Foxboro Evo™ Process Automation System

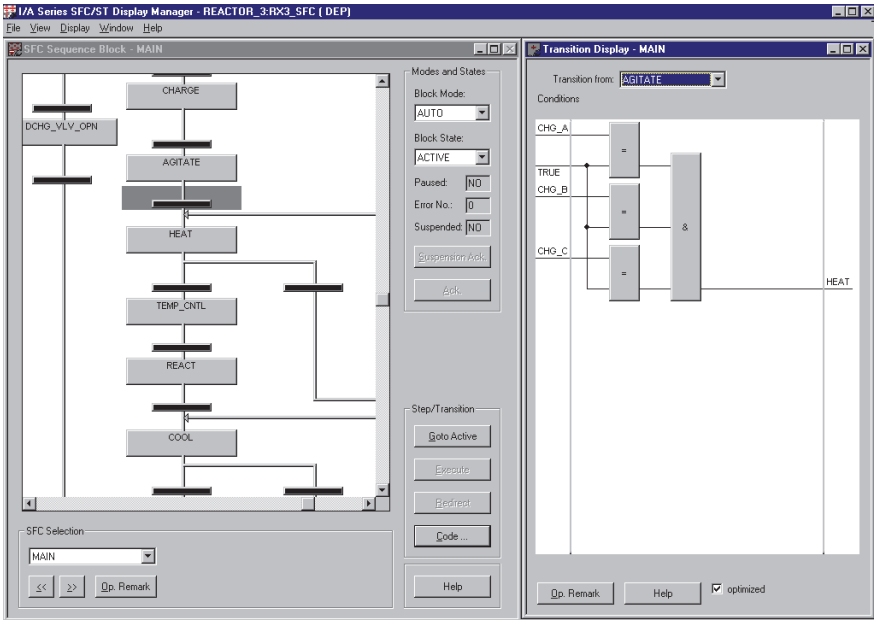
Product Specifications

Foxboro®

by Schneider Electric

PSS 31S-4SFCCFG

Sequential Function Chart/Structured Text (SFC/ST) Configurator and Display Manager



The SFC/ST Configurator and Display Manager provide a graphical method to configure and view Foxboro Evo™ system sequence control blocks.

OVERVIEW

The SFC/ST software package is a graphical user interface (GUI) based on sequential function charts and structured text. It provides an alternate, advanced method of configuring and displaying Foxboro Evo system sequence control blocks (DEP, EXC, and IND).

Configuration and operation views follow the IEC 61131-3 standard for both sequential function chart (SFC) elements and structured text (ST).

EASE OF CONFIGURATION

Whether it is a simple repetitive sequencing operation or the phase logic of a flexible batch application, the SFC/ST Configurator enables engineers to configure sequence control blocks using IEC 61131-3 sequential function charts and structured text. The SFC/ST Configurator eliminates the housekeeping of procedural languages such as program flow control statements, saving both design and test time.

The SFC/ST Display Manager greatly speeds up debugging and commissioning through easy-to-use default displays for sequence blocks configured with the SFC/ST Configurator. The system creates these displays automatically, thus saving the engineering time to build them.

EASE OF OPERATION

When monitoring a sequential operation, the user must frequently determine, quickly and easily, where a particular sequence is in its logic and why it has not advanced beyond that point. Using the SFC/ST Display Manager, the user has quick access to a live, updating display of the sequential function chart for each SFC/ST configured sequence block.

The user can quickly see, via highlighting, which step or transition is currently executing. A simple double-click on a transition shows what condition is being tested, complete with live, updating values of the process variables, for example, level and weight.

SFC/ST CONFIGURATOR OVERVIEW

The SFC/ST Block and Library Configurator enables the user to graphically configure sequence logic using SFC elements and ST. The SFC/ST Library Configurator is not called via ICC, allowing the user to configure routine and block library elements independent of any block creation. Both configurators support subroutines and exception handling, as well as all the existing standard Foxboro® High Level Batch Language (HLBL) commands, with syntax that meets IEC 61131-3 standards.

The SFC/ST Configurator features:

- Main routine configuration with SFC and ST
- Step specification with ST
- Transition specification with ST
- Function blocks to operate on data or alter the flow of control
- Functions to operate on data
- Subroutine configuration with SFC and ST
- Standard Block Exception (SBX) handler configuration with SFC and ST
- User-defined Library of Named Object Files containing SFC structures
- Macros to facilitate the re-use of ST code snippets by name
- Constants, variables, and user labels
- File inclusion to handle collections of #defines and declarations
- Operator remarks
- Sequence logic version control with full history of sequence editing available to view or print out
- Printed documentation for SFC graphical structures, steps, transitions, and comments
- Password protection class setting of certain SFC/ST Display Manager graphical objects to prevent a class of users from performing operations such as changing block mode and state, and tracing ST code (SFC/ST Block Configurator only).

SFC/ST DISPLAY MANAGER OVERVIEW

The SFC/ST Display Manager enables the user to view SFC/ST sequence block data during run time and to perform certain block operations.

The views of an SFC/ST sequence block feature:

- Dynamic, colored view of current position in SFC
- Display of step code
- Dynamic indication of transition evaluations in either graphical or text form
- Display of constants, variables, and user labels
- Embedded operator remarks assisting with sequence and process descriptions.

The views also enable the operator to:

- Change the block mode
- Execute steps and transitions in the semi-automatic (S-AUTO) mode
- Redirect the block's control flow in the manual (MAN) mode
- Trace through ST code of steps and transitions.

Protection class settings lock access to operation buttons, options lists, and menu commands to prevent certain users from performing one or more of the above operations.

SYSTEM INTEGRATION

The SFC/ST software package is available on Windows Foxboro Evo or I/A Series® workstations, and Solaris I/A Series workstations. The user can freely develop and interchange sequence source code between each system.

The SFC/ST Configurator is accessible through the Integrated Control Configurator (ICC). The operator uses the ICC to insert a new sequence block in a control station. The operator can then edit the block's sequence logic using either the traditional HLBL Sequence Logic Editor or the SFC/ST Configurator. For subsequent changes, the user must use the same configurator.

SFC/ST STANDARD ELEMENTS

The SFC/ST software package includes the IEC 61131-3 standard elements described below.

It uses programs, function blocks, and functions as organizational elements for the sequence blocks.

It uses SFCs (see Figure 1) for structuring the internal organization of the sequence block's main routine, subroutines, and SBX routines.

It uses ST to describe the steps and transitions within SFCs (see Figure 1).

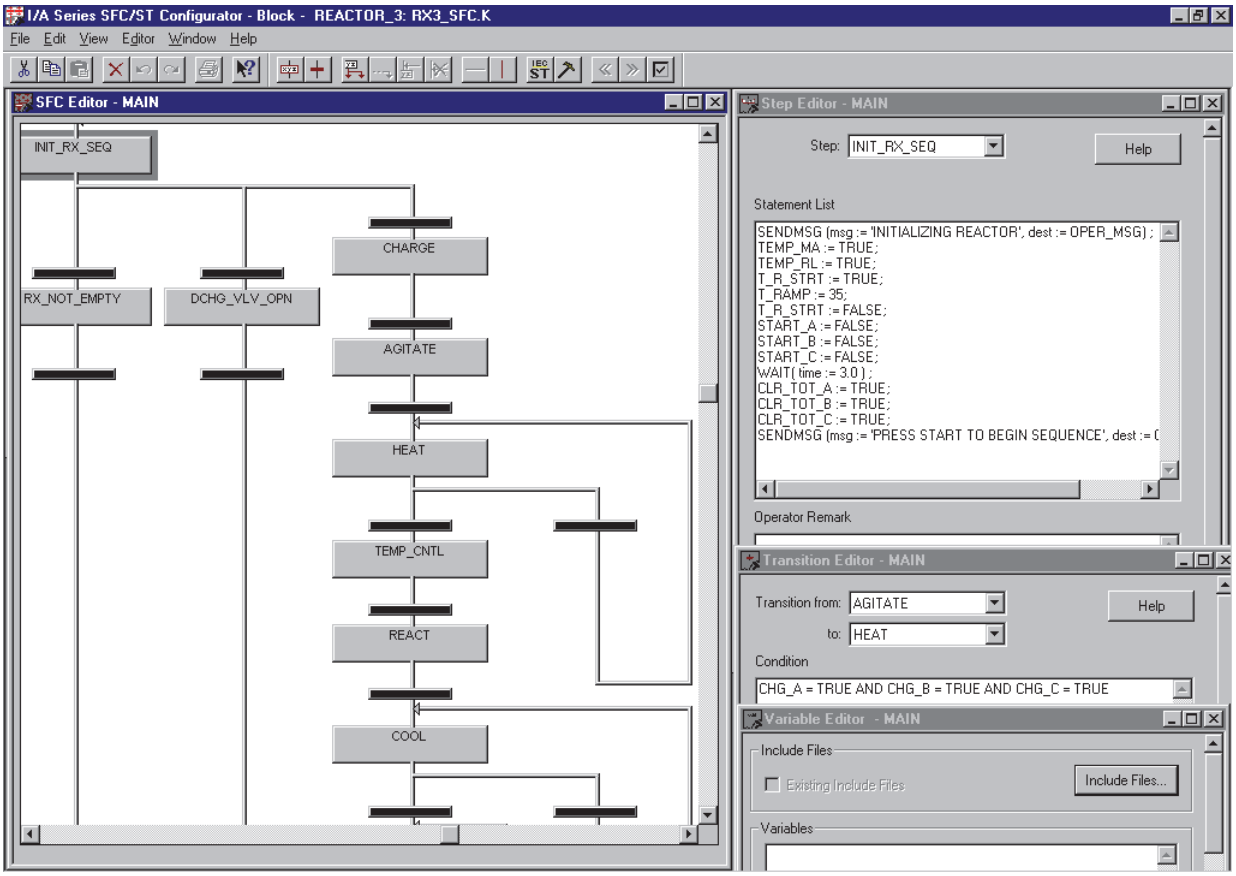


Figure 1. SFC/ST Configurator Windows for SFC, Step, Transition and Variable Editors

SFC Network

An SFC network consists of steps and transitions interconnected by directed links. Each transition has an associated transition condition. The transition must be true before processing proceeds to the following step. The steps run serially. Each transition can have only one preceding step and only one successor step.

Sequences may have up to 100 parallel alternative steps/transitions in the graphical structures and a maximum of 127 steps. The functional content of these steps is restricted only by the memory available for the sequence block, which is up to 32 KB for each block, equating to several thousand lines of code.

Step Specification with Structured Text

Within a step, the user can define ST statements with operator remarks (optional). The system executes each statement once during execution of the step. The supported statement types are:

- Assignment
- Subprogram control
- Selection
- Empty.

NOTE

Within steps, the system does not support iteration statements such as FOR, WHILE, and REPEAT.

Transition Specification with Structured Text

A transition represents the condition whereby control passes from one or more steps preceding the transition to one or more successor steps along the corresponding directed link. Each transition has an associated transition condition that is the result of the evaluation of a single Boolean expression.

Operator Remarks

Within a step's statement list or a transition's condition, the operator can insert remarks.

Function Blocks

The SFC/ST function blocks:

- Perform operations on data (procedural)
- Alter the flow of control
- Manipulate the elements of an array.

The following function blocks are available:

Procedural	Flow Control	Array
ABORT	EXIT	MULT_ARRAY
ACTCASES	RETRY	SET_ARRAY
ACTIVATE	WAIT	SUM_ARRAY
BIT_PATTERN		
OVERWRITE_STR		
SENDCONF		
SENDMSG		
SET_SBXS		
START_TIMER		
STOP_TIMER		
TSENDCONF		

An array function block can manipulate all elements of an array in one stroke. It cannot manipulate the elements of an external array (FPN array).

Functions

The SFC/ST function types are:

- Transfer
- Arithmetic
- Concatenation
- String extraction.

The following functions are available:.

Transfer	Arithmetic	Concatenation	String Extraction
ORD	ABS	CONCAT_STR	MID
ROUND	SQRT		
TRUNC			

Subroutines

An SFC/ST sequence block subroutine is an SFC network that is callable from the statements of the other SFC networks in the same block. Recursive subroutines are not valid, that is, subroutines cannot call themselves, directly or indirectly.

The user can define a subroutine with input/output parameters to repeat the same action on different sets of parameters.

The three SFC/ST sequence block types support subroutines. Each block supports a maximum of 128 subroutines. The user defines the subroutine structure or network using a sequential function chart.

Standard Block Exception (SBX) Handler

SBXs allow the user to specify logic for execution when a particular event occurs during block processing. Unlike subroutines, the user cannot invoke SBXs from ST statements. The SBX names are:

- TO_SYS_ERROR
- TO_USR_ERROR
- TO_INACTIVE
- TO_MANUAL
- TO_PAUSED.

There are five events for which the user can define an SBX. The name of an SBX is one from the predefined set, and it specifies to which event an SBX is related. This event is either an error (user error or system error) or a change of the block's state to inactive, manual or paused. Each SBX has one SFC network.

NOTE

SBXs do not support input/output parameters.

Macros

A sequence of tokens (replacement text), called the body of the macro, is associated with the macro name (see Figure 2). When the name of the macro is recognized in the program source text, the system treats it as a call to that macro and effectively replaces the name with a copy of the body.

Constants

Constants are identifiers which act as synonyms to values. The operator defines constants with the Constant Editor before using them in statements. Constants are known throughout the whole block, including statements in the main routine, subroutines, and SBXs. Comments and operator remarks for constants are optional.

Variables

Variables provide a means of identifying data objects whose contents may change, for example, data associated with a:

- Compound or block parameter
- Shared variable
- Block or subroutine local variable
- Input/output parameter.

Comments and operator remarks for variables are optional.

User Labels

Each SFC/ST sequence block has a number of parameters that are available to the user. These user parameters also have standard names. The operator can attach a specific name (a so-called user label) to them (see Figure 3). Comments and operator remarks for user labels are optional.

File Inclusion

There are three types of file inclusion:

- Inclusion of data declarations and definitions

  This file inclusion makes it easy to handle collections of #defines and declarations. An included data declaration or definition may itself contain #include lines. #include is the preferred way to tie the declarations together for a large configuration. This type of inclusion is allowed within the macro definitions and within the declarations of Constants, Variables and User Labels.

- Inclusion of main routines, subroutines, and SBX routines

  This file inclusion makes it easy to re-use routines in multiple sequence blocks. An included main routine, subroutine, or SBX routine may not contain another routine inclusion.

- Inclusion of a whole sequence block

  This file inclusion makes it easy to re-use whole blocks. An included block may include the main routine, subroutines, and SBX routines, but may not include a whole block again.

Both the Block and Library Configurator provide functions to handle all these types of file inclusion, except for creating and editing include files containing data declaration and definitions. They can be created and edited with any text editor. Any source line of the form #include “file_name” is replaced by the contents of the file “file_name”. Naturally, when an included file is changed, all blocks that depend on it must be recompiled.

SFCDM PROTECTION CLASS SETTING

To prevent a group of users from performing certain operations, the SFC/ST Configurator enables the user to set a protection class for operation buttons, options lists, and menu commands of the SFC/ST Display Manager (SFC DM). See Figure 4 and Figure 5.

During startup, the SFC DM enables or disables these graphical objects, depending on the access level setting of the current FoxView environment. For example, if the user sets a protection class of 77 for the block mode, a user group with access level 77 cannot select any of the block mode options.

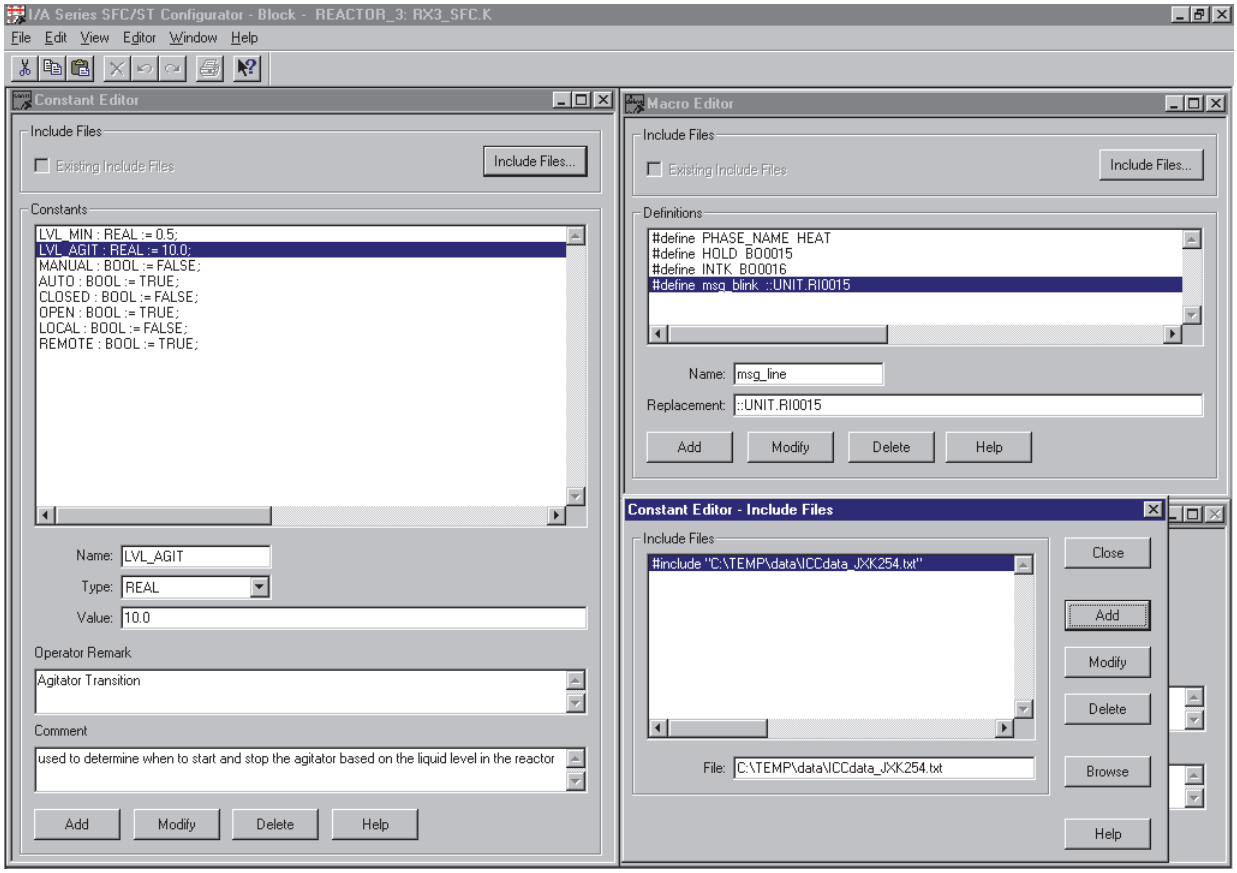


Figure 2. SFC/ST Configurator Windows for Constant and Macro Editors

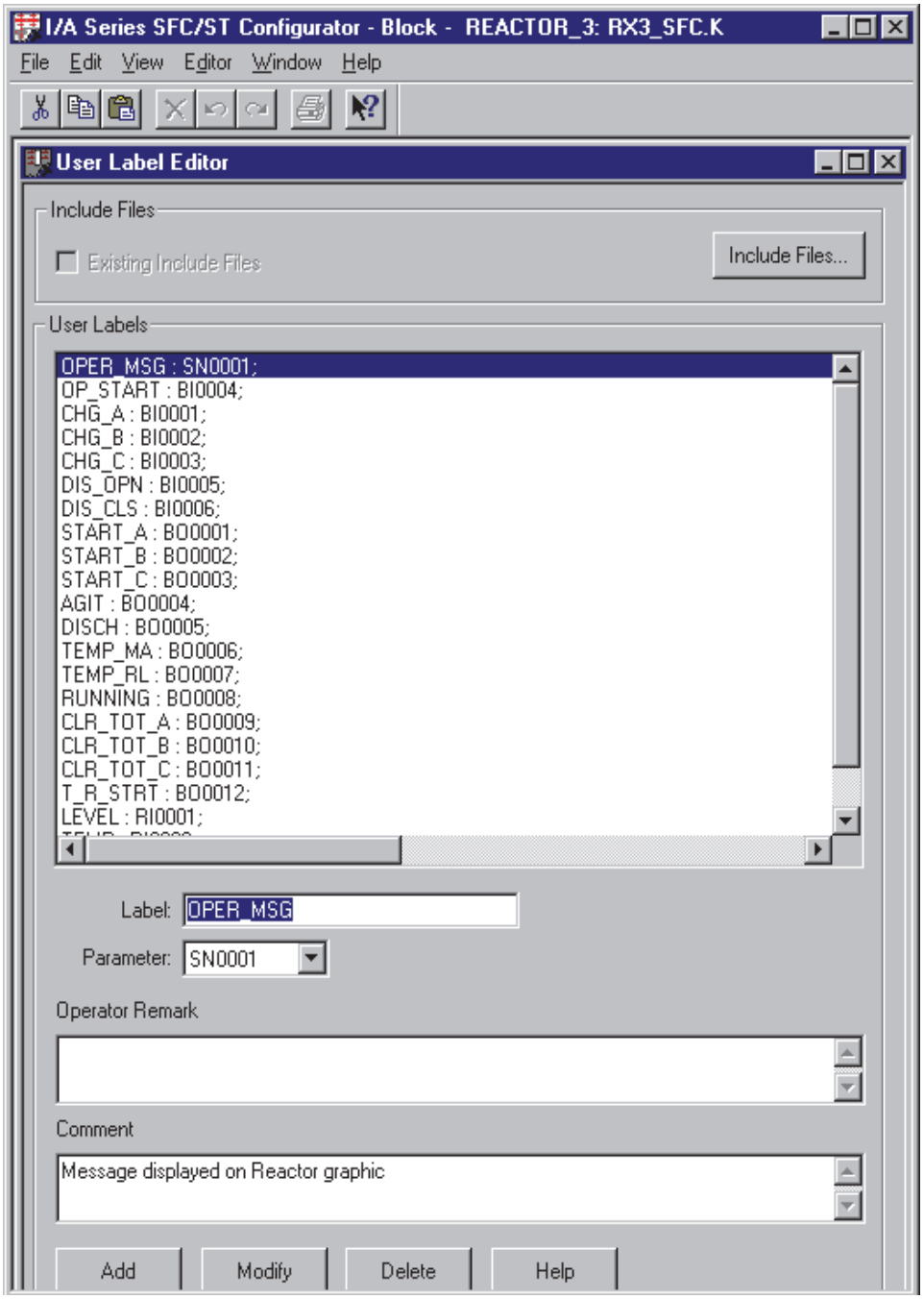


Figure 3. SFC/ST Configurator Window for User Label Editor

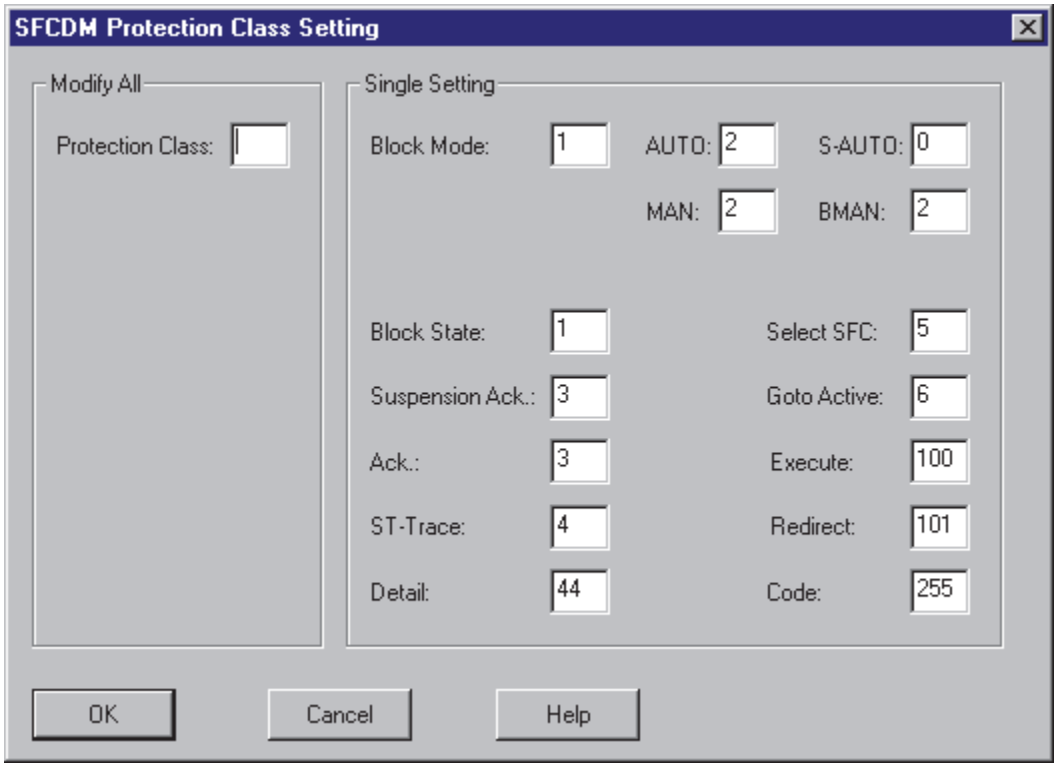


Figure 4. SFCDM Protection Class Setting Dialog Box

SFC/ST DISPLAY MANAGER FUNCTIONS

The SFC/ST Display Manager (see Figure 5) enables the user to view the following sequence block data during run time:

- SFCs of the main routine, subroutines, and SBXs with highlighting of the active step or transition(s)
- ST statements of steps
- Transition conditions with periodically updated operand values (graphical or text form)
- Constant definitions
- User label definitions with periodically updated values
- Variable definitions
- Operator remarks for any SFC/ST element
- ST-Trace information, for example, the currently active ST statement of a step
- Block state (ACTIVE or INACTIVE)
- Block Paused state (DEP blocks only)
- Block Suspend state
- Block mode (AUTO, S-AUTO, MAN, BMAN)
- Block errors.

The user can perform the following operations via the SFC/ST Display Manager:

- Block state changes (ACTIVE, INACTIVE)
- Block mode changes (AUTO, S-AUTO, MAN, BMAN)

- Stepping through the steps and transitions of the main routine in the S-AUTO and MAN modes
- Redirection of the control flow in the main routine in the MAN mode
- Tracing through the ST code of steps and transitions (one statement at a time) in BMAN mode
- Suspension Acknowledgement for resuming execution of the block
- Acknowledgement of all current block alarms.

Protection class settings lock access to operation buttons, options lists, and menu commands to prevent certain users from performing one or more of the above operations.

Locking of graphical objects depends on the access level setting of the current FoxView™ environment. A user group with the access level specified by the protection class settings cannot select the objects

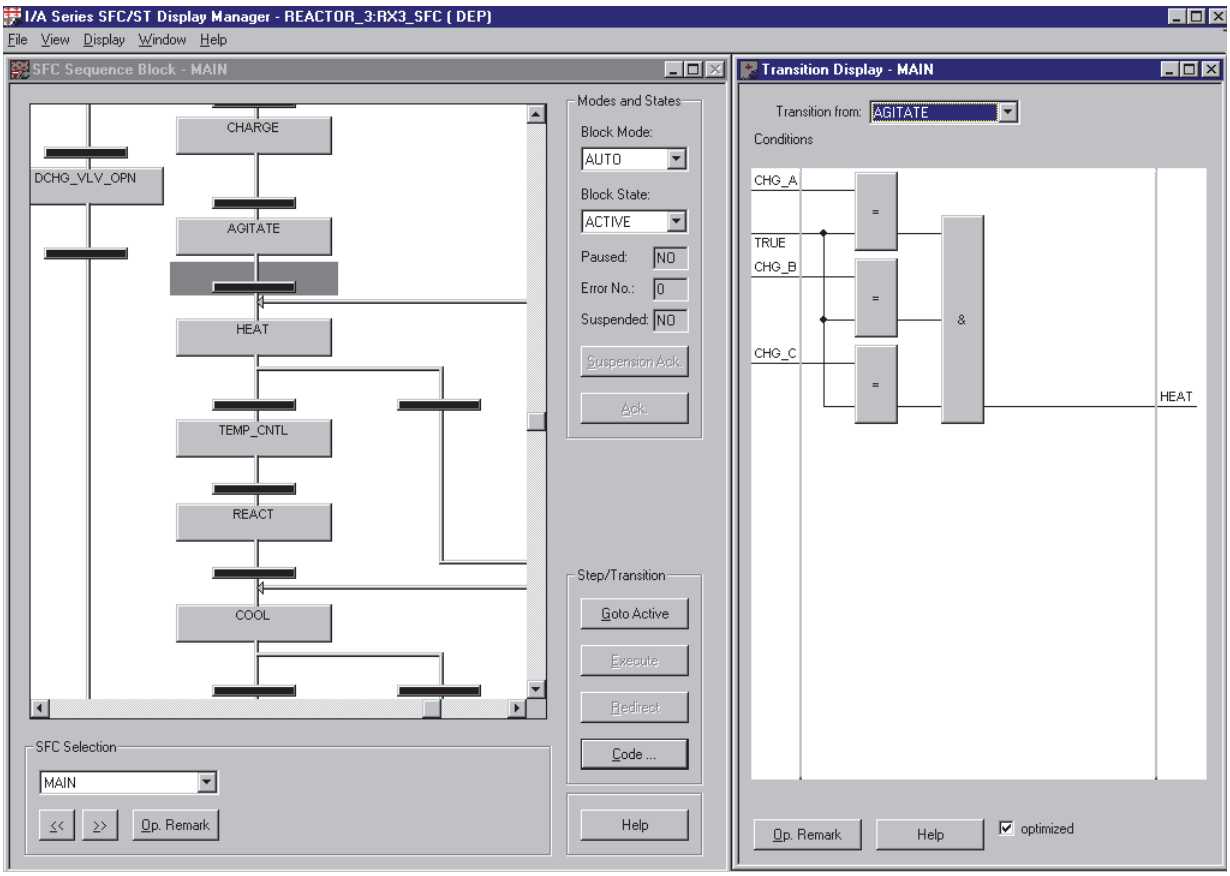


Figure 5. SFC/ST Display Manager Main Window

GRAPHICAL DISPLAY TRANSITIONS

Transitions between steps are displayed graphically as function block diagrams (FBD) or as Boolean expressions (ST). In the graphical FBD mode, different colors are used to indicate whether the partial results are TRUE or FALSE, as shown in Figure 6.

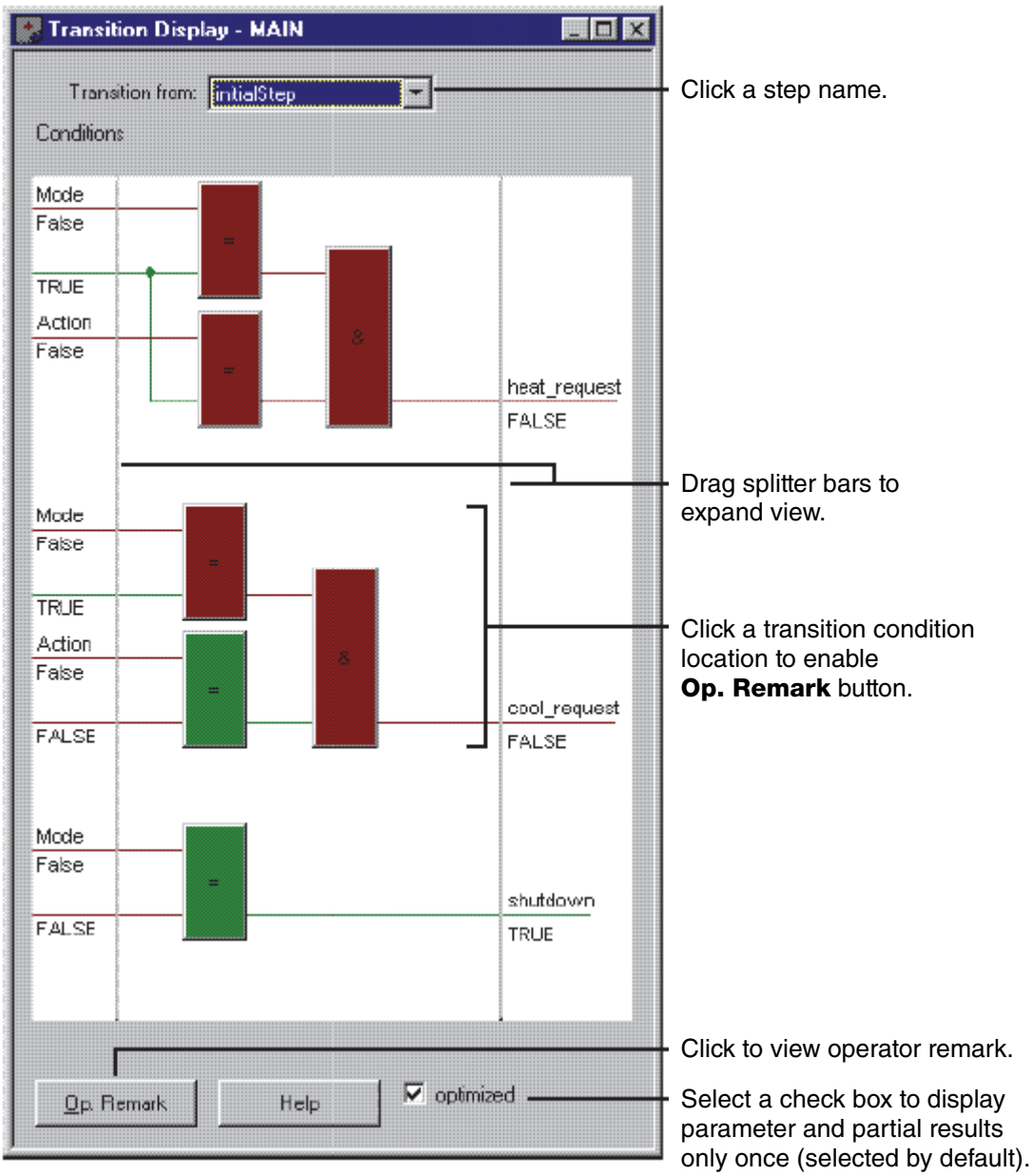


Figure 6. Graphical Transition Display Window with Active Transitions

FUNCTIONAL SPECIFICATIONS

Foxboro Evo Platforms Supported
Windows Foxboro Evo or I/A Series workstations and Solaris I/A Series workstations at I/A Series software v6.2 to v8.8 or Foxboro Evo Control Core Services v9.0 or later.

Maximum Number of Steps/Transitions in the Graphical Structures of a Sequence
100 parallel alternative steps/transitions

Functional Content of Steps
Restricted only by the memory available for the sequence block, which is up to 32 KB for each block, equating to several thousand lines of code.

Maximum Number of Subroutines in a Sequence
128

Foxboro®
by Schneider Electric

Invensys Systems, Inc
10900 Equity Drive
Houston, TX 77041
United States of America
http://www.invensys.com

Global Customer Support
Inside U.S.: 1-866-746-6477
Outside U.S.: 1-508-549-2424
Website: https://support.ips.invensys.com

MB 031 1214