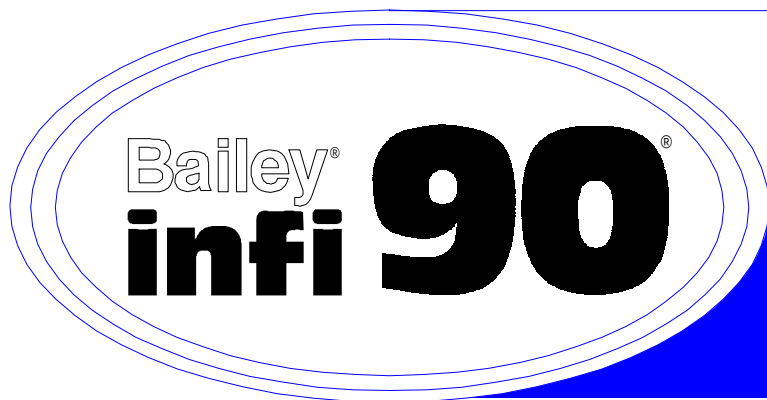
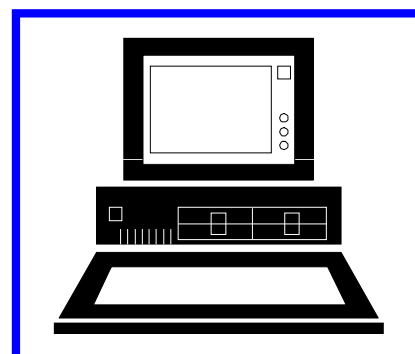
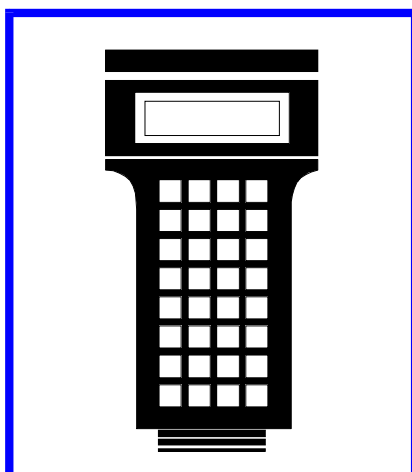
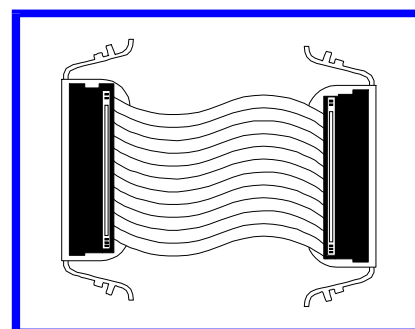
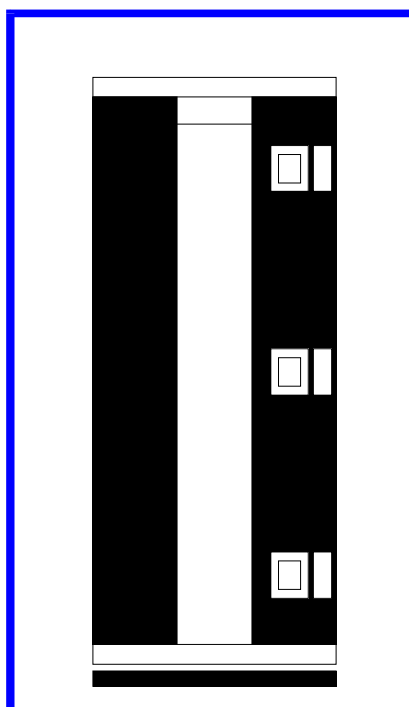
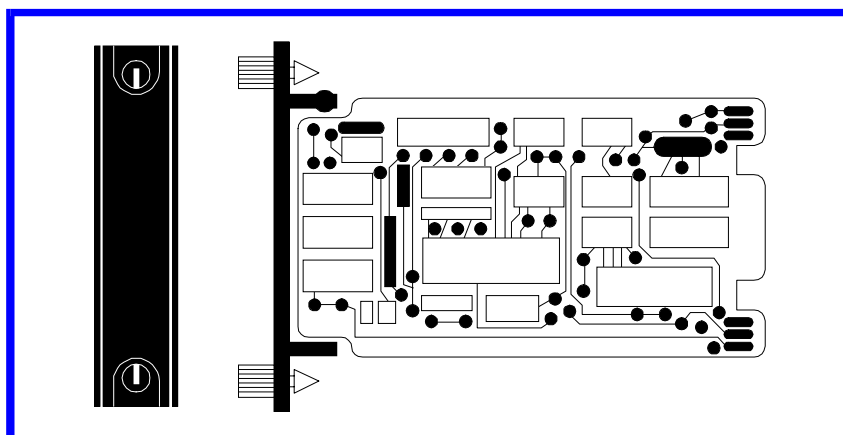
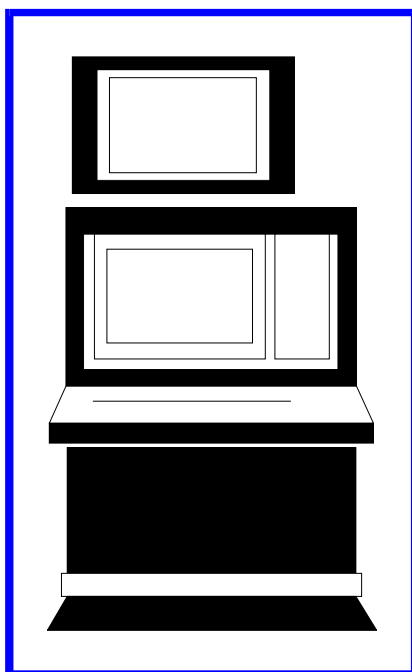




WCAD01

Instruction

Module Configuration Tools (Release 2.0)



WARNING notices as used in this instruction apply to hazards or unsafe practices that could result in personal injury or death.

CAUTION notices apply to hazards or unsafe practices that could result in property damage.

NOTES highlight procedures and contain information that assists the operator in understanding the information contained in this instruction.

WARNING

INSTRUCTION MANUALS

DO NOT INSTALL, MAINTAIN, OR OPERATE THIS EQUIPMENT WITHOUT READING, UNDERSTANDING, AND FOLLOWING THE PROPER **Elsag Bailey** INSTRUCTIONS AND MANUALS; OTHERWISE, INJURY OR DAMAGE MAY RESULT.

RADIO FREQUENCY INTERFERENCE

MOST ELECTRONIC EQUIPMENT IS INFLUENCED BY RADIO FREQUENCY INTERFERENCE (RFI). CAUTION SHOULD BE EXERCISED WITH REGARD TO THE USE OF PORTABLE COMMUNICATIONS EQUIPMENT IN THE AREA AROUND SUCH EQUIPMENT. PRUDENT PRACTICE DICTATES THAT SIGNS SHOULD BE POSTED IN THE VICINITY OF THE EQUIPMENT CAUTIONING AGAINST THE USE OF PORTABLE COMMUNICATIONS EQUIPMENT.

POSSIBLE PROCESS UPSETS

MAINTENANCE MUST BE PERFORMED ONLY BY QUALIFIED PERSONNEL AND ONLY AFTER SECURING EQUIPMENT CONTROLLED BY THIS PRODUCT. ADJUSTING OR REMOVING THIS PRODUCT WHILE IT IS IN THE SYSTEM MAY UPSET THE PROCESS BEING CONTROLLED. SOME PROCESS UPSETS MAY CAUSE INJURY OR DAMAGE.

NOTICE

The information contained in this document is subject to change without notice.

Elsag Bailey, its affiliates, employees, and agents, and the authors and contributors to this publication specifically disclaim all liabilities and warranties, express and implied (including warranties of merchantability and fitness for a particular purpose), for the accuracy, currency, completeness, and/or reliability of the information contained herein and/or for the fitness for any particular use and/or for the performance of any material and/or equipment selected in whole or part with the user of/ or in reliance upon information contained herein. Selection of materials and/or equipment is at the sole risk of the user of this publication.

This document contains proprietary information of Elsag Bailey, Elsag Bailey Process Automation, and is issued in strict confidence. Its use, or reproduction for use, for the reverse engineering, development or manufacture of hardware or software described herein is prohibited. No part of this document may be photocopied or reproduced without the prior written consent of Elsag Bailey.

Preface

This instruction describes the utilities provided with the Module Configuration Tools WCAD01 2.0 release. These tools provide the user with the ability to perform the following:

- Design module configurations
- Load, save, and verify configurations.
- Configuration maintenance.
- Monitor, trend, and, tune configurations.

The module tools user is required to have a thorough familiarity with personal computer operation and formal training in configuring control strategies. This product release maintains the functionality of the Module Configuration Tools WCAD01 1.1 release and provides the following enhancements:

- Certified to run under Microsoft® Windows NT® as well as Windows® 3.1 and Windows 95.
- Windowed replacements for the TXTEWS, print, trend, tune, and GSLV (global save, load, and verify) tools.
- A function code editor with filtering capabilities.
- Simultaneous multi-tasking INFI-NET® communications support.
- CLIF (configure logic interface function) support.
- New software license management.

List of Effective Pages

Total number of pages in this instruction is 282, consisting of the following:

Page No.	Change Date
Preface	Original
List of Effective Pages	Original
iii through xii	Original
1-1 through 1-7	Original
2-1 through 2-13	Original
3-1 through 3-19	Original
4-1 through 4-18	Original
5-1 through 5-63	Original
6-1 through 6-6	Original
7-1 through 7-50	Original
8-1 through 8-23	Original
9-1 through 9-16	Original
10-1 through 10-13	Original
11-1 through 11-23	Original
12-1 through 12-5	Original
A-1	Original
B-1 through B-8	Original
C-1 through C-2	Original
Index-1 through Index-3	Original

When an update is received, insert the latest changed pages and dispose of the superseded pages.

NOTE: On an update page, the changed text or table is indicated by a vertical bar in the outer margin of the page adjacent to the changed area. A changed figure is indicated by a vertical bar in the outer margin next to the figure caption. The date the update was prepared will appear beside the page number.

Table of Contents

	<i>Page</i>
SECTION 1 - INTRODUCTION	1-1
OVERVIEW	1-1
INTENDED USER	1-1
FEATURES	1-1
TOOLS	1-2
INSTRUCTION CONTENT	1-3
HOW TO USE THIS INSTRUCTION	1-4
DOCUMENT CONVENTIONS	1-4
GLOSSARY OF TERMS AND ABBREVIATIONS	1-5
REFERENCE DOCUMENTS	1-6
SUPPORTED PRODUCTS	1-6
COMPATIBILITY	1-7
SPECIFICATIONS	1-7
 SECTION 2 - INSTALLATION	 2-1
INTRODUCTION	2-1
HARDWARE REQUIREMENTS	2-1
SOFTWARE REQUIREMENTS	2-2
SOFTWARE INSTALLATION	2-2
SOFTWARE RELEASE INFORMATION	2-4
SOFTWARE REGISTRATION	2-5
SentinelSuperPro Parallel Port Key Installation	2-6
Parallel Port Driver Installation	2-6
Installing the Software License	2-7
Software Product Operation	2-8
SOFTWARE LICENSING MANAGEMENT	2-9
CLIF INSTALLATION	2-11
 SECTION 3 - USING THE MODULE TOOLS	 3-1
INTRODUCTION	3-1
CONFIGURING A CONTROL STRATEGY	3-1
APPLICATION LAUNCHER	3-2
MODULE TOOLS EXECUTIVE	3-3
Starting the Module Tools Executive	3-3
Opening a Project	3-4
Parts of the Module Tools Executive Screen	3-5
Using the Project Tree	3-6
Project Tree Structure	3-7
Module Tools Executive Menus	3-9
File Menu	3-9
Project Menu	3-10
Cabinet Menu	3-10
Folder Menu	3-11
Loop Menu	3-11
Pcu Menu	3-12
Module Menu	3-12
Utilities Menu	3-13
Help Menu	3-13
ON-LINE HELP	3-14
SINGLE POINT TAG DATA ENTRY AND MODULE CONFIGURATION	3-14
MODULE CONFIGURATION PROCEDURE	3-15
SUPPLEMENTAL CONFIGURATION UTILITIES	3-16

Table of Contents (continued)

	<i>Page</i>
SECTION 3 - USING THE MODULE TOOLS (continued)	
Supplemental On-Line Utilities	3-17
Supplemental Off-Line Utilities	3-17
Supplemental DOS Utilities	3-18
WINDOWS CONVENTIONS	3-18
SECTION 4 - MANAGING PROJECTS	4-1
INTRODUCTION	4-1
PROJECT DIRECTORY STRUCTURE	4-1
PROJECT OPERATIONS	4-1
Defining a Project	4-2
Opening a Project	4-4
Closing a Project	4-4
EDITING PROJECT INFORMATION	4-4
EDITING JOB INFORMATION	4-5
DEFINING THE PROJECT STRUCTURE	4-5
Adding Loops	4-5
Adding Documentation Cabinets	4-6
Adding Documentation Folders	4-6
Adding PCU Cabinets	4-7
Adding Modules	4-7
Editing Module Information	4-9
Adding Sheets	4-9
Importing Configuration Sheets	4-10
Importing Standard Drawings	4-12
Importing Documentation Sheets	4-12
Importing an Existing Module	4-13
DELETING LOOPS, CABINETS, PCUS, AND MODULES	4-15
DELETING SHEETS	4-15
MANAGING LIBRARIES	4-15
Adding a Library	4-16
Compressing a Library	4-17
Deleting a Library	4-17
UPGRADING THE PROJECT TDT DATABASE	4-18
SECTION 5 - GRAPHICAL MODULE CONFIGURATION	5-1
INTRODUCTION	5-1
GMC APPLICATION FEATURES	5-2
MODULE CONFIGURATION OVERVIEW	5-2
General Procedure	5-2
General Notes	5-3
Module Configuration and Tag Assignment	5-3
Operating Modes	5-5
STARTING THE GMC APPLICATION	5-5
CLOSING THE GMC APPLICATION	5-6
GMC APPLICATION WINDOW	5-7
Parts of the GMC Application Window	5-7
Menu Items	5-9
On-Line Help	5-10
DOCUMENT OPERATIONS	5-10
Opening a Document	5-10
Opening a Cross-Referenced Sheet	5-11

Table of Contents (continued)

	<i>Page</i>
SECTION 5 - GRAPHICAL MODULE CONFIGURATION (continued)	
Creating a New Document	5-11
Saving a Document	5-11
Printing a Document	5-12
Revising a Sheet	5-12
Properties	5-13
Closing a Document	5-13
DOCUMENT WINDOWS	5-13
Working with Multiple Document Windows	5-14
Sizing, Moving, and Arranging Document Windows	5-15
WORKING IN THE DOCUMENT WORKSPACE	5-16
Choosing Commands	5-16
Using the Tool Bar	5-17
View Options.....	5-18
Environment.....	5-19
Magnifying and Panning.....	5-21
Magnifying.....	5-21
Panning.....	5-22
Using the Cursor.....	5-22
Using the Grid	5-23
Measuring Distance	5-24
Libraries	5-25
Properties	5-25
Layers.....	5-25
MODULE CONFIGURATION COMPONENTS	5-26
PLACING AND DRAWING ENTITIES	5-27
ARC	5-28
BORDER.....	5-28
CIRCLE	5-30
CONNECTION POINT	5-30
CONSTANT BLOCK	5-30
FUNCTION CODE.....	5-31
IO CONNECTOR.....	5-33
LINE	5-37
MACRO	5-38
RECTANGLE	5-39
REFERENCE CONNECTOR	5-40
SHAPE	5-42
SIGNAL.....	5-43
TEXT.....	5-44
DEFINING TAG AND TREND POINTS	5-45
SELECTING ENTITIES.....	5-45
EDITING ENTITIES	5-47
Adding and Modifying Entity Attributes	5-47
Assigning Tags	5-49
Changing Block Numbers.....	5-50
Changing Cross-Reference Data	5-50
Changing Layers	5-50
Changing Line Style	5-51
Cutting, Copying, and Pasting Entities	5-51
Deleting Entities	5-52
Deleting Entity Attributes.....	5-52
Deleting Tag Assignments	5-52
Exploding Shapes	5-53

Table of Contents (continued)

	<i>Page</i>
SECTION 5 - GRAPHICAL MODULE CONFIGURATION (continued)	
Modifying Specifications	5-53
Modifying Text	5-53
Moving Entities.....	5-54
Moving Signal Segments	5-54
Repeating the Last Command	5-54
Rotating Entities.....	5-55
Scaling Entities	5-55
Selection Arrow	5-56
Undoing and Redoing Operations.....	5-56
FUNCTION CODE EDITOR.....	5-56
Function Code Edit Window.....	5-56
Controls	5-58
Viewing/Filtering Options	5-58
Editing Specifications	5-59
USING LIBRARIES	5-59
Assigning a User Library to a Sheet	5-60
Deleting Items From a User Library.....	5-60
Forming Shapes, Macros, and Borders	5-60
USING A STANDARD DRAWING.....	5-62
MONITORING A CONFIGURATION	5-63
 SECTION 6 - COMPILE, CROSS-REFERENCE, AND BIND	 6-1
INTRODUCTION	6-1
COMPILER	6-1
CROSS-REFERENCER.....	6-1
BINDER.....	6-2
STARTING THE COMPILER, CROSS-REFERENCER, AND BINDER	6-2
COMPILER RESULTS.....	6-4
CROSS-REFERENCER RESULTS	6-4
BINDER RESULTS	6-5
 SECTION 7 - CONFIGURATION MAINTENANCE	 7-1
INTRODUCTION	7-1
STARTING THE CMU APPLICATION	7-1
EXITING THE APPLICATION	7-1
APPLICATION WINDOW	7-1
Parts of the Application Window.....	7-2
CMU Application Menus	7-4
Context Menus	7-5
Choosing Commands.....	7-5
On-Line Help	7-7
WINDOW OPERATIONS	7-7
Project Window.....	7-8
Project View	7-8
Configuration View	7-11
Multiple Document Windows	7-15
Sizing, Moving, and Arranging Windows	7-15
Function Code Edit Window.....	7-16
PROPERTIES	7-16
General	7-17
Project View.....	7-17

Table of Contents (continued)

	<i>Page</i>
SECTION 7 - CONFIGURATION MAINTENANCE (continued)	
Configuration View.....	7-19
Reports	7-20
PROJECT OPERATIONS	7-21
Project Tree Structure	7-22
Opening a Project	7-23
Closing a Project	7-23
Creating a New Project	7-23
Viewing Options	7-24
Refreshing the Project View	7-24
Editing Project Information	7-24
Editing the JHD File	7-25
PCU OPERATIONS	7-25
MODULE OPERATIONS.....	7-25
Module Hardware Configuration.....	7-26
Editing Module Header (MHD)	7-26
Add Module	7-27
Module Reports.....	7-28
Inspect/Mode Change	7-28
On-Line Configuration	7-28
HARDWARE OPERATIONS (PRIMARY AND BACKUP).....	7-28
View Configuration.....	7-28
Editing and Listing Configurations	7-29
Download Configuration File	7-33
Save Module Configuration	7-34
CONFIGURATION OPERATIONS.....	7-35
View Configuration.....	7-35
Filter/View.....	7-35
Go To Block Number	7-36
Editing and Listing Configurations	7-36
Saving a Configuration	7-38
Edit File Header (CFG)	7-38
CAD SHEET OPERATIONS	7-39
View Configuration.....	7-39
Filter/View.....	7-40
Go To Block Number	7-40
List Configuration	7-41
FILE SYSTEM OPERATIONS.....	7-41
MONITOR/TREND OPERATIONS	7-41
VERIFY OPERATIONS	7-42
REPORT OPERATIONS	7-43
Calculating Configuration Utilization	7-44
Trace	7-45
Specification Listing	7-45
Specification Dump.....	7-45
FUNCTION CODE EDITOR	7-47
Function Code Edit Window	7-47
Controls.....	7-48
Viewing/Filtering Options	7-49
Editing Specifications.....	7-50
UTILITIES	7-50

Table of Contents (continued)

	<i>Page</i>
SECTION 8 - INSPECT	8-1
INTRODUCTION	8-1
ENHANCED STATUS REPORTING MODULES	8-1
STARTING THE APPLICATION	8-1
EXITING THE APPLICATION	8-2
APPLICATION WINDOW	8-2
Parts of the Application Window	8-2
Menu Items	8-4
On-Line Help	8-4
Tool Bar	8-4
View Options	8-5
WINDOW OPERATIONS	8-5
INSPECT OPERATIONS	8-5
PCU Address	8-7
View Tabs	8-7
Preferences	8-8
ON-LINE CONFIGURATION	8-9
On-Line Configuration Procedure A	8-11
On-Line Configuration Procedure B	8-15
MODULE MODE OPERATIONS	8-20
REPORT OPERATIONS	8-21
 SECTION 9 - SAVE, LOAD, VERIFY	 9-1
INTRODUCTION	9-1
SLV UTILITY	9-1
Starting the SLV Utility	9-2
Parts of the Application Window	9-2
View Operations	9-3
Script File Operations	9-4
Script Properties	9-5
SLV SCRIPT GENERATOR	9-5
SLV FILE UTILITIES	9-6
SLV SCRIPT COMMANDS	9-7
Delay	9-8
Load	9-8
Path	9-9
Save	9-9
Set	9-10
Target	9-12
V	9-13
Verify	9-13
Script File Example	9-14
 SECTION 10 - TREND, MONITOR, AND TUNE	 10-1
INTRODUCTION	10-1
STARTING THE APPLICATION	10-1
EXITING THE APPLICATION	10-1
APPLICATION WINDOW	10-2
Parts of the Application Window	10-2
Menu Items	10-3
On-Line Help	10-4
Tool Bar	10-4

Table of Contents (continued)

	<i>Page</i>
SECTION 10 - TREND, MONITOR, AND TUNE (continued)	
WINDOW OPERATIONS	10-4
GROUP OPERATIONS	10-5
Opening a Group	10-5
Creating a New Group	10-5
Editing a Group Item	10-6
Saving a Group	10-7
Graph Display Options	10-7
MONITOR AND TREND A GROUP	10-8
TUNE A POINT	10-10
FUNCTION CODE EDITOR	10-10
Function Code Edit Window	10-11
Controls	10-12
Viewing/Filtering Options	10-12
Editing Specifications	10-13
SECTION 11 - SUPPLEMENTAL CONFIGURATION UTILITIES	11-1
INTRODUCTION	11-1
TAG SYNCHRONIZATION	11-1
PRINTING CAD SHEETS	11-3
Print Style Setup	11-3
Printing	11-6
Print Queue Manager	11-7
VIEWING LOG FILES	11-8
CONFIGURING WINTOOLS COMMUNICATIONS	11-9
DSOE CONFIGURATION	11-10
DSOE Configuration Overview	11-11
Configure External Drawings	11-11
Configure SEM Executive Block	11-12
Create Point List From CAD Files	11-14
Configure DSOE Point List	11-15
Update CAD Files from Point List	11-17
Configure Complex Trigger List	11-17
Compile SEM Module Configuration	11-19
Display DSOE Configuration	11-19
Display Configured Points, Recordable Points, Alarm Points and Simple Triggers	11-19
Display Complex Triggers	11-21
Print DSOE Configuration	11-22
SECTION 12 - DOS UTILITIES	12-1
INTRODUCTION	12-1
PASSWORD SECURITY	12-1
UPGRADING 4.x FILES	12-2
APPENDIX A - FILE NAME EXTENSIONS	A-1
EXTENSIONS	A-1
APPENDIX B - SYSTEM LIBRARY	B-1
INTRODUCTION	B-1
SYSTEM SYMBOLS	B-1

Table of Contents (continued)

	<i>Page</i>
APPENDIX C - SHEET NAMING TABLES	C-1
INTRODUCTION	C-1
NUMBERING UP TO 1296 SHEETS	C-1
NUMBERING MODULES.....	C-1
NUMBERING UP TO 360 SHEETS OR 250 PCUS AND LOOPS	C-2

List of Figures

<i>No.</i>	<i>Title</i>	<i>Page</i>
1-1.	Module Tools	1-2
2-1.	Example License Installation Screen	2-7
2-2.	CLIF Installation.....	2-12
3-1.	Configuration Process Overview	3-1
3-2.	Application Launcher	3-2
3-3.	Bailey WinTools Group	3-3
3-4.	Module Tools Executive	3-5
4-1.	Project Directory Structure	4-2
5-1.	GMC Application Window	5-8
5-2.	GMC Tool Bar	5-18
5-3.	Function Code Connections	5-36
7-1.	CMU Application Window	7-2
7-2.	Project View	7-9
7-3.	Configuration View	7-12
8-1.	Inspect Application Window	8-3
9-1.	SLV Window	9-3
10-1.	Trend, Monitor, and Tune Application Window	10-2
11-1.	Print Style	11-5
11-2.	Print Queue Manager.....	11-8
11-3.	DSOE Configuration Utilities Menu.....	11-13
11-4.	Sample SEM Executive Block Configuration	11-13
11-5.	Sample Configure DSOE Point List	11-16
11-6.	Sample Complex Trigger Definition	11-18
11-7.	Display DSOE Configuration Menu	11-20
11-8.	Print DSOE Configuration Screen	11-23
12-1.	Upgrade Screen	12-3
B-1.	System Library Symbols (1 of 4)	B-5
B-2.	System Library Symbols (2 of 4)	B-6
B-3.	System Library Symbols (3 of 4)	B-7
B-4.	System Library Symbols (4 of 4)	B-8

List of Tables

<i>No.</i>	<i>Title</i>	<i>Page</i>
1-1.	Glossary of Terms and Abbreviations	1-5
1-2.	Reference Documents	1-6
1-3.	Supported Products	1-6
1-4.	WCAD01 Specifications	1-7
3-1.	Windows Actions.....	3-18
3-2.	Windows Dialog Box Buttons	3-19
4-1.	Job Information	4-3
4-2.	Module Information.....	4-8
5-1.	Sheet Properties	5-13
5-2.	Function Code Specifications	5-57
7-1.	Job Information	7-23
7-2.	Module Information.....	7-27
7-3.	Add Function Code	7-32
7-4.	Copy Function Code	7-33
7-5.	Add Function Code	7-37
7-6.	CFG File Header Information	7-39
7-7.	SDF File Structure	7-46
7-8.	Function Code Specifications	7-48
8-1.	Inspect Fields.....	8-5
8-2.	Report Options.....	8-22
9-1.	Script Properties	9-5
10-1.	Point Definition	10-6
10-2.	Trend Graph Properties	10-7
10-3.	Group Monitor Table	10-8
10-4.	Status Indicators	10-9
10-5.	Sort Order.....	10-9
10-6.	Function Code Specifications	10-11
11-1.	Print Style Pages	11-5
11-2.	Communication Support	11-9
11-3.	SEM Executive Block Fields	11-14
11-4.	SOECHANDEF Attributes.....	11-16
11-5.	Cursor Movements	11-16
11-6.	Boolean Trigger Operands	11-18
11-7.	Display DSOE Configuration Menu Selections	11-20
11-8.	Screen Movements	11-21
11-9.	Display Complex Triggers Field Descriptions	11-21
12-1.	Upgrade Error Messages	12-4
A-1.	File Descriptions	A-1
B-1.	System Library.....	B-1
C-1.	1296 Sheet Numbering	C-1
C-2.	Module Number Translation.....	C-2
C-3.	Sheet and PCU Translation	C-2

Trademarks and Registrations

Registrations and trademarks used in this document include:

[™] Digital	Trademark of Digital Equipment Corporation.
® HP	Registered trademark of Hewlett-Packard Company.
® INFI 90	Registered trademark of Elsag Bailey Process Automation.
® INFI-NET	Registered trademark of Elsag Bailey Process Automation.
® Microsoft	Registered trademark of Microsoft Corporation.
® MS-DOS	Registered trademark of Microsoft Corporation.
® Network 90	Registered trademark of Elsag Bailey Process Automation.
® PCV	Registered trademark of Elsag Bailey Process Automation.
® Pentium	Registered trademark of Intel Corporation.
® PostScript	Registered trademark of Adobe Systems, Incorporated.
[™] SentinelSuperPro	Trademark of Rainbow Technologies.
® Windows	Registered trademark of Microsoft Corporation.
® Windows NT	Registered trademark of Microsoft Corporation.

SECTION 1 - INTRODUCTION

OVERVIEW

The Module Configuration Tools WCAD01 are used to configure and maintain the module configurations in an INFI 90® OPEN Strategic Enterprise Management System. The tools perform the following engineering tasks:

- Project and module configuration management.
- Module configuration design and documentation.
- Inspect module operation.
- Load, save, and verify configurations.
- On-line process monitoring, trending, and tuning.
- On-line configuration of MFC, MFP, and BRC modules.
- Maintain configurations within the modules.

The tools are installed on an engineering work station and accessed from an application called the module tools executive running under the Microsoft® Windows® operating environment. The module tools executive is included with the software release. This instruction discusses operation of the module tools executive and operation of each of the module tools.

Figure 1-1 shows the module tools primary functions in relation to an INFI-NET® network and the console tools. The network interface unit allows the work station to communicate with any module in any process control unit and across multiple loops of the network.

INTENDED USER

Proficiency with personal computer operation (MS-DOS® and Windows) and formal INFI 90 OPEN system configuration training are required before using the module tools.

FEATURES

Features of the module tools include:

- User-friendly graphical interface.
- Project-oriented approach to module configuration design and maintenance.
- Configure and document INFI 90 OPEN process control strategy graphically using a system library containing function codes, symbols, and borders.
- Full INFI-NET network communications support.

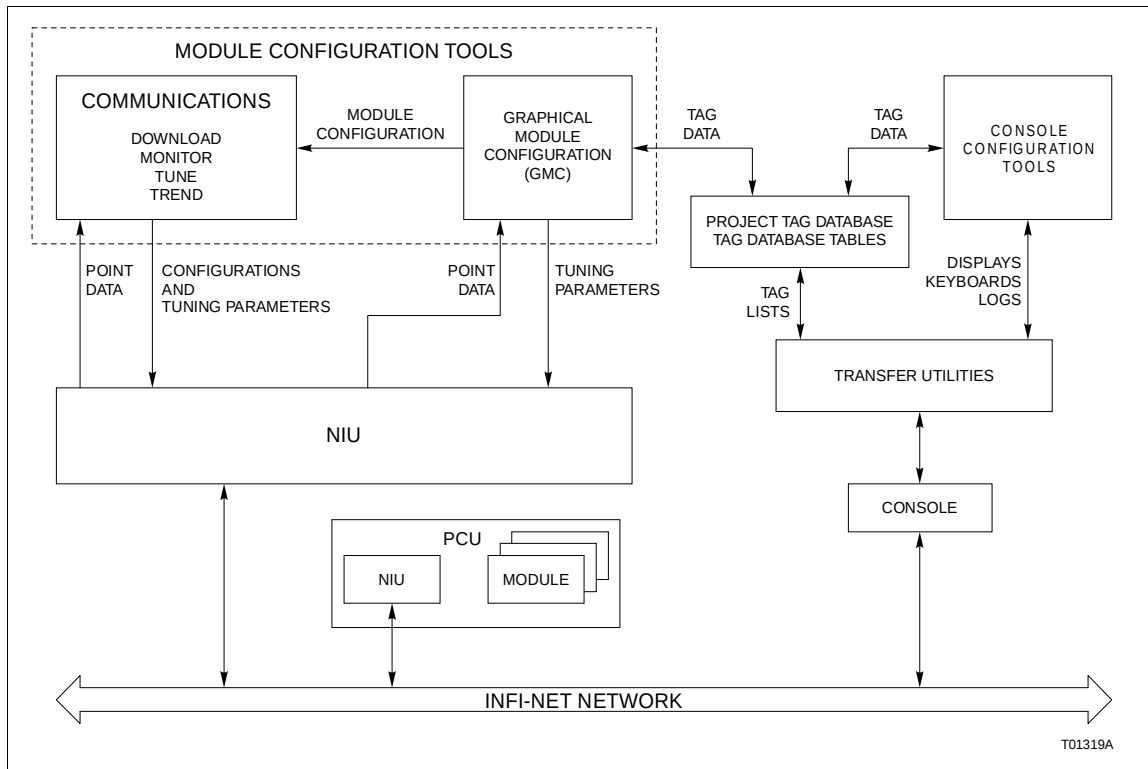


Figure 1-1. Module Tools

- Load, save, and verify configurations individually or by groups.
- On-line monitoring and tuning of function blocks.
- Single point tag data entry within the tools.
- On-line help for utilities running under the Windows operating environment.

TOOLS

The following paragraphs provide a brief description of each primary tool included with the module tools software release.

Module Tools Executive

Defines, organizes, and selects projects for configuration and provides a menu system for the module tools. In addition, provides utilities to manage projects and to batch print configuration drawings.

Graphical Module Configuration (GMC)

Graphically configures modules by inserting and connecting graphical representations of function codes. It also provides symbol support and documentation functions. The GMC tool generates one or more sheets (**CAD** files) that make up a module configuration drawing.

Cross-Referencer	Performs a documentation function that resolves input and output references in the sheets. It does this across modules, PCUs, and loops.
Compiler	Scans module configuration drawing files (CAD) and generates a configuration file (CFG) that can be downloaded to the target module.
Binder	Automatically resolves input function code parameters to corresponding output function codes.
Configuration Maintenance Utility (CMU)	Displays function block contents and configures function blocks in configuration files and module-resident configurations. Generates reports on configuration files including spare and configured blocks in a module and memory, block usage, cycle time, and special function codes used in a configuration.
Inspect	Inspects module status and operation within a PCU node, changes module modes, and controls on-line configuration of MFC, MFP, and BRC controller modules.
Save, Load, Verify	Saves module-resident configurations to disk files, downloads configurations from disk files to modules, and verifies configuration files.
Trend	Monitors and trends any function block output and tunes specifications.
Supplemental Utilities	Provides configuration print support, TDT table synchronization, DSOE configuration, communications configuration, log file viewing, SCAD 4.x upgrade support, and password protection for specified utilities.

INSTRUCTION CONTENT

This instruction contains information to operate the module tools. It includes a Table of Contents, List of Figures, List of Tables, and Index giving several options to locate specific information quickly. The following sections are in this instruction:

Introduction	Provides a general overview of the module tools and this instruction.
Installation	Describes requirements, how to install the module tools, and software licensing.
Using the Module Tools	Describes the operation of the module tools, how to prepare the tools for operation, and provides a general procedure for using the tools. It also describes the module tools executive operation.
Managing Projects	Provides information on defining and opening projects, adding modules to a project, and deleting modules from a project.

Graphical Module Configuration	Provides information on configuring modules using the graphical module configuration (GMC) application.
Compile, Cross Reference, and Bind	Describes the compiler, cross-referencer, and binder utilities.
Configuration Maintenance	Describes the configuration maintenance utility (CMU) operations.
Inspect	Describes the inspect application.
Save, Load, Verify	Describes features of the SLV application.
Trend, Monitor, and Tune	Describes the trend application.
Supplemental Configuration Utilities	Describes the TDT synchronization, communication configuration, print, and DSOE configuration utilities.
DOS Utilities	Covers operation of supplemental DOS-based utilities.
Appendices	Appendices provide supplementary information on the following: • File type descriptions. • Standard symbols and borders in the system library. • Reference tables for converting between decimal numbers and two-character representation for use in CAD sheet names.

HOW TO USE THIS INSTRUCTION

Read Section 1 for an overview of the module tools and this instruction. Proceed to Section 2 for system requirements and procedures for installation. Refer to Section 3 for general information on using the module tools and the module tools executive. Refer to Section 4 to create and define the project structure. Refer to Sections 5 through 12 for detailed operation of each tool. Refer to the Windows documentation for information on using the Windows operating environment and graphical user interface.

DOCUMENT CONVENTIONS

This document uses standard text conventions to represent keys, display items, file names, and user input.

Key Identifies a keyboard key.

Example: Press **Enter**.

Key- Key Means to press and hold the first key, then press the second key.

Example: Press **Ctrl- F8**.

Key Key Means to press and release the first key, then press the second key.

Example: Press **F S**.

Display item Any item that displays on the screen appears as italic text.

Example: *Configure Display*

File name File names and file extensions appear as bold-italic text.

Example: ***EUDSCP.CF***

BOLD User input that must be entered exactly as shown.

italic User input that is a variable parameter; user supplies actual term or value.

Example: **ADAP -cpath\CONSOLE.CFG**

GLOSSARY OF TERMS AND ABBREVIATIONS

Table 1-1 provides a glossary of terms and abbreviations used in this instruction. It contains those terms and abbreviations that are unique to Elsag Bailey or have a definition that is different from standard industry usage.

NOTE: Refer to the Windows documentation for a description of Windows specific terms.

Table 1-1. Glossary of Terms and Abbreviations

Term	Definition
EWS	Engineering work station.
Exception Report	Information update generated when the status or value of a point changes by more than a specified significant amount; abbreviated as XR.
Executive Block	Fixed function block that determines overall module operating characteristics.
FC	Function code.
Function Block	The occurrence of a function code at a block address of a module.
Function Code	An algorithm which manipulates specific functions. These functions are linked together to form the control strategy.
GES	General edit system.
INFI-NET	Advanced data communication highway.

Table 1-1. Glossary of Terms and Abbreviations (continued)

Term	Definition
Loop	1. A data communication network with a ring topology. 2. That portion of an analog process control loop that resides within the controller and typically consists of: an analog input measuring the process variable, a manual/auto station generating a set point, a PID control algorithm and an analog output driving a final control element.
Module Address	A unique identifier of a specific device or a communication channel. Refers to Controlway or module bus address.
NIU	Network Interface Unit. Term for all local and remote interfaces, computer interfaces, and console interfaces to the INFI-NET communication system.
Node	A point of interconnection to a network.
Plant Loop	Network 90® data communication highway.
Tag	An analog or digital process value defined in the console or PCV® as a control or monitor point.

REFERENCE DOCUMENTS

Table 1-2 lists reference documents.

Table 1-2. Reference Documents

Number	Title
I-E96-199-90	Configured Logic Interface Function (CLIF)
I-E96-217	IMSPM01 Serial Port Module
I-E96-221	IMCPM02 Communications Port Module
I-E96-602	INFI-NET to Plant Loop Gateway (INIPL01)
I-E96-610	INFI-NET to Computer Interfaces (INICI01 and INICI03)
I-E96-620	Plant Loop to Computer Interface (INPCI01)
I-E96-621	Plant Loop to Computer Interface (INPCI02)
WBPEEU1210005B0	Function Code Application Manual
WBPEEU1310010B0	Console Configuration Tools (WLDG)
WBPEEU1350254A0	Application Programming Interface (semAPI) NT

SUPPORTED PRODUCTS

The module tools support the products listed in Table 1-3.

Table 1-3. Supported Products

Type	Product
Communication highways	INFI-NET Communication Highway Plant Loop Process Communication Highway
Controllers	Command Series Controllers Network 90 Controllers INFI 90 OPEN Controllers

Table 1-3. Supported Products (continued)

Type	Product
Interfaces	Computer Interface Command Network 90 Computer Interfaces INFI 90 OPEN Computer Interfaces semAPI

COMPATIBILITY

The module tools WCAD01 release 2.0 file format follows SCAD 5.4 release conventions. Existing WCAD01 release 1.0/1.1 and SCAD 5.x files can be imported into WCAD01 release 2.0. In addition, utilities are provided to upgrade files from revision SCAD 4.x.

NOTES:

1. The structure of the TDT database is release dependent, reflecting changes needed, for example, as tag types and fields are added. The TDT database must be of the same revision level as the configuration tools used. After installing WCAD01 release 2.0 and prior to using it for configuration, a TDT database from an existing project must be upgraded for use with release 2.0. This can be done from within WCAD01 release 2.0 or WLDG01 release 2.0.
2. Furthermore, to preserve database compatibility, the module and console tools must be of the same revision level for use with any given project. Release WCAD01 1.0/1.1 and WLDG01 release 1.0/1.1 must be used together or WCAD01 release 2.0 and WLDG01 release 2.0 must be used together.

SPECIFICATIONS

Table 1-4 lists specifications for the WCAD01 2.0 release.

Table 1-4. WCAD01 Specifications

Specification	Value
Drawing information per sheet	30,000 ¹ bytes
Function code specification size	3,000 ¹ bytes
Maximum sheets per module	1,296
Number of attributes	750 ¹
Number of loops	250
Number of PCUs per loop	250
Number of active modules per PCU	32

NOTE: 1. Maintains compatibility with SCAD 5.4.

SECTION 2 - INSTALLATION

INTRODUCTION

This section provides the following installation procedures:

- Installing the module configuration tools WCAD01.
- Installing the product license registration.
- Installing CLIF function codes.

NOTE: Integral to the project scheme of configurations maintained by the console and module configuration tools is the project TDT database (described in the **Console Configuration Tools (WLDG)** instruction). It contains tag data intended to be used and maintained by both the console tools and the module tools. The structure of the TDT database is release dependent, reflecting changes needed, for example, as tag types and fields are added. This has the following consequences:

1. The configuration tools used to configure consoles and modules within a project must be of the same release revision. WLDG01 release 1.0/1.1 and WCAD01 release 1.0/1.1 are used together and WLDG01 release 2.0 and WCAD01 release 2.0 are used together. They cannot be intermixed for use within the same project.

2. The TDT database must be of the same revision level as the configuration tools used. After installing WCAD01 release 2.0 and prior to using it for configuration, a TDT database from an existing project must be upgraded for use with release 2.0. This can be done from within WLDG01 release 2.0 or WCAD01 release 2.0. Once the TDT database is upgraded it can no longer be used with the earlier release configuration tools.

HARDWARE REQUIREMENTS

The following hardware is required to run the software:

- A certified computer system or one meeting the minimum configuration requirements.
- A SentinelSuperPro™ parallel port key (included with the software product).
- In order to configuration sheets, a Windows compatible printer is required.
- In order to communicate with modules, the engineering work station must be connected to a network interface unit. This typically requires a dedicated serial port on the computer. Refer to **CONFIGURING WINTOOLS COMMUNICATIONS** in Section 11 for information.

Certified Computer Systems

Computer systems certified for use with WinTools are:

- Digital™ Celebris 486 (66 MHz 80486).
- Digital Celebris 590 (90 MHz Pentium®).
- Digital Venturis 5100 (100 MHz Pentium).

Minimum System Configuration

System performance is directly dependent on the hardware configuration. Increasing the amount of memory, using an advanced processor, and using a graphics accelerator will increase system performance. Computer systems not certified must meet the following minimum configuration:

- 80486 processor.
- 16 megabytes memory.
- 100 megabytes free hard disk space.
- 3.5-inch high density floppy disk drive.
- 640×480, 64 color graphics display and adapter board.
- Windows-compatible mouse or other pointing device.
- One parallel printer port.

SOFTWARE REQUIREMENTS

One of the following software configurations is required in order to install and run the software:

- MS-DOS 5.x or later and one of the following versions of Windows: Windows 3.1, Windows 3.11, or Windows for Workgroups 3.11.
- Windows 95.
- Windows NT 3.51 or 4.0.

SOFTWARE INSTALLATION

This procedure installs the tools software. The installation directory is C:\EWS.

NOTES:

1. This version of the tools uses a new software licensing and copy protection system. If installing this version over a previous version (release 1.0 or 1.1) of the tools, the software key for the previous version should be removed before continuing. This must be done if the release 1.0 or 1.1 software key is to be saved. Use the original installation disk 1 from the previous version and refer to the previous version instruction for the software key removal procedure.
2. The installation process deletes files from an existing C:\EWS directory. If there are any user files in an existing C:\EWS directory that need to be saved, back them up before continuing.

3. After software installation, software registration must be completed to properly run the configuration tools. Refer to **SOFTWARE REGISTRATION** for information.

1. Insert disk 1 into the appropriate floppy disk drive.
2. Using any available Windows method, run the installation program **install.exe** located on disk 1.
3. An installation dialog box appears with installation options. Choose *Install*. The installation process begins; as it proceeds, the percentage completed is displayed by the installation program.
4. Follow the directions to insert each installation disk when requested. Other messages may appear on the screen.
5. After the software has been installed, a dialog box appears with options to change the engineering work station configuration as needed for the tools. Choose one of the following options to continue.

Go ahead and modify - for Windows 3.1 and Windows 95, modifies the **autoexec.bat** and **config.sys** files. For Windows NT, modifies the registry.

Create example files - for Windows 3.1 and Windows 95, writes example files to the root directory (**autoexec.exm** and **config.exm**). These files can be examined and the appropriate changes made to the **autoexec.bat** and **config.sys** files manually as needed. For Windows NT, writes the file **WCAD.REG** to the C:\EWS directory. This file can be used to update the registry by using the **regedit.exe** utility.

Bypass - continues without making any changes.

NOTE: If the configurations files are not modified correctly, problems running the utilities could result.

6. At the next dialog box, choose an option to install the tools icons.

Install icon into existing group - for Windows 3.1/NT 3.51, installs the tools icons into an existing group. When installed, the icon can be double clicked to launch the application. For Windows 95/NT 4.0, the tools programs are installed into an existing menu and folder.

Install icon into Bailey Wintools group - for Windows 3.1/NT 3.51, installs the tools icons into the *Bailey Wintools* group. When installed, the icon can be double clicked to launch the

application. For Windows 95/NT 4.0, the tools programs are installed into a *Bailey WinTools* menu and folder.

Skip this part - continues without making any changes. The tools icons must be installed manually or the tools run manually (e.g., from file manager or explorer).

7. A message appears noting that the license installation utility must be run and that parallel port drivers for the SentinelSuperPro parallel port key may need installed. This material is covered in **SOFTWARE REGISTRATION**. Click on OK to continue.

8. The installation dialog box appears. Choose *Exit* to continue.

9. If changes have been made to the configuration files, exit Windows and reboot the computer.

10. After restarting Windows, examine the **WCAD.WRI** file for release notes. If the tools icons were installed as in Step 6, double click on the *WCAD Readme* icon in the *Bailey WinTools* group (Windows 3.1/NT 3.51) or select *WCAD Readme* from the *Bailey WinTools* menu (Windows 95/NT 4.0) to open the file.

SOFTWARE RELEASE INFORMATION

The following information applies to this software release:

Compatible WLDG Release	For a given project, the console configuration tools WLDG01 release 2.0 must be used with this release of the module tools.
Registration	The software must be registered for proper operation. Refer to SOFTWARE REGISTRATION .
Project Definition	Projects defined in WCAD01 release 1.0/1.1 are compatible with WCAD01 release 2.0.
TDT Database	WCAD01 release 1.0/1.1 TDT database must be upgraded. Refer to UPGRADING THE PROJECT TDT DATABASE in Section 4.
Module Configurations	Module configurations from early versions of the module configurations can be imported or upgraded for use with WCAD01 release 2.0.
Communications Configuration	The engineering work station can with communicate with modules over the INFI-NET network. Refer to CONFIGURING WIN-TOOLS COMMUNICATIONS in Section 11 to set up communications parameters.

CLIF Function Codes Any SCAD 5.4 installed CLIF function codes must be re-installed to add them to the updated function code database. Refer to **CLIF INSTALLATION**.

SOFTWARE REGISTRATION

For proper operation of protected features or products, the software must be registered. This section describes how to register a single license for use on a single engineering work station.

1. Read this procedure completely and the information given in **SOFTWARE LICENSING MANAGEMENT**.
2. Locate and have available the following:
 - The registration number of the software license to be registered. This is located on the registration form shipped with the software product.
 - The instructions for obtaining a software license code. These instructions came with the software registration form.
 - The SentinelSuperPro parallel port key to be used with this software license. This can be the SentinelSuperPro parallel port key included with this or any Elsas Bailey software product.

The parallel port key provides a machine identifier that is required in order to obtain the software license code. The machine identifier is printed on the parallel port key and is displayed in the license installation dialog box.

NOTE: Although a parallel port key is shipped with each software product, only one parallel port key is required for each engineering work station. That is, a single parallel port key can be used to register more than one software product. Other configurations are supported; refer to **SOFTWARE LICENSING MANAGEMENT** for more information.

3. Obtain the software license code. Refer to the registration form instructions that came with the software product.
4. Install the SentinelSuperPro parallel port key. Refer to **SentinelSuperPro Parallel Port Key Installation**.
5. Install the parallel port drivers if needed. Refer to **Parallel Port Driver Installation**.
6. Install the software license. Refer to **Installing the Software License**.

SentinelSuperPro Parallel Port Key Installation

1. Shut down the computer and turn off power.
2. Locate the computer's LPT1 parallel port and disconnect any existing parallel port cable.

NOTE: It is possible to have more than one parallel port key installed on LPT1 to support other products. They can be installed in any order.

3. Install the SentinelSuperPro parallel port key onto LPT1.
4. The parallel port key is a feed through device. Reconnect any cable removed in Step 2 to the parallel port key.
5. Restart the computer.

Parallel Port Driver Installation

During software installation, drivers for the SentinelSuperPro parallel port key are copied into the WinTools product directory structure. These drivers may need to be installed, depending on the computer operating system and configuration.

1. Determine if the drivers need to be installed:

**Windows 3.1/3.11 or
Windows 95**

Installation of these drivers is optional and need to be installed only if conflicts occur between the SentinelSuperPro key and any additional parallel port device connected through this key.

Windows NT 3.51/4.0

Installation of these drivers is required. If these drivers were previously installed (e.g., as part of another product/revision installation), then installation at this time is not required. The installation must be performed from an administration authorized account on the NT work station computer.

2. Locate the drivers and readme file in the appropriate directory:

C:\EWS\RAINBOW\WIN_31

C:\EWS\RAINBOW\WIN_95

C:\EWS\RAINBOW\WIN_NT

3. Follow the instructions given in the readme file.
4. After installing the drivers, reboot the computer.

Installing the Software License

1. Start the license registration procedure by doing one of the following:

- Choose *License Install* from the Windows 95/NT 4.0 *Bailey WinTools* menu.
- Click on the license installation registration icon in the Windows 3.1/3.11 or Windows NT 3.51 *Bailey WinTools* group.

The license registration program executable is **licinst.exe** and is located in the C:\EWS directory.

2. The *Bailey Controls License Installation* dialog box appears as shown in Figure 2-1.

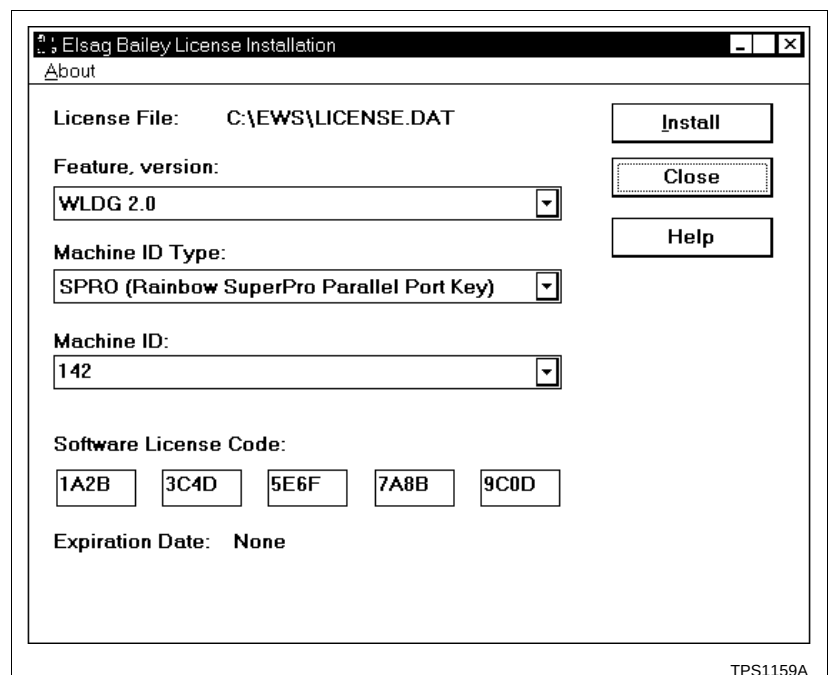


Figure 2-1. Example License Installation Screen

3. Insure that the product being registered appears in the *Feature, version* text box (e.g., **WLDG 2.0** or **WCAD 2.0**). If not, select it from the list. This list shows all installed products that require license installation. The list is updated when a product is installed during the software installation procedure.

4. Insure that the *Machine ID Type* text box displays *SPRO (Rainbow SuperPro Parallel Port Key)*. If not, select this option from the drop down list box.

There are three *Machine ID Type* options:

SPRO Normal license installation.

DEMO Use of the DEMO option is coordinated through the appropriate Elsag Bailey Sales/Marketing organization. This type of machine identifier license has an expiration date that must be entered.

MAINTENANCE SPRO Use of the MAINTENANCE SPRO option is coordinated through the Elsag Bailey Technical Services organization. This type of machine identifier license also has an expiration date that must be entered.

5. Insure that the machine identifier appears in the *Machine ID* text box. This should be the same machine identifier that is printed on the parallel port key that is to be used to register the software.

NOTES:

1. Normally, a machine will have a single parallel port key installed at any given time. However, if there is more than one parallel port key installed, insure the correct machine identifier appears in the *Machine ID* text box. If not, type it in or select it from the drop down list box.

2. If the parallel port key to be used is not installed at the time of registration, it can still be registered by typing in the corresponding machine identifier in the *Machine ID* text box.

6. Type the software license code into the software license text box as shown and tab out of the box. Refer to Step 3 of **SOFTWARE REGISTRATION**.

7. If all information is entered correctly, the *Install* button is enabled. Click on *Install* to install the software registration. A message box appears indicating the registration is successfully installed.

8. Record the software product, software registration number, machine identifier, and software license code for future reference.

Software Product Operation

This section describes the behavior of the software product protected by the license manager.

Normal Operation If the license is successfully installed, then the user will see no interaction with the license manager.

- Unlicensed Operation** If the software product is unlicensed or the license cannot be validated and depending on the feature, one of the following occurs:
- A message box appears informing the user that the product license cannot be validated. However, after clicking on *OK*, the feature will operate normally.
 - A message box appears informing the user that the product license cannot be validated. After clicking on *OK*, the feature will terminate.

SOFTWARE LICENSING MANAGEMENT

The license management system limits the use of a unique software license to a single instance at any given time. It also allows a great deal of flexibility for users in how this is implemented. When a license is registered, the machine identifier (provided by the parallel port key) and the software license code become a matched pair for that license. Both must reside on the engineering work station at the same time to enable the associated software license. A software license allows the protected software product to be used. Software licensing management has the following components:

License Manager A product that controls the usage of features, applications, or products by enforcing software license agreements. It uses a license file to track those machine identifiers that have the right to use certain features by storing software license codes paired to the machine identifiers. When a protected software product is run, it reads the machine identifier (from the parallel port key) and runs through the list of software license codes in the license file to find a match. If found, the software product is enabled.

Parallel Port Key The SentinelSuperPro parallel port key is a device that plugs into the computer's parallel port. A single parallel port key per engineering work station may be used with one or more products (e.g., WCAD01 and WLDG01).

Typically, a single parallel port key is used per engineering work station for all EWS products. The license installation program can access additional keys on the parallel port if necessary.

Machine Identifier The parallel port key has a unique machine identifier code as displayed in the *License Installation* dialog box. The user provides this machine identifier code as part of the registration process. The machine identifier code is paired with a software

license code during registration. When the software licensing management system sees a matched machine identifier and software license code on the engineering work station, it enables the associated software license.

Software License Code	The software license code is a unique code returned to the customer by Elsag Bailey Process Automation. It is entered by the customer into the <i>License Installation</i> dialog box during registration. It is a function of the registration number and the machine identifier code. The software license code is paired with the machine identifier. When the software licensing management system sees a matched machine identifier and software license code on the engineering work station, it enables the associated software license.
Registration Number	The registration number is a unique number provided with each purchased product. The user provides this number to Elsag Bailey Process Automation when registering a product in order to receive a software license code paired to the machine identifier for that product.
Post-Registration	Once a license is registered: • The software license code entered into the <i>License Installation</i> dialog box and the machine identifier provided by the parallel port key installed on the engineering work station must be a registered pair in order to run the protected software. This limits the use of a license to a single instance at any given time. • A single copy of the software can be installed on multiple engineering work stations. After registering and receiving the software license code, enter the software license code into the <i>License Installation</i> box of each copy. To use the protected software at a particular engineering work station, move the parallel port key providing the machine identifier paired with that software license code to the engineering work station on which the protected software is to be used. • More than one registration can be performed on a single copy of the software provided there are a sufficient number of purchased licenses and associated parallel port keys. This would be useful in the following circumstance: that configuration personnel are assigned a parallel port key to be kept with them. If multiple licenses are purchased and each of several engineering work stations has one or more registrations, then configuration personnel could run the configuration tools on any machine having the software license code paired with the machine identifier provided by their parallel port key by moving the parallel port key to that engineering work station. A copy of the software on an engineering work station can be registered more than once by running the <i>License Installation</i> application and

entering the software license code paired with the appropriate machine identifier provided by the parallel port key.

License Manager Files

The following files are used by the license manager and can be viewed by the user. They reside in the C:\EWS directory.

license.dat - a list of installed Elsasg Bailey products.

license.err - a list of messages displayed by the license manager.

CLIF INSTALLATION

A configured logic interface function code (CLIF) is an Elsasg Bailey engineered solution unique to a customer's application and is specifically licensed to that customer. CLIF function codes and associated documentation are distributed separately from the WCAD01 software product. CLIF function codes must be installed into the function code database being used by the project. Once installed, CLIF function codes are treated the same as any other function code by the module tools. CLIF function codes can be installed in the standard database located in the C:\EWS directory or a project specific database located in the project directory structure.

Function code numbers 232 through 240 are reserved for storing CLIF function codes. If more than nine CLIF function codes are needed, they can be defined to overwrite existing standard function codes. Standard function codes replaced by CLIF function codes cannot be used by any configurations using the modified function database. For information on CLIF function codes, refer to the **Configured Logic Interface Function (CLIF)** instruction.

This procedure installs CLIF function codes into a function code database. The CLIF function code source files must be located in a \CLIF directory either under C:\EWS or the project directory. During installation the CLIF source files from \CLIF are merged with the original standard function code database files. The merged function code database is stored either in the C:\EWS directory or a \FCDB directory under the project directory. If necessary, the CLIF function code source files are first copied from a distribution location to a \CLIF directory.

A project specific function code database should only be used if multiple projects with different CLIF function codes are being used on the same engineering work station.

The CLIF installation must be performed at the following times:

- If new CLIF function codes are to be added to a function code database.

- If installing a new release of the product, the new standard function code database is copied which overwrites any CLIF function codes installed in the standard function code database.
- If importing CAD sheets with CLIF function codes into a project that does not have the needed CLIF function codes installed.

NOTE: One of the CLIF distribution files has been modified from its previous SCAD 5.x file format (FNnnnnn.LBR has been changed and its new name is FNnnnnn.LB2). An updated file **must** be obtained from Bailey Controls prior to installing existing SCAD 5.x CLIF function codes in WCAD01 2.0. This new file contains only additional function code database definition information used by WinTools but in no way changes the existing SCAD 5.x executable function codes that have been downloaded into modules and validated.

To install CLIF function codes:

1. Start the CLIF installation procedure by doing one of the following:

**Windows 3.1 or
Windows NT 3.51**

- Click on the CLIF installation icon in the Bailey WinTools group.

**Windows 95 or
Windows NT 4.0**

- Choose *CLIF Install* from the *Bailey WinTools* menu.

The *CLIF Install* dialog box appears as shown in Figure 2-2.

2. The default location for installing CLIF function codes is the module tools product directory (C:\EWS). If a project directory is to be the target, select *WinTools Project*, then type in a project path and name (e.g., *c:\project1\project1.prj*) or use the browse button.

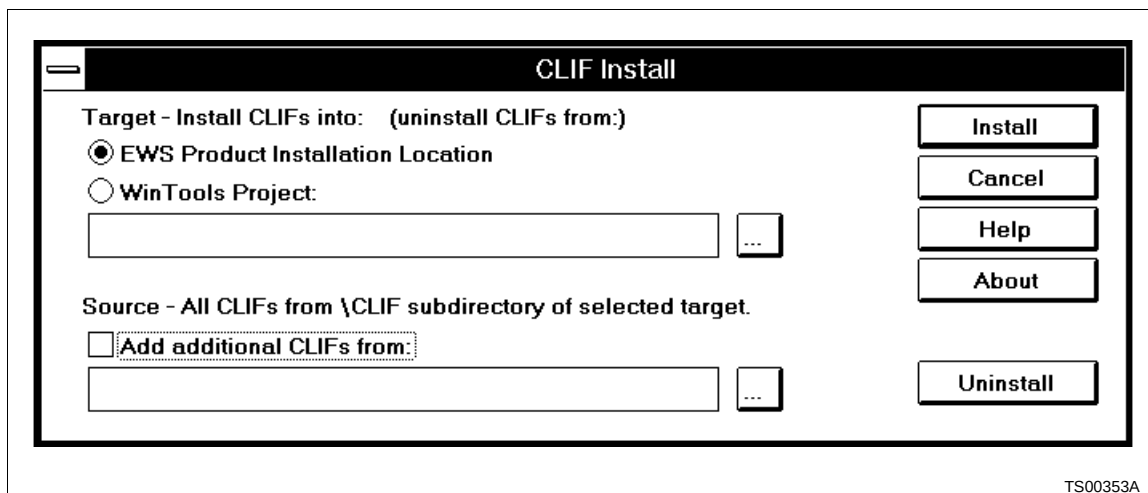


Figure 2-2. CLIF Installation

3. If new or updated CLIF function codes need to be copied to the C:\EWS\CLIF or *project*\CLIF directory, click on *Add additional CLIFs from:* and type in a path or use the browse button to select the distribution location. The additional CLIF function code files are copied to either the C:\EWS\CLIF or the *project*\CLIF directory, depending on the selected target.

4. Click on *Install* to continue. *Install* does the following:

- If *Add Additional CLIFs from:* is checked, copies all CLIF function codes found in the selected distribution location to either the C:\EWS\CLIF or the *project*\CLIF directory, depending on the selected target.
- Merges all CLIF function codes residing in the selected \CLIF directory with the original standard function code database files distributed with the product (C:\EWS\FCDB.ORG and associated **ORG** files).

The merged function code database files are stored in either the C:\EWS directory or the *project*\FCDB directory. Any symbols and faceplates distributed with the CLIF function codes are copied to either the C:\EWS\CLIF or *project*\CLIF directory.

NOTE: If CLIF function codes are installed into the C:\EWS directory, a copy of the standard function code database and associated files are preserved in **FCDB.ORG** and associated **ORG** files.

In operation the WinTools applications look for and use CLIF function code files in the project directory structure first; and if not found, in C:\EWS.

To un-install CLIF function codes:

1. Start the CLIF installation procedure by doing one of the following:

**Windows 3.1 or
Windows NT 3.51**

- Click on the CLIF installation icon in the Bailey WinTools group.

**Windows 95 or
Windows NT 4.0**

- Choose *CLIF Install* from the *Bailey WinTools* menu.

2. The *CLIF Install* dialog box appears as shown in Figure 2-2. Do one of the following:

- Select *EWS Product Installation Location* to restore the original function code database file into C:\EWS.
- Select *WinTools Project* to delete the project's \FCDB directory.

3. Click on *Uninstall* to continue.

SECTION 3 - USING THE MODULE TOOLS

INTRODUCTION

This section describes how to use the module tools. It contains the following information:

- An overview of the configuration process.
- A description of the application launcher.
- Operation of the module tools executive showing how the tools are accessed.
- The significance of single point tag data entry.
- A general procedure for performing module configuration.
- A description of the conventions used by the module tools within Windows.

CONFIGURING A CONTROL STRATEGY

The module tools provide module based control strategy development for an INFI 90 OPEN system. Figure 3-1 shows an overview of the configuration process which is described in the following paragraphs. Refer to **MODULE CONFIGURATION PROCEDURE** for a step-by-step procedure for configuring modules.

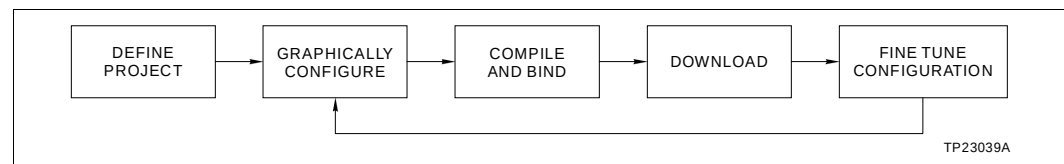


Figure 3-1. Configuration Process Overview

Define Project Identifies and organizes the loops, PCU cabinets, and modules within the project. Project specific information can also be entered for use by automatic borders in **CAD** sheets.

Graphically Configure Generates module configurations by dropping and connecting graphical representations of function code algorithms and other compilable entities onto configuration sheets. A module configuration consists of one or more of these sheets. Each sheet is a separate file and is the portion of a configuration that the user works on at one time. Each module has a module information file that defines module specific information.

Compile and Bind	Prepares the module configuration for downloading to the module by compiling and binding the configuration. Compiling scans the sheets for a particular module (CAD files) and generates a single configuration file (CFG) that can be loaded into the module. Binding resolves input function code parameters to corresponding output function codes for function codes that communicate across modules. Once the configuration is prepared for downloading it resides in a single file with a CFG extension.
Download	Communicates with modules in order to download configurations.
Fine Tuning	Communicates with modules in order to monitor and trend live system values and tune parameters to adjust the operation of the control strategy.

APPLICATION LAUNCHER

The application launcher provides the capability of launching the following module tools applications: the module tools executive, the graphical module configurator, the configuration maintenance utility, inspect, the monitor, trend, and tune utility, and the error log viewer. Figure 3-2 shows the application launcher window. Its appearance and menu contents depend on the installed utilities.

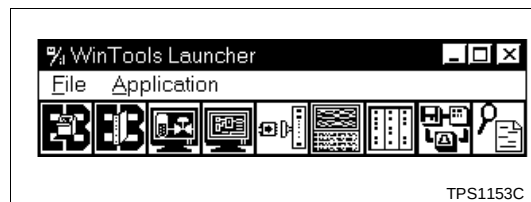


Figure 3-2. Application Launcher

To start the application launcher from Windows 3.1 or Windows NT 3.51, click on the *Toolset* icon in the *Bailey WinTools* program group. To start it from Windows 95/NT 4.0, click *Start* and choose *Toolset* from the *Bailey WinTools* menu under the *Programs* menu.

To make the application launcher stay on top of other applications, choose *Always On Top* from the *File* menu ([Alt] [F] [A]).

To exit the application launcher, choose *Exit* from the *File* menu ([Alt] [F] [X]).

Applications can be launched either by clicking on the appropriate button or choosing the application from the *Application* menu. The always-on-top state and location are saved between sessions.

MODULE TOOLS EXECUTIVE

The module tools executive is a Windows application that serves as a launching point for the module configuration tools. It also performs project management functions including opening and defining projects and defining the project structure. The project management functions are described in [Section 4](#).

Starting the Module Tools Executive

The module tools executive can be started by any of the available methods of the Windows operating environment. The following describes three common methods. Refer to the Windows documentation for other methods. Only one copy of the module tools executive can be running at a time.

From the Windows Program Manager

To start the module tools executive from Windows program manager (Windows 3.1 or Windows NT 3.51):

1. Open the *Bailey WinTools* group.
2. Double click on the module tools executive icon. Refer to [Figure 3-3](#).

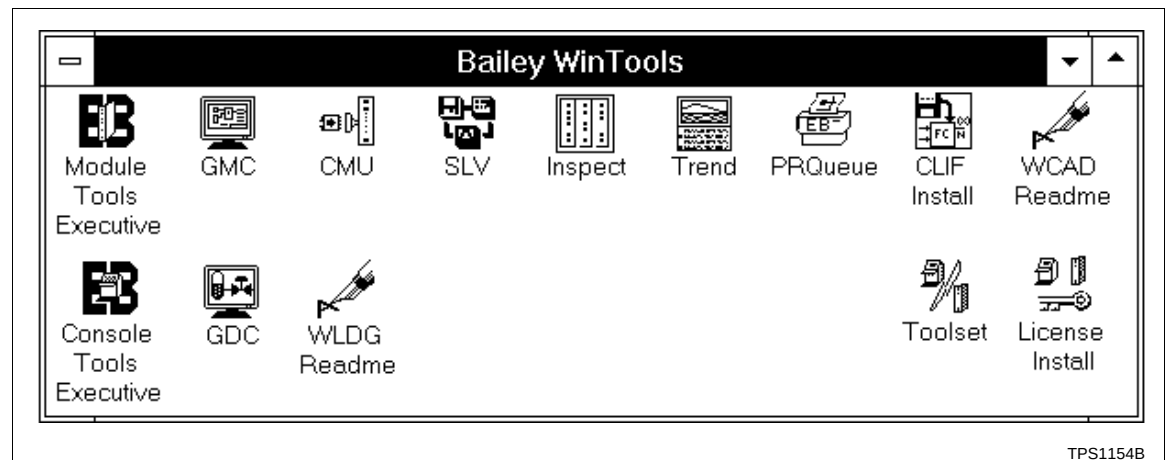


Figure 3-3. *Bailey WinTools* Group

From the Windows Start Menu

To start the module tools executive from the start menu (Windows 95 or Windows NT 4.0):

1. Click *Start* and point to the *Programs* menu.
2. Locate the *Bailey WinTools* menu and choose *Module Tools Executive*.

From the Application Launcher

If the application launcher is running, the module tools executive can be started by clicking on the module tools button or choosing *Module Tools* from the *Application* menu.

When the module tools executive application window appears, its size, location, and state of the project tree are restored from the last session. After a project is opened, configuration tools can be chosen from the module tools executive menus. Refer to **Module Tools Executive Menus**.

Before a project is opened, only the *File*, *Utilities*, and *Help* menus are accessible. After a project is opened, the *Project* menu is available as well and the project tree appears. As loops, PCUs, modules, or other items are selected from the project tree, appropriate menus are enabled or disabled.

The module tools appear as items in their respective menu. Items are selected from the menus in the standard Windows fashion. Use of the mouse, menu keys, and shortcut keys are supported.

Context menus are also supported and can be used as an alternative to the menu bar menus. Context menus are accessed by clicking the secondary mouse button. The context menu that appears depends on the object currently selected. Files (**CAD** sheets) have context menu support in the file selection window. Other project tree items have context menu support in the project tree window. Refer to **Module Tools Executive Menus** for information on the menu items.

Dialog boxes are also used and their operation conforms to standard windows conventions. In general, only one method of operating a dialog box is described; however, any available method can be used.

Opening a Project

This procedure assumes that a project has been defined. Only one project can be open at a time.

To open a project:

1. Choose *Open* from the *File* menu. The *Open Project* dialog box appears. This is a standard Windows file open dialog box. The last opened project name appears in the *File Name* box.
2. Select or type the name of a project in the *File Name* box. Project files have a **PRJ** extension. If necessary, change the drive and directory.
3. Click on *OK* when done.

NOTE: If *Auto Open on Start-Up* from the *File* menu is toggled on (checked), the last opened project is automatically opened when the module tools executive is launched.

The menu bar shows the *Project* and *Utilities* menu. The project description appears beside the project folder graphic at the top of the project tree.

Parts of the Module Tools Executive Screen

Figure 3-4 shows the module tools executive application window with a view of an example project. In the figure, some of the loops and modules have been opened. The module branch is currently selected as indicated by the highlighting and the appearance of the *Module* menu.

The application window consists of the following parts:

- Title Bar** The title bar is a standard Windows title bar showing the application title and containing a control menu and the maximize and restore buttons.
- Menu Bar** Before a project is opened, the menu bar has only the *File*, *Utilities*, and *Help* menus enabled. Other menus are enabled after a project, loop, or module is opened and depending on the branch of the project tree selected. For example, a *Module* menu appears when a module is selected. Utilities are launched by opening a menu and choosing the item of interest.

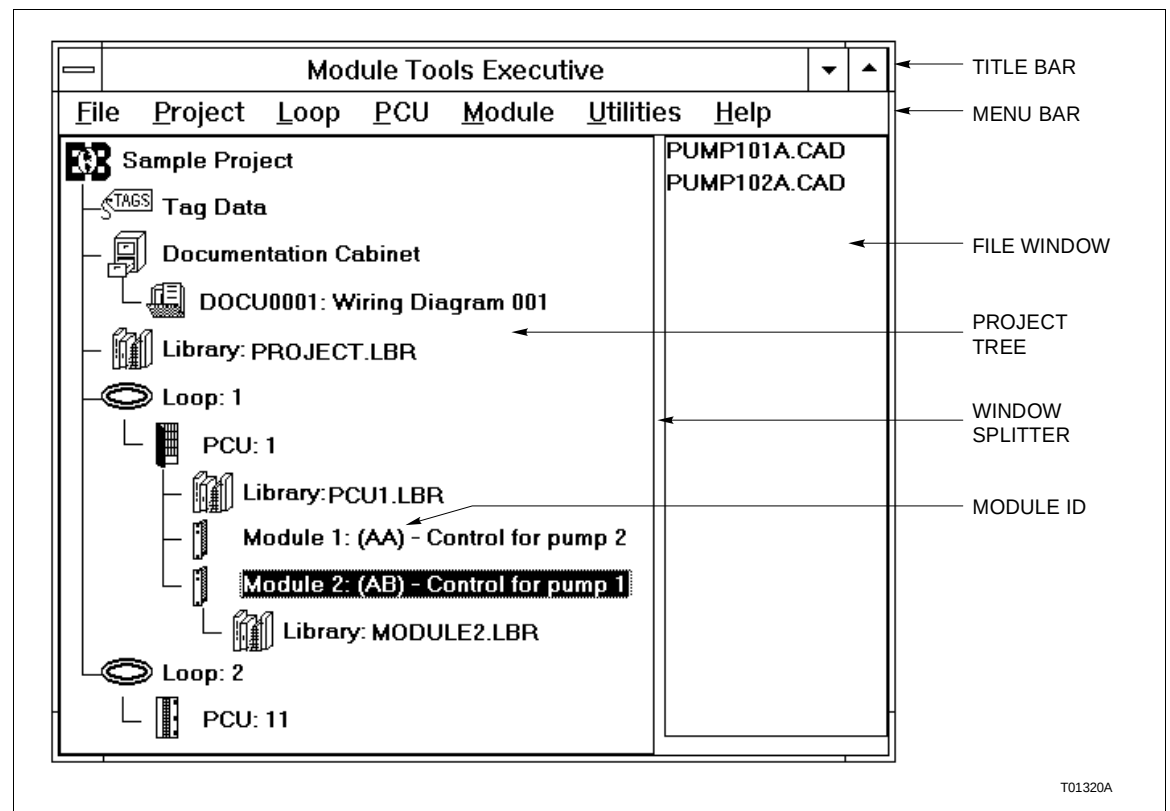


Figure 3-4. Module Tools Executive

- Project Tree** The project tree graphically shows the project structure. If a loop or module is open, then its contents are visible. The menu bar items dynamically reflect the selection from the tree. The project structure is described in **Project Tree Structure**. Operation of the project tree is described in **Using the Project Tree**.
- File List** The file list displays a list of the configuration files from the currently selected module. As an aid, the latest revision and previous revisions of any configuration sheet are visually distinguished in the listing. Before running certain utilities from the module menu, files need to be selected. One or more files can be selected using standard Windows operations.
- Splitter Bar** The window splitter can be used to apportion the amount of the window given to the project tree and file window. To do so, drag the window splitter in the desired direction.
- Scroll Bars** Scroll bars appear as necessary to facilitate viewing and accessing parts of the tree or file list not currently in the window.

Refer to the following sections for information on using the project tree.

Using the Project Tree

The project tree displays a hierarchical view of the project structure as shown in Figure 3-4. The topmost element in the tree is the project indicated by the project description. The project branch contains the project database, documentation cabinet, and loops; the loop branch contains the PCU cabinets; the PCU branch contains the modules; and the module branch contains the configuration sheets. Libraries can be added to any branch.

Each branch has an icon identifying the type of branch. Loop and PCU branches show their address. Module branches show the module address, two-character module identifier, and description. Libraries show the library file name and documentation folders show the folder name which is also the directory name.

The project tree serves three purposes: 1) By defining the project and adding branches, it organizes the project. 2) By opening folders associated with each branch, it gives access to the branch contents. 3) By selecting a branch, it makes utilities appropriate to the branch available.

The loop, documentation cabinets, documentation folders, PCUs, and module icons show the current state of the branch by changing form. If an icon is opened, the branch contents are visible; if the icon is closed, the branch contents are not visible.

The loops, PCU cabinets, modules, libraries, and sheets can be viewed by opening the respective branch.

To open a branch, double click on the respective closed icon.

To close a branch, double click on the respective open icon.

A selected branch appears highlighted. To select a branch of the project tree, click anywhere on the branch of interest: project, documentation cabinet, loop, PCU, module, or library. The branch highlights and the menu bar reflects the branch type selected. If a module or documentation folder is selected, the file list shows the associated **CAD** sheets.

Refer to the **Module Tools Executive Menus** for menu descriptions. The menu item that appears when a branch is selected has the same title as the branch selected. Refer to **Section 4** for information on organizing a project.

Project Tree Structure

The module tools executive and the module tools have been designed to facilitate a project oriented approach in configuring a control strategy. Modules and their configurations are organized and maintained from a common interface (the module tools executive) and structure (the project tree). A typical structure for a module configuration is shown in the project tree of Figure 3-4. The contents of each branch and file is described in the following paragraphs.

Project Contains the loops, tag data, and documentation cabinet. The project also maintains a number of files including the project file (**PRJ**), the border information file (**JHD**), and global save, load, and verify lists (**SLV**). The project file contains the project description and location of the tag data tables. The project name and location is defined by the user when the project is created. The project file is shared by the module and console tools. The border file is used by the automatic borders to automatically fill in project information.

Tag Data Contains the tag data tables for a project. Each project has a tag database called the tag database tables (TDT) that serves as the repository for tag data in the project. These tables are automatically created when a project is defined and are used by the module and console tools. Tag data in the **CAD** sheets and console displays can be used to update this database and tag data in the database can be used to update the **CAD** sheets and console displays. For more information on the tag data tables, refer to the **Console Configuration Tools** instruction.

Documentation Cabinet One documentation cabinet is permitted in a project. The documentation cabinet contains the documentation folders. The

documentation cabinet is available to help organize documentation drawings but is not required.

- Documentation Folder** Documentation folders contain sheets intended for documentation purposes only; these sheets are not intended to be compiled. If these sheets are located in the module directory, then an attempt would be made to compile and bind them along with the actual module configuration sheets. One advantage of using documentation folders is that the sheets contained within a documentation folder are not compiled.
- Library** Libraries are the user libraries containing user symbols, macros, and borders. They can exist at any level of the project tree; however, it is recommended that they be placed at the project level. The library file name is shown next to the library icon.
- Loop** Loops contain PCUs and loop libraries. The loop address is entered by the user when the loop is added to the project. It is shown next to the loop icon as *Loop: nnn*, where nnn is the loop address. When a loop is added to the project, a corresponding directory is created underneath the project directory. The loop directory is designated by Lnnn, where nnn is the loop address.
- PCU** PCUs contain modules and PCU libraries. The PCU address is entered by the user when the PCU is added to the project. It is shown next to the PCU icon as *PCU:nnn*, where nnn is the PCU address. When a PCU is added to the project, a corresponding directory is created underneath its loop directory. The PCU directory is designated by Pnnn, where nnn is the PCU address.
- Module** Modules contain the configuration sheet files (**CAD**) and module libraries. The module address is entered by the user when the module is added to the project. It is shown next to the module icon as *Module: nn*, where nn is the module address. The module identifier in parentheses and description appear after the address. When a module is added to the project, a corresponding directory is created underneath its PCU directory. The module directory is designated Mnn, where nn is the module address. The module also maintains other files in its directory: a module information file (**MHD**) and, if the module has been compiled, the compiled configuration file (**CFG**) and files used by the binder (**REF** and **BND**). Log files (**LOG**) generated by the compiler, cross-referencer, or binder are located in the module directory or its PCU directory. Report and log files are located in the module directory.
- Sheets** The sheets within a module are the individual **CAD** files that comprise the module configuration. The sheets are visible in the file list when the module is selected.

The project structure and corresponding directory structure are automatically created by the module tools executive in the

process of defining the project. The location of configuration files and libraries is tracked by the project and presented to the user via the project tree. In operation, current project information is made available to the utilities. For example, the selected loop, PCU, or module is passed to the compiler. [Section 4](#) describes how to define the project structure and how to manage the project.

NOTE: It is important to not relocate, rename, delete, or otherwise modify any files that are maintained or used by the project. Extreme care should be used when modifying any files or directories using DOS or Windows utilities. An entire project tree can be moved as long as all data is maintained in its location within the tree.

The file window updates to reflect a selection made in the project tree. It displays a list of files in the configuration for the selection. For example, if a module is selected, all configuration sheets in the module appear in the file window. The file window can be used to select files for processing by the tools. Refer to the specific tool description for information on how the file window is used for that tool.

The file window does not update dynamically if another application adds or removes files from the directory. Click on the appropriate branch to refresh the file list.

Module Tools Executive Menus

Each module configuration menu item is briefly described in the following subsections. For a complete description of a particular tool, refer to the appropriate section of the instruction. On-line help is also available for the Windows utilities.

NOTES:

1. Unless otherwise referenced, operation and use of a particular menu item is discussed in Section 4.
2. The same menu item may appear on more than one menu; however, its operation may be somewhat different depending on the menu from which it is chosen. Refer to each menu item description for differences in operation.

As a convenience, context menus are also available by clicking the secondary mouse button. If the tool operates on a file (for example, printing), select the module, select the **CAD** sheets to print, and click the secondary mouse button. A context menu appears from which printing can be chosen. Otherwise, click the secondary mouse button while in the project tree window. A context menu similar to the menu bar menu appears in either case.

FILE MENU

The *File* menu contains project selection and definition items.

New	Defines a new project.
Open	Provides a file open dialog box and defaults to the last project worked on.
Close	Closes the current project.
Auto Open on Start-Up	This is a toggle that can be used to automatically load the last opened project when the console tools executive is launched. It is toggled on when a check mark appears beside this menu item.
Exit	Closes the module tools executive application.

PROJECT MENU

The *Project* menu contains project related items.

Add Loop	Adds a loop to the current project.
Add Documentation Cabinet	Adds a documentation cabinet to the project.
Compile/Xref/Bind	Provides a Windows front end for the compile, cross-reference, and bind utilities. These utilities run from a DOS shell. When launched from the <i>Project</i> menu, all modules in the project are presented for selection for processing. Refer to Section 6 for complete information.
Copy User Library	Copies an existing user library to the selected branch directory and adds the library to the project tree.
Compress User Library	Compresses a user library by removing items marked for deletion. Refer to MANAGING LIBRARIES in Section 4 for information.
Delete User Library	Deletes the selected user library from the project tree and from disk.
Upgrade TDT	Upgrades the TDT format. Refer to UPGRADING THE PROJECT TDT DATABASE in Section 4.
Edit Project Info	Shows the project path and permits editing the project description.
Edit JHD (Border Info)	Permits editing project information that is inserted into automatic borders.

CABINET MENU

The *Cabinet* menu contains documentation cabinet related items.

Add Documentation Folder	Adds a new documentation folder.
Delete Document Cabinet	Deletes the documentation cabinet and all associated folders and files below it from disk.

FOLDER MENU

The *Folder* menu contains documentation related items.

Edit Sheet	Launches the GMC application to edit the selected documentation sheet.
Add Sheet	Adds a sheet to the selected folder and launches the graphical module configuration application with that sheet in a document window.
Import/Copy Sheet(s)	Imports a sheet from another module within the project, from another project, or from an SCAD 5.x job. Refer to Importing Configuration Sheets in Section 4 for information.
Delete Sheets	Deletes the selected document from disk.
Delete Document Folder	Deletes the selected folder and all associated files below it from disk.
Print Style Setup	Sets printing options for the print utility. Refer to PRINTING CAD SHEETS in Section 11.
Print Sheets	Prints the documentation sheets for a folder. Refer to PRINTING CAD SHEETS in Section 11.
Edit Module Info (MHD)	Displays the module information dialog box to permit editing module information.
Configuration Maintenance	Launches the configuration maintenance utility. Refer to Section 7 .

LOOP MENU

The *Loop* menu contains loop related items.

Add PCU	Adds a PCU to the project under the selected loop.
Delete Loop	Deletes the selected loop and all PCUs and files below it from the project. Associated files and directories are also deleted from disk.
Compile/Xref/Bind	Provides a Windows front end for the compile, cross-reference, and bind utilities. These utilities run from a DOS shell. When launched from the <i>Loop</i> menu, all modules in the loop are presented for selection for processing. Refer to Section 6 for complete information.

Tag Synchronize Provides options to update tag data in the global tag table from the module configuration and to update tag data in the module configuration from the global tag database. When chosen from the *Loop* menu, all modules in the loop are presented for selection for processing. Refer to **TAG SYNCHRONIZATION** in Section 11 for complete information.

PCU MENU

The *PCU* Menu contains PCU cabinet related items.

Add Module Adds a module to the project under the selected PCU.

Import Module Adds an existing module to the project. The module and all its sheets are added to the project tree. The existing module can be created by SCAD 5.4 or later or WCAD01.

Delete PCU Deletes the selected PCU and all modules and files below it from the project. Associated files and directories are also deleted from disk.

Compile/Xref/Bind Provides a Windows front end for the compile, cross-reference, and bind utilities. When launched from the *PCU* menu, all modules in the PCU cabinet are presented for selection for processing. These utilities run from a DOS shell. Refer to **Section 6** for information.

Tag Synchronize Provides options to update tag data in the global tag database from the module configuration and to update tag data in the module configuration from the global tag database. When chosen from the *PCU* menu, all modules in the PCU cabinet are presented for selection for processing. Refer to **TAG SYNCHRONIZATION** in Section 11 for complete information.

MODULE MENU

The *Module* menu contains module configuration items.

Edit Sheet Launches the graphical module configuration application with the selected sheet in a document window.

Add Sheet Adds a sheet to the selected module folder and launches the graphical module configuration application with that sheet in a document window.

Import/Copy Sheet(s) Imports a sheet from another module within the project, from another project, or from an SCAD 5.x job. Refer to **Importing Configuration Sheets** in Section 4 for information.

Delete Sheet Deletes selected sheets from the project and from disk.

Delete Module	Deletes the selected module and all configuration sheets below it from the project. Associated files and directories are also deleted from disk.
Compile/XRef	Provides a Windows front end for the compile and cross-reference utilities. When launched from the <i>Module</i> menu, only the selected module configuration is processed. These utilities run from a DOS shell. Refer to Section 6 for information.
Print Style Setup	Sets printing options for the print utility. Refer to PRINTING CAD SHEETS in Section 11.
Print Sheets	Prints selected configuration sheets for a module. Refer to PRINTING CAD SHEETS in Section 11.
Tag Synchronize	Provides options to update tag data in the global tag table from the module configuration and to update tag data in the module configuration from the global tag database. When chosen from the <i>Module</i> menu, only the selected module configuration is processed. Refer to TAG SYNCHRONIZATION in Section 11 for complete information.
Edit Module Info (MHD)	Displays the module information dialog box to permit editing module information.
Configuration Maintenance	Launches the configuration maintenance utility (CMU). Refer to Section 7 for information.

UTILITIES MENU

The *Utilities* menu contains these items.

Communications Setup	Configures communications for WinTools. Refer to CONFIGURING WINTOOLS COMMUNICATIONS in Section 11 for information.
View Log File	Launches the log viewer application. Refer to VIEWING LOG FILES in Section 11 for information.
Global Save/Load/Verify	Launches the SLV file utilities. Refer to Section 9 for information.
DSOE Configuration	Launches the DSOE configuration utilities. Refer to DSOE CONFIGURATION in Section 11.

HELP MENU

The *Help* menu contains help and related items.

Contents	Activates help and provides a table of contents for the console tools executive help.
-----------------	---

Search for Help on	Activates the help search function.
How to Use Help	Displays standard Windows information on using Windows help.
About	Provides information about the WCAD product.

ON-LINE HELP

On-line help is available for the Windows utilities and provides command descriptions, procedures, and reference material similar to that found in the instruction. The on-line help can be used for convenient and quick access to the configuration utility operating procedures. The help system conforms to the standard Windows convention for on-line help. Help can be accessed by the following methods:

- [F1]** Pressing **[F1]** brings up the help *Contents* screen for the currently active utility.
- Help Menu** Selecting *Contents* from the *Help* menu brings up the help *Contents* screen for the currently active utility.

SINGLE POINT TAG DATA ENTRY AND MODULE CONFIGURATION

One feature of the module tools is support for single point tag data entry. For tag information that is used in both the function code definition in the **CAD** sheet and the tag definition in the tag data tables, it would be convenient to have a single source and update the other location automatically.

Exception reporting function codes as defined in the **CAD** configuration sheets have associated tag information: tag name, tag description, tag type, address, and sheet name. They are the logical source for the tag address, sheet name, and tag type. The tag definition in the tag data tables is considered the source for the tag name and description. When a tag is defined, it is given a unique internal tag identifier and the tag data tables is considered the source for the tag identifier as well. Since there are two sources and two storage locations of tag information, there must be some mechanism for propagating information from a source to a storage location and insuring that changes are properly reflected. This mechanism is called tag synchronization.

Tag synchronization relies on a process called tag linking to establish a link between the exception reporting function code and a tag definition in the tag database tables. The tag link is established by attempting to match one of the following pieces of tag information: tag identifier, tag address, and name. Once the link is established, tag synchronization can verify and update tag data as described previously.

Tag synchronization is performed at the following times:

- Each time a CAD sheet is opened or a tag is assigned to an exception reporting function code from the GMC application. Refer to **Module Configuration and Tag Assignment** in Section 5 for more information.
- Whenever a sheet is imported into a module or a module is imported into a PCU cabinet. Refer to **Importing Configuration Sheets** and **Importing an Existing Module** in Section 4 for more information.
- When explicitly requested using *Tag Synchronize* from the module tools executive *Loop*, *PCU*, or *Module* menu. Refer to **TAG SYNCHRONIZATION** in Section 11 for more information.

The ability to link tags and synchronize tag information has several consequences: 1) Tag data defined in one configuration is available for use in other configurations; 2) Changes made to tag data in one configuration are reflected in another configuration (either automatically or by using one of the synchronization utilities); 3) Promotes single point tag data entry that can reduce data input and prevent input of invalid data.

MODULE CONFIGURATION PROCEDURE

The following is a general procedure for configuring modules.

1. **Define or open the project:** Start the module tools executive and use the *File* menu to define or open a project. Refer to **MODULE TOOLS EXECUTIVE**.
2. **Define the border information:** The border file is used to specify project related information and is required for each project. The information from the border file appears on the border of the configuration sheets in that project. When a project is created in the module tools, a dialog box requesting this information automatically appears. **Section 4** provides detailed information on defining project specific information.
3. **Establish the project structure:** Add libraries, loops, PCU cabinets, and modules as required for the project. Refer to **Section 4** for information.
4. **Select or define the module to be configured:** The module header definition (MHD) file is used to define module specific information such as module address, type, revision, etc. Every module must have an MHD file and a dialog box requesting module information automatically appears when a module is added to the project. **Section 4** provides detailed information on creating an **MHD** file.

5. **Graphically configure the module:** Use the graphical module configuration (GMC) application to perform module configuration. Module configurations typically include several sheets. Each sheet is stored as a separate file and consists of function codes and their connections and documentation entities. Refer to [Section 5](#) for details on the GMC application.
6. **Compile the module configurations:** Compilation generates module configurations that can be executed by the modules. After compilation, a configuration may be downloaded to the target module. Compiling also creates the files needed to use the module binding utility. Refer to [Section 6](#) for details about compilation.
7. **Perform cross referencing on the module configurations:** After creating a module configuration, cross referencing can optionally be performed. The cross-reference utility documents references between sheets, process control units and loops. It is recommended that this function be performed as it helps to identify configuration errors. Refer to [Section 6](#) for information.
8. **Perform module binding on the module configurations:** This is an optional step depending on the process control configuration. If modules receive input from modules in other PCUs or loops, then special function codes are used to receive these inputs and their input specification must be the address of the source function block. The input specifications can be entered manually, but the preferred method is to use the binder utility. Module binding automatically enters input specifications for loop and PCU I/O type function codes. After binding, the module configurations must be compiled again. Refer to [Section 6](#) for details about binding.
9. **Download the module configurations:** Use the configuration maintenance utility to download the module configurations to their target module. Refer to **MODULE OPERATIONS** in Section 7. The SLV utility can also be used to download configurations, refer to [Section 9](#) for information. Inspect is used to change module modes, sequence on-line configuration, and return modules to execute mode, refer to [Section 8](#) for information.
10. **Fine tune the control strategy:** Refer to **MONITORING A CONFIGURATION** in Section 5 and [Section 10](#) for more information.

SUPPLEMENTAL CONFIGURATION UTILITIES

This section describes additional operations that can be performed during the course of configuring a control strategy.

Supplemental On-Line Utilities

After a configuration is downloaded, the following can be performed at any time:

Modify the module mode: The module must be put into execute mode to begin executing its configuration. Refer to [Section 8](#) for information on module modes.

Configure module blocks: The module configuration can be changed after it is downloaded. Refer to [HARDWARE OPERATIONS \(PRIMARY AND BACKUP\)](#) in [Section 7](#).

Supplemental Off-Line Utilities

The following can be performed at any time while configuring a module:

Print CAD sheets: The print utility can be used to print the module configurations. The printed sheets can be used as a hard copy reference of the system configuration. Refer to [PRINTING CAD SHEETS](#) in [Section 11](#).

Report on spare and configured blocks: Reports on the spare and configured blocks of **CAD** (sheet) and **CFG** (compiled configuration) files can be generated at any time. Refer to [REPORT OPERATIONS](#) in [Section 7](#).

Report on module memory, block, CPU time, and special function code usage: Module utilization can be calculated at any time. Refer to [REPORT OPERATIONS](#) in [Section 7](#).

Compress a user library: Items marked for deletion in a user library can be deleted from the library. Refer to [MANAGING LIBRARIES](#) in [Section 4](#).

View logs: Log files and reports can be viewed using the log viewer application. Refer to [VIEWING LOG FILES](#) in [Section 11](#).

Synchronize tag data: Tag data from the module configuration can be used to update the tag database and tag data from the database can be used to update tag information in the module configuration. Refer to [TAG SYNCHRONIZATION](#) in [Section 11](#) for complete information.

SLV file utilities: Create, edit, and verify a save/load/verify list. An SLV list can be used to save, load, and verify a list of module configurations in one operation. Refer to [Section 9](#) for complete information.

Distributed sequence of events configuration: Configure the INSEM01 module to provide one millisecond resolution to sequence-of-events related changes-of-state of up to 5,000 input points. Refer to **DSOE CONFIGURATION** in Section 11 for information.

Supplemental DOS Utilities

The following utilities must be run from the DOS prompt:

Password security: Password security is available for the GMC utility. Refer to **PASSWORD SECURITY** in Section 12 for information.

Upgrade 4.x files to 5.x format: This utility must be used to upgrade 4.x files before they are imported into a project. Refer to **UPGRADING 4.x FILES** in Section 12 for information.

WINDOWS CONVENTIONS

The module tools run under the Windows operating environment. The Windows operating environment presents standard methods of interfacing to the utilities. Complete information for using the Windows environment is located in the Windows documentation. Some of the more common user actions are described in Table 3-1. Table 3-2 describes buttons found in dialog boxes.

Table 3-1. Windows Actions

Action	Description
Click	Press and release a mouse button. Unless otherwise specified, it refers to the primary mouse button. For example, clicking on a menu makes it drop down. Clicking on an object selects it.
Ctrl -Click	Press Ctrl while clicking to select multiple objects.
Choose	Cause a command or other action to be performed.
Double click	Press and release the mouse button twice in succession. For example, double clicking on a file in the <i>Open</i> dialog box opens the file.
Drag	Press and hold the mouse button while moving the mouse. For example, dragging is used to position an entity or draw a selection window.
Point	Move the mouse cursor to a designated position.
Multi-select	More than one item can be selected at a time. In the module tools executive file window, multiple files can be selected using the following method: Point to a starting file and drag the cursor down the file list. Each file in turn is selected. Release the mouse button when done.
Select	Mark or highlight an item or entity for some subsequent action or command.

Table 3-2. Windows Dialog Box Buttons

Action	Description
OK	Accepts changes in a dialog boxes and returns to the application.
Cancel	Cancels any changes in a dialog box and returns to the application.
Apply	Accepts changes in a dialog box. The changes are reflected in the application but the dialog box remains. This can be used to verify changes before leaving the dialog box.
Default	Returns dialog box entries to system defaults.
Help	Activates help for the dialog box.

SECTION 4 - MANAGING PROJECTS

INTRODUCTION

This section describes the activities involved in managing a project. These activities include:

- Defining and opening a project.
- Adding modules and other items to a project.
- Deleting items from a project.
- Editing border and module information files.
- Managing libraries.

PROJECT DIRECTORY STRUCTURE

The project directory structure shown in Figure 4-1 reflects the organization of the project in the project tree. It is created as items are added to the project tree. This directory structure is enforced and is the only structure recognized by the module tools executive. In Figure 4-1 directories and files for both the console and module tools are shown; however, the module project tree only shows module related items. The following additional rules regarding the directory structure are observed:

- A project must reside on the user's local disk. Network drives are not supported.
- A project can consist of either module configurations, console configurations, or both.
- A project root directory can have one project in it (designated by a **PRJ** file).
- The project file name (**PRJ**), the border information file name (**JHD**), and the tag data table files all use the same name with different extensions.
- Changes to the project directory structure and files should be made via the module tools executive only.

PROJECT OPERATIONS

Before using a configuration utility, a project must be defined or opened. When a project is defined, it is automatically opened. Opening a project means making it the current project. The current project tree appears in the module tools executive application workspace and current project information is made available to the utilities. Only one module tools executive and project can be active at any one time.

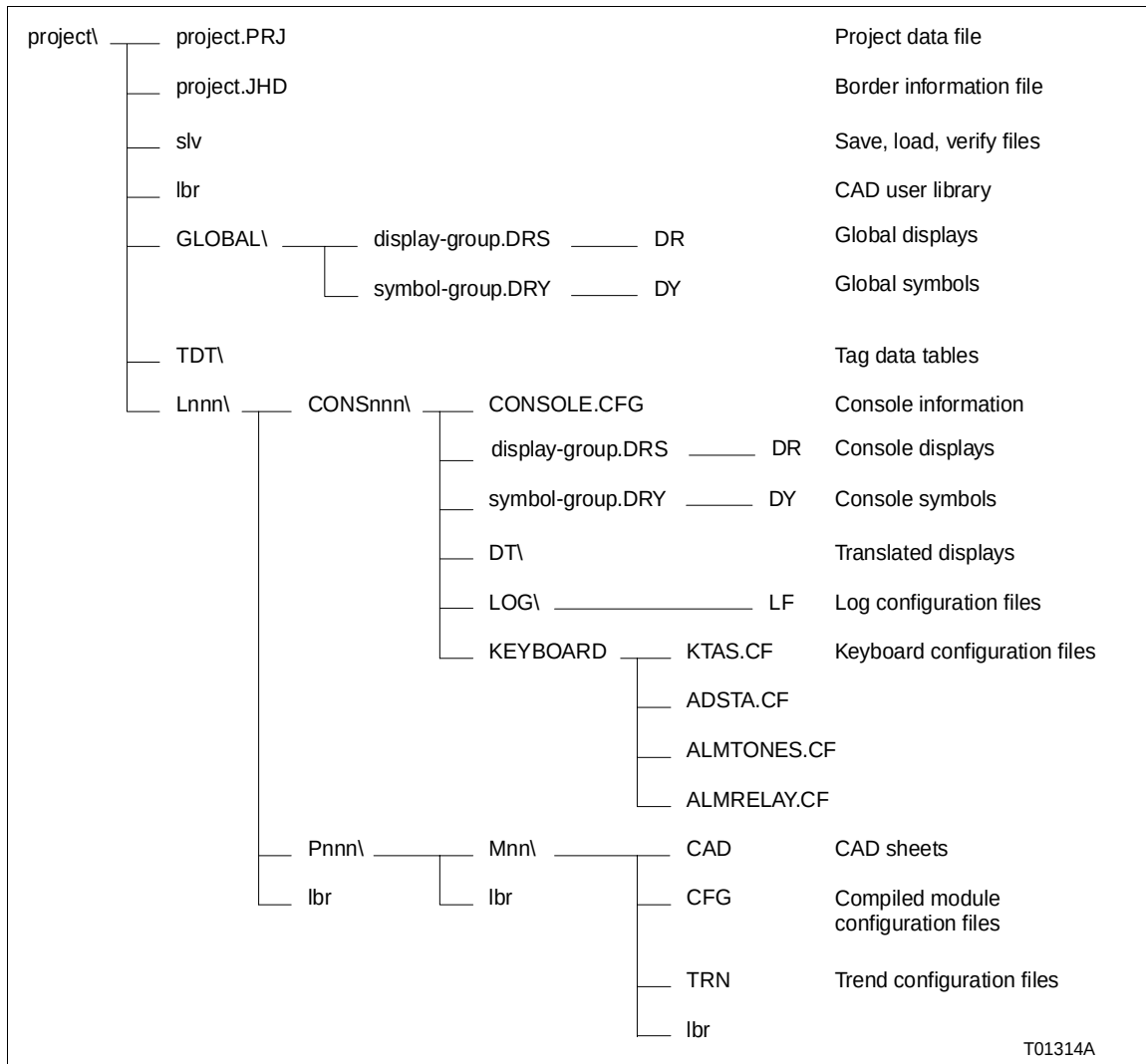


Figure 4-1. Project Directory Structure

Defining a Project

When a project is defined, the project and border information files are created and stored in the project directory. These files use the project name and have different extensions. Each project must be stored in a unique path.

To define a new project:

1. If necessary, use the file manager or DOS commands to create a directory for the new project. The module tools executive will not create a directory for the project.
2. If a project is currently opened, it must be closed before defining a new project. Use *Close* from the *File* menu (**Alt F C**) to close a project.

3. Choose *New* from the *File* menu (**Alt F N**). The *New Project* dialog box appears.
4. If necessary, select a drive and directory to specify a starting path for the project. This must be a local drive. The project files will be stored in this path.
5. Type in a name for the new project in the *File Name* box and click on *OK*.
6. The *Project Information* dialog box appears. Type in a description for the project and click on *OK*.
7. The *Job Info* dialog box appears. Refer to Table 4-1 for field descriptions and fill in the fields appropriately.

Table 4-1. Job Information

Field	Description
Customer name	Twenty-character field for the customer name. An entry in this field is optional.
Plant site	Ten-character field for the plant site. An entry in this field is optional.
Contracting firm	Thirty-character field for the contracting firm. An entry in this field is optional.
Job number	Ten-character field for the project number. This field requires an entry. It is recommended to use the project directory name.
Customer order number 1	Ten-character field for the customer order number. An entry in this field is optional.
Customer order number 2	Ten-character field for the customer order number. An entry in this field is optional.

8. When done, click on *OK* to save the border information in the *project_name.JHD* file in the project directory. The information entered into these fields is automatically inserted into a configuration sheet border provided the border type accepts the field.

After creating a project:

- It is automatically opened.
- The project description appears next to the project folder at the root of the project tree.
- The project files are created and stored in the project directory.
- A subdirectory (TDT) for the tag data tables is created containing an empty tag database (except for the default EUD and LSD lists).

- The tag icon reflecting the tag data tables appears under the project folder.
- Loops, libraries, and a documentation cabinet can be added to the project.

Opening a Project

The *Open Project* dialog box defaults to the last project worked on. The default project can be opened or any defined project can be selected from the *File Name* box. Project files have a **PRJ** extension.

To open a project:

1. If a project is currently opened, it must be closed before opening a new project. Use *Close* from the *File* menu (**Alt F C**) to close a project.
2. Choose *Open* from the *File* menu (**Alt F O**). The *Open Project* dialog box appears.
3. Select or type the name of a project in the *File Name* box. If necessary, select a drive and directory. Only projects on the user's local drive should be accessed.
4. Click on *OK* to continue.

After opening a project:

- The executive reads the directory structure for the project and displays tag data, loop, and documentation cabinet directories as icons in the project tree.
- The project description appears next to the project folder icon at the root of the project tree.

NOTE: Only one project can be opened at a time.

Closing a Project

An opened project must be closed before another project can be opened or defined. To close a project, use *Close* from the *File* menu (**Alt F C**).

EDITING PROJECT INFORMATION

The project information can be edited at any time using *Edit Project Info* from the *Project* menu (**Alt P P**).

EDITING JOB INFORMATION

Every project must have an associated job information file (**JHD**). Job information is project specific information that is automatically inserted into an automatic border. If the project is created using the module tools executive, the job information can be defined at that time. If the project was created using the console tools executive then the job information must be defined by explicitly requesting to do so. The job information can also be edited at any time.

To define or edit border information:

1. Open or define the project for which border information is to be defined using the module tools executive *File* menu.
2. Choose *Edit JHD (Border Info)* from the *Project* menu.
3. The *Job Info* dialog box appears. Refer to Table 4-1 for field descriptions and fill in the fields appropriately.
4. When done, click on *OK* to save the border information in the *project_name.JHD* file in the project directory. The information entered into these fields is automatically inserted into a configuration sheet border provided the border type accepts the field.

DEFINING THE PROJECT STRUCTURE

The project structure is defined by adding branches to the project tree. The *Project*, *Loop*, and *Module* menus have an item to add a branch one level below themselves. For example, the *Project* menu has an *Add Loop* item. After a branch is added to the project, the branch information is automatically recorded in the project directory. There is no need or provision to save the project structure manually.

NOTE: Once a project structure is defined, do not use DOS, Windows, or any utilities to modify project directories or files. To remove branches from the project tree, refer to **DELETING LOOPS, CABINETS, PCUS, AND MODULES**.

Adding Loops

In the module tools executive, loops contain PCU cabinets and loop libraries. Any number of loops can be added, up to the maximum supported.

To add a loop to the project:

1. Choose *Loop* from the *Project* menu (**Alt P L**).

2. The *Add Loop* dialog box appears. Enter a unique loop address from 0 to 250 in the *Loop* text box and click on *OK* to continue.

After adding a loop:

- A loop directory is created under the project directory designated LOOPnnn, where nnn is the loop address.
- The loop branch is added to the project tree.
- PCU cabinets can be added to the project.

Adding Documentation Cabinets

Documentation cabinets contain documentation folders. One documentation cabinet can be added to a project.

To add a documentation cabinet, choose *Add Documentation Cabinet* from the *Project* menu (**Alt P D**).

After adding a documentation cabinet to the project:

- A documentation cabinet directory is created under the project directory designated DOC.
- The documentation cabinet branch is added to the project tree.
- Documentation folders can be added to the project.

Adding Documentation Folders

Documentation folders can be used to organize documentation sheets. These sheets are used for documentation purposes only. They are neither tag linked nor compiled.

To add a documentation folder to the documentation cabinet:

1. Select the documentation cabinet.
2. Choose *Add Documentation Folder* from the *Cabinet* menu (**Alt C A**).
3. Enter the following information in the *Add Document Folder* dialog box that appears:
 - *Directory Name* - A valid DOS name for the folder. This is also the directory in which the documentation sheets will be located.
 - *Module Name* - A valid 5-character module name. Even though the documentation sheets will not be compiled,

compilable symbols may be needed for documentation. A module name must be given to make the appropriate symbols from the system library available.

Click on *OK* to continue.

4. A *Documentation Module* dialog box appears similar to the *Module Information* dialog box. Refer to Table 4-2 for information.

After adding a documentation folder to the documentation cabinet:

- A documentation folder directory is created under the project documentation cabinet directory (DOC) and given the folder name.
- The documentation folder branch is added to the project tree.
- Documentation sheets can be added to the folder.

Adding PCU Cabinets

PCU cabinets contain modules and PCU libraries. To add a PCU cabinet to a loop:

1. Select the loop to which to add a PCU cabinet.
2. Choose *Add PCU* from the *Loop* menu (**Alt L A**).
3. The *Add PCU* dialog box appears. The loop address appears in the *Loop* text box and cannot be changed. Enter a unique PCU address from 1 to 250 in the *PCU Address* text box and click on *OK* to continue.

NOTE: A PCU address within the valid range may conflict with a console address defined in the console tools executive. If so, the module tools executive detects the conflict and informs the user.

After adding a PCU cabinet:

- A PCU cabinet directory is created under the loop directory designated Pnnn, where nnn is the PCU address.
- The PCU branch is added to the project tree.
- Modules can be added to the project.

Adding Modules

Modules contain the configuration sheets and any copied libraries. Each module must have an associated module

information file. Module information defines characteristics of a module used by the other configuration utilities. The information is stored in a module information file (**MHD**).

To add a module to a PCU cabinet:

1. Select the PCU to which to add a module.
2. Choose *Add Module* from the *PCU* menu (**Alt U A**).
3. The *Add Module* dialog box appears. The loop and PCU address are shown but cannot be edited. Enter a unique module address from 0 to 31 in the *Module Address* text box and click on *OK* to continue.
4. The *Module Information* dialog box appears. Refer to Table 4-2 and fill in the requested information. The module type and firmware revision must be provided. Click on *OK* when done.

Table 4-2. Module Information

Field	Description
Module file name prefix	Five-character module name also given to the module configuration and the module information file (MHD). It also forms the first five characters of each CAD sheet in the configuration. Any five characters can be used for the module name. For example, an abbreviation relating the purpose of the module (e.g., PUMP1). A previous method of naming modules included the loop, PCU, and module address using two characters for the loop address, two characters for the PCU address and one character for the module address. Appendix C contains translation tables to use when the numeric address exceeds the allowable number of characters. This field can only be edited when the module is initially added to the PCU folder.
Configuration description	User provided description of the configuration purpose. This will appear after the module address in the project tree.
Module type	Type of module.
Firmware revision	Firmware revision of the module.
Module ID	Two-character alphanumeric module identifier. This identifier is used by the cross-referencer and must be unique for each module in the project.
Drawing title	Drawing title.
Drawing number	Seven-character field for a drawing number.

The information entered into the module information fields (except for module type and firmware revision) is automatically inserted into a configuration sheet border provided the border

type accepts the field. Some of the border information comes from the border information file (**JHD**). Certain fields in the module information dialog box can be edited at any time. Refer to **Editing Module Information**.

After adding a module:

- A module directory is created under the PCU directory designated Mnn, where nn is the module address.
- The module branch is added to the project tree.
- Sheets can be added to the module.

Editing Module Information

Every module must have an associated module information file. Module information defines characteristics of a module used by the other configuration utilities. Some of this information is automatically inserted into an automatic border. The information is stored in a module information file (**MHD**). The module information is defined and the module information file created when the module is initially added to the PCU folder. The module information can also be edited at any time. This procedure describes how to edit a module information file.

To edit module information:

1. Open the project and then open the loop and PCU folders containing the module of interest.
2. Select the module.
3. Choose *Edit Module Info* from the *Module* menu (**Alt M I**). The *Module Info* dialog box appears. Refer to Table 4-2 for field descriptions and fill in the appropriate information.
4. When done, click on *OK* to update the module information file.

The information entered into the module information fields (except for module type and firmware revision) is automatically inserted into a configuration sheet border provided the border type accepts the field. Some of the border information comes from the border information file (**JHD**).

Adding Sheets

To add a sheet to a module or documentation folder:

1. Select the module or folder to which to add a sheet from the project tree.

2. Choose *Add Sheet* from the *Module* menu (**Alt M A**) or *Folder* menu (**Alt O A**). A dialog box appears requesting the sheet number and revision of the sheet to add. Sheet numbering should begin at 0 or 1 and continue through 99. Refer to [Appendix C](#) for alpha-numeric sheet numbering tables when the sheet number exceeds 99. A one-character revision letter is entered after the sheet number.
3. Type in the sheet number and revision and click on *OK*.

After adding a sheet:

- If the GMC configuration is not already running, the GMC application is activated with the new sheet in its document window.
- If the GMC application is running, the GMC application is made the active application and the newly added sheet appears in a new document window.

The sheet is not added to the project until after it is saved from the GMC application.

Importing Configuration Sheets

A sheet can be imported from another module within the project, from another project, or from an SCAD 5.4 job into an existing module. When a sheet is imported, it is renamed using the target module name and the original sheet number and revision. If there is an existing sheet number that would cause a conflict, then a prompt appears to give the imported sheet a new sheet number. Tag synchronization must be performed on sheets when they are imported in order to verify and update tag data and add new tags if needed. It is recommended that import be run with tag synchronization in test mode first, any errors fixed, then import and tag synchronization performed.

NOTES:

1. If the source and target module are different types, then there is a possibility for the imported sheet to contain invalid function codes.
2. If the sheet has an assigned library not in its project tree path, then the library must be imported. Refer to [MANAGING LIBRARIES](#) for information.

To import a sheet:

1. Select the target module from the project tree.
2. Run tag synchronization in test mode:
 - a. Choose *Import Sheet* from the *Module* menu (**Alt M O**).

- b. In the dialog box that appears, locate and select the sheet to be imported.
 - c. Click on *OK* to continue. The tag synchronization dialog box appears.
 - d. Test synchronization should be performed first. In a test synchronization, no files are copied; the test synchronization is performed on the original files as if they were in the target module. Select *Test Sync*.
 - e. There may be tags assigned in the **CAD** sheets that are not defined in the TDT database. The affect of adding these tags to the TDT database can be tested. To do so, check the *Add new tags from CAD file* option. Checking this option does not actually add tags in the *Test Sync* mode.
 - f. Click on *OK* to proceed.
3. Warnings and errors are written to the log file ***cadsync.log*** placed in the module directory. Warnings indicate unsynchronized tag data that can be updated; that is, the tag data in the **CAD** sheet and TDT database do not match and tag synchronization can correct it. Errors indicate a conflict in the data that tag synchronization cannot correct. Warnings can be examined to determine if the updates ought to be made. Errors must be corrected before proceeding with actually performing the synchronization.
4. Copy and perform tag synchronization:
 - a. Choose *Import Sheet* from the *Module* menu (**Alt M O**).
 - b. In the dialog box that appears, locate and select the sheet to be imported.
 - c. Click on *OK* to continue. The tag synchronization dialog box appears.
 - d. Select *Perform Sync* and check *Add new tag from CAD file* if needed.
 - e. Click on *OK* to proceed.

After importing a sheet:

- A renamed copy of the original sheet is made and placed in the target module directory.
- The imported sheet appears in the file window.

Importing Standard Drawings

Standard drawings are available for Bailey Controls termination unit, cabinet, and wiring diagrams. These drawings can be used in any appropriate configuration. The available standard drawings and a short description can be found in the file **README.DWG** in the installation directory. Standard drawings can be imported using the procedure in **Importing Configuration Sheets**. The standard drawings are located in C:\EWS\DRAWINGS.

Some standard drawings have associated libraries that must be imported as well. Refer to **MANAGING LIBRARIES** for more information.

Importing Documentation Sheets

A documentation sheet can be imported from another folder or module within the project, from another project, or from an SCAD 5.4 job into an existing folder. When a sheet is imported, it is renamed using the target module name and the original sheet number and revision. If there is an existing sheet number that would cause a conflict, then a prompt appears to give the imported sheet a new sheet number. Tag synchronization is not performed on documentation sheets.

NOTES:

1. If the source and target module are different types, then there is a possibility for the imported sheet to contain invalid function codes.
2. If the sheet has an assigned library not in its project tree path, then the library must be imported. Refer to **MANAGING LIBRARIES** for information.

To import a documentation sheet:

1. Select the target folder from the project tree.
2. Choose *Import Sheet* from the *Folder* menu (**Alt O O**).
3. In the dialog box that appears, locate and select the sheet to be imported.
4. Click on *OK* to continue and import the sheet to the new folder.

After importing a configuration sheet:

- A renamed copy of the original sheet is made and placed in the target module directory.
- The imported sheet appears in the file window.

Importing an Existing Module

An existing module can be imported into the current PCU folder. The module must have a module information file (**MHD**) and can be from an SCAD 5.4 job or a WCAD01 project. This method can also be used to copy a module within a PCU folder or from one PCU folder to another. Tag synchronization is performed on the module configuration sheets when the module is imported. It is recommended that import be run with tag synchronization in test mode first, any errors fixed, then import and tag synchronization performed.

NOTES:

1. A module cannot be imported to an existing module of the same address.
2. If there are libraries associated with the module that is being imported, the libraries must be imported separately. Refer to **MANAGING LIBRARIES**.

To import a module into a PCU:

1. Select the target PCU to receive the imported module.
2. Run tag synchronization in test mode:
 - a. Choose *Import Module* from the *PCU* menu (**Alt U I**).
 - b. Enter the destination module address in the dialog box that appears and click on *OK*.
 - c. A file search dialog box appears from which a module can be selected. If necessary, change the drive and directory and locate the module information file (**MHD**) of the module to be imported. Click on *OK* when done.
 - d. A module information dialog box appears requesting information on the imported module. Refer to Table 4-2 and fill in the appropriate fields.
 - e. Click on *OK* when done. The tag synchronization dialog box appears.
 - f. Test synchronization should be performed first. In a test synchronization, no files are copied; the test synchronization is performed on the original files as if they were in the target module. Select *Test Sync*.
 - g. There may be tags assigned in the CAD sheets that are not defined in the TDT database. The affect of adding these tags to the TDT database can be tested. To do so, check the *Add new tags from CAD file* option. Checking this option does not actually add tags in the *Test Sync* mode.

- After adding a module:

- A module directory is created under the PCU directory designated Mnn, where nn is the module address.
- A new module information file is created according to the information given and all **CAD** sheets associated with the source module are copied to the destination module directory and renamed using the specified module name.
- The module branch is added to the project tree and the imported sheets appear in the file window.
- New sheets can be added to the imported module.

DELETING LOOPS, CABINETS, PCUS, AND MODULES

Deleting a loop, cabinet, PCU, or module deletes the item and its contents from the project. All associated directories and files are also deleted from disk.

To delete a branch:

1. Select the branch to be deleted.
2. Under the branch menu, locate the delete item for the branch and choose it.
3. A confirmation dialog box appears. Click on *OK* to confirm the deletion. The branch and its contents are immediately deleted from the project.

NOTE: Refer to **MANAGING LIBRARIES** for information on deleting libraries.

DELETING SHEETS

To delete a configuration sheet:

1. Open the module or documentation folder containing the sheet to be deleted. A list of the configuration sheets for that module or folder appears in the file window.
2. Select one or more configuration sheets to be deleted.
3. Choose *Delete Sheet* from the *Module* menu (**Alt M D**) or *Folder* menu (**Alt O D**).

MANAGING LIBRARIES

This section provides procedures to add, delete, and compress user libraries. When adding and deleting libraries to a project, some planning must be given to the consequences of doing so.

Keep the following in mind when using libraries:

- A sheet can only be assigned a single library.
- Libraries can be added at any level of the project tree; however, it is recommended to add libraries at the project level only.
- A sheet can be assigned a library from anywhere in its project tree path. This includes its module, the PCU

containing its module, loop containing its PCU, or from the project.

- After symbols are dropped into a sheet, the library is still required to maintain the symbol in the sheet. The dropped symbol must be able to reference its source in the library.
- If a library is not found at the sheet level (having the assigned library name), the search continues up the project tree structure until a library with the assigned name is found.
- If the symbol is not found in the assigned library, no other libraries are searched.
- A project can have libraries having the same library name at different levels; these libraries need not be the same physical library. (This is not a recommendation for handling libraries in this manner.) Furthermore, different libraries can contain symbols having the same name; these symbols need not be identical symbols.

If a symbol cannot be found by the GMC application, it reports an error. The following problems can result when libraries are improperly used:

- Deleting or moving libraries will obscure references symbols have to the original library they were dropped from.
- Deleting or moving sheets will obscure references symbols have to the original library they were dropped from.
- If the symbol's reference to a library is obscured, the sheet's search for the symbol may find a symbol of the same name in a library of the same name as the original reference. However, the found symbol may not be the correct symbol and no error is reported from the GMC application.

In general, avoid doing things that could potentially cause problems with libraries.

Adding a Library

Libraries can be added at the project, loop, PCU, or module level.

NOTE: A new library can be created from the GMC application. Refer to **USING LIBRARIES** in Section 5.

To add a library:

1. Select the project, loop, PCU, or module to which to add a library.
2. Choose *Copy User Library* from the *Project* menu (**Alt P U**). Locate and select the library to copy from the dialog box that appears. Library names have an **LBR** extension. Click on *OK* to continue.

After adding a library:

- The library file is copied to the target directory corresponding to the project tree selection.
- The library appears in the project tree at the selected level.
- The library can be assigned to configuration sheets.

Compressing a Library

User libraries can be compressed to remove symbols marked for deletion. This reduces the library file size. Symbols are marked for deletion using *Delete Item* from the *GMC Library* menu.

To compress a library:

1. Select the library to be compressed.
2. Choose *Compress User Library* from the *Project* menu (**Alt P P**).
3. A confirmation dialog box appears. Click on *OK* to confirm the compression and continue.

Deleting a Library

To delete a library:

1. Select the library to be deleted.
2. Choose *Delete User Library* from the *Project* menu (**Alt P E**).
3. A confirmation dialog box appears. Click on *OK* to confirm the deletion. The library is deleted from the project and from disk.

UPGRADING THE PROJECT TDT DATABASE

This procedure upgrades a WinTools release 1.0/1.1 TDT database to WinTools release 2.0 format for the current project. The upgrade adds the TAGTYPESRC field to the tag list table.

If the TDT database needs upgraded, a message indicating a TDT version mismatch appears when the project is opened. The project will open and the project tree appears, however any tools that operate on the TDT database are disabled (the menu items are grayed out). After upgrading the TDT database, the tools that operate on the TDT database are enabled and the *Upgrade TDT* menu item is disabled.

To upgrade the TDT database:

1. Choose *Upgrade TDT* from the *Project* menu (**Alt P R**).

The *Upgrade TDT* dialog box appears. It shows the current TDT database, its revision, and the amount of disk space needed to perform the upgrade. If *Backup TDT file(s) before upgrading* is checked, any current files that need to be modified during the upgrade are copied to the directory \WT1X under the current TDT directory. This can be used to restore the TDT database to its pre-upgrade condition by copying the files back to the parent TDT directory.

2. Click on *Upgrade* to continue.
3. A confirmation message appears. Click on *Yes* to continue. The upgrade process proceeds.
4. When done, a message appears indicating the results of the upgrade. Click on *OK* to exit the upgrade utility.

SECTION 5 - GRAPHICAL MODULE CONFIGURATION

INTRODUCTION

Module configurations form the basis of a process control system. A module configuration consists of algorithms called function codes. Function codes can monitor analog and digital inputs, process data, communicate among themselves, and send control signals to external devices. A process control system may involve one or more modules residing in one or more process control cabinets. Function codes are also available to facilitate communication between modules either in the same or different PCU cabinet or loop. Exception reporting function codes output point data onto the INFI-NET network and are able to be monitored by the consoles if the point is defined as a tag in the console database.

The graphical module configuration (GMC) application provides a means of developing and editing module configurations using graphical representations of function code algorithms and their interconnections. The function codes and other graphic elements are placed or drawn on documents called sheets. Typically, a configuration is made up of a number of sheets in which each sheet is a separate file containing a portion of the configuration. Sheets can also contain graphic elements for documentation purposes; however, documentation elements are not actually part of the configuration.

If the project tag data tables are available when a sheet is opened, then tag linking is enabled in which an attempt is made to establish a link between each exception reporting function code and a corresponding tag in the tag data tables. This enables synchronization of specific tag information between the configuration sheets and the tag data tables.

The GMC application can also be used to monitor and tune a process control system using the monitor/tune mode of operation. In this mode, function code specifications appear and live values from the module are displayed in the configuration sheet. Tunable parameters can be tuned and the process values monitored while making adjustments.

This section describes the operation of the GMC application and provides procedures outlining activities in developing and editing module configurations including:

- Features of the graphical module configuration application.
- Module configuration overview.
- Launching the GMC application.

- Parts of the GMC application window, its menu items, and on-line help.
- Document operations: opening, saving, printing, and closing a document.
- Document window operation.
- Working in the document work space.
- Placing and drawing configuration entities.
- Selecting configuration entities.
- Editing configuration entities.
- Using libraries and forming shapes, macros, and borders.
- Monitoring and tuning a process.

GMC APPLICATION FEATURES

This version of the GMC application features a Windows interface with mouse and pull-down menu support. Using this version provides the following benefits:

- Single point tag data entry.
- Support for 640×480 and greater display resolution.
- Module configuration files produced by the Module Configuration Tools SCAD 5.4 and earlier can be used with the GMC application.
- On-line help for GMC menu items and procedures.

MODULE CONFIGURATION OVERVIEW

This section provides general information on module configuration, the effects of tag linking, and the operating modes of the GMC application.

General Procedure

The following steps provide a general procedure for configuring modules:

1. Define or open a project and add loops, PCU cabinets, and modules as needed. If the Console Configuration Tools WLDG 2.0 are used for console configuration for the project, then use the same project for both products. This allows the two products to share tag data via the tag database tables.

2. Configure modules using the information provided in this section. Assign tags to exception reporting function codes. These tags are added to the global tag list provided tag linking is enabled (the tag data tables exist and the modules are configured within the context of the project).
3. If the tag list is edited after module configurations are done or changes are made to the module configurations with tag linking disabled, run tag synchronization as described in **TAG SYNCHRONIZATION** in Section 11.
4. Translate the configurations to module format, refer to **Section 6**.
5. Download the configuration to the module, refer to **Section 7**.
6. Refer to **MONITORING A CONFIGURATION** in this section or **Section 10** to tune the process.

General Notes

The following general usage notes apply to this version of the GMC application:

- All mouse operations are performed using the primary mouse button unless otherwise stated. The primary mouse button is the left button by default but can be changed in the Windows control panel.
- A border must be placed into each configuration sheet.
- Documentation cabinets can be added to the project to hold documentation drawings.
- Tag linking and therefore tag synchronization and tag assignment is enabled only if a TDT database exists and the configuration sheet is opened within the context of the project (the sheet is in a directory managed by the project). A TDT database is created automatically when a project is created (from the module tools executive or the console tools executive).

Module Configuration and Tag Assignment

Tags can be assigned to exception reporting function codes and linked to the tag definition in the tag database tables. This facilitates single point tag data entry and tag data synchronization. Single point tag data entry means that tag data need only be entered in one location. For example, since the exception reporting function codes have an address needed by the tag definition in the tag database tables, this information can be taken directly from the function code instead of

entered manually into the tag data tables. Tag synchronization insures that when changes are made to tag data in one location, it is updated in other locations (**CAD** sheet or tag data tables).

When tags are initially assigned to function codes in a **CAD** sheet, the GMC application can update tag data in the tag data tables and update tag data in the **CAD** sheet from the tag data tables. Subsequently, tag synchronization keeps the tag information updated when changes are made. Tag synchronization occurs automatically when a **CAD** sheet is opened within the GMC application. Tag synchronization can also be initiated from the module tools executive. Refer to **TAG SYNCHRONIZATION** in Section 11 for information on the tag synchronization utility.

Tag linking is enabled when a configuration sheet is opened within the context of a project and the tag data tables are available. The following notes apply to GMC operation with tag linking enabled:

- Tags can be assigned to and deleted from exception reporting function codes using *Assign Tag* and *Delete Tag* under the *Functions* menu.
- When a tag is assigned, an existing tag can be used or a new tag defined. If a new tag is defined, the GMC application adds the tag to the tag database tables along with the tag name, description, and sheet name.
- Once a tag is assigned to an exception reporting function code, the GMC application gets the tag name and description from the tag data tables each time the sheet is opened. The tag name and description cannot be edited from the GMC application other than when defining a new tag.
- If a tag link is assigned to an exception reporting function code, then each time the configuration sheet is opened the tag data tables are automatically updated with the address and sheet name from the function code tag assignment and the tag type is verified.
- At most one tag can be assigned to an exception reporting function code.
- When a tag is deleted from an exception reporting function code, the tag link is removed and the tag remains in the tag data tables. However, the address is removed from the tag definition in the tag data tables.
- The GMC application issues a warning if the user closes a sheet without saving it after tags are assigned or deleted.

If tag linking is not enabled, the following notes apply to GMC operation:

- The assign tag and delete tag functions are disabled.
- If exception reporting function code block addresses are changed, the GMC application issues a warning of a possible out-of-synchronization condition between the tag data tables and the sheet but the block address change is allowed.

Operating Modes

The GMC application has two modes of operation: configure and monitor/tune. Both modes display the module configuration in graphical format. Monitoring and tuning cannot be performed in the configure mode. Configuration can be performed in the monitor/tune mode; however, changes other than in tunable specifications are not reflected in the module until it is compiled and downloaded.

Configure

This is the mode in which module configurations are normally created. When the GMC application is launched, it enters this mode automatically.

Monitor/Tune

The monitor/tune mode permits monitoring and tuning of function code blocks. When this mode is entered, the function code block specifications are updated with actual values from the module. Tunable parameters can be tuned in this mode. To use this mode, the engineering work station must be connected to the INFI-NET network with a network interface unit and the configuration to be monitored must be previously downloaded to the module. Refer to **MONITORING A CONFIGURATION** for information on monitoring and tuning a configuration.

STARTING THE GMC APPLICATION

The GMC application can be started either from within the module tools executive or from outside the executive using the application launcher or any method available from Windows. This procedure assumes that the module tools executive or application launcher is running and is the active application. Refer to **Section 3** for information on the module tools executive and the application launcher.

Opening an Existing Sheet

To start the GMC application within the context of a project and open an existing configuration sheet, do one of the following:

- Select a sheet from the file window then choose *Configure Sheet* from the module tools executive *Module* menu. The

GMC application appears with the selected sheet in a document window.

- Double click on the sheet of interest in the module tools executive file window. The GMC application window appears with the chosen sheet in a document window.

Adding a New Sheet

To start the GMC application within the context of a project and add a new sheet to the module:

1. Choose *Add Sheet* from the module tools executive *Module* menu. A dialog box requesting the last three characters of the sheet name appears. The first five characters come from the module name and appear automatically.
2. Enter the sheet name and click on *OK*. The GMC application window appears with a new, blank sheet.

Starting GMC Outside the Executive

To start the GMC application outside the module tools executive, do one of the following:

- Click on the *GMC* button in the application launcher or choose it from the application launcher *Application* menu.
- For Windows 3.1 and Windows NT 3.51, double click on the *GMC* icon in the *Bailey WinTools* program group.
- For Windows 95 and Windows NT 4.0, select *GMC* from the *Bailey WinTools* menu.

Only one instance of the GMC application can be running. If the GMC application is running and it is launched a second time, focus returns to the GMC application. It will contain a new document window if a file had been selected or added.

CLOSING THE GMC APPLICATION

Closing the GMC application closes the application and returns control to the previous application.

To close the GMC application, do one of the following:

- Choose *Exit* from the *File* menu (**Alt F E**).
- Choose *Close* from the control menu (**Alt Space Bar C**) or use the shortcut key combination **Alt-F4**.

If any document windows contain unsaved changes, a prompt appears to save the documents before exiting. When the GMC application is closed, any open document windows are automatically closed.

GMC APPLICATION WINDOW

This section describes the GMC application window, the GMC menus, and how to access on-line help.

Parts of the GMC Application Window

The GMC application runs under the Windows operating environment. The parts of the GMC application window are shown in Figure 5-1 and described in the following paragraphs.

Title Bar	The title bar is a standard Windows title bar showing the application title. The name, revision, loop, PCU, module, and library of the sheet appear in a document title bar. If the document is maximized, this information appears appended to the application title.
Menu Bar	The menu bar contains the graphical module configuration application menus. Each GMC command is listed as an item on the appropriate menu. Menu selections are made by clicking on the menu and selecting an item, using the menu keys [Alt] [menu key] [item key] , or using a shortcut key combination if one is available.
Tool Bar	The tool bar shows <i>Edit</i> and <i>Tools</i> menu commands in graphic form. The tool bar is an alternative method of selecting a tool to place or draw an entity. In addition, certain file and edit functions reside on the tool bar. The tool bar appears only when enabled using <i>Tool Bar</i> from the <i>View</i> menu ([Alt] [V] [T]).
Application Work Space	The application work space contains one or more document windows. Each document window contains one sheet of a configuration. If a document is minimized, it appears as an icon at the bottom of the application work space.
Document Window	The document window provides the work space on which configurations are drawn. Each document window contains one sheet of the configuration. More than one document window can be opened at a given time in the application work space. If a document window is maximized, then only the maximized document window is visible in the application work space. The document windows can be arranged by sizing, moving, cascading, or tiling.
Status Bar	The status bar appears at the bottom of the application window. It displays messages on the current status of the GMC application and brief help instructions when working on a document. The status bar appears only when enabled using <i>Status Bar</i> from the <i>View</i> menu ([Alt] [V] [S]).

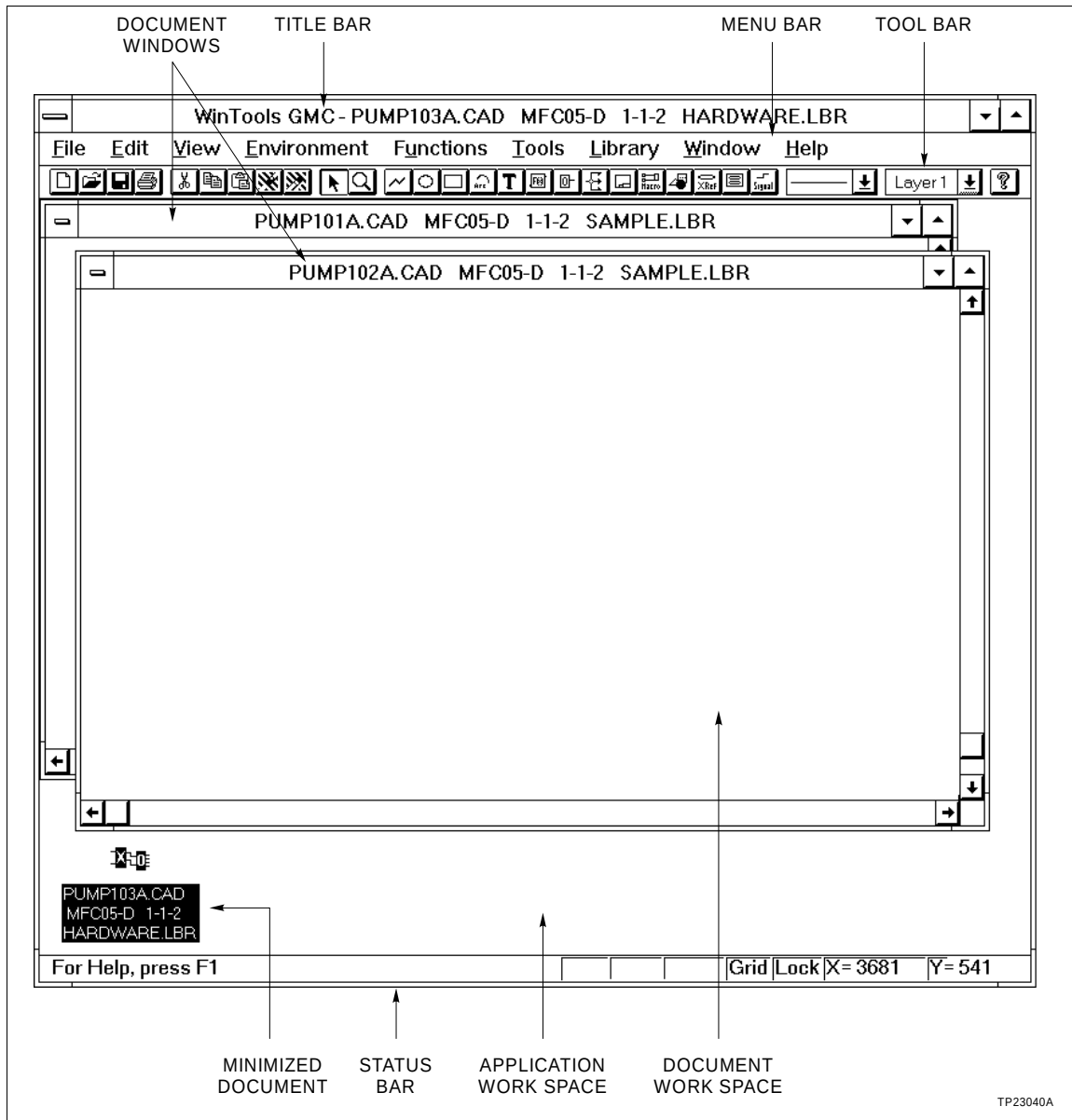


Figure 5-1. GMC Application Window

Coordinates Box

The coordinates box shows the X and Y coordinates of the cursor when it is in the document work space. The coordinates range from 0,0 to 10,000,10,000. The coordinates can be used to position and size entities. The coordinates box is part of the status bar.

The GMC application window also contains scroll bars, maximize and minimize buttons, and a control-menu box. All parts of the GMC application window conform to standard Windows conventions. For more information on these and other standard parts of an application window, refer to the Windows documentation.

Menu Items

The following paragraphs briefly describe each GMC menu. For a complete description of each menu item, refer to the subsections that follow.

A menu item followed by a right pointing triangle indicates a cascading menu. That is, if chosen, another menu appears to the right of the current menu with more items to choose from. A menu item followed by ellipsis (...) indicates a dialog box appears when the item is chosen. Some items are toggles. When the item is toggled on, a check mark appears to the left of the item. Underlined characters in the menu and menu items indicate the keys used in a menu key sequence **Alt** **menu key** **item key**. A key combination to the right of an item indicates it has a shortcut key combination. Pressing the shortcut keys in combination cause the command to be performed immediately without bringing up the menu.

File	Contains items to open, save, revise, close, and print documents. Refer to DOCUMENT OPERATIONS .
Edit	Contains items to edit and modify entities. Refer to EDITING ENTITIES . Contains items to select entities by various methods. Refer to SELECTING ENTITIES .
View	Contains items to control the display of the tool bar, status bar, grid, and properties of certain entities, refer to WORKING IN THE DOCUMENT WORKSPACE . Contains items to refresh the screen and fit the sheet to the window, refer to WORKING IN THE DOCUMENT WORKSPACE .
Environment	Contains items to configure the grid and set the default layer, line style, and text size for a sheet. Refer to WORKING IN THE DOCUMENT WORKSPACE .
Functions	Contains items to edit and modify entities, refer to EDITING ENTITIES . Contains the monitor/tune mode toggle, refer to MONITORING A CONFIGURATION .
Tools	Contains items to place and draw entities, refer to PLACING AND DRAWING ENTITIES . Contains items to magnify and measure distance, refer to WORKING IN THE DOCUMENT WORKSPACE .
Library	Assigns a user library to a sheet, creates shapes, macros, and borders, and deletes items from the user library. Refer to USING LIBRARIES for information.
Window	Contains items to manage the document windows. Refer to DOCUMENT WINDOWS .

Help Contains items to access help, refer to **On-Line Help**.

On-Line Help

On-line help is available for the GMC application. It provides command descriptions, procedures and reference material similar to that found in the instruction. The on-line help can be used for convenient and quick access to the GMC application operating procedures. The help system conforms to the standard Windows conventions for on-line help. Refer to **ON-LINE HELP** in Section 3 or the Windows documentation for more information.

DOCUMENT OPERATIONS

This section describes how to open, save, print, revise, and close documents. In the GMC application, the documents are the module configuration sheets. A sheet is a portion of a configuration drawing. Each sheet is saved as a separate file and appears in its own document window when opened. The first five characters of the sheet name come from the module name. For example, if a module configuration is named BOILR and consists of two sheets, the sheet names may be as follows: **BOILR01A.CAD** and **BOILR02A.CAD**.

Opening a Document

Existing documents can be opened to make changes or continue editing.

To open an existing document:

1. Choose *Open* from the *File* menu (**[Alt][F][O]**) or use the shortcut key combination **[Ctrl][O]**.
2. The *Open* dialog box appears. If necessary, change the drive and directory.
3. Choose the file to be opened by one of the following methods:
 - Select the file name from the file list box and click on *OK*.
 - Double click on the file name in the file list box.
 - Type the file name into the file name box and click on *OK*.
 - The most recently used files appears at the bottom of the *File* menu. To open one of these files, click on the file name.

The chosen file appears in a new document window and becomes the active window.

Opening a Cross-Referenced Sheet

A sheet cross-referenced to another open sheet can be automatically opened.

To open a cross-referenced sheet:

1. Select the reference connector or IO connector having the reference to the sheet to be opened.
2. Choose *Open Xref Sheet* from the *File* menu (**[Alt] [F] [F]**). The cross-referenced sheet appears in a new document window and becomes the active window. If the cross-referenced sheet is already open, then it becomes the active window.

Creating a New Document

To create a new document:

1. Choose *New* from the *File* menu (**[Alt] [F] [N]**) or use the short-cut key combination **[Ctrl]-[N]**.
2. Select the module (**MHD** file) to which the sheet is to be added from the dialog box that appears. If necessary, change the drive and directory to locate the module. Click on *OK* to continue.
3. Enter a sheet number and revision in the *Sheet Revision* dialog box that appears. Click on *OK* to continue.
4. A new document window appears with the title reflecting the module information file selected and with the sheet number provided.

Sheet Naming Conventions

Sheet names must conform to the sheet naming conventions required by the compiler and cross-referencer. This is enforced by the GMC application. The conventions are as follows:

- The first five characters of a sheet are the first five characters of the associated module information file name (**MHD**).
- The last three characters of the sheet file name must be the two-character sheet number and the one-character revision level.
- Sheet numbers should be consecutive, starting at 00 or 01. If sheet numbering goes beyond 99, refer to [Appendix C](#) for two-character equivalents.

Saving a Document

A document can be saved at any time. It is recommended to save documents regularly to prevent loss of data.

To save a document, choose *Save* from the *File* menu (**Alt F S**) or tool bar or use the shortcut key combination **Ctrl-S**.

Printing a Document

Two printing options are available for printing configurations: 1) Printing the current document from within the GMC application using *Print* from the *File* menu; and, 2) Batch printing using *Print* from the module tools executive *Module* menu. The latter method is described in **PRINTING CAD SHEETS** in Section 11.

Printing the Document

To print the current document from the GMC application:

1. Choose *Print* then *File* from the *File* menu (**Alt F P F**) or use the shortcut keys **Ctrl-P**. The standard Windows *Print* dialog box appears.
2. Set options as needed then click on *OK* to continue. The entire current document is printed to the active printer.

Printing the Current View

To print the current view from the GMC application:

1. Choose *Print* then *View* from the *File* menu (**Alt F P V**). The standard Windows *Print* dialog box appears.
2. Set options as needed then click on *OK* to continue. The current view is printed to the active printer.

Previewing the Document

To view the print preview of the entire document, choose *Print Preview* then *File* from the *File* menu (**Alt F V F**). A standard Windows print preview window appears.

Previewing the Current View

To view the print preview of the current view, choose *Print Preview* then *View* from the *File* menu (**Alt F V V**). A standard Windows print preview window appears.

Print Setup

To set up the printer, choose *Print Setup* from the *File* menu (**Alt F R**). A standard Windows *Print Setup* dialog box appears. The printer, orientation, and paper size can be set from this dialog box. To set printer options, click on *Options*. A setup dialog box specific to the printer appears. Refer to the printer documentation for complete information on setting up a specific printer.

Revising a Sheet

A number of revisions of the same sheet can be maintained as an archive or as a backup while testing changes to the configuration.

To create a new revision of a sheet, choose *Rev CAD Sheet* from the *File* menu (**Alt F T**). The original document is closed and a

new document appears with a revision one greater than the previous revision.

Properties

This item displays information about the sheet as described in Table 5-1. To display the properties, choose *Properties* from the *File* menu (**Alt F E**).

Table 5-1. Sheet Properties

Property	Description
Bytes in file	Total number of bytes in the sheet, including the file header.
Bytes in DBF section	Size in bytes of the shape information in the sheet. SCAD 5.x limit is 30,000 bytes.
Bytes in specification section	Size in bytes of the function code specification information in the sheet. SCAD 5.x limit is 3,000 bytes.
Attributes defined	Number of entity attributes defined. SCAD 5.x limit is 750 attributes.

NOTE: Although the GMC application does not limit these properties, it is recommended that the SCAD5.x limits not be exceeded for any sheet that will be compiled.

To display a warning when the file is saved if the SCAD 5.x limits are exceeded, check *Show 5.X Sheet Limits Warning on Write*.

Closing a Document

When editing of a sheet is finished, it can be closed using *Close* from the *File* menu (**Alt F C**). If the sheet contains unsaved changes, a prompt appears to save it first. When a sheet is closed the document window closes and the next document window becomes the current window.

DOCUMENT WINDOWS

When a sheet is opened, it appears in a document window. A GMC document window includes a title bar, control menu, maximize and minimize buttons, and scroll bars. Operation of these components conforms to the Windows conventions and complete information can be found in the Windows documentation. Document windows have the following features:

- The sheet file name and the following information appear in the document window title bar: module type and revision, loop, PCU, and module address, and library name. If the document window is maximized, this information appears in the GMC application title bar.

- Each document window contains one sheet of a configuration.
- The GMC application can have a number of document windows open at one time. The number of open document windows depends on the available memory.
- The document windows can be arranged by tiling or cascading (from the *Window* menu). If the document windows are cascaded, the active window appears in the forefront.
- If a document window is minimized, it appears as an icon at the bottom of the GMC application work space; it is visible if the GMC application work space is not covered by a document window.
- Entities can be copied and pasted between the document windows.
- More than one document window can contain the same sheet.
- The document windows can only be moved within the GMC application window.

Working with Multiple Document Windows

More than one document window (each containing a single sheet) can be opened at a time. If a document window is maximized, it is the only window visible in the work space.

Opening Multiple Document Windows

To open more than one sheet at a time, use *New* or *Open* from the *File* menu. The newly opened sheet appears in a new document window and becomes the active window.

Switching between Document Windows

To switch between document windows, do one of the following:

- If the document window to be made active is visible, click anywhere on the window. It becomes the active window.
- A list of open documents appears under the *Window* menu. Choose the document from this list.
- If the document window is not visible, arrange the other windows (by tiling, cascading, moving, or sizing) to make it visible first, then click anywhere on the window.
- If the document window has been minimized, double click on its icon.

Going to the Next Window

To make the next window the active window, press **Ctrl-F6**. The next document window in the order it was opened becomes the active window.

Displaying a New View A new view of the current document can be opened in a new document window. To do so, choose *New Window* from the *Window* menu **Alt W N**. Each view shows the same sheet, but the views can be zoomed and panned separately. Configuration changes to one view are immediately reflected in the other view.

Sizing, Moving, and Arranging Document Windows

Document windows can be arranged within the GMC application work space. They can be maximized, minimized, tiled, cascaded, and otherwise arranged by the user by sizing and moving. Refer to the Windows documentation for complete information on using document windows.

- | | |
|---------------------------------------|---|
| Maximizing a Document | To maximize a document, click on the document window maximize button. This is the up-arrow button to the far right of the document title bar. When a document is maximized, it takes up the entire document work space and its name appears in the application title bar. It does not have a separate document window in this case. |
| Restoring a Maximized Document | To restore a maximized document, click on the restore button. This is the double-arrowed button located underneath the application maximize button. The document appears in a document window with the same size and position as it had before being maximized. |
| Minimizing a Document | To minimize a document window, click on the document window minimize button. This is the down-arrow button located to the right of the document title bar. |
| Restoring a Minimized Document | To restore a minimized document, double click on the document icon. The document appears in a document window with the same size and position as it had before being minimized. |
| Tiling Documents | To tile document windows, use <i>Tile</i> from the <i>Window</i> menu. |
| Cascading Documents | To cascade document windows, use <i>Cascade</i> from the <i>Window</i> menu. |
| Sizing Documents | To size document windows, point to the document window corner. When the cursor changes to a double ended arrow, drag the border to resize the window. |
| Moving Documents | To move document windows, point to the document window title bar and drag. |
| Arranging Document Icons | To arrange document icons, use <i>Arrange</i> from the <i>Window</i> menu. The document icons are arranged along the bottom of the GMC application work space. |

WORKING IN THE DOCUMENT WORKSPACE

The document work space refers to that area of the document window excluding the menu bar and scroll bars. The work space has a relative coordinate system extending from 0,0 to 10,000,10,000. A border should be placed into the sheet to contain all the configuration entities (refer to **BORDER**). The document work space can be zoomed and panned to gain a better view of any area of interest; for example, to have finer control when drawing, placing, or editing entities. Refer to **Magnifying and Panning** for more information.

The coordinate system is used as an aid in drawing, placing, sizing, and positioning entities by defining a mouse grid and a display grid. The mouse grid is set to some multiple of the coordinate units and is the finest resolution at which entities can be positioned. The display grid is set to some multiple of the mouse grid and can be toggled on and off. Refer to **Using the Grid** for more information.

The remainder of this section describes some of the procedures that can be performed in the document work space. These procedures assume basic familiarity with the Windows operating environment. Use of the mouse and typical methods of operation are stressed. Alternative methods of performing actions may be found in the Windows documentation.

Choosing Commands

All GMC commands and functions are listed as an item on one of the menus. Menu operation conforms to the Windows interface standards. The mouse or keyboard can be used to choose an item from a menu. When a command is chosen, brief help instructions appear in the status line.

Mouse To use the mouse:

1. Click on the menu containing the command item. The menu drops down to reveal a list of the menu items. The first item is selected by default.
2. Do one of the following:
 - Point to an item and click. The command is carried out. If the mouse button is held down, the status line displays a brief description of the item; the command is carried out if the mouse button is released.
 - Drag the highlighted bar through the list and release the mouse when an item is highlighted. Before the mouse button is released, the status line provides a brief description of the item.

A command can be canceled while the mouse button is pressed by moving the mouse pointer off the item before releasing the mouse button.

Each menu and each menu item has an underlined letter that indicates the key sequence to use to choose the item. To use the menu key sequence, sequentially press and release the keys in the sequence **Alt** **menu key** **item key** where **menu key** is the underlined menu character and **item key** is the underlined item character. The menu is opened and the item chosen.

Keyboard

To use the **Alt** keyboard cursor key sequence:

1. Press **Alt**. The *File* menu drops down.
2. Do one of the following:
 - To select an item from the *File* menu, use the **↑** and **↓** arrow keys to select the item and press **Enter** to choose it. While an item is selected and before it is chosen, the status line provides a brief description of the item.
 - To select an item from another menu, use the **←** and **→** arrow keys to select a menu, then the **↑** and **↓** arrow keys to select an item. Press **Enter** to choose the item. While an item is selected and before it is chosen, the status line provides a brief description of the item.

The mouse, menu key sequence, and **Alt** keyboard cursor key methods can be used together in any combination that makes sense to do so.

Some commands also have a shortcut key combination that bypasses the menu system. If a command has a shortcut key combination, it is listed to the right of the menu item. The shortcut key combinations are also given in the command description. When a shortcut key combination is used, the command is performed immediately; the menu is not brought up.

Shortcut Keys

To use a shortcut key combination, press all keys in the combination simultaneously. For example, **Ctrl-S** immediately saves the document.

Repeating the Last Command

Editing operations can be repeated by choosing *Repeat Last Command* from the *Edit* menu (**Alt E L**) or using the shortcut key combination **Ctrl-Enter**.

Using the Tool Bar

The tool bar provides an alternate and more convenient method of selecting a tool for drawing and placing entities. All the commands that draw or place entities on the sheet and

perform commonly used functions are represented by an icon on the tool bar. The tool bar is shown in Figure 5-2.

Choosing a Tool To choose a tool from the tool bar, click on the tool. The cursor changes shape to reflect the chosen tool. Whenever the GMC application is launched, the selection arrow is automatically made the default tool.

Canceling a Tool To cancel the use of a tool once chosen, either press **Esc** or choose the selection arrow (or another tool).

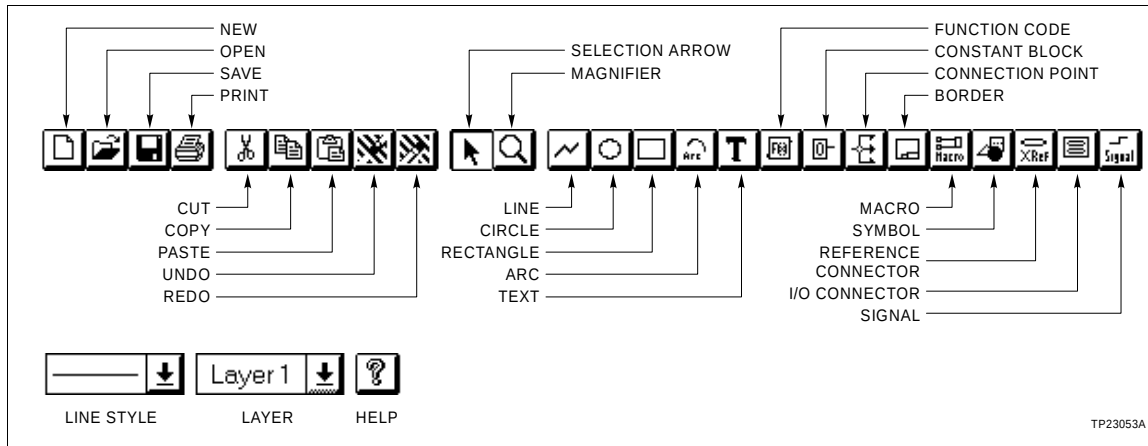


Figure 5-2. GMC Tool Bar

View Options

The display of certain features in the GMC application can be controlled through items on the *View* menu. Most of these items operate as toggles. If a toggle is on, a check mark appears beside the item on the *View* menu. If a toggle is off, the check mark is absent. To change the state of the toggle, choose the item from the *View* menu. The view options affect any open document. The tool bar and status bar always default to their view on state and visible layers default to all layers visible whenever the GMC application is launched. The states of the remaining view options are saved in an initialization file and are restored when the GMC application is launched.

Attributes Toggles the display of entity attributes on and off. Refer to **Adding and Modifying Entity Attributes** for information on entity attributes.

Fit in Window Fits the entire document area to the current size of the document window. The *Fit in Window* function can be used after magnifying an area to return to the original view or after resizing the application window.

Grid (Display)	Toggles the display of the grid on and off. Refer to Using the Grid for information.
Refresh	After performing drawing and editing operations, stray graphics may occasionally litter the document. This is not part of the document and only appears on the screen. The <i>Refresh</i> view function redraws the display and removes the stray graphics.
Spares	Toggles the display of spare specifications associated with each function code on and off. Specifications must be toggled on to view the spares.
Specs	Toggles the display of function code specification numbers and their values on and off.
Spec Descriptors	Toggles the display of descriptors associated with each function code specification on and off. Specifications must be toggled on to view the descriptors.
Status Bar	Toggles the status bar display on and off. Refer to Parts of the GMC Application Window for information on the status bar.
Symbol Origin	Toggles the display of symbol origin points on and off. The symbol origin is the point on a symbol (function code, macro, border, or shape) that is mapped to the insertion point when it is placed. The symbol origin is identified by a × character and is defined when the entity is formed.
Symbol Ref	Toggles the display of the symbol references on and off. A symbol reference is a point on a function code or other configuration entity at which a connection is made with a signal line. The symbol reference is identified by a + character. Refer to SIGNAL for more information.
Text	Toggles the display of text on and off. Refer to TEXT for information on the text entity.
Tool Bar	Toggles the tool bar on and off. Refer to Using the Tool Bar for information.
Layer Properties	Sets the current layer and controls the display and selection of entities according to their layer property. This is the same function as the <i>Set Layer Properties</i> on the <i>Environment</i> menu. Refer to Environment for information.

Environment

The environment refers to properties of the application work space that can be customized or set to default values. Each sheet can have its own environment and it remains in effect for the current editing session or until changed by the user.

Set Layer Properties	Sets the current layer and controls the display and selection of entities according to their layer property.
Current Layer	The current layer is the layer on which new entities are drawn or placed. To set the current layer: 1. Choose <i>Set Layer Properties</i> from the <i>Environment</i> menu (Alt N L). The <i>Layer Properties</i> dialog box appears. 2. Select a layer from the <i>Current Layer</i> pick list. 3. Click on <i>OK</i> to have the changes take effect. NOTE: The current layer can also be selected from the layer box on the tool bar.
Visible Layers	If a layer is set to visible, then entities on that layer are displayed; if a layer is set to not visible, then entities on that layer are not displayed. The current layer is always visible. To set visible layers: 1. Choose <i>Set Layer Properties</i> from the <i>Environment</i> menu (Alt N L). The <i>Layer Properties</i> dialog box appears. 2. Do either of the following: • Check the <i>Visible</i> property of all layers that are to be made visible; uncheck all layers that are to be hidden. • Use the <i>Set All Visible</i> or <i>Clear All Visible</i> to set all layers to visible or not visible. 3. Click on <i>OK</i> to have the changes take effect.
Selectable Layers	If a layer is set to selectable, then selection operations can be performed on that layer; if a layer is set to not selectable, then entities on that layer cannot be selected. To be selectable, a layer must also be visible. To set selectable layers: 1. Choose <i>Set Layer Properties</i> from the <i>Environment</i> menu (Alt N L). The <i>Layer Properties</i> dialog box appears. 2. Do either of the following: • Check the <i>Selectable</i> property of all layers that are to be made selectable; uncheck all layers that are to be not selectable. • Use the <i>Set All Selectable</i> or <i>Clear All Selectable</i> to set all layers to selectable or not selectable. 3. Click on <i>OK</i> to have the changes take effect.

Default Line Style	Sets the line style for line entities. To set the default line style, do one of the following: - Choose <i>Set Default Line Style</i> from the <i>Environment</i> menu (Alt N N) and pick a line style from the list that appears. - Choose a line style from the line style pick list on the tool bar.
Default Character Size	Sets the default character size for the text entity. To set the default text size, choose <i>Set Default Character Size</i> from the <i>Environment</i> menu (Alt N C) and pick a text size from the dialog box that appears.
Grid Control	Defines the mouse and display grid resolution. Refer to Using the Grid for information.
Grid Lock	Restricts drawing, placing, and editing entities to the grid spacing. Refer to Using the Grid for information.
Pan Lock	Toggles auto-panning on and off. Refer to Magnifying and Panning for information.

Magnifying and Panning

This section describes the magnify and pan operations.

MAGNIFYING

An area of the document can be magnified to perform detailed work. The document can also be sized to fit the entire document into the document window. The following functions affect the magnification of the document.

Magnify Fits a windowed region to the document window. To magnify a document using the magnifier tool:

1. The magnifier works by defining a rectangular window that is magnified to fit the document window. Choose the magnifier from the *Tools* menu (**Alt T M**) or use the tool bar. The cursor changes to the magnifier cursor.
2. Position the magnifier at one corner of the window and drag the cursor to the diagonally opposite corner. Release the mouse button. The windowed area is magnified to fit the current size of the document window.

Alternatively, point to and click with the magnifier on any point within the document to magnify and center the area surrounding the point. Once full magnification is reached, the windowed area or point is centered within the document window when magnification is used.

A windowed area can be zoomed out by defining the magnify window such that it extends outside the document window (using auto-panning). The *Fit in Window* function also affects the magnification of the document. Refer to **View Options**.

**Undoing and Redoing
Magnification**

The undo and redo functions work with the magnify and fit to window functions. To restore the view prior to using magnify or fit to window, choose *Undo* from the *Edit* menu (**Alt E U**). *Redo* (**Alt E R**) can be used to redo the magnify or fit to window operation.

PANNING

Panning can occur under two conditions: If the document is zoomed in, panning can be used to move the view left or right and up or down to view other areas of the document at the same zoom. Panning can also be used if the document window has been sized such that even at full size, the window itself is not big enough to display the document. Panning is performed using the scroll bars attached to the side of the document window and by auto-panning.

Scroll Bars

To pan a document using the document window scroll bars, use the horizontal and vertical scroll bars to move the visible portion of the document around the document window. If the document is zoomed to fit in the window and the window is maximized, panning cannot be performed.

Auto-Panning

To pan a document using auto-panning, point the cursor anywhere in the document window and drag the cursor to any edge of the window. As the cursor reaches the edge of the window, the window pans toward that edge of the document. This also works while drawing, moving, or sizing an entity or drawing a selection window. If the document is zoomed to fit the window and the window is maximized, panning cannot be performed.

Pan Lock

Pan lock is a toggle that turns the auto-panning function on and off. Pan lock must be off for auto-panning to work. To toggle it, choose *Pan Lock* from the *Environment* menu (**Alt N P**).

Using the Cursor

The cursor is used to draw and position entities, to select entities for editing, and to move, size and otherwise edit entities. The cursor is moved around the document work space using the mouse in the standard Windows environment manner. Depending on the tool selected or the action being performed, the cursor takes on different shapes. The default cursor is the selection arrow. When a drawing tool or editing command is

chosen, the cursor takes on the shape of the chosen tool or a shape that represents the chosen command.

The coordinates box in the lower right of the application window gives the current cursor position in document coordinate units. If *Snap to Grid* in the *Environment* menu is toggled on, cursor movement is restricted to the grid increment when placing, drawing, or editing entities.

Keyboard Operations

The keyboard can also be used to move the cursor using the keyboard cursor keys. When using the keyboard, **[Space Bar]** performs a primary mouse button click and **[Enter]** performs a double click.

Using the Grid

The grid can be used as an aid in positioning entities when they are drawn, placed, or edited. This can be done in two ways: by visually aligning the entity to the display grid or by using *Grid Lock* and force the entity to lie on grid points. Three functions from the *View* and *Environment* menus affect the grid.

Grid Control

Grid control sets the number of coordinate units between orthogonally adjacent grid points (spacing) and the number of displayable grid points. If *Grid Lock* is toggled on, then drawing, placing, and editing of entities are restricted to grid points. If *Grid* on the *View* menu is toggled on, the displayable grid points are displayed.

Grid control is defined by two numbers. The first number is the number of coordinate units between orthogonally adjacent grid points (spacing) and second number is the number of displayable grid points. Three presets are available: *Grid 100/1*, *Grid 20/1*, and *Grid 10/1*. A fourth setting, *Custom Grid*, is user configurable.

Preset Grid Spacing

To use a preset grid spacing, choose the desired spacing from *Grid Control* under the *Environment* menu: *Grid 100/1* (**[Alt][N][G][0]**), *Grid 20/1* (**[Alt][N][G][2]**), or *Grid 10/1* (**[Alt][N][G][1]**).

NOTE: As a general practice, it is best to place function codes with the grid set to *Grid 100/1* and to make signal connections with the grid set to *Grid 20/1*.

Custom Grid Spacing

To set a custom grid spacing:

1. Choose *Grid Control*, then *Custom Grid* from the *Environment* menu (**[Alt][N][G][C]**). The *Grid Spacing* dialog box appears.
2. Set the *Grid Spacing* box to an integer value between 5 and 400.

3. Set the *Display Nth Grid Point* box to an integer value between 1 and 400.
4. Click on *OK* when done.

NOTE: The grid may not be displayed if the spacing on the display is determined to be too dense.

Grid (Display) This is a toggle that displays the grid points. To toggle it, choose *Grid* from the *View* menu (**[Alt] [V] [G]**). When *Grid* is toggled on, a check mark appears beside the *Grid* item in the *View* menu.

Grid Lock This is a toggle that confines entity drawing, placing, and editing to the grid points set by *Grid Control*. To toggle it, choose *Grid Lock* from the *Environment* menu (**[Alt] [N] [K]**). When *Grid Lock* is toggled on, a check mark appears beside the *Grid Lock* item in the *Environment* menu.

Grid lock works somewhat differently depending on the entity: line, polygon, and rectangle vertices must lie on the grid points; circle and arc center points must lie on grid points; text string start points must lie on grid points; and symbol origins must lie on grid points.

NOTE: *Grid (display)* and *Grid Lock* operate independently of each other.

Measuring Distance

The distance measure function measures the distance between two coordinate points in the document window.

To measure the distance between two points:

1. Choose *Distance Measure* from the *Tools* menu (**[Alt] [T] [D]**). The cursor shape changes to the distance measure tool.
2. Position the cursor at any point in the document window and drag to any other point. The status bar displays the following information:

$(startx, starty)$ to $(endx, endy)=(x,y), | distance |$

where:

<i>startx, starty</i>	Starting X and Y coordinates.
<i>endx, endy</i>	Ending X and Y coordinates.
<i>x, y</i>	Distance along each coordinate axis.
<i>distance</i>	Distance between the two points.

Libraries

Libraries are supported by the GMC application. A system library is available and can be used by any sheet. User libraries can be created and can be assigned to one or more sheets.

- | | |
|---------------|--|
| System | The module configuration utilities includes a complete INFI 90 OPEN system library containing standard borders, function codes, and logic shapes. Items in the system library cannot be modified. Refer to the appropriate command for information on placing items from the system library. Refer to Appendix B for a description of the entities included in the system library. |
| User | The user can create libraries to store user-defined shapes, macros, and borders that are used frequently. A library must be assigned to a sheet before creating and placing user shapes. Once a user library is created, it can be assigned to one or more sheets. A sheet can be assigned at most one user library and that assignment cannot be changed. Refer to USING LIBRARIES for information on assigning a user library to a sheet and creating shapes, macros, and borders for a user library. |

Properties

Each type of entity has properties that can be defined and edited. All entity types have layer and entity attributes. Other properties depend on the entity type. For example, line has line style, function code has block number and specifications, and cross-reference connectors have a descriptor and the cross-reference. These properties all can be edited using the procedures given in **EDITING ENTITIES**.

Layers

Layer is a property of an entity that helps organize the sheet. Each entity is drawn or placed onto one of 16 layers. Layers can be toggled visible or not visible and selectable or not selectable. If a layer is set to visible, entities on that layer are visible; otherwise they are not visible. If a layer is set to selectable, entities on that layer can be selected; otherwise they cannot be selected. The selectable settings allows precisely controlling from which layers entities can be selected. An entity must be visible to be selectable. The current layer is always set to be visible. Each document window can have its layer environment set independently of the others.

Entities with a common purpose can be placed on the same layer. For example, assume that all documentation is placed on one layer and configuration entities placed on another. When making changes to the configuration, the documentation layer could be made not visible or not selectable. This would prevent

inadvertently selecting a documentation entity when trying to select a configuration entity.

The current layer is set with the layer box on the tool bar or using *Set Layer Properties* under the *Environment* menu. Entities are drawn or placed onto the current layer. Refer to **Environment** for information on setting the default, visible, and selectable layers and to **Changing Layers** for information on changing the layer currently applied to an entity.

MODULE CONFIGURATION COMPONENTS

The components that are drawn or placed into a configuration are referred to as entities. Certain entities are used for documentation purposes only; they are not part of the configuration and are not used by the compiler or binder. This includes arcs, circles, lines, rectangles, shapes, and text. Function codes, other configuration entities, and signal lines make up the actual module configuration. Complex entities include shapes, borders, and macros. All entities have certain properties that can be modified after they are placed into the sheet; for example, the layer on which the entity resides. A brief description of each type of entity is given in the following paragraphs.

- | | |
|-------------------------------|---|
| Border | A border is a complex entity that surrounds all other entities forming a configuration sheet. The border has a locator grid, space for the drawing number and other documentation. Optionally, the border automatically fills in the loop, PCU, and module address from the border information (JHD) and module definition files (MHD). Standard borders reside in the system library. Borders can also be created by the user and reside in the user library. Refer to BORDER for information on placing borders and a list of available borders. Refer to Forming Shapes, Macros, and Borders for information on creating a border. |
| Configuration Entities | These are the entities that are used to make up the actual configuration that is downloaded to the module. Refer to FUNCTION CODE , CONNECTION POINT , CONSTANT BLOCK , IO CONNECTOR , REFERENCE CONNECTOR , and SIGNAL . |
| Documentation Entities | This refers to entities used for documentation purposes only; they are not part of the configuration loaded onto the module. Refer to ARC , CIRCLE , LINE , RECTANGLE , SHAPE , and TEXT for more information. |
| Macro | A macro is a user-created collection of configuration and documentation entities referred to by a single name and saved as a single unit in a user library. When a macro is used, it automatically exploded into its component entities. Refer to MACRO for information on placing a macro and Forming Shapes, Macros, and Borders for information on creating a macro. |

Signal	Signals appear as orthogonal lines in a sheet but are designed to connect function codes. Refer to SIGNAL for detailed information.
Standard Drawings	The product release includes standard drawings that can be modified to suit a particular application. Standard drawings of INFI 90 OPEN termination, cabinet, and wiring diagrams are included. To list the available standard drawings and for information on using them in a sheet, refer to USING A STANDARD DRAWING .
System Shapes	A system shape is a collection of entities referred to by a single name and saved as a single unit in the system library. System shapes are documentation entities. The individual components of a shape cannot be edited once the shape is placed into a sheet unless it is first exploded. A system library is provided with the product release and contains predefined system shapes. Refer to Appendix B for a description of available system shapes. Refer to Shape for information on placing a system shape.
Entity Attributes	Entity attributes are assigned to an entity using <i>Modify Attributes</i> from the <i>Edit</i> menu. They appear as text under an entity and can be used in place of the text entity. The advantage of using entity attributes is that they remain connected to the entity and do not impact the memory usage for a configuration. Refer to Adding and Modifying Entity Attributes for more information.
User Shapes	A user shape is a user-created collection of entities referred to by a single name and saved as a single unit in a user library. User shapes are documentation entities. The individual components of a shape cannot be edited while it is still a shape. <i>Explode</i> can be used to separate a user shape into its component entities; however, once a user shape is exploded, it loses its identity as a shape within the sheet. A user shape cannot contain compilable entities. Refer to SHAPE for information on placing user shapes and Forming Shapes, Macros, and Borders for information on creating user shapes.

PLACING AND DRAWING ENTITIES

Entities are placed or drawn using their respective tool. The procedure for placing or drawing an entity depends on the specific entity. Brief help is given in the status line when placing or drawing an entity. On-line help is also available for the entity commands.

NOTE: A GMC module configuration and each sheet in the configuration have limits on the amount of memory they can use and therefore the number of entities that can be added to a sheet. To find out memory and block usage for a sheet or configuration, refer to **REPORT OPERATIONS** in Section 7.

After an entity is placed, properties (layer, entity attributes, etc.) can be changed and it can be otherwise edited and modified (cut, pasted, scaled, etc.); refer to **EDITING ENTITIES**. Before it can be modified in any way, an entity must be selected; refer to **SELECTING ENTITIES**.

The following subsections provide information on drawing or placing each type of entity.

NOTE: The number of documentation entities used affects the number of compilable entities allowed in a sheet since sheets have a limited file size.

ARC

An arc is a primitive entity defined by a center point, a start point, and an end point. To draw an arc:

1. Choose *Arc* from the tool bar or the *Tools* menu (**[Alt] [T] [A]**). The cursor shape changes to the arc cursor.
2. Position the cursor to mark the center point of the arc. Drag the cursor to simultaneously size the arc and to define its start point. A circular shape appears with a radius line as this is being done. The radius line defines the size of the arc and its start point. Release the mouse button when the arc is sized and the start point is defined.
3. Drag the cursor to define the end point of the arc and release the mouse button when positioned.

BORDER

A border is a complex documentation entity that is used to put a border on a drawing. The border contains a locator grid and, depending on the type of border, information from the border information (**JHD**) and module information (**MHD**) files. The border locator grid is used for documentation and locating references between sheets. When a border is placed onto a sheet, the appropriate border and module information is automatically inserted into the border.

A border should be placed into a new sheet. It provides a convenient place to insert information such as the sheet number, loop, PCU, and module address, and customer information. As well, a border is required to perform cross referencing. Some borders are automatic. An automatic border is one that automatically inserts information such as the loop, PCU, and module address. If an automatic border is used, the border information from the border information file (**JHD**) and the module information file (**MHD**) is automatically read into the border when it is placed. Any time the border or module

information is updated, the border is automatically updated when the sheet is opened.

Borders are placed using *Border* from the tool bar or *Tools* menu. Standard borders reside in the system library. Borders can also be created by the user and stored in the user library. Refer to ***Forming Shapes, Macros, and Borders*** to create a border. The following lists and describes automatic borders available in the system library:

AVBORD	Vertical with the Bailey Controls logo.
AVGBORD	Vertical with the Bailey Controls logo, without loop, PCU, and module addresses shown. Use for generic drawings such as cabinet or power drawings.
AVBORDG	Vertical without the Bailey Controls logo.
AVGBORG	Vertical without the Bailey Controls logo, loop, PCU, and module addresses.
AHBORD	Horizontal with the Bailey Controls logo.
AHGBORD	Horizontal with the Bailey Controls logo, without loop, PCU, and module addresses shown. Use for generic drawings such as cabinet or power drawings.
AHBORDG	Horizontal without the Bailey Controls logo.
AHGBORG	Horizontal without the Bailey Controls logo, loop, PCU, and module addresses.

To place a border:

1. Choose *Border* from the *Tools* menu (**Alt T B**) or use the tool bar. The cursor shape changes to the border cursor.
2. Position the insertion point to define the lower left corner of the border and click. A list box appears containing borders from the system library and borders in the user library assigned to the sheet. The library containing the border is listed to the right of the border name: either *System* or the name of the user library assigned to the sheet.
3. The list of borders can be limited to either the system library, the user library, or both by clearing or checking the *System Library* and *User Library* options at the bottom of the dialog box.
4. Select a border by doing one of the following:
 - Select a border from the list box using the mouse or keyboard.

- Type a border name in the text box at the top of the list. As a name is typed the border matching the typed name in alphabetical order is automatically selected in the list box.
5. Click on *OK* to place the border.

CIRCLE

Circles are primitive entities defined by a center point and a circumference point. To draw a circle:

1. Choose *Circle* from the tool bar or the *Tools* menu (**Alt T C**). The cursor shape changes to the circle cursor.
2. Position the cursor to mark the center point of the circle. Drag the cursor to size the circle. Release the mouse button when the circle is sized.

CONNECTION POINT

Connection points can be used to increase the number of connections to a function code or other configuration entity. A total of four signal connections can be made to a connection point. Connection points are placed using *Connection Point* from the *Tools* menu. If a connection point is placed on an existing signal, it splits the signal into two signals and connects them. If a connection point is placed on the intersection of two signals, it splits both signals and connects the four new signals.

To place a connection point:

1. Choose *Connection Point* from the *Tools* menu (**Alt T P**) or use the tool bar. The cursor shape changes to the connection point cursor.
2. Position the insertion point to define where the connection point is to be placed and click. The connection point is placed.

CONSTANT BLOCK

Constant blocks are configuration entities giving a constant output such as 0, 1, or 100.0. The constant blocks are valid only for certain types of modules and only appear in the pick list if valid for the module. They are placed using *Constant Block* from the *Tools* menu. The following is a list of constant blocks available in the system library:

- B0** Boolean zero (fixed block 0).

B1	Boolean one (fixed block 1).
R-1	Real negative one (fixed block 4).
R-100	Real negative one-hundred (fixed block 3).
R0	Real zero (fixed block 5).
R1	Real one (fixed block 6).
R100	Real one-hundred (fixed block 7).
RMAX	Maximum real number allowed: +9E18 (fixed block 9).
RMIN	Minimum real number allowed: -9E18 (fixed block 8).]
SETBLK	Modifiable block number (IMLMM02 only).
ZERO	Integer or real zero (fixed block 2).

To place a constant block:

1. Choose *Constant Block* from the *Tools* menu (**Alt T K**) or use the tool bar. The cursor shape changes to the constant block cursor.
2. Position the insertion point to define the placement of the origin of the constant block and click.
3. Select a constant block from the list that appears by doing one of the following:
 - Select a constant block from the list box using the mouse or keyboard.
 - Type a constant block name in the text box at the top of the list. As a name is typed the constant block matching the typed name in alphabetical order is automatically selected in the list box.
4. Click on *OK* to place the constant block.

FUNCTION CODE

Function codes are algorithms that are the basic components of the control strategy. The function code algorithms manipulate data and are linked together to form a control strategy during the module configuration process. Each function code has a unique graphical representation. This graphical representation is the same as given in the **Function Code Application Manual** which also gives a complete description of each

function code. Function codes are connected by signals (analog or digital) using the signal tool.

NOTES:

1. CLIF function codes can be installed into the project or the EWS directory. If a CLIF function code has replaced an existing function code in the function code database, the original shape is no longer available as follows: within the project if it has been installed in the project directory or within all projects if it has been installed in the EWS directory. Replaced function codes are marked with an asterisk (*) and cannot be placed into a drawing.
2. Existing drawings containing any original function code that has subsequently been overwritten with a CLIF function code cannot be opened.
3. In any sheet based on a module type that supports CLIF function codes, the function code pick list will mark the overwritten function code with an asterisk and will change its description to the CLIF function code.
4. In any sheet based on a module type that does not support CLIF function codes, the function code pick list will mark the overwritten function code with an asterisk and will show the original function code description. Any such function code cannot be placed into the sheet.

To place and use a function code:

1. Choose *Function Code* from the tool bar or *Tools* menu (**Alt T F**). The cursor shape changes to reflect the function code tool.
2. Position the insertion point to mark the origin point of the function code and click. A *Select Function Code* dialog box appears.
3. By default, the function codes are sorted numerically. To sort the function codes alphabetically, clear the *Sort by FC Number* check box.
4. Select the function code of interest from the list box by doing one of the following:
 - Use the mouse or keyboard to locate the function code of interest in the list box and select it.
 - If the function codes are sorted numerically, type the function code number in the text box at the top of the list. As the number is typed the matching function code is automatically selected in the list box.
 - If the function codes are sorted alphabetically, type the function code name in the text box at the top of the list. As the name is typed the matching function code is automatically selected in the list box.

5. Optionally, enter a block number in the *Block Number* text box.
6. Click on *OK*. The function code is placed into the sheet.
7. Do any of the following:
 - Use *Signal* to connect the function code input specification to the function code originating the signal. If necessary, use OREFs and IREFs for signals between sheets as discussed in **REFERENCE CONNECTOR**. Use the function codes listed in **IO CONNECTOR** for signal connections between modules. Connection points can be added to expand the number of allowable connections, refer to **CONNECTION POINT** for information.
 - Fill in the remaining function code specifications, block number, and other attributes using *Modify Specs* from the *Functions* menu. The GMC application does not permit entering function code input specifications using *Modify Specs* unless the function code receives information from another module. Refer to the list of function codes in **IO CONNECTOR**.
 - Fill in the block number using *Change Block Number* from the *Functions* menu.
 - If the function code is an exception reporting function code, create a tag link to the tag data tables using *Assign Tag* from the *Function* menu.

In general, all function codes for a sheet or configuration can be placed, then all the connections made, then all the specifications and block numbers specified rather than performing each step separately for each function code.

NOTES:

1. As function codes are placed and assigned block numbers, it may become unclear which block numbers have been used and which are available. The trace report function can be used to report available block numbers. Refer to **REPORT OPERATIONS** in Section 7 for more information.
2. A report on the configuration utilization can be run at any time. This shows the memory, CPU time, and special function code usage for the module. Refer to **REPORT OPERATIONS** in Section 7 for information.
3. Only function codes valid for the module type appear in the function code pick list.

IO CONNECTOR

Function codes may need to communicate across modules in the implementation of a control strategy. The modules may be

in the same or different PCU cabinet in the same or different loop. Function codes called IO connectors are used to receive input from the source function code and pass it along to the destination function code. The following function codes accept input from another module:

- FC 25** Analog input, same PCU node (AI/B).
- FC 26** Analog input, different PCU node, same loop (AI/L).
- FC 41** Digital input, same PCU node (DI/B).
- FC 42** Digital input, different PCU node, same loop (DI/L).
- FC 46** Digital input list, same PCU node (DIL), logic master module.
- FC 63** Analog input list, same PCU node (AIL/B).
- FC 64** Digital input list, same PCU node (DIL/B).
- FC 121** Analog input, different loop (AI/S).
- FC 122** Digital input, different loop (DI/L).
- FC 178** Data acquisition analog input, loop (DAANG I/L), source is FC177.
- FC 193** User defined data import (DATAIMPT).
- FC 212** Data acquisition digital input, loop (DADIG I/L), source is FC211.

The IO connector needs the source address as an input specification in order to receive input from function codes in another module. The input specification can be filled in manually using *Modify Specs* from the *Functions* menu. However, the recommended method is to do so automatically using the binder utility.

These function codes have cross-reference data as a property. The cross-reference data includes a description and cross-reference. The cross-reference description is used by the binder and cross-referencer utilities.

When using the binder utility, an output reference connector is attached to the source function code and the same cross-reference description is assigned to both the output reference and the IO connector. Using this information, the binder utility can automatically fill in the input specifications of the IO connector from the source function code. The cross-referencer also uses the cross-reference description to fill in the cross-reference. The cross-reference is a documentation function used as an aid in locating signal source and destinations among sheets.

Refer to [Section 6](#) for information on these utilities. The following procedures assume that the binder utility is to be used.

Placing an IO Connector

To place an IO connector:

1. Choose *IO Connector* from the *Tools* menu (**Alt T I**) or from the tool bar. The cursor shape changes to reflect the IO connector tool.
2. Position the insertion point to mark the origin point of the IO connector and click. A *Select Function Code* dialog box appears.
3. By default, the IO connector function codes are sorted numerically. To sort IO connectors alphabetically, clear the *Sort by FC Number* check box.
4. Select the IO connector of interest from the list box by doing one of the following:
 - Use the mouse or keyboard to locate the IO connector of interest in the list box and select it.
 - If the IO connectors are sorted numerically, type the function code number in the text box at the top of the list. As the number is typed the IO connector matching the typed number in numerical order is automatically selected in the list box.
 - If the IO connectors are sorted alphabetically, type the IO connector name in the text box at the top of the list. As the name is typed the IO connector matching the typed name in alphabetical order is automatically selected in the list box.
5. Click on *OK*. The IO connector is placed into the sheet.
6. Use *Signal* to connect the IO connector output to its destination function code.
7. Fill in the any function code specifications using *Modify Specs* from the *Functions* menu.
8. Fill in the block number using *Change Block Number* from the *Functions* menu.

Different Modules, Same PCU Node

To receive a signal from another module in the same PCU node (refer to Example 3 in Figure [5-3](#)):

1. Attach an output reference to the output of the source function code. Use an OREF for a signal that exits the sheet on the right; use an IREFO for a signal that exits the sheet on the left.

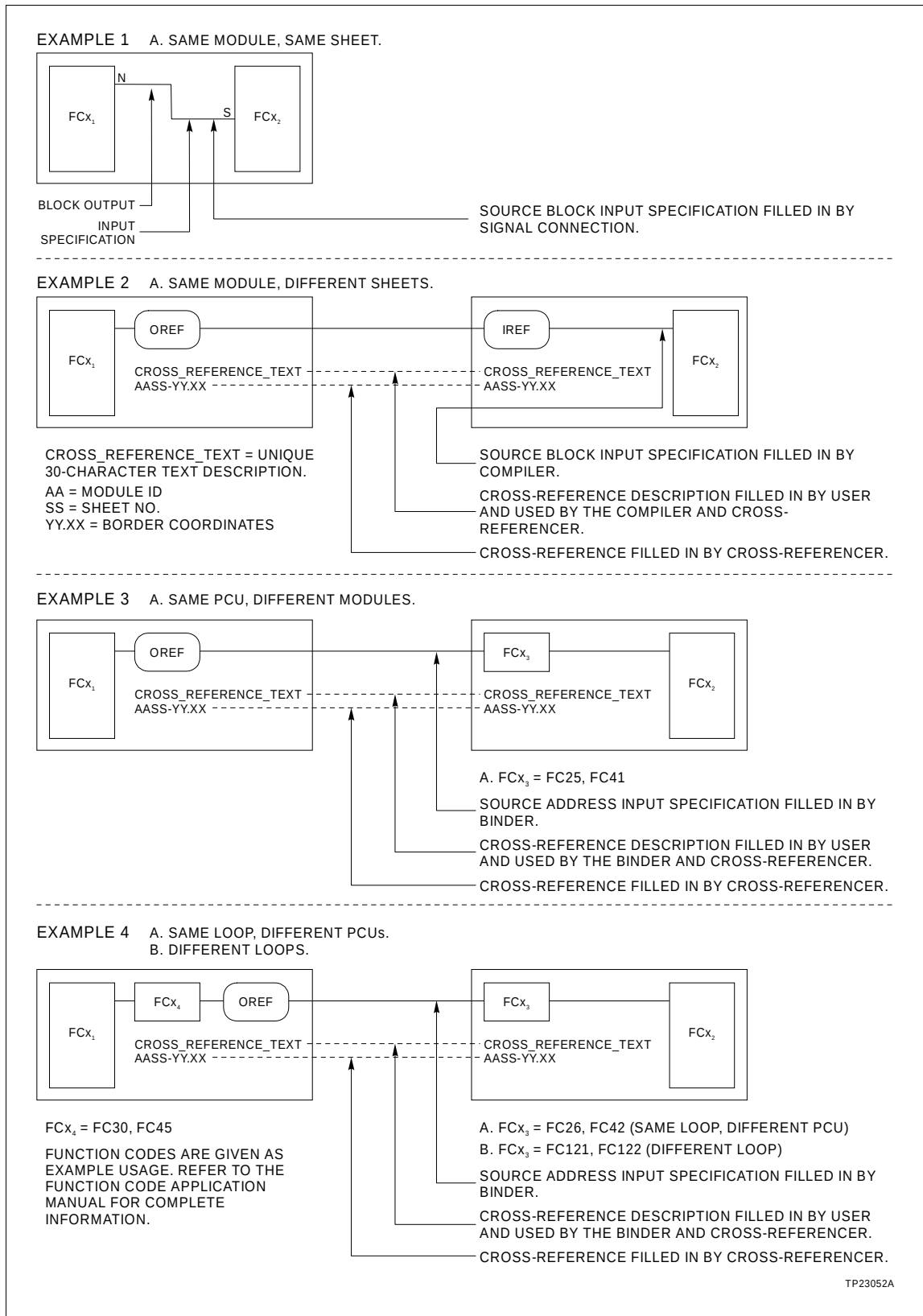


Figure 5-3. Function Code Connections

2. Enter a unique cross-reference description for the output reference using *Change XRef Data* from the *Functions* menu. The descriptor can have up to 30 characters.
3. Go to the destination sheet and place an appropriate function code from the preceding list (same PCU node) using *IO Connector* from the *Tools* menu (**Alt T I**) or from the tool bar. Connect a signal from it to the destination function code.
4. Use *Change XRef Data* from the *Functions* menu and give the input connector the same cross-reference description given to the corresponding output reference.

When the binder utility is run on both modules simultaneously, it enters the source function code module and block address as input specifications in the input connector.

Different PCU Nodes or Different Loops

To receive a signal from another module in a different PCU node or different loop (refer to Example 4 in Figure 5-3):

1. Attach an exception reporting function code to the source function code.
2. Attach an output reference to the output of the exception reporting function code. Use an OREF for a signal that exits the sheet on the right; use an IREFO for a signal that exits the sheet on the left.
3. Enter a unique cross-reference description for the reference using *Change XRef Data* from the *Functions* menu. The descriptor can have up to 30 characters.
4. Place an appropriate function code from the preceding list into the sheet that is to receive the output reference using *IO Connector* from the *Tools* menu (**Alt T I**) or from the tool bar. Connect a signal from it to the destination function code.
5. Use *Change XRef Data* from the *Functions* menu and give the input function code the same descriptor given to the corresponding output reference.

When the binder utility is run on both modules simultaneously, it enters the exception reporting function code address as input specifications in the input connector.

LINE

This includes single line segments, polylines, and orthogonal line segments used for documentation. Each line segment is defined by a start and end point. These types of lines cannot connect function codes (refer to **SIGNAL** for lines that do connect function codes).

Lines or polylines are primitive entities defined by a minimum of two and a maximum of 20 vertex points for a total of 19 segments per line entity.

To draw a single line segment:

1. If necessary, define the line type.
2. Choose *Line* from the tool bar or the *Tools* menu (**Alt T L**). The cursor shape changes to the line tool cursor.
3. Position the cursor to mark the start point and click to anchor the line segment at this point.
4. Position the cursor to mark the end point of the line segment. As the cursor moves, the line rubber bands to fit between the points. When positioned, double click to fix the line segment between these points.

To draw a polyline:

1. If necessary, define the line type.
2. Choose *Line* from the tool bar or the *Tools* menu (**Alt T L**). The cursor shape changes to the line tool cursor.
3. Position the cursor to mark the start point and click to anchor the line segment at this point.
4. Position the cursor to mark the end point of the line segment. As the cursor moves, the line rubber bands to fit between the points. When positioned, click to fix the first line segment between these points.
5. Do any of the following:
 - To continue drawing line segments, continue to position the cursor and click at each end point.
 - To terminate the operation, double click at the end point of the last segment.

NOTE: The number of documentation entities affects the number of compilable entities allowed in a sheet since sheets have a limited file size.

MACRO

Macros are user-created complex entities stored in a user library. Macros can contain documentation and compilable entities.

To place a macro:

1. Choose *Macro* from the *Tools* menu    or use the tool bar. The cursor shape changes to the macro cursor.
2. Position the insertion point to define the placement of the macro origin and click. A list box appears containing macros from the user library assigned to the sheet. The library containing the macro is listed to the right of the macro name.

Although the *System Library* and *User Library* options at the bottom of the dialog box are present, currently there are no macros in the system library and these options should not be used for macros.

3. Select a macro by doing one of the following:
 - Select a macro from the list box using the mouse or keyboard.
 - Type a macro name in the text box at the top of the list. As a name is typed the macro matching the typed name in alphabetical order is automatically selected in the list box.
4. Click on *OK* to continue.
5. If the macro was created with an assigned block number, then a dialog box requesting a starting block number for the macro appears. Enter the block number and click on *OK* to continue. The macro is placed.

RECTANGLE

Rectangles are defined by two points in diagonally opposite corners.

To draw a rectangle:

1. Choose *Rectangle* from the tool bar or the *Tools* menu (  ). The cursor shape changes to the rectangle tool cursor.
2. Position the cursor to mark the first corner point and click.
3. Drag the cursor to the diagonally opposite corner. As the cursor moves, the rectangle rubber bands to fit between the corners.
4. When the second corner is positioned, release the mouse button to anchor the second corner.

REFERENCE CONNECTOR

Reference connectors have cross-reference data as a property. They are used for the following purposes:

- OREF, IREFO, and IREF connectors are used by the compiler to enter the block input specification in a function code receiving a signal from another sheet within the same module.
- OREF and IREFO connectors are used by the binder to enter address input specifications from a source function code into an IO connector receiving a signal from another module.
- OREF, IREFO, and IREF connectors along with associated IO connectors are used by the cross-referencer utility to automatically provide cross-reference information. The cross-reference information shows the module, sheet, and border coordinates of a corresponding source or destination. This is a documentation function used as an aid in locating signal sources and destinations.
- BDIN, BDIND, BDOUT, and BPB connectors are used in manually cross referencing.

The following is a list and description of the available reference connectors:

BDIN	Boolean control input, for logic drawings.
BDIND	Boolean operator indications, for logic drawings.
BDOUT	Boolean control output, for logic drawings.
BDB	Boolean push button input, for logic drawings.
IREF	Input reference (off sheet signal connection).
IREFO	Left facing output reference.
OREF	Output reference (off sheet signal connection).

NOTE: Only IREF, IREFO, and OREF connectors are used by the compiler, cross-referencer, or binder.

Within a sheet, connections between function codes are graphically drawn using *Signal*. This forces the input specification for a function code to match the output block of the corresponding function code that is sourcing the signal. However, most module configurations require the use of more than one sheet to draw the configuration. When using more than one sheet, references to function code inputs and outputs across

sheets must be made explicitly. Three symbols are used to make these references: IREF (input reference), IREFO (an output that exits the left side of the page), and OREF (output reference). Each of these symbols has an associated 30-character text descriptor. Text descriptors given as part of an input reference are needed and used by the compiler to locate the corresponding output reference with the matching text descriptor in order to fill in the function code input specification. Refer to Example 2 in Figure 5-3 for examples of how cross-reference connectors are used.

Placing a Reference Connector

To place a reference connector:

1. Choose *Reference Connector* from the *Tools* menu (**Alt T E**) or from the tool bar. The cursor shape changes to the reference connector cursor.
2. Position the insertion point to define the placement of the origin of the reference connector and click.
3. Select a reference connector from the list that appears by doing one of the following:
 - Select a reference connector from the list box using the mouse or keyboard.
 - Type a reference connector name in the text box at the top of the list. As a name is typed the reference connector matching the typed name in alphabetical order is automatically selected in the list box.
4. Click on *OK* to place the reference connector.

Connecting Function Codes on Different Sheets

To use input and output reference connectors:

1. If a signal goes off the sheet, place an OREF on the right side of the sheet using *Reference Connector* from the *Tools* menu (**Alt T E**) or from the tool bar and connect it to the signal. (Use an IREFO for a signal that goes off the sheet on the left side.)
2. Enter a unique text descriptor for the reference using *Change Xref Data* from the *Functions* menu. The descriptor can have up to 30 characters.
3. Place an IREF on the left side of each sheet that is to receive the output reference connection and connect a signal from its output to the appropriate function code input.
4. Enter the same text descriptor given to the corresponding output reference for the IREF.

The cross-referencer also uses the text descriptors to fill in module, sheet number, and border location in the reference as

a documentation aid. The text descriptors associated with input and output symbols are used in cross referencing the configurations. The drawing borders have locators (x,y coordinates) printed in the margins. These locators are used as a guide to find input/output symbols. Output symbols such as OREFs are cross-referenced to input symbols such as IREFs in the format *mmss-yy.xx*, where:

mm	Unique two-character module identifier defined in the MHD file.
ss	Sheet number.
yy	Y coordinate (with respect to border locator).
xx	X coordinate (with respect to the border locator).

Cross-Referencer Preparation

To prepare a configuration for the cross-referencer:

1. Fill in the module *ID* field when creating a module header or fill it in on an existing header. This field requires two unique user-defined alphanumeric characters for each module. These two characters are automatically inserted when using an automatic border; however, if a non-automatic border is used, these characters must be manually entered on the sheet (above the drawing number). Refer to **BORDER** for more information.
2. Enter sheet numbers above the drawing number. Insure that they are the same as the last two numbers of the drawing file. The sheet number is automatically inserted when using an automatic border.
3. Use IREF, OREF, and IREFO cross-reference connectors appropriately in each sheet as previously described.

The cross-referencer generates a report (**I90XREF.OUT**) listing all text descriptors found and their associated sources and destinations. For more information on the compiler and cross-referencer, refer to **Section 6**.

SHAPE

Shape places a system shape or user shape into the current document. System shapes reside in the system library; user shapes reside in the assigned user library. Refer to **USING LIBRARIES** for information on user libraries and creating user shapes.

To place a shape:

1. Choose *Shape* from the tool bar or *Tools* menu (**[Alt] [T] [H]**). The cursor shape changes to the shape cursor.

2. Position the insertion point to define the placement of the shape origin and click. A list box appears containing shapes from the system library and shapes in the user library assigned to the sheet. The library containing the shape is listed to the right of the shape name: either *System* or the name of the user library assigned to the sheet.
3. The list of shapes can be limited to either the system library, the user library, or both by clearing or checking the *System Library* and *User Library* options at the bottom of the dialog box.
4. Select a shape by doing one of the following:
 - Select a shape from the list box using the mouse or keyboard.
 - Type a shape name in the text box at the top of the list. As a name is typed the shape matching the typed name in alphabetical order is automatically selected in the list box.
5. Click on *OK* to continue and place the shape.

SIGNAL

Signals connect function codes and other compilable entities. Each entity has one or more reference points to which the connections are made. Each reference point can have up to four connections. For entities other than the connection point, one connection of each reference point is automatically used by the entity itself, leaving three available connections.

NOTES:

1. Connected entities must have the same signal type to be compiled.
2. Signals automatically align to grid points. It is best to make connections with *Grid Lock* on and to insure that the entities reference points align to the grid.
3. Signals are only drawn in orthogonal segments.
4. If a connected entity is moved, the connection remains and the signal routes itself to fit the repositioned entity.
5. Signals can only have a solid or dashed line style. If the default line style is not one of these types, it will default to solid when a signal is drawn.

Connect Entities

To connect compilable entities:

1. Choose *Signal* from the tool bar or *Tools* menu (**Alt** **T** **G**). The cursor shape changes to the signal cursor.

2. Position the cursor on an entity reference point and click. The signal will attach itself to the closest available reference point if within close proximity (20 grid points).
3. Route the signal as desired by clicking to form orthogonal line segments.
4. Position the cursor on the second entity's reference point and double click. A signal made up of orthogonal segments is drawn from one connection point to the other.

Partial Connections To connect a signal line to one entity:

1. Choose *Signal* from the tool bar or *Tools* menu (**[Alt] [T] [G]**). The cursor shape changes to the signal cursor.
2. Position the cursor on one of the connection points and click. The signal will attach itself to the closest available connection point if within close proximity (20 grid points).
3. Route the signal as desired by clicking to form orthogonal line segments.
4. Position the cursor at the desired sheet location and double click to terminate the signal. A double line signal reference marks the end of the unconnected signal.

NOTES:

1. When rerouting a signal line, the unconnected end of the signal will remain anchored at its current location.
2. When moving the last segment of an unconnected signal, the end point will snap to the nearest grid point of *Grid Lock* is on.

Connect to Signal End To connect a compilable entity to a signal end, select the entity and move the desired reference point to the end of a grid aligned signal end.

NOTES:

1. The signal end should be aligned to the grid to ease the connection process.
2. Connection points attached to the end of an unconnected signal end must be used to move the signal end beyond the nearest grid location.

Signals can also be moved and split, refer to ***Moving Signal Segments*** for more information.

TEXT

Text is used for documentation purposes only.

To insert text into a sheet:

1. If necessary, define the text size.
2. Choose *Text* from the tool bar or *Tools* menu (**Alt T T**). The cursor shape changes to the text cursor.
3. Position the cursor to mark the starting point of the text string and click. A *Text* dialog box appears.
4. Type a text string, set its size and rotation, and click on *OK* when done to insert the text into the display.

NOTE: The text dialog box supports the Windows clipboard cut, copy, and paste functions. These are activated by clicking the secondary mouse button when in the text box.

DEFINING TAG AND TREND POINTS

If point data is to be used by a console as defined in the console's database as a tag or trend point, then that point must have an appropriate exception reporting function code configured in the module configuration. For a trend point, a trending function code must also be defined. Refer to the **Function Code Application Manual** for a list of exception reporting and trending function codes.

SELECTING ENTITIES

Entities must be selected before being edited. Entities are selected using the mouse or using items from the *Select By* submenu under the *Edit* menu. When an entity is selected, it is surrounded by a bounding box. The various selection methods are described in the following text.

By Pointing and Clicking

To select an entity by pointing and clicking:

1. Choose the selection arrow from the tool bar or from the *Tools* menu (**Alt T S**).
2. Point to the entity to be selected and click.

This method is not guaranteed to select the entity of interest. If one entity is on top of the other, the wrong entity may be selected. If this happens, reposition the cursor and try again or use another method.

By Window

To select one or more entities by window:

1. Choose the selection arrow from the tool bar or from the *Tools* menu (**Alt T S**).

2. A window will be formed around the entities of interest. Position the selection arrow at one corner of the window and drag to form the window. Release the mouse button. All entities entirely within the window are selected.

Clicking the selection arrow can select an entity and prevent a window from being drawn. To circumvent this, hold down **Ctrl-Alt** when drawing the window.

All To select all entities on the current sheet, choose *Select All* from the *Edit* menu.

By Unassigned Block To select all function codes with unassigned block numbers, choose *Select By*, then *By Unassigned Block* from the *Edit* menu (**Alt E B U**).

By Block To select a function code having a block number matching a user provided block number:

1. Choose *Select By*, then *By Block* from the *Edit* menu (**Alt E B B**).

2. In the dialog box that appears, enter the block number of the function code to be selected and click on *OK* to continue.

By Layer To select all entities on user specified layers:

1. Choose *Select By*, then *By Layer* from the *Edit* menu (**Alt E B L**).

2. A dialog box appears from which layers can be checked for selection. Check each layer on which all entities are to be selected and click on *OK* to continue.

By Xref To select entities having a cross-reference matching a user provided text string:

1. Choose *Select By*, then *By Xref* from the *Edit* menu (**Alt E B X**).

2. In the dialog box that appears, enter a text string matching the cross-reference (in the form mmss-yy.xx as described in **REFERENCE CONNECTOR**) of the entities to be selected and click on *OK* to continue.

By Attribute To select an entity having an entity attribute matching a user provided attribute:

1. Choose *Select By*, then *By Attribute* from the *Edit* menu (**Alt E B A**).

2. Select an attribute type from the dialog box that appears and click on *OK* to continue.

3. In the dialog box that appears, enter the value of the attribute to be selected and click on *OK* to continue.

By Tag

To select a function code having a tag name matching a user provided tag name:

1. Choose *Select By*, then *By Tag Link* from the *Edit* menu (**Alt E B T**).
2. In the dialog box that appears, enter the tag name corresponding to the exception reporting function code to be selected and click on *OK* to continue.

By Ordering (Next/Previous)

The next and previous selection methods are used in conjunction with other selection techniques. To select entities using next and previous:

1. Select a group of entities by any of the other methods.
2. Choose *Next* from the *Edit* menu (**Alt E N**) or use the shortcut key **Tab**. The first entity in the order (in which the entities are placed or drawn) is selected. Further use of *Next* cycles through all entities in the group according to their order. Once the last entity is selected, choosing *Next* again selects all the entities in the original group.

Previous (*Previous* from the *Edit* menu (**Alt E P**) or the shortcut keys **Shift-Tab**) works in a similar manner to *Next*, except the previous entity is selected according to entity order.

EDITING ENTITIES

Once an entity is drawn or placed, editing operations can be performed on it. Most entities can be cut, copied, deleted, moved, and entity attributes assigned or modified. Other actions performed on an entity depend on the entity type. Before an entity can be edited, it must be selected as described in **SELECTING ENTITIES**.

Adding and Modifying Entity Attributes

Entity attributes provide an alternate means of documenting entities drawn or placed into a sheet. A entity attribute consists of an attribute type, an attribute value of up to 33 characters, and an occurrence number. Each sheet can have up to 750 entity attributes. Entity attributes appear below the entity to which they are assigned. Entity attributes are defined and edited using *Modify Attributes* from the *Functions* menu.

The attribute types include Tag Name, Tag Descriptor, Output, Unit, Directive, and Program Reference. Most of these attributes are for Bailey Controls internal use; however, the

Tag Name and *Tag Descriptor* attributes (occurrence number zero) are used with the tag linking feature in exception reporting function codes. Any attribute can also be searched for using *By Attribute* from the *Select* item under the *Edit* menu. Other entity attributes can be defined and used for documentation purposes.

NOTE: Occurrence number zero of the *Tag Name* and *Tag Description* attributes in exception reporting function codes are used with the tag assignment feature and should not be used for any other purpose in exception reporting function codes.

The occurrence number allows an entity to have more than one entity attribute of the same type. For example, a function code with two block outputs could have a different output entity attribute for each of its block outputs. As well, two different entities could have identical entity attributes (the same type and value).

There are two advantage in using entity attributes versus text entities:

- Text entered as an entity attribute remains with the entity if it is moved. The position of the entity attributes is fixed, appearing below the entity to which they are assigned.
- Although entity attributes increase the file size of the sheet, they are stored separately from and do not add to the configuration information space which has a 30,000 byte limit.

Adding an Attribute

To define an entity attribute for an entity:

1. Select the entity.
2. Choose *Modify Attributes* from the *Function* menu (**Alt** **U** **A**). The *Modify Entity Attributes* dialog box appears.
3. If no attributes have been defined, a blank record appears in the dialog box in which the first attribute can be defined. To add subsequent attributes, press **↓** until another blank record appears.
4. Choose an attribute type from the drop down list box or use **→** and **←** to cycle through the choices. The first attribute of a given type is given an occurrence number of one. Subsequent attributes of the same type are given occurrence numbers automatically incremented from the previously defined attribute.
5. Type a text string in the value text box.
6. Click on *OK* to accept the entity attribute. The entity attribute is visible provided *Attribute* under the *View* menu is

toggled on and the default display setting for that attribute type is **Y** in the attribute type file (**ATTRIBUT.I90**).

After an entity has one or more entity attributes defined, the entity attributes can be modified.

Modifying an Attribute

To modify the entity attributes of an entity:

1. Select the entity.
2. Choose *Modify Attributes* from the *Functions* menu (**Alt U A**). The *Modify Attributes* dialog box appears listing all the entity attributes defined for that entity.
3. Make changes as follows:
 - To change the attribute type, click on the type field and enter a new type from the drop down list box or use **→** and **←** to cycle through the choices.
 - To change the value, click on the value field and enter a new text string.
 - To remove an attribute, select the attribute and click on *Remove*.
 - To remove all attributes for the selected entity, click on *Remove All*.
4. Click on *OK* to accept the changes and continue.

Assigning Tags

Tags can be assigned to exception reporting function codes to support single point tag data entry. Refer to **Module Configuration and Tag Assignment** for information on how the tag assignments are used.

To assign a tag to an exception reporting function code:

1. Select the function code.
2. Choose *Assign Tag* from the *Functions* menu. A dialog box appears with a list of tags defined in the TDT database.
3. If *View unassigned tags only* is checked (the default), then only tags having their loop, PCU, module, and block address set to zero are listed. Uncheck this option to list all tags in the TDT database matching the default tag type of the function code.

4. Do one of the following:
 - To use a tag already defined, select it from the pick list and click on *OK* to continue.
 - To define a new tag, click on *New* and fill in the appropriate tag information in the dialog box that appears.

Changing Block Numbers

After a function code is placed, it must be assigned a block number. The block number can also be changed at any time after it is assigned.

To assign or change a function code block number:

1. Select the function code of interest.
2. Choose *Change Block Number* from the *Functions* menu (**Alt U B**). The *Block Number* dialog box appears.
3. Enter a new block number and click on *OK* to continue. If the function code is tag linked, the TDT database is updated with the new block address.

Changing Cross-Reference Data

This function enters text descriptions for cross referencing entities (IO connectors and reference connectors) and can be used to change or enter the cross-reference data manually.

To edit cross-reference data:

1. Select a cross-reference entity.
2. Choose *Change Xref Data* from the *Functions* menu (**Alt U X**).
3. In the dialog box that appears, enter or edit the text description. The cross-reference itself can be filled in automatically using the cross-referencer or can be entered manually. Click on *OK* to continue.

Changing Layers

After an entity has been placed, its layer property can be changed.

To change the layer of an entity:

1. Select one or more entities.
2. Choose *Change Layer* from the *Functions* menu (**Alt U Y**). The *Change Layer* dialog box appears.
3. Select a new layer from the list box and click on *OK* to continue.

Changing Line Style

After a line or signal is drawn, its line style property can be changed.

To change the line style of a line or signal:

1. Select the line or signal.
2. Choose *Change Line Style* from the *Functions* menu (**Alt U L**). The *Change Line Style* dialog box appears.
3. Select a new line style from the list box and click on *OK* to continue. Solid and dashed are the only valid line styles for a signal.

Cutting, Copying, and Pasting Entities

These editing commands are often used together. Cutting an entity copies it to the GMC clipboard and deletes it from the sheet. Copying an entity copies it to the GMC clipboard but does not affect the entity. An entity must be selected before it is cut or copied. Pasting an entity places the contents of the GMC clipboard into the current document.

Cut To cut an entity:

1. Select the entity.
2. Choose *Cut* from the *Edit* menu (**Alt E T**) or use the short-cut key combination **Ctrl-X**.

NOTE: If two entities connected by a signal are selected and cut, the signal is cut as well. If only one of the entities is selected and cut, the signal is deleted.

Copy To copy an entity:

1. Select the entity.

2. Choose *Copy* from the *Edit* menu (**Alt E C**) or use the shortcut key combination **Ctrl-C**.

Paste To paste an entity, choose *Paste* from the *Edit* menu (**Alt E P**) or use the shortcut key combination **Ctrl-V**. The pasted entity appears slightly offset from the original.

NOTES:

1. Although the cut, copy, and paste commands work in a similar manner as in other Windows applications, the Windows clipboard is not used in these operations. Entities cannot be copied to the Windows clipboard nor can entities be placed from the Windows clipboard.
2. Tag links are removed from pasted entities on a cut and paste or copy and paste.

Deleting Entities

Entities can be deleted from a document. This is different than cutting an entity since a deleted entity is not pasted to the GMC clipboard.

To delete an entity:

1. Select the entity.
2. Choose *Delete* from the *Edit* menu (**Alt E D**) or use the shortcut key **Del**.

NOTE: If an entity is deleted and has a signal connected to it, the signal is deleted as well, whether or not the signal had been selected.

Deleting Entity Attributes

This function deletes a common entity attribute from a group of entities.

To delete an entity attribute:

1. Select a group of entities having a common entity attribute.
2. Choose *Delete Attributes* from the *Function* menu (**Alt U D**).
3. Select an attribute type and occurrence number from the dialog box that appears and click on *OK* to continue.

Deleting Tag Assignments

After an exception reporting function code has a tag link assigned, it can be deleted. When the assignment is deleted, the tag name and description are removed from the function

code and the loop, PCU, module, and block address of that tag are set to zero in the tag database.

To delete a tag assignment:

1. Select the exception reporting function code.
2. Choose *Delete Tag* from the *Functions* menu (**[Alt] [U] [E]**).
3. A confirmation message appears. Click on *OK* to continue.

Exploding Shapes

A shape can be separated into its component entities in order to edit one or more of the individual entities, for example. This is referred to as exploding it. Once the shape is exploded, it loses its identity as a shape. Not all shapes can be exploded.

To explode a shape:

1. Select the shape to explode.
2. Choose *Explode* from the *Edit* menu (**[Alt] [E] [X]**).

Modifying Specifications

When a function code is placed it is given default specification values. These specification values can be modified to suit the needs of the process control application.

To modify function code specification values:

1. Select one or more function codes.
2. Choose *Modify Specs* from the *Functions* menu (**[Alt] [U] [S]**).

A *Function Code Edit* dialog box appears similar to the one shown in Figure 7-1. Only editable specifications are shown. It provides the function code number, description, and block number in its heading. If the block number has not been defined, a negative block number is shown (as a place holder). In tabular form, the current specification values, data type, whether the specification is tunable, and a description are provided. The *Next* and *Prev* buttons will cycle through the selected function codes. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

Modifying Text

Text strings can be modified as needed. To modify a text string:

1. Select the text to be modified.

2. Choose *Modify Text* from the *Functions* menu (**Alt U T**). The *Modify Text* dialog box appears listing the properties defined for the selected text string.
3. Make any changes to the text string, character size, or rotation.
4. Click on *OK* to accept the changes and continue.

Moving Entities

After an entity has been placed or drawn it can be repositioned as needed.

To move an entity:

1. Select one or more entities.
2. Position the selection arrow within one of the selected entities and drag the entity to its new position.

If a signal or an entity having a signal connected to it is moved, the signal automatically adjusts its length and position to remain connected to the entity.

Moving Signal Segments

The individual segments of the signal can be moved to route them as needed. A signal segment can also be split and either of the new segments moved.

Moving a Signal Segment

To move a signal segment:

1. Select the signal.
2. Position the selection arrow directly on the segment to be moved and drag it to its new position. Connecting segments automatically adjust themselves to remain connected.

Splitting a Signal Segment

To split a signal segment, double click on the signal. Although there is no visible indication, either of the new segments can be moved independently as previously described.

Repeating the Last Command

The last editing operation can be repeated any number of times. To repeat the last editing operation, choose *Repeat Last Command* from the *Edit* menu (**Alt E L**) or use the short cut keys **Ctrl-Enter**.

Rotating Entities

Documentation entities and shapes containing only documentation entities can be rotated after they are drawn or placed. The rotation occurs about the origin point of the entity and is restricted to rotation angles of multiples of 90 degrees. A positive rotation angle produces a counter-clockwise rotation of an entity; a negative rotation angle produces a clockwise rotation of an entity.

To rotate an entity:

1. Select one or more entities.
2. Choose *Rotate* from the *Edit* menu (**[Alt][E][O]**). The cursor changes to the rotate cursor.
3. Position the cursor and click to mark the rotation point.
4. The *Rotate* dialog box appears. Select a rotation angle and direction. Click on *OK* to continue.

The rotate cursor remains as the cursor until another tool is selected.

Scaling Entities

Documentation entities and shapes containing only documentation entities can be scaled after they are drawn or placed. The scaling occurs about the origin point of the entity and is restricted to integer scale factors. A positive scale factor increases the size of an entity; a negative scale factor reduces the size of an entity. For text, scale is a multiplier of the text size.

To scale an entity:

1. Select one or more entities.
2. Choose *Scale* from the *Edit* menu (**[Alt][E][S]**). The scale cursor appears.
3. Position the cursor and click to mark the scale point.
4. The *Scale* dialog box appears. Select a scale factor. Use a negative scale factor to reduce the size of the entity. Click on *OK* to continue.

The scale cursor remains as the cursor until another tool is selected.

Selection Arrow

The selection arrow is used to select and move entities. It is the default tool when a sheet is opened. To use the selection arrow, choose *Selection Arrow* from *Tools* menu (**[Alt] [T] [S]**) or from the tool bar.

Undoing and Redoing Operations

A history of drawing, placing, and editing actions performed is maintained for the current editing session. These actions can be successively undone from last to first. *Undo* can be used to quickly undo an operation that produced unsatisfactory results. If an action is undone, it can be redone using *Redo*. *Redo* reverses the action of the last undo operation.

- Undo** To undo an operation, choose *Undo* from the *Edit* menu (**[Alt] [E] [U]**) or tool bar or use the short cut keys **[Ctrl] [Z]**
- Redo** To redo the last undo operation, choose *Redo* from the *Edit* menu (**[Alt] [E] [R]**) or tool bar or use the short cut keys **[Ctrl] [A]**

FUNCTION CODE EDITOR

The function code editor facilitates viewing, configuring, and editing function blocks. It incorporates several features that make configuring function blocks easier. User-friendly features of the function code editor include specification filtering, application tabs, and specification translation.

- Specification Filtering** Specification filtering refers to the ability to filter and view the specifications by type (all, inputs, internal, tunable, on-line configurable, and spare).
- Application Tabs** This feature filters function code specifications by suitability to function. For example, a function code 216 can be used for TC, RTD, high level, and low level applications. When this function code is configured, a tab appears for each of these applications. By clicking on the appropriate tab, only those specifications needing configuration for that application are displayed.
- Specification Translation** Specification translation refers to the ability to configure complex specifications by picking options from radio buttons and pick lists. The configuration is automatically translated to the numeric value and entered as the specification value.

Function Code Edit Window

The function code editor is used to view or modify function code specifications. An example function code editor window is shown in Figure 7-1.

Title Bar	The title bar shows the application window or view it is assigned to and the loop, PCU, and module address being configured.
Function Code Definition	The function code definition bar shows the block address, function code number, and function code description being configured.
Tab Bar	The tab bar is used by function codes with advanced specification filtering requirements. By clicking a particular tab, only the specifications needed to configure the function code for that usage is displayed. All function codes have a standard and filter tab. The standard tab shows all configurable specifications. The filter tab shows a filtered subset of the specifications based on the application's filter or view options. Additional tabs may be present to provide a logical subset of specifications to ease the configuration process.
Controls	The controls are described in the following subsection.
Specification Display Frame	This area can show all the specifications (as filtered by the tab bar). The view splitter adjusts the amount window space used by the specification display frame and the specification details frame. Scroll bars can be used to view parts of the specification display frame not currently visible. Table 5-2 describes the columns used in this area.

Table 5-2. Function Code Specifications

Item	Description
Spec	Specification number as listed in the Function Code Application Manual .
Tune	Indicates whether the specification is tunable or not.
Type	Data type of the specification: Integer, Real, Character.
Value	Specification value.
Translation	Meaning of the specification value for binary or decimal coded specifications. An out of range message appears for integer and real values if they are out of range. If the translation needs more room, it is extended into the description column and the description appears in the following line.
Description	Description of the specification.

Specification Details Frame	This area is used to configure the function code specifications. If the specification requires an integer, real, or string value, it can be entered here. If the specification requires a binary or decimal coded value, pick lists appear from which, for example, input devices can be selected. For a function code 216, input types of thermocouple, RTD, high level, low level, and user defined appear. After selecting an input type, a subtype can be selected. These selections are automatically translated into the
------------------------------------	--

decimal code and entered as the function code specification. The behavior and appearance of specification details depend on the specific function code being configured.

Controls

The presence of the controls depends on the application and the function being performed. The function code editor controls operate in the following manner:

<i>APPLY</i>	Applies the current specification values to the configuration. If on-line or tuning, or performing hardware configuration, the values are applied to the module configuration residing in the module. Otherwise, these values are saved with the configuration on the next save.
<i>RESET</i>	Resets all specifications to their values prior to the current editing session or since they have been applied.
<i>CLOSE</i>	Closes the function code edit window.
<i>DEFAULT</i>	Each specification has a default value. This button sets the current specification to its default value.
<i>DEFAULT ALL</i>	Sets all specifications to their default values.
<i>PREV FC</i>	Displays the next selected function code if more than one function code is selected for editing specifications.
<i>NEXT FC</i>	Displays the previous selected function code if more than one function code is selected for editing specifications.

Viewing/Filtering Options

These options affect the display of the specifications in the function code editor when the filter tab is selected. The function code edit window must be the active window to set its filter options. Filter options appear on the applications *View* menu.

Overview	Displays all usable specifications defined for the function code.
Inputs	Displays input specifications only.
Internal	Displays internal specifications only.
Tunable	Displays tunable specifications only.

On-Line Configurable	Displays on-line configurable specifications only.
Show Spares	Display spare and reserved specifications for any of the other filter options.

Editing Specifications

Specifications can be edited by typing a value in the *Value* field of the specification display frame or in the specification details frame. Changes do not occur until **Enter** is pressed. If a range value is entered that exceeds the specification's defined minimum or maximum as displayed in the status field, an out of range message appears in the translation field. When editing specifications, a context menu can be used for editing operations (undo, cut, copy, paste, and delete).

If the specification uses a binary or decimal coded value, a pick list appears for the specification in the specification details frame. The value can be entered by selecting from a pick list rather than entering the binary or decimal code. If a value is entered that cannot be represented by the pick list or available buttons, an undefined message appears.

If a value is entered that exceeds the storage capability of the data type, an *Illegal* message is displayed and the cursor remains on that specification field until it is changed to an allowable value.

After making editing changes, use *Apply* to apply the changes to the module configuration if on-line or make the changes available for saving to disk.

USING LIBRARIES

Libraries contain pre-drawn entities that can be placed into a configuration sheet. The system library included with the software release contains function codes, system shapes, and system borders. User libraries can be created and contain user shapes, macros, and user borders created in the GMC application by the user. Each sheet can have one user library attached to it.

Entities from the system library can always be placed provided the module to which the sheet is attached accepts the entity. User shapes, user borders, and macros must be available and defined in the user library assigned to the sheet. Libraries are assigned to a sheet using *Assign Library* from the GMC *Library* menu. Entities are added to a library using *Form Shape*, *Form Macro*, and *Form Border* from the GMC *Library* menu. Refer to **MANAGING LIBRARIES** in Section 4 for more information on user libraries.

Assigning a User Library to a Sheet

Before macros, shapes, or user borders can be used in a sheet or created, the sheet must have a user library assigned to it.

To assign a library to a sheet:

1. Choose *Assign Library* from the *Library* menu (**Alt L A**). A dialog box appears from which an library can be assigned.
2. The user library must be in the directory path of the **CAD** file. Do one of the following:
 - To use an existing library, locate it by specifying the path and then select it from the file list. Click on *OK* to continue.
 - To create and use a new library, type the library name in the file text box and click on *OK* to continue. A dialog box appears to confirm the creation of a new library. Click on *OK* to continue.

Deleting Items From a User Library

Items can be deleted from the library assigned to the currently active configuration sheet.

To delete an item from the current library:

1. Choose *Delete Item* from the *Library* menu (**Alt L D**). A symbol selection dialog box appears showing items in the current library that are not currently in use on any open document.
2. Select the item to delete and click on *OK* to continue. This marks the item for deletion but does not actually remove it from the library. Deleted entities can no longer be placed. Once an item is marked for deletion, it cannot be unmarked.
3. Use *Compress User Library* from the module tools executive *Project* menu to permanently remove all items marked for deletion from the user library.

Forming Shapes, Macros, and Borders

Shapes, macros, and borders can be created and saved to the library assigned to the current configuration sheet.

NOTE: When forming shapes, macros, and borders, certain tools and other commands are not available since they are not intended to be used when forming a shape, macro, or border.

- | | |
|-------------------|---|
| Form Shape | <p>To create a shape:</p> <ol style="list-style-type: none"> 1. If entities from the current configuration sheet are to be included in the shape, they can be selected now. A shape can contain documentation entities only. 2. Choose <i>Form Shape</i> from the <i>Library</i> menu (Alt L S). A new document window opens containing any selected entities. 3. Do any of the following: <ul style="list-style-type: none"> • Draw, place, or paste any documentation entities to be included in the shape. • Edit the entities as needed. • The shape origin defaults to the lower left extents of the included entities and appears as a bright red x. The shape origin can be moved using <i>Set Origin</i> from the <i>Tools</i> menu (Alt T N). • Save the shape using <i>Save</i> from the <i>File</i> menu. On an initial save, a prompt appears requesting a shape name. After saving, editing can continue. 4. When done, use <i>Close</i> from the <i>File</i> menu to complete the shape. A prompt appears requesting to save it to the user library. Click on <i>Yes</i> to save it to the user library. |
| Form Macro | <p>To create a macro:</p> <ol style="list-style-type: none"> 1. If entities from the current configuration sheet are to be included in the macro, they can be selected now. A macro can contain compilable entities. 2. Choose <i>Form Macro</i> from the <i>Library</i> menu (Alt L M). A new document window opens containing any selected entities. 3. Do any of the following: <ul style="list-style-type: none"> • Draw, place, or paste any documentation entities to be included in the macro. • Edit the entities as needed. • The macro origin defaults to the lower left extents of the included entities and appears as a bright red x. The macro origin can be moved using <i>Set Origin</i> from the <i>Tools</i> menu (Alt T N). |

- Save the macro using *Save* from the *File* menu. On an initial save, a prompt appears requesting a macro name. After saving, editing can continue.
- 4. When done, use *Close* from the *File* menu to complete the macro. A prompt appears requesting to save it to the user library. Click on *Yes* to save it to the user library.

Form Border To form a border:

1. If entities from the current configuration sheet are to be included in the border, they can be selected now. A border cannot contain compilable entities.
2. Choose *Form Border* from the *Library* menu (**Alt L B**). A new document window opens containing any selected entities and the standard place holder entities for the project and module information that are used in automatic borders.
3. Do any of the following:
 - Position the place holder entities as needed.
 - Draw, place, or paste any documentation entities to be included in the border.
 - Edit the entities as needed.
 - The border origin defaults to the lower left extents of the included entities and appears as a bright red x. The border origin can be moved using *Set Origin* from the *Tools* menu (**Alt T N**).
 - Save the border using *Save* from the *File* menu. On an initial save, a prompt appears requesting a border name. After saving, editing can continue.
4. When done, use *Close* from the *File* menu to complete the border. A prompt appears requesting to save it to the user library. Click on *Yes* to save it to the user library.

USING A STANDARD DRAWING

Standard drawings are available for Bailey Controls termination unit, cabinet, and wiring diagrams. These drawings can be used in any appropriate configuration by copying them from the module configuration application installation directory. Some standard drawings have associated standard libraries and must be copied as well. The available standard drawings and a short description can be found in the file **README.DWG** in the module configuration application installation directory. The file can be viewed or printed using Notepad. Refer to

Importing Standard Drawings in Section 4 for more information.

MONITORING A CONFIGURATION

The monitor/tune mode permits monitoring and tuning of function code blocks. When this mode is entered, the function code block specifications are updated with actual values from the module. Tunable parameters can be tuned in this mode. To use this mode, the engineering work station must be connected to the INFI NET network with a network interface unit and the configuration to be monitored must be previously downloaded to the module.

Monitor operates as a toggle. When a check mark appears beside the *Monitor* in the *Functions* menu, the GMC application is in the monitor/tune mode. Repeatedly choosing the *Monitor* item toggles the monitor/tune mode on and off.

To monitor and tune a configuration:

1. Open the sheet containing the portion of the configuration to be monitored and tuned.
2. Choose *Monitor* from the *Functions* menu (**Alt U O**). Current data from the module appears above the output line.
3. Examine any function block outputs of interest. To tune function code specifications, choose *Modify Specs* from the *Functions* menu (**Alt U S**). A function code editor appears similar to that shown in Figure 7-1. Only tunable parameters are shown. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

If tunable parameters are modified and the dialog box accepted, the changes are transmitted to the module.

SECTION 6 - COMPILE, CROSS-REFERENCE, AND BIND

INTRODUCTION

This section describes the use of the compiler, cross-referencer, and binder utilities. The compiler and binder utilities prepare module configurations for downloading to the target modules. The cross-referencer performs a documentation function. These utilities are launched from the module tools executive, but are DOS applications.

COMPILER

After designing a configuration using the GMC application, it must be compiled before it can be downloaded to the target module. The compiler scans all **CAD** files (sheets) for a given module and generates a single configuration file (**CFG**) that can be downloaded to the module. It also fills in function code block input specifications for function codes connected by signals and checks the configuration for errors. The compiler accepts single or multiple module configurations for processing.

CROSS-REFERENCER

Configurations often require multiple sheets and commonly a signal on one sheet continues to a location on another sheet. The cross-referencer utility uses the text descriptors associated with IO connectors and reference connectors to automatically fill in the module, sheet number, and border location in the reference. Cross referencing is not required, but is recommended as a documentation aid. The cross referencing strings make it easier to follow the control logic from sheet to sheet and to verify the input/output references. The cross referencing information becomes a permanent part of the sheet (**CAD** file). Since cross referencing modifies the sheets, the configuration must be compiled after performing cross referencing. Refer to **PLACING AND DRAWING ENTITIES** in Section 5 for information on preparing a configuration drawing for using the cross-referencer.

The cross-referencer scans an entire configuration drawing for input/output references with matching text descriptors. When a match is found, a cross-reference string is dropped at both places which links the two by module ID, sheet number, and row-column border location in the format:

mmss-yy.xx

where:

<i>mm</i>	Two unique characters identifying each module (defined in the MHD file).
<i>ss</i>	Sheet number.
<i>yy</i>	Y coordinate (with respect to the border coordinates).
<i>xx</i>	X coordinate (with respect to the border coordinates).

The sheet borders have locators (x,y coordinates) printed in the margins and are the coordinates the cross-referencer uses. To trace an input/output symbol that has been cross-referenced, use the cross-reference string from a known input/output symbol to identify the sheet corresponding to the given module and sheet number. Then examine the region on the sheet within the given coordinate boundaries for a symbol with an identical text descriptor.

BINDER

Within a sheet, function code input parameters are filled in using the *Signal* tool by connecting the output of one function code to the input of another. Among sheets of the same module configuration, function code input parameters are filled in by the compiler. Function codes may also need to communicate across modules to implement a control strategy. In this case, entering function code input parameters is called binding. The binder utility performs this function automatically. Otherwise, the function code input parameters must be entered manually.

Binder fills in the specifications of IO connector function codes by matching text descriptors associated with these function codes with OREFs associated with other function codes. It also detects any unmatched inputs or any type mismatches between matched IO connector/OREF pairs. Refer to **PLACING AND DRAWING ENTITIES** in Section 5 for information on preparing a configuration for the binder.

A module configuration must be compiled prior to using binder. The compiler generates **BND** and **REF** files for each module. These files contain the information needed by the binder utility.

STARTING THE COMPILER, CROSS-REFERENCER, AND BINDER

These utilities are grouped together under a common item on the module tools executive *Project*, *Loop*, and *PCU (Compile/Xref/Bind)* and *Module* menu (*Compile/Xref*). Each utility must be run separately and the user is responsible for sequencing

them correctly. Compiling must be performed before binding since the compiler generates files used by the binder utility. Cross referencing can be performed at any time. Compiling must be performed after cross referencing and binding since these utilities modify the CAD sheets. This means that compiling may need to be performed more than once. The recommended order is:

1. Compile.
2. Cross-reference.
3. Bind.
4. Compile.

Cross referencing and compiling can be performed at the module, PCU, loop, and project level; binding can be performed at the PCU, loop, and project level. In operation, if errors are found, they are logged and the utility continues processing the files.

NOTES:

1. This procedure assumes that the module tools executive is running and is the active application.
2. The cross-referencer, compiler, and binder use only the latest revision of a sheet if there is more than one.
3. The compiler requires that the expanded memory manager driver be passed a *no page frame* parameter (rather than a *no EMS* parameter). Refer to Section 2 for information on how to do this. If not done, the compiler will halt with a 286.2230 error, mistakenly specifying that it is out of memory.

To process one or more module configurations:

1. Select the project, loop, PCU, or module from the project tree.

NOTE: Cross referencing and binding should be performed on an entire group of modules that communicate with each other. Otherwise, unresolved cross-references and binder errors may result.

2. Choose *Compile/Xref/Bind* from the *Project, Loop, or PCU* menu or *Compile/Xref* from the *Module* menu. The *Compile/Xref/Bind* dialog box appears.

Select the utility to be run and if *Cross Reference* is checked, the following options can be checked:

Generate OUT File

Generate OUT File causes the cross-referencer to generate the **I90XREF.OUT** file. Refer to **CROSS-REFERENCER RESULTS** for a description of this file.

Blank Unresolved Refs

Blank Unresolved Refs causes any unresolved cross-references to be cleared. This serves the following purpose: Assume a group of modules has been successfully cross-referenced. Later, a number of changes are made to the configurations which invalidate some of the cross-references. If cross referencing is performed afterwards on the entire group and *Blank Unresolved Refs* is checked, unresolved cross-references will be cleared; otherwise, these cross-reference strings will remain from the previous cross-reference. However, consider changes affecting only a subgroup of the original group and only this subgroup were cross-referenced. It may be better not to clear the unresolved cross-references since there would be a number of valid unresolved cross-references resulting from not cross referencing the entire group.

3. Click on *OK* to continue.
4. A module selection dialog box appears. Add the modules of interest to the *Target List* and click on *OK* to continue.

If errors are encountered, they are written to a log file. Refer to the following sections for information on the results of cross referencing, compiling, and binding.

COMPILER RESULTS

After a successful compile, a downloadable configuration file (*module_name.CFG*) resides in the module directory. The **CFG** file can be downloaded to its respective module using *Communicate* from the module tools executive *Module* menu. In addition, the compiler generates *module_name.BND* and *module_name.REF* files required by the binder. These files are placed in the module directory. If errors are found during a compile, reconcile the errors and run the compiler again.

The compiler makes two passes through the file; the first pass lists the file and sheet being compiled and displays errors found and their absolute coordinate locations (x, y). The second pass lists the module type, number of blocks configured, RAM and NVRAM utilization, compilation time, and the number of errors found. The information produced from each pass is written to a *module_name.LOG* file located in the module directory. To view and print the log file, use the *Log Viewer* from the module tools executive *Utilities* menu.

CROSS-REFERENCER RESULTS

After successfully cross referencing the sheets, each sheet contains appropriate module, sheet, and border locations for each IO connector and reference connector.

The cross-referencer also generates the following files:

I90XREF.OUT	Contains the results of running the cross-referencer. The <i>Generate OUT File</i> option in the <i>Compile/Xref/Bind</i> dialog box must be checked to generate this file. It includes the following columns: Description - Lists all the reference descriptors found on the sheet, PCU, or loops, depending on the selection made from the project tree. Source - Lists the coordinates of the source reference. Destination - Lists the coordinates and sheet number of the destination reference.
I90XREF.ERR	Contains any errors found by the cross-referencer.
I90XREF.LOG	Lists everything that happened while running the cross-referencer.

These files reside in the module directory. To view and print these files, use the *Log Viewer* from the module tools executive *Utilities* menu.

The cross-referencer makes two passes. The first pass identifies errors in the files being cross referenced. If an error is found, a description of the error appears on the screen and is written to the error log file. The second pass enters the cross referencing information (module, sheet, and border location) into the sheets (**CAD** files). Since the cross-referencer modifies the **CAD** files, the configuration must be compiled again.

If the cross-referencer detects any errors when opening or creating a file, a message identifies the file causing the error. The cross-referencer attempts to continue executing. If the directory does not contain the necessary files, the cross-referencer prompts to change to the correct directory.

NOTE: If inconsistencies occur due to cross-reference descriptors, cross-reference descriptors can be changed manually in the GMC application.

When the cross-referencer is finished, control returns to the module tools executive. If any errors occur, a message appears indicating that errors have occurred. If the binder or compiler are selected to run, the module tools executive runs them.

BINDER RESULTS

After binding, the input specifications for function codes that receive input from another module are filled in for the **CAD** files

(sheets). Binder produces a log file (***I90BIND.LOG***) that lists errors found during the binding process. The log file is stored in the project directory. To view and print the log file, use the *Log Viewer* from the module tools executive *Utilities* menu.

After correcting any errors, run the binder again. Since the binder modifies the configuration, it must be compiled afterwards.

SECTION 7 - CONFIGURATION MAINTENANCE

INTRODUCTION

This section describes how to use the configuration maintenance utility (CMU) to perform the following functions:

- View the project structure.
- View module file and hardware configurations.
- Edit module blocks.
- Download configurations from disk files to modules.
- Save module-resident configurations to disk files.
- Verify configurations.
- Edit module configuration files.
- Generate module reports.

STARTING THE CMU APPLICATION

The CMU application can be started from the executive, the application launcher, or directly from Windows.

To start the CMU application, do one of the following:

Executive	Choose <i>Configuration Maintenance</i> from the module tools executive <i>Module</i> menu ([Alt] [M] [N]).
Application Launcher	Click on the <i>CMU</i> icon in the application launcher or choose it from the <i>Application</i> menu ([Alt] [A] [N]).
Windows 3.1 or Windows NT 3.51	Double click on its icon in the <i>Bailey WinTools</i> group.
Windows 95 or Windows NT 4.0	Choose <i>CMU</i> from the <i>Bailey WinTools</i> menu.

EXITING THE APPLICATION

Exiting the CMU application closes the application window. If there are unsaved changes to a configuration file, a prompt appears to save the file before exiting.

To exit the CMU application, choose *Exit* from the *File* menu ([Alt] [F] [X]).

APPLICATION WINDOW

This section describes the CMU application window, the CMU menus, choosing commands, and how to access on-line help.

Parts of the Application Window

The CMU application runs under the Windows operating environment. The parts of the CMU application window are shown in Figure 7-1 and described in the following paragraphs.

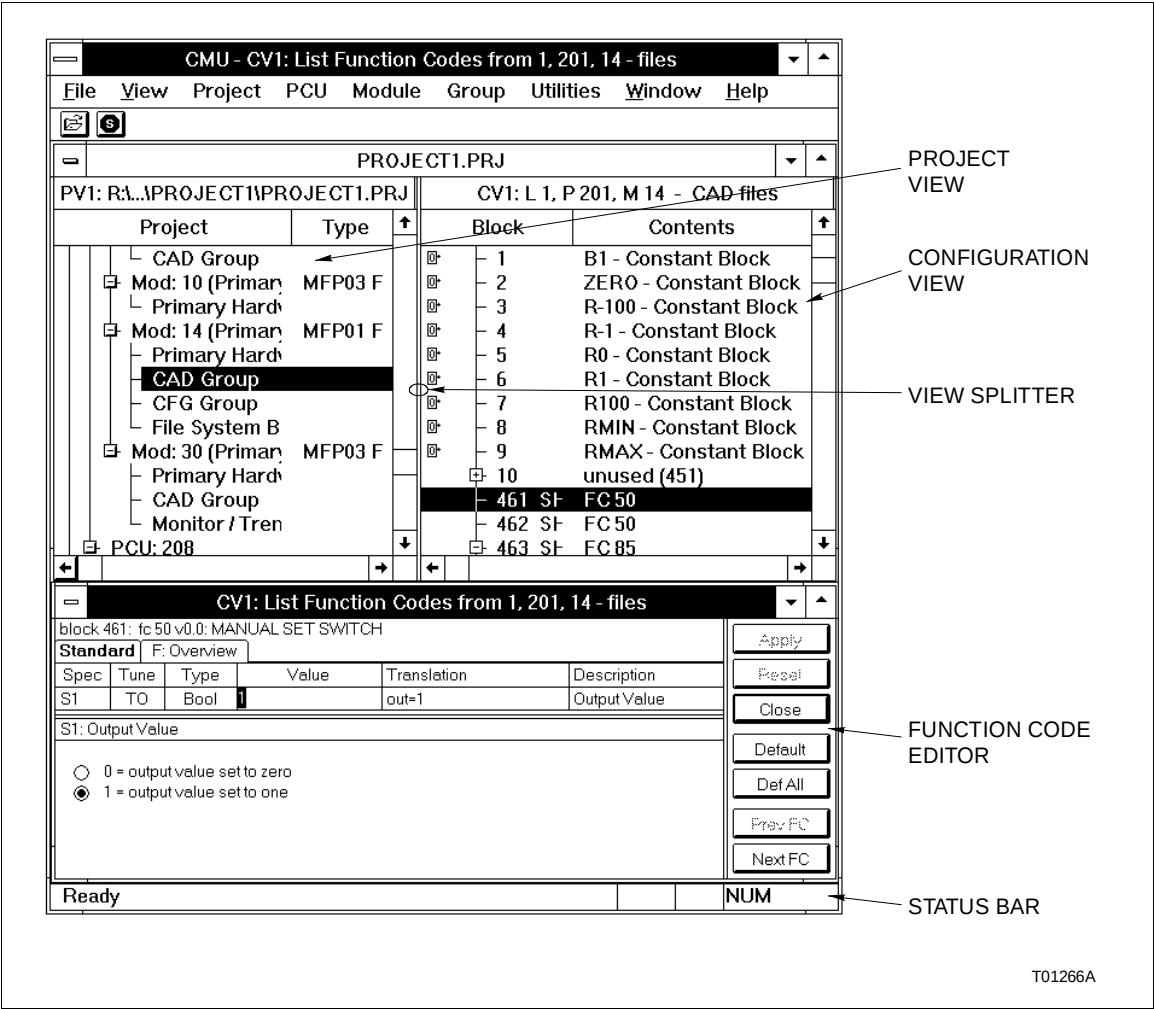


Figure 7-1. CMU Application Window

- Title Bar** The title bar is a standard Windows title bar showing the application title. The project name appears in a project window title bar. If the project window is maximized, this information appears appended to the application title.
- Menu Bar** The menu bar contains the CMU application menus. Each CMU command is listed as an item on the appropriate menu. Menu selections are made by clicking on the menu and selecting an item, using the menu keys **Alt** **menu key** **item key**, or using a shortcut key combination if one is available.
- Tool Bar** The tool bar provides an alternate and more convenient method of choosing commands. File and scan commands are

represented by an icon on the tool bar. The tool bar appears only when enabled using its properties setting. To choose a command from the tool bar, point and click on the icon.

Application Work Space

The application work space contains one project window and up to four function code edit windows. The project window contains up to two project views and two configuration views. A function code edit window can exist for each of the two project views and each of the two configuration views. If a window is minimized, it appears as an icon at the bottom of the application work space.

Project View

A project view shows a project tree similar to the executive. The project tree has at its nodes the various items defined in the project: for example, loops, PCUs, and modules. Additional CAD, CFG, and hardware groups are used to list and maintain configurations. It can be expanded and contracted to show more or less detail and can be filtered in various ways to show only specific information. Refer to **PROJECT OPERATIONS** for information on the project tree and refer to **WINDOW OPERATIONS** for information on manipulating the view. One purpose of the project tree is to make menu items available by selecting a particular group; for example, the *Module* menu appears when a *Module* group is selected. Up to two views of the same project can be displayed by splitting the view. Each view can be filtered, scrolled, and otherwise manipulated separately.

Configuration View

A configuration view can show any of the following: a *CAD Group*, a *CFG Group*, *Primary Hardware*, or *Backup Hardware*. The configuration views show the function blocks in a configuration. The configuration tree can be expanded and contracted and filtered in various ways. Up to two views can be displayed at once by splitting the view. Refer to **WINDOW OPERATIONS** for information on viewing a configuration and manipulating the view. Refer to **CAD SHEET OPERATIONS**, **CONFIGURATION OPERATIONS**, and **HARDWARE OPERATIONS (PRIMARY AND BACKUP)** for information on filtering each type of view.

Function Code Editor

The function code editor displays a function code's specifications and values in a window from which they can be listed or edited. The function code editor is called whenever a function code is added to a block or the block is modified. Up to four function code editors can be displayed within the CMU application: one for each of the project views and one for each of the configuration views. For information on using the function code editor, refer to **FUNCTION CODE EDITOR**.

View Splitter

The percentage of space within the project window given to each of the views can be adjusted using the view splitter.

Status Bar

The status bar appears at the bottom of the application window. It displays messages on the current status of the CMU

application and brief help instructions when working in a view. The status bar appears only when enabled using its properties setting.

The display and behavior of certain features in the CMU application can be controlled through the *Properties* item on the *File* menu. Refer to **PROPERTIES** for more information.

CMU Application Menus

The following paragraphs briefly describe each CMU menu. For a complete description of each menu item, refer to the subsections that follow.

NOTE: The same menu item may appear on more than one menu; however, its operation may be somewhat different depending on the menu from which it is chosen. Refer to each menu item description for differences in operation.

A menu item followed by a right pointing triangle indicates a cascading menu. That is, if chosen, another menu appears to the right of the current menu with more items to choose from. A menu item followed by ellipsis (...) indicates a dialog box appears when the item is chosen. Some items are toggles. When the item is toggled on, a check mark appears to the left of the item. Underlined characters in the menu and menu items indicate the keys used in a menu key sequence **Alt** **menu key** **item key**. A key combination to the right of an item indicates it has a shortcut key combination. Pressing the shortcut keys in combination cause the command to be performed immediately without bringing up the menu.

- | | |
|----------------|--|
| File | Refer to PROJECT OPERATIONS for information on opening and closing a project. Refer to CONFIGURATION OPERATIONS for information on saving a configuration view. Refer to Project Window for information on closing a configuration view. Refer to PROPERTIES for information on properties and automatic open on start-up. |
| View | Controls the items that appear in the project tree, configuration views, and function code editor. Refer to PROPERTIES , PROJECT OPERATIONS , HARDWARE OPERATIONS (PRIMARY AND BACKUP) , CONFIGURATION OPERATIONS , and CAD SHEET OPERATIONS for information. |
| Project | Contains project items. Refer to PROJECT OPERATIONS for information. |
| PCU | Contains PCU items. Refer to PCU OPERATIONS for information. |

Module	Contains module items. Refer to MODULE OPERATIONS for information.
Primary Hardware	Contains primary hardware items. Refer to HARDWARE OPERATIONS (PRIMARY AND BACKUP) for information.
Backup Hardware	Contains backup hardware items. Refer to HARDWARE OPERATIONS (PRIMARY AND BACKUP) for information.
Configurations	Contains configuration file (CFG) items. Refer to CONFIGURATION OPERATIONS for information.
CAD Sheets	Contains CAD sheet items. Refer to CAD SHEET OPERATIONS for information.
Monitor/Trend	Contains monitor and trend items. Refer to MONITOR/TREND OPERATIONS for information.
File System	Contains file system and program group (C and BASIC language) items. Refer to FILE SYSTEM OPERATIONS for information.
Utilities	Refer to UTILITIES for information.
Window	Controls the appearance of the windows and views within the application. Refer WINDOW OPERATIONS for information.
Help	Accesses on-line help. Refer to On-Line Help for information.

Context Menus

As a convenience, context menus are available by clicking the secondary mouse button. After selecting an item in the project view, configuration view, or function code editor, click the secondary mouse button. When clicking in the configuration view, the menus that appear also depend on the item currently selected in the project view. A menu appropriate in context for the selected item appears. Refer to **PROPERTIES, PROJECT OPERATIONS, PCU OPERATIONS, MODULE OPERATIONS, HARDWARE OPERATIONS (PRIMARY AND BACKUP), CONFIGURATION OPERATIONS, CAD SHEET OPERATIONS, FILE SYSTEM OPERATIONS, MONITOR/TREND OPERATIONS, or REPORT OPERATIONS** for information on the menu items.

Choosing Commands

All CMU commands and functions are listed as an item on one of the menus. Menu operation conforms to the Windows interface standards. The mouse or keyboard can be used to choose

an item from a menu. When a command is chosen, brief help instructions appear in the status line.

Mouse To use the mouse:

1. Click on the menu containing the command item. The menu drops down to reveal a list of the menu items. The first item is selected by default.
2. Do one of the following:
 - Point to an item and click. The command is carried out. If the mouse button is held down, the status line displays a brief description of the item; the command is carried out if the mouse button is released.
 - Drag the highlighted bar through the list and release the mouse when an item is highlighted. Before the mouse button is released, the status line provides a brief description of the item.

A command can be canceled while the mouse button is pressed by moving the mouse pointer off the item before releasing the mouse button.

Each menu and each menu item has an underlined letter that indicates the key sequence to use to choose the item. To use the menu key sequence, sequentially press and release the keys in the sequence **Alt** **menu key** **item key** where **menu key** is the underlined menu character and **item key** is the underlined item character. The menu is opened and the item chosen.

Keyboard To use the **Alt** keyboard cursor key sequence:

1. Press **Alt**. The *File* menu drops down.
2. Do one of the following:
 - To select an item from the *File* menu, use the **↑** and **↓** keys to select the item and press **Enter** to choose it. While an item is selected and before it is chosen, the status line provides a brief description of the item.
 - To select an item from another menu, use the **←** and **→** keys to select a menu, then the **↑** and **↓** keys to select an item. Press **Enter** to choose the item. While an item is selected and before it is chosen, the status line provides a brief description of the item.

The mouse, menu key sequence, and **Alt** keyboard cursor key methods can be used together in any combination that makes sense to do so.

Some commands also have a shortcut key combination that bypasses the menu system. If a command has a shortcut key combination, it is listed to the right of the menu item. The shortcut key combinations are also given in the command description. When a shortcut key combination is used, the command is performed immediately; the menu is not brought up.

Shortcut Keys

To use a shortcut key combination, press all keys in the combination simultaneously. For example, **Ctrl-S** immediately saves the document.

On-Line Help

On-line help is available for the CMU application. It provides command descriptions, procedures and reference material similar to that found in the instruction. The on-line help can be used for convenient and quick access to the CMU application operating procedures. The help system conforms to the standard Windows conventions for on-line help. Refer to **ON-LINE HELP** in Section 3 or the Windows documentation for more information.

WINDOW OPERATIONS

There are two types of windows within the CMU application window: the project window and function code editor window. One project window can be opened. Within the project window there can be two project views and two configuration views. Up to four function code editor windows can be opened, each corresponding to one of the four views within the project window.

Each type of window includes a title bar, control menu, maximize and minimize buttons, and scroll bars. Operation of these components conforms to the Windows conventions and complete information can be found in the appropriate Windows documentation. Project and function code edit windows have the following features:

- The project name appears in the project window title bar. The corresponding view (PV1, PV2, CV1, or CV2) and loop, PCU, and module address appear in the function code editor window title bar. This information also appears in the CMU application title bar for the active window.
- The project window can contain two project views, designated PV1 and PV2. Both views are of the same project, but can be scrolled and viewed separately. The project window can also contain up to two configuration views, designated CV1 and CV2. A configuration view can be of the primary or backup hardware, the **CAD** sheets, or the **CFG** configuration.

- The windows can be arranged by tiling or cascading (from the *Window* menu). If the windows are cascaded, the active window appears in the forefront.
- If a window is minimized, it appears as an icon at the bottom of the CMU application work space; it is visible if the CMU application work space is not covered by a window.
- The windows can only be moved within the CMU application window.

The project window is discussed in the following section. Refer to **FUNCTION CODE EDITOR** for information on the function code editor window.

Project Window

The project window can contain up to four views: two project views and two configuration views. The project views are different views of the same project; the configuration views can be of the same or different primary hardware, backup hardware, module configuration file (**CFG**) or **CAD** sheet configuration. To make a view active, click anywhere within the view.

PROJECT VIEW

The project view is a tree showing the nodes, groups, and hardware within a project. Figures 7-2 shows an example project view.

Selecting An item in the project tree is selected in order to perform some operation on it. Either the mouse or keyboard can be used to select an item. When an item is selected, its text appears highlighted.

To select an item using the mouse:

1. If necessary, expand the project tree to make the item visible.
2. If necessary, use the scroll bars to bring the item into view.
3. Point to the item and click.

To select an item using the keyboard:

1. If necessary, expand the project tree to make the item visible.
2. Use any of the following keys to bring the item into view: **Home**, **End**, **Page Up**, **Page Down**, **↑**, **↓**.
3. Highlight the node to be selected using **↑** and **↓**.

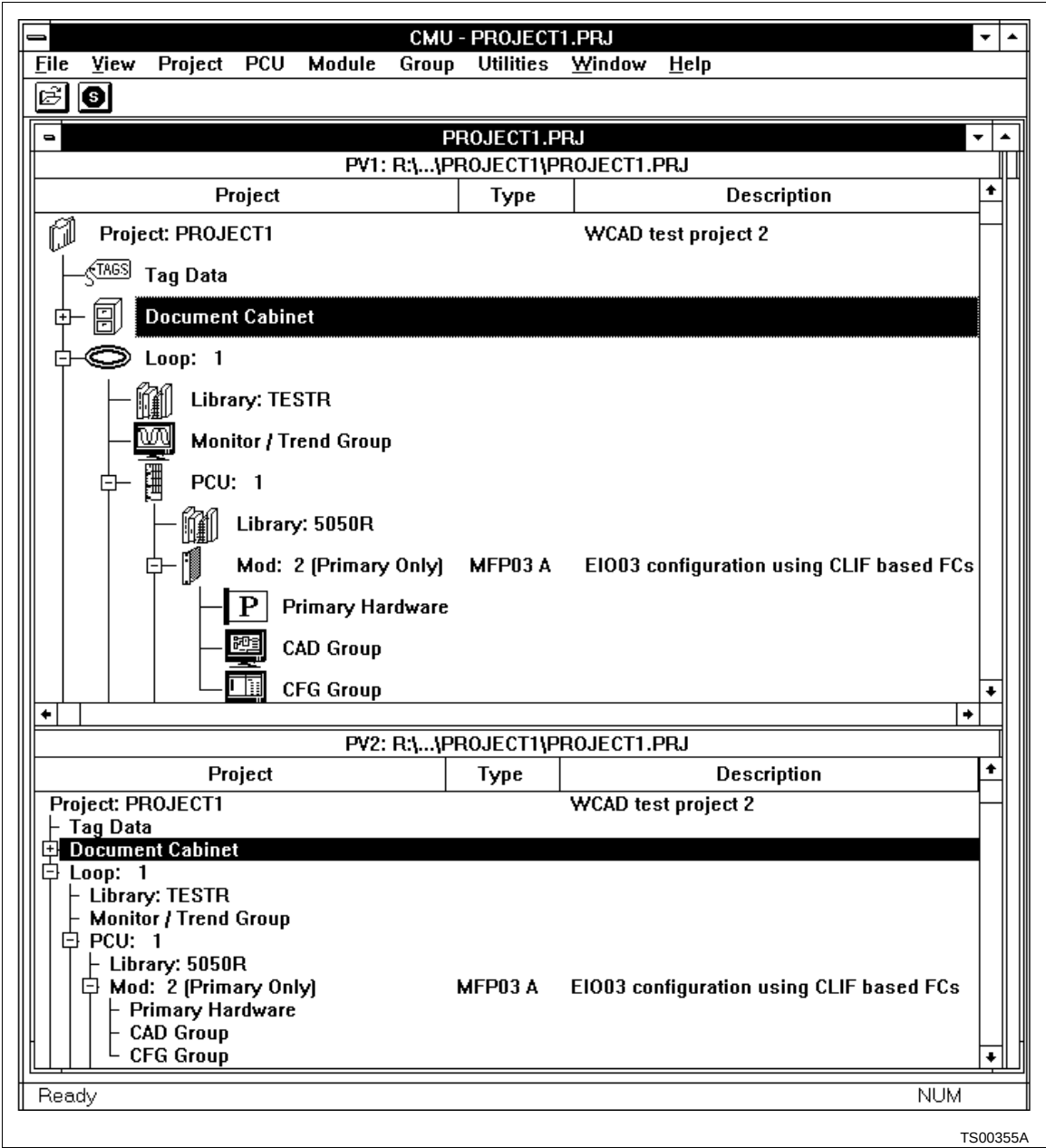


Figure 7-2. Project View

Expanding and Collapsing

The project tree and its nodes can be expanded for viewing and selection purposes. Nodes can be collapsed to make navigation in the project tree easier and to focus on specific items of interest. A node can be fully or partially expanded. If a node is fully expanded, all items under the expanded node are also expanded. If a node is partially expanded, it is only expanded down to the next node level. For example, if a project is fully expanded, all nodes, groups, and other items in the project are visible. If a project is partially expanded, only the items directly under the project are visible: tag data, loops, and if present,

document cabinets, and monitor trend groups. PCUs, modules, and other items under the loops are not visible.

Any node that can be expanded has an associated node symbol. The node symbol has a + or - indicator. A + indicator means that it can be expanded. A - indicator means that it cannot be expanded or is already expanded.

To partially expand or collapse a node using the mouse, do one of the following:

- Click on the node symbol.
- Double click on the node label. This is the text describing what the node is: *Project*, *Loop*, *PCU*, or *Mod*.

To partially expand or collapse a node using the keyboard:

1. Select the node.
2. Do one of the following
 - Press **Enter** to partially expand or collapse it.
 - Press **+** on the numeric keypad to partially expand it.
 - Press **-** on the numeric keypad to collapse it.

To fully expand a node, use the mouse or keyboard methods while holding down **Ctrl**. For example, **Ctrl-Enter** from the top level of the project tree expands all levels with this one action.

NOTE: If a node is partially expanded, it cannot be fully expanded. Any of the previous actions will collapse it. It can then be fully expanded.

Filtering The view can be filtered, for example, to show hardware only or specific module types. Refer to **PROPERTIES** and **PROJECT OPERATIONS** for information.

Splitting The view can be split to show two views of the same project using the *Split* command. The *Split* command operates as a toggle. When split, a checkmark appears beside the *Split* menu item. Each view can be independently scrolled, items selected, and filtered.

To split or unsplit the project view, choose *Split* from the *Window* menu (**Alt W S**) with the project view active.

Sizing The project views can be sized within the project window using the view splitters. One view splitter is available to adjust the amount of horizontal space given to the project views and configuration views. The other view splitter adjusts the amount of vertical space given to each project view.

Column Splitters The column splitters adjusts the amount of space given to the *Project*, *Type*, and *Description* columns within the view.

CONFIGURATION VIEW

A configuration view shows the following information within a configuration (**CAD** sheets, **CFG** file, or hardware): constant blocks available in the module, function codes at assigned block addresses, function codes at unassigned block addresses, function code outputs, and unused block addresses. Figure 7-3 shows an example configuration view. The *Block* column shows the block addresses (and sheet numbers for **CAD** views); the *Contents* column shows the constant blocks, function code numbers, and function code outputs. Unused addresses are indicated by *unused*. If unused blocks are collapsed, the number of unused blocks in a series of unused blocks is shown in parenthesis. The actual configuration view depends on filtering options.

The configuration view is used to visualize and navigate through the function code blocks making up the module's configuration. Maintenance operations can be performed from the configuration view on any of the configuration types (**CAD** sheet, **CFG** file, or hardware).

The configuration can be partially or fully expanded.

If the configuration is partially expanded, then only the following are shown:

- Blocks that are the base address of the function block outputs.
- The first in a series of unconfigured blocks.

If the configuration is expanded from a base address block, then all blocks used by the function code are shown.

If the configuration is expanded from the first in a series of unused blocks, then all unused block addresses are shown up to the next configured block.

If the configuration is fully expanded from the module level, all block addresses for the entire module are shown.

NOTE: Expanding the configuration to show all function blocks in the module or any large range of blocks can take a considerable amount of time.

Viewing To view a configuration using the mouse:

1. Select a *Primary Hardware*, *Backup Hardware*, *CAD Group*, or *CFG Group* item.

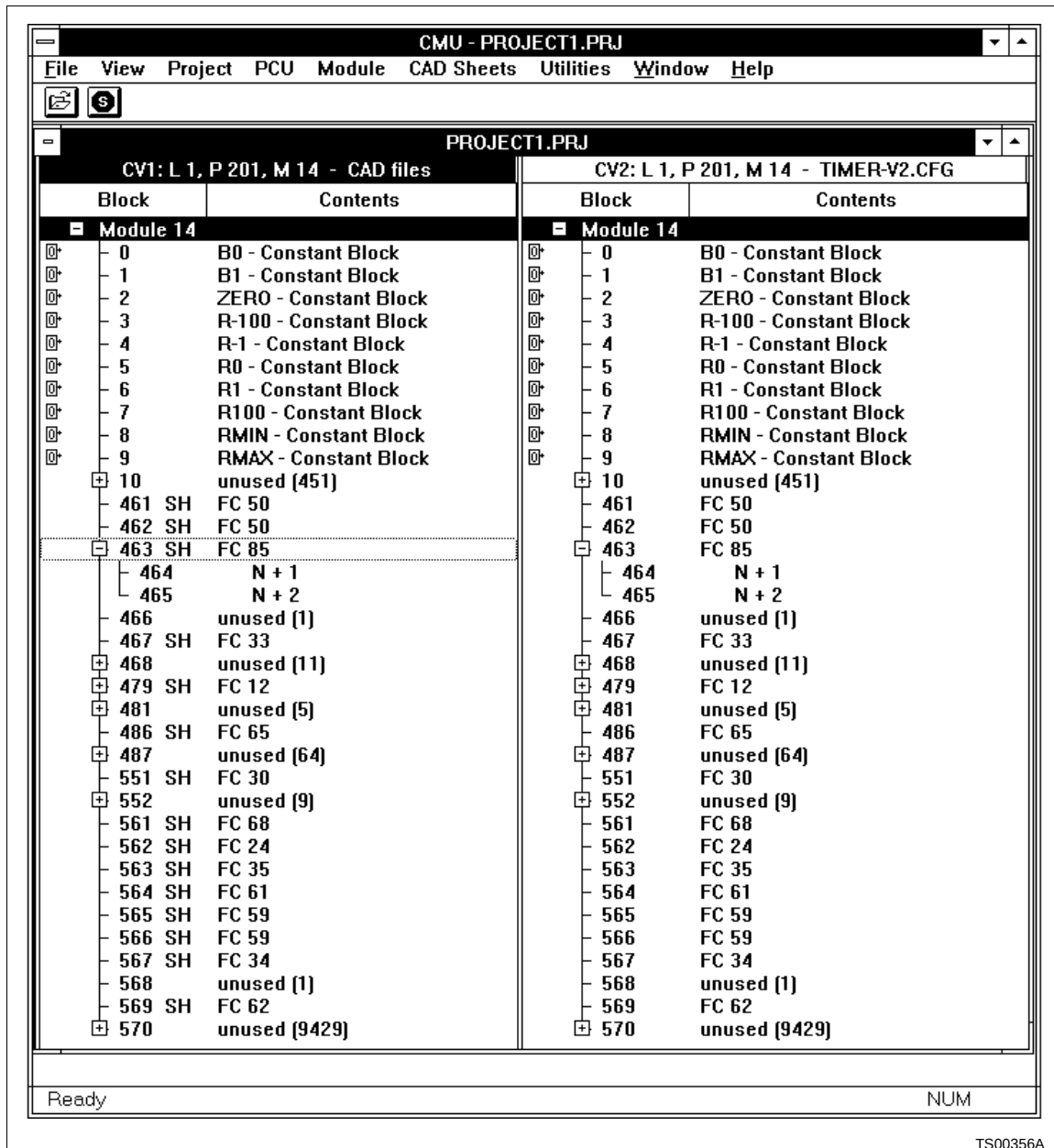


Figure 7-3. Configuration View

2. Point to and drag the item to the configuration view area.

To view a configuration using the keyboard:

1. Select a *Primary Hardware*, *Backup Hardware*, *CAD Group*, or *CFG Group* item.
2. Choose *View Configuration* from the respective *Primary HW*, *Backup HW*, *CAD Sheets*, or *Configurations* menu or use the context menu.

The following can happen when generating a configuration view:

- If the view is the first view opened, it appears in the configuration view area.
- If the configuration view is not split and another view is already open, the current view is overwritten.
- If the configuration view is split and another view is already open, the mouse method replaces the view to which the new view is dragged.
- If the configuration view is split and the keyboard method is used, a prompt appears requesting into which configuration view to place the new view.

Selecting

An item (function block address) in the configuration is selected in order to perform some operation on the item. Either the mouse or keyboard can be used to select an item. When an item is selected, it appears highlighted.

To select an item using the mouse:

1. If necessary, expand the view to make the item visible.
2. If necessary, use the scroll bars to bring the item into view.
3. Point to the item and click.

To select a item using the keyboard:

1. If necessary, expand the view to make the item visible.
2. Use any of the following keys to bring the item into view: **Home**, **End**, **Page Up**, **Page Down**, **↑**, **↓**.
3. Highlight the item to be selected using **↑** and **↓**.

Expanding and Collapsing

The configuration can be expanded for viewing and selection purposes. It can be collapsed to make navigation easier and to focus on specific items of interest. The module node can be fully or partially expanded. If it is fully expanded, all used and unused block addresses shown. If it is partially expanded, only used blocks and the first in a series of unused blocks are shown.

Any node that can be expanded has an associated node symbol. The node symbol has a + or - indicator. A + indicator means that it can be expanded. A - indicator means that it cannot be expanded or is already expanded.

To expand or collapse a node using the mouse, do one of the following:

- Click on the node symbol.
- Double click on the node label. This is the text describing what the node is: the function code number or unused.

To expand or collapse a node using the keyboard:

1. Select the node.
2. Do one of the following
 - Press **Enter** to partially expand or collapse it.
 - Press **+** on the numeric keypad to partially expand it.
 - Press **-** on the numeric keypad to collapse it.

To fully expand the module, use the mouse or keyboard methods while holding down **Ctrl**.

NOTE: If a node is partially expanded, it cannot be fully expanded. Any of the previous actions will collapse it. It can then be fully expanded.

Filtering	The view can be filtered, for example to show only configured or certain types of function codes. Refer to PROPERTIES, HARDWARE OPERATIONS (PRIMARY AND BACKUP), CONFIGURATION OPERATIONS , and CAD SHEET OPERATIONS for complete information.
Replacing	To replace a view with another view, open a new view.
Closing a View	To close a view, choose <i>Close Configuration View</i> from the <i>File</i> menu.
Splitting	The configuration view area can be split into two separate views to show two different configuration views using the <i>Split</i> command. The <i>Split</i> command operates as a toggle. When split, a checkmark appears beside the <i>Split</i> menu item. Each view can be independently scrolled, items selected, and filtered. To split or unsplit the configuration view, choose <i>Split</i> from the <i>Window</i> menu (Alt W S) with the configuration view active.
Sizing	The configuration views can be sized within the project window using the view splitters. One view splitter is available to adjust the amount of horizontal space given to the project views and configuration views. The other view splitter adjusts the amount of space given to each configuration view.

- | | |
|------------------------|--|
| Column Splitter | The column splitter adjusts the amount of space given to the <i>Block</i> and <i>Contents</i> columns within the view. |
| Saving a View | <p>If the configuration view is of a module configuration CFG file, it can be saved to disk (e.g., after making changes). To save a configuration view to disk, do one of the following:</p> <ul style="list-style-type: none"> • To save it under the same name, choose <i>Save Configuration View</i> from the <i>File</i> menu. • To save it under a new name, choose <i>Save Configuration View As</i> from the <i>File</i> menu. |

Multiple Document Windows

- | | |
|----------------------------------|---|
| Switching between Windows | <p>One project window and up to four function code edit windows can be opened within the application work space at any one time. If a window is maximized, it is the only window visible in the work space.</p> <p>To switch between windows, do one of the following:</p> <ul style="list-style-type: none"> • If the window to be made active is visible, click anywhere on the window. It becomes the active window. • A list of open windows appears under the <i>Window</i> menu. Choose the window from this list. • If the window is not visible, arrange the other windows (by tiling, cascading, moving, or sizing) to make it visible first, then click anywhere on the window. • If the window has been minimized, double click on its icon. |
| Going to the Next Window | <p>To make the next window the active window, press Ctrl+F6. The next window in the order in which they were opened becomes the active window.</p> |

Sizing, Moving, and Arranging Windows

- | | |
|----------------------------|--|
| Maximizing a Window | <p>Document windows can be arranged within the application work space. They can be maximized, minimized, tiled, cascaded, and otherwise arranged by the user by sizing and moving. Refer to the Windows documentation for complete information.</p> <p>To maximize a window, click on the window maximize button. This is the up-arrow button to the far right of the group title bar. When a window is maximized, it takes up the entire work space and its name appears in the application title bar. It does not have a separate window in this case.</p> |
|----------------------------|--|

Restoring a Maximized Window	To restore a maximized window, click on the restore button. This is the double-arrowhead button located underneath the application maximize button. The window appears with the same size and position as it had before being maximized.
Minimizing a Window	To minimize a window, click on the window minimize button. This is the down-arrow button located to the right of the window title bar.
Restoring a Minimized Window	To restore a minimized window, double click on the window icon. The window appears with the same size and position as it had before being minimized.
Tiling Windows	To tile windows, use <i>Tile</i> from the <i>Window</i> menu.
Cascading Windows	To cascade windows, use <i>Cascade</i> from the <i>Window</i> menu.
Sizing Windows	To size a window, point to the window corner. When the cursor changes to a double ended arrow, drag the border to resize the window.
Moving Windows	To move a window, point to the window title bar and drag.
Arranging Window Icons	To arrange window icons, use <i>Arrange</i> from the <i>Window</i> menu. The icons are arranged along the bottom of the application work space.

Function Code Edit Window

Up to four function code edit windows can be opened within the application work space. Refer to **FUNCTION CODE EDITOR** for information on the function code edit window.

PROPERTIES

Properties set default parameters for the appearance and operation of the configuration maintenance utility. Properties are categorized as follows:

- General.
- Project View.
- Configuration View.
- Report.

Each of these property categories is discussed in the following subsections. The property dialog box is accessed from the *File* menu.

After setting any of the properties, do one of the following from the properties dialog box:

- Click on *Save As Default*, then *OK* to save these settings and return to the CMU window. These settings are applied

and are used the next time the CMU application is launched.

- Click on *OK* to return to the CMU window. These settings are valid for the current session only.
- Click on *Cancel* to return to the CMU window without making any changes.

General

General properties control the display of the tool bar and status bar, whether the last opened project is automatically opened on start up, and set the default text editor.

To access the *General* properties page, choose *Properties* from the *File* menu. The *General* properties page appears.

Tool Bar

To toggle the tool bar display on and off, click *Toolbar* on the *General* properties page. If the check box is on, the tool bar appears underneath the menu bar. Refer to ***Parts of the Application Window*** for information.

Status Bar

To toggle the status bar display on and off, click *Status Bar* on the *General* properties page. If the check box is on, the status bar appears at the bottom of the application menu. Refer to ***Parts of the Application Window*** for information.

Auto Open on Start Up

Automatic open on start up allows the last opened project to be automatically opened when the CMU application is launched.

To use the automatic open on start up feature, do one of the following:

- From the *General* properties page, set the check box on *Automatically open last project at start-up*.
- Toggle it on by choosing *Auto Open on Start Up* from the *File* menu.

Default File Viewer

This sets the default file viewer used for log files, reports, trace, and calculation. The file viewer can be any Windows compatible application (***notepad.exe***, ***write.exe***, etc.) that can read a text file.

To set the default file viewer, type the application path and name into the text box on the *General* properties page or use the *Browse* button to search for an application.

Project View

Project view properties control the appearance of items in the project tree.

To access the *Project View* properties page:

1. Choose *Properties* from the *File* menu. The *General* properties page appears.
2. Click on the *Project View* tab. The *Project View* page appears.

The *Project View* properties page also appears if the project view is the current view and *Advanced Filter* is chosen from the *View* menu.

Bitmaps Items in the project tree have an associated bitmap that can be displayed or not as a user preference. To display the bitmaps, set the *View Bitmaps* checkbox on the *Project View* properties page.

Hardware Hardware refers to the modules and their associated configuration groups. Other items that can appear in the project tree include libraries, monitor/trend groups, and documentation folders. Whether to display only hardware related items or all items in the project tree can be set as a user preference.

To display hardware related items only, do one of the following:

- Set the *View only Hardware Components* checkbox on the *Project View* properties page.
- With the project view as the current view, choose *View Hardware Only* from the *View* menu.

Module Filter The view of the project tree can be filtered by module type, allowing only modules and related items of the selected module types to be displayed. All other nodes in the project tree are collapsed.

To use module filtering:

1. Set the *Use Filter* checkbox on the *Project View* properties page to enable the buttons and pick list in the *Module Filter* frame. All available modules are displayed in the pick list.
2. Selected module types appear highlighted in the pick list. Do any of the following to select module types:
 - To select all module types, click on *Select All*.
 - To clear all module types, click on *Clear All*.
 - To select an individual module type, click on the type in the pick list. To continue selecting module types, hold down **Ctrl** while pointing and clicking.

- To select a range of module types, click on the first module type to select and drag to the last module type in the range. Alternatively, click on the first module type to select, point to the last module type in the range, press **Shift** and click.

Module filtering can also be turned on and off by choosing *Use Module Filter* from the *View* menu while the project view is the current view.

Configuration View

Configuration view properties control the appearance of items in the configuration tree diagram.

To access the *Configuration View* properties page:

1. Choose *Properties* from the *File* menu. The *General* properties page appears.
2. Click on the *Configuration View* tab. The *Configuration View* page appears.

The *Configuration View* properties page also appears if the configuration view is the current view and *Advanced Filter* is chosen from the *View* menu.

Unused Blocks and Defined Function Codes

This property permits viewing only unused blocks, only blocks with defined function codes, or both.

To show unused blocks only, select *Show unused blocks* on the *Configuration View* properties page. Alternatively, choose *Unused Blocks* from the *View* menu with a configuration view as the current view.

To show blocks with defined function codes only, select *Show defined Function Codes* on the *Configuration View* properties page. Alternatively, choose *Defined FCs* from the *View* menu with a configuration view as the current view.

To show both unused blocks and blocks with defined function codes, select *Show both* on the *Configuration View* properties page. Alternatively, choose *Both* from the *View* menu with a configuration view as the current view.

Range of Blocks

The view can be limited to show a range of blocks. To show only a range of blocks,

1. Set the *Show only base blocks in the range* checkbox on the *Configuration View* properties page. The *From* and *To* text boxes are enabled.
2. Enter the desired range in the *From* and *To* text boxes.

- Function Code Filter** The view can be limited to show only blocks containing specified function code types. To do so:
1. Set the *Use Function Code Filter* checkbox on the *Configuration View* properties page. The function code numbers text box is enabled.
 2. Enter the desired function code numbers in the format shown in the dialog box.
- NOTES:**
1. The unused blocks and defined function codes, range of blocks, and function code filter properties can be used in combination with each other.
 2. Choosing *View All* from the *View* menu with the configuration view as the current view overrides all property settings and displays all blocks in the configuration.
- Order** The order property displays the blocks in the configuration view ordered numerically by block number, sheet number, or function code number. To order the configuration view, click on the desired order in the *Order by* frame of the *Configuration View* properties page.

Reports

Report properties control how reports are generated.

To access the *Reports* properties page:

1. Choose *Properties* from the *File* menu. The *General* properties page appears.
2. Click on the *Reports* tab. The *Reports* page appears.

- Trace/Calcutilization** These properties determines the type of information contained in the reports. To choose the type of information to report, click on the desired information type.

To use the configuration view filtering, set the *Use Current Configuration View Filter* checkbox. Only those function blocks passed by the filtering are traced.

To automatically view the report, set the *Automatically view the report* checkbox. The report will appear in the default file viewer set on the *General* properties page. Otherwise, the report must be opened manually.

To mark address conflict errors in the report, set the *Mark errors* checkbox.

To specify the number of columns used in the detailed trace information portion of the report, enter a number from 1 to 6 in the *Number of columns* box.

**Specification Dump
Options**

This property determines the file format used in the specification report. To set the file format, click on the desired format in the *Specification Dump Options* frame. Refer to **REPORT OPERATIONS** for information on the file formats.

PROJECT OPERATIONS

The project tree displays a hierarchical view of the module tools project structure as shown in Figure 3-4. The topmost element in the tree is the project indicated by the project description. The project node contains the project database, documentation cabinet, and loops; the loop node contains the PCU cabinets; the PCU node contains the modules; and the module node contains the primary and backup hardware, configurations, and CAD sheets. Libraries and monitor/trend groups can be added to any node. Each node has an icon (if displayed) identifying the type of node.

NOTE: The project tree displayed in the CMU application shows the same project and in a similar manner as in the module tools executive. However, the CMU application does not perform any project operations that affect its contents (other than adding modules). The module tools executive must be used to add or delete items from the project. Refer to **Section 4** for information on organizing a project.

The project tree serves three purposes: 1) By defining the project and adding nodes, it organizes the project. 2) By opening folders associated with each node, it gives access to the node contents. 3) By selecting a node, it makes utilities appropriate to the node available.

The loop, documentation cabinets, documentation folders, PCUs, and module icons show the current state of the node by changing form. If an icon is opened, the node contents are visible; if the icon is closed, the node contents are not visible. The loops, PCU cabinets, modules, libraries, and sheets can be viewed by opening the respective node.

To open a node, double click on the respective closed icon.

To close a node, double click on the respective open icon.

A selected node appears highlighted. To select a node of the project tree, click anywhere on the node of interest. The node highlights and the menu bar reflects the node type selected.

Refer to the **CMU Application Menus** for menu descriptions. The available menu items depend on the node selected.

Project Tree Structure

The module tools have been designed to facilitate a project oriented approach in configuring a control strategy. Refer to **Project Tree Structure** in Section 3 for detailed information on the project tree. The contents of each item within the project tree used by the CMU application is described in the following paragraphs.

Project	Contains the loops, tag data, and documentation cabinet. The project also maintains a number of files including the project file (PRJ) and the border information file (JHD). The project file contains the project description and location of the tag data tables. The project name and location is defined by the user when the project is created. The project file is shared by the module and console tools.
Loop	Loops contain PCUs and loop libraries. The loop address is shown next to the loop icon as <i>Loop: nnn</i> , where nnn is the loop address.
Monitor/Trend Group	Contains group definitions used by the trend, monitor, and tune application. Refer to Section 10 for information.
PCU	PCUs contain modules and PCU libraries. The PCU address is shown next to the PCU icon as <i>PCU:nnn</i> , where nnn is the PCU address.
Module	Modules contain the CAD sheet groups, CFG configuration groups, module hardware (primary and backup), and file system groups. The module icon indicates whether it is a stand alone primary, a redundant pair, or a primary/backup pair. The module address is shown next to the module icon as <i>Module: nn</i> , where nn is the module address. The module type appears after the address. The module also maintains other files in its directory: a module information file (MHD), if the module has been compiled, the compiled configuration file (CFG), and report and log files.
Primary Hardware	An icon for the primary module in a stand alone primary or primary/backup pair. An icon for both the primary and redundant module in a primary/redundant pair. It is used to perform operations on the primary module.
Backup Hardware	An icon for the backup module in a primary/backup pair that is accessible for on-line configuration. It is used to perform operations on the backup module.
CAD Group	Represents the set of individual CAD sheet files that comprise the module configuration generated by the graphical module configuration application. It is used to perform operations on the CAD files.

CFG Group	Represents one or more compiled module configuration CFG files. It is used to perform operations on the CFG files.
File System Group	Represents one or more Batch, C, or BASIC program files saved from the module hardware.

Opening a Project

To open a project:

1. Choose *Open Project* from the *File* menu. An *Open Project* dialog box appears.
2. Locate the project to be opened and click on *OK*.

Closing a Project

To close a project, choose *Close Project* from the *File* menu.

Creating a New Project

This function creates a new project in an existing directory selected by the user. It can be used if access to the hardware is needed but the project is unavailable. The **PRJ**, **JHD**, and tag database files are created.

To create a new project:

1. Choose *New Project* from the *File* menu. The *Create New Project* dialog box appears.
2. Select a path and enter a project file name, then click on *OK*. The *Job Information* dialog box appears.
3. Enter the job information by referring to Table 7-1.
4. Click on *OK* to continue.

Table 7-1. Job Information

Field	Description
Customer name	Twenty-character field for the customer name. An entry in this field is optional.
Plant site	Ten-character field for the plant site. An entry in this field is optional.
Contracting firm	Thirty-character field for the contracting firm. An entry in this field is optional.
Job number	Ten-character field for the project number. This field requires an entry. It is recommended to use the project directory name.

Table 7-1. Job Information (continued)

Field	Description
Customer order number 1	Ten-character field for the customer order number. An entry in this field is optional.
Customer order number 2	Ten-character field for the customer order number. An entry in this field is optional.

Viewing Options

These options affect the display of the project tree in the project view. The project view must be the current view to perform these operations.

Viewing All Project Components	Filters can be applied to the project view. To remove the effect of these filters, choose <i>View All</i> from the <i>View</i> menu.
Viewing Hardware Only	Rather than view all components in a project, only the modules and related components can be displayed. To do so, choose <i>View Hardware Only</i> from the <i>View</i> menu.
Module Filtering	Module filtering is set using the <i>Project View</i> properties page of the <i>Properties</i> dialog box (refer to PROPERTIES). Module filtering can be turned on and off. To do so, choose <i>Use Module Filter</i> from the <i>View</i> menu.
Advanced Filtering	Properties affecting the appearance of the project view can be configured. To do so, choose <i>Advanced Filter</i> from the <i>View</i> menu. The <i>Project View</i> properties page of the <i>Properties</i> dialog box appears. Refer to PROPERTIES for information. Alternatively, the <i>Project View</i> properties page of the <i>Properties</i> dialog box can be displayed by choosing <i>Properties</i> from the <i>File</i> menu, then clicking on the <i>Project View</i> tab.

Refreshing the Project View

Changes made to the project from the executive are not immediately reflected in the CMU project tree. To have these changes appear, the project view must be refreshed.

To refresh the project view, choose *Refresh Project View* from the *Project* menu.

Editing Project Information

The CMU application can edit the project information file containing the project description. This is the same file as used by the executive.

To edit the project information file:

1. Choose *Edit Project Info* from the *Project* menu. The *Project Information* dialog box appears.
2. Edit the project location and description as needed and click on *OK* to accept the change.

Editing the JHD File

Every project must have an associated job information file (***JHD***). Job information is project specific information that is automatically inserted into an automatic border. The CMU application can edit the job information file (***JHD***). This is the same job information file used by the executive.

To define job information:

1. Choose *Edit JHD File* from the *Project* menu.
2. The *Job Information* dialog box appears. Refer to Table 7-1 for field descriptions and fill in the fields appropriately.
3. When done, click on *OK* to save the job information in the *project_name.JHD* file in the project directory. The information entered into these fields is automatically inserted into a configuration sheet border provided the border type accepts the field.

PCU OPERATIONS

The *Inspect/Mode* change PCU operation launches the inspect application and passes it the PCU address. The inspect application appears with the PCU tab active and the modules from the selected PCU node visible in the inspect window. The modules can be inspected and their mode changed. Refer to Section 8 for more information.

To launch the inspect application:

1. Select the PCU containing the modules of interest.
2. Choose *Inspect/Mode Change* from the *PCU* menu.

MODULE OPERATIONS

These operations perform actions on a module or related file. Some operations are performed with the project view active and others are performed with the configuration view active.

Module Hardware Configuration

This function sets the type of module hardware configuration expected by the CMU application for MFC, MFP, and BRC modules. It has no effect on the actual hardware and should be set to the actual hardware configuration. This setting affects menu selections. Download operations check it and can provide notification if potential inconsistencies exist.

The module hardware configuration can be one of three types:

- Primary module only
- Primary/redundant module
- Primary/backup module

The type is indicated in the project tree by the module and hardware icon and the associated text. To set the module hardware configuration type:

1. Select the module of interest from the project tree.
2. With the project view active, choose the desired option from the *Module* menu. The project tree updates to reflect the change.

Editing Module Header (MHD)

Every module must have an associated module information file. Module information defines characteristics of a module used by the other configuration utilities. Some of this information is automatically inserted into an automatic border. The information is stored in a module information file (**MHD**). The module information is defined and the module information file created when the module is initially added to the PCU folder within the module tools executive. The module information can also be edited at any time. This procedure describes how to edit a module information file from the CMU application.

To edit module information:

1. Open the project and then open the loop and PCU nodes containing the module of interest.
2. Select the module.
3. Choose *Edit Header (MHD)* from the *Module* menu or the context menu. The *Module Info* dialog box appears. Refer to Table 7-2 for field descriptions and fill in the appropriate information.
4. When done, click on *OK* to update the module information file.

The information entered into the module information fields (except for module type and firmware revision) is automatically inserted into a configuration sheet border provided the border type accepts the field. This file can also be edited from the module tools executive.

Table 7-2. Module Information

Field	Description
Module file name prefix	Five-character module name also given to the module configuration and the module information file (MHD). It also forms the first five characters of each CAD sheet in the configuration. Any five characters can be used for the module name. For example, an abbreviation relating the purpose of the module (e.g., PUMP1). A previous method of naming modules included the loop, PCU, and module address using two characters for the loop address, two characters for the PCU address and one character for the module address. Appendix C contains translation tables to use when the numeric address exceeds the allowable number of characters. This field can only be edited when the module is initially added to the PCU folder.
Configuration description	User provided description of the configuration purpose. This will appear after the module address in the project tree.
Module type	Type of module.
Firmware revision	Firmware revision of the module.
Module ID	Two-character alphanumeric module identifier. This identifier is used by the cross-referencer and must be unique for each module in the project.
Drawing title	Drawing title.
Drawing number	Seven-character field for a drawing number.

Add Module

This function adds a module to the currently opened project.

To add a new module:

1. Choose *Add Module* from the *Project* menu. The *Add Module* dialog box appears.
2. Enter the loop, PCU, and module address for the module and click on *OK* to continue.
3. Complete the *Module Information* dialog box that appears and click on *OK* to continue. Refer to Table 7-2 for information.

The necessary loop, PCU, and module directories and the **MHD** file are created and the project tree is updated.

Module Reports

Refer to **REPORT OPERATIONS** for trace, calculation and specification reports. Refer to **Section 8** for summary, status, error, and problem reports.

Inspect/Mode Change

The *Inspect/Mode* change module operation launches the inspect application and passes it the module address. The inspect application appears with the inspect tab active and the selected module visible in the inspect window. The module can be inspected and its mode changed. Refer to **Section 8** for more information.

To launch the inspect application:

1. Select the module of interest.
2. Choose *Inspect/Mode Change* from the *Module* menu.

On-Line Configuration

The *Online Configuration* module operation launches the inspect application and passes it the module address. The inspect application appears with the on-line configuration tab active and the selected module visible in the inspect window. The module is ready to begin the on-line configuration process. Refer to **Section 8** for more information.

To launch the inspect application:

1. Select the module of interest. It must be a module with its hardware option set to *Primary/backup module*.
2. Choose *Online Configuration* from the *Module* menu.

HARDWARE OPERATIONS (PRIMARY AND BACKUP)

These operations perform actions on a module or related file. Some operations are performed with the project view active and others are performed with the configuration view active.

View Configuration

This function displays a view of the actual configuration residing in a module in a configuration view.

To view a module resident configuration:

1. Select a *Primary Hardware* or *Backup Hardware* item from the project tree and do one of the following:
 - Point to the selected *Primary Hardware* or *Backup Hardware* item and drag it to the configuration view area.
 - Choose *View Configuration* from the *Primary HW* menu or *Backup HW* menu or use the context menu.
2. A select block range dialog box appears. If a range of blocks is to be read from the module, click on *Read from Module only Blocks in Range* and fill in the desired range.
3. Click on *OK* to continue. The hardware is scanned and a configuration view containing the function codes found is displayed.

Scanning can be stopped by clicking on the *Stop* button in the tool bar.

NOTE: Firmware in certain modules can locate their first configured block immediately and the view generation proceeds quickly. Older modules must search addresses sequentially until they locate their first configured block and the view generation requires more time.

If there is a current configuration view, it is overwritten by the new configuration. A maximum of two configuration views are available. To use a second configuration view, make the configuration view the active view and choose *Split* from the *Window* menu (**Alt W S**). For more information on using the views, refer to **WINDOW OPERATIONS**.

Close Configuration View

To close a configuration view, choose *Close Configuration View* from the *File* menu.

Editing and Listing Configurations

Function code operations can be performed on a module configuration resident in the module. The operations that can be performed depend on what view is current and, if in the configuration view, whether an unused or configured block is selected.

NOTES:

1. Any operation that requires a module mode change must have the mode changed by the inspect application. If a module is not in the proper mode, a message appears prompting to launch the inspect application to do so.
2. Editing changes can be saved to a disk file (**CFG**). Refer to **Save Module Configuration** for information.

3. When editing the module hardware configuration from the project view, changes are not automatically reflected in any corresponding configuration view. The hardware configuration view must be re-loaded to view the changes.

List Function code specifications can be listed from the project view or the configuration view. List allows function code specifications to be viewed but no changes can be made. To list function code specifications:

1. Do one of the following:
 - From the configuration view, select a configured block to be listed.
 - From the project view, select the primary or backup hardware group.
2. Choose *Configure Function Code*, then *List* from the *Primary Hardware* menu or *Backup Hardware* menu or use the context menu.
3. If the listing is initiated from the project view, a dialog box appears, requesting a block number. Enter a block number and click on *OK* to continue.
4. A *Function Code Editor* for the selected block appears in which the specifications can be viewed. Refer to **FUNCTION CODE EDITOR** for information on using the function code edit window.

Tune A block in a primary module can be tuned from the project view only. The primary module must be in execute mode.

To tune a block:

1. If necessary, use *inspect* to place the module in execute mode.
2. Select the primary module to be tuned.
3. Choose *Configure Function Code*, then *Tune* from the *Primary Hardware* menu or use the context menu.
4. A dialog box appears, requesting a block number. Enter a block number and click on *OK* to continue.
5. A *Function Code Editor* for the selected block appears in which the tunable specifications can be edited. No other editing can be performed. Refer to **FUNCTION CODE EDITOR** for information on using the function code edit window.

Modify A block in a primary or backup module can be modified from the project view only. The primary or backup module must be in configure mode.

NOTE: If modifying a backup module, on-line configuration restrictions apply. Specifically, modifying specifications that would increase a function block's memory utilization are not allowed.

To modify a block:

1. If necessary, use inspect to place the module in configure mode.
2. Select the hardware item to be modified.
3. Choose *Configure Function Code*, then *Modify* from the *Primary Hardware* menu or *Backup Hardware* menu, or use the context menu. A *Modify Function Code* dialog box appears.
4. Enter the block number to be modified and click on *OK* to continue.
5. A *Function Code Editor* for the selected block appears in which the modifications can be made. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.
6. Click on *Apply* to apply the changes to the module.
7. When done, use the inspect application to return the module to the execute mode in an appropriate manner.

Add Function codes can be added to blocks in a primary or backup module from the project view only. The primary or backup module must be in configure mode.

NOTE: If adding a block to a backup module, on-line configuration restrictions apply. Specifically, blocks can only be added following the last block used within a segment.

To add function codes to blocks:

1. If necessary, use inspect to place the module in configure mode.
2. Select the primary or backup hardware item to which a block is to be added.
3. Choose *Configure Function Code*, then *Add* from the *Primary Hardware* menu *Backup Hardware* menu or context menu. An *Add Function Code* dialog box appears.

4. Enter the requested information. Refer to Table 7-3 for a description of the fields.

Table 7-3. Add Function Code

Field	Descriptions
Block address	Address at which the first function code is to be added.
Count	Number of function codes to add.
Offset	Block address offset at which each subsequent function code is added.
Function code	Function code type to add.

5. Click on *OK* to continue. A *Function Code Editor* for the selected block appears in which the specifications can be edited. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

6. Click on *Add* in the function code editor to add the function code to the indicated block address.

If more than one function code is being added, the function code editor shows the next function code. Continue until all function codes are added.

7. When done, use the inspect application to return the module to the execute mode in an appropriate manner.

Copy Function codes can be copied from one block to other blocks in a primary or backup module from the project view only. The primary or backup module must be in configure mode.

NOTE: If copying blocks within a backup module, on-line configuration restrictions apply. Specifically, blocks can only be added following the last block used within a segment.

To copy function blocks:

1. If necessary, use inspect to place the module in configure mode.
2. Select the primary or backup hardware item within which blocks are to be copied.
3. Choose *Configure Function Code*, then *Copy* from the *Primary Hardware* menu *Backup Hardware* menu or context menu. A *Copy Function Code* dialog box appears.
4. Enter the requested information. Refer to Table 7-4 for a description of the fields.
5. Click on *OK* to continue. A *Function Code Editor* for the selected block appears in which the specifications can be

Table 7-4. Copy Function Code

Field	Descriptions
Source block address	Block address of the block that is to be copied.
Destination block address	Block address at which the first block is to be copied.
Count	Number of times the source block is to be copied.
Offset	Block address offset at which each subsequent block is copied.

edited. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

6. Click on *Add* in the function code editor to add the function code to the indicated block address.

If the source block is being copied more than once, the function code editor shows the next target block. Continue until all function codes are added.

7. When done, use the inspect application to return the module to the execute mode in an appropriate manner.

Delete Function codes can be deleted from blocks in a primary module from the project view only. The primary module must be in configure mode.

To delete a function codes from a block:

1. If necessary, use inspect to place the primary module in configure mode.
2. Select the primary hardware item to which a block is to be deleted.
3. Choose *Configure Function Code*, then *Delete* from the *Primary Hardware* menu or context menu. A *Delete Function Code* dialog box appears.
4. Enter the block address to be deleted and click on *OK* to continue.
5. A *Function Code Editor* for the selected block appears. Click on *Delete* to delete the function code. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

Download Configuration File

This section describes how to download disk file configurations to a module. In operation, the CMU application writes an SLV

script and passes it to the SLV application which performs the actual download.

A function code configuration file (**CFG**) or a file system block (**NBS**) containing BASIC, C, or Batch program or data files can be downloaded. C, Batch and data file system blocks are done in a single download and cannot be done separately.

NOTE: The download operation checks the module status before downloading and issues warnings for the following conditions:

1. If the module is not in configure mode.
2. If the primary module of a primary/backup pair is selected.
3. If the module status indicates an inconsistency with the declared module hardware configuration.

To download a configuration:

1. Select the *Primary Hardware* or *Backup Hardware* item to be the target of the download.
2. Choose *Download Configuration File* from the *Primary Hardware* menu or *Backup Hardware* menu, or use the context menu. A *Download Configuration File* dialog box appears.
3. Select the type of files to download and select the files from the file list.
4. If appropriate, select *Initialize Module*.
5. Click on *OK* to continue. The CMU application writes an appropriate SLV script file into C:\EWS\CMU.SLV and the SLV application is launched to perform the download. The log file from the SLV application is in C:\EWS\CMU.LOG.
6. When done, use the inspect application to return the module to execute mode.

Save Module Configuration

This section describes how to save module-resident function code configurations and programs to a disk file. In operation, the CMU application writes an SLV script and passes it to the SLV application which performs the actual save.

This operation creates a configuration file with either a **CFG** or **NBS** extension, depending on the blocks saved. Function code blocks are saved in a **CFG** file. File system blocks, consisting of either BASIC, C, or Batch programs or data files are stored in an **NBS** file.

To save a module-resident configuration to a disk file.

1. Select the *Primary Hardware* or *Backup Hardware* item to be the source of the save.

2. Choose *Save Configuration File* from the *Primary Hardware* menu or *Backup Hardware* menu, or use the context menu. A *Save Module Configuration File to Disk* dialog box appears.
3. Select or enter a file name and select the type of files to save.
4. Click on *OK* to continue. The CMU application writes an appropriate SLV script file into C:\EWS\CMU.SLV and the SLV application is launched to perform the download. The log file from the SLV application is in C:\EWS\CMU.LOG.

CONFIGURATION OPERATIONS

These operations perform actions on a compiled configuration file (**CFG**). Some operations are performed with the project view active and others are performed with the configuration view active.

View Configuration

This function displays a view of the module configuration **CFG** file in a configuration view.

To view a **CFG** configuration, do one of the following:

- Point to a **CFG** configuration in the project tree and drag it to the configuration view area.
- Select a **CFG** configuration from the project tree and choose *View Configuration* from the *Configurations* menu or context menu.

If there is a current configuration view, it is overwritten by the new configuration. A maximum of two configuration views are available. To use a second configuration view, make the configuration view the active view and choose *Split* from the *Window* menu (**Alt W S**). For more information on using the views, refer to **WINDOW OPERATIONS**.

Close Configuration View

To close a configuration view, choose *Close Configuration View* from the *File* menu.

Filter/View

These options affect the display of the configuration in the configuration view. The configuration view must be the current view to perform these operations.

Viewing All Configuration Components

Filters can be applied to the configuration view. To remove the effect of these filters, choose *View All* from the *View* menu.

Viewing Unused and Configured Blocks	Any combination of unused blocks, configured blocks (defined function codes), or both can be displayed in the configuration view. To do so, choose the respective item from <i>View</i> menu.
Advanced Filtering	Properties affecting the configuration view appearance can be set. To do so, choose <i>Advanced Filter</i> from the <i>View</i> menu. The <i>Configuration View</i> properties page of the <i>Properties</i> dialog box appears. Refer to PROPERTIES for information. Alternatively, the <i>Configuration View</i> properties page of the <i>Properties</i> dialog box can be displayed by choosing <i>Properties</i> from the <i>File</i> menu, then clicking on the <i>Configuration View</i> tab.

Go To Block Number

This function locates a block number in the configuration view and scrolls the view to display the block. It only locates blocks that are displayed in the configuration view.

To locate a block:

1. If necessary, set the configuration view (filter) to display the type of block to be located (unused or configured).
2. With the configuration view the active view, choose *Go To Block Number* from the context menu or use the short cut key combination **Ctrl-G**.
3. Enter a block number in the dialog box and click on *OK*.

Editing and Listing Configurations

Function code operations can be performed on a **CFG** configuration file. All configuration operations are performed from the configuration view. The operations that can be performed depend on whether an unused or configured block is selected.

Changes should be made to a **CFG** configuration only when a quick modification is required to the control logic prior to loading the configurations into a module. It is recommended that the changes be made to the **CAD** sheets and re-compile them to maintain consistent documentation.

NOTE: After performing these operations, any changes must be saved to the file or to a new file to make them permanent. Refer to ***Saving a Configuration***.

- List** A function block can be listed from the configuration view. List allows function codes to be viewed. No changes can be made.

To list a function block:

1. If necessary, drag the *CFG Group* to be modified to a configuration view.

2. Select the configured block to be listed.
3. Choose *Configure Function Code*, then *List* from the *Configurations* menu or context menu.
4. A *Function Code Editor* for the selected block appears in which the specifications can be viewed.

Modify A function block can be modified from the configuration view. To modify a function block:

1. If necessary, display the *CFG Group* to be modified in a configuration view.
2. Select the configured block to be modified.
3. Choose *Configure Function Code*, then *Modify* from the *Configurations* menu or context menu.
4. A *Function Code Editor* for the selected block appears in which the modifications can be made. Refer to **FUNCTION CODE EDITOR** for information on using the function code edit window.
5. Click on *Apply* to apply the changes to the configuration view.

Add A function code can be added from the configuration view. To add a function code:

1. Display the *CFG Group* of interest in a configuration view.
2. Select an unused function block. If necessary, use the *View* menu or *Filter* from the context menu to turn on display of unused function blocks.
3. Choose *Configure Function Code*, then *Add* from the *Configurations* menu or context menu. An *Add Function Code* dialog box appears.
4. Enter the requested information. Refer to Table 7-5 for a description of the fields.

Table 7-5. Add Function Code

Field	Description
Block address	Address at which the first function code is to be added.
Count	Number of function codes to add.
Offset	Block address offset at which each subsequent function code is added.
Function code	Function code type to add.

5. Click on *OK* to continue. A *Function Code Editor* for the selected block appears in which the specifications can be edited. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

6. Click on *Add* in the function code editor to add the function code to the indicated block address make the change appear in the configuration view.

If more than one function code is being added, the function code editor shows the next function code. Continue until all function codes are added.

If an address conflict is encountered, the original *Add Function Code* dialog box is displayed with the remaining add count and allows a new address to be entered.

Saving a Configuration

After making changes to a configuration, the changes can be saved to the original file or to a new file.

To save changes to the original file, choose *Save Configuration View* from the *File* menu.

To save changes to a new file:

1. Choose *Save Configuration View As* from the *File* menu. A save file dialog box appears.
2. Enter the new file name and click on *OK* to continue.

Edit File Header (CFG)

This function display the information in a configuration file header. The **CFG** file header contains job and module information, loop and address information, and can show module utilization information. The job and module information can be edited.

To display the file header:

1. If necessary, display the *CFG Group* of interest in a configuration view.
2. With the configuration view active, choose *Edit Header (CFG)* from the *Configuration* menu or context menu. The *Edit CFG Header* dialog box appears.
3. Do any of the following:
 - Review the loop and address information.

- Review or edit the file header. Refer to Table 7-6 for field information. To fill in module information from the **MHD** file, click on *Update from MHD*.
- Review the utilization information. To acquire current values, click on *Update*.

Table 7-6. CFG File Header Information

Field	Description
Customer name	Twenty-character field for the customer name. An entry in this field is optional.
Plant site	Ten-character field for the plant site. An entry in this field is optional.
Contracting firm	Thirty-character field for the contracting firm. An entry in this field is optional.
Job number	Ten-character field for the project number. This field requires an entry. It is recommended to use the project directory name.
Configuration description	User provided description of the configuration purpose. This will appear after the module address in the project tree.
Module type	Type of module.
Firmware revision	Firmware revision of the module.
Module ID	Two-character alphanumeric module identifier. This identifier is used by the cross-referencer and must be unique for each module in the project.
Drawing title	Drawing title.
Drawing number	Seven-character field for a drawing number.

CAD SHEET OPERATIONS

These operations perform actions on CAD sheet configuration files. Some operations are performed with the project view active and others are performed with the configuration view active.

View Configuration

This function displays a view of the **CAD** sheet configuration for a module in a configuration view.

To view a **CAD** sheet configuration,

1. Select a *CAD Group* from the project tree.
2. Do one of the following:
 - Point to the selected *CAD Group* and drag it to the configuration view area.

- Choose *View Configuration* from the *CAD Sheet* menu or use the context menu.

If there is a current configuration view, it is overwritten by the new configuration. A maximum of two configuration views are available. To use a second configuration view, make the configuration view the active view and choose *Split* from the *Window* menu (**[Alt][W][S]**). For more information on using the views, refer to **WINDOW OPERATIONS**.

Close Configuration View

To close a configuration view, choose *Close Configuration View* from the *File* menu.

NOTES:

1. The latest revision of all **CAD** sheets in the module directory are read.
2. The *CAD Group* configuration view shows function codes with unassigned block addresses first using an internally assigned negative block number.
3. If duplicate block numbers are encountered (in different **CAD** sheet files), the duplicates are displayed but are highlighted in red.

Filter/View

These options affect the display of the configuration in the configuration view. The configuration view must be the current view to perform these operations.

Viewing All Configuration Components

Filters can be applied to the configuration view. To remove the affect of these filters, choose *View All* from the *View* menu.

Viewing Unused and Configured Blocks

Any combination of unused blocks, configured blocks (defined function codes), or both can be displayed in the configuration view. To do so, choose the respective item from *View* menu.

Advanced Filtering

Properties affecting the configuration view appearance can be set. To do so, choose *Advanced Filter* from the *View* menu. The *Configuration View* properties page of the *Properties* dialog box appears. Refer to **PROPERTIES** for information. Alternatively, the *Configuration View* properties page of the *Properties* dialog box can be displayed by choosing *Properties* from the *File* menu, then clicking on the *Configuration View* tab.

CAD Group configuration views also have an option to be sorted by sheet number.

Go To Block Number

This function locates a block number in the configuration view and scrolls the view to display the block. It only locates blocks that are displayed in the configuration view.

To locate a block:

1. If necessary, set the configuration view (filter) to display the type of block to be located (unused or configured).
2. With the configuration view the active view, choose *Go To Block Number* from the context menu or use the short cut key combination **Ctrl-G**.
3. Enter a block number in the dialog box and click on *OK*.

List Configuration

A function code can be listed from the configuration view. List allows function codes to be viewed but no changes can be made. To list a function code:

1. Display the *CAD Group* to be listed to a configuration view.
2. Select the block to be listed.
3. Choose *Configure Function Code*, then *List* from the *CAD Sheets* menu or context menu.
4. A *Function Code Editor* for the selected block appears in which the specifications can be viewed. Refer to **FUNCTION CODE EDITOR** for information on using the function code edit window.

FILE SYSTEM OPERATIONS

These operations are not available at this time.

MONITOR/TREND OPERATIONS

These operations launch the trend application in monitor mode or to edit the group definition. If a group is currently open in the trend application, it is closed and the selected group is opened.

Monitor a Group

To monitor a group:

1. Select the *Monitor/Trend Group* of interest from the project tree.
2. Choose *Monitor Group* from the *Monitor/Trend* menu or from the context menu.

The monitor, trend, and tune application is launched with scan on and begins monitoring and trending the group.

Edit a Group Definition

To edit a group definition:

1. Select the *Monitor/Trend Group* of interest from the project tree.
2. Choose *Edit Group Definition* from the *Monitor/Trend* menu or from the context menu.

The monitor, trend, and tune application is launched with scan off, allowing the group to be modified.

VERIFY OPERATIONS

Verify operation can be performed to verify the following:

- **CAD** sheets using the **CFG** file as a reference.
- **CAD** sheets using the module-resident configuration as a reference.
- **CFG** file using the module-resident configuration as a reference.
- **CFG** file using another **CFG** file as a reference.

The verification process checks for specification differences, block differences, function code inconsistencies, and configuration file problems.

Specifications

Specification differences are defined as any integer or boolean specifications not equal in value, any real specifications that do not match the module's rounded value, or any string specifications that do not match.

Blocks

Block differences are instances such as the same block number from each source being assigned a different function code.

Function Codes

Function code inconsistencies are situations where the data files used by the work station to define function codes differ in definition to the module.

Inputs

Any function code input with a different block address where the source is a **CFG** file (**CAD** sheet inputs cannot be verified against a reference).

This function also checks all configuration files for valid data. An update line appears giving the current block under verification and the percentage complete. Since the percentage complete is based on the contents of the configuration file, differences between the sources will affect this number. Do not be alarmed if the verification ends prior to recognition of 100 percent complete. This happens if the block totals between the module and **CFG** file are not identical.

The process completes with a count of block totals for both the module and the configuration file. This information is also included in the verification file.

In operation, the CMU application writes a script and passes it to the SLV utility which performs the actual verification. The results are stored in C:\EWS\CMU.SLV and C:\EWS\CMU.LOG contains the SLV log information. The file is located in the module directory and is overwritten each time verification is run on the module **CAD** sheets or **CFG** file. It can be viewed or printed at any time using the log file viewer.

To verify **CAD** sheets or a **CFG** file:

1. Select the *CAD Group* or *CFG Group* to be verified from the project tree.
2. Choose *Verify Against Reference* from the *CAD Sheets* or *Configuration* menu or use the context menu.

The verify *Source/Reference Selection* dialog box appears. The source defaults to either *CAD Files* or *CFG Files* depending if a *CAD Group* or *CFG Group* was selected in the project tree.

3. If the source is *CAD Files*, an *Update Specs* option appears. To update specifications within the **CAD** sheets from either the module-resident configuration or the **CFG** file, check this option. If in a matching function block there is a discrepancy in a specification, it is copied from the module-resident configuration or the **CFG** file to the appropriate **CAD** sheet file. This does not include block addresses on input specifications because they cannot be verified.
4. Select a reference for the verification process: *Module Hardware* or *CFG File*, then
 - If *Module Hardware* is selected, either the *Backup* or *Primary* module must also be selected.
 - If *CFG File* is selected, a file list appears and the reference **CFG** file must be selected.
5. Click on *OK* to continue. The CMU application writes a script based on the selected option and passes it to the SLV utility which does the actual verification.

REPORT OPERATIONS

Four types of reports can be generated by the CMU application:

Configuration Utilization	Calculates and lists module configuration utilization.
Trace	Lists spare and configured blocks.

Specification List Lists each function code, its specifications and specification details (including whether it is tunable, its type, value, translation, and description).

Specification Dump Reports all configured blocks and their associated specification values in tabular format.

Reports can be performed for any of the configuration views (*CAD Group*, *CFG Group*, or either hardware group) based on the configuration view properties (filter settings) and report properties. The reports are written to disk files and can be viewed using the log file viewer. Reports are viewed automatically when generated if the *Automatically view the report* property is set. Refer to **PROPERTIES** for information on setting report properties.

Calculating Configuration Utilization

There are limits placed on configurations with respect to the amount of memory, number of blocks, and number of certain types of function codes utilized. During the course of configuring a module, the user may wish to know the current state of the utilizations so as to not exceed the limits. The utilizations totaled for the configuration view are:

RAM Number of bytes required to run the configuration.

EEPROM or NVM Number of bytes of non-volatile memory required.

Blocks Number of bytes required to accommodate the configured blocks of the drawing.

CPU Approximate cycle time of the configuration. (If Basic or C programs are part of the configuration this time may be significantly affected.)

Special Function Codes The number of times a function code is used that is limited in occurrences for a single configuration.

To calculate configuration utilization:

1. Display the *CAD Group*, *CFG Group*, or hardware group for the module of interest in a configuration view.
2. With the configuration view active, choose *View Report*, then *Calc Utilization* from the *Module* menu or use the context menu.

The results are written to the log file **CALCUTLZ.LOG** in the module directory and can be viewed or printed at any time

using the log file viewer. The log file contains the following information:

- Module and revision targeted for the configuration.
- Path of the configuration.
- Calculated utilization totals for each sheet.
- Total utilizations for the entire configuration.
- Percentage of RAM and EEROM/NVM memory used.

Trace

The trace report lists all spare, fixed, and configured blocks for the configuration view. This report is a useful tool for locating available (spare) blocks when additional function code logic needs to be added.

To generate a trace report:

1. Display the *CFG Group* or *CAD Group* for the module of interest in a configuration view.
2. With the configuration view active, choose *View Report*, then *Trace* from the *Module* menu or use the context menu.

The report is written to a text file named **xxxxx.BLK** where xxxxx is the first five characters of the module configuration drawing. It can be viewed or printed at any time using the log file viewer.

Specification Listing

This report lists each specification and whether it is tunable, its type, value, translation, and description.

1. Display the *CFG Group* or *CAD Group* for the module of interest in a configuration view.
2. With the configuration view active, choose *View Report*, then *Spec Listing* from the *Module* menu or use the context menu.

The results are written to the log file and can be viewed or printed at any time using the log file viewer.

Specification Dump

The specification dump reports all configured blocks and their associated specification values for the configuration view into a tabular text file. This report is intended to be imported into a

spread sheet or database program. The report can be stored in one of two formats as set by the report properties: tab separated or fixed space.

To generate a specification dump:

1. Display the *CFG Group* for the module of interest in a configuration view.
2. With the configuration view active, choose *View Report*, then *Spec Dump* from the *Module* menu or use the context menu.

The results are written to files as described in the following paragraphs:

Tab Separated

This is the preferred format as it does not truncate data and is more versatile. It produces a flexible tab separated text file that can be imported into most popular spreadsheet, database, and other programs that can read text files. Each line in the report contains the complete specification data for one function code separated by tabs. This report is named **xxxxxx.TXT**, where *xxxxxx* refers to the first five characters of the module configuration drawing.

Fixed Space

This format is provided for backward compatibility. This format is also called system data format (SDF) generated by the SCAD products. To use the data stored in this file, it can be read into a space separated text file. Any application that supports this file format can be used to access the SDF report. This report is named **xxxxxx.SDF**, where *xxxxxx* refers to the first five characters of the module configuration drawing.

The structure of the SDF format file is as follows:

- Each line of the **SDF** file is 1228 characters long and contains the complete specification data for one function code; refer to Table 7-7.

Table 7-7. SDF File Structure

Character Positions	Function
1 - 4	Start block.
5 - 8	End block (if more than one block used).
9 - 11	Function code number.
10 - 28	S1 value
29 - 47	S2 value
· ·	
· ·	
1210 - 1228	S64 value

- All numeric specification values are stored as real numbers in a 19.3 format. Each value occupies 19 character positions, with the last four positions being a decimal point and three decimal places.
- Strings are truncated at 16 characters.

FUNCTION CODE EDITOR

The function code editor facilitates viewing, configuring, and editing function blocks. It appears whenever a function code is to be edited within the module tools and is used by the following utilities: graphical module configuration, configuration maintenance utility, and trend.

The function code editor incorporates several features that make configuring function blocks easier. User-friendly features of the function code editor include specification filtering, application tabs, and specification translation.

Specification Filtering

Specification filtering refers to the ability to filter and view the specifications by type (all, inputs, internal, tunable, on-line configurable, and spare).

Application Tabs

This feature filters function code specifications by suitability to function. For example, a function code 216 can be used for TC, RTD, high level, and low level applications. When this function code is configured, a tab appears for each of these applications. By clicking on the appropriate tab, only those specifications needing configuration for that application are displayed.

Specification Translation

Specification translation refers to the ability to configure complex specifications by picking options from radio buttons and pick lists. The configuration is automatically translated to the numeric value and entered as the specification value.

Function Code Edit Window

The function code editor is used to view or modify function code specifications. An example function code editor window is shown in Figure 7-1.

Title Bar

The title bar shows the application window or view it is assigned to and the loop, PCU, and module address being configured.

Function Code Definition

The function code definition bar shows the block address, function code number, and function code description being configured.

Tab Bar

The tab bar is used by function codes with advanced specification filtering requirements. By clicking a particular tab, only the specifications needed to configure the function code for

that usage is displayed. All function codes have a standard and filter tab. The standard tab shows all configurable specifications. The filter tab shows a filtered subset of the specifications based on the application's filter or view options. Additional tabs may be present to provide a logical subset of specifications to ease the configuration process.

Controls The controls are described in the following subsection.

Specification Display Frame This area can show all the specifications (as filtered by the tab bar). The view splitter adjusts the amount window space used by the specification display frame and the specification details frame. Scroll bars can be used to view parts of the specification display frame not currently visible. Table 7-8 describes the columns used in this area.

Table 7-8. Function Code Specifications

Item	Description
Spec	Specification number as listed in the Function Code Application Manual.
Tune	Indicates whether the specification is tunable or not.
Type	Data type of the specification: Integer, Real, Character.
Value	Specification value.
Translation	Meaning of the specification value for binary or decimal coded specifications. An out of range message appears for integer and real values if they are out of range. If the translation needs more room, it is extended into the description column and the description appears in the following line.
Description	Description of the specification.

Specification Details Frame This area is used to configure the function code specifications. If the specification requires an integer, real, or string value, it can be entered here. If the specification requires a binary or decimal coded value, pick lists appear from which, for example, input devices can be selected. For an FC 216, input types of thermocouple, RTD, high level, low level, and user defined appear. After selecting an input type, a subtype can be selected. These selections are automatically translated into the decimal code and entered as the function code specification. The behavior and appearance of specification details depend on the specific function code being configured.

Controls

The presence of the controls depends on the application and the function being performed. The function code editor controls operate as described in the following paragraphs.

<i>APPLY</i>	Applies the current specification values to the configuration. If on-line or tuning, or performing hardware configuration, the values are applied to the module configuration residing in the module. Otherwise, these values are saved with the configuration on the next save.
<i>ADD</i>	Appears in place of <i>APPLY</i> when adding a function code to a block.
<i>DELETE</i>	Appears in place of <i>APPLY</i> when deleting a function code from a block.
<i>RESET</i>	Resets all specifications to their values prior to the current editing session or since they have been applied.
<i>CLOSE</i>	Closes the function code edit window.
<i>DEFAULT</i>	Each specification has a default value. This button sets the current specification to its default value.
<i>DEFAULT ALL</i>	Sets all specifications to their default values.
<i>PREV FC</i>	Displays the function code at the next block in the configuration.
<i>NEXT FC</i>	Displays the function code at the previous block in the configuration.

Viewing/Filtering Options

These options affect the display of the specifications in the function code editor when the filter tab is selected. The function code edit window must be the active window to set its filter options. Filter options appear on the applications *View* menu.

Overview	Displays all usable specifications defined for the function code.
Inputs	Displays input specifications only.
Internal	Displays internal specifications only.
Tunable	Displays tunable specifications only.
On-Line Configurable	Displays on-line configurable specifications only.
Spares	Display spare and reserved specifications for any of the other filter options.

Editing Specifications

Specifications can be edited by typing a value in the *Value* field of the specification display frame or in the specification details frame. Changes do not occur until **Enter** is pressed. If a range value is entered that exceeds the specification's defined minimum or maximum as displayed in the status field, an out of range message appears in the translation field. When editing specifications, a context menu can be used for editing operations (undo, cut, copy, paste, and delete).

If the specification uses a binary or decimal coded value, a pick list appears for the specification in the specification details frame. The value can be entered by selecting from a pick list rather than entering the binary or decimal code. If a value is entered that cannot be represented by the pick list or available buttons, an undefined message appears.

If a value is entered that exceeds the storage capability of the data type, an *Illegal* message is displayed and the cursor remains on that specification field until it is changed to an allowable value.

After making editing changes, use *Apply* to apply the changes to the module configuration if on-line or make the changes available for saving to disk.

UTILITIES

Com Port Configuration	This item configures WinTools communications. Refer to CONFIGURING WINTOOLS COMMUNICATIONS in Section 11 for information.
View Log	This item launches the log file viewer. Refer to VIEWING LOG FILES in Section 11 for information.
Inspect/Mode Change	This item launches the inspect application. Refer to Section 8 for information.

SECTION 8 - INSPECT

INTRODUCTION

This section describes how to use the inspect application to perform the following configuration functions:

- Inspect module status and operation within a PCU node.
- View status and problem reports.
- Change module modes.
- Sequence qualified hardware through the on-line configuration process.

Typically, the inspect application is used to inspect the contents of a PCU node before a download or configuration operation to insure module readiness, to put modules into execute mode after configuration, and to facilitate on-line configuration.

The inspect application scans module addresses within a PCU node for status and other information. It can inspect any module or group of modules within a PCU node it has access to over the INFI-NET network. Real time status information is displayed in list format for each inspected module. Status details and problem reports can also be displayed upon user request.

ENHANCED STATUS REPORTING MODULES

The following modules have enhanced status reporting capabilities that are available when connected to the INFI-NET network:

- IMMFP01/02/03/03B controllers with revision F.2 or later firmware.
- BRC Harmony Bridge Controller.

The enhanced status reporting capability facilitates on-line configuration in particular. Difference in these and other modules and Plant Loop operation are noted where needed.

STARTING THE APPLICATION

The inspect application can be started from the application launcher, the CMU application, or directly from Windows.

To start the inspect application, do one of the following:

CMU	Choose <i>Inspect/Mode Change</i> from the <i>CMU PCU</i> or <i>Module</i> menu.
Application Launcher	Click on its icon in the application launcher or choose it from the <i>Applications</i> menu.
Windows 3.1 or Windows NT 3.51	Double click on its icon in the <i>Bailey WinTools</i> group.
Windows 95 or Windows NT 4.0	Choose <i>Inspect</i> from the <i>Bailey WinTools</i> menu.

EXITING THE APPLICATION

Exiting the inspect application closes the application window. To exit the inspect application, choose *Exit* from the *File* menu (**Alt F X**).

APPLICATION WINDOW

This section describes the application window, the menus, how to access on-line help, and view options.

Parts of the Application Window

The inspect application runs under the Windows operating environment. The parts of the inspect application window are shown in Figure 8-1 and described in the following paragraphs.

Title Bar	The title bar is a standard Windows title bar showing the application title.
Menu Bar	The menu bar contains the inspect application menus. Each inspect command is listed as an item on the appropriate menu. Menu selections are made by clicking on the menu and selecting an item, using the menu keys Alt menu key item key , or using a shortcut key combination if one is available.
Tool Bar	The tool bar shows selected menu commands in graphic form. The tool bar is an alternative method of choosing commands. The tool bar appears only when enabled using <i>Tool Bar</i> from the <i>View</i> menu (Alt V T).
Application Work Space	The application work space contains one inspect window. The inspect window displays the inspect information and status/report information for the selected loop and PCU address.

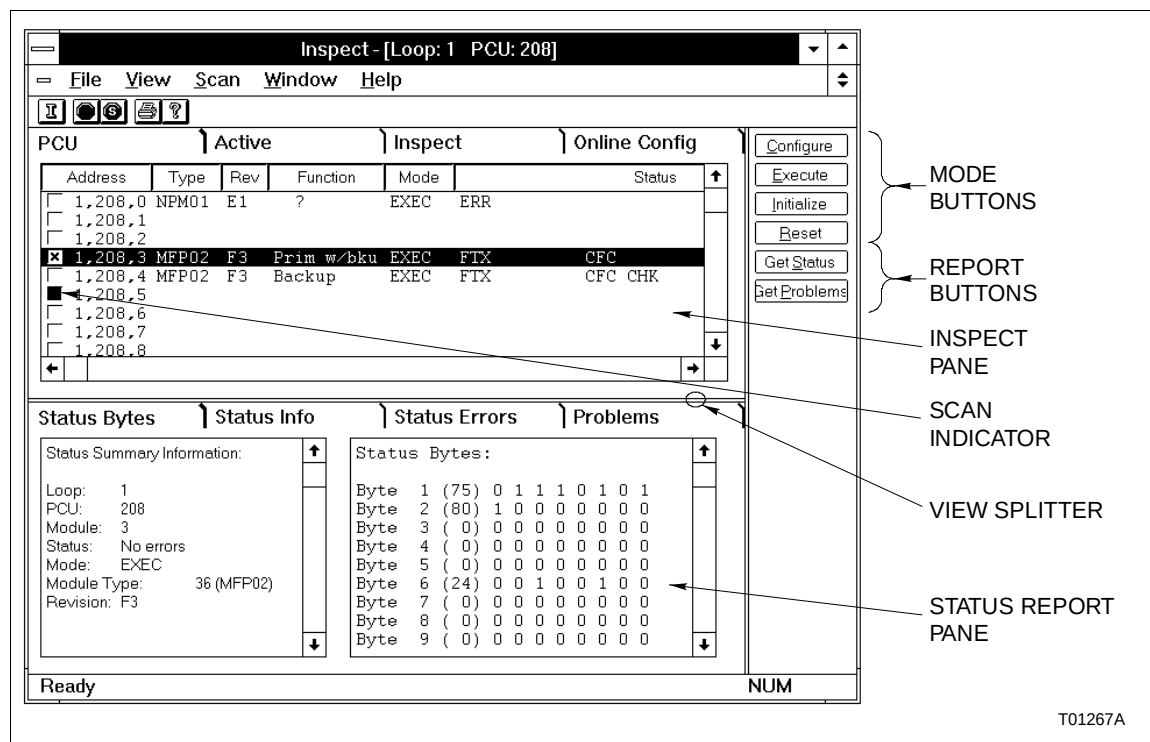


Figure 8-1. Inspect Application Window

Inspect Pane Displays PCU addresses and associated module and status information. Tabs facilitate display of all PCU address, active modules, user selected modules, and entering on-line configuration mode. Modules are selected within the inspect pane as the target for mode changes, status and problem reports, and on-line configuration. Multi-select methods are supported when selecting modules.

Scan Indicator This is the highlighted check box on the left side of the inspect pane. It indicates the current PCU addresses being examined. Its movement through the list of addresses indicates that scanning is being performed.

Status/Report Pane Displays module status and problem report information in selected formats. Tabs are used to select the format of interest.

Mode Buttons Mode buttons are used to change the operating mode of selected modules.

Report Buttons Report buttons are used to gather and update status information and problem reports for the selected module.

View Splitter The percentage of space given to the inspect pane and to the status/report pane within the window can be adjusted using the view splitter.

Status Bar The status bar appears at the bottom of the application window. It displays messages on the current status of the inspect application and brief help instructions when working in the application. The status bar appears only when enabled using *Status Bar* from the *View* menu (**Alt V S**).

Menu Items

The following paragraphs briefly describe each inspect menu. For a complete description of each menu item, refer to the subsections that follow.

- File** Contains items to save, view, and print status and report information, refer to **REPORT OPERATIONS**. Contains an item to set preferences, refer to **INSPECT OPERATIONS**. Contains an item to exit the application, refer to **EXITING THE APPLICATION**.
- View** Contains items to control the display of the tool bar and status bar, refer to **View Options**. Contains an item to select a loop and PCU to inspect, refer to **INSPECT OPERATIONS**.
- Scan** Contains items to start and stop scanning PCU addresses, refer to **INSPECT OPERATIONS**.
- Window** Contains items to manage windows. However, the inspect application supports a single inspect window within the inspect application window. The inspect window should be left maximized within its application window.
- Help** Contains items to access help, refer to **On-Line Help**.

On-Line Help

On-line help is available for the inspect application. It provides command descriptions, procedures and reference material similar to that found in the instruction. The on-line help can be used for convenient and quick access to the inspect application operating procedures. The help system conforms to the standard Windows conventions for on-line help. Refer to **ON-LINE HELP** in Section 3 or the Windows documentation for more information.

Tool Bar

The tool bar provides an alternate and more convenient method of choosing commands. File, scan, and help commands are represented by an icon on the tool bar. To choose a command from the tool bar, point and click on the icon.

View Options

The display of certain features in the application can be controlled through items on the *View* menu. These items operate as toggles. If a toggle is on, a check mark appears beside the item on the *View* menu. If a toggle is off, the check mark is absent. To change the state of the toggle, choose the item from the *View* menu. The tool bar and status bar always default to their view on state whenever the application is launched.

- Status Bar** Toggles the status bar display on and off. Refer to **Parts of the Application Window** for information on the status bar.
- Tool Bar** Toggles the tool bar on and off. Refer to **Tool Bar** for information.

WINDOW OPERATIONS

The inspect application supports a single inspect window within the inspect application window. The inspect window should be left maximized within its application window.

INSPECT OPERATIONS

The inspect application can be used to inspect and view all addresses in a PCU node, the active addresses in PCU node, or only selected addresses in a PCU node. Active addresses are those returning a response; selected addresses are those selected by the user by clicking on the scan indicator check box and are indicated by the appearance of an X in this field.

In operation, the inspect application polls the list of designated addresses and displays the information described in Table 8-1. As each address is polled, the scanning indicator moves through the scan indicator check box and appears with a high-lighted background within the field.

Table 8-1. Inspect Fields

Field	Description
Address	Loop, PCU node, and module address.
Type ¹	The module type at this address, e.g, <i>NPM01</i> , <i>MFP02</i> , etc. A blank field indicates no module present or detected.
Revision	Firmware revision of the module.
Function ¹	The following module function information can be reported by control modules with enhanced status reporting capabilities operating on an INFI-NET network; otherwise, a question mark (?) is displayed.

Table 8-1. Inspect Fields (continued)

Field	Description
Function (continued)	
Primary SA	Module is operating as a primary, stand-alone controller.
Prim w/red	Module is operating as a primary, redundant controller. The redundant backup module cannot be observed or modified. Both modules respond to the same module address.
Prim w/bkup	Module is operating as a primary, redundant controller that is configurable using on-line configuration procedures. The primary module has an address n.
Backup	The module is operating as a backup module for a primary controller. The backup module is configurable using the on-line configuration procedures. The backup module has the address of n+1.
?	Unable to determine the module's function.
Mode	
EXEC	Module is in execute mode.
CONFIG	Module is in configure mode.
ERROR	Module is in error mode.
Status	The status field has seven subfields. ERR and BSY overwrite each other and appear in the same location.
ERR	Module is in an error state.
BSY	Module is updating its configuration or performing low level maintenance functions and may not perform the desired operation at this time. This status goes away as the module finishes its tasks.
FTX	First time execute. The module has just changed into execute mode (MFC, MFP, and BRC modules only).
BAC	Bad backup configuration (MFC, MFP, and BRC modules only). This status is set by a primary modules. It means the backup is not ready to take over should the primary fail. It can also exist in a backup module if on-line configuration swapped the module roles for testing.
CFG	On-line configuration change has been performed (MFC, MFP, and BRC modules only). This status indicates that the backup has been put into configure mode and that the configuration was changed.
CFC	Configuration is completed. (Configuration transferred from primary to backup.) Reported for modules with enhanced status reporting capability only.
CHK	Checkpointing is completed. (Process data transferred from primary to backup.) Reported for modules with enhanced status reporting capability only.

NOTE: 1. Partial information followed by a question mark (?) may be present for modules that lack enhanced reporting capability or if a module is in configure mode and the remainder of this data is available only when the module is in execute mode.

PCU Address

The inspect application is limited to displaying information about modules within a single PCU node at any given time. To inspect a PCU node:

1. Choose *Specify Loop, PCU* from the *View* menu (**Alt V S**) or use the inspect icon in the tool bar.
2. A dialog box appears in which the loop and PCU address can be entered. A list box also maintains a list of the previous 10 addresses that have been inspected and modules selected for the *Inspect* view (using the scan indicator check box). Use the list box to select an address or enter an address into the *Loop* and *PCU* fields.
3. Click on *OK* to continue and automatically start scanning. The inspect pane will show the module list at the PCU address and associated information.

View Tabs

Inspect views are displayed by clicking on the appropriate view tab: *PCU*, *Active*, or *Inspect*.

PCU View The PCU view shows all addresses within the PCU node.

To inspect all addresses with the PCU node:

1. Click on the *PCU* tab in the inspect pane.
2. If necessary, use the scroll bars or resize the pane using the window splitter to view addresses of interest.

The scanning indicator moves through the entire list of addresses. It may momentarily disappear from view depending on the size of the inspect pane.

Active View The active view shows only those addresses returning a response to the scanning request. As modules change mode, they may disappear and reappear from the list.

To view all active addresses:

1. Click on the *Active* tab in the inspect pane.
2. If necessary, use the scroll bars or resize the pane using the window splitter to view addresses of interest.

Scanning is performed on all addresses in this view. The scanning indicator may momentarily disappear from view as it is scanning addresses that are not displayed.

Inspect View The inspect view shows only those addresses requested by the user.

To view user requested addresses:

1. Click on the scan indicator check box next to each address to be viewed to place a checkmark (X) in the field. This can be done in any of the three views.

NOTE: The selected addresses are stored in the PCU node selection list. Module addresses are restored according to this list if the list is used to select a PCU node.

2. Click on the *Inspect* tab in the inspect pane.
3. If necessary, use the scroll bars or resize the pane using the window splitter to view addresses of interest.

Only the addresses requested are polled and scanning should be faster than in the other view modes.

Preferences

Preferences allow the user to set the communications time out period and report option.

Communications Time-Out

The communications time-out preference can be adjusted as necessary depending on the size and complexity of the module configurations. A large and complex module configuration requires more time for completing checkpointing, for example. However, setting the communications time-out to a high value will take longer to report failed communications in other circumstances.

Automatically View Report

If this option is set, then after a report file is generated it is immediately displayed in (and can be printed from) the default log file viewer. Refer to **PROPERTIES** in Section 7 for setting the default log file viewer.

Set Preferences

To set the preferences:

1. Choose *Preferences* from the file menu (**Alt F F**). The *Preferences* dialog box appears.

2. Do either of the following:
 - Enter a communications time-out number in seconds. The default is 30 seconds.
 - Click on the *Automatically view report* check box to toggle this option on or off.
3. Click on *OK* to accept the preference settings.

ON-LINE CONFIGURATION

On-line configuration allows configuration changes to be made within the following context: 1) the process remains under control of the process control system and 2) the original configuration is preserved until the new configuration is tested and accepted. It requires the control system to have a primary/backup module pair and for the modules to be configured appropriately for on-line configuration operation. During on-line configuration, the backup module is taken off-line and configuration changes made. The primary module remains in control of the process. After configuration changes have been made, the functional role of the physical modules can be switched and the changed configuration put in control of the process. Options allow for testing of the changed configuration and to revert to the original configuration.

The on-line configuration view is used to guide the user through the on-line configuration processes by validating the primary/backup module selection and by prompting the user through the correct module mode changes. Utilizing control modules with enhanced status reporting capability improves guidance of module selection and sequencing by providing additional status information to the inspect application. This information was previously available only via the control module LED indicators.

The on-line configuration view can be exited and re-entered at different points throughout this process, typically while configuration changes are being made with other tools. The on-line configuration view utilizes module status to select a point at which to re-enter the on-line configuration process. Manual control of mode changes are available from the other three inspect views utilizing the mode change buttons.

NOTES:

1. The term ***primary/backup module pair*** specifically refers to controller modules having addresses *n* and *n+1* and having switch settings enabled for on-line configuration. Refer to the specific module instruction for more information.

2. During on-line configuration, the primary/backup module pair may contain two configurations: the **original configuration** refers to the configuration residing in the primary/backup module pair before entering on-line configuration; the **new or test configuration** refers to any modifications to the original or any other configuration downloaded to the primary/backup module pair.
3. During on-line configuration, the functional role of the physical modules comprising the primary/backup pair may switch. The module at address n is always the module controlling the process. Upon exiting from on-line configuration, the roles and addresses of the physical modules may be reversed from that when on-line configuration was entered. This depends on the actual steps performed during on-line configuration.
4. Throughout the on-line configuration process it is the users responsibility to correctly make the on-line configuration changes and wait for the proper module conditions to be established before each mode change. Incorrect use can still result in an error in the primary module that is in control of the process. Refer to the module instruction for detailed sequencing and status information.
5. On-line configuration restrictions do not allow the following:
 - a. Deleting existing function codes.
 - b. Modifying specifications that would increase a function code block's memory utilization.
 - c. Adding function code blocks within already used block space within the module's control segments. Blocks can only be added following the last block used within a segment.

Modules with Enhanced Status Reporting

Control modules with enhanced status reporting capability can report their configuration complete (CFC) and checkpointing done (CHK) status when operating on an INFI-NET network. These are related to the backup module's LED 7 and LED 8 states. These control modules can also report their function role: Primary w/bkup or Backup. The inspect application can display these in the status and mode fields and can be used to verify status during on-line configuration.

MFC modules, pre-F.2 MFP modules, and any control module operating on Plant Loop must be physically checked to verify these states. There are two procedures given for on-line configuration:

Procedure A is used for modules having enhanced status reporting capability operating on the INFI-NET network.

Procedure B is used for MFC modules and pre-F.2 MFP modules or any modules operating on Plant Loop.

On-Line Configuration Procedure A

To begin the on-line configuration process of a primary/backup control module pair having enhanced status reporting capability:

1. Display the primary/backup module pair in the inspect pane. A properly executing primary/backup module pair should show the following conditions in the inspect pane. Insure these conditions appear before continuing:

Address	Function	Mode	Status
N	Primary w/bkup	EXEC	CFC CHK
N+1	Backup	EXEC	FTX CHK

2. Select either the primary or backup module or both.
3. Click on the *Online Config* tab.
4. A message box should appear indicating that a primary/backup module pair has been successfully detected and requesting a confirmation. The modules must have enhanced status reporting capability for this to be detected properly. Click on *OK* to continue. The message box also indicates what will occur if on-line configuration is continued (described in Step 5).

NOTES:

If incorrect or invalid module selections are made, one of the following may occur:

1. If a message box appears indicating that the role of the module could not be determined, then the modules lack enhanced status reporting capability or are not on an INFI-NET network. Refer to ***On-Line Configuration Procedure B*** or insure that the correct modules of interest are selected.
2. If no modules are selected, the system signals with a tone.
3. If more than two modules are selected, a message appears indicating that at least one and no more than two modules must be selected.
4. If a message box appears indicating that a valid primary/backup module pair was not selected, a valid primary/backup module pair must be selected to perform on-line configuration. Also check the redundancy and on-line configuration switch settings for these modules and that the redundant cable link is properly installed.

5. There is a short wait for the backup module to go into configure mode. After successfully entering on-line configuration mode, the following occurs:

- The primary/backup module pair are the only modules visible in the inspect pane. The backup module mode should show *CONFIG* (after a short delay required for switching modes).
- The mode change buttons are disabled. This is to prevent the user from making mode changes out of sequence and, possibly, inadvertently removing control from the process.
- A new set of buttons appears in the inspect pane. These are described in the following paragraphs. The buttons are enabled and disabled depending on valid operations at each step. Initially, only the *Go Back to Original Configuration* button is enabled.
- A new button appears beneath the module mode buttons. The *Stop* button is used to exit on-line configuration. It is enabled only at appropriate times. It interrupts the communications time out period. It can be used, for example, if a module's checkpointing does not appear to be completing.

NOTE: Periodically, when waiting for a mode change, a prompt may appear requesting to continue or exit. The time period is determined by the communication time out preference setting. The *Stop* button performs the same function, if for example, it is needed to exit before the time out period expires.

6. Observe the inspect pane and wait for the following conditions before continuing:

Address	Function	Mode	Status		
N	Primary w/bkup	EXEC	ERR	BAC	CFC
N+1	Backup	CONFIG	CFC		

7. Any available tool can be used to change the backup module configuration (e.g., the CMU or SLV application). After configuration changes have been made, the inspect pane shows the following conditions:

Address	Function	Mode	Status		
N	Primary w/bkup	EXEC	ERR	BAC	CFC
N+1	Backup	CONFIG	CFG CFC		

8. The following can be performed:

Go back to original configuration
Test new configuration

If no configuration changes are made, the only valid operation is *Go Back to Original Configuration*.

The following paragraphs describe the operations that can be performed during on-line configuration. The buttons are enabled and disabled depending on valid operations at each step. Informational messages appear beneath the on-line configuration buttons indicating the state of the requested operation.

Test New Configuration

Clicking on *Test new configuration* puts the new configuration in control of the process; the original configuration is preserved within the primary/backup module pair.

The new configuration is controlling the process when the inspect pane shows the following conditions. After clicking on *Test new configuration*, observe the inspect pane and wait for the following conditions before continuing:

Address	Function	Mode	Status
N	Primary w/bkup	EXEC	ERR BAC CFG CFC CHK
N+1	Backup	EXEC	ERR BAC CFC CHK

After testing the configuration, the following can be performed:

Accept new configuration
Go back to original configuration
Make additional configuration changes

NOTE: If there is an illegal new configuration, the original configuration will remain in control.

Accept New Configuration

Clicking on *Accept new configuration* exits on-line configuration mode with the new configuration in control of the process. The original configuration is removed from the primary/backup module pair.

The new configuration is accepted and controlling the process when the inspect pane shows the following conditions. After clicking on *Accept new configuration*, observe the inspect pane and wait for the following conditions before continuing:

Address	Function	Mode	Status
N	Primary w/bkup	EXEC	CFC CHK
N+1	Backup	EXEC	FTX CHK

After accepting the configuration, the following can be performed:

Exit on-line configuration

Go Back to Original Configuration

Clicking on *Go back to original configuration* exits on-line configuration mode with the original configuration in control of the process. The changed configuration is removed from the primary/backup module pair.

The original configuration is accepted and controlling the process when the inspect pane shows the following conditions. After clicking on *Go back to original configuration*, observe the inspect pane and wait for the following conditions before continuing:

Address	Function	Mode	Status
N	Primary w/bkup	EXEC	CFC CHK
N+1	Backup	EXEC	FTX CHK

After going back to the original configuration, the following can be performed:

Exit on-line configuration

Make Additional Configuration Changes

Clicking on *Make additional configuration changes* permits making configuration changes after testing a new configuration. The original configuration is put back into control of the process and the test configuration is returned to the *CONFIG* mode.

The original configuration is put in control of the process and the primary/backup module pair is ready to accept configuration changes to the test configuration when the inspect pane shows the following conditions. After clicking on *Make additional configuration changes*, observe the inspect pane and wait for the following conditions before continuing:

Address	Function	Mode	Status
N	Primary w/bkup	EXEC	ERR BAC CFC CHK
N+1	Backup	CONFIG	CFG CFC

After making additional configuration changes, the following can be performed:

Test new configuration Go back to original configuration

Cancel On-Line Configuration

During on-line configuration, if a module does not respond after the communications time-out period as given in the preferences, a message box appears indicating the configuration has not reached its status conditions yet (e.g., checkpointing of all segments). The user has the option of continuing to wait for the module to complete its changes or to stop and leave the on-line configuration view. Some complex or incorrectly configured modules may never complete their changes. The user can

re-enter the on-line configuration later. If the time-out period is lengthy, the user may wish to terminate the wait operation before the time-out period expires. To do so, click on *STOP*.

NOTE: If the on-line configuration process is exited, the state the modules are left in depends on the action occurring when it is stopped. The user is responsible for recovering from this state using the PCU, active, or inspect view and the mode change buttons.

Exit On-Line Configuration

When done (after accepting the new configuration or going back to the original configuration), click on the *Inspect*, *PCU*, or *Active* tab to exit the inspect application on-line configuration mode.

Re-Enter On-Line Configuration

If during on-line configuration, the *PCU*, *Active*, or *Inspect* tabs are used (and before accepting the new configuration or going back to the original configuration), on-line configuration can be re-entered by re-selecting the modules and clicking on the *Online Config* tab. The state of the on-line configuration process is evaluated when on-line configuration is re-entered.

On-Line Configuration Procedure B

This procedure describes the steps required for performing on-line configuration of a primary/backup control module pair lacking enhanced status reporting capability or any primary/backup control module pair operating on Plant Loop.

To perform on-line configuration:

1. Display the primary/backup module pair in the inspect pane.
2. Do one of the following:
 - Select either the primary or backup module of interest.
 - Select both modules of the valid primary/backup pair of interest.
3. Click on the *Online Config* tab. One of the following should occur:
 - If a single module has been selected (either the primary or backup module), a message box appears indicating that the role of the module could not be determined. The user must confirm the primary or backup module by clicking on the appropriate button: *Primary* or *Backup*. A confirmation message appears indicating what will happen if on-line configuration is entered.

- If both modules of the valid primary/backup pair have been selected, a confirmation message appears indicating what will happen if on-line configuration is entered.

NOTES:

If incorrect or invalid module selections are made, one of the following may occur:

1. If no modules are selected, the system signals with a tone.
 2. If more than two modules are selected, a message appears indicating that at least one and no more than two modules must be selected.
 3. If a message box appears indicating that a valid primary/backup module pair was not selected, a valid primary/backup module pair must be selected to perform on-line configuration. Also check the redundancy and on-line configuration switch settings for these modules and that the redundant cable link is properly installed.
4. After successfully entering on-line configuration mode, the following occurs:
 - The primary/backup module pair are the only modules visible in the inspect pane. The backup module mode should show *CONFIG* (after a short delay for switching modes).
 - The mode change buttons are disabled. This is to prevent the user from making mode changes out of sequence and, possibly, inadvertently removing control from the process.
 - A new set of buttons appears in the inspect pane. These are described in the following paragraphs.
 - Two new buttons appear beneath the module mode buttons. These are used to confirm continuation of on-line configuration and to cancel on-line configuration. These buttons are enabled only at appropriate times.

5. Observe the inspect pane and the modules' LEDs and wait for the following conditions before continuing:

Address	Mode	Status	LED 7	LED 8
N	EXEC	ERR BAC		
N+1	CONFIG		Off	On/Blink

6. Any available tool can be used to change the backup module configuration (e.g., the CMU or SLV application). After configuration changes have been made, the inspect pane shows the following conditions:

Address	Mode	Status	LED 7	LED 8
N	EXEC	ERR BAC		
N+1	CONFIG	CFG	Off	On/Blink

7. The following can be performed:

Go back to original configuration
Test new configuration

NOTE: If no configuration changes are made, the only valid operation is *Go Back to Original Configuration*.

The following paragraphs describe the operations that can be performed during on-line configuration. The buttons are enabled and disabled depending on valid operations at each step. Informational messages appear beneath the on-line configuration buttons indicating the state of the requested operation.

Test New Configuration

Test new configuration puts the new configuration in control of the process; the original configuration is preserved within the primary/backup module pair.

To test the new configuration:

1. Click on *Test new configuration*. A message appears in the status bar indicating to wait for LED 8. The *LED 8 On Continue* button is enabled.
2. The new configuration is ready to assume control when the backup module's LED 8 is on. Physically check for this condition, then click on *LED 8 On Continue*.

The roles of the physical modules are switched and the new configuration is controlling the process when the inspect pane and the modules' LEDs show the following conditions. Observe the inspect pane and modules' LEDs and wait for the following conditions before continuing:

Address	Mode	Status	LED 7	LED 8
N	EXEC	ERR BAC CFG		
N+1	EXEC	ERR BAC	Off	On/Blink

After testing the configuration, the following can be performed:

Accept new configuration
Go back to original configuration
Make additional configuration changes

NOTE: If there is an illegal new configuration, the original configuration will remain in control.

Accept New Configuration

Clicking on *Accept new configuration* exits on-line configuration mode with the new configuration in control of the process. The original configuration is removed from the primary/backup module pair.

The new configuration is accepted and controlling the process when the inspect pane and modules' LEDs show the following conditions and the message *On-line Configuration Complete* appears in the status bar. After clicking on *Accept new configuration*, observe the inspect pane and wait for the following conditions before continuing:

Address	Mode	Status	LED 7	LED 8
N	EXEC			
N+1	EXEC	FTX	Off	On/Blink

After accepting the configuration, the following can be performed:

Exit on-line configuration

Go Back to Original Configuration

Go back to original configuration exits on-line configuration mode with the original configuration in control of the process. The new configuration is removed from the primary/backup module pair.

1. Click on *Go back to original configuration*. A message appears in the status bar indicating to wait for LED 8. The *LED 8 On Continue* button is enabled.
2. The original configuration is ready to assume control when the backup module's LED 8 is on. Physically check for this condition, then click on *LED 8 On Continue*.

The original configuration is accepted and controlling the process when the inspect pane and modules' LEDs show the following conditions. After clicking on *Go back to original configuration*, observe the inspect pane and modules' LEDs and wait for the following conditions before continuing:

Address	Mode	Status	LED 7	LED 8
N	EXEC			
N+1	EXEC	FTX	Off	On/Blink

After going back to the original configuration, the following can be performed:

Exit on-line configuration

Make Additional Configuration Changes

Make additional configuration changes permits making configuration changes after testing a new configuration. The original configuration is put back into control of the process and the test configuration is returned to the *CONFIG* mode.

1. Click on *Make additional configuration changes*. A message appears in the status bar indicating to wait for LED 8. The *LED 8 On Continue* button is enabled.

2. The original configuration is ready to assume control when the backup module's LED 8 is on. Physically check for this condition, then click on *LED 8 On Continue*.

The original configuration is put in control of the process and the primary/backup module pair is ready to accept configuration changes to the test configuration when the inspect pane and modules' LEDs shows the following conditions. Observe the inspect pane and modules' LEDs and wait for the following conditions before continuing:

Address	Mode	Status	LED 7	LED 8
N	EXEC	ERR BAC		
N+1	CONFIG	CFG	Off	On/Blink

After making additional configuration changes, the following can be performed:

Test new configuration

Go back to original configuration

Cancel On-Line Configuration

During on-line configuration, if a module does not respond after the communications time-out period as given in the preferences, a message box appears indicating the configuration has not reached its status conditions yet (e.g., checkpointing of all segments). The user has the option of continuing to wait for the module to complete its changes or to stop and leave the on-line configuration view. Some complex or incorrectly configured modules may never complete their changes. The user can re-enter the on-line configuration later. If the time-out period is lengthy, the user may wish to terminate the wait operation before the time-out period expires. To do so, click on *STOP*.

NOTE: If the on-line configuration process is exited, the state the modules are left in depends on the action occurring when it is stopped. The user is responsible for recovering from this state using the PCU, active, or inspect view and the mode change buttons.

Exit On-Line Configuration

When done (after accepting the new configuration or going back to the original configuration), click on the *Inspect*, *PCU*, or *Active* tab to exit the inspect application on-line configuration mode.

Re-Enter On-Line Configuration

If during on-line configuration, the *PCU*, *Active*, or *Inspect* tabs are used (and before accepting the new configuration or going

back to the original configuration), on-line configuration can be re-entered by re-selecting the modules and clicking on the *Online Config* tab. The state of the on-line configuration process is preserved and returned to when on-line configuration is re-entered.

MODULE MODE OPERATIONS

These operations change the module mode. Refer to the appropriate module instruction for details of what occurs for a particular module during a mode change.

NOTE: Any mode changes should be made with extreme care to prevent inadvertent loss of control of the process. An appropriate confirmation message appears prior to performing each mode change request.

Module mode changes are made by selecting one or more modules and clicking on the respective mode change button: *Configure*, *Execute*, *Initialize*, or *Reset*. Multiple modules can be selected using multiselect capabilities. Confirmation dialog boxes may appear after choosing a mode change. If more than one module is selected, a confirmation dialog box appears for each module.

Configure The configure mode takes the module off-line and allows configuration changes to be made using any available utility (e.g., the CMU or SLV applications). All selected modules that are in execute mode are put into configure mode for this mode change. When in configure mode, the module mode field changes to *CONFIG* in the inspect pane.

NOTE: For a primary/backup pair, if the primary module is put into configure mode, the backup module is also put into configure mode. If the backup module is put into configure mode, the primary module remains in execute mode.

Execute Execute places the module in execute mode. The inspect application is the only application that can do this. The SLV application can put a module into configure mode, but the inspect application must be used to return modules to the execute mode. All selected modules that are in configure mode are put into execute mode for this mode change. When in execute mode, the module mode field changes to *EXEC* in the inspect pane.

NOTE: For a primary/backup pair, if either the primary or backup module is selected, both are put into execute mode.

Initialize Initialize clears out the function block and BASIC configuration space. It does not affect the file system blocks (e.g., C and Batch programs). Modules must be in configure mode to be initialized. All selected modules that are in configure mode are initialized for this mode change.

Reset Reset forces the module to start executing its configuration from the beginning. It does not affect the configuration within the module. Its affect on the module operation depends on the module type and whether it is currently in execute or configure mode. Refer to the module instruction for details of the reset mode change operation.

NOTE: For a primary/backup pair, if both modules are selected and both are in execute mode, the functional role of the two modules is swapped (the backup becomes the primary, and the primary becomes the backup, and the primary is reset).

REPORT OPERATIONS

The status and problem report operations provide the ability to detect module or configuration problems. Status and problem report information is not automatically updated and must be requested by the user each time it is needed. The status and problem reports are displayed in the status pane and can be saved to disk. If saved to disk, the reports can optionally be viewed and printed using the default file viewer.

Generate Report To generate status or problem reports for a single module, do one of the following:

- Double click on the module in the inspect pane.
- Select the module of interest and click on *Get Status* or *Get Problems* to generate either or both report types.

View Reports The status and problem reports can be viewed by clicking on the appropriate tab in the status pane: *Status Bytes*, *Status Info*, *Status Errors*, or *Prb Reports*.

NOTE: The status and problem reports shown are not necessarily for the same module.

Status Bytes This report displays the module status bytes. Modules with type codes of 31 and less display five bytes of status. Extended module types display 16 bytes. In addition, this function also displays the status summary information for the selected modules: address, status, mode, type, and firmware revision.

Status Info This report describes the set bit flags in the status bytes and any byte conditions. The descriptions include the byte number, bit number, current value, and the interpretation of the condition.

Status Errors Errors are reported only if the error summary bit in the status byte is set to binary one. If the error summary bit is zero, this option does not appear on the status byte display page. It is different from the status information report in that it identifies only error conditions. It does not give information such as byte

numbers, byte values, etc. This report detects any non-default or error conditions within the module.

Problem Reports

This report gathers all module problem reports and displays them. The problem reports are displayed in a tree list ordered by problem type and address information. The number of problems of each type is shown at the beginning of each type. Expanding any node displays problem reports of that type or other nodes. The hierarchy of the nodes is dependent on the problem report type. **Ctrl-Enter** expands or collapses all child nodes below the selected node.

The right pane view displays the problem reports in the TXTEWS product format. It also displays the relevant bytes if an undefined message is reported.

NOTES:

1. If a large number of reports must be retrieved, the *Get Report* button will gray until all reports are obtained or the capacity to report them is exceeded.
2. If the capacity of either the left or right pane view of the problem reports is exceeded, a message is added to the report indicating that the report is truncated.

Saving to Disk

The information displayed in the inspect pane and the status or problem reports are saved based on report options. The reports can also be viewed and printed from the default log file viewer. If the *Automatically View Report* option in the *Preferences* dialog box is active, then the default log file viewer is launched and the reports are displayed in it.

To save the current status and problem reports to disk:

1. Choose *Reports* from the *File* menu (**Alt-F-P**). The *Report to disk options* dialog box appears.
2. Refer to Table 8-2 for descriptions and check report options as needed. The report options are saved as preferences.

Table 8-2. Report Options

Option	Description
Report entire PCU	Adds entries for all addresses, even if nothing is detected; otherwise, only active modules appear.
Report status and error information	Includes a report section with the information contained in the <i>Status Info</i> and <i>Error Info</i> views.
Report status bytes	Includes a report section with the information contained in the <i>Status Bytes</i> view.
Report problem reports	Includes a report section with the information contained in the <i>Prb Reports</i> view left pane. This option reports the problems in the tree oriented format. It can be either the full report or only the currently visible tree nodes manually expanded or collapsed by

Table 8-2. Report Options (continued)

Option	Description
Report problem reports (continued)	the user. Refer to the Report all problem reports in tree option.
Report all problem reports in tree	Affects how the problem reports are saved. If checked, all problem reports are saved; otherwise, only those viewable in the problem report tree.
Report problem reports like TXTEWS	Includes a report section with the problem reports presented as individually listed problems (as in the TXTEWS product) in the order they were received.

3. Click on *OK* to continue. A *Save As* dialog box appears.
4. Use the default (..\EWS\INSPECT.RPT) or select a path and enter a file name.
5. Click on *OK* to continue.

If the *Automatically view report* preference is enabled, the report appears in the default file viewer.

SECTION 9 - SAVE, LOAD, VERIFY

INTRODUCTION

The save, load, verify utility runs scripts that download configurations from disk files to modules, saves module-resident configurations to disk files, and verifies that source and reference configurations are identical. Scripts are lists containing one or more SLV script commands. This section describes:

- The SLV utility.
- The SLV script generator.
- SLV file utilities.
- SLV commands.

SLV UTILITY

The SLV utility runs SLV scripts from the CMU application, the script generator, and from user-written SLV files. The SLV utility can only process one script file at a time from any source. If the CMU application attempts to launch the SLV utility while it is processing a script, a busy response is returned.

When launched from the CMU application, the scripts are run in a background mode and the SLV utility can be transparent to the user in these operations. The CMU application writes a script to save, load, or verify a configuration and passes it to the SLV utility. Refer to **HARDWARE OPERATIONS (PRIMARY AND BACKUP)** in Section 7 for information on performing saving and loading and **VERIFY OPERATIONS** in Section 7 for information on verifying from the CMU application.

The SLV script generator is accessed from within the SLV utility. It can generate a single command line script based on user-selections from a dialog box. There is no need to know the SLV script command syntax.

The module tools executive launches the SLV utility in order to run a user-written script. Alternatively, user-written scripts can be opened from the SLV utility if it is already running. Refer to **SLV FILE UTILITIES** for information on writing SLV scripts and **SLV SCRIPT COMMANDS** for information on SLV command descriptions and syntax.

The following files are associated with the SLV utility:

file_name.SLV

SLV files contain the SLV scripts. The SLV files can be located in any directory and do not necessarily have to be associated with the project.

file_name.LOG **LOG** files contain processing reports and errors issued by the SLV utility while running the associated script. **LOG** files are overwritten each time the script is run and are located in the directory in which the script is located.

file_name.VFY **VFY** files contain the output of the verify command and are located in the respective module directory.

Where the file name for the **LOG** and **VFY** file is identical to the associated **SLV** file name.

CMU.SLV and CMU.LOG These files are generated only when the SLV utility is launched from the CMU application and are the script and log messages for the CMU generated script. The **SLV** file is passed to the SLV utility. The files are located in the EWS_HOME directory.

Starting the SLV Utility

The SLV utility can be launched from within the context of a project from the module tools executive or CMU application.

To launch the SLV utility without a project, do one of the following:

Application Launcher Click on its icon in the application launcher or choose it from the *Applications* menu.

Windows 3.1 or Windows NT 3.51 Double click on its icon in the *Bailey WinTools* group.

Windows 95 or Windows NT 4.0 Choose *SLV* from the *Bailey WinTools* menu.

The SLV window appears as shown in Figure 9-1

Exiting the SLV Utility To exit the SLV utility, choose *Exit* from the *File* menu (**Alt F X**).

Parts of the Application Window

Title Bar The title bar is a standard Windows title bar showing the application title.

Menu Bar The menu bar contains the SLV application menus. Each SLV menu command is listed as an item on the appropriate menu. Menu selections are made by clicking on the menu and selecting an item, using the menu keys **Alt menu key item key**, or using a shortcut key combination if one is available.

Tool Bar The tool bar shows selected menu commands in graphic form and is an alternative method of choosing commands. The tool bar appears only when enabled using *Tool Bar* from the *View* menu.

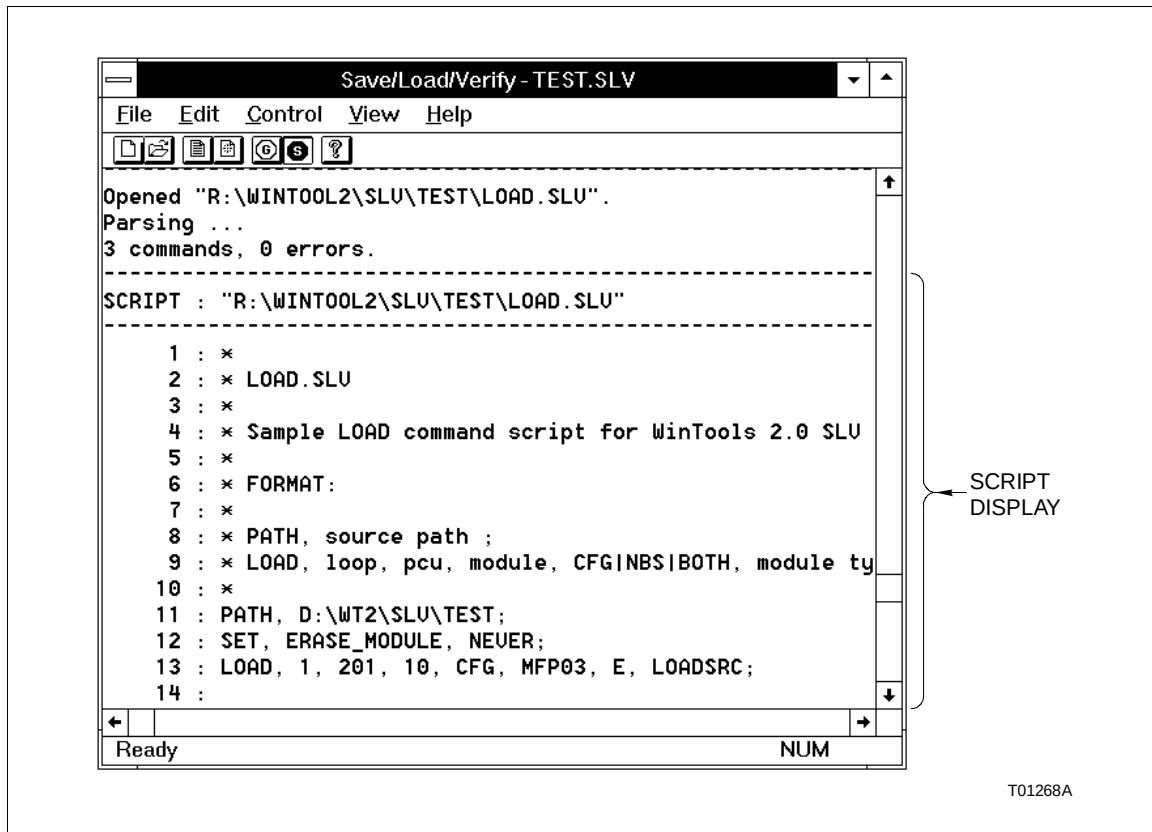


Figure 9-1. SLV Window

- | | |
|-------------------------------|--|
| Application Work Space | The application work space displays a list of messages and scripts from the SLV application. The message list can be scrolled within a maximum number of lines defined in the SLV properties. The list cannot be edited. |
| Status Bar | The status bar provides operation status and short descriptions of the SLV application commands. |

View Operations

- | | |
|-----------------------|---|
| Tool Bar | The tool bar can be toggled on and off by choosing <i>Toolbar</i> from the <i>View</i> menu (Alt V T). |
| Status Bar | The status bar can be toggled on and off by choosing <i>Status Bar</i> from the <i>View</i> menu (Alt V S). |
| Log File | This menu item opens the log file for the current script. To open the log file, choose <i>Log File</i> from the <i>View</i> menu (Alt V L). |
| Verify Results | This menu item opens the verify file for the last verify command. To open the verify file, choose <i>Verify Results</i> from the <i>View</i> menu (Alt V V). |

Message List Size The SLV window has a variable size buffer for displaying SLV messages, scripts, and results. To set the buffer size in number of lines:

1. Choose *Properties* from the *File* menu (**Alt F P**). The *Properties* dialog box appears.
2. Enter a number from 10 to 1000 to set the number of lines in the buffer and click on *OK*.

After the number of lines set for the buffer size is reached, the oldest lines are cleared and cannot be retrieved.

Font The font used in the SLV message list window can be set to any available Windows screen font. To set the message list font:

1. Choose *Properties* from the *File* menu (**Alt F P**). The *Properties* dialog box appears.
2. Click on *Font*. A standard Windows font selection screen appears.
3. Select the desired font and click on *OK*.
4. Click on *OK* to exit the *Properties* dialog box.

Scrolling the Message List Use the scroll bars to scroll through the message list.

Clear Message List Buffer To clear the message list buffer, choose *Clear All* from the *Edit* menu (**Alt E A**).

Script File Operations

This section describes how to run and display user-written script files. Refer to **SLV FILE UTILITIES** for information on writing script files.

Running a Script File To run a user-written script file:

1. Choose *Open* from the SLV utility *File* menu (**Alt F O**).
2. Locate and choose the file of interest.

The file is parsed and checked as it is opened. A message appears in the SLV window indicating the file has been loaded.

3. Choose *Start* from the *Control* menu (**Alt C S**) to run the script file or use the shortcut key **F5** or tool bar button.

As the script runs, messages may appear in the SLV window. Depending on the commands issued and how the properties

are set, confirmation prompts may appear, for example, before a module mode is changed.

When done, the message *Execution Complete* appears.

- Displaying a Script File** To display an opened script file, choose *Display Script* from the *Edit* menu (**Alt E D**).
- Starting and Stopping a Script** To start and stop a script, choose *Start* (**F5**) or *Stop* (**F6**) from the *Control* menu, or use the tool bar buttons.

Script Properties

Script properties can be set that control how a script behaves when certain situations are encountered. These properties are in effect for all scripts until changed or overridden within a script file using the SLV **SET** command.

To set script properties:

1. Choose *Properties* from *File* menu (**Alt F P**). The *Properties* dialog box appears.
2. Refer to Table 9-1 and set the properties as needed.

Table 9-1. Script Properties

Property	Description
Script error	Determines action on a script error: Continue processing at the next command in the script; Stop processing the script commands; Prompt the user.
Overwrite file	Determines action on a file overwrite: Always overwrite files; Never overwrite files; or Prompt the user.
Mode change	Determines action on a module mode change: Allow mode changes; Forbid mode changes; or Prompt the user.
Erase module	Determines action on a module erase (initialize): Always erase; Never erase; or Prompt the user.
Edit headers	Determines action on a CFG file header edit: Always edit; or Never edit.

SLV SCRIPT GENERATOR

The script generator automatically writes single-line scripts based on user choices from a dialog box. This can be used to quickly perform a save, load, or verify operation if, for example, the project is unavailable. These scripts are temporary and are not saved. Refer to **SLV SCRIPT COMMANDS** for information on the script fields.

Save To generate a save script:

1. Choose *New* from the *File* menu (**Alt F N**). The *Script Generator* dialog box appears.
2. Select the *Save* command and click on *OK*. The *Save Command* dialog box appears.
3. Enter information as needed and click on *OK* to continue. The script command line appears in the SLV window.
4. To run the script, choose *Start* (**F5**) from the *Control* menu or use the tool bar button.

Load To generate a load script:

1. Choose *New* from the *File* menu (**Alt F N**). The *Script Generator* dialog box appears.
2. Select the *Load* command and click on *OK*. The *Save Command* dialog box appears.
3. Enter information as needed and click on *OK* to continue. The script command line appears in the SLV window.
4. To run the script, choose *Start* (**F5**) from the *Control* menu or use the tool bar button.

Verify To generate a verify script:

1. Choose *New* from the *File* menu (**Alt F N**). The *Script Generator* dialog box appears.
2. Select the *Verify* command and click on *OK*. The *Save Command* dialog box appears.
3. Enter information as needed and click on *OK* to continue. The script command line appears in the SLV window.
4. To run the script, choose *Start* (**F5**) from the *Control* menu or use the tool bar button.

SLV FILE UTILITIES

The SLV file utilities are used to create, edit, and check SLV files. SLV files are text files containing a list of SLV script command lines (scripts). Refer to **SLV SCRIPT COMMANDS** for SLV script command descriptions and examples. SLV files can be located anywhere and do not need to be associated with a project.

- | | |
|--------------------------|---|
| Open an SLV File | <p>To open an SLV script file:</p> <ol style="list-style-type: none"> 1. Choose <i>Global Save/Load/Verify Utilities</i> from the module tools executive <i>Utility</i> menu (A H T G). The <i>Global Save/Load/Verify</i> dialog box appears. 2. Do one of the following: <ul style="list-style-type: none"> • To open an existing file, locate and select the file. • To create a new file, set the drive and directory as needed, then type in the file name. 3. Click on <i>Open</i>. The file is opened in the default WinTools file editor. 4. Refer to SLV SCRIPT COMMANDS and add or edit command lines as needed. When done, save the file and close the editor. The <i>Global Save/Load/Verify</i> dialog box re-appears. |
| Check an SLV File | <p>SLV files can be parsed and checked for syntax errors prior to running the file. To check an SLV script file:</p> <ol style="list-style-type: none"> 1. Choose <i>Global Save/Load/Verify Utilities</i> from the module tools executive <i>Utility</i> menu (A H T G). The <i>Global Save/Load/Verify</i> dialog box appears. 2. Locate and select the file. 3. Click on <i>Check</i>. The file is opened and checked for errors. The SLV utility is launched and results are displayed in the SLV message list. |
| Run an SLV File | <p>SLV files can be run from the module tools executive. To run an SLV script file:</p> <ol style="list-style-type: none"> 1. Choose <i>Global Save/Load/Verify Utilities</i> from the module tools executive <i>Utility</i> menu (A H T G). The <i>Global Save/Load/Verify</i> dialog box appears. 2. Locate and select the file to run. 3. Click on <i>Run</i>. The file is passed to the SLV utility for processing. Results are displayed in the SLV message list. |

SLV SCRIPT COMMANDS

This section describes the save, load, verify, and associated commands. New commands and command forms have been implemented with this release; however, SLV scripts from previous releases of the module tools are supported.

SLV script commands have the following features:

- Case insensitive: **SAVE**, **Save**, and **save** are treated the same.
- Descriptive and short form command names for many commands: **Save** and **S** are treated the same.
- Comments can be included. Comment lines begin with the * character.
- The following can be used for path separators: \, \\, and /.

Delay

PURPOSE: The delay command specifies a processing delay. It can be used to reduce the load on the loop.

COMMAND: **Delay**, *seconds*;

D, *seconds*;

where:

<i>seconds</i>	Whole number of seconds of the delay.
----------------	---------------------------------------

Example * The following provides a delay of 20 seconds.

Delay, 20;

Load

PURPOSE: Downloads a module configuration (**CFG**) or BASIC or C program from a disk file to the module at the specified loop, PCU, and module address. The load command has four subcommands to specify the type of file to download to the module.

COMMAND: **Load**, *loop*, *PCU*, *module*, *sub_command*, *type*, *revision*, *file_name*;

L, *loop*, *PCU*, *module*, *sub_command*, *type*, *revision*, *file_name*;

where:

<i>loop</i> <i>PCU</i> <i>module</i>	Loop, PCU, and module address of the target module.
--	---

<i>sub_command</i>	The sub_command field accepts the following entries:
--------------------	--

CFG	- download only the module configuration.
------------	---

<i>sub_command</i> (continued)	B_PROG - download only the associated BASIC language program. C_PROG - download only the associated C language program. This <i>sub_command</i> may erase existing CFG configurations in the module if the file system size is different from the current module size and the module needs to be reformatted. ALL - download the module configuration and associated BASIC and C programs.
<i>type</i>	Module type; for example, MFP03, COM01 .
<i>revision</i>	Module revision.
<i>file_name</i>	Module configuration file name.

NOTE: All **Load** command parameters except for *sub_command* are optional if an associated **Target** command is used prior to the **Load** command. Refer to the **Target** command for an example.

Example *Download the configuration pump4 to the module at address 1, 2, 2.
Load, 1, 2, 2, CFG, MFP03, F, c:\utility\l1\p2\m2\pump4;

Path

PURPOSE: The path command allows changing the current working directory. The specified path remains in effect as the current directory until another path command is issued or is encountered.

COMMAND: **Path**, [*drive*:]*path*;
P, [*drive*:]*path*;

where:

<i>drive</i>	Any valid drive.
<i>path</i>	Any valid path.

Example * Change the current working directory to C:\Project1\L1\M2.
Path, C:\Project1\L1\P3\M2;

Save

PURPOSE: Saves a module resident configuration or program to a disk file. The save command has three subcommands to specify the type of file to save to disk.

COMMAND: **Save**, *loop*, *PCU*, *module*, *sub_command*, *type*, *revision*, *file_name*;

S, *loop*, *PCU*, *module*, *sub_command*, *type*, *revision*, *file_name*;

where:

loop Loop, PCU, and module address of the
PCU source module.
module

sub_command The *sub_command* field accepts the fol-
lowing entries:

CFG - saves only the module
configuration.

NBS - saves only the associated
programs.

Both - saves the module configuration
and associated programs.

type Module type; for example, **MFP03**,
COM01.

revision Module revision.

file_name Module configuration file name.

NOTE: All Save command parameters except for *sub_command* are
optional if an associated Target command is used prior to the Save
command. Refer to the Target command for an example.

Example *Save the module resident configuration at address 1, 1, 6 to disk file Pump1;
 *and associated programs.

Save 1, 1, 6, Both, MFP03, F, Pump1;

Set

The set commands temporarily set SLV utility preferences for
the duration of the current script. After running the script, any
preferences set by this command revert to the default that is
set from the SLV utility preference dialog box. There are five
Set command to set each of the five properties.

PURPOSE: Determines what to do if a script error is encountered while
running a script.

COMMAND: **Set**, **Script_Error**, [**Continue** | **Stop** | **Prompt**];

where:

Continue Continue processing the script.

Stop Stop processing the script.

Prompt Prompt the user.

(no parameter) Revert to preference setting.

PURPOSE: Determines what to do if a file will be overwritten while running a script.

COMMAND: **Set, Overwrite_File, [Always | Never | Prompt];**

where:

Always Always overwrite files.

Never Never overwrite files.

Prompt Prompt the user.

(no parameter) Revert to preference setting.

PURPOSE: Determines what to do if a module mode change is required while running a script.

COMMAND: **Set, Mode_Change, [Allow | Forbid | Prompt];**

where:

Allow Allow mode changes.

Forbid Forbid mode changes.

Prompt Prompt the user.

(no parameter) Revert to preference setting.

PURPOSE: Determines what to do if a module needs to be erased encountered while running a script.

COMMAND: **Set, Erase_Module, [Always | Never | Prompt];**

where:

Always Always erase.

Never Never erase.

Prompt Prompt the user.

(no parameter) Revert to preference setting.

PURPOSE: Determines what to do if a CFG file header needs to be edited while running a script.

COMMAND: **Set, Edit-Headers, [Always | Never];**

where:

Always	Always edit.
Never	Never edit.
(no parameter)	Revert to preference setting.

Example *While running the current script, set the properties so that the user
 *is prompted when a script error, file overwrite, mode change, or module
 *erase is encountered and always allow headers to be edited.

```
Set, Script_Error, Prompt;
Set, Overwrite_File, Prompt;
Set, Mode_Change, Prompt;
Set, Erase_Module, Prompt;
Set, Edit_Headers, Always;
```

Target

PURPOSE: The target command sets a target loop, PCU, and module address, module type, and module revision for the duration of the script or until another target command is encountered or issued. The target address is substituted into **Load**, **Save**, and **Verify** commands in which the address fields are left blank. This can be when writing a script file to copy commands for use throughout a project and then inserting **Target** commands as needed to set the addresses properly.

COMMAND: **Target**, *loop*, *PCU*, *module*, [*type*], [*revision*];

T, *loop*, *PCU*, *module*, [*type*], [*revision*];

where:

<i>loop</i>	Loop address.
<i>PCU</i>	PCU address.
<i>module</i>	Module address.
<i>type</i>	Module type; for example, MFP03 , COM01 .
<i>revision</i>	Module revision.

Example * Set the target module to 1,1,5, load the configuration, then change the
 * target to 1,2,5 and perform the same operation.
 * The script can be copied as many times as needed by changing
 * the target address. This permits a cut-and-past operation instead of having
 * to retype the Load and Verify command lines over each time.

```
Target, 1, 1, 5, MFP01, A;
Load, , , , CFG, , ;
Verify, , ;
```

Target, 1, 2, 5, ,;
Load, , , , **CFG**, , ,;
Verify, ,;

V

PURPOSE: Verifies a configuration file (**CFG**) against a module-resident configuration.

NOTE: This is an obsoleted command provided for compatibility with SLV script command files created with previous revisions of the module tools. Use the Verify (Vy) command for creating new scripts.

COMMAND: **V**, *loop*, *PCU*, *module*, , *type*, *revision*, *file_name*;

where:

<i>loop</i>	Loop, PCU, and module address of the reference module.
<i>PCU</i>	
<i>module</i>	
<i>type</i>	Module type; for example, MFP03, COM01 .
<i>revision</i>	Module revision.
<i>file_name</i>	The CFG file name to be verified against the reference module.

Example *Verify the configuration file **PUMP1.CFG** against the module-resident configuration at address 1, 1, 6.

V, 1, 1, 6, ,MFP03, F,Pump1;

Verify

PURPOSE: Verifies a configuration in the form of a **CFG** file or **CAD** sheets against a **CFG** file or module-resident configuration.

COMMAND: **Verify**, *source*, *reference*, **Yes** | **No**, , *source_info*, *reference_info*;

Vy, *source*, *reference*, **Yes** | **No**, , *source_info*, *reference_info*;

where:

<i>source</i>	The source field can be one of the following: CFG - indicates the source is a CFG file. CAD - indicates the source is a group of CAD sheets.
---------------	--

<i>reference</i>	The reference field can be one of the following: CFG - indicates the reference is a CFG file. Module - indicates the reference is a module-resident configuration.
Yes No	This field is required only if the <i>source</i> field is CAD. Otherwise, leave it blank. Yes - update CAD sheet specifications. No - do not update CAD sheet specifications.
<i>source_info</i>	The <i>source_info</i> field depends on the source type: If the source type is CFG, <i>source_info</i> must be the CFG file name in the form: <i>[drive:][path]file_name[.CFG]</i> If the source type is CAD, <i>source_info</i> must be the module header file name.
<i>reference_info</i>	The <i>reference_info</i> field depends on the reference type: If the reference type is CFG, <i>reference_info</i> must be the CFG file name in the form: <i>[drive:][path]file_name[.CFG]</i> If the reference type is Module, <i>reference_info</i> must be the module address, module type, and module revision in the form: <i>loop, PCU, module, type, revision</i>
Example	*Verify the configuration file PUMP1.CFG against the module-resident configuration at address 1, 1, 6. This example performs the same verify as the example in the V command. Verify, CFG, Module, , C:\Project1\L1\P1\M6\Pump1,1, 1, 6, ,MFP03, F;
Example	* Verify a CAD configuration against a CFG configuration and update the CAD sheet specifications. *Make C:\Project1\L1\P1\M6 the current working directory first. Path, C:\Project1\L1\P1\M6; Verify, CAD, CFG, Yes, Pump1, Pump1AA.CFG;

Script File Example

The following is an example of an SLV script file.

```

*SLV Script for Unit 1, Area 51 systems
* Save current configurations for Pump 1 and 2
* Verify against new configuration to check for any unknown changes in the hardware
* Load new configuration in pump 1 and 2
* Verify Pump 3 for any field changes
* Script Revision: C, 4/02/97
*
*
* Set SLV to continue processing script commands if any of the commands has an error.
SET SCRIPT_ERROR, CONTINUE;
*
* Set SLV to overwrite the CFG file if it already exists.
SET OVERWRITE_FILE, ALWAYS;
*
*PUMP1
*
* Set Target address for the next series of Commands
TARGET, 1, 23, 3, MFP02, F;
*
* Set path to project location to shorten file names in commands
PATH, D:\UNIT1\L1\P23\M03;
*
*Save Pump 1 module configuration as backup 3
SAVE, , , CFG, , PUMP1-B3;
*
* Verify new configurations sheets against the saved CFG of the current configuration
* do not update the CAD sheets, use disk files to prevent unnecessary loop traffic
VERIFY, CAD, CFG, NO, PUMP1, PUMP1-B3;
*
* Set SLV to automatically put the modules into configure mode if needed;
SET Mode_Change, ALLOW;
*
* Set SLV to erase the old configuration prior to loading the new configuration.
SET Erase_Module, ALWAYS;
*
* LOAD new configuration into the target module using Target and Path from above
LOAD, , , , CFG, , , PUMP1;
*
****PUMP2
*
* Set Target address for the next series of Commands
TARGET, 1, 34, 3, MFP02, F;
*
* Set path to project location to shorten file names in commands
PATH, D:\UNIT1\L1\P34\M03;
*
*Save Pump 2 module configuration as backup 3
SAVE, , , , CFG, , PUMP2-B3;
*
* Verify new configurations sheets against the saved CFG of the current configuration
* do not update the CAD sheets, use disk files to prevent unnecessary loop traffic
VERIFY, CAD, CFG, NO, PUMP2, PUMP2-B3;
*
*
**** PUMP3
*
* LOAD new configuration into the target module using Target and Path from above
LOAD, , , , CFG, , , PUMP2;
*
* Set Target address and type for the next series of Commands
TARGET, 1, 43, 3, MFP02, F;
* Set path to project location to shorten file names in commands
PATH, D:\UNIT1\L1\P34\M03;

* Verify CAD sheets against the existing configuration in the hardware and update any field changed

```

* specifications.

VERIFY, CAD, MODULE, YES, PUMP3, . . . , ;

*

*End of Script

SECTION 10 - TREND, MONITOR, AND TUNE

INTRODUCTION

This section describes how to use the trend application to perform the following functions:

- Define and save trend and monitor groups.
- Monitor up to 32 point values in real time.
- Trend up to 10 point values in real time.
- Tune function blocks as they execute the control strategy.

The trend application polls function blocks for values. It can trend, monitor, and tune any point in any module it has access to over the INFI-NET network. Real time point values are displayed graphically for trending and in tabular format for monitoring. These values are not recorded by the application and cannot be recalled.

NOTE: Monitoring and tuning can also be performed using the GMC application. Refer to **MONITORING A CONFIGURATION** in Section 5.

STARTING THE APPLICATION

To start the trend application, do one of the following.

CMU	1. Select a monitor/trend group and choose <i>Monitor Group</i> or <i>Edit Group Definition</i> from the <i>CMU Monitor/Trend</i> menu. 2. Select a monitor/trend group from the pick list that appears and click on <i>OK</i>.
Application Launcher	Click on its icon in the application launcher or choose <i>Trend</i> from the <i>Application</i> menu (Alt A T).
Windows 3.1 or Windows NT 3.51	Double click on the <i>Trend</i> icon in the <i>Bailey WinTools</i> group.
Windows 95 or Windows NT 4.0	Choose <i>Trend</i> from the <i>Bailey WinTools</i> menu.

EXITING THE APPLICATION

Exiting the trend application closes the application window. If there are unsaved changes to the group, a prompt appears to save the group before exiting.

To exit the trend application, choose *Exit* from the *File* menu (**Alt F X**).

APPLICATION WINDOW

This section describes the trend application window, the trend menus, and how to access on-line help.

Parts of the Application Window

The parts of the trend application window are shown in Figure 10-1 and described in the following paragraphs.

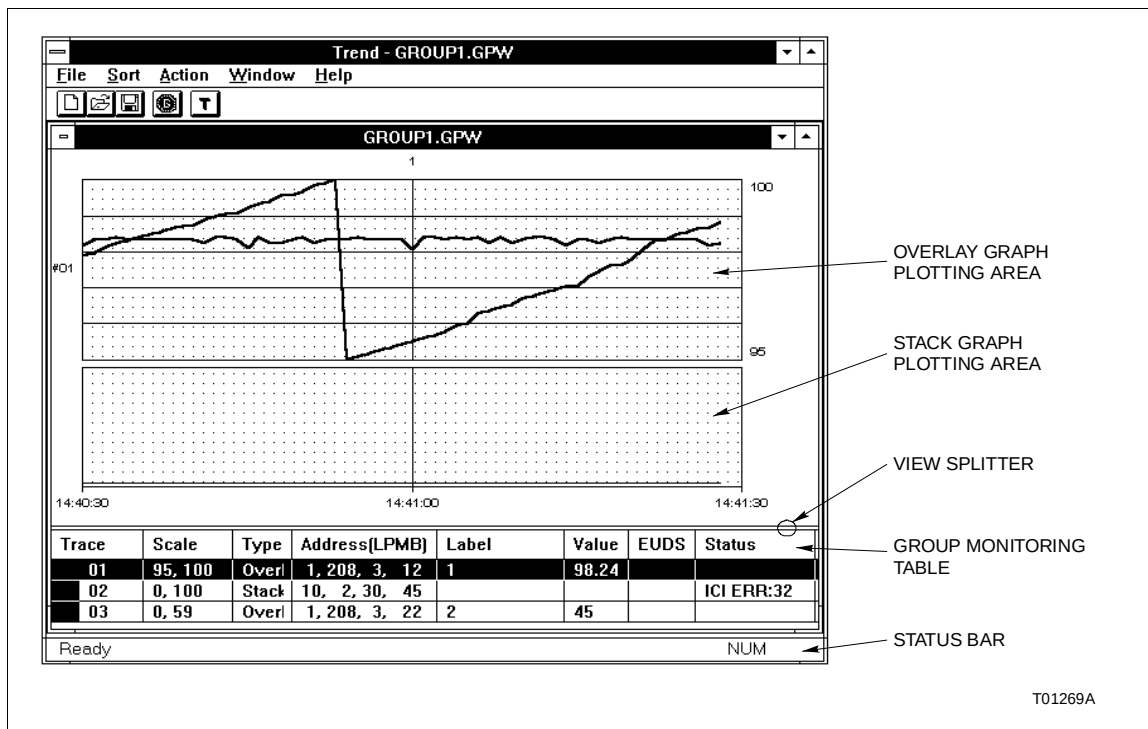


Figure 10-1. Trend, Monitor, and Tune Application Window

- Title Bar** The title bar is a standard Windows title bar showing the application title. The group name appears in a document title bar. If the document is maximized, this information appears appended to the application title.
- Menu Bar** The menu bar contains the trend application menus. Each trend command is listed as an item on the appropriate menu. Menu selections are made by clicking on the menu and selecting an item, using the menu keys **Alt [menu key] [item key]**, or using a shortcut key combination if one is available.
- Tool Bar** The tool bar shows *File* and *Action* menu commands in graphic form. The tool bar is an alternative method of choosing commands.

Application Work Space	The application work space contains one group window and one function code edit window. The group window contains the trend and monitor data for that group. If a group or function code edit window is minimized, it appears as an icon at the bottom of the application work space.
Group Window	The group window contains the trend plotting areas and group monitor table for the group.
Overlay Graph Plotting Area	Any traces assigned to the overlay graph appear in this area. Traces are overlaid on each other in the overlay graph with scale limits spanning the full height of the plotting area.
Stack Graph Plotting Area	Any traces assigned to the stack graph appear in this area. Each trace occupies a separate plotting region and they are stacked on top of each other in the stack graph. Scale limits span the trace's assigned region. If a trace exceeds its assigned scale limit, it overlays the other regions. The size of the assigned region is determined by evenly dividing the area by the number of traces assigned to the area.
Group Monitoring Table	This table shows all defined points and associated information about them for the group. A maximum of 32 points can be defined. If more points are defined than can fit in the space provided, scroll bars appear to facilitate access to all points.
View Splitter	The percentage of space given to the plot areas and to the group monitoring table within a group window can be adjusted using the view splitter.
Function Code Edit Window	Refer to FUNCTION CODE EDITOR for information.
Status Bar	The status bar appears at the bottom of the application window. It displays messages on the current status of the trend application and brief help instructions.

Menu Items

The following paragraphs briefly describe each trend menu. For a complete description of each menu item, refer to the subsections that follow.

- | | |
|-------------|---|
| File | Contains items to open, save, close, and create groups. Refer to GROUP OPERATIONS . Contains the <i>Properties</i> item to set graph options. Refer to Graph Display Options for information. |
| View | Appears when the function code edit window has the focus. It has items to filter the function code editor view. Refer to FUNCTION CODE EDITOR for information. |

Sort	Appears when the group window has the focus. Contains items to sort the group definition table. Refer to MONITOR AND TREND A GROUP .
Action	Contains items to start and stop scanning, refer to MONITOR AND TREND A GROUP for information. Contains items to add, edit, and delete points, refer to Editing a Group Item for information. Contains an item to tune a point, refer to TUNE A POINT for information.
Window	Contains items to manage the group and function code editor windows. Refer to WINDOW OPERATIONS .
Help	Contains items to access help, refer to On-Line Help .

On-Line Help

On-line help is available for the trend application. It provides command descriptions, procedures and reference material similar to that found in the instruction. The on-line help can be used for convenient and quick access to the trend application operating procedures. The help system conforms to the standard Windows conventions for on-line help. Refer to **ON-LINE HELP** in Section 3 or the Windows documentation for more information.

Tool Bar

The tool bar provides an alternate and more convenient method of choosing commands. File and scan commands are represented by an icon on the tool bar. To choose a command from the tool bar, point and click on the icon.

WINDOW OPERATIONS

The trend application window can contain two windows: a group window and a function code editor window. When a group is opened, it appears in the group window. The trend group window includes a title bar, control menu, maximize and minimize buttons, and scroll bars. Operation of these components conforms to the Windows conventions and complete information can be found in the Windows documentation. Group windows have the following features:

- The group name appears in the group window title bar. If the group window is maximized, this information appears in the trend application title bar.
- If the group window is minimized, it appears as an icon at the bottom of the trend application work space; it is visible if the trend application work space is not covered by a group window.

- The group window can only be moved within the trend application window.

For information on the function code editor window, refer to **FUNCTION CODE EDITOR**.

GROUP OPERATIONS

A group refers to a defined collection of points that can be monitored. A maximum of 32 points can be defined in a group; of these, up to 10 can also be defined as trended points.

Opening a Group

WinTools group definition files (**WGP**) are located in any project directory. To open a group:

1. Choose *Open* from the *File* menu (**[Alt][F][O]**) or use the shortcut key combination **[Ctrl][O]**.
2. The *Open* dialog box appears. If necessary, change the drive and directory.
3. Choose the file to be opened by one of the following methods:
 - Select the file name from the file list box and click on *OK*.
 - Double click on the file name in the file list box.
 - Type the file name into the file name box and click on *OK*.
 - The most recently used groups appear at the bottom of the *File* menu. To open one of these groups, click on the group name.

The chosen group appears in the group window and becomes the active window.

Creating a New Group

To create a new group:

- Choose *New* from the *File* menu (**[Alt][F][N]**) or use the shortcut key combination **[Ctrl][N]**.

A new blank group appears in the group window. Refer to **Editing a Group Item** for information on adding points to the group. A newly created group is given the temporary name *Groupn*, where n is the nth created group during the current session.

Editing a Group Item

Points can be added to and deleted from the group. Existing points in the group can be edited.

NOTE: The group cannot be edited when scanning is on. If necessary, turn scanning off by choosing *Stop* from the *Action* menu (**[Alt] [A] [S]**) or tool bar.

Adding a Point To add a point to the group:

- Choose *Add* from the *Action* menu (**[Alt] [A] [A]**). A new, empty point definition row appears in the group definition table. Use *Edit* from the *Action* menu to edit the point definition.

Deleting a Point To delete a point from the group:

- Select the point by pointing and clicking on it in the group definition table.
- Choose *Delete* from the *Action* menu (**[Alt] [A] [D]**). The point definition is removed from the group.

Editing a Point To edit a point in the group:

- Select the point by pointing and clicking on it in the group definition table.
- Choose *Edit* from the *Action* menu (**[Alt] [A] [E]**). The *Edit Trend/Monitor Point* dialog box appears.
- Refer to Table 10-1 and fill in the point definition information as needed.

Table 10-1. Point Definition

Field	Description
Label	Descriptive label for the point to be monitored.
Address	Loop, PCU, module, and block address of the point to be monitored.
Type	None - no trace is assigned, the point is monitored. Overlaid - the trace is presented in the upper overlay graph area. All traces in this area are overlaid. Stacked - the trace is presented in the lower stack graph area. All traces in this area are stacked. Typically, digital traces are easier to view if they are stacked rather than overlaid.
Trace	Assigns a trace number and color for the point defined for trending. Enabled only if the <i>Type</i> field is <i>Overlaid</i> or <i>Stacked</i> . There are a maximum of 10 traces allowed. If a trace number is used that was previously assigned to another point, then the trace assignment is removed from the old point.

Table 10-1. Point Definition (continued)

Field	Description
Scale limits	The high and low scale limits for an analog point. For digital points, it is recommended to use 0 and 1.
Units	The engineering unit descriptor or logic state descriptor.

- Click on *OK* to accept the definition.

Saving a Group

A group can be saved at any time. It is recommended to save after any changes to prevent loss of data. Groups can be saved to any project directory.

To save a group, choose *Save* from the *File* menu (**Alt F S**) or tool bar or use the shortcut key combination **Ctrl-S**.

A group can also be saved under a different name using *Save As* from the *File* menu (**Alt F A**).

Graph Display Options

Trend graph properties determine how the plots appear. The properties are saved with each group.

To set the graph display options:

- Choose *Properties* from the *File* menu (**Alt F T**). The *Trend Graph Properties* dialog box appears.

NOTE: The graph properties cannot be edited when scanning is on. If necessary, turn scanning off by choosing *Stop Scanning* from the *Action* menu (**Alt A S**) or tool bar.

- Refer to Table 10-2 and set the options as needed.

Table 10-2. Trend Graph Properties

Field	Description
Graph width	Time interval in minutes that the width of the graph represents. Valid entry is 2 to 1440 (one day).
Time label	Time interval between vertical time marks and labels in minutes. The entry must be greater than 20% of the graph width value.
Grid refresh	As the traces update, the horizontal and vertical grid lines may break up where they had been overwritten by the traces. Grid refresh determines how often the grid is updated.

Table 10-2. Trend Graph Properties (continued)

Field	Description
Sample interval	Time between samples in seconds. Affects the rate that the trend graph is updated and the <i>Value</i> is updated in the group monitor table. The entry must be greater than 1% of the graph width.
Graph reset	Percentage of the trailing edge of the plot area that is cleared when the trace reaches the right edge of the graph.
Overlay region	Percentage of total graph area allowed for the overlay graph. The remaining area is given to the stack graph.
Y-axis ticks	The percentage given is the vertical distance between any pair of grid lines as a percentage of the total height of the overlay plot area.

3. Click on *OK* to save the properties and return to the group window.

MONITOR AND TREND A GROUP

To monitor and trend points in a group:

1. Open the group of interest or make it the active window.
2. Start displaying data by choosing *Start Scanning* from the *Action* menu (**Alt A S**) or tool bar. Graphed data will appear in the trend graph areas and the *Value* fields of the group monitor table will display values received from the INFI-NET network.

As the trend traces reach the right edge of the plot area, a trailing edge of the plot equal to the *Graph Reset* percentage defined in the *Trend Graph Properties* dialog box is cleared and the plot is shifted by that amount to the left.

Refer to Table 10-3 for a description of the columns in the group definition table.

NOTE: When scanning is active, changes cannot be made to the point definitions or the graph properties.

Table 10-3. Group Monitor Table

Column	Description
Trace	Trace color and number as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point.
Scale	Scale limits as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point.
Type	Graph type as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point.

Table 10-3. Group Monitor Table (continued)

Column	Description
Address	Loop, PCU, module, and block address as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point.
Label	Descriptive label as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point.
Value	Current value as reported over the INFI-NET for the monitored point. Only valid when scanning. Updated at intervals set by <i>Sample Interval</i> in the <i>Trend Graph Properties</i> dialog box.
EUDS	Engineering unit descriptor as assigned in the <i>Edit Trend/Monitor Point</i> dialog box for that point. For digital points, valid logic state descriptors can be entered.
Status	Current status as reported over the INFI-NET for the monitored point. Only valid when scanning. Updated at intervals set by <i>Sample Interval</i> in the <i>Trend Graph Properties</i> dialog box. Refer to Table 10-4 for <i>Status</i> descriptions.

Table 10-4. Status Indicators

Indication	Description
Blank	OK.
*	Bad quality.
A	Indicates that a point is in alarm.
C	Calibration correction; values out of range.
d	Low deviation.
D	High deviation.
H	High alarm (may be preceded by alarm level, e.g., 3H).
L	Low alarm (may be preceded by alarm level, e.g., 2L).
S	Point is being serviced (NAMM02 only).

3. The points can be arranged in the group monitor table using the *Sort* menu. Refer to Table 10-5 for a description of the sort orders.

Table 10-5. Sort Order

Method	Description
Trace	Orders trended points by trace number.
Type	Orders the points by trace type.
Label	Orders the points alphabetically by the <i>Label</i> .
Address	Orders the points by address.

4. When done, turn scanning off by choosing *Stop Scanning* from the *Action* menu (**Alt** **A** **S**) or tool bar.

TUNE A POINT

Function blocks can be tuned by observing the trend plots or monitored values and making corrections to tunable specifications.

To tune function block specifications:

1. Choose *Tune* from the *Action* menu (**Alt A T**) or tool bar. A *Select FC Address for Tuning* dialog box appears.
2. The last used points are shown in the pick list. Select a point to be tuned from the pick list or type in a new address.
3. Click on *OK* to continue.

The function code editor window appears and the *View* menu contains filter options for the function code editor. The *Standard* tab shows only tunable specifications. The second tab shows the filtered view as set by the *View* menu options. Refer to **FUNCTION CODE EDITOR** for information on using the function code editor.

4. Only tunable specifications can be edited. Make any needed changes to the tunable specifications of the function block.
5. Click on *Apply* to send the changes to the module.

FUNCTION CODE EDITOR

The function code editor facilitates tuning function blocks. It incorporates several features that make configuring function blocks easier. User-friendly features of the function code editor include specification filtering, application tabs, and specification translation.

Specification Filtering	Specification filtering refers to the ability to filter and view the specifications by type (all, inputs, internal, tunable, on-line configurable, and spare).
Application Tabs	This feature filters function code specifications by suitability to function. For example, a function code 216 can be used for TC, RTD, high level, and low level applications. When this function code is configured, a tab appears for each of these applications. By clicking on the appropriate tab, only those specifications needing configuration for that application are displayed.
Specification Translation	Specification translation refers to the ability to configure complex specifications by picking options from radio buttons and pick lists. The configuration is automatically translated to the numeric value and entered as the specification value.

Function Code Edit Window

The function code editor is used to view or modify function code specifications. An example function code editor window is shown in Figure 7-1.

Title Bar	The title bar shows the loop, PCU, and module address being configured.
Function Code Definition	The function code definition bar shows the block address, function code number, and function code description being configured.
Tab Bar	The tab bar is used by function codes with advanced specification filtering requirements. By clicking a particular tab, only the specifications needed to configure the function code for that usage is displayed. All function codes have a standard and filter tab. The standard tab shows all configurable specifications. The filter tab shows a filtered subset of the specifications based on the application's filter or view options. Additional tabs may be present to provide a logical subset of specifications to ease the configuration process.
Controls	The controls are described in the following subsection.
Specification Display Frame	This area can show all the specifications (as filtered by the tab bar). The view splitter adjusts the amount window space used by the specification display frame and the specification details frame. Scroll bars can be used to view parts of the specification display frame not currently visible. Table 10-6 describes the columns used in this area.

Table 10-6. Function Code Specifications

Item	Description
Spec	Specification number as listed in the Function Code Application Manual.
Tune	Indicates whether the specification is tunable or not.
Type	Data type of the specification: Integer, Real, Character.
Value	Specification value.
Translation	Meaning of the specification value for binary or decimal coded specifications. An out of range message appears for integer and real values if they are out of range. If the translation needs more room, it is extended into the description column and the description appears in the following line.
Description	Description of the specification.

Specification Details Frame	This area is used to configure the function code specifications. If the specification requires an integer, real, or string value, it can be entered here. If the specification requires a binary or
------------------------------------	---

decimal coded value, pick lists appear from which, for example, input devices can be selected. For a function code 216, input types of thermocouple, RTD, high level, low level, and user defined appear. After selecting an input type, a subtype can be selected. These selections are automatically translated into the decimal code and entered as the function code specification. The behavior and appearance of specification details depend on the specific function code being configured.

Controls

The presence of the controls depends on the application and the function being performed. The function code editor controls operate in the following manner:

<i>APPLY</i>	Applies the current specification values to the configuration. If on-line or tuning, or performing hardware configuration, the values are applied to the module configuration residing in the module. Otherwise, these values are saved with the configuration on the next save.
<i>RESET</i>	Resets all specifications to their values prior to the current editing session or since they have been applied.
<i>CLOSE</i>	Closes the function code edit window.
<i>DEFAULT</i>	Each specification has a default value. This button sets the current specification to its default value.
<i>DEFAULT ALL</i>	Sets all specifications to their default values.
<i>PREV FC</i>	Displays the function code at the next block in the configuration.
<i>NEXT FC</i>	Displays the function code at the previous block in the configuration.

Viewing/Filtering Options

These options affect the display of the specifications in the function code editor when the filter tab is selected. The function code edit window must be the active window to set its filter options. Filter options appear on the applications *View* menu.

Overview	Displays all usable specifications defined for the function code.
Inputs	Displays input specifications only.
Internal	Displays internal specifications only.

Tunable	Displays tunable specifications only.
On-Line Configurable	Displays on-line configurable specifications only.
Spares	Display spare and reserved specifications for any of the other filter options.

Editing Specifications

Specifications can be edited by typing a value in the *Value* field of the specification display frame or in the specification details frame. Changes do not occur until **Enter** is pressed. If a range value is entered that exceeds the specification's defined minimum or maximum as displayed in the status field, an out of range message appears in the translation field. When editing specifications, a context menu can be used for editing operations (undo, cut, copy, paste, and delete).

If the specification uses a binary or decimal coded value, a pick list appears for the specification in the specification details frame. The value can be entered by selecting from a pick list rather than entering the binary or decimal code. If a value is entered that cannot be represented by the pick list or available buttons, an undefined message appears.

If a value is entered that exceeds the storage capability of the data type, an *Illegal* message is displayed and the cursor remains on that specification field until it is changed to an allowable value.

After making editing changes, use *Apply* to apply the changes to the module configuration.

SECTION 11 - SUPPLEMENTAL CONFIGURATION UTILITIES

INTRODUCTION

This section describes the following supplemental configuration utilities:

- Tag synchronization.
- Configuration print utility.
- Error log file viewer.
- Communications configuration.
- DSOE configuration.

TAG SYNCHRONIZATION

Tag synchronization is used to verify that tag data in the **CAD** sheets and the tag data tables agree and update it if it does not. Tag synchronization is performed whenever a **CAD** sheet is opened or a tag is assigned to an exception reporting function code by the GMC application. At times, it may be more convenient to run tag synchronization using a separate utility. Some examples are:

- If changes are made in the global tag database that affect a number of sheets. These sheets could all be updated at once, rather than opening each one through the GMC application just to update tag data.
- If a **CAD** sheet were modified outside its project environment (for example, on another engineering work station) and then copied back into the project, tag data may need to be synchronized.
- If errors are encountered when opening a **CAD** sheet in the GMC, the GMC issues a message and deletes all tag assignments. In this case, it may be better to close the sheet without saving it, then run tag synchronization to find and correct the problems.

The tag synchronization utility can update tag data in the global tag database from the module configurations and update tag data in the module configurations from the global tag database. These operations can be performed on all modules in one or more loops, on all modules in one or more PCUs, or on one or more selected modules. For each exception reporting function code in a **CAD** sheet that has a tag assigned to it, tag synchronization does the following once a link between the

exception reporting function code and a tag in the tag database tables is verified or established:

- Verifies that the tag address, function code type, and sheet name data in the tag data tables match the data in the **CAD** sheet; and if not, updates the tag data tables with the information from the **CAD** sheet.
- Verifies that the tag name, tag description, and internal tag identifier in the **CAD** sheet match the data in the tag data tables; and if not, updates the **CAD** sheet with the information in the tag data tables.

To test synchronization of tag data:

1. Select a loop, PCU, or module from the project tree.
2. From the respective menu, select *Tag Synchronize*. The *Tag Synchronize* dialog box appears.
3. Select *Test Sync*.
4. There may be tags assigned in the **CAD** sheets that are not defined in the TDT database. The effect of adding these tags to the TDT database can be tested. To do so, check the *Add New Tags from CAD File* option. Checking this option does not actually add tags in the *Test Sync* mode.
5. Click on *OK* to proceed.

Warnings and errors are written to the log file **cadsync.log** placed in the module directory. Warnings indicate unsynchronized tag data that can be updated; that is, the tag data in the **CAD** sheet and TDT database do not match and tag synchronization can correct it. Errors indicate a conflict in the data that tag synchronization cannot correct. Warnings can be examined to determine if the updates ought to be made. Errors must be corrected before proceeding with actually performing the synchronization.

To perform synchronization of tag data:

1. Select a loop, PCU, or module from the project tree.
2. From the respective menu, select *Tag Synchronize*. The *Tag Synchronize* dialog box appears.
3. Select *Perform Sync*.
4. There may be tags assigned in the **CAD** sheets that are not defined in the TDT database. These tags can be added to the TDT database by checking *Add New Tags from CAD File*.
5. Click on *OK* to proceed.

PRINTING CAD SHEETS

This section describes how to print multiple **CAD** configuration sheets from the WCAD executive print utility. This option use a **CAD** print queue to output a set of sheets as a background batch process. This printing option is an efficient alternative to manually using the GMC application to open and print each individual sheet.

The print utility uses standard Windows print drivers that provides the ability to render the **CAD** sheets to a wide variety of modern printer and plotter devices that come with these drivers. Printer and plotter support is determined strictly on the ability of the vendor provided device drivers to support the required Windows drawing commands and fonts. Devices that support color can be used to print **CAD** sheets using the standard layer colors, except for automatic swapping of the black background and the white layer colors when printing. Device driver supported scaling provides the ability to print the **CAD** sheets to any size paper option supported by the printer or plotter.

Print Style Setup

When printing **CAD** sheets, the printer to use and the options that affect how the sheets are printed can be pre-defined as a print style. This print style can be selected at print time from a list of known print styles. The pre-defined style options can be used as defined or can be modified for the current print request. This eliminates the need to continually select the options each time a print job is started.

Several standard styles can be defined using different printers, paper sizes, and **CAD** sheet options. These WinTools printer definition files are stored in EWSHOME\CONFIG directory as *stylename.WPF* files. A **CURRENT.WPF** file is automatically created from the user selected style that includes any temporary changes not saved in the style. If no user style has been defined the **CURRENT.WPF** will use the current system default printer and default **CAD** sheet options. If a style references a device driver that has been removed from the system, the current default printer device is used.

NOTE: The system maintains a *current* binary configuration file that contains the last saved or used print style options and Windows print driver settings. **CAD** sheets are always printed using the *current* file. The *current* file is updated from print style options selected in the *Print Style* dialog box at the following times:

1. By selecting a print style from the *Print* dialog box.
2. By choosing *Load Setup* from the *Print Style* dialog box.
3. By choosing *Save Setup* from the *Print Style* dialog box.
4. By clicking on *OK* in the *Print Style* dialog box.

Operating System Issues Each binary printer definition file contains information from the operating system dependent printer device drivers. Device drivers for Windows 3.1 cannot be used on a Windows NT system for example; thus, print styles set up in Windows 3.1 likewise cannot be used in a Windows NT environment. If the engineering work station allows starting multiple operating systems, it is highly recommended that the style names contain a prefix or suffix indicating the operating system for which it is configured. The *Style* description field can contain the detailed definition of the print set up as needed.

Example: P2BC-31 and P2BC-NT can be matching style names indicating: **Printer 2**, **B-size paper**, and **Color** for Windows 3.1 and Windows NT respectively

Driver Issues Testing has shown driver behavior and printing quality can vary greatly from driver to driver, even to the same output device. If printing difficulties are observed, it is recommended a different driver compatible with the output device be tried. Most vendors periodically provide driver updates. Postscript[®] drivers appear to be more reliable than several HP-PCL drivers tested. Some previously supported devices can no longer be supported due to the lack of a suitable Windows driver. Review the release notes for printer device support issues and some observed difficulties with specific drivers.

To configure a print style:

1. Choose *Print Style Setup* from the module tools executive *Module* menu or *Folder* menu.
2. The *Style Setup* dialog box appears as shown in Figure 11-1. The current print style options are displayed. It has three pages. The *Sheet* page appears initially; the other pages can be accessed by clicking on the respective tab.
3. Do any of the following:
 - Make changes to the print style options. Refer to Table 11-1 for a description of the options for each page and select options as needed.
 - To load a predefined style, click on *Load Setup* and select a printer definition file from the dialog box that appears.
 - To save the print style configuration, click on *Save Setup* and enter an eight-character file name in the dialog box that appears.
 - To choose or set up a printer, click on *Printer Setup*. The standard windows print setup dialog box appears. Any installed printer can be chosen and setup using the

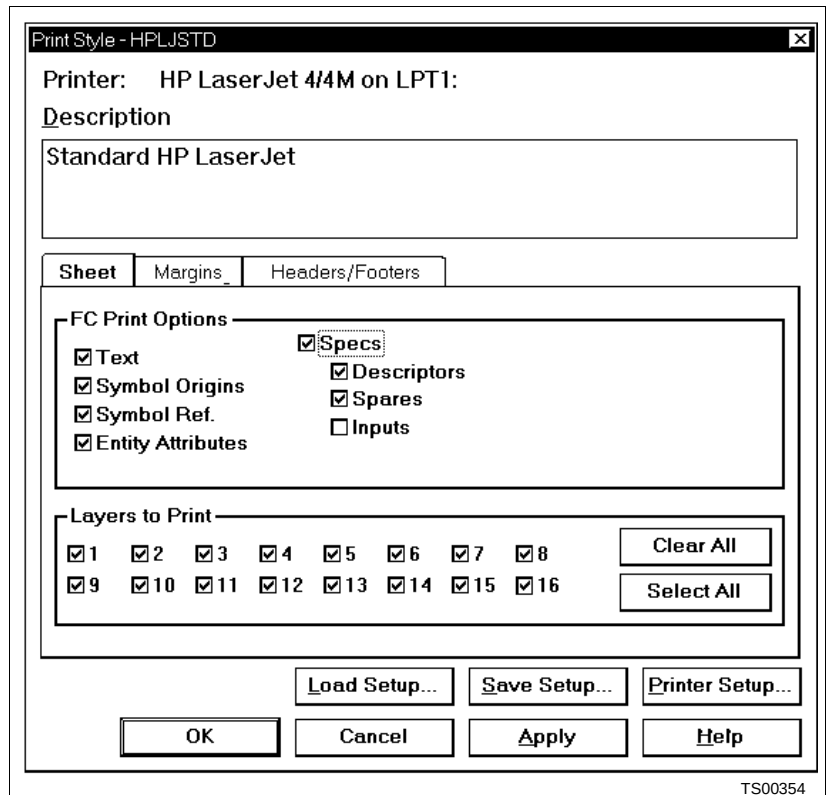


Figure 11-1. Print Style

standard windows printer setup dialog box for that printer. Printer options are saved with the WCAD01 printer definition file and do not affect the standard Windows printer definition file for that printer.

- To accept the current options and continue, click on **OK**. If changes have been made but not saved, then the current options are used for printing, but are not saved to a user printer definition file.

NOTE: Any changes made to the current printer definition file in this way are in effect until a user print style is loaded from the *Print Style* dialog box or selected from the *Print* dialog box. Therefore the print style with these changes can be used for subsequent printing if needed.

Table 11-1. Print Style Pages

Page	Description
Sheet	These options configure what information from the sheet is actually printed.

Table 11-1. Print Style Pages (continued)

Page	Description
Sheet (continued)	
Function code print options	If checked, the following are printed: Text - text strings entered as entities. Symbol Origins - marks the symbol insertion point (+). Symbol Ref. - symbol connection points (x). Entity Attributes - text attributes assigned to an entity. Specs - function code specifications. If specifications are enabled, the following can be enabled: Descriptors - cross reference descriptors. Spares - spare specifications. Inputs - function code default input values.
Layers to print	If checked, the entities on that layer are printed.
Margins	These options determine how the sheet is printed.
Margins	Determine the maximum size of the printed area on a page. Enter the desired margins in units of inches.
Alignment	Position the printed area on the page within the margins.
Auto-select orientation	Overrides the printer's default orientation to print the CAD sheet according to the best fit of its aspect ratio. This should normally be left enabled.
Scaling	A setting of 100 maximizes the drawing within the defined margins. The drawing can be sized by entering a scale factor from 50 to 100. The aspect ratio is maintained.
Header/Footers	Configures the headers and footers printed on a sheet. Headers and footers can be printed on the left or right side or in the center of the page by clicking on the respective option. Type any text to be included in the header or footer in the text box. Codes can be used to automatically insert variable information such as date, time, and file name.

Printing

To print **CAD** sheets:

1. Select the module or documentation folder containing the sheets to be printed from the project tree.
2. Select the sheets to be printed from the list of configuration sheets in the file window.
3. Choose *Print* from the *Module* menu (**Alt M P**) or *Folder* (**Alt O P**) menu.
4. A dialog box appears showing the current print style name and description and target printer. Do any of the following:
 - If the print style shown is to be used, click on *OK*. The print queue manager is launched if not already running and the

sheets are added to the print queue and will be printed using the defined print style.

- To cancel the print request, click on *Cancel*.
- To use a different print style, select it from the *Styles* drop down list box.

NOTE: Even though the *Print* dialog box may show a valid user print style definition, it may have changes in it from using the *Print Style* dialog box. If necessary, to insure that the base user print style is used, re-select it from the *Styles* list box to copy it into the **current** file.

- To make changes or look at the print style, click on *Style Setup*. Refer to **Print Style Setup** for information.

The *Exit Print Queue Manager when Done* box should be left checked.

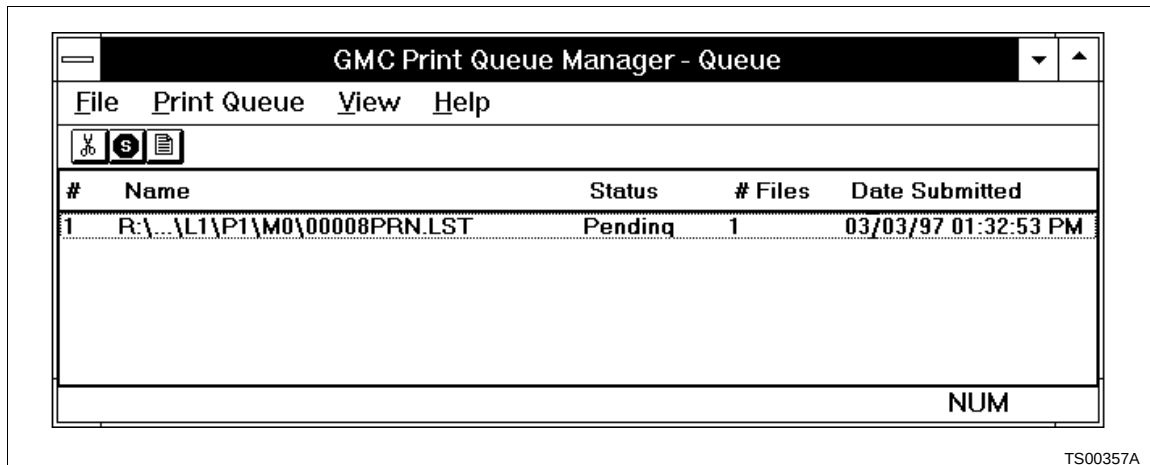
Print Queue Manager

When a sheet is printed, it is sent to the module tools print queue manager. The print queue manager runs in the background to oversee the printing process. It can also be made the current window to inspect the progress of print jobs and manually control printing features. The print queue manager initially appears when printing is started.

The print queue manager application window is shown in Figure 11-2. Each print job is shown along with its job number, status, and date submitted to the print manager. The job tree can be expanded to show individual files.

The following operations can be performed:

Exit Print Manager	Exiting the print manager cancels all queued print jobs. The print manager normally automatically closes when all jobs are printed. If necessary, to exit the print manager, choose <i>Exit</i> from the <i>File</i> menu (Alt F X).
Delete a Job	Jobs can be deleted from the print queue. To delete a job from the print queue, select it (or use multi-select to select a group of jobs) from the print queue display window and choose <i>Delete Job(s)</i> from the <i>Print Queue</i> menu (Alt P D).
Pause and Restart Printing	To pause or restart the printer, choose <i>Pause Printing</i> (Alt P P) or <i>Resume Printing</i> (Alt P R) from the <i>Print Queue</i> menu. If paused, the printer will continue printing any job in progress, then pause.
View the Print Log	A log is maintained containing information about printed jobs. To view the print log, choose <i>View Print Log</i> from the <i>Print Queue</i> menu (Alt P V).



TS00357A

Figure 11-2. Print Queue Manager

- Setup the Print Style** Print styles can be changed from the print manager provided the file (job) has not started printing. To re-define a print style, choose *Print Style Setup* from the *Print Queue* menu ([Alt] [P] [S]) then refer to **Print Style Setup**.
- Display Tool Bar and Status Bar** The tool bar and status bar can be toggled on and off by choosing the respective item from the *View* menu.
- Activate Help** To activate print queue help, choose *Contents* from the *Help* menu ([Alt] [H] [C]) or press [F1].

VIEWING LOG FILES

A log file viewer application is included with the module configuration utilities. It displays and prints any of the log files generated by the utilities. To use the log file viewer:

1. Choose *View Log* from the module tools executive *Utilities* menu ([Alt] [T] [V]). The log viewer application window appears. The application window is similar to other Windows applications. It has two menus:

- File** Contains menu items to open and print log files and to exit the application.
- Help** Contains one item to display information about the application.

2. Choose *Open* from the *File* menu. A standard file open dialog box appears. Choose the file of interest from this dialog box.

To print an opened file, choose *Print* from the *File* menu.

To quit the application, choose *Exit* from the *File* menu.

CONFIGURING WINTOOLS COMMUNICATIONS

The module tools support two types of communication, referred to as INFI 90 OPEN communications and ICI communications. ICI communication supports the semAPI library functions; INFI 90 OPEN communication does not. Refer to Table 11-2 for communications support information.

Table 11-2. Communication Support

Type	Operating System	Connection/NIU
INFI 90 OPEN communications	Windows 3.1, 3.11, and Windows for Workgroups 3.11	CIC01
	Windows 95	INICI01 serial port only
	Windows NT 3.51	INICI03 serial port only
		IMSPM01
		IMCPM02
ICI communications	Windows 95	INICI03 serial port (local)
	Windows NT 3.51	INICI03 SCSI port (local)
		OSM (network)

Setup and operation of INFI 90 OPEN communication is straightforward. Refer to **Application Programming Interface (semAPI) NT** for information on ICI communication. The communication settings are also used for the module tools communications.

WinTools communication supports the semAPI's 20 logical communication paths, designated numerically from 1 to 20. Logical communication path 0 is reserved for I90 communications and defaults to:

Port - COM1.

Baud - 9600 baud.

Handshaking - hardware.

These settings can be changed if needed and other logical communication paths configured. Each of the configured logical communication paths are saved and can be selected whenever needed.

Selecting a Communication Path

To select a logical communication path:

1. Choose *Communications Setup* from the executive *Utilities* menu (**Alt T C**) or CMU *Utilities* menu. The communications setup dialog box appears.

2. Choose a logical communication path from the drop down list box. If semAPI is not installed, only the INFI 90 OPEN communications path (0) is available.
3. Click on *OK* to use the selection and store it as the current communication path.

Configuring INFI 90 Communications

To configure an INFI 90 OPEN communication path:

1. Choose *Communications Setup* from the executive *Utilities* menu (**[Alt][T][C]**) or CMU *Utilities* menu. The communications setup dialog box appears.
2. In the *INFI 90 Communications* box, click on *Configure*. The *Configure I90 Communications* dialog box appears.
3. Configure the communications port, baud rate, and hand-shaking as needed for use with the communications unit. A 32-character description can be given to the logical path.
4. Click on *OK* to save the information and return to the previous dialog box.

Configuring ICI Communications

To configure an ICI communication path (if semAPI is installed):

1. Choose *Communications Setup* from the executive *Utilities* menu (**[Alt][T][C]**) or CMU *Utilities* menu. The communications setup dialog box appears.
2. In the *ICI Communications* box, click on *Configure*. The *ICI Configuration* dialog box appears.
3. Refer to the ***Application Programming Interface (semAPI) NT*** for Windows for configuration information.

DSOE CONFIGURATION

The distributed sequence of events (DSOE) product provides one-millisecond resolution to sequence of events related changes of state of up to 5,000 digital input points.

The DSOE utility is used to create and maintain the configuration for the INSEM01 module. Normally module configurations are configured using the GMC application. In the SEM module case, additional MFP module configuration data needed to configure the SEM module and complex trigger definitions (trigger events based on multiple digital signals) make this alternate configuration arrangement necessary.

Refer to **DSOE Configuration Overview** for a summary of the configuration process. Following the overview each heading details a menu selection of the DSOE utility.

NOTE: DSOE configuration is a DOS utility launched from the module tools executive.

DSOE Configuration Overview

The following is a sequential overview of the DSOE configuration process.

1. Create a standard module header (**MHD**) file for each MFP module that will be processing the IMSED01 Distributed Sequence of Events Digital Module.
2. Create configuration drawings (**CAD** files) using the GMC application for each MFP module. Use the standard external drawing **0000152A.CAD** (located in C:\EWS directory) as a template for each SED module being terminated.
3. Cross reference and compile the MFP module configurations.
4. Download the compiled (**CFG** files) configurations to the actual modules.
5. Create a module header (**MHD**) file for the SEM module.
6. Configure the executive block of the SEM module. Refer to **Configure SEM Executive Block** for information.
7. Create the SOE point list. Refer to **Create Point List From CAD Files** for details.
8. Create the SOE trigger list file for the SEM module. Refer to **Configure Complex Trigger List** for information.
9. Create the module configuration (**CFG**) file for the SEM module. Refer to **Compile SEM Module Configuration** for details.
10. Download the SEM module configuration (**CFG**) files.

Configure External Drawings

The external drawing **0000152A.CAD** uses function codes 241 and 242 with the addition of SOE channel definition attributes (SOECHANDEF) attached to function code 242 to define the additional information required to configure each digital input processed by the INSEM01 module.

Function code 241 is required in only the first external drawing (**0000152A.CAD**) processed by the MFP module. The remaining external drawing should replace function code 241 with an output reference to the previous digital I/O modules function code 242 specification S2.

Function code 242 has 16 predefined SOECHANDEF attributes used to define channel processing by the SEM module.

Display and modify the attributes as necessary using *Modify Attributes* from the GMC *Functions* menu.

The SOECHANDEF attributes are displayed on **CAD** drawings as follows:

Attribute Type, Occurrence (or Channel) Number, Value

Value consists of information defining tag name (up to 14 characters), simple trigger, recordable event, zero state alarm, one state alarm, and configured in system for each digital I/O input.

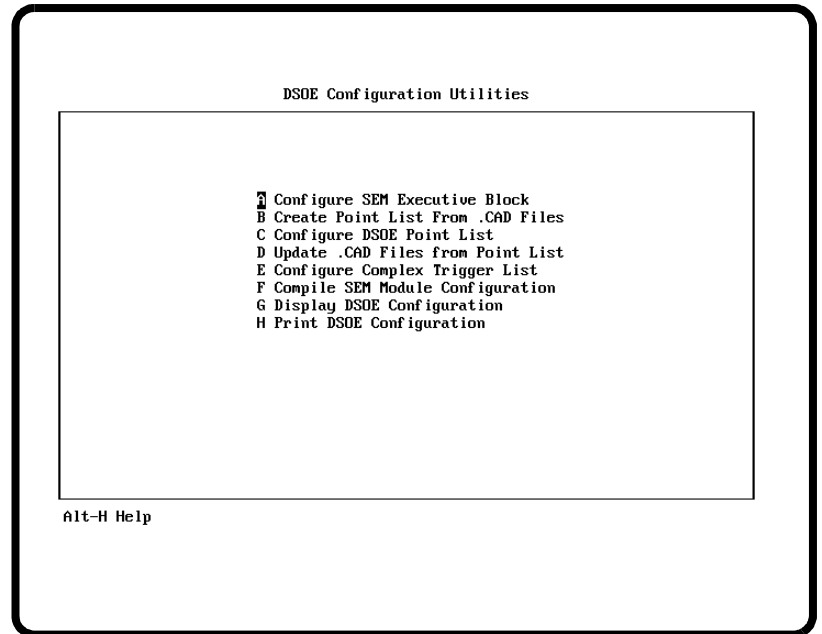
Configure SEM Executive Block

This function is used to edit the SEM executive block (function code 243) for a specific INSEM01 module. The data of function code 243 is stored in an **xxxxx.XEC** file.

NOTE: A module header (**xxxxx.MHD**) file must be defined for the INSEM01 module before configuring the executive block.

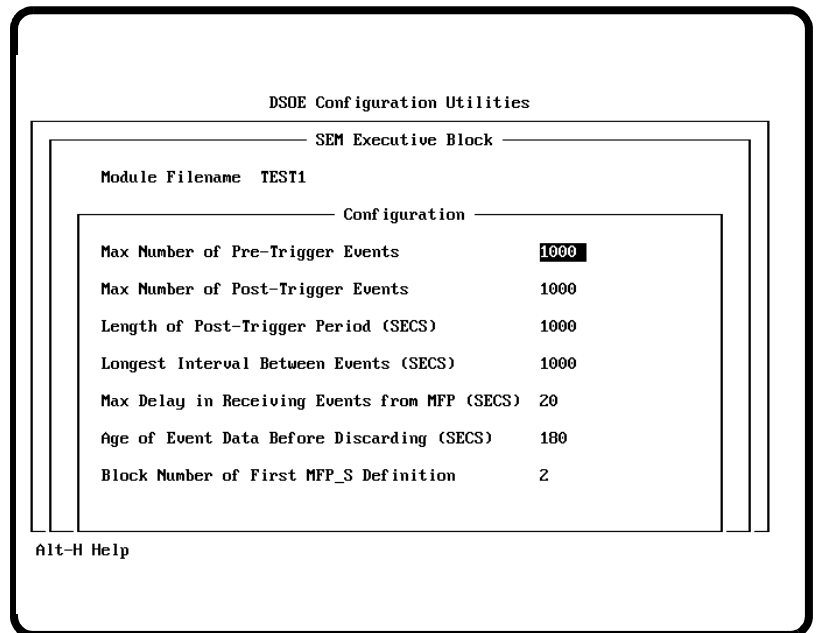
To configure an SEM executive block:

1. From the module tools executive *Utilities* menu, choose *DSOE Configuration* (**Alt T D**).
2. The DSOE configuration utilities menu shown in Figure 11-3 appears. Select *Configure SEM Executive Block*.
3. Enter the **MHD** file name of the SEM module into the *Module Filename* field and press **Enter**.
4. Executive block configuration fields appear as shown in Figure 11-4. Enter the appropriate information into the fields and press **Enter**. Refer to Table 11-3 for field descriptions.
5. After completing the input, press **F10** to save the changes.



TPS1093A

Figure 11-3. DSOE Configuration Utilities Menu



TPS1094A

Figure 11-4. Sample SEM Executive Block Configuration

Table 11-3. SEM Executive Block Fields

Field	Options	Description
Max Number of Pre-Trigger Events	0 to 5,000	Maximum number of events recorded previous to the trigger event.
Max Number of Post-Trigger Events	0 to 5,000	Maximum number of events recorded after the trigger event.
Length of Post-Trigger Period (SECS)	0 to 86,000	Maximum period in which events are recorded after the trigger event.
Longest Interval Between Events (SECS)	0 to 86,000	Maximum time interval between two consecutive events.
Max Delay in Receiving Events from MFP (SECS)	0 to 60	Maximum delay in receiving data from the MFP module.
Age of Event Data Before Discarding (SECS)	0 to 32,699	Maximum time data can be stored in the internal buffer of the SEM module while waiting for a request from the console.
Block Number of First MFP_S Definition	1 to 1998	Block number of the first FC 244.

Create Point List From CAD Files

This function creates a DSOE point list (**xxxxx.SOE**) from the **CAD** files that comprise the user-selected MFP modules. This function extracts the MFP - digital I/O module and channel information from function codes 241 and 242, and adds the information to the DSOE point list which represents the information required for function codes 244 and 245. The results of this function are written to a log file called **CADTOSOE.LOG**.

The following procedure details creating a point list from **CAD** files:

1. From the *DSOE Configuration Utilities* menu select *Create Point List From .CAD Files*.
2. A selection screen displays **CFG** files of the current directory. Select the appropriate MFP configuration files by pressing **[Enter]**. After selecting the appropriate files, press **[F10]**.
3. Enter the name of the DSOE point list and press **[Enter]**. Generally, it is good practice to name the point list file the same as the module header file of the SEM module.

Example: If the SEM module header is **TEST1.MHD**, then name the point list file **TEST1.SOE**.

4. After the SOE point list is created, a log file (**CADTOSOE.LOG**) appears that describes the status of the operation.

5. Press **F10** to escape to the DSOE configuration utilities menu.

Configure DSOE Point List

This function is used to view and edit a DSOE point list (**xxxxx.SOE**). Refer to **Create Point List From CAD Files** for details about creating a point list. This function modifies the SOECHANDEF attribute information for individual DSOE points without having to open the **CAD** file using the CAD environment.

This function does not allow the INFI 90 OPEN address, the I/O module address, or the **CAD** sheet name to be modified.

To configure a DSOE point list:

1. From the *DSOE Configuration Utilities* menu select *Configure DSOE Point List*.
2. Type the SOE point list name and press **Enter**. A screen appears listing the addresses of the loop, PCU, module, I/O module and sheet name of a selected configuration drawing.
3. Press **Enter** to list the SOECHANDEF attribute information for the selected address (Figure 11-5 shows a sample screen). Refer to Table 11-4 for a description of the attributes.
4. Do any of the following:
 - Use **Page Up** and **Page Down** to scroll through the other selected I/O module configurations.
 - Press **Esc** and enter the loop, PCU, module, and I/O module addresses of a particular configuration drawing and press **Enter**.
 - Enter the tag name of a configured channel and press **Enter**.

NOTE: To advance through address fields press **Tab**.

5. Edit the DSOE attributes as desired. Table 11-5 describes cursor movements inside the screen.
6. When editing is complete, press **F10** to save the changes and **Esc** to exit the attributes screen.

NOTE: Perform *Update CAD Files from Point List* to automatically update the attribute values in the affected **CAD** sheets.

Configure DSOE Point List - test1.SOE						
Loop 1	PCU 1	Module 1	Slave Address 0	Sheet 1010100A		
Channel	Tagname	Simple Trigger	Recordable Event	0 State Alarm	1 State Alarm	Configured In System
0	DAVE	NONE	1T00	Y	N	Y
1	TO BOILER	0T01	0T01	N	Y	Y
2	FROM BOILER	0T01	0T01	N	Y	N
3	ALARM LEVEL	0T01	0T01	N	Y	Y
4	POWER	0T01	0T01	N	Y	Y
5	TO VALVE	0T01	0T01	N	Y	Y
6	FROM VALVE	0T01	0T01	N	Y	Y
7	SOE-NAME-008	0T01	0T01	N	Y	N
8	SOE-NAME-009	NONE	NONE	N	N	N
9	SOE-NAME-010	NONE	NONE	N	N	N
10	SOE-NAME-011	NONE	NONE	N	N	N
11	SOE-NAME-012	NONE	NONE	N	N	N
12	SOE-NAME-013	NONE	NONE	N	N	N
13	SOE-NAME-014	NONE	NONE	N	N	N
14	SOE-NAME-015	NONE	NONE	N	N	N
15	SOE-NAME-016	NONE	NONE	N	N	N

TPS1095A

Figure 11-5. Sample Configure DSOE Point List

Table 11-4. SOECHANDEF Attributes

Field	Options	Description
Channel	0 to 15	Input channel number on the I/O module.
Tagname	Up to 14 alphanumeric characters	Unique tag name of the input channel.
Simple Trigger ¹	None, 0 to 1 1 to 0, Both	Defines which digital input state changes will trigger an SOE event in the SEM module.
Recordable event ¹	None, 0 to 1, 1 to 0, Both	Defines which digital state changes will be reported to the SOE event log on the console.
Zero (0) state alarm	Y or N	Alarmed if SOE tag is zero (0).
One state alarm	Y or N	Alarmed if SOE tag is one (1).
Configured in System	Y or N	Indicates whether the digital input channel will also be processed by the SEM module as an SOE point. If no (N), the digital channel is only seen by the INFI 90 Open system as a standard digital input.

NOTE:

1. Choice help available.

Table 11-5. Cursor Movements

Key	Moves Cursor
 , 	Inside an individual field
 , 	Through fields in a column
	Through fields in a row
	Top of list
	Bottom of list

Update CAD Files from Point List

This function updates the SOECHANDEF attribute information in the **CAD** files from a DSOE point list (**xxxxx.SOE**). This function reads the SOECHANDEF attributes of function code 242 from the **CAD** files corresponding to the selected MFP modules and searches for the corresponding records in the DSOE point list by address (loop, PCU, module, I/O module address and channel). If the point exists in the list and attribute information has changed, then the SOECHANDEF attribute in the **CAD** file is updated. The results of this function is written to a log file called **SOETOCAD.LOG**.

To update **CAD** files from a DSOE point list:

1. From the *DSOE Configuration Utilities* menu select *Update .CAD Files from Point List*.
2. A selection screen displays **CFG** files of the current directory. Select the appropriate MFP configuration files by pressing **Enter**. After selecting the desired files, press **F10**.
3. Enter the name of the DSOE point list name and press **Enter**.
4. After updating the selected files, a log (**SOETOCAD.LOG**) describes the status of the update. Press **F10** to exit to the DSOE configuration utilities menu.

Configure Complex Trigger List

This function configures DSOE complex trigger definitions. A complex trigger is a condition that is based upon the status of multiple digital signals.

Complex trigger definitions are stored in an SOE trigger list file (**xxxxxTRG.SOE**). Operands in the various trigger expressions must be points defined in the corresponding DSOE point list (**xxxxx.SOE**). Up to 256 complex triggers can be configured in an SEM module. Each trigger can have up to 16 variables.

To configure a complex trigger definition:

1. From the *DSOE Configuration Utilities* menu select *Configure Complex Trigger List*.
2. Enter the first five characters of the trigger file name and press **Tab**.

NOTE: The name of the trigger file (**xxxxxTRG.SOE**) must match the **MHD** file name of the SEM module header. If these names do not match, an error occurs making it impossible to create a complex trigger file.

3. Enter the trigger number (zero to 256) and press **Enter**. The screen shown in Figure 11-6 appears.

Var	SOE-Tagname	ID	Ring	PCU	Mod	Slave	Channel	Configured
A	DQUE	1	1	1	1	0	0	YES
B	TO BOILER	2	1	1	1	0	1	YES
C	FROM BOILER	3	1	1	1	0	2	NO
D								
E								
F								
G								
H								
I								
J								
K								
L								
M								
N								
O								
P								

Infix Expression (((A & ~B) | C) ^ A)
 Postfix Expression AB~&Cif^

Alt-H Help

TPS1096A

Figure 11-6. Sample Complex Trigger Definition

4. Enter the tag name of the A variable into the *SOE-Tagname* field associated with the desired tag names and press **Enter**. Input up to 16 variables.
5. Scroll down to the *Infix Expression* or the *Postfix Expression* fields using **↓**.
6. Input an Infix or Postfix boolean expression using operations symbols and the letters appearing in the *Var* field associated with the desired tag names and press **F10**. Refer to Table 11-6 for expression symbols.

NOTE: Input either expression type. The other expression type is automatically generated, so that both Infix and Postfix are displayed.

Table 11-6. Boolean Trigger Operands

Postfix Symbols	Infix Symbols	Operation
~	~	NOT
&	&	AND
		OR
^	^	XOR
	()	Parentheses

Compile SEM Module Configuration

This function is used to create a module configuration (**xxxxx.CFG**) file for a specific INSEM01 module. The **CFG** file created is the same format as the standard **CFG** file by the compile function. The following DSOE configuration files must exist before a **CFG** file can be created:

- Module header (**xxxxx.MHD**).
- DSOE executive block (**xxxxx.XEC**).
- DSOE point list (**xxxxx.SOE**).

NOTE: A trigger list file (**xxxxxTRG.SOE**) is not required, however, if one exists it will be included.

To compile an SEM module configuration:

1. From the DSOE configuration utilities menu select *Compile SEM Module Configuration*.
2. Enter the SEM module configuration file name and press **ENTER**. When compilation is complete the **xxxxx.LOG** displays describing the status of the operation.

NOTE: Make sure the SEM executive block has been configured. Refer to **Configure SEM Executive Block** for information.

3. After compiling the SEM module (thereby creating a **CFG** file), download the configuration to the module by using the CMU application.

Display DSOE Configuration

This function displays configuration data from specific DSOE point (**xxxxx.SOE**) and trigger (**xxxxxTRG.SOE**) list files. To display the configuration data from the **xxxxx.CFG** file, use the CMU application.

DISPLAY CONFIGURED POINTS, RECORDABLE POINTS, ALARM POINTS AND SIMPLE TRIGGERS

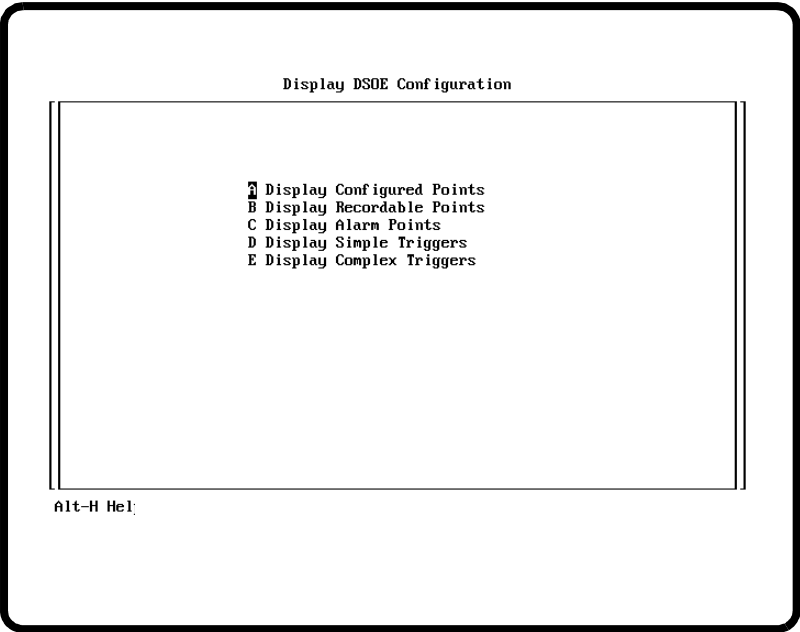
The following procedure details the following menu items:

Display Configured Points
Display Recordable Points
Display Alarm Points
Display Simple Triggers

The display functions are sub-sets of the entire SOE point list based on the selection (i.e., if *Display Alarm Points* is selected, only the SOE points that are alarmed will display).

To display these items:

- 1. From the *DSOE Configuration Utilities* menu select *Display DSOE Configuration*.
- 2. Select an item from the menu shown in Figure 11-7. Table 11-7 lists and describes the fields that display.
- 3. When the information is displayed the entire screen is not visible. Table 11-8 describes how to maneuver in this screen.



TPS1097A

Figure 11-7. Display DSOE Configuration Menu

Table 11-7. Display DSOE Configuration Menu Selections

Menu Item	Field	Description
Display Configured Points	Tagname	Unique tag name of the input channel.
Display Recordable Points	SOE Index	Index number used by the SEM module and function codes. They are automatically assigned by the configuration tools to each digital input channel.
Display Alarm Points	Loop Address	Address of loop.
Display Simple Triggers	PCU Address	Address of PCU.
	Module Address	Address of the module.
	I/O Module Address	Address of I/O module.
	Channel Address	Input channel number on the I/O module.
	Trigger Status	Defines which digital input state changes will trigger an SOE event in the SEM module.
	Recordable Status	Defines which digital state changes will be reported to the SOE event log on the console.

Table 11-7. Display DSOE Configuration Menu Selections (continued)

Menu Item	Field	Description
Display Configured Points	Alarm Zero	Alarmed if SOE tag is zero (0).
Display Recordable Points	Alarm One	Alarmed if SOE tag is one (1).
Display Alarm Points	Configured Status	Indicates whether the digital input channel will also be processed by the SEM module as an SOE point. If no (N), the digital channel is only seen by the INFI 90 Open system as a standard digital input.
Display Simple Triggers (continued)		
	CAD File	CAD drawing file of the MFP module where the digital input channels of the I/O modules are.

Table 11-8. Screen Movements

Key	Moves Screen
F1	Left (by half screen)
F2	Right (by half screen)
F3	Far left margin
F4	Far right margin
Page Down, Page Up	Down, up (one screen)
↑, ↓	Up, down (by row)
←, →	Right, left (by column)

DISPLAY COMPLEX TRIGGERS

To display complex triggers:

1. From the *DSOE Configuration Utilities* menu select *Display DSOE Configurations*.
2. Select *Display Complex Triggers*. Table 11-9 lists and describes the fields that display.

NOTE: The display functions are for display only. To edit configuration, refer to **Configure DSOE Point List**.

Table 11-9. Display Complex Triggers
Field Descriptions

Field	Description
Variable	Variable of complex trigger (up to 16 variables).
Tagname	Tag name of SOE point.
SOE Index	SOE index.
Loop Address	Address of loop.
PCU Address	Address of PCU.
Module Address	Address of the module.
I/O module Address	Address of I/O module.
Channel Address	Digital address of I/O module.

Table 11-9. Display Complex Triggers
Field Descriptions (continued)

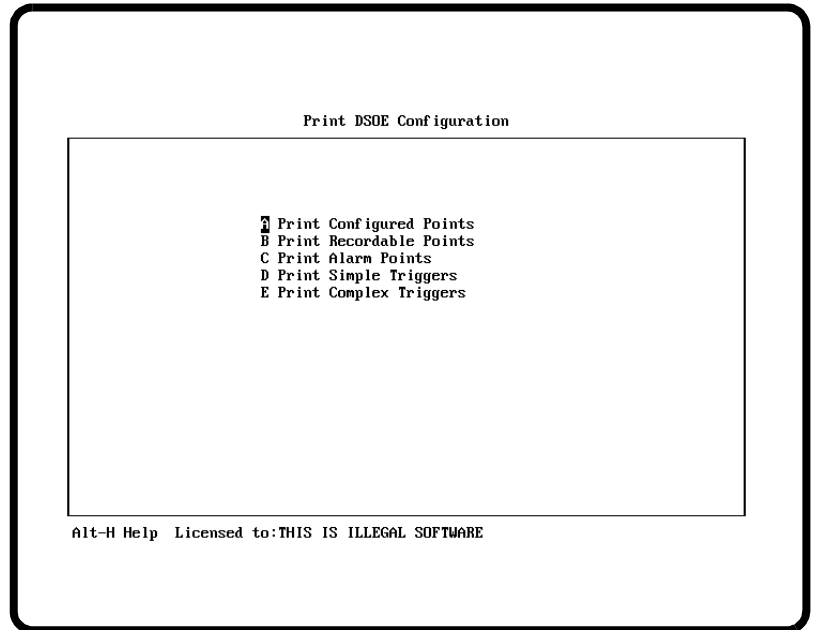
Field	Description
Trigger Status	Defines which digital input state changes will trigger an SOE event in the SEM module.
Recordable Status	Defines which digital state changes will be reported to the SOE event log on the console.
Alarm Zero	Alarmed if SOE tag is zero (0).
Alarm One	Alarmed if SOE tag is one (1).
Configured Status	Indicates whether the digital input channel will also be processed by the SEM module as an SOE point. If no (N), the digital channel is only seen by the INFI 90 Open system as a standard digital input.
CAD File	Name of the module configuration file (CFG).
Infix Expression	Infix expression of the complex trigger.
Postfix Expression	Postfix expression of the complex trigger.
Spec	Specification number of SOE tag name being used in the complex trigger (i.e., S1).
Value	Specification value of SOE tag being used in the complex trigger (i.e., -1).
Operand/Operator	Digital input channel or a boolean logic operator (i.e., AND, OR, NOT, etc.) used to define the complex trigger expression.

Print DSOE Configuration

This function prints configuration data from specific DSOE point and trigger list (**xxxxx.SOE** and **xxxxxTRG.SOE**) files.

To print DSOE configurations:

1. From the *DSOE Configuration Utilities* menu select *Print DSOE Configuration*.
2. Enter the module file name of the SEM **MHD** file and press **Enter**. The menu shown in Figure 11-8 appears. The same information shown in the *Display DSOE Configuration* menu selections will print.



TPS1098A

Figure 11-8. Print DSOE Configuration Screen

SECTION 12 - DOS UTILITIES

INTRODUCTION

This section documents supplemental DOS utilities that must be run from the DOS prompt:

- Password security.
- Upgrading 4.x files.

PASSWORD SECURITY

Password utility protects the GMC (graphical module configurator) and PW (password) utilities from unauthorized users. When these utilities are launched, a prompt appears requesting the password. If the password is not entered correctly after three tries, the application closes.

The password software is part of the engineering work station software. It is loaded when the engineering work station software is loaded. After specifying the passwords, the password utility should be deleted from the C:\EWS directory. This safeguards password security from unauthorized personnel. If passwords need to be changed, the password utility can be restored from the original release floppy disks. If more convenient password access is preferred, copy the password utility to another floppy disk. In any case, be sure that the floppy disks are stored securely.

Password settings are retained through subsequent releases. The password protection is maintained even if the current protected program is replaced by an unprotected copy.

To define passwords:

1. From the DOS prompt, type:

PW *utility*.EXE

where:

utility Program to protect: **GMC.EXE** or **PW.EXE**.

The following prompt appears:

*Enter new password (maximum 10 characters, Esc to quit,
ENTER for none)*

If the given utility has been previously protected, the prompt asks for the current password before prompting for the new password.

2. Input a password of up to ten characters. As the password is entered, periods (.) appear in place of the keystrokes. When the password is completed press **Enter**. The following prompt appears:

Retype password:

3. Retype the password. This is done so that an errant key-stroke is not saved in the password. If the passwords do not match the program will prompt for the password again.

After successfully protecting a utility, the next time the protected utility is executed a password entry prompt will appear before the utility is executed. The correct password must be entered before continuing.

NOTE: A forgotten password cannot be retrieved. In such a case, remove password protection and then define a new password.

To remove password protection:

1. At the DOS prompt, type

PW *utility*.EXE **Enter**

where:

utility Program to protect: **GMC.EXE** or **PW.EXE**.

The following prompt appears:

*Enter new password (maximum 10 characters, Esc to quit,
ENTER for none)*

2. To remove password protection, press **Enter**. The password prompt will not appear the next time the utility is started.

UPGRADING 4.x FILES

SCAD 5.2 and later no longer separate configuration sheets into two files (**DBF** and **SPC**). The two files are combined into a single **CAD** file for each sheet of a configuration drawing. Because of this, existing **JHD**, **MHD**, and **LBR** files from earlier versions must also be upgraded.

NOTE: After upgrading files, the old version of the files are *not* preserved.

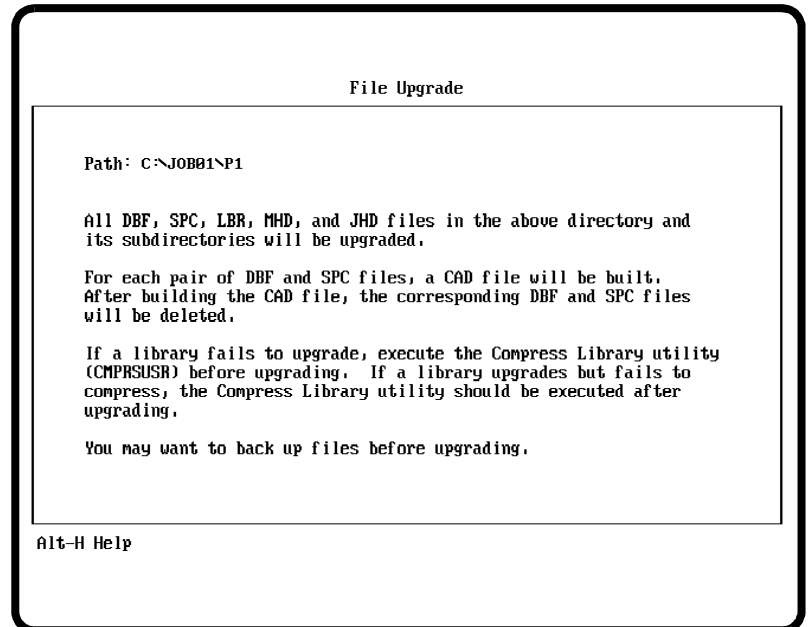
To upgrade SCAD 4.x files:

1. From the DOS prompt, type;

UPGRADE Enter

The screen shown in Figure 12-1 appears.

2. Read the text that appears.



TPS0320B

Figure 12-1. Upgrade Screen

3. Input the desired path and press F10. The working directory is the default directory shown in the *Path* field. The following prompt appears:

Upgrade files in the directory and its subdirectories (Y/N)?

4. Answer the prompt appropriately.
 - Type **N** and the cursor returns to the *Path* field. Another directory can be selected for upgrading.
 - Type **Y** and the **JHD** (job header definition), **MHD** (module header definition), and **LBR** (library) files located in the specified path are upgraded. As the files are upgraded, the file name appears along with the status of the upgrade. The upgrade information is stored to a log file called **UPGRADE.LOG**. This file can be listed using DOS commands.

After a file has been upgraded it is no longer compatible with earlier versions of SCAD. If a **CAD** file contains user shapes that are not associated with a user library, the shapes will be deleted. A list of these shapes is included in the log file. Library compression is required for **LBR** files that contain shapes or macros marked for deletion (refer to **Compressing a Library** in Section 4).

NOTE: If excessive difficulties arise when attempting to upgrade files, CTRL386 may be the source of the problems. If CTRL386 is active on the engineering work station, reboot (power off/on) the work station. Run SETUP386 and set AutoBoot to N and save the configuration. Reboot the work station and answer **N** to the prompt: *Start Contrl/386 (Y/N)*. Rerun the upgrade.

If errors occur during the upgrade function, the error messages appear on screen and are also written to the **UPGRADE.LOG** file. Table 12-1 lists the error messages along with a corrective action.

Table 12-1. Upgrade Error Messages

Message	Corrective Action
Not a valid CAD/TXT file Skipping	The file size is wrong, a CAD file does not exist, or an LBR needs to be compressed.
Full path name of the current directory too long	Update the current directory subdirectories one at a time.
Bad module type specified in file	The MHD file to be upgraded contains a bad module type. After upgrading the file, correct this field in the MHD file using <i>Module Information</i> from the module tools executive <i>Module</i> menu.
Not enough disk space to upgrade or error while copying file.	Free some disk space on the volume which contains the files to be upgraded and try again.
Corrupt - Skipping	The file upgraded is corrupt and unrecoverable.
Invalid entries deleted	The file contains entities that are not considered valid.
User shapes deleted	The CAD file contains user shapes that are not associated with a user library.
Path not found	The selectable path name does not exist. Run the program after determining the correct path
Error writing file	An I/O error has occurred while processing the file. Retrieve the file from your backup and try again.
File already upgraded	File has been upgraded.
Error reading file	An I/O error has occurred while processing the file. Retrieve the file from your backup and try again.
Can't initialize GES	Contact technical support.
Memory allocation failure	Contact technical support.
Error opening file	An I/O error has occurred while processing the file. Retrieve the file from your backup and try again.
File empty, skipping to next file	File truncated (may be corrupt). The file contains invalid entities. It has been truncated.

Table 12-1. Upgrade Error Messages (continued)

Message	Corrective Action
Cannot upgrade C:\EWS files in user mode	The program can not upgrade files located in the C:\EWS directory if the right password is not found supplied when UPGRADE is invoked.
Can't find/can't load system library C:\EWS\SHAPE.LBR	The system library does not exist in the C:\EWS directory. Install the module configuration utilities software.

APPENDIX A - FILE NAME EXTENSIONS

EXTENSIONS

Table A-1 lists and describes the various files and file types that are used in the module tools. If the full file name is not given, the description refers to all files having the given extension.

Table A-1. File Descriptions

File	Description
BND	Created by the compiler for the binder.
CAD	These are the drawing files created by GMC software.
CFG	Configuration files. Compiling the GMC drawing files or saving the configuration stored in a module using the CMU application creates this file.
JHD	Job header file. Stored on disk in the job directory.
LBR	Library files. Stored in the job level directory.
LOG	Log file created by the compiler for the binder.
MHD	Module header file. Stored on disk in the project directory.
NBS	BASIC/C and Batch 90 configuration files.
OLD	Generated if compress fails. Stored on disk in the current directory.
OUT	Contains cross reference documentation information.
PRJ	Project file.
README.DWG	List of standard drawings.
README.REV	List of modules and latest revisions.
REF	Created by the compiler for the binder.
SLV	SLV utility script command file.
VFY	SLV utility verification file. Contains any verification errors detected during this function.
WGP	WinTools group definition file.
WPF	WinTools CAD printer definition file.
XRF	Contains all messages generated by cross reference. Stored on disk in the current directory.

APPENDIX B - SYSTEM LIBRARY

INTRODUCTION

The module configuration utilities provide a system library that consists of a group of standard entities for use by the graphical module configuration application. This appendix lists and describes the entities in the system library.

SYSTEM SYMBOLS

Table B-1 lists and describes each entity in the system library. Figures B-1 through B-4 show the system library entities.

Table B-1. System Library

Name	Description
Automatic Borders	
AHBORD	Horizontal border with Bailey logo and loop, PCU, and module.
AHBORDG	Horizontal border without Bailey logo, but with loop, PCU, and module.
AHGBORD	Horizontal border with Bailey logo, but without loop, PCU, and module.
AHGBORDG	Horizontal border without Bailey logo and loop, PCU, and module.
AVBORD	Vertical border without Bailey logo, but with loop, PCU, and module.
AVBORDG	Vertical border with Bailey logo and loop, PCU, and module.
AVGBORD	Vertical border with Bailey logo, but without loop, PCU, and module.
AVGBORDG	Vertical border without Bailey logo and loop, PCU, and module.
DBORDHA	Obsolete border.
DBORDHAG	Obsolete border.
DBORDVA	Obsolete border.
DBORDVAG	Obsolete border.
Connection Point	
N90CNECT	Signal connection point.
Constant Blocks	
B0	Boolean Zero (Fixed Block 0).
B1	Boolean one (fixed block 1).
R-1	Real negative one (fixed block 4).
R-100	Real negative one-hundred (fixed block 3).
R0	Real zero (fixed block 5).

Table B-1. System Library (continued)

Name	Description
R1	Real one (fixed block 6).
R100	Real one-hundred (fixed block 7).
RMAX	Maximum real number allowed - +9e18 (fixed block 9).
RMIN	Minimum real number allowed - -9e18 (fixed block 8)
SETBLK	Modifiable block number for IMLMM02 and NMLMM02 modules only.
ZERO	Integer or real zero (fixed block 2).
Do Not Use The Following Symbols (Errors May Result)	
PR0	Component shape contained in function code shapes.
PR1	Component shape contained in function code shapes.
PR2	Component shape contained in function code shapes.
PR3	Component shape contained in function code shapes.
PR4	Component shape contained in function code shapes.
PR8BUB	Component shape contained in function code shapes.
PRBUB	Component shape contained in function code shapes.
PRMA	Component shape contained in function code shapes.
System Shapes	
BDIN	Boolean control input - for logic drawings.
BDIND	Boolean operator indication - for logic drawings.
BDOUT	Boolean control output - for logic drawings.
BORDHD	Obsolete border.
BORDVD	Obsolete border.
BPB	Boolean push button input - for logic drawings.
DBORD	Obsolete border.
DBORD1	Obsolete border.
DBORDH	Obsolete border.
DBORDV	Obsolete border.
FU	Fuse.
GND	Ground - used for wiring drawings.
PRA	Forward (right) facing arrow head.
PRAI2	Shape used for standard external connection drawings.
PRAR	Reverse (left) facing arrow head.
PRBRF	Bracket open to left (forward facing).
PRBRR	Bracket open to right (reverse facing).
PRCF	Cable designation - forward facing.
PRCLC2	Shape used for standard external connection drawings.
PRCLC3	Shape used for standard external connection drawings.
PRCR	Cable designation - reverse facing.
PRD	Small dot.
PRD4	Large dot.
PRIF	Shape used for standard external connection drawings.

Table B-1. System Library (continued)

Name	Description
PRIF1	Shape used for standard external connection drawings.
PRIF2	Shape used for standard external connection drawings.
PRIF3	Shape used for standard external connection drawings.
PRIF4	Shape used for standard external connection drawings.
PRIF5	Shape used for standard external connection drawings.
PRIOF	Shape used for standard external connection drawings.
PRIOR	Shape used for standard external connection drawings.
PRIR1	Shape used for standard external connection drawings.
PRIR10	Shape used for standard external connection drawings.
PRIR2	Shape used for standard external connection drawings.
PRIR3	Shape used for standard external connection drawings.
PRIR4	Shape used for standard external connection drawings.
PRIR5	Shape used for standard external connection drawings.
PRIR6	Shape used for standard external connection drawings.
PRIR7	Shape used for standard external connection drawings.
PRIR8	Shape used for standard external connection drawings.
PRIR9	Shape used for standard external connection drawings.
PRISB	Isolated system ground bus - for wiring drawings.
PRNTD	Shape used for standard external connection drawings.
PRNTE	Shape used for standard external connection drawings.
PRNTS	Notes used on standard external connection drawings.
PROF1	Shape used for standard external connection drawings.
PROF2	Shape used for standard external connection drawings.
PROR	Shape used for standard external connection drawings.
PRSLOC	Slave location box for external connection drawings.
PRSSO	Shape used for standard external connection drawings.
PRT5	Wire connection for wiring drawings - for wiring drawings.
PRTB	Terminal block wire connection - for wiring drawings.
PRTLOC	Termination location box for external connection drawings.
PRTW	Twist with shape - component of shape TW.
PRWC	Wire connection shape - for wiring drawings.
PRWCS	Small wire connection shape - for wiring drawings.
PRWL	Wire to cable connection (wire on left) - for wiring drawings.
PRWR	Cable to wire connection (wire on right) - for wiring drawings.
PRXU	Dipshunt used for standard external connection drawings.
PRXUB	Dipshunt used for standard external connection drawings.
PRXUB1	Dipshunt used for standard external connection drawings.
RCC	Relay contact - energize to close.
RCO	Relay contact - energize to open.

Table B-1. System Library (continued)

Name	Description
REL	Relay coil.
RES	Resistor - horizontal orientation.
RESV	Resistor - vertical orientation.
TB12	Twelve point terminal block.
TW	Twist with shape for twisted pair wires - for wiring drawings.
V125	125 VDC source symbol.
V24	24 VDC source symbol.
Reference Connectors	
IREF	Input reference bubble (off page signal connection).
IREFO	Left-facing output reference bubble (not frequently used).
OREF	Output reference bubble (off page signal connection).

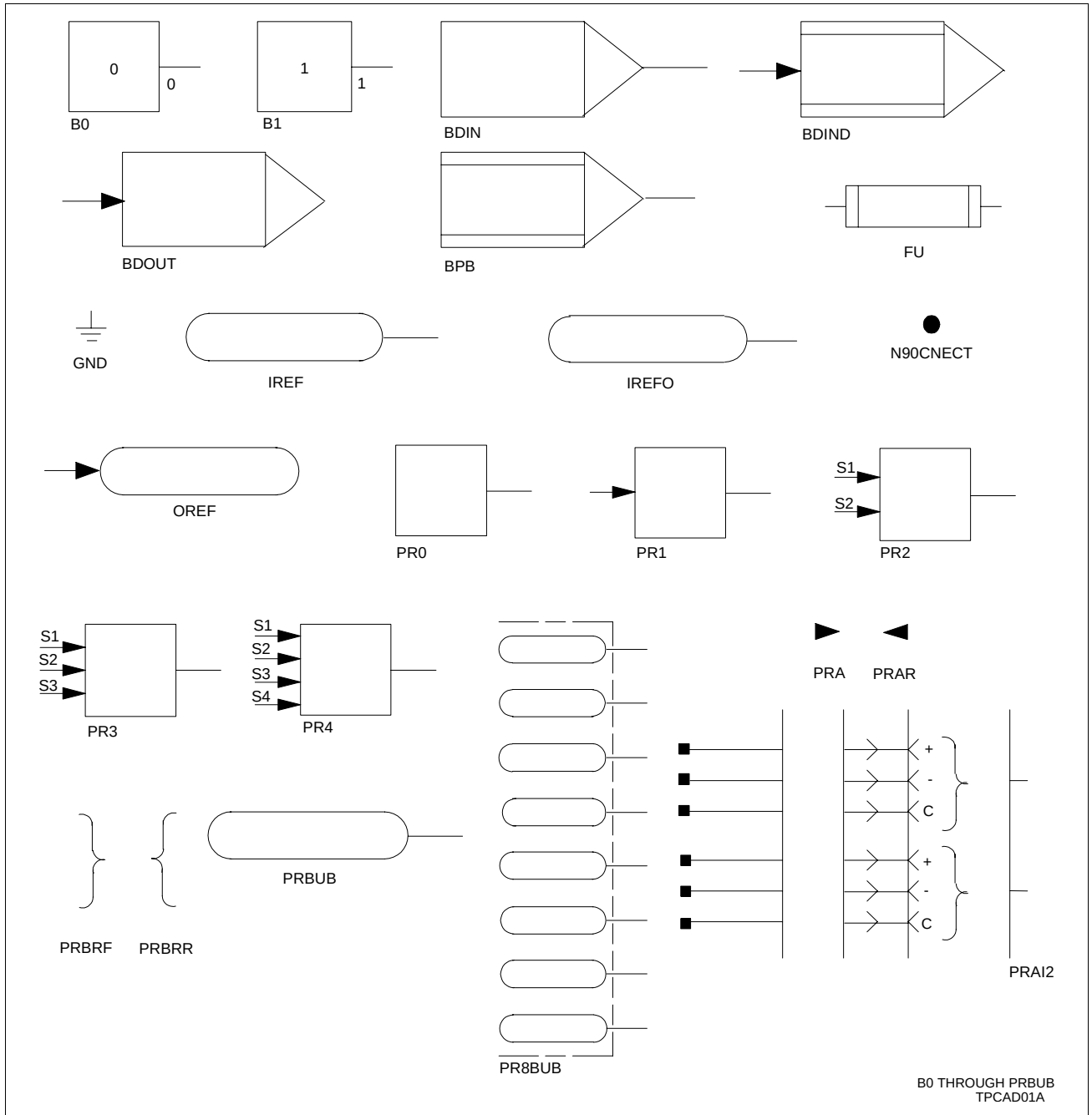


Figure B-1. System Library Symbols (1 of 4)

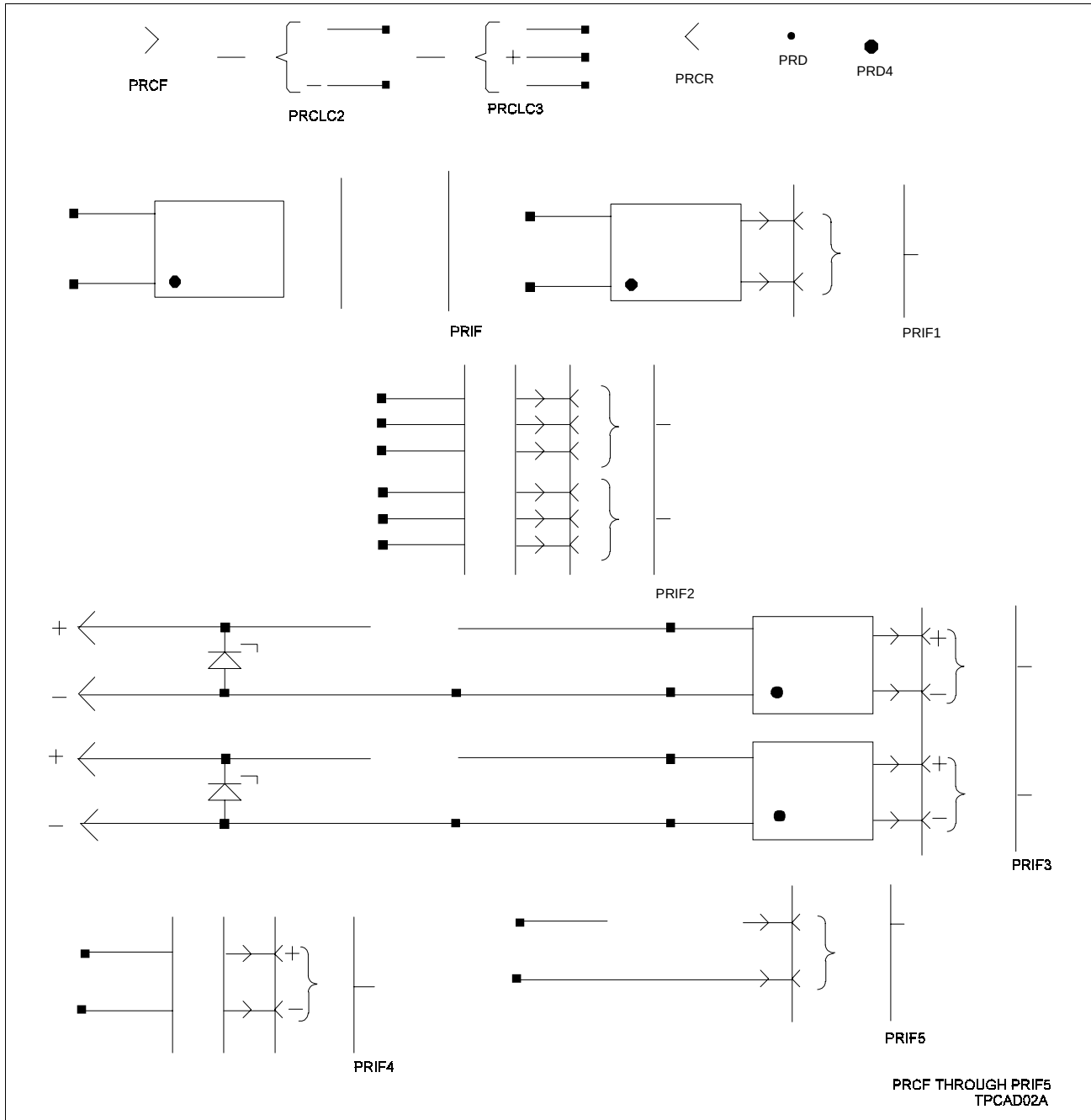


Figure B-2. System Library Symbols (2 of 4)

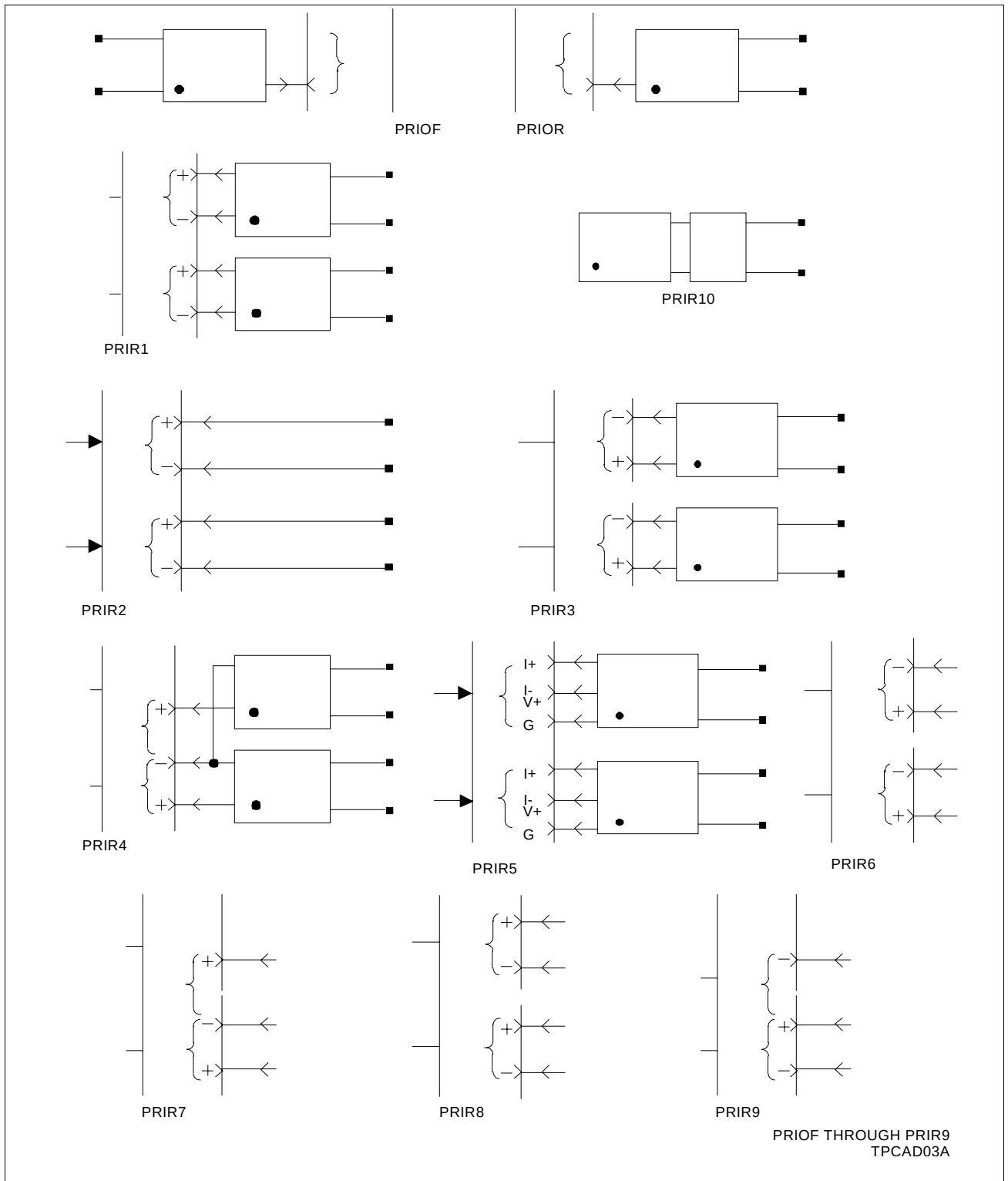


Figure B-3. System Library Symbols (3 of 4)

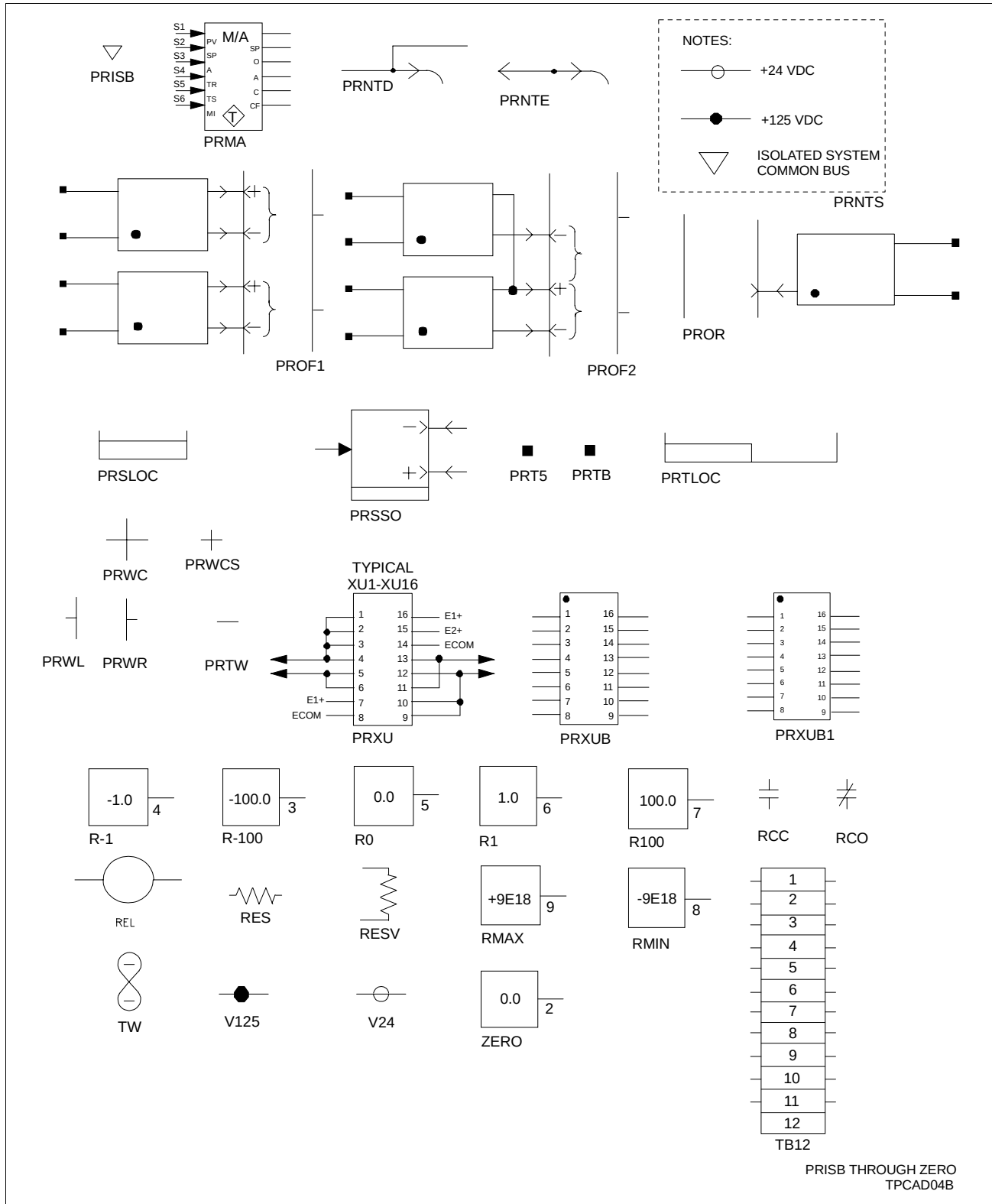


Figure B-4. System Library Symbols (4 of 4)

APPENDIX C - SHEET NAMING TABLES

INTRODUCTION

This section provides translations for use when naming sheets. This is necessary when sheet numbering goes past 99 (since the last three characters of a sheet name must be *ssr*, where *ss* is the sheet number and *r* is the revision). It is also necessary if using the PCU number in the sheet name and PCU numbering goes past 99.

NUMBERING UP TO 1296 SHEETS

Table C-1 shows one method used for numbering sheets that can handle up to 1,296 sheets.

Table C-1. 1296 Sheet Numbering

Range	Numbering
First 100	00 through 99
Next 26	0A through 0Z (not recommended)
Next 26	1A through 1Z (not recommended)
.	.
.	.
.	.
Next 26	9A through 9Z (not recommended)
Next 36	A0 through A9, then AA through AZ
Next 36	B0 through B9, then BA through BZ
.	.
.	.
.	.
Last 36	Z0 through Z9, then ZA through ZZ

NUMBERING MODULES

Table C-2 provides a one-character translation for decimal numbers up to 35 if using the module number in the sheet name.

Table C-2. Module Number Translation

Module	Decimal Number	Module	Decimal Number	Module	Decimal Number
A	10	J	19	S	28
B	11	K	20	T	29
C	12	L	21	U	30
D	13	M	22	V	31
E	14	N	23	W	32
F	15	O	24	X	33
G	16	P	25	Y	34
H	17	Q	26	Z	35
I	18	R	27		

NUMBERING UP TO 360 SHEETS OR 250 PCUS AND LOOPS

Table C-3 provides a two-character translation for decimal numbers up to 359. Table C-3 was used prior to release 5.3.

Table C-3. Sheet and PCU Translation

Range	Numbering
First 100	0 through 99
Next 10	A0 through A9
Next 10	B0 through B9
.	.
.	.
.	.
Last 10	Z0 through Z9

B

Binder	6-2
Results	6-5
Starting	6-2
Boolean trigger expressions	11-18

C

CMU	
Application window	7-1
CAD sheet operations	7-39
Configuration maintenance	7-1
Configuration operations	7-35
Editing specifications	5-59, 7-50, 10-13
File system operations	7-41
Function code editor	7-47
Hardware operations	7-28
Module operations	7-25
Monitor/trend operations	7-41
PCU operations	7-25
Project operations	7-21
Properties	7-16
Report operations	7-43
Utilities	7-50
Verify operations	7-42
Viewing/filtering options	5-58, 7-49, 10-12
Window operations	7-7
Communications	
Configuring	11-9
Compiler	6-1
Results	6-4
Starting	6-2
Compiling	
SEM module	11-19
Configuration utilization	7-44
Configure	
Complex trigger list	11-17
Conventions	
Document	1-4
Cross referencer	6-1
Results	6-4
Starting	6-2

D

DSOE	11-10
Compile SEM	11-19
Complex trigger list	11-17
Configuration	11-10, 11-11, 11-19
Configure point list	11-15
Display	11-19
Display configuration	11-19

Distributed sequence of events	11-10
External drawing	11-11
Point list	11-14, 11-15
Point list from .CAD	11-14
Print	11-22
SEM executive block	11-12
Update .CAD files from point list	11-17

F

Features	1-1
Files	
BND	3-8, 6-4
CAD	3-8, 7-22
CFG	3-8, 7-22
I90BIND.LOG	6-6
I90XREF.ERR	6-5
I90XREF.LOG	6-5
I90XREF.OUT	6-3, 6-5
JHD	3-7, 7-22
LOG	3-8, 6-4
MHD	3-8, 7-22
PRJ	3-7, 7-22
REF	3-8, 6-4
SPC	12-2
Function code editor	5-56, 7-47, 10-10
Controls	5-58, 7-48, 10-12
Window	5-56, 7-47, 10-11

G

Glossary	1-5
----------------	-----

H

Hardware key	2-9
--------------------	-----

I

Inspect	8-1
Application window	8-2
Inspect operations	8-5
Module mode operations	8-20
On-line configuration	8-9
Report operations	8-21
Window operations	8-5
Installation	2-2
Intended user	1-1

L

Launcher	3-2
Library	4-15

Index (continued)

Compressing	4-17
License management.....	2-9
Log files.....	11-8
Viewing.....	11-8
Loop numbering	C-2

M

Modify file header.....	7-38
Module configuration	
Attributes	5-25
Automatic borders	5-29
Borders.....	5-28
Changing line style	5-51
Choosing commands.....	5-16
Closing a sheet.....	5-13
Closing the application	5-6
Configuration components	5-26
Copying entities.....	5-51
Cursor.....	5-22
Cutting entities.....	5-51
Deleting entities.....	5-52
Distance	5-24
Document windows	5-13
Document workspace	5-16
Drawing entities.....	5-27
Dropping entities	5-27
Editing entities.....	5-47
Environment	5-19
Function code editor.....	5-56
Function codes.....	5-31
General procedure	3-15
Graphically configuring.....	5-1
Grid.....	5-23
Input and output references	5-40
Layers.....	5-25
Libraries.....	5-59
Magnifying	5-21
Menu items.....	5-9
Monitor/tune	5-5, 5-63
Multiple document windows	5-14
Multiple modules in a project.....	5-33
Multiple sheets in a configuration	5-40
On-line help.....	5-10, 8-4, 10-4
Operating modes.....	5-5
Panning	5-22
Parts of the window.....	5-7, 8-2
Pasting entities.....	5-51
Printing a document	5-12
Receiving signals from another module	5-35, 5-37
Redo.....	5-56
Refreshing the view.....	5-19

Saving a document	5-11
Selecting entities.....	5-45
Standard drawings.....	5-62
Starting	5-5
Supplemental utilities.....	3-16
Toolbar.....	5-17, 7-2, 8-4, 10-4
Undo	5-56
Module numbering	C-1
Module tools executive	3-3
Menus.....	3-9
Parts of the screen.....	3-5
Starting	3-3
Modules	4-7
Adding.....	4-7
Adding sheets	4-9
Defining module information	4-7
Editing module information	4-9
Importing.....	4-13
Modify file header	7-38
Using standard drawings	4-12

O

On-line help	3-14
--------------------	------

P

Password security	12-1
PCU numbering	C-2
Print	11-3
Printing.....	11-6
Queue manager	11-7
Project	
Adding Libraries.....	4-16
Adding modules	4-7
Adding PCUs	4-7
Adding Sheets	4-9
Adding standard drawings	4-12
Closing.....	4-4
Defining.....	4-2
Defining border information	4-5, 7-25
Defining module information	4-7
Defining structure.....	4-5
Deleting branches from the project tree.....	4-15
Deleting libraries	4-17
Editing module information	4-9
Editing project information	4-4
Importing existing modules	4-13
Managing.....	4-1
Managing libraries	4-15
Opening	4-4
Upgrading the TDT database	4-18

Project directory structure..... 4-1
Project structure 3-7, 7-22
 Global tag database..... 3-7
 Loop..... 3-8, 7-22
 PCU folder 3-8, 7-22
 Project folder..... 3-7, 7-22
 Sheet 3-8, 7-22
Project tree 3-6
 Using..... 3-6, 7-21

R

Related documents 1-6
Release information 2-4
Requirements
 Certified computers..... 2-2
 Hardware 2-1
 Minimum configuration..... 2-2
 Software..... 2-2

S

Sheet numbering C-1, C-2
SLV
 Save, load, verify 9-1
 Script commands 9-7
 Script file utilities..... 9-6
 Script generator 9-5
 SLV utility..... 9-1
 Application window 9-2
 Properties..... 9-5
 Script file operations 9-4

Software key 2-10
Supported products..... 1-6
Synchronizing tag data 11-1
System library B-1
 System symbols B-1

T

Trace spare block numbers 7-45
Trend
 Application window..... 10-2
 Function code editor..... 10-10
 Graph display options 10-7
 Group operations 10-5
 Monitor a group 10-8
 Trend a group..... 10-8
 Trend, monitor, and tune..... 10-1
 Tune a point 10-10
 Window operations..... 10-4

U

Upgrading files 12-2

Visit Elsas Bailey on the World Wide Web at <http://www.bailey.com>

Our worldwide staff of professionals is ready to meet *your* needs for process automation.
For the location nearest you, please contact the appropriate regional office.

AMERICAS

29801 Euclid Avenue
Wickliffe, Ohio USA 44092
Telephone 1-216-585-8500
Telefax 1-216-585-8756

ASIA/PACIFIC

152 Beach Road
Gateway East #20-04
Singapore 189721
Telephone 65-391-0800
Telefax 65-292-9011

EUROPE, AFRICA, MIDDLE EAST

Via Puccini 2
16154 Genoa, Italy
Telephone 39-10-6582-943
Telefax 39-10-6582-941

GERMANY

Graefstrasse 97
D-60487 Frankfurt Main
Germany
Telephone 49-69-799-0
Telefax 49-69-799-2406