

User Manual

Connection to 3S sarti

Part Number: 80 860.630
Version: 3
Date: 08.12.2006
Valid for: TSwin .net 4.1x + SP3

Version	Date	Modifications
1	04.07.2005	First edition
2	24.11.2005	Validation extended, chapter "Important Notes" added
3	06.12.2006	Validation limited, Chapter Protocol Parameters extended

This manual, including all illustrations contained herein, is copyright protected. Use of this manual by any third party in departure from the copyright provision is forbidden. No part of this manual may be reproduced, translated or electronically or photographically archived or altered without the express written consent from Süttron electronic GmbH. Violations shall be cause for damage liability.

Süttron electronic reserves the right to make any changes that contribute to technical improvement.

Overall Table of Contents

1	Important Notes	1-1
	1.1 Symbols	1-1
	1.2 Safety Notes	1-1
	1.3 Intended Use.....	1-1
	1.4 Target Group.....	1-1
2	3S sarti.....	2-1
	2.1 Data Types.....	2-1
	2.1.1 Single Variables.....	2-1
	2.1.2 String Variables	2-1
	2.2 Programming	2-2
	2.2.1 Protocol parameters	2-2
	2.2.1.1 Connection Name	2-2
	2.2.1.2 Path for Variable List *.sym	2-2
	2.2.1.3 Controller Address	2-2
	2.2.1.4 Port Number	2-2
	2.2.1.5 Byte Order	2-3
	2.2.1.6 Offline Operation.....	2-3
	2.2.2 Polling Area	2-3
	2.2.3 Status Messages	2-3
	2.2.4 Date and Time	2-4
	2.2.5 Variant Buffer.....	2-4
	2.2.6 Tables	2-4
	2.3 Error Messages.....	2-5
	2.4 Applications.....	2-6
	2.4.1 CoDeSys Version 2.2 or Higher.....	2-6
	2.4.1.1 Declaring Global Variables	2-6
	2.4.1.2 Activate Output into Symbol File.....	2-6
	2.4.1.3 Target System Settings	2-8
	2.4.1.4 Variable List	2-9
A	Index.....	A-1

1 Important Notes

1.1 Symbols

The symbols in this manual are used to draw your attention on notes and dangers.



Danger

This symbol is used to refer to instructions which, if ignored or not carefully followed could result in personal injury.



Note

This symbol indicates application tips or supplementary notes.



Reference to source of information

This symbol refers to detailed sources of information on the current topic.

1.2 Safety Notes

- Read this manual carefully before using the software. Keep this manual in a place where it is always accessible to all users.
- The user manual, in particular the safety notes, must be observed by all personnel working with the software and the programmed device.
- Observe the accident prevention rules and regulations that apply to the operating site.
- Installation and operation must only be carried out by qualified and trained personnel.

1.3 Intended Use

- The software has to be used for programming operating devices exclusively. Every other use is not permitted.

1.4 Target Group

All configuration and programming work in connection with the automation system must be performed by trained personnel only (e.g. qualified electricians, electrical engineers).

The configuration and programming personnel must be familiar with the safety concepts of automation technology.

2 3S sarti

The 3S sarti protocol connects an emPower Line operating device with TSvisRT CE to the CoDeSys SP runtime system as a SymARTI client. It provides random read and write access to all global data objects of the soft PLC.

TSwin adopts the data objects of the project_name.SYM file which are created when the CoDeSys project is compiled.

The operating device uses the symbolic name to access a data object of the Symbol Data Dictionary (SDD) file.

2.1 Data Types

The length of a variable is determined by the length defined in the programming software CoDeSys.

2.1.1 Single Variables

You can access variables of the following type: BOOL, BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, REAL, and STRING. Floating point numbers are interpreted in IEEE format. The variable type REAL is required for this purpose.

2.1.2 String Variables

For string variables, the variable type STRING(N) is used, where N is the length of the string.



Please note that the length of the string variables in the symbol file is by 1 larger than defined.

Example: STRING(40) has a length of 41.

2.2 Programming

2.2.1 Protocol parameters

2.2.1.1 Connection Name

Enter a connection name to any used connection.

Table 2-1 Connection name

Configurable Values	Default Value
any	PLC_LOCAL

2.2.1.2 Path for Variable List *.sym

This parameter specifies the directory in which the variable list *.sym is stored. You have to assign a connection list for any used connection.

The variable list *.sym is created by the programming software **CoDeSys** when compilation takes place.

TSwin reads all entries from this file, which are according to the following criteria:

- the line begins with a fullstop and
- the symbolic name is shorter than 80 characters.



CoDeSys enters only global variables into the variable list *.sym!

2.2.1.3 Controller Address

Enter the IP address or the host name for any used connection into the controller address field.

Table 2-2 Controller address

Configurable Values	Default Value
0.0.0.0 to 255.255.255.255	localhost

2.2.1.4 Port Number

This parameter specifies the port number with up to 5 digits, which is used to communicate with the operating device. You can choose the port number freely.

Table 2-3 Port number

Configurable Values	Default Value
0 to 65535	1200

2.2.1.5 Byte Order

Select the byte order of the target for any used connection.

Table 2-4 Byte order

Configurable Values	Default Value
Intel	X
Motorola	

2.2.1.6 Offline Operation

Check the Offline operation check box to write communication errors to the message buffer without reboot of the project.

If the check box is not checked the communication error is displayed at the operating devices display and afterwards the project is rebooted.

Table 2-5 Offline operation

Configurable Values	Default Value
ON	X
OFF	

You must create the appropriate error messages like the following. The error codes are represented by the integrated system variables ComErrorCode and ComErrorSubcode within the message texts.

The message numbers 9901 to 9916 represent the connections 1 to 16.

9901	Communication Error, Code: 99999, Subcode: 99999
9902	Communication Error, Code: 99999, Subcode: 99999
9903	Communication Error, Code: 99999, Subcode: 99999
9904	Communication Error, Code: 99999, Subcode: 99999
9905	Communication Error, Code: 99999, Subcode: 99999
9906	Communication Error, Code: 99999, Subcode: 99999
9907	Communication Error, Code: 99999, Subcode: 99999
9908	Communication Error, Code: 99999, Subcode: 99999
9909	Communication Error, Code: 99999, Subcode: 99999
9910	Communication Error, Code: 99999, Subcode: 99999
9911	Communication Error, Code: 99999, Subcode: 99999
9912	Communication Error, Code: 99999, Subcode: 99999
9913	Communication Error, Code: 99999, Subcode: 99999
9914	Communication Error, Code: 99999, Subcode: 99999
9915	Communication Error, Code: 99999, Subcode: 99999
9916	Communication Error, Code: 99999, Subcode: 99999

Figure 2-1 Error messages for offline operation

2.2.2 Polling Area

The poll area is used to manage the write coordination byte (WCB), the serial message channel and the LEDs in the function keys. This area is continuously polled by the operating device.

This protocol requires you to set up the poll area with three single variables.

Table 2-6 Data types for the poll area

Area	Valid Data Types
KBS (write coordination byte)	BYTE, USINT, WORD, UINT
Message Channel	WORD, UINT
Function Key LEDs	BYTE, USINT, WORD, UINT, DWORD, UDINT, ARRAY[1..N]

2.2.3 Status Messages

Status messages are the static assignment of flags (bits) in the controller to plain text messages in the operating device. For status message addressing, use the data types BYTE, USINT, WORD, UINT, DWORD, UDINT, or ARRAY[1..N]. The following applies when using ARRAY: The type size multiplied by N provides the size of the message system in bytes.

2.2.4 Date and Time

The variables for synchronizing the time and date must use the data types USINT or ARRAY [1..N] OF BYTE.

Table 2-7 Byte lengths for the date and time

Variable	Length
Date with a 2-digit year	3 Bytes
Date with a 4-digit year	4 Bytes
Time	3 Bytes
Weekday	1 Byte

2.2.5 Variant Buffer

The variable for the variant buffer must use the data type BYTE or USINT.

2.2.6 Tables

The variable for representation of tables must use the data type ARRAY [1..N]. The ARRAY [1..N] has to be of one of the following base data types:

- BOOL,
- BYTE,
- WORD,
- DWORD,
- SINT,
- INT,
- DINT,
- USINT,
- UINT,
- UDINT,
- REAL or
- STRING.

2.3 Error Messages

Error messages are displayed on the operating device along with a code and sub-code. Error messages are composed as follows:

Communication Error

Code XXXXX

Subcode XXXXX

Retries XXXXX

Table 2-8 3S sarti error messages

Code	Subcode	Error Type	Possible Cause
51	02	Memory overrun	
60	02	Channel to the CoDeSys SP can not be opened	
	03	Login command can not be executed	
	04	Login command can not be terminated	
	08	Symbol table in CoDeSys SP not found	
61	08	Symbol table in CoDeSys SP not found	
	09	Wrong access to symbol table in CoDeSys SP	
	10	Symbol table was changed on server, but not updated on client after 3 repetitions	
	11	Variable in symbol table not found	
	12	Wrong access to symbol table in CoDeSys SP	
	13	Wrong access to symbol table in CoDeSys SP	
	14	The variable is not the first item in the table	
80	20	Variable types not the same	Recompile the project using the current symbol file and load it into the operating device again.
	30	Invalid symbol	

2.4 Applications

2.4.1 CoDeSys Version 2.2 or Higher

The programming software takes the global variables from the symbol file project_name.SYM and inserts them into the variable list.

The symbolic names cannot be longer than 80 characters.

The entries in the variable list cannot be modified.

2.4.1.1 Declaring Global Variables

To declare global variables in CoDeSys:

1. Select **Auto Declare** from the **Edit** menu.

The **Declare Variable** dialog opens.

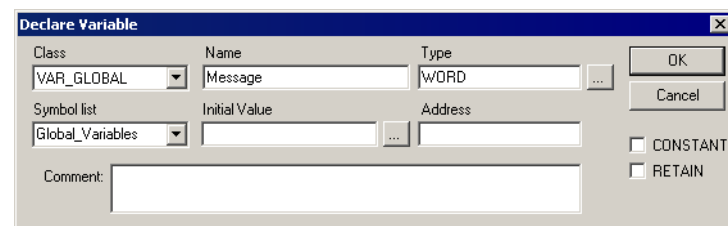


Figure 2-2 Example of a variable declaration for global variables

2. Select the VAR_GOBAL class from the **Class** field.
3. Enter a name (Message) and a type (WORD).
4. Repeat step 3 for all additional global variables.
5. Click **OK** to confirm your input.

The **Global_Variables** window opens.

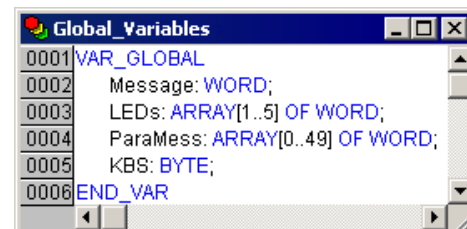


Figure 2-3 Window Global variables

2.4.1.2 Activate Output into Symbol File

Specify the following settings in CoDeSys to write the global variables into a symbolic file.

1. Select **Options** from the **Project** menu.
2. Select **Symbol configuration**.

The **Options** dialog will look as follows:

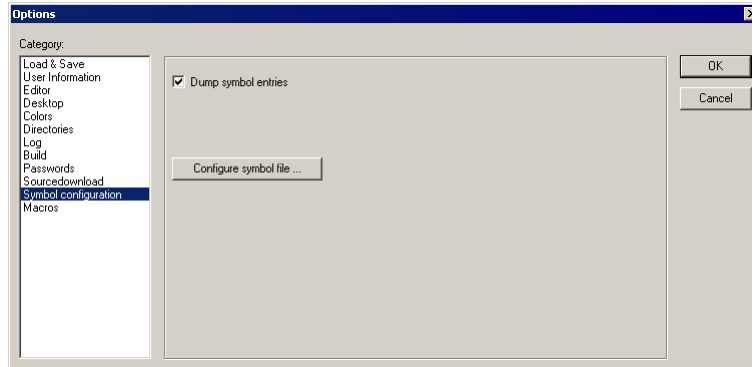


Figure 2-4 Dialog Options - symbol configuration

3. Select the **Dump symbol entries** check box.
4. Click the **Configure symbol file** button.

The **Set object attributes** window opens.

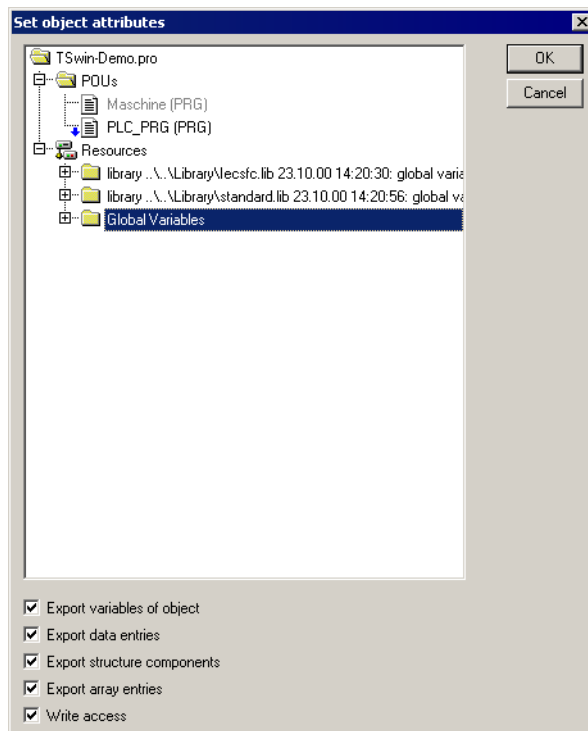


Figure 2-5 Dialog Set object attributes

5. Select the **Global variables** entry.
6. Click **OK** to confirm your selection.

You are returned to the **Options** dialog.

Now you need to specify the position where the symbol file is to be stored.

1. Select **Directories** from the **Options** dialog.

The **Options** dialog will look as follows:

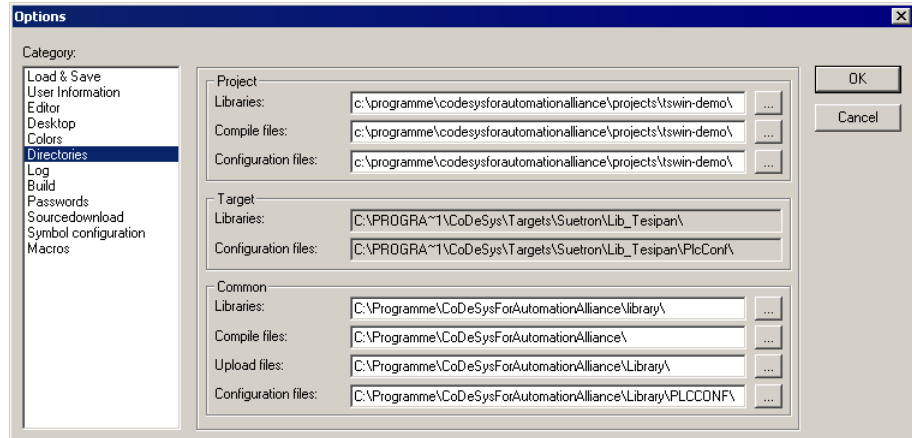


Figure 2-6 Dialog Options - directories

2. From the **Project** area, select a directory for the **compile files**.
3. Click **OK** to confirm your selection.

You are returned to the **Options** dialog.

The symbol file will not be created until a compilation process takes place and is stored in the same directory as the project!

2.4.1.3 Target System Settings

Select the following settings for the target system to ensure the symbol file is sent to the target system:

1. Open the **Resources** tab.
2. Double-click **Target settings**.

The **Target settings** dialog opens.

3. Open the **General** tab.
4. Select the **Download Symbol File** check box.

The **Target settings** dialog might look like the example below:

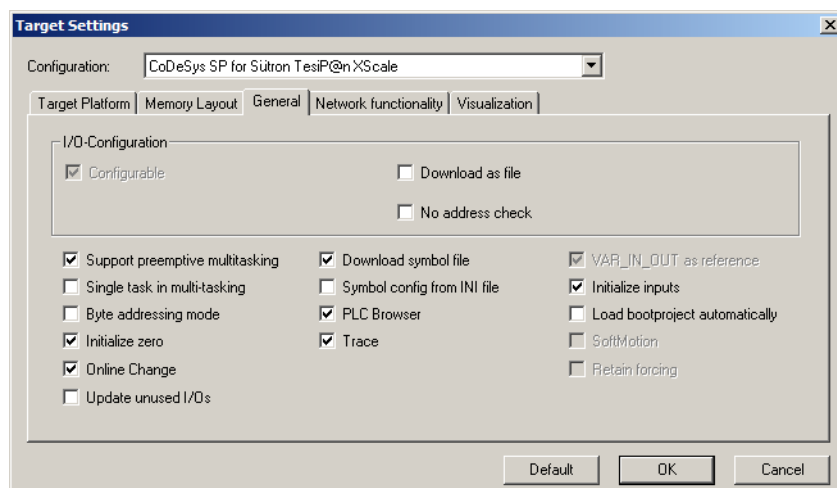
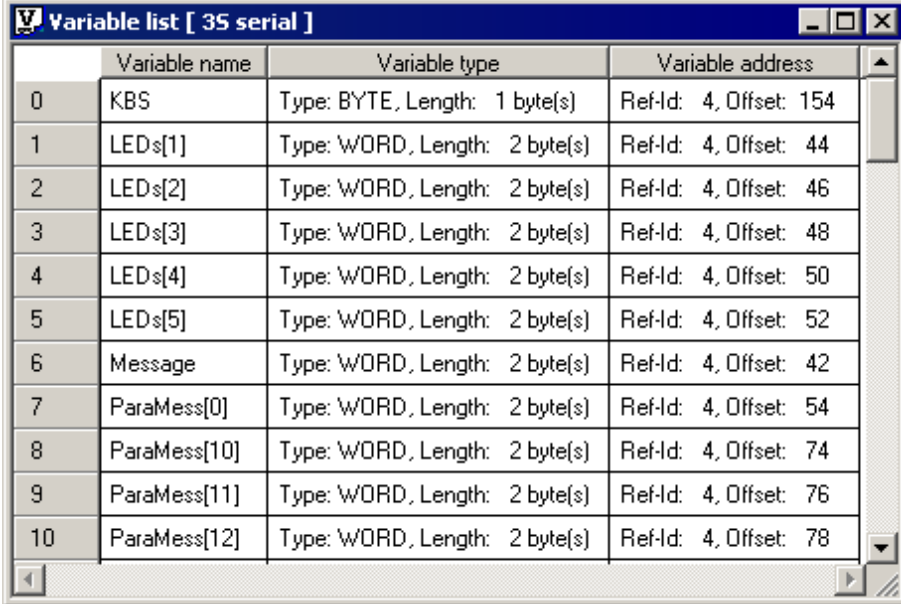


Figure 2-7 Dialog Target settings

2.4.1.4 Variable List

The programming software automatically places the symbolic variable entries created in the example into the variable list if you specified the correct directory and name in the communications parameters.



The screenshot shows a window titled "Variable list [3S serial]" with a table containing 11 rows of variable information. The table has four columns: an index, the variable name, the variable type and length, and the variable address (reference ID and offset).

	Variable name	Variable type	Variable address
0	KBS	Type: BYTE, Length: 1 byte(s)	Ref-Id: 4, Offset: 154
1	LEDs[1]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 44
2	LEDs[2]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 46
3	LEDs[3]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 48
4	LEDs[4]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 50
5	LEDs[5]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 52
6	Message	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 42
7	ParaMess[0]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 54
8	ParaMess[10]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 74
9	ParaMess[11]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 76
10	ParaMess[12]	Type: WORD, Length: 2 byte(s)	Ref-Id: 4, Offset: 78

Figure 2-8 Variable list

This makes the variables globally available in the programming software.

A Index

Nummern

3S sarti..... 2-1

E

Error messages

 3S sarti..... 2-5

I

Important notes..... 1-1

Intended use 1-1

P

Protocol parameters

 3S sarti..... 2-2

S

Safety notes 1-1

Symbols 1-1

T

Target group 1-1



