Bailey infi 90®

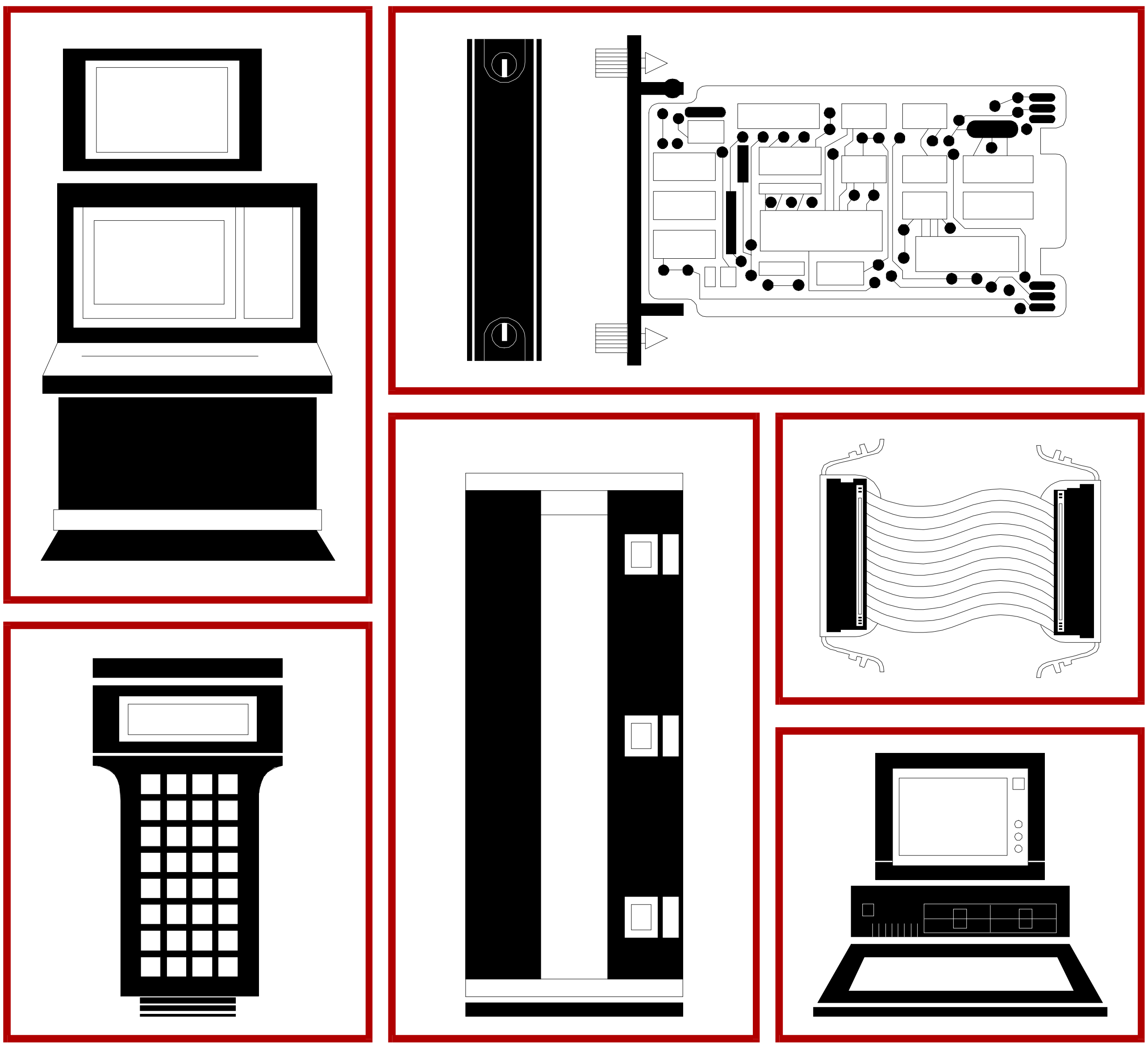
40 Series

Instruction

Configuration

Operator Interface Station

(Release J)



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 **Elsag Bailey** INSTRUCTIONS AND MANUALS; OTHERWISE, INJURY OR DAMAGE MAY RESULT.

RADIO FREQUENCY INTERFERENCE

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

POSSIBLE PROCESS UPSETS

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

Preface

This instruction gives the procedures to configure a Signature Series Work Station or 40 Series Operator Interface Station. The instruction reflects the J.1 software release for the console.

NOTE: In this instruction, the 40 series refers to only the IIOIS42, IIOIS43, IIOIS42^{PLUS}, IIOIS41^{PLUS2}, Signature IS42, IS43 and Signature IS42^{PLUS} consoles.

There are three additional instructions that explain how to set up and use the console:

Hardware Provides hardware installation, troubleshooting, maintenance, repair, and replacement procedures.

File Utilities Provides software load, upgrade, and maintenance procedures, and save and restore configuration procedures.

Operation Gives a brief console and INFI 90® Open system overview to familiarize the reader. It then explains the operations that can be performed after configuring the console.

This instruction provides configuration information for the OIS application only. Refer to the ***Operation*** instruction for those configurations performed outside the OIS application such as touch pad, touch screen, and screen copy printer configuration. These are windows applications.

In addition to standard console functions, this instruction discusses certain functions that apply to systems, such as the open access system and user task interface, that are only available to meet unique plant requirements. Unless applicable to your plant operations, disregard the options that support these nonstandard systems and the sections in this instruction that describe the options.

List of Effective Pages

Total number of pages in this instruction is 464 consisting of the following:

Page No.	Change Date
Preface	Original
List of Effective Pages	Original
iii through xvii	Original
1-1 through 1-10	Original
2-1 through 2-13	Original
3-1 through 3-19	Original
4-1 through 4-7	Original
5-1 through 5-29	Original
6-1 through 6-32	Original
7-1 through 7-10	Original
8-1 through 8-10	Original
9-1 through 9-54	Original
10-1 through 10-4	Original
11-1 through 11-21	Original
12-1 through 12-47	Original
13-1 through 13-9	Original
14-1 through 14-9	Original
15-1 through 15-11	Original
16-1 through 16-12	Original
17-1 through 17-9	Original
18-1 through 18-24	Original
19-1 through 19-5	Original
20-1 through 20-5	Original
21-1 through 21-11	Original
A-1 through A-6	Original
B-1 through B-6	Original
C-1 through C-5	Original
D-1 through D-5	Original
E-1 through E-5	Original
F-1 through F-8	Original
Index-1 through Index-5	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

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

List of Figures

List of Figures (continued)

List of Figures (continued)

List of Tables

List of Tables (continued)

List of Tables (continued)

Trademarks and Registrations

Registrations and trademarks used in this document include:

™ @aGlance/IT	Registered trademark of Intuitive Technology Corporation.
™ DECnet	Registered trademark of Digital Equipment Corporation.
™ MOTIF	Registered trademark of Open Software Foundation, Incorporated.
™ OpenVMS	Registered trademark of Digital Equipment Corporation.
™ X Windows	Registered trademark of Massachusetts Institute of Technology.
® IBM	Registered trademark of International Business Machines Corporation.
® INFI 90 OPEN	Registered trademark of Elsag Bailey Process Automation
® INFI-NET	Registered trademark of Elsag Bailey Process Automation.
® Network 90	Registered trademark of Elsag Bailey Process Automation.

SECTION 1 - INTRODUCTION

OVERVIEW

This instruction contains information and instructions necessary to configure an 40 Series Signature Operator Interface Stations or an 40 Series Operator Interface Stations. The operator interface station (OIS) provides integrated operations interface, data acquisition, and reporting capabilities in addition to process control for the INFI 90 OPEN Strategic Enterprise Management System.

Figure 1-1 shows the primary console components of IIOIS40 Series consoles. The IS40 Series consoles have the same primary components but their locations are different. An auxiliary console has most of the same components. Refer to the ***Operation*** instruction for a description of each component.

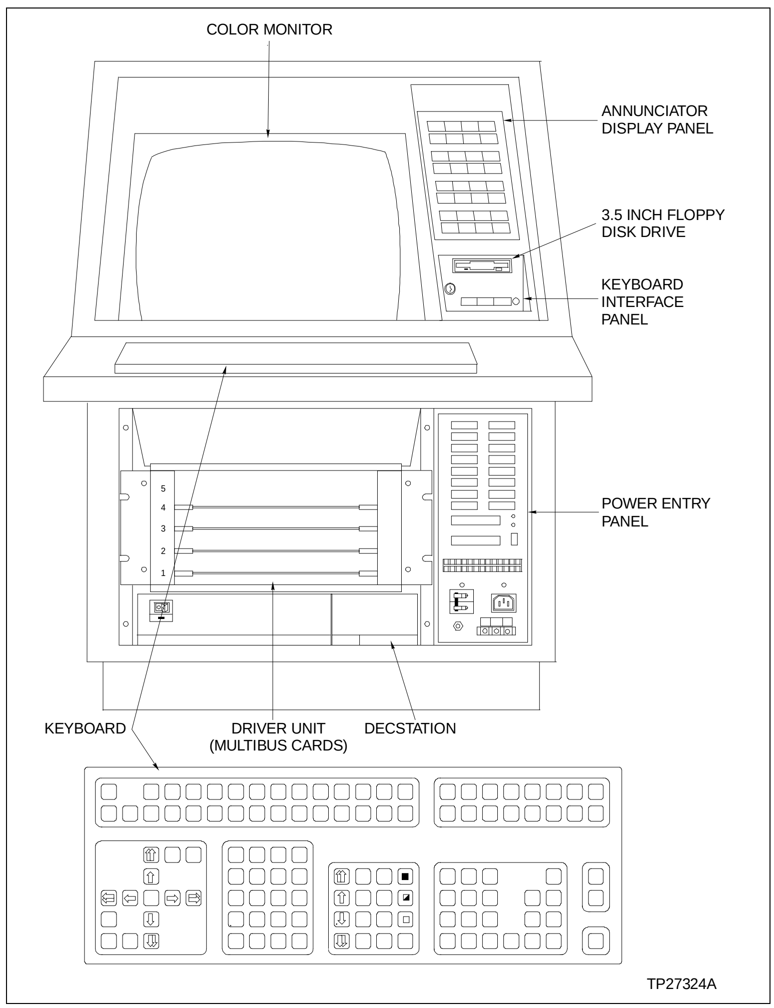


Figure 1-1. Operator Interface Station

CONSOLE TYPES

The terms main console and auxiliary console are general terms used throughout this instruction to refer to certain types of consoles:

Main Console IS42, IIOIS42, IS43, IIOIS431, IIOIS432, or IIOIS433 console; an IIOIS43A or IIOIS43D driver cabinet is also a main console, but it requires an auxiliary console for interface.

Auxiliary Console IIOIC411, IIOIC4121, IIOIC4122, IIOIC4123, IIOIC413, IIOIC414, or IIOIC42 console.

The same process operations can be performed on both main consoles (OIS) and auxiliary consoles (OIC).

INTENDED USER

Use this instruction as a reference when configuring the console. This instruction is ***not*** a tutorial for process control and assumes the reader has a general knowledge of graphics-based process control systems. It explains console interaction with the INFI 90 OPEN system only. After completely reading and understanding the information presented, the system engineer or technician should have the knowledge required to tailor the console operation to the specific requirements of the process.

This instruction assumes the reader is familiar with X™ windows using MOTIF™ style windows and window manager.

INSTRUCTION CONTENT

This instruction contains information necessary to configure the console. Step-by-step procedures to set up each console function, component, and peripheral for proper operation are provided. It includes a Table of Contents, List of Figures, List of Tables, and Index, which give several options to locate specific information quickly. Appendices supplement information presented in the individual sections.

The sections in this instruction include:

Introduction Describes the instruction content and intended user. It also highlights console features.

General Information Contains general information needed to configure the console such as saving and exiting a configuration. It also explains the steps necessary to begin operating the OIS application: Start-up, key lock position, and password log-in.

System

Contains the procedures to set up overall console operation. The console must be set up as a system before any other configuration is performed. Specifically, this section:

- Describes console operating parameters and explains how to enable and account for supported peripherals.
- Gives the procedures to set up exception report processing and the order the console performs its remaining processing responsibilities.
- Explains how to set the time and date format.

Window Management

Contains the procedures to direct windows to local screens and remote nodes. Windows are directed to various locations by using the *X Device Definition* function or by logging into the OISWIN account from a remote node.

Password Security

Contains the procedures to set console security features including:

- Security options.
- Security levels.
- User password definitions.
- Display masks.
- Tag security level and group assignments.
- Log security level assignment.

Tag Database

Provides procedures to define a tag and to define alarm comments, engineering unit descriptors, and logic state descriptors. It also explains tag broadcasting.

Display Generation

Explains options for creating displays and the symbols used in displays. It also explains procedures to assemble a display source file into OIS format and to view errors encountered during the assembly process.

Keyboard and ADP

Gives procedures to assign displays, user tasks, and key macros to keyboard keys and annunciator display panel (ADP) pushbuttons. It also explains key macro definition.

Alarm Management

Explains how to enable console indications that notify of an alarm condition such as indicators and internal and external annunciators. It provides the procedures and describes the requirements to configure:

- Automatic inhibiting.
- Global acknowledge and silence.
- Groups.
- Management options.
- Priorities.
- Quality indications (both good and non good quality).
- Relays (external annunciators).
- Remote acknowledge.

- Summaries.
- Summary reports.
- Tones.

Open Access System Explains how to identify an open access system (OAS) to the console. The open access system is a network node that can be used to store trend data and logs for the console. Trend data stored by the open access system can be viewed on the console in trend displays.

Trends Explains the requirements and procedures to define a trend. A trend definition contains information needed to collect and present trend data in displays and logs.

Logging Contains the procedures to create and enable recording process and console data in hard copy reports: System event (and operator actions), custom, and sequence of events (SOE) logs.

Peripherals Provides a list of available printer types for logging and explains printer configuration and assignment. It also gives the procedures to enable automatic failover if a printer fails.

Operator Configurable Displays Explains how to modify the operation of the operator configurable displays function.

Display Call-Up Options Contains the procedures to set up primary, automatic, and multiwindow display call-ups.

Text Definition and Substitution Provides the procedures to change default console text to user-defined text. Most text can be changed by text substitution. It also identifies the various text definition functions and explains how to define text for:

- Error codes of remote motor control blocks.
- Text selector function blocks.

XY Plot Definition Contains the procedures to define an XY plot for use in a plot display.

Trend Pen Provides the procedures to enable trend pen recorders and to select the process variables to be tracked. This sets up the interface logic between recording devices and the console.

@aGlance/IT™ Provides the procedures to select UTI instead of @aGlance/IT, name the @aGlance/IT server(s), and identify the user-written server command. It defines an index number for a user-written server command which is used in other configurations to enable executing the user-written server command from the console.

UTI User Task Definition Explains how to identify a user-written program enabled through the user task interface (UTI). It defines an index number for a program which is used in other configurations to enable executing the program from the console.

Alternate Language Explains requirements for alternate language. The procedures in this section should be performed prior to any other configuration to allow entering alternate language characters during configuration.

HOW TO USE THIS INSTRUCTION

This instruction provides configuration information for the OIS application only. Refer to the ***Operation*** instruction for those configurations performed outside the OIS application such as touch pad, touch screen, and screen copy printer configuration. These are windows applications.

To attain optimum use of all available features and functions, it is important to become familiar with the entire contents of the instruction before configuring the console. To use the instruction:

1. Before configuring the console, refer to the ***Operation*** instruction:

- For an overview of the INFI 90 OPEN system and console functions.
- To become familiar with the keyboards, annunciator display panels, display system, and menu structure.
- For a description of window operations specific to the OIS application running on the console.

2. Read Section 2 for general configuration information for the OIS application. This section gives a general description of configuration pages, data input procedures, and configuration save and exit procedures. It then describes OIS application start-up, the key lock and its function, and steps to log in a password (password for the OIS application).

3. Read Section 3 for a description of system configuration requirements and procedures.

4. Read the remaining sections as needed. For example, to configure logs, read Section 12. Not all configuration procedures are required. Plant requirements determine the configurations required.

5. Refer to the ***Operation*** instruction for information about OIS utilities. The instruction provides information about the screen copy, touch screen, and touch pad utilities.

6. Refer to the ***Operation*** instruction for operating procedures after the console has been configured.

7. Refer to the ***Operation*** instruction for information on auxiliary keyboards and error codes after the console has been configured.

Hint To find information on a specific option, refer to the index. All options described in this instruction appear under the heading ***Option***. Not all options are described in this instruction. Refer to the ***Operation*** instruction for options not listed.

Read the ***notes*** in text. Notes provide:

- Additional information.
- Information that should be considered before performing a certain operation or function.

DOCUMENT CONVENTIONS

NOTE: The OIS application is case sensitive. For example, entering **tagname** will not call the tag **TagName**. They are two distinct names.

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 in this document.

Example: *A OIS Configuration* (menu selection)
General Functions Menu (display title)
SELECT a Cell Item from the Menu (message)
Tag name or index number (prompt)

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

Example: ***DISPL1.DU***
DT

GLOSSARY OF TERMS AND ABBREVIATIONS

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

Table 1-1. Glossary of Terms and Abbreviations

Term	Definition
Display element	A discrete element used in creating a process display; station faceplate, annunciator, trend, and deviation overview are display element examples.
EUD	Engineering unit descriptor.
EWS	Engineering work station.
Exception report	Information update generated when the status or value of a point changes by more than a specified significant amount; abbreviated as XR.
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.
INFI-NET®	Advanced data communication highway.
LSD	Logic state descriptor. A character string which is associated with the state of a digital point.
MCS	Management command system. Integrated operator console with data acquisition and reporting capabilities. It provides digital access into the process for control and monitoring.
MFC	Multi-function controller module. A multiple loop controller with data acquisition and information processing capabilities.
MFP	Multi-function processor module. A multiple loop controller with data acquisition and information processing capabilities.
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.
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.
OIS	Operator interface station. Integrated operator console with data acquisition and reporting capabilities. It provides a digital access into the process for flexible control and monitoring.
Peer-to-peer	Refers to the communication between different modules at the same communication hierarchical level within an INFI 90 system.
Plant Loop	Network 90® data communication highway.
Static element	The background graphics and text of a display. These display elements do not change as the process changes.
Symbol	A drawing element. Using symbols for commonly used drawing elements eliminates redrawing.
Tag	An analog or digital process value defined in the console or PCV as a control or monitor point.

REFERENCE DOCUMENTS

This instruction provides configuration information only for the console. Table 1-2 lists additional documents that relate to the console and are referenced in this instruction.

Table 1-2. Reference Documents

Number	Title
I-E93-917-1	Sequential Events Recorder (SER)
I-E96-200	Function Code Application Manual
I-E96-631	INFI 90 OPEN Distributed Sequence of Events
I-E96-703	C Utility Program
I-E96-754	User Task Interface
I-E96-770	Module Configuration Tools (WCAD)
I-E96-771	Console Configuration Tools (WLDG)
I-E96-825	Software Global Database Manager (SGDM)
WBPEEUI220756A1	Operation (40 Series)
WBPEEUI220758A1	File Utilities (40 Series)
WBPEEUI220759A1	Display Builder Reference
WBPEEUI220760A0	Hardware, Operator Interface Console (IIOIS43)
WBPEEUI220761A0	Hardware, Operator Interface Station (IS43)
WBPEEUI350255A0	Open Data Server/Client

CAPABILITIES SUMMARY

Table 1-3 summarizes console software capabilities. Refer to the specifications in the ***Hardware*** instruction for power requirements, certification, and operating environment. Also, refer to the instruction for the type and number of peripherals supported by a specific type of console (e.g., IIOIS432, IIOIS43D, IIOIS43A, etc.).

Table 1-3. Software Capabilities Summary

Function	Description		Capability[1]
Archiving System[2]	Data types		Events Logs Module configurations Process (tag) data Trend data
	Storage media		Magnetic tape Open access system (OAS) Optical disk
	Storage period		4 hrs (minimum) 99 wks (maximum)
Database	Alarm comments		20,000
	Tags	IIMCP01	10,000
		IIMCP02	30,000
	Trends	Definitions	10,000
		Display only	10,000
		External source (OAS node)	5,000
		Save to disk	2,000

Table 1-3. Software Capabilities Summary (continued)

Function	Description		Capability[1]
Display System	Displays (graphics)		1,500
	Dynamics	Dynamic bars per display[3]	400
		Dynamics per display[4]	200
		Tags displayed per console[5]	1,600
	Operator configurable displays		25
Logging System	Custom	Logs	300
		Collection period	1 sec (minimum) 999 hrs (maximum)
		Collection triggers	Demand Process event Time
		Print triggers	Collection completion Demand Process event Time
		Saved copies (retentions)	9
	Event	Number of events	1,000
		Types	Continuous (as events occur) Periodic
		Print period	1 hour (minimum) 24 hours (maximum)
	Sequence of events	Logs	160
		Recorders INSEM01 modules SOE (FC 210) function blocks	32
		Saved copies (retentions)	9
Peripherals	Annunciator display panels		32 (4 per keyboard interface)
	Keyboards	Auxiliary engineering (QWERTY)	8
		Operator	8
		Logical relays	48 (6 per keyboard interface)
		Logical tones	20 (5 per keyboard interface)
	Printers	Copy screen[2]	4
		Logging	4
Trending System	Enhanced trend	Display resolution	1 sec (minimum)
	Standard trend	Display resolution: Fast Normal	 15 sec (minimum) 1 min (minimum)
	Operator assignable trends		20

Table 1-3. Software Capabilities Summary (continued)

Function	Description	Capability[1]
Windowing System	Windows	8
	Transport protocol	DECnet™ TCP/IP

NOTES:

1. Values are maximums unless stated otherwise.
2. Refer to the ***Operation*** instruction for information.
3. Maximum if a display contains only dynamic bars.
4. Maximum if a display contains any combination of dynamics.
5. Maximum in all windows at one time.

SECTION 2 - GENERAL INFORMATION

INTRODUCTION

This section gives general information needed to configure the console. The console must be configured before it can collect and display INFI 90 OPEN process data, and displays must be designed and built to show the collected process information. Security, alarm management, and data recording capabilities can also be set up.

This instruction mainly covers OIS application configuration. Refer to the discussion on windows operations in the ***Operation*** instruction (Table 1-2 lists instruction numbers.) for information on using and operating with windows. Refer to the discussion on windows applications and utilities for information on:

- Print screen configuration utility.
- Touch screen setup utility.
- Touch pad configuration utility.

GENERAL FUNCTIONS MENU

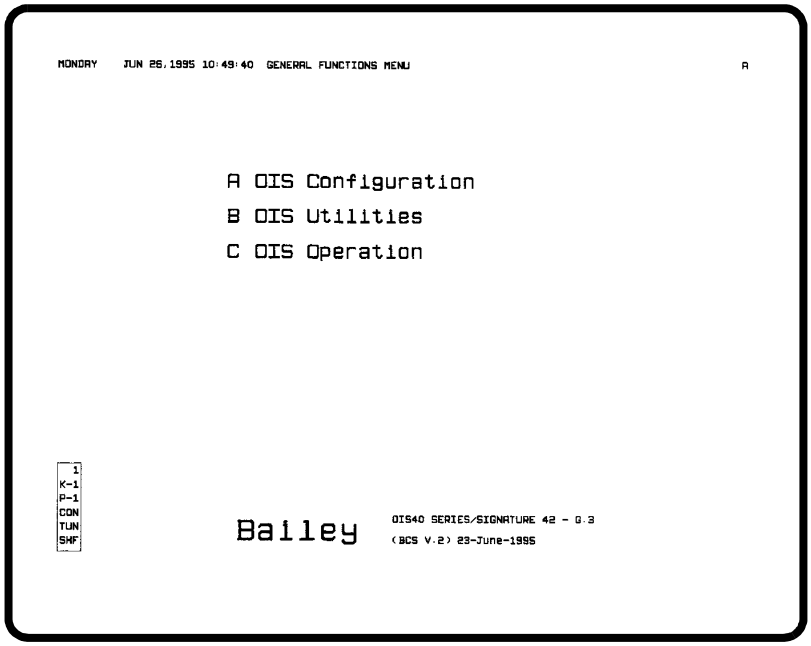
The *General Functions Menu* provides access to configuration functions (Figure 2-1). Press to call this page from another. All configuration procedures in this instruction start from the *General Functions Menu* and use either the *A Configuration* or *B Utilities* options. Refer to Appendix B for a tree structured view of the *General Functions Menu* hierarchy.

CONFIGURATIONS

The console configurations reside on the hard disk mostly in ***CF*** files. Each configuration option has its own configuration page or set of pages. Configuration pages are interactive, fill-in-the-blank type displays. The same configuration page is used to initially define a configuration and to edit an existing configuration. The procedures and the input fields for either purpose remain the same.

All inputs to the console are checked for validity. If an entry is invalid, the input cursor positions on the field until it is corrected, cleared, or reset to its original entry.

In this instruction, a configuration procedure will list the valid entries for that configuration. The text strings given are defaults. Optionally, the text substitution function can be called to view the valid entries (text strings) for a configuration. For example, select item *9 Tag Type* in the text substitution



TPS0752F

Figure 2-1. General Functions Menu

function to find the valid tag types that can be entered during tag configuration. The text substitution function can also be used to change the default text strings for a particular configuration. If text has been changed through text substitution, the function must be called to obtain a list of valid entries. The text strings given in the procedures are no longer valid. Refer to ***TEXT SUBSTITUTION*** in Section 16 for the procedures to use the function.

Sections of this instruction explain the configuration procedures as if performing the initial console configuration. Some procedures may not apply for your particular operation and can be skipped. If modifying an existing configuration, some steps may not be required depending on the modification. Valid entries for input fields are given either in the configuration procedures or in tables related to the procedures.

The order in which this instruction presents the configuration procedures may or may not be the best order for your specific needs. System, database, and display generation are the minimum configuration requirements. After the console is completely configured, a backup copy of the configuration can be made using a tape drive. Refer to the ***File Utilities*** instruction for the procedures to back up (save) the configuration (Table 1-2 lists instruction numbers).

Using the Console Configuration Tools (WLDG)

The preferred method for configuring the console is by using the console configuration tools (WLDG) program then transferring configuration files to the console. Configuration options available on the console can be used to modify or make additions to console configurations regardless of where the configuration was created.

The console configuration tools program supports most console configuration requirements. The following console functions can be configured, defined, or created using the tools program:

- Alarm comments.
- Annunciator display panels.
- Automatic displays and pop ups.
- Custom logs.
- Engineering unit descriptors.
- Keyboards.

  Alarm relays.
  Alarm tones.
  Function keys.

- Logic state descriptors.
- Partial system configuration (general parameters).

  Number of tags.
  Number of trends.
  Number of keyboards.
  Number of annunciator display panels.

- Tags.
- Trends.
- User-created displays.

  Alarm summary.
  Site specific.
  Trend.
  XY plot.

The console configuration tools program provides tools and an easy method for creating displays. A utility for building displays is provided on the console; however, it is better suited for display editing than display creation.

If changes are made to a configuration or display from the console, it is suggested that the configuration files affected be copied to the work station running the console configuration tools program. This will maintain the configuration file integrity. Refer to the ***Console Configuration Utilities*** instruction for information on transferring files between the console and the console configuration tools program (Table 1-2 lists instruction numbers).

Moving the Input Cursor

An input cursor normally appears as a highlighted field (reverse video) that allows data entry. The cursor control keys on the keyboard are used to move the input cursor between input fields of a configuration page. For some configuration pages, a select field allows moving the input cursor to a specific field. A number appearing next to an input field on a configuration page indicates that this feature exists for the page. To use the select field to move the input cursor:

1. Press **ESC** while on a configuration page to call the select field. The field also appears when a configuration page that has this feature is initially called.

2. Enter the index number of the input field to which to move, then press **ENTER**.

Saving a Configuration

A change to a configuration can be saved by selecting a save option, if available, or by pressing **ENTER** In some cases, pressing **NEXT PAGE** or **PREV PAGE**, or moving from the current input field, also saves a configuration change. Pressing **ENTER** will insure the change is saved.

Press **ESC** before saving to exit the configuration page without saving any changes.

Exiting a Configuration Page

Press **ESC** while on any configuration page to exit the page. Continue to press **ESC** to eventually return to the *General Functions Menu.*

KEYBOARD

Configure the console using either the standard operator keyboard or an auxiliary engineering keyboard. Both provide the same functions; although, it may be easier and faster to perform configuration procedures with the auxiliary engineering keyboard. The operator keyboard is set up to facilitate process control and monitoring operations where the auxiliary engineering keyboard is a QWERTY style keyboard better suited for data input. Refer to to cross reference auxiliary engineering keyboard keys to operator keyboard keys.

Keys used in configuration procedures are located in the alphabetic character block, cursor control block, and numeric keypad block sections of the operator keyboard. Refer to the ***Operation*** instruction for the location of specific keys within a keyboard block (Table 1-2 lists instruction numbers).

Table 2-1 lists keys specific to configuration and their functions.

Table 2-1. Configuration Keys

Key	Function
A-Z, 0-9, punctuation	Enters alphanumeric data and values, ASCII characters, and punctuation.
CLEAR	Clears the input field on which the input cursor is positioned. Use the key to erase an entry that is in error or is old data.
ENTER	Enters current data to update configuration files. In most cases, this key must be pressed after completing all entries on a configuration page.
ESC	Returns to a previous menu and eventually the *General Functions Menu* from the current page. Any data keyed in may be lost unless **ENTER** is pressed before **ESC**. On some displays, this key calls a select field that can be used to specify, then move the input cursor to a specific input field. It also provides a cancel option. Prompts explain additional use of this key.
HOME	Resets the current input field to the text string it held when the page was first called. To reset a field, press **HOME** while on the field but before moving to the next field or pressing any other key.
SPACE	Fills a character position in an input field with a blank space.
TAB	Moves the input cursor to the next logical input field for input determined by the console. If there is an error, the cursor stays on the field in error and an error message appears. Tabbing also enters data for the field from which the cursor was tabbed.
TAB BACK	Performs the same functions as **TAB**, but moves to the previous logical input field.
⇧ ⇩ ⇦ ⇨	Moves the input cursor one field in the direction of the arrow. Use these keys to randomly move about a page.
← →	Moves the input cursor within an input field one position in the direction of the arrow.

OPERATING THE OIS APPLICATION

1. Start up the OIS application if it is not already running. Refer to ***OIS START-UP*** in this section for the procedures.

2. After start-up, check to see if there is a flashing *A* at the upper right corner of the window. The *A* indicates an operator action is required. Use the *Operator Action Requests* option to view the action message. If the following message appears as an operator request for action, use the *Set Date and Time* option to set Greenwich Mean Time (GMT).

GREENWICH MEAN TIME MUST BE INITIALIZED

Refer to the discussion on operator action requests in the ***Operation*** instruction for information about using the *Operator Action Requests* option. Refer to the discussion on setting the date and time in the ***Operation*** instruction for information on how to set GMT time.

3. Log in a password. Refer to ***SECURITY PASSWORD*** in this section for the procedures.

OIS START-UP

NOTE: Refer to the ***File Utilities*** instruction for the procedures to log into an account. Refer to the ***Operation*** instruction for a description of the *Login Window* option for logging into an account.

Start-up is automatic after powering up the console. A load sequence executes after applying power. Power for the console turns on from the power entry panel. To turn the power on:

1. Open the front cabinet door to gain access to the main power circuit breaker.

2. Set the breaker to the on (or up) position. The POWER ON lamp should light to indicate power to the console. The console runs a power up sequence, then an automatic start-up sequence. A complete start-up takes several minutes.

The automatic start-up sequence for the console consists of starting the windowing system, opening the session manager and message windows, and initializing the OIS application. The OIS application appearing in an open window or as an icon identifies a successful start-up.

OIC Start-Up

An OIS application window or icon does not appear on an auxiliary console unless directed to that console from a main console. Use *X Device Definition* to direct a window to an auxiliary console.

Log into the OISWIN account to perform remote window assignment. Logging into the OISWIN account initiates an interactive program that allows viewing and changing window assignments. The account can be used to redirect a window and must be used to initially direct a window to an auxiliary console of a driver cabinet.

Refer to Section 4 for a description of both *X Device Definition* and remote window assignment.

Aborting Automatic Start-Up

The ability to cancel the automatic start-up of the OIS application is provided. This can be performed, for example, if the OISENGR account is to be logged into to run the database builder. Canceling the automatic start-up eliminates the need to wait for the OIS application to initialize, then shutting down the application by using a pull-down menu, terminal window, or VT-series terminal.

To cancel the automatic start-up of the OIS application:

1. Power up the console as normal.

2. Wait for the session manager icon to appear.

3. Immediately open the session manager.

4. Choose *Startup/Shutdown*.

5. Choose *Abort Auto OIS*.

This must be performed before the console begins initializing the OIS application. The console waits approximately 30 seconds after initializing all other applications before it begins initializing the OIS application.

Start-Up

Normally, the OIS application automatically starts up after powering up the console. A session manager pull-down menu provides the ability to start the OIS application if it has been shut down or after aborting the automatic start-up sequence. Start-up can also be performed remotely from a terminal window or a VT-series terminal.

> **NOTE:** A start-up affects the OIS application running on both the main console and on the auxiliary consoles to which the main console directs its windows.

To start up the OIS application running on the main console:

1. From the session manager window, choose *Startup/Shutdown*.

2. Choose *OIS Startup*. Do ***not*** select *OIS Startup* again, or *OIS Shutdown* or *OIS Reset*, while the console performs its start-up sequence. Look at the message window to determine the status of the start-up.

After the start-up sequence completes, an OIS application window and icons appear to indicate a successful start-up. Normal process operations can now be performed.

Typing **OISSTARTUP** from a logged in VT-series terminal or a terminal window performs the same function. For a driver cabinet, start up the application from an auxiliary console or a VT-series terminal.

Shutdown

A session manager pull-down menu provides the ability to shut down the OIS application on the main console or from an auxiliary console. Shutdown can also be performed remotely from a terminal window or a VT-series terminal.

NOTE: A shutdown affects the OIS application running on both the main console and on the auxiliary consoles to which the main console directs its windows.

To shut down the OIS application running on the main console:

1. From the session manager window, choose *Startup/ Shutdown*.

2. Choose *OIS Shutdown*. After a short time, all displays and icons disappear. Do ***not*** select *OIS Shutdown* again, or *OIS Startup* or *OIS Reset*, while the console performs its shutdown sequence. Look at the message window to determine the status of the shutdown.

Typing **OISSHUTDOWN** from a logged in VT-series terminal or a terminal window performs the same function. For a driver cabinet, shut down the application from an auxiliary console or a VT-series terminal.

Reset

Some configuration procedures require a reset to enter changes to the OIS operating parameters. A reset also may be required due to a system problem. A reset of the OIS application does not require a physical shutdown of the entire console. A session manager pull-down menu provides the ability to reset the OIS application on the main console or from an auxiliary console. Reset can also be performed remotely from a terminal window or a VT-series terminal.

NOTE: A reset affects the OIS application running on both the main console and on the auxiliary consoles to which the main console directs its windows.

To reset the OIS application running on the main console:

1. From the session manager window, choose *Startup/ Shutdown*.

2. Choose *OIS Reset*. After a short time, all displays and icons disappear, then reappear at reset completion. Do ***not*** select *OIS Reset* again, or *OIS Startup* or *OIS Shutdown*, while the console performs its reset sequence. Look at the message window to determine the status of the reset.

Typing **OISRESET** from a logged in VT-series terminal or a terminal window performs the same function. For a driver cabinet, reset the application from an auxiliary console or a VT-series terminal.

KEY LOCK

IIOIS40 Series Console

For the IIOIS42 and IIOIS43 consoles, the used to limit access to tuning and configuration functions. The only way to change the key lock position is to insert a key and turn. After changing the switch position, remove the key. This locks the key switch in its current position.

To access configuration functions, put the key lock in the position. To access tuning functions, put the key lock in the TUNE position. Switching the key lock back to the LOCK position before saving any modifications discards any and all modifications.

IS40 Series Console

The IS42 and IS43 consoles do not have a TUNE/CONFIG key lock. Instead, the same functionality is handled with passwords. Password security is configured with default passwords that control the key lock functionality: BOTHLOCK, CFGLCK, NOLOCK, and TUNLCK.

The NOLOCK password disables the CONFIG/TUNE key lock functionality. The CFGLCK password disables the CONFIG key lock, but leaves the TUNE key lock enabled. Use this password to configure the OIS application. In this case, no tuning or access to functions that require tuning access rights will be allowed. The TUNLCK password disables the TUNE key lock, but leaves the CONFIG key lock enabled. Use this password to access tuning functions. In this case, no OIS application configuration will be allowed.

If key lock is not in the correct position when attempting to access a certain function, or in the case of the IS40 series console the correct password is not logged in, the following prompt appears:

Function under TUNE Keylock Control

- ***or*** -

Function under CONFIG Keylock Control

Password security configuration can be used to enable or disable the key. If desired, passwords can be created to disable key lock functionality entirely. Refer to Section 5 for a detailed explanation of password security.

SECURITY PASSWORD

The operations that can be performed within the OIS application depend on the key lock position and password security if implemented. A default user is automatically logged in during start-up. The user that is logged in depends on password security configuration.

Log-In

A password must be logged in to gain access to OIS application functions. The operations that can be performed and the configurations that can be accessed by a user depend on password security configuration. To log in:

1. Press **COM'D LINE MENU**.

2. Press **F** to select the *PASSWORD* option. A field with all asterisks appears.

3. Enter a valid password. Passwords are defined during password security configuration. If site specific passwords have not been defined, use a default password: OPERATOR, BOTHLOCK, CFGLCK, NOLOCK, TUNLCK, or MAINT.

> **NOTE:** The BOTHLOCK, CFGLCK, NOLOCK, and TUNLCK passwords are not available on IIOIS42 and IIOIS43 consoles. They are used for IS42 and IS43 consoles which do not have a key lock.

By default, the OPERATOR password gives access to all functions except security maintenance and enables the key The MAINT password denies access to all functions except security maintenance and disables the key

For an IS40 series console, the NOLOCK password is the same as the OPERATOR password except it disables the CONFIG/ TUNE key lock functionality. The BOTHLOCK password works the same as the OPERATOR password. Use either password to deny access to all functions that require tuning and configuration access. The TUNLCK password is similar to OPERATOR except it enables the CONFIG key lock and disables the TUNE key lock. Use this to access all functions that require tuning access. The CFGLCK password is also similar to OPERATOR except it disables the CONFIG key lock and enables the TUNE key lock. Use this to access all functions that require configuration access.

4. Press ENTER. The user is prompted if an invalid password was entered.

The top field in the keyboard status block at the lower left corner of the window identifies the index number of the currently logged in user. If the log-in is accepted, the field updates to the identification number of the user logging in. If the security level does not give access to the current window, the user cannot log in. A password log-in is recorded in the event log (or operator actions log).

A new password can be entered at any time without having to log out the current user. Repeat the log-in procedure to enter the password of another user. An option of password security can be enabled to automatically log a user off the system and log in a default user when the OIS application is inactive for a certain period of time.

A password log-in remains in effect until changed by entering another password or by an automatic log-out. To prevent unauthorized personnel from using the current access rights, log out before leaving.

Log-Out

A log-out cancels the current password access rights and logs in the default user. To log out:

1. Press COM'D LINE MENU.

2. Press G to select the *LOG OUT* option. As soon as it is selected, the user index number in the keyboard status block updates to the default user's index number.

> **NOTE:** If the OIS application is reset, it comes back on-line with the access rights of the last entered user. The default user is logged in after a complete shutdown and power up.

NETWORK DEFINITIONS

The console requires access to the DECnet network to perform some of its functions. The access is used to:

- Direct windows to auxiliary consoles or other nodes on the network.
- Send logs to either shared or private printers connected to the network. Printers connect either through a terminal server or an auxiliary console.
- Send a screen copy to a printer connected to the network. Printers used for printing screens connect through a terminal server only, not an auxiliary console.
- Archive logs and events to an open access system.
- Display trended data collected by an open access system.

Any system connected to the network, whether it is an auxiliary console or a terminal server connecting printers, for example, is a network node. Each node on the network has a unique name to identify it to other nodes. The configuration procedures on the console that enable the previous network capabilities require a node name to locate a specific network node. Before a node name can be identified in a configuration, it must first be identified in the console network database.

The SYSTEM account is used to identify node names for those nodes the console is to access. This is normally performed during initial software installation. The procedures associate a node name to the hardware address of a network system. The SYSTEM account also allows viewing a list of current node names known by the console and renaming, adding, and removing node names. This can be performed at any time. Refer to the ***File Utilities*** instruction for further explanation and for the procedures (Table 1-2 lists instruction numbers).

Window Management

Auxiliary consoles do not have direct access to the INFI-NET or Plant Loop communication highway. Because of this, they use and operate through windows generated by a main console. Window management configuration is required to direct windows to auxiliary consoles (or other network nodes). A node name is used to direct a window to a specific node. A window can be sent using either DECnet or TCP/IP transport protocol. This is selectable on a per window basis.

Keyboard Definitions

An auxiliary console uses an IIMKM02 keyboard interface to connect an operator keyboard and an auxiliary engineering keyboard. Other types of nodes connect a keyboard directly without using an IIMKM02 interface. Performing keyboard definitions defines the keyboard type for the main console and those auxiliary consoles that receive windows from the main console. A node name is entered during this procedure to identify the specific auxiliary console to which a keyboard is assigned.

Open Access System

An open access system operates as a system node on the network with direct connection to the INFI-NET or Plant Loop communication highway. It therefore has its own unique node name. This node name is used in those functions that connect to the open access system.

SECTION 3 - SYSTEM

INTRODUCTION

This section explains the console configurations categorized as system configurations. System configurations affect the overall console operation:

- Define general parameters that affect operation and enable supported peripherals. General parameters configuration must be performed ***before*** any other configuration or before any console operation.
- Set up console processing responsibilities. This is performed to maximize operation based on the intended application for the console.
- Define time and date format. The time and date appearing in the display title line and other functions conform to configured formats.

GENERAL PARAMETERS

Use the *General Parameters* option to define the console operating parameters. Figure 3-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *A General Parameters*

The last step of this procedure resets the OIS application. Press **ESC** before pressing any other keys to exit this page without having to reset. To define the fields:

1. Enter the appropriate data into each field of the page. Table 3-1 explains the fields. Refer to this table when entering data. Press **ENTER** after each change.

The following message appears after **ENTER** is pressed or after moving from a changed field:

Do Not Restart the OIS While It Is Processing

Before further input or resetting the OIS application, wait for the message:

Restart System, Updates Have Been Done

2. After the restart message appears and after all changes have been made, reset the OIS application by using the procedures given in ***Reset*** in Section 2.

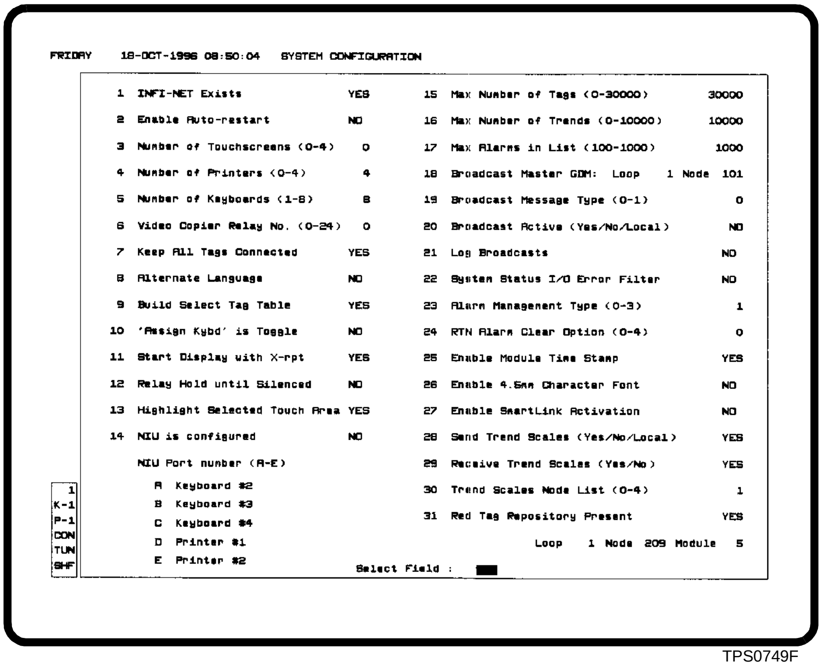


Figure 3-1. General Parameters Page

Table 3-1. General Parameters Page Fields

Field	Valid Entry	Purpose
INFI-NET exists	YES/NO	Type of communication highway on which the console is operating. YES = INFI-NET system. NO = Plant Loop system.
Enable auto - restart	YES/NO	Not applicable for this console. NO = default; leave at default.
Number of touch screens	0	Touch screen is supported; however, this field does not affect touch screen. Touch screen is enabled and set up outside the OIS application using the touch screen setup utility. Refer to the discussion on touch screen in the ***Operation*** instruction for an explanation. 0 = default; leave at default.
Number of printers	0 - 4	Number of printers that can be accessed for logging. Up to four printers can be accessed. Printers connect to the network through terminal servers or auxiliary consoles. 0 = default; no printers connected. 1 to 4 = one to four printers.

Table 3-1. General Parameters Page Fields (continued)

Field	Valid Entry	Purpose
Number of keyboards	1 - 8	Number of keyboards supported by the main console including itself and any auxiliary consoles. Refer to ***KEYBOARD DEFINITIONS*** in this section for the procedures to define the following for each keyboard: *Keyboard Type* *Trackball* *Number of ADP Panels* *X Server Node*
Video copier relay number	0 - 24	Not applicable for this console. 0 = default; leave at default.
Keep all tags connected	YES/NO	Determines when exception reports are monitored, either continuously or only when required. YES = monitor exception reports continuously for all tags. Provides a continuous update of database values. NO = monitor exception reports for tags currently displayed and for tags that data is currently being collected for only. **NOTE:** Must be set to YES in order for @aGlance/IT commands to access data on the console.
Alternate language	YES/NO	YES = enable alternate language character entry and options. NO = default; disable alternate language. Refer to Section 21 for further explanation.
Build select tag table	YES/NO	A tag table is used to automatically activate control for a tag selected with **TUNE**, **DETAILS**, or **OP PARAMS**. YES = enable building a tag table. NO = default; disable building a tag table, which disables automatic control activation.
'Assign kybd' is toggle	YES/NO	Configures **SWITCH CRT**. YES = use when operating with two keyboards. The key toggles between two windows instead of requiring a window number to be entered. NO = use when operating with three or more keyboards. After pressing **SWITCH CRT**, a prompt for window number appears.
Start display with exception report	YES/NO	Determines how values, states, and alarm indications for tags are presented during initial display call-up. YES = wait for a new exception report before displaying data. Dynamic fields for a tag remain blank until a new exception report is received for the tag. NO = display any values, states, or alarms received in previous exception reports.

Table 3-1. General Parameters Page Fields (continued)

Field	Valid Entry	Purpose
Relay hold until silenced	YES/NO	YES = an alarm relay remains closed until **SILENCE** is pressed. NO = default; an alarm relay remains closed until the alarm condition driving the relay returns to normal or is acknowledged. **NOTE:** In either case, a relay opens when the duration set for the relay expires. Refer to ***ALARM RELAYS*** in Section 9 for further explanation.
NIU is configured	YES/NO	Not applicable for this console. NO = default; leave at default.
Maximum number of tags	0 - 10000 or 30000	Disk space allocation for the tag database. Up to 30,000 tags are supported. If using an IIMCP01 module, the maximum number of tags is 10,000; if using an IIMCP02 module, the maximum number is 30,000. Change the size in 2,500 increments up to 10,000 (e.g., 2,500, 5,000, 7,500, or 10,000). For a database size greater than 10,000, use 30,000. **NOTE:** Decreasing this number erases any tags in the database that have index numbers greater than the new maximum value. It is recommended to leave this field at its default even if the maximum number of tags is not required.
Maximum number of trends	0 - 10000	Disk space allocation for the trend database. This should be close to the current database requirements and can be increased later to allow for more trend definitions if necessary. Up to 10,000 trend definitions are supported. The maximum number of each trend type is 2,000 local one-minute standard trends, 5,000 remote trends, and 10,000 local display only trends. The maximum decreases with a combination of one-minute standard, 15-second standard, and enhanced trends. **NOTE:** Decreasing this number erases any trend definitions that have index numbers greater than the new maximum value.
Maximum alarms in list	100 - 1000	Maximum number of alarms saved for display in an alarm summary. A single alarm list is maintained and used for all alarm summaries. Up to 1,000 alarms can be saved. Refer to ***ALARM SUMMARY*** in Section 9 for further explanation.
Broadcast master GDM[1]	0 - 250	For tag broadcasting; address of an interface unit connecting a work station running the global database manager (GDM) to the INFI-NET loop. Required to broadcast tag changes made from the console and to receive broadcast tag lists (database) from a GDM work station. Several GDM work stations can connect to the loop at one time. This identifies the GDM work station maintaining the database for this console. Valid *Loop* and *Node* address entries are: 0 = no loop connection. Must be 0 for Plant Loop system. 1 to 250 = loop and node address. Related fields are: *Broadcast Message Type* *Broadcast Active* *Log Broadcasts*

Table 3-1. General Parameters Page Fields (continued)

Field	Valid Entry	Purpose
Broadcast message type	0 - 1	Not applicable for this console. 0 = default; leave at default.
Broadcast active[1]	YES/NO/ LOCAL	For tag broadcasting; configures tag broadcast abilities. YES = enable receiving single tag changes and a complete tag list from a GDM work station. Also enables broadcasting single tag changes made from this console to the GDM work station. NO = disable both receiving and sending of any tag broadcasts. This disables broadcasting completely. LOCAL = enable receiving single tag changes and a complete tag list from a GDM work station; disable broadcasting single tag changes to the GDM work station.
Log broadcasts	YES/NO	Not applicable for this console. NO = default; leave at default.
Highlight selected touch area	YES/NO	Configures touch point highlighting. Touch screen or the mouse or trackball are used to select touch points. Highlighting provides visual verification of selection. YES = enable touch point highlighting. NO = disable touch point highlighting.
System status I/O error filter[1]	YES/NO	Determines if the console makes an entry in the alarm summary list or the event log for a system status (*S*) alarm caused by a remote I/O error or local I/O error in a PCU module. YES = enable filtering of remote or local I/O status errors. A remote I/O error or local I/O error does not cause a system status alarm. NO = default; disable filtering of remote or local I/O status errors. Must be NO when ***not*** operating on INFI-NET system.
Alarm management type[2]	0 - 3	0[3] = default; an alarm list (i.e., alarm summary) is built in chronological order as alarms occur. Return-to-normals appear as new entries. 1 = an alarm list is built with same priority alarms grouped together and chronologically ordered within the group. Return-to-normals appear as new entries. 2[3] = same as option 0 but enables fixed position return-to-normal. 3 = same as option 1 but enables fixed position return-to-normal. Refer to ***ALARM SUMMARY*** in Section 9 for further explanation.
RTN alarm clear option	0 - 4	Not applicable for this console. 0 = default; leave at default.
Enable module time-stamp	YES/NO	YES = time-stamp in an exception report is received and processed. Time recorded in the event log for an event and in the alarm summary for an alarm is to the tenth of a second. NO = default; time-stamp in an exception report is not used. The time recorded in the event log for an event and in the alarm summary for an alarm is the time the exception report was read and is to the second. Refer to ***Module Time-Stamp*** in this section for further explanation.

Table 3-1. General Parameters Page Fields (continued)

Field	Valid Entry	Purpose
Enable 4.5 mm character font	YES/NO	YES = enable 4.5 millimeter characters. Character size 3 for text in a display results in 4.5 millimeter characters rather than triple height characters (372 display units). Also enables the 4.5 millimeter character height option in alarm summary format configuration. NO = default; disable 4.5 millimeter characters. Character size 3 produces triple height characters. **NOTE:** The 4.5 millimeter character font cannot be enabled if *Alternate Language* is enabled. Refer to ***4.5 Millimeter Characters*** in this section for further explanation.
Enable SmartLink Activation	YES/NO	Offered as an option. YES = activates automatic call-up and allows for manual call-up of SmartLink software from a function block. NO = disables both automatic and manual call-up of SmartLink software.
Send trend scales	YES/NO/ LOCAL	YES = enables global trend scale change. Trend scale changes are sent to all nodes in the node list defined for the trend scale setup. NO = default; disables the global trend scale change option. LOCAL = enables global trend scale change on the local node only.
Receive trend scales	YES/NO	YES = enables the global trend scale change receive option. NO = default; disables the global trend scale change receive option.
Trend scales node list	0 - 4	Node list for broadcasting global trend scale changes. The default is 0.
Red Tag Repository Present[4]	YES/NO	YES = establishes central red tag information repository at a Multi-Function Processor (MFP) module. Required to enter address of module where repository is to be established. Software checks to ensure address is within configured range. NO = No central red tag information repository established.

NOTES:

1. Fields are valid only when operating on INFI-NET communication highway.
2. When using options 0 and 1, a tag that returns to its normal state generates a new alarm line at the beginning of the alarm summary, which scrolls the last entry off a full summary. Options 2 and 3 are fixed position return-to-normal options. If used, a tag that returns to its normal state maintains its current position in the list, but changes to a specified return-to-normal color.
3. If *Alarm Management Type* is set to 0 or 2, all alarm entries in alarm summaries default to priority one (*P1*). Priority sorting is not implemented.
4. Requires a dedicated MFP with special Central Red Tag Information Repository software installed. Contact Elsag Bailey for more information (SRTR01).

Module Time-Stamp

When data is sent in an exception report, one of two things happens depending on console configuration. If module time-stamp is enabled, the module time-stamp in the exception report will be used in the event log and alarm summaries. The displayed, printed, and saved time for these two reports is to the tenth of a second. The event log and alarm summaries will be maintained in chronological order.

If module time-stamp is not enabled, the time the exception report was read is the time used in the event log and alarm summaries. The time is to the second. Some events or alarm changes may appear out of order in the event log and alarm summaries.

A controller module needs to be set up with time-stamp enabled to have process events and alarm changes recorded to the tenth of a second. Plant Loop system does not support this type of time stamping. This may cause a disparity in time-stamps if the same value is being reported from a node on a Plant Loop system and a node on an INFI-NET system.

ENABLING MODULE TIME-STAMP

To enable module time-stamp:

1. Follow the steps given in ***GENERAL PARAMETERS*** in this section.

2. Enter **YES** in the *Enable Module Time Stamp* field. The default is *NO* for disabled.

DEFINING THE TIME FORMAT

The default format for time displayed or printed in the event log and alarm summaries is:

hour:minute:second.tenth of a second

To modify the format:

1. Follow the steps given in ***TIME AND DATE FORMAT*** in this section.

2. Define the *Time Format with Tenths Resolution* field to create the desired format.

4.5 Millimeter Characters

The *4.5 mm Character Font* field on the general parameters page gives the ability to display characters in 4.5 millimeter character height. The characters can be used for static text or dynamic text in any display. When enabled, any text using character size 3 (372 display units) appear in 4.5 millimeter characters. An alarm summary with a maximum of 24 alarm lines can also be created to use the 4.5 millimeter characters.

ENABLING 4.5 MILLIMETER CHARACTERS

To enable 4.5 millimeter characters:

1. Follow the steps given in ***GENERAL PARAMETERS*** in this section.

2. Enter **YES** in the *4.5 mm Character Font* field. The default is *NO* for disabled.

USING IN A DISPLAY

Specify a character size of 3 to use the 4.5 millimeter characters in a display when creating the display with the graphical display configuration (GDC). In an escape command, this equates to a character size of 372.

To create a 24-line alarm summary using 4.5 millimeter characters:

1. Create a line format using a line option of 2 and a character height of 4.5 millimeter. Refer to ***Defining a Line Format*** in Section 9 for the procedures.

2. Edit the display containing the summary to use the line format previously created.

Use the *Operator Display's Faceplates* option to select this line format to be used in operator configurable displays. Refer to Section 14 for the procedures.

Global Trend Scale Change Broadcast

The general parameters *Send Trend Scales* and *Receive Trend Scales* fields enable or disable global trend scale change broadcasting. When trend scale change is enabled in local mode the high and low trend scale limits are saved in the ***.DU*** file for the console. The changes are not made to other active displays on the console but next time the display is opened the scale is updated to the new values. An event log entry is created when a change is made to a local display file.

When trend scale change is enabled in global mode, the high and low trend scale values are broadcast on INFI-NET or Plant Loop. The request is sent to the node list configured in *Trend Scales Node List* field on the general parameters page. The broadcast identifies the trend by the ***.DU*** file name, trend element number, trace number and trend index.

To receive a global trend scale broadcast the option must be enabled on the general parameters page. When a trend scale change request is received an event log entry is generated and the request is processed. The changes are made to the display file on the console but active displays are not changed until they are recalled. An event log entry is generated to record that the trend scales operation was performed.

The trend scale change does not apply to operator assignable trend displays or to tuning displays. The option is only available for distributed and enhanced trend displays which allow changes to the high and low scale limits from the display.

To enable trend scale change broadcasting:

1. Follow the steps given in ***GENERAL PARAMETERS*** in this section.

2. In the *Send Trend Scales* enter **YES** for global or **LOCAL**.

3. In the *Receive Trend Scales* enter **YES** to receive trend scale change broadcasts.

4. In the *Trend Scales Node List* enter the node list number to use for broadcasting trend scale changes. A valid entry is from 0 to 4.

Refer to ***Defining a Node List*** in Section 9 for the procedures to define a node list.

KEYBOARD DEFINITIONS

Each console (main and auxiliary) has a single keyboard interface module (IIMKM02). This module controls the keyboards (operator and auxiliary), annunciator display panels, and touch screen for the console. Specifically, the module controls the keyboards connected to the KEYBOARD and AUX KBD ports at the keyboard interface panel. To enable touch screen, a touch screen controller card that mounts on the IIMKM02 module is required.

The *Keyboard Definitions* option:

- Identifies the type of keyboard interface being used for a keyboard. The type is either IIMKM02 keyboard interface (MKM2) or X keyboard interface (XKBD).

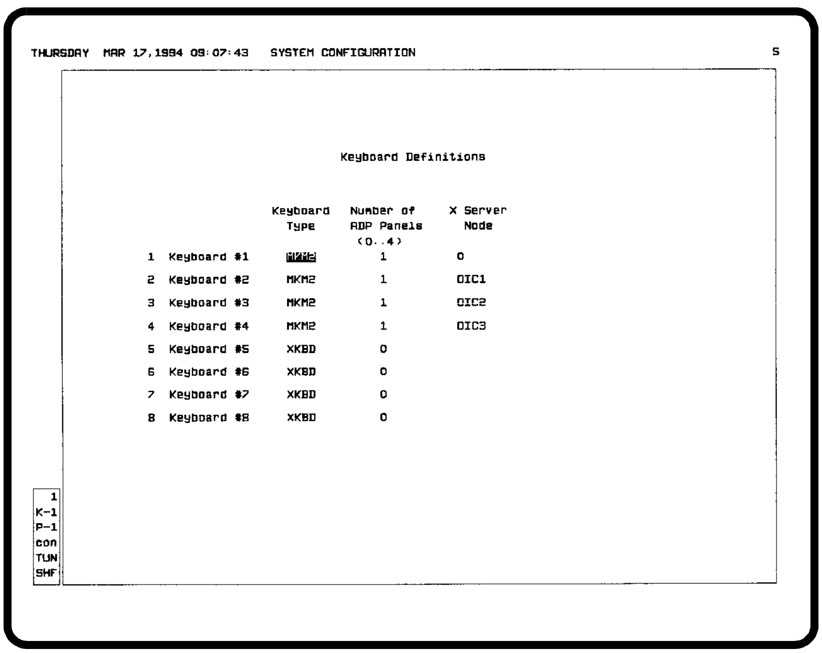
- Enables and identifies the number of annunciator display panels being controlled by the keyboard interface. An IIMKM02 module is required to support annunciator display panels (ADP).

- Identifies the node name of the console where the IIMKM02 keyboard interface resides if used.

It is recommended that a keyboard be associated with a window during definition. For example, if an auxiliary console uses keyboard two, then window two should also appear on that console. This is not a requirement, however. A window can be initially directed or redirected to any node at any time regardless of keyboard definition.

Use the *Keyboard Definitions* option to define keyboards. Figure 3-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *H Keyboard Definitions*



TPS0750B

Figure 3-2. Keyboard Definitions Page

The last step of this procedure resets the OIS application. Press **ESC** before pressing any other keys to exit this page without having to reset. To define the fields:

1. Enter the appropriate data into each field of the page. Table 3-2 explains the fields. Refer to this table when entering data. Press **ENTER** after each change.

The following message appears after **ENTER** is pressed or after moving from a changed field:

Update of Files In Progress, Please Wait

Before further input or resetting the OIS application, wait for the message:

Restart System, Updates Have Been Made

2. After the restart message appears and after all changes have been made, reset the OIS application by using the procedures given in ***Reset*** in Section 2.

Table 3-2. Keyboard Definitions Page Fields

Field	Purpose
Keyboard type	Type of keyboard interface being used for the keyboard. By default, keyboard numbers correspond to window numbers (i.e., keyboard one belongs to window one). A valid entry is: MKM2 = IIMKM02 module is being used to interface the keyboard (i.e., keyboard connects at the KEYBOARD port or AUX KBD port). Use this type if the keyboard connects to a main console (i.e., this console) or if it connects to an auxiliary console of the main console. Refer to ***MKM2 Keyboard Type*** in this section for further explanation. XKBD = use this type if the keyboard does not connect through an IIMKM02 module. Or, use this type if the window associated with the keyboard is being sent to another main console or to an auxiliary console tied to another main console. Refer to ***XKBD Keyboard Type*** in this section for further explanation. **NOTE**: Define an *X Server Node* if the type is MKM2.
Number of ADP panels	Number of annunciator display panels (ADP) supported by the keyboard interface of the keyboard. The keyboard type must be MKM2 to define ADP panels. Each keyboard interface supports four panels. 1 = default; one panel. 2 to 4 = two to four panels.
X server node	Node name of a main or auxiliary console to which the keyboard connects. Required for an MKM2 type keyboard. 0 = use zero if the keyboard connects to this main console. *nodename* = enter the name of the auxiliary console to which the keyboard connects. The console checks the validity of the name after pressing **ENTER**.

NOTE: For proper keyboard and ADP operation, the keyboard definitions must agree with the window assignments (*X Server Node*) made through *X Device Definition*.

MKM2 Keyboard Type

To define a keyboard as an MKM2 type requires the keyboard be connected through an IIMKM02 keyboard interface module. A main console and auxiliary console both support an IIMKM02 module. The module connects and enables the keyboards (operator and auxiliary type), any annunciator display panels, and touch screen.

Both the operator keyboard and auxiliary engineering keyboard can be assigned to any window on their home screen by setting input focus with the mouse. Both keyboards also can be assigned to the eight windows generated by the console to which it is connected. Input focus is automatically given to a window the keyboard is switched to when using **SWITCH CRT**.

XKBD Keyboard Type

The XKBD keyboard type is for a keyboard that connects to a terminal without use of an IIMKM02 keyboard interface module. In this case, the keyboard still maintains all the operator keyboard functions, but can only be assigned to windows that appear on the screen of the terminal. The keyboard is assigned to a window by setting input focus with the mouse.

The XKBD keyboard type is also for any other node not tied to the main console that is to receive a window from the main console. For example, a personal computer, a work station, or another main console is defined with an XKBD keyboard type. An auxiliary console of another main console that is to receive a window is also defined with an XKBD keyboard type.

OPERATOR CONTROL OPTIONS

Use the Operator Control Options menu selections to define the available console control options. Figure 3-3 shows the page used. First press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
- *E System*
  - *A System*
    - *J Operator Control Options*

Press **ESC** before pressing any other keys to exit this page without having to reset. A control option is activated by entering a yes response into the appropriate field and pressing the **ENTER** key after each change. Table 3-3 explains the fields. Refer to this table when activating the control options.

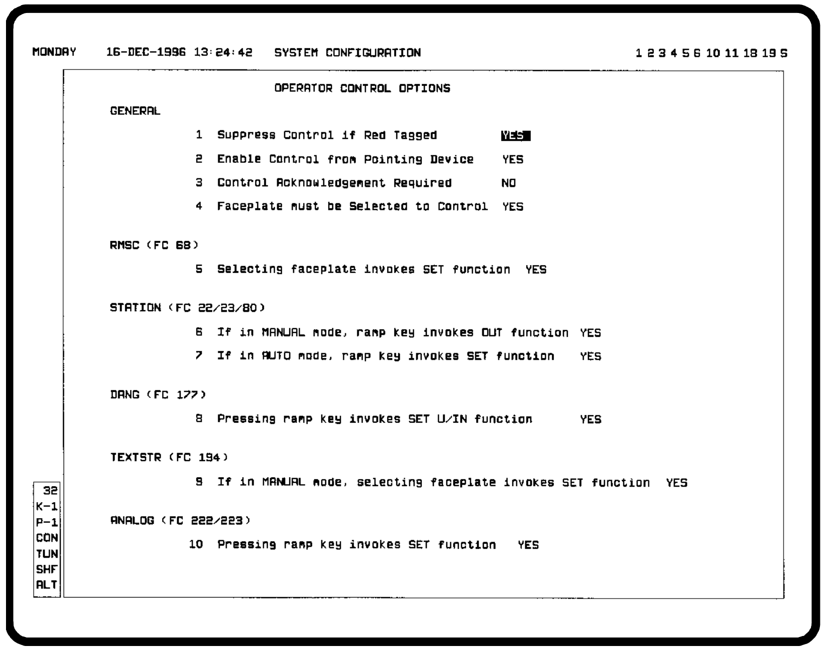


Figure 3-3. Operator Control Options Page

Table 3-3. Operator Control Options Page Fields

Field	Valid Entry	Purpose
Suppress control if red tagged	YES/NO	YES = default; disable control from the console for any red tagged block. NO = enable control from the console even if a function block is red tagged. Red tag status is informational only.
Enable control from pointing device	YES/NO	YES = default; enable touch pads. NO = disable touch pads. Refer to the discussion on touch pads in the ***Operation*** instruction for an explanation of touch pad operation (Table 1-2 lists instruction numbers).
Control acknowledgment required	YES/NO	Determines if an acknowledgment is required before a control action is performed. Affects the following tags: DADIG MSDD RMSC DANG RCM STATION DD RMCB TEXTSTR YES = **ENTER** must be pressed to acknowledge a control action. NO = default; no acknowledgment required.
Faceplate must be selected to control[1]	YES/NO	Prevents faceplate selection and push button operations from occurring at the same time. YES = default; faceplate must first be selected, via keyboard or touch-area, before push buttons will respond. NO = pressing push button on faceplate both selects faceplate and initiates push button operation.

Table 3-3. Operator Control Options Page Fields (continued)

Field	Valid Entry	Purpose
Selecting faceplate invokes SET function	YES/NO	YES = activates the TARGET input field, allowing the constant value to be changed, when the RMSC function block is selected. NO = default; requires operator to press the **SET** key in order to activate the TARGET input field.
If in MANUAL mode, ramp key invokes OUT function	YES/NO	YES = activates the TRGT OUT input field, allowing the control output value of a STATION element in the MANUAL mode to be changed, when a STATION faceplate is selected. NO = default; requires operator to press the **OUT** key in order to activate the TRGT OUT input field.
If in AUTO mode, ramp key invokes SET function	YES/NO	YES = activates the TRGT RAT input field, allowing the ratio index of a STATION element in the AUTOMATIC mode to be changed, when a STATION faceplate is selected. NO = default; requires operator to press the **SET** key in order to activate the TRGT RAT input field.
Pressing ramp key invokes SET U/IN function	YES/NO	YES = activates the SET U/IN input field, allowing the user-insert value of a PCU module to be changed, when a DANG faceplate is selected. NO = default; requires operator to press the **SET** key in order to activate the SET U/IN input field.
If in MANUAL mode, selecting faceplate invokes SET function	YES/NO	YES = activates the TGT TEXT input field when the user- defined data export block is in manual and the TEXTSTR element faceplate is selected. NO = default; requires operator to press the **SET** key in order to activate the TGT TEXT input field.
Pressing ramp key invokes SET function	YES/NO	YES = activates the TARGET input field, allowing the set point value of a PCU module to be changed, when a ANALOG control faceplate is selected. NO = default; requires operator to press the **SET** key in order to activate the TARGET input field.

NOTES:

1. This control option is not valid unless the OIS push-button display escape (ei 107,115) is used when building the display. Refer to the ***Display Builder Reference*** instruction for more information. (Table 1-2 lists instruction numbers.)

CIU TASK PROCESSING

During operation, the console performs its processing responsibilities in a specific order every cycle. A cycle is one second. CIU task processing configuration adjusts the number of exception reports the console is to process during a cycle. It also assigns a priority to the remaining console processing responsibilities.

> **NOTE:** Normally, the defaults for the *CIU Task Processing* option are adequate for proper console operation. It may be beneficial to become fully familiar with this function before performing initial console configuration to maximize processing capabilities.

The ability to set the number of exception reports to process during a cycle allows adjusting the console operation for its intended use. If the main responsibility of the console, for example, is to present real-time data for operations, then the number of exception reports to process should remain high. A lower number may be desired for a console used for less critical tasks.

The tasks the console is responsible for processing during a cycle include:

- Control commands initiated by the console.
- Exception reports received from the process.
- Standard and enhanced trend data.
- Operator assignable trends (fast trends).
- Data for MFC data source XY plots.
- Sequence of events log data.
- PCU configuration operations.
- PCU management operations.
- Module firmware reports.

Of these responsibilities, control commands are always processed first, then exception reports. After that, the order of processing depends on this configuration.

When all tasks have been completed or the one-second time elapses, the processing order is restarted. This sequence continues to repeat under normal operating conditions. Due to the limited time in a cycle, some tasks may not be executed in the current cycle. The processing may be delayed until the next or subsequent cycles depending on the system load.

If all processing requirements of the cycle are completed and it is determined that there is sufficient time remaining in the cycle, processing of control commands and exception reports will continue until the one-second time elapses. The *Max number of XR's to be processed each second* set during configuration still applies, however. No more than this number of exception reports will be processed each cycle even if there is processing time remaining in the cycle.

With a configuration that requires a large number of exception reports to be processed, it is possible that some lower priority tasks will execute in later cycles. If the number of exception reports to process is too large, the processing of exception reports might not be completed during a single cycle. For these reasons, care must be taken when deciding the number of exception reports to process and the priority of each remaining task.

Use the *CIU Task Processing* option to define the console task processing responsibilities. Figure 3-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

→ *E System*

→ *L CIU Task Processing*

Task Processing Settings

The settings made with the *CIU Task Processing* option are:

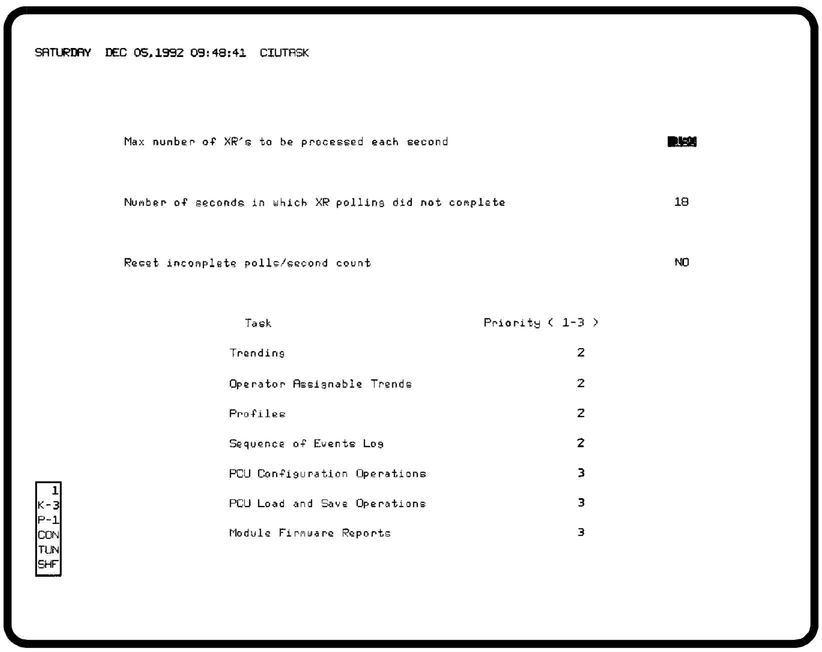


Figure 3-4. CIU Task Processing Page

Max number of XR's to be processed each second - determines the maximum number of exception reports to be processed during a cycle. The range is from 1 to 1,600; the default is 150.

When the console is equipped with a serial interface unit (IIMCP01 module), a good starting point for the field is the default of 150. When the console has a SCSI interface unit (IIMCP02 module), a value of 600 exception reports is recommended to start.

Number of seconds in which XR polling did not complete - gives a count of the number of times one of two events occurred. The count increments each time no exception report polling occurred during a cycle. This can happen, for example, if during the end of the current cycle the console has polled for some type of data other than exception report data and the processing of that data does not complete until the end of the next cycle.

The count also increments each time not all exception reports are returned to the console during a cycle. This can occur when the buffer containing exception report data fills, which will cause the processing of any waiting data to extend into the next cycle. For example, if set to process 100 exception reports and the console buffer fills after 50 reports, the processing of the remaining 50 reports will be completed during the next cycle. This assumes that after the first 50 reports, there are 50 more waiting. If the 50 are processed and there are no more reports, then the count does not increment.

Look at the *not complete* field to determine if the console is requesting a large number of exception reports from its communications interface unit. A number in this field does ***not*** indicate data loss, only that there is a large amount of data remaining to be processed.

Reset incomplete polls/second count - resets the incomplete poll counter to zero. The field defaults to *NO* before and after initiating a reset. Reset the counter after making adjustments to this page.

Task - provides a list of console processing responsibilities.

Priority (1-3) - indicates and allows changing the priority associated with a task.

To define the console task processing responsibilities:

1. In the *Max number of XR's to be processed each second* field, enter a number from 0 to 1600. A maximum of this many exception reports will be processed during a single cycle.

Refer to the *Number of seconds in which XR polling did not complete* field when adjusting the maximum number of exception reports. The *Max number of XR's to be processed each second* field may need to be adjusted again later if changes to the system are made.

2. In the *Priority* column of each task listed, enter a priority from 1 to 3. A task having a priority of one executes before a priority two task and a priority two task executes before a priority three task.

For tasks having the same priority level, processing occurs on a first-in, first-out basis except in the case of trending. If there are other tasks with the same priority as trending, they execute before trending.

3. Press **ENTER**.

Incomplete Poll Counter

During approximately the first ten minutes after the console comes on-line, the number of incomplete polls may increase since the entire database must be reported. As a result, the processing of exception report specifications takes priority over all other processing except control operations. After ten minutes, the processing should return to the order dictated by this configuration. It is suggested that the incomplete poll counter be reset after the console has been on-line for at least ten minutes.

To reset the incomplete poll counter:

1. In the *Reset incomplete polls/second count* field, enter **YES**.

2. Press **ENTER** to reset the counter to *0*.

TIME AND DATE FORMAT

The time and date presented by the console can be configured. All displays and logs that show a time and date conform to a defined format. The format for log outputs, summary displays, and even the title line can be defined.

The time and date can appear in any order desired. For example, a date can appear in conventional format as month-day-year or military format as day-month-year. The title line order can be time-date-weekday or date-weekday-time. The day of the week width can be adjusted to show a completely spelled out day, such as *MONDAY*, or limited to appear as only *MON*.

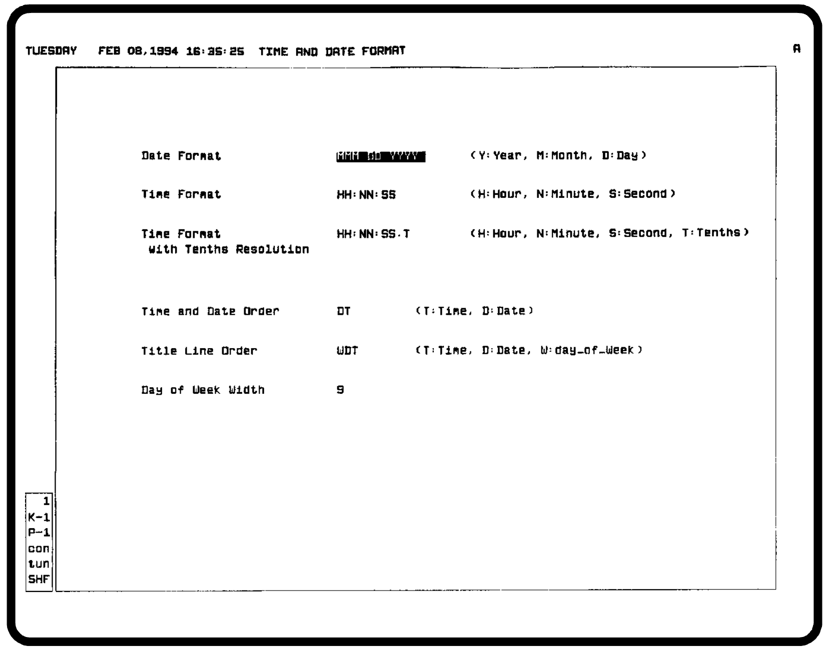
Use the *Time/Date Format* option to define the time and date format. Figure 3-5 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *G Time/Date Format*

To define the time and date formats:

1. Specify the date format in the *Date Format* field. This is the date format used in all functions. For example, to display the date as 28-FEB-96, enter **DD-MMM-YY**. To display the date as FEB 28, 1996, enter **MMM DD, YYYY**.

2. Specify the time format in the *Time Format* field. This is the time format used in all functions unless module time-stamp is



TPS0159B

Figure 3-5. Time and Date Format Page

enabled. For example, to display the time as 06:17:35, enter **HH:NN:SS**.

3. Specify the time format to use when module time-stamp is enabled in the *Time Format with Tenths Resolution* field. This is the format used in alarm summaries and the event log. For example, to display the time as 06:17:35.4, enter **HH:NN:SS.T**.

4. In the *Time and Date Order* field, specify the order of appearance for the time and date. This sets the order for all functions except the title line. For example, to have the date appear before the time as 28-FEB-96 06:17:35, enter **DT**.

5. Specify the title line time, date, and week day order of appearance in the *Title Line Order* field. For example, to have the title line appear in time, date, and week day order, enter **TDW**.

6. In the *Day of Week Width* field, specify the number of characters that are to appear at the title line for the week day (maximum ten). For example, to show a week day as a three-letter abbreviation, enter **3**. The display will show *MON* for Monday.

7. Press **ENTER**.

SECTION 4 - WINDOW MANAGEMENT

INTRODUCTION

The OIS application supports eight windows. These are X windows. A main console has the ability to both send and receive X windows using DECnet or TCP/IP transport protocol. A window can be sent to another node to give access to the OIS application running on the console.

The types of network nodes that can receive a window from the console include:

- Auxiliary consoles.
- Other main consoles.
- Work stations.
- Engineering work stations (personal computers) running X windows software.

An auxiliary console does not have direct access to the INFI-NET communication highway; therefore, the operation of an auxiliary console ties directly to a main console. A window must be directed to an auxiliary console from a main console to perform process control and monitoring from that console. If the OIS application is started up, shut down, or reset from a main console that is sending windows, the application also starts up, shuts down, or resets on the auxiliary consoles to which the main console is directing its windows. Conversely, the OIS application starts up, shuts down, or resets on the main console if it is started up, shut down, or reset from an auxiliary console. This is also true for any other type of node that is receiving a window from a main console.

WINDOW ASSIGNMENTS

Windows are sent to network nodes based on window assignments. A window assignment can be made in two different ways: X device definition and OISWIN account. A node name identifies the specific node that is to receive a window.

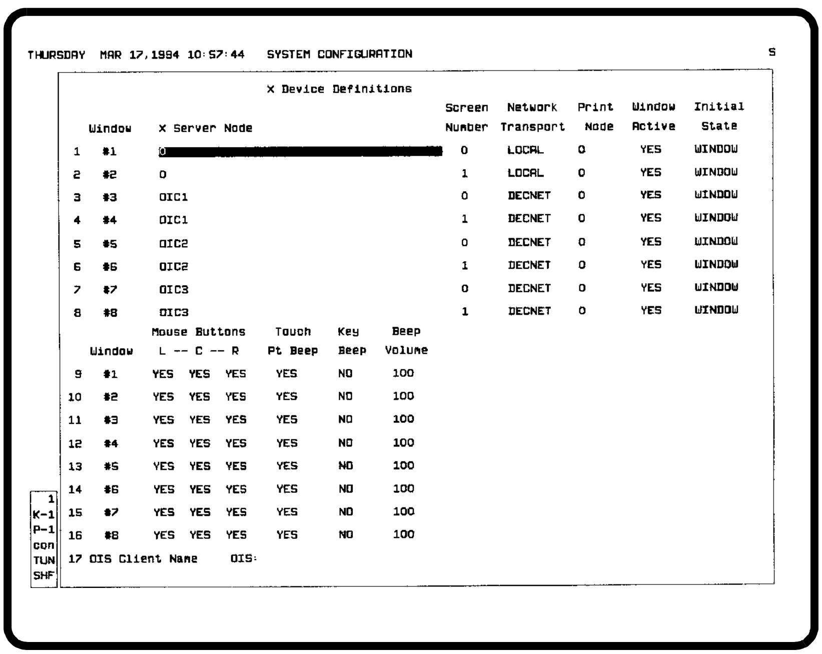
The *X Device Definition* option of the OIS application can be used to initially define window assignments or to redirect windows. Menu selections provide access to the X device definition page. The OISWIN account provides a remote window management capability. Logging into this account initiates an interactive program that can be used to initially define window assignments or to redirect windows. Window assignments can be changed at any time by using either function.

Access to a specific node on the network can fail for several reasons. For example, the node can be powered down or a hardware problem can exist. When a failure of a node is detected, the window being sent to the node is immediately deactivated. The X device definition page or OISWIN account interactive program displays the current status of each window. After a problem is corrected, sending a window can be reinitiated by toggling the status back to an active status (*YES*).

X DEVICE DEFINITION

Use the X Device Definition option to make window assignments. Figure 4-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
- *E System*
  - *A System*
    - *G X Device Definition*



THURSDAY MAR 17, 1994 10:57:44 SYSTEM CONFIGURATION S

X Device Definitions

	Window	X Server Node	Screen Number	Network Transport	Print Node	Window Active	Initial State
1	#1	0	0	LOCAL	0	YES	WINDOW
2	#2	0	1	LOCAL	0	YES	WINDOW
3	#3	OIC1	0	DECNET	0	YES	WINDOW
4	#4	OIC1	1	DECNET	0	YES	WINDOW
5	#5	OIC2	0	DECNET	0	YES	WINDOW
6	#6	OIC2	1	DECNET	0	YES	WINDOW
7	#7	OIC3	0	DECNET	0	YES	WINDOW
8	#8	OIC3	1	DECNET	0	YES	WINDOW

	Window	Mouse Buttons L	C	R	Touch Pt Beep	Key Beep	Beep Volume
9	#1	YES	YES	YES	YES	NO	100
10	#2	YES	YES	YES	YES	NO	100
11	#3	YES	YES	YES	YES	NO	100
12	#4	YES	YES	YES	YES	NO	100
13	#5	YES	YES	YES	YES	NO	100
14	#6	YES	YES	YES	YES	NO	100
15	#7	YES	YES	YES	YES	NO	100
16	#8	YES	YES	YES	YES	NO	100

17 OIS Client Name OIS:

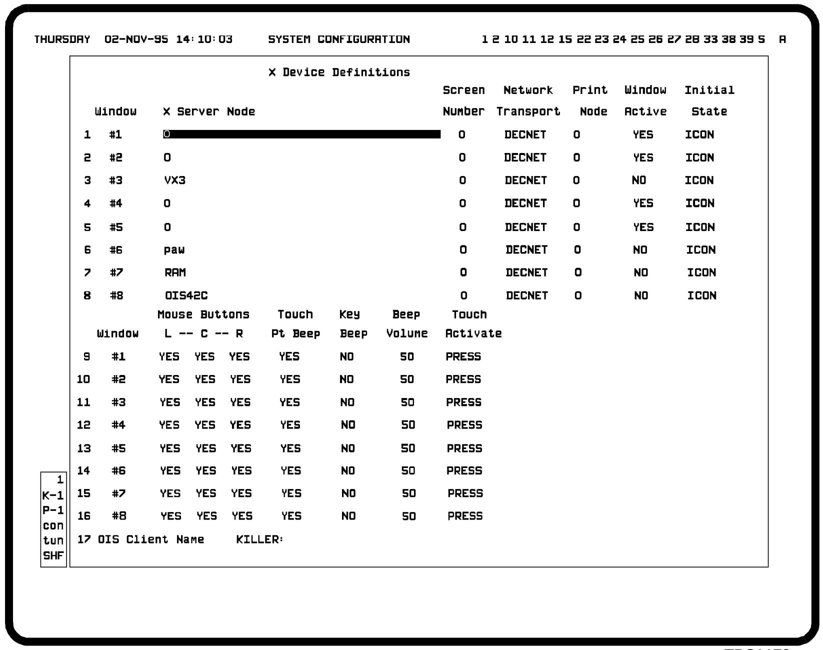
1
K-1
P-1
con
TUN
SHF

TPS0748B

Figure 4-1. X Device Definition Page

To define the fields of this page:

1. Enter the appropriate data into each field of the page. Table 4-1 explains the fields. Refer to this table when entering data.



TPS1178a

Figure 4-2. X Device Definition Page

2. After making all changes, press **ENTER**. If reassigning a window, it disappears from the network node that it was previously assigned to and appears on the new target node after a few seconds.

> **NOTE:** The window currently being used to make window assignments can be reassigned, but the window disappears after pressing **ENTER** terminating the session.

Some input fields on the X device definition page take effect immediately. Others take effect only after:

- OIS application reset.
- Reassigning the window whose attributes have been changed to another network node.
- Deactivating then reactivating the window.

Refer to Table 4-1 to determine which fields require these additional actions. If required, continue to the next step.

3. Either reset the OIS application or deactivate then reactivate the window being defined:

- Reset the OIS application by using the procedures given in ***Reset*** in Section 2.
- Enter **NO** in the *Window Active* field to deactivate the window, then enter **YES** to reactivate the window.

Table 4-1. X Device Definition Page Fields

Field	Description
X server node	Destination node. A valid entry is: 0 = use to direct the window to the local screen of a 40 series console only. This must be zero if using *LOCAL* as the *Window Network Transport*. *nodename*[1] = destination node name. Enter a name for an auxiliary console or any other type of node. A DECnet node name is limited to six characters. A TCP/IP node name can be up to 39 characters. **NOTE**: The *X Server Node* field on the keyboard definitions page must match this field.
Screen number	Targeted screen. A valid entry is: 0 = lower screen. Also use this when the node receiving the window does not support more than one screen. 1 = upper screen. If used for a node that is not set up for or capable of dual screens, the window will not appear on the destination node. This entry is not valid for an IIOIS40 PLUS since the upper screen is controlled by an auxiliary console. 2 to 9 = use for nodes that support more than two screens.
Network transport	DECNET = DECnet transport protocol. Use this when the window is to appear on an auxiliary console. DECNET can also be used for a window that is to appear locally. LOCAL = use this when the window is to appear locally. The *X Server Node* field must be zero to use this type. Local transport is more efficient than DECnet when operating a window locally. TCPIP = Internet standard transport protocol.
Print node	Print server node. This is the network node responsible for printer configuration, OIS application window capture, and directing the image to the print queue manager. Any console running the remote screen copy server node software can be specified as the *Print Node*. A valid entry is: 0 = this console. *nodename*[1] = name of a node that is to be the print server node.
Window active	Shows window status and allows manually changing the status. A valid entry is: YES = activate the window. NO = deactivate the window.
Initial state	Initial start-up state: Window or icon. This can be used to prevent a window from coming up over any currently open windows. A valid entry is WINDOW or ICON.
Mouse buttons[2]	Configures mouse buttons. This only affects the mouse when operating within the OIS application window. It specifies which mouse buttons can be used to activate a touch point of a display. A valid entry is: YES = enable button. NO = disable button. The center button (MB2) functions the same as the **ESC**.
Touch point beep	Configures a touch point tone used to identify a mouse or touch screen selection. A valid entry is: YES = enable tone. NO = disable tone.

Table 4-1. X Device Definition Page Fields (continued)

Field	Description
Key beep[2,3]	Configures a key tone. This only affects a non MKM keyboard[4]. A valid entry is: YES = enable tone for a non MKM keyboard. NO = disable tone for a non MKM keyboard. Leave at *NO* for an MKM keyboard.
Beep volume[2,3]	Tone volume for a non MKM keyboard[4]. The volume range is from -100 (no beep) to 100 (loudest). Leave this field at default for an MKM keyboard.
Touch Activation	Configures how touch screen commands are activated by operator. A valid entry is: PRESS = activates screen command when operator presses down. Default setting. RELEASE = activates screen command when operator presses down and lifts off.
OIS client name[2]	Text that appears in the title bar and icon for the window.

NOTES:

1. Refer to the ***File Utilities*** instruction for information on assigning node names to network nodes (Table 1-2 lists instruction numbers).
2. A change to the field requires an OIS application reset or the window to be deactivated then reactivated.
3. The key tone for a MKM keyboard (i.e., keyboard that connects to the KEYBOARD port or AUX KBD port) cannot be disabled.
4. A non MKM keyboard is any that is not connected to a console at the IIMKM02 keyboard interface. A keyboard that connects to the KEYBOARD port or AUX KBD port at the keyboard interface panel is an MKM keyboard.

The error message *Unable to Open Window as Requested* displays if a node assignment is invalid or if a node is not accessible. If a window assignment fails, check the node that is to receive the window and correct any problem. Then enter **YES** in the *Window Active* field to reinitiate sending the window.

REMOTE WINDOW ASSIGNMENT

Remote window assignment allows redirecting the windows of a main console from a remote network node. An interactive program (OISWIN account) gives the ability to define or change window assignments. It provides the capability to redirect a window to any node from any remote node even if that node does not have ready access to an OIS application window. The interactive program can be run from either a VT-series terminal or a terminal window.

To run the interactive program:

1. Log in from the remote network node. Refer to the ***File Utilities*** instruction for the procedures to log into an account from a VT-series terminal (Table 1-2 lists instruction numbers). Refer to the ***Operation*** instruction for the procedures to use the *Login Window* option to open a terminal window.

2. At the dollar sign ($) prompt, type:

   SET HOST *nodename* **[Return]**

where:

nodename	Name of a main console for which to view or change window assignments.

3. Log into the OISWIN account by entering a *Username* and *Password*. The default password is OISWIN. Press **Return** after each entry. This brings up a menu with the following choices:

1 - Display OIS window node assignments
2 - Activate OIS window
3 - Deactivate OIS window
4 - Redirect OIS window
5 - Exit

The OISWIN account is used only for remote window management. The interactive program run from the account allows viewing the current window assignments and activating, deactivating, or redirecting a window.

View To view the current window assignments, type:

1 **Return**

This shows the following for each window:

- Targeted X server node name.
- Targeted screen number (*0* = lower, *1* = upper).
- Transport protocol being used (*DECNET*, *LOCAL*, or *TCPIP*).
- Status (*Yes* = active, *No* = inactive).

Activate To activate a window:

1. Type: **2** **Return**

2. Enter the window number to activate.

3. Press **Return**. Use the *Display OIS window node assignments* option to verify the status.

Deactivate To deactivate a window:

1. Type: **3** **Return**

2. Enter the window number to deactivate.

3. Press **Return**. Use the *Display OIS window node assignments* option to verify the status.

Redirect To redirect a window to another node:

1. Type: **4** **Return**

2. Enter the window number to redirect.

3. Press **Return**.

4. Enter the node name to which to redirect the window.

5. Press **Return**.

6. Enter a target screen number as either 0 for lower or 1 for upper.

7. Press Return.

8. Enter the type of transport protocol to use as either **DECNET**, **LOCAL**, or **TCPIP**. Refer to Table 4-1 for a description of these protocol types.

9. Press Return.

Exit To exit the interactive, type:

5 Return

SECTION 5 - PASSWORD SECURITY

INTRODUCTION

This section explains the procedures to define a password security strategy for the console. This security strategy only affects operations within the OIS application. Establish security requirements prior to any console configuration. Appendix C contains worksheets that can be used to map out security before configuration or to record information defined during configuration. Information in these worksheets can be transferred directly to the console configurations.

NOTE: Security maintenance functions are provided to allow key personnel to assign access rights to security levels and to set up user IDs and passwords. These functions should be protected at a high level of security.

CONFIGURATION REQUIREMENTS

Password security can be used to give or deny access to console functions and displays. A password determines the functions or displays a user can access. The password must be logged in before access to functions will be granted.

All areas of operation, control, tuning, configuration, and alarm management can be secured under password protection. Security can be applied to an entire display and to individual tags. This gives the ability to organize displays and related tags into security areas. A display can be configured for unlimited access or access limited by password security. System security functions can be modified or changed at any time.

The CONFIG/TUNE key lock can be used along with passwords for additional security. Before defining password security, the key lock alone limits access to security configuration.

Complete password security configuration consists of:

- Security options configuration.
- Logical CRT definition.
- Security level configuration.
- User password definition.
- Display mask definition.
- Tag security level and group configuration.
- Log security level configuration.

The configurations listed represent all security features. Some may not apply to your specific security strategy. Password security should be considered before tag database, custom log, and SOE log configuration. Attributes affecting password security are defined during these configuration procedures.

DEFAULT PASSWORDS

The default console passwords are OPERATOR, and MAINT. Table 5-1 describes the purpose of each password. Table 5-2 gives the default configuration for these passwords.

Table 5-1. Default Passwords

Password	Access
OPERATOR	Gives access to all functions except security maintenance and enables the key lock.
MAINT	Denies access to all functions except security maintenance and disables the key lock.

NOTE:

1. Not available on IIOIS42 and IIOIS43 consoles. Used only for IS42 and IS43 consoles, which do not have a key lock.

Table 5-2. Default Configuration for Passwords

Password	User Index	Security Level	Security Tag Groups	Database	Display	Keyboard	System	Log	Time/Date	Real Trend	Archive	Pens	PCU	Alarm Inhibit	Red Tag[1]
OPERATOR	1	1	1111111111111111	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
MAINT	2	2	0000000000000000	N	N	N	Y	N	N	N	N	N	N	N	N

NOTE:

1. Red tag region is not enabled for the passwords because to properly implement red tagging each user should be assigned a unique red tag key.

DEFAULT SECURITY LEVELS

By default, the security level assigned to all system displays is level one. The default security level assigned to logs and tags when they are initially defined is also level one. Because of this and until their security level is changed to another, all system displays, logs, and tags are affected by a change to security level one. This needs to be considered when setting up password security, especially as it pertains to system displays.

Security level one should be used as the ***most unrestricted*** level. It is recommended that security level one be left as is, except for possibly key lock requirements, with all access rights enabled even if no user is assigned the security level. This will prevent undesirable restrictions to system displays, logs, and tags.

The security level of a log or tag can be changed at any time. It can be changed from its default to some other level to implement any desired restrictions. The security level of a system display cannot be changed unless the display exists as a ***DU*** file. In this case, the *Display Mask Definition* option can be used to change the security level of the display. Appendix E provides a listing of system displays and identifies the ***DU*** system displays. Security level and the methods to change it for a system display, log, or tag are discussed in more detail later in this section.

Example: If security level one is changed to disable tuning, any system display normally used for tuning will not allow tuning. This effectively disables any tuning from the console. For the tuning display, this can be rectified by changing the security level of the display to some other level that has tuning enabled. The security level can be changed since this display exists as a ***DU*** file. For the block details display and the tuning option of PCU configuration, however, tuning cannot be enabled since the security level of these system displays cannot be changed from their default.

A different method to restrict tuning (i.e., tuning display, block details display, and the tuning option) for a user is to assign any other security level except level one to the user and disable tuning access in that security level. This restricts the user rather than the display. The same method can be applied to other console functions.

Table 5-3 gives the configuration for predefined security levels.

Table 5-3. Default Configuration for Security Levels

Security Level[1]	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation	Configuration Key Lock	Tune Key Lock
1	Y	Y	Y	Y	Y	Y	N	Y	Y	Y	Y
2	Y	N	N	N	N	N	Y	Y	N	N	N

NOTE: 1. All logical CRTs are enabled (i.e., 1111111111111111).

SECURITY OPTIONS

The first step in setting up a password security is to define the operating parameters. The *Security Options Configuration* function sets up the operating parameters:

Number of Levels - identifies the number of security levels that password security uses.

Number of Passwords - sets the number of passwords that will be defined. This should correspond to the total number of users.

Automatic Logout - used to automatically log out an inactive user after a specified time period from five minutes to 12 hours.

Default User - identifies which password (i.e., user index number) is to be logged in after start-up and after an automatic log-out.

The *Default User* can be set up as desired. For example, the default user can be set up to permit complete access to the system limited only by the key lock or can be set up to deny all access until an operator logs in a valid password.

NOTE: If setting up a default user and the default user is to have restricted access, do ***not*** use security level one as the restricted level. Define another security level with the desired restrictions and assign this level to the default user.

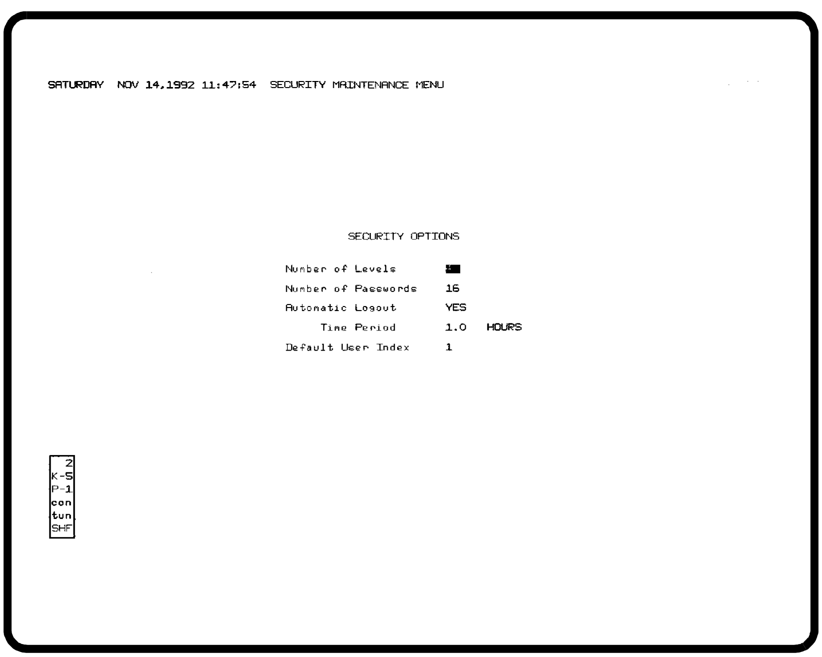
Use the *Security Options Configuration* option to define the operating parameters for password security. Figure 5-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

→ *G Security*

→ *A Security Options Configuration*

NOTE: Care must be taken when making changes in this function. Decreasing the number of security levels or user passwords deletes any levels or passwords with index numbers greater than the newly entered maximum. Changes made to the number of security levels could affect display, log, and tag security.



TPS0148B

Figure 5-1. Security Options Configuration Page

To define the fields of this page:

1. In the *Number of Levels* field, enter the number of security levels required. A valid entry is from 4 to 16.

A security level defines a general level of clearance. The number of security levels required depends on the security requirements of the plant. Refer to ***SECURITY LEVEL*** in this section for an explanation of security levels.

2. Enter the total number of passwords required in the *Number of Passwords* field. A valid entry is from 1 to 128. This should be large enough to accommodate all personnel requiring a password.

3. In the *Automatic Logout* field, enter **YES** to have a user automatically logged out. A log-out occurs after an inactive period of time determined by the *Time Period* fields. Enter **NO** to disable automatic log-out.

4. Enter an automatic log-out time in the *Time Period* fields. This requires two entries. The first entry is a number, the second is either MINUTES or HOURS. The minimum is five minutes, maximum 12 hours.

5. In the *Default User Index* field, enter the index number of the user that is to be logged in at start-up or after an automatic log-out. Each password defined in the console has a user index number.

6. Press **ENTER**.

CONSOLE AND LOGICAL CRT DEFINITION

Console definition and logical CRT definition are used to define windows as logical CRTs. After these configurations are set, access to console windows can be enabled or disabled. Refer to the example at the end of this section to see how console definition and logical CRT definition work together.

To locate a logical CRT, the security function first identifies a console by its address. After the specific console is identified, the function then identifies the specific console window the logical CRT assignment is referencing. For this type of console, more than likely a console definition is not required. This can be seen in the example that follows.

Example: Table 5-4 shows an example configuration. The example shows the configuration for a console generating eight windows. Windows one and two remain on the main console with one appearing on the lower screen and the other appearing on an upper screen. Windows three and four go to an auxiliary console with an upper and lower monitor. Windows five through eight each go to an auxiliary console with only one monitor.

Table 5-4. Example Console and Logical CRT Definition

Window		Console Definition			Logical CRT Definition		
No.	Screen	Console	Loop	Node	Logical CRT	Console	Physical CRT
1	1	1	1	100	1	1	1
2	2				2	1	2
3	1				3	1	1
4	2				4	1	2
5	1				5	1	1
6	1				6	1	1
7	1				7	1	1
8	1				8	1	1

NOTE: The console definition in this example could be defined as shown or left at its default of loop zero and node zero.

Defining a Console

Console definition is used to identify the hardware address of a console. The definition is used strictly by the console for password security to distinguish between consoles. The address is the console loop and node number. Console definition allows for up to eight definitions. After a console has been defined, its *Console #* is used in logical CRT definition. Only one console definition is required; however, all eight can be defined to allow flexibility in creating a security strategy common to several consoles.

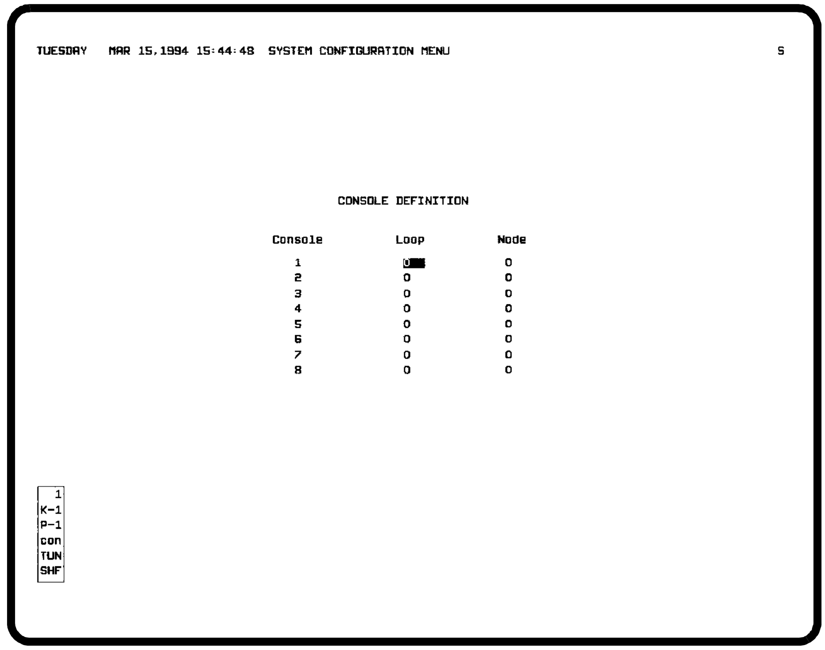
NOTES:

1. Optionally, the console definitions can be left at their defaults. Use a console number of one, in this case, when defining a logical CRT.

2. If used, console definition must be completed before logical CRT definition. Console definition establishes a console number required in logical CRT definition.

Use the *Console Definition* option to define consoles to be used in logical CRT definitions. Figure 5-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *J Console Definition*



TPS0152B

Figure 5-2. Console Definition Page

To define a console:

1. In the *Loop* field for a console, enter a loop number. A valid entry is from 0 to 250 for INFI-NET system or 0 and 1 for Plant Loop system. Enter **0** in the field to have the console use its own loop address automatically.

2. Enter a node number for the console in the *Node* field. A valid entry is from 0 to 250 for INFI-NET system or 0 to 63 for Plant Loop system. Enter **0** in the field to have the console use its own node address automatically.

3. Press **ENTER**.

Defining a Logical CRT

A logical CRT definition must be created to identify a window as a logical CRT. Access to a logical CRT is given or denied in the configuration of a security level. The ability to access a particular window for a user is based on logical CRT access rather than window number. Also, the ability to call up a display in a particular window is based on logical CRT access.

Normally the keyboard can be switched between the console windows by pressing **SWITCH CRT**. Password security can be used to limit this ability. If the operator does not have proper access based on logical CRT settings, switching to a restricted window will not be permitted.

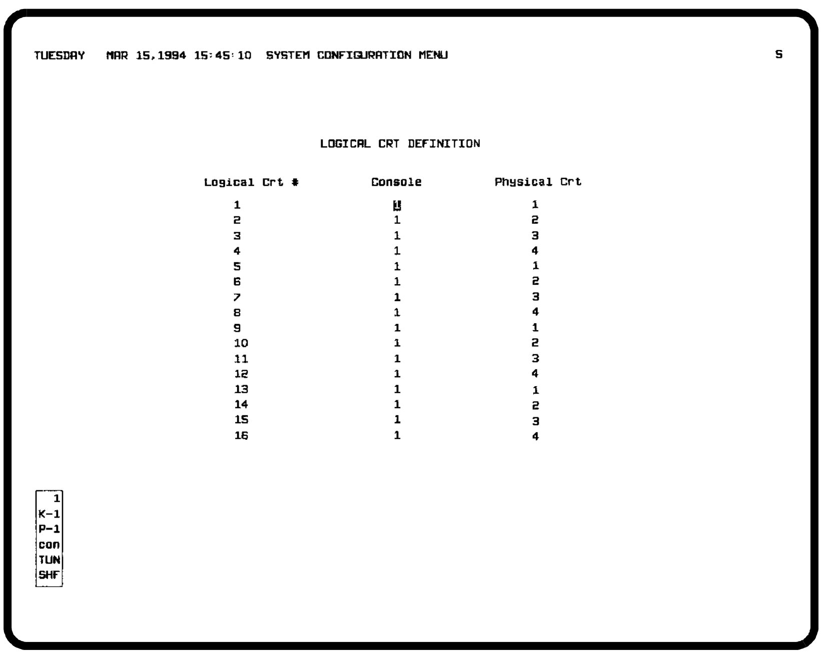
Also, normally a display can be called up in any window. If the security level of a display denies access to a particular window based on logical CRT settings, the display cannot be called to that window.

To define a logical CRT, a console must first be defined using the *Console Definition* option. Refer to ***Defining a Console*** in this section for an explanation of console definition. The console definitions are used internally. They are only intended to give flexibility when creating a security strategy. Using these definitions, a single configuration can be set up to be used on multiple consoles.

> **NOTE:** The number of logical CRT assignments required depends on the number of windows supported by the console.

Use the *Logical CRT Definition* option to create logical CRT definitions. Figure 5-3 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *I Logical CRT Definition*


TUESDAY MAR 15,1994 15:45:10 SYSTEM CONFIGURATION MENU S

LOGICAL CRT DEFINITION

Logical Crt #	Console	Physical Crt
1	1	1
2	1	2
3	1	3
4	1	4
5	1	1
6	1	2
7	1	3
8	1	4
9	1	1
10	1	2
11	1	3
12	1	4
13	1	1
14	1	2
15	1	3
16	1	4

1
K-1
P-1
con
TUN
SHF

TPS0153C

Figure 5-3. Logical CRT Definition Page

To define a logical CRT:

1. In the *Console* field for a logical CRT, enter a console number. A valid entry is from 1 to 8. This is the number of a console that was previously defined. Refer to ***Defining a Console*** in this section. If console definition was not performed or will not be used, enter **1** as the console number.

2. Enter a window number in the *Physical Crt* field. This defines which window the *Logical Crt #* is to identify. A valid entry is from 1 to 8 (i.e., 1 for window one, 2 for window two, etc.).

3. Press **ENTER**.

SECURITY LEVEL

Part of defining a password for a user is to give the user a security level. A security level defines a general level of clearance. For example, one security level could be set up to give full monitor, configuration, and control access. Another security level could give full access to all areas except password security configuration. And another level could allow only monitoring (viewing) all areas. Up to 16 security levels can be defined. Further restrictions can be placed on the user by setting up region accesses in the password.

Besides governing the abilities of a user, a security level for a tag or a log determines the actions that can be performed on the tag or log. For a display, a security level determines on which windows the display can be called and determines the actions that can be performed on the display once called.

> **NOTE:** The security level of the user and the security level of a display, log, or tag do not have to match for a user to access the display, log, or tag. The security level of a display, log, or tag determines the operations that can be performed on that display, log, or tag.

Refer to ***DISPLAY ACCESS***, ***Tag Access***, and ***Log Access*** later in this section to see how the security level of a user and the security level of a display, tag, or log work together.

Access Rights

Access rights of a security level define the first level of restriction for a user. Further restrictions are placed on the user with region access settings in the password. Refer to ***USER PASSWORD*** in this section for an explanation of functions affected by region accesses. Besides a user having a security level, each display, log, and tag in the console has a security level assigned. This allows defining access rights on a per display, log, and tag basis.

Some functions are only affected by the access rights of a security level and not by a region access of a password. These include:

- All password security configuration functions.
- Log status summary.
- Module firmware reports.
- Module problem reports.
- Operator configurable displays.
- Operating parameters.
- Tag summaries.

The access rights that can be enabled or disabled in a security level include:

Alarm Management

Determines alarm silence or acknowledge abilities. Silence and acknowledge are performed by pressing keyboard keys. Additionally, this access affects alarm operations performed by using the operating parameters page. A user having this access right can perform alarm management for a tag that appears on a display as long as the following allow it:

- Display security level.
- Tag security level.
- User security level.

Configuration

Controls the ability to enter into the console configuration functions. A user having this access right can configure any function permitted through specific region access rights. It protects functions such as defining tags and trends, defining alarm tones and relays, assigning annunciator display panel push-buttons, assigning keyboard function keys, etc. It does not, however, affect password security configuration.

Control

Determines process control abilities. Control is performed by using faceplate type device mimics. A user having this access right can perform process control for a tag that appears on a display as long the following allow it:

- Display security level.
- Tag security level.
- User security level.

Control through the following tags is affected by this access right:

DADIG	MSDD	RMSC
DD	RCM	STATION
DANG	RMCB	TEXTSTR

Diagnostic/Debug Terminal

Limits access to DDT functions. A user must have this access to use the DDT functions. Refer to the ***File Utilities*** instruction for further explanation (Table 1-2 lists instruction numbers).

Log Activation Controls the ability to activate or deactivate logs and cancel queued log prints by using the log status function.

Monitor Allows a user to view all areas of the system. A user having this access right can view any function permitted through specific region access rights and also all functions not protected by region access. No actions except to view can be performed.

Operation Determines access to general operations such as log by name, display by name, and operator configurable displays.

Security Maintenance Controls the ability to modify security levels and user passwords. This access should be limited to personnel responsible for console security. Specifically, the functions it affects are:

- Display mask definition.
- Security level configuration.
- Security options configuration.
- User password definition.

Tune Controls the ability to perform tuning and block details operations. Additionally, it allows the module operations available from the operating parameters, module problem reports, and module firmware reports pages.

Key Lock

Security level configuration can be used to either enable or disable key lock. The key lock position determines access to tuning and configuration functions. The console can be set up to require key lock along with the configuration and tuning access rights of password security or to not require key lock.

Defining a Security Level

After the *Number of Levels* has been set through *Security Options Configuration*, each security level must be defined separately. Security level configuration defines the access rights, key lock function, and logical CRT access for each level. Up to 16 security levels (minimum four) can be defined. Each user is assigned a security level during password definition.

NOTES:

1. Changes to a security level take effect only on the next log-in. If a change alters the password definition of the currently logged in user, log the user in again to put the changes into effect.

2. It is recommended that security level one not be changed. Use security level one as the most unrestricted level.

Use the *Security Level Configuration* option to define a security level. Figure 5-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *G Security*
→ *B Security Level Configuration*

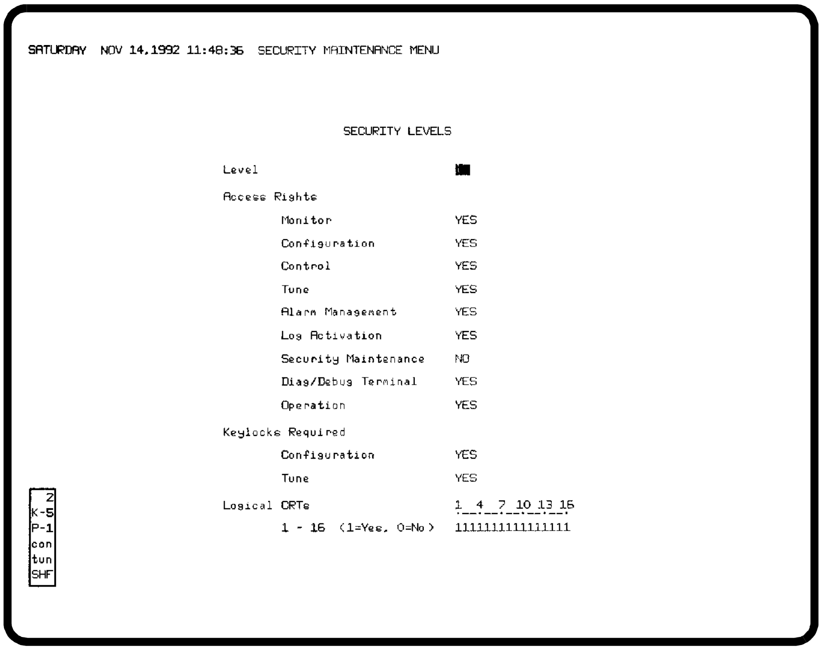


Figure 5-4. Security Level Configuration Page

To define a security level:

1. Select a level to define.

   a. In the *Level* field, enter the number of the security level to define. A valid entry is from 1 to 16. The maximum may actually be something less than 16 depending on the number of levels enabled during security options configuration.

   b. Press **ENTER**. The current settings for the level appear.

2. For each type of *Access Rights*, enter **YES** to grant access for this security level. Enter **NO** to deny access.

3. In the *Keylocks Required* fields, enter **YES** to enable the key lock. Enter **NO** to disable. If *YES,* security is based on both access rights and key lock. The key lock must be in the CONF position to perform configuration and TUNE to perform any tuning. If *NO*, security is predicated on access rights alone and the key lock position is ignored.

4. Up to 16 logical CRTs can be defined through logical CRT definition. After being defined, the *Logical CRTs* fields either give or deny access to windows defined as logical CRTs. Specifically, the fields determine which windows a user can log into and also determine which windows can be accessed using SWITCH CRT.

The fields under the *1..4..7..10..13..16* heading correspond to logical CRTs one through 16. For this security level to access a logical CRT, enter **1** in the field corresponding to the logical CRT; enter **0** to not allow access. For example, enter **1000000100000000** to allow access to only logical CRTs one and eight.

5. Press ENTER.

USER PASSWORD

A password both determines the operations that a user can perform and protects the access given to the user from unauthorized use. Up to 128 passwords can be defined.

To identify the currently logged in user, an index number appears in the keyboard status block in an OIS application window. Every password has an index number. The index number in the keyboard status block updates whenever a new user is logged in. A log-in is also recorded in the event log (or operator actions log). Refer to ***Log-In*** and ***Log-Out*** in Section 2 for the procedures.

> **NOTE:** The default passwords are MAINT for security maintenance access and OPERATOR, BOTHLOCK, CFGLCK, NOLOCK, and TUNLCK for normal operations.

Region Access

Access rights of a security level define the first level of restriction for a user. Further restrictions are placed on the user through region access settings in a password definition. The region accesses of a password work with the access rights of a security level to determine the actions a user can perform. A region directly affects the access rights enabled during security level configuration. For example, if a user is to be given tag configuration access, then the configuration access right must be enabled in the security level assigned to the user and the database region must be enabled in the password of the user.

The following information lists the specific menu options each category of region access affects. The submenus of an option are also affected by the region access. Notice that there are some cases where more than one region access is needed to access a function. Refer to Appendix B for a tree structured view of the menu system.

Alarm Inhibit Region

D Alarm Group Inhibit

Archive Region

F Archival Storage/Retrieval

Database Region

A Database

E System

B Alarm Management

G Print Database List

Display Region

B Display

Keyboard Region

D Keyboard

Log Region

C Logging

E System

C Printer Assignment
H Printer Color Maps

K Cancel Print

Pens Region

I Trend Pen Cluster

PCU Configuration Region

A PCU Configuration

Real Trend Region

E Operator Assignable Trends

Red Tag Region

G Red Tag Status

System Region

E System

F User Task

Time/Date Region

B Set Time and Date

NOTE: If a menu option is not listed here, it is protected under security level access rights only.

Defining a Password

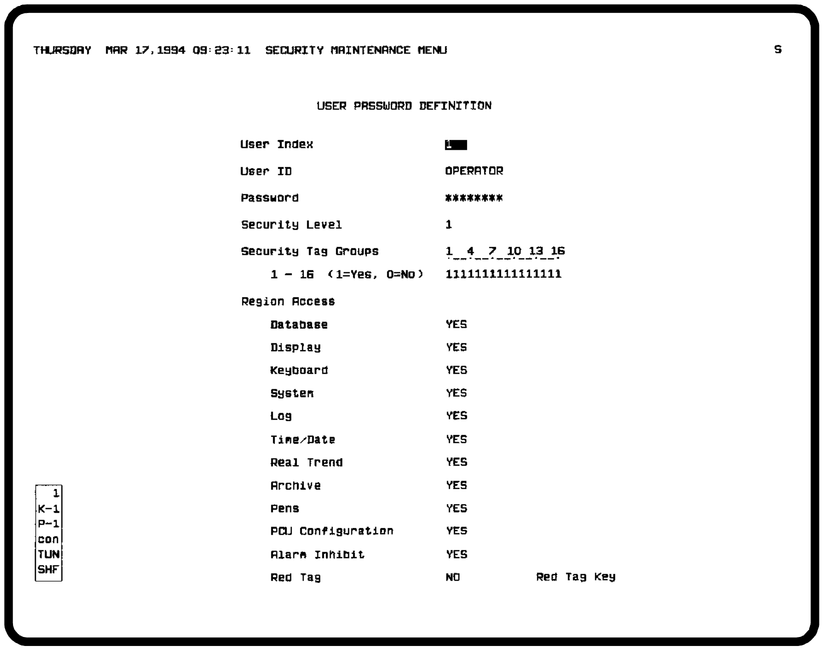
User password definition creates a password. The password definition:

- Defines a user ID. The user ID can be up to eight characters. ***The default password is the user ID.*** Passwords are normally hidden except when they are being entered.
- Defines a security level. The security level is one defined with security level configuration. Refer to ***SECURITY LEVEL*** in this section for further explanation.
- Enables or disables access to security tag groups. This determines the groups that the password gives access to if tag grouping is being used. Refer to ***TAG SECURITY LEVEL AND GROUP*** in this section for further explanation.
- Enables or disables access to functional regions. A functional region is a group of console configuration, utility, or operation options. Access to an entire region can be given or denied depending on password definition. Refer to ***Region Access*** in this section for further explanation.

NOTE: Changes to a user password definition take effect on the next log-in. If a change alters the password definition of the currently logged in user, log the user in again to put the changes into effect.

Use the *User Password Definitions* option to define a password. Figure 5-5 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *G Security*
→ *C User Password Definitions*



TPS0150C

Figure 5-5. User Password Definitions Page

To define a password:

1. Select a password to define.

   a. In the *User Index* field, enter the index number of a password to define. All passwords have an index number. A valid entry is from 1 to 128. The maximum may actually be something less than 128 depending on the number of passwords enabled during security options configuration.

   b. Press **ENTER**. The current settings for the password appear.

2. Enter a descriptive identifier in the *User ID* field. The field allows for eight characters. If a password has not been entered, this entry becomes the password by default.

3. Enter a password of up to eight characters in the *Password* field. The password will be visible during configuration but appears as ******** any other time the page is called.

4. Enter a security level in the *Security Level* field. This is one defined during security level configuration. A valid entry is from 1 to 16. The maximum may actually be something less than 16 depending on the number of security levels enabled during security options configuration.

5. A tag can be assigned to any of 16 different security groups during its definition. The *Security Tag Groups* fields either give or deny access to the security groups for this password.

The fields under the *1..4..7..10..13..16* heading correspond to security groups one through 16. For this user to access a group, enter **1** in the field corresponding to the group; enter **0** to not allow access. For example, enter **1000100100001000** to allow access to groups one, five, eight, and 13 only.

The default tag group set in the configuration of a tag is one; therefore, at least group one should be enabled even if tag grouping is not being used.

6. For each type of *Region Access*, enter **YES** to give access or **NO** to deny access. These fields should be consistent with the security level *Access Rights* assigned.

Region Access can be used to deny or grant access at a higher level than that of security level configuration (*Access Rights*). For example, a security level can give configuration access, but a password can deny access to database configuration.

7. A red tag key can be assigned to this user when enabling the *Red Tag* region. Enter up to three characters in the *Red Tag Key* field to be the red tag key assigned to this user. A valid entry can consist of the characters A through Z (uppercase only), 0 through 9, colon (:), dollar ($), period (.), and space. A key defined as three spaces, however, is not permitted.

A user that has a red tag key assigned can set or remove red tag status for a function block using only the assigned key. A user not assigned a red tag key has the ability to set and remove any red tag key even if the key is assigned to another user. Refer to the discussion on red tagging in the ***Operation*** instruction for more information (Table 1-2 lists instruction numbers).

8. Press **ENTER**.

PASSWORD ACCESS LEVEL

The access rights of a security level and the region accesses of a password work together to determine the access level for a user: No access, view only access, or full access. At a minimum, view only access is required to enter into a function.

Keep in mind that some functions are protected under access rights of a security level only and not by region accesses of a password. In this case, access level is determined by access rights alone. Refer to ***Access Rights*** in this section to identify those functions protected only under access rights of a security level.

Example: The following tables give examples of access level. Table 5-5 describes the conditions that determine access to configuration functions for a user.

Table 5-5. Password Access for Configuration

Access Level	Monitor Access Right[1]	Configuration Access Right[1]	Region Access[2]
No access	Yes/No	Yes/No	No
View only	Yes	No	Yes
Full access	Yes	Yes	Yes

NOTES:
1. Set through *Security Level Configuration.*
2. Set through *User Password Definition.*

Table 5-6 describes the conditions that determine access to red tagging. The password of the user first determines if the red tag status can be called for a given function block, then if a red tag can be set or removed for that block.

Table 5-6. Password Access for Red Tagging

Red Tag Access Level	Monitor Access Right	Configuration Access Right	Red Tag Region Access
No access	Yes/No	Yes/No	No
View only	Yes	No	Yes
Set/remove	Yes/No	Yes	Yes

DISPLAY ACCESS

Display access rights are set on a per display basis. Each display identified by its name can have a primary and secondary security level assigned. Two security level checks are made before a display is called to the window.

The first check tests to see if the display can be called to the current window. The logical CRT settings of the security level assigned to the display determine on which windows the display can be called. The primary security level of the display is tested first. If this test determines that the display cannot be called to the current window, the secondary level is then tested. If both tests fail, the display will not be called up.

Example:

1111 1111 0000 0000 - Security level logical CRT assignment
0100 0000 0000 0000 - Logical CRT ID of current window
0100 0000 0000 0000 - Test pass

1111 1111 0000 0000 - Security level logical CRT assignment
0000 0000 1000 0000 - Logical CRT ID of current window
0000 0000 0000 0000 - Test fail

Whichever security level test passes becomes the security level that is used in a second check made by the console. For example, if the primary security level test fails, but the secondary level passes, then the secondary level determines the actions that can be performed on the display.

The second check tests:

- Security level access rights of the display being called up.
- Security level key lock settings of the display to see if the key lock is enabled or disabled.
- Access rights of the display currently on the window.
- Access rights of the currently logged in user.

Table 5-7 shows some examples of the second test. Example one demonstrates a display call-up for a display with a security level of 1,4 with security level one passing the first test. Example two shows the same display with security level four passing the first test instead. Notice that the actions allowed are different even though in both cases a user with security level two is calling the display. The order in which the primary and secondary security levels are selected (1,4 versus 4,1) also has an affect on how a display operates.

> **NOTE:** Since the current display has an affect on the operations that can be performed from a called display, in some cases it may be necessary to call a display twice to enable the correct access rights.

Table 5-7. Example Display Security Level Check Results

Example	Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation
1	Called display (level 1)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	Current display (level 1)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 2)	Y	N	Y	Y	N	N	N	N	Y
	Actions allowed[1]	Y	N	Y	Y	N	N	N	N	Y
2	Called display (level 4)	Y	N	N	N	N	N	N	N	N
	Current display (level 1)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 2)	Y	N	Y	Y	N	N	N	N	Y
	Actions allowed[1]	Y	N	N	N	N	N	N	N	N

NOTE:

1. The resulting actions allowed are determined by the called display, current display, and user.

Notice that for the second check to pass and the display to be called, the same access right must be enabled for:

- Display being called.
- Display currently on the screen.
- User calling the display.

For example, if the display being called and the display currently on the window had only monitor access and the user had only diagnostic/debug access, the display would not be called. If, however, all three had monitor access, then the display would be called.

Security Level Assignment

Security level assignments for displays can be made either by using the *Display Mask Definition* option of password security or during display creation using a security mask (**sm**) escape command. The security level assigned to an operator configurable display is the same as the security level of the user that was logged in when the display was created.

If the security level for a display is set during its creation, display mask definition is not required. It is recommended that changes that are to be permanent be made to the security mask escape command in a display source file (***DT***). Refer to the ***Display Builder Reference*** instruction for the **sm** command format (Table 1-2 lists instruction numbers).

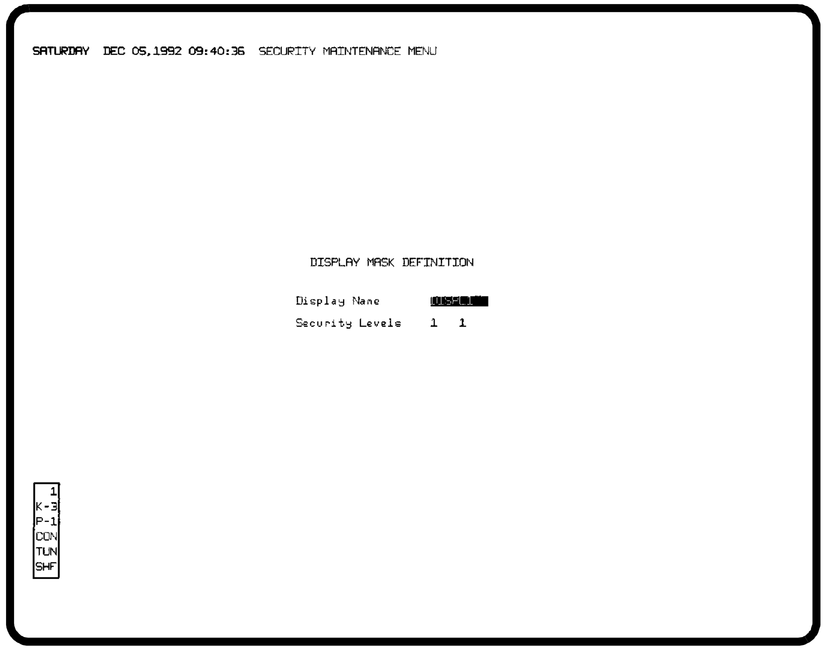
NOTE: A security level assignment made with the *Display Mask Definition* option overrides any made with the **sm** command and a security level assignment for an operator configurable display.

Display Mask Definition

Display mask definition defines a primary and secondary security level for a display. Do ***not*** define a display mask for a display unless it is to have limited access.

Use the *Display Mask Definition* option to define a security level for a display. Figure 5-6 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *G Security*
→ *D Display Mask Definition*



TPS0151B

Figure 5-6. Display Mask Definition Page

To define a display mask:

1. In the *Display Name* field, enter the name of the display to define. The display must reside on the hard disk as a ***DU*** display file. Enter the name without its extension.

2. In the first *Security Levels* field, enter the primary security level for the display. A valid entry is from 1 to 16. The maximum may actually be something less than 16 depending on the number of security levels enabled during security options configuration.

3. In the second *Security Levels* field, enter a secondary security level. This field is optional. A valid entry is from 1 to 16. This defines a second valid security level for the display and is checked only if the primary security level test fails.

4. Press **ENTER**.

   NOTE: If the display is already in the display cache when the display mask is changed, either call some other display on ***all*** console windows to clear the cache memory or reset the OIS application.

TAG SECURITY LEVEL AND GROUP

The ability to group process area related tags into up to 16 different security groups is provided. Also, the ability to assign a security level to a tag is provided. The tag group determines which users have access to the tag and the security level determines the actions that can be performed on the tag. The default security level assigned to a tag is level one. The default security group for a tag is group one.

A tag is put into a security group during tag configuration. Operations involving a tag such as control, alarm acknowledge, and tuning require access to the security group of the tag before they can be performed. During password definition, a user can be given or denied access to a group or groups. Access to the groups is enabled or disabled on an individual group basis. Access to any combination of the 16 groups can be given to a user.

Tag Access

To perform an action on a tag the following must be met.

- Tag security level allows the desired action (i.e., control, tune, etc.).
- Password definition of the logged in user gives access to the security group to which the tag belongs.
- User security level allows the desired action.

The tag and user security levels in this case do not have to agree. The tag and user access rights determine the actions that can be performed. As long as the user password gives access to a security group, the user security level has tag related access rights enabled, and the tag security level allows the same access rights, the user can perform operations on a tag.

Example: Example one in Table 5-8 shows one set of conditions that allow users to perform tuning on a tag. Examples two and three show a set of conditions that prevent tuning but still allow users to access the tag for other operations. Example three prevents tag tuning (and also configuration) for all users even if they have tuning access rights. All three assume the user has password access to the security group of the tag.

Table 5-8. Example Tag Security Level Check Results

Example	Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation
1	Tag (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 4)	Y	N	Y	Y	Y	Y	N	N	Y
	Actions allowed[1]	Y	N	Y	Y	Y	Y	N	N	Y
2	Tag (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 5)	Y	N	Y	N	Y	N	N	N	Y
	Actions allowed[1]	Y	N	Y	N	Y	Y	N	N	Y

Table 5-8. Example Tag Security Level Check Results (continued)

Example	Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation
3	Tag (level 5)	Y	N	Y	N	Y	N	N	N	Y
	User (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	Actions allowed[1]	Y	N	Y	N	Y	N	N	N	Y

NOTE:

1. The resulting actions allowed are determined by the tag and user.

Tag Security Level and Group Assignment

To assign a security level and group to a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter the desired security level in the *Security Level* field for the tag being configured. A valid entry is from 1 to 16.

3. Enter the number of a group in the *Security Group* field to assign the tag to that group. A valid entry is from 1 to 16.

LOG SECURITY LEVEL

Security levels can be defined for custom and sequence of events (SOE) logs. The security level defined for a log is checked before the user will be permitted to perform log status operation on the log. These operations include activating the log, deactivating the log, and canceling log printouts.

Log Access

The following criteria must be met for the user to perform an action on a log:

- Log security level gives log activation access.
- User security level gives log activation access.

The tag and user security levels in this case do not have to agree. The log and user access rights determine the actions that can be performed.

Example: Example one in Table 5-9 shows one set of conditions that allow a user to perform log operations. Examples two and three of this table show a set of conditions that prevent log operations. Example three prevents log operations for all users even if they have log activation access rights.

Table 5-9. Example Log Security Level Check Results

Example	Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation
1	Log (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 4)	Y	N	Y	Y	Y	Y	N	N	Y
	Actions allowed[1]	Y	N	Y	Y	Y	Y	N	N	Y
2	Log (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	User (level 5)	Y	N	Y	N	Y	N	N	N	Y
	Actions allowed[1]	Y	N	Y	N	Y	N	N	N	Y
3	Log (level 5)	Y	N	Y	N	Y	N	N	N	Y
	User (level 2)	Y	Y	Y	Y	Y	Y	Y	Y	Y
	Actions allowed[1]	Y	N	Y	N	Y	N	N	N	Y

NOTE:
1. The resulting actions allowed are determined by the log and user.

Custom Log Security Level Assignment

To assign a security level to a custom log:

1. Follow the steps given in ***Defining a Log*** in Section 12.

2. Enter a security level in the *Security Level* field of the log. A valid entry is from 1 to 16.

SOE Log Security Level Assignment

To assign a security level to an SOE log:

1. Follow the steps given in ***Defining an SOE Log*** in Section 12.

2. Enter a security level in the *Security Level* field of the log. A valid entry is from 1 to 16.

PASSWORD SECURITY CONFIGURATION FILES

The password security configuration resides in several different files in the [DATA.USN02] directory. Table 5-10 lists the files and describes the information contained in them.

Table 5-10. Password Security Configuration Files

File Name	Description
ACCESS.CF	Security mask and identification information for each window (user access rights mask, display access rights mask, tag security groups mask, user ID, user area access mask, user index number, logical CRT identification, and red tag key).

Table 5-10. Password Security Configuration Files (continued)

File Name	Description
CONDEF.CF	Console definition information.
LOGCRT.CF	Logical CRT definition information.
PASSWORD.CF	User password definition information.
SECURITY.CF	Security options data.

These files can be saved before password security configuration to allow returning to a default configuration. They should be saved after configuration as a backup. A complete configuration can be transferred to other consoles by copying these files.

EXAMPLE SECURITY CONFIGURATION

This section describes an example security strategy. This is only one possible strategy and is neither a required implementation nor a recommended implementation. The plant requirements for security should be established prior to any console configuration.

The plant in this example has three consoles numbered one, two, and three. All three have only one screen. Console one is a main console that sends windows to consoles two and three which are auxiliary consoles. Windows one through three are on console one, windows four through six are on console two, and windows seven and eight are on console three. Consoles one and two are operator consoles and console three is a maintenance console.

Security Options

The operating parameters (security options) for this example are set up as:

Number of Levels = 5
Number of Passwords = 6
Automatic Logout = No
Default User = 2

Security Levels

Five levels of access are described in this example, each representing the requirements for different plant personnel.

Level one - control system engineer; responsible for the entire console and therefore requires unlimited access.

Level two - process engineer; responsible for control of the process. Access to all functions except password security and the DDT function is required.

Level three - operator; responsible for monitoring, controlling, and tuning the process. Additionally, the operator needs to acknowledge alarms and access trending. Each operator is responsible for only a single group of related process points.

Level four - control system technician; requires the ability to access the database to view process and system points (tags).

Level five - maintenance mechanic; requires the ability to monitor various displays.

Table 5-11 is the worksheet mapping the security level access rights as described.

Table 5-11. Access Rights Worksheet

Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation	Configuration Key Lock	Tune Key Lock
1	X	X	X	X	X	X	X	X	X		
2	X	X	X	X	X	X			X	X	X
3	X		X	X	X				X	X	X
4	X								X	X	X
5	X									X	X

User Passwords

A password contains region access rights which map the areas of responsibility for each user. For this example, six users require passwords:

User one - control system engineer; since this user is responsible for the entire console, access to all areas is required.

User two - process engineer; needs to control and set up the process. This requires the ability to:

- Add or delete tags in the database.
- Monitor all process and system displays.
- Set up and activate logs.
- Access trend displays to analyze the process.
- Archive data for historical purposes.
- Isolate and record portions of the process for analysis using trend pens.

- Make changes to control modules.
- Control console alarming.
- Take tags in and out of service using red tagging.

User three - operator; needs the ability to:

- Monitor, control, and tune the area one process points (tags).
- Access the database to view tag configurations.
- Create operator displays on-the-fly.
- Access trend displays to monitor the process.
- Turn trend pens on or off.
- View the red tag status for a tag.

User four - operator; this user has the same responsibilities as user three except monitor, control, and tuning of area two process points.

User five - control system technician; only needs to view the database.

User six - maintenance mechanic; only needs to view various displays.

Table 5-12 is the worksheet mapping the region accesses as described.

NOTE: The security tag groups portion of Table 5-12 is filled in later in this example.

Table 5-12. Region Access Worksheet (without Security Tag Groups)

User	Security Level	Database	Display	Keyboard	System	Log	Time/Date	Real Trend	Archive	Pens	PCU Configuration	Alarm Inhibit	Red Tag	Security Tag Groups															
														1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	X	X	X	X	X	X	X	X	X	X	X	X																
2	2	X	X			X		X	X	X	X	X	X																
3	3	X	X					X		X			X																
4	3	X	X					X		X			X																
5	4	X																											
6	5		X																										

Security Tag Groups

In this example, the database contains 5,000 tags, each accessing various process and system points. Tags 200 through 400 pertain to process area one, and tags 700 through 850 pertain to process area two. Process area one tags are assigned to security group two, and process area two tags are assigned to security group three. User three (operator) has responsibility for tags in security group two, and user four (operator) has responsibility for tags in security group three.

Table 5-13 is the worksheet mapping the security tag groups for each security level. For users five and six, all security groups are enabled, but their security levels of five and six respectively allow monitor access only. This prevents these users from performing any type of action on any tags.

> **NOTE:** The region accesses in Table 5-13 were defined in a previous discussion. Refer to ***User Passwords*** in this section for an explanation.

Table 5-13. Region Access Worksheet (with Security Tag Groups)

User	Security Level	Database	Display	Keyboard	System	Log	Time/Date	Real Trend	Archive	Pens	PCU Configuration	Alarm Inhibit	Red Tag	Security Tag Groups															
														1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
2	2	X	X			X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3	3	X	X					X		X			X		X														
4	3	X	X					X		X			X			X													
5	4	X												X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	5		X											X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Window Access

Users can be restricted to certain consoles and windows. In this example, users one and two can log in and perform operations on any of the three consoles. User three can log in and perform operations on consoles one and two (windows one through six), user four on consoles one and two, and users five and six only on console three (windows seven and eight).

Tables 5-14 and 5-15 are the worksheets mapping the window access as described. A console definition worksheet was not used and the *Console Definition* option was left at its defaults.

Table 5-14. Logical CRT Definition Worksheet

Logical CRT No.	Console No.	Physical CRT No.
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8

NOTE: Physical CRT no. can be interpreted as window number.

Table 5-15. Security Level Logical CRT Access Worksheet

Security Level	Logical CRT Access															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	X	X	X	X	X	X	X	X								
2	X	X	X	X	X	X	X	X								
3	X	X	X	X	X	X										
4							X	X								
5							X	X								

User Profiles

Table 5-16 shows the user profiles created in this example. The user ID, password, and red tag key are defined for a user during password definition.

Table 5-16. User Profiles Worksheet

User	Security Level	User ID	Password	Red Tag Key	Logical CRT Access																Security Tag Groups Access															
					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	DCSENG	DCSENG	DCS	X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
2	2	PROCENG	PROCENG	PCE	X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3	3	OP1	OP1	OP1	X	X	X	X	X	X												X														
4	3	OP2	OP2	OP2	X	X	X	X	X	X													X													
5	4	PROCTEC	PROCTEC								X	X									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	5	MAINTMEC	MAINTMEC								X	X									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

SECTION 6 - TAG DATABASE

INTRODUCTION

The tag database contains information for each variable or point in the process control system that the console is to monitor or allow control. The database also defines attributes used in displays, logs, archiving, password security, and alarm management. Other database features related to the tag database and defined separately are tag descriptors and alarm comments. This section explains tag, tag descriptor, and alarm comment configuration.

There are two types of descriptors: Engineering unit and logic state. An engineering unit descriptor gives the unit of measurement for an analog value. A logic state descriptor identifies a digital state for a switching device. These identifiers appear in most functions and in displays that incorporate a tag descriptor escape command (**ec 37**, **ec 38**, and **ed 37**). Consoles on a common communication highway should use the same list of descriptors.

An alarm comment for a tag appears on a display when a process value exceeds a certain threshold or a digital variable changes to a specific state. The control configuration in a PCU module determines the alarm thresholds and states that trigger an alarm. A comment appears on alarm summaries or on any display that incorporates an alarm comment escape command (**ec 33** and **ed 33**). An alarm comment gives the ability to display text that describes, for example, either the purpose of the alarm or operator actions required to correct the alarm.

TAGS

The console allows on-line tag configuration. Tags can be added or deleted and almost any tag attribute can be changed, then immediately implemented by the console. The exception is deleting, adding, or modifying an N90STA tag that defines an NNIU01 module (SCM module type). This type of change requires resetting the OIS application before being implemented.

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. Also, a tag can represent a system control module or interface unit.

During database configuration, a tag is defined for each process variable that the console is to monitor and for each process device that is available for operator control. System tags are defined to monitor the INFI 90 Open system. The console does not require a tag to support PCU module management functions such as tuning and configuration. After a tag is defined, its index number or 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 console. Not all PCU function blocks can be assigned a tag.

Each tag type available on the console provides access to one or more function codes (FC). Table 6-1 lists and describes the console tag types.

Table 6-1. Tag Types

Tag Type	Support	Function Code[1]	Purpose
ANALOG	Analog exception report Analog in/channel Analog out/channel	FC 30 FC 222 FC 223	Acquires an analog exception reported value.
DAANALG[2]	M/A station - basic M/A station - cascade M/A station - ratio Control station	FC 21 FC 22 FC 23 FC 80	Acquires an analog exception reported value providing enhanced alarm management capabilities. This tag type is for data acquisition only. It incorporates a DA macro that supports a control station function code (FC 21, 22, 23, or 80).
	Data acquisition analog	FC 177	
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.
DADIGTL	Multi-state device driver	FC 129	Acquires a digital exception reported state providing enhanced alarm management capabilities. This tag type is for data acquisition only. It incorporates a DA macro that supports a multi-state device driver (FC 129) function code.
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.
DEVSTAT	Console peripherals	N/A	For internal console use only; no values are received over the communication highway for this type of tag. It is used to define printer type of a printer used for logging.
DIGITAL	Digital exception report Digital in/channel Digital out/channel	FC 45 FC 224 FC 225	Acquires a digital exception reported state.

Table 6-1. Tag Types (continued)

Tag Type	Support	Function Code[1]	Purpose
INTANG	User task	N/A	For internal console use only; no values are received over the communication highway for this type of tag. It can be used to read an analog value from a UTI program to be used in console functions.
INTDIG	User task	N/A	For internal console use only; no values are received over the communication highway for this type of tag. It can be used to read a digital value from a UTI program to be used in console functions.
	SOE log	N/A	Can be used to write descriptive tag information for a point being recorded in an SOE log.
MSDD	Multi-state device driver	FC 129	Acquires an exception reported status for a three-state device. It also allows initiating manual control.
N90STA	System status	N/A	Reads detailed status information and problem reports from INFI 90 Open equipment over the communication highway. **NOTE:** For Plant Loop nodes to receive the same time-stamped message to support distributed trending, all nodes on the loop must have an N90STA tag.
RCM	Remote control memory	FC 62	Acquires an exception reported set or reset state of a device. It also allows initiating device control.
	SOE log	FC 99 FC 210	Monitors the operation of a sequential events recorder, SOE (FC 210) function block, or INSEM01 module and allows requesting sequence of events summary reports.
RMCB	Remote motor control	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 console. It also contains a color and blink option.
TEXTSTR	User-defined data export	FC 194	Enables communication between the console and a C language or batch program running in a PCU module to allow transfer of text strings.

NOTES:

1. Refer to the ***Function Code Application Manual*** for more information and for descriptions of function code specifications (Table 1-2 lists instruction numbers).

2. Does not enable the full functionality of the data acquisition analog (FC 177) function code.

TAG CONFIGURATION

Tag database configuration requirements include:

- Enabling tag broadcasting if desired. Refer to ***TAG BROADCASTING*** in this section for the procedures. This should be performed before configuring any tags if used.

- Defining engineering unit descriptors. Refer to ***Engineering Unit Descriptors*** in this section for the procedures.

- Defining logic state descriptors. Refer to ***Logic State Descriptors*** in this section for the procedures.

- Defining alarm comments. Refer to ***ALARM COMMENTS*** in this section for the procedures.

- Defining tags. Tag descriptors and alarm comments should be entered before defining tags.

NOTE: The preferred method for creating a tag database is by using the console configuration tools (WLDG). The console should mainly be used to update or make changes to the existing database.

During start-up or reset, each tag in the database list is downloaded to the communications interface unit of the console. The interface unit requires this to establish communication with PCU modules. Tag configuration cannot take place while the tag database list is being downloaded.

Defining a Tag

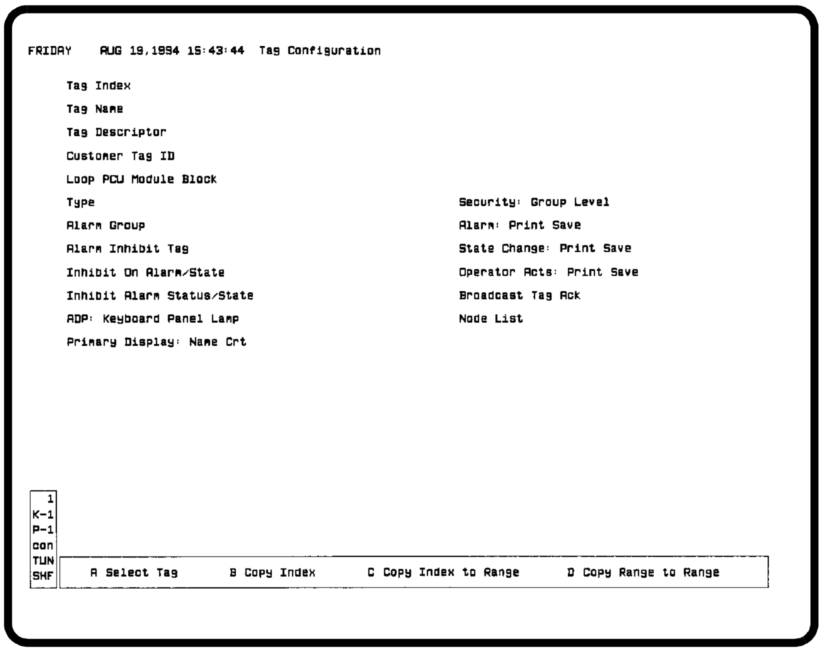
Each tag in the database has a name of up to 14 characters and a tag index number from 1 to 30,000. A tag name and index number identifies the tag throughout all functions. Each process point that is to be monitored or controlled and each control module that is to be monitored requires a tag in the database.

The database allows for a 32-character tag descriptor that describes the purpose of the tag. An additional 32-character customer identifier also can be entered to further explain the process point. These descriptions help to identify the tag while on certain displays.

In the database, certain tag attributes remain the same 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.

Use the *Tag* option to define a tag. Figure 6-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *A Tag*



TPS0014D

Figure 6-1. Tag Page

Use the tag configuration submenu to select a tag to configure or edit. The submenu also provides the capability to copy tags. Copying can be used to expedite tag configuration.

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.

To define a tag:

1. Call the definition page for a tag.

   a. Choose *A Select Tag* from the submenu. A *Tag Name or Index* field appears.

   b. Type a tag name or an index number for an existing tag. To add a new tag, either type the index number of a known undefined tag or leave the field blank to call the next undefined tag. A valid entry is from 1 to 30000. The maximum may actually be something less than 30000 depending on the number of tags enabled during system configuration.

c. Press **ENTER** to call the tag.

NOTE: The *Print Database List* option can be used to compile and print a list of undefined tags. It can also be used to compile and print a complete tag database or a listing of selected tags. Refer to the discussion on tag operations in the ***Operation*** instruction for an explanation of this option (Table 1-2 lists instruction numbers).

2. Enter the appropriate data into each field of the page. Table 6-2 explains the fields that are common to all tag types. Tables 6-3 through 6-16 explain additional fields that relate to specific types of tags. Refer to these tables when entering data.

3. Press **ENTER**.

To enter additional tags, press **ESC** to call the submenu to the screen. Or, press **NEXT PAGE** or **PREV PAGE** to sequence to the next or previous numbered tag without having to use the *Select Tag* option.

NOTE: The *Alarm Groups* option can be used to make alarm group assignments. The option can assign a range of tags to an alarm group. Refer to ***ALARM GROUPS*** in Section 9 for the procedures.

Table 6-2. Tag Database - General Fields

<table>
<tr><th>Field</th><th colspan="2">Description</th></tr>
<tr><td>Tag index</td><td colspan="2">Non input field; shows the index number of the tag selected for editing.</td></tr>
<tr><td>Tag name</td><td colspan="2">14-character tag name. The name appears in most functions to identify the tag.</td></tr>
<tr><td>Tag descriptor</td><td colspan="2">32-character descriptor normally used to describe the purpose of the tag. The descriptor appears in most functions.</td></tr>
<tr><td>Customer tag ID</td><td colspan="2">Additional 32-character tag identifier.</td></tr>
<tr><td rowspan="5">Loop, PCU, module, block</td><td colspan="2">Hardware communication loop, PCU, and module number, and software function block number.

NOTE: For a DEVSTAT, INTANG, and INTDIG tag, use 0-0-0-0. Use 0 as the block number for a N90STA tag. Refer to Tables 6-15 and 6-16 for module number to use for certain types of N90STA tags.</td></tr>
<tr><td>Loop</td><td>0 to 250 (INFI-NET system up to 250, Plant Loop system 0 or 1).</td></tr>
<tr><td>PCU</td><td>0 to 250 (INFI-NET system up to 250, Plant Loop system up to 63).</td></tr>
<tr><td>Module</td><td>0 to 31.</td></tr>
<tr><td>Block</td><td>0 to 9998.</td></tr>
<tr><td>Type</td><td colspan="2">Valid tag types are:

ANALOG, DAANALG, DADIG, DADIGTL, DANG
DD, DEVSTAT, DIGITAL, INTANG, INTDIG
MSDD, N90STA, RCM, RMCB, RMSC
STATION, TEXT, TEXTSTR</td></tr>
</table>

Table 6-2. Tag Database - General Fields (continued)

<table>
<tr><th>Field</th><th colspan="2">Description</th></tr>
<tr><td>Alarm group[1]</td><td colspan="2">Alarm group assignment. A valid entry is from 0 to 99. Default is D for a DEVSTAT tag and S for an N90STA tag. Use 0 to leave the tag unassigned. An unassigned tag does not trigger alarm group alarm indications when it enters an alarm condition.

NOTE: This type of console does not use the DEVSTAT tag to monitor the status of its peripherals. Alarm group D is considered invalid for alarming purposes even though the console assigns a DEVSTAT tag to this group.</td></tr>
<tr><td>Alarm inhibit tag[1]</td><td colspan="2">Configures automatic alarm inhibiting. This is the tag used to inhibit alarm indications for the 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 fields are Inhibit On Alarm/State and Inhibit Alarm Status/State.</td></tr>
<tr><td>Inhibit on alarm/ state[1]</td><td colspan="2">Configures automatic alarm inhibiting. This is the inhibit tag condition that is to trigger 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.</td></tr>
<tr><td>Inhibit alarm status/ state[1]</td><td colspan="2">Configures automatic alarm inhibiting. This is the specific alarm condition or digital state that is to trigger alarm inhibiting. Not valid unless a tag name is entered in the Alarm Inhibit Tag field. A valid entry for this field depends on the type of tag selected as the alarm inhibit tag and the Inhibit On Alarm/State field.

If the Inhibit On Alarm/State field is set to ALARM, a valid entry is H, 2H, 3H, L, 2L, 3L, HD, LD, A, and blank.

If the Inhibit On Alarm/State field is set to STATE, a valid entry is 0, 1, 2, 3, or blank.</td></tr>
<tr><td rowspan="4">ADP:[1]</td><td colspan="2">Annunciator display panel (ADP) indicator that is to light when this tag goes into alarm. The entries identify the keyboard to which the panel is tied, specific panel, and specific LED (Lamp).</td></tr>
<tr><td>Keyboard</td><td>0 to 8; using 0 allows panels to mimic each other. For example, use 0 to have a two-keyboard console with one annunciator display panel per keyboard be configured so both panels use the same tag entries and perform the same function.</td></tr>
<tr><td>Panel</td><td>0 to 4; use 0 to specify no panel assignment for this tag.</td></tr>
<tr><td>Lamp</td><td>0 to 32; use 0 to specify no lamp assignment for this tag.</td></tr>
<tr><td rowspan="2">Primary display:[1]</td><td>Name</td><td>Name of any assembled display (DU). This display becomes the primary display called from the alarm summary, from the operating parameters function, or with DISPLAY.</td></tr>
<tr><td>CRT</td><td>Window number on which the display is to appear. A valid entry is from 0 to 8. Use 0 to cause the display call-up to occur on the current window. Enter any number from 1 to 8 to designate call-up on a specific window.</td></tr>
<tr><td rowspan="2">Security:</td><td>Group</td><td>A tag can be assigned to one of 16 security groups. Access to the group can be permitted by enabling the group number in the password of a user. A valid entry is from 1 to 16.</td></tr>
<tr><td>Level</td><td>A tag can be assigned to one of 16 security levels. Operations that can be performed on the tag are based on security level. A valid entry is from 1 to 16.</td></tr>
</table>

Table 6-2. Tag Database - General Fields (continued)

Field	Description	
Alarm:	Print	Determines if alarm events for this tag appear in a continuous printout of the event log. A valid entry is: YES = enable printing alarm events as they occur. NO = disable printing alarm events as they occur.
	Save	Determines if alarm events for this tag are saved to disk to appear in a periodic printout of the event log. This also configures alarm events for archiving. A valid entry is: YES = enable saving alarm events. NO = disable saving alarm events.
State change:	Valid for DADIG, DIGITAL, DADIGTL, INTDIG, RCM, RMCB, DD, MSDD, and TEXT tag types.	
	Print	Determines if state change events for this tag appear in a continuous printout of the event log. A valid entry is: YES = enable printing state change events as they occur. NO = disable printing state change events as they occur.
	Save	Determines if state change events for this tag are saved to disk to appear in a periodic printout of the event log. This also configures state change events for archiving. A valid entry is: YES = enable saving state change events. NO = disable saving state change events.
Operator actions:	Print	Determines if operator action events for this tag appear in a continuous printout of the event log. A valid entry is: YES = enable printing operator action events as they occur. NO = disable printing operator action events as they occur.
	Save	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. A valid entry is: YES = enable saving operator action events. NO = disable saving operator action events.
Global alarm management [1]	Configures alarm change broadcasting. It determines if an alarm change for this tag is to be broadcast on the communication highway to change the alarm inhibit and scan on/off state, and acknowledge and silence alarms on other consoles or computers. A node list selected in the *Node list* field determines to which nodes to broadcast the changes. A valid entry is: YES = broadcast alarm change. NO = do not broadcast alarm change.	
Node list [1]	Alarm management node list for broadcasting alarm changes. It selects which nodes are to receive broadcast changes. Not valid unless the *Global Alarm Mgt* field is *YES*. A valid entry is: 0 = node list automatically compiled by the console during start-up. The list contains the first 32 N90STA tags in the database that define either a console or a computer. 1 to 4 = one of four user-defined node lists.	

NOTE:
1. Relates to alarm management.

Table 6-3. Tag Database - ANALOG and INTANG Tag Fields

Field	Description
Engineering unit	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 places	Number of decimal places 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: Return to normal Bad quality High Low A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
I/O subtype	Indicates if a tag is derived from an enhanced I/O function code (FC 222 or FC 223) or a standard I/O function code FC 30. The options for this field are: STD = FC 30 ENH/IN = FC 222 ENH/OUT = FC 223
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal High Low Enter **0** to not associate a comment with a condition.

Table 6-4. Tag Database - DANG Tag Fields

Field	Description
Engineering unit	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 places	Number of decimal places 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: Return to normal, High, High deviation Bad quality, Low, 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 8; default is 1. Priority one is the highest priority.
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal, Low, Low deviation Three-high, Two-low, High rate of change Two-high, Three-low, Low rate of change High, High deviation Enter **0** to not associate a comment with a condition.

Table 6-5. Tag Database - DAANALG Tag Fields

Field	Description
Engineering unit	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 places	Number of decimal places to show for this tag. A valid entry is from 0 to 4.
Alarm type	STANDARD = standard high and low alarm limit alarming. FIXED = multilevel alarming using fixed alarm levels (e.g., three-high to three-low). VARIABLE = multilevel alarming using variable alarm levels. Alarm levels are based on the output values of other function blocks. The *Hi Var Alrm Tag* and *Low Var Alrm Tag* fields identify the function blocks setting the variable alarm levels.
Alarm deadband	Deadband value that either adds or subtracts from the alarm limit depending on the direction of deviation. This is a four-character floating point value used for three-high, two-high, two-low and three-low alarm limits. Valid only when alarm type is FIXED or VARIABLE.
Priority[1]	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: Return to normal High Bad quality Low Three-high Two-low Two-high Three-low A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Comment[1]	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Low Three-high Two-low Two-high Three-low High Enter **0** to not associate a comment with a condition.
Limit[1]	For reference only; the actual limits are set in the PCU module. Information entered in these fields displays when the operating parameters page is called for the tag. The alarm conditions requiring a limit value include: Three-high Two-low Two-high Three-low High Low
High variable alarm tag	Tag whose value determines the high variable alarm limit. Valid only when the alarm type is VARIABLE.
Low variable alarm tag	Tag whose value determines the low variable alarm limit. Valid only when the alarm type is VARIABLE.

NOTE:

1. Three-high, two-high, two-low, and three-low are required only when using the FIXED or VARIABLE alarm type.

Table 6-6. Tag Database - DADIG, DADIGTL, and RCM Tag 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 conditions that can have a priority assigned include: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.	
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.	

NOTE:
1. Refer to ***Defining a Logic State Descriptor*** in this section for the procedures to call a list of valid logic state descriptors.

Table 6-7. Tag Database - DIGITAL and INTDIG Tag 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 conditions that can have a priority assigned include: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.	
I/O subtype	Indicates if a tag is derived from an enhanced I/O function code (FC 224 or FC 225) or a standard I/O function code FC 45. The options for this field are: STD = FC 45 ENH/IN = FC 224 ENH/OUT = FC 225	
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.	

NOTE:
1. Refer to ***Defining a Logic State Descriptor*** in this section for the procedures to call a list of valid logic state descriptors.

Table 6-8. Tag Database - STATION Tag Fields

<table>
<tr><th>Field</th><th>Description</th></tr>
<tr><td>Engineering unit</td><td>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.</td></tr>
<tr><td>Decimal places</td><td>Number of decimal places to show for this tag. A valid entry is from 0 to 4.</td></tr>
<tr><td>Tuning block</td><td>Function block that is to appear in the block details portion of a tuning display (normally a PID block). A valid entry is:

0 = default; use if the function block providing the station block input is a PID block. If used and the function block is not a PID block, a prompt for block number will appear when the tuning display is called.

11 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.</td></tr>
<tr><td>Priority</td><td>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:

Return to normal   Low
Bad quality   High deviation
High   Low deviation

A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.</td></tr>
<tr><td>Comment</td><td>Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include:

Return to normal   High deviation
High   Low deviation
Low

Enter 0 to not associate a comment with a condition.</td></tr>
</table>

Table 6-9. Tag Database - DD, MSDD, and RMCB Tag Fields

<table>
<tr><th>Field</th><th colspan="2">Description</th></tr>
<tr><td rowspan="4">State:[1]</td><td>Zero</td><td>Six-character logic state descriptor that is to identify a zero state.</td></tr>
<tr><td>One</td><td>Six-character logic state descriptor that is to identify a one state.</td></tr>
<tr><td>Two
(MSDD only)</td><td>Six-character logic state descriptor that is to identify a two state.</td></tr>
<tr><td>Three
(MSDD only)</td><td>Six-character logic state descriptor that is to identify a three state.</td></tr>
<tr><td rowspan="2">Feedback one:[1]</td><td>Zero</td><td>Six-character logic state descriptor that is to identify a zero state for feedback one.</td></tr>
<tr><td>One</td><td>Six-character logic state descriptor that is to identify a one state for feedback one.</td></tr>
<tr><td rowspan="2">Feedback two:[1]</td><td>Zero</td><td>Six-character logic state descriptor that is to identify a zero state for feedback two.</td></tr>
<tr><td>One</td><td>Six-character logic state descriptor that is to identify a one state for feedback two.</td></tr>
</table>

Table 6-9. Tag Database - DD, MSDD, and RMCB Tag Fields (continued)

Field	Description	
Feedback three:[1] (MSDD only)	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.
Feedback four:[1] (MSDD only)	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.
Permissive one:[1] (RMCB only)	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.
Permissive two: (RMCB only)	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 (RMCB only)	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.	
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: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.	
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.	

NOTE:
1. Refer to ***Defining a Logic State Descriptor*** in this section for the procedures to call a list of valid logic state descriptors.

Table 6-10. Tag Database - RMSC Tag Fields

Field	Description
Engineering unit	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 places	Number of decimal places to show for this tag. A valid entry is from 0 to 4.

Table 6-10. Tag Database - RMSC Tag 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: Return to normal Bad quality A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Comment	Index number of an alarm comment that is to appear when the tag enters return-to-normal condition. Enter **0** to not associate a comment with the condition.

Table 6-11. Tag Database - DEVSTAT Tag Fields

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: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.
Device type	Type of peripheral device. A valid device type is: PRINTER UNDEFINED
Printer number	Number to select one of four possible logging printers for a device type of *PRINTER*.
Printer type	Type of printer for a device type of *PRINTER*. A valid entry is: ANSI (low speed, no color) ANSI COLOR (low speed, color) ANSI HIGH (high speed, no color) IBM® (low speed, no color) IBM COLOR (low speed, color) IBM B&W 24 IBM COL 24 UNDEFINED

Table 6-12. Tag Database - TEXT Tag Fields

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: Return to normal Bad quality A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Comment	Index number of an alarm comment that is to appear when the tag enters return-to-normal condition. Enter **0** to not associate a comment with the condition.

Table 6-13. Tag Database - TEXTSTR Tag Fields

Field	Description
Text length	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 console uses the value to determine the maximum length of the text string it will accept. A valid entry is from 1 to 80.
Control enabled	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 or touch pads. The results of the operator actions can be seen on the screen. When disabled, the console provides information but does not allow control. A valid entry is: YES = default; enable control. NO = 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: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.

Table 6-14. Tag Database - N90STA Tag Fields

Field	Description
Module type	Required by the console to properly interpret module status reports received from a control module. The information appears on system status pages. In some cases, the module type is for a module that is part of the network interface unit of a node. A node type is presented instead of a module type on the system status overview and node status summary pages in this case. Refer to Table 6-15 for valid module types for consoles and Table 6-16 for valid module types for PCU modules.
Global Alarm Mgt./ Silence:	Configures global alarm management broadcasting. The fields provide an alternate means for turning broadcasting on and off for this tag instead of having to modify a node list.
	YES = broadcast alarm acknowledge, silence, inhibit and scan changes. NO = default; do not broadcast.
NIU port 0/1: terminal language	Not applicable for this console; leave these fields at default.
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: Return to normal Bad quality Alarm A valid entry is from 1 to 8; default is 1. Priority one is the highest priority.

Table 6-14. Tag Database - N90STA Tag Fields (continued)

Field	Description
Comment	Index number of an alarm comment that is to appear when the tag enters a certain alarm condition. The conditions that can have an alarm comment include: Return to normal Alarm Enter **0** to not associate a comment with a condition.

Table 6-15. N90STA Tag - Module Type for Consoles

Module Type[1]	Nomenclature	Description
COMPUTER[2]	N/A	Computer connected to the communication highway through a network interface unit.
OIS10	IIOIS10 IIOIS11	Operator interface station
OIS20	IIOIS201/A/D IIOIS251/A/D	
OIS40	IIOIS401/A/D IIOIS411/A/D IIOIS421 IIOIS431	
OIU	NOIU01/02/03	Operator interface unit
MCS	NMCS02 NMED01	Management command system (MCS and MCS PLUS)
SIG12	IS12	Signature console
SIG42	IS42 IS43	

NOTES:

1. All consoles use a module address of 2.
2. Used to access computers running process interface software packages (e.g., SGDM, XRS 90 Data Management, 1090 Process Management, etc.).

Table 6-16. N90STA Tag - Module Types for PCU Modules

Module Type	Nomenclature	Description
AMMH	NAMM01	Analog master module (high)
AMML	IMAMM03 NAMM02/03	Analog master module (low)
AOM	IMAOM01, NAOM01	Analog output module
BIM[1]	INBIM02, NBIM01/02	Bus interface module
BTM[2]	INBTM01, NBTM01	Bus transfer module
CBC	CBC01	Batch command controller
CLC	CLC03/04	Loop command controller
COM/QRC	IMCOM03/04 NCOM02/03/04 IMQRC01, NQRC01	Controller module and quick response controller module
CSC	CSC01	Sequence command controller

Table 6-16. N90STA Tag - Module Types for PCU Modules (continued)

Module Type	Nomenclature	Description
CTM	NCTM01	Configuration and tuning module
CTT	NCTT01/02	Configuration and tuning terminal
GCM/PPT[3]	INPPT01, NGCM02	Plant Loop to Plant Loop transfer module (remote)
IIT01[4]	INIIT01, NBCM01	INFI-NET to INFI-NET transfer module (local)
IIT02[4]	INIIT02, NGCM03	INFI-NET to INFI-NET transfer module (remote)
IPT01[4]	INIPT01, NGCM04	INFI-NET to Plant Loop transfer module (local)
IPT02[4]	INIPT02	INFI-NET to Plant Loop transfer module (remote)
LCM01	NLCM01	Large controller module
LCM02	NLCM02	
LCM03	NLCM03	
LMM01	NLMM01	Logic master module
LMM02	IMLMM02, NLMM02	
LSM/PCT[5]	INPCT01, NLSM01/02	Plant Loop to computer transfer module (interface unit for MCS and MCS PLUS console and 5,000-tag OIU console)
MFC	IMMFC03/04/05 NMFC01/02/03/04/05	Multi-function controller module
MFP	IMMFP01/02/03	Multi-Function processor module
MPC	IMMPC01, NMPC01	Multi-processing module
NPM[1]	INNPM01	Network processing module (Controlway)
PBUG	N/A	Module bus debugger
PIM[6]	NPIM01	Processor input module (interface unit for OIU)
PTM[3]	INPTM01, NPTM01	Point table module
SBM[1]	NSBM01	Superloop bus module
SCM[7]	NSCM01, NNIU01	Serial communication module (network interface unit)
SEM[3]	IMSEM01	Sequence of events module

Table 6-16. N90STA Tag - Module Types for PCU Modules (continued)

Module Type	Nomenclature	Description
SSM/ICT[5]	INICT01, NSSM01	INFI-NET to computer transfer module (interface unit for MCS and MCS PLUS console and 5,000-tag OIU console)

NOTES:
1. PCU node; use a module address of either 0 or 1.
2. Use a module address of 3.
3. Use a module address of 2.
4. Communication module; use a module address of 0 or 1.
5. MCS and MCS PLUS console or 5,000-tag OIU interface module; use a module address of 2.
6. OIU interface module; use a module address of 2.
7. Use a module address of 0.

Configuration Submenu

The options of the tag configuration page submenu (Figure 6-1) allow selecting a tag for editing and allow copying tags. A tag that has been copied has the same hardware address as the original tag except the function block address is set to some invalid number. The tag is ***not*** defined until after entering a unique tag name and address. Before using the copy features, verify that a tag is not being copied to an already existing tag since this overwrites an existing tag.

The *Select Tag* option selects a single tag for editing. To use the option:

1. Press **A**. A *Tag Name or Index* field appears.

2. Key in the name of a tag or its index number. If the field is left blank or cleared, the next available, undefined tag will be called.

3. Press **ENTER** to call the tag.

The *Copy Index* option copies a single tag to a single index number. To use the option:

1. Press **B**. The following prompt appears:

Copy Index ___ To ___

2. The index number of the current tag appears in the first input field. Either use this tag number or enter a different one.

3. Enter an index number to which to copy this tag in the *To* field.

4. Press **ENTER** to execute the copy.

The *Copy Index to Range* option copies a single tag to a range of index numbers. To use the option:

1. Press **C**. The following prompt appears:

Copy Index ___ To ___,___

2. The index number of the current tag appears in the first input field. Either use this tag number or enter a different one.

3. Enter a range of index numbers to which to copy this tag in the *To* fields.

4. Press **ENTER** to execute the copy.

The *Copy Range to Range* option copies of range of tags to a range of index numbers. To use the option:

1. Press **D**. The following prompt appears:

Copy Range ___,___ To ___,___

2. The index number of the current tag appears in the first *Range* field. Either use this tag number or enter a different one.

3. In the second *Range* field, enter another tag index number to complete the range of tags to copy.

4. Enter a range of index numbers to which to copy the range of tags in the *To* fields.

5. Press **ENTER** to execute the copy.

Deleting a Tag

Change the *Tag Type* field of a tag to *UNDEF* to delete the tag. As soon as **UNDEF** is typed and **ENTER** is pressed, a default undefined configuration appears.

Changing Tag Attributes

Care must be taken when making changes to the database since the tag database is the foundation for ***all*** other functions. For example, deleting or changing a tag that is used to inhibit alarming of other tags requires changing or modifying all tags that use it for inhibiting.

Another example is making a node type change to a node N90STA tag. This, in most cases, then requires making changes to all the module N90STA tags for that node. The recommended procedure for changing the node type of a node N90STA tag is to first delete all PCU module N90STA tags for that node, make the appropriate changes to the node type, then add any module N90STA tags required.

A validation check is run by the console when changing a digital type tag (e.g., DIGITAL, DD, MSDD, etc.) to an analog type tag (e.g., ANALOG, STATION, DAANALG, etc.) before the change is made. The check is performed to determine if the state of the digital tag is being used for alarm inhibiting. If it is, the following message appears:

Alarm Inhibit Mode Conflict on Tag n

The tag type change cannot be completed until correcting the mode conflict for tag n. A possible solution is to change the *Inhibit On Alarm/State* field for tag n to **ALARM**. The validation check is also performed during a copy operation.

TAG BROADCASTING

The console has the ability to broadcast and receive any change made to a tag in its database. The broadcast ability allows any change made from this console to be incorporated and implemented on other consoles. The receive ability allows any change made from another console to be incorporated and implemented by this console. This requires a work station connected to the INFI-NET communication highway and running the global database manager (GDM) software program. Refer to the ***Software Global Database Manager*** instruction for a description of program operations and for work station requirements (Table 1-2 lists instruction numbers).

When a change is made to a tag and that change is then broadcast, the GDM program automatically performs a consistency check to validate the change. It verifies that the tag name to loop, PCU, module, and block number is unique across the entire database. This allows the GDM program to maintain database integrity for all consoles or devices on the communication highway that use the database.

Additionally, an entire tag list can be broadcast to this console from the GDM work station; however, this requires the console to be taken off-line by running a command file from a VT-series terminal or a terminal window.

General parameters page fields must be set during system configuration to enable tag broadcasts. The console must be operating on INFI-NET communication highway to enable this feature.

Enabling On-Line Tag Broadcasts

To configure the console for tag broadcasting:

1. Follow the steps given in ***GENERAL PARAMETERS*** in Section 3. The fields that enable on-line broadcasting of tag changes are *Broadcast Master EWS:* and *Broadcast Active*. Additionally, the *Broadcast Message Type* field must be set to provide proper communication.

2. In the *Broadcast Master EWS: loop and node* fields, enter the loop and node number of an interface unit. This should be the address of the interface unit that connects the work station running the GDM program to the communication highway. A valid entry is from 1 to 250 for both fields. This identifies the specific work station that is maintaining the database of this console. Several work stations running the global database manager can be connected to the loop at one time.

3. Enter **YES** in the *Broadcast Active* field to enable both transmitting and receiving single tag changes and to enable receiving a broadcast of a complete tag list. This field has two other choices: NO or LOCAL. Enter **NO** to disable both transmitting and receiving of any type of tag broadcast. Enter **LOCAL** to allow receiving a broadcast of a tag list and any single tag changes from the GDM work station. Using LOCAL disables broadcasting single tag changes to the GDM work station.

4. Refer to ***GENERAL PARAMETERS*** in Section 3 for proper setting of the *Broadcast Message Type* field.

Tag Broadcast Errors

The tag configuration page can be exited anytime after **ENTER** has been pressed. A message verifying a successful broadcast or indicating a broadcast failure will not be seen, however, if the page is exited before the GDM program can issue a reply.

If the broadcast was successful, the tag configuration changes save to the hard disk. If the GDM program rejects the tag change, the console displays an error message and does ***not*** update its database. If the configuration is exited and the GDM program rejects the change, the update is considered to be aborted and following message is recorded in the event log (operator actions log):

Tag <index number>, <tag name> Update aborted by <user ID> at <CRT n>

All tag changes are logged in the event log. This provides a way to recover edits made to a tag.

An error message appears if tag updates are made on other windows supported by the console simultaneously. The tag broadcast function can broadcast and validate only one tag at a time. Another broadcast attempt while the console is already processing a broadcast causes a busy reply from the broadcast function.

Error messages are given if a broadcast fails. These include:

Tag Update Failed - Duplicate Tagname

Tag Update Failed - Duplicate Address

The console places the input cursor on the input field in error. Correct the field in error and press **ENTER** to save the tag and to attempt another broadcast.

The following message is a general error message:

Tag Update Failed

It normally appears when attempting a tag broadcast to an off-line GDM program or if a communication problem exists. This returns the input cursor to the *Tag Name* field. Refer to the ***Software Global Database Manager*** instruction for additional information (Table 1-2 lists instruction numbers).

Enabling Off-Line Tag List Broadcast

An entire tag list can be broadcast to this console from a work station running the global database manager (GDM) program; however, this requires the OIS application to be put in an off-line tag mode by running a utility from a VT-series terminal or a terminal window. The procedure is similar to running the database builder. The following sections give the procedures to put the application in off-line tag mode to receive a broadcast tag list, then to put it back on-line.

OFF-LINE TAG MODE

Before initiating a broadcast from a work station, perform the following steps:

1. Shut down the OIS application if it is currently running. Use the procedures given in ***Shutdown*** in Section 2.

2. At a VT-series terminal or a terminal window, log into the OISENGR account. Refer to the ***File Utilities*** instruction for the procedures to log into an account from a VT-series terminal. Refer to the ***Operation*** instruction for the procedures to use the *Login Window* option to open a terminal window logged into an account.

3. At the prompt ($), type:

   TAGMODE Return

ON-LINE MODE

Upon completion of the tag list broadcast, put the OIS application back in an on-line mode by doing one of two things. At the prompt ($) of a logged in VT-series terminal or a terminal window, type:

OISRESET Return

- ***or*** -

Choose *OIS Reset* from the *Startup/Shutdown* pull-down menu of the session manager.

> **NOTE:** The *OIS Reset* option first shuts down the TAGMODE utility, then starts the OIS application.

ALARM COMMENTS

An alarm comment is text associated with an alarm condition of a tag in the database. An alarm comment 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 comments 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.

Alarm comments reside in an alarm comment file (***ALMDESC.CF***). Alarm comment configuration modifies or creates this alarm list. The number of comments in the file is variable with a maximum of 20,000 entries.

The preferred method for creating an alarm comment list is by using the console configuration tools (WLDG) then transferring the comment file to the console. Refer to the ***Console Configuration Utilities*** instruction for the procedures (Table 1-2 lists instruction numbers). The configuration method on the console is better suited for making modifications to an existing list.

> **NOTE:** If changes to the alarm comment list are made on the console, it is suggested that the alarm comment file be transferred back to the engineering work station where the alarm comment file is being maintained. Transferring the file back maintains the database integrity.

Alarm comment configuration pages allow viewing, modifying, adding, deleting, and printing alarm comments. Use the *Alarm Comments* option to define an alarm comment. Figure 6-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
- *A Database*
  - *I Alarm Comments*

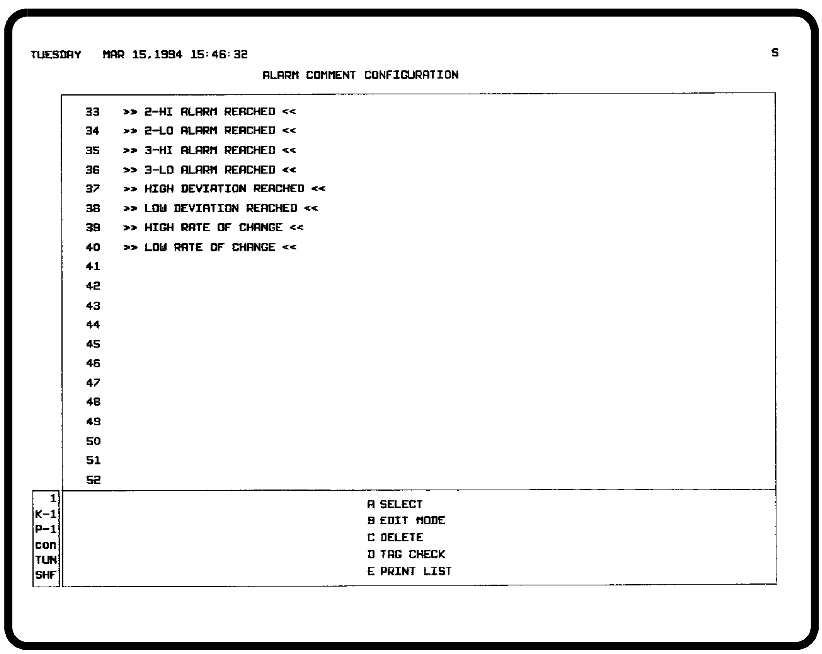


Figure 6-2. Alarm Comments Page - Initial Page

Paging Through Entries

The entire alarm comment file contents can be viewed by paging through the alarm comment configuration pages. Press

NEXT PAGE or **PREV PAGE** to call the next or previous page of comments to the screen. Press **HOME** to return to the first page and the beginning of the file.

Using the Select Option

The *A SELECT* option calls an alarm comment by index number. After being selected, the comment appears as the first entry on the page. This also positions an input cursor on the comment for immediate editing. To use this option:

1. Press **A** to select the option.

2. Key in the index number of an alarm comment and press **ENTER**.

3. Change, modify, or clear the selected comment. While in this mode, the input cursor can be moved to other comments in the list.

4. Press **ENTER** to update the alarm comment file.

Using the Edit Option

The *B EDIT MODE* option puts the current page into editing mode and enables an input cursor. Pressing **ENTER** before any other selection does the same thing. To use this option:

1. Press **B** or **ENTER** to enable editing mode.

2. Change, modify, or clear any comments. While in this mode, the input cursor can be moved to other comments in the list.

3. Press **ENTER** to update the alarm comment file.

Using the Delete Option

The *C DELETE* option specifies a single, list, or range of alarm comments to delete from the alarm comment file. After selecting this option, the next page presents two deletion choices: *LIST* or *RANGE*.

Use the *A LIST* option to specify a single or list of up to five alarm comments to delete. To use this option:

1. Press **C** to select *DELETE*.

2. Press **A** to select *LIST*. A single input field appears.

3. Key in the index number of a comment to delete. If only one comment is to be deleted, go to the next step. Press **TAB** to

enter additional index numbers. Up to four additional comments can be specified for deletion.

4. Press **ENTER** to initiate the deletion and update the alarm comment file.

Use the *B RANGE* option to specify a from and to range of alarm comments to delete. To use this option:

1. Press **C** to select *DELETE*.

2. Press **B** to select *RANGE*. The page presents two input fields:

   START: ___

   END: __

3. Enter the index number of the first comment in the range in the *START* field.

4. Enter the index number of the last comment in the range in the *END* field. If only one comment is to be deleted, enter the same index number in both fields.

5. Press **ENTER** to initiate the deletion and update the alarm comment file.

Using the Tag Check Option

The *D TAG CHECK* option allows checking the tag database to determine which tags currently use an alarm comment. To use this option:

1. Press **D** to select *TAG CHECK*. The page presents a single *INDEX* field.

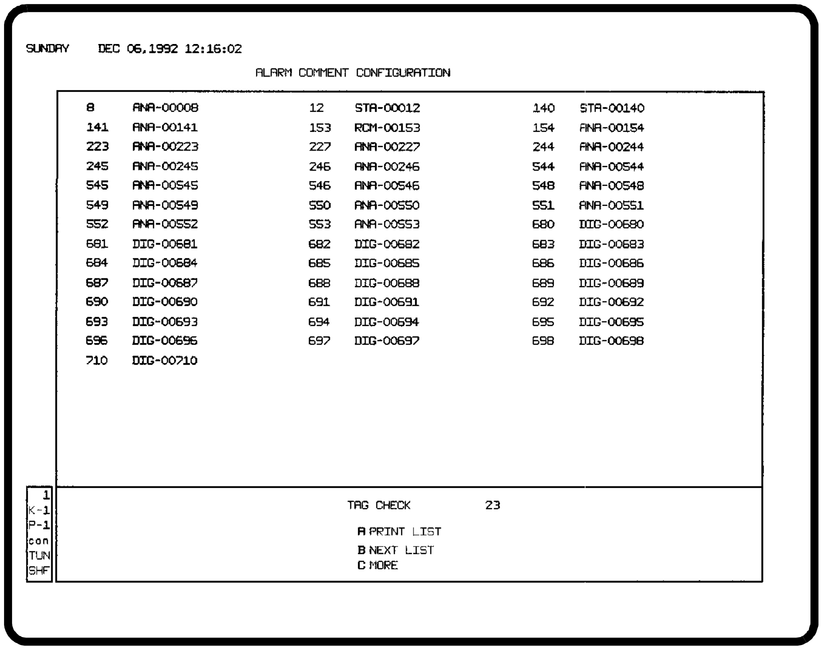
2. Enter the index number of a comment, not a tag index number.

3. Press **ENTER**. The next page to appear gives a list of tags that use the selected comment. Each entry shows the index number and name of a tag. Refer to Figure 6-3 for a tag check page example.

When the tag check page for a comment is on the screen:

- Select *A PRINT LIST* to print the current tag list to a printer. The keyboard status block shows the printer at which a printout will occur.

- Select *B NEXT LIST* to call the *INDEX* field back to the screen to initiate another tag check.



TPS0222B

Figure 6-3. Alarm Comments Page - Tag Check Option

- Select *C MORE* to view additional tags if the list spans more than one page.

Using the Print List Option

The *E PRINT LIST* option specifies a range of alarm comments to print. The keyboard status block shows the printer at which a printout will occur. To use this option:

1. Press **E** to select *PRINT LIST*. The page presents two input fields:

   START: ___

   END: __

2. Enter the index number of the first comment in the range in the *START* field.

3. Enter the index number of the last comment in the range in the *END* field. If only one comment is to be printed, enter the same index number in both fields.

4. Press **ENTER** to initiate the print.

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

Engineering unit descriptors relate to tag types that present analog values. They describe the unit of measurement (e.g., DEG F, GPM, AMPS, LB/HR) for a value received from the process. A descriptor follows the tag value throughout all 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 6-17 lists the fixed engineering unit descriptors and their index numbers. The fixed descriptors can be modified by using the text substitution function. Refer to ***TEXT SUBSTITUTION*** in Section 16 for more information.

Table 6-17. Engineering Unit Descriptors

Index	Descriptor	Index	Descriptor
0	(blank)	8	GPM
1	(blank)	9	CFS
2	%	10	CFM
3	DEG F	11	LB/HR
4	DEG C	12	GAL
5	PSIA	13	AMPS
6	PSIG	14	IN HG
7	IN H2O	15	KLB/HR

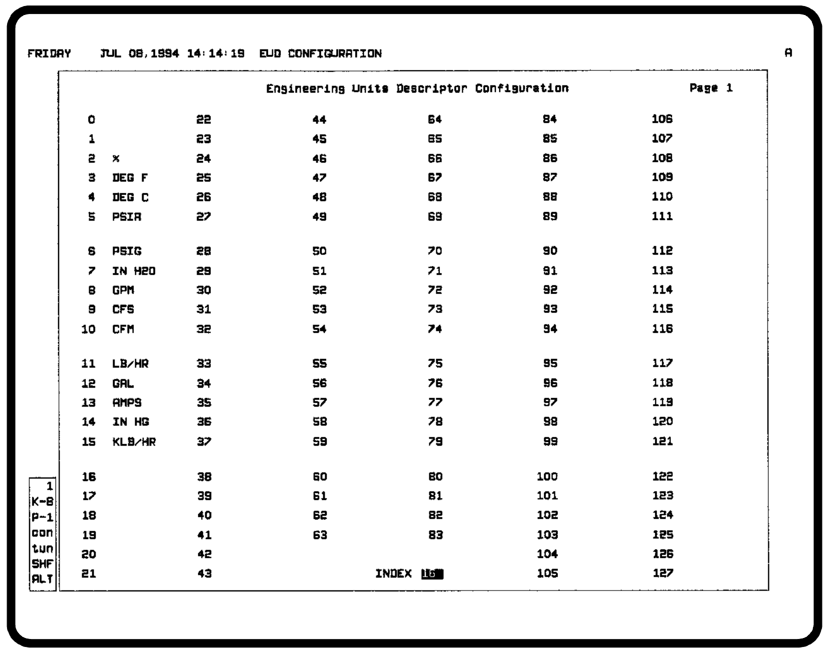
The actual reporting of engineering unit descriptors comes from PCU modules. After an engineering unit is defined in the module configuration, the module sends an index number to the console to identify the engineering unit associated with an exception reported value. This index number is then cross referenced with the database list of descriptors. Since the modules report the engineering unit descriptor index number, 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 or off-line by using the console configuration utilities program.

Defining an Engineering Unit Descriptor

Use the *Engineering Units* option to define an engineering unit descriptor. Figure 6-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *D Engineering Units*



TPS0015C

Figure 6-4. Engineering Units Page

There are two pages of descriptors. Page one shows descriptors zero through 127; page two shows descriptors 128 through 255. Use **NEXT PAGE** and **PREV PAGE** to move between the pages. New descriptors can be added or existing descriptors can be edited through these pages.

NOTE: Descriptors zero through 15 are fixed and cannot be changed through this configuration.

To define an engineering unit descriptor:

1. Select a descriptor to define.

   a. Initially, 16 is entered in the *INDEX* field. This is the first user-definable descriptor. Either use this number or enter another number from 16 to 255.

   b. Press **ENTER**. This moves the input cursor to the specified descriptor.

2. Type a new engineering unit descriptor (six characters maximum) or edit the existing descriptor.

3. Press **ENTER**.

After the input cursor is on a descriptor field, it can be moved to any other using the configuration keys (Table 2-1). Press **NEXT PAGE** to view the second page of descriptors to make changes if desired.

Logic State Descriptors

Logic state descriptors relate to tag types that present logic states for digital devices. These descriptors show the current logic state (e.g., 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 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 6-18 lists the fixed logic state descriptors and their index numbers. The fixed descriptors can be modified by using the text substitution function. Refer to ***TEXT SUBSTITUTION*** in Section 16 for more information.

Table 6-18. Logic State Descriptors

Index	Descriptor	Index	Descriptor
0	ZERO	8	LOW
1	ONE	9	HIGH
2	ON	10	EMPTY
3	OFF	11	FULL
4	NO	12	RUN
5	YES	13	STOP
6	CLOSED	14	TRIP
7	OPEN	15	(blank)

Additional logic state descriptors can be added to the database by using the *Logic State Descriptors* option or off-line by using the console configuration utilities program.

Defining a Logic State Descriptor

Use the *Logic State Descriptors* option to define a logic state descriptor. Figure 6-5 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

↳ *A Database*

↳ *C Logic State Descriptors*

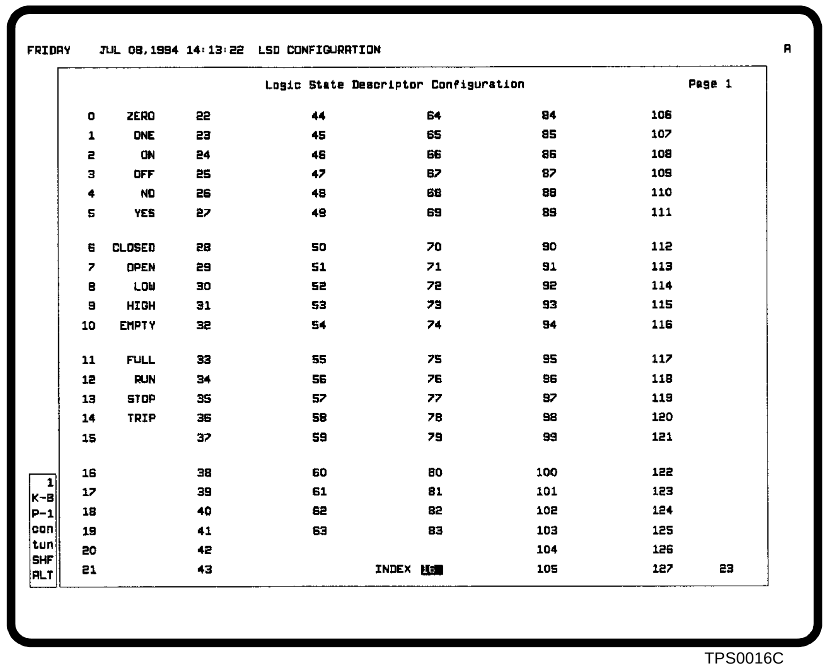


Figure 6-5. Logic State Descriptors Page

There are two pages of descriptors. Page one shows descriptors zero through 127; page two shows descriptors 128 through 255. Use **NEXT PAGE** and **PREV PAGE** to move between the pages. New descriptors can be added or existing descriptors can be edited through these pages.

> **NOTE:** Descriptors zero through 15 are fixed and cannot be changed through this configuration. LSD 15 is defined as blank.

To define a logic state descriptor:

1. Select a descriptor to define.

   a. Initially, 16 is entered in the *INDEX* field. This is the first user-definable descriptor. Either use this number or enter another number from 16 to 255.

   b. Press **ENTER**. This moves the input cursor to the specified descriptor.

2. Type a new logic state descriptor (six characters maximum) or edit the existing descriptor.

3. Press **ENTER**.

After the input cursor is on a descriptor field, it can be moved to any other using the configuration keys (Table 2-1). Press **NEXT PAGE** to view the second page of descriptors to make changes if desired.

SECTION 7 - DISPLAY GENERATION

INTRODUCTION

This section explains two methods available for creating complete displays and symbols used in displays: graphical display configuration (GDC) and elementary line editor (ELE). It also gives the procedures to process a display or symbol source file and to view errors encountered during this processing.

A variety of summary, interactive, and informational displays are used to convey information about the process and results of control actions. The different types of displays include summaries, graphic overviews, graphic details, and group displays. These displays can be assigned to tags, keyboard keys, and ADP pushbuttons.

A complete display is normally built by drawing shapes, static text, and dynamic elements directly in a display file. Additionally, symbols that are used repeatedly are created once and stored in symbol files (***DL***) separate for display source files (***DU***). These symbol files can then be inserted as complete entities into a display.

A faceplate symbol is used to mimic process devices such as stations, device drivers, multi-state device drivers, etc. A set of standard faceplate symbols is provided. These symbols can be inserted into displays as is or they can be modified. Also, these standard faceplate symbols are assigned for use in operator configurable displays.

Displays and symbols are not console specific. A display on one console can be transferred and used on another console.

CONSOLE CONFIGURATION UTILITIES

The preferred method for creating a display or symbol is to use a configuration tool called graphical display configuration (GDC). The GDC utility is available in the console configuration tools (WLDG) program which runs on a Elsag Bailey engineering work station. The GDC utility is an interactive display editing program for creating displays, faceplate symbols, and other reusable symbols such as pumps, tanks, and valves. This configuration tool can be used to create new displays or to edit existing displays. Like the console, the console configuration utilities program provides a set of standard display templates and symbols. These too can be used as is or modified.

Displays and symbols created by using the GDC utility reside in display source files (***DT***). After a display source file is downloaded to the console, the *Display Generator* function must be used to assemble and store the file as a ***DU*** file (i.e., display) or ***DL*** file (i.e., symbol file). File utilities can be used to transfer unassembled ***DT*** files to the console for storage.

A feature available with the console configuration utilities program and enabled from the console is display broadcasting. Broadcasting allows the console to receive a display source file (***DT***) over the communication highway (INFI-NET system only) from an engineering work station running the console configuration utilities program. The engineering work station initiates the transfer, not the console. A display file is automatically run through the *Display Generator* function after being received. The new display is then available for call-up. The operations performed by the console on received display files are transparent.

ELEMENTARY LINE EDITOR

The console elementary line editor function can be used to create or edit a display. The editor is accessed while in the DDT function. The ELE function is better suited for display editing than it is for display creation.

When using the editor, a display source file (***DT***) is created by entering commands that define each element or enable each feature of the display. These commands define display interactives, static data, dynamic data and symbols, control points, touch points, key selects, etc. Each line in the source file contains a single escape command. Refer to the ***Display Builder Reference*** instruction for a listing and explanation of display and symbol commands and for the format that must be followed when creating a display or symbol file.

Some prior work is necessary when using the ELE function to build a display. Each element of a display must first be mapped out with positional x,y coordinates. After the coordinates are known for a particular element, they can be used in the command that enables the element. Refer to the discussion on the display system in the ***Operation*** instruction for further explanation of the x,y coordinate system.

The source file (***DT***) for a display or symbol can reside either on the hard disk or on a floppy disk. Displays created with the console configuration utilities program and transferred to the console can be edited with the elementary line editor. Display files created with the elementary line editor must be assembled by using the *Display Generator* function before they can be used in operations.

SYMBOL LIBRARY

Normally, when a display is called up, the console must search for, open, read, and close individual symbol files (***DL***) for every symbol used in a display. This takes time, especially if a display contains many symbols. The symbol library can be used to reduce the time it takes to call displays. The symbol library contains a list of all symbols that are part of displays. The advantage to using the symbol library is that the files that create a complete display are read from a single library file.

Anytime a new symbol file is loaded onto the console or an existing symbol file is modified, it must be added to the symbol library. When calling a display, the console searches the symbol library first. If a symbol is not in the library, the console searches its display directories.

Symbols used in displays can be added to the symbol library using the **SYMLIB** command. Refer to the ***File Utilities*** instruction for the procedures to use the symbol library utility. A symbol is automatically added to the symbol library when the *Display Generator* or the **PROCDT** or **DOT** commands are used to assemble the symbol source file (***DT***).

DISPLAY CAPABILITIES

There is a limit to the number of tags that can concurrently be displayed on all console windows. The limit is 1,600 tags. This limit implies a maximum of 200 tags per display; however, a single display can actually contain up to 400 tags. A display will not be called if its tag count added to the tag count of all currently active displays exceeds this maximum. The following message appears when a display call-up attempts to exceed the maximum:

Display/Window not called over 1600 tags

Determining the Tags per Display Count

To determine the tag count for a display, count the tags used for all display elements. This includes those used in pop ups. Count multiple instances of the same tag in a display as one tag.

Determining the Tags per Console Count

To determine the total tag count for all active displays, add the count of each display for all windows even if the same display appears on multiple windows.

Determining the Elements per Tag Count

Each tag can have up to 255 escape configuration (**ec**) elements per display. Each tag can have up to a total of 255 escape dynamics (**ed**) and escape alarm (**ea**) elements per display. Even if a single tag is referenced in several elements of a display, it counts as only one tag. Refer to the ***Display Builder Reference*** instruction for an explanation of the elements (Table 1-2 lists instruction numbers).

DISPLAY FILE SIZE

The file size limit for an assembled display (***DU***) is 64 kilobytes. An error will be indicated after using the *Display Generator* function if a display has too many elements and exceeds this limit:

Unable to build file

- ***or*** -

Output Buffer Overflow. File too large

The error message will appear in an error file. Refer to ***DISPLAY ERRORS*** in this section for an explanation. All elements of a display have different size requirements; therefore, a maximum number of display elements per display cannot be determined.

NOTE: File sizes are measured in 500-byte blocks when viewed using OpenVMS™ utilities. Two blocks is equal to one kilobyte. For example, a file size measured as 128 blocks is 64 kilobytes. The DDT function shows file sizes measured directly in bytes.

DISPLAY GENERATOR

Use the *Display Generator* option as the last step in display creation. All displays or symbols, whether created with the SODG utility or with the elementary line editor, must be processed using the *Display Generator* option before they can be called on the console.

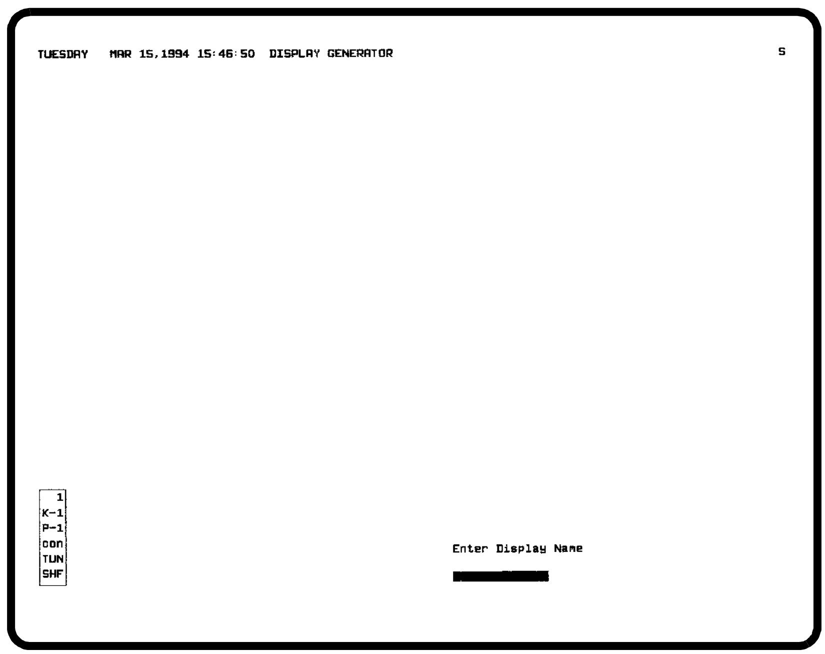
NOTE: The **DOT** and **PROCDT** commands perform the same function as the *Display Generator* option. If using either of these commands, processing a display with the *Display Generator* option is not required. Refer to the ***File Utilities*** instruction for an explanation of these commands (Table 1-2 lists instruction numbers).

The display generator takes a ***DT*** source file, assembles it into a ***DU*** display file or ***DL*** symbol file, and transfers it to one of several directories specifically for assembled display or symbol files ([DATA.USN04] to [DATA.USN0E]). The console automatically determines to which directory a file stores. These directories are checked when a display is called by name, ADP pushbutton, or keyboard key. It also references these directories when adding a faceplate symbol, trend element, or alarm summary element to an operator configurable display. When assembling a display, the display source file and any symbol source files it uses must reside on the hard disk or floppy disk.

NOTE: This procedure is not required when broadcasting a display source file to the console from an engineering work station. Any source files received in a broadcast are automatically run through the display generator.

Use the *Display Generator* option to process a display source file (***DT***). Figure 7-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Display*
→ *A Display Generator*



TPS0054B

Figure 7-1. Display Generator Page

To process a display file:

1. Enter the name of a display file without its extension in the *Enter Display Name* field. Wild card characters (i.e., * and ?) can be used to process several files or all files that follow a specific name pattern.

Unassembled display or symbol source files, whether transferred from an engineering work station, created with the elementary line editor, or provided with the software, normally reside in the [DATA.USN54] directory. To process a display file that resides in this directory, type just the file name. A directory name is not needed.

If the display file is in another directory, specify that directory name. For example, a display file resides in the [DATA.USND1] user directory. In the input field, type:

0.D1:*filename*

NOTE: This procedure overwrites existing ***DU*** display files.

2. Press ENTER. Successful file processing is indicated by the file name appearing with a *DU* extension. The file name appears with an asterisk (*) beside it and a *DT* extension (**filename.DT*) if an error was encountered during processing. Error messages for the file can be viewed by using the *Show Display Errors* function.

3. When all file processing is complete, the message *Completed* appears. Wait until all processing is completed before exiting the page.

Floppy Disk

To process a display file that resides on floppy disk:

1. Transfer the display file (***DT***) to the [OIS.CONFIG] directory. Refer to the ***File Utilities*** instruction for the procedures to transfer a file from floppy disk.

2. Use the **XLATEDT** utility to transfer the display file from the [OIS.CONFIG] directory to the [DATA.USN54] directory. This also translates the file into an OIS usable format. Refer to the ***File Utilities*** instruction for information about the **XLATEDT** utility.

3. Process the display file with the *Display Generator* as previously described.

DISPLAY ERRORS

Errors encountered during display file processing can be viewed on the console. A file in error is identified as **filename.DT* on the display generator page. The *Show Display Errors* option is used to view the errors.

Display or symbol file errors are saved in an error file. An error file can be identified by a ***ER*** extension. The name of an error file is the same as the name of the original display source file. For example, the error file for a source file named ***DISPL1.DT*** will be ***DISP1.ER***.

Use the *Show Display Errors* option to view the source file errors for a display. Figure 7-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Display*
→ *D Show Display Errors*

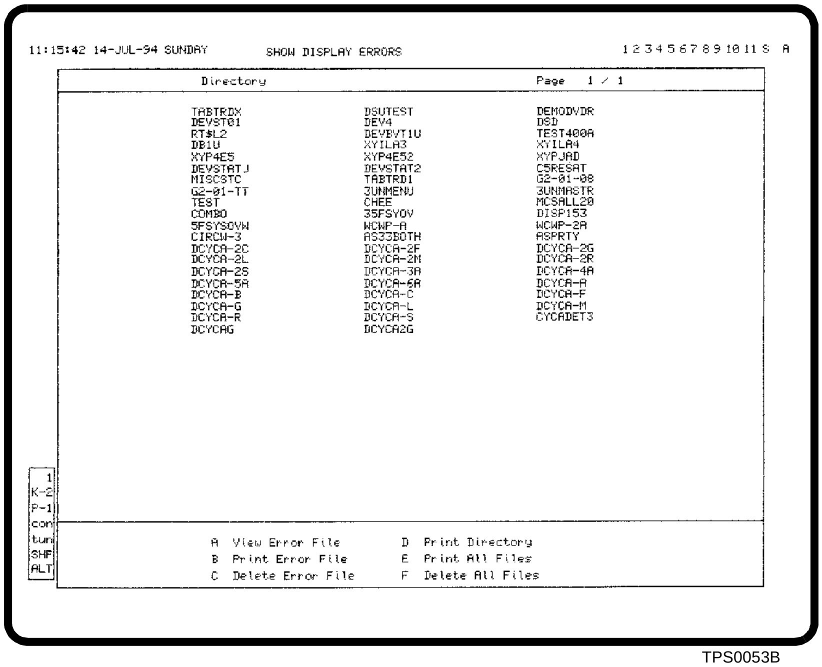


Figure 7-2. Show Display Errors Page

After viewing the errors and correcting them, the source file can be run through the display generator again.

Viewing an Error File

The show display errors page lists the names of all error files. A file name appears without its ***ER*** extension. The name that appears in the list is the same as the name of the source file in which the display generator function encountered an error.

Use the *View Error File* option to call and view the contents of an error file (Figure 7-2). An error file contains descriptions of all errors encountered by the display generator function.

To view the contents of an error file:

1. Select *A View Error File* from the directory options menu. An *Enter File Name* field appears.

2. Type the file name as it appears in the directory listing. Do not include the ***ER*** extension. For example, to view the errors encountered while assembling the display source file ***DISP1.DT***, enter **DISP1**.

3. Press **ENTER**. Press **NEXT PAGE** or **PREV PAGE** to view the entire file contents.

4. Press **ESC** to exit the page. Exiting leaves the error file intact for future viewing.

Two options at the bottom of the page allow printing or deleting the error file contents: *A Print Current File* and *B Delete Current File*.

Select *A Print Current File* to print the error file to a printer. The print option directs a printout to the printer currently assigned to the keyboard. Look at the keyboard status block to determine which printer.

Select *B Delete Current File* to delete the error file from the directory listing and the hard disk.

Printing an Error File

Use the *Print Error File* option to print the errors listed in a single error file (Figure 7-2). A print option is also available after calling an error file by using the *View Error File* option.

To print the contents of a single error file:

1. Select *B Print Error File* from the directory options menu. An *Enter File Name* field appears.

2. Type the file name as it appears in the directory listing.

3. Press **ENTER**. The entire contents of the file prints.

Use the *Print All Files* option to print the contents of all error files listed in the directory.

The print options direct a printout to the printer currently assigned to the keyboard. Look at the keyboard status block to determine which printer.

Deleting an Error File

Use the *Delete Error File* option to delete an error file when no longer required (Figure 7-2). Error files should be deleted from the hard disk to free hard disk space. A delete option is also available after calling an error file by using the *View Error File* option.

To delete a single error file:

1. Select *C Delete Error File* from the directory options menu. An *Enter File Name* field appears.

2. Type the file name as it appears in the directory listing.

3. Press ENTER.

Use the *Delete All Files* option to delete all error files listed in the directory.

Printing an Error File Directory Listing

Use the *Print Directory* option to print a directory listing (Figure 7-2). The same file names that appear on the page appear in a printout. The option directs a printout to the printer assigned to the keyboard. Look at the keyboard status block to determine which printer.

ELE DISPLAY EDITING

Access to the elementary line editor (ELE) is through the DDT function. At the *$* prompt of a terminal window, type:

DDT ENTER

The ELE function can be used to create a new display or symbol source file (***DT***). It can also be used to call and edit an existing display source file. These files reside in the [DATA.USN54] directory by default, on floppy disk, or in other user directories.

NOTE: Access to DDT functions can be limited through password security. This prevents unauthorized editing of display and command files.

To create a display file:

1. Change to the [DATA.USN54] directory, floppy disk, or a user directory. For example, type:

FD00 54 ENTER

2. Create the display source file by using the **ASF** file allocation command. The file name must have a ***DT*** extension. For example, type:

ASF . AREA1.DT 50 ENTER

3. Start the elementary line editor to open the file created in Step 2. For example, type:

ELE . AREA1.DT ENTER

4. Enter all applicable escape commands to create the display by using the editing commands of the elementary line editor.

5. Exit and save the file by typing:

.X **ENTER**

6. Use the *Display Generator* function to process and assemble the display.

Refer to the ***File Utilities*** instruction (Table 1-2 lists instruction numbers) for the procedures to use the DDT function, an explanation of the command used to create and allocate files (**ASF**), and an explanation of the editing commands of the elementary line editor (i.e., list, edit, delete, insert, save, and exit).

Refer to the ***Display Builder Reference*** instruction for an explanation of required formats for display and symbol files and for an explanation of the graphic and escape commands used to create the different features and elements of a display.

Refer to ***DISPLAY GENERATOR*** in this section for procedures to process and assemble displays and symbols for console use.

NOTE: If a standard display template or symbol is to be modified, it is suggested that the original file be first copied and renamed. Make the modifications to the new, renamed file rather than the original.

SECTION 8 - KEYBOARD AND ADP

INTRODUCTION

This section explains how to define key macros, and how to assign displays, key macros, and user tasks to keyboard function keys and annunciator display panel (ADP) pushbuttons.

- Key macros incorporate multiple keystrokes into a single key.
- Assigning a display to a key or pushbutton allows calling the display with a single key or pushbutton press.
- Assigning a user-written program (i.e., user task) to a key or pushbutton allows activating that program with a single key or pushbutton press. This provides an easier method of activation than having to open a terminal window and entering the **EXE** command.

KEY MACROS

Key macros incorporate multiple keystrokes into a single key press. Up to 96 macros can be defined, with each macro containing up to 50 keystrokes. After pressing a keyboard key or ADP pushbutton which has a macro assigned, each action specified in the key macro is performed.

A key macro can include any key sequence that is built into a display and most keyboard keys. Also, any ADP pushbutton can be defined in a key macro. Each macro can contain a maximum of five display call-ups and selects. After being defined, a macro can be assigned to a function key and an ADP pushbutton.

Macros also accept nested definitions. If a macro contains a key or ADP pushbutton that has a macro already assigned, that macro executes as a part of the macro. There is no limit to the number of nesting levels possible. Care should be taken, however, when creating nested key macros. Make sure the nested macro is not defined in such a way that it causes a continuous loop to be executed.

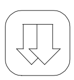
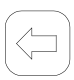
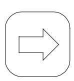
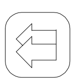
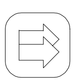
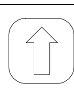
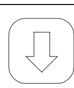
NOTE: A key macro should ***not*** be activated while a display has an element selected for control. Deactivate control before activating the macro.

Key Macro Configuration Keys

Key macros can incorporate any ADP pushbutton and most keyboard keys except those used in macro configuration. Also,

COM'D LINE MENU, **MISC MENU**, and **SWITCH CRT** cannot be used in a key macro. Table 8-1 describes the keys that are not available for macros and their function for macro configuration.

Table 8-1. Key Macro Configuration Keys

Key	Description
CLEAR	Erases the current macro definition and returns the input cursor to the first or starting input field of the macro.
HOME	Calls the *Enter Macro Number* field for selecting a macro to define or edit.
TAB	Moves the input cursor to the first input field in the next macro definition. If the next macro definition is not displayed, the next page is brought up. This does not save the macro from which the input cursor was tabbed.
TAB BACK	Moves the input cursor to the first input field in the previous macro definition. If the previous macro definition is not displayed, the previous page appears. This does not save the macro from which the input cursor was tabbed.
	Performs the same function as **TAB BACK** except that it also saves the macro definition being exited.
	Performs the same function as **TAB** except that it also saves the macro definition being exited.
	Moves the input cursor within the current macro definition to the previous key location. Use this to move to a specific key definition, then delete or overwrite that definition.
	Moves the input cursor within the current macro definition to the next key location. Use this to move to a specific key definition, then delete or overwrite that definition.
	Moves the input cursor within the current macro definition to the previous key location and deletes the key definition from which the cursor was moved. Use this to delete a key.
	Moves the input cursor within the current macro definition and inserts an *undef* key definition. Use this to insert a key between existing keys.
	Moves the input cursor to the previous macro definition making it the currently selected macro. This calls the previous page if pressed while on the upper macro definition.
	Moves the input cursor to the next macro definition making it the currently selected macro. This calls the next page if pressed while on the lower macro definition.

NOTE: The configuration keys are located in the cursor control, alphabetic characters, and numeric keypad blocks of the keyboard, not the station and remote control block.

Defining a Key Macro

Use the *Macros* option to define a key macro. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

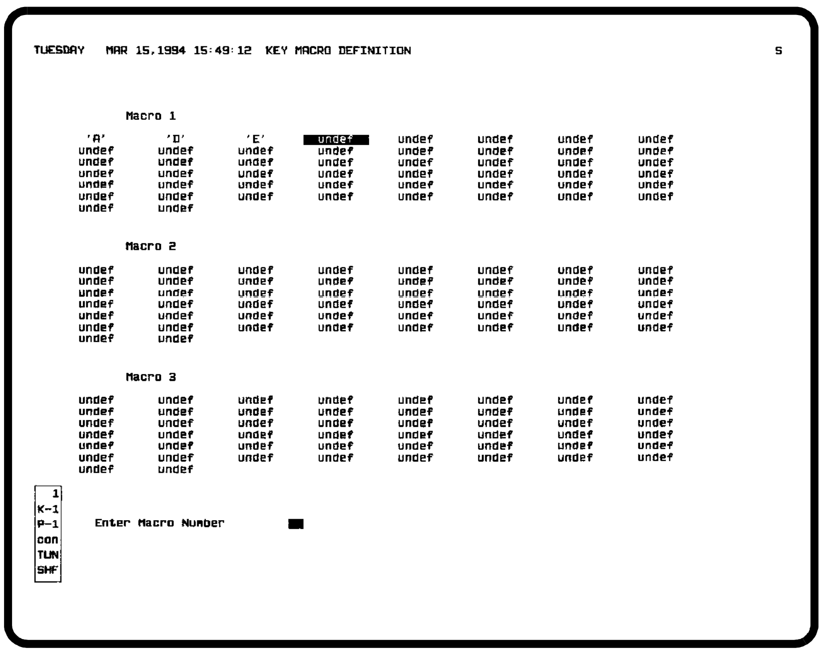
A OIS Configuration
→ *D Keyboard*
→ *C Macros*

To define a macro:

1. Select a macro to define.

   a. In the *Enter Macro Number* field, enter a number from 1 to 96.

   b. Press **ENTER**. Figure 8-1 shows an example of the next page that appears.



TPS0224B

Figure 8-1. Macros Page

2. Use the macro configuration keys (Table 8-1) when defining a macro. Macros are processed from left to right, top to bottom. Up to 50 keys can be incorporated into a single key macro. Enter a macro sequence by pressing keyboard keys and ADP pushbuttons in the order that they are to be performed.

   NOTE: **COM'D LINE MENU**, **MISC MENU**, and **SWITCH CRT** cannot be used in a key macro.

When a key or pushbutton is pressed, the *undef* key definition is replaced by an abbreviated key name in brackets or a character enclosed in single quotes. For example, *<NXT PAG>*, *<GEN FUN>*, *<F14>*, *'7'* or *'*'*.

3. Save the macro definition by either moving from the macro with the double-up or double-down arrow keys or press **HOME** Repeat the steps to define additional macros.

4. When macro definition is complete, press **HOME** then **ESC** to exit.

Moving to another macro by using the double-up and double-down arrow keys or pressing **HOME** saves a macro. As soon as a macro is saved, it is available for use. A macro number is used to assign a macro to a keyboard key or ADP pushbutton.

Editing a Key Macro

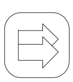
A key macro can be edited at any time. Edits can include deleting, inserting, or changing a key in the macro. To call a key macro for editing:

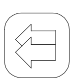
1. Press **HOME** to call the *Enter Macro Number* field if the field is not already present. This input field appears when the configuration page is first called.

2. Call the macro by entering its macro number.

3. Press **ENTER**.

The following macro configuration keys (Table 8-1) are used to edit a macro:


Inserts an *undef* key entry into the macro sequence.


Deletes a key entry from the macro sequence.

To insert a key into a key macro:

1. Use the macro configuration keys to highlight the key entry that is to occur immediately before the key to be inserted.

2. Press the double-right arrow key. This inserts an *undef* key definition into the list.

3. Highlight the *undef* field.

4. Press the key that is to be inserted. This replaces the *undef* with that key.

To delete a key in a key macro:

1. Use the macro configuration keys to highlight the key entry to delete.

2. Press the double-left arrow key. The key entry will no longer appear and the subsequent key entries will move back one position in the macro.

To change a key in a key macro:

1. Use the macro configuration keys to highlight the key entry that is to be changed.

2. Press the key to which it is to be changed.

Deleting a Key Macro

To delete a key macro:

1. Move to the macro by using the *Enter Macro Number* field. Optionally, press **TAB** or **TAB BACK** to move to a macro (Table 8-1).

2. Press **CLEAR**.

3. Save the change by either moving away from the macro with the double-up or double-down arrow keys or press **HOME**. Press **ESC** prior to any other key to cancel the deletion.

KEYBOARD

Any assembled display (***DU***) can be assigned to a function key either directly or through a multiwindow display list. A multiwindow display list defines up to eight displays that are to appear on specific windows when a function key is pressed. Refer to ***MULTIWINDOW DISPLAYS*** in Section 15 for the procedures to define a multiwindow display list.

Any key macro or user task can also be assigned to a function key. Before a key macro can be assigned, however, it must have been previously defined. Refer to ***KEY MACROS*** in this section for macro definition procedures. A user task also must have been previously defined before it can be assigned. Refer to ***@aGlance/IT CONFIGURATION AND DEFINITION*** in Section 19 for @aGlance/IT definition procedures, and ***SECUTI USER TASK DEFINITION*** in Section 20 for user task definition procedures.

Assignable keys include:

32 function keys (e.g., **AREA n** or **Fn**.

DISPLAY SUMM

ALARM SUMM

SYSTEM STATUS SUMM

HELP

NOTE: Use **SHIFT** in combination with the function keys (i.e., **F1** through **F16**) to access function keys 17 through 32. For example, press **SHIFT**-**F1** to access F17.

Making a Function Key Assignment

Use the *Function Keys* option to make a key assignment. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

→ *D Keyboard*

→ *A Function Keys*

Key assignments are specific to a keyboard. After selecting the *Function Keys* option, a *KEYBOARD* field appears. To configure function keys for a keyboard:

1. Call the configuration for a specific keyboard.

   a. Enter the number of a keyboard from 1 to 8 in the *KEYBOARD* field.

   b. Press **ENTER**. The next page shows the current assignments for that keyboard (Figure 8-2). If no previous assignments were made, each key contains a default configuration. This default causes a function key press to call a display named *BLANK*.

2. Three input fields for a function key define its assignment:

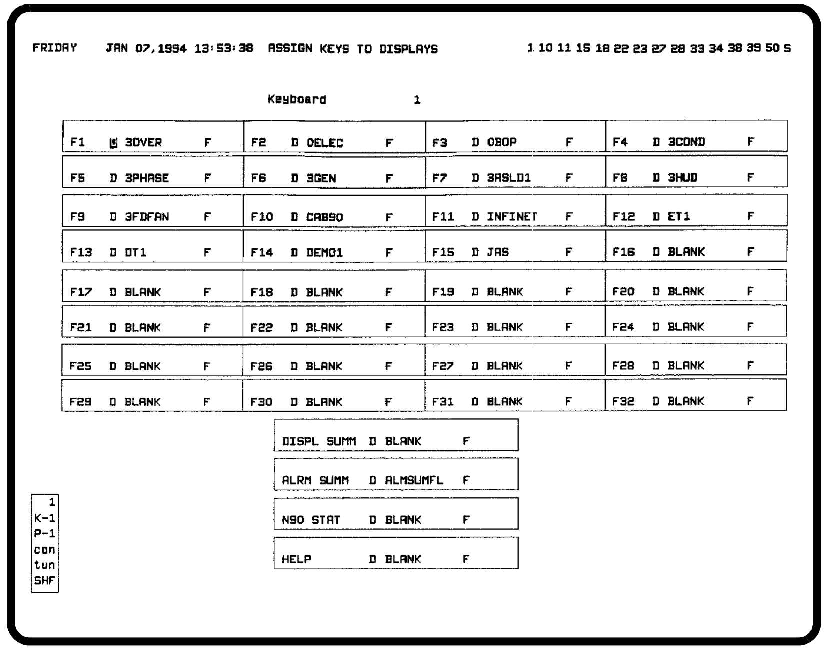
Field one - selects display call-up, key macro activation, or user task activation.

Field two - identifies a specific display, multiwindow display list, key macro, or user task.

Field three - specifies a window assignment.

Move to a specific key and define its fields. Table 8-2 describes how the key fields should be defined depending on the desired function.

3. Continue to define all keys.



TPS0117B

Figure 8-2. Function Keys Page - Definition Page

Table 8-2. Function Key and ADP Assignments

Function	**Field**		
	1	**2**[1]	**3**
Key macro activation	M	1 to 96 = key macro index number	Blank or leave at default
Multiwindow display call-up	L	1 to 256 = multiwindow display list index number	Blank or leave at default
Single display call-up	D	Display name (assembled display file without ***DU*** extension)	0 = all windows supported by this console 1 to 8 = specified window only. For example, enter **1** for window one only F = window assignment follows the keyboard assignment
User task activation	A	1 to 50 = user task index number	Blank or leave at default

NOTE:

1. An index number that is undefined can be assigned. Pressing the key or ADP pushbutton will have no affect, however.

4. Press **ENTER**.

NOTE: The default alarm summary display name is ALMSUMFL. The default system status display name is N90STAT1.

Clearing a Function Key Assignment

To delete a key assignment, move to the specific key and enter the following information in the three key fields:

D
BLANK
F

This causes the key to display the default ***BLANK.DU*** display when pressed.

ANNUNCIATOR DISPLAY PANEL

Any assembled display (***DU***) can be assigned to an ADP pushbutton either directly or through a multiwindow display list. A multiwindow display list defines up to eight displays that are to appear on specific windows when a pushbutton is pressed. Refer to ***MULTIWINDOW DISPLAYS*** in Section 15 for the procedures to define a multiwindow display list.

Any key macro or user task can be assigned to an ADP pushbutton. Before a key macro can be assigned, however, it must have been previously defined. Refer to ***KEY MACROS*** in this section for macro definition procedures. A user task also must have been previously defined before it can be assigned. Refer to ***@aGlance/IT CONFIGURATION AND DEFINITION*** in Section 19 for @aGlance/IT definition procedures, and ***UTI USER TASK DEFINITION*** in Section 20 for user task definition procedures.

An ADP lamp assignment is made through tag database configuration. An ADP pushbutton assignment should coincide with this lamp assignment. This allows calling a display related to an alarm (indicated by the lamp being lit red) to perform actions required to correct the alarm.

Making an ADP Pushbutton Assignment

Use the *ADP* option to make a pushbutton assignment. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *D Keyboard*
→ *B ADP*

Pushbutton assignments are specific to an annunciator display panel. After selecting the *ADP* option, the following fields appear:

KEYBOARD ___ *ADP* ___ *LAMP* ___

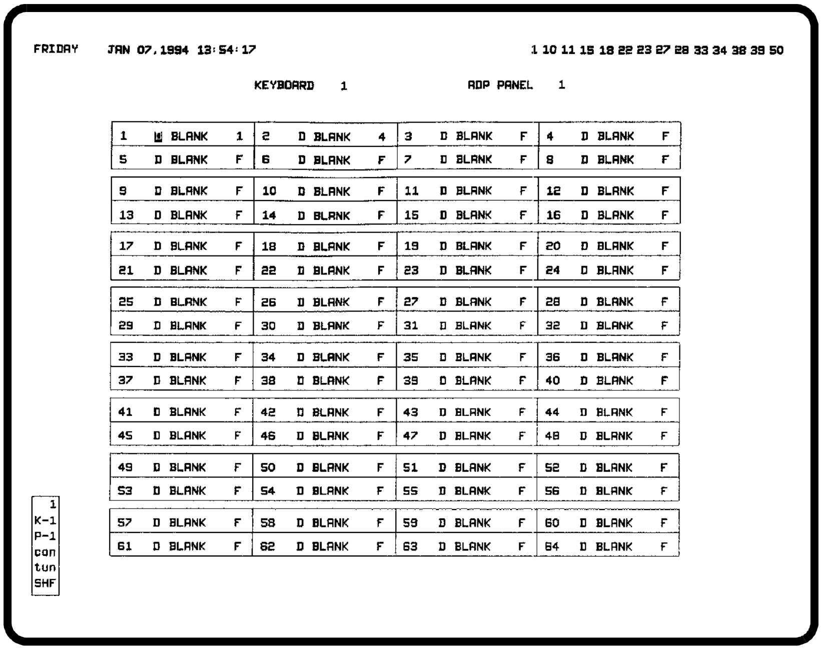
To configure pushbuttons for an annunciator display panel:

1. Call the configuration for a specific panel:

   a. Enter the keyboard number the annunciator display panel is assigned to in the *KEYBOARD* field, ADP number the assignments are being made for in the *ADP* field, and a specific lamp (pushbutton) being assigned in the *LAMP* field.

   NOTE: Pushbutton assignments 33 through 64 are not valid for this console. Attempting to access these assignments causes an error.

   b. Press **ENTER**. This calls the second page of the configuration. The next page shows the current assignments for that panel (Figure 8-3). If no previous assignments were made, each pushbutton contains a default configuration. This default causes a pushbutton press to call a display named *BLANK*.



TPS0119B

Figure 8-3. ADP Page - Definition Page

2. Three input fields for a pushbutton define its assignment:

Field one - selects display call-up, key macro activation, or user task activation.

Field two - identifies a specific display, multiwindow display list, key macro, or user task.

Field three - specifies a window assignment.

Move to a specific pushbutton and define its fields. Table 8-2 describes how the pushbutton fields should be defined depending on the desired function.

3. Continue to define all pushbuttons.

4. Press **ENTER**.

Clearing an ADP Pushbutton Assignment

To delete a pushbutton assignment, move to the specific pushbutton and enter the following information in the three pushbutton fields:

D
BLANK
F

This causes that pushbutton to display the default ***BLANK.DU*** display when pressed.

SECTION 9 - ALARM MANAGEMENT

INTRODUCTION

Alarm management configuration sets standard console indications and responses to alarm conditions for operator ease of alarm processing. Alarm management starts at the PCU module level. The actual alarm limits that trigger a process alarm are set in the control configuration of PCU modules. Alarm limits set at the module level insure the alarm trigger conditions remain the same for any and all consoles.

Through the tag database, individual process points are monitored to trigger alarm indications. This allows managing process alarms from a single point. Alarm management should be considered before creating the tag database since several attributes for a tag pertain to alarming.

Alarm management configuration encompasses the following configurations:

- Alarm quality options.
- Automatic tag inhibiting.
- Global acknowledge and silence.
- Management options.
- Priorities.
- Relays.
- Remote acknowledge.
- Summary format.
- Summary report.
- System configuration (*General Parameters*).
- Tones.

This section discusses alarm management configuration only. Refer to the discussion on alarm management in the ***Operation*** instruction for a description of alarm management operations (Table 1-2 lists instruction numbers).

ALARM MANAGEMENT OPTIONS

Use the *Alarm Management Options* function to:

- Configure the triggering of alarm tones and relays. Tones and relays can be triggered by alarm group or by alarm priority but not both. The default is by alarm group.

- Enable or disable the alarm group indicators at the top of the window. This configuration only affects the top line indicators. All other alarm group indications are not affected. The alarm group indicators are enabled by default.

- Enable or disable the alarm priority indicators along the left edge of the window. This configuration only affects the indicators that appear along the left edge. All other alarm priority indications are not affected. The alarm priority indicators are disabled by default.

- Determine how the alarm priority indicators are to be displayed. The indicators can appear as numeric indicators only or drawn as pushbuttons.

An alarm group indicator identifies an area or group of tags in alarm. An alarm priority indicator identifies the existence of an alarm with a certain priority level.

An alarm priority indicator, whether displayed as a numeric indicator only or as a pushbutton indicator, is a touch point. The touch point allows calling an alarm summary directly for a particular alarm priority level. By default, the summary that is called will contain only alarms with the selected priority level. The touch points can be configured, however, to call up any display desired. Refer to ***ALARM PRIORITIES*** in this section for more information.

Figure 9-1 shows the page used to set alarm management options. To choose the option, first press **GENL FCTNS MENU** then select the following menu items in the sequence shown.

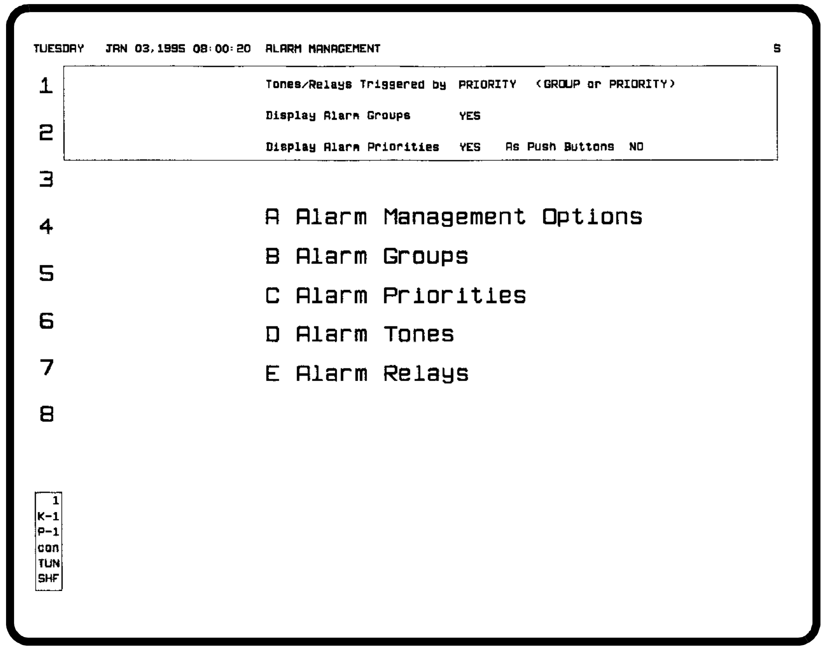
A OIS Configuration
→ *E System*
→ *B Alarm Management*
→ *A Alarm Management Options*

Selecting the option enables the input cursor in the boxed area at the top of the page. To set the alarm management options:

1. In the *Tone/Relays Triggered by* field, enter **GROUP** to have alarm tones and relays triggered by alarm group. Enter **PRIORITY** to trigger by alarm priority. A change to this field does not take effect until the OIS application is reset.

If alarm group triggering is selected, *Alarm Groups* configuration is required. When a tag goes into an alarm condition, the tone and relay defined for the alarm group to which the tag is assigned will be activated. Refer to ***ALARM GROUPS*** in this section for alarm group configuration requirements.

If alarm priority triggering is selected, *Alarm Priorities* configuration is required. When a tag goes into an alarm condition, the tone and relay assigned to the priority of the tag alarm condition will be activated. Refer to ***ALARM PRIORITIES*** in this section for alarm priority configuration requirements.



TPS1101A

Figure 9-1. Alarm Management Page

2. Enter **YES** in the *Display Alarm Groups* field to have the alarm group indicators appear at the top of the page. Enter **NO** to disable the indicators. A change to this field takes effect immediately without requiring a reset.

3. Enter **YES** in the *Display Alarm Priorities* field to have the alarm priority indicators appear at the left of the page. Enter **NO** to disable the indicators. A change to this field takes effect immediately without requiring a reset.

4. Enter **YES** in the *As Push Buttons* field to display the alarm priority indicators as pushbuttons. Enter **NO** to display the indicators as numbers only. A change to this field takes effect immediately without requiring a reset.

> **NOTE:** The *Alarm Management Type* field on the *General Parameters* page must be set to option 1 or 3 when alarm priority touch points (i.e., left edge indicators) are enabled. If not, a touch point will call an alarm summary but the summary will be empty. Refer to ***Setting the Sort Option*** in this section for more information about the *Alarm Management Type* field.

5. If a change was made to the *Tones/Relays Triggered by* field, reset the OIS application by using the procedures given in ***Reset*** in Section 2. A reset does not have to be performed immediately, but a change will not take effect until after a reset.

ALARM GROUPS

The number of points defined in a single process can be substantial. Alarm groups can be used to keep related process tags together in a group for easier management.

NOTE: This type of console does not use the DEVSTAT tag to monitor the status of its peripherals. Alarm group D is considered invalid for alarming purposes even though the console assigns a DEVSTAT tag to this group.

Alarm group configuration requirements include:

- Enabling alarm group triggering of tones and relays (refer to ***ALARM MANAGEMENT OPTIONS*** in this section). This type of triggering is enabled by default.
- Assigning tags to alarm groups.
- Configuring an alarm group. After tags have been assigned to an alarm group, the alarm indications given by the console (alarm tones and relays) for alarms in the different groups must then be configured.

Assigning a Tag to an Alarm Group

Each tag in the database can be assigned to an alarm group. The alarm group assignment is optional, however. There are two ways to assign tags to an alarm group: Tag configuration and alarm group configuration. Alarm group assignments should be considered during initial database configuration.

Alarm group numbers range from 1 to 99. Group S is reserved for system devices. Tags in the database can be grouped by assigning the same alarm group number to each tag.

When any tag in a group enters an alarm condition, a group indicator (e.g., *1* to *99*, or *S*) displays at the upper right corner of the window on all displays. This alarm group indicator allows quickly identifying a problem area in the process. The indicator also appears as part of an alarm status/quality/group field and can be displayed in a separate alarm group field if desired. The line format used in standard alarm summaries contains an alarm status/quality/group field to show alarm group for a tag.

USING TAG CONFIGURATION

Tag configuration can be used to assign a tag to an alarm group. This procedure, however, can be used to make the assignment for only one tag at a time. To define the alarm group for a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter the number of a group in the *Alarm Group* field for a tag. The same number should be used for all tags that are to be part of the same group. A valid entry is from 0 to 99. N90STA tags are automatically assigned to group S and DEVSTAT tags are automatically assigned to group D. Enter **0** to leave the tag unassigned.

USING ALARM GROUP DEFINITION

Alarm group definition can be used to assign an alarm group to a range of tags. To do this, follow the steps given in ***Assigning by Tag Range*** in this section.

Configuring an Alarm Group

Each alarm group is configured separately. The attributes defined for an alarm group affect all tags assigned to that group. Alarm group configuration defines an alarm tone, return-to-normal tone, alarm relay, and return-to-normal relay for an alarm group.

An alarm tone assigned to a group sounds to notify an operator of a tag in that group either entering an alarm condition or returning to its normal condition. An alarm relay assigned to a group closes for either an alarm or a return-to-normal condition and can be used to trigger an external alarm annunciator.

Use the *Alarm Groups* option to configure an alarm group. Figure 9-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

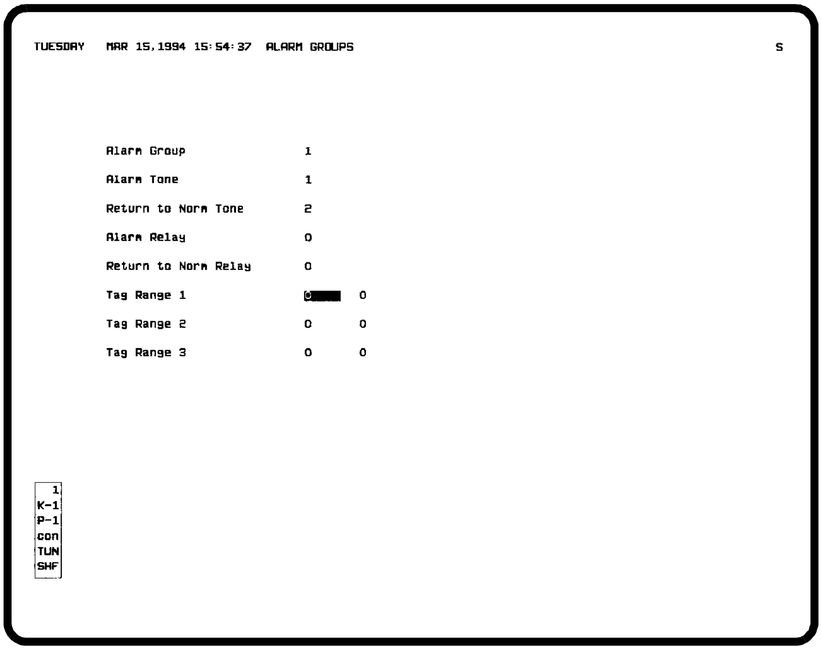
A OIS Configuration
→ *E System*
→ *B Alarm Management*
→ *B Alarm Groups*

To define an alarm group:

1. Select an alarm group to define.

   a. Enter the number of an alarm group in the *Alarm Group* field. A valid entry is from 1 to 99, or S.

   b. Press **ENTER**.



TPS0001B

Figure 9-2. Alarm Groups Page

2. Enter the appropriate data into each field of the page. Table 9-1 explains the alarm group fields. Refer to this table when entering data.

3. Press **ENTER**.

   NOTE: Alarm group assignments for tags do not take effect until the OIS application is reset when using the *Tag Range* fields to make the assignments.

4. Reset the OIS application by using the procedures given in ***Reset*** in Section 2. The alarm tones and relays assigned to the group will not trigger until the OIS application is reset. A reset does not have to be performed immediately, but an alarm tone and relay assignment will not take effect until after a reset.

Table 9-1. Alarm Groups Page Fields

Field	Description
Alarm group	Identifies and calls an alarm group for configuration. A valid group number is: 1 to 99 = process tags. S = alarm group for system tags. The console automatically assigns N90STA type tags to this group. **NOTE**: This type of console does not present alarm indications for DEVSTAT tags in alarm group D.

Table 9-1. Alarm Groups Page Fields (continued)

Field	Description
Alarm tone[1]	Logical tone that is to sound when a tag in this group enters an alarm condition. A valid entry is: 0 = default; no tone assigned. 1 to 20 = logical tone.
Return-to-normal tone[1]	Same as the *Alarm Tone* field except that this tone sounds when a tag in this group returns to its normal condition. A valid entry is: 0 = default; no tone assigned. 1 to 20 = logical tone.
Alarm relay[2]	Logical relay that is to close when a tag in this group enters an alarm condition. A valid entry is: 0 = default; no relay assigned. 1 to 48 = logical relay.
Return-to-normal relay[2]	Same as the *Alarm Relay* field except that this relay closes when a tag in this group returns to its normal condition. A valid entry is: 0 = default; no relay assigned. 1 to 48 = logical relay.
Tag range 1/2/3	Used to expedite initial alarm group assignment or to change existing alarm group assignments in the tag database. Up to three ranges of index numbers can be entered to assign the tags in the range to the current alarm group. A valid entry is: 0 = range not assigned. 1 to 30000 = depends on the maximum number of tags enabled during system configuration. **NOTE:** The configuration that was performed more recently, either tag configuration or this configuration, overrides the previous configuration.

NOTES:

1. Alarm tone configuration determines the duration, pitch, and volume of a tone and selects the keyboard that drives the tone.
2. Alarm relay configuration determines the duration that a relay remains closed, selects the keyboard that drives a relay, and selects a specific physical relay.

Assigning by Tag Range

Use the *Tag Range* fields of alarm group configuration to initially define or to modify alarm group assignments for tags in the database. This can be used to modify the alarm group assignment for all tags that fall within a specified range or ranges. To use the *Tag Range* fields to make or modify alarm group assignments:

1. Perform the steps as described in ***Configuring an Alarm Group*** in this section.

2. In the *Tag Range* fields, define up to three tag ranges. The tag ranges specify which tags in the database are to be assigned to the currently selected alarm group.

For example, if a range is defined as one to 50 and the configuration page shows *15* in the *Alarm Group* field, then tags one to 50 will be assigned to alarm group 15.

3. Press **ENTER** to initiate the alarm group assignment.

4. The alarm group assignments made using the range fields do not take effect until the OIS application is reset. After making all assignments, reset the OIS application by using the procedures given in ***Reset*** in Section 2.

The tag range entries are not retained after exiting the configuration page. The fields reset to *0* to allow defining additional tag ranges if desired.

ALARM PRIORITIES

The number of points defined in a single process can be substantial. Alarm priorities can be used to identify and group alarms by priority for easier management. There are eight priority levels for alarms with priority one being the highest.

Each tag type has a variable number of alarm conditions associated with it. Priority can be assigned to each tag alarm level or the alarm state and to the return-to-normal condition. Also, bad quality when used to trigger alarming can be given a priority.

Alarm priorities configuration requirements include:

- Enabling alarm priority triggering of tones and relays (refer to ***ALARM MANAGEMENT OPTIONS*** in this section). The default is alarm group triggering.
- Assigning priorities to tag alarm conditions.
- Configuring alarm priorities. After assigning a priority to the alarm conditions of tags, the alarm indications given by the console (alarm tones and relays) for alarms in the different priorities must then be configured.

Assigning an Alarm Priority to a Tag Alarm Condition

Each alarm condition of a tag in the database can be assigned an alarm priority level. The alarm priority assignment is optional, however. The priority defaults to level one for all tags if priorities are not being used. Alarm priority assignments should be considered during initial database configuration.

If enabled, a priority indicator (e.g., *1* to *8*) displays at the left edge of the window on all displays when a tag goes into alarm. An alarm priority indication is also given in an alarm priority field if built into a display. The alarm priority field will display a text string that is associated with a priority level. Refer to the alarm management discussion in the ***Operation*** instruction for a listing of these descriptors and the priority level they associate with (Table 1-2 lists instruction numbers). The line format used in standard alarm summaries contains an alarm priority field (e.g., *P1*).

Tag configuration is used to assign priority levels. To define priority levels for the alarm conditions of a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter the number of a priority level in the *PRIORITY* column for each alarm condition of a tag. A valid entry is from 1 to 8.

Configuring Alarm Priorities

Priority configuration defines an alarm tone, return-to-normal tone, alarm relay, return-to-normal relay, and primary display name for each alarm priority level. The primary display is the name of a display to be called when the corresponding alarm priority pushbutton is selected.

An alarm tone assigned to a priority sounds to notify an operator that an alarm with that priority has occurred or an alarm with that priority has returned to its normal condition. An alarm relay assigned to a priority closes for either an alarm occurrence or a return-to-normal condition and can be used to trigger an external alarm annunciator.

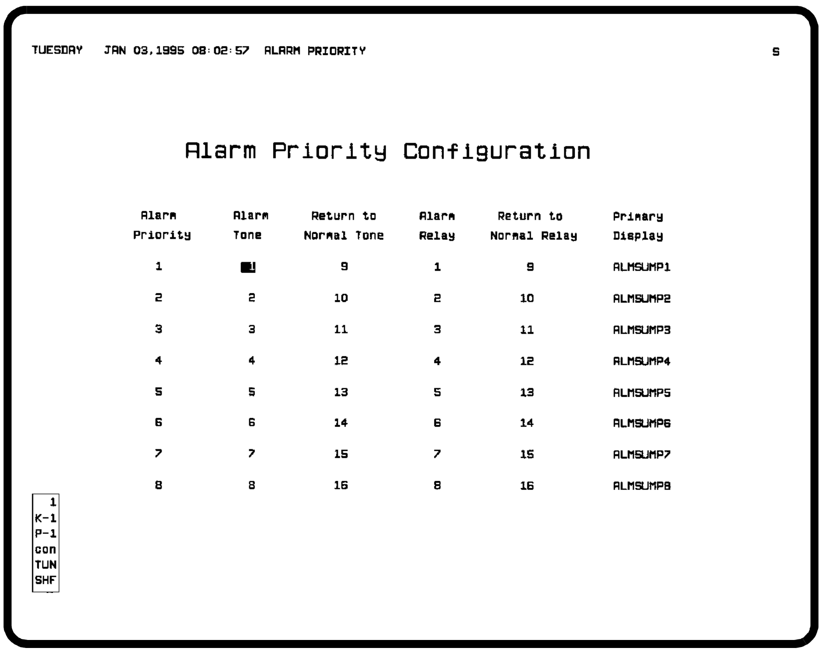
The default displays for each priority are ***ALMSUMPn.DT*** where ***n*** is the priority level from 1 to 8. For example, the display ***ALMSUMP2.DT*** is the priority two alarm summary display. Only priority two alarms will appear in this display.

This configuration also modifies the tone priorities for alarm tones. It defaults the tone priority of the logical tones assigned as the alarm tone for alarm priorities one through eight to a tone priority of one to eight respectively. It also defaults the tone priority of the logical tones assigned as the return-to-normal tones for priorities one through eight to a tone priority of nine to 16 respectively. These defaults can be modified with alarm tones configuration if the defaults are not adequate. Any subsequent change to the alarm priorities configuration, however, will return the tone priorities to the defaults. The configuration that was performed more recently, either alarm priorities configuration or alarm tones configuration, overrides the previous configuration. Refer to ***ALARM TONES*** in this section for an explanation of the affects of logical tone priorities.

Use the *Alarm Priorities* option to configure alarm priorities. Figure 9-3 shows the page used. To choose the option, first

press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *B Alarm Management*
→ *C Alarm Priorities*



TPS1100A

Figure 9-3. Alarm Priorities Page

To define alarm priorities:

1. Enter the appropriate data into each field for the priority being configured. Table 9-2 explains the alarm priority fields. Refer to this table when entering data.

2. Press **ENTER**.

Table 9-2. Alarm Priorities Page Fields

Field	Description
Alarm tone[1]	Logical tone that is to sound when an alarm with this priority occurs. A valid entry is: 0 = default; no tone assigned. 1 to 20 = logical tone. The logical tone number must be unique for each priority. This is required for tone prioritization purposes.

Table 9-2. Alarm Priorities Page Fields (continued)

Field	Description
Return-to-normal tone[1]	Same as the *Alarm Tone* field except that this tone sounds when an alarm with this priority returns to its normal condition. A valid entry is: 0 = default; no tone assigned. 1 to 20 = logical tone. Again, the logical tone number must be unique for each priority. The same logical tone number cannot be used for both an alarm tone and a return-to-normal tone.
Alarm relay[2]	Logical relay that is to close when an alarm with this priority occurs. A valid entry is: 0 = default; no relay assigned. 1 to 48 = logical relay.
Return-to-normal relay[2]	Same as the *Alarm Relay* field except that this relay closes when an alarm with this priority returns to its normal condition. A valid entry is: 0 = default; no relay assigned. 1 to 48 = logical relay.
Primary display	Assigns a display to the priority that is to be called when the alarm priority indicator touch point is selected. Alarm management options configuration is used to configure the indicators. The default displays are built to show only the alarms with the priority selected: Priority one = ALMSUMP1 Priority five = ALMSUMP5 Priority two = ALMSUMP2 Priority six = ALMSUMP6 Priority three = ALMSUMP3 Priority seven = ALMSUMP7 Priority four = ALMSUMP4 Priority eight = ALMSUMP8 Any assembled display (***DU***) can be used in place of a default display. The primary display for a priority can be left blank. If left blank, the pushbutton for that priority will not function.

NOTES:

1. Alarm tone configuration determines the duration, pitch, and volume of a tone and selects the keyboard that drives the tone for each logical tone. Physical tones need not be unique since the same physical tone can be assigned to multiple logical tones.
2. Alarm relay configuration determines the duration that a relay remains closed, selects the keyboard that drives the relay, and selects a specific physical relay.

ALARM TONES

An alarm tone can be assigned to an alarm group or to an alarm priority. An alarm tone can be triggered by group or by priority but not both. If triggered by group, a tone sounds when a tag in a group goes into alarm. If triggered by priority, a tone sounds based on the alarm priority assigned to the alarm condition a tag is currently in. For example, if the high alarm condition of a tag is assigned a priority of one and that tag enters a high alarm condition, the tone assigned to priority one will sound.

Refer to ***Configuring an Alarm Group*** in this section for the procedures to assign a tone to an alarm group. Refer to ***Configuring Alarm Priorities*** in this section for the procedures to assign a tone to an alarm priority.

A tone is identified and assigned by its logical tone number. 20 logical tones are supported. A logical tone is defined by selecting a keyboard annunciator and defining a priority, duration, and volume during tone configuration.

Tone configuration also sets a global alarm silence toggle for each logical tone and defines a node list to use when broadcasting tone silences. This is a function of global alarm acknowledge and silence. Tone configuration allows toggling silence broadcasting on a per tone basis. Refer to ***GLOBAL ALARM MANAGEMENT*** in this section for more information.

Logical tone configuration provides a test feature that allows hearing a tone to:

- Verify a keyboard annunciator is working.
- Verify correct configuration.
- Determine if the pitch, volume, and duration are sufficient to identify an alarm.

Defining a Logical Alarm Tone

Only one tone can sound at a time. Each tone has a logical priority, pitch, volume, and duration set in its configuration. The priority is used to identify which tone should sound first in the event two tones trigger at the same instant. A triggered tone normally continues to sound until it completes its entire duration, is manually silenced, or is overridden by a higher priority tone.

When alarms in different alarm groups or priorities occur simultaneously, the logical priority of the tone assigned to each of the groups or priorities in alarm is checked. This is done to determine which tone has priority and is to sound. If a tone with a higher logical priority is activated, the current tone automatically silences and the higher priority tone sounds. When the higher priority tone ceases and if lower priority tones are active, the next highest priority tone will be sounded.

A lower priority tone does not affect a higher priority tone. When a triggered lower priority tone is preempted by a higher priority tone, it remains active and its duration continues to increment. It will continue to remain active until the duration expires and may be sounded again when the higher priority tone ceases.

Other factors within an alarm group or priority also affect alarm tones. If multiple tags within a single group or priority go into alarm at the same instant, only one tone sounds to identify the entire group or priority. If there is a delay between tag alarms in a single group or priority, each alarm occurrence triggers a new alarm indication. A new alarm resets the duration counter of the current tone to zero, then the counter begins to increment until reaching its duration before the tone turns off.

> **NOTE:** Alarm priorities configuration modifies the tone priorities for alarm tones. The configuration that was performed more recently, either alarm priorities configuration or alarm tones configuration, overrides the previous configuration. Refer to ***ALARM PRIORITIES*** in this section for an explanation of the affects of alarm priorities configuration on logical tone priorities.

Use the *Tones* option to configure a logical alarm tone. Figure 9-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
- → *E System*
  - → *B Alarm Management*
    - → *D Alarm Tones*

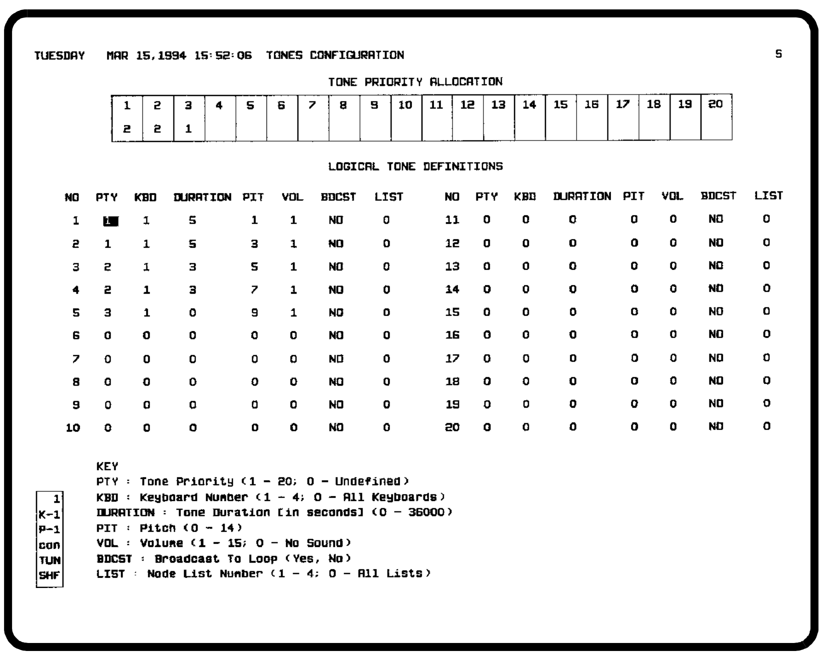


Figure 9-4. Tones Page

To define a logical tone:

1. Select a tone to define.

   a. Press **A** to select *CONFIGURE*. A *Tone Number* field appears.

b. Enter the number of a logical tone. A valid entry is from 1 to 20.

c. Press **ENTER**. The input cursor appears on the *PTY* field for that logical tone. The counter in the *TONE PRIORITY ALLOCATION* box for one of the tone priorities changes from cyan to green when a tone with that priority is being configured.

2. The *LOGICAL TONE DEFINITIONS* portion of the page displays the current alarm tone definitions and is the area used to change a tone definition. Enter a priority for the tone. A valid entry is 0 for undefined or 1 to 20.

After entering, the counter in the *TONE PRIORITY ALLOCATION* box at the top of page increments to the total number of tones with that priority assigned. For example, if tones six and ten are set to priority four, the *TONE PRIORITY ALLOCATION* box will indicate a *2* under priority level *4*.

3. Enter the number of the physical keyboard that is to provide this tone in the *KBD* field. A valid entry is 0 for all keyboards or 1 to 8 for a specific keyboard. If the *KBD* column is set to all zeroes, only five tones can be defined.

4. Enter the tone duration in seconds in the *DURATION* field. This is the amount of time this tone is to sound when triggered. A valid entry is 0 for no tone or 1 to 36000.

5. Enter the pitch of this tone in the *PIT* field. A valid entry is from 0 to 14. Each pitch number selects a different frequency of tone.

6. Enter the volume of this tone in the *VOL* field. A valid entry is 0 for no sound or 1 (softest) to 15 (loudest).

NOTE: The next two fields enable or disable global silencing.

7. Enter **YES** in the *BDCST* field to enable broadcasting a tone silence for this logical tone. A tone is silenced by pressing **SILENCE**. If broadcasting is enabled, the tone silences on all consoles that are contained in a node list. Enter **NO** in the field to disable broadcasting.

8. If the *BDCST* field is *YES*, enter a node list number in the *LIST* field. This field determines to which nodes to broadcast the silence. A valid entry is:

0 = broadcast the silence to the nodes in a node list automatically compiled by the console during start-up. This list contains the first 32 N90STA tags in the database that define a console or a computer.

1 = node list one.

2 = node list two.

3 = node list three.

4 = node list four.

If the *BDCST* field is *NO*, the *LIST* field is not valid. Refer to the ***GLOBAL ALARM MANAGEMENT*** in this section for more information about silence broadcasting and additional configuration requirements for broadcasting.

9. Press ENTER.

Testing a Logical Alarm Tone

Tone configuration provides a test option that sounds a tone using its configured pitch and volume. The test duration is set in the test function. To test a tone:

1. Press B to select *TEST*. The following fields appear:

Tone Number ___ *Test Duration* ___

2. Enter the number of a tone to test in the *Tone Number* field.

3. Enter the test duration in seconds (0 to 36000) in the *Test Duration* field.

4. Press ENTER. A *Test in progress* message and a *Test Time:* field appear. The *Test Time:* field counts down the test duration. The tone sounds until the test time decrements to zero.

The counter in the *TONE PRIORITY ALLOCATION* box for one of the tone priorities changes from cyan to yellow when a tone with that priority is being tested. Testing a tone turns off any active tones that are currently sounding and also prevents any new tones from sounding during the test. Press ESC to discontinue a test in progress.

ALARM RELAYS

An alarm relay can be assigned to an alarm group or to an alarm priority. An alarm relay can be triggered by group or by priority but not both. If triggered by group, a relay closes when a tag in a group goes into alarm. If triggered by priority, a relay closes based on the alarm priority assigned to the alarm condition a tag is currently in. For example, if the high alarm condition of a tag is assigned a priority of one and that tag enters a high alarm condition, the relay assigned to priority one will close.

Refer to ***Configuring an Alarm Group*** in this section for the procedures to assign a relay to an alarm group. Refer to ***Con-***

figuring Alarm Priorities in this section for the procedures to assign a relay to an alarm priority.

A relay is identified and assigned by its logical relay number. 48 logical relays are supported. A logical relay is defined by selecting a physical relay and defining a duration during relay configuration. The duration is the number of seconds the relay closes.

Logical relay configuration provides a test feature that allows closing a relay to:

- Verify a keyboard relay is working.
- Verify correct configuration.
- Verify an external annunciator is working.
- Determine if the duration is sufficient for an external alarm annunciator.

Setting the Relay Hold Option

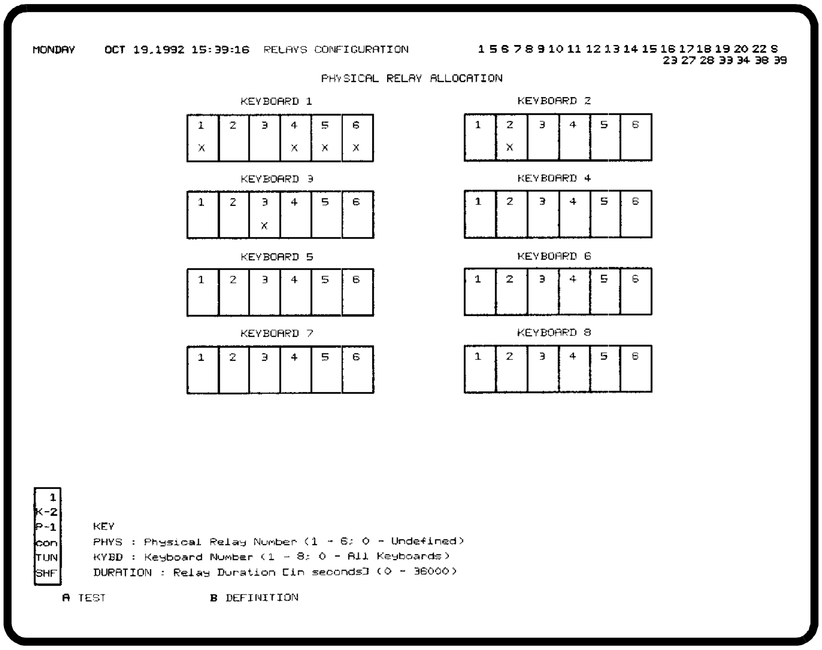
Normally, a relay remains closed until the alarm driving the relay returns to normal, the alarm is acknowledged, or the duration set for the relay expires. A *Relay Hold until Silenced* field on the general parameters page provides an option to change the operation of the relays.

If the *Relay Hold until Silenced* field is set to *YES*, an alarm relay that closes due to an alarm condition remains closed until **SILENCE** is pressed or while the previously described conditions are in effect. This does not remove any of the previously described functionality. If the field is left at its default of *NO*, a relay maintains its original functionality as stated previously. The duration is set on a per relay basis. Refer to ***GENERAL PARAMETERS*** in Section 3 for the procedures to set the *Relay Hold until Silenced* field.

Defining a Logical Alarm Relay

Use the *Relays* option to configure a logical alarm relay. Figure 9-5 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *B Alarm Management*
→ *E Alarm Relays*



TPS0776A

Figure 9-5. Relays Page - Status

The opening relay definition page identifies the relays that have previously been configured. An *X* displays in the *PHYSICAL RELAY ALLOCATION* box of a keyboard for each configured relay.

Press **C** at any time during or after defining relays to select the *LOGICAL* option (Figure 9-6). This exits the current page and returns to the opening relay configuration page (Figure 9-5) to allow viewing the results of relay configuration.

To define a logical relay:

1. Select a relay to define.

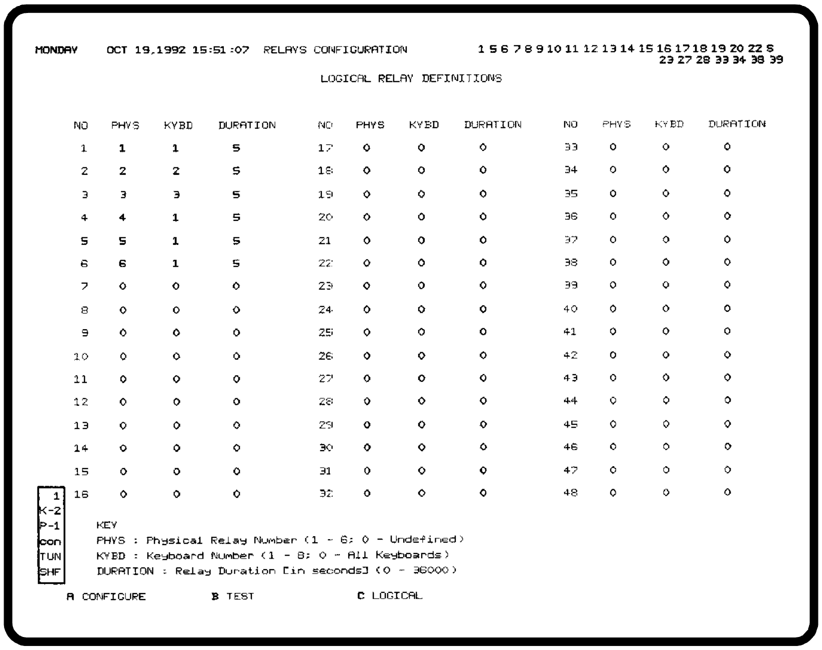
   a. Press **B** to select *DEFINITION*. Figure 9-6 shows the page that appears after selecting the option.

   b. Press **A** to select *CONFIGURE*. A *Relay Number* field appears.

   c. Enter the number of a logical relay. A valid entry is from 1 to 48.

   d. Press **ENTER**. The input cursor appears on the *PHYS* field for that logical relay.

2. Enter the physical relay number in the *PHYS* field. A valid entry is 0 for undefined or 1 to 6.



MONDAY OCT 19,1992 15:51 :07 RELAYS CONFIGURATION 1 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 22 S 23 27 28 33 34 38 39

LOGICAL RELAY DEFINITIONS

NO	PHYS	KYBD	DURATION	NO	PHYS	KYBD	DURATION	NO	PHYS	KYBD	DURATION
1	1	1	5	17	0	0	0	33	0	0	0
2	2	2	5	18	0	0	0	34	0	0	0
3	3	3	5	19	0	0	0	35	0	0	0
4	4	1	5	20	0	0	0	36	0	0	0
5	5	1	5	21	0	0	0	37	0	0	0
6	6	1	5	22	0	0	0	38	0	0	0
7	0	0	0	23	0	0	0	39	0	0	0
8	0	0	0	24	0	0	0	40	0	0	0
9	0	0	0	25	0	0	0	41	0	0	0
10	0	0	0	26	0	0	0	42	0	0	0
11	0	0	0	27	0	0	0	43	0	0	0
12	0	0	0	28	0	0	0	44	0	0	0
13	0	0	0	29	0	0	0	45	0	0	0
14	0	0	0	30	0	0	0	46	0	0	0
15	0	0	0	31	0	0	0	47	0	0	0
16	0	0	0	32	0	0	0	48	0	0	0

1
K-2
P-1
con
TUN
SHF

KEY
PHYS : Physical Relay Number (1 - 6; 0 - Undefined)
KYBD : Keyboard Number (1 - 8; 0 - All Keyboards)
DURATION : Relay Duration [in seconds] (0 - 36000)

A CONFIGURE B TEST C LOGICAL

TPS0775A

Figure 9-6. Relays Page - Definition Option

If a *Physical Device Overlap* error message occurs when attempting to enter a physical relay number, that relay number has been used previously. This can occur, for example, if a physical relay has been assigned to keyboard one and an attempt is made to assign that same physical relay number to keyboard two without first changing the keyboard number. In this case, the console thinks the same relay is being assigned in two different places. To avoid this, move to the *KYBD* field of the relay being defined and make the keyboard assignment before the physical relay assignment.

3. Enter the number of the physical keyboard that is to provide this relay in the *KYBD* field. A valid entry is 0 for all keyboards or 1 to 8 for a specific keyboard. If the *KYBD* column is set to all zeros, only six logical relays can be defined.

4. Enter the relay duration in seconds in the *DURATION* column. This is the amount of time this relay is to close when triggered. A valid entry is 0 to disable the relay or 1 to 36000.

5. Press **ENTER**.

Testing a Logical Alarm Relay

Relay configuration provides a test option for alarm relays. The test duration is set in the test function. To test a relay:

1. Select *TEST* from either configuration page. The following fields appear:

Relay Number ___ *Test Duration* ___

2. Enter the number of a relay to test in the *Relay Number* field.

3. Enter the test duration in seconds (0 to 36000) in the *Test Duration* field.

4. Press **ENTER**. A *Test in progress* message and a *Test Time:* field appear. The *Test Time:* field counts down the test duration.

The *X* displaying in the *PHYSICAL RELAY ALLOCATION* box for one of the keyboards changes from cyan to yellow when a relay driven by the keyboard is being tested. The relay remains closed until the test time decrements to zero. Press **ESC** to discontinue a test in progress.

ALARM QUALITY OPTION

The *Alarm Quality Option* configuration:

- Defines an override color scheme to be used for non good and good quality indications.
- Determines whether the last known good value reported for a tag or a bad quality string is to be displayed and logged if a tag enters a bad quality condition.
- Enables bad quality alarming.
- Determines if the override colors will affect the alarm status/quality/group field of a display.

NOTE: The configuration also affects console return-to-normal processing (i.e., *Blink Alarms* field for a *No Alarm* condition). Refer to ***Setting Return-to-Normal Operation*** in this section for an explanation.

The normal or alarm conditions for a tag are considered good quality. Non good quality refers to any quality indication besides a good quality indication such as bad, inhibit, suspect, etc. When discussing alarming, there is a distinct difference between tag alarm condition and tag quality. Tag quality refers to the validity of an exception reported point. A point reporting good quality means that the control configuration has determined that it is receiving valid input from a process device and is transmitting valid data for that point. It does not matter whether the point is in a no alarm (normal) or alarm condition. A point can report good quality but still be in an alarm condition.

The *Alarm Quality Option* configuration can be used to set override foreground and background colors for alarming that are to be used instead of alarm colors defined in displays and symbols. These colors also can be used to override the default console alarm colors. If alarm colors were not defined consistently among various displays, this allows standardizing the colors for consistency and to make alarms more easily recognizable. The colors set with this option affect the fields and dynamic symbols in a display that indicate alarms.

Bad quality alarming can be enabled to use the quality reported for a process point in alarm management. If enabled, a bad quality report triggers all the same alarm indications as any other alarm. Refer to the discussion on alarm management in the ***Operation*** instruction for a description of alarm status and quality indications for bad quality (Table 1-2 lists instruction numbers).

Setting the Alarm Quality Options

Use the *Alarm Quality Option* to define override alarm colors and bad quality alarming. Figure 9-7 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *D Alarm Quality Option*

Some changes made to this page require a reset while others take effect immediately. To define the fields:

1. Enter the appropriate data into each field of the page. Table 9-3 explains the alarm quality options page fields. Refer to this table when entering data.

2. Press **ENTER**. If making color, blink, or reverse video changes only, skip the next step. The console does not require a reset.

3. Reset the OIS application by using the procedures given in ***Reset*** in Section 2.

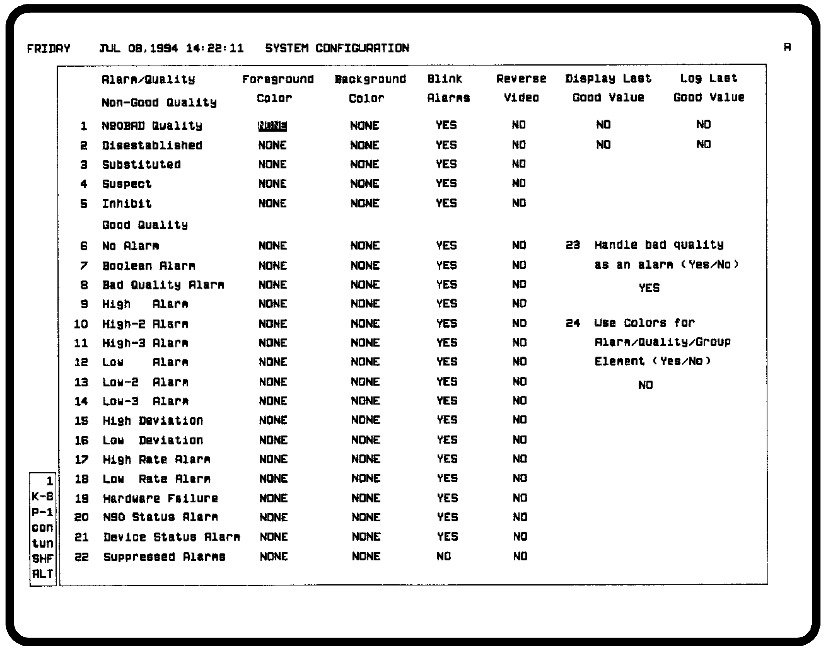


Figure 9-7. Alarm Quality Option Page

Table 9-3. Alarm Quality Options Page Fields

Field	Description
Foreground color[1]	Foreground color that is to appear for a particular non good quality or alarm condition (good quality). A valid entry is: NONE = use default alarm colors. 0 to 63 = override color.
Background color[1]	Background color that is to appear for a particular non good quality or alarm condition. A valid entry is: NONE = use default alarm colors. 0 to 63 = override color.
Blink alarms	Determines if the specific alarm condition or non good quality should blink when unacknowledged. A valid entry is: YES = enable blink. NO = disable blink.
Reverse video	Used to highlight a specific alarm condition or non good quality instead of specifying a foreground and background color. A valid entry is: YES = enable reverse video. This causes default foreground and background colors to appear in reverse. NO = disable reverse video. Use this when specifying foreground and background colors.
Display last good value	Determines if the value of a tag displays as a bad quality string or as the last known good value when the tag enters a bad quality condition. A valid entry is: YES = show last known good value. NO = show a bad quality string.

Table 9-3. Alarm Quality Options Page Fields (continued)

Field	Description
Log last good value	Determines if the value of a tag is logged as a bad quality string or as the last known good value when the tag enters a bad quality condition. The option affects custom logs only. A valid entry is: YES = log last known good value. NO = log a bad quality string.
Handle bad quality as an alarm	Configures bad quality alarming. A valid entry is: YES = enable bad quality alarming. A bad quality condition causes the console to show all normal alarm indications for a process variable being reported as in bad quality. NO = disable bad quality alarming.
Use colors for alarm/ quality/group element	Determines if the alarm status/quality/group field of a tag uses a default color scheme or the color scheme defined on this page. A valid entry is: YES = use alarm colors set on this page. NO = use default alarm colors. The console ignores the *Foreground Color*, *Background Color*, *Blink Alarms*, and *Reverse Video* fields for the alarm status/quality/group fields.

NOTE:
1. Call the CLRPAL system display to cross reference a color index number to an actual color. Use display by name and enter **!CLRPAL**.

Setting Return-to-Normal Operation

The *Blink Alarms* field for the *No Alarm* condition determines how a return-to-normal condition is to be treated. Enter **YES** in the *Blink Alarms* field for the *No Alarm* condition to cause any tag that returns to normal from an alarm to blink. The tag can then be acknowledged using **ACK ALARM** or **PAGE ACK**. When this feature is enabled:

- All acknowledge keys work for the alarm status/quality/group field of a tag that returns to a normal condition.
- Alarm and quality indicators blink in the tag summaries function to indicate an unacknowledged return-to-normal condition.
- An alarm status/quality/group field and a value field blink for an unacknowledged return-to-normal condition.
- Green is used as the foreground color to indicate a normal condition.

AUTOMATIC TAG ALARM INHIBITING

An alarm feature that can be enabled in the tag database is automatic alarm inhibit. This feature allows selecting a tag that is to automatically inhibit the alarm indications of another based on its current condition or state. For an analog type of inhibit tag, only one of its alarm conditions can be used to trigger inhibiting. Either the alarm state or digital state of a digital type of tag can be used to trigger inhibiting.

An alarm inhibit tag and its condition that triggers inhibiting are defined in the database for a tag. Three attributes in the database define alarm inhibiting: *Alarm Inhibit Tag*, *Inhibit On Alarm/State*, and *Inhibit Alarm Status/State*. To define automatic alarm inhibiting for a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter the name of the tag that is to inhibit alarming in the *Alarm Inhibit Tag* field. Any type of tag except RMSC can be used.

3. Enter **ALARM** in the *Inhibit On Alarm Status/State* field for alarm condition triggering or enter **STATE** for digital state triggering.

4. Select the alarm condition or digital state that is to trigger an alarm inhibit.

If *Inhibit On Alarm/State* field is *ALARM*, select the specific alarm condition that is to trigger alarm inhibit for this tag in the *Inhibit Alarm Status/State* field. Any valid alarm condition that the inhibit tag can be in is valid. Leave the field blank to select the no alarm condition. Enter an alarm condition as **H**, **L**, **2H**, **2L**, **3H**, **3L**, **HD**, **LD, HF,** or **A**. Use **A** to select the alarm condition for a digital type of tag.

If the *Inhibit On Alarm/State* field is *STATE*, select the specific state that is to trigger alarm inhibit for this tag in the *Inhibit Alarm Status/State* field. Any valid state that the inhibit tag can be in is valid. Enter the state as **0**, **1**, **2**, or **3**.

Leave the *Alarm Inhibit Tag* field blank to disable automatic alarm inhibiting. The *Inhibit On Alarm/State* and *Inhibit Alarm Status/State* fields cannot be accessed when it is blank.

ALARM SUMMARY

Up to 1,000 alarms can be saved in an alarm summary list for display in alarm summaries. This single alarm summary list is used for ***all*** alarm summaries. The number of alarms being saved for alarm summaries depends on system configuration.

When using chronological sorting for alarm summaries, the most recent alarms up to the maximum will be saved. When using priority sorting, alarms are saved based on alarm priority rather than time of occurrence. For example, all priority one alarms are saved, then priority two alarms, then priority three alarms, etc., until the maximum number of alarms is reached.

An alarm summary can appear as part of a user-created display or an operator configurable display. An alarm summary can also be a complete display. The number of alarm entries that will appear in any given alarm summary depends on the maximum number of alarms being saved and the actual alarm summary display or element. If filtering of alarms is built into an alarm summary, then something less than the maximum number of alarms being saved will appear in the summary.

An alarm summary can be assigned to **ALARM SUMM** through keyboard configuration for easy access. A standard alarm summary display is provided and assigned to this key by default.

Alarm summary configuration:

- Sets chronological or priority sorting for alarms. This affects all summaries.
- Sets maximum number of current alarms. This affects all summaries.
- Defines the format and content of line entries. Each summary can use a different format.
- Sets priority override colors. A *Priority Colors* option sets the foreground and background color for each priority level for both alarm and return-to-normal alarm entries. These colors override any existing color scheme. This affects all summaries.
- Defines the titles that appear in alarm summary report printouts.

Building an Alarm Summary

An alarm summary can be built as a complete display or as an element of a display. The graphical display configuration (GDC) in the console configuration tools program is normally used to build an alarm summary. The console elementary line editor (ELE) function can be used to build a summary, if desired, by defining individual escape commands. Refer to ***DISPLAY GENERATION*** in Section 7 for a description of these utilities for building displays.

The alarm summary escape command (**as 82** or **as 83**) defines the overall characteristics of the summary. Depending on the command used, the following characteristics are defined in the command parameters:

- Alarm groups or priorities (all or selected) that appear.
- Alarm types (all, acknowledged only, or unacknowledged only) that appear.

- Size.
- Key selector.
- Touch points enabled or disabled.
- Primary display selectors enabled or disabled.
- Orientation of primary display selectors.
- Position in the window.
- Line format used for alarm entries.
- Colors.

The alarm summary escape command gives three alarm group or alarm priority options:

- Range of alarm groups or priorities only.
- Individual groups or priorities only.
- Both range and individual groups or priorities.

These options allow filtering the summary. A summary can be limited to show alarms for specific alarm groups or for specific alarm priorities. It can also be set to show all alarms, acknowledged alarms only, or unacknowledged alarms only. Refer to the ***Display Builder Reference*** instruction for a description of alarm summary escape command parameters (Table 1-2 lists instruction numbers).

Setting the Sort Option

System configuration defines the type of sorting, either chronological or priority, used for all alarm summaries. To set the option:

1. Follow the steps given in ***GENERAL PARAMETERS*** in Section 3.

2. Choose one of the following options for the *Alarm Management Type* field:

    0 = default; alarms are saved in chronological order as they occur. A return-to-normal appears as a new entry.

    1 = alarms are saved based on priority. An alarm summary will present alarms within the same priority grouped together, then chronologically ordered within the priority. A return-to-normal appears as a new entry.

    2 = same as option 0 but employs fixed position return-to-normal.

3 = same as option 1 but employs fixed position return-to-normal.

If using options zero or one, a tag that returns to its normal condition generates a new alarm line at the beginning of the alarm summary. If the maximum number of alarms have been saved, this new entry causes the last entry to scroll off the summary. Options two and three enable a fixed position return-to-normal feature. If used, a tag that returns to its normal condition maintains its current position in the list but changes to some return-to-normal color.

If using options zero or two, all alarm entries in alarm summaries default to priority one (*P1*). Priority sorting is not implemented. Also, the alarm priority touch point indicators will call a summary when selected but the summary will be empty.

Setting the Maximum Number of Alarms

System configuration defines the number of alarms, up to 1,000, that are to be saved for display in alarm summaries. A single alarm list that all alarm summaries use is maintained by the console. If a summary is created to show all alarms being saved by the console, up to 1,000 alarm entries can appear in the summary. If a summary is limited to certain alarm groups or priorities, the maximum number of alarms that will appear in the summary is something less than 1,000. This assumes, however, that not all alarms being saved are in the alarm groups or priorities to which the summary is restricted.

Example: The maximum of 1,000 alarms are being saved. Of the 1,000 currently being saved, 600 are in alarm group one and 400 are in group two. In an alarm summary that is not restricted to any particular alarm group, all 1,000 alarms will appear in the summary. In an alarm summary that is restricted to alarm group two, only the 400 group two alarms being saved will appear in the summary.

To set the maximum number of alarms to save:

1. Follow the steps given in ***GENERAL PARAMETERS*** in Section 3.

2. Enter the desired maximum number of alarms to save in the *Max Alarms in List* field. A valid entry is from 100 to 1000.

Line Formats

Alarm summary format configuration allows tailoring the format and content of entries in alarm summaries (i.e., line formats). Up to 106 line formats numbered from 0 to 105 can be created. Of these, line formats 0 through 4 are fixed formats and cannot be modified. The remaining 101 can be user-defined. A format, after being specified in an alarm summary escape command, defines the attributes and their order of appearance for all entries that appear in the summary. The alarm summary report also uses one of the line formats to determine the attributes and order of appearance in the printed summary version.

This part of the section describes the characteristics that can be set and attributes that can be included in an alarm line entry of an alarm summary. Specifically, it discusses:

- Line option.
- Character height.
- Type, color, and position of alarm line elements.

It also explains the requirements for using a line format in an alarm summary.

Each line format requires a line option and character height specification. The line option and character height work together. The type, position, and color of each data item that is to appear in an alarm line is configurable.

CHARACTER HEIGHT AND LINE OPTION

Character height is either single height, double height, or 4.5 millimeter. Line options are zero, one, or two.

Character Height

Single height produces a character that is 124 display units or one line. Double height produces a character that is 248 display units or two lines. Both use a fixed spacing factor of 0.0. Less double height characters than single height characters can appear in a line entry. The 4.5 millimeter option produces 4.5 millimeter characters on the largest sized window.

Line Option

Line option zero allocates two lines. With this option, an alarm entry appears as either two lines of single height characters or one line of double height characters. The 4.5 millimeter character height cannot be used with this option.

Line option one allocates four lines and is intended for two lines of complex characters. If not using a complex language set, it allows for four lines of single height characters or two lines of double height characters. The 4.5 millimeter character height cannot be used with this option.

Line option two allocates one line. With this option only single height or 4.5 millimeter characters are permitted. Line option 2

must be used to produce 4.5 millimeter characters in a summary.

> **NOTE:** Alternate language is enabled during system configuration (i.e., *Alternate Language* attribute set to *YES*). Extended characters occupy the same amount of space as standard English characters. Complex characters occupy twice the space used by standard English and extended characters; therefore, they occupy the same amount of space as two single height lines.

Table 9-4 lists the default line option and character height for formats zero to four. These formats cannot be modified. The line formats can be viewed by using the *Display/Modify Format* option.

Table 9-4. Default Alarm Line Formats

Format Number	Line Option	Character Height
0	0	Single
1	1	Single
2	2	Single
3	0	Double
4	1	Double

Before defining line formats for alarm summaries, the alarm summary element size should be considered. The element size is set in the alarm summary escape command when the element is built. Table 9-5 shows the effects that a selected element size, line option and character height have on the capabilities of an alarm summary. Element capabilities refers to the number of lines per alarm entry and number of alarm entries that can appear on a single summary page.

Table 9-5. Alarm Summary Element Options

Element Size	Line Option	Character Height	Number of Alarms	Lines per Alarm Entry
Full (32)	0 0 1 1	Single Double Single Double	16 16 8 8	2 1 4 2
Half (33)	0 0 1 1 2	Single Double Single Double Single	8 8 4 4 16	2 1 4 2 1
Quarter (34)	0 0 1 1 2	Single Double Single Double Single	4 4 2 2 8	2 1 4 2 1

Table 9-5. Alarm Summary Element Options (continued)

Element Size	Line Option	Character Height	Number of Alarms	Lines per Alarm Entry
Eighth (35)	0	Single	2	2
	0	Double	2	1
	1	Single	1	4
	1	Double	1	2
	2	Single	4	1
Full (36)	2	Single	24	1
	2	4.5 mm	24	1

ELEMENTS OF A LINE FORMAT

The data items that will appear in an entry of an alarm summary are determined by a line format. This is true for the standard alarm summary or any displays that use an alarm summary element. An alarm summary escape command (**as 82** or **as 83**) sets the actual alarm summary element size ranging from full size to ⅛ -size.

The console can display 64 different colors, 16 base colors and three shades of each base color. Through line formatting, a color can be assigned to each element of an alarm line.

Elements that can be included in an alarm line entry include:

Alarm comment - up to 64-character comment from the database.

> **NOTE:** Only one type of alarm comment field can be used in a single alarm line format (e.g., alarm comment cannot be used with alarm comment/text string).

Alarm comment/text string - up to 80-character text string or 64-character alarm comment. The text string appears for a TEXTSTR tag while an alarm comment appears for all other tags. No local truncation indication will appear.

Alarm comment/text string (with local truncation) - up to 80-character text string or 64-character alarm comment. The text string appears for a TEXTSTR tag while an alarm comment appears for all other tags. The summary indicates any local truncation with an ellipsis (. . .).

Alarm condition - alarm status, quality, and group.

Current value - current value or state and the status.

Date of alarm - date of alarm occurrence.

Engineering units - engineering unit associated with an analog value.

Exceeded limit value - violated alarm limit for tags that report analog values.

Latched alarm date - date of occurrence latched to the alarming tag. This displays the date of occurrence in a return-to-normal entry.

Latched alarm time - time of occurrence latched to the alarming tag. This displays the time of occurrence in a return-to-normal entry.

Latch alarm time (tenth second resolution) - time of occurrence to the tenth of a second latched to the alarming tag. Module time-stamp must be enabled to use this. Refer to ***Module Time-Stamp*** in Section 3 for an explanation.

Priority - two-character indicator that shows alarm priority. Priority for a tag is set in the tag database.

Remote truncation status - truncation status to indicate the presence or absence of remote truncation for a text string. Remote truncation is performed by a PCU module or the communications interface unit of the console.

Tag description - tag description from the database.

Tag index - tag index number from the database.

Tag name - tag name from the database.

Text string - up to 80-character text string of a TEXTSTR tag. A local truncation indication will not appear.

> **NOTE:** Only one type of text string field can be used in a single alarm line format (e.g., text string cannot be used with alarm comment/text string).

Text string (with local truncation) - up to 80-character text string of a TEXTSTR tag. The summary indicates any local truncation with an ellipsis (. . .).

Time of alarm - time of alarm occurrence.

Time of alarm (tenth second resolution) - time of alarm occurrence to the tenth of a second. Module time-stamp must be enabled to use this. Refer to ***Module Time-Stamp*** in Section 3 for an explanation.

USING A LINE FORMAT

After defining a line format, the alarm summary display or any display that uses an alarm summary element must be edited to include the desired format. For example, to use line format five in a summary display, the alarm summary escape command (**as 82** or **as 83**) must be edited to include line format five as its format parameter.

Use the *Operator Displays' Faceplates* function to select the line format used for an alarm summary that is part of an operator configurable display. To select the format:

1. Refer to Section 14 for the procedures to call the function.

2. Call page 3 of the function.

3. Change the *Alarm format record number* field to the desired line format to use. This entry affects all summaries that are created as part of an operator configurable display.

The line format for operator configurable displays can be changed at any time. The line format number currently being used is recorded when a display is saved. This number does not change unless the display is called and saved again through the operator configurable displays function.

Defining a Line Format

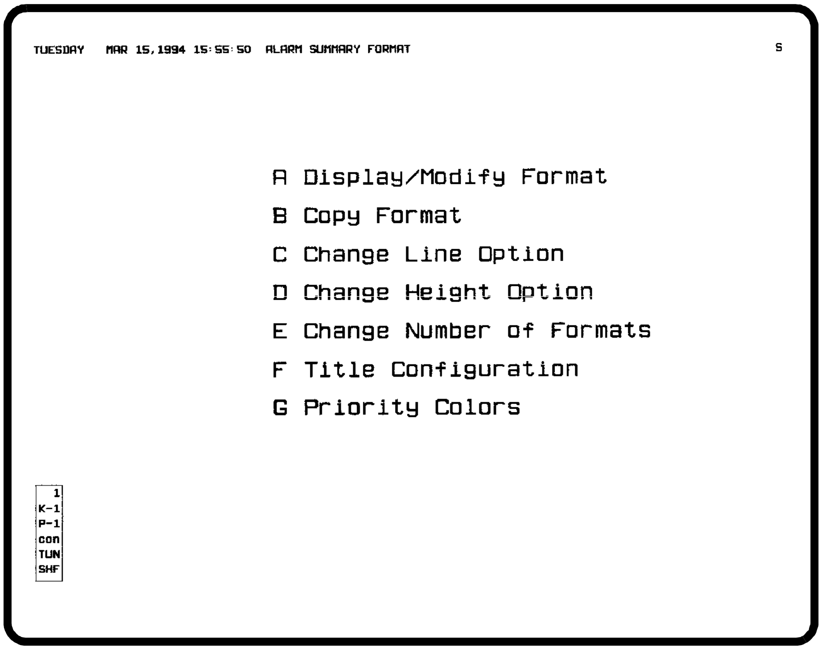
This part of the section describes the procedures to define a line format to be used in an alarm summary. Only one line format can be selected for an alarm summary, either a displayed summary or a printed summary report. Up to 106 (zero to 105) line formats can be defined.

Before defining a line format, the maximum number of formats for the system must be specified. After specifying the number of formats, select the line option and character height for each line format. Then format the alarm line by selecting each data item that is to appear in the line and putting them in the desired order of appearance.

Use the *Alarm Summary Format* option to format alarm summaries. Figure 9-8 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *E Alarm Summary Format*

The procedures in the following sections can be used to add new line formats or to edit existing line formats. Line formats can be entered or edited at any time. Existing line formats can be viewed or edited by using the *Display/Modify Format* option. Any modifications made to a format are immediately implemented in all user-created displays and operator configurable



TPS0003B

Figure 9-8. Alarm Summary Format Page - Options Menu

displays that use the format. Calling any display containing an alarm summary shows the results of the changes.

CREATING A NEW LINE FORMAT

To create a new line format:

1. Increase the total number of line formats if necessary using the *Change Number of Formats* option.

2. Select the line option for the new format by using *Change Line Option.*

3. Select the character height for the new format by using *Change Height Option.*

4. Add the desired elements to the new format by using the *Display/Modify Format* option.

CHANGE NUMBER OF FORMATS OPTION

The *Change Number of Formats* option is used to set the total number of formats (Figure 9-8). The total number of formats needed depends on the number of alarm summary displays or reports that require a unique line format. If several alarm summary displays or reports are used, they can all use the same line format or can all use different line formats. To set the number of formats:

1. Choose *E Change Number of Formats*. This clears the option menu and displays a *Number of Formats* field.

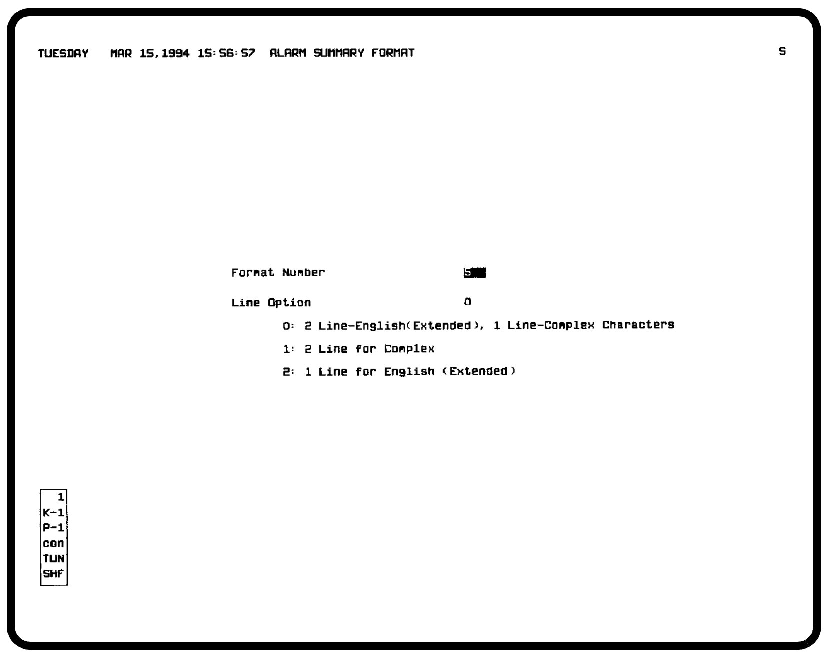
2. Enter the total ***number of formats*** required. A valid entry is from 5 to 106. Five predefined line formats (0 to 4) are provided.

3. Press **ENTER**.

CHANGE LINE OPTION

The *Change Line Option* is used to select the line option for a line format (Figure 9-8). To select the line option:

1. Choose *C Change Line Option*. This clears the option menu and displays the line option fields (Figure 9-9).



TPS0004B

Figure 9-9. Alarm Summary Format Page - Line Options

2. Enter the line format number that the line option is being set for in the *Format Number* field. A valid entry is from 5 to 105 depending on the total number of formats set previously. Formats 0 through 4 cannot be modified.

3. In the *Line Option* field, enter the desired line option for this line format. A valid entry is:

0 = summary allows for two lines of single height characters or one line of double height characters for each entry.

1 = summary allows for four lines of single height characters or two lines of double height characters for each entry.

2 = summary allows for one line of single height or 4.5 millimeter characters only.

NOTE: An extended character is a normal, one-display line character; a complex character is a double height, two-display line character.

Refer to Table 9-5 to determine the affects of this option on the capabilities of an alarm summary.

4. Press **ENTER**.

CHARACTER HEIGHT OPTION

The *Change Height Option* is used to select the character height for a line format (Figure 9-8). To select the character height:

1. Choose *D Change Height Option*. This clears the option menu and displays character height selection fields.

2. Enter the line format number that the character height is being selected for in the *Format Number* field. A valid entry is from 5 to 105 depending on the total number of formats set previously. Formats 0 through 4 cannot be changed.

3. In the *Character Height Option* field, enter the desired character height. A valid entry is 0 for single height characters, 1 for double height characters, or 2 for 4.5 millimeter characters. This must be consistent with the line option chosen.

NOTE: Use a character height of 2 only when 4.5 millimeter characters are enabled and only when line option 2 is used. Refer to ***4.5 Millimeter Characters*** in Section 3 for the procedures to enable 4.5 millimeter characters.

Refer to Table 9-5 to determine the affects of this option on the capabilities of an alarm summary.

4. Press **ENTER**.

DISPLAY/MODIFY FORMAT OPTION

The *Display/Modify Format* is used to view or modify an existing format or to create a new format (Figure 9-8). To use the option:

1. Choose *A Display/Modify Format*. This clears the option menu and an *Enter Format Number* field appears.

2. Enter the line format number to view, edit, or define in the field. A valid entry is from 0 to 105. The maximum may actually be something less than 105 depending on the total number of formats enabled previously. Formats 0 through 4 can be viewed only.

3. Press **ENTER** to call the format. A menu of options appears (Figure 9-10).

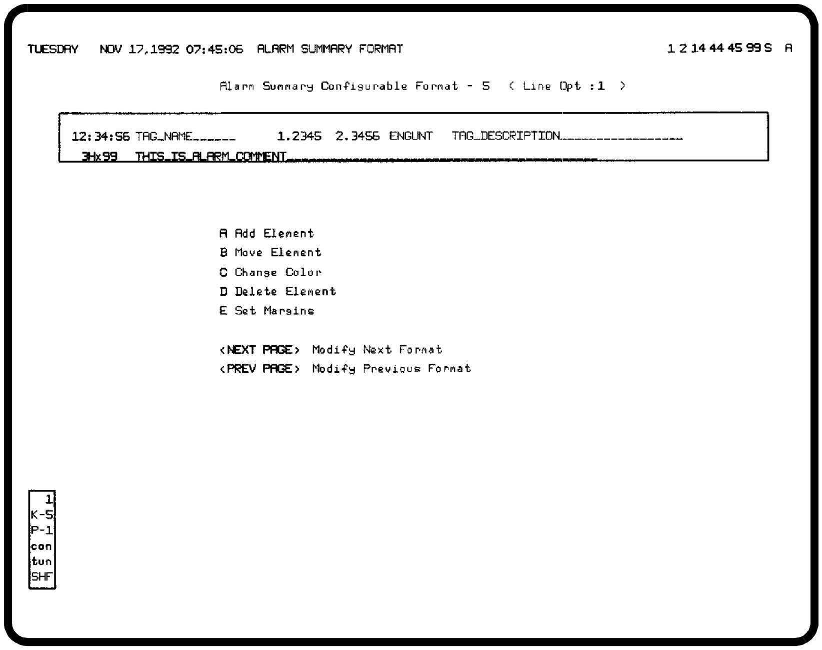


Figure 9-10. Alarm Summary Format Page - Display/Modify Format Menu

The *Alarm Summary Configurable Format - __ (Line Opt: __)* field at the top of the page shows the number of the selected line format and the line option previously chosen for the line format. Press **NEXT PAGE** or **PREV PAGE** to sequence to the next or previous line format.

The boxed area shows the actual format that will be used for an entry in a summary that uses the format. If any of formats five through 105 are selected, the options presented allow modifying the line elements shown in this boxed area. The options are described in the following sections.

Adding an Element

Use the *A Add Element* option to add new elements to the current format (Figure 9-10). When entering line elements, the following margins must be maintained:

- For line formats using line option zero, line elements must be positioned within a maximum 400 to 9600 x-coordinate and zero to 310 y-coordinate.
- For line formats using line option one, line elements must be positioned within a maximum 400 to 9600 x-coordinate and zero to 620 y-coordinate.
- For line option two, line elements must be positioned within a maximum 400 to 9600 x-coordinate and zero to 155 y-coordinate.

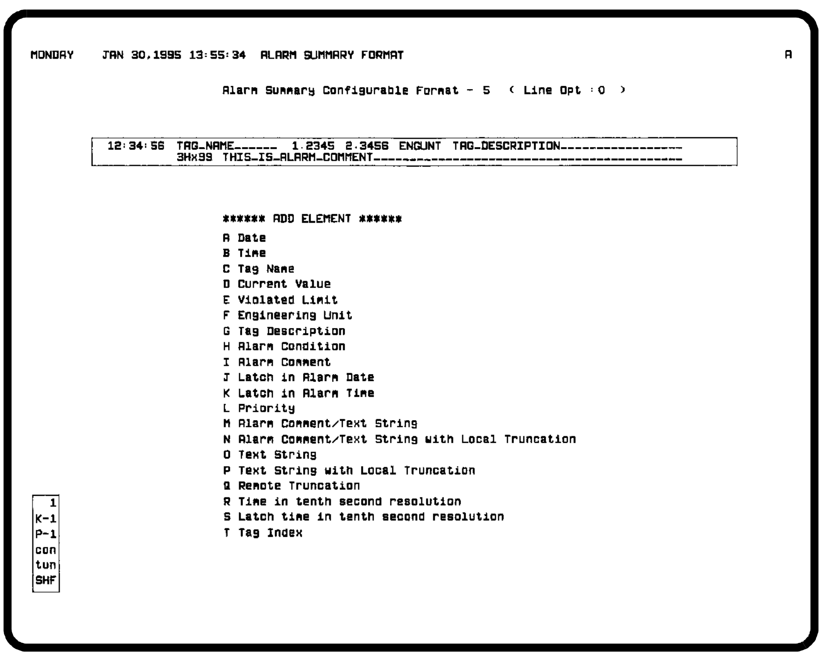
The margins can be seen in the grid that appears after choosing any *Add Element* option.

> **NOTE:** The line format left and right margin can be modified by using the *Set Margins* option.

All the different types of line elements are entered in basically the same way. Only one of each type of line element can be defined in a single format. The message *Duplicate definition* appears if an attribute has already been defined in the format. Each element of a line format can display in a different color.

To add an element:

1. Choose option *A*. This calls up an *Add Element* menu (Figure 9-11).



TPS0006D

Figure 9-11. Alarm Summary Format Page - Add Element Menu

2. Select one of the element options (i.e., *A* through *T*). Input fields appear after selection that allow entering an x-coordinate and a y-coordinate, and in some cases a length. The length only needs to be specified when choosing an alarm comment or a text string option. For all others the field is informational only.

3. Enter an x-coordinate and y-coordinate, and a length if required.

4. Press **ENTER**. An error message appears if an entered coordinate attempts to place a line element outside the margin limits, causes the element to overlap another, or causes the element to extend beyond the margins. After entering, a color menu appears.

5. Enter a color code. The color graph shows the actual colors and their respective color codes. Select a color by its number.

6. Press **ENTER**. The new element should appear in the boxed area at its respective position and in its chosen color.

7. Press **ENTER**. Further line modification can be performed by using the *Display/Modify Format* menu items.

If the position and character length of an element are not correct, they can be adjusted by using the *Move Element* option. If the color of an element is not correct, it can be changed by using the *Change Color* option.

Moving an Element

Use the *B Move Element* option to move elements of a line format and to change the character length of an element if necessary (Figure 9-10). To move an element or change its character length:

1. Choose option *B*.

2. Select an element.

    a. Press **TAB** or **TAB BACK** to select an element. An element blinks when selected.

    b. Press **ENTER**.

3. Enter a new x-coordinate and y-coordinate, and a length if required.

4. Press **ENTER**. The element should appear in the boxed area in its respective position and length. An error message appears if an entered coordinate attempts to place the line element out-

side the margin limits, causes the element to overlap another, or causes the element to extend beyond the margins.

5. Press **ENTER**. Further line modification can be performed by using the *Display/Modify Format* menu items.

Changing the Color of an Element

Use the *C Change Color* option to change the color of a line element (Figure 9-10). To change the color of an element:

1. Choose option *C*.

2. Select an element.

   a. Press **TAB** or **TAB BACK** to select an element. An element blinks when selected.

   b. Press **ENTER**.

3. Enter a color code. The color graph presented shows the actual colors and their respective color codes. Select a color by its number.

4. Press **ENTER**. The element should appear in the boxed area in its new color.

5. Press **ENTER**. Further line modification can be performed by using the *Display/Modify Format* menu items.

Deleting an Element

Use the *D Delete Element* option to delete a line element that is no longer required (Figure 9-10). To delete an element:

1. Choose option *D*.

2. Select an element. Press **TAB** or **TAB BACK** to select an element. An element blinks when selected.

3. Press **ENTER**. This deletes the selected element. The element should no longer appear in the boxed area.

4. Press **ENTER**. Further line modification can be performed by using the *Display/Modify Format* menu items.

Setting the Margins for a Format

Use the *E Set Margins* option to reduce the size of a line by adjusting the left and right margins (Figure 9-10). To change either or both left and right margins:

1. Select option *E*. Two input fields appear. The upper field is for the left margin, lower for the right. An entry can be made in both or only one of these fields.

2. Enter the new margins. A valid entry is from 400 to 9600.

3. Press ENTER. The outline of the boxed area should be reduced or expanded to the new limits. Any elements added or moved must fall within these new limits.

4. Press ENTER. Further line modification can be performed by using the *Display/Modify Format* menu items.

After changing the margins, make sure all elements of a line format are within the new x-coordinate limits. Any element outside the margins will not appear in the summary. The upper and lower limits are still maintained by the line option selection.

COPY FORMAT OPTION

The *B Copy Format* option is used to copy an existing line format (Figure 9-8). To copy a format:

1. Choose option *B*. The following fields appear:

Copy From Format # __ To __

2. Key in the line format number to copy in the *Copy From Format #* field. A valid entry depends on the total number of formats set previously. The range is from 0 to 105 for the ***from*** entry. The maximum may actually be something less than 105 depending on the total number of formats enabled previously.

3. In the *To* field, enter the line format number to which to copy. The range is from 5 to 105 for the ***to*** entry.

4. Press ENTER to initiate the copy. If copying a line format over an existing format, confirmation is required before the copy will be initiated.

Setting Priority Colors

Alarm entries in an alarm summary can be sorted by priority. This must first be enabled with the *Alarm Management Type* field on the general parameters page, however. Refer to ***Setting***

the Sort Option for an explanation. Priorities for alarms are set in the tag database.

A foreground and background color can be defined for each alarm priority level. These colors override any color scheme defined for a line format. Priority colors are set separately for alarm and return-to-normal conditions.

Use the *Alarm Summary Format* option to define priority colors. Figure 9-8 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

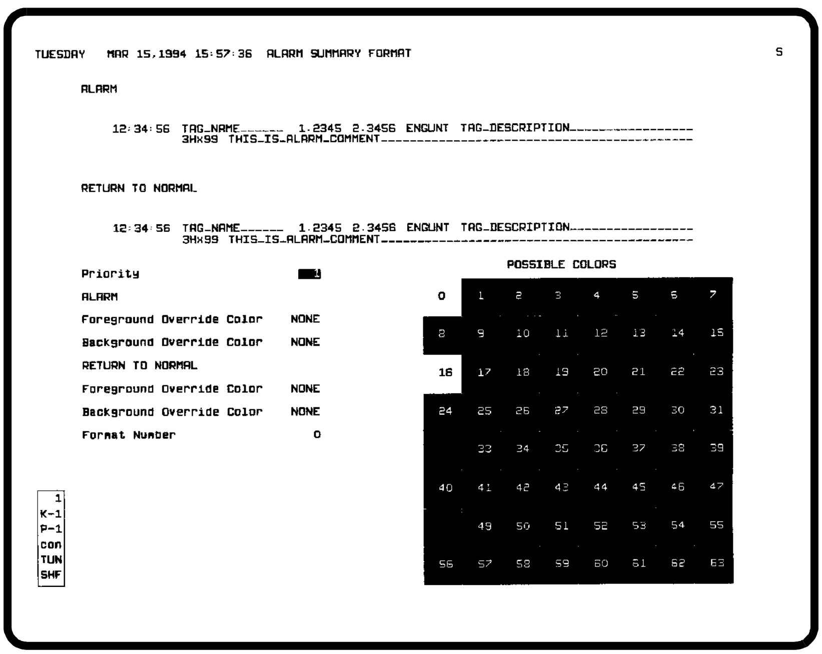
A OIS Configuration

→ *E System*

→ *E Alarm Summary Format*

To define a foreground and background color for a priority level:

1. Select *G Priority Colors*. This calls the page used to define priority colors (Figure 9-12).



TPS0007C

Figure 9-12. Alarm Summary Format Page - Priority Colors

2. In the *Priority* field, enter the number of a priority level to define. A valid entry is from 1 to 8. Press **NEXT PAGE** or **PREV PAGE** to sequence to the next or previous priority level.

NOTE: Enter **NONE** in the override color fields to use the colors selected during line format configuration without using priority color override.

3. Enter a color code, from the displayed color graph, in the *Foreground Override Color* field for *ALARM*. A valid entry is from 0 to 63 or NONE. This selects the foreground color to be used for an alarm entry in an alarm summary with this priority.

4. Enter a color code in the *Background Override Color* field for *ALARM*. A valid entry is from 0 to 63 or NONE. This selects the background color to be used for an alarm entry in an alarm summary with this priority.

5. Enter a color code in the *Foreground Override Color* field for *RETURN TO NORMAL*. A valid entry is from 0 to 63 or NONE. This selects the foreground color to be used for a return-to-normal entry in an alarm summary with this priority.

6. Enter a color code in the *Background Override Color* field for *RETURN TO NORMAL*. A valid entry is from 0 to 63 or NONE. This selects the background color to be used for a return-to-normal entry in an alarm summary with this priority.

7. The *Format Number* field allows entering the number of a defined line format to be shown at the top of the page. This provides an example of what the line will actually look like in an alarm summary for both alarm and return-to-normal conditions. A change to this field does not affect the configuration of alarm summaries, only the line format displayed at the top of the page.

Enter the number of a line format. A valid entry is from 0 to 105. The maximum may actually be something less than 105 depending on the total number of formats enabled previously.

8. Press **ENTER**. The line format appears at the top of the page in the selected alarm and return-to-normal priority colors. This also updates ***this*** configuration.

ALARM SUMMARY REPORT

An alarm summary report creates a hard copy list of current alarms. Up to 1,000 alarm entries can appear in a report. A report will list alarms for only those tags that are part of alarm groups selected in the report definition. Each report can contain the alarms for a range of alarm groups and a maximum of ten additional alarm groups. Up to 12 different reports can be generated.

Printing for the reports can be scheduled to occur at a specific time, then at time intervals. Printing can also be left unscheduled to occur after being triggered by a specific process event. Either or both types of scheduling can be used. A process event that can trigger an alarm summary report is a change in:

- Analog alarm condition of a trigger tag.
- Digital state of a trigger tag.
- Alarm status (to alarm from normal, or from alarm to normal) of a trigger tag.

An alarm summary report can span several pages. Each page has a title line followed by a maximum of 30 alarm entries. The title line format is configurable; up to 12 different titles can be created. Each report can have the same title or each can use a unique title.

Alarm summary report configuration includes:

- Defining a title for a report.
- Formatting the entries of a report.
- Defining each summary report and scheduling all reports.

Defining a Summary Report Title

Up to 12 titles can be defined then assigned to each of 12 possible reports. The title is two lines with a maximum of 132 characters per line.

Use the *Title Configuration* option to define titles for alarm summary reports. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

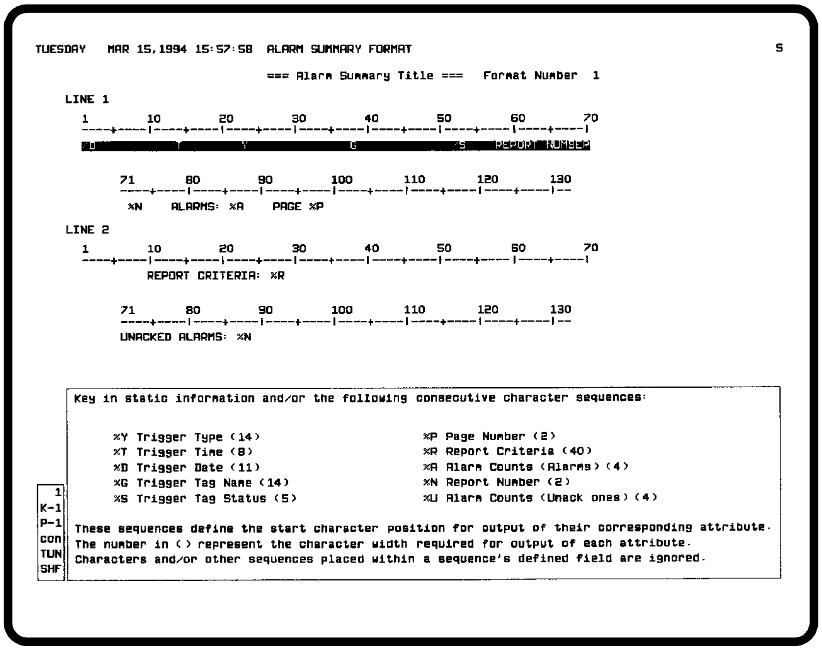
A OIS Configuration
→ *E System*
→ *E Alarm Summary Format*
→ *F Title Configuration*

To define a report title:

1. Enter the number of a report title in the *Enter Format Number* field. A valid entry is from 1 to 12.

2. Press **ENTER**. This calls a page similar to Figure 9-13.

3. Define the title elements.



TPS0052B

Figure 9-13. Alarm Summary Format Page - Alarm Summary Title

Initially, the page contains a default title definition. This title can either be used as is or it can be modified. Title elements can be deleted, added, or moved.

Each element that is to appear in a title is entered as a separate sequence code. The sequence code position on the input line determines its starting character position.

At the bottom of the page is a list of two-character sequence codes. These codes are used to define each element that is to appear in the title. Static text can also be added. The available title elements (sequence codes) are:

Report Number (*%N*) - two characters that identify the report number from 1 to 12.

Report Criteria (*%R*) - 40 characters that identify an alarm group range and an alarm group list.

Trigger Type (*%Y*) - 14 characters that identify the trigger type (time or tag).

Trigger Time (*%T*) - eight characters that show the trigger time for either a time or tag triggered report.

Trigger Date (*%D*) - 11 characters that show the trigger date for either a time or tag triggered report.

Trigger Tag Name (*%G*) - 14-character name of the trigger tag of a tag triggered report.

Trigger Tag Status (*%S*) - five characters that identify the alarm condition, status, or state of the trigger tag for the report.

Alarm Counts (Alarms) (*%A*) - four characters that indicate the total number of current alarms that meet the report criteria.

Alarm Counts (Unack ones) (*%U*) - four characters that indicate the total number of current unacknowledged alarms that meet the report criteria.

Page Number (*%P*) - two characters showing the current page number.

All or only specific codes can be used as long as the 132 character maximum is maintained. The number in brackets ( ) next to each sequence code identifies the number of character spaces each type of element occupies. This is the minimum number of spaces to leave between each code. Any characters placed within these reserved spaces are ignored. For static text, each character occupies a single character space. The codes can be placed anywhere within the highlighted *LINE* fields and in any order.

To define a title, enter the sequence codes or static data that are to appear in the title in the *LINE 1* and *LINE 2* fields. The placement of a code in these fields directly corresponds to its position in the final printout. Each *LINE* field is separated into two parts; however, in the final printout they become one line.

Allow at least the number of spaces indicated in the brackets between sequence codes. The code itself counts as two spaces. To define a sequence code:

1. Enter a sequence code exactly as it is shown at the bottom of the page. For example, to show the page number in the title enter **%P**.

2. Press **ENTER**.

The title number is used during alarm summary report configuration to assign the title to a report.

Formatting a Report Entry

Alarm entries are formatted for printing based on the line format chosen for the report. Alarm summary format configuration defines a line format. Each of 12 reports can use the same or a different line format. These formats can be the console default line formats (zero to four) or user-defined formats (five to 105).

To define a line format for a report:

1. Follow the steps given in ***Defining a Line Format*** in this section.

2. After a line format has been created, follow the steps given in ***Defining and Scheduling Alarm Summary Reports*** in this section to assign the format to a report (*Format Record* field).

A printed alarm entry is limited to two lines. As a result, choosing an alarm format for a report of more than two display lines may give undesirable and possibly unreadable results in the printed report.

Defining and Scheduling Alarm Summary Reports

Alarm summary report configuration enables up to 12 individual reports and schedules report printing. Print scheduling is for ***all*** reports, not individual reports. The alarms that print are the current (snapshot) alarms at the time the report starts. When the line format used in the report contains a value field, the printed summary version contains the tag value at the time of alarm occurrence or the time at which the tag returned to normal.

Alarm summary reports can print at a specific time, then at time intervals (i.e., *PERIODIC REPORTS*). Reports can also be scheduled to print after a specific process event occurs (i.e., *TRIGGERED REPORTS*). Either or both types of scheduling can be used. A print time is used to schedule periodic report printing; a time interval reschedules the report. A tag is monitored to start printing for a triggered report. The trigger can be the alarm condition of an analog type of tag or the alarm state or digital state of a digital type of tag.

Only one trigger condition can be processed at a time. Any new trigger that occurs while an alarm summary report is being processed is ignored regardless of whether the trigger is a specified time or a tag event.

Each summary report can present alarms for all alarm groups or only selected alarm groups if desired. Either a range of groups, a list of groups, or both can be defined for each report. Using both allows specifying a limited range, then up to ten additional groups (e.g., one to ten range and 12, 13, 19, 20, etc.).

Use the *Alarm Summary Report* option to schedule printing for all alarm summary reports and to define each report. Figure 9-14 shows the page used. To choose the option, first

press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

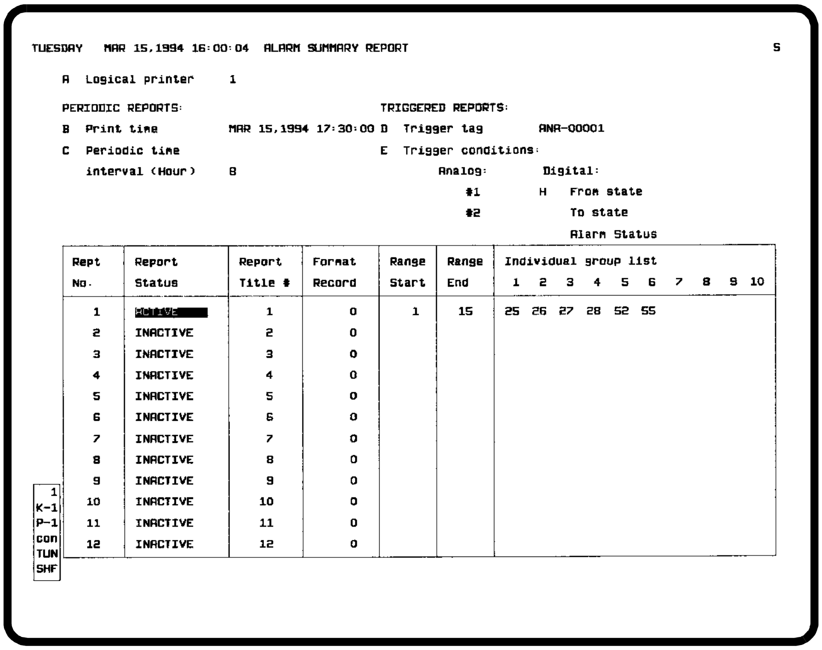
→ *E System*

→ *F Alarm Summary Report*

SCHEDULING REPORTS

To schedule alarm summary reports:

1. Enter **A** in the *Report number or parameter letter* field.



TUESDAY MAR 15,1994 16:00:04 ALARM SUMMARY REPORT S

A Logical printer 1

PERIODIC REPORTS:
B Print time MAR 15,1994 17:30:00
C Periodic time interval (Hour) 8

TRIGGERED REPORTS:
D Trigger tag ANA-00001
E Trigger conditions:
Analog: #1 #2
Digital: H From state To state Alarm Status

Rept No.	Report Status	Report Title #	Format Record	Range Start	Range End	Individual group list 1	2	3	4	5	6	7	8	9	10
1	ACTIVE	1	0	1	15	25	26	27	28	52	55				
2	INACTIVE	2	0												
3	INACTIVE	3	0												
4	INACTIVE	4	0												
5	INACTIVE	5	0												
6	INACTIVE	6	0												
7	INACTIVE	7	0												
8	INACTIVE	8	0												
9	INACTIVE	9	0												
10	INACTIVE	10	0												
11	INACTIVE	11	0												
12	INACTIVE	12	0												

1
K-1
P-1
con
TUN
SHF

TPS0008B

Figure 9-14. Alarm Summary Report Page

2. Press **ENTER**. This places the input cursor on the *Logical printer* field (Figure 9-14).

3. Enter the number of a logical printer to be used for printing reports. A valid entry is from 1 to 16.

4. In the *Print time* field, enter a time and date to start the first print. If using only tag triggering, clear this field.

5. In the *Periodic time interval (Hour)* field, enter the time interval (in hours) between printing of subsequent reports after the initial print. A valid entry is from 1 to 24. If using only tag triggering, clear this field.

6. In the *Trigger tag* field, enter the name or index number of the tag that is to trigger printing. This can be an analog or digital type of tag. If using only periodic triggering, clear this field.

7. Select a trigger condition.

If the trigger tag is analog, enter one or two alarm conditions for the tag that are to trigger the reports. A valid entry is N (return-to-normal), H, 2H, 3H, L, 2L, 3L, or A depending on the type of trigger tag. If using only periodic triggering, clear these fields.

NOTE: HF, HD, LD, HR, or LR cannot be used as a trigger condition.

If the trigger tag is digital, define the from or to state transition and the alarm status that are to trigger the reports. A valid *From state* and *To state* entry is the logic state descriptors set in the database for the trigger tag. A valid *Alarm status* entry is blank (no status triggering), ALARM, or RTN (return-to-normal). If using only periodic triggering, clear these fields.

8. If this was a change to the configuration of an existing report, press **ENTER**. If further configuration (i.e., individual report definitions) is required, move to the report fields at the bottom of the page and continue with the steps given in ***DEFINING REPORTS*** in this section.

DEFINING REPORTS

Each alarm summary report is defined separately. To define a report:

1. For an existing report, the *Report Status* field may indicate active. To edit the report, enter **INACTIVE** in the field. When initially defining, the report should already be inactive.

Making an existing report inactive takes that report off-line. The inactive report will not print if triggered. All other active reports, however, still print when triggered.

2. Enter the number of a title to use for this report in the *Report Title #* field. A valid entry is from 1 to 12. This is the title that was created earlier. Refer to ***Defining a Summary Report Title*** in this section.

3. Enter the number of a line format to use for this report in the *Format Record* field. This determines the line format for all entries in the report. A valid entry is from 0 to 105.

4. Enter the first alarm group in a range of groups in the *Range Start* field. A range can be specified with or without a list. Leave the field blank or clear it to not use a range.

5. Enter the last alarm group in a range of alarm groups in the *Range End* field. Leave the field blank or clear it to not use a range.

NOTE: Enter a range of 1 to S for all alarm groups.

6. Enter up to ten alarm groups that are to appear in this report in the *Individual group list* fields. A list can be specified with or without a range. Leave the fields blank or clear them to not use a list.

7. Turn the report on by entering **ACTIVE** in the *Report Status* field. Reports can be turned on or off individually and at any time.

8. Press **ENTER**.

GLOBAL ALARM MANAGEMENT

Some normal indications given for an alarm are an alarm tone, a flashing alarm group indicator, and a flashing alarm status/ quality/group field. When processing the alarm, the alarm tone must be silenced and the alarming tag must be selected and acknowledged to stop the flashing indications and tone. The acknowledge and silence are accomplished by pressing certain keyboard keys. An alarm can be manually inhibited and the scan mode can be turned off and on through the operating parameters page.

These alarm changes can be broadcast to other nodes over the communication highway. This can be used to silence, acknowledge, inhibit and change scan mode of an alarm on other consoles at the same time the alarm is changed on this console. This can also be used to change an alarm on this console from another console. Global alarm acknowledge and silence configuration is used to set up this plant wide alarm acknowledge and silence ability.

A node list determines to which INFI 90 Open nodes on the loop a change is broadcast. A node list that contains the first 32 N90STA tags in the database that define either a console or a computer is automatically compiled during start-up. This is node list zero. Additionally, up to four lists can be user-defined. Each list can contain a maximum of 32 node addresses. Each node that appears in a user-defined node list must be defined in the database as an N90STA tag. If a node in the list is not a console or computer, it is ignored and no message is sent to that node.

Each tag in the database can be defined to broadcast an alarm change to the nodes contained in one of the node lists. This enables broadcasting at the tag level. Silence broadcasting is enabled during alarm tone definition. Each tone can be defined to broadcast to a specific list of nodes. Acknowledge, silence, inhibit and scan on/off functions can all share the same node lists.

Each user-defined node list has a toggle that controls the broadcasting of messages to all nodes contained in the list. The node list toggle ability allows turning an entire list of nodes on or off. This prevents having to modify the database. If this toggle is set to *NO*, no messages are sent to the nodes in the list regardless of the broadcast setting in the tag database.

For N90STA type tags, alarm management attributes in the database act as toggles for turning alarm management broadcasting on or off for the node. The attributes determine if an alarm management message is broadcast to the N90STA node if its node address appears in a node list. This provides a means of disabling broadcasting for a single node defined in a node list without having to remove the node from the list. Alarm management functions are configured separately.

The following configurations are required to completely configure alarm management broadcasting:

- Node list configuration.
- Tag configuration.
- Alarm tone definition.
- General parameters configuration.

The console can be set up to receive or not receive broadcasts from other consoles and to send or not send broadcasts.

Defining a Node List

Up to four user-defined node lists can be defined. A node list specifies the nodes to which the console broadcasts acknowledges and silences. Each list can contain up to 32 nodes. The nodes must be consoles or computers connected to the loop. A node list can be selected for individual tags and tones.

There are two pages for defining node lists: Page one for list one and two, page two for list three and four. Use the *Global Alarm Management* option to define the node lists. Figure 9-15 shows the page used. To define node lists one and two, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *B Global Alarm Management*

To define node lists three and four, first press **NEXT** ant to return to node list one and two press.

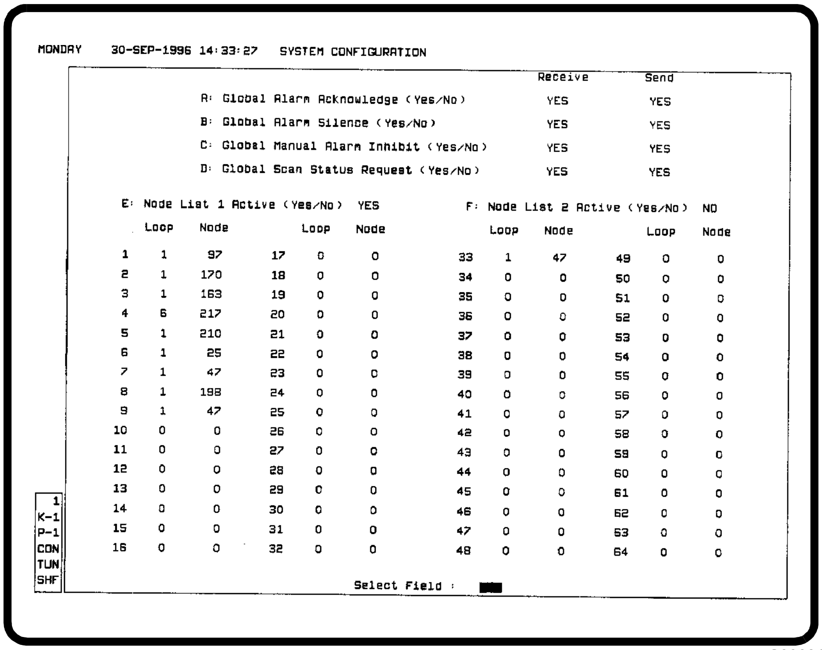


Figure 9-15. Global Alarm Management Page

Any change or addition made to a node list requires the OIS application to be reset. To define a node list:

1. In the *Loop Number* and *Node Number* fields, enter a loop and node number. A valid loop number is 0 to 250 for INFI-NET system and 0 or 1 for Plant Loop system. A valid node number is 0 to 250 for INFI-NET system and 0 to 63 for Plant Loop system. If a node specified in the list is not a console or computer (i.e., interface unit connecting the computer), it is ignored and no messages are sent to the node.

2. After all nodes are defined, press **ENTER**

3. Reset the OIS application by using the procedures given in ***Reset*** in Section 2.

If required, define both node lists on the page before resetting the console. Define the second pair of node lists after the console comes back on-line.

Tag Configuration

When setting up global alarm acknowledge, tag configuration is performed for two purposes. First, a node must be defined as an N90STA type tag before it can be added to a node list. Second, an attribute in the database enables alarm acknowledge broadcasting for a tag and another attribute defines the node list that is to be referenced when broadcasting.

Broadcasting must be enabled for each tag in the database individually. To enable broadcasting for a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter **YES** in the *Global Alarm Mgt* field to enable broadcasting alarm changes for the tag.

3. Enter **0** in the *Node List* field to direct broadcasts to the node list compiled by the console; enter **1**, **2**, **3**, or **4** to direct broadcasts to a specific user-defined list of nodes.

Alarm Tone Silence

Alarm tone configuration enables alarm tone silence broadcasting. Each tone must be enabled individually to have an alarm tone silence broadcast on the communication highway.

To enable broadcasting an alarm tone:

1. Follow the steps given in ***Defining a Logical Alarm Tone*** in this section.

2. Enter **YES** in the *BDCST* field for a tone to enable broadcasting an alarm silence for the tone.

3. Enter **0** in the *NODE* field to direct broadcasts to the node list compiled by the console; enter **1**, **2**, **3**, or **4** to direct broadcasts to a specific user-defined list of nodes.

Changing the Global Alarm Management Status

The *Global Alarm Management* option both enables and disables receiving and sending alarm changes on the communication highway. The console can be set up to transmit only, receive only, or transmit and receive alarm changes. This option also allows activating or deactivating a node list to selectively enable or disable broadcasts to a complete list of nodes.

Use the *Global Alarm Management* option to enable sending and receiving alarm changes. Figure 9-15 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *B Global Alarm Management*

GLOBAL ALARMING

The global alarm management receive and send abilities can be toggled on or off. To enable or disable these global features:

1. Enter **YES** in the *Receive* column of *Global Alarm Acknowledge* field to receive on this console any alarm acknowledges sent on the communication highway by other nodes. Enter **NO** to disable receiving alarm acknowledges.

2. Enter **YES** in the *Send* column of *Global Alarm Acknowledge* field to broadcast alarm acknowledges from this console. Enter **NO** to disable sending alarm acknowledges.

3. Enter **YES** in the *Receive* column of *Global Silence* field to receive on this console an alarm tone silence sent on the communication highway by other nodes. Enter **NO** to disable receiving alarm tone silences.

4. Enter **YES** in the *Send* column of *Global Silence* field to broadcast silences from this console. Enter **NO** to disable sending alarm tone silences.

5. Enter **YES** in the *Receive* column of *Global Manual Alarm Inhibit* field to receive on this console any manual alarm inhibit sent on the communication highway by other nodes. Enter **NO** to disable receiving manual alarm inhibits.

6. Enter **YES** in the *Send* column of *Global Manual Alarm Inhibit* field to broadcast manual alarm inhibits from this console. Enter **NO** to disable sending manual alarm inhibits.

7. Enter **YES** in the *Receive* column of *Global Scan Status Request* field to receive on this console any scan status request sent on the communication highway by other nodes. Enter **NO** to disable receiving scan status requests.

8. Enter **YES** in the *Send* column *Global Scan Status Request* field to broadcast scan status requests from this console. Enter **NO** to disable sending scan status requests.

If this node is sending an alarm function to another, the node that is to receive the alarm function must have its global receive ability enabled.

NODE LISTS

A node list must be made active before any broadcasts to nodes in that list can take place. Each node list can be activated or deactivated separately.

To activate a node list, enter **YES** In the *Node List n Active* field. Enter **NO** in the field to deactivate a list.

REMOTE ALARM ACKNOWLEDGE

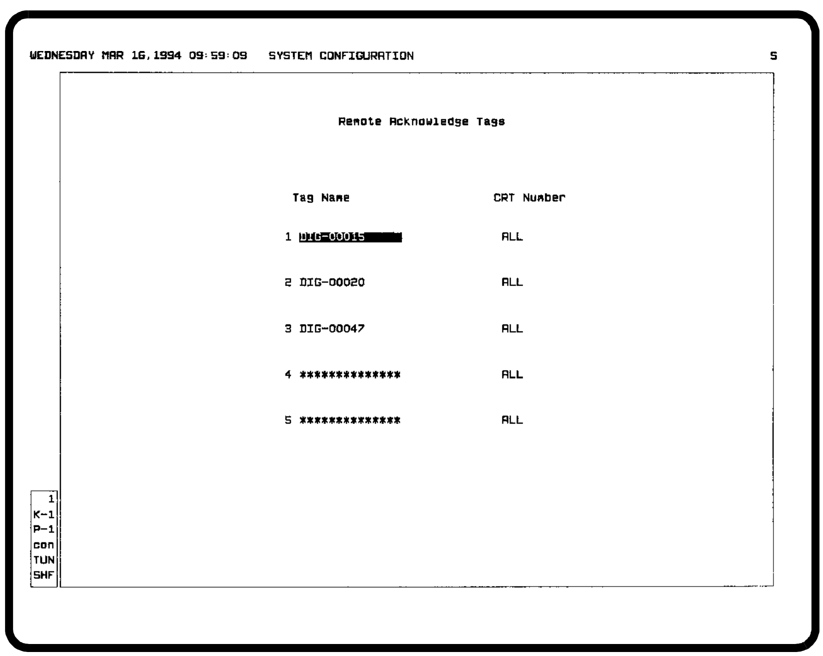
The *Remote Acknowledge Tag Assignments* option is used to define up to five tags that are to be used as remote acknowledge tags. A digital type of tag can be defined as a remote acknowledge tag.

Remote alarm acknowledgment can be used to acknowledge alarms on the main console and its auxiliary consoles, but only auxiliary consoles supported by this console. A remote acknowledge tag can be set up to acknowledge alarms on all or only a specific window. Using the same tag as the remote acknowledge tag on other consoles enables using this single tag for acknowledging alarms on several consoles at one time.

An alarm acknowledge occurs when the remote acknowledge tag changes to its one state. The tag must return to its zero state before another acknowledge can occur.

Use the *Remote Acknowledge Tag Assignments* option to define remote acknowledge tags. Figure 9-16 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *C Remote Acknowledge Tag Assignments*



TPS0051B

Figure 9-16. Remote Acknowledge Tag Assignments Page

The last step of this procedure is to reset the OIS application. Press **ESC** before pressing any other keys to exit this page without having to reset. To define a remote acknowledge tag:

1. Enter a tag name or index number of a digital type of tag in the *Tag Name* field.

2. Enter the window number for which this tag is to perform acknowledgment in the *CRT Number* field. A valid entry is 1 to 8 or ALL.

3. Reset the OIS application by using the procedures given in ***Reset*** in Section 2.

SECTION 10 - OPEN ACCESS SYSTEM

INTRODUCTION

An open access system (OAS) provides a centralized, plant-wide data acquisition and archival storage platform. An open access system is a node on the network. Access to an OAS node is provided to view trend data collected by the node and to archive logs generated by the console. The open access system is directly connected to both the INFI-NET or Plant Loop communication highway and the DECnet network. The OAS node acquires its real-time process data through its INFI-NET or Plant Loop connection. It uses the DECnet connection to transfer data between itself and the console and also for networking capabilities.

This section explains the console configurations necessary to identify and access an OAS node, retrieve trend data collected by the node, and set up log archiving to the node:

- OAS definition.
- Trend definition.
- Archival storage setup.

OPEN ACCESS SYSTEM DEFINITION

The OAS definition procedure is used to identify an OAS node as the log collection node and an OAS node as the trend data source node for the console. The same node or different nodes can be used for these purposes. The node names must be known to the console in its network database. Refer to ***NETWORK DEFINITIONS*** in Section 2 for further explanation.

Use the *OAS Definitions* option to define the log collection node and the trend data source node for the console. Figure 10-1 shows the page used. To call this page, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *I OAS Definitions*

To define the fields of this page:

1. Enter the name of the OAS node that is to receive archived events and logs generated by the console in the *Log Collection Node* field.

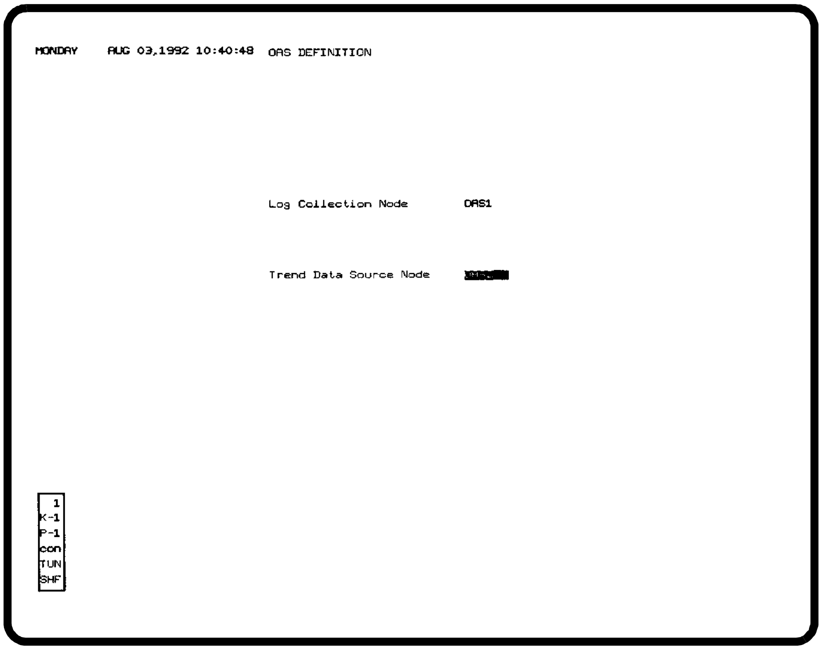


Figure 10-1. OAS Definitions Page

2. In the *Trend Data Source Node* field, enter the name of the node from which trend data is to be retrieved.

3. Press **ENTER**.

4. Reset the OIS application by using the procedures given in ***Reset*** in Section 2 to put the changes into effect.

TRENDS

The INFI-NET or Plant Loop communication highway access enables the open access system to collect and store trended data from the distributed trending system. The open access system has its own trend database to define its trend data collection responsibilities. Data collected by an open access system can be retrieved by the console to be viewed in trend displays. To do this, a remote trend (external source) must be defined in the console database. Transfer of data between an OAS node and the console is over the DECnet network.

For console purposes, remote trends are display only. When a trend display containing remote trends is called to the screen, a request is made to the open access system for any collected, historical data. Local data collection then begins on the console after receiving the data collected by the open access system. Local data collection ceases after removing the trend display. No data is stored by the console for those trends defined as external source.

DEFINING REMOTE TRENDS

To view remote trend data on the console, the same trend must be defined in the database of both the console and the open access system. The trend defined on the console must be an external source trend.

To define a remote trend:

1. Follow the steps given in ***Defining a Trend*** in Section 11. The procedures for defining a local and remote trend are the same.

2. The entry in the *Trend Usage* field determines whether the trend is a local or remote trend. Enter **EXTERNAL SRC** to define the trend as a remote trend.

3. In the *External Source Node* field, enter an OAS *nodename* or **1** to identify the node that is the trend data source for the trend definition. The *nodename* must match the name defined in the *External Source Node* field on the OAS definition page. Enter **1** to have the console automatically retrieve the node name from the OAS definition page.

LOGS

Most logs that can be generated by the console can be archived to an open access system. These include the system event log, custom logs, and SOE logs. After being archived, the open access system provides the ability to display or print the logs and to archive them to a storage medium for permanent storage. Logs archived to the open access system cannot be retrieved directly from the open access system using console archival retrieval procedures. They can, however, be retrieved directly from the storage medium to which the open access system archived them.

OAS ARCHIVAL STORAGE

Archiving of system events and logs to an open access system is set up in the same way as archiving to a storage medium. Turn on archiving for system events and logs using data type to volume definition. Then direct and schedule the archive to an OAS node.

To set up archival storage:

1. Turn on archiving of logs.

    a. Follow the procedures for using the *Data Type to Volume Definition* option of archiving. Refer to the discussion on data type to volume definition in the ***Operation*** instruction for the procedures (Table 1-2 lists instruction numbers). Also refer to the discussion on archiving in the instruction for an explanation of additional archival storage requirements.

b. Turn archiving on for the *Events* data type to archive the event log and the *Logs* data type to archive custom logs and SOE logs.

2. Enable and schedule archiving to the OAS.

a. Follow the procedures for using the *Volume to Media Definition* option of archiving. Refer to the discussion on volume to media definition in the ***Operation*** instruction for the procedures.

b. Enter **OAS** in the *Media Type* field.

c. Set the time span and time of day to schedule the archives. Archives automatically occur after being scheduled.

NOTE: Consider the amount of hard disk space available for temporary storage of archived data when scheduling an archive to an open access system.

SECTION 11 - TRENDS

INTRODUCTION

This section explains trend definition. To access and use PCU collected trend data in console trending functions, a trend must first be defined in the trend database. Up to 10,000 trends can be defined during trend definition. A definition provides information required to collect data from the INFI 90 OPEN distributed trending system.

Trending functions can be configured after establishing the trend database. Trending functions for the console are:

- Trend graph displays. Shows variables in a continuous line on a graph referenced to time.
- Tabular trend displays. Shows the variables in text format.
- Trend plot displays. Presents a set of process variables in a two-dimensional grid.
- Trend logs. Provide a hard copy of historical trend data.

Refer to the discussion on trend operations in the ***Operation*** instruction for additional information on trending and for an explanation of displays that present trended data (Table 1-2 lists instruction numbers).

TREND DATA SOURCE

One of two platforms can be the data source for trend data. The console can collect and store data locally on its hard disk or it can acquire the data from an open access system (OAS). Collecting data from an OAS node is remote data collection. A trend definition determines which occurs for a specific trend.

Local Trends

A local trend is used to collect and store data directly from a trend block in a PCU module. A local trend definition provides information required to establish communication between the trend block and console for data collection. It also identifies to the console the collection mode being used by the trend block.

Remote Trends

A remote trend is used to retrieve data collected by an open access system. In this case, the definition tells the console to which OAS node to direct a request for historical trend data.

A remote trend is a display only trend. The historical data for a trend display containing a remote trend comes from an open access system. The historical data that appears in a log for a remote trend also comes from the open access system. After receiving the historical data, the console then begins local trending based on the trend defined in its database. Local data collection continues until the trend display is removed or the log completes its data collection.

STANDARD TREND

A standard trend is a trend that utilizes function code 66 (FC 66) to perform data collection. An analog exception reporting point can be trended using a trend block in a PCU module. The trend block is always associated with an exception reporting block. The PCU module performs the initial trend data compression and calculations (e.g., average, sum, etc.) for data it collects. This data is eventually sent to the console. Digital values can also be trended, but require conversion from a digital value to an analog (real) value.

For a standard trend, the length of time trend data is stored by the console can be adjusted from hours to 92 days. This is the maximum amount of historical data that can appear on a display. Any data displayed past this time span comes from the archiving function.

A characteristic determined by the PCU module is the mode of collection for a trend. For a standard trend block the collection modes are sample, maximum, minimum, sum, or average. This determines the final value that a module sends to the console based on the values seen during the resolution interval (i.e., 15 seconds or one minute). Table 11-1 lists and describes the trend modes for a standard trend.

The standard trend block collects data for a single process variable and can implement only one mode of collection. Only one trend definition can reference a standard trend block. To perform both maximum mode and minimum mode trending for a process point requires two trend blocks for the point.

Table 11-1. Trend Modes (Standard Trend)

Mode	Description
Average	Average of values collected during the 15 seconds or 1 minute interval.
Maximum	Maximum value seen during the 15 seconds or 1 minute interval.
Minimum	Minimum value seen during the 15 seconds or 1 minute interval.
Sample	Saves a value every 15 seconds or 1 minute.
Sum	Total of values collected during the 15 seconds or 1 minute interval.

Figures 11-1, 11-2, and 11-3 show the PCU module configurations required for standard trending. Refer to the ***Function Code Application Manual*** for information about the function codes used (Table 1-2 lists instruction numbers). ***TRENDING EXAMPLES*** contains examples of standard trends.

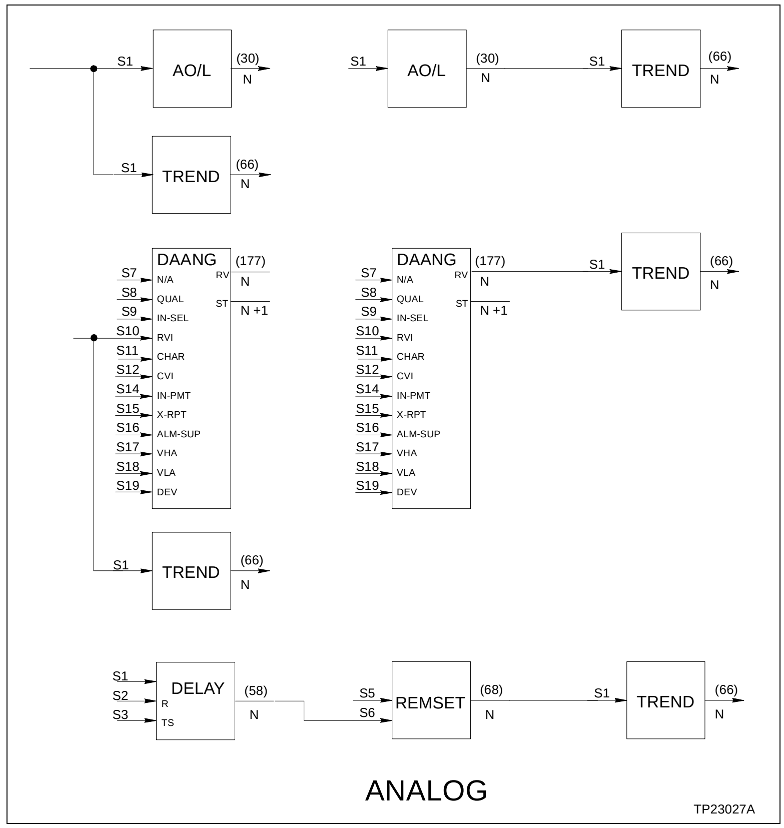


Figure 11-1. Standard Trend, Analog Block Configuratio

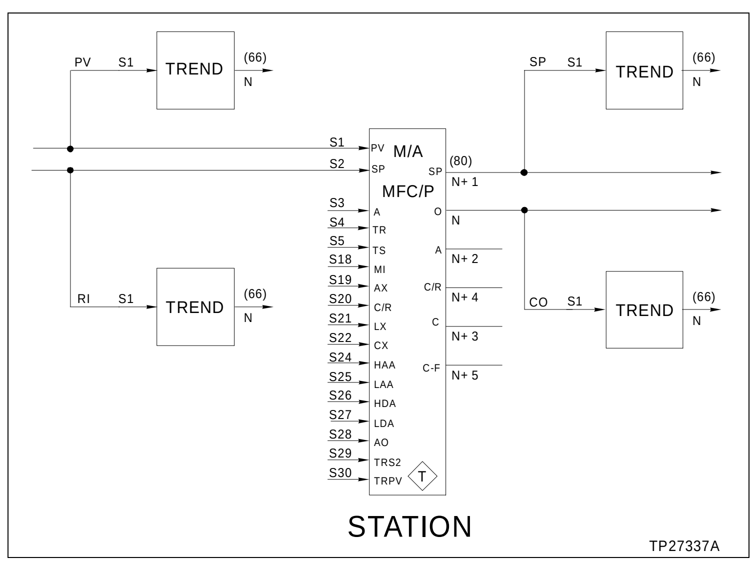


Figure 11-2. Standard Trend, Station Block Configuration

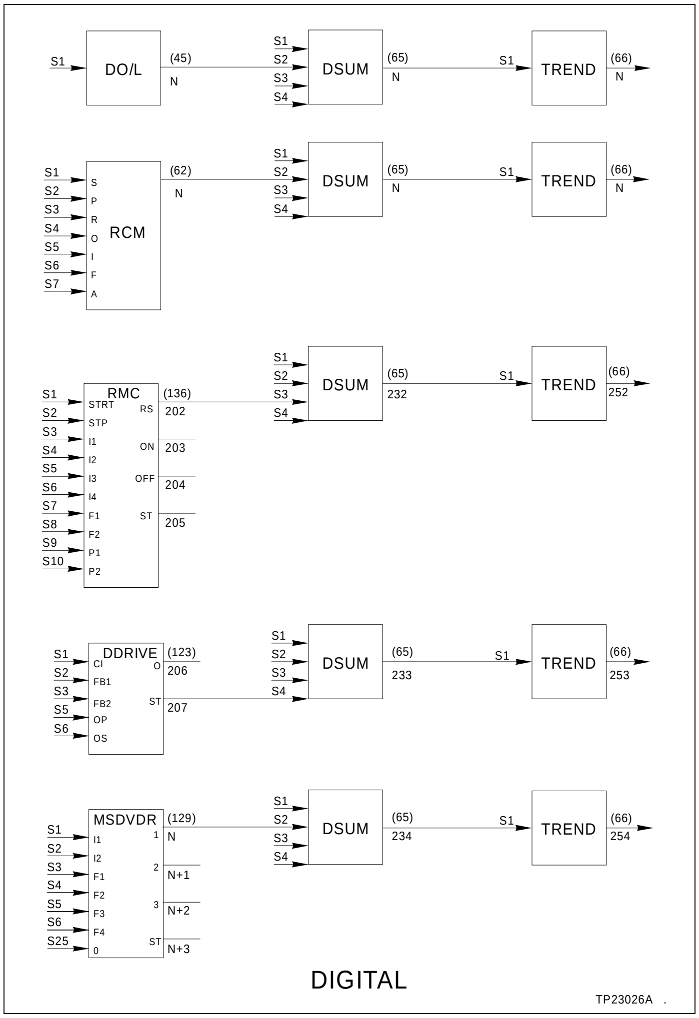


Figure 11-3. Standard Trend, Digital Block Configuration

ENHANCED TREND

An enhanced trend is a trend that utilizes function code 179 (FC 179) to perform data collection. The console must be operating on the INFI-NET system to use enhanced trending. With this type of trending, the exception reporting function blocks that can be trended include:

- Analog exception report (FC 30).
- Digital exception report (FC 45).
- Remote control memory (FC 62).
- Remote manual set constant (FC 68).
- Control station (FC 80).
- Device driver (FC 123).
- Multi-state device driver (FC 129).
- Remote motor control block (FC 136).
- Data acquisition analog (FC 177).
- Data acquisition digital (FC 211).

The enhanced trend block records analog values and digital states from these function blocks and also records alarm status and quality. The PCU module performs the initial trend data compression and calculations (e.g., average, sum, etc.) for data it collects. This is the data that is eventually sent to the console.

The amount of enhanced trend data that the console stores on disk can be adjusted. This is the maximum amount of historical data that can appear on a display and is set as a number of events. Any data that displays past this number of events comes from the archiving function.

A characteristic determined by the PCU module is the mode of collection for a trend. For an enhanced trend block the collection modes are sample, maximum, minimum, sum, average, or range. This determines the final value that a module sends to the console based on the values seen during a statistical time period. Table 11-2 lists and describes the trend modes for an enhanced trend.

Table 11-2. Trend Modes (Enhanced Trend)

Mode	Description
Average	Average of all values collected during the period set in the trend block.
Maximum	Maximum value seen during the period set in the trend block.
Minimum	Minimum value seen during the period set in the trend block.
Range	Maximum minus minimum samples collected over the period set in the trend block.
Sample	Value based on the input sampling time set in the trend block.
Sum	Total of all values collected during the period set in the trend block.

The enhanced trend block collects data for all variables associated with a function block being trended. For most function blocks, this is normally a single variable that is the block output. For a station block, however, the enhanced trend block collects process variable (PV), set point (SP), ratio index (RI), and control output (CO) values.

The enhanced trend block can also implement all or any combination of its six collection modes (i.e., sample, average, sum, etc.) when collecting data. Up to six trend definitions can reference a single enhanced trend block. A trend can be defined for each of six trend modes for a nonstation block. For a station block, up to six trends can be defined in any combination of data types (i.e., PV, SP, RI and CO) and trend modes. To trend all data types of a station block using all six trend modes for each data type requires four enhanced trend blocks.

Figures 11-4, 11-5, and 11-6 show the PCU module configurations required for enhanced trending. Refer to the ***Function Code Application Manual*** for information about the function codes used (Table 1-2 lists instruction numbers). ***TRENDING EXAMPLES*** at the end of this section contains examples of enhanced trends.

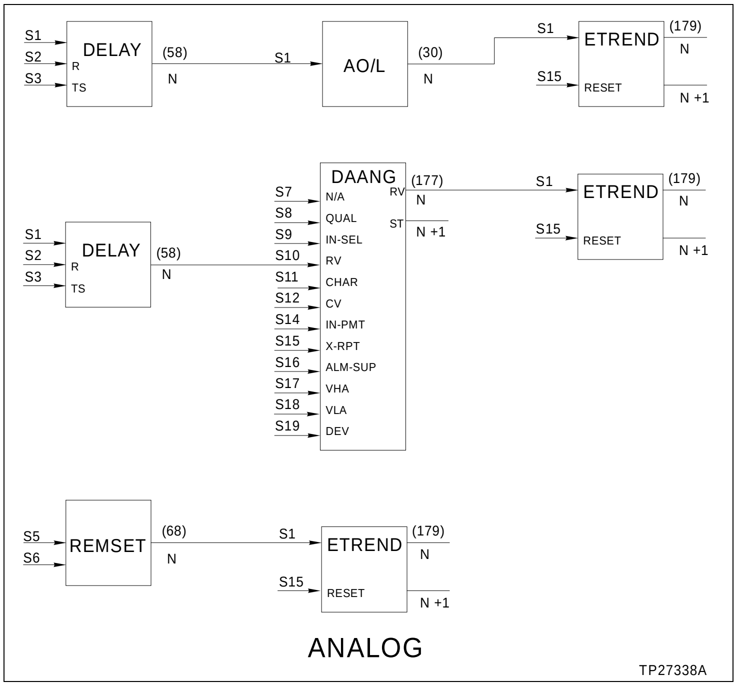


Figure 11-4. Enhanced Trend, Analog Block Configuration

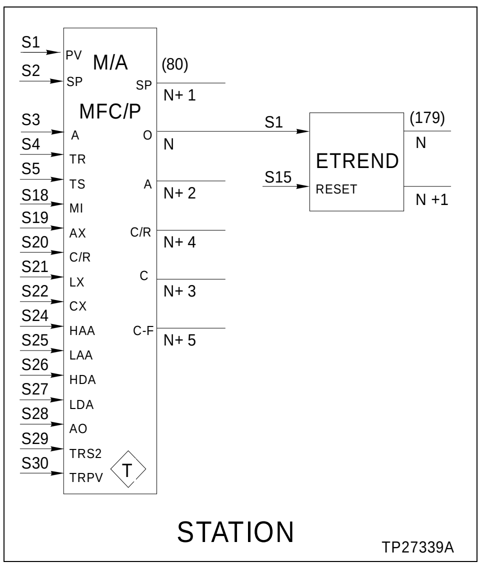


Figure 11-5. Enhanced Trend, Station Block Configuration

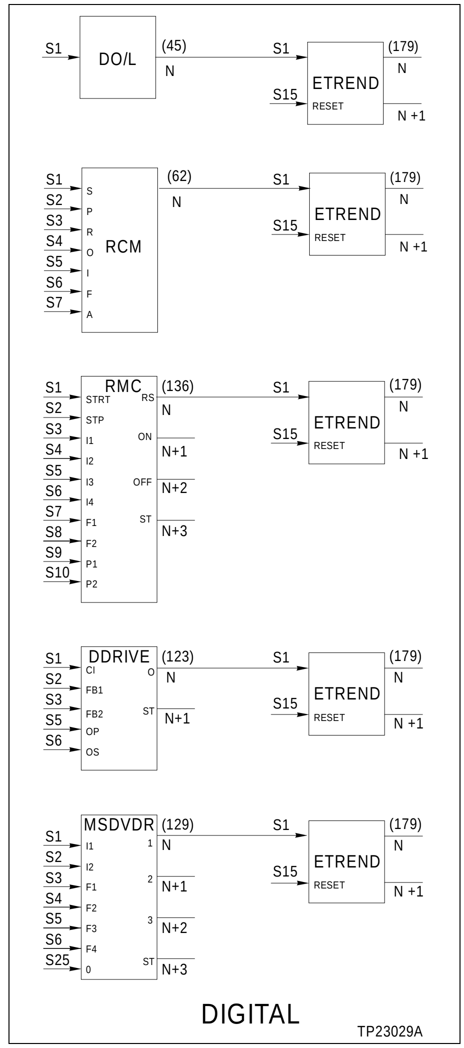


Figure 11-6. Enhanced Trend, Digital Block Configuration

DISPLAY RESOLUTION

The display resolution for a trend element determines how often a trend display is updated. The display resolution for a standard trend is 15 seconds or one minute. The display resolution for an enhanced trend can very depending on the collection resolution of the block.

For an enhanced trend, the trend definition defines an initial display resolution used for a trend element that is part of an operator configurable display. This does not affect a user-created trend display since the resolution is defined when the display is created.

The display resolution set for a display does not have to be the same as the collection resolutions of the trends on the display. The display resolution, however, should be the least common multiple or any multiple of the trend collection resolutions. For example, if an enhanced trend block is set for two-second collection resolution, then the display resolution can be set to ten seconds instead.

If an operator configurable display contains all enhanced trends, the display resolution is ignored if it is not the least common multiple or any multiple of ***all*** trends on the display. The display resolution is set as the least common multiple of all the trend collection resolutions instead. This is only true for operator configurable displays. A user-created trend display presents an error message if the display resolution is not the least common multiple or any multiple of ***all*** trend collection resolutions.

TREND DATABASE

Each trend the console processes has a trend definition in the database. A trend definition is required to collect trend data locally from a PCU module and remotely from an open access system.

> **NOTE:** Trend definitions are required to retrieve data collected by the distributed trending system. Operator assignable trends do not use trend definitions to acquire data. Refer to the ***Operation*** instruction for an explanation of operator assignable trends (Table 1-2 lists instruction numbers).

To define a local or remote trend, both an exception reporting block and a trend block must be in the control configuration of a PCU module. A tag must also be defined in the tag database for the exception reporting block. The tag is used to display real-time values and attributes along with historical trend data. A tag as part of a trend definition creates a link between an exception reporting block and a trend block.

Additionally, the OAS definition procedure must be performed to identify the data source node for remote trends. The node name is then entered during definition of a trend.

The console database supports up to 10,000 trend definitions. Of these, it allows for a maximum of:

- 10,000 with a *DISPLAY ONLY* usage type.
- 5,000 with an *EXTERNAL SRC* usage type.
- 2,000 with a *SAVE TO DISK* or *ARCHIVE* usage type.

The maximum for remote trends (*EXTERNAL SRC*) is 5,000 since the open access system supports a maximum of 5,000 trend definitions. The trend database size can be adjusted during system configuration.

A combination of one-minute standard, 15-second standard, and enhanced trends affects the maximum number of *SAVE*

TO DISK or *ARCHIVE* trends that can be processed. The maximum number of enhanced trends is not known at this time.

Refer to Table 11-3 for guidelines to determine the number of trends the console is capable of processing. This table assumes only standard trends are defined and all have a *SAVE TO DISK* or *ARCHIVED* trend usage type.

Table 11-3. Trend Capabilities for Standard Trends

One Minute Trends	15 Second Trends	Total System Trends
2,000	0	2,000
1,800	100	1,900
1,600	200	1,800
1,400	300	1,700
1,200	400	1,600
1,000	500	1,500
800	600	1,400
600	700	1,300
400	800	1,200
200	900	1,100
0	1,000	1,000

Sizing the Trend Database

Not all configurations require the maximum number of trend definitions. The database size (i.e., number of trend definitions) should be large enough to accommodate the current trending requirements and be increased in size as necessary.

To set the maximum number of trends:

1. Follow the steps given in ***GENERAL PARAMETERS*** in Section 3.

2. Enter the number of trend definitions currently required in the *Max Number of Trends* field. A valid entry is from 0 to 10000. This number should be close to, but greater than the actual number of trend definitions that will be created.

> **NOTE:** Decreasing the number of trends erases all trend definitions that have index numbers greater than the new maximum value.

This procedure can be used to initially set or subsequently change the database size. Hard disk space is allocated based on the value entered. Entering a value that reflects the current database requirements frees unused disk space for other functions. The number of trends can be increased at any future time to account for additional process requirements.

Defining a Trend

Several attributes define a trend. The attributes that must be specified for a trend definition depend on its usage type. Two attributes that must be defined for all trends are trend type and trend mode. The type and mode must match the type and mode configured for the trend block in a PCU module. These entries inform the console of the type of trend block being referenced and the collection method being used by the block. Local trending is implemented after calling a trend display containing remote trends or when a log is collecting data. The attributes defined for a remote trend must still match the PCU module trend block to correctly perform local trending.

NOTES:
1. The order in which each PCU module is polled for trend data is determined by the console and not the trend index number order.

2. Enhanced trends can be defined only if the console is operating on an INFI-NET system.

3. Up to six trend definitions can reference a single enhanced trend block.

Additional trend definition attributes pertain to:

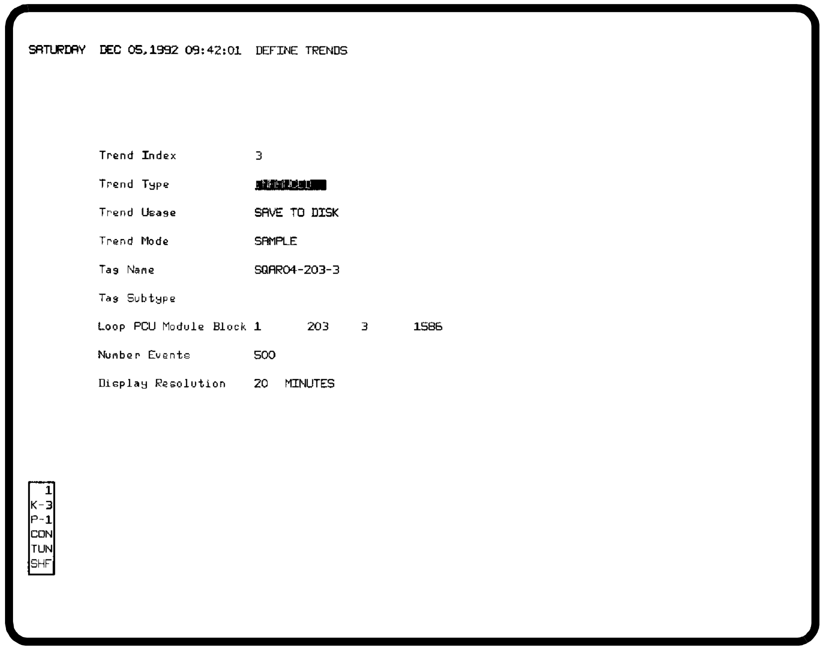
- Archiving.
- Data storage.
- Displays.
- STATION tag trends. A subtype identifies which STATION tag variable is being trended.

This procedure must be completed ***before*** performing any other trending configuration. It establishes index numbers for trends which are used in all other trending functions.

NOTE: The preferred method for creating the trend database is by using the console configuration tools (WLDG). The console should mainly be used to update or make changes to the existing database.

Use the *Trend* option to define trends. Figure 11-7 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *D Trend*



TPS0232C

Figure 11-7. Trend Page

To define a trend:

1. Select the trend to define.

   a. Initially, the input cursor positions on the *Trend Index* field. Enter the index number of a trend definition. A valid entry is from 1 to 10000. The maximum may actually be something less than 10000 depending on the number of trends enabled during system configuration.

   b. Press **ENTER**.

2. Enter the appropriate data into each field of the page. Refer to Table 11-4 for field explanations.

3. Press **ENTER**.

Table 11-4. Trend Definition Page Fields

Field	Description
Trend index	Shows the trend index number and calls a trend definition. A valid entry is from 1 to 10000.
Trend type	ENHANCED = enhanced trend (FC 179). FAST = 15-second standard trend (FC 66). NORMAL = one-minute standard trend (FC 66). UNDEFINED = use to remove a trend definition. **NOTE:** Must match the type of trend block being referenced.

Table 11-4. Trend Definition Page Fields (continued)

Field	Description
Trend usage	ARCHIVED = same as SAVE TO DISK except that it also enables saving data for archiving. The data is temporarily stored to an archival storage directory for eventual transfer to storage medium. This type of trend is a local trend. DISPLAY ONLY = no data is stored for this type of trend. The amount of historical data that can be accessed is limited to the amount of data the PCU module can store. This type of trend is a local trend. EXTERNAL SRC = remote trend. Requires an entry in the *External Source Node* field. No data is stored by the console for this type of trend. The open access system is responsible for collecting and storing trend data. SAVE TO DISK = historical trend data is stored for display and logging. This type of trend is a local trend.
Trend mode	Collection mode being implemented by the trend block. For a ***standard trend***, a valid entry is: AVERAGE SAMPLE MAXIMUM SUM MINIMUM For an ***enhanced trend***, a valid entry is: AVERAGE RANGE MAXIMUM SAMPLE MINIMUM SUM **NOTE:** Must match the trend mode defined in the PCU module trend block.
Tag name	Name or index number of a tag. This must be the tag that references the function block that sources the values to the trend block. This is used for display and archiving purposes. For a standard trend, a tag is recommended but not required. A tag is required for an enhanced trend.
Tag subtype	Required when trending a control station (STATION tag). It identifies which station variable is being trended. A valid entry is: CO = control output. PV = default; process variable. RI = ratio index. SP = set point.
Loop, PCU, module, block	Hardware loop, PCU, and module number and software function block number of the trend block in a PCU module. This is ***not*** the address of the function block sourcing the data.
Time span	For ***standard trends*** only; sets the maximum amount, limited by time, of historical trend data to be saved. The *Time Span* requires two entries: A number and HOURS or DAYS. The maximum time span is 92 days.
Number events	For ***enhanced trends*** only; sets the maximum number of events to be saved to saved. This setting depends on the type of event being saved, the anticipated duration between samples, and the amount of free disk space available for storage. To calculate disk space usage, each event consumes: ANALOG - 12 bytes DANG - 24 bytes DADIG - 10 bytes Digital type - 10 bytes RMSC - 12 bytes STATION - 24 bytes.

Table 11-4. Trend Definition Page Fields (continued)

Field	Description
Display resolution	For ***enhanced trends*** only; sets the display resolution for a trend element in an operator configurable display. It is only used by the operator configurable displays function. The *Display Resolution* requires two entries: A number and SECONDS, MINUTES, or HOURS. The minimum resolution period is one second; the maximum is eight hours. **NOTE:** The operator configurable displays function sets the display resolution of a trend element to this resolution only if it is a multiple of the collection resolutions for all trends in that element. If it is not, the display resolution that will be used is the least common multiple of all trend collection periods.
External source node (not shown in Figure 11-7)	OAS node from which the console is to retrieve trend data. A valid entry is: *nodename* = OAS node providing trend data. 1 = node name is retrieved from the OAS definition function.

Archiving a Trend

If archiving a standard trend, the time span set in the definition for the trend must be at least the archive time span plus two hours. This provides some margin between trend storage and archive storage to prevent loss of archived data.

If archiving an enhanced trend, the time span for archiving must be set to a value equal to or greater than the maximum event recording time set in the module configuration, which is set with S8 of FC 179. This will guarantee that at least one new event will occur during an archival period. If it is not set this way, gaps may appear in the trend line for historical data.

DISPLAY REQUIREMENTS

Trended data can appear on the console in either a trend display or XY plot. Trends can appear in a graphic overview, graphic detail, or group display. A trend can also appear in an operator configurable display. The index number of a trend definition is used to include a trend in a display.

Trend Display

Standard trend displays are provided.Two types of trend displays are provided - standard and custom. With standard trend displays the trend elements are limited to a number of set sizes. The custom trend displays are different in that: the size of the trend element can be set to any desirable dimensions so long as it is not larger than the allowable maximum display size (i.e. full size), and the trend data can be presented on the display a number of ways (left to right, right to left, top to bottom and bottom to top). Whether using standard elements or creating custom elements, the escape commands in the source file of a trend display must be edited to reference the index numbers of trend definitions. A trend definition allows the console to acquire trended data, then present this data in displays. Refer to Section 7 for the procedures to modify or create displays.

The display resolution for a trend element determines how often the trends on the display are updated. This is set as one of the parameters of a trend element escape command (**et 53** for standard, **et 153** for custom). Also, a display resolution is entered in the definition of an enhanced trend. This display resolution, however, is only used by the operator configurable displays function.

The display resolution must either match the collection resolution of ***all*** trends on the display or must be the least common multiple or any multiple of ***all*** trend collection resolutions. For example, if a trend element contains five standard trends and four are 15-second (fast) trends and one is a one-minute (normal) trend, the least common multiple to use as the display resolution is one minute. Two minutes, three minutes, four minutes, etc., can also be used since they are still valid multiples.

NOTE: For standard trends, the least common multiple is recommended for display resolution. The display resolution can, however, be set to 15 seconds even if the trend element contains a one-minute trend. This is only true for standard trends.

If a trend element contains three enhanced trends with collection resolutions of two seconds, three seconds, and four seconds respectively, the least common multiple to use as the display resolution is 12 seconds. However, 24 seconds, 48 seconds, 72 seconds, etc., are still valid.

XY Plot

Standard XY plot displays of various set sizes are provided. XY plots require additional configuration to display trended data or other types of data. A plot index number that defines the variables to display must be referenced in the XY plot display source file. XY plot configuration establishes these plot index numbers. Refer to Section 17 for the procedures.

NOTE: An XY plot does not support enhanced trends.

TRENDING EXAMPLES

Both a PCU module and the console must be setup to trend process points. In the module, a trend block must be added for each point that is to be trended. The console requirements are as follows:

- Tag for each point being trended (always recommended, but only required for enhanced trends or if trended values are to be archived).
- Trend definition for each point being trended.
- Displays created to present trended values (only required if trended values are to be displayed).
- Logs created to record trended values (only required if hard copy record is desired).
- Archiving setup to store trended values (only required if trended values are to be archived for permanent storage).

This section includes two trending examples: A standard trend example and an enhanced trend example. Both examples describe module configuration, tag configuration, trend definition, and display creation requirements. Logging and archiving are not included in the examples. The examples do not give step-by-step procedures, but rather explain the settings required in each configuration to successfully trend the process points.

Module Configuration

A PCU module is responsible for collecting and storing the data the console displays. Before the module can collect and store data for trending, it must be properly configured. A module can be configured from the console or by using the module configuration tools (WCAD) program on an engineering work station. Refer to the ***Module Configuration Tools*** instruction for an explanation of module configuration utilities (Table 1-2 lists instruction numbers).

Tag Configuration

A tag is required for an enhanced trend. A tag is recommended but not always required for a standard trend. A tag is required for a standard trend, however, to drive the alarm status/quality/group field normally found on a trend display and for archiving. The tag references an exception reporting block, and not a trend block, in a PCU module. Tags can be created from the console or by using the console configuration tools (WLDG) on an engineering work station. Refer to ***Defining a Tag*** in Section 6 for the procedures to create a tag from the console. Refer to the ***Console Configuration Utilities*** instruction for an explanation of using the console configuration utilities to create tags.

Trend Definition

A trend definition is required on the console for each process point being trended. The definition references a trend block in a PCU module. Trend definitions can be created from the console or by using the console configuration tools (WLDG) on an engineering work station. Refer to ***Defining a Trend*** in this section for the procedures to create a trend definition from the console. Refer to the ***Console Configuration Utilities*** instruction for an explanation of using the console configuration utilities to create trend definitions.

Display Creation

A trend display must be created to present trended values on a console. The following examples use a single, full size display to present the data for the trends described. Normally, a display is created by using the graphical display configuration (GDC) of the console configuration tools program. Refer to the ***Console Configuration Utilities*** instruction for an explanation of the SODG utility.

Standard Trend (FC 66)

The following example describes the configurations required to trend and display the values for four process points using standard trend blocks (FC 66). Three of the process points are the process variable (PV), set point (SP), and control output (CO) of a station. A sample of these values is to be taken every 15 seconds. The remaining process point is an analog point. Both maximum and minimum statistical trending values are to be collected for this point. The statistical period (interval) is one minute.

MODULE CONFIGURATION

Table 11-5 describes the trend block specifications for this example. Figure 11-8 shows the block configurations.

Table 11-5. Example 1 - Trend Block (FC 66) Specifications

Block Address	Specification		
	S1	S2	S3
200	100	0 (sample)	1 (15 sec)
201	101	0 (sample)	1 (15 sec)
202	90[1]	0 (sample)	1 (15 sec)
203	110	3 (maximum)	0 (1 min)
204	110	2 (minimum)	0 (1 min)

NOTE:

1. Address of the block providing the PV input.

TAG CONFIGURATION

The exception reporting blocks in this example are one control station (FC 80) block and one analog exception report (FC 30) block (Figure 11-8). Two tags are required in this example: One for the station block and one for the analog exception report block. Table 11-6 describes the configuration of each tag. The table describes only the fields necessary for this example.

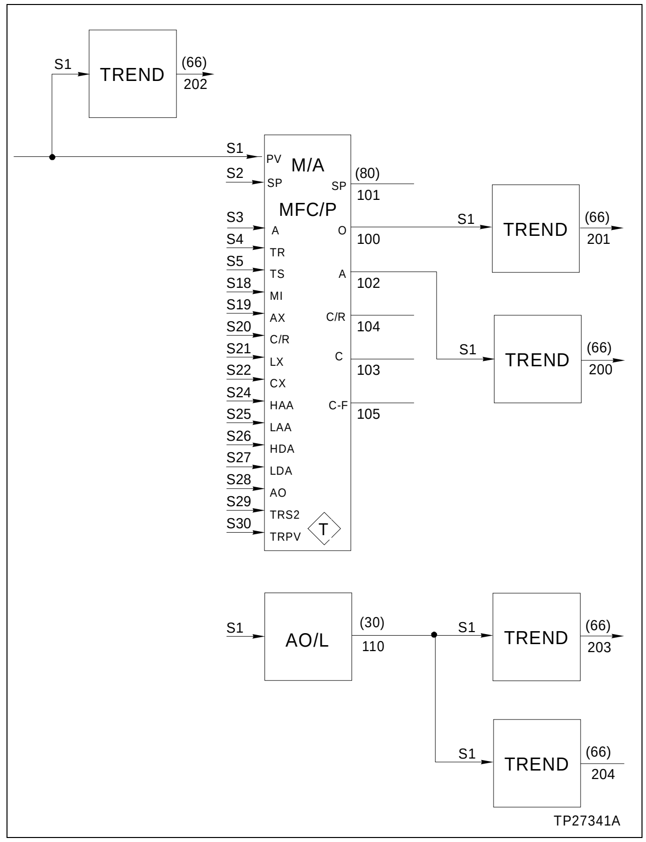


Figure 11-8. Example 1 - Trend Block (FC 66) Configuration

Table 11-6. Example 1 - Tag Configurations

Field	Trend				
	1	2	3	4	5
Tag index	500			510	
Tag name	STA-00500			ANA-00510	
Tag descriptor	STATION TAG1			ANALOG TAG1	
Loop, PCU, module, block	1-1-5-100			1-1-5-110	

TREND DEFINITION

Five trend definitions are required in this example. Table 11-7 describes each definition. The trends are being saved to disk with a maximum of 12 hours of data being stored for each trend in this example.

Table 11-7. Example 1 - Trend Definitions

Field	Trend				
	1	2	3	4	5
Trend index	600	601	602	603	604
Trend type	FAST	FAST	FAST	NORMAL	NORMAL
Trend usage	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK
Trend mode	SAMPLE	SAMPLE	SAMPLE	MAXIMUM	MINIMUM
Tag name	STA-00500	STA-00500	STA-00500	ANA-00510	ANA-00510
Tag subtype	CO	SP	PV	N/A	N/A
Loop, PCU, module, block	1-1-5-200	1-1-5-201	1-1-5-202	1-1-5-203	1-1-5-204
Time span	12 HOURS	12 HOURS	12 HOURS	12 HOURS	12 HOURS

NOTE: N/A = not applicable.

DISPLAY

All five trends in this example are to appear in a single trend display. Table 11-8 describes the display requirements necessary to properly display the example trends as described.

Table 11-8. Example 1 - Display Requirements

Attribute	Requirement	
Size	Full	
Display resolution (period)	1 minute	
Type	Trend 1	CO
	Trend 2	SP
	Trend 3	PV
	Trend 4	AN
	Trend 5	AN
Tag name	Trend 1	STA-00500
	Trend 2	STA-00500
	Trend 3	STA-00500
	Trend 4	ANA-00510
	Trend 5	ANA-00510
Tag index	Trend 1	500
	Trend 2	500
	Trend 3	500
	Trend 4	510
	Trend 5	510
Trend index	Trend 1	600
	Trend 2	601
	Trend 3	602
	Trend 4	603
	Trend 5	604

Enhanced Trend (FC 179)

The following example describes the configurations required to trend and display the values for four process points using enhanced trend blocks (FC 179). Three of the process points are the process variable (PV), set point (SP), and control output (CO) of a station. A sample of these values is to be taken every 30 seconds. The remaining process point is an analog point. Both maximum and minimum statistical trending values are to be collected for this point. The statistical period (interval) is one minute.

MODULE CONFIGURATION

Figure 11-9 shows the block configurations for this example. Table 11-9 describes the trend block specifications. The default was used for those specifications not shown in the table.

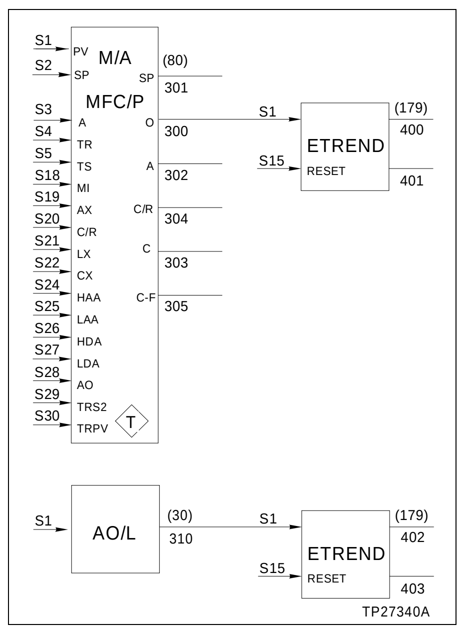


Figure 11-9. Example 2 - Trend Block (FC 179) Configuration

Table 11-9. Example 2 - Trend Block (FC 179) Specifications

Block Address	Specification					
	S1	S2	S3	S6	S9	S10
400	300	4 (FC 80)	1 (sample)	30	*	*
402	310	0 (FC 30)	6 (maximum/ minimum)	0	1	1

NOTE: * = not required for sampling mode; left at default.

TAG CONFIGURATION

The exception reporting blocks in this example are one control station (FC 80) block and one analog exception report (FC 30) block (Figure 11-9). Two tags are required in this example: One for the station block and one for the analog exception report block. Table 11-10 describes the configuration of each tag. The table describes only the fields necessary for this example.

Table 11-10. Example 2 - Tag Configurations

Field	Trend				
	1	2	3	4	5
Tag index	900			910	
Tag name	STA-00900			ANA-00910	
Tag descriptor	STATION TAG2			ANALOG TAG2	
Loop, PCU, module, block	1-1-5-300			1-1-5-310	

TREND DEFINITION

Five trend definitions are required in this example. Table 11-11 describes each definition. The trends are being saved to disk with a maximum of 1,000 events being stored for each trend in this example.

Table 11-11. Example 2 - Trend Definitions

Field	Trend				
	1	2	3	4	5
Trend index	1000	1001	1002	1003	1004
Trend type	ENHANCED	ENHANCED	ENHANCED	ENHANCED	ENHANCED
Trend usage	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK	SAVE TO DISK
Trend mode	SAMPLE	SAMPLE	SAMPLE	MAXIMUM	MINIMUM
Tag name	STA-00900	STA-00900	STA-00900	ANA-00910	ANA-00910
Tag subtype	CO	SP	PV	N/A	N/A
Loop, PCU, module, block	1-1-5-400	1-1-5-400	1-1-5-400	1-1-5-402	1-1-5-402
Number events	1000	1000	1000	1000	1000
Display resolution	1 MINUTES	1 MINUTES	1 MINUTES	1 MINUTES	1 MINUTES

NOTE: N/A = not applicable.

DISPLAY

All five trends in this example are to appear in a single trend display. Table 11-12 describes the display requirements necessary to properly display the example trends as described.

Table 11-12. Example 2 - Display Requirements

Attribute	Requirement	
Size	Full	
Display resolution (period)	1 minute (least common multiple of all trends)	
Type	Trend 1	CO
	Trend 2	SP
	Trend 3	PV
	Trend 4	AN
	Trend 5	AN
Tag name	Trend 1	STA-00900
	Trend 2	STA-00900
	Trend 3	STA-00900
	Trend 4	ANA-00910
	Trend 5	ANA-00910
Tag index	Trend 1	900
	Trend 2	900
	Trend 3	900
	Trend 4	910
	Trend 5	910
Trend index	Trend 1	1000
	Trend 2	1001
	Trend 3	1002
	Trend 4	1003
	Trend 5	1004

SECTION 12 - LOGGING

INTRODUCTION

This section explains log configuration. Logging provides a means of recording process data, then making a hard copy printout. The types of logs the console can generate include custom, system events (and operator actions), sequence of events (SOE), and alarm summary reports. Each type of log is defined separately.

NOTE: This section explains custom, system events, and SOE log configuration. Refer to ***ALARM SUMMARY REPORT*** in Section 9 for an explanation of alarm summary report configuration.

The custom and SOE logs can be set up to save up to nine most recently generated reports. The event log can save up to 1,000 most recent events. Refer to the discussion on printing and displaying logs in the ***Operation*** instruction for procedures to demand a printout of these retentions (Table 1-2 lists instruction numbers).

SYSTEM EVENT LOG

The system event log records process events such as digital tag, analog tag, console operator action, and module status events. The events normally appear in a single system event log. Events can be separated to appear in two distinct logs if desired: Event log and operator actions log. The event log records process and system events and the operator actions log records operator control and alarm management events.

Three configuration procedures must be performed to enable system event logging:

- Tag configuration to enable event recording for individual tags.
- Event log configuration.
- Event log format configuration.

Scheduling a Printout

The system event log can print continuously to a printer or the screen and can print periodically to a printer. Continuous printing sends event data to the printer or screen every time a new event occurs. This provides a near real-time, dynamic printout. In this case, entries are sent line-by-line. For periodic printing, events are collected over a certain period of time, then the entire log is formatted before being sent to the printer.

Events must be saved for periodic printing. Events that are to print continuously do not have to be saved. Conversely, not all events saved have to be printed continuously. This is determined on a per tag basis.

Event log display or printing can be operator demanded at any time by using the log by name function regardless of the printing schedule defined for the log. The function can display either the saved data or can display in real-time mode. The function can only demand a printout of saved data, however. Refer to the discussion on logging in the ***Operation*** instruction for the procedures to demand an event and operator actions log printout (Table 1-2 lists instruction numbers).

Configuring a Tag

In the tag database, events are classified into three different general categories: Alarms, state changes, and operator actions. Recording of each of these different categories of events is enabled separately for a tag. For example, a tag can be set up in the database to record alarm events but to not record state change or operator action events. Event log configuration provides the master switch to turn logging for each category of event on or off. The events are further divided into additional categories on the event log configuration page to provide more control.

Three separate attributes for a tag enable logging of events for the tag: *Alarms-Print Save*, *State Change-Print Save* and *Optr Acts-Print Save*. The *Print* and *Save* fields for each category of event determine if events appear in a continuous printout and a periodic printout of the event log.

To enable recording of tag events:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter **YES** in the *Print* field for each category of event to enable printing that type of event to a printer as an event occurs.

3. Enter **YES** in the *Save* field for each category of event to enable saving that type of event for periodic printing (and archiving).

These fields can be set when a tag is initially defined or at any time to add new events to an already existing event log.

Configuring the Event Log

Up to 1,000 most recent events can be saved for periodic printing and archiving. The events save to the ***EVENTS.LG*** file. The file should ***not*** be directly manipulated using file utilities.

Event log configuration sets the operating parameters that affect the printing and archiving of events. It also determines if a specific category of events (alarm, state change, or operator actions) will be recorded. Settings in the database enable or disable recording of each category of event at the tag level.

For example, if the *Alarms-Save* field in the database is set to *NO* for a tag and the event log configuration *Alarms-Save* field is set to *YES*, an alarm for the tag does not appear in a periodic printout. Conversely, if the *Alarms-Save* field in the database is set to *YES* for a tag, but the event log configuration *Alarms-Save* field is set to *NO*, an alarm for the tag still does not appear in a periodic printout. In the second case, no alarm events are saved for any tags.

On the system event log configuration page, events are classified into two groups: Process events and operator actions. Process events are alarms, digital state changes, and PCU module events. Operator actions include operator control and configuration actions, alarm acknowledges, and operator notes.

Use the *System Event Log* option to enable event logging and to schedule printing. Figure 12-1 shows the page used. To choose this option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

→ *C Logging*

→ *B System Event Log*

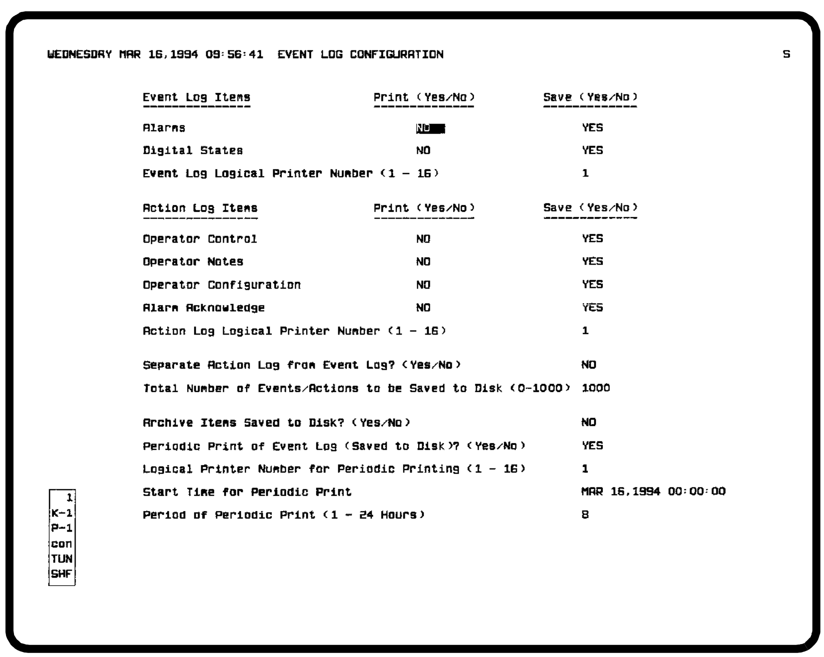


Figure 12-1. System Event Log Page

To configure the system event log:

1. Enter the appropriate data into each field of the page. Refer to Table 12-1 for field explanations.

2. Press **ENTER**.

Table 12-1. System Event Log Page Fields

Field	Description
Items - print	Configures continuous printing of events. Each category of event is enabled or disabled individually. This affects all tags that have corresponding event item *Print* fields set to *YES* in the database. A valid entry is: YES = enable printing events as they occur. NO = disable printing events as they occur.
Items - save	Configures saving of events. An event must be saved for periodic printing and archiving. Each category of event is enabled or disabled individually. This affects all tags that have corresponding event item *Save* fields set to *YES* in the database. A valid entry is: YES = enable saving events. NO = disable saving events.
Event log logical printer number[1]	Logical printer used for printing events. This affects the continuously printing log. A valid entry is from 1 to 16.
Action log logical printer number[1]	Logical printer used for printing operator actions. This affects the continuously printing log. A valid entry is from 1 to 16.
Separate action log from event log?	Used to separate the system events log into two distinct logs for periodic printing: Event log and operator actions log. A valid entry is: YES = enable separate logs. NO = disable separate logs. This maintains a single log containing both process events and operator actions.
Total number of events/actions to be saved to disk	Total number of events to be saved to disk. Events must be saved for periodic printing and archiving. A valid entry is from 0 to 1000. **NOTE:** Any number except zero must be entered in this field to enable the *Save* field entries.
Archive items saved to disk?	YES = enable archiving. Events store to an archival storage directory. NO = disable archiving.
Periodic print of event log (saved to disk)?	YES = enable periodic printing. NO = disable periodic printing. **NOTE:** If *YES*, the *Logical Printer Number for Periodic Printing*, *Start Time for Periodic Print*, and *Period of Periodic Print* fields require an entry.
Logical printer number for periodic printing[1]	Logical printer where a periodic printout will occur. This field relates to the *Periodic Print of Event Log* field and is valid only if that field is *YES*. A valid entry is from 1 to 16.
Start time for periodic print	Time and date at which an initial periodic print should occur. Any subsequent prints occur at a defined period. This field relates to the *Periodic Print of Event Log* field and is valid only if that field is *YES*.
Period of periodic print	Time period in hours between each print after the initial start time. This field relates to the *Periodic Print of Event Log* field and is valid only if that field is *YES*. A valid entry is from 1 to 24.

NOTE:

1. If the continuous event log shares a printer with the periodic event log or other types of logs, each event log entry is inserted in the printout between completed log printouts. A continuous event log can only be directed to a private printer.

CONTINUOUS PRINT

To enable printing events for a tag in a continuous printout:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter **YES** in the *Print* field in the database for each category of event to enable printing that type of event to a printer as an event occurs.

3. Switch continuous printing on by entering **YES** in the *Print* field on the event log configuration page for each category of event to be continuously printed.

PERIODIC PRINT

To print events for a tag in a periodic printout:

1. Follow the steps given in ***Defining a Tag*** in Section 6 to define a tag.

2. Enter **YES** in the *Save* field in the database for each category of event to enable saving that type of event for periodic printing (and archiving).

3. Enable saving events by entering **YES** in the *Save* field on the event log configuration page for each category of event to be saved for periodic printing.

A starting date and time and a print period of up to 24 hours determine when and how often printing occurs. The periodic print function can be turned off at any time. The start time and period for a log can be redefined at any time. Rescheduling does not affect data that has already been archived.

If the period is extended, the printout does not occur until the new period has been completed. If the period is shortened and the current time has already passed the new period, the accumulated events print immediately. The next print time is based on the start time and the new period. If the start time is changed, events print immediately. The log reschedules to the new time and period.

Defining the Event Log Format

Event log format configuration allows changing the format and content of an entry in the event log or operator actions log. Default formats are provided for each type of event. If the default format for a specific type of event meets plant requirements, no further configuration is necessary.

The format and content of each type of event in the system event log can be modified to tailor the log to specific plant requirements. Modifications are limited to defining a title, defining the attributes of an event, and adding any amplifying data (e.g., date, time, tag name, tag description, etc.) that is to appear in a printout.

The event log format options further divide the process events and operator actions into distinct events. Each type of event appears as a separate entry in the event log and can be formatted separately.

Process event:

- Analog alarm events.
- Device status alarm event.
- Digital events.
- Module alarm event.
- Node alarm event.
- Text selector event.

Operator action:

- Configuration event.
- Information event.
- Operator action events.
- Operator notes.

An entry in the event log is either one or two lines with a maximum of 132 characters per line. The information that appears in each line can be arranged in any order desired. Each entry can contain all available attributes or only those attributes needed by plant managers.

Use the *Event Log Format* option to format entries made for each different type of event. Figure 12-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

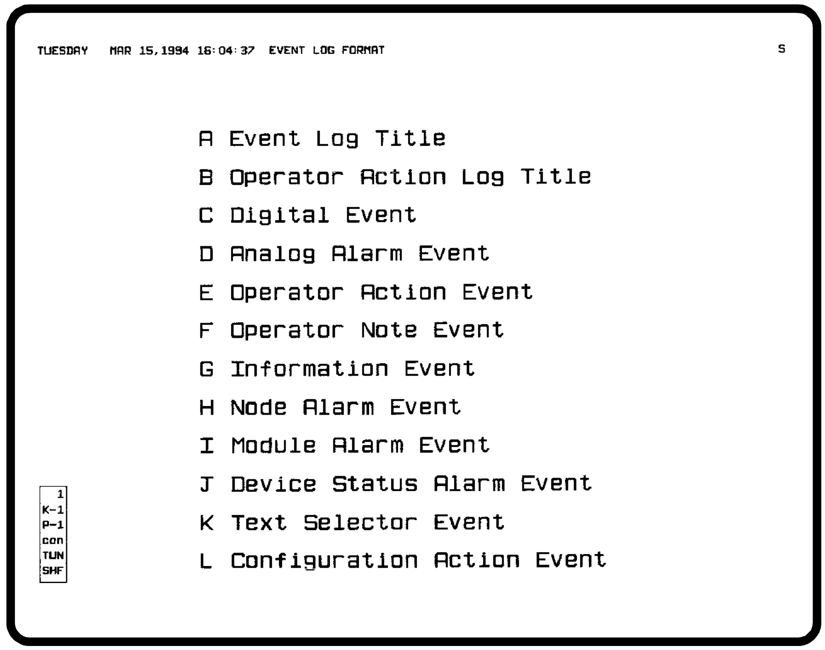
→ *C Logging*

→ *E Event Log Format*

The *Event Log Format* menu has several options which select the different types of events for editing. The formatting procedures are the same for all options.

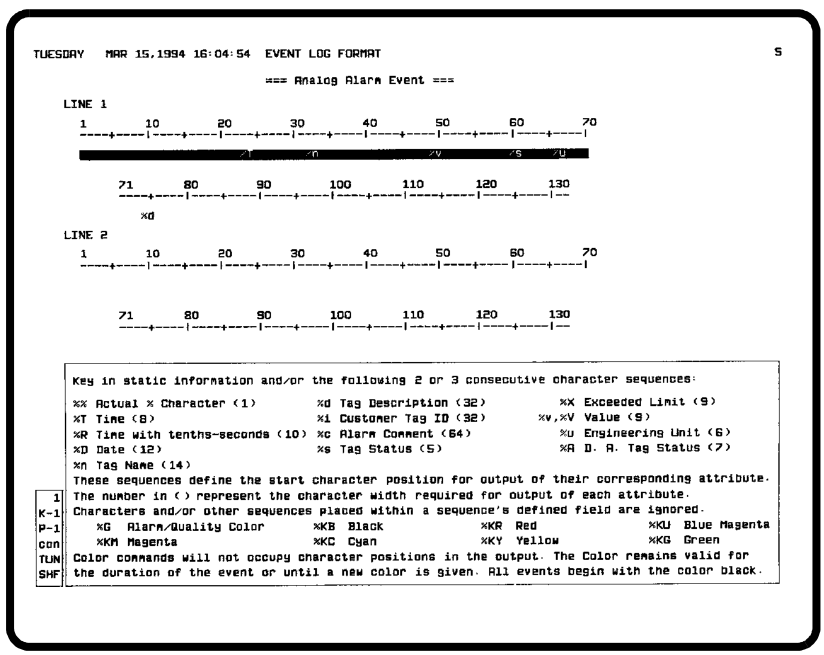
To format an event:

1. Choose an event type from the *Event Log Format* menu. This brings up the formatting page for the event (refer to Figure 12-3 for an example).



TPS0123B

Figure 12-2. Event Log Format Menu Page



TPS0124B

Figure 12-3. Event Log Format Page - Definition Page

At the bottom of this page is a list of two-character sequence codes that define each item that can appear in the log printout. Each type of event has a different set of sequence codes. Color codes can be used if the console is equipped with a color capable printer.

All or only some codes can be used as long as the 132-character maximum is maintained. The number in brackets ( ) next to each sequence code identifies the number of character spaces each type of element occupies. This is the minimum number of spaces to leave between each code. Any characters placed within these reserved spaces are ignored. The codes can be placed anywhere within the highlighted *LINE* fields and in any order.

2. Enter the sequence codes or static data that are to appear for an event. Enter a sequence code exactly as it is shown at the bottom of the page. For example, to show the date enter **%D**.

The placement of a code within the *LINE 1* and *LINE 2* fields directly corresponds to its position in the final printout. Each *LINE* field is separated into two parts; however, in the final printout they become one line.

3. Press **ENTER**.

Page Type Printer Control Codes

Event log printouts can be customized using the control codes in the HP PCL-5 printer control language. These codes are used to change the format of the text to be printed. To enter the codes:

1. Choose an event type from the *Event Log Format* menu. This brings up the formatting page for the event (refer to Figure 12-3 for an example).

2. Enter the printer control code. The following is an example:

   %%<FN.CN>

   This code changes the font to courier. Use the control codes that were set up in the printer ***.DNF*** file.

3. Press **ENTER**.

Event Log Considerations

The same information that appears in a periodic print of the event log and operator actions log appears in an archive of the system event log. Process events and operator action events are not separated for archiving.

If the target printer for a periodic printout or an archive device for archiving of the event log is not functioning, the file buffer may overflow while waiting for the printer or archive device. This can cause a loss of event data from the periodic printout and loss of event data on the archive medium.

If the OIS application is reset during a periodic print period, the accumulated events do not print until the original log period has been reached after the console comes back on-line. If the time period passed while the console was off-line, the accumulated events print immediately after the console comes back on-line. The next periodic print occurs at its originally scheduled time based on the defined log start time and not the restart time for the console.

CUSTOM LOGS

Some standard custom logs are snapshot, trend, and trip. These logs are derived from periodic, standard, and trip log types. A snapshot log takes a sample of current process values at a designated time, on demand, or when a specific process event occurs. A trend log records data collected from the distributed trending system. The trend log can also contain snapshot process values. A trip log retrieves historical, trended data to record pretrip event information and collects current data to record post trip event data. Custom logs are not limited to only these types of logs. These are, however, standard logs for the console.

Log Types

Custom log types are periodic, trip, and standard. The difference between each type is the method used to collect data.

The ***standard*** type is intended to be used for a log that records trend data. It can be used to create trend logs that contain data retrieved from the distributed trending system. The log continuously collects trend data at defined intervals during its entire collection period. This type of log can also contain current, snapshot tag values collected from the process at the end of the log collection period.

The ***trip*** type used for a log allows retrieving data for a specified amount of time prior to a plant fault and collecting data for a specified amount of time after a plant fault. This data consists of historical, trended process values for pretrip data and current process values for post trip data. The amount of pretrip and post trip data depends on log configuration. Only event triggering can be used to start data collection for this type of log.

The ***periodic*** type is intended to be used for a log that records tag data. Process values are collected at a specified time, after an event triggers collection, or on demand. Current, snapshot values are collected at the end of the log collection period. Normally, the collection period is set to zero to have data collection begin as soon as the log is triggered or made active. A periodic log can also retrieve data collected from the distributed trending system. If used this way, however, the periodic log acts similar to and must be configured in the same way as a standard type log.

Scheduling

Data collection and printing for a log can be scheduled to occur at a specified time then at time intervals or to occur after a specific process event. A log can also be defined but unscheduled to allow an operator to demand or start the collection or printing at any time. The time span or length of time that a log collects data can also be defined.

For scheduling purposes, custom logging provides three variations in log operation: Periodic, event triggered, or operator demanded. A periodic log produces an operations summary at a specified time and time intervals. An event triggered log produces a summary after a certain event occurs. It can be used to produce a batch report or prefault and post fault report. Operator demanded logs are unscheduled and initiated through operator actions.

Collection types are time, event, or demand (operator demanded). Print types are time, event, demand, or collect (collection completion). These types are used, in any combination, to schedule logging. Printing takes place at a designated printer.

Collection Trigger Types

A ***time*** collection trigger type is only used for trend logs (standard type) and snapshot logs (periodic type). When using this type of collection, a log has a specific starting time. Data collection for the log begins at the scheduled time.

An ***event*** collection trigger type can be used to start data collection for a log when a certain process event takes place. A trigger tag is monitored and data collection begins after a selected event occurs for the tag. An event can be an analog alarm or a digital state change. Two trigger conditions for data collection can be specified for a log.

A ***demand*** collection trigger type can be used to start data collection for a log as soon as it becomes active. Activating a log can be performed at the time the log is created or later by using the log status function. This type of collection creates an unscheduled log. It allows the operator to start log collection at any time.

Using Repeat Cycle

The repeat cycle option is used to automatically reschedule data collection for a log. It can be enabled when using any of the different collection types. Log printing, with repeat cycle enabled, occurs at the normally scheduled print cycle which is dependent on the print type configured for the log.

With the repeat cycle enabled for a ***time*** collection type log, the log continues to repeat its collection period after it completes its initial collection period. By not setting this option, log data collection stops after completion of the initial collection period.

With the repeat cycle enabled for an ***event*** collection type log, the log begins its collection period each time the event defined to trigger the log takes place. An event that takes place during a collection cycle does not affect or trigger subsequent collections. By not scheduling a repeat cycle, collection occurs only the first time the event occurs.

With the repeat cycle enabled for a ***demand*** collection type log, the log begins its collection period as soon as it is made active and continues to repeat its collection period until it is made inactive. By not scheduling a repeat cycle, the log does not repeat its collection period until it is made inactive then active again.

Print Trigger Types

A log with a ***time*** print trigger type has a specified starting time for printing. If a log continues to repeat its collection period, printing repeats at the defined time.

A log with an ***event*** print trigger type starts after a specified process event occurs, regardless of how much of the collection period is complete. A trigger tag is monitored and log data begins to print after a selected event occurs for the tag. An event can be an analog alarm or a digital state change. This is similar to a demand print type for a log but the demand is based on a process event. Two conditions that trigger printing can be specified for a log.

A log with a ***collect*** print trigger type begins printing immediately after completing its collection period. This type of print does not occur before a log has completed its entire collection period.

A log with a ***demand*** print trigger type is an unscheduled log. The log does not print any of its data until the operator manually demands the printout using the *Log by Name* function. Refer to the discussion on logging in the ***Operation*** instruction for the procedures to demand a printout of a custom log (Table 1-2 lists instruction numbers).

NOTE: A retained copy of a log or any currently collected data for a log can be demanded for printout using the *Log by Name* function.

Saved Logs

The logging system can save up to nine generations of each custom log. A custom log resides in a file named:

LOGnnnn.Ln

where:

nnnn	Log index number from 0001 to 0300.
.Ln	Log retention ranging from ***L0*** to ***L8***. Log file ***L0*** extension is the newest log.

The saved logs are the nine most recently generated (i.e., logs after completed collection). The oldest log is removed when a new log is retained. Any custom log can be saved as a permanent record by using the archiving function.

Defining a Log

A custom log generator function is used to tailor each type of report to specific plant requirements. Log definition is performed to delineate the operating parameters of a log (i.e., type and data collection and printing cycles). Log definition also sets the number of columns and rows in a log, which is important for determining how many values can appear in the log.

After log definition (i.e., NEXT PAGE), an empty log shell containing a number of cells provides the means to format the log. This is where the data that is to appear in a log printout and also the data that appears in a log archive is detailed. The number of cells is equal to the number of rows times the number of columns established during log definition. The format is limited by the number of cells defined and to available cell definition types.

As in a spreadsheet, log cells contain or define the data that is to appear in a printout. A cell of a custom log can contain:

ASCII text characters - up to 80 characters per cell of descriptive text (e.g., headings, legends, descriptions) or printer codes (e.g., color, new line, form feed, compressed print).

Trend value - a trend value based on a selected trend definition.

Tag attribute - tag database configuration data (e.g., tag name, description, engineering units, or alarm groups).

Tag value - a tag value current at the time of data collection based on a selected tag (mainly used for snapshot logs).

Alarm or quality - alarm state or quality status for a tag, not a trend definition.

Calculation result - sum, difference, division, product, average, minimum, or maximum value for selected cells.

Console time and date - either the current time at log output or the log start time.

Offset time - an offset time from the start of the log and previous cells.

Constant value - a constant value which can be used in calculations.

Use the *Report Generator* option to define the operating parameters of a custom log and to format the log. Figure 12-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *C Logging*
→ *A Report Generator*

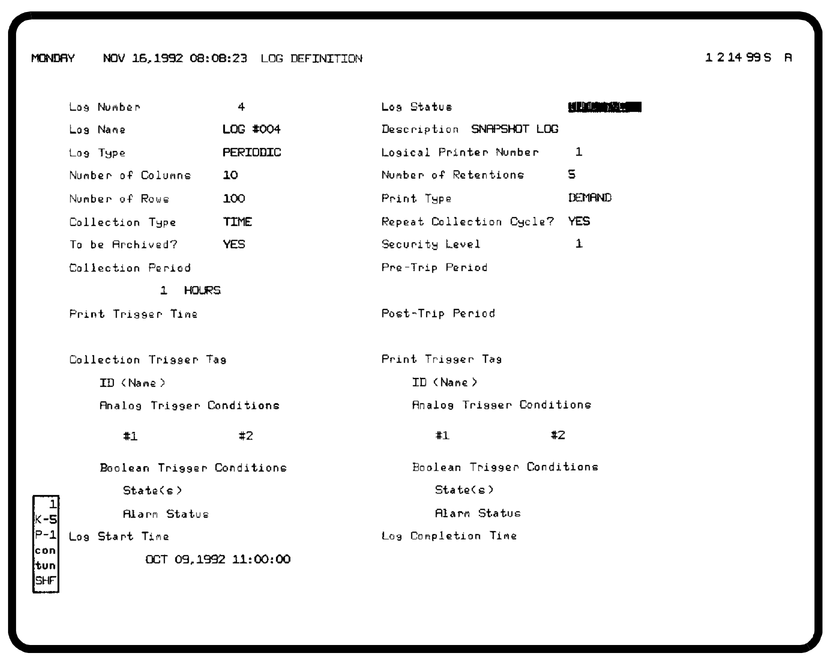


Figure 12-4. Report Generator Page

The fields that require an entry depend on the log type, collection type, and print type chosen. The input cursor can only be moved to required fields. All other fields are left blank.

To define a log:

1. Select the log to define and format.

   a. Enter the index number (or the name if previously defined) of a log in the *Log Name/Index* field. A valid index entry is from 1 to 300.

   b. Press **ENTER**. The definition page for the log appears.

2. Initially, the cursor positions on the *Log Status* field. A log must be inactive to edit. If the log is currently active, type **INACTIVE** and press **ENTER**. Making a log inactive disables log collection and printing for the log. A queued print will still occur, however.

3. Enter the appropriate data into each field of the page. Refer to Table 12-2 for field explanations.

4. Press **ENTER**.

5. Press **NEXT PAGE** to call the cell configuration page. This is the page used to format the log.

6. Define the log cells. This creates the log appearance and content. Refer to ***Formatting Log Cells*** in this section for the procedures to define a cell. The number of rows and columns in a log is set during definition. This determines the number of cells that can be used to present data in a log output.

Table 12-2. Log Definition Page Fields

Field	Description
Log number	Non input field; shows the index number of the log being defined. Up to 300 definitions are supported.
Log status	Shows the current log status and allows changing the status to turn the log on or off. A valid entry is: ACTIVE = activate the log (ON). INACTIVE = deactivate the log (OFF). Must be inactive to edit. The log can be made active as the last step of log definition, or it can be left inactive and turned on later using log status functions. A log begins its collection period as soon as it is made active.
Log name	Eight-character name. This name appears in other functions to identify the log (e.g., log status display).
Description	32-character description. This description appears in other functions.
Log type	A valid entry is PERIODIC, TRIP, or STANDARD.
Logical printer number	Logical printer where a printout is to occur. A valid entry is from 1 to 16.
Number of columns	This field along with the number of rows field determine the number of cells in a log. A valid entry is from 1 to 64.

Table 12-2. Log Definition Page Fields (continued)

Field	Description
Number of retentions	Number of historical copies to be saved. Up to nine most recently generated can be saved. The oldest log is removed when a new log is retained. A valid entry is 0 for no retentions or 1 to 9.
Number of rows	This field along with the number of columns field determine the number of cells in a log. A valid entry is from 1 to 250.
Print type	COLLECT = log prints as soon as it completes its collection period. DEMAND = a printout does not occur until the operator demands it using the *Log by Name* function. EVENT = a process event (i.e., tag alarm condition or state change) triggers printing. Related fields are: *Print Trigger Tag* *Analog Trigger Conditions* *Boolean Trigger Conditions* TIME = a specified time triggers printing. An entry is required in the *Print Trigger Time* field.
Collection type	DEMAND = collection does not start until the operator changes the log status to active. This is an unscheduled log. EVENT = a process event (i.e., tag alarm condition or state change) triggers collection. Related fields are: *Collection Trigger Tag* *Analog Trigger Conditions* *Boolean Trigger Conditions* TIME = a specified time triggers collection. An entry is required in the *Log Start Time* field.
Repeat collection cycle	Configures automatic log rescheduling. YES = enable rescheduling. For a ***time*** triggered log, the log repeats collection at the intervals specified by its collection period. This begins after the initially scheduled collection time. For an ***event*** triggered log, the log collects every time a specified process event occurs. For a ***demand*** triggered log, the log continues to repeat its collection period until it is made inactive. NO = disable rescheduling. The log does not repeat its collection period.
To be archived?	YES = enable archiving. NO = disable archiving.
Security level	Security level to be assigned to the log. The level determines the operations that can be performed on the log. A valid entry is from is 1 to 16.

Table 12-2. Log Definition Page Fields (continued)

Field	Description
Collection period	Amount of time the log continues to collect data after it begins collection. A valid numeric entry is from 0 to 999; the unit is either SECONDS, MINUTES, or HOURS. For a ***trip*** type log, this field should equal the total amount of pretrip and post trip time (i.e., *Pre-Trip Period* field added to the *Post-Trip Period* field). For a ***periodic*** type log, tag data is collected at the end of the collection period, not the log start time. In most cases, this should be zero. For a ***standard*** type log, trend data is collected at intervals during the entire collection period.
Pretrip period	Amount of historical or pretrip data, measured in time, that is to be retrieved after a process trip occurs. This is valid for trip type logs only. A valid numeric entry is from 0 to 999; the unit is either SECONDS, MINUTES, or HOURS.
Print trigger time	Starting time for printing when using a time print type.
Post trip period	Amount of data that is to be collected after a process trip occurs. This is valid for trip type logs only. A valid numeric entry is from 0 to 999; the unit is either SECONDS, MINUTES, or HOURS.
Collection trigger tag name	Name or index number of a tag that is to be monitored to trigger data collection. This is valid for event collection type logs only and is independent of the print trigger tag. Related fields are: *Analog Trigger Conditions* *Boolean Trigger Conditions*
Print trigger tag name	Name or index number of a tag that is to be monitored to trigger printing. This is valid for event print type logs only and is independent of the collection trigger tag. Related fields are: *Analog Trigger Conditions* *Boolean Trigger Conditions*
Analog trigger conditions (collection or print trigger tags)	One or two (i.e., *#1* and *#2*) alarm conditions of the analog trigger tag that are to start data collection or printing. A valid entry depends on the type of tag identified as the analog trigger tag: N = trigger on return-to-normal condition. (blank) = no trigger. Leave the fields blank to not use alarm as the trigger. 3H = trigger on 3H alarm condition. 2H = trigger on 2H or 3H alarm condition. H = trigger on H, 2H, or 3H alarm condition. L = trigger on L, 2L, or 3L alarm condition. 2L = trigger on 2L or 3L alarm condition. 3L = trigger on 3L alarm condition. **NOTE:** HF, HD, LD, HR, or LR cannot be used as a trigger condition.

Table 12-2. Log Definition Page Fields (continued)

Field	Description	
Boolean trigger conditions (collection or print trigger tags)	State	Digital state of the boolean trigger tag that is to start data collection or printing. A valid entry depends on the type of tag identified as the boolean trigger tag and is one of the logic state descriptors defined in the database for the tag. Leave the field blank to not use digital state as the trigger.
	Alarm status	Alarm status condition of the boolean trigger tag that is to start data collection or printing. Alarm status triggering can be based on a transition from normal to alarm or alarm to normal. One of the digital states of a tag is designated as an alarm state in the PCU module. A valid entry is: ALARM = normal to alarm status trigger. RTN = alarm to normal status trigger.
Log start time	Starting time for data collection when using a time collection type. For a ***periodic*** type log, data collection begins at the start time as long as the collection period is zero. If not, data collection begins at the start time plus the collection period. For ***trip*** and ***demand*** type logs, this field is not valid.	
Log completion time	Non input field; the completion time is automatically calculated.	

Formatting Log Cells

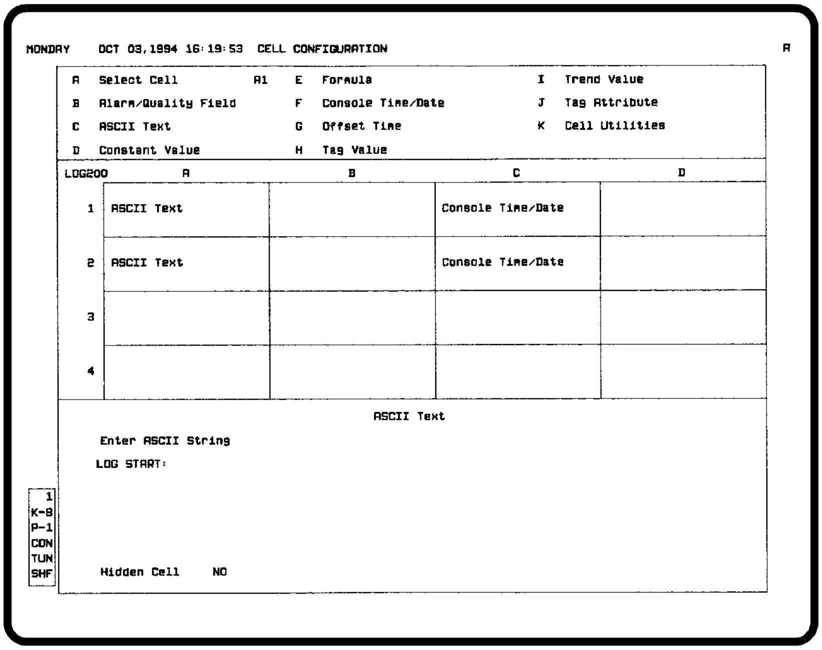
The cell configuration page is used to create the final format and content of a custom log (Figure 12-5). Press **NEXT PAGE** while on the log definition page to call this page. Each cell on the screen occupies the same relative position in the final printout.

At the top of the page is the cell configuration main menu. This menu lists all possible cell definitions and provides access to utilities for:

- Copying cells.
- Copying rows.
- Copying columns.
- Deleting cells.
- Setting column width.
- Printing a sample of the log format.

Refer to ***Using Cell Utilities*** in this section for an explanation of the utilities and for the procedures to use them.

The middle of the page shows a group of 16 cells. To the upper left of this block of cells is the log name. The letters across the top of the cells identify each column of cells. Columns start at A and end at a maximum of BL (column 64). The numbers down the left side identify each row of cells. The number of rows is from 1 to 250. Use the log definition page to define the number of rows and columns for a log.



TPS0754B

Figure 12-5. Cell Configuration Page

If a cell has already been defined, a name indicating the cell type appears in that cell (Figure 12-5). If not, the cell is blank. Use the *A Select Cell* option to select a cell within the log area and to reposition the page. A cell selected through this option becomes the upper left cell in the block of 16 cells.

The attributes related to a cell are shown at the bottom of the page. After selecting a cell, the attributes that define the particular type of cell appear in this area and can be changed.

> **NOTE:** ASCII text, console time and date, offset time, and tag attribute type cells can cross column boundaries for ease of definition. Refer to ***Using Cell Utilities*** in this section for procedures to set cell widths.

To define a cell:

1. Select a cell.

   a. Choose *A Select Cell.*

   b. In the input field, enter the address of a cell to define or edit. Type the column address first then the row address. Columns use alphabetic characters from A to BL (64) and rows number from 1 to 250. For example:

   AC40

   c. Press **ENTER**.

2. Select one of the cell definition options from the menu at the top of the page. For example, press **F** if the cell is to contain the console time and date. After being selected, the attributes that define the cell appear at the bottom of the page.

3. Define each field for the selected type of cell. Refer to the sections that follow for an explanation of each cell type.

4. Press **ENTER** to accept the cell definition.

5. Press **ESC**. The name of the selected type should appear in the cell after being successfully defined.

Use the *Print Log Format* option accessed through *K Cell Utilities* to print a proof of the created log. Artificial values are used wherever a process value should appear. Refer to ***Using Cell Utilities*** in this section for an explanation of the option.

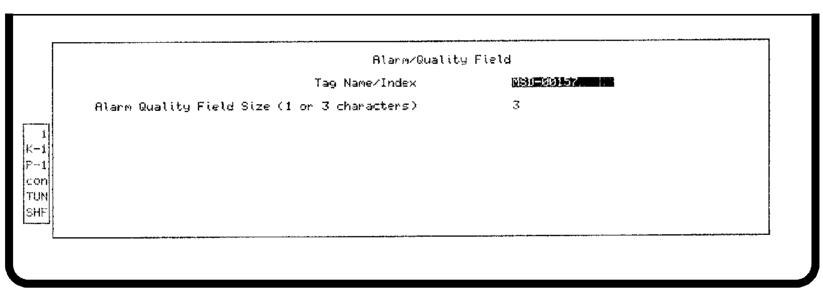
HIDDEN CELL

A cell designated as a hidden cell does not appear in the final printout. The result of the hidden cell is available to the logging function when performing calculations, however. For example, a hidden cell could be used for data or formulas needed to produce an intermediate value but its data or value is not necessarily needed in the final log output.

To designate a cell as hidden, enter **YES** in the *Hidden Cell* field. To remove a hidden cell designation enter **NO** in the field.

ALARM/QUALITY

Use the *B Alarm/Quality Field* option (Figure 12-5) to create a cell that gives the alarm and quality status for a tag. This type of cell can only be used to show the status associated with a tag value (snapshot value), not a trended value. Trended values do not carry status. Figure 12-6 shows the fields that must be defined for this cell type.



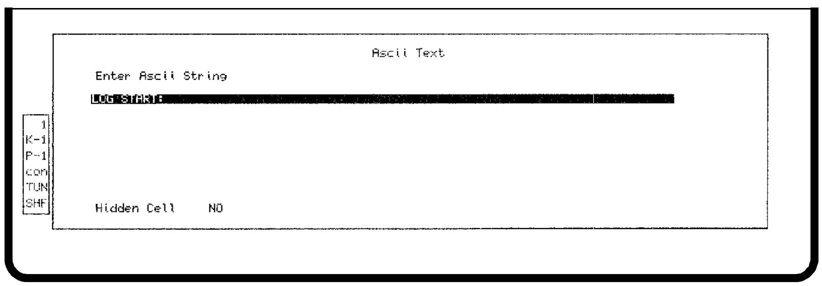
TPS0755A

Figure 12-6. Alarm/Quality Field Option

A status field width can be selected in the *Alarm Quality Field Size* field: Either one or three characters. A three-character field shows two characters for the alarm status and one character for the quality. It is the same as a five-character alarm status/quality/group field that appears on a display without the two characters for the alarm group. A one-character field will have an abbreviated alarm status overridden by bad quality similar to a one-character alarm status field on a display. Refer to the discussion on alarm management in the ***Operation*** instruction for a description of alarm and quality indications (Table 1-2 lists instruction numbers).

ASCII TEXT

Use the *C ASCII Text* option (Figure 12-5) to create a cell that contains a static text string. This string can be up to 80 characters long. The maximum number of characters is 5,120 per log. Figure 12-7 shows the input field for this cell type.



TPS0756A

Figure 12-7. ASCII Text Option

Additionally, printer control can be performed using this type of cell. The percent (%) character followed by a one or two-character code defines a printer command. All non valid command codes print as static text. Table 12-3 lists the printer control codes.

Table 12-3. Printer Control Codes

Code	Function
%N	New line (carriage return and line feed)
%F	Form feed (top of form and carriage return)
%C0	Turn compressed print off
%C1	Turn compressed print on (allows maximum of 200 characters per line)
%<XX.XX>	All control codes available in the HP PCL-5 printer control language. Refer to the printer documentation for available codes.

The code **%C1** puts the printer into compressed mode. The printer does not automatically reset to normal mode (132 characters per line) when another log prints. A **%C0** command must be defined to return to normal, uncompressed print. Place this command in the last cell.

The page mode printer commands require subitems separated by a period (.). Also multiple codes can be entered by separating the commands by a space. The following is an example of a page mode printer command:

%<OR.L FN.LG TR.L PSZ.B>

This command sets the printer to landscape, letter gothic, lower tray and B size paper. For more information on page mode printers refer to the ***File Utilities*** instruction.

> **NOTE:** Key in **%%** to print the percent character (%); the escape sequence character is ignored.

A color can be specified for a log using an ASCII cell definition. For example, the color code **%K2** sets the color output to red. This code does not print in the final output. Table 12-4 lists the standard color codes used.

Table 12-4. Printer Color Codes

Code	Color
%K0	Black
%K2	Red
%K3	Green
%K5	Cyan
%K6	Magenta
%K7	Yellow
%K12	Blue-magenta

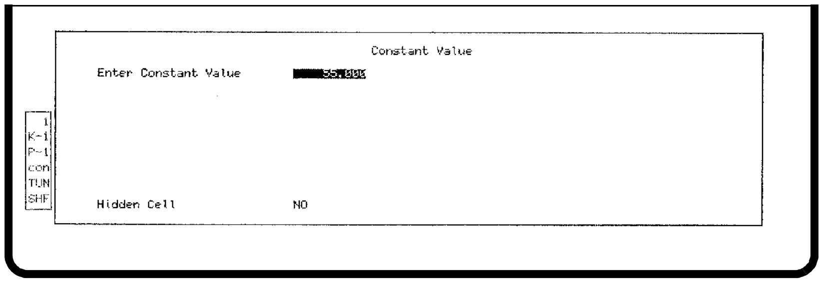
The table lists the codes that support the current color printer capabilities. Colors can be remapped through printer color map configuration. If remapped, the number portion of the code can range from zero to 63 as in screen color codes.

The number denoting the color is terminated by any non numeric character. If the character is %, it is not in the output. The % character acts as the terminating character for the color and allows a numeric character to be output immediately following the color code. For example, to output the number three following a color code, enter:

%K5%3

CONSTANT VALUE

Use the *D Constant Value* option (Figure 12-5) to create a cell that contains some constant value. The constant value can be retrieved by a formula type cell. If the value is not to be used with a formula cell, define the constant as an ASCII text cell instead. Figure 12-8 shows the input field for this cell type.



TPS0757A

Figure 12-8. Constant Value Option

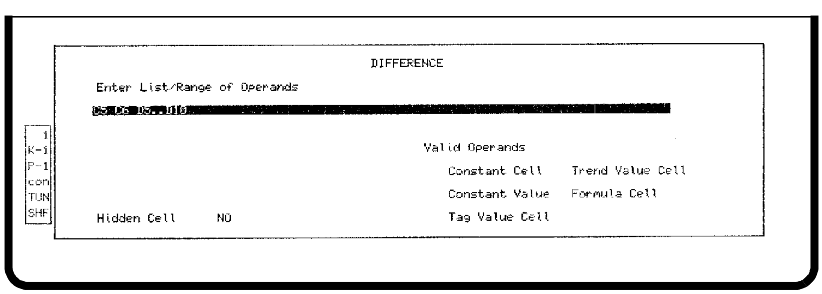
FORMULA

Use the *E Formula* option (Figure 12-5) to create a cell that is a function of other cells. Specifically, a formula cell can be defined to perform calculations using operands that are the result of other cells. After choosing the option, the following menu options appear:

B SUM
C DIFFERENCE
D PRODUCT
E DIVISION
F AVERAGE
G MINIMUM
H MAXIMUM

NOTE: The average is calculated as the sum divided by the number of operands.

All formulas can be defined in the same way. After selecting one of the formula options, an *Enter List/Range of Operands* field appears (Figure 12-9). Use the field to specify the cells that are to be part of the equation. The field allows for up to 80 characters.



TPS0758A

Figure 12-9. Formula Option

The cell types that can be used as operands in a formula include:

- Constant value cell.
- Tag cell (analog value only).
- Trend cell (analog value only).
- Formula cell.

A column address and row address reference a cell. Columns use alphabetic characters from A to BL (64) and rows number from 1 to 250. For example:

BD12

In formula cell definition, enter the row address as **0** to specify all valid cell types in an entire column. Enter **ZZ** as the column address to specify all valid cell types in an entire row. Specify a list of operands by entering individual cell addresses each separated by a comma or a blank space. Enter a range of operands by specifying a starting cell then ending cell separated by an ellipsis of two periods (..). It is possible to specify a range of cells within a list of individual cells.

Example: **A16,BZ1,J70,M1** or **A16 BZ1 J70 M1** (individual cells)
A1 . . AZ1 (range of cells from A1 to AZ1)
A1 . . L1 C7 L22 (range from A1 to L1 and C7 and L22)

If any one of the cells involved in the calculation is bad quality, it is omitted from the calculation. The result of the formula cell is marked with the suspect indicator (?). This indicator overwrites the right-most character (least significant digit position) of the formula cell result.

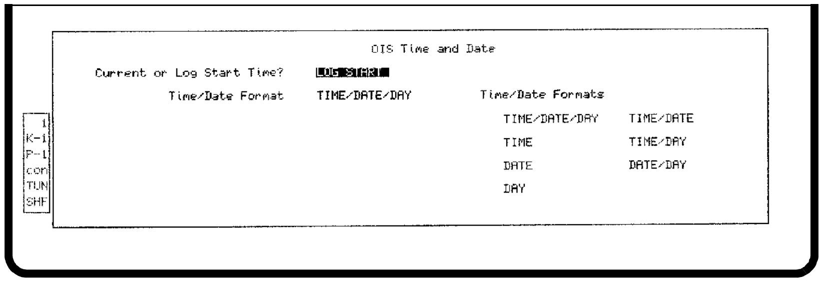
Example: Cell C3 contains an averaged trend value. Cell C4 contains another averaged trend value. Cell D4 is defined as a formula cell to calculate the difference between C4 and C3. To to this, select *C DIFFERENCE* and enter the cell address of the two operands as:

C4,C3

This defines the formula cell as the calculation C4 - C3. The calculation result appears in a printout.

CONSOLE TIME AND DATE

Use the *F Console Time and Date* option (Figure 12-5) to create a cell that contains a time and date. The time and date can be either the time of the log output or the time the log started (start collection time). Figure 12-10 shows the fields that must be defined for this cell type.



TPS0759A

Figure 12-10. Console Time and Date Option

The *Current or Log Start Time?* field determines whether the time is the current time at log completion or the time the log started. Enter **CURRENT** or **LOG START**.

The *Time/Date Format* field determines the content of the printed time and date. The options for this field are:

TIME	16:32:00
DATE	28-FEB-96
DAY	Tuesday
TIME/DATE	16:32:00 28-FEB-96
TIME/DAY	16:32:00 Tuesday
DATE/DAY	28-FEB-96 Tuesday
TIME/DATE/DAY	16:32:00 28-FEB-96 Tuesday

Time and date format configuration sets the actual order and format for the time (e.g., 16:32:00 or 16:32) and for the date (e.g., 28-FEB-96 or February 28, 1996). Refer to ***TIME AND DATE FORMAT*** in Section 3 for the procedures to define the format.

The following describes the *LOG START* time for the different log types:

Standard (trend) - the time that the log starts collecting. Normally, the first trend cell is offset zero from the log start time. In this case, the log start time is also the time of the first trend cell.

Trip - the time of the event.

Periodic (snapshot) - the time the snapshot was taken.

The following describes the *CURRENT* time for the different log types:

Standard - the time that log collection completed. This does not include the processing time required to format the log for printing.

Trip - the time that log collection completed (i.e., the time of the latest post trip data that has been collected). This time does not include the processing time required to format the log for printing.

Periodic - the time that log collection completed. This does not include the processing time required to format the log for printing.

OFFSET TIME

Use the *G Offset Time* option (Figure 12-5) to define a series of horizontal or vertical cells to show time offsets. The first time is an offset from the log start time, and the subsequent times are offset from each other. An offset is defined as a count and a unit such as a count of 45 and a unit of seconds. An offset of 45 seconds, for example, causes a cell to show a time 45 seconds offset from the previous cell. Figure 12-11 shows the fields that must be defined for this cell type.

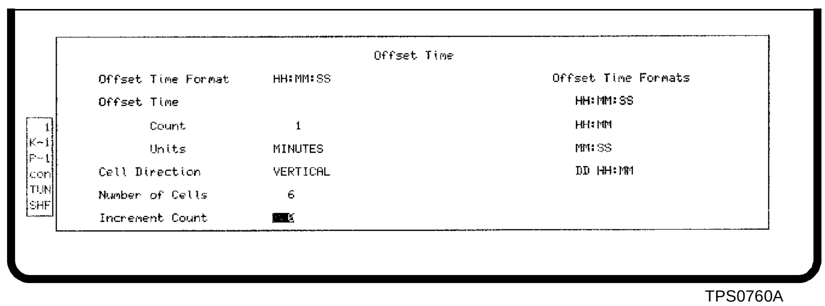


Figure 12-11. Offset Time Option

The options for the time format are:

HH:MM:SS	12:30:00
HH:MM	12:30
DD HH:MM	20 12:30
MM:SS	30:00

The *Cell Direction* field determines whether the cells are to be defined consecutively across a given row or consecutively down a given column. Enter either **VERTICAL** or **HORIZONTAL**.

The *Number of Cells* field defines the number of consecutive cells that will show a time. This must not exceed the maximum number of cells in a row or a column.

The *Offset Time* fields set the initial offset time for the first cell. This time is relative to the log start time. Specify the time as a count from zero to 999 and a unit of SECONDS, MINUTES, or HOURS.

The *Increment Count* field sets the time difference between consecutive cells after the first cell. This increases the count for the next offset time by the specified number for each consecutive cell. The unit associated with this count is the unit defined for the *Offset Time* field.

TAG VALUE

Use the *H Tag Value* option (Figure 12-5) to create a cell that shows the current value or state of a tag. The value is a snapshot taken at completion of the collection period. The following tag types require a *Tag Value Type* specification to determine the actual value or condition the cell is to contain:

DADIG	DEVSTAT	STATION
DANG	MSDD	TEXTSTR
DD	RMCB	

For example, PV for process variable, CO for control output, MD for mode, ST for status, etc., must be selected to identify the specific tag value to log. When the tag name or index number is entered, a list of subtypes appears. Figure 12-12 shows the fields that must be defined for this cell type.

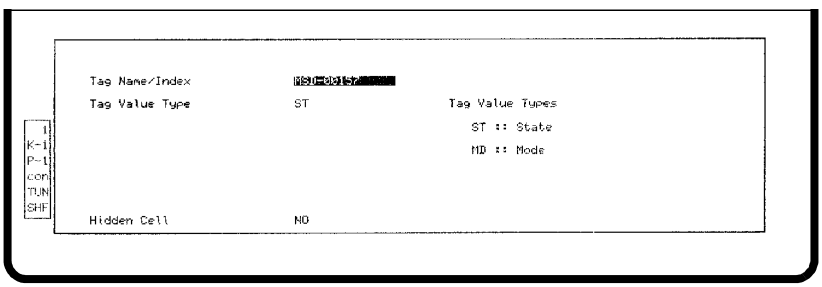


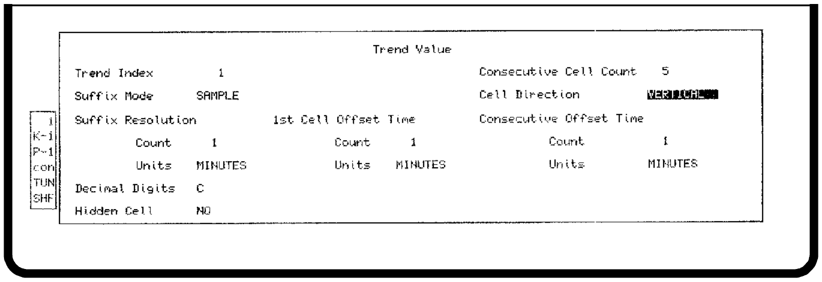
Figure 12-12. Tag Value Option

TREND VALUE

Use the *I Trend Value* option (Figure 12-5) to define a series of horizontal or vertical cells to show trend values. The option can also be defined to create a single trend value cell. The cell must reference a valid trend definition.

A trend block performs calculations on samples it collects to derive the final value that a PCU module sends to the console. The trend value cell allows performing a second calculation on the values it receives from the trend block. Cell attributes determine the type of calculation.

Figure 12-13 shows the fields that must be defined for this cell type. Refer to Table 12-5 for an explanation of the fields. Press **ENTER** after defining the fields. The cell definition is copied into the number of consecutive cells specified.



TPS0762A

Figure 12-13. Trend Value Option

Table 12-5. Trend Value Cell Fields

<table>
<tr><th>Field</th><th>Description</th></tr>
<tr><td>Trend index</td><td>Index number of the trend definition that provides the values for the cells. A valid entry is from 1 to 10000.</td></tr>
<tr><td>Suffix mode</td><td>Type of calculation to perform on trend data. The Suffix Resolution field determines the amount of time samples are to be taken for use in the calculation. A valid entry is:

AVERAGE = sum of all samples divided by the number of samples. Not valid for a digital trend.

MINIMUM = minimum value during the period.
MAXIMUM = maximum value during the period.
SAMPLE = default; presents a single value and no calculation is performed.
SUM = sum of all values during the period. Not valid for a digital trend.

NOTE: This is a second calculation if the module uses any collection mode other than SAMPLE.</td></tr>
<tr><td>Suffix resolution</td><td>Period of time over which samples are to be collected for a suffix calculation. This should be set, at a minimum, to the trend collection resolution. It can also be set to any multiple of the collection resolution. Specify the resolution as a count from 0 to 999 and a unit of SECONDS, MINUTES, or HOURS. For example, if the trend is a one-minute trend, the resolution can be set to two minutes, three minutes, four minutes, etc.</td></tr>
<tr><td>Decimal digits</td><td>Number of decimal places to the right of the decimal point. A valid entry is from 0 to 6 or C. Use C to choose precision based on column width:
<table><tr><th>Width</th><th>Digits</th></tr><tr><td>1</td><td>0</td></tr><tr><td>2</td><td>0</td></tr><tr><td>3</td><td>0</td></tr><tr><td>4</td><td>1</td></tr><tr><td>5</td><td>2</td></tr><tr><td>6</td><td>2</td></tr><tr><td>7</td><td>2</td></tr><tr><td>8</td><td>3</td></tr></table></td></tr>
<tr><td>Consecutive cell count</td><td>Number of consecutive cells that will be defined as trend cells. This must not exceed the maximum number of cells in a row or column.</td></tr>
<tr><td>Cell direction</td><td>Direction of consecutive cells. A valid entry is VERTICAL or HORIZONTAL.</td></tr>
</table>

Table 12-5. Trend Value Cell Fields (continued)

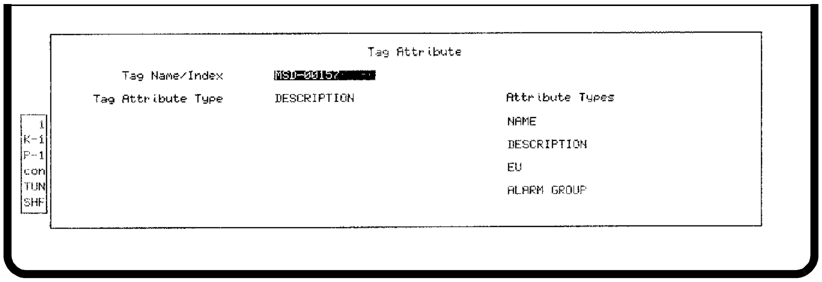
Field	Description
1st cell offset time	Time relative to the log start time for the first trend cell. Specify as a count from 0 to 999 and a unit of SECONDS, MINUTES, or HOURS.
Consecutive offset time	Time difference between consecutive trend cells after the first cell. Each consecutive cell will be offset from the previous cell by this much time. Specify as a count from 0 to 999 and units of SECONDS, MINUTES, or HOURS. **NOTE:** The *Units* must match that of the *1st Cell Offset Time*.

Data values in the log are listed in ascending order from either the starting time or the earliest log value. Therefore, it is not possible for the data in a log to be printed in reverse chronological order.

Problems may occur if the collection period defined for the log exceeds the time span defined for the trend. In this case, trend data would be overwritten before a calculation could be completed. The value is marked as suspect (?) if this occurs. If the number of samples collected is less than expected, a calculation is performed but again the result is marked as suspect. If the trend is not defined or is no longer defined in the database, the value is marked as bad quality.

TAG ATTRIBUTE

Use the *J Tag Attribute* option (Figure 12-5) to create a cell that contains tag configuration data. The tag name or index number must be specified along with the type of data requested. The following configuration items for the tag can be selected for the cell: Name, description, engineering unit, or alarm group. Figure 12-14 shows the fields that must be defined for this cell type.

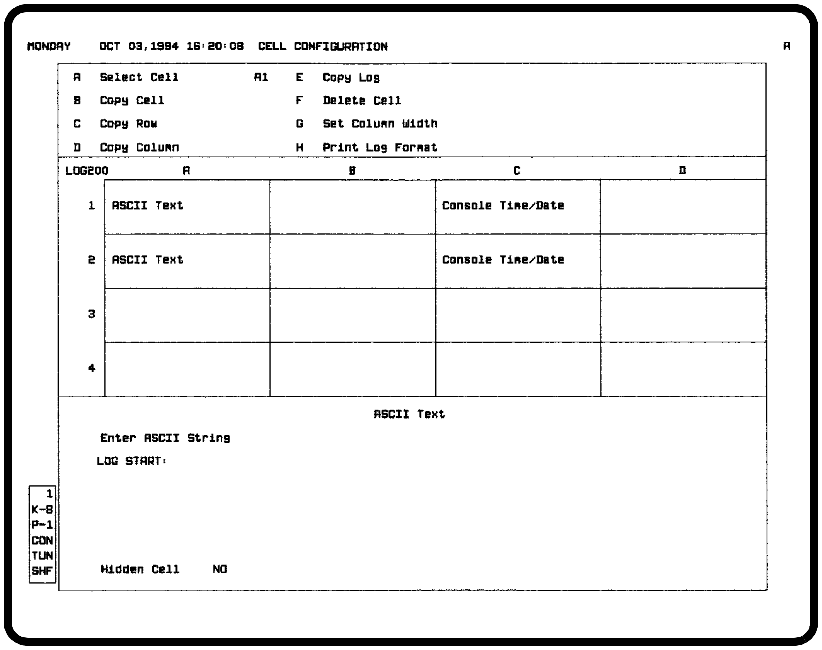


TPS0763A

Figure 12-14. Tag Attribute Option

Using Cell Utilities

The *K Cell Utilities* option (Figure 12-5) gives access to utilities that can be used to expedite cell definition, delete cells, copy an entire log, set column widths, and initiate a sample log printout for proofing. Figure 12-15 shows the options available after choosing this option. Refer to the following sections for an explanation of the options.



TPS0764B

Figure 12-15. Cell Utilities Options

COPY CELL

Use the *B Copy Cell* option (Figure 12-15) to copy a cell definition to another cell location. To use the option:

1. Press B to select the option.

2. Enter the column and row address of the source cell in the *Copy From:* field. The source initially defaults to the currently selected cell.

3. Enter the address of the destination cell in the *To:* field.

4. Press ENTER to initiate the copy.

COPY ROW

Use the *C Copy Row* option (Figure 12-15) to copy an entire row of cell definitions to another row. The destination row and source row initially default to the currently selected row. To use the option:

1. Press C to select the option.

2. Enter a source row in the *Copy From:* field.

3. Enter a destination row in the *To:* field.

4. Press **ENTER** to initiate the copy. The function copies the ***entire*** source row of cell definitions to the destination row.

COPY COLUMN

Use the *D Copy Column* option (Figure 12-15) to copy an entire column of cell definitions to another column. The source column and destination column initially default to the currently selected column. To use the option:

1. Press **D** to select the option.

2. Enter a source column in the *Copy From:* field.

3. Enter a destination column in the *To:* field.

4. Press **ENTER** to initiate the copy. The function copies the ***entire*** source column of cell definitions to the destination column.

COPY LOG

Use the *E Copy Log* option (Figure 12-15) to copy an entire log definition including cell definition to another log definition file. The source log is always the log currently being defined or reviewed. To use the option:

1. Press **E** to select the option.

2. Enter the index number of a log in the *To:* field. This is the destination of the copy.

3. Press **ENTER** to initiate the copy.

DELETE CELL

Use the *F Delete Cell* option (Figure 12-15) to delete one or more cells. To use the option:

1. Press **F** to select the option.

2. Enter the column and row address of the first cell to delete.

3. Enter the address of the last cell to delete. If deleting a single cell definition, either leave the field blank or enter the same cell that was entered as the first cell to delete.

4. Press **ENTER** to initiate the deletion. This makes the cell or cells *Undefined*. Press **ESC** before pressing **ENTER** to return to the cell definition menu without deleting any cells.

SET COLUMN WIDTH

Use the *G Set Column Width* option (Figure 12-15) to define the width for each column of the log. Columns default to a width of six printable characters. A column width is used to determine the starting point for each column and for justifying values within a column. ASCII text, console time and date, offset time, and tag attribute type cells can cross column boundaries for ease of definition.

To use this option:

1. Press **G** to select the option.

2. Enter the address of the column to adjust.

3. Enter a width from 1 to 99. Before making this adjustment, however, consider the appearance of the entire log and also the number of characters a printer can output on a single line.

4. Press **ENTER** to initiate the change.

PRINT LOG FORMAT

Use the *H Print Log Format* option (Figure 12-15) to print a sample of the log currently being defined for proofing. Figure 12-16 is an example of a printout generated with this option. On the printout, the rows and columns are numbered with the starting point of a column marked by a bar (¦) symbol. Actual tag attributes and ASCII text will be shown, but times, values, alarm status, and qualities will be artificial values or conditions in order to provide a more realistic presentation.

To use the option:

1. Press **H** to select the option.

2. Specify a logical printer number from 1 to 16. This directs the printout to a specific printer.

3. Press **ENTER** to start the print.

```
 :1        :   :3        : :6        : :7        : :9        : :11       : :13       : :15       : :17       : :19       : :21       : :23
 001



                                                    TRIP LOG


 TRIP LOG FOR  11:28:27 31-JAN-91 THURSDAY                                          DATE PRINTED 11:28:27 31-JAN-91 THURSDAY


          1.  3-FT-154   FDWTR FLOW                   7.   3-PT-165   DRUM LEVEL
          2.  3-FT-177   SH ATTEMP SPRAY FLOW         8.   3-PT-020   MAIN STEAM PRESS
          3.  3-FT-179   RH ATTEMP SPRAY FLOW         9.   3-PT-193   BFP DISCH HEADER PRESS
          4.  3-FT-162   BFP 3A MOTOR AMPS
          5.  3-FT-164   BFP 3B MOTOR AMPS
          6.  3-FT-165   DRUM LEVEL


                                                      PRE-TRIP

                  1          2          3          4          5          6          7          8          9

  TIME

 00:00:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:00:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:00:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:00:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:01:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:01:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:01:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:01:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:02:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:02:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:02:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:02:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:03:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:03:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:03:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:03:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:04:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:04:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:04:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:04:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568

                                                     POST-TRIP

 00:05:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:05:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:05:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:05:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:06:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:06:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:06:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:06:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:07:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:07:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:07:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:07:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:08:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:08:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:08:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:08:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:09:00      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:09:15      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:09:30      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
 00:09:45      1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568   1234.568
```

TPS0142A

Figure 12-16. Example Trip Log Printout

Activating a Custom Log

To start data collection for a defined and formatted log, make it active. A log can be activated as the final step when it is created, or it can be left inactive and activated later using log status functions.

To activate a log:

1. Call the log definition page for the log to activate. Press **ESC** while on the cell configuration page to return to the log definition page.

2. In the *Log Status* field, enter **ACTIVE**.

3. Press **ENTER**.

When a log is activated, the validity of the log header configuration is checked. If any field is configured incorrectly, an error message displays and the cursor positions on the field in error.

Log Errors

A check for invalid tag and trend references is made when a log is activated. An invalid reference error is shown in the diagnostic log function and on the operator information events page. Refer to the discussion on operator information events in the ***Operation*** instruction for further explanation (Table 1-2 lists instruction numbers). Also, at the end of each log printout, an error report is included to notify of any invalid tag index, trend index, or formula reference.

The *Log Last Good Value* field on the alarm quality options page determines how bad quality or disestablished points in a log are handled. The field selects to either log a bad quality string or the last good value for the point. Enter **YES** in the field to have the last good value appear in a log. Enter **NO** to have a bad quality string appear. Refer to ***ALARM QUALITY OPTION*** in Section 9 for the procedures to set the field.

Scheduling Examples

Trend Log

A trend log (standard type) with a *TIME* collection type and *COLLECT* print type has a start time of 12:00 p.m. (12:00), its collection period is 12 hours and its defined columns and rows are 20 and 50 respectively. A trend value cell in the log is set for 30-minute offset time, 30-minute resolution and 24 consecutive cells (vertical cells).

The first value will not be collected until 12:30 p.m. (12:30), then every 30 minutes until the collection period is complete. The collected values occupy 24 of 50 rows. The log prints at 12:00 a.m. (24:00). The example assumes the remaining rows and columns contain data defined with additional cell attributes.

Trip Log

A trip log with an *EVENT* collection type and *DEMAND* print type has a one-hour pretrip period and one-hour post trip period. Its collection trigger is a high alarm (H) for an analog tag, and its alarm status is *ALARM*. The defined columns and rows for the log are 20 and 100 respectively. A trend value cell in the log is set for zero-minute offset time, two-minute resolution and 62 consecutive cells (vertical cells). The specified trigger tag reaches a high alarm point at 1:00 P.M. (13:00) which starts data collection for the trip log.

The first value will be the trend value at 12:00 P.M. (12:00). The additional values will be collected at two-minute intervals until the post trip period is complete at 2:00 P.M. (14:00). The collected values will occupy 61 of 100 rows. The log prints when demanded using log retrieval procedures. The example assumes the remaining rows and columns contain other data defined with additional cell attributes.

Snapshot Log

A snapshot log (periodic type) with a *DEMAND* collection type and *COLLECT* print type has a zero collection period. The defined columns and rows for the log are 50 and 20 respectively. It contains 20 individual tag value cells listed horizontally. The log is left inactive during configuration.

At 3:00 p.m. (15:00), the operator enters the log status function and changes the log status to active. Data collection begins immediately and the log prints as soon as all data has been collected and the log has been formatted. The printout contains the 20 specified values as they were at 3:00 p.m. (15:00). The values occupy all 20 columns and one of ten rows. The example assumes the remaining rows and columns contain other data defined with additional cell attributes.

Deleting a Log Definition

A log definition is deleted by removing a log definition file from the hard disk.in two ways: OpenVMS **DELETE** command from a terminal window or **DEL** command of the DDT function. To delete a log definition by using the DDT function:

1. After starting the DDT function, change to the directory containing the log definition file. Type:

   FD00 *xx* **Return**

where:

xx	2C for logs 1 to 70.
	2D for logs 71 to 140.
	2E for logs 141 to 210.
	2F for logs 211 to 280.
	30 for logs 281 to 300.

2. Type:

   DEL LOG*nnnn***.LF** **Return**

where:

nnnn	Log number.

Refer to the ***File Utilities*** instruction for the procedures to start the DDT function and a description of the **DEL** command.

SOE LOGS

The data for sequence of events (SOE) logging can come from various sources. Sequence of events logging provides a means to retrieve, store, and print data collected by a sequential events recorder (SER), INSEM01 Sequence of Events Module, or SOE (FC 210) function block.

To use an ***SER recorder***, either a multi-function processor (MFP) module or multi-function controller (MFC) module and associated I/O modules are required in addition to the recorder. In this case, the MFP or MFC module provides the interface between the recorder and the console. The module must be dedicated to events logging.

To use the ***SOE (FC 210) function block***, an MFP or MFC module and associated I/O modules are required. In this case, an MFP or MFC module reports the SOE data to the console. The module must be dedicated to events logging.

To use a ***SEM module*** requires setting up a sequence of events system. This system consists of an INSEM01 module and one or more subsystems containing a multi-function processor module, IMSET01 module, IMSED01 module, and associated I/O modules. In this case, the SEM module reports the SOE data to the console. The MFP module provides the interface between the SEM module and any SOE subsystems. Refer to the ***INFI 90 Distributed Sequence of Events*** instruction for an explanation of these modules (Table 1-2 lists instruction numbers).

All data collection and storage is performed at the module level. The responsibility for collecting event data and notifying the console of data to send resides with the module. After receiving a data present notification in an exception report generated by an MFP, MFC, or SEM module, the console:

- Polls and collects the data from the module.
- Formats the data into a log for printing.
- Directs the log to one of its printers.

The ability to manually request (trigger) data collection for a summary type log is provided.

Sequential Events Recorder

When data for an SOE log comes from a sequential events recorder, an MFP or MFC module first collects the data from the recorder and stores it. The module then notifies the console that it has data to send. The console then polls the MFP or MFC module for the data. This notify/request/reply sequence occurs for all log types except a summary log. The operator must initiate data collection for a summary log by changing the state of an RCM tag.

Before configuring an SOE log on the console, configure both the MFP or MFC module and the recorder. An SOE log (FC 99) function block allows the module to collect data from a recorder. This function code defines the type of report and maximum age of data saved and sent to the console. The module automatically removes data that is older than the specified age limit.

The function code also allocates data storage space depending on the type of report. Each SOE log function block handles only one type of report. The report includes data on all points configured in the recorder to be that type.

The recorder has function codes (independent of INFI 90 Open function codes) that:

- Define normally closed or open conditions for each of its input points.
- Turn scan off or on for each point.
- Allocate maximum prefault memory storage or sets a prefault window that determines the amount of data collected before a process fault.
- Designate specific prefault, post fault, snapshot, and summary events.
- Set logic that triggers prefault, post fault, and snapshot reporting.

The recorder continuously polls and saves data for each of its inputs. The MFP or MFC module polls a recorder every 250 milliseconds and saves collected event data in memory. How long the MFP or MFC module saves event data depends on the module configuration (i.e., maximum age of data saved). After the module retrieves data, it initiates an exception report to notify the console that it is ready to send data.

Refer to the ***Sequential Events Recorder*** instruction and ***Function Code Application Manual*** for further information (Table 1-2 lists instruction numbers).

INSEM01 Module

When data for an SOE log comes from a SEM module, the SEM module first gets the data from an associated MFP module. The SEM module notifies the console when it has data to send. The console then polls the SEM module for the data. This notify/request/reply sequence occurs only for a standard log. The operator must initiate the data collection for a summary log by changing the state of an RCM tag.

Before configuring an SOE log on the console, configure the SOE system. Specifically, configure the following function blocks in the SEM module:

- Executive block (FC 243).
- Addressing interface definition (FC 244).
- Input channel interface (FC 245).
- Trigger definition (FC 246).

Configure the following function blocks in the MFP modules associated with the SEM module:

- DSOE data interface SEM to MFP (FC 241).
- DSOE digital event interface (FC 242).

The executive block (FC 243) defines the maximum age of data saved and sent to the console. The module automatically removes data that is older than the specified age limit. Refer to the ***Function Code Application Manual*** for further information (Table 1-2 lists instruction numbers).

Sequence of Events (FC 210) Function Block

When data for an SOE log comes from an SOE (FC 210) function block, the MFP or MFC module collecting data gets its input directly from a digital input module (IMDSI02). The module provides the interface to the process. The MFP or MFC module notifies the console when it has data to send. The console then polls the MFP or MFC module for the data. This notify/request/reply sequence occurs only for a snapshot log. The operator must initiate the data collection for a summary log by changing the state of an RCM tag.

Before configuring an SOE log on the console, configure the SOE (FC 210) function block in an MFP or MFC module. This function code defines the maximum age of data saved and sent to the console. The module automatically removes data that is older than the specified age limit. The function code also allocates data storage space. Refer to the ***Function Code Application Manual*** for further information (Table 1-2 lists instruction numbers).

Console Requirements

Up to 160 SOE logs in any combination can be generated. When using a recorder, this accommodates up to 32 MFP or MFC module and recorder pairs, each capable of five different reports.

The steps required to completely configure an SOE log on the console include:

1. Define SOE log operating parameters. This affects all SOE logs.

2. Define an RCM tag as the trigger tag for the SOE log.

3. Define an SER input list to associate each input with a tag.

4. Define the SOE log.

SOE LOG OPERATING PARAMETERS

The sequence of events logging parameters defined on the console specify:

- Total number of recorders, SEM modules, or SOE (FC 210) function blocks that will be providing data.

- Total number of SOE logs to produce.

- Number of inputs per recorder, SEM module, or SOE (FC 210) function block.

All SOE logs use these parameters.

TRIGGER TAG

Each log has a dedicated RCM tag. This tag is the trigger tag for the log and is monitored to determine if a module has SOE data ready to send. The tag is also required to demand a printout of a summary log.

For a recorder or SOE (FC 210) function block, the RCM tag points to a sequence of events log function block in an MFP or MFC module, either FC 99 or FC 210 respectively. The tag is defined with the loop, PCU, and module address of the MFP or MFC module interfacing events data, and the block number of the SOE log function block. The block number is the memory address in the module that contains the function code.

For a SEM module, the RCM tag points to fixed block 5,000 (standard report) or 5,001 (summary report) in the SEM module. The tag is defined with the loop, PCU, and module address that points to the SEM module interfacing events data.

SER INPUT LIST

Before SOE data can be correctly printed, a correlation between recorder, SEM module, or SOE (FC 210) function block inputs and tags in the database is required. An SER input list contains a list of console index numbers (1 through

1,536) that correspond to each point being reported to the console by a recorder, module, or function block. A tag must be defined for each input to associate a tag with the input. This association is needed to show the correct tag name, description, and logic state descriptor in an SOE log entry.

NOTE: The tags defined in the input list during SER definition must be digital state reporting tags (e.g., DIGITAL, RMCB, DD).

For a ***recorder***, the physical connection point for field wiring is at the recorder terminal block. The point of connection determines the input number for a field input to the recorder. A recorder input number corresponds to a console input. For example, recorder input four corresponds to console input four.

For a ***SEM module***, the number of points reported to the console depends on the size of the SOE system. Each SOE subsystem can report 16 inputs to the SEM module. The n through n+15 outputs of the DSOE digital event interface function block in the first MFP module correspond to console inputs one through 32. For example, block output n+5 corresponds to console input six. The input numbers for any additional inputs depend on how the SOE subsystems are linked.

The ***SOE (FC 210) function block*** can have up to 32 inputs: 16 come from one digital input module, the other 16 from a second module. The n+1 through n+32 block outputs correspond to console inputs one through 32. For example, block output n+5 corresponds to console input five.

Defining SOE Log Operating Parameters

The first step in configuring the console for SOE logging is setting the operating parameters. This identifies the number of recorders, SEM modules, or SOE (FC 210) function blocks that are available to collect data from, the number of logs to create, and the number of inputs per recorder, SEM module, or function block. It establishes a base for later configuration procedures and determines the amount of disk space to allocate for SOE logging.

Use the *SOE General Parameters* option to set the SOE operating parameters. Figure 12-17 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *E SOE General Parameters*

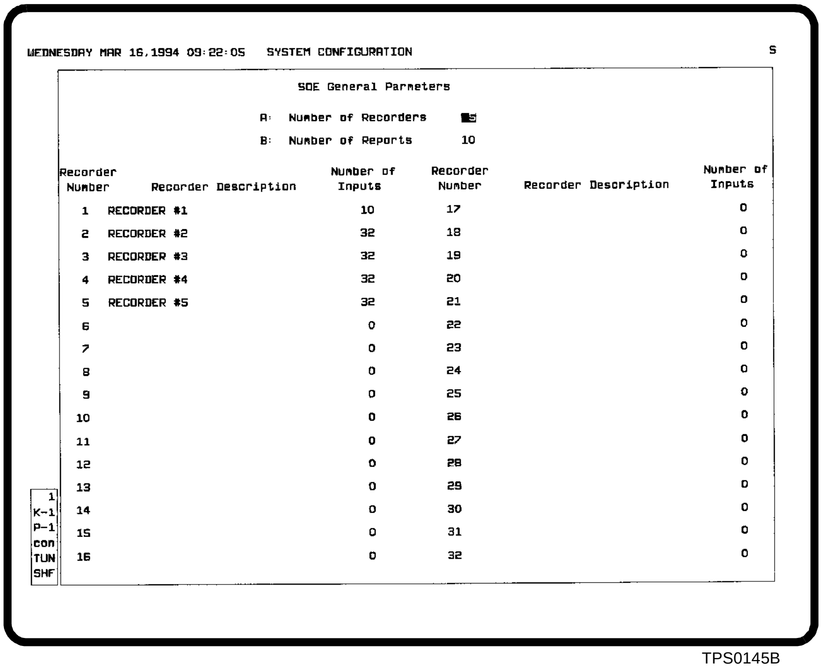


Figure 12-17. SOE General Parameters Page

To define the fields of this page:

1. Enter the appropriate data into each field of the page. Table 12-6 describes the fields. Refer to this table when entering data.

2. Press **ENTER**.

Table 12-6. SOE General Parameters Page Fields

Field	Description
Number of recorders	Number of sequential events recorders, INSEM01 modules, or SOE (FC 210) function blocks in the system available for data collection. Up to 32 recorders, modules, or function blocks are supported. A valid entry is from 0 to 32.
Number of reports	Number of SOE logs to be produced. Up to 160 SOE logs are supported. A valid entry is from 0 to 160.
Recorder description	Optional field; 32-character description.
Number of inputs	Number of inputs for a sequential events recorder, INSEM01 module, or SOE (FC 210) function block. A valid entry is from 0 to 1536. For a ***sequential events recorder***, set this to the maximum number of inputs to the recorder. 1,536 inputs per recorder are supported. The SER hardware configuration determines the actual number of inputs of which each recorder is capable. A base unit per recorder is 128. For an ***INSEM01 module***, set this to the maximum number of inputs being reported by the module. 1,500 sequence of events points are supported. For an ***SOE (FC 210) function block***, set this to the maximum number of inputs being recorded by the function block. A function block supports up to 32 inputs.

Defining an SOE Log RCM Tag (Trigger Tag)

Each log requires an RCM tag. For example, if the *Number of Reports* field on the SOE general parameters page is set to 25, then 25 RCM tags for SOE logging must be defined.

RECORDER OR SOE (FC 210) FUNCTION BLOCK

An RCM tag is used to monitor the SOE log (FC 99) function block or SOE (FC 210) function block in an MFP or MFC module. This function block is monitored to determine when to request SOE data. The operator also uses this tag to initiate data collection and printing of an SOE summary log. Refer to the discussion on generating sequence of events summary logs in the ***Operation*** instruction for the procedures to initiate data collection for a summary log (Table 1-2 lists instruction numbers).

To define the RCM tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Define a tag as a normal RCM tag except define the *Loop*, *PCU*, and *Mod* fields as the hardware address of the MFP or MFC module being used for SOE logging. The *Block* field must be the address of the SOE log (FC 99) function block or SOE (FC 210) function block.

INSEM01 MODULE

An RCM tag is used to monitor fixed block 5,000 or 5,001 in the SEM module. Block 5,000 is monitored to determine when to request SOE data for a standard log. Block 5,001 allows the operator to initiate data collection and printing of an SOE summary log. Refer to the discussion on generating sequence of events summary logs in the ***Operation*** instruction for the procedures to initiate data collection for a summary log.

To define the RCM tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Define a tag as a normal RCM tag except define the *Loop*, *PCU*, and *Mod* fields as the hardware address of the SEM module. The *Block* field must be set to either 5000 for a standard type log or 5001 for a summary type of log.

Defining an SER Input List

After setting the general parameters for SOE logging, an SER input list is required for each recorder, SEM module, or function block. This list links each input of a SEM module, recorder or function block to a tag defined in the console database. This list provides the ability to record tag database data (i.e., tag name, description, and logic state descriptor) in the log.

Use the *SER Definition* option to define SER input lists. Figure 12-18 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *C Logging*
→ *C Sequence of Events Log*
→ *A SER Definition*

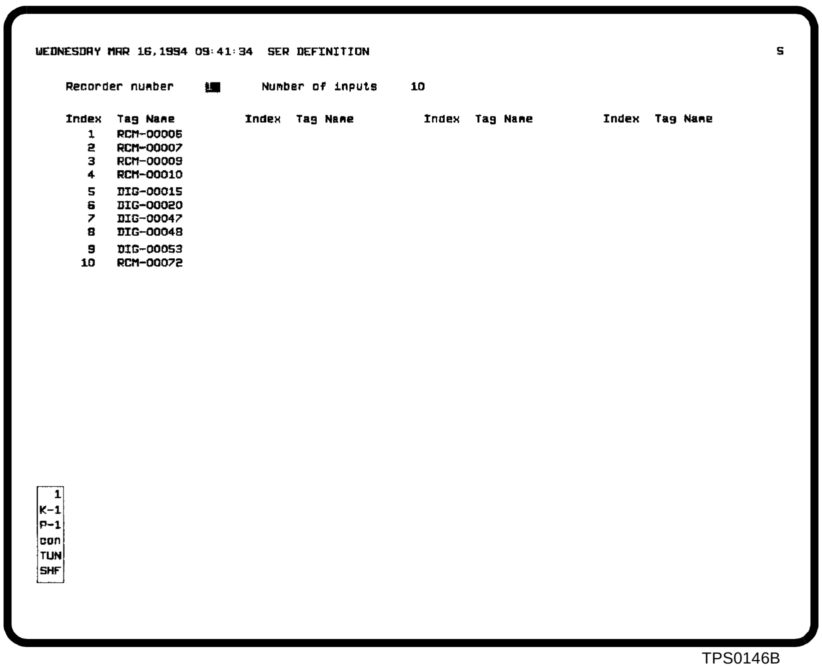


Figure 12-18. SER Definition Page

To define an SER input list:

1. Select a recorder.

   a. In the *Recorder number* field, type the number of a recorder (i.e., physical recorder, SEM module, or SOE (FC 210) function block) to define. A valid entry depends on the number of recorders set during SOE general parameters configuration (maximum 32). Refer to ***Defining SOE Log Operating Parameters*** in this section.

   b. Press **ENTER**.

The *Number of inputs* field reflects the number of inputs defined for the selected recorder, SEM module, or function block. This is also the number of index numbers that require definition.

2. Enter a tag name in the *Tag Name* field for each index number.

A maximum of 128 index numbers can display on the page. Press **NEXT PAGE** to call the next set of 128 index numbers. Continue to enter tag names for each index number until all inputs (i.e., index numbers) are defined.

3. Press **ENTER**.

Defining an SOE Log

Each SOE log is defined separately. The total number of logs that need to be defined depends on the *Number of Reports* field on the SOE general parameters page. The RCM tag (trigger tag) used for logging is identified in the definition of a log.

The SOE log (FC 99) function block can produce a standard, summary, prefault, post fault, and snapshot report. The SEM module and the SOE (FC 210) function block can produce a snapshot or summary report.

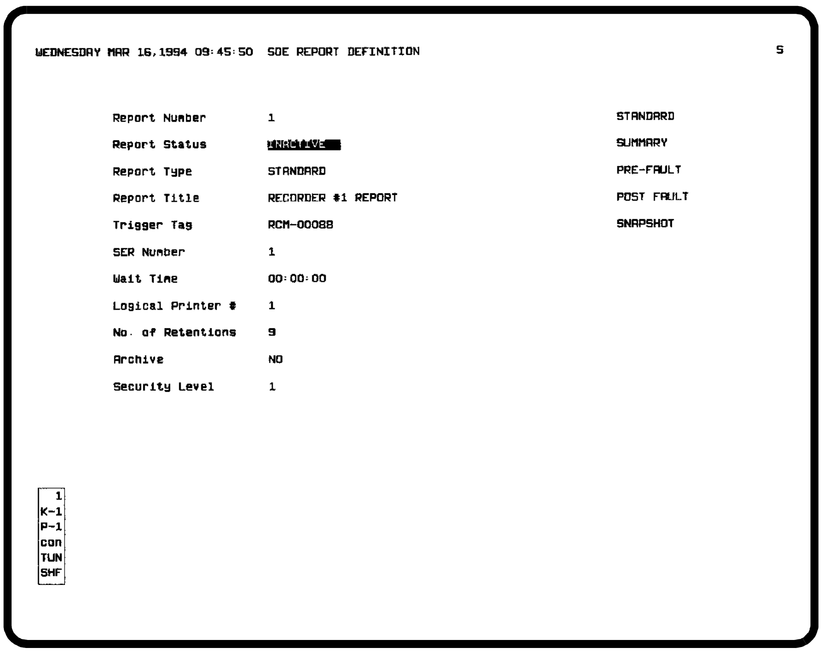
Each log has a unique identifier number and a 32-character (maximum) descriptor. A report trigger tag entered during SOE log configuration is the RCM tag associated with the MFP, MFC, or SEM module that is reporting SOE event data. An SER number defines the input list to use for the log. The number of retentions identifies the number of logs to save on disk for later printing.

A wait time can be entered for post fault and snapshot reports. This value tells the console how long in minutes to wait after notification of data from the MFP, MFC, or SEM module before polling the module for the data. The wait time should be shorter than the age limit set in the module configuration. This field is not valid for any other types of reports. The console continues to collect data until zero points are returned.

Use the *SOE Report Definition* option to define each SOE log. Figure 12-19 shows the page used. To choose the option, first

press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
C Logging
C Sequence of Events Log
B SOE Report Definition



TPS0147B

Figure 12-19. SOE Report Definition Page

To define a log:

1. Select a log.

   a. In the *Report Number* field, type the number of a log to define. A valid entry depends on the number of logs set during SOE general parameters configuration (maximum 160). Refer to ***Defining SOE Log Operating Parameters*** in this section.

   b. Press **ENTER**.

2. Initially, the input cursor positions on the *Report Status* field. A log must be inactive to edit. If the log is currently active, enter **INACTIVE** in the field and press **ENTER** Making a log inactive disables data collection and printing for the log. A queued print will still occur, however.

3. Enter the appropriate data into each field of the page. Table 12-7 describes the fields. Refer to this table when entering data.

4. Press **ENTER**.

Table 12-7. SOE Report Definition Page Fields

Field	Description
Report status	Shows the current log status and allows changing the status. A valid entry is: ACTIVE = activate the log (ON). INACTIVE = deactivate the log (OFF). The log must be inactive to edit.
Report type	Report type for printing purposes only. The actual report type is determined by the MFP, MFC, or SEM module. A valid entry is: PRE-FAULT SNAPSHOT SUMMARY POST FAULT STANDARD
Report title	32-character title. The title appears in the printout.
Trigger tag	Name of the RCM tag defined to support this SOE log.
SER number	Recorder (i.e., physical recorder, SOE (FC 210) function block, or SEM module) that provides data for this log. This is also the number of an SER input list. This field provides the link between SER input list and log. A valid entry is from 1 to 32. The maximum may be something less than 32 depending on the number of recorders enabled during SOE general parameters configuration.
Wait time	Amount of time the console is to wait after notification of data from the MFP, MFC, or SEM module before polling the module for the data. Valid for all report types except summary. **NOTE:** The wait time should be shorter than the age limit set in the module configuration.
Logical printer #	Logical printer used for printing. A valid entry is from 1 to 16.
No. of retentions	Number of log generations that are to be saved on disk. Up to nine most recently generated can be saved. The oldest log is removed when a new log is retained. A valid entry is 0 for no retentions or 1 to 9.
Archive	YES = enable archiving. NO = disable archiving.
Security level	Security level assigned to the log. The level determines the operations that can be performed on the log. A valid entry is from 1 to 16.

Activating an SOE Log

To start data collection for a defined log, change its status to active. A log can be activated as the final step of definition or it can be activated later by using the log status function. Refer to the discussion on log status in the ***Operation*** instruction for procedures to activate an SOE log.

To activate a log:

1. Call the SOE report definition page. Refer to ***Defining an SOE Log*** in this section for procedures to call the page.

2. In the *Report Status* field, enter **ACTIVE.**

3. Press **ENTER**.

Example SOE Log

Figure 12-20 is an example of an SOE log. The log format is fixed. The title line contains the following information:

- Report type from SOE report definition.
- Report number from SOE report definition.
- Report title from SOE report definition.
- Report triggered date and time. If SEM event triggered, this is the time of that trigger. Otherwise, this is the time the RCM tag (trigger tag) transitioned from zero to one.

STANDARD SOE REPORT 5 SEM STANDARD REPORT PAGE

TRIGGERED ON AUG 25,1994 19:05:13

Trg	Event Time	State	AID	Tag Name	Tag Description	SER Index
	19:04:40.564	ON	A	RCM-00090	RCM TAG 0090	16
	19:04:41.076	ZERO		RCM-00010	RCM TAG 0010	1
	19:04:41.076	ZERO		DIG-00015	DIGITAL TAG 0015	2
	19:04:41.076	ZERO		DIG-00020	SECOND STATE CHANGER	3
	19:04:41.076	ZERO		DIG-00047	DIGITAL TAG 0047	4
	19:04:41.076	ZERO		DIG-00048	DIGITAL TAG 0048	5
	19:04:41.076	ZERO		DIG-00053	DIGITAL TAG 0053	6
	19:04:41.076	ZERO		DIG-00057	DIGITAL TAG 0057	7
	19:04:41.076	ZERO		DIG-00064	DIGITAL TAG 0064	8
	19:04:41.076	ZERO		DIG-00065	DIGITAL TAG 0065	9
	19:04:41.076	STRTUP		RCM-00072	RCM TAG 0072	10
	19:04:41.076	ZERO		DIG-00077	DIGITAL TAG 0077	11
	19:04:41.076	ZERO		DIG-00080	DIGITAL TAG 0080	12
	19:04:41.076	NORMAL		DIG-00856	P206M20 DIG ALRM (BANK 19)	13
	19:04:41.076	CLOSED		RCM-00088	RCM TAG 0088	14
	19:04:41.076	ZERO		RCM-00089	RCM TAG 0089	15
	19:04:41.076	OFF		RCM-00090	RCM TAG 0090	16
	19:05:09.748	SHTDWN	A	RCM-00010	RCM TAG 0010	1
	19:05:10.260	ONE	A	DIG-00015	DIGITAL TAG 0015	2
	19:05:10.772	ONE	A	DIG-00020	50 SECOND STATE CHANGER	3
	19:05:11.284	ONE	A	DIG-00047	DIGITAL TAG 0047	4
	19:05:11.796	ONE	A	DIG-00048	DIGITAL TAG 0048	5
	19:05:12.308	ONE	A	DIG-00053	DIGITAL TAG 0053	6
	19:05:12.820	ONE	A	DIG-00057	DIGITAL TAG 0057	7
	19:05:13.332	ONE	A	DIG-00064	DIGITAL TAG 0064	8
>>>	19:05:13.844	ONE	A	DIG-00065	DIGITAL TAG 0065	9

TPS1142A

Figure 12-20. Example SOE Log Printout

The body of the report contains the following information:

- Trigger indicator. A >>> appears next to the event time if the recorded event was the SEM trigger (Figure 12-20).
- Event time.
- State (logic state descriptor).
- Alarm ID (blank or A for alarm state).
- Tag name.
- Tag description.
- SER input list index number.

SECTION 13 - PERIPHERALS

INTRODUCTION

This section discusses the functions used to configure console peripherals. Specifically, this section includes information on:

- System configuration for peripherals.
- Network configuration for printers.
- Defining DEVSTAT tags for printers.
- Defining physical printers as logical printers.
- Changing the color map for log printing.
- Enabling printer failover.

NOTE: This section discusses the configuration options that affect printers used for logging. Refer to the discussion on the screen copy function in the ***Operation*** instruction for information on configuration, selection, and failover for printers used for screen printing (Table 1-2 lists instruction numbers).

ENABLING PERIPHERALS

Several different peripheral devices are used to support data storage and recording and to support process monitoring and control. These include keyboards, annunciator display panels, monitors, printers, touch screens, and data storage devices. Besides enabling and defining the number of peripherals being used, peripheral configuration requirements are few. The number of each type of device being used is accounted for during system configuration. Refer to the ***Hardware*** instruction for information on peripheral connections (Table 1-2 lists instruction numbers).

PERIPHERAL ERRORS

The console has the ability to monitor the status of its own peripherals and indicate problems through operator action requests and the diagnostic log. An operator action request message is automatically generated when a keyboard or printer failure occurs.

NETWORK CONFIGURATION FOR PRINTERS

Each printer used for log printing must be identified to the console by using the **DEFINEDEVICES** procedure. This is also required for a printer that is used for screen printing. Refer to the ***File Utilities*** instruction for an explanation of this procedure (Table 1-2 lists instruction numbers).

DEFINING DEVSTAT TAGS FOR PRINTERS

A DEVSTAT tag identifies the type of printer being used for logging. The number of printers the console can access is set during system configuration. Page mode printers should not have DEVSTST tags but all other printers require a DEVSTAT tag to identify its type.

PRINTERS

A single console supports four physical printers for logging. The number of physical printers is defined on the general parameters page during system configuration. The following types of printers can be used for printing logs:

- ANSI low speed, no color.
- ANSI low speed, color.
- ANSI high speed, no color.
- IBM, no color.
- IBM, color.
- IBM, black and white, 24-pin.
- IBM, color, 24-pin.
- Printers that support HP PCL-5 printer control language.

Refer to the discussion on the screen copy function in the ***Operation*** instruction for information on printers used for printing a copy of a screen (Table 1-2 lists instruction numbers).

In addition to defining a DEVSTAT tag for each log printer, printer configuration requirements include:

- Printer assignment.
- Printer color maps.

A printer assignment identifies a physical printer as a logical printer. A logical printer number is used in functions to assign a printer rather than a physical printer number. Printer color maps configuration is only required if the default colors for log printing are to be changed.

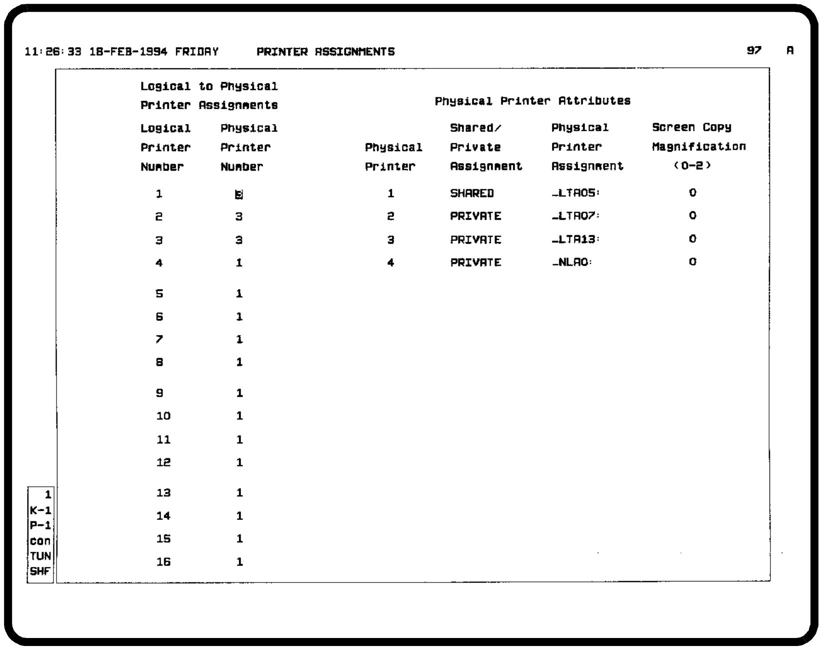
Printer Assignment

A printout is directed to a specific physical printer by logical printer number. Logical printer assignment associates a logical printer number to a physical printer. Up to 16 logical printer assignments can be made. The printer assignments also define a printer as a shared printer or a private printer. Refer to ***Private/Shared Printer*** in this section.

Use the *Printer Assignment* option to define a logical printer. Figure 13-1 shows the page used. To choose the option, first

press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *C Printer Assignment*



TPS0223C

Figure 13-1. Printer Assignment Page

To make a printer assignment:

1. In the *Physical Printer Number* field for a logical printer, enter the number of a physical printer. A valid entry is from 1 to 4. Whenever a printout is directed to this logical printer, it will occur at the physical printer defined here.

> **NOTE:** Only logical printers one through nine can be assigned using the *ASGN PRN* option.

2. Enter either **PRIVATE** or **SHARED** in the *Shared/Private Assignment* field. This determines if the printer is dedicated to a console or if it is available to other main or auxiliary consoles on the network.

> **NOTE:** The fields in the *Screen Copy Magnification* column are not used by this console.

3. Press **ENTER**.

The printer assignments are put into effect immediately after pressing **ENTER**. All new output is routed to the physical printer defined here. Anything queued to a printer before this configuration update remain queued to that printer.

The *Physical Printer Assignment* field shows the port to which each physical printer is attached. This is for reference only. Changing port assignments requires running the **DEFINEDEVICES** utility. Refer to the ***File Utilities*** instruction for the procedures (Table 1-2 lists instruction numbers).

Private/Shared Printer

A private printer is required to print events and operator actions (i.e., system event log) that are configured to print as they occur. This type of log prints line-by-line. As a result, the methods used by the console to allocate and deallocate a printer for this type of log make it impractical to use a shared printer. If desired, a private printer can also be used to print page-oriented logs such as a custom or SOE log.

The designation in the *Shared/Private Assignment* field can be changed as needed between shared and private. Only one console can designate a printer as private. All other consoles must have the printer deleted from their printer assignment list. For the console designating the printer as private, no further restrictions apply. After the designation is changed to private, any type of printout can be sent to that printer.

When changing a printer from private to shared, the event log or operator actions log must first be directed to another private printer before the change is made if either is being sent to the printer being changed. Refer to ***SYSTEM EVENT LOG*** in Section 12 for the procedures. When complete, the printer can be designated as a shared printer and any console can then direct page-oriented type logs or reports to the printer. The continuously printing system event log can no longer be sent to the printer.

Printer Color Maps

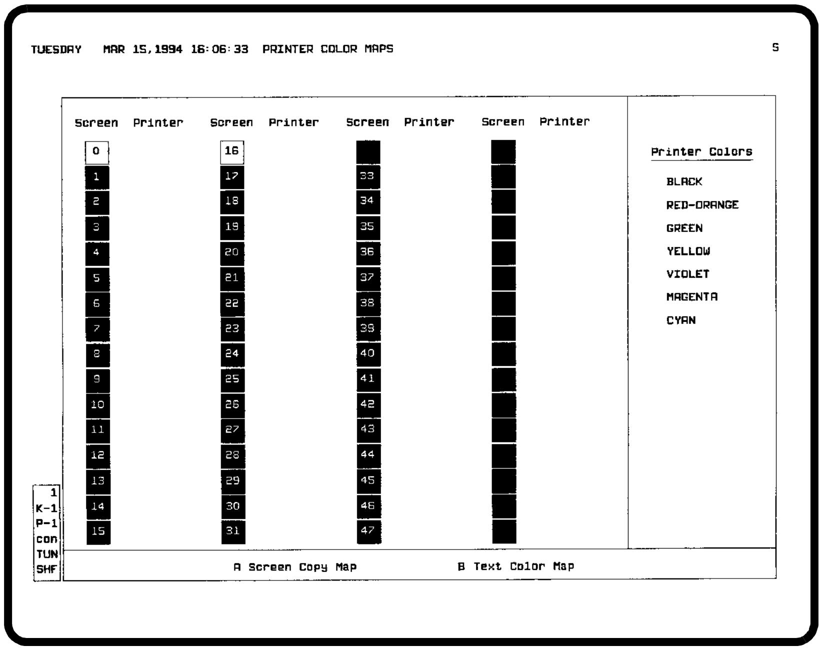
The printer color maps function sets the colors used for printing text files. Text file printouts are generated by functions such as logging. The available printer colors for these types of printouts are:

Black	Magenta
Blue-magenta	Red
Cyan	Yellow
Green	

NOTE: The *Screen Copy Map* option does not function for this console (Figure 13-2).

Use the *Printer Color Maps* option to set printer colors for text prints. Figure 13-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *H Printer Color Maps*



TPS0155B

Figure 13-2. Printer Color Maps Page

Text Color Map

During log configuration, colors can be selected for the printout of a log. Color codes (e.g., %K0) set the printer to a specific color. Any text following a color code prints in a color.

Part of the code is a number that identifies a color. The number can range from 0 to 63, which also corresponds to the 64 available screen colors. Text color mapping allows mapping the codes for colors to any available printer colors.

A default printer color map for text is provided. Table 13-1 shows the color mapping for some color codes. The color codes shown in the table are the standard color codes used. Any number from zero to 63 can be used in a color code. For example, in the default color map, %K3, %K19, %K35 and %K51 can all actually be used to select a green text color (Figure 13-3).

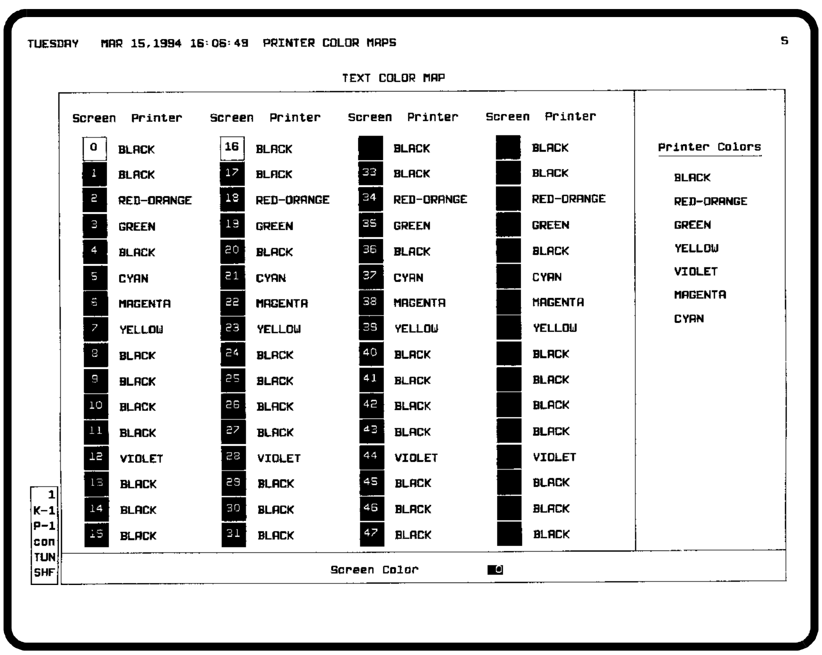
Table 13-1. Color Mapping for Color Codes

Code	Color
%K0	Black
%K2	Red
%K3	Green
%K5	Cyan
%K6	Magenta
%K7	Yellow
%K12	Blue-magenta

The text color map is referenced when a log prints. For example, to have a log entry print in red, the log must have a color code that selects a color index set to *RED-ORANGE*. If using the default settings, color index number two is set to red; therefore, the log code should be %K2.

To change the default mapping or to define additional color codes:

1. Choose *B Text Color Map*. Figure 13-3 shows the next page that appears after choosing the option.



TPS0157B

Figure 13-3. Printer Color Maps Page - Text Color Map Option

2. Select a color code to define.

   a. Enter the index number of the color to change in the *Screen Color* field.

Optionally, the *Screen Color* field can be left at its default zero. The cursor can then be moved to a specific color.

b. Press **ENTER**. The input cursor positions on the selected color. The number in the *Screen* column is the same as the code number. For example, select screen color 40 to define printer color code %K40.

3. Enter a printer color that this color code is to produce when used in a log. The available colors appear under the *Printer Colors* heading.

4. Press **ENTER**. Any changes made are written to the hard disk and put into immediate effect for all current and subsequent printings. If a color printer is in the middle of a print, a change in color causes the printer to change to the new color in the middle of the print.

Example: If it is desired to use color code %K15 to print green rather than its default black, change its printer color from *BLACK* to *GREEN*.

Logging Printer Failover

The peripheral device failover function can be defined to enable automatically transferring the work load of a failed printer used for logging (event, custom, summaries, etc.) to a working printer with no information lost or delayed. A printer used for logging can only be reassigned to another printer already accounted for on the console. A printer is accounted for by using the **DEFINEDEVICES** procedure and performing system configuration.

NOTES:

1. This section discusses failover for printers used for logging. Refer to the discussion on the screen copy function in the ***Operation*** instruction for information on failover for printers used for screen printing (Table 1-2 lists instruction numbers).

2. The fields for monitor failover are not valid for this console.

PRINTER FAILURE

When it is determined that a printer has failed, a check for a failover assignment is made. If a working printer is found, the currently queued work for the failed printer reroutes to the working printer (backup printer). The backup printer does not cancel any printing it is currently performing. It completes the current print job then alternates between any additional print jobs assigned to it and any additional assignments made to the failed printer. The backup printer also reprints from the beginning a print job the failed printer was in the middle of when it failed.

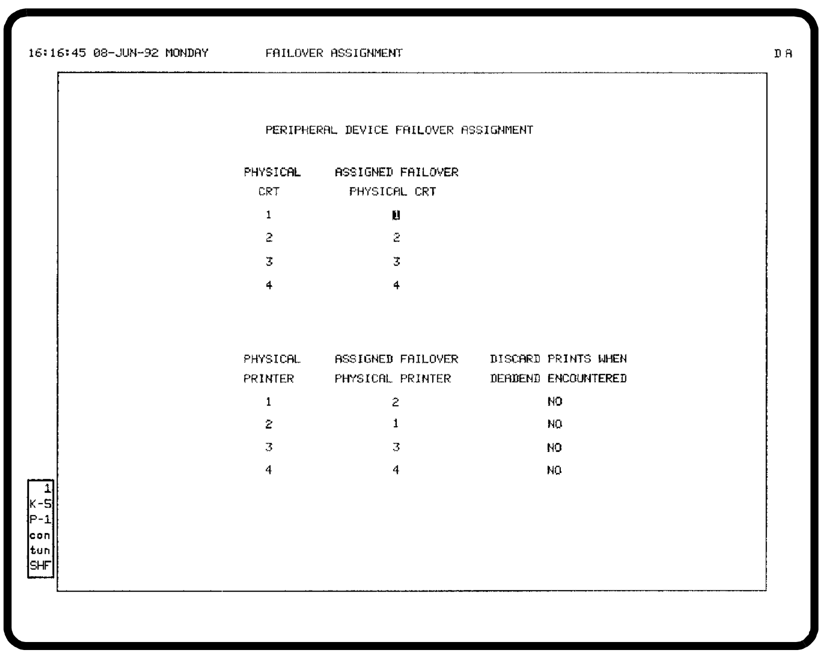
The failover processing continues as long as the console identifies a primary printer as being off-line. When it is determined that the failed printer is back on-line, failover automatically discontinues and any new work is directed to the primary printer. Also, the primary printer assumes any print jobs already queued to the backup printer. The backup printer continues with its own work.

NOTE: For a failed printer to be recognized as being back on-line, the operator action that indicated the failure must be completely processed and cleared.

DEFINING FAILOVER DEVICES

Use the *Peripheral Failover* option to define printer failover. Figure 13-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *K Peripheral Failover*



TPS0962B

Figure 13-4. Peripheral Failover Page

To define failover assignments:

1. Move to the fields in the *ASSIGNED FAILOVER PHYSICAL PRINTER* column. The monitor failover fields do not function for this console.

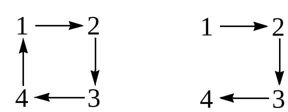
2. In the *ASSIGNED FAILOVER PHYSICAL PRINTER* column, enter a printer number from 1 to 4 for each physical printer. The designation selects the printer that is to automatically assume the work load of the physical printer if it should fail. If a physical printer is to not automatically failover, define itself as the failover printer. A failover assignment cannot be made to a physical printer the console does not currently support.

> **NOTE:** Failover from a private printer to a shared printer is not permitted. A private printer can only failover to another private printer.

3. The *DISCARD PRINTS WHEN DEADEND ENCOUNTERED* column determines whether printing assignments made to a failed printer are to be discarded if a working printer cannot be found or if they are to be saved until a working printer can be found. Enter **YES** to discard or **NO** to save. The discard option works whether a backup printer is selected or not.

4. Press **ENTER**.

Example: The peripheral failover page allows setting up a printer failover sequence. For example, printer one can be set up to failover to printer two, printer two to printer three, printer three to printer four, and printer four to printer one. By setting up failover in this way, all supported printers will be checked to find a working printer.



SECTION 14 - OPERATOR CONFIGURABLE DISPLAYS

INTRODUCTION

This section explains the configuration used to change operator configurable displays operation. The *Operator Displays' Faceplate* option is used to define the operating parameters for the operator configurable displays function. A default configuration is provided that can be used as is or can be modified if desired. This configuration is not required unless the default configuration for the operator configurable displays function requires changes.

NOTE: Become thoroughly familiar with the operator configurable displays function ***before*** attempting operator displays' faceplates configuration.

Hint If it is desired to replace a default, standard faceplate symbol used by the function with a user-created faceplate symbol, the easiest way to incorporate this new symbol is to create it in the exact same size as the symbol it is replacing. This will avoid having to modify the operating parameters for the function to accommodate for a symbol with a unique size. By doing this, the standard symbol can easily be replaced by specifying the name of the user-created symbol in place of the default symbol name. If the exact size cannot be maintained, using a multiple of the standard size will also make incorporating the symbol somewhat easier.

OPERATOR DISPLAYS' FACEPLATES

The parameters defined with the *Operator Displays' Faceplates* option are used by the operator configurable displays function. The parameters determine the appearance of the editing page for the function and are also used to construct the final, saved display file created with the function. The operator displays' faceplates function:

- Defines the faceplate symbols and trend symbols used in operator configurable displays. The default symbols used to create operator configurable displays are provided with the console. Any user-created symbols can be used instead of these default symbols.

- Defines the colors used in operator configurable trend displays.

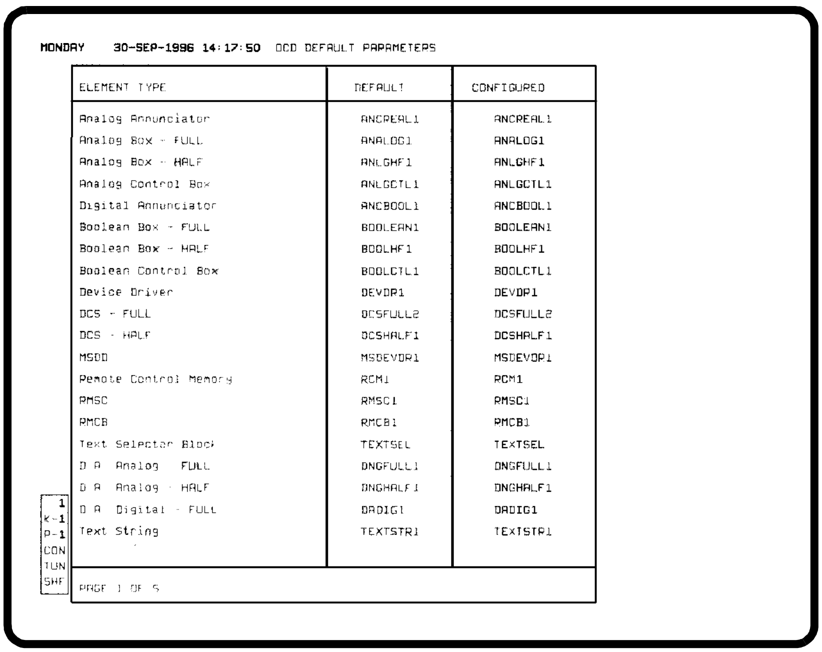
- Configures the display format used during creation of an operator configurable display (i.e., title position, box separation, element height, etc.).

- Selects the line format used in an alarm summary element of an operator configurable display.

- Defines the number of horizontal and vertical boxes a certain type of display element consumes when setting up a display. This depends on the faceplate symbol type and size.

Use the *Operator Displays' Faceplates* option to configure the operator configurable displays function. Figure 14-1 is the first page of the function. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Display*
→ *C Operator Displays' Faceplates*



MONDAY 30-SEP-1996 14:17:50 OCD DEFAULT PARAMETERS

ELEMENT TYPE	DEFAULT	CONFIGURED
Analog Annunciator	ANCREAL1	ANCREAL1
Analog Box - FULL	ANALOG1	ANALOG1
Analog Box - HALF	ANLGHF1	ANLGHF1
Analog Control Box	ANLGCTL1	ANLGCTL1
Digital Annunciator	ANCBOOL1	ANCBOOL1
Boolean Box - FULL	BOOLEAN1	BOOLEAN1
Boolean Box - HALF	BOOLHF1	BOOLHF1
Boolean Control Box	BOOLCTL1	BOOLCTL1
Device Driver	DEVDR1	DEVDR1
DCS - FULL	DCSFULL2	DCSFULL2
DCS - HALF	DCSHALF1	DCSHALF1
MSDD	MSDEVDR1	MSDEVDR1
Remote Control Memory	RCM1	RCM1
RMSC	RMSC1	RMSC1
RMCB	RMCB1	RMCB1
Text Selector Block	TEXTSEL	TEXTSEL
D A Analog FULL	DNGFULL1	DNGFULL1
D A Analog - HALF	DNGHALF1	DNGHALF1
D A Digital - FULL	DADIG1	DADIG1
Text String	TEXTSTR1	TEXTSTR1

PAGE 1 OF 5

TPS0017C

Figure 14-1. Operator Displays' Faceplates Page - 1 of 5

This configuration encompasses five pages. The *DEFAULT* column lists the defaults set up for operator configurable displays. Access is limited to the *CONFIGURED* column. Any changes are made in this column. The default parameters always appear for reference to easily recover from undesired changes. Press **NEXT PAGE** and **PREV PAGE** to sequence through each page of this function.

Pages 1 and 2

Page one defines the symbol files that are to be used for each type of element that can be part of an operator configurable display. The appearance of a faceplate and the interaction that can be performed by using the faceplate are defined in the symbol file. The element types include:

Analog Annunciator.
Analog box - full and half size.
Analog Control Box.
Boolean box (digital) - full and half size.
Boolean Control Box.
Device Driver.
Digital Annunciator.
DCS (digital control station) - full and half size.
MSDD (multi-state device driver).
Remote Control Memory.
RMSC (remote manual set constant).
RMCB (remote motor control block).
Text Selector Block.
D. A. Analog - full and half size.
D. A. Digital.
Text String.

The default identified for each of these element types is the standard symbol provided with the console. The name is the assembled symbol file name without its ***DL*** extension. This page allows entering the name of a user-created symbol to be used in place of the default symbol. The name must be of an assembled symbol file.

Page two performs the same function as page one but is used to substitute user-created trend display elements for defaults. It also allows changing the default colors that identify each of five possible trends that can appear in an operator configurable display trend element. A list of available colors and their index numbers are provided on the display.

Page 3

Page three specifies horizontal (x-coordinate) and vertical (y-coordinate) starting points used when constructing the operator configurable displays editing page and the final, saved display file. These positions are also used to construct touch points. Page three defines:

- Display title position (i.e., display name).
- Starting position for drawing the display.
- Separation space between each box that appears on the editing page.

- Number of boxes that appear horizontally on the editing page.
- Number of boxes that appear vertically on the editing page.
- Height and width of an *unconfigured* box on the editing page.
- Height and width of a full and half size trend box on the editing page.
- Line format to use for an alarm summary element.

Screen space available as user space is 400 to 9,600 horizontal (x-coordinate) and 400 to 7,200 vertical (y-coordinate). The title line is positioned by default at the center and top of the window above the user space.

Pages 4 and 5

Page four and page five define the number of horizontal and vertical unconfigured boxes an element occupies on the editing page. This must be consistent with the number of horizontal and vertical boxes and the height and width of an *unconfigured* box defined on page three.

The actual size of the faceplate symbol being used determines the number of horizontal and vertical boxes to allocate for the faceplate. Since the function does not know the exact size of the symbol it is incorporating, it estimates the size with the unconfigured boxes to maintain adequate spacing between symbol elements. When performing configuration, page three, page four, and page five are dependent on each other.

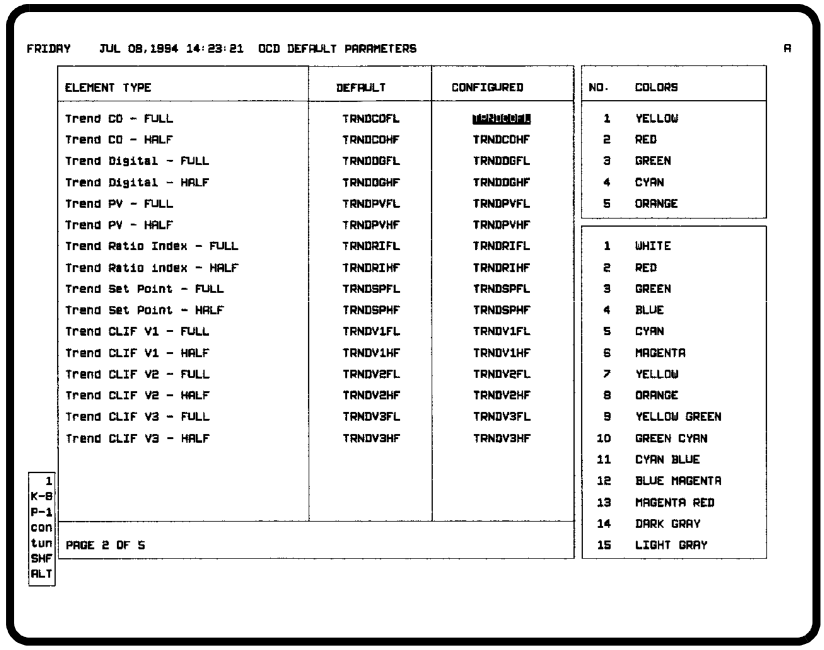
Defining Symbol Files

Page one and page two define the symbol files to use for each type of tag and the symbol files to use in a trend element. Page two also defines colors used in trend elements. Figure 14-1 shows *PAGE 1 OF 5* and Figure 14-2 shows *PAGE 2 OF 5*.

On *PAGE 1 OF 5* and *PAGE 2 OF 5*, enter the name of a symbol in the *CONFIGURED* column for each type of tag or trend element. This becomes the symbol that will be used for the tag or trend element. The name entered must be of an assembled symbol file (***DL***) that resides in any of the [DATA.USN44] through [DATA.USN4E] directories. If the symbol file does not exist, the following error message appears:

Display Not Found in Given Directory

Press **HOME** to restore the default or previous file name. Anytime a tag element is configured in an operator configurable



TPS0018C

Figure 14-2. Operator Displays' Faceplates Page - 2 of 5

display, the symbol file defined here will be used for the tag based on its tag type.

Defining Trend Colors

On *PAGE 2 OF 5* the current color choices for trend elements appear in the upper right box. The colors listed in the *COLORS* column are for each of five possible trends that can appear in an operator configurable display trend element. The numbers in the *NO.* column correspond to trends one through five.

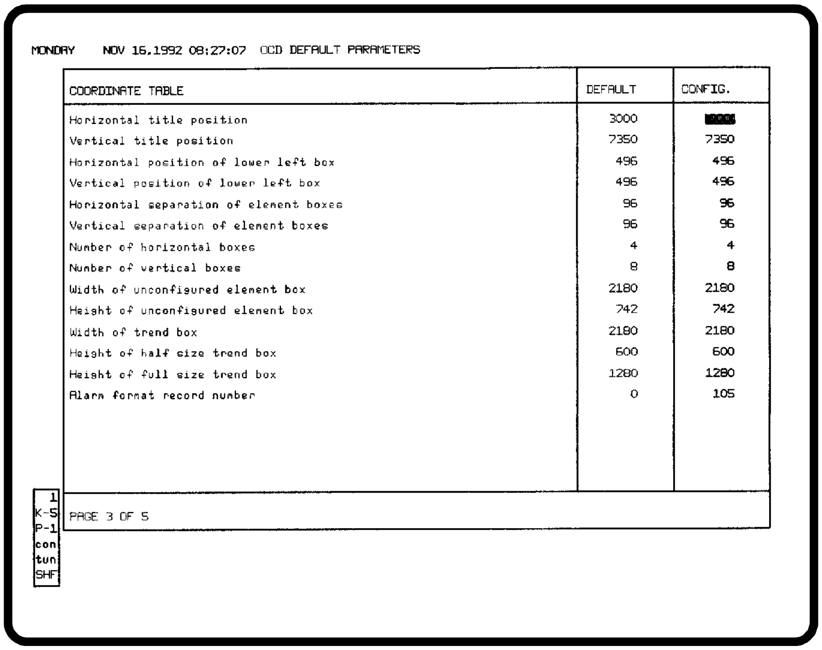
To define a trend color:

1. In the *COLORS* column, enter the index number or name of a color. The bottom right box of the page contains a list of available colors. The names appear in their actual colors.

2. Press **ENTER**.

Defining Operating Parameters

Page three defines various parameters used to build an operator configurable display. These parameters are also used when the operator configurable displays editing page is called. Figure 14-3 shows *PAGE 3 OF 5*. Page four and page five determine the number of unconfigured boxes each tag element, trend element, and alarm summary element consumes. The height and width specified in the source file of a faceplate symbol determines its actual size when drawn, not the operator configurable displays function.



MONDAY NOV 16,1992 08:27:07 OCD DEFAULT PARAMETERS

COORDINATE TABLE	DEFAULT	CONFIG.
Horizontal title position	3000	[illegible]
Vertical title position	7350	7350
Horizontal position of lower left box	496	496
Vertical position of lower left box	496	496
Horizontal separation of element boxes	96	96
Vertical separation of element boxes	96	96
Number of horizontal boxes	4	4
Number of vertical boxes	8	8
Width of unconfigured element box	2180	2180
Height of unconfigured element box	742	742
Width of trend box	2180	2180
Height of half size trend box	600	600
Height of full size trend box	1280	1280
Alarm format record number	0	105

PAGE 3 OF 5

1
K-S
P-1
con
tun
SHF

TPS0019B

Figure 14-3. Operator Displays' Faceplates Page - 3 of 5

When creating symbol files to be used in operator configurable displays, a standard height and width should be used for all faceplate symbols. A standard size is important since this configuration affects all faceplate symbols of the operator configurable displays function and assumes the same size has been used for all symbols.

The height and width limits set during *PAGE 3 OF 5* configuration determine the size of an unconfigured box that appears on the operator configurable displays editing page. The height and width are also used to construct touch points. This configuration also affects the title position, the starting position, and the separation between elements for both the editing page and the final saved display file.

NOTE: Configuration of *PAGE 4 OF 5* and *PAGE 5 OF 5* is dependent on configuration of *PAGE 3 OF 5*.

Page 3 To define the fields of *PAGE 3 OF 5*:

1. In the *title position* fields, enter the horizontal (x-coordinate) and vertical (y-coordinate) position at which the display title is to appear. A valid entry for the horizontal position is from 400 to 9600; a valid entry for the vertical position is from 400 to 7350.

2. In the *position of lower left box* fields, enter the horizontal and vertical position at which to start drawing an operator configurable display. A valid entry for the horizontal position is from 400 to 9600; a valid entry for the vertical position is from 400 to 7200.

3. In the *separation of element boxes* fields, enter the horizontal and vertical spacing between elements. This defines the number of display units separating each unconfigured box and each element of the final display. At least 40 display units of separation are required for proper touch point construction. A valid entry is from 1 to 2000.

> **NOTE:** When defining the next parameters, the source files for the faceplate symbols should be referenced.

4. Enter the number of boxes that are to display horizontally in the *Number of horizontal boxes* field and the number that are to display vertically in the *Number of vertical boxes* field. There can be no more than 32 boxes total on a single page. A valid entry for these two fields is from 1 to 32.

For example, if the horizontal number of boxes is 16, then the vertical can be no more than two. These values are dependent on the actual height and width of each faceplate symbol.

The number of unconfigured boxes each faceplate symbol consumes is set on page four and five. When setting these attributes, the height and width of a box and separation between each box must be considered.

5. Enter a width in display units for an unconfigured box in the *Width of unconfigured element box* field. Enter a height in the *Height of unconfigured element box* field. This sets the size of an unconfigured box that appears on the editing page. The width of a box should be the same as the width or x-offset set in the source files of the faceplate symbols. The height should be the same as the height or y-offset. The entries must be no more than the available user space which is from 1 to 9200 horizontally and 1 to 6800 vertically.

> **NOTE:** This software release does not support variable trend box sizes. The *Width of trend box*, *Height of half size trend box*, and *Height of full size trend box* fields cannot be changed from default.

6. Enter the number of a line format in the *Alarm format record number* field. This determines the format an alarm summary element uses. There are 106 possible formats: 0 through 4 are fixed and 5 through 105 are user-definable. Refer to ***ALARM SUMMARY REPORT*** in Section 9 for further explanation and also for the procedures to define a line format.

7. Press **ENTER**.

8. Press **NEXT PAGE** to call *PAGE 4 OF 5*.

Pages 4 and 5

Page four and five define the number of unconfigured boxes a certain type of display element consumes. For example, default faceplate symbols consume one horizontal and up to four vertical boxes depending on the chosen size. Alarm summaries use four horizontal and up to eight vertical boxes depending on the chosen size. Figures 14-4 and 14-5 show *PAGE 4 OF 5* and *PAGE 5 OF 5* respectively. The actual size defined in the assembled symbol file and the parameters defined on *PAGE 3 OF 5* should be referenced when defining the number of boxes an element consumes.

MONDAY NOV 16,1992 08:01:02 OCD DEFAULT PARAMETERS

ELEMENT TYPE	DEFAULT		CONFIGURED	
	HORIZON.	VERTICAL	HORIZON.	VERTICAL
Analog Annunciator	1	2	1	2
Analog Box, full size	1	2	1	2
Analog Box, half size	1	1	1	1
Boolean box, full size	1	2	1	2
Boolean Box, half size	1	1	1	1
Digital Annunciator	1	2	1	2
Digital Control Station - Full	1	4	1	4
Digital Control Station - Half	1	2	1	2
Multi State Device Driver	1	2	1	2
Remote Control Block	1	2	1	2
Remote Manual Set Constant Block	1	2	1	2
1/4 - 1/2 screen Trend box	2	2	2	2
1/4 - 1 screen Trend box	4	2	4	2
1/2 - 1/2 screen Trend box	2	4	2	4
1/2 - 1 screen Trend Box	4	4	4	4
Full Screen Trend Box	4	8	4	8
Alarm Summary - 1/8 screen	4	2	4	2
Alarm Summary - 1/4 screen	4	4	4	4
Alarm Summary - 1/2 screen	4	8	4	8
Alarm Summary - Full screen	4	8	4	8

1
K-5
P-1
con
tun
SHF

PAGE 4 OF 5

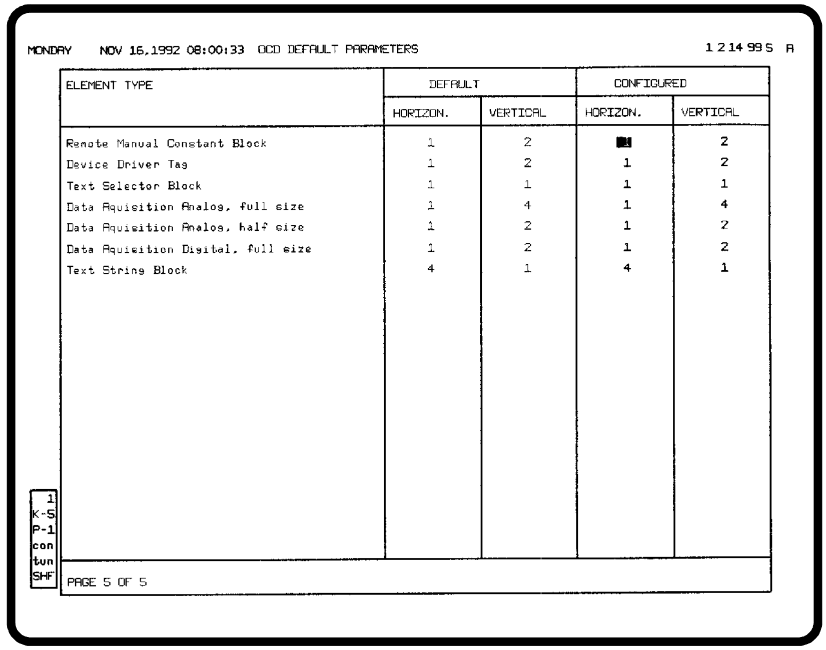
TPS0020B

Figure 14-4. Operator Displays' Faceplates Page - 4 of 5

To define the fields of *PAGE 4 OF 5* and *PAGE 5 OF 5*:

1. Enter the number of horizontal and vertical unconfigured boxes an element will consume in the *HORIZON.* and *VERTICAL* fields for each element. Press **NEXT PAGE** and **PREV PAGE** to move between *PAGE 4 OF 5* and *PAGE 5 OF 5*.

2. Press **ENTER**.


MONDAY NOV 16,1992 08:00:33 OCD DEFAULT PARAMETERS 1 2 14 99 S A

ELEMENT TYPE	DEFAULT		CONFIGURED	
	HORIZON.	VERTICAL	HORIZON.	VERTICAL
Remote Manual Constant Block	1	2	1	2
Device Driver Tag	1	2	1	2
Text Selector Block	1	1	1	1
Data Aquisition Analog, full size	1	4	1	4
Data Aquisition Analog, half size	1	2	1	2
Data Aquisition Digital, full size	1	2	1	2
Text String Block	4	1	4	1

Figure 14-5. Operator Displays' Faceplates Page - 5 of 5

SECTION 15 - DISPLAY CALL-UP OPTIONS

INTRODUCTION

This section explains the various display call-up methods that can be set up and describes the configurations required to enable the different call-up options. Additionally, this section explains and gives the procedures to set up a primary display.

CALL-UP OPTIONS

Several options are provided for display call-up:

- Keyboard key.
- Annunciator display panel (ADP) pushbutton.
- Display key select.
- Display touch point.
- Automatic display.

A user-written program (i.e., user task) also has the ability to call up displays. Refer to either Section 19 or Section 20 for an explanation.

Function Key and ADP Pushbutton

A single display can be assigned to a keyboard function key and ADP pushbutton. A multiwindow display call-up can also be assigned to a function key or an ADP pushbutton in the same way as a single display call-up. A multiwindow display function allows defining a list of displays to be called to specified windows. Refer to Section 8 for an explanation of the procedures to make key and pushbutton assignments. Refer to ***MULTIWINDOW DISPLAYS*** in this section for the procedures to define a multiwindow display list.

Key Select and Touch Point

A single display or a multiwindow display call-up can be built into a display as a key select and touch point. Refer to the ***Display Builder Reference*** instruction for information about the display commands that are used to enable these types of display call-ups (Table 1-2 lists instruction numbers).

Automatics

An automatic display can be a complete display or a pop up element that appears after being triggered by a process event. Refer to ***AUTOMATIC DISPLAYS AND POP UPS*** in this section for an explanation.

PRIMARY DISPLAY

A primary display assigned to a tag provides quick access to tag functions. The window the display is to appear on is configurable. One use for this capability is to assign a display on which actions can be performed to correct a problem. A problem is usually identified by a tag in alarm.

The primary display for a tag can be called in three different ways. The first is to select the tag element on a page, then press DISPLAY. The second method requires using a selector that is part of an alarm summary. The third method is to select the primary display option from the operating parameters page. A tag must be assigned a primary display during tag configuration to enable this display call-up feature.

To configure a primary display for a tag:

1. Follow the steps given in ***Defining a Tag*** in Section 6.

2. Enter the name of the display in the *Name* position of the *Primary Disp-Name Crt* field. The name must be of an assembled display (***DU***).

3. Enter the number of the window that the display is to appear on when called in the *Crt* position. A valid entry is from 0 to 8. Use 0 to cause the display call-up to occur on the current window. Use any other valid number to designate a call-up on a specific window.

AUTOMATIC DISPLAYS AND POP UPS

Any assembled display that resides on the console can be defined as an automatic. An automatic display will appear in place of the current display when activated. A pop up can also be defined as an automatic. An automatic pop up will appear on the current display at a specified location when activated. A set of DIGITAL tags are monitored to trigger the automatic activation of these displays or pop ups. This enables automatic activation based on process changes. Only a DIGITAL tag can trigger activation.

Automatic displays and pop ups are configured in sets. Up to 30 sets with up to 100 trigger tags and associated displays and pop ups per set are supported. Each set contains a list of DIGITAL tags. Each tag in the list has an assembled display (***DU***) or an assembled pop up symbol (***DL***) assigned to it. A pop up also requires an x,y coordinate to determine its position on the page when called.

Each set also has an assigned master display and termination display. The master display is called to activate the set. The termination display is called to deactivate the set. Refer to the discussion on automatic displays and pop up elements in the ***Operation*** instruction for further explanation on activating or deactivating a set (Table 1-2 lists instruction numbers).

The triggering of an automatic display or pop up occurs when a DIGITAL tag in the active set changes from its zero state to its one state. This requires, however, that there is no previously triggered display or pop up and the DIGITAL tag that triggered the last display or pop up has returned to its zero state.

A denotation symbol can be assigned to a set to identify a display or pop up as an automatic. To use a denotation symbol, a symbol file (***DL***) containing the denotation symbol must be created and assigned to a tag set during automatic display configuration. The denotation symbol does not appear for a display or pop up if the display or pop up is called manually.

The procedures required to create automatic displays or pop ups include:

- Creating a pop up element.
- Creating a denotation symbol.
- Configuring the tag set.

Configuring a tag set requires:

- Assigning a master display.
- Assigning a termination display.
- Assigning a denotation symbol.
- Setting the number of tags.
- Defining the tags that are part of the set and the displays or pop ups that they trigger.

Creating a Pop Up Element

A pop up element must first be created as a ***DT*** symbol source file using either the graphical display configuration (GDC) tool of the console configuration tools or the console elementary line editor. It then must be assembled into a ***DL*** symbol file by using the *Display Generator* function before it can be used in an automatic display or pop up group. Create this symbol in the normal way. The standard control and data acquisition elements provided with the console and the console configuration tools can be used as pop up elements. Refer to Section 7 for an explanation of the GDC tools and the elementary line editor.

When using automatic pop ups, it is important to reserve a portion of all displays for pop ups. This prevents a pop up from overwriting display elements. Normally, an area of a display is reserved for pop up elements activated by key select or touch point. This same area can be used for automatic pop ups.

The automatic displays function gives the ability to specify a substitution tag to be used for an automatic pop up element. A substitution tag is the tag that is to be substituted for any tag already defined in the symbol file. This allows using the same symbol source file for several pop ups instead of having to create a dedicated source file for each.

Automatic displays configuration sets an x,y coordinate that determines where the pop up is to appear. This coordinate in most cases, determines where the lower left corner of the pop up element will be positioned. In some cases, the actual position may not be the lower left corner of the element but instead a position offset from the corner. This depends on the reference (**rf**) command in the symbol source file. If the command is **rf 0,0** then the reference point for the symbol is the lower left corner. Any other setting for this command offsets the reference point from the corner position.

Creating a Denotation Symbol

Optionally, each tag set can be configured with a unique or the same denotation symbol. Whenever a display or pop up is triggered, this denotation symbol appears on the display at a specified location. The symbol allows easily recognizing that a display or pop up is an automatic. It can also be used to indicate to which tag set the display or pop up belongs.

The denotation symbol must first be created using either the GDC tool or elementary line editor as a ***DT*** symbol source file. It then must be assembled into a ***DL*** symbol file by using the *Display Generator* function before it can be used in an automatic display or pop up group. Create this symbol in the normal way. The symbol, however, must not contain any interactive escape commands (**ei**). Interactives are key selects and touch points. Refer to Section 7 for an explanation of the GDC tool and the elementary line editor. Refer to the ***Display Builder Reference*** instruction for an explanation of the graphic commands used to build displays and symbols (Table 1-2 lists instruction numbers).

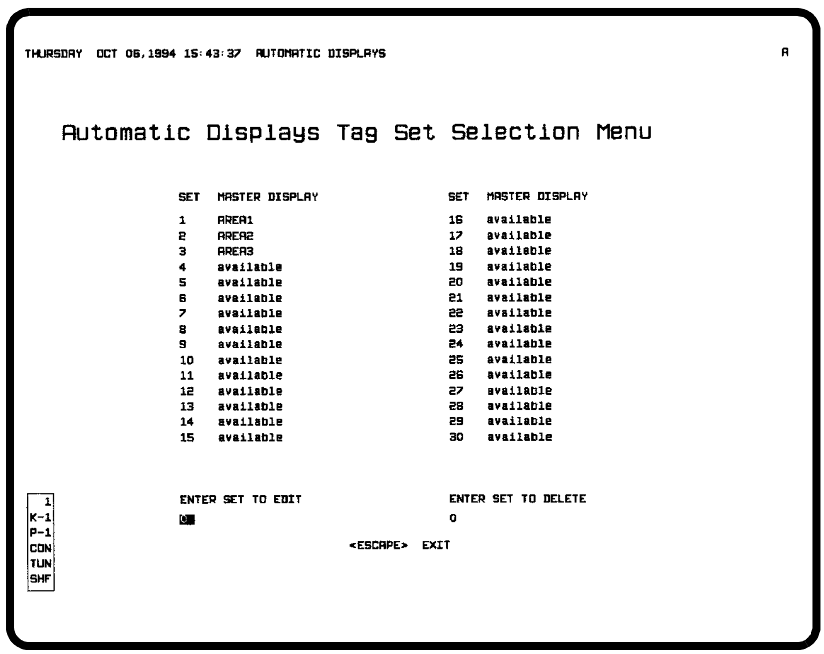
During configuration, the name of the denotation symbol and the x,y coordinate location where the symbol is to appear are defined. The name is the assembled display file name without its ***DL*** extension. The coordinate, in most cases, determines where the lower left corner of the symbol will be positioned. In some cases, the actual position may not be the lower left corner of the symbol but instead a position offset from the corner. Refer to ***Creating a Pop Up Element*** in this section for further explanation.

Example: To have a text string such as AUTO appear to indicate an automatic, the complete symbol would consist of the letters A-U-T-O and a color and size. Specify the name of the symbol (file name) as the *Denotation Symbol* during configuration. This causes the string *AUTO* to appear at its defined location when a tag in the active set triggers its display or pop up.

Defining a Tag Set

Use the *Automatic Displays* option to define automatic displays and pop ups. Figure 15-1 is the first page of the function. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Display*
→ *E Automatic Displays*



TPS0753B

Figure 15-1. Automatic Displays Page - Tag Set Selection

The tag set selection page enables defining a new, deleting an existing, or editing an existing tag set. Up to 30 tag sets can be defined, each set containing up to 100 trigger tags. Each tag has an associated display or pop up assigned.

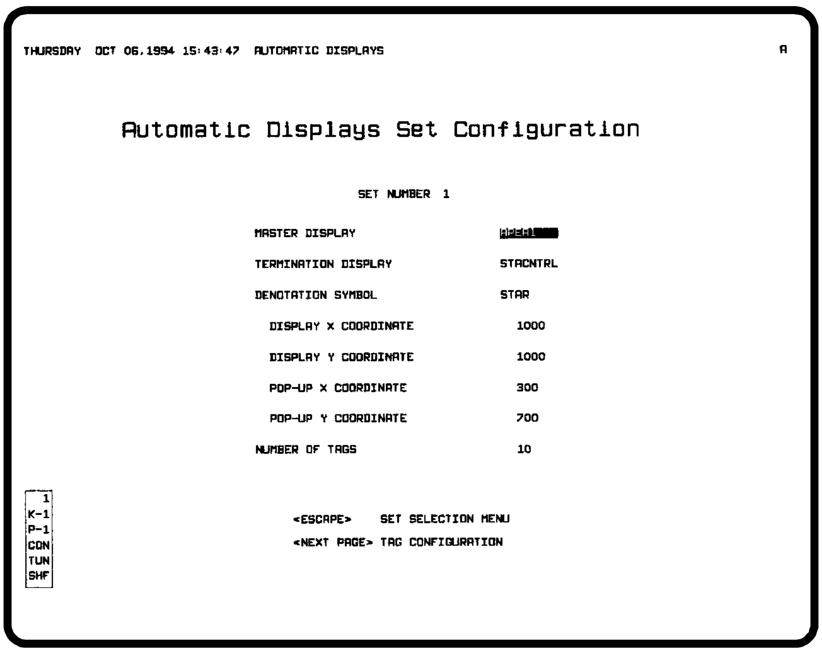
The first page identifies any previously defined tag sets and those sets that are still available for configuration. If a tag set has previously been defined, the name of a master display appears for that set; if not, *available* appears.

SELECTING A TAG SET

A tag set must be selected before it can be defined or edited. To select a tag set:

1. Enter the number of a tag set in the *ENTER SET TO EDIT* field. A valid entry is from 1 to 30. A previously defined set can be identified by the name of its master display appearing on the selection page.

2. Press **ENTER** to call the set. This calls a tag set configuration page (Figure 15-2). The configuration page defines the operating parameters for the selected set.



TPS0765B

Figure 15-2. Automatic Displays Page - Tag Set Configuration

DEFINING THE OPERATING PARAMETERS OF A TAG SET

The operating parameters that must be defined for a tag set include:

- Master display used to activate the set.
- Termination display used to deactivate the set.
- Denotation symbol to identify a display or pop up as an automatic. If the denotation symbol is unique, it can even identify the set that triggered the automatic display or pop up.

- Number of tags. This determines the maximum number of tags for the set.

When the master display of a set is called, the set becomes the active set. All tags that are part of an active tag set are monitored to trigger displays or pop ups. To define the operating parameters of the selected tag set:

1. Initially, the input cursor positions on the *MASTER DISPLAY* field (Figure 15-2). The field designates the display that the operator or the automatic display function must call to activate the tag set. Enter the name of an assembled display file (***DU***) without its extension. Any display that resides on the hard disk can be a master display.

2. The *TERMINATION DISPLAY* field designates the display that the operator or automatic display function must call to deactivate the tag set. Enter the name of an assembled display file (***DU***) without its extension. Any display that resides on the hard disk can be a termination display.

3. The *DENOTATION SYMBOL* field is optional. If a denotation symbol is to be used for this set, it must already have been created. Enter the name of the assembled symbol file (***DL***) without its extension. Refer to ***Creating a Denotation Symbol*** in this section for further explanation. Leave the field blank if not using a denotation symbol and skip to Step 5.

4. Define an x,y coordinate for the denotation symbol. The same denotation symbol is used for both displays and pop ups.

Display Enter an x-coordinate in the *DISPLAY X COORDINATE* field and a y-coordinate in the *DISPLAY Y COORDINATE* field if the tag set contains automatic displays. A valid entry is from 0,0 to 9999,7499; although, user space is from 400,400 to 9600,7200.

Pop Up Enter an x-coordinate in the *POP-UP X COORDINATE* field and a y-coordinate in the *POP-UP Y COORDINATE* field if the tag set contains automatic pop ups. A valid x-coordinate entry is from -9999 to 9999. A valid y-coordinate is from -7499 to 7499. The position is based on a reference point (**rf**) defined in the symbol file. Refer to ***Creating a Denotation Symbol*** in this section for further information.

5. Enter the number of tags that are to be part of this tag set in the *NUMBER OF TAGS* field. A valid entry is from 1 to 100.

6. Press **ENTER**.

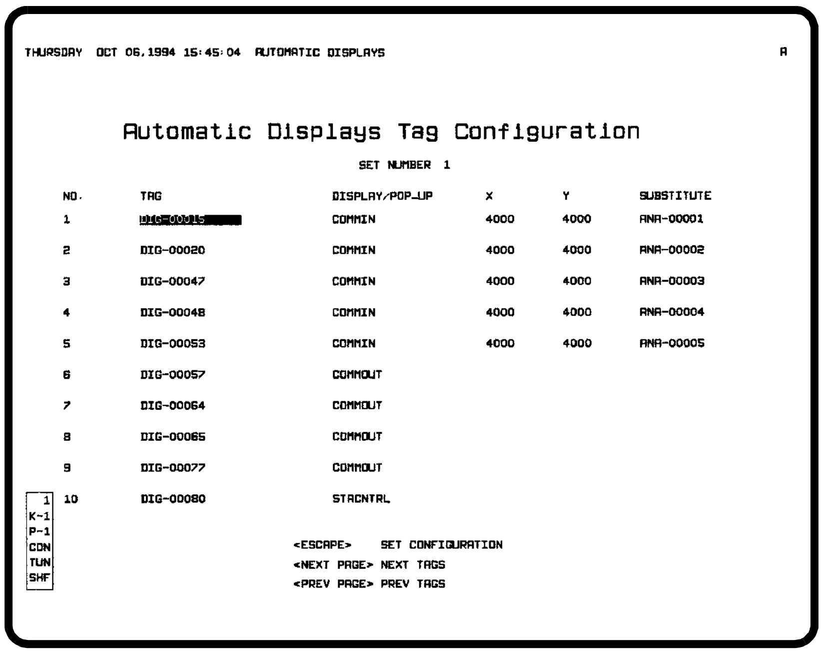
7. Press **NEXT PAGE** to call the tag configuration page. This is the page used to define the tags of the set and to define the

displays or pop ups they are to trigger. Follow the steps given in ***DEFINING THE TAGS OF A SET*** in this section.

DEFINING THE TAGS OF A SET

After the operation of a tag set has been defined, the tags that are to be monitored must be defined. Additionally, the display or pop up each tag in the set is to trigger must be identified.

Figure 15-3 shows the page that appears after pressing **NEXT PAGE** while on the tag set configuration page. This is the page used to define the list of tags for the set.



TPS0766B

Figure 15-3. Automatic Displays Page - Tag Configuration

Up to 100 tags can be defined. The tag configuration page shows only ten tag entries at a time. Press **NEXT PAGE** or **PREV PAGE** to sequence to the next or previous ten entries.

To define the fields of the page:

1. In the *TAG* column, enter the name or index number of a DIGITAL tag.

2. Enter the name of a display or a pop up element in the *DISPLAY/POP-UP* column. This is the display or pop up the tag is to trigger. The name must be of an assembled display without its ***DU*** extension or an assembled symbol without its ***DL*** extension.

3. If the tag triggers a pop up, enter an x,y coordinate in the *X* and *Y* column. This defines where the pop up will appear when triggered. A valid x-coordinate is from 0 to 9600. A valid y-coordinate is from 0 to 7200. If the tag triggers a display, do not enter any coordinates.

4. The *SUBSTITUTE* field is optional and is for a pop up element only. If the index number of the tag the pop up element is to present information for is already defined in the symbol file of the element, leave this field blank. Enter a tag name or an index number in this field if the pop up is to use a substitute tag in place of the tag already defined in the symbol. Refer to ***Creating a Pop Up Element*** in this section for further explanation.

5. Press **ENTER**.

Deleting a Tag Set

To delete a tag set:

1. Enter the number of a tag set to delete in the *ENTER SET TO DELETE* field on the tag set configuration page (Figure 15-1). A valid entry is from 1 to 30.

2. Press **ENTER** to initiate the deletion. An *available* replaces the display name after deletion.

MULTIWINDOW DISPLAYS

A multiwindow display function allows defining a list of displays that are to be called up on various windows simultaneously. After a list is defined, it can be assigned to a function key or ADP pushbutton for activation. Also, a key select or touch point built into a display can reference a multiwindow display list. Using this function, a single action can call multiple displays each appearing on a different window. Display call-ups can only be directed to the windows supported by the console.

To enable a multiwindow display call-up:

1. Define a display list.

2. Configure a function key or ADP pushbutton to use the list.

3. Create a display with a key select (**ei 108,113**) or touch point (**ei 108,113**) to use the list.

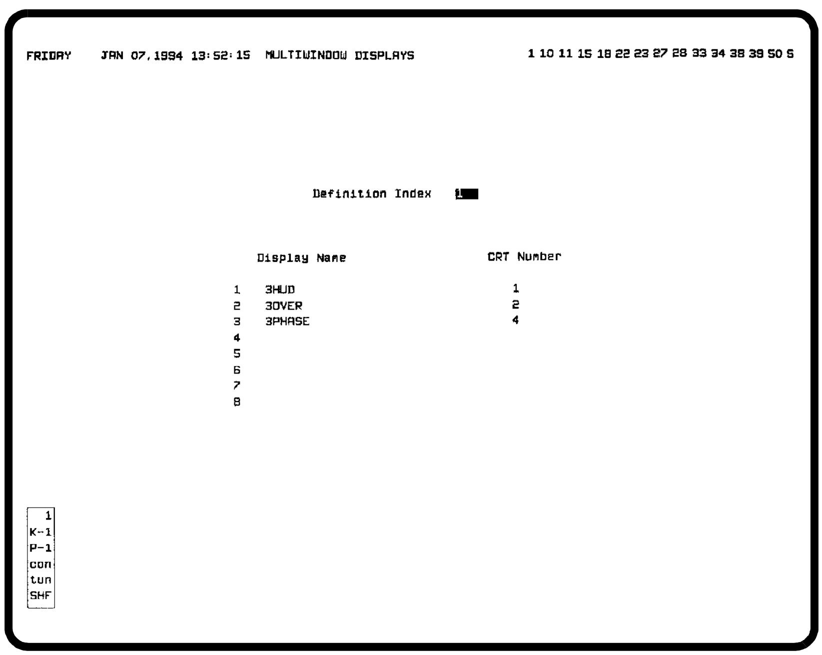
This section gives the procedures to define a display list and explains the requirements for assigning a list to a function key or ADP pushbutton. Refer to the ***Display Builder Reference*** instruction and the ***Console Configuration Utilities*** instruction for an explanation of key select and touch point display commands (Table 1-2 lists instruction numbers).

Defining a Display List

A multiwindow display list defines a list of up to eight displays and designates a target window for each display. Up to 256 display lists can be defined. When a list is activated, each display in the list is called to its target window.

Use the *MultiWindow Display Activation* option to define a display list for multiwindow call-ups. Figure 15-4 shows the page used. To choose the option, first press **GENL FCTNS MENU** then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Display*
→ *F MultiWindow Display Activation*



TPS1040A

Figure 15-4. Multiwindow Display Activation Page

To define or modify a display list:

1. Select a display list to edit.

   a. Enter the number of a list in the *Definition Index* field. A valid entry is from 1 to 256.

   b. Press **ENTER**.

2. In the *Display Name* column, enter the name of any assembled display (***DU***) without its extension. Up to eight names can be entered into the list.

3. For each display in the list, enter the number of a target window in the *CRT Number* column. A valid entry is from 1 to 8. Use a window number only once in a list.

4. Press **ENTER**.

Assigning to a Function Key or ADP Pushbutton

After a display list is defined, a function key or ADP pushbutton can be configured to use the list. To configure a key or pushbutton:

1. Follow the steps given in ***Making a Function Key Assignment*** in Section 8 to configure a function key. Follow the steps given in ***Making an ADP Pushbutton Assignment*** in Section 8 to configure an ADP pushbutton.

2. In either case, enter **L** in the first position of a key or pushbutton assignment.

3. Enter the index number of a display list in the second position of the assignment.

4. Leave the last field at its default.

SECTION 16 - TEXT DEFINITION AND SUBSTITUTION

INTRODUCTION

This section explains the functions used to define text and to make substitutions for system text. Text substitution can be used to replace default text that appears in displays and configuration pages and in logs. Text definition is used to define the text strings selectable by a text selector function block (FC 151) and by a remote motor control block (FC 136) in a PCU module. Also, alarm comment, logic state descriptor, and engineering unit descriptor configurations are text definition functions.

TEXT SUBSTITUTION

Most text that appears in displays can be modified by using the text substitution function. The text substitution function can change dynamic text, error messages, configuration specific entries, prompts, etc. With alternate language enabled, text substitution gives the ability to substitute alternate language characters for default English text.

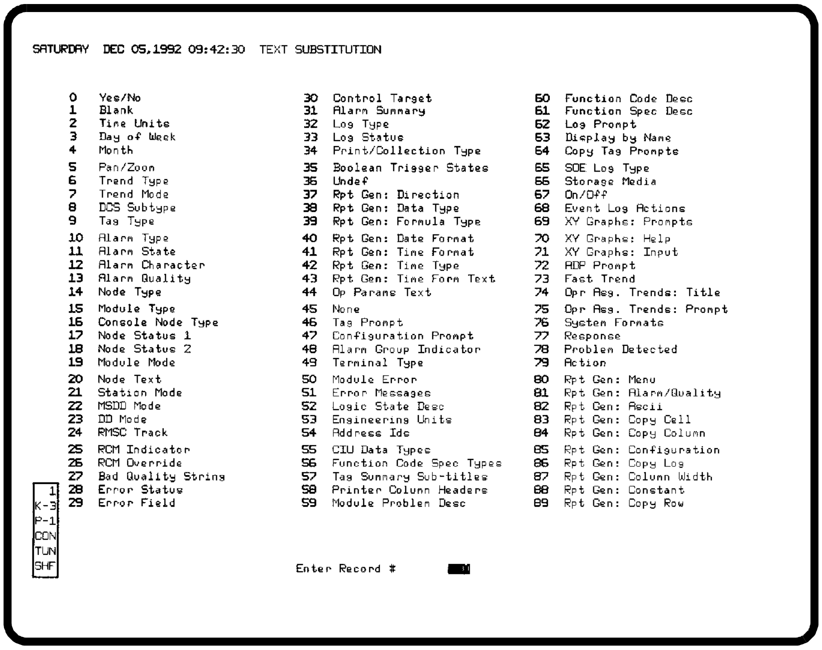
The console text that can be substituted are organized into configurable text sets. The text strings that make up a set are related. The substitution options of a set appear next to the *Language* field at the bottom of the page after calling the set: *DEFAULT*, *EXTENDED*, and *FORLANG*. If a text set cannot be modified, only *DEFAULT* appears next to the *Language* field. Other options are explained later in this section.

Not all text sets allow substitution with alternate language characters. If alternate language text is to be used, alternate language must be enabled first. Refer to Section 21 for additional requirements and for specific information about alternate language.

Use the *Text Substitution* option to substitute system text. Figure 16-1 shows the first page of this function. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *E Text Substitution*

The text substitution pages provide a complete list of text sets that can be substituted. The list extends over several pages.



TPS0160B

Figure 16-1. Text Substitution Page

Calling a Text Set

To call a text set:

1. Find a specific set in the list to modify. Press **NEXT PAGE** and **PREV PAGE** to move between the pages.

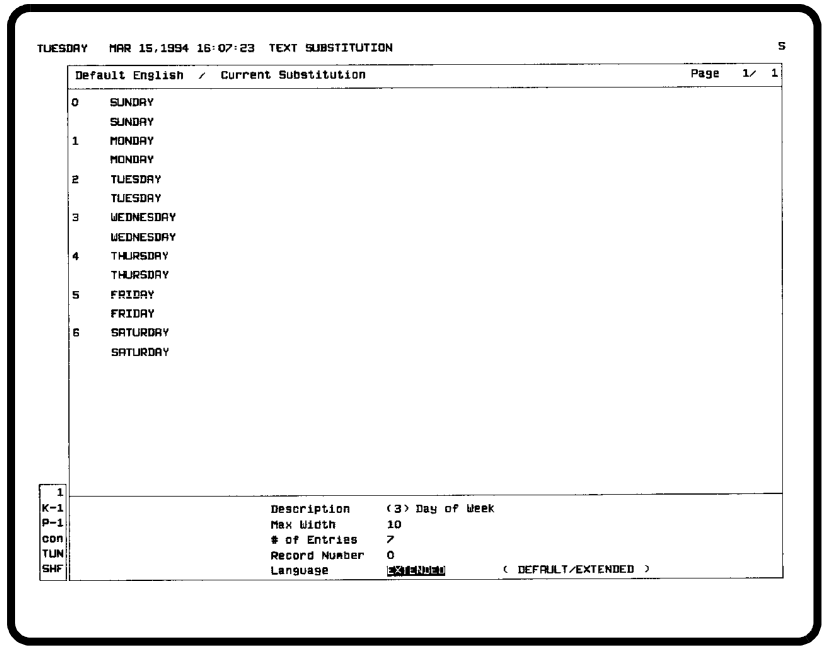
   NOTE: Do not make any changes to configurable text set *191 Window Network Transport*. The text in this set is used for window management.

2. Enter the record number of the set in the *Enter Record #* field. For example, enter **3** to modify the text for days of the week.

3. Press **ENTER**. The page that appears next lists the text strings for the selected text set. Refer to Figure 16-2 for an example.

For each text string, there are two entries. The default text displays in cyan, the substitute text in green. If no changes have been previously made to a text string, both entries are the same.

After calling a set, three informational fields and two input fields appear at the bottom of the page. The input fields are explained in the steps that follow. The informational fields and their purpose are:



TPS0055B

Figure 16-2. Example Configurable Text Set Page

Description - shows the name of the currently selected text set. The number in brackets ( ) is the text set number.

Max Width - indicates the maximum number of characters that can be entered for each text string of the set.

# of Entries - shows the number of text strings in the text set. The text strings in a set may extend over several pages.

Press **NEXT PAGE** and **PREV PAGE** to access the next or previous text set after calling a set. The *Description* field updates to show the currently selected text set.

Substituting Text

The *Language* field is used to put the page into editing mode. Depending on the text set selected, options for editing are FORLANG and EXTENDED. The FORLANG option will not appear unless alternate language was previously enabled and the console is in complex character mode. To define substitute text for a set:

1. Enter **EXTENDED** in the *Language* field to substitute default text with standard English characters or with extended alternate language characters. Enter **FORLANG** to substitute with alternate language complex characters.

2. In the *Record Number* field, enter the number of a specific text string to modify. A valid entry is from 0 to the number of records shown in the *# of Entries* field minus one.

Optionally, the *Record Number* field can be left at zero to position the input cursor on the first entry, then the configuration keys can be used to search for a specific entry.

If the number of entries is substantial, use the *Record Number* field to move the input cursor and scroll the page to a higher numbered entry. For example, if the text set has 100 entries, enter **50** to position the cursor and scroll the page to text string number 50. The cursor can then be moved up or down in the list.

3. Press **ENTER**. This positions the input cursor on the specified text string.

4. Key in the substitution text.

5. Press **ENTER**. Any changes are immediately implemented.

Resetting a Text Set

The *Language* field is used to reset text strings of a text set to defaults. To reset all text for a set to its original default text:

1. Enter **DEFAULT** in the *Language* field.

2. Press **ENTER**.

TAG TEXT SELECTOR

The *Tag Text Selector* option defines text strings that can be associated with status reported for DD, MSDD, and RMCB tags. These text strings can describe the good, bad, and waiting condition being reported by a device driver or multi-state device driver function block and the good, alarm, and waiting condition being reported by a remote motor control function block. The conditions are exception reported by a PCU module.

In the module, a text selector function block (FC 151) must be defined in the control configuration to reference a DD, MSDD, and RMCB function block. The text selector function block can also be configured as a stand-alone text selector.

Each text selector message defined in the console has a message number. Up to 10,000 messages can be defined. The text selector function block in a PCU module selects one of these messages by its number. The function block exception reports a message number and also a color and blink parameter. A TEXT tag must be configured in the console to receive this information.

If the message number received from the block is outside the range of messages defined in the console, error text appears. The *Error Field* text set in the text substitution function defines this error text. Color is still a function of the exception report in this case.

The text (message) can display in a graphic by using the text dynamic graphic command (**ed 79**). The graphic command does not specify any colors in its parameters. The color and blink specifications are received in an exception report generated by the function block. Color and blink configured in the PCU module should be consistent with the overall console color and blink settings.

Use the *Tag Text Selector* option to define text selector messages. Figure 16-3 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration

→ *A Database*

→ *F Tag Text Selector*

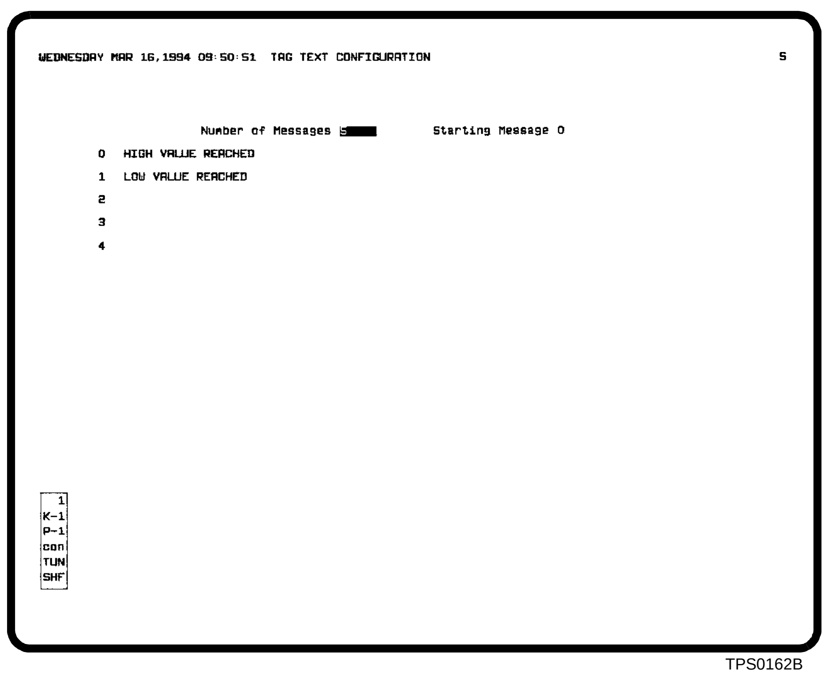


Figure 16-3. Tag Text Selector Page

Setting the Number of Messages

The *Number of Messages* field is used to adjust the maximum number of text selector messages.

NOTE: Care must be taken when changing the *Number of Messages* field since this field indirectly affects PCU module configurations. This number establishes the range of valid message numbers, which becomes the valid message numbers that can be set in the PCU module text selector function blocks. Decreasing this number erases any entries above the newly entered value, which could make message numbers in PCU modules invalid.

To set the maximum number of text selector messages:

1. In the *Number of Messages* field, enter a number for the total number of messages required. A valid entry is from 1 to 10000. Do ***not*** change this field if it is already set to the appropriate maximum.

2. If desired, enter a starting message.

3. Press **ENTER**.

Pressing **ENTER** both updates the configuration and allocates enough hard disk space for all required messages. It also calls 20 messages for display starting with the first message entered in the *Starting Message* field. The input cursor positions on this starting message after pressing **ENTER**.

Increase Size

If the total number of messages is changed from the current value, the file that contains the messages is resized. When the size is increased, the current messages copy to the same location in the file and the remaining new messages fill with blanks. During file update, the following message appears:

Update of Files in Progress, Please Wait

Do ***not*** continue until the message disappears.

Decrease Size

If the number of messages is decreased, the following prompt appears:

WARNING; ARE YOU SURE? ESCAPE IF NOT

Press **ESC** to cancel. Press **ENTER** to continue. After continuing, the file is reduced to its new size. All messages with index numbers equal to and less than the new number of messages are saved. All messages with index numbers greater than the new number of messages are deleted. During file update, the following message appears:

Update of Files in Progress, Please Wait

Do ***not*** continue until the message disappears.

Selecting a Starting Message

The tag text selector page can present only 20 text messages at a time. If the total number of messages is substantial, use the *Starting Message* field to move the input cursor and scroll the page to a higher numbered entry. For example, if the number of messages is 100, enter **50** to position the cursor and scroll the page to text message number 50. The cursor can then be moved up or down in the list.

To select a starting message:

1. Enter the number of a message in the *Starting Message* field. A valid entry is from 0 to the total number of messages minus one.

2. Press **ENTER**. The selected message then becomes the first message shown.

Optionally, the *Starting Message* field can be left at zero to call the first entry in the list, then the cursor can be moved to a specific entry.

Defining a Message

Up to 10,000 messages with up to 80 characters per message are supported. To define or edit a message:

1. Move to the message. Either move the cursor or:

    a. Press **ESC** to position the cursor in the *Starting Message* field.

    b. Enter the number of a message.

    c. Press **ENTER**.

2. Type in a message of up to 80 characters.

3. Press **ENTER**.

TEXT Tag Alarms

The text selector function block does not send alarm status to the console. Therefore, the only time a TEXT tag appears in alarm is when its status is bad quality. This, however, requires that the console be configured to use bad quality as one of its alarm conditions, which depends on alarm quality options configuration.

When the tag exhibits bad quality, either a bad quality text string replaces the currently displayed text string or the last known good string appears in the display. This also depends on alarm quality option configuration. The bad quality string can be modified through text substitution.

REMOTE MOTOR CONTROL TEXT

The *Remote Motor Control Text* option defines text sets that contain text strings for remote motor control function blocks (FC 136). The function block can report any of ten different error codes that identify its current status. 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 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.

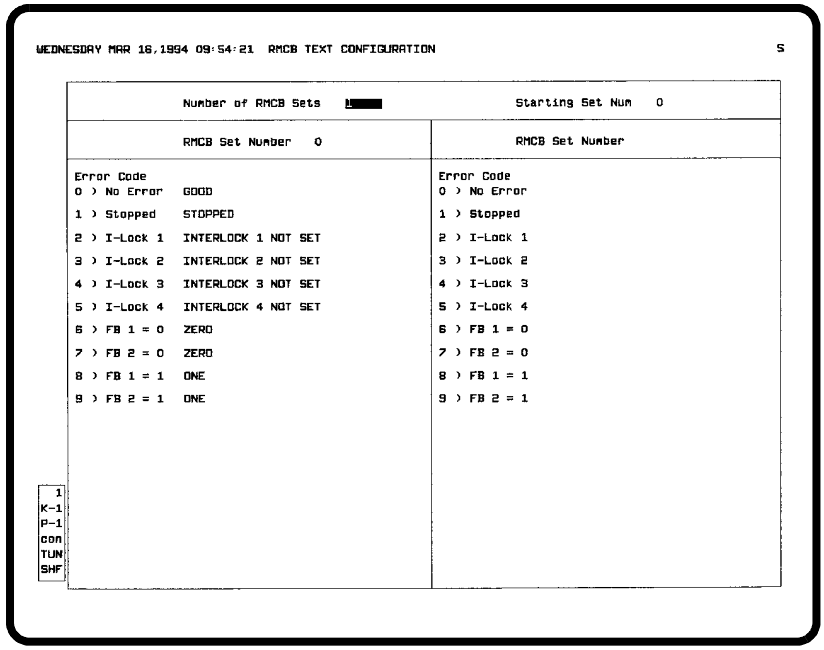
NOTE: The interlock codes indicate not set or logic zero conditions for each of four interlocks.

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.

A text message can appear in any display by using the **ed 92** graphic command. The standard RMCB faceplate element contains this command.

Use the *Remote Motor Control Text* option to define text strings for text sets. Figure 16-4 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *G Remote Motor Control Text*



TPS0163B

Figure 16-4. Remote Motor Control Text Page

Setting the Number of Text Sets

The *Number of RMCB Sets* field is used to adjust the maximum number of text sets.

> **NOTE:** Care must be taken when changing the *Number of RMCB Sets* field since this field directly affects the tag database. This number establishes the range of text sets, which becomes the valid range for an entry in the *Text Set* field of an RMCB tag. Decreasing this number erases any entries above the newly entered value, which could make message numbers in tags invalid.

To set the maximum number of text sets:

1. In the *Number of RMCB Sets* field, enter a number for the total number of text sets required. A valid entry is from 1 to 100. Do ***not*** change the field if it is already set to the appropriate maximum.

2. If desired, enter a starting set number.

3. Press **ENTER**.

Pressing **ENTER** both updates the configuration and allocates enough hard disk space for all required sets. It also calls two sets for display starting with the first set indicated in the *Starting Set Num* field. The input cursor positions on the first configurable error code.

Increase Size If the number of text sets is changed from the current value, the file that contains these sets is resized when **ENTER** is pressed. When the size is increased, the current sets copy to the same location in the file and the remaining new sets fill with blanks. During file update, the following message appears:

Update of Files in Progress, Please Wait

Do ***not*** continue until the message disappears.

Decrease Size If the number of text sets is decreased, the following prompt appears:

WARNING; ARE YOU SURE? ESCAPE IF NOT

Press **ESC** to cancel. Press **ENTER** to continue. After continuing, the file is reduced to its new size. All text sets with index numbers equal to and less than the new number of sets are saved. All text sets with index numbers greater than the new number of sets are deleted. During file update, the following message appears:

Update of Files in Progress, Please Wait

Do ***not*** continue until the message disappears.

Selecting a Starting Text Set

The remote motor control text page can show only two text sets at a time. If the number of sets is substantial, use the *Starting Set Num* field to move the input cursor and scroll the page to a higher numbered set. For example, if the number of sets is 100, enter **50** to position the cursor and scroll the page to text set number 50.

To select a starting set:

1. Enter the number of a text set in the *Starting Set Num* field. A valid entry is from 0 to the total number of sets minus one.

2. Press **ENTER**. The selected set then becomes the first set shown on the left side of the page.

Optionally, the *Starting Message* field can be left at zero to call the first set, then the cursor can be moved to a specific set.

Defining a Text Set

Up to 100 text sets each with ten RMCB error code messages are supported. To define or edit a text set:

1. Move to the text set. Either move the cursor or:

   a. Press **ESC** to position the cursor in the *Starting Set Num* field.

   b. Enter the number of a text set.

   c. Press **ENTER**.

2. Enter a text string of up to 15 characters for each error code. Both sets on the page can be edited at once.

3. Press **ENTER**.

ALARM COMMENT TEXT

An alarm comment is a text string that can be associated with an alarm condition of a process device (i.e., tag). The purpose of this text depends on how it applies to the device. It can be, for example, text that describes the alarm condition the device is currently in or text that describes any operator action required to correct the condition.

An alarm comment can be up to 64 characters. It can appear in an alarm summary or in any display. An alarm summary, however, must use a line format that is defined with an alarm comment field and a display must use an alarm comment escape command (**ec 33** or **ed 33**).

Refer to ***ALARM COMMENTS*** in Section 6 for further explanation of alarm comments and for alarm comment definition procedures.

ENGINEERING UNIT AND LOGIC STATE DESCRIPTOR TEXT

An engineering unit descriptor is a text string that shows the current unit of measurement for a process value. The descriptor is for an analog value only. Each descriptor defined in the console has an index number. A function block in a PCU module reports a value and also an engineering unit index number along with the value. This index number is used to select an engineering unit descriptor. The reported index number is cross referenced against the console defined list of engineering unit descriptors to present the appropriate text.

A logic state descriptor is a text string that describes the current operating state of a process device. The descriptor is for a digital state only. The method used to select the appropriate descriptor for a digital state is different than that used for an analog value. A PCU module does not send an index number along with an exception reported digital state. The tag database is referenced to determine which logic state descriptor to associate with a digital state. Each tag has a logic state descriptor defined for each of its possible states.

Refer to ***TAG DESCRIPTORS*** in Section 6 for further explanation and for the procedures to define descriptors.

SECTION 17 - XY PLOT DEFINITION

INTRODUCTION

This section explains XY plot definition. The XY plot function is used to present sets of process values in a two-dimensional graph. The data of a single XY plot consists of a coordinate pair whose x-axis value represents one process variable and y-axis value represents a different process variable. Each axis is bounded by a low and high limit.

Standard XY plot display templates are provided. Displays created with these templates must be modified to use plots defined through XY plot definition.

XY PLOT

XY plot definition is necessary to define the data source and operating parameters for a plot of an XY plot display. Each plot definition has an index number. An XY plot display uses this index number to select the data to display. The values plotted can be distributed trend values or exception reported tag values. The values can also come from a data file in an multi-function processor (MFP) module (i.e., MFC data source).

Enhanced Trend Support

The X-Y plot subsystem supports enhanced trends when the data source is trend. When both trends of the plot are enhanced trends the minimum sample period is one second. When one of the trends is an enhanced trend and the other is a distributed trend, the sample period must be a multiple of the distributed trend resolution.

Plot Sampling Constraints

There is a limit to the total number of XY plot samples that can be processed. A sample is a single piece of data. The limit is based on the total number of samples for all currently active plots. Any combination of plots can be active; however, the total samples for all plots cannot exceed 4,800 samples (maximum 1,440 samples per window). If all plots are 480 sample plots, then the maximum number of active plots is ten.

Table 17-1 shows some combinations that can be used that stay within the 4,800 sample limit. The elements can be combined in other ways. The number of plots per element is set during display creation as one of the XY plot escape command (**et 154**) parameters and is a maximum of five. XY plot definition sets the number of samples for a tag or trend plot. A C program providing data determines the number of samples for an MFC data source plot.

Table 17-1. Active Plot Sample Constraints

Element Size (%)	Elements per Window	Plots per Element	Samples per Plot	Number of Windows	Total per Window[1]	Total Samples[2]
100 × 100	1	5	120	4	600	2,400
100 × 100	1	3	480	3	1,440	4,320
100 × 50	2	5	120	4	1,200	4,800
50 × 50	4	5	60	4	1,200	4,800

NOTES:
1. Maximum 1,440.
2. Maximum 4,800.

Defining a Plot

A plot is identified by its index number. Up to 80 plots can be defined. A plot definition specifies the operating parameters for a single plot. During display creation, the index number is defined as one of the parameters in the commands used to build the plot display. The plot definition:

- Enables and disables data collection.
- Defines a data source.
- Sets plot operating parameters.

Use the *XY Plot* option to define a plot. Figure 17-1 shows the page used. To choose the option, first press **GENL FCTNS MENU** then select the following menu items in the sequence shown.

A OIS Configuration
→ *A Database*
→ *H XY Plot*

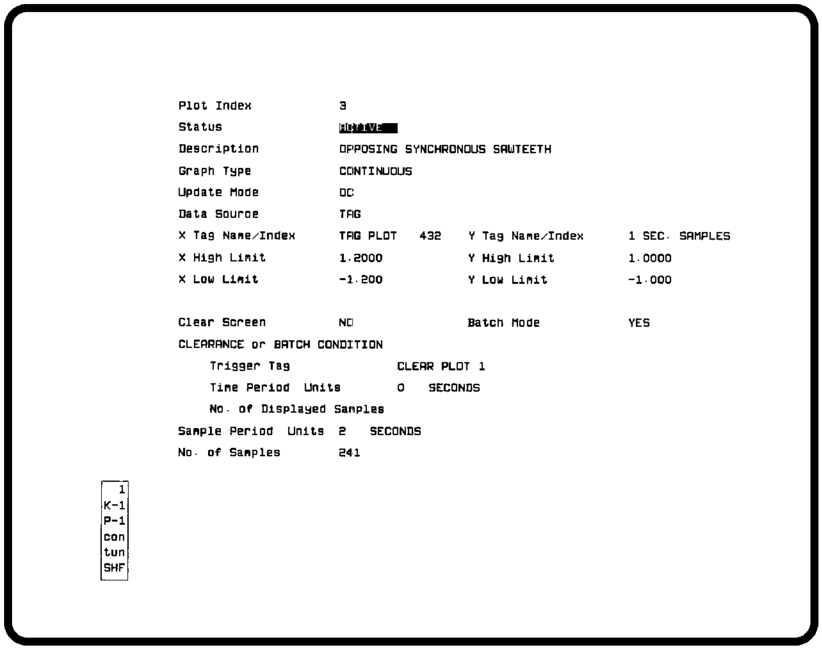
To define a plot:

1. Select a plot definition.

   a. Initially, the input cursor appears on the *Plot Index* field. Enter the number of a plot to define or edit. A valid entry is from 1 to 80.

   b. Press **ENTER**.

2. When the plot definition appears, the input cursor positions on the *Status* field. A plot can only be edited if it is inactive. Enter **INACTIVE** in this field if the plot is currently active.



TPS0021C

Figure 17-1. XY Plot Page

3. Some fields on this page remain the same for all XY plot definitions although they are defined differently. Others depend on the type of data source chosen.

Table 17-2 explains the fields that apply to all types of plots. Table 17-3 explains the additional fields for a tag data source plot. Table 17-4 explains the additional fields for a trend data source plot. Table 17-5 explains the additional fields for an MFC data source plot. Enter the appropriate data into each field of the page. Refer to these tables when entering data. Refer to ***MFC DATA SOURCE PLOT*** in this section for an explanation of the MFC data source.

4. Press **ENTER**.

Table 17-2. XY Plot Definition Page - General Fields

Field	Description
Plot index	Calls a plot definition and shows the index number of the plot being defined. A valid entry is from 1 to 80.
Status	Shows the current plot status and allows changing the status. A valid entry is: ACTIVE = activate. Data collection begins after the plot is activated. INACTIVE = deactivate. The plot must be inactive to make any changes.
Description	32-character description. This description is only used here. It does not appear elsewhere.
Graph type	CONTINUOUS = default; leave at default.

Table 17-2. XY Plot Definition Page - General Fields (continued)

Field	Description
Update mode	Method used to display collected data. Points plot from left to right. A valid entry is: SP = single point. Only one point per coordinate pair displays on the graph at a time. When new coordinates are received, the cross hair moves to indicate the position of the plotted point and the previous point is removed. The cross hair represents the most recent value plotted. MP = multipoint. This is the same as single point, except that any previous points appear as dots. The most recent value is represented by the cross hair. CC = continuous curve. This is the same as multipoint, except that the previous points are joined with a line. Points are joined from left to right. OC = operating curve. This is the same as continuous curve, except that the points are connected in the order that they are received. **NOTE:** Use CC mode for MFC data source plots.
Data source	Type of plot. This determines the data source for the plotted values. A valid entry is: TAG = tag plot using exception reported values. TREND = trend plot using distributed trend data. MFC = data comes from a file generated by a C language program running in a MFP module.

Table 17-3. XY Plot Definition Page - Tag Plot

Field	Description
X tag name/ index	Name or index number of the tag that provides the exception reported value for the x-axis variable.
X high limit	X-axis high scale value. Defaults to the full scale limit of the tag but can be changed.
X low limit	X-axis low scale value. Defaults to the zero scale limit of the tag but can be changed.
Y tag name/ index	Name or index number of the tag that provides the exception reported value for the y-axis variable.
Y high limit	Y-axis high scale value. Defaults to the full scale limit of the tag but can be changed.
Y low limit	Y-axis low scale value. Defaults to the zero scale limit of the tag but can be changed.
Clear screen	Configures plot clearing. Several options are available. For example, a plot can clear after a certain number of samples are plotted, after a defined time period expires, or when a trigger tag trips to its one state. A valid entry is: YES = enable automatic clearing. The *No. of Displayed Samples*, *Time Period Units*, and *Trigger Tag* fields must be defined. NO = disable automatic clearing.
No. of display samples	Number of samples to display before clearing the plot. A valid entry is: 2 to 480 = must be less than or equal to the number set in the *No. of Samples* field. For example, entering 120 clears the plot when the 121st sample is received. The 121st sample then plots as the first sample of the next plot. blank = disable this type of clearing.
Time period - units	Time period for plot clearing. The timer begins to decrement at the first sample. The time period is expressed as a count from 0 to 99 and a unit of SECONDS, MINUTES, HOURS, or DAYS. Leave these fields blank to not use this type of clearing.
Trigger tag	Name or index number of a DIGITAL or RCM tag that is to clear the plot when its state changes from zero to one. Leave this field blank to not use this type of clearing.

Table 17-3. XY Plot Definition Page - Tag Plot (continued)

Field	Description
Sample period units	Data collection resolution. The sample period is expressed as a count from 0 to 99 and a unit of SECONDS or MINUTES. Use a sample period that is a multiple of two seconds. The range is from 2 seconds to 30 minutes. Data collection begins when the plot is made active and continues at this sampling rate.
Number of samples	Maximum number of samples to be collected for the plot. A valid entry is from 2 to 480. For example, if a multipoint plot is being displayed and a value of 120 had been defined as the number of samples, a maximum of 120 samples would be plotted. The oldest sample scrolls off the display when a new sample is received after the maximum number of samples are on the display. Refer to Table 17-1 for guidelines when setting this field.

Table 17-4. XY Plot Definition Page - Trend Plot

Field	Description
X trend index	Index number of the trend definition that supplies the x-axis variables. A valid entry is from 1 to 10000.
X tag name	Non input field; tag associated with the x-axis trend. Blanks are shown if no tag is assigned to the trend.
X high limit	X-axis high scale value. The field defaults to the full scale limit of the tag specified in the *X Tag Name* field if one is assigned. Otherwise, the field defaults to zero. This field can be changed.
X low limit	X-axis low scale value. The field defaults to the zero scale limit of the tag specified in the *X Tag Name* field if one is assigned. Otherwise, the field defaults to zero. This field can be changed.
Y trend index	Index number of the trend definition that supplies the y-axis variables. A valid entry is from 1 to 10000.
Y tag name	Non input field; tag associated with the y-axis trend. Blanks are shown if no tag name is assigned to the trend.
Y high limit	Y-axis high scale value. The field defaults to the full scale limit of the tag specified in the *Y Tag Name* field if one is assigned. Otherwise, the field defaults to zero. This field can be changed.
Y low limit	Y-axis low scale value. The field defaults to the zero scale limit of the tag specified in the *Y Tag Name* field if one is assigned. Otherwise, the field defaults to zero. This field can be changed.
Clear screen	Configures plot clearing. Several options are available. For example, a plot can clear after a certain number of samples are plotted, after a defined time period expires, or when a trigger tag trips to its one state. A valid entry is: YES = enable automatic clearing. The *No. of Displayed Samples*, *Time Period Units*, and *Trigger Tag* fields must be defined. NO = disable automatic clearing.
Batch mode	Configures plot triggering and time duration. For example, When a tag is triggered, the plot is cleared and collection of data starts. After data is collected for the specified duration, data collection stops. A valid entry is: YES = enable automatic batch mode. The *Trigger Tag* must be defined and either or both the *Time Period Units* and *No. of Displayed Samples* fields must be entered. NO = disable batch mode

Table 17-4. XY Plot Definition Page - Trend Plot (continued)

Field	Description
No. of display samples	Number of samples to display before clearing the plot. A valid entry is: 2 to 480 = must be less than or equal to the number set in the *No. of Samples* field. For example, entering 120 clears the plot when the 121st sample is received. The 121st sample then plots as the first sample of the next plot. Leave these fields blank to not use this type of clearing.
Time period - units	Time period for plot clearing. The timer begins to decrement at the first sample. The time period is expressed as a count from 0 to 99 and a unit of SECONDS, MINUTES, HOURS, or DAYS. Leave these fields blank to not use this type of clearing.
Trigger tag	Name or index number of a DIGITAL or RCM tag that is to clear the plot when its state changes from zero to one. Leave this field blank to not use this type of clearing.
Sample period units	Data collection resolution. Data collection begins when the plot is made active and continues at this sampling rate. This field requires two entries: Count and units. The sample period must be based on the resolution of the trends specified in the *X Trend Index* and *Y Trend Index* fields. A trend can have a resolution of one minute or 15 seconds. Enter a period that is equal to or a multiple of the greater of the two resolutions. For example, if the x-trend has a 15-second resolution and the y-trend has a one-minute resolution, the period and units can be one minute to 30 minutes. If both trends are 15-second trends, the period and units can be 15 seconds, 30 seconds ... 30 minutes.
Number of samples	Maximum number of samples to be collected for the plot. A valid entry is from 2 to 480. For example, if a multipoint plot is being displayed and a value of 120 had been defined as the number of samples, a maximum of 120 samples would be plotted. The oldest sample scrolls off the display when a new sample is received after the maximum number of samples are on the display. Refer to Table 17-1 for guidelines when setting this field.

Table 17-5. XY Plot Definition Page - MFC Data Source Plot

Field	Description
MFC file ID	Each MFC data source file is identified in an MFP module by a number from 1 to 32,759; however, the console only accepts a number from 1 to 9999. Enter the number that identifies the file used for this plot.
MFC address	Loop, PCU, and module number of the MFP module providing the MFC data source file.
Trigger tag	Tag that is set by the MFP module to notify the XY plot task that a data file is ready for collection. This must be an analog exception reporting block (i.e., FC 30) defined as an ANALOG tag in the database. Refer to ***MFC DATA SOURCE PLOT*** in this section for further explanation. The C program must set the analog exception report block to the size (in bytes) of the data file. The value must also be greater than the high alarm condition of the block. The ANALOG tag high alarm condition triggers the XY plot task.
X high limit	X-axis high scale value.
X low limit	X-axis low scale value.
Y high limit	Y-axis high scale value.
Y low limit	Y-axis low scale value.
Border color	Color for the border of the XY plot. A valid entry is from 0 to 63 (default BLUE).
Reference line color[1,2]	Color of a reference line. Each reference line is displayed as a solid line. Specify a color for those x-lines and y-lines that are used. A valid entry is from 0 to 63 (default CYAN).

Table 17-5. XY Plot Definition Page - MFC Data Source Plot (continued)

Field	Description
X/Y axis legend plot	24-character legend displayed on the XY plot.

NOTES:

1. A reference line is disabled if its value is either greater than the high limit or less than the low limit. Reference line values are defined in the MFC data file. High and low limits are defined on the XY plot interactive display.
2. A single plot can have only five reference lines. If the plot does not use a reference line, the color is ignored.

MFC DATA SOURCE PLOT

For an MFC data source plot, typically a C program running in an MFP module writes data to a file which the console then reads and plots. Function blocks set up in the MFP module trigger the transfer of data from the module to the console.

Refer to Appendix F for the required MFC data source file structures. Refer to the ***C Utility Program*** instruction for information about the C utility program (CUP). The instruction explains the function calls used to write information to the module and the procedures to load a C program into an MFP module (Table 1-2 lists instruction numbers).

Interface Logic

Figure 17-2 shows the interface logic required to trigger a data file read by the console. The function blocks provide the interface between a C program executing in the module and the console. Define the blocks as described in Table 17-6.

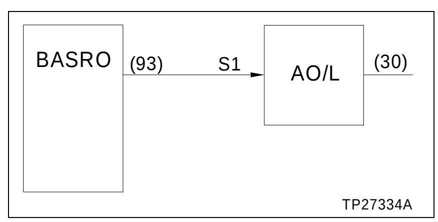


Figure 17-2. Interface Logic

Table 17-6. Interface Logic Specifications

Function Code	Specification	
FC 30	S1	Address of BASIC real output block
	S2 to S4	Default
	S5	1.0
	S6	-1.0
	S7	Default
FC 93	S1	Default

Data Transfer

Figures 17-3 and 17-4 show the data flow between the C program and the console. Figure 17-3 depicts a set trigger condition and Figure 17-4 depicts a reset trigger condition.

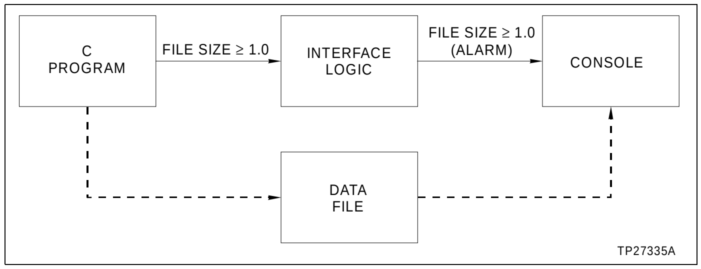


Figure 17-3. Data Transfer - Set Condition

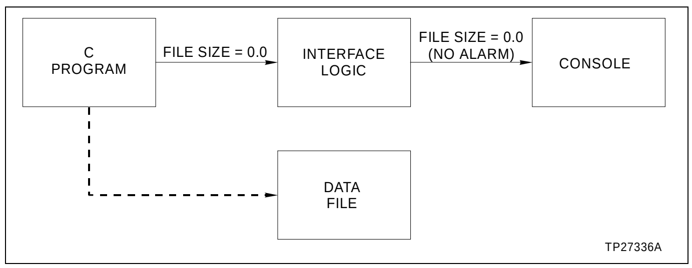


Figure 17-4. Data Transfer - Reset Condition

Set Condition

After writing data to a file, the C program writes the size of the data file to the BASIC real output block (FC 93) of the interface logic. The BASIC real output block outputs this file size to the analog exception report (FC 30) block (Figure 17-2). A file size greater than the high alarm limit set in the analog exception report block triggers a high alarm condition. The console interprets the high alarm condition as an indication to read the data file. The plot definition identifies the module and file (MFC file ID) the console is to read.

For this data transfer to work, an ANALOG tag is required in the console to monitor the analog exception report block. This tag is identified in a plot definition as the trigger tag. The trigger tag is considered set when the value exceeds the high alarm limit (≥1.0) and the last value was a no alarm condition. An ANALOG tag is used instead of a digital type of tag to provide a means of sending the size (in bytes) of the data file at the same time the trigger is provided. The floating point value set in the analog exception report block is rounded and converted to an integer.

Reset Condition

The C program must return the analog exception report block to a no alarm condition (0.0) to enable the next data file transfer. The time in seconds should be no less than approximately:

$$5 \times numfiles$$

where:

numfiles	Total number of files that use the trigger tag.

NOTE: For proper XY plot operation, the tag database must contain a system status tag (i.e., N90STA) for this console that specifies its own address.

Guidelines

To limit console loading and to provide reasonable XY plot response time, the following configuration restrictions are recommended when using the MFC data source for XY plots:

- One plot per plot element.
- Maximum of three plots per window.
- Maximum of six plot elements on all windows at one time.
- Maximum of 480 data points per plot.
- Only ten plots active at one time.

PLOT DISPLAY REQUIREMENTS

Edit the source file (***DT***) of an XY plot display to use a plot definition. Enter the index number of a plot definition as one of the parameters in the display escape commands. These include, for example, the XY plot (**et 154**), plot control (**ei 107** and **ei 108**), and plot information (**su 154**) escape commands. Refer to the ***Display Builder Reference*** instruction for a description of the parameters for these commands (Table 1-2 lists instruction numbers).

NOTE: Standard XY plot display templates (***XYPFL5.DT*** and ***XY1A.DT***) are provided with the console. When creating a display with one of these templates, the display file being used should be copied and renamed first. The new display can be called or assigned by its name.

The elementary line editor of the console or the graphical display configuration (GDC) of the console configuration tools can be used to edit an XY plot display. After editing, the ***DT*** source file must be assembled into a ***DU*** display file by using the *Display Generator* function. The display can then be called for viewing by name or assigned to a keyboard key or ADP pushbutton.

Refer to Section 7 for an explanation of the *Display Generator* function, console configuration utilities program, and elementary line editor. Refer to Section 8 for procedures to assign a display to a function key or ADP pushbutton.

SECTION 18 - TREND PEN

INTRODUCTION

This section explains the trend pen cluster function. The trend pen cluster function provides an interface between the console and trend pen recorders or indicator stations. It can also be set up for generic use as an output transfer function. Trend pen refers to some type of pen or chart recorder. Indicator station refers to a device that displays values through bar indicators or digital readouts.

The trend pen cluster function enables assigning trend pen recorders or indicator stations to track a set of process variables selected from the console. The process variables being tracked can be changed at any time. The limits for each are:

- 64 trend pen recorders with up to 32 pens per recorder.
- 16 indicator stations with up to eight bars or indicators per station.
- 32 output transfer groups with up to 32 transfer units per group.

The steps required to configure a trend pen, indicator station, or transfer group include:

1. Module configuration.
    a. Adding the interface and output logic.
    b. Adding a configure block.
2. Console configuration.
    a. Defining tags.
    b. Configuring the trend pen cluster function.

Separate options for trend pen, indicator station, and output transfer are provided to distinguish between devices. A separate option for each is provided for identification purposes only. This section explains trend pen cluster configuration as if setting up a trend pen recorder. The procedures to set up an indicator station or transfer group are essentially the same as setting up a recorder.

MODULE REQUIREMENTS

The trend pen cluster function requires a dedicated PCU module. Pen assignment changes made from the console place the module in ***configure*** mode and can cause problems if the module is configured to perform other functions. The module driving the pen recorder must be either a multi-function processor (MFP) or multi-function controller (MFC) module. Along with the MFP or MFC module, some type of I/O module is needed that can provide an analog output to drive a pen such as an analog output (AOM) module for module bus or an analog output (ASO) module for Controlway or module bus.

The MFP or MFC module sends values to the I/O module providing the analog output to the pen. The I/O module converts these values to voltages that drive the physical pens connected to it. Normally, the I/O module holds its outputs until the MFP or MFC module sends other values. When the MFP or MFC module is put into configure mode by using the trend pen cluster function, the console sets each pen controlled by the module to its off position. The I/O module continues to drive the pens with the last signal received, which is off while in configure mode, until the module is put into execute mode and each pen is successively turned back on.

Trend pen recorders can be grouped into one trend pen module or a different module can be used for each recorder. The number of recorders a single module can control depends on the I/O capability of the module and the number of pens for each recorder. All recorders that a single module controls are affected when the module mode is changed during configuration changes or pen reassignments.

MODULE CONFIGURATION

The console requires the proper configuration in the PCU module to drive the pens of a trend pen recorder. The PCU configuration that drives a single pen of a recorder consists of three parts: Input logic, interface logic, and output logic. Each pen of a recorder must be set up with its own set of logic. For example, if the recorder drives eight pens then eight logic sets must be configured.

Figures 18-1 and 18-2 show the PCU module configurations required to set up the logic for a pen. The console automatically sets up and downloads the proper input logic during trend pen cluster configuration. The interface logic and output logic must be configured prior to trend pen cluster configuration.

The function blocks that make up the input logic and interface logic must reside in eight consecutive block addresses in the module (i.e., n to n+7). The n block of the input and interface

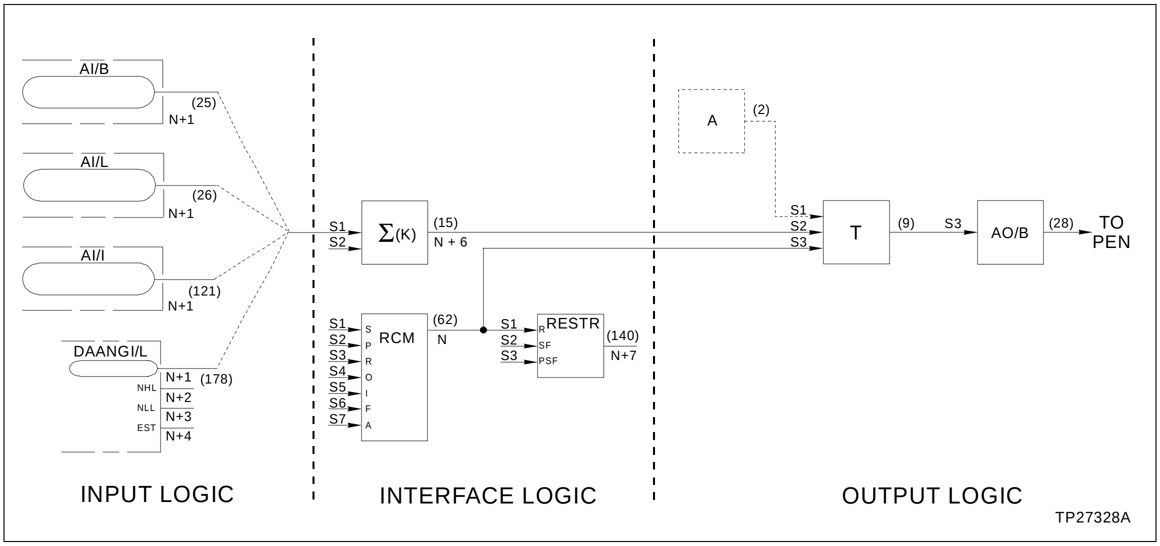


Figure 18-1. Analog Configuration

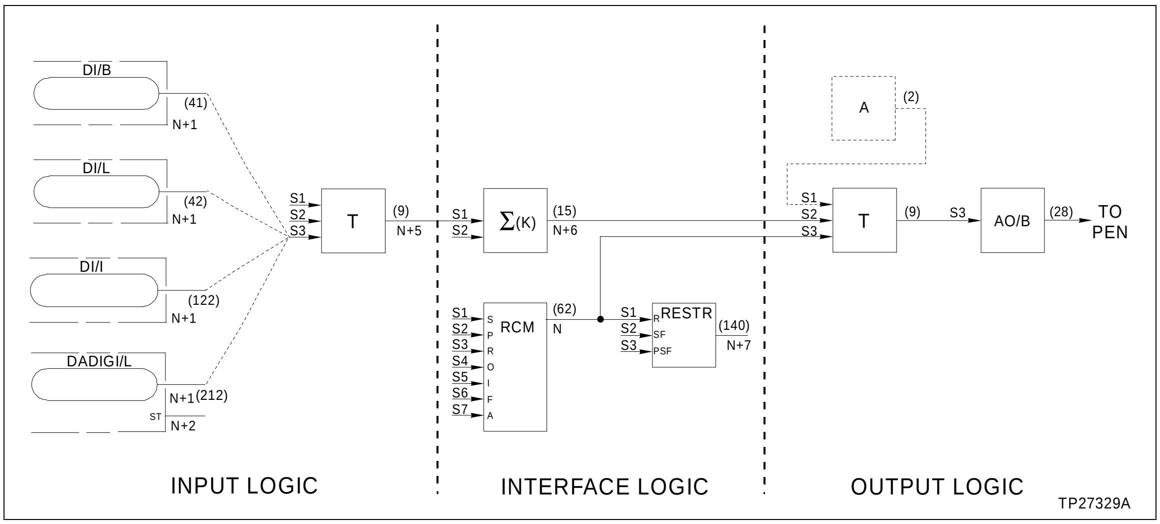


Figure 18-2. Digital Configuration

logic can be located in any valid configurable block address in the module. This is also true for the output logic.

> **NOTE:** The console requires that the function blocks for a pen reside in eight consecutive block addresses (i.e., n to n+7). It is recommended that ten consecutive block addresses be reserved for each pen (i.e., n to n+9).

Additionally, the module must contain an RCM block the console uses to put the module into configure and execute mode. This block is referenced by an RCM tag defined in the console. The tag is referred to as the configure tag. Refer to ***Configure Block*** and ***CONFIGURE TAG*** in this section for further explanation of the configure tag.

Restrictions

All inputs for the trend pen logic must come from outside the MFP or MFC module dedicated to controlling a recorder. Do not use a module dedicated to a trend pen recorder to acquire analog or digital inputs from process devices. The trend pen cluster function expects the source for a pen to be outside of this module. All blocks used in the input logic acquire values from outside the module.

Interface Logic

The interface logic for a pen consists of three function blocks: Remote control memory (FC 62) block, two-input summer (FC 15) block, and restore (FC 140) block (Figures 18-1 and 18-2).

Block n

The RCM block (block n) allows turning the pen on or off. Its output selects the input to the transfer block in the output logic. A pen can be turned on or off from within the trend pen cluster function.

Block n+6

The summer block (block n+6) passes the input signal from the input logic to the output logic. The summer block allows scaling the input before it is output to the pen. The input is scaled by changing the zero and span for a pen from within the trend pen cluster function.

Block n+7

The restore block (block n+7) is used to remember and restore the state of the RCM block when the module changes mode. This insures that the pen returns to the same state it was in when the module returns to execute mode from configure mode. The trend pen cluster function sets all pens to off when it puts the module into configure mode.

NOTE: If an existing configuration is set up with the interface logic blocks in addresses n, n+3 and n+4, it can still be used without modification. The only limitation is that it cannot reference a DANG or DADIG tag. To use these tags, set up the interface logic blocks in addresses n, n+6 and n+7.

For each pen, define the interface logic function blocks as described in Table 18-1.

Table 18-1. Interface Logic

Block Address	Function Code	Spec	Value
n	FC 62	S1 to S8	Default
n+6	FC 15	S1 to S4	Default[1]
n+7	FC 140	S1	Address of block n
		S2 and S3	1
		S4 to S9	Default

NOTE:

1. Modified by trend pen cluster function.

Input Logic

The trend pen cluster function automatically configures the PCU module with the proper input logic based on the function block (tag) selected as the input source for a pen. The function block in the input logic is the n+1 block. Refer to Figures 18-1 and 18-2. Only one of the input blocks shown in these figures is used for a pen and depends on where the input comes from and its type.

Analog If the input to a pen is an analog signal, the console adds one of the following function blocks to the PCU module:

- Analog input (same PCU node), FC 25.
- Analog input (different PCU node), FC 26.
- Analog input/INFI-NET, FC 121.
- Data acquisition analog input/loop, FC 178.

Digital If the input to a pen is a digital signal, the console adds one of the following function blocks to the PCU module:

- Digital input/Controlway/module bus, FC 41.
- Digital input/loop, FC 42.
- Digital input/INFI-NET, FC 122.
- Data acquisition digital input/loop, FC 212.

Additionally, the console adds an analog transfer block (FC 9) for a digital input (Figure 18-2). This is block n+5. The transfer block converts the digital input to an analog output to the pen.

Output Logic

The output logic depends on the type of I/O module providing the analog output to drive a pen or an indicator station. It depends on the type of output from an output transfer group. Figures 18-1 and 18-2 show an analog output (FC 28) block in the output logic. This block transfers a signal to an IMAOM01 Analog Output Module (module bus only). Another option is to use an analog output (FC 149) block to transfer a signal to an IMASO01 Analog Output Module (Controlway or module bus).

In any case, an analog transfer (FC 9) block must be part of the output logic. This function block selects one of two inputs depending on its state. It selects the S2 input which comes from the summer block (n+6) when the pen is on and the S1 input when the pen is off.

For each pen, define the output logic function blocks as described in Table 18-2.

Table 18-2. Output Logic - Summer Block

Function Code	Spec	Value
FC 9	S1	5
	S2	Address of block n+6
	S3	Address of block n
	S4 and S5	Default

Optional The S1 input to the summer block can be used as shown in Figures 18-1 and 18-2 to drive the pen to some predetermined value. If this is desired, configure a manual set constant (FC 2) block. Then use its address as the summer block S1 specification. Tune the block output to the desired rest position.

Configure Block

An RCM block in the PCU module is required to control the module mode during configuration. The console automatically puts the module into configure mode when sending an entire configuration or changes to a configuration to the module. After the changes have been made to the module configuration, the module is then put into execute mode. The block output is logic one when the module is in execute mode and logic zero when in configure mode.

The configure block also triggers any user-configured logic to drive the pens to a predetermined condition. This requires setting up additional logic during initial module configuration.

Only one configure block is required in a module even if several recorders are controlled by the module. If desired, more than one configure block can be defined to allow associating a specific configure block with a specific recorder. The association is made during trend pen cluster configuration. Define the function block as described in Table 18-3.

Table 18-3. Configure Block

Function Code	Spec	Value
FC 62	S1 to S4	Default
	S5	1
	S6 to S8	Default

Optional The feedback signal to the configure block determines when the module changes mode. If desired, a delay can be built into the feedback to insure that the logic has time to position the pens to their off position before the module goes into configure mode. Use two not (FC 33) blocks and a timer (FC 35) block to create the delay logic as shown in Figure 18-3. Define the function blocks as described in Table 18-4.

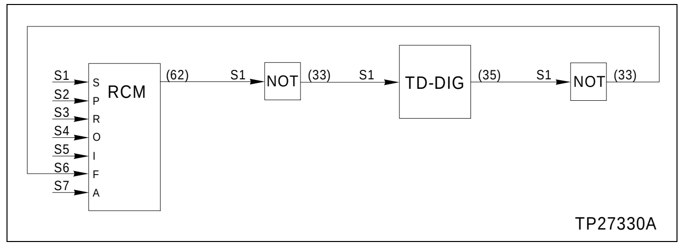


Figure 18-3. Configure Block with Delay

Table 18-4. Configure Control Block with Delay

Function Code	Spec	Value
FC 33 (1st)	S1	Address of RCM block (configure block)
FC 33 (2nd)	S1	Address of timer block
FC 35	S1	Address of 1st not block
	S2	1
	S3	Delay in seconds from 1 to 10. Do not delay more than ten seconds
FC 62	S1 to S4	Default
	S5	1
	S6	Address of 2nd not block
	S7 and S8	Default

Do not create a delay of more than ten seconds. One second is recommended. The trend pen cluster function waits a minimum of ten seconds but not more than 15 seconds for the RCM feedback signal to go to logic zero when changing to configure mode. When using the *Change mode to configure* option, the following error message appears if after 15 seconds the feedback is still a logic one:

No response to the Requested Action

In this case, however, the module is still changed to configure mode.

When using the *Submit edits made* and *Submit entire configuration* options, any download attempt is canceled and the same error message is presented if after 15 seconds the feedback is still a logic one. The difference here is that the module remains in execute mode assuming it was in execute mode when the attempt was made.

CONSOLE CONFIGURATION

On the console, both tag configuration and trend pen cluster configuration must be performed to enable trend pens. Module configuration must be completed before configuring the console. Refer to ***MODULE CONFIGURATION*** in this section for module configuration requirements. Tag configuration must be completed before trend pen cluster configuration.

The trend pen cluster function has three configuration pages: General parameters, device definition, and device assignment. Configure these pages in this order.

NOTE: It may be beneficial to dedicate a single console to controlling all or a specific set of trend pen recorders, indicator stations, or output transfer groups to prevent confusion and conflicts. This is ***not*** a requirement, however. Any console, after being set up, can reconfigure and reassign pens, stations, or transfer groups. The changes made on a console, however, are not broadcast to all consoles. Therefore, any changes to the module configuration caused by a reassignment or configuration change are only reflected on the console on which the changes were made.

Tag Configuration

The trend pen cluster function requires tags to be configured in the console database for three different purposes:

- To access the configure block. The block is used to automatically put the PCU module in configure mode to download either an entire configuration or changes to an existing configuration. It is also used to put the module in execute mode after downloading the configuration changes.

- To access the RCM block (block n) associated with a pen. This allows changing the state of the RCM block from the console to turn a pen on or off.

- To access and assign a process variable to a pen. In most cases, this tag will already have been defined and used in other functions.

Follow the steps given in ***Defining a Tag*** in Section 6 to define a tag.

CONFIGURE TAG

The configure tag is the RCM tag that references the configure block in the PCU module. Refer to ***Configure Block*** in this section for more information about this block.

Define an RCM type of tag as the configure tag of a recorder. Make sure the tag description is meaningful since this description appears in the trend pen cluster function to identify the configure tag for the module (or recorder depending on module configuration).

PEN RCM TAG

An RCM block (block n) is part of the interface logic for a pen. Refer to ***Interface Logic*** in this section for information about the block. An RCM tag that references this block must be defined for each pen. The tag gives the console access to the block for turning a pen on and off. It also contains information used to identify the pen in the trend pen cluster function.

Define an RCM type of tag for each pen of a recorder. Make sure the tag description is meaningful since this description appears in the trend pen cluster function to identify the pen. Specifically, the first four characters of the description are used to differentiate between pens while on the device assignment page.

PROCESS VARIABLE TAG

The process variable tag is used to assign a process variable to a pen. Any analog or digital type of tag in the database can be assigned to a pen. This tag is normally already defined. If not, define a tag for each process variable that is to be assigned.

Trend Pen Cluster Configuration

Use the *Trend Pen Cluster* option to configure a trend pen recorder (or indicator station and output transfer group). To call the main menu of the function, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

B OIS Utilities
→ *I Trend Pen Cluster*

After choosing one of the options from the main menu, a submenu of options appears:

A Trend Pen Recorder
B Indicator Station
C Output Transfer

The remainder of this section explains the *General Parameters*, *Device Definition*, and *Device Assignment* options assuming option *A Trend Pen Recorder* is chosen from the submenu. The *B Indicator Station* and *C Output Transfer* options essentially work the same.

Configuration changes such as device definition, pen assignment, and pen reassignment can only be made when the module is in ***configure*** mode. The trend pen cluster function handles changing the module mode and downloading changes to the module.

GENERAL PARAMETERS

The general parameters page:

- Sets the number of recorders.
- Identifies the configure tag for a recorder.
- Defines a recorder description.
- Sets the number of pens for a recorder.
- Determines key lock requirements.

Figure 18-4 shows the general parameters page. From the main menu select the following menu items in the sequence shown to call the page.

A General Parameters

→ *A Trend Pen Recorder*

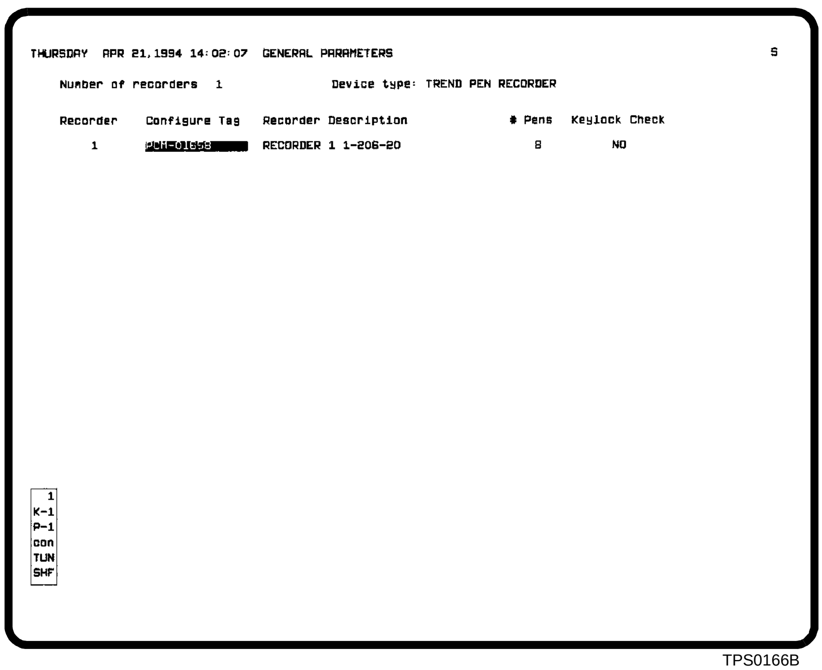


Figure 18-4. Trend Pen Cluster - General Parameters Option

The fields at the top of the page indicate:

Number of recorders - maximum number of recorders that can currently be configured. The number of recorders can be increased at any time to accommodate up to 64 recorders (16 indicator stations and 32 output transfer groups).

Device type: - type of device being configured. This field defaults to *TREND PEN RECORDER* after calling the page (also *INDICATOR STATION* or *OUTPUT TRANSFER*). The field cannot be changed.

Setting the Number of Recorders

If the number of recorders is currently set to zero, a *Number of recorders* field appears when the general parameters page is called. To set the number of recorders:

1. Enter a number from 1 to 64 (16 for indicator stations and 32 for output transfer groups). This number can be greater than the actual number of recorders if desired.

2. Press **ENTER**. The fields at the top of the page will update to reflect the change.

Changing the Number of Recorders

The number of recorders can be changed at any time. It can be increased up to the maximum or decreased to zero if desired. Deleting configured recorders by changing the number of recorders is not permitted. For example, if the current number of recorders is ten and recorders one through five have configure tags defined, then the maximum can only be decreased to five. The configure tags for all recorders must be cleared before the number of recorders can be decreased to zero.

To change the number of recorders:

1. Press **ESC** while on the general parameters page (Figure 18-4). Two input fields appear: *Number of recorders* and *Recorder number*. The *Recorder number* field is used to move the input cursor to a specific recorder for editing.

2. In the *Number of recorders* field, enter the new number of recorders. A valid entry is from 0 to 64 (16 for indicator stations and 32 for output transfer groups).

3. Press **ENTER**. The fields at the top of the page will update to reflect the change.

Configuring a Recorder

After setting the number of recorders, each recorder must be configured separately. The general parameters page (Figure 18-4) shows only 16 recorders at a time. Press **NEXT PAGE** or **PREV PAGE** to view additional pages. To configure a recorder:

1. Move to the specific recorder to configure. Optionally:

   a. Press **ESC** to call a *Recorder number* field.

   b. Type in the number of the recorder to configure.

c. Press **ENTER**. This moves the input cursor to a specific recorder for editing.

2. In the *Configure Tag* field, enter the name or index number of the configure tag for this recorder. This is the RCM tag that references the configure block in the PCU module. Refer to ***Configure Block*** and ***CONFIGURE TAG*** in this section for further explanation.

The same configure tag can be used for more than one recorder. Depending on the module configuration, there may be one configure tag for the entire module or one configure tag for each recorder.

3. In the *Recorder Description* field, enter a description of up to 32 characters for the recorder. The description appears on the device definition and device assignment pages to identify this recorder.

4. In the *# Pens* field, enter the number of pens for this recorder. A valid entry is from 0 to 32 (8 for an indicator station and 32 for an output transfer group). This entry determines the number of pen definitions and pen assignments that will be required. The number can be greater than the actual number of pens if desired.

5. In the *Keylock Check* field, enter **YES** to require the function to check the key lock status before allowing changes. Enter **NO** to have the function ignore the key lock.

6. Press **ENTER**.

Repeat this procedure for each recorder that is to be configured.

Changing or Removing a Configure Tag

When a configure tag is removed, the pen definitions and pen assignments for the recorder are marked faulty. If the configure tag is changed to reference a different module, all of its pen definitions and assignments are marked faulty.

To remove a recorder from the general parameters list:

1. Move to the specific recorder to remove.

2. Clear the *Configure Tag* field for the recorder.

3. Press **ENTER**. Leaving an undefined recorder in the list is permitted.

DEVICE DEFINITION

The device definition page identifies the RCM tag associated with each pen of a recorder. Figure 18-5 shows the device definition page. To call the page, from the main menu select the following menu items in the sequence shown.

B Device Definition

→ A Trend Pen Recorder

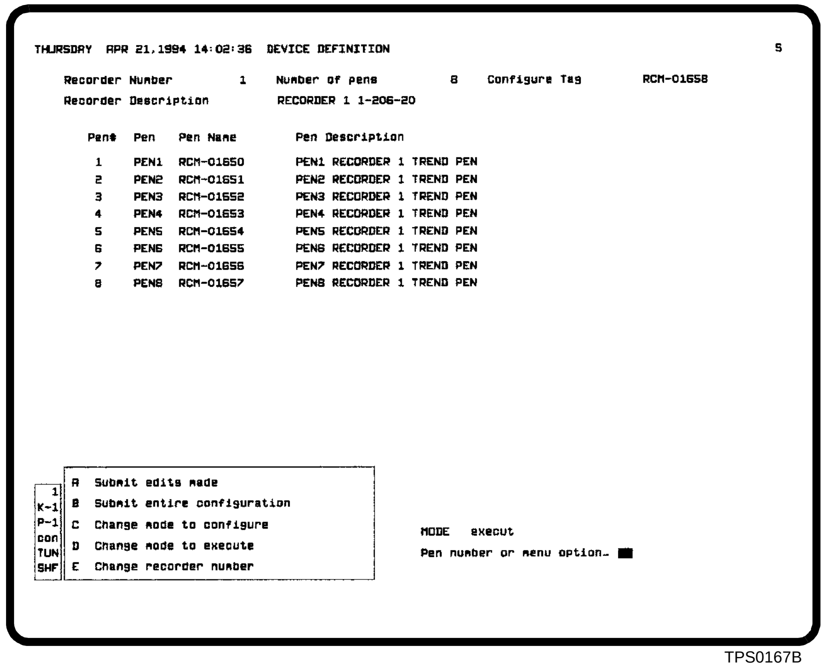


Figure 18-5. Trend Pen Cluster - Device Definition Option

The device definition page shows only 16 pens at a time. Press **NEXT PAGE** or **PREV PAGE** to view additional pages. The fields at the top of the page indicate:

Recorder Number - number of the recorder for which the pens are being defined. The page can be changed to another recorder at any time.

Number of pens - number of pens for the recorder as defined on the general parameters page.

Configure Tag - configure tag assigned to this recorder as defined on the general parameters page.

Recorder Description - description of the recorder as defined on the general parameters page (Figure 18-4).

To change to another recorder:

1. Press **ESC** to call the options menu. Refer to ***Device Definition Options Menu*** in this section for an explanation of the options menu.

2. Type: **E** **ENTER**

3. Enter a recorder number. A valid entry depends on the number of recorders set previously.

4. Press **ENTER**. This calls the recorder and updates the page.

Defining a Pen

The only information required to define a pen is the RCM tag for the pen. This is the tag that allows turning the pen on and off. It references the first block (block n) of the logic that drives the pen. Refer to ***PEN RCM TAG*** for further explanation.

To define a pen:

1. Move to the specific pen to configure.

2. Enter the name or index number of the RCM tag associated with the pen.

3. Press **ENTER**. The pen is marked with an asterisk and the remaining information for the pen is retrieved from the tag database:

Pen - first four characters of the pen description.

Pen Description - tag description as defined for the RCM tag.

Anytime edits are made to a pen, the pen is marked with an asterisk (*) and the following message appears:

EDITS HAVE BEEN MADE

The asterisk and message inform that what is displayed does not necessarily reflect what is in the module. A pen without a pen assignment is also marked with an asterisk. Make pen assignments on the device assignment page.

To save any edits:

1. Press **ESC** to call the options menu. Refer to ***Device Definition Options Menu*** in this section for an explanation of the options menu.

2. Download the information to the module. Type:

A ENTER

- ***or*** -

B ENTER

Submitting configuration changes puts the module into configure mode, downloads the information, then puts the module into execute mode. The module status can be seen in the *MODE* field. The following message appears while the download is in progress:

Operation in progress, Please Wait

NOTE: If this is the initial definition for a pen or a pen assignment does not already exist, use the *Submit edits made* option to save the changes. This does not download any information to the module. Wait until pen assignments are made, then submit the changes to download all information to the module at one time.

Changing or Removing a Pen Definition

A pen definition can be changed or deleted at any time. The module, however, must be in ***configure*** mode to do this. When changing or deleting a definition, the console makes all the necessary changes to the module configuration.

To remove a pen definition:

1. Press ESC to call the options menu. Refer to ***Device Definition Options Menu*** in this section for an explanation of the options menu.

2. Put the module into configure mode. Type: **C** ENTER

3. Wait for the *MODE* field to indicate configure mode, then move to the specific pen assignment to remove.

4. Clear the *Tag Name* field.

5. Press ENTER or move from the field. The remaining information for the pen is cleared.

6. Press ESC to call the options menu.

7. Download the information to the module. Type:

A ENTER

- ***or*** -

B ENTER

When a pen definition is removed, its pen assignment is marked with an asterisk.

Device Definition Options Menu

Press **ESC** while on the device definitions page (Figure 18-5) to call an options menu with the following choices:

A Submit edits made
B Submit entire configuration
C Change mode to configure
D Change mode to execute
E Change recorder number

Option A The *Submit edits made* option saves any edits made, and if a pen already has a pen assignment, the console downloads the configuration to the PCU module. If a pen does not have an assignment, it is saved but not downloaded. The console downloads only the part of the module configuration for which it is responsible, for example, block n+1, block n+5, block n+6 and block n+7 for a pen in block address n. If a download for a pen fails, it is marked with an asterisk (*) next to its number. Pens without a pen assignment are also marked with the asterisk.

Option B The *Submit entire configuration* option downloads the configuration of all pens with assignments to the module. If a download for a pen fails, it is marked with an asterisk (*) next to its number.

Option C The *Change mode to configure* option puts the module where the configure tag of the recorder resides into configure mode. This function is useful when changing RCM tag assignments of previously defined pens. A pen already defined with an RCM tag cannot be redefined with another RCM tag while the module is in execute mode.

Using this option, the logic required to turn a pen off is executed first before the module goes into configure mode. If the module is put into configure mode using some other means, the logic to turn the pen off will not be executed first, leaving the pen at its last position when the module was in execute mode.

Option D The *Change mode to execute* option puts the module where the configure tag of the recorder resides into execute mode. This option allows changing the mode without having to leave the function.

Option E The *Change recorder number* option changes to a different recorder. After selecting the option key in a recorder number, then press **ENTER**.

DEVICE ASSIGNMENT

The device assignment page assigns a process variable to a pen, sets scaling parameters, and allows turning individual pens on and off. Figure 18-6 shows the device assignment page. From the main menu select the following menu items in the sequence shown to call the page.

C Device Assignment

→ *A Trend Pen Recorder*

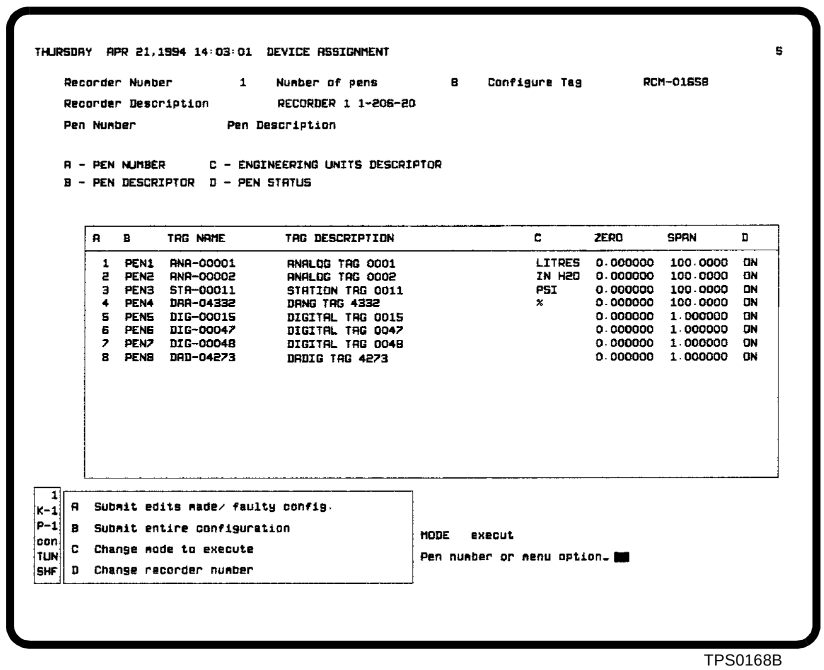


Figure 18-6. Trend Pen Cluster - Device Assignment Option

The device assignment page shows only 16 pens at a time. Press **NEXT PAGE** or **PREV PAGE** to view additional pages. The fields at the top of the page indicate:

Recorder Number - number of the recorder for which the pens are being assigned. The page can be changed to another recorder at any time.

Number of pens - number of pens for the recorder as defined on the general parameters page.

Configure Tag - configure tag assigned to this recorder as defined on the general parameters page.

Recorder Description - description of the recorder as defined on the general parameters page (Figure 18-4).

Pen Number - number of the pen currently selected for assignment.

Pen Description - description from the device definition page (Figure 18-5) of the pen currently selected for assignment.

To change to another recorder:

1. Press **ESC** to call the options menu. Refer to ***Device Assignment Options Menu*** in this section for an explanation of the options menu.

2. Type:

   D **ENTER**

3. Enter a recorder number. A valid entry depends on the number of recorders set previously.

4. Press **ENTER**. This calls the recorder and updates the page.

Making a Pen Assignment

The only information required to assign a process variable to a pen is the name or index number of the tag that references the variable. Refer to ***PROCESS VARIABLE TAG*** in this section for further explanation. A pen assignment can be changed at any time.

To make a pen assignment:

1. Move to the pen to assign.

The *A* and *B* fields identify a pen. The descriptor in the *B* field is the first four characters of the pen description as defined on the device definition page (Figure 18-5).

2. Enter the name or index number of a process variable tag. This can be any analog or digital type of tag.

3. Press **ENTER** or move from the field. The remaining information for the pen is updated:

TAG DESCRIPTION - description as defined in the database for the tag.

C - engineering unit descriptor for the tag. This field is filled in for analog types of tags only.

ZERO - default zero specification, which can be changed to scale the output to the pen.

SPAN - default span specification, which can be changed to scale the output to the pen. This is the difference between the low limit and high limit.

D - on or off status for the pen, which can be changed. The default is off.

> **NOTE:** The *ZERO* and *SPAN* values together set the full scale range for the pen.

Refer to ***Scaling the Output*** in this section for information about zero and span changes. Refer to ***Changing Pen Status*** in this section for information about turning a pen on or off.

Anytime edits are made to a pen, the pen is marked with an asterisk (*) and the following message appears:

EDITS HAVE BEEN MADE

The asterisk and message inform that what is displayed does not necessarily reflect what is in the module. A pen with a fault is also marked with an asterisk.

Making a pen assignment for an undefined pen is permitted. The pen will be marked with an asterisk until it is defined on the device definition page (Figure 18-5). Also, the configuration for the pen is not downloaded to the module until the pen is defined.

To save any edits:

1. Press ESC to call the options menu. Refer to ***Device Assignment Options Menu*** in this section for an explanation of the options menu.

2. Download the information to the module. Type:

   A ENTER

   - ***or*** -

   B ENTER

Submitting configuration changes puts the module into configure mode, downloads the information, then puts the module into execute mode. The module status can be seen in the *MODE* field. The following message appears while the download is in progress:

Operation in progress, Please Wait

A successful download is indicated by the asterisk being removed from a pen. A configuration error during download does not abort the download. The console will continue to submit the changes and indicate the download status after completion. A pen marked with an asterisk after the console completes the download indicates that the recorder module has not been successfully downloaded with the configuration.

Scaling the Output

The *ZERO* and *SPAN* fields on the device assignment page (Figure 18-6) can be used to scale the output sent to a pen:

ZERO	Lowest output value.
SPAN	Difference between the lowest output value and the highest output value.

Scaling can be used to provide better resolution for the recorder. The summer block (FC 15) in the interface logic of a pen performs the scaling. The trend pen cluster function uses the *ZERO* and *SPAN* fields to set the S3 and S4 specifications for the summer block. Both specifications are tunable, which means they can be changed while the module is in execute mode.

To scale the output to a pen:

1. In the *ZERO* field for the pen, enter the desired zero value.

2. In the *SPAN* field for the pen, enter the desired span value.

3. Press ESC to call the options menu. Refer to ***Device Assignment Options Menu*** in this section for an explanation of the options menu.

4. Download the information to the module. Type:

   A ENTER

Use the *Submit edits made/faulty config.* option to download the scale changes. This does ***not*** put the module into configure mode and allows making scale changes while the module executes its configuration.

Example: The input signal ranges from 100 to 200 units. The specific signal range of interest is from 150 to 200 units, which is the 50 percent to 100 percent level of the input signal. It is desired to have this portion of the input signal drive the trend pen full scale to represent the 150 to 200 unit range. To do this, set the following:

ZERO = 150 Low limit.

SPAN = 50 High limit minus low limit; 200-150.

The input summer provides a scaling function as follows:

Output = (<S1> × S3) + (<S2> × S4)

Based on the zero and span entries, the trend pen cluster function calculates the proper gain and sets the S3 specification for the summer block (100 percent scale range/span). In the logic for a pen, S2 of the summer block is defaulted to provide a 0.0 value and S4 is defaulted to 1.0 so the equation reduces to:

Output = (<S1> × S3)

For this example, the gain is calculated as:

S3 = 100.0/50.0 = 2.0

When the input equals 150.0, the output equals:

150.0 × 2.0 - 300.0 = 0%

This is an output in percent. Zero percent corresponds to 150 units.

When the input equals 200.0, the output equals:

200.0 × 2.0 - 300.0 = 100%

This is an output in percent. 100 percent corresponds to 200 units.

When the input equals 175.0, the output equals:

175.0 × 2.0 - 300.0 = 50%

The sum block output is a normalized value ranging from zero to 100. The output may not always be in this range. If the desired signal must be within a certain range, a high/low limit block (FC 6) may be needed and placed immediately after the sum block.

Changing Pen Status

A pen can be turned on or off while the recorder module is in execute mode. The trend pen cluster function changes the state of the RCM tag for a pen to change its status.

To change the status for a pen:

1. In the *D* field for the pen, enter **ON** to turn the pen on or **OFF** to turn it off.

2. Press **ENTER** or move from the field to send the changes to the module.

Optionally, a display can be built containing an RCM element that references the RCM tag for a pen to turn the pen on or off.

Changing or Removing a Pen Assignment

A pen assignment can be changed or deleted at any time. When changing or deleting an assignment, the console makes all of the necessary changes to the module configuration.

To remove a pen assignment:

1. Move to the specific pen assignment to remove.

2. Clear the *Pen Name* field.

3. Press **ENTER** or move from the field. The remaining information for the pen is cleared.

4. Press **ESC** to call the options menu.

5. Download the information to the module. Type:

A **ENTER**

- ***or*** -

B **ENTER**

When a pen assignment is removed, its pen definition is marked with an asterisk.

Device Assignment Options Menu

Press **ESC** while on the device assignments page (Figure 18-6) to call an options menu with the following choices:

A Submit edits made/faulty config.
B Submit entire configuration
C Change mode to execute
D Change recorder number

Option A The *Submit edits made/faulty config.* option saves any edits made and downloads the configuration to the PCU module. The console downloads only the part of the module configuration for which it is responsible, for example, block n+1, block n+5, block n+6 and block n+7 for a pen in block address n. If a download for a pen fails, it is marked with an asterisk (*) next to its number.

NOTE: Use the *Submit edits made/faulty config.* option when making scale and pen status changes. In these cases, the option does ***not*** put the module into configure mode.

Option B The *Submit entire configuration* option downloads the configuration of all pens with assignments to the module. If a download for a pen fails, it is marked with an asterisk (*) next to its number.

Option C The *Change mode to execute* option puts the module where the configure tag of the recorder resides into execute mode. This option allows changing the mode without having to leave the function.

Option D The *Change recorder number* option changes to a different recorder. After selecting the option key in a recorder number, then press ENTER.

Download Description

All pens are temporarily marked faulty (*) when a download is requested. The asterisk is removed from each pen as its download completes successfully. The following occurs during a download:

1. The configure tag is set to its zero state to signal the module to go to configure mode, which also triggers the background logic to place all pens in the predetermined configuration or off position.

2. The trend pen cluster function places the module in ***configure*** mode when the feedback signal to the configure tag is a zero state.

If the configure tag feedback signal remains in a one state, the download procedure cancels. The configuration to be downloaded is saved in the console and marked faulty.

3. After the module is in configure mode, each block of the pen configuration that the console is responsible for is modified, created, or deleted depending on the type of operation.

4. As the download completes for a pen, the pen is marked accordingly by the console depending on the download status. A download failure does not stop the download procedure. The download continues until all pens are processed.

5. After all pens are downloaded, the console puts the module into execute mode. It also returns the pen to the state it was in before the download occurred, either on or off.

The download process can be canceled by pressing ESC or calling another display. The pens that were not downloaded remain in a fault condition until they are successfully downloaded.

Configuration Files

The trend pen cluster configuration files reside in the [DATA.USN02] directory. Table 18-5 lists and describes the trend pen configuration files.

Table 18-5. Trend Pen Configuration Files

File Name	Description
TPCPF001.CF	Configuration information from the general parameters page for trend pen recorders.
TPCPF002.CF	Configuration information from the general parameters page for indicator stations.
TPCPF003.CF	Configuration information from the general parameters page for output transfer groups.
PIO1DFnn.CF	Pen definition and assignment information for trend pen recorder number nn. For example, ***PIO1DF01.CF*** is for trend pen recorder one.
PI02DFnn.CF	Indicator station definition and assignment information for indicator station nn.
P103DFnn.CF	Output transfer group definition and assignment information for output transfer group nn.

The configuration files need to be saved as a set on a per device basis. For example, when transferring a trend pen cluster configuration for trend pen recorders, transfer ***TPCPF001.CF*** and all ***PI01DDFnn.CF*** files. Additionally, the corresponding tags in the database need to be transferred or verified on the destination console.

The configuration is not verified by the console. This must be performed manually after the transfer is complete. Verify referenced tags, set up the referenced target modules, and go through each trend pen recorder assignment.

SECTION 19 - @aGlance/IT CONFIGURATION AND DEFINITION

INTRODUCTION

This section explains @aGlance/IT™ requirements to configure the console and enable @aGlance/IT user-written client programs. The @aGlance/IT servers on the console allow for user-written client programs to have access to real-time, historic, and event log data. Commands that will allow a user-written client program to call-up a console window display are available. The ability to send a command to a user-written server program in response to a key select or touch point is also supported. Refer to the ***Open Data Server/ Client*** instruction for further explanation of @aGlance/IT and its capabilities (Table 1-2 lists instruction numbers).

> **NOTE:** The J.1 operating software prohibits running the @aGlance/IT and User Task Interface features on the console at the same time. @aGlance/IT is automatically enabled when the operating software is installed. Refer to Section 20 for instructions on how to disable @aGlance/IT and enable User Task Interface.

Several methods for sending a command to an @aGlance/IT user-written server program are provided:

- Function key.
- Annunciator display panel (ADP) pushbutton.
- Touch point (i.e., mouse or touch screen).
- Key select.

@aGlance/IT definition is required for two purposes. The first is to identify a user-written server program so that it can be assigned to a function key or ADP pushbutton. The second is to determine how a window display call-up, initiated by a user-written client program, is to operate on the console.

@AGLANCE/IT CONFIGURATION

@aGlance/IT requires special configuration in order for a console to allow @aGlance/IT clients to access it and for the console to access other @aGlance/IT servers. Each @aGlance/IT server can provide access to a maximum of five different clients at the same time. All remote clients and servers must be defined on the local console. Refer to the ***File Utilities*** instruction for information on how to configure the network (Table 1-2 lists instruction numbers). @aGlance/IT has the ability to determine which clients can access the console and the amount of access a client has. Refer to the ***Open Data***

Server/Client instruction for more information (Table 1-2 lists instruction numbers).

NOTES:
1. @aGlance/IT, for a single client/server configuration, is automatically installed with the J.1 operating software. Additional licensed client/servers may be acquired and configured on the INFI 90 Open system. Contact Elsag Bailey for more information.

2. When installing the J.1 operating software, the TCP/IP network must be configured and the PORTMAPPER must be enabled to use @aGlance/IT. See ***File Utilities*** instruction on how to configure and enable these required features (Table 1-2 lists instruction numbers).

FUNCTION KEY AND ADP PUSHBUTTON ACTIVATION

The @aGlance/IT definition creates a list that identifies up to 50 user-written server commands and assigns an index number to each command. The index number of a program can then be defined in a function key or ADP pushbutton assignment. After the assignment is made, the server command can be activated by pressing the key or pushbutton. Refer to Section 8 for the procedures to assign a user-written server command to a function key or ADP pushbutton.

Each @aGlance/IT definition requires entering a server name, server command, and remain connected setting. If desired, a command argument string can be entered that will be passed to the command upon activation.

Use the *@aGlance/IT Definition* option of the user task configuration function to define @aGlance/IT commands. Figure 19-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *F User Task*
→ *A @aGlance/IT Definition*

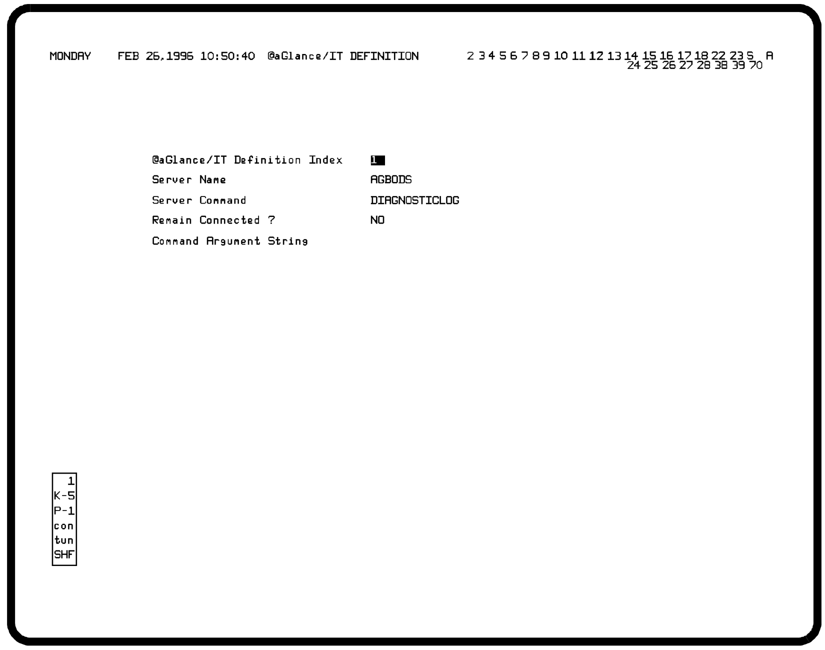
To define an @aGlance/IT user task:

1. Select an @aGlance/IT definition.

   a. Enter a number from 1 to 50 in the *@aGlance/IT Definition Index* field.

   b. Press **ENTER**. If the index number has already been defined, the previously defined information for that index number appears.

2. Enter the name of a server in the *Server Name* field. This is the name of the user-written @aGlance/IT server program. The



TPS1196A

Figure 19-1. @aGlance/IT Page - Definition Option

server name can be a logical or the actual name of a server. If using a logical name, it must appear in the system logical name table. The maximum length for this field is 32 characters.

3. Enter the name of the user defined command to be activated in the *Server Command* field. The maximum length for this field is 32 characters.

4. Enter **YES** or **NO** in the *Remain Connected* field. A **YES** response keeps the client connected to the server after the command is finished executing. A **NO** response disconnects the client from the server once the command is finished executing. Faster subsequent access to the server is achieved by remaining connected.

5. If desired, enter an argument string to be passed to the program upon activation in the *Command Argument String* field. Up to 80 characters can be entered.

6. Press **ENTER**. The index number can now be used in other configurations to identify the specific user-written server command.

KEY SELECT AND TOUCH POINT ACTIVATION

To enable activating a user-written server program by touch point or key select, the select feature must be built into the display during its creation. The display must incorporate an interactive escape command (**ei 107,116** or **ei 108,116**) to enable activating the server program by touch point or by pressing a certain key sequence. The escape command requires the same information as defined with the *Definition* option of the @aGlance/IT user task definition function. Refer to Section 7 for an explanation of the methods used to create displays. Refer to the ***Display Builder Reference*** instruction for a description of the interactive escape commands (Table 1-2 lists instruction numbers).

USER TASK OPTIONS

The call-up of a window display by an @aGlance/IT user-written client program can operate in two different ways on the console. By default, a display activation must be acknowledged before the display is called up. Optionally, the acknowledge requirement can be disabled which will cause an activated display to immediately replace any current display.

If enabled, a dialog box that identifies the window number where the display is to be called up appears. This box contains some dialog generated by the program and two control buttons. The control buttons allow accepting (i.e., *DISPLAY* button) or rejecting (i.e., *IGNORE* button) the display call-up. A time-out parameter in the user-written client program can be used to automatically initiate a display call-up rejection if there is no response within a certain amount of time. The time-out can also be set to zero to have the program wait for a response indefinitely.

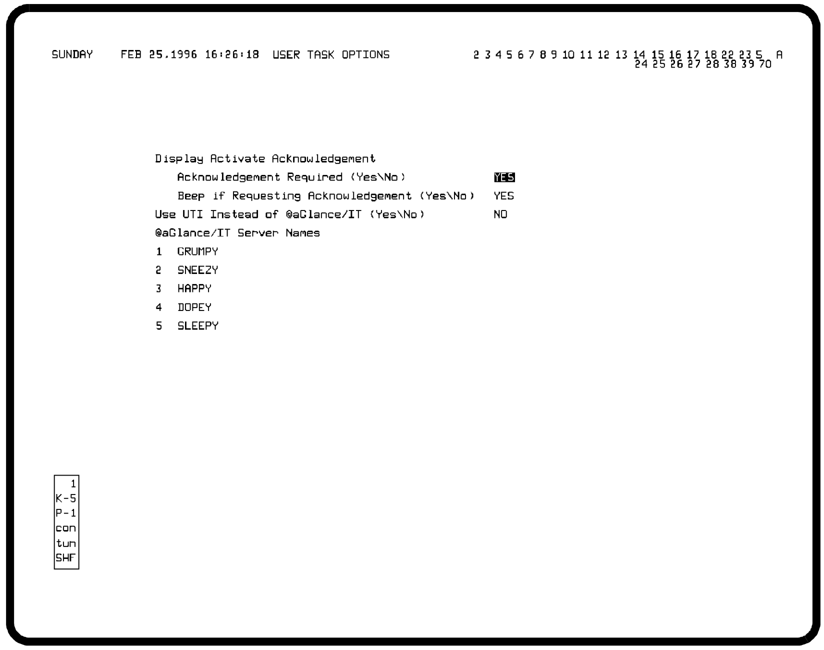
Use *Options* of the user task definition function to define user task operation. Figure 19-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *F User Task*
→ *C Options*

To define the fields of this page:

1. In the *Acknowledgement Required (Yes\No)* field, enter **YES** to have the user task dialog box appear when a display is called up by an @aGlance/IT user-written client program. This enables either accepting or rejecting the window display call-up. Enter **NO** to have the called display immediately replace the current display.

2. Enter **YES** in the *Beep if Requesting Acknowledgement (Yes\No)* field to have a tone sound when the user task dialog box appears. Enter **NO** to disable the tone.



TPS1197A

Figure 19-2. User Task Page - Options

3. The *Use UTI Instead of @aGlance/IT (Yes/No)* field must be set to **NO** to disable UTI and enable @aGlance/IT.

   NOTE: The system must be reset in order for @aGlance/IT to be activated. Refer to Section 2 for procedures on how to reset the system.

4. Enter names of @aGlance/IT servers. The console can have a maximum of five licensed @aGlance/IT servers. When the software is installed all five @aGlance/IT server names are set to the default of EBOIS. It is recommended that each @aGlance/IT server on the console be assigned a unique name. In addition, it is recommended that @aGlance/IT server names on consoles on the same network, that are controlling different processes, also be assigned unique names.

   NOTE: The system must be reset in order for changes in the @aGlance/IT server names to take effect. Refer to Section 2 for procedures on how to reset the system.

5. Press **ENTER**.

SECTION 20 - UTI USER TASK DEFINITION

INTRODUCTION

This section explains the configurations required to enable running a user-written program from the console. The user task interface (UTI) allows a user-written program (user task) to access both static and dynamic database data from the console. It also can be written to call up displays on the console. Refer to the ***User Task Interface*** instruction for further explanation of a user task and its capabilities (Table 1-2 list instruction numbers).

> **NOTE:** In order to use user task interface, the @aGlance/IT™ feature must be disabled. Refer to ***USER TASK OPTIONS*** for the procedure to disable @aGlance/IT and enable user task interface.

Several methods for activating these programs are provided:

- Function key.
- Annunciator display panel (ADP) pushbutton.
- Touch point (i.e., mouse or touch screen).
- Key select.

User task definition is required for two purposes. The first is to identify a user-written program so that it can be assigned to a function key or ADP pushbutton. The second is to determine how display call-up initiated by a user-written program is to operate on the console.

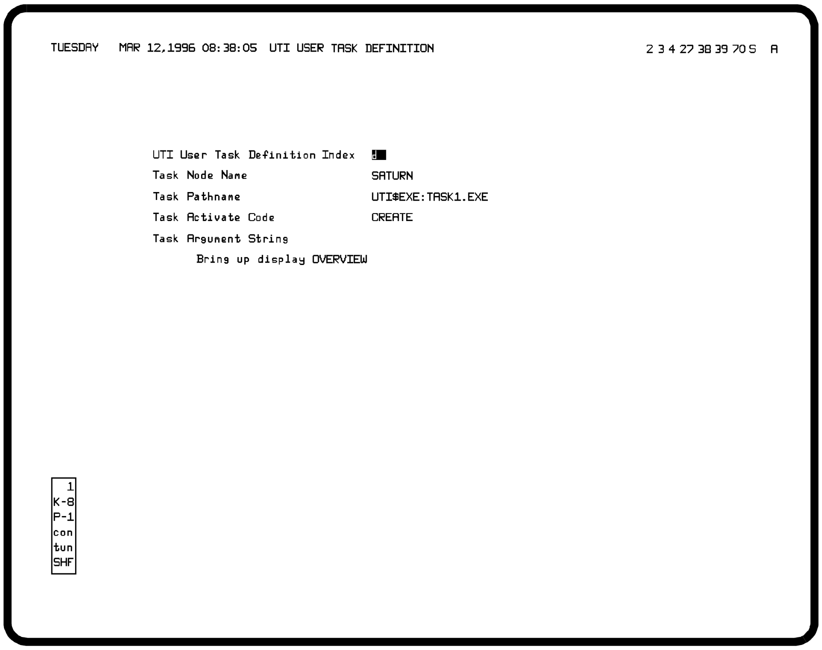
FUNCTION KEY AND ADP PUSHBUTTON ACTIVATION

User task definition creates a list that identifies up to 50 user-written programs and assigns an index number to each program. The index number of a program can then be defined in a function key or ADP pushbutton assignment. After the assignment is made, the task can be activated by pressing the key or pushbutton. Refer to Section 8 for the procedures to assign a user-written program to a function key or ADP pushbutton.

Each user task definition requires entering a task node name, task pathname, and task activation code. If desired, a task argument string can be entered to be passed to the program upon activation.

Use the *Definition* option of the user task definition function to define user tasks. Figure 20-1 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *F User Task*
→ *B UTI Definition*



TPS0230D

Figure 20-1. User Task Page - Definition Option

NOTES:

1. If the user task interface has not been enabled (refer to ***USER TASK OPTIONS*** for procedure), the following message will appear at the bottom of the page: UTI User Task Activation NOT Enabled.

2. UTI definitions can be created and saved without UTI enabled.

To define a user task:

1. Select a user task definition.

   a. Enter a number from 1 to 50 in the *UTI User Task Definition Index* field.

   b. Press **ENTER**. If the index number has already been defined, the previously defined information for that index number appears.

2. Enter the name of a node in the *Task Node Name* field. This is the node that contains the user task interface and user-written program. The node name can be a logical or the actual name of a node. If using a logical, it must appear in the system

logical name table. Another option is to enter **0** if the user task interface and user-written program reside on this console. The maximum length for this field is 32 characters.

3. Enter the name of the UTI program to be activated in the *Task Pathname* field. The name can be a logical or the actual program name. If using a logical, it must appear in the system logical name table of the UTI node. In either case, include the complete path along with the name (i.e., device and directory). The maximum length for this field is 32 characters.

4. Enter the type of activation in the *Task Activate Code* field. A valid entry is **CREATE** or **WAKE**.

If using create, the console expects the program to ***not*** be running. An error message occurs if the program is running and an attempt is made to execute (i.e., create) it again by using a key or pushbutton. Infrequent, one-shot programs that do not require rapid activation should use this activation method.

The wake option expects the program to already be running, but not currently active (i.e., hibernating). If it is not running, the console will execute then activate the program. Frequently activated programs that could possibly be required to respond to multiple activation requests should use the wake activation method.

5. If desired, enter an argument string to be passed to the program upon activation in the *Task Argument String* field. Up to 80 characters can be entered.

6. Press **ENTER**. The index number can now be used in other configurations to identify the specific user-written program.

KEY SELECT AND TOUCH POINT ACTIVATION

To enable activating a user-written program by touch point or key select, the select feature must be built into the display during its creation. The display must incorporate an interactive escape command (**ei 107,113** or **ei 108,113**) to enable activating the program by touch point or by pressing a certain key sequence. The escape command requires the same information as defined with the *Definition* option of the user task definition function. Refer to Section 7 for an explanation of the methods to create displays. Refer to the ***Display Builder Reference*** instruction for a description of the interactive escape commands (Table 1-2 list instruction numbers).

USER TASK OPTIONS

The call-up of a display by a user-written program can operate in two different ways on the console. By default, a display activation must be acknowledged before the display is called up.

Optionally, the acknowledge requirement can be disabled which will cause an activated display to immediately replace any current display.

If enabled, a dialog box that identifies the window number where the display is to be called up appears. This box contains some dialog generated by the program and two control buttons. The control buttons allow accepting (i.e., *DISPLAY* button) or rejecting (i.e., *IGNORE* button) the display call-up. A time-out parameter in the user-written program can be used to automatically initiate a display call-up rejection if there is no response within a certain amount of time. The time-out can also be set to zero to have the program wait for a response indefinitely.

Use *Options* of the user task definition function to define user task operation and enable user task interface. Figure 20-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *F User Task*
→ *C Options*

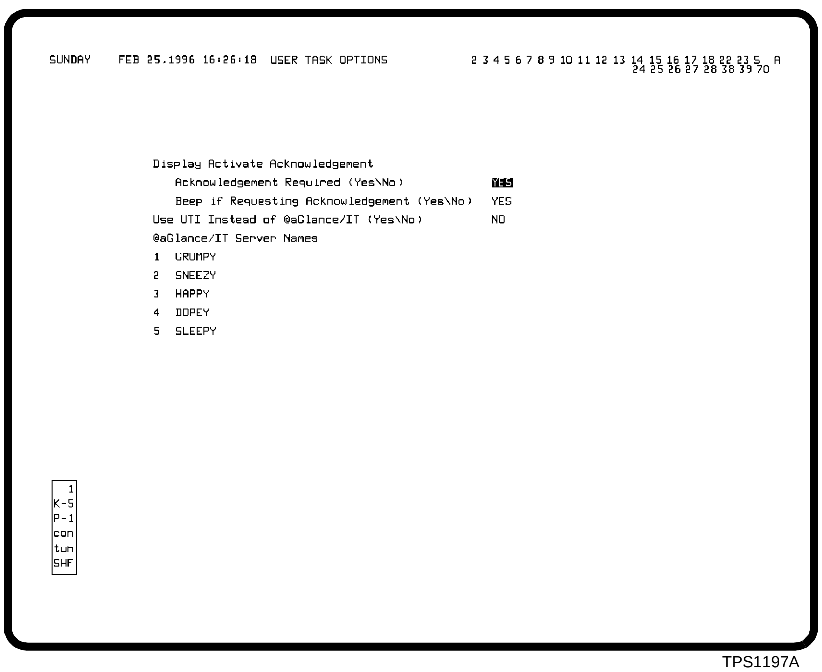


Figure 20-2. User Task Page - Options

To define the fields of this page:

1. In the *Acknowledgement Required (Yes\No)* field, enter **YES** to have the user task dialog box appear when a display is called up by a user-written program. This enables either accepting or rejecting the display call-up. Enter **NO** to have the called display immediately replace the current display.

2. Enter **YES** in the *Beep if Requesting Acknowledgement (Yes\No)* field to have a tone sound when the user task dialog box appears. Enter **NO** to disable the tone.

3. Enter **YES** in the Use UTI Instead of @aGlance/IT (Yes/No) field to disable @aGlance/IT and enable UTI. Enter **NO** to disable UTI.

NOTE: The system must be reset in order for UTI to be activated. Refer to ***GENERAL INFORMATION*** in Section 2 for procedures on how to reset the system.

4. Press **ENTER**.

NOTE: @aGlance/IT server names do not apply to UTI.

SECTION 21 - ALTERNATE LANGUAGE

INTRODUCTION

The console gives the ability to create, use, and maintain an alternate language configuration. Alternate language should be enabled prior to performing any other configuration to allow entering alternate language characters during configuration. This section discusses:

- Enable alternate language character entry.
- Translate system displays.
- Translate default system text.

CONFIGURATIONS

The configuration procedures that pertain to alternate language include:

- System configuration (*General Parameters*). Enables alternate language.

- Character definition (*Change Attributes*). Enables either extended language mode or complex language mode and identifies the specific language being used.

- System attributes for alternate language assignment. Enables entering complex characters in database and system fields. This is only required when in complex character mode.

- System display translator. Used to translate the text of system displays.

- Text substitution. Used to translate various system text.

- Alarm summary format configuration. Used to set up the alarm summary to properly display alternate language characters.

Alternate language must be enabled during system configuration before any alternate language configuration or text substitution can take place. Access to alternate language configuration pages is not permitted until alternate language is enabled.

> **NOTE:** Both the *Character Definition* and *System Attribute for Alternate Language Assignment* options require a reset at completion to be implemented. If enabling complex characters, perform both configurations before resetting the OIS application.

EXTENDED CHARACTERS

An extended character set provides for 256 characters. Most alternate languages can be supported with extended characters. Extended characters follow the same character size and spacing characteristics as the default English ASCII characters. This is required to prevent character spacing problems since both English characters and extended alternate language characters can be entered at the same time in a single field. If the 256 character limit does not support all characters of a language or if the default character size and spacing cannot be maintained to support the language characters, a complex character set is required.

For most alternate languages, the characters required to support the alternate language are contained in the upper 128 character region of an extended set while the English (ASCII) characters are contained in the lower 128 character region. This allows using a single character set for both English and alternate language characters. Enabling extended alternate language allows the console to recognize the characters in the extended region along with the English characters.

For languages that contain more than 128 unique characters, a separate extended character set is required. Enabling extended alternate language in this case allows the console to recognize both the English character set and the alternate language character set.

To enter extended characters, the console must have a keyboard that contains the additional characters of the alternate language. Refer to the discussion on entering alternate language characters in the ***Operation*** instruction for additional information about extended character entry (Table 1-2 lists instruction numbers). A printer that supports the alternate language characters is also required to print text correctly.

The only configuration that is required to put the console into extended character mode is system configuration. Refer to ***ENABLING ALTERNATE LANGUAGE*** in this section for the procedures. After enabling alternate language, the default is extended character mode. The character definition *Change Attributes* option determines whether the console operates in extended or complex character mode.

COMPLEX CHARACTERS

The complex character set is a separate, independent set for languages that use complicated and intricate characters. The complex set gives additional flexibility for creating over 3,000 unique characters.

To enter complex characters, a standard keyboard used in a character Romanization mode is required. Refer to the discussion on entering alternate language characters in the ***Operation*** instruction for additional information on character Romanization (Table 1-2 lists instruction numbers).

A printer that supports the characters of the alternate language is also required to print text correctly.

The complex set is separate from the standard English ASCII character set. For this set, the character height and spacing requirements are expanded to allow creating intricate characters that cannot be reproduced while maintaining the size and spacing standards set for default ASCII characters. The character height and width is roughly double that of standard characters.

In complex character mode, the different types of input fields used by the console must be enabled for alternate language character entry. This is required to identify which fields need to be adjusted to compensate for complex character size and spacing. Use the *System Attribute for Alternate Language Assignment* option to enable the different types of input fields. Once enabled, the console treats any occurrence of that type of field as if it were an alternate language field.

To put the console in complex character mode:

1. Enable alternate language. Refer to ***ENABLING ALTERNATE LANGUAGE*** in this section for the procedures.

2. Use the *Change Attributes* option of character definition to select complex character mode. Refer to ***CHARACTER DEFINITION*** in this section for the procedures.

3. Set database and system attributes to enable alternate language entry. Refer to ***ENABLING SYSTEM ATTRIBUTES FOR ALTERNATE LANGUAGE ASSIGNMENT*** in this section for the procedures.

INPUT FIELDS

An input field will accept either standard ASCII characters only, standard ASCII characters and extended characters, or complex characters only depending on the language. File names, display names, and numeric value fields require English ASCII characters only.

The complex character code causes the length of a standard displayable input field such as *Tag Name* to be half the normal size. The field still, however, requires the same amount of memory to store.

ENABLING ALTERNATE LANGUAGE

System configuration (*General Parameters*) is used to put the console into alternate language mode. Changes made to the general parameters page of system configuration require a reset after completion.

To enable alternate language:

1. Follow the steps given in ***GENERAL PARAMETERS*** in Section 3.

2. Enter **YES** in the *Alternate Language* field to enable alternate language mode and to enable the *Character Definition*, *System Attribute for Alternate Language Assignment*, and *System Display Translator* options. Enter **NO** to disable alternate language. If disabled, the console does not allow entering into the alternate language functions.

CHARACTER DEFINITION

Use the *Character Definition* option to:

- Select either extended character mode or complex character mode.
- Set a left-to-right or right-to-left character path.
- Identify the default character code to use for a space.
- Identify the language type.

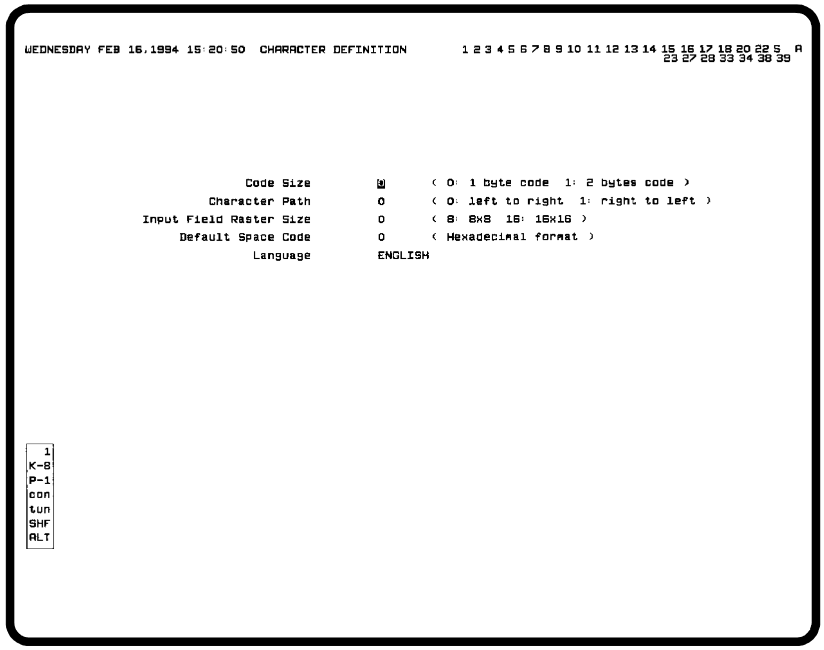
Use the *Change Attributes* option to define the operating parameters for an alternate language. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *D Character Definition*
→ *A Change Attributes*

NOTE: Changes made on this page require a reset to take effect. If the page can be used with the defaults that are already set, do not make any changes and exit this page. The current settings are already established.

To define the fields of this page:

Figure 21-1 shows the page that appears after selecting the *Change Attributes* option. Enter **0** in the *Code Size* field for one-byte extended character mode. Enter **1** in the field for two-byte complex character mode. After setting, either moving away from this field or pressing **ENTER** updates the *Input Field Raster Size* and *Default Space Code* fields.



TPS0013B

Figure 21-1. Character Definition Page - Change Attributes Option

1. The *Input Field Raster Size* field cannot be changed from its current setting.

2. For extended characters, enter **0** in the *Character Path* field to have characters in input fields be entered from left to right. Enter **1** in the field to have characters be entered from right to left. This field affects input fields only and does not affect text returned to fields.

3. The *Default Space Code* field selects the character code that is to be used when either **CLEAR** or **SPACE** is pressed. The default is character code 20 (ASCII blank) for extended characters and A1A1 for complex characters. In most cases, leave this field at its default.

4. Enter the type of language being enabled in the *Language* field.

5. Press **ENTER**. The following message appears:

Reset console for change to take effect

NOTE: If changing to complex character mode, perform system attribute for alternate language assignment configuration ***before*** resetting since it requires a reset at completion.

6. Reset the OIS application by using the procedures given in ***Reset*** in Section 2.

ENABLING SYSTEM ATTRIBUTES FOR ALTERNATE LANGUAGE ASSIGNMENT

Specific database and system attributes must be enabled to allow entering complex alternate language characters. The attributes are enabled by using the *System Attribute for Alternate Language Assignment* option. This configuration is only required if enabling complex character mode.

The configuration also defines a spacing attribute used to adjust character spacing for complex characters. The spacing factor adds pixels to adjust the spacing between characters.

Use the *System Attribute for Alternate Language Assignment* option to enable system attributes for complex character entry. Figure 21-2 shows the page used. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *E System*
→ *A System*
→ *F System Attribute for Alternate Language Assignment*

To define the fields on this page:

1. Enter **YES** for each attribute type to allow entering complex alternate language characters. Leave a field at *NO* to not allow entering complex characters for that type of attribute.

2. Enter a spacing factor in the *Spacing Adjust* field. A valid entry is from 0 to 9. This many pixels are added between each character.

3. Leave the *Romanization* field at its default *NO* unless the particular alternate language uses character Romanization. If it does, enter **YES**.

4. Press **ENTER**.

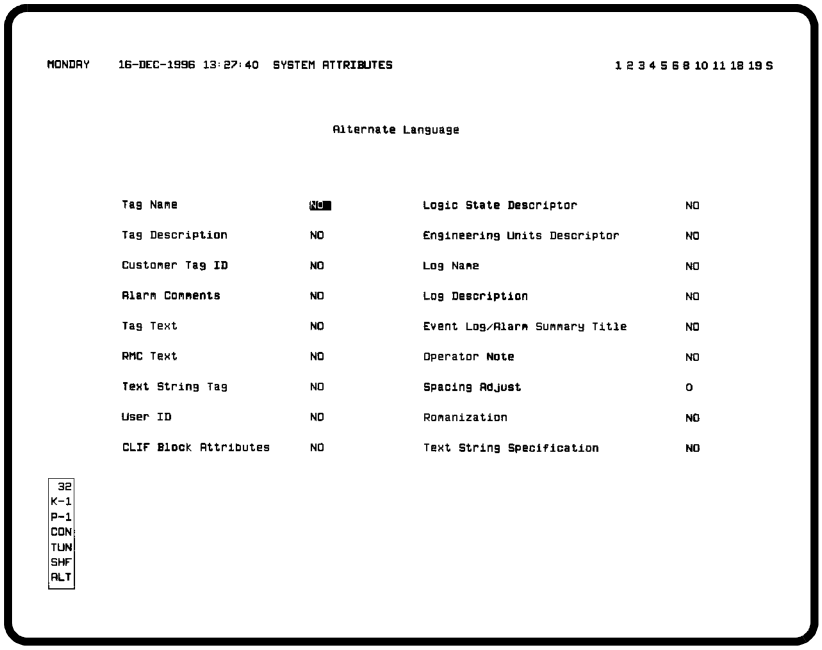


Figure 21-2. System Attribute for Alternate Language Assignment Page

5. Reset the OIS application by using the procedures given in ***Reset*** in Section 2.

DISPLAY AND TEXT TRANSLATION

After alternate language is enabled, standard English text that appears in system displays and most system text can be translated to the alternate language. This can be accomplished by using the *System Display Translator* option and the *Text Substitution* option respectively.

System Displays

The *System Display Translator* option is used to translate text of system displays. This function can be used to modify system menus and interactive pages. The changes are made to the system display files (***DS***) and display partial files (***DP***).

Text Substitution

Text substitution allows changing most console text strings including text that appears in displays, logs, messages, etc. Refer to ***TEXT SUBSTITUTION*** in Section 16 for information about text substitution using alternate language characters.

SYSTEM DISPLAY TRANSLATOR

The display translator is used to translate default system display text to alternate language text. It can also be used to modify the default English text if desired. Character string length, height, and position can also be adjusted. The translator can only be accessed when alternate language is enabled.

Translation is performed on text strings that reside in display files: System displays and display partials. System displays reside on the hard disk in files with ***DS*** extensions. Display partials are separate from system display files and reside on the hard disk in ***DP*** files. System displays and display partials must be translated separately since they reside in different types of files and in different directories.

Display partials are called from system displays. User entries on certain system displays call display partials. The display partial called depends on the entry. For example, the tag configuration page has a standard section at the top of the page that applies to all tag types. After a tag type is entered, the console calls a display partial that contains the additional fields specific to that tag type.

Use the *System Display Translator* option to translate displays. To choose the option, first press **GENL FCTNS MENU**, then select the following menu items in the sequence shown.

A OIS Configuration
→ *B Displays*
→ *B System Display Translator*

Selecting Displays to Translate

The initial page called with the *System Display Translator* option has a single input field. One of four possible modes of operation can be enabled depending on the entry in this field:

- Translate a single system display (i.e., ***DS*** file).
- Translate all system displays.
- Translate a single display partial (i.e., ***DP*** file).
- Translate all display partials.

NOTE: *Appendix E* contains a list of system display file names. The displays that reside as ***DU*** files cannot be modified using this function. The source files (***DT***) for these displays must be edited instead.

SINGLE FILE MODE

A file name must be known to translate a single system display or a single display partial. Enter the name of a system display or a display partial without its ***DS*** or ***DP*** extension. To translate a system display, type the file name as:

!*filename* **ENTER**

To translate a display partial, type the file name as:

filename **ENTER**

The exclamation point (!) is required to distinguish between a system display file and a display partial file.

WILD CARD MODE

When a file name is not known or when all system displays or display partials are to be translated, wild card mode can be used. The wild card character is the asterisk (*). In wild card mode, each file on the hard disk is called alphabetically and each text entry in a called file is sequenced through for editing. The configuration page that appears provides an option to select or pass over each text entry in the file (display).

To enter wild card mode to translate all or unknown system displays, type:

!* ENTER

To enter wild card mode to translate all or unknown display partials, type:

***** ENTER

Wild card mode can be exited at any time. To exit wild card mode:

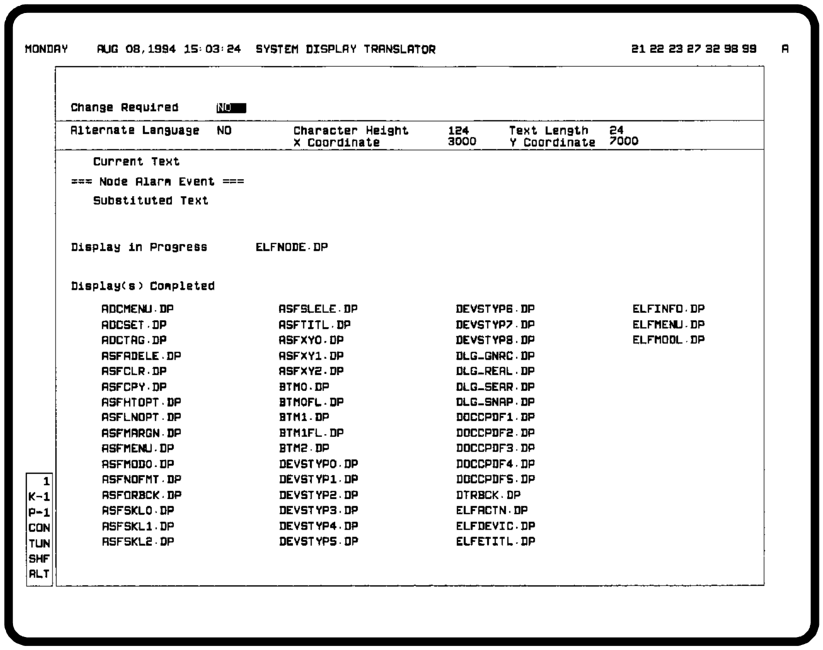
1. Press ESC. A *QUIT WILDCARD* prompt appears.
2. Enter **YES**.
3. Press ENTER.

Translating a Display

Figure 21-3 shows an example of the configuration page used to translate displays. The page appears after selecting the displays to translate.

If in single file mode, only the text entries in the selected file can be translated. All text entries in all files on the hard disk are sequenced through and can be translated when in wild card mode. Refer to ***Selecting Displays to Translate*** in this section for an explanation of the modes.

The current text entry of a file can be seen in the *Current Text* field. For any text entry in a display file that is not to be translated, leave the *Change Required* field at *NO* and press ENTER. The next text entry in the file is called automatically. In wild card mode, when all text entries in a given file are either translated or passed over, the next display file is automatically called and the text in the file can then be edited.



TPS0059B

Figure 21-3. System Display Translator Page - Text Substitution Display

The *Display in Progress* field shows the current file being processed. If in single file mode, the field does not change. When in wild card mode, the field shows the current file in the sequence of files. It updates to the name of the next file in the sequence after all text entries for the current file have been processed. Any files that have either been passed over or previously translated appear under the *Display(s) Completed* heading.

To select and edit the current text entry:

1. Enter **YES** in the *Change Required* field, then press **ENTER**. This positions the input cursor on the *Alternate Language* field.

2. Enter **YES** in the *Alternate Language* field.

   NOTE: If possible, use the current character height, text length, and coordinate settings. When changing any of these attributes, the entire display layout must be considered, not just the current text string.

3. Enter a character height for this text string in the *Character Height* field. The height is limited to 250.

4. Enter a text string length in the *Text Length* field. The length can be adjusted to allow more or less characters if required. The maximum length is 80 characters.

5. In the *X Coordinate* and *Y Coordinate* fields, enter an x,y coordinate. This is used to position the text string. It identifies the position where the console is to begin drawing the string.

A valid entry for the *X Coordinate* ranges from 400 to 9600. For the *Y Coordinate*, the entry can be from 400 to 7300. The y-coordinate should not exceed 7200 except for title line entries. It should not be less than 500 for complex characters.

> **NOTE:** Exercise caution when changing text position. Touch points do not follow the new position.

6. Press **ENTER**. This positions the input cursor on the *Substituted Text* field.

7. Enter the text that is to appear in place of the current text.

8. Press **ENTER** to save and implement the change. The change can be viewed after exiting this function and calling the translated display.

9. After pressing **ENTER**, the input cursor positions on the *Change Required* field again, and the next text entry in the file appears. Repeat the steps to continue making changes.

10. When all text entries for the file, or for all files if in wild card mode, have been sequenced through, the message *Processing Complete* appears. If in single file mode, press **ESC** to exit. If in wild card mode:

    a. Press **ESC** to call the *QUIT WILDCARD* prompt.

    b. Enter **YES** and press **ENTER** to exit.

APPENDIX A - TASK CODES

INTRODUCTION

Task codes are used to identify specific console functions. This section provides a list of task codes and their meaning separated into interactive tasks, resident tasks, and window and network tasks.

CONFIGURATION AND OPERATION TASK CODES

Table A-1 lists the codes that apply to configuration programs and programs that interact with the operator.

Table A-1. Configuration and Operation Task Codes

Code	Description
ACC	Alarm comment file; displays and verifies the contents of the alarm comment file.
ADC	Automatic display configuration; assigns DIGITAL tags, displays, and pop up elements to automatic display tag sets.
ADS	Annunciator display panel configuration; assigns displays, user task programs, and key macros to annunciator display panel (ADP) pushbuttons.
AGC	Alarm group configuration; configures alarm groups for tones, relays, and tag ranges.
AMC	Alarm management configuration; selects method of triggering alarm tones and relays, enables or disables alar group and alarm priority indicators, and determines how left edge alarm priority indicators are displayed.
APC	Alarm priorities configuration; configures alarm priorities for tones, relays, and primary display.
APT	Application processor task definition; defines user task interface parameters and priority.
APO	Application task option; configures the user task interface options.
ASF	Alarm summary format configuration; configures line formats, colors, and titles for periodic alarm summaries.
CDT	Character definition; allows the selection of a foreign language font and font related items.
CEL	Report generator cell definition; defines the format of custom logs through a spreadsheet-like interactive.
CMD	Communication module details; reports the specifications for selected communications modules. This is part of the INFI-NET diagnostics function.
CTP	Interface unit task priority; sets the priority of transactions for the communications interface unit.
DAC	Display archive data interaction; displays the directories of retrieved archive data.
DED	Display generator; translates ***DT*** display source files into binary ***DU*** and ***DL*** files.
DLG	Display log; displays custom logs and SOE logs. It also enables viewing the continuous events log and the periodic events log.
DMD	Display mask definition; sets security levels for individual displays.
DVT	Archive data type to volume definition; assigns volume name, overwrite priority, and on or off status changes for archive volumes.
EEC	Event and error counters; reports the event and error counters for selected nodes. This is part of the INFI-NET diagnostics function.

Table A-1. Configuration and Operation Task Codes (continued)

Code	Description
ELF	Event log format configuration; configures event log item formats and colors.
EUD	Engineering unit descriptors definition; configures engineering unit descriptors.
EVT	Event log configuration; selects alarms, state changes, operator notes, operator actions, logical printer numbers, etc., for event logs.
GEN	Report generator configuration; configures start times, completion times, log types, number of retentions, etc., for custom logs.
HIC	Tag historian; defines groups of tag data to be archived.
INH	Group alarm inhibit; inhibits entire alarm groups by individual group or range of groups.
KTA	Key to action assignment; assigns displays, user task programs, and key macros to keyboard function keys.
LCD	Logical CRT definition/console definition; associates a logical CRT number with a specific console and window. This information is used with the password security function.
LOG	Logging parameters configuration; defines logging shift times.
LSD	Logic state descriptors definition; configures logic state descriptors.
LST	Log status summary; displays current log status for all log types. Allows activation, deactivation, and cancelation of log reports.
LTR	Loop topology report; displays the topology of a loop. This is part of the INFI-NET diagnostics function.
MAC	Key macro definition; configures macros of key sequences to be processed when ADP pushbuttons or function keys are pressed.
MCF	System configuration; configures primary system attributes.
MFR	Module firmware; displays existing module firmware revision levels for the selected modules.
MIS	Archive miscellaneous definitions; defines disk space available for archive storage and retrieval and the fullness warning level.
MPR	Module problem report; displays module status bytes and module problem reports using soft key escapes.
MWC	Multiwindow configuration; defines a display list containing display names and target windows for function key and ADP pushbutton activation.
NTR	Node topology report; reports the topology for selected nodes. This is part of the INFI-NET diagnostics function.
OAD	OAS node definition; configures the open access system (OAS) node used for trend data and log collection.
OCC	Operator configurable display configuration; assigns faceplate names, trend colors, etc., for operator configurable displays.
OCD	Operator configurable display builder; creates operator displays using faceplates for tags, trends, and alarm summaries.
OTA	Operator assignable trend configuration; assigns tags and function blocks for operator assignable two-second trend displays.
PA	Printer assignment; assigns logical printer numbers to actual physical printers.
PAS	Alarm summary report definition; configures periodic and tag triggered alarm summary reports.
PCM	Printer color maps; defines text and line color substitution for hard copies on systems with color printers.
PDF	Peripheral device failover; configures the failover device and method of failover for printers.
PENA	Trend pen general parameters.
PENB	Trend pen device definition.

Table A-1. Configuration and Operation Task Codes (continued)

Code	Description
PENC	Trend pen device assignment.
PLS	PCU management; downloads, saves, and verifies PCU module configurations from the console.
POI	PCU configuration; configures function blocks, module mode changes, etc.
PRM	Operating parameters; displays the characteristics of a tag such as current value or state, alarm limits, alarm comment, etc. at a single display. It also enables operations specific to certain types of tags such as manually inhibit, scan on or off, substitute value, etc.
QPC	Queued print cancelation; displays a list of files queued to a printer and gives the ability to cancel any or all queued print requests.
REL	Relay configuration; configures logical relays for duration, keyboard number, and physical relay of a keyboard.
RET	Archive retrieval interaction; retrieves archived trends, logs, events, PCU configurations, and tag data.
RMC	RMCB text configuration; configures text for remote motor control block error codes.
RTQ	Red tag notice summary; generates summary of red tag key information that is stored in the central red tag information repository.
RTT	Red tag status; allows viewing and modifying red tag status for a function block. Modify includes set and reset operations.
SAT	System attribute configuration; selects system attributes such as tag name, alarm comment, and tag descriptor for alternate languages.
SDE	Show display errors; retrieves errors encountered within a display source file (***DT***) during processing of the file with the display generator.
SMC	Security mask configuration; defines security level, access rights, key lock function, and logical CRT access.
SOEA	SOE recorder configuration; configures any parameters associated with sequence of event recorders.
SOEB	SOE report definition; configures any parameters associated with sequence of events reports.
STO	Archive storage interaction; enables demanding archival storage of events, logs, PCU configurations, trends, and tag data.
TCF	Tag configuration; configures attributes for all tag types such as tag name, descriptor, block address, etc.
TDC	Trend definition; configures attributes for distributed trends such as trend resolution, collection period, and block address.
TDF	Time and date format; defines the format to display the system time and date on the console.
TDP	Trend list to printer; prints the trend database using different search constraints such as trend mode, hardware address, and wild card name options.
TIM	Set time and date; adjusts time and date for the console, which also propagates to the INFI 90 OPEN system through time synchronization.
TLP	Tag list to printer; prints tag configuration using different search attributes such as tag type and wild card names.
TON	Tone configuration; configures logical tones for duration, keyboard number, volume, pitch, and physical annunciator of a keyboard.
TSC	Text substitution; configures all ASCII text throughout the console.
TST	Text selector block text configuration; configures text message numbers for support of the text selector block (function code 151).
TSU	Tag summaries; displays and prints tag data using data criteria such as tags in alarm, bad quality, off scan, and inhibited.

Table A-1. Configuration and Operation Task Codes (continued)

Code	Description
UPD	User password definition; defines user ID, password, security level, access to individual tag groups and regional access for individual passwords.
VMD	Archive volume to media definition; defines archival volume descriptors, media type, time span, and time of day to output.
XCP	Exception statistics; reports the exception statistics for selected nodes. This is part of the INFI-NET diagnostics function.
XYC	XY plot definition; defines XY plot parameters such as status, plot type, and mode.

RESIDENT TASK CODES

Table A-2 lists the codes that apply to background programs (data collection, I/O interfaces, etc.) that make up the core of the system.

Table A-2. Resident Task Codes

Code	Description
ADX	Automatic display executive; handles the automatic call up of displays and pop up elements based on the state transition of a DIGITAL tag.
AS	Alarm summary; handles display of alarms in custom alarm summaries.
BTM	PCU configuration; handles the reading of block data, deleting blocks, and writing blocks for control module configuration.
CCP	Keyboard interaction; handles interfacing between the console and keyboard for input from the operator keyboard, touch point, annunciator display panel, and auxiliary engineering keyboard.
CHR	Chronological scheduler; handles periodic scheduling of logs, periodic alarm summaries, and operator assignable trends.
CIU	Communications interface unit task; handles interfacing of the console with the IIMCP01 module. Translates messages within the system (trend polls, control requests, tag connects, etc.) into command and reply sequences for the interface unit. Polls exception reports and routes the reports to database tasks.
DBR	Database retrieval executive; handles data retrieval for the tag historian.
DBS	Database storage management; handles data storage for the tag historian.
DDT	Diagnostic/debug terminal; utility and diagnostic task used to create files, delete files, copy files, display memory, run command files, etc.
DOT	Display optimizing translator; handles optimization of user-configured displays and real-time trend displays.
DSU	Dynamic display manager; handles dynamic updates of tag data on faceplates and any graphic displays.
DSX	Display executive task; handles the forwarding of display requests to all tasks that need to search the display file for dynamic items, escape sequences, trend elements, alarm elements, etc.
ERR	Operator messages; handles display of operator error messages on the bottom of the window.
EVL	Event log manager; handles formatting of event log items for print and storage to disk.
FCP	Function code processor; interfaces function block operation requests from the virtual module of the console to its communications interface unit.
GPL	General polling data manager; handles collection of tag exception report data and block output data for operator assignable trends.

Table A-2. Resident Task Codes (continued)

Code	Description
INI	Initialization; handles start-up of all console tasks, reading of database into memory, and initialization of various memory tables during start-up.
INT	Interactive program executive; handles activation of any interactive type program or background task.
ITn	Interactive loader; handles relocation of any interactive program called to window one (IT0) to window eight (IT7).
LDC	Logging data collector; handles collection of tag and trend data for custom logs.
NCP	Node communications processor; handles tag broadcast messages to and from a global database manager (GDM) work station and display file transfers from a console configuration tools (WLDG) work station.
OCn	Operator control; handles operator control of process control and data acquisition tags for window one (OC1) to window eight (OC8).
ODSn	OIS @aGlance/IT Data Server; processes @aGlance/IT client requests for servers one (ODS1) through server five (ODS5).
OPS	Operator status; handles output of time, date, alarm groups, and keyboard status information.
ORA	Operator request for action; handles display and interaction with operator for action requests such as asking for a floppy disk to be installed in the drive for archive purposes.
PRX	Periodic executive task; handles periodic scheduling of alarm summary and event logs.
PST	Print spooler; handles interfacing with printers for printing logs, events, screen copies, files, etc.
PSX	Password security executive; controls the password security functions. Checks access to various regions and any log-in and log-out on the console.
RDM	Magnetic tape data archive manager.
RDO	Optical disk data archive manager.
RDX	Archive executive task.
RPT	Report format; handles formatting of custom logs for print and archive.
RRX	Archive retrieval executive task.
RSX	Archive storage executive task.
RTM	Red tag manager task; handles red tag key information being stored to and retrieved from the central red tag information repository.
SCP	Segment control processor; executes the function code blocks in the virtual module of the console. It supports up to eight segment control blocks in the virtual module.
SCT	Print screen; sets up the OIS application window for the copy screen function to capture.
SEL	Sequence of events data collection; handles collection of SOE data for SOE logs.
SLC	Soft key select interactive; handles interfacing with operator for call up of a tuning display, block details display, operator assignable trends display, etc., with the soft key escape.
TD1	Trend display; handles display of distributed trend data and operator assignable trend data. It also handles panning, zooming, and time cursor movement on trend displays.
TDD	Tag dynamic database manager; handles alarm processing of exception reports, database update from exception reports, and requests from within the system for tag dynamic data.
TGD	Tag static database manager; handles connects and disconnects to the tag database to acquire static configuration data (names, descriptors, etc.).
TRD	Trend data manager; handles trend definition changes and return of distributed trend data for display, logging, and archive purposes.
VDI	Virtual display metafile interpreter; handles interface with the graphics card for display data. Also does translation of display commands into graphics card format.

Table A-2. Resident Task Codes (continued)

Code	Description
XYM	XY plot data manager; handles data collection and management for XY plots.
XYP	XY plot; handles drawing and updating of XY plots.

WINDOW AND NETWORK TASK CODES

Table A-3 lists the codes that apply to window and network operations.

Table A-3. Window and Network Task Codes

Code	Description
BLK_00n	Manages blinking of colors for window one (001) through window eight (008).
CCP_SYM_00n	Processes an X window keyboard event for the CCP task for window one (000) through window eight (007).
MBOX_SERVER	Manages sending and receiving of mail messages through prioritized mailbox queues.
MILLISEC	Increments the millisecond counter for the console.
MKISYM	Reads keystrokes from the operator keyboard for the CCP task.
RDC	Sends archived logs to an OAS node for storage.
RSCT	Remote screen copy task; captures display screen.
RSCT_SERVER	Remote screen copy server task; handles capturing and printing of display screen.
TRD_CLIENT	Handles requests from the TRD task for trend data from an external source. Sends a request to the external node, then routes any received data to the TRD task.
UTI	Routes user task requests (e.g., read value, get trend data, etc.).
UTI_ACT	Client task that handles user task activation through touch point, key select, or ADP pushbutton.
UTI_PRT	Handles various print requests for user tasks.
UTI_SERVER	Handles local and remote requests from user task nodes to access INFI 90 Open process data and certain OIS resources and replies to these requests.
V_CPUQMON	Monitors execution of OIS tasks.
VDI_SYM_00n	Processes X window graphic requests for the VDI task for window one (000) through window eight (007).

APPENDIX B - MENU STRUCTURE

MENUS

Most console functions are accessed through menu selections. This section provides a tree structured view of the menu hierarchy for the console.

The general functions menu has three separate choices, each providing access to a group of related operations, utilities, or configurations. Figure B-1 shows the menu options accessed after selecting either *A OIS Configuration*, *B OIS Utilities* or *C OIS Operation* from the menu. Figures B-2 and B-3 show the *OIS Configuration* menu structure and Figures B-4, B-5, and B-6 show the *OIS Utilities* menu structure.

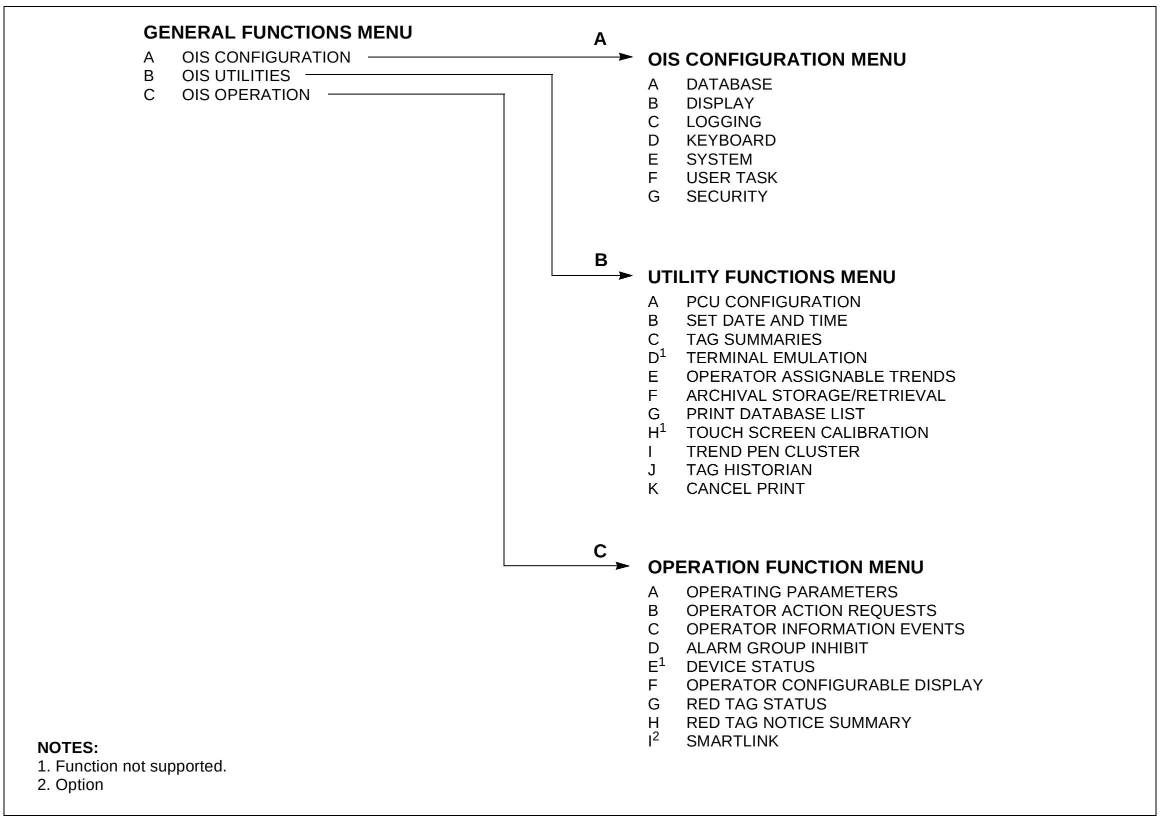


Figure B-1. Menu Structure - General Functions Menu

OIS CONFIGURATION MENU

A DATABASE
B DISPLAY
C LOGGING
D KEYBOARD
E SYSTEM
F USER TASK
G SECURITY

A

DATABASE CONFIGURATION

A TAG
B TREND
C LOGIC STATE DESCRIPTORS
D ENGINEERING UNITS
E TEXT SUBSTITUTION
F TAG TEXT SELECTOR
G REMOTE MOTOR CONTROL TEXT
H XY PLOT
I ALARM COMMENTS

B

DISPLAY CONFIGURATION

A DISPLAY GENERATOR
B SYSTEM DISPLAY TRANSLATOR
C OPERATOR DISPLAYS' FACEPLATES
D SHOW DISPLAY ERRORS
E AUTOMATIC DISPLAYS
F MULTIWINDOW DISPLAY ACTIVATION

C

LOGGING CONFIGURATION

A REPORT GENERATOR
B SYSTEM EVENT LOG
C SEQUENCE OF EVENTS LOG
D LOGGING PARAMETERS
E EVENT LOG FORMAT

C

SOE CONFIGURATION

A SER DEFINITION
B SOE REPORT DEFINITION

E

EVENT LOG FORMAT

A EVENT LOG TITLE
B OPERATOR ACTION LOG TITLE
C DIGITAL EVENT
D ANALOG ALARM EVENT
E OPERATOR ACTION EVENT
F OPERATOR NOTE EVENT
G INFORMATION EVENT
H NODE ALARM EVENT
I MODULE ALARM EVENT
J DEVICE STATUS ALARM EVENT
K TEXT SELECTOR EVENT
L CONFIGURATION ACTION EVENT

Figure B-2. Menu Structure - OIS Configuration (1 of 2)

OIS CONFIGURATION MENU

A DATABASE
B DISPLAY
C LOGGING
D KEYBOARD
E SYSTEM
F USER TASK
G SECURITY

D

KEYBOARD CONFIGURATION

A FUNCTION KEYS
B ADP
C MACROS

E

SYSTEM CONFIGURATION

A SYSTEM
B ALARM MANAGEMENT
C PRINTER ASSIGNMENTS
D CHARACTER DEFINITION
E ALARM SUMMARY FORMAT
F ALARM SUMMARY REPORT
G TIME/DATE FORMAT
H PRINTER COLOR MAPS
I LOGICAL CRT DEFINITION
J CONSOLE DEFINITION
K PERIPHERAL FAILOVER
L CIU TASK PROCESSING

F

USER TASK CONFIGURATION

A @AGLANCE/IT DEFINITION
B UTI DEFINITION
C OPTIONS

G

SECURITY MAINTENANCE MENU

A SECURITY OPTION CONFIGURATION
B SECURITY LEVEL CONFIGURATION
C USER PASSWORD DEFINITION
D DISPLAY MASK DEFINITION

A

OIS SYSTEM CONFIGURATION

A GENERAL PARAMETERS
B GLOBAL ALARM MANAGEMANT
C REMOTE ACKNOWLEDGE TAG ASSIGNMENTS
D ALARM QUALITY OPTION
E SOE GENERAL PARAMETERS
F SYSTEM ATTRIBUTE FOR ALTERNATE LANGUAGE ASSIGNMENT
G X DEVICE DEFINITIONS
H KEYBOARD DEFINITIONS
I OAS DEFINITIONS
J OPERATOR CONTROL OPTIONS

B

ALARM MANAGEMENT

A ALARM MANAGEMENT OPTIONS
B ALARM GROUPS
C ALARM PRIORITIES
D ALARM TONES
E ALARM RELAYS

D

CHARACTER DEFINITION

A CHANGE ATTRIBUTES

E

ALARM SUMMARY FORMAT

A DISPLAY/MODIFY FORMAT
B COPY FORMAT
C CHANGE LINE OPTION
D CHANGE HEIGHT OPTION
E CHANGE NUMBER OF FORMATS
F TITLE CONFIGURATION
G PRIORITY COLORS

Figure B-3. Menu Structure - OIS Configuration (2 of 2)

UTILITY FUNCTIONS MENU

A PCU CONFIGURATION
B SET DATE AND TIME
C TAG SUMMARIES
D[1] TERMINAL EMULATION
E OPERATOR ASSIGNABLE TRENDS
F ARCHIVAL STORAGE/RETRIEVAL
G PRINT DATABASE LIST
H[1] TOUCH SCREEN CALIBRATION
I TREND PEN CLUSTER
J TAG HISTORIAN
K CANCEL PRINT

A

PCU CONFIGURATION

A MODULE CONFIGURATION
B PCU MANAGEMENT
C INFI-NET DIAGNOSTICS

C

TAG SUMMARIES

A M GENERAL TAGS
B N ACKNOWLEDGED TAGS
C O UNACKNOWLEDGED TAGS
D P ALARMS
E Q TAGS BY QUALITY
F R RED TAGS
G[1] S[1] UNCLEARED TAGS
H T SUPPRESSED TAGS
I U TAGS BY ATTRIBUTE
X ENTER SUMMARY START TAG
Y ENTER SUMMARY END TAG
Z ENTER SEARCH PATTERN

E

OPERATOR ASSIGNABLE TRENDS

A OPERATOR TREND ASSIGNMENT
B TREND DISPLAY LIST 1 AND 2
C TREND DISPLAY LIST 3 AND 4

A

MODULE CONFIGURATION

A SELECT LOOP/PCU/MODULE
B CHANGE MODULE MODE
C MODIFY A BLOCK
D ADD A BLOCK
E DELETE A BLOCK
F TUNE A BLOCK
G COPY A BLOCK
H MOVE A BLOCK
I FIND FUNCTION CODE
J DISPLAY BLOCKS
K SHOW FUNCTION CODES

B

PCU MANAGEMENT

A SELECT LOOP/PCU/MODULES/MEDIA
B CHANGE MODE OF MODULES
C SAVE CONFIGURATION TO MEDIA
D LOAD CONFIGURATION FROM MEDIA
E VERIFY CONFIGURATION AGAINST MEDIA
F PRINT CONFIGURATION FROM MEDIA
G PRINT CONFIGURATION FROM MODULES
H DISPLAY DIRECTORY OF CONFIGURATIONS

C

INFI-NET DIAGNOSTICS

A SELECT LOOP/NODE/SORT
B LOOP TOPOLOGY
C NODE TOPOLOGY
D COMMUNICATION MODULE DETAILS
E EVENT AND ERROR COUNTERS
F EVENT AND ERROR COUNTERS (RESETTABLE)
G RESET EVENT AND ERROR COUNTERS
H EXCEPTION STATISTICS
I RESET EXCEPTION STATISTICS
J[1] PERFORMANCE STATISTICS
K PRINT NETWORK TOPOLOGY

NOTE: 1. Function not supported.

Figure B-4. Menu Structure - OIS Utilities (1 of 3)

UTILITY FUNCTIONS MENU

- A PCU CONFIGURATION
- B SET DATE AND TIME
- C TAG SUMMARIES
- D[1] TERMINAL EMULATION
- E OPERATOR ASSIGNABLE TRENDS
- F ARCHIVAL STORAGE/RETRIEVAL → F

F — ARCHIVAL CONFIGURATION

- A ARCHIVAL CONFIGURATION → A
- B DIRECTORY OF ARCHIVE VOLUME → B
- C DIRECTORY OF RETRIEVED DATA → C
- D RETRIEVE DATA → D
- E STORE DATA → E
- F USE NEW VOLUME → F

A — DEFINE ARCHIVING

- A DATA TYPE TO VOLUME DEFINITION
- B VOLUME TO MEDIA DEFINITION
- C MISCELLANEOUS DEFINITIONS

B — RETRIEVE ARCHIVED DATA

DIRECTORY OF ARCHIVE VOLUME

CRT	PRT	
A[1]	D[1]	FLOPPY
B	E	MAGTAPE
C	F	OPTICAL

C — DIRECTORY OF RETRIEVED DATA

- A TAG DATA
- B TREND DATA
- C PCU CONFIGURATIONS

D — RETRIEVE ARCHIVED DATA

- A RETRIEVE EVENTS
- B RETRIEVE LOGS
- C RETRIEVE TAGS
- D RETRIEVE TREND DATA
- E RETRIEVE PCU CONFIGURATION
- F UNUSED ARCHIVE TREND INDICES

E — ARCHIVAL STORAGE

- A STORE EVENTS
- B STORE LOGS
- C STORE TAG DATA
- D STORE TREND DATA
- E PCU CONFIGURATION

F — ARCHIVAL STORAGE

USE NEW VOLUME

- A EVENTS
- B LOGS
- C TAGS
- D TRENDS
- E PCU CONFIGURATION

Remaining UTILITY FUNCTIONS MENU items:

- G PRINT DATABASE LIST
- H[1] TOUCH SCREEN CALIBRATION
- I TREND PEN CLUSTER
- J TAG HISTORIAN
- K CANCEL PRINT

NOTE: 1. Function not supported.

Figure B-5. Menu Structure - OIS Utilities (2 of 3)

G

DATABASE LIST MENU

A TAG LIST → A
B TREND LIST → B

UTILITY FUNCTIONS MENU

A PCU CONFIGURATION
B SET DATE AND TIME
C TAG SUMMARIES
D[1] TERMINAL EMULATION
E OPERATOR ASSIGNABLE TRENDS
F ARCHIVAL STORAGE/RETRIEVAL
G PRINT DATABASE LIST → G
H[1] TOUCH SCREEN CALIBRATION
I TREND PEN CLUSTER → I
J TAG HISTORIAN → J
K CANCEL PRINT

A

TAG LIST TO PRINTER

A PRINT ALL TAGS IN RANGE
B PRINT ALL DEFINED TAGS IN RANGE
C PRINT ALL UNDEFINED TAGS IN RANGE
D ENTER START TAG
E ENTER END TAG
F ENTER SEARCH PATTERN

B

TREND LIST TO PRINTER

A LIST BY TAG NAME
B LIST BY TREND TYPE
C LIST BY COLLECTION MODE
D LIST BY TREND USAGE
E LIST BY MODULE ADDRESS
F LIST BY TAG TYPE
G ENTER START TREND INDEX
H ENTER END TREND INDEX

I

TREND PEN CLUSTER

A GENERAL PARAMETERS
B DEVICE DEFINITION
C DEVICE ASSIGNMENT

J

TAG HISTORIAN CONFIGURATION

A HISTORIAN GROUP CONFIGURATION
B TAG HISTORIAN RETRIEVAL → B
C PRINT/DELETE RDF

B

RETRIEVAL OPTIONS

A RETRIEVE BY ALARM GROUP
B RETRIEVE BY HISTORIAN GROUP
C RETRIEVE BY WILDCARD
D RETRIEVE BY INDIVIDUAL TAGS

NOTE: 1. Function not supported.

Figure B-6. Menu Structure - OIS Utilities (3 of 3)

APPENDIX C - PASSWORD SECURITY WORKSHEETS

INTRODUCTION

This section provides a set of worksheets that can be used to record the following information for password security.

Security level access rights	Table C-1
Security level region access	Table C-2
Window access	Tables C-3, C-4 and C-5
Display security level	Table C-6
User profiles	Table C-7

SECURITY LEVEL ACCESS RIGHTS WORKSHEET

Table C-1. Access Rights Worksheet

Security Level	Monitor	Configuration	Control	Tune	Alarm Management	Log Activation	Security Maintenance	Diag/Debug Terminal	Operation	Configuration Key Lock	Tune Key Lock
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											

SECURITY LEVEL REGION ACCESS WORKSHEET

Table C-2. Region Access Worksheet

User	Security	Database	Display	Keyboard	System	Log	Time/Date	Real Trend	Archive	Pens	PCU	Alarm Inhibit	Red Tag	Security Tag Groups															
														1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

WINDOW ACCESS WORKSHEET

Table C-3. Console Definition Worksheet

Console No.	Loop Address	Node Address	Console No.	Loop Address	Node Address
1			5		
2			6		
3			7		
4			8		

Table C-4. Logical CRT Definition Worksheet

Logical CRT No.	Console No.	Physical CRT No.	Logical CRT No.	Console No.	Physical CRT No.
1			9		
2			10		
3			11		
4			12		
5			13		
6			14		
7			15		
8			16		

NOTE: Physical CRT No. can be interpreted as window number.

Table C-5. Security Level Logical CRT Access Worksheet

Security Level	Logical CRT Access															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																

DISPLAY SECURITY LEVEL WORKSHEET

Table C-6. Display Security Level Worksheet

Display Name	Primary Level	Secondary Level	Display Name	Primary Level	Secondary Level

USER PROFILES WORKSHEET

Table C-7. User Profiles Worksheet

User	Security Level	User ID	Password	Red Tag Key	Logical CRT Access																Security Tag Groups Access															
					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

APPENDIX D - HARD DISK UTILIZATION

CALCULATIONS

An important consideration in the configuration process is the amount of hard disk space utilized. This appendix explains how to calculate the amount of space a configuration will occupy on the hard disk.

OIS Software and Tag Database

For calculations, the OIS software and tag database occupies approximately 100 megabytes.

Displays

The total amount of hard disk space to allocate for storage of displays varies. The minimum amount of space to allow should be one megabyte. To calculate the amount of hard disk space consumed by displays, use the formula:

No. of displays × average display size × retention factor

The amount of space needed for a single display varies from four kilobytes for a simple menu to up to 40-kilobytes for a complex display. The average display size depends on the complexity of the actual display files. Using a 40-kilobyte average display size calculates a ***worst case*** amount. The utilization example given later in this section uses a 15-kilobyte average display file size.

When transferring a display source file and its associated symbol source files (***DT***) to the console, they initially reside on the hard disk as PC (DOS) format files. They are then converted to an OpenVMS format and reside in ***DT*** files in another directory. After being converted, the original ***DT*** files in PC format still remain on the hard disk. After being converted, a display (***DT***) source file must be processed using the graphical display configuration tool. The result of this processing is a usable ***DU*** display file and associated ***DL*** symbol files. The ***DU*** and ***DL*** files occupy approximately the same amount of disk space as their corresponding ***DT*** source files.

All three sets of display files must be considered when calculating the amount of hard disk space consumed by displays. This determines the value used as the retention factor in the calculation. After a display is processed and has no errors, its associated ***DT*** files can be removed from the hard disk if desired. The retention factor should be one if only the ***DU***/***DL*** files are retained on the hard disk, two if ***DU/DL*** and one set of ***DT*** files, and three if ***DU/DL*** and both sets of ***DT*** files.

Trend Data

The minimum amount of space to allow is one megabyte. The maximum depends on the size of the hard disk drive and how much space is allocated for other functions.

Standard Trend

Use the following formula to calculate the disk space usage for a standard trend (normal and fast):

(numsamples × 4 bytes) + 48 bytes

where:

numsamples Total number of samples collected over a certain period of time. The number of samples is equal to:

time period ÷ collection resolution

Table D-1 gives examples of calculated trend data disk space utilization for standard trends.

Table D-1. Standard Trend Data Disk Space Utilization

Trend Type	Utilization per Trend (bytes)
15 second, trended over 1 day	23,040 + 48
15 second, trended over 2 days	46,080 + 48
15 second, trended over 3 days	69,120 + 48
15 second, trended over 4 days	92,160 + 48
15 second, trended over 5 days	115,200 + 48
15 second, trended over 6 days	138,240 + 48
15 second, trended over 7 days	161,280 + 48
1 minute, trended over 1 day	5,760 + 48
1 minute, trended over 2 days	11,520 + 48
1 minute, trended over 3 days	17,280 + 48
1 minute, trended over 4 days	23,040 + 48
1 minute, trended over 5 days	28,800 + 48
1 minute, trended over 6 days	34,560 + 48
1 minute, trended over 7 days	40,320 + 48

Intermediate time frames can be calculated by interpolation. For example, a 15-second trend, trended over eight hours:

$$\left(\frac{8}{24} \times 23040\right) + 48 = 7728 \text{ bytes}$$

500 one-minute trends, trended over seven days:

$$(500 \times 40320) + 48 = 20.2 \text{ Mbytes}$$

Enhanced Trend The disk space usage for enhanced trends depends on the variable being trended. Use one of the following formulas to calculate the disk space usage for an enhanced trend:

ANALOG	*(numevents × 12 bytes) + 48 bytes*
DADIG	*(numevents × 10 bytes) + 48 bytes*
DANG	*(numevents × 24 bytes) + 48 bytes*
Digital type	*(numevents × 10 bytes) + 48 bytes*
RMSC	*(numevents × 12 bytes) + 48 bytes*
STATION	*(numevents × 24 bytes) + 48 bytes*

where:

numevents Depends on the trend definition. The maximum number of events the console is to save on its hard disk is set during definition of a trend. Refer to ***Defining a Trend*** in Section 11 for further explanation.

Example: 100 ANALOG trends with 10,000 events saved per trend:

$$100 \times (10000 \times 12 + 48) = 12.0 \text{ M}$$

Log Data

Custom Logs Use the following formulas to calculate the disk space for custom log files:

1. Log definition files consume (approximately):

   [6.5 kbytes + (6 bytes × no. of columns) × no. of rows] × no. of logs defined

2. Log retention files consume (approximately):

   (no. of rows × 132 characters) × total no. of retentions defined for all logs

3. Log output files consume (approximately):

   (132 characters per row × no. of rows) × n

where:

n	Number of log output files. Ranges from one log output file to possibly 35 log output files per log (when logs are generated faster than the printer can handle).

4. Log data files consume (approximately):

(8 bytes per dynamic value cell defined + 1.5 kbytes) × n

where:

n	Number of log data files. Ranges from one data file to possibly 35 data files per log (when logs are generated faster than the printer can handle).

SOE Logs Use the following formulas to calculate the disk space for SOE log files:

1. Each data file consumes:

   8 bytes × no. of events

   Total = (8 × no. of events) × no. of SOE logs defined

2. Each output file consumes (approximately):

   132 characters × no. of events

3. Each SOE retention consumes (approximately):

   132 characters × no. of events

4. Each log is limited to nine retentions. Its data files may range from one to 35 files per SOE log, similar for its output files.

CONSIDERATIONS

The amount of hard disk space required by the console is dynamic. The amount of space needed at any given time depends on the operations or tasks the console is performing. Use the formulas in this appendix to calculate the requirements, then add the following percentages to accommodate for the dynamic needs of the console to insure optimum performance:

- 20 percent if configured for logging.
- 10 percent if not configured for logging.

UTILIZATION EXAMPLE

The following characteristics are used in this example:

- 667 displays (15 kilobytes average display file size, ***DU/DL*** and one set of ***DT*** files retained on the hard disk).
- 200 standard trends.
- 200 enhanced trends (100 ANALOG, 50 digital type, and 50 STATION).
- 500-megabyte formatted hard disk.

NOTE: The console can use a variety of different hard disk drives. The example uses a 500-megabyte hard disk drive for calculations.

Table D-2 shows the calculations and remaining unused disk space for this example utilization.

Table D-2. Example Utilization

Data		Utilization	Total Bytes
OIS software and tag database		100.0 M[1]	100.0 M
Displays		667 ¥ (15*k* ¥ 2) = 20.0 M	120.0 M
Trends[2]	Standard	200 at 15-sec resolution over 7 days = *32.3 M* 200 ¥ (161280 + 48) = 32.3 M	152.3 M
	Enhanced	200 with 10,000 events being saved per trend = *29.0 M* 100 ANALOG: 100 ¥ (10000 ¥ 12 + 48) = 12.0 M 50 digital type: 50 ¥ (10000 ¥ 10 + 48) = 5.0 M 50 STATION: 50 ¥ (10000 ¥ 24 + 48) = 12.0 M	181.3 M
		Unused disk space[3] =	318.7 M

NOTE:

1. Assumes a default database of 30,000 tags.
2. Depends on the types of trends being configured. Use the worst possible case for number of trends.
3. Available for other functions such as archiving and logging. Based on a 500-megabyte hard disk drive.

APPENDIX E - SYSTEM DISPLAYS

FILE NAMES

Table E-1 lists the file names for system displays (***DS*** and ***DU***).

Table E-1. System Display Files

Name	Description
ACCSTC	Alarm comments
ADCCONF	Automatic displays
ADSSTC	Annunciator display panels
AGCSTC	Alarm groups
ALMSUMFL	Alarm summary (full)[1]
ALMSUMP1	Alarm summary for priority one alarms (full size)[1]
ALMSUMP2	Alarm summary for priority two alarms (full size)[1]
ALMSUMP3	Alarm summary for priority three alarms (full size)[1]
ALMSUMP4	Alarm summary for priority four alarms (full size)[1]
ALMSUMP5	Alarm summary for priority five alarms (full size)[1]
ALMSUMP6	Alarm summary for priority six alarms (full size)[1]
ALMSUMP7	Alarm summary for priority seven alarms (full size)[1]
ALMSUMP8	Alarm summary for priority eight alarms (full size)[1]
ALRMFSTR	Alarm summary/operator assignable trend display[1]
AMCFUN	Alarm management
ANLGPRM	Operating parameters (ANALOG)
ANSWPRM	Operating parameters (INTANG)
APCSTC	Alarm priorities
APTFUN	User task menu
ARCFUN	Archival menu
ASFSTC	Alarm summary format menu
BLANK	Blank[1]
BLOCKDET	Block details
CALTEST	Touch screen calibration
CANPRNT	Cancel queued print
CDTSTC	Character definition menu
CELDEF	Cell definition (custom log)
CIUTASK	CIU task processing
CLRPAL	Color palette
COMMODD	Communication module details
CONFIGA	Module configuration menu
DADGPRM	Operating parameters (DADIG)
DANGPRM	Operating parameters (DANG)
DANLPRM	Operating parameters (DAANALG)

Table E-1. System Display Files (continued)

Name	Description
DBSFUN	Database configuration menu
DBSLIST	Database list menu
DDRPRM	Operating parameters (DD)
DEDSTC	Display generator
DEVSTAT	Operating parameters (DEVSTAT)
DEVSTAT1	Device status[1]
DEVSTAT2	Device status[1]
DGSWPRM	Operating parameters (INTDIG)
DGTLPRM	Operating parameters (DIGITAL)
DLG	Display log function
DMDSTC	Display mask definition
DSPFUN	Display configuration menu
DTRSTC	System display translator
ELFSTC	Event log format menu
EUDSTC	Engineering unit descriptors
EVERCTN	Event and error counters (nonressetable)
EVERCTR	Event and error counters (ressetable)
EVTLOG	Event log
FAILOVR	Peripheral failover assignment
FIRMWARE	Firmware revision[1]
FLTUNE	Station tuning display[1]
FSTRLS1	Operator assignable display (list 1)
FSTRLS12	Operator assignable display (list 2)[1]
GENFUN	General functions menu
HICSTAT	Tag historian menu
INDFUN	INFI-NET diagnostics menu
INHAGRP	Alarm group inhibit
INTAPO	User task options
INTAPT	User task definition
INTOCC	Operator configurable display parameters
INTPAS	Alarm summary report
INTPENA	Trend pen general parameters
INTPENB	Trend pen device definition
INTPENC	Trend pen device assignment
INTSOE	SOE log menu
INTSOEA	SER definition
INTSOEB	SOE report definition
INTTLP	Tag list to printer
INTXYC	XY plot definition
KEYMAC	Key macro definition
KTASTC	Function key assignment

Table E-1. System Display Files (continued)

Name	Description
KYBFUN	Keyboard menu
LCDSTC	Logical CRT definition
LCDSTC2	Console definition
LOGFUN	Logging menu
LOGPRM	Logging parameters
LOOPTOP	Loop topology report
LSDSTC	Logic state descriptors
LSTSTC	Log status display
MCBPRM	Operating parameters (RMCB)
MCFSTC	System configuration menu
MCSFUN	Configuration menu
MISCSTC	Miscellaneous functions
MON68K	Monitor 68K[1]
MPRSTC	Module problem report
MSDPRM	Operating parameters (MSDD)
MSTPRM	Operating parameters (N90STA)
MWCDEF	Multiwindow displays
N90STAT1	System status[1]
NODETOP	Node topology report
NODSTAXX	Node status[1]
OAD	OAS definition
OCDSTC	Operator configurable displays
OPASTRD	Operator assignable trends menu
OPASTRD2	Operator assignable trends menu[1]
OPRFUN	Operation function menu
ORAACTS	Operator request for action
ORAHALF	Operator request for action/operator information
ORAINFO	Operator information
OTASTC	Operator trend assignment lists
PCMSTC	Printer color maps
PCUFUN	PCU configuration menu
PLSSTC	PCU management menu
PRTASS	Printer assignments
RAEVENT	Retrieve events
RCMPRM	Operating parameters (RCM)
RCPRTLG	Retrieve logs
RDATVOL	Data type to volume definition
RDEFARC	Define archiving menu
RDIRMD	Directory of archive volume menu
RDIRRET	Directory of retrieved data menu
RELSTC	Relays

Table E-1. System Display Files (continued)

Name	Description
RERETTD	Retrieve trends
RGDIRTD	Directory of retrieved trends
RJUNUSD	Unused archive trend indices
RMCSTC	RMCB text
RMISC	Miscellaneous definitions
RMSCPRM	Operating parameters (RMSC)
RPCU	Retrieve PCU configurations
RPCUDIR	Directory of retrieved PCU configurations
RPCUDR	Directory of retrieved PCU configurations
RPTGEN	Log definition
RRETRIV	Retrieve archived data menu
RSTORE	Archival storage menu
RTAGS	Retrieve tag data
RTDIRTG	Directory of retrieved tags
RTTSTC	Red tag status
RUSENEW	Use new volume menu
RVOLMED	Define archiving
SATSTC	System attributes for foreign language assignment
SDESTC	Show display errors
SECFUN	Security maintenance menu
SECLEVL	Security levels
SECOPT	Security options
SETTIM	Set date and time
SMPLTUNE	Station tuning display[1]
STNPRM	Operating parameters (STATION)
SYSFUN	System configuration menu
TABTRDX	Tabular trend
TAGSTC	Tag
TDCSTC	Define trends
TDFSTC	Time and date format
TDPSTC	Trend list to printer
TERMEM	Terminal emulation
TEXTSEL	Tag text selector
TONESTC	Tones
TPCMENU	Trend pen cluster menu
TSCSTC	Text substitution
TSTRPRM	Operating parameters (TEXTSTR)
TSUSTC	Tag summaries
TXTPRM	Operating parameters (TEXT)
UNDFTAG	Operating parameters (UNDEF)
UPDSTC	User password definition

Table E-1. System Display Files (continued)

Name	Description
UTLFUN	Utility functions menu
XCPSTAT	Exception statistics
XY1A	XY plot (MFC)[1]
XYPFL5	XY plot (tag or trend)[1]

NOTE:

1. Exists as a ***DU*** file. The security level of the display can be changed using the *Display Mask Definition* option.

APPENDIX F - XY PLOT SOURCE FILE STRUCTURE

MFC DATA SOURCE FILE LOCATION AND NAMING

The data for an MFC data source XY plot comes from a file created in a multi-function processor (MFP) module. The file is converted to a model one, mode two float data file; refer to ***DATA FILE STRUCTURE*** in this section. This converted file is saved to the console hard disk.

A data file resides in the [DATA.USN59] directory and is in the format:

XYxxnnnn.IF

where:

xx	Plot index number.
nnnn	File ID as defined in the plot definition.

The XY plot data file in an MFP module read by the console must be created using a C program with the structures defined in the ***xymfcfil.h*** header file. The C utility program (CUP) downloads this program to the MFP module. A CUP version 1.0.022 or later program is required to communicate with the MFP module. For information about the CUP program, refer to the ***C Utility Program*** instruction (Table 1-2 lists instruction numbers).

> **NOTE:** When configuring the CUP program, make the number of module bus file (MBF) buffers equal to the number of XY plot files to transfer and the size of each MBF buffer equal to the size of the largest XY plot file.

DATA FILE STRUCTURE

There are two file models, each having two modes of operation.

Model 1 - alarm state information for each data point is provided in a character bit stream. Only a high or low alarm condition can be represented.

Model 2 - alarm state information for each data point is provided in a character array. Alarm conditions are defined as stated in ***mcs$include:alarm_states.h***.

Mode 1 - x data values (data sets) repeated only, with y values determined by the file position of x data values.

- ***or*** -

Y data values (data sets) repeated only, with x values determined by the file position of y data values.

Mode 2 - both x and y data values (data sets) are repeated.

A complete file structure for each model and mode consists of the following elements in the order shown:

- Standard header.
- Padded memory:

  8 bytes for model 2, mode 1.
  12 bytes for model 1, mode 1.
  16 bytes for model 2, mode 2.
  24 bytes for model 1, mode 2.

- Data set.

NOTE: Each element must be the same model and mode.

Example: The following table shows the file structure for model 1, mode 1:

Header
12 bytes
Data set

The sections that follow show the required data file structures (standard header and data sets) when creating an MFC data source file.

Reference Lines

The following are specific to reference lines:

1. The *x_refline* is for a horizontally drawn reference line. The *y_refline* is for a vertically drawn reference line.

2. An alarm bar is always drawn from an x or y point of origin to reference line 0.

Standard Header Element for All File Models and Modes

```
#include "mcs$include:quality.h" /* include QGOOD and QBAD as          */
                        /* quality indicators                 */

#define NUM_SPARE           6            /* number of spare bytes in ext headers  */
#define MAX_REFLINES        5            /* number of x/y reference lines        */
#define CHAR_BQ             0x80         /* char bad quality value              */
#define SHORT_BQ            0x8000       /* short bad quality value             */
#define FLOAT_BQ            0x80000000   /* float bad quality value             */
#define NONE_INSERTED       0
#define X_INSERTED          1
#define Y_INSERTED          2

#define CHAR_ELEM           0x00         /* char data element type                   */
#define SHORT_ELEM          0x01         /* short data element type                  */
#define FLOAT_ELEM          0x02         /* float data element type                  */
#define REAL2_ELEM          0x03         /* N90 real 2 element type                  */
#define MODEL_1             0x00         /* file Model 1                             */
#define MODEL_2             0x10         /* file Model 2                             */
#define MODE_1X             0x00         /* file Mode 1, x data values only          */
#define MODE_1Y             0x01         /* file Mode 1, y data values only          */
#define MODE_2              0x02         /* file Mode 2, x and y data values         */
#define MDL1MOD1X           0x00         /* Model 1, Mode 1, x data values only      */
#define MDL1MOD1Y           0x01         /* Model 1, Mode 1, y data values only      */
#define MDL1MOD2            0x02         /* Model 1, Mode 2, x and y data values     */
#define MDL2MOD1X           0x10         /* Model 2, Mode 1, x data values only      */
#define MDL2MOD1Y           0x11         /* Model 2, Mode 1, y data values only      */
#define MDL2MOD2            0x12         /* Model 2, Mode 2, x and y data values     */


/*------- Standard Header information for all File Models and Modes ---------*/
typedef struct {
  unsigned char elem_datatype;  /* element data type          */
                        /*   0x00 = char          */
                        /*   0x01 = short         */
                        /*   0x02 = float         */
                        /*   0x03 = REAL2         */

  unsigned char file_MdlMod;        /* file Model and Mode                */
                                    /* 0x00 Model 1, Mode 1, x data values only */
                                    /* 0x01 Model 1, Mode 1, y data values only */
                                    /* 0x02 Model 1, Mode 2, x and y data values*/
                                    /* 0x10 Model 2, Mode 1, x data values only */
                                    /* 0x11 Model 2, Mode 1, y data values only */
                                    /* 0x12 Model 2, Mode 2, x and y data values*/

  unsigned short num_datasets;      /* Number of x elements, y elements,      */
                                    /* or xy element data sets in the file.   */
  unsigned char x_data_stat;        /* x Data Status           */
                                    /*   0x00 = good quality       */
                                    /*   0x01 = bad quality        */

  unsigned char y_data_stat;        /* y Data Status           */
                                    /*   0x00 = good quality       */
                                    /*   0x01 = bad quality        */

  unsigned char x_eng_unit;  /* Engineering Unit            */
  unsigned char y_eng_unit;  /* Engineering Unit            */

  float x_average;        /* Average x data value         */
  float y_average;        /* Average y data value         */

  float x_high_lim;       /* X high limit                 */
  float y_high_lim;       /* Y high limit                 */
  float x_low_lim;        /* X low limit                  */
  float y_low_lim;        /* Y low limit                  */
```

Standard Header Element for All File Models and Modes *(continued)*

```
float       x_refline[MAX_REFLINES];    /* X HORIZONTAL reference lines */
float       y_refline[MAX_REFLINES];    /* Y VERTICAL reference lines   */

unsigned char x_refline_drw[MAX_REFLINES]; /* X reference line draw flag   */
unsigned char y_refline_drw[MAX_REFLINES]; /* Y reference line draw flag   */

unsigned char draw_step;            /* 1 if to draw in step graph format */
                                    /* 0 connect the dots               */

unsigned char val_inserted;         /* 0 no values inserted */
                                    /* 1 x values inserted  */
                                    /* 2 y values inserted  */
unsigned char spare[NUM_SPARE];

                                    /* THE MODEL STRUCTURE FOLLOWS THE HEADER AND ...  */

                                    /* PROFILE DATA GOES AFTER THE MODEL STRUCTURE     */

} XYMFC_HEADER;
```

Data Sets for Model 1, Mode 1

```
/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 1, Mode 1, char data type ----------------------*/
typedef struct {
   unsigned char *alm_Hstate;   /* bit stream (num_datasets / 8) bits long   */
   unsigned char *alm_Lstate;   /* bit stream (num_datasets / 8) bits long   */

   char          *data;         /* num_datasets data value elements          */
} MDL1MOD1_DATASET_C;

/*--------------------------- Data Set --------------------------------------*/
/*------------ File Model 1, Mode 1, short/REAL2 data type ------------------*/
typedef struct {
   unsigned char *alm_Hstate;   /* bit stream (num_datasets / 8) bits long   */
   unsigned char *alm_Lstate;   /* bit stream (num_datasets / 8) bits long   */

   short         *data;         /* num_datasets data value elements          */
} MDL1MOD1_DATASET_S;

/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 1, Mode 1, float data type ---------------------*/
typedef struct {
   unsigned char *alm_Hstate;   /* bit stream (num_datasets / 8) bits long   */
   unsigned char *alm_Lstate;   /* bit stream (num_datasets / 8) bits long   */

   float         *data;         /* num_datasets data value elements          */
} MDL1MOD1_DATASET_F;
```

Data Sets for Model 1, Mode 2

```
/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 1, Mode 2, char data type ----------------------*/
typedef struct {
   unsigned char *x_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *x_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */

   char        *x_data;      /* num_datasets x data value elements       */
   char        *y_data;      /* num_datasets y data value elements       */
} MDL1MOD2_DATASET_C;

/*--------------------------- Data Set --------------------------------------*/
/*------------ File Model 1, Mode 2, short/REAL2 data type ------------------*/
typedef struct {
   unsigned char *x_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *x_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */

   short       *x_data;      /* num_datasets x data value elements       */
   short       *y_data;      /* num_datasets y data value elements       */
} MDL1MOD2_DATASET_S;

/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 1, Mode 2, float data type ---------------------*/
typedef struct {
   unsigned char *x_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *x_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Hstate; /* bit stream (num_datasets / 8) bits long   */
   unsigned char *y_alm_Lstate; /* bit stream (num_datasets / 8) bits long   */

   float       *x_data;      /* num_datasets x data value elements       */
   float       *y_data;      /* num_datasets y data value elements       */
} MDL1MOD2_DATASET_F;
```

Data Sets for Model 2, Mode 1

```
/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 2, Mode 1, char data type ----------------------*/
typedef struct {
  unsigned char *status;     /* num_datasets status bytes                 */

  char         *data;       /* X or Y data value elements                */
} MDL2MOD1_DATASET_C;

/*--------------------------- Data Set --------------------------------------*/
/*------------ File Model 2, Mode 1, short/REAL2 data type ------------------*/
typedef struct {
  unsigned char *status;     /* num_datasets status bytes                 */

  short        *data;       /* X or Y data value elements                */
} MDL2MOD1_DATASET_S;

/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 2, Mode 1, float data type ---------------------*/
typedef struct {
  unsigned char *status;     /* num_datasets status bytes                 */

  float        *data;       /* X or Y data value elements                */
} MDL2MOD1_DATASET_F;
```

Data Sets for Model 2, Mode 2

```
/*--------------------------- Data Set ------------------------------------*/
/*--------------- File Model 2, Mode 2, char data type ----------------------*/
typedef struct {
   unsigned char *x_status;     /* num_datasets status bytes              */
   unsigned char *y_status;     /* num_datasets status bytes              */

   char         *x_data;       /* X or Y data value elements             */
   char         *y_data;       /* X or Y data value elements             */
} MDL2MOD2_DATASET_C;

/*--------------------------- Data Set --------------------------------------*/
/*------------ File Model 2, Mode 2, short/REAL2 data type ------------------*/
typedef struct {
   unsigned char *x_status;     /* num_datasets status bytes              */
   unsigned char *y_status;     /* num_datasets status bytes              */

   short        *x_data;       /* X or Y data value elements             */
   short        *y_data;       /* X or Y data value elements             */
} MDL2MOD2_DATASET_S;

/*--------------------------- Data Set --------------------------------------*/
/*--------------- File Model 2, Mode 2, float data type ---------------------*/
typedef struct {
   unsigned char *x_status;     /* num_datasets status bytes              */
   unsigned char *y_status;     /* num_datasets status bytes              */

   float        *x_data;       /* X or Y data value elements             */
   float        *y_data;       /* X or Y data value elements             */
} MDL2MOD2_DATASET_F;
```

Index

Index (continued)

O

P

Index (continued)

W

X

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 WBPEEUI220757A1 Litho in U.S.A. 497