program editor means “move the cursor to the next operand type or operand offset” (see page 5-8).

CLR Key


Pressing the **CLR** key while you are editing a program element or variable causes the following different operations, depending on the cursor position.

- On the instruction position of an unmodified statement, the **CLR** key clears the entire line.
- On the operand type position, the **CLR** key clears the whole operand.
- On the operand offset position, the **CLR** key only clears the digit under the cursor. All remaining digits of the operand are shifted one position to the left.

During non-edit operations, the **CLR** key escapes the operation and returns you to the PG menu.

During other functions, the **CLR** key terminates that function.

If you are prompted for a response to a question, the **CLR** key is the “no” answer.

The  key is a “yes” answer to a question and acknowledges error messages.

Use **Shift** **CLR** to undo an operation or escape a function.

Keys for Entering Instructions

The group of keys for entering instructions is located under the display. These keys are active only while you are editing and are sometimes assigned two functions. These functions activate selection menus for special instructions. They can also be used to enter instructions directly.

There are four different ways to enter instructions. Table 4-1 illustrates each of the four entry methods.

Table 4-1 Four Ways to Enter Instructions

Instruction	Key Entry	Process
A	A/ANx	Use the key assigned to a specific instruction.
AN	A/ANx A/ANx	Press the key assigned to a specific instruction twice to enter the NOT instruction.
AB=	Shift A/ANx 3 1	Use shift and the specific instruction key to view a menu with all versions of the specified instruction.
ANDB	Menu 5 1 1	Use the Menu button to select instructions that are not printed on the keys.

Section 5.3 contains the detailed description for entering instructions. Appendix A contains the description of all instructions.

Operands

A statement list program is made up of statements that are delineated by at least one network. Each statement takes on the following form:

instruction [operand] [,operand] [,operand] where [] indicate optional elements

The operand is the data to be operated on, or a description of the location of the data to be operated on by the instruction. Operands take on two forms:

- Addresses include the addressing mode, address identifier, size, and offset. Address modes are either direct or indirect.
- Constants are either integer, hex, ASCII hex or floating point numbers.

Examples:

Constants:

Integer	#123
Hexadecimal	16#A987
ASCII Hex	16"41
Floating point	#-1.23E+6

Direct address:

VB0 (V is address identifier, B is size for byte, 0 is offset)

MW16 (M is address identifier, W is size for word, 16 is offset)

SMD42 (SM is address identifier, D is size for double word, 42 is offset)

I0.0 (I is address identifier, bit size is indicated by ".", with the first 0 being the byte number and the second zero being the bit number)

Indirect address:

*AC0

*VD100

To set up indirect address pointer (MOVD only):

```
MOVD &VB0 VD100
```

Keys for Entering Operand Types and Operand Offsets

The group of keys for entering operand address identifiers is also located under the display. The numerical keys to the right of the display are used to enter operand offsets.

Example:

- Enter an operand address identifier, such as SMB.

When you press **Shift** **Q/SM**, the SMB operand is entered.

- Enter an operand offset:

Press one of the numerical keys: **0** to **9**.

Section 5.4 contains the detailed description of entering operands. Appendix A.3 contains the complete key description.

Entering a Period and Comma

In addition to the normal cursor movement, **→** also has the following functions:

- Enters a period for bit-size operand offsets
- Moves the cursor to the entry field of the second operand. The comma is automatically generated as soon as the operand type of the second parameter is entered.

General Remarks

The blue-lettered functions are always applicable if you did not use **Shift** to switch to the white-lettered functions.

Note

If you press **Shift** before you press a key, a flashing “S” appears at the bottom right of the display. When you press the relevant key, the “S” display disappears and the basic key functions are active again.

















Confirming Messages

The programming device displays system messages in the lower display line. You must confirm these messages by pressing **↵** before continuing with your inputs.

Key Colors and Key Functions

The keys on the PG 702 are different colors and are grouped by function. Table 4-2 lists the colors of the keys and their functions.

Table 4-2 Key Colors and Functions

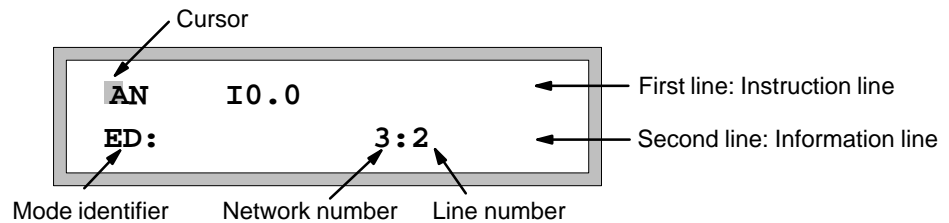
Color	Key	Function
Yellow		<ul style="list-style-type: none"> Return key for concluding or confirming functions Answers "yes" to a "yes/no" question
Red		<ul style="list-style-type: none"> Deletes the following items: <ul style="list-style-type: none"> Complete lines Parts of a line (depending on cursor position) Entire entry fields Answers "no" to a "yes/no" question
		Aborts an operator input sequence (ESC). Restores original line.
White		Exclusively a Shift key: after  is pressed, the white functions above the other keys are activated. "S" is displayed on the bottom right corner of the display
Green	   	<ul style="list-style-type: none"> Moves within a line or moves in the menus Scrolls from line to line
	 	<ul style="list-style-type: none"> Changes from one menu to another
	  or 	Changes to another network
Light Gray	Instruction, operand, and menu keys	<ul style="list-style-type: none"> Enters instructions in the STL editor. Use the  key to change to the white instructions. Enters operands Accesses various menus
Dark Gray	Number keys to the right of the display	<ul style="list-style-type: none"> Enters operand offsets in the STL editor. Use the  key to change to the white instructions. Enters values for write and force variables

The Display

The PG 702 uses a two-line display: the Instruction line and the Information line. Each line can display 20 characters.

The PG 702 display lines perform different tasks that depend on the operating status and the function to be carried out.

Example of a display while you are editing a program block is shown below:



Note

If you press **Shift** **FCT** during edit mode, the PG 702 displays two lines of the program instead of displaying a program line and an information line. Press **Shift** **FCT** or use the arrow keys to return to the standard edit mode display.

Information Line

The operating mode and location within the program are displayed in the information line. Error messages also appear on the information line.

The identifiers describe the operating mode currently activated. These are displayed when room permits for the current function.

- ED STL edit mode (edit program block, change CPU configuration)
- DE data editor mode (edit data block)
- DG debug mode (read, write, and force variables)

4.2 Starting Up the PG 702 / Password Protection

Startup

As soon as the PG 702 is powered up, it identifies itself with the following message:

```
PG702 VERSION x.y
```

When this message disappears, the PG menu is activated and replaces the identifier message.

```
1 CPU PROGRAM BLOCK      >
2 CPU CONFIGURATION      >
```


Password Protection

You can use a password for the CPU to restrict access to functions that you specify. The password allows only authorized persons to access the CPU functions and the CPU memory.

There are various protection levels preventing access to the CPU in varying degrees. Please refer to the *SIMATIC S7-200 Programmable Controller System Manual* for explanations of the individual protection levels.

Password Prompt

If a function is protected by a password, the PG 702 displays the message, NEED PASSWORD.



Confirm the message by pressing the  key. Use the ENTER PASSWORD function to enter your password. See Section 4.11 for the input sequence.


4.3 Working with Menus

Using Menu

Some functions in the various PG 702 operating states must be selected by using a menu. You cannot select these functions directly with a key.

Selecting a Function from a Menu

The number of functions a menu contains can vary. If there are more functions than can be displayed in both lines, you can use the  or  key to scroll to display the hidden functions.

The same principle is applicable for all menus. Within a menu, you can press a numerical key to activate another menu (if there is one). You can also activate or select the menu item displayed in the upper line using the  key.


You can use the  key or the  key to move between the additional submenus.

Figure 4-2 shows an example of a menu sequence.

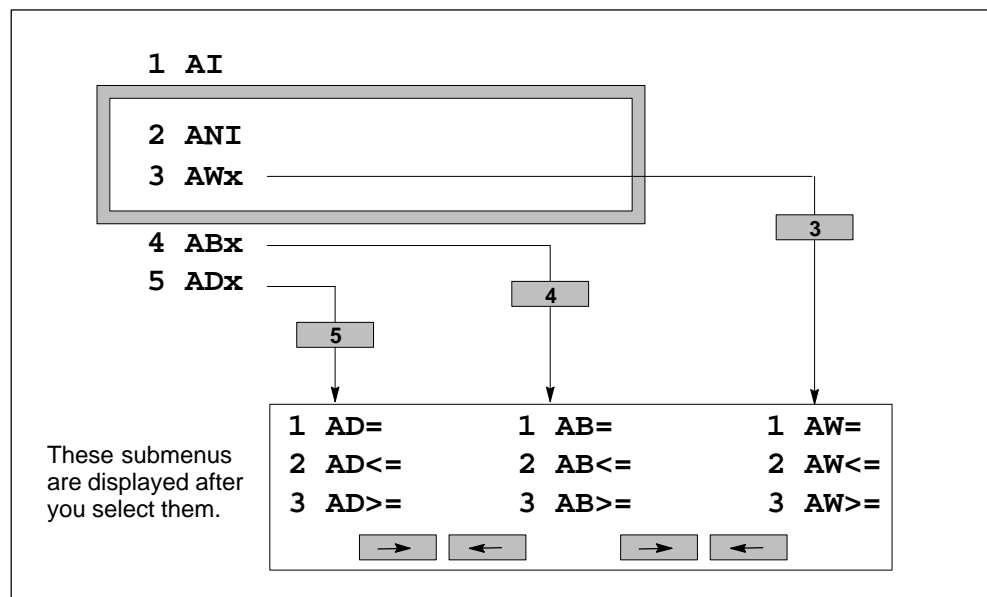


Figure 4-2 Example of a Menu Sequence

Exiting a Menu

You can exit the current menu in two ways:

- You can use the **0** key to return to the previous menu. Depending on the level you are in, press this key to return to the previous menu, or press this key several times to return to the main menu (PG menu).
- You can immediately return to the PG menu by pressing the **CLR** key.

Aborting an Entry

You can abort an entry by pressing the **Shift** and **CLR** keys.

4.4 Operating Modes of the PG 702

The PG 702 operates in three modes: the STL edit mode, the data edit mode, and the debug mode. These modes differ in the use of the display, the functions of some keys, and the set of inputs allowed.

You can also invoke a set of PG menu functions in almost any mode to perform an action without changing the operating mode of the PG 702.

Table 4-3 shows the operating modes of the PG 702, the two-letter identifier shown at the beginning of the second line of the display (when space permits), and the sequence of keys to press to enter the various functions.

Table 4-3 PG 702 Operating Modes

Mode	Second Line of Display	Functionality	Keystroke(s)
STL Edit	ED	Edit User Program Block	PG-Menu 1 1 or PG-Menu 1 3
		Edit CPU Configuration	PG-Menu 2 ...
Data Edit	DE	Edit User Data Block	PG-Menu 3 1 or PG-Menu 3 3
Debug	DG	Read and write variables (status chart)	PG-Menu 8 1 ...
		Force and unforce variables	PG-Menu 8 2 ...
		Single scan	PG-Menu 8 3

Program Block Edit Mode

The PG 702 enters the STL edit mode whenever a program block is uploaded. To exit the STL edit mode, press the **PG-Menu** key.

Data Block Edit Mode

The PG 702 enters the data edit mode whenever a data block is uploaded. The letters **DE** appear at the beginning of the second line of the display. To exit the data edit mode, press the **PG-Menu** key or the **Shift** and **CLR** keys.

Debug Mode

Use the **PG-Menu** **8** key sequence to enter the debug mode. The letters **DG** appear at the beginning of the second line of the display in debug mode. In the debug mode, you can modify and monitor variables, force variables, and perform the single scan function.

4.5 CPU Program Block PG-Menu 1

You can use the PG menu function CPU PROGRAM BLOCK to manipulate the CPU program block. From the main PG menu, select 1 to access the menu for the following functions:

- Upload from the CPU
- Download to the CPU
- Edit the program block

Uploading from the CPU

The PG 702 must be connected to an S7-200 CPU to upload from the CPU. You can send an upload command by selecting the first CPU program block function, UPLOAD FROM CPU. The PG 702 executes a program upload from the CPU. If a program is found, the first line of your program is displayed and is ready for editing. If an empty program is uploaded (see Section 3.2), the first line of the display is empty.

Uploading Large Programs


Ladder editors contained in programming packages like STEP 7–Micro/WIN or STEP 7–Micro/DOS automatically create networks. Such editors usually create many small networks within a program. The PG 702 works with these large programs without difficulty.

Statement list editors in programming packages create programs in which the user must manually create networks. This means that a program created using a statement list editor could contain one large network.

When the PG 702 uploads and displays a program, it deals with one network at a time. If the entire program is one network, this network may be very large. If the network contains more than 250 statement list instructions, the PG 702 automatically breaks the network into two or more networks of 250 statement list instructions each. Use the Shift ↓ key combination to switch networks and view the entire program.



You can edit these networks in the normal manner. The PG 702 allows networks to grow to 349 statement list instructions. After you enter the maximum number of statements, the PG 702 beeps to indicate that it is out of memory and cannot accept any more statements into the current network.

If the network has reached the maximum size and you are inserting statements at the end of the network, press Shift ↓ to go to the next network and continue.

If you are editing in the middle of the network and the network has reached the maximum allowed size, you should download the current program to the CPU. If the download is successful, upload the program from the CPU. The networks are resized to the 250 statement maximum when you upload the program. If there is an error during the download, the CPU will not accept the downloaded program. The PG 702 displays the error. Press  to confirm the error message. The editor reopens after confirmation of the error and automatically resizes the networks to the 250 statement maximum.

Downloading to the CPU


In order to run or test a program, you must download it to the CPU. Select the second CPU program block function, DOWNLOAD TO CPU, to download your program to the CPU. Once the program has been downloaded, the PG 702 displays the main PG menu again.

The program can also be downloaded from within the editor by pressing the   key sequence. Once the program has been downloaded, the PG 702 returns to the program editor.

Note

Programs being edited in the PG 702 are stored in RAM, and are not retained through a power loss. When entering large programs, you should occasionally perform the DOWNLOAD TO CPU function. This prevents losing the entire program in the event of a power loss or inadvertent PG 702 cable disconnection.

Exiting the Program Editor

To exit the program editor, press the  key. This allows access to the other functions of the PG 702.

Returning to the Program Editor

If you leave the program block edit mode to perform another function, you can continue your editing work by selecting the third CPU program block function, EDIT PROGRAM BLOCK. The PG 702 displays the first line of your program and shows the program block edit mode identifier (**ED**) in the second line of the display.

See Chapter 5 for additional information about editing the program block and moving within the editor.

4.6 CPU Configuration PG-Menu 2

The CPU configuration functions modify the system data block. These functions upload the current configuration, modify the specified field in the CPU configuration, and then download the new configuration to the CPU.

To access the menu for the CPU configuration, select 2 from the main PG menu. The functions listed in Table 4-4 are available.

Table 4-4 CPU Configuration Menu

Menu Text on Display	Description of Function	Key Sequence
1 RETENTIVE RANGES	Set retentive ranges	2 1
2 OUTPUT TABLE>	Set up the output table	2 2
1 FREEZE OUTPUTS	Freeze the outputs at their current states	2 2 1
2 SET OUTPUT TABLE	Select the state for each output	2 2 2
3 SET INPUT FILTERS>	Set input filter delay times	2 3
1 I0.0 → I0.3	Set the input filter times for I0.0 through I0.3	2 3 1
2 I0.4 → I0.7	Set the input filter times for I0.4 through I0.7	2 3 2
3 I1.0 → I1.3	Set the input filter times for I1.0 through I1.3	2 3 3
4 I1.4 → I1.5	Set the input filter times for I1.4 through I1.5	2 3 4
4 PORT 0 PARAMETERS>	Set/read port 0 parameters	2 4
1 SET PORT 0 ADDR	Set/read CPU station address	2 4 1
2 SET BAUD RATE	Set/read baud rate	2 4 2
3 SET HSA	Set/read highest station address	2 4 3
4 SET RETRY COUNT	Set/read retry count	2 4 4
5 SET GAP UPDATE	Set/read gap update factor	2 4 5
5 PORT 1 PARAMETERS>	Set/read port 1 parameters	2 5
1 SET PORT 1 ADDR	Set/read CPU station address	2 5 1
2 SET BAUD RATE	Set/read baud rate	2 5 2
3 SET HSA	Set/read highest station address	2 5 3
4 SET RETRY COUNT	Set/read retry count	2 5 4
5 SET GAP UPDATE	Set/read gap update factor	2 5 5
6 DP PORT PARMS>	Set/read DP port parameters	2 6
1 SET DP PORT ADDR	Set the CPU station address	2 6 1
2 READ DP STATUS	Read the DP status	2 6 2

Setting Retentive Ranges PG-Menu 2 1

To define retentive ranges of user memory, select 1 from the CPU configuration menu. You can fill a table of up to six entries and specify a data range (from, to) for each of the entries in the fields that appear on the display. Figure 4-3 shows the display that allows you to define the beginning of a retentive range.

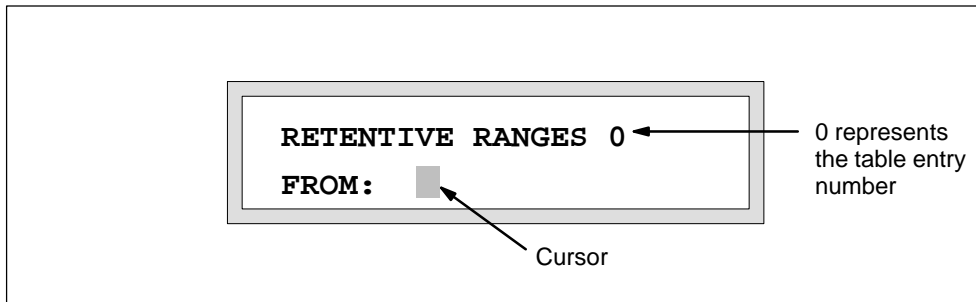


Figure 4-3 Defining the Beginning of a Retentive Range

Press the appropriate operand keys to enter the beginning of the range, and confirm the selection by pressing the ← key. The display prompts you to enter the end of the range as shown in Figure 4-4. The range is inclusive.

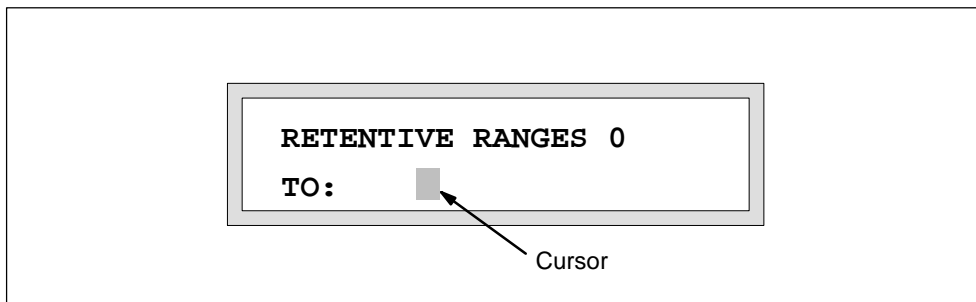


Figure 4-4 Defining the End of a Retentive Range

After you press the **←** key to confirm the end of the range selection, you can continue to set the range for additional entries. Whenever an operand has been specified previously, that operand is displayed as the default to the right of the colon on the second line of the display (see Figure 4-5).

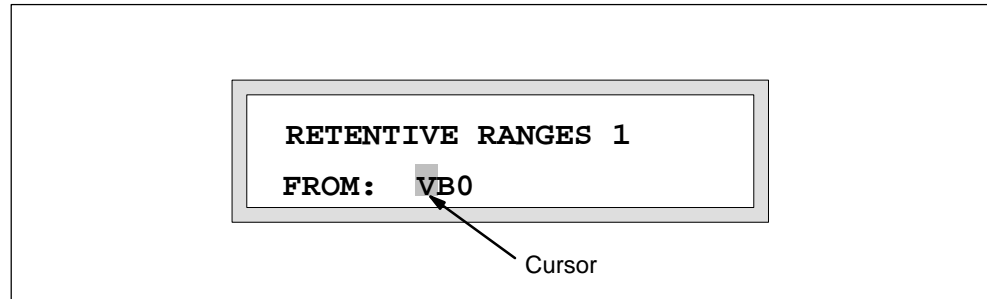


Figure 4-5 Viewing the Default Operand

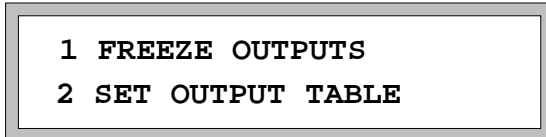
Press the **←** key (or the **↑** or **↓** keys) to accept the displayed operand or overwrite the operand with another one.

- To clear part of the operand address, place the cursor on the character you want to change and press **CLR**.
- To clear an entire operand field, place the cursor on the operand address identifier and press **CLR**. The whole operand is then cleared to blank spaces and you can enter a new value.

After you configure all ranges, press **FCT** **1** to write them to the CPU.

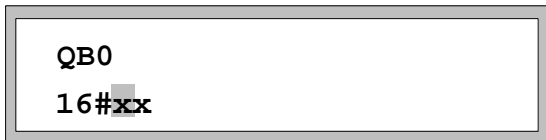
Setting the Output Table PG-Menu 2 2

Use the OUTPUT TABLE function to determine the behavior of the outputs when the CPU enters the STOP mode. Whenever the CPU enters the STOP mode, the output points of the CPU are either frozen at their current value or set to the value in the output table. When you select 2 from the CPU configuration menu, the PG 702 allows you to freeze the outputs or define the output table.



If you select 1, the outputs are to be frozen and you return to the PG menu.

If you select 2, the outputs are to be set to the values specified in the output table. You can configure the state for each output by entering the hexadecimal number that represents the output states for the specified eight output points.



The cursor is positioned on the first x in the first line. Enter a hexadecimal value and press the ← key to confirm. The cursor then moves to the next value. You can also use the up and down arrow keys to move among the outputs and accept the values you enter. Use the FCT 1 key sequence to save the values to the CPU.

Setting the Input Filters PG-Menu 2 3

Use the SET INPUT FILTERS function to set the input filter delay times. Input filter delay times are set in groups of input points. After you select 3 from the CPU configuration menu, scroll through the display to view the groups shown in Figure 4-6 on the PG 702 (only two are displayed at a time).

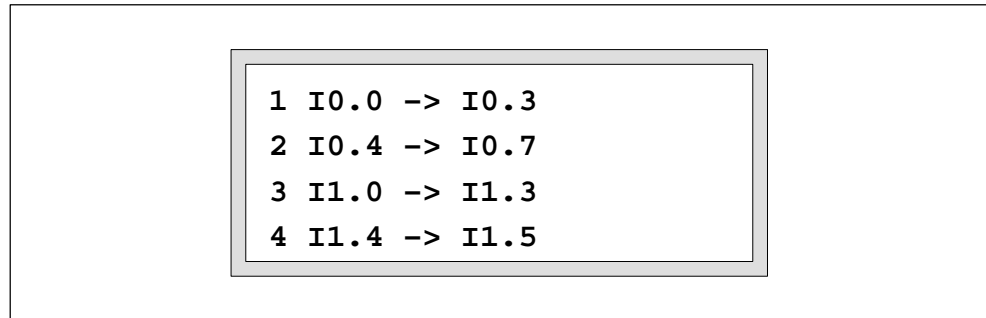
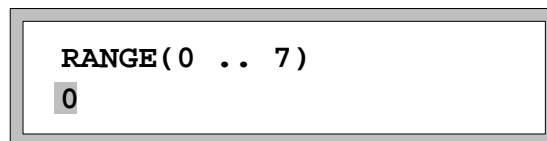


Figure 4-6 Sample of I/O Groups


To select a group of input points, enter the number of the group (1 through 4) or place the cursor on the group you want to select and press the ← key. The following screen appears to indicate the values that can be entered and what the current value is:



You can select eight different predefined delays for each group of input points. Table 4-5 shows which value to enter in the PG 702 in order to assign a particular delay time to a CPU. The predefined delay values differ, based on CPU model.

Table 4-5 Input Delay Time Values

Value	Delay Time on CPU 212	Delay Time on CPU 214, CPU 215, CPU 216
0	0.3 ms	0.2 ms
1	0.5 ms	0.3 ms
2	0.9 ms	0.6 ms
3	1.7 ms	1.1 ms
4	1.7 ms	1.1 ms
5	3.3 ms	2.2 ms
6	6.6 ms	4.4 ms
7	13.1 ms	8.7 ms

After you enter the appropriate value, press the  key to write the selection to the CPU.

Setting the Port 0 Parameters PG-Menu 2 4

Use the PORT 0 PARAMETERS function to set the port 0 attributes. After you select 4 from the CPU configuration menu, scroll through the display to select the functions shown in Table 4-6:

Table 4-6 Port 0 Parameter Functions

Menu Text on Display	Description of Function	Key Sequence
4 PORT 0 PARAMETERS>	Set/read port 0 parameters	2 4
1 SET PORT 0 ADDR	Set/read station address	2 4 1
2 SET BAUD RATE	Set/read baud rate	2 4 2
3 SET HSA	Set/read highest station address	2 4 3
4 SET RETRY COUNT	Set/read retry count	2 4 4
5 SET GAP UPDATE	Set/read gap update factor	2 4 5

When you select an attribute, the PG 702 displays the valid range and the current value for the attribute, as shown below for the station address.

```

RANGE(1 .. 126)
2
```

You can then enter a value and press the ← key to confirm your entry. The value is immediately downloaded to the CPU.

If you select 2 (SET BAUD RATE), the following display appears:

```

BAUD RATE: 1
1:9600    2:19200
```

Select 1 or 2 and press the ← key to confirm your entry.

Press Shift CLR to cancel the operation.

Note

The highest station address, retry count, and gap update factor parameters can generally remain the default (factory-set) values unless you want to attempt advanced networking.

Setting the Port 1 Parameters PG-Menu 2 5

Use the PORT 1 PARAMETERS function to set the port 1 attributes. After you select 5 from the CPU configuration menu, scroll through the display to select the functions shown in Table 4-7.

Table 4-7 Port 1 Parameter Functions

Menu Text on Display	Description of Function	Key Sequence
5 PORT 1 PARAMETERS>	Set/read port 1 parameters	2 5
1 SET PORT 1 ADDR	Set/read station address	2 5 1
2 SET BAUD RATE	Set/read baud rate	2 5 2
3 SET HSA	Set/read highest station address	2 5 3
4 SET RETRY COUNT	Set/read retry count	2 5 4
5 SET GAP UPDATE	Set/read gap update factor	2 5 5

When you select an attribute, the PG 702 displays the valid range and the current value for the attribute. You can then enter a value and press the ← key to confirm your entry. The value is immediately downloaded to the CPU.

Press Shift CLR to cancel the operation.

Note

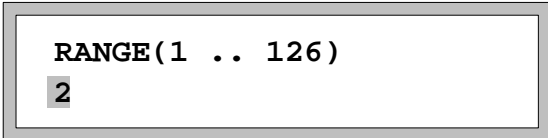
The highest station address, retry count, and gap update factor parameters can generally remain the default (factory-set) values unless you want to attempt advanced networking.

Setting the DP Port Address and Reading DP Status PG-Menu 2 6

Use the DP PORT PARMS function to set the address on the DP port and read the following DP parameters:

- DP status
- Master address
- Output buffer address
- Size of output buffer
- Size of input buffer

This feature applies only to the DP port (port 1) of the CPU 215. When you select 1 (SET DP PORT ADDR), the following screen is displayed:



A screenshot of a terminal window showing the text "RANGE(1 .. 126)" on the first line. On the second line, the number "2" is displayed with a small cursor box to its left, indicating it is the current input value.

Enter a value and press the ← key. The value is immediately downloaded to the CPU.

Press Shift CLR to cancel the operation.

When you select **2** (READ DP STATUS), a scrollable screen is displayed. The first line shows the parameter name and the second line displays the value for that parameter. Use the **↑** and **↓** keys to scroll through the parameters shown in Figure 4-7.

Press **CLR** or **Shift CLR** to cancel this function.

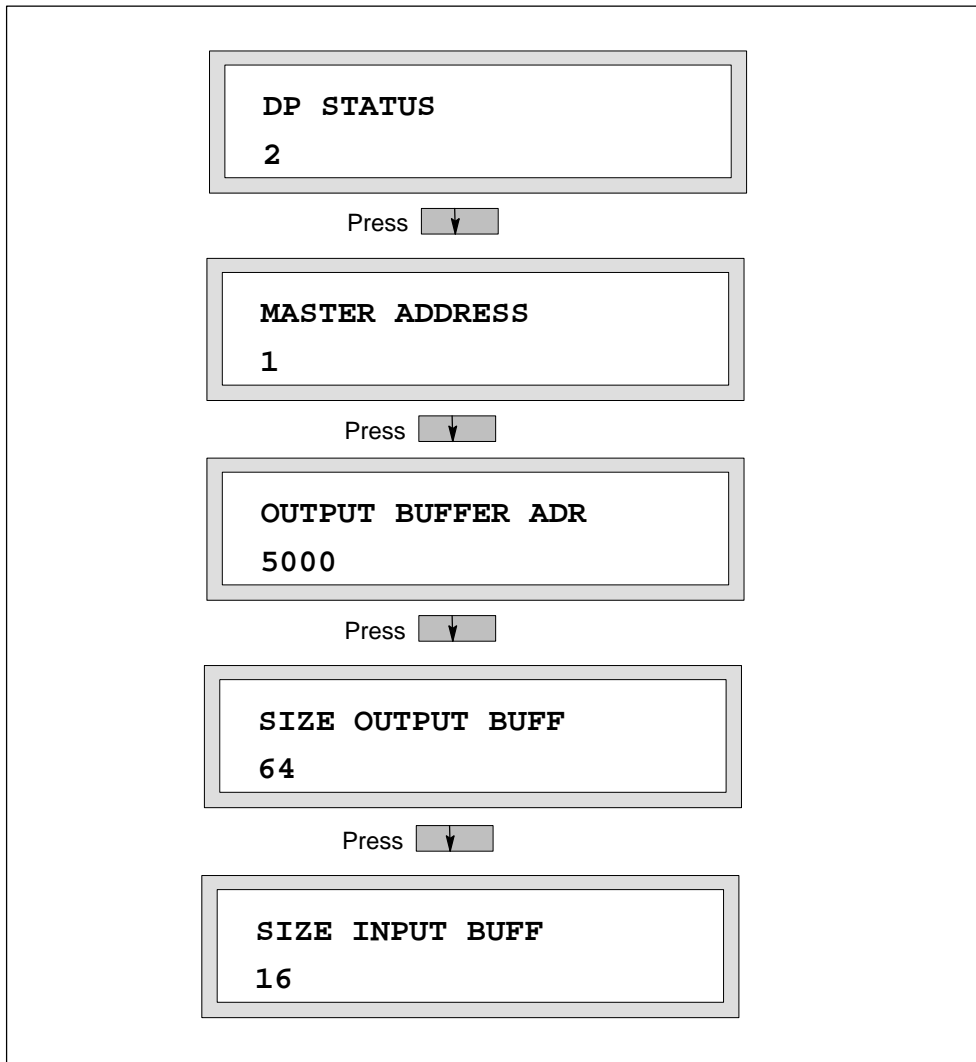


Figure 4-7 DP Status Parameter Screens

Refer to the *SIMATIC S7-200 Programmable Controller System Manual* for a description of the DP status values.

4.7 CPU Data Block PG-Menu 3

You can use the PG menu function CPU DATA BLOCK to manipulate the CPU data block. The data block is often referred to as DB1, and is used for configuring V memory.

Regardless of the existence or size of the current data block in the CPU, the PG 702 data block editor always allows edits and downloads of data blocks equal to the full range of the permanent portion of V memory. For some CPUs, this is the full amount of V memory, for other CPUs, this is a subset of V memory.

Refer to the *S7-200 Programmable Controller System Manual* for CPU-specific information regarding the size of permanent V memory area.

From the PG menu, select 3 to access the menu for the functions shown in Table 4-8:

Table 4-8 CPU Data Block Functions

Menu Text on Display	Description of Function	Key Sequence
3 CPU DATA BLOCK>	Manipulate CPU data block (DB1)	3
1 UPLOAD FROM CPU	Upload data block from CPU and enter data block editor	3 1
2 DOWNLOAD TO CPU	Download data block from PG memory to CPU memory	3 2
3 EDIT DATA BLOCK	Enter data block editor	3 3

Uploading from the CPU

To upload the data block from the CPU, ensure that the PG 702 is connected to the CPU, and select the first CPU data block function, UPLOAD FROM CPU. The PG 702 executes a data block upload from the CPU. If a data block is found in the CPU, the PG 702 starts the data block editor and displays the first value in the data block, VW0, as shown in Figure 4-8.

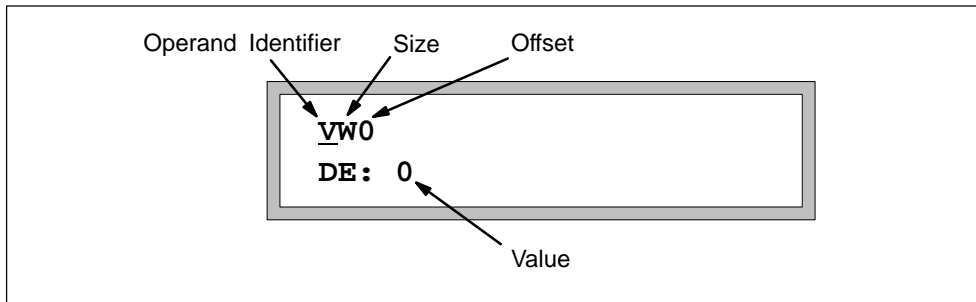
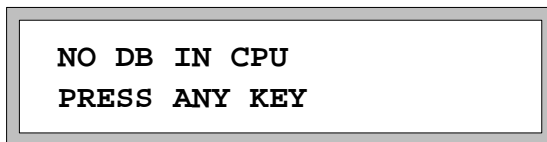


Figure 4-8 View of a Data Block Value

The PG 702 uploads the data block that is defined in the CPU. (For example, if only four bytes are defined, the PG 702 uploads four bytes.) The data block size that is downloaded is the size of the permanent area of the V memory of the attached CPU. If, for example, the PG 702 is connected to a CPU 216 and the data block defined is four bytes, the PG 702 uploads four bytes and downloads 5120 bytes.

If a data block is not present, the following is displayed:



The PG 702 then starts the data block editor so a data block can be created.

Downloading to the CPU

To download your data, select the second CPU data block function, DOWNLOAD TO CPU. Once the data block has been downloaded, the PG 702 displays the main PG menu again. You can also press **FCT** **1** from within the data block editor to download to the CPU. After the download is complete, the PG 702 returns to the data block editor.

Editing the Data Block










If you leave the data block editor to perform another function, you can continue your editing work by selecting the third CPU data block function, EDIT DATA BLOCK, and the PG 702 displays the first value in the data block.

Working Within the Data Block Editor





Type and format selections within the data block remain in effect until they are changed. The default type is VW and the default format is signed integer (INT).




Table 4-9 lists the PG 702 keys and their functions within the data block editor.

Table 4-9 Key Definitions in the Data Block Editor





Key(s)	Function
	Scroll to next entry
	Scroll to previous entry
Shift 	Scroll down 10 entries
Shift 	Scroll up 10 entries
Shift 	Go to first entry
Shift 	Go to last entry
	Enter/make selection
	Move cursor left
	Move cursor right
V/Cx	Toggle to change data size: VB, VW, VD
Shift V/Cx	Toggle to change data format: INT (signed integer), UINT (unsigned integer), HEX (hexadecimal), FLOAT (floating point).
CLR	Delete a character
Shift CLR	Cancel the operation. Redisplay the original value. (Also serves as ESC key).
FCT	Data block function menu
PG-Menu	Return to PG menu

Positioning the Cursor Within the Data Block Editor

When you select a new line, the cursor is positioned on the first column in the line. Use the  or  key to move the cursor to the next or previous data line. You can only scroll to the last permanent V memory location. If you are editing a line and press either the  or  key, the edit function is cancelled and the new entry is displayed.


Use the  key to position the cursor on the address field of the operand. If you press the  key again, the cursor moves to the next digit in the address field. Use the  key to move the cursor to the previous digit or operand type.


Fast Moving




You can advance the cursor immediately to the beginning or end of the data block by pressing the **Shift** key together with the  or  key. If you use the **Shift** key together with the  or  key, the cursor moves 10 elements at a time.




Entering Statements Within the Data Block

When the cursor is placed on the operand identifier, you can modify the operand size by repeatedly pressing the **V/Cx** key. The operand size toggles through VB, VW, VD, then back to VB. The cursor then moves to the offset field.

Enter the offset value by pressing the desired digits. After the offset has been entered, press  to advance the cursor to the value field on line 2 of the display.

To change the data format, position the cursor on the data value field (second line) and press the **Shift** and **V/Cx** keys together. The data format options toggle through INT, UINT, HEX, and FLOAT. FLOAT is only valid for operand size double (VD). Press  to confirm your selection.

To change the value, use the **CLR** key to delete a digit and the  and  keys to move among the digits. Enter the desired digits and press  to apply the value. The PG 702 then displays the next entry.

After you have entered all of your values, press the **FCT** key and select  (DOWNLOAD TO CPU) to download the data block to the CPU. After the data block is downloaded, the display shows the last entry before the download. Alternately, you can download the data block from the PG menu by selecting **PG-Menu**  .

4.8 Flash Operations PG-Menu 4

To permanently store your edited program, you must download your program to the CPU and then transfer it to the PG flash (non-volatile) memory.

The FLASH OPERATIONS functions shown in Table 4-10 allow you to transfer and compare entire programs (program block, data block, and CPU configuration). These functions work with the non-volatile memory of the PG 702. The program block and data block editors store information temporarily in RAM until a power cycle, and are unaffected by the flash operations.

Table 4-10 Flash Operation Functions

Menu Text on Display	Description of the Function	Key Sequence
4 FLASH OPERATIONS >	Transfers and compares	4
1 CPU MEM > PG FLASH	Copies CPU memory to PG flash memory	4 1
2 PG FLASH > CPU MEM	Copies PG flash memory to CPU memory	4 2
3 CPU MEM = PG FLASH	Compares program in PG flash memory to program in CPU	4 3

CPU Memory to PG Flash PG-Menu 4 1

The first flash function (CPU MEM > PG FLASH) uploads the entire S7-200 CPU program (program block, data block, and CPU configuration) to the non-volatile memory of the PG 702.

If the program block, data block, CPU configuration, and output table exist in the CPU, the PG 702 stores the data to the PG flash memory. Before reading the blocks from the CPU, the PG 702 erases the entire flash memory. This ensures that any old data is destroyed. However, because the PG 702 erases the original flash memory, you cannot return to the old flash memory program if there is a problem with the uploads from the CPU.

The blocks are uploaded and stored to flash memory in the following sequence:

1. Program block
2. Data block
3. CPU configuration
4. Output table data

If there is a communication error at any point in the operation, the copy to flash memory is aborted and the PG 702 displays the communication error. The flash memory retains whatever was uploaded before the communication error occurred.

PG Flash to CPU Memory **PG-Menu** **4** **2**

The second flash function (PG FLASH > CPU MEM) downloads the entire program from the non-volatile memory of the PG 702 to the S7-200 CPU.

First, the PG 702 checks to see that there is at least one block stored in the PG flash memory. If a CPU program has ever been copied to PG flash memory, there will at least be CPU configuration information present in the flash memory, even if there is no program block or data block.

The program block is copied to the CPU first, followed by the data block, and finally the CPU configuration.

If there is a communication error at any point in the operation, the function is aborted and the PG 702 displays the error. Remaining blocks are not downloaded.

The downloaded program block and data block are exact copies of the blocks that were previously uploaded to the flash memory. If the blocks do not fit into the CPU (for instance, when you copy a large CPU 216 program into flash memory and then try to download that program into a CPU 214), a communication error occurs and the function is aborted.

The CPU configuration and output table data are handled differently from the program block and data block so that configuration information from one CPU type can be downloaded to another CPU type. Because not all CPUs support the same CPU configuration fields and formats, exact copies of the CPU configuration do not allow for model independence. In order to ensure compatibility, the PG 702 uses the CPU configuration from the target CPU and overwrites its values with values from the configuration that is stored in PG flash memory. When copying the flash memory configuration to a CPU, the PG 702 first uploads the current CPU configuration (if present) or default CPU configuration (always present) from the CPU. The values from the CPU configuration that are stored in flash memory are written into the CPU configuration that was uploaded from the target CPU. In this way, only those fields that are supported by the target CPU are downloaded to that CPU. Fields that are not supported are ignored. For example, if a CPU 216 program is copied into the PG 702 flash and then this program is downloaded to a CPU 212, the port 1 configuration of the CPU 216 is not copied to the CPU 212.

Note

There is no warning given if a particular field is not downloaded to the CPU.

If the current CPU address and baud rate configured in the PG 702 do not match the values downloaded to the CPU, the PG 702 will automatically switch itself to match the new CPU address and baud rate. The new CPU address or baud rate values are not stored in the PG 702. When you power cycle the PG 702, it reverts to the previous CPU address and baud rate values.

For example, if the target CPU port is set to address 2, and the configuration stored in PG flash requires this port to be address 10, the PG 702 will copy from flash memory to the CPU using address 2, then switch to address 10 so that the PG 702 can continue to communicate with the CPU using the new address. When the PG 702 is power cycled, it reverts to the previously stored permanent address for the CPU port. In this example, it reverts to address 2. This allows the PG 702 to be ready to program the next CPU.

Note

If a block does not exist in the PG flash memory, the PG 702 deletes that particular block from the target CPU. This is true for all blocks. The output table information is always written to the CPU, whether or not CPU configuration exists.

Compare Program in Flash to Program in CPU Memory PG-Menu 4 3

The third flash function (CPU MEM = PG FLASH) compares the program in the non-volatile flash memory of the PG 702 to the program in the S7-200 CPU in the following order:

1. Program block
2. Data block
3. CPU configuration

If there are no blocks in flash memory, the comparison aborts and the PG 702 displays an error message noting that the flash memory is empty. If there is at least one block, the comparison proceeds. If there is a mismatch, the PG 702 displays a message showing which block did not match. If there is a communication error during the comparison, the comparison is aborted and the PG 702 displays the communication error message.

The CPU configuration and output table data are handled differently from the program block and data block. The current CPU configuration is first uploaded from the CPU. If there is no current CPU configuration in the CPU, the configuration structure in the flash memory is checked to see if there was a CPU configuration in the original CPU. If the status is not the same, the comparison fails. If the CPU configuration exists in both places, the PG 702 compares the CPU configuration fields from the uploaded CPU configuration to the corresponding fields stored in the configuration structure from the PG flash memory. Only those fields present in the uploaded CPU configuration are compared. If a field does not exist in the connected CPU, that field is not compared to the values in flash memory. The output table data is always compared whether or not a CPU configuration exists.

Compatibility

To increase compatibility between older and newer versions of the same model of CPU, the PG 702 reads the default CPU configuration from the CPU before downloading and updates all common features. Features in the CPU configuration that are not supported are ignored.

Programs are upwardly compatible between CPU models. See Table 4-11.

Table 4-11 Compatibility Between CPU Models

Original CPU	Target CPU
CPU 212	CPU 214, CPU 215, CPU 216
CPU 214	CPU 215, CPU 216
CPU 215	CPU 216

Programs are downwardly compatible between CPU models (for example, from a CPU 216 to a CPU 212), only when common feature sets are used and when program sizes allow.

4.9 CPU Functions PG-Menu 5

You can select the CPU functions to make CPU mode changes, clear or copy the CPU memory, read the I/O configuration, read and set the time-of-day clock, and read CPU release and error information. Table 4-12 lists the CPU Functions available in the PG 702.

Table 4-12 CPU Functions

Menu Text on Display	Description of Function	Key Sequence
5 CPU FUNCTIONS>	Execute CPU functions	5
1 PUT CPU IN RUN	Start user program execution	5 1
2 PUT CPU IN STOP	Stop user program execution	5 2
3 CLEAR CPU MEMORY>	Clear user memory in the S7-200 CPU	5 3
1 CLEAR ALL	Clear all user CPU memory	5 3 1
2 CLEAR PROG BLOCK	Clear the program block in the CPU	5 3 2
3 CLEAR DATA BLOCK	Clear the data block in the CPU	5 3 3
4 CPU MEM > MEM CRT	Copy CPU memory to CPU memory cartridge	5 4
5 READ I/O CONFIG	Read I/O configuration	5 5
6 TIME-OF-DAY>	Read/Write CPU time-of-day clock	5 6
1 READ TIME-OF-DAY	Read the CPU time-of-day clock	5 6 1
2 WRITE TIME-OF-DAY	Write the CPU time-of-day clock	5 6 2
7 READ CPU VERSION	Read software release and ASIC release	5 7
8 READ CPU ERROR	Read error from S7-200 CPU	5 8

Put CPU in RUN/STOP **PG-Menu** **5** **1** or **PG-Menu** **5** **2**

To change the CPU mode, select a RUN/STOP function. To place the CPU in RUN, press **1**. To place the CPU in STOP, press **2**. Confirm your selection by pressing **←** or cancel the operation by pressing **CLR**.



Clear CPU Memory **PG-Menu** **5** **3**

To clear memory in the S7-200 CPU, select the CLEAR CPU MEMORY function. You can then choose to clear all the CPU memory, clear the program block, or clear the data block. Confirm your selection by pressing **←** or cancel the operation by pressing the **CLR** key.

Copy CPU Memory **PG-Menu** **5** **4**

To copy CPU memory to the CPU memory cartridge, select the CPU MEM > CPU CRT function. Confirm your selection by pressing **←** or cancel the operation by pressing the **CLR** key.

Read I/O Configuration **PG-Menu** **5** **5**

The READ I/O CONFIG function displays the contents of a table containing the module number and type and the number of input and output points on that module. Use the  and  keys to scroll through the table. A sample view of the I/O Configuration display is shown in Figure 4-9.

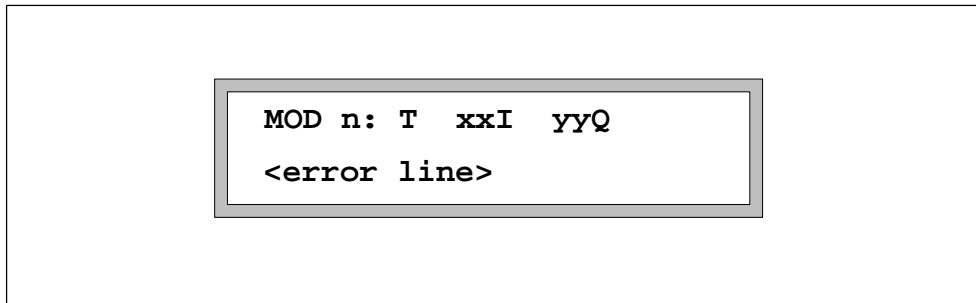


Figure 4-9 Sample View of the Read I/O Configuration Display

The following items are shown in the I/O configuration display:

- n: Module number 0-6
- T: Kind of module
 - D: discrete module
 - A: analog module
- xx: Number of inputs
- yy: Number of outputs

The error line can display the following messages:

- NO ERRORS
- NO MODULE (displayed if there is no module in the given slot)
- ERR C: I/O configuration error
- ERR R: Out of range error
- ERR P: No user power error

Use the **CLR** key to terminate the read I/O configuration function.

Read/Write Time-Of-Day Clock **PG-Menu** **5** **6**

To read the time-of-day clock in the CPU, press **PG-Menu** **5** **6** **1**. The READ TIME-OF-DAY function displays the time-of-day clock in the following format:

```
YY-MM-DD  HH:MM:SS
<DAY OF THE WEEK>
```

For example, February 19, 1998 at 8:15 a.m.:

```
98-02-19  08:15:00
THURSDAY
```

Press any key to exit the function.

Press **PG-Menu** **5** **6** **2** to change the time-of-day clock. The WRITE TIME-OF-DAY function displays the following screen first:

```
YEAR: xx
<valid range>
```

The display prompts you to enter the year, month, day, hour, minute, second, and day of the week. Confirm each entry by pressing the **←** key. The display then shows your entries and asks if you want to write the new time-of-day to the CPU. Confirm by pressing **←**.

Use the keys shown in Table 4-13 to modify the time-of-day clock.

Table 4-13 Valid Keys for Modifying the Time-of-Day Clock

Key	Function
↓	Enter new value/go to next value
←	Enter new value/go to next value
CLR	Delete character
Shift CLR	Escape from function
0 through 9	Digits to enter value
0 through 7	Digits to enter day of the week. Enter 0 to turn the day of the week display off. Enter 1 through 7 to specify the day of the week (1 = Sunday).

Figure 4-10 shows the WRITE TIME-OF-DAY screens from the PG 702 display.

YEAR: 98
(0-99)

Press or to enter a new value.

MONTH: 2
(1-12)

Press or to enter a new value.

DAY: 26
(1-31)

Press or to enter a new value.

HOUR: 9
(0-23)

Press or to enter a new value.

MINUTE: 14
(0-59)

Press or to enter a new value.

SECOND: 25
(0-59)

Press or to enter a new value.

DAY OF WEEK: 5
0:OFF 1-7(SUN=1)

Press or to enter a new value.

98/02/26 09:14:25
WRITE TO CPU?

Press yes () or no ().

Figure 4-10 Write Time-of-Day Screens

Read CPU Version PG-Menu 5 7

To read the software release version and the ASIC release, select the READ CPU VERSION function. Press any key to return to the main PG menu.

Read CPU Error PG-Menu 5 8

To read a list of errors from the S7-200 CPU, select the READ CPU ERROR function. Press any key to return to the main PG menu.

```
NF:16#0000    F:16#0000
PRESS ANY KEY
```

- NF: Code for non-fatal errors
- F: Code for fatal errors

See the *SIMATIC S7-200 Programmable Controller System Manual* for a listing of the non-fatal and fatal error codes, and their descriptions.

4.10 PG Configuration PG-Menu 6

Use the PG CONFIGURATION functions shown in Table 4-14 to find and select a CPU address, and set the PG address, baud rate, language, and beep volume. You can also set the access capability level to restricted or unrestricted.

Table 4-14 PG Configuration Functions

Menu Text on Display	Description of Function	Key Sequence
6 PG CONFIGURATION>	Modify PG configuration	6
1 FIND CPU ADDRESS	Find S7-200 CPU address to communicate to	6 1
2 SELECT CPU ADDR	Select S7-200 CPU address	6 2
3 SET PG ADDRESS	Set PG station address	6 3
4 SET BAUD RATE	Set baud rate (9600 or 19200)	6 4
5 SET LANGUAGE	Set display language	6 5
1 GERMAN >	Set the display to German	6 5 1
1 SIMATIC	Set SIMATIC mnemonic	6 5 1 1
2 INTERNATIONAL	Set international mnemonic	6 5 1 2
2 ENGLISH >	Set the display to English	6 5 2
1 SIMATIC	Set SIMATIC mnemonic	6 5 2 1
2 INTERNATIONAL	Set international mnemonic	6 5 2 2
3 FRENCH >	Set the display to French	6 5 3
1 SIMATIC	Set SIMATIC mnemonic	6 5 3 1
2 INTERNATIONAL	Set international mnemonic	6 5 3 2
4 ITALIAN >	Set the display to Italian	6 5 4
1 SIMATIC	Set SIMATIC mnemonic	6 5 4 1
2 INTERNATIONAL	Set international mnemonic	6 5 4 2
5 SPANISH >	Set the display to Spanish	6 5 5
1 SIMATIC	Set SIMATIC mnemonic	6 5 5 1
2 INTERNATIONAL	Set international mnemonic	6 5 5 2
6 SET BEEP VOLUME	Set beeper loudness	6 6
7 CAPABILITY LEVEL>	Set restricted or unrestricted access capability	6 7
1 RESTRICTED	Set the access to restricted	6 7 1

Find CPU Address **PG-Menu** **6** **1**

To find the CPU address within the PG configuration, press **PG-Menu** **6** **1**. The PG 702 displays the current CPU address. Press any key to return to the PG menu.

Note

If the PG 702 and CPU are set to different baud rates, the PG 702 reports CPU NOT FOUND. Change the PG 702 baud rate (see below) and try to find the CPU address again.

Select CPU Address **PG-Menu** **6** **2**

To select a CPU address within the PG configuration that you want to communicate to, press **PG-Menu** **6** **2**. Enter the CPU address you want to select and press **←** to save the value and return to the PG menu.

Set PG Address **PG-Menu** **6** **3**

To set your PG address, press **PG-Menu** **6** **3**. Enter the address you want and press **←** to write the address and return to the PG menu.

Set Baud Rate **PG-Menu** **6** **4**

To set the baud rate, press **PG-Menu** **6** **4**. Press **1** to select a baud rate of 9600. Press **2** to select a baud rate of 19200. Press **←** to confirm your selection and return to the PG menu.

Set Language **PG-Menu** **6** **5**

English is set at the factory as the default language. To change the language, press **PG-Menu** **6** **5**. Press one of the following keys to make your language selection:

1 German

2 English

3 French

4 Italian

5 Spanish

Press one of the following keys to select the mnemonic (the choice appears after you select the menu language):

1 SIMATIC

2 International

Set Beep Volume **PG-Menu** **6** **6**

To set the beep volume, press **PG-Menu** **6** **6**. Press one of the following keys to make your selection:

- 0** No volume
- 1** Quiet beep
- 2** Louder beep
- 3** Louder beep
- 4** Loudest beep

Press **←** to confirm your selection and return to the PG menu.

Capability Level **PG-Menu** **6** **7**

Use this function to choose between two levels of access to the PG 702:

- The restricted level allows the user to access only the PG Configuration menu and the Flash Operation menu. Only two of the three flash functions are available in restricted mode (PG Flash to CPU, and Compare).
- The unrestricted level allows the user access to all the PG 702 functions. This level is the default operating mode.

The PG 702 powers up at the access level it was in when power was lost. If the PG 702 powers up at restricted access and you have forgotten your PG password, press the **FCT** key during powerup. This deletes the PG password and sets the PG 702 back to unrestricted access.

Entering the Restricted Mode

To restrict the access level to your PG 702, press **PG-Menu** **6** **7** **1**. The PG 702 displays the following:

```

ENTER PASSWORD
0
  
```

Enter a numerical PG password and press **←**. The range of the PG password is 0 to 999,999,999. The PG password is saved to the non-volatile memory of the PG 702.

The PG 702 now displays the restricted PG menu:

```

1 FLASH OPERATIONS >
2 PG CONFIGURATION >
  
```

The functions listed in Table 4-15 are available under restricted access.

Table 4-15 PG Menu Functions Available Under Restricted Capability

Menu Text on Display	Description of the Function	Key Sequence
1 FLASH OPERATIONS >	Transfers and compares	1
1 PG FLASH > CPU MEM	Transfers PG flash memory to CPU memory	1 1
2 CPU MEM = PG FLASH	Compares program in flash memory to program in CPU	1 2
2 PG CONFIGURATION >	Modifies PG configuration	2
1 FIND CPU ADDRESS	Finds S7-200 CPU address to communicate to	2 1
2 SELECT CPU ADDR	Selects S7-200 CPU address	2 2
3 SET PG ADDRESS	Sets the PG station address	2 3
4 SET BAUD RATE	Sets the baud rate (9600 or 19200)	2 4
5 SET LANGUAGE >	Sets the display and mnemonic language	2 5
6 SET BEEP VOLUME	Sets the beeper loudness	2 6
7 CAPABILITY LEVEL >	Sets access to restricted or unrestricted	2 7
1 UNRESTRICTED	Sets the access to unrestricted	2 7 1

Exiting the Restricted Mode

To unrestrict the access level to your PG 702, press

PG-Menu **2** **7** **1**. The PG 702 displays the following:

```
ENTER PASSWORD
0
```

Enter your PG password and press **←**.

The PG 702 now displays the normal PG menu:

```
1 CPU PROGRAM BLOCK >
2 CPU CONFIGURATION >
```

4.11 Passwords PG-Menu 7

Use the CPU PASSWORDS function shown in Table 4-16 to enter a CPU password to gain access to the CPU services.

Table 4-16 Passwords Function

Menu Text on Display	Description of Function	Key Sequence
7 PASSWORDS>	Enter or modify the password	7
1 ENTER PASSWORD	Enter user password (log in to CPU)	7 1
2 NEW/CHNG PASSWORD	Change or create a new password	7 2
3 DELETE PASSWORD	Delete the password (use default password)	7 3
4 PASSWORD LOGOUT	Release the password (log out from CPU)	7 4

You can enter the CPU password either at the beginning of the session or whenever the CPU password is requested. When the CPU password is requested, the following display appears:

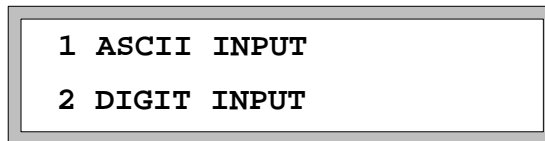
NEED PASSWORD

PRESS ANY KEY

After you press any key to clear the display, press the PG-Menu key to access the PG menu and then press 7 (PASSWORDS) followed by 1 (ENTER PASSWORD) to enter the CPU password.

Enter Password

Select this function to enter a password into the CPU. After you select (ENTER PASSWORD), the following screen appears:



```
1 ASCII INPUT
2 DIGIT INPUT
```

If you select ASCII input, enter a text password using the procedure to enter ASCII strings (see Appendix B) or select digit input to enter a numeric password on the screen shown below. The CPU password must be eight characters long. If you select a password that is less than eight characters, the remaining character slots will be filled with blanks.



```
8 DIGITS:
0
```

New/Change Password

Select the NEW/CHNG PASSWORD to set or modify the CPU password and restriction class. After you select (NEW/CHNG PASSWORD), you will see the sequence of screens shown above.

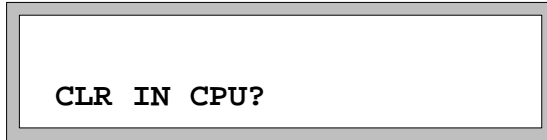
After you enter the CPU password, you are prompted to reenter the password for validation. You must enter the password in the same way as before. If it is different from your first entry, a message is given and you must restart the entire procedure.

You can use the key to cancel this function.



After the password has been verified, you are prompted to enter the restriction class. Select the level of restriction for the CPU and confirm your selection by pressing the key. Please refer to the *SIMATIC S7-200 Programmable Controller System Manual* for information about the different restriction levels.

Delete Password

Select the DELETE PASSWORD function to delete a previously set CPU password. This function is allowed only when you have already entered the correct CPU password. If the password entered is correct, the following message is displayed:







CLR IN CPU?

Enter  for yes or  for no.

If you attempt to delete the CPU password without having previously entered the password, the following message is displayed:








NEED PASSWORD

You must use the    (ENTER PASSWORD) key combination to enter the correct CPU password or press the  key to terminate the function.

Password Logout

Select the PASSWORD LOGOUT function to log off the CPU before disconnecting the PG 702 so that the password protection is immediately in effect. If this function is not used before you disconnect the PG 702 from the CPU, the CPU can be accessed without entering a password for up to two minutes. This means another user could connect within the two-minute time frame and edit the program without having to enter the password.

If You Forget Your Password

If you forget your CPU password, you can regain access to your CPU, but not to the information contained in the CPU. You can clear all CPU memory to recover the CPU by pressing      .

4.12 Debug Read/Write Variables PG-Menu 8

Use the DEBUG R/W VARS function shown in Table 4-17 to test the program. Chapter 6 describes these functions in more detail. Press PG-Menu and scroll down to view these functions.

Table 4-17 Debug Read/Write Variables

Menu Text on Display	Description of Function	Key Sequence
8 DEBUG R/W VARS>	Single scan and read/write variables	8
1 STATUS CHART>	Read/write variables	8 1
1 SETUP CHART	Create variable list	8 1 1
2 WRITE CHART VARS	Write variable list	8 1 2
3 READ CHART VARS	Read variable list	8 1 3
2 FORCE FUNCTIONS>	Force variables and I/O	8 2
1 FORCE VARIABLES	Force variables	8 2 1
2 FORCE I/O	Force I/O	8 2 2
3 UNFORCE ALL VARS	Unforce all variables and I/O	8 2 3
3 SINGLE SCAN	Execute single scan	8 3

5

Editing STL Programs

This chapter explains how to edit lines of STL instructions, how to display instruction lines, and how to use editing aids (such as insert, delete and find).

Chapter Overview

Section	Description	Page
5.1	Using the Edit Functions	5-2
5.2	Setting the Edit Mode and Loading the Program into the CPU	5-3
5.3	Entering Instructions	5-4
5.4	Entering Operands	5-8
5.5	Using the Display for Editing	5-11
5.6	Using Editing Aids	5-14
5.7	GOTO Network or Line Number	5-15
5.8	Find Operand or Instruction	5-16

5.1 Using the Edit Functions

Editing Programs

To edit an existing program, connect the PG 702 to the CPU, and press the **PG-Menu** **1** **1** key sequence (CPU PROGRAM BLOCK, UPLOAD FROM CPU). This loads the program to the PG 702 and sets the PG 702 to program block edit mode. All modifications of the program are first made in the PG 702. You must then download the program to the CPU.

Keys

In the program block edit mode, you can enter instructions and operands using designated keys. A given key is activated depending on the cursor position. Sections 5.3 and 5.4 contains more information about these keys.

Statement List Program

A statement list (STL) program consists of statement list statements that are delimited by at least one network. A statement list statement is in the following form:

instruction [operand] [,operand] [,operand] where [] indicate optional elements

The operand is the data to be operated on, or a description of the location of the data to be operated on, by the instruction. Operands take two forms:

- Operands are broken down into addressing mode, operand identifier, size, and offset. Addressing modes are either direct or indirect.
- Constants are either integer, hex, hex ASCII, or floating point numbers.

See the *SIMATIC S7-200 Programmable Controller System Manual* for more details.

5.2 Setting the Edit Mode and Loading the Program into the CPU

Program Block Edit Mode

You can edit or create a user program using the PG 702. The program block edit mode is set automatically when you upload a program to the PG 702. You will see a display such as the following one:

```
LD  I0.0
ED:      1:1
```

Downloading a Program to the CPU

To download the program to the CPU, use the **FCT** **5** key sequence (DOWNLOAD TO CPU) or the **PG-Menu** **1** **2** key sequence (PROGRAM BLOCK, DOWNLOAD TO CPU).

```
LOAD TO CPU?
```

You can load the program to the CPU or abandon it.

- Press **↵** to load the program to the CPU.
- Press **CLR** to abandon the function and return to the mode you were in when you initiated the download.

If the S7-200 CPU is password-protected and the password has not yet been entered, you must enter a password (see Section 4.11).

5.3 Entering Instructions

Instruction Entry Function

You can now change or enter instructions. Each erroneous entry is rejected with a system message or a beep tone.

Keys for Entering Instructions

Figure 5-1 shows you the keys that are active when you enter instructions.

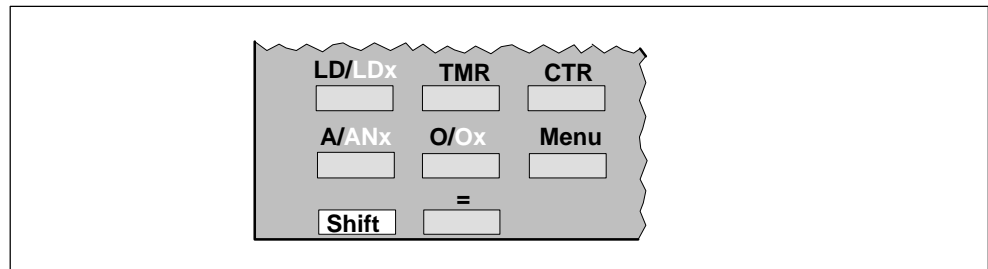


Figure 5-1 Keys for Entering Instructions

Entering Instructions

To enter instructions, position the cursor on the instruction line (see Section 5.5). You can enter instructions in two different ways:

- Select an instruction by pressing the corresponding instruction key. See Section A.1 for a complete list of instructions with corresponding keys on the PG 702.
- Select an instruction by pressing the **Menu** key. This action brings up a list of instruction categories. Enter the numerical value that corresponds to the proper instruction category. A > character at the end of the line means the category is subdivided. If the PG 702 display area is not large enough for the instructions, you can scroll in the menu area or activate additional submenus. See Section A.2 for a complete list of the instruction set menus by category.


The last page of this manual contains an alphabetical listing of all instructions and the keystrokes you use to enter them.

Entering Instructions with Instruction Keys

You can enter some instructions without using a menu:

- Press one of the keys shown in Figure 5-1 to display the instruction in front of the slash.
- Press the key repeatedly to display the negated form of the instruction, for example, LDN, AN or ON.

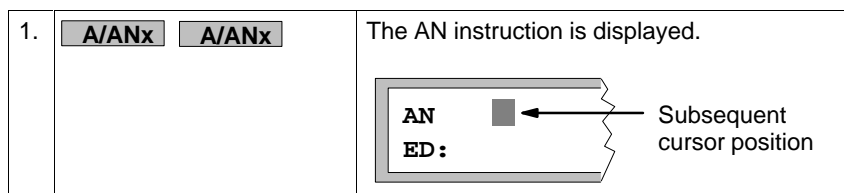
Note

If you press a key not shown in Figure 5-1, the entry is rejected with the “SYNTAX ERROR” message. This message must always be confirmed with . A list of the relevant keys is included in Section A.1.

Example: Entry Via Key

The example below shows you how to enter an instruction (AN). The cursor must be in the instruction entry field (at the beginning of the first line).

Proceed as follows:



This example is not complete and serves to explain this input sequence only.

Entering an Instruction with a Selection Menu

Some instructions are not displayed directly; you must select them from a menu. Section A.1 describes these instructions and the sequence to select them. You can activate the selection menus by:

- Pressing **Shift** and then one of the **LD/LDx**, **A/ANx**, or **O/Ox** keys. The instruction after the slash is active. The first instructions in the relevant menu are displayed. For example:

```

1 AI
2 ANI
    
```

An instruction in the menu that has a > at the end of the line indicates that there are additional menus. To open another menu, activate this function by pressing the number key that matches the number in front of the instruction on the PG 702 display.


- Pressing **Menu** to access all instructions that are not displayed with other keys. For example:

```

1 PROGRAM CONTROL >
2 ROTATE & SHIFT >
    
```

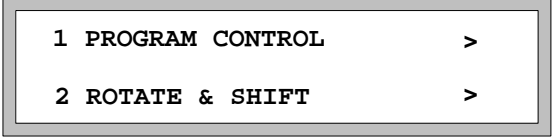




Procedure for Entering from a Menu

Use this procedure to select an instruction from a menu:

1. Select the desired menu by pressing the appropriate number key or using the  key if the function appears in the upper line.
2. Select an instruction by pressing the appropriate number key. The number key for the particular instruction is located in front of the instruction in the menu. You can also enter an instruction not shown currently on the display.

Example: Using a Selection Menu to Enter an Instruction

The example below shows you how to enter the rotate instruction, RLD, using a selection menu. Proceed as follows:

1.	Menu	<p>The following menu is displayed.</p> 
2.	2	<p>The submenu for rotate instructions is displayed.</p> 
3.	2	<p>Select the rotate left instructions.</p> 
4.	↓	<p>Scroll in the menu until the RLD instruction is displayed.</p> 
5.	3	<p>The RLD instruction is displayed.</p> 

These instructions are not complete and serve to explain this input sequence only.

5.4 Entering Operands

Operand Entry Function

After you enter the instruction, the cursor is in the entry field for the operand (see illustration of display in Section 5.5). The PG 702 is ready to accept only an operand. Erroneous operator entries are rejected with a message or a beep.

After you enter the operand identifier, the cursor is positioned to the entry field for the operand offset. The PG 702 is ready to accept only an operand offset. Erroneous operator entries are rejected with a message.

Keys for Entering Operand Identifiers and Operand Offsets

Figure 5-2 shows the keys that can be used for entering operand identifiers or operand offsets.

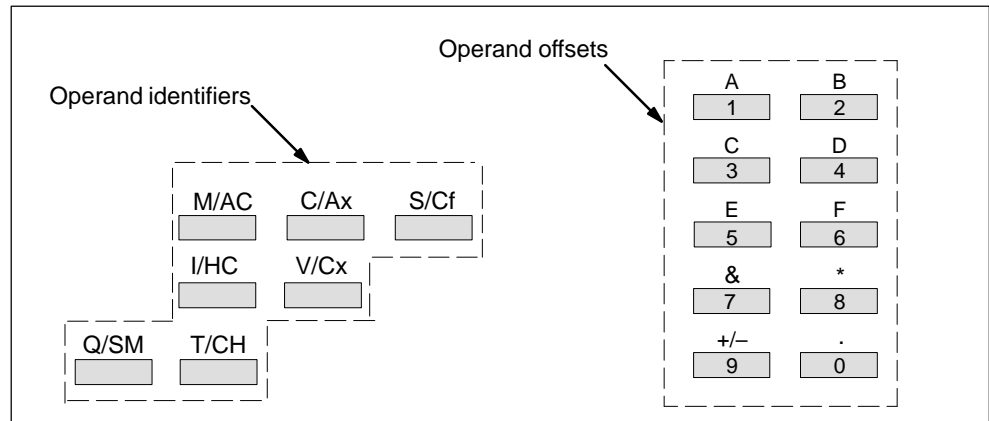


Figure 5-2 Keys for Entering Operand Identifiers and Operand Offsets

Operand Entry Styles



Entering an operand depends on the respective instruction. A beep signal rejects an operand that does not belong to the current instruction. For example, an LDB = instruction permits only byte operands, such as VB, IB, or QB. You can enter an operand identifier:

- Individually by using the operand key (see Figure 5-2)
- Individually with the **Shift** key and with an operand identifier key
- By using the **→** or **←** to scroll across each character in an instruction
- By using the **Shift** **→** or **Shift** **←** to jump to another instruction element

Section A.3 describes the operand identifier keys.

Entering Operand Offsets

You can enter the operand offsets as follows:

- With the number keys (see Figure 5-2)
- Using the  to enter the period between the individual digits (for example, 0.0)
- Using the  to acknowledge the error message to an erroneous input

Entering an Individual Operand



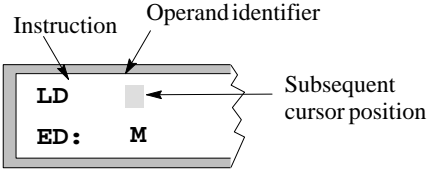



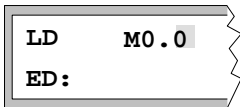
When you press a key or key combination, an operand identifier or operand offset is entered starting at the cursor position. If you enter an illegal key combination, a warning beep sounds and the key combination is rejected. A corresponding operand is displayed when you enter a specific key combination. The cursor moves to the entry field for an operand offset.

You enter the operand in two steps:

1. Enter the operand identifier (for example, I, Q, M).
2. Enter the operand offset (for example, 0.0 or 2222).

The example below shows you how to enter an operand directly.

Prerequisite: The cursor must be in the entry field for instructions.

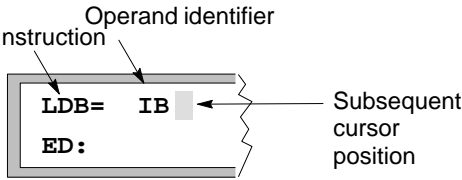
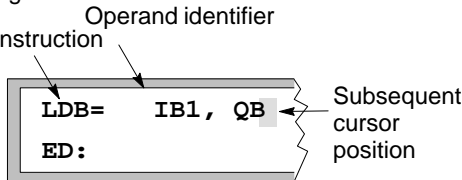
	Key	Result
1.		LD is displayed.
2.		Operand M is displayed. 
3a.		Enter 0
b.		Enter "." (period inserted automatically)
c.		Enter 0 

Entering Multiple Operands

An instruction can have more than one operand. Each operand is separated by a comma.

The example below shows you how to enter an instruction that has more than one operand.

Prerequisite: The cursor must be positioned on the entry field for instructions.

	Press Key	Display
1.	Shift LD/LDx 3	The menu with the LDB= instruction is displayed.
2.	1	The LDB= instruction is displayed in the entry field for instructions.
3.	I/HC	The IB operand is displayed. 
4.	1	Operand offset 1 is displayed in the entry field for operand offsets.
5.	→	The cursor is positioned in the entry field for the second operand.
6.	Q/SM	The QB operand is displayed, and a comma is generated. 
7.	1 ←	Operand offset 1 is displayed in the entry field for the second operand offset. Terminate entry.

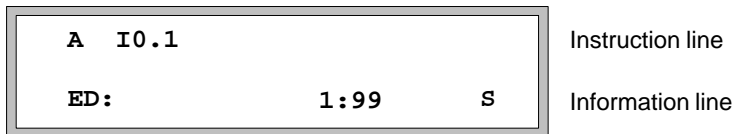
5.5 Using the Display for Editing

PG 702 Display

The PG 702 display shown below gives you information about your S7-200 program instruction and the PG 702 function being performed.

The display is subdivided in two lines:

- Instruction line
- Information line



Instruction Line

The instruction line consists of an instruction and its operands.

- The instructions are displayed in their full length.
- The operand identifier is the first part of the operand.
- The operand size is the second part of the operand.
- The operand offset is the third part of the operand.

Figure 5-3 shows an instruction line with the character spacing allowed. The information line normally seen in a PG 702 display is not shown in this example.

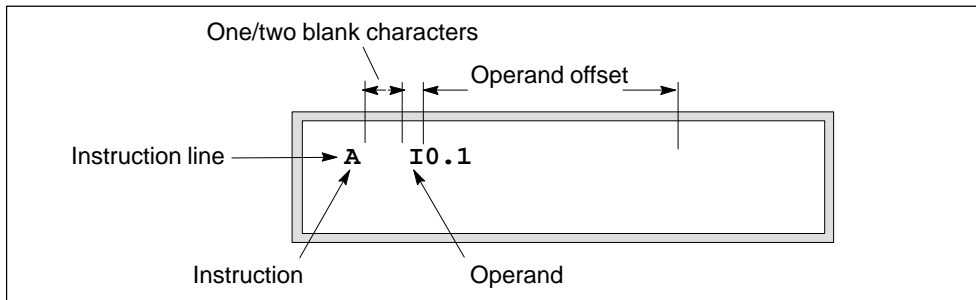


Figure 5-3 Sample Instruction Line with Character Spacing

If the complete instruction line cannot be shown on the display, you can scroll through the instruction line with the **→** key or **Shift** **→** keys.

Information Line

An information line is always displayed in the program block edit mode. This line contains supplementary information about the instruction line. See Figure 5-4. The information line displays information related to the particular instruction line, and warnings and error messages.

The information line displays the active mode, the current network and instruction lines, and the Shift (S) display.

- The active mode display shows ED (program block edit mode).
- The network and instruction line display shows the number of the currently displayed network and instruction line. The first network or line has number 1.
- The Shift display shows when the **Shift** key is pressed.
 - A flashing S means that the **Shift** key is pressed. The S remains until you press another key.
 - A blank Shift display means that the **Shift** key is not pressed.

The instruction line normally seen in a PG 702 display is not shown in Figure 5-4.

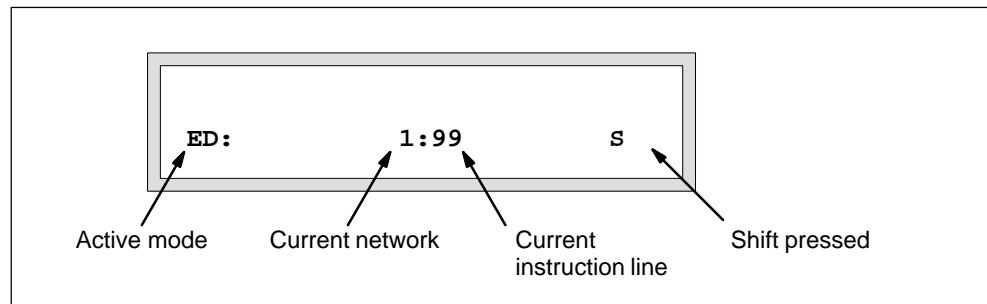


Figure 5-4 Sample Information Line with Supplementary Information

Information Line with Warnings and Error Messages













Warnings and error messages also appear in the information line.

- Warnings disappear automatically the next time you make an entry. They do not have to be confirmed.
- Acknowledge error messages with **←**.

Positioning the Cursor

As soon as you have loaded a program block, the cursor points to the first instruction in the instruction line. You can position the cursor with the keys shown in Table 5-1.

Table 5-1 Explanations for Positioning the Cursor in the Instruction Line



Cursor Movement	Key
Positions the cursor inside an instruction line	 
Positions the cursor to the beginning of the next line Positions the cursor to the beginning of the previous line	 
Positions the cursor to the beginning of the network preceding the current network	 
Positions the cursor to the beginning of the network following the current network	 
Positions the cursor to the beginning of the next instruction element	 
Positions the cursor to the beginning of the previous instruction element	 

Deleting Parts of an Instruction or Complete Instruction Lines

Always use the **CLR** key to delete a part of, or a complete instruction line. The following items are deleted depending on the position of the cursor:

- Position the cursor at the beginning of the instruction or at the beginning of the operand identifier to delete everything to the right of the cursor.
- Position the cursor inside the operand offset or jump label to delete the character at the cursor location.

Terminating an Entry

After you have made all of the necessary changes and additions, you can terminate an instruction line entry with  or . The next line is displayed.

Canceling an Entry

To cancel an entry in the current instruction line, use the **CLR** key to clear the current instruction line. Then use **Shift CLR** to restore the previous entry.

5.6 Using Editing Aids

You have a number of functions available for editing your program in the program block edit mode. Press the **FCT** key to activate the menus with the editing aids.

The selection menu with the first two editing aids functions is displayed:



Editing Aids Available

The editing aids described in Table 5-2 are available after you press the **FCT** key. The following sections contain additional information on GOTO and FIND.

Table 5-2 Editing Aids

Function	Key	Explanation
INSERT >	1	Inserts a new (blank) line or network
INSERT LINE	1 1	Inserts a new (blank) line in front of the current line in the program (to be filled subsequently with an instruction)
INSERT NETWORK	1 2	Inserts a new (blank) network in front of the current network in the program (to be filled subsequently with instructions)
DELETE >	2	Deletes the current line or network
DELETE LINE	2 1	Deletes the current line in the program and moves the cursor to the next line
DELETE NETWORK	2 2	Deletes the current network in the program and moves the cursor to the next network
GOTO	3	Goes to specified network and line number
FIND>	4	Search program
OPERAND	4 1	Searches for operand
INSTRUCTION	4 2	Searches for instruction
REPEAT	4 3	Repeats last search
DOWNLOAD TO CPU	5	Downloads program block to CPU

5.7 GOTO Network or Line Number

To go to a specific network and line number, use the **FCT** **3** key combination. The display shows the current network, line 1.

```
NETWORK #: 1
LINE #: 1
```


Enter the network number and line number that you want to go to and press the  key. The editor moves to the specified statement and displays it as the current line. If you enter a network number that is greater than the highest network number in your program, the PG 702 displays the specified line in the last network. If you enter a line number that is greater than the highest line in a specific network, the PG 702 displays the last line in that network. All statement list programs begin with network 1 and line 1. If you enter a zero, the PG 702 displays the first line or network. Table 5-3 contains examples of GOTO entries and results.


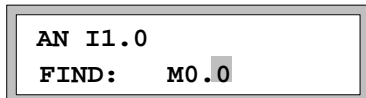
Table 5-3 GOTO Results

Entry	Action
<pre>NETWORK #: 0 LINE #: 0</pre>	Goes to network 1, line 1
<pre>NETWORK #: 4 LINE #: 0</pre>	Goes to network 4, line 1
<pre>NETWORK #: 4 LINE #: 1</pre>	Goes to network 4, line 1
<pre>NETWORK #: 3000 LINE #: 0</pre>	Goes to last network, line 1
<pre>NETWORK #: 3000 LINE #: 500</pre>	If there are only 10 lines in the last network, this entry goes to the last line in the last network.


5.8 Find Operand or Instruction

The tables on the following pages explain the keys you must press to activate the individual FIND functions.

Operand: To find an operand in a program block, follow these directions.

	Activity	Key Sequence	Result
1.	Activate the FIND > OPERAND function	FCT 4 1	 <p>The display shows the current instruction in line 1. Enter the operand to find with the same keystrokes used when editing programs.</p>
2.	Enter the operand you are looking for	M/AC 0 → 0	
3.	Start the search run	←	<p>Result of the search run:</p> <ul style="list-style-type: none"> • Operand found. The cursor is on the line that contains the specified operand. • Operand not found. The cursor remains in its previous position. The "NOT FOUND" message appears in the information line.

Instruction: To find an instruction in a program block, follow these directions.

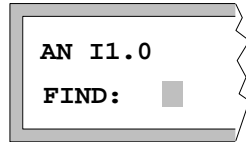
	Activity	Key Sequence	Result
1.	Activate the FIND > INSTRUCTION function	FCT 4 2	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> AN I1.0 FIND: █ </div> <p>The display shows the current instruction in line 1. Enter the instruction to find with the same keystrokes or menu numbers used when editing programs.</p>
2	Enter the instruction you are looking for	LD/LDx	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> AN I1.0 FIND: LD </div>
3	Start the search run		<p>Result of the search run:</p> <ul style="list-style-type: none"> • Instruction found. The cursor is on the line that contains the specified instruction. • Instruction not found. The cursor remains in its previous position. The “NOT FOUND” message appears in the information line.

Repeat: To repeat the last FIND function, follow these directions.

	Activity	Key Sequence	Result
	Activate the FIND > REPEAT function	FCT 4 3	This function is only valid when a FIND function was previously invoked. Otherwise, the error message “NOT FOUND” is displayed. The last FIND function is invoked again.

Example: Entering the Object You Are Searching For

When you activate a FIND function, an entry field appears in the second line of the display.



- Enter the object (instruction/operand) you are searching for after "FIND."
- Enter operands with the same key strokes used when editing programs (see Section 5.4). The located result appears in the first line of the display.
- Enter instructions with the same key strokes or menu numbers used when editing programs (see Section 5.3).

6

Debugging and Testing the Program

This chapter contains tips on debugging and troubleshooting programs created with the PG 702.

Chapter Overview

Section	Description	Page
6.1	Debug Functions (PG-Menu Function 8)	6-2
6.2	Status Chart (PG-Menu Function 8, 1)	6-3
6.3	Forcing and Unforcing Variables and I/O (PG-Menu Function 8, 2)	6-5
6.4	Single Scan Debug Function	6-8

6.1 Debug Functions **PG-Menu** **8**

Debug functions are intended to aid in troubleshooting the S7-200 system. Debug functions include single scan, forcing and unforcing I/O and variables, and reading and writing I/O and variables. A complete list of the debug functions available is shown in Table 6-1.

Table 6-1 Debug Functions Menu

Menu Text on Display	Description of Function	Key Sequence
1 STATUS CHART>	Read and write CPU variables	8 1
1 SETUP CHART	Create monitor list of variables to monitor	8 1 1
2 WRITE CHART VARS	Write variables in monitor list	8 1 2
3 READ CHART VARS	Read variables in monitor list	8 1 3
2 FORCE FUNCTIONS>	Forcing functions, variable and I/O	8 2
1 FORCE VARIABLES	Force specified variables	8 2 1
2 FORCE I/O	Force specified bits	8 2 2
3 UNFORCE ALL VARS	Unforce all variables	8 2 3
3 SINGLE SCAN	Perform single scan operation	8 3

6.2 Status Chart PG-Menu 8 1

Setting up the Status Chart PG-Menu 8 1 1

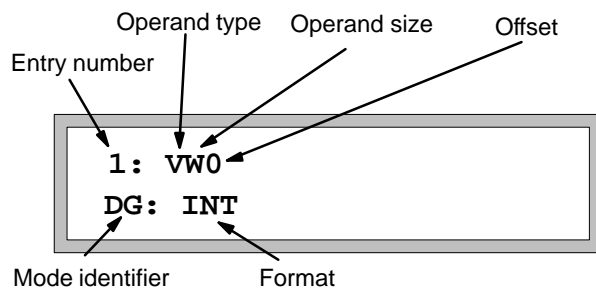
Select the SETUP CHART function to define the list of variables to be monitored. You can define up to nine variables. After you have started this function, you can press the **FCT** key to access the PG editing functions shown in Table 6-2.

Table 6-2 PG Functions Available in Status Chart Setup

Menu Text on Display	Brief Description of Function
1 INSERT LINE	Inserts a new line before the current one in the list of variables
2 DELETE LINE	Deletes the current line in the variable list and moves the cursor to the next line
3 EXIT	Exits the status chart setup function. The chart is lost if a power cycle is performed

When you first apply power to the PG 702, this list is empty. Use the operand keys to fill the list with one operand per line. You can insert and delete lines in the same way that you use the STL editor. Enter the first operand and press **←** to move the cursor to the next line. INT is displayed. Press **←** to accept the default integer format or use the **Shift V/Cx** key combination to toggle the format through unsigned integer, hexadecimal, and floating point number. Press **←** to confirm your selection and move to the next entry. The format for chart variables can only be edited during chart setup. The format remains in effect until it is changed or a power cycle is performed.

The following is an example of the display:



Note

Floating point format requires a double word variable.

Press the **FCT** **3** key sequence to exit the setup function and return to the PG menu. The status chart is available until the next power cycle. You must set up a status chart before you can read or write any variables.

Writing Status Chart Variables **PG-Menu** **8** **1** **2**

When you select the WRITE CHART VARS function, the cursor is automatically placed in the value field. Type in the value and press the **↵** key. The PG 702 immediately writes the value to the CPU. Your cursor then moves to the next variable in the chart and reads that value. If you are on the last variable in the chart, the PG 702 reads the same value again.

Use the **↑** and **↓** keys to scroll through the list of monitor values.

Entering Floating Point Numbers

To enter a floating point number, select the variable to be modified. Enter the number in floating point number format. You can use one of two different formats:

- [{+|-}] digits. [digits] (235.24)
- [{+|-}] digits. [digits] [E [{+|-}] digits] (-2.3456E-10)

Digits may be one or more decimal digits.

Reading Status Chart Variables **PG-Menu** **8** **1** **3**

When you select the READ CHART VARS function, the first variable and its value are displayed. Use the **↑** and **↓** keys to scroll through the status chart.

To change the format of the variable, return to the setup chart function and use the **Shift** **V/Cx** keystroke combination.

6.3 Forcing and Unforcing Variables and I/O PG-Menu 8 2






Select the FORCE VARIABLES function to view and edit force information for data variables, or select the FORCE I/O function to view and edit force information for input and output points.

Every time one of these functions is selected, the PG 702 reads and displays the force information contained in the CPU.

You can scroll through the list to view force information, force a variable to a different value, unforce the displayed variable, or scroll to the end of the list and add a new variable.

Use the keys listed in Table 6-3 to work with the force information.

Table 6-3 Valid Keys for Modifying the Force Information

Key	Function
	Scroll to next entry
	Scroll to previous entry
	Enter operand or value
	Move cursor left
	Move cursor right
CLR	Delete a character
Shift CLR	Cancel the operation and return to PG menu
FCT	Force function menu. Unforce an element
0 through 9	Digits to enter address or value
Operand keys	Enter the operand type

Adding a New Force Variable

Select the key combination **PG-Menu** **8** **2** **1**. The force variable list is read from the S7-200 CPU. If no values were forced, the following display is shown:

```
1 :
DG: 0
```

To force a value, enter the variable type and address and then press **↵**. The cursor is repositioned at the value field. Enter the value and press **↵**. The variable is now forced in the CPU. The PG 702 displays the PG menu.

If a force variable list already exists in the CPU, the PG 702 displays a screen like the following:

```
1 :  VB10
DG: 25
```

To enter a new variable, use the **↓** key to scroll to the end of the list until an empty entry is displayed:

```
10 :
DG: 0
```




Enter the operand and press **↵**. Enter the value and press **↵**. The variable is now forced in the CPU. The PG 702 displays the PG menu.

Changing the Format of a Variable

To change the format of a variable, toggle the **Shift** **V/Cx** keys after you enter the value field and before you start editing. The following formats are available:

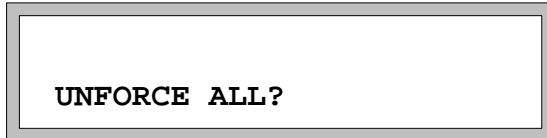
- Integer
- Unsigned integer
- Hexadecimal
- Floating point



Unforcing a Single Variable

To unforce a single variable, use the  key to scroll to the variable you want to unforce. Press  . The variable in the CPU is no longer forced. The PG 702 displays the PG menu.


Unforce All Variables

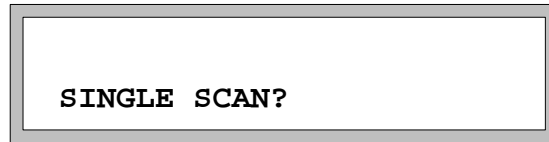
Select the UNFORCE ALL VARIABLES function to unforce all forced variables and I/O. When you select this function, the following message is displayed:




Press the  key to unforce all variables and I/O. Press the  key to cancel the function and leave the force status as it is.

6.4 Single Scan Debug Function PG-Menu 8 3

A single scan may be executed when the S7-200 CPU is in the STOP mode. When SINGLE SCAN is selected, you are prompted to press  to execute a single scan.



After you press , the command is executed and the function is cleared. If you press the **CLR** key, the function is cancelled and the display returns to the main PG menu.

Overview of Menus and Instruction Sets

A

Instructions are entered in one of three ways depending on the individual instruction.

Enter commonly used instructions like LD, OUT, and AND directly from the keyboard using the **LD**, **=** and **AND** keys respectively. Enter negated or similar versions of commonly used instructions, like LDN, ON, AN, directly from the keyboard by pressing the **LD**, **=** and **AND** keys twice. Enter variations of these instructions such as LD Immediate or AND Immediate by pressing **Shift** and the appropriate **LDx** or **ANx** keys. This brings up a menu of instructions from which to choose.

Access other instructions by pressing the **Menu** key. This action brings up a list of instruction categories. Enter the appropriate numerical value corresponding to the proper instruction category. A > character at the end of the line means the category is subdivided.

This appendix contains tables with complete listings of all instructions, assigned mnemonics and keystrokes to access the instructions as well as the instruction set menus.

For a quick reference for all instructions in alphabetical order, see the last page of this manual.

Chapter Overview

Section	Description	Page
A.1	Logic Operations	A-2
A.2	Instruction Set Menus	A-6
A.3	Operands	A-13
A.4	PG Menu Functions	A-16

A.1 Logic Operations

Table A-1 AND Instructions

Instruction	Instruction Function	Key Sequence
A	ANDs bit value	A/ANx
AN	ANDs negated bit value	A/ANx A/ANx
1 AI	ANDs bit value directly	Shift A/ANx 1
2 ANI	ANDs negated bit value directly	Shift A/ANx 2
3 ABx >	Following instructions via ABx	Shift A/ANx 3
1 AB=	ANDs result (bit value) of byte comparison (=)	Shift A/ANx 3 1
2 AB<=	ANDs result (bit value) of byte comparison (<=)	Shift A/ANx 3 2
3 AB>=	ANDs result (bit value) of byte comparison (>=)	Shift A/ANx 3 3
4 AWx >	Following instructions via AWx	Shift A/ANx 4
1 AW=	ANDs result (bit value) of word comparison (=)	Shift A/ANx 4 1
2 AW<=	ANDs result (bit value) of word comparison (<=)	Shift A/ANx 4 2
3 AW>=	ANDs result (bit value) of word comparison (>=)	Shift A/ANx 4 3
5 ADx >	Following instructions via ADx	Shift A/ANx 5
1 AD=	ANDs result (bit value) of byte comparison (=)	Shift A/ANx 5 1
2 AD<=	ANDs result (bit value) of byte comparison (<=)	Shift A/ANx 5 2
3 AD>=	ANDs result (bit value) of byte comparison (>=)	Shift A/ANx 5 3
6 ARx >	Following instructions via ARx	Shift A/ANx 6
1 AR=	ANDs result (bit value) of real comparison (=)	Shift A/ANx 6 1
2 AR<=	ANDs result (bit value) of real comparison (<=)	Shift A/ANx 6 2
3 AR>=	ANDs result (bit value) of real comparison (>=)	Shift A/ANx 6 3

Table A-2 OR Instructions

Instruction	Function of the Instruction	Key Sequence
O	ORs bit value	O/Ox
ON	ORs negated bit value	O/Ox O/Ox
1 OI	ORs bit value directly	Shift O/Ox 1
2 ONI	ORs negated bit value directly	Shift O/Ox 2
3 OBx >	Following instructions via OBx	Shift O/Ox 3
1 OB=	ORs result (bit value) of bytecomparison (=)	Shift O/Ox 3 1
2 OB<=	ORs result (bit value) of byte comparison (<=)	Shift O/Ox 3 2
3 OB>=	ORs result (bit value) of byte comparison (>=)	Shift O/Ox 3 3
4 OWx >	Following instructions via OWx	Shift O/Ox 4
1 OW=	ORs result (bit value) of word comparison (=)	Shift O/Ox 4 1
2 OW<=	ORs result (bit value) of word comparison (<=)	Shift O/Ox 4 2
3 OW>=	ORs result (bit value) of word comparison (>=)	Shift O/Ox 4 3
5 ODx >	Following instructions via ODx	Shift O/Ox 5
1 OD=	ORs result (bit value) of byte comparison (=)	Shift O/Ox 5 1
2 OD<=	ORs result (bit value) of byte comparison (<=)	Shift O/Ox 5 2
3 OD>=	ORs result (bit value) of byte comparison (>=)	Shift O/Ox 5 3
6 ORx >	Following instructions via ORx	Shift O/Ox 6
1 OR=	ORs result (bit value) of real comparison (=)	Shift O/Ox 6 1
2 OR<=	ORs result (bit value) of real comparison (<=)	Shift O/Ox 6 2
3 OR>=	ORs result (bit value) of real comparison (>=)	Shift O/Ox 6 3

Table A-3 Assignment, Set/Reset Instructions

Instruction	Function of the Instruction	Key Sequence
=I	Assignment	= =
1 S	Sets	Shift = 1
2 R	Resets	Shift = 2
3 SI	Sets bit value directly	Shift = 3
4 RI	Resets bit value directly	Shift = 4

Table A-4 Load Instructions

Instruction	Function of the Instruction	Key Sequence
LD	Lloads bit value	LD/LD _x
LDN	Lloads negated bit value	LD/LD _x LD/LD _x
1 LDI	Lloads bit value directly	Shift LD/LD _x 1
2 LDNI	Lloads negated bit value directly	Shift LD/LD _x 2
3 LDBx >	Following instructions via LDBx	Shift LD/LD _x 3
1 LDB=	Lloads result (bit value) of word comparison (=)	Shift LD/LD _x 3 1
2 LDB<=	Lloads result (bit value) of word comparison (<=)	Shift LD/LD _x 3 2
3 LDB>=	Lloads result (bit value) of word comparison (>=)	Shift LD/LD _x 3 3
4 LDWx >	Following instructions via LDWx	Shift LD/LD _x 4
1 LDW=	Lloads result (bit value) of word comparison (=)	Shift LD/LD _x 4 1
2 LDW<=	Lloads result (bit value) of word comparison (<=)	Shift LD/LD _x 4 2
3 LDW>=	Lloads result (bit value) of word comparison (>=)	Shift LD/LD _x 4 3
5 LDDx >	Following instructions via LDDx	Shift LD/LD _x 5
1 LDD=	Lloads result (bit value) of word comparison (=)	Shift LD/LD _x 5 1
2 LDD<=	Lloads result (bit value) of word comparison (<=)	Shift LD/LD _x 5 2
3 LDD>=	Lloads result (bit value) of word comparison (>=)	Shift LD/LD _x 5 3
6 LDRx >	Following instructions via LDRx	Shift LD/LD _x 6
1 LDR=	Lloads result (bit value) of real comparison (=)	Shift LD/LD _x 6 1
2 LDR<=	Lloads result (bit value) of real comparison (<=)	Shift LD/LD _x 6 2
3 LDR>=	Lloads result (bit value) of real comparison (>=)	Shift LD/LD _x 6 3

Table A-5 Timer/Counter Instructions

Instruction	Function of the Instruction	Key Sequence
TON	Starts timer as on-delay	TMR
TONR	Starts timer as retentive on-delay	TMR TMR
CTU	Counts upwards	CTR
CTUD	Counts upwards/downwards	CTR CTR
HSC	Activates fast counter	CTR CTR CTR
HDEF	Defines mode for fast counter	CTR CTR CTR CTR

A.2 Instruction Set Menus

The structure of the following menus activated by the **Menu** key has three levels. You can move between the individual levels in various ways.

- Between the instruction types of levels 1 to 3 using the numerical keys
- Between the individual instruction types of level 2 using the following keys:



All instructions and the relevant key combinations you have to enter for the display of these instructions are listed in the following tables. For general information on these instructions refer to Section 5.3.

Program Control **Menu** **1**

Table A-6 Label Instructions

Instruction	Function of the Instruction	Key Sequence
1 LABEL>	Label instructions	Menu 1 1
1 JMP	Jumps to jump label	Menu 1 1 1
2 LBL	Defines jump label	Menu 1 1 2
3 CALL	Calls up subroutine	Menu 1 1 3
4 SBR	Begins subroutine	Menu 1 1 4

Table A-7 Return Instructions

Instruction	Function of the Instruction	Key Sequence
2 RETURN>	Return instructions	Menu 1 2
1 RET	Unconditional return from subroutine	Menu 1 2 1
2 RETI	Unconditional return from interrupt routine	Menu 1 2 2
3 CRET	Conditional return from subroutine	Menu 1 2 3
4 CRETI	Conditional return from interrupt routine	Menu 1 2 4

Table A-8 Interrupt Instructions

Instruction	Function of the Instruction	Key Sequence
3 INTERRUPTS>	Interrupt instructions	Menu 1 3
1 INT	Starts interrupt routine	Menu 1 3 1
2 DISI	Disables all user interrupt events	Menu 1 3 2
3 ENI	Enables all user interrupt events	Menu 1 3 3
4 ATCH	Attaches interrupt routine to an event	Menu 1 3 4
5 DTCH	Detaches interrupt routine from an event	Menu 1 3 5

Table A-9 Miscellaneous Control Instructions

Instruction	Function of the Instruction	Key Sequence
4 SCRx >	Following instructions via SCRx	Menu 1 4
1 LSCR	Loads SCR bit onto logic stack	Menu 1 4 1
2 SCRT	SCR Transition	Menu 1 4 2
3 SCRE	SCR End	Menu 1 4 3
5 FOR	Executes the instructions between the FOR and the NEXT	Menu 1 5
6 NEXT	Marks the end of the FOR loop	Menu 1 6
7 STOP	Terminates the execution of the program immediately by causing a transition of the CPU from RUN to STOP	Menu 1 7
8 MEND	Unconditional end of user program	Menu 1 8
9 END	Conditional end of user program	Menu 1 9

Table A-10 Rotate and Shift Instructions

Instruction	Function of the Instruction	Key Sequence
1 RRx >	Rotate to the right	Menu 2 1
1 RRB	Rotates byte to the right	Menu 2 1 1
2 RRW	Rotates word to the right	Menu 2 1 2
3 RRD	Rotates double word to the right	Menu 2 1 3
2 RLx >	Rotate to the left	Menu 2 2
1 RLB	Rotates byte to the left	Menu 2 2 1
2 RLW	Rotates word to the left	Menu 2 2 2
3 RLD	Rotates double word to the left	Menu 2 2 3
3 SRx >	Shift to the right	Menu 2 3
1 SRB	Shifts byte to the right	Menu 2 3 1
2 SRW	Shifts word to the right	Menu 2 3 2
3 SRD	Shifts double word to the right	Menu 2 3 3
4 SLx >	Shift to the left	Menu 2 4
1 SLB	Shifts byte to the left	Menu 2 4 1
2 SLW	Shifts word to the left	Menu 2 4 2
3 SLD	Shifts double word to the left	Menu 2 4 3
5 SHRB	Shifts value to shift register	Menu 2 5

Table A-11 Table Instructions

Instruction	Function of the Instruction	Key Sequence
1 ATT	Enters the value in the table	Menu 3 1
2 LIFO	Removes last value from the table	Menu 3 2
3 FIFO	Removes first value from the table	Menu 3 3
4 FND=	Locates value in the table: is equal	Menu 3 4
5 FND<>	Locates value in the table: not equal	Menu 3 5
6 FND>	Locates value in the table: is greater than	Menu 3 6
7 FND<	Locates value in the table: is less than	Menu 3 7

Table A-12 Conversion Instructions

Instruction	Function of the Instruction	Key Sequence
1 BCDI	Converts BCD into integer	Menu 4 1
2 IBCD	Converts integer into BCD	Menu 4 2
3 ENCO	Converts a hexadecimal digit into a bit position	Menu 4 3
4 DECO	Converts a bit position into a hexadecimal digit	Menu 4 4
5 SEG	Generates a bit pattern for a 7-segment display	Menu 4 5
6 ATH	Converts ASCII string into a hexadecimal value	Menu 4 6
7 HTA	Converts a hexadecimal value into an ASCII string	Menu 4 7
8 DTR	Converts a double word integer into a real number	Menu 4 8
9 TRUNC	Converts a real value into integer	Menu 4 9

Table A-13 Word Logic Instructions

Instruction	Function of the Instruction	Key Sequence
1 ANDx >	AND	Menu 5 1
1 ANDB	ANDs bytes	Menu 5 1 1
2 ANDW	ANDs words	Menu 5 1 2
3 ANDD	ANDs double words	Menu 5 1 3
2 ORx >	OR	Menu 5 2
1 ORB	ORs bytes	Menu 5 2 1
2 ORW	ORs words	Menu 5 2 2
3 ORD	ORs double words	Menu 5 2 3
3 XORx >	Exclusive OR instruction	Menu 5 3
1 XORB	Exclusive ORs instruction (bytes)	Menu 5 3 1
2 XORW	Exclusive ORs instruction (words)	Menu 5 3 2
3 XORD	Exclusive ORs instruction (double words)	Menu 5 3 3
4 INVx >	Ones complement of integer	Menu 5 4
1 INVB	Ones complement of integer (8-bit)	Menu 5 4 1
2 INVW	Ones complement of integer (16-bit)	Menu 5 4 2
3 INVD	Ones complement of integer (32-bit)	Menu 5 4 3

Table A-14 Data Move Instructions

Instruction	Function of the Instruction	Key Sequence
1 MOVB	Transfers a byte	Menu 6 1
2 MOVW	Transfers a word	Menu 6 2
3 MOVD	Transfers a double word	Menu 6 3
4 MOVR	Transfers a real number (double word)	Menu 6 4
5 SWAP	Exchanges the bytes of a word	Menu 6 5
6 BMB	Transfers specified number of bytes	Menu 6 6
7 BMW	Transfers specified number of words	Menu 6 7
8 BMD	Transfers specified number of double words	Menu 6 8
9 FILL	Fills memory with bit pattern	Menu 6 9

Table A-15 Stack Instructions

Instruction	Function of the Instruction	Key Sequence
1 EU	Positive edge recognition	Menu 7 1
2 ED	Negative edge recognition	Menu 7 2
3 ALD	ANDs the first and second stack levels	Menu 7 3
4 OLD	ORs the first and second stack levels	Menu 7 4
5 LPS	Duplicates top of stack value	Menu 7 5
6 LRD	Copies second stack value	Menu 7 6
7 LPP	Shifts top of stack value from the stack	Menu 7 7
8 NOT	Inverts top of stack value	Menu 7 8

Table A-16 Mathematical Instructions

Instruction	Function of the Instruction	Key Sequence
1 + >	Add instruction	Menu 8 1
1 +I	Adds two integers (16-bit)	Menu 8 1 1
2 +D	Adds two integers (32-bit)	Menu 8 1 2
3 +R	Adds two reals (32-bit)	Menu 8 1 3
2 – >	Subtract instruction	Menu 8 2
1 –I	Subtracts one integer from another (16-bit)	Menu 8 2 1
2 –D	Subtracts one integer from another (32-bit)	Menu 8 2 2
3 –R	Subtracts one real from another (32-bit)	Menu 8 2 3
3 INCx >	Increment instruction	Menu 8 3
1 INCB	Increments byte by one	Menu 8 3 1
2 INCW	Increments word by one	Menu 8 3 2
3 INCD	Increments double word by one	Menu 8 3 3
4 DECx >	Decrement instruction	Menu 8 4
1 DECB	Decrements byte by one	Menu 8 4 1
2 DECW	Decrements word by one	Menu 8 4 2
3 DECD	Decrements double word by one	Menu 8 4 3
5 MUL	Multiplies integers	Menu 8 5
6 *R	Multiplies real	Menu 8 6
7 DIV	Divides one integer by another	Menu 8 7
8 /R	Divides real	Menu 8 8
9 SQRT	Square root	Menu 8 9

Table A-17 Miscellaneous Instructions

Instruction	Function of the Instruction	Key Sequence
1 COMMUNICATIONS>	Communication instructions	Menu 9 1
1 NETR	Network read	Menu 9 1 1
2 NETW	Network write	Menu 9 1 2
3 XMT	Transfers message from the buffer	Menu 9 1 3
4 RCV	Receives message into the buffer	Menu 9 1 4
2 TODR	Reads out the clock time and date	Menu 9 2
3 TODW	Writes the clock time and date	Menu 9 3
4 WDR	Resets the monitoring time	Menu 9 4
5 PID	PID loop instruction	Menu 9 5
6 NOP	Do-nothing instruction	Menu 9 6
7 PLS	Pulse instruction	Menu 9 7

A.3 Operands

You can enter operands using the PG 702 in either the edit program block (ED) mode or the debug (DG) mode.

Entering operands in the edit program block mode is context-sensitive. If you select an instruction that operates on words, the PG 702 automatically selects the correct data size for the data type.

Entering data types in the debug mode is not context-sensitive. In this mode, you must press the appropriate data type key. The first time the appropriate data type key is pressed, the byte size is displayed. The second time displays the word size and the third time displays the double word size. Pressing the key again returns the data type to byte size.

The following tables describe the key sequences for addresses and address values.

Note

If the addresses listed in the tables below are not accepted, this is usually because the instruction and address are not allowed in that particular combination.

Key Functions

The keys are depicted in Table A-18 to show which function of the key is active when entering addresses and address values.



Key is activated for the function marked above the key.



Key is activated for the function stated on the key.

Table A-18 Operands

Address Identifier	Description	Key Sequence
Q QB QW QD	Dependent on the instruction Output Output byte Output word Output double word	Q / SM
I IB IW ID	Dependent on the instruction Input Input byte Input word Input double word	I / HC
M MB MW MD	Dependent on the instruction Bit Memory Bit Memory byte Bit Memory word Bit Memory double word	M / AC
S SB SW SD	Dependent on the instruction Sequence Control Relay Memory Sequence Control Relay Memory byte Sequence Control Relay Memory byte word Sequence Control Relay Memory byte word double word	S / Cf
SM SMB SMW SMD	Dependent on the instruction Special Bit Memory Special Bit Memory byte Special Bit Memory word Special Bit Memory double word	Shift Q / SM
V VB VW VD	Dependent on the instruction Variable Memory Variable Memory byte Variable Memory word Variable Memory double word	V / Cx
AC	Accumulator	Shift M / AC
AIW	Analog inputs	Shift C / Ax

Table A-18 Operands

Address Identifier	Description	Key Sequence
AQW	Analog outputs	Shift C / Ax Shift C / Ax
T	Timer	T / CH
C	Counter	C / Ax
HC	High-speed counter	Shift I / HC
16#	Hexadecimal constant: Byte constant Word constant Double word constant	Shift T / CH
16"	Hexadecimal ASCII constant: Byte constant Word constant Double word constant	Shift T / CH Shift T / CH
#	Decimal constant Byte constant Word constant Double word constant	Shift V / Cx
#	Floating Point constant Double word constant	Shift S / Cf
*	Pointer for indirect addressing	Shift * 8
&	Address of a memory location for indirect addressing	Shift & 7
.	Period in operand offset (for example, 0.0) Dependent on the cursor position	→
.	Decimal point (for example, 23.56) for entering floating point constants	Shift * 8
,	Comma (between the individual addresses, generated automatically)	→
E	Exponent for scientific notation	Shift 5

A.4 PG Menu Functions

The PG menu functions allow you to select several services supported by the CPU or PG devices such as CPU mode changes, memory contents transfers, password protection and setting the time-of-day (TOD) clock. Table A-19 contains a complete list of PG menu functions.

Table A-19 PG Menu Functions

Menu Text on Display	Description of the Function	Key Sequence
1 CPU PROGRAM BLOCK >	Uploads/Downloads Program Block (OB1)	1
1 UPLOAD FROM CPU	Uploads program block (OB1) from CPU and enter editor	1 1
2 DOWNLOAD TO CPU	Downloads program block (OB1) from PG memory to CPU memory	1 2
3 EDIT PROG BLOCK	Edits program block (OB1)	1 3
2 CPU CONFIGURATION >	CPU Configuration	2
1 RETENTIVE RANGES	Sets retentive ranges	2 1
2 OUTPUT TABLE	Sets up output table	2 2
1 FREEZE OUTPUTS	Freezes the outputs at their current state	2 2 1
2 SET OUTPUT TABLE	Selects the state for each output	2 2 2
3 SET INPUT FILTERS >	Sets input filter times	2 3
1 I0.0 → I0.3	Sets the input filter times for I0.0 through I0.3	2 3 1
2 I0.4 → I0.7	Sets the input filter times for I0.4 through I0.7	2 3 2
3 I1.0 → I1.3	Sets the input filter times for I1.0 through I1.3	2 3 3
4 I1.4 → I1.5	Sets the input filter times for I1.4 through I1.5	2 3 4
4 PORT 0 PARAMETERS >	Sets/reads port 0 parameters	2 4
1 SET PORT 0 ADDR	Sets/reads port 0 address	2 4 1
2 SET BAUD RATE	Sets baud rate	2 4 2
3 SET HSA	Sets highest station address	2 4 3
4 SET RETRY COUNT	Sets retry count	2 4 4
5 SET GAP UPDATE	Sets gap update factor	2 4 5

Table A-19 PG Menu Functions

Menu Text on Display	Description of the Function	Key Sequence
5 PORT 1 PARAMETERS >	Sets/reads port 1 parameters	2 5
1 SET PORT 1 ADDR	Sets/reads port 1 address	2 5 1
2 SET BAUD RATE	Sets baud rate	2 5 2
3 SET HSA	Sets highest station address	2 5 3
4 SET RETRY COUNT	Sets retry count	2 5 4
5 SET GAP UPDATE	Sets gap update factor	2 5 5
6 DP PORT PARMS >	Sets/reads DP port parameters	2 6
1 SET DP PORT ADDR	Sets the CPU station address	2 6 1
2 READ DP STATUS	Reads DP status	2 6 2
3 CPU DATA BLOCK >	Edits Data Block	3
1 UPLOAD FROM CPU	Uploads data block (DB1) from CPU and enter editor	3 1
2 DOWNLOAD TO CPU	Downloads data block (DB1) from PG memory to CPU memory	3 2
3 EDIT DATA BLOCK	Edits data block (DB1)	3 3
4 FLASH OPERATIONS >	Transfers and Compares	4
1 CPU MEM > PG FLASH	Transfers CPU memory to PG flash	4 1
2 PG FLASH > CPU MEM	Transfers PG flash to CPU memory	4 2
3 CPU MEM = PG FLASH	Compares program in flash to program in CPU	4 3
5 CPU FUNCTIONS	Executes CPU functions	5
1 PUT CPU IN RUN	Starts user program execution	5 1
2 PUT CPU IN STOP	Stops user program execution	5 2
3 CLEAR CPU MEMORY >	Clears all user memory in the S7-200 CPU	5 3
1 CLEAR ALL	Clears all user CPU memory	5 3 1
2 CLEAR PROG BLOCK	Clears the program block in the CPU	5 3 2
3 CLEAR DATA BLOCK	Clears the data block in the CPU	5 3 3
4 CPU MEM > MEM CRT	Copies CPU memory to CPU memory cartridge	5 4
5 READ I/O CONFIG	Reads I/O configuration	5 5
6 TIME-OF-DAY	Reads/writes CPU time-of-day clock	5 6
1 READ TIME-OF-DAY	Reads the CPU time-of-day clock	5 6 1
2 WRITE TIME-OF-DAY	Writes the CPU time-of-day clock	5 6 2

Table A-19 PG Menu Functions

Menu Text on Display	Description of the Function	Key Sequence
7 READ CPU VERSION	Reads software release and ASIC release	5 7
8 READ CPU ERROR	Reads error from S7-200 CPU	5 8
6 PG CONFIGURATION >	Modifies PG Configuration	6
1 FIND CPU ADDRESS	Find S7-200 CPU address to communicate to	6 1
2 SELECT CPU ADDR	Select S7-200 CPU address	6 2
3 SET PG ADDRESS	Set PG station address	6 3
4 SET BAUD RATE	Set baud rate (9600 or 19200)	6 4
5 SET LANGUAGE >	Set language (once language is selected, select mnemonic)	6 5
1 GERMAN >	Set the language displayed to German	6 5 1
1 SIMATIC	Set the mnemonic to SIMATIC	6 5 1 1
2 INTERNATIONAL	Set the mnemonic to international	6 5 1 2
2 ENGLISH >	Set the language displayed to English	6 5 2
1 SIMATIC	Set the mnemonic to SIMATIC	6 5 2 1
2 INTERNATIONAL	Set the mnemonic to international	6 5 2 2
3 FRENCH >	Set the language displayed to French	6 5 3
1 SIMATIC	Set the mnemonic to SIMATIC	6 5 3 1
2 INTERNATIONAL	Set the mnemonic to international	6 5 3 2
4 ITALIAN >	Set the language displayed to Italian	6 5 4
1 SIMATIC	Set the mnemonic to SIMATIC	6 5 4 1
2 INTERNATIONAL	Set the mnemonic to international	6 5 4 2
5 SPANISH >	Set the language displayed to Spanish	6 5 5
1 SIMATIC	Set the mnemonic to SIMATIC	6 5 5 1
2 INTERNATIONAL	Set the mnemonic to international	6 5 5 2
6 SET BEEP VOLUME	Set beeper loudness	6 6
7 CAPABILITY LEVEL>	Sets mode to restricted or unrestricted	6 7
1 RESTRICTED	Sets the mode to restricted	6 7 1

Table A-19 PG Menu Functions

Menu Text on Display	Description of the Function	Key Sequence
7 PASSWORDS >	Enter Passwords	7
1 ENTER PASSWORD	Enter user password (log in to CPU)	7 1
2 NEW/CHNG PASSWORD	Change or create a new password	7 2
3 DELETE PASSWORD	Delete the password (use default password)	7 3
4 PASSWORD LOGOUT	Release the password (log out from CPU)	7 4
8 DEBUG R/W VARS >	Single Scan and Read/Write Variables	8
1 STATUS CHART >	Read/write variables	8 1
1 SETUP CHART	Create variable list	8 1 1
2 WRITE CHART VARS	Write variable list	8 1 2
3 READ CHART VARS	Read variable list	8 1 3
2 FORCE FUNCTIONS >	Force variables and I/O	8 2
1 FORCE VARIABLES	Force variables	8 2 1
2 FORCE I/O	Force I/O	8 2 2
3 UNFORCE ALL VARS	Unforce all variables and I/O	8 2 3
3 SINGLE SCAN	Execute single scan	8 3

ASCII Characters

B

The PG 702 provides a menu to enter ASCII characters. You can enter the ASCII characters shown in Table B-1.

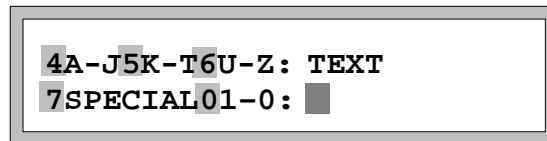
To enter an ASCII character, select the individual characters by entering the number of the row (to the left of the character) followed by the number of the column (above the character). The unassigned fields in the ASCII table generate a blank character.

Table B-1 ASCII Characters

	1	2	3	4	5	6	7	8	9	0
1	a	b	c	d	e	f	g	h	i	j
2	k	l	m	n	o	p	q	r	s	t
3	u	v	w	x	y	z				
4	A	B	C	D	E	F	G	H	I	J
5	K	L	M	N	O	P	Q	R	S	T
6	U	V	W	X	Y	Z				#
7	+	-	*	/	%	@		&	!	?
8	.	,	;	:	_	'	`	"	^	\
9	=	\$	<	>	()	[]	{	}
0	1	2	3	4	5	6	7	8	9	0

Entering ASCII Characters

When you select a function to enter ASCII characters (like **PG-Menu** **7** **2** NEW/CHG PASSWORD), the PG 702 shows the following display:



Entry Functions in the Menu

The menu displayed above provides you with the following ways to enter characters:

- Use functions 1 to 3 to enter: lower case letters a to j, k to t, and u to z
- Use functions 4 to 6 to enter: upper case letters A to Z
- Use functions 7 to 9 to enter: special characters
- Use function 0 to enter: numbers

The column numbers 1 to 3 for lower-case letters are not displayed but they correspond to the groupings assigned to numbers 4 to 6 (1=4, 2=5, 3=6). After you select the column number (1 to 0), you see a display like the ones shown in Figure B-1.

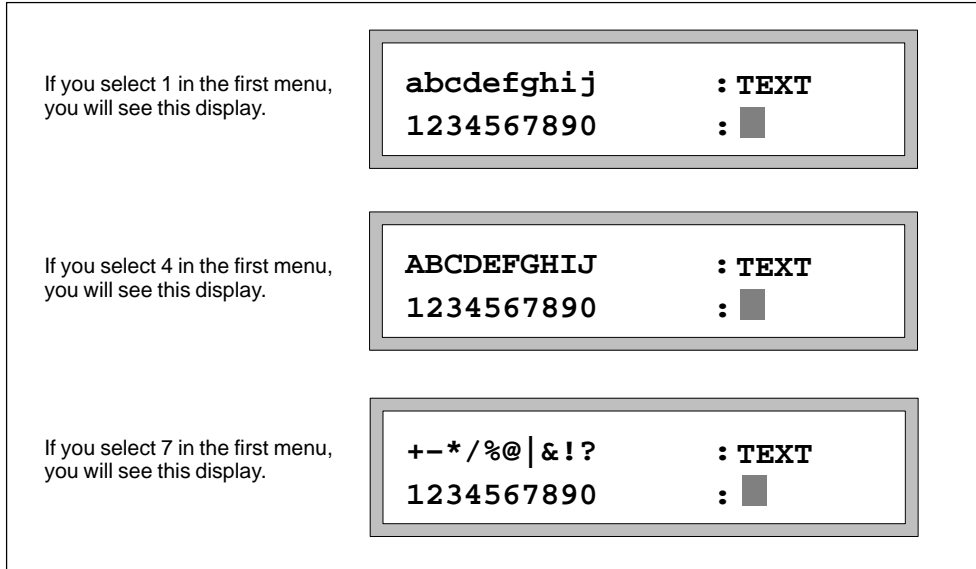


Figure B-1 Sample ASCII Character Displays

If you make the wrong selection in the first menu, you can use the arrow keys to scroll through the three selection menus as shown in Figure B-2.

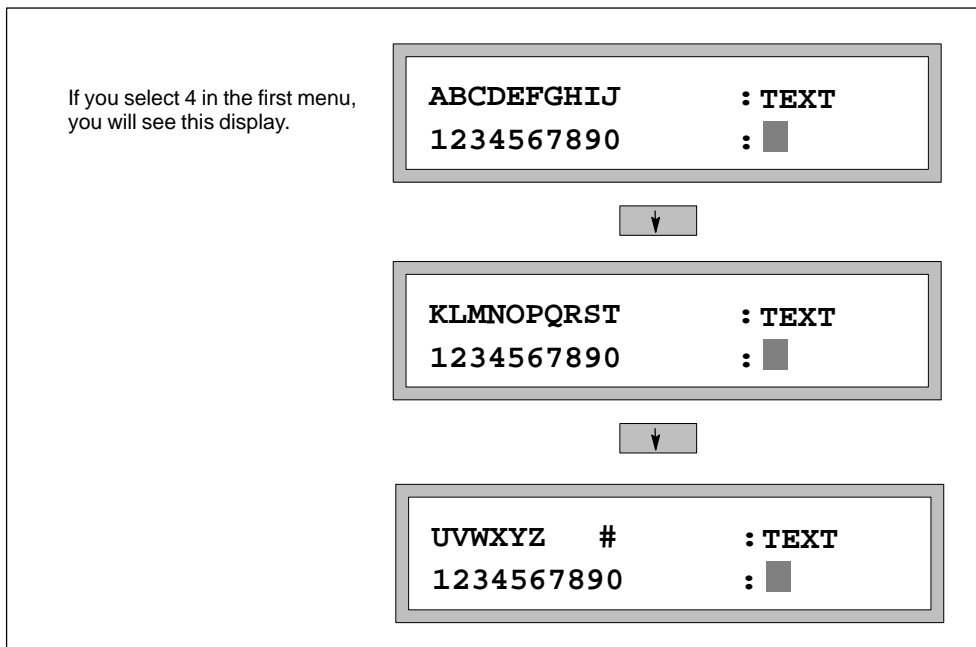
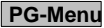









Figure B-2 Scrolling through the Second ASCII Character Menu

Example of an ASCII Entry

The following example shows you how you can use the ASCII character menu to display an ASCII character from the table in the PG 702.

Task: Display character ampersand (@).

	Explanation	Key
1.	How to activate the ASCII function a. Activate "PG functions" menu. b. Select menu with password functions. c. Activate a password function. d. Select an ASCII function.	   ..  
2.	Set "SPECIAL" by pressing key Horizontal row 7 is selected from the ASCII character table and displayed in the following menu. The position of the character is marked by digits in the second row of the menu.	
3.	Now select the @ character by pressing key The sixth character in row 7 is selected. This character is displayed at the cursor position.	
4.	Terminate your entry. The new character is included in the instruction line.	

Error Codes

C

This appendix contains the error codes that can appear on the PG 702 display.


On the PG 702 display, the second line is used as a message line to display errors or other important information. Press the  key to acknowledge the error message and clear the display.

Table C-1 Protocol Errors (80xx)

80	xx	Protocol Error
80	01	Not permitted in this operating mode
81	04	Context is not supported; error in PDU structure; service unknown
84	04	Client or Server has detected a serious error
85	00	Wrong PDU size

Table C-2 Test and Commissioning Errors (D0xx)

D0	xx	Test and Commissioning Functions
D0	A1	Function not permitted in current protection level

Table C-3 Object Management System Errors (D2xx)

D2	xx	Object Management System Functions
D2	01	Syntactical error in the block name
D2	02	Syntactical error at function parameters
D2	03	Error at block type
D2	07	Block already present on EEPROM
D2	09	Block is not present
D2	0D	SDB cannot be evaluated
D2	0F	CPU-specific maximum block size exceeded
D2	30	Wrong resource indication (block longer than in block header/resource details)
D2	40	Coordination rule violated
D2	41	Protection level of function not sufficient

Table C-4 Password Protection Errors (D6xx)

D6	xx	Protection Functions
D6	01	Syntax error at function parameters
D6	02	Wrong password entered
D6	03	Connection already legitimized
D6	04	Connection already enabled
D6	05	No legitimization possible, because no password present

Table C-5 Clocktime Errors (DCxx)

DC	xx	Clocktime Functions
DC	01	Error in date and time-of-day indication

Table C-6 Error of PPI-DRV (FFxx)

FF	xx	Error of PPI-DRV
FF	00	Timeout on Layer 7
FF	01	Error at connection establishment
FF	02	Length of block name !=9
FF	03	Invalid block type
FF	04	Invalid block number
FF	05	Invalid block status
FF	06	Block name does not correspond to conventions
FF	07	No memory
FF	08	Precondition for Read Status does not exist
FF	09	No further block info available
FF	0A	Error at reading/setting CPU time
FF	0B	No result data arrived
FF	F0	CPU-Get-Data without result

Technical Specifications and Hardware

D

This appendix contains the technical specifications and hardware information for the PG 702. It also gives guidelines for handling Electrostatic Sensitive Devices (ESD).

D.1 PG 702 Specifications

The specifications for the PG 702 are shown in Table D-1.

Table D-1 Technical Specifications for the PG 702

PG 702 Performance Data	
Order number	6ES7-702-0AA01-0YA0
Dimensions (H x W x D)	72 mm x 144 mm x 27 mm (2.8 in. x 5.6 in. x 1.1 in.)
Weight	approx. 0.25kg (0.55 lb.)
Display	STN display/ 2 lines of 20 characters each; 5 mm line height / LED backlighting / normal temperature range
Keyboard	Silicone elastomer keyboard / 33 keys
Programmer interface	PPI interface RS485 interface
Supply voltage (V_N)	24VDC ⁽¹⁾ ; (12VDC to 32VDC, safety extra-low voltage) The PG 702 has no integral protection against high-energy spurious pulses in the μ s range (surge). If no relevant protective measures are provided for the power supply, a surge arrester should be connected in series.
Current consumption (I_N)	Typically 70 mA, max. 100 mA at $V_N = 24$ V (no fuse in the PG 702)
Degree of protection	IP 30
Safety	
VDE regulation	VDE 0805 \equiv EN 60950 \equiv IEC 950
Noise emission	<45dB(A) to DIN 45635 (no fan)
Emitted interference Limit class	B to VDE 0875, T11 \equiv \times EN55011
Electromagnetic Compatibility (EMC)	
Conducted interference on DC supply lines	± 2 kV (to IEC 801-4; burst) ± 1 kV ⁽²⁾ (to IEC 801-5; μ s pulse/line to line) ± 2 kV ⁽²⁾ (to IEC 801-5; μ s pulse/line to ground)
Noise immunity on signal lines	± 1 kV (to IEC 801-4;burst)
Immunity against discharge of static electricity	± 6 kV Contact discharge (to IEC 801-2; ESD) ± 8 kV Air discharge (to IEC 801-2; ESD)
Immunity against high-frequency noise interference	10V/m with 80% amplitude modulation for 1 kHz, 10 kHz to 80 Mhz (to IEC 801-6)

Table D-1 Technical Specifications for the PG 702

PG 702 Performance Data	
Climatic Conditions	
Temperature	Tested to DIN IEC 68-2-1, DIN IEC 68-2-2
Operating	± 0° C to +45° C (32° F to +113° F)
Storage/transport	– 20° C to +60° C (- 4° F to +140° F)
Relative humidity	Tested to DIN IEC 68-2-3
Operating	5% to 95% at 25° C (77° F); no condensation
Storage/transport	5% to 95% at 25° C (77° F); no condensation
Mechanical Environmental Conditions	
Vibration	Tested to DIN IEC 68-2-6
Operating	10 Hz to 60 Hz, amplitude of 0.075mm 60 Hz to 500 Hz; acceleration of 9.8m/s ²
Shock	Tested to DIN IEC 68-2-27
	Half-sine: 100m/s ² (10g), 1ms
Special Features	
Quality assurance	To ISO 9001
Maintenance	Maintenance-free (no battery)
Installation in panels	No accessories required for installation in panels

- (1) The 24 VDC supply voltage should be limited to a rated current of ≤ 3A by means of either an overcurrent limiter or a fuse.
- (2) Only in connection with additional measures, such as overvoltage diverters, e.g. of Messrs. Dehn, type RZ/E 24V-, Order No. 917 024.

D.2 Programmer Cable (PPI Interface for the PG 702)

Figure D-1 shows the PPI interface.

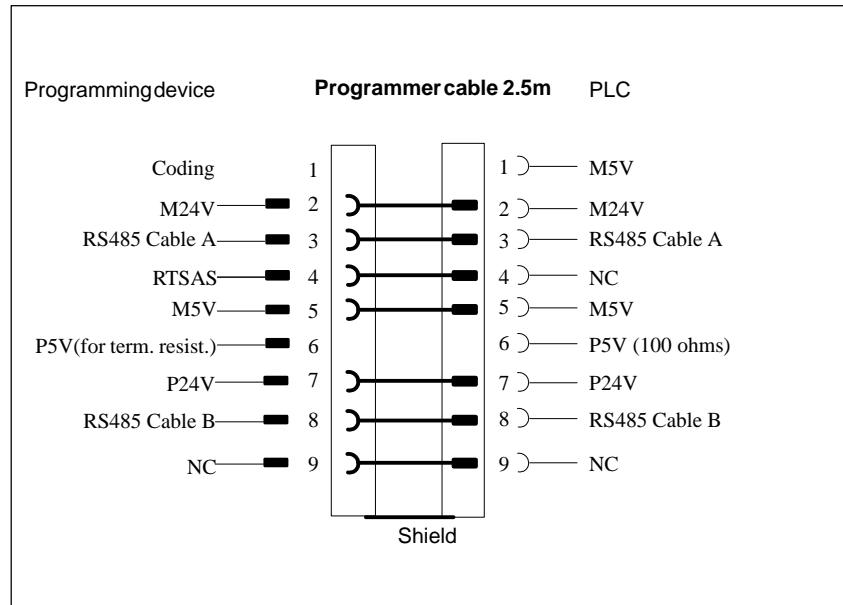


Figure D-1 PPI Interface

D.3 Handling Electrostatic Sensitive Devices (ESD)

What is an ESD?

Almost all SIMATIC/TELEPERM modules are equipped with large-scale integrated ICs or components with MOS technology. Due to their design, these electronic elements are very sensitive to overvoltages and thus to any electrostatic discharge. These **E**lectrostatic **S**ensitive **D**evelopments are commonly referred to by the abbreviation **ESD**.

The symbol below appears on cabinets, racks or packings to point out the use of electrostatic sensitive devices and that the respective modules are sensitive to contact with any person.



ESDs are subject to voltages and energies far below the range that can still be perceived by human beings. These voltages are present even if you touch a component or module without previously being electrostatically discharged. In most cases, the damage caused to components by such overvoltages is not immediately noticeable and results in failure only after a prolonged period of time.

An electrostatic discharge is

- perceptible from 3500 Volt
- audible from 4500 Volt
- visible from 5000 Volt

A fraction of these voltages is capable of destroying or damaging electronic devices.

Handling PG 702 When the Case is Open for Installation

- If possible, avoid any contact with electrostatic sensitive devices. Hold modules without touching the pins of components or printed conductors.
- Touch electrostatic sensitive devices only in the following cases:
 - If you wear a wristband complying with ESD specifications
 - If you use special ESD footwear or ground straps attached to your shoes when walking on an ESD floor.
- Before working on electronic devices, first discharge your body by simply touching a conductive, grounded part (e.g. bare metal parts of cabinets, water pipes, etc.).
- Protect the modules against contact with chargeable and highly insulating materials, such as plastics, insulating table tops or clothes made of plastic fibers.
- Place the modules only on conductive surfaces (tables with ESD surface, conductive ESD foam plastic, ESD bags, ESD containers).
- Avoid direct contact of electrostatic sensitive devices with visual display units, monitors or TV sets.

D.4 Approbation

Radio Interference Suppression

Manufacturer's/Importer's Certification
<p>This is to certify that the PG 702 programming device, 6ES7702-0AA01-0YA0, EAN 41 25515 16 1189, is interference-suppressed (to limit class B) in compliance with regulations 1046/84 and 483/86 laid down by the German Federal Ministry of Post and Telecommunications, as well as the German DIN VDE 0875T.11/7.92 standard (5DIN EN 55011 5 CISPR 11).</p> <p>Radio and television reception can be impaired if components and devices not conforming with these regulations are used. The Federal (German) Approvals Office for Telecommunications (BZT) has been notified of the introduction of this device and permission was given to verify compliance of the series with the regulations.</p> <p>SIEMENS AG</p>

Supplementary Note on Interference Suppression

Any other devices connected to the product in question must also be interference-suppressed in conformity with regulation no. 1046/84 or 243/91 of the German Federal Ministry of Post and Telecommunications or guideline 89/336/EEC issued by the European Community. Products complying with these regulations are provided with the manufacturer's certificate or a certificate issued by the Federal (German) Approvals Office for Telecommunications (BZT), or a CE mark. Products that do not fulfill the relevant stipulations require an individual approval by the Federal (German) Approvals Office for Telecommunications (BZT).

PG 702 Troubleshooting Guide

E

Table E-1 contains common problems with the PG 702, a list of possible causes, and the recommended solutions.

Table E-1 Troubleshooting

Problem	Possible Causes	Solutions
Cannot communicate with CPU.	Wrong station address	Select the correct CPU station address using the PG configuration menu.
	PG baud rate does not match CPU baud rate	Change PG 702 baud rate.
	CPU not powered up	Apply power to the S7-200 CPU.
	Bad communication cable	Replace communication cable.
	PG 702 powers up before the CPU comes on line	Power cycle the PG 702.
	Received an unrecoverable communication error	Power cycle the PG 702.
CPU in wrong mode.	Mode switch on CPU in the wrong position	Put mode switch on CPU in TERM position.

Index

A

- Aborting, entry, 4-11
- Aborting an operator entry, CLR key, 4-7
- Access capabilities, password, 4-9
- Adding a new force variable, 6-6
- Address
 - direct entry, 5-10
 - entering identifiers, 5-8
 - keys, 4-6, 5-8
 - multiple, 5-11
- Address value
 - entering, 5-8
 - keys, 4-6, 4-7
- AND instructions, A-2
- Approbation, D-7
- ASCII
 - characters, B-1
 - sample character display, B-3
- ASCII characters, entering, B-2
- Assistance, requesting technical, iv

C

- Capability level, 4-44
- Clear CPU Memory, 4-36
- Clearing an existing program, 3-5
- Clocktime errors, C-2
- CLR key, 4-3
 - aborting an entry, 4-7
 - no, 4-7
- Colors of key, 4-7
- Commissioning and test errors, C-1
- Compare Program in Flash to Program in CPU Memory, 4-33
- Configuration functions
 - Output Table, 4-16
 - Port 1 parameters, 4-16
 - Set input filters, 4-16
- Configuration functions for CPU, 4-16
- Connecting the PG 702, 2-2, 3-2
- Constants, 5-2

- Conversion instructions, A-9
- Copy CPU memory, 4-36
- Counter/Timer instructions, A-5
- CPU
 - compatibility, 4-34
 - downloading data to, 4-27
 - downloading program to, 4-15
 - upload program from, 4-14
 - uploading program to PG 702, 3-4
- CPU configuration, functions, 4-16
- CPU Data Block, 4-26
- CPU functions, 4-35
 - Copy CPU Memory, 4-36
 - Put CPU in RUN/STOP, 4-36
 - Read CPU error, 4-40
 - Read CPU Version, 4-40
 - Read I/O Configuration, 4-37
 - Read/Write Time-of-Day Clock, 4-38
- CPU program block, 4-14
 - editing, 1-3
- Creating a new program, 3-4
- Creating a program, 5-3
- Creating a program, 1-3
- Cursor
 - position, 4-7
 - positioning, 5-14

D

- Dark grey key, 4-7
- Data Block
 - Edit mode, 4-12
 - editing, 4-28
 - entering statements, 4-29
 - view of value, 4-27
- Data Block Editor
 - cursor movement, 4-29
 - key definitions, 4-28
- Data Blocks, editing, 1-3
- Data Editor, mode, 4-8

- FLOAT, 4-28
- floating point, 4-28
- HEX, 4-28
- hexadecimal, 4-28
- INT, 4-28
- UINT, 4-28
- integer
 - signed, 4-28
 - unsigned, 4-28
- Data Move instructions, A-10
- DE mode, 4-8
- Debug
 - function menu, 6-2
 - functions, 6-2
 - single scan function, 6-8
- Debug mode, 4-8, 4-12
- Debug Read/Write variables, 4-50
- Delays for input points, 4-21
- Delete Password, 4-49
- Deleting
 - instruction line, 5-14
 - parts of an instruction, 5-14
- DG mode, 4-8
- Display, 4-8
 - Edit mode, 5-12
 - information line, 5-13
- Downloading, program to CPU, 5-3
- Downloading to CPU, 4-15, 4-27
- DP Port Address, 4-24
- DP port parms, 4-16
- DP status, 4-24
- DP Status parameter screens, 4-25

E

- ED mode, 4-8
- Edit
 - a program, 5-3
 - functions, 5-2
- Edit mode, 4-8
- Editing
 - CPU program blocks, 1-3
 - data blocks, 1-3
 - system data blocks, 1-3

- Editing aids, 5-15
 - GO TO, 5-16
 - key, 4-7
- Editing Data Block, 4-28
- Editing Programs, 5-2
- Electrostatic Sensitive Devices, handling, D-5
- Enter Password, 4-48
- Entering instructions
 - keys, 4-4
 - sample task, 3-7
 - ways to enter, 4-4
- Entering restricted mode, 4-45
- Entry
 - aborting, 4-11
 - canceling, 5-14
- Error codes
 - clocktime, C-2
 - object management system, C-1
 - password protection, C-2
 - PPI-DRV, C-2
 - protocol, C-1
 - test and commissioning, C-1
- Error messages, display, 5-13
- Example
 - clearing an existing program, 3-5
 - entering instructions, 4-6
 - of ASCII entry, B-4
 - powering up/setting the PG 702, 3-2
- Example of menu sequence, 4-10
- Exiting restricted mode, 4-46

F

- Find CPU address, 4-42
- FIND function, activating, 5-17
- Flash operations, 4-30
- FLOAT, data format, 4-28
- Floating point numbers, 6-4
- Force values, valid keys for modifying, 6-5
- Force variable, adding a new, 6-6
- Forcing, variables, 6-5
- Function, selecting from menu, 4-10
- Functionality of the PG 702, 1-2

Functions

- Clear CPU to Memory, 4-36
- CPU data block, 4-26
- Debug Read/Write Variables, 4-50
- DP Port Params, 4-16
- Flash Operations, 4-30
- Passwords, 4-47
- PG configuration, 4-41
- Port 0, 4-22
- Port 0 parameters, 4-16
- Port 1, 4-23
- Retentive ranges, 4-16

G

- GoTo Network or line number, 5-16
- Green keys, 4-7

H

- Handling Electrostatic Sensistive Devices, D-5

Instructions, 5-4

- AND, A-2
- Conversion, A-9
- Counter/Timer, A-5
- Data Move, A-10
- entering, 4-4
- entering with a selection menu, 5-7
- entering without menu, 5-5
- insert, 5-4
- Interrupt, A-7
- keys, 5-4
- Label, A-6
- LOAD, A-4
- mathematical, A-11
- miscellaneous, A-12
- miscellaneous control, A-7
- OR, A-3
- Return, A-6
- Rotate and Shift, A-8
- Shift and Rotate, A-8
- Stack, A-10
- Table, A-8
- Timer/Counter, A-5
- via menu, 5-6
- Word logic, A-9

- INT, data format, 4-28

- Interrupt instructions, A-7

- HEX, data format, 4-28

I

- I/O groups sample, 4-20

- Information line, 5-13

- edit mode, 5-13

- operating mode, 4-8

- Input delay time values, 4-21

- Input filters, setting, 4-20

- Instruction

- entry, A-6

- line, 5-4

- Instruction line, 4-8

- deleting, 5-14

- single line, 5-12

- Instruction Set

- menus, A-6

- overview of, A-1

K

- Key definitions of Data Block Editor, 4-28

- Key functions, 4-3, A-14

- Key labeling, 4-3

- Keypad

- colors of, 4-7

- key functions, 4-7

- PG 702, 4-2

- Keys, 3-3, 5-2

- address, 5-8

- address values, 4-6

- addresses, 4-6

- CLR key, 4-3

- entering instructions, 4-4

- for entering operand types, 4-6

- instructions, 5-4

- meaning of colors, 4-7

- shift key, 4-3

- switching functions, 4-6

- to modify Time-of-Day Clock, 4-38

- valid for modifying force information, 6-5

L

- Label instructions, A-6
- Language of PG 702, setting the, 1-2
- Light gray key, 4-7
- Line number, 4-8
- Load instructions, A-4
- Logic operations, A-2

M

- Mathematical instructions, A-11
- Menu
 - Debug function, 6-2
 - example of sequence, 4-10
 - exiting, 4-11
 - instruction set, A-6
 - overview, A-1
 - PG functions, A-16
 - selecting, 4-10
 - working with, 4-10
- Menu sequence, 4-10
- Menus with the editing aids, activating, 5-15
- Messages
 - confirming, 4-6
 - system, 3-2
- Miscellaneous
 - control instructions, A-7
 - instructions, A-12
- Mode identifier, 4-8
- Modes, operating, 4-12
- Mounting the PG 702, 2-4

N

- Network number, 4-8
- New/Change Password, 4-48
- No key, 4-7

O

- Object management system errors, C-1
- Operand, 4-5, 5-2, A-13
 - entering individual, 5-10
 - entering multiple, 5-11
 - entry styles, 5-8
- Operand offsets, 5-9
- Operating modes, 4-12
 - identifier, 4-8
- Or instructions, A-3

- Output table, 4-16
 - setting, 4-19
- Overview
 - instruction sets, A-1
 - of menu, A-1

P

- Panel/cabinet mounting, 2-4
- Password
 - access capabilities, 4-9
 - logout, 4-49
 - protection, 4-9
 - protection errors, C-2
- Password prompt, 4-9
- Passwords
 - delete password, 4-49
 - enter function, 4-48
 - function, 4-47
 - New/Change password, 4-48
- PG 702
 - additional assistance, iv
 - application areas, 1-2
 - connecting to the CPU, 2-2, 3-2
 - example of how to power up and set the PG 702, 3-2
 - external power supply, 2-2
 - first power up, 3-2
 - keypad, 4-2
 - mounting the, 2-4
 - operating modes, 4-12
 - order number, iii
 - related manual information, iv
 - setting the programming language, 3-3
 - uploading CPU program to, 3-4
- PG Configuration functions, 4-41
 - capability level, 4-44
 - entering restricted mode, 4-45
 - Find CPU Address, 4-42
 - Select CPU Address, 4-42
 - Set Baud Rate, 4-42
 - Set Beep Volume, 4-44
 - Set Language, 4-43
 - Set PG Address, 4-42
- CPU Memory to PG Flash, 4-30
- PG Flash to CPU Memory, 4-31
- PG functions key, 4-7
- PG menu functions, A-16
 - under restricted mode, 4-45

- Port 0
 - functions, 4-22
 - parameters, 4-16
 - setting parameters, 4-22
 - Port 1
 - functions, 4-23
 - parameters, 4-16
 - setting parameters, 4-23
 - Position, cursor, 4-7
 - Positioning, cursor, 5-14
 - Power supply, external, 2-2
 - Power up, confirming messages, 3-2
 - Powering up the PG 702, 3-2
 - PPI interface, D-4
 - PPI-DRV errors, C-2
 - Problems, troubleshooting, E-1
 - Program, sample task, 3-7
 - Program Block, Edit Mode, 5-3
 - Program Block Edit mode, 4-12
 - Program Editor, exiting and returning, 4-15
 - Programmer cable PPI Interface, D-4
 - Programming
 - creating a program, 3-4
 - keys, 3-3
 - setting the language, 3-3
 - Protocol errors, C-1
 - Put CPU in RUN/STOP, 4-36
- R**
- Radio interference suppression, D-7
 - Read CPU error, 4-40
 - Read CPU version, 4-40
 - Read I/O Configuration, 4-37
 - Read/Write Time-of-Day Clock, 4-38
 - Reading, Status Chart variables, 6-4
 - Reading DP status, 4-24
 - Red key, 4-7
 - Restricted mode
 - exiting, 4-46
 - functions available, 4-45
 - Retentive ranges, 4-16
 - setting, 4-17
 - Return instructions, A-6
 - Rotate and Shift instructions, A-8
- S**
- Sample program, increments memory word, 3-7
 - Sample task, downloading program to CPU, 3-12
 - Select CPU address, 4-42
 - Selecting, menu, 4-10
 - Set baud rate, 4-42
 - Set beep volume, 4-44
 - Set input filters, 4-16
 - Set language, 4-43
 - Set PG address, 4-42
 - Setting
 - DP port address, 4-24
 - input filters, 4-20
 - output table, 4-19
 - port 0 parameters, 4-22
 - port 1 parameters, 4-23
 - Setting retentive ranges, 4-17
 - Shift and Rotate Instructions, A-8
 - Shift key, 4-3
 - Single scan debug function, 6-8
 - Specifications, D-2
 - Stack instructions, A-10
 - Starting up PG 702, 4-9
 - Statement list program, delimiters, 4-4
 - Status Chart
 - entering floating point numbers, 6-4
 - reading variables, 6-4
 - setting up, 6-3
 - writing variables, 6-4
 - Switching key functions, 4-6
 - System data block, editing, 1-3
- T**
- Table instructions, A-8
 - Technical Assistance, requesting, iv
 - Technical specifications, D-2
 - Terminating an entry, 5-14
 - Test and commissioning errors, C-1
 - Time-of-Day Clock, valid keys to modify, 4-38
 - Timer/Counter instructions, A-5
 - Troubleshooting, E-1

U

UINT, data format, 4-28

Unforcing

 a single variable, 6-7

 all variables, 6-7

 variables, 6-5

Upload a program from CPU, 4-14

Uploading, an existing CPU program to PG,
 3-4

Uploading CPU program to PG 702, 3-4

Uploading from CPU, 4-27

Using the PG 702, 1-2

W

Warnings, display, 5-13

White key, 4-7

Word Logic instructions, A-9

Write Time-of-Day Clock screens, 4-39

Writing, Status Chart variables, 6-4

Y

Yellow key, 4-7

Yes key, 4-7

Siemens AG
A&D AS E46

Oestliche Rheinbrueckenstr. 50
D-76181 Karlsruhe
Federal Republic of Germany

From:

YourName: _ _ _ _ _

YourTitle: _ _ _ _ _

Company Name: _ _ _ _ _

Street: _ _ _ _ _

City, Zip Code _ _ _ _ _

Country: _ _ _ _ _

Phone: _ _ _ _ _

Please check any industry that applies to you:

- | | |
|--|--|
| <input type="checkbox"/> Automotive | <input type="checkbox"/> Pharmaceutical |
| <input type="checkbox"/> Chemical | <input type="checkbox"/> Plastic |
| <input type="checkbox"/> Electrical Machinery | <input type="checkbox"/> Pulp and Paper |
| <input type="checkbox"/> Food | <input type="checkbox"/> Textiles |
| <input type="checkbox"/> Instrument and Control | <input type="checkbox"/> Transportation |
| <input type="checkbox"/> Nonelectrical Machinery | <input type="checkbox"/> Other _ _ _ _ _ |
| <input type="checkbox"/> Petrochemical | |



Remarks Form

Your comments and recommendations will help us to improve the quality and usefulness of our publications. Please take the first available opportunity to fill out this questionnaire and return it to Siemens.

Please give each of the following questions your own personal mark within the range from 1 (very good) to 5 (poor).

- 1. Do the contents meet your requirements?
- 2. Is the information you need easy to find?
- 3. Is the text easy to understand?
- 4. Does the level of technical detail meet your requirements?
- 5. Please rate the quality of the graphics/tables:

Additional comments:
