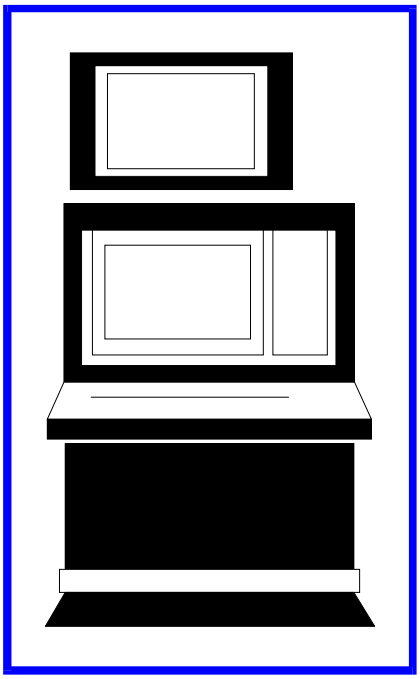
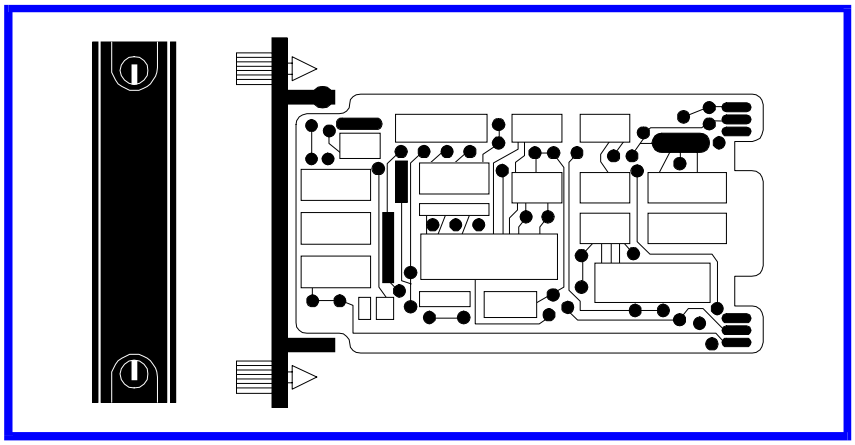
Bailey® infi 90®

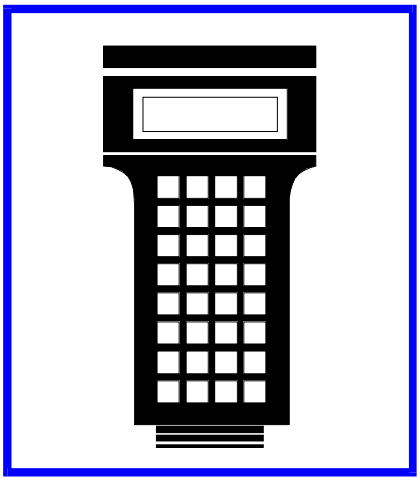
PWC 90

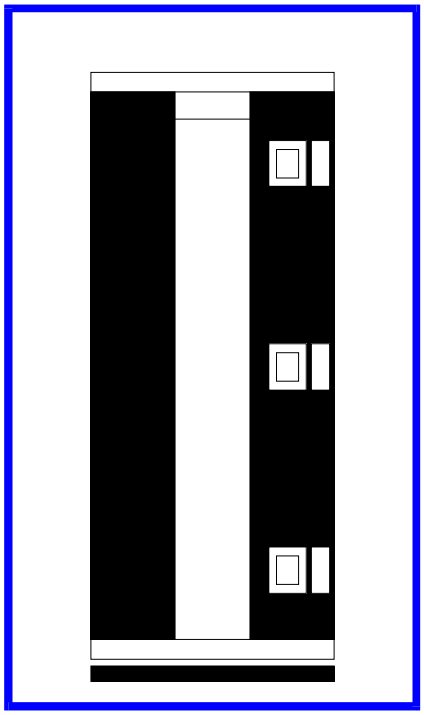
Instruction

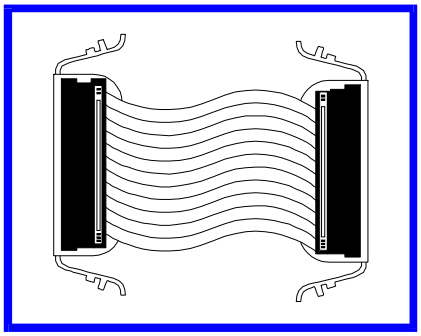
Configuration
Personal Work Center 90
(Release 1.0)

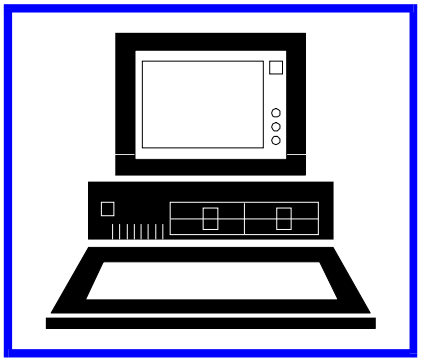












Bailey Process Control and Automation Solutions from Elsag Bailey Group

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 **Bailey Controls** 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.

AVERTISSEMENT

MANUELS D'OPÉRATION

NE PAS METTRE EN PLACE, RÉPARER OU FAIRE FONCTIONNER L'ÉQUIPEMENT SANS AVOIR LU, COMPRIS ET SUIVI LES INSTRUCTIONS RÉGLEMENTAIRES DE **Bailey Controls** TOUTE NÉGLIGENCE À CET ÉGARD POURRAIT ÊTRE UNE CAUSE D'ACCIDENT OU DE DÉFAILLANCE DU MATÉRIEL.

PERTURBATIONS PAR FRÉQUENCE RADIO

LA PLUPART DES ÉQUIPEMENTS ÉLECTRONIQUES SONT SENSIBLES AUX PERTURBATIONS PAR FRÉQUENCE RADIO. DES PRÉCAUTIONS DEVRONT ÊTRE PRISES LORS DE L'UTILISATION DU MATÉRIEL DE COMMUNICATION PORTATIF. LA PRUDENCE EXIGE QUE LES PRÉCAUTIONS À PRENDRE DANS CE CAS SOIENT SIGNALÉES AUX ENDROITS VOULUS DANS VOTRE USINE.

PERTURBATIONS DU PROCÉDÉ

L'ENTRETIEN DOIT ÊTRE ASSURÉ PAR UNE PERSONNE QUALIFIÉE EN CONSIDÉRANT L'ASPECT SÉCURITAIRE DES ÉQUIPEMENTS CONTRÔLÉS PAR CE PRODUIT. L'AJUSTEMENT ET/OU L'EXTRACTION DE CE PRODUIT PEUT OCCASIONNER DES À-COUPS AU PROCÉDÉ CONTRÔLE LORSQU'IL EST INSÉRÉ DANS UNE SYSTÈME ACTIF. CES À-COUPS PEUVENT ÉGALEMENT OCCASIONNER DES BLESSURES OU DES DOMMAGES MATÉREILS.

NOTICE

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

Bailey Controls Company, 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 Bailey Controls Company, 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 Bailey Controls Company.

Preface

This instruction covers the configuration of the Personal Work Center 90 (PWC90) software. PWC90 operates in the Windows NT™ version 4.0 operating system. It communicates with the INFI 90® OPEN system via the INICI03 connected to the INFI-NET® system.

The PWC90 console provides an interface to the INFI 90 OPEN system.

Special features of the PWC90 console include:

- On-line process graphic configuration.
- Configuration and on-line changing of the tag database.
- Monitoring and logging of alarms.
- Ability to create process control graphics.
- Multilevel security system.
- Creation of spreadsheets using Microsoft EXCEL.
- Microsoft Windows access to other software applications.
- Tag summaries application.
- Module details application.
- Block details application.
- Server diagnostics application.
- Fixed and/or custom display hierarchy.
- Open standard data server (@aGlance™).

List of Effective Pages

Total number of pages in this instruction is 418, consisting of the following:

Page No.	Change Date
Preface	Original
List of Effective Pages	Original
iii through xix	Original
1-1 through 1-8	Original
2-1 through 2-68	Original
3-1 through 3-12	Original
4-1 through 4-98	Original
5-1 through 5-21	Original
6-1 through 6-13	Original
7-1 through 7-12	Original
8-1 through 8-9	Original
9-1 through 9-19	Original
10-1 through 10-9	Original
11-1 through 11-21	Original
12-1 through 12-29	Original
A-1 through A-12	Original
B-1 through B-44	Original
C-1 through C-22	Original
Index-1 through Index-2	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

Page

SECTION 1 - INTRODUCTION 1-1
INTRODUCTION 1-1
INTENDED USER 1-1
FUNCTIONAL DESCRIPTION 1-1
User Interface 1-2
PWC90 Console Management Features 1-2
Hierarchical Displays 1-2
Graphic Configuration 1-3
Historical Database 1-3
Logging 1-3
Trending 1-3
System Status Display 1-4
PWC90 Console Utilities 1-4
Historical Block Data Collection 1-4
@aGlance 1-4
Tag Configuration 1-4
Tag.Atom List 1-4
Bailey Submodels 1-5
EXCEL Functions 1-5
DOCUMENT CONVENTIONS 1-5
REFERENCE DOCUMENTS 1-5
GLOSSARY OF TERMS AND ABBREVIATIONS 1-6

SECTION 2 - SYSTEM CONFIGURATION FUNCTIONS 2-1
INTRODUCTION 2-1
GETTING STARTED 2-1
System Default State 2-1
Logging In 2-2
Logging Out 2-4
ARCHIVAL GROUPS 2-4
Edit Archival Group 2-5
Group Name 2-6
Period and Start Time 2-6
Area 2-7
Add Archival Group 2-7
Delete Archival Group 2-8
Delete All Archival Groups 2-8
List Tags in an Archival Group 2-8
Exiting Archival Group Configuration 2-8
MESSAGE CONFIGURATION 2-8
Message Text 2-9
Message Priority 2-9
Message Destination Area 2-9
Send the Operator Message 2-9
ALARM GROUP CONFIGURATION 2-10
Accessing Alarm Group Configuration 2-10
Adding an Alarm Group 2-11
Group Name 2-12
On List 2-12
Off List 2-13
Configure Filtering of Message Classes and Types 2-13
Edit an Alarm Group 2-15
Delete an Alarm Group 2-16

Table of Contents (continued)

Page

SECTION 2 - SYSTEM CONFIGURATION FUNCTIONS (continued)

CONFIGURING NETWORK PARAMETERS 2-16
Alarm Setup 2-16
System Title 2-17
Top Alarm Line Presentation 2-17
Priority Color 2-18
Audible Alarm Configuration 2-19
Default Alarm Setup 2-21
Mini-Alarm Window Field Width 2-21
Exiting Alarm Setup 2-22
Summary - Alarm Setup 2-23
Alarm Vectors 2-24
Configuring the Triggering Tagname 2-25
Specifying the Target Display 2-26
Changing Existing Alarm Vectoring Assignments 2-28
Configuring Multiple Alarm Vectoring Assignments 2-28
Deleting Alarm Vectoring Assignments 2-28
Using the Vector Button 2-29
Area Name Assignment 2-29
Event Bit Name Configuration 2-30
Goto Function Button 2-31
Find Function Button 2-32
Delete All Function Button 2-32
Exiting Event Bit Name Configuration 2-32
Printer Assignment 2-32
Adding a Device 2-32
Changing Device Characteristics 2-36
Device Sort 2-37
Deleting a Device 2-37
Deleting All Devices 2-37
Message Routing 2-38
Message Type Filtering 2-39
Message Color Definition 2-40
Message Indentation Definition 2-41
Message Class: Process Alarm Messages 2-42
Message Class: System Event Messages 2-43
Message Class: Operator Messages 2-43
Message Class: Operating Errors 2-43
Message Class: Operator Actions 2-44
Message Filter Parameter: Priority 2-44
Message Filter Parameter: Area 2-44
Operator Window Parameters 2-45
Update Rates 2-45
Replacement Size 2-46
Return to Default Settings 2-47
Apply Changes 2-47
Exiting the Operator Window Parameter Configuration 2-47
DISPLAY COLORS 2-47
SECURITY 2-49
Security Activation 2-49
Display Center Default 2-50
Default Login Access Level 2-51
Default Login Message Filtering 2-52
Area Access Assignment (Default Login) 2-53

Table of Contents (continued)

Page

SECTION 2 - SYSTEM CONFIGURATION FUNCTIONS (continued)
Clear and Set Multiple Message Types 2-53
Exiting Default Login Configuration 2-53
Function Access 2-53
Server and Tag Access 2-56
Function Access Level Assignment Guidelines 2-57
User Login 2-57
Add a User Login 2-58
Copy a User Login 2-62
Edit a User Login 2-64
Delete a Login 2-64
Cancel Login Configuration 2-64
Change Password 2-64
Summary - Security 2-66

SECTION 3 - HIERARCHICAL DISPLAYS 3-1
INTRODUCTION 3-1
HIERARCHICAL DISPLAY DESCRIPTIONS 3-3
Plant Summary Display 3-3
Group Display 3-5
Point Tagnames 3-5
Point Legends 3-5
Point Display 3-6
HIERARCHICAL DISPLAY CONFIGURATION 3-7
Summary Configuration 3-7
Group Configuration 3-8
End Group Configuration 3-11
Point Displays 3-11
SUMMARY - HIERARCHICAL DISPLAYS 3-11

SECTION 4 - GRAPHICS CONFIGURATION 4-1
INTRODUCTION 4-1
ACCESSING GRAPHIC CONFIGURATION 4-1
CREATING MODELS 4-2
Working View Area 4-3
Message Area 4-4
Object Create Control Panel 4-5
Creating Objects - Straight Lines 4-6
Creating Objects - Rectangles 4-6
Creating Objects - Circles 4-7
Creating Objects - Markers 4-8
Creating Objects - Polylines 4-8
Creating Objects - Splines 4-9
Creating Objects - Closed Splines 4-10
Creating Objects - Arcs 4-11
Creating Objects - Polygons 4-12
Using the Text Button 4-13
Using the Filled Text Rectangle Button 4-14
Changing Text Object Attributes 4-15
Creating Objects - Submodels 4-16
Object Change Control Panel 4-19
Select List 4-20

Table of Contents (continued)

Page

SECTION 4 - GRAPHICS CONFIGURATION *(continued)*

Extent and Reference Points 4-20
Rotate 4-21
Move 4-22
Scale 4-23
Movept 4-24
Copy 4-25
End 4-26
Delete 4-26
Status Control Panel 4-27
Fill Mode 4-27
Grid Mode 4-28
Point Snap Mode 4-28
Redraw Button 4-29
Text Attribute Control Panel 4-29
Fill Pattern Attribute Panel 4-30
Lines Attribute Panel 4-31
Color Attribute Panel 4-31
Background Color Configuration 4-32
Menu Bar 4-33
Installed and Off-line Graphic and Submodel Files 4-33
File Pull-Down Menu 4-35
Edit Pull-Down Menu 4-40
View Pull-Down Menu 4-43
Select Pull-Down Menu 4-46
Options Pull-Down Menu 4-47
Window Pull-Down Menu 4-48
Object Placement Example 4-50
DYNAMICS 4-51
Configure Attributes - Dynamic Attributes for Objects 4-52
Configuring the Expression - Operators and Operands 4-53
Configuring the Attribute Settings 4-59
Configure Submodel 4-65
Create Custom Dynamics and Declare Variables 4-67
Custom Dynamics 4-67
Declaring Variables in Graphics 4-68
Declaring Variables in Submodels 4-71
Naming Variables 4-72
Dynamic Configuration Examples 4-74
Example 1 - Single Expression 4-75
Example 2 - Multiple Expressions 4-75
Example 3 - Using the True Expression 4-76
Example 4 - Creating a New Submodel With Dynamics 4-76
Example 5 - Configuring Dynamic Submodels 4-77
Example 6 - Simulating Motion Using the Blink Attribute 4-77
Example 7 - Using Text to Indicate Conditions 4-80
BAILEY SUBMODELS 4-81
EXPRESSIONS AS SUBMODEL VARIABLES 4-81
PWC SYMBOLS 4-84
Valves and Actuators 4-84
Containers and Vessels 4-84
Rotating Equipment (Pumps and Blowers) 4-85
Reciprocating Equipment 4-85
Mixing Equipment 4-85

Table of Contents (continued)

Page

SECTION 4 - GRAPHICS CONFIGURATION (continued)

Material Handling Equipment 4-85
Electrical Equipment 4-85
Filters 4-86
Scrubbers and Precipitators 4-86
Separators 4-86
Heat Transfer Devices 4-86
HVAC Equipment 4-86
Miscellaneous 4-87
ymbol Illustrations 4-87
INFI 90 OPEN FACEPLATE SUBMODELS 4-91
CUSTOM MODELS FOR POINT AND GROUP DISPLAYS 4-92
Group Display Customization Techniques 4-92
LOOP AND NODE STATUS DISPLAYS 4-93
EXAMPLE CONFIGURATION 4-93
CREATING SUMMARY GRAPHICS 4-94
i90_Console (Using i90_NodeSummPB, i90_NodeErrSummPB) 4-96
i90_ProcessModule (Using i90_ModuleSummPB, i90_NodeErrSummPB) 4-97
i90_ProcessNode (Using i90_NodeVectorSummPB, i90_NodeErrSummPB) 4-97
i90_Bridge (Using i90_NodeVectorSummPB, i90_NodeErrSummPB) 4-98

SECTION 5 - HISTORICAL DATABASE 5-1

INTRODUCTION 5-1
EDIT HISTORICAL DATABASE ITEM 5-2
Point Information 5-3
Historical Database Tagname 5-3
Historical Database Point Legend 5-4
Collection Information 5-4
Collection Rate 5-4
Computations 5-8
Storage Rate 5-9
Collection Option 5-10
Archive Group 5-11
Tag Atom 5-12
Event Bit 5-12
Active Value 5-12
Acknowledge Bit 5-13
Storage Information 5-13
Add Data Storage Definitions 5-13
Edit Data Storage Definitions 5-16
Delete Data Storage Definitions 5-16
Convenience Buttons 5-16
Replace 5-16
Add 5-16
Prev/Next 5-17
Return to Default Settings 5-17
Historical Database Configuration and Trend Compression 5-18
OTHER HISTORICAL DATABASE OPERATIONS 5-18
Add 5-18
Find 5-19
Delete Historical Database 5-19
Delete All Historical Database 5-19
Exiting Historical Database Configuration 5-20

Table of Contents (continued)

Page

SECTION 5 - HISTORICAL DATABASE (continued)
SUMMARY - HISTORICAL DATABASE 5-20

SECTION 6 - LOGS 6-1
INTRODUCTION 6-1
ACCESSING LOG CONFIGURATION 6-1
EVENT LOGS 6-2
Add 6-2
Configure Event Log Name 6-3
Configure Message Classes and Types 6-3
Edit 6-5
Delete 6-6
Scheduling 6-6
Schedule Button 6-6
Scheduled Time Parameter 6-6
Event Bit Name Parameter 6-7
Device Parameter 6-7
Print Banner Parameter 6-8
Next Occurrence Indicator 6-8
Exiting Event Log Scheduling 6-9
Print 6-9
SPREADSHEET LOGS 6-10
Add 6-10
Edit 6-10
Delete 6-10
Schedule 6-11
Print 6-12
Clearing the Recovery Log Notify Window 6-12
Spreadsheet Log Special Functions 6-12

SECTION 7 - TRENDING 7-1
INTRODUCTION 7-1
ACCESSING HISTORICAL TRENDS 7-1
CONFIGURING HISTORICAL TRENDS 7-3
Title 7-5
Scale Bar Limits 7-5
Segment Size 7-6
Trace Configuration 7-7
Historical Tag 7-7
Collection Info 7-9
Trend Trace Limits: High and Low 7-9
Trace Default State: On or Off 7-9
Display Options 7-10
Description 7-10
Type 7-10
SUMMARY - TRENDS 7-10

SECTION 8 - SYSTEM STATUS DISPLAY 8-1
INTRODUCTION 8-1
GENERAL INFORMATION FOR ALL NODE TYPES 8-3
Node Name 8-3
Info Menu 8-3

Table of Contents (continued)

Page

SECTION 8 - SYSTEM STATUS DISPLAY (continued)
Network Info 8-4
Show Available Services 8-4
INFORMATION FOR SPECIFIC NODE TYPES 8-5
PWC90 Nodes 8-5
INFI 90 OPEN Server Nodes 8-6
UTILITIES MENU 8-7
Device Status 8-7
Printer Queue Control 8-7
Printer Queue Information 8-9

SECTION 9 - PWC90 UTILITIES 9-1
INTRODUCTION 9-1
DATABASE MAINTENANCE 9-2
Historical Database Maintenance 9-2
Locating Specific Historical Tags 9-5
Historical Block Database Maintenance 9-6
Archival Database Maintenance 9-6
Locating Specific Archival Tags 9-8
Deleting Archival Data 9-8
Archival Block Database Maintenance 9-9
CRT PRINT 9-9
Selecting the Image Type 9-9
Selecting the Printer 9-10
Printing the Image 9-10
RELEASE ALL CONFIGURATION LOCKS 9-10
PWC90 INFI-90 TIME SYNCHRONIZATION 9-11
Current Adjust Status 9-11
New Adjust Target 9-12
OTHER APPLICATIONS 9-13
ACCESSING OTHER APPLICATIONS 9-13
PWC90 HELP 9-13
QUICK KEYS 9-13
Accessing the Quick Keys Feature 9-13
Configuring Quick Keys 9-15
Configuring Quick Keys 33 Through 64 9-16
Clearing Quick Key Assignments 9-16
Exiting Quick Keys 9-17
Summary - Quick Keys 9-17
CRT CONTEXT KEYS 9-17
Preparing the Context 9-18
Initial Display Layout - Login Context Key 9-18
Accessing CRT Context Keys 9-18
Configuring CRT Context Keys 9-19
Clearing CRT Context Keys 9-20
Exiting CRT Context Keys 9-20
Summary - CRT Context Keys 9-20

SECTION 10 - HISTORICAL BLOCK DATA COLLECTION 10-1
INTRODUCTION 10-1
ADD HISTORICAL BLOCK 10-2
Historical Block Name 10-3

Table of Contents (continued)

Page

***SECTION 10 - HISTORICAL BLOCK DATA COLLECTION** (continued)*

Historical Block Legend 10-3
Period and Start Time 10-3
Node 10-4
Number Retained 10-4
Historical Block Collection Options 10-5
Raw Data Option 10-5
Pause Option 10-5
Reset Option 10-5
Tag Atom Assignment 10-5
Historical Block Data Computations 10-6
Configuring Multiple Historical Blocks 10-7
Exiting Historical Block Configuration 10-8
HISTORICAL BLOCK OPERATIONS 10-8
Edit 10-8
Copy 10-8
Delete 10-8
Find 10-9
Exiting Historical Database Configuration 10-9

SECTION 11 - @aGlance/IT SERVER INTERFACE 11-1

INTRODUCTION 11-1
@aGlance/IT API for the INFI 90 OPEN System 11-1
Requirements for the INFI 90 OPEN @aGlance/IT Server 11-2
@aGlance/IT SERVER OPERATION 11-2
Configuring the @aGlance/IT Server Internal Interface 11-2
Registering Client Nodes 11-3
Starting a Server and Server Names 11-3
Starting a Server With Non-default Options 11-4
Configure @aGlance/it Current Tagname 11-5
Viewing @aGlance/it Server Process Data 11-10
Backup/Restore @aGlance/IT Configuration 11-12
SUPPORTED FEATURES FOR USE WITH @aGlance/IT CLIENTS 11-13
@aGlance/IT Methods Supported 11-13
Default Attributes Names 11-14
Format of Timestamps 11-14
Server-Specific Parameters for History-Related Methods 11-15
Server-Specific Status Information 11-16
SECURITY FOR THE @aGlance/IT SERVER 11-17
Security for Access to the @aGlance/IT Server for PWC90 11-17
Security for INFI 90 OPEN Database Access 11-18
MONITORING 11-18
Monitor Time Interval 11-18
Monitor Types 11-19
Polled Monitoring 11-20
Deadband-absolute Monitoring 11-20
Deadband-percent Monitoring 11-21
Monitor Request Size 11-21
SUPPORTED @aGlance/IT CLIENT APPLICATIONS 11-21
@aGlance/IT ADD-IN FOR MICROSOFT EXCEL 11-21

Table of Contents (continued)

Page

SECTION 12 - TAG CONFIGURATION 12-1
INTRODUCTION 12-1
TAGS 12-1
AVAILABLE TAGS LIST 12-3
FILE OPTIONS 12-3
Print Option 12-4
Purge Deleted Tags 12-4
Exit Option 12-4
EDIT OPTIONS 12-4
New Tag 12-4
General Tab 12-9
INFI 90 Tab 12-11
Digital Tab 12-12
Remote Control Memory 12-13
Copy Tag 12-14
Change Operating Status of a Tag 12-14
Search 12-15
VIEW OPTIONS 12-15
Sort 12-15
Filter 12-16
Operating Status Filters 12-17
TOOLS 12-17
Import Trends 12-18
Import Configuration 12-19
Configure Text 12-21
Indexed Text 12-21
Tag Descriptors 12-23
Alarm Priority Text 12-26
Substitutable Text 12-26
Remote Motor Control Text 12-27

APPENDIX A - PWC90 TAG.ATOMS A-1
ATOM SUFFIXES A-1
TAG.ATOM LIST A-1

APPENDIX B - BAILEY SUBMODELS B-1
INTRODUCTION B-1
SUBMODELS FOR CONTROL AND DATA ENTRY B-1
FP_DYbargraph B-2
FP_DYbargraph_ec B-4
FP_DYctrlpb B-5
FP_DYdial B-6
FP_DYentertextpb B-7
FP_DYentervaluepb B-8
FP_DYgroupmod B-9
FP_DYHisttrend B-10
FP_DYHorizbar B-11
FP_DYHoriztrend B-12
FP_DYhslider B-14
FP_DYmeter B-15
FP_DYmpoppb B-16
FP_DYpointmod B-17

Table of Contents (continued)

Page

***APPENDIX B - BAILEY SUBMODELS** (continued)*

FP_DYpump_lf B-18
FP_DYpump_rt B-20
FP_DYput1pb B-22
FP_DYputpb B-23
FP_DYputpbstat B-25
FP_DYputpbstat2 ..... B-27
FP_DYstatpb B-29
FP_DYtagpb B-30
FP_DYtank B-31
FP_DYtank_ec B-32
P_DYTrendpb B-36
FP_DYvalue B-37
FP_DYvectorpb B-39
FP_DYVertbar B-40
FP_DYverttrend B-41
FP_DYvslider B-43
FP_DYwheel B-44

APPENDIX C - EXCEL FUNCTIONS C-1

INTRODUCTION C-1
fparchive( ) C-2
fpblkarch( ) C-3
fpblkclose( ) C-4
fpblkopen( ) C-5
fpblksample( ) C-6
fpblkvald( ) C-8
fphget( ) C-9
fphgetd( ) C-10
fphput( ) C-11
fphputd( ) C-12
fphxgetval( ) C-13
fphxnumval( ) C-14
fpput( ) C-15
fpputd( ) C-16
fpsendmsg( ) C-17
fpudevice( ) C-18
fpuprint( ) C-19
fpverbose( ) C-20
fpxammts( ) C-21

List of Figures

No.	Title	Page
2-1.	System Window Menu with Cascading Login/Logout Menu	2-2
2-2.	PWC90 Console Login Window	2-3
2-3.	Archival Group Configuration Window	2-5
2-4.	Archival Group Scheduling Window	2-6
2-5.	Area Selection Window	2-7
2-6.	Message Configuration Window	2-9
2-7.	Alarm Group Selection Window	2-11
2-8.	Alarm Group Configuration Window	2-12
2-9.	Alarm Group Filtering Window	2-14
2-10.	Alarm Setup Window	2-17
2-11.	Mini-Alarm Format Window	2-22
2-12.	Alarm Vector Configuration Window	2-25
2-13.	Select Graphic Window	2-26
2-14.	Printer Selection Window	2-28
2-15.	Area Setup Display	2-30
2-16.	Event Bit Configuration Window	2-31
2-17.	Printer Assignment Display	2-33
2-18.	Device Characteristics	2-33
2-19.	Sort Criteria Window	2-37
2-20.	Message Routing Assignment Window	2-39
2-21.	Device Routing Assignment Window	2-40
2-22.	Message Color Selection Window	2-41
2-23.	Message Indentation Definition Window	2-42
2-24.	Operator Window Parameters Window	2-45
2-25.	Display Colors Window	2-48
2-26.	Security Activation/Deactivation Windows	2-50
2-27.	Default Login Configuration Window	2-51
2-28.	Access Level Selection Window	2-52
2-29.	Function Access Configuration	2-54
2-30.	User Login Configuration Window	2-58
2-31.	User Login Window	2-59
2-32.	Access Level Selection Window	2-61
2-33.	Copy User Login Window	2-63
2-34.	Change Password Window	2-65
2-35.	Illegal Entry Pop Up Window	2-66
3-1.	Operator Window Select Menu	3-2
3-2.	Summary Display Example	3-4
3-3.	Group Display Example	3-6
3-4.	Point Display Example	3-7
3-5.	Configure Summary Window	3-8
3-6.	Configure Group Window	3-9
3-7.	Graphic List and Trend List Windows	3-10
4-1.	Graphics Editor Display, Working View Maximized	4-3
4-2.	Drawing Lines	4-6
4-3.	Drawing Rectangles	4-7
4-4.	Drawing Circles	4-7
4-5.	Creating Object markers	4-8
4-6.	Drawing Polylines	4-9
4-7.	Drawing Splines	4-10
4-8.	Drawing Closed Splines	4-11
4-9.	Drawing Arcs	4-12
4-10.	Drawing Polygons	4-13
4-11.	Inserting Text	4-14

List of Figures (continued)

No.	Title	Page
4-12.	Filed Text Rectangle	4-15
4-13.	Submodel Selection Window	4-17
4-14.	Object Change Control Panel	4-20
4-15.	Rotate Objects	4-21
4-16.	Move Objects	4-22
4-17.	Scale object	4-23
4-18.	Move Point	4-24
4-19.	Copy object	4-25
4-20.	Delete object	4-26
4-21.	Status Control Panel	4-27
4-22.	Fill Pattern Attribute Panel	4-30
4-23.	Lines Attribute Control Panel	4-31
4-24.	Color Attribute Control Panel	4-31
4-25.	Select Model to Load Window	4-35
4-26.	Edit Colors Window	4-41
4-27.	Zoom Window	4-44
4-28.	Model of ISA Standard Actuator Valve Symbol	4-50
4-29.	Actuator Valve Symbol with Pointer at New Reference Point	4-51
4-30.	Dynamics Pull-Down Menu	4-52
4-31.	Expression Dynamics Window	4-53
4-32.	Expressions Window	4-58
4-33.	Blink Window	4-63
4-34.	Spin Configuration Window	4-64
4-35.	Submodel Configuration Window	4-66
4-36.	Edit Custom Dynamics Window	4-67
4-37.	Variables Window	4-69
4-38.	Declare Variable Window	4-70
4-39.	Submodel Variable Configuration Window	4-78
4-40.	Bailey Symbols	4-87
4-41.	ISA Standard Symbols - Valves and Actuators	4-88
4-42.	ISA Standard Symbols - Containers and Vessels	4-88
4-43.	ISA Standard Symbols - Rotating and Reciprocating Equipment	4-89
4-44.	ISA Standard Symbols - Mixing and Material Handling Equipment	4-89
4-45.	ISA Standard Symbols - Electrical Equipment	4-90
4-46.	ISA Standard Symbols - Filters, Scrubbers, Precipitators and Separators	4-90
4-47.	ISA Standard Symbols - Heat Transfer Devices and HVAC Equipment	4-91
4-48.	LoopStatusSummary Graphic	4-95
4-49.	ProcessNodeSummary Graphic	4-96
5-1.	Historical Database Configuration Window	5-2
5-2.	Historical Tag Configuration Window	5-3
5-3.	Collection Rates Window	5-5
5-4.	Historical Tag Configuration Window for Manual Data Point Configuration	5-6
5-5.	Data Type Window	5-6
5-6.	Historical Tag Configuration Window for Custom Collection Rate Configuration	5-7
5-7.	Historical Tag Computations Window	5-9
5-8.	Storage Rate Window	5-10
5-9.	Collection Options Window	5-10
5-10.	Archive Group Window	5-11
5-11.	Historical Collection Storage Options Window	5-13
5-12.	Historical Tag Configuration Window with Default Settings	5-17
6-1.	Log Configuration Window	6-2
6-2.	Event Log Configuration Window	6-3

List of Figures (continued)

No.	Title	Page
6-3.	Event Log Scheduling Window	6-7
6-4.	Printer Selection Window	6-8
6-5.	Print Event Log Window	6-9
7-1.	Trend Index Entry Pop-up Window	7-2
7-2.	Select Trend Pop-up Window	7-2
7-3.	Historical Trend Display (Template)	7-4
7-4.	Trend Configuration Window	7-5
7-5.	Trend Trace Configuration Window	7-7
8-1.	System Status Display	8-1
8-2.	SSD Object for a PWC90 Console	8-2
8-3.	Network Info Window Example	8-4
8-4.	Services Window Example	8-5
8-5.	PWC90 Node SSD Icon	8-5
8-6.	SSD Object for a Server	8-6
8-7.	Device Status Window	8-7
8-8.	Printer Queue Control Window	8-8
9-1.	PWC90 Utilities Menu	9-1
9-2.	Database Maintenance Menu	9-3
9-3.	Historical Database Maintenance Window	9-3
9-4.	Archival Database Maintenance Window	9-7
9-5.	CRT Print Setup Window	9-9
9-6.	PWC90 System Time Adjust Window	9-11
9-7.	Default Quick Keys Keypad Display	9-14
9-8.	Quick Key Label Assignment Window	9-15
9-9.	Quick Key Pop-up with Text String Assignments	9-16
9-10.	CRT Context Keys Keypad	9-19
9-11.	CRT Context Key Assignment Window	9-20
10-1.	Historical Block Database Configuration Window	10-2
10-2.	Historical Block Configuration Window	10-3
10-3.	Storage Nodes Window	10-4
10-4.	Historical Tag Computations Window	10-7
11-1.	@aGlance Configuration Window	11-5
11-2.	@aGlance Current Tagnames Window	11-6
11-3.	@aGlance Add Tagname Window	11-6
11-4.	@aGlance Edit Tagname Window	11-8
11-5.	@aGlance History Tagnames Window	11-9
11-6.	@aGlance Servers Window	11-11
11-7.	@aGlance Server View Window	11-11
11-8.	PWC90 Configuration Data Window from the Backup/Restore Subsystem.	11-13
12-1.	Available Tag List Window	12-3
12-2.	General Tab Window	12-9
12-3.	INFI 90 Tab	12-11
12-4.	Digital Tab Window	12-12
12-5.	Remote Control Memory Window	12-13
12-6.	Sort Window	12-16
12-7.	Tag List Filtering Window	12-17
12-8.	Trend Importer Default Window	12-18
12-9.	Tag Importer Default Window	12-20
12-10.	Tag Importer Window with Check Boxes	12-20
12-11.	Index Text Window	12-22
12-12.	Substitutable Text Tab Page	12-27
12-13.	RCMB Text Tag	12-28

List of Figures (continued)

No.	Title	Page
B-1.	Dynamic Bargraph (FP_DYbargraph)	B-2
B-2.	Control Pushbutton (FP_DYctrlpb)	B-5
B-3.	Dial Submodel (FP_DYdial)	B-6
B-4.	Enter Text Pushbutton Submodel (FP_DYentertextpb)	B-7
B-5.	Enter Value Pushbutton (FP_DYentervaluepb)	B-8
B-6.	Group Display Submodel (FP_DYgroupmod)	B-9
B-7.	Historical Trend Model (FP_DYHisttrend)	B-10
B-8.	Dynamic Horizontal Bar (FP_DYHorizbar)	B-11
B-9.	Horizontal Trend Submodel (FP_DYHoriztrend)	B-12
B-10.	Horizontal Slider Submodel (FP_DYhslider)	B-14
B-11.	Meter Submodel (FP_DYmeter)	B-15
B-12.	Model Pop-up Pushbutton (FP_DYmpoppb)	B-16
B-13.	Point Display Submodel (FP_DYpointmod)	B-17
B-14.	Dynamic Pump - Left (FP_DYpump_lf)	B-18
B-15.	Dynamic Pump - Right (FP_DYpump_rt)	B-20
B-16.	Put1 Pushbutton Submodel (FP_DYput1pb)	B-22
B-17.	Put Pushbutton Submodel (FP_DYputpb)	B-23
B-18.	Putpbstat Submodel (FP_DYputpbstat)	B-25
B-19.	FP_DYputpbstat2 Submodel	B-27
B-20.	Statpb Submodel (FP_DYstatpb)	B-29
B-21.	Tagname Pushbutton (FP_DYtagpb)	B-30
B-22.	Dynamic Tank Submodel (FP_DYtank)	B-31
B-23.	Dynamic Tank Submodel with Configurable Erase Color (FP_DYtank_ec)	B-32
B-24.	FP_DYtoggle Submodel	B-34
B-25.	Historical Trend Pushbutton (FP_DYtrendpb)	B-36
B-26.	Dynamic Value (FP_DYvalue)	B-37
B-27.	Vector Pushbutton Submodel (FP_DYvectorpb)	B-39
B-28.	Dynamic Vertical Bar (FP_DYVertbar)	B-40
B-29.	Vertical Trend Model (FP_DYVerttrend)	B-41
B-30.	Vertical Slider Submodel (FP_DYvslider)	B-43
B-31.	Dynamic Wheel (FP_DYwheel)	B-44

List of Tables

No.	Title	Page
1-1.	Reference Documents	1-5
1-2.	Terms and Abbreviations	1-6
2-1.	Priority Color Assignments	2-19
2-2.	Default Field Widths	2-22
2-3.	Default Device Characteristics	2-36
2-4.	Default Function Access Level Assignment	2-54
2-5.	Server and Tag Access Fields	2-56
3-1.	Group Status Conditions	3-4
4-1.	Bailey Supplied Submodel Types	4-36
4-2.	View Size Information	4-44
4-3.	Operator Usage - Description and Examples	4-54
4-4.	Operators In Order Of Precedence	4-55

List of Tables (continued)

No.	Title	Page
4-5.	Operator Aliases	4-56
4-6.	Attribute Values Table	4-60
4-7.	Blink Settings	4-79
4-8.	Input Conditions	4-80
4-9.	Expressions and Settings	4-80
4-10.	Blink Settings	4-80
4-11.	I90 Faceplate Submodels	4-91
5-1.	Historical Computation Atoms	5-8
7-1.	Historical Computation Atoms	7-8
10-1.	Historical Computation Atoms	10-6
11-1.	Options for Use When Starting a Server	11-4
11-2.	Supported @aGlance/IT Methods	11-13
11-3.	Default Attribute Name	11-14
11-4.	AAG_StatusList Information	11-17
11-5.	Supported Monitor Type Identifier Strings	11-20
12-1.	Tag Types	12-2
12-2.	Tag Database - General Tab Fields	12-5
12-3.	INFI 90 Tab Fields	12-5
12-4.	Tag Database - Analog Tab Fields	12-5
12-5.	Tag Database - Data Aqu. Analog Tab Fields	12-6
12-6.	Tag Database - Digital and Rem Ctrl Memory Tab Fields	12-6
12-7.	Tag Database - Station Tab Fields	12-7
12-8.	Device Driver Tab Fields	12-7
12-9.	Multi-State Device Driver Tab Fields	12-7
12-10.	Rem Motor Ctrl Block Tab Fields	12-8
12-11.	Tag Database - RMSC Tag Fields	12-8
12-12.	Tag Database - TEXTSTR Tag Fields	12-8
12-13.	INFI 90 OPEN Address Values	12-12
12-14.	Operating Status Descriptions	12-14
12-15.	Engineering Unit Descriptors	12-24
12-16.	Logic State Descriptors	12-25
12-17.	Alarm Priorities	12-26
A-1.	INFI 90 Analog	A-1
A-2.	INFI 90 Analog Base	A-1
A-3.	INFI 90 Analog Export	A-2
A-4.	INFI 90 ASCII STRING	A-2
A-5.	INFI 90 Base	A-2
A-6.	INFI 90 DAANG	A-2
A-7.	INFI 90 DADIG	A-4
A-8.	INFI 90 Device Driver	A-4
A-9.	INFI 90 Device Driver Base	A-4
A-10.	INFI 90 Digital	A-5
A-11.	INFI 90 Digital Base	A-5
A-12.	INFI 90 Digital Export	A-5
A-13.	INFI 90 Export Base	A-5
A-14.	INFI 90 Import Base	A-5
A-15.	INFI 90 MODULE STATUS	A-5
A-16.	INFI 90 MSDD	A-6
A-17.	INFI 90 RCM	A-6
A-18.	INFI 90 RCM Base	A-7
A-19.	INFI 90 RMCB	A-7
A-20.	INFI 90 RMSC	A-7

List of Tables (continued)

No.	Title	Page
A-21.	INFI 90 Server	A-7
A-22.	INFI 90 Station	A-9
A-23.	INFI 90 Text Tag	A-9
A-24.	INFI 90 UDXR Base	A-10
A-25.	Server Base	A-10
A-26.	Tag Base	A-11
C-1.	Sample Type Atoms	C-6
C-2.	Cell Address and Contents	C-13
C-3.	Cell Address And Contents	C-14
C-4.	Data Type and Syntax	C-21
C-5.	Cell Address And Contents	C-22

Trademarks and Registrations

Registrations and trademarks used in this document include:

™ @aGlance/IT	Trademark of Intuitive Technology Corporation.
® INFI 90	Registered trademark of Elsag Bailey Process Automation.
® INFI-NET	Registered trademark of Elsag Bailey Process Automation.
™ Pentium	Trademark of INTEL corporation.
™ Windows NT	Trademark of Microsoft corporation.

SECTION 1 - INTRODUCTION

INTRODUCTION

The Personal Work Center 90 (PWC90) is an operating station for use with the INFI 90 OPEN system and other compatible equipment. The ***Personal Work Center 90 Configuration*** instruction describes the operations required to configure the PWC90 console. It is designed to accomplish the following objectives:

- Provide instructions on how to perform configuration tasks on the PWC90 console.
- Provide suggestions and examples for configuration activities.
- Provide additional information as necessary.

INTENDED USER

The ***Personal Work Center 90 Configuration*** instruction is intended for use by personnel engaged in the configuration of Bailey INFI 90 OPEN equipment. The PWC90 console communicates with the INFI 90 OPEN system via an ethernet data highway network. Familiarity with the INFI 90 OPEN system is necessary to understand the functions of a PWC90 console.

FUNCTIONAL DESCRIPTION

INFI 90 OPEN system is a distributed process control and information management system which provides the capability to divide process functions among many system components to provide operating flexibility, system reliability, reduced system costs and ease of implementation. Distribution combined with redundancy also allows system control and operating functions to be divided among system elements to increase reliability and flexibility and to reduce risk.

Automation of most processes requires more than control at the unit operations level. The system must also include an inherent ability to provide process management functions such as scheduling and graphical presentation of plant process conditions as well as reporting functions. The PWC90 console, process control units (PCUs), field wiring interface terminal boards, and Batch 90 process control software form the basic building blocks of the INFI 90 OPEN Distributed Control System. The PWC90 console is the human/machine interface to this system, providing the process operator, process engineer, instrument engineer and maintenance personnel with global

access to all process and system parameters required by each to perform their respective tasks.

The PWC90 console provides the process operator with a window to the process. Using interactive process graphics and hierarchical displays, the process operator can monitor and control all analog loops and digital devices interfaced to the network, as well as sequential batch operations.

The PWC90 console provides the engineer with an interface through which to configure and change graphic and hierarchical displays, database I/O, Log (report) formats, and security features (including access levels for operating personnel). Most changes are immediate, on-line and network-wide, requiring no compilation time before downloading and therefore no interruption of the process.

The PWC90 console provides maintenance personnel with the capability to globally monitor the operating status of any system component on the network, and to diagnose component failures from any PWC90 console.

User Interface

The PWC90 console provides the process operator, engineer and maintenance technician with a window based interface to both the process and the PWC90 console itself.

The PWC90 console uses Windows NT from Microsoft as an operating system. The user interface is described in detail in ***Personal Work Center 90 Operation*** instruction.

PWC90 Console Management Features

The PWC90 console allows the user to configure many items which aid in managing use of the system. These include:

- Security features.
- User logins.
- Network parameters.
- Archival groups.
- Messages.
- Message routing.
- Network device assignment.

These items are described in Section 2.

Hierarchical Displays

Hierarchical displays are a set of preconfigured, network wide graphical representations emulating traditional instrument displays. A three tiered hierarchy of displays exists to provide

the user with information about the process. This hierarchy consists of summary, group and point displays.

A summary display provides an overview of 24 groups, arranged in six rows of four blocks. There are 128 summary displays available.

A group display provides an operation overview of four, six or eight points simultaneously. All process control actions and alarm acknowledgment for points within a group can be performed from the group display on a point-by-point basis.

A point display provides the most detailed information and operation functions about a single point in the hierarchical display system. From this level the operator can perform regulatory changes. Hierarchical display configuration is discussed in Section 3.

Graphic Configuration

Up to 1024 network-wide graphic displays can be configured using the PWC90 console. Because they are network wide, a graphic display can be configured on any PWC90 console; a copy of the file containing the graphic is then distributed to all PWC90 consoles on the network whenever a graphic is saved or installed; the graphic can then be viewed on any PWC90 console. Graphic configuration is discussed in Section 4.

Historical Database

The historical database provides the means to collect data from the PCU(s) and store it for use by various other applications on the PWC90 console, such as logging, trending and data archiving. Historical database configuration is discussed in Section 5.

Logging

Logs are used to collect and format data for use in reports and spreadsheet calculations. The two types of logs available on the PWC90 console include event logs and spreadsheet logs. Logs are available network wide. Log configuration is discussed in Section 6.

Trending

Trending functions display the values of one to eight points in an analog trend format similar to that presented by a conventional strip chart recorder. Three types of trend recording are provided on the PWC90 console, including current trending, historical trending and archival trending. Trend displays are network wide. Trend configuration is discussed in Section 7.

System Status Display

The system status display provides an overview of the nodes connected to the INFI 90 OPEN network. Nodes include PWC90 consoles and INFI 90 OPEN server nodes. The system status display is described in Section 8.

PWC90 Console Utilities

A number of useful maintenance and information management functions are provided as PWC90 console utilities. Utilities discussed in this configuration instruction include:

- CRT print.
- Release all configuration locks.
- PWC90 help.
- User help.
- Quick keys.
- CRT context keys.

These utilities are discussed in Section 9.

Historical Block Data Collection

The historical block database is designed for the efficient collection of data from a large number of INFI 90 OPEN tag.atoms. Like the data collected in the historical database, the historical block data is stored for use by other applications, such as logging, trending and data archiving. Historical block configuration is discussed in Section 10.

@aGlance

The @aGlance/IT server interface is an optional utility which provides a live data link between the PWC90 console and a variety of software applications resident on other distributed control systems, supervisory control systems, personal computers, and so on. For more information refer to Section 11.

Tag Configuration

The tag configuration application permits an operator to configure the database or make adjustments to the database from the PWC90 console. Section 12 explains the use of the tag configuration application.

Tag.Atom List

A list of tag.atom names and configurable parameters is provided to aid in configuration of the PWC 90 console. This list is located in Appendix A at the back of this instruction.

Bailey Submodels

Submodels are included in the PWC90 software for ease of graphic configuration. For a description of these graphic submodels refer to Appendix B.

EXCEL Functions

There are several EXCEL functions available with the PWC software which provide an interface to the Microsoft EXCEL spreadsheet package. For a description of all of these functions refer to Appendix C.

DOCUMENT CONVENTIONS

This document uses standard text conventions throughout to represent keys, user data inputs, and display items:

KEY	Identifies a keyboard key.
Example:	Press ENTER.
USER INPUT	Indicates a fixed input that must be entered exactly as shown.
Example:	Type **FIRMWARE.**
Display item	Any item that displays on the screen appears as italic text.
Example:	*Restore File* button. *User Administrator logged in on PWC1*.
File name	Any file names and file extensions appear in bold italics.
Example:	***query1.qry***

REFERENCE DOCUMENTS

This instruction provides information only for the configuration of the PWC90 console. Table 1-1 list additional documents that relate to the operation of the INFI 90 OPEN system.

Table 1-1. Reference Documents

Number	Document Title
WBPEEUI220755A0	Personal Work Center 90 Operation Instruction
WBPEEUI220763A0	Personal Work Center 90 Installation Instruction
I-E96-200	Function Code Application Manual

GLOSSARY OF TERMS AND ABBREVIATIONS

Table 1-2 contains those terms and abbreviations that are unique to Elsag Bailey or have a definition that is different from standard industry usage.

Table 1-2. Terms and Abbreviations

Term	Definition
Atom	A single data element in an INFI 90 OPEN server tag. Atoms can exist in various forms, including (but not limited to) a single bit (e.g., Auto/Manual), a floating point number, a tag-name or a multi-character legend; there is no predefined limit to the size of an atom. See also Module.
Banner	A header page which precedes a printout, which contains identifying. This is especially useful when the user initiating the printout is not in the immediate vicinity of the printer.
Closed Loop Control System	A control system in which the controlled variable is measured and compared with a standard (set point) representing the desired performance. From this comparison (feedback), the system compensates to eliminate any deviation between the set point and the controlled variable.
Control Module	Directs field processes through an I/O module; the multi-function processor is an example.
DCS	Digital control station. A panel mounted operator device that provides monitoring and allows manipulation of a single process control loop.
Demand Output	The desired control output of the analog control station when it is in bypass operation.
Device Driver	An INFI 90 OPEN/Network 90 system function code that provides control for a single input device.
Display Element	A digital element used in creating a process display; station faceplate, annunciator, trend, and deviation overview are display element examples.
EUD	Engineering unit descriptor.
Exception Report	Information update generated when the status or value of a point changes by more than a specified significant amount; abbreviated as XR.
Fatal Error	An error that causes the device to go into a fail mode and send its output to a defined value.
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.
GMS	(Graphical Modeling System) A PWC90 System application program licensed from SL Corporation. GMS is used to create PWC90 System graphical displays. Each PWC90 System incorporates an RT (runtime) license to utilize GMS models.
INFI-NET	Advanced data communication highway.
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.
LSD	Logic state descriptor. A character string which is associated with the state of a digital point.
MFC	Multi-function controller module. A multiple loop controller with data acquisition and information processing capabilities.

Table 1-2. Terms and Abbreviations (continued)

Term	Definition
MFP	Multi-function processor module. A multiple loop controller with data acquisition and information processing capabilities.
Mini-Alarm Window	The Mini-Alarm Window is located to the right of the System Window on the PWC90 console screen. The Mini-Alarm Window automatically appears upon system start-up and is visible at all times. The Mini-Alarm Window provides a title bar, an alarm display area, an event display area, Alarm Acknowledge buttons to acknowledge alarms and events, and a Filter button and More Alarms indicator.
Model	When used in the graphical context of GMS, a model is an object or group of objects. Models can be used as objects in other models.
Module Address	A unique identifier of a specific device or a communication channel. Refers to Controlway or module bus address.
Module Bus	Peer-to-peer communication link used to transfer information between intelligent modules within a process control unit.
MSDD	Multi-state device driver; provides control for a dual input device.
Node	A point of interconnection to a network.
Node Address	A unique identifier of a specific device or a communication channel. Refers to Plant Loop, Superloop or INFI-NET address.
Object	When used in the graphical context of GMS, an entity such as a circle, rectangle, or text field.
Operator Window	Located on the PWC90 System console screen below the system window and Mini-Alarm window, the Operator Window provides the operator interface to the process, and displays program output from Operator Window applications such as hierarchical displays, trends, graphics, alarm review, etc. The Operator Window is composed of a title bar, a menu bar, and a display area. Up to four Operator Windows may be present simultaneously on a PWC90 console.
Primary Module	Active module in a redundant configuration.
Personal Work Center (PWC)	The Personal Work Center is a Windows NT-based system which supports the PWC90 System software.
Process Area	A functional partition of the process. Tags are assigned to and often located in a particular Process Area. PWC90 functions can be assigned by Area.
Quick Key	A menu button on the PWC90 console screen which, when pressed, provides the user with the ability to vector directly to one of 64 possible displays.
Ratio Index	An analog input to a station function block.
Ratio Station	An analog control station configuration that maintains a ratio between two variables. The ratio of an externally generated wild variable times the controlled variable becomes the set point that provides regulation of a third variable. One of three station types available in the normal operating mode.
Red Tag	A logical flag that indicates a point is out of service.
RMSC	Remote manual set constant; a function code which generates exception reports consisting of a status and an analog value.
Secondary Module	Backup module in a redundant configuration.
Specifications	Function block input values that determine function block characteristics.
Symbol	A drawing element. Using symbols for commonly used drawing elements eliminates redrawing.

Table 1-2. Terms and Abbreviations (continued)

Term	Definition
System Window	A small window located at the top left corner of the PWC90 console screen. It appears automatically upon system start-up, and is visible at all times. The system window provides a date/time indicator and a menu button for access to other system features and utilities.
Tag	An analog or digital process value defined in the console as a control or monitor point.

SECTION 2 - SYSTEM CONFIGURATION FUNCTIONS

INTRODUCTION

Each Personal Work Center (PWC90) computer contains parameters which must be configured to achieve optimum performance locally and to allow the unit to function effectively on the INFI 90 OPEN system. This chapter provides basic information required to get started using the system, then moves on to discuss system configuration functions, including definition of archival groups, message configuration, security, and network parameter configuration. Network parameters include the system title, alarm setup, assignment of area names and event bit names, printer configuration, message configuration, and network device assignment.

GETTING STARTED

One of two states can exist on a PWC90 console that is running:

- System default state (no user logged in).
- User logged in state.

System Default State

The system default state occurs after a PWC90 console initial startup, after a user has logged out of a PWC90 console, or when a PWC90 console is restarted. When the system default state exists:

- The security access level, Quick keys and CRT context keys are those defined in the default login setup.
- The username in the title bar of the mini-alarm window is *Default*.
- The displays contained in the operator windows are those defined in the initial login displays of the default login setup. If no initial login displays are defined for the default login, a single operator window is opened, which contains the system status display.

When the system default state exists, the system responds as though a user named *Default* is logged in on the system. Depending on the security level assigned to this default user, various activities can be performed on the system, such as opening operator windows, viewing displays or printing the contents of the screen.

When the PWC90 console is started initially and no security levels have yet been assigned, the default user is automatically assigned to the highest security level available (Supervisor level 3). This allows the user performing the initial system configuration to create user login accounts and assign to them appropriate access levels. The default user account can then be re-assigned to a lower access level. System security, including the concept of access levels, is discussed in ***SECURITY*** in this section.

Logging In

The user logged in state exists when a user (other than default) has logged in. To login, click on the system window *Menu* button, then select the *User* option from the system window pull down menu. A second menu cascades from the first, as shown in Figure 2-1.

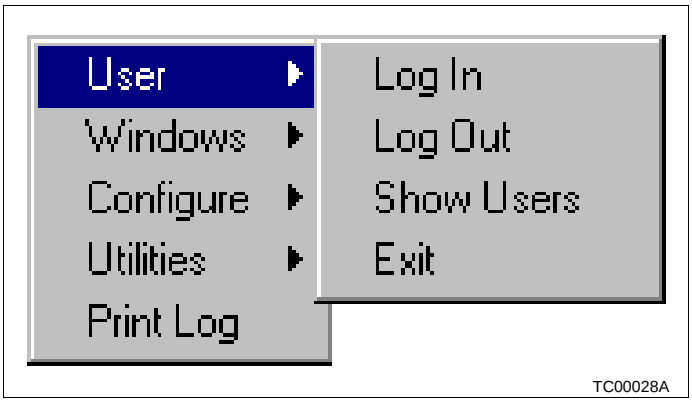


Figure 2-1. System Window Menu with Cascading Login/ Logout Menu

This menu allows the user to login to (or logout of) the PWC90 console. The security system of access levels controls the user's access to PWC consoles, areas and priorities as well as network and INFI 90 OPEN server functions. When active, the user's access to these parameters is defined by the system default or by the user's login configuration. When the security system is inactive (a user selectable state, refer to ***Security Activation*** in this section), no restrictions are placed on the user's access abilities.

To login, click on the login button; the PWC90 console *Login* window appears, as shown in Figure 2-2.

In the entry box next to the *Login* prompt enter the login name of the user to be logged in, using either the PWC90 console keyboard or by clicking on the appropriate keys on the video keyboard which appears in the *Login* window. The login name should be followed by Return (or by clicking on *OK*). Next, enter

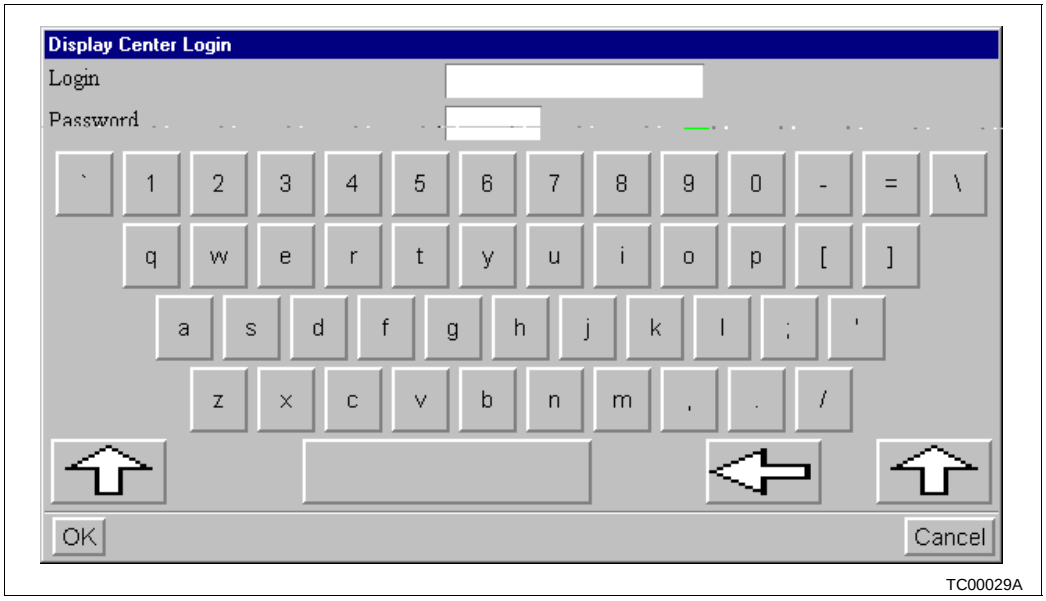


Figure 2-2. PWC90 Console Login Window

the password in the same manner, again followed by Return (or by clicking on *OK*).

The operation can be aborted at any time prior to completion. To do so, click on *Cancel* which appears on the PWC90 console *Login* window.

> **NOTE:** If an incorrect login name or password is entered during a login attempt, a system event message will be generated and a pop up window containing an OK button and the message *Login failure. Try again?* will appear.

Click on *OK* to close the pop up window and return to the PWC90 console *Login* window.

The initial displays assigned in the user's login configuration are placed on the PWC90 console used for the login operation. The Quick keys (refer to ***QUICK KEYS*** in Section 9 and ***CRT CONTEXT KEYS*** in Section 9) assigned in the user's login configuration are available to the user. The user's name is displayed in the title bar of the mini-alarm window.

A user can login on a system which already has an active login on it. This action causes the previous user to be logged out. If the new user has defined an initial CRT context (i.e., CRT Login Context Key), the screen will display this new environment. If no initial screen context has been defined, the displays on the screen remain unchanged.

Logging Out

When a user logs out of PWC90 console, the following actions take place only on the PWC90 console from which the user is logging out:

- The security access level, Quick keys and CRT context keys change to those defined for the default user.
- The username in the title bar of the mini-alarm window changes to *Default*.
- The login context changes to that of the default user, and the login parameters will change to those of the default user. (In other words, when an actual user logs out, the PWC90 console returns to the default login state; it is as if the default user had logged in.)

The logout procedure is very similar to that used to login. To logout, click on the system window *Menu* button, then select the *User* option from the system window pull down menu. A second menu cascades from the first, as shown in Figure 2-1. Click on *Log Out*; the current user will be logged out.

ARCHIVAL GROUPS

The PWC90 console can be configured to collect data from the INFI 90 OPEN database and place it in the historical database (refer to ***HISTORICAL DATABASE*** in Section 5), which resides on the PWC90 console. Eventually, the collected data is aged out of the historical database and the disk space it occupied is overwritten by new data. Data collected in the historical database can be preserved by placing it (archiving it) onto a removable storage device.

A set of data is configured to be archived by creating and scheduling an archival group through archival group configuration. Historical database points are then assigned to a configured archival group through historical database point configuration (refer to ***ADD DATA STORAGE DEFINITIONS*** in Section 5). At a user configurable cyclic interval, a message will be broadcast indicating that the data for an archival group is ready to be archived. The data associated with the archival group can be manually archived or restored through the system window archival utilities.

Access archival group configuration by using the pull down menus available from the system window *Menu* button. Click on *Menu*; from the resulting pull down menu, click on the *Configure* option, then click on *Archival Groups* in the resulting cascading menu. The *Archival Group Configuration* window (Figure 2-3) will be displayed.

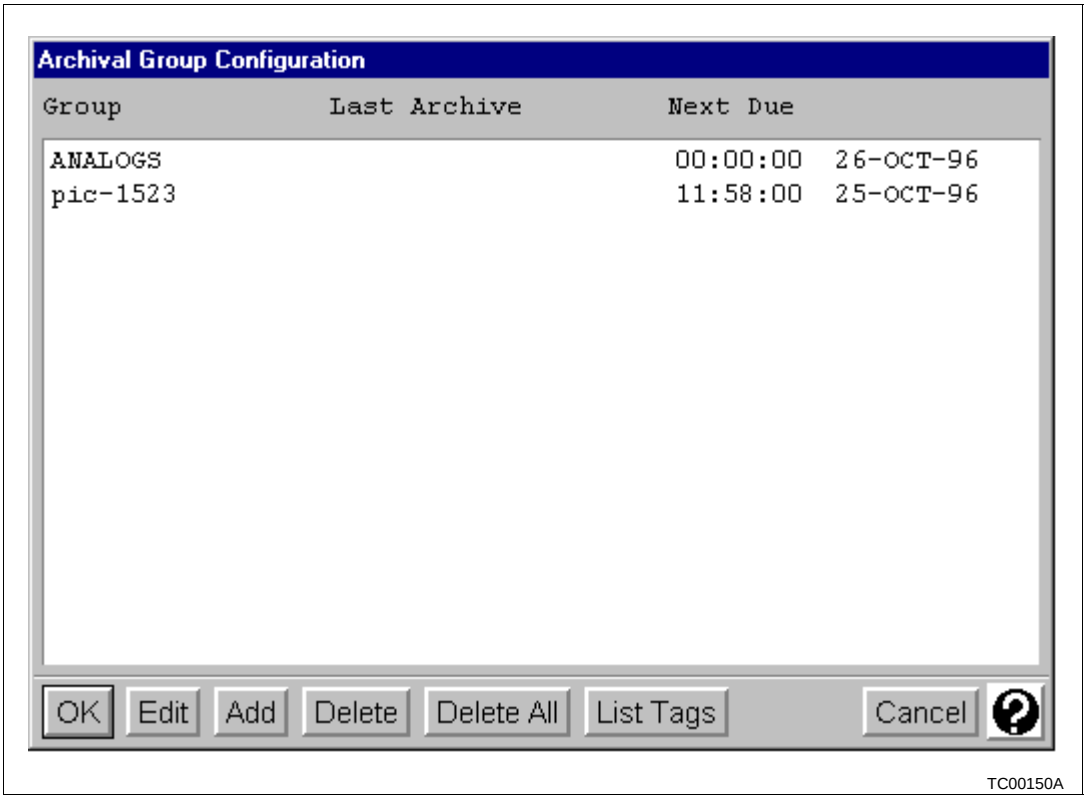


Figure 2-3. Archival Group Configuration Window

The top area of this window contains a scrollable list of all archival groups which have been configured. Each entry in the list contains the *Group* name, the time and date the group was *last archived*, and the time and date that the group is *Next Due* to be archived. Below this list are buttons which are used to *Add, Edit, Delete* and *Delete All* the archival groups.

Edit Archival Group

To make changes to an existing archival group, click on the desired archival group name (in the list box), then click on *Edit*. The *Archival Group Scheduling* window (Figure 2-4) will be displayed.

This window allows the following parameters to be changed:

- Group name.
- Period and start time.
- Area.

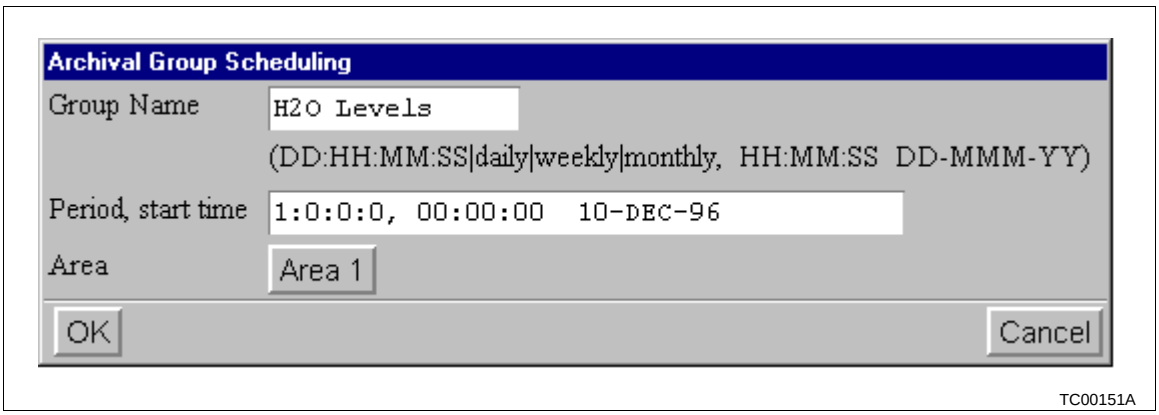


Figure 2-4. Archival Group Scheduling Window

GROUP NAME

Click on the *Group Name* entry box to gain focus on it, then enter the name which is to be used to identify the archival group. This name can be up to 15 characters in length.

PERIOD AND START TIME

The period and start time allow for the scheduling of the data archive for the archival group. Both the cyclic archival *Period* (user defined archive interval) and the *Start Time* (which defines the beginning of the *Period*) are entered in the same entry box.

The *Period* is entered first; the words *Daily, Weekly* and *Monthly* are valid entries to define the *Period*. Alternatively, the user can define a customized *Period* by defining the number of days, hours, minutes and seconds in the *Period* using the following format:

DD:HH:MM:SS

Enter a comma (,) after the *Period* to separate it from the *Start*[ing] *Time*. After the *Period* has been defined, the *Start*[ing] *Time* (hours, minutes, seconds) and date (day, month, year) which marks the beginning of the interval, is entered in the following format:

HH:MM:SS DD-MMM-YY

The default *Period* is daily (1:0:0:0) and the default *Start Time* is midnight of the current day (i.e., 00:00:00 30-May-96). Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name.

An acknowledgable message will be broadcast at the scheduled time indicating that the archival group is ready to be archived.

This message is not repeated until the next scheduled data archive time for that group.

AREA

Each archival group can be associated with a specific plant area. The *Area* assignment is used to route the message to the devices (files, printers) which have been configured (via message routing configuration, refer to ***Message Routing*** in this section) to receive messages associated with that area. The message will also be sent to all PWC90 consoles where users who have access to that area are logged in. On the PWC90 console consoles, the message will be displayed in the message line (bottom line) of the mini-alarm window. To assign the operator message to an area, click on the *Area* button. The *Area Selection* window (Figure 2-5) will be displayed.

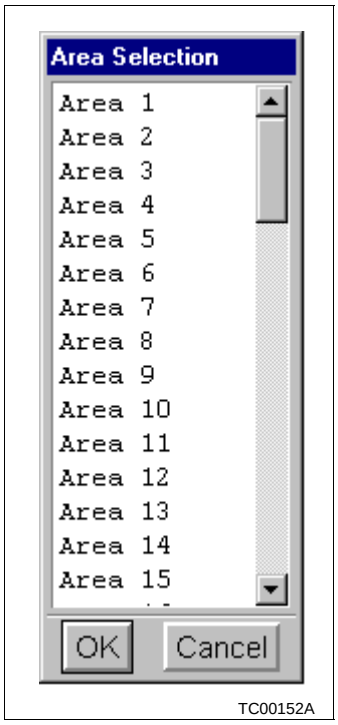


Figure 2-5. Area Selection Window

The *Area Selection* window contains a scrollable list of 64 areas. Click on the desired area, then click on *OK* to complete the selection operation. (Click on *Cancel* to exit without changing the *Area* selection.) The default is *Area 1*.

Add Archival Group

To create a new archival group, click on *Add* in the *Archival Group Configuration* window. The *Archival Group Scheduling*

window (Figure 2-4) will be displayed with default values. Configure as described in ***Edit Archival Group*** in this section.

Delete Archival Group

The *Delete* button is used to delete an archival group. In the list box, click on the name of the group to be deleted; it will be highlighted. Click on *Delete* to delete the highlighted group.

Delete All Archival Groups

To delete all archival groups, click on *Delete All*. A confirmation box will appear. Click on *Yes* if all archival groups are to be deleted; click on *No* to exit without deleting archival groups.

List Tags in an Archival Group

To view a list of the tags in an archival group, first select the desired group in the list displayed in the *Archival Group Configuration* window, then click on *List Tags* at the bottom of the window. A pop up window titled *List Tags* will be displayed, which contains a list of all the tagnames in the selected archival group. The button also contains an *OK* button. Click on *OK* to close the pop up window.

Exiting Archival Group Configuration

Click on either *OK* or *Cancel* to exit archival group configuration and close the *Archival Group Configuration* window.

MESSAGE CONFIGURATION

A method of communication between operators is available on the PWC90 console, allowing text messages known as operator messages to be sent from one plant area to another. The message configuration feature provides this method of generating operator messages from a PWC90 console.

To create an operator message, access message configuration by using the pull down menus available from the system window *Menu* button. Click on *Menu*; from the resulting pull down menu, click on the *Configure* option, then click on *Message Configuration* in the resulting cascading menu. The *Message Configuration* window (Figure 2-6) will be displayed.

The window is used to configure the text, priority and destination of the operator message.

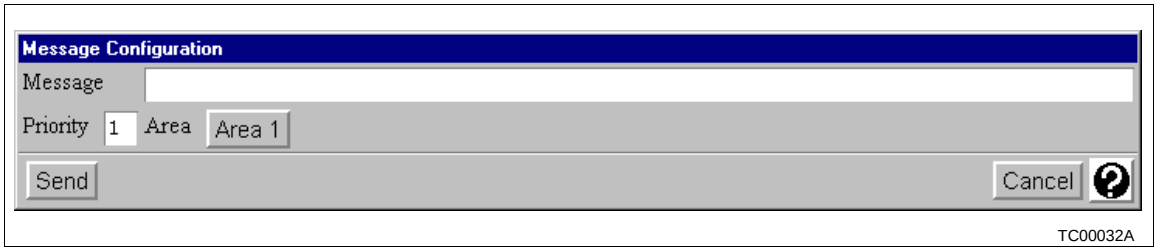


Figure 2-6. Message Configuration Window

Message Text

To configure the text of the message, click on the text entry box located at the top of the *Message Configuration* window to gain focus on it, then enter a message of up to 80 ASCII characters.

Message Priority

Each message is associated with a priority. To set the priority, click on the *Priority* text entry box and enter the number of the desired priority (range of 1 - 16).

Message Destination Area

Each message can be associated with a specific plant area. The *Area* assignment is used to route the message to the devices (files, printers) which have been configured (via message routing configuration, ***Message Routing*** in this section) to receive operator messages associated with that area. The message will also be sent to all PWC90 consoles where users who have access to that area are logged in. On the PWC90 consoles, the message will be displayed in the message line (bottom line) of the mini-alarm window. To assign the operator message to an area, click on the *Area* button. The *Area Selection* window (Figure 2-5) will be displayed.

The *Area Selection* window consists primarily of a scrollable list box which contains 64 areas. To select the destination area for the operator message, click on the desired area, then click on *OK*. The default is *Area 1*. (To exit without changing the *Area* assignment, click on *Cancel*.)

Send the Operator Message

To send the operator message and exit *Message Configuration*, click on *Send*. To exit without sending the message, click on *Cancel*.

ALARM GROUP CONFIGURATION

Alarm groups can be used to change the state of atoms in the INFI 90 OPEN server, including changes which cause momentary contacts to be energized for a predetermined cycle, such as an on/off switch which is controlled by a Digital I/O Slave Module in the PCU. Typically, these are used to trigger an audible or visual alarm at an annunciator panel.

Alarm group operation is initiated when process alarms are detected. Alarm group configuration provides the ability to filter process alarms by message type and class. All process alarms are evaluated against the *Filter* configuration of the alarm group; if the message is filtered out, that group is ignored and the next alarm group is checked. If the process alarm is not filtered out and the alarm is unacknowledged, then the alarm group is considered to be on. When this condition exists, the *On list* (a list of tag.atoms and associated values) is downloaded to the INFI 90 OPEN server. When all alarms for a particular group are acknowledged, the group is considered to be off, and a different list of tag.atoms and values, known as the *Off list* is processed.

Accessing Alarm Group Configuration

Alarm group configuration is accessed by using the pull down menus available from the system window *Menu* button. Click on *Menu*, then click on the *Configure* option in the resulting pull down menu. Click on *Alarm Groups* in the resulting cascading menu. The *Alarm Group Selection* window (Figure 2-7) will be displayed.

In the top portion of this window is the *Alarm Group Selection List*. This list contains the name of each alarm group which has been configured. At the bottom of the window are buttons labelled *Add, Delete* and *Edit,* as well as the usual *OK*, *Cancel* and *Help* buttons.

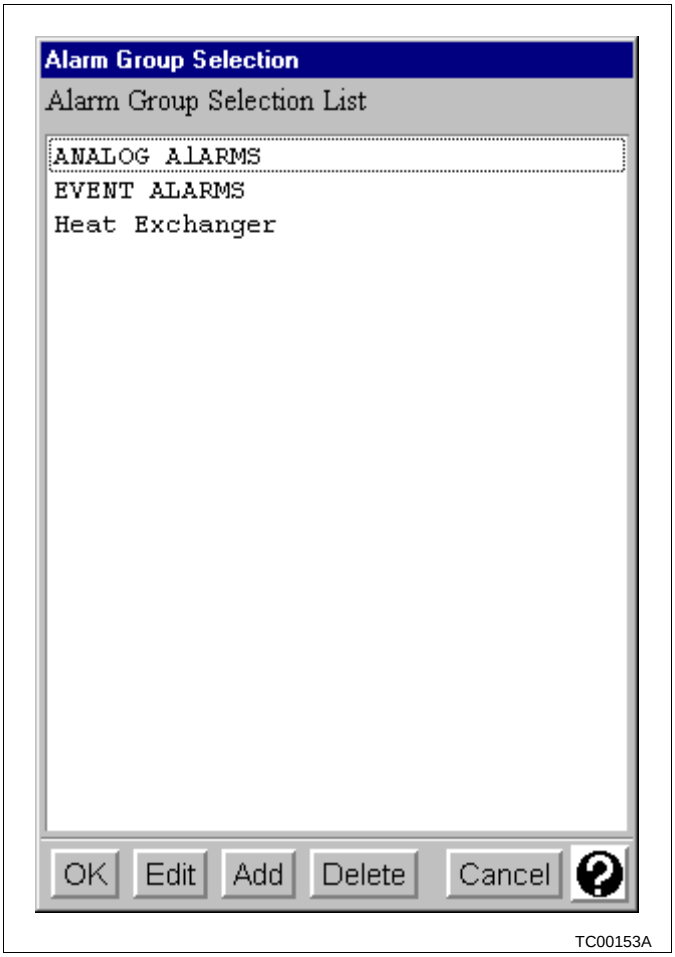


Figure 2-7. Alarm Group Selection Window

Adding an Alarm Group

To create an alarm group, click on *Add* at the bottom of the *Alarm Group Selection* window. The *Alarm Group Configuration* window (Figure 2-8) will be displayed.

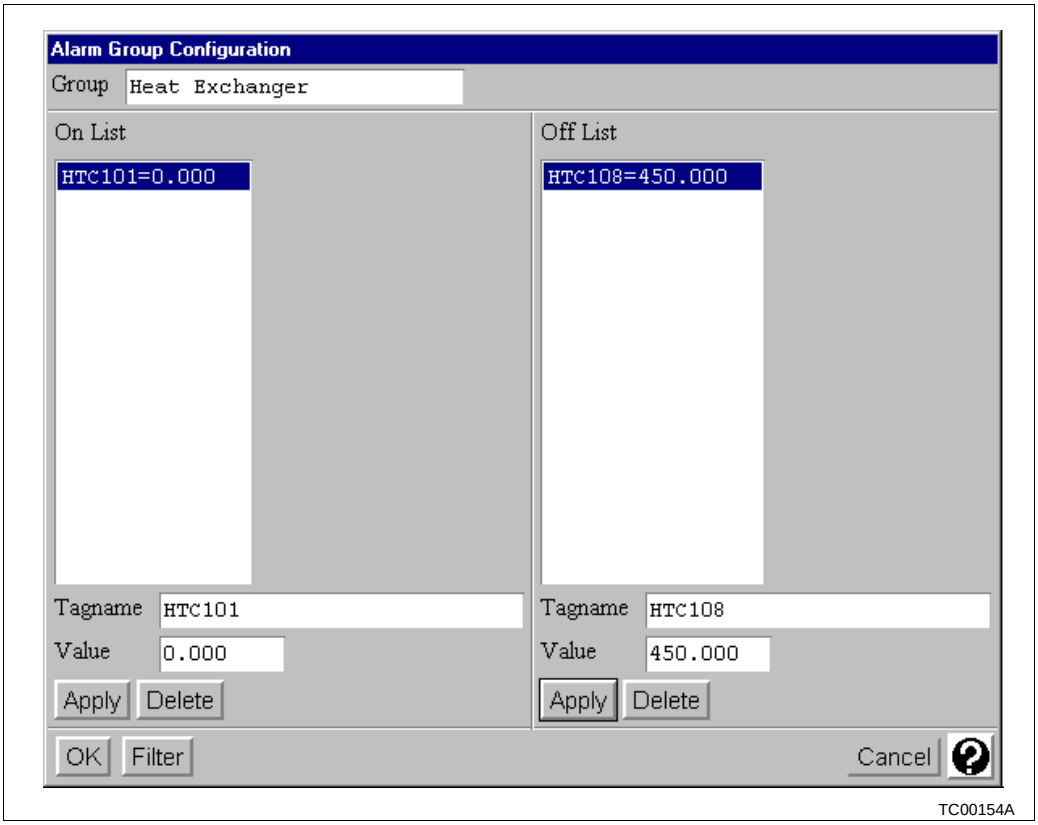


Figure 2-8. Alarm Group Configuration Window

GROUP NAME

The first item to appear in the *Alarm Group Configuration* window is a text entry box labelled *Group Name*. Enter a string of up to 32 alphanumeric characters to be used as the name of the alarm group. Once the configuration of the alarm group has been completed, this name will appear in the *Alarm Group Selection List* found in the parent window (Figure 2-7).

ON LIST

Below the *Group Name* entry box, the main area of the *Alarm Group Configuration* window is divided in half vertically. The left side is used to configure and display the *On List*, while the right side is used to configure and display the *Off List*. The *On List* is a scrolled list of tagname.atoms and values; when a process alarm passes through the filter, these are downloaded to the INFI 90 OPEN server and are typically used to trigger the contacts used on the alarm annunciator.

Two text entry fields appear beneath the *On List*. The first of these is labelled *Tagname*. Click on this entry box to gain focus on it, then enter the tagname.atom which will be added to the *On List*. Click on the second entry box, labelled *Value*, and

enter the value to be downloaded to the previously named tagname.atom in the INFI 90 OPEN server, where they can be used to change the state of the atoms that trigger the contacts that drive the annunciator panel.

Once the tagname.atom and value have both been entered, click on *Apply*. The tagname.atom and value will be entered in the *On List*.

To delete an entry from the *On List*, first click on that entry in the *On List*. It will be highlighted. Click on *Delete*, and the item will be removed from the list immediately.

OFF LIST

Like the *On List*, the *Off List* is a scrolled list of tagname.atom and values. When all process alarms capable of triggering the alarm group have been acknowledged, the values in the *off list* are downloaded to their associated tagnames in the INFI 90 OPEN server. This allows the contacts which drive the alarm annunciator to be returned to their original state.

Configuration and deletion of items in the *Off List* is accomplished in the same manner as for the *On List*, described above.

CONFIGURE FILTERING OF MESSAGE CLASSES AND TYPES

The *Filter* button, located at the bottom of the *Alarm Group Configuration* window, is used to define the ability of various message types and classes to affect the alarm group which is being configured. Essentially, message types and classes which should not trigger the alarm group can be filtered out.

To filter message types for the alarm group which is being configured, click on *Filter*. The *Alarm Group Filtering* window, shown in Figure 2-9, will be displayed.

The left side of the this window contains a button for each message class which can be filtered out for the alarm group under configuration. These include:

- Process alarms.
- Priority.
- Area.

Process alarms is the default message class, which is shown in Figure 2-9. Click on the button corresponding to the message class to be filtered. The name of the selected message class appears in the upper right corner of the window. A list of message types for the selected message class appears in the message type display section on the right side of the window. Each of the message types is preceded by a toggle button. Click on these toggle buttons to select which message types for each

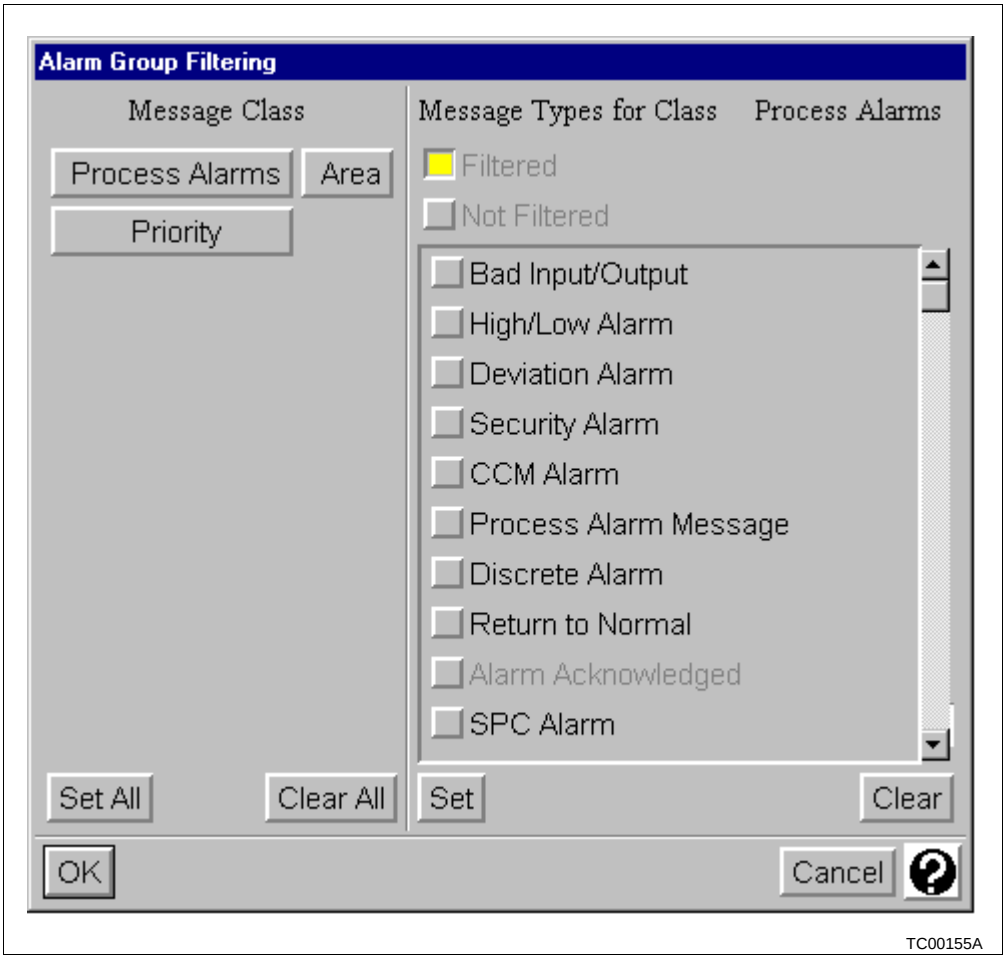


Figure 2-9. Alarm Group Filtering Window

message class are to be routed to a device. By default, all message types are not filtered (i.e., all will trigger the alarm group). This is indicated by the toggle button being in the out position, and gray in color. (When a toggle button is pressed in to filter a message type on this display, its color also changes to yellow to make the selected state more obvious.)

All message types for the currently selected message class can be set (toggle buttons pressed in), or cleared (toggle buttons released, placed in out position) simultaneously, by clicking on the *Set* or *Clear* buttons.

All message types for all message classes can be set (toggle buttons pressed in), or cleared (toggle buttons released, placed in out position) simultaneously, by clicking on the *Set All* or *Clear All* buttons.

Message Class: Process Alarm Messages

Process alarm messages are displayed in the top two lines of the mini-alarm window, known as the alarm display area, and also in the alarm review display. Process alarms are displayed in the following format:

<<Priority>><<Area>><<Time>><<Date>><<Tagname>><<AlarmCondition>> <<Legend>>

Process alarm messages are classified into the following message types:

- Bad Input/Output.
- High/Low alarm.
- Deviation alarm.
- Security alarm.
- Process alarm message.
- Discrete Alarm.
- Return to Normal.

Note that the alarm acknowledged message class is not available for selection from this window, and appears stippled out.

Message Filter Parameter: Priority

To filter messages by priority, click on the *Priority Message Class* button on the *Alarm Group Filtering* window (Figure 2-9). The message type area of the window will list all priorities. Click on each priority for which routing is to be changed, or use the *Clear* or *Set* buttons to clear or set the routing status of all priorities.

Message Filter Parameter: Area

To filter messages by area, click on the *Area Message Class* button on the *Alarm Group Filtering* window (Figure 2-9). The message type area of the window will list all areas. Click on each area for which routing is to be changed, or use the *Clear* or *Set* buttons to clear or set the routing status of all areas.

Exiting Alarm Group Filtering

Click on *OK* to exit the *Alarm Group Filtering* window and save the configuration, or click on *Cancel* to exit the window without saving changes.

EDIT AN ALARM GROUP

Configuration of an existing alarm group can be modified using the *Edit* button, located at the bottom of the *Alarm Group Selection* window. To do so, select the desired alarm group in the list

box by clicking it; it will be highlighted. Click on *Edit* to display the *Alarm Group Configuration* window for the selected alarm group. Configure new entries as described above.

DELETE AN ALARM GROUP

To delete an alarm group, click on the *Group Name* in the *Alarm Group Selection List*; the item will be highlighted. Click on *Delete* at the bottom of the *Alarm Group Configuration* window. A confirmation box will appear, which asks:

Do You Want to Delete <<name>>

where <<*name*>> is the name of the highlighted alarm group. Click on *Yes* to complete the *Delete* operation, or click on *No* to exit without making deletions.

CONFIGURING NETWORK PARAMETERS

To configure parameters which are global to the network, select the *Network Parameters* option from the *Configure* menu. This action produces the cascading menu, which provides configuration access to the following Network Parameters:

- Alarm setup.
- Alarm vectors.
- Area names.
- Event bit names.
- Printer assignment.
- Message routing.
- Network device assignments.
- Operator window parameters.

Alarm Setup

The parameters configured via alarm setup affect the system title and the display of alarm information, both of which appear in the mini-alarm window.

Select the *Alarm Setup* option on the PWC90 console *Parameters* menu to configure the following parameters:

- System title (appears in title bar of mini-alarm window.)
- Alarm presentation on top line of mini-alarm window (with all alarms acknowledged.)
- Alarm presentation on top line of mini-alarm window (With Unacknowledged alarms.)
- Audible alarm configuration.
- Priority color assignments.

- Mini-alarm window field widths.

Selecting the *Alarm Setup* option causes the following display to appear (Figure 2-10).

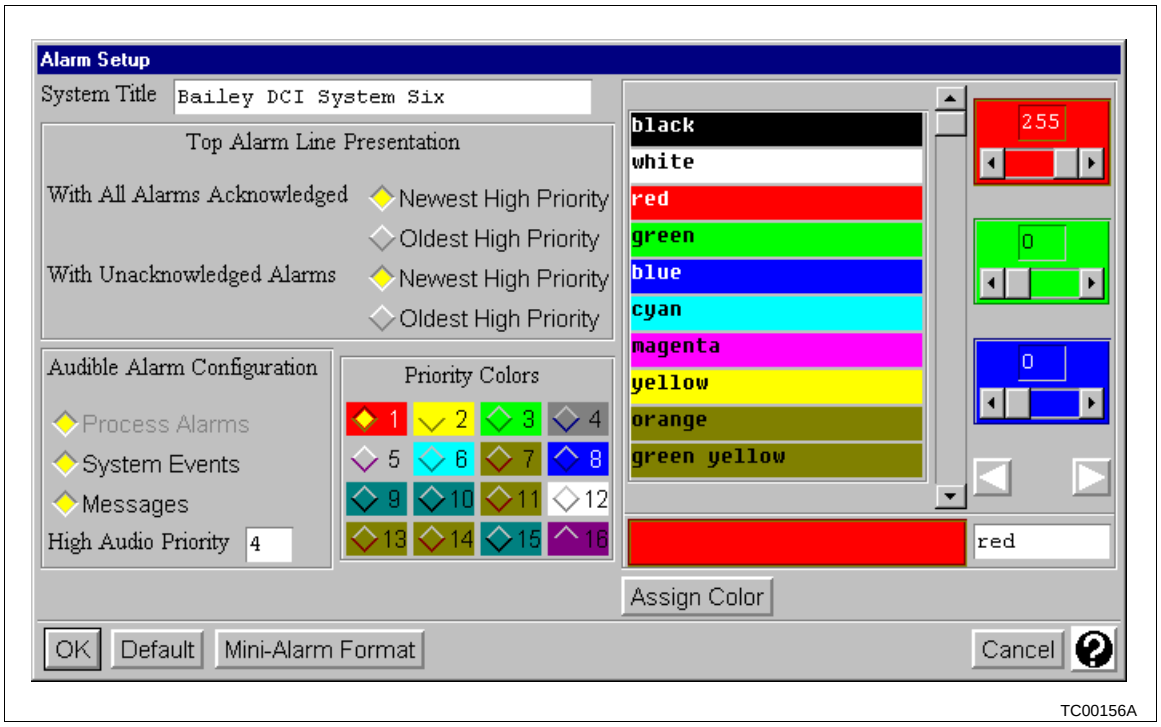


Figure 2-10. Alarm Setup Window

The main work area of the *Alarm Setup* window is divided into several smaller areas, including the *System Title* entry box, *Top Alarm Line Presentation* configuration, *Audible Alarm* configuration, *Priority Colors* configuration, and a color list which contains a selection entry box and an *Assign Color* button.

NOTE: Changes to these parameters take effect when the PWC90 console is restarted.

SYSTEM TITLE

The first item which appears in the *Alarm Setup* window is a text entry box labelled *System Title*. Click on this box to gain focus, then enter a text string of up to 32 characters which is to be used as the system title. The system title appears in the title bar of the mini-alarm window. The default system title is ***ELSAG BAILEY INFI 90 OPEN***.

TOP ALARM LINE PRESENTATION

The top two lines of the mini-alarm window are used to display messages regarding process conditions which are in alarm. The top alarm line can be configured to display either the new-

est or oldest alarm which has the highest priority. (Priority one is the highest priority; Priority 16 is the lowest.) Furthermore, this may be configured to behave differently, depending on whether unacknowledged alarms exist, or if all alarms have been acknowledged. By default, the system will display the newest high priority alarm, whether or not unacknowledged alarms exist.

To configure the alarm presentation in the top alarm line with all alarms acknowledged, position the pointer on the appropriate box, then press and hold the left mouse button. The two possible choices will be displayed:

- *Newest high priority.*
- *Oldest high priority.*

The current choice will appear as a pushbutton. Change the setting by dragging the mouse up or down, then release the left mouse button.

For the change to take effect, the mini-alarm window must be re-opened; to do this, restart the PWC90 console.

The top alarm line presentation when unacknowledged alarms exist can be configured in the same manner.

PRIORITY COLOR

There are 16 alarm priorities available on the PWC90 console. Each of these priorities can be configured to display in a different color, to aid in rapid identification of the importance of the alarm. A palette of 16 priority color assignments is preconfigured on the PWC90 console. The default palette uses the following colors:

This color palette can be modified, using any of the 256 colors available from the *Alarm Setup* display. Use the following steps to configure a color for a priority:

1. Click on the *Priority Color* button to be changed. The number of the priority appears above the selection entry box, which is located below the list box. The name of the current color choice appears inside the selection entry box. The current color is displayed in the color sample box located above the list box.

2. In the list box, click on the name of the color to be assigned to the selected priority. This color will replace the current color in the color sample box. Click on colors until a satisfactory color is found.

Alternatively, click on the selection entry box, use the **Backspace** to remove the previous color name, and type in the name of the desired color using the keyboard. If this method of color selec-

Table 2-1. Priority Color Assignments

Level	Color	Rank
Priority 1	Red	(Highest Priority)
Priority 2	Yellow	
Priority 3	Green	
Priority 4	Slateblue	
Priority 5	Hotpink	
Priority 6	Cyan	
Priority 7	Darkorange	
Priority 8	Blue	
Priority 9	Aquamarine	
Priority 10	Seagreen	
Priority 11	Chocolate	
Priority 12	Seagreen1	
Priority 13	Sandybrown	
Priority 14	Tomato	
Priority 15	Turquoise4	
Priority 16	Deeppink	(Lowest Priority)

tion is used, the newly selected color will not appear in the color sample box.

3. Click on *Assign Color* to assign the newly chosen color to the selected priority.

AUDIBLE ALARM CONFIGURATION

Audible alarms are sounds produced by an annunciator. On the Personal Work Center 90, the annunciator is located within the PC. These sounds accompany incoming alarms to notify the operator of their presence. The user can configure audible alarms to accompany system events and messages; process alarms are always accompanied by audible alarms.

The types of messages (including process alarms, system events and operator messages) received by a user are determined by the configuration of message filtering for that user's login account (described in ***User Message Filtering*** in this section). This means that not all PWC90 consoles will necessarily receive the same set of alarms and messages. When there are no unacknowledged process alarms, system events or messages in the mini-alarm window on a particular PWC90 console, the audible alarm will be automatically turned off for that PWC90 console until the next unacknowledged alarm or message appears in the mini-alarm window.

NOTE: That there can still be unacknowledged alarms on the system.

The *Audible Alarm Configuration* area is located at the middle left side of the *Alarm Setup* window, and is composed of the following parameters:

- Process alarms.
- System Events.
- Messages.
- High Audio Priority.

Process Alarms and Audible Alarm Configuration

The *Process Alarms* parameter appears solely for informational purposes. The face of the associated button displays the legend *On*, which is stippled out, indicating that it cannot be accessed. Process alarms are always *On*; that is, incoming process alarms will always be announced by an audible alarm.

Audible Alarms for System Events

A toggle button labelled *System Events* allows the user to determine whether system event messages will be accompanied by an audible alarm. The face of the toggle button reflects the currently selected state: *On* indicates that audible alarms will accompany system Event messages, while *Off* indicates that no audible alarm will sound when a system Event message appears in the third line of the mini-alarm window. Click on the toggle button to alternate between the two states.

Audible Alarms for Messages

A toggle button labelled *Messages* allows the user to determine whether operator messages will be accompanied by an audible alarm. The face of the toggle button reflects the currently selected state: *ON* indicates that audible alarms will accompany operator messages, while *off* indicates that no audible alarm will sound when an operator message appears in the fourth line of the mini-alarm window. Click on the toggle button to alternate between the two states.

Audible Alarm Priority Assignment

Two different audible alarm sounds are available on the PWC90 console: a steady beep and a rapid beep. The rapid beep is used with high priority alarms. The *High Audio Priority* parameter is used to configure which priorities will be accompanied by which sound.

Click on the text entry box for this parameter to gain focus, then enter the number of the priority at which the sound characteristics of the audible alarm change. The alarm priority which corresponds to this number, and all priorities higher

(i.e., with lower priority numbers) will be accompanied by the rapid beep.

For example, the default *High Audio Priority* setting is four; this means that alarms and messages with priority numbers one through four will be accompanied by a rapid beep, while alarms and messages with priority numbers five through 16 will receive the steady beep.

DEFAULT ALARM SETUP

Return to the default color set by clicking *Default*, near the bottom left corner of the *Alarm Setup* window. A modal pop up window appears, containing the question:

Are You Sure?

The user must respond to a modal pop up window before any other action can be taken. Click on *Yes* to return to the default color set, or click on *No* to remain with the currently assigned colors.

MINI-ALARM WINDOW FIELD WIDTH

The top two lines in the mini-alarm window display information about process alarms. This information is displayed in a number of fields. The odering of the fields is the same as the alarm review display (discussed in the ***Personal Work Center 90 Operation*** instruction). The width of each of these fields is configurable. To configure field widths, click on *Mini-alarm Format* located at the bottom of the *Alarm Setup* window. The *Mini-alarm Format* window (Figure 2-11) will be displayed.

The default size of each field width is as shown in Table 2-2:

The default sizes used reflect the maximum number of spaces required for each piece of information. The default sizes total 131 characters. Not more than 85 characters can be displayed on the process alarm line of the mini-alarm window. If the default settings are used, a portion of the legend cannot be viewed in the mini-alarm window.

NOTE: The spaces used to separate the fields are automatically supplied by the PWC software.

Click on *Default* to return all entries to their default settings.

NOTE: The PWC90 console must be restarted to effect changes made via the mini-alarm format window.

The configuration of the mini-alarm field widths will be used network wide, so that all mini-alarm lines on all PWC90 consoles on the network will have the same appearance. The con-

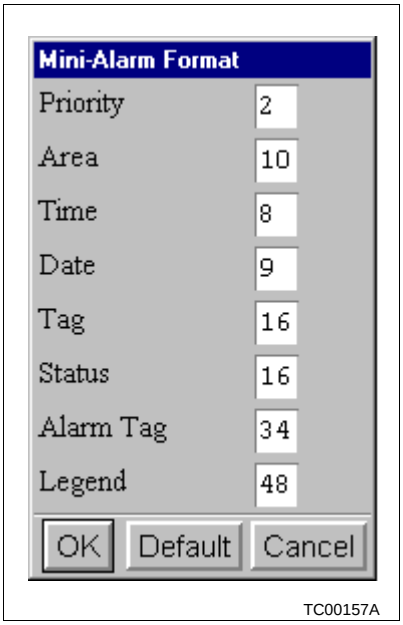


Figure 2-11. Mini-Alarm Format Window

Table 2-2. Default Field Widths

Field Name	Default Field Width
Priority	2
Area	10
Time	8
Date	9
Tag	16
Status	16
Alarm Tag	22
Legend	48

figuration of the mini-alarm field widths has no effect on the *Alarm Review* display.

EXITING ALARM SETUP

To exit the *Alarm Setup* display, choose one of two options. To exit and save all changes to the configuration, click on *OK*. To exit without saving any changes, click on *Cancel*.

Changes to these parameters take effect when the PWC90 console is restarted.

SUMMARY - ALARM SETUP

Alarm Setup

Alarm setup is an option on the PWC90 console *Parameters* menu which provides access to the configuration of the following items:

- Alarm presentation on the top line of the mini-alarm window when all alarms are acknowledged.
- Alarm presentation on the top line of the mini-alarm window when unacknowledged alarms exist.
- System title.
- Priority color assignments.
- Audible alarm configuration.
- Mini-alarm window field widths.

Audible Alarms

Audible alarms are sounds produced by an annunciator. On the PWC90 console, the annunciator is located within the PC. These sounds accompany incoming alarms to notify the operator of their presence. The user can configure audible alarms to accompany system events and messages; process alarms are always accompanied by audible alarms.

Default Button

The *Default* button appears near the lower left corner of the *Alarm Setup* window. Click on this button to return alarm priority color choices to the default color palette. A modal pop up window provides the opportunity to abort the operation.

Priority Color

Each of the 16 alarm priorities on the PWC90 console can be assigned to display in a different color, to aid in rapid assessment of the alarm's importance. A default palette of 16 colors is preconfigured on the PWC90 console. To quickly change a color assignment, select the priority color button to be changed by clicking it, then locate the new color in the color list and click on it. Finally, click on *Assign Color* to make the assignment.

System Title

The system title is a string of up to 32 alphanumeric characters which appears in the title bar of the mini-alarm window. This string can be used to identify the installation name. (The default system title is ELSAG BAILEY INFI 90 OPEN system). The system title is configured via a text entry box at the top of the *Alarm Setup* window. Click on this box to gain focus, then enter the desired text string.

Top Alarm Line Presentation

The top two lines in the mini-alarm window are used to display messages regarding process conditions which are in alarm. The top alarm line can be configured to display either the newest or oldest alarm which has the highest priority. This can be

configured to behave differently, depending on whether unacknowledged alarms exist, or if all alarms have been acknowledged. By default, the system will display the newest high priority alarm, whether or not unacknowledged alarms exist.

Two pushbuttons exist to control the alarm presentation under the two possible conditions:

- All alarms acknowledged.
- Existence of unacknowledged alarms.

In either case, click on the pushbutton, then drag the pointer to the desired selection.

Alarm Vectors

Select *Alarm Vectors* from the *Network Parameters* menu to view the *Alarm Vector Configuration* window (Figure 2-12). This window is used to assign specific displays to tagnames. If this assignment has been configured, and the associated point goes into alarm, the user need only click on *V* in the alarm line of the point (in either the mini-alarm window or alarm review window) to call up the assigned display.

> **NOTE:** If alarm vectoring is not configured for a tagname, clicking the *Vector* button will cause the point display for that tag to be displayed by default (no configuration is necessary.)

The *Alarm Vector Configuration* window is divided vertically into two sections. The left portion contains a list box labelled *Tagname*. This contains a list of all tagnames which have been configured with alarm vectors. When the list contains more items than can be displayed at one time, additional entries can be viewed by clicking on the up and down arrow buttons.

The right portion of the window is labelled *Select Vector Display Type* and contains a fixed list of display types available for vectoring. These include:

- Graphic.
- Summary.
- Group.
- Point.
- Trend.
- System status.
- Alarm review.
- Event review.
- Message review.
- Printer review.
- Event history.
- Unit overview.

Below the *Display Type* list box are two blank fields labelled *Display Type* and *Index*. Below these fields are two text entry

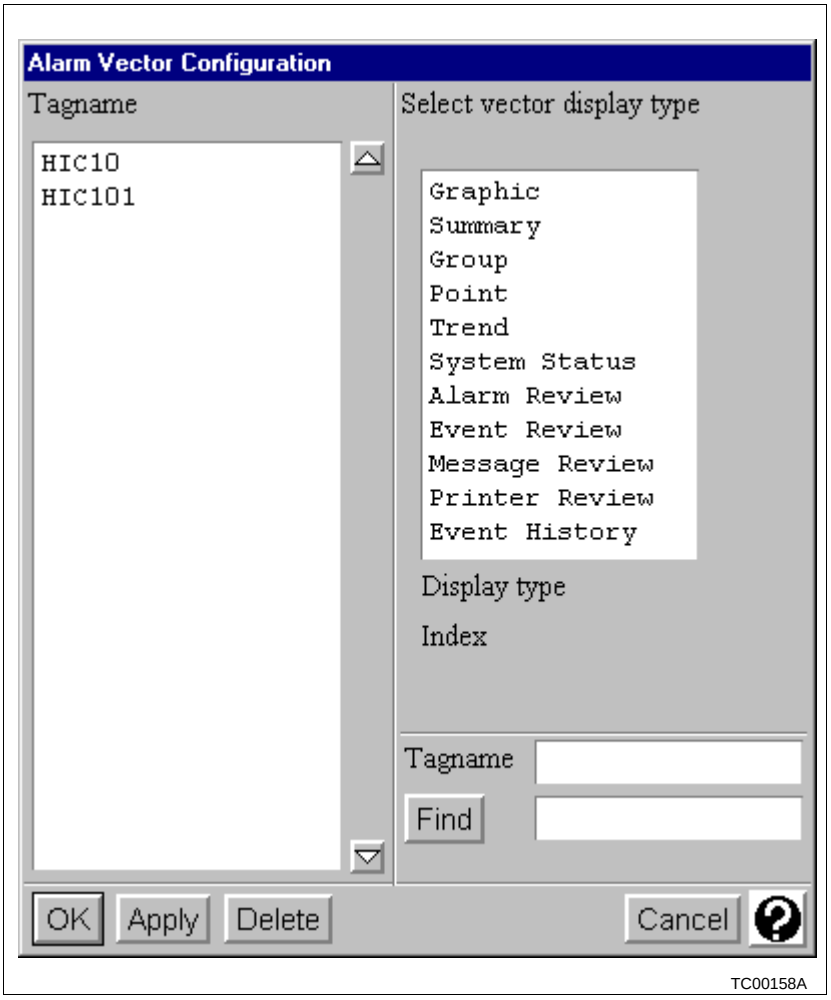


Figure 2-12. Alarm Vector Configuration Window

boxes. The first entry box is labelled *Tagname*, the second appears to the right of and is used in conjunction with the *Find* button.

To assign an alarm vector to a tagname, first configure the tagname, then specify the target display. This procedure is described below.

CONFIGURING THE TRIGGERING TAGNAME

When the *Alarm Vector Configuration* window first appears, the *Tagname* entry box is automatically highlighted and ready to accept input. Enter the tagname for which vectoring is being assigned.

When configuration of both the tagname and display type has been completed for the alarm vector, the tagname will be inserted into the *Tagnames* list box, in alphabetical order; the

list will shift within the list box so that the newest entry is displayed at the top of the list box.

> **NOTE:** No warning will be given if the tagname does not exist; the string will be accepted and entered into the *Tagnames* list box.

SPECIFYING THE TARGET DISPLAY

Once a tagname has been entered in the *Tagname* entry box or selected from the *Tagnames* list box, select the display which will be used when that tagname is in alarm and the vector button is clicked. (Vector buttons are located on both the alarm lines in the mini-alarm window and on each line of the alarm review display).

To do so, position the pointer on the desired type in the list box labelled *Select Vector Display Type* and click on it. The selection will be highlighted and will automatically be displayed in the field labelled *Display Type*, located immediately below the *Display Type* list box.

If graphic, summary, group, point or trend has been selected as the *Display Type*, more than one display is available for that type; a pop up window will appear to aid the user in specifying the index of the desired display. in the case of graphic, summary, group and trend display types, the window which appears will be similar to the following (Figure 2-13).

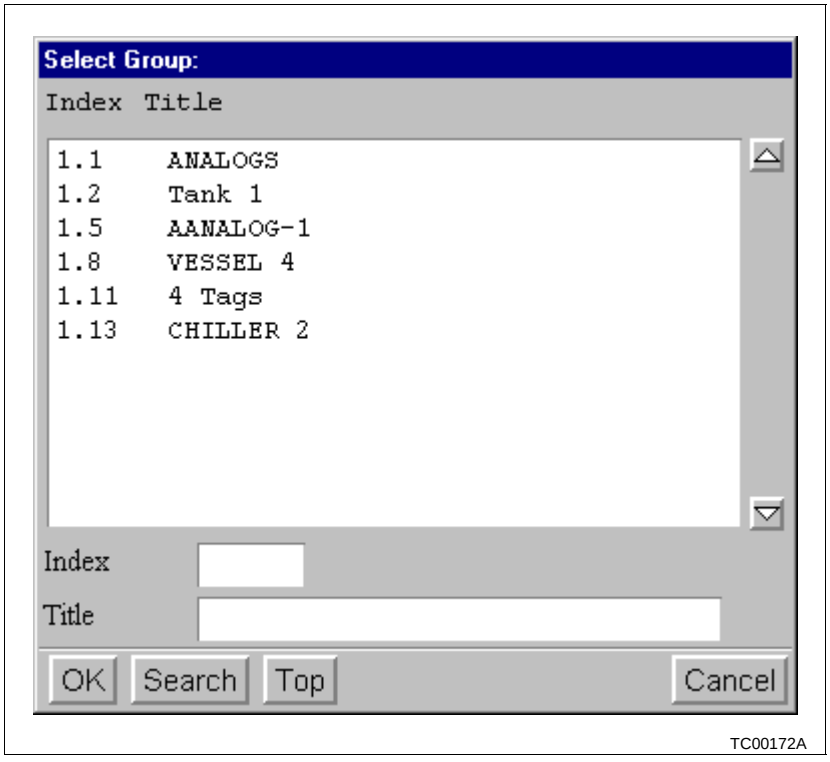


Figure 2-13. Select Graphic Window

Use the scroll bar and the up and down arrow buttons to view the list of available displays for the selected display type. If all or part of the title is known, enter it in the *Title* entry box, then click on *Search*. If found, the item will be displayed at the top of the list box, highlighted. Similarly, if the index number is known, enter it in the *Index* box, then click on *Search* to locate it in the list, or click on *OK* to make the selection.

To move quickly to the first item in the list box, click on *Top*.

When the desired display has been selected, click on *OK* to complete the operation, or click on *Cancel* to exit without making a selection.

If *Point* has been selected as the *Display Type*, a much simpler pop up window will appear. The *Point Selection* window consists of a text entry box labelled *Enter Tagname*, an *OK* button and a *Cancel* button. Enter the tagname of the point in the entry box, then click on *OK* to complete the operation, or click on *Cancel* to exit without specifying a tagname. If a tagname has been specified, use of the vector button will cause the point display associated with the specified tagname to appear. This allows the user to vector a point display other than that of the point which is in alarm.

If *Printer Review* is selected as the display type, a *Printer Selection* window (Figure 2-14) will appear. This window contains a list of devices configured on the network. Click on the name of the device containing the messages which are to be displayed. The device name will be highlighted. Click on *OK* (or double click on the device name) to complete the operation, or click on *Cancel* to exit without selecting a Device. If a device has been specified, use of the vector button will cause the printer review display for the selected device to appear.

If only one display is available for the chosen display type (as is the case for system status, alarm review, event review, message review and event historian displays), no index number is required. Click on the desired type in the *Display Type* list. The name of the type will be displayed in the *Display Type* field below the list, and the *Index* field will display a pair of dashes (--).

Target display selection is not saved until either the *OK* or *Apply* button in the *Alarm Vector Configuration* window has been clicked.

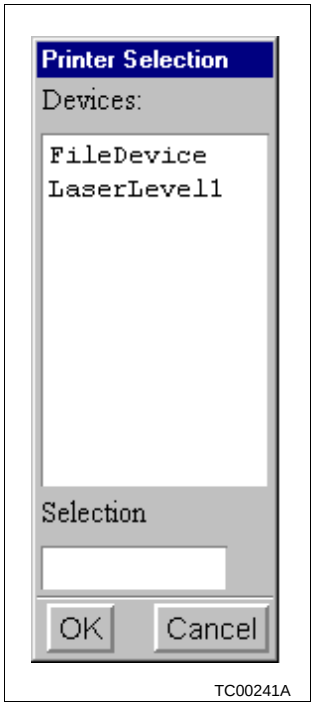


Figure 2-14. Printer Selection Window

CHANGING EXISTING ALARM VECTORING ASSIGNMENTS

Once points have been configured for alarm vectoring, use the *Find* button to quickly locate and highlight any entry in the *Tagnames* list box. To do so, enter the desired tagname (or any known portion thereof) in the text entry box to the right of the *Find* button, then click on *Find*. The first entry containing the desired string will appear at the top of the *Tagnames* list box, highlighted. Click on *Find* again to find the next entry which matches the entered text string.

CONFIGURING MULTIPLE ALARM VECTORING ASSIGNMENTS

Multiple alarm vectoring assignments can be configured without exiting the *Alarm Vector Configuration* window by using the *Apply* button. To do so, first enter the tagname (or select it from the list in the *Tagnames* list box) and specify the target display. Click on *Apply*. The tagname will be entered in the *Tagnames* list, the alarm vectoring assignment will be completed. Repeat the information entry and use of the *Apply* button as often as necessary.

DELETING ALARM VECTORING ASSIGNMENTS

To delete an alarm vectoring assignment, click on the desired tagname in the *Tagnames* list box; it will be highlighted. Click on *Delete* at the bottom of the *Alarm Vector Configuration* window. A confirmation box will appear, which queries:

Delete Selected Items?

Click on *Yes* to delete the vectoring assignment, or click on *No* to exit the operation without making any changes.

If more than one item is to be deleted, all items to be deleted must occupy adjacent lines. Position the pointer on the first item to be deleted and press the left mouse button. Continue to hold the mouse button down, and drag the pointer down to the last item to be deleted. Release the mouse button. The confirmation box will appear. Proceed as described above.

USING THE VECTOR BUTTON

Vector buttons appear on each of the alarm lines in the mini-alarm window (the top two lines), and on each line used in the alarm review display. A *V* is displayed on the face of each vector button.

If alarm vectoring has been configured for the point which is in alarm, clicking the vector button on an alarm line in the mini-alarm window will cause the assigned display to appear (by default) in the operator window with the lowest index number (i.e., Operator Window #1). Clicking the vector button on an alarm line in the alarm review display will cause the assigned display to appear (by default) in the operator window which contains the alarm review display. In either case, the display can optionally be made to appear in a different window by pressing and holding the left mouse button when selecting a vector button, then dragging the pointer to a location inside frame of the alternate window. When the mouse button is released, the display will appear within that window.

Area Name Assignment

Select *Area Names* from the *Network Parameters* menu to view the *Area Setup* display, shown in Figure 2-15.

The *Area Setup* display is used to configure a ten character name for each of the 64 process areas. To do so, click on the text entry box and enter up to ten characters.

These names are used network wide for user reference; it is recommended that they be unique.

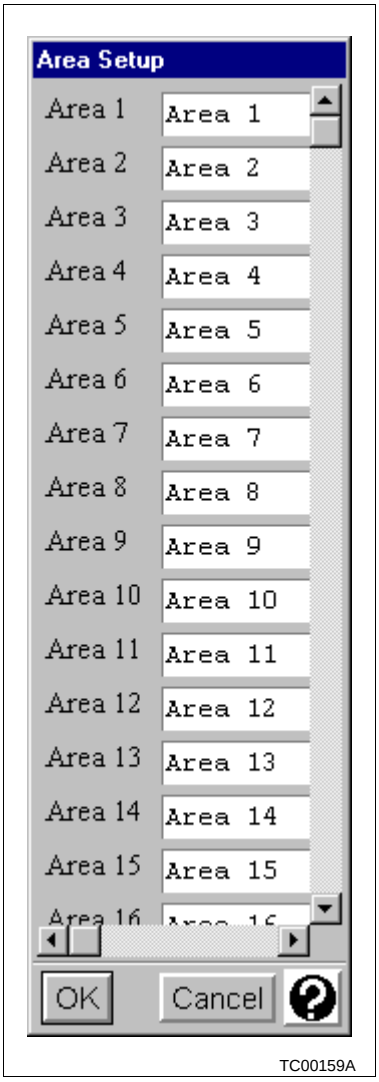


Figure 2-15. Area Setup Display

Event Bit Name Configuration

Event bits are a set of 4096 network wide event flags which can be set to indicate that an event has occurred. Using event bit name configuration, a name can be assigned to each event bit. Such a name can be mnemonic and descriptive of the event bit's function. Other features on the PWC90 console, such as the historical database, can then be configured to reference the event bit by name.

Select *Event Bit Names* from the *Network Parameters* menu to display the *Event Bit Configuration* window (Figure 2-16).

The top area of this window contains a list of the first 15 event bits and their names (if any), which can have a maximum of 15 characters. Below this list are the *Goto* and *Find* buttons, used to access other event bits. A button labelled *Delete All* is also

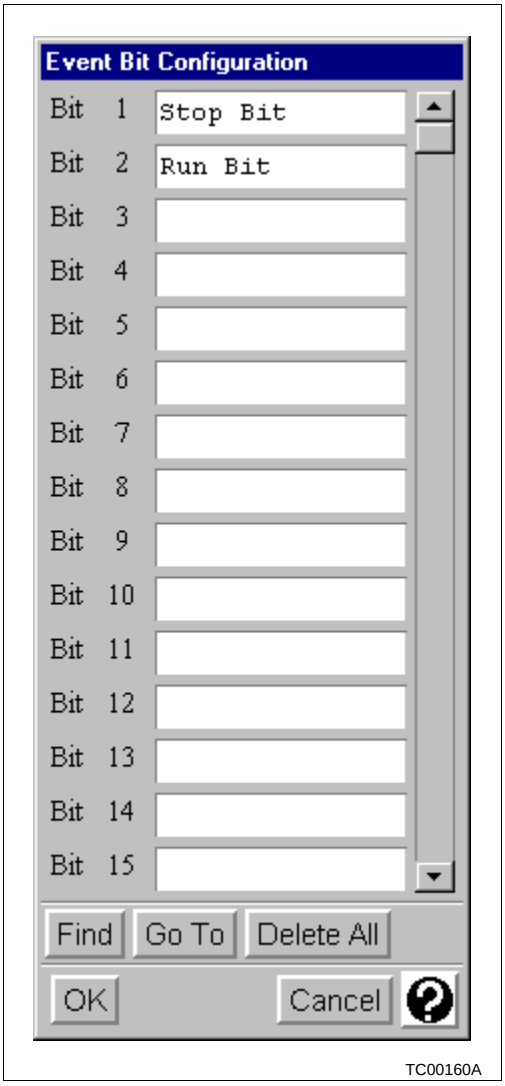


Figure 2-16. Event Bit Configuration Window

available for clearing the event bit name configuration. The functions of these buttons are described below.

To enter or change an event bit name, first click on the text entry box (located to the right of the event bit number) to gain focus, then enter an event bit name of up to 15 characters.

GOTO FUNCTION BUTTON

The *Goto* button is used to access an event bit by its index number. To do so, click on the *Goto* button; a pop up window will be displayed. Click on the text entry box labelled *Enter Event Bit Index* to gain focus on it, then enter the index number of the desired event bit and click on *OK*. A list of 15 event bits, starting at the requested event bit, will be displayed in the event bit configuration window.

FIND FUNCTION BUTTON

The *Find* button is used to access an event bit by name. To do so, click on *Find*; a pop up window will be displayed. Click on the text entry box labelled *Enter Event Bit Name* to gain focus on it, then enter the name of the desired Event Bit and click on *OK*. A list of 15 event bits, starting at the requested event bit, will be displayed in the event bit configuration window.

DELETE ALL FUNCTION BUTTON

The *Delete All* button is used to delete all configured event bit names. to do so, click on this button; a confirmation box will appear. Click on *Yes* if all event bit names are to be deleted; click on *No* to exit without deleting the names.

EXITING EVENT BIT NAME CONFIGURATION

Click on *OK* to exit event bit configuration and save any changes, or click on *Cancel* to exit without saving any changes.

Printer Assignment

Printer Assignment is used to define the devices which will receive messages. These devices can be printers or files.

Selecting *Printer Assignment* from the *Network Parameters* menu causes the *Printer Assignment* display (Figure 2-17) to appear. This display lists the currently configured devices, and allows the attributes of these devices to be added, deleted or changed.

ADDING A DEVICE

To add a device to the list, click on *Add*.

This action will cause a *Device Characteristics* display similar to the example in Figure 2-18 to appear. The information contained in the data fields will vary, depending on the actual configuration of the system. If no data has been configured for the device, default information will be displayed on the pushbuttons. Device characteristics which can be configured include:

- Device name.
- Backup device.
- Node name.
- Device type.
- Port number.
- File size (in lines).
- Use ETX/ACK protocol.
- Emulation.

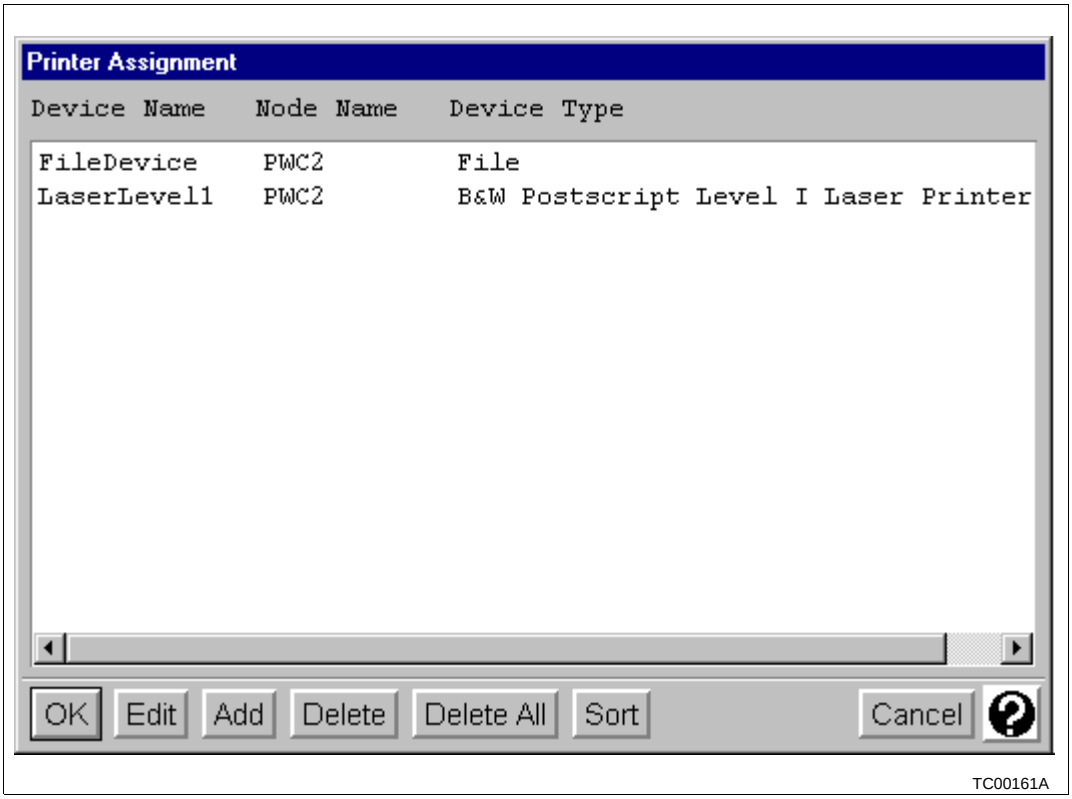


Figure 2-17. Printer Assignment Display

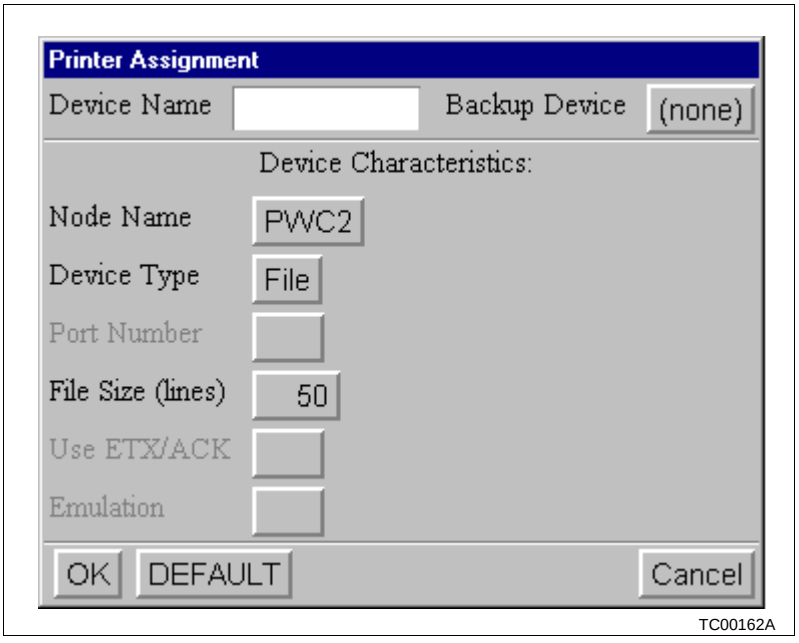


Figure 2-18. Device Characteristics

Device Name

The device name is an 11 character alphanumeric string used to identify the printer or file. To enter or change the device name, click on the *Device Name* entry box. A bar cursor appears, signaling that the entry box will receive input. Using the keyboard, enter the name to be used for the device. Spaces cannot be included in the device name. Each device name on the network must be unique in order to avoid confusion; while it is possible to assign the same device name to more than one device, this is not good practice.

> **NOTE:** The name of the printer configured for PWC90 must be exactly the same as the name of the corresponding printer configured for Windows NT, otherwise printing will not work, and no diagnostic messages will be issued.

Backup Device

The backup device parameter is used to select an alternate device, to which messages will be sent in the event that the primary device fails. To assign a backup device, click on *Backup Device*; a list of available (previously configured) devices will appear. Highlight the device which is to be used as the backup device by clicking on it with the mouse. Click on *OK* to complete the selection process, or click on *Cancel* to exit without making changes.

Node Name

The node name is the name of the node to which the device is connected. The local node name appears by default. To change node names, click on *Node Name* pushbutton. A window will appear which contains a list of available nodes. Highlight the desired node by clicking on it. Click on *OK* to complete the selection process, or click on *Cancel* to exit without making changes. Each node name on the network must be unique in order to avoid confusion; while it is possible to assign the same node name to more than one device, this is not good practice.

Port Number

The port number to which printer output is sent is dependent upon which type of node and which device type have been selected. When the selected node name is that of a printer server, this parameter can be configured. In the case of other node types, printer output is directed according the rules defined in this subsection.

When *File* has been selected as the device type, no port is required and the *Port Number* button is stippled out.

When *Line Printer* is selected, and the selected node name is a PWC90 computer, printer output will automatically be sent to the second serial port on the PWC90 computer. The *Port Number* button will be stippled out.

When *Line Printer* is selected, and the selected node name is a printer server, the port number button is available. To change this parameter, click on *Port Number*. A window titled *Available Ports* will be displayed. This window contains a list of available ports for the selected node name. Highlight the port to be used by clicking on it with the mouse. Click on *OK* to complete the selection process, or click on *Cancel* to exit without making changes.

When either of the PostScript printer options (Black & White PostScript Level I Laser Printer or Color or Black & White PostScript Level II Laser Printer) is selected, the printer output will automatically be directed to the parallel port on that PWC90 computer. If the selected node name is that of a printer server, the port number can be selected. To change this parameter, click on *Port Number*. A window titled *Available Ports* will be displayed. This window contains a list of available ports for the selected node name. Highlight the port to be used by clicking on it with the mouse. Click on *OK* to complete the selection process, or click on *Cancel* to exit without making changes.

File Size

All messages sent to a device are automatically stored in a file, which can be viewed using the *Printer Review* selection available via the *Select* button on the operator window menu bar. If the user elects to define a file (instead of a printer) as the device to which messages are sent, this is the file which is used. In either case, the default file size is 50 lines.

The file size parameter is used to configure the number of lines to be stored in the file. To change this parameter, click on *File Size*. A window will appear which contains a list of supported sizes; choices include 50, 100, 500, 1000, 5000 or 10,000 lines. Highlight the desired selection. Click on *OK* to complete the selection process, or click on *Cancel* to exit without making changes.

Use ETX/ACK Protocol

The *Use ETX/ACK* parameter is available for use when the device type parameter is set to line printer. Line printers supported on INFI 90 OPEN system use the ETX/ACK protocol, which is selected when the word *Yes* appears on the face of the button; this is the default state. If the line printer assigned to

this device uses a different protocol (i.e., XON/XOFF), click on this button to toggle to the No state.

NOTE: If a device type other than line printer is selected, then this parameter is stippled out, and the face of the button is blank.

Emulation

A button labelled *Emulation* is used to configure this parameter. It is available for use when the device type parameter is set to line printer. Click on *Emulation* to toggle between *Epson* (the default) and *IBM Proprinter* emulation; choose the emulation supported on your printer.

NOTE: If a device type other than line printer is selected, then this parameter is stippled out, and the face of the button is blank.

Default Device Characteristics

Default device characteristics are listed in Table 2-3.

Table 2-3. Default Device Characteristics

Device Characteristic	Default
Default	Entry box is empty.
Backup Device	(none)
Node Name	Defaults to the name of the local node.
Type	File
Port Number	Button face is blank by default when the type characteristic is set to File; otherwise it defaults to 1.
File Size	50
Use ETX/ACK	YES
Emulation	EPSON

To return all parameters to default settings, click on *Default*.

CHANGING DEVICE CHARACTERISTICS

If a device has already been configured, the characteristics of the existing device can also be changed via the *Printer Assignment* display. To do so, highlight the desired device on the list of devices by clicking on it with the mouse; the item will be highlighted. Next, click on *Edit*. This action will cause the *Device Characteristics* display for that device to appear. Device characteristics can then be changed as described in ***ADDING A DEVICE*** in Section 2.

DEVICE SORT

The list of configured devices can be sorted by device name or by node name. Sorting by device name places the devices in alphanumeric order according to name, regardless of the node to which they are attached. Sorting by node name sorts the devices primarily by node, and then by device name.

To sort the list of devices, click on *Sort*. The *Sort Criteria* window will appear, as shown in Figure 2-19.

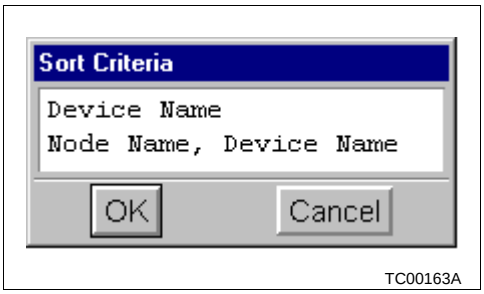


Figure 2-19. Sort Criteria Window

Two options appear in the *Sort Options* window:

- *Device name.*
- *Node name, device name.*

Click the method to be used for the sort, then click on *OK* to complete the selection process. The results of the sort operation will be seen immediately in the device list found in the *Printer Assignment* display.

Click on *Cancel* on the *Sort Options* window to exit the window without performing a sort operation.

DELETING A DEVICE

To delete a device from the *Printer Assignment Device* list, click on the item in the list which is to be deleted, then click on *Delete*. The item will be removed from the list immediately.

DELETING ALL DEVICES

To delete all devices in the *Printer Assignment Device* list, click on *Delete All*. A confirmation box appears requesting that the user confirm the decision. Click on *Yes* to delete all devices, or *No* to exit the display without deleting any devices.

Message Routing

Information on various topics is provided by the system in the form of messages. Messages are divided into categories called message classes; this enables the system to distinguish between different kinds of messages and allows efficient message sorting.

Messages within each class are further distinguished by separation into message types. Definitions of types within a message class enables the PWC90 console to quickly process and/or sort messages within a class.

Message classes on the PWC90 console include:

- Process alarms.
- System events.
- Operator messages.
- Operating status messages.
- Operating errors.
- Operator actions.

Process alarms, system events and operator messages provide information which must be immediately available to the operator; messages from these three classes are displayed in the mini-alarm window on the PWC90 console screen. The remaining message classes reflect status changes in the system rather than alarm conditions; messages belonging to these classes are not displayed in the mini-alarm window. Logging of messages is configurable.

In addition to message classes, message routing also allows the filtering of messages based on the following parameters:

- Priority.
- Area.

Message routing enables messages of any message class to be separately routed to any device which has been configured through the *Printer Assignment* menu selection (refer to ***Event Bit Name Configuration*** in this section). Message routing also allows the selection of which message types in each message class are to be routed to any device, filtering out messages which are not important for the specific application.

Select *Message Routing* from the *Network Parameters* menu to display the pop up window shown in Figure 2-20.

The *Message Routing Assignment* window contains a matrix of devices and message classes. Only devices which have been configured through the *Printer Assignment* window (refer to ***Printer Assignment*** in this section) will appear as part of the matrix.

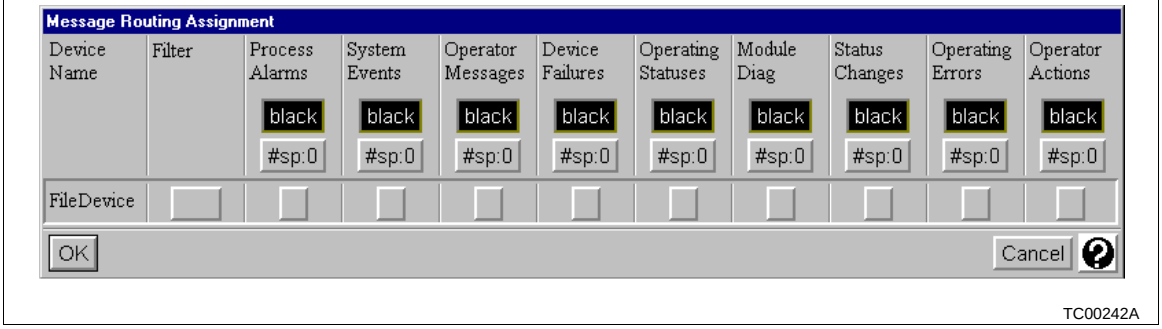


Figure 2-20. Message Routing Assignment Window

Each message class has a button for each available device. To route messages to a device, click on the button that corresponds to both the message class and device desired. An *X* will appear on the face of the button. If no messages are being routed for a device/message class combination, the button is blank.

Each device has a *Filter* button, which provides access to message filtering functions. Each message class in the matrix has a *Color* pushbutton used to define the color of messages in that message class, and a *Space* pushbutton, used to define the number of spaces that messages of that message class are to be indented.

MESSAGE TYPE FILTERING

If all message types for all message classes are being routed to a device, the face of the *Filter* button will be blank. If any message types for any of the message classes have been filtered (not routed to a device), the *Filter* button will display the letter *F*.

To filter message types for a particular device, click on *Filter* for the device which is to receive filtered messages. The *Device Routing Assignment* window, shown in Figure 2-21, will be displayed. The left side of the *Device Routing Assignment* window contains a button for each message class. *Process Alarms* is the default message class, which is shown in Figure 2-21. Click the button corresponding to the message class to be filtered. The name of the selected message class appears in the upper right corner of the window. A list of message types for the selected message class appears in the message type display section on the right side of the window. Each of the message types is preceded by a toggle button. Click on these toggle buttons to select which message types for each message class are to be routed to a device. By default, all message types are routed (indicated by the toggle button in the in or pressed position.)

All message types for the currently selected message class can be set (toggle buttons pressed in), or cleared (toggle buttons released, placed in out position) simultaneously, by clicking *Set* or *Clear*.

All message types for all message classes can be set or cleared (toggle buttons released, placed in out position) simultaneously, by clicking *Set All* or *Clear All*.

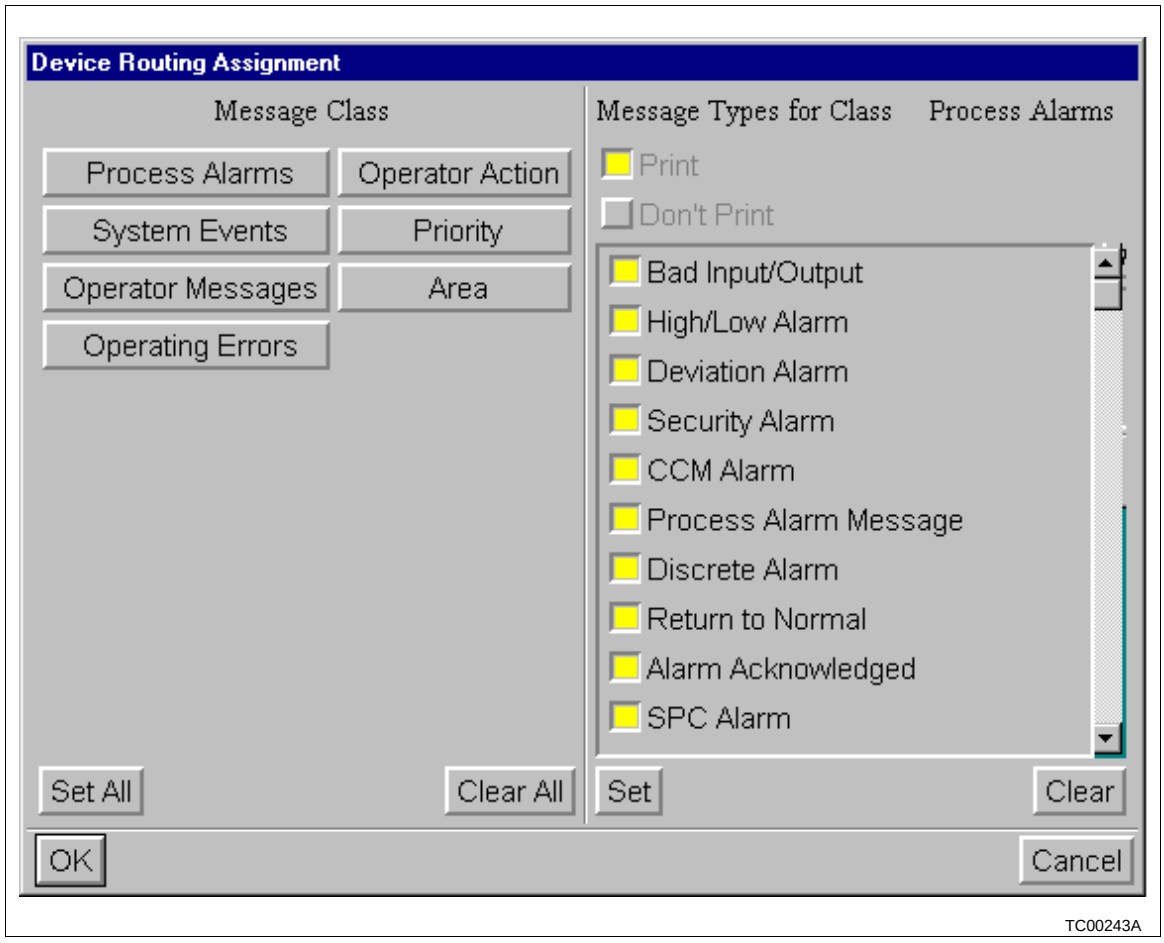


Figure 2-21. Device Routing Assignment Window

MESSAGE COLOR DEFINITION

By default, messages which are sent to a printer are printed in black, and messages which are sent to a file appear as black characters on a white background.

To define a different color to be used when a message from a particular class is sent to a file (or to an optional color printer), click the color selection button located immediately below the message class name. The pop up window shown in Figure 2-22 will appear.

Figure 2-22. Message Color Selection Window

Click the desired color to highlight it, then click on *OK* to complete the selection process. Click on *Cancel* to exit the operation without changing the color definition.

> **NOTE:** The message color defined here is used when the message is logged to a device (file or printer); it is not the same as alarm priority color (refer to ***PRIORITY COLOR*** in this section), and will not affect messages in the mini-alarm window or alarm review display.

MESSAGE INDENTATION DEFINITION

The *Space* button, located immediately below the *Color* button, is used to define the number of spaces of indentation to be used for each message class. Where a number of different message classes are routed to the same printer, setting a different indentation for each message class can assist the user looking for messages of a particular class. The face of the *Space* button indicates the number of spaces each message class will be indented. By default, printing of all message classes begins at position zero (the first position on the left of the screen or printer), and the face of the *Space* button reflects this as ***sp:n***, where *n* represents the number of spaces defined for indentation.

Click on *Space*; the *Indentation* window (Figure 2-23) will appear. This window contains the possible choices for indenting messages, from the default of zero to the maximum of 36 spaces. Highlight the desired indentation by clicking on it, then click on *OK* to complete the selection process, or *Cancel* to exit the operation without making changes.

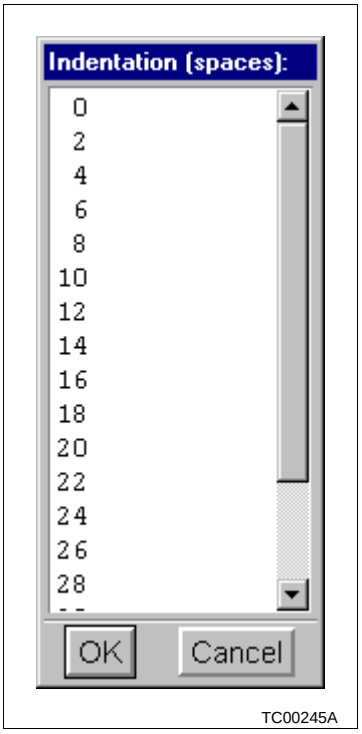


Figure 2-23. Message Indentation Definition Window

MESSAGE CLASS: PROCESS ALARM MESSAGES

Process alarm messages are displayed in the top two lines of the mini-alarm window, known as the alarm display area.

Process alarms are displayed in the following format:

<<Priority>> <<Area>> <<Time>> <<Date>> <<Tagname>> <<Alarm Condition>> <<Legend>>

Process alarm messages are classified into the following message types:

- Bad Input/output.
- High/Low alarm.
- Deviation alarm.
- Security alarm.
- Process alarm message.
- Discrete alarm.
- Return to normal.
- Alarm acknowledged.

Process alarm messages can be filtered by message type.

MESSAGE CLASS: SYSTEM EVENT MESSAGES

System event messages are displayed in the third line of the mini-alarm window, known as the event display area.

System event messages are displayed in the format of the following example:

<<Time>> <<Date>> INFI 90 OPEN CIU IS OFFLINE

System event messages are classified into the following message types:

- Operator station events.
- Network events.
- Return to normal.
- Event acknowledged.
- Node down.
- Node up.

System events can be filtered by message type.

MESSAGE CLASS: OPERATOR MESSAGES

Operator messages are messages sent by an operator to all operators in an area, and appear in the fourth line of the mini-alarm window.

Operator messages are displayed in the format of the following example:

<<Priority>><<Area>><<Time>><<Date>> Acid Tank Level High.

Operator messages are classified into the following message types:

- Operator message.
- Operator call.
- Message acknowledged.
- Operator call acknowledge.

Operator messages can be filtered by message type.

MESSAGE CLASS: OPERATING ERRORS

Operating error messages are sent to a device designated via the message routing assignment matrix. This device can be a printer or a file.

Operating error messages are displayed in the format of the following example:

<<Time>><<Date>><<Device>> Insufficient permission to Add/Remove Red Tags.

Operating error messages are classified into the following message type:

- Operating message.

Operating messages can be filtered.

MESSAGE CLASS: OPERATOR ACTIONS

Operator action messages are sent to a device designated via the message routing assignment matrix. This device can be a printer or a file.

Network operator action messages are displayed in the format of the following example:

<<Area>> <<Time>> <<Date>> <<Username>> <<Terminal Name>> PUT HS-5441.C_DPV NEW=1 OLD=0

PWC90 console operator action messages are displayed in the format of the following example:

<<Time>><<Date>><<Username>><<Terminal Name>>graphic 1 installed

Operator action messages are classified into the following message types:

- Network operator action.
- Operator action.

Operating messages can be filtered.

MESSAGE FILTER PARAMETER: PRIORITY

To filter messages by priority, click on the *Priority Message Class* button on the *Device Routing Assignment* window (Figure 2-21). The message type area of the window will list all priorities. Click on each priority for which routing is to be changed, or use the *Clear* or *Set* buttons to clear or set the routing status of all priorities.

MESSAGE FILTER PARAMETER: AREA

To filter messages by area, click on the *Area Message Class* button on the *Device Routing Assignment* window (Figure 2-21). The *Message Type* area of the window will list all areas. Click on each area for which routing is to be changed, or use the *Clear* or *Set* buttons to clear or set the routing status of all areas.

Operator Window Parameters

The *Operator Window Parameters* menu item is used to access features which allow the user to:

- Choose update rates for operator windows containing graphic, group, point or trend displays; the rate can further be set separately for windows both with and without keyboard focus.

- Determine the maximum allowable memory size of operator windows; this is used to avoid delays by triggering replacement of operator windows in memory.

When this menu item is clicked on, the operator window Parameters window (Figure 2-24) will be displayed.

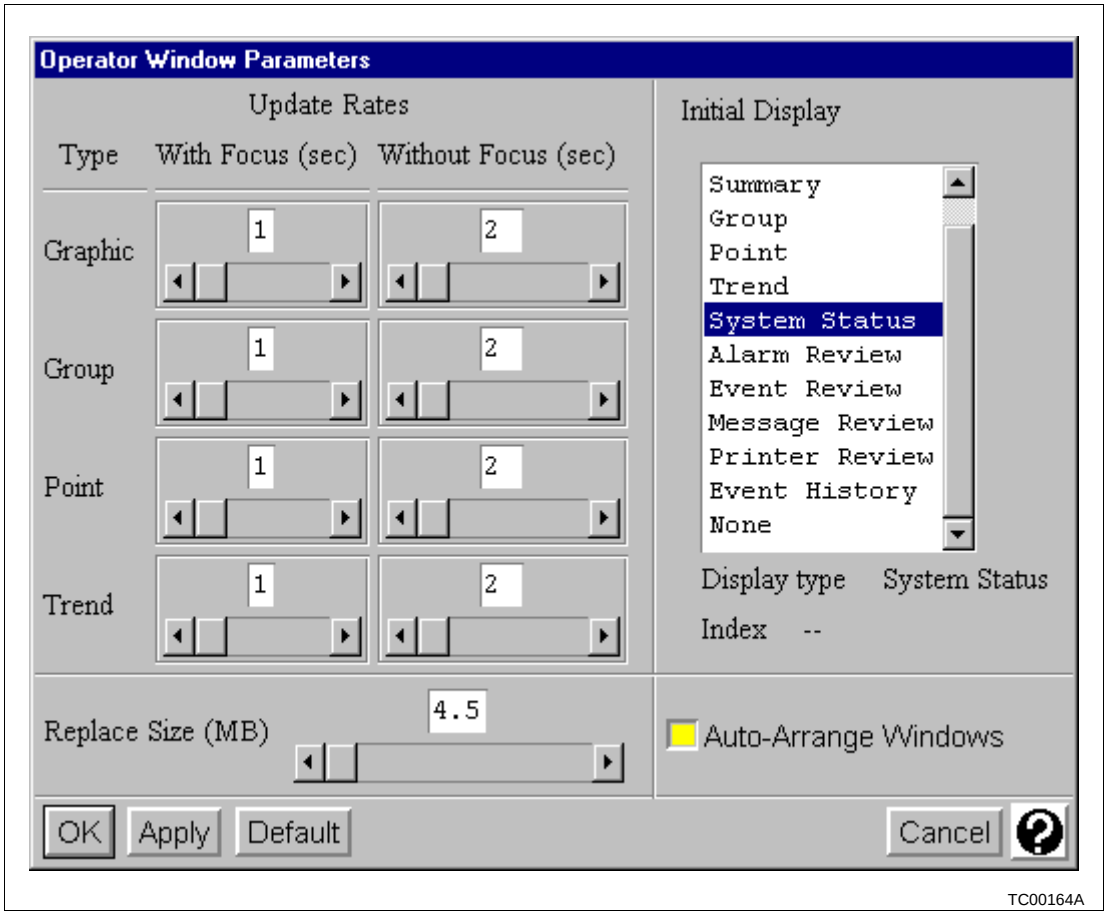


Figure 2-24. Operator Window Parameters Window

UPDATE RATES

The top portion of the operator window parameters window contains four pair of sliders used to determine how fast a win-

dow containing a graphic, group, point or trend display will be updated.

The first slider in each pair determines the update rate for windows containing the specified display type which has keyboard focus. Update rates for windows with keyboard focus range from one (the default) to five seconds.

The second slider in each pair determines the update rate for windows containing the specified display type, but which do not have keyboard focus. The minimum update rate for these windows is two seconds (the default), and the maximum is ten seconds.

For each display type, the update rate selected for the without focus state must be greater than or equal to the rate selected for the with focus state; i.e., the number of seconds chosen on the right slider must be greater than or equal to the number of seconds selected on the left slider. If this rule is not followed, using the *Apply* or *OK* will result in the display of a pop up window containing an *OK* button and the following error message:

Focus value > no focus value

Click on *OK* to close the error message window, then adjust the slider settings.

Changes made to the update rate are network wide and will have an immediate effect on all open operator windows containing graphic, group, point or trend displays; it is not necessary to close and re-open operator windows to change the rate at which the display is updated.

REPLACEMENT SIZE

Each time a new display is called up in an operator window, the amount of memory used by that window will increase. When very complex displays are used often in operator windows, a pop up window containing the following message may be encountered when a new display is called up:

Please wait...

This happens when the amount of memory used by an operator window has grown large. The system automatically replaces the operator window, in memory, with a new version using the minimum amount of memory required for the display, reducing the amount of memory consumed.

The center section of the *Operator Window Parameters* window contains a slider labelled *Replace Size (MB*). The number of megabytes selected via this slider determines the amount of system memory which can be used by operator windows before

they are replaced. The default setting is the minimum, 4.5 MB, which is the recommended setting. If few or no *Please Wait* messages are encountered, this setting should be left untouched.

If the *Please Wait* message is encountered often, the amount of memory available to operator windows can be increased by moving the *Replace Size* slider to the right. Replacement size can be increased to a maximum of 5.5 MB, in increments of 0.1 MB.

Increasing the replacement size will reduce the incidence of *Please Wait* pop up windows; it will, however, increase the amount of system memory used by operator windows.

RETURN TO DEFAULT SETTINGS

To return the settings on the *Operator Windows Parameter* window to the default settings, click on *Default* at the bottom of the window, then click on *OK* or *Apply* to save the default settings.

APPLY CHANGES

Once changes have been made to the items in the *Operator Window Parameters* window, the changes can be saved without exiting the window by clicking on the *Apply* button. This is especially useful when configuring update rates, as the user can watch the effect that changes to this feature produce on a display, and adjust the rates without closing and re-opening the configuration window.

EXITING THE OPERATOR WINDOW PARAMETER CONFIGURATION

To exit the *Operator Window Parameters* window, click on *OK* to close the window and save any pending changes, or click on *Cancel* to exit the configuration session without committing any changes made since the last time *Apply* was clicked.

DISPLAY COLORS

Choose the *Display Colors* option from the *Network Configuration* menu to access the *Display Colors* dialog (Figure 2-25). This dialog gives the ability to modify the color of the following components:

- Window text.
- Window background.
- Edit text.
- Edit text background.
- Button text.

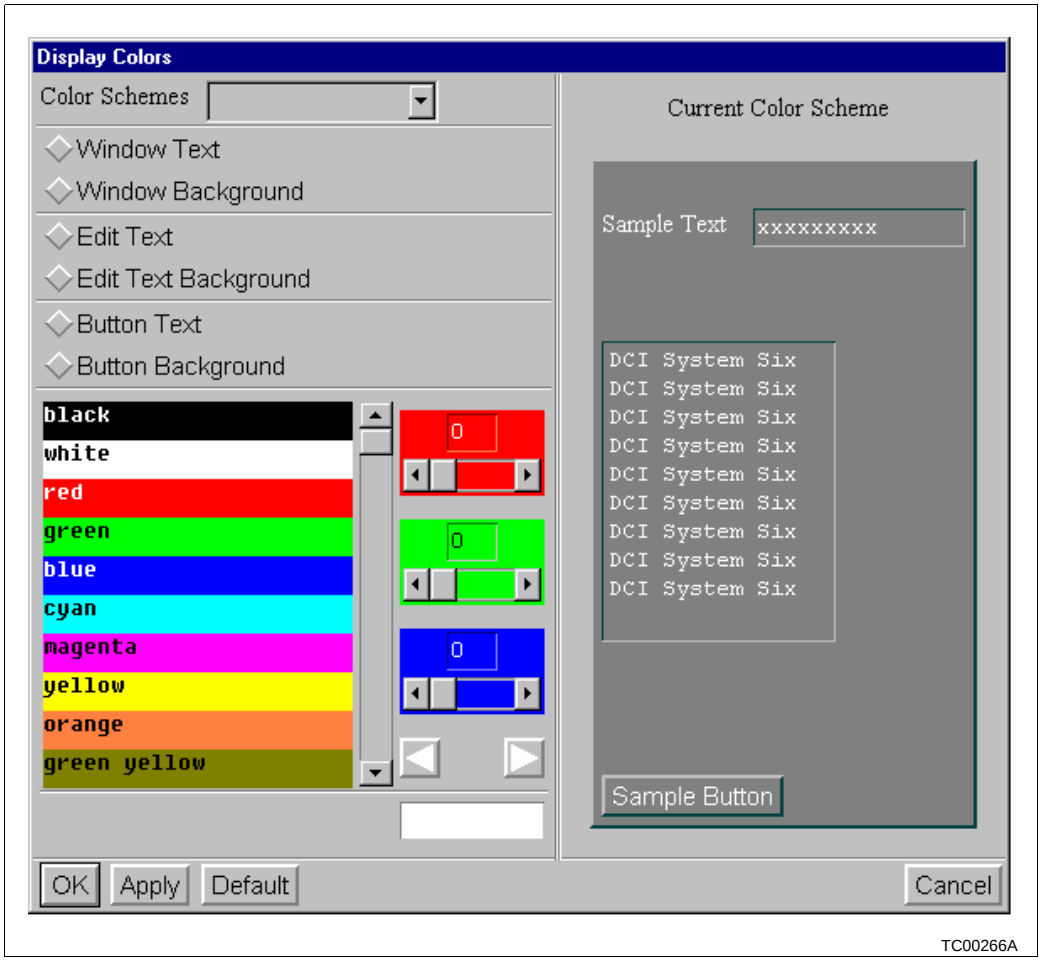


Figure 2-25. Display Colors Window

- Button background.

To change the color of one of these components, select the corresponding radio button beside that component, and then either click on one of the predefined 10 colors, or use the RGB sliders to select a custom color (which will be displayed below the predefined colors).

The current color scheme is displayed in the right half of the dialog for sample viewing. For easy resetting, four default color schemes are included in the *Color Schemes* pull down menu:

- Default.
- Grays.
- Browns.
- Blues.

Click on *Default* to reset to default colors. Once the colors are chosen, click on *Apply* to make the changes without exiting the *Display Colors* window, or click on *OK* to make the changes and exit the window. Click *Cancel* to exit the window without

applying the changes made since the last apply of the current session.

SECURITY

The PWC90 console security system:

- Provides network wide accounts.
- Defines user access to PWC90 console functions, process control functions and plant areas.
- Defines each user's filtering of alarms and messages by area, priority and message class.
- Defines each user's set of Quick keys.
- Provides assignment of security parameters for PWC90 console default condition (no user logged in) and each individual user.
- Permits activation or deactivation of the security checking.

At the initial system startup, the PWC90 console security system is in the deactivated state (refer to ***Security Activation*** in this section), and the condition known as the default login (described in ***Display Center Default*** in this section) exists. Until otherwise reconfigured, all functions are available directly from the PWC90 console, as the default login is initially set at supervisor level 3 (the highest access level.)

> **NOTE:** Read this section completely before reconfiguring the default login. Also, be sure to create a new privileged login account with an access level of supervisor level 3 before lowering the access level of the default login.

To configure PWC90 console security system features, click on the system window *Menu* button, then click on the *Configure* option in the resulting pull down menu, then click on the *Security* option in the resulting cascading menu. The security menu provides access to the following:

- Security activation.
- Display center default.
- Function Access.
- User login.
- Change password.

Security Activation

Security on the PWC90 console can be active or inactive. When security is inactive, the access level is set to that defined for the default login. (The default login, described in further detail in

Display Center Default in this section, is used at the initial PWC90 console startup, and whenever there is no user logged in on a PWC90 console.) Message filtering is enabled for all log-ins when security is inactive.

Selecting the *Security Activation* option from the *Security* menu displays the *Security Activation* window shown in Figure 2-26.

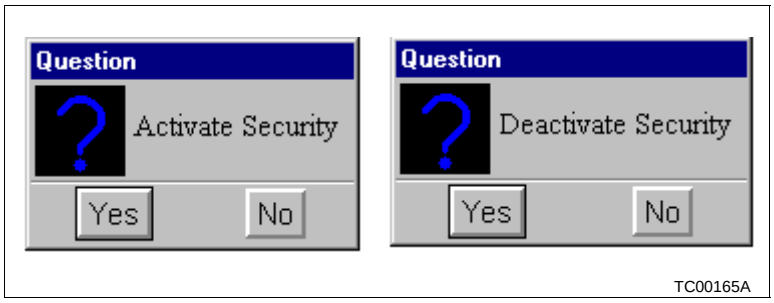


Figure 2-26. Security Activation/Deactivation Windows

If security has not been activated, the window prompts the user to *Activate Security*; if security has already been activated, the window prompts the user to *Deactivate Security*.

The *Yes* button is automatically highlighted; click on it, or press Return to take the action indicated by the prompt. This change takes place when the user exits the *Security Activation* window. To exit without making a change, click on *No*.

Display Center Default

Two security conditions exist on the PWC90 console:

- No user logged in.
- User logged in.

When no user is logged in (a state which occurs at system start-up or after a user logs out) some actions can still be performed at the PWC90 console. This state is known as the default login, and is denoted by the word *Default* in the mini-alarm window menu bar, where the user name is normally displayed. The scope of actions permitted to the default login depends on the configuration of the PWC90 console default.

Selecting *Display Center Default* from the *Security* menu displays the *Default Login Configuration* window shown in Figure 2-27.

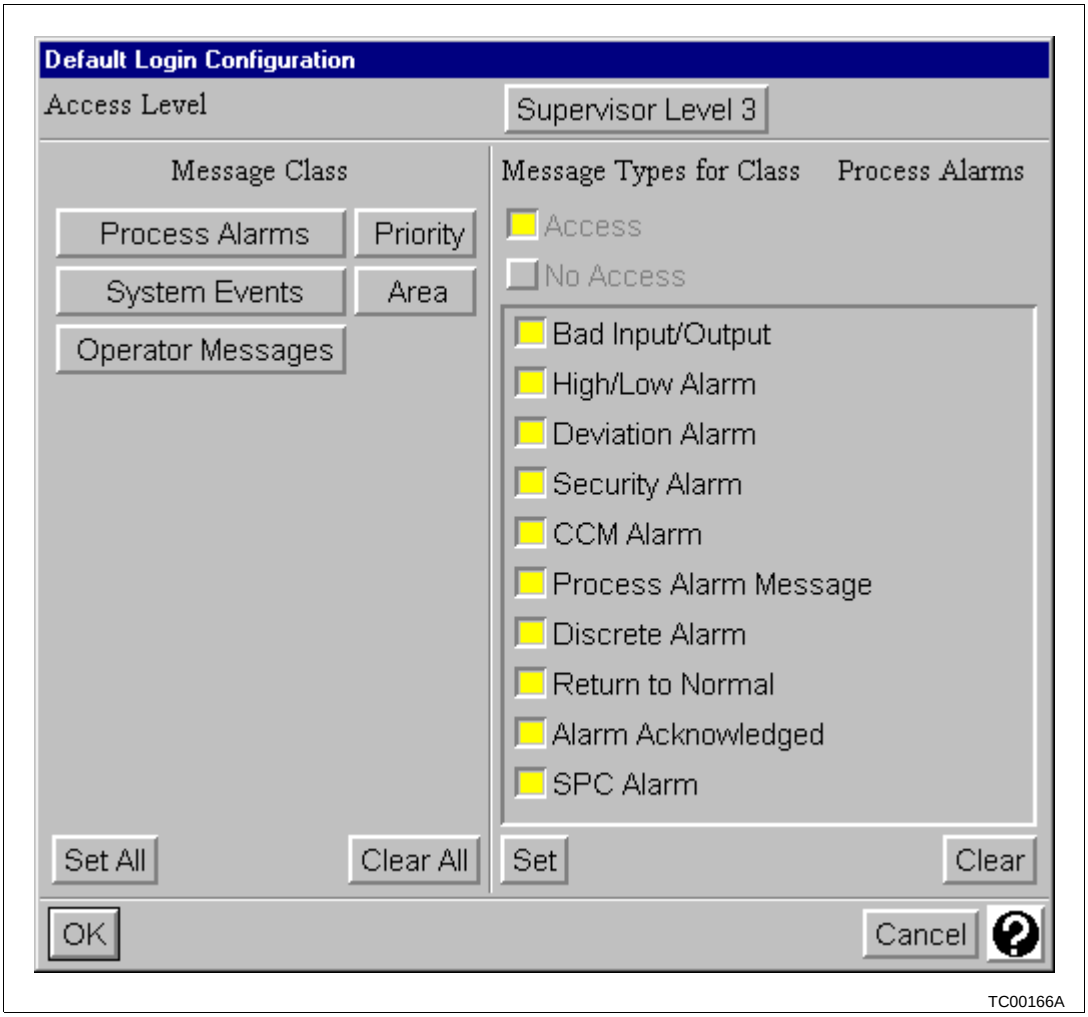


Figure 2-27. Default Login Configuration Window

DEFAULT LOGIN ACCESS LEVEL

At the top of the *Default Login Configuration* window is a button which displays the current *Access Level*. Access levels are assigned to both login accounts (including the default login account) and functions. To access a function, the access level of a user's login account must be equal to or higher than the access level assigned to the function. function access levels are configured using the *Function Access* option (refer to ***Function Access*** in this section) on the *Security* menu. User login accounts are assigned an access level using *User Login Configuration* (refer to ***User Login*** in this section).

To change the access level of the default login account, click on *Access Level*. The *Access Level Selection* window (Figure 2-28) will appear on the screen.

The *Access Level Selection* window displays the nine access levels available on the PWC90 console. Operator level one is the

lowest access level; normally users with this access level have the least permission to access PWC90 console functions. The highest available access level, supervisor level 3 should provide maximum access to functions. The current access level of the default login account is highlighted when the *Access Level Selection* window appears. To select a different access level, position the pointer on the desired level and click on it to highlight it. Click on *OK* to effect the change and exit the window, or click on *Cancel* to exit without making changes.

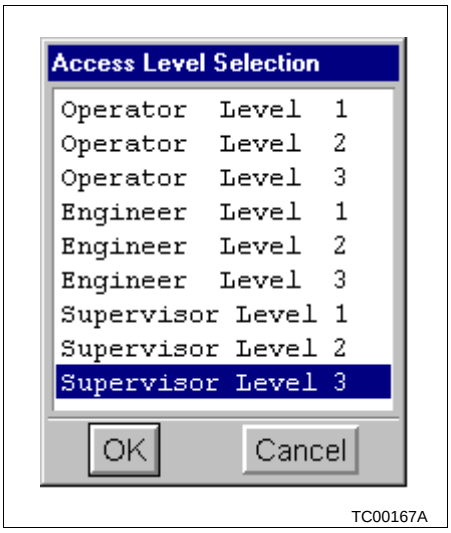


Figure 2-28. Access Level Selection Window

When the PWC90 console is started up initially, the default login is set to supervisor level 3. (The initial screen setup to be displayed, as defined by the CRT context keys for the default login, consists of one operator window containing the system status display.)

DEFAULT LOGIN MESSAGE FILTERING

The area of the *Default Login Configuration* window (Figure 2-27) is used to configure message filtering for the default login account.

On the left is a set of buttons, each representing a message class or other message filtering parameter, including:

- Process alarms.
- System events.
- Operator messages.
- Priority.
- Area.

Click on one of these buttons to select the message class (or parameter) for which message filtering is to be configured. This selection will appear at the top of the right side of the window.

The list of message types for that message class will be displayed in the right side of the window. Message classes and message types are discussed in ***Message Routing*** in this section.

The message type list is used to include or exclude (filter) message types out of the mini-alarm and alarm review for the default login. Each message type has a toggle button associated with it. Click on the toggle button to reverse the status of the message type; if the button appears to be pressed in, messages of that message type are accessible to the default login. If the button appears to be raised or out, no access is available to that message type. Sample buttons labeled *Access* and *No Access* appear above the message type list. The *Access* and *No Access* buttons are included only to provide a visual example of how the toggle button appears in each state, and have no other function.

By default, all message types in all message classes are accessible to the default login.

AREA ACCESS ASSIGNMENT (DEFAULT LOGIN)

The *Area* filter parameter serves a dual purpose. This item allows the user to determine not only which areas the default login can receive messages from, but also which areas can be accessed by the default login.

CLEAR AND SET MULTIPLE MESSAGE TYPES

It is possible to clear or set multiple message types simultaneously, eliminating the need to click on numerous toggle buttons. Click on *Set* or *Clear* to set or clear all message types in the selected message class. Click on *Set All* or *Clear All* to set or clear all message types in all message classes available via the default login on the PWC90 console.

EXITING DEFAULT LOGIN CONFIGURATION

To exit the *Default Login Configuration* window and save changes, click on *OK*. To exit without making changes, click on *Cancel*.

All parameter changes take effect when the user logs out of the PWC90 console or when the PWC90 console is restarted.

Function Access

Use the *Function Access* window to assign access levels to PWC90 console and network functions. Click on the *Function Access* option from the *Security* menu to display the *Function Access Configuration* window (Figure 2-29).

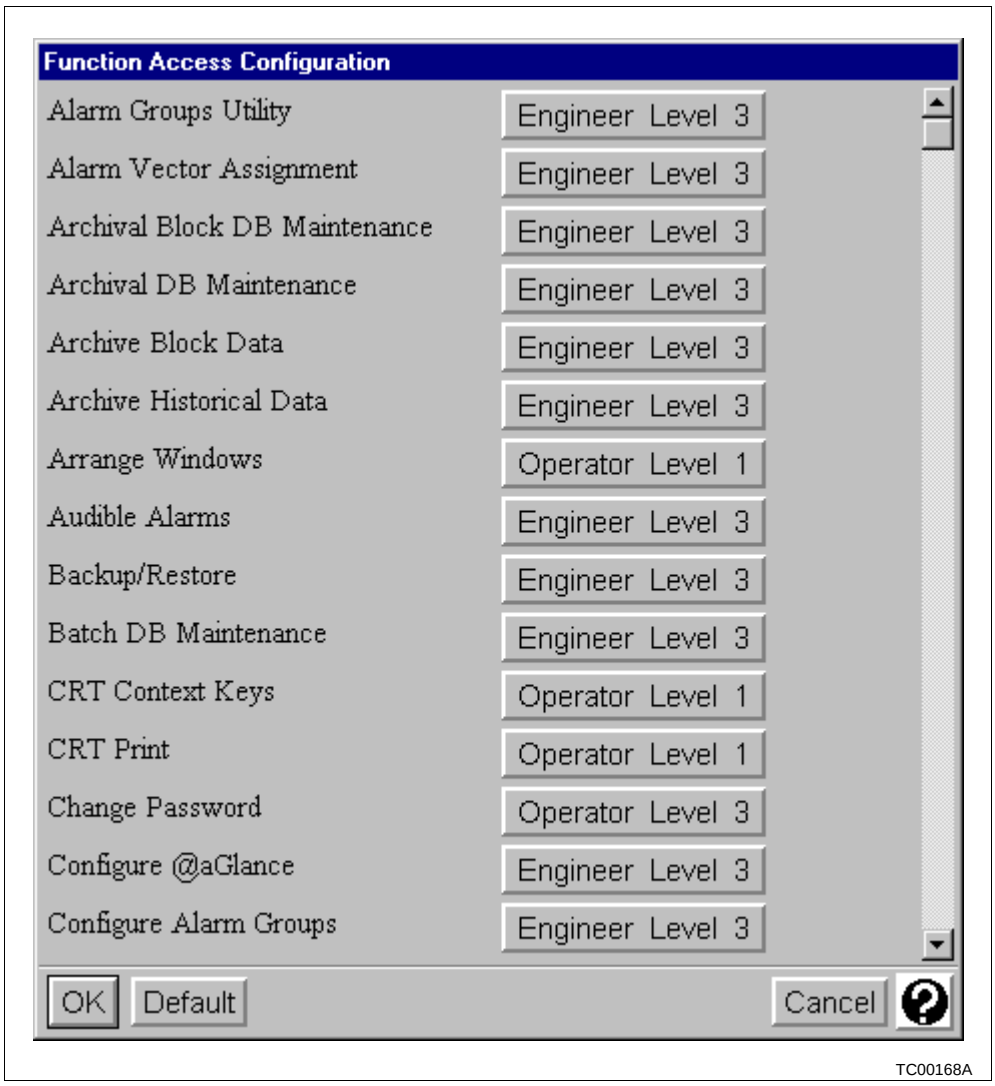


Figure 2-29. Function Access Configuration

This scrollable window contains a list of all function names available via the pull down and cascading menus on the PWC90 console. Each function name has an associated button which displays the access level (i.e., Engineer Level 3) assigned to the function. A default set of function access levels is configured on the PWC90 console. Table 2-4 shows the default access level assignments.

Table 2-4. Default Function Access Level Assignment

Function	Default Access Level
Alarm Groups Utility	Engineer Level 3
Alarm Vector Assignment	Engineer Level 3
Archival Block DB Maintenance	Engineer Level 3
Archival DB Maintenance	Engineer Level 3
Archive Block Data	Engineer Level 3

Table 2-4. Default Function Access Level Assignment (continued)

Function	Default Access Level
Archive Historical Data	Engineer Level 3
Audible Alarms	Engineer Level 3
Arrange Windows	Operator Level 1
Backup/Restore	Engineer Level 3
CRT Context Keys	Operator Level 1
CRT Print	Operator Level 1
Change Password	Operator Level 3
Configure @aGlance	Engineer Level 3
Configure Alarm Groups	Engineer Level 3
Configure Alarm Setup	Supervisor Level 1
Configure Archival Groups	Engineer Level 3
Configure Area Names	Supervisor Level 1
Configure Configuration Reports	Engineer Level 3
Configure PWC90 console Default	Supervisor Level 1
Configure Event Bit Names	Supervisor Level 1
Configure Function Access	Supervisor Level 1
Configure Graphic	Engineer Level 3
Configure Historical Blocks	Engineer Level 3
Configure Historical DB	Engineer Level 3
Configure Historical Trend	Engineer Level 3
Configure Logs	Engineer Level 3
Configure Message Routing	Supervisor Level 1
Configure Messages	Engineer Level 3
Configure Operator Window Params	Supervisor Level 1
Configure Spreadsheet Displays	Engineer Level 3
Configure Summary	Engineer Level 3
Configure User Login	Supervisor Level 1
PWC90 Users	Engineer Level 3
Display Configuration Reports	Operator Level 3
Edit User Help	Operator Level 3
Filter Alarms in Mini-Alarm	Operator Level 1
Historical Block DB Maintenance	Engineer Level 3
Historical DB Maintenance	Engineer Level 3
Manual Data Entry	Engineer Level 3
Network Device Assignment	Supervisor Level 3
Network Level Shutdown	Supervisor Level 3
Open New Window	Operator Level 1
Other Applications	Engineer Level 3
Print Log	Engineer Level 3
Printer Assignment	Supervisor Level 1

Table 2-4. Default Function Access Level Assignment (continued)

Function	Default Access Level
Program Configuration	Supervisor Level 1
Program Scheduling	Supervisor Level 1
Release Configuration Locks	Operator Level 3
Restore Block Data	Engineer Level 3
Restore Historical Data	Engineer Level 3
Security Activation	Supervisor Level 3
Spreadsheet Displays	Engineer Level 3

To change the access level of a function, click on the button next to that function. The *Access Level Selection* window (Figure 2-28) will appear. (This is the same window which appears when configuring message filtering for the default login and user accounts.)

Click on the new access level to be assigned to the function; the selection will be highlighted. Click on *OK* to make the assignment, or click on *Cancel* to exit without making changes.

Changes take place when the user exits from *Function Access* configuration. A function currently in use will not be affected until the next time it is accessed.

Server and Tag Access

Security access to particular server and/or tag functionality may be configured through the Tag Configuration program, through the editing of certain fields of the I90ScannerTag. Setting the value for these fields dictates the minimum access requirements to perform the specific function. Values for the fields in Table 2-5 may be set.

NOTE: Making any change to the I90Scannertag requires Supervisor Level 3 access. This is not configurable.

Table 2-5. Server and Tag Access Fields

Field	Description	Default Level
Read operations	Allows user to read information from the server.	Read-only (not currently enforced)
Control	Allows user to perform control operations.	Operator Level 1
Scan State Changes	Allows the user to modify the state of a tag.	Operator Level 2
Tag Database Change	Allows the user to modify the tag configuration database.	Engineer Level 1
Inhibit Alarms	Allows user to inhibit alarms on tags.	Operator Level 2
Add/Remove Red Tags	Allows the user to add or remove red tags from tags.	Operator Level 2

Table 2-5. Server and Tag Access Fields (continued)

Field	Description	Default Level
Control Red Tags	Allows the user to override tags which are red-tagged, and perform control operations.	Supervisor Level 1
Tune tags	Allows user to perform tuning operations on blocks.	Engineer Level 1

Function Access Level Assignment Guidelines

When assigning function access levels, it is important to remember that care must be taken to maintain an adequate level of security by selecting the appropriate function access level for a function. The following guidelines were used to make the default *Function Access Level* assignments:

Supervisor Level:

- Security functions (except changing password).
- Functions used in configuration of network wide parameters.

Engineer Level:

- Functions used in configuration and maintenance of process displays.
- Functions used in configuration and maintenance of logs.
- Functions used in configuration and maintenance of Historical database.

Operator Level:

- Access to displays.
- Configuration of user help text.
- Changing of password.

User Login

Unique user login names can be configured on the PWC90 console for use on all PWC90 console nodes on the INFI 90 OPEN system. These names can be up to 20 characters in length and must be unique on the network, as they are contained in a table which is distributed network wide. There is no fixed limit to the number of login names which can be configured.

To configure a user login, click on *User Login* in the *Security* menu. The *User Login Configuration* window (Figure 2-30) appears.

Figure 2-30. User Login Configuration Window

The *Login Configuration* window contains a list box and a number of buttons. If any logins have already been configured, the usernames of the logins appear in the list box.

ADD A USER LOGIN

To create a user login, click on *Add*. The *User Login* window (Figure 2-31) will be displayed.

The top portion of the *User Login* window contains several text entry boxes and push buttons.

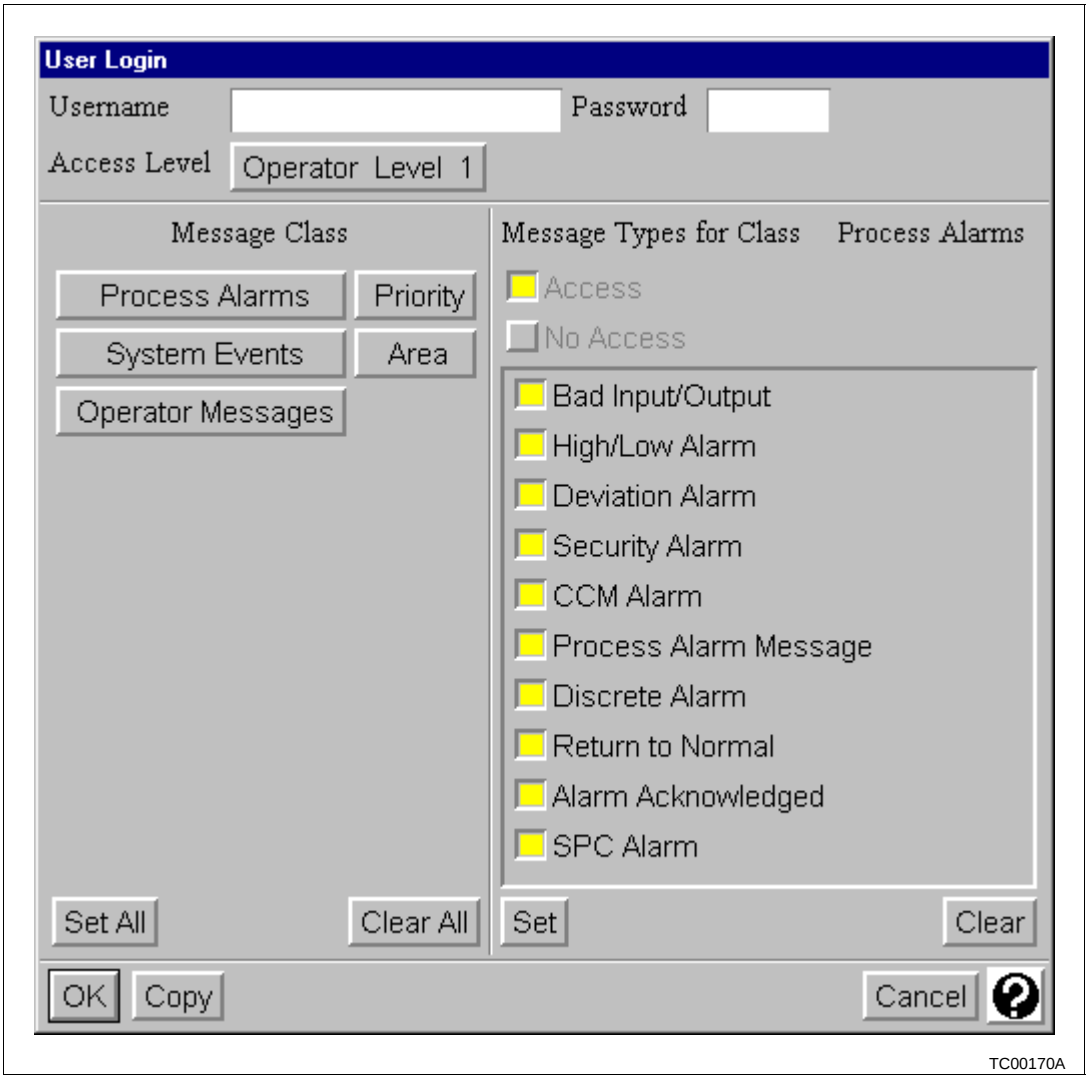


Figure 2-31. User Login Window

User Name

The entry box in the top left corner is labelled *Username*. Enter the Name (to a maximum of 20 characters) for the new user. The *Username* can be the name of a person (Cheryl, Clifford, Bob, etc.), or it can be a descriptive term such as *Operator, Supervisor, Sr. Engineer*, and so on. Usernames can be entered in either upper or lower case, depending on the user's preference.

NOTE: Remember that the PWC90 console is case sensitive, and the manner in which the username is configured is the way it must be entered each time the user logs in. It may be more convenient to use all upper or all lower case characters, rather than a mixture of the two.

User Password

The *Password* text entry box follows the *Username* text entry box. The password can have a maximum of seven ASCII characters, and defaults to the first seven characters of the username. To change the password, click on the associated entry box, use **Backspace** to erase the default password, and enter the desired ASCII string.

> **NOTE:** Unsuccessful user login attempts will cause a system event message to be generated. This message will be displayed in the system event display area of the mini-alarm window and stored in the event historian. The message will include the time and date, the node name, and a message in the following format:

Example: User <current user login name> Invalid Login Attempt on Terminal <PWC Name> <attempted user login name>

Where <current user login name> is the name of the user currently logged in, the <PWC Name> is the identity of the PWC90 console on which the attempt is made and <attempted user login name> is the unsuccessful user login name.

For example, if the user "fred" attempted to login using an incorrect password at a PWC90 console where the default login state existed, the following message would appear:

Example: 16:42:43 10-JUL-95 PWC1 User Default Invalid Login Attempt on Terminal PWC1_PWC:0 fred.

User Access Level

Access levels (previously discussed in ***DEFAULT LOGIN ACCESS LEVEL*** in this section) are assigned to both login accounts (including the default login account) and functions. To access a function, the access level of a user's login account must be equal to or higher than the access level assigned to the function. Function access levels are configured using the function access option (refer to ***Function Access*** in this section) on the *Security* menu.

New logins default to Operator Level 1, the lowest of the nine access levels on the PWC90 console. The *Access Level* button, located below the *Password* entry box on the *User Login* window, displays the current *Access Level*.

To change the access level of a login, click on the *Access Level* button. The *Access Level Selection* window (Figure 2-32) will appear on the screen. Note that the *Access Level Selection* window is the same as that used when configuring *Function Access* and the default login access level.

The *Access Level Selection* window displays the nine access levels available on the PWC90 console. Operator level 1 is the low-

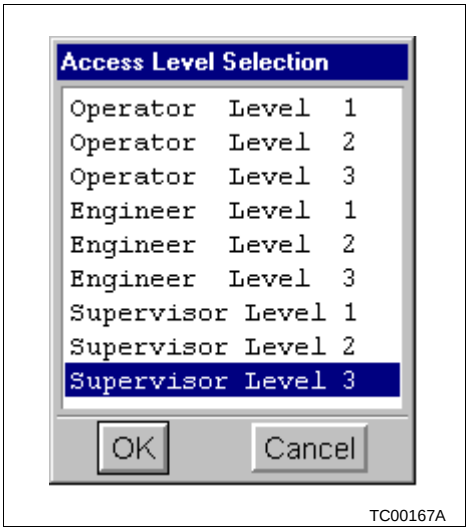


Figure 2-32. Access Level Selection Window

est access level; normally users with this access level have the least permission to access PWC90 console functions. The highest available access level, Supervisor level 3, should provide maximum access to functions. The current access level of the login account is highlighted when the *Access Level Selection* window appears. To select a different access level, position the pointer on the desired level and click on it to highlight it. Click on *OK* to effect the change and exit the window, or click on *Cancel* to exit without making changes.

User Message Filtering

The area of the *User Login* window (Figure 2-31) below the *Access Level* button is used to configure message filtering for the Login account.

The display is the same as that used to configure message filtering for the default login account (refer to ***DEFAULT LOGIN ACCESS LEVEL*** in this section). On the left is a set of eight buttons, each representing a message class or other message filtering parameter, including:

- Process alarms.
- System events.
- Operator messages.
- Priority.
- Area.

Click on one of these buttons to select the message class (or parameter) for which message filtering is to be configured. This selection will appear to the right of the *Message Types For Class* label. The list of message types for that message class will be displayed in the right side of the window. Message

classes and message types are discussed in ***Message Routing*** in this section.

The message type list is used to include or exclude (filter) message types out of the mini-alarm and alarm review for the login. Note that each message type has a toggle button associated with it. Click on this toggle button to reverse the status of the message type; if the button appears to be pressed in, messages of that message type are accessible to the login. If the button appears to be raised or out, no access is available to that message type. Sample buttons labeled *Access* and *No Access* appear above the message type list. The *Access* and *No Access* buttons are included only to provide a visual example of how the toggle button appears in each state, and have no other function.

By default, all message types in all message classes are initially set to the *Access* state when the login account is created.

Area Access Assignment

The *Area* filter parameter serves a dual purpose. use this item to determine not only which areas the user login can receive messages from, but also which areas can be accessed by the user login.

Clear and Set Multiple Message Types

It is possible to clear or set multiple message types simultaneously, eliminating the need to click on numerous toggle buttons. Click on *Set* or *Clear* to set or clear all message types in the selected message class. Click on *Set All* or *Clear All* to set or clear all message types in all message classes on the PWC90 console.

COPY A USER LOGIN

Once a login has been configured, it is possible to assign some or all of the same parameters to another login. The *Copy* button on the *User Login* window provides this ability.

For example, a login with the username User_1 has been created. A new login with the same permissions and the username User_2 is needed. To accomplish this, first click on the *User Login* item on the *Security* menu to call up the *User Login Configuration* window (Figure 2-30). Click on *Add*. In the resulting *User Login* window, click on *Username* entry box to gain focus on it, then enter the name of the new user (i.e., User_2), then click on *Copy*, located next to the *OK* button, near the bottom left corner of this window. The *Copy User Login* window (Figure2-33) appears.

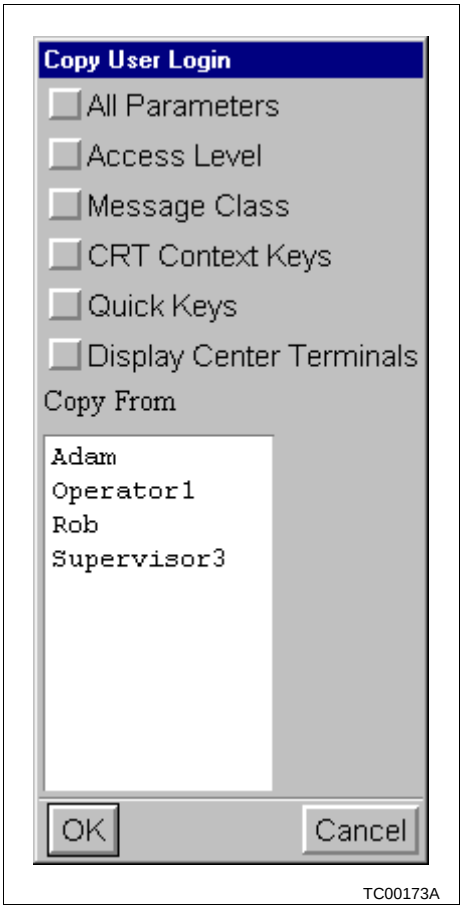


Figure 2-33. Copy User Login Window

A list box in the *Copy User Login* window contains all the usernames of logins currently configured on the PWC90 console. This list box is labelled *Copy From*. Locate the username to serve as the source for the *Copy* operation, and click on it to select it.

Above the list box is a set of choices; each choice has a toggle button associated with it. The toggle buttons allow the user to select which parameters are to be copied to the new login. Parameters which can be selected include:

- All parameters.
- Access level.
- Message class.
- CRT context keys.
- Quick keys.
- PWC90 consoles.

Click on the toggle button for each type of parameter to be copied from the login selected in the list box. Click on *OK* to complete the operation, or click on *Cancel* to exit without making changes.

EDIT A USER LOGIN

To revise an existing user login, click on the *User Login* item from the *Security* menu to call up the *User Login Configuration* display (Figure 2-30). Locate the desired user name in the list box, using the scroll bar if necessary. Click on the username to select it, then click on *Edit*. The *User Login* window (Figure 2-31) will appear, containing the configuration for the selected username. Make changes to the existing parameters as described in the paragraphs above.

Changes take effect when the user next logs into the PWC90 console.

DELETE A LOGIN

To delete an existing login, click on the *User Login* item from the *Security* menu to call up the *User Login Configuration* display. Locate the desired user name in the list box, using the scrollbar if necessary. Click on the username to select it, then click on *Delete* at the bottom of the *User Login Configuration* display. A confirmation box will appear, which asks:

Delete selected items?

Click on *Yes* to delete the selected login, or click on *No* to exit without deleting anything.

CANCEL LOGIN CONFIGURATION

To exit the *User Login Configuration* window (Figure 2-30) and abandon any changes made, click on *Cancel* located at the bottom of that window.

Change Password

When a login is initially configured, the password for the login defaults to the first seven characters of the username entered for the login. One of a new user's first actions should be to change this password, to prevent unauthorized persons from using the account.

A good password is an essential component of system security. The user should use the following guidelines when choosing a password:

- Avoid the obvious choices, such as names, social security numbers, words that appear in a dictionary, license plate numbers, etc.

- Make it complex - use a mixture of upper and lower case letters, as well as numbers, in the password.

- Use all seven characters - a password that is too short is easy to guess.

While the user is not prevented from using a single character or spouse's name as a password, this is not good practice.

To change a password, login on the system and click on the *Change Password* item from the *Security* menu. The *Change Password* window (Figure 2-34) appears.

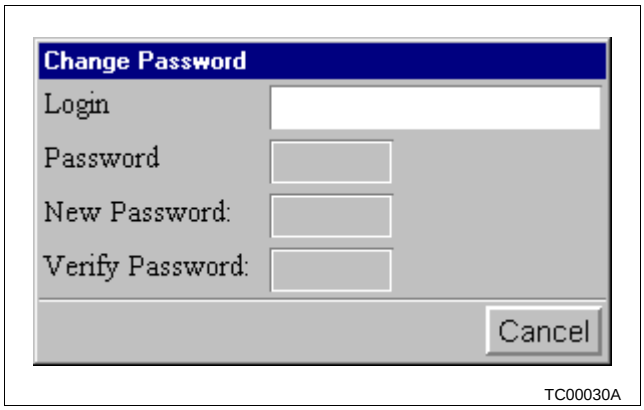


Figure 2-34. Change Password Window

Enter a valid username in the text entry box labelled *Login*, followed by Return. Next, enter the current password for that username in the *Password* text entry box, again followed by Return. Note that the password will not be displayed on the screen, for security reasons. For most users, this will be one's own username and password, although it is possible for a user with a higher access level to change the password of a user with a lower access level. Note that the user is forced to prove possession of the original password before it can be changed.

Next, enter the *New Password* in the appropriate text entry box, followed by Return. Enter the new password again, this time in the *Verify Password* text entry box. Because the passwords do not appear on the screen, this step is essential to make sure that there were no typing mistakes when the new password was entered.

After the new password is typed in the *Verify Password* text entry box, enter Return to complete the operation. If the new password and the verification entry match, the new password is accepted and the *Change Password* window is removed from the screen. If they do not match, a pop up window (Figure 2-35) appears, advising the user to try again.

Figure 2-35. Illegal Entry Pop Up Window

Summary - Security

Security The PWC90 console security system includes the mechanisms for activating/deactivating security checking, creating network wide user login accounts, defining access levels of PWC90 console functions, process control functions and plant areas, defining each user's set of Quick keys, and defining security parameters for both the PWC90 console default condition and for each individual user.

Access Level Access levels are parameters used to control the ability of users to access PWC90 console functions. Nine access levels are available on the PWC90 console, and are configured for *User Login* accounts, the *Default Login* account, and for all PWC90 console functions (via the *Function Access* menu option from the *Security* menu). The lowest access level (least permission) is Operator Level 1; the highest (all functions accessible) is Supervisor Level 3. The access level of a user must be equal to or higher than the access level of the function in order for the user to access the function.

Add Button The *Add* button appears at the bottom of a number of configuration windows. Click on this button to enable creation of additional entries of the type defined in the window, i.e., user logins.

Clear Button The *Clear* button appears on both the *Default Login Configuration* and *User Login* configuration windows. Click on this button to remove access to all message types for the selected message class.

Clear All Button The *Clear All* button appears on both the *Default Login Configuration* and *User Login* configuration windows. Click on this button to remove access to all message types for all message classes.

PWC90 Console Default The PWC90 console default item on the *Security* menu calls up the *Default Login Configuration* window. This allows the user to define the access level, message classes and message types available when the *Default Login* condition exists. (The *Default Login* condition exists when the system starts up or when no

user is logged in on the PWC90 console.) ***Display Center Default*** in this section provides additional details on this item.

Edit Button

The *Edit* button appears at the bottom of a number of configuration windows. Click on this button to edit entries of the type defined in the window, i.e., user logins.

Function Access

The access levels of PWC90 console and network functions are configured using the *Function Access Configuration* window. This window contains a list of all function names available via the pull down and cascading menus on the PWC90 console. Next to each function name is a button which displays the current access level of the function. To change the access level of a function, click on this button; next, click on the new access level in the resulting *Access Level Selection* pop up window. Click on *OK* to complete the assignment, or click on *Cancel* to exit without making changes.

Password

A password is a string of up to seven ASCII characters which must be supplied by a user attempting to login on the PWC90 console. A user's password should be kept confidential; to help maintain this confidentiality, the PWC90 console does *not* echo the password on the screen while it is typed. When a login account is configured, the password for that account defaults to the first seven characters in the username. To change a password, click on the *Change Password* item from the *Security* menu. Enter the username and current password (each followed by Return) in the resulting *Change Password* window. Type the new password in response to the *New Password* prompt, and again in response to the *Verify Password* prompt. Since the password is not echoed on the screen, this verification step insures that an accidental typing error will not prevent the user from gaining access via the new password.

Remember that the PWC90 console is case sensitive; upper and lower case letters are *not* interchangeable.

Set Button

The *Set* button appears on both the *Default Login Configuration* and *User Login* configuration windows. Click on this button to permit access to all message types for the selected message class.

Set All Button

The *Set All* button appears on both the *Default Login Configuration* and user login configuration windows. Click on this button to permit access to all message types for all message classes.

User Login

A user login (often referred to simply as a login) is an individualized set of parameters configured for each user (or type of user) on the system. The use of user logins enhances the security of a system. Use the *User Login* item on the *Security* menu to configure login accounts. ***User Login*** in this section provides additional details.

Username

The username is the string of up to 20 characters which will be used to login to the PWC90 console. This can be the name of a person (Smith, mary, etc.) or it can be a descriptive term such as Operator, Sr. Engineer, and so on. Usernames are entered in the *Username Selection* entry box on the *Login Configuration* window, which is available via the *User Login* item on the *Security* menu. The PWC90 console is case sensitive, so the username must be entered exactly as it has been configured. It may be convenient to use all upper or lower case letters, instead of a mixture of the two.

Once the user has logged in, the username will appear on the title bar of the mini-alarm window.

SECTION 3 - HIERARCHICAL DISPLAYS

INTRODUCTION

A number of displays and functions are accessed by clicking the *Select* item on the operator window menu bar. These include:

- Graphic displays.
- Summary displays.
- Group displays.
- Point displays.
- Quick keys.
- Trend displays.
- System status display (includes database options).
- Alarm review.
- Event review.
- Message review.
- Event historian.
- Printer review.

The first four provide access to screens which can include any addressable point in any INFI 90 OPEN server database. A distinction is made in this configuration instruction between graphic displays and hierarchical displays.

Graphic displays are completely configurable. These displays can include objects and submodels such as line segments, equipment symbols and bargraphs; they can include process database parameters such as tagnames and process values. Graphic displays can contain plant process diagrams, complete with dynamically updated data, colors, and other display attributes. Graphic display configuration is accessed from the *Configure* option available from the system window *Menu* button, and is discussed in Section 4 of this instruction.

Hierarchical displays are pre-formatted displays provided by the PWC90 software, and organized into a hierarchy; the format of hierarchical displays cannot be modified. The following displays are considered hierarchical displays:

- Summary displays (128 maximum).

- Group displays (24 per summary display, 3072 maximum).

- Point displays (four, six or eight per group display) are available for each tag.

  NOTE: The graphical objects used to create the group and point displays are stored on the PWC90 console as submodels, making it possible for the user to create graphics displays composed of the same objects which make up the hierarchical displays. This allows the user to create graphics which have the appearance of modified group or point displays. Refer to Section 4 for details on graphic configuration.

The user can assign points to these displays and specify legends for each display by using the methods described in this section.

In order to configure hierarchical displays, an operator window must be opened and must have keyboard focus. Click on *Select* in the operator window menu bar, then click on one of the hierarchical display items (*Summary, Group* or *Point*) in the resulting pull-down menu (Figure 3-1).

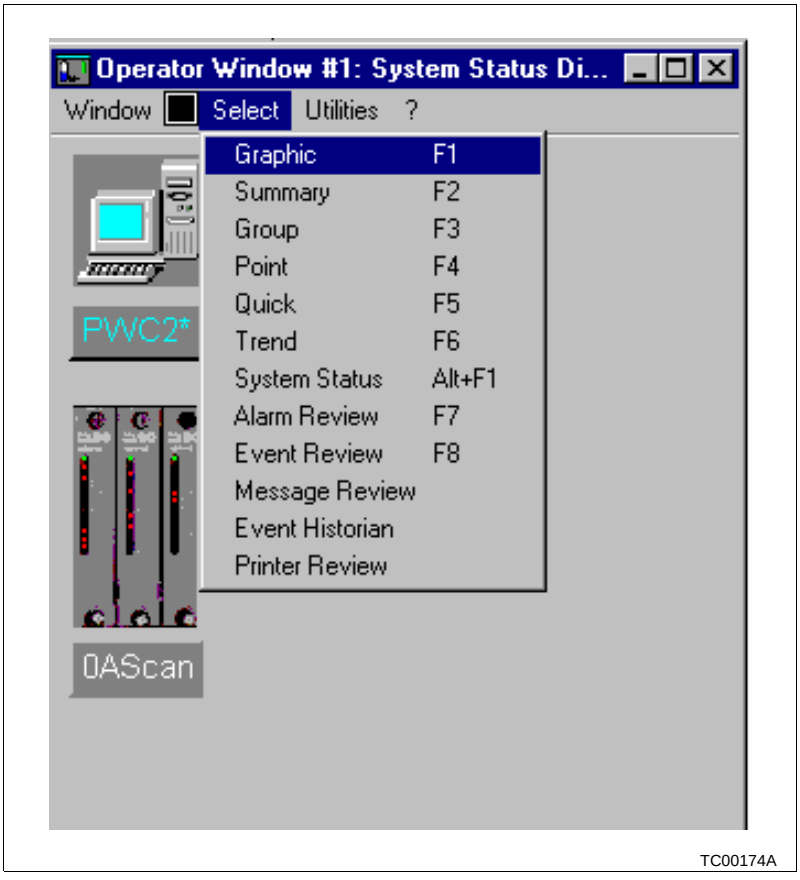


Figure 3-1. Operator Window Select Menu

HIERARCHICAL DISPLAY DESCRIPTIONS

The purpose of the following section is to describe how the user can organize the hierarchy of process point displays, which are an enhancement of conventional panel instrument displays.

With the pre-formatted hierarchical displays available on a PWC90 console, the user can oversee a process or an area of responsibility, as well as focus on a single point. The PWC90 console display hierarchy ranges from a summary display that presents the status of the points in 24 groups, to a single point display for detailed examination and manipulation of process variables.

Plant Summary Display

A summary display provides an overview of 24 groups and is arranged in six rows of four blocks, each representing a group (Figure 3-2). A legend for the 24 groups is displayed at the top of the screen. There is no restriction on the number of characters in the legend; however, the legend, which is centered, can extend beyond both edges of the display if it is longer than 45 to 50 characters.

A pushbutton is located in each block on the summary display; the pushbutton contains the number of the group represented by the block (1 through 24). The status of the group is indicated by the color of the pushbutton and the text which appears to the right of the pushbutton. The most serious status which exists within the group of points is the status which will be displayed. Table 3-1 lists status conditions in order of decreasing severity, with the most serious condition at the top.

Click on one of the pushbuttons to display the *Group Info* window, which displays the status of each point in the group and provides access to the group display or to individual point displays. Refer to the ***Personal Work Center 90 Operation*** instruction for a discussion of the *Group Info* window.

There are 128 summary displays available.

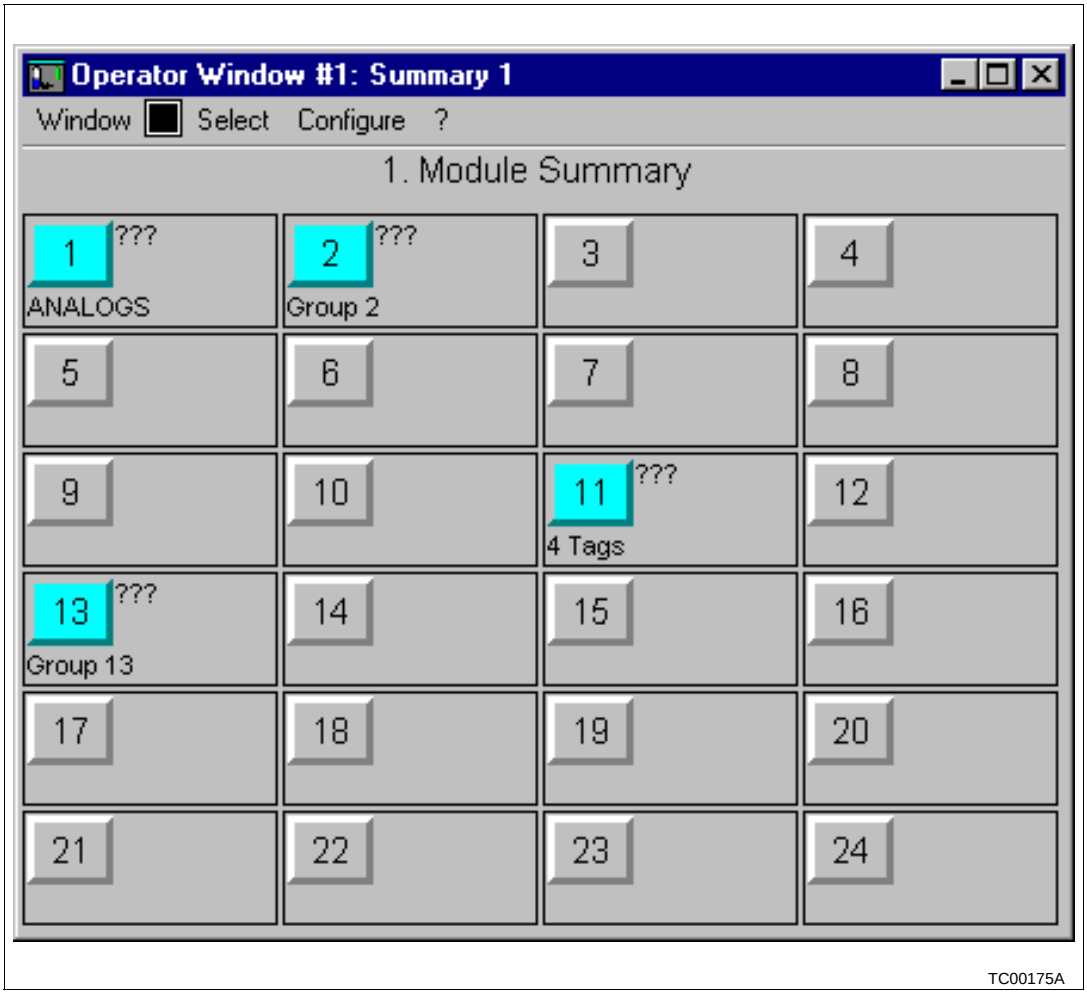


Figure 3-2. Summary Display Example

Table 3-1. Group Status Conditions

Color	Text	Status
Red	Unack. Alarms	At least one point in the group is in an alarm state, and the alarm is unacknowledged.
Yellow	(Unack).	At least one point in the group was previously in alarm, and, although the point in alarm has returned to normal, the alarm has not been acknowledged.
Yellow	Alarms	At least one point in the group is in an alarm state, and the alarm has been acknowledged.
Cyan	???	The status of at least one point in the group is unknown because the point has bad quality and is indicated in the server.
Green	[no text]	All points in the group are in normal state or alarm inhibit state, and no alarms need acknowledgment.
Gray	[no text]	If the group has not been configured, the pushbutton appears in gray.

Group Display

A group display provides an overview of four, six or eight points (Figure 3-3). The top line displays a legend for the group. (There is no restriction on the number of characters in the legend; however, the legend, which is centered, can extend beyond both edges of the display if it is longer than 45 to 50 characters). The display is divided into four, six or eight blocks; the information within each block relates to one point.

POINT TAGNAMES

At the top of each block on the group display is a button which contains the first 12 characters of the point's tagname. If the tagname is longer than 12 characters, the last (twelfth) character displayed in the tagname is a tilde (~). Click on this button to call up the point display for that tagname in the operator window.

POINT LEGENDS

The point legend is displayed below the tagname in three lines of up to 12 characters each, for a total of 36 characters. If it is necessary to truncate the point legend (which can be configured to a maximum of 48 characters), the thirty-sixth character is a tilde (~). There are 3072 group displays available.

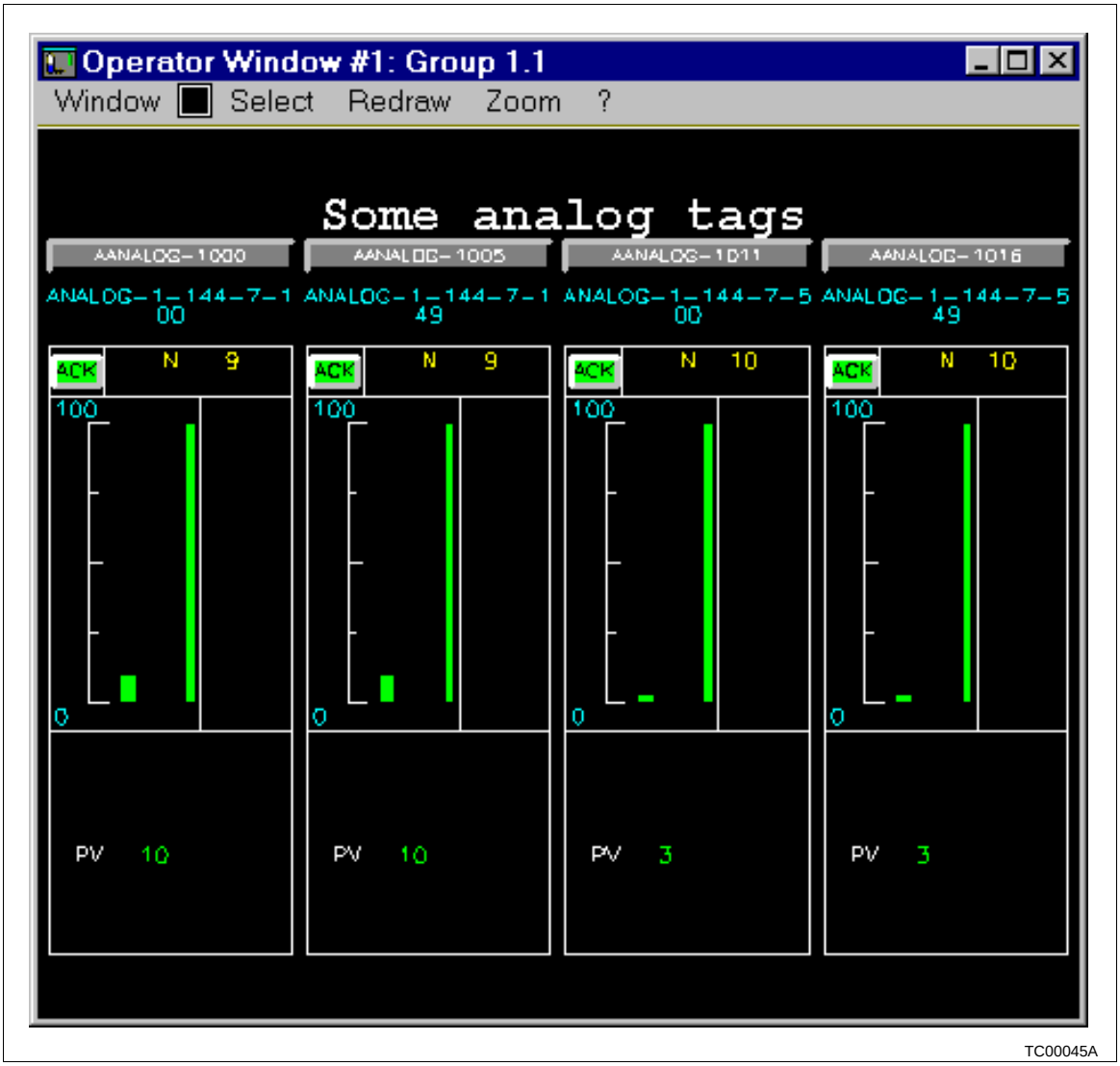


Figure 3-3. Group Display Example

Point Display

The point display provides a more detailed display of one particular tag from the group display. The point display includes a 48 character legend associated with the tag, but maintained in the tag database. Figure 3-4 provides an example of a point display.

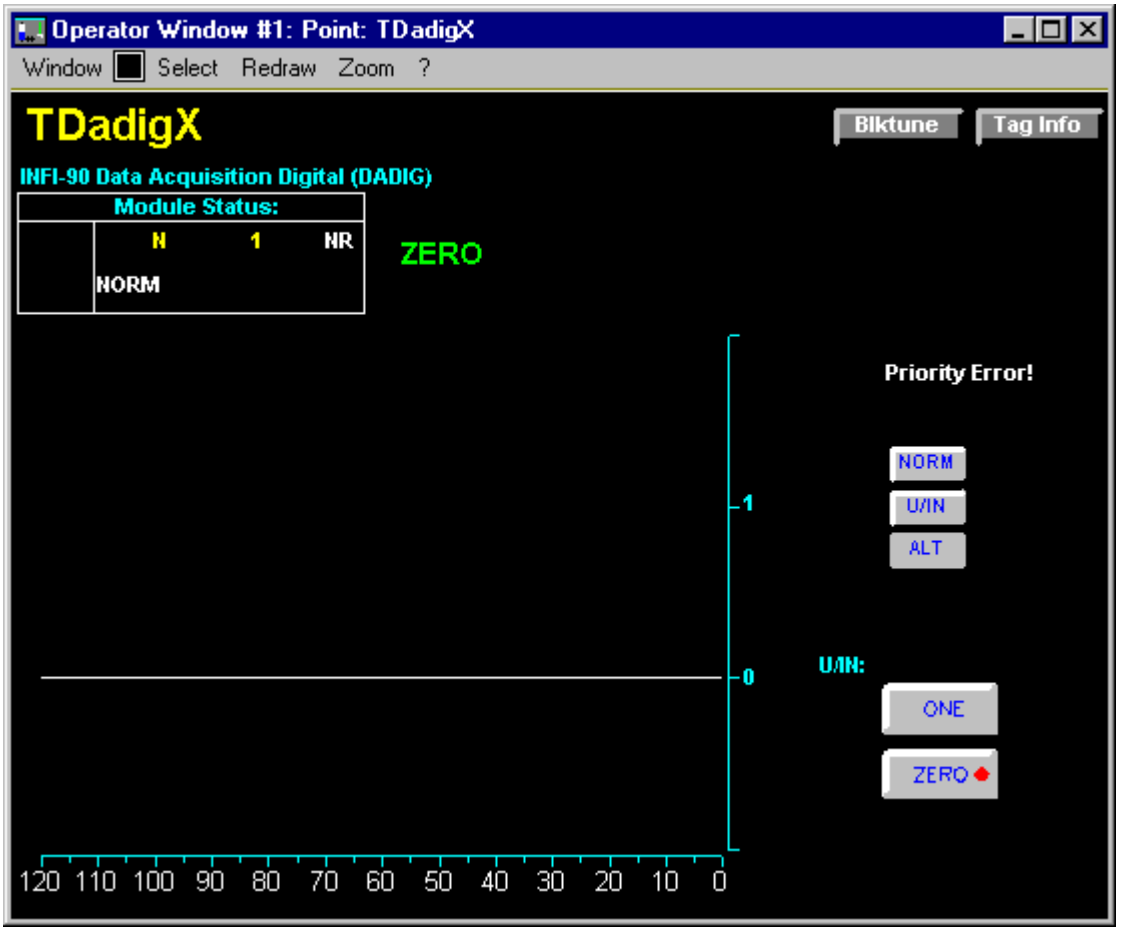


Figure 3-4. Point Display Example

HIERARCHICAL DISPLAY CONFIGURATION

Configuration of summary and group displays is accomplished via the *Configure* option on the menu bar of the summary display. Click on this option to produce the *Configure* pull down menu. The *Configure* pull down menu provides access to summary and group configuration.

Summary Configuration

Click on *Summary* in the *Configure* menu. The *Configure Summary* (Figure 3-5) window will be displayed.

This window contains an entry box labelled *Description* and a *Delete* button, in addition to *OK* and *Cancel* buttons. Click on the *Description* entry box to enter a text string of up to 48 characters; this text will be used as the legend which appears next to the index number of the summary at the top of the summary display.

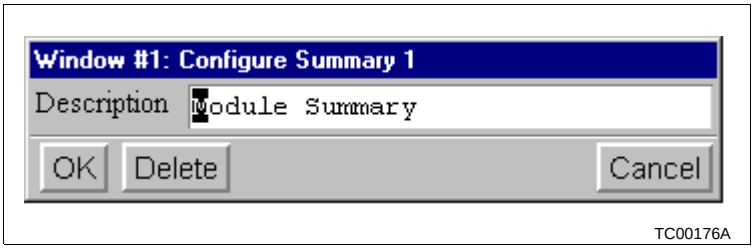


Figure 3-5. Configure Summary Window

Click on *OK* to complete the operation, or click on *Cancel* to exit without making changes.

To remove the current summary and all groups configured on it, click on *Delete*. A confirmation box will be displayed. Click on *Yes* to delete the summary and associated groups, or click on *No* to return to the *Configure Summary* window.

Group Configuration

To configure a group, click on *Groups* in the *Configure* window (Figure 3-6). A message appears below the menu bar, stating:

Select group to configure...

This message will remain, and the operator window will remain in group configuration mode until the user clicks on *End Config* in the *Configure* menu.

Once group configuration mode has been enabled, click on the pushbutton containing the number of the group to be configured. The *Configure Group* window (Figure 3-6) will be displayed.

The top half of the *Configure Group* window contains eight entry boxes in which to enter the tagnames of up to eight INFI 90 OPEN tagnames which are to be included in the group.

The text entry cursor automatically appears in the entry box for the first tagname. Enter a tagname; if a tagname already exists, click on a different entry box. Previously entered tagnames can be deleted using **Delete** on the keyboard; **Backspace** can be used to delete text to the left of the cursor. Note that tagnames which do not exist in any server on the network can be entered without error.

An entry box labelled *Description* is located below the *Tagname* entry area. Click on the *Description* entry box to enter a text string of up to 48 characters. This string will be used as both the group legend (at the top of the group display) and as the label of the group's block on the summary display.

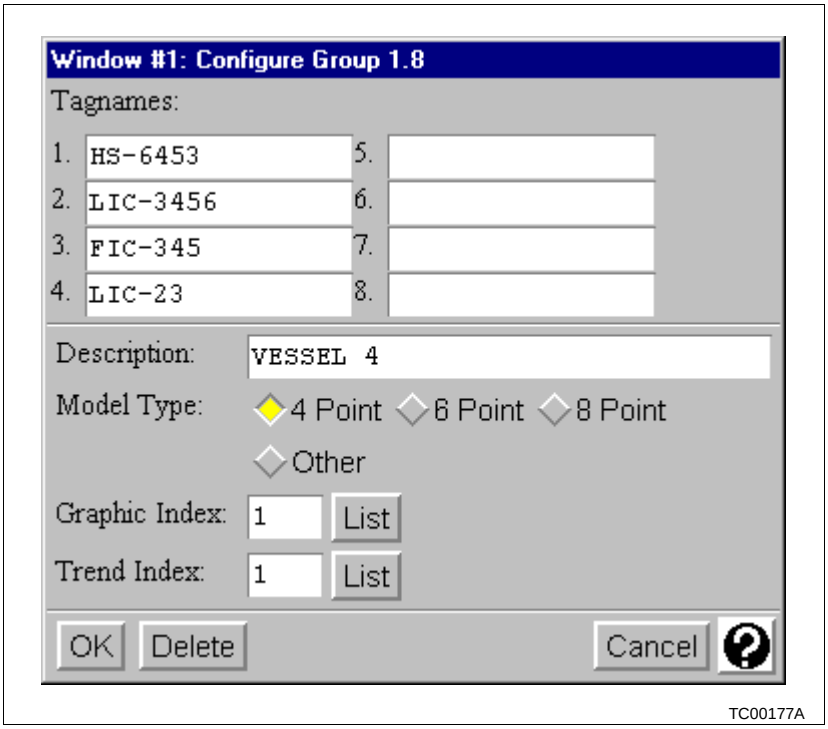


Figure 3-6. Configure Group Window

A group of radio buttons, labelled *Model Type*, is located below the *Description* entry box. These buttons are labelled as follows:

- Four point.
- Six point.
- Eight point.
- Other.

Click on one of the first three selections to choose a standard four, six or eight point group display format. The space allowed for each point on the group display varies, depending on the maximum number of points in the format. For example, a group display formatted for four points (Figure 3-3) provides twice as much room per point on the display than a group display formatted for eight points.

It is also possible to assign a graphic model to be used as the template for the group display. To do so, click on the radio button labelled *Other*; an entry box appears to the left of this selection. Click on the entry box, then enter the name of the model to be used as the template for the group display.

Two entry boxes, one for a *Graphic Index* number and one for a *Trend Index* number, are located below the *Model Type* selection area. The configuration procedure is identical for both parameters. Entering an index number in the entry box enables the corresponding button (*Graphic* or *Trend*) on the

Group Information window to vector directly to a specific graphic or trend display. (The *Group Information* window provides the operator with access to group and point displays and can be called up by clicking on the group pushbutton on the summary display. This window is described in the ***Personal Work Center 90 Operation*** instruction). To view a list of available displays, click on *List* located to the right of the entry box. A pop-up window containing a list of all available displays will appear (Figure 3-7).

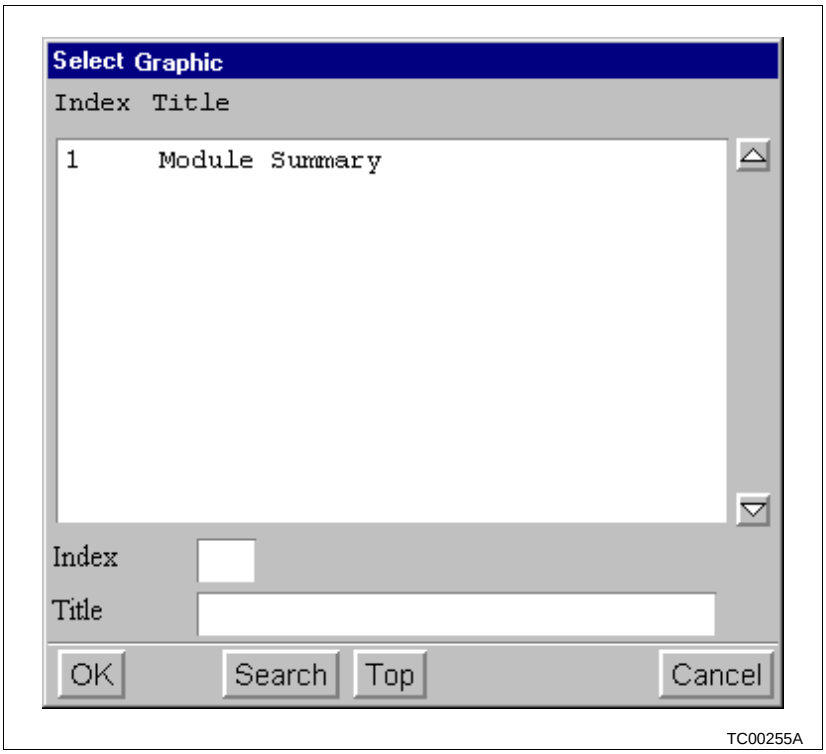


Figure 3-7. Graphic List and Trend List Windows

A display can be selected from the *List* window in one of two ways:

1. Click on the desired display in the *Graphic List Window* (a double-click on the item is equivalent to both selecting the item and clicking *OK*),

- ***or*** -

2. Enter the index number of the desired display in the Index entry box.

If part of a display title is known, enter that portion of the title in the *Title Entry* box, then click on *Search*. The list in the *List* box will shift so that the first title encountered which contains that string will appear at the top of the list box, highlighted. Click on *Top* to move to the top of the graphic list.

Once a display has been selected, click on *OK* to complete the operation and enter the display index number in the appropriate *Index* entry box on the *Configure Group* window. Click on *Cancel* to exit either the *Select Graphic* or *Select Trend* window without making a selection.

End Group Configuration

Once group configuration mode has been entered (refer to ***Group Configuration*** in this section), the operator window will remain in that mode and the message

Select group to configure...

will remain below the menu bar, allowing the configuration of one group after another from that summary display, until you click on *End Configuration* in the *Configure* menu.

Point Displays

Point displays are the most specialized hierarchical displays. Each point display provides detailed information about a single tag. (An example is provided in Figure 3-4). Pre-configured point display templates for the various tag types are provided on the PWC90 console. The type of the tag automatically determines which template will be used to display information about that point.

Information on the display and function of point displays is provided in the ***Personal Work Center 90 Operation*** instruction.

SUMMARY - HIERARCHICAL DISPLAYS

Hierarchical Displays

Hierarchical displays are preprogrammed displays provided by the PWC90 console, and organized into a hierarchy. The following displays are considered hierarchical displays:

- Summary displays (128 maximum).
- Group displays (3072 maximum; 24 per summary display).
- Point displays (four, six or eight per group display).

End Configuration

The *End Config* item is available from the *Configure* menu, which appears on the operator window menu bar when a summary is displayed in the operator window. Click on this item to end group configuration.

Group Display

A group display provides an overview of four, six or eight points. The top line displays a group legend of up to 48 characters. The display is divided into four, six or eight blocks; the information within each block relates to one point. There are 3072 group displays available.

List Button

Two *List* buttons are found on the *Configure Group* window, one to aid in selection of a *Graphic Index* number, and one to aid in selection of a *Trend Index* number. Click on one of the *List* buttons to display a window containing a list of all configured displays for the associated type (*Trends* or *Graphics*). Select a display to be associated with the group.

Entering an index number in the entry box enables the corresponding button (*Graphic* or *Trend*) on the *Group Info* window to vector directly to a specific graphic or trend display. (The *Group Info* window provides the operator with access to group and point displays and can be called up by clicking the group pushbutton on the summary display. This window is described in the ***Personal Work Center 90 Operation*** instruction).

Point Display

Point displays are the most specialized hierarchical displays. Each point display provides detailed information about a single tag. (An example is provided in Figure 3-4). Pre-configured point display templates for the various tag types are provided on the PWC90 console. The type of the tag automatically determines which template will be used to display information about that point. Therefore there is no need to configure point displays.

Search Button

If only a part of the display title is known, enter that portion of the title in the *Title* entry box, then click on *Search*. The list in the *List* box will shift so that the first title encountered which contains that string will appear at the top of the list box, highlighted. Click on *Top* to move to the top of the graphic list.

Summary Display

A summary display provides an overview of 24 groups and is arranged in six rows of four blocks, each representing a group. At the top of the screen a legend for the 24 groups is displayed. The legend can consist of up to 48 characters.

A pushbutton is located in each block on the summary display; the pushbutton contains the number of the group represented by the block (1 through 24). The color of the pushbutton provides information regarding the status of the group (Table 3-1).

Click on this pushbutton to display the *Group Info* window, which provides access to the group display or to individual point displays. Refer to the ***Personal Work Center 90 Operation*** instruction for details on use of the *Group Info* window.

Config

The *Config* option appears on the operator window menu bar when a summary is displayed in the window. ***HIERARCHICAL DISPLAY CONFIGURATION*** in this section provides details.

Top Button

Click on Top to move to the top of the graphic list or trend list in the appropriate list window.

SECTION 4 - GRAPHICS CONFIGURATION

INTRODUCTION

Up to 1024 network wide graphic displays can be configured using the PWC90 console. Because they are network wide, a graphic display can be configured on any PWC90 console; a copy of the file containing the graphic is then distributed to all PWC90 consoles on the network whenever a graphic is saved or installed. The graphic can then be viewed on any PWC90 console on the network.

Graphics configuration on the PWC90 console is performed via the graphics editor, an object oriented graphic modeling system based on the Sherrill-Lubinsky GMS (SL-GMS) package.

The graphics editor is used for interactive creation of a static background, dynamic display points, control points and display vectoring.

Graphics are configured in a two step process: first, static objects such as circles, lines, rectangles and text strings are placed on the display, creating a model. The objects used to create the model are sometimes referred to as the parts of the model. Various dynamic attributes, such as visibility, fill color, and edge color, can then be assigned to any object (or combination of objects) on the graphic. These dynamic attributes can be linked to atoms configured in the INFI 90 OPEN global database. A change in the status of an atom can then trigger a visible change in the graphical object.

A model can be saved as either a graphic or a submodel; submodels can be used and reused as objects within other models.

Both the placement of graphic objects and assignment of dynamic attributes are accomplished using the same graphic configuration display.

ACCESSING GRAPHIC CONFIGURATION

Graphic configuration is accessed by using the pull-down menus available from the system window *Menu* button. Position the pointer on the *Menu* button, and click the left mouse button. From the resulting pull-down menu, click on *Configure*, then click on *Graphic* in the resulting cascading menu. The graphics editor, which is composed of the *Graphic Configuration, Mode* and *Attributes* windows (Figure 4-1) will appear.

Only one graphics editor session can be opened per PWC90 console. If one has been opened, and the user again selects the

Configure Graphic menu item, a small pop-up window will appear, bearing the message:

Window Already Open.

If the *Graphic Configuration* display is not visible, it may be necessary to lower (or close) operator windows in order to locate the *Graphic Configuration* window.

CREATING MODELS

The graphics editor is composed of a set of three windows entitled *Graphic Configuration, Mode* and *Attributes* (Figure 4-1). These windows contain a number of discrete parts which, used together, enable the user to create objects, models, submodels and graphics on the PWC90 console, and provide tools for controlling the attributes of these items. The size, shape and location of the individual windows which make up the graphics editor can be changed to suit the user's needs.

In addition to an informative title bar, the *Graphic Configuration* window includes:

- Working view child window(s).
- Message area.
- Menu bar.

The *Mode* window contains:

- Object create control panel.
- Object change control panel.
- Status control panel.

The *Attributes* window contains:

- Color attribute control panel.
- Edge attribute control panel.
- Fill pattern attribute control panel.
- Text attribute control panel.

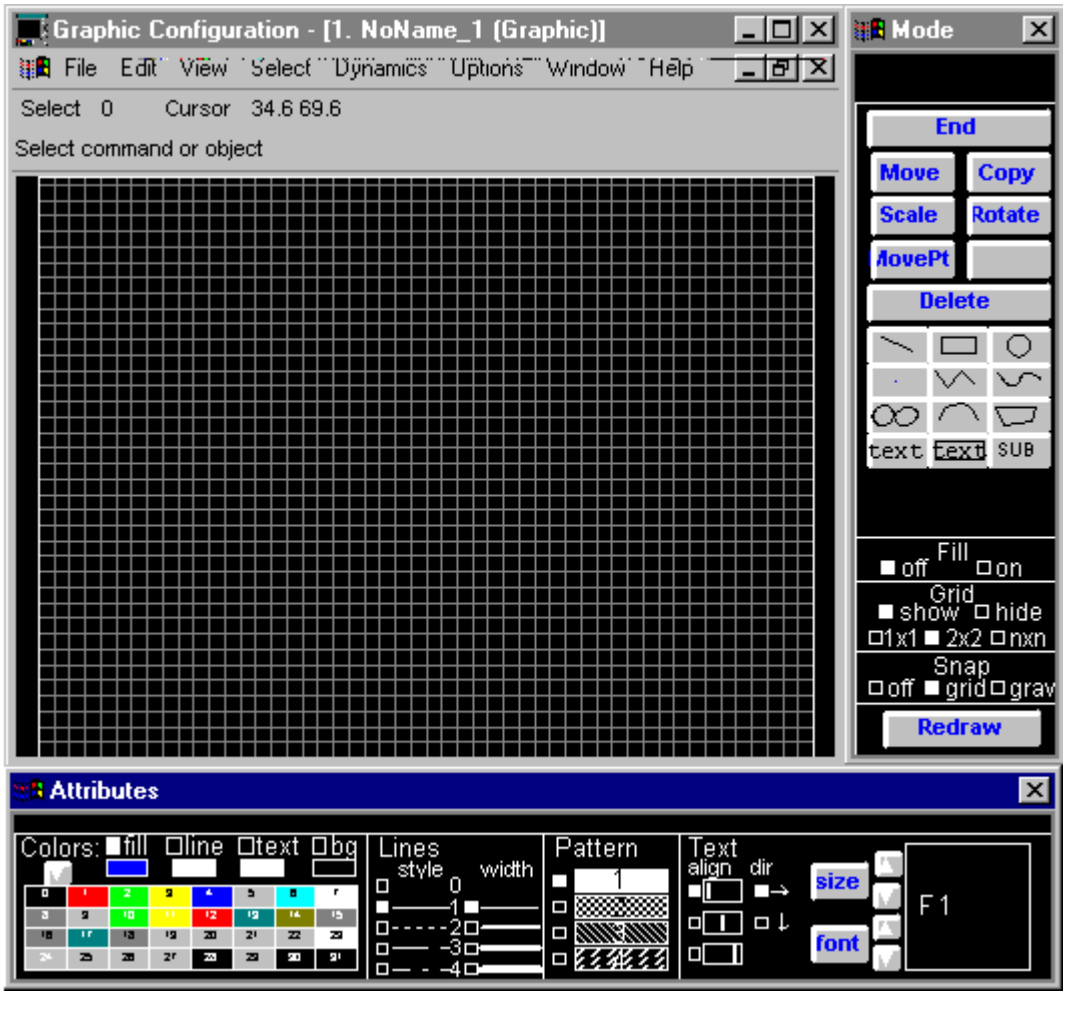


Figure 4-1. Graphics Editor Display, Working View Maximized

Working View Area

The working view area of the *Graphic Configuration* window is the graphic drawing area, where objects are placed during configuration. When the *Graphic Configuration* window is first opened, a single working view occupies the entire lower portion of the window. Click on *Child Window Maximize/restore* located at the right end of the menu bar on the *Graphic Configuration* window to convert the default working view into a child window (once this button has been clicked on, it disappears from the menu bar).

The resulting child window contains a discrete working view graphic drawing area with its own border and title bar; the title bar contains *Maximize, Iconify* and window menu buttons. Additional child windows can be opened using the *Window* item on the menu bar of the parent *Graphic Configuration* win-

dow (described in ***WINDOW PULL-DOWN MENU*** in this section). This allows configuration of several graphics simultaneously, and objects can be copied from one working view to another. The title bar of each child window contains the number of the child window, the name of the graphic which is being configured, and (in parentheses) the type of display (graphic or submodel) which is being configured. An asterisk (*) at the end of the string in the title bar indicates that unsaved changes have been made in the working view of that child window. The child windows can be resized, moved and iconified within the lower portion of the *Graphic Configuration* window. The child windows cannot be moved outside of the *Graphic Configuration* window.

NOTE: On the PWC90 console, the maximize and iconify buttons on the child window title bar(s) contain up and down arrows.

When using the *Graphic Configuration* window available from the PWC90 software, the window menu button on a child window can be used to *Move, Size, Minimize, Maximize, Close* or move to the *Next* child window. If only one child window exists within the *Graphic Configuration* window, it cannot be closed.

When a child window is selected or active, it has a blue border; the borders of inactive child windows are gray.

When a child window is maximized, it loses its individual border and title bar, and fills the lower portion of the *Graphic Configuration* window. The window menu button and child window Maximize/Restore button for the working view are again displayed on the menu bar of the *Graphic Configuration* window.

Each working view uses coordinates from (0,0) to (100,75) by default; these coordinates can be changed via the *Zoom* and *Pan* operations (refer to ***VIEW PULL-DOWN MENU*** in this section). Note that this is a relative coordinate system for placement of objects in the working view, rather than a set of absolute screen coordinates. When a model is first loaded into the working view, a *Zoom to fit* operation is automatically performed so that the entire graphic is displayed; once *Zoom* and *Pan* operations have been performed, the working view may not contain the entire model.

NOTE: The graphics editor maintains an aspect ratio four wide by three high, to maintain correct proportions of models during scaling operations.

Message Area

The message area is located above the working view area and below the menu bar of the *Graphic Configuration* window. Prompts to the user are displayed at the lower left portion of the message area; error messages are displayed in the lower right corner of the message area.

The *Select* and *Cursor* information fields are also located in the message area.

Select reflects the number of objects on the screen which the user has clicked on, and which will consequently be affected by any configuration action.

Cursor provides the coordinates of the pointer for the working view in which the cursor (also referred to as the pointer) is located.

Object Create Control Panel

The *Object Create Control Panel* is used to select the type of object to be created and added to the current model. Located in the *Mode* window, the *Object Create Control Panel* is composed of 12 buttons; each button represents a different type of object. When a button is clicked, the button is highlighted, and remains highlighted until the *End* button or another button is clicked, or until the user selects a different *Graphic Configuration* child window. The manner in which some buttons work can be modified by using the *Fill* option. Buttons representing closed objects such as circles, rectangles, polygons and closed splines can be created as filled objects. The function of these buttons is changed by first clicking the object button, then clicking on the *On* selection in the *Fill* portion of the *Status Control Panel*. The object button will now create filled objects.

Most of the objects provided in the *Object Create Control Panel* require that a series of points be chosen in the working view. These points define the object, such as a center point and a point on the radius for making circles. The prompts for the necessary actions are displayed in the message area (refer to ***Message Area*** in this section). The cursor becomes a plus sign (+) when the graphics editor expects a point to be selected.

The creation of any type of object is terminated by clicking on *End*, by clicking on another button in either the *Object Create Control Panel* or *Object Change Control Panel*, or by selecting a different *Graphic Configuration* child window.

Whenever the pointer is in the *Graphic Configuration* window, pressing the right mouse button is equivalent to clicking on *End*.

Double click on *Menu* in the *Mode* window (at the left end of the title bar) to close it; a button labelled *M* will then be displayed in the message area of the graphics editor display. Clicking on the *M* button will restore the *Mode* window to view. (The *Attributes* window can be closed and restored in the same manner. When closed, the *Attributes* window is represented by a button labelled with an *A* in the message area of the graphics editor display).

The following sections provide more detail about creating the different types of objects.

CREATING OBJECTS - STRAIGHT LINES

Straight lines are created by clicking on the *Line* button. Straight lines have only two points. If a third point is picked (clicked on), it is the beginning point of the next line.

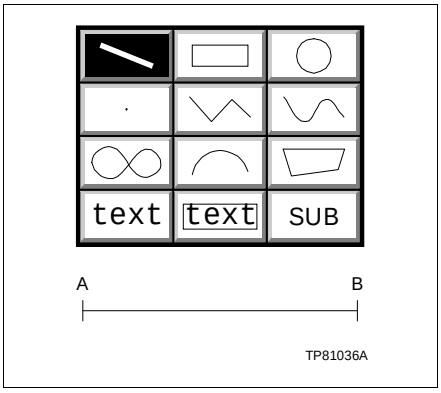


Figure 4-2. Drawing Lines

Example:

1. Click on *Line*.

2. Pick the first point, which will be the beginning of the line (A).

3. Pick the second point for completion of the line (B).
(The line appears on the screen).

4. Repeat steps 2 and 3 to create more lines.

5. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - RECTANGLES

Rectangles are created by clicking on the *Rectangle* button and choosing two opposing corners for each rectangle.

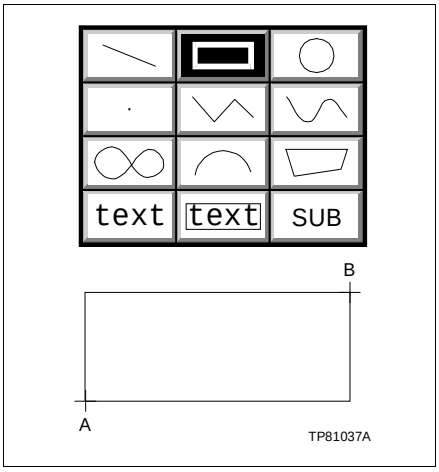


Figure 4-3. Drawing Rectangles

Example:

1. Click on *Rectangle*.

2. Pick the first corner of the rectangle (A).

3. Pick the second (opposite) corner of the rectangle (B). (The rectangle appears on the screen).

4. Repeat steps 2 and 3 to create more rectangles.

5. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - CIRCLES

Circles are created by clicking on the *Circle* button. The location and size of the circle are indicated by picking the center and a point on the edge.

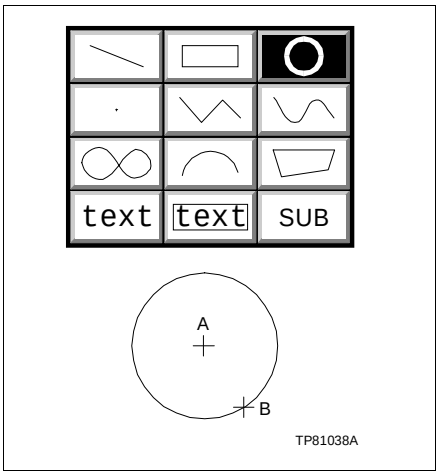


Figure 4-4. Drawing Circles

Example:

1. Click on *Circle*.

2. Pick the center of the circle (A).

3. Pick point on the edge (B).
(A circle appears on the screen).

4. Repeat steps 2 and 3 to create more circles.

5. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - MARKERS

Markers are created by clicking on the *Marker* button. Markers are small characters used to indicate positions of single points. Markers are used while constructing objects to help in placement. They are created by picking (clicking on) points in the working view.

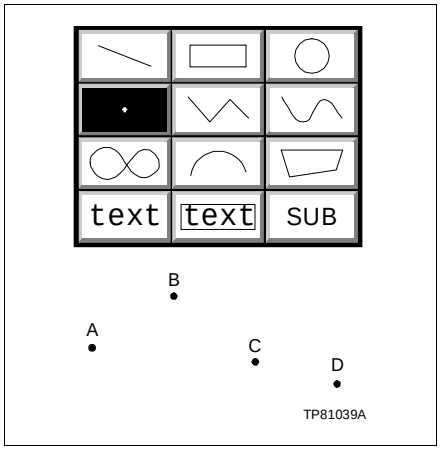


Figure 4-5. Creating Object markers

Example:

1. Click on *Marker*.

2. Pick the point for the marker (A).
(A marker appears on the screen).

3. Repeat step 2 to create more markers (b,c,d).

4. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - POLYLINES

Polylines add straight lines to a model. Polylines differ from Lines in that Lines have only two points, while polylines can

have more than two points connected by straight lines. Polylines are created by clicking on polyline, then choosing a starting point, then as many additional points as needed. A straight line is displayed between each pair of points chosen. Click on *End* to indicate that the last point in the polyline has been chosen.

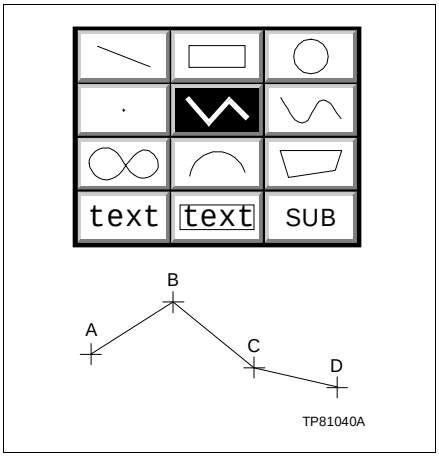


Figure 4-6. Drawing Polylines

Example:

1. Click on *Polyline*.
2. Pick the first point (A).
3. Pick the second point (B).
4. Pick the third point (C).
5. Pick the fourth point (D), (and so on).
6. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - SPLINES

Splines are created by clicking on *Spline*. The graphics editor uses the points picked to make the smoothest appearing curved line (the spline) which passes through these points. Points are connected by a polyline as they are selected. When *End* is clicked, the polyline is redrawn as a spline.

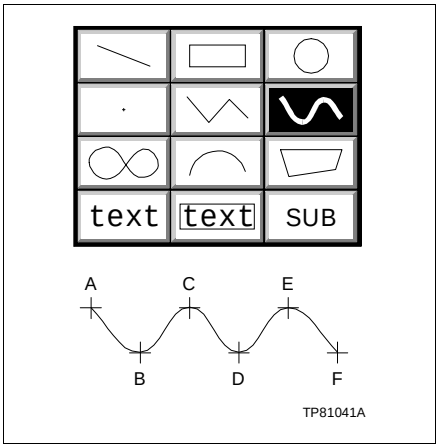


Figure 4-7. Drawing Splines

Example:

1. Click on *Spline*.

2. Pick the first point (A).

3. Pick the second point (B).

4. Pick the third point (C), and so on.

5. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

NOTE: The polyline connecting the chosen points is replaced by a spline.

CREATING OBJECTS - CLOSED SPLINES

Closed splines are created by clicking on *Closed Spline*. The graphics editor connects the points with a polyline until *End* (or a different *Object Create Control Panel* button) is clicked; this signals the editor to close (connect the end points) with a polyline and convert it to a closed spline.

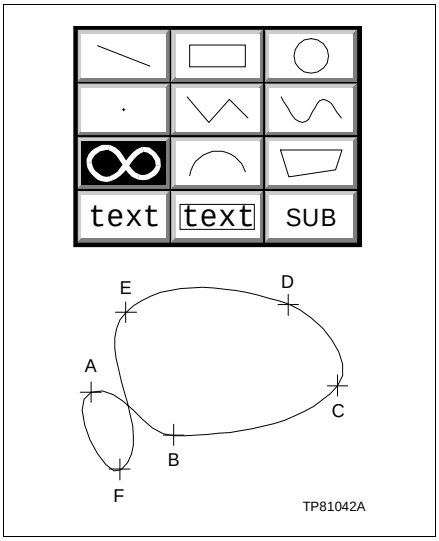


Figure 4-8. Drawing Closed Splines

Example:

1. Click on *Closed Spline*.
2. Pick the first point (A).
3. Pick the second point (B).
4. Pick the third point (C), and so on.
5. Pick the fourth point (D), and so on.
6. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

NOTE: The polyline connecting the chosen points is replaced by a closed spline.

CREATING OBJECTS - ARCS

An *Arc* is created by clicking on *Arc*. Arcs are portions of a circle and are constructed by picking the two endpoints of the Arc, then picking a point along the edge of the Arc.

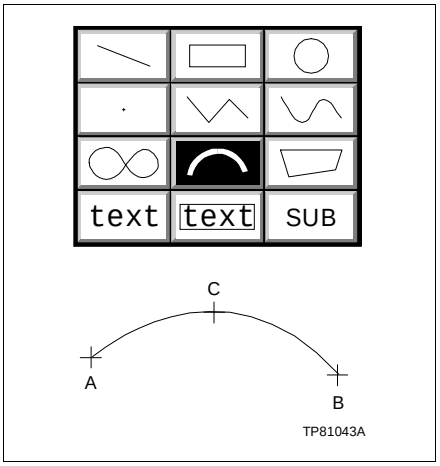


Figure 4-9. Drawing Arcs

Example:

1. Click on *Arc*.

2. Pick the first endpoint of the arc (A).

3. Pick the second endpoint of the arc (B).

4. Pick a point along the edge of the arc (C).
(An arc will appear on the screen).

5. Repeat steps 2, 3 and 4 to construct additional arcs.

6. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CREATING OBJECTS - POLYGONS

Polygons are created by clicking on *Polygon*. Polygons are many edged objects which can be added to a model. Polygons can appear as triangles, trapezoids, or any shape composed of straight edges. The graphics editor automatically connects the first point chosen with the last point when *End* or another *Object Create Control Panel* button is clicked.

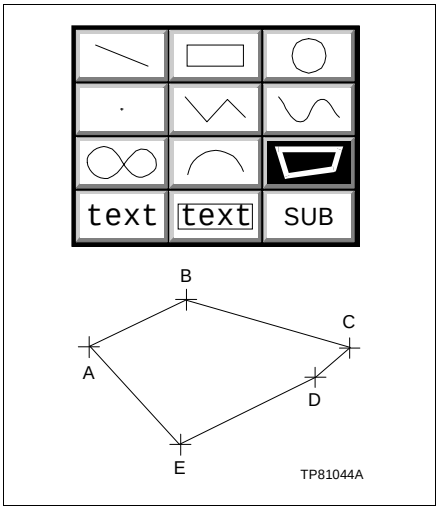


Figure 4-10. Drawing Polygons

Example:

1. Click on *Polygon*.

2. Pick the first point (A).

3. Pick the second point (B).

4. Pick the third point (C), and so on.

5. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

A polygon appears on the screen. The first and last point are automatically connected.

USING THE TEXT BUTTON

There are two *Object Create Control Panel* selections for creating text objects. Whether *Text* or *Filled Text Rectangles* are chosen, the attributes used for the text are the same.

Text is added by clicking *Text*. Text is any printable character entered from the keyboard. A text object is created by picking a point to position the text. Next, characters are entered from the keyboard.

After text has been entered, text attribute modifications (to alignment, font, and so on) can be made; these modifications affect the text most recently entered, until the next point for positioning text is chosen.

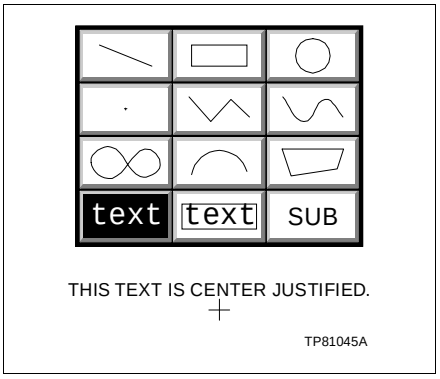


Figure 4-11. Inserting Text

Example:

1. Click on *Text*.

2. Pick the point to position the text.

3. Enter the characters (type them in via the keyboard, then press the *Enter* key).

4. Modify text attributes, if necessary (refer to ***Text Attribute Control Panel*** in this section).

5. Repeat steps 2 through 4 to enter additional text objects.

6. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

USING THE FILLED TEXT RECTANGLE BUTTON

Filled text rectangles are created by clicking on *Filled Text Rectangle*. Filled text rectangles are filled rectangles with text written on top of them. it is faster to erase and redraw filled text rectangles than it is to erase and redraw ordinary text. Changing text requires the old text be erased by being redrawn in the erase color, a relatively slow operation. Text in filled text rectangles is deleted by erasing the filled rectangle, a much faster operation. Filled text rectangles are used when creating Text objects to be changed by a process display.

Filled text rectangles are created by picking the two opposing corners for the rectangle and entering text followed by Return. To change text attributes, the *Text Attribute Control Panel* is used before the corners of the rectangle are picked or after text is entered.

> **NOTE:** Text is not prevented from appearing outside of a filled text rectangle. Text appearing outside the filled rectangle is not erased when text inside the rectangle is changed. It is important that a large enough filled text rectangle be used.

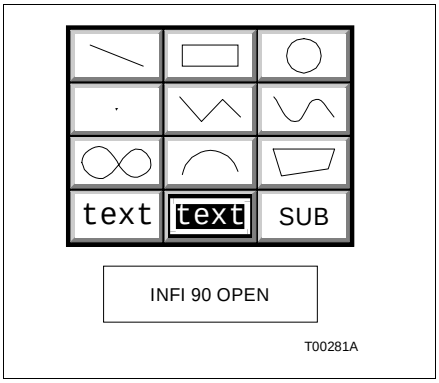


Figure 4-12. Filed Text Rectangle

Example:

1. Click on *Filled Text Rectangle*.

2. Modify text attributes, if necessary (refer to ***Text Attribute Control Panel*** in this section).

3. Pick the first corner for the rectangle.

4. Pick the second corner for the rectangle.
(A filled rectangle appears on the screen).

5. Enter characters.
(Text appears positioned within the rectangle).

6. Repeat steps 2 through 5 to configure additional filled text rectangles.

7. To terminate the procedure click on *End*, or click on a different child window, or click on another button on either the *Object Create Control Panel* or the *Object Change Control Panel*.

CHANGING TEXT OBJECT ATTRIBUTES

The *Text Attribute Control Panel*, located in the *Attributes* window, controls four characteristics (attributes) of the text being created or modified: size, font, direction, and alignment. Other text attributes which can be changed are color and content.

Text size is changed by clicking on *Size* in the *Text Attribute Control Panel*. The text will change to the size indicated in the box to the right of the *Size* button. Cycle through the available sizes by clicking on up or down arrow pushbuttons located immediately to the right of the *Size* button; do this until the selected text object(s) is displayed in the desired size.

Text font is changed in the same manner as text size: click on *Font* in the *Text Attribute Control Panel*; the text will change to the font indicated in the box to the right of the *Font* button. Cycle through the available fonts by clicking on up or down

pushbuttons located immediately to the right of the *Font* button. Do this until the selected text object(s) is displayed in the desired font.

Text direction (vertical or horizontal) is changed by clicking on the appropriate box located directly below the heading *dir* in the *Text Attribute Control Panel*. The arrows to the right of the filled box indicate in which direction the selected text is to be displayed.

Text alignment sets the offset of the text, relative to the point in the working view area chosen for positioning the text object. The text can be set as left (the default setting), centered, or right justified. The alignment is chosen by clicking on the appropriate box located directly below the word *align* in the *Text Attribute Control Panel*. The lines inside the rectangle next to the filled square indicates which alignment will be used to display the selected text.

Once entered, text content can be edited by selecting the *Text* option from the *Edit* menu available from the *Graphic Configuration* window menu bar. The user is prompted for the replacement text in the message area, in the same location normally used to enter text (refer to ***EDIT PULL-DOWN MENU*** in this section).

Text Color is controlled using the *Color Attribute Control Panel* located at the left end of the *Attributes* window. Text color is set by clicking on *Text* (or the square next to the word *Text*) located in the *Color Attribute Control Panel*. The square is filled after it has been selected. The color of the selected text can then be changed by clicking on the desired color in the *Color Attribute Control Panel*.

Double click on *Menu* in the *Attributes* window (at the left end of the title bar) to close it; a button labelled *A* will then be displayed in the message area of the graphics editor display. Clicking on *A* will restore the *Attributes* window to view. (The *Mode* window can be closed and restored in the same manner. When closed, the *Mode* window is represented by a button labelled with an *M* in the message area of the graphics editor display).

CREATING OBJECTS - SUBMODELS

Graphics can be composed of objects (such as rectangles, lines and circles), and submodels (such as valves and pumps). A submodel is a model (collection of objects) which has been saved using the model type submodel. (The model type can be defined using the *Options* selection on the *Graphic Configuration* window menu bar). Including a submodel in a graphic is referred to as making an instance of the submodel. There can be many instances of the same submodel in a graphic.

Submodels have their own attributes (such as fill color and edge color). These attributes can be dynamic. The submodels used in the PWC90 console are external submodels; when the submodel itself is altered, all models containing instances of that submodel will reflect the change. Submodels cannot be altered directly from the attribute control panels on the *Graphic Configuration* window; instead, both static and dynamic attributes of submodels are altered using selections from the *Dynamics* menu, which is accessed via the *Dynamics* option on the *Graphic Configuration* window menu bar. (refer to ***Configure Attributes - Dynamic Attributes for Objects*** in this section).

Submodels are accessed by clicking on *Sub* on the *Object Create Control Panel*. Selecting this button causes the *Submodel Selection* window to appear (Figure 4-13).

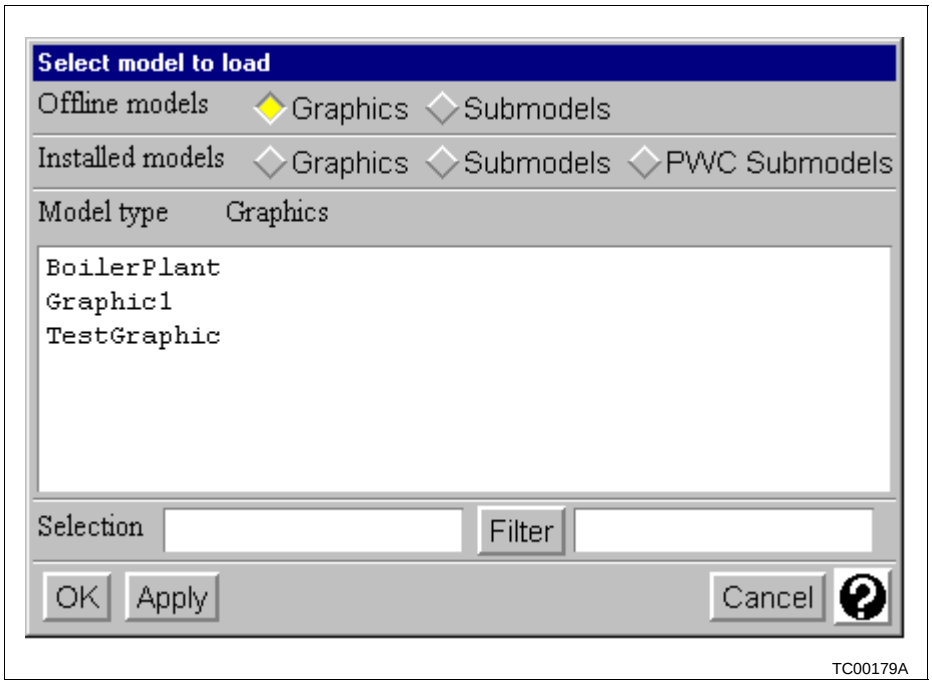


Figure 4-13. Submodel Selection Window

The title of this window prompts the user to select submodel to instantiate. Select from the various model types; after a model type is selected, a list of submodels available for that model type will appear in the list box area of the window. Model types include the following:

Off-Line Submodels

Off-line Submodels include all user configured, uninstalled submodels available on the system. (Refer to ***INSTALLED AND OFF-LINE GRAPHIC AND SUBMODEL FILES*** in this section for a discussion of off-line vs. installed submodels).

Installed Submodels

Installed Submodels include all user configured submodels which have been installed on the system, via the *Install* option (available from the *File* option on the menu bar of the *Graphic Configuration* window). Only installed submodels can be used in installed graphics. (Refer to ***INSTALLED AND OFF-LINE GRAPHIC AND SUBMODEL FILES*** in this section for a discussion of off-line vs. installed submodels).

> **NOTE:** If a graphic is installed which contains submodels which have not been installed on the system those submodels will not appear, nor will there be any warning to indicate that they are not installed.

PWC Submodels

PWC Submodels are supplied by Bailey. Submodels with prefixes ***FP_DY, FP_CW*** and ***i90_***, contain preconfigured dynamic attributes. These submodels are used:

- To display dynamic data from the database, using various forms such as values and bargraphs.
- For the control of modules.
- For the assignment of display vectoring.

The assignment of database points to these dynamics is accomplished through the *Configure Submodel* menu.

PWC Symbols are standard static submodels preconfigured and supplied by Bailey. Symbols are objects which represent standard items such as vessels, tanks and storage bins. Symbols which correspond to those defined by the Instrument Society of America (ISA) have the prefix ***ISA_***. NonISA static symbols created by Bailey contain the prefix ***FP_ST***. For a complete list of PWC Symbols, refer to ***PWC SYMBOLS*** in this section. Dynamic attributes can be assigned to these submodels via the *Configure Dynamic Attributes* menu item.

Locating Submodels via the Filter Button

The *Filter* option allows the user to search for a particular string of alphanumeric characters occurring in file names, or for a particular file. The search operation is case sensitive; lower case letters cannot be substituted for upper case, and vice versa. Click directly on the *Filter* entry box. Type in the string or file name for which to search, then click on *Filter*. The list box will display the names of all files containing the string. For example, searching the PWC Symbols category for the string AGIT will yield ISA_AGIT and ISA_AGIT_MOTR in the list box.

Selecting a Submodel

To select a particular submodel, click on the model name in the list box to highlight it, then click on *OK* (or double click on the model name in the list). A model can also be selected by clicking on the *Selection* field, then typing in the model name via the keyboard.

Selecting Multiple Submodels Using the Apply Button

Multiple submodels can be selected and placed during a session, using the following actions.

1. Select the desired submodel by clicking once on the model name in the list.

2. Click on *Apply* (instead of *OK*). Keyboard focus automatically changes to the *Graphic Configuration* window.

3. Move the pointer to the working view area; extent indicators appear to aid in placement of the submodel.

4. Move the submodel extent indicators to the desired location.

5. Click on the desired location.

6. If multiple instances of the same submodel are desired on the graphic, repeat steps 4 and 5.

7. Repeat steps 1 through 6 to place additional submodels on the display.

8. Click on *Cancel* or on *End* to end the session.

Object Change Control Panel

The *Object Change Control Panel* is located in the *Mode* window, above the *Object Create Control Panel*.

The buttons on the *Object Change Control Panel* modify objects on the select list. Once a change option is initiated from the *Object Change Control Panel*, changes are made to objects on the select list until *End* is clicked. All changes except *Copy* can be reversed by selecting the *Undo* command from the *Edit* menu (refer to ***EDIT PULL-DOWN MENU*** in this section), prior to clicking on *End*.

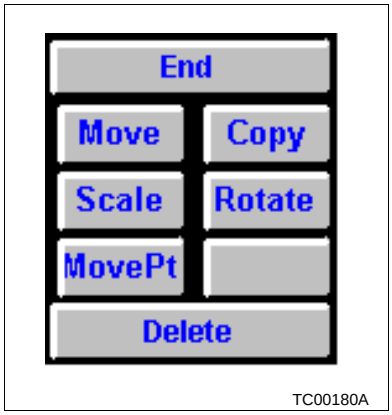


Figure 4-14. Object Change Control Panel

SELECT LIST

The select list is a list of objects in the current model which have been selected. An object is added to the select list by the following methods:

1. By clicking on the object. (The select list may or may not be cleared before the newly selected object is added to the select list, according to the current select mode option, which is set from the *Options* pull-down menu).

2. By using the *Select* pull-down menu.

3. By creating a new object. The select list is cleared, then the new object is added to the select list.

The number of objects on the select list is shown after the word *Select* in the *Message Area*. The select list is cleared by clicking *End*.

EXTENT AND REFERENCE POINTS

The extent of an object, also known as an extent rectangle, is the smallest rectangle, with a horizontal base, which completely encloses all parts of an object. The extent of an object is used to determine the location of the object's center; thus the center of any object is also the center of the object's extent.

Each object has one reference point. By default, the reference point of an object is defined as the center of the object's extent.

Three of the options available from the *Object Change Control Panel* make use of the reference point of an object. These options are known as object transformation options, and include:

- Rotate.

- Move.
- Scale.

The reference point of a selected object can be changed by selecting the *Change Ref* item from the select menu on the menu bar of the *Graphic Configuration* window, then clicking on a point other than the center of the object. If a single object is on the select list when the *Change Ref* function is invoked, this change in reference point remains in effect permanently for the instance of that submodel in the graphic. If more than one object has been selected, however, the change in reference point is in effect only while the objects are selected.

Use the *Reset Ref* option on the *Select* menu to set the reference point of each object on the select list back to its default, the center of the object's extent.

The reference point can be used in conjunction with the point snap mode discussed in ***GRID MODE*** in this section to aid in the placement and interconnection of objects and submodels on a graphic. Refer to ***OBJECT PLACEMENT EXAMPLE*** in this section for an example.

ROTATE

The *Rotate* button rotates selected objects. By default, an object rotates around the reference point, which by default is the center of the object's extent. When the *Rotate* button is clicked, a dashed reference line appears, with the left end of this line at the object's reference point. The reference line is an aid for choosing the angle of rotation. The angle to rotate the selected object(s) is chosen by clicking a point relative to the reference point and reference line.

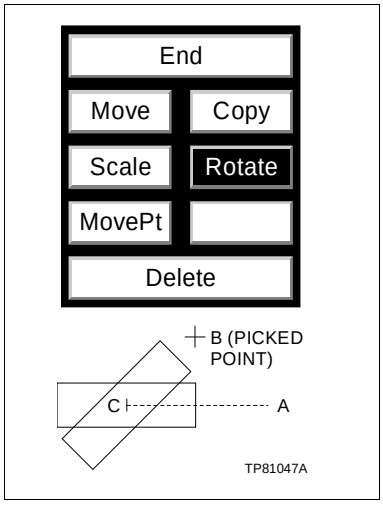


Figure 4-15. Rotate Objects

Example:

1. Select the object(s) to be rotated.

2. Click on *Rotate*. (Reference line appears (A)).

3. Click a point relative to the reference point and line (B). (Object(s) rotate about the reference point (C)).

4. Click *End* to stop picking rotation angles.

MOVE

The *Move* button is used to change the location of selected objects. When the *Move* button is clicked and the pointer is moved to the working view area, four extent indicators appear. These indicators represent the extent of an object on the screen; the pointer is positioned on the center or reference point of the extent. (Remember that the location of the reference point can be changed using the *Change Ref* option from the *Select* pull-down menu).

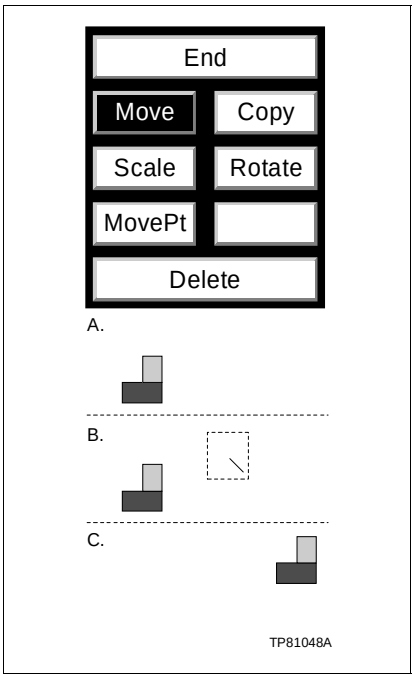


Figure 4-16. Move Objects

Place the pointer on the location which is to be the new center of the object and click on this location. The object will be moved to this spot. Thus, the object is positioned relative to the reference point.

Example:

1. Select the object(s) to be moved (A).

2. Click on *Move*. (Corner indicators appear (B)).

3. Click the point where the object(s) are to be displayed. (Objects are displayed at new position (C)).

4. Repeat step 3 or click on *End* to terminate the procedure.

The *Move* operation can also be initiated by positioning the pointer on the object to be moved, pressing and holding the left mouse button, then moving the mouse in order to drag the object to a new location. *Move* mode will remain in effect and the object can be repositioned until the right mouse button is clicked or another object or action is selected from either the *Object Create* or *Object Change Control Panel*.

SCALE

The *Scale* button is used to scale objects. Scaling is the action of making objects larger or smaller. By default, scaling is equal; that is, objects are increased or decreased in height or width by the same proportion. The *Unequal* option (refer to ***OPTIONS PULL-DOWN MENU*** in this section) can be chosen from the *Options* pull-down menu available from the *Graphic Configuration* window menu bar. The *Unequal* option allows objects to be distorted as they are scaled. For example, if a circle is scaled unequally, an ellipse is created.

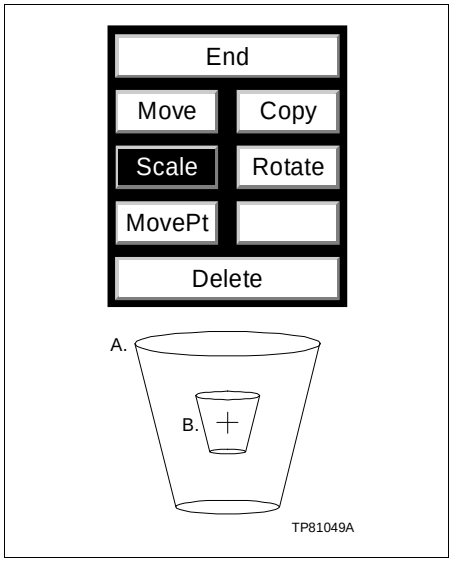


Figure 4-17. Scale object

Objects are scaled by clicking them to add them to the select list, then clicking on *Scale*. The objects are surrounded by a solid border. The scaling factor is chosen by clicking inside the border to make the objects smaller, or clicking outside the border to make the objects larger.

An object is scaled about its reference point, which by default is the center of the object's extent. The reference point can be changed by using the *Change Ref* option, available from the *Select* pull-down menu.

Example:

1. Select the object(s) to be scaled.

2. Click on *Scale*. (Solid border appears (A)).

3. Click inside the object's extent to make the object(s) smaller, or outside the object's extent to make them larger. (Objects appear larger or smaller (B)).

4. Repeat step 3, or click on *End* to stop scaling.

MOVEPT

The *MovePT* button on the *Object Change Control Panel* changes one or more points defining a selected object. Only one object at a time can be selected. If several objects have been selected, only the first item is affected.

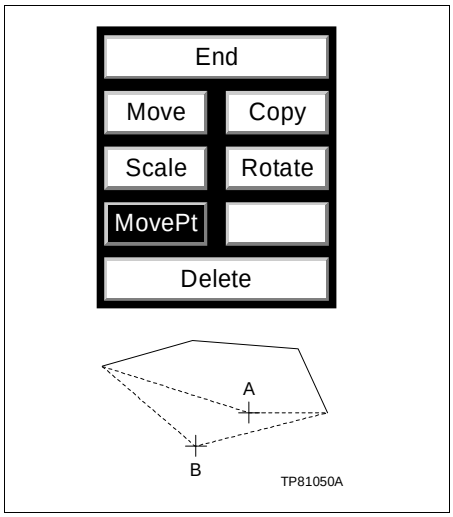


Figure 4-18. Move Point

When *MovePT* is clicked, the points defining the selected object are marked with ***x*** markers. The desired point is selected by clicking it. The other markers disappear. The selected point is moved by clicking in the working view, where the object is redrawn. New positions can be picked until *End* is clicked. The *MovePT* action can be cancelled if *End* has not been clicked. To do so, click on *Edit* on the menu bar, then click on *Undo Move Point* in the resulting pull-down menu. This method can be used to cancel any object change action.

Markers created via the *Marker* button and points on text objects cannot be moved.

Example:

1. Select the object(s) to be changed.

2. Click on *MovePT* (Markers appear at defined points).

3. Click the point to be moved (A). (Other markers disappear).

4. Click the new position where the point is to be located. (Object is redrawn (B)).

5. Repeat step 4 until satisfied with placement of point, then click on *End*.

COPY

Copies of an object are created by clicking the *Copy* button. Select the object(s) to be copied, then click on *Copy*. When the pointer is moved to the working view area, four extent indicators appear. The indicators represent the extent of the object(s) to be copied, with the pointer positioned to represent the center or reference point of the extent. Place the pointer on the location which is to be the new center of the object and click on this location. The object will be copied to this spot.

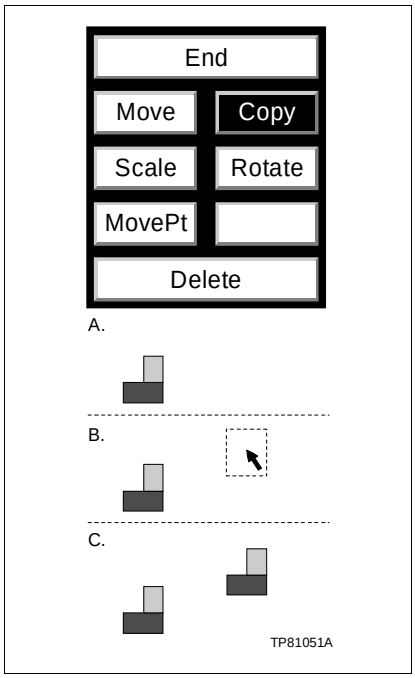


Figure 4-19. Copy object

Example:

1. Select the object(s) to be copied (A).

2. Click on *Copy*.

3. Move the pointer back to the working view area.

(Four extent indicators appear around the pointer, which represents the reference point or center of the extent (B)).

4. Position the extent indicators at location desired and click on it. (Copy appears (C)).

5. Repeat step 4 to create additional copies of the object.

6. To terminate the procedure, click on *End*, or click on a different child window, or click on another button on either the *Object Create* or *Object Change Control Panel*.

NOTE: You cannot undo a Copy operation, you must delete the object, as described in ***DELETE*** in this section.

END

The *End* button is located at the top of the *Object Change Control Panel*. Click on this button to complete an action and/or clear the selection list. When using the graphics editor in the *Graphic Configuration* window, a click on the right mouse button corresponds to a click on *End*.

NOTE: When multiple child windows are open within the *Graphic Configuration* window, switching between child windows is the equivalent of clicking on *End*.

DELETE

The *Delete* button, located at the bottom of the *Object Change Control Panel*, is used to delete selected objects.

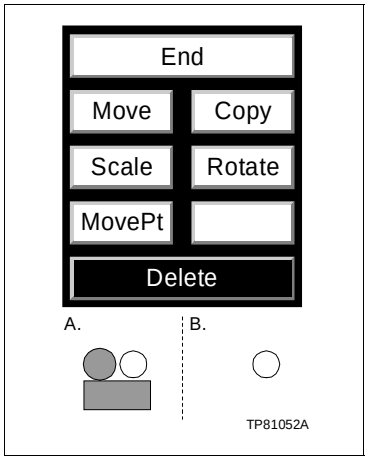


Figure 4-20. Delete object

Example:

1. Select the objects to be deleted (A).

2. Click on *Delete*.

3. (Selected objects disappear from the working view (B) and from the select list).

Status Control Panel

Options in the *Status Control Panel* turn the editor's grid *On* or *Off*, restrict points to grid intersections or points on other objects, and turn the *Fill* mode *On* or *Off*.

Located below the *Object Create Control Panel* in the *Mode* window, the *Status Control Panel* provides information on how modes in the graphics editor are set and is used to change these modes. The modes which can be monitored and changed via the *Status Control Panel* include fill, grid and snap. The *Redraw* button is also located in this control panel.

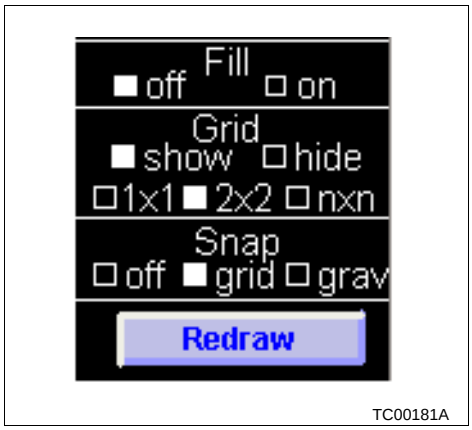


Figure 4-21. Status Control Panel

FILL MODE

The *Fill* mode area of the *Status Control Panel* is used to set the capability to fill selected objects to *On* or *Off*.

If *Fill* is set to *On*, objects in the select list will be filled with the fill color currently selected in the *Color Attribute Control Panel*, and the style currently selected in the *Fill Pattern Control Panel*. If an object had previously been filled and then unfilled, the old fill attributes (color and style) are retained and reused if the object is refilled.

Setting *Fill* to *On* can also affect the buttons in the *Object Create Control Panel*. If a button used for creating closed objects is clicked and *Fill* is set to *On*, the *Object Create Control Panel* button creates filled objects. Setting *Fill* to *Off* reverses this behavior.

Example:

1. Turn fill mode Off.
2. Select the object(s) to be filled.
3. Optionally change the current fill color and pattern.

4. Set fill mode to *On*. (Selected objects are displayed as filled).

GRID MODE

The grid used in the working view area(s) of the *Graphic Configuration* window aids in the construction of models. The grid is used much like a piece of graph paper, as a guide in designing objects.

The *Grid* section of the *Status Control Panel* provides two pair of options which control the grid. The *Show* and *Hide* options are used to control the visibility of the grid. Click on *Show* or the square indicator box next to it to make the grid visible; click on *Hide* or its associated indicator box to make the grid invisible.

The choices 2x2 and 1x1 are used to select the grid size. The numbers represent working view units; the default working view is 100 units wide by 75 units high, regardless of the size of the window or the display screen. The 1x1 option will cause the grid to reflect this and divide the working view into 100 units by 75 units. The 2x2 option provides a grid with a line drawn every two units. To make the selection, click on either the description or the associated indicator box.

The choice nxn allows the user to specify the size of the units which compose the grid. When this option is selected, the message area will prompt: *Enter grid size (now ZxZ)*, where Z is the size currently defining the grid units, i.e., 1x1. Enter a value from 0.2 to 10 in response to this prompt. The grid size will be adjusted accordingly. If a value outside these limits is entered, the grid will be resized using the closest available size, i.e., if 500 is entered, the 10x10 grid size will be used.

Although the grid is useful by itself, it is much more powerful when Snap mode (refer to ***POINT SNAP MODE*** in this section) is set to the grid selection.

POINT SNAP MODE

Point snap mode is a valuable tool which aids in the selecting of points and aligning objects. Snapping helps pick the same or similar points while building objects. Point snap mode can have one of three possible values: *Off, Grid,* or *Grav* (short for gravity). By default, point snap mode is *Grid*.

When *Grid* is on, every point that is picked is adjusted to the nearest grid intersection, whether the grid is visible or not, and always works with the current grid size. This is useful when creating horizontal or vertical lines, arrays of buttons, or other symmetric displays.

Objects created from the *Object Create Control Panel* and submodels provided by Bailey contain predefined points. Pre-

defined points can be used to rotate, scale, and make connections to the object. For polylines, lines, and splines, the predefined points are those used when these objects were created. Circles have predefined points at the 12, three, six and nine o'clock positions. Rectangles have predefined points at all four corners, and at the midpoint of each side. Arcs have predefined points at the center and ends of the arc.

Switching *Grav* on causes the point selection to snap to the closest predefined point of an object; that is, clicking a point on or very near an object causes the nearest predefined point in the object to be picked. For example, if a point near the corner of an existing rectangle is picked while the user is creating a new object, the predefined point at the corner of the rectangle will be used in the new object. *Grav* helps pick points to connect objects together.

Point snap mode can be used in conjunction with the *Change Ref* option from the *Select* pull-down menu to aid in the placement and interconnection of objects and submodels on a graphic. An example is provided in ***OBJECT PLACEMENT EXAMPLE*** in this section.

REDRAW BUTTON

Click on *Redraw* to redraw the screen. This is especially useful after objects have been moved, rotated, scaled or deleted or otherwise altered while the *Grid* option is set to *Show*.

Text Attribute Control Panel

The *Text Attribute Control Panel*, located at the right end of the *Attributes* window, controls four characteristics (attributes) of the text being created or modified: size, font, direction and alignment. Additional text attributes which can be changed are color (changed via the *Color Attribute Control Panel*), and content. Refer to ***USING THE TEXT BUTTON*** in this section for a discussion of the entry of actual text, which controls content.

Text size is changed by clicking on *Size* in the *Text Attribute Control Panel*. The text will change to the size indicated in the box to the right of the *Size* button. Cycle through the available sizes by clicking on up or down arrow pushbuttons located immediately to the right of the *Size* button; do this until the selected text object(s) is displayed in the desired size.

Text font is changed in the same manner as text size: click on *Font* in the *Text Attribute Control Panel*. The font will change to the font indicated in the box to the right of the *Font* button. Cycle through the available fonts by clicking on up or down pushbuttons located immediately to the right of the *Font* button. Do this until the selected text object(s) is displayed in the desired font. Note that an example of the currently selected

text font appears in the box which occupies the right end of the *Text Control Attribute Panel*. The example is shown in the selected font in the format F*x*, where *x* is the number of the font. The font number is used when configuring dynamics, (refer to ***DYNAMICS*** in this section).

Text direction (vertical or horizontal) is changed by clicking on the appropriate box located directly below the heading *dir* in the *Text Attribute Control Panel*. The arrows to the right of the filled box indicate in which direction the selected text is to be displayed.

Text alignment sets the offset of the text, relative to the point in the working view area chosen for positioning the text object. The text can be set as left (the default setting), centered, or right justified. The alignment is chosen by clicking on the appropriate box located directly below the word *align* in the *Text Attribute Control Panel*. The lines inside the rectangle next to the filled square indicate which alignment will be used to display the selected text. Editing of the text is accomplished through the *Edit Text* option from the *Edit* pull-down menu in the *Graphic Configuration* window.

Fill Pattern Attribute Panel

The *Fill Pattern Attribute Control Panel* is used to define the fill pattern of selected objects. It is located directly to the left of the *Text Attribute Control Panel*, below the working view area of the *Graphic Configuration* window.

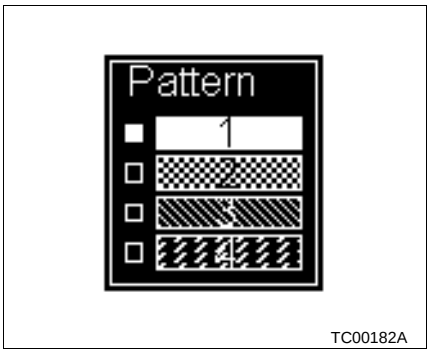


Figure 4-22. Fill Pattern Attribute Panel

The four fill patterns available are displayed in rectangles, each accompanied by an indicator box. The fill pattern of the selected objects is changed by clicking on either the rectangle containing the desired fill pattern, or on the indicator box next to it. The indicator box of the selected fill pattern is filled in, and the object will reflect the change. The fill pattern remains in effect for all subsequent objects, until a different selection is made.

Lines Attribute Panel

The *Lines Attribute Control Panel* is used to set the desired width and style of a line or the edge of an object.

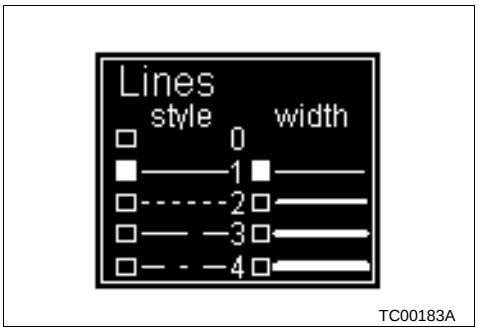


Figure 4-23. Lines Attribute Control Panel

This panel is located immediately to the left of the *Fill Pattern Attribute Control* Panel in the *Attributes* window. To select a new width or style for a line or edge of an object, click on the example or on the indicator box next to the example. The indicator box next to the selected style or width is filled in, and the select line(s) or object edge will reflect the change. Only one edge style and one edge width can be in effect at any one time. These attributes will remain in effect for all subsequent objects, until a different selection is made.

Color Attribute Panel

The *Color Attribute Control Panel* sets the color of selected objects and the background (bg) color and is located at the bottom left corner of the *Graphic Configuration* window, below the working view area.

Figure 4-24. Color Attribute Control Panel

Located at the top of this panel are rectangular selection boxes for the four different types of color attributes:

- Fill (i.e., circles, polygons).

- Line (lines and edge colors).
- Text (text objects).
- Background (bg).

These boxes are used to show which attributes will be affected by the selection of a color. Each color attribute type also has an associated radio button, which is filled in when the attribute type is selected. Only one of these color attribute types can be selected at a time.

Sixty-four colors are available, numbered zero through 63. These are contained on two pages of 32 colors each, which are displayed one page at a time below the color attribute type buttons. To move between the two pages of color buttons, click on the arrow button near the upper left corner of the *Color Attribute Control Panel*. Click on a color button; all objects on the select list will be changed to the selected color, as will the selection box for each color attribute type which is selected. The selected color remains in effect for all subsequent objects, until a different color selection is made.

Colors 32 through 63, which are on the second page of color buttons, appear in black by default. Definitions for these colors can be altered via the *Edit Colors* option on the *Edit* pull-down menu. (refer to ***CREATING OBJECTS - CLOSED SPLINES*** in Section 4 for details).

> **NOTE:** Editing these colors will impact the appearance of existing models.

Background Color Configuration

Black is the default background color for graphic displays. This may not be the ideal choice in all circumstances; for example, the user may wish to use the background color of a graphic to identify the plant area to which the graphic applies. To change the background color of the graphic being configured, select the toggle button labelled *bg* (located in the upper right corner of the *Color Attribute Control Panel*), then click on the button representing the desired color from the palette. The background of the graphic will change to that color.

The background color is assigned during configuration. If the user decides to change the background color, the graphic must be reconfigured and installed again; i.e., the background color cannot be changed while the graphic is on display in an operator window. Note that when configuring dynamic submodels, background color is configurable as the erase color, the color used to replace portions of the drawings when they are removed from the submodel. When a color is assigned to a submodel via the *bg* toggle button, the following message will be displayed in the right side of the message area:

erase color = n

where *n* is the number of the color selected from the color palette. When an instance of a submodel is made on a graphic, the instance assumes the background color of the graphic. When portions of the submodel are erased, the background color assigned during submodel configuration is displayed in place of the original color used. A good example of this is a dynamic submodel such as a tank when used in a larger graphic display. As the level decreases in the tank, the color representing the ingredients is replaced by the background color selected for the tank, instead of the background color of the graphic.

NOTE: This feature is for use only on graphic displays, and does not allow the user to change the background color of other display types (such as point, group, trend and summary displays, and so on). The background colors of these display types were chosen to provide the best contrast with other colors used in these displays.

Menu Bar

The menu bar of the *Graphic Configuration* window provides access to pull-down menus for use with the graphics editor. The pull-down menus are used to perform file, edit, and selection operations, dynamic attribute assignment, and selection of options.

INSTALLED AND OFF-LINE GRAPHIC AND SUBMODEL FILES

Two types of graphics and submodels are available on the PWC90 console: installed versions and off-line versions. Installed graphics and submodels are operational; i.e., they are in use on the system. Having off-line versions of these graphics and submodels permits the user to modify them without affecting the operational versions.

Once created, a graphic or submodel is saved as a file on the PWC90 console. If either the *Save* or *Save As* option is used to store a graphic the file containing the graphic will be stored as an off-line graphic in a location on the PWC90 console hard disk reserved for offline graphics.

Graphics are installed via the installation pop-up window in the graphics editor (refer to ***Installation Operations*** in this section). An installed graphic is assigned an index number from 1 to 1024 (inclusive) and the installation process moves the file containing the graphic from the off-line disk location (known as a directory) to a disk location which contains only installed graphics. More than one index number can be assigned to the same graphic. This index number is used when calling up a graphic for display in an operator window.

Only installed submodels can be used in installed graphics. An installed submodel is one which has been installed via the installation pop-up window in the graphics editor. The installation process moves the file containing the submodel to the disk location (or directory) which contains only installed submodels. When a graphic is installed, and contains references to an off-line submodel, the user will be prompted if those submodels should also be installed.

All models whether installed, off-line, graphic or submodel are network wide, and the effects of save, install, uninstall and delete operations are evident on all PWC90 consoles on the network.

Graphic and Submodel Type Descriptions

Off-Line Graphics Off-line graphics include all user configured, uninstalled graphics available on the system.

Off-Line Submodels Off-line submodels include all user configured, uninstalled submodels available on the system.

Installed Graphics Installed graphics include all user configured graphics which have been installed on the system, via the *Install* option (available from the *File* option on the menu bar of the *Graphic Configuration* window).

Installed Submodels Installed submodels include all user configured submodels which have been installed on the system, via the *Install* option (available from the *File* option on the menu bar of the *Graphic Configuration* window). Only installed submodels can be used in installed graphics.

PWC Submodels PWC submodels are submodels which have been supplied by Bailey. Submodels with prefixes ***FP_DY, FP_CW*** and ***i90_***, contain preconfigured dynamic attributes. These submodels are used:

- To display dynamic data from the tag database, using various forms such as values and bargraphs.
- For the control of INFI 90 OPEN tags.
- For the assignment of display vectoring.

The assignment of tag database points to these dynamics is accomplished through the *Configure Submodel* menu.

For a complete description of each Bailey Submodel, refer to ***BAILEY SUBMODELS*** in Appendix B.

PWC Symbols are standard static submodels preconfigured and supplied by Bailey. Symbols are objects which represent

standard items such as vessels, tanks and storage bins. Symbols which correspond to those defined by the Instrument Society of America (ISA) have the prefix ***ISA_***. NonISA static symbols created by Bailey contain the prefix ***FP_ST***. For a complete list of PWC Symbols, refer to ***PWC SYMBOLS*** in this section. Dynamic attributes can be assigned to these submodels via the *Configure Dynamic Attributes* menu item.

FILE PULL-DOWN MENU

Click on *File* in the menu bar of the *Graphic Configuration* window to view the *File* pull-down menu. The choices on this menu are used to perform various file operations.

Opening Files - New and Open

Click on *New* to create a new graphic or submodel.

Click on *Open* to open an existing graphic or submodel for further configuration. A pop-up window will appear (Figure 4-25), which prompts in the title bar: *Select Model To Load*.

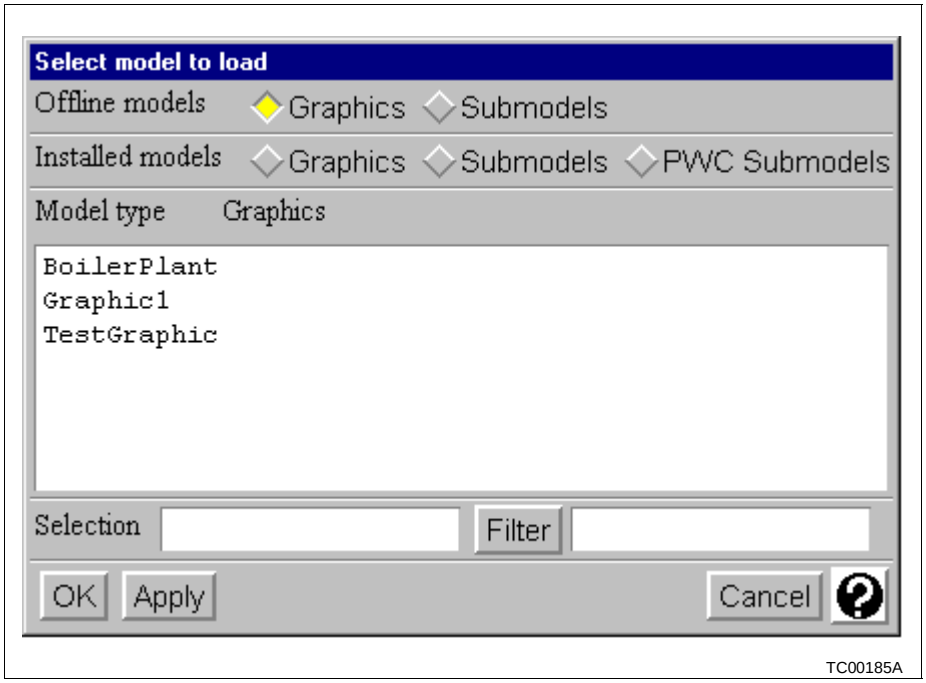


Figure 4-25. Select Model to Load Window

Like the *Submodel Selection* window (Figure 4-13) which it closely resembles, this window is used to specify the type of model to be opened, and provides a scrollable list box containing the names of all available models of the chosen model type. (***Graphic and Submodel Type Descriptions*** in this section provides a description of each model type). The letters at the beginning of names of models supplied by Bailey reflect the

type of the model. Table 4-1 shows a list of the name prefix and the coresponding model type.

Table 4-1. Bailey Supplied Submodel Types

Model Name Prefix	Model Type
FP_DY_	Dynamic Submodels
FP_ST_	Static Submodels
ISA_	ISA Standard Symbols
FP_IN_	Submodels for internal use
i90_	INFI90 OPEN Submodels

It is recommended that custom submodels be given names different from those supplied by Bailey. If models created by the user are given the same names as those supplied by Bailey, and then are installed, the custom models will be used by the PWC90 console, and the Bailey supplied versions will be ignored. This can create problems in the hierarchical displays, other graphics, and submodels which have already been created and which reference the Bailey supplied versions.

To select a model, double click on the desired filename in the list box, or click on the desired filename to select (highlight) it, then click on *OK*. A model can be selected by clicking on the *Selection* field, then typing in the model name via the keyboard. The *Filter* button can also be used to locate filenames containing a specific string, as described in ***Locating Submodels via the Filter Button*** in this section.

To load a model without closing the selection window, click on *Apply* (instead of *OK*). This allows the user to load one model after another without clicking on *File* and *Open* each time. This is useful when attempting to locate a particular model. It may be helpful to move the selection window to one side or corner, so that the working view area is unobstructed.

Saving Files - Save And Save As

When changes have been made to the current working graphic or submodel, and have not yet been saved, an asterisk (*) will be displayed after the filename of the graphic or submodel in the title bar of the window in which it resides. To save the current working graphic or submodel, click on *Save* in the *File* pull-down menu. To rename and save the current working graphic or submodel, click on *Save As* in the *File* pull-down menu.

The *Save* and *Save As* options will save the graphic or submodel currently in the selected working view as an offline version; the installed version, if any, will not be modified.

> **NOTE:** When the *Install* option is selected, the Save operation is performed automatically as part of the installation process.

Deleting Files - Delete

To delete an off-line graphic or a submodel which is either off-line or installed, click on the *Delete* option from the *File* pull-down menu. A pop-up window appears, with the title *Select Model File(s) To Delete*. Select the model type to be deleted, then double click on the file to be deleted from the resulting list box, or click on the file to be deleted to highlight (select) it, and click on *OK*. To guard against accidental deletions, a confirmation box appears; this box requires the user to click on *Yes* or *No* before proceeding with the delete operation. Click on *No* to exit without deleting files. If *Yes* is selected the operation will proceed, and a pop-up window with a *Cancel* button will appear. This pop-up window states:

Looking for references to submodel.

If *Cancel* is clicked, the search will be terminated. If *Cancel* is not clicked, the operation will continue. If no references to the installed submodel(s) are found, the window will close and the submodel(s) will be deleted. If references are found, the *Cancel* pop-up window will close and a *WARNING* pop-up window will appear with text similar to the following:

WARNING! <submodel name> is referenced by:
graphics: <reference_1>, <reference_2>, <reference_3>
submodels: <reference_1>, <reference_2>, <reference_3>

Delete <submodel name>?

Click on *No* to exit without deleting the submodel, or click on *Yes* to delete it.

> **NOTE:** Once the submodel is deleted, it will no longer be available for use by other submodels and graphics which reference it.

To delete more than one graphic or submodel during a *Delete* session, click on each item to be deleted so that it is highlighted, then click on *OK* in the *Select Model File(s) To Delete* Window. If multiple installed submodels have been selected for deletion, this sequence of looking for references and listing any references found will be repeated for each installed submodel. Another way to delete more than one graphic or submodel during a *Delete* session is to use the *Apply* button (instead of *OK* or double clicking) after selecting an item from the list box. After the model has been deleted, the user will be returned to the model list, and another model can be selected for deletion.

Applying Submodel Changes - Reload

The *Reload* option on the *File* pull-down menu is used to reload a currently displayed graphic or submodel in the selected working view, applying any changes which have been made to the submodels referenced by it.

For example, assume that two child windows are open within the *Graphics Configuration* window. In child window one there is a graphic called *Button_pad* which contains ten pushbuttons, all created using a submodel named toggle_pb. To change the face color of all buttons which reference this submodel, it is necessary only to change the configuration of the submodel. To do so, select child window two and load the submodel toggle_pb. Next, change the face color definition via *Submodel Configuration* (Refer to ***Configure Submodel*** in this section) and save the changes to the submodel. Note that the instances of the submodel in the graphic *Button_pad* in child window one still reflect the original button face color. Select child window one, then click on *Reload* in the *File* pull-down menu. The face color will be changed on all pushbuttons which reference the submodel toggle_pb.

> **NOTE:** A graphic or submodel must have a name (i.e., it must have been saved) before the reload operation can be performed successfully on it.

Installation Operations

Once a model has been configured, it can be installed for use, to be accessed via an operator window. Click on the *Install* item from the *File* pull-down menu; a pop-up window will appear, containing a list of available models. The current working model (if any) will be highlighted. To highlight a different model, click on it. To install the model, double click on the highlighted item, or click on *OK*. To install more than one graphic or submodel during an Install session, use the *Apply* button (instead of *OK* or double clicking) after selecting a model. After the model has been installed, the user will be returned to the model list, and another model can be selected for installation. The *Filter* button can also be used to locate model names containing a specific string, as described in ***Locating Submodels via the Filter Button*** in Section 4. A model can also be installed by clicking on the *selection* field, then typing in the model name via the keyboard.

A second pop-up window appears, entitled *Install Graphic*. This window contains three entry boxes, which are used to enter or change the model name, graphic index number and title of the model. (To quickly review the list of currently installed graphics, click on *List*). Click on the desired entry box, then enter information via the keyboard.

The *Install Current* menu item provides a shortcut to the *Install Graphic* pop-up window. The name of the model currently in the working view is automatically specified as the model name when this item is used.

If a previously assigned graphic index number is entered, a pop-up window appears to warn of the conflict, displaying the following message:

Another graphic is using this index. Replace it?

Click on *Yes* to replace the existing graphic, or click on *No* to return to the *Install Graphic* pop-up window and enter a different graphic index number.

The *Uninstall* option is used to remove a graphic from operational use. To do so, click on the *Uninstall* option in the *File* menu. A pop-up window appears, containing a scrollable list box which contains a list of all installed graphics. Highlight the desired graphic, or enter the graphic index number or graphic title, then double click or click *OK*. A pop-up window appears to request confirmation that a graphic is to be uninstalled; this window displays the message:

Uninstall Graphic n?

where *n* represents a number from one to 1024. Click on *Yes* to uninstall the graphic; click on *No* to let the graphic remain installed.

Listing Component Models - Cross Ref

Click on *Cross Ref* to display a cross reference list of all Bailey and user defined models (including symbols, submodels, etc.). which are used as components of a given model. Clicking on this item causes a pop-up window to appear. The title bar of the window prompts *Select Model To Cross Reference*. Click on the desired model type, then select the desired item from the resulting list displayed in the list box. When this item has been selected (highlighting it, then double clicking or clicking on *OK*), a new pop-up window is displayed. This window displays the model name, model type, and name (if any) under which it is installed. This information is followed by a list of references, which contains the name of each submodel used in the graphic. Click on *OK* to remove this pop-up window from the screen.

Exiting Graphic Configuration - Exit

To exit from the graphics editor and *Graphic Configuration* mode, click on the *Exit* option available from the *File* item on the *Graphic Configuration* menu bar. If no changes have been made to the current model, the display will be exited immedi-

ately. If changes have been made, a small pop-up window queries:

Save changes to <model name>?

Click on *Yes* to save changes, click on *No* to abandon changes. If multiple child windows are open in the *Graphic Configuration* window, a prompt will be displayed for each unsaved model before the utility is exited. The graphics editor is then removed from the screen.

To cancel the exit operation and remain in the graphics editor, click on *Cancel*.

EDIT PULL-DOWN MENU

Click *Edit* in the menu bar of the *Graphic Configuration* window to view the *Edit* pull-down menu.

The *Undo* menu item is used to *undo* the last action. For example, if an object is accidentally deleted from the display, click on *Edit*, then click on *Undo Delete*. The actual text of the *Undo* command in the menu can vary, depending on nature of the last action; i.e., it may actually read *Undo Delete, Undo Rotate, Undo Move Point*, and so on. The *Undo* option must be used before the *End* button is clicked.

Undo cannot be used to remove newly created objects; this can be accomplished by clicking *Delete* immediately after the object has been created, since an object is automatically placed on the select list as soon as it has been created.

The *Text* item on the *Edit* pull-down menu is used to modify the text in an object. To do so, first click on the object(s) to be edited, then click on *Text* from the *Edit* menu bar option. The text used in the text object is displayed for edit in the message area, above the working view. By default, text is added to or deleted from the end of the existing text. To make changes elsewhere in the text string, click the pointer on the location where the text is to be added, deleted or changed. (Refer to ***USING THE TEXT BUTTON*** in this section for details on entering the text). Text can then be modified at any location in the string. This is useful for correcting misspellings, or revising text in graphics or submodels which have been copied and require modification.

The *Name* item on the *Edit* pull-down menu is used to assign names to individual objects for purposes of identification. The object *Name* is also used by special dynamics; for details, refer to the ***SL-GMS Reference Manual***, which is available as an option from Bailey.

The *Colors* item on the *Edit* pull-down menu is used to view and modify the color definitions for colors with index numbers 32 through 63. Select one of these colors (refer to ***Color Attribute Panel*** in this section), then click on *Colors* to display the *Edit Colors* window (Figure 4-26).

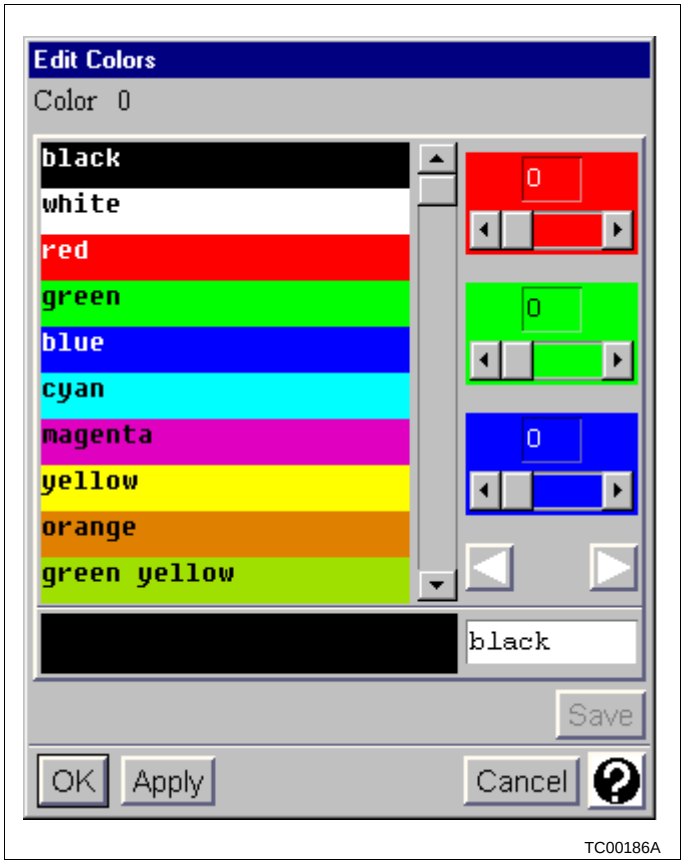


Figure 4-26. Edit Colors Window

The upper left corner of the *Edit Colors* window contains the index number of the color to be defined.

Each available color on the monitor is created using the three primary colors red, green and blue in varying intensities; a value from 0 to 255 is used to define the intensity of each primary color. Three color definition boxes in the primary colors red, green and blue appear on the right side of the main display portion of the *Edit Colors* window. Each color definition box contains a horizontal slider and above it an entry box. The entry box contains a number from zero to 255 (inclusive), which reflects the location of the slider and the current intensity value for that primary color. Thus, a color definition is composed of three values. For example, a color definition with a value of 0 (zero) for all three color intensities produces the color black (all sliders positioned at the extreme left); a value of 255 for all three color intensities yields white (all sliders posi-

tioned at the extreme right); a color definition of RED = 255, GREEN = 255, BLUE = 0 produces yellow, and so on.

A sample color box is located at the bottom left of the main display portion of the *Edit Colors* window. Initially this sample color box reflects the color of the selected object (if any), or the last color to be defined. The sample color box changes color as the color definitions are adjusted, providing immediate feedback regarding the result of the color mix. The name of the color (or an identifying string) will appear in the text entry box to the right of the sample color block. Alternatively, in the text entry box type the name of the color (to a maximum of ten characters) or the identifier of the color (if known), followed by Enter to change to that color. (An error message will result if the color name exceeds ten characters).

With the *Edit Colors* window displayed, select a color index number by clicking on a color button (from colors 32 through 63) in the *Color Attribute Panel*. The current color definition for that color button is displayed. To modify the color, use the mouse to drag the sliders in one or more of the three color definition boxes, or change the value in the color definition entry box(es). The result of each change is reflected in the color sample box. Objects previously drawn with the selected color index can be redrawn using the new color by clicking on the *OK* or *Apply* button, and then clicking on *Redraw*. Use of the *Apply* button allows the user to define colors for additional color indexes without leaving the *Edit Colors* window. Modifications to color indexes 32 through 63 can be saved permanently by clicking on *Save*.

Alternatively, predefined colors can be selected from the listbox which occupies the left side of the main display portion of the *Edit Colors* window. Use the scrollbar on the list box to view available colors and their names; click on the desired entry to select it.

> **NOTE:** Modifications to colors 32 through 63 will take effect only in those operator windows which are opened *after* the new color configuration has been saved.

The *Raise* item on the *Edit* pull-down menu is used to modify the order in which objects are drawn on the screen. Objects are normally drawn in the order in which the user created them. If two objects overlap, the object which was created more recently is drawn later, and therefore appears on top of (and obscuring) the older object.

Use the *Raise* item to put the currently selected object(s) at the end of the drawing order; they will then be drawn last, and will appear on top of all other objects which overlap.

> **NOTE:** The above discussion of the *Raise* item applies to the drawing order of *static* objects. However, when a model is displayed in an operator window, and the attributes of an object in the model are modified according to the dynamics defined for that object (i.e., the object's fill color is changed in response to some change in a tagname.atom), the object must be redrawn, which places it on top of other objects. For this reason, it is recommended that no objects be placed on top of dynamic objects. If it is necessary to do so, dynamics must be also configured for the objects on top, so that all objects involved are redrawn when a dynamic change occurs.

The *Lower* item on the *Edit* pull-down menu performs in a manner opposite to that of the *Raise* item. Use the *Lower* item to put the currently selected object(s) at the beginning of the drawing order; they will then be drawn first, and will appear beneath all other objects which overlap.

When an object in the working view is selected (clicked on), it blinks several times (rapidly) to indicated that it has been selected. This may cause a temporary change in the drawing order. The actual drawing order can be restored at any time by clicking on *Redraw*.

The *Redraw* option available from the *Edit* pull-down menu can be used to redraw the screen as necessary. This button corresponds to the *Redraw* button located at the bottom of the *Status Control Panel*.

Grouped Objects and Expressions in this section provides a details on the use of the *Group* and *Ungroup* items.

VIEW PULL-DOWN MENU

Click on *View* in the menu bar of the *Graphic Configuration* window to show the *View* pull-down menu.

The *Zoom* item is used to change:

- The portion of the graphic which is visible in the working view area of the *Graphic Configuration* window (the *Pan* operation).

- The amount of coordinate space which can be visible in the working view, known as the *view size* (the *Zoom* operation).

The view size always has an Aspect Ratio four (wide) by three (high), to maintain correct proportions of models during scaling operations. The default view size is 100 x 75 units; all graphics configured using early releases of PWC90 console software (Release Series 1 and earlier versions) were restricted

to this view size. The maximum view size is 400 units by 300 units; thus, the available area for a graphic is actually 16 times the size of the default view size. The user can maintain the same view size and pan to see additional areas of the coordinate space, or use the zoom feature to increase/decrease the view size in the working view (Table 4-2).

Table 4-2. View Size Information

View Size	Number Of Units	Lower Left Coordinate	Upper Right Coordinate
Default	100 x 75	0, 0	100, 75
Maximum	400 x 300	-150, -112.5	250, 187.5
Minimum	20 x 15	Can be located anywhere within the Maximum view size coordinates.	+

Coordinates for the minimum, maximum and default view sizes are shown in the table above.

After each zoom or pan operation, the lower left and upper right coordinates of the resulting view size are displayed in the right side of the message area. (These coordinates can also be displayed by clicking on *Show Dimensions* in the View menu).

To access the *Zoom* and *Pan* functions, click on *Zoom* in the *View* menu. The *Zoom* window (Figure 4-27) will be displayed.

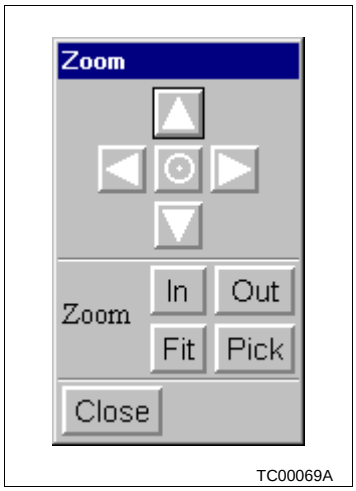


Figure 4-27. Zoom Window

The top portion of the *Zoom* window provides *Pan* functions, while the bottom part of the window is devoted to the *Zoom* operation.

NOTE: The use of Grid Mode (***GRID MODE*** in this section) is unaffected by Zoom and Pan operations.

Pan Operations - Up, Down, Left, Right, Center

The *Pan* operations (Up, Down, Left, Right, Center) are used to change the portion of the view shown in the working view portion of the graphics editor; they do not change the view size, or the model which is being edited.

Each of the *Pan* functions is represented on the *Zoom* window by a button. The face and location of the button indicates its function. The face of each of the directional operations (Up, Down, Left, Right) contains an arrow pointing in the appropriate direction, and each button is located in the appropriate position around the center button. The face of the center button contains a bull's eye.

Each click on one of the directional *Pan* buttons moves the visible portion of the view in the selected direction by 1/3 of the corresponding view dimension.

After the center button is clicked, the next click in the active child window will center the view on the point clicked.

Zoom Operations - In, Out, Fit, Pick

The *Zoom* operations (*In, Out, Fit, Pick*) change the size of the working view portion of the graphics editor, but do not change the model which is being edited.

Each time the *Zoom In* button is clicked, the view size is reduced by 62%. The center of the view remains unchanged. Objects in the model which are visible *appear* correspondingly larger, but are not actually modified; remember that this operation merely magnifies a portion of the graphic for the convenience of the configurer. If the view has been fully zoomed in, no further change occurs to the view size.

Each time the *Zoom Out* button is clicked, the view size is enlarged by 62%. The center of the view remains unchanged. Objects in the model which are visible *appear* correspondingly smaller, but are not actually modified; this operation simply allows the entire graphic to be displayed in the working view for the convenience of the configurer. If the view has been fully zoomed out, no further change occurs to the view size.

Click on *Zoom Fit* to adjust the view size by automatically zooming out or zooming in so that the entire model is visible and fills the working view. If there are no objects outside the default view size (100 units by 75 units), the view size will not change (i.e., the view will *not* zoom in order to make the objects present fill the working view). When a model is initially loaded into the working view, a *Zoom* Fit operation is automatically performed. (This is also done when an installed graphic is displayed in an operator window).

The dimensions of the default view size are enclosed in a rectangle composed of dotted lines. When zooming or panning outside the default view dimensions, this rectangle or portions thereof may be visible within the working view. This rectangle is for reference only; it is not part of the model and will not be visible when the graphic is displayed in an operator window.

The *Zoom Pick* button allows the user to select the diagonal corners (i.e., lower left and upper right corners) of a box representing the desired view size. The rectangle resulting from these selections is called the zoom box. If a zoom box smaller than the minimum view size is selected, the minimum view size is used, with the same center as the selected zoom box.

> **NOTE:** The contents of the working view after a *Zoom* Pick operation may not reflect exactly the contents of the zoom box used to make the selection. This is because the working view is restricted to an Aspect Ratio of 4 (wide) by 3 (high), while the zoom box is not. In order to reconcile the two, the 4 x 3 Aspect Ratio is applied to the largest dimension of the zoom box selected by the user.

Show Dimensions Menu Item

If the lower left and upper right coordinates of the current view size are not currently displayed in the right side of the message area, click on *Show Dimensions* in the *View* menu to display them.

SELECT PULL-DOWN MENU

Click on *Select* in the menu bar of the *Graphic Configuration* display to view the *Select* pull-down menu.

By Extent - This option selects all objects that are displayed completely inside a rectangle called the *extent*. The user is prompted to pick two points which form the opposing corners of the extent rectangle that must completely surround the objects to be selected. Any objects selected are added to the existing select list, if the multi select option is enabled.

By Name - This option selects the object specified by the user in response to the *Name* prompt which appears in the message area when this item is selected. (The *Name* item on the *Edit* pull-down menu is used to assign names to individual objects, as specified in ***EDIT PULL-DOWN MENU*** in this section).

All - This option selects all objects in the current model in the working area.

Show - This option flashes all objects that are currently selected (more obvious when item is filled).

Change Reference - This option is used to change the reference point of a selected object, and with Reset Ref, is described in ***EXTENT AND REFERENCE POINTS*** in this section.

Reset Ref - This option is used to change the reference point of each object on the select list back to its default, the center of the object's extent.

Deeper - This option permits selection of objects within a group (refer to ***Grouped Objects and Expressions*** in this section). Each time *Deeper* is chosen, the *Depth*= indicator is displayed in the message area. A number following this indicator reveals how deep within groups an object can be selected. The default level is zero. A level of one permits selecting objects within groups, a level of two allows selecting an object which is inside of a group which is inside of another group, and so on.

OPTIONS PULL-DOWN MENU

Click on *Options* in the menu bar of the *Graphic Configuration* window to view the *Options* pull-down menu.

Model Type - Select graphic or submodel; items stored as submodels are stored in a different location than items stored as graphics. Items stored as submodels can be used as objects in other graphics, with different data.

After a model has been created and saved as a graphic, it may later be used as a submodel. To convert a model from a graphic to a submodel, do the following:

1. Load the graphic into the graphics editor, as described previously in this chapter.

2. Click on *Options* in the *Graphic Configuration* window menu bar. In the resulting pull-down menu, select the option *Submodel* to define the *Model Type*.

3. Click on *File* in the *Graphic Configuration* window menu bar, and select the *Save As* item from the resulting pull-down menu. Enter a name which is different from the name which was used to load the graphic, and press Enter. The model will be saved as a submodel with this name.

4. (It is important to use a new name for the submodel during the *Save As* operation. If the name of the original graphic is used, a graphic and a submodel with the same name will exist, which can cause confusion).

5. If the graphic is no longer needed, delete it using the *Delete* option on the *File* pull-down menu.

Single Select and *Multi Select* - These options control the behavior of the graphics editor during object selection operations. In *Single Select* mode (the default), clicking on an object to select it automatically causes previously selected objects to be deselected. In *Multi Select* mode, clicking on an object to select it causes the object to be added to the list of previously selected objects (if any).

> **NOTE:** All objects are deselected when the *End* button or right mouse button are clicked, or when a different working view child window is selected. Also, the Single Select and Multi Select modes have no effect on selections made via the *Select* pull-down menu (refer to ***SELECT PULL-DOWN MENU*** in this section).

Equal Scaling - when this option is selected, the scaling factors in the *x* and *y* directions are forced to be equal during scaling operations. Thus, a square will always remain a square when *Equal Scaling* is selected.

Unequal Scaling - When the *Unequal Scaling* option is selected, the *x* and *y* scaling factors may be unequal when an object or objects are scaled. This means a square can become a rectangle or a circle can become an ellipse after scaling.

Hilite - Enclose each submodel in the working area inside a rectangle.

Show Position - This option displays the coordinates of the current pointer position in the working view. This information is displayed to the right of the *Select* status indicator in the message area. (The working view area is always considered 100 units wide by 75 units high, regardless of how the *Graphic Configuration* window which contains it has been sized).

WINDOW PULL-DOWN MENU

Click on *Window* in the menu bar of the *Graphic Configuration* window to view the *Window* pull-down menu.

New - When the *Graphic Configuration* window is first opened, it contains a single child window, which has been maximized to fill the window frame of the parent window. *Restore* and *Window Menu* buttons for the child window are located at opposite ends of the *Graphic Configuration* window menu bar. When the New option is clicked on:

- An additional child window is opened.
- If an existing child window was maximized, it will be restored within the parent window (i.e., it will be reduced to match the new child window in size, will have its own window frame).

Each time the *New* item is clicked on, an additional child window will be opened, until the maximum number of child windows is reached. This allows multiple graphics and submodels to be open for configuration simultaneously. The title bar of each child window contains the number of the child window, the model name (by default, NoName_1, NoName_2, and so on), and the model type (graphic or submodel). When the maximum number of child windows has been reached, attempts to open an additional child window will cause the following message to be displayed in the message area.

Can't open any more windows

An object created in one child window can be selected, dragged and dropped into other child windows. All attributes configured for the object, including dynamics, will be copied as well.

> **NOTE:** It is possible to edit the same submodel (or graphic) in two or more child windows simultaneously. If a submodel or graphic has already been opened in a child window, attempting to open it for configuration in another child window will cause a confirmation pop-up window to appear, which advises that the item is open for configuration by another user; clicking on *Yes* in this pop-up window allows the item to be opened for configuration in another child window. Because configuration changes made in the different child windows are independent of each other, changes made in one child window can completely overwrite changes made to the same graphic or submodel in another child window. To avoid confusion, save the item to a different name in each child window.

Cascade - The *Cascade* and *Tile* options are used to rearrange working view child windows so that all are visible in the *Graphic Configuration* window. The *Cascade* option causes the child windows to overlap so that each title bar is visible.

Tile - Click on Tile to arrange the open working view child windows in smaller sizes to fit next to each other in the *Graphic Configuration* window.

Child Window List - The bottom portion of the *Window* pull-down menu contains a list of child windows which are currently open in the *Graphics Configuration* window. The number of the child window, the name of the submodel or graphic it contains, and the model type (graphic or submodel) are included in the list. If unsaved changes have been made in the child window, an asterisk (*) will be displayed following the model type. When one of the entries in the list is clicked on, the associated child window will be selected; it will also be raised, if necessary.

To close an individual child window, click on the window menu button found at the left end of the child window's title bar, then click on *Close*. A confirmation box will be displayed, which provides the option of saving changes, discarding changes, or

abandoning the *Close* operation to leave the child window intact.

NOTE: The last child window displayed in the *Graphics Configuration* window cannot be closed.

OBJECT PLACEMENT EXAMPLE

The following example attempts to illustrate the steps required to create and modify a submodel instance (Figure 4-28) using the graphics editor.

1. Create instance of ISA_VLVE_ACTR (Figure 4-28) in the working view of *Graphic Configuration* window (refer to ***CREATING OBJECTS - SUBMODELS*** in this section).

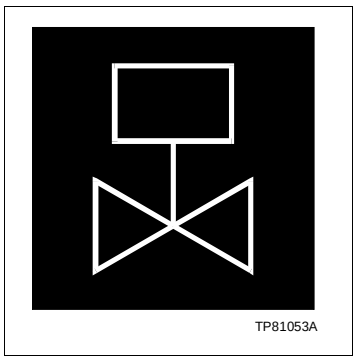


Figure 4-28. Model of ISA Standard Actuator Valve Symbol

2. Move the valve to a different position. Notice that the pointer appears in the center of the submodel. This and all subsequent steps require that the submodel be selected, i.e., it must be on the select list.

3. Scale to the desired size, if necessary.

4. Set *Grav* mode in the *Status Control Panel*.

5. Select *Change Ref*[erence] from the *Select* pull-down menu.

6. Place the pointer at the center of the left side of the valve body (one of the predefined submodel points; refer to ***POINT SNAP MODE*** in this section) and click (Figure 4-29).

7. Set *Grid* mode in the status control panel.

8. Move the submodel to the desired grid intersection. Notice that the cursor now appears at the new reference point.

This reference point change allows for lines to be easily connected to the sides of the valve body.

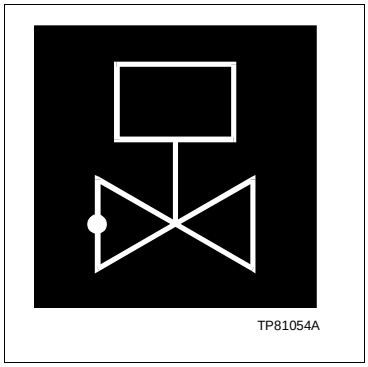


Figure 4-29. Actuator Valve Symbol with Pointer at New Reference Point

DYNAMICS

Dynamics are specifications for changes in graphic objects. These changes can be in response to changes in variables such as atoms in an INFI 90 OPEN server. For example, when an atom changes state or value, this can be translated into a graphical change in a screen object, such as a change in color of a filled rectangle.

Dynamics can also specify nongraphic actions, to be performed in response to user actions, such as vectoring to a different display or changing the setpoint of a INFI 90 OPEN control module.

Dynamics can be configured by several methods, all of which are accessed from the *Dynamics* pull-down menu (which is available from the menu bar on the *Graphic Configuration* window):

- *Configure Attributes* dynamics which affect a limited set of graphic attributes of an object or submodel, based on the value of a user defined expression, can be configured through the *Configure Attributes* item.

- *Configure Submodel* dynamics defined in a submodel can be configured (customized) for an instance of that submodel, via the *Configure Submodel* item.

- *Create Custom* dynamics and *Declare Variable*s. Custom dynamics can be created, using the GML language, via the

Create Custom Dynamics and *Declare Variables* menu items.

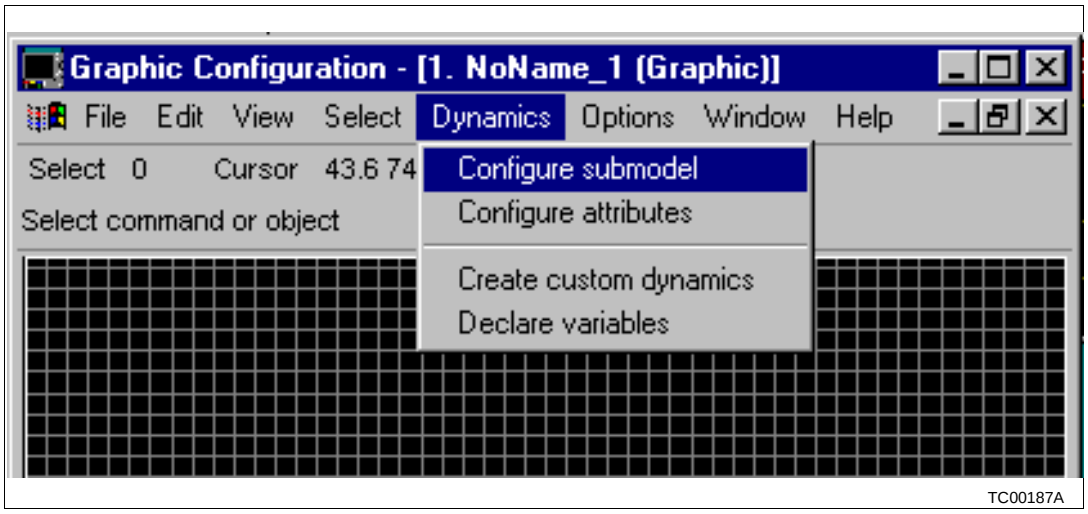


Figure 4-30. Dynamics Pull-Down Menu

NOTE: The following alternative methods can be used to access configuration windows for submodels and expression dynamics. If the pointer is positioned over a submodel instance, and the center mouse button is clicked, then the *Submodel Configuration* window appropriate to the submodel will be displayed. Submodel configuration is discussed in ***BAILEY SUBMODELS*** in Appendix B. If the pointer is positioned over a graphic object (other than a submodel instance) and the center mouse button is clicked, then the *Expression Dynamics* window will be displayed. (The *Expression Dynamics* window is also accessed via the *Configure Attributes* item on the *Dynamics* pull-down menu).

Configure Attributes - Dynamic Attributes for Objects

To configure dynamic attributes for an object (including submodels), the object must exist in the working view and must be selected. (Click on the object to select it). Next, click on *Configure Attributes* in the *Dynamics* menu. The *Expression Dynamics* window (Figure 4-31) will appear. (Alternatively, just position the pointer on the object to be configured and click the center mouse button. This action will both select the object and open the *Expression Dynamics* window).

The *Expression* entry box on the *Expression Dynamics* window allows the user to enter mathematical operators, parentheses, variables and data values in a syntax like that of the C programming language, to create an expression which can be true (1) or false (0). Below the *Expression* entry box are two sets of attributes with associated toggle buttons and entry boxes. The first set of attributes is labelled *True*; the second set is labelled *False*. This allows the dynamic attributes of an object to be configured separately for both true and false conditions, after

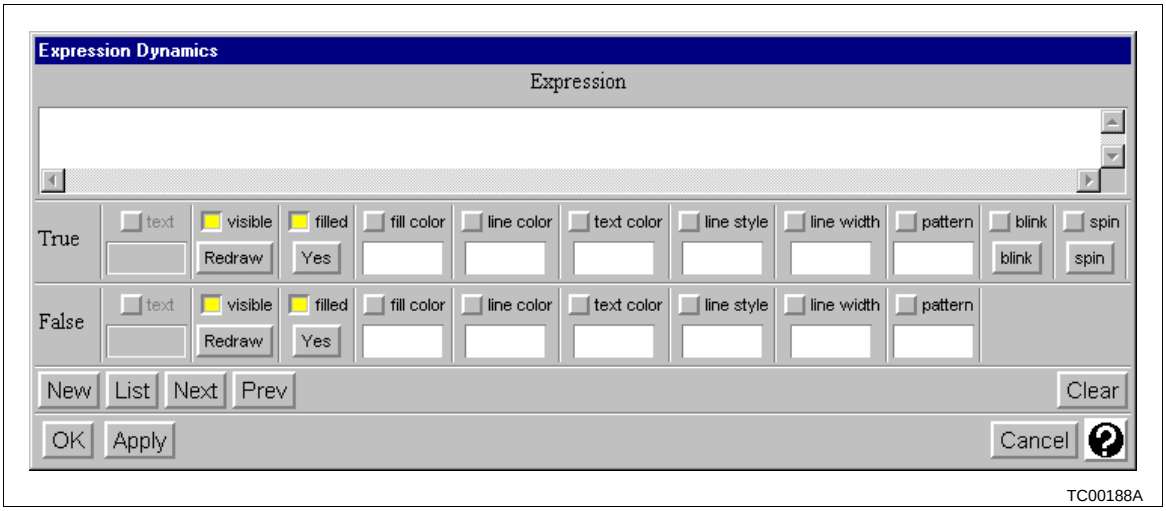


Figure 4-31. Expression Dynamics Window

the expression has been configured. Dynamic attributes which can be configured via the *Expression Dynamics* window include:

- Visibility.
- Line color.
- Text.
- Line style.
- Text color.
- Line width.
- Fill.
- Blink.
- Fill color.
- Spin.
- Fill pattern.

Examples are provided in ***Dynamic Configuration Examples*** in this section.

> **NOTE:** Configuring an expression (or set of expressions) which exceeds 3500 bytes for a given object will cause an error message to be displayed.

CONFIGURING THE EXPRESSION - OPERATORS AND OPERANDS

Operators are symbols used in expressions to represent arithmetic, logical, or relational operations. An expression consists of operators and operands, or values. Operands used in these expressions can be of the following types:

- A tag database point (in the form tagname.atom) - the current value of the atom is used in the expression.

- A numeric constant, i.e., 3.14.

- A variable name, such as *LENGTH*, can be used if the model loaded in the graphics editor is a submodel, i.e., in the entry for the less than operator (<) in Table 4-3 below, the example could be written *LENGTH* < 100, instead of LIC-454.C_APV < 100.

In the following examples, several database points are used as operands. Assume that atoms C_APV, C_ASP and C_AOUTPUT have numeric values, while atoms EB_PRALMST and EB_INHB have logical values, i.e., 0 (false) or 1 (true).

Table 4-3. Operator Usage - Description and Examples

Operator	Description	Usage	Example
<	less than	Evaluates two expressions and is true if the one on the left is less than the one on the right.	LIC-454.C_APV < 100
<=	less than or equal to	Evaluates two expressions and is true if the one on the left is less than or equal to the one on the right.	LIC-454.C_APV <= 100.7
>	greater than	Evaluates two expressions and is true if the one on the left is greater than the one on the right.	LIC-454.C_APV > LIC-454.C_ASP
>=	greater than or equal to	Evaluates two expressions and is true if the one on the left is greater than or equal to the one on the right.	LIC-454.C_APV >= 0.25
==	equal	Evaluates two expressions and is true if the one on the left is equal to the one on the right.	FIC-477.EB_INHB == 0
!=	not equal	Evaluates two expressions and is true if they are not equal.	FIC-477.EB_PRALMST= 0
+	add	The addition operator causes the two values on either side of it to be added together yielding the *sum* of the two values.	LIC-454.C_APV + LIC-454.C_ASP > 500
-	subtract	The number after the - sign is subtracted from the number before the sign.	FIC-477.EB_OUTPUT - LIC-454.C_ASP == 0
*	multiply	The multiplication operator multiplies the values on either side of it, yielding the *product* of the two values.	LIC-454.C_APV * 0.5 < 4658.0
/	divide	The value to the left of the **/** is divided by the value to the right.	(LIC-454.C_APV + 500) / 2 > FIC-477.EB_OUTPUT
-	minus	The minus sign is used to indicate or change the algebraic sign of a value. This is a unary operator which takes just one operand.	FIC-477.EB_OUTPUT > -50
!	logical not	Evaluates the expression on the right and is true if that expression is false, and vice versa.	FIC-477.EB_PRALMST
&&	logical and	Evaluates two expressions and is true if and only if both are true.	FIC-477.EB_PRALMST&& FIC-477.EB_INHB
||	logical or	Evaluates two expressions and is true if either one or both are true.	LIC-454.C_APV - LIC-454.C_ASP < 0.5 || LIC-454.C_ASP - LIC-454.C_APV < 0.5

(The example for the logical Or is True if the values of the C_APV and C_ASP atoms of process tag LIC-454 are within 0.5 of each other).

> **NOTE:** A single tagname.atom, with no operators, is also a valid expression; this is useful with bit type atoms which can have a value of 1 or 0.

When a atom can be represented a text or a numeric, this can be specified using the ***_T*** or the ***_N*** suffixes. This is required when the fprimary format is not suitable for comparing to a string or a value. Note that the combined length of the atom and suffix can not exceed 16 characters.

Because expressions can be complex, and often include multiple operators and operands, a set of rules exists to determine which operators get performed first, which get performed second, third, and so on. This order is known as precedence. Some operators, such as * (multiply) and / (divide) have equal precedence. If operators of equal precedence are used in an expression, they are performed according to the order in which they appear in the expression, left-to-right. (The unary operators ! (logical not) and - (minus) are exceptions to this rule and are evaluated right-to-left; unary operators act on just one operand. All other operators which can be used in a *Dynamics* expression are binary operators, and act on two operands).

Precedence is an important concept to remember, as its effects on the evaluation of an expression can confound the unwary. For example, consider the following expression:

*2 + 3 * 4*

The resulting value of this expression is 14, not 20. Because the * (multiplication) operand has a higher precedence than the + (addition) operand, the multiplication portion of the expression (3x4) is performed first. If the expression were written:

*(2 + 3) * 4*

the value of the expression would be 20, because expressions contained within parentheses are evaluated first.

Table 4-4 below contains a list of the operators which can be used in a *Dynamics* expression on the PWC90 console; in this table, the operators with equal precedence are grouped together. Operator groups with higher precedence are located higher in the table.

Table 4-4. Operators In Order Of Precedence

Symbol	Description
()	parentheses
!	logical not (unary)

Table 4-4. Operators In Order Of Precedence (continued)

Symbol	Description
-	minus (unary)
*	multiply
/	divide
+	add
-	subtract
<	less than
<=	less than or equal
>	greater than
>=	greater than or equal
==	equal
!=	not equal
&&	logical AND
||	logical OR

In addition to the symbols listed in Table 4-4, aliases exist for some operators. These aliases, listed in Table 4-5, are provided as an aid to the user. They can only be used when creating an expression. Once the expression has been saved, the aliases will be replaced by the standard operator symbols. Aliases must be entered in upper case letters, and must be separated from adjacent operands by spaces.

Table 4-5. Operator Aliases

Alias	Standard Symbol
GT	>
GE	>=
LT	<
LE	<=
EQ	==
=	==
NE	!=
OR	||
AND	&&
NoT	!

Expression True

Entering a single tagname.atom or mathematical expression, as described above, allows the user to configure dynamic attributes for selected objects. It is also possible to change static attributes of submodels via the *Expression Dynamics* window. To do so, first select the submodel to be changed. Click on *Dynamics* in the menu bar, then click on *Configure*

Attributes to call up the *Expression Dynamics* window. Type the word **True** (upper case, lower case, or capitalized) in the *Expression* entry box. Change the attributes of the submodel by entering values in the set of attribute setting boxes labelled *True* (refer to ***CONFIGURING THE ATTRIBUTE SETTINGS*** in this section).

It is also possible to configure the static attributes of objects in this manner. This is not recommended, however, as this method is time consuming and inefficient when compared with object configuration performed directly from the various control panels located in the *Mode* and *Attributes* windows, as described in ***CREATING MODELS*** in this section.

Grouped Objects and Expressions

A group of objects and/or submodels can be configured so that they are all affected by the same expression. (This should be done prior to selecting the *Configure Attributes* item). To accomplish this, *Multi Select* mode (refer to ***OPTIONS PULL-DOWN MENU*** in this section) must first be selected. Next, click on all objects that are to be grouped, then click on *Edit* in the *Graphic Configuration* window menu bar, then click on *Group* in the resulting cascading menu.

Once a set of objects has been placed in a group, any operations (i.e., *Move, Delete*) will be performed on all objects in the group.

The *Ungroup* item on the *Edit* menu will remove all objects from the group. Any dynamics attached to the group via the *Configure Attributes* entry will be deleted.

Multiple Expressions

More than one expression can be configured for an object. This is useful if the attributes of the object should change under more than one condition. For example, a filled rectangle can be configured to be:

Green When the value of a tag database point is less than 50.

Yellow When the value is between 50 and 100 (inclusive).

Red When the value is above 100.

To add another expression, click on *New* in the *Expression Dynamics* window (located below the *False* attribute section). The label at the top of the window will change to:

Expression(n of m)

where *m* is the total number of expressions for this object, and *n* is the expression currently displayed. Use the *Prev* and *Next* buttons (located below the *False* attribute section) to move between configured expressions).

To delete the expression currently on display, click on *Clear* (above the Help icon).

To make permanent any changes to the *Expression Dynam* window, click on *OK* or *Apply*. To exit without making changes, click on *Cancel*.

To view a list which summarizes all the expressions and associated attribute settings which have been configured for an object, click on *List*. The *Expressions* window (Figure 4-32) will be displayed. Click on an expression in the list to cause that expression to be displayed in the *Expression Dynamics* window. To close the *Expressions* window, click on *OK*.

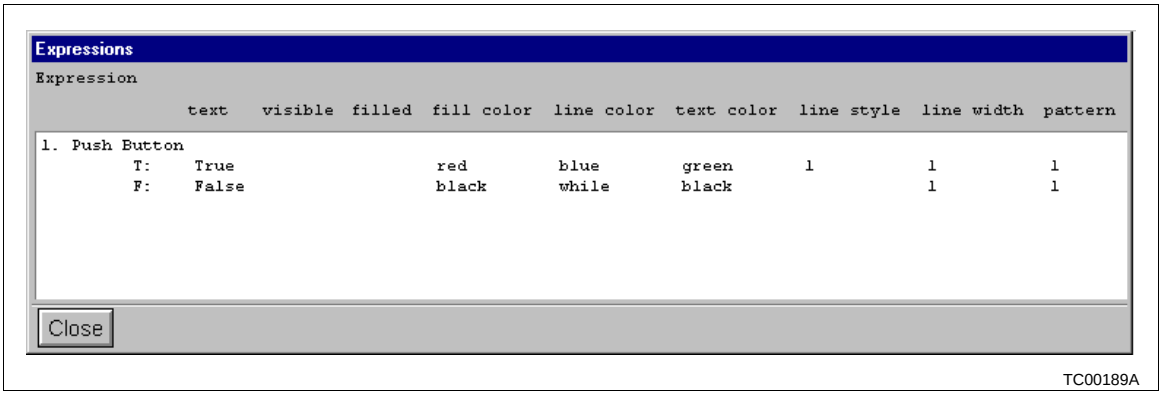


Figure 4-32. Expressions Window

Note that it is not necessary to configure both *True* and *False* attribute settings for an expression.

An expression is considered active when it evaluates to true and has settings for *True*, or when it evaluates to false and has settings for *False*. While expressions can be configured so that only one expression is active at a time, it is also possible to configure expressions so that this is not the case; for example:

1. An object has more than one active expression. The expressions are evaluated, and their settings applied, in the order in which they were defined. This means that the effects of an expression with a low number (i.e., 1) may be overridden by an expression with a higher number (i.e., 10).

2. An object has no active expressions. If none of the expressions with *True* settings are true, and none of the expressions with *False* settings are false, then the object is not changed.

CONFIGURING THE ATTRIBUTE SETTINGS

Below the *Expression* entry box are two rows of attribute setting boxes. The first row is labelled *True*; a text description and toggle button are located at the top of each attribute setting box. Assignments configured here control the appearance of the selected object or submodel when the expression in the *Expression* entry box evaluates to *True* (1).

Below this section of the *Expression Dynamics* window is a second row of attribute setting boxes, identical to the first, except these are labelled *False*. These assignments control the appearance of the selected item when the expression in the *Expression* entry box evaluates to *False* (0).

A value or variable can be entered in each of the attribute setting boxes which contains a text entry box. If a variable is entered in such an attribute setting box, and the object is saved as a submodel, that variable can later be used via the *Configure Submodel* menu item (refer to ***Configure Submodel*** in this section) to assign the attribute independently, for each instance of the submodel. This is particularly useful when a submodel is instanced more than once when creating a new submodel. For example, if a submodel called ***FP_DYbargraph*** contains a variable called TagAtom, and ***FP_DYbargraph*** is instanced twice on a new submodel, this variable could be assigned the variable name TagAtom1 in the first instance, and TagAtom2 in the second instance.

Some attribute setting boxes contain a button instead of a text entry box. Click on this button to configure the attribute setting; in some attribute setting boxes, the button toggles between two states (i.e., *Yes* and *No*), while in others the button provides access to additional configuration windows.

Click in the text/toggle button portion or in the button/text entry portion of the attribute setting box to select the attribute for configuration; the toggle button will turn yellow, indicating that the attribute has been selected. If the attribute has a text entry box, a blinking cursor will appear in that entry box and the desired value, expression, or variable can be entered. If neither portion of the attribute setting box is clicked on, the attribute will remain unaffected. If the attribute is selected but no entry is made in the entry box, the attribute definition will use the default value.

Each attribute has a default value, and (except for the *Text* attribute, and some *Blink* and *Spin* attribute parameters) a minimum and maximum permitted value. The default value is used when the attribute entry box has been clicked on but no entry is made, or when the value entered is outside the range of the permitted values (less than the minimum or greater than

the maximum). Default values for the attributes are as follows. (Table 4-6)

> **NOTE:** It is possible to configure dynamic attributes which are unnecessary for an object; configuration of such items will have no effect on the object. For example, if the object is a straight line, and the Filled attribute is configured, it will have no effect on the appearance of the line.

Table 4-6. Attribute Values Table

Attribute	Default	Minimum	Maximum
Text	Empty string ()	none	none
Visible	1 (visible)	0	1
Filled	1 (filled)	0	1
Fill Color	0 (black)	0	63
Line Color	0 (black)	0	63
Text Color	0 (black)	0	63
Line Style	1 (solid line)	0	4
Line Width	1 (narrow)	1	4
Pattern	1 (solid fill)	1	4
Blink Period	5 (tenths of a second)	1	none
Blink Phase	0 (Blink Periods)	0	none
Blink Color 1	1 (white)	0	63
Blink Duration 1	1 (Blink Periods)	1	none
Blink Color 2	0 (black)	0	63
Blink Duration 2	1 (Blink Periods)	1	none
Steady Color	7 (white)	0	63
Spin Period	5 (tenths of a second)	3	none
Spin Angle	20 (degrees)	-180	+180

Text

The *Text* attribute is used to change the text string in a text object or filled text rectangle; it does not change text size or font, and it cannot be used to change text in a previously configured submodel. If the attribute box is selected and nothing is entered, the string defaults to an empty string ().

Visible

The *Visible* attribute is used to configure the selected object to be visible, invisible, or redrawn when the expression evaluates to true or false. If the toggle button for the attribute is selected, the face of the selection button will display the word *Yes* (visible, the default), *Redraw*, or *No* (invisible). Clicking on the selection button will cause it to cycle through these three states. The visible/invisible attributes are useful, for example,

if an object containing a warning message is to be visible only under certain conditions. The *Redraw* attribute is equivalent to the *Yes* attribute, but with the additional feature that the object will be redrawn each time the value of the expression (or any tagname or variable used in the expression) which governs it changes. This is useful for objects which overlap and must remain on top of some dynamic object.

Filled

Use the *Filled* attribute to configure the selected object to be filled or unfilled when the expression evaluates to true or false. If the toggle button for the attribute is selected, the face of the selection button will display the word *Yes* (the default) or *No*. Clicking on the selection button will cause it to toggle back and forth between *Yes* (filled) and *No* (unfilled).

Fill Color

Use the *Fill Color* attribute to configure the fill color of the selected object. With the toggle button for the attribute selected, simply move the pointer to the *Color Attribute Control Panel* (in the lower left corner of the *Graphic Configuration* window) and click on the desired color. The number of the selected color (from zero to 63, displayed on the face of each color button) will appear in the entry box portion of the *Fill Color* attribute setting box. Alternatively, the number of the desired color can also be entered via the keyboard by typing the number when the *Fill Color* attribute is selected. If this attribute is selected and no entry is made in this entry box, the attribute will default to zero (black).

Line Color

The *Line Color* attribute affects both lines and edges of objects in the selected object. Enter the number of the color to be used as the edge color by either clicking on the desired color in the *Color Attribute Control Panel* or by entering the number of that color via the keyboard. (The color number, from zero to 63, is located on the face of the color sample in *Color Attribute Control Panel*). If this attribute is selected and no entry is made in this entry box, the attribute will default to zero (black).

Text Color

The *Text Color* attribute affects objects in the selected object. Enter the number of the color to be used as the text color by either clicking on the desired color in the *Color Attribute Control Panel* or by entering the number of that color via the keyboard. (The color number, from zero to 63, is located on the face of the color sample in the *Color Attribute Control Panel*). If this

attribute is selected and no entry is made in this entry box, the attribute will default to zero (black).

Line Style

The *Line Style* attribute affects edges and lines in the selected object. The left side of the *Color Attribute Control Panel* (Figure 4-24) contains examples of the five different line/edge styles, which are labelled from zero to four, and also contains an indicator box for each style. Enter the number of the desired style in the *Line Style* entry box, either by clicking on the appropriate indicator box in the *Lines Attribute Control Panel*, or by entering the number via the keyboard. If this attribute is selected and no entry is made in this entry box, the attribute will default to one (solid line).

Line Width

Like the *Line Style* attribute, the *Line Width* attribute affects edges and lines in the selected object. The right side of the *Lines Attribute Control Panel* (Figure 4-23) contains examples of the five different line/edge widths, which are labelled from zero to four. (Line/Edge Width zero is the same as Line/Edge Width 1; no check box exists for Width zero). Enter the number of the desired width in the *Line width* entry box, either by clicking on the appropriate indicator box in the *Lines Attribute Control Panel*, or by entering the number via the keyboard. If this attribute is selected and no entry is made in this entry box, the attribute will default to be (narrow).

Pattern Attribute

The *Pattern* attribute allows the user to specify one of four fill patterns. Examples of available fill patters are displayed in the *Fill Pattern Attribute Panel* (Figure 4-22). The patterns are labelled from one to four. Enter the number of the desired pattern in the *Pattern* entry box, either by clicking on the appropriate indicator box in the *Fill Pattern Attribute Panel*, or by entering the number via the keyboard. If this attribute is selected and no entry is made in this entry box, the attribute will default to one (solid fill).

NOTE: To observe changes to submodel dynamics on a graphic in an operator window, the graphic must be saved and re-installed.

Blink

The multiphase *Blink* attribute (available only on objects in graphic displays) is used to cause an object to alternate between two colors when the expression is true, or to be displayed steadily in a third color when the expression is false.

Click within the *Blink* attribute setting box to select the attribute; the *Blink* window (Figure 4-33) will appear.

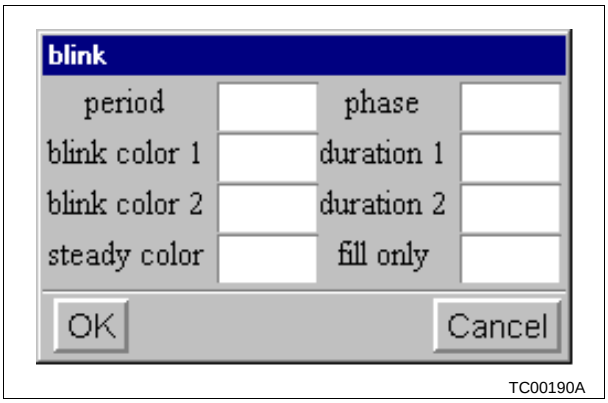


Figure 4-33. Blink Window

Use the entry boxes in the *Blink* window to enter values for the following parameters:

The *Period* is the blink period, which is measured in tenths of a second. The number entered here defines how many tenths of a second are in the blink period for this object. The minimum value for the period is one (one tenth of a second); the default value for the period is five (1/2 second).

The *Phase* is the initial delay time (in blink periods) which elapses before blinking starts after the expression becomes true. By using different phase values for different objects, motion can be simulated (refer to ***Dynamic Configuration Examples*** in this section for an example). The phase value is multiplied by the period to determine how long (in tenths of a second) to delay the start of blinking. Both the minimum and default value for this parameter is zero (zero).

Blink Color 1 is the first of the two blink colors between which the object will alternate. Enter this value by clicking on the desired color in the *Color Attribute Panel* (Figure 4-24), or by typing it in via the keyboard.

The value entered for *Duration* 1 is multiplied by the *Period* to determine how long (how many tenths of a second) the object will be displayed in *Blink Color* 1. Both the minimum value and default value for this parameter are one (1 blink period).

Blink Color 2 is the second of the two blink colors between which the object will alternate. Again, enter this value by clicking on the desired color in the *Color Attribute Panel* (Figure 4-24), or by typing it in via the keyboard.

The value entered for *Duration* 2 is multiplied by the *Period* to determine how long (how many tenths of a second) the object

will be displayed in *Blink Color* 2. Both the minimum value and default value for this parameter are one (1 blink period).

The *Steady Color* parameter is used to define the nonblinking color in which the object will be displayed when the expression is false. Like *Blink Colors* 1 and 2, this value can be entered either by clicking on the desired color in the *Color Attribute Panel* (Figure 4-24), or by typing it in via the keyboard. As discussed in ***Multiple Expressions*** in this section, multiple expressions can be configured for an object or group of objects. Expressions are evaluated in the order that they were defined. This means that the effects of Expression 1 may be overwritten by a subsequent expression (Expression 10, for example). When multiple color controlling expressions have been defined for an object, assigning the value -1 (minus 1) to the *Steady Color* parameter causes the object to be displayed in the last color used (when the expression controlling the *Blink* attribute evaluates to False). The default setting for the *Steady Color* parameter is -1.

The *Fill Only* parameter is designed for use with objects that belong to the same group. (***Grouped Objects and Expressions*** in this section describes how objects are placed in a group). If the *Fill Only* parameter is set to false (0), then all portions of all objects in the group (i.e., the fill color, line color and text color (if any)) will change color when either *Blink* or *Steady Color* parameters are in effect. If the *Fill Only* attribute is set to true (1), then only the fill color of objects in the group will be changed, while other attributes of objects (i.e., line color, text color) remain unaffected. The *Fill Only* parameter is visible on the *Blink Configuration* window only when a group of objects is being configured. The default setting for the *Fill Only* parameter is 1 (true).

Spin

The *Spin* attribute causes an object to be rotated about its reference point in periodic increments, for as long as the expression is true. Click within the *Spin* attribute box to select the attribute. The *Spin Configuration* window (Figure 4-34) will appear.

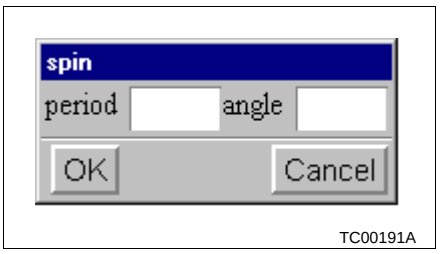


Figure 4-34. Spin Configuration Window

Use the *Spin Configuration* window to enter values for the following parameters:

- Period.
- Angle.

As in the *Blink* attribute, the *Period* parameter for the *Spin* attribute is measured in tenths of a second. The number entered for this parameter defines the length of time it will take (in tenths of a second) for the object to be moved the number of degrees specified in the Angle parameter. The minimum value for the Spin Period is three (tenths of a second), the default is five (one half second).

The number entered in the *Angle* parameter entry box specifies the number of degrees the object will turn per period. A positive number results in counterclockwise movement, while a negative number causes clockwise movement. When the expression is false, the object returns to its original position.

> **NOTE:** The spin feature is already incorporated in the dynamic submodels FP_DYwheel, FP_DYpump_rt and FP_DYpump_lf, which are supplied by Bailey.

Combining Attributes

Note that when either the *Blink* or *Spin* attribute is selected, all other attributes are stippled out and made unavailable. It is possible to combine the *Blink* and *Spin* attributes with the other dynamic attributes and with each other by using multiple expressions (***Multiple Expressions*** in this section), and creating a separate expression each for the *Blink* attribute, the *Spin* attribute and all other attributes. Avoid configuring two different sets of *Blink* (or *Spin*) dynamics (for the same object) which could be true simultaneously, as this can cause unpredictable results.

Configure Submodel

A submodel may already contain internal dynamics, configured by Bailey or by the user during an earlier configuration session. This can be done via the *Expression Dynamics* window discussed above, by either creating a variable name in the *Expression* entry box, or if a variable was entered in an attribute setting box, instead of a value. Refer to ***CONFIGURING THE EXPRESSION - OPERATORS AND OPERANDS***, ***CONFIGURING THE ATTRIBUTE SETTINGS*** and Example 3 in ***Dynamic Configuration Examples*** in this section.

The *Submodel Configuration* window (Figure 4-35) is used to assign values (which can be in the form of tag database points) to dynamic variables. To display this window, a submodel in the working view area should first be selected; click on the sub-

model to select it. Next, click on *Dynamics* in the *Graphic Configuration* window menu bar, then click on *Configure Submodel* in the resulting pull-down menu. The selected submodel will be highlighted by a surrounding white rectangle.

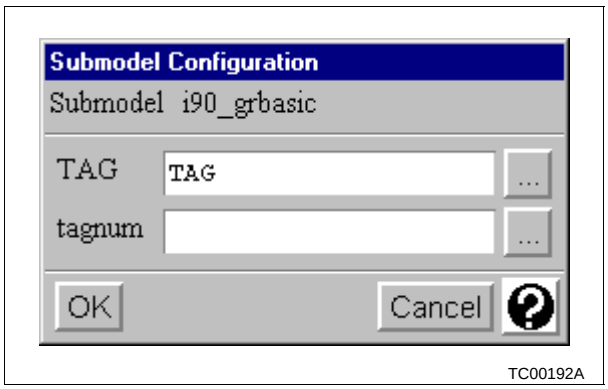


Figure 4-35. Submodel Configuration Window

On the *Submodel Configuration* window, all variable names in the submodel's dynamics are listed on the left. Next to each variable is a text entry box, which contains any value which has been assigned to the variable during configuration. If no value has been assigned to a variable, the entry box is blank.

The value which can be assigned to a variable can be one of the following types:

- Numeric constant.
- String constant.
- Tag database point (in the form tagname.atom).
- Different variable type (if the current model type is defined as submodel).
- Expression.

Following each text entry box is an *Expanded Entry* button with an ellipsis (**...**) on the button face. To view a part of the text string or expression that is not visible in the text entry box, click on this button. A pop-up window will be displayed. This window contains an enlarged, scrollable version of the associated text entry box, allowing more of the variable value to be visible. The variable value can be edited in this pop-up window, and can be up to 128 characters in length. Click on *OK* to exit and save any changes, or click on *Cancel* to exit without modifying the variable's value.

Example 4 in ***Dynamic Configuration Examples*** in this section shows the configuration of a user created submodel. The

list of dynamic Bailey Submodels and their variables can be found in ***BAILEY SUBMODELS*** in Appendix B.

Create Custom Dynamics and Declare Variables

As described above, it is possible to configure complex dynamic properties for an object using the *Configure Attributes* menu item. The dynamics which can be configured using this method may not be adequate for all applications, however. Two items on the *Dynamics* menu, *Create Custom Dynamics* and *Declare Variable*s, allow configuration of additional dynamic properties.

CUSTOM DYNAMICS

Custom dynamic properties can be configured using DynProps to give an object dynamic properties which are not available using the methods described earlier. Configuration using DynProps, however, is also more complex. DynProps are text statements containing descriptions of dynamic behavior, written using the syntax of the GML language. The capabilities and syntax of DynProps are discussed in Chapter 4 of the ***SL-GMS Reference Manual***, which is available as an option from Bailey.

Configuring DynProps

Select the *Create Custom Dynamics* item from the *Dynamics* menu to display the DynProp, if any, attached to the currently selected object. The text of the DynProp is displayed in the *Edit Custom Dynamics* window (Figure 4-36).

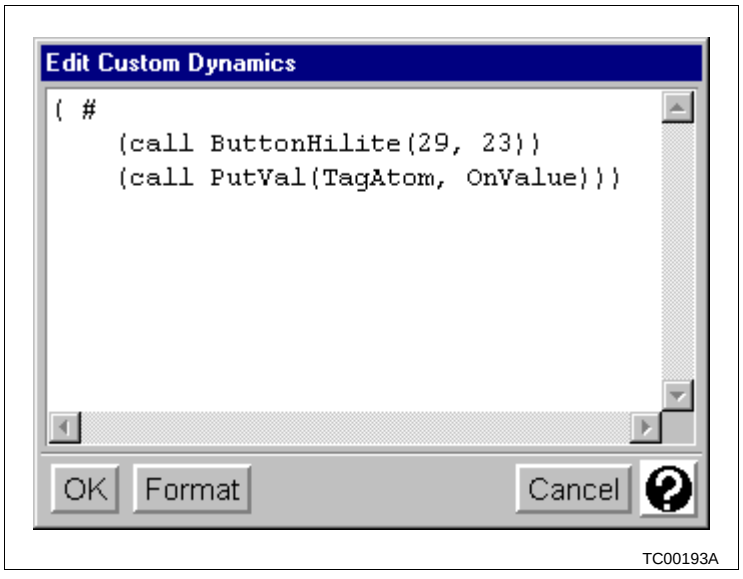


Figure 4-36. Edit Custom Dynamics Window

Modify the DynProp by adding or deleting text, then click on *OK*. If the syntax of the DynProp is correct, the *Edit Custom Dynamics* window will close. If the syntax is incorrect, an error message will appear and the window will remain on display. The *Undo* button at the bottom of the window can be used to undo the last modification to the DynProp text. Click on *Format* (also located at the bottom of the window) to automatically indent the DynProp text according to convention; the content of the DynProp will not be changed.

An explanation of the capabilities and the syntax of DynProps is beyond the scope of this manual. Refer to the ***SL-GMS Reference Manual*** for details.

DynProp Variables

A DynProp will contain references to variables. For example, the fill color of an object in a submodel can be changed using the following DynProp:

```
( * (fcolor bar_color))
```

In this example, bar_color is a variable. The DynProp is then saved by clicking on *OK* and exiting the *Edit Custom Dynamics* window, and the submodel must be saved (using the *Save, Save As* or *Install Current* items from the *File* pull-down menu).

When an instance of this submodel is placed on a graphic, selected and the *Configure Submodel* menu item clicked on, *bar_color* and an associated text entry box will appear in the list of variables in the resulting *Submodel Configuration* window. Enter a color index number between zero and 63 in the entry box to assign that number as the value of the variable.

In other cases, it may be desirable to have the value of a variable retrieved from the global database when the graphic is displayed in the operator window. In such cases the *Declare Variables* menu item must be used.

DECLARING VARIABLES IN GRAPHICS

The *Declare Variables* menu item provides a way to incorporate dynamic data from the global database into graphic displays. *Declare Variables* works in conjunction with the *Create Custom Dynamics* menu item. For example, a text object can be configured to display the value of a setpoint (C_ASP) atom in the global database on a graphic display. (This could also be done using the FP_DYvalue submodel, described in ***FP_DYvalue*** in Appendix B).

To do so, first configure the DynProp for the text object as follows:

(* (stext _psetpoint %.1f))

For the purposes of this example, the tagname used for the point in the global database is PIC-1515.

Next, the value of the variable _psetpoint must be tied to the value of the desired tag.atom in the global database (for the purposes of this example, we will use PIC-1515.C_ASP). To do so, click on *Declare Variables*. The *Variables* window (Figure 4-37) will be displayed.

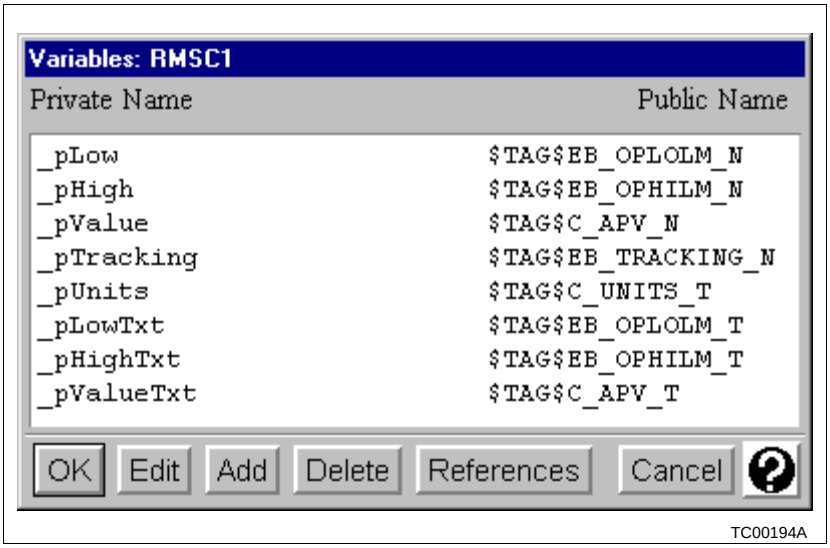


Figure 4-37. Variables Window

The top portion of the *Variables* window contains a list box. Once a variable has been configured, it will occupy a line in the list box, with the *Private* name of the variable on the left side of the entry, and the corresponding *Public* name on the right. The private name of a variable is the name used in the DynProp, (i.e., _psetpoint in this example). The public name of a variable defines the global database point which supplies the value for the variable (i.e., PIC-1515.C_ASP in this example).

Below the list box are a number of buttons, including *Edit, Add, Delete, References*, *OK* and *Cancel*.

To create a new variable, click on *Add*. The *Declare Variable* window (Figure 4-38) appears.

The top of this window contains two text entry boxes labelled *Private Name* and *Public Name*. Enter the private name in the first entry box; the private name must begin with the prefix *_p*. If this prefix is not entered by the user, it will be prepended automatically.

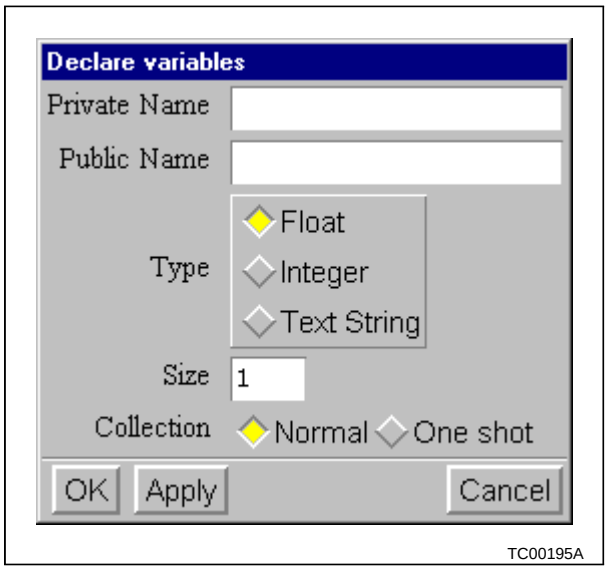


Figure 4-38. Declare Variable Window

Enter the public name in the second entry box. If the model being configured is a graphic, the public name is automatically enclosed in quotation marks.

The *Declare Variable* window is also used to specify how the value should be collected from the global database, once the configured graphic is on display in the operator window. A set of three radio buttons, labeled *Type*, is used to specify the type of value to be collected; the available types include:

- Float (for floating point numbers).
- Integer (for integers).
- Text String (for tagnames, legends, and so on).

Use the type *Float* (the default) if the variable type is numeric but could be either floating point or integer. For the purposes of this example, the C_ASP atom has a floating point value, so type *Float* would be selected.

Below the *Type* selection area is an entry box labelled *Size*, which is used to specify array variables. Entering a value greater than one declares an array the size of the value entered. If the size is one (the default), then the variable is not an array. For the purposes of this example, a size of one would be used.

Below the *Size* entry box is *Collection* area. The selections available here include:

- Normal.
- One shot.

These choices, each of which is accompanied by a radio button, are used to specify how often the value is to be collected from the global database. A normal collection is performed once each second (or every two seconds if the model is in an operator window which does not have keyboard focus). A one shot collection is performed only once, when the graphic is first called up in an operator window. In this example, the value of PIC-1515.C_ASP is to be updated, so the *Normal* collection would be chosen.

Click on *OK* to complete the variable declaration and exit the window; click on *Apply* to complete the variable declaration but leave the window open to declare additional variables. (In the parent *Variables* window (Figure 4-37), click on *Add* or select an existing variable and click on *Edit*; either of these actions will reset the *Declare Variable* window and allow configuration of a new or existing variable).

Once a variable is declared, it will appear in the list box in the *Variables* window (Figure 4-37). To change a variable declaration, select the desired item on this list, then click on *Edit*. The *Declare Variable* window containing that declaration's parameters will be displayed.

To see which objects in a model have DynProps that reference a variable, select the variable by clicking on it in the list box, then click on *References*. A box will be drawn around each object which has a reference to the variable, and the *Variable References* window will appear, displaying the count of such objects.

To delete a variable, click on the appropriate entry in the list box to select it, then click on *Delete*. A variable can be deleted only if it is not referenced by any object in the model; otherwise an error message will appear.

DECLARING VARIABLES IN SUBMODELS

When declaring a variable in a graphic (described above), the public name of the variable must be the name of a point in the global database (i.e., a tag.atom). This is not necessary when declaring a variable in a submodel. Here, the public name can be a generic name. This allows the public name to be assigned to a tag.atom or a constant each time an instance of the submodel is configured on a graphic.

> **NOTE:** Quotation marks should not be placed around the public name if the model being configured is a submodel unless it is the name of a tag.atom. When configuring dynamic objects on a submodel, the type submodel should be specified via the options pull-down menu before attempting to use the *Declare Variables* menu item. If this order is not followed, quotation marks will automatically be placed around the public name and will cause an error.

No change is required to the DynProp; the following version remains a valid example:

(* (stext _psetpoint "%7.1"))

The name _psetpoint would remain the private name for the variable, but the public name would now be configured as some generic name such as setpoint instead of PIC-1515.C_ASP.

When an instance of this submodel is configured on a graphic (using the *Configure Submodel* item), setpoint will appear in the list of variables for the submodel. Assign the name of a tag.atom, or a numeric constant, to the variable.

NAMING VARIABLES

The following naming conventions apply to variable names:

- A variable name can be up to 60 characters in length.
- The first character of a variable's name must be a letter.

Subsequent characters can be letters, numbers, or the underscore character. With few exceptions (described below), no other characters should be used. The dash (-) is especially to be avoided, since that character is interpreted as the subtraction operator.

Exceptions to the above naming conventions are:

- The private name of a variable must begin with _p.

- Pairs of dollar sign ($) characters can be used in the public name of a variable in a submodel, as part of the string substitution feature described below.

When an instance of a submodel is configured on a graphic, supply values for the submodel variables, as described in ***Configure Submodel*** in Section 4. A string substitution feature is available which allows configuration of multiple variables which share a characteristic, such as the same tagname, or the same atom type.

This makes it easy to configure a submodel that will display several different atoms from the same tagname, such as the C_APV, C_ASP and C_AOUTPUT atoms. If the atom names are known, the string substitution feature can be used to declare variables for the submodel in such a way that only the tagname must be supplied when an instance of the submodel is configured.

To use this feature, the part of the variable name that will assume the same value in all cases must be enclosed between '$' characters. In Example 1 below, the following variable

names could be configured for a submodel, using the *Declare Variable*s window:

Example 1:

```
$tag$C_APV
$tag$C_ASP
$tag$C_AOUTPUT
```

Save the submodel, then place an instance of it on a graphic and select the instance. Click on *Configure Submodel* in the *Dynamics* menu. When the *Submodel Configuration* window is displayed, only the string tag will appear as a variable that needs to be configured, rather than a separate entry for each atom. Enter the tagname followed by a period ()., i.e., PIC-1515.. In the submodel, this will result in the replacement of the string tag with the string PIC-1515. in each of the three variables, as follows:

```
$tag$C_APV    =   PIC-1515.C_APV
$tag$C_ASP    =   PIC-1515.C_ASP
$tag$C_AOUTPUT =   PIC-1515.C_AOUTPUT
```

Note that all three variables were configured in the submodel by supplying one value, namely the value of the string for tag.

The user must supply the period (.) after the tagname, otherwise erroneous values such as PIC-1515C_APV would result. While the period (). alone cannot be included in the variable name (i.e., ***$tag.$C_APV*** is not a legal variable name), it can, however, be used when preceded by a backslash character (i.e., \.). Alternatively, the sequence #d can be used in the variable name to represent a period. Both of these methods are shown in Example 2.

Example 2:

```
$tag$\.C_APV     or     $tag$#dC_APV
$tag$\.C_ASP     or     $tag$#dC_ASP
$tag$\.C_AOUTPUT or     $tag$#dC_AOUTPUT
```

If these variable names are used, only the tagname (i.e., PIC-1515) has to be supplied by the user when configuring the submodel.

A submodel could be designed to show the value of the same atom from ten different tagnames. Since the atom name is the same in all ten cases, string substitution could be used in naming the variables, as follows:

```
$tag1$#d$atom$
$tag2$#d$atom$
$tag3$#d$atom$
.
```

.
.
$tag10$#d$atom$

When an instance of this submodel is configured, the *Submodel Configuration* window will contain 11 variables to configure, one named ***atom*** and the ten ***tagn*** variables. If the #d sequence is not used, the period (). should be supplied as part of the atom name. Note that it is necessary to enclose both the ***atom*** and ***tagn*** portions of the variable names between '$' characters, so that both portions of the variable name can be configured.

It may be necessary to include an array index number in a variable name. Like the period, the square brackets ([]) cannot be included directly in variables; they can, however, be used when each is preceded by a backslash character. For example, \[n\] can the used to represent an array index, where n represents the index number. Alternatively, #n can also be used to represent the array index. The following examples shown two possible alternatives to represent the same item:

tag\.C_APV\[3\] or tag#dC_APV#3

The array index number represented by n must be an integer.

NOTE: The index number can, like the tagname, be assigned as a variable, i.e., tag\.C_APV\[$INDEX$\] or tag#dC_APV#$INDEX$

In this case, both *Tag* and *Index* can be assigned values when configuring an instance of the submodel which uses this variable.

It is important to remember that string substitution in variable names, as described in this subsection, occurs at the time an instance of a submodel is configured on a graphic. If the submodel itself is later changed to use additional variables containing the same or other substitute strings, then all instances of that submodel will need to be reconfigured on graphics which use the submodel. (If the only change made to the submodel was to add variable names using the same substitute string (i.e., tag) as used previously in the submodel, then for each graphic select each instance of the submodel, click on *Configure Submodel* in the *Dynamics* pull-down menu, click on *OK* in the submodel configuration window, then install the graphic).

Dynamic Configuration Examples

The following five sections provide practical examples of configuring dynamics for submodels.

EXAMPLE 1 - SINGLE EXPRESSION

Process Conditions: A valve has 1 limit switch (tagname VLV101). VLV101 is closed (VLV101.C_AOUTPUT=1) when the valve is open and is open (VLV101.C_AOUTPUT=0) when the valve is closed.

Objective: Assign dynamics to an instance of submodel ISA_VLVE_ACTR on a graphic so that the submodel instance is filled in white when the valve is open and is unfilled (i.e., filled with black) when the valve is closed.

Procedure: 1. Make an instance of Bailey Symbol ISA_VLVE_ACTR on a graphic.

2. With the Submodel still selected, select the *Configure Attributes* item from the *Dynamics* pull-down menu in the *Graphic Configuration* window menu bar. The *Expression Dynamics* window will appear.

3. Configure the expression and settings from the following table:

Expression	True Settings	False Settings
VLV101.C_AOUTPUT	fill color = 7	fill color = 0

EXAMPLE 2 - MULTIPLE EXPRESSIONS

Process Conditions: A valve has two limit switches (TAGS VLV101 and VLV102). VLV101 is closed (VLV101.C_AOUTPUT=1) when the valve is open and VLV102 is closed (VLV102.C_AOUTPUT=1) when the valve is closed.

Objective: Assign dynamics to an instance of the submodel ISA_VLVE_ACTR on a graphic so that the submodel instance is filled in white when the valve is open, unfilled when the valve is closed, and filled in yellow when the state is unknown (i.e., both limit switches open or closed).

Procedure: 1. Make an instance of Bailey Symbol ISA_VLVE_ACTR on a graphic.

2. With the submodel still selected, select the *Configure Attributes* item from the *Dynamics* pull-down menu in the *Graphic Configuration* window menu bar. The *Expression Dynamics* window will appear.

3. Configure the expressions and settings from the following table:

Number	Expression	True Settings	Falsesettings
1	VLV101.C_AOUTPUT=1 && VLV102.C_AOUTPUT=0	fill color = 7	None

Number	Expression	True Settings	Falsesettings
2	VLV102.C_AOUTPUT=1 && VLV101.C_AOUTPUT=0	fill color = 0	None
3	VLV101.C_AOUTPUT=0 && VLV102.C_AOUTPUT=0	fill color = 3	None
4	VLV101.C_AOUTPUT=1 && VLV102.C_AOUTPUT=1	fill color = 3	None

EXAMPLE 3 - USING THE TRUE EXPRESSION

Process Conditions: None.

Objective: Make a static instance of the submodel ISA_VLVE_ACTR. Set the color of the valve to cyan.

Procedure: 1. Make an instance of Bailey Symbol ISA_VLVE_ACTR on a Graphic.

2. With the submodel still selected, select the *Configure Attributes* item from the *Dynamics* pull-down menu in the *Graphic Configuration* window menu bar. The *Expression Dynamics* window will appear.

3. Configure the expression and settings from the following table

Expression	True Settings	False Settings
TRUE	fill color = 6	none

EXAMPLE 4 - CREATING A NEW SUBMODEL WITH DYNAMICS

Process Conditions: None.

Objective: Create a new submodel (DynamicValve) with the same dynamics as the submodel instance shown in ***EXAMPLE 2 - MULTIPLE EXPRESSIONS*** in this section. Use variable names in place of INFI 90 OPEN database points so that instances of Dynamic Value can be configured for different INFI 90 OPEN database points.

Procedure: 1. From the *File* pull-down menu in the *Graphic Configuration* window menu bar, open the Bailey symbol ISA_VLVE_ACTR. Notice that the *Model Type* changes to *Submodel*.

2. Select the *All* item from the Select pull-down menu (discussed in ***SELECT PULL-DOWN MENU***).

3. Select the *Group* item from the *Edit* pull-down menu (discussed in ***Grouped Objects and Expressions***).

4. Select *Configure Attributes* from the *Dynamics* pull-down menu. In the resulting *Expression Dynamics* window, configure

the expressions and settings from the following table:

Number	Expression	True Settings	False Settings
1	open_switch=1 && closed_switch=0	fill color=7	None
2	closed_switch=1 && open_switch=0	fill color=0	None
3	open_switch=0 && closed_switch=0	fill color=3	None
4	open_switch=1 && closed_switch=1	fill color=3	None

5. Select *Save As* from the *File* pull-down menu and enter the name DynamicValve when prompted for the model name. The new submodel is now placed in the disk location dedicated to storage of off-line custom submodels (i.e., the off-line submodel directory).

6. Select Install Current from the *File* pull-down menu.

NOTE: The Submodel must be installed before it can be used in an installed Graphic.

Refer to Example 5 for the use of this new submodel.

EXAMPLE 5 - CONFIGURING DYNAMIC SUBMODELS

Process Conditions: Same as Example 2

Objective: Configure an instance of the submodel dynamic valve that was built in Example 4.

Procedure: 1. Make an instance of dynamic valve on a graphic (from the list of *Installed Submodels*).

2. With the submodel still selected, select *Configure Submodel* from the *Dynamics* pull-down menu. The *Submodel Configuration* window will appear. Configure the variables as shown in Figure 4-39.

EXAMPLE 6 - SIMULATING MOTION USING THE BLINK ATTRIBUTE

Process Conditions: None; True is used in all expressions in this example.

Objective: Simulate motion in a particular direction.

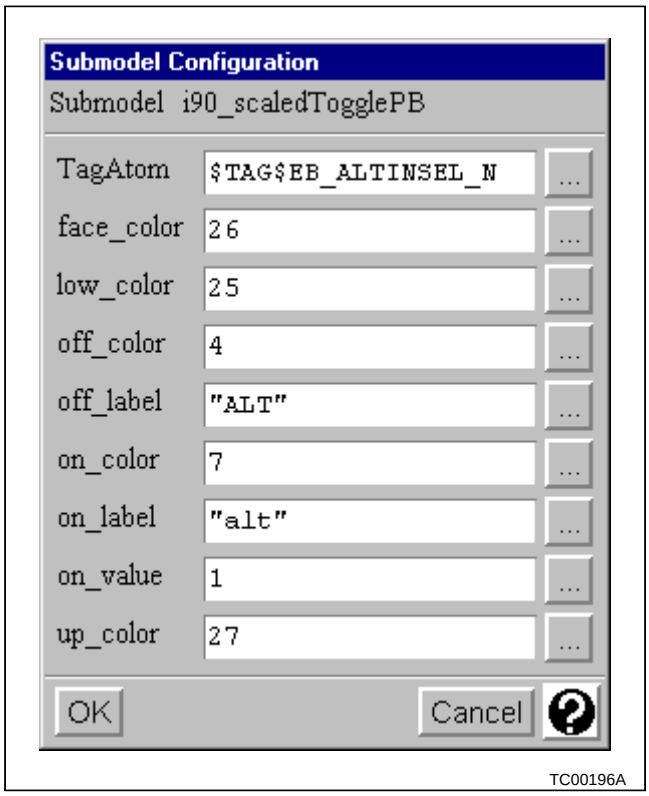


Figure 4-39. Submodel Variable Configuration Window

Procedure:

1. Create a new graphic model; select *New* from the *File* menu and the model type *Graphic* from the *Options* menu, if necessary.

2. Create instances of the items which will be used to emulate motion; for the purposes of this example, create 6 short, consecutive lines. Use the grid setting for the *Snap* option to aid in keeping them neatly in line.

3. Click on the first object to select it, then click on the *Dynamics* item on the *Graphic Configuration* window menu bar. Click on the *Configure* Attributes item in the resulting pull-down menu. The *Expression Dynamics* window will appear.

4. Configure the *Expression* in the *Expression Dynamics* window. For purposes of this example, enter **TRUE**. The same expression should be used for all objects in the group simulating motion.

5. Select the *Blink* option from the *Settings When True* area. The *Blink Configuration* window will appear. Configure the settings for each of the objects using settings listed in the Table 4-7, repeating steps 3 and 4 as necessary.

Table 4-7. Blink Settings

Line Segment Number	Expression	Period	Phase	Blink Color 1	Duration1	Blink Color 2	Duration 2	Steady Color
1	TRUE	1	0	4	1	15	5	15
2	TRUE	1	1	4	1	15	5	15
3	TRUE	1	2	4	1	15	5	15
4	TRUE	1	3	4	1	15	5	15
5	TRUE	1	4	4	1	15	5	15
6	TRUE	1	5	4	1	15	5	15

6. Select *Save* from the *File* pull-down menu and enter a name such as *DynamicLine* when prompted for the model name. The new graphic is now placed in the disk location dedicated to storage of off-line graphics (i.e., the off-line graphics directory).

7. Select *Install Current* from the *File* pull-down menu to install the model as a graphic display.

8. When the graphic installed in Step 7 is displayed in an operator window, it will appear as a gray line with a blue pulse along it.

> **NOTE:** The only setting which differs between the 6 objects is the PHASE setting; each segment is assigned a consecutively higher PHASE number, which means that the blink cycle for that object will begin a tenth of a second after that of its predecessor. Also, the values entered for Duration 1 and Duration 2 added together equal the number of segments in the line. This allows the first pulse to end just before the next begins.

Variations:

1. To make the blue pulse appear to move more slowly, increase the length of the *Period* for all six segments, i.e., change the value entered for this parameter from 1 to a higher number.

2. To have two blue pulses move along the line in this example, decrease the value of *Duration* 2 by changing the value entered from five to two or three.

> **NOTE:** As the user becomes more familiar with the graphics editor, the number of steps necessary to create a graphic such as this one can be decreased. For example, to create the original line with a color pulse, the user might create just the first line segment and configure its dynamics from scratch, then copy it to create the additional 5 segments of the line. It is then only necessary to change the values configured in the Phase parameter for each segment.

EXAMPLE 7 - USING TEXT TO INDICATE CONDITIONS

Process Conditions: Switches on a tank indicating High Level (LI101HL.C_AOUTPUT) and High High Level (LI101HHL.C_AOUTPUT) are used to indicate alarm conditions.

Objective: Display a text message on a graphic to indicate the condition of these switches, using the parameters in Table 4-8:

Table 4-8. Input Conditions

Input Condition	Text	Color
LI101HL.C_AOUTPUT = 0 and LI101HHL.C_AOUTPUT = 0	Normal	Green
LI101HL.C_AOUTPUT = 1 and LI101HHL.C_AOUTPUT = 0	Hi Level	Yellow
LI101HL.C_AOUTPUT = 1 and LI101HHL.C_AOUTPUT = 1	Hi Hi Level	Red
LI101HL.C_AOUTPUT = 0 and LI101HHL.C_AOUTPUT = 1	Unknown	White Flash

Procedure:

1. Put a text object on a Graphic.

2. With the object still selected, click on *Configure Attributes* in the *Dynamics* pull-down menu in the *Graphic Configuration* window menu bar. The *Expression Dynamics* window will appear.

3. Configure the expression and settings from Tables 4-9 and 4-10:

Table 4-9. Expressions and Settings

Number	Expression	True Settings	False Settings
1	LI101HL.C_AOUTPUT == 0 && LI101HHL.C_AOUTPUT == 1	-Blink (See Below)	None
2	LI101HL.C_AOUTPUT == 0 && LI101HHL.C_AOUTPUT == 0	text = Normal text color = 2	None
3	LI101HL.C_AOUTPUT == 1 && LI101HHL.C_AOUTPUT == 0	text = Hi Level text color = 3	None
4	LI101HL.C_AOUTPUT == 1 && LI101HHL.C_AOUTPUT == 1	text = Hi Hi Level text color = 1	None
5	LI101HL.C_AOUTPUT == 0 && LI101HHL.C_AOUTPUT == 1	text = Unknown text color = 7	None

Table 4-10. Blink Settings

Period	Phase	Blink Color 1	Duration 1	Blink Color 2	Duration 2	Steady Color
2	1	7	1	0	1	0

BAILEY SUBMODELS

The assignment of tag database points to these dynamics is accomplished through the *Configure Submodel* menu item on the *Dynamics* pull-down menu. Alternatively, the *Submodel Configuration* windows can also be accessed by positioning the pointer on the submodel and clicking the center mouse button.

> **NOTE:** If a dynamic submodel is copied from one child window to another within the *Graphic Configuration* window, the configuration for the submodel is copied as well.

For a complete description of all the submodels refer to Appendix B.

EXPRESSIONS AS SUBMODEL VARIABLES

As shown in preceding portions of this chapter, values are assigned to submodel variables when a graphic is configured. The values assigned are of two types: static and dynamic. In the Bailey submodels used here as examples, variables which can be assigned a tagname (i.e., TagAtom, TagName, TAG) are dynamic variables. All other variables which can be assigned for the Bailey submodel, such as font, face_color, etc., are static variables.

> **NOTE:** In the case of non-Bailey submodels, dynamic variables are those which have been declared so that they can be assigned tag-names. Refer to ***Create Custom Dynamics and Declare Variables*** in Section 4 for information on creating custom dynamics and declaring variables.

The value assigned to a *static* variable must be a numeric or string constant. For example, the pushbutton submodel ***FP_DYputpb*** in Appendix B has a static variable named button_label. This variable must be assigned a string constant, such as Press Here.

The value assigned to a *dynamic* variable can be a numeric or string constant, but can also be the name of an INFI 90 OPEN tag.atom. For example, the submodel ***FP_DYvalue*** in Appendix B has a dynamic variable named *TagAtom*. If the name of an INFI 90 OPEN tag.atom such as PIC-1515.C_ASP is assigned to this variable, then the value of PIC-1515.C_ASP will appear on the graphic when it is displayed in an operator window.

Additionally, an *expression* can be configured as the value of a dynamic variable. The expression can include:

- Constants.
- INFI 90 OPEN tag.atom names.
- Arithmetic operators.
- Logical operators.

For example, the following expression could be assigned to the *TagAtom* dynamic variable of the ***FP_DYvalue*** submodel. The expression would cause the value of PIC-1515.C_ASP to be converted from degrees Centigrade to Fahrenheit before it is displayed on a graphic:

*9/5 * PIC-1515.C_ASP + 32*

The following rules govern the use of expressions assigned to submodel dynamic variables.

1. An expression can include INFI 90 OPEN tag.atoms, constants, arithmetic and logical operators, and parentheses.

2. The arithmetic operators are:

+	addition
-	subtraction
*	multiplication
/	division
^	exponentiation

3. The logical operators are:

!	NOT
&&	AND
||	OR
==	EQUAL
!=	NOT EQUAL

4. Operators should be surrounded by spaces. For example, the following expression is invalid:

PIC-1515.C_ASP+10

The expression should be written as follows:

PIC-1515.C_ASP + 10

5. The set of rules which determines which operators are performed first is known as precedence, and is described in ***CONFIGURING THE EXPRESSION - OPERATORS AND OPERANDS*** in Section 4 of this instruction. Precedence can be controlled using parentheses. Operator precedence used here follows the precedence used in the C Language (with the exception of the exponentiation operator (^), which is not available in

C). For convenience, the operators available for use in expressions assigned to submodel variables are listed here. Operators grouped together have equal precedence. Higher precedence operator groups are located higher in the list.

( )	Parentheses
!	Logical not
^	Exponentiate
*	Multiply
/	Divide
+	Add
-	Subtract
==	Equal
!=	Not equal
&&	Logical and
| |	Logical or

6. Expressions are limited to 120 alphanumeric characters. For example, the following expression contains 14 characters: (PIC-1515.C_ASP + 10).

7. Syntax errors in expressions are not detected by the graphics editor; they are detected when the graphic is displayed in an operator window.

8. Expressions must be enclosed in quotes. This is done automatically if the model being edited is a graphic, but must be done by the user when a submodel is being edited.

9. Submodel variables cannot be included in expressions. For example, the following expression is invalid:

_px + 10

NOTE: This section describes the assignment of expressions to submodel dynamic variables. This should not be confused with the *Configure* Attributes feature (refer to ***Configure Attributes - Dynamic Attributes for Objects*** in Section 4) of the graphics editor. The *Configure Attributes* feature allows the user to use another form of expression to configure dynamics which affect a limited set of graphic attributes of an object or submodel. The expression used in the *Configure Attributes* feature is governed by a different set of rules. Examples of the form of expression used with *Configure Attributes* feature are provided in ***Dynamic Configuration Examples*** in this section.

PWC SYMBOLS

PWC Symbols are standard static submodels preconfigured and supplied by Bailey. Symbols are objects which represent standard items such as vessels, tanks and storage bins. Symbols which correspond to those defined by the Instrument Society of America (ISA) have the prefix ***ISA_***. NonISA symbols created by Bailey contain the prefix ***FP_ST***. (Dynamic attributes can be assigned to these submodels via the *Configure Dynamic Attributes* menu item).

The following sections contain tables which list the available ISA and Bailey Symbols by category. Each table lists the name of the symbol, followed by a brief description.

Valves and Actuators

Symbol Name	Description	Illustration
FP_STvlv3	3 Way Valve Body	Figure 4-40, Item A
FP_STvlv3actr	3 Way Valve with On-Off Actuator	Figure 4-40, Item B
FP_STvlv3matr	3 Way Valve with Manual Actuator	Figure 4-40, Item C
FP_STvlv3tact	3 Way valve with throttling Actuator	Figure 4-40, Item D
ISA_ACTR	On-Off Actuator	Figure 4-41, Item A
ISA_BVLV	Butterfly Valve	Figure 4-41, Item H
ISA_CVLV	Check Valve	Figure 4-41, Item F
ISA_MATR	Manual Actuator	Figure 4-41, Item B
ISA_RVLV	Relief Valve	Figure 4-41, Item E
ISA_TACT	Throttling Actuator	Figure 4-41, Item C
ISA_VLV3	3 Way Valve Body	Figure 4-41, Item G
ISA_VLV3_ACTR	3 Way Valve with On-Off Actuator	Figure 4-41, Item M
ISA_VLV3_MATR	3 Way Valve with Manual Actuator	Figure 4-41, Item L
ISA_VLVE	Valve Body	Figure 4-41, Item D
ISA_VLVE_ACTR	Valve with On-Off Actuator	Figure 4-41, Item I
ISA_VLVE_MATR	Valve with Manual Actuator	Figure 4-41, Item K
ISA_VLVE_TACT	Valve with throttling Actuator	Figure 4-41, Item J

Containers and Vessels

Symbol Name	Description	Illustration
FP_STcyltank	Cylindrical Tank	Figure 4-40, Item G
ISA_ATNK	Atmospheric Tank	Figure 4-42, Item E
ISA_BINN	Storage Bin	Figure 4-42, Item F
ISA_DTWR	Distillation Tower	Figure 4-42, Item A
ISA_FTNK	Floating Roof Tank	Figure 4-42, Item G
ISA_GHDR	Gas Holder	Figure 4-42, Item H

Symbol Name	Description	Illustration
ISA_JVSL	Jacketed Vessel	Figure 4-42, Item D
ISA_PVSL	Pressure Storage Vessel	Figure 4-42, Item I
ISA_RCTR	Reactor	Figure 4-42, Item B
ISA_VSSL	Vessel	Figure 4-42, Item C
ISA_WHPR	Weigh Hopper	Figure 4-42, Item J

Rotating Equipment (Pumps and Blowers)

Symbol Name	Description	Illustration
ISA_BLWR_LF	Blower - Outlet on Left	Figure 4-43, Item B
ISA_BLWR_RT	Blower - Outlet on Right	Figure 4-43, Item A
ISA_CMPR	Compressor	Figure 4-43, Item F
ISA_PUMP_LF	Pump - Outlet on Left	Figure 4-43, Item C
ISA_PUMP_RT	Pump - Outlet on Right	Figure 4-43, Item D
ISA_TURB	Turbine	Figure 4-43, Item E

Reciprocating Equipment

Symbol Name	Description	Illustration
ISA_RECP	Reciprocating Compressor	Figure 4-43, Item G

Mixing Equipment

Symbol Name	Description	Illustration
ISA_AGIT	Agitator	Figure 4-44, Item E
ISA_AGIT_MOTR	Agitator with Motor	Figure 4-44, Item F
ISA_IMIX	Inline Mixer	Figure 4-44, Item G

Material Handling Equipment

Symbol Name	Description	Illustration
ISA_CNVR	Conveyor	Figure 4-44, Item A
ISA_MILL	Mill	Figure 4-44, Item D
ISA_RFDR	Rotary Feeder	Figure 4-44, Item B
ISA_RSTD	Roll Stand	Figure 4-44, Item C
ISA_SCNV	Screw Conveyor	Figure 4-44, Item H

Electrical Equipment

Symbol Name	Description	Illustration
ISA_CBRK	Circuit Breaker	Figure 4-45, Item A
ISA_DLTA	Delta Connection	Figure 4-45, Item B

Symbol Name	Description	Illustration
ISA_FUSE	Fuse	Figure 4-45, Item C
ISA_MCTR	Manual Contactor	Figure 4-45, Item D
ISA_MOTR	Motor	Figure 4-45, Item E
ISA_MOTR_BASE	Motor with Base	Figure 4-45, Item F
ISA_STAT	State Indicator	Figure 4-45, Item G
ISA_WYEC	WYE Connection	Figure 4-45, Item I
ISA_XFMR	Transformer	Figure 4-45, Item H

Filters

Symbol Name	Description	Illustration
ISA_LFLT	Liquid Filter	Figure 4-46, Item A
ISA_VFLT	Vacuum Filter	Figure 4-46, Item D

Scrubbers and Precipitators

Symbol Name	Description	Illustration
ISA_EPCP	Electrostatic Precipitator	Figure 4-46, Item B
ISA_SCBR	Scrubber	Figure 4-46, Item C

Separators

Symbol Name	Description	Illustration
ISA_CSEP	Cyclone Separator	Figure 4-46, Item E
ISA_RSEP	Rotary Separator	Figure 4-46, Item F
ISA_SDRY	Spray Dryer	Figure 4-46, Item G

Heat Transfer Devices

Symbol Name	Description	Illustration
ISA_FAXR	Forced Air Exchanger	Figure 4-47, Item D
ISA_FURN	Furnace	Figure 4-47, Item C
ISA_KILN	Rotary Kiln	Figure 4-47, Item E
ISA_XCHG1	Exchanger	Figure 4-47, Item A
ISA_XCHG2	Exchanger	Figure 4-47, Item B

HVAC Equipment

Symbol Name	Description	Illustration
ISA_CTWR	Cooling Tower	Figure 4-47, Item F
ISA_EVPR	Evaporator	Figure 4-47, Item G
ISA_FNXR	Finned Exchanger	Figure 4-47, Item H

Miscellaneous

Symbol Name	Description	Illustration
FP_STfillarrow	Filled Arrowhead	Figure 4-40, Item E
FP_STrtarrow	Right Arrow	Figure 4-40, Item F

ymbol Illustrations

The illustrations in this section provide examples of the Bailey supplied and ISA symbols described in Sections ***Valves and Actuators*** through ***Miscellaneous***.

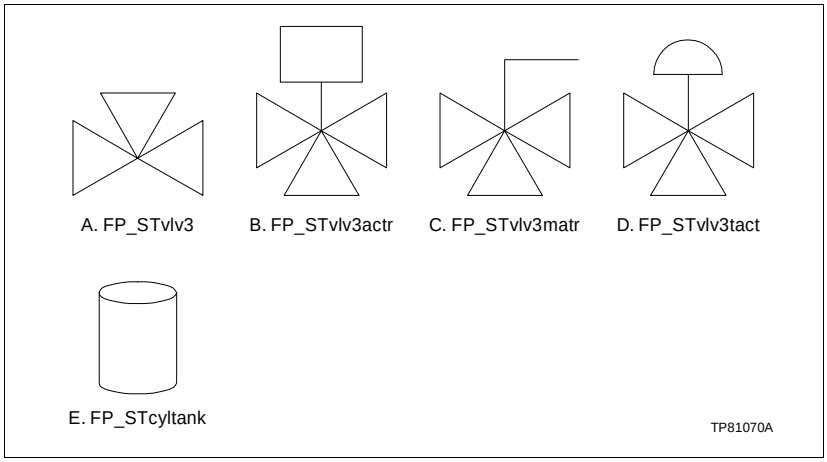


Figure 4-40. Bailey Symbols

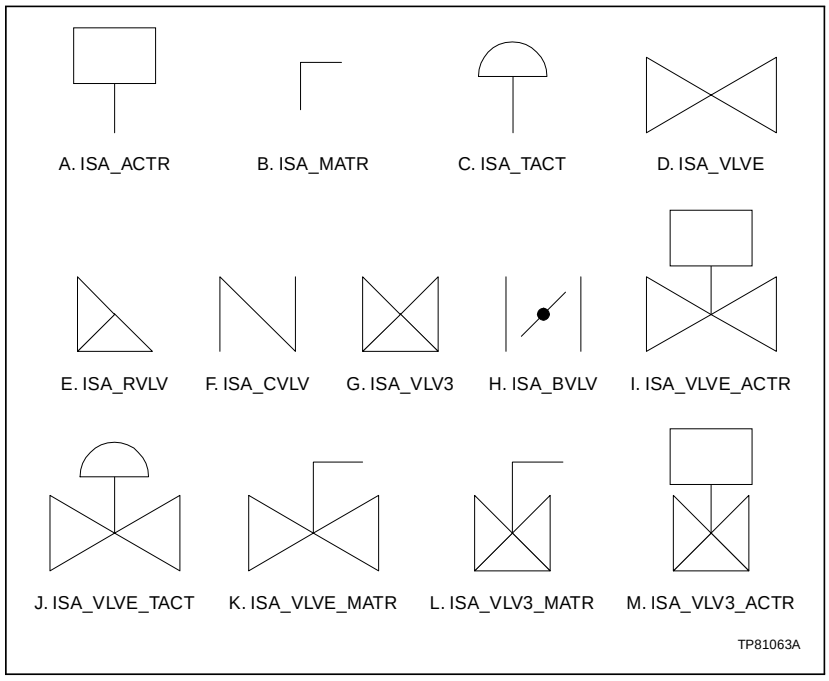


Figure 4-41. ISA Standard Symbols - Valves and Actuators

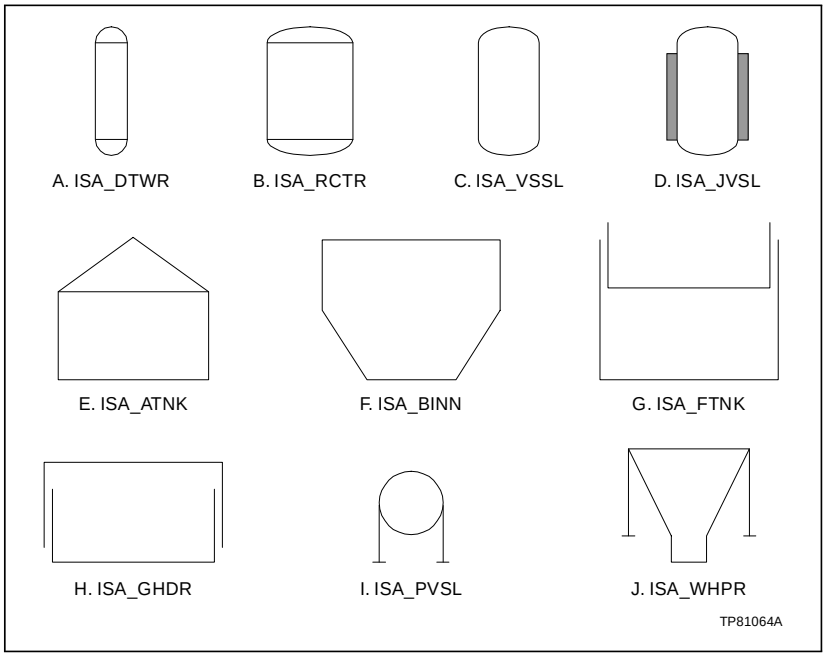


Figure 4-42. ISA Standard Symbols - Containers and Vessels

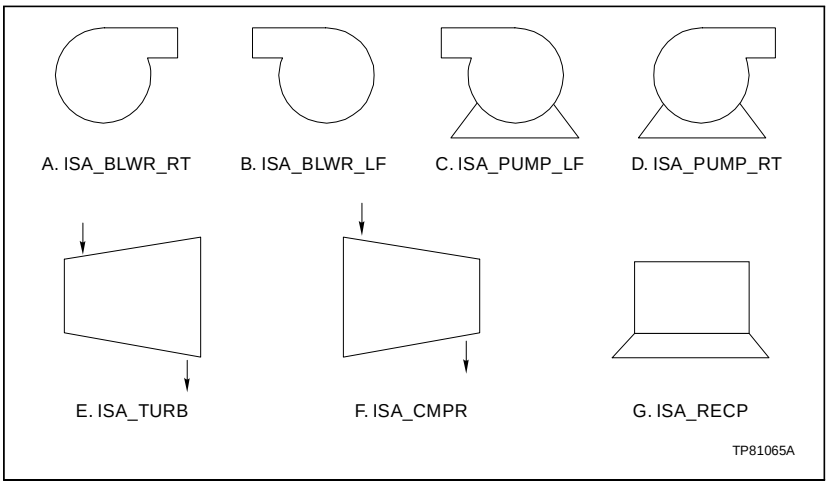


Figure 4-43. ISA Standard Symbols - Rotating and Reciprocating Equipment

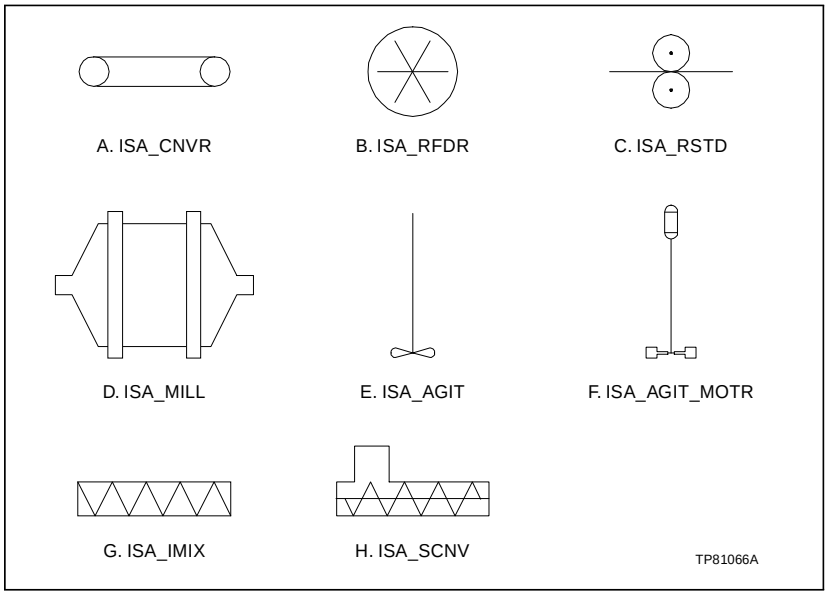


Figure 4-44. ISA Standard Symbols - Mixing and Material Handling Equipment

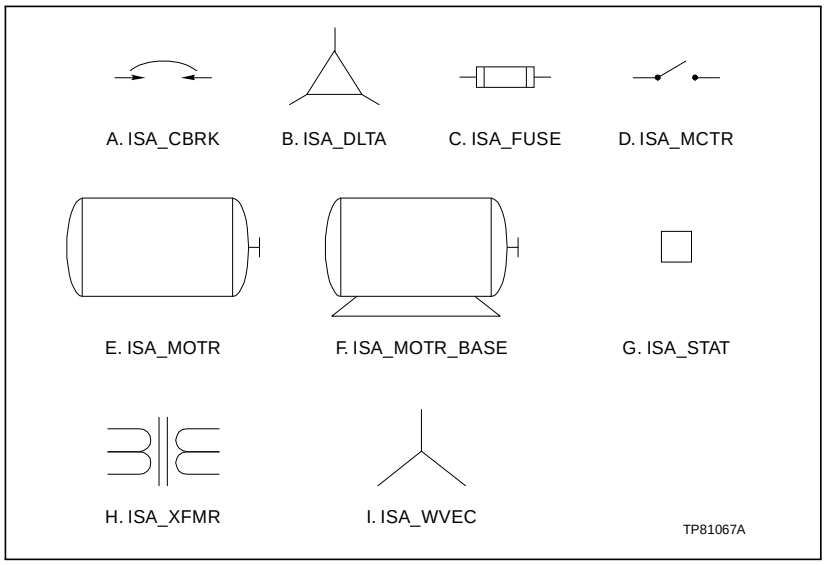


Figure 4-45. ISA Standard Symbols - Electrical Equipment

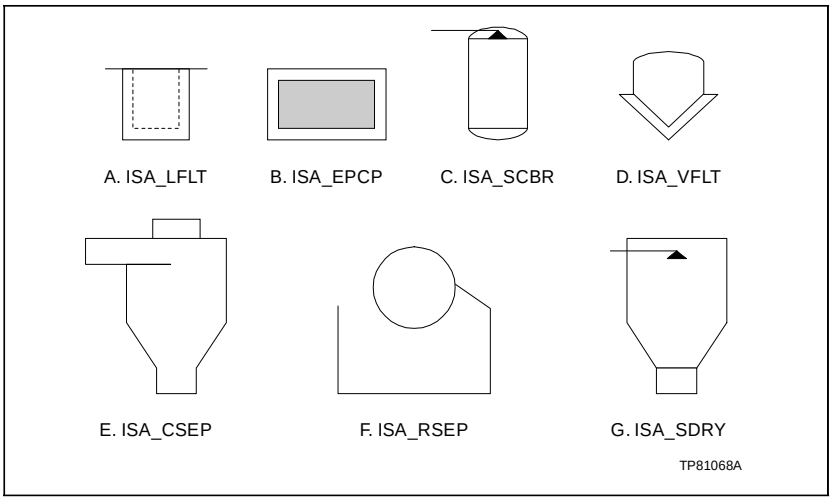


Figure 4-46. ISA Standard Symbols - Filters, Scrubbers, Precipitators and Separators

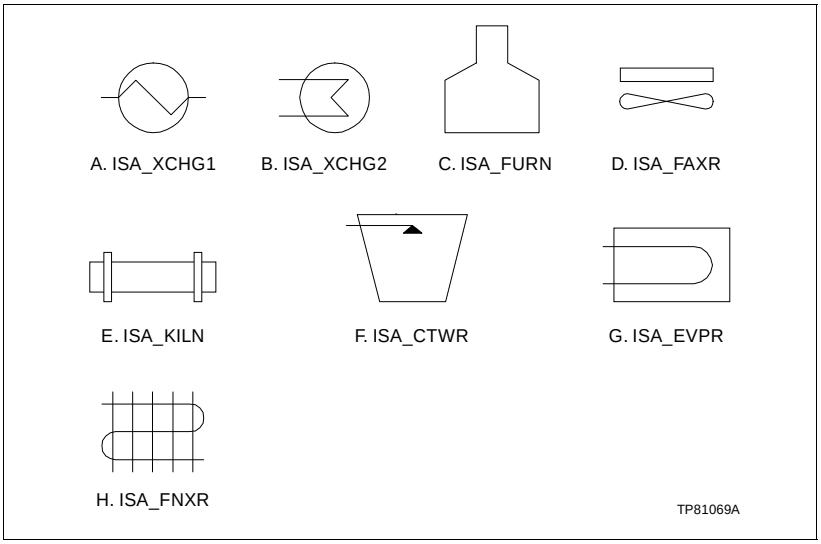


Figure 4-47. ISA Standard Symbols - Heat Transfer Devices and HVAC Equipment

INFI 90 OPEN FACEPLATE SUBMODELS

INFI 90 OPEN Faceplate Submodels are submodels created by Bailey which provide access to the same submodels used to create point and group (Hierarchical) displays on the PWC90 console. If these submodels are being used in a custom process graphic, the tagname should be followed by a period. Refer to Table 4-11 for a list of faceplate submodels.

Table 4-11. I90 Faceplate Submodels

Tag Type	Model For Point	Model For Group
Analog	tuneANALOG1	ANALOG1
Digital	tuneBOOLEAN1	BOOLEAN1
Station	tuneDCSFULL2	DCSFULL2
Remote Control Memory (RCM)	tuneRCM1	RCM1
Remote Manual Set Constant (RMSC)	tuneRMSC1	RMSC1
Device Driver (DD)	tuneDEVDR1	DEVDR1
Multi-State Device Driver (MSDD)	tuneMSDEVDR1	MSDEVDR1
Remote Motor Control Block (RMCB)	tuneRMCB1	RMCB1
Text Selector	tuneTEXT1	TEXT1
Text String	tuneTEXTSTR1	TEXTSTR1
Analog Export	tuneANGRPT1	ANGRPT1
Digital Report	tuneDIGRPT1	DIGRPT1
Module Status	tuneMODULE1	MODULE1
Data Acquisition Analog (DANG)	tuneDNGFULL1	DNGFULL1
Data Acquisition Digital (DADIG)	tuneDADIG1	DADIG1
INFI 90 OPEN Server	tuneSERVER1	SERVER1

Table 4-11. I90 Faceplate Submodels (continued)

Tag Type	Model For Point	Model For Group
Analog (Annunciator Faceplate)		ANCREAL1
Digital (Annunciator Faceplate)		ANCBOOL1

CUSTOM MODELS FOR POINT AND GROUP DISPLAYS

Custom models can be created for use as point and group displays. This section presents guidelines which should be followed when creating custom models intended for use as point or group displays.

- Configure the model as a submodel, using the instructions and data provided in the preceding sections of this chapter.
- Use a generic name for the public name when declaring a variable for the tagname to be used in the submodel; this allows the public name to be assigned to a tagname each time an instance of the submodel is configured on a graphic. Refer to ***DECLARING VARIABLES IN SUBMODELS*** in Section 4 and ***NAMING VARIABLES*** in Section 4 for details on declaring and naming variables in submodels. Note that when naming the generic tagname variable, the string TAG should be used, and upper case letters should be used.
- Install the model as a submodel; installation of graphics and submodels is described in ***INSTALLED AND OFF-LINE GRAPHIC AND SUBMODEL FILES*** in Section 4 and ***Applying Submodel Changes - Reload*** in Section 4.
- Remember that group models should be configured in the appropriate shape, so that they fit correctly on a group display which also contains models for other points.

INFI 90 OPEN Faceplate submodels, supplied by Bailey, can be used as examples when creating new point and group displays. Table 4-11 in provides lists of model names of these INFI 90 OPEN Faceplate submodels for use as point and group displays.

Group Display Customization Techniques

Techniques which exist for customizing group displays on the PWC90 console include the following:

1. Use the graphics editor to modify a copy of a Bailey supplied group model (i.e., submodel ***DCSFULL2***) and install it as a user submodel with the same name. The PWC90 console software checks the disk location which contains the user submodels before it checks the location which contains the PWC

submodels. Therefore, if both a user submodel and an PWC submodel with the same name exist, the user submodel will be found first and will be used on PWC90 group displays.

2. Use the graphics editor to modify copies of one or more of the submodels which compose a Bailey group model. Each group model is composed of several submodels. For example, the model ***DCSFULL2*** uses the submodel ***FP_DYvslider*** as the slider used to change the setpoint on a controller module group display. The submodel ***FP_DYvslider*** could be loaded into the graphics editor, then modified, then installed as a user submodel with the same name. The modified version would then be used for all Station Tags modules and all other modules which have group and point display models that incorporate ***FP_DYvslider***.

This technique allows a group display to be customized without making modifications directly to a copy of the model for the group display.

LOOP AND NODE STATUS DISPLAYS

Module Status tags (i90_ModuleStatus in PWC90 or N90STA on other INFI90 consoles) must be configured for every node on an INFI90 loop and for every module in an INFI90 process node.

The assignment of loop numbers, node numbers, and module addresses can be arbitrary as long as the following rules are followed:

- INFI90 loops are numbered from 1 to 250. The main loop in an INFI90 system is Loop 1. All other loops are satellites of Loop 1. In a Plant Loop system, the only loop is Loop 0.
- INFI90 nodes are numbered from 1 to 250. Plant Loop nodes are numbered from 1 to 63.
- Module addresses are numbered from 0 to 31. Process nodes have a master module (NPM, SBM, or BIM) which is address 0. If using redundancy, the redundant master module is address 1. Process modules (COM, LMM, AMM, MFC, or MFP) are usually given addresses greater than 2.
- An ICI, CIU, or SEM has only one module whose address is 2.
- A bridge node (IIT) has only one module whose address is 0.

EXAMPLE CONFIGURATION

A factory has two process loops joined by a bridge. The main loop (Loop 1) has four consoles nodes (100, 101, 102, and 103),

three process nodes (10, 20, and 40), and a bridge node (14) to a second loop (Loop 25). The second loop has one console node (200), one process node (30), and a bridge node (18). Assume that Node 10 on Loop 1 has process modules 3, 5, and 7. Assume that Node 20 on Loop 1 has process modules 8 and 12. Assume that Node 40 on Loop 1 has process modules 16, 22, and 24. And assume that Node 30 on Loop 25 has process module 9.

The following module status tags might be created:

ConsoleA	Loop 1, Node 100, Module 2
ConsoleB	Loop 1, Node 101, Module 2
ConsoleC	Loop 1, Node 102, Module 2
ConsoleD	Loop 1, Node 103, Module 2
ConsoleE	Loop 25, Node 200, Module 2
ProcessNodeA	Loop 1, Node 10, Module 0
ProcessNodeB	Loop 1, Node 20, Module 0
ProcessNodeC	Loop 25, Node 30, Module 0
ProcessNodeD	Loop 1, Node 40, Module 0
BridgeA	Loop 1, Node 14, Module 0 (actually first node of the bridge)
BridgeB	Loop 25, Node 18, Module 0 (actually second node of the bridge)
ProcessModuleA	Loop 1, Node 10, Module 3
ProcessModuleB	Loop 1, Node 10, Module 5
ProcessModuleC	Loop 1, Node 10, Module 7
ProcessModuleD	Loop 1, Node 20, Module 8
ProcessModuleE	Loop 1, Node 20, Module 12
ProcessModuleF	Loop 25, Node 30, Module 9
ProcessModuleG	Loop 1, Node 40, Module 16
ProcessModuleH	Loop 1, Node 40, Module 22
ProcessModuleJ	Loop 1, Node 40, Module 24

CREATING SUMMARY GRAPHICS

A loop status graphic (Figure 4-48) should be created for every process loop, to show the status of all of that loop's nodes. A node status graphic (Figure 4-49) should be created from every process node to show the status of all of that node's process modules.

Access the graphics configuration application. Open the configuration graphic ***i90_ProcessNodeStatus*** and save with the name of the first process node (i.e., ***First-Process-Node***). Configure the submodel for the master module (at top of graphic) with the name of the process node's module status tag (i.e., ***ProcessNodeA***). Configure the process module submodels (***i90_ProcessModule***) with the names of that node's process module status tags (i.e., ***ProcessModuleA***, ***ProcessModuleB***, and ***ProcessModuleC***). Copy process module submodels if more are needed than are shown in the graphic. Delete any

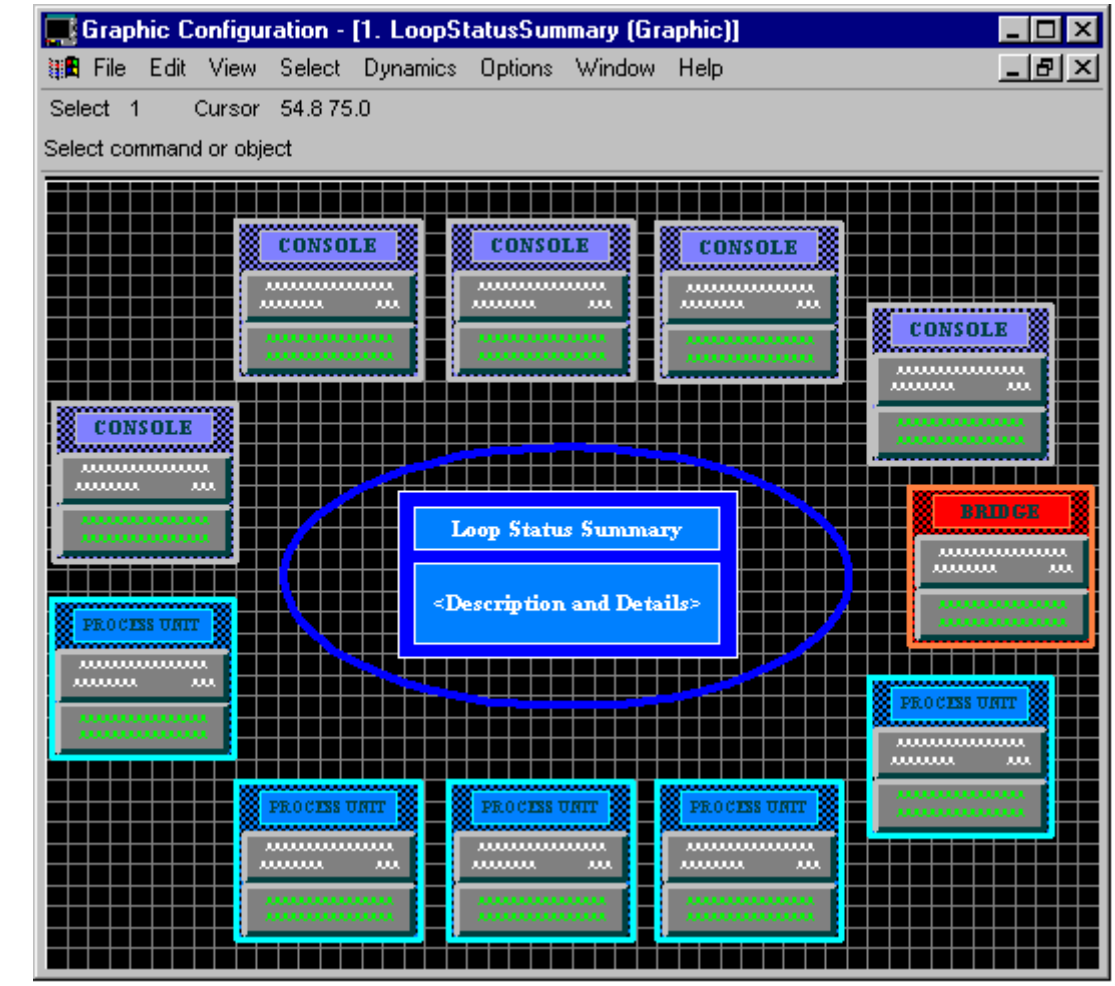


Figure 4-48. LoopStatusSummary Graphic

unused process module submodels. Install this process node graphic. Repeat for all the other process nodes.

After creating graphics for all of the process nodes, create graphics for the process loops by opening the configuration graphic ***i90_LoopStatus***, saving it with a new name (i.e., ***Main-INFI90-Loop***), editing the node submodels (***i90_Console***, ***i90_ProcessNode***, and ***i90_Bridge***), and installing the edited graphic. Each node submodel requires the name of a module status tag. An ***i90_ProcessNode*** submodel must also have the name of a node status graphic. An ***i90_Bridge*** submodel must also have the name of a loop status graphic.

The ***i90_ProcessModule***, ***i90_Console***, ***i90_ProcessNode***, and ***i90_Bridge*** submodels appear similar. Each has an icon (representing module, console, process node, and bridge respectively) plus two buttons (joined together but colored differently). The first button shows the tag's name (i.e., *Process-*

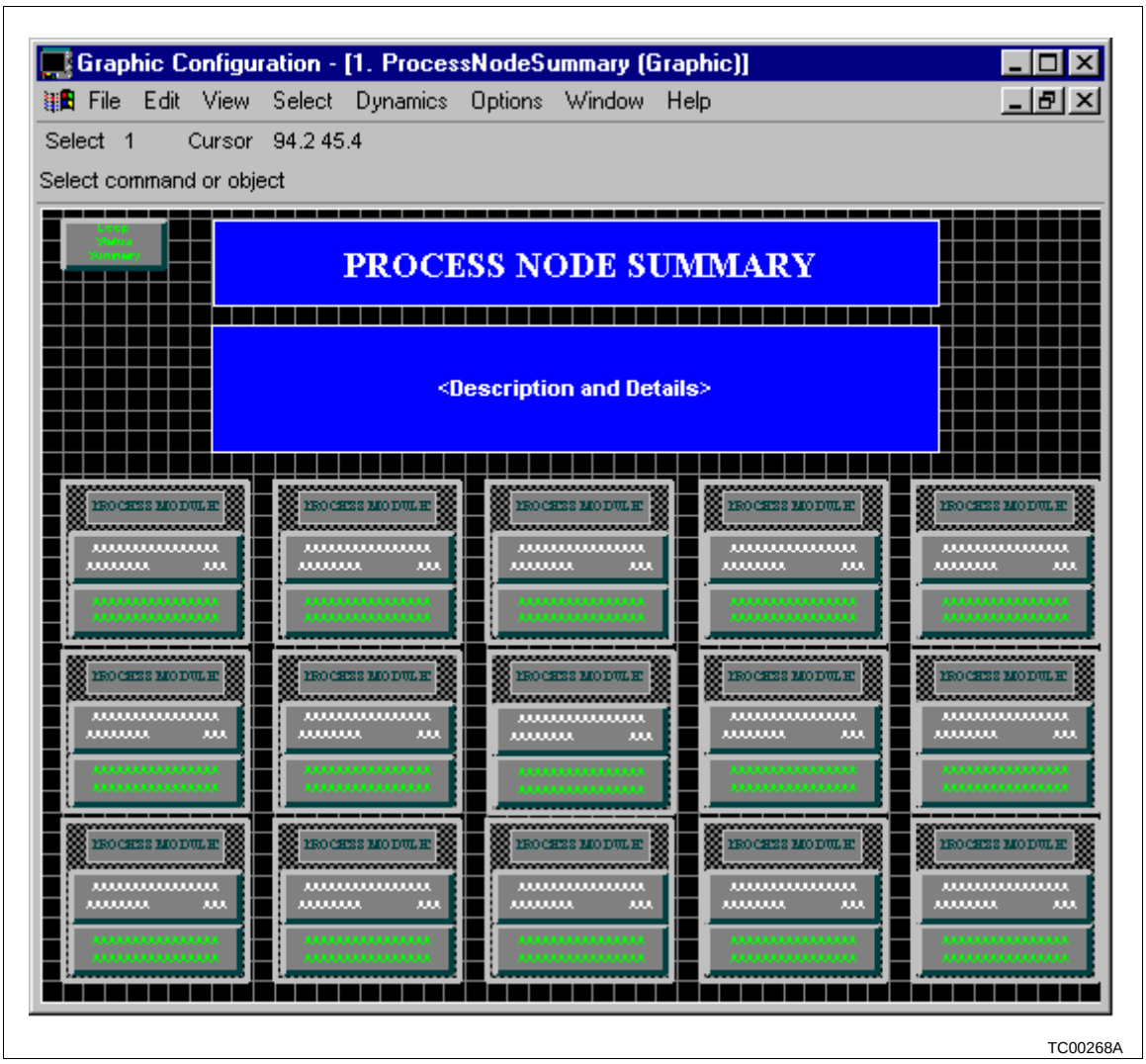


Figure 4-49. ProcessNodeSummary Graphic

ModuleA), module or node type (i.e., *IMMFP02*), and address (module address for ***i90_ProcessModule***, node numbers for ***i90_Console***, ***i90_ProcessNode***, and ***i90_Bridge***). The second button shows the current status (i.e., *Execute*). If the user clicks on the second button, the module details application will be launched). If the user clicks on the first button in an ***i90_ProcessModule*** or ***i90_Console*** submodel, a module status faceplate will pop up. If the user clicks on the first button in an ***i90_ProcessNode*** or ***i90_Bridge*** submodel, the appropriate node status or loop status display will be appear in the operator window.

i90_Console (Using i90_NodeSummPB, i90_NodeErrSummPB)

Atoms used:

- EB_TAG = module status tag's name.

- EB_MODTYP = gives the module type.
- EB_GRPDSPL = gives the name of the faceplate graphic to launch when button is pushed.
- EB_NODE = gives the node address of the module.
- EB_NOFFLINE = indicates if the module is on-line or off-line.
- EB_MODERR = indicates if an errors are present in the module.

User configurable variables:

- Tagname = module status tag's name

i90_ProcessModule (Using i90_ModuleSummPB, i90_NodeErrSummPB)

Atoms used:

- EB_TAG = module status tag's name.
- EB_MODTYP = gives the module type.
- EB_GRPDSPL = gives the name of the faceplate graphic to launch when button is pushed.
- EB_MODULE = gives the module address.
- EB_NOFFLINE = indicates if the module is on-line or off-line.
- EB_MODERR = indicates if an errors are present in the module.

User-configurable variables:

- Tagname = module status tag's name

i90_ProcessNode (Using i90_NodeVectorSummPB, i90_NodeErrSummPB)

Atoms used:

- EB_TAG = module status tag's name
- EB_MODTYP = gives the module type.
- EB_GRPDSPL = gives the name of the faceplate graphic to launch when button is pushed.
- EB_NODE = gives the node address of the module.
- EB_NOFFLINE = indicates if the module is on-line or off-line.
- EB_MODERR = indicates if an errors are present in the module.

User-configurable variables:

- Tagname = module status tag's name.

- Index = index of the process graphic to vector to. String must be entered: **graphic**:*x* where *x* is the graphic index. Index should match the graphic index assigned to the loop status graphic.

i90_Bridge (Using i90_NodeVectorSummPB, i90_NodeErrSummPB)

Atoms used:

- EB_TAG = module status tag's name.
- EB_MODTYP = gives the module type.
- EB_GRPDSPL = gives the name of the faceplate graphic to launch when button is pushed.
- EB_NODE = gives the node address of the module.
- EB_NOFFLINE = indicates if the module is on-line or off-line.
- EB_MODERR = indicates if an errors are present in the module.

User-configurable variables:

- Tagname = module status tag's name.

- Index = index of the process graphic to vector to. String must be entered: **graphic**:*x* where *x* is the graphic index. Index should match the graphic index assigned to the Loop Status graphic.

  NOTES:
  1. **i90_NodeSummPB** launches a faceplate for the tag (NODE address displayed on button).

  2. **i90_NodeVectorSummPB** vectors to the process graphic specified during configuration (process node summary).

  3. **i90_NodeErrSummPB** launches an instance of module details for the tag.

  4. **i90_ModuleSummPB** launches a faceplate for the tag (MODULE address displayed on button).

The generic configuration display ***i90_LoopStatus*** will have ten ***i90_Console*** submodels, ten ***i90_ProcessNode*** submodels, and one ***i90_Bridge*** submodel to be configured. The engineer will delete any unused submodels or copy submodels if more are required. The generic configuration display ***i90_ProcessNodeStatus*** will have room for the master module's ***i90_ProcessModule*** submodel and ten additional ***i90_ProcessModule*** submodels. Again, the engineer will delete any unused submodels or copy submodels if more are required.

The ***i90_Console*** and ***i90_ProcessModule*** submodels expect only a tagname to be configured. The ***i90_ProcessNode*** and ***i90_Bridge*** submodels expect both a tagname and a graphic index to be configured.

SECTION 5 - HISTORICAL DATABASE

INTRODUCTION

The PWC90 console can be configured to collect information from the INFI 90 OPEN Server and store it for use by various applications such as logging, trending and data archiving. This collected data is stored in the historical database.

Historical database configuration is used to create and edit historical database tagnames; each historical database tagname is assigned to a point in a INFI 90 OPEN server real-time tag database. Parameters such as collection rate, storage rate and retention span time are also defined via historical database configuration. Up to 2,500 historical database points can be collected per node (PWC90); using synchronized data collection, up to 5,000 historical database points can be collected network-wide.

Access historical database configuration using the pull-down menus available from the system window *Menu* button. Click on *Menu*; from the resulting pull-down menu, click on the *Configure* option, then click on *Historical Database* in the resulting cascading menu. The *Historical Database Configuration* window (Figure 5-1) will be displayed. The top area of this window contains a scrollable list of all the points which have been configured in the historical database. Below this list are buttons which are used to add, locate, edit and delete points in the historical database.

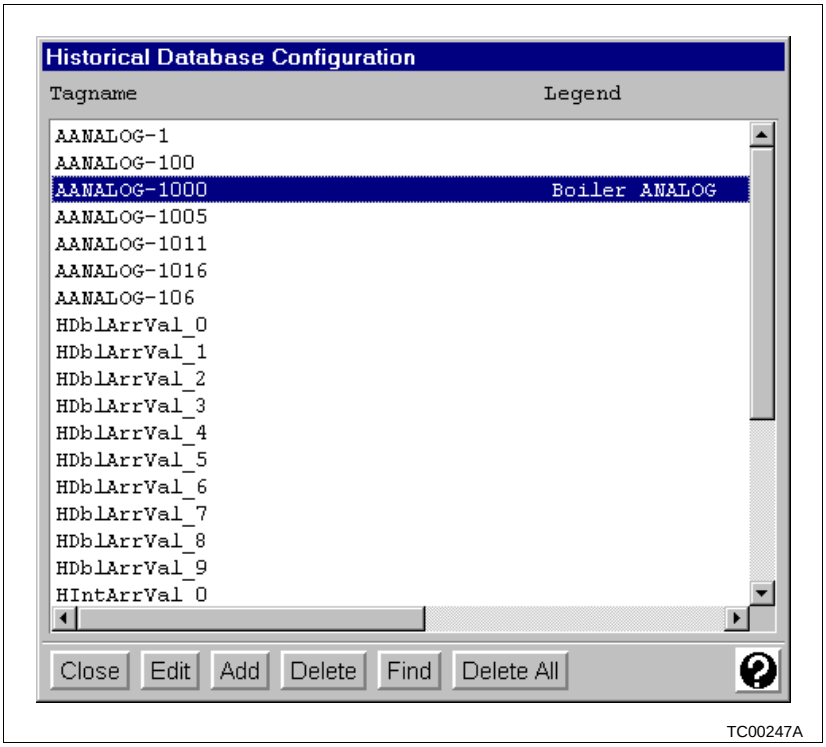


Figure 5-1. Historical Database Configuration Window

EDIT HISTORICAL DATABASE ITEM

To make changes to an existing point in the historical database, first click on the desired point, and then click on *Edit*. The *Historical Tag Configuration* window (Figure 5-2) will appear, containing the current parameters configured for the tag. The window is divided into three main categories:

- Point information.
- Collection information.
- Storage information.

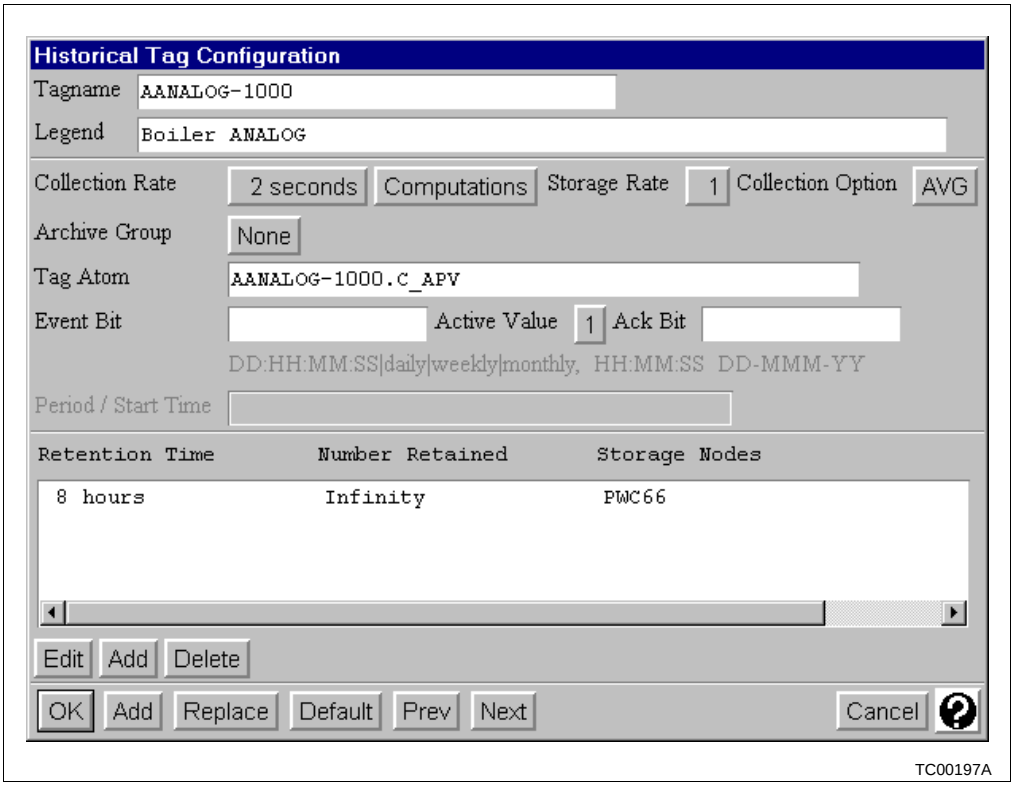


Figure 5-2. Historical Tag Configuration Window

Point Information

The portion of the display devoted to point information contains two entry boxes, labelled *Tagname* and *Legend*.

HISTORICAL DATABASE TAGNAME

Click on the *Tagname* entry box to enter the tagname which is used to identify the point in the historical database. This tagname can be up to 37 characters in length, and cannot contain spaces. The historical database tagname must also be unique on the network.

> **NOTE:** The historical database tagname is separate and different from the INFI 90 OPEN tagname. If data is to be gathered from several different atoms belonging to the same INFI 90 OPEN tag, a separate historical database tagname must be configured for each one.

If no entry is made in the *Tagname* entry box, the historical database tagname will default to the INFI 90 OPEN TAG.ATOM entered in the *Collection Information* portion of the *Historical Tag Configuration* window (described in ***COLLECTION OPTION*** in Section 5).

HISTORICAL DATABASE POINT LEGEND

Click on the *Legend* entry box to enter the string which is used as the legend for the historical database point under configuration. This legend can be up to 63 characters in length and is used in the *Trend Info* window (which is accessed via the *Info* button on a trend display). If no Legend is configured in the historical database for a point, the string configured in the Tag Description atom will be used instead.

Collection Information

The portion of the *Historical Tag Configuration* window devoted to collection information is used to configure the parameters required to collect and store data gathered from a point in a tag database.

COLLECTION RATE

The *Collection Rate* determines the interval between collections, the frequency with which data is gathered from an associated point in the INFI 90 OPEN server real-time tag database. The default collection rate is one minute. Click on the button which displays the collection rate; the *Collection Rates* window (Figure 5-3) will appear. Valid collection rates are shown in Figure 5-3 and range from one second to one day, plus three additional options: *None, Manual* and *Custom.* (The window in this illustration has been resized to show all available collection rates; on the PWC90 console, use the scrollbar or resize the window to view collection rates which are not immediately visible in the list box portion of the window).

Select the desired collection rate (or option) by positioning the pointer on it and clicking the left mouse button. The selected item will be highlighted. Click on *OK* to complete the operation, or click on *Cancel* to exit without changing the current collection rate.

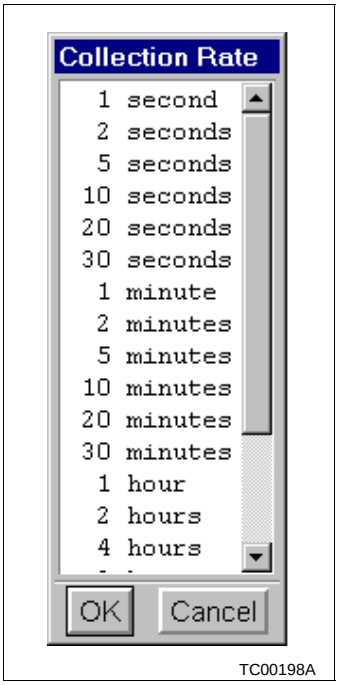


Figure 5-3. Collection Rates Window

Collection Rate Option None

Use the *None* option when configuring a historical database tag for a point which is to be collected based on an event or state, rather than a fixed time interval.

Collection Rate Option Manual

Use the *Manual* option when configuring historical database points which are to contain manually entered data, such as laboratory data. If the *Manual* option is selected, the label *(Manual)* will appear on the face of the *Collection Rate* button, and the collection information section of the *Historical Tag Configuration* window will be revised (Figure 5-4) so that only the *Collection Rate* and *Archive Group* buttons from the original display remain available, and a new button labelled *Data Type* replaces the *Storage Rate* and *Collection Option* buttons.

The data type is set to *Numerical* by default, and the label *Numerical* appears on the face of the *Data Type* button. Click on *Data Type* to display the *Data Type* window (Figure 5-5); a list box in this window contains a list of the available data types.

Available data types are:

- Numerical.
- Alphanumerical (32 characters).
- Alphanumerical (64 characters).

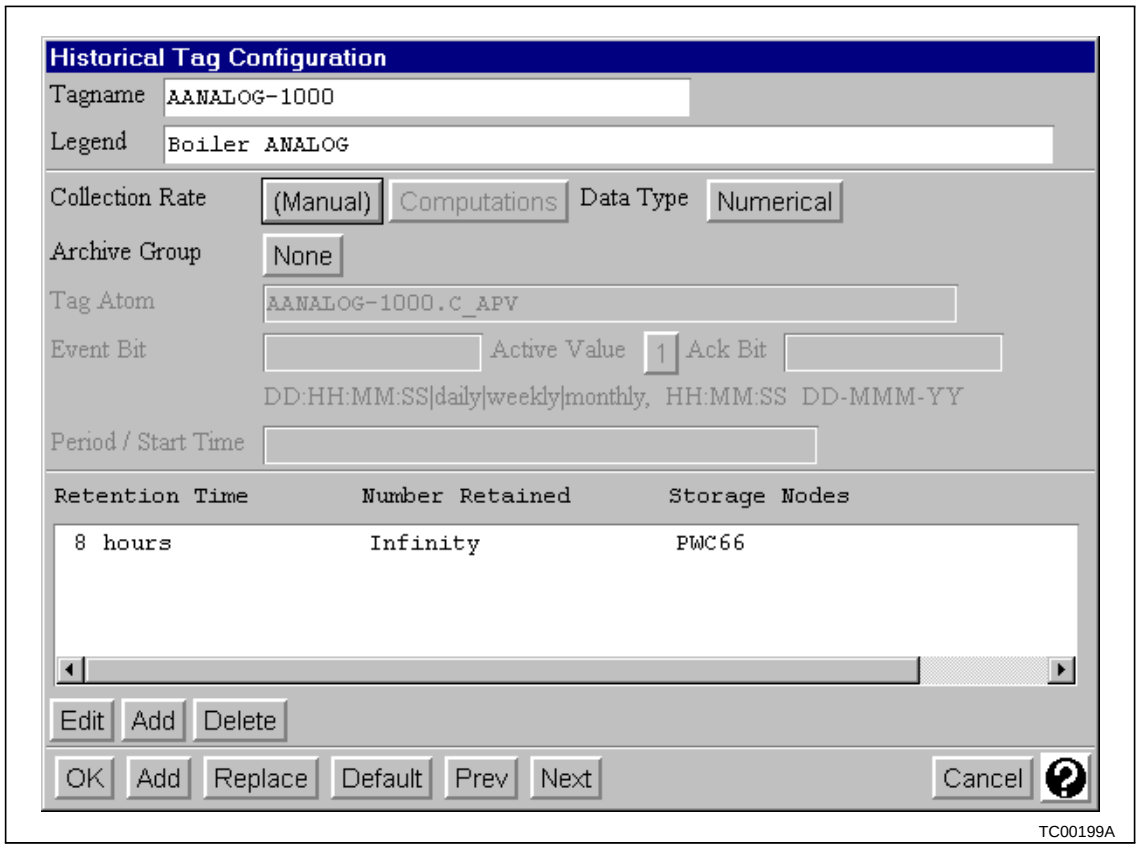


Figure 5-4. Historical Tag Configuration Window for Manual Data Point Configuration

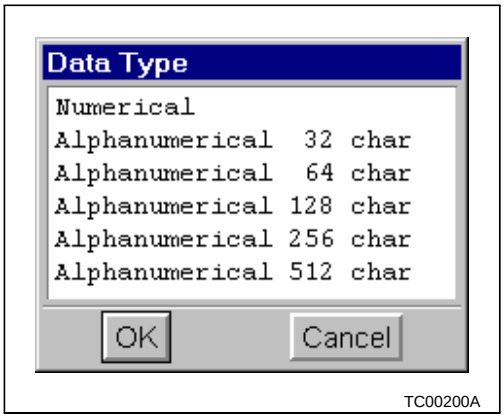


Figure 5-5. Data Type Window

- Alphanumerical (128 characters).
- Alphanumerical (256 characters).
- Alphanumerical (512 characters).

Click on the desired data type in the list box; it will be highlighted.

Click on *OK* to complete the operation, or click on *Cancel* to exit the *Data Types* window without changing the current selection.

Collection Rate Option Custom

The *Custom* collection rate option is used when a user-configured collection rate is desired.

When the *Custom* option is selected, the label *(Custom)* will appear on the face of the *Collection Rate* button, and the collection information portion of the *Historical Tag Configuration* window will be revised (Figure 5-6) so that the entry box in which the period and start time are defined is available.

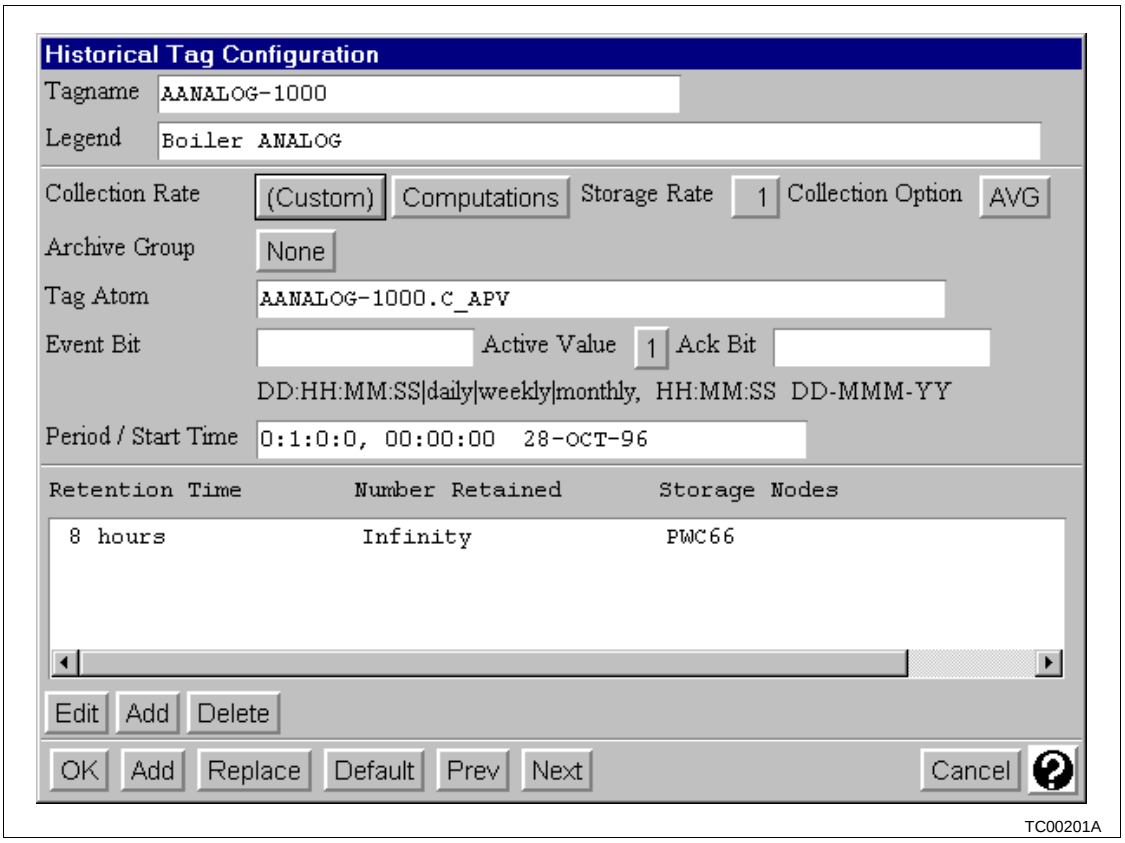


Figure 5-6. Historical Tag Configuration Window for Custom Collection Rate Configuration

Both the *Period* (user-defined collection interval) and the *Start Time* (which defines the beginning of the *Period*) are entered in the same entry box. The *Period* is entered first; the words daily, weekly and monthly are valid entries to define the *Period*. Alternatively, the user can define a customized *Period* by defining the number of days, hours, minutes and seconds in the *Period* using the following format:

DD:HH:MM:SS

Enter a comma (,) after the *Period* to separate it from the *Start*[ing] *Time*. After the *Period* has been defined, the *Start*[Ing] *Time* (hours, minutes, seconds) and date (day, month, year) which marks the beginning of the interval, is entered in the following format:

HH:MM:SS DD:MMM:YY

The default *Period* is hourly (0:1:0:0) and the default *Start Time* is midnight of the current day (i.e., 00:00:00 30-JUN-96). Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name.

COMPUTATIONS

The PWC90 console can be configured to perform calculations on historical data as it is collected. Available calculations are:

- Average.
- Minimum.
- Maximum.
- Standard deviation.

Computations can be performed hourly, daily, monthly or yearly. The computed values are stored in historical computation atoms. The computed values stored in these atoms can be used by other applications, such as Trending. Atoms are also provided for two additional values, minimum time and *Maximum Time*. These atoms are used to store the timestamp of the first instance of the minimum and maximum collection values. The atom names are shown in Table 5-1.

Table 5-1. Historical Computation Atoms

Period	Average	Minimum	Minimum time	Maximum	Maximum Time	Standard Deviation
HOURLY	.HAV	.HMN	.HTMN	.HMX	.HTMX	.HSD
DAILY	.DAV	.DMN	.DTMN	.DMX	.DTMX	.DSD
MONTHLY	.MAV	.MMN	.MTMN	.MMX	.MTMX	.MSD
YEARLY	.YAV	.YMN	.YTMN	.YMX	.YTMX	.YSD

Click on *Computations* to display the *Historical Tag Computations* window (Figure 5-7).

The *Historical Tag Computations* window is composed of a matrix of toggle buttons which allows selection of combination(s) of computation rate and computation type. Each row in the matrix represents a computation rate; each column represents a computation type. By default, all buttons are deselected and displayed in gray and appear raised. Click on the toggle button which corresponds to the desired computation

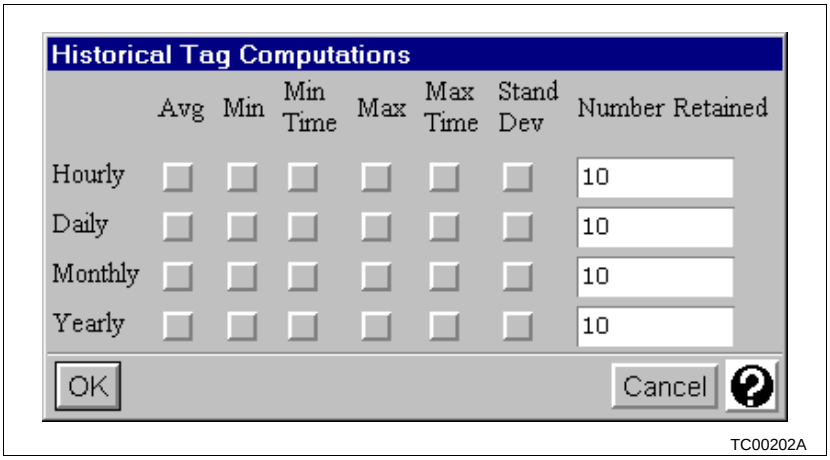


Figure 5-7. Historical Tag Computations Window

rate and computation type; selected buttons are displayed in yellow and appear pushed in.

Use the *Number Retained* text entry box at the end of each row to define the number of values to be retained for each computation type selected for that computation rate. Once the retention limit has been reached, the value obtained by each new computation will replace the oldest value. Because the retention limit is defined on a per computation rate basis, it is not possible to set the *Number Retained* to a different value for different computation types with the same computation rate. For example, it is not possible to set a different *Number Retained* for the hourly/average and hourly/minimum values. However, it is possible to configure different retention limits for hourly/average and daily/average values. Enter **inf** (infinity) in the *Number Retained* text entry box to retain all values collected; otherwise, enter a numeric value (zero is not a valid numeric value). To retain no data for a particular computation type, all toggle buttons for the computation type must be deselected.

> **NOTE:** No checks are performed to validate collection rate vs. computation rate. The user must consider the collection rate of the data and take care to configure appropriate computation rates; i.e., while it is technically feasible to perform hourly computations on data which is collected once a day (daily), such a configuration would result in many unnecessary and non-productive computations.

STORAGE RATE

Storage Rate is used to define the number of collections which will occur before a value is actually stored in the historical database; it is a multiple of the collection rate. For example, if a point has a collection rate of one second and a storage rate of ten, then only data for every tenth collection would actually be stored in the historical database.

The default storage rate is one; to change the storage rate, click on the button which displays the value of the storage rate. The *Storage Rate* (Figure 5-8) window will appear.

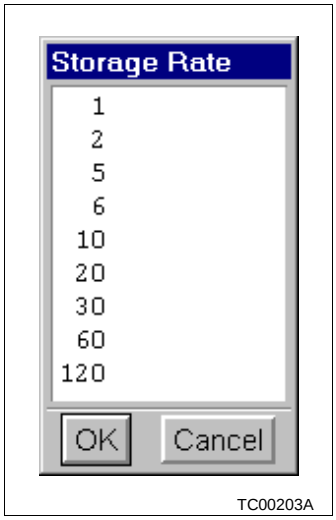


Figure 5-8. Storage Rate Window

Valid storage rates are 1, 2, 5, 6, 10, 20, 30, 60 and 120. Select the desired storage rate by clicking on it. The selected item will be highlighted. Click on *OK* to complete the operation, or click on *Cancel* to exit without changing the current storage Rate.

COLLECTION OPTION

The *Collection Option* parameter on the *Historical Tag Configuration* window is used to define the form in which collected values are stored. Available options include *AV*era*G*e, *MIN*imum, *MAX*imum and *INST*antaneous. The default collection option is *AVG*; to change this, click on the button which displays the current collection option. The *Collection Options* window will appear (Figure 5-9).

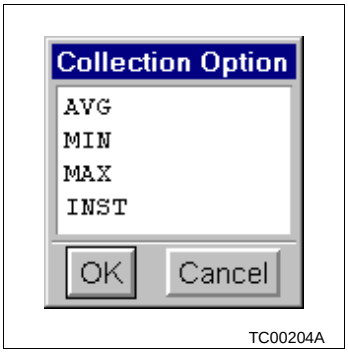


Figure 5-9. Collection Options Window

The range of data points over which the collection options operate is determined by the collection rate and storage rate. For example, if the collection rate equals one and the storage rate equals ten, then the collection options *MIN*, *MAX*, and *AVG* will operate using the last ten data samples that have been collected since the last storage time. If the *INST* collection option has been specified, the last value collected at storage time will be used.

Select the desired collection option by clicking on it. The selected item will be highlighted. Click on *OK* to complete the operation, or click on *Cancel* to exit without changing the current collection option.

ARCHIVE GROUP

An archive group is a set of data which is scheduled for archiving at a specified time. (The name and scheduling information for archive groups is configured using the *Archival Groups* item on the *Configure* menu available from the system window *menu* button; refer to ***ARCHIVAL GROUPS*** in Section 2 for details).

The button to the right of the *Archive Group* label displays the name of the currently assigned Archive Group. If no archive group has been assigned to the point, it displays the legend *None*. To assign the historical database point to an archive group, click on this button. The *Archive Group* window (Figure 5-10) is displayed.

Figure 5-10. Archive Group Window

Click on the name of the desired archive group. It will be highlighted. Click on *OK* to complete the selection operation, or click on *Cancel* to exit without selecting a new archive group.

The default selection is *None*.

TAG ATOM

Each point in the historical database is assigned to a tag.atom in the INFI 90 OPEN server. In the entry box labelled *Tag Atom* enter the tagname and atom of a point in a INFI 90 OPEN server which is to be scanned. To do so, click on the entry box and enter the tagname and atom, separated by a period (.).

EVENT BIT

The information collected into the historical database is gathered based on one of the following:

- Collection rate (refer to ***COLLECTION RATE*** in Section 5).
- Event bit condition.
- Collection rate during the duration of an event bit condition.

Event bits are a set of 4096 network-wide event flags which can be set to indicate that an event has occurred. Event bits are set and reset via INFI 90 OPEN blockware program commands. Names can be assigned to event bits in the same manner as they are assigned to plant areas; this assignment is performed from the *Network Parameters* menu discussed in ***CONFIGURING NETWORK PARAMETERS*** in Section 2.

The assignment of an event bit to a point in the historical database allows the collection of data when the condition indicated by the event bit occurs. If the collection rate is set to *None*, then the data will be collected only on transition of the event bit to the active value (refer to ***ACTIVE VALUE*** in Section 5). If a Collection Rate other than *None* has been selected, data will be collected at that rate as long as the event bit is set to the active value.

To assign an event bit to trigger data collection for the historical database point, click on the *Event Bit* entry box to gain focus on it, then enter the event bit number (1 to 4096) or the event bit name (as defined using submenus of the *Network Parameters* menu).

ACTIVE VALUE

The active value is relevant only when the event bit is used. Use the active value to define the state of the event bit (0 or 1) which triggers the collection; the *Active Value* button displays the value which currently defines whether an event bit of 1 or 0 will trigger collection of data for the historical database point. Click on the button to toggle between the two values.

ACKNOWLEDGE BIT

The acknowledge bit is an event bit. it is used to confirm that a collection operation has completed. Click on the *Acknowledge Bit* entry box to gain focus, then enter the name or number of the event bit to be used as the acknowledge bit. When the event bit is set, it is toggled from its current state to the alternate state (i.e., from 1 to 0 or from 0 to 1).

A change to the state of the acknowledge bit can be used to trigger another collection.

Storage Information

The portion of the *Historical Tag Configuration Display* devoted to storage information provides the ability to configure the length of time data will be retained, the number of collections to be retained, and to specify which PWC90 console nodes will be used to store the data collected for the point.

ADD DATA STORAGE DEFINITIONS

To configure a new set of storage options, click on *Add* below the list box in the storage information portion of the *Historical Tag Configuration* window. The *Historical Collection Storage Options* window (Figure 5-11) will be displayed.

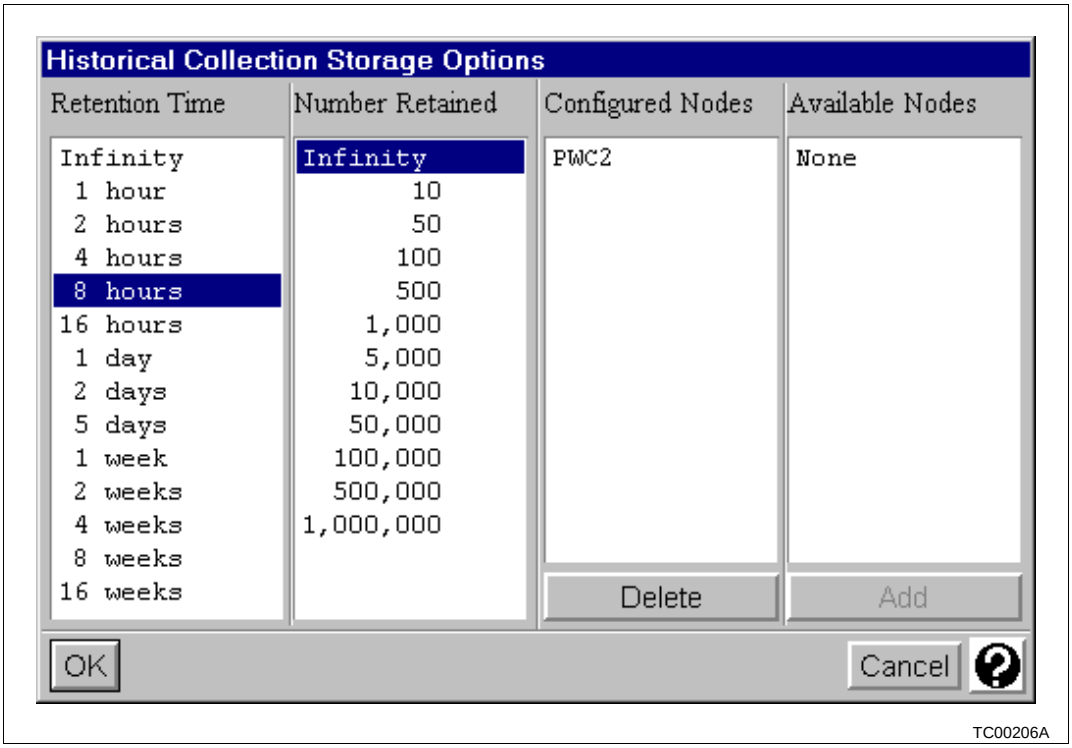


Figure 5-11. Historical Collection Storage Options Window

The *Historical Collection Storage Options* window is divided vertically into four list boxes containing entries for the following:

- Retention time.
- Number retained.
- Configured nodes.
- Available nodes.

Retention Time Option

The retention time is the length of time collected data is retained on a PWC90 console. The following retention times are available from the *Retention Time* list box:

- Infinity.
- 1 hour.
- 2 hours.
- 4 hours.
- 8 hours.
- 16 hours.
- 1 day.
- 2 days.
- 5 days.
- 1 week.
- 2 weeks.
- 4 weeks.
- 8 weeks.
- 16 weeks.

Click on the desired entry to select it; it will be highlighted. If a retention time other than *Infinity* is selected, the data is stored in a circular buffer. If *Infinity* is selected, data will be retained until the *Number Retained* specification is exceeded. When the time interval exceeds that which has been configured, the oldest data will be overwritten by new data. The amount of available disk space and the number, frequency and retention rate of other collections on the system should be considered carefully when configuring the retention time. The default time interval is eight hours.

> **NOTE:** If both a time interval and a number of samples (refer to ***Number Retained Option*** in Section 5) are configured, the first one to be completed will be used as the retention limit. For example, if both a time interval of one hour and 100 samples are configured for a retention span, and only 60 samples per hour are collected, then the 61st sample will overwrite the oldest sample stored.

Number Retained Option

To configure the number of collections to be retained for the selected historical tag, use the *Number Retained* list box, which contains the available options:

- Infinity.
- 10.
- 50.
- 100.
- 500.
- 1,000.
- 5,000.
- 10,000.
- 50,000.
- 100,000.
- 500,000.
- 1,000,000.

Click on the desired choice to select it; it will be highlighted. The default selection is *Infinity*, in which case data will be retained until the *Retention Time* is exceeded. The amount of available disk space and the number, frequency and retention rate of other collections on the system should be considered carefully when configuring the number of collections to be retained.

> **NOTE:** If both a time interval (Refer to ***Retention Time Option*** in this section) and a number of samples are configured, the first one to be completed will be used as the retention limit. For example, if both a time interval of one hour and 100 samples are configured for a retention span, and only 60 samples per hour are collected, then the 61st sample will overwrite the oldest sample in the circular buffer.

Node Assignment

The *Configured Nodes* and *Available Nodes* list boxes provide the ability to select and change the node on which data will be stored for this particular combination of *Retention Time* and *Number Retained*.

The available nodes list box contains an entry for each node which has not yet been assigned to store data for the selected Historical Tag. Click on the desired node name to select it, then click on *Add* at the bottom of the list box. The entry will be removed from the *Available Nodes* list box, and added to the *Configured Nodes* list box. When no more nodes are available the *Add* button will appear stippled out, and the list box will contain the word *None*.

The *Configured Nodes* list box contains an entry for each node which has already been assigned to store data for the selected historical tag. To delete a node from this list box, click on the desired node name to select it, then click on *Delete* at the bottom of the list box. The entry will be removed form the *Configured Nodes* list box, and added to the *Available Nodes* list box.

When no nodes are configured, the *Delete* button will appear stippled out, and the list box will contain the word *None*.

> **NOTE:** Two collection rates for the same historical tag cannot be stored on the same node; use a different historical tag, or assign the collections to different nodes.

Exiting the Historical Collections Storage Options Window

To exit the *Historical Collections Storage Options* Window and retain selections made in the list boxes, click on *OK*. To exit and discard any selections or changes, click on *Cancel*.

EDIT DATA STORAGE DEFINITIONS

To change an existing data storage definition, click on the desired entry in the list box on the *Historical Tag Configuration* window (Figure 5-2), then click on *Edit* below the list box. The *Historical Collections Storage Options* window (Figure 5-11) will be displayed. Configuration is performed in the same manner as that for adding data storage definitions, described in ***ADD DATA STORAGE DEFINITIONS*** in Section 5.

DELETE DATA STORAGE DEFINITIONS

To delete a data storage definition, click on the desired entry in the list box on the *Historical Tag Configuration* window (Figure 5-2), then click on *Delete* below the list box. The entry will be removed from the list box. At least one storage node must be specified; attempts to remove the last entry from the list box will result in an error message.

Convenience Buttons

In addition to the usual *OK* and *Cancel* buttons, five additional buttons appear at the bottom of the *Historical Tag Configuration* window (Figure 5-2). These include *Replace, Add, Prev, Next* and *Default.* These buttons provide a convenient way to configure multiple historical database points without exiting the *Historical Tag Configuration* window.

REPLACE

If the configuration of an existing historical database point has been changed, click on *Replace*. The existing configuration on the system for that historical database point will be replaced by the revised configuration.

ADD

To create a new historical database point from the *Historical Tag Configuration* window, using the current parameter config-

uration, click on *Add*. The *Historical Tag Configuration* window is ready to receive new input.

Alternatively, if an existing historical database point has already been configured, select it for *Edit*, change the tagname and any other characteristics as desired, then click on *Add*. The new point will be created. This feature allows creation of a group of historical database points with the same characteristics, without having to re-configure the same parameters for every point, and without leaving the *Historical Tag Configuration* window.

PREV/NEXT

Use the *Prev* and *Next* buttons to move to configuration displays for the historical database points which are adjacent to the one currently on display. Historical tagnames are listed in alphanumeric order; the actual order of the points is that shown in the list box of the *Historical Database Configuration* window (Figure 5-1).

RETURN TO DEFAULT SETTINGS

Click on *Default* to change all parameters to their default settings. Default settings are illustrated in Figure 5-12.

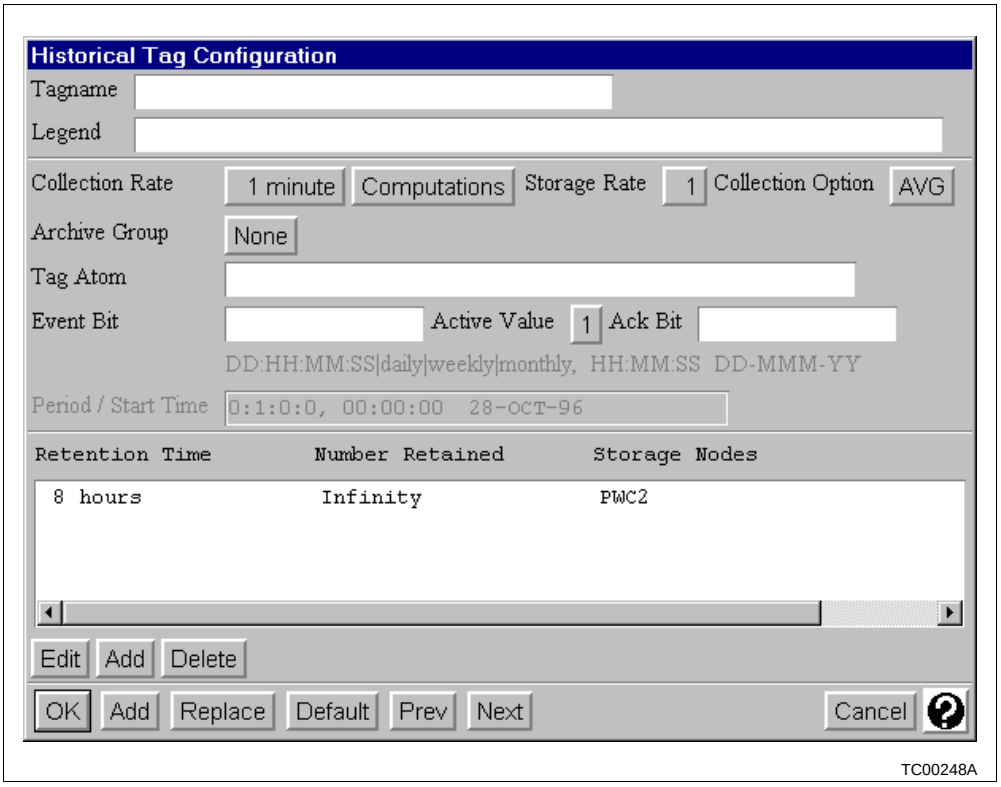


Figure 5-12. Historical Tag Configuration Window with Default Settings

Historical Database Configuration and Trend Compression

When configuring historical database points which will later be viewed in compressed form on a trend display, it is important to remember that the primary factor associated with the length of time it takes to display a compressed trend is the amount of data which must be processed. When the PWC90 console that is controlling the operator window which contains the trend display is different from the PWC90 console on which the data is stored, then data must be transferred across the system network, which increases the length of time needed to compress the trend display. It can take over twice as long to compress a large trend under these circumstances; thus, transferring less data across the network becomes even more important under these circumstances.

> **NOTE:** Compression of trend displays is discussed in detail in the ***Personal Work Center 90 Operation*** instruction.

The periodic collection rate, collection option and storage rate of historical database points can be configured so that data compression can be performed in the historical database as a background task, rather than as a function of the trend display. This can reduce the length of time required to display the compressed data.

For instance, an initial compression ratio of ten to one can be achieved within the historical database by storing a value that is the average of ten collection samples (collected at one second), and displayed on a trend display so that one value is displayed for each point on the plot. Subsequent compression of the trend display using a compression ratio of 50:1 will be much faster, and will yield exactly the same trace as that displayed when averaging of all collection samples (ten at a time, using the *Seconds* and *Points* parameters described in) and compression (using a 50:1 compression ratio) are performed via the trend display. By storing a value that is the average of ten samples, the total amount of data extracted from the historical database and averaged is one tenth of what would be required if every sample were stored and averaged on the trend display.

OTHER HISTORICAL DATABASE OPERATIONS

Add

To create a new historical database point, click on *Add* on the *Historical Database Configuration* window. The *Historical Tag Configuration* window will be displayed with default values (Figure 5-12).

Configure as described in ***EDIT HISTORICAL DATABASE ITEM*** in Section 5.

Find

The *Find* button is located in the lower left corner of the *Historical Database Configuration* window (Figure 5-2), and can be used to ascertain the presence of a given point in the historical database. If the point does exist, the *Find* button quickly locates it on the *Historical Tagname* list and highlights. Once located, the point configuration can be viewed, edited or deleted as desired.

To use this feature, click on *Find*. A pop-up window appears, which contains an entry box, and *OK*, *Apply* and *Cancel* buttons. Click on the entry box to focus on it, then enter the desired historical database tagname. If only a portion of the tagname is entered, the first entry where the first letters of the tagname matches the entered string will be chosen. For example, if the string HR is entered, and the first tagname to begin with these characters is HRANO1, then this is the tagname that would be highlighted in the *Historical Tagname* list.

Click on *Apply* to locate the first instance of the string; the *Find* pop-up window will remain open; click on *Apply* again to locate the next instance of the string, or enter a new string to be located. Click on *OK* to perform the *Find* operation and close the *Find* pop-up window. Click on *Cancel* to exit without attempting to find the tagname in question, or to close the pop-up window.

Delete Historical Database

The *Delete* button is used to delete a point or group of adjacent points in the historical database. Click on the tagname of the point to be deleted; it will be highlighted. If more than one point is to be deleted, position the pointer on the first tagname to be deleted, then press the left mouse button and drag the pointer until all the points to be deleted are highlighted. Click on *Delete*; a confirmation box will appear, which asks:

Delete Selected Items?

Click on *Yes* to delete all highlighted points, or click on *No* to exit the operation without deleting anything.

Delete All Historical Database

To delete all points in the historical database, click on *Delete All*. A confirmation box will appear. Click on *Yes* if all points in the database are to be deleted; click on *No* to exit without deleting points.

Exiting Historical Database Configuration

Click on *Close* to exit historical database configuration and close the *Historical Database Configuration* window.

SUMMARY - HISTORICAL DATABASE

Historical Database

The historical database is used to store data collected from the INFI 90 OPEN server real-time tag database; this data is used by various applications such as logging, trending and data archiving.

Acknowledge Bit

An acknowledge bit is an event bit which is used to confirm that a collection operation has completed in the historical database. When this bit is set, it is toggled from its current state to the alternate state (i.e., from 1 to 0 or from 0 to 1).

Active Value

The active value is used to define the state of the event bit (0 or 1) which triggers a collection; the face of the active value button displays the selected value. This parameter is relevant only when the event bit is used.

Archive Group

An archive group is a set of data which is scheduled for archiving at a specified time. (The name and scheduling information for archive groups is configured using the *Archival Groups* item on the *Configure* menu available from the system window *Menu* button; refer to ***ARCHIVAL GROUPS*** in Section 2 for details).

The face of the archive group button displays the name of the currently assigned archive group. If no archive group has been assigned, it displays the legend *None*. To assign the currently selected historical database point to an archive group, click on this button, then select the name of the desired archive group in the resulting pop-up window. The default selection is *None*.

Collection Option

The collection option is used to define the form in which collected values are stored. Available collection options are *AV*era*G*e, *MIN*imum, *MAX*imum and *INST*antaneous. The default option is *AVG*; to change the option, click on the button which displays the current collection option; a new collection option can be selected from the resulting pop-up window.

Collection Rate

The collection rate is the interval between collections, the frequency with which data is gathered from an associated point in the INFI 90 OPEN server. The default collection rate is one minute. Valid collection rates range from one second to one day, plus three additional options: *None, Manual,* and *Custom*.

The *None* option is used when collection is based on an event condition, rather than a time interval. The *Manual* option is used when data is to be entered manually, such as lab data.

The *Custom* option allows the user to define a custom time interval other than the standard intervals supplied by Bailey.

Computations

The PWC90 console can be configured to perform computations on data as it is collected. Available calculations are: average, minimum, maximum and standard deviation. Computations can be performed hourly, daily, weekly or yearly. The computed values are stored in historical computation atoms, which can be accessed by other applications, such as trending. Atoms are also available to store the timestamp of the first instance of the minimum and maximum collection values.

Data Type

When manual data points are configured, the user can define a data type for the manually entered data. Available data types include:

- Numerical.
- Alphanumerical (32 characters).
- Alphanumerical (64 characters).
- Alphanumerical (128 characters).
- Alphanumerical (256 characters).
- Alphanumerical (512 characters).

Event Bit

Event bits are a set of 4096 network-wide event flags which can be set to indicate that an event has occurred. Event bits can be addressed by name (refer to Section 2 for details on event bit name assignment) or number. Assignment of an event bit in the historical database allows the data to be collected when the condition indicated by the event bit occurs.

Historical Database Tagname

The historical database tagname is the tagname used to identify the point in the historical database. It can be up to 37 characters in length, cannot contain spaces, and must be unique on the network. If no historical database tagname is configured, it will default to the tagname of the associated server database point.

Storage Rate

The storage rate is used to define the number of collections which will occur before a value is actually stored in the historical database; it is a multiple of the collection rate, i.e., if a point has a collection rate of one second and a storage rate of ten, then only data for every tenth collection would actually be stored in the historical database. The default storage rate is one.

SECTION 6 - LOGS

INTRODUCTION

A log is used to collect and format data for use in reports and spreadsheet calculations. Two types of logs are available on the PWC90 console; these include:

- Event logs.
- Spreadsheet logs.

Event logs are used to examine the historical database and create lists of all events which occurred within a specific time span.

Spreadsheet logs are used to obtain data from, change, and put data into the INFI 90 OPEN server database, and to load the data into spreadsheets, allowing the user to format, store, and perform calculations using live process data. Spreadsheet logs are network-wide; i.e., a spreadsheet log can be configured or changed on any PWC90 console on the network, and the resulting changes are reflected on all other PWC90 consoles. Spreadsheet log configuration is performed using Microsoft EXCEL.

> **NOTE:** Logs are sent to a printer device when generated. If a printer has been configured with the device type of file, logs configured to print to that printer device will be sent to that file, rather than to an actual hardcopy printer.

ACCESSING LOG CONFIGURATION

Log configuration is accessed by using the pull-down menus available from the system window *Menu* button. Click on *Menu*. From the resulting pull-down menu, click on the *Configure* option, then click on *Logs* in the resulting cascading menu. The *Log Configuration* window (Figure 6-1) will appear.

Two radio buttons, each representing a different log type (event, spreadsheet), are located at the top of the *Log Configuration* window. By default, the *Event Logs* is selected. To change the type of log to be configured, Click on the radio button associated with the desired log type.

A list box appears below the log type buttons. The log name of each log configured (if any) for the selected log type will appear in the list box.

At the bottom of the window, in addition to the usual *Close* and *Help* buttons, are buttons used to configure, schedule and

print logs. These are the *Add, Delete, Edit, Schedule* and *Print* buttons.

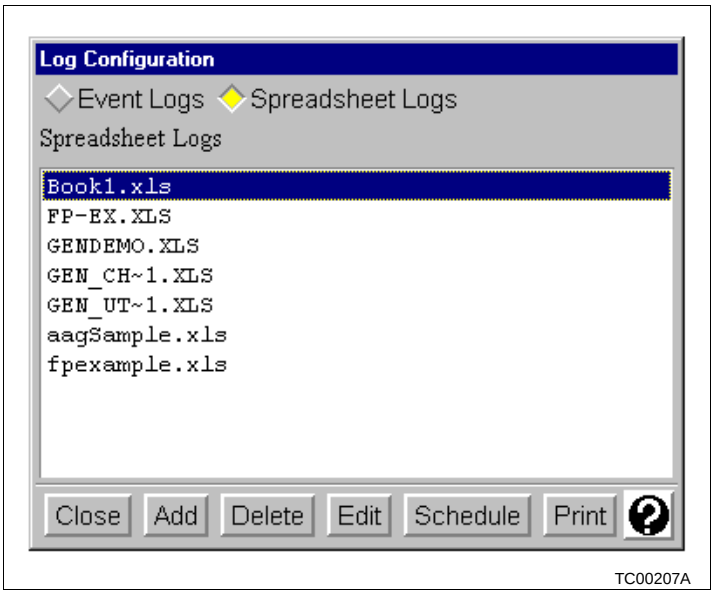


Figure 6-1. Log Configuration Window

EVENT LOGS

Event log configuration is used to create logs which obtain information about events which occurred in the historical database within a specified time span. To configure event logs, click on the *Event Logs* toggle button, which is located at the top of the *Log Configuration* window.

Add

To create an event log, click on *Add* at the bottom of the *Log Configuration* window; the *Event Log Configuration* window (Figure 6-2) will be displayed.

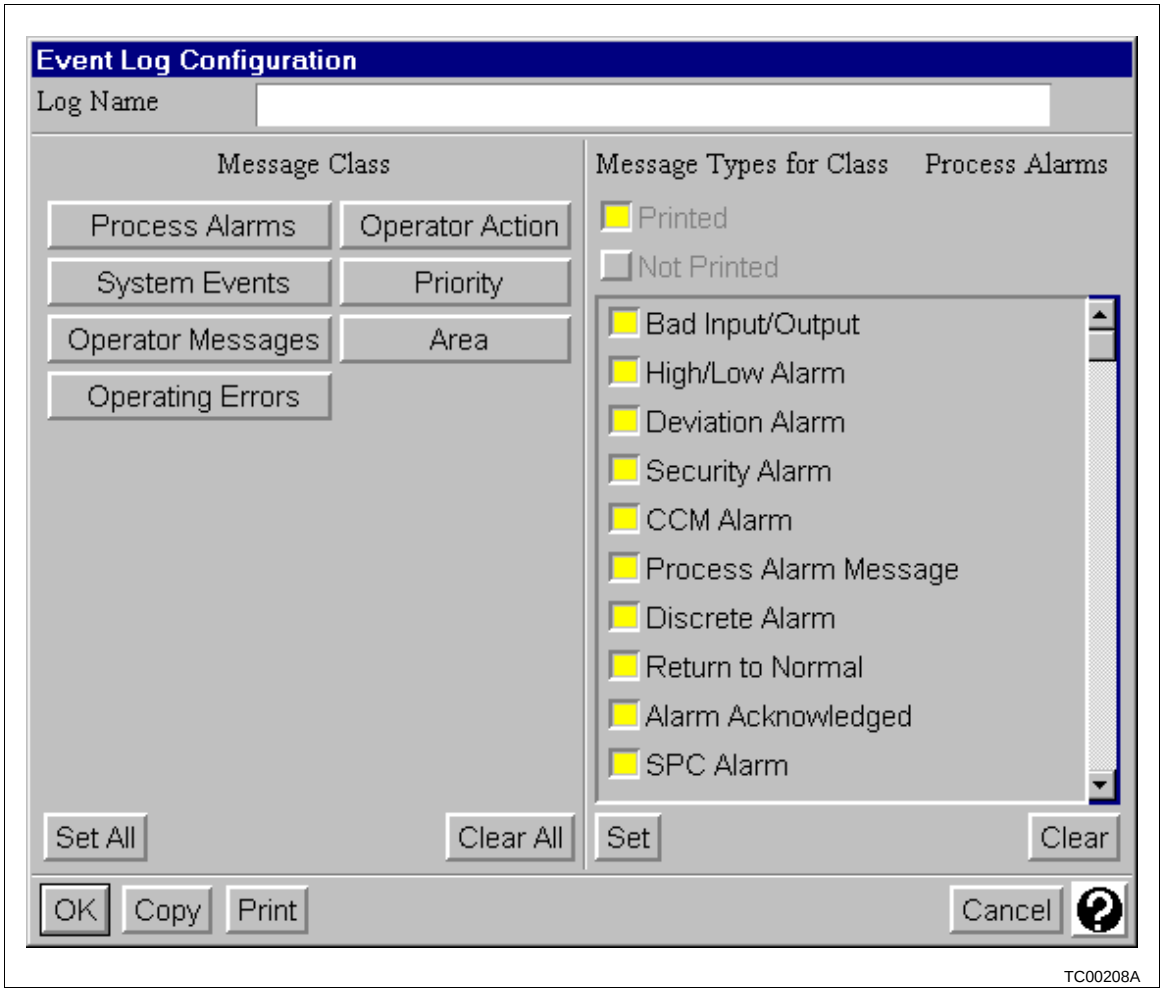


Figure 6-2. Event Log Configuration Window

CONFIGURE EVENT LOG NAME

The first item to appear in the *Event Log Configuration* window is a text entry box labelled *Log Name.* Enter a string of up to 50 alphanumeric characters to be used as the name of the log; the log name entered here is used in the *Event Logs* list box on the *Log Configuration* window.

CONFIGURE MESSAGE CLASSES AND TYPES

The events contained in an event log can be limited or filtered by message class and message type, by a method similar to that used when configuring filtering for ***Message Routing*** in Section 2. The main display area of the *Event Log Configuration* window (Figure 6-2) is divided into two portions: the left side is labelled *Message Class*, and contains buttons which display the available message class names. The right side of the main display area contains a list box, which in turn contains a scrollable list of all the message types available for the

selected message class. Click on the desired message class button to select it. message classes include:

- Process alarms.
- System events.
- Operator messages.
- Operating errors.
- Operator actions.
- Priority.
- Area.

The list of message types displayed in the list box reflects the message types available for the currently selected message class. Each message type has a button associated with it; click on this button to select or de-select the message type. If a message type is selected it will appear pressed in, while a de-selected item appears to be raised. Samples of the selected and de-selected states appear above the message type list box; the selected state is labelled *Printed*, while the de-selected state is labelled *Not Printed*. Note that these labels are stippled out, indicating that they are not available for actual use.

Process Alarms Message Class

Process alarm messages are classified into the following message types:

- Bad Input/Output.
- High/Low alarm.
- Deviation alarm.
- Security alarm.
- Process alarm message.
- Discrete alarm.
- Return to normal.
- Alarm acknowledged.

System Events Message Class

System Event messages are classified into the following message types:

- INFI 90 OPEN events.
- PWC90 events.
- Network events.
- Return to normal.
- Event acknowledged.
- Node down.
- Node up.

Operator Messages Message Class

Operator messages are messages sent by an operator to all operators in an area, and appear in the fourth line of the

mini-alarm window. Operator messages are classified into the following message types:

- Operator message.
- Message acknowledged.

Operating Errors Message Class

Operating error messages are classified into the following message type:

- Operating message.

Operator Actions Message Class

Operating Error messages are classified into the following message types:

- Network operator actions.
- PWC90 operator actions.

Priority Message Class

When the priority message class is selected, the list box contains a scrollable list of the 16 available priorities. By default, all priorities are selected. If a priority has been selected, event messages regarding historical database points with that priority will be included when the event log is generated.

Area Message Class

When the area message class is selected, the list box contains a scrollable list of the 64 available areas. By default, all areas are selected. If an area has been selected, event messages from historical database points in that area will be included when an event log is generated.

Clearing And Setting Event Log Message Types

It is possible to clear or set multiple message types simultaneously, eliminating the need to click on numerous toggle buttons. Click on *Set* or *Clear* to set or clear all message types in the selected message class. Click on *Set All* or *Clear All* to set or clear all message types in all event log message classes on the PWC90 console.

Edit

If an event log has already been configured, the configuration can be revised using the *Edit* button, located at the bottom of the *Log Configuration* window. To do so, first click on the desired log name in the list box to select it, then click on *Edit*.

The *Event Log Configuration* window (Figure 6-2) will be displayed, containing the parameters configured for that event log. Changes can be made to these parameters as described in ***Add*** in Section 6.

Delete

To delete an event log, click on the log name in the list box which appears in the *Log Configuration* window (Figure 6-1), then click on *Delete* at the bottom of the window. A confirmation box will appear, which queries:

Delete selected items?

Click on *Yes* to delete the event log, or click on *No* to exit the operation without deleting the log.

Scheduling

Event logs can be scheduled to print on a regular basis. Click on the log to be scheduled to select it, then click on *Schedule* at the bottom of the *Log Configuration* window (Figure 6-1). The *Log Scheduler* window will appear.

SCHEDULE BUTTON

The first item in the *Log Scheduler* window is a toggle button labelled *Schedule By*; click on this button to toggle between the available choices, which are:

- Not Scheduled.
- Time.
- Event.
- Time And Event.

The currently selected state appears on the face of the button. If *Time* is selected, the summary log will print at the scheduled time. If *Event* is selected, the event log will print when the specified event bit (if any) is set. If *Time And Event* is selected, the log will print at both the specified time and whenever the specified event bit is set. If *Not Scheduled* is selected, the other scheduling parameters can be configured, but the log will not be printed.

SCHEDULED TIME PARAMETER

The period (user-defined interval) at which the Event log will be printed and the starting time of that period appear on the face of the button labelled *Scheduled Time*. To configure these parameters, click on *Scheduled Time*. A pop-up window (Figure 6-3) will appear.

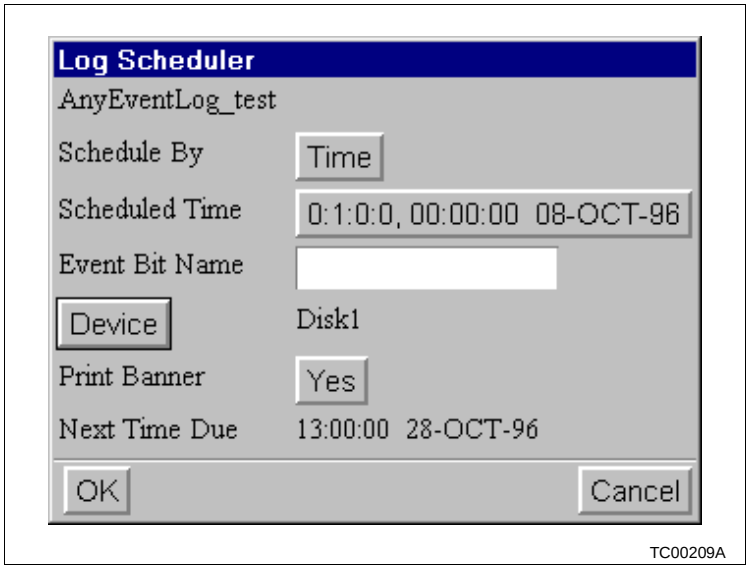


Figure 6-3. Event Log Scheduling Window

The period is defined first; the words daily, weekly and monthly are valid entries to define the period. Alternatively, the user can define a customized period by defining the number of days, hours, minutes and seconds in the period using the following format:

DD:HH:MM:SS

The period must be followed by a comma (,). After the period has been defined, the starting time (hours/minutes/seconds and day/month/year) is entered in the following format:

HH:MM:SS DD-MMM-YY

The default period is one hour (0:1:0:0), the 24 hour clock is used, and the default starting time is midnight of the current day (i.e., 00:00:00 19-APR-96). Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name.

EVENT BIT NAME PARAMETER

If the log is to be triggered by an event bit, click on the text entry box labelled *Event Bit Name* to gain focus, then enter the number or name of the event bit to be used to trigger the log.

DEVICE PARAMETER

A button labelled *Device* is located below the *Event Bit Name* entry box. If a device has been selected, the device name will appear to the right of the *Device* button; if no device has been selected, the space to the right of the *Device* button will be

empty. (A device can be a printer or a file; refer to ***Printer Assignment*** in Section 2 for information on how to configure print devices). To select or change the printer device, click on this button. The *Printer Selection* window (Figure 6-4) will be displayed.

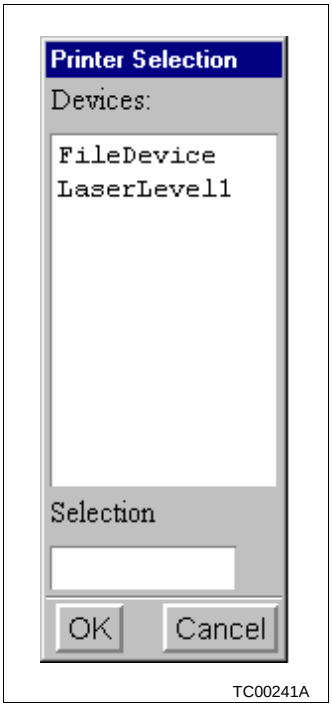


Figure 6-4. Printer Selection Window

A list box in the *Printer Selection* window contains the names of line and laser printers configured on the network. Click on the desired printer name in the list box. It will be highlighted. Click on *OK* to complete the operation, or click on *Cancel* to exit the *Printer Selection* window without changing the current printer selection.

PRINT BANNER PARAMETER

A *banner* is a page bearing information such as log name and the time and date of the log, which is printed immediately prior to the log itself. A button labelled *Print Banner* is located below the *Printer Device* button. The face of the *Print Banner* button displays *Yes* if a banner page is to be printed, or *No* if the banner is to be omitted. Click on the button to toggle from one state to the other.

NEXT OCCURRENCE INDICATOR

The last item in the *Log Scheduler* window, labelled *Next Time Due*, provides the user with the time and date of the next scheduled log.

EXITING EVENT LOG SCHEDULING

Click on *OK* to complete the scheduling operation and exit the *Log Scheduler*, or click on *Cancel* to exit without making changes to the current scheduling configuration.

Print

The *Print* button on the *Log Configuration* window allows the user to print logs on demand. Click on this button to display the *Print Event Log* window (Figure 6-5).

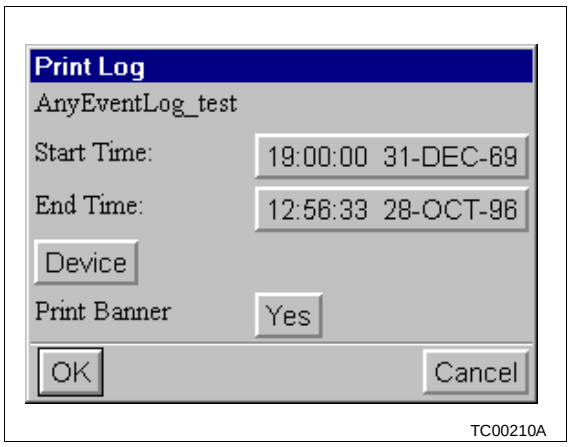


Figure 6-5. Print Event Log Window

The first item on the *Print Event Log* window is a pushbutton labelled *Start Time*. The time and date displayed on the face of this button indicate the earliest time and date to be used for messages included in the event log. The default starting time is 19:00:00 31-DEC-69; to change this, click on the button. A pop-up window appears, containing an entry box and the prompt:

Enter Start Time (HH:MM:SS DD-MMM-YY).

Enter the desired starting time in the indicated format. Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name. Click on *OK* to complete the operation, or click on *Cancel* to exit without changing the starting time.

Similarly, the second item on this window is a pushbutton labelled *End Time*. The time and date displayed on the face of this button indicate the latest time and date allowed on messages to be included in the event log. The default end time is the current time and date.

The third item on this window is a button labelled *Device*, and is used to identify the print device to which the log is to be sent.

The face of the button displays the print device name, or a question mark (?) if no print device has been selected. Click on this button to configure the print device to which the log is to be printed. The *Printer Selection* window will be displayed. Highlight the desired printer by clicking on it, then click on *OK* to complete the selection operation, or click on *Cancel* to exit without changing the device selection.

The fourth item is a pushbutton labelled *Print Banner*. The face of the associated button displays *Yes* if a banner is to be printed, or *No* if the banner is to be omitted. Click on this button to toggle from one state to the other. Click on *OK* to complete the log printing operation, or click on *Cancel* to exit changing the current selections.

SPREADSHEET LOGS

Spreadsheet logs are used to obtain data from, change, and put data into the INFI 90 OPEN global database. The data is loaded into spreadsheets, allowing the user to format, store, and perform calculations on live process data. The spreadsheet logs are accessed by clicking on the radio button labelled *Spreadsheet Logs*, located on the *Log Configuration* window (Figure 6-1). The list box in the *Log Configuration* window will list all spreadsheet logs which have been configured on the network.

Add

To create a spreadsheet log, click on *Add* at the bottom of the *Log Configuration* window. A blank EXCEL spreadsheet is opened. For information on the operation of EXCEL refer to the documentation provided with the EXCEL software. After a spreadsheet has been created and saved, it will appear in the edit list.

Edit

If a spreadsheet log has already been configured, the configuration can be accessed for revision by using the *Edit* button, located at the bottom of the *Log Configuration* window. To do so, first click on the desired log name in the list box to highlight it, then click on *Edit* button.

Delete

To delete a spreadsheet log, click on the log name in the list box which appears in the *Log Configuration* window (Figure 6-1), then click on *Delete* at the bottom of the window. A confirmation box will appear which queries:

Delete selected items?

Click on *Yes* to delete the summary log, or click on *No* to exit the operation without deleting the log.

Schedule

Spreadsheet logs can be scheduled to print on a regular basis, in the same manner as summary and event logs. Click on the filename of the log to be scheduled to select it, then click on *Schedule* at the bottom of the *Log Configuration* window (Figure 6-1). The *Log Scheduler* window (Figure 6-3) will appear.

Spreadsheet log scheduling configuration is identical to that for summary logs and event logs, described in ***Scheduling*** in Section 6.

Printing And Scheduling Microsoft Excel Spreadsheet Logs

To schedule and print a Microsoft EXCEL spreadsheet, first create a subroutine called AUTO_OPEN, which checks to see if the EXCEL spreadsheet was opened for configuration, or for printing as the result of use of the fpuprint function call. This subroutine must called AUTO_OPEN, and placed in a file in the XLSTART directory (found in the directory tree under which the Microsoft EXCEL software is installed). (If you have questions about how Microsoft EXCEL is installed at your site, see your system administrator). The AUTO_OPEN subroutine will be executed each time a Microsoft EXCEL spreadsheet is opened, as are all files in the XLSTART directory.

If the spreadsheet was opened for printing, it obtains the printer device using the fpudevice function call and initiates the printout. The subroutine then exits EXCEL. If the spreadsheet was opened for configuration, the subroutine returns.

```
' Auto_Open Macro
'

Declare Function fpuprint Lib "FPEXCEL.XLL" () As String
Declare Function fpudevice Lib "FPEXCEL.XLL" () As String
Sub Auto_Open()
    Dim flag As String

    Application.DisplayAlerts = False
    flag = fpuprint()
    If flag = "1" Then
    Application.ActivePrinter = fpudevice()
    ActiveWindow.SelectedSheets.PrintOut Copies:=1
    Application.Quit
    End If
End Sub
```

Print

The *Print* button on the *Log Configuration* window allows the user to print logs on demand. Click on this button to display the *Print Spreadsheet Log* window (Similar to Figure 6-5).

The first item on this window is the log name. Below the log name is a button labelled *Device*, which is used to identify the print device to which the log is to be sent. If a device has been selected, the device name will appear to the right of the *Device* button; if no device has been selected, this space will be empty. Click on this button to display the *Printer Selection* window (Figure 6-4). Highlight the desired printer by clicking on it, then click on *OK* to complete the selection operation, or click on *Cancel* to exit without changing the device selection.

Refer to ***Printing And Scheduling Microsoft Excel Spreadsheet Logs*** in Section 6 for information on macros needed for printing spreadsheet logs.

The second item is a pushbutton labelled *Print Banner*. The face of the associated button displays *Yes* if a banner is to be printed, or *No* if the banner is to be omitted. Click on this button to toggle from one state to the other.

Click on *OK* to complete the selection operation, or click on *Cancel* to exit changing the current selections.

Clearing the Recovery Log Notify Window

A *Recovery Log Notify* window will appear when a user attempts to edit or print a spreadsheet log, if the log was previously open for edit or was printing when EXCEL was terminated abnormally. To stop this window from appearing each time the log is requested to print, go into log configuration. Attempting to edit the spreadsheet log will also cause the *Recovery Log Notify* window to appear. To prevent this window from appearing every time an attempt is made to print or edit the log, select the filename of the log to load it and select the *Remove Recovery File* option in the window, then click on *OK*. After the spreadsheet is loaded, save the spreadsheet. If the spreadsheet is not saved, the *Recovery Log Notify* window will continue to appear each time an attempt is made to print or load the spreadsheet.

Spreadsheet Log Special Functions

A number of special functions have been developed by Bailey for use with the Microsoft EXCEL spreadsheet package. These include:

- fphget.
- fphgetd.

- fpsendmsg.
- fpput.
- fpputd.
- fphput.
- fphputd.
- fparchive.
- fpverbose.
- fpuprint.
- fpudevice.
- fpblkarch.
- fpblkclose.
- fpblkopen.
- fpblksample.
- fpblkvald.
- fphxnumval.
- fphxgetval.
- fpxammts.

For an explanation of these functions refer to Appendix C.

SECTION 7 - TRENDING

INTRODUCTION

Types of trend recording available on the PWC90 console include:

- Current trending.
- Historical trending.
- Archival trending.

The current, historical and archival trending functions display the values of one or more points in an analog trend format similar to that presented by a conventional strip chart recorder.

A current trend for each point appears as a part of the point display; current trends are not configurable, and are not stored. Current trending for a point exists only when the point is on display. Long term trending is handled separately, and is available in two forms:

Up to 1024 network-wide historical trends (each of which can be configured to display up to eight traces) are available on the PWC90 console. The configuration of these trends can be performed on any PWC90 console, and is then distributed to all other PWC90 consoles on the network. (The storage location of the actual trend data can be configured by the user via the historical or historical block database options, available from the *Configure* option on the system window menu).

Archival trending permits historical trends to be transferred to a removable tape cartridge, creating a permanent record which can be retrieved and displayed at a later date.

Any non-text (i.e., non-ASCII) point which exists in the historical database or historical block database can be trended.

Refer to the ***Personal Work Center 90 Operation*** instruction for details on use of the various features of the historical trend display.

ACCESSING HISTORICAL TRENDS

Historical trend configuration is performed in the operator window. The pointer must first be in an operator window which has keyboard focus.

To enter trend configuration using the menus, click on *Select* in the menu bar; in the resulting pull-down menu, click on Trend. A pop-up window will appear (Figure 7-1), containing an entry box and numeric keypad.

Figure 7-1. Trend Index Entry Pop-up Window

To enter the index number of the trend to be configured, click on displayed keypad numbers to enter the trend index number, or click on the entry box and then enter the trend index number via the keyboard. The left arrow (←) on the keypad corresponds to **Backspace** on the keyboard.

Trends can also be accessed using the *List* button on the *Trend Index Entry* pop-up window. Click on *List* to display the *Select Trend* pop-up window (Figure 7-2), which contains the number and (if assigned) the legend for all configured trends.

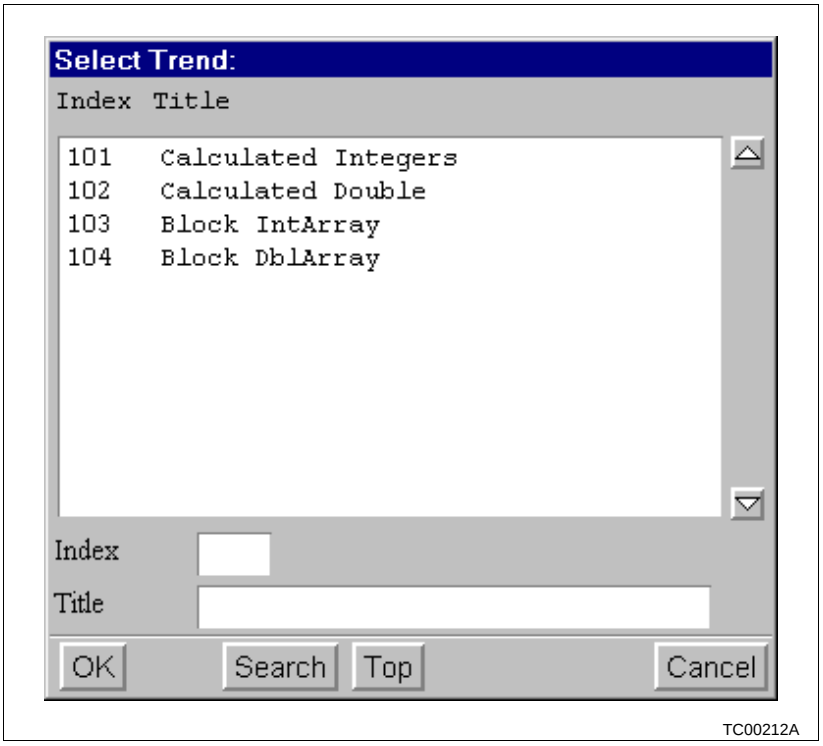


Figure 7-2. Select Trend Pop-up Window

The *Select Trend* pop-up window displays a list box containing the index number and title of all trends which have been configured. Up and down arrow buttons to the right of the list box provide access to additional pages of the list, while a scrollbar below the list box can be used to read trend titles which are too long to be displayed in their entirety.

Below the list box, entry boxes are provided for entry of the trend index number and trend title. Enter the trend index number in the *Index* entry box and click on *Search* to locate a trend via its index number. Alternatively, enter a trend title or portion thereof in the *Title* entry box, followed by a click on *Search*, to locate the next instance of the specified text.

Click on *Top* to return to the first part (top) of the list.

To complete the trend selection operation, click on *OK*; the chosen trend will be displayed in the operator window (Figure 7-3). Click on *Cancel* to exit without selecting a trend for configuration.

CONFIGURING HISTORICAL TRENDS

Historical trending permits the assignment of any non-text (i.e., non-ASCII) point in the historical database or historical block database to a multi-point trend display, whether or not the point is included in a hierarchical display. A maximum of eight points can be trended on a multi-point display. All points on a multi-point trend display need not have the same data collection rate.

Once a trend has been chosen, the trend display for that index number appears in the operator window. If no parameters have yet been configured for the trend, a template of the historical trend display appears. This contains all the basic parts of an historical trend display, but has no legend, no trace assignments, and default scale bar limits of 0.00 and 100.00. Scale bar limits have a default precision of two decimal places to the right of the decimal point.

To configure trend parameters and assign tagnames to traces, click on *Configure* in the menu bar of the trend display. A pop-up window entitled *Trend Configuration* appears (Figure 7-4).

The following parameters are configured in the *Trend Configuration* window:

- Title.
- High and low scale bar limits.
- Trend type (historical, archival).
- Segment size (Seconds and Points).
- Traces (to a maximum of eight per trend display).

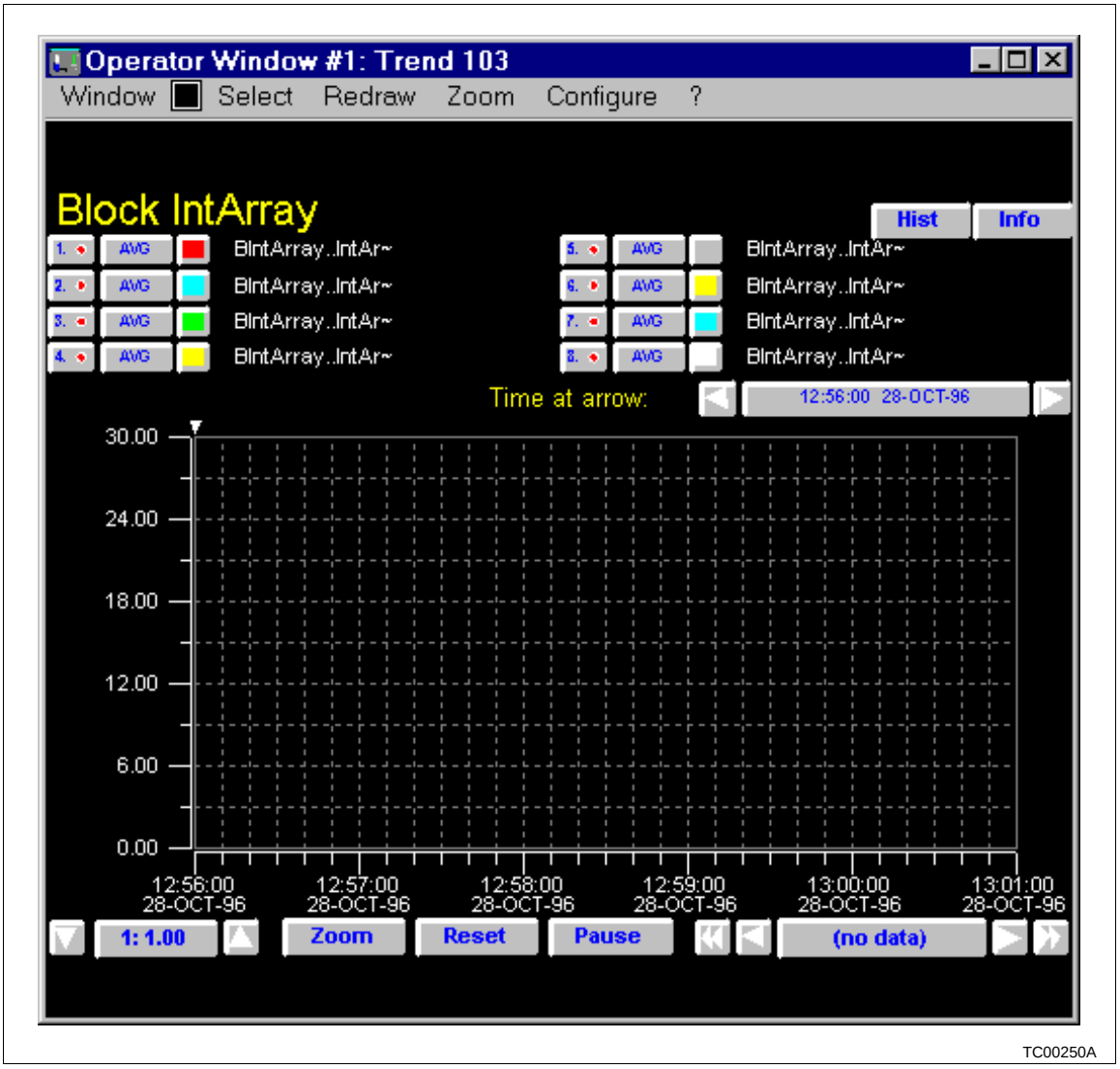


Figure 7-3. Historical Trend Display (Template)

- Trace range for each trace.
- Default trace state (On or Off).
- Display options (display as *MIN*, *MAX*, *AVG* or *INST*).

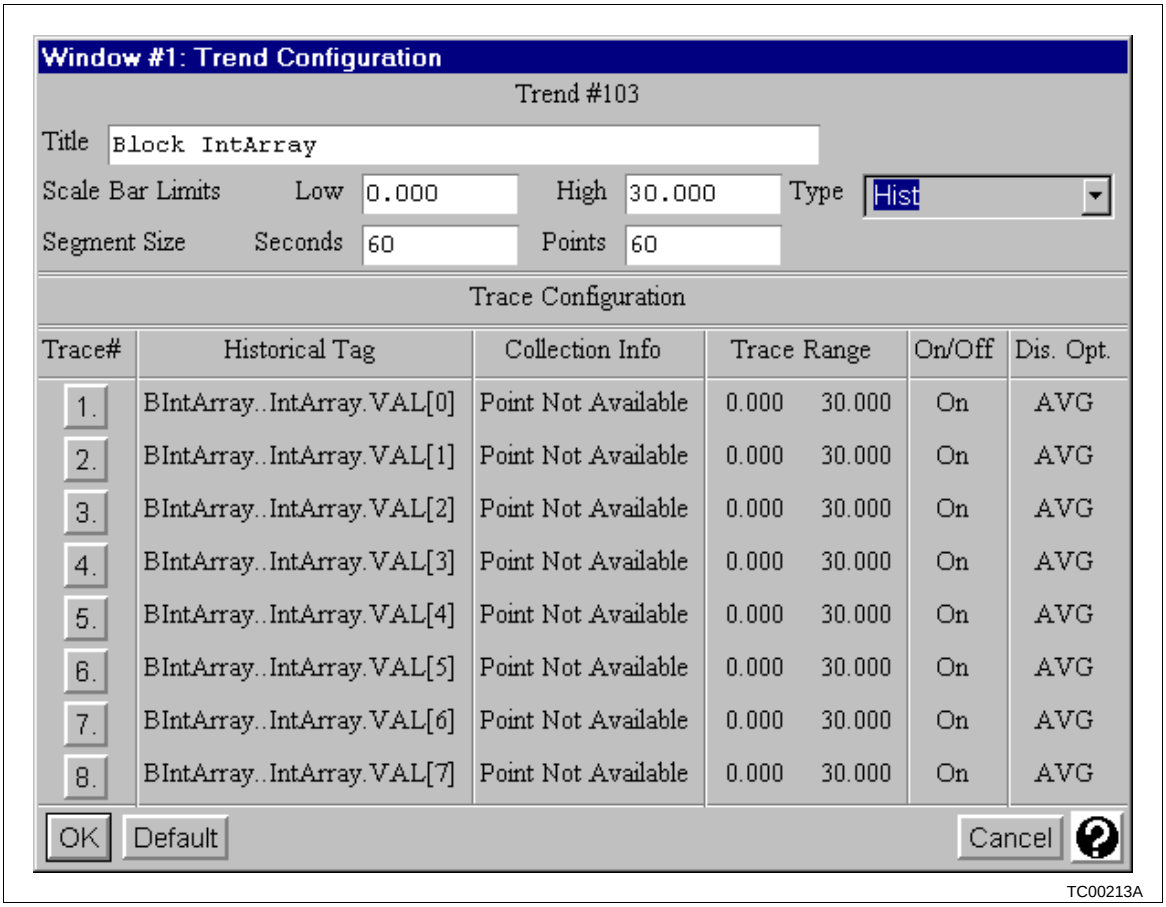


Figure 7-4. Trend Configuration Window

Title

The *Title* parameter is used to configure a trend display title of up to 48 characters in length. Once configured, the trend title appears at the top of the trend display. To configure this item, click on the entry box which follows the word *Title*; use the keyboard to enter the desired alphanumeric text string.

Scale Bar Limits

The *Scale Bar Limits*, *High* and *Low*, are used to configure the lowest and highest scale numbers on the vertical axis of the trend display. These limits become the default limits which appear, in white, each time the display is called up or reset via the *Reset* button (which appears at the bottom of the display). If no limits are specified, the *High* limit defaults to 100.00 and the *Low* limit to 0.00.

Segment Size

The horizontal axis of the trend is divided into five segments; the end of each segment is marked by a long tick-mark and a time and date stamp. Two parameters determine the size of a segment: *Seconds* and *Points*.

The *Seconds* parameter is used to assign the number of seconds in a segment. By default, this parameter is set to 60 seconds.

The *Points* parameter determines the maximum number of trend points which can be plotted in a segment. By default, a maximum of 60 points can be plotted per segment. The number of points determines the resolution of the trace. Do not assign more than 600 points to a segment.

To obtain a normal (non-compressed) display, the number of *Points* assigned per segment should not exceed the number of *Seconds* per segment.

To position points on the trend graph, the PWC90 console divides the number of seconds in the segment by the number of points in the segment, yielding the number of time intervals in the segment.

number of seconds / number of points

The default settings of 60 points and 60 seconds per segment thus yield the equivalent of a 1:1 compression ratio. It is possible to imitate the effects of data compression by varying these settings. For example, setting *Seconds* to 120, with *Points* set to 60, creates a trend trace which appears to have a compression ratio of 2:1.

> **NOTE:** The tick-marks which divide the segments on the time axis (x axis) of the trend graph are for visual reference only, and do not necessarily reflect the actual number of time intervals used by this algorithm.

The PWC90 console locates the points which lie within each time interval. If one point is found, the value of the point is used to plot it on the trend display. If more than one point exists within the interval, the assigned display option (*MIN*imum, *MAX*imum, *AV*era*G*e or *INST*antaneous) is used to calculate one value from all points which exist in the interval. This value is then plotted on the trend display. (If the *INST*antaneous display option is used, the last value found will be used to plot the point on the trend display). The resulting value is used to plot the point on the trend display. If no points are found in an interval, no point is plotted for that interval.

This method of plotting trends permits greater flexibility in trend configuration. For example, points with different collec-

tion rates can be plotted on the same trend graph, and non-periodic data can also be trended.

Trace Configuration

Up to eight traces can be configured on each Trend display. Each trace appears in a different color; the color for each trace is indicated on the colored button which appears to the left of the tagname, above the trend graph.

To configure a trend trace, click on the desired *Trace Number* button on the *Trend Configuration* window. A pop-up window entitled *Trend Trace Configuration* will appear. Trend trace parameters which can be configured via this window include:

- Historical tagname.
- High and low limits for the trace.
- Default trace state (On or Off).
- Display options (*MIN, MAX, AVG, INST*).
- Description.

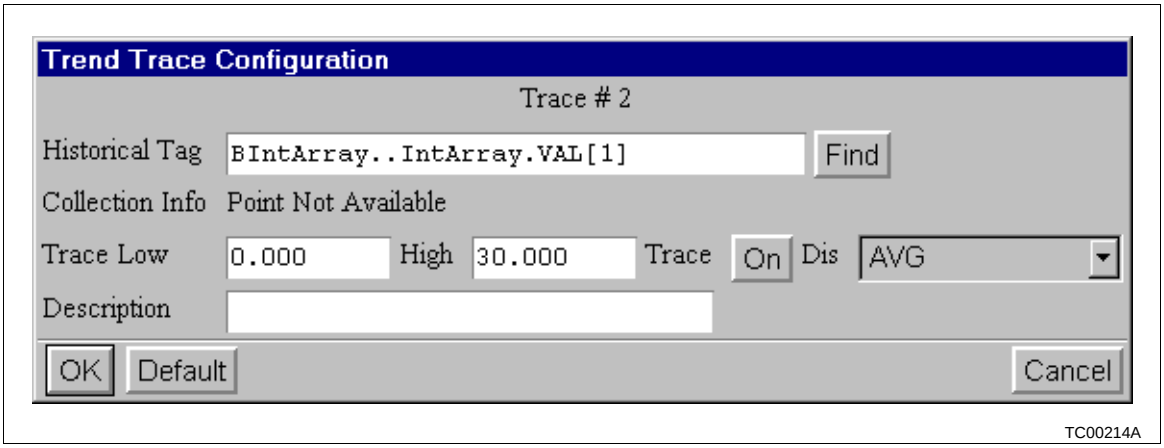


Figure 7-5. Trend Trace Configuration Window

HISTORICAL TAG

The *Historical Tag* entry box allows entry of the tagname of a point in the historical database, an historical computation point, or a point in the historical block database; the item entered will be assigned to the trend trace. The tagname can be typed in directly by clicking on the entry box to give it focus, then entering the information via the keyboard.

Historical Database Tagnames

If the tagname of a point in the historical database is not immediately known, this information can be located via the *Find* button, located immediately to the right of the *Historical Tag* entry box. Enter the known portion of the historical data-

base tagname in the *Historical Tag* entry box, then click on *Find*. The *Historical Tags* window will be displayed, with the first instance containing that string displayed in a list box, highlighted.

Clicking on *Find* with no entry in the *Historical Tag* entry box causes the *Historical Tags* window to be displayed. This window contains a list of all tagnames that are configured in the historical database.

Historical Computation Tagnames

Historical computation points are computed points in the historical database, and are specified in the format:

<htag>.<comptype>

where *htag* is the tagname of a point in the historical database and *comptype* is the atom for the computation type. Atom names for each computation type and computation rate are shown in Table 7-1.

Table 7-1. Historical Computation Atoms

Period	Average	Minimum	Minimum Time	Maximum	Maximum Time	Standard Deviation
Hourly	.HAV	.HMN	.HTMN	.HMX	.HTMX	.HSD
Daily	.DAV	.DMN	.DTMN	.DMX	.DTMX	.DSD
Monthly	.MAV	.MMN	.MTMN	.MMX	.MTMX	.MSD
Yearly	.YAV	.YMN	.YTMN	.YMX	.YTMX	.YSD

For example, RAND1.HAV contains the hourly average for the historical database point named RAND1.

Historical Block Database Block Names

The historical block database is designed for the efficient collection of data from a large number of INFI 90 OPEN server tag.atoms. Each block name in the historical block database can represent data from multiple tag.atoms in the INFI 90 OPEN server, which is collected and stored as a single entity called an historical block. All tags associated with an historical block are assigned the same parameter values for collection rate, storage node and computation options.

To reference data in an historical block, use the following format:

<block name>..<tagname>.atom.comptype

where *block name* is the name of the historical block, *tagname* is the tagname of the tag in the INFI 90 OPEN server real-time

tag database, *atom* is the INFI 90 OPEN atom of the point and *comptype* is the computation type atom. For example:

CALBLK1..CT2_LI01.C_AOUTPUT.HMX

represents the hourly maximum for the C_AOPUTPUT atom of the point CT2_LI01, stored in the historical block named CALBLK1, while

CALBLK1..CT2_LI01.C_AOUTPUT

represents the raw data for the C_AOUTPUT atom of the point CT2_LI01, stored in the historical block named CALBLK1.

COLLECTION INFO

The *Collection Info* field provides information on how the tagname has been configured in the historical database, including the tagname.atom, the collection rate, the storage rate and the storage option. (The storage option determines the form in which the data is stored: *MIN*imum, *MAX*imum, *AV*era*G*e or *INST*antaneous value). These parameters cannot be configured via trend configuration, but must be configured using the *Historical Database* item available via the system window *Configure* pull-down menu.

TREND TRACE LIMITS: HIGH AND LOW

Because each trace on the trend can represent data collected at a different (even non-periodic) collection rate, individual high and low trend trace limits can be configured for each trace. To set the *Low* limit for the trace, click on the *Low* entry box, then enter the value for the low limit. To set the *High* limit for the trace, click on the *High* entry box and enter the value for the high limit.

The trend graph is initially presented with a default set of scale bar limits, which are configured on the *Trend Configuration* window (refer to ***Scale Bar Limits*** in Section 7); these can be replaced by the individual limits for a particular trend trace by clicking the colored button associated with the trace. The individual trend trace limits will then be displayed as the scale bar limits, in the same color as the trend trace. To return to the default scale bar limits, click on *Reset* located at the bottom center of the display.

TRACE DEFAULT STATE: ON OR OFF

Each trace can be configured to be visible (On) or invisible (Off), by default, whenever the trend display is called up. To change the default configuration, click on the *Trace* toggle button. The default state of the trace will be displayed on the face of the button. When the trend is visible (ON), a red dot appears next

to the trace number on the trace display button, which appears above the trend graph on the display. On the trend display, the user can toggle between the ON and OFF states by clicking this button.

DISPLAY OPTIONS

The *DIS* (Display Option) drop-down list box allows selection of the form in which the data is to be presented on the trend display. Click on the drop-down arrow button to the right of the *DIS* box to view a list of choices, which include *AV*era*G*e (the default), *MIN*imum, *MAX*imum and *INST*antaneous. Click on the desired choice in the drop-down list. The *DIS* box will reflect the selected display option.

DESCRIPTION

Click on the *Description* entry field to enter a text string of up to 31 characters. This string appears at the upper left hand corner of the trend graph whenever the colored button associated with the trace is clicked.

Type

The *Type* parameter is used to select the type of trend to be configured. Available trend types are historical, archival and batch. The type of trend selected determines which database will be used to provide the data for the trend. To make the selection, click on the drop-down arrow to the right of the *Type* box to view a list of choices (*HIST, ARCH*). Click on the desired choice in the drop-down list. The *Type* box will reflect the selected database.

SUMMARY - TRENDS

Archival Trends Archival trending permits historical trends to be transferred to a removable tape cartridge, creating a permanent record which can be retrieved and displayed at a later date.

Collection Info Collection Info is a field in the *Trend Configuration* window which provides information on how the tagname has been configured in the historical database, including the tagname.atom, the collection rate, the storage rate and the storage option.

Current Trends A current trend for each point appears as a part of the point display; current trends are not configurable, and are not stored. Current trending for a point exists only when the point is on display.

Display Option (DIS) The *DIS* (Display Option) drop-down list box is the feature which determines the form in which the data is to be presented

on the trend display - as a *MIN*imum, *MAX*imum, *AV*era*G*e or *INST*antaneous value.

Historical Database The historical database is a set of points in the PWC90 console which collects and stores data gathered from one or more INFI 90 OPEN servers.

Historical Block Database The historical block database is a set of points in the PWC90 console designed for the efficient collection and storage of data from a large number of INFI 90 OPEN server tag.atoms, which can reside in one or more INFI 90 OPEN servers.

Historical Tag A historical tag is the tagname assigned to a point in the historical database. The historical database is configured using the historical database selection on the *Configure* menu available from the system window *Menu* button.

Historical Trends Up to 500 network wide historical trends are available on the PWC90 console (each of which can be configured to display up to eight traces). The configuration of these trends can be performed on any PWC90 console, and is then distributed to all other PWC90 console nodes on the network. (The storage location of the actual trend data can be configured via the historical database option, available from the configure option on the system window menu).

List Button The *List* button appears on the *Trend Index Entry* window. Click on this button to display the *Select Trend* window, which contains the number and (if assigned) the legend for all configured trends.

Scale Bar Limits The scale bar limits, *High* and *Low*, are used to configure the default lowest and highest scale numbers on the vertical axis of the trend display. These limits become the default limits which appear, in white, each time the display is called up or reset via the *Reset* button (which appears at the bottom of the display). If no limits are specified, the *High* limit defaults to 100.00 and the *Low* limit to 0.00.

Search Button The *Search* button is located on the *Select Trend* window, which is accessed by clicking on the *List* button on the *Trend Index Entry* window. Enter a trend index number in the index entry box, or an alphanumeric text string used in the title entry box, then click on *Search*. The trend with that index number or first instance of a trend containing that string in its title will be displayed and highlighted at the top of the list box.

Segment The horizontal axis of the trend graph is divided into five segments, each of which is denoted by a long tick-mark and a time and date stamp.

Segment Size The actual segment size is determined by two parameters: *Seconds* and *Point*s. The *Seconds* parameter is used to assign the number of seconds in a segment. By default, this parameter is

set to 60 seconds. The *Points* parameter determines the maximum number of trend points which can be plotted in a segment. By default, a maximum of 60 points can be plotted per segment. To position points on the trend graph, the PWC90 console divides the number of seconds in the segment by the number of points in the segment, yielding the number of time intervals in the segment. The default settings of 60 points and 60 seconds per segment thus yield the equivalent of a 1:1 compression ratio.

Trace A trace is a record of the collected data for an historical database point, presented in visual form on a trend display. Up to eight traces can be configured on each trend display, and each trace is displayed in a different color.

Trace Default State The trace default state is used to determine the default condition of the trace. A trace can be visible (ON) or invisible (OFF), by default, whenever the trend display is called up.

Trend Trace Limits Trend trace limits are the limits defined for each individual trend trace. They replace the default scale bar limits when the colored button associated with the trend trace is clicked.

Trend Title The title parameter is used to configure a trend display title of up to 48 characters in length. Once configured, the trend title appears at the top of the trend display. To configure this item, click on the entry box which follows the word *Title*; use the keyboard to enter the desired alphanumeric text string.

Trend Type The type parameter is used to determine which database (historical, archival) will be used to supply the data for the trend.

SECTION 8 - SYSTEM STATUS DISPLAY

INTRODUCTION

The system status display (SSD), shown in Figure 8-1, provides information about each node connected to the PWC90 console network. A node can be a PWC90 console or an INFI 90 OPEN server. Personal Work Center 90s (PWC90s) on the network are labelled as PWCs.

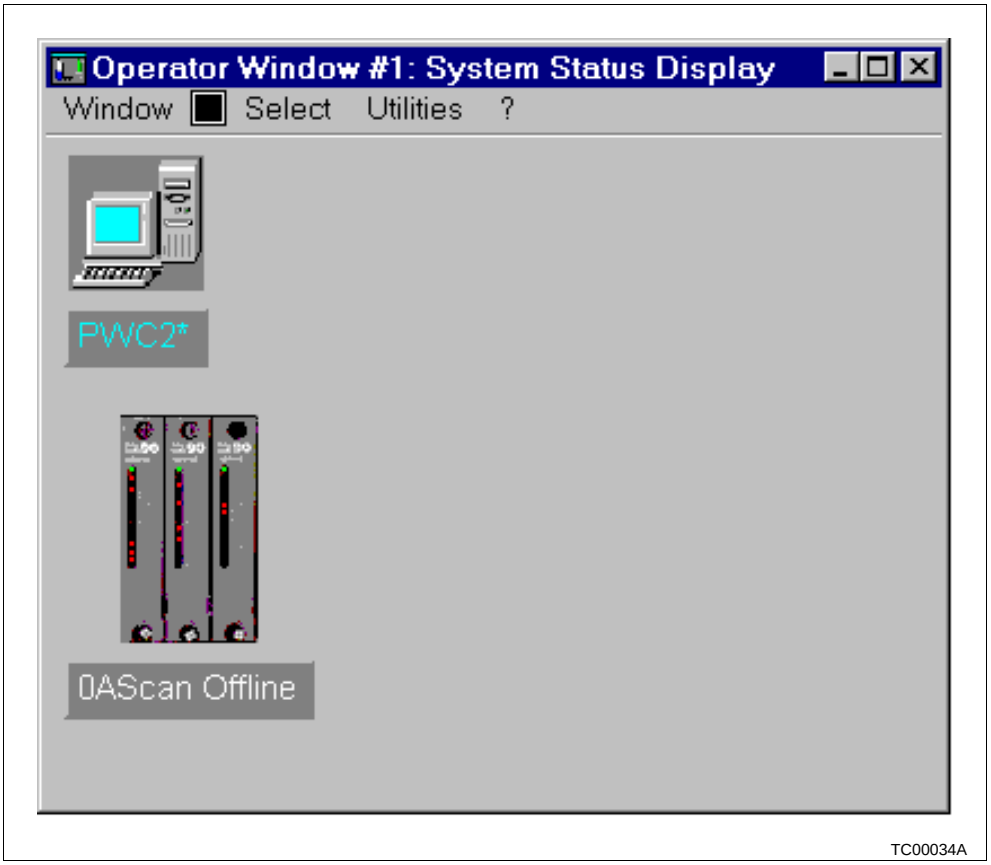


Figure 8-1. System Status Display

Each node on the network is depicted by a pictorial representation known as an SSD object.

An SSD object is composed of a schematic illustration of the node above a button containing the name of the node and information on the state of the node and its connections to the PWC90 console network. The appearance of the SSD object varies, depending on the type and state of the node. Click on the button to access a pull-down menu for the node represented by the SSD object. Like the appearance of the SSD object, the entries on the pull-down menu vary with the type of

node represented. Figure *8-2* is an example of the SSD object for a PWC90 console.

Figure 8-2. SSD Object for a PWC90 Console

The system status display is organized into rows of nodes, using the following rules:

1. A row contains nodes of only one type of device.

2. The rows of device types are ordered as follows, from top to bottom:

- Personal Work Center (labelled PWC).
- INFI 90 OPEN Server.
- other.

If no nodes of a particular device type exist, subsequent rows of other devices are moved up in the SSD (i.e., no blank rows are created).

3. A row can contain up to eight nodes. If more than eight nodes of one type exist on the network, they are displayed on multiple, consecutive rows.

4. Within a row, nodes are ordered left-to-right according to device number; i.e., PWC1 appears to the left of PWC2, and so on.

If more nodes exist on the network than can be shown in the normal viewing area of the SSD, scrollbars will appear at the right and bottom sides of the display. These can be used to view the portions of the SSD which extend beyond the normal viewing area.

If the operator window containing the SSD is positioned so that the SSD object of a node is near the right edge of the screen, it may be difficult to use the pull-down menu for the node in the

normal manner. There are two possible solutions to resolve this problem.

1. Resize or move the window so that the node is not at the right edge.

2. Use the bottom scrollbar to shift the contents of the window so that the node is not at the right edge.

GENERAL INFORMATION FOR ALL NODE TYPES

All SSD objects, regardless of node type, share certain features, including:

- Node name.
- Network connection information.
- *Info* menu.

These features are discussed in the following paragraphs.

Node Name

The *Node Name* is displayed on the face of the button which is located below the schematic illustration of the node (the node icon). The node name is composed of a three character descriptor which indicates the device type (i.e., PWC) followed by the device number (i.e., PWC2). The color of the node name indicates the network status of the node, as follows:

Cyan Indicates that the node is the local PWC90 node. The node icon of the local PWC90 node is also displayed in cyan.

> **NOTE:** The local node is also indicated by an asterisk (❇) following the node name and by the cyan color of the node icon.

White Indicates the node is on the network. The node can be accessed by the local PWC90.

Black (Text and Node Icon) on Red (Button Color) Indicates that the node is not currently on the network. The node can no longer be accessed from the local PWC90 console. (However, the node was accessible at some point in time following the most recent restart of the local PWC90. If the node is still inaccessible at the time of the next local PWC90 restart, it will be removed entirely from the SSD). The pull-down menu of the node cannot be used in this state. The node icon is displayed in black, as well as the node name.

> **NOTE:** In this case the node name is also enclosed in parentheses.

Info Menu

Click on the button containing the node name to display the pull-down menu for a node. The pull-down menu for each node

contains a button labeled *Info*. Click on this button to access a cascading menu titled *Info Menu*. This *Info Menu* contains, at the minimum, the following two entries:

- *Network Info*.
- *Show Available Services*.

NETWORK INFO

Clicking on the *Network Info* item in the *Info Menu* will cause a *Network Info* window (similar to the example in Figure 8-3) to be displayed.

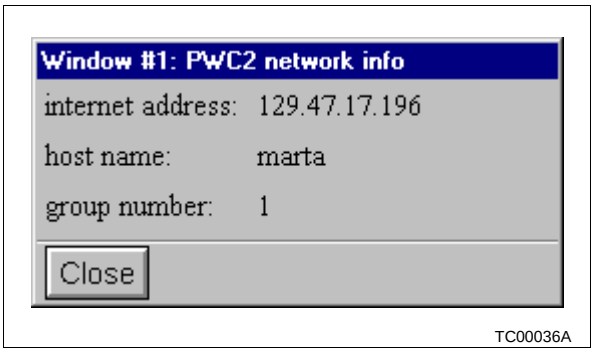


Figure 8-3. Network Info Window Example

This window contains read-only information about the node selected from the system status display. The information will include the Internet (IP) address, the host name (if any), and the ethernet address of the node (if known).

Click on *OK* to return to the system status display.

SHOW AVAILABLE SERVICES

Clicking on the second item in the *Info Menu, Show Available Services*, will cause a *Services* window (similar to the example in Figure 8-4) to be displayed.

This window contains a read-only list of the services available on the node selected from the system status display. This information is used by technical support personnel during trouble-shooting. If a node function does not work as expected, check this list to find out if the service is actually available on the node. Note that each data transaction on the network is preceded by a code number identifying the service associated with the transaction. These code numbers are shown in brackets preceding each item on the list of available services.

Click on *Close* to return to the system status display.

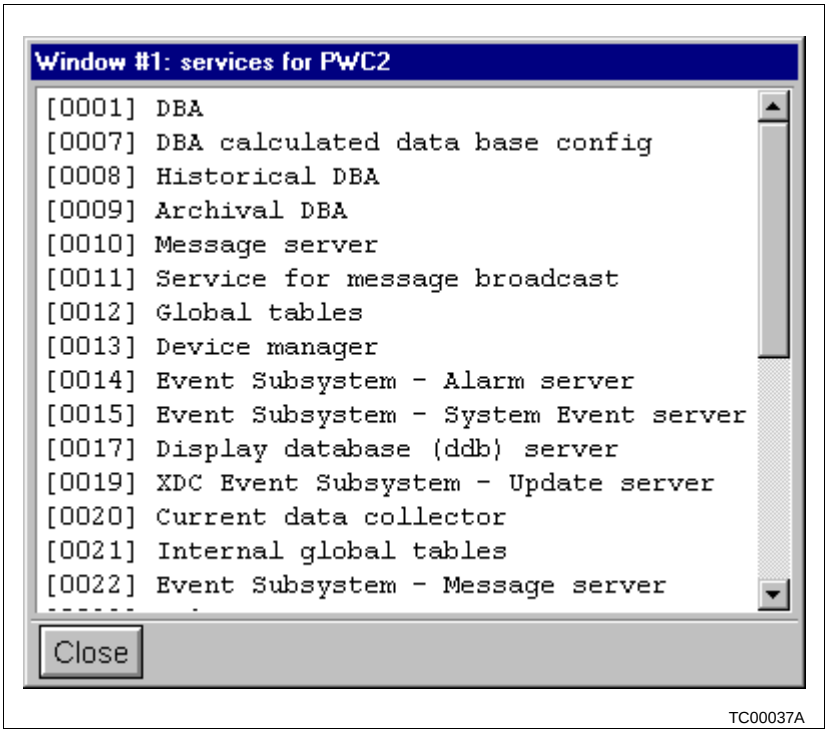


Figure 8-4. Services Window Example

INFORMATION FOR SPECIFIC NODE TYPES

This following paragraphs describe the information shown on SSD objects which is specific to individual node types.

PWC90 Nodes

The SSD object for an PWC90 is shown in Figure 8-5. The pull-down menu contains only the *Info* item.

Figure 8-5. PWC90 Node SSD Icon

INFI 90 OPEN Server Nodes

The INFI 90 OPEN server node provides access to the INFI 90 OPEN network of up to 250 rings, each of which may have up to 250 nodes of 31 process modules (Figure 8-6). Multiple servers may access the same INFI 90 OPEN network but must use unique tagnames. The server button gives access to a pull down menu of available services.

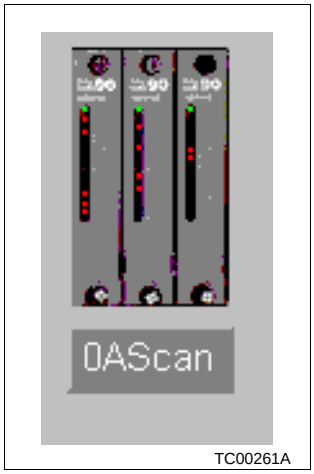


Figure 8-6. SSD Object for a Server

- Info.
- Server configuration.
- Server diagnostics.
- Server tag summary.
- INFI-90 block details.
- INFI-90 module status.

The Info option displays a pop-up window that provides information on the server such as:

- Type of server.
- Server host.
- Server name.

Server Configuration invokes the ***Tag Configuration*** application. For more information on this application refer to Section 12.

Server Diagnostics invokes the *Server Diagnostics* application. For more information on this application refer to ***Personal Work Center 90 Operation*** instruction.

Server Tag Summary invokes the ***Tag Summaries*** application. For more information on this application refer to ***Personal Work Center 90 Operation*** instruction.

INFI-90 Block Details invokes the ***Block Details*** application. For more information on this application refer to ***Personal Work Center 90 Operation*** instruction.

INFI-90 Module Status invokes the ***Module Details*** application. For more information on this application refer to ***Personal Work Center 90 Operation*** instruction.

UTILITIES MENU

The *Utilities* option, located on the system status display menu bar, provides access to printer device status information and printer control utilities. To access these functions, click on the *Utilities* item on the menu bar; a pull-down menu containing the *device status* option will appear.

Device Status

A third item, *Device Status*, is available from the *Utilities* option on the *System Status Display* menu bar. Click on this item to display the *Device Status* window (Figure 8-7).

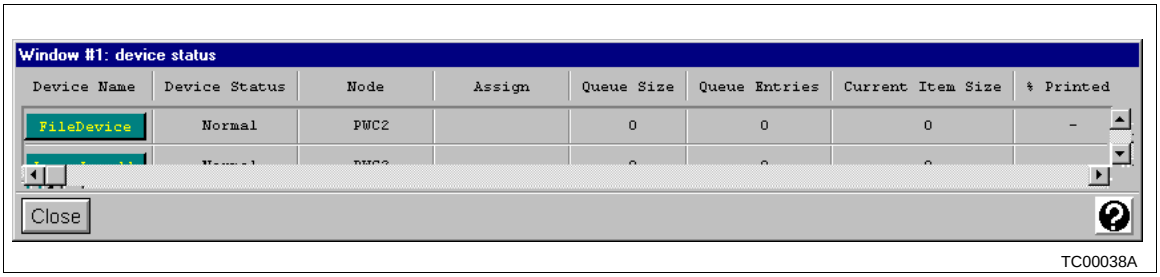


Figure 8-7. Device Status Window

This window provides information about items being printed and access to job and printer control. The following information is provided for each device:

- Device name.
- Device status.
- Node.
- Active backup unit name (Assign).
- Queue size.
- Number of queue entries.
- Size of current job.
- Percent of current job printed.

PRINTER QUEUE CONTROL

The column labelled *Device Name* contains a button for each device which has been configured via the *Printer Assignment* window (refer to ***Printer Assignment*** in Section 2). The name

of the device appears on the face of the button. The color of the button indicates the status of the device:

Green Indicates the device is operating normally.

Red Indicates the device has failed.

Yellow Indicates the device has been placed in suspend mode.

> **NOTE:** When the button color changes to indicate the device status, the text on the button also changes color so that it remains visible. The text color has no other significance.

In addition to button color, the *Device Status* field to the right of each *Device Name* button provides status indication in text form, i.e., it will display the word *Normal, Failed* or *Suspended*.

When the *Device Name* button is pressed, the *Printer Queue Control* window (Figure 8-8) will be displayed.

The first item in the *Printer Queue Control* window is the name of the device. Below this identifier the window is divided into two main sections. The first of these is labelled *Cancel* and contains two buttons:.

- *Current Item*.
- *All Items*.

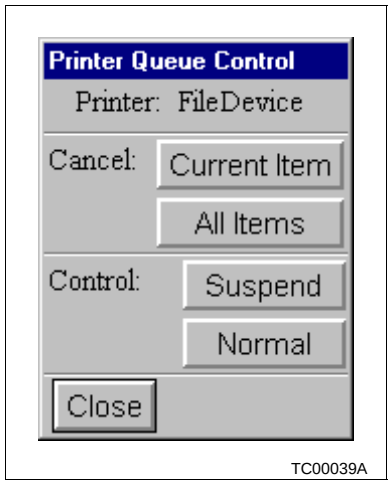


Figure 8-8. Printer Queue Control Window

Click on *Current Item* to terminate the job which is active in the device. Click on the *All Items* button to terminate the active job and delete all other entries from the queue for that device.

Cancelling a job removes its entry from the print queue; however, any portion of the job which has already been loaded into buffer space (in any relevant device, such as the printer, the PWC90 console, and so on) will be printed. Therefore, it is not

unusual for a few pages of a job to print out after the job has been cancelled.

The next portion of the window is labelled *Control*, which also contains two buttons:

- *Suspend*.
- *Normal*.

Click on *Suspend* to suspend operation of the device (*Suspend* mode). When this action is performed, processing of any jobs in the queue will be completed, but no new jobs will be accepted in the queue. Click on *Normal* to return to normal operation.

Click on *Close* to close the *Printer Queues Control* window.

PRINTER QUEUE INFORMATION

The *Node* field on the *Device Status* window displays the name of the node which is currently serving the print device.

When a device is operating normally, nothing is displayed in the *Assign* field. When a device fails, the *Assign* field will display the device name of the backup unit which is actually servicing the queue for the failed device. Backup device assignments are configured via *Printer Assignment* (refer to ***Printer Assignment*** in Section 2).

The *Queue Size* field displays the number of bytes in all jobs in the device print queue, while the *Queue Entries* field indicates the number of jobs in the queue. The *Current Item Size* indicates the size, in bytes, of the job which is currently active. The *% Printed* field indicates what portion of the currently active job has been completed.

SECTION 9 - PWC90 UTILITIES

INTRODUCTION

Personal Work Center 90 utilities provide a variety of useful maintenance and information management functions, including:

- Backup and restore functions.
- Archiving historical and historical block data.
- Restoring archival and archival block data.
- Database maintenance.
- Manual data entry.
- Alarm groups.
- PWC90 console screen printouts.
- Release of configuration locks.
- Scientific calculator.
- Other applications access.

The PWC90 utilities are accessed via the system window *Menu*. To do so, click on system window *Menu*, then click on *Utilities* on the resulting pull-down menu. The *Utilities* menu will then be displayed (Figure 9-1).

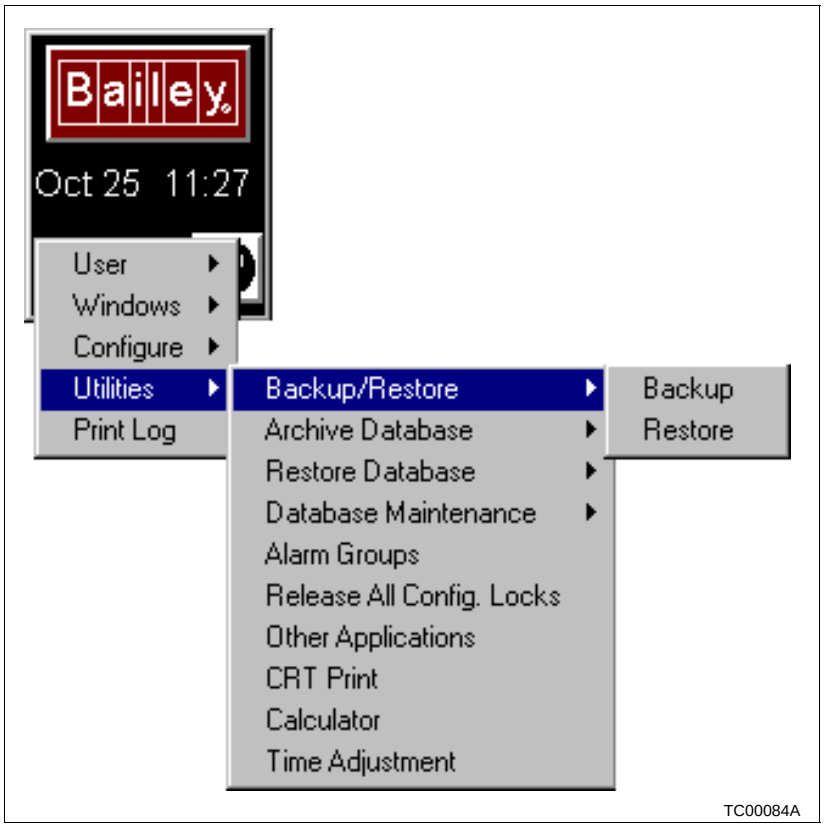


Figure 9-1. PWC90 Utilities Menu

Most of these utilities do not relate to configuration operations and are discussed in the ***Personal Work Center 90 Operation*** instruction. Several items which may be of use to the configurer, *Database Maintenance, CRT Print and Release All Config. Locks*, are discussed below. The on-screen scientific calculator works like a pocket scientific calculator; instructions for its use are outside the scope of the PWC90 documentation, consult the Windows NT documentation.

A number of utility-type features of the PWC90 console are also described here, although they are not accessed via the *Utilities* menu. These include:

- Quick keys.
- CRT context keys.
- PWC90 console help.
- Process help.

DATABASE MAINTENANCE

The database maintenance menu item provides access to the following utilities:

- Historical manual entry (which is described in the ***Personal Work Center 90 Operation*** instruction).
- Historical database maintenance.
- Historical block database maintenance.
- Archival database maintenance.
- Archival block database maintenance.

Historical Database Maintenance

This feature displays read-only information about historical database disk usage. This information can be of value when assessing whether the collection rate configured for a point is causing the point to use an undesirable amount of disk space.

To access historical database maintenance click on system window *Menu*, then click on *Utilities* in the resulting pull-down menu. The *Utilities* menu will then be displayed. Click on *Database Maintenance* in the *Utilities* menu to access the cascading menu shown in Figure 9-2.

The *Database Maintenance* menu is used to perform manual data entry as described in the ***Personal Work Center 90 Operation*** instruction, and to access database maintenance windows for the historical, historical block, archival and archival block databases. The *Historical Database Maintenance Window* is shown in Figure 9-3.

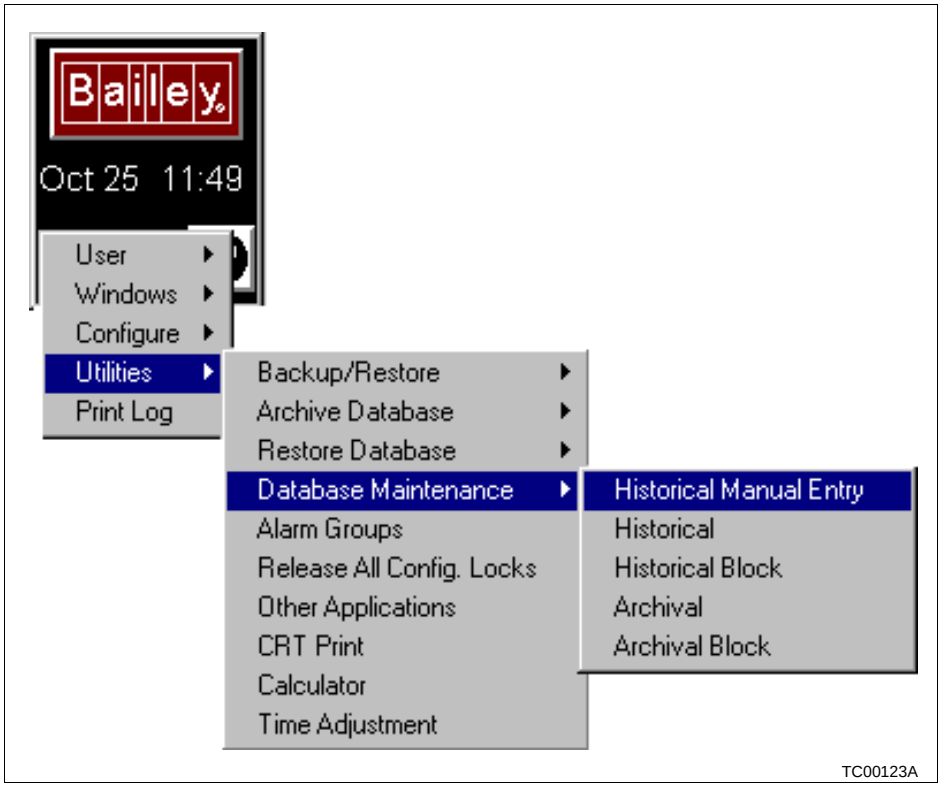


Figure 9-2. Database Maintenance Menu

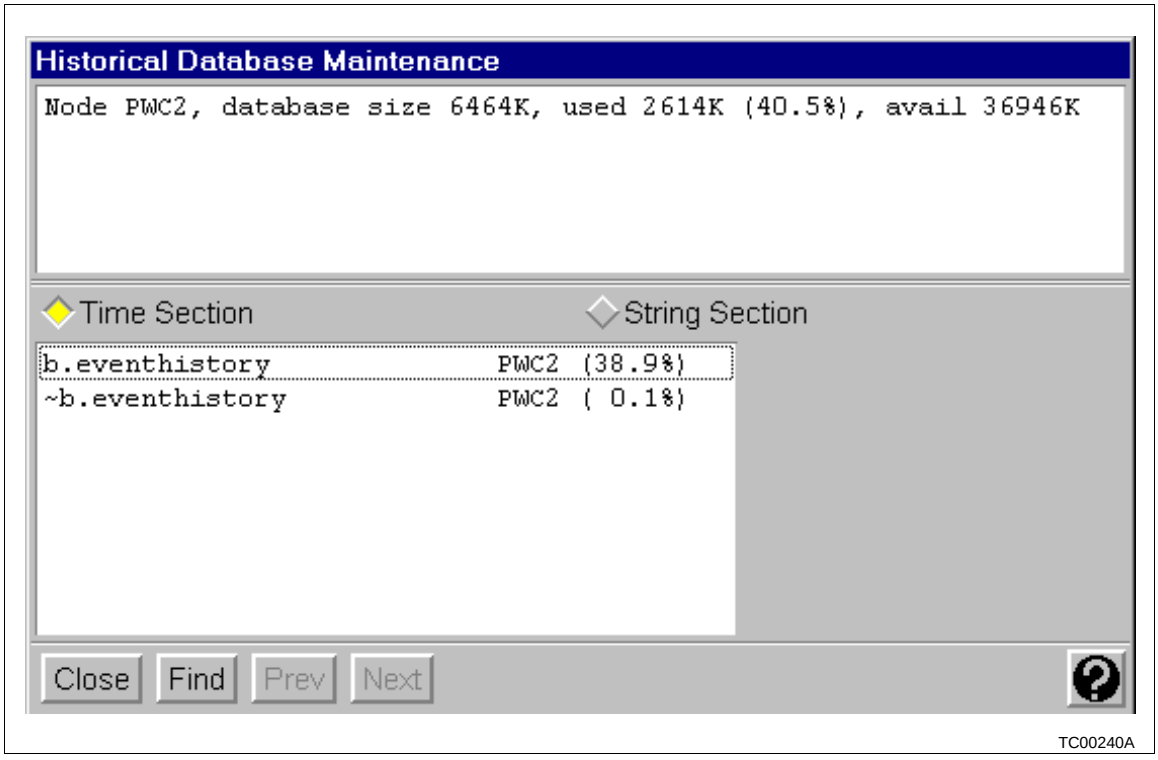


Figure 9-3. Historical Database Maintenance Window

At the top of the window is a list box which contains entries describing database usage on each node on the PWC90 console network. Each entry includes:

- Node name.
- Size of the disk space allocated for the database in kB.
- Number of kB of the allocated space actually used (and the percentage).
- Total number of kB available on the hard drives of the node.

Note that as the historical database expands, additional disk space will be allocated for the database. However, once allocated, space is never removed from the historical database allocation.

Data which is collected and stored in databases on the PWC90 console is keyed (internally) for identification within a database by either a time or a string. This key is assigned automatically by the PWC90 console software application which is performing the collection operation; it cannot be specified by the user. For example, historical data and archival data are keyed by time, while historical block data and archival block data are keyed by strings. Other applications can use both types of key to store data.

While the distinction between time keyed data and string keyed data is generally invisible to the user, the historical database maintenance window uses this feature. Information is displayed in the tagname list box in the lower part of the window based on the type of key which is selected.

Radio buttons located above the tagname list box are used to select the key type to be used. When the *Time Section* radio button (the default) is selected, all the tags which are keyed by time will be listed; click on the *String Section* radio button to list all the tags (if any) which are keyed by strings.

> **NOTE:** The tagname list box lists historical database points in every historical database on the PWC90 console network (not just the local node).

The data contained in the list box at the top of the historical database maintenance window provides an overview of the system usage by the historical database on each node. The tagname listbox in the lower part of the window provides information on system usage by individual tags. Data displayed here includes:

- Tagname (preceded by an *s*).

- Node or nodes where the point is part of the historical database.

- Percentage of the node's historical database used to store that point.

This information can be used to fine-tune the data collection process to make the most efficient use of resources. For example, to view which collection items are using the most disk space, on which systems those collections are being stored, and where potential configuration changes could be made to balance the storage of data.

Following the list of standard time-based tagnames (preceded by S) the list contains an entry ***b.eventhistory***. This represents the event historian information which is also stored in the historical database on the hard disk of each PWC90 console on the system.

Following the ***b.eventhistory*** entry the window contains another alphabetical list of the tagnames in the historical database on the system. These tagnames are preceded by the tilde character (~). The percentage displayed for a ~tagname represents the space used to store special information used by the system software to synchronize the historical database when the system comes back up after a shutdown.

The event historian information is stored in a circular file; the size of the file is preset by the system software. Therefore, no user actions can reduce the space allocated for the event historian information. However, the information about disk usage for each historical database point can be used by configurers when considering whether they can afford to store points at various collection rates.

Locating Specific Historical Tags

If more historical database points have been configured than can appear in the historical tag list box, the *Prev* and *Next* buttons will become available. Click on *Next* to display the historical tags which follow those currently on display in the list box. Click on *Prev* to display the historical tags which precede those currently on display in the list box.

The *Find* button, located at the bottom of the *Historical Database Maintenance* window is used to locate a given point in the historical database. If the point does exist, the *Find* button quickly locates it in the *Historical Tag* list box and highlights.

To use this feature, click on *Find*. A pop-up window appears, which contains an entry box, and *OK*, *Apply* and *Cancel* buttons. Click on the entry box to focus on it, then enter the desired historical tagname. If only a portion of the tagname is

entered, the first entry where the first letters of the tagname match the entered string will be chosen. For example, if the string HR is entered, and the first tagname to begin with these characters is HRANO1, then this is the tagname that would be highlighted in the *Historical Tag* list box.

Click on *Apply* to locate the first instance of the string; the *Find* pop-up window will remain open; click on *Apply* again to locate the next instance of the string, or enter a new string to be located. Click on *OK* to perform the *Find* operation and close the *Find* pop-up window. Click on *Cancel* to exit without attempting to find the tagname in question, or to close the pop-up window.

Historical Block Database Maintenance

The historical block database provides for efficient collection of data from a large number of INFI 90 OPEN server tag.atoms. Historical block data collection is discussed in Section 10.

The historical block database maintenance window is used to display read-only information about historical block database disk usage. This information can be of value when assessing whether the block collection configuration is causing the block to use an undesirable amount of disk space.

To access *historical block* database maintenance click on *Historical Block* in the *Database Maintenance* menu shown in Figure 9-2. The features in this window are used as described in ***Historical Database Maintenance*** in Section 9.

Archival Database Maintenance

The archival database is the storage location for historical database information restored to the system from archival tapes. The archival database maintenance utility is used to display information about archival database disk usage, and to delete information from the archival database. Note that this is the only way that information is ever removed from the archival database. It is recommended that the archival database be assessed periodically to determine whether information in the archival database is still needed; if not, delete it. This will avoid wasting disk space. As the archival database expands, additional disk space will be allocated for the database. However, once allocated, space is never removed from the archival database allocation unless the *Clear Database* button is used as described below.

To use the utility click on *Archival* in the *Database Maintenance* menu (Figure 9-2). The archival database maintenance window shown in Figure 9-4 will be displayed.

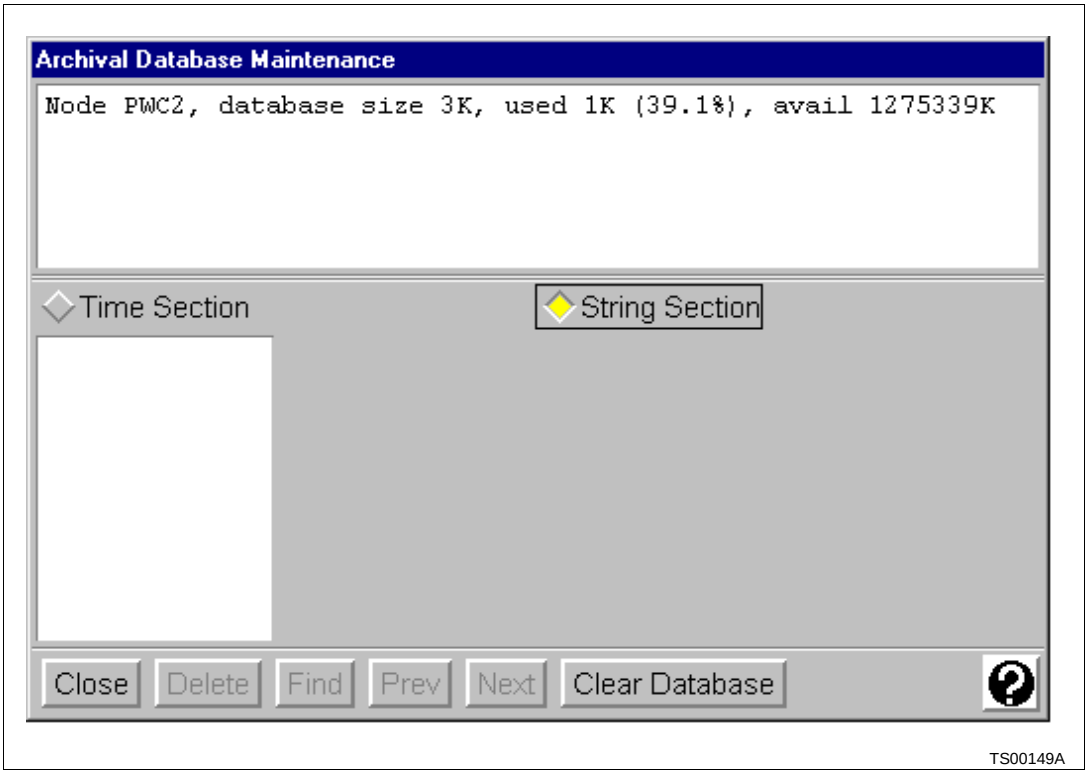


Figure 9-4. Archival Database Maintenance Window

The list box at the top of the window includes the following information:

- Node name.
- Size of the disk space allocated for the database in KB.
- Number of KB of the allocated space actually used (and the percentage).
- Total number of KB available on the node's hard drives.

Archival data is keyed (internally) by time within the archival database, so the *Time Section* radio button (below the list box containing disk usage information) should be selected. (Refer to ***Historical Database Maintenance*** in Section 9 for a more detailed discussion of the *Time Section* and *String Section* radio buttons). This will cause the tagname of every archival database point in every archival database on the system (not just the local node) to be listed alphabetically in the tagname list box. The information displayed includes:

- Tagname preceded by an *s*.
- Node or nodes where the point is part of the archival database.

- Percentage of the node archival database used to store that point.

The *Archival Database Maintenance* window also contains the following buttons: *OK*, *Delete*, *Find*, *Prev*, *Next* and *Clear Database*.

LOCATING SPECIFIC ARCHIVAL TAGS

If more archival database points have been configured than can appear in the archival tag list box, the *Prev* and *Next* buttons will become available. Click on *Next* to display the historical tags which follow those currently on display in the list box. Click on *Prev* to display the archival tags which precede those currently on display in the list box.

The *Find* button, located at the bottom of the *Archival Database Maintenance* window is used to locate a given point in the archival database. If the point does exist, the *Find* button quickly locates it in the archival tag list box and highlights.

To use this feature, click on *Find*. A pop-up window appears, which contains an entry box, and *OK*, *Apply* and *Cancel* buttons. Click on the entry box to focus on it, then enter the desired archival tagname. If only a portion of the tagname is entered, the first entry where the first letters of the tagname match the entered string will be chosen. For example, if the string HR is entered, and the first tagname to begin with these characters is HRANO1, then this is the tagname that would be highlighted in the *Archival Tag* list box.

Click on *Apply* to locate the first instance of the string; the *Find* pop-up window will remain open; click on *Apply* again to locate the next instance of the string, or enter a new string to be located. Click on *OK* to perform the *Find* operation and close the *Find* pop-up window. Click on *Cancel* to exit without attempting to find the tagname in question, or to close the pop-up window.

DELETING ARCHIVAL DATA

To delete all data for a point from the archival database on all nodes, click on the tagname to select it, then click on *Delete*. All data for that point in the local node archival database will be deleted. The space allocated for the point will not be released.

To delete all data from the archival database on all nodes and release the space allocated for it, click on *Clear Database*. A confirmation window will open containing the message:

Database will be cleared.
Are you sure?

To abort the operation click on *No* in the confirmation window. To delete all the archival data stored on all nodes, click on *Yes* in the confirmation window. The data will be deleted and the window will close.

Archival Block Database Maintenance

The archival block database is the storage location for historical block database information restored to the system from archival tapes. The archival block database maintenance utility displays information about archival block database disk usage. This information can be of value when assessing whether the block collection configuration is causing the block to use an undesirable amount of disk space.

To access archival block database maintenance click on *Archival Block* in the *Database Maintenance* menu shown in Figure 9-2. The features in this window are used as described in ***Archival Database Maintenance*** in Section 9.

CRT PRINT

The *CRT Print* selection on the *Utilities* menu is used to print the screen content (or portions thereof) of the PWC90 console. Click on this item to display the *CRT Print Setup* window (Figure 9-5).

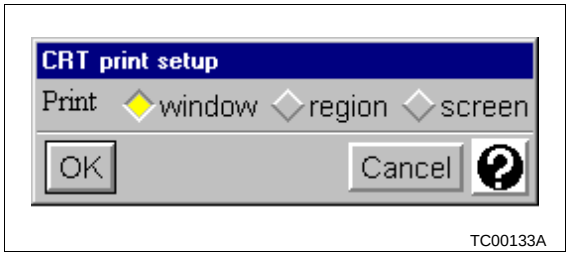


Figure 9-5. CRT Print Setup Window

Selecting the Image Type

The first area of the *CRT Print Setup* window is labelled *Print*; this label is followed by three choices, *Window*, *Region*, and *Screen*. Each choice is preceded by a radio button, as only one of these items can be selected at a time.

Click on the *Window* radio button if the image is to be defined by the borders of a selected window. Click on the *Region* radio button if the image is to be a portion of the screen defined by the user. Click on the *Screen* radio button if the image is to contain the entire contents of the screen.

Selecting the Printer

The second item in the *CRT Print Setup* window is a button labelled *Printer*. Click on this button to display the printer selection window, which contains a list of available laser and CRT color copy printers.

Printing the Image

After the image type and printer are set to the desired values, click on *OK*. The *CRT Print Setup* window will close.

If the *Screen* image type was selected, no further action is necessary.

If the *Window* image type was selected, the pointer changes to a crosshair shape (✣). Move the pointer to any location within the window to be printed, then click the left mouse button.

If the *Region* image type was selected, the pointer changes to an upper left corner shape. Move the pointer to the screen location which corresponds to the upper left corner of the region to be printed, then press and hold the left mouse button. The pointer changes to a lower right corner shape. Drag the pointer to the lower right corner of the region and release the mouse button.

RELEASE ALL CONFIGURATION LOCKS

When certain configuration operations are in progress, software locks are set to warn other users who may attempt configuration of the same item simultaneously (i.e., User 2 attempts to configure Summary 10 when User 1 has already begun configuring Summary 10). This applies to the following configuration operations:

- Graphics (locked by model name).
- Summaries (locked by index number).
- Trends (locked by index number).

Security functions, including:

- Function access.
- Default configuration.
- User login.
- Security activation.

NOTE: All groups in a summary are locked when any one group in that summary is being configured.

Normally a message is displayed warning the second user that configuration of the item is already in progress by another, and providing an option to override the configuration lock.

Click on *Yes* to go ahead and override the lock, or click on *NO* to exit without attempting to configure the item. Click on the question mark (*?*) to exit the warning window and open a window which lists the other users currently engaged in configuration of the item. The information provided about these users includes the node they are logged in on, the PWC90 console and time they commenced configuration.

If more than one user has an item open for configuration, the only changes which will take effect will be the set of changes made by the last user to save the item.

If an abnormal condition arises while a configuration operation is in progress (a PWC90 console shutdown as the result of a power failure, for example), the configuration lock will remain set when the system is recovered. The last item on the *Utilities* menu, *Release All Config. Locks*, is used to clear all such lost configuration locks. Click on this menu item to clear all locks which have been imposed on configuration on the network. Under normal conditions, use of this utility should not be necessary.

PWC90 INFI-90 TIME SYNCHRONIZATION

A *PWC90 System Time Adjust* dialog can be launched from the PWC utility menu (Figure 9-6), which will allow the user to adjust system time to a maximum of plus or minus five minutes.

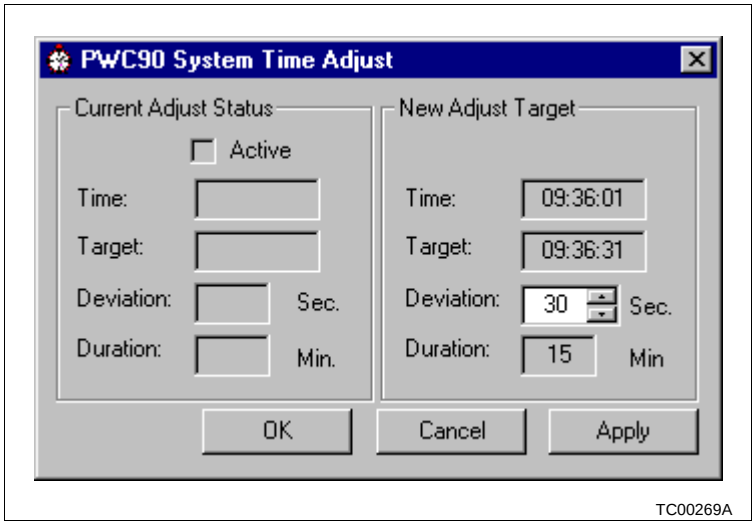


Figure 9-6. PWC90 System Time Adjust Window

Current Adjust Status

The *Current Adjust Status* section of the dialog provides information to the user about the current state of the system time

service. All fields in this area will be dimmed when there is no time adjust operation in progress.

The *Active* check box will be blank when no time adjust operation is in progress. The *Time* field provides the user with the current system time. The *Target* field provides the user with the target time of the current adjustment operation. The *Deviation* field provides the user with the current system time deviation from the current target time. The *Duration* field will provide the approximate length of time, in minutes, needed to complete the time adjustment operation. It will be based on the current deviation and the fixed time adjustment rate of two seconds of adjustment per minute. These fields will be adjusted very half a second when an adjustment is active, otherwise they will be blank.

New Adjust Target

The *New Adjust Target* section of the dialog provides information the user needs to issue a new time adjustment to the system.

The *Time* field provides the user with the current system time. The *Target* field provides the user with the target time of the new adjustment operation. These two fields will be updated every half a second. The *Deviation* field will be used to enter a time adjustment, in seconds, into the system. Changing the deviation will result in the target and duration fields being updated to indicate the affect of the proposed change prior to committing it. This field will be limited to plus or minus 300 seconds. The *Duration* field will provide the approximate length of time, in minutes, needed to complete the new time adjustment operation. It will be based on the current deviation of the fixed time adjustment rate of two seconds of adjustment per minute. This field will be updated every time the user changes the deviation field.

Three buttons are provided at the bottom of the dialog. Use the *Apply* button to accept the deviation request in the new deviation field. The application will issue a request to the PWC time server to adjust the time to the new settings. The application will then issue a notification to each PWC90 server to indicate the start of a time adjustment. This will include all servers in all sub-domains. This button will be dimmed and inactive if the current user does not have Database Configuration access. Use of the *OK* button will perform the functions defined for the Apply button, and will then close the dialog. This button will be dimmed and inactive if the current user does not have Database Configuration access. The *Cancel* button can be used to close the dialog without accepting the new deviation settings.

OTHER APPLICATIONS

The *Other Applications* option on the *Utilities* menu is used to access and run standard and (if available) custom application programs. Selecting the *Other Applications* option causes the Windows Explorer to be displayed. Consult the Windows NT on-line help for more information on the Windows Explorer.

ACCESSING OTHER APPLICATIONS

To access this item, click on the system window *Menu* button, then click on *Utilities* in the resulting pull-down menu. The *Utilities* menu will then be displayed. Click on *Other Applications* in the *Utilities* menu. The Windows Explorer window will be displayed.

PWC90 HELP

The PWC90 help is context sensitive help file system. When a help button is clicked in any window, the help system opens an information window describing the function of that window or related functions. After the help system has been opened it will operate similar to a Windows NT help file.

QUICK KEYS

The PWC90 console quick keys feature provides each user with a personalized, pop-up keypad containing from 32 to 64 keys for each operator window. Each of these keys can be configured to call up a different display within an operator window. Because the set of quick keys is unique for every user, it can be customized to fit each user's individual needs.

Quick keys operate within operator windows, allowing the user to call up a specific pre-configured display by clicking on a single button. This can save a significant number of keystrokes, and is ideal for use with frequently-viewed displays. By default, the parent operator window is the recipient of the new display, although drag and drop operation can be used to cause the display to appear in a different operator window. Alternatively, the same quick keys keypad can be displayed simultaneously in all operator windows on the screen. The quick keys keypads can remain on display indefinitely, and can be used as often as desired to view the various displays available via quick key assignments. The keypads can be moved and resized to suit the needs of the user and the size of the parent window.

Accessing the Quick Keys Feature

To access the quick keys feature, the user must be in an operator window with keyboard focus. The *Quick Keys* pop-up keypad can be summoned using either the mouse or the PWC90

console keyboard. Using the mouse, press the *Select* button on the menu bar of the window. From the resulting cascading menu, select the *Quick* option. Selecting the *Quick* option will cause a pop-up window to appear, as shown in Figure 9-7.

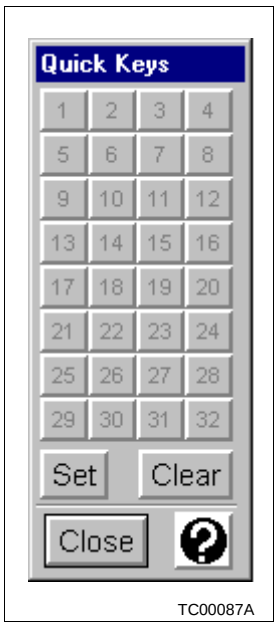


Figure 9-7. Default Quick Keys Keypad Display

If quick keys have been configured, position the pointer on the desired quick key, and click the left mouse button. The display assigned to that quick key will be displayed by default in the operator window which is the parent to the quick key keypad. To have the display appear in an operator window other than the parent window, position the pointer on the desired quick key, then press and hold the left mouse button down, while dragging the pointer to a location inside the frame of the operator window which is to contain the display. When the mouse button is released, the display will be dropped into that operator window.

> **NOTE:** A minimum of 32 quick keys are displayed. If more than 32 quick keys have been configured, the quick key with the highest number is the last quick key on the display. Unconfigured quick keys between 1 and the highest number quick key are unavailable for use at this time, and will appear with stippled identifying numbers.

Quick keys can also be selected via the keyboard. To do so, the *Quick Key* pop-up window must have keyboard focus. A quick key can then be selected by using the arrow keys to highlight (move black border to) the desired key; press the *Return* key to complete the selection.

To delete the quick key keypad from the display, position the pointer on the *Close* button, and click the left mouse button.

Configuring Quick Keys

To configure quick keys, the user must be in an operator window with keyboard focus. In this window, call up the display which is to be assigned to a quick key. Click on *Set*; stippling now appears on previously assigned quick keys, protecting them from accidental re-assignment; all other quick keys are available for assignment. The pointer assumes the shape of a plus sign (✛). Click on the quick key which is to be used to call up the display which appears in the operator window. A new pop-up window is displayed, as shown in Figure 9-8.

Figure 9-8. Quick Key Label Assignment Window

This window allows the user to enter a text string of up to six characters in length. This text will appear on the quick key. The default text string is the number of the quick key (1 to 64); the user can choose to discard this numeric identifier. By using mnemonic identifiers and identifying the quick keys in this manner, the user can see at a glance how keys are assigned. For an example, Figure 9-9.

To exit *Set* mode without making an assignment, press the *Set* button a second time; the quick keys keypad will remain on display, allowing the user to use or make further changes to the quick keys.

Once a quick key assignment has been made, the user need only access the quick keys keypad display and press the assigned quick key to call up that display in any operator window with keyboard focus.

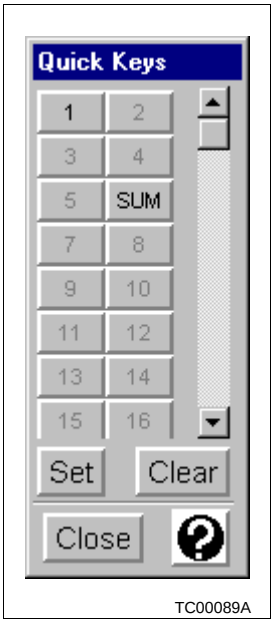


Figure 9-9. Quick Key Pop-up with Text String Assignments

Configuring Quick Keys 33 Through 64

The basic quick key pop-up window contains a minimum of 32 buttons by default. Once all 32 buttons have been assigned, an additional 32 buttons can be configured, one at a time, for a total of 64 quick keys. When quick keys which will appear outside the current *Quick Key* pop-up (i.e., quick key 33) are configured, scroll bars appear on the *Quick Key* pop-up window. Use the scroll bars to view the newly created quick keys. After the *Quick Key* pop-up has been closed, then re-opened, all configured quick keys will be included in the display. The size and shape of the *Quick Keys* pop-up window can be changed using the resize handles on the window's frame.

If more than 32 quick keys have been configured, and any of them have been cleared, no new quick keys will be created until assignments for all the existing ones have been configured. For example, if 40 quick keys are assigned, then quick keys 25 through 30 are cleared, all 40 will remain on display, but new assignments must be configured for quick keys 25 through 30 before quick key 41 becomes available.

Clearing Quick Key Assignments

Once a quick key has been assigned to a display, it cannot be reassigned to a different display until it has been cleared. To clear a quick key click the *Clear* button, then click the quick key which is to be cleared. The cleared quick key will then be available for assignment to a different display. To exit *Clear* mode without clearing a quick key, click the *Clear* button

again; the quick keys keypad will remain on display, allowing the user to use or make further changes to the quick keys.

Exiting Quick Keys

To exit quick keys and remove the keypad from the display, click on *Close*.

Summary - Quick Keys

Quick Keys The *Quick Keys* selection offers each user a personalized set of 32 to 64 keys, each of which can be assigned to call up a different display in the active operator window. This saves time and keystrokes, and is ideal for accessing frequently-viewed displays.

Close The *Close* button on the quick keys keypad display is used to remove the keypad display from view.

Clear The *Clear* button on the quick keys keypad display is used to clear quick key assignments. To do so, first click on *Clear*. Next, click on the quick key to be cleared. The previous display assigned to the key will be cleared, and the key is then available for re-assignment.

Help The help icon on the quick keys keypad display provides a description of the quick keys feature.

Set The *Set* button on the quick keys keypad display is used to make unused quick keys available for assignment. To do so, click on *Set*. Quick keys which have already been assigned are unavailable, and are stippled out. Click on the desired quick key, then enter a label, if desired, in the resulting pop-up window. Click on *OK* or press Return to complete the operation. The chosen quick key is then assigned to call up the display currently in the operator window. Click on *Cancel* to close the *Label Assignment* pop-up window without changing the label on the quick key.

CRT CONTEXT KEYS

Each user can set up a personalized working environment or context, which can include multiple operator windows and their contents, as well as the size and position of quick key keypads associated with the operator windows. Different users have different contexts, depending on their area and responsibilities, and each user may have a number of different contexts which are used repeatedly. Using CRT context keys saves time and effort by bypassing the numerous keystrokes necessary to open multiple operator windows and call up the appropriate displays.

Preparing the Context

The size, location and contents of each operator window on a screen can be stored and assigned to a CRT context key. In order to assign this information, the user must first set up the working environment on a PWC90 console. For example, a user might have two windows open on a screen: one in which a point is being controlled and one containing an associated graphic. By assigning this set of displays to a CRT context key, the user can return to this screen environment with ease by selecting the appropriate CRT context key.

Initial Display Layout - Login Context Key

The login context key is assigned to the user's initial display layout, which appears automatically each time the user logs in. If a user has no context assigned to the login context key, the previous contents of the screen remain after the user logs in. In the case of the *Default User* state (no user logged in) the initial display layout consists of one operator window containing the system status display. The user can change the contents of this window and/or increase the number of windows in the *Default User* initial layout; clearing this layout will always cause it to return to the single operator window/system status display combination.

Accessing CRT Context Keys

CRT context keys allow each user to save up to 17 working contexts (the login context key plus 16 additional keys) and assign each context to one individual key which appears on the screen. To access the CRT context keys option, first click on the system window *Menu* button, then click on *Windows* in the resulting pull-down menu. A second menu will cascade from the first, containing the *CRT Context Keys* option. Click on *CRT Context Keys* a window known as the *CRT Context Keys* keypad will be displayed. This window contains the set of CRT context keys available in the current user's login configuration (Figure 9-10).

Figure 9-10. CRT Context Keys Keypad

Configuring CRT Context Keys

When the *CRT Context Keys* keypad is first called up, any CRT context keys which have not been assigned appear stippled, as they are not yet available to the user. Normal alphanumeric text appears on any *CRT Context Keys* which have been assigned a context.

The *Set* button allows the user to assign the displays that are currently on the CRT to any one of the CRT context keys. (Any pop-up windows which are present are ignored).

Position the pointer on the *Set* button and click the left mouse button. CRT context keys which have already been assigned now appear stippled, and unassigned keys are available. Position the pointer on the key to be assigned, and click the left mouse button. A pop-up window will be displayed, as shown in Figure 9-11.

> **NOTE:** To exit Set mode without making an assignment, click on *Set* a second time.

The user can enter a text string of up to eight alphanumeric characters, which will appear on the key. The key number is the default string, and can be changed by the user. Click on *OK* to complete the selection procedure, or click on *Cancel* to exit without making changes.

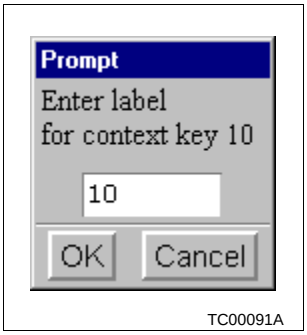


Figure 9-11. CRT Context Key Assignment Window

Clearing CRT Context Keys

CRT context keys must be cleared before reassignments are made. To clear a CRT context key, click on *Clear*. Next, click on the CRT context key which is to be cleared. All CRT context assignments associated with that key will be destroyed, and the key is available for a new assignment.

NOTE: To exit Clear mode without clearing an assignment, click on *Clear* a second time.

Exiting CRT Context Keys

The CRT context keys keypad will remain on the screen until the user clicks on the *Close* button. If the user elects to have this display remain on the screen, the CRT context keys keypad can be moved to a convenient screen location by positioning the pointer on the title bar, pressing the left mouse button, then dragging the display to the desired location. The CRT context keys keypad may be obscured by another window when keyboard focus is changed. It can be raised to the top again by clicking on and giving keyboard focus to its parent, the system window.

Summary - CRT Context Keys

CRT Context Key

A CRT context key is any one of the 16 pushbuttons on the CRT context keys keypad, used to recall a pre-configured screen environment. The CRT context keys keypad is displayed by:

1. Clicking on the system window *Menu* button.

2. Clicking on the *Windows* item in the resulting pull-down menu.

3. Clicking on the *CRT Context Keys* item in the resulting cascading menu.

Clear Button

The *Clear* button on the CRT context keys keypad is used to clear a CRT context key of previous assignments. To do so, first click on *Clear*. Next, click on the CRT context key to be cleared. The previous CRT Context assigned to the key will be cleared.

Context

A user's context is the user's working environment on the PWC90 console. This includes the preferred size, shape, location, number and contents of operator windows on the user's screen, as well as the size and position of any quick key keypads associated with the operator windows.

Login Layout Button

The *Login Layout* button is located at the top of the CRT context key keypad, above the CRT context keys and below the title bar. The displays assigned to this button will appear automatically when the user logs in.

Set Button

The *Set* button on the CRT context keys keypad is used to assign the displays currently on the screen to any one of the CRT context keys on the keypad. (Any pop-up windows present will be ignored). To do so, first make sure that the size, location and contents of the operator windows present are as desired. Next, click on *Set*. Keys which have already been assigned are unavailable and will be stippled out; click on the CRT context key to be set to call up the current displays. Enter the key label in the resulting pop-up window, and click on *Close* or press **Return** to complete the operation.

SECTION 10 - HISTORICAL BLOCK DATA COLLECTION

INTRODUCTION

The historical block database is designed for the efficient collection of data from a large number of tag.atoms. Like the data collected in the historical database, the historical block data is stored for use by other applications, such as logging, trending and data archiving.

Historical block configuration is used to create and edit historical blocks. Each historical block is a single entity which contains data collected from multiple points in the INFI 90 OPEN server tag database. All tags which are named in the configuration of the historical block share the same settings for collection rate, storage node, computations, pause and raw data.

The following guidelines should be observed when configuring historical block data collection.

- For optimum performance, each historical block should contain from 50 to 500 tag.atoms.
- The recommended collection period is 30 seconds or longer.
- No more than 500 tags should be collected per instance in time from the same INFI 90 OPEN server node (i.e., no more than 500 tags should share the same start time). If more than 500 tags are being collected, offset the time interval. For example, collect each set of tags every 30 seconds with a start time deviation of 15 seconds.
- Up to 5,000 points can be collected per PWC90 server using historical block data collection.

Access historical block configuration by using the pull-down menus available from the system window *Menu* button. Click on *Menu*; from the resulting pull-down menu, click on *Configure*, then click on *Historical Blocks* in the resulting cascading menu. The *Historical Block Database Configuration* window (Figure 10-1) will be displayed. The main portion of this display is a list box which contains the block name and legend for all the historical blocks which have been configured in the historical block database. Below this list are buttons which are used to add, copy, locate, edit and delete blocks in the historical block database.

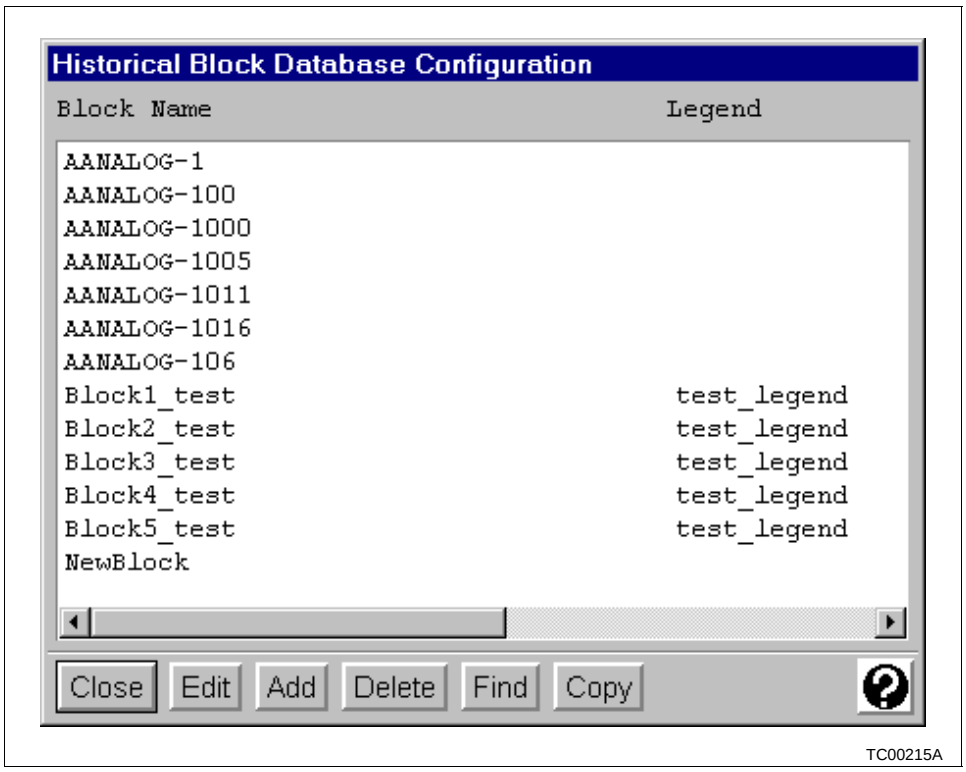


Figure 10-1. Historical Block Database Configuration Window

ADD HISTORICAL BLOCK

To create a block in the historical block database, click on *Add* in the *Historical Block Database Configuration* window. The *Historical Block Configuration* window (Figure 10-2) will appear. The window is divided into two main portions:

- Collection and storage parameter configuration.
- INFI 90 OPEN tag.atom specifications.

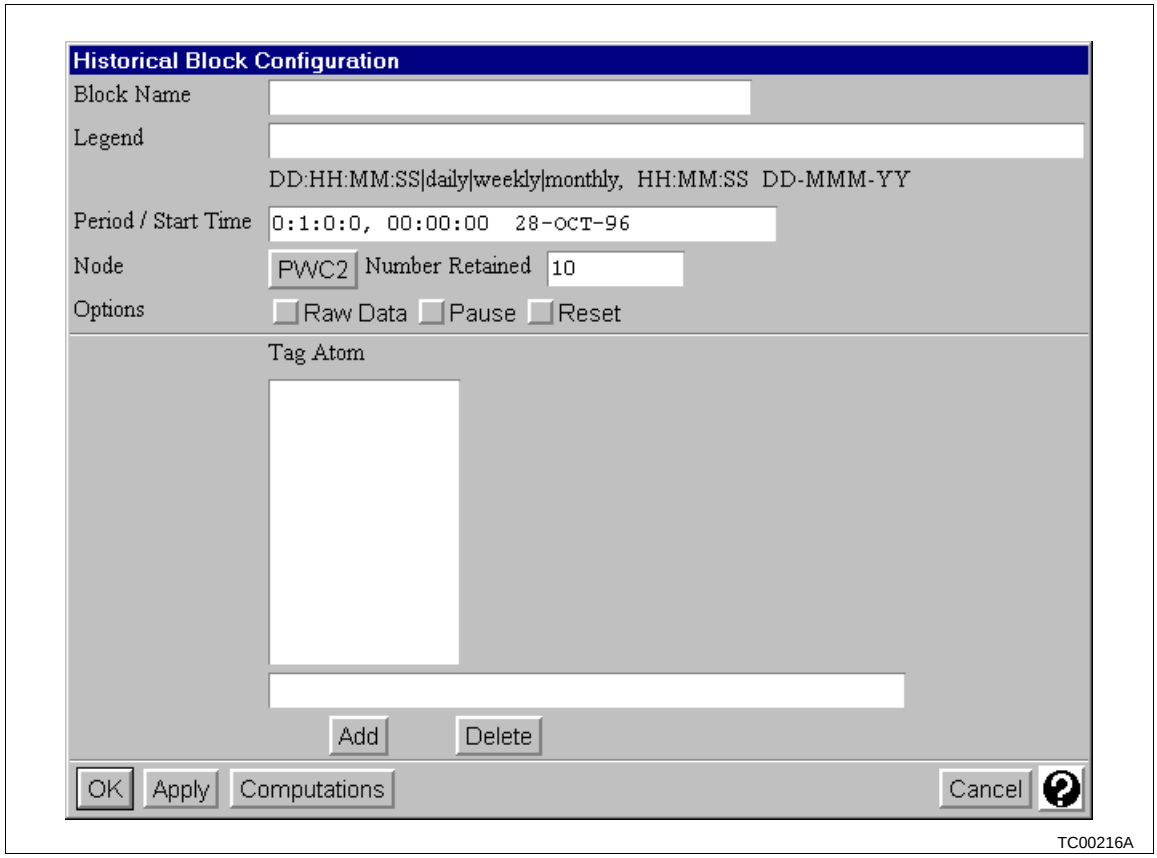


Figure 10-2. Historical Block Configuration Window

Historical Block Name

Click on the *Block Name* entry box to enter the string to be used to identify the block in the historical block database. This name can be up to 37 characters in length, cannot contain spaces, and must begin with an alphabetic character. The block name must also be unique on the network.

A block name must be specified; if no entry is made in the *Block Name* entry box, an error message will be displayed.

Historical Block Legend

Click on the *Legend* entry box to enter the string which is to be used as the legend for the historical database point under configuration. This legend can be up to 63 characters in length.

Period and Start Time

Both the *Period* (user-defined collection interval) and the *Start Time* (which defines the beginning of the *Period*) are entered in the same entry box. The *Period* is entered first; the words daily, weekly and monthly are valid entries to define the *Period*. Alter-

natively, the user can define a customized *Period* by defining the number of days, hours, minutes and seconds in the *Period* using the following format (leading zeros are required with single digit entries):

DD:HH:MM:SS

(The minimum recommended period is 30 seconds) Enter a comma (,) after the *Period* to separate it from the *Start*[ing] *Time*. After the *Period* has been defined, the *Start*[ing] *Time* (hours, minutes, seconds) and date (day, month, year) which mark the beginning of the interval, are entered in the following format:

HH:MM:SS DD-MMM-YY

The default *Period* is hourly (00:01:00:00) and the default *Start Time* is midnight of the current day (i.e., 00:00:00 30-MAY-96). Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name.

Node

The node on which the historical block is stored can be specified. The face of the *Node* button will display the name of the node selected to store data collected for the historical block. By default, this is the name of the local node. Click on *Node*; the *Storage Nodes* pop-up window (Figure 10-3) will be displayed.

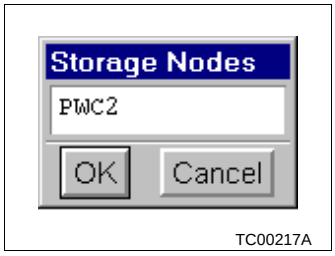


Figure 10-3. Storage Nodes Window

The *Storage Nodes* window contains a list of available nodes. Click on the node to be used to store the historical block data; it will be highlighted. Click on *OK* to complete the operation, or click on *Cancel* to exit the window without selecting a new node.

Number Retained

The text entry box labelled *Number Retained* is used to specify the number of collections to be retained for each tag.atom in the historical block under configuration. Once the retention limit has been reached, each new collection will replace the oldest collection. Enter **inf** (infinity) in the *Number Retained*

text entry box to retain all values collected; otherwise, enter a numeric value (zero is not a valid numeric value). To retain no data for the tags in the historical block, deselect the raw data toggle button described below.

> **NOTE:** The amount of available disk space and the number, frequency and retention rate of other collections on the system should be considered carefully when configuring the retention time.

Historical Block Collection Options

Three additional features are provided via toggle buttons:

- Raw data.
- Pause.
- Reset.

By default, these buttons are deselected (displayed in gray and appear raised). When selected, they appear pressed in and are displayed in yellow.

RAW DATA OPTION

When the *Raw Data* toggle button is selected, raw data collected from the INFI 90 OPEN servers will be retained in the historical block in addition to any computations which have been selected.

PAUSE OPTION

When the *Pause* option is selected, data collection will be stopped. To restart the collection, deselect the Pause toggle button. Click on *Apply* or *OK* on the *Historical Block Configuration* window to make changes to the *Pause* toggle button take effect.

RESET OPTION

When the *Reset* option is selected, all previously collected data values stored for the historical block will be deleted, and any portions of the database reserved for tags which have been deleted from the configuration of the block will be compressed. This does not affect the configuration of the historical block; all parameters and tags for the block will still be defined.

Tag Atom Assignment

The tag atom assignments portion of the historical block configuration window is used to specify the tag.atoms in the INFI 90 OPEN tag database from which data will be collected for the historical block. This portion of the historical block configuration window is composed of a list box, a text entry box, an *Add*

button and a *Delete* button. In the text entry box, enter the name of a valid tag.atom in the tag database. Click on *Add* to assign it to the current historical block. The name will be placed in the list box. As additional items are configured, they too will be added to the list box.

To delete one or more tag.atoms from the historical block, click on the item(s) to be deleted in the list box, then click on *Delete*. The entry (or entries) will be removed from the historical block configuration and from the list box.

Historical Block Data Computations

The PWC90 console can be configured to perform calculations on historical block data as it is collected. Available calculations are:

- Average.
- Minimum.
- Maximum.
- Standard deviation.

Computations can be performed hourly, daily, weekly or yearly. The computed values are stored in historical computation atoms. The computed values stored in these atoms can be used by other applications, such as trending. Atoms are also provided for two additional values, *Minimum Time* and *Maximum Time*. These atoms are used to store the timestamp of the first instance of the minimum and maximum collection values. The atom names are shown in Table 10-1.

Table 10-1. Historical Computation Atoms

Period	Average	Minimum	Minimum Time	Maximum	Maximum Time	Standard Deviation
Hourly	.HAV	.HMN	.HTMN	.HMX	.HTMX	.HSD
Daily	.DAV	.DMN	.DTMN	.DMX	.DTMX	.DSD
Monthly	.MAV	.MMN	.MTMN	.MMX	.MTMX	.MSD
Yearly	.YAV	.YMN	.YTMN	.YMX	.YTMX	.YSD

Click on *Computations*. The *Historical Tag Computations* window (Figure 10-4) will be displayed.

The *Historical Tag Computations* window is composed of a matrix of toggle buttons which allow the user to select individually the desired combination(s) of computation rate and computation type. Each row in the matrix represents a computation rate; each column represents a computation type. By default, all buttons are deselected and displayed in gray and appear raised. Click on the toggle button which corresponds to the desired computation rate and computation type; selected buttons are displayed in yellow and appear pushed in.

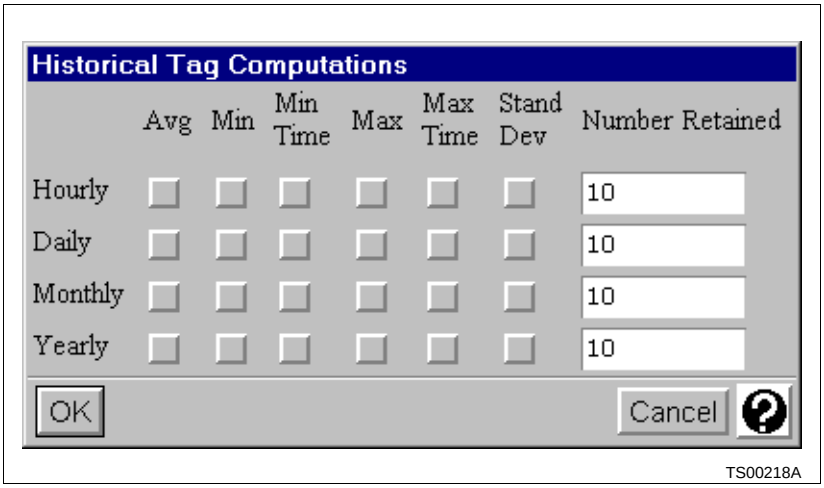


Figure 10-4. Historical Tag Computations Window

Use the *Number Retained* text entry box at the end of each row to define the number of values to be retained for each computation type selected for that computation rate. Once the retention limit has been reached, the value obtained by each new computation will replace the oldest value. Because the retention limit is defined on a per computation rate basis, it is not possible to set the *Number Retained* to a different value for different computation types with the same computation rate. For example, it is not possible to set a different *Number Retained* for the hourly/average and hourly/minimum values. However, it is possible to configure different retention limits for hourly/average and daily/average values. Enter **inf** (infinity) in the *Number Retained* text entry box to retain all values collected; otherwise, enter a numeric value (zero is not a valid numeric value). To retain no data for a particular computation type, all toggle buttons for the computation type must be deselected.

NOTES:

1. No checks are performed to validate collection rate vs. computation rate. The user must consider the collection rate of the data and take care to configure appropriate computation rates; i.e., while it is technically feasible to perform hourly computations on data which is collected once a day (daily), such a configuration would result in many unnecessary and non-productive computations.

2. The amount of available disk space and the number, frequency and retention rate of other collections on the system should be considered carefully when configuring the retention time.

Configuring Multiple Historical Blocks

Once the configuration of an historical block is complete, click on *Apply*. The current configuration session will be saved, and the *Historical Block Configuration* window will remain open so that additional historical blocks can be configured.

Exiting Historical Block Configuration

Click on *OK* to close the *Historical Block Configuration* window and save any configuration changes made during the current session (or since the *Apply* button was last clicked). Click on *Cancel* to exit the window without saving the current session. *Cancel* will not affect configuration activity which was committed using the *Apply* button.

HISTORICAL BLOCK OPERATIONS

Edit

To make changes to an existing historical block, first click on the desired block name in the *Historical Block Database Configuration* window (Figure 10-1), then click on *Edit*. The *Historical Block Configuration* window (Figure 10-2) will be displayed, containing the values currently configured for that block.

> **NOTE:** The block name cannot be changed when an existing historical block is edited; use the *Copy* button to copy the configuration of an existing historical block to a new block name.

Configure as described in ***ADD HISTORICAL BLOCK*** in Section 10.

Copy

To copy the configuration of an existing historical block to a new block name, use the *Copy* button. Click on the desired block name in the *Historical Block Database Configuration* window (Figure 10-1), then click on *Copy*. The *Historical Block Configuration* window (Figure 10-2) will be displayed, containing the values currently configured for that block, except for the *Block Name* and *Legend* text entry boxes, which are blank.

Configure as described in ***ADD HISTORICAL BLOCK*** in Section 10.

Delete

The *Delete* button is used to delete a block in the historical block database. Click on the name of the block to be deleted; it will be highlighted. Click on *Delete* to delete the highlighted block. A confirmation window will appear, which asks if the block is to be deleted, and which contains two buttons labelled *Yes* and *No*. Click on *Yes* to delete the block, or click on *No* to exit the operation without deleting the selected blocks.

Find

The *Find* button is located in the lower left corner of the *Historical Block Database Configuration* window (Figure 10-1) is used to locate a given *Historical Block* in the historical block database. If the block does exist, the *Find* button quickly locates it on the *Block Name* list and highlights. Once located, the configuration of the block can be viewed, edited or deleted as desired.

To use this feature, click on *Find*. A pop-up window appears, which contains an entry box, and *Ok*, *Apply* and *Cancel* buttons. Click on the entry box to focus on it, then enter the desired block name. If only a portion of the name is entered, the first entry that contains the entered string will be chosen. For example, if the string HR is entered, and the first block name to begin with these characters is HRANO1, then this is the block that would be highlighted in the *Block Name* list. The *Find* feature can only be used to locate strings in block names; it is not applicable to block legends.

Click on *Apply* to locate the first instance of the string; the *Find* pop-up window will remain open; click on *Apply* again to locate the next instance of the string, or enter a new string to be located. Click on *OK* to perform the *Find* operation and close the *Find* pop-up window. Click on *Cancel* to exit without attempting to find the block name in question, or to close the pop-up window.

Exiting Historical Database Configuration

Click on *Close* to close the *Historical Block Database Configuration* window.

SECTION 11 - @aGlance/IT SERVER INTERFACE

INTRODUCTION

The @aGlance/IT server interface is an optional utility which provides a live data link between the PWC90 console and a variety of software applications resident on other distributed control systems, supervisory control systems, personal computers, and so on. Easy access to this plant data assists those involved in problem analysis and resolution, product improvement, process improvement and resource optimization. Obtaining process data directly from the process control system eliminates the possibility of error introduced in manual data entry. @aGlance/IT provides a clean, standardized interface which allows data to be transferred between different hardware platforms while retaining its meaning, despite inherent differences in the way data is formatted on different systems (i.e., byte order, floating point formatting, and so on).

Tools to import live process data include a broad range of applications, such as relational databases, expert systems, spreadsheets, simulation tools, report generators, visualization tools, statistical analysis packages, planning tools, charting tools and graphic tools. Each class of tools (i.e., spreadsheets, statistical packages) has a different way of interacting with users.

In order for the exchange of data to take place, @aGlance/IT is required for each participating network partner.

The architecture of the @aGlance/IT software is based on a client-server model. An @aGlance/IT *server* typically runs on the system which owns the process control data (i.e., a PWC90 console); a process known as an @aGlance/IT *client* typically runs on the system which will be a consumer of that data (i.e., a PC or a supervisory control computer). A system can run both client and server applications simultaneously. Client software must be provided by the user.

@aGlance/IT API for the INFI 90 OPEN System

Any @aGlance/IT client can connect to any @aGlance/IT Server, assuming the minimum requirements for network transport are met to access the server running on the PWC90 console.

The @aGlance/IT Application Programming Interface (API) on an PWC90 console consists of:

- An @aGlance/IT server which can provide data to @aGlance/IT clients.

- Support for @aGlance/IT methods (functions) used by:

  Commercially available @aGlance/IT client applications (i.e., the @aGlance/IT Add-In for EXCEL) and users engaged in creating custom @aGlance/IT client applications.

Also provided is a window-based interface for:

- Identifying data structures in the INFI 90 OPEN server databases (by tagname and atom).
- Defining translations for the INFI 90 OPEN server atoms to other terminologies, such as the Instrument Society of America's SP72 standard.
- Viewing information about servers and server activity.

The implementation of @aGlance/IT on an PWC90 network permits the access of up to 500 points per second from either the client or server side.

Requirements for the INFI 90 OPEN @aGlance/IT Server

The following requirements must be met in order to successfully run the @aGlance/IT API server:

- TCP/IP must be installed on the node on which the @aGlance/IT server is to run; TCP/IP is normally always present on a Windows NT PWC90 console. There are no special installation actions necessary to install it.
- A properly installed license for the @aGlance/IT API for the PWC90 network must be installed via the ***Personal Work Center 90 Installation*** instruction.

  NOTE: A single client licence is installed as standard. A multiple client licence is optionally available.

@aGlance/IT SERVER OPERATION

The @aGlance/IT API for the PWC90 network translates between the @aGlance/IT methods and data structures, and those of the INFI 90 OPEN server database.

Configuring the @aGlance/IT Server Internal Interface

There are two aspects to @aGlance/IT server configuration: external interface configuration and internal interface configuration. The @aGlance/IT *external interface* configuration is used to identify tagnames and attributes (atoms) in the INFI 90 OPEN server database, and is described in ***Starting a Server and Server Names*** below. The @aGlance/IT internal interface

configuration is used to identify the nodes on which @aGlance/IT servers will run, and is described here. This must be done so that the nodes can be recognized by @aGlance/IT clients. This is necessary because servers identify themselves to @aGlance/IT at run time, and client applications specify, at run time, the names of the servers with which they wish to communicate. When a client issues a request to communicate with a server application, @aGlance/IT will look for it on servers which have been identified using the aag_admin utility. The @aGlance/IT internal interface is configured using the aag_admin utility.

After an @aGlance/IT server has been defined, the @aGlance/IT item available from the system window *Menu* button provides the ability to identify the set of tagnames and their associated attributes (atoms) which will be recognized by the @aGlance/IT servers. It also allows you to configure translations of the attribute (atom) names, to make them recognizable to applications on other platforms.

Registering Client Nodes

Use the @aGlance/IT aag_admin utility to register all potential client nodes in each PWC90 @aGlance/IT server network database. This function identifies and characterizes nodes from which clients will access the server. Because the PWC90 @aGlance/IT Server communicates using TCP/IP, the aag_admin utility will prompt the user to enter an internet host name. (If you have questions about internet host names in use at your site, see your system administrator).

The @aGlance/IT *Administration* menu will be displayed. *Server Setup* is used to tell @aGlance/IT what clients are permitted to access the local node and servers on that system. Enter H at the menu prompt to access the @aGlance/IT on-line Help system for information on how to use the @aGlance/IT *Administration* menu to add and delete client nodes to and from the network database of the @aGlance/IT server for a PWC90 network.

Starting a Server and Server Names

Typically, the maximum number of @aGlance/IT servers for which the node is licensed are started when the system is started. A Server started automatically by the system will be assigned a default name in the format:

EB*i*<node name>

where the *i* is an index number used to identify the server, and the <node name> is name of the PWC90 node on which the server is running. For example, if the node named PWC4 is

licensed for two servers, two servers will be started automatically. The servers will be named ***EB1PWC4*** and ***EB2PWC4***.

A maximum of two @aGlance/IT servers can run simultaneously on an PWC90 console, and each of these servers can have up to five simultaneous client connections.

STARTING A SERVER WITH NON-DEFAULT OPTIONS

It is possible to customize parameters such as server name and interval for the servers which are started automatically on system start up. To do so, you must edit the shell script aag_start, located in the directory /Pwc/bin. You should be familiar with the concept of shell programming before attempting to do this. If you have questions about shell scripts, the aag_start shell script, or the @aGlance servers at your site, see your system administrator. Server options to the aagServer command which are available when starting a server are described in Table 11-1.

Table 11-1. Options for Use When Starting a Server

Option And Argument	Description
-server <server name>	The argument supplied with this option is the name to be used for the @aGlance/IT server. The server name can have a maximum of 20 characters. If this option is omitted, the default server name aagServer will be used. On a system licensed for the @aGlance/IT server for the PWC90 network, the number of servers for which the system is licensed will be started automatically when the system is booted. (The maximum number of servers per node is two).
-nsessions <number>	The argument (number) supplied with this option determines the maximum number of sessions allowed for the @aGlance server. The maximum number of sessions allowed per server is four, which is the default used if the -nsessions option is not specified. If an argument with a value less than one is supplied with this option, the number of sessions will be set to one. If an argument with a value greater than four is supplied, the number of sessions will be set to four.
-interval <seconds>	The argument (seconds) supplied with this option defines the number of seconds in the poll interval. This argument is an integer; the minimum number of seconds that can be specified is one.
-monitor	When this option is used, the monitor feature will be enabled. Monitoring provides the means whereby an @aGlance/IT client is advised (at a default or specified poll interval) regarding the values contained in specified data points.
-daemon	When this option is used, the server will run as a background process.
-clientshutdown	When this option is used, a client which requests a server shutdown using the AAG_Shutdown method will be able to shutdown the server, provided that the session which is requesting the shutdown is the only session running.
-evtmessage	Each time a client starts or ends a session, a message is displayed in the system event display area of the mini-alarm window, the event review display, and is also entered into the event historian. The message contains the following data in the order shown: <Time><Date><Node Name><@aGlance Server Name>(<INFI 90 OPEN login name>:<Remote Host name>:<Remote User account identification>)<session ID> When the -evtmessage option is used, these messages will be suppressed.

Table 11-1. Options for Use When Starting a Server

Option And Argument	Description
-server <server name>	The argument supplied with this option is the name to be used for the @aGlance/IT server. The server name can have a maximum of 20 characters. If this option is omitted, the default server name aagServer will be used. On a system licensed for the @aGlance/IT server for the PWC90 network, the number of servers for which the system is licensed will be started automatically when the system is booted. (The maximum number of servers per node is two).

NOTE: Clients do not need to be aware of which node the server is running on.

The @aGlance/IT external interface configuration is used to identify tagnames and attributes (atoms) in the INFI 90 OPEN server database. Access @aGlance/IT external interface configuration by using the pull-down menu available from the system window *Menu* button. Click on the *Menu* button; from the resulting pull-down menu, click on the *Configure* option, then click on *@aGlance/IT* in the resulting cascading menu. The *@aGlance Configuration* window (Figure 11-1) will be displayed.

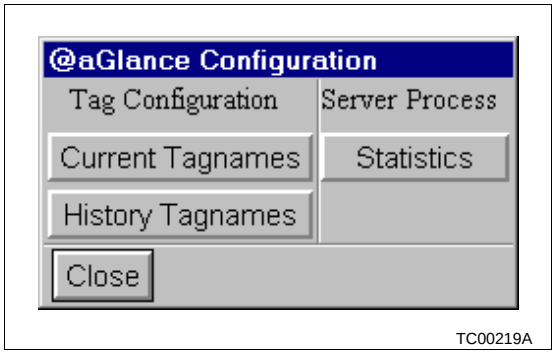


Figure 11-1. @aGlance Configuration Window

The *@aGlance Configuration* window is used to access configuration of current and historical data in the PWC90 console. It also provides access to information on the @aGlance/IT servers resident on the PWC90 network

CONFIGURE @aGlance/IT CURRENT TAGNAME

Click on the *Current Tagnames* button to open the @aGlance *Current Tagnames* window (Figure 11-2).

The *@aGlance Current Tagnames* window contains a scrollable list of all the tagnames in the INFI 90 OPEN server database which have been configured for @aGlance/IT to recognize.

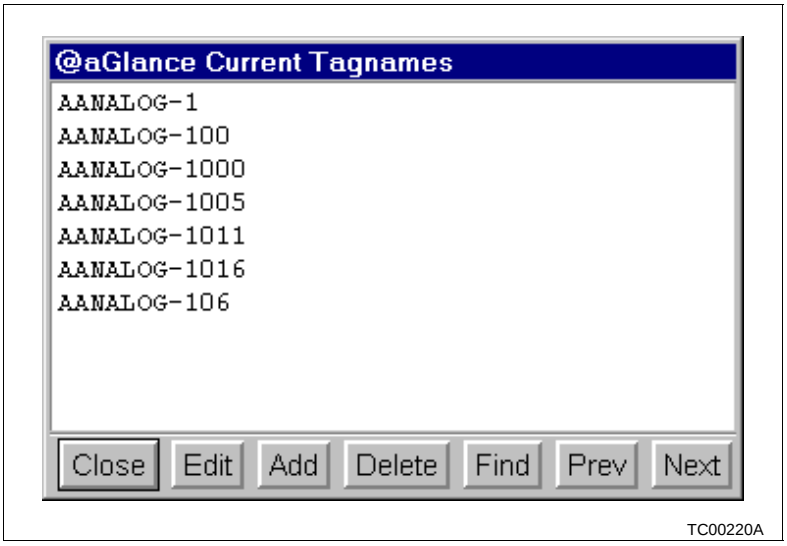


Figure 11-2. @aGlance Current Tagnames Window

Add an @aGlance/IT Tagname

To add a new tagname to the list, Click the *Add* button at the bottom of the window. The *@aGlance Add Tagname* window (Figure 11-3) will be displayed.

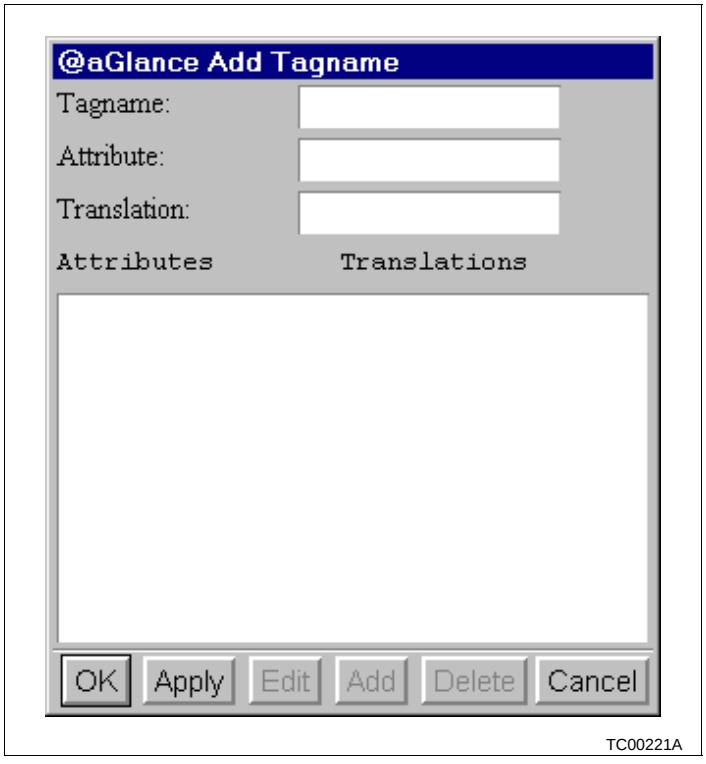


Figure 11-3. @aGlance Add Tagname Window

Text entry boxes labeled *Tagname, Attribute*, and *Translation* appear in the top portion of the *Add Tagname* window. In the *Tagname* text entry box, enter a tagname from the INFI 90 OPEN server database. In the *Attribute* text entry box, enter the text string to be used to represent the attribute when it is accessed by an @aGlance/IT client application. (If no attribute name is entered it will default to *Out*). The type of name entered for the *Attribute* depends on the format needed by the client application. The text string entered can be a INFI 90 OPEN server atom name, or it can be an SP72 name or other conventional name for a piece of data. It is the responsibility of the user to know the data naming conventions and/or requirements of the client application.

> **NOTE:** In general @aGlance/IT database access conforms to the ISA-SP72 tag/attribute model. This is equivalent to the tagname.atom model used in INFI 90 OPEN server database access.

In the *Translation* text entry box enter an atom name from the INFI 90 OPEN server database. The atom will then be represented by the attribute with which it is now associated when it is accessed by an @aGlance/IT client application. Associating an atom name and an attribute in this manner effectively translates the INFI 90 OPEN server system atom into the client application's terminology.

Once the attribute and translation have been entered, click on the *Apply* button to cause these items to be added to the list of attributes configured for the tagname. The tagname remains in the *Tagname* text entry box and the *Attribute* and *Translation* text entry boxes will be emptied allowing additional attributes to be configured for the same tagname.

Note that after at least one attribute has been applied, the *Edit, Add* and *Delete* buttons on the *@aGlance Add Tagname* window become available. After entering input for the additional attributes click on the *Add* button to add the additional items to the configuration.

To change the configuration of an item, click on that item in the list box, then click on *Edit*; the entries will appear in the appropriate text entry boxes available for revision.

To delete an attribute/translation entry click on the desired item in the list box then click on the *Delete* button; the entry will be removed from the list.

Clicking *Ok* will close the *@aGlance Add Tagname* window; changes which have been applied during the configuration session will be retained; items placed in the text entry box but not applied will be discarded. To discard all changes made during the configuration session, click on *Cancel*.

Edit an @aGlance/IT Tagname

To edit the configuration for an item in the list appearing in the *@aGlance Current Tagnames* window (Figure 11-2), click on the desired entry in the list box then click on the *Edit* button at the bottom of the window. The *@aGlance Edit Tagnames* window (Figure 11-4) will be displayed.

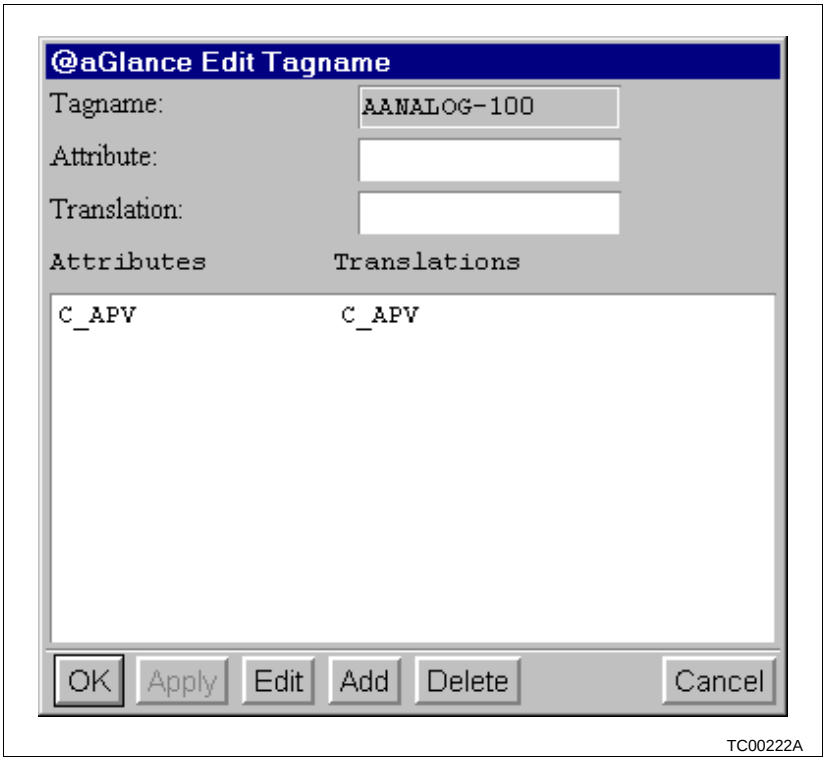


Figure 11-4. @aGlance Edit Tagname Window

The *Apply* button is unavailable in this window; use the *Edit, Add, Delete* and *Cancel* buttons as described for the *@aGlance Add Tagname* window described in the preceding subsection.

Note that tagnames themselves can not be edited. To change a tagname, delete the entry for it and add a new one.

Find an @aGlance/IT Tagname

If many current tagnames have been configured, use the *Find* button at the bottom of the *@aGlance Current Tagnames* window to quickly locate a particular tagname. Click on the *Find* button; the *@aGlance Find Tagname* window will be displayed. This window contains an entry box and *OK, Apply* and *Cancel* buttons. Click on the entry box to focus on it then enter the desired tagname. If only a portion of the tagname is entered the first entry where the first letters of the tagname match the entered string will be chosen. For example if the string HR is

entered and the first tagname to begin with these characters is HRANO1 then this is the tagname that would be highlighted in the *@aGlance Current Tagnames* list.

Click on *Apply* to locate the first instance of the string; the pop-up window will remain open; click on *Apply* again to locate the next instance of the string or enter a new string to be located. Click on *OK* to perform the *Find* operation and close the *@aGlance Find Tagname* window. Click on *Cancel* to exit without attempting to find the tagname in question, or to close the pop-up window. Note that the *Find* operation is case-sensitive.

Viewing @aGlance/IT Current Tagnames

If the number of current tagnames which have been configured exceeds the number which can be displayed in the *@aGlance Current Tagnames* window, click on the *Next* button to view the next portion of the current tagnames list. Click on the *Prev* button to move to a preceding portion of the list.

Configure @aGlance/IT History Tagnames

The @aGlance/IT interface can be configured to access data in the historical, archival, historical block and archival block databases. To do so click on the *History Tagnames* button to open the *@aGlance History Tagnames* window (Figure 11-5).

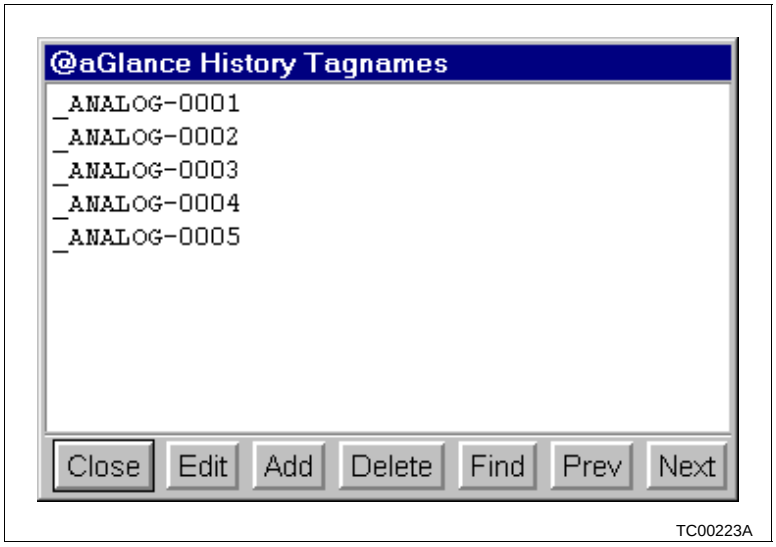


Figure 11-5. @aGlance History Tagnames Window

Configuration of @aGlance history tagnames is accomplished in the same manner as configuration of @aGlance/IT current tagnames. Follow the procedures described above for configuration of @aGlance/IT current tagnames, using tagnames con-

figured in the historical, archival, historical block and archival block databases.

File name syntax plays a role in determining which database will be accessed for data. Each historical database tagname must be unique on the PWC90 network and represent a INFI 90 OPEN server tagname.atom combination. To make a permanent record of the data, it can be archived to tape. Archived data can be restored for later viewing and use. the restored data is known as the archival database. data from the historical and archival databases is requested in the format:

<tagname>.<atom>

A historical block is a single entity which contains data collected from multiple points in a INFI 90 OPEN server. All the tagnames configured for collection by a historical block share the same collection parameters (i.e., collection rate). Like historical data, historical block data can be archived to tape to create a permanent record. This archived data can be restored later for viewing and use. The restored data is the archival block data. Data from the historical and archival databases is requested in the format:

<blockname>..<tagname>.<atom>

NOTE: If no attribute name is entered when performing configuration using a tagname from the historical database, archival database, historical block database or archival block database then the attribute name will default to HST. If the attribute name ARC is specified only the archival database or archival block database will be accessed.

VIEWING @aGLANCE/IT SERVER PROCESS DATA

By default, the maximum number of @aGlance/IT server processes for which the PWC90 console is licensed will be started when the system is booted. (A maximum of two @aGlance/IT servers can run simultaneously on an PWC90 which is resident on the Pentium hardware platform, and each of these servers can have up to five simultaneous client connections).

To display a list of servers which are running on the PWC90 network, click on the *Statistics* button found on the *@aGlance Configuration* window. The *@aGlance Servers* window (Figure 11-6) will be displayed.

The name of the server node and the server name for each server process will appear in the list box which occupies the main portion of the window. To view more detailed information for a particular server process, click on the entry for that process in the list box to select it then click on the *View* button at the bottom of the *@aGlance Servers* window. The *@aGlance Server View* window (Figure 11-7) will be displayed.

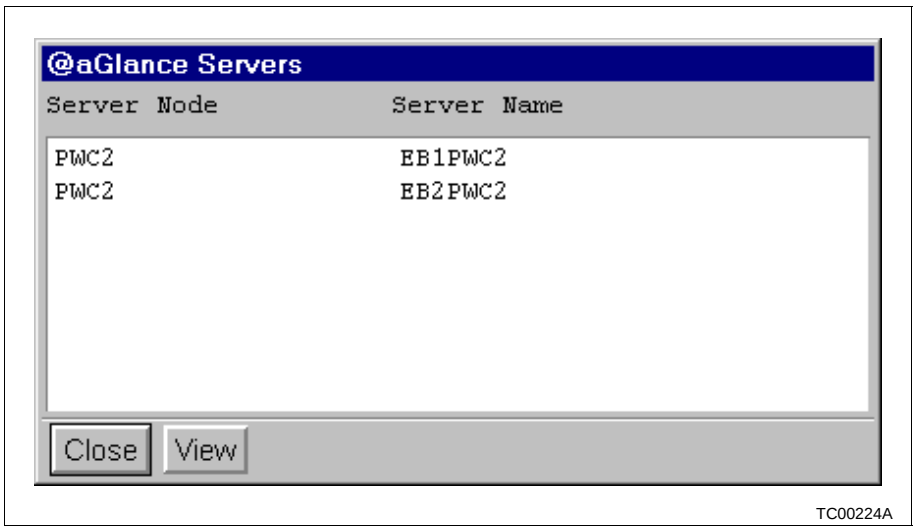


Figure 11-6. @aGlance Servers Window

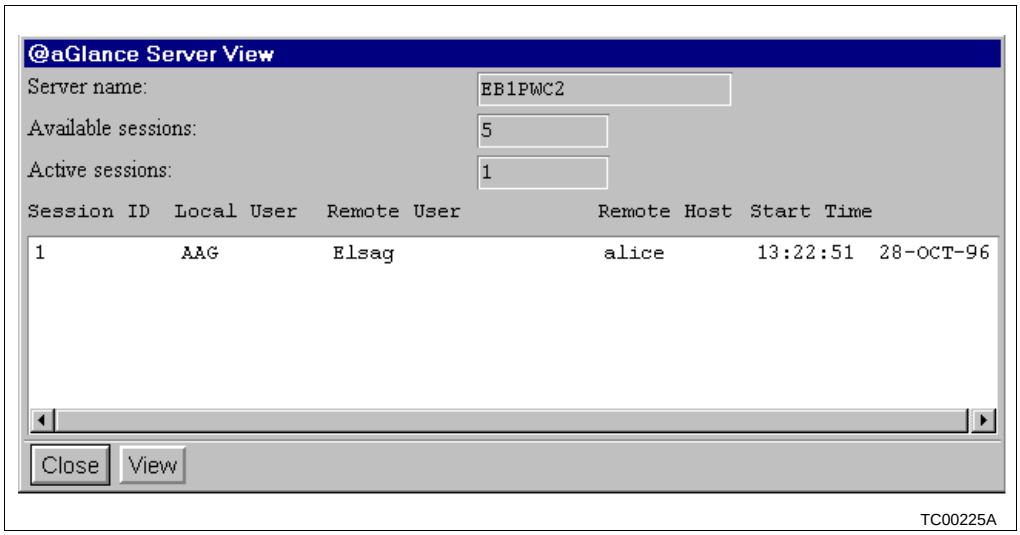


Figure 11-7. @aGlance Server View Window

The @a*Glance Server View* window displays the following information about the selected server.

- Server name.
- Available sessions - the number of sessions which can be started by @aGlance/IT Clients on the selected server.
- Active sessions - the number of client-initiated sessions currently active on the selected server.

A list box contains an entry for each active session on the server. Each entry provides the following information about the session:

- Session ID (the Session ID is incremented by one each time a session is started).
- Login name of the local user.
- Login name of the remote user.
- Host name of the remote host (the host on which the @aGlance/IT Client which started this session resides).
- Start time; i.e., the time the client session was initiated in the format HH:MM:SS DD-MMM-YY.

Additional information is available about each active session. To view information about a session, click on the desired entry in the list box then click on the *View* button found at the bottom of the *@aGlance Server View* window. An *@aGlance Session Detail View* window will be displayed.

This window provides information about the requests initiated by the server during the selected active session. A @aGlance/IT client requests data using features known a methods. Corresponding call backs exist in the server. When an @aGlance/IT client issues a call to an @aGlance/IT method, the corresponding callback in the server provides the information to the client. The *@aGlance Session Detail View* window lists the available callbacks for the active session the number of transactions performed per callback the number of tags and the number of atoms requested per callback. This information is cumulative over the life of the session and is lost when the session ends.

NOTE: For entries in this table for which data can be supplied by @aGlance, the applicable value or a zero will appear. If there is no data provided for an item the entry box will be blank.

Backup/Restore @aGlance/IT Configuration

Use the Backup/Restore utility found on the *Utilities* pull down menu to backup @aGlance/IT configuration data. @aGlance/IT configuration data is backed up as part of the PWC90 console configuration, and is present as a selectable option on the *PWC90 Configuration Data* window (Figure 11-8). Refer to the ***Personal Work Center 90 Operation*** instruction for details on how to use the Backup/Restore utility.

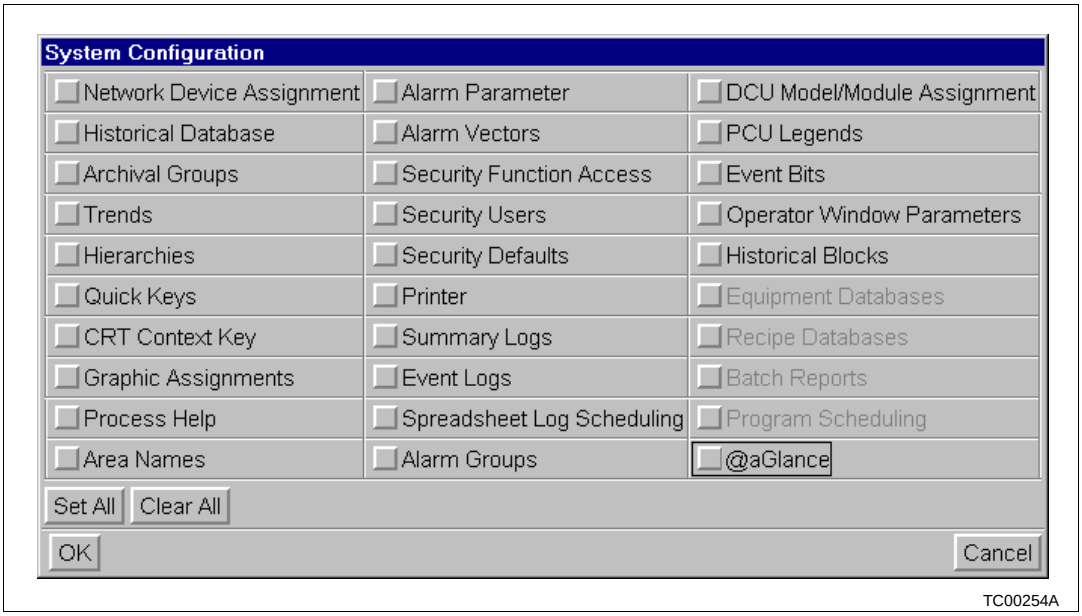


Figure 11-8. PWC90 Configuration Data Window from the Backup/Restore Subsystem.

SUPPORTED FEATURES FOR USE WITH @aGlance/IT CLIENTS

@aGlance/IT Methods Supported

When an @aGlance/IT client application requests information from another system running an aGlance/IT server, a set of parameters is passed from the client application to the @aGlance/IT client software. The @aGlance/IT client software locates the system on which the desired server is running and connects to the server. The @aGlance/IT client instructs the server which function (known as a method), to execute, and passes on the parameters from the client application. The method specified by the @aGlance/IT client software corresponds to a callback function in the server. A call_back list in the @aGlance/IT server software registers the available callback functions. (If there is no corresponding callback function in the server the request will fail).

The PWC90 @aGlance/IT Server supports the methods listed in Table 11-2.

Table 11-2. Supported @aGlance/IT Methods

Method	Description	Use
AAG_GetList	Get a list of data point values.	These methods are used to access data in the current database. The default attribute name used with these methods is OUT.
AAG_GetTable	Get a table of data point values.	
AAG_PutList	Put a list of data point values.	
AAG_PutTable	Put a table of data point values.	

Table 11-2. Supported @aGlance/IT Methods (continued)

Method	Description	Use
AAG_GetHistory	Get a table of historical values.	These methods are used to access data in the historical, historical block, archival and archival block databases. The default attribute name used with these methods is HST. The ARC attribute name can be used to specify access limited to the archival database.
AAG_PutHistory	Put a table of historical values.	
AAG_GetHistEvent	Get compressed historical data	
AAG_GetHisStat	Get statistical values from a server	
AAG_GetTags	Get a list of tagnames.	These methods are used to obtain information on the names used to identify data.
AAG_GetAttrs	Get a list of attribute names.	
AAG_Monitor	Register a monitor request.	These methods are used to start and cancel monitor mode.
AAG_CancelMonitor	Cancel a monitor request.	
AAG_Command	Execute a server-specific request	This method is used to obtain batch data.
AAG_Shutdown	Terminate the server.	This is a maintenance-related method.

Default Attributes Names

If no attribute name is entered when tag configuration is performed to access data in the current database of a INFI 90 OPEN system, a default attribute name will be provided. Default attribute names are shown in Table 11-3.

Table 11-3. Default Attribute Name

INFI 90 OPEN Database	Default Attribute Name
Current Database	OUT
Historical/Archival Database	HST
Historical/Archival Block Database	HST

Format of Timestamps

In the historical, historical block, archival and archival block databases, a timestamp is stored with collected data. The time and date, in the format described above, are typically used to define the time period for which data is to be retrieved.

Time and date stamps recognized and returned by the @aGlance/IT server for the INFI 90 OPEN server are in the format:

HH:MM:SS DD-MMM-YY

where HH is the number of hours since midnight, MM is the number of minutes, and SS is the number of seconds. The time is followed by the date (day/month/year). Note that the 24 hour clock is used, and the month is identified by the first three letters of the month name, in English (i.e., JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC).

SERVER-SPECIFIC PARAMETERS FOR HISTORY-RELATED METHODS

The @aGlance/IT methods used to access the historical, historical block, archival and archival block database include AAG_GetHistory, AAG_GetHistEvent, AAG_GetHistStat and AAG_PutHistory. Each of these methods passes parameters to the @aGlance/IT Server for the PWC90 console. Some of the parameters passed must be in a format specific to the server. These parameters are discussed in the following subsections.

Server-Specific AAG_GetHistory Method Parameters

The AAG_GetHistory method returns the historical values for a specified list of tag and attribute names. The time range is defined by a start time, an interval time, and a number of intervals. For each time interval, a value is returned for each of the tag/attribute pairs.

When this method is used by an @aGlance/IT client during a session with an @aGlance/IT Server for the PWC90 console, the start, interval and time_list parameters of this method require that a server-specific format be used for time specification.

The *start* parameter is a pointer to a character string which contains the start time. The start time must be in the format HH:MM:SS DD-MMM-YY.

The *interval* is a pointer to a character string containing the time interval. The following format is used to specify the interval by defining the number of days, hours, minutes and seconds in the interval using the following format:

DD:HH:MM:SS

The *time_list* parameter of the GetHistory method is a list of times associated with each historical record. (The time_list parameter is of the AAGTime data type). This list is filled in by the server. The number of entries in the list is determined by the number of requested time intervals specified in the GetHistory *count* parameter.

Server-Specific AAG_GetHistEvent Method Parameters

The AAG_GetHistEvent method allows a client to retrieve compressed historical data from a server. No time interval is specified for receiving interpolated data. Instead, a start time and an end time are specified by the client, and the server returns historical values as stored in the historical, archival, historical block or archival block database; compressed and without interpolation.

The *start* parameter is a pointer to a character string which contains the start time. The start time must be in the format HH:MM:SS DD-MMM-YY.

The *end* parameter is a pointer to a character string which contains the end time. The end time must be in the format HH:MM:SS DD-MMM-YY.

The *time_list* parameter is a list of timestamps returned by the server, which specifies the times at which samples where recorded. Time stamps are in the format:

HH:MM:SS DD-MMM-YY

Server-Specific AAG_GetHistStat Method Parameters

The GetHistStat method is used by a client to retrieve statistical data from a server. Statistical values returned include the maximum, minimum, average, and standard deviation over a time period.

The *start* parameter is a pointer to a character string which contains the start time. The start time must be in the format HH:MM:SS DD-MMM-YY.

The *end* parameter is a pointer to a character string which contains the end time. The end time must be in the format HH:MM:SS DD-MMM-YY.

The *type_list* is a list of character string pointers containing the statistical types for the @aGlance/IT Server for the PWC90 console. The following standard @aGlance types should be used: AAG_HIST_MIN, AAG_HIST_MAX, AAG_HIST_AVG, AAG_HIST_STDDEV.

Server-Specific AAG_PutHistory Method Parameters

The AAG_PutHistory method allows a client to write data into the historical, archival, historical block or archival block database via the @aGlance/IT Server for the PWC90 console. For example, this could be used to record laboratory data along with process data.

The time_list parameter is a list of times associated with the values being sent to the server. These must be in the format HH:MM:SS DD-MMM-YY. This list must contain the same number of entries as the list of tag names.

SERVER-SPECIFIC STATUS INFORMATION

AAG_StatusList is a an @aGlance/IT data type used to report information regarding the success or failure of a request to an @aGlance/IT server. The data type contains three elements:

- Value indicating success or failure.
- Text.
- Value representing a severity code.

The Value and Text elements are server dependent; the Severity codes are defined by the @aGlance/IT software. The contents of these elements are shown in Table 11-4.

Table 11-4. AAG_StatusList Information

Element	Contents When Access Succeeds	Contents When Access Fails
Value	D_SUCCESS (always zero)	INFI 90 OPEN system DBA error status code (sign inverted)
Text	(null)	INFI 90 OPEN system DBA error status description
Severity	AAG_SUCCESS_SEVERITY	AAG_ERROR_SEVERITY

This AAG_StatusList data type is used with the following methods supported by the @aGlance/IT Server for the PWC90 console: AAG_GetList, AAG_GetTable, AAG_PutList, AAG_PutTable, AAG_GetHistory, AAG_PutHistory, AAG_GetHistEvent, AAG_GetHistStat, and AAG_Monitor.

SECURITY FOR THE @aGlance/IT SERVER

The two aspects to security for the @aGlance/IT API for the PWC90 console are:

- Client access to the @aGlance/IT server for PWC90.
- Access to the INFI 90 OPEN server database.

Security for Access to the @aGlance/IT Server for PWC90

Client proxy definition is used to specify which clients are permitted to access the server applications on the local node. A client proxy account is a mapping from a remote host name and user id to a local id. Client proxies are configured on the node in which the @aGlance/IT Server for PWC90 is installed. Client proxy definition is performed using the @aGlance/IT aag_admin utility. (Access to the aag_admin utility is described in ***Registering Client Nodes***) There are three types of client proxy accounts:

- Separate proxy account for a user on a remote node.
- Single proxy account for all users on a remote node.
- Single proxy account for all users on all remote nodes.

The simplest procedure is to set up a single account for all users on all remote nodes. The default proxy is called AAG. To use this default proxy, configure a PWC90 user named AAG, using user login configuration as described in Section 2. The desired access level and area permissions should be configured for the user named AAG.

Client proxy definition is a function of the aag_admin utility provided by Intuitive Technology Corporation; it does not contain provisions for passwords on client proxy accounts. Security is based on the @aGlance/IT server for PWC90 matching the remote node and user names with the local proxy configuration at runtime.

See Subsection ***Registering Client Nodes*** for details on how to access the aag_admin utility. See the @aGlance/IT on-line help subsystem for information on client proxies and how to configure them.

Security for INFI 90 OPEN Database Access

Database access security for the @aGlance/IT Server for the PWC90 consoles is based on the access level and area permissions configured on the remote node for the user running the @aGlance/IT client software. The access level and area permissions correspond to the local user name specified in the @aGlance/IT proxy account. Each session can be associated with a different proxy account.

MONITORING

Monitoring provides a means whereby an @aGlance/IT Client is advised regarding the values contained in specified data points. The @aGlance/IT server supported on the PWC90 console is based on polled-interval data collection. That is, the data points specified by the client are polled at specific time intervals and the values in the data points are returned to the client.

The request for monitoring is made by the @aGlance/IT client. A typical client is the Microsoft EXCEL spreadsheet, with the @aGlance/IT Add-in for Microsoft EXCEL installed. This manual assumes that the user is familiar with Microsoft EXCEL and @aGlance/IT for Excel; it is beyond the scope of this manual to document the use of these products; consult the appropriate product documentation for details.

Monitor Time Interval

The time interval at which the data points are polled is referred to as the *poll interval*. The default poll interval used by the @aGlance/IT server supported on the PWC90 console is five seconds. This means that an @aGlance/IT client that has requested the monitor feature would be updated every five seconds. To change the default time interval, edit the aag_start shell script (discussed in ***STARTING A SERVER WITH NON-DEFAULT OPTIONS***). Revise the desired entry (or entries) in the aag_start shell script to include the -i (interval) flag and an integer argument specifying the number of seconds in the interval. (The minimum number of seconds is one and the

maximum is 86399, which is the number of seconds in one day, minus one second).

Refer to ***STARTING A SERVER WITH NON-DEFAULT OPTIONS*** for more information on starting a server with non-default options.

A client can specify a different time interval using an optional server command line argument to override the default. To do so, the client must supply an ASCII text string to the interval parameter of the AAG_monitor facility. The string, which will ultimately be passed to the PWC90 software, must be in the following format:

DD:HH:MM:SS | DAILY | WEEKLY | MONTHLY, HH:MM:SS DD-MMM-YY

The poll interval is defined first; the words daily, weekly and monthly are valid entries to define the poll interval. A customized poll interval is created by defining the number of days, hours, minutes and seconds in the poll interval using the following format:

DD:HH:MM:SS

The poll interval can be followed by a comma (,) or a space and an optional starting time. The default poll interval of five seconds would appear as 00:00:00:05. After the poll interval has been defined, the starting time (hours/minutes/seconds and day/month/year) is entered in the following format:

HH:MM:SS DD-MMM-YY

The 24 hour clock is used, and the default starting time is midnight of the current day (i.e., 00:00:00 08-NOV-91). Note that the month is identified by the first three letters of the month name, in English (i.e., JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC).

If the time specifying string cannot be processed, the default monitor time interval will be used.

Monitor Types

Three types of monitoring are supported. These are:

- Polled Monitoring.
- Deadband-Absolute Monitoring.
- Deadband-Percent Monitoring.

The argument *monitor_type* is a server-specific argument to the aag_monitor method; the argument is supplied by the

@aGlance/IT client software. Identifying strings defined in the @aGlance/IT server for PWC90 are shown in Table 11-5.

Table 11-5. Supported Monitor Type Identifier Strings

Monitor Type	Identifying String
Polled	MON_POLL (default)
Deadband-Absolute	MON_DEAD_ABS
Deadband-Percent	MON_DEAD_PCT

If the monitor_type argument is specified by the client and is not one of the strings defined in Table 11-5, the request will fail. Errors are returned to the client via the AAG_INVARG status.

POLLED MONITORING

Polled monitoring updates the client application at the poll interval. This is the default monitor type, and is performed if no argument is supplied for the monitor_type argument.

DEADBAND-ABSOLUTE MONITORING

Deadband-absolute monitoring used when the client is to be updated only when certain conditions exist. This type of monitoring uses the *monitor_args* array, a server-specific argument list. Each floating point or integer value represented by a tag.attribute combination is associated with a deadband value (D), based on the last value updated (U). A current value (C) is within the deadband if the following inequalities are true:

((C>=(U-D))&&(C<=(U+D)))

The client application will be updated if:

- For floating point and integer values, any value falls outside the deadband.

  - ***or*** -

- For string and boolean values, if any value does not compare exactly to the update value.

  - ***or*** -

- If any status change occurred.

The argument list of the monitor_args array must contain at least one element or the client request will fail. Errors are returned to the client via the AAG_INVCOUNT status.

DEADBAND-PERCENT MONITORING

Deadband-percent monitoring differs from deadband-absolute monitoring only in that the deadband value is a percentage of the magnitude of the last stored value. The percentage is expressed as a floating point value with the range 0<=D<=1. The deadband condition in this case is:

((C>=U-(U*D))&&(C<=U+(U*D)))

The argument list of the monitor_args array must contain at least one element or the client request will fail. Errors are returned to the client via the AAG_INVCOUNT status.

Alternatively, the deadband condition can be stated as:

((C<(U-(U*D))) || (C>(U+(U*D))))

If this evaluates to TRUE, then an update will be sent. If the equation evaluates to FALSE, no update will be sent.

Monitor Request Size

A maximum of 100 tag.attribute pairs are allowed per monitor request. Larger requests will fail. Errors are returned to the client via the AAG_INVCOUNT status.

SUPPORTED @aGlance/IT CLIENT APPLICATIONS

Add-Ins are software packages which provide client applications with an @aGlance/IT client interface, thus providing the application with the ability to interface with systems which include an @aGlance/IT server. Clients and servers can have totally different hardware architectures; the differences in underlying network protocols are resolved by the @aGlance/IT software.

@aGlance/IT ADD-IN FOR MICROSOFT EXCEL

The @aGlance/IT Add-In for Microsoft EXCEL is supported on PWC90 consoles. The user is responsible for loading and configuring both Microsoft EXCEL and the @aGlance/IT Add-In for Microsoft EXCEL. Consult the vendor documentation supplied with the products for information on installing and configuring them.

The @aGlance/IT Add-In for Microsoft EXCEL supports polled monitoring.

SECTION 12 - TAG CONFIGURATION

INTRODUCTION

The tag configuration utility permits the user to configure, add and change the status of tags in the Personal Work Center 90 (PWC90) database. It provides several features that can be applied to all configuration data.

> **NOTE:** All tag changes are recorded as operator actions in the event log. Concordance checking is automatically performed to detect configuration errors. These include data over limits, duplication of tag names or hardware addresses, invalid descriptors, and invalid alarm inhibiting conditions.

The tag configuration utility is invoked by selecting *Server Configuration* from the server pull-down menu in the system status display. This section describes the functions of the server configuration application.

TAGS

The PWC90 console allows on-line tag configuration. Tags can be added or deleted and almost any tag attribute can be changed, then immediately updated by the PWC90 console.

Each entry in the database is in the form of a tag. A tag represents either an analog or digital exception reporting block or a station, device driver, or control block in a PCU module. Define a tag for each process variable that the PWC90 console is to monitor and for each process device that is available for control. Also, a tag can represent a system control module or interface unit.

The PWC90 console does not require a tag to support INFI 90 OPEN module management functions such as tuning and configuration. After a tag is defined, its name is used in other functions to identify the tag.

A tag contains all information required to find a point in the process control configuration (i.e., function block) and to establish communication between it and the PWC90 console. Not all processor function blocks can be assigned a tag.

Each tag type available on the PWC90 console may provide access to one or more function codes (FC). Table 12-1 lists and describes the PWC90 console tag types.

Table 12-1. Tag Types

Tag Type	Support	Function Code[1]	Purpose
ANALOG	Analog exception report	FC 30	Acquires an analog exception reported value.
DADIG	Data acquisition digital	FC 211	Acquires a digital exception reported state providing enhanced alarm management capabilities. It also allows selecting the input source for the function block in a PCU module and writing a user-inserted value to the block.
DANG	Data acquisition analog	FC 177	Acquires an analog exception reported value providing enhanced multiple level alarming as well as deviation and rate alarming. It also allows selecting the input source for the function block in a PCU module and writing a user-inserted value to the block.
DD	Device driver	FC 123	Acquires an exception reported set or reset state for a device. It also allows initiating manual control.
DIGITAL	Digital exception report	FC 45	Acquires a digital exception reported state.
MSDD	Multi-state device driver	FC 129	Acquires an exception reported status for a three-state device. It also allows initiating manual control.
RCM	Remote control memory	FC 62	Acquires an exception reported set or reset state of a device. It also allows initiating device control.
RMCB	Remote motor control block	FC 136	Acquires an exception reported start or stop state of a device. It also allows initiating device control.
RMSC	Remote manual set constant	FC 68	Acquires an exception reported constant value and allows changing the value stored in a PCU module.
STATION	M/A station - basic M/A station - cascade M/A station - ratio Control station	FC 21 FC 22 FC 23 FC 80	Monitors the exception reported variables from a control station. It also allows changing the control output, set point, or ratio index.
TEXT	Text selector	FC 151	Acquires an exception reported text selector message. The message, generated by the PCU module, contains a message number that corresponds to a text string defined on the PWC90 console. It also contains a color and blink option.
TEXTSTR	User-defined data export	FC 194	Enables communication between the PWC90 console and a C language or batch program running in a PCU module to allow transfer of text strings.

NOTE:

1. Refer to the ***Function Code Application Manual*** for more information and for descriptions of function code specifications (Table 1-2 lists instruction numbers).

AVAILABLE TAGS LIST

When tag configuration is invoked, the *Tag Configuration Utility* window appears which contains a list of all available tags in the selected server database (Figure 12-1). The list provides five information fields about the tags:

- Tag Name - a unique name for each tag in the database.
- Description - a text string describing the tag.
- Tag Type - type of the tag (i.e., Analog, Digital).
- Operating status - the status of the tag (Refer to ***Change Operating Status of a Tag*** in this section).
- Modified - the date and time the last modification was applied to the tag.

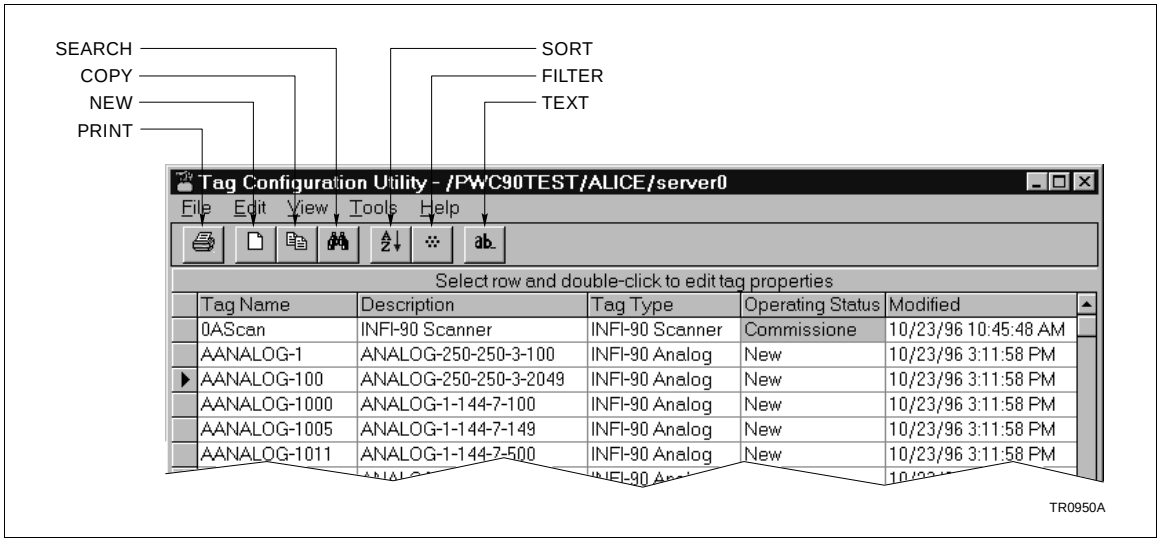


Figure 12-1. Available Tag List Window

The tool bar located in this window provides most of the functions available in the pull-down menus. The following subsections describe the functionality of the server configuration utility.

FILE OPTIONS

Open the *File* menu by clicking on *File* or using ALT-F combination. The *File* options menu provides a *Print, Purge Deleted Tags* and an *Exit* option. Each one of these items is described in the following paragraphs along with keyboard access and the tool bar icons.

Print Option

In the *File* menu, the *Print* option will print a summary report of all tags in the current tag list. Access the print option by clicking on *Print* or press P on the keyboard when the file menu is open. The summary will only include the fields in the tag list. The print configuration will be presented in the standard Microsoft Print dialog, and will be managed by the Windows NT print manager. The print function can also be accessed by the printer icon on the tool bar.

Purge Deleted Tags

The *Purge Deleted Tags* option deletes all the tags labeled *Deleted* in the *Operating Status* column from the database. A list of the deleted tags can be viewed by selecting *Deleted* in the *View* menu. This function is initiated by clicking on *Purge Deleted Tags* in the *File* menu. A dialog box asking the user to confirm the action appears. Click on *OK* button to confirm the purge or click on *Cancel* to exit this function to not remove the deleted tags from the database.

Exit Option

To exit the tag configuration utility click on *Exit* in the *File* menu or press *X* when the file menu is open.

EDIT OPTIONS

Open the *Edit* menu by clicking on *Edit* in the *Available Tags* window menu bar. The *Edit* options menu provides a *New Tag*, *Copy Tag*, *Change Status Of A Tag* and a *Search* option each one of these items is described in the following paragraphs along with keyboard access and the tool bar icons.

New Tag

The New Tag option can be accessed by clicking on *New Tag* in the *Edit* menu or by clicking on the new tag icon in the tool bar. A list of available template tags is displayed. Click on the desired template tag or use the arrow keys to select it. Then double click on the tag to create the new tag.

A series of tabbed pages for the new tag appears. In the database, certain tag attributes are common and have the same function for all tag types although they are defined differently for each type. Other attributes are dependent on the tag type. Each different tag type has a separate tab pages. For an example in this instruction, a Remote Control Memory tag will be used. Refer to Tables 12-2 to 12-12 for an explanation of the tag specific fields for all of the tag types.

Table 12-2. Tag Database - General Tab Fields

Field	Description
Tag name	16-character tag name; appears in all functions to identify the tag.
Description	40-character descriptor normally used to describe the purpose of the tag; appears in most functions.
Customer tag ID	Additional 32-character tag identifier.
Area[1]	Alarm group assignment. A valid entry is from 1 to 64.
Inhibit tag[1]	Configures automatic alarm inhibiting. This tag is used to inhibit alarm indications for selected tag. Alarm inhibiting is based on alarm condition or state of the inhibit tag. A valid entry is any tag name or index number (except an RMSC tag). Leave blank to disable automatic alarm inhibiting. Related field is *Inhibit State*.
Inhibit state[1]	Configures automatic alarm inhibiting. This is the condition of the inhibit tag that triggers alarm inhibiting. Not valid unless a tag name is entered in the *Alarm Inhibit Tag* field. A valid entry is: ALARM = default; inhibiting based on alarm condition. STATE = inhibiting based on digital state. Only valid for a digital type alarm inhibit tag.
Save operator actions:	Determines if operator action events for this tag are saved to disk to appear in a periodic printout of the event log. This also configures operator action events for archiving. An X in the box enables this function. With a blank box this function is disabled.
Permit manual inhibit	Determines if an operator can inhibit the tag from the PWC90 console. An X in the box enables this function. A blank box disables the function.

NOTE:
1. Relates to alarm management.

Table 12-3. INFI 90 Tab Fields

Field	Description
Index	Index number of an Alarm Text Index that is to appear when the tag enters a certain alarm condition. Enter 0 to not associate a comment with a condition.
Return to normal	A valid entry is from 1 to 16, default is 16. Priority one is the highest priority.
Bad quality	A valid entry is from 1 to 16, default is 16. Priority one is the highest priority.
Text	Alarm text string.
Loop	The communication highway for the INFI 90 OPEN system. Valid entry 0 to 250.
Node	An interconnection point on the data highway. Valid entry 0 to 250.
Module	A device in the INFI 90 OPEN control system. Valid entry 0 to 31.
Block	Location of a specific function code in the B90 program. Valid entry 1 to 9998.
ICI index	Index determines an address for export tags. Valid Entry 1 to 30000.

NOTE:
INFI 90 OPEN tab page for Analog export and for Digital Export does not show Loop, Node, Module and Block but does show ICI Index.

Table 12-4. Tag Database - Analog Tab Fields

Field	Description
Engineering units	Non input fields; a PCU module reports the index number that associates an engineering unit of measurement with this tag. A list of valid fixed and user-defined engineering units can be viewed on the EUD configuration page.
Decimal digits	Number of digits to show for this tag. A valid entry is from 0 to 4.

Table 12-4. Tag Database - Analog Tab Fields (continued)

Field	Description
Priority	Relative priority or importance within an alarm group of each alarm condition for this tag. Priority relates to alarm management and the alarm summary. The conditions that can have a priority assigned include: High Low A valid entry is from 1 to 16; default is 16. Priority one is the highest priority.
Index & text	Index and value of an Alarm Text String that is to appear when the tag enters a certain alarm condition. The conditions that can have an Alarm Text Index include: High Low Enter 0 to not associate a comment with a condition.

Table 12-5. Tag Database - Data Aqu. Analog Tab Fields

Field	Description
Digits	Number of digits to show for this tag. A valid entry is from 0 to 4.
Priority	Relative priority or importance within an alarm group of each alarm condition for this tag. Priority relates to alarm management and the alarm summary. The conditions that can have a priority assigned include: High deviation, Low deviation Three-high, Two-low, High rate of change, Two-high, Three-low, Low rate of change A valid entry is from 1 to 16; default is 16. Priority one is the highest priority.
Index & text	Index and value of an Alarm Text String that is to appear when the tag enters a certain alarm condition. The conditions that can have an Alarm Text Index include: Low deviation. Three-high, Two-low, High rate of change, Two-high, Three-low, Low rate of change, High deviation Enter 0 to not associate a comment with a condition.

Table 12-6. Tag Database - Digital and Rem Ctrl Memory Tab Fields

Field	Description	
State:[1]	Zero	Six-character logic state descriptor that is to identify a zero state.
	One	Six-character logic state descriptor that is to identify a one state.
Priority	Relative priority or importance within an alarm group of each alarm condition for this tag. Priority relates to alarm management and the alarm summary. The condition that can have a priority assigned is: Alarm A valid entry is from 1 to 16; default is 16. Priority one is the highest priority.	
Index & text	Index and value of an Alarm Text string that is to appear when the tag enters a certain alarm condition. The condition that can have an Alarm Text Index is: Alarm Enter 0 to not associate a comment with a condition.	
Save state change	Determines if a state change for this tag is saved to disk to appear in a periodic printout of the event log. An X in the box enables this function. A blank box disables the function.	

Table 12-6. Tag Database - Digital and Rem Ctrl Memory Tab Fields (continued)

Field	Description	
First feedback state	Zero	Six-character logic state descriptor that is to identify a zero state for feedback one.
	One	Six-character logic state descriptor that is to identify a one state for feedback one.
Which state is the alarm state	Determines which state is the alarm state for the selected tag. It is selected from a pull-down menu which contains: Zero One Neither	

NOTE:
1. Refer to ***Defining A Logic State Descriptor (LSD)*** in this section for the procedures to call a list of valid logic state descriptors.

Table 12-7. Tag Database - Station Tab Fields

Field	Description
Tuning block	Function block that is to appear in the block details portion of a tuning display (normally a PID block). A valid entry is: 1 to 9998 = block address of the function block providing the station block input or any function block in the PCU module. This allows the tuning display to be requested without any further input.
Priority	Relative priority or importance within an alarm group of each alarm condition for this tag. Priority relates to alarm management and the alarm summary. The conditions that can have a priority assigned include: High deviation Low deviation A valid entry is from 1 to 16; default is 16. Priority one is the highest priority.
Index & text	Index and Value of an Alarm Text String that is to appear when the tag enters a certain alarm condition. The conditions that can have an Alarm Text Index include: High deviation Low deviation Enter **0** to not associate a comment with a condition.
Deviation limit	Information only field.
Set point zero	Information only field

Table 12-8. Device Driver Tab Fields

Field	Description	
Second feedback state	Zero	Six-character logic state descriptor that is to identify a zero state for feedback two.
	One	Six-character logic state descriptor that is to identify a one state for feedback two.

Table 12-9. Multi-State Device Driver Tab Fields

Field	Description	
Logic state	Two	Six-character logic state descriptor that is to identify a two state.
	Three	Six-character logic state descriptor that is to identify a three state.

Table 12-9. Multi-State Device Driver Tab Fields (continued)

Field	Description	
Third feedback state	Zero	Six-character logic state descriptor that is to identify a zero state for feedback three.
	One	Six-character logic state descriptor that is to identify a one state for feedback three.
Fourth feedback state	Zero	Six-character logic state descriptor that is to identify a zero state for feedback four.
	One	Six-character logic state descriptor that is to identify a one state for feedback four.

Table 12-10. Rem Motor Ctrl Block Tab Fields

Field	Description	
First permissive state:	Zero	Six-character logic state descriptor that is to identify a zero state for permissive one.
	One	Six-character logic state descriptor that is to identify a one state for permissive one.
Second permissive state:	Zero	Six-character logic state descriptor that is to identify a zero state for permissive two.
	One	Six-character logic state descriptor that is to identify a one state for permissive two.
Text set	Number of a text set defined through remote motor control block text configuration. This selects the text set that contains the text identifiers for each of ten different error codes that can be returned in an exception report from a remote motor control function block.	

Table 12-11. Tag Database - RMSC Tag Fields

Field	Description
Engineering units descriptor	Non input field; a PCU module reports the index number that associates an engineering unit of measurement with this tag. A list of valid fixed and user-defined engineering units can be viewed on the EUD configuration page.
Decimal digits	Number of decimal places to show for this tag. A valid entry is from 0 to 4.
Low limit	Non input field.
High limit	Non input field.

Table 12-12. Tag Database - TEXTSTR Tag Fields

Field	Description
Maximum text width	Maximum text string length. If this string length is less than the actual length of the exception reported text string, truncation will occur. This is considered to be remote truncation since the communications interface unit of the PWC90 console uses the value to determine the maximum length of the text string it will accept. A valid entry is from 0 to 80.

Table 12-12. Tag Database - TEXTSTR Tag Fields (continued)

Field	Description
Enable control by operator	Determines whether or not operator control of the text string export block in a PCU module is permitted. When enabled, control can be performed by using the keyboard. The results of the operator actions can be seen on the screen. When disabled, the PWC90 console provides information but does not allow control. A valid entry is: Checked = default; enable control. Cleared = disable control.
Priority	Relative priority or importance within an alarm group of each alarm condition for this tag. Priority relates to alarm management and the alarm summary. The conditions that can have a priority assigned include: Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Index & text	Index and value of an Alarm Text string that is to appear when the tag enters a certain alarm condition. The conditions that can have an Alarm Text Index include: Return to normal Alarm Enter 0 to not associate a comment with a condition.

GENERAL TAB

Each tag in the database has a name of up to 16 characters (12 for scanner tags). A tag name identifies the tag throughout all functions. Each process point that is to be monitored or control module that is to be monitored requires a unique tag in the database (Figure 12-2).

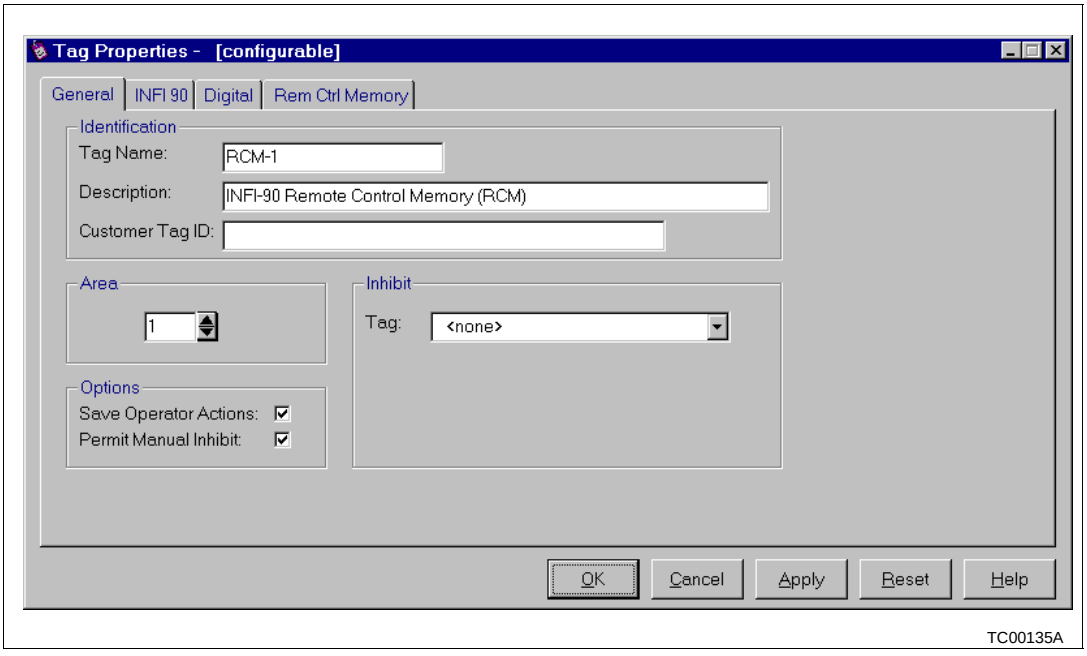


Figure 12-2. General Tab Window

The database allows for a 40-character tag *Description* that describes the purpose of the tag. An additional 32-character *Customer Tag ID* also can be entered to further explain the process point. These descriptions help to identify the tag while on certain displays. To edit the fields click on them and enter the desired text.

Next select an alarm group for the tag by clicking on the up or down arrows beside the *Area* field. The values available are one - 64.

Inhibit Tag field allows for a tag alarm to be inhibited automatically if the inhibiting tag enters a certain state. From the *Inhibit Tag* list box select a tag that will be inhibited and in the *Inhibit State* field enter the state at which the tag alarm will be inhibited. The first entry in the *Inhibit Tag* list box is a blank string. Use this string to disable the *Inhibit Tag* option.

Click on the *Save Operator Action* check box with the mouse or when the box has keyboard focus select and clear the check box with the space bar. When the check box is set the operator actions will be saved to a file for use in summaries.

Click on the *Permit Manual Inhibit* check box with the mouse or when the box has keyboard focus, select and clear the check box with the space bar. When the check box is set the *Permit Manual Inhibit* is enabled. This option allows the operator the ability to inhibit the alarms of this tag. The operator must have the proper security level to use this function (refer to ***Server and Tag Access*** in Section 2.

There are five buttons at the bottom of the tab page each one is explained below.

OK - Applies the changes and exits the tab page and goes to the previous tab page or window.

Cancel - Exit without applying the changes to the tag.

Apply - Applies the changes to the tag without exiting the tab page.

Reset - Click on this button to reset all of the fields to their default values.

Help - Displays a help screen that explains the functions of the tab page.

> **NOTE:** When *Apply* or *OK* is clicked and there is an error in one of the values a dialog box appears defining the error.

INFI 90 TAB

To switch to the INFI 90 OPEN tab page, click on *INFI 90 OPEN*. In this window the *Return to Normal* and the *Bad Quality Priority* are set. This is done by using the arrow keys beside the respective field to raise and lower the value (Figure 12-3).

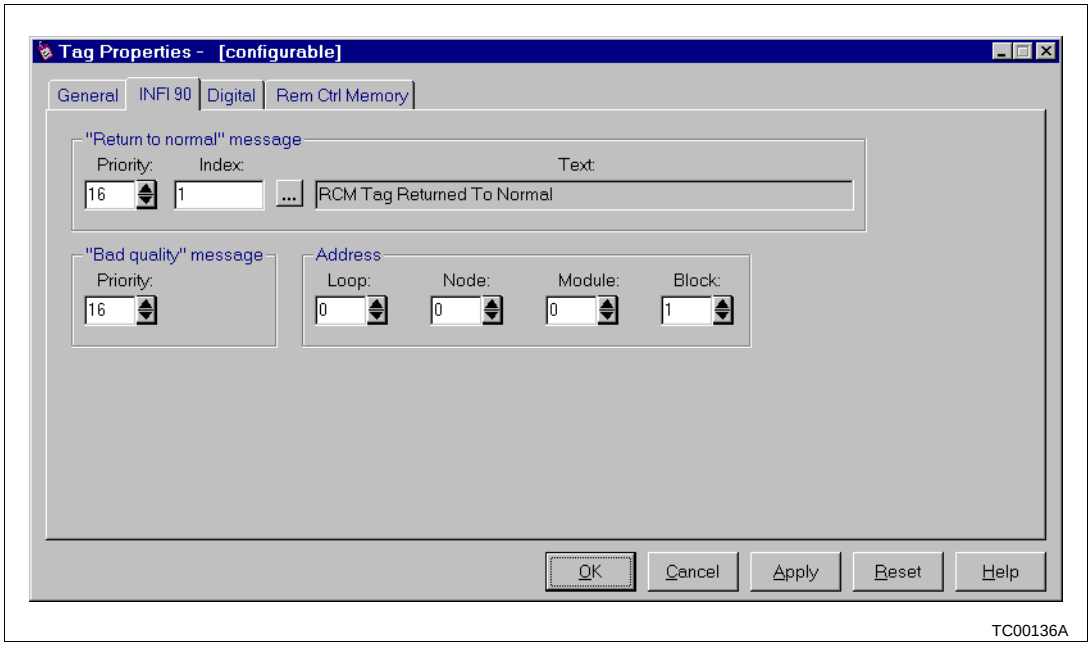


Figure 12-3. INFI 90 Tab

The *Alarm Comment* text number for the *Return to Normal* Alarm Comment can be set by clicking on the *Index* text field and entering the text number. Then click on the button beside the text field to display the configured text in the *Text* window. If an incorrect Index number is entered a dialog box is displayed stating the error and the index value returns to its default value. The alarm priority index numbers are set by the Indexed Text configuration function explained in ***INDEXED TEXT*** in this section.

Next the INFI 90 OPEN address for the tag is entered in the *Loop, Node, Module* and *Block* fields. This is done by using the arrow keys beside each field to raise or lower the number, or by clicking on the field and entering a number from the keyboard. Table 12-13 gives details on each of these fields. If an incorrect value number is entered in any of the fields a dialog box is displayed stating the error and the value returns to its default value of zero.

Table 12-13. INFI 90 OPEN Address Values

Field Name	Description	Possible Values
Loop	The communication highway for the INFI 90 OPEN system.	0 to 250
Node	An interconnection point on the data highway	0 to 250
Module	A device in the INFI 90 OPEN control system.	0 to 31
Block	Location of a specific function code in the Multi-Function processor program.	1 to 9998

DIGITAL TAB

Open the Digital tab page by clicking on the tab page labeled *Digital* (Figure 12-4). The digital tab page provides the ability to set the alarm priority and the index text for the alarm. Also, the logic state descriptors and the *Save State Change* option can be set in this tab.

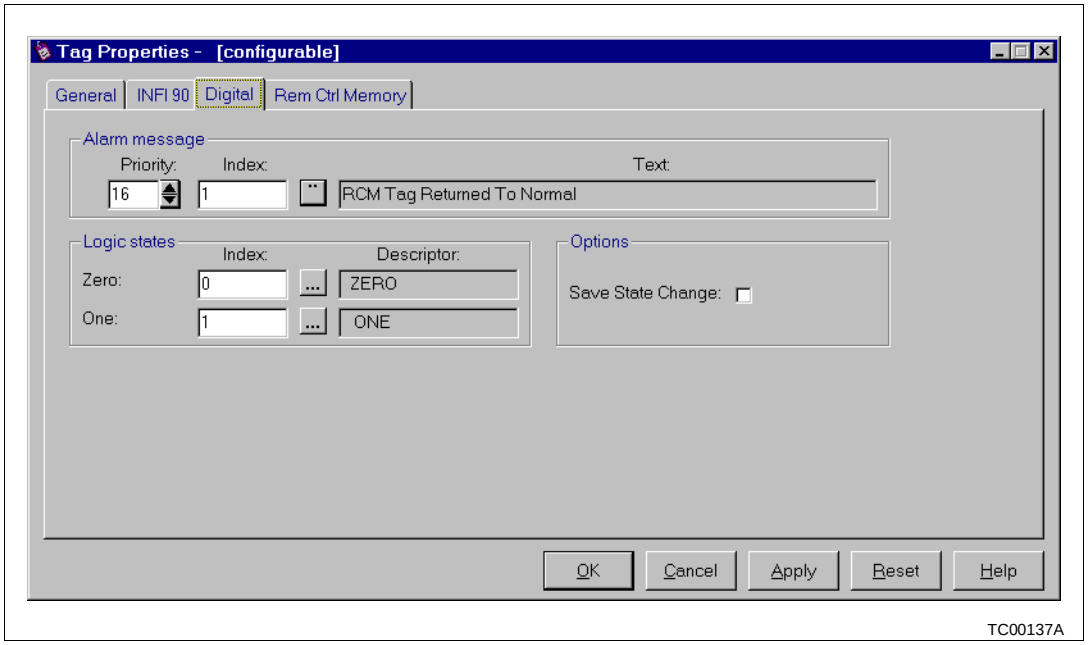


Figure 12-4. Digital Tab Window

The *Alarm Priority* of the tag can be set in the *Alarm Priority* text field by using the arrow keys to raise and lower the value. The values range from one to 16 and the default value is 16 (lowest priority).

Also the text number for the *Alarm Priority* can be set by clicking on the *Index* text field and entering the text number. Then click on the button beside the text field to display the configured text in the *Text* window. If an incorrect Index number is entered a dialog box is displayed stating the error and the index value returns to its default value of nothing. The alarm

comment index numbers are set by the text configuration function explained in ***INDEXED TEXT*** in this section.

The next two fields are the *Index* fields for *Logic State Zero* and *Logic State One*. Click on the field and enter an index number. Then click on the button beside the text field to display the configured text in the *Description* window. If an incorrect Index number is entered a dialog box is displayed stating the error and the index value returns to its default value of nothing. The *Logic State Zero* and *Logic State One* index numbers are set by the text configuration function explained in ***INDEXED TEXT*** in this section.

Click on the *Save State Changes check box* with the mouse or when the box has key board focus select and deselect it with the space bar. When the check box is set the state changes will be saved to a file for use in summaries.

REMOTE CONTROL MEMORY

Open the Remote Control Memory tab page by clicking on the tab labeled *Rem Ctrl Memory*. The two fields in this tab page are the *Index* fields for *First Feedback State Zero* and *First Feedback State One*. Click on the field and enter an index number. Then click on the button beside the text field to display the configured text in the *Description* window. If an incorrect Index number is entered a dialog box is displayed stating the error and the index value returns to its default value of nothing. The *First Feedback State Zero* and *First Feedback State One* index numbers are set by the text configuration function explained in ***INDEXED TEXT*** in this section (Figure 12-12).

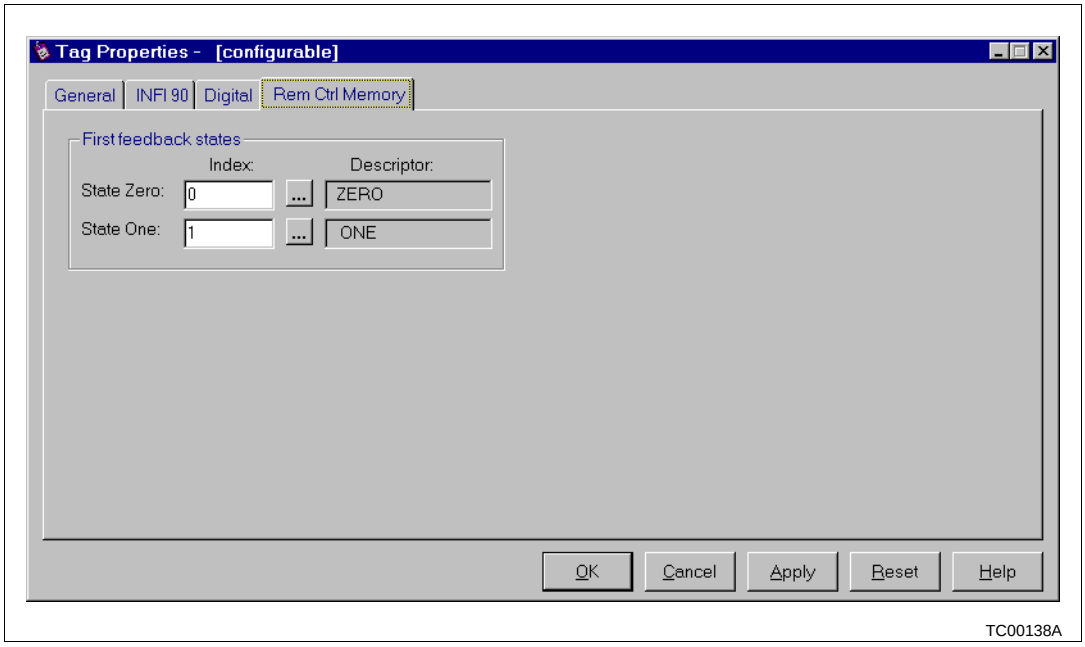


Figure 12-5. Remote Control Memory Window

Copy Tag

To copy a tag, select a tag in the *Available Tag* window by clicking on it. Then select *Copy Tag* from the *Edit* menu or click on the copy icon on the tool bar.

A tab window for the copied tag appears. In the database, certain tag attributes are common for all tag types although they are defined differently for each type. Other attributes are dependent on the tag type. Each different tag type has a different tab window. Tables 12-2 to 12-12 explain the tag specific fields for all of the tag types. Change the name and other attributes of the tag as needed. When finished click on *Apply* to apply the changes and remain in the tag configuration tab or click *OK* to apply the changes and exit. Click on *Cancel* to abort the tag copying function without making the changes to the tag.

Change Operating Status of a Tag

To change the status of a tag, select a tag in the main window by clicking on it. Then select *Change Operating Status* from the *Edit* menu or click the status field of the desired tag and a arrow icon will be enabled. Click on this icon. Both of these actions invoke a *Change Operating Status* dialog box select the status change for the tag and click on *OK*. The status change options vary dependent on the current status of the tag. For example: if the tag status is *Deleted* the only option available in the *Status Change* dialog is *New*.

> **NOTE:** Multiple tags can be selected by clicking on one tag and dragging across the desired tags while the mouse button is depressed.

Table defines the operating statuses available. Only tags marked New or Commissioned will actually be loaded into the PWC90 system and be available to users. The other statuses are for use in predefining default values for different tag types(templates) or for tracking tags which are not currently in service.

Table 12-14. Operating Status Descriptions

Operating Status	Description
New	Indicates that a tag is currently in use, but in a precommissioned state.
Commissioned	Indicates that a tag has been verified as correct and commissioned for use.
Deleted	Indicates that a tag has been deleted from use. Deleted tags are not physically removed from the tag configuration database until *Purge Deleted Tags* operation is performed. In this way, deleted tags may be reinstated at a later date.
Template	Indicates that this tag entry is not a real tag, but is a template for copying when creating new tags. Tag templates may be defined which may contain default values for any tag field. Creating a new tag is really just a copy of an existing template tag (either pre-defined or user-defined).

Table 12-14. Operating Status Descriptions (continued)

Operating Status	Description
Ignored	Indicates that this tag which has not yet been commissioned, should be ignored by the system.
Suspended	Indicates that this tag, which has already been commission, should be ignored by the system.
Error	Indicates that this tag, which has been imported into the tag configuration database using the *Tag Importer*, was found to contain errors, and will be ignored by the system. Once the error has been corrected, the *Operating Status* of the tag will go to *New*.

Search

The Search function is activated by selecting *Search* from the *View* menu or by clicking on the search icon in the tool bar. A *Tag Search* dialog box is displayed. Enter the full name of the desired tag in the *Enter Tag Name* field. If the full tag name is not known type the first few characters in the tag name and the search function will find the first occurrence of these characters in the tag name list. To start the search click on *OK*. To exit the search function click on *Cancel*.

VIEW OPTIONS

Display the *View* menu by clicking on *View* in the *Available Tags* window menu bar. The *View* option allows the user to *Sort* and *Filter* the tags by the use of SQL query statements.

Sort

Open the *Sort* window (Figure 12-6) by clicking on *Sort* in the *View* pull-down menu or click the sort icon in the tool bar. In the *Sort* window select the options by which the tags should be sorted. All the arrangements are done in alphabetical order except *Modified* which is done by date and time. To select a sorting option click on the desired Field Name in the *Field Names* list. The selected item will be highlighted. Then click on *ADD->* to add it to the *Sort Order* field. More than one option can be added to this clause. To clear the *Sort Order* click on Clear. The order of the sorting can also be set to *Ascending* or *Descending*. For example if the modified option is selected with an *Ascending* order it will go from oldest date to the newest and with *Descending* order it will go from the newest date to the oldest.

> **NOTE:** The *Ascending* option is the default. If the *Descending* option is desired it must be selected before the Field Name is added to the clause.

Another method of sorting the tag list is by clicking on the heading of each column. This function will sort alphabetically

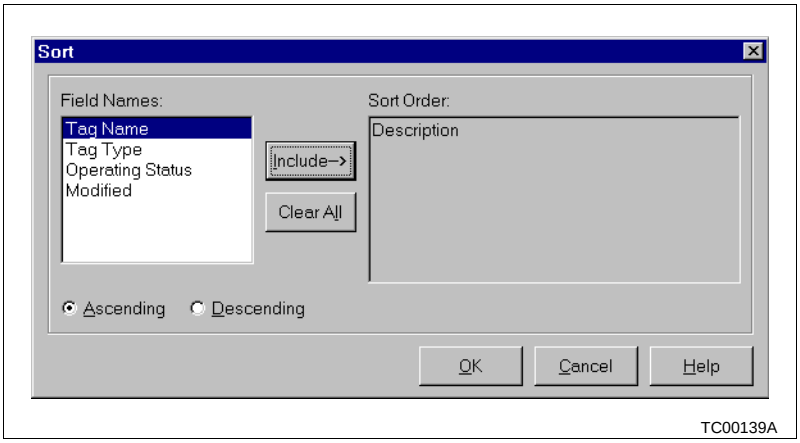


Figure 12-6. Sort Window

for names and text or numerically for date depending on the column heading selected. This option only arranges the items in ascending order.

Filter

Open the filter option by clicking on *Filter* in the *View* menu. This displays the *Filter* window (Figure 12-7). The Filter function allows creation of a custom filter that is applied to the tag list.

To create a Where filter:

1. Click on the *Field Name* pull-down menu and select a field to filter the tags by.

2. Click on the *Operation* pull-down menu and select an operator for the Where clause.

3. Click on the *Value* field and enter a value that corresponds to the *Field Name* selected and the *Operator*.

> **NOTE:** For Tag Type and Operating Status the value is selected from a list.

4. Click on the logic criterion for the entered clause.

5. When the Where clause has been completed click on *OK* to filter the Available Tag list or click on *Cancel* to exit this function without filtering.

To clear the *Resulting Filter* field click on *Clear All*.

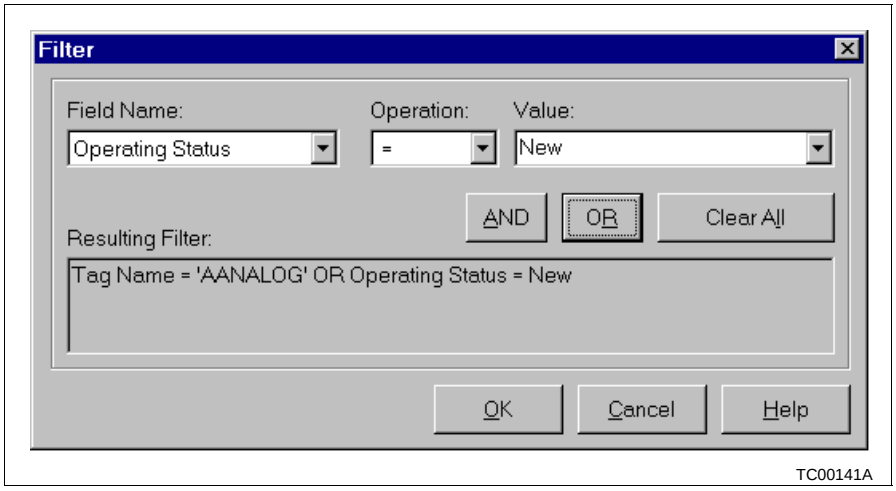


Figure 12-7. Tag List Filtering Window

Operating Status Filters

The options in the *Operating Status* dialog are:

- New.
- Commissioned.
- Deleted.
- Template.
- Ignored.
- Suspended.

When one of these options is selected only the tag names that it describes will be listed. For example if *Deleted* is selected only the deleted tags will be listed. To select any of these seven options just click on the desired one. When one of these options are selected a check appears to the left of the option in the *View* menu. To deselect this view option click on it a second time. Check all options to view the full list.

> **NOTE:** Using the filter command overrides these operating status filters and disables them the menu items.

TOOLS

The *Tools* menu contains the *Configure Text...*, the *Import Configuration* and *Import Trends* functions. The *Configure Text...* function allows the user to edit the text indexes for the database. The *Import Configuration* function is used to import configurations from other databases. The *Import Trends* converts OIS/PCV configuration databases into XDC historical databases.

Import Trends

The PWC90 Trend Importer converts an OIS/PCV configuration database in to a ***.XDC*** historical database configuration file. The application generates a log file for recording each incidence of a change or rejected trend configuration. Three input file types of .***TTR*** files are supported L.3, S.1, and PVC.

Start the application by clicking on *Trend Importer* in the *Tools* menu. When the application is started the user is presented with the window shown in Figure 12-9.

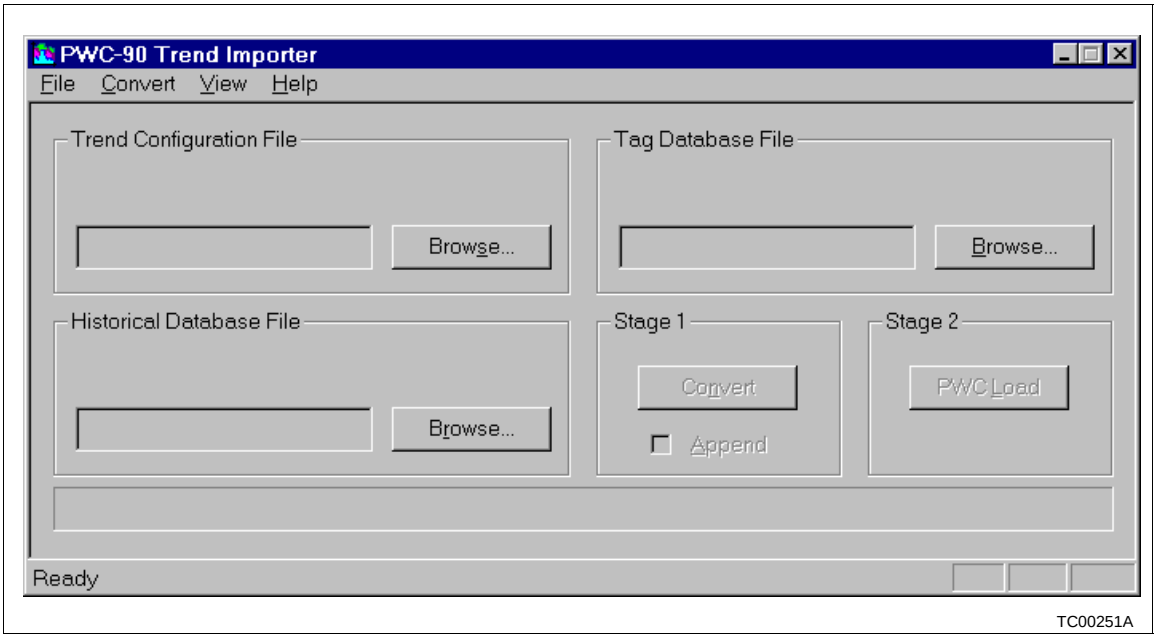


Figure 12-8. Trend Importer Default Window

The only options available when the application is first opened are the browse buttons. These buttons open a typical Windows NT file selection window.

To convert a configuration database:

1. Select a Trend Configuration (.***TTR***) file by using the browse button or *Select Source* option in the file pull-down menu.

2. Select a Historical Database (.***XDC***) file by using the browse button or *Select Target* option in the file pull-down menu.

3. Select a Tag Database (.***TTG***) file by using the browse button or *Select .TTG file* option in the file pull-down menu.

NOTE: Only after these three files have been selected will the Convert button be enabled.

4. Click on *Convert* to convert the Trend to the Historical database file.

> **NOTE:** Only after the files have been converted will the PWC Load button be enabled.

5. Click on *PWC Load* to load the Historical database file into the PWC90 system.

Import Configuration

The *Tag Import Utility* enables a user to add configuration data from other Elsag Bailey INFI 90 OPEN consoles and offline configuration tools (SLDG and WLDG in particular).

The configuration data is provided as files of ASCII text strings and consists of Alarm Comments, Engineering Unit Descriptors, Logic State Descriptors, Remote Motor Control Block Error Code Text Sets, Text Selections, and Tag Definitions. The Tag Definitions may be in one of two formats: SLDG BCS L.3 and SLDG BCS S.1. The formats of the other files are fixed. Alarm Comments are embedded in Tag Definitions for the L.3 format, therefore this format does not permit importing a separate file of Alarm Comments.

Files with the extension ***.CP*** must be assumed to use run-length compression (SLDG standard) during importation.

The *Tag Importer* utility requires the use of a temporary working directory to perform file modifications (i.e. file decompression, record delimiter insertion). The *Tag Importer* utility must use the temporary working directory whose name is specified in one of the user environment variables defined as ***temp*** or ***tmp*** in Main | Control Panel | System. Normally these variables are defined by Windows NT to specify a directory at the system root called ***\temp***. However, these variables may be configured otherwise.

Start the application by clicking on *Tag Importer* in the *Tools* menu. When the application is started the user is presented with the window shown in Figure 12-9.

1. In the *Definition Type* field select the type of tags to be imported. The selections are:

- RMCB error code text sets.
- Alarm comments.
- Text selections.
- Engineering unit descriptors.
- Logic state descriptors.
- Tag definitions.

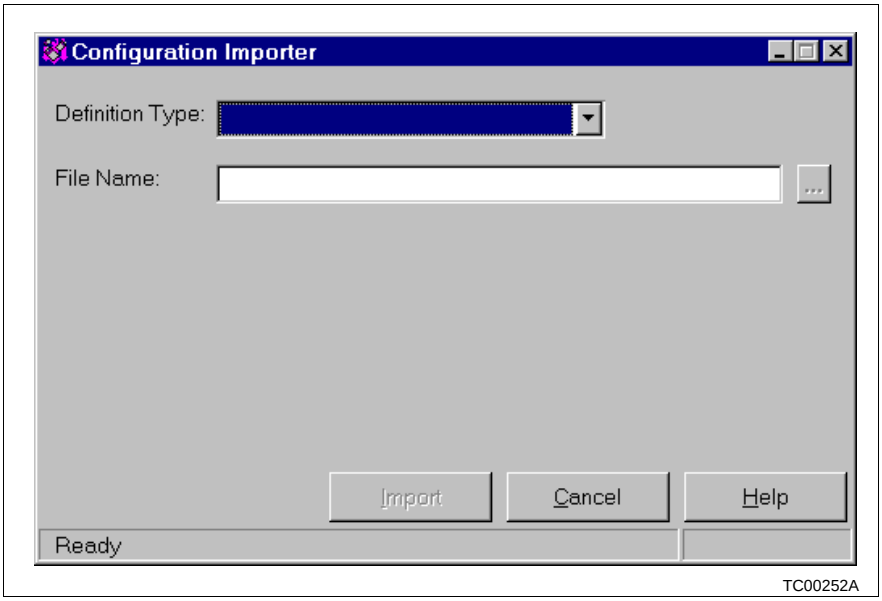


Figure 12-9. Tag Importer Default Window

2. When *Tag Definitions* is selected three Check boxes appear in the window (Figure 12-10). If any of the following functions are needed click on the appropriate check box.

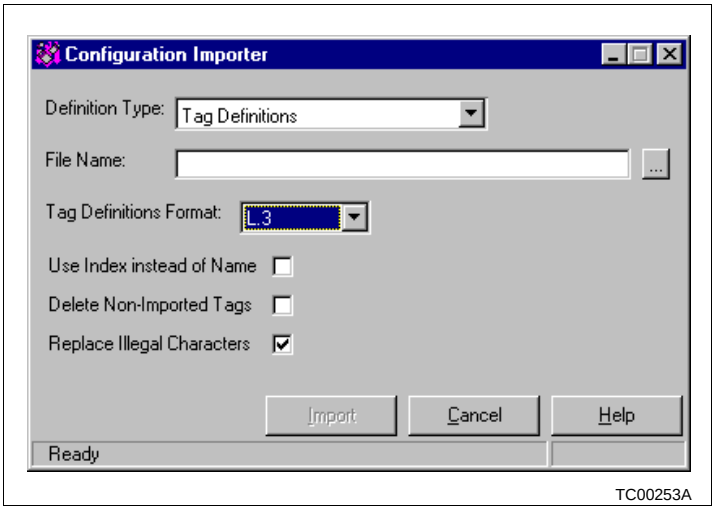


Figure 12-10. Tag Importer Window with Check Boxes

- *Use Index instead of Name* - (if enabled) When Tag Definitions are imported the Tag Index field (***EB_TAGINDEX***) will be used, instead of the Tag Name field (***EB_TAG***).

- *Delete Non-imported Tags* - (if enabled) When Tag Definitions are imported the existing tag definitions in the database, will have their Operating Status set to Deleted (***EB_LIFECYCLE*** = 100). All tag definitions which are

Updated and Appended, will have their Operating Status set to New (***EB_LIFECYCLE*** = 0).

- *Replace Illegal Characters* - (if enabled) When Tag Definitions are imported any characters other than 'A' through 'Z', 'a' through 'z', '-', '_', or trailing blanks, will be replaced by '_' in Tag Names (***EB_TAG*** field of ***CEbTag*** and ***impFlatDatabase*** tables).

3. After selection the *Definition Type*, click on the browse button beside the *File Name* field. A common Windows NT file selection window appears. Select the path and the filename to be imported.

4. Select the *Tag Definitions Format*: S.1 or L.1.

5. Click on *Import* to perform the operation.

A pop-up window appears confirming the completion of the Tag Import Function.

Configure Text

The *Configure Text...* function allows for the configuration of text indexes for Alarm Text, Text Selections, Engineering Unit Descriptors, Logic state Descriptors, Remote Motor Control Block (RMCB) error code text sets (only for INFI 90 OPEN RMCB tags) and Substitutable text. To start the Text Configuration function click it in the Tools menu or click on the [Text] icon in the tool bar.

INDEXED TEXT

The *Indexed Text* tab page contains the text type name box that contains the selectable options for the table on the page. They are:

- Alarm comment.
- Text selection text.
- Engineering unit description.
- Logic state descriptor.
- Alarm priority text.

Each selection invokes a table with an *Index* and a *Text* column. Below the table is a set of buttons which are used to scroll through the table. Pressing the outer most buttons of this control immediately moves the table to the beginning or the end. The other two buttons scroll through each table entry. The text field in this control displays the record number that has cursor focus out of the total number of records in the table (Figure 12-11).

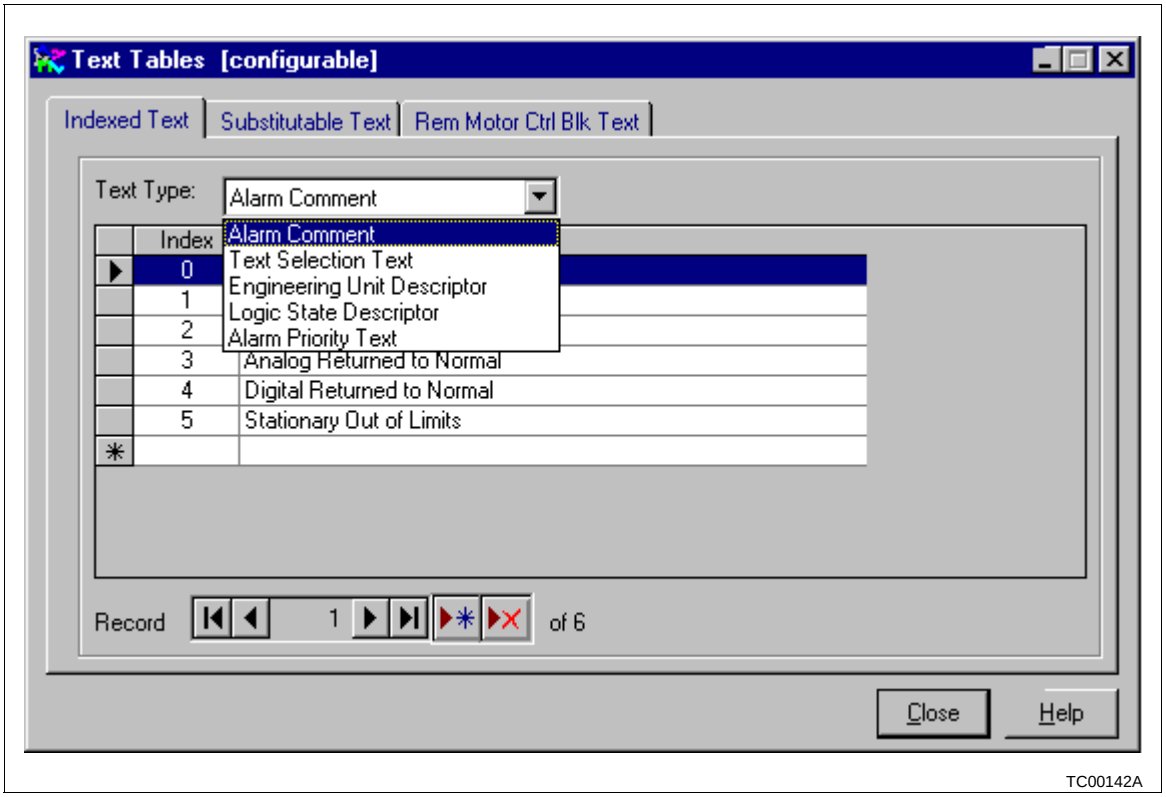


Figure 12-11. Index Text Window

Each one of the selections is explained in the following sections.

Alarm Comment

The *Alarm Comment* is the default text type name in the *Text Type Field*. The table has two columns; one labeled Index the other labeled Text. The maximum number of entries in this table is 20,000. An *Alarm Comment* Index is associated with an alarm condition of a tag in the database. An alarm text can be created for each alarm condition of a tag. Each tag type has different possible alarm conditions, which include:

- Return to normal (RTN).
- Alarm (A).
- High alarm (H).
- Two-high alarm (2H).
- Three-high alarm (3H).
- Low-alarm (L).
- Two-low alarm (2L).
- Three-low alarm (3L).
- High deviation (HD).
- Low deviation (LD).
- High rate of change (HR).
- Low rate of change (LR).
- Hardware failure (HF).

All *Alarm Comment* indexes have an assigned index number. The index number allows using a single comment with several tags. In this way, a comment does not have to be redefined for each tag. Enter an *Alarm Comment* Index number for each alarm condition of a tag during its configuration to associate a comment with a condition.

To add a *Alarm Comment* index number go to the bottom of the table and use the blank row. Click the cell in the *Index* column and insert a number. Then click on the *Text* cell beside the number and enter the desired text for this number.

Text Selection Text

To display the text choice table, select *Text Selection Text* from the *Text Type* box. The table has two columns; one labeled *Index* the other labeled *Text*. There can be up to 10,000 text indexes in this table. These index numbers are used with the Text Selector function code.

To add a *Text Selection Text* index number go to the bottom of the table and use the blank row. Click the cell in the *Index* column and insert a number. Then click on the *Text* cell beside the number and enter the desired text for this number.

TAG DESCRIPTORS

Tag descriptors are used to present the unit of measurement for a process value and the current state for a process device. There are two types of descriptors: Engineering unit descriptors (EUD) and logic state descriptors (LSD).

Engineering Unit Descriptors (EUD)

Engineering unit descriptors relate to tag types that present analog values. They describe the unit of measurement (i.e., DEG F, GPM, AMPS, LB/HR) for a value received from the process. A descriptor follows the tag value throughout all PWC90 console functions.

A list of common engineering units is provided. A total of 256 engineering unit descriptors can be defined in the database: Zero through 15 are fixed and 16 though 255 can be user-defined. Table 12-15 lists the fixed engineering unit descriptors and their index numbers.

The actual reporting of engineering unit descriptors comes from PCU modules. After an engineering unit is defined in the module configuration, the module sends an index number to the PWC90 console to identify the engineering unit associated with an exception reported value. This index number is then cross referenced with the database list of descriptors. Since the modules report the engineering unit descriptor index number,

Table 12-15. Engineering Unit Descriptors

Index	Descriptor
0	(blank)
1	(blank)
2	%
3	DEG F
4	DEG C
5	PSIA
6	PSIG
7	IN H2O
8	GPM
9	CFS
10	CFM
11	LB/HR
12	GAL
13	AMPS
14	IN HG
15	KLB/HR

all devices on a common communication highway should use the same EUD list.

Additional engineering unit descriptors can be added to the database by using the *Engineering Units* option.

Defining an Engineering Unit Descriptor (EUD)

Use the *Engineering Units* option to define an engineering unit descriptor.

> **NOTE:** Descriptors zero through 15 are fixed and cannot be changed through this configuration.

To add a *Engineering Unit* descriptor index number go to the bottom of the table and use the blank row. Click the cell in the *Index* column and insert a number. Then click on the *Text* cell beside the number and enter the desired text for this number.

Logic State Descriptors (LSD)

Logic state descriptors relate to tag types that present logic states for digital devices. These descriptors show the current logic state (i.e., on or off, zero or one, run or stop, or closed or open) of a device. A descriptor, after being defined for a tag, follows the tag throughout all PWC90 console functions.

Unlike an engineering unit descriptor, a PCU module does not report an index number of a logic state descriptor. The tag database is referenced for the correct descriptor.

A list of common logic state descriptors is provided. A total of 256 logic state descriptors can be defined in the database: Zero through 15 are fixed and 16 through 255 can be user-defined. Table 12-16 lists the fixed logic state descriptors and their index numbers.

Table 12-16. Logic State Descriptors

Index	Descriptor
0	ZERO
1	ONE
2	ON
3	OFF
4	NO
5	YES
6	CLOSED
7	OPEN
8	LOW
9	HIGH
10	EMPTY
11	FULL
12	RUN
13	STOP
14	TRIP
15	(blank)

Additional logic state descriptors can be added to the database by using the *Logic State Descriptors* option.

Defining A Logic State Descriptor (LSD)

Use the *Logic State Descriptors* option to define a logic state descriptor.

> **NOTE:** Descriptors zero through 15 are fixed and cannot be changed through this configuration. LSD 15 is defined as blank.

To add a *Logic State* descriptor index number go to the bottom of the table and use the blank row. Click the cell in the *Index* column and insert a number. Then click on the *Text* cell beside the number and enter the desired text for this number.

Alarm Priority Text

The *Alarm Priority Text* is the last text type name in the *Text Type Field*. The table has two columns; one labeled Index the other labeled Text. There is 16 provided Alarm Priorities. An *Alarm Priority* index is associated with an alarm priority of a tag in the database. An alarm priority text appears in the first field of the Mini-alarm window and in the alarm review display. Each alarm priority can be assigned different possible indexes, which include shown in Table 12-17.

Table 12-17. Alarm Priorities

Index	Descriptor
1	DIRE
2	dire
3	CRUCIAL
4	crucial
5	CRITICAL
6	critical
7	ALERT
8	alert
9	WARNING
10	warning
11	ADVISE
12	advise
13	NOTIFY
14	notify
15	INFORM
16	inform

The Alarm Priority text is another method of determining the priority of a problem in the process.

SUBSTITUTABLE TEXT

To select the *Substitutable Text* tab page click on the *Substitutable Text* tab. This tab page contains a table with a list of atoms and their description. Below the table is a set of buttons which are used to scroll through the table. Clicking the outer most buttons of this control immediately moves the table to the beginning or the end. The other two buttons scroll through each entry. The text field displays the number that has cursor focus and the maximum number of records in the table. The *Substitutable Text* tab page is shown in Figure 12-12.

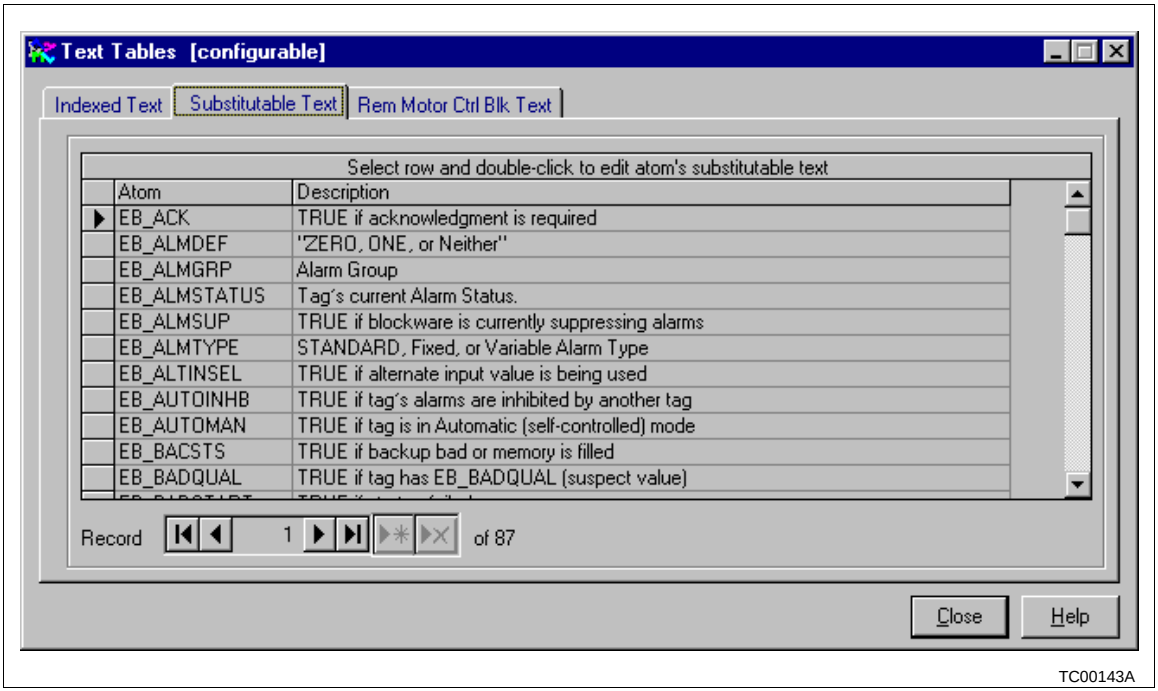


Figure 12-12. Substitutable Text Tab Page

To edit the text, choose the atom to be edited by selecting the atom by the use of the arrow keys or the mouse then double click the mouse on the selected atom.

This displays a table with three columns *Value*, *Tag Type* and *Text*. Click on the row beside the * or click *New Record* to enter a new row or click on the cell of an existing value to edit it. Click on the grayed box to the left of the Value column to select an entire row. When the entire row is selected it can be deleted by pressing **Delete** on the key board or clicking on *Delete*. A confirmation dialog box will appear to confirm this action.

There are two buttons at the bottom of the table window; they are *Done* and *Default*. Press *Default* to insert the default values into the table. *Done* accepts the values into the database and closes the window.

REMOTE MOTOR CONTROL TEXT

To select the *Rem Motor Ctrl Blk* text tab page click on the *Rem Motor Ctrl Blk Text* tab. This tab page contains a horizontally scrolling list displaying different *Rem Motor Ctrl Blk* sets. Each of the lists has a set number at the top with a set name directly underneath it. The set name is followed by a list of the ten text strings for an RCMB tag. There is a maximum of 99 sets permitted (Figure 12-13).

The *Remote Motor Control Text* option defines text sets that contain text strings for remote motor control function blocks

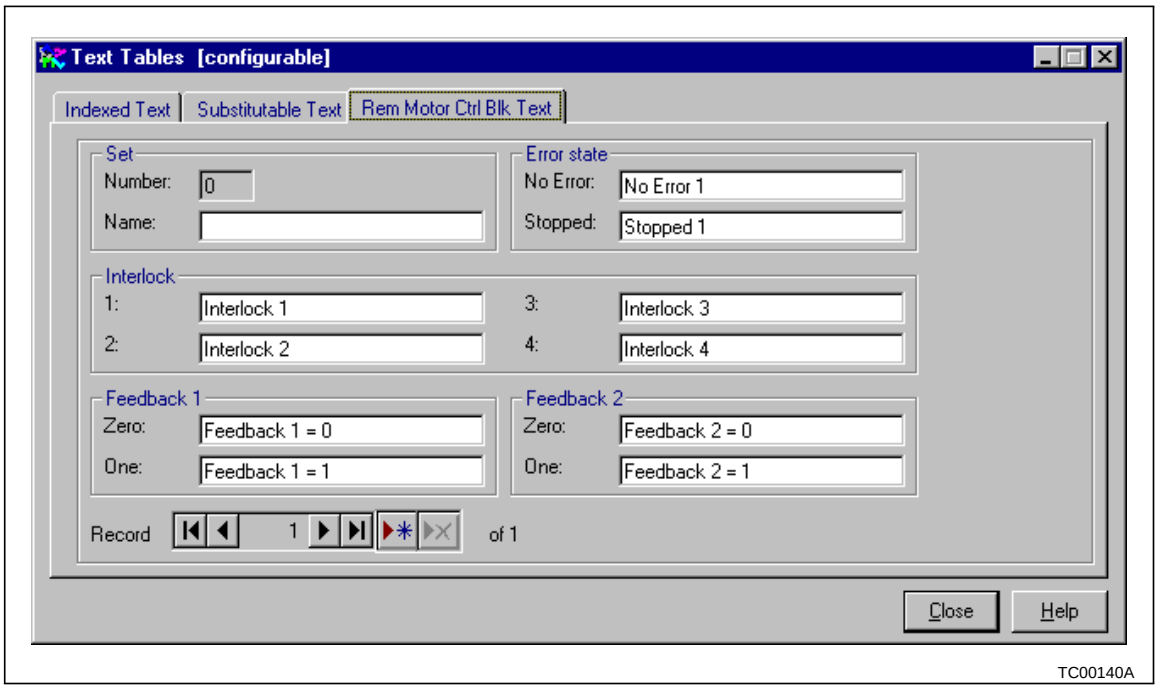


Figure 12-13. RCMB Text Tag

(FC 136). The function block can report any of ten different text strings that identify its current operation. These error codes are reported to identify the condition that caused a bad start of a device. An RMCB tag must be configured for the PWC90 console to receive an exception report. Error codes and status reported include:

- No error.
- Stopped.
- Interlock one.
- Interlock two.
- Interlock three.
- Interlock four.
- Feedback one = 0.
- Feedback two = 0.
- Feedback one = 1.
- Feedback two = 1.

NOTE: The interlock codes indicate not set or logic zero conditions for each of four interlocks.

Each error code text set has an assigned number. Up to 100 sets can be defined. A *Text Set* field for an RMCB tag selects which text set is to be used for the tag. The specific text string that displays depends on the error code returned in an exception report from the RMCB block.

To add a text set click on *New Record.* This creates a text set with a number equal to (maximal used + 1). Edit the informa-

tion in the fields of the new text set and click on Ok to save the changes.

To delete the current set click on *Delete Record*. Set zero cannot be deleted.

> **NOTE:** Operations are performed immediately and can not be canceled.

APPENDIX A - PWC90 TAG.ATOMS

ATOM SUFFIXES

There are two atom suffixes that are used with the PWC90 tag.atom combinations.

1. The ***_N*** suffix is used to identify the format as numeric. Hence the complete identification for a numeric tag looks like this:

    tag.atom_N

2. To identify a tag.atom combination as a string format, no suffix is used. Hence the complete identification for a numeric tag looks like this:

    tag.atom

TAG.ATOM LIST

This appendix contains lists of all of the available tag atoms in the PWC90 system.

Table A-1. INFI 90 Analog

Atom Name	Write	Text	Numeric	Description
C_PVHIAP	No	No	Yes	Alarm if tag's value is above this limit.
C_PVLOAP	No	No	Yes	Alarm if tag's value is below this limit.
EB_CALIBSTS	No	Yes	Yes	Calibration Status.
EB_HICMT	No	Yes	No	High alarm state comment index.
EB_HIPRIO	No	Yes	Yes	Priority of high alarm message (1 to 16).
EB_LOCMT	No	Yes	No	Low alarm state comment index.
EB_LOPRIO	No	Yes	Yes	Priority of low alarm priority message (1 to 16).

Table A-2. INFI 90 Analog Base

Atom Name	Write	Text	Numeric	Description
C_APV	Yes[1]	No	Yes	Last reported value for tag.
C_RANGHI	No	No	Yes	zero+span of tag's value in engineering units.
C_RANGLO	No	No	Yes	Base value (zero) of tag's value in engineering units.
C_UNITS	No	Yes	No	Engineering units of measure index.
EB_DECDIG	No	No	Yes	Number of decimal places to display (0 to 7).
EB_SPAN	No	No	Yes	Span of tag's value in engineering units.

NOTE:
1. Only if EB_SCANST is False.

Table A-3. INFI 90 Analog Export

Atom Name	Write	Text	Numeric	Description
C_APV	Yes	No	Yes	Last reported value for tag.
C_PVHIAP	No	No	Yes	Alarm if tag's value is above this limit.
C_PVLOAP	No	No	Yes	Alarm if tag's value is below this limit.
C_RANGHI	No	No	Yes	Zero+Span of tag's value in engineering units.
C_RANGLO	No	No	Yes	Base value (zero) of tag's value in engineering units.
C_UNITS	No	Yes	No	Engineering units of measure index.
EB_DECDIG	No	No	Yes	Number of decimal places to display (0 to 7).
EB_HICMT	No	Yes	No	High alarm state comment.
EB_HIPRIO	No	Yes	Yes	Priority of high alarm message (1 to 16).
EB_LOCMT	No	Yes	No	Low alarm state comment.
EB_LOPRIO	No	Yes	Yes	Priority of low alarm message (1 to 16).
EB_SPAN	No	No	Yes	Span of tag's value in engineering units.

Table A-4. INFI 90 ASCII STRING

Atom Name	Write	Text	Numeric	Description
EB_TXTSTR	Yes	Yes	No	Text (data string) currently limited to 80 bytes.

Table A-5. INFI 90 Base

Atom Name	Write	Text	Numeric	Description
EB_BADPRIO	No	Yes	Yes	Priority of Bad Quality messages.
EB_BLOCK	No	No	Yes	Block Address of tag (Currently 1 to 30,000).
EB_DISEST	No	Yes	Yes	When true, indicates that the tag is not established in the CIU.
EB_LOOP	No	No	Yes	0 for Plant Loop, 1 to 250 for INFI-NET.
EB_MODULE	No	No	Yes	Currently 0 to 31.
EB_NODE	No	No	Yes	1 to 64 for Plant Loop, 1 to 250 for INFI-NET.
EB_RTNCMT	No	Yes	No	Return to normal state comment.
EB_RTNPRIO	No	Yes	Yes	Priority of Return To Normal message.

Table A-6. INFI 90 DAANG

Atom Name	Write	Text	Numeric	Description
EB_ALMSUP	Yes	Yes	Yes	TRUE if blockware is currently suppressing alarms.
EB_ALMTYPE	No	Yes	No	STANDARD, Fixed, or Variable Alarm Type.
EB_ALTINSEL	Yes	Yes	Yes	TRUE if alternate input value is being used.
EB_AUTOMAN	Yes	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_CNTRREF	No	No	Yes	Display reference for bidirectional dynamic bar.
EB_COMMQUAL	No	Yes	Yes	TRUE if communications are bad.

Table A-6. INFI 90 DAANG (continued)

Atom Name	Write	Text	Numeric	Description
EB_CONSTR	No	Yes	Yes	TRUE if tag's value is currently being constrained.
EB_CONSTRPER	No	Yes	Yes	TRUE if the input value constraint is enabled.
EB_DEVALM	No	Yes	Yes	Deviation Alarm Status (0-Normal, 1-Low Alarm).
EB_FORCEXR	Yes[1]	No	Yes	Force block to send XR (always returns 0).
EB_HARDFAULT	No	Yes	Yes	True if blockware detects a hardware failure.
EB_HI2CMT	No	Yes	No	High high alarm state comment.
EB_HI2PRIO	No	Yes	Yes	Priority of high high alarm message.
EB_HI3CMT	No	Yes	No	High high alarm state comment.
EB_HI3PRIO	No	Yes	Yes	Priority of high high high alarm message.
EB_HICONSTRLMT	No	No	Yes	High Constraint Limit.
EB_HIDEVCMT	No	Yes	No	High deviation alarm state comment.
EB_HIDEVPRIO	No	Yes	Yes	Priority of high deviation alarm message.
EB_HIRATECMT	No	Yes	No	High rate of change alarm state comment.
EB_HIRATEPRIO	No	Yes	Yes	Priority of high change rate alarm message.
EB_HIREF	No	No	Yes	High reference limit on display.
EB_LO2CMT	No	Yes	No	Low low alarm state comment.
EB_LO2PRIO	No	Yes	Yes	Priority of low low priority message.
EB_LO3CMT	No	Yes	No	Low low low alarm state comment.
EB_LO3PRIO	No	Yes	Yes	Priority of low low low priority message.
EB_LOCONSTRLMT	No	No	Yes	Low Constraint Limit.
EB_LODEVCMT	No	Yes	No	Low deviation alarm state comment.
EB_LODEVPRIO	No	Yes	Yes	Priority of low deviation priority message.
EB_LORATECMT	No	Yes	No	Low rate of change alarm state comment.
EB_LORATEPRIO	No	Yes	Yes	Priority of low change rate priority message.
EB_LOREF	No	No	Yes	Low reference limit on display.
EB_MULTLVL	No	Yes	Yes	TRUE if multilevel alarming is enabled.
EB_NOREPORT	Yes	Yes	Yes	TRUE if blockware has scanning disabled.
EB_NXTHILMT	No	No	Yes	Next higher alarm limit.
EB_NXTLOLMT	No	No	Yes	Next lower alarm limit.
EB_OUTRANGE	No	Yes	Yes	TRUE if value is suspect or out of range.
EB_PERINSEL	No	Yes	Yes	TRUE if operator is permitted to select input value.
EB_QUALOVR	No	Yes	Yes	TRUE if blockware is overriding the quality status.
EB_RATEALM	No	Yes	Yes	NO_RATE, LOW_RATE, or HIGH_RATE.
EB_REALM	No	Yes	Yes	TRUE if periodic re-alarming is enabled.
EB_REDTAG	No	Yes	Yes	TRUE if red-tagged (disables control actions).
EB_SRC	No	Yes	Yes	Composite mode (C_MODE and EB_ALTINSEL).
EB_USERVAL	Yes	No	Yes	Manually entered (user-inserted) value for tag.

Table A-6. INFI 90 DAANG (continued)

Atom Name	Write	Text	Numeric	Description
EB_VARALM	No	Yes	Yes	TRUE if variable alarming is enabled.

NOTE:
1. Writing any value to this atom results in a force XR request being sent to blockware.

Table A-7. INFI 90 DADIG

Atom Name	Write	Text	Numeric	Description
EB_ALMSUP	Yes	Yes	Yes	TRUE if blockware is currently suppressing alarms.
EB_ALTINSEL	Yes	Yes	Yes	TRUE if alternate input value is being used.
EB_AUTOMAN	Yes	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_FORCEXR	Yes[1]	No	Yes	Force block to send XR (always returns 0).
EB_LATCHED	No	Yes	Yes	TRUE if output state has been latched.
EB_NOREPORT	No	Yes	Yes	TRUE if blockware has scanning disabled.
EB_OUTRANGE	No	Yes	Yes	TRUE if value is suspect or out of range.
EB_PERINSEL	No	Yes	Yes	True if operator is permitted to select input value.
EB_QUALOVR	No	Yes	Yes	TRUE if blockware is overriding the quality status.
EB_REALM	No	Yes	Yes	TRUE if periodic re-alarming is enabled.
EB_REDTAG	No	Yes	Yes	TRUE if red-tagged (disables control actions).
EB_SRC	No	Yes	Yes	Composite mode (C_MODE and EB_ALTINSEL).
EB_USERTYPE	No	Yes	Yes	User Type code range (0 to 255).

NOTE:
1. Writing any value to this atom results in a force XR request being sent to blockware.

Table A-8. INFI 90 Device Driver

Atom Name	Write	Text	Numeric	Description
EB_DDMODE	Yes	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_FBSTS	No	Yes	Yes	Current feedback status (GOOD, BAD).
EB_REMOTE	No	Yes	Yes	TRUE if tag is in Remote mode.
EB_STSOVR	No	Yes	Yes	TRUE if the tag's status is overridden.

Table A-9. INFI 90 Device Driver Base

Atom Name	Write	Text	Numeric	Description
EB_FB2	No	Yes	Yes	Current state of feedback two.
EB_FB2LS0	No	Yes	No	Zero state descriptor for feedback two.
EB_FB2LS1	No	Yes	No	One state descriptor for feedback two.

Table A-10. INFI 90 Digital

Atom Name	Write	Text	Numeric	Description
EB_ALMDEF	No	Yes	Yes	Zero, One, or Neither.

Table A-11. INFI 90 Digital Base

Atom Name	Write	Text	Numeric	Description
C_DPV	Yes[1]	No	Yes	Current logic state of the tag.
C_SPV	No	Yes	No	Current logic state description of C_DPV state.
EB_ALMCMT	No	Yes	No	Alarm comment.
EB_ALMPRIO	No	Yes	Yes	Alarm Priority number or description (1 to 16).
EB_LS0	No	Yes	No	Logic state descriptor for state zero.
EB_LS1	No	Yes	No	Logic state descriptor for state one.
EB_SAVSTCHG	No	Yes	Yes	TRUE if state changes are to be logged.

NOTE:
1. Only if EB_SCANST is False or derived class is controllable.

Table A-12. INFI 90 Digital Export

Atom Name	Write	Text	Numeric	Description
C_DPV	Yes	Yes	Yes	Current logic state of the tag.
C_SPV	No	Yes	No	Current Symbolic state of the tag.
EB_ALMCMT	No	Yes	No	Alarm comment.
EB_ALMDEF	No	Yes	No	ZERO, ONE, or Neither.
EB_ALMPRIO	No	Yes	Yes	Alarm Priority Number or description (1 to 16).
EB_LS0	No	Yes	No	Logic state descriptor index for state zero.
EB_LS1	No	Yes	No	Logic state descriptor index for state one.
EB_SAVSTCHG	No	Yes	Yes	TRUE if state changes are to be logged.

Table A-13. INFI 90 Export Base

Atom Name	Write	Text	Numeric	Description
EB_ICIINDEX	No	No	Yes	ICI Index to be used for export point.

Table A-14. INFI 90 Import Base

Atom Name	Write	Text	Numeric	Description
EB_SPECSRCVD	No	Yes	Yes	TRUE if the blockware specifications have been received.
EB_SPECTIMESTAMP	No	Yes	Yes[1]	Time of last spec exception from block-ware.

NOTE:
1. Returns a time structure containing a tinet and nsecs.

Table A-15. INFI 90 MODULE STATUS

Atom Name	Write	Text	Numeric	Description
EB_BACSTS	No	Yes	Yes	TRUE if backup bad or memory is filled.

Table A-15. INFI 90 MODULE STATUS (continued)

Atom Name	Write	Text	Numeric	Description
EB_ICI_TYPE	No	Yes	Yes	TRUE if module is an ICI device.
EB_LOCIOSTS	No	Yes	Yes	TRUE if local I/O status is bad.
EB_MODERR	No	Yes	Yes	TRUE if module errors exist.
EB_MODMODE	No	Yes	Yes	Module mode.
EB_MODREV	No	Yes	Yes	Module Revision.
EB_MODTYP	No	Yes	No	Description of module type.
EB_MODVER	No	Yes	Yes	Module version.
EB_NCOMMERR	No	Yes	Yes	Summary of Communication errors, TRUE if exist.
EB_NMODERR	No	Yes	Yes	Summary of Module errors, TRUE if exist.
EB_NOFFLINE	No	Yes	Yes	TRUE if module is off-line.
EB_NSYSERR	No	Yes	Yes	Summary of System errors, TRUE if exist.
EB_NTYP	No	Yes	Yes	INFI 90 Node description.
EB_REMIOSTS	No	Yes	Yes	TRUE if remote I/O status is bad.
EB_SECONDARY	No	Yes	Yes	True if Module is Secondary of a redundant pair.
EB_XMITACK	No	Yes	Yes	TRUE if global acknowledgments should be sent.
EB_XMITSIL	No	Yes	Yes	TRUE if global alarm silences should be sent.

Table A-16. INFI 90 MSDD

Atom Name	Write	Text	Numeric	Description
C_IPV	No	Yes	Yes	Last Good State: Zero.
C_ISP	Yes[1]	Yes	Yes	Requested state: Zero.
EB_AUTOMAN	Yes	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_CNTRLOVR	No	Yes	Yes	TRUE if tag's control is overridden.
EB_FB3	No	Yes	Yes	Current state of feedback three.
EB_FB3LS0	No	Yes	No	Zero state descriptor for feedback three.
EB_FB3LS1	No	Yes	No	One state descriptor for feedback three.
EB_FB4	No	Yes	Yes	Current state of feedback four.
EB_FB4LS0	No	Yes	No	Zero state descriptor for feedback four.
EB_FB4SL1	No	Yes	No	One state descriptor for feedback four.
EB_LS2	No	Yes	No	Logic state descriptor for state two.
EB_LS3	No	Yes	No	Logic state descriptor for state three.
EB_STSOVR	No	Yes	Yes	TRUE if the tag's status is overridden.

NOTE:
Only if it is in manual mode, and then it can only be set to 1,2, or 3.

Table A-17. INFI 90 RCM

Atom Name	Write	Text	Numeric	Description
EB_OUTOVR	No	Yes	Yes	TRUE if the tag's state is overridden.

Table A-17. INFI 90 RCM (continued)

Atom Name	Write	Text	Numeric	Description
EB_RESCMD	No	Yes	Yes	Reset Command Received.
EB_RESIN	No	Yes	Yes	Logic Reset input Received.
EB_SETCMD	No	Yes	Yes	Set Command Received.
EB_SETIN	No	Yes	Yes	Logic Set Input Received.
EB_SETPER	No	Yes	Yes	TRUE if the tag's state can be set.

Table A-18. INFI 90 RCM Base

Atom Name	Write	Text	Numeric	Description
EB_FB1	No	Yes	Yes	Current state of feedback one.
EB_FB1LS0	No	Yes	No	Zero state descriptor for feedback one.
EB_FB1LS1	No	Yes	No	One state descriptor for feedback one.
EB_REDTAG	No	Yes	Yes	TRUE if red-tagged (disables control actions).
EB_USERTYPE	No	Yes	Yes	User Type Code range (0 to 255).

Table A-19. INFI 90 RMCB

Atom Name	Write	Text	Numeric	Description
EB_BADSTART	No	Yes	Yes	TRUE if startup failed.
EB_ERRCODE	No	Yes	Yes	Status of RMCB startup (0 to 9).
EB_FAULT	Yes[1]	Yes	Yes	TRUE if interlock/feedback lost while running.
EB_PM1LS0	No	Yes	No	Zero state descriptor for permissive one.
EB_PM1LS1	No	Yes	No	One state descriptor for permissive one.
EB_PM2LS0	No	Yes	No	Zero state descriptor for permissive two.
EB_PM2LS1	No	Yes	No	One state descriptor for permissive two.
EB_STARTPM1	No	Yes	Yes	Current state of start permissive one.
EB_STARTPM2	No	Yes	Yes	Current state of start permissive two.
EB_TXTSET	No	Yes	Yes	RMCB text message set for error codes.

NOTE:
1. Any value written will acknowledge the fault.

Table A-20. INFI 90 RMSC

Atom Name	Write	Text	Numeric	Description
EB_OPHILM	No	No	Yes	High Limit Value.
EB_OPLOLM	No	No	Yes	Low Limit Value.
EB_TRACKING	No	Yes	Yes	TRUE if tag's value is tracking an input value.

Table A-21. INFI 90 Server

Atom Name	Write	Text	Numeric	Description
EB_ALMCMT	No	Yes	No	Alarm Comment.
EB_ALMPRIO	No	Yes	Yes	Alarm Priority number or description (1 to 16).
EB_ANG_XR_CNT	Yes[1]	No	Yes	Number of Analog tag XR's received.

Table A-21. INFI 90 Server (continued)

Atom Name	Write	Text	Numeric	Description
EB_ASCII_XR_CNT	Yes[1]	No	Yes	Number of ASCII tag XR's received.
EB_DAANG_XR_CNT	Yes[1]	No	Yes	Number of DAANG tag XR's received.
EB_DADIG_XR_CNT	Yes[1]	No	Yes	Number of DADIG tag XR's received.
EB_DD_XR_CNT	Yes[1]	No	Yes	Number of DD tag XR's received.
EB_DIG_XR_CNT	Yes[1]	No	Yes	Number of Digital tag XR's received.
EB_ICI_MODE	No	Yes	Yes	Current mode of server's ICI.
EB_ICI_REV	No	Yes	Yes	Revision level of server's ICI.
EB_ICI_TYPE	No	Yes	Yes	Type of server's ICI.
EB_INFINET	No	Yes	Yes	True if server is operating on INFI NET.
EB_LOOP	No	Yes	Yes	INFI Loop address of server.
EB_LOOP_TIME	No	Yes	No	Current INFI-NET Loop Time.
EB_MOD_TSTAMP	No	Yes	Yes	True if Module Time stamping is enabled.
EB_MSDD_XR_CNT	Yes[1]	No	Yes	Number of MSDD tag XR's received.
EB_MSTAT_XR_CNT	Yes[1]	No	Yes	Number of Module Status tag XR's received.
EB_NODE	No	No	Yes	INFI Node address of server.
EB_RCM_XR_CNT	Yes[1]	No	Yes	Number of RCM tag XR's received.
EB_RED_TAG_LVL	No	No	Yes	Security Level required to modify red tag status.
EB_RESTARTS	Yes[1]	No	Yes	Number of ICI restarts.
EB_RMCB_XR_CNT	Yes[1]	No	Yes	Number of RMCB tag XR's received.
EB_RMSC_XR_CNT	Yes[1]	No	Yes	Number of RMSC tag XR's received.
EB_RT_CTRL_LVL	No	No	Yes	Security Level required to override red tag lockout.
EB_RTNCMT	No	Yes	No	Return to Normal Alarm Comment.
EB_RTNPRIO	No	Yes	Yes	Return to Normal Priority number or description.
EB_SPECWRITE	No	Yes	Yes	True if I90 Specs will be written back to configure.
EB_STAT_XR_CNT	Yes[1]	No	Yes	Number of Station tag XR's received.
EB_TEXT_XR_CNT	Yes[1]	No	Yes	Number of Text tag XR's received.
EB_TS_ACC	No	No	Yes	Time Sync accuracy of this server.
EB_TS_ENBL	No	Yes	Yes	True if Time Synchronizing is enabled
EB_TS_MSTR	No	Yes	Yes	True if this server is current time master.
EB_TS_MSTR_ACC	No	No	Yes	Current time sync accuracy.
EB_TS_MSTR_LOOP	No	No	Yes	Loop address of current time master.
EB_TS_MSTR_NODE	No	No	Yes	Node address of current time master.
EB_TS_UPDATE	No	No	Yes	Time Sync master update period.
EB_TS_WAIT	No	No	Yes	Time Sync update period.
EB_TUNE_LVL	No	No	Yes	Security Level required to Tune.
EB_XR_RATE	No	No	Yes	The rate at which exception reports are being received.

NOTE:
1. Any value written will reset this atom to 0.

Table A-22. INFI 90 Station

Atom Name	Write	Text	Numeric	Description
C_AOUTPUT	Yes	No	Yes	Last reported control output for tag.
C_ASP	Yes	No	Yes	Last reported setpoint for tag.
C_RATIO	Yes	No	Yes	Last reported ratio index for tag.
EB_AUTOMAN	Yes	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_BYPASS	No	Yes	Yes	TRUE if analog Control Output is bypassed.
EB_CMPTRSTS	No	Yes	Yes	TRUE if Computer is still sending OK signal.
EB_DEVALM	No	Yes	Yes	Deviation Alarm Status (0-Normal, 1-Low Alarm.
EB_DEVLMT	No	No	Yes	Alarm if setpoint and process value differ by more.
EB_HIDEVCMT	No	Yes	No	High deviation alarm state comment.
EB_HIDEVPRIO	No	Yes	Yes	Priority of high deviation alarm message (1 to 16).
EB_LODEVCMT	No	Yes	No	Low deviation alarm state comment.
EB_LODEVPRIO	No	Yes	Yes	Priority of low deviation priority message (1 to 16).
EB_MODELOCK	No	Yes	Yes	TRUE if blockware will not accept mode changes.
EB_OUTSTS	No	Yes	Yes	TRUE if (hand held) Station has failed.
EB_OUTTRACKING	No	Yes	Yes	TRUE if station output is tracking an input value.
EB_REDTAG	No	Yes	Yes	TRUE if 'red-tagged' (disables control actions).
EB_SPHIGH	No	No	Yes	Tag's setpoint zero+span value in engineering unit.
EB_SPSPAN	No	No	Yes	Tag's setpoint span value in engineering units.
EB_SPTRACKING	No	Yes	Yes	TRUE if tag's value is tracking an input value.
EB_SPZERO	No	No	Yes	Tag's setpoint zero value in engineering units.
EB_STNLVL	Yes	Yes	Yes	TRUE if block under Computer control.
EB_STNMODE	Yes	Yes	Yes	TRUE if station is in either Cascade or Ratio mode.
EB_STNTYPE	No	Yes	Yes	BASIC_WITH_SETPOINT.
EB_TUNEBLK	No	No	Yes	Block number of feedback's tuning block (1 to 9998).

Table A-23. INFI 90 Text Tag

Atom Name	Write	Text	Numeric	Description
EB_BLINK	Yes[2]	Yes	Yes	TRUE if displayed text should be blinking.
EB_COLOR	Yes[2]	No	Yes	Color to display the text in (range 0 to 63).
EB_MSG	Yes[2]	Yes	Yes[1]	Current Text for tag.

NOTES:
1.Any value written will reset this atom to zero.
2. Only if EB_SCANST is false.

Table A-24. INFI 90 UDXR Base

Atom Name	Write	Text	Numeric	Description
EB_ALMCMT	No	Yes	No	Alarm Comment.
EB_ALMPRIO	No	Yes	Yes	Alarm Priority number or description (1 to 16).
EB_ALMSUP	Yes	Yes	Yes	TRUE if blockware is currently suppressing alarms.
EB_AUTOMAN	Yes[1]	Yes	Yes	TRUE if tag is in Automatic (self-controlled) mode.
EB_BLKLEN	No	No	Yes	Maximum length of data string from blockware.
EB_CNTRLPER	No	Yes	Yes	TRUE if operator control is enabled for this tag.
EB_LOCTRUNC	No	Yes	Yes	TRUE if local truncation occurred.
EB_MODELOCK	No	Yes	Yes	TRUE if blockware will not accept mode changes.
EB_NAK	No	Yes	Yes	TRUE if blockware has rejected data from console.
EB_OISLEN	No	No	Yes	Maximum length of accepted data strings.
EB_ORIGLEN	No	No	Yes	Length of data originally sent.
EB_PDTECHO	No	Yes	Yes	TRUE if blockware has echoed data from console.
EB_QUALOVR	No	Yes	Yes	TRUE if blockware is overriding the quality status.
EB_RCVLEN	No	No	Yes	Length of data as finally received.
EB_REALM	No	Yes	Yes	TRUE if periodic re-alarming is enabled.
EB_REDTAG	No	Yes	Yes	TRUE if red-tagged (disables control actions).
EB_SEQNUM	No	No	Yes	Every new UDXR report increments sequence no.
EB_SETPRIO	Yes[2]	No	Yes	Alarm priority of last control action.
EB_STRLOCK	No	Yes	Yes	TRUE if blockware will not accept operator data.
EB_TRUNC	No	Yes	Yes	TRUE if remote truncation occurred.

NOTES:
1. Only if EB_CNTRLENABLE is True.
2. Priority will be used for all subsequent text string

Table A-25. Server Base

Atom Name	Write	Text	Numeric	Description
EB_ALARM_CNT	No	No	Yes	Number of tags in alarm.
EB_BIND_CNT	Yes[1]	No	Yes	Number of bind requests received.
EB_CONTEXT_CNT	No	No	Yes	Number of Event contexts currently open.
EB_CTRL_LVL	No	No	Yes	Security Level required to perform Control.
EB_DBASE_LVL	No	No	Yes	Security Level required to change configuration.
EB_DOMAIN	No	Yes	No	Server's Domain.
EB_GET_CNT	Yes[1]	No	Yes	Number of get request received.
EB_HOST	No	Yes	No	Name of host server is running on.

Table A-25. Server Base (continued)

Atom Name	Write	Text	Numeric	Description
EB_INHB_LVL	No	No	Yes	Security Level required to inhibit Alarms.
EB_MAX_TAGS	No	No	Yes	Maximum number of tags supported by server.
EB_NUM_TAGS	No	No	Yes	Number of tags contained in the server.
EB_PUT_CNT	Yes[1]	No	Yes	Number of put requests received.
EB_READ_LVL	No	No	Yes	Security Level required to perform Reads.
EB_SCANST_LVL	No	No	Yes	Security Level required to change Scan State.
EB_SERV_STAT	No	Yes	Yes	Server Status.

NOTE:
1. Any value written will reset this atom to 0.

Table A-26. Tag Base

Atom Name	Write	Text	Numeric	Description
CBNA	No	Yes[2]	No	Batch Name.
C_DESC	No	Yes	No	Description of tag.
EB_ACK	Yes	Yes	Yes	TRUE if acknowledgment is required.
EB_ALARMTIME	No	Yes	Yes[3]	Time of last alarm change event.
EB_ALMGRP	No	Yes	Yes	Alarm Group.
EB_ALMSTATUS	No	Yes	Yes	Tag's current Alarm Status.
EB_AUTOINHB	No	Yes	Yes	TRUE if tag's alarms are inhibited by another tag.
EB_BADQUAL	No	Yes	Yes	TRUE if tag has EB_BADQUAL (suspect value).
EB_CURCMT	No	Yes	No	Alarm comment of current alarm state.
EB_CURPRIO	No	Yes	Yes	Current alarm priority number or description (1 to 16).
EB_CUSTTAGID	No	Yes	No	Customer's label for this tag.
EB_GRPDSPL	No	Yes	No	Name of display to show for tag's faceplate.
EB_GRPINHB	No	Yes	Yes	TRUE if tag's alarm group is inhibited.
EB_INHB	No	Yes	Yes	TRUE if any of EB_AUTOINHB or EB_GRPINHB or EB_MAN.
EB_INHBST	No	Yes	Yes	State of tag which inhibits this tag's al.arms.
EB_INHBTAG	No	Yes	No	Name of tag which inhibits this tags alarms.
EB_LATCHTIME	No	Yes	Yes[3]	Time of last change into the alarm state.
EB_MANINHB	Yes[1]	Yes	Yes	TRUE if tag's alarms are manually inhibited.
EB_MANINHBPER	No	Yes	Yes	TRUE if manual inhibit is permitted.
EB_PNTDSPL	No	Yes	No	Name of display to show for 'tuning' tag.
EB_PRALMST	No	No	Yes	TRUE if tag is in alarm.
EB_QUAL	No	Yes	No	Tag's current quality (collection of quality chars).
EB_SAVOPACT	No	Yes	Yes	TRUE if operator actions are to be logged.
EB_SCANSTS	Yes	Yes	Yes	TRUE if scanning is enabled for this tag.

Table A-26. Tag Base (continued)

Atom Name	Write	Text	Numeric	Description
EB_SUBST	No	Yes	Yes	TRUE if tag's state or value is being substituted.
EB_TAG	No	Yes	No	Name of tag.
EB_TAGTYPE	No	Yes	No	Tag Type description.
EB_TAGUUID	No	Yes	No	Unique Tag ID.
EB_XR_COUNT	Yes[4]	No	Yes	Total number of realtime data events.
EB_XRTIMESTAMP	No	Yes	Yes[3]	Time of last realtime data event.

NOTES:
1. Only writable if EB_MANINHBPER is True.
2. Only provide for compatibility with PWC. Returned blank string.
3. Returns a structure containing both a time of and msecs.
4. Any Value written will reset count to zero.

APPENDIX B - BAILEY SUBMODELS

INTRODUCTION

Bailey Submodels are supplied by Bailey and contain pre-configured dynamic attributes. These submodels are used:

- To display dynamic data from the tag database, using various forms such as valves and bargraphs.
- For the control of INFI 90 OPEN tags.
- For the assignment of display vectoring (i.e., the ability to call up a specific display directly from another display).

SUBMODELS FOR CONTROL AND DATA ENTRY

Some control and data entry submodels use the variables low_color, up_color, and face_color as shading colors to give the button a three dimensional appearance. If these variables are not configured, the default colors listed with the variable description will be used. As a general rule, up_color should be lighter than face_color, which in turn should be lighter than low_color.

FP_DYbargraph

bargraph

PURPOSE:

The submodel **FP_DYbargraph** (Figure B-1) represents the value of a dynamic process variable in the form of a rectangular bar.

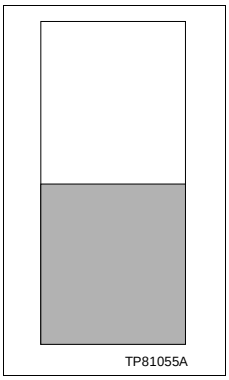


Figure B-1. Dynamic Bargraph (FP_DYbargraph)

SUBMODEL:

FP_DYbargraph

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN server tag database point (or a constant value) that is used to determine the length of the bar (i.e., FIC-447.C_APV).
edge_color	Number of the desired outline color of the bar as displayed in the Color Attribute Control Panel. Default: 7 (White)
edge_width	Number of the desired thickness of the outline of the bar (Enter Range 0 - 3). Default: 1
fill_color	Number of the desired fill color of the bar as displayed in the Color Attribute Control Panel. Default: 0 (Black)
fill_dir	Number representing the direction that the bar fills as the value of Tag.atom increases from lower_limit to upper_limit. up - 0 right - 1 down - 2 left - 3 Default: 0
lower_limit	Tag.atom of the INFI 90 OPEN server tag database point or the constant value that represents the lower limit of the bar (i.e., FIC-447.C_RANGLO). Default: None

FP_DYbargraph (continued)

bargraph

Variable	Description
upper_limit	Tag.atom of the INFI 90 OPEN server tag database point or the constant value that represents the upper limit of the bar (i.e., FIC-447.C_RANGHI). Default: None

FP_DYbargraph_ec

bargraph

PURPOSE:

Like **FP_DYbargraph**, the submodel **FP_DYbargraph_ec** (Figure B-1) represents the value of a dynamic process variable in the form of a rectangular bar, with one additional feature. As portions of the bargraph are erased due to changes in the process variable which controls the level of the fill color, the color defined as the erase color will replace the fill color. (For **FP_DYbargraph**, the erase color is the background color of the graphic on which it appears).

SUBMODEL:

FP_DYbargraph_ec

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point (or a constant value) that is used to determine the length of the bar (i.e., FIC-447.C_APV).
edge_color	Number of the desired outline color of the bar as displayed in the Color Attribute Control Panel. Default: 7 (White)
edge_width	Number of the desired thickness of the outline of the bar (Enter Range 0 - 3). Default: 1
erase_color	Number of the color to be used in portions of the bar where the fill_color is erased. Choose a number from the Color Attribute Control Panel. Default: 0 (Black)
fill_color	Number of the desired fill color of the bar as displayed in the Color Attribute Control Panel. Default: 0 (Black)
fill_dir	Number representing the direction that the bar fills as the value of Tag.atom increases from lower_limit to upper_limit. up - 0 right - 1 down - 2 left - 3 Default: 0
lower_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the lower limit of the bar (i.e., FIC-447.C_RANGLO). Default: None
upper_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the upper limit of the bar (i.e., FIC-447.C_RANGHI). Default: None

NOTE: The bargraph model is sized as a 10 x 10 unit rectangle. To change its size, change its reference point to one of its corners using the Change Reference menu item and then scale it to fit in the desired area.

FP_DYctrlpb

pushbutton

PURPOSE:

The submodel **FP_DYctrlpb** (Figure B-2) is a model of a push-button which, when pressed, displays (in a pop-up window) the appropriate group display model for the configured tag database point, allowing for control of that point from a graphic.

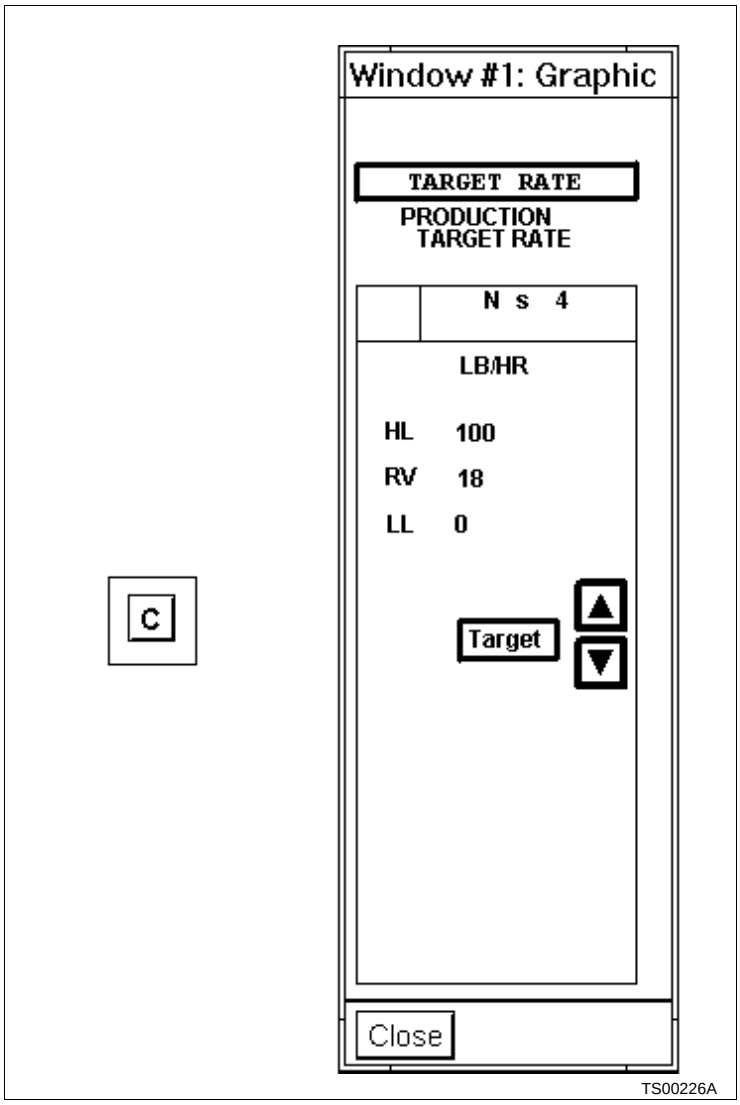


Figure B-2. Control Pushbutton (FP_DYctrlpb)

SUBMODEL:

FP_DYctrlpb

Variable	Description
TagName	Tagname of the tag database point to be displayed when the button is pressed.

FP_DYdial

miscellaneous display

PURPOSE:

The submodel **FP_DYdial** (Figure B-3) represents the value of an INFI 90 OPEN tag database point in the form of a dial. The white dial hand represents the process variable, while the red hand represents a setpoint.

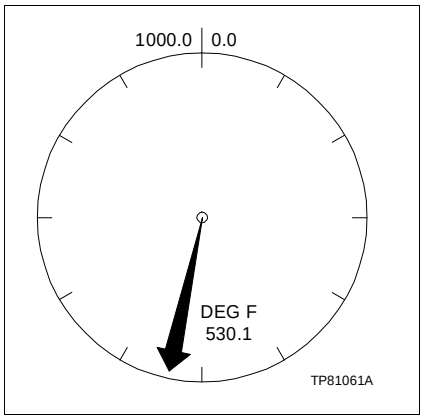


Figure B-3. Dial Submodel (FP_DYdial)

SUBMODEL:

FP_DYdial

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point (or a constant value) that is to be displayed (i.e., FIC-447.C_APV).
lower_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the lower limit of the dial (i.e., FIC-447.C_RANGLO). Default: None
setpoint	Tag.atom of the INFI 90 OPEN tag database point (or a constant value) that represents the setpoint of the dial (i.e., FIC-447.C_ASP). Default: None
units	ASCII string describing the Engineering Units (i.e., DEG C). Default: eun
upper_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the upper limit of the dial (i.e., FIC-447.C_RANGHI). Default: None

FP_DYentertextpb

pushbutton

PURPOSE:

The submodel **FP_DYentertextpb** (Figure B-4) is a pushbutton that, when pressed, displays in a pop up window a means of entering an ASCII string into a INFI 90 OPEN Tag.atom.

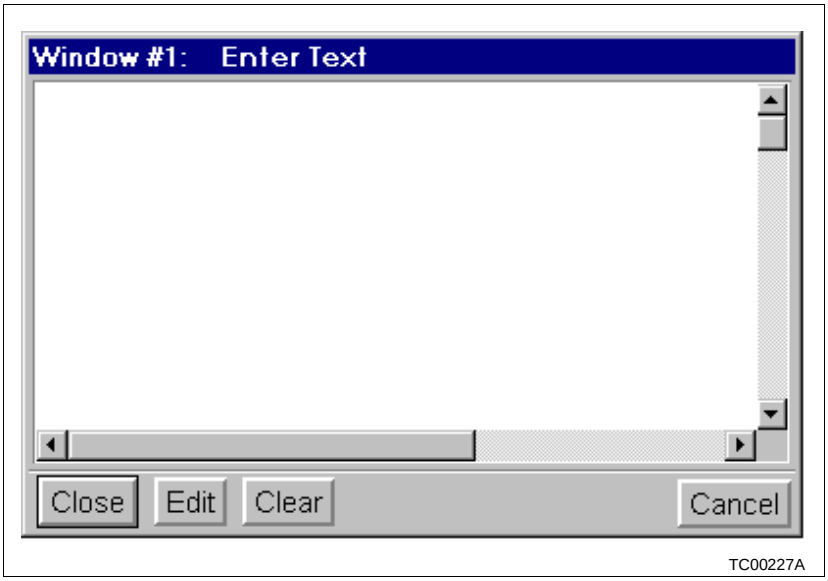


Figure B-4. Enter Text Pushbutton Submodel (FP_DYentertextpb)

SUBMODEL:

FP_DYentertextpb

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point to which the data will be written (i.e., REAC-TOR-01A.EB_TEXTSTR).
Title	Label to be displayed in the title bar of the pop up window. (i.e., Enter the text:). Default: None
button_label	Label to be displayed on the button (i.e., SP). Default: None
face_color	Number of the desired face color of the button as dis-played in the Color Attribute Control Panel. Default: 26 (used for FP_DYctrlpb)
low_color	Number of the desired lower edge color of the button as displayed in the Color Attribute Control Panel. Default: 29 (used for FP_DYctrlpb)
txt_color	Number of the desired text color of the button label as dis-played in the Color Attribute Control Panel. Default: 4 (used for FP_DYctrlpb)
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel. Default: 23 (used for FP_DYctrlpb)

FP_DYentervaluepb

pushbutton

PURPOSE: The submodel **FP_DYentervaluepb** (Figure B-5) is a pushbutton which, when pressed, displays in a pop up window a means of entering a floating point value into a INFI 90 OPEN server tag.atom.

Figure B-5. Enter Value Pushbutton (FP_DYentervaluepb)

SUBMODEL: **FP_DYentervaluepb**

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point to which the data will be written (i.e.,FIC-477.ASP.sp).
button_label	Label to be displayed on the button (i.e., SP). Default: None
face_color	Number of the desired face color of the button as displayed in the Color Attribute Control Panel (i.e., 26 used for FP_DYctrlpb). Default: 26
low_color	Number of the desired lower edge color of the button as displayed in the Color Attribute Control Panel (i.e., 29 used for FP_DYctrlpb). Default: 29
txt_color	Number of the desired text color of the button label as displayed in the Color Attribute Control Panel (i.e., 4 used for FP_DYctrlpb). Default: 4
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel (i.e., 23 used for FP_DYctrlpb). Default: 23

FP_DYgroupmod

hierarchical display

PURPOSE:

The submodel **FP_DYgroupmod** is used to display the model of an INFI 90 OPEN server tag database point as used on a group display. In the graphics editor, **FP_DYgroupmod** appears simply as an unfilled rectangle which covers the extent of the submodel. Once the graphic containing this submodel has been installed, a properly configured instance of **FP_DYgroupmod** appears as shown in Figure B-6.

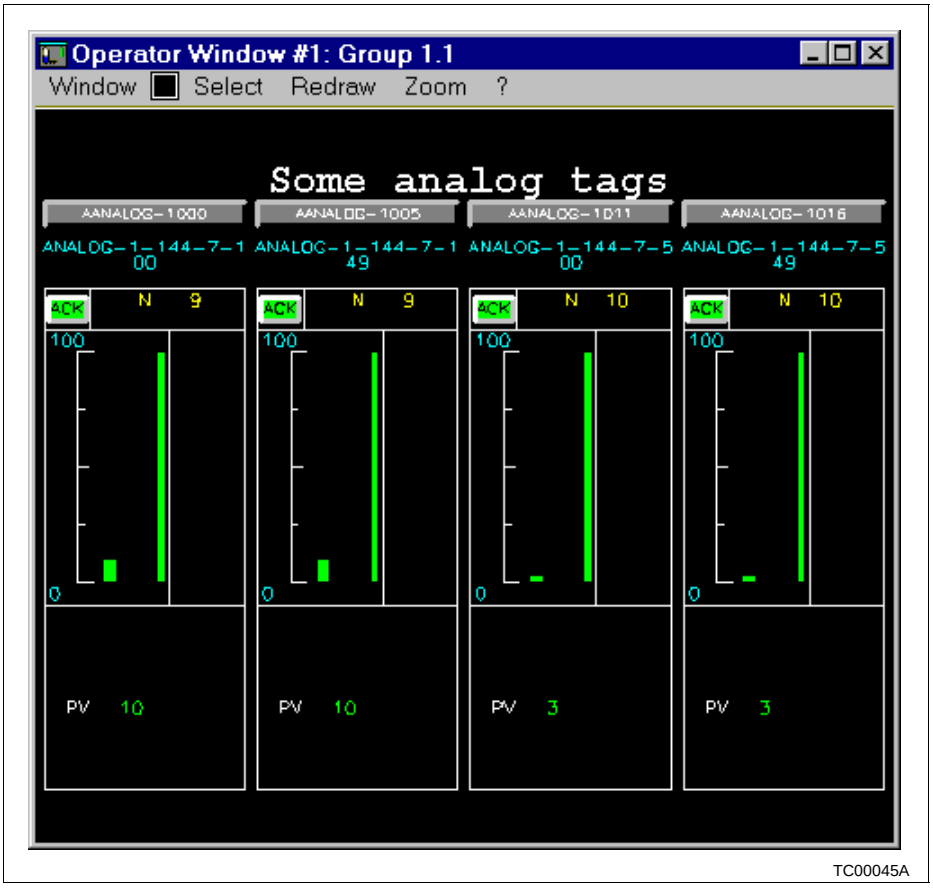


Figure B-6. Group Display Submodel (FP_DYgroupmod)

SUBMODEL:

FP_DYgroupmod

Variable	Description
TAG	Enter the Tagname of the INFI 90 OPEN tag database point that is to be displayed.

FP_DYHisttrend

trend

PURPOSE:

In the graphics editor, **FP_DYHisttrend** appears simply as an unfilled rectangle which covers the extent of the submodel. Once the graphic containing this submodel has been installed, a properly configured instance of **FP_DYHisttrend** appears as shown in Figure B-7. The submodel **FP_DYHisttrend** is an historical trend display.

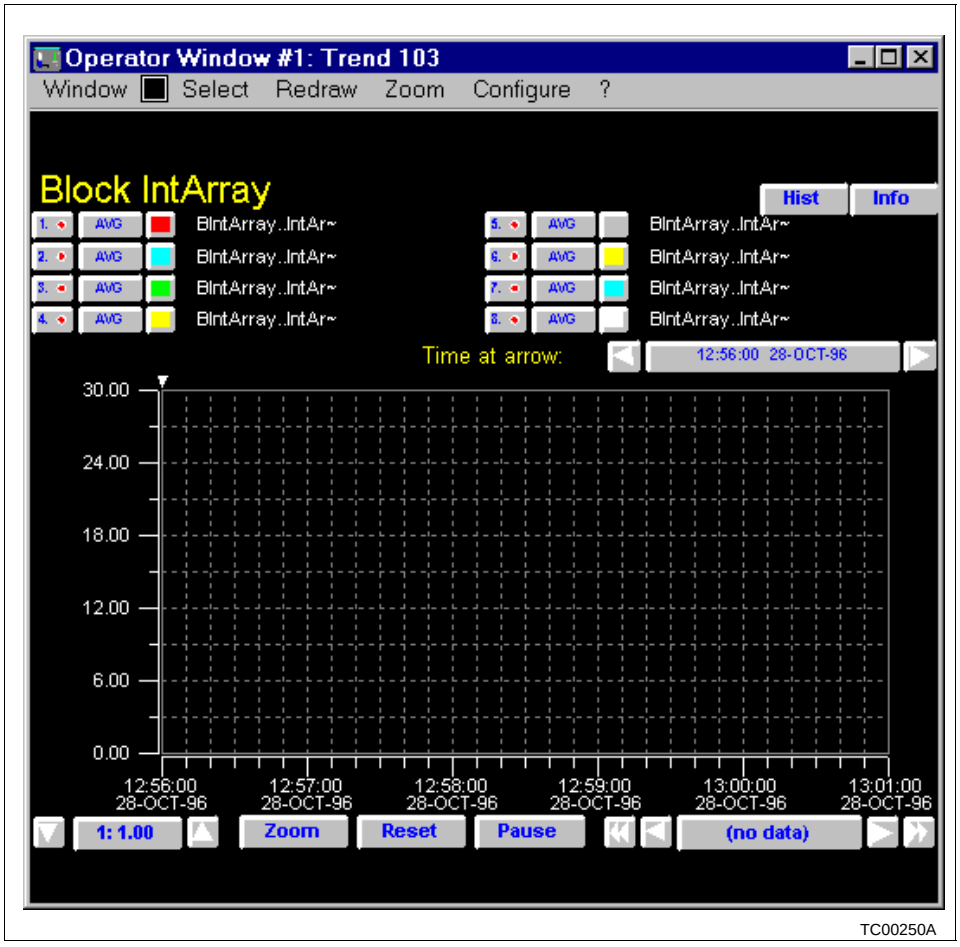


Figure B-7. Historical Trend Model (FP_DYHisttrend)

SUBMODEL:

FP_DYHisttrend

Variable	Description
trend_index	Enter the index number of the Historical Trend to be displayed.

FP_DYHorizbar

bargraph

PURPOSE:

The submodel **FP_DYHorizbar** (Figure B-8) represents the value of a dynamic process variable in the form of a horizontal bar-graph with a scale.

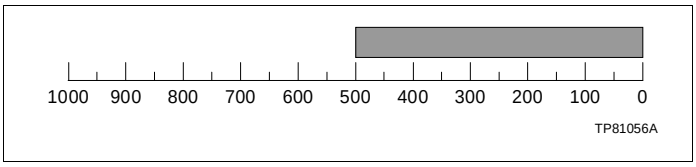


Figure B-8. Dynamic Horizontal Bar (FP_DYHorizbar)

SUBMODEL:

FP_DYHorizbar

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN server (or a constant value) that is used to determine the length of the bar (i.e., FIC-447.C_APV).
axis_color	Number of the desired color of the x-axis as displayed in the Color Attribute Control Panel. Default: 0 (Black)
bar_color	Number of the desired fill color of the bar as displayed in the Color Attribute Control Panel. Default: 7 (White)
direction	Number from the table below representing the direction that the bar fills as the value of Tag.atom increases from lower_limit to upper_limit. right - 1 left - 0 Default: 1
label_color	Number of the desired color of the x-axis labels as displayed in the Color Attribute Control Panel. Default: 0 (Black)
lower_limit	Tag.atom of the INFI 90 OPEN server tag database or the constant value that represents the lower limit of the bar (i.e., FIC-447.C_RANGLO). Default: None
upper_limit	Tag.atom of the INFI 90 OPEN server tag database or the constant value that represents the upper limit of the bar (i.e., FIC-447.C_RANGHI). Default: None
x_ticks	Number of tick marks to be displayed on the scale. Default: 0

FP_DYHoriztrend

trend

PURPOSE:

The submodel **FP_DYHoriztrend** (Figure B-9) displays a current trend for an INFI 90 OPEN tag database point. The trace is drawn from the right of the submodel. The data is scanned once every second and is not stored historically. The data is plotted on the y-axis and the number of samples is plotted on the x-axis.

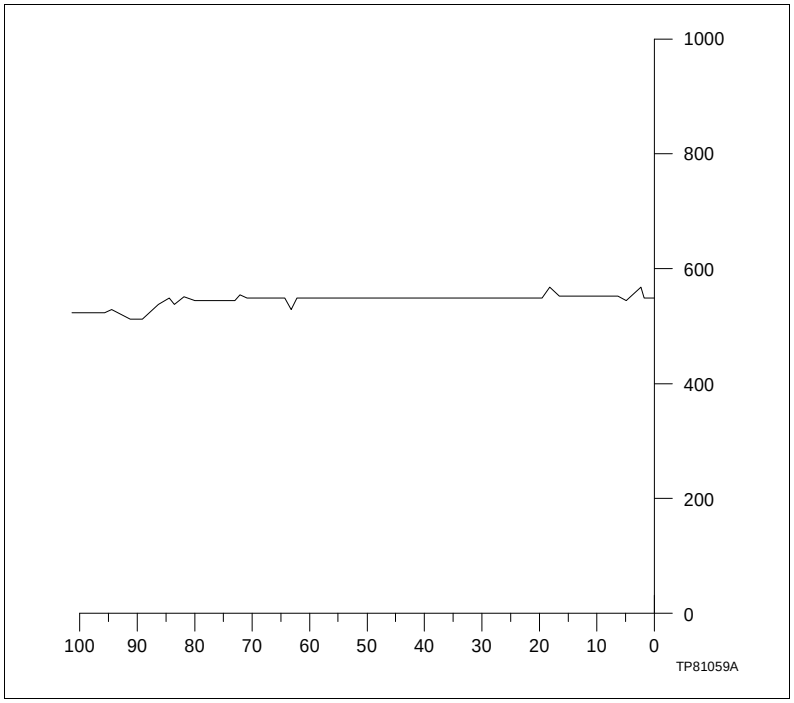


Figure B-9. Horizontal Trend Submodel (FP_DYHoriztrend)

SUBMODEL:

FP_DYHoriztrend

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point that is to be trended (i.e., FIC-447.C_APV).
num_points	Number of samples displayed on the trend.
axis_color	Number of the desired axis color as displayed in the Color Attribute Control Panel. Default: 0 (Black)
label_color	Number of the desired label color as displayed in the Color Attribute Control Panel. Default: 0 (Black)
trace_color	Number of the desired trace color as displayed in the Color Attribute Control Panel. Default: 7 (White)
x_ticks	Number of tickmarks on the X axis. Default: 0

FP_DYHoriztrend *(continued)*

trend

Variable	Description
y_ticks	Number of tickmarks on the Y axis. Default: 0
lower_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the lower limit of the trace (i.e., FIC-447.C_RANGLO). Default: None
upper_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the upper limit of the trace (i.e., FIC-477.C_RANGHI). Default: None

FP_DYhslider

slider

PURPOSE: The submodel **FP_DYhslider** (Figure B-10) provides a horizontal slider for control of INFI 90 OPEN tag database points. The following example changes the output of a controller module.

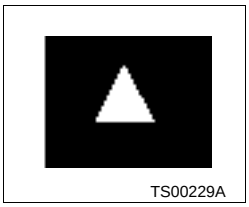


Figure B-10. Horizontal Slider Submodel (FP_DYhslider)

SUBMODEL: **FP_DYhslider**

Variable	Description
display_atom	Tag.atom of the INFI 90 OPEN tag database point represented by the position of the solid arrow (i.e., tcPID.OUT).
lower_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the lower limit of the slider (i.e., FIC-447.C_RANGLO). Default: None
target_atom	Tag.atom of the INFI 90 OPEN tag database point to be changed when the slider is released. The value is represented by the position of the hollow arrow (i.e., FIC-447.C_AOUTPUT). Default: None
upper_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the upper limit of the slider (i.e., FIC-447.C_RANGHI). Default: None

FP_DYmeter

miscellaneous display

PURPOSE:

The submodel **FP_DYmeter** (Figure B-11) displays an analog meter representation of the value of an INFI 90 OPEN tag database point.

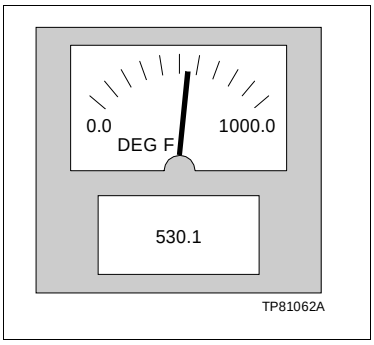


Figure B-11. Meter Submodel (FP_DYmeter)

SUBMODEL:

FP_DYmeter

Variable	Description
TagAtom	Tag.atom of the INFI 90 OPEN tag database point (or a constant value) that is to be displayed (i.e., FIC-447.C_APV).
lower_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the lower limit of the dial (i.e., FIC-447.C_RANGLO). Default: None
units	ASCII string describing the Engineering Units (i.e., DEG C). Default: eun
upper_limit	Tag.atom of the INFI 90 OPEN tag database point or the constant value that represents the upper limit of the dial (i.e., FIC-447.C_RANGHI).

FP_DYmpoppb

pushbutton

PURPOSE: The submodel **FP_DYmpoppb** (Figure B-5) is a pushbutton which, when pressed, displays a model, chosen by the user, in a pop-up window.

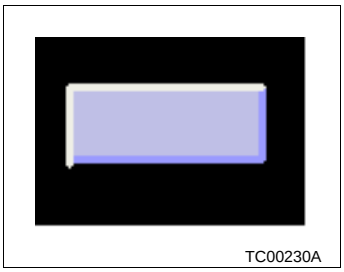


Figure B-12. Model Pop-up Pushbutton (FP_DYmpoppb)

SUBMODEL: **FP_DYmpoppb**

Variable	Description
button_label	Enter the label to be displayed on the button. Default: None
face_color	Enter the number of the desired face color of the button as displayed in the Color Attributes Control Panel. Default: 26 (Also used for FP_DYctrlpb).
low_color	Enter the number of the desired lower edge color of the button as displayed in the Color Attributes Control Panel. Default: 29 (Also used for FP_DYctrlpb).
model	Enter the name of the model that will appear in the pop up window when the button is pressed. (The model name should be that of an installed (on-line) graphic or submodel). User must define variable. Default: 0
scroll_flag	Enter a 1 if a narrow pop up window is desired; Enter a 0 if a wide pop up window is desired. Default: 0
tag	Enter the Tagname of any variable in the model that uses the variable TAG (refer to ***DECLARING VARIABLES IN SUBMODELS*** in Section 4 for a discussion on Declaring variables in Submodels).
txt_color	Enter the number of the desired text color of the button label as displayed in the Color Attributes Control Panel. Default: 4 (Also used for FP_DYctrlpb).
up_color	Number of the desired upper edge color of the button as displayed in the Color Attributes Control Panel. Default: 23 (Also used for FP_DYctrlpb).

FP_DYpointmod

hierarchical display

PURPOSE:

The submodel **FP_DYpointmod** (Figure B-13) displays the point display of an INFI 90 OPEN tag database point. In the graphics editor, **FP_DYpointmod** appears simply as an unfilled rectangle which covers the extent of the submodel. Once the graphic containing this submodel has been installed, a properly configured instance of **FP_DYpointmod** appears as shown in Figure B-13.

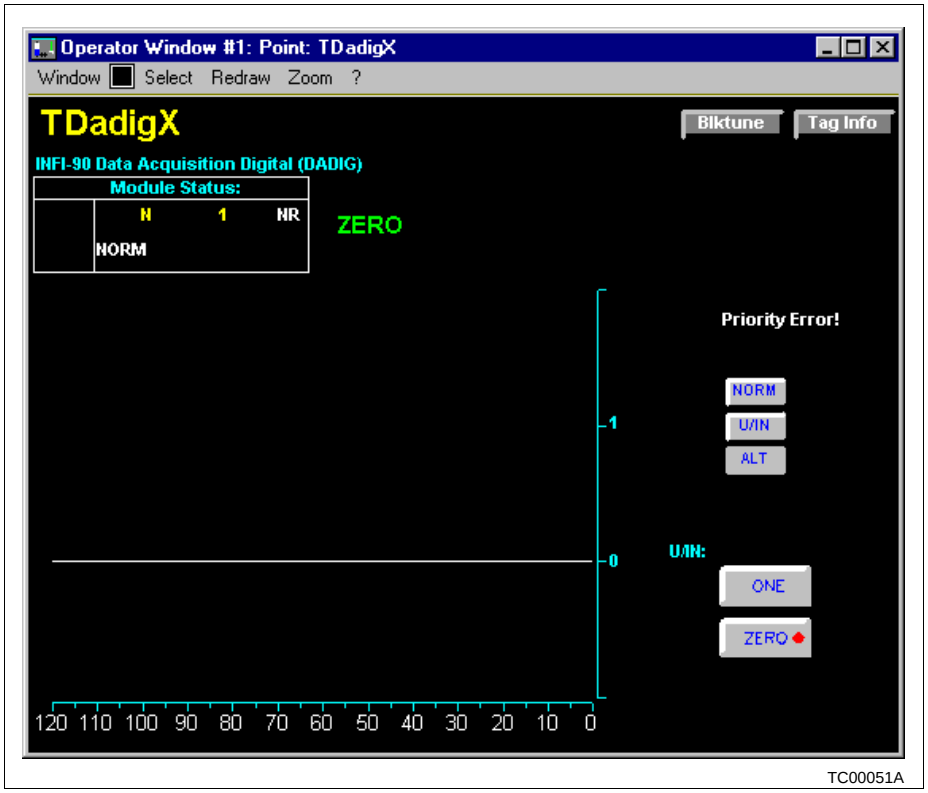


Figure B-13. Point Display Submodel (FP_DYpointmod)

SUBMODEL:

FP_DYpointmod

Variable	Description
TAG	Enter the Tagname of the tag database point that is to be displayed.

FP_DYpump_lf

rotating equipment display

PURPOSE:

The submodel **FP_DYpump_lf** (Figure B-14) is a pump with an outlet on the left. The submodel can be configured to change fill color dynamically, and also contains a wheel which can be animated so that it appears to spin in a counter-clockwise direction.

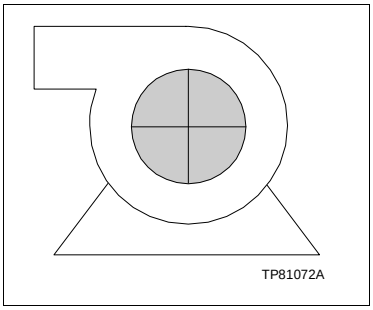


Figure B-14. Dynamic Pump - Left (FP_DYpump_lf)

SUBMODEL:

FP_DYpump_lf

Variable	Description
on_tagatom	Tagname.atom of a INFI 90 OPEN database point (i.e., HS-5441.C_ADPV); the value of this point will be used by the variable *on_value* (described below) to initiate the spin dynamic and change the fill color in the submodel.
on_value	Value compared with the value of the INFI 90 OPEN server database point configured in the variable on_tagatom. When the value of the tagname.atom matches the value configured here for on_value, the color of the pump will change to that defined in the variable pump_on_color and the wheel portion of the submodel will appear to spin.
period	Frequency at which the wheel spokes will appear to change position. The period is measured in tenths of a second; the number entered here defines how many tenths of a second are in the period (i.e., a 5 entered here means the period would be 0.5 seconds in duration). Default: 1 (One tenth of a second).
pump_off_color	Number of the fill color to be used for the pump submodel when the INFI 90 OPEN server database point configured in the variable *on_tagatom* is not equal to the value configured in the variable *on_value*. Default: 0 (Black)
pump_on_color	Index number of the fill color to be used for the pump submodel when the INFI 90 OPEN server point configured in the variable on_tagatom is equal to the value configured in on_value. Default: 0 (Black)

FP_DYpump_lf *(continued)*

rotating equipment display

Variable	Description
spoke_color	Index number of the color to be used for the spokes of the wheel portion of the submodel. Default: 0 (Black)
wheel_color	Index number of the color to be used for the interior of the wheel portion of the submodel. Default: 0 (Black)

FP_DYpump_rt

rotating equipment display

PURPOSE:

The submodel **FP_DYpump_rt** (Figure B-15) is a pump with an outlet on the right. The submodel can be configured to change fill color dynamically and contains a wheel which can be animated so that it appears to spin in a clockwise direction.

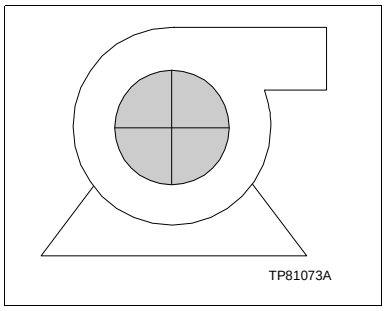


Figure B-15. Dynamic Pump - Right (FP_DYpump_rt)

SUBMODEL:

FP_DYpump_rt

Variable	Description
on_tagatom	Tagname.atom of a INFI 90 OPEN database point (i.e., HS-5441.C_DPV); the value of this point will be used by the variable on_value (described below) to initiate the spin dynamic and change the fill color in the submodel.
on_value	Value compared with the value of the INFI 90 OPEN database point configured in the variable on_tagatom. When the value of the tagname.atom matches the value configured here for on_value, the color of the pump will change to that defined in the variable pump_on_color and the wheel portion of the submodel will appear to spin.
period	Frequency at which the wheel spokes appear to change position. The period is measured in tenths of a second; the number entered here defines how many tenths of a second are in the period (i.e., a 5 entered here means the period would be 0.5 seconds in duration). Default: 1
pump_off_color	Number of the fill color to be used for the pump submodel when the tag database point configured in the variable *on_tagatom* is not equal to the value configured in the variable *on_value*. Default: 0 (Black)
pump_on_color	Index number of the fill color to be used for the pump submodel when the tag database point configured in the variable on_tagatom is equal to the value configured in *on_value*. Default: 0 (Black)
spoke_color	Index number of the color to be used for the spokes of the wheel portion of the submodel. Default: 0 (Black)

***FP_DYpump_rt** (continued)*

rotating equipment display

Variable	Description
wheel_color	Index number of the color to be used for the interior of the wheel portion of the submodel. Default: 0 (Black)

FP_DYput1pb

pushbutton

PURPOSE:

The submodel **FP_DYput1pb** (Figure B-16) is a pushbutton which, when pressed, writes a numeric value or ASCII character into one tag database point. The following example shows an instance of **FP_DYput1pb** configured to start the motor MCV-R01A01 (RCM tag) when the button is pressed (clicked on).

> **NOTE:** **FP_DYput1pb** is basically a simpler version of **FP_DYputpb**, and can be used in cases where one value is to be written to one tag database point. (**FP_DYputpb** allows up to four values to be written to as many as four tag database points).

Figure B-16. Put1 Pushbutton Submodel (FP_DYput1pb)

SUBMODEL:

FP_DYput1pb

Variable	Description
button_label	Label to be displayed on the button (i.e., Start). Default: None
put_tagatom	Tag.atom of the tag database point to which the data configured in put_value will be written (i.e., MCV-R01A01.C_DPV). Default: None
put_value	Value that will be written to the tag database point configured in put_tagatom (i.e., 1). Default: None

FP_DYputpb

pushbutton

PURPOSE:

The submodel **FP_DYputpb** (Figure B-17) is a pushbutton which, when pressed, writes numeric values and/or ASCII text into up to four tag database points. The following example shows an instance of **FP_DYputpb** configured to start the motor MCV-R01A01 (RCM tag) when the button is pressed (clicked on).

Figure B-17. Put Pushbutton Submodel (FP_DYputpb)

SUBMODEL:

FP_DYputpb

Variable	Description
button_label	Label to be displayed on the button (i.e., Start). Default: None
face_color	Number of the desired face color of the button as displayed in the Color Attribute Control Panel. Default: 26 (used for FP_DYctrlpb)
low_color	Number of the desired lower edge color of the button as displayed in the Color Attribute Control Panel. Default: 29 (used for FP_DYctrlpb)
put_tagatom	Tag.atom of the tag database point to which the data configured in put_value will be written (i.e., MCV-R01A01.C_DPV). Default: None
put_tagatom2	Tag.atom of the tag database point to which the data configured in put_value2 will be written. Default: None
put_tagatom3	Tag.atom of the tag database point to which the data configured in put_value3 will be written. Default: None
put_tagatom4	Tag.atom of the tag database point to which the data configured in put_value4 will be written. Default: None
put_value	Value that will be written to the tag database point configured in put_tagatom (i.e., 1). Default: None
put_value2	Value that will be written to the tag database point configured in put_tagatom2. Default: None

***FP_DYputpb** (continued)*

pushbutton

Variable	Description
put_value3	Value that will be written to the tag database point configured in put_tagatom3. Default: None
put_value4	Value that will be written to the tag database point configured in put_tagatom4. Default: None
txt_color	Number of the desired text color of the button label as displayed in the Color Attribute Control Panel. Default: 4 (used for FP_DYctrlpb)
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel. Default: 23 (used for FP_DYctrlpb)

FP_DYputpbstat

pushbutton

PURPOSE:

The submodel **FP_DYputpbstat** (Figure B-18) is a pushbutton which, when pressed, writes values into up to four tag database points and shows a red status light when the status of a tag database point is set to a configurable value. The following example shows an instance of **FP_DYputpbstat** configured to put the Station tag FIC477 into AUTO mode when the button is pressed (clicked on). Additionally, when the EB_AUTOMAN atom of the controller is set, the status light appears in red.

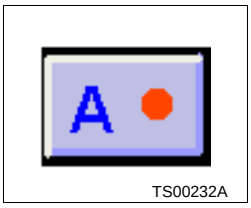


Figure B-18. Putpbstat Submodel (FP_DYputpbstat)

SUBMODEL:

FP_DYputpbstat

Variable	Description
button_label	Enter the label to be displayed on the button (i.e., A). Default: None
on_value	Enter the value of the tag database point configured in the variable statatom that will set the status light on. Default: 0
put_tagatom	Enter the Tag.atom of the tag database point to which the data configured in put_value will be written (i.e., FIC-447.EB_AUTOMAN). Default: None
put_tagatom2	Enter the Tag.atom of the tag database point to which the data configured in put_value2 will be written. Default: None
put_tagatom3	Enter the Tag.atom of the tag database point to which the data configured in put_value3 will be written. Default: None
put_tagatom4	Enter the Tag.atom of the tag database point to which the data configured in put_value4 will be written. Default: None
put_value	Enter the value that will be written to the tag database point configured in put_tagatom (i.e., 1). Default: None
put_value2	Enter the value that will be written to the tag database point configured in put_tagatom2. Default: None

FP_DYputpbstat (continued)

pushbutton

Variable	Description
put_value3	Enter the value that will be written to the tag database point configured in put_tagatom3. Default: None
put_value4	Enter the value that will be written to the tag database point configured in put_tagatom4. Default: None
statatom	Enter the Tag.atom of the tag database point that sets the status light on when its value is equal to on_value (i.e., FIC-447.EB_AUTOMAN). Default: None

FP_DYputpbstat2

pushbutton

PURPOSE:

The submodel **FP_DYputpbstat2** (Figure B-19) is a pushbutton that, when pressed, writes values into up to four tag database points. The button label changes color and an asterisk is displayed based on the value of a tag database point. The following example shows an instance of **FP_DYputpbstat2** configured to put the Station tag FIC477 into AUTO mode when the button is pressed (clicked on). Additionally, when the EB_AUTOMAN atom of the controller is set, an asterisk (*) is displayed to the right of the label, and the label Auto appears in red.

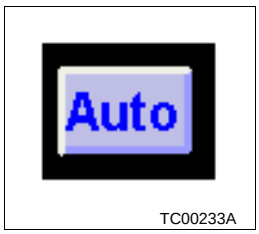


Figure B-19. FP_DYputpbstat2 Submodel

SUBMODEL:

FP_DYputpbstat2

Variable	Description
button_label	Label to be displayed on the button (i.e., Auto). Default: None
face_color	Number of the desired face color of the button as displayed in the Color Attribute Control Panel. Default: 26 (used for FP_DYctrlpb)
low_color	Number of the desired lower edge color of the button as displayed in the Color Attribute Control Panel. Default: 29 (used for FP_DYctrlpb)
off_color	Number of the desired color of the button label when the tag database point configured in statatom is not equal to the value configured in on_value. Use the numbers displayed in the Color Attribute Control Panel. Default: 0 (Black)
on_color	Number of the desired color of the button label when the tag database point configured in statatom is equal to the value configured in on_value. Use the numbers displayed in the Color Attribute Control Panel. Default: 0 (Black)
on_value	Value of the tag database point configured in the variable statatom that will cause an asterisk to be displayed on the button face and cause the button label to be displayed in the color configured in on_color. Default: 0

FP_DYputpbstat2 (continued)

pushbutton

Variable	Description
put_tagatom	Tag.atom of the tag database point to which the data configured in put_value will be written (i.e., FIC-447.EB_AUTOMAN). Default: None
put_tagatom2	Tag.atom of the tag database point to which the data configured in put_value2 will be written. Default: None
put_tagatom3	Tag.atom of the tag database point to which the data configured in put_value3 will be written. Default: None
put_tagatom4	Tag.atom of the tag database point to which the data configured in put_value4 will be written. Default: None
put_value	Value that will be written to the tag database point configured in put_tagatom (i.e.,1). Default: None
put_value3	Value that will be written to the tag database point configured in put_tagatom3. Default: None
put_value4	Value that will be written to the tag database point configured in put_tagatom4. Default: None
statatom	Tag.atom of the tag database point that is sets the status light on when its value is equal to on_value (i.e., FIC-447.EB_AUTOMAN). Default: None
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel. Default: 23 (used for FP_DYctrlpb)

FP_DYstatpb

pushbutton

PURPOSE:

The submodel **FP_DYstatpb** (Figure B-20) is a pushbutton which, when pressed, writes a value into a single tag database point and shows a red status light when the status of a tag database point is set to put_value. The following example shows an instance of **FP_DYstatpb** configured to put the Station tag FIC-477 into AUTO mode when the button is pressed (clicked on). Additionally, when the EB_AUTOMAN atom of the controller is set, the status light appears in red.

> **NOTE: FP_DYstatpb** is basically a simpler version of **FP_DYputpbstat**, and can be used in cases where a value is to be written to only one tag database point. (**FP_DYputpbstat** allows up to four values to be written to as many as four tag database points).

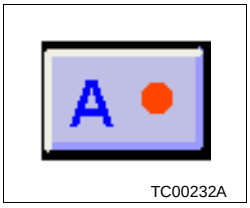


Figure B-20. Statpb Submodel (FP_DYstatpb)

SUBMODEL:

FP_DYstatpb

Variable	Description
button_label	Label to be displayed on the button (i.e., A). Default: None
put_tagatom	Tag.atom of the tag database point to which the data configured in put_value will be written (i.e., FIC-447.EB_AUTOMAN). This tag.atom also sets the status light on when its value is equal to put_value (i.e., FIC-447.EB_AUTOMAN). Default: None
put_value	Value that will be written to the tag database point configured in put_tagatom (i.e., 1); this is also the value used by put_tagatom to set the status light on. Default: None

FP_DYtagpb

pushbutton

PURPOSE:

The submodel **FP_DYtagpb** (Figure B-21) is a pushbutton containing the tagname of a tag database point which, when pressed, changes the display in the operator window to the point display for the tagname.

Figure B-21. Tagname Pushbutton (FP_DYtagpb)

SUBMODEL:

FP_DYtagpb

Variable	Description
TagName	Tag (TagName.tag) of the tag database point to be displayed when the button is pressed.
face_color	Number of the desired face color of the button as displayed in the Color Attribute Control Panel. Default: 26 (used for FP_DYctrlpb)
low_color	Number of the desired lower edge color of the button as displayed in the Color Attribute Control Panel. Default: 29 (used for FP_DYctrlpb)
txt_color	Number of the desired text color of the button label as displayed in the Color Attribute Control Panel. Default: 4 (used for FP_DYctrlpb)
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel. Default: 23 (used for FP_DYctrlpb)
tagnum	For use by summary display only. User input not required.

FP_DYtank

tank

PURPOSE:

The submodel **FP_DYtank** (Figure B-22) is a tank with a rounded bottom and top. The fill in both the rounded bottom and the rectangular vessel body is based on the value of a dynamic process variable.

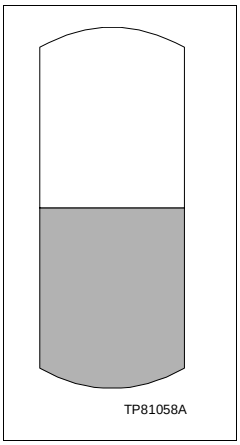


Figure B-22. Dynamic Tank Submodel (FP_DYtank)

SUBMODEL:

FP_DYtank

Variable	Description
TagAtom	Tag.atom of the tag database point (or a constant value) that is used to determine the height of the tank fill (i.e., FIC-447.C_APV).
fill_color	Number of the desired fill color of the tank as displayed in the Color Attribute Control Panel. Default: 0 (Black)
lower_limit	Tag.atom of the tag database point or the constant value that represents the lower limit of the rectangular area (i.e., FIC-447.C_RANGLO). Default: None
upper_limit	Tag.atom of the tag database point or the constant value that represents the upper limit of the rectangular area (i.e., FIC-447.C_RANGHI). Default: None
low level	Tag.atom of the tag database point or the constant value that represents the value above which the rounded bottom is filled (i.e., 0). Default: None

FP_DYtank_ec

tank

PURPOSE:

Like **FP_DYtank**, the submodel **FP_DYtank_ec** (Figure B-23) is a tank with a rounded bottom and top. The fill in both the rounded bottom and the rectangular vessel body is based on the value of a dynamic process variable. An additional feature, erase color, allows configuration of a color to be used in portions of the submodel where the fill_color is erased. (For **FP_DYtank**, the erase color is the background color of the graphic on which it appears).

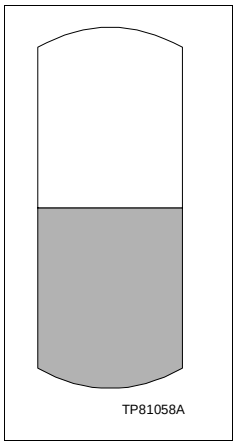


Figure B-23. Dynamic Tank Submodel with Configurable Erase Color (FP_DYtank_ec)

SUBMODEL:

FP_DYtank_ec

Variable	Description
TagAtom	Tag.atom of the tag database point (or a constant value) that is used to determine the height of the tank fill (i.e., FIC-447.C_APV).
fill_color	Number of the desired fill color of the tank as displayed in the Color Attribute Control Panel. Default: 0 (Black)
lower_limit	Tag.atom of the tag database point or the constant value that represents the lower limit of the rectangular area (i.e., FIC-447.C_RANGLO). Default: None
upper_limit	Tag.atom of the tag database point or the constant value that represents the upper limit of the rectangular area (i.e., FIC-447.C_RANGHI). Default: None
low level*	Tag.atom of the tag database point or the constant value that represents the value above which the rounded bottom is filled (i.e., 0). Default: None

FP_DYtank_ec *(continued)*

tank

Variable	Description
erase_color	Number of the color to be used in portions of the bar where the fill_color is erased. Choose a number from the Color Attribute Control Panel. Default: 0 (Black)

FP_DYtoggle

toggle

PURPOSE:

The submodel **FP_DYtoggle** (Figure B-24) is a pushbutton that, when pressed, toggles the value of a tag database point between one and zero. The color and text of the button label changes based on the value of a tag database point. The following example shows an instance of **FP_DYtoggle** configured to toggle the state of a RCM tag's C_DPV bit when the button is clicked on.

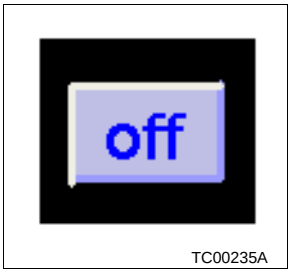


Figure B-24. FP_DYtoggle Submodel

SUBMODEL:

FP_DYtoggle

Variable	Description
TagAtom	Tag.atom of the tag database point that is to be toggled between one and zero.
face_color	Number of the desired face color of the button; numbers are displayed in the Color Attribute Control Panel. Default: 26 (used for FP_DYctrlpb)
low_color	Number of the desired lower edge color of the button; use the numbers displayed in the Color Attribute Control Panel. Default: 29 (used for FP_DYctrlpb)
off_color	Enter the number of the desired color of the button label when the tag database point configured in TagAtom is not equal to the value configured in on_value. Use the numbers displayed in the Color Attribute Control Panel. Default: 0 (Black)
off_label	Text to appear on the button face when the tag database point configured in TagAtom is not equal to the value configured in on_value. Default: None.
on_color	Number of the desired color of the button label when the tag database point configured in TagAtom is equal to the value configured in on_value. Use the numbers displayed in the Color Attribute Control Panel. Default: 0 (Black)
on_label	Text to appear on the button face when the tag database point configured in TagAtom is equal to the value configured in on_value. Default: None.

FP_DYtoggle *(continued)*

toggle

Variable	Description
on_value	Value of the tag database point configured in the variable TagAtom that will change the button label to the text defined in the variable on_label. Default: 0
up_color	Number of the desired upper edge color of the button label as displayed in the Color Attribute Control Panel. Default: 23 (used for FP_DYctrlpb)

P_DYTrendpb

pushbutton

PURPOSE:

The submodel **FP_DYtrendpb** (Figure B-25) is a pushbutton which, when clicked on, displays (in a pop-up window) the Historical Trend with which it has been configured.

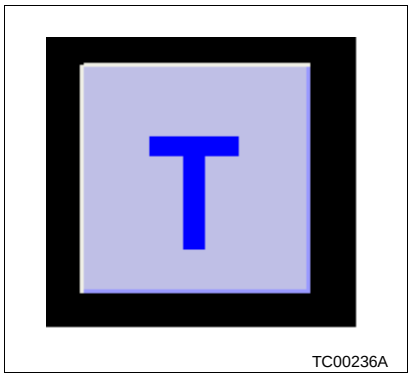


Figure B-25. Historical Trend Pushbutton (FP_DYtrendpb)

SUBMODEL:

FP_DYtrendpb

Variable	Description
trend_index	Index number of the Historical Trend to be displayed when the button is clicked.

FP_DYvalue

dynamic value

PURPOSE:

The submodel **FP_DYvalue** (Figure B-26) is used for the dynamic display of data from a tag database point. It is used for both numeric and ASCII data types.

Figure B-26. Dynamic Value (FP_DYvalue)

SUBMODEL:

FP_DYvalue

Variable	Description
TagAtom	Tag.atom of the tag database point (or a constant value) to be displayed (i.e., FIC-447.C_APV).
Type	Native - Displays the data in the format configured in the tag module (based on the FMT atom for floating point values). This format is always used to display text data. Value - Allows for user configurable format (refer to Decimal places). Default: Native
Decimal Places	Number of digits to the right of the decimal place. Used only when value is selected as the type; otherwise this item is not displayed. Default: 2
Color	Number of the desired color as displayed in the color attribute control panel. Default: 7 (White)
Font	Number of the desired font as displayed in the text attribute control panel. Default: 7 (Helvetica Bold)
Size	Size of the displayed data; 4.0 is the largest size and 1.0 is the smallest. Default: 2.0

FP_DYvalue (continued)

dynamic value

Variable	Description
Align	Left - The first (leftmost) digit of the dynamic value will be aligned with the starting location (left border) of the box which defines this submodel for positioning on the graphics editor display. Center - The dynamic value will be centered on the starting location (left border) of the box which defines this submodel for positioning on the graphics editor display. Right - The last (rightmost) digit of the dynamic value will be aligned with the starting location (left border) of the box which defines this submodel for positioning on the graphics editor Display. Default: Left

FP_DYvectorpb

pushbutton

PURPOSE:

The submodel **FP_DYvectorpb** (Figure B-27) is a pushbutton which, when clicked, causes the display in an operator window to change to a new display, specified by the user via configuration. By default, this change occurs in the window in which the vector pushbutton is located; to cause the new display to appear in a different operator window, use the drag and drop mode of operation; i.e., instead of clicking on the vector pushbutton, position the pointer on it and press the left mouse button. Continue to hold down the mouse button and drag the pointer until it is completely inside the frame of the operator window in which the new display is to be dropped. Release the mouse button, and the new display will be dropped into the window.

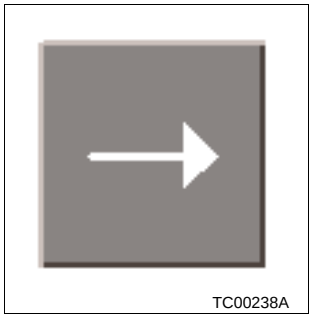


Figure B-27. Vector Pushbutton Submodel (FP_DYvectorpb)

SUBMODEL:

FP_DYvectorpb

Variable	Description
Display type[1]	Select the type of display to vector to.
Index	Select the index of the desired display type or the Tagname from the pop-up window that appears following the selection of the display type.

NOTE:
1. User must define variable

FP_DYVertbar

bargraph

PURPOSE: The submodel **FP_DYVertbar** (Figure B-28) represents the value of a dynamic process variable in the form of a vertical bargraph with a scale.

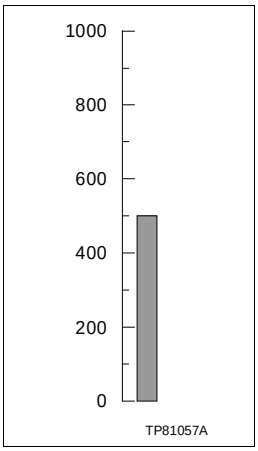


Figure B-28. Dynamic Vertical Bar (FP_DYVertbar)

SUBMODEL: **FP_DYVertbar**

Variable	Description
TagAtom	Tag.atom of the tag database point (or a constant value) that is used to determine the length of the bar (i.e., FIC-447.C_APV).
bar_color	Enter the number of the desired fill color of the bar as displayed in the Color Attribute Control Panel. Default: 7 (White)
axis_color	Enter the number of the desired color of the y-axis as displayed in the Color Attribute Control Panel. Default: 0 (Black)
direction	Number from the table below representing the direction that the bar fills as the value of Tag.atom increases from lower_limit to upper_limit. up - 1 down - 0 Default: 1
label_color	Number of the desired color of the y-axis labels as displayed in the Color Attribute Control Panel. Default: 0 (Black)
lower_limit	Tag.atom of the tag database point or the constant value that represents the lower limit of the bar (i.e., 0). Default: None
upper_limit	Tag.atom of the tag database point or the constant value that represents the upper limit of the bar (i.e., 100). Default: None
y_ticks	Number of tick marks to be displayed on the y axis. Default: 0

FP_DYverttrend

trend

PURPOSE:

The submodel **FP_DYVerttrend** (Figure B-29) displays a current trend for a tag database point. The trace is drawn from the top of the submodel. The data is scanned once every second and is not stored historically. The data is plotted on the x-axis and the number of samples is plotted on the y-axis.

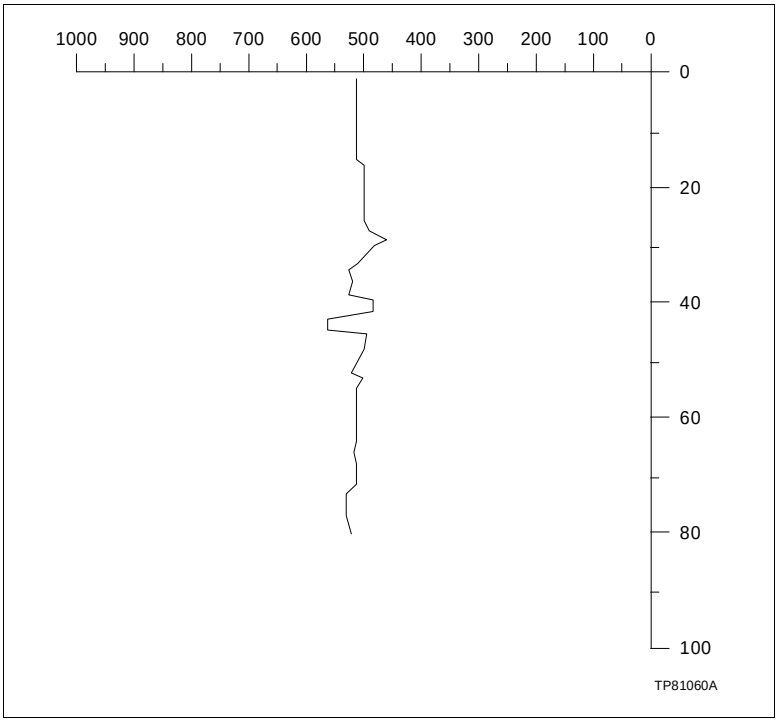


Figure B-29. Vertical Trend Model (FP_DYVerttrend)

SUBMODEL:

FP_DYVerttrend

Variable	Description
TagAtom	Tag.atom of the tag database point that is to be trended (i.e., FIC-447.C_APV).
num_points	Number of samples displayed on the trend.
axis_color	Number of the desired axis color as displayed in the Color Attribute Control Panel. Default: 0 (Black)
label_color	Number of the desired label color as displayed in the Color Attribute Control Panel. Default: 0 (Black)
trace_color	Number of the desired trace color as displayed in the Color Attribute Control Panel. Default: 7 (White)
x_ticks	Number of tickmarks on the X axis. Default: 0

FP_DYverttrend (continued)

trend

Variable	Description
y_ticks	Number of tickmarks on the Y axis. Default: 0

FP_DYvslider

slider

PURPOSE:

The submodel **FP_DYvslider** (Figure B-30) provides a vertical slider for control of tag database points. The following example changes the setpoint of a controller module.

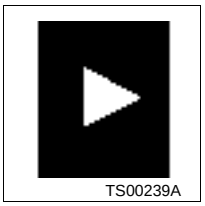


Figure B-30. Vertical Slider Submodel (FP_DYvslider)

SUBMODEL:

FP_DYvslider

Variable	Description
display_atom	Tag.atom of the tag database point represented by the position of the solid arrow (i.e., FIC-477.C_ASP).
lower_limit	Tag.atom of the tag database point or the constant value that represents the lower limit of the slider (i.e., FIC-447.C_RANGLO). Default: None
target_atom	Tag.atom of the tag database point to be written to when the slider is released. The value is represented by the position of the hollow arrow (i.e., FIC-477.C_ASP). Default: None
upper_limit	Tag.atom of the tag database point or the constant value that represents the upper limit of the slider. (i.e., FIC-447.C_RANGHI). Default: None

FP_DYwheel

rotating equipment display

PURPOSE: The submodel **FP_DYwheel** (Figure B-31) is an animated wheel that can be configured so that it appears to spin.

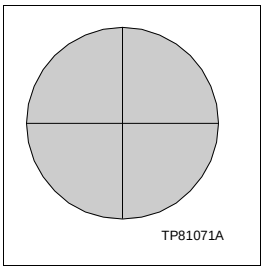


Figure B-31. Dynamic Wheel (FP_DYwheel)

SUBMODEL: **FP_DYwheel**

Variable	Description
dir	Tagname.atom of a point in the tag database, or enter the value representing the direction in which the wheel will spin. clockwise - 1 counterclockwise - 0
on_tagatom	Tagname.atom of a point in the tag database (i.e., HS-5441.C_DPV); the value of this point is used by the variable *on_value* (described below) to activate the spin dynamic attribute of the submodel.
on_value	Value; the value entered here will be compared with the value of the tag database point configured in the variable on_tagatom. When the value of the tag.atom matches the value configured here for *on_value*, the wheel will appear to spin.
period	Frequency at which the wheel spokes will appear to change position. The period is measured in tenths of a second; the number entered here defines how many tenths of a second are in the period (i.e., a 5 entered here means the period would be.5 seconds in duration). The higher the number used to define the period, the slower the object will appear to spin. Default: 1 (One tenth of a second).
spoke_color	Index number of the color to be used for the wheel spokes. Default: 0 (Black)
wheel_color	Number of the color to be used for of the wheel interior. Default: 0 (Black)

APPENDIX C - EXCEL FUNCTIONS

INTRODUCTION

This section describes the functions available with the PWC software which provide an interface to the Microsoft EXCEL spreadsheet package. These functions are for use with Microsoft EXCEL only. Experience and/or training with Microsoft EXCEL is assumed; it is beyond the scope of this manual to document use of EXCEL.

In Microsoft EXCEL, a function is invoked by writing an = sign, followed by the purpose:

For example: **=fpudevice().**

In Microsoft EXCEL, the parentheses must be included, whether or not the call has parameters.

fparchive( )

PURPOSE: The **fparchive()** function is used in a spreadsheet to switch between the archival and the historical database to retrieve values. The same function calls used on the historical database are used on the archival database to retrieve data.

COMMAND: **double fparchive** (**double** *flag*)

Parameter	Description
flag	Use the value 1 to select the archival database for data retrieval or use 0 to select the historical database.

RETURNS: Current value of the archival flag.

Example: =fparchive(1)

fpblkarch( )

PURPOSE:

The **fpblkarch()** function is used in a spreadsheet to switch between the historical block archival database and the historical block database. the same function calls used on the historical block database are used on the historical block archival database to retrieve data.

COMMAND:

double *fpblkarch(double *flag*)

Parameter	Description
flag	Use the value 1 to select the historical block archival database for data retrieval or use 0 to select the historical block database.

RETURNS:

Current value of the archival flag.

Example:

This example selects the Historical Block Archival Database as the source for data retrieval.

```
=fpblkarch(1)
```

fpblkclose( )

PURPOSE: The **fpblkclose()** function is used to close the descriptor for the named historical block. Typically this function is called after opening an historical block and retrieving all required data.

COMMAND: **char** ***fpblkclose**(**char** **blockname*)

Parameter	Description
blockname	The name of block to be closed.

RETURNS: Name of the block.

Example: =fpblkclose("CALBLK1")

fpblkopen( )

FUNCTION:

The **fpblkopen()** function is used to open a descriptor for the named historical block. This function must be called (once) prior to retrieving any data contained in the block.

COMMAND:

char ***fpblkopen**(**char** **blockname*)

Parameter	Description
blockname	The name of block to be opened.

RETURNS:

Name of the block.

Example: =fpblkopen("CALBLK1")

fpblksample( )

PURPOSE: The **fpblksample()** function is used to obtain historical block data for the block name, sample type, and time interval specified. Each data value can then be retrieved using the **fpblkvald** function with the appropriate single tag, time specification.

Command **double** ***fpblksample**(**char** **blockname*, **char** **type*, **double** *stime*, **double** *etime*)

Parameter	Description
blockname	The name of block to be opened.
type	The sample type indicator (Table C-1); null for raw data.
stime	Start Time and Date in spreadsheet format.
etime	End Time and Date in spreadsheet format.

Table C-1. Sample Type Atoms

Description	Extension
Average	
Hourly Average	.HAV
Daily Average	.DAV
Monthly Average	.MAV
Yearly Average	.YAV
Minimum	
Hourly Minimum	.HMN
Daily Minimum	.DMN
Monthly Minimum	.MMN
Yearly Minimum	.YMN
Minimum Time	
Hourly Minimum Time	.HTMN
Daily Minimum Time	.DTMN
Monthly Minimum Time	.MTMN
Yearly Minimum Time	.YTMN
Maximum+	
Hourly Maximum	.HMX
Daily Maximum	.DMX
Monthly Maximum	.MMX
Yearly Maximum	.YMX
Maximum Time	
Hourly Maximum Time	.HTMX
Daily Maximum Time	.DTMX
Monthly Maximum Time	.MTMX
Yearly Maximum Time	.YTMX

Table C-1. Sample Type Atoms (continued)

Description	Extension
Standard Deviation	
Hourly Standard Deviation	.HSD
Daily Standard Deviation	.DSD
Monthly Standard Deviation	.MSD
Yearly Standard Deviation	.YSD

RETURNS: Number of block samples found for time interval between stime and etime.

Example: This example gets all Hourly Average samples which were collected for the last 24 hours.

```
=fpblksample("CALBLK1",".HAV",datevalue("12-JAN-94")+time-
value("12:00:00"),datevalue("12-JAN-94")+timevalue("13:00:00"))
```

fpblkvald( )

PURPOSE: The **fpblkvald()** function gets the value from the Historical Block database which is associated with the specified block name, tagname.atom, sample type, and timestamp.

COMMAND: **double** ***fpblkvald**(**char** **blockname*, **char** **tag*, **char** **type*, **double** *time*)

Parameter	Description
blockname	The name of the block.
tag	The tagname and atom for which the value will be retrieved.
type	The sample type indicator (refer to ***fpblksample( )***).
time	Time and Date in spreadsheet format.

RETURNS: Value of the specified tagname.atom, or an error message.

Example: This example gets the Hourly Standard Deviation value for LIC-454.C_AOUTPUT from the Historical Block named CALBLK1 for 4:00 PM on 10 January 1994.

=fpblkvald("CALBLK1", "LIC-454.C_AOUTPUT", ".HSD", datevalue ("10-JAN-94") + timevalue("16:00:00))

fphget( )

PURPOSE:

The **fphget()** function is used in the spreadsheet as a formula. This function gets the value of a specified tagname from the Historical database. The function **fphget()** returns the value in ASCII format.

COMMAND:

char fphget(**char** **tag*, **double** *time*)

Parameter	Description
tag	Tagname for which the value will be obtained from the Historical database.
time	Time and date in spreadsheet format (refer to the ***Microsoft EXCEL Instruction*** for information on spreadsheet format).

RETURNS:

Value of the specified tagname, or an error message.

EXAMPLE:

This example gets the value for FIC101 from the Historical database, for noon on 12 January 1994.

```
=fphget("FIC101", DATEVALUE("12-JAN-94") + TIMEVALUE("12:00:00"))
```

fphgetd( )

PURPOSE:

The **fphgetd()** function is used in the spreadsheet as a formula. This function gets the value of a specified tagname from the Historical database. The function **fphgetd()** returns the value as a double precision data type.

COMMAND:

double fphgetd(**char** **tag*, **double** *time*)

Parameter	Description
tag	Tagname for which the value will be obtained from the historical database.
time	Time and date in spreadsheet format (refer to the ***Microsoft EXCEL Instruction*** for information on spreadsheet format).

RETURNS:

Value of the specified tagname, or an error message.

Example:

This example gets the value for FIC101 from the historical database, for noon on 12 January 1994.

```
=fphgetd("FIC101", DATEVALUE("12-JAN-94") + TIMEVALUE("12:00:00"))
```

fphput( )

PURPOSE:

The **fphput()** function is used to put a value from the spreadsheet into the historical database, using the tagname specified. The function **fphput()** takes and places the value in ASCII format.

COMMAND:

char fphput(**char** **tag*, **char** **time*, **char** **val*)

Parameter	Description
tag	Tagname of the point in the historical database into which the value will be placed.
time	Time and date in spreadsheet format (refer to the ***Microsoft EXCEL Instruction*** for information on spreadsheet format).
val	ASCII value to be stored in the historical database.

RETURNS:

Value put into the historical database, or an error message.

Example:

This example puts the value for 12.34 (in ASCII format) into the tagname FIC101 in the historical database, for the current time.

```
=fphput("FIC101", DATEVALUE("22-SEP-95") + TIMEVALUE("14:10:00"), "12.34")
```

fphputd( )

PURPOSE:

The **fphputd()** function is used to put a value from the spreadsheet into the historical database, using the tagname specified. The function **fphputd()** takes and places the value in double precision format.

COMMAND:

double fphputd(char **tag*, **char** **time*, **double** *val*)

Parameter	Description
tag	Tagname of the point in the historical database into which the value will be placed.
time	Time and date in spreadsheet format (refer to the ***Microsoft EXCEL Instruction*** for information on spreadsheet format).
val	Double precision value to be stored in the historical database.

RETURNS:

Value put into the historical database, or an error message.

Example:

This example puts the value for 12.34 into the tagname FIC101 in the historical database, for the current time.

```
=fphputd("FIC101", DATEVALUE("22-SEP-95") + TIMEVALUE("14:10:00"),
12.34)
```

fphxgetval( )

PURPOSE:

The **fphxgetval()** function is used in a spreadsheet formula in conjunction with the function fphxnumval. The user must configure the spreadsheet to first call the **fphxnumval()** function, which retrieves *all* the values from the historical database during a specified time period, and returns the number of values retrieved for the time period specified.

The individual values can be placed in cells using the fphxgetval function. The user supplies an index number from one to *n* (where *n* is the number of values returned by the **fphxnumval()** function), and the type of data to be retrieved (a string value, a double value or the time the value was collected).

COMMAND:

fphxgetval(*index, type*)

Parameter	Description
index	Index is a number from 1 to *n* (where *n* is the number of values returned by the fphxnumval function)
type	The type argument specifies the type of the data to be returned. The type is represented by a letter: d - return a double precision value. s - return a string value. t - return the time the value was collected.

RETURNS:

Type of data requested.

Example:

The following example gets the first 30 values (or as many as are available if less than 30 values exist) for the tagname FIC101 from the Historical database, for the last 24 hours. The second function call will return the third of these values as a double precision value. Cell A3 contains the Time/Date of the previous day.

Assignments to support the example are shown in Table C-2. Note that the contents of Cell A1 are subtracted from the contents of Cell A2 (A2-A1) and the result is displayed in Cell A3.

Table C-2. Cell Address and Contents

Cell Address	Cell Contents
CELL A1	1
CELL A2	=now()
CELL A3	+A2-A1

The actual example is shown below:

```
=fphxnumval("FIC101",A3,A2,30)
=fphxgetval (3, "d")
```

fphxnumval( )

PURPOSE:

The **fphxnumvals()** function is used in the spreadsheet as a formula. The user can specify the number of values to be obtained from the historical database (for a specified tag) using the n_vals argument. A starting and ending time further define the data to be obtained. For the specified tagname, this function will return the specified number of values, or (if fewer values exist for the specified time period in the historical database) the maximum number of values available.

The actual number of values obtained for the time period specified is returned. Once data has been obtained using the fphxnumval function, the fphgetd function (refer to ***fphgetd( )***) can be used to return the actual data values to the spreadsheet.

COMMAND:

int fphxnumvals(*tag*, *stime, etime, n_vals*)

Parameter	Description
tag	Tagname for which the values will be obtained from the historical database.
stime	Starting time and date in spreadsheet format (refer to the ***Microsoft EXCEL Instruction*** for information on spreadsheet format).
etime	Ending time and date in spreadsheet format.
n_vals	The number of values to be obtained from the Historical database. If a positive number (*n*) is used, the first *n* values between the specified start time (stime) and end time (etime) will be obtained; if a negative number (*-n*) is used, the last *n* values between stime and etime will be obtained. A number must be specified for the n_vals argument; there is no default.

RETURNS:

Number of values retrieved from the Historical Database.

Example:

The following example gets the first 30 values (or as many as are available if less than 30 values exist) for the tagname FIC101 from the Historical database, for the last 24 hours. Cell A3 contains the Time/Date of the previous day.

Assignments to support the example are shown in Table C-3. Note that the contents of Cell A1 are subtracted from the contents of Cell A2 (A2-A1) and the result is displayed in Cell A3.

Table C-3. Cell Address And Contents

Cell Address	Cell Contents
CELL A1	1
CELL A2	=now()
CELL A3	+A2-A1

=fphxnumval("FIC101",A3,A2,30)

fpput( )

PURPOSE:

The **fpput()** function is used to put a value from the spreadsheet into the current database using the tagname and atom specified. The function **fpput()** takes and places the value in ASCII format.

COMMAND:

char fpput(char ******tag*, **char** ******val*)

Parameter	Description
tag	Tagname and atom in the current database into which the value will be placed.
val	ASCII value to be stored in the current database.

RETURNS:

Value put into the current database, or an error message.

Example:

This example puts the value 12.34 into the current database, in ASCII format, for the tagname FIC-101.C_AOUTPUT.

```
=fpput("FIC-101.C_AOUTPUT", "12.34")
```

fpputd( )

PURPOSE:

The **fpputd()** function is used to put a value into the current database using the tagname and atom specified. The function **fpputd()** takes and places the value in double precision.

COMMAND:

double fpputd(**char** **tag*, **double** *val*)

Parameter	Description
tag	Tagname and atom in the current database into which the value will be placed.
val	The double precision value to be stored in the current database.

RETURNS:

Value put into the current database, or an error message.

Example:

This example puts the value 12.34 in the current database, in double precision format, for the tagname FIC-101.C_AOUTPUT.

```
=fpputd("FIC-101.C_AOUTPUT", 12.34)
```

fpsendmsg( )

PURPOSE:

The **fpsendmsg()** function is used to generate a message that will be displayed in the fourth line of the mini-alarm window.

COMMAND:

char fpsendmsg(**char** **message*, **double** *priority*, **double** *area*)

Parameter	Description
message	Character string which will be used as the message. Note that only the first 80 characters entered will be used; any additional input will be truncated.
timepriority	Priority (1 to 16) to be assigned to the message.
area	Number of the area to be assigned to the message. The message will be broadcast to all areas; however, it will be displayed only on PWC90 consoles where the currently logged in user has access to the area assigned to the message and where the operator messages message class has not been filtered out.

RETURNS:

Message.

Example:

This example sends the message with a priority of one and an area of three.

```
=fpsendmsg("The average pH level is unacceptable",1,3)
```

fpudevice( )

PURPOSE: The **fpudevice()** function is used to return the device name of the device selected by the user to which the spreadsheet is to be sent for printing.

COMMAND: **char fpudevice** ( )

RETURNS: Device name.

Example: This example returns the device name of the selected printer (or file).

```
=fpudevice()
```

fpuprint( )

PURPOSE:

The *fpuprint()* function is used to indicate the mode in which the EXCEL spreadsheet was entered. If a 0 is returned, it indicates that the spreadsheet is in configuration mode, which was entered via the *Configuration* option available via the system window *Menu* button. If a 1 is returned, the spreadsheet is in print mode, which was entered via the *Print Log* option available via the system window *Menu* button.

COMMAND:

char fpuprint ( )

> **NOTE:** When the PRINT LOG option is selected, the EXCEL spreadsheet is displayed as an icon on DCT1 only.

RETURNS:

Mode of the current EXCEL spreadsheet (1 or 0).

Example:

This example returns a 0 if the spreadsheet was entered via the *Configuration* option, or a 1 if it was entered via the *Print Log* option.

```
=fpuprint()
```

fpverbose( )

PURPOSE: The **fpverbose()** function is used to turn on and off messages to the console device. The message provides more detailed information about an error that has occurred in a spreadsheet cell.

COMMAND: **double** *fpverbose* (**double** *flag*)

Parameter	Description
flag	Use the value 1 to turn on messages, use 0 to turn off messages.

RETURNS: Current value of the verbose flag.

Example: This example turns on descriptive error messages.

```
=fpverbose(1)
```

fpxammts( )

PURPOSE:

The **fpxammts()** function is used to get a range of values from the historical database, using the tagname specified. The function **fpxammts()** computes the minimum, maximum, average, total and standard deviation of the values obtained. It also gets the times that the minimum and maximum values occurred. Using the type field the user can retrieve any of these values. If multiple calls are made successively using the same tag, stime, etime and totfac, the values are cached, resulting in one DBA transaction.

COMMAND:

double fpxammts(*tag, stime, etime, totfac, type*)

Parameter	Description
tag	Tagname of the point in the historical database for which values from the database will be obtained.
stime	Time in the historical database at which to start retrieving values (i.e., start time).
etime	Time in the historical database at which to stop retrieving values (i.e., end time).
totfac	Totalization factor, used when computing the total so that the value used in the log reflects the type of units desired by the user. For example, if the value stored in the historical database reflects gallons per minute, that value is multiplied by the value in *totfac* so that the resulting value reflects gallons per day instead of gallons per minute.
type	Type of data requested. Data types which can be requested and the syntax used to request them are as follows.

Table C-4. Data Type and Syntax

Data Type Specifier	Description
imin	Minimum value
imax	Maximum value
iavg	Average value
itot	Total value
istd	Standard deviation
tmin	Time of minimum value
tmax	Time of maximum value

RETURNS:

Value of the type requested.

Example:

Assignments to support the example are shown in Table C-5. Note that the contents of Cell A1 are subtracted from the contents of Cell A2 (A2-A1) and the result is displayed in Cell A3.

```
=fpxammts("HIST1",A3,A2,.02,"imin")
=fpxammts("HIST1",A3,A2,.02,"tmin")
```

fpxammts( ) *(continued)*

Table C-5. Cell Address And Contents

Cell Address	Cell Contents
CELL A1	1
CELL A2	=now()
CELL A3	+A2-A1

This example gets the data for the last twenty-four hours for the tag "HIST1" from the historical database. The first call returns the minimum value and the second call returns the time the minimum value was collected.

Index

Symbols

@aGlance/IT 11-1
- API 11-1
- Client applications 11-21
- Methods 11-13
- Requirements 11-2
- Security 11-17
- Server operation 11-2

A

Alarm
- Group configuration 2-10
- Tag comments 12-22, 12-26
- Vectors 2-24

Archival groups 2-4
Atom suffixes A-1

C

Change password 2-64
Configuration locks 9-10
Configuring network parameters 2-16
Creating summary graphics 4-94
CRT
- Context keys 9-17
- Print 9-9

D

Database maintenance 9-2
Defining
- Engineering units 12-24
- Logic state descriptors 12-25

Displays
- Group 3-5
- Point 3-6
- Summary 3-3

E

Engineering unit descriptors 12-23
Event bits 2-30
Event log 6-2
- Add 6-2
- Delete 6-6
- Edit 6-5
- Print 6-9

EXCEL Functions 1-5, C-1

G

Glossary 1-6
Graphics
- Change object 4-19
- Configuration 4-1
- Creating objects 4-5
- Dynamics 4-51

Group
- Configuration 3-8
- Displays 3-5

H

Hierarchical displays 3-3
Historical block
- Add 10-2
- Copy 10-8
- Delete 10-8
- Edit 10-8
- Find 10-9

Historical block data 10-1
Historical database 5-1
- Edit 5-2
- Operations 5-18

L

Log configuration 6-1
Logging in 2-2
Logging out 2-4
Logic state descriptors 12-24
Loop status display 4-93
Loop status summary 4-95

M

Message
- Configuration 2-8
- Routing 2-38

N

Node status display 4-93

P

Point displays 3-6
Printer assignment 2-32
Process node summary 4-96

Index (continued)

Q

Quick keys....9-13

R

Remote motor control text....12-27

S

Security....2-49
- Activation....2-49

Spreadsheet log....6-10
- Add....6-10
- Delete....6-10
- Edit....6-10
- Schedule....6-11

Summary displays....3-3
System status display....8-1

T

Tag atoms....A-1
Tag database
- Alarm comments....12-22, 12-26
- Defining a tag....12-9
- Tag descriptors....12-23
- Tag types....12-1
- Tags....12-1

Trending....7-1
Trends....7-10

U

User login....2-57
- Add....2-58
- Cancel....2-64
- Delete....2-64
- Edit....2-64

Utilities....9-1

Visit Elsag 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

Form WBPEEUI220762A Litho in U.S.A. 297

Copyright © 1997 by Elsag Bailey Process Automation, As An Unpublished Work
® Registered Trademark of Elsag Bailey Process Automation
™ Trademark of Elsag Bailey Process Automation