Training document for the company-wide automation solution

Totally Integrated Automation (T I A)

MODULE B6

Conversion STEP 5 => STEP 7

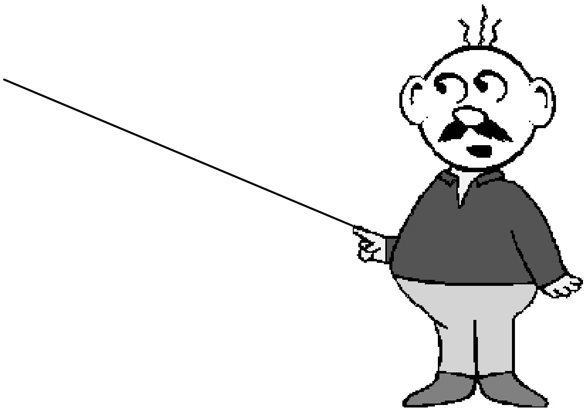
We thank the company Michael Dziallas Engineering and the instructors of vocational schools as well as further persons for the support with the production of the document.

The following symbols stand for the specified modules:

ℹ **Information**

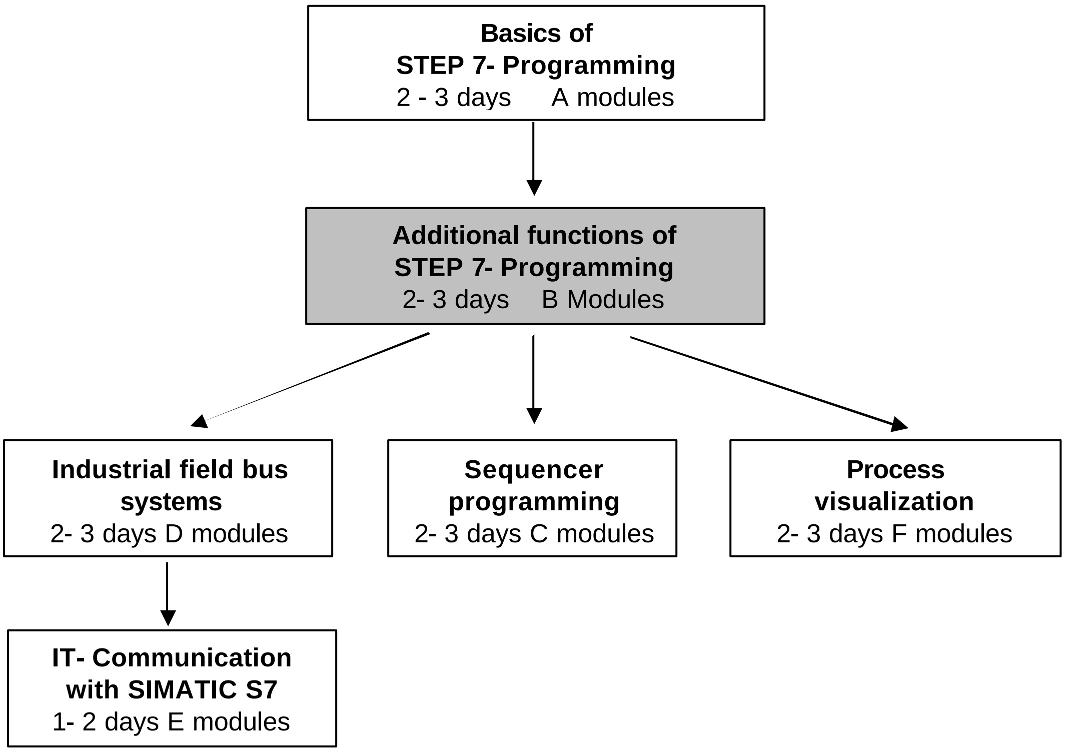
🖥 **Programming**

⚠ **Notes**



1. FORWARD

The module B6 is assigned content wise to **Additional functions of STEP 7- Programming**.



Basics of STEP 7- Programming
2 - 3 days A modules

Additional functions of STEP 7- Programming
2- 3 days B Modules

Industrial field bus systems
2- 3 days D modules

Sequencer programming
2- 3 days C modules

Process visualization
2- 3 days F modules

IT- Communication with SIMATIC S7
1- 2 days E modules

Learning goal:

In this module, the reader should learn how to convert STEP 5- Programs into executing STEP 7- Programs

Requirements:

For the successful use of this module, the following knowledge is assumed:

- Knowledge in the use of Windows 95/98/2000/ME/NT4.0
- Basics of PLC- Programming with STEP 7 (e.g. Module A3 - 'Startup' PLC programming with STEP 7)

Required hardware and software

1 PC, Operating system Windows 95/98/2000/ME/NT4.0 with

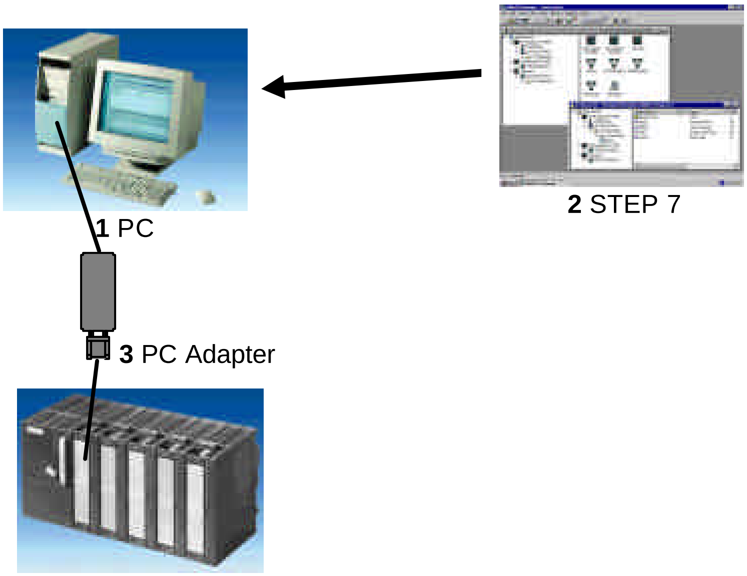
- Minimal: 133MHz and 64MB RAM, approx. 65 MB free hard disk space
- Optimal: 500MHz and 128MB RAM, approx. 65 MB free hard disk space

2 Software STEP 7 V 5.x

3 MPI- Interface for the PC (e.g. PC- Adapter)

4 PLC SIMATIC S7-300 with at least one digital in- and output module. The inputs must be lead through a functional unit.
Example configuration:

- Power supply: PS 307 2A
- CPU: CPU 314
- Digital input: DI 16x DC24V
- Digital output: DO 16x DC24V / 0.5 A



2. NOTES FOR CONVERSION STEP 5 => STEP 7

By conversion it is to be noted that it is not possible to convert all programs.
For example, there are many standard functions in STEP 7 that are no longer in a form like STEP 5.

Simplifications are mostly given here so that these functions are simply no longer required or were replaced with simple parameterization in a configurations tool.

This almost always applies when a function module is used (e.g. FM for step motor triggers).

In this case, these blocks and their calls should be deleted before the conversion from the STEP 5-Program.

After the conversion of the remaining program lines, this function must be taken and programmed with the mean of STEP 7 in operation.

Programs that are created with the normal instruction set from STEP 5, always allows you to fully convert.

Mostly only the addressing must be adjusted.

3. S5- DATA CONVERSION

The conversion from S5- Data, that would be generated with STEP 5, takes place with the tool S5 data conversion.

The following steps must be followed by the user in order to convert S5- Data:

1. For the processing of the conversion, you must provide the follow data in a directory.

-	<Name>ST.S5D	Program data (the S5-Program to be converted)

It is by complex programs that the calling sequence of the blocks remain preserved. The following additional data is important:

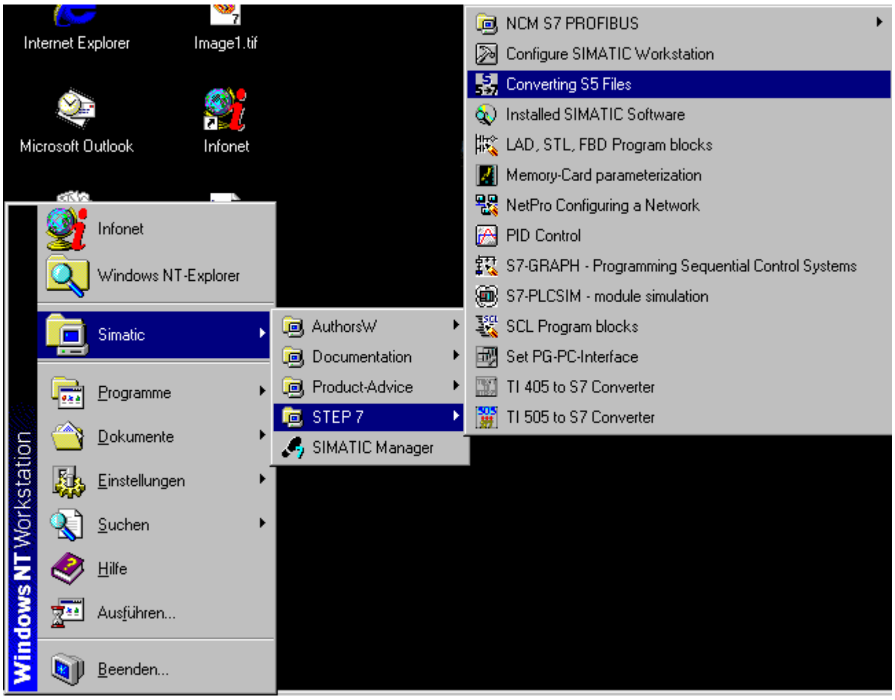
-	<Name>XR.INI	Cross reference list (contains the program structure)

If you would also like to convert the symbol data, the additional data is required:

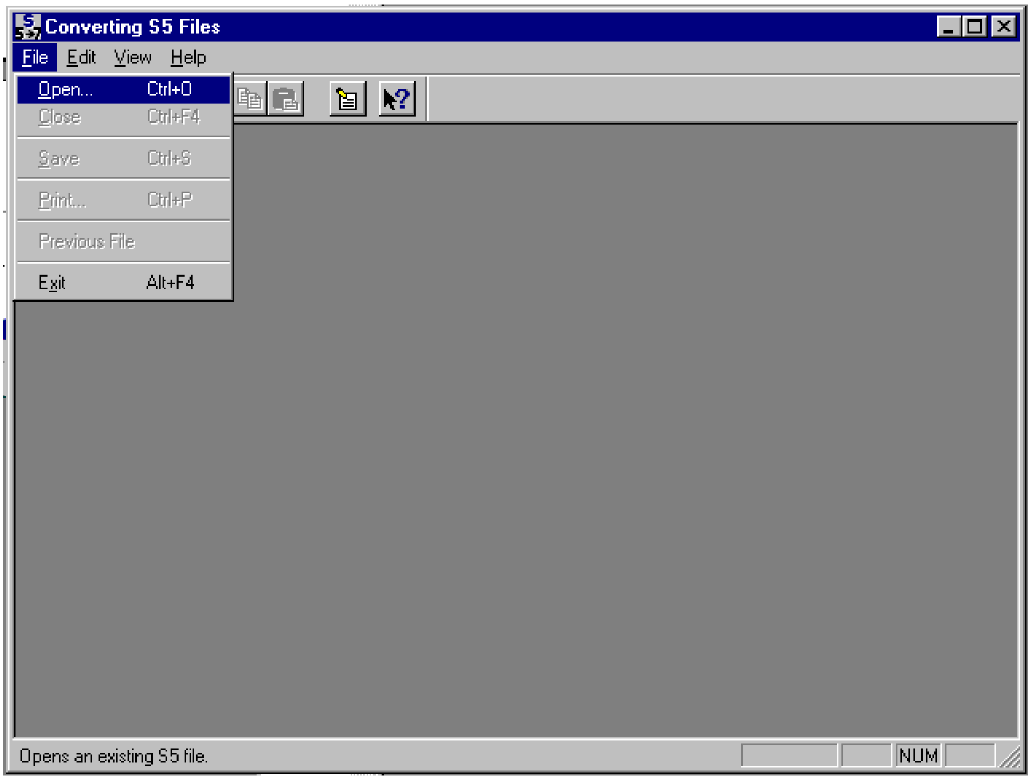
-	<Name>Z0.SEQ	symbol table

All newly created data by the conversion is deposited into the same directory.

2. Call conversion tool **Converting S5 Files** (→ START → Simatic → STEP 7 → Converting S5 Files).

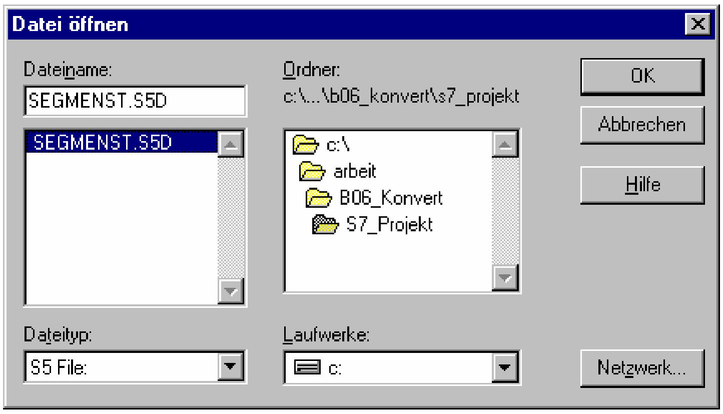


3. In the tool **Converting S5 Files**, **open** the S5-Program data (→ File → Open)



Opens an existing S5 file.

4. The S5-Program data with the ending ***ST.S5D** is chosen. (→ segmenST.S5T → OK)



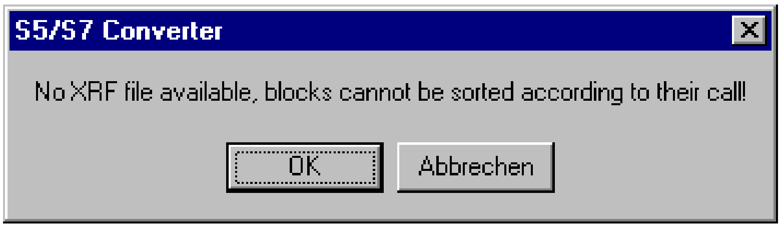
Dateiname: SEGMENST.S5D

Ordner: c:\...\b06_konvert\s7_projekt

Dateityp: S5 File:

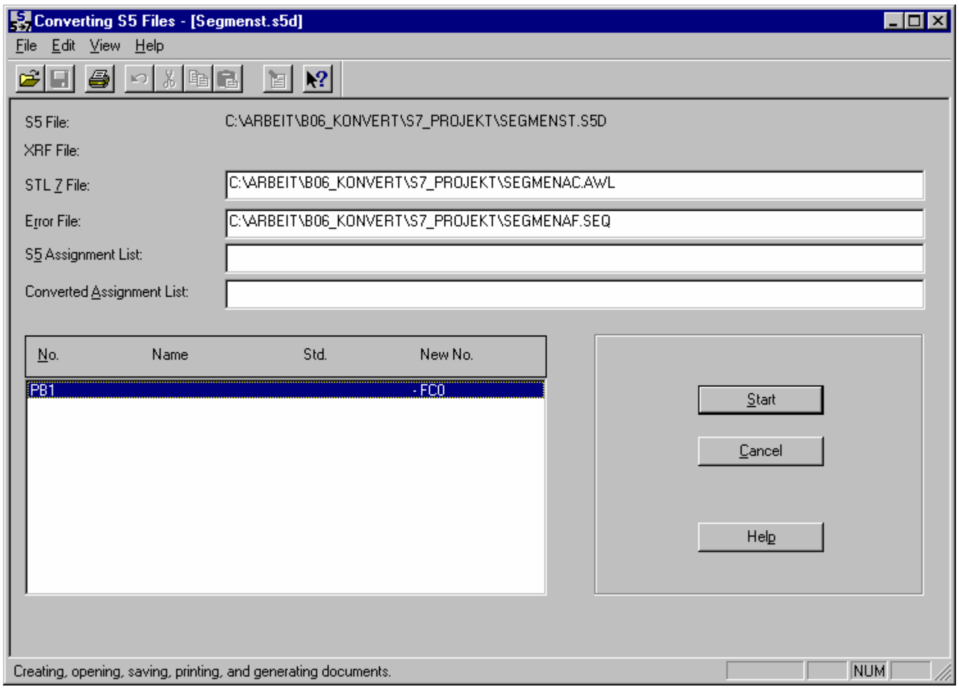
Laufwerke: c:

5. If no cross-reference list of the type ***XR.INI** is available, you receive the message: **No XRF file available, blocks cannot be sorted according to their call**. You can proceed further with a click on **OK** (→ OK).



No XRF file available, blocks cannot be sorted according to their call!

6. In the resulting dialog box, completely participating data and blocks of the conversion are displayed.
With a click on a data block name, you can modify the standard input for the name of the new data and the number of blocks.
Activate the name and start the conversion with the **Start-** Button. (→ Start).



S5 File: C:\ARBEIT\B06_KONVERT\S7_PROJEKT\SEGMENST.S5D

XRF File:

STL 7 File: C:\ARBEIT\B06_KONVERT\S7_PROJEKT\SEGMENAC.AWL

Error File: C:\ARBEIT\B06_KONVERT\S7_PROJEKT\SEGMENAF.SEQ

S5 Assignment List:

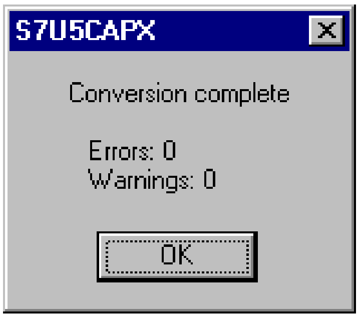
Converted Assignment List:

No.	Name	Std.	New No.
PB1			- FC0

Creating, opening, saving, printing, and generating documents.

7. During the conversion, the actual state of the conversion is displayed in a window (Status window). With a click on the command button **Cancel**, you can end the conversion.

8. Activate the end of the conversion with a click on the command button **OK** in the dialog box (→ OK).

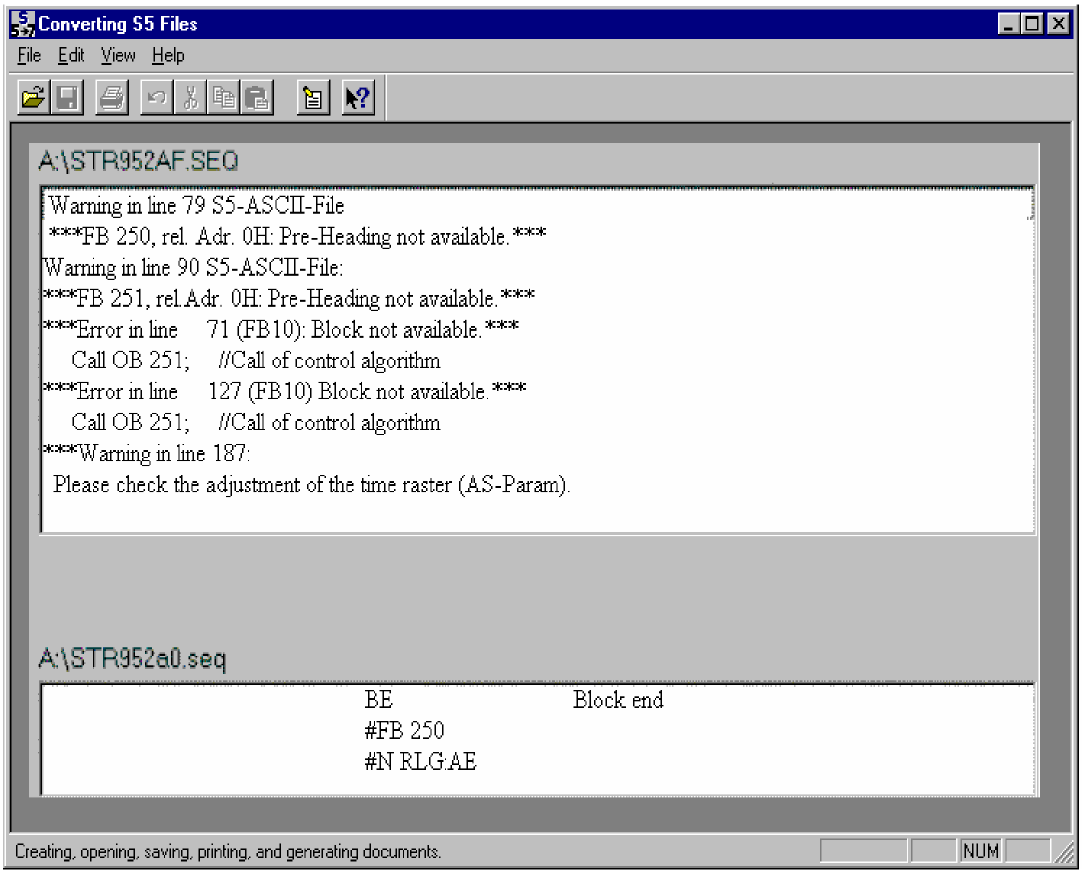


S7U5CAPX

Conversion complete

Errors: 0
Warnings: 0

9. If problems arise by the conversion, erros and warnings are displayed in a separate message window. There notes are also given to error recovery. Follow the recommended corrective measures to remove the errors.



A:\STR952AF.SEQ

```
Warning in line 79 S5-ASCII-File
 ***FB 250, rel. Adr. 0H: Pre-Heading not available.***
Warning in line 90 S5-ASCII-File:
***FB 251, rel.Adr. 0H: Pre-Heading not available.***
***Error in line    71 (FB10): Block not available.***
    Call OB 251;    //Call of control algorithm
***Error in line    127 (FB10) Block not available.***
    Call OB 251;    //Call of control algorithm
***Warning in line 187:
 Please check the adjustment of the time raster (AS-Param).
```

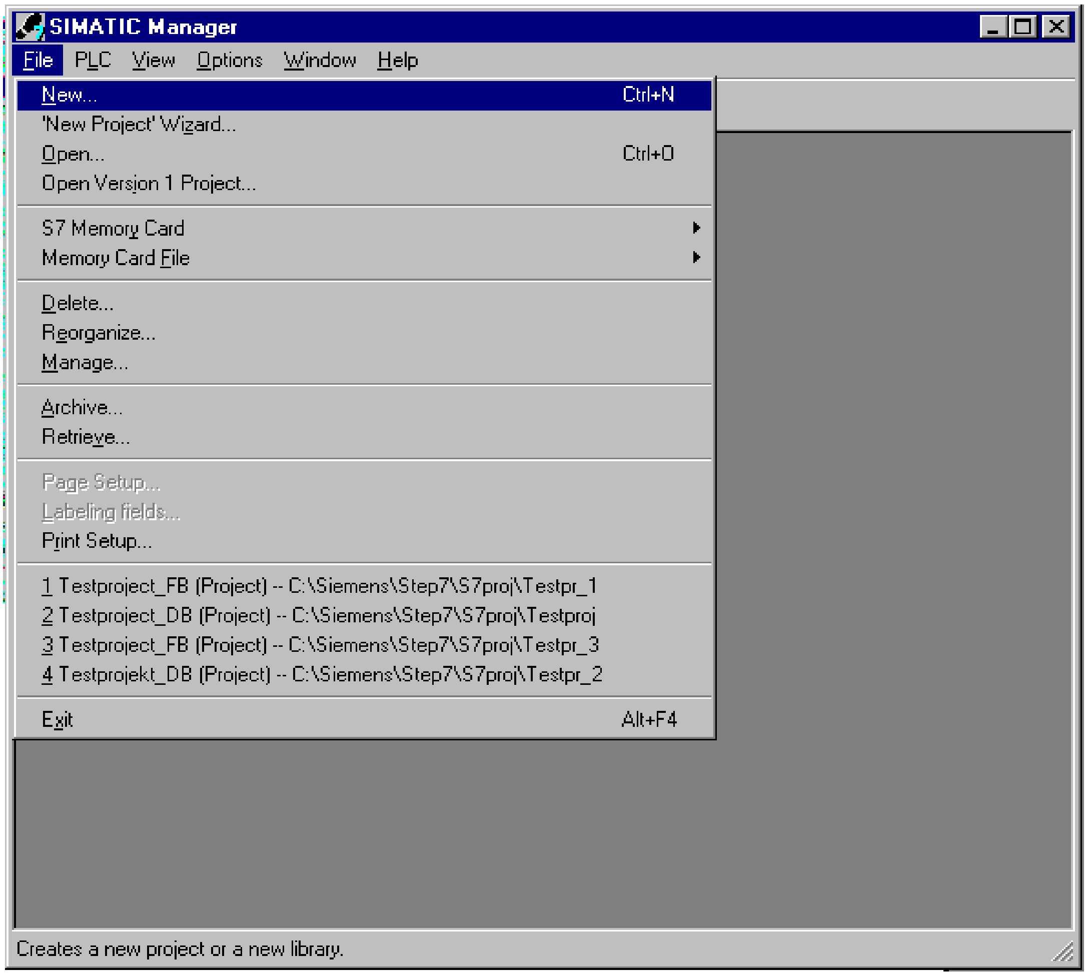
A:\STR952a0.seq

```
                    BE                    Block end
                    #FB 250
                    #N RLG:AE
```

Creating, opening, saving, printing, and generating documents.

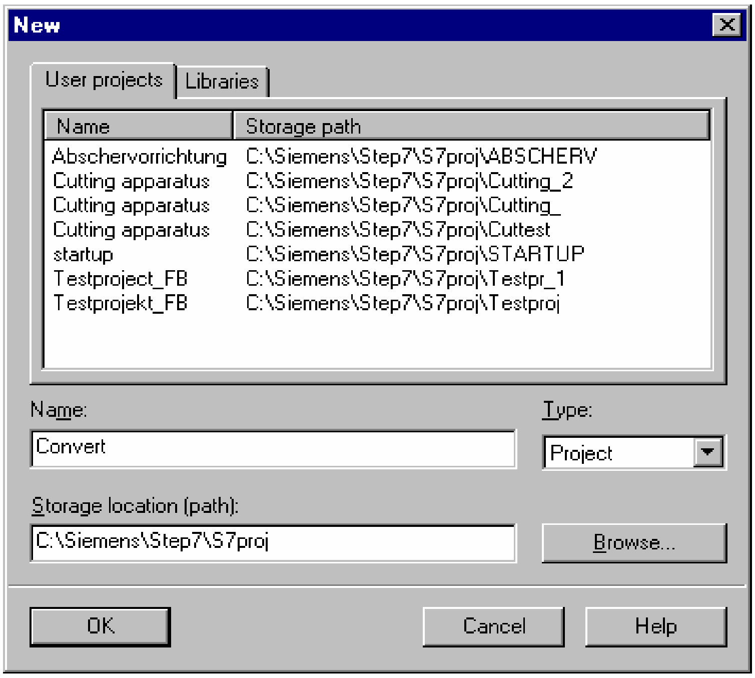
10. After the following conversion of STL data, the data must be incorporated into a STEP 7- Project. This project is generated with the **SIMATIC Manager**, which is opened with a double click (→ SIMATIC Manager).

11. Create a new project (→ File → New)


SIMATIC Manager
File
PLC
View
Options
Window
Help
New... Ctrl+N
'New Project' Wizard...
Open... Ctrl+O
Open Version 1 Project...
S7 Memory Card
Memory Card File
Delete...
Reorganize...
Manage...
Archive...
Retrieve...
Page Setup...
Labeling fields...
Print Setup...
1 Testproject_FB (Project) -- C:\Siemens\Step7\S7proj\Testpr_1
2 Testproject_DB (Project) -- C:\Siemens\Step7\S7proj\Testproj
3 Testproject_FB (Project) -- C:\Siemens\Step7\S7proj\Testpr_3
4 Testprojekt_DB (Project) -- C:\Siemens\Step7\S7proj\Testpr_2
Exit Alt+F4

Creates a new project or a new library.

12. Create a project with the name **Convert**. (→ 'Convert' → OK).



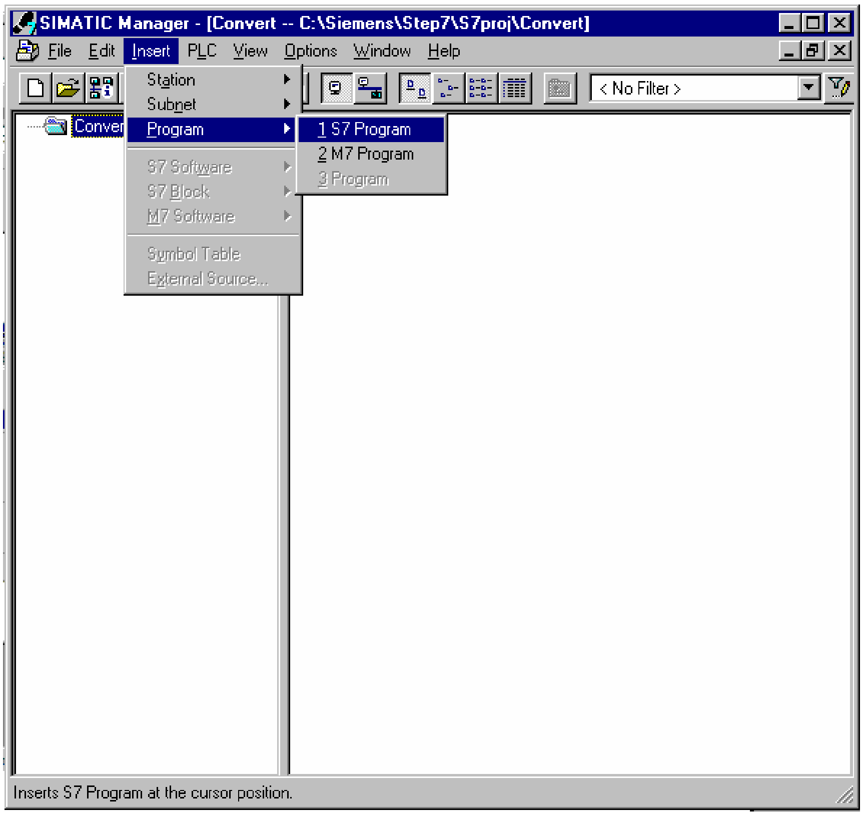
Name	Storage path
Abschervorrichtung	C:\Siemens\Step7\S7proj\ABSCHERV
Cutting apparatus	C:\Siemens\Step7\S7proj\Cutting_2
Cutting apparatus	C:\Siemens\Step7\S7proj\Cutting_
Cutting apparatus	C:\Siemens\Step7\S7proj\Cuttest
startup	C:\Siemens\Step7\S7proj\STARTUP
Testproject_FB	C:\Siemens\Step7\S7proj\Testpr_1
Testprojekt_FB	C:\Siemens\Step7\S7proj\Testproj

Name: Convert

Type: Project

Storage location (path): C:\Siemens\Step7\S7proj

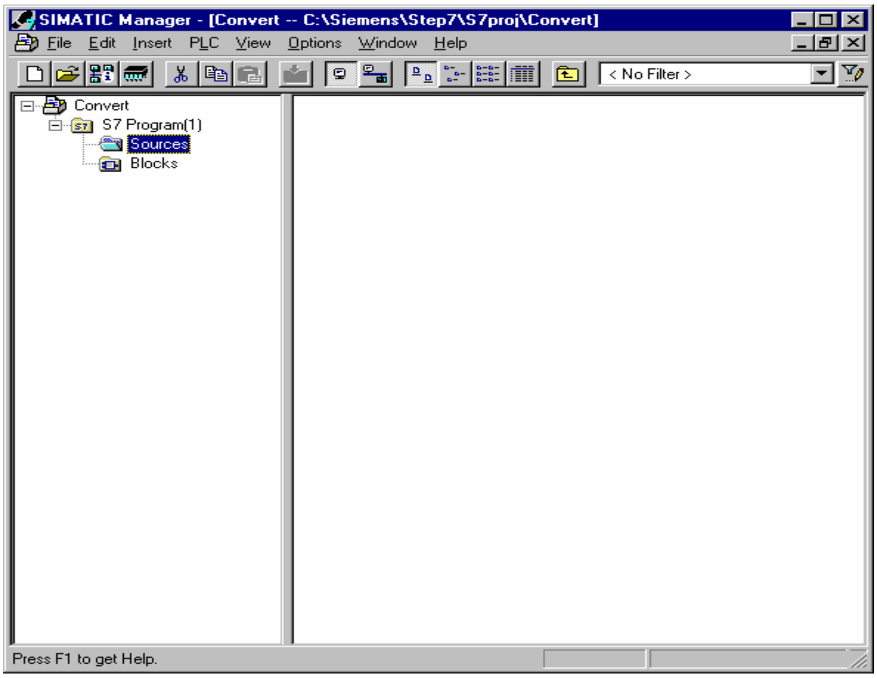
13. Insert a new **S7-Program** (→ Insert → Program → S7-Program).



SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]

Inserts S7 Program at the cursor position.

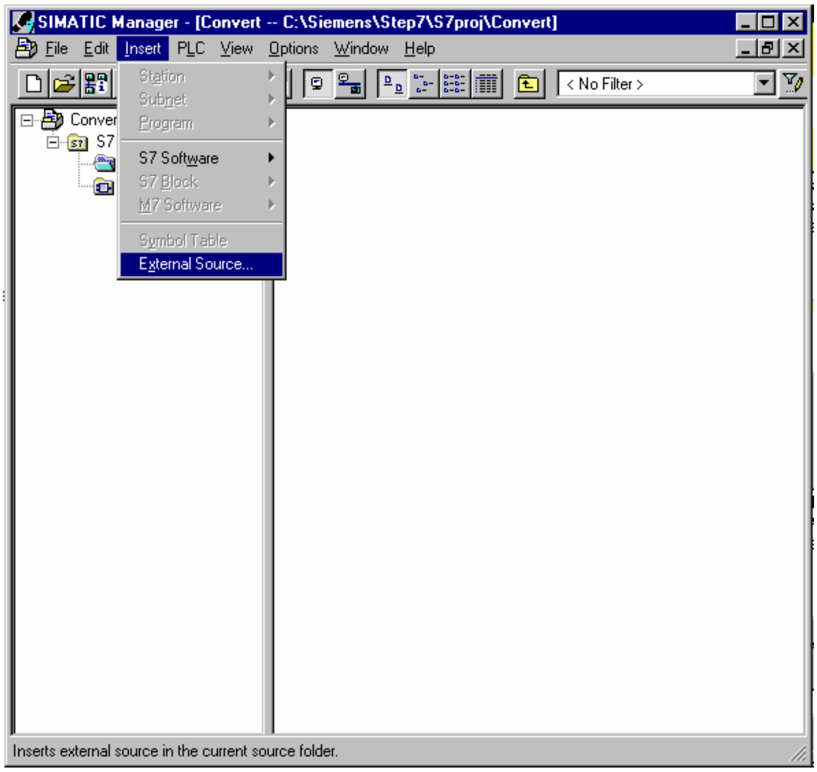
14. Highlight the folder **Sources** (→ Sources).



SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]

Press F1 to get Help.

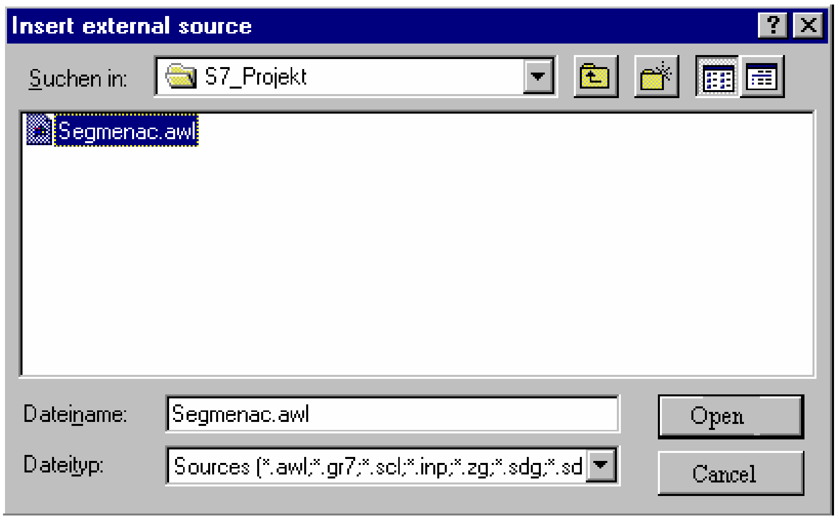
15. Then the ***.AWL** (German abbreviation for STL) data created during the conversion can be inserted in the directory as **External Sources** (→ Insert → External Sources).



SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]

Inserts external source in the current source folder.

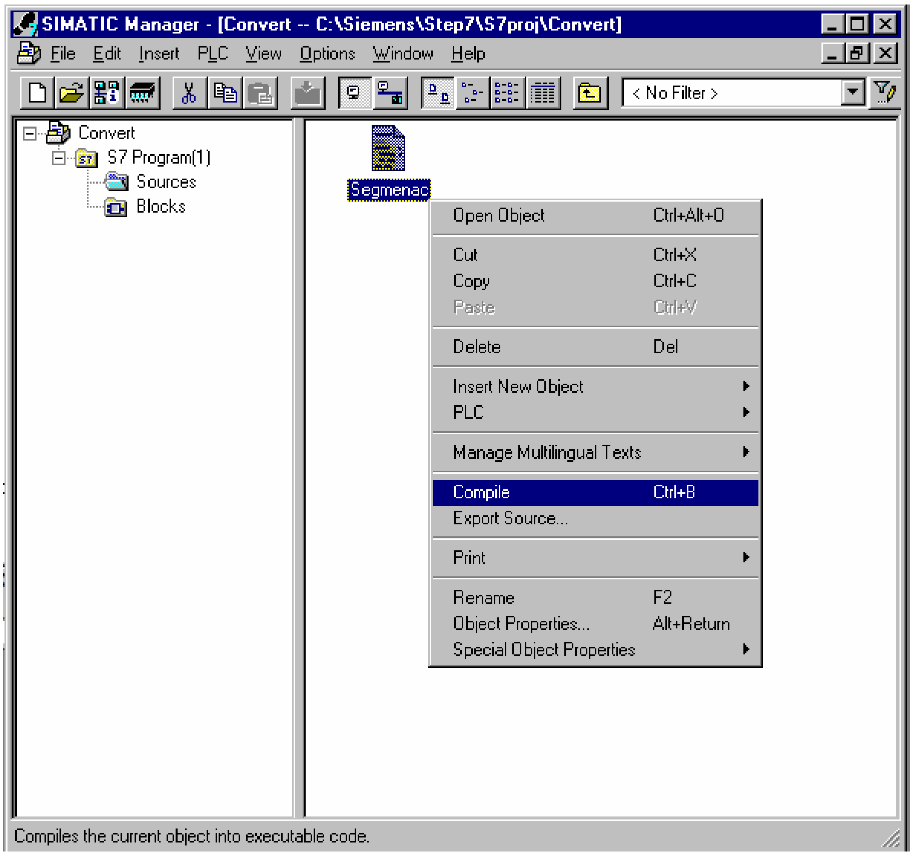
16. Choose the data **Segmenac.AWL** (→ Segmenac.awl → Open).



Dateiname: Segmenac.awl

Dateityp: Sources (*.awl;*.gr7;*.scl;*.inp;*.zg;*.sdg;*.sd

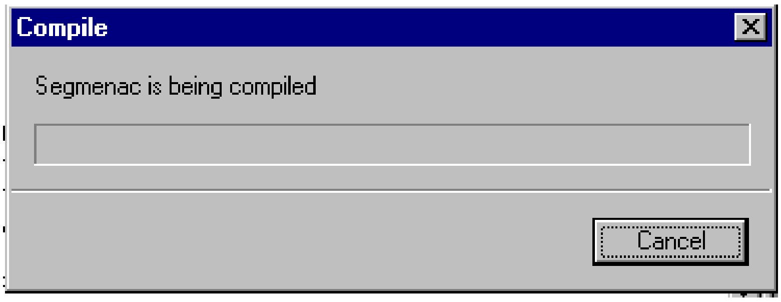
17. Now the source file **Segmenac** must be chosen with the right-mouse button and **Compile** must be started.(→ Segmenac → Compile).



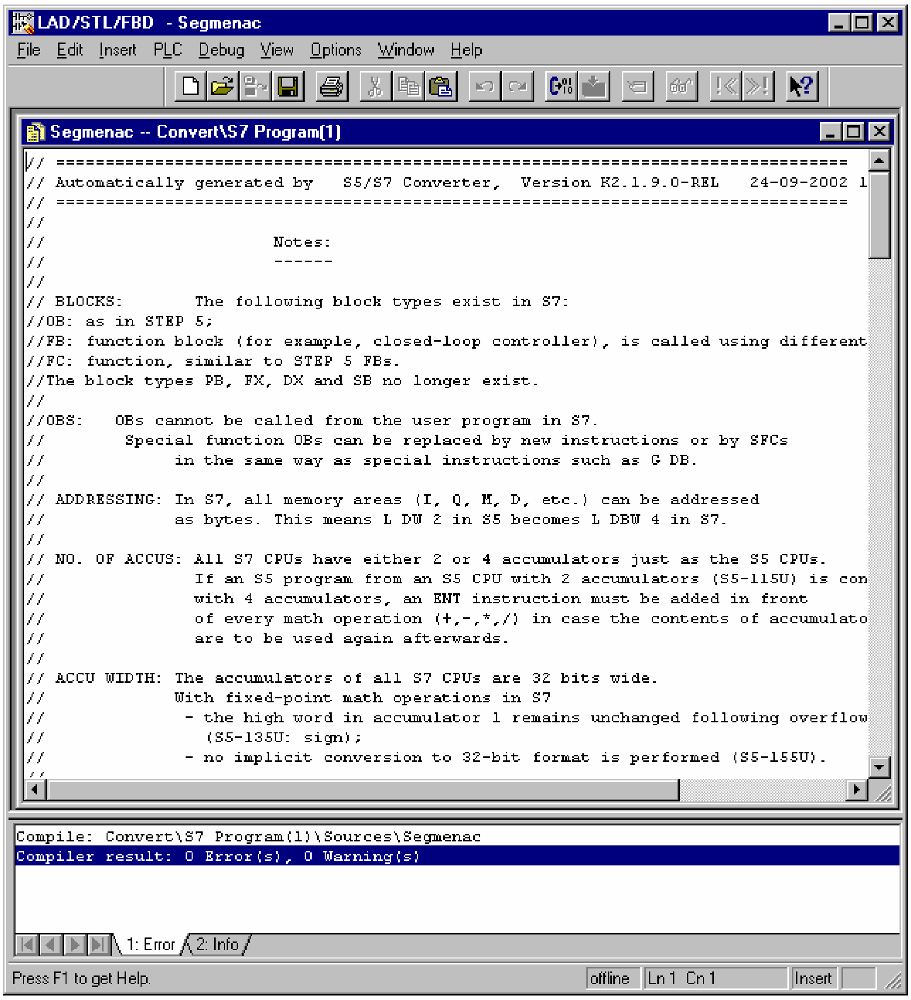
SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]
File Edit Insert PLC View Options Window Help
< No Filter >
Convert
S7 Program(1)
Sources
Blocks
Segmenac
Open Object Ctrl+Alt+O
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Delete Del
Insert New Object
PLC
Manage Multilingual Texts
Compile Ctrl+B
Export Source...
Print
Rename F2
Object Properties... Alt+Return
Special Object Properties

Compiles the current object into executable code.

18. The status of the compilation is displayed. With **Cancel**, the activity can be stopped.



19. After the following compilation, the source file and a compile protocol are included in the warning and error display in the tool **LAD/STL/FBD**. The notes on the important revision of the complied blocks are given in the source data. This tool can be closed with ☒ (→ ☒).


LAD/STL/FBD - Segmenac
File Edit Insert PLC Debug View Options Window Help
Segmenac -- Convert\S7 Program(1)

```
// ============================================================================
// Automatically generated by   S5/S7 Converter,  Version K2.1.9.0-REL   24-09-2002 1
// ============================================================================
//
//                      Notes:
//                      ------
//
// BLOCKS:       The following block types exist in S7:
//OB: as in STEP 5;
//FB: function block (for example, closed-loop controller), is called using different
//FC: function, similar to STEP 5 FBs.
//The block types PB, FX, DX and SB no longer exist.
//
//OBS:   OBs cannot be called from the user program in S7.
//        Special function OBs can be replaced by new instructions or by SFCs
//             in the same way as special instructions such as G DB.
//
// ADDRESSING: In S7, all memory areas (I, Q, M, D, etc.) can be addressed
//             as bytes. This means L DW 2 in S5 becomes L DBW 4 in S7.
//
// NO. OF ACCUS: All S7 CPUs have either 2 or 4 accumulators just as the S5 CPUs.
//               If an S5 program from an S5 CPU with 2 accumulators (S5-115U) is con
//               with 4 accumulators, an ENT instruction must be added in front
//               of every math operation (+,-,*,/) in case the contents of accumulato
//               are to be used again afterwards.
//
// ACCU WIDTH: The accumulators of all S7 CPUs are 32 bits wide.
//             With fixed-point math operations in S7
//              - the high word in accumulator 1 remains unchanged following overflow
//                (S5-135U: sign);
//              - no implicit conversion to 32-bit format is performed (S5-155U).
```

```
Compile: Convert\S7 Program(1)\Sources\Segmenac
Compiler result: 0 Error(s), 0 Warning(s)
```

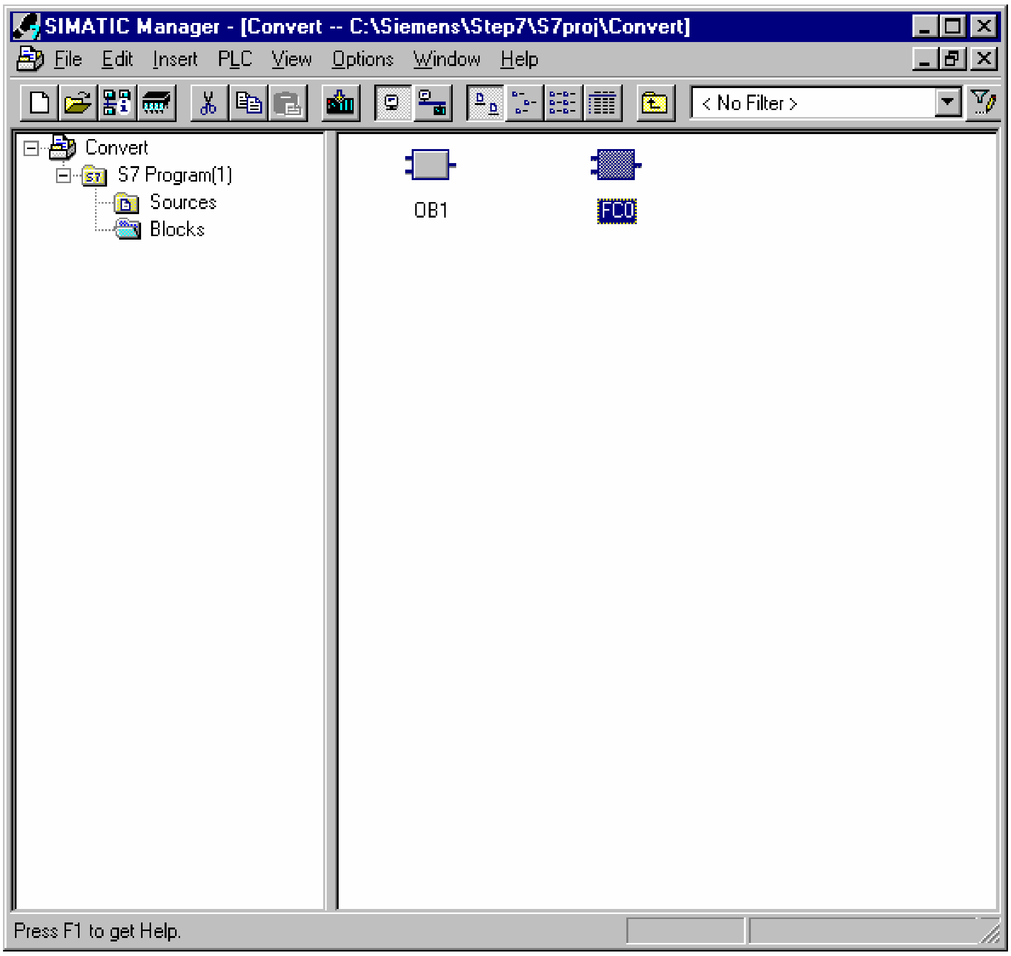
20. If no error is present, the conversion is finished and the generation of the program blocks stand in the data directory **Blocks**.

4. REWIRING

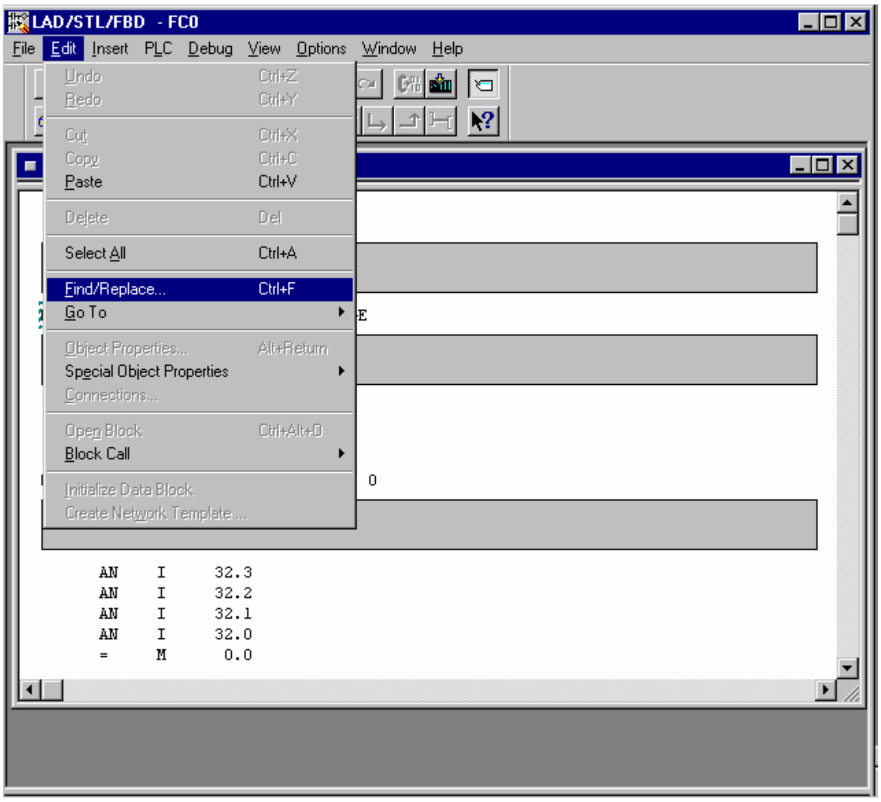
Because the addressing in S7 is no longer identical to the addressing in S5, all alternative operands (most In-/ and Outputs!) must be rewritten.
This rewiring takes place block wise in the tool **LAD, STL, FBD: Program blocks**.

The following steps must be followed by the user in order to rename the operands in a block:

1. Open the desired block in ‘SIMATIC Manager‘ with a double clock (→).

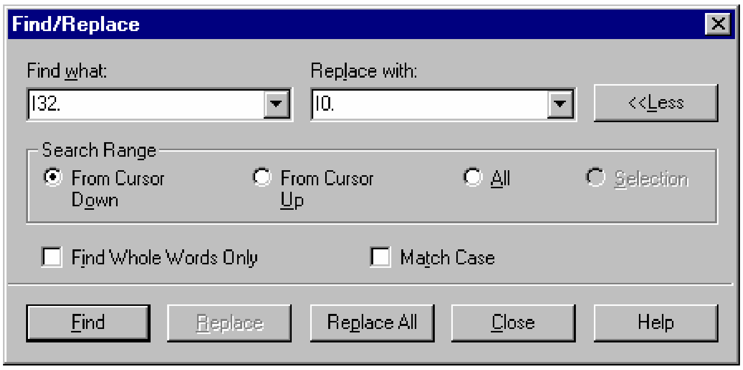

SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]
File Edit Insert PLC View Options Window Help
< No Filter >
Convert
S7 Program(1)
Sources
Blocks
OB1
FC0
Press F1 to get Help.

2. In the menu of the tool **LAD, STL, FBD: Program blocks**, choose the function **Find/Replace**. (→ Edit → Find/Replace)



```
AN   I    32.3
AN   I    32.2
AN   I    32.1
AN   I    32.0
=    M     0.0
```

3. The operations of the replaced data should be found with **Find what** and the new name is entered with **Replace with**. The search range is actuated with **All**, so that the whole block is processed. After, click on **Replace All** (→ Find what → I32. → Replace with → I0. → All → Replace all).



Find what: I32.

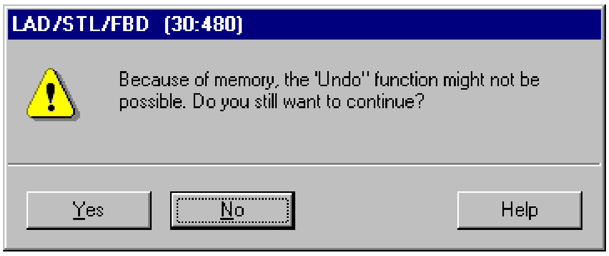
Replace with: I0.

Search Range
● From Cursor Down ○ From Cursor Up ○ All ○ Selection

□ Find Whole Words Only □ Match Case

Note: The search function does not only search for operands, but also character strings. Therefore only the first part of the operation needs to be given. It searches for all inputs with the byte address 32.

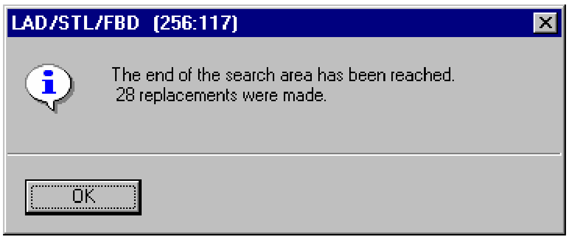
4. In this case, many operands will be modified. The function Undo is no longer possible, when the warning is confirmed with ‘Yes‘ (→ Yes).



LAD/STL/FBD (30:480)

Because of memory, the ''Undo'' function might not be possible. Do you still want to continue?

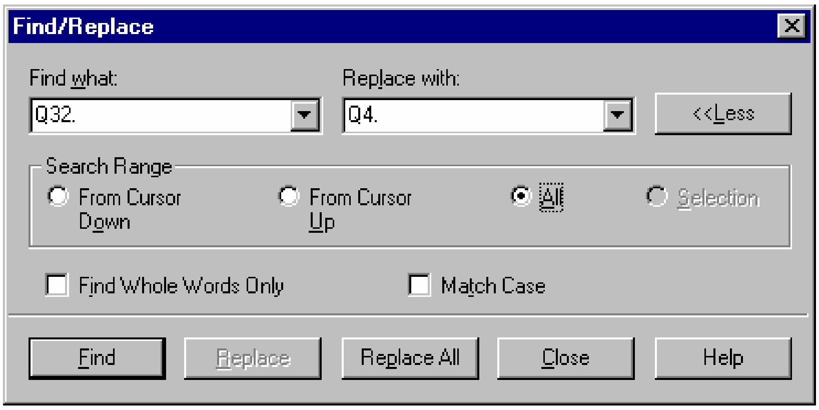
5. At the end of the rewiring, the count of the replaced words is given. Confirm this warning with ‘OK’ (→ OK).



LAD/STL/FBD (256:117)

The end of the search area has been reached.
28 replacements were made.

6. The outputs must also be given new byte addresses.The operations of the replaced data should be found with **Find what** and the new name is entered with **Replace with**. The search range is actuated with **All**, so that the whole block is processed. After, click on **Replace All** (→ Find what → Q32. → Replace with → Q4. → All → Replace all).



Find/Replace

Find what: Q32.

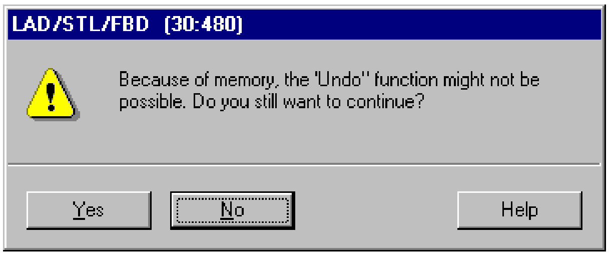
Replace with: Q4.

Search Range

○ From Cursor Down ○ From Cursor Up ● All ○ Selection

□ Find Whole Words Only □ Match Case

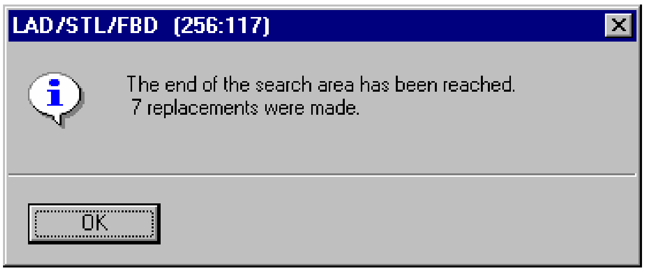
7. In this case, many operands will be modified. The function Undo is no longer possible, when the warning is confirmed with ‘Yes‘ (→ Yes).



LAD/STL/FBD (30:480)

Because of memory, the 'Undo'' function might not be possible. Do you still want to continue?

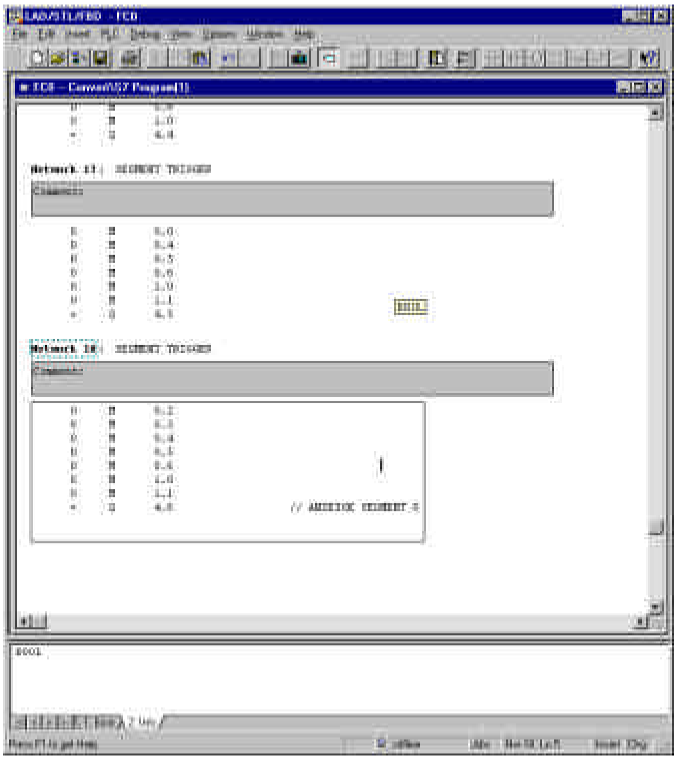
8. At the end of the rewiring, the count of the replaced words is given. Confirm this warning with ‘OK’ (→ OK).



LAD/STL/FBD (256:117)

The end of the search area has been reached.
7 replacements were made.

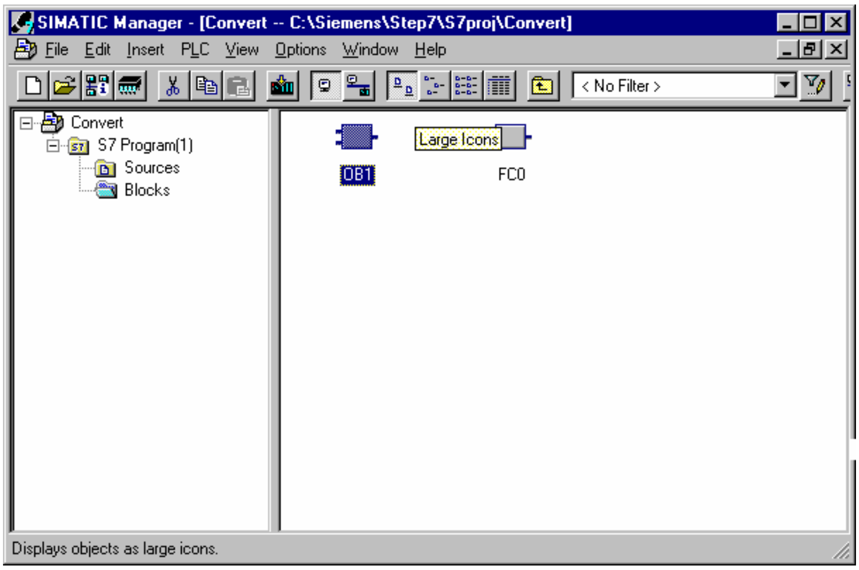
9. The FC0 is now adjusted to the peripherals of the SIMATIC S7-300 and can be saved 💾 and downloaded into the PLC 📥 (→ 💾 → 📥).



LAD/STL/FBD - FC0

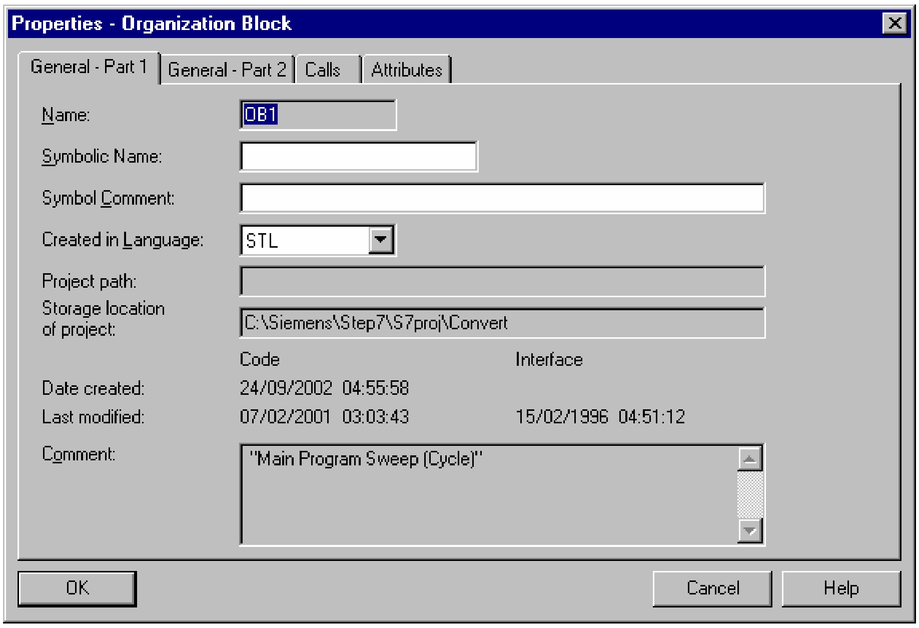
FC0 -- Convert\S7 Program(1)

10. For the programming of the FC call, open the **OB1** in **SIMATIC Manager** with a double click (→ SIMATIC Manager → OB1).


SIMATIC Manager - [Convert -- C:\Siemens\Step7\S7proj\Convert]
File Edit Insert PLC View Options Window Help
< No Filter >
Convert
S7 Program(1)
Sources
Blocks
Large Icons
OB1
FC0

Displays objects as large icons.

11. The properties of the OB1 are held and accepted with **OK** (→ OK).



Name: OB1

Symbolic Name:

Symbol Comment:

Created in Language: STL

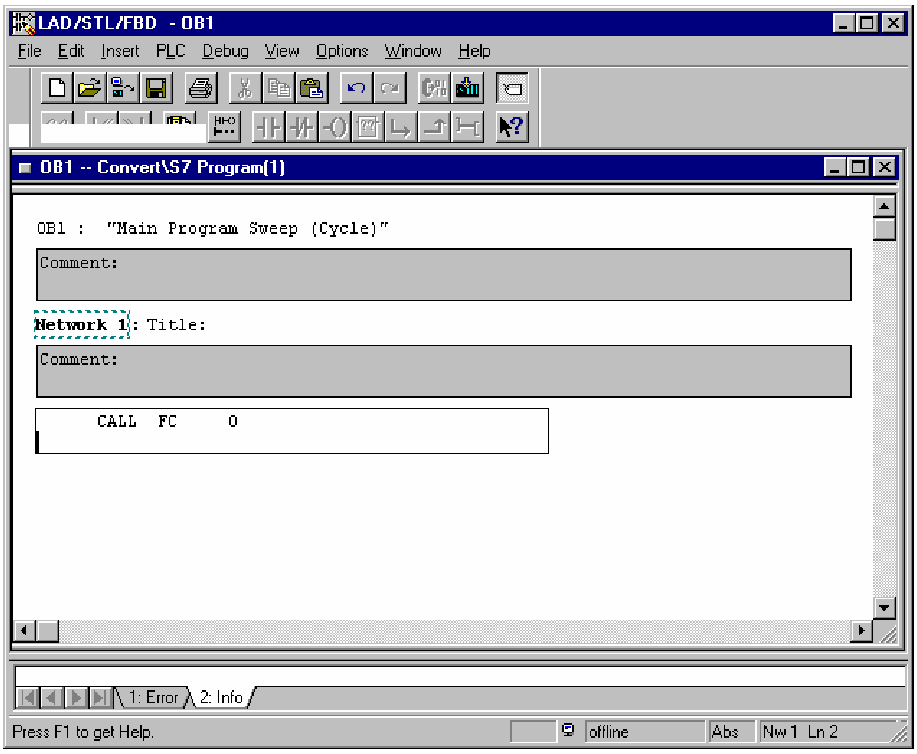
Project path:

Storage location of project: C:\Siemens\Step7\S7proj\Convert

	Code	Interface
Date created:	24/09/2002 04:55:58	
Last modified:	07/02/2001 03:03:43	15/02/1996 04:51:12

Comment: "Main Program Sweep (Cycle)"

12. Now the call of the **FC0** with the instruction **CALL FC 0** in Network 1 from the OB1 can be entered. Then the OB1 is saved and downloaded into the PLC (→ CALL FC 0 → →).



LAD/STL/FBD - OB1
File Edit Insert PLC Debug View Options Window Help
OB1 -- Convert\S7 Program(1)

```
OB1 :   "Main Program Sweep (Cycle)"
Comment:

Network 1: Title:
Comment:

      CALL  FC     0
```

1: Error
2: Info
Press F1 to get Help.
offline
Abs
Nw 1 Ln 2

13. Through the switching of the mode switch, the program will start.
In this example, a BCD coded value can be given to the input group with the first 4 switches (I0.0 / I0.1 / I0.2 / I0.3).
This value can then be connected to a seven segment display that will display the first byte (AB4) of the output module.