

SIEMENS

SIMATIC S7-200

S7-200 One-Hour Primer

Training Documents

Preface

Installing the hardware

1

Installing the software

2

Basic settings

3

First programming exercise

4

More exercises

5

Appendix

A

Safety Guidelines

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.



Danger

indicates that death or severe personal injury **will** result if proper precautions are not taken.



Warning

indicates that death or severe personal injury **may** result if proper precautions are not taken.



Caution

with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

Caution

without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

Notice

indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by **qualified personnel**. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

Prescribed Usage

Note the following:



Warning

This device 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. Correct, reliable operation of the product requires proper transport, storage, positioning and assembly as well as careful operation and maintenance.

Trademarks

All names identified by ® are registered trademarks of the Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Preface

Dear S7-200 user,

A programmable controller runs a control program that you create to solve most automation tasks. The S7-200 Micro PLC can help you win the automation race to improve safety, quality, and production speed while reducing project costs.

The S7-200 Micro PLC is being used for more and more applications because it combines power with an attractive price and simple operation.

To make your first steps in the world of S7-200 as simple as possible, we have created a special Starter Kit.

This One-Hour Primer is intended to teach you the basic skills to be able to use the S7-200 within the shortest possible time.

And now, we wish you a simple and quick start and every success.



Contents of the S7-200 starter kit



Item	Quantity	Available
S7-200 CPU 222 AC/DC Relay	1	
Input Simulator for CPU 222	1	
Software: STEP 7-Micro/WIN V4	1	
Output Simulator on mounting rail	1	
PC/PPI Programming Cable (USB/RS 485)	1	
S7-200 documentation	1	
One Hour Primer (including CD with programming exercises)	1	
Screwdriver	1	

Practice examples for the One-Hour Primer can be found on the STEP 7-Micro/WIN documentation CD at: \English\Manuals\PrimerEx, or on your Programming Exercises CD.

Safety guidelines

The One-Hour Primer is a quick introduction to the world of S7-200. It is not a substitute for the *S7-200 Programmable Controller System Manual*. Therefore, you should observe the instructions given in the *S7-200 Programmable Controller System Manual*, especially the safety guidelines.

Table of contents

	Preface	3
1	Installing the hardware	7
1.1	Assembly and Installation of the Hardware	7
1.2	Details of the S7-200 (CPU 222)	8
1.3	Wiring diagrams for the assembled hardware.....	9
2	Installing the software	13
2.1	Installing STEP 7-Micro/WIN for the One-Hour Primer	13
2.2	Starting STEP 7-Micro/WIN	14
2.3	Using the Help system.....	15
3	Basic settings	17
3.1	Connect the communication cable.....	17
3.2	Configuring the S7-200 communication.....	18
3.3	Performing the first function test	19
4	First programming exercise	21
4.1	Writing your first program	21
4.2	Opening the first programming exercise	22
4.3	Downloading the first exercise program to the CPU	23
4.4	Function and Test of the first exercise program.....	24
4.5	Elements of ladder logic (LAD)	25
4.6	Transforming a circuit diagram	26
4.7	Elements of the first exercise program.....	27
4.8	Status view (online)	28
4.9	Statements	29
5	More exercises	31
5.1	First program modification	31
5.1.1	AND logic operation.....	31
5.1.2	Inserting a logic gate.....	32
5.1.3	Entering the operand and testing.....	33
5.1.4	Deleting a contact or an operand.....	34
5.2	Second modification to the program	35
5.2.1	OR logic operation.....	35
5.2.2	Inserting an OR element.....	36
5.3	Third modification to the program	37
5.3.1	On-delay timer	37

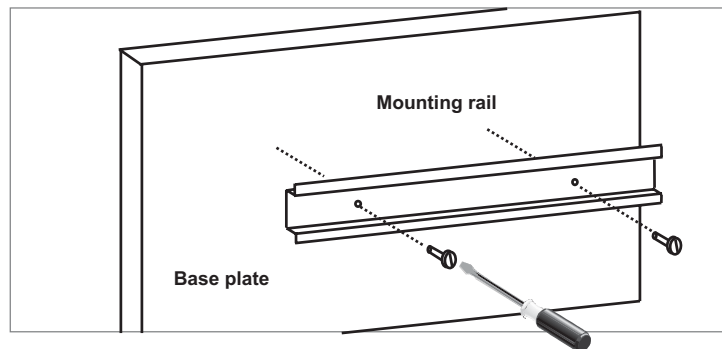
5.3.2	Understanding the on-delay timer function	38
5.3.3	Programming the on-delay timer.....	39
5.4	Working with projects.....	40
5.4.1	Programming with symbols.....	40
5.4.2	Creating a new project.....	42
5.5	Do you want to learn more?.....	45
A	Appendix	47
A.1	Of bits, bytes and words	47
A.2	Address areas of the S7-200	48
A.3	Cyclic program execution in the S7-200 PLC.....	49

Installing the hardware

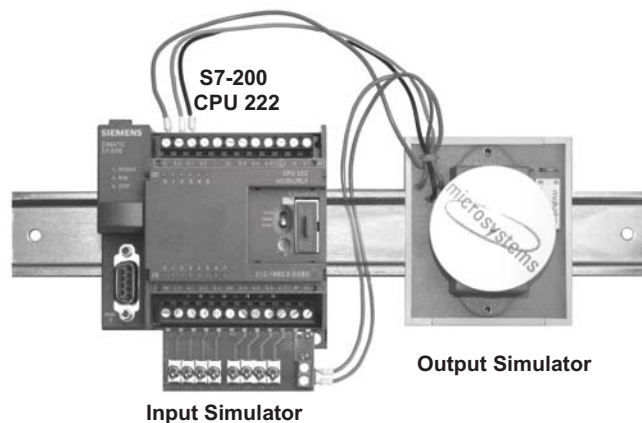
1.1 Assembly and Installation of the Hardware

Assembly and Installation of the Hardware

1. Attach the enclosed mounting rail on a base plate as shown here.

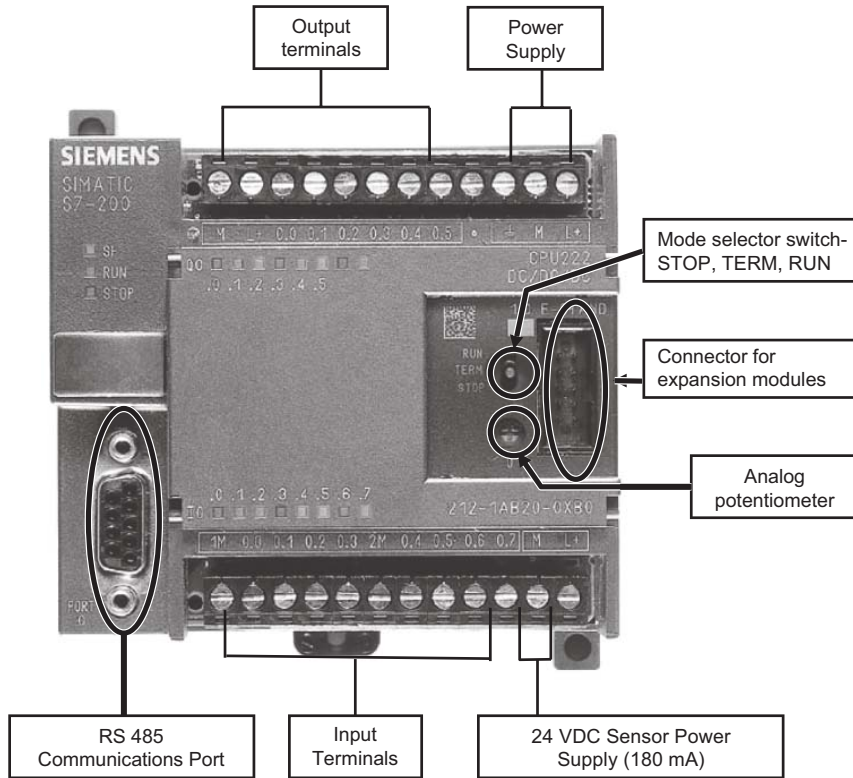


2. Attach the enclosed input simulator to the input terminal block located on the bottom of the S7-200.



3. Attach the assembled S7-200 to the mounting rail.
4. Attach the output simulator to the mounting rail.

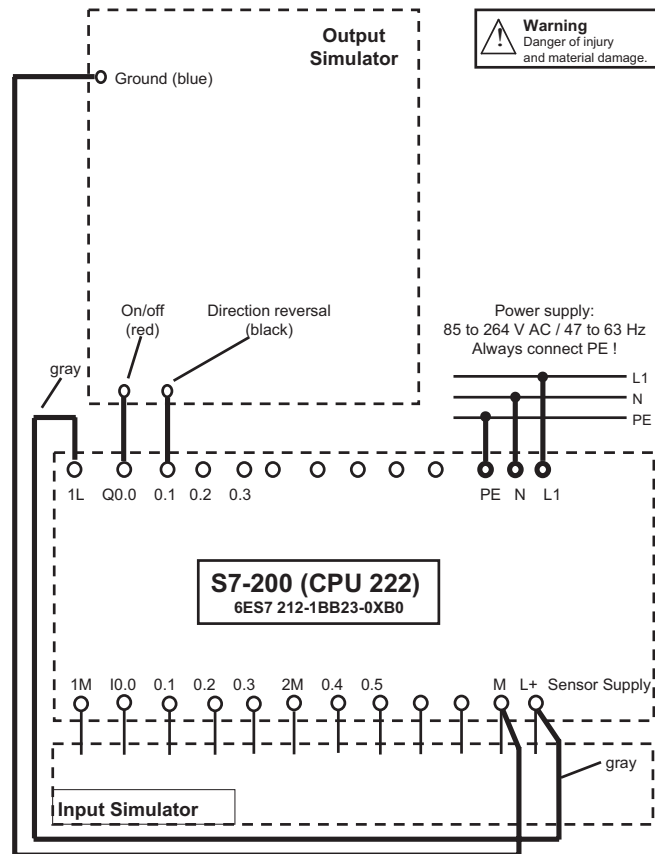
1.2 Details of the S7-200 (CPU 222)



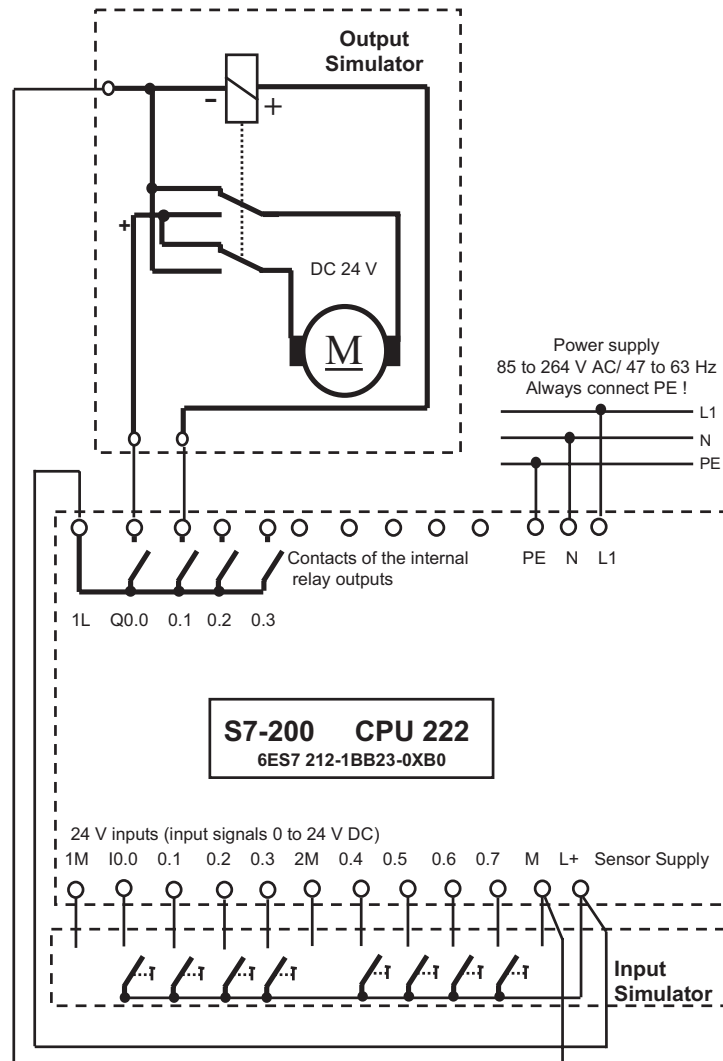
1.3 Wiring diagrams for the assembled hardware

Wiring diagram for the assembled hardware

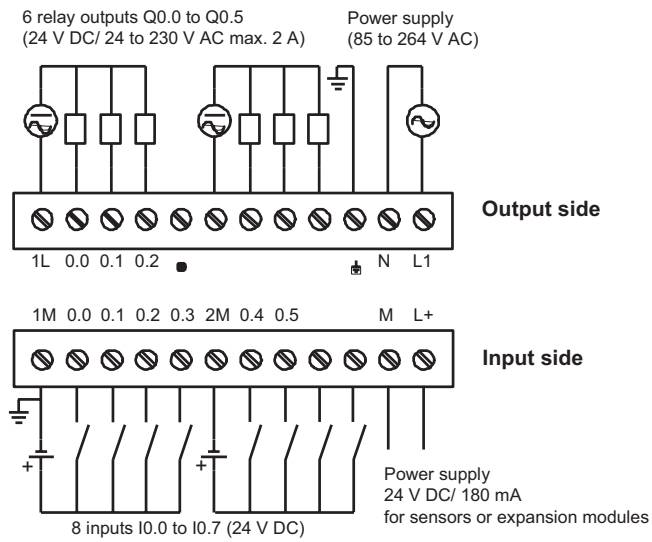
Connect the bold wires as shown in the diagram below. The gray wire is only mechanically attached to the output simulator. Either end of the gray wire can be connected to L+ or 1L.



Circuit diagram for the assembled hardware



Wiring diagram of the S7-200 (CPU 222 AC/DC/RLY))



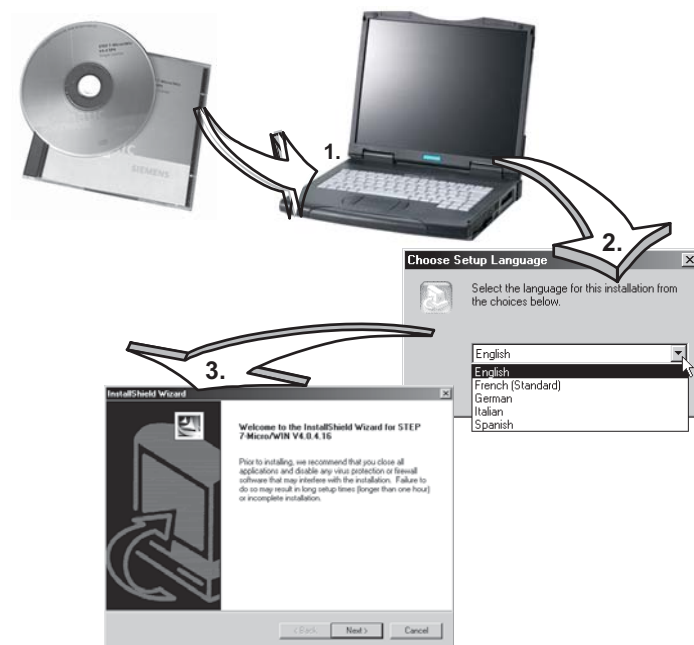
CPU 222 AC/DC/RLY (AC Power/DC inputs/Relay outputs)
6ES7 212-1BB23-0XB0

Installing the software

2.1 Installing STEP 7-Micro/WIN for the One-Hour Primer

In order to install the STEP 7-Micro/WIN V4 programming software you need a PC or programming device (PG) with a Microsoft Operating System. The software will run on Windows 2000 Service Pack 3 or later, Windows XP Home, or Windows XP Professional.

1. Insert the CD into your PC or PG that has Microsoft Operating System running.
2. Select the installation.
3. Start the installation wizard and follow the instructions.

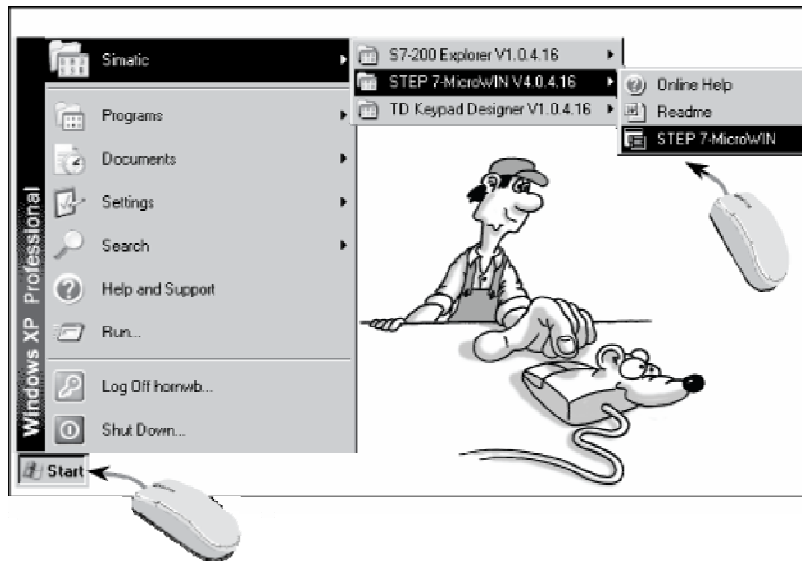


Note: If you already have this STEP 7-Micro/WIN version installed, you are prompted to remove the old software and reboot your system. After the old software is removed, open and close the CD drive to restart the installation process.

At the end of the installation, the entry, "STEP 7-Micro/WIN" appears in the Windows Start menu, **Start > SIMATIC**.

2.2 Starting STEP 7-Micro/WIN

From the Windows Start menu, select the **SIMATIC > STEP 7-Micro/WIN V40.4.xx > STEP 7-Micro/WIN** to start the STEP7-Micro/WIN programming software.



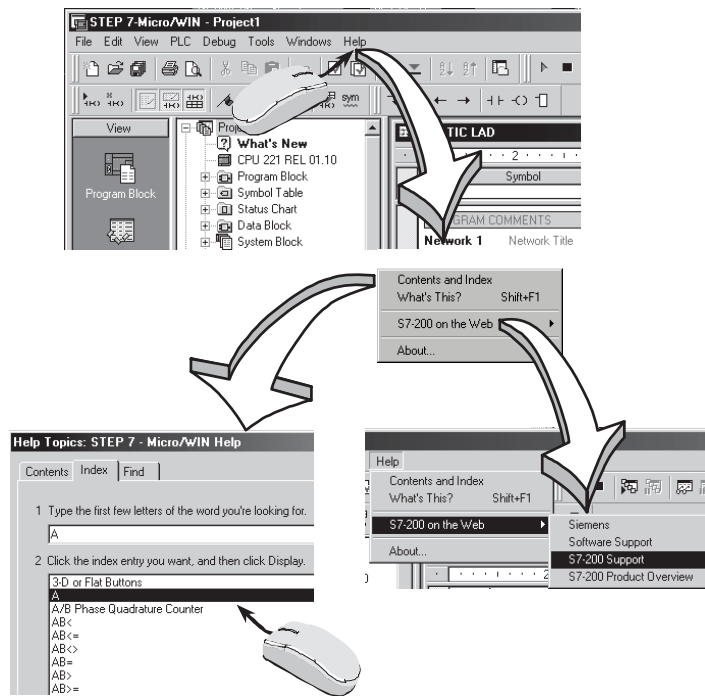
2.3 Using the Help system

The Help window tabs, Contents, Index, and Find (full text search) help you find a topic starting with a view of all Help topics.

Select a menu item, or open a dialog box, for which you want help and press the "F1" key to access the context-sensitive help for that item.

What's This? Help provides definitions of interface elements. You can also access **What's This?** Help by pressing the shift and F1 keys simultaneously. The cursor changes to a question mark; use it to click on the item for which you want help.

If your computer has access to the Internet you can download or retrieve information about SIMATIC S7-200 hardware and software using the **S7-200 on the Web** menu item in the Help menu.



Basic settings

3.1 Connect the communication cable

The PC/PPI Programming Cable (USB/RS 485) connects a PC to the S7-200 PLC.

On your PC, use a standard Universal Serial Bus (USB) peripheral connector. Connect the USB/PPI cable to the PC and the PLC on your PC.

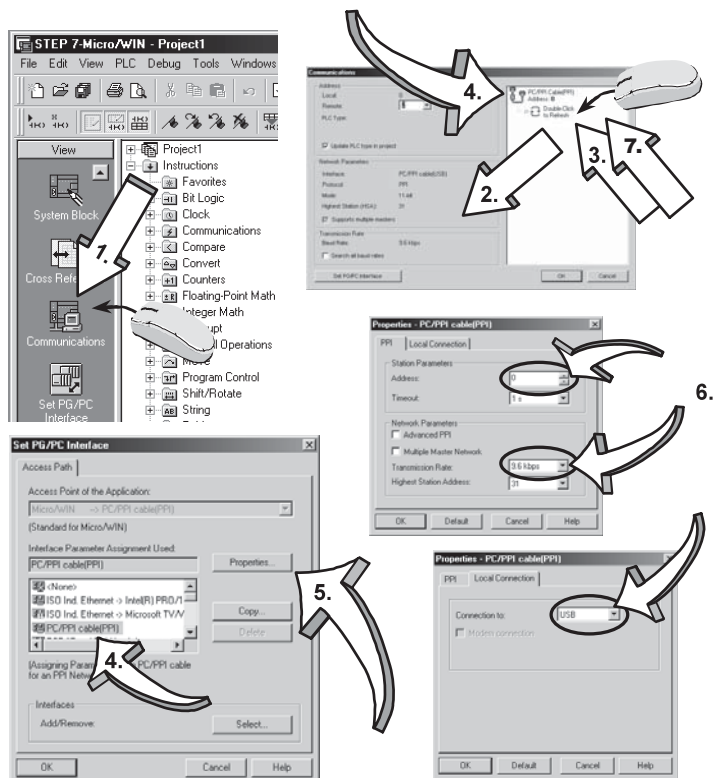
Supply the PLC with the operating voltage.

The STOP or RUN LED should be illuminated.



3.2 Configuring the S7-200 communication

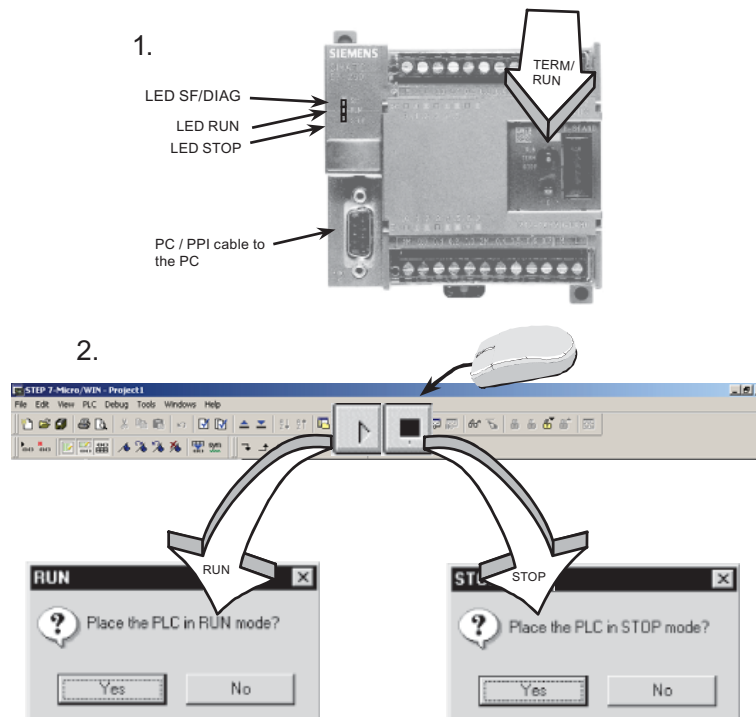
1. Click the communications icon in the navigation bar.
2. Review the communications settings.
3. Double-click the corresponding field to refresh. The connected CPU should now be recognized and registered automatically.
4. If the CPU is not recognized, or if a pop-up window tells you that communication is not possible, double-click the field **PC/PPI cable**.
5. Select **PC/PPI cable** in the **Set PG/PC Interface** dialog and select **Properties**.
6. In the **PPI** tab, set the station address to 0 and set the transmission rate to 9.6 kbps. In the **Local Connection** tab, select the USB port. Click OK to confirm all settings and close the **Properties** dialog. Click OK to close the **Set PG/PC Interface** dialog. Note that the station address cannot be the same number as the CPU address.
7. Double-click the corresponding field in the communications field to refresh. The CPU will now be recognized and registered automatically. The process could take a few seconds. Now click OK to close the **Communications** dialog.



3.3 Performing the first function test

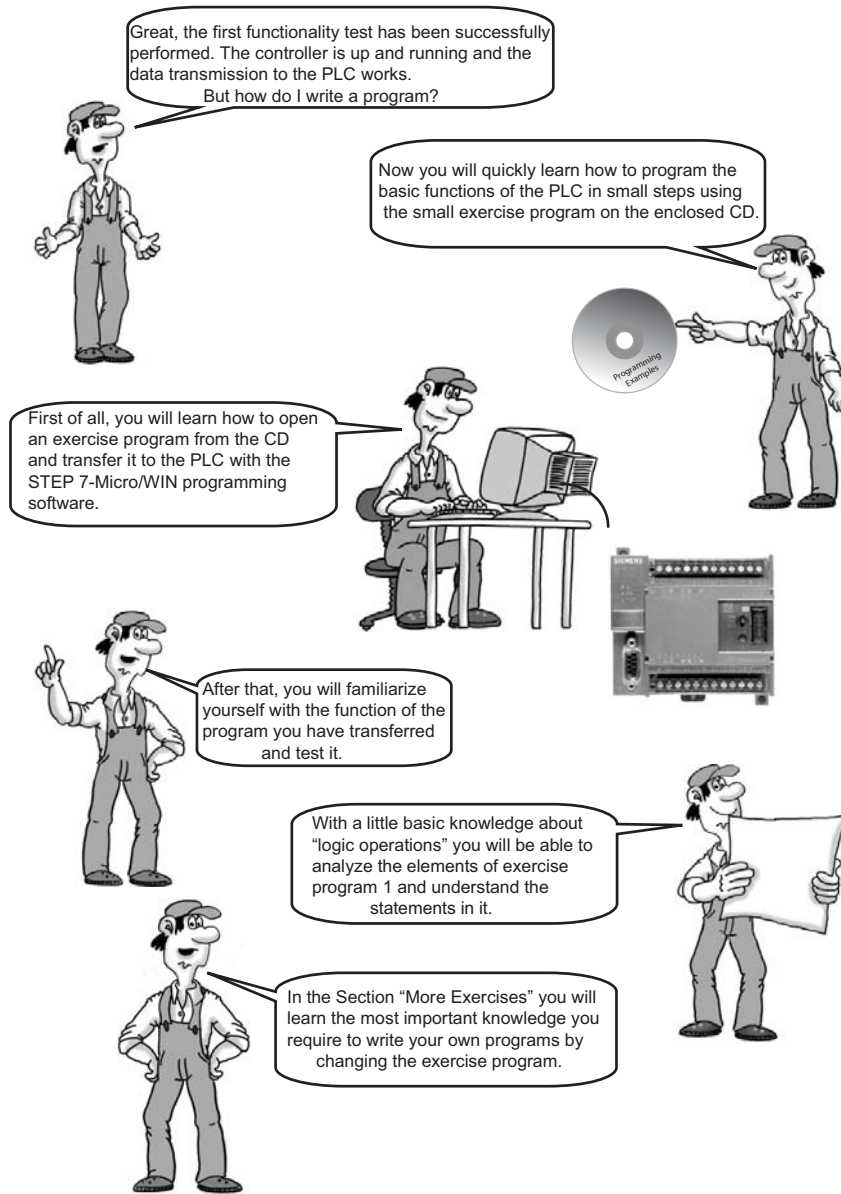
To perform the first function test, follow the steps below:

1. Set the mode selector of the PLC to the TERM or RUN position. The mode selector is located behind the small cover at the front side of the CPU.
Note that remote control of the operating mode (RUN or STOP) is possible only in the TERM or RUN position.
2. From the PC, switch the S7-200 to STOP mode and back to RUN.
The green RUN LED on the PLC lights up in RUN mode. The yellow STOP LED on the PLC lights up in STOP mode. If you can switch the PLC operating modes from the PC, the connection between the PC and PLC is configured correctly.
If the mode does not change, check the connecting cable between the PC and PLC and the communication settings within STEP 7-Micro/WIN.

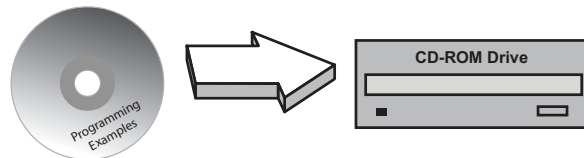


First programming exercise

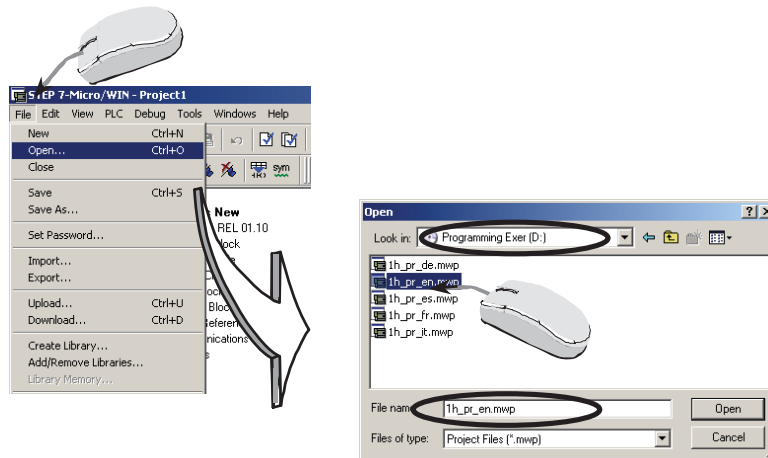
4.1 Writing your first program



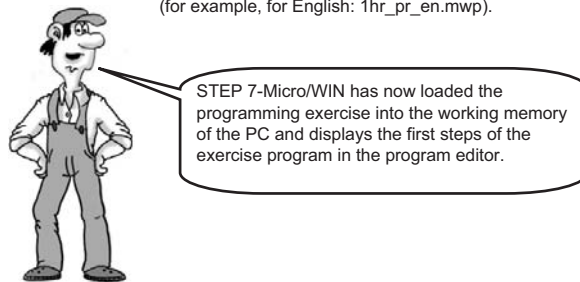
4.2 Opening the first programming exercise



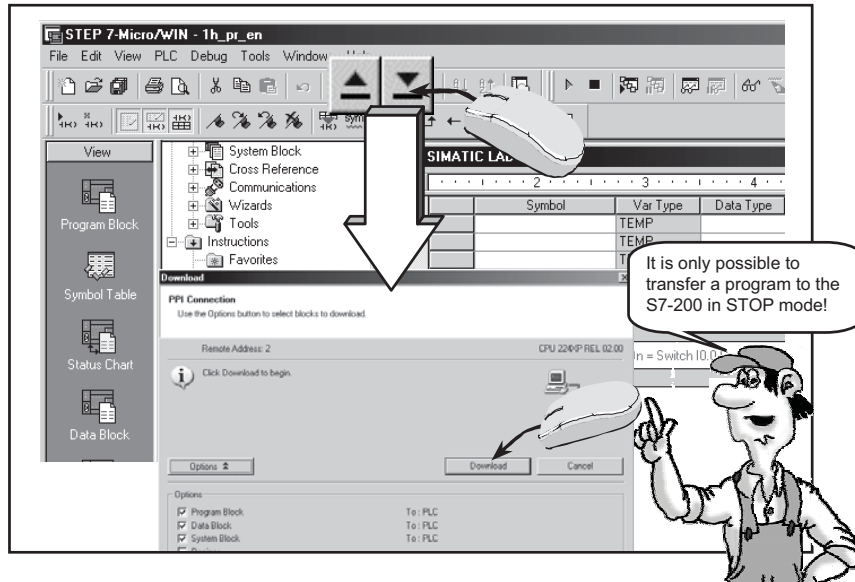
1. Insert the Programming Exercises CD into the CD ROM drive of your PC. This CD is located in the envelope on the last page of this manual.



2. To open the first programming exercise, select **File > Open**. From the Open window, look in the CD ROM and open the STEP 7-Micro/WIN project with the appropriate language (for example, for English: 1hr_pr_en.mwp).



4.3 Downloading the first exercise program to the CPU



Runtime edits are available on the CPU 224, CPU 224XP and CPU 226. Runtime edits allow program edits while in RUN mode.

You can download the program by following these steps:



With the mode selector in TERM or RUN position, click the Download icon. The project is automatically compiled. If there are no errors in the project, a window appears to prompt you to put the PLC back in STOP mode.

Click the OK button in this window to put the PLC in STOP mode. The project is then downloaded to the PLC.

Once the project has been successfully downloaded to the PLC, a window appears to prompt you to put the PLC back in RUN mode. Click the YES button to put the PLC back in RUN mode.

You can also load a program from the PLC onto your programming device/PC.



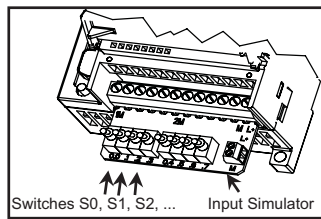
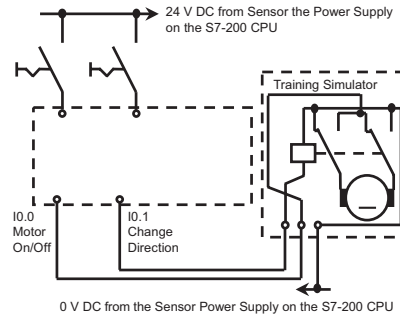
Click the Upload icon to upload the program from the PLC to the programming device/PC. It overwrites the program currently displayed on the monitor. Therefore, you should ensure that you always have a current version of your program on the hard disc, CD or diskette when you leave a plant.

4.4 Function and Test of the first exercise program

In the first exercise program, switch S0 turns the motor on and off. Switch S1 changes the direction of the motor.

Switches S0 and S1 are the first two switches on the input simulator. These switches provide 24 V DC to inputs I0.0 and I0.1 on the CPU. The output simulator is connected to Q0.0 and Q0.1 on the CPU. Q0.0 turns the motor on and off. Q0.1 changes the direction of the motor.

With the exercise program, the signal state of I0.0 is assigned to output Q0.0 and the signal state of I0.1 is assigned to Q0.1



Now let's test it!

The operating voltage is connected to the CPU. The assembly is wired correctly. You have opened the program exercise from the CD and downloaded it to the CPU. The CPU is in RUN mode (the green RUN LED is on). Now operate switches S0 and S1 and monitor the results.










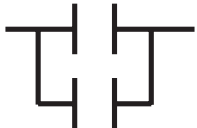
Actions	Reactions		
Switch S0 operated	LED I0.0 is on	LED Q0.0 is on	Motor turns
Switch S0 & S1 operated	LEDs I0.0 & I0.1 are on	LEDs Q0.0 & Q0.1 are on	Motors turns in the opposite direction



LEDs I0.0 to I0.7 indicate the signal states of inputs I0.0 to I0.7.
LEDs Q0.0 to Q0.5 indicate the signal states of outputs Q0.0 to Q0.5.
I and Q are the customary international symbols for inputs and outputs (I0.0 Q0.0).

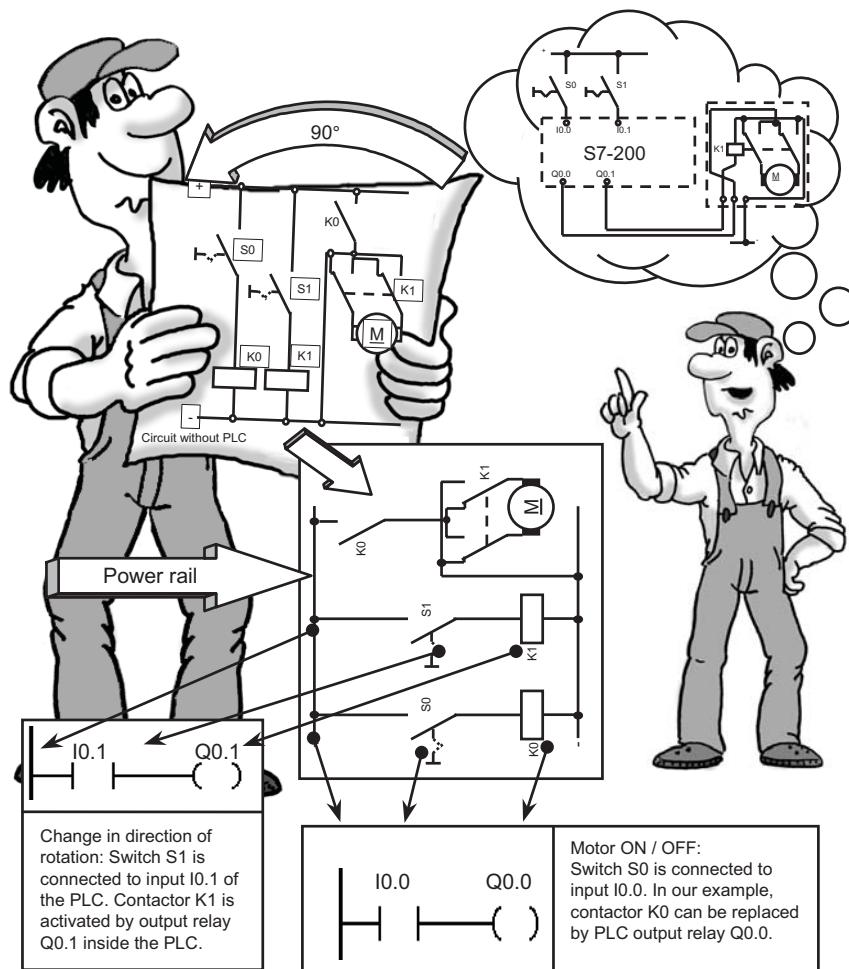
4.5 Elements of ladder logic (LAD)

0 and 1 are the only data states in digital control logic. The 0 state is designated as false and the 1 state is true. This is why we say a PLC scan is either 0 (false) or 1 (true).

Schematic diagram elements	PLC scan logic elements	PLC program ladder diagram elements
	Positive logic Input sensor scan Is current flowing? If yes, then the result of the scan is true (Result is 1)	
	Negative logic Input sensor scan: Is <u>no</u> current flowing? If yes (no current), then the result of the scan is true (Result is 0)	
	Output Coil: If the value "true" (current) is passed to a coil it is activated (The coil is ON)	
	Series circuit connection (AND logic) The first switch AND the second switch must be closed in order to pass current	
	Parallel circuit connection (OR logic) The first switch OR the second switch must be closed in order to pass current	

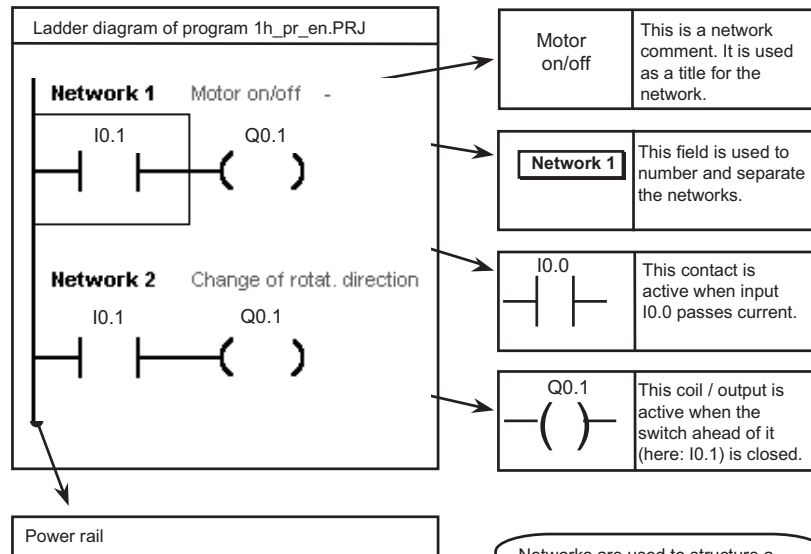
4.6 Transforming a circuit diagram

How do you transform a circuit diagram into a PLC program? Rotate your circuit diagram 90° to the left. Your power rail will then appear on the left, with the grounding rail on the right. In the middle you will see the switching elements of your circuit. The circuit logic of a machine (e.g. time relays or flip-flops) which used to be implemented by wiring together switches, auxiliary contactors and control contactors, etc., is now handled by the PLC. Control elements such as input switches, selector switches, etc. on the input side, and power contactors (such as motor contactors, polarity reversers, valves, etc.) on the output side cannot be replaced by the PLC.

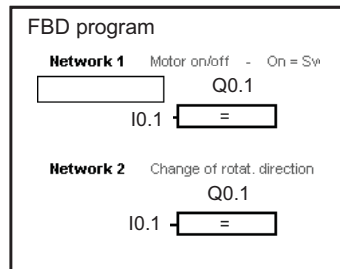
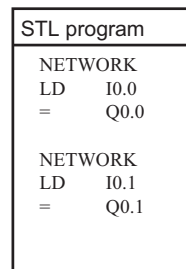


4.7 Elements of the first exercise program

Let's have a closer look at the structure of the PLC program in a ladder diagram (LAD). This type of representation closely resembles a circuit diagram.



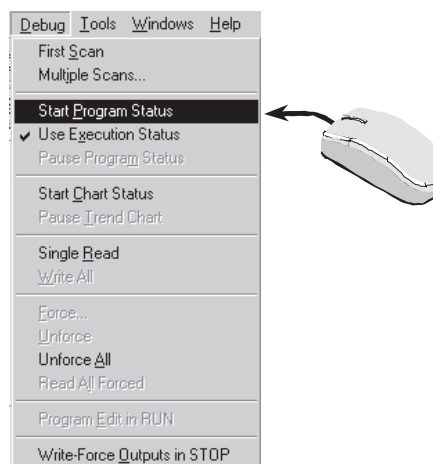
Networks are used to structure a program. You insert a current path into each network.



4.8 Status view (online)

Select the menu item **Debug > Start Program Status** to activate the status view of the ladder diagram.

You can now view the status of the operands in the PLC.



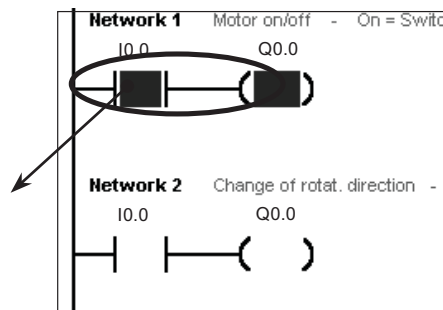
In the example, switch S0 is connected to input I0.0.

If you now actuate the switch and observe the program in ladder status view, the operations which are passing current (true) are filled in blue.

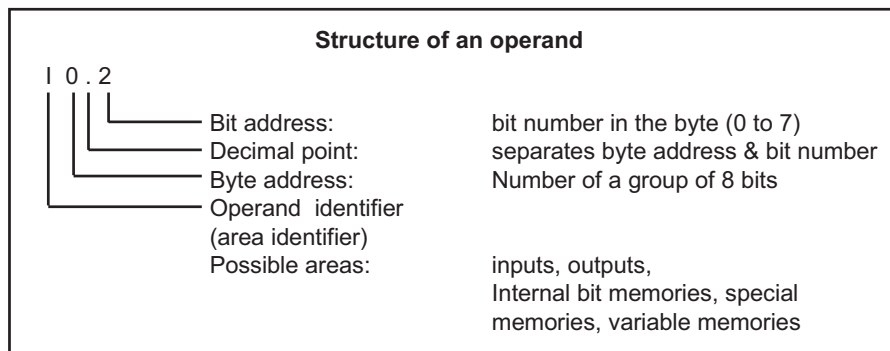
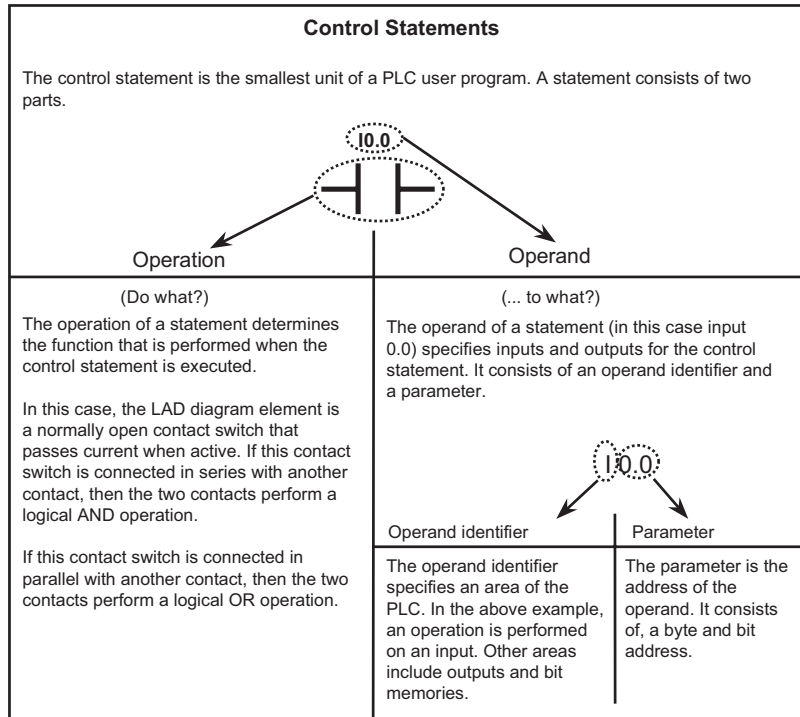
The status of the operations is read cyclically from the PLC and updated on the screen.

In this context, online means looking inside the controller with a PC/programming device to view the current states and their cyclic changes.

Very fast events cannot be tracked in this way because the time required to transfer the data to the screen is too long. A status view is also possible for a function block diagram (FBD).



4.9 Statements



More exercises

5.1 First program modification

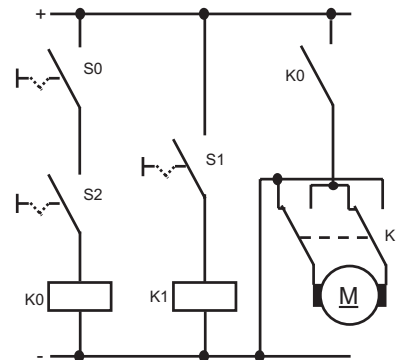
5.1.1 AND logic operation

Objective: Switch S2 and switch S0 must be closed to switch on the fan motor.

As before, switch S1 is closed to reverse the direction of rotation of the fan motor.

A description of the function shown above:

The motor runs when S0 AND S2 are closed. In ladder diagram that means: when contacts I0.0 AND I0.2 are closed, current flows from the power rail to coil Q0.0.

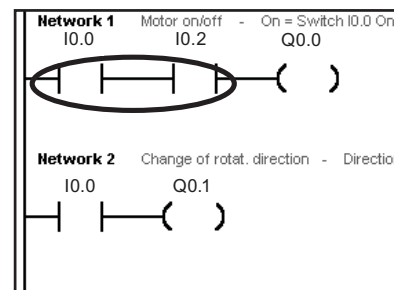


Circuit without PLC

The contacts are connected in series (AND logic). In the ladder program, the logic is expressed as follows:



We do not need to wire switch S2, since it is already connected to input I0.2 via the simulator. The next page describes how to select, insert, and delete a logic gate, and how to name the operands.



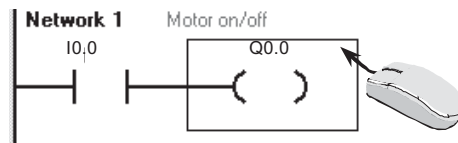
Ladder diagram of the circuit

5.1.2 Inserting a logic gate

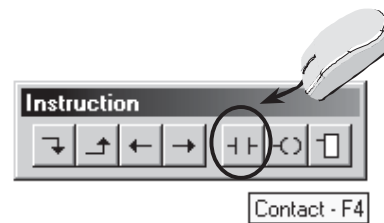
If you want to connect a further normally-open contact for input I0.2 in series between normally-open contact I0.0 and coil Q0.0 (i.e. insert an AND link between I0.0 and I0.2), you first need to select a valid location to put the new contact.

Click the next element to the right of the insert point. In this case, put the selector box on coil Q0.0.

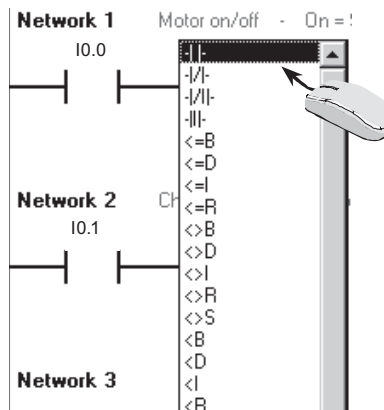
The insert point will be to the left of the selector box.



Click the contact button in the tool bar or press the corresponding function key.



Select the type of contact from the drop list. Choose top item of the list. This is the normally-open contact.




5.1.3 Entering the operand and testing

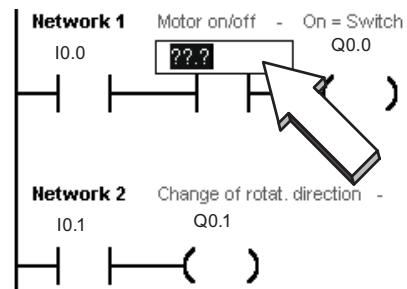
After you insert the new contact, you must specify the correct operand.

Click the operand field and then input the operand: I0.2.

Press the Enter key to confirm the input.

Don't forget to save your changes! 

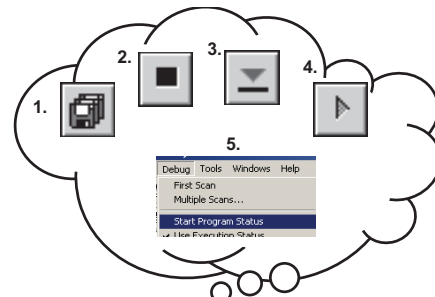
Click the field with the mouse at any time to select it again.



If you want to test a modification:

Download the modifications to the PLC and test the program. Actuate switches S0 and S2 to start the motor.

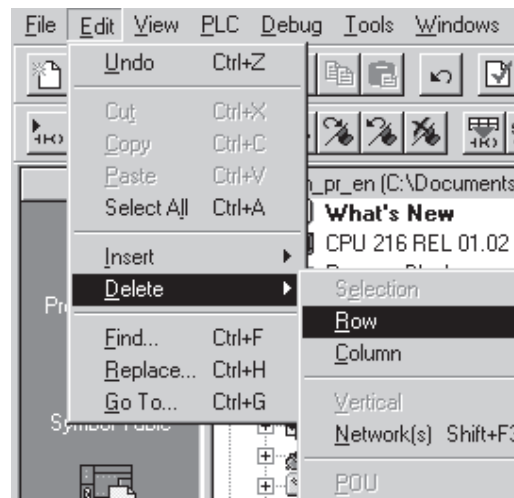
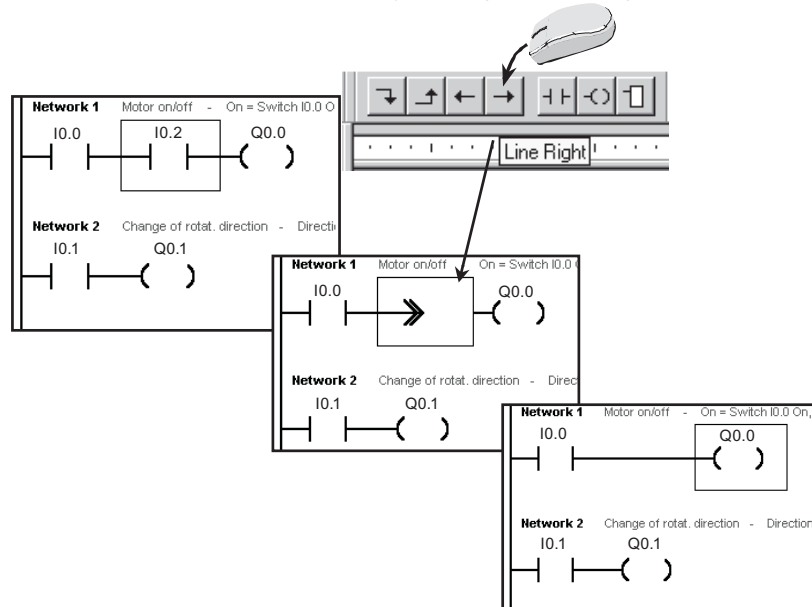
View your program in ladder status mode and observe the switch state change displayed in the ladder diagram.



5.1.4 Deleting a contact or an operand

If you want to delete the I0.2 contact, select the contact with the mouse and press the DEL key.

You must reconnect I0.0 to Q0.0 by clicking the Line Right button.



If you want to delete a selection, row, column, vertical, network(s) or POU proceed as follows:

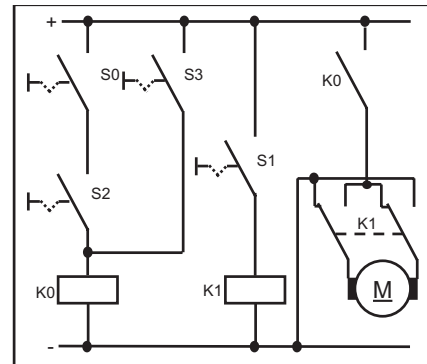
Select the desired object.

Open the Delete menu by selecting **Edit > Delete**.

5.2 Second modification to the program

5.2.1 OR logic operation

Objective:
 Switches S0 and S2 are to be actuated in order to switch on the motor. Switch S3 alone is to be used as an alternative in order to switch on the motor. Switch S1 is to be used to reverse the direction of rotation of the motor.

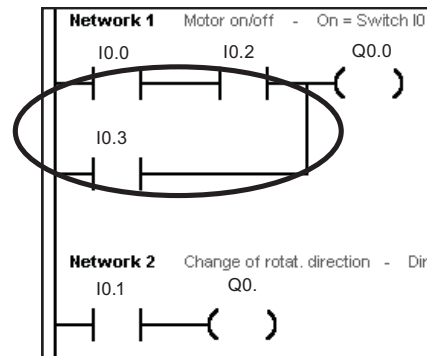


Circuit without PLC

A description of the function shown above:
 When (S0 AND S2) OR S3 are actuated, the motor runs. In the ladder diagram that means: when contacts I0.0 AND I0.2 or I0.3 are closed, current flows from the power rail to coil Q0.0. This is a parallel connection from S0 and S2 to S3 (OR logic operation). In our program, the logic is expressed as follows:



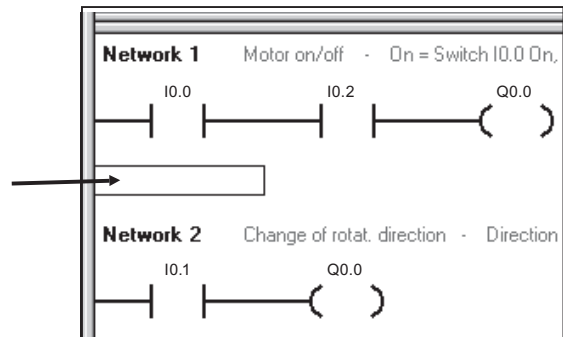
We do not need to wire switch S3, since it is already connected to input I0.3 via the simulator. The next page describes how to select and insert an OR logic gate with a connecting line.



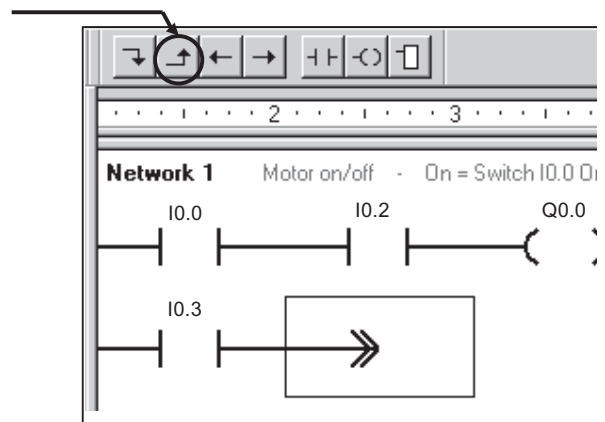
Ladder diagram of the circuit

5.2.2 Inserting an OR element

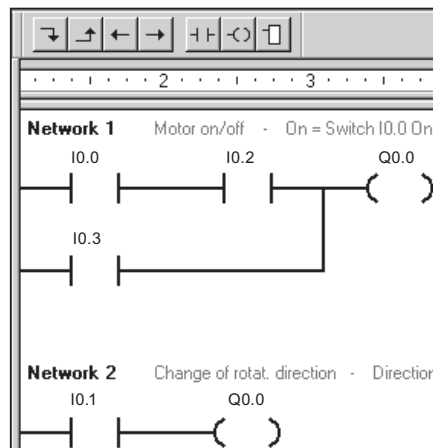
Use the mouse to select a free position in the same network and insert a normally-open contact at this location.



Click the mouse on the Line Up button in the LAD toolbar



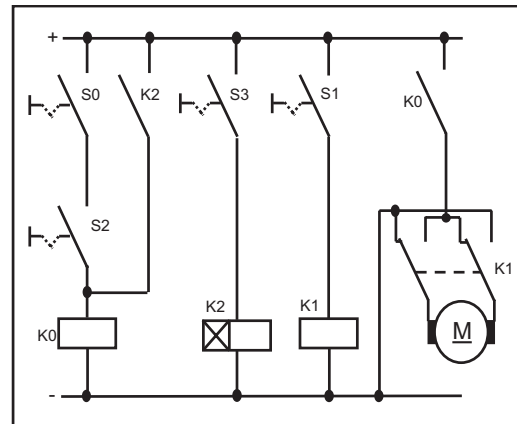
Now the parallel OR branch is complete.



5.3 Third modification to the program

5.3.1 On-delay timer

Objective:
 In the following program modification, you insert an on-delay timer in exercise program 1.
 When input I0.3 (S3) is activated on the simulator, a waiting time is started. Output Q0.0, and thus the motor, is not activated until the waiting time has expired.

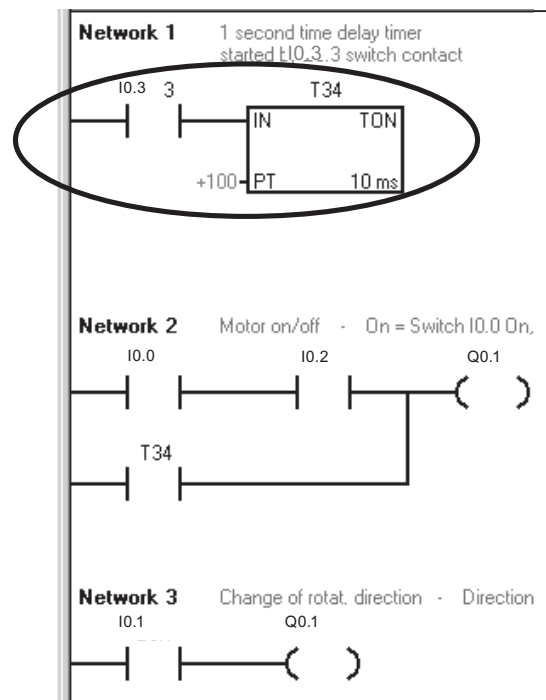


Circuit without PLC

No wiring work on the PLC is required for the additional timer function.

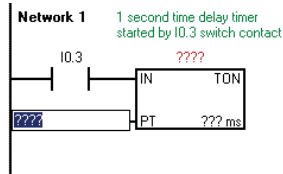
All necessary switches and actuators are already wired. The timer relay is replaced by a PLC timer function.

The following page introduces the on-delay timer function of the S7-200. We will add a time delay network and create this three network program.



Ladder diagram of the circuit

5.3.2 Understanding the on-delay timer function



You require an on-delay of 1s. The S7-200 CPU222 has 256 timers. The timers have the designations T0 to T255. Use T34 in this example.

Before the timer will run, it must be started. In our example, this is done with I0.3 at the input "IN".



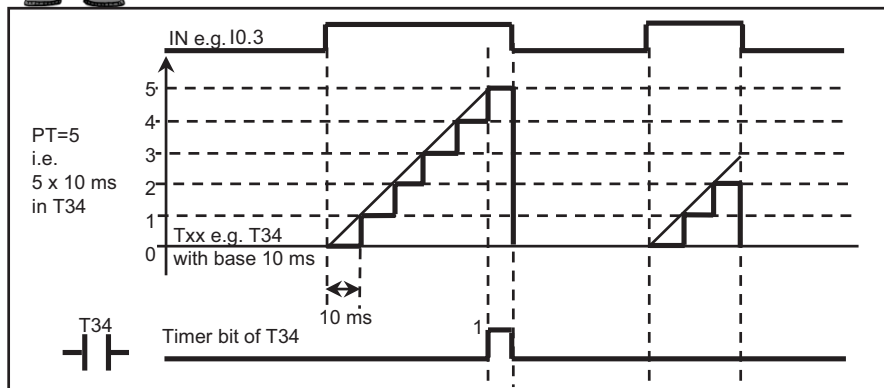
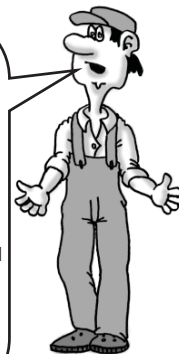
Set the time value of 1s by writing the value 100 to PT. The time value is calculated from the factor PT (here 100) and the timebase of the timer. (T34 has a timebase of 10 ms, $100 \times 10 \text{ ms} = 1 \text{ s}$).

Timebase	Txx
1 ms	T0, T32, T64, T96
10 ms	T1-T4, T33-T36, T65-68, T97-T100
100 ms	T5-T31, T37-T63, T69-T95, T101-T255



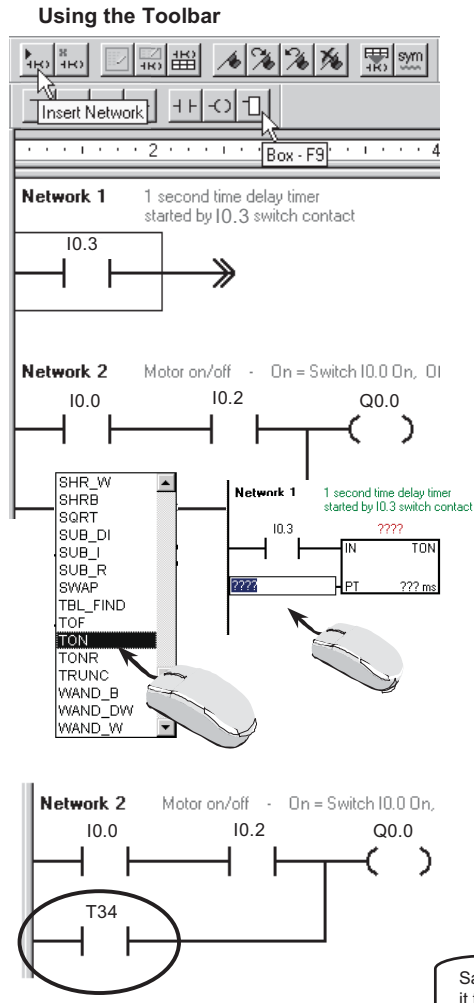
For CPU 222, timers with timebases of 1 ms, 10 ms and 100 ms are available. Timers can only be used once.

Every timer in the S7-200 has a status bit (time elapsed/not elapsed). This bit is called the timer bit. For the timer T34, it is called T34. Input I0.3 must have the value "1", that is 1s, before timer bit T34 is "1". If the delay has not elapsed when the enable signal "IN" is canceled, the timer is set to "0" and the timer bit is not set (see the timing chart below). You can use the timer bit like contact operands.



Timing chart for a time value of 50 ms

5.3.3 Programming the on-delay timer



1. To evaluate a timer contact inserted in our OR logic network, you must create that timer box in a previous network. Select a part of the first network with a mouse click. Then, click the **Insert Network-F3** toolbar button.

2. Insert a normally-open contact for input I0.3 in the first column of the new network 1. Then put a timer in the second column as a on-delay timer type. Select the **Box-F9** toolbar button.

3. Then, put a timer in the second column as a on-delay timer box. Select the **Box-F9** toolbar button. Scroll down until the function TON (on-delay timer) appears and select it by clicking the mouse.

4. Select the fields for PT and TON and input your values, in this case 100 and T34.

5. In the last step, the timer bit T34 must now appear in network 2 in the position of I0.3. Make the correction.

Save the program and download it to the PLC and test its function.



5.4 Working with projects

5.4.1 Programming with symbols

So far you have been working in the PLC program with operands in the "PLC language" such as I0.3 or T34, but if the program is longer, it is not easy to associate these operands with their actual function within a control system. The program would be easier to read if we could switch the operand names to plain text. This is exactly what you can do with symbolic programming.

1. For symbolic addressing, the symbol table must be filled in. Select the symbol table from the Navigation bar, Instruction tree, or select **View > Component > Symbol table**.

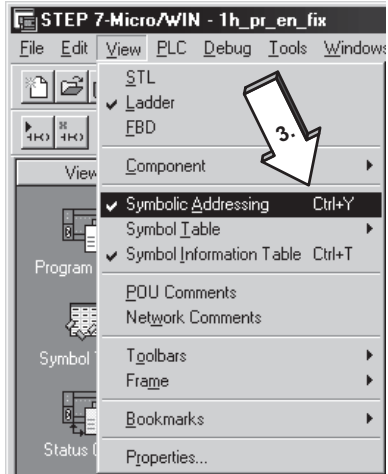
The screenshot shows the STEP 7-Micro/WIN interface. The 'View' menu is open, showing 'Component' > 'Symbol Table'. The Symbol Table window is displayed at the bottom, containing the following data:

	Symbol	Address	Comment
1	S0	I0.0	Switch 0
2	S1	I0.1	Switch 1
3	S2	I0.2	Switch 2
4	S3	I0.3	Switch 3
5	Motor_ON	Q0.0	Turns motor ON
6	Direction	Q0.1	Changes the direction of the motor
7	Delay_1s	T34	Waits 100 x 10ms = 1 second
8			
9			

2. A window is displayed in which you can edit the symbol table. Enter the name that you want to appear as plain text later on under "Symbol". Under "Address", enter the operand that you want to have associated with the symbolic name. Under "Comment", you can enter any text that you find helpful. Don't forget to save your work.

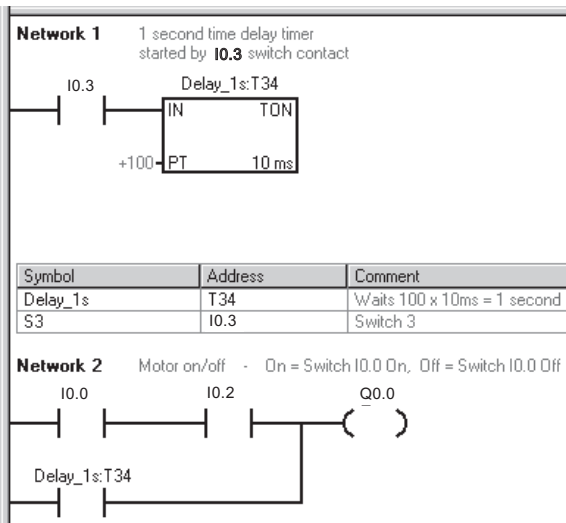


Using a symbol in a network



3. Select the menu item **View > Symbolic Addressing** to switch the display mode. If the **Symbol Information table** entry is also checked, you can see a symbol summary table at the bottom of each network.

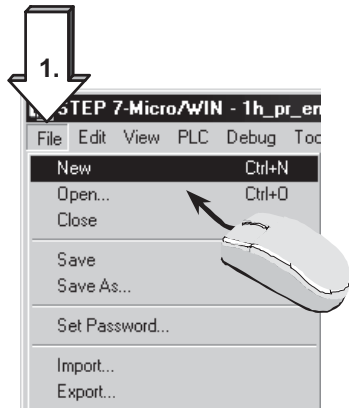
4. If you have selected symbolic addressing and changed to LAD, STL or FBD, the symbolic addresses should now be visible. You can also switch the Symbolic Addressing checkmark off and on again to refresh the program editor window.



5. The figure above shows the ladder diagram of the user program with symbolic addressing. You can, of course, also program with symbolic addresses. As the operand, you must then enter "S3", for example, you only have to make sure that this operand has already been entered in the symbol table.

5.4.2 Creating a new project

If you want to write a new program of your own, you need a sort of container to put your program file in. In STEP 7-Micro/WIN, this container is the project. An S7-200 project contains all additional information for your project, for example, the symbol table and comments etc.

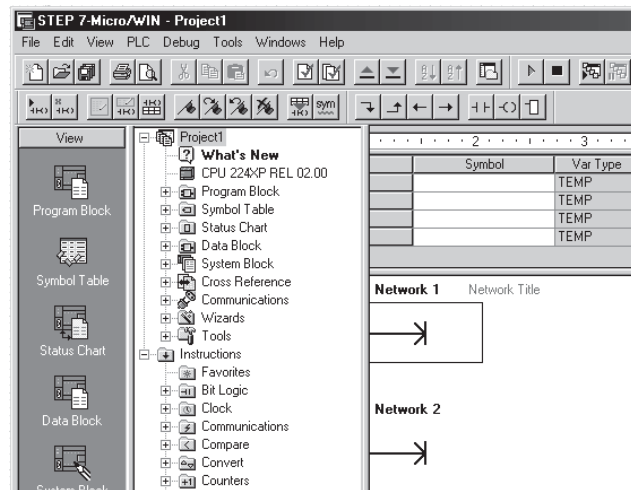


1. Create a new project with menu command **File > New** .

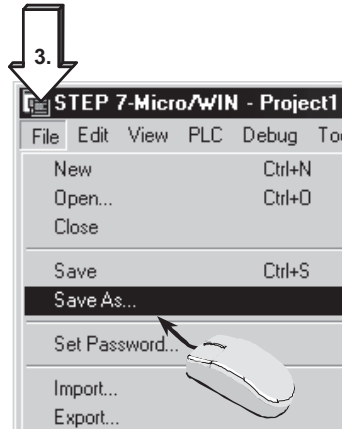
2. The result is a project with the name **Project1**. You can start programming now.



See the next page for more.

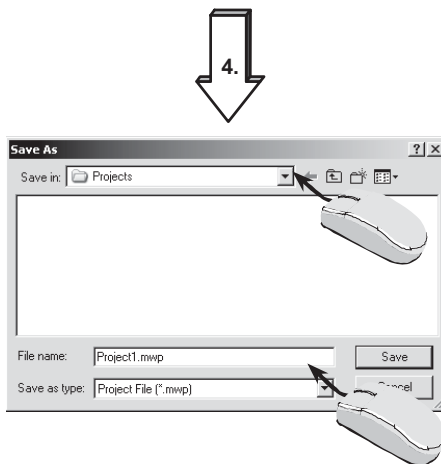


Save the project under a new name



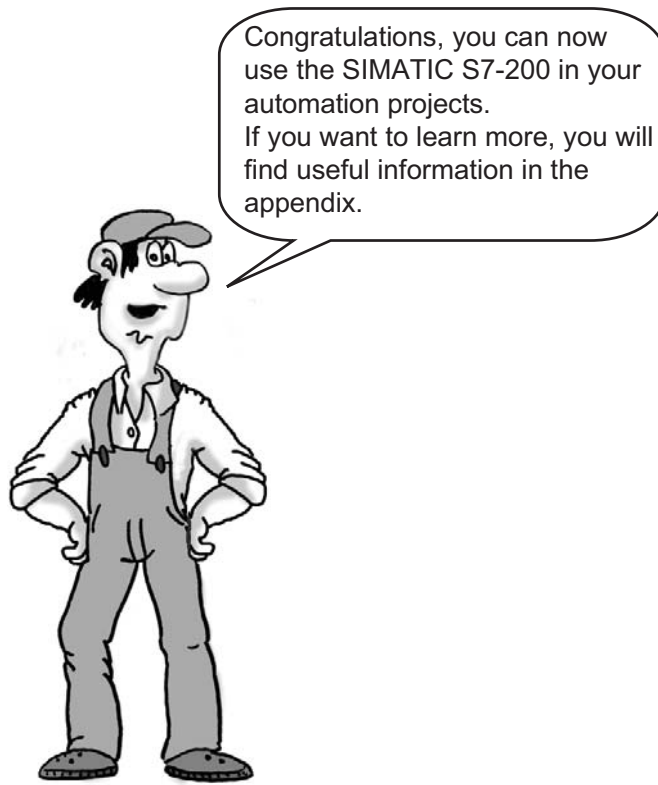
3. Save Project1 immediately, after, or during programming under a name that means something to you.

Select the menu **File > Save As** to open the dialog box.



4. In this box, give a name to your project and select the hard disk drive and the path/folder where you want to store your project. Terminate your input with **Save**.





5.5 Do you want to learn more?

You will find more examples in the "Samples" directory in your STEP 7-Micro/WIN directory.

The examples of the "One Hour Primer" are continued in the "Two Hour Primer".

In addition, a CD ROM with "Tips & Tricks" for the S7-200 is available. You can obtain the "Two Hour Primer" and the "Tips & Tricks" from your SIMATIC representative.

You will find more information in the manuals for the S7-200.

For further training you can attend a S7-200 course at your Siemens Training Center or with your SIMATIC representative.

Unanswered questions?
Technical problems?
The SIMATIC representatives will be glad
to help.



Please contact the SIMATIC representative from whom you purchased the Starter Kit. If you cannot contact your SIMATIC representative, please call the Siemens Technical Support at:

Technical Support:

Worldwide (Nuermberg): 49 (180) 5050-222

United States (Johnson City): 1 (423) 262-2522, 1 (800) 333-7421 (USA only)

Asia/Australia (Beijing): 86 10 64 75 75 75

Appendix

A

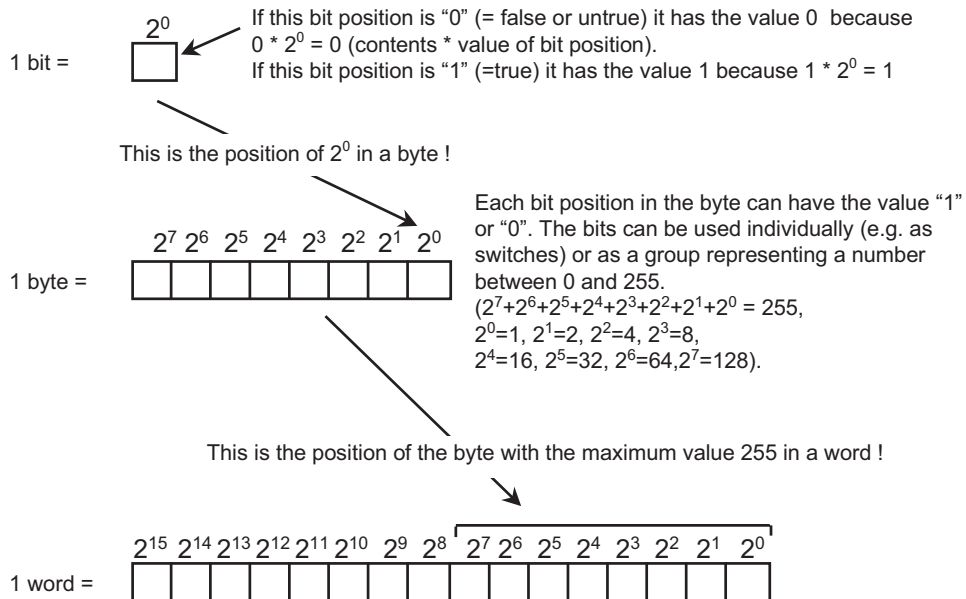
A.1 Of bits, bytes and words ...

The smallest unit of information in a digital system is known as a "bit". A bit can only have the states "0" (false or not true) or "1" (true).

A light switch, for example, only has the states "light on" or "light off". The value of the light switch in answer to the question "Is the light on?" is either true (the light is switched on) or false (the light is switched off). The light switch thus has an information width of one bit. The state "light switch on but bulb defective" is ignored in this example.

In a PLC, bits are organized into groups. A group of 8 bits is called a byte. Each bit in the group is defined exactly by a separate position with its own address. Each bit has a byte address and bit addresses 0 to 7. A group of 2 bytes is called a word. A group of 4 bytes is called a double word

The numbers in this logical system are binary numbers and they are counted with the base 2 number system.

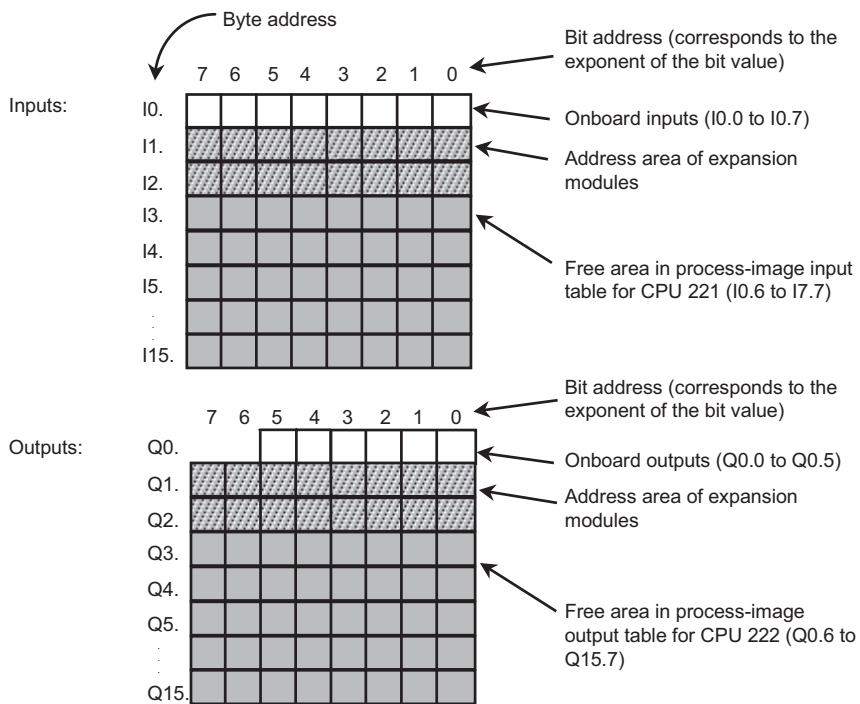


A word in a PLC can represent an integer number from -32768 to +32767. The bit with the value 2^{15} is used to denote a negative number (when position 2^{15} has the value "1" the number is negative).

A.2 Address areas of the S7-200

The diagram below shows the memory addresses of the S7-200 PLC where inputs and outputs are mapped. For example, if a voltage is applied to a physical input, this "1" signal is mapped at an address in the memory that is assigned to this input. The memory areas which form the interface with the screw terminals (inputs and outputs) of the PLC are called the "process image of the inputs" (PII) and the "process image of the outputs" (PIO). These memory areas contain an image of the signals from your control process.

Address areas of an S7-200 PLC with CPU 222



A.3 Cyclic program execution in the S7-200 PLC

All SIMATIC PLCs operate a scan cycle. During each cycle, the external switch states are first read in from the inputs, and then stored in the process-image input table. The control program is then executed on the basis of the image values. The outputs in the process-image output table are subsequently controlled in accordance with the circuit logic of the ladder program. The last step is to transfer the new output states from the process-image output table to the physical outputs. This execution cycle is repeated endlessly in RUN mode.

