Honeywell

Update Custom Displays

L61214

AG

TotalPlant

Notices and Trademarks

This module supports **TotalPlant** Solution (TPS) system network.

TPS is the evolution of TDC 3000^X.

Honeywell Inc.
Industrial Automation and Control
Automation College
2820 West Kelton Lane
Phoenix, AZ 85053-3028
1-800 852-3211

Table of Contents

INTRODUCTION **1**

Module Overview 1

UPDATING CUSTOM DISPLAYS **3**

When is Custom Display Data Collected? 3
What is a Collection Set? 5
Collection Group 8
Configuring Rate and Group 9
Collection Rate Guidelines 11
Optimizing UCN Data Gathering 13
Summary of Collection Set Guidelines 19
Collection Set Examples 20
Update by Target Action 22
UPDATE Actor 23
DMD_UPD Actor 27

WRITTEN EXERCISE **29**

Configure Collection Set 29
Written Exercise Solutions 31

LAB EXERCISE **33**

Update Actors 33
Configure Collection Set 34
Lab Exercise Solutions 39
Directions 55

PROFICIENCY EVALUATION **57**

Criterion Test 57
Self-Evaluation 59
Directions 60

Figures and Tables

Figure 1 – Normal Display Update Cycle 4
Figure 2 - Data Access of Logical Nodes 5
Figure 3 - Default Collection Entries 6
Figure 4 - Good Collection Set Example—Requires Three Data Requests 20
Figure 5 - Bad Collection Set Example—Requires Nine Data Requests 21
Figure 6 - Format of UPDATE Actor 23
Figure 7 - Example of Target Using Update Actors 25
Figure 8 - Example of Target Using Update 26
Figure 9 – Format of DEMAND UPDATE Actor 27
Figure 10 - Target With DMD_UPD Actor 28
Figure 11 - Lab Exercise Solutions 40
Figure 12 - Lab Exercise Solutions 41
Figure 13 - Lab Exercise Solutions 42
Figure 14 - Lab Exercise Solutions 43
Figure 15 - Lab Exercise Solutions 44
Figure 16 - Lab Exercise Solutions 45
Figure 17 - Lab Exercise Solutions 46
Figure 18 - Lab Exercise Solutions 47
Figure 19 - Test - Update Actor 58

Table 1 - Procedure for Set Collection Command 7
Table 2 – Update Types 9
Table 3 - Collection Group 10
Table 4 - Parameters with a Slow Rate 11
Table 5 - Parameters For Invocation Only 12
Table 6 - Processing of UCN Data Requests 14
Table 7 - Processor Type Vs Point Type 16
Table 8 - Example of Homogeneous Collection Groups 17
Table 9 - Collection Set Guidelines 19
Table 10 - UPDATE Actor Examples 24

Acronyms

AM..... Application Module
APM ..... Advanced Process Manager
CG .....Computer Gateway
HG .....Hiway Gateway
IDB..... Intermediate Data Block
LCN..... Local Control Network
LM.....Logic Manager
NIM .....Network Interface Module
PM..... Process Manager
UCN .....Universal Control Network
US..... Universal Station

References

Publication Title	Publication Number	Binder Title	Binder Number
Picture Editor Reference Manual	SW09-650	Implementation/Engineering Operations-2	TPS 3032-2
Actors Manual	SW09-655	Implementation/Engineering Operations-2	TPS 3032-2

Introduction

Module Overview

Introduction

This course module discusses several Picture Editor functions available to you to update data in a custom display:

- the Set Collection command,
- the Optimize Collection Groups command,
- the Update actor, and
- the Demand Update actor

Objectives

Given a prebuilt custom display, the Picture Editor manuals, and guidelines for defining display update rates

- determine the most efficient update rates and groupings for variables in the custom display,
- use the Set Collection command and the Optimize Collection Groups command to configure update rates and groups,
- determine when and where in a target action the Update actor or Demand Update actor is required, then configure the actor.

Sample test items

Configure the Collection Set of display REACT_OV according to the guidelines in this course module. Compare your results to the configuration solutions. Be able to explain any discrepancies to your course manager.

Updating Custom Displays

When is Custom Display Data Collected?

Parts of a display

A Custom display contains two types of objects:

- constant - static objects, such as lines, solids, or text without conditions that never change; they do not have values, variants, or conditions associated with them.
- updating- dynamic objects, such as values, variants, conditions, and bars that may change based on system variables.

Display invocation

Display execution occurs:

- at invocation, and
- every four seconds following invocation (screen update).

When a custom display is invoked, the Universal Station (US) accesses all data used in the display, then evaluates the display's conditions, variants, value expressions, and bars to reflect the current data:

1. constant objects are written to the screen only once,
2. all variable data is collected,
3. conditions, variants, and values are evaluated by using collected data, and
4. all dynamic parts of the display are drawn on the screen.

After invocation

Following display invocation, the Universal Station updates the display every four seconds or, if the FAST key is pressed, the station updates display variables assigned to the Fast Collection Group once every second.

The display update happens as follows:

1. all variable data scheduled for collection on the current cycle is collected (some data may be on a slow schedule and is collected every other cycle, for example),
2. conditions and variants are evaluated using the collected variable data regardless of when it was collected, and
3. all dynamic parts of the display are redrawn on the screen.

Continued on next page

When is Custom Display Data Collected? Continued

Data collection example

The normal custom display update cycle is four seconds. Every four seconds the display's variants and conditions are evaluated based on the collected data currently available to the display.

In Figure 1, F200.PTINAL has a user-configured Collection Rate of eight seconds; the variant containing the variable is evaluated on the normal 4-second cycle.

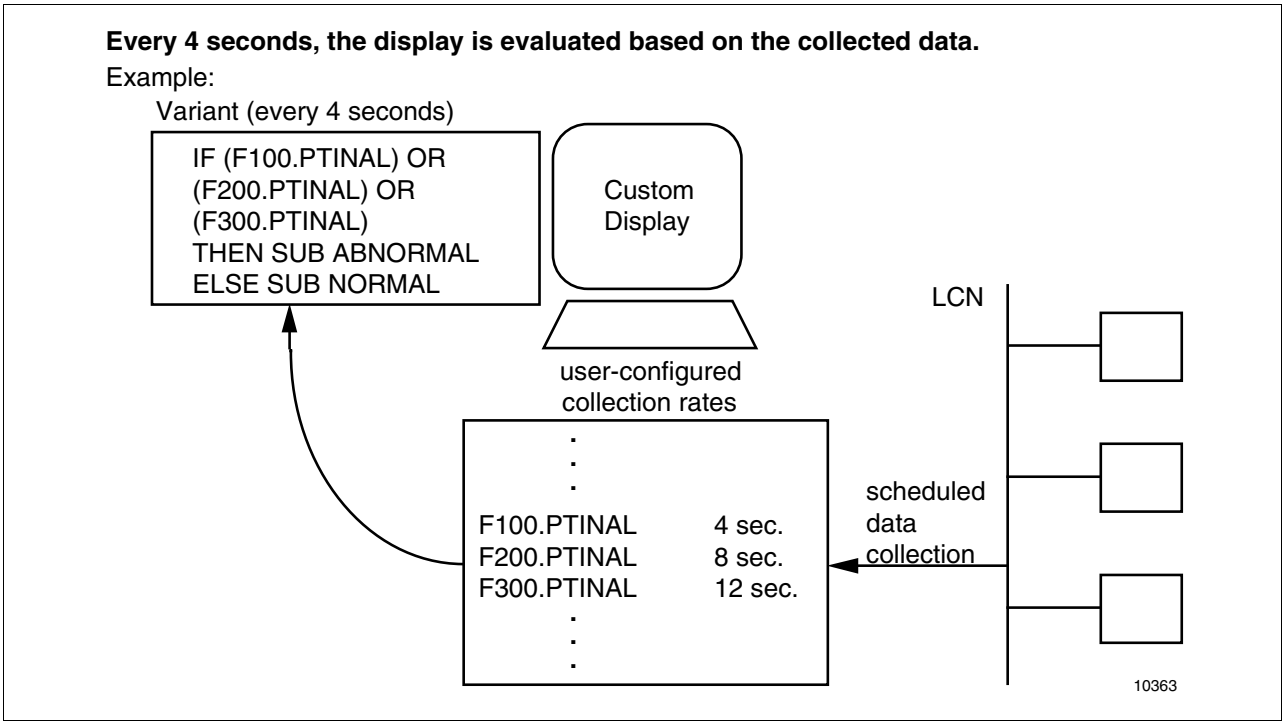


Figure 1 – Normal Display Update Cycle

Reducing system loading

An important concept associated with reducing the impact of a custom display on overall system loading is reduction of the rate of data gathering. If designed well, custom displays respond almost immediately to operator requests and place a minimal load on Gateways, Application Modules, and the LCN in general.

Initial planning of any custom display can be thoroughly optimized using these Picture Editor functions:

- Collection Set configuration for the display and its overlays,
  - Collection Rate
  - Collection Group
- Target action using UPDATE actor, and DMD_UPD actor.

Continued on next page

What is a Collection Set? Continued

The list of variables used in a custom display or overlay is referred to as the Collection Set. It provides a means to sort custom display variables by:

- Collection Group, and
- Collection Rate.

The Picture Editor's Set Collection command lists the variables used in a custom display or overlay and provides entry ports to configure each variable's Collection Group and Collection Rate.

Sorting data

Collection Groups are mainly used for sorting data access requests. To aid the sorting performed by the US and the LCN data owners, you should sort display variables into Collection Groups by:

- Collection Rate,
- Logical node (process network or AM/CG unit), and
- in the case of the UCN, by process-connected node.

Custom displays request data from the LCN by using the data access mechanism called a data request message. As shown in Figure 2, the US sorts these data request messages by logical node. A logical node for the AM is a unit. A logical node for the gateways is a process network. Depending on the data owner, requests may be sorted further when received by the gateway (NIM or HG). You aid this sorting process by configuring Collection Groups

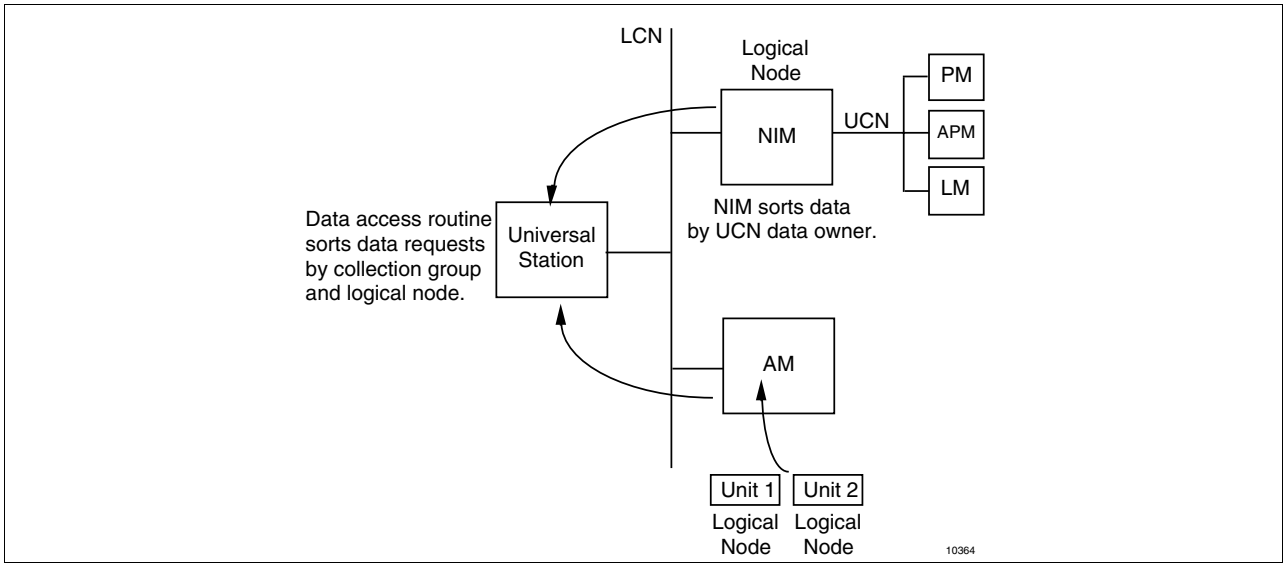


Figure 2 - Data Access of Logical Nodes

Continued on next page

What is a Collection Set? Continued

Collection Set display

Figure 3 shows an example of the display that appears after you enter the initial Set Collection (S C) command.

Entry ports for Collection Rate and Collection Group ID appear for all variables referenced in the custom display or overlay. Initially, all variables are defaulted to a 4-second update rate and are all grouped into one Collection Group:

Defaults:

Collection Rate - 1 (4-seconds)

Group ID - 0

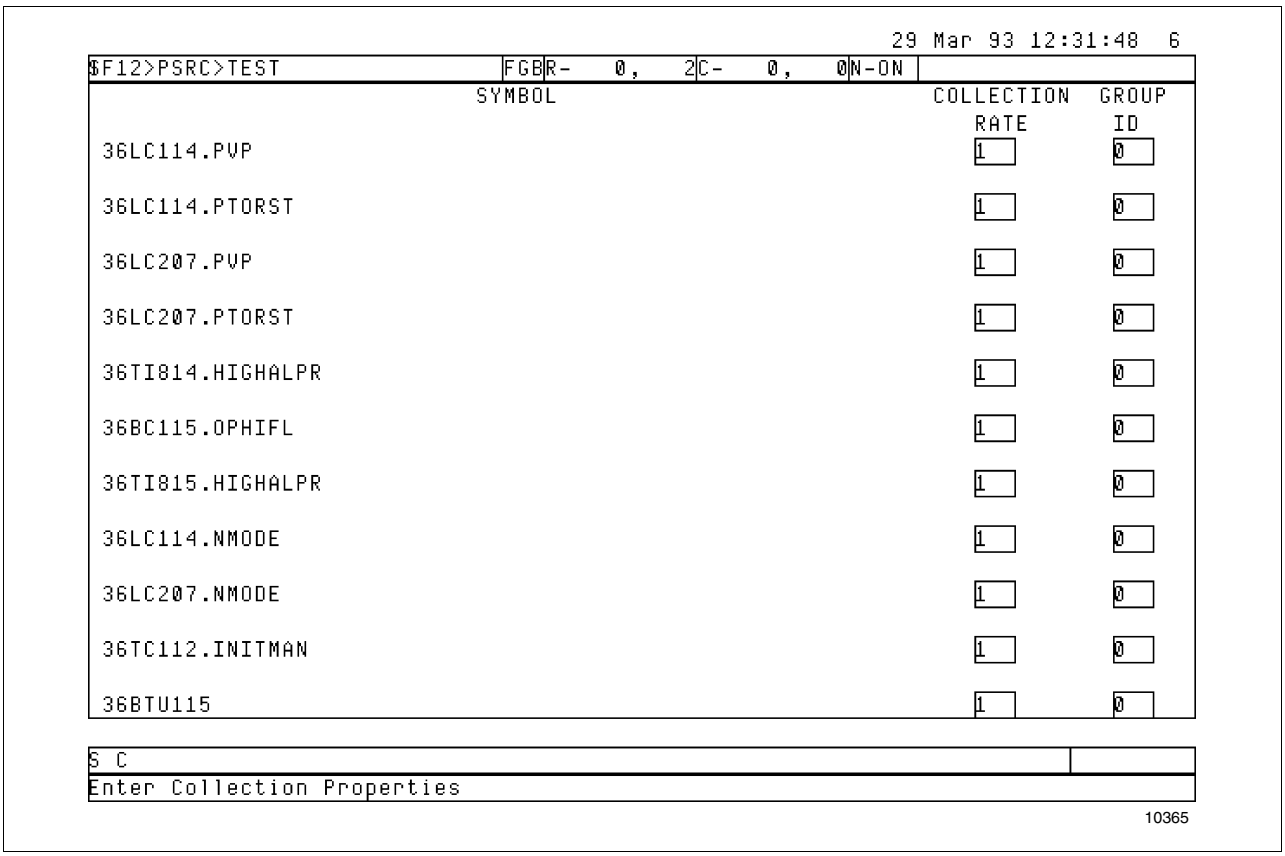


Figure 3 - Default Collection Entries

Continued on next page

What is a Collection Set? Continued

Procedure

Table 1 describes the procedure to enter the Set Collection command:

Table 1 - Procedure for Set Collection Command

Step	Action
1	Type in and enter the Set Collection command: S C Result: A list of all the variables used in the custom display appears with two entries for each variable (see Figure 3).
2	If the Collection Set has more than one page (11 variables per page), press [PAGE FWD] or [PAGE BACK] to view the entire list.
3	Type-in these "collection properties" for the variables, as necessary: • Collection Rate • Group ID NOTE: To cancel the S C command, press the [CANCEL] or [DEL] key. The collection properties revert to their previous specifications.
4	Press [ENTER] to enter the Collection Set data.

Set COLLECTINH command

With R600, the Collection Rate and Group ID of variables, as defined in a subpicture, can be inherited in the destination (main) picture. The syntax for the command is:

SET COLLECTIONINH <FAST/MAIN/SLOW/SUBPICTURE>
where:

- FAST indicates the faster of the collection rates for the variable (main picture or subpicture) is to be used
- MAIN indicates the collection rate for the variable defined in the main picture is to be used
- SLOW indicates the slower of the collection rates for the variable (main picture or subpicture) is to be used
- SUBPICTURE indicates the collection rate defined for the variable in the subpicture is to be used.

MAIN is the default value and gives the same functionality that existed before R600.

Collection Group

Why use groups?

Collection Groups provide a means for you to sort custom display variables in order to optimize data collection.

All variables in a custom display are initially defaulted to Collection Group zero (0), which means that the data access mechanism of the Universal Station sorts the data requests without any “help” from you.

There is no other function in the Picture Editor that does more to optimize display response or control LCN loading than properly built Collection Groups; however, Collection Groups cannot overcome a bad display design (the course module “Designing Custom Displays” discusses guidelines for display design).

Collection Groups reduce gateway loading by limiting a single LCN data collection request to only those variables assigned to a specific Collection Group. By reducing gateway activity, the gateway responds faster, consequently, display response increases.

Recommendation

If a custom display has more than 200 parameters and no user-established Collection Groups, the display performance will not be optimal.

An LCN data request message contains blocks of up to 200 parameters per logical node.

Because the data request messages are created from Collection Groups established within the custom display, it is not useful to have more than 200 variables in a single Collection Group.

Configuring Rate and Group

Types of update

The Collection Rate is a screen update multiplier (0 - 255, default = 1). Table 2 lists the types of update available through configuration of the Collection Rate and Collection Group.

Table 2 – Update Types

<table>
<tr><th>Update</th><th>Description</th><th>Rate</th><th>Group</th></tr>
<tr><td>invocation only (no update)</td><td>For variables that do not change, such as EUDESC, or change slowly or infrequently.</td><td>0</td><td>0 - 245</td></tr>
<tr><td>scheduled update</td><td>For variables that require a regularly scheduled update, you should specify a collection rate, according to how fast the variable changes.

Update Cycle = Collection Rate x 4 sec.

Examples:
<table>
<tr><th>Rate</th><th>Update Cycle</th></tr>
<tr><td>0</td><td>only once when first displayed</td></tr>
<tr><td>1</td><td>every 4 seconds (default)</td></tr>
<tr><td>2</td><td>every 8 seconds</td></tr>
<tr><td>3</td><td>every 12 seconds</td></tr>
<tr><td>.
.</td><td>.
.</td></tr>
<tr><td>15</td><td>once per minute</td></tr>
<tr><td>.
.</td><td>.
.</td></tr>
<tr><td>255</td><td>every 17 minutes</td></tr>
</table></td><td>1 - 255</td><td>0 - 245</td></tr>
<tr><td>fast update</td><td>For critical variables that at certain times require a faster update than the scheduled rate, you should specify FST as the Group ID.

The fast update rate is 1 sec.

The Fast update is activated by using the FAST button; otherwise the scheduled collection rate applies.</td><td>1 - 255</td><td>FST</td></tr>
<tr><td>demanded update</td><td>For variables that need to be updated during a target action, you should use the Update or Demand Update actors in the target action (discussed later).

NOTE:
Operates independent of Collection Rate.</td><td>see NOTE</td><td>0 - 245</td></tr>
</table>

Continued on next page

Configuring Rate and Group, Continued

Group ID

The Group ID specifies the Collection Group to which the variable is associated (default = 0 or CZ).

Table 3 describes the possible Collection Group entries.

Table 3 - Collection Group

Entry	Range
0-245	Assigns variable to a specific Collection Group (127 groups max. per custom display). NOTE 1: Numbering of groups is arbitrary within the 0-245 limit allowed by the Picture Editor. The wider range of numbers allows more flexibility when assigning group numbers. NOTE 2: Group numbers do not represent any priority. Group 0 and group 245 have equal priority; they are used merely to identify different groups.)
FST	Assigns variable to the Fast Collection Group, which updates every *1 second* after the [FAST] key is pressed by the operator or activated through the INITIAL action QUE_KEY[FAST].
CZ	CZ is automatically shown in the Group ID port for variables appearing in the display's standard Honeywell change zone. This identifier helps you locate change zone variables to change their update rate, if desired.

Limits

The maximum number of variables in a single Collection Group is 512. If there are over 512 variables, this compiler error is given:

```
Collection Set Overflow.
```

Collection Rate Guidelines

Basic rule

The default update rate for custom displays is every four seconds, but it is not always necessary to collect data at that rate. You should consider the possibility of data collection at longer periods, particularly for data not expected to change rapidly.

The level of a large tank, or the temperature of a relatively slow process may not change significantly over periods of tens of seconds or even minutes, so there is no need to update values such as these every four seconds.

Consider the case of a bar graph indicating a tank level of 0-40,000 gallons, where the bar is 40 pixels high; one pixel (minimum movement) represents 1000 gallons. Can the level change 1000 gallons in 4 seconds? Could the bar be updated only once a minute and still be useful to the operator?

Slow parameters

Many variables might be left at a rate of one (4-second update), but variables that rarely change, such as those listed in Table 4, can be set to a Collection Rate of 15 (1-minute update), for example.

Table 4 - Parameters with a Slow Rate

Parameter Name	Description
PTEXECST	Point Execution State (Active/Inactive)
ALENBST	Alarm Enable/Disable State
K, T1, T2, TD, etc.	Tuning Constants
PVEUHI and PVEULO	Range Limits
PVHI and PVLO, etc.	Alarm Limits
SPHI and SPLO	Setpoint Limits
Configuration	Point Configuration Data

Continued on next page

Collection Rate Guidelines, Continued

Invocation only parameters

Collection Rate zero is a special case and very important. Any variable assigned to Collection Rate zero is collected at display callup (invocation) and is not updated on a collection schedule. Table 5 lists the variables suitable for no scheduled update.

Table 5 - Parameters For Invocation Only

Parameter Name	Description
NAME	Entity Name
EUDESC	Engineering Unit Descriptor
PTDESC	24-character Point Description
KEYWORD	8-character descriptor
STATE1 and STATE2	Digital State Descriptors
NMODE	Normal Mode
NMODATTR	Normal Mode Attribute
PERIOD	Processing Period
CYCLE	Scan Cycle
(other)	parameters not likely to change for the duration the display is on the screen

Fast updates

You can choose to put a few parameters in a fast update group that allows the operator or the DEFINE INIT target to place the parameters on a 1-second update.

For “fast update” parameters in a custom display, you can choose to build a custom change zone in a small part of the display and put the fast update items in this change zone.

Put the slowest possible update rate on all other parameters in the picture.

Entity names

Entity names of system points should be set to Collection Rate zero (0). For example, the last variable listed in the Collection Set in Figure 3 is the name of a process point - 36BTU115. Its Collection Rate is currently set to the default of one (4-second update); it should be changed to zero (update on invocation only).

Optimizing UCN Data Gathering

UCN data access concepts

Each LCN data request message to a NIM requires an overhead of NIM processing time to read the request and sort it for processing. This overhead is independent of the size of each request. Optimization of UCN data collection requires data request messages be homogeneous with respect to the location of the requested parameters. This requirement arises from the way in which parameter requests are processed in the PM, APM, and HPM.

A completely homogeneous UCN request is generated by a Collection Group that contains parameters from

- a single UCN node, and
- a single processor type (control or I/O) within that node.

However, if these criteria are applied to more than a few UCN nodes, the schematic will create a large number of data request messages, placing a heavy transaction load on the NIM.

Since the NIM sorts by node, when more than a few UCN nodes are involved, display builders need only be concerned about grouping by processor type.

The combination of user grouping by processor type and NIM sorting by UCN node creates a completely homogeneous UCN request message without placing a heavy transaction load on the NIM.

To optimize the UCN data gathering efficiency of a custom display

- group data requests by processor type and update rate.
- use parameters at the highest possible data owner level, and

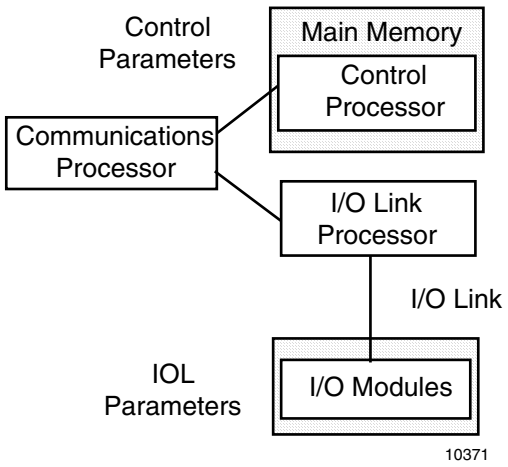
Optimizing UCN Data Gathering, Continued

Grouping UCN variables

Table 6 describes the effects of grouping UCN variables (data requests) by I/O and Control processor type.

Table 6 - Processing of UCN Data

Parameter Request	Description
control parameters	Obtained directly from memory. Relatively small cost in processing time in the node's communications processor.
I/O parameters	The communications processor creates a second request to the I/O Link processor with an associated cost in processing time. Overhead cost occurs each time a new I/O Link parameter is encountered in a data request message. In a message consisting of only I/O Link parameters, the number of I/O Link processor requests is considerably reduced, because the communications processor can package several parameters into a single I/O Link request.
Mixture	If there is a mixture of control and I/O Link parameters, the processor uses a significant amount of time "shifting gears" between the two types.



Continued on next page

Optimizing UCN Data Gathering, Continued

Optimize Collection Groups Command

With R530, the Picture Editor has a new command, Optimize Collection Groups, that will sort UCN variables in the collection set by logical node and processor type. The syntax for the command is:

OPTIMIZE (O) (OPT) <UCN> <Control Group> <I/O Link Group>

Where: UCN is the UCN process network number. The valid range is 1-20.

Control Group is the collection group number that contains parameters from the control processor. The valid range is 0-245.

I/O Link Group is the collection group number that contains parameters from the I/O link. The valid range is 0-245.

If a point referenced in the collections set does not exist, or if the data owner of the point is not responding, an error file will be created. The error file name will be the schematic file name and the file extension will be .ER. The error file will be created in the directory that contains the schematic source file. The content of the error file can be viewed from the Command Processor using the Print command.

Optimizing UCN Data Gathering, Continued

Determining the highest UCN data owner

In order to build custom displays that use parameters from the highest possible data owner, you must determine what NIM-resident parameters can be used instead of parameters resident in the process device.

For example:

Use HIGHAL (NIM-resident) instead of PVHIFL(process device-resident).

Shown below is an excerpt from a Parameter Reference Dictionary. In this excerpt, the NIM is shown to be the data owner of the HIGHAL parameter (PtRes: NIM).

HIGHAL

Type:	**E:ALMTYPE**
Lock:	**View**
Default:	**NoAlarm**
PtRes:	**NIM**

Determining UCN processor type

When configuring Collection Groups, the processor type can be determined from the point type (see Table 7).

Table 7 - Processor Type Vs Point Type

I/O Processor	Control Processor
Point Types: • Analog Input • Analog Output • Digital Input • Digital Output • Pulse Input • STIM (smart transmitter interface module)	Point Types: • Flag, Numeric, Timer • Digital Composite • Logic • Process Module • Regulatory (Control and PV) • Time (APM/HPM) • String (APM/HPM) • Device Control (APM/HPM) • Array (APM/HPM)
NOTE: All point types have some NIM-resident parameters. Group them with either the control or IOL parameters, because they have no effect on UCN node communications loading.	

Continued on next page

Optimizing UCN Data Gathering, Continued

Example

For this example, assume the UCN contains five PMs (node numbers 21, 23, 25, 27, 29), and a custom display is being created that requests both control and IOL data from all of the PMs. Some of the data never changes (such as descriptors and point names).

In this example, it has been determined that the best overall display operation and system loading is achieved using three rates:

- invocation only,
- the default 4-second update,
- and an 8-second update.

Table 8 lists the groups that were established.

In this example, at invocation, the NIM will receive five requests, then two requests every four seconds and four requests every eight seconds.

If the parameters were all assigned to the same Collection Group, the LCN would make one request to the NIM.

Table 8 - Example of Homogeneous Collection Groups

Rate	Group	Membership
0	0	All nonupdating parameters (invocation only)
1	101	Control parameters collected every 4 seconds
1	102	IOL parameters collected every 4 seconds
2	201	Control parameters collected every 8 seconds
2	202	IOL parameters collected every 8 seconds

Recommendations

The creation of “pure” homogeneous groups works well when the display

- contains a single update rate in addition to the invocation-only group, and
- accesses a small number of UCN nodes.

Continued on next page

Optimizing UCN Data Gathering, Continued

Limitations

Increased Bookkeeping

Achieving “pure” homogeneous requests for any but the simplest display requires considerable effort on the part of the designer just to perform the bookkeeping involved with the SET COLLECTION function.

Too Many Groups

Increasing the number of groups (and thereby the number of data request messages) also has an effect on the NIM. Reducing the load on the individual UCN nodes by increasing the number of groups can severely increase the load on the NIM without changing the number of parameters being serviced.

For example, adding a fourth rate to Table 8, such as parameters that update every 12 seconds, adds 2 more groups.

ATTENTION

ATTENTION—With or without Collection Groups, if there are more than eight UCN nodes in the custom display (16 nodes for R410), the NIM’s limit on outstanding UCN requests causes some additional delays and overheads.

Summary of Collection Set Guidelines

Summary

Table 9 lists the Collection Set guidelines that improve custom display performance and minimize custom display loading of nodes.

Table 9 - Collection Set Guidelines

Guideline	Comment
COLLECTION RATE: Set the update rate of constant parameters to zero.	There is a large class of data (entity names, descriptors, engineering units, etc.) that do not need to be updated at all; this data can be collected once, at custom display invocation.
Set the slowest possible update rate on each variable.	Determine the Collection Rate for each variable in the display by looking at how fast the actual data changes. Don't collect the temperature of Lake Michigan every four seconds!
Set each collection group to have only one Collection Rate.	A Collection Group containing more than one Collection Rate is inefficient.
DON'T use "too many" rates.	There is operational overhead associated with each new Collection Rate; therefore, Honeywell recommends using only a few different rates. Realistically, only a few are required to reduce the LCN loading.
COLLECTION GROUP: Group by Collection Rate and logical node.	If possible, variables assigned to a Collection Group should have the same rate and the same logical node address: • All variables should be on the same process network or, for an AM or CG, in the same unit.
Use the Optimize Collection Groups command to sort UCN I/O Link variables from UCN control variables.	To configure a UCN Collection Group it should contain parameters from: • a single UCN, and • a single point type (control or I/O Link).
DON'T use "too many" groups.	Use as few groups as possible (no more than 15). The NIM and HG handle up to 15 data access requests in a message queue; consequently, you can overload the gateway using **too many** Collection Groups. Group as many variables as possible in the same Collection Group. Do not group invocation-only parameters Don't use groups at all if the variables are at four seconds and are from the same logical node.
DON'T put too many variables in one group.	Limit the number of parameters to no more than 200 entries per Collection Group; LCN data access messages are limited to a maximum of 200 parameter requests per message anyway.
CAUTION Before R420 Only Group all ACKSTAT collectors in the same group with same rate.	Before R420, if ACKSTAT parameters are divided between Collection Groups and/or are given different Collection Rates, the references will not work properly and ultimately will cause the US to crash because of a lack of heap memory.
For R420 and later, DON'T use more than 20 rates/groups with ACKSTAT references.	For R420 and later, a picture or overlay may not have more than 20 Collection Rate/Group ID combinations with ACKSTAT references

Collection Set Examples

Description

If you need three Collection Rates and three Collection Groups, you can cause the US to perform three data requests or nine data requests, depending on how you organize the Collection Set variables.

Good example

In Figure 4, the Set Collection command (S C) was executed to display the collection set variables. The Collection Set has been configured with these Collection Rates:

0 = variables update at invocation only

1 = variables update every four seconds

2 = variables update every eight seconds

Notice that each Collection Group in the example contains variables assigned to the same Collection Rate, consequently, the US performs only one data request per Collection Group.

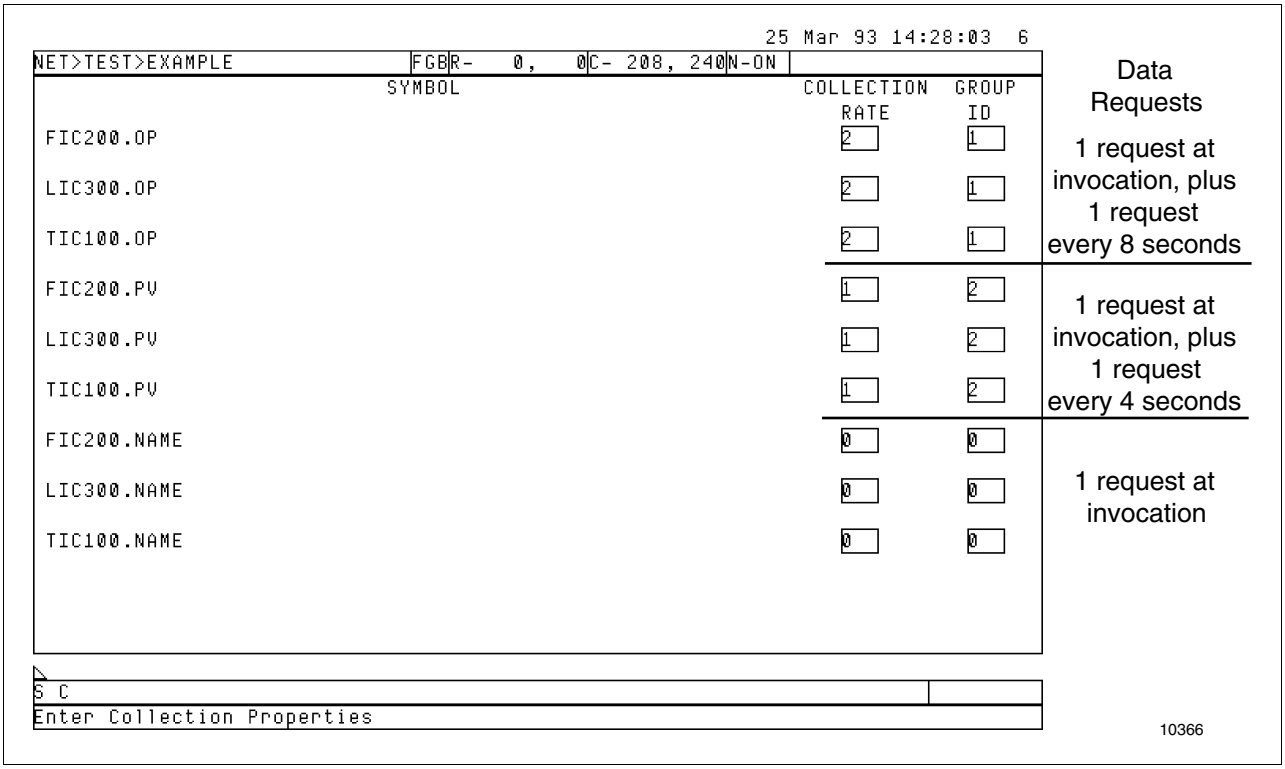


Figure 4 - Good Collection Set Example—Requires Three Data Requests

Continued on next page

Collection Set Examples, Continued

Bad example

Figure 5 shows a poorly configured Collection Set; that is, each Collection Group contains variables from three different Collection Rates. The US performs nine data requests to collect this data:

Data Requests = # (Rate x Group) Assigned

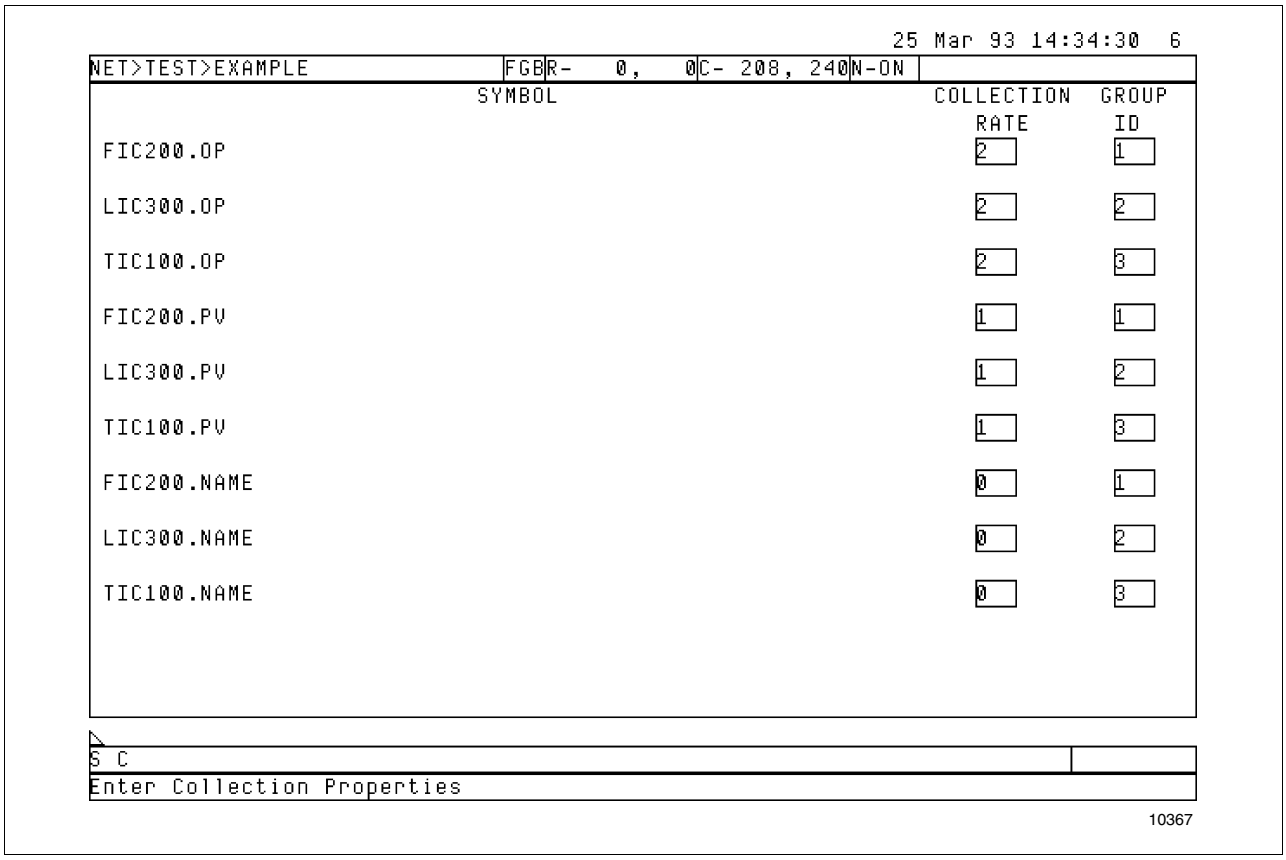


Figure 5 - Bad Collection Set Example—Requires Nine Data Requests

Update by Target Action

Update Actors

You would use update actors in a target action to immediately reflect on the display changes made by the target. Without update actors, the display is updated during the normal 4-second cycle.

Update actors do the following:

- update *all* variables in a specific screen region (UPDATE actor),
- update only *DDB variables* in a specific screen region (UPDATE actor),
- update variables assigned to a specific Collection Group and in a specific screen region (UPDATE actor), and
- update all variables in a specific screen region (DMD_UPD actor).

Screen regions

The screen regions that can be specified in update actors are:

0 - region in which target is located,

1 - prompt region (top line),

2 - main display,

3 - standard overlay (is displayed using MULT_OV actor), and

4 - background overlay (lower priority update, is displayed using BACK_OV actor).

UPDATE Actor

Description

The UPDATE actor is used in a target action to request collection and display of DDB variables, all variables, or grouped variables in a specific region of the screen, as would be necessary to

- allow the operator to demand an update of “slow” values, and
- immediately see a change made by the target or the operator (if the variable is on a slow or unscheduled update, or if it is desirable to see the change sooner than the 4-second normal update cycle).

Format

Specify two parameters in the Update actor:

- the screen region of the variables to be updated, and
- the variables to be updated.

Figure 6 shows the format of the Update Actor.

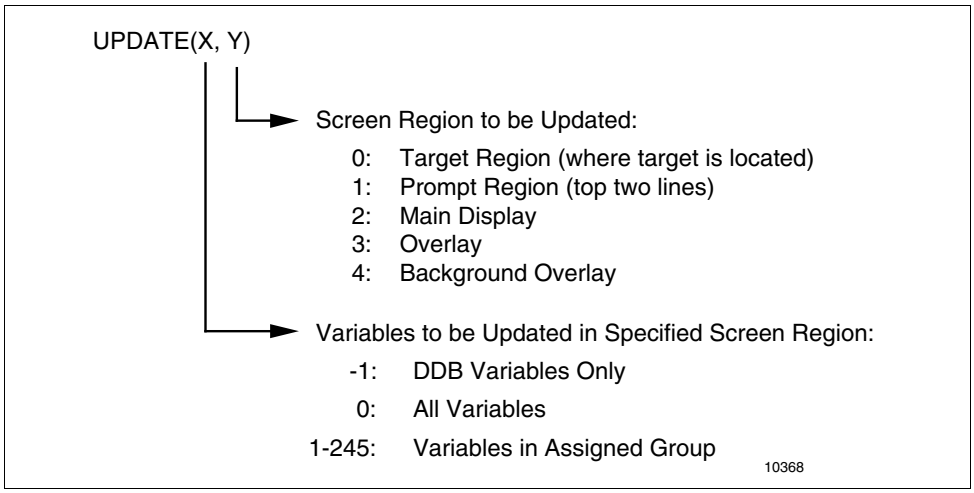


Figure 6 - Format of UPDATE Actor

Background overlay

The Background Overlay (screen region 4) is updated at a lower priority than the other screen regions. Typically, it is used to display internetwork parameters accessed by a Network Gateway.

DDB variables

Display Database variables, such as Reals, Integers, Booleans, Strings, Enumerations, and Times, are assigned to Collection Group -1.

Continued on next page

ATTENTION

ATTENTION—The Collection Group IDs CZ and FST cannot be used by the Update actor. The Update actor should not be used in DEFINE INITIAL action.

Actor examples

Table 10 describes several UPDATE actor examples.

Table 10 - UPDATE Actor Examples

Example	Description
`UPDATE(0,3)`	Updates all variables (0) located in overlays (3), other than the background overlay.
`UPDATE(-1,0)`	Updates DDB variables (-1) in the region where the target is located (0).
`UPDATE(5,2)`	Updates variables that are assigned to Collection Group 5 and are located in the main display (2).

Continued on next page

UPDATE Actor, Continued

Target examples

Figures 7 and 8 contain examples of targets with Update actors.

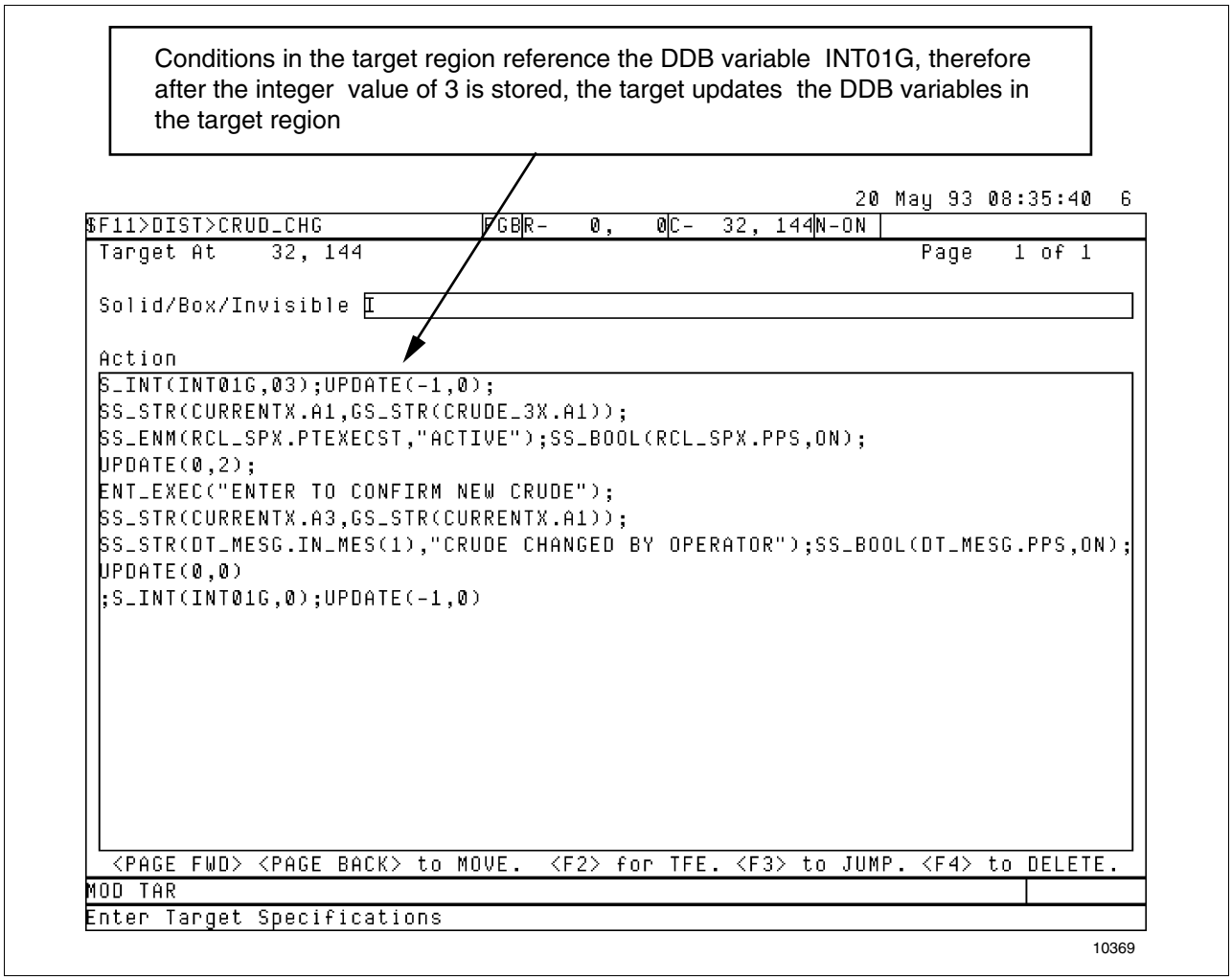


Figure 7 - Example of Target Using Update Actors

Continued on next page

UPDATE Actor, Continued

Target examples, continued

```
                                                    20 May 93 10:01:15  6
$F11>DIST>CRUD_CHG                 FGBR-   0,   0C-  32,  80N-ON
 Target At    32, 80                                          Page   1 of 1

 Solid/Box/Invisible B

 Action
S_INT(INT01G,9);UPDATE(-1,0);
RS_SYS(CHG_SPX.D(3),25,1,3,"SP CHANGE PERIOD IN MIN",TRUE,1);
SS_ENM(CHG_SPX.PTEXECST,"ACTIVE");
SS_STR(DT_MESG.IN_MES(1),"AVSP CHANGED BY OPERATOR");SS_BOOL(DT_MESG.PPS,ON);
SS_REAL(DT_MESG.IN_PRIO(1),2.0);
S_INT(INT01G,0);UPDATE(-1,0);
UPDATE(0,0);
SS_ENM(SPC13.S1REQSTS,"RESTRT");SS_ENM(SPC14.S1REQSTS,"RESTRT");
SS_ENM(SPC15.S1REQSTS,"RESTRT");SS_ENM(SPC16.S1REQSTS,"RESTRT");
SS_ENM(SPC17.S1REQSTS,"RESTRT");SS_ENM(SPC18.S1REQSTS,"RESTRT");
SS_ENM(SPC19.S1REQSTS,"RESTRT");SS_ENM(SPC20.S1REQSTS,"RESTRT");
SS_ENM(SPC21.S1REQSTS,"RESTRT");SS_ENM(SPC12.S1REQSTS,"RESTRT");
PROMPT("WARNING: SPQC DATA RESET");
DELAY(0,2,0);PROMPT_C;

  <PAGE FWD> <PAGE BACK> to MOVE.  <F2> for TFE. <F3> to JUMP. <F4> to DELETE.
MOD
Enter Target Specifications
```

10370

UPDATE(X, Y)

Screen Region to be Updated:

- 0: Target Region (where target is located)
- 1: Prompt Region (top two lines)
- 2: Main Display
- 3: Overlay
- 4: Background Overlay

Variables to be Updated in Specified Screen Region:

- -1: DDB Variables Only
- 0: All Variables
- 1-245: Variables in Assigned Group

10368

Figure 8 - Example of Target Using Update

DMD_UPD Actor

Description

The Demand Update actor can be used to demand an update of a specific screen region. Do not use the Demand Update actor in a DEFINE INITIAL action.

Format

Figure 9 shows the format of the DMD_UPD actor.

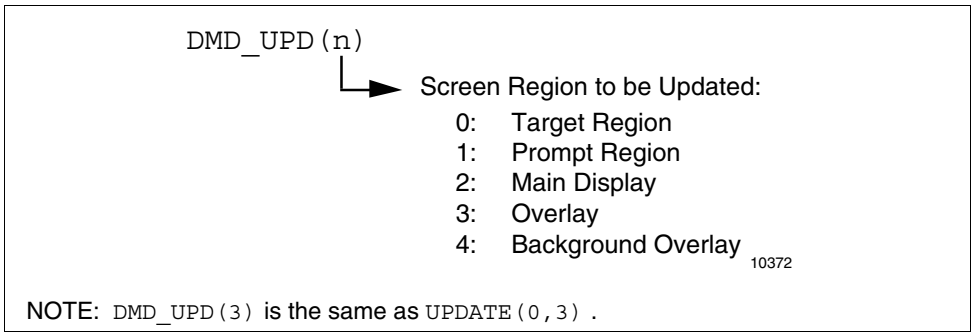


Figure 9 – Format of DEMAND UPDATE Actor

Example

In the example in Figure 10, the Demand Update actor updates a change zone called in as an overlay:

- `UPDATE (0,0)` updates all variables in region where target resides,
- `DMD_UPD(3)` updates overlay (user change zone) on invocation.

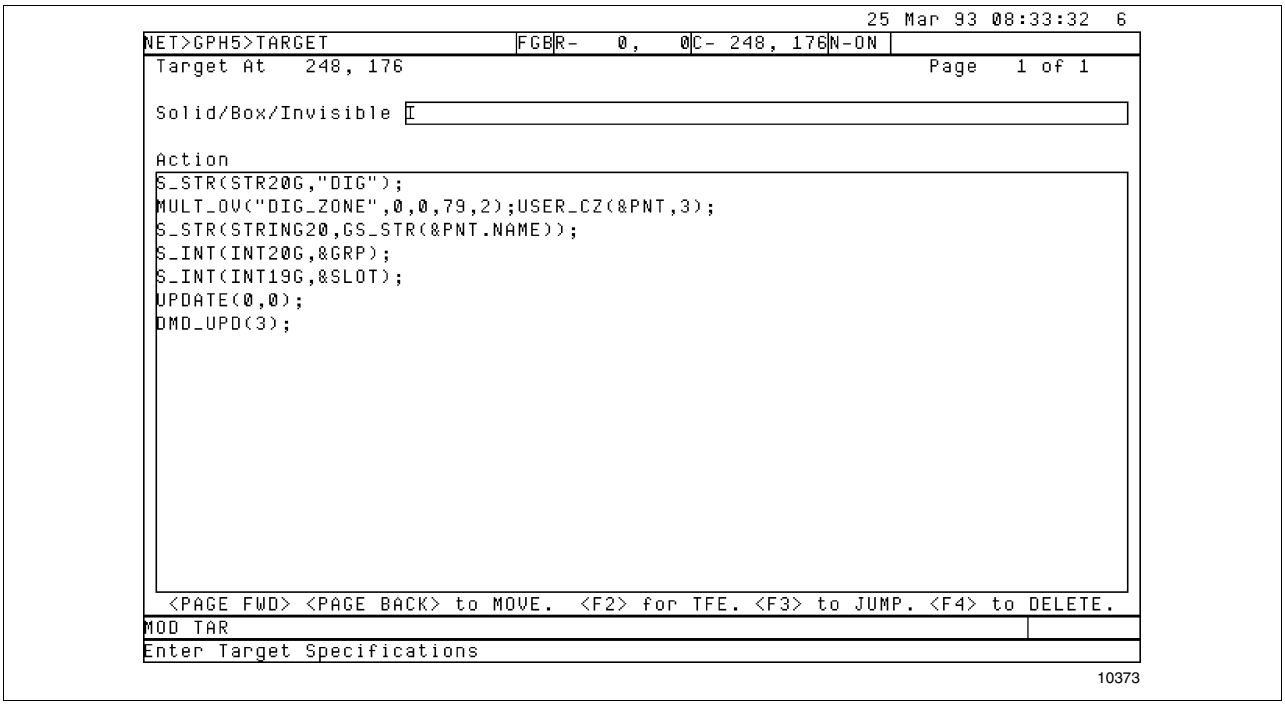


Figure 10 - Target With DMD_UPD Actor

Written Exercise

Configure Collection Set

Instructions

This written exercise asks you to apply the guidelines discussed in this course module. Assign Rates and Groups to the variables below. Use the table on the next page to help you estimate the process change rate. Assign a 12-second rate to variables that seldom change.

Variable Name	Rate	Group	Notes
$STNNUM(see note)			
ACKSTAT(FC104)			
ACKSTAT(LC103)			
BOOL05G			
FC104.K			
FC104.MODE			
FC104			
FC104.OPHILM			
FC104.OP			
FC104.OPLOLM			
FC104.PV			
FC104.SP			
FC104.T1			
FY105.PVTV			
FY106.C1			
FY106.D			
HS6501.PV			
HS6502.PV			
LC103.K			
LC103.MODE			
LC103.OP			
LC103.PV			
LI103			
LI103.PVHITP			
LI103.PVLOTP			
LI103.PV			
NOTE: The system DDB variable $STNNUM is an integer representing the station number on which the custom display is running (Table A-3 of the Actors Manual lists other system DDB variables).			

Continued on next page

Configure Collection Set, Continued

Point descriptions

The following table describes the purpose of each point listed in the Collection Set on the previous page. To help you determine how to assign Collection Groups, the point type and residency is also provided.

Point	Point Type	Residency
FC104 boiler feedwater flow	PID	Process Node 7 or Box 7 Network 1 or Data Hiway 1
FY105 boiler feedwater summer	Regulatory	Unit 39
FY106 % load calculator	Regulatory	Unit 39
HS6501 feedwater pump	Digital Composite	Process Node 7 or Box 7 Network 1 or Data Hiway 1
HS6502 feedwater pump	Digital Composite	Process Node 7 or Box 7 Network 1 or Data Hiway 1
LC103 drum level controller	PID	Process Node 10 or Box 10 Network 1 or Data Hiway 1
LI103 drum level indicator	Analog Input	Process Node 10 or Box 10 Network 1 or Data Hiway 1

Parameters

If you are curious about a variable listed in the Collection Set on the previous page, look it up in the appropriate parameter manual, depending on where the parameter resides:

Residency	Binder and Manual
Data Hiway	TPS 3034-1, Hiway Gateway Parameter Reference Dictionary
AM	TPS 3035-2, Application Module Parameter Reference Dictionary
UCN	TPS 3040-2, PM Parameter Reference Dictionary TPS 3042-2, APM Parameter Reference Dictionary TPS 3066-2, HPM Parameter Reference Dictionary
DDB	TPS 3032-2, Actor's Manual, Appendix A

End of Exercise

Written Exercise Solutions

Groups and rates

Check your answers against the recommendations in the table below (variables are listed by Collection Group number). If you have any questions about the answers, ask your course manager for assistance.

Variable Name	Rate	Group	Notes
$STNNUM	*0*	*0*	*Station number—update at invocation only*
BOOL05G	*0*	*0*	*Global DDB variable. Assign a rate if it needs a scheduled update; that is, the variable DDBs do not need to be grouped unless you display them as values.*
FC104	*0*	*0*	*Entity name—update at invocation only*
LC103	*0*	*0*	*Entity name—update at invocation only*
FY106.C1	*0*	*0*	*Constant in calculation—never changes*
FY106.D	*0*	*0*	*Constant in calculation—never changes*
ACKSTAT(FC104)	*1*	*2*	*After R420—Group ACKSTAT variables with normal operating variables.* *Before R420—Group ACKSTAT variables together in the same group.*
ACKSTAT(LC103)	*1*	*2*	
FC104.MODE	*1*	*2*	*Normal operating parameters—4-second update* *Parameters are grouped together because they all reside on the same network or data hiway and, in the case of a UCN, all of these parameters are control parameters.*
FC104.OP	*1*	*2*	
FC104.PV	*1*	*2*	
FC104.SP	*1*	*2*	
HS6501.PV	*1*	*2*	
HS6502.PV	*1*	*2*	
LC103.MODE	*1*	*2*	
LC103.OP	*1*	*2*	
LC103.PV	*1*	*2*	
LI103.PV	*1*	*3*	*IOL parameter*
FC104.K	*3*	*4*	*Limits and constants that seldom change—slower update.* *Parameters are grouped together because they all reside on the same network/data hiway and, in the case of a UCN, all of the parameters are control parameters.*
FC104.OPHILM	*3*	*4*	
FC104.OPLOLM	*3*	*4*	
FC104.T1	*3*	*4*	
LC103.K	*3*	*4*	
LI103.PVHITP	*3*	*5*	*IOL parameters*
LI103.PVLOTP	*3*	*5*	
FY105.PVTV	*1*	*6*	*Normal operating parameter—4-second update. Sorted into a separate group because it resides in a different logical node than the other parameters (AM unit 39).*

Lab Exercise

Update Actors

Objective

In this exercise, you will investigate the actors UPDATE and DMD_UPD.

Instructions

1. Call up MOOSE, then select ACTORS.
2. Select UPDATE ACTORS and investigate the examples.

Configure Collection Set

Objective

In this exercise, you will configure the Collection Set of a custom display.

1. Call up display

Read custom display REACT_OV into the Picture Editor:

```
R $Fn>DEMO>REACT_OV
```

2. Printout Collection Set

Get a printout of the Collection Set to use as a worksheet for this exercise: 1.) Select the entire display, 2.) After the display appears flashing white, enter the print command with the symbol (variable) option:

```
PR $Pn SY
```

Notice that the variables in the printout are in alphabetical order.

3. Display Collection Set

Display the Collection Set of REACT_OV: S C

Use the Page Forward key to look at all the variables.

4. Determine configuration

Configure the Collection Rate and Collection Group ID using:

- the general guidelines provided previously in this course module, and
- the specific entity information provided on the next 3 pages.

Use your printout to do the configuration on paper first.

5. Enter configuration

After determining the rates and groups, enter the configuration into the Collection Set display. Use the Optimize Collection Groups command to speed your data entry. After completing the configuration, write the display to your cartridge disk.

Slick trick!

The first time you configure a display's Collection Set, type an erroneous entry into the first port; for example, type 999 into the Collection Rate port. This allows you to press [ENTER] after each subsequent entry (for error checking of that entry) without canceling the Collection Set, However, the cursor always goes back to the first Collection Rate entry.

When you finish the Collection Set, return to the first page, replace the erroneous entry with the correct value, and press [ENTER]. The data is entered, and the main display appears. If you press [ENTER] when there are no errors, the Collection Set clears from the screen and the main display appears.

Continued on next page

Configure Collection Set, Continued

Point descriptions

The following table describes the purpose, point type, and residency of each point in the Collection Set of display REACT_OV.

Point	Point Type	Residency
BRSLGDSP	Custom AM	Unit GC
FVL21841 Ingredient B Discharge Valve	Digital Composite	Process Node 19 UCN 1
FVL22841 Ingredient C Discharge Valve		
DVL23841 Ingredient D Discharge Valve		
AG24841 Reactor 1 Discharge Valve		
FY21841 Ingredient D Feed Pump	Totalizer (Reg PV)	
FY22841 Reactor 1 Feed Meter		
FVL21842 Reactor 1 Ingredient D Header Valve	Digital Composite	
FVL22842 Reactor 1 Header Feed Valve		
DVL23842 Reactor 1 Ingredient A Header Valve	Digital Composite	
AG24842 Reactor 1 Ingredient B Header Valve		
FVL21843 Reactor 1 Header Valve Recycle		
INGA841 Reactor 1 Level	Numeric	
AGI24841 Reactor1 Discharge Valve Input	Digital Input	

Continued on next page

Configure Collection Set, Continued

Point descriptions, continued

Point	Point Type	Residency
FVI21841 Ingredient B Discharge Valve Input	Digital Input	Process Node 19 UCN 1
FVL22843 Reactor 1 TC105 Recirculate	Digital Composite	
DVL23843 Ingredient B, C Line Pump		
AG24843 Ingredient D Discharge Pump		
FVL21844 Reactor 1 Discharge Pump		
FVL22844 Recycle Tank Discharge Pump	Digital Composite	
TI21841 Reactor 1 Pressure	Analog Input	
SIMLT841 Parallel Add of Ingredient D	Proc. Mod.	
REACT841 Reactor Initialization Sequence		
FILL2841 Reactor 1 Sequence		
FILL3841 Reactor Sequence Simulation		
FILL2842 Reactor 2 Simulation		
TIC21841 Reactor 1 Temperature	PID	
TIC21842 Reactor 1 Jacket Temperature		
DVL23844 Rinse Water Feed for B, C	Digital Composite	

Continued on next page

Configure Collection Set, Continued

Point descriptions, continued

Point	Point Type	Residency
FVL21181 Reactor 2 Discharge Valve	Digital Composite	Process Box 24 Data Hiway 2
FI21181 Reactor 2 Feed Meter	Analog Input	
FVL22181 Reactor 2 Ingredient D Header Valve	Digital Composite	
DVL23181 Reactor 2 Header Feed Valve		
AG24181 Reactor 2 Ingredient B Header Valve		
FVL21182 Reactor 2 Ingredient E Header Valve		
FVL22182 Reactor 2 Header Valve Recycle		
INGA181 Reactor 2 Level	Numeric	
DVL23182 Reactor 2 TC205 Recirculate	Digital Composite	
AG24182 Reactor 2 Discharge Pump	Digital Composite	
LI24181 Reactor 2 Pressure	Analog Input	
SIMLT181 Reactor 2 Sequence	Proc. Mod.	
TIC21181 Reactor 2 Temperature	PID	
TIC21182 Reactor 2 Jacket Temperature		

Lab Exercise Solutions

Check your answers

Check your answers against the Collection Set shown on the following pages.

If you have any questions about the solutions, refer to the tables that provide more explanation at the end of the lab solutions.

If you still have questions, ask your course manager for assistance.

NOTE — Save your worksheet printout for the end of module test.

Print your configuration

When you are satisfied with your configuration, create a printout of your completed Collection Set to show your course manager for the end of module test (use the SYmbol option of the Print command).

Continued on next page

Lab Exercise Solutions, Continued

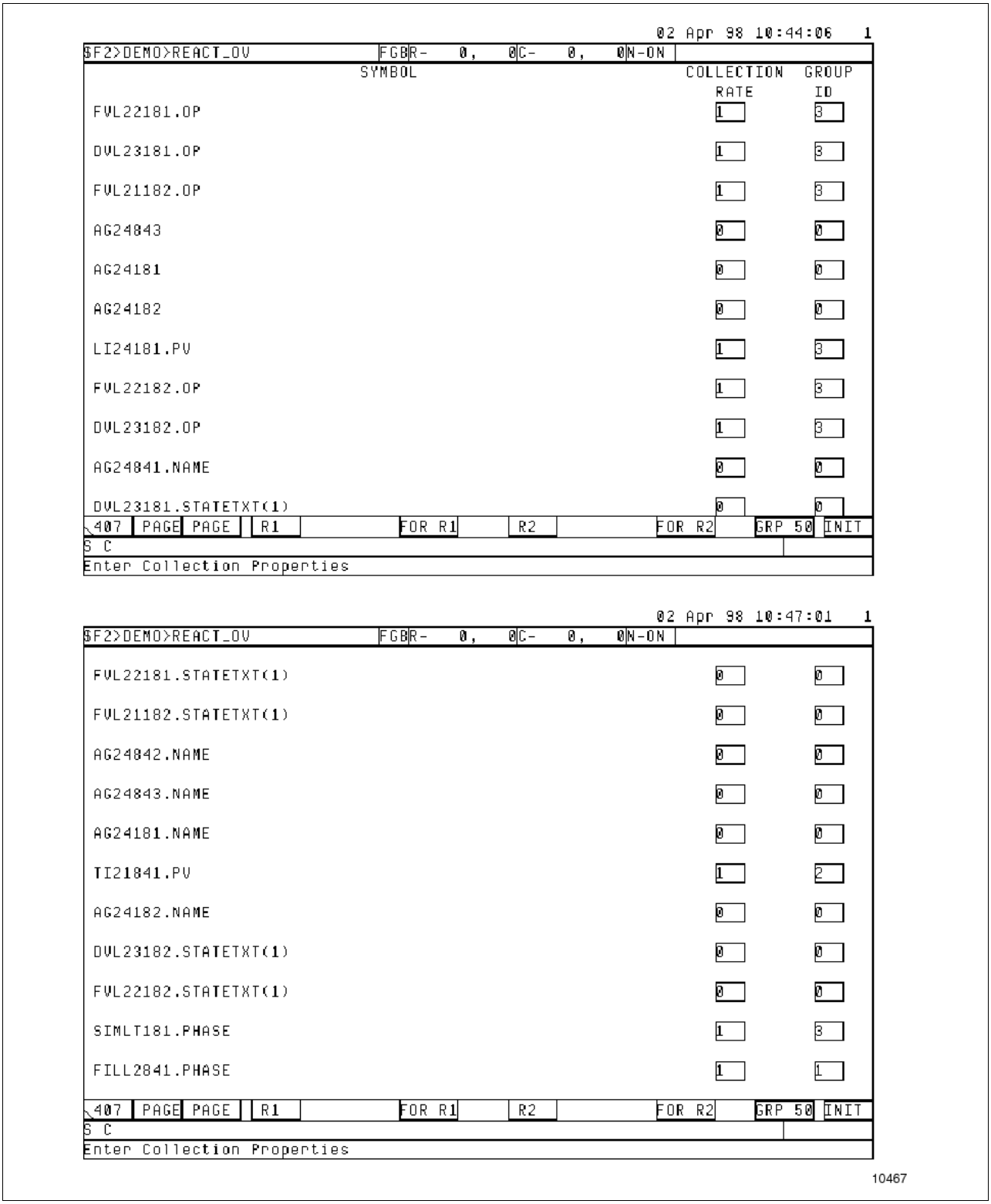


Figure 11 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

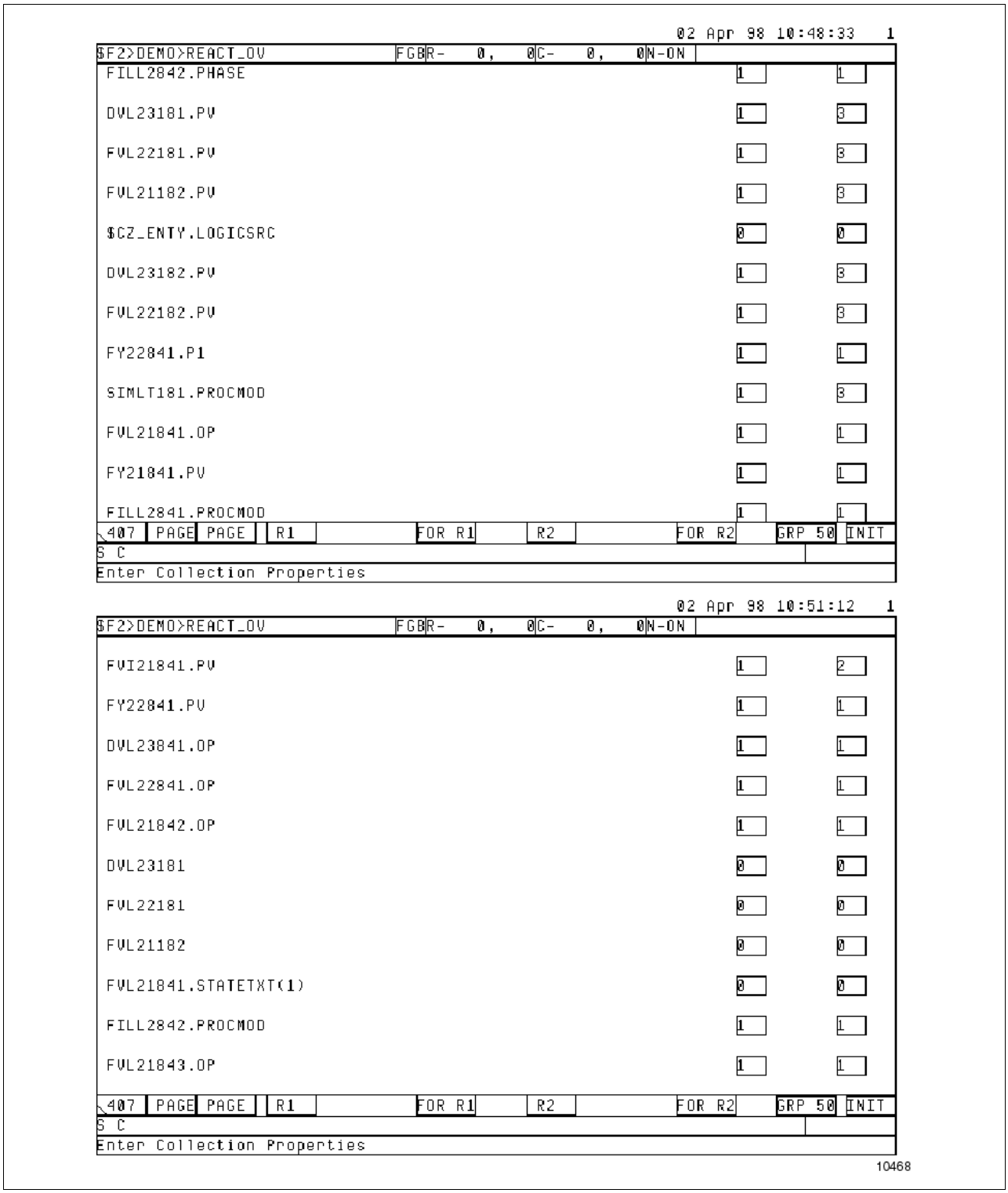


Figure 12 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

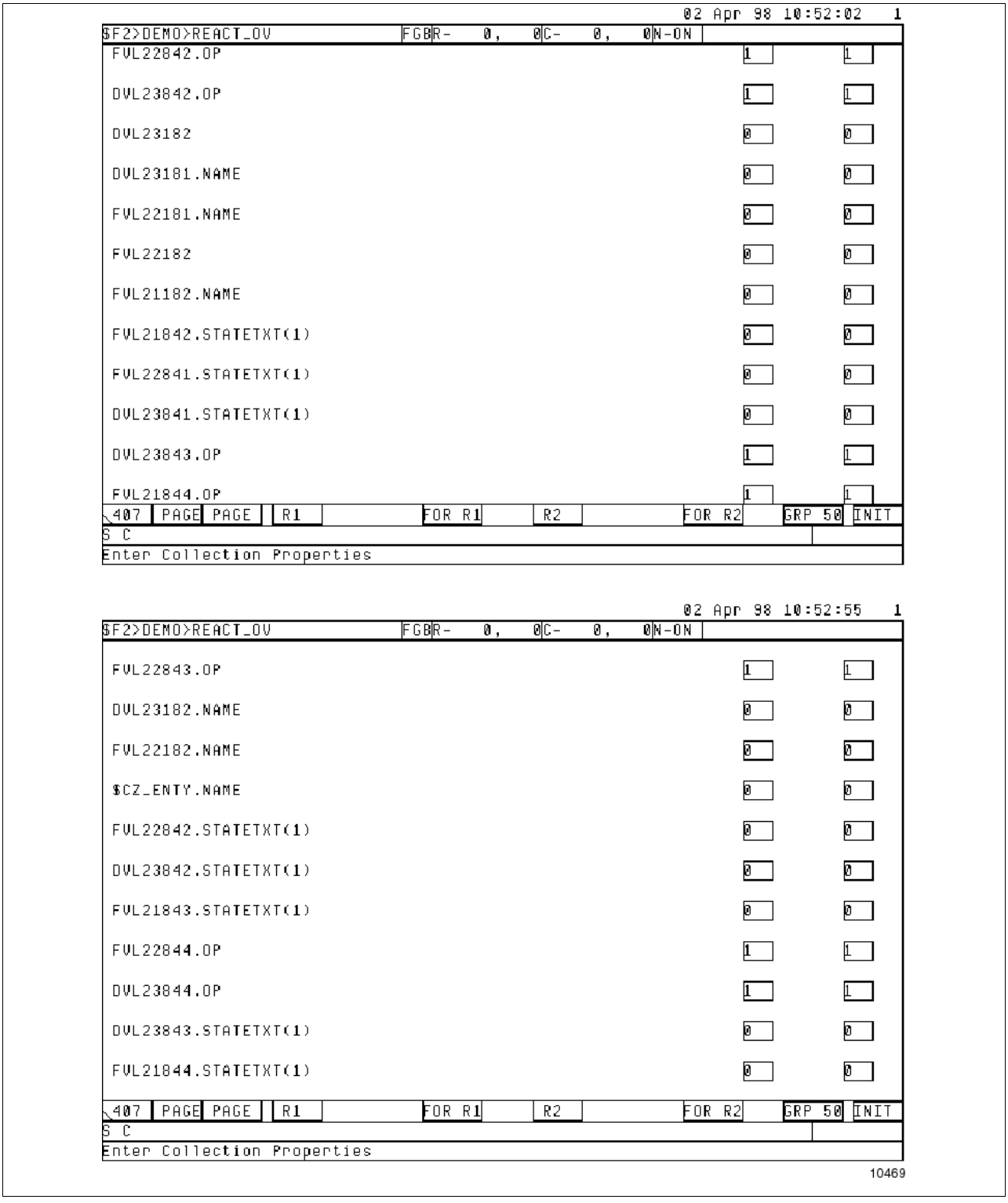


Figure 13 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

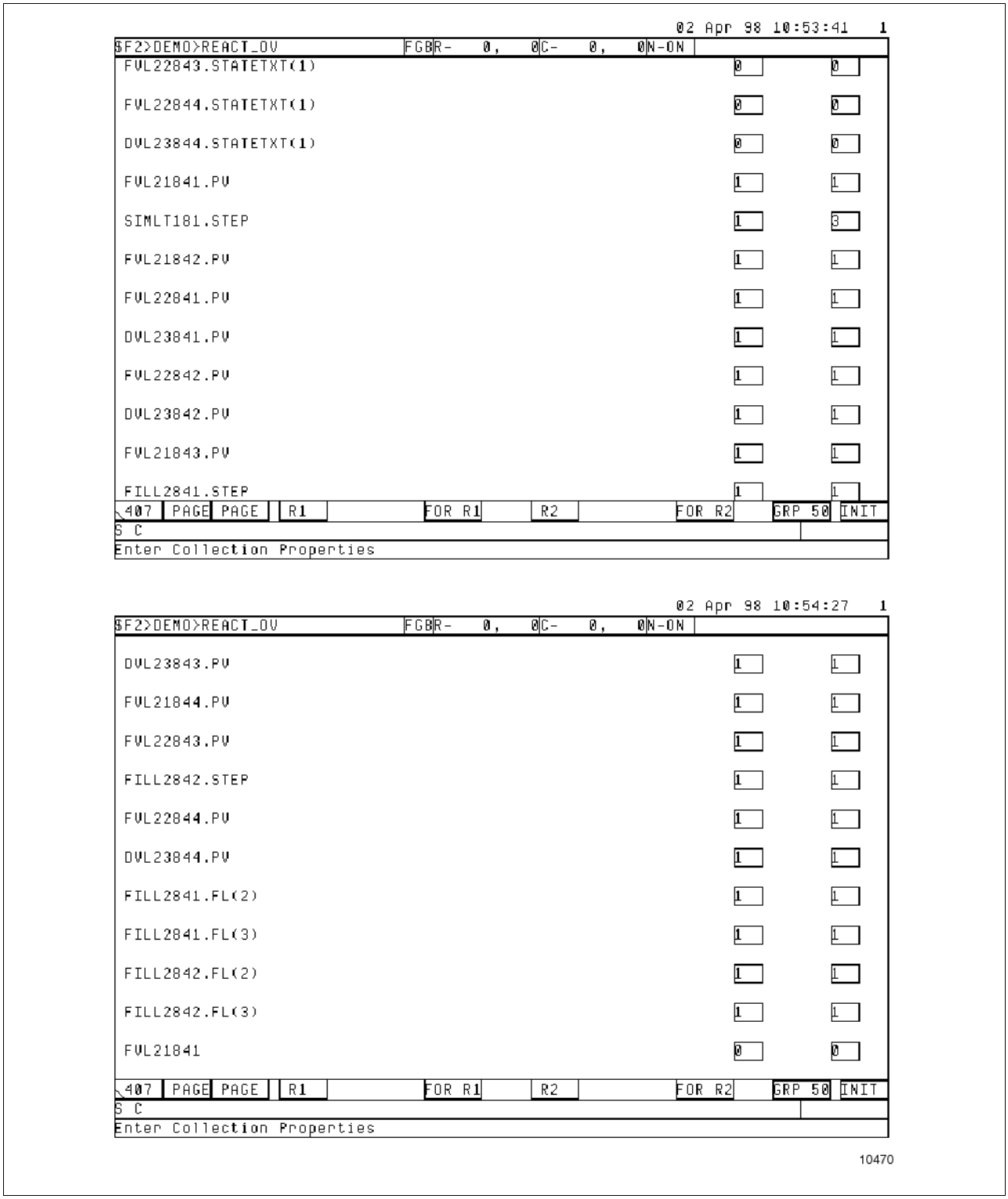


Figure 14 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

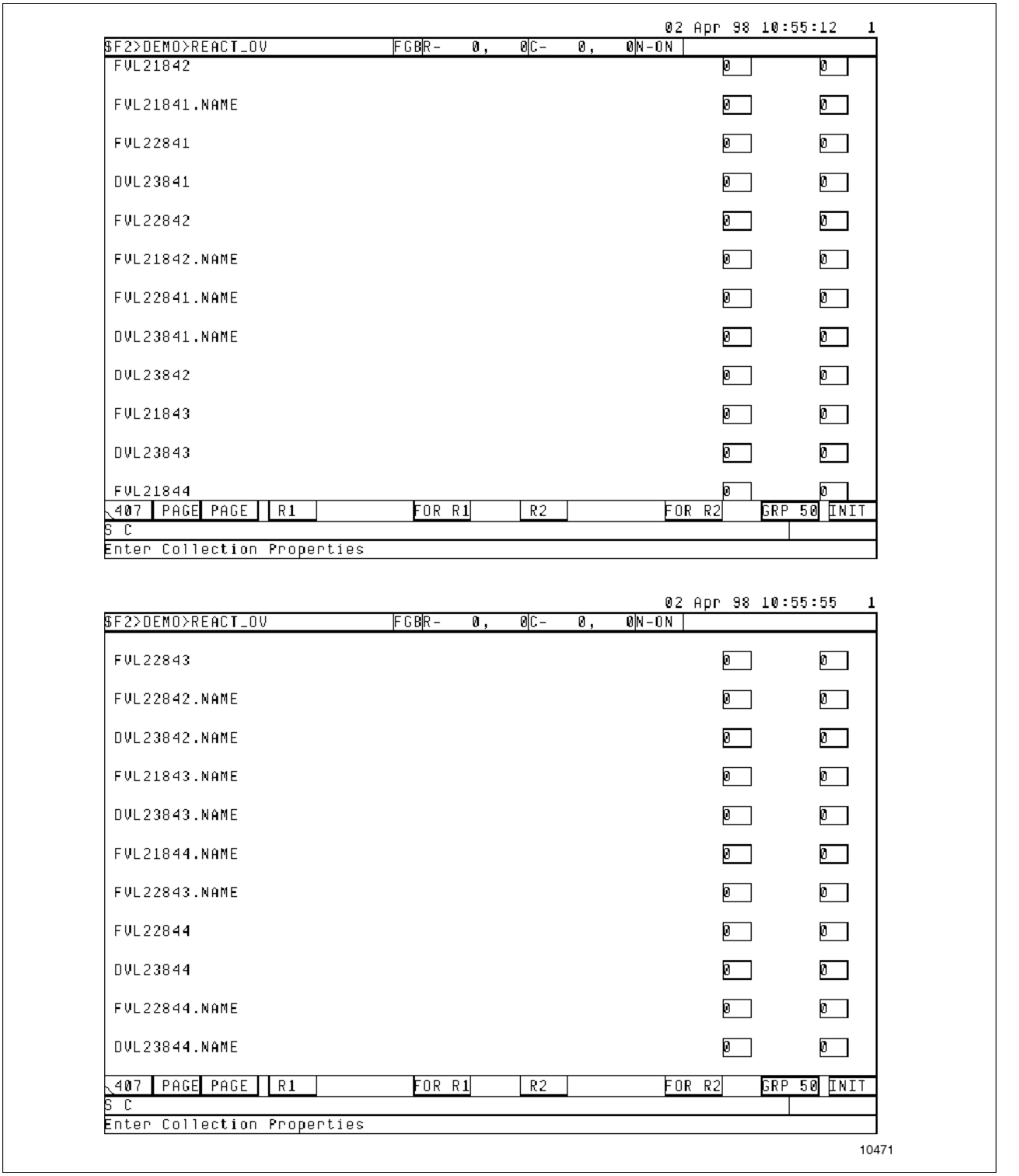


Figure 15 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

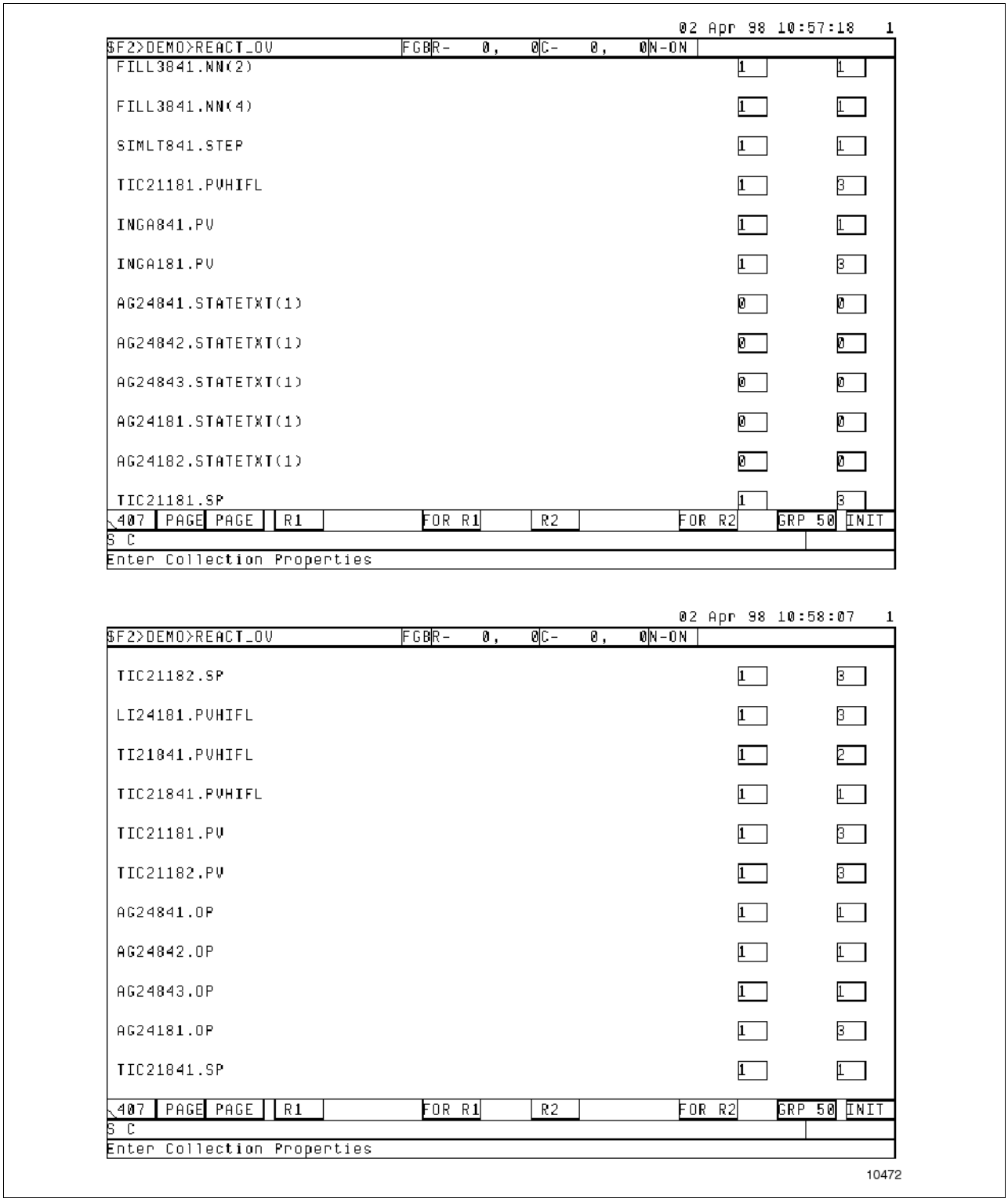


Figure 16 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

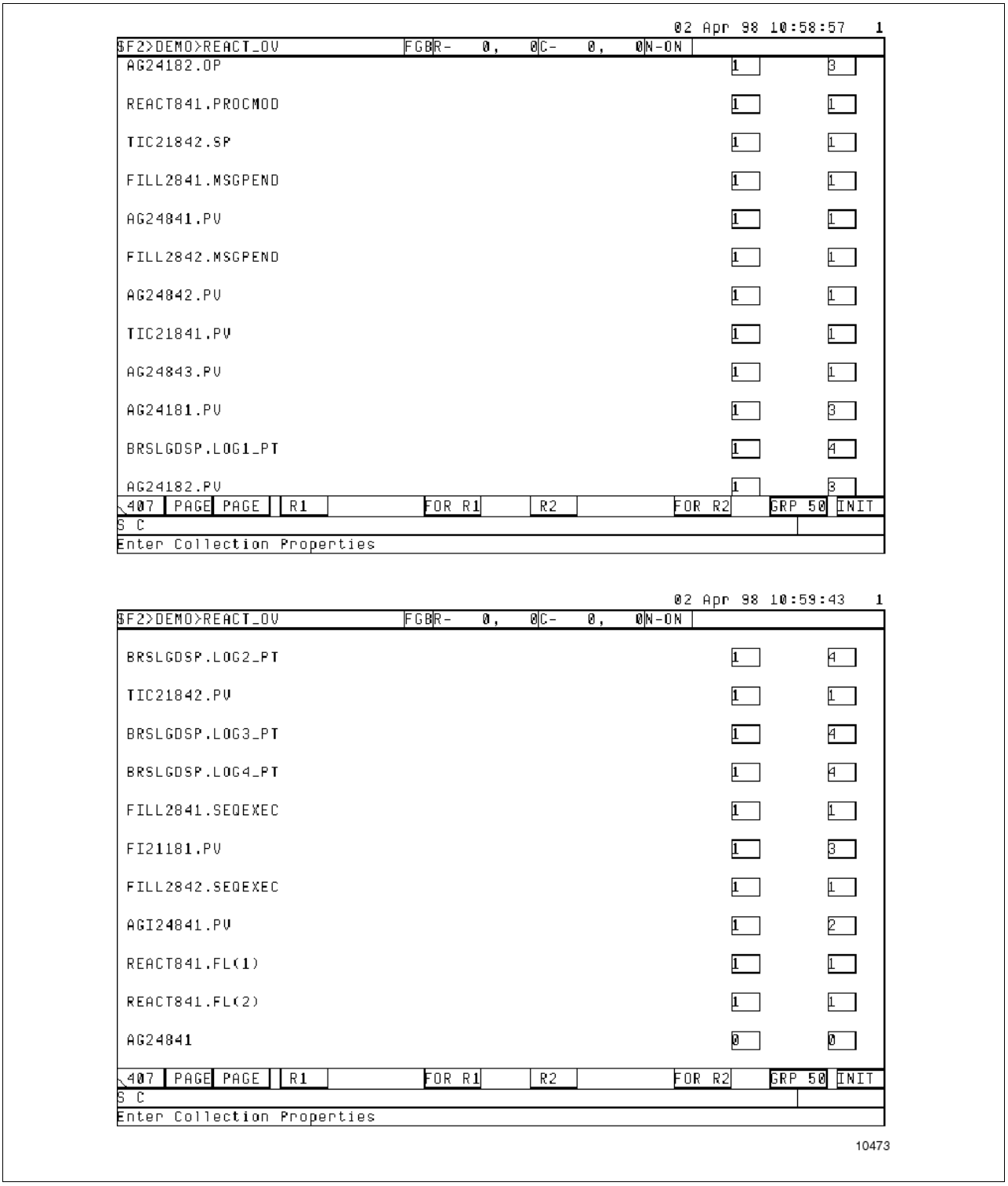


Figure 17 - Lab Exercise Solutions

Continued on next page

Lab Exercise Solutions, Continued

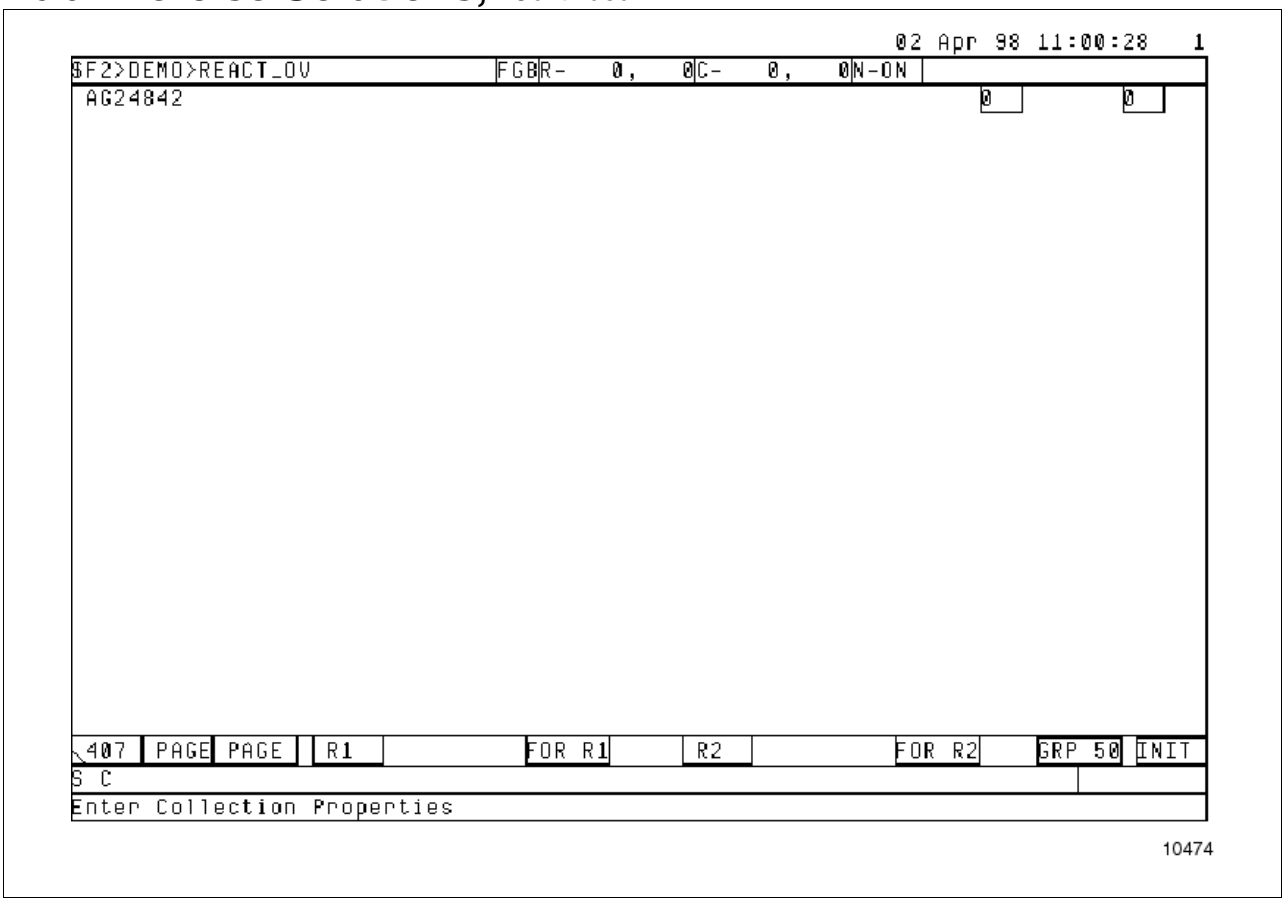


Figure 18 - Lab Exercise Solutions

Lab Exercise Solutions, Continued

More Explanation

The following tables provide more explanation about how the Collection Set should be configured in this lab exercise.

Variable Name	Rate	Group	Notes
$CZ_ENTY.LOGICSRC	*0*	*0*	*Logic input source of entity currently selected in the change zone — update at invocation only.*
$CZ_ENTY.NAME	*0*	*0*	*Entity name of entity currently selected in the change zone — update at invocation only.*
AG24181	*0*	*0*	*Entity name or label — update at invocation only.*
AG24181.NAME	*0*	*0*	*It is not necessarily advantageous to group invocation*
AG24181.STATETXT(1)	*0*	*0*	*only parameters, so leave them the default Group 0.*
AG24182.NAME	*0*	*0*	
AG24182	*0*	*0*	
AG24182.STATETXT(1)	*0*	*0*	
AG24841.NAME	*0*	*0*	
AG24841	*0*	*0*	
AG24841.STATETXT(1)	*0*	*0*	.
AG24842.NAME	*0*	*0*	
AG24842	*0*	*0*	
AG24842.STATETXT(1)	*0*	*0*	
AG24843	*0*	*0*	
AG24843.NAME	*0*	*0*	
AG24843.STATETXT(1)	*0*	*0*	
DVL23181	*0*	*0*	
DVL23181.NAME	*0*	*0*	
DVL23181.STATETXT(1)	*0*	*0*	

Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
DVL23182	*0*	*0*	
DVL23182.NAME	*0*	*0*	
DVL23182.STATETXT(1)	*0*	*0*	
DVL23841	*0*	*0*	
DVL23841.NAME	*0*	*0*	
DVL23841.STATETXT(1)	*0*	*0*	
DVL23842	*0*	*0*	
DVL23842.NAME	*0*	*0*	
DVL23842.STATETXT(1)	*0*	*0*	
DVL23843.NAME	*0*	*0*	
DVL23843.STATETXT(1)	*0*	*0*	
DVL23843	*0*	*0*	
DVL23844.NAME	*0*	*0*	
DVL23844.STATETXT(1)	*0*	*0*	
DVL23844	*0*	*0*	
FVL21182.NAME	*0*	*0*	
FVL21182.STATETXT(1)	*0*	*0*	
FVL21182	*0*	*0*	
FVL21841.NAME	*0*	*0*	
FVL21841.STATETXT(1)	*0*	*0*	
FVL21841	*0*	*0*	
FVL21842.NAME	*0*	*0*	
FVL21842.STATETXT(1)	*0*	*0*	
FVL21842	*0*	*0*	
FVL21843	*0*	*0*	
FVL21843.NAME	*0*	*0*	
FVL21843.STATETXT(1)	*0*	*0*	

Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
FVL21844.NAME	*0*	*0*	*Entity name or label—update at invocation only.*
FVL21844	*0*	*0*	*It is not necessarily advantageous to group invocation-*
FVL21844.STATETXT(1)	*0*	*0*	*only parameters, so leave them at the default Group 0.*
FVL22181	*0*	*0*	
FVL22181.NAME	*0*	*0*	
FVL22181.STATETXT(1)	*0*	*0*	
FVL22182.NAME	*0*	*0*	
FVL22182	*0*	*0*	
FVL22182.STATETXT(1)	*0*	*0*	
FVL22841	*0*	*0*	
FVL22841.NAME	*0*	*0*	
FVL22841.STATETXT(1)	*0*	*0*	
FVL22842	*0*	*0*	
FVL22842.NAME	*0*	*0*	
FVL22842.STATETXT(1)	*0*	*0*	
FVL22843	*0*	*0*	
FVL22843.NAME	*0*	*0*	
FVL22843.STATETXT(1)	*0*	*0*	
FVL22844.NAME	*0*	*0*	
FVL22844.STATETXT(1)	*0*	*0*	
FVL22844	*0*	*0*	

Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
AG24841.OP	*1*	*1*	*Normal operating parameters—4-second update.*
AG24841.PV	*1*	*1*	*Parameters are grouped together because they all*
AG24842.OP	*1*	*1*	*reside on the same UCN and*
AG24842.PV	*1*	*1*	*they are all control parameters (see Table 8).*
AG24843.OP	*1*	*1*	
AG24843.PV	*1*	*1*	
DVL23841.OP	*1*	*1*	
DVL23841.PV	*1*	*1*	
DVL23842.PV	*1*	*1*	
DVL23842.OP	*1*	*1*	
DVL23843.PV	*1*	*1*	
DVL23843.OP	*1*	*1*	
DVL23844.PV	*1*	*1*	
DVL23844.OP	*1*	*1*	
FILL2841.FL(2)	*1*	*1*	
FILL2841.FL(3)	*1*	*1*	
FILL2841.MSGPEND	*1*	*1*	
FILL2841.PHASE	*1*	*1*	
FILL2841.PROCMOD	*1*	*1*	
FILL2841.SEQEXEC	*1*	*1*	
FILL2841.STEP	*1*	*1*	
FILL2842..FL(2)	*1*	*1*	
FILL2842.FL(3)	*1*	*1*	
FILL2842.MSGPEND	*1*	*1*	
FILL2842.PHASE	*1*	*1*	
FILL2842.PROCMOD	*1*	*1*	
FILL2842.SEQEXEC	*1*	*1*	
FILL2842.STEP	*1*	*1*	

Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
FILL3841.NN(2)	*1*	*1*	
FILL3841.NN(4)	*1*	*1*	
FVL21841.OP	*1*	*1*	
FVL21841.PV	*1*	*1*	
FVL21842.OP	*1*	*1*	
FVL21842.PV	*1*	*1*	
FVL21843.PV	*1*	*1*	
FVL21843.OP	*1*	*1*	
FVL21844.OP	*1*	*1*	
FVL21844.PV	*1*	*1*	
FVL22841.OP	*1*	*1*	
FVL22841.PV	*1*	*1*	
FVL22842.OP	*1*	*1*	
FVL22842.PV	*1*	*1*	
FVL22843.OP	*1*	*1*	
FVL22843.PV	*1*	*1*	
FVL22844.OP	*1*	*1*	
FVL22844.PV	*1*	*1*	
FY21841.PV	*1*	*1*	
FY22841.P1	*1*	*1*	
FY22841.PV	*1*	*1*	
INGA841.PV	*1*	*1*	
REACT841.FL(1)	*1*	*1*	
REACT841.FL(2)	*1*	*1*	
REACT841.PROCMOD	*1*	*1*	
SIMLT841.STEP	*1*	*1*	
TIC21841.PVHIFL	*1*	*1*	
TIC21841.PV	*1*	*1*	

Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
TIC21841.SP	*1*	*1*	
TIC21842.PV	*1*	*1*	
TIC21842.SP	*1*	*1*	
AGI24841.PV	*1*	*2*	*Normal operating parameters—4-second update Parameters are separated into a new group because they reside on the same UCN but they are all I/O Link parameters (see Table 8).*
FVI21841	*1*	*2*	
TI21841.PV	*1*	*2*	
TI21841.PVHIFL	*1*	*2*	
AG24181.OP	*1*	*3*	*Normal operating parameters - 4 second update. Parameters are separated into a new group because they reside in a Data Hiway. Sorting by control and I/O point type is not necessary for Data Hiway variables.*
AG24181.PV	*1*	*3*	
AG24182.OP	*1*	*3*	
AG24182.PV	*1*	*3*	
DVL23181.OP	*1*	*3*	
DVL23181.PV	*1*	*3*	
DVL23182.OP	*1*	*3*	
DVL23182.PV	*1*	*3*	
FI21181.PV	*1*	*3*	
FVL21182.OP	*1*	*3*	
FVL21182.PV	*1*	*3*	
FVL22181.OP	*1*	*3*	
FVL22181.PV	*1*	*3*	
FVL22182.OP	*1*	*3*	
FVL22182.PV	*1*	*3*	
INGA181.PV	*1*	*3*	
LI24181.PV	*1*	*3*	
LI24181.PVHIFL	*1*	*3*	
SIMLTI81.PHASE	*1*	*3*	
SIMLT181.PROCMOD	*1*	*3*	
SIMLT181.STEP	*1*	*3*	
TIC21181.PVHIFL	*1*	*3*	

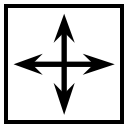
Continued on next page

Lab Exercise Solutions, Continued

More Explanation, continued

Variable Name	Rate	Group	Notes
TIC21181.PV	*1*	*3*	
TIC21181.SP	*1*	*3*	
TIC21182.PV	*1*	*3*	
TIC21182.SP	*1*	*3*	
BRSLGDSP.LOG1_PT	*1*	*4*	*These variables are in a separate group because they are from a different logical node—AM unit GC.*
BRSLGDSP.LOG2_PT	*1*	*4*	
BRSLGDSP.LOG3_PT	*1*	*4*	
BRSLGDSP.LOG4_PT	*1*	*4*	

Directions



DIRECTIONS—This is the end of the study material for this module. Discuss questions concerning the study material or the lab activities with a colleague or a course manager.

If you are satisfied that you have achieved the objectives of this module, continue with the next section, the Proficiency Evaluation.

Proficiency Evaluation

Criterion Test

Instructions

1. Describe to your course manager the purpose of the Update actor in the target shown in Figure 19.

2. In the lab exercise, you configured the Collection Set of display REACT_OV, according to guidelines in this course module. You compared your results to the lab exercise solutions.

   Successful completion of the lab exercise satisfies the test requirement.

   Do the following to demonstrate that you successfully completed the lab exercise:

   a. In the space provide below, list the rules of Collection Set configuration that you used to configure the Collection Set of display REACT_OV. Be prepared to discuss these rules with your course manager.

   b. Show your course manager the worksheet printout and the final printout of the completed Collection Set for display REACT_OV.

   c. Explain to your course manager any discrepancies between your Collection Set configuration and the solutions provided in the lab exercise.

Continued on next page

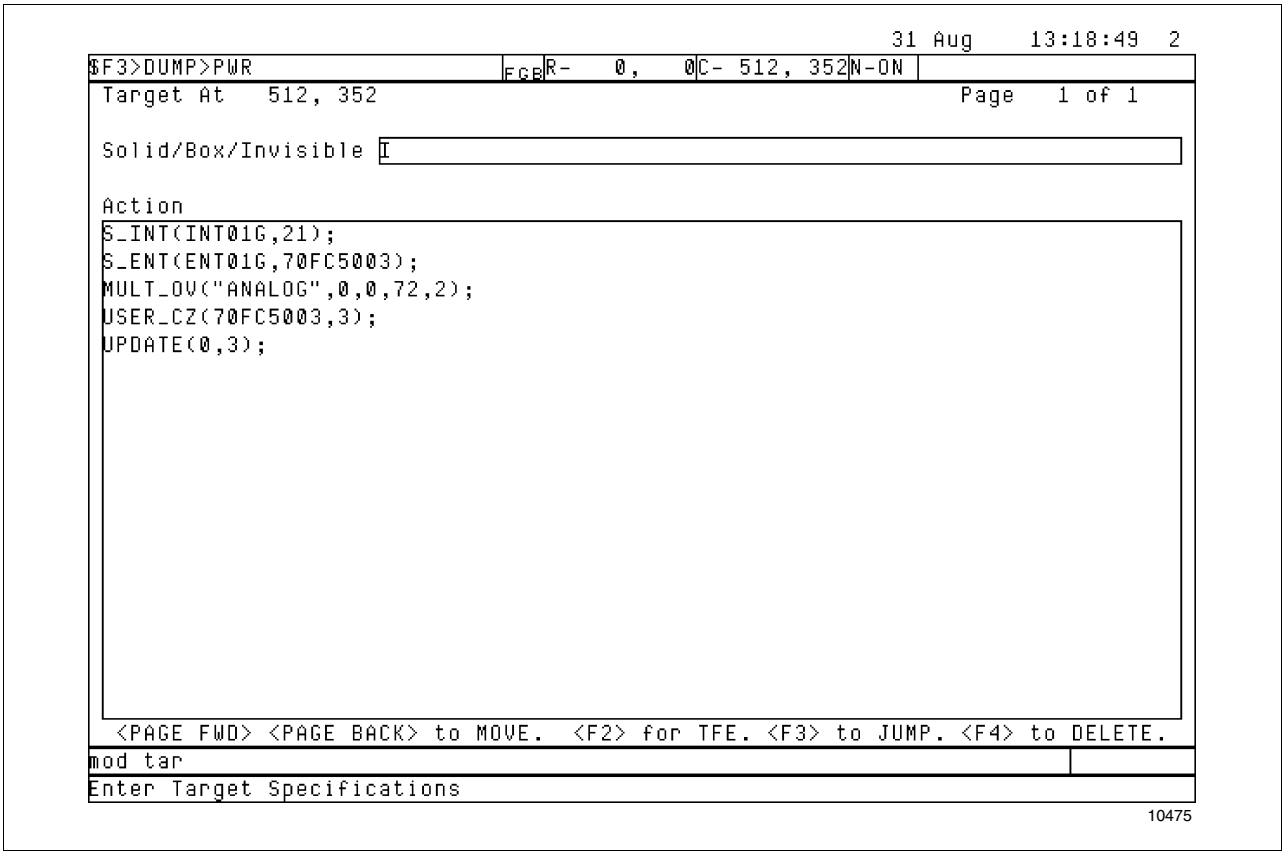


Figure 19 - Test - Update Actor

Self-Evaluation

Answers

1. *The purpose of the Update actor in Figure 19 is to update the overlay with the capability that causes it to respond to the mode, SP, OP, and ramp keys on the keyboard.*

2.

   a. *RULE: Do not schedule parameters that do not change.*

      RULE: Provide a separate group for each Collection Rate.

      RULE: Provide a separate group for each logical node (network or AM/CG unit).

      RULE: Provide a separate group for control and I/O processor types on a UCN:

      Group of invocation only variables

      Group of scheduled control processor variables for network 1

      Group of scheduled I/O processor variables for network 1

      Group of scheduled variables for network 2

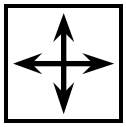
      Group of scheduled variables for the AM unit

   b. *Your Collection Set printout should appear similar to the Lab Exercise Solutions.*

   c. *You may have used any group numbers between 0 - 245.*

      You may have configured certain parameters to be in the FST group.

Directions



Directions—This is the end of this module.

Use your course map to

- Get your course manager to sign off this module.
- Choose your next eligible module.

If you have a question

- Ask your course manager.

Last Page