

SIMATIC S5

**PG 605U
Programmer**



Manual

EWA 4NEB 810 1056-02d



Siemens has developed this document for its licensees and customers. The information contained herein is the property of Siemens and may not be copied, used, or disclosed to others without prior written approval from Siemens. Users are cautioned that the material contained herein is subject to change by Siemens at any time and without prior notice.

Siemens shall not be responsible for any damages, including consequential damages, caused by reliance on material presented, including but not limited to typographical, electronic, arithmetic, or listing errors.

| | |
|---|--|
|  |  WARNING |
| | <p>Hazardous voltage.</p> <p>Can cause death, severe personal injury, or substantial property damage.</p> <p>Restrict use to qualified personnel. See safety instructions.</p> |

Only qualified personnel should install or maintain this equipment after becoming thoroughly familiar with all warnings, safety notices, and maintenance procedures contained in this manual. The successful and safe operation of this equipment is dependent upon proper handling, installation, operation, and maintenance.

The following are definitions of the terms “qualified person”, “danger”, “warning”, and “caution”, as applicable for this document.

Qualified Person

One who is familiar with the installation, construction, and operation of this equipment and the hazards involved. In addition, the person should have the following qualifications:

- Be trained and authorized to use and tag circuits and equipment in accordance with established safety practices
- Be trained in the proper care and use of protective equipment in accordance with established safety practices
- Be trained in rendering first aid

DANGER

Indicates loss of life, severe personal injury, or substantial property damage will result if proper precautions are not taken.

WARNING

Indicates loss of life, severe personal injury, or substantial property damage can result if proper precautions are not taken.

CAUTION

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

STEP 5® and SIMATIC® are registered trademarks of Siemens AG.



Preface

How To Use This Book

System Overview

1

Technical Specifications

2

Installation Guidelines

3

Programming and Operator-Process Communication

4

Appendices

**A/B/
C/D**

Alphabetical Keyword Index

Preface

This book provides an overview of the PG 605U programmer. It describes the hardware and installation procedures. It also provides information on programming, applications, error/fault messages, and technical specifications. Also included is a STEP 5 operation set reference section.

This book is intended for engineers, programmers, and maintenance personnel who have a general knowledge of programmable controller concepts.

If you have any questions about the PG 605U programmer not answered in this book, please contact your local Siemens representative. In the United States of America, please call 1-800-322-7224.

How to Use this Book

This section discusses information that may be helpful as you use this book.

Contents of this Book

- Chapter 1 - System Overview

This chapter provides an overview of the PG 605U programmer. It also lists the controllers that can be programmed using the PG 605U.

- Chapter 2 - Technical Specifications

This chapter describes the technical specifications for the PG 605U, the principle of operation, and the keypad.

- Chapter 3 - Installation Guidelines

This chapter discusses the various options for connecting the PG 605U to SIMATIC S5 programmable controllers.

- Chapter 4 - Programming and Operator-Process Communication

This chapter discusses programming on the PG 605U programmer, and describes the various options available to the user.

- Appendix A - STEP 5 Operations

This appendix lists all the operations possible in STEP 5 programming.

- Appendix B - Error Codes

This appendix lists all of the error codes and their meanings.

- Appendix C - Siemens Addresses Worldwide

This appendix lists Siemens companies and representatives worldwide.

- Index

The index contains an alphabetical list of key terms and subjects covered in this book and their corresponding page numbers.

- Remarks Form

The Remarks Form is provided for your comments and recommendations. If you are in the United States of America, please use the postage-paid form.

Reference Material

It is recommended that you have the following books that support the PG 605U programmer:

Catalog ST 59

Training

Contact your local Siemens representative for information on training courses to aid you in becoming familiar with this product. Consult Appendix C for a list of Siemens offices worldwide:

Conventions

The following conventions are used in this book and are listed for your reference.

| Convention | Definition | Example |
|-----------------|---|---|
| <XXXXX XXXXXXX> | Uppercase characters in angle brackets indicate a key on the keyboard of the PG 605U. | Press <ENTER> |
| “XXXXX XXXXXXX” | Uppercase characters in quotation marks indicate a character or number to be entered exactly as shown. A box that indicates a type of hazard, describes its implications, and tells you how to avoid the hazard is a cautionary statement. Some cautionary statements include a graphic symbol representing an electrical or radio-frequency hazard. All cautionary statements have one of the following levels of caution: <ul style="list-style-type: none"> • A danger indicates that loss of life, severe personal injury, or substantial property damage will result if proper precautions are not taken. • A warning indicates that loss of life, severe personal injury, or substantial property damage can result if proper precautions are not taken. • A caution indicates that minor personal injury or property damage can result if proper precautions are not taken | Enter “A” or Enter “4”, and an A or a 4 appears in the display. |

1 System Overview

2 Technical Specifications

3 Installation Guidelines

4 Programming and Operator-Process Communication

| | |
|---|--------|
| Figures | |
| 1-1. The PG 605U Programmer | 1. - 2 |
| Tables | |
| 1-1. Programmer Interface Options | 1. - 1 |

1 System Overview

You can use the PG 605U to develop, correct and debug STEP 5 user programs. You can also print the programs with the help of a printer adapter.

The table below shows you how the two versions of the PG 605U programmer can be connected to the associated programmable controllers.

Table 1-1. Programmer Interface Options

| AGs \ PGs | PG 605U Order number 6ES5 605-0UA11 | | PG 605U Order number 6ES5 605-0UA12 |
|-----------|--|--|---|
| | without line adapter 984 | with line adapter 984 (6ES5 984-1UA31, 6ES5 984-1UA41) | |
| S5-90U | | • | • |
| S5-95U | | • | • |
| S5-100U | • | • | • |
| S5-101U | • | • | • |
| S5-115U | • | • | • |



Figure 1-1. The PG 605U Programmer

1 System Overview

| | | |
|----------|---|-------|
| 2 | Technical Specifications | |
| 2.1 | Technical Specifications of the PG 605U | 2 - 1 |
| 2.2 | Principle of Operation of the PG 605U | 2 - 1 |
| 2.3 | The PG 605U Keypad | 2 - 2 |

3 Installation Guidelines

4 Programming and Operator-Process Communication

| | |
|--|-------|
| Figures | |
| 2-1. PG 605U Keypad | 2 - 2 |
| Tables | |
| 2-1. Overview of the PG 605U Function Keys | 2 - 3 |
| 2-2. Overview of the PG 605U Cursor Control Keys | 2 - 4 |

2 Technical Specifications

2.1 Technical Specifications of the PG 605U

| | |
|--|--|
| Power supply: | 6ES5 984-0UA11 5V/0.5A 6ES5 984-0UA12 5V/0.1A via the PLC connecting cable |
| Interface: | Serial TTY current loop |
| Ambient temperature: | +5 to +40 °C |
| Transport and storage temperature: | -20 to +60 °C |
| Degree of protection: | IP50 (No protection against water) |
| Dimensions (WxHxD) (in mm): | 120 x 45 x 200 mm/4.7 x 1.8 x 7.9 in. |
| Weight: | approx. 0.9 kg/2lbs. |
| Adapter with printer connector: | 6ES5 984-1UA11 6ES5 984-1UB11 (only for S5-101U) |
| 984 cable adapter (for cables > 3 m/10 ft. and <1000 m/3280 ft. for S5-110S) | 6ES5 984-1UA31 (for 220/240 V) 6ES5 984-1UA41 (for 110/120 V) |

2.2 Principle of Operation of the PG 605U

The PG 605U has a buffer, which can accommodate one program block (up to 1023 statements). Program entries and corrections are stored here first. The block only becomes effective when it is transferred to the programmable controller.

Only one block can be processed at a time in the PG 605U. If a block already in the PLC memory is to be displayed, corrected or debugged, it is automatically copied into the programmer memory when it is called up, and so remains available in the PLC memory. The block previously in the programmer is overwritten.

Note:

When the programmer is unplugged, its memory contents are lost.

2.3 The PG 605U Keypad

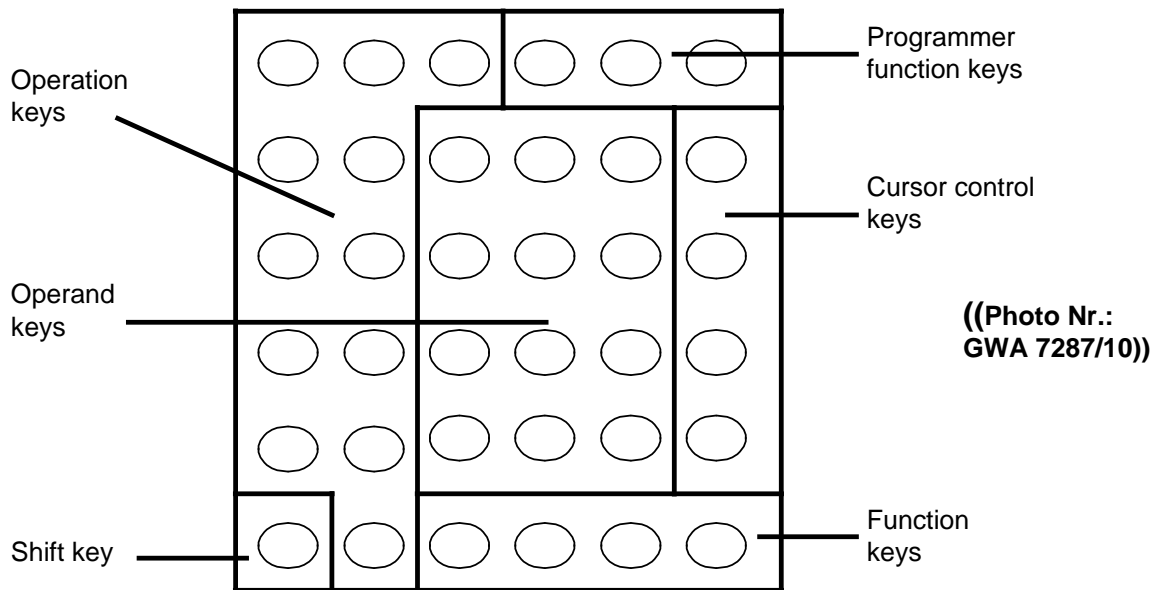






Figure 2-1. PG 605U Keypad

The PG 605U has a number of dual-function keys. The secondary functions (marked in white) are activated by first pressing the white <SHIFT> key. If you require several secondary functions in succession, you must press the <SHIFT> again for each function. Press the <SHIFT> key once again to reset. A broken line at the bottom left of the display indicates that a secondary function has been selected.

Table 2-1. Overview of the PG 605U Function Keys

| Key | Function |
|-----|--|
| | <SHIFT> : The <SHIFT> key activates the secondary functions (printed in white) |
| | <DISPLAY> key <SEARCH> key (Input/output; program-dependent signal status display) |
| | <INPUT> key |
| | <INSERT> key (Input/output) |
| | <DELETE> key <REMOVE> key (In initial state: Overall Reset of PLC) (Input/output) |
| | <ENTER> key (Terminate input/output) |
| | <STATUS> (Signal status display, manipulate timers and counters) |
| | <SPECIAL> key (Special functions) |
| | <BREAK> key |
| | <CLEAR> (Clear display, acknowledge errors) |

Table 2-2. Overview of the PG 605U Cursor Control Keys

| Key | Function | |
|---|---------------|--|
|  | <ARROW LEFT> | Enter labels (input/display) Move cursor to the left (in manipulating timers and counters) |
|  | <ARROW RIGHT> | Terminate labels (input) Move cursor to the right (in manipulating timers and counters) |
|  | <ARROW UP> | Last statement, decrement address |
|  | <ARROW DOWN> | Next statement, increment address, enter |

- 1 System Overview
- 2 Technical Specifications

| | | |
|------------|---|--------------|
| 3 | Installation Guidelines | |
| 3.1 | Connecting the PG 605U | 3 - 1 |
| 3.2 | Positions in which the PG 605U can be Used | 3 - 2 |

- 4 Programming and Operator-Process Communication

Figures

| | | | |
|------|--|-------|-------|
| 3-1. | Connecting the PG 605U to an S5-101U Programmable Controller | | 3 - 1 |
| 3-2. | The PG 605U as a Handheld and Desktop Unit | | 3 - 2 |
| 3-3. | Cutout Dimensions for Mounting in a Control Panel | | 3 - 2 |

3 Installation Guidelines

3.1 Connecting the PG 605U

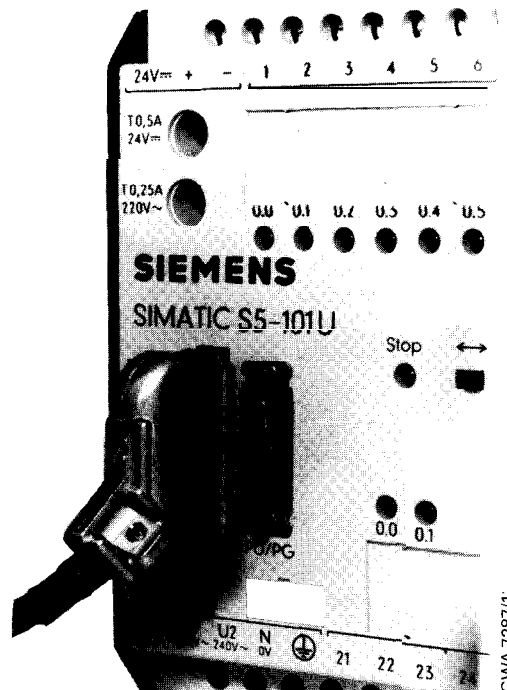


Figure 3-1. Connecting the PG 605U to an S5-101 U Programmable Controller

To connect the PG 605U, plug the programmer connector into the connecting socket of the programmable controller and lock it. This establishes all the necessary connections, including the power supply.

After connection, the programmer executes a short self-test and then enters the initial state. The following appears on the display:

```

: PG 605U V 1.4
* COMMAND *
  
```

The PG 605U programmer is switched off by unlocking and unplugging the programmer connector.

3.2 Positions in which the PG 605U can be Used

The PG 605U programmer is suitable for

- Handheld use (→ Figure 3-2.)
- Desktop use (→ Figure 3-2.)
- Mounting in a control panel (→ Figure 3-3.)

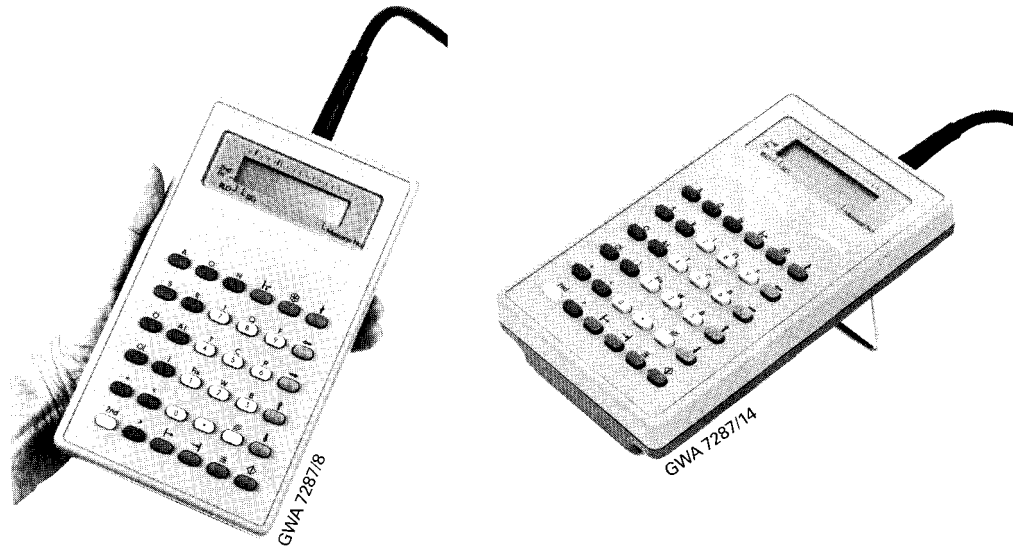


Figure 3-2. The PG 605U as a Handheld and Desktop Unit

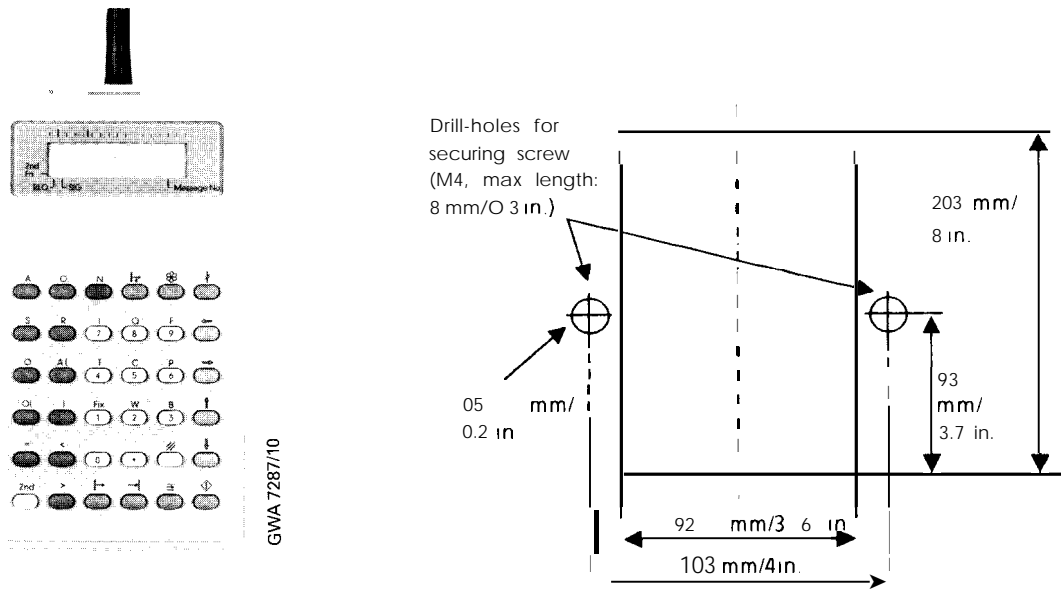


Figure 3-3. Cutout Dimensions for Mounting in a Control Panel

- 1 System Overview
- 2 Technical Specifications
- 3 Installation Guidelines

| 4 Programming and Operator-Process Communication | |
|---|--|
| 4.1 | General 4 - 1 |
| 4.2 | Input 4 - 2 |
| 4.2.1 | Selecting Input 4 - 2 |
| 4.2.2 | Entering Statements 4 - 2 |
| 4.2.3 | Entering Labels (Only in FBs) 4 - 4 |
| 4.2.4 | Programming Data Blocks 4 - 5 |
| 4.3 | Display 4 - 6 |
| 4.3.1 | Display from the PLC 4 - 6 |
| 4.3.2 | Display from the Programmer 4 - 7 |
| 4.4 | Correction 4 - 8 |
| 4.4.1 | Deleting Statements 4 - 8 |
| 4.4.2 | Inserting Statements 4 - 8 |
| 4.4.3 | Overwriting Statements 4 - 8 |
| 4.5 | Cursor Control 4 - 8 |
| 4.6 | Search Function 4 - 9 |
| 4.6.1 | Searching for Statements or Operands 4 - 9 |
| 4.6.2 | Searching for Labels (Only in FBs) 4 - 9 |
| 4.6.3 | Searching for Addresses 4 - 9 |
| 4.7 | Test Functions 4 - 10 |
| 4.7.1 | STATUS (Program-Dependent Signal Status Display) 4 - 10 |
| 4.7.2 | STATUS VAR (Direct Signal Status Display) 4 - 11 |
| 4.7.3 | FORCE VAR 4 - 11 |
| 4.7.4 | Manipulating Timers and Counters (from Version V1.2 Onward) 4 - 12 |
| 4.8 | Enquiry Functions 4 - 14 |
| 4.8.1 | Directory Functions 4 - 14 |
| 4.8.2 | Interrupt Stack (ISTACK) 4 - 14 |
| 4.8.3 | Displaying the System Parameters 4 - 19 |
| 4.9 | Special Functions 4 - 21 |
| 4.9.1 | Starting and Stopping the PLC via the Programmer 4 - 21 |
| 4.9.2 | Compressing the PLC Memory 4 - 21 |
| 4.9.3 | Display Check 4 - 22 |
| 4.9.4 | Key Check 4 - 23 |
| 4.10 | Delete Functions 4 - 25 |
| 4.10.1 | Overall Reset of the PLC 4 - 25 |
| 4.10.2 | Deleting Blocks 4 - 25 |

Figures

| | | |
|-------|---|--------|
| 4-1. | PG 605U Display Panel | 4 - 1 |
| 4-2. | Example for Entering OB 1 | 4 - 2 |
| 4-3. | Statement Entry Flowchart | 4 - 3 |
| 4-4. | Flowchart for Entering Labels | 4 - 4 |
| 4-5. | Display of the Various Data Formats | 4 - 5 |
| 4-6. | Example for Displaying OB 1 from the PLC | 4 - 6 |
| 4-7. | Example for Displaying OB 1 from the Programmer | 4 - 7 |
| 4-8. | Typical STATUS Display | 4 - 10 |
| 4-9. | Representation of the Signal State | 4 - 10 |
| 4-10. | Typical FORCE Display | 4 - 11 |
| 4-11. | Test Pattern of the PG 605U Display Check | 4 - 22 |
| 4-12. | Assignment of the Keys for the Key Check | 4 - 23 |
| 4-13. | Schematic Sequence of the Key Check | 4 - 24 |

Tables

| | | |
|------|---|--------|
| 4-1. | Interrupt Stack (S5-90U, S5-95U, S5-100U, S5-101U, S5-115U) | 4 - 15 |
| 4-2. | Explanation of the ISTACK Bits | 4 - 17 |
| 4-3. | System Parameters | 4 - 19 |
| 4-4. | CPU Identifier in SYSPAR | 4 - 20 |

4 Programming and Operator-Process Communication

4.1 General

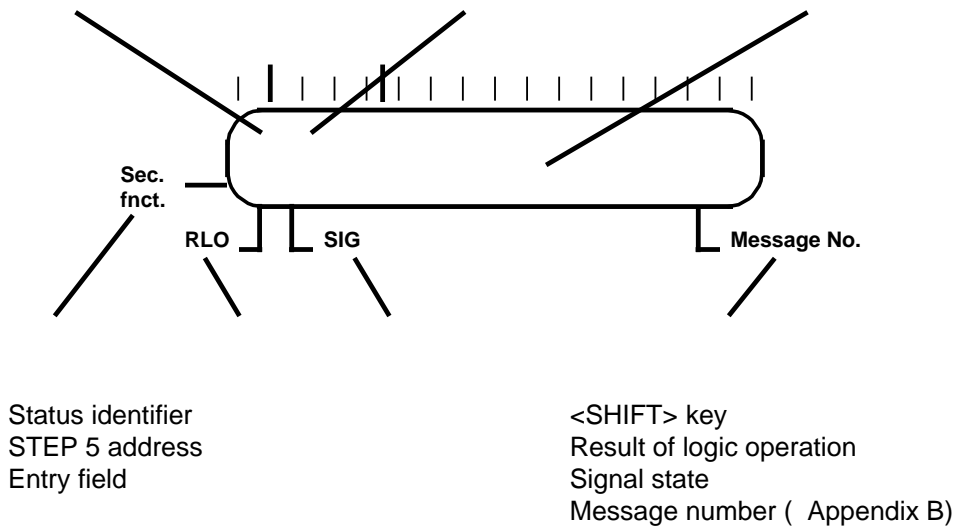


Figure 4-1. PG 605U Display Panel

After it has been connected to the programmable controller and has completed the short self-test, the programmer (PU) prompts the user to enter a command.

4.2 Input

4.2.1 Selecting Input

Press the <INPUT> key (Table 2-1). Press the <SHIFT> key and use the operand keys to enter the block type (OB, PB, FB) and the block number (1 to 255). Terminate the entry and execute the command with the <ENTER> key. The example (Figure 4-2) shows the entry of OB 1.

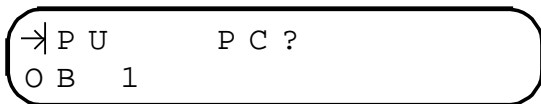


Figure 4-2. Example for Entering OB 1

4.2.2 Entering Statements

You can enter statements in the INPUT and DISPLAY functions.

Select an OB, PB or FB, and enter, for example, the statement "A I 1.0". Press the <ENTER> key. The STEP address counter will display the next address. Enter your program up to the last statement. Terminate the entry by pressing the <ENTER> key once again. The following display appears:



Press the <ENTER> key again to transfer the block into the programmable controller.

If you press the <BREAK> key twice, the programmer aborts input (which is not stored) and returns to the previous function.

The block remains in the programmer after aborting entry, and you can continue to manipulate it by pressing <DISPLAY> and <ENTER>.

Note:

If you wish to display a block on the PG 635, 675, 685 and 695 CRT-based programmers, you must enter "BLD 255" (End of segment) after a maximum of 255 STEP 5 statements.

4.2.3 Entering Labels (Only in FBs)

You can enter labels in the INPUT and DISPLAY functions.

In the INPUT function, labels consist of an "X" and a two-digit number and in the DISPLAY and STATUS functions of an "M" and a number (0 to 99). Labels are always assigned to a statement and can only be transferred, inserted or removed together with the statement.

To enter label 10, for example, proceed as follows: press the <ARROW LEFT> key (Table 2-2) to insert the label, then the <1> and <0> keys for the label number and, finally, the <ARROW RIGHT> key to terminate the entry.

You can either enter the associated statement (4.2.2), overwrite the label by repeating the entry or delete the label by pressing the <CLEAR> key.

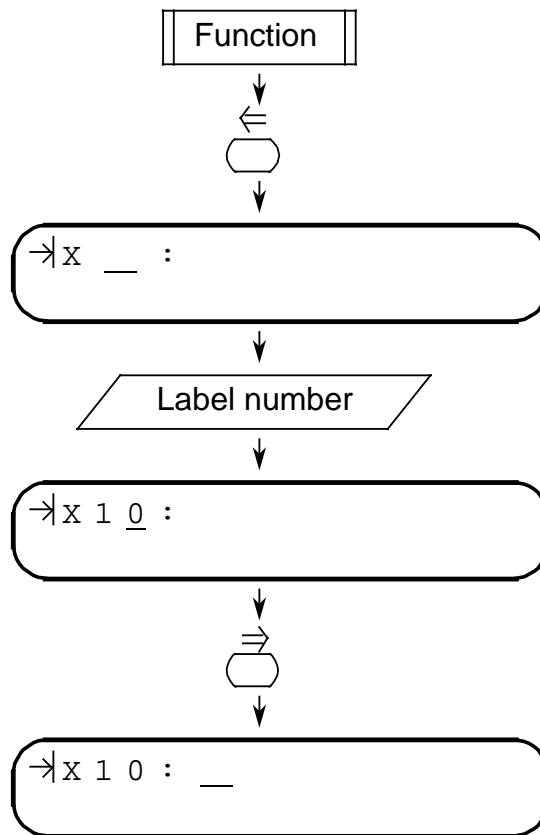


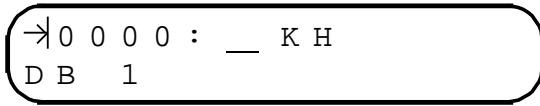
Figure 4-4. Flowchart for Entering Labels

Note:

New labels are generated when transferring blocks from the PLC to the programmer. You can rename the labels you have entered.

4.2.4 Programming Data Blocks

To program a data block, select the INPUT or DISPLAY function. Press the <SHIFT> and <*> keys, and enter the block number instead of entering "FB", "OB" or "PB". The following display appears when you press the <ENTER> key:



You can now switch to other formats using the <ARROW RIGHT> key.

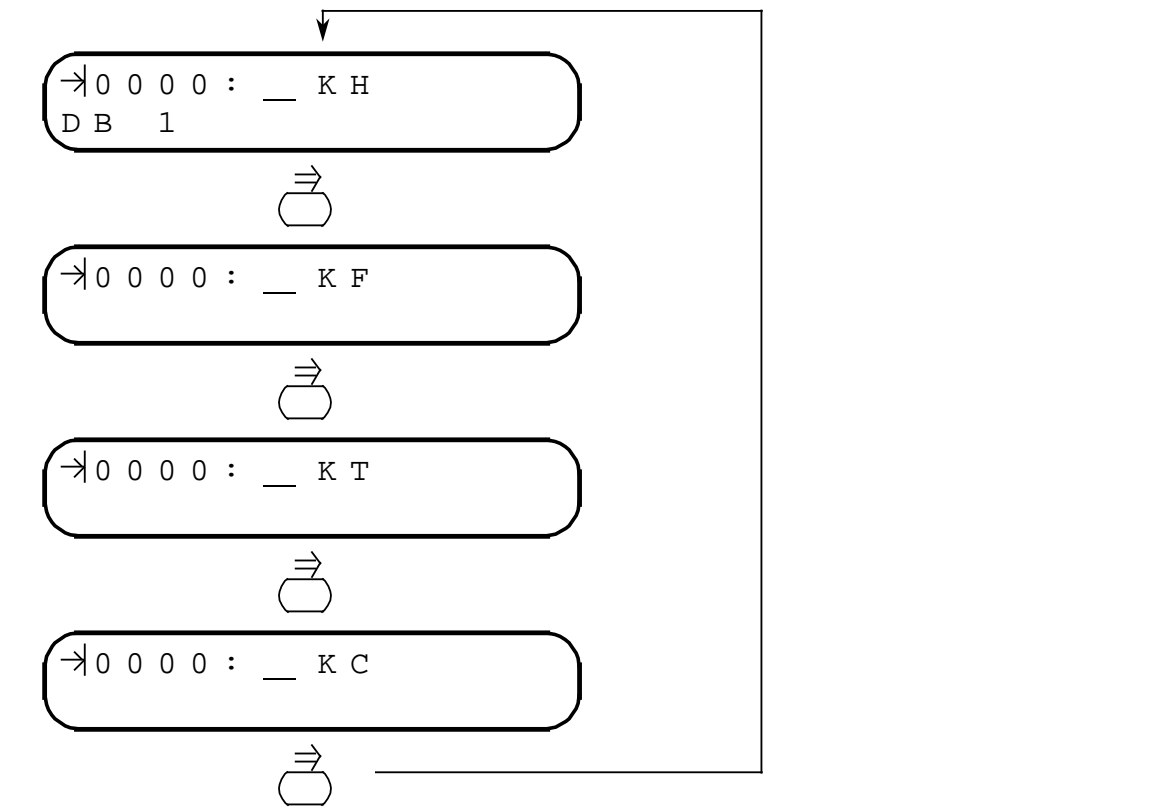


Figure 4-5. Display of the Various Data Formats

You can enter the following formats:

- KH Constant (Hexadecimal): 0 to FFFF
 <SHIFT> key<0> = A
 <SHIFT> key<1> = B
 <SHIFT> key<2> = C
 <SHIFT> key<3> = D
 <SHIFT> key<4> = E
 <SHIFT> key<5> = F
- KF Constant (Fixed-point number): -32768 to 32767
- KT Constant (Time): 0.0 to 999.3
- KC Constant (Counter): 0 to 999

After you have entered a data word, you can key in a repetition factor (1 to 255) after first pressing the <SHIFT> and <*> keys. The number can then be inserted with the <INSERT> key or overwritten with the <ENTER> key.

4.3 Display

4.3.1 Display from the PLC

To select the DISPLAY function, press the <DISPLAY> key (Table 2-1). Then enter the block type and block number using the <SHIFT> key and the operand keys. Output to the display is terminated and the command executed with the <ENTER> key. The first statement appears in the display.

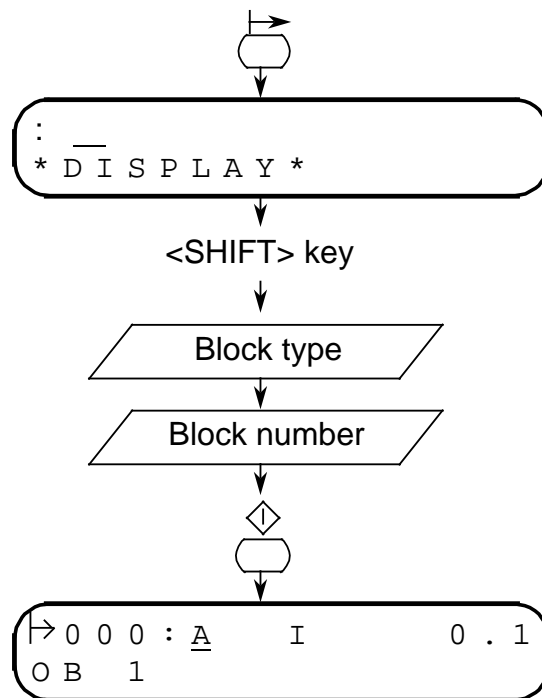


Figure 4-6. Example for Displaying OB 1 from the PLC

You can now display the subsequent statements by pressing the <ARROW LEFT> key. See 4.4 for corrections.

4.3.2 Display from the Programmer

Press the <DISPLAY> key (Table 2-1.) and the <ENTER> key. If you have just terminated an entry or display with the <BREAK> key, you can display the block by pressing the <DISPLAY> and <ENTER> keys.

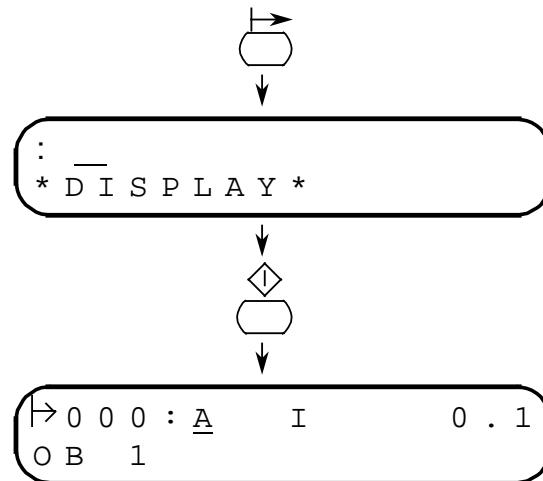


Figure 4-7. Example for Displaying OB 1 from the Programmer

If the block has already been overwritten in the PLC, error message "*84" appears (block not in programmer; Appendix B). If you want to process this block, you must first transfer it from the PLC to the programmer (4.3.1).

4.4 Correction

Corrections are possible in the INPUT and DISPLAY functions.

4.4.1 Deleting Statements

Select the desired statement with the cursor control keys or by using the Search function (4.6). Press the <SHIFT> key and the <INSERT> key. The statement selected is deleted, and the next statement is displayed.

4.4.2 Inserting Statements

Use the cursor control keys or the Search function to select the statement in front of which the new statement is to be inserted. Enter the new statement, e.g. A I 1.0. Press the <INSERT> key. The new statement is inserted and the display shows the statement in front of which the new statement has been inserted.

4.4.3 Overwriting Statements

Select the statement to be overwritten with the cursor control keys or by the Search function. Enter the new statement, e.g. AN I 1.0. Press the <ENTER> key. The statement is overwritten and the next statement is displayed.

4.5 Cursor Control

You can control the cursor in the INPUT, DISPLAY and STATUS (program-dependent signal status display) functions, and also with the Search function (4.6).

Press the <ARROW UP> key to display the previous statement and the <ARROW DOWN> key for the next statement.

Note:

Both cursor control keys are of the auto-repeat type, i.e. the function is automatically repeated if they are held down.

4.6 Search Function

The Search function is possible in the INPUT, DISPLAY and STATUS (program-dependent signal status display) functions. This function searches through the entire user program in the programmer. The search begins and ends at the address displayed before commencement of the search function.

4.6.1 Searching for Statements or Operands

After selecting the function, enter the search item, e.g. A I 1.0 (statement) or I 1.0 (operand). Press the <SEARCH> key (Table 2-1). When the search item has been found, the statement with the corresponding address is displayed. Press the <SEARCH> key again to continue searching. You can quit the search by pressing any other key. If the search item is not present, message *86 is displayed (Appendix B).

Note:

The shift key must be pressed before the B, Y or W when entering "IB", "QB", "FY", "IW", "QW" or "FW".

4.6.2 Searching for Labels (Only in FBs)

After selecting the Search function, press the <ARROW LEFT> key. Enter the label, e.g. F10, and press the <SEARCH> key. The desired label appears in the display along with the corresponding statement.

4.6.3 Searching for Addresses

After selecting the Search function, start the address search by pressing the <SPECIAL> key (Table 2-1). Enter the address you are looking for. The search is started when the <SEARCH> key is pressed. If you have not yet started the search with the <SEARCH> key and you want to stop, press the <CLEAR> key. The last statement is then displayed.

Hexadecimal numbers can be entered using the <SHIFT> key as follows:

| | |
|-----------------|-------|
| <SHIFT> key <0> | for A |
| <SHIFT> key <1> | for B |
| <SHIFT> key <2> | for C |
| <SHIFT> key <3> | for D |
| <SHIFT> key <4> | for E |
| <SHIFT> key <5> | for F |

Note:

If the address entered is greater than the number of addresses in the user program, the last statement of the user program is displayed.

If the search address is in a multi-word statement (in two-word statements, e.g.L KF +1), the next statement is displayed.

4.7 Test Functions

4.7.1 STATUS (Program-Dependent Signal Status Display)

The program-dependent signal status display is an image of the current process status shown on the program display and is updated cyclically in the display.

Note:

The PLC must be in RUN mode for program-dependent signal status display.

Start program-dependent signal status display by pressing the <FORCE> key (Table 2.1). After pressing the <SHIFT> key, enter the block type (OB, PB, FB) and the block number (0 to 255). Press the <ENTER> key to execute the function. The first statement of the block selected appears in the display (4.5 and 4.6).

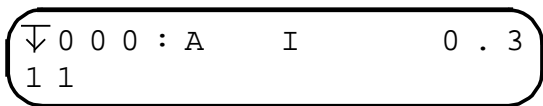


Figure 4-8. Typical STATUS Display

Press the <BREAK> key to return to the initial state.

Representing the Signal State

The signal state for the selected statement is shown in the second line of the display.

| | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|
| Digital operation | <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td>R</td><td></td><td></td><td>X</td><td>X</td><td>X</td><td>X</td><td></td><td></td><td>Y</td><td>Y</td><td>Y</td><td>Y</td><td></td><td></td><td></td><td></td> </tr> </table> <p>R = RLO XXXX = ACCUM 1 YYYY = ACCUM 2</p> | R | | | X | X | X | X | | | Y | Y | Y | Y | | | | |
| R | | | X | X | X | X | | | Y | Y | Y | Y | | | | | | |
| Binary operation | <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td>R</td><td>S</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table> <p>R = RLO S = SIG</p> | R | S | | | | | | | | | | | | | | | |
| R | S | | | | | | | | | | | | | | | | | |
| Miscellaneous operations | <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td>R</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table> <p>R = RLO</p> | R | | | | | | | | | | | | | | | | |
| R | | | | | | | | | | | | | | | | | | |
| Counter operation | <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td>R</td><td>S</td><td></td><td>K</td><td>Z</td><td>:</td><td>X</td><td>X</td><td>X</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table> <p>R = RLO S = SIG XXX = Count</p> | R | S | | K | Z | : | X | X | X | | | | | | | | |
| R | S | | K | Z | : | X | X | X | | | | | | | | | | |
| Timer operation | <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td>R</td><td>S</td><td></td><td>K</td><td>T</td><td>:</td><td>X</td><td>X</td><td>X</td><td>.</td><td>Y</td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table> <p>R = RLO S = SIG XXX = Time Y = Time base</p> | R | S | | K | T | : | X | X | X | . | Y | | | | | | |
| R | S | | K | T | : | X | X | X | . | Y | | | | | | | | |

RLO = Result of logic operation
SIG = Signal state

Figure 4-9. Representation of the Signal State

4.7.2 STATUS VAR (Direct Signal Status Display)

The STATUS VAR function enables you to display current binary signal states or digital values for specific operands, such as are found during program scanning in the PLC at the system checkpoint (point prior to scanning the cyclic S5 program).

The STATUS VAR function is started with the <FORCE> key. Press the <ENTER> key after entering the operand identifier and the parameters (e.g. I 1.0, T 1, QB 1). The signal state appears in the display. Press the <BREAK> key to return to the initial state.

Note:

The <SHIFT> key must be pressed before the "B" or "Y" when entering "IB", "QB" or "FY".

4.7.3 FORCE VAR

The FORCE function can only be implemented in connection with STATUS VAR. Input bytes, output bytes and flag bytes can be forced.

Select direct signal state display first (4.7.2). The following appears when you press <FORCE>.

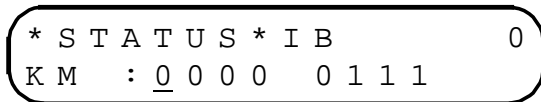


Figure 4-10. Typical FORCE Display

You can now position the cursor with the <ARROW RIGHT> and <ARROW LEFT> keys and set the signal state. "1" signifies "Signal state 1", "0" signifies "Signal state 0". Press the <ENTER> key to execute the FORCE function, and observe the signal state.

Press <BREAK> to return to the STATUS VAR function. The FORCE function is not executed.

4.7.4 Manipulating Timers and Counters (from Version V1.2 Onward)

You may have to adapt the user program when using the PG 605U from version V1.2 onward in order to be able to manipulate timers and counters.

The PLC can be in the RUN or STOP mode. As with the FORCE VAR function, the manipulating of timers and counters can only be implemented within the direct signal state display function.

The manipulating of timers and counters affects the data block set in the programmer. The data word with the number of the counter or timer to be manipulated is influenced.

The preset DBs are:

- DB11 for timers
- DB12 for counters

Program Structure

The following sequence of operations must be programmed in the user program to transfer the setting to the timer:

```
Example: C DB 11 * ; The reserved data block is set
         L DW 5    ; Data word 5 is loaded into the accumulator
         SI T5     ; Start timer T5 as pulse
```

Note:

For reasons of clarity, the number of the data word and the number of the timer or counter should be identical.

Timers and counters in the S5-101U are located in different areas in the data block. In the other PLCs, timers and counters can have their own data block.

Permissible parameters for timers and counters:

| | |
|----------|----------------------------------|
| 0 to 15 | S5-100U CPU 100 |
| 0 to 31 | S5-90U, S5-100U CPU 102 |
| 0 to 127 | S5-95U, S5-100U CPU 103, S5-115U |

* Not required in the case of the S5-101U

Special Features of the S5-101U

Since there is only one data block in the S5-101U for timers and counters, data words 0 to 15 are accessed for manipulating timers, and data words 16 to 32 for manipulating counters.

Permissible parameters for timers and counters: 0 to 15

Assignment for timers:

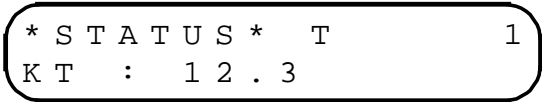
DW 0 : T0
 DW 1 : T1
 :
 :
 DW 15 : T15

Assignment for counters:

DW 16 : Z0
 DW 17 : Z1
 :
 :
 DW 31 : Z15

Note:
 Before the programmable controller is started, all variable timers and counters must be preset with values that are not critical for the process to be controlled.

Select STATUS VAR and press the <STATUS> key (Table 2-1.). DB 11 appears as the default in the display (not in the case of the S5-101U). Type in the block containing the data word to be manipulated. Press the <ENTER> key to transfer the DB setting. The time set appears in the display. It consists of a value between 1 and 999 (before the point) and the time base 0 to 3 (after the point).



Meaning of the Time Base

| | | | | |
|--------|-------|------|----|-----|
| Base | 0 | 1 | 2 | 3 |
| Factor | 0.01s | 0.1s | 1s | 10s |

You can enter a new time after pressing the <STATUS> key. Press the <ENTER> key to transfer the value entered and to store it in the data block. Press the <BREAK> key to return to the initial state.

If either the block selected or the relevant data word is not available, message "*88" appears (Appendix B).

4.8 Enquiry Functions

4.8.1 Directory Functions

DIR (all blocks)

Select the DISPLAY function with the <DISPLAY> key and enter a <1>. Press the <ENTER> key. The first block of the block list and the relevant initial address are displayed. You can scroll through the block list with the <ARROW DOWN> key. You can terminate block list display and return to the initial state with the <BREAK> key.

DIR (individual blocks)

Select the DISPLAY function with the <DISPLAY> key and enter a <1>. Then enter the block type (OB, PB, FB or "*" for DB) and the block number (0 to 255). Press the <ENTER> key. The data of the block selected appears in the display. Press the <BREAK> key to return to the initial state.

4.8.2 Interrupt Stack (ISTACK)

There can be various reasons for the PLC stopping, e.g. the user changes the operating mode or there is a fault in the PLC. You can find out the reason for the PLC stopping with the help of the ISTACK, which is displayed as a bit pattern and in hexadecimal code.

Select the DISPLAY function with the <DISPLAY> key and enter a <2>. Press the <ENTER> key. The interrupt stack is displayed as a bit pattern. See Tables 4-1 and 4-2 of your programmable controller manual for an explanation of the display.

You can scroll the display with the <ARROW DOWN> key. Press the <BREAK> key to return to the initial state.

Table 4-1. Interrupt Stack (S5-90U, S5-95U, S5-100U, S5-101U, S5-115U)

| Byte | ISTACK Display | | | | | | | | System Data Word |
|------|-------------------|----------------|------------|--------------|----------------|---------------|-----------|--------------|------------------|
| 1 | | PBST SCH | BST SCH | SCH TAET | ADR BAU | SPAB BR | NAU AS | QUITT | SD5 |
| 2 | CA-DA | CE-DA | | REMA NENZ | | | | | |
| 3 | STOP ZU- STAND | STOP AN- ZEIGE | NEU- START | | BATT- PUFFER | | BARB | BARB ENDE | SD6 |
| 4 | | UAFEHL | MAFEHL | E0VH | | AF | | | |
| 5 | ASP NUR EPROM | ASP NUR RAM | KOPF NINT | PROM SCH END | ASP NUR EEPROM | PROM ADR FEHL | ASP LÜCKE | RAM ADR FEHL | SD7 |
| 6 | KEIN ASP | SYNCH FEHL | NINEU | | | | SUMF | UR LADEN | |
| 7 | IRRELEVANT | | | | | | | | |
| 8 | IRRELEVANT | | | | | | | | |
| 9 | STOP SCHAL- TER | | SUF | TRAF | NNN | STS | STUEB | FEST | SD214 |
| 10 | NAU | QVZ | KOLIF | ZYK | SYSFE | PEU | BAU | ASP FA | |
| 11 | | | | | | | | | SD213 |
| 12 | ANZ1 | ANZ0 | OVF | | OR | STATUS | VKE | ERAB | |
| 13 | 6th nesting level | | | | | OR | VKE | FKT | SD212 |
| 14 | | | | | | | | | |
| 15 | 4th nesting level | | | | | OR | VKE | FKT | SD211 |
| 16 | 5th nesting level | | | | | OR | VKE | FKT | |

Table 4-1. Interrupt Stack (S5-90U, S5-95U, S5-100U, S5-101U, S5-115U) (Continued)

| Byte | ISTACK Display | | | | System Data Word |
|------|---------------------------------------|----|-----|-----|------------------|
| 17 | 2nd nesting level | OR | VKE | FKT | SD210 |
| 18 | 3rd nesting level | OR | VKE | FKT | |
| 19 | Nesting depth (0 to 6) | | | | SD209 |
| 20 | 1st nesting level | OR | VKE | FKT | |
| 21 | Starting address of data block (high) | | | | SD208 |
| 22 | Starting address of data block (low) | | | | |
| 23 | Block stack pointer (high) | | | | SD207 |
| 24 | Block stack pointer (low) | | | | |
| 25 | Step address counter (high) | | | | SD206 |
| 26 | Step address counter (low) | | | | |
| 27 | Instruction register (high) | | | | SD205 |
| 28 | Instruction register (low) | | | | |
| 29 | ACCUM 2 (high) | | | | SD204 |
| 30 | ACCUM 2 (low) | | | | |
| 31 | ACCUM 1 (high) | | | | SD203 |
| 32 | ACCUM 1 (low) | | | | |

Table 4-2. Explanation of the ISTACK Bits

| Byte | ISTACK Display | Explanation |
|------|---|--|
| 1 | ENDSCH: PBSTSCH: BSTSCH: SCHTAET: ADRBAU: SPABBR: NAUAS: QUITT: | Terminate block shift Shift block prior to inserting PROM Shift block Shift in progress Address list generation Memory shift abort Interface module power failure Acknowledgement for SHIFT BLOCK |
| 2 | CA-DA: CE-DA: NSTOP ANZEIGE: REMANENZ: | Interprocessor communication flag output address list available Interprocessor communication flag input address list available Startup already executed following Overall Reset Set for retentive |
| 3 | STOPZUSTAND: STOPANZEIGE: NEUSTART: WIEDERANLAUF: BATTPUFFER: DATUMEIN: BARB: BARBENDE: | PLC at "STOP" (external request) PLC at "STOP" (internal request) PLC in cold restart routine Request for PLC operation Battery backup available for power supply unit Contents of date and time of day locations no longer valid PLC in program check mode PLC flagged end of program check |
| 4 | UAFEHL: MAFEHL: E0VH: WANAU: AF: OBWIED AKT: OBNAU AKT: | Interrupt error word invalid Group flag for machine error word SD7 Process interface module I/O available Warm restart executed after power failure Interrupt handling enabled Warm restart OB executing Warm restart OB (after power failure) executing |
| 5 | TEST BST NIO: QVZ TEST NIO: ASP NUR EPROM: ASP NUR RAM: KOPFNINT: PROMSCHEND: ASP NUR EEPROM: WECKFEHL: PROMADRFEHL: ASPLUECKE: RAM ADR FEHL: | Bad test block Bad time-out test Only EPROM user memory available Only RAM user memory available Block header cannot be interpreted Shift terminated before EPROM was inserted Only EEPROM user memory available Error in time interrupt service routine Addressing error in EPROM Address gap in user memory Addressing error in user RAM |

Table 4-2. Explanation of the ISTACK Bits (Continued)

| Byte | ISTACK Display | Explanation |
|------|---|---|
| 6 | KEIN ASP: SYNCHFEHL: NINEU: NIWIED: RUFNVHBST: QVZNINT: SUMF: URLADEN: | No user memory module inserted Synchronization error Cold restart not possible Warm restart not possible Non-existent block invoked Time-out cannot be interpreted Checksum error Bootstrap |
| 9 | STOPSCHALTER: TF: SUF: TRAF: NNN: STS: STUEB: FEST: | Mode selector at STOP Test panel Substitution error Transfer error Programming error, illegal operation or illegal block call Programmable STOP Block stack overflow or parameter list for block already assigned Error in CPU self-test routine |
| 10 | NAU: QVZ: KOLIF: ADF: ZYK: SYSFE: PEU: TI: BAU: ASPFA: | Power failure Time-out Errors in interprocessor communication flag transfer list Addressing error Scan time exceeded Error in SYSID block I/O not ready Time processing interrupted by NAU/BAU Battery failure Wrong user memory |
| 12 | ANZ1/ANZ0: OVF: CARRY: OR: STATUS: VKE: ERAB: | Condition codes bits for arithmetic, logic and shift operations (CC1/CC2) Arithmetic overflow Carry Identification bit for OR buffer Status identifier Result of logic operation (RLO) Identification bit for first scan |
| 13 | OR: VKE: FKT: | OR value for nesting level RLO for nesting level Function value for nesting level ("A" or "O") |

4.8.3 Displaying the System Parameters

The system parameters (words) are displayed in hexadecimal code.

Select the DISPLAY function with the <DISPLAY> key and enter a <3>. Press the <ENTER> key. The system parameters appear in the display. See Table 4-4 and your programmable controller manual for an explanation of the display.

You can scroll the display and view other system parameters with the <ARROW DOWN> key. Press the <BREAK> key to terminate system parameter display and return to the initial state.

Table 4-3. System Parameters

| System Parameters | |
|-------------------|---|
| PAR 1 | Input modules 1 |
| PAR 2 | Output modules 1 |
| PAR 3 | Process input image 1 |
| PAR 4 | Process output image 1 |
| PAR 5 | Flag memory 1 |
| PAR 6 | Time register 1 |
| PAR 7 | Data register 1 |
| PAR 8 | SD area in PLC memory 1 |
| PAR 9 | PLC software release |
| PAR 10 | End address of user memory 1 |
| PAR 11 | System program memory 1 |
| PAR 12 | Length of the DB list ² |
| PAR 13 | Length of the SB list ² |
| PAR 14 | Length of the PB list ² |
| PAR 15 | Length of the FB list ² |
| PAR 16 | Length of the OB list ² |

¹ Address

² Length in bytes

Table 4-3. System Parameters (Continued)

| System Parameters | |
|-------------------|--|
| PAR 17 | Length of the TB list ¹ |
| PAR 18 | Reserve |
| PAR 19 | Length of DB0 list ¹ |
| PAR 20 | Controller input buffer, 1st or 2nd CPU identifier (Table 4-5.) |
| PAR 21 | Length of block header |
| PAR 22 | CPU identifier (Table 4-5.) |
| | Programmer interface software release |

¹ Length in bytes

Table 4-4. CPU Identifier in SYSPAR

| Bit | | CPU identifier of the PLC | Bit | | | | 2nd CPU identifier of the PLC |
|----------------|----------------|--|----------------|----------------|----------------|----------------|---------------------------------------|
| 2 ¹ | 2 ⁰ | | 2 ³ | 2 ² | 2 ¹ | 2 ⁰ | |
| 0 | 0 | S5-90U/S5-95U/S5-100U/ S5-101U/ S5-115U | 0 | 0 | 0 | 1 | S5-90U/S5-95U/ S5-100U |
| 1 | 1 | no CPU | 0 | 0 | 1 | 0 | S5-101U |
| | | | 0 | 1 | 0 | 0 | S5-115U |
| | | | 1 | 1 | 1 | 1 | CPU identifier in byte 44 is valid |

To display new PLCs in the SYSPAR display, a second CPU identifier must be defined. The "Controller input buffer -1" information (bytes 39 and 40 from the SYSPAR) is not evaluated by the PG 605U. The second CPU identifier is coded in byte 40.

4.9 Special Functions

4.9.1 Starting and Stopping the PLC via the Programmer

To execute the PLC START special function, the mode selector on the PLC must be set to RUN. Select the PLC START function with the <SPECIAL> key (Table 2-1.) and enter a <2>. Press the <ENTER> key. The display prompts you to confirm that the PLC is to be started. Press the <ENTER> key. If you want to return to the initial state, press the <BREAK> key.

Select the PLC STOP function with the <SPECIAL> key (Table 2-1.) and enter a <1>. Press the <ENTER> key. The display prompts you to confirm that the PLC is to be stopped. Press the <ENTER> key. If you want to return to the initial state, press the <BREAK> key.

4.9.2 Compressing the PLC Memory

The COMPRESS PLC MEMORY function using the <SPECIAL> and <3> keys is used to remove invalid blocks in the PLC (resulting from corrections and block deletions) and thus compress the PLC memory.

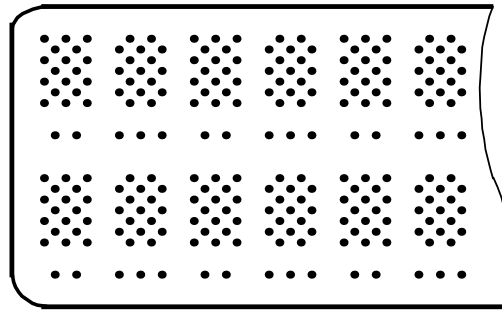
Select the COMPRESS PLC MEMORY function with the <SPECIAL> and <3> keys. Press the <ENTER> key to compress the PLC memory.

4.9.3 Display Check

You can select the DISPLAY CHECK function with the <SPECIAL> and <4> keys. Press the <ENTER> key. A test pattern appears in the display. You can check this display for errors.

You can invert the display with the <ARROW DOWN> key. Press the <ARROW DOWN> key again to change the pattern back.

The display check is terminated approximately 6 s after pressing the <BREAK> key.



Test pattern inverted

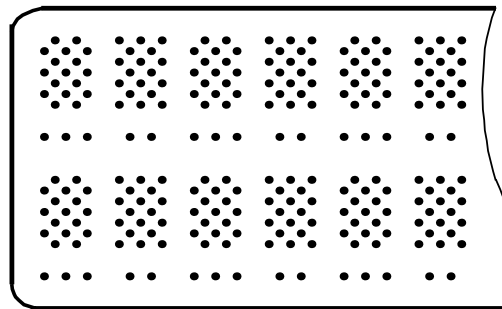


Figure 4-11. Test Pattern of the PG 605U Display Check

4.9.4 Key Check

The key check is started by pressing the <SPECIAL> and <5> keys. Press the <ENTER> key to transfer the command and execute the test.

Note:

The key check cannot be aborted.

Numbers are assigned to the individual keys. The display prompts you to press the relevant key.

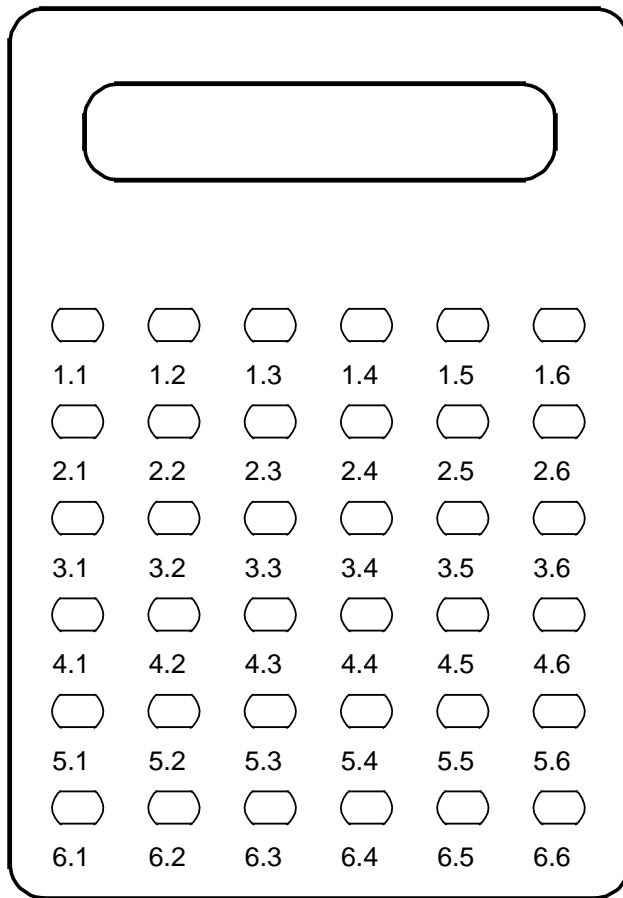


Figure 4-12. Assignment of the Keys for the Key Check

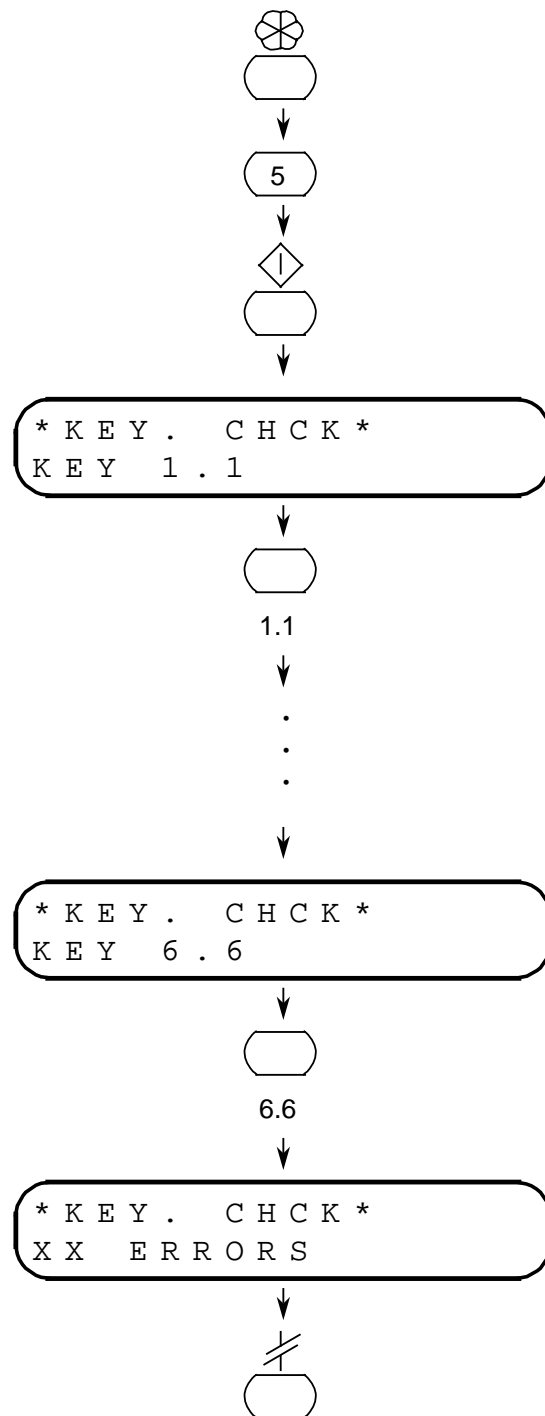


Figure 4-13. Schematic Sequence of the Key Check

The number of errors appears in the second line of the display.

After completion of the key check, press the <BREAK> key to return to the initial state.

4.10 Delete Functions

4.10.1 Overall Reset of the PLC

Note:

The PLC must be stopped for the Overall Reset function.

Select the Overall Reset function with the <SHIFT> and <INSERT> keys (Table 2-1). After pressing the <ENTER> key, the display prompts you to confirm that Overall Reset of the PLC is to be executed. Press the <ENTER> key again to execute the command for Overall Reset. If you want to return to the initial state instead, press the <BREAK> key.

4.10.2 Deleting Blocks

Note:

The PLC must be stopped for the DELETE BLOCK function (not in the case of the S5-115U).

Select the DELETE BLOCK function by pressing the <SHIFT> and <INSERT> keys (Table 2-1). Press the <SHIFT> key again and enter the block type (OB, PB, FB or "*" for DB) and the block number (0 to 255). If you press the <ENTER> key, the display prompts you to confirm that the block indicated is to be deleted. Press the <ENTER> key again to execute the command and erase the block in question. If you want to return to the initial state instead, press the <BREAK> key.

- 1 System Overview
- 2 Technical Specifications
- 3 Installation Guidelines
- 4 Programming and Operator-Process Communication

Appendices

- Appendix A STEP 5 Operations
- Appendix B Error Codes
- Appendix C Function Overview
- Appendix D SIEMENS Addresses Worldwide

| A STEP 5 Operations | | |
|----------------------------|------------------------------|-------|
| A.1 | Logic Operations | A.- 1 |
| A.2 | Set/Reset Operations | A.- 3 |
| A.3 | Compare Operations | A.- 3 |
| A.4 | Arithmetic Operations | A.- 4 |
| A.5 | Timer and Counter Operations | A - 5 |
| A.6 | Load Operations | A.- 6 |
| A.7 | Transfer Operations | A.- 7 |
| A.8 | Block Calls | A.- 7 |
| A.9 | Bit Test Operations | A.- 8 |
| A.10 | Jump Operations | A.- 8 |
| A.11 | Miscellaneous Operations | A - 8 |

A STEP 5 Operations

Appendix A contains an overview of the STEP 5 operation set. Each character in a STEP 5 operation is entered by pressing the corresponding key (Table 2-1). The "=" character is used as the "Assign" key.

A.1 Logic Operations

| Operation | Parameter | Function group * | | | Keys | Parameter |
|-----------|--------------|------------------|---|---|--------|-----------|
| | | B | E | S | | |
| A I | 0.0 to 127.7 | X | | | A I | |
| A Q | 0.0 to 127.7 | X | | | A Q | " |
| A F | 0.0 to 255.7 | X | | | A F | " |
| AN I | 0.0 to 127.7 | X | | | A N I | " |
| AN Q | 0.0 to 127.7 | X | | | A N Q | " |
| AN F | 0.0 to 255.7 | X | | | A N F | " |
| O I | 0.0 to 127.7 | X | | | O. I | " |
| O Q | 0.0 to 127.7 | X | | | O. Q | " |
| O F | 0.0 to 255.7 | X | | | O. F | " |
| ON I | 0.0 to 127.7 | X | | | O. N I | " |
| ON Q | 0.0 to 127.7 | X | | | O. N Q | " |
| ON F | 0.0 to 255.7 | X | | | O. N F | " |
| A T | 0 to 127 | X | | | A T | " |
| AN T | 0 to 127 | X | | | A N T | " |
| A C | 0 to 127 | X | | | A C | " |
| AN C | 0 to 127 | X | | | A N C | " |
| O T | 0 to 127 | X | | | O. T | " |
| ON T | 0 to 127 | X | | | O. N T | " |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

| Operation | Parameter | Function group * | | | Keys |
|-------------------------|-----------|------------------|---|---|--|
| Binary logic operations | | B | E | S | |
| O C | 0 to 127 | X | | | <input type="checkbox"/> O. <input type="checkbox"/> C Parameter |
| ON C | 0 to 127 | X | | | <input type="checkbox"/> O. <input type="checkbox"/> N <input type="checkbox"/> C " |
|) | | X | | | <input type="checkbox"/>) |
| A(| | X | | | <input type="checkbox"/> A(|
| O(| | X | | | <input type="checkbox"/> O(|
| O | | X | | | <input type="checkbox"/> O |

| Operation | Parameter | Function group * | | | Keys |
|-------------------------|-----------|------------------|---|---|--|
| Logic operations (word) | | B | E | S | |
| UW | | | X | | <input type="checkbox"/> * <input type="checkbox"/> 2 <input type="checkbox"/> 4 |
| OW | | | X | | <input type="checkbox"/> * <input type="checkbox"/> 2 <input type="checkbox"/> 5 |
| XOW | | | X | | <input type="checkbox"/> * <input type="checkbox"/> 2 <input type="checkbox"/> 6 |

<Shift> key

- * B = Basic operation set
- E = Expanded operation set
- S = System operations

A.2 Set/Reset Operations

| Operation | | Parameter | Function group * | | | Keys |
|----------------------|---|--------------|------------------|---|---|------------------------|
| Set/reset operations | | | B | E | S | |
| S | I | 0.0 to 127.7 | X | | | [S] [I] Parameter |
| S | Q | 0.0 to 127.7 | X | | | [S] [Q] " |
| S | F | 0.0 to 255.7 | X | | | [S] [F] " |
| R | I | 0.0 to 127.7 | X | | | [R] [I] " |
| R | Q | 0.0 to 127.7 | X | | | [R] [Q] " |
| R | F | 0.0 to 255.7 | X | | | [R] [F] " |
| = | I | 0.0 to 127.7 | X | | | [=] [I] " |
| = | Q | 0.0 to 127.7 | X | | | [=] [Q] " |
| = | F | 0.0 to 255.7 | X | | | [=] [F] " |

A.3 Compare Operations

| Operation | | Parameter | Function group * | | | Keys |
|--------------------|----|-----------|------------------|---|---|-------------|
| Compare operations | | | B | E | S | |
| ! | = | F | X | | | [=] [F] |
| | >< | F | X | | | [>] [<] [F] |
| | > | F | X | | | [>] [F] |
| | >= | F | X | | | [>] [=] [F] |
| | < | F | X | | | [<] [F] |
| | <= | F | X | | | [<] [=] [F] |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

A.4 Arithmetic Operations

| Operation | Parameter | Function group * | | | Keys |
|------------------------------|-----------|------------------|---|---|---|
| | | B | E | S | |
| Arithmetic Operations | | | | | |
| + F | | X | | | <input type="checkbox"/> + <input type="checkbox"/> F |
| - F | | X | | | <input type="checkbox"/> - <input type="checkbox"/> F |
| D | 0 to 255 | | X | | <input type="checkbox"/> * <input type="checkbox"/> 3 <input type="checkbox"/> 1 |
| I | 0 to 255 | | X | | <input type="checkbox"/> * <input type="checkbox"/> 3 <input type="checkbox"/> 0 |
| CFW | | | X | | <input type="checkbox"/> * <input type="checkbox"/> 2 <input type="checkbox"/> 2 |
| CSW | | | X | | <input type="checkbox"/> * <input type="checkbox"/> 2 <input type="checkbox"/> 3 |
| SLW | 0 to 15 | | X | | <input type="checkbox"/> S <input type="checkbox"/> <input type="checkbox"/> L <input type="checkbox"/> W |
| SRW | 0 to 15 | | X | | <input type="checkbox"/> S <input type="checkbox"/> R <input type="checkbox"/> W |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

<Shift> key

A.5 Timer and Counter Operations

| Operation | | Parameter | Function group * | | | Keys |
|------------------------------|---|-----------|------------------|---|---|--|
| Timer and counter operations | | | B | E | S | |
| SI | T | 0 to 127 | X | | | <input type="checkbox"/> SI T Parameter |
| SD | T | 0 to 127 | X | | | <input type="checkbox"/> SD T " |
| SE | T | 0 to 127 | X | | | <input type="checkbox"/> * 1 9 " |
| SS | T | 0 to 127 | X | | | <input type="checkbox"/> * 2 0 " |
| SF | T | 0 to 127 | X | | | <input type="checkbox"/> * 2 1 " |
| R | T | 0 to 127 | X | | | <input type="checkbox"/> R T " |
| S | C | 0 to 127 | X | | | <input type="checkbox"/> S C " |
| R | C | 0 to 127 | X | | | <input type="checkbox"/> R C " |
| CU | C | 0 to 127 | X | | | <input type="checkbox"/> CU C " |
| CD | C | 0 to 127 | X | | | <input type="checkbox"/> CD C " |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

<Shift> key

A.6 Load Operations

| Operation | | Parameter | Function group * | | | Keys | | | | |
|-----------------|----|-------------------|------------------|---|---|--------------------------|---|---|---|-----------|
| Load operations | | | B | E | S | | | | | |
| L | IB | 0 to 127 | X | | | <input type="checkbox"/> | L | I | B | Parameter |
| L | IW | 0 to 126 | X | | | <input type="checkbox"/> | L | I | W | " |
| L | QB | 0 to 127 | X | | | <input type="checkbox"/> | L | Q | B | " |
| L | QW | 0 to 126 | X | | | <input type="checkbox"/> | L | Q | W | " |
| L | FY | 0 to 255 | X | | | <input type="checkbox"/> | L | F | Y | " |
| L | FW | 0 to 254 | X | | | <input type="checkbox"/> | L | F | W | " |
| L | DW | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 0 | " |
| L | DR | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 1 | " |
| L | DL | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 2 | " |
| L | T | 0 to 127 | X | | | <input type="checkbox"/> | L | T | | " |
| L | C | 0 to 127 | X | | | <input type="checkbox"/> | L | C | | " |
| L | PY | 0 to 255 | X | | | <input type="checkbox"/> | L | P | Y | " |
| L | PW | 0 to 254 | X | | | <input type="checkbox"/> | L | P | W | " |
| LC | T | 0 to 127 | X | | | <input type="checkbox"/> | * | 0 | 3 | " |
| LC | C | 0 to 127 | X | | | <input type="checkbox"/> | * | 0 | 4 | " |
| L | KT | 0.0 to 999.3 | X | | | <input type="checkbox"/> | L | | K | T |
| L | KC | 0 to 999 | X | | | <input type="checkbox"/> | L | | K | C |
| L | KF | - 32768 to +32767 | X | | | <input type="checkbox"/> | L | | K | F |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

<Shift> key

A.7 Transfer Operations

| Operation | | Parameter | Function group * | | | Keys | | | | |
|---------------------|----|-----------|------------------|---|---|--------------------------|---|---|---|-----------|
| Transfer operations | | | B | E | S | | | | | |
| T | IB | 0 to 127 | X | | | <input type="checkbox"/> | T | I | B | Parameter |
| T | IW | 0 to 126 | X | | | <input type="checkbox"/> | T | I | W | " |
| T | QB | 0 to 127 | X | | | <input type="checkbox"/> | T | Q | B | " |
| T | QW | 0 to 126 | X | | | <input type="checkbox"/> | T | Q | W | " |
| T | FY | 0 to 255 | X | | | <input type="checkbox"/> | T | F | Y | " |
| T | FW | 0 to 254 | X | | | <input type="checkbox"/> | T | F | W | " |
| T | DW | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 5 | " |
| T | DR | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 6 | " |
| T | DL | 0 to 255 | X | | | <input type="checkbox"/> | * | 0 | 7 | " |
| T | PY | 0 to 255 | X | | | <input type="checkbox"/> | T | P | B | " |
| T | PW | 0 to 254 | X | | | <input type="checkbox"/> | T | P | W | " |

A.8 Block Calls

| Operation | | Parameter | Function group * | | | Keys | | | | |
|-------------|----|-----------|------------------|---|---|--------------------------|----|--------------------------|----|-----------|
| Block calls | | | B | E | S | | | | | |
| JU | PB | 0 to 255 | X | | | <input type="checkbox"/> | JU | <input type="checkbox"/> | PB | Parameter |
| JU | FB | 0 to 255 | X | | | <input type="checkbox"/> | JU | <input type="checkbox"/> | FB | " |
| JC | PB | 0 to 255 | X | | | <input type="checkbox"/> | JC | <input type="checkbox"/> | PB | " |
| JC | FB | 0 to 255 | X | | | <input type="checkbox"/> | JC | <input type="checkbox"/> | FB | " |
| C | DB | 0 to 255 | X | | | <input type="checkbox"/> | * | 1 | 3 | " |
| BEC | | | X | | | <input type="checkbox"/> | * | 1 | 1 | " |
| BEU | | | X | | | <input type="checkbox"/> | * | 1 | 2 | " |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

<Shift> key

A.9 Bit Test Operations

| Operation | Parameter | Function group * | | | Keys |
|---------------------|---------------|------------------|---|---|---|
| Bit test operations | | B | E | S | |
| TB D | 0.0 to 255.15 | | X | | <input type="checkbox"/> * 2 7 Parameter |
| TBN D | 0.0 to 255.15 | | X | | <input type="checkbox"/> * 2 8 " |

A.10 Jump Operations

| Operation | Parameter | Function group * | | | Keys |
|-----------------|-----------|------------------|---|---|--|
| Jump operations | | B | E | S | |
| JU = | 0 to 99 | | X | | <input type="checkbox"/> JU = Parameter |
| JC = | 0 to 99 | | X | | <input type="checkbox"/> JC = " |
| JZ = | 0 to 99 | | X | | <input type="checkbox"/> * 1 4 " |
| JO = | 0 to 99 | | X | | <input type="checkbox"/> * 1 5 " |
| JP = | 0 to 99 | | X | | <input type="checkbox"/> * 1 6 " |
| JM = | 0 to 99 | | X | | <input type="checkbox"/> * 1 7 " |
| JN = | 0 to 99 | | X | | <input type="checkbox"/> * 1 8 " |

A.11 Miscellaneous Operations

| Operation | Parameter | Function group * | | | Keys |
|--------------------------|---|------------------|---|---|---|
| Miscellaneous operations | | B | E | S | |
| NOP 0 | 0 to 255 (Segment end: BLD 255; Chapter x.x; Blank line: BLD 130) | X | | | <input type="checkbox"/> * 0 9 Parameter |
| STP | | X | | | <input type="checkbox"/> * 1 0 " |
| BLD | | X | | | <input type="checkbox"/> * 0 8 " |

* B = Basic operation set
 E = Expanded operation set
 S = System operations

<Shift> key

B Error Codes

B Error Codes

Appendix B presents a list of error codes, a description of each code, and suitable correction measures.

| Error code | Description | Corrective measure |
|-------------------------------------|--|--|
| *** Hardware errors *** | | |
| *01 | Operating system cannot execute on this controller | Replace programmer |
| *02 | Bad PG RAM | Replace programmer |
| *03 | Bad internal processor RAM | Replace programmer |
| *** Interface errors *** | | |
| *04 | Buffer overflow | Retry function |
| *05 | Parity error | Retry function |
| *06 | Line break | Retry function |
| *07 | Timer expired | Retry function, briefly unplugging programmer, if necessary |
| *08 | Operation cannot be interpreted | Retry function, briefly unplugging programmer, if necessary |
| *** Errors flagged by PC *** | | |
| *11 | Block number too high | Rename block |
| *12 | Not enough room in memory or illegal block | Compress PC memory or rename block |
| *14 | Block in EPROM | Remove EPROM (PC Operator's Guide) |
| *15 | No address list or wrong mode | Select correct mode and retry function |
| *18 | Function not possible on this controller | |
| *20 | Bad program memory | Replace program memory |
| *22 | Wrong mode | Set correct mode and retry function |
| *29 | No ISTACK in cycle | Set PC to STOP and retry function |
| *30 | Block/block list does not exist | |
| *32 | DMA error | Retry function |
| *33 | USART error | Retry function, replacing PC or interface module, if necessary |


















































| Error code | Description | Corrective measures |
|--|--|--|
| *** Errors flagged by PC *** | | |
| *35 | PC memory error | Replace PC |
| *39 | No shift due to STOP | Set PC to RUN and retry function |
| *42 | No RAM submodule | |
| *44 | Unknown error | Retry function |
| *45 | Address refer. non-existent submodule | |
| ***Force function errors *** | | |
| *46 | Wrong PC type | |
| *47 | Block too long | |
| *48 | Wrong transfer length | |
| *49 | No CPU | |
| ***Programmer error functions *** | | |
| *50 | Wrong key or key disabled | Press correct key |
| *51 | Unknown command | Enter correct command |
| *56 | End of programmer memory | Terminate block |
| *59 | Label already defined | Enter new label |
| *70 | No such statement | |
| *71 | Address too long | |
| *73 | Parameter out of range | Reenter operation with correct parameter |
| *75 | Invalid STEP 5 statement | Enter correct statement |
| *76 | Operation illegal in this block | |
| *77 | Illegal jump label | Enter permissible jump label |
| *78 | End of programmer memory | Terminate block |
| *79 | Number of jump labels exceeds 99 Block cannot be loaded | |
| *81 | Function illegal in this programmer mode | Change programmer mode |
| *82 | Label missing or jump displacement exceeded | Correct block |
| *83 | Illegal multiplier | Enter permissible value (1 to 255) |

| Error code | Description | Corrective measures |
|------------|--|--|
| *84 | No block in programmer | |
| *85 | Operand cannot be forced | |
| *86 | Search item not found | |
| *87 | Data cannot be displayed in specified format | Choose different format |
| *88 | No data word for timer/counter | Select/create data block |
| *90 | This block cannot be processed | |
| *97 | PRINTER/PC selector on adapter in wrong position | Set selector to PRINTER position Retry function |

C Function Overview

C Function Overview

The table below shows you the main functions of the PG 605U

| Funktion | Tasten | Kapitel |
|---|--|---------|
| Eingabe (Datenbausteine ) |  Baustein  | 4.2 |
| Ausgabe |  Baustein  | 4.3 |
| Löschen von Anweisungen |  | 4.4.1 |
| Einfügen von Anweisungen |  | 4.4.2 |
| Überschreiben von Anweisungen | Neue Anweisung  | 4.4.3 |
| Positionieren |   | 4.5 |
| Suchlauf | Suchbegriff  | 4.6 |
| Status |  Baustein  | 4.7.1 |
| Status Var |  Operand  | 4.7.2 |
| Steuern Var |  Operand   | 4.7.3 |
| Buch (alle Bausteine) |    | 4.8.1 |
| Buch (einzelne Bausteine) |   Baustein  | 4.8.1 |
| Unterbrechungsstack (USTACK) |    | 4.8.2 |
| Systemparameter |    | 4.8.3 |
| AG-Stop |    | 4.9.1 |
| AG-Start |    | 4.9.1 |
| AG-Speicher komprimieren |    | 4.9.2 |
| Anzeigetest |    | 4.9.3 |
| Tastaturtest |    | 4.9.4 |
| AG urlöschen |   | 4.10.1 |
| Baustein löschen |  Baustein  | 4.10.2 |

D SIEMENS Addresses Worldwide

D SIEMENS Addresses Worldwide

European Companies and Representatives

Austria

Siemens AG Österreich
Vienna
Bregenz
Graz
Innsbruck
Klagenfurt
Linz
Salzburg

Belgium

Siemens S.A.
Brussels
Liège
 Siemens N.V.
Brussels
Antwerp
Gent

Bulgaria

RUEN office of the
 INTERPRED corporation,
 agency of the
 Siemens AG Sofia
Sofia

Czechoslovakia

EFEKTIM
 Engineering Consultants,
 Siemens AG
Prague

Denmark

Siemens A/S
Copenhagen, Ballerup
Højbjerg

Federal Republic of Germany

Branch offices of the
 Siemens AG
Berlin
Bremen
Cologne
Dortmund
Düsseldorf
Essen
Frankfurt/Main
Hamburg

Federal Republic of Germany (continued)

Hanover
Leipzig
Mannheim
Munich
Nuremberg
Stuttgart

Finland

Siemens Osakeyhtiö
Helsinki

France

Siemens S.A.
Paris, Saint-Denis
Lyon, Caluire-et-Cuire
Marseilles
Metz
Seclin (Lille)
Strasbourg

Great Britain

Siemens Ltd.
London, Sunbury-on-
 Thames
Birmingham
Bristol, Clevedon
Congleton
Edinburgh
Glasgow
Leeds
Liverpool
Newcastle

Greece

Siemens A.E.
Athens
Thessaloniki

Hungary

SICONTACT GmbH
Budapest

Iceland

Smith & Norland H/F
Reykjavik

Ireland

Siemens Ltd.
Dublin

Italy

Siemens S. p. A.
Milan
Bari
Bologna
Brescia
Casoria
Florence
Genoa
Macomer
Padua
Rome
Turin

Luxemburg

Siemens S.A.
Luxembourg

Malta

J.R. Darmanin & Co., Ltd.
Valletta

Netherlands

Siemens Nederland N.V.
The Hague

Norway

Siemens A/S
Oslo
Bergen
Stavanger
Trondheim

Poland

PHZ Transactor S.A.
Warsaw
Gda sk-Letnica
Katowice

Portugal

Siemens S.R.A.L.
Lisbon
Faro
Leiria
Porto

Romania

Siemens birou de
consultații tehnice
Bukarest

Spain

Siemens S.A.
Madrid

Sweden

Siemens AB
Stockholm
Eskilstuna
Göteborg
Jönköping
Luleå
Malmö
Sundsvall

Switzerland

Siemens-Albis AG
Zürich
Bern
Siemens-Albis S.A.
Lausanne, Renens

Turkey

ETMAŞ
Istanbul
Adana
Ankara
Bursa
Izmir
Samsun

USSR

Siemens AG Agency
Moscow

Yugoslavia

General Export
OOUR Zastupstvo
Belgrade
Ljubljana
Rijeka
Sarajewo
Skopje
Zagreb

Non-European Companies and Representatives**Africa****Algeria**

Siemens Bureau
Alger
Algier

Angola

Tecnidata
Luanda

Burundi

SOGECOM
Bujumbara

Egypt

Siemens Resident
Engineers
Cairo-Mohandessin
Alexandria
Centech
Zamalek-Cairo

Ethiopia

Addis Electrical
Engineering Ltd.
Addis Abeba

Ivory Coast

Siemens AG
Succursale Côte d'Ivoire
Abidjan

Kenya

Achelis (Kenya) Ltd.
Nairobi

Libya

Siemens AG
Branch Office Libya
Tripoli

Mauritius

Rey & Lenferna Ltd.
Port Louis

Morocco

SETEL
Société Electrotechnique
et de Télécommunica-
tions S.A.
Casablanca

Mozambique

Siemens Resident
Engineer
Maputo

Namibia

Siemens Resident
Engineer
Windhoek

Nigeria

Electro Technologies
Nigeria Ltd. (Eltec)
Lagos

Rwanda

Etablissement Rwandais
Kigali

Simbabwe

Electro Technologies
Corporation (Pvt.) Ltd.
Harare

South Africa

Siemens Ltd.
Johannesburg
Cape Town
Durban
Middleburg
Newcastle
Port Elizabeth
Pretoria

Sudan

National Electrical &
Commercial Company
(NECC)
Khartoum

Swaziland

Siemens (Pty.) Ltd.
Mbabane

Tanzania

Tanzania Electrical
Services Ltd.
Dar-es-Salaam

Tunesia

Sitelec S.A.
Tunis

Zaire

SOFAMATEL S.P.R.L.
Kinshasa

Zambia

Electrical Maintenance
Lusaka Ltd.
Lusaka
Mining projects:
General Mining
Industries Ltd.
Kitwe

America**Argentina**

Siemens S.A.
Buenos Aires
Bahía Blanca
Córdoba
Mendoza
Rosario

Bolivia

Sociedad Comercial e
Industrial Hansa Ltd.
La Paz

Brazil

Siemens S.A.
São Paulo
Belém
Belo Horizonte
Brasília
Campinas
Curitiba
Florianópolis
Fortaleza
Porto Alegre
Recife
Rio de Janeiro
Salvador de Bahía
Vitoria

Canada

Siemens Electric Ltd.
Montreal, Québec
Toronto, Ontario

Chile

INGELSAC
Santiago de Chile

Colombia

Siemens S.A.
Bogotá
Baranquilla
Cali
Medellín

Costa Rica

Siemens S.A.
San José

Ecuador

Siemens S.A.
Quito
OTESA
Guayaquil
Quito

El Salvador

Siemens S.A.
San Salvador

Guatemala

Siemens S.A.
Ciudad de Guatemala

Honduras

Representaciones Electro-
industriales S. de R.L.
Tegucigalpa

Mexico

Siemens S.A.
México, D.F.
Culiacán
Gómez Palacio
Guadalajara
León
Monterrey
Puebla

Nicaragua

Siemens S.A.
Managua

Paraguay

Rieder & Cia., S.A.C.I.
Asunción

Peru

Siemsa
Lima

Uruguay

Conatel S.A.
Montevideo

Venezuela

Siemens S.A.
Caracas
Valencia

**United States
of America**

Siemens Energy &
Automation Inc.
Roswell, Georgia

Asia**Bahrain**

Transitec Gulf

Manama

or

Siemens Resident Engineer

Abu Dhabi**Bangladesh**

Siemens Bangladesh Ltd.

Dhaka**Hong Kong**

Jesben & Co., Ltd.

Hong Kong**India**

Siemens India Ltd.

Bombay**Ahmedabad****Bangalore****Calcutta****Madras****New Dehli****Secundarabad****Indonesia**

P.T.Siemens Indonesia

Jakarta

P.T. Dian-Graha ElektriKa

Jakarta**Bandung****Medan****Surabaya****Iran**

Siemens Sherkate

Sahami Khass

Teheran**Iraq**

Samhiry Bros. Co. (W.L.L.)

Baghdad

or

Siemens AG (Iraq Branch)

Baghdad**Japan**

Siemens K.K.

Tokyo**Jordan**Siemens AG (Jordan
Branch)**Amman**

or

A.R. Kevorkian Co.

Amman**Korea (Republic)**Siemens Electrical
Engineering Co., Ltd.**Seoul****Pusan****Kuwait**National & German
Electrical and Electronic
Service Co. (INGEECO)**Kuwait, Arabia****Lebanon**

Ets. F.A. Kettaneh S.A.

Beirut**Malaysia**

Siemens AG

Malaysian Branch

Kuala Lumpur**Oman**

Waleed Associates

Muscat

or

Siemens Resident Engineers

Dubai**Pakistan**Siemens Pakistan
Engineering Co., Ltd.**Karachi****Islamabad****Lahore****Peshawer****Quetta****Rawalpindi****People's Republic of China**Siemens Represen-
tative Office**Beijing****Guangzhou****Shanghai****Philippine Islands**Maschinen & Technik Inc.
(MATEC)**Manila****Qatar**Trags Electrical Engineering
and
Air Conditioning Co.**Doha**

or

Siemens Resident Engineer

Abu Dhabi**Saudi Arabia**Arabia Electric Ltd.
(Equipment)**Jeddah****Damman****Riyadh****Sri Lanka**

Dimo Limited

Colombo**Syria**

Siemens AG

(Damascus Branch)

Damascus**Taiwan**

Siemens Liaison Office

Taipei

TAI Engineering Co., Ltd.

Taipei**Thailand**

B. Grimm & Co., R.O.P.

Bangkok**United Arab Emirates**

Electro Mechanical Co.

Abu Dhabi

or

Siemens Resident Engineer

Abu Dhabi

Scientechnic

Dubai

or

Siemens Resident Engineer

Dubai

Asia (continued)

Yemen (Arab Republic)

Tihama Tractors &
Engineering Co.o., Ltd.

Sanaa

or

Siemens Resident Engineer

Sanaa

Australasia

Australia

Siemens Ltd.

Melbourne

Brisbane

Perth

Sydney

New Zealand

Siemens Liaison Office

Auckland

Alphabetical Keyword Index

| | | | |
|-----------------------------|------------|--|--|
| A | | | |
| Arithmetic operations | A-4 | | |
| B | | | |
| Binary logic operations | A-1 | | |
| Bit test operations | A-8 | | |
| Block call | A-7 | | |
| Buffer | 2-1 | | |
| C | | | |
| Compare operations | A-3 | | |
| Connecting | 3-1 | | |
| Correction | 4-8 | | |
| Counter operations | A-5 | | |
| CPU identifier | 4-20 | | |
| Cursor control | 4-8 | | |
| - key | 2-2 | | |
| D | | | |
| Data blocks | 4-12, 4-13 | | |
| - programming | 4-5 | | |
| Delete functions | 4-25 | | |
| Deletion | | | |
| - block | 4-25 | | |
| - statement | 4-8 | | |
| Desktop use | 3-2 | | |
| Dimensions | 2-1 | | |
| Direct signal state display | 4-12 | | |
| Directory function | 4-14 | | |
| Display | 4-6 | | |
| - from the PLC | 4-6 | | |
| - from the programmer | 4-7 | | |
| - system parameters | 4-19 | | |
| Display check | 4-22 | | |
| E | | | |
| Enquiry function | 4-14 | | |
| Entering | | | |
| - labels | 4-4 | | |
| - statements | 4-2 | | |
| Error codes | B-1 | | |
| F | | | |
| FORCE | 4-11 | | |
| FORCE VAR | 4-11 | | |
| Function key | 2-2 | | |
| H | | | |
| Handheld use | 3-2 | | |
| I | | | |
| Initial state | 4-11 | | |
| Input | 4-2 | | |
| - selection | 4-2 | | |
| Insertion | | | |
| - statement | 4-8 | | |
| Installation guidelines | 3-1 | | |
| Interface | 2-1 | | |
| Interrupt stack | 4-14 | | |
| ISTACK | 4-14 | | |
| - bits | 4-17, 4-18 | | |
| J | | | |
| Jump operations | A-8 | | |
| K | | | |
| Key check | 4-23 | | |
| Keypad | 2-2 | | |
| L | | | |
| Label entry | 4-4 | | |
| Load operations | A-6 | | |
| M | | | |
| Manipulating | | | |
| - counters | 4-12 | | |
| - timers | 4-12 | | |
| Memory | | | |
| - compressing | 4-21 | | |
| Miscellaneous operations | A-8 | | |
| Mounting in a control panel | 3-2 | | |
| O | | | |
| Online | 1-1, 3-1 | | |
| Offline | 1-1, 3-1 | | |
| Operand key | 2-2 | | |
| Operation | | | |
| - arithmetic | A-4 | | |
| - binary logic | A-1 | | |
| - bit test | A-8 | | |
| - compare | A-3 | | |
| - counter | A-5 | | |
| - jump | A-8 | | |
| - load | A-6 | | |
| - miscellaneous | A-8 | | |
| - set/reset | A-3 | | |
| - STEP 5 | A-1 | | |
| - timer | A-5 | | |
| - transfer | A-7 | | |

| | |
|--------------------------|------------|
| Operation key | 2-2 |
| Overall reset | 4-25 |
| Overwriting | |
| - statement | 4-8 |
| P | |
| Power | |
| - supply | 2-1 |
| Program block | 2-1 |
| Program-dependent | |
| signal state display | 4-10 |
| Programming | 4-1 |
| - data blocks | 4-5 |
| S | |
| Search function | 4-8, 4-9 |
| - address | 4-9 |
| - label | 4-9 |
| - operand | 4-9 |
| - statement | 4-9 |
| Search item | 4-9 |
| Set/reset operations | A-3 |
| Shift key | 2-2 |
| Signal state | 4-10 |
| Signal state display | |
| - direct | 4-11 |
| - program-dependent | 4-10 |
| Special function | 4-21 |
| Starting | 4-21 |
| Statement | |
| - deletion | 4-8 |
| - entry | 4-2 |
| - insertion | 4-8 |
| - overwriting | 4-8 |
| STATUS | 4-10 |
| STATUS VAR | 4-11 |
| STEP 5 operations | A-1 |
| Stopping | 4-21 |
| System overview | 1-1 |
| System parameters | 4-19, 4-20 |
| - display | 4-19 |
| T | |
| Technical specifications | 2-1 |
| Test function | 4-10 |
| Test pattern | 4-22 |
| Timer operations | A-5 |
| Transfer operations | A-7 |
| W | |
| Weight | 2-1 |

