

Find Names Command

SW11-508

Find Names Command

**SW11-508
Release 500**

12/95

Copyright, Notices, and Trademarks

Printed in U.S.A. – © Copyright 1995 by Honeywell Inc.

Revision 01– 12/95

While this information is presented in good faith and believed to be accurate, Honeywell disclaims the implied warranties of merchantability and fitness for a particular purpose and makes no express warranties except as may be stated in its written agreement with and for its customer.

In no event is Honeywell liable to anyone for any indirect, special or consequential damages. The information and specifications in this document are subject to change without notice.

TotalPlant is a registered trademark of Honeywell Inc.

TDC 3000 is a trademark of Honeywell Inc.

Other trademarks (if any) that appear in this document are used only to the benefit of the trademark owner, with no intention of trademark infringement. A list of trademarks is available; contact us at the address below.

The following standard document conventions may be used in this document. Check Section 1 for conventions unique to this document.

Item	Convention	Meaning
<KEYCAPS>	Pointed brackets around a term	A key on a computer keyboard
[TARGET]	Square brackets around a term	For touch screen monitors, an area that can be touched to invoke a system action
entry data	Courier bold text	Data you enter into the system
screen/program data	Courier text	Screen or program data

Honeywell
Industrial Automation and Control
Automation College
2820 West Kelton Lane
Phoenix, Arizona 85023-3028

(602) 313-5669

Table of Contents

SECTION 1—INTRODUCTION	1
1.1 About this Document	
Basics.....	1
Document background.....	1
About this document	1
Prerequisite skills	1
1.2 Find Names Concept	
Definition	2
Purpose.....	2
1.3 References	
Additional information.....	3
SECTION 2—FIND NAMES USES	5
2.1 Overview	
Introduction	5
Typical uses	5
Types of searches.....	5
2.2 Application Examples	
Introduction	6
Point deletion	6
Point modification.....	6
Documentation needs	6
Identify resources being used and resources available	6
Locate program references	7
Multiple references.....	7
Locate entity references.....	7
Why look for untagged references?	7
Why look for system references?	7
Locate fragmented points.....	8
2.3 Search Concepts	
Introduction	9
Overall approach.....	9
Where to send the data.....	9
What's next	9
SECTION 3—NAVIGATION PATHS AND ENTRY CHOICES	11
3.1 Navigation Overview	
Begin with the Command Processor display.....	11
Example of a Find Names search.....	11
Navigation paths from Find Names display.....	11
Navigation to Find Names Menu, part 1.....	12
Navigation to Find Names Menu, part 2.....	12
Navigation to Find Names Menu, part 3.....	13
Points built versus entities.....	13
3.2 Navigation to UCN Checkpoint Entity References	
Quick overview.....	15
Navigation Path to UCN Checkpoint (Entity References)	15

Table of Contents

3.3 UCN Checkpoint Display Orientation	
Introduction	17
UCN Checkpoint Display Interpretation	17
3.4 Entry Choices for UCN Checkpoint Files	
Introduction	18
Explanation of choices	18
Checkpoint (Entity References) Port Description	18
.....	19
Valid entries for "Slots to search?"	20
3.5 Navigation to UCN Checkpoint PM Sequences	
Introduction	22
Navigation Path to UCN Checkpoint PM Sequences (Program References)	22
3.6 Navigation to UCN Checkpoints for Both	
Introduction	24
Navigation Path to UCN Checkpoint for Both (Program References)	24
3.7 Example Search (Peer-to-Peer)	
Search description, example	25
Example search	25
Example Node-to-Node (Peer-to-Peer) search, part 1	25
Example Node-to-Node (Peer-to-Peer) search, part 2	26
3.8 Navigation to History Groups	
Introduction	27
Navigation to History Groups	27
3.9 Entry Choices for History Groups	
Introduction	28
Meaning of entries	28
3.10 Navigation to Picture Files	
Introduction	29
Navigation to Picture Files	29
3.11 Entry Choices for Picture Files	
Introduction	30
Meaning of entries	30
3.12 Navigation to Area Database Files (Schematics)	
Introduction	31
Navigation to Area Database Files (Schematics) figure	31
3.13 Entry Choices for Area Database Files (Schematics)	
Meaning of entries	32
3.14 Navigation to Area Database Files (Entities)	
Introduction	33
Navigation to Area Database Files (Entities) figure	33
3.15 Entry Choices for Area Database Files (Entities)	
Meaning of entries	34
Area database entities to search	34

Table of Contents

3.16	Items That Cannot Be Found	
	Source files missing information	36
	Finding instances of a deleted entity.....	36
SECTION 4—FIND NAMES EXAMPLES		37
4.1	Find Names Command Line	
	Introduction	37
	Search syntax	37
	Command Line Format	37
4.2	Example Search From Command File	
	Peer search example	38
	Example command file.....	38
SECTION 5—FIND NAMES ERROR MESSAGES		41
5.1	UCN Checkpoint Search Errors	
	Introduction	41
5.2	UCN Sequence Program Search Errors	
	Introduction	42
SECTION 6—USER REFERENCE INFORMATION		43
6.1	Object, Reference, and File Relationships	
	Finding connections between objects	43
	Files searched by Find Names.....	43
6.2	Using Find Names Displays	
	Entries on search-narrowing displays	45
	Organizational Summary menus.....	45
6.3	Valid Entries for Search-Narrowing Ports	
	Entity-related ports	48
	AM-related ports.....	48
	CM-related ports	48
	HG-related ports.....	49
	UCN-related ports	49
	Picture Editor-related ports	51
	Text File-related ports	52
	Area-related ports	53
	History-related ports.....	53
6.4	Special Objects for UCN Searches	
	Introduction	54
	Purpose.....	54
	Suppressing output columns.....	54
	Using wildcard characters	54
	Output for entities not on the UCN	54
	Output for a Reference Node object	55
	Output for a Reference Module object	55
	Output for a Reference Slot object.....	55
	Interpreting Reference Slot outputs	55

Table of Contents

Find Names Command

Section 1—Introduction

1.1 About this Document

Basics

Purpose	Describes the Find Names command.
Intended audience	Operators, maintenance technicians, or engineers who need to search files on media (HM or cartridge, for example) for references between entities on the UCN, LCN, or Hiway. This function can be used with the Documentation Tool.
For product release	TDC 3000 ^X Release 410 through 500.

Document background

This section

- Defines the Find Names command
- Describes prerequisite skills
- Describes find Names concepts
- Gives references to related documents

Other sections of this document provides user reference information including:

- Typical uses of Find Names
- Types of Find Names searches
- Application examples
- Search concepts
- Relationship of Find Names to Documentation Tool

About this document

This document discusses using Find Names for a UCN node, such as the Advanced Process Manager. Concepts discussed herein also apply to the Process Manager and Logic Manager unless otherwise noted.

Prerequisite skills

This document assumes the reader is familiar with point configuration and LCN database-building tasks.

1.2 Find Names Concept

Definition The Find Names command is a TDC 3000^X Engineering Personality function that allows you to search files from the History Module (NET) and removable media, including cartridges (\$Fn), or floppy disks (\$Fn), for references between entities on the UCN, LCN, or Data Hiway.

Find names does not search through the databases in LCN nodes and process-connected devices. For more information see the *Documentation Tool* manual.

ATTENTION

Note that searches through Picture Editor (schematic or custom display) data and Button Configuration data are searches of your source files. Picture Editor and Button Configuration object files cannot be searched.

Purpose Once retrieved, the data can be displayed as a report and printed.

1.3 References

Additional information

Document access and listing

Throughout this document, references to other documents are made by title only.

- To access documents provided electronically, select the document title in the main topics list (Select Main from the Topics button).
- To access anything in the paper documentation, find the title in the following document listing, and use the binder number provided there.

Table 1-1 Document Listing

Document Title	Binder Number	Binder Name
<i>Documentation Tool</i>	TDC 3030-1	Implementation/Startup & Reconfiguration-1
<i>Command Processor Operation</i>	TDC 3030-1	Implementation/Startup & Reconfiguration-1

Section 2—Find Names Uses

2.1 Overview

- Introduction** This section provides an overview of
- Typical uses of Find Names
 - Types of Find Names searches
 - Application examples
 - Search concepts
 - Relationship of Find Names to Documentation Tool
- Typical uses** The following are typical uses of Find Names when managing UCN database:
- Check for database connections before deleting a point.
 - Check for database connections before modifying a point.parameter.
 - Identify resources currently being used.
 - Locate program references.
 - Locate multiple references.
 - Locate entity references.
 - Untagged or hardware references
 - Point.parameter references
 - System references
 - Locate fragmented points.
- Types of searches** Users have the following search options when using Find Names:
- Checkpoints and CL programs
 - History group definition
 - Area database files
 - Picture files (schematics) and Free Format Logs
 - Button files
 - User text files
 - or, search all of the above.

2.2 Application Examples

- Introduction** This section provides some application examples of Find Names searches. Later in this document you will see how to use the Find Names displays to search the database. The examples that follow state a typical use and the rationale for it.
- Point deletion** An engineer, for example, desires to know what impact deleting a point will have on the database. Because the TDC 3000^X system uses an “internal reference” to that point in displays, groups, etc., it’s important to know ahead of time what the database connections are before you delete a point.
- In some cases, deleting a point may affect numerous database connections, so your search could include
- Checkpoints (UCN, CG, and AM) and CL programs
 - History group definitions
 - Area database files (for example, operating groups)
 - Picture files (schematics, subpictures) and Free Format Logs
 - Button files (for example, a button that calls up the point detail display)
 - User text files (includes CL and user files.)
- Point modification** For example, say you modified a point parameter’s alarm type from deviation alarms to rate of change alarms. You may want to search the picture files to locate references to the previous alarm configuration.
- Documentation needs** An engineer desires to document what has been built. An example is to get a listing of points built by point type. From Find Names you can get a listing of what is used by slot number. You can then take the point listing to Documentation Tool and sort the points in alphabetical order. See the *Documentation Tool* document for additional information.
- Identify resources being used and resources available** You can get a listing of what resources have been used in your database, and therefore determine remaining available resources. This includes
- slots in a controller
 - point connections
 - point positions in an operating group
 - points built in a particular module or by type, such as:
 - PMM (Process Manager Module)
 - IOP (Input/Output Processor)
 - Digital composites

- Regulatory control

Locate program references

Find Names searches programs, for example, to see what point (entity) references, messages, variables, elements of an array, are used. These searches can be performed on both the source and object files.

Multiple references

For example, process points may have inadvertently been built that use the same output hardware resource. When that happens, the control point could oscillate, because two points are trying to drive the same output hardware. Find Names can search for those hardware references so that you can identify the offending reference.

Locate entity references

Entity references consist of three types:

1. Untagged or hardware references
2. Point.parameter references (your most common search)
3. System references

Why look for untagged references?

Untagged references are important to look for before replacing IOP hardware. Some sites use untagged references in their control connections.

If your site uses untagged connections, a good practice is to use Find Names to look for any untagged references before replacing an IOP.

Another reason to check for untagged references is to see if that untagged connection has already been configured before you assign a “new” point to the slot.

Also, because CL programs permit the use of DI hardware references, use Find Names to search CL/PM program listings and review what CL programs are currently loaded and running before replacing a DI IOP.

Why look for system references?

System references are used when a point name (tagname) cannot be built against the resource (such as a string or time variable). Knowing system format (syntax) helps you locate resources faster. System references always begin with a dollar sign (\$). Some examples of system references follow:

- Node and node specific references, for example:
 - \$NM03N09 (UCN network 3, node 9)
 - \$NM03B09 (UCN network 3, node 9, node specific)
- Area database references, for example:
 - \$Ogroup(400), for operating group 400

Find Names Uses

2.2 Application Examples

Locate fragmented points

Fragmented points, also called “phantom” or “ghost” points, are points that are typically deleted at the NIM level only, but maintain their connections at the PM level. This can have the undesirable effect of impacting control strategies. Some effects include the possibility that the fragmented point continues to control. Other effects include the possibility that other new points cannot be loaded into the fragmented point’s slot number.

2.3 Search Concepts

- Introduction** This section introduces approaches and techniques used during Find Names searches. Your approach may vary from the ones listed here.
- Overall approach** Note that a Find Names search is performed on a database file. While node data is returned, it is a file search not a node query. This has led users to adopt typical approaches in doing a search, such as
- Copy the file to be searched to local removable media. This helps reduce the HM load. Additionally, this prevents the possibility of accidentally corrupting your master file.
 - If removable media is not available, copy the file to a user directory on the HM and perform the search.
 - If a checkpoint is searched, a good procedure is to demand a checkpoint before the Find Names search to ensure that the file has the most current data. (Usually you need to demand a checkpoint if you are searching database during the implementation phase. If a system is up and running, the checkpoint on the system is in all likelihood current.)
- Where to send the data** Users have several options of where to send the data as a result of a Find Names search. The Find Names results can be sent to
- a screen display,
 - a printer,
 - a text file (defined with a dataout, DO, command), or a
 - a virtual printer (specified with a dataout, DO, command).
- The rationale behind sending the data to text files is that the Find Names results can then be sorted or filtered (if necessary) with the Documentation Tool.
- What's next** The next section presents an overview of the navigation paths you can take on a Find Names search.

Find Names Uses
2.3 Search Concepts

Section 3—Navigation Paths and Entry Choices

3.1 Navigation Overview

Begin with the Command Processor display

The Find Names command is invoked through the Command Processor display. You can call up this display by selecting [COMMAND PROCESSOR] on the Engineering Main Menu.

You can also “escape” to the Command Processor display from any Engineering Personality activity except Logic Blocks. To do so, press the <ESC> key. When you are finished with Find Names, you can return to the activity you were using by holding <CTL> and pressing <HELP>.

Example of a Find Names search

ATTENTION

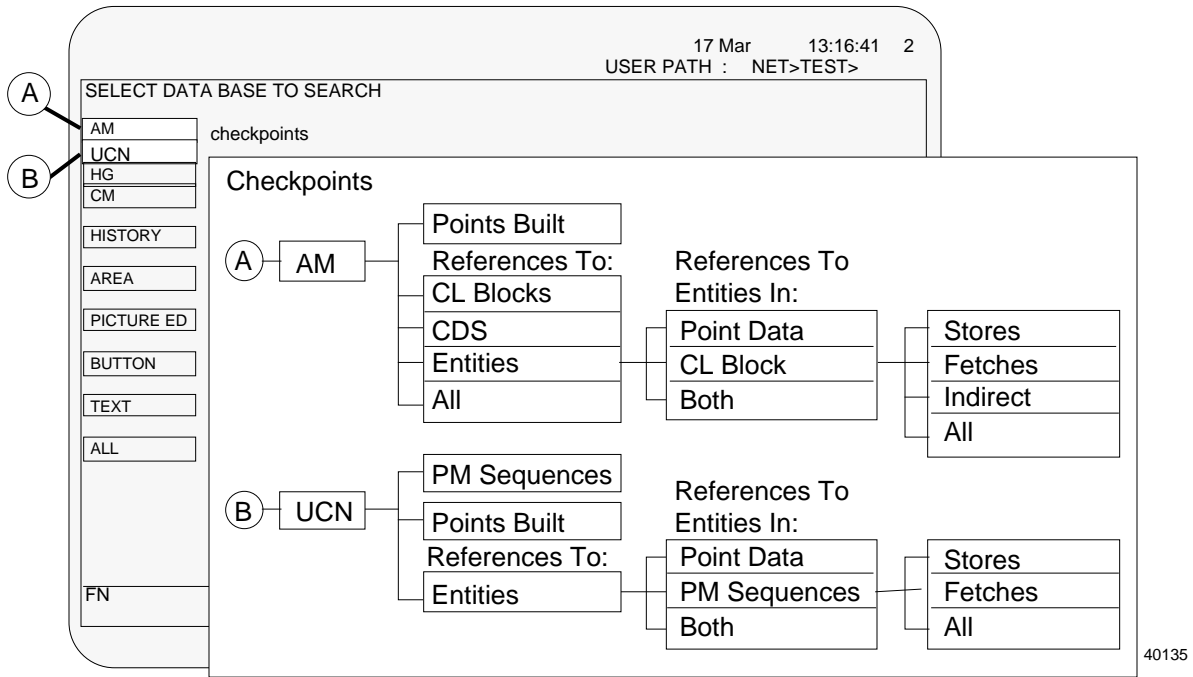
If you are willing to allow full searches of the entire selected database, you can simply press <ENTER> as soon as the search-narrowing display appears.

The Find Names function is a pattern checker, only. User inputs are not validated by the system; if you input the wrong information there are no prompts to help you correct the input syntax.

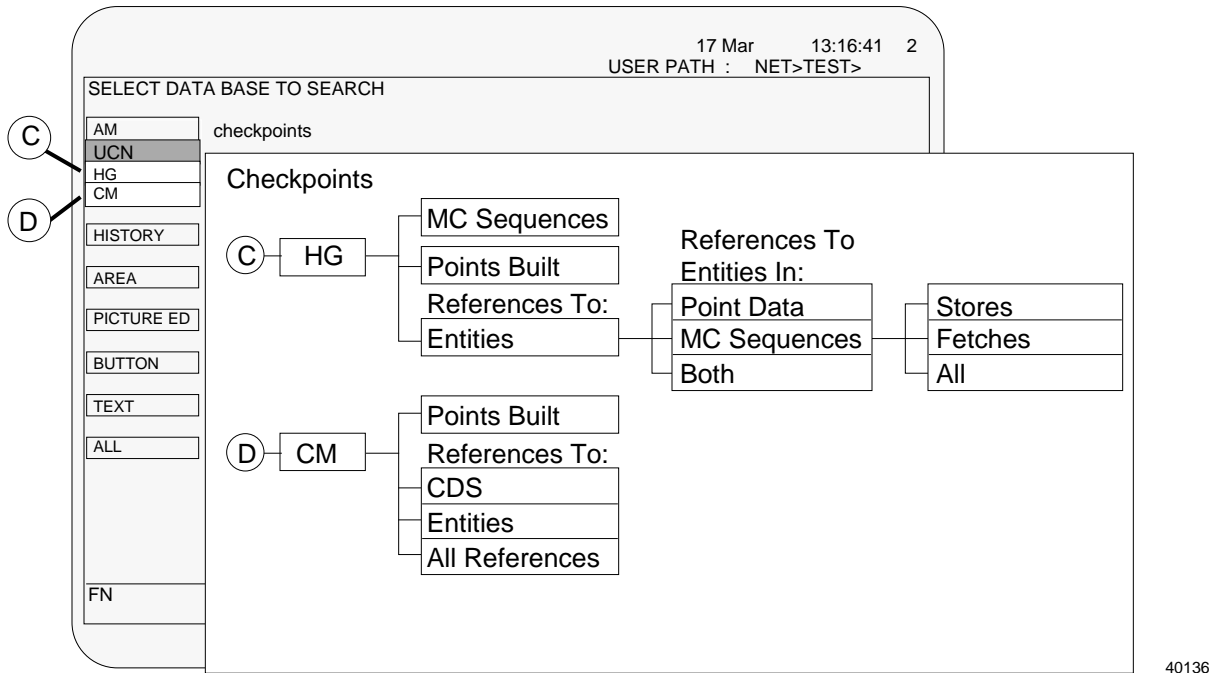
Navigation paths from Find Names display

The navigation paths from the Find Names display to the Find Names menus are shown in the following figure, which is presented in three parts.

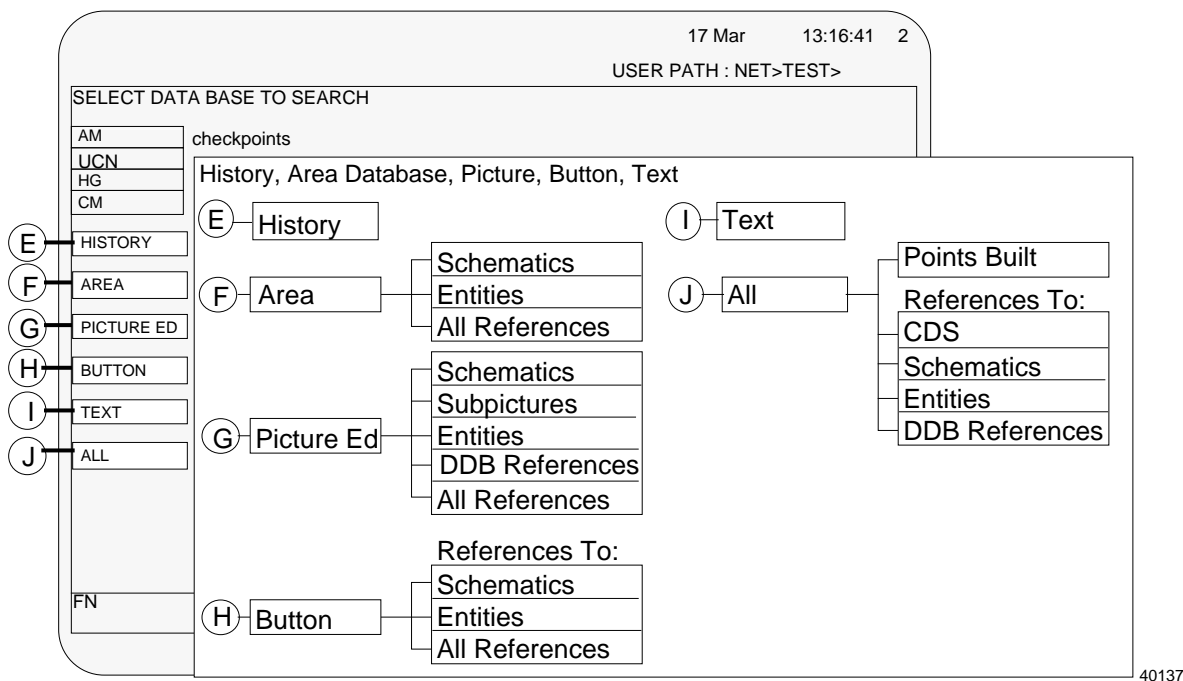
Navigation to Find Names Menus, part 1



Navigation to Find Names Menus, part 2



Navigation to Find Names Menus, part 3



Points built versus entities

Did you ever wonder what the differences are between the “points built” and “references to entities” selections and end up on an undesired path? There’s an easy way to remember the difference:

- Points built gives you a listing of the “points built” in a node or module. For example, the following points built could appear in a list:
 - TIC100
 - TIC200
 - TIC300
- References to entities gives you a listing of points and points.parameters, where the points are those found to reference the points.parameters you specified.
- For example, the following are what might appear in a list of points and the points.parameters they are found to reference:
 - FIC100 TIC100.PV
 - FIC200 TIC200.PV
 - FIC300 TIC300.PV

Navigation Paths and Entry Choices

3.1 Navigation Overview

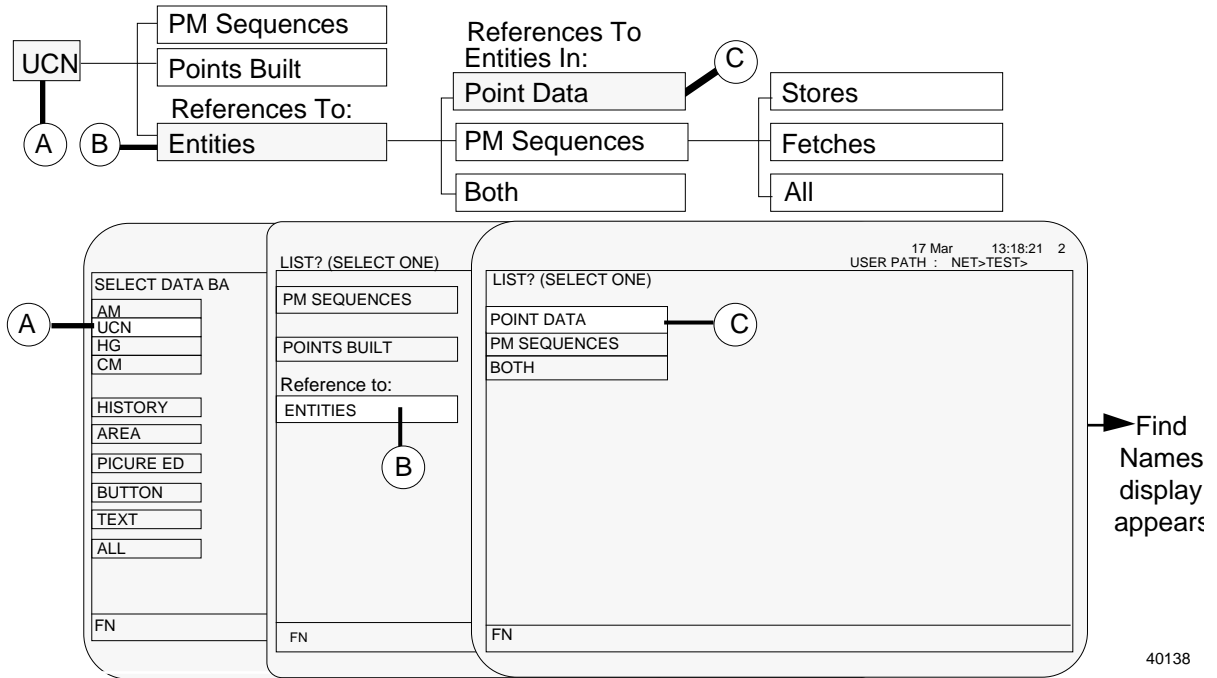
ATTENTION

Find Names does not support Logic Manager checkpoint searches.

3.2 Navigation to UCN Checkpoint Entity References

Quick overview The path to UCN checkpoint data is shown in the following figure, which is presented in two parts.

Navigation Path to UCN Checkpoint (Entity References)



Navigation Paths and Entry Choices

3.2 Navigation to UCN Checkpoint Entity References

17 Mar 13:19:11 2
 USER PATH : NET>TEST>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of "*" (ALL)

ENTITY REFERENCES
 NODE REFERENCES
 MODULE REFERENCES
 SLOT REFERENCES

UCN checkpoints to search?
 UCN NODES to search?

PROCESS MODULES to search?
 SLOTS to search?
 ENTITIES to search?

Where is the data base?
The pattern specified will be used for the ucn ckpt and pm sequence objects

SUPPRESS columns of output? NO YES

FN	UCN_CP	03	NODE_CP	05	MODULE	*	SLOT	*	ENTITY	*	ENT_REF	FVL21000.*	RN	*	REF_MOD	*	REF
<u>UCNCP</u>	<u>NODE</u>	<u>MODULE</u>	<u>SLOT</u>	<u>ENTITY</u>	<u>ENT_REF</u>	<u>RN</u>	<u>REF_MOD</u>	<u>REF_SL</u>									
3	5	APMM	LG1	SETNK000	FVL21000.D1	5	APMM	DC(1)									
3	5	APMM	LG1	SETNK000	FVL21000.I0	5	APMM	DC(1)									

FIND NAMES COMPLETE

40139

3.3 UCN Checkpoint Display Orientation

Introduction

An overview of your checkpoint choices is shown in following figure. The display can be thought of as having three parts:

1. What references am I looking for? These entries are asking for an entity.parameter reference.
2. What files do I search? These entries ask you what files (in this example, checkpoints) to search.
3. Where am I going to look for the reference? This entry means where is that reference residing or connected (PMM, APMM, or point type.)

UCN Checkpoint Display Interpretation

17 Mar 13:19:11 2
USER PATH : NET>TEST>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of "*" (ALL)

ENTITY REFERENCES

NODE REFERENCES

MODULE REFERENCES

SLOT REFERENCES

UCN checkpoints to search?

UCN NODES to search?

PROCESS MODULES to search?

SLOTS to search?

ENTITIES to search?

Where is the data base?

The pattern specified will be used for the ucn ckpt and pm sequence objects

SUPPRESS columns of output? NO YES

FN	UCN_CP	03	NODE_CP	05	MODULE	* SLOT	* ENTITY	* ENT_REF	FVL21000.*	RN	* REF_MOD	* REF
<u>UCNCP</u>	<u>NODE</u>	<u>MODULE</u>	<u>SLOT</u>	<u>ENTITY</u>	<u>ENT_REF</u>	<u>RN</u>	<u>REF_MOD</u>	<u>REF_SL</u>				
3	5	APMM	LG1	SETNK000	FVL21000.D1	5	APMM	DC(1)				
3	5	APMM	LG1	SETNK000	FVL21000.I0	5	APMM	DC(1)				

FIND NAMES COMPLETE

40139

3.4 Entry Choices for UCN Checkpoint Files

Introduction How you use the Find Name selections determines how broad or narrow your search (scope of search) becomes from the Find Names display.

Explanation of choices The following figure shows the screen displayed after following the navigation path shown in Navigation Path to UCN Checkpoint (Entity References), page 15.

Descriptions of the UCN checkpoint port entries shown below are given in , page 19.

Note that additional ports appear depending on the path you chose for your search. For examples, see:

Navigation Path to UCN Checkpoint PM Sequences (Program References), page 22

Navigation Path to UCN Checkpoint for Both (Program References), page 24.

Checkpoint (Entity References) Port Description

17 Mar 13:19:11 2
 USER PATH : NET>TEST>

1 Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of " * " (ALL)

2 ENTITY REFERENCES FVL21000.*

3 NODE REFERENCES *

4 MODULE REFERENCES *

5 SLOT REFERENCES *

5 UCN checkpoints to search? 03

6 UCN NODES to search? 05

8 PROCESS MODULES to search? *

9 SLOTS to search? *

9 ENTITIES to search? *

10 Where is the data base? DEFAULT NET LOCATION

13 The pattern specified will be used for the ucn ckpt and pm sequence objects OTHER

14 SUPPRESS columns of output? NO YES

Find Names display

Find Names output

UCNCP	NODE	MODULE	SLOT	ENTITY	ENT_REF	RN	REF_MOD	REF_SL
3	5	APMM	LG1	SETNK000	FVL21000.D1	5	APMM	DC(1)
3	5	APMM	LG1	SETNK000	FVL21000.I0	5	APMM	DC(1)

FIND NAMES COMPLETE

40141

The following table describes the meaning of the port entries for the figures UCN Checkpoint Display Interpretation, page 17, Checkpoint (Entity References) Port Description, page 18, and Navigation Path to UCN Checkpoint PM Sequences (Program References), page 22.

Table 3-1 Description of Port Entries

Item	Port Description	Valid Entries	Scope of search/output
1	ENTITY REFERENCES	*, ?, for all or part of pattern to match. EXAMPLES: TIC1??.*, TIC*.*, !DI??S??.*, \$NM??B??.*	Entity reference, typically in the form tagname.parameter. System, hardware, and point references permitted.
2	NODE REFERENCES	*, ?, UCN node number 1-64	UCN node number of the entity references. (R410 and later)
3	MODULE REFERENCES	*, ?, APMM, PMM, IOPn EXAMPLES: IOP*, APMM, IOP5	Control partition (APMM, PMM) or IOP of the entity references. (R410 and later)
4	SLOT REFERENCES	*, ?, FLnnnn, TMnnnn, NMnnnn, RPnnnn, RCnnnn, LGnnnn, DCnnnn, PMnnnn, IOPn EXAMPLES: IOP*, RC10, LG20, FL[3-5]	Slot number of the entity references (such as flags, timers, numerics, reg PV, reg. control, logic slots, dig. composites, process module, or IOP slot). (R410 and later)
5	UCN checkpoints to search?	*,?, UCN network number 01-20	The .CP checkpoint file (&Inn, where nn is a number 1-20).
6	UCN NODES to search?	*, ?, UCN node number 1-64	UCN node-specified; this port appears when POINT DATA or BOTH selected from References to Entities menu
7	PROCESS Managers to search?	*, ?, UCN node number 1-64	UCN node-specified; this port appears when PM SEQUENCES or BOTH selected from References to Entities menu
8	PROCESS MODULES to search?	*, ?, APMM, PMM, DI, DO, HLA ILLAI, AO, STIM, LLMUX, PI, SI, DISOE	UCN module-specified.
9	Slots to search?	*, ?, FLnnnn, TMnnnn, NMnnnn, RPnnnn, RCnnnn, LGnnnn, DCnnnn, PMnnnn, IOPn EXAMPLES: IOP*, RC10, LG20, FL[3-5]	Specified slots, such as flags, timers, numerics, reg PV, reg. control, logic slots, dig. composites, process module, or IOP slot. Reference: Valid entries for "Slots to search?", page 20

Navigation Paths and Entry Choices

3.4 Entry Choices for UCN Checkpoint Files

Table 3-1 Description of Port Entries (continued)

Item	Port Description	Valid Entries	Scope of search/output
10	ENTITIES to search?	*, ?, entity name EXAMPLES: TIC1??, TIC*, !DI??S??, \$NM??B??	Entity, typically in the form tagname. System, hardware, and point entities permitted. This port appears when POINT DATA or BOTH selected from References to Entities menu.
11	PROCESS MODULE entities to search?	*, ?, entity name EXAMPLES: REAC??, REA*	Searches the specified process module entities for references to the tagname.parameter specified (in Item 1). This port appears when PM SEQUENCES or BOTH selected from References to Entities menu.
12	PM sequence programs to search?	*, ?, program name	Searches the specified sequence programs for references to the tagname.parameter specified (in Item 1). This port appears when PM SEQUENCES or BOTH selected from References to Entities menu.
13	Where is the database?	Defaults to network unless a different path is opened by selecting OTHER target.	Pathname to database file.
14	SUPPRESS columns of output?	Selection option from screen.	Suppresses the columns of output specified by the user. The # symbol can be used to suppress columns if using command line entry.

Valid entries for “Slots to search?” The following table lists valid entries for the **Slots to search?** entry port (Item 9 in , page 19 table) for the PM and the AM.

Table 3-2 Valid Entries for Slots to Search

Slot Type	Type Identifier	PM Slot Number	APM Slot Number
Logic	LG	1-80	1-80
Digital Composite	DC	1-255	1-255
Timer	TM	1-64	1-64
Flag	FL	1-1023	1-16,384
Numeric	NM	1-2047	1-16,384
Regulatory Control	RC	1-160	1-160
Regulatory PV	RP	1-80	1-80

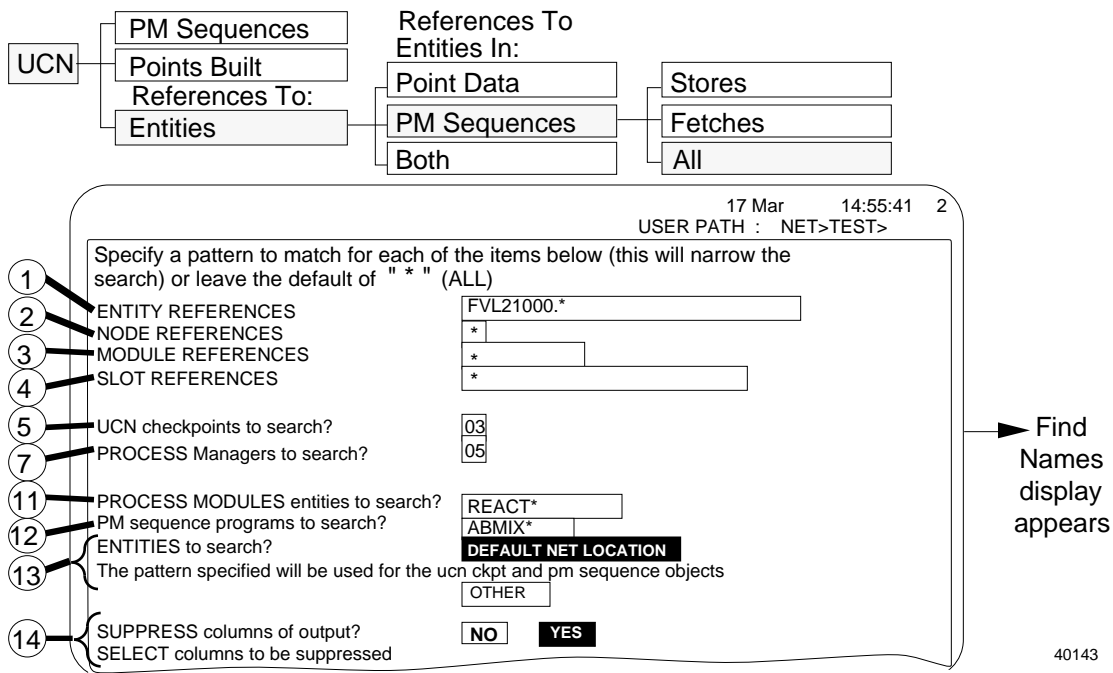
Table 3-2 Valid Entries for Slots to Search (continued)

Slot Type	Type Identifier	PM Slot Number	APM Slot Number
Process Module	PM	1-160	1-160
Array Point	AR	N/A	1-256
Device Control	DV	N/A	1-160

3.5 Navigation to UCN Checkpoint PM Sequences

Introduction The navigation path to the Find Names display for UCN checkpoint entity references/PM sequences is shown in the following figure, which is presented in two parts.

Navigation Path to UCN Checkpoint PM Sequences (Program References)



Find
Names
output

UCNCP	PM	PMOD_ENT	PM_SEQ	ENT_REF	RN	REF_MOD	REF_SL
3	5	REACT000	ABMIX000	FVL21000.MODATTR	5	APMM	DC(1)
3	5	REACT000	ABMIX000	FVL21000.MODE	5	APMM	DC(1)
3	5	REACT000	ABMIX000	FVL21000.OP	5	APMM	DC(1)

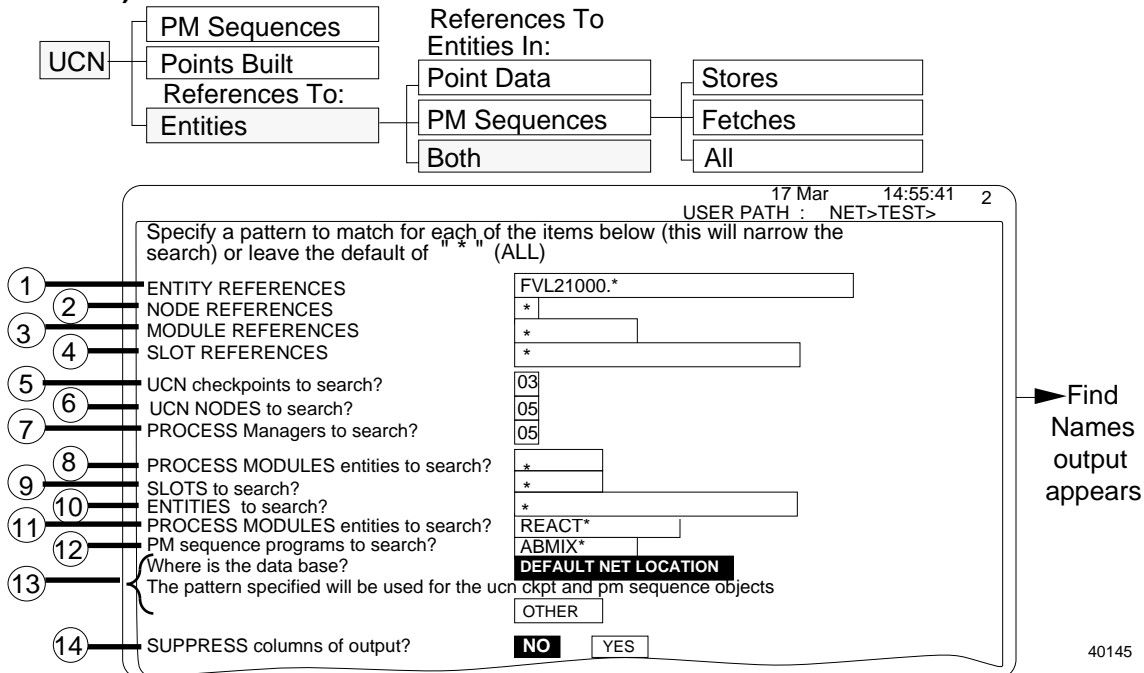
FINDNAMESCOMPLETE

40144

3.6 Navigation to UCN Checkpoints for Both

Introduction The navigation path to the Find Names display for UCN checkpoint entity references for both point data and PM sequences is shown in the following figure, which is presented in two parts.

Navigation Path to UCN Checkpoint for Both (Program References)



Find Names output

FN	UCN_CP	03	NODE_CP	05	MODULE	*	SLOT	*	ENTITY	*	ENT_REF	FVL21000.*	RN	*	REF_MOD	*	REF
UCNCP	NODE	MODULE	SLOT	ENTITY	ENT_REF	RN	REF_MOD	REF_SL									
3	5	APMM	LG1	SETNK000	FVL21000.D1	5	APMM	DC(1)									
3	5	APMM	LG1	SETNK000	FVL21000.I0	5	APMM	DC(1)									
FIND NAMES COMPLETE																	
FNUCN_CP	03	PM05	PMOD_ENT	REACT*	PM_SEQ	ABMIX*	ENT_REF	FVL21000.*	RN*	REF_MOD*	REF_SL*						
UCNCP	PM	PMOD_ENT	PM_SEQ	ENT_REF	RN	REF_MOD	REF_SL										
3	5	REACT000	ABMIX000	FVL21000.MODATTR	5	APMM	DC(1)										
3	5	REACT000	ABMIX000	FVL21000.MODE	5	APMM	DC(1)										
3	5	REACT000	ABMIX000	FVL21000.OP	5	APMM	DC(1)										
FINDNAMESCOMPLETE																	

40146

3.7 Example Search (Peer-to-Peer)

Search description, example

The **Node References** entry (Item 2 in previous figure) and the **UCN nodes to search?** (Item 6) or **Process Managers to search?** (Item 7) entries can be used to generate a search for peer-to-peer connections between two UCN nodes. You can do this to provide a more organized search, by node-to-node, instead of using “wild card” references that list all node references to peer nodes *and* local connections within the node.

The advantage of using a “wild card” search is that you get all references, but you would have to sort through the node-to-node references again (Documentation Tool can do this for you). The advantage of a narrow search is that you get only the data you need.

Example search

For example, a user could ask for all entity references from node 3 to node 5 on UCN 1. This actually requires a search for references from node 3 to node 5 and it also requires a search from node 5 to node 3.

The following figure, which is presented in two parts, illustrates this approach, note the printout results from this type of narrow search.

Example Node-to-Node (Peer-to-Peer) search, part 1

17 Mar 13:19:17 2
USER PATH : NET>TEST>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of " * " (ALL)

ENTITY REFERENCES \$NM??B*
 NODE REFERENCES 03
 MODULE REFERENCES *
 SLOT REFERENCES *

UCN checkpoints to search? 03
 UCN NODES to search? 05

PROCESS MODULES to search? *PMM
 SLOTS to search? LG1
 ENTITIES to search? *
 Where is the data base? DEFAULT NET LOCATION
 The pattern specified will be used for the ucn ckpt and pm sequence objects
 OTHER

SUPPRESS columns of output?
 Select columns to be suppressed NO YES

MEDIA	UCN_CP	NODE_CP	MODULE	SLOT	ENTITY	ENT_REF	RN	REF_MOD	REF_SL
FN	#UCN_CP_03	NODE_CP_05	MODULE *PMM	#SLOT LG1	ENTITY * ENT_REF \$NM??B*.*	RN 03	REF_MOD *	REF_SL *	
NODE	MODULE	ENTITY	ENT_REF	RN	REF_MOD	REF_SL			
5	APMM	SETNK000	\$NM03B03.FL(1)	3	PMM	FL(1)			

FIND NAMES COMPLETE

40147

Navigation Paths and Entry Choices
 3.7 Example Search (Peer-to-Peer)

Example Node-to-Node (Peer-to-Peer) search, part 2

17 Mar 13:19:17 2
 USER PATH : NET>TEST>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of "*" (ALL)

ENTITY REFERENCES
 NODE REFERENCES (D)
 MODULE REFERENCES
 SLOT REFERENCES

UCN checkpoints to search? (E)
 UCN NODES to search? (F)

PROCESS MODULES to search?
 SLOTS to search?
 ENTITIES to search?
 Where is the data base?
 The pattern specified will be used for the ucn ckpt and pm sequence objects

SUPPRESS columns of output?
 Select columns to be suppressed

FN	#UCN_CP	03	NODE_CP	03	MODULE	*PMM	#SLOT	LG1	ENTITY	*	ENT_REF	\$NM??B*. *	RN	05	REF_MOD	*	REF_SL	*	
NODE	MODULE	ENTITY	ENT_REF	RN	REF_MOD	REF_SL													
3	PMM	TANK100	\$NM03B05.FL(2)	5	APMM	FL(2)													
3	PMM	TANK100	\$NM03B05.FL(3)	5	APMM	FL(3)													
3	PMM	TANK100	\$NM03B05.FL(1)	5	APMM	FL(1)													

FIND NAMES COMPLETE

40148

3.8 Navigation to History Groups

Introduction The navigation path to the Find Names display for History Groups is shown in the following figure.

Navigation to History Groups

History

17 Mar 10:44:00 6
USER PATH : \$F11>HEAT>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of " * " (ALL)

ENTITY REFERENCES

HISTORY units to search?

HISTORY groups to search?

Where is the data base? **DEFAULT NET LOCATION**

SUPPRESS columns of output?

Hit ENTER -execute command, CANCEL -start over, MENU -exit, HELP for info

FN
Loading Overlay

FN	H_UNIT	01	H_GRP	*	ENT_REF	FIC21841.*
MEDIA	HU	HGRP			ENT_REF	
NET>I002	01	1			FIC21841.PV	
NET>I002	01	1			FIC21841.SP	
NET>I002	01	1			FIC21841.CV	

FIND NAMES COMPLETE

10:45:07 6
\$F11>HEAT>

40149

3.9 Entry Choices for History Groups

Introduction Recall that History Groups are assigned by unit numbers. Each group is then defined during History Groups configuration to collect data for up to 20 tagname.parameters. A historized value is not limited to just PVs, but can include any valid UCN parameter.

Meaning of entries The entries in Navigation to History Groups, page 27 are described as follows:

- Entity References
 - This refers to the tagname.parameter to search for in the history groups. The “*” character can be used to wild card the tagname or the parameter name or both (*.*)).
- History units to search?
 - This refers to the unit id as defined during NCF configuration. Because history is stored by unit number, the NCF.CF file must be used in order to determine the unit number of the unit id specified. Find Names uses the path as defined under “Volume Pathnames” to find the NCF. The “wild card” default entry of “*” searches all history units.
- History groups to search?
 - This refers to the history group number defined during History Group configuration. The “wild card” default entry of “*” searches all history groups.
- Where is the database?
 - This refers to the database residing on a history volume defined during NCF configuration. A selection of DEFAULT NET LOCATION directs the search to the history volumes on the network. Selecting OTHER opens up a new entry port where a pathname for the search can be entered.

3.10 Navigation to Picture Files

Introduction The navigation path to the Find Names display for pictures is shown in the following figure.

Navigation to Picture Files

The screenshot shows a terminal window titled "Picture Ed" with a "Loading Overlay". The user path is "\$F11>HEAT>". The search criteria are: ENTITY REFERENCES: FIC11121.*; PE SRC to search?: *; Where is the data base?: NET>PSRC; SUPPRESS columns of output?: NO. The output is as follows:

```

FN
Loading Overlay
FN MEDIA NET>PSRC PE * ENT_REF FIC11121.*
  MEDIA   PE      ENT_REF
-----
NET>PRSC FEEDHXXX  FIC11121.PV
FIND NAMES COMPLETE
    
```

On the left side of the terminal window, there is a vertical menu with the following options: Schematics, Subpictures, Entities, DDB References, and All References. The "All References" option is currently selected.

40150

3.11 Entry Choices for Picture Files

Introduction This section considers a typical picture file search a UCN administrator would perform. For example, a search for entity references would include the path shown in [Navigation to Picture Files](#), page 29.

Meaning of entries The following information applies to the entries in [Navigation to Picture Files](#), page 29:

- Entity References
 - This refers to the tagname.parameter to search for in the picture editor *and* Free Format Log source files. The “*” character can be used to wild card the tagname or the parameter name or both (*.*) .
- PE SRC to search?
 - This refers to the picture editor *and* Free Format Log source files to search. A “wild card” default entry of “*” searches all source files in the pathname specified in the next entry.
- Where is the database?
 - This entry defines where the source files reside. Selecting DEFAULT NET LOCATION directs the search to all volumes and directories on NET. Selecting OTHER opens up a new port where a pathname for the search can be entered.

3.12 Navigation to Area Database Files (Schematics)

Introduction

Having located the schematic names from your entity reference search, your next step could also include a search for what Area the schematic resides in. The navigation path to the Find Names display for Area database is shown in the following figure.

Navigation to Area Database Files (Schematics) figure

17 Mar 11:28:48 6
 USER PATH : \$F11>HEAT>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of "*" (ALL)

SCHEMATICS

AREA DATA BASE to search?

ENTITIES to search?

Where is the data base?

OTHER

SUPPRESS columns of output? NO YES

Hit ENTER -execute command,

17 Mar 11:36:42 6
 USER PATH : \$F11>HEAT

FN
 Loading Overlay

FN	AREA	* ENTITY	* SCHEM	FEEDHXXX	SCHEM
MEDIA	AREA	ENTITY	ENTITY		
NET-&D01	1	\$OABSTRT			FEEDHXXX
NET-&D03	3	\$OABSTRT			FEEDHXXX
NET-&D04	4	\$OABSTRT			FEEDHXXX
NET-&D05	5	\$OABSTRT			FEEDHXXX
NET-&D06	6	\$OABSTRT			FEEDHXXX
NET-&D07	7	\$OABSTRT			FEEDHXXX
NET-&D08	8	\$OABSTRT			FEEDHXXX
NET-&D010	10	\$OABSTRT			FEEDHXXX

FIND NAMES COMPLETE

Area

- Schematics
- Entities
- All References

40151

3.13 Entry Choices for Area Database Files (Schematics)

Meaning of entries The entries in the above figure are described as follows:

- Schematics
 - This refers to which schematics to search for in the Area database files.
- Area database to search?
 - This refers to the &Dnn (where nn is a number from 1 to 10) Area database files. A “wild card” default value of “*” searches all areas.
- Entities to search?
 - This refers to the Area database entities listed in Area database entities to search, page 34. While a picture can be referred to as an Associated display, your most likely search will include the pathname catalog (\$OABSTRT), which lists the schematics intended to be memory-resident. Following a schematic modification, knowing what schematics are memory-resident helps you identify which Universal Stations need to be reloaded.
- Where is the database?
 - This entry defines where the area database files reside. Selecting DEFAULT NET LOCATION directs the search to the area database directories on NET. Selecting OTHER opens up a new entry port where a pathname for the search can be entered.

3.14 Navigation to Area Database Files (Entities)

Introduction Your entity reference search could also include a search for where the reference resides in the Area database. The navigation path to the Find Names display for Area database is shown in the following figure.

Navigation to Area Database Files (Entities) figure

17 Mar 11:43:37 6
 USER PATH : \$F11>HEAT>

Specify a pattern to match for each of the items below (this will narrow the search) or leave the default of "*" (ALL)

ENTITY REFERENCES

AREA DATA BASE to search?

ENTITIES to search?

Where is the data base?
 DEFAULT NET LOCATION
 OTHER

SUPPRESS columns of output? NO YES

Hit ENTER -execute command, CANCEL -start over, MENU -exit, HELP for info

17 Mar 11:57:09 6
 USER PATH : \$F11>HEAT

FN
 Loading Overlay

FN	AREA	* ENTITY AREA	* ENT_REF ENTITY	ENT_REF
NET-&D04	4		\$ANNDATA(21)	FIC11121
NET-&D04	4		\$ANNDATA(30)	FIC11121
NET-&D04	4		\$OGROUP(81)	FIC11121
NET-&D04	4		\$OUNTRND(21)	FIC11121

FIND NAMES COMPLETE

Area
 Schematics
 Entities
 All References

40152

3.15 Entry Choices for Area Database Files (Entities)

Meaning of entries The entries in the above are described as follows:

- Entity References
 - This refers to the tagname.parameter to search for.
- Area Database to search?
 - This refers to the &Dnn (where nn is a number from 1 to 10) Area database files. A “wild card” default value of “*” searches all areas.
- Entities to search?
 - This refers to the Area database entities listed in the following table. In other words, what do I want to search in the Area database? For example, “\$OGROUP(nnn)” is the Area database operating group *entity*.

Area database entities to search Area database entity names and what they represent are listed in the following table.

Table 3-3 Area Database Entities to Search

Entity	Meaning
\$ANNDATA	Annunciator policy
\$ANPOLC	LCN node annunciation policy
\$DMONBLK(n)	Hiway box annunciation policy
\$OABSTRT	Pathname catalog
\$OBTCHSM	Batch summary display
\$OFFLOG(n)	Free format log
\$OGROUP(n)	Group display
\$OLOG(n)	Standard log
\$OMODGRP(n)	Process module group
\$OPRCJOR(n)	Process journal
\$OPRTRND(n)	Printed trend
\$OREPORT(n)	Report
\$SOEJOR(n)	SOE Journal
\$OSYSJOR(n)	System Journal
\$OSYSTAT	System Status
\$OTREND	Area Trend display
\$OUNITSM(N)	Unit Summary display
\$OUNTRND(n)	Unit Trend display

Table 3-3 Area Database Entities to Search (continued)

Entity	Meaning
\$RTJNL	Real Time journal assignment
\$RUNTSTS	Unit Assignment display

3.16 Items That Cannot Be Found

Source files missing information

There are some cases where it is not possible to find and list names that are related to schematics (custom displays) and Button Configuration because the source files don't contain all of the information needed to find these names. These cases are as follows:

- Entity names that are parameters of the actor DETAIL [DETAIL (“FC101”)] are not listed.
- Where indirection is used to name an entity, the entity name cannot be listed. For more information, see *Control Language/Application Module Reference Manual*, Rules and Elements of CL/AM section.
- Find Names locates schematic names through actors SCHEM and OVERLAY. A name can be found and listed only if it is a literal string [SCHEM(“XYZ”)]. If a name is a read string [OVERLAY (R_STR(A100.NAME))], question marks (???????) appear in the name port.
- Names denoted by an alphanumeric string preceded by “&” (&A>PRESS) cannot be listed.

Finding instances of a deleted entity

If a reserved entity or a data point has been deleted, you still may be able to find objects that used the entity before it was deleted.

When Find Names cannot convert an internal ID to an entity name, it sends an error message. Each such message indicates a possible usage of the deleted entity. Therefore, you can enter some string with wildcard characters in the `Entity references` port. The string should be one you are not likely to find; your goal is to output only error messages, not a lot of matches. An example might be `*AAAAAAA`.

The method above works for everything except for Picture Editor source files. Picture Editor source files store entity names as strings, so you find references to a deleted entity just by entering its name in the `Entity references` port.

Section 4—Find Names Examples

4.1 Find Names Command Line

Introduction As you probably suspected, Find Names can be a part of an (EC) file. You can then build your own utility that engineers, operators, and technicians can easily use.

Search syntax The format of the command line can be seen after you invoke a search. The following example reviews the format for a UCN checkpoint search for entity references.

First consider the command for an entity references to point data search. The command line appears as in the following figure. For valid field entries, see 3-1 Description of Port Entries, page 19.

Command Line Format FN #UCN_CP 03 NODE_CP 03 MODULE *PMM #SLOT LG1 ENTITY * ENT_REF \$NM??B*.* RN 05 REF_MOD * REF_SL *

where

#UCN_CP 03	UCN Checkpoint to Search	Port Entry #5
NODE_CP 03	UCN Node to Search	Port Entry #6
MODULE *PMM	Process Modules to Search	Port Entry #8
#SLOT LG1	Slots to Search	Port Entry #9
ENTITY *	Entities to Search	Port Entry #10
ENT_REF \$NM??B*.*	Entity Reference	Port Entry #1
RN 05	UCN Node # of Entity Reference	Port Entry #2
REF_MOD *	Module # of Entity Reference	Port Entry #3
REF_SL*	Slot # of Entity Reference	Port Entry #4

4.2 Example Search From Command File

Peer search example

The command file shown below incorporates Find Names commands that search for peer-to-peer connections. This example could be useful as a utility for engineers or system administrators because it filters out local connections and retrieves just the peer connections.

Presumably the user could have Find Names output the results to a text file, where the results could be filtered even further by Documentation Tool or to a PC tool (after going through a text file converter.) The reason you would want to take it to Documentation Tool is to find out the parameter the entity reference is connected to. Knowing the parameter will help you identify whether it is a push or pull transfer.

Example command file

```
LARRY2.EC
03/23/93 08:34:39

& THIS FILE USES FIND NAMES COMMANDS TO SEARCH FOR PEER TO PEER
& CONNECTIONS. R410 IS REQUIRED. WHEN A FIND NAMES SEARCH IS PERFORMED
& FOR PEER TO PEER CONNECTIONS, TWO SEARCHES ARE REQUIRED -- A SEARCH
& FOR CONNECTIONS FROM NODE "A" TO NODE "B" AS WELL AS A SEARCH FROM
& NODE "B" TO NODE "A". NOTE: THIS COMMAND FILE DOES NOT SEARCH
& SEQUENCE PROGRAMS, ALTHOUGH IT WOULD BE VERY EASY TO ADD THE FIND
& NAMES COMMANDS TO DO THAT

&V1 "TYPE IN PM NODE NUMBER (1-31) FOR THE PEER TO PEER CONNECTIONS"

& CREATE A FILE AND USE FOR A STRING COMPARE. THIS IS DONE BECAUSE
& WHEN YOU SEARCH FOR PEER CONNECTIONS, IT IS NOT NECESSARY TO SEARCH
& FOR LOCAL CONNECTIONS. FOR EXAMPLE, IT IS NOT NECESSARY TO SEARCH
& PM#9 FOR CONNECTIONS TO PM#9.

DO NET>JIMK>^V1.XP
DO
&P FILE USED FOR UCN NODE NUMBER COMPARE

& CREATE A FILE AND OUTPUT THE FIND NAMES RESULTS. THIS FILE CAN THEN BE
& FILTERED AT DOCUMENTATION TOOL OR TAKEN TO A TEXT FILE CONVERTER.
& THIS IS DONE TO IDENTIFY WHETHER THE CONNECTION IS A "PUSH" OR "PULL"
& CONNECTION.

DO CONNECTS.XX
& OPEN A FILE THAT CONTAINS A LIST (IN COLUMN FORMAT) OF UCN NODE
& NUMBERS THE FILE IS CALLED NUMBAH.XX, AND CAN BE BUILT BY A USER.

&F NET>JIMK>NUMBAH.XX
&L NEXT
&IF NOT READ
&THEN &G FINISH
```

```
&IF EXISTS NET>JIMK>^R1.XP
&THEN &G NEXT
&P
&P
&P LIST OF NODE REFERENCES FROM NODE ^V1 TO NODE ^R1
&P
FN UCN_CP * NODE_CP ^V1 MODULE * SLOT * ENTITY * ENT_REF * RN ^R1 REF_MOD * REF_SL *
&P
&P
&P
&P LIST OF NODE REFERENCES FROM NODE ^R1 TO NODE ^V1
&P
FN UCN_CP 03 NODE_CP ^R1 MODULE * SLOT * ENTITY * ENT_REF * RN ^V1 REF_MOD * REF_SL *
&P
&P
&P
&G NEXT
&L FINISH
DO
DL NET>JIMK>^V1.XP
&P THIS COMPLETES YOUR SEARCH, ADIOS!
```

ATTENTION

To get the FN command line into a text file that has an 80-character limitation, you can

- command a dataout (DO) to a file,
- invoke and run Find Names,
- wait for Find Names to complete, or abort (Ctl - Break) while Find Names is outputting, and then
- edit the file. (Delete line and insert line are shortcuts to move the FN command line.)

Find Names Examples

4.2 Example Search From Command File

Section 5—Find Names Error Messages

5.1 UCN Checkpoint Search Errors

Introduction

Find Names error messages encountered during a UCN checkpoint search and a typical cause are listed below.

- Corrupted Data in File
 - A point does not have a corresponding checkpoint record in a PM/APM checkpoint file.
 - Possible cause: It is possible that more points exist in the NIM's checkpoint file of a particular point type than exist as checkpoint records in the node's checkpoint file.
- Incorrect File Version
 - The version/revision filed in a point's checkpoint record in a PM/APM checkpoint file does not match the version/revision numbers against which Find Names was compiled.
 - Possible cause: It is possible that Find Names was not released with the newer release of the PM or APM personality.
- Invalid File Configuration Revision
 - The revision of the NIM/PM/APM master file or checkpoint file does not match the revision against which Find Names was compiled. The error is reported by the RNOS File System (error 80) when trying to open the file.
- Memory Unavailable
 - Memory could not be retrieved for a particular structure that is used by Find Names to process the Find Names command line. This error is fatal and aborts Find Names processing.
- Value Out of Range
 - A connection value could not be converted from its internal format to its external format.
 - Possible cause: It is possible that a point does not exist in the NIM's checkpoint file for a point referenced in the connection value. For example, the point could have been deleted with LOADSCOP=NIMONLY and done when two NIMs are used for peer-to-peer.
 - Possible cause: It is possible that the IOP configuration of the node has been changed and checkpointed without the corresponding hardware address used in a connection being updated. For example, !DO05S01.OP is used in a point's connection, but module 5 is no longer a DO module but a STIM module.

5.2 UCN Sequence Program Search Errors

Introduction

Find Names error messages encountered during a UCN sequence program search and a typical cause are listed below.

- Memory Unavailable
 - Memory could not be obtained for data structures used by Find Names to process the list of possible sequence program names or to process an individual sequence program.
- Sequence Name not in Checkpointed Library
 - The name of the sequence currently being processed has not been put in the NIM library.
 - Possible cause: The sequence was compiled without using the -CL (update library) switch.
 - Possible cause: Find Names is not reading the most recent NIM checkpoint file.
- Sequence Must be Recompiled with Latest Compiler
 - The CL compiler used to compile a PM sequence was older than R300. This does not apply to an APM sequence that is not recognized by R300 Find Names.

Section 6—User Reference Information

6.1 Object, Reference, and File Relationships

Finding connections between objects

When they execute, many objects obtain values from other objects. Find Names helps you find connections between objects. It can perform the searches listed in the following table.

Table 6-1 Finding Connections Between Objects

FN can look through...	for references to...	An example of such a reference is...
Entities	Entities	The value in A100.CISRC is received from B100.PV.
UCN entities	Nodes, modules, slots	A100.LISRC (1) is referencing B100 This entity is on node n, module n, slot n.
Entities	CDS	The value in A100.PKGNAME(1) is received from MYPKG.
Entities	CL Blocks	The value in A100.BLKNAME(1) is received from MYCL.
Sequence programs	Nodes, modules, slots	
Sequence programs	Box parameters & hardware addresses	IF ARRAYPT.STR8(1) = "ST" is a reference to a box "string" parameter.
CL Blocks	Entities (store, fetch, indirect)	IF ARRAYPT.STR8(1) = "ST" THEN
Picture Editor source	Entities	ADD-VALUE(B100.PV)
Picture Editor source	DDB locations	ADD-VALUE(BOOL01G)
Picture Editor source	Subpictures	SUBPICTURE(MYVALVE)
Picture Editor source	Schematics	SCHEM("MYPLANT")
Text files	Any string	IF ARRAYPT.STR8(1) = "ST" THEN
Area database	Entities	\$OGROUP(1) contains entity A100
Area database	Schematics	
History unit	Entities	History unit 1, group 5, collects values from A100.PV

Files searched by Find Names

Find Names searches files, only. It does not search data stored in nodes' memories. The following table lists the files that are searched by each type of Find Name search.

Table 6-2 File Types Searched by Find Names

FN can look through...	for references to...	File resident data accessed in the search
Entities	Entities	Point records in checkpoint files
Entities	CDS	Point records in checkpoint files

User Reference Information

6.1 Object, Reference, and File Relationships

Table 6-2 File Types Searched by Find Names (continued)

FN can look through...	for references to...	File resident data accessed in the search
Entities	CL Blocks	Point records in checkpoint files
Sequence programs	Entities, box parameters, and hardware addresses	Sequence library in checkpoint files, Process Module point records from checkpoint files, sequence object files
CL Blocks	Entities	Reference lists in AM checkpoints
Picture editor source	Entities	Picture editor source files
Picture editor source	DDB locations	Picture editor source files
Picture editor source	Schematics	Picture editor source files
Text files	Any string	Text files found on user volumes
Area database	Entities	Area database files
Area database	Schematics	Area database files
History units	Entities	History files

6.2 Using Find Names Displays

Entries on search-narrowing displays

The character strings you enter in each port are matched against each item returned during the search. If matching data is found, a line is output.

To search for particular items or subsets, substitute for the default wildcard. The wildcard “?” and “*” may appear anywhere in a string. Their meanings are

- ? match 1 character
- * match 0 or more characters

For example, to search for references to entities that start with “A,” put A* in the ENTITY REFERENCES port.

Organizational Summary menus

You may need to use local records to determine specific inputs for your system. You can use the Organizational Summary menus within the System Menu to find much of this information. Another good source is the Command Processor’s List Volumes (LV) command.

For information about using the Organizational Summary menus, refer to Areas, Units, Groups, and Schematics.

For information about the List Volumes command see *Command Processor Operations*, List Volumes subsection.

The following table lists the valid entries for all of the ports that can appear.

Table 6-3 Valid Entries for Search-Narrowing Ports

Screen Descriptor	Max no. of Characters	Valid Entries
AM checkpoints to search?	2	AM UNIT NUMBER, 1-100
AREA DATA BASE to search?	2	Area number, 1-10
BUTTON CONFIG source to search?	8	Button config. source file name without .KS suffix.
CDS	8	CDS custom GDF file name without .GD suffix
CL BLOCKS	8	CL block file name without .CL suffix
CL BLOCKS to search?	8	CL block file name without .CL suffix
CM Checkpoints to search?	2	CG unit number, 1-100, &np/&Hun Note: np = HM node pair number un = unit number
ENTITIES TO SEARCH?	35	Entity name (tag name) or parameter in an entity in NAME . PARAM form

Table 6-3 Valid Entries for Search-Narrowing Ports (continued)

Screen Descriptor	Max no. of Characters	Valid Entries
ENTITY FETCH REFERENCES	38 AM/CL, 45 MC	Parameter in an entity in NAME . PARAM form
ENTITY INDIRECT REFERENCES	38	Parameter in an entity in NAME . PARAM form
ENTITY REFERENCES	38 AM/CL, HM, HG, 45 MC, CM	Parameter in an entity in NAME . PARAM form
ENTITY STORE REFERENCES	38 AM/CL, 45 MC	Parameter in an entity in NAME . PARAM form
HG checkpoints to search?	2	Process network (Data Hiway) number, 1-20
HISTORY Groups to search?	3	HG number, 1-120, (Find Name - History)
HISTORY Units to search?	2	Unit ID as configured under Unit Names
MODULES to search?	6	PM/APM module identifiers: PMM, DI, DO, HLAI, LLAI, AO, STIM, LLMUX, PI, SI, APMM, DISOE
NODES to search?	2	UCN node number, 1-40 (UCN Status display)
PE SRC to search?	8	Picture Editor (schematic) source file name without.DS suffix
PM SEQUENCES	8	PM/APM sequence CL program name
POINTS BUILT	29	Entity name (tag name) or parameter in an entity in NAME . PARM form
PROCESS Managers to search?	2	UCN Node number 1-40
PROCESS MODULE entities to search?	16	Data Hiway or UCN Process Network Process Module Point Type entity name (tag name)
SCHEMATICS	8	Picture Editor (schematic) source file name without .DS suffix
SLOTS to search?	6	Slot-type identifier and slot number as defined in the following table (example RC02)
SUBPICTURES	8	Subpicture file name without .DS suffix
TEXT files to search?	11	User's text file name with 2-char. suffix, for example PMEXBLD . EB or JOHN . XX
UCN checkpoints to search?	2	UCN process network number (1-20)

Table 6-3 Valid Entries for Search-Narrowing Ports (continued)

Screen Descriptor	Max no. of Characters	Valid Entries
Where is the database?	16	NET for Network or \$Fn for removable media Note: n = number of removable medium (floppy or cartridge)
MC SEQUENCES	8	MC sequence CL program name

The following table defines valid entries for the Slots to Search port. This port applies only to UCN checkpoint data. Valid entries for the slot number are in this form: RC01, where the first two characters are the slot type and the numbers define the slot number.

Table 6-4 Valid Entries for “Slots to Search”

Slot Type	Slot Type	PM Slot Number	APM Slot Number
Logic	LG	1-80	1-80
Digital Composite	DC	1-255	1-255
Timer	TM	1-64	1-64
Flag	FL	1-1023	1-2047
Numeric	NM	1-2047	1-2047
Regulatory Control	RC	1-160	1-160
Regulator PV	RP	1-80	1-80
Process Module	PM	1-160	1-160
Array Point	AR	N/A	1-256
Device Control	DV	N/A	1-160

6.3 Valid Entries for Search-Narrowing Ports

Entity-related ports The following table defines valid entries for ports related to entities.

Table 6-5 Valid Entries for Entity-Related Ports

Port Names	Valid Entries	Entry Examples
Entity references	“Entity” can include index but no range	
Entity store, fetch, and indirect references	“Parameter can be indexed. The index can contain a wildcard and a range.	
Entities to search	entity	A100 or \$OGROUP(1)
	entity.parameter	A100 .PV
	entity.parameter(n)	A100 .MYPAR(1?)
	entity.parameter(n..nn)	A100 .MY*(1..10)
	entity.parameter (*..nn)	A100 .MY*(*..10)
	entity.parameter(n..*)	A100 .MYPAR (10..*)

AM-related ports The following table defines valid entries for ports in which you define searches through Application Module checkpoint files.

Table 6-6 Valid Entries for AM Checkpoint Searches

Port Names	Valid Entries	Entry Examples
Entity references (store, fetch, indirect)	Reference: Entity-related ports, page 48	
Entities to search?	Reference: Entity-related ports, page 48	
CDS	1-8 character identifier	MYCDS, MY*, ??CDS
CL Blocks	1-8 character identifier	MYBLOCK, MY*, ??BLOCK
AM checkpoints to search?	1-2 character unit id	01, *, ?X

CM-related ports The following table defines valid entries for ports in which you define searches through Computing Module checkpoint files.

Table 6-7 Valid Entries for CM Checkpoint Searches

Port Names	Valid Entries	Entry Examples
Entity references	Reference: Entity-related ports, page 48	
Entities to search?	Reference: Entity-related ports, page 48	
CDS	1-8 character identifier	MYCDS, MY*, ??CDS
CM checkpoints to search?	1-2 character unit id	01, *, ?X

HG-related ports The following table defines valid entries for ports in which you define searches through Hiway Gateway checkpoint files.

Table 6-8 Valid Entries for HG Checkpoint Searches

Port Names	Valid Entries	Entry Examples
Entity references	Reference: Entity-related ports, page 48	
Entities to search?	Reference: Entity-related ports, page 48	
MC sequences to search?	1-8 char identifier	MYPRG, MY*, ???0501
Process module entities to search?	Reference: Entity-related ports, page 48	
HG checkpoints to search?	Hiway number (1-20)	05, *

UCN-related ports The following table defines valid entries for ports in which you define searches through Universal Control Network checkpoint files.

Table 6-9 Valid Entries for UCN Checkpoint Searches

Port Names	Valid Entries	Entry Examples
Entity references	Reference: Entity-related ports, page 48	
Nodes references	Node number (1-64)	1, *, 2?
Modules references	PMM, APMM, DI _{nn} , DO _{nn} , HLA _{nn} , LLA _{nn} , AO _{nn} , STIM _{nn} , LLMUX _{nn} , PI _{nn} , SI _{nn} , DISO _{Enn}	DI*, DI12
Slot references See the table titled 6-4 Valid Entries for "Slots to Search", page 47	Slot type and index	fl (1)
	Index may be blank	fl
	(n . . nn)	fl (20 . . 50)
	(* . . nn)	fl (* . . 50)
	(n . . *)	fl (200 . . *)
	AI (index) analin	
	AO (index) analout	
	DI (index) digin	
DO (index) digout		

User Reference Information

6.3 Valid Entries for Search-Narrowing Ports

Table 6-9 Valid Entries for UCN Checkpoint Searches (continued)

Port Names	Valid Entries	Entry Examples
Slot References (continued)	FL (index) flag	
	TM (index) timer	
	NM (index) numeric	
	RP (index) reg pv	
	RC (index) reg ctl	
	LG (index) logic	
	DC (index) digital comp	
	PM (index) process module	
	STR8 (index) string(8)	
	TIME (index) time	
	AR (index) array	
Slot References (continued)	DV (index) device control	
	SInn:FL (index) Note: SI = serial I/O	
	SInn"NM (index) Note: SI = serial I/O	
	SInn:STR8 (index) Note: SI = serial I/O	
Entities to search?	Reference: Entity-related ports, page 48	
Process modules to search?	PMM, APMM, DIInn, DOInn, HLAInn, LLAInn, AOnn, STIMnn, LLMUXnn, PInn, SInn, DISOEnn	AO*, DI16
Slots to search?	Slot type and number. Parentheses are not used here and ranges are not legal for the slot number.	FL* FL100 FL2
	FLnn flag	
	TMnn timer	
	NMnn numeric	
	RPnn reg pv	
	RCnn reg ctl	

Table 6-9 Valid Entries for UCN Checkpoint Searches (continued)

Port Names	Valid Entries	Entry Examples
SLOTS to search? (continued)	LGnn logic	
	DCnn digital comp	
	PMnn process module	
	ARnn array	
	DVnn device control for IO modules, slot number	
Process module entities to search?	Reference: Entity-related ports, page 48	
UCN nodes to search?	UCN node number (1-64). No LMs.	1, 3*, *
Process Managers to search?	UCN PM/APM node number (1-64)	1, 3?, *
UCN checkpoints to search?	UCN (process network) number (1-20)	5, *

Picture Editor-related ports

The following table defines valid entries for ports related to searches through Picture Editor source files.

Table 6-10 Valid Entries for searches through Picture Editor Source Files

Port Names	Valid Entries	Entry Examples
Schematics	1-8 character file name	MYSCHM, *SCH?A
Subpictures	1-8 character file name	MYPIC, MYPIC*
DDB references	1-8 character file name	*, MYREAL, MY*
DDB area	*	*_GBL, SYSTEM_GBL
	"local",	
	"customer_gbl",	
	"honeywell_gbl",	
	"system_gbl",	
	"*_gbl"	

Table 6-10 Valid Entries for searches through Picture Editor Source Files

Port Names	Valid Entries	Entry Examples
DDB data type	"INTEGER"	ENTITY, VARIABLE
	"REAL"	
	"BOOLEAN"	
	"STRING"	
	"ENUM"	
	"DATE_TIME"	
	"ENTITY"	
	"VARIABLE"	
DDB index	Integer	10, 11, *

Text File-related ports

The following table defines valid entries for ports related text file searches.

Table 6-11 Valid Entries for Text File Searches

Port Names	Valid Entries	Examples	
Pattern	1-80 character string enclosed in matching delimiters. The following characters have special meanings inside the pattern specified for a text file search.		
	?	Match 1 character.	msg0?
	*	Match 0 or more characters	msg*
	[]	Characters are enclosed in the brackets. Match any character found in the bracketed set.	a[123]00
Pattern (continued)	[~]	Not these characters. Match any characters NOT found in the bracketed set.	a[~456789]00
	[x-y]	Match any characters in the range "x to y."	a[1-3]00
	!	Match 0 or more occurrences of the previous character.	a10!
	@	Override special meaning of following characters.	why@?
Text files to search?	1-8 character file name and suffix.	*.CL, heat*.cl	

Area-related ports The following table defines valid entries for ports related to area database searches.

Table 6-12 Valid Entries for Area Database Searches

Port Names	Valid Entries	Entry Examples
Schematics	1-8 character identifier	MYSCHEM, *SCH?A
Entity references	Reference: Entity-related ports, page 48	
Entities to search?	Reference: Entity-related ports, page 48	
Area database to search?	Area number (1-10)	1, *

History-related ports The following table defines valid entries for ports related to searches through HM history unit files.

Table 6-13 Valid Entries for History-Related Ports

Port Names	Valid Entries	Entry Examples
Entity references	Reference: Entity-related ports, page 48	
History units to search?	1-2 character unit ID	01, *
History groups to search?	Group number (1-150)	2, 1?

6.4 Special Objects for UCN Searches

Introduction

In software release R410 and later, three command-line objects are available to identify connections in UCN checkpoint files or in CL/APM and CL/PM sequence programs. These objects are listed in the following table.

Table 6-14 Command Line Objects

Object	Form
Node Reference	RN
Module Reference	RM or REF_MOD
Slot Reference	RS, REF_SL, or REF_SLOT

Purpose

The purpose of these objects is that:

- They can be used to display the ultimate UCN node, module, and slot referenced by an entity reference.
- They can be used to restrict the referenced entities displayed to particular hardware specified by the user.
- They can find peer-to-peer connections between UCN devices.

The distinction of ultimate reference is important with respect to APM array points. For those points, although they reside in the APMM module, they can map data in a device connected to a Serial Interface. In such a case, the ultimate module referenced is the SI IOP, not the APMM.

Suppressing output columns

Note, that because each of these three special objects on the Find Names command line can be preceded by the output suppression character (# -the pound sign as in #RN), you can control the individual display of these three columns of output. If all three columns were suppressed, for example, the output would be limited to display entity references to a particular set of hardware without showing these three components.

Using wildcard characters

These special objects can be used with wildcard characters. For example, “?” to match a single character and “*” to match 0 or more characters. In order to use wildcard characters, you must first know the kinds of output that the three special objects can return. This output is described following.

Output for entities not on the UCN

For all three objects, if the entity reference is not on the UCN (for example, the reference is to an AM point on the LCN, as when the EIPPCODE parameter of some point refers to the AM point), the text returned for each object is a single blank (“ ”). A single blank is also returned for each object in the case that the entity reference is NULL. In these two cases, the entity reference does not have known UCN components, so no referenced node, module, or slot is listed.

Output for a Reference Node object The text returned for the RN object (the node referenced in the entity reference) consists of only the 1- or 2-digit UCN node number and nothing else (for example, “17” for PM 17 or “5” for APM 5 on the UCN).

Output for a Reference Module object The text returned for the RM object (the module referenced in the entity reference) is either “PMM” for the PM’s PMM module, “APMM” for the APM’s APMM module, “LMM” for the LM’s LMM module, or “IOP#” for an IOP in a PM, PMI, or APM which includes the 1- or 2-digit IOP number in the returned text (for example, “IOP5” or “IOP40”).

Output for a Reference Slot object The text returned for the RS object (the slot referenced in the entity reference) usually consists of just a 2- to 4-character slot type mnemonic followed by a slot number or subrange in parentheses (for example, “RP(3)” for REGPV slot 3 or “TIME(1..2)” for APM box TIMES 1 and 2).

Parentheses are displayed for the RS object (the referenced slot), but not for the existing SLOT object (the referencing slot). These parentheses for the RS object are needed to delimit the slot number or subrange from the slot-type mnemonic in order to allow special patterns to be specified on the Find Names command line for the RN object (for example, “RN TIME(10..*)” to find all references to APM box TIMES 10 or higher). The complete list of all RS output is shown in the table that follows.

Interpreting Reference Slot outputs In the following table, “m” and “n” represent variable numbers or slot numbers that, depending on the context, can range up to 5 digits (99999). For example,

- “STR8(m)” means STR8 variable “m,”
- “TIME(m..n)” means TIME variables “m” through “n,”
- “FL(m)” means flag slot “m,” and
- “NM(m..n)” means numeric slots “m” through “n.”

Also, “SI<slot>” means a slot (1-32) in a Serial Interface (SI) IOP (such as “SI3” or “SI24”) to which an array point is built that maps data in an SI IOP-connected device.

Table 6-15 Reference Slot Output Meaning

Output	Meaning
BOX	A box parameter is being referenced that is not a flag numeric (NN), timer (TMxxx), string (STR8), or time (TIME) parameter. An example is !BOX.NCTLSLOT.
FL(m)	A single box flag is being referenced.
FL(m...n)	A subrange of box flags is being referenced.
NM(m)	A single box numeric is being referenced.
NM(m...n)	A subrange of box numerics is being referenced.

User Reference Information

6.4 Special Objects for UCN Searches

Table 6-15 Reference Slot Output Meaning (continued)

Output	Meaning
STR8(m)	A single box string8 is being referenced.
STR8(m...n)	A subrange of box string8s is being referenced.
TIME(m)	A single box time is being referenced.
TIME(m...)	A subrange of box times is being referenced.
SI<slot>:FL(m)	A single SI flag is being referenced.
SI<slot>:FL(m...n)	A subrange of SI flags is being referenced.
SI<slot>:NM(m)	A single SI numeric is being referenced.
SI<slot>:NM(m...n)	A subrange of SI numerics is being referenced.
SI<slot>:STR8(m)	A single SI string8 is being referenced.
SI<slot>:STR8(m...n)	A subrange of 51 string8s is being referenced.
AI(m)	A single Analog Input slot is being referenced.
AO(m)	A single Analog Output slot is being referenced.
DI(m)	A single Digital Input slot is being referenced.
DO(m)	A single Digital Output slot is being referenced.
RP(m)	A single Regulatory PV slot is being referenced.
RC(m)	A single Regulatory Control slot is being referenced.
DC(m)	A single Digital Composite slot is being referenced.
LG(m)	A single Logic slot is being referenced.
PM(m)	A single Process Module slot is being referenced.
TM(m)	A single Timer slot is being referenced.
DV(m)	A single Device Control slot is being referenced.
AR(m)	A single Array slot is being referenced.

Index

A

- AM-related ports *48*
- APM *1*
- Application Examples
 - Documentation needs *6*
 - Identify resource usage and availability *6*
 - Locate entity references *7*
 - Locate fragmented points *8*
 - Locate program references *7*
 - Multiple references *7*
 - Point deletion *6*
 - Point modification *6*
- Area Database Files (Entities)
 - Entry choices for *34*
 - Navigation to entries *33*
- Area Database Files (Schematics)
 - Entry choices for *32*
 - Navigation to entries *31*
- Area-related ports *53*

B

- Button Configuration, Limitations of Find Names *36*

C

- CM-related ports *48*
- Command files, Find Names in *37*
- Command Processor display *11*

D

- Digital Output command *9*

E

- EC (.EC) files, Find Names in *37*
- Entities
 - Finding deleted entities *36*
 - Valid port entries *48*
 - versus points built *13*
- Error Messages *41*
- Examples
 - Peer search (command file) *38*

F

- Files searched by Find Names *43*

- Find Names
 - Application Examples *6*
 - Concepts and definition *2*
 - display menus, accessing *11*
 - display, accessing *11*
 - display, usage *45*
 - Entry Choices *11*
 - Examples *37*
 - Files searched *43*
 - Items that cannot be found *36*
 - Navigation Paths *11*
 - Types of searches *5*
 - Typical uses *5*

H

- HG-related ports *49*
- History Groups
 - Entry Choices for *28*
 - Navigation to entries *27*
- History-related ports *53*

L

- LM *1*
- LM checkpoint searches *14*

O

- Object, reference, and file relationships *43*

P

- Pattern checker *11*
- Peer-to-peer search example *38*
- Picture Editor-related ports *51*
- Picture Files
 - Entry choices for *30*
 - Navigation to entries *29*
- PM *1*
- Points built/entities differences *13*
- printer *9*

R

- Reference Module object *54*
- Reference Node object *54*
- Reference Slot object *54*

S

- Schematics (custom displays), Limitations of Find Names *36*
- screen display *9*

Index

Search narrowing

- AM-related ports *48*
- Area-related ports *53*
- CM-related ports *48*
- HG-related ports *49*
- History-related ports *53*
- Picture Editor-related ports *51*
- Slots to Search port *47*
- Text file-related ports *52*
- UCN-related ports *49*
- Valid port entries *48*
- Wildcard usage *45*

Searches

- Command file syntax *37*
- From Command File, example *38*
- Initiating *11*
- Types *5*

Slots to Search port *47*

T

text file *9*

Text file-related ports *52*

U

UCN Checkpoint (Entity References)

- EC file command example *37*
- Entry choices for *18*
- Navigation to entries *15*

UCN Checkpoint (Program References)

- Entry choices for *18*
- Navigation to entries *22*

UCN Checkpoint data, accessing

- PM sequences (program references) *22*

UCN Checkpoint Display *17*

UCN searches, special objects for *54*

- Non-UCN entities, output for *54*
- Reference Module object, output for *55*
- Reference Node object, output for *55*
- Reference Slot object, output for *55*
- Reference Slot outputs, interpreting *55*
- Suppressing output columns *54*
- Wildcards *54*

UCN-related ports *49*

Ultimate UCN node, module *54*

User reference information *43*

V

virtual printer *9*

W

Wildcards *45*

READER COMMENTS

Honeywell IAC Automation College welcomes your comments and suggestions to improve future editions of this and other publications.

You can communicate your thoughts to us by fax, mail, or a toll-free telephone call. We would like to acknowledge your comments; please include your complete name and address.

BY FAX: Use this form; detach and fax to 1-602-313-4108.

BY TELEPHONE: In the U.S.A. use our toll-free number 1-800-822-7673 (available in the 48 contiguous states except Arizona; in Arizona dial 1-602-313-5558).

BY MAIL: Use this form; detach, fold, tape closed, and mail to us.

Title of Publication: **Find Names Command**

Issue Date: **11/95**

Publication Number: **SW11-508**

Writer: **S. Cascone**

COMMENTS: _____

RECOMMENDATIONS: _____

NAME: _____ **DATE:** _____

TITLE: _____

COMPANY: _____

ADDRESS: _____

CITY: _____ **STATE:** _____ **ZIP:** _____

TELEPHONE: _____ **FAX:** _____

(If returning by mail, please tape closed; Postal regulations prohibit use of staples.)

Communications concerning technical publications should be directed to:

Honeywell Inc.
Industrial Automation and Control
Automation College
2820 West Kelton Lane
Phoenix, Arizona 85023-3028

FOLD

FOLD

From: _____



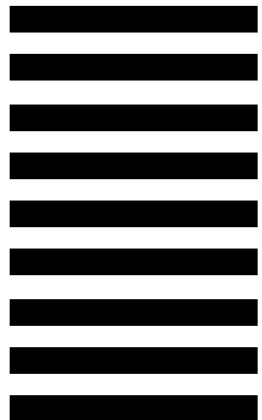
NO POSTAGE
NECESSARY
IF MAILED
IN THE USA

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 4332 PHOENIX, ARIZONA

POSTAGE WILL BE PAID BY ...

Honeywell

**Industrial Automation and Control
2820 West Kelton Lane
Phoenix, Arizona 85023-3028**



Cut Along Line

Attention: Manager, Information Development Services

FOLD

FOLD

Additional Comments:

Honeywell

Industrial Automation and Control
Honeywell Inc.
16404 North Black Canyon Highway
Phoenix, Arizona 85023-3033

Helping You Control Your World