RMV9000 Process Control System

RMV9000 to DeltaV Serial Interface User Guide

Mynah Part Number IOD-1193
August, 2010
Document Version 3.2
(For Series 2 Serial Card)

MYNAH℠

RMV9000 Process Control System

RMV9000 to DeltaV Serial Interface User Guide

Mynah Part number IOD-1193
Previous Emerson Part Number 09210-1209-0002
August, 2010
Document Version 3.2
(For Series 2 Serial Card)

MYNAH℠

August, 2010

The information in this document is subject to change without notice and should not be construed as a commitment by Mynah Technologies. Mynah Technologies assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Emerson Process Management.

_____

_____

The following are trademarks of Emerson Process Management or Fisher-Rosemount Systems Inc.

RMV9000	MVCU	System3	Diogenes	PeerWay
HART	Smart	DeltaV		

Table of Contents

Section 1 – Introduction 1

Media Kit Contents 2
DeltaV Hardware Required (V3 RMV Interface) 3
DeltaV Licenses Required 3

Section 2 – Software Installation 4

Introduction 4
Controller and Serial Card Configuration 4
Installation Instructions (new systems) 5
Upgrade Instructions (to V3.2) 6
Installing and Downloading the MVCU3 Protocol Image 8
Default Serial Port and Dataset Installation 15
Installing DeltaV Templates and Format Specification Files 16
MVCU Faceplates and Detail Displays 17
Installing DeltaV Security Parameters 17
Installing Sample Modules 18

Section 3 – Hardware Setup 19

Introduction 19
MVCU Serial Ports 20
DeltaV Serial Ports 22
MVCU to DeltaV Cable Configurations 23
  Simplex Cards (Type 1 or Type 2) 23
Series 2 Serial Card Diagnostics 27
Series 2 Serial Card Switchover Logic 27

Section 4 - Adding/Editing Serial Cards, Devices, and Data Sets 28

Adding Additional Serial Cards 28
Adding Additional Serial Devices 31
Adding Additional Data Sets 32
Creating New Loop Tagnames 35

Appendix A - Data Set Configuration and Register Assignments 38

Data Set Configuration Options 39

Addressing Modes 41

Normal Contiguous Addressing 41
Sparse Space Addressing 41

Scanning Modes 42

Normal Continuous Scanning 42
On Demand Scanning 42
Block Transfer Scanning 42

Virtual Data Sets **42**

Data Set attributes by Data Type **43**

Appendix B - Module Data Availability **60**

Module Data Availability **61**

Analog Input Block 61
Analog Output Block 61
Analog Node Block 62
Digital Input Block 62
Digital Output Block 62
Digital Node Block 63
PID Function 63
RATIO Function 64
AUTO/MANUAL STATION Function 64
MANUAL LOADER Function 65
TOTALIZER Function 65
TIMER 66
COUNTER 66
ACTUATOR 67
Floating Point Node Block 67

Appendix C – Dataset Planning Forms **69**

Appendix D - Host Computer Access **102**

Introduction 103
Design Overview 109
Module Architectural Overview 111
Host Access Properties 114
Host Read/Write Parameters and Variables 115

Section 1 – Introduction

This document is intended to provide basic information related to installing the RMV9000 Interface software components and utilities, setting up the hardware interface, and offers DeltaV configuration notes.

This manual describes the functionality and use associated with the DeltaV “Series 2” Programmable Serial Card (PSC). This card can only be used in a Simplex configuration.

This RMV Interface cannot be downloaded into the “Series 1” PSC. Conversely, the V1 driver cannot be used with the “Series 2” PSC.

The RMV9000 to DeltaV serial interface utilizes the DeltaV’s programmable serial card. An MVCU3 protocol image is downloaded into the serial card and allows the DeltaV controller to interrogate the MVCU3. The interface is bi-directional and allows the DeltaV to both read data out of the MVCU3 as well as write data to the MVCU3 controller. The following MVCU3 data types are supported:

- Analog Inputs - Analog Outputs - Analog Nodes
- Digital Inputs - Digital Outputs - Digital Nodes
- Analog Functions - Digital Functions -Dual Digital Inputs
- Floating Point Nodes - Analog Algorithm Floating Point Data

The interface will allow operators to change setpoints, outputs, modes, digital states, and tuning parameters. The interface also provides access to all status flags such as Auto/Manual modes and alarm flags.

Media Kit Contents

The RMV9000 to DeltaV media kit includes the following components:

1. Distribution folder containing:
    - MVCU3 Serial Protocol Image
    - DeltaV Control Module “Templates”
    - iFix Control Faceplate Displays
    - iFix Detail Displays
    - Default Serial Card configuration with default Data Sets
    - Sample Modules
    - Data Set Planning worksheet (Excel)

2. Serial Interface User Guide

DeltaV Hardware Required (V3 RMV Interface)

Based upon the limitations of the DeltaV programmable serial card, one serial card is usually required for each MVCU3 controller. The DeltaV serial card offers 32 datasets and research has shown that 25 to 30 datasets are requires to support all data to be exchanged between the MVCU3 and the DeltaV controller.

The following DeltaV hardware is required:

1. DeltaV M5+, MD, MD Plus or MX controller.

2. DeltaV Series 2 Programmable Serial Card (1 per MVCU3 Controller)

3. DeltaV Carriers, Power Supply, Ethernet Switch, and Pro+ Workstation

DeltaV Licenses Required

The RMV9000 to DeltaV interface requires DeltaV software release V7.1 or better. The following DeltaV software licenses are required:

1. Pro+ Workstation Licenses sufficient to support the number of serial interfaces. One (1) DST is required for each Loop Tagname configured. Estimate 200 - 250 Loop Tagnames per MVCU3 controller.

2. DeltaV system Control licenses. (Estimate 200 – 250 DSTs per MVCU3 interface) The Control license is required to provide access to the Scaling control block to convert unscaled analog integers into engineering units. Monitored licenses do not provide sufficient features to support the MVCU3 interface.

3. Serial Card (ModBus) License. For DeltaV versions prior to v9.3, one serial license is required for each serial port. Each serial license provides access to 16 DataSets. Two serial port licenses are required to access all 32 serial card data sets. For DeltaV v9.3 or later, a port license is not required.

Section 2 – Software Installation

Introduction

The MVCU3 interface Control Module “templates”, and standard Data Sets provided on the distribution media are designed with a Programmable Serial Card located in Slot 1 on the DeltaV carrier. Additionally, the sample Controller and Serial Card configurations are programmed with the MVCU address as 1. While the Serial Card may be located in any slot on the DeltaV carrier, the default configurations will have to be reconfigured should the Serial Card be installed in a slot other than slot 1.

The DeltaV Pro+ and DeltaV controller require the proper licenses loaded prior to installing the MVCU3 interface software. It is also important that the DeltaV Registration Utility be executed so that subsequent software updates are available.

Check the Controller licensing properties prior to continuing the installation process.

Controller and Serial Card Configuration

Prior to downloading the MVCU3 protocol image, the controller and serial card need to be configured so the PRO+ Workstation will be able to download the MVCU3 protocol image. Using the DeltaV Explorer commission the controller with the name **CTLR1**.

The distribution media contains a default serial card configuration. This Serial Card configuration also contains default Dataset configurations.

The default serial port configuration initializes the serial ports as follows:

Simplex PSC
- 19,200 Kbaud
- 8 data bits
- 2 stop bits
- No Parity
- RS232

Default data sets are included in the importable serial card configuration. Refer to the “Data Set Configuration and Register Assignments” (Appendix A) of this user guide for details associated with the default data set configurations.

Installation Instructions (new systems)

When installing the RMV to DeltaV interface software on a new system; follow the following instructions.

Step 1: Make Folders

- Create the folder - **C:\DeltaV\ctlr\ProgSerial\RMV9000**
  Note: Creates a folder for the RMV Serial Driver.

- Create the folder - **C:\DeltaV\DVData\Import-Export\Template\RMV9000**
  Note: Creates a folder for the RMV control module templates.

- Create the folder - **C:\DeltaV\DVData\Import-Export\IO_RMV9000**
  Note: Creates a folder for the default Datasets.

- Create the folder - **C:\DeltaV\DVData\BulkEdit\RMV9000**
  Note: Creates a folder for the RMV Format Specification files.

Step 2: Copy Files from the distribution folder to the C: drive

- Copy files - \ **MVCU_Protocol *.*** to **C:\DeltaV\ctlr\ProgSerial\RMV9000**
  Note: Copies the RMV serial driver to the target folder.

- Copy files - **Copy \MVCU_Templates *.*** to **C:\DeltaV\DVData\Import-Export\Template\RMV9000**
  Note: Copies the RMV module templates to the target folder.

- Copy files - **Copy \IO_RMV9000 *.*** to **C:\DeltaV\DVData\Import-Export\IO_RMV9000**
  Note: Copies the default DataSets to the target folder.

- Copy files - **Copy \Format_Files *.*** to **C:\DeltaV\DVData\BulkEdit\RMV9000**
  Note: Copies the Format Specification Files to the target folder.

- Copy files - **Copy \MVCU_Faceplate_Displays *.*** to **C:\DeltaV\DVData\Graphics-iFix\Pic\Faceplate**
  Note: Copies the RMV Faceplates to the target folder.

- Copy files - **Copy \MVCU_Detail_Displays *.*** to **C:\DeltaV\DVData\Graphics-iFix\Pic\Detail**
  Note: Copies the RMV Detail Displays to the target folder:

  After the software installation; you may proceed to download the RMV driver into the PSC, install and import default communication card configurations, and import default data set configurations. These steps are described in subsequent sections of this User Guide.

Upgrade Instructions (to V3.2)

When upgrading the RMV to DeltaV interface software to V3.2, the following pre-installation steps are required.

This "Upgrade" option will delete all older Templates, Faceplate and Detail Displays, the RMV driver, and Format Specification files.

If you have customized the "standard" RMV Interface Faceplates, Detail Displays, module templates, or Format Specification Files; archive them before running the "Upgrade" installation procedure. These modified files may then be converted using the standard DeltaV migration utilities.

Note the following:

- The V3.2 RMV Interface module templates have been converted to be DeltaV V11.3 compatible.
- The V3.2 RMV Interface Faceplates and Detail Displays have been converted to use the "DVSYS" browse path.

Step 1: Delete Old RMV Interface Files

- Delete the files - **C:\DeltaV\ctlr\ProgSerial\RMV9000 *.***
  Note: Deletes the old RMV Driver

- Delete the files - **C:\DeltaV\DVData\Import-Export\Template\RMV9000 *.***
  Note: Deletes the old RMV module templates

- Delete the files - **C:\DeltaV\DVData\Import-Export\IO_RMV9000 *.***
  Note: Deletes the old default DataSets

- Delete the files - **C:\DeltaV\DVData\BulkEdit\RMV9000 *.***
  Note: Deletes the old RMV Format Specification Files

- Delete the files - **C:\DeltaV\DVData\Graphics-iFix\Pic\Faceplate MVCU_*.***
  Note: Deletes the old RMV Faceplates

- Delete the files - **C:\DeltaV\DVData\Graphics-iFix\Pic\Detail MVCU_*.***
  Note: Deletes the old RMV Detail Displays

- Delete the files - **C:\DeltaV\DVData\Import-Export\RMV9000_Modules *.***
  Note: Deletes the old RMV module templates.
  This folder was used in a previous release and may not be still exist.

- Remove the folder - **C:\DeltaV\DVData\Import-Export\RMV9000_Modules**
  Note: This will remove the old RMV module template folder.
  This folder was used in a previous release and may not be still exist.

Step 2: Make Folders

- Create the folder - **C:\DeltaV\DVData\Import-Export\Template\RMV9000**
  Note: Creates a folder for the RMV control module templates.

Step 3: Copy Files from the distribution folder to the C: drive

- Copy files - **\MVCU_Protocol *.*** to **C:\DeltaV\ctlr\ProgSerial\RMV9000**
  Note: Copies the RMV serial driver to the target folder.

- Copy files - **Copy \MVCU_Templates *.*** to **C:\DeltaV\DVData\Import-Export\Template\RMV9000**
  Note: Copies the RMV module templates to the target folder.

- Copy files - **Copy \IO_RMV9000 *.*** to **C:\DeltaV\DVData\Import-Export\IO_RMV9000**
  Note: Copies the default DataSets to the target folder.

- Copy files - **Copy \Format_Files *.*** to **C:\DeltaV\DVData\BulkEdit\RMV9000**
  Note: Copies the Format Specification Files to the target folder.

- Copy files - **Copy \MVCU_Faceplate_Displays *.*** to **C:\DeltaV\DVData\Graphics-iFix\Pic\Faceplate**
  Note: Copies the RMV Faceplates to the target folder.

- Copy files - **Copy \MVCU_Detail_Displays *.*** to **C:\DeltaV\DVData\Graphics-iFix\Pic\Detail**
  Note: Copies the RMV Detail Displays to the target folder:

After the software installation; you may proceed to download the RMV driver into the PSC, install and import default communication card configurations, and import default data set configurations. These steps are described in subsequent sections of this User Guide

Installing and Downloading the MVCU3 Protocol Image

Step 1 - Copy Protocol Image

The MVCU3 Protocol Image and download utilities are located in the **MVCU_Protocol** directory on the distribution media. Create a folder named **C:\DeltaV\ctl\ProgSerial\RMV9000.** Copy the contents of the **MVCU_Protocol** directory from the distribution folder into the **C:\DeltaV\ctl\ProgSerial\RMV9000** folder on the Pro+ Workstation.

The **C:\DeltaV\ctl\ProgSerial\RMV9000** folder will include the following files:

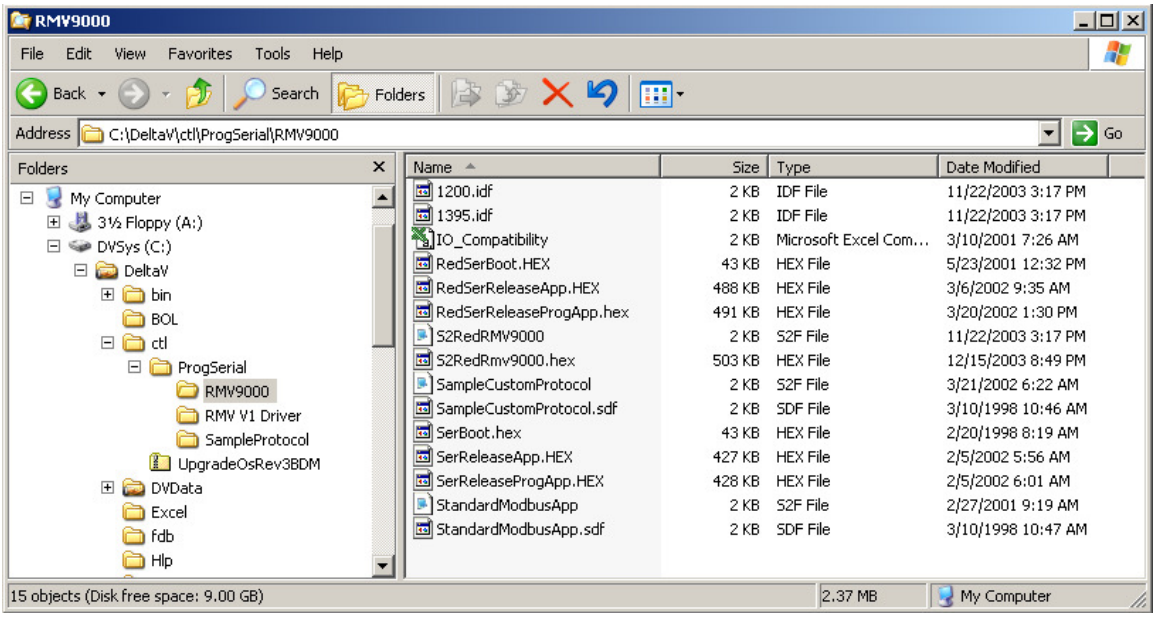


Figure 1 - Contents of C:\DeltaV\ctl\ProgSerial\RMV9000 folder

Step 2 - Download Protocol Image

1. Launch the DeltaV Controller Upgrade Utility by selecting **Start** | **DeltaV** | **Installation** | **Controller Upgrade Utility.**

2. Select the “Upgrade I/O Modules” option and click on NEXT.

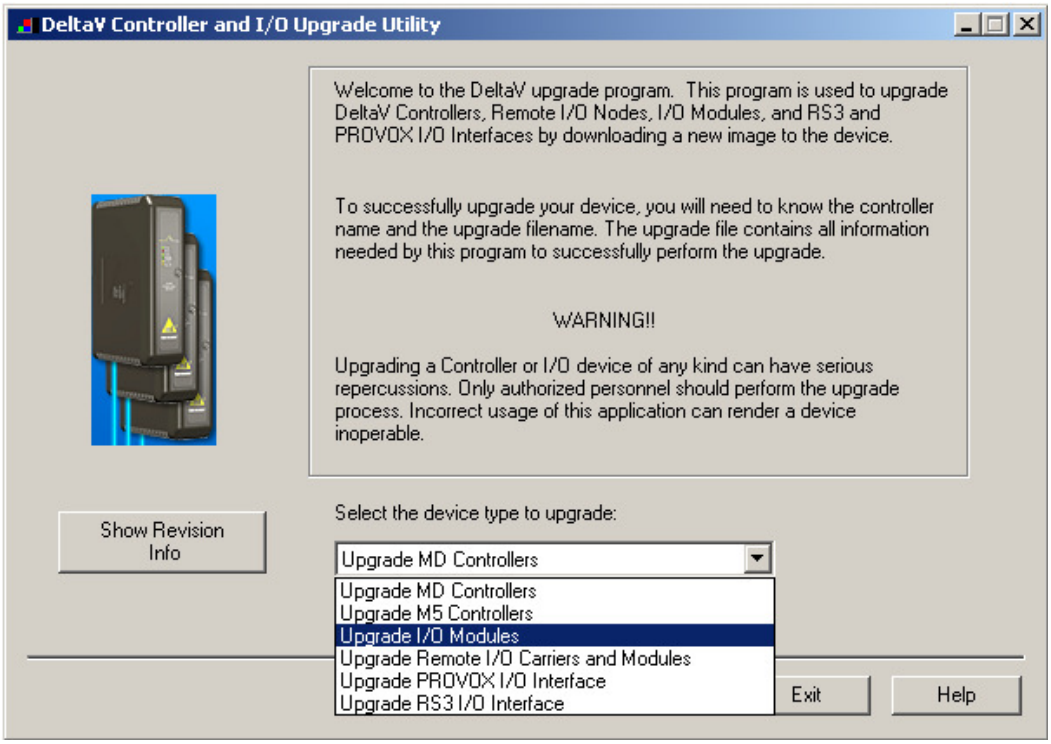


Figure 2 - Controller Upgrade Utility

3. Select the controller and click on NEXT.

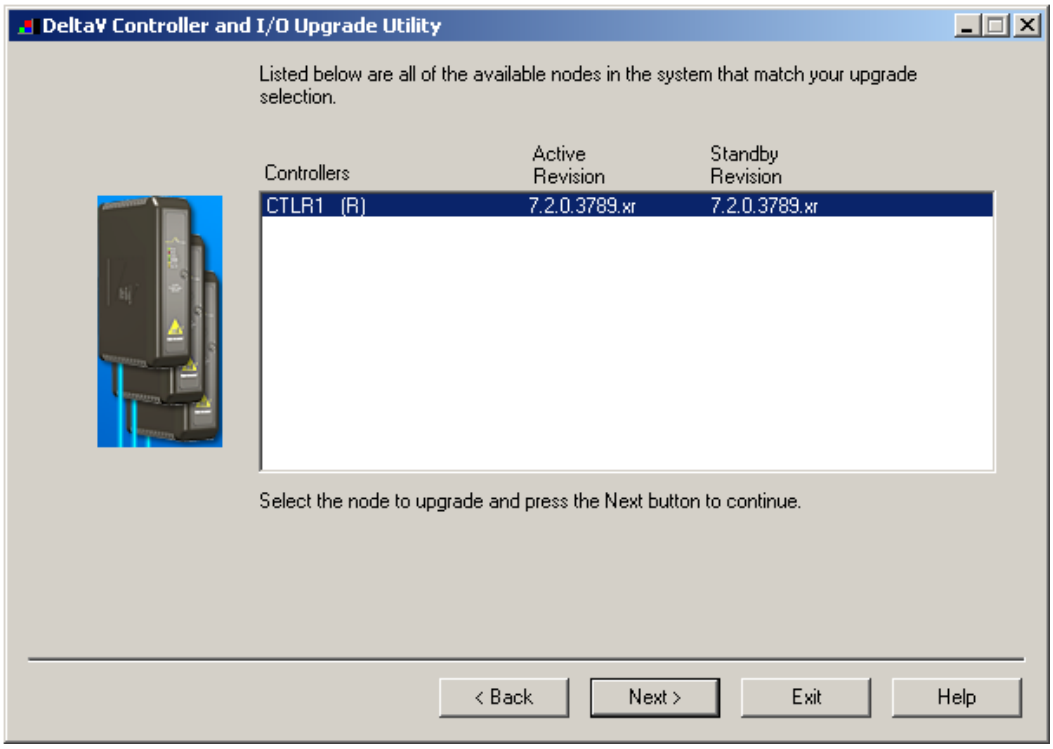


Figure 3 - Controller Upgrade Utility - Select Controller

4. Browse down to the **C:\DeltaV\ctlr\ProgSerial\RMV9000** folder then select the Programmable Serial Card you want to download the MVCU3 driver to, then click NEXT.

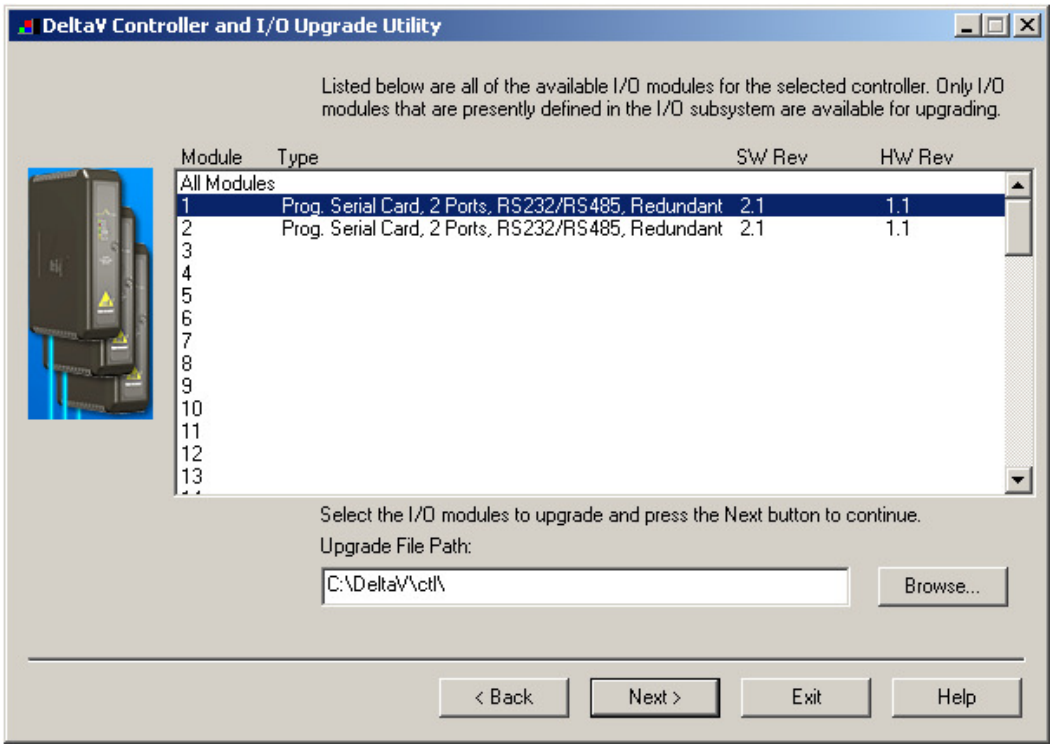


Figure 4 - Controller Upgrade Utility - Select Serial Card

5. Browse down to the **C:\DeltaV\ctlr\ProgSerial\RMV9000** folder and select the **RMV9000.S2F** file and click OPEN as shown below.

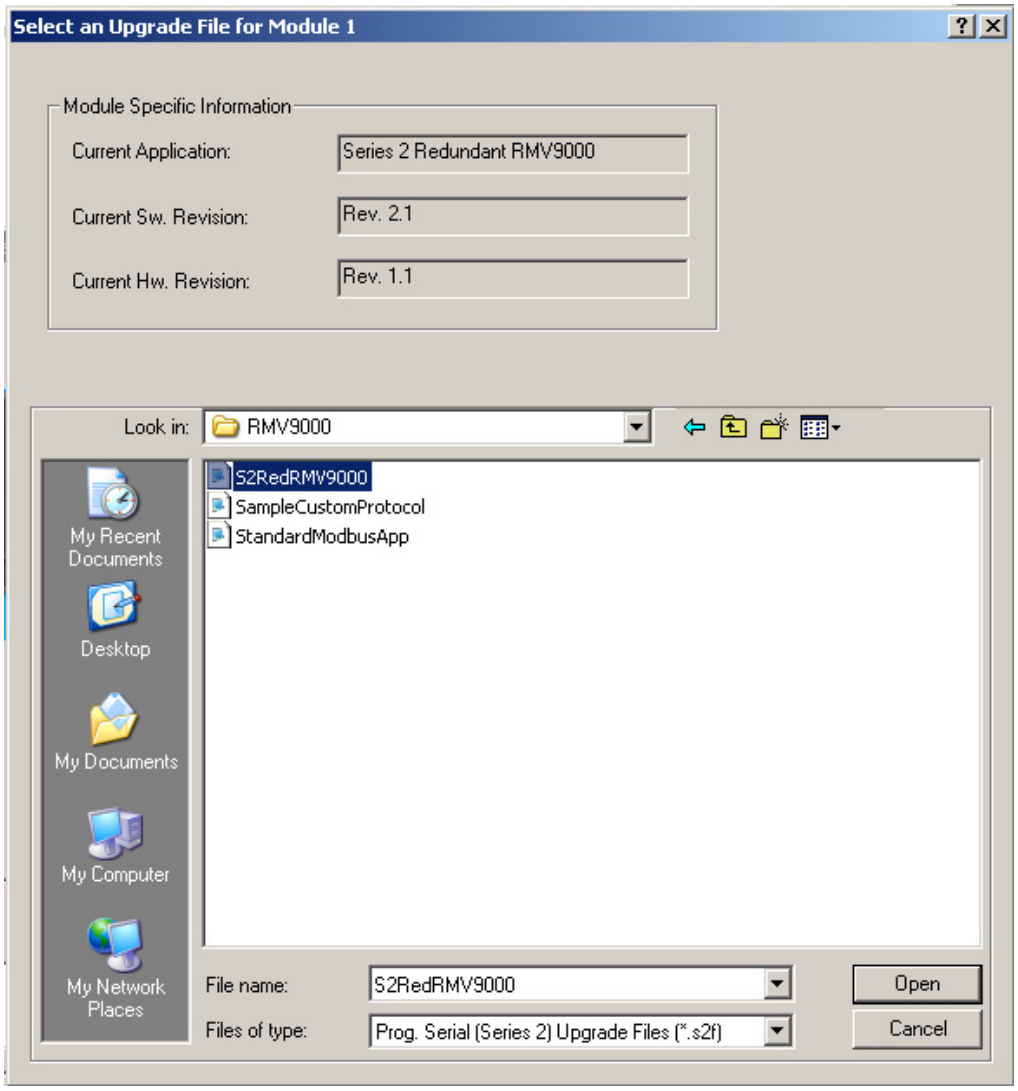


Figure 5 - Select Upgrade File

6. Click NEXT to begin the download of the MVCU3 protocol image into the selected serial card. The download will take several minutes. Once the download is complete, this screen will be displayed.

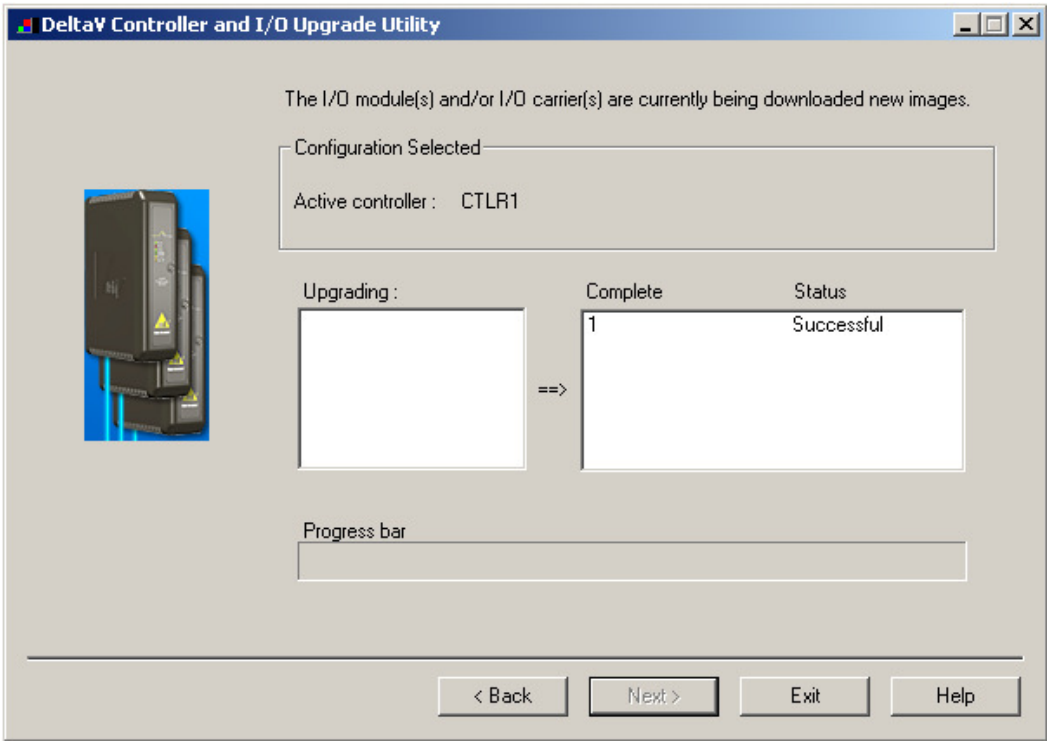


Figure 6 - Controller Upgrade Utility - Download complete

7. When the download is complete, click on BACK to confirm the MVCU3 protocol image has downloaded properly.

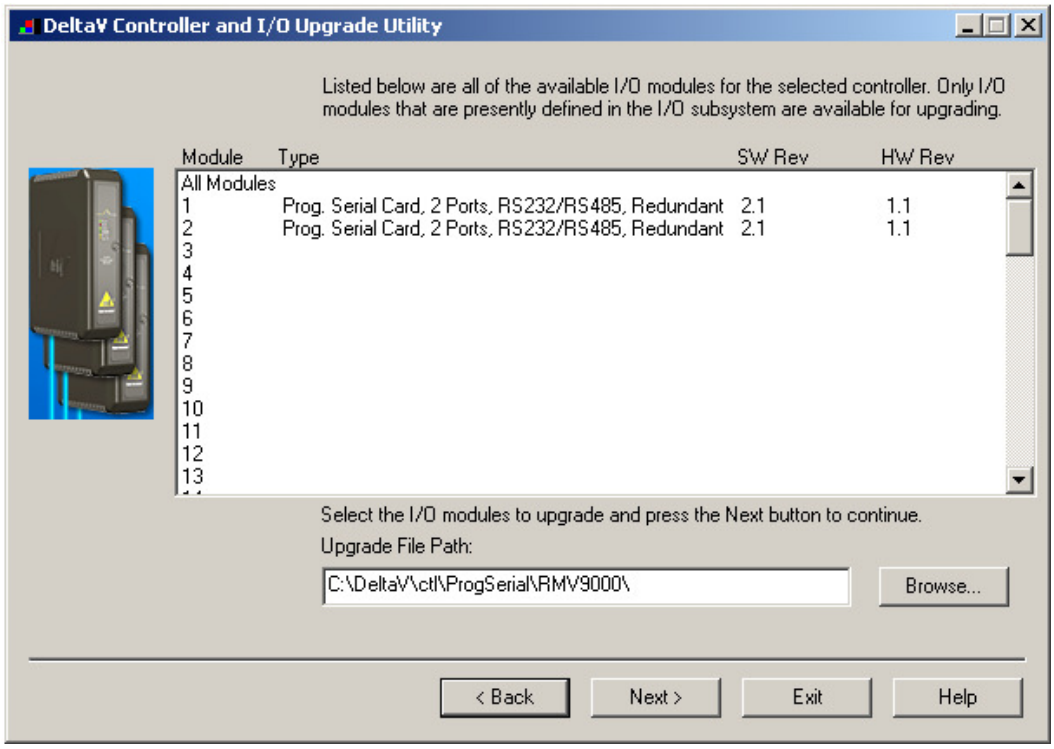


Figure 7 - Controller Upgrade Utility - Confirm Revision

The current software version is V3.2

Default Serial Port and Dataset Installation

The MVCU3 data that is available utilizing the DeltaV control modules provided may be found in Appendix B of this User Guide. To install the default serial port and Dataset configuration follow these steps.

1. Using the DeltaV Explorer, Click on **CTLR1** | **I/O** and select the **File** | **Import** | **Standard DeltaV Format** from the Explorer menu bar. Import the **IO_RMV9000** file from the **C:\DeltaV\DVData\Import-Export\IO_RMV9000** folder. Disregard and "Unresolved Reference" errors that may appear in the log.

2. After the **IO_RMV9000** file has been imported, download the Physical Network using the DeltaV Explorer.

The "Default" serial port configuration is:

Simplex PSC
- 19,200 Kbaud
- 8 data bits
- 2 stop bits
- No Parity
- RS232

Installing DeltaV Templates and Format Specification Files

The following procedure outlines the method for copying and importing control module templates supplied on the distribution media (CD) so they can be reused to support the MVCU3 data types.

Step 1 - Importing Module Templates

1. Using the DeltaV Explorer, create a new category named **RMV9000** under the **Library** | **Module Templates** directory tree.

2. The module templates are compressed into the “RMV9000_Templates.FHX” file. These templates are copied to the **C:\DeltaV\DVData\Import-Export\Template\RMV9000** folder on the Pro+ Workstation during installation.

3. Using the DeltaV Explorer, CLICK on **Library** | **Module Templates** | **RMV9000** template category and select the **File** | **Import** | **Standard DeltaV Format** from the Explorer menu bar. Import the “RMV9000_Templates.FHX” module templates for later reuse.

Step 2 – Installing Format Specification Files

The Format Specification Files are used by the Bulk Editor to create customer “database” configurations. These configurations are the set of control modules that support the Item “tagnames”. The DeltaV explorer will allow importing User Defined Format files and will automatically create control modules. The contents of the Format Specification Files are based upon the Module Templates.

The Format Specification files are automatically installed in the **C:\DeltaV\DVData\BulkEdit\RMV9000** folder during installation.

MVCU Faceplates and Detail Displays

The RMV Interface includes Faceplate and Detail Displays that are supplied on the distribution media (CD) so they can be used by the DeltaV Operate control environment.

MVCU Faceplate Displays

The MVCU Faceplate displays are automatically copied to the **C:\DeltaV\DVData\Graphics-iFix\Pic\Faceplate** folder on the Pro+ station during installation.

MVCU Detail Displays

The MVCU Detail displays are automatically copied to **C:\DeltaV\DVData\Graphics-iFix\Pic\Detail** folder on the Pro+ station during installation.

Installing DeltaV Security Parameters

The RMV Interface uses the Parameter Security feature of DeltaV to regulate access to control, loop tuning, and administrative attributes on the Faceplate and Detail displays. The security levels applied to RMV Interface DeltaV parameters are operationally equivalent to those used by OSP or IGS operator stations.

An enhanced Parameter Security file is copied into the **C:\DeltaV\DVData\Import-Export\Template\RMV9000** folder on the Pro+ station during installation.

Importing Security Parameters

Using the DeltaV Explorer, CLICK on the **File** | **Import** | **Standard DeltaV Format** from the Explorer menu bar. Select the “Security” file from the **C:\DeltaV\DVData\Import-Export\Template\RMV9000** folder on the Pro+ Workstation and import the Parameter Security file. Download the Pro+ station.

Installing Sample Modules

Installing Sample Control Modules (optional)

The distribution media (CD) includes a set of sample modules that may be used in conjunction with the sample Graphics. These modules and graphics may be used for demonstration and test and are not supported as part of actual RMV Interface.

1. Use the NT Explorer, create a folder named **RMV9000_Modules** under the **C:\DeltaV\DVData\Import-Export** folder.

2. Copy the Sample Modules from the **Sample_Modules** directory on the distribution media (CD) to the **C:\DeltaV\DVData\Import-Export\RMV9000_Modules** folder.

3. Using the DeltaV Explorer, CLICK on **CTLR1** | **Area_A** plant area and select the **File** | **Import** | **Standard DeltaV Format** from the Explorer menu bar. Import all of the Control Module "templates" for later reuse. Down load the Physical Network once all of the modules have been imported.

Section 3 – Hardware Setup

Introduction

This section of the MVCU3 Interface document will contain information relative to the physical serial connection between the MVCU and the DeltaV serial card.

The MVCU interface to DeltaV will only support MVCU controllers that are equipped with MVCU3 CPU boards. The MVCU controller was "repackaged" in 1989 from a "drawer" style configuration into a "front access" hardware configuration. Although the MVCU was redesigned as a "front access" configuration, many users have upgraded their "drawer" style MVCUs with MVCU3 CPU boards and may take advantage of this interface serial interface to the DeltaV system.

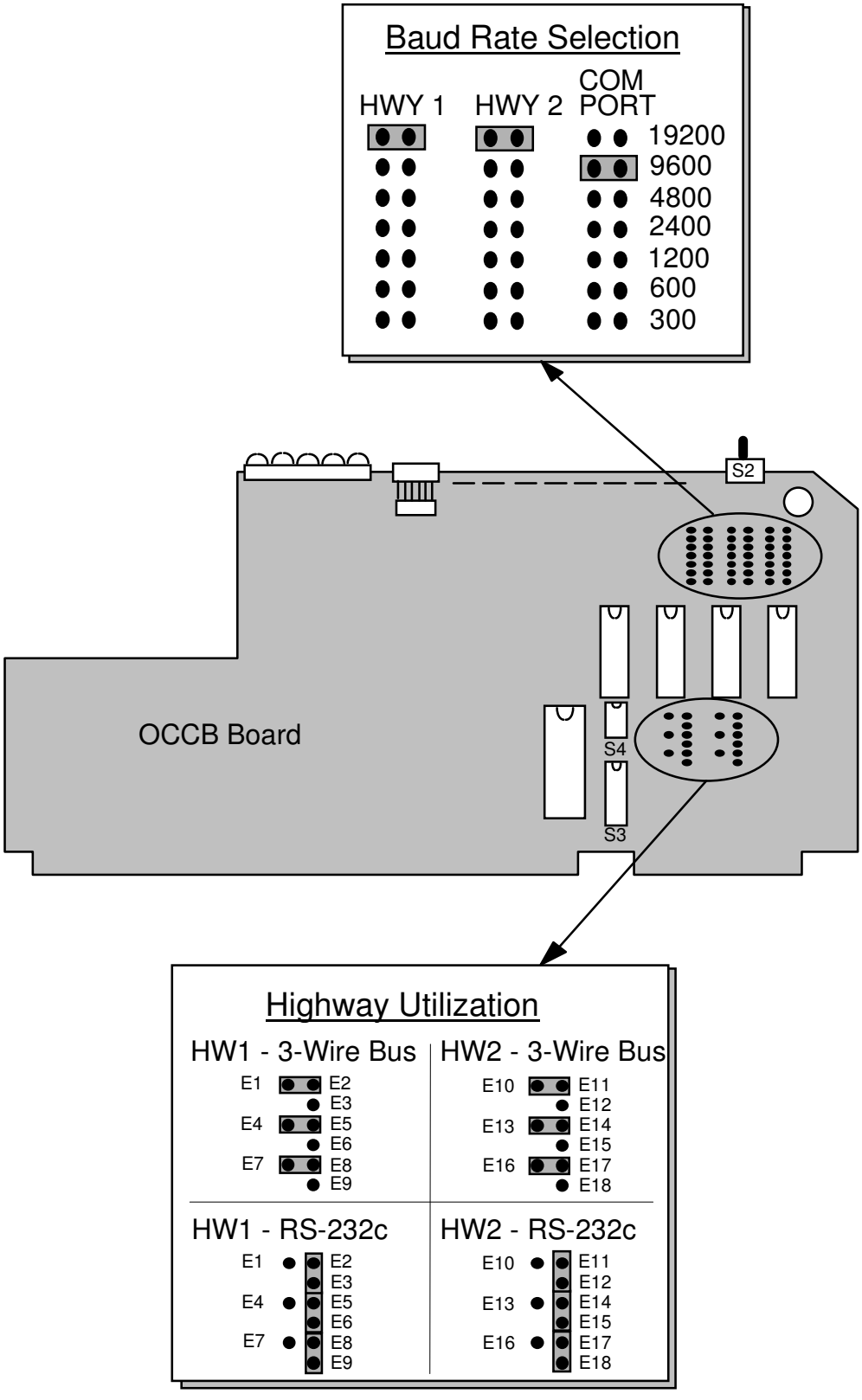
With respect to the physical RS232 connection between the "drawer" style and the "front access" MVCU, the drawer" style utilizes a 25-pin connector while the "front access" MVCU features a 9-pin connector. The correct RS232 cable pin outs for these different MVCU hardware configurations will be detailed in this section of the User Guide.

Tabulated below are the different MVCU derivatives that may utilize the serial interface to DeltaV.

MVCU Model Number	DeltaV Connection	25- pin Connector	9 - pin Connector	Upgradeable to MVCU3
8010				
8011				
8012				
8013	X	X		X
8014	X	X		X
8015	X	X		X
8021				
8023	X		X	X
8024	X		X	X
8025	X		X	X
8033	X		X	
8034	X		X	
8035	X		X	

MVCU Serial Ports

The MVCU type 2 and 3 controllers, regardless of hardware packaging all feature three (3) communication interfaces. Highways A and B are jumper selectable between 3-wire bus (OCB) and RS232 communications. Highways A and B may be converted from normal 3-wire bus (OCB) implementation to RS232 configuration to provide a redundant communication interface to the DeltaV serial card. Additionally, all MVCU communication ports feature individual Baud rate selections. The jumper settings to convert Highways A and B from 3-wire bus to RS232 and to change communication Baud rates are illustrated in the below.



MVCU Jumper Settings

DeltaV Serial Ports

The DeltaV Serial card may be configured for either RS232 to RS422/485. If RS422/RS485 is used, the signal from the DeltaV PSC will have to be converted to RS232. Many RS485 to RS232 converters are available. Due to the low line voltage of the MVCU's RS232 signals, "line-powered" converters may not work. It is recommended that "externally" powered converters be used.

The Telebyte Model 365 signal converter is recommended.

See the table below for the permitted serial port configurations:

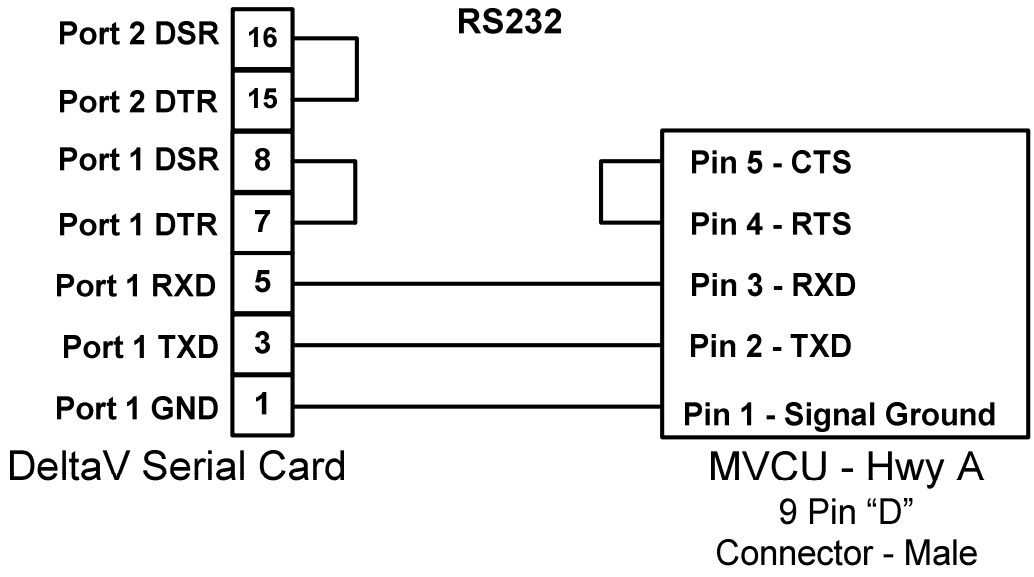
	RS232	RS422 / RS485 (Half Duplex)	Notes
Simplex Serial Cards	**Direct Connection**	**Need Converter**	**Simplex Terminal Block**

MVCU to DeltaV Cable Configurations

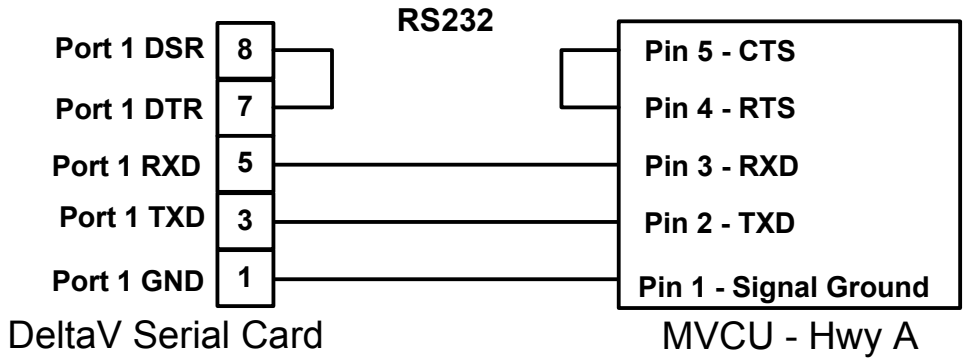
Simplex Cards (Type 1 or Type 2)

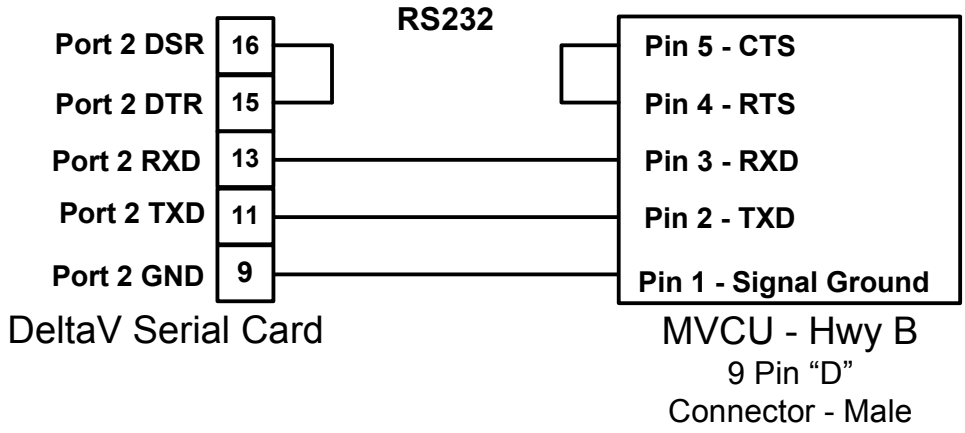
The cable used and the connection to DeltaV is dependent upon the MVCU controller type and the method of communications. The following illustrations depict the different cable configurations. Terminal 15 to Terminal 16 must be jumpered to access Virtual Data Sets.

DeltaV to Front Access MVCU Connection (RS232)
Single Connection to MVCU

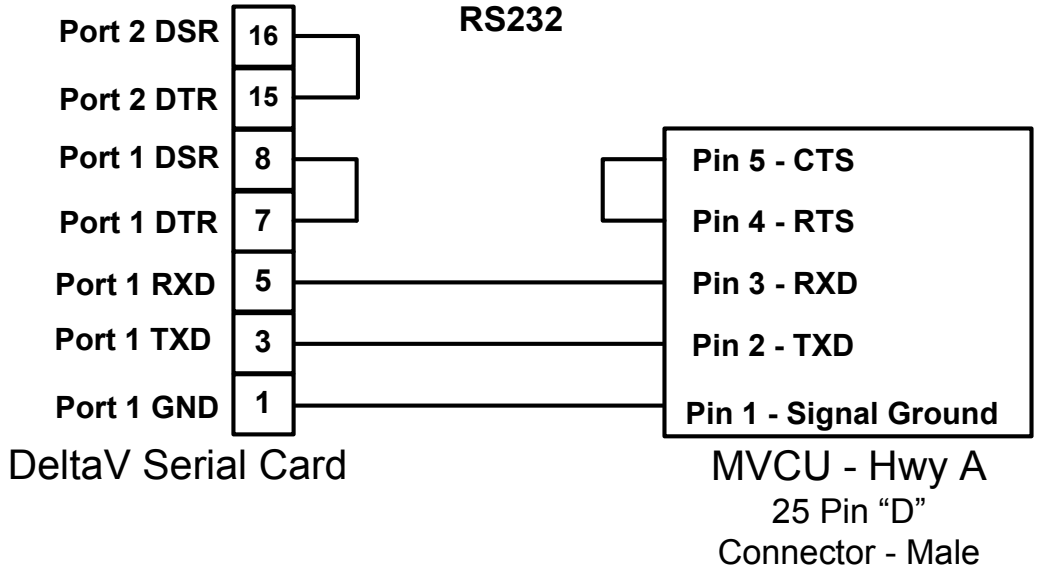


DeltaV to Front Access MVCU Connection (RS232)
Redundant Connection to MVCU

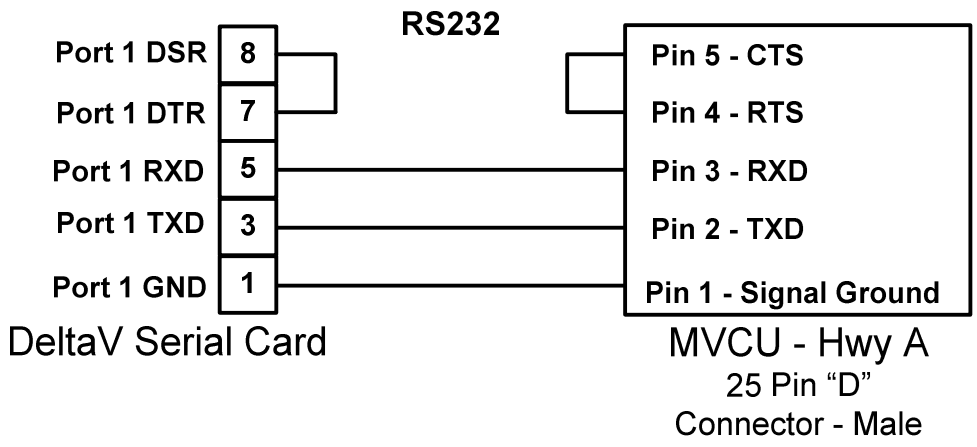


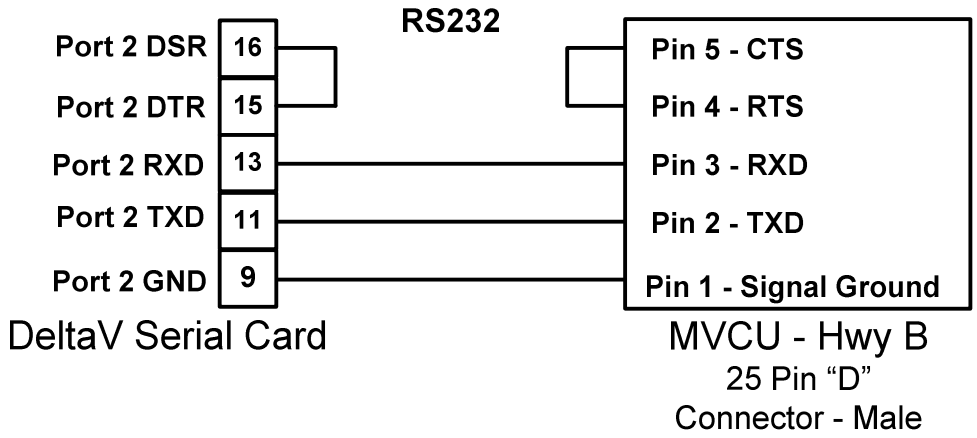


DeltaV to "Drawer Style" MVCU Connection (RS232)
Single Connection to MVCU



DeltaV to "Drawer Style" MVCU Connection (RS232)
Redundant Connection to MVCU





Series 2 Serial Card Diagnostics

The following LED indicators are provided by the RMV Interface Driver.

LED Indicator	Card / Port Condition
Green – Solid	Card is “Selected”
Green – Blinking	Card is “Non-Selected” (standby)
Port LED – Yellow Solid	Port is communicating properly
Port LED – Yellow Blinking	Port has lost communications
Port LED – Not Illuminated	Port not configured

Series 2 Serial Card Switchover Logic

When using the RMV Interface with the Series 2 PSC; the following card switchover logic is used:

- Loss of communications due to line breaks or disconnection with the MVCU will not force the Active card to switch with to the Standby card.
- The standby card will only be “selected” as diagnosed by internal card diagnostic sensing the failure of the Active card or my manual switchover using the DeltaV diagnostics application.
- If the Active card is inadvertently “flashed” with a new protocol image, the standby card will be selected.

Section 4 - Adding/Editing Serial Cards, Devices, and Data Sets

Adding Additional Serial Cards

To add additional Programmable Serial cards to the DeltaV controller carrier, first install the card and the termination block.

Step 1 – Add a Serial Card

After the hardware is installed, select the **CTRL1** | **I/O** icon from the DeltaV explorer tree. Right click the mouse and select **New Card** from the option list. The following dialog box will be presented. Select the **Prog Serial Card**. Do NOT use the Auto Sense option to configure the Serial Card.

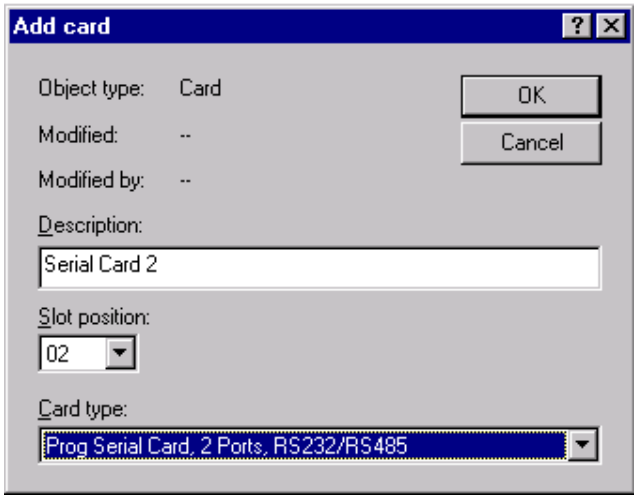


Figure 9 - Add Serial Card

Step 2 - Configure Serial Port Parameters

Once the serial card is sensed by the controller, communication port parameters may be configured. Click on the serial port (P01 or P02) associated with the serial card added to the carrier, then right click the port icon and select **Properties**. The following display will be presented.

1. Click on the Enable check box to enable the serial port then select the **Advanced** tab.

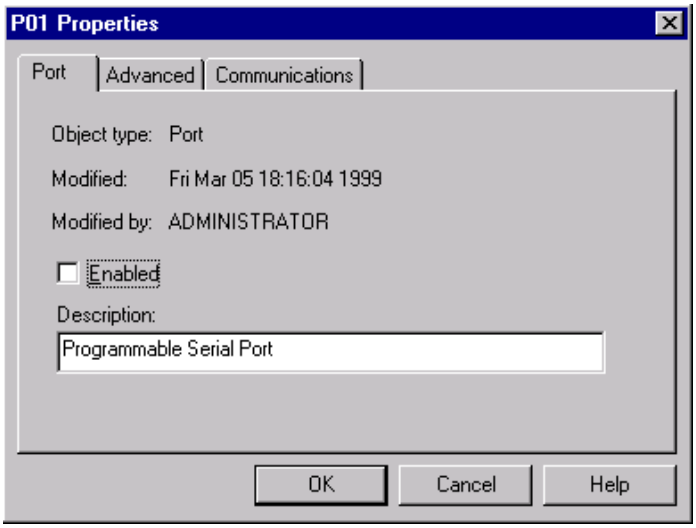


Figure 10 - P01 Properties - Port

3. Select the Advanced options shown below then select the **Communications** tab.
   - NOTE: The “Send outputs on startup” must be “deselected”.

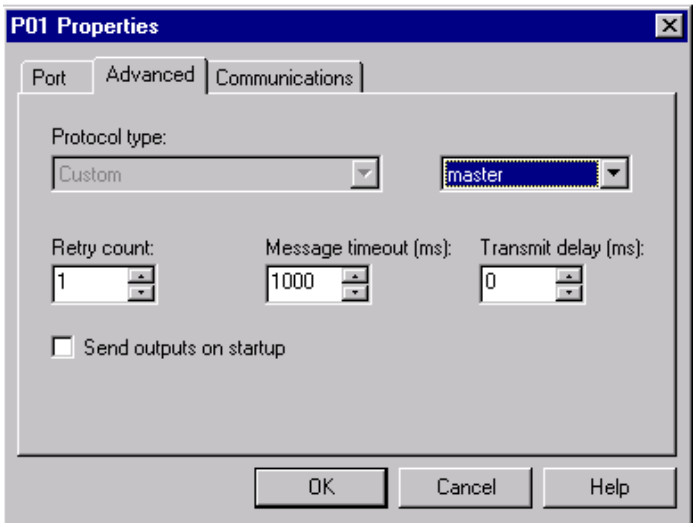


Figure 11 - P01Properties - Advanced

4. Select the **Communications** tab and select the baud rate and port type, The MVCU is has a fixed protocol frame of 8 data bits, no parity, and 2 stop bits. After all tab options are selected click on **OK** then download the "Physical Network" using the DeltaV Explorer.

   - Select RS232 or RS422/RS485 as the Port Type]

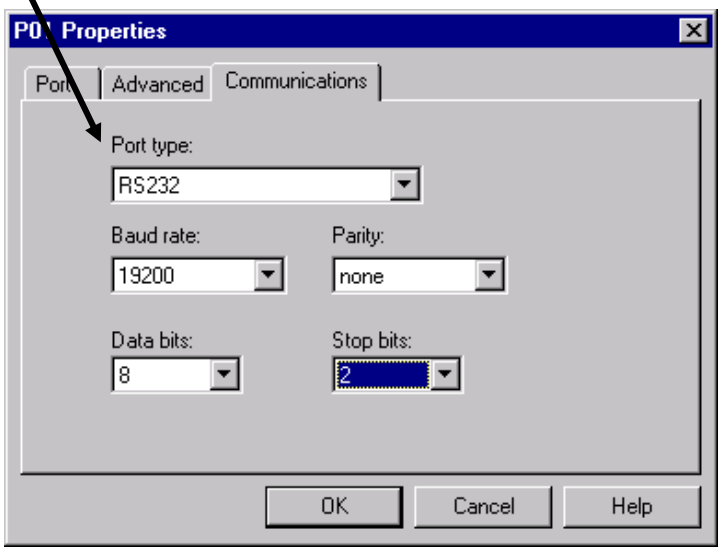


Figure 12 - P01Properties - Communications

Step 3 - Downloading Serial Port Parameters

Once all of the new serial port parameters have been configured, download the "Physical Network" using the DeltaV Explorer.

Adding Additional Serial Devices

To configure additional serial devices connected to the serial card, click on the communication port icon (P01 or P02) then right click the mouse and select the **New Serial device** option.

*** Note that the RMV9000 to DeltaV interface is hard coded for redundant communications and serial ports 1 and 2 MUST be configured as the same device address.

After selecting the **New Serial device** option, the following dialog box will be presented.

Step 1 - Configuring a new Serial Device

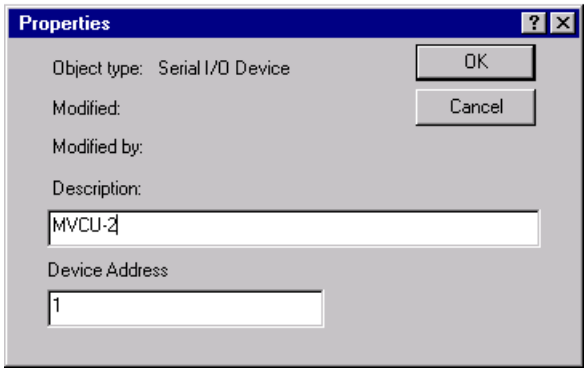


Figure 13 - Properties - New Device

Enter the name and device address of the new device then click on the **OK** button then download the “Physical Network” using the DeltaV Explorer. After the new device has been established, the communication parameters may be configured according to the guidelines found in the preceding section “Configure Serial Port Parameters”.

Step 2 - Downloading New Serial Device

Once the new serial device has been configured, download the “Physical Network” using the DeltaV Explorer.

Adding Additional Data Sets

A set of default Data Sets are included with the interface distribution. Depending upon the actual Items (tagnames) that need to be supported based on the CCM database entries for the MVCU, Data Sets may need to be added or deleted to the serial card configuration. The scanning method and addressing selections for the Data Set is MVCU specific.

NOTE : The MVCU Interface utilizes both of the Serial Cards ports in a load sharing polling method. To achieve the best serial data throughput, Data Sets should be as evenly split between the two serial ports as possible.

To add additional Data Sets to a serial card, the following procedure is to be followed.

General Notes

1) Data Sets 1 to 16 are assigned to Port 1 of the Serial Card, Data Sets 17 to 32 are assigned to Port 2.

2) A method of "redundancy "is featured by the serial interface that will redirect messages to the active port should one of the ports fail. This feature called "Virtual" Data Sets. This unique feature allows Data Sets 17 - 32 to be mapped to port 1 even though there is no physical connection between the MVCU3 and Port 2, for example.

3) Refer to the "Data Set Configuration and Register Assignments" (Appendix A) of this user guide for specific configuration and mapping requirements.

Step 1 - Adding Data Sets

To add more data sets, select the serial device (DEV01) with a left mouse click, then select **New Dataset** from the menu. Refer to the "Data Set Configuration and Register Assignments" (Appendix A) of this user guide for correct configuration. The following screens illustrate the default configuration for Analog Input Blocks.

1. Enter the Description, Data Direction, Output Mode, and Read back mode as detailed in the "Data Set Configuration and Register Assignments" (Appendix A) of this user guide.

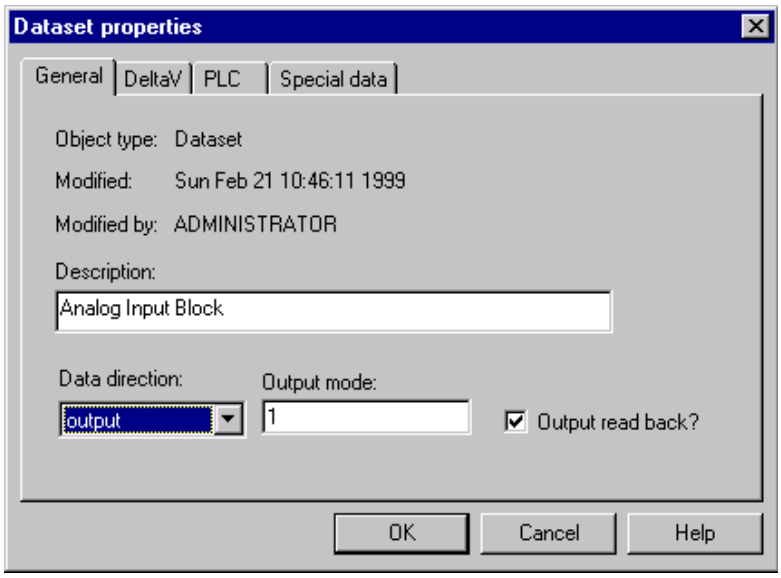


Figure 14 - Add Dataset Properties - General

2. Enter the DeltaV data type and Dataset tag (new assignment) as detailed in the "Data Set Configuration and Register Assignments" (Appendix A) of this user guide.

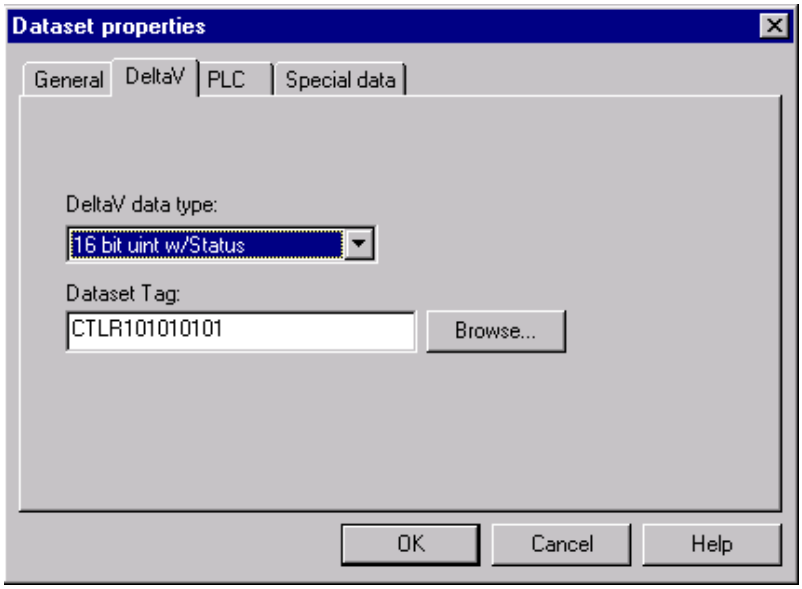


Figure 15 - Add Dataset Properties - DeltaV

3. Enter the Device data type, Data start address, and Number of values as detailed in the “Data Set Configuration and Register Assignments” (Appendix A) of this user guide.

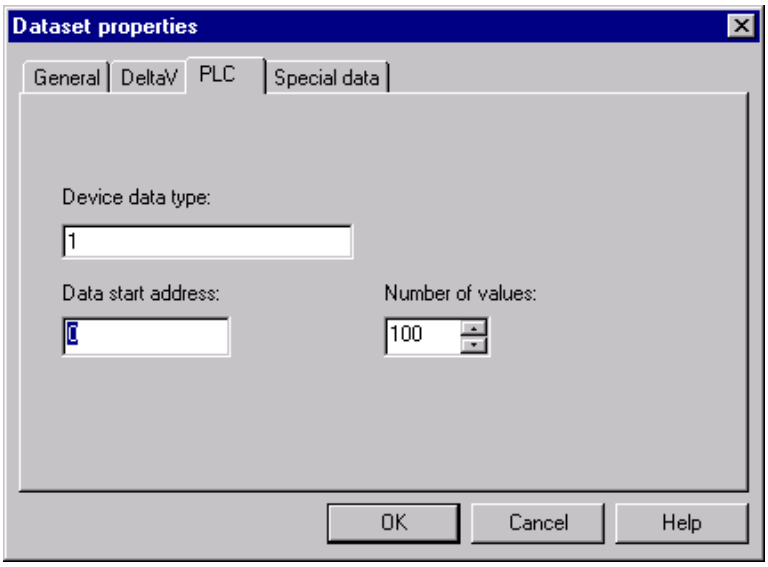


Figure 16- Add Dataset Properties - PLC

4. Enter the Special data value as detailed in the “Data Set Configuration and Register Assignments” (Appendix A) of this user guide.

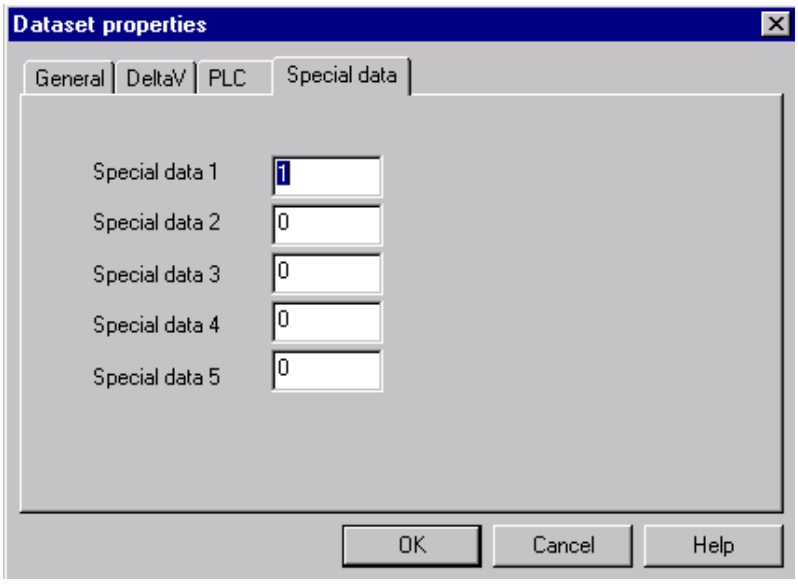


Figure 17- Add Dataset Properties - Special Data

Creating New Loop Tagnames

Installation specific loop Tagnames are created based upon the control module "templates" supplied on the distribution media. Once these control modules are imported into **AREA_A**, they may be replicated. Refer to Appendix B of this manual to review the data that is available for the different data types that are provided by the interface.

Step 1 - Open and Copy the Module Template

Open the module with the same data type as the loop tagname with Control Studio. Copying the module template using the **Save_As...** option found under the **File** option on the Control Studio menu bar. Save the module template using the loop tagname that will be used by the operator interface and database. By naming the new module with the "tagname" of the loop object, the Graphics Studio will be able to Browse and identify the new item.

Step 2 - Edit Module Parameters

After the module has been copied, reassign the module's external references. For modules that utilize Sparse Space addressing, assign the internal address of the MVCU data type as the ADDRESS parameter in the control module. Depending upon the data type of the new module, I/O scaling and alarm limits must be specified. Refer to Appendix C of this manual as a guide for parameter addressing.

Step 3 - Module Download

After the new module has been edited, assign the module to a controller node such as **CTLR1**. Once the module has been assigned to a controller, download the "Physical Network" using the DeltaV Explorer.

Contained below are several samples of control module "templates" supplied on the distribution media.

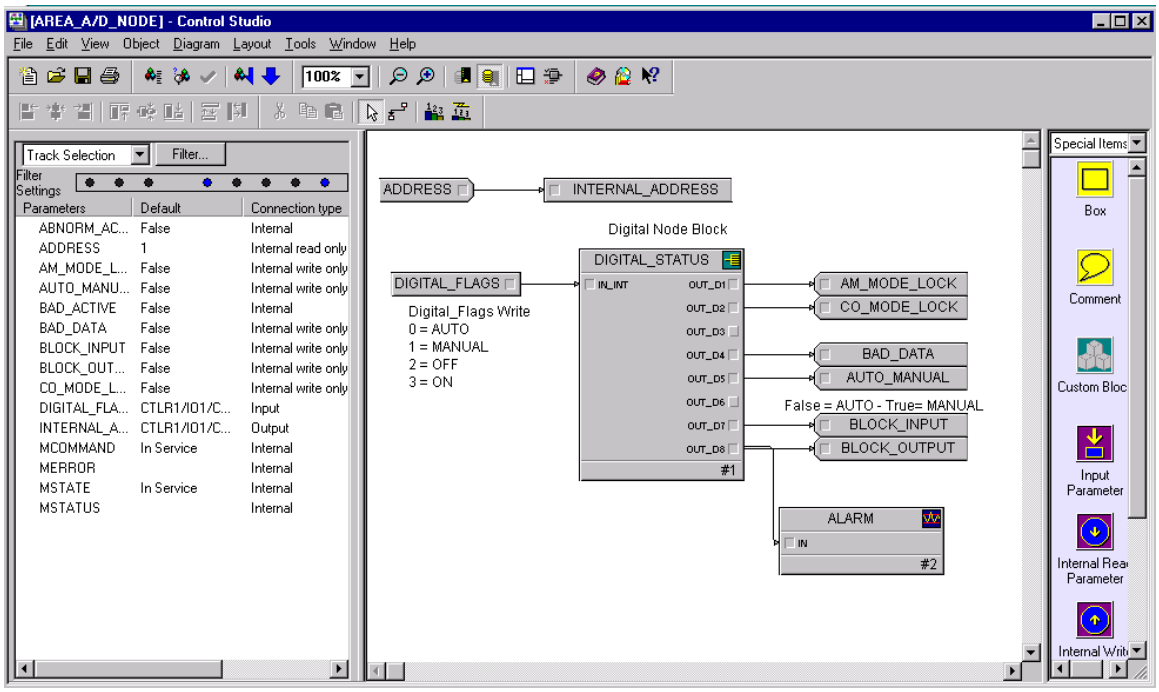


Figure 18- Digital Node Module

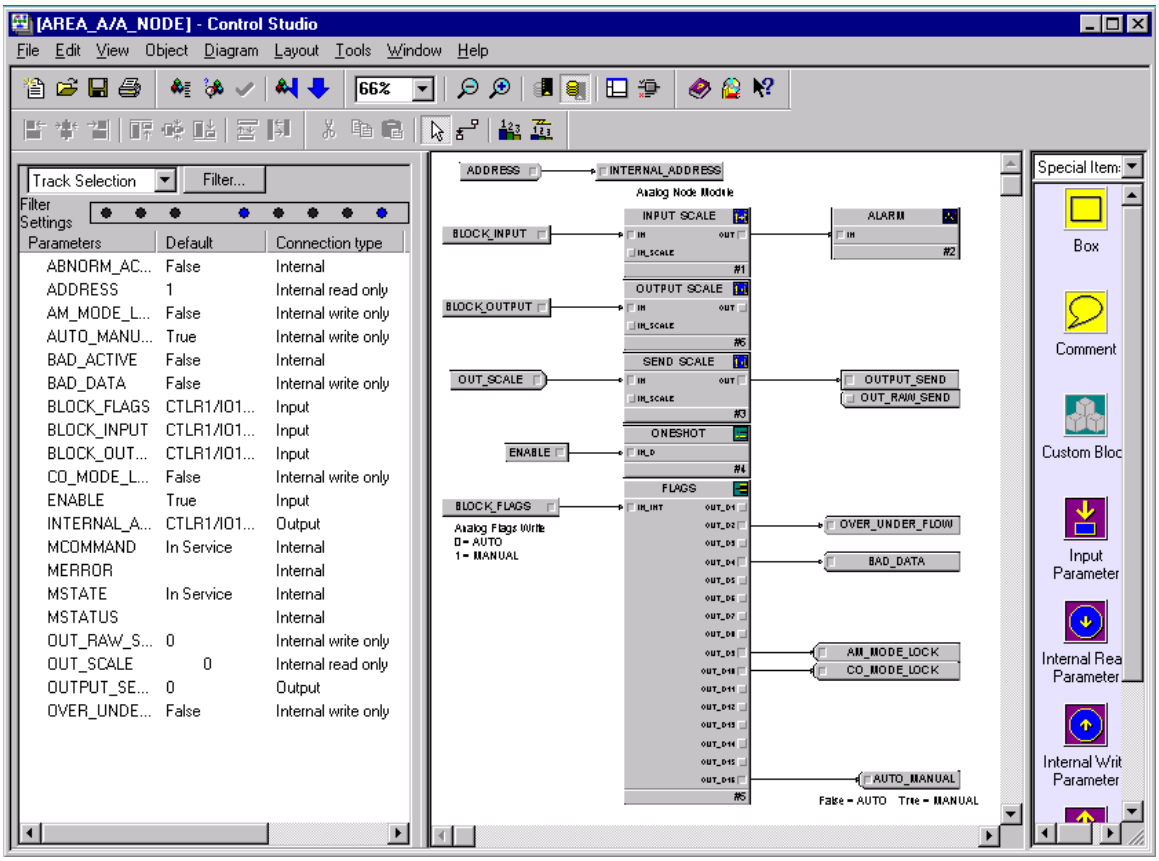


Figure 19- Analog Node Module

Appendix A - Data Set Configuration and Register Assignments

Data Set Configuration and Register Assignments

This document defines the correct DeltaV serial card configurations for use with the RMV/MVCU interface. The RMV interface supports several DeviceData Types, Addressing Modes, and Scanning Modes. The relationship of these, and the default Data Set configurations, will be outlined in this document.

The DeltaV serial card supports 32 Data Sets and each Data Set contains 100 registers. Floating point and 32 bit integer data sets are 50 registers deep. Data Sets 1-16 are associated with Port 1 of the Serial Card and Data Sets 17-32 are associated with Port 2. The number of Data Types that populate a Data Set is determined by the Addressing Mode, Scanning Mode, and the Data Type. All of the variables in the data set must be the same type.

Data Set Configuration Options

Attribute	Defined Values
Description	User Defined
Data Direction	Output
Output Mode	1 = Single Value
Output Read Back	Enabled
DeltaV Data Type	Variable Dependent (defined below)
Dataset Tag	The number of the dataset
DeviceDataType	0 = MVCU Device Status 1 = Analog Input Block 2 = Analog Output Block 3 = Analog Node Block 4 = Digital Input Data 5 = Digital Output Data 6 = Digital Node Data 7 = Analog K Factors 8 = Analog Algorithm Data 9 = Digital Algorithm Data

	10 = Floating Point Node 11 = Analog Algorithm Floating Data
Data Start Address	0
Number of values	DataType and Scan Mode Dependent (defined below)
SpecialData1	Address/Scan Mode 0 = Normal Contiguous Addressing w/ normal continuous scanning 1 = Sparse Space Addressing w/ normal continuous scanning 2 = Normal Contiguous Addressing w/ on demand scanning 3 = Sparse Space Addressing w/ on demand scanning 4 = Block Transfer – Sparse Space Addressing w/ normal continuous scanning. Only applies to ACLs and DCLs. 5 = Block Transfer - Sparse Space Addressing w/ on demand scanning. Only applies to ACLs and DCLs. (not used)
SpecialData2	MVCU element starting address NOTE – the field DeviceStartAddress is only used to set the starting DeltaV register number, SpecialData2 is the actual address used when reading the MVCU. This field only applies to the contiguous addressing modes 0 and 2.
SpecialData3	Refresh scan time, used with addressing modes 2, 3 and 5. This is the amount of time, in seconds, between automatic refresh scans of the data set. A value of 0 disables the automatic refresh. A write to the data set that changes a register value causes an immediate scan and resets the timer.
SpecialData4	ScanOffset, used with addressing modes 2, 3, and 5. This is the offset from time zero to apply the refresh scan time in seconds.

Addressing Modes

The MVCU3 to DeltaV interface supports two types of scanning modes to maximize the efficiency of the registers available in the 32 total data sets. If all the possible data in the MVCU were to be scanned, the number of total registers available would be quickly depleted. For this reason two Addressing methods were developed for the best use of serial card memory.

Normal Contiguous Addressing

This scanning method will read MVCU data types in a block. All addresses are contiguous. In the case of Digital Inputs and Digital Outputs, for example, this is the most efficient use of Data Sets even though all of the values read may not be needed by the user interface.

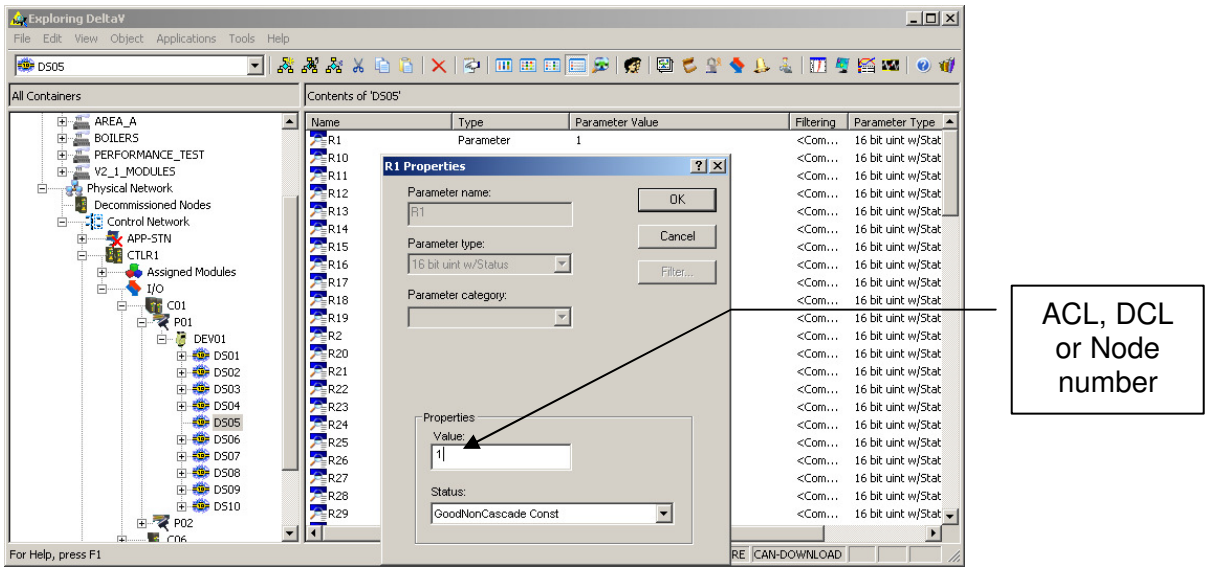
Sparse Space Addressing

In most cases it is not desired to read contiguous data types. For example, reading a big block of contiguous Analog Algorithms is a waste of Data Set memory and communication bandwidth. Sparse Space Addressing is a method for selective scanning of discontiguous variables from the MVCU. In this method the Data Set is configured to read selectable variables. This is the most efficient way to scan Analog Algorithms, for example, since they require many registers.

This is how Sparse Space Addressing works. The upper portion of the Data Set is reserved for Sparse Space Pointers. Using an Internal Address parameter in a module, a Block or Function address is written to the Sparse Space Pointer register of the Data Set. The Serial Card will then query the MVCU for the data associated with the address. The returned data is then written into the appropriate data area of the Data Set. The dynamic parameters within the modules then read data out of dynamic data of the Data Set.

Note: The dataset register “initial value” needs to be configured and contain the address of the ACL, DCL, spare space node, ACL number for K factor and floating point datasets.

This is essential for the MVCU “prescan” to work properly before a serial card redundancy switch.



Scanning Modes

Normal Continuous Scanning

This method of scanning is used when a variable needs to be scanned continuously. This method of scanning is used most of the time.

On Demand Scanning

This scanning method is used for Analog Function. On Demand scanning of Tuning Parameters is triggered off of a configurable timer specified by Special Data 3. This method of scanning improved system data throughput.

Block Transfer Scanning

Block Transfer is used to scan Analog and Digital functions. The MVCU protocol for Analog and Digital function allows block transfer of discontiguous functions. Essentially, a list of sparse space pointers are sent to the MVCU and the MVCU returns all data in a single response message.

Virtual Data Sets

The RMV interface incorporates Virtual Data sets. Virtual data sets allow data sets 17-32 that are assigned to Port 2 of the serial card to communicate over Port 1. This means in effect that Port 1 can utilize all 32 set even though data sets 17-32 are configured for Port 2.

Data Set attributes by Data Type

The table below defines the default Data Set configuration by MVCU Data Type.

Normal Contiguous Addressing Address/Scan Mode 0			
MVCU Data Type	**Number of Elements per Data Set**	**First Element Register Number and Interval**	**Number of Registers and DeltaV Data Type**
Analog Inputs	32	R1, R4, R7, … (3)	96 unsigned integer 16
Analog Out	32	R1, R4, R7, … (3)	96 unsigned integer 16
Analog Nodes	32	R1, R4, R7, … (3)	96 unsigned integer 16
Digital Inputs	64	R1, R2, R3, …	64 unsigned integer 16
Digital Outputs	64	R1, R2, R3, …	64 unsigned integer 16
Digital Nodes	100	R1, R2, R3, …	100 unsigned integer 16

Sparse Space Addressing Address/Scan Mode 1 & 3			
MVCU Data Type	**Number of Pointers/Elements per Data Set**	**First Element Register Number and Interval**	
MVCU Status	1	R1	74 unsigned integer 16
Analog K factors	6	R8, R15, R22, … (7)	49 Floating Point
Analog Nodes (Optional)	25	R26, R29, R32, … (3)	100 unsigned integer 16
Floating Point Nodes	12	R13, R16, R19, … (3)	48 Floating Point
Analog Algorithm Floating Data	25	R26, R27, R28, ... (1)	50 Floating Point

Note: The dataset register “initial value” needs to be configured and contain the address of the ACL, DCL, spare space node, ACL number for K factor and floating point datasets

Block Transfer Sparse Space Addressing Continuous Scanning Address/Scan Mode 4			
MVCU Data Type	**Number of Pointers/Elements per Data Set**	**First Element Register Number and Interval**	
Analog Algorithms	12	R13, R20, R27, … (7)	96 unsigned integer 16
Digital Algorithms	12	R13, R16, R19, … (3)	48 unsigned integer 32

Default Data Sets

The following default Data Sets are provided with the DeltaV interface package.

PORT 1 Data Sets

Data Set Number	Device Data Type	DeltaV Data Type	Device Data Type	Data Start Address	Number of Values	Special Data 1	Special Data 2
1	MVCU Device Status	Unsigned integer 16 bit w/Status	0	0	74	1	0
2	Analog Input Block 1 - 32	Unsigned integer 16 bit w/Status	1	0	96	0	1
3	Digital Input Block 1 – 64	Unsigned integer 16 bit w/Status	4	0	64	0	1
4	Digital Node Block 1 –100	Unsigned integer 16 bit w/Status	6	0	100	0	1
5	Analog Algorithm	Unsigned integer 16 bit w/Status	8	0	96	4	0
6	Floating Point Node	Floating Point w/Status	10	0	48	1	0
7	Analog K Factor	Floating Point w/Status	7	0	49	3	0

PORT 2 Data Sets

1	Analog Input Block 33 –64	Unsigned integer 16 bit w/Status	1	0	96	0	33
2	Analog Output Block 1 – 32	Unsigned integer 16 bit w/Status	2	0	96	0	1
3	Digital Output Block 1 – 64	Unsigned integer 16 bit w/Status	5	0	64	0	1
4	Analog Node Block (Contiguous Address) 1 – 32	Unsigned integer 16 bit w/Status	3	0	96	0	1
5	Digital Algorithm	Unsigned integer 32 bit w/Status	9	0	48	4	0
6	Analog K Factor	Floating Point w/Status	7	0	49	3	0
7	Analog Algorithm Floating Point Data	Floating Point w/Status	11	0	50	1	0

Data Mapping

MVCU Device Status

MVCU Device Status is mapped as follows. Scanning is addressing mode 1 – Sparse Space Addressing.

MVCU Device Status (*DeviceDataType* 0)	
MVCU Data	**DeltaV Register**
Side-Side Checksum Mismatch	R1
Side A Status and Master CPU Status	R2
Reserved	R3
Side A Hardware Errors	R4
Side A Hardware Errors and Card Ack Failure	R5
Side A Checksum Errors	R6
Not Used	R7
Side B Status and Master CPU Status	R8
Reserved	R9
Side B Hardware Errors	R10
Side B Hardware Errors and Card Ack Failure	R11
Side B Checksum Errors	R12
Not Used	R13
Year (00-99) – Read from MVCU	R14
Month (1-12) – Read from MVCU	R15
Day of Month (1-31) – Read from MVCU	R16
Hour (0-23) – Read from MVCU	R17
Minute (0-59) – Read from MVCU	R18
Second (00-59) – Read from MVCU	R19
Reset Dataset Statistics	R20

Comm Error - Invalid Address (code 02)	R21
Comm Error - Invalid Count (code 03)	R22
Comm Error - Invalid Block (code 04)	R23
Comm Error - Invalid Data (code 05)	R24
Comm Error - Invalid Mode (code 06)	R25
Comm Error - Mode Transfer Lock (code 07)	R26
Comm Port 1 Failure	R27
Comm Port 2 Failure	R28
Reset MVCU Diagnostics	R29
Send Date and Time – Send R63-R68 to MVCU	R30
Data Set 1 - Error Accumulator	R31
Data Set 2 - Error Accumulator	R32
. .	. .
Data Set 31 - Error Accumulator	R61
Data Set 32 - Error Accumulator	R62
Year (00-99) – Write to MVCU	R63
Month (1-12) – Write to MVCU	R64
Day of Month (1-31) – Write to MVCU	R65
Hour (0-23) – Write to MVCU	R66
Minute (0-59) – Write to MVCU	R67
Second (00-59) – Write to MVCU	R68
Scan Time – Port 1 (1/100's of a second)	R69
Scan Time - Port 2 (1/100's of a second)	R70
Message Count - Port 1	R71
Message Count - Port 2	R72
Line Errors - Port 1	R73
Line Errors - Port 2	R74

MVCU Device Status is mapped into 16 bit unsigned integer registers. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Analog Blocks

Analog Input, Output, and Node MVCU data types are mapped as follows. Scanning is addressing mode 0 – Contiguous Addressing.

Analog Input, Output, and Node Block Data (*DeviceDataType* 1, 2, and 3)	
MVCU Data	**DeltaV Register**
Block 1, Status Byte 1 and 2	R1 (uint16) Status byte 1 is mapped to bits 16-9 Status byte 2 to bits 8-1.
Block 1, Output	R2
Block 1, Input	R3
. .	. .
Block 32, Status Byte 1 and 2	R94 Status byte 1 is mapped to bits 16-9 Status byte 2 to bits 8-1.
Block 32, Output	R95
Block 32, Input	R96

Analog blocks are mapped to three 16 bit unsigned integer registers. This allows 25 analog blocks to be mapped into a single data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Analog Blocks (Optional Sparse Space Addressing)

Analog Node MVCU data types are mapped as follows. Scanning is addressing mode 1 – Sparse Space.

Analog Node Block Data (*DeviceDataType* 3)	
MVCU Data	**DeltaV Register**
Block 1, Sparse Space Pointer – Internal Address	R1
. .	. .
Block 25, Sparse Space Pointer – Internal Address	R25
Block 1, Status Byte 1 and 2	R26 (uint16) Status byte 1 is mapped to bits 16-9 Status byte 2 to bits 8-1.
Block 1, Output	R27
Block 1, Input	R28
. .	. .
Block 25, Status Byte 1 and 2	R98 Status byte 1 is mapped to bits 16-9 Status byte 2 to bits 8-1.
Block 25, Output	R99
Block 25, Input	R100

Analog blocks are mapped to three 16 bit unsigned integer registers. This allows 25 analog blocks to be mapped into a single data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Digital Input and Output Blocks

Digital Input and Digital Output MVCU data types are mapped as follows. Scanning is addressing mode 0 – Contiguous Addressing.

Digital Input and Output Block Data **(*DeviceDataType* 4 and 5)**	
MVCU Data	**DeltaV Register**
Block 1, Status and data	R1 (uint16)
. .	. .
Block 64, Status and Data	R64
Unused	R66 – R100 (these can be used to read additional digital nodes)

Digital Input and Output blocks are mapped to a single 16 bit unsigned integer register. This allows all 64 digital inputs and 64 digital outputs to be mapped into a single data set each. The digital blocks must be in consecutive memory locations. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Digital Node Blocks

Digital Node MVCU data types are mapped as follows. Scanning is addressing mode 0 – Contiguous Addressing.

Digital Node Block Data (***DeviceDataType*** **6)**	
MVCU Data	**DeltaV Register**
Block 1, Status and data	R1 (uint16)
. .	. .
Block 100, Status and Data	R100

Digital Node blocks are mapped to a single 16 bit unsigned integer register. This allows 100 digital nodes to be mapped into a single data set each. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Analog K Factors

Analog K Factors MVCU data types are mapped as follows. Scanning is addressing mode 1 – Sparse Space Addressing.

Analog K Factors **(*DeviceDataType* 7)**	
MVCU Data	**DeltaV Register**
Block 1, Sparse Space Pointer – Internal Address	R1
. .	. .
Block 6, Sparse Space Pointer – Internal Address	R6
Unavailable	R7
Block 1, K factor 1	R8 (floating point)
Block 1, K factor 2	R9
Block 1, K factor 3	R10
Block 1, K factor 4	R11
Block 1, K factor 5	R12
Block 1, K factor 6	R13
Block 1, Indirect Flags	R14 Flags are treated as an 8-bit uint, then floated. They can be extracted with a DeltaV BFO block.
. .	. .
Block 6, K factor 1	R43(floating point)
Block 6, K factor 2	R44
Block 6, K factor 3	R45
Block 6, K factor 4	R46
Block 6, K factor 5	R47
Block 6, K factor 6	R48
Block 6, Indirect Flags	R49 Flags are treated as an 8-bit uint, then floated. They

	can be extracted with a DeltaV BFO block.
Unavailable	R50 - R100

Analog K Factors are mapped to six floating point registers. This allows 6 blocks of Tuning Parameters to be mapped to a data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Analog Algorithm Data

Analog Algorithm MVCU data types are mapped as follows. Scanning is addressing mode 4 – Block Transfer - Sparse Space Addressing.

Analog Algorithm Data **(*DeviceDataType* 8)**	
MVCU Data	**DeltaV Register**
Function 1, Sparse Pointer – Internal Address	R1
. .	. .
Function 12, Sparse Pointer – Internal Address	R12
Function 1, Status and Alarms	R13 (uint 16) Status is mapped to bits 16-9. Alarms are mapped to bits 8-1
Function 1, Lock1 and Lock2	R14 Lock1 is mapped to bits 16-9 and Lock2 is mapped to bits 8-1
Function 1, Process Variable	R15
Function 1, Flag1 and Flag2	R16 Flag1 is mapped to bits 16-9 and Flag2 is mapped to bits 8-1
Function 1, Controller Output	R17
Function 1, Setpoint	R18
Function 1, Auxiliary Variable	R19
. .	. .
Function 12, Status and Alarms	R90 (uint 16) Status is mapped to bits 16-9. Alarms are mapped to bits 8-1
Function 12, Lock1 and Lock2	R91 Lock1 is mapped to bits 16-9 and Lock2 is mapped to bits 8-1
Function 12, Process Variable	R92
Function 12, Flag1 and Flag2	R93 Flag1 is mapped to bits 16-9 and Flag2 is mapped to bits 8-1
Function 12, Controller Output	R94
Function 12, Setpoint	R95
Function 12, Auxiliary Variable	R96
Unused	R97 – R 100

Analog Algorithms are mapped to seven 16 bit unsigned integer registers. This allows 12 blocks to be mapped to a data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Digital Algorithm Data

Digital Algorithm MVCU data types are mapped as follows. Scanning is addressing mode 4 – Block Transfer - Sparse Space Addressing.

Digital Algorithm Data **(*DeviceDataType* 9)**	
MVCU Data	**DeltaV Register**
Function 1, Sparse Pointer – Internal Address	R1
. .	. .
Function 12, Sparse Pointer – Internal Address	R12
Function 1, Data, Flag1	R13 (uint32) Unused bits 32-17. Data is mapped to bits 16-9. Flag1 is mapped to bits 8-1
Function 1, Current Value (Timers & Counters)	R14
Function 1, Set Point (Timers & Counters)	R15
. .	. .
Function 12, Data, Flag1	R46 (uint32) Unused bits 32-17. Data is mapped to bits 16-9. Flag1 is mapped to bits 8-1
Function 12, Current Value (Timers & Counters)	R47
Function 12, Set Point (Timers & Counters)	R48
Unused	R49 – R50
Unavailable	R51 – R100

Digital Algorithms are mapped to three 32 bit unsigned integer registers. This allows 12 blocks to be mapped to a data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Floating Point Node Data

Floating Point Node Data MVCU data types are mapped as follows. Scanning is addressing mode 1 – Sparse Space Addressing.

Floating Point Nodes **(*DeviceDataType* 10)**	
MVCU Data	**DeltaV Register**
Block 1, Sparse Space Pointer – Internal Address	R1
. .	. .
Block 12, Sparse Space Pointer – Internal Address	R12
Block 1, Flag1 & Flag2	R13 (floating point) Flag1 and Flag2 are treated as a 16 bit uint with Flag1 in the upper 8 bits, then floated.
Block 1, Output	R14 (floating point)
Block 1, Input	R15 (floating point)
. .	. .
Block 12, Flag1 & Flag2	R46 (floating point) Flag1 and Flag2 are treated as a 16 bit uint with Flag1 in the upper 8 bits, then floated.
Block 12, Output	R47 (floating point)
Block 12, Input	R48 (floating point)
Unavailable	R49 – R100

Floating point nodes are mapped to three floating point registers. This allows 12 nodes to be mapped to a data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Analog Algorithm Floating Point Data

Analog Algorithm Floating Point Data MVCU data types are mapped as follows. Scanning is addressing mode 1 – Sparse Space Addressing. For a Totalizer, the Totalizer's accumulator value will be returned. This is a read only data set.

Analog Algorithm Floating Point Data **(*DeviceDataType* 11)**	
MVCU Data	**DeltaV Register**
Block 1, Sparse Space Pointer – Internal Address	R1
. .	. .
Block 25, Sparse Space Pointer – Internal Address	R25
Block 1, Floating Data	R26 (floating point)
. .	. .
Block 25, Input	R50 (floating point)

Analog Algorithm Floating Point data is mapped to one floating point registers. This allows 25 the Analog Algorithm Floating Point data to be mapped to a data set. The data set must be defined as an OUTPUT data set mode type 1, with read back.

Appendix B - Module Data Availability

Module Data Availability

This document tabulates the data and functionality that is available in the DeltaV Module "templates" that are provided with the RMV/MVCU DeltaV interface. These module templates are used to create "loop details" and define the serial card scanning protocol. The following module templates are provided.

Analog Input Block	Digital Input Block	Floating Point Node
Analog Output Block	Digital Output Block	Ratio Station
Analog Node Block	Digital Node Block	Timer
PID Controller	Counter/Sequencer	Totalizer
Auto Manual Station	Manual Loader	Actuator
AI Block SQRT	AN Block SQRT	AO Block SQRT
FN Block SQRT	PID SQRT	A/M Station SQRT

Analog Input Block

The following variables, status flags, mode writes, and function blocks are available in the A_INPUT module.

Variable Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Scaler
Block Output	Output Mode Lock	Auto/Manual State	Output Scaler
	A/M Mode Lock		Input Alarm Block
	Out of Service		
	Bad Data		
	Over/Under Flow		

Analog Output Block

The following variables, status flags, mode writes, and function blocks are available in the A_OUTPUT module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Scaler
Block Output	Output Mode Lock	Auto/Manual State	Output Scaler
	A/M Mode Lock		Input Alarm Block
	Bad Data		
	Over/Under Flow		

Analog Node Block

The following variables, status flags, mode writes, and function blocks are available in the A_NODE module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Scaler
Block Output	Output Mode Lock	Auto/Manual State	Output Scaler
	A/M Mode Lock	Internal Address	Input Alarm Block
	Bad Data		
	Over/Under Flow		

Digital Input Block

The following variables, status flags, mode writes, and function blocks are available in the D_INPUT module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	
	A/M Mode Lock		
	Out of Service		
	Bad Data		

Digital Output Block

The following variables, status flags, mode writes, and function blocks are available in the D_OUTPUT module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	
	A/M Mode Lock		
	Bad Data		

Digital Node Block

The following variables, status flags, mode writes, and function blocks are available in the D_NODE module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	
	A/M Mode Lock		
	Bad Data		

PID Function

The following variables, status flags, mode writes, and function blocks are available in the PID module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Process Variable	Back Calculation Slave	Setpoint Value	PV Scaler
Setpoint	Back Calculation Active	Output Value	SP Scaler
Controller Output	Setpoint Source	Auto/Manual State	CO Scaler
Auxiliary Variable	Input 1 High Alarm	Console/Cascade	AUX Scaler
Internal Address	Input 1 Low Alarm	Address	PV Alarm Block
Address	Input 2 High Alarm	K1 – Gain	SP Alarm Block
	Input 2 Low Alarm	K2 – Reset	CO Alarm Block
	Output High Alarm	K3 – Rate	
	Output Low Alarm	K4 – Filter	
	Console/Cascade Mode Lock	K5 – Db or Gap	
	Setpoint Mode Lock	K6 – Bias	
	K1 Mode Lock		
	K2 Mode Lock		
	K3 Mode Lock		
	K4 Mode Lock		
	K5 Mode Lock		
	K6 Mode Lock		
	Auto/Manual State		
	Output Mode Lock		
	A/M Mode Lock		
	Out of Service		
	Bad Data		
	Under/Over Flow		
	Calculation Error		

RATIO Function

The following variables, status flags, mode writes, and function blocks are available in the RATIO module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Process Variable	Setpoint Source	Ratio Value	PV Scaler
Controller Output	Input 1 High Alarm	Output Value	CO Scaler
Auxiliary Variable	Input 1 Low Alarm	Auto/Manual State	AUX Scaler
Internal Address	Input 2 High Alarm	Console/Cascade	PV Alarm Block
Address	Input 2 Low Alarm	Address	CO Alarm Block
	Output High Alarm	K1 – Ratio	
	Output Low Alarm	K2 – Max Ratio	
	Console/Cascade Mode Lock	K3 – Min Ratio	
	Auto/Manual State	K4 – Gain	
	Output Mode Lock	K6 – Bias	
	A/M Mode Lock		
	Out of Service		
	Bad Data		
	Under/Over Flow		
	Calculation Error		

AUTO/MANUAL STATION Function

The following variables, status flags, mode writes, and function blocks are available in the AM_STN module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Process Variable	Back Calculation Slave	Output Value	PV Scaler
Controller Output	Back Calculation Active	Auto/Manual State	CO Scaler
Internal Address	Input 1 High Alarm	Address	PV Alarm Block
Address	Input 1 Low Alarm	K1 – Ramp Up Rate	CO Alarm Block
	Input 2 High Alarm	K2 – Ramp Down Rate	
	Input 2 Low Alarm	K6 – Bias	
	Output High Alarm		
	Output Low Alarm		
	K1 Mode Lock		
	K2 Mode Lock		
	K6 Mode Lock		
	Auto/Manual State		
	Output Mode Lock		
	A/M Mode Lock		
	Out of Service		
	Bad Data		
	Under/Over Flow		

MANUAL LOADER Function

The following variables, status flags, mode writes, and function blocks are available in the MAN_LDR module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Process Variable	Back Calculation Slave	Output Value	PV Scaler
Controller Output	Back Calculation Active	Auto/Manual State	CO Scaler
Internal Address	Input 1 High Alarm	Address	PV Alarm Block
Address	Input 1 Low Alarm		CO Alarm Block
	Output High Alarm		
	Output Low Alarm		
	Auto/Manual State		
	Output Mode Lock		
	A/M Mode Lock		
	Out of Service		
	Bad Data		
	Under/Over Flow		

TOTALIZER Function

The following variables, status flags, mode writes, and function blocks are available in the TOTALIZER module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Process Variable	Back Calculation Slave	K1-Counts	PV Scaler
Controller Output	Back Calculation Active	Output Value	CO Scaler
Auxiliary Variable	Input 1 High Alarm	Auto/Manual State	AUX Scaler
Internal Address	Input 1 Low Alarm	Address	PV Alarm Block
Address	Output High Alarm		CO Alarm Block
	Output Low Alarm		
	K1 Mode Lock		
	Auto/Manual State		
	Output Mode Lock		
	A/M Mode Lock		
	Out of Service		
	Bad Data		
	Under/Over Flow		
	Calculation Error		

TIMER

The following variables, status flags, mode writes, and function blocks are available in the TIMER module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	Multiply
Current Value	A/M Mode Lock	Timer Setpoint	
Setpoint Value	Out of Service	Address	
Internal Address	Bad Data		
Address			

COUNTER

The following variables, status flags, mode writes, and function blocks are available in the COUNTER module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	
Current Value	A/M Mode Lock	Counter Setpoint	
Setpoint Value	Out of Service	Address	
Internal Address	Bad Data		
Address			

ACTUATOR

The following variables, status flags, mode writes, and function blocks are available in the ACTUATOR module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	
Block Output	Output Mode Lock	Auto/Manual State	
Internal Address	A/M Mode Lock	Address	
Address	Out of Service		
	Bad Data		
	Auto Input 1		
	Auto Input 2		
	Confirm Open		
	Confirm Close		
	Alarm Output		
	Output 1		
	Output 2		
	Output 3		

Floating Point Node Block

The following variables, status flags, mode writes, and function blocks are available in the F_NODE module.

Variables Reads	Status Flags	Variable and Mode Writes	Function Blocks
Block Input	Auto Manual	Block Output Value	Input Alarm Block
Block Output	Output Mode Lock	Auto/Manual State	
Internal Address	A/M Mode Lock	Address	
Address	Out of Service		
	Bad Data		
	Over/Under Flow		

Appendix C – Dataset Planning Forms

DeltaV Serial Card Data Set Configuration

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	P01
Device Number	DEV1
MVCU Tagname	MVCU-1

PORT 1 - Data Sets	Description	Data Type	Special Data 1	Special Data 2
1	MVCU Status	MVCU Status - 0	1	0
2	Analog Inputs 1-32	Analog Input - 1	0	1
3	Digital Inputs 1- 64	Digital Inputs - 4	0	1
4	Digital Nodes 1-100	Digital Nodes - 6	0	1
5	Analog Algorithms	ACLs - 8	4	0
6	Floating Point Nodes	F Nodes - 10	1	0
7	Analog K Factors	K Factors - 7	3	0
8				
9				
10				
11				
12				
13				
14				
15				
16				

PORT 2 - Data Sets	Description	Data Type		
1	Analog Inputs 33-64	Analog Input - 1	0	33
2	Analog Outputs 1-32	Analog Outputs - 2	0	1
3	Digital Outputs 1-64	Digital Outputs - 5	0	1
4	Analog Nodes 1-32	Analog Nodes - 3	1	0
5	Digital Algorithms	DCLs - 9	4	0
6	Analog K Factors	K Factors - 7	3	0
7	Analog Algorithms - Float data	AA Floating Point - 11	1	0
8				
9				
10				
11				
12				
13				
14				
15				
16				

MVCU Status Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	P01
Device Number	
Data Set Number	1
MVCU Tagname	

Register Description	Register
Side-Side Checksum Mismatch	R1
Side A Status and Master Processor Status	R2
Reserved	R3
Side A hardware errors	R4
Side A hardware errors and Card Ack Failure	R5
Side A Checksum Errors	R6
Not Used	R7
Side B Status and Master Processor Status	R8
Reserved	R9
Side B hardware errors	R10
Side B hardware errors and Card Ack Failure	R11
Side B Checksum Errors	R12
Not Used	R13
Year (00-99) - Read	R14
Month (01-12) - Read	R15
Day of month (01-31) - Read	R16
Hour (00-23) - Read	R17
Minute (00-59) - Read	R18
Second (00-59) - Read	R19
Reset Statistics	R20
Comm Diagnostic 02 - Invalid Address	R21
Comm Diagnostic 03 - Invalid Count	R22
Comm Diagnostic 04 - Invalid Block	R23
Comm Diagnostic 05 - Invalid Data	R24
Comm Diagnostic 06 - Invalid Mode	R25
Comm Diagnostic 07 - Mode Transfer Lock	R26
Comm Port 1 Failure	R27
Comm Port 2 Failure	R28
Reset MVCU Diagnostics	R29
Download Date and Time	R30

MVCU Status Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	P01
Device Number	
Data Set Number	1
MVCU Tagname	

Description	Register
DataSet 1 - Error Accumulator	R31
DataSet 2 - Error Accumulator	R32
DataSet 3 - Error Accumulator	R33
DataSet 4 - Error Accumulator	R34
DataSet 5 - Error Accumulator	R35
DataSet 6 - Error Accumulator	R36
DataSet 7 - Error Accumulator	R37
DataSet 8 - Error Accumulator	R38
DataSet 9 - Error Accumulator	R39
DataSet 10 - Error Accumulator	R40
DataSet 11 - Error Accumulator	R41
DataSet 12 - Error Accumulator	R42
DataSet 13 - Error Accumulator	R43
DataSet 14 - Error Accumulator	R44
DataSet 15 - Error Accumulator	R45
DataSet 16 - Error Accumulator	R46

Description	Register
DataSet 17 - Error Accumulator	R47
DataSet 18 - Error Accumulator	R48
DataSet 19 - Error Accumulator	R49
DataSet 20 - Error Accumulator	R50
DataSet 21 - Error Accumulator	R51
DataSet 22 - Error Accumulator	R52
DataSet 23 - Error Accumulator	R53
DataSet 24 - Error Accumulator	R54
DataSet 25 - Error Accumulator	R55
DataSet 26 - Error Accumulator	R56
DataSet 27 - Error Accumulator	R57
DataSet 28 - Error Accumulator	R58
DataSet 29 - Error Accumulator	R59
DataSet 30 - Error Accumulator	R60
DataSet 31 - Error Accumulator	R61
DataSet 32 - Error Accumulator	R62

Year (00-99) - Write	R63
Month (01-12) - Write	R64
Day of month (01-31) - Write	R65
Hour (00-23) - Write	R66
Minute (00-59) - Write	R67
Second (00-59) - Write	R68
Scan Time Port1 (1/100 of sec.)	R69
Scan Time Port2 (1/100 of sec.)	R70
Status Messages Port 1	R71
Status Messages Port 2	R72
Line Errors Port 1	R73
Line Errors Port 2	R74

Analog Input Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R1
	1	Block 1	Block Output	R2
			Block Input	R3

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R4
	2	Block 2	Block Output	R5
			Block Input	R6

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R7
	3	Block 3	Block Output	R8
			Block Input	R9

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R10
	4	Block 4	Block Output	R11
			Block Input	R12

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R13
	5	Block 5	Block Output	R14
			Block Input	R15

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R16
	6	Block 6	Block Output	R17
			Block Input	R18

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R19
	7	Block 7	Block Output	R20
			Block Input	R21

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R22
	8	Block 8	Block Output	R23
			Block Input	R24

Analog Input Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R25
	9	Block 9	Block Output	R26
			Block Input	R27

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R28
	10	Block 10	Block Output	R29
			Block Input	R30

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R31
	11	Block 11	Block Output	R32
			Block Input	R33

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R34
	12	Block 12	Block Output	R35
			Block Input	R36
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R37
	13	Block 13	Block Output	R38
			Block Input	R39

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R40
	14	Block 14	Block Output	R41
			Block Input	R42

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R43
	15	Block 15	Block Output	R44
			Block Input	R45

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R46
	16	Block 16	Block Output	R47
			Block Input	R48

Analog Input Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R49
	17	Block 17	Block Output	R50
			Block Input	R51

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R52
	18	Block 18	Block Output	R53
			Block Input	R54

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R55
	19	Block 19	Block Output	R56
			Block Input	R57

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R58
	20	Block 20	Block Output	R59
			Block Input	R60

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R61
	21	Block 21	Block Output	R62
			Block Input	R63

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R64
	22	Block 22	Block Output	R65
			Block Input	R66

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R67
	23	Block 23	Block Output	R68
			Block Input	R69

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R70
	24	Block 24	Block Output	R71
			Block Input	R72

Analog Input Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R73
	25	Block 25	Block Output	R74
			Block Input	R75

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R76
	26	Block 26	Block Output	R77
			Block Input	R78

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R79
	27	Block 27	Block Output	R80
			Block Input	R81

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R82
	28	Block 28	Block Output	R83
			Block Input	R84

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R85
	29	Block 29	Block Output	R86
			Block Input	R87

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R88
	30	Block 30	Block Output	R89
			Block Input	R90

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R91
	31	Block 31	Block Output	R92
			Block Input	R93

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R94
	32	Block 32	Block Output	R95
			Block Input	R96

Analog Output Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R1
	1	Block 1	Block Output	R2
			Block Input	R3

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R4
	2	Block 2	Block Output	R5
			Block Input	R6

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R7
	3	Block 3	Block Output	R8
			Block Input	R9

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R10
	4	Block 4	Block Output	R11
			Block Input	R12

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R13
	5	Block 5	Block Output	R14
			Block Input	R15

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R16
	6	Block 6	Block Output	R17
			Block Input	R18

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R19
	7	Block 7	Block Output	R20
			Block Input	R21

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R22
	8	Block 8	Block Output	R23
			Block Input	R24

Analog Output Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R25
	9	Block 9	Block Output	R26
			Block Input	R27

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R28
	10	Block 10	Block Output	R29
			Block Input	R30

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R31
	11	Block 11	Block Output	R32
			Block Input	R33

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R34
	12	Block 12	Block Output	R35
			Block Input	R36
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R37
	13	Block 13	Block Output	R38
			Block Input	R39

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R40
	14	Block 14	Block Output	R41
			Block Input	R42

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R43
	15	Block 15	Block Output	R44
			Block Input	R45

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R46
	16	Block 16	Block Output	R47
			Block Input	R48

Analog Output Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R49
	17	Block 17	Block Output	R50
			Block Input	R51

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R52
	18	Block 18	Block Output	R53
			Block Input	R54

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R55
	19	Block 19	Block Output	R56
			Block Input	R57

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R58
	20	Block 20	Block Output	R59
			Block Input	R60

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R61
	21	Block 21	Block Output	R62
			Block Input	R63

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R64
	22	Block 22	Block Output	R65
			Block Input	R66

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R67
	23	Block 23	Block Output	R68
			Block Input	R69

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R70
	24	Block 24	Block Output	R71
			Block Input	R72

Analog Output Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R73
	25	Block 25	Block Output	R74
			Block Input	R75

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R76
	26	Block 26	Block Output	R77
			Block Input	R78

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R79
	27	Block 27	Block Output	R80
			Block Input	R81

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R82
	28	Block 28	Block Output	R83
			Block Input	R84

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R85
	29	Block 29	Block Output	R86
			Block Input	R87

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R88
	30	Block 30	Block Output	R89
			Block Input	R90

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R91
	31	Block 31	Block Output	R92
			Block Input	R93

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R94
	32	Block 32	Block Output	R95
			Block Input	R96

Analog Node Block Data Set - Contiguous Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R1
	1	Block 1	Block Output	R2
			Block Input	R3

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R4
	2	Block 2	Block Output	R5
			Block Input	R6

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R7
	3	Block 3	Block Output	R8
			Block Input	R9

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R10
	4	Block 4	Block Output	R11
			Block Input	R12

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R13
	5	Block 5	Block Output	R14
			Block Input	R15

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R16
	6	Block 6	Block Output	R17
			Block Input	R18

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R19
	7	Block 7	Block Output	R20
			Block Input	R21

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R22
	8	Block 8	Block Output	R23
			Block Input	R24

Analog Node Block Data Set - Contiguous Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R25
	9	Block 9	Block Output	R26
			Block Input	R27

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R28
	10	Block 10	Block Output	R29
			Block Input	R30

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R31
	11	Block 11	Block Output	R32
			Block Input	R33

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R34
	12	Block 12	Block Output	R35
			Block Input	R36
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R37
	13	Block 13	Block Output	R38
			Block Input	R39

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R40
	14	Block 14	Block Output	R41
			Block Input	R42

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R43
	15	Block 15	Block Output	R44
			Block Input	R45

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R46
	16	Block 16	Block Output	R47
			Block Input	R48

Analog Node Block Data Set - Contiguous Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R49
	17	Block 17	Block Output	R50
			Block Input	R51

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R52
	18	Block 18	Block Output	R53
			Block Input	R54

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R55
	19	Block 19	Block Output	R56
			Block Input	R57

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R58
	20	Block 20	Block Output	R59
			Block Input	R60

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R61
	21	Block 21	Block Output	R62
			Block Input	R63

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R64
	22	Block 22	Block Output	R65
			Block Input	R66

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R67
	23	Block 23	Block Output	R68
			Block Input	R69

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R70
	24	Block 24	Block Output	R71
			Block Input	R72

Analog Node Block Data Set - Contiguous Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Dynamic Data		**Register**
Tagname (Module Name)	Internal Address	Block Number	Block Flags	R73
	25	Block 25	Block Output	R74
			Block Input	R75

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R76
	26	Block 26	Block Output	R77
			Block Input	R78

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R79
	27	Block 27	Block Output	R80
			Block Input	R81

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R82
	28	Block 28	Block Output	R83
			Block Input	R84

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R85
	29	Block 29	Block Output	R86
			Block Input	R87

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R88
	30	Block 30	Block Output	R89
			Block Input	R90

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R91
	31	Block 31	Block Output	R92
			Block Input	R93

Tagname (Module Name)	Internal Address	Block Number	Block Flags	R94
	32	Block 32	Block Output	R95
			Block Input	R96

Analog Node Block Data Set - Sparse Space Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers	
Tagname (Module Name)	Internal Address	Block Number	Register
		Block 1	R1

Dynamic Data Area	
Block Flags	R26
Block Output	R27
Block Input	R28

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 2	R2

Block Flags	R29
Block Output	R30
Block Input	R31

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 3	R3

Block Flags	R32
Block Output	R33
Block Input	R34

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 4	R4

Block Flags	R35
Block Output	R36
Block Input	R37

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 5	R5

Block Flags	R38
Block Output	R39
Block Input	R40

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 6	R6

Block Flags	R41
Block Output	R42
Block Input	R43

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 7	R7

Block Flags	R44
Block Output	R45
Block Input	R46

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 8	R8

Block Flags	R47
Block Output	R48
Block Input	R49

Analog Node Block Data Set - Sparse Space Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers	
Tagname (Module Name)	Internal Address	Block Number	Register
		Block 9	R9

Dynamic Data Area	
Block Flags	R50
Block Output	R51
Block Input	R52

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 10	R10

Block Flags	R53
Block Output	R54
Block Input	R55

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 11	R11

Block Flags	R56
Block Output	R57
Block Input	R58

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 12	R12

Block Flags	R59
Block Output	R60
Block Input	R61

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 13	R13

Block Flags	R62
Block Output	R63
Block Input	R64

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 14	R14

Block Flags	R65
Block Output	R66
Block Input	R67

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 15	R15

Block Flags	R68
Block Output	R69
Block Input	R70

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 16	R16

Block Flags	R71
Block Output	R72
Block Input	R73

Analog Node Block Data Set - Sparse Space Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R74
		Block 17	R17	Block Output	R75
				Block Input	R76
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R77
		Block 18	R18	Block Output	R78
				Block Input	R79
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R80
		Block 19	R19	Block Output	R81
				Block Input	R82
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R83
		Block 20	R20	Block Output	R84
				Block Input	R85
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R86
		Block 21	R21	Block Output	R87
				Block Input	R88
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R89
		Block 22	R22	Block Output	R90
				Block Input	R91
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R92
		Block 23	R23	Block Output	R93
				Block Input	R94
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R95
		Block 24	R24	Block Output	R96
				Block Input	R97
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R98
		Block 25	R25	Block Output	R99
				Block Input	R100

Digital Input Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

Tagname (Module Name)	Internal Address	Block Number	Register
	1	Block 1	R1
	2	Block 2	R2
	3	Block 3	R3
	4	Block 4	R4
	5	Block 5	R5
	6	Block 6	R6
	7	Block 7	R7
	8	Block 8	R8
	9	Block 9	R9
	10	Block 10	R10
	11	Block 11	R11
	12	Block 12	R12
	13	Block 13	R13
	14	Block 14	R14
	15	Block 15	R15
	16	Block 16	R16
	17	Block 17	R17
	18	Block 18	R18
	19	Block 19	R19
	20	Block 20	R20
	21	Block 21	R21
	22	Block 22	R22
	23	Block 23	R23
	24	Block 24	R24
	25	Block 25	R25
	26	Block 26	R26
	27	Block 27	R27
	28	Block 28	R28
	29	Block 29	R29
	30	Block 30	R30
	31	Block 31	R31
	32	Block 32	R32

Tagname (Module Name)	Internal Address	Block Number	Register
	33	Block 33	R33
	34	Block 34	R34
	35	Block 35	R35
	36	Block 36	R36
	37	Block 37	R37
	38	Block 38	R38
	39	Block 39	R39
	40	Block 40	R40
	41	Block 41	R41
	42	Block 42	R42
	43	Block 43	R43
	44	Block 44	R44
	45	Block 45	R45
	46	Block 46	R46
	47	Block 47	R47
	48	Block 48	R48
	49	Block 49	R49
	50	Block 50	R50
	51	Block 51	R51
	52	Block 52	R52
	53	Block 53	R53
	54	Block 54	R54
	55	Block 55	R55
	56	Block 56	R56
	57	Block 57	R57
	58	Block 58	R58
	59	Block 59	R59
	60	Block 60	R60
	61	Block 61	R61
	62	Block 62	R62
	63	Block 63	R63
	64	Block 64	R64

Digital Output Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

Tagname (Module Name)	Internal Address	Block Number	Register
	1	Block 1	R1
	2	Block 2	R2
	3	Block 3	R3
	4	Block 4	R4
	5	Block 5	R5
	6	Block 6	R6
	7	Block 7	R7
	8	Block 8	R8
	9	Block 9	R9
	10	Block 10	R10
	11	Block 11	R11
	12	Block 12	R12
	13	Block 13	R13
	14	Block 14	R14
	15	Block 15	R15
	16	Block 16	R16
	17	Block 17	R17
	18	Block 18	R18
	19	Block 19	R19
	20	Block 20	R20
	21	Block 21	R21
	22	Block 22	R22
	23	Block 23	R23
	24	Block 24	R24
	25	Block 25	R25
	26	Block 26	R26
	27	Block 27	R27
	28	Block 28	R28
	29	Block 29	R29
	30	Block 30	R30
	31	Block 31	R31
	32	Block 32	R32

Tagname (Module Name)	Internal Address	Block Number	Register
	33	Block 33	R33
	34	Block 34	R34
	35	Block 35	R35
	36	Block 36	R36
	37	Block 37	R37
	38	Block 38	R38
	39	Block 39	R39
	40	Block 40	R40
	41	Block 41	R41
	42	Block 42	R42
	43	Block 43	R43
	44	Block 44	R44
	45	Block 45	R45
	46	Block 46	R46
	47	Block 47	R47
	48	Block 48	R48
	49	Block 49	R49
	50	Block 50	R50
	51	Block 51	R51
	52	Block 52	R52
	53	Block 53	R53
	54	Block 54	R54
	55	Block 55	R55
	56	Block 56	R56
	57	Block 57	R57
	58	Block 58	R58
	59	Block 59	R59
	60	Block 60	R60
	61	Block 61	R61
	62	Block 62	R62
	63	Block 63	R63
	64	Block 64	R64

Digital Node Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

Tagname (Module Name)	Internal Address	Block Number	Register
	1	Block 1	R1
	2	Block 2	R2
	3	Block 3	R3
	4	Block 4	R4
	5	Block 5	R5
	6	Block 6	R6
	7	Block 7	R7
	8	Block 8	R8
	9	Block 9	R9
	10	Block 10	R10
	11	Block 11	R11
	12	Block 12	R12
	13	Block 13	R13
	14	Block 14	R14
	15	Block 15	R15
	16	Block 16	R16
	17	Block 17	R17
	18	Block 18	R18
	19	Block 19	R19
	20	Block 20	R20
	21	Block 21	R21
	22	Block 22	R22
	23	Block 23	R23
	24	Block 24	R24
	25	Block 25	R25

Tagname (Module Name)	Internal Address	Block Number	Register
	26	Block 26	R26
	27	Block 27	R27
	28	Block 28	R28
	29	Block 29	R29
	30	Block 30	R30
	31	Block 31	R31
	32	Block 32	R32
	33	Block 33	R33
	34	Block 34	R34
	35	Block 35	R35
	36	Block 36	R36
	37	Block 37	R37
	38	Block 38	R38
	39	Block 39	R39
	40	Block 40	R40
	41	Block 41	R41
	42	Block 42	R42
	43	Block 43	R43
	44	Block 44	R44
	45	Block 45	R45
	46	Block 46	R46
	47	Block 47	R47
	48	Block 48	R48
	49	Block 49	R49
	50	Block 50	R50

Digital Node Block Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

Tagname (Module Name)	Internal Address	Block Number	Register
	51	Block 51	R51
	52	Block 52	R52
	53	Block 53	R53
	54	Block 54	R54
	55	Block 55	R55
	56	Block 56	R56
	57	Block 57	R57
	58	Block 58	R58
	59	Block 59	R59
	60	Block 60	R60
	61	Block 61	R61
	62	Block 62	R62
	63	Block 63	R63
	64	Block 64	R64
	65	Block 65	R65
	66	Block 66	R66
	67	Block 67	R67
	68	Block 68	R68
	69	Block 69	R69
	70	Block 70	R70
	71	Block 71	R71
	72	Block 72	R72
	73	Block 73	R73
	74	Block 74	R74
	75	Block 75	R75

Tagname (Module Name)	Internal Address	Block Number	Register
	76	Block 76	R76
	77	Block 77	R77
	78	Block 78	R78
	79	Block 79	R79
	80	Block 80	R80
	81	Block 81	R81
	82	Block 82	R82
	83	Block 83	R83
	84	Block 84	R84
	85	Block 85	R85
	86	Block 86	R86
	87	Block 87	R87
	88	Block 88	R88
	89	Block 89	R89
	90	Block 90	R90
	91	Block 91	R91
	92	Block 92	R92
	93	Block 93	R93
	94	Block 94	R94
	95	Block 95	R95
	96	Block 96	R96
	97	Block 97	R97
	98	Block 98	R98
	99	Block 99	R99
	100	Block 100	R100

Analog K Factor Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R8
		Block 1	R1	K2 Factor	R9
				K3 Factor	R10
				K4 Factor	R11
				K5 Factor	R12
				K6 Factor	R13
				Status Flags	R14

Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R15
		Block 2	R2	K2 Factor	R16
				K3 Factor	R17
				K4 Factor	R18
				K5 Factor	R19
				K6 Factor	R20
				Status Flags	R21

Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R22
		Block 3	R3	K2 Factor	R23
				K3 Factor	R24
				K4 Factor	R25
				K5 Factor	R26
				K6 Factor	R27
				Status Flags	R28

Analog K Factor Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R29
		Block 4	R4	K2 Factor	R30
				K3 Factor	R31
				K4 Factor	R32
				K5 Factor	R33
				K6 Factor	R34
				Status Flags	R35

Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R36
		Block 5	R5	K2 Factor	R37
				K3 Factor	R38
				K4 Factor	R39
				K5 Factor	R40
				K6 Factor	R41
				Status Flags	R42

Tagname (Module Name)	Internal Address	Block Number	Register	K1 Factor	R43
		Block 6	R6	K2 Factor	R44
				K3 Factor	R45
				K4 Factor	R46
				K5 Factor	R47
				K6 Factor	R48
				Status Flags	R49

Analog Algorithm Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers	
Tagname (Module Name)	Internal Address	Block Number	Register
		Block 1	R1

Dynamic Data Area	
Function Flags 1	R13
Function Flags 2	R14
Process Variable	R15
Function Flags 3	R16
Controller Output	R17
Setpoint	R18
Auxiliary Variable	R19

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 2	R2

Function Flags 1	R20
Function Flags 2	R21
Process Variable	R22
Function Flags 3	R23
Controller Output	R24
Setpoint	R25
Auxiliary Variable	R26

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 3	R3

Function Flags 1	R27
Function Flags 2	R28
Process Variable	R29
Function Flags 3	R30
Controller Output	R31
Setpoint	R32
Auxiliary Variable	R33

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 4	R4

Function Flags 1	R34
Function Flags 2	R35
Process Variable	R36
Function Flags 3	R37
Controller Output	R38
Setpoint	R39
Auxiliary Variable	R40

Analog Algorithm Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers	
Tagname (Module Name)	Internal Address	Block Number	Register
		Block 5	R5

Dynamic Data Area	
Function Flags 1	R41
Function Flags 2	R42
Process Variable	R43
Function Flags 3	R44
Controller Output	R45
Setpoint	R46
Auxiliary Variable	R47

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 6	R6

Function Flags 1	R48
Function Flags 2	R49
Process Variable	R50
Function Flags 3	R51
Controller Output	R52
Setpoint	R53
Auxiliary Variable	R54

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 7	R7

Function Flags 1	R55
Function Flags 2	R56
Process Variable	R57
Function Flags 3	R58
Controller Output	R59
Setpoint	R60
Auxiliary Variable	R61

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 8	R8

Function Flags 1	R62
Function Flags 2	R63
Process Variable	R64
Function Flags 3	R65
Controller Output	R66
Setpoint	R67
Auxiliary Variable	R68

Analog Algorithm Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers	
Tagname (Module Name)	Internal Address	Block Number	Register
		Block 9	R9

Dynamic Data Area	
Function Flags 1	R69
Function Flags 2	R70
Process Variable	R71
Function Flags 3	R72
Controller Output	R73
Setpoint	R74
Auxiliary Variable	R75

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 10	R10

Function Flags 1	R76
Function Flags 2	R77
Process Variable	R78
Function Flags 3	R79
Controller Output	R80
Setpoint	R81
Auxiliary Variable	R82

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 11	R11

Function Flags 1	R83
Function Flags 2	R84
Process Variable	R85
Function Flags 3	R86
Controller Output	R87
Setpoint	R88
Auxiliary Variable	R89

Tagname (Module Name)	Internal Address	Block Number	Register
		Block 12	R12

Function Flags 1	R90
Function Flags 2	R91
Process Variable	R92
Function Flags 3	R93
Controller Output	R94
Setpoint	R95
Auxiliary Variable	R96

Digital Algorithm Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R13
		Block 1	R1	Current Value	R14
				Setpoint	R15
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R16
		Block 2	R2	Current Value	R17
				Setpoint	R18
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R19
		Block 3	R3	Current Value	R20
				Setpoint	R21
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R22
		Block 4	R4	Current Value	R23
				Setpoint	R24
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R25
		Block 5	R5	Current Value	R26
				Setpoint	R27
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R28
		Block 6	R6	Current Value	R29
				Setpoint	R30
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R31
		Block 7	R7	Current Value	R32
				Setpoint	R33
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R34
		Block 8	R8	Current Value	R35
				Setpoint	R36

Digital Algorithm Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R37
		Block 9	R9	Current Value	R38
				Setpoint	R39

Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R40
		Block 10	R10	Current Value	R41
				Setpoint	R42

Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R43
		Block 11	R11	Current Value	R44
				Setpoint	R45

Tagname (Module Name)	Internal Address	Block Number	Register	Digital Flags	R46
		Block 12	R12	Current Value	R47
				Setpoint	R48

Floating Point Node Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 1	R1	Block Flags	R13
				Block Output	R14
				Block Input	R15
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 2	R2	Block Flags	R16
				Block Output	R17
				Block Input	R18
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 3	R3	Block Flags	R19
				Block Output	R20
				Block Input	R21
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 4	R4	Block Flags	R22
				Block Output	R23
				Block Input	R24
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 5	R5	Block Flags	R25
				Block Output	R26
				Block Input	R27
Tagname (Module Name)	Internal Address	Block Number	Register		
		Block 6	R6	Block Flags	R28
				Block Output	R29
				Block Input	R30

Floating Point Node Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

		Sparse Space Pointers		Dynamic Data Area	
Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R31
		Block 7	R7	Block Output	R32
				Block Input	R33

Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R34
		Block 8	R8	Block Output	R35
				Block Input	R36

Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R37
		Block 9	R9	Block Output	R38
				Block Input	R39

Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R40
		Block 10	R10	Block Output	R41
				Block Input	R42

Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R43
		Block 11	R11	Block Output	R44
				Block Input	R45

Tagname (Module Name)	Internal Address	Block Number	Register	Block Flags	R46
		Block 12	R12	Block Output	R47
				Block Input	R48

Analog Algorithm Floating Point Data Set Addressing

Controller Name	CTLR1
I/O Port	IO1
Card Slot Number	C01
Serial Port Number	
Device Number	DEV1
Data Set Number	

Tagname		**Sparse Space Pointers**	
(Module Name)	Internal Address	Block Number	Register
		Function 1	R1
		Function 2	R2
		Function 3	R3
		Function 4	R4
		Function 5	R5
		Function 6	R6
		Function 7	R7
		Function 8	R8
		Function 9	R9
		Function 10	R10
		Function 11	R11
		Function 12	R12
		Function 13	R13
		Function 14	R14
		Function 15	R15
		Function 16	R16
		Function 17	R17
		Function 18	R18
		Function 19	R19
		Function 20	R20
		Function 21	R21
		Function 22	R22
		Function 23	R23
		Function 24	R24
		Function 25	R25

Dynamic Data Area	
Floating Point Value	R26
Floating Point Value	R27
Floating Point Value	R28
Floating Point Value	R29
Floating Point Value	R30
Floating Point Value	R31
Floating Point Value	R32
Floating Point Value	R33
Floating Point Value	R34
Floating Point Value	R35
Floating Point Value	R36
Floating Point Value	R37
Floating Point Value	R38
Floating Point Value	R39
Floating Point Value	R40
Floating Point Value	R41
Floating Point Value	R42
Floating Point Value	R43
Floating Point Value	R44
Floating Point Value	R45
Floating Point Value	R46
Floating Point Value	R47
Floating Point Value	R48
Floating Point Value	R49
Floating Point Value	R50

Appendix D - Host Computer Access

Introduction

One of the elements required by the RMV/DeltaV interface is the support of “Advanced Process Control” (APC) applications. Within the RMV system these applications resided on DEC VAX or Alpha computers and exchanged data with the CCM through RMT/host. In the new architecture where the DeltaV replaces the CCM and RMT/host, a new mechanism is needed to support APC applications.

DeltaV, and most current APC applications, incorporate a new technology named OPC that will allow network exchange of variables and text strings between client/server platforms. Using OPC, APC applications may establish a connection with a DeltaV system for bi-directional data exchange. The RMV/DeltaV interface has been enhanced to support APC applications, through OPC, that will allow APC reads and writes to both DeltaV parameters and several MVCU attributes.

This release of the RMV/DeltaV interface includes the following features in support of APC applications:

- Enhanced control modules that “Host” OPC (APC) parameters
- Module transfer block to switch between “console” and “host’ values
- Module bumpless transfer logic
- Faceplate and Detail Display support of “Host Access”
- Enhanced Format Spec. Files to support “Host Access”

The intent of this section of the RMV/DeltaV interface module is to describe the “Host Access” implementation and tabulate the variables, or parameters, that may be exchanged between DeltaV and APC applications.

Host Read Access

The Host application, through OPC, has read access to any parameter contained within any control module. Read access will not be controlled by the RMV DeltaV interface. All data types are supported by the DeltaV interface as listed below.

Analog Item Types
Analog Input Block
Analog Output Block
Analog Node Block
Floating Point Node Block
PID
Ratio Station
Auto/Manual Station
Manual Loader
Totalizer

Digital Item Types
Digital Input Block
Digital Node Block
Digital Output Block
Dual In
Timer
Counter
Actuator

Read Requirements

The following module parameters are samples of variables that may be read by the Host Computer. The OPC links are DeltaV module parameters. A complete tabulation of parameter and variable read references is contained later in this Appendix. Note that the module name (Tagname) precedes all of the parameter references contained in these tables.

AI/AO
Process Variable OPC Link = AI1/OUT.CV
Output OPC Link = OUTPUT_SCALE/OUT.CV
Current Mode OPC Link = AUTO_MANUAL.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm state, alarm limits, and control limits.

AN/FN
Process Variable OPC Link = AI1/OUT.CV
Output OPC Link = OUTPUT_SCALE/OUT.CV
Setpoint OPC Link = SP_LIMIT/OUT.CV
Current Mode OPC Link = AUTO_MANUAL.CV
Computer SP Mode OPC Link = COMP_SP_ENAB.CV
Computer CO Mode OPC Link = COMP_CO_ENAB.CV

Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm state, alarm limits, and control limits.

PID
Process Variable OPC Link = AI1/OUT.CV
Setpoint OPC Link = SP_INPUT_SCALE/OUT.CV
Output OPC Link = CO_INPUT_SCALE/OUT.CV
Aux Var OPC Link = AV_SCALE/OUT.CV
Current SP Mode OPC Link = SP_SOURCE.CV
Current CO Mode OPC Link = AUTO_MANUAL.CV
Computer SP Mode OPC Link = COMP_SP_ENAB.CV
Computer CO Mode OPC Link = COMP_CO_ENAB.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm state, alarm limits, and control limits.

Ratio and Totalizer
Process Variable OPC Link = AI1/OUT.CV
Output OPC Link = CO_INPUT_SCALE/OUT.CV
Aux Var OPC Link = AV_SCALE/OUT.CV
Current CO Mode OPC Link = AUTO_MANUAL.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm state, alarm limits, and control limits.

Auto/Manual Station and Manual Loader
Process Variable OPC Link = AI1/OUT.CV
Output OPC Link = CO_INPUT_SCALE/OUT.CV
Aux Var OPC Link = AV_SCALE/OUT.CV
Current CO Mode OPC Link = AUTO_MANUAL.CV
Computer CO Mode OPC Link = COMP_CO_ENAB.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm state, alarm limits, and control limits.

DI/DO
Input OPC Link = BLOCK_INPUT.CV
Output OPC Link = BLOCK_OUTPUT.CV
Current Mode OPC Link = AUTO_MANUAL.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm status.

DN
Input OPC Link = BLOCK_INPUT.CV
Output OPC Link = BLOCK_OUTPUT.CV
Current Mode OPC Link = AUTO_MANUAL.CV
Computer CO Mode OPC Link = COMP_CO_ENAB.CV
Host Write Enabled OPC Link = HOST_WRITE.CV
Other DeltaV Parameters such as alarm status.

Dual Digital Input

Input 1	OPC Link = BLOCK_OUTPUT.CV
Input 2	OPC Link = BLOCK_OUTPUT1.CV
Host Write Enabled	OPC Link = HOST_WRITE.CV

Other DeltaV Parameters such as alarm status.

Timer

Current Value	OPC Link = T_VALUE/OUT.CV
Setpoint	OPC Link = T_SETPOINT/OUT.CV
Output	OPC Link = BLOCK_OUTPUT.CV
Current Mode	OPC Link = AUTO_MANUAL.CV
Host Write Enabled	OPC Link = HOST_WRITE.CV

Other DeltaV Parameters such as alarm status.

Counter

Current Value	OPC Link = C_VALUE/OUT.CV
Setpoint	OPC Link = C_SETPOINT/OUT.CV
Output	OPC Link = BLOCK_OUTPUT.CV
Current Mode	OPC Link = AUTO_MANUAL.CV
Host Write Enabled	OPC Link = HOST_WRITE.CV

Other DeltaV Parameters such as alarm status.

Actuator

Input 1	OPC Link = AUTO_INPUT_1.CV
Input 2	OPC Link = AUTO_INPUT_2.CV
Input 3	OPC Link = CONFIRM_OPEN.CV
Input 4	OPC Link = CONFIRM_CLOSED.CV
Output 1	OPC Link = OUTPUT_1.CV
Output 2	OPC Link = OUTPUT_2.CV
Output 3	OPC Link = OUTPUT_3.CV
Output 4	OPC Link = ALARM_OUTPUT.CV
Current Mode	OPC Link = AUTO_MANUAL.CV
Host Write Enabled	OPC Link = HOST_WRITE.CV

Other DeltaV Parameters such as alarm status.

Host Write Access

Variables and parameters that may be written from the Host to DeltaV control modules fall into two categories. These are:

- Parameters that are only written to control modules
- Variables and Modes that are written to control modules and also to the MVCU

Parameters written to Modules

Parameters that are written only to control modules do not require module logic. If the **HOST_WRITE** module parameter is true, the host may write to any module parameter. There are no data type limitations when writing to a module parameter.
These parameters influence module behavior, faceplate appearance, and operator control capability. The parameters that may be written include the following, for example.

- Alarm Limits
- Alarm Enable
- Analog SP (for deviation alarming)
- Control Limits (CO and SP)
- ON/OFF Names and Colors
- COMP_CO_ENAB (computer CO write enabled)
- COMP_SP_ENAB (computer SP write enabled)
- APC State (text strings)

Variables and Modes written to the MVCU

Variables that are written from the Host to the MVCU require control module logic to control switching between Host and “console” mode. The variables and modes that may be written from the host to the MVCU include the following:

- Analog CO
- Analog Setpoint
- Digital State (DN)
- Auto/Manual Mode
- Cons/Casc Mode

Write Requirements

The following is a sample of data elements that may be written from the Host Computer to the MVCU based upon DeltaV MVCU module parameters. The OPC links are DeltaV module parameters. A complete list of parameter and variable write references is contained later in this Appendix. Note that the module name (Tagname) precedes all of the parameter references contained in these tables.

AN/FN
Computer Setpoint OPC Link = COMP_SP.CV
Computer Output OPC Link = COMP_CO.CV
A/M Mode OPC Link = BLOCK_FLAGS.CV (0=Auto, 1=Manual)
SP Computer Mode OPC Link = COMP_SP_ENAB.CV (0=Console, 1=Computer)
CO Computer Mode OPC Link = COMP_CO_ENAB.CV (0=Console, 1=Computer)
Other module parameters such as alarm limits, control limits and APC mode.

PID
Computer Setpoint OPC Link = COMP_SP.CV
Computer Output OPC Link = COMP_CO.CV
Cons/Casc Mode OPC Link = FUNCTION_FLAGS1.CV (2=Console, 3=Cascade)
A/M Mode OPC Link = FUNCTION_FLAGS3.CV (0=Auto, 1=Manual)
SP Computer Mode OPC Link = COMP_SP_ENAB.CV (0=Console, 1=Computer)
CO Computer Mode OPC Link = COMP_CO_ENAB.CV (0=Console, 1=Computer)
Other module parameters such as alarm limits, control limits and APC mode.

Auto/Manual Station
Computer Setpoint OPC Link = COMP_SP.CV
Computer Output OPC Link = COMP_CO.CV
A/M Mode OPC Link = FUNCTION_FLAGS3.CV (0=Auto, 1=Manual)
SP Computer Mode OPC Link = COMP_SP_ENAB.CV (0=Console, 1=Computer)
CO Computer Mode OPC Link = COMP_CO_ENAB.CV (0=Console, 1=Computer)
Other module parameters such as alarm limits, control limits and APC mode.

Manual Loader
Computer Output OPC Link = COMP_CO.CV
CO Computer Mode OPC Link = COMP_CO_ENAB.CV (0=Console, 1=Computer)
Other module parameters such as alarm limits, control limits and APC mode.

Totalizer
Computer Setpoint OPC Link = COMP_SP.CV
SP Computer Mode OPC Link = COMP_SP_ENAB.CV (0=Console, 1=Computer)
Other module parameters such as alarm limits, control limits and APC mode.

DN
Computer Output OPC Link = COMP_CO.CV
A/M Mode OPC Link = DIGITAL_FLAGS.CV (0=Auto, 1=Manual)
CO Computer Mode OPC Link = COMP_CO_ENAB.CV (0=Console, 1=Computer)
Other DeltaV Parameters such as alarm status and APC mode.

Design Overview

This section will provide an overview of the RMV/DeltaV Host Computer Access functionality. Several parameters, and control logic, have been added to module templates in this release that provided host access. New module parameters included are described below.

HOST_WRITE

- This module parameter is used as a permissive that will allow the Host to write to DeltaV module parameters.
- Parameters may include alarm levels or control limits. When the parameter is true (=1) DeltaV parameters may be written from the Host.
- The Host application will be required to read the status of the **HOST_WRITE** parameter prior to writing data to module parameters. The **HOST_WRITE** parameter would be enabled from the Detail Display and requires an Administrator level of access to enable.
- There is really no security enforced by DeltaV. This is a flag that the APC may use to control writing to DeltaV parameters.

COMP_CO_ENAB

- The module parameter, **COMP_CO_ENAB**, will be used to determine if either the Host CO or the Operator (Console) CO is transmitted to the MVCU.
- Either the Host Application (APC) or the DeltaV faceplate will be able to enable, or disable, the **COMP_CO_ENAB** parameter.
- The Host CO value will be written to the MVCU when the **COMP_CO_ENAB** parameter is true (=1).
- Module control logic insures bumpless transfer between Host and Console control.

COMP_CO

- The host application will continually write host calculated values to the **COMP_CO** module parameter.
- The parameter **COMP_CO** serves as an input to the Transfer selector block.
- The **COMP_CO** value will be written to the MVCU when the **COMP_CO_ENAB** parameter is true (=1).
- Module control logic insures bumpless transfer between Host and Console control.

COMP_SP_ENAB

- The module parameter, **COMP_SP_ENAB**, is used to determine if either the Host SP or the Operator (Console) SP is transmitted to the MVCU.
- In some cases, such as the Analog Node, the Setpoint is used for deviation alarming and not written to the MVCU.
- Either the Host Application (APC) or the DeltaV faceplate will be able to enable, or disable, the **COMP_SP_ENAB** parameter.
- The Host SP value will be written to the MVCU when the **COMP_SP_ENAB** parameter is true (=1).
- Module control logic insures bumpless transfer between Host and Console control.

COMP_SP

- The host application will continually write host calculated values to the **COMP_SP** module parameter.
- The parameter **COMP_SP** serves as an input to the Transfer selector block.
- The **COMP_SP** value will be written to the MVCU, or used for deviation alarms, when the **COMP_SP_ENAB** parameter is true (=1).
- Module control logic insures bumpless transfer between Host and Console control.

APC_MODE

- There is a requirement for the host computer to write the digital state names to the CCM that annunciate the status of the APC applications. This was a CCM-based mechanism that allowed the OSP/IGS to display graphically the current state of the MPC application.
- A parameter has been added to most module templates with “host access” capability called **APC_MODE.** A text string may be written to the parameter by the APC application indicating the APC mode of the point. The APC mode would be displayed on the faceplates. The **APC_MODE** text string is limited to 6 characters.

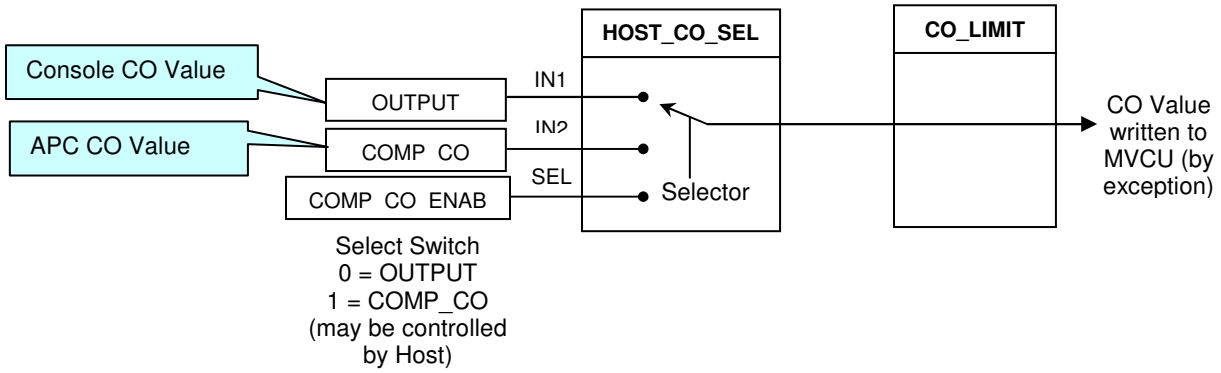
The **APC_MODE** parameter has been added to these Module Templates.

- Analog Node Block (Integer and Float)
- PID
- Manual Loader
- Auto/Manual Station
- Totalizer

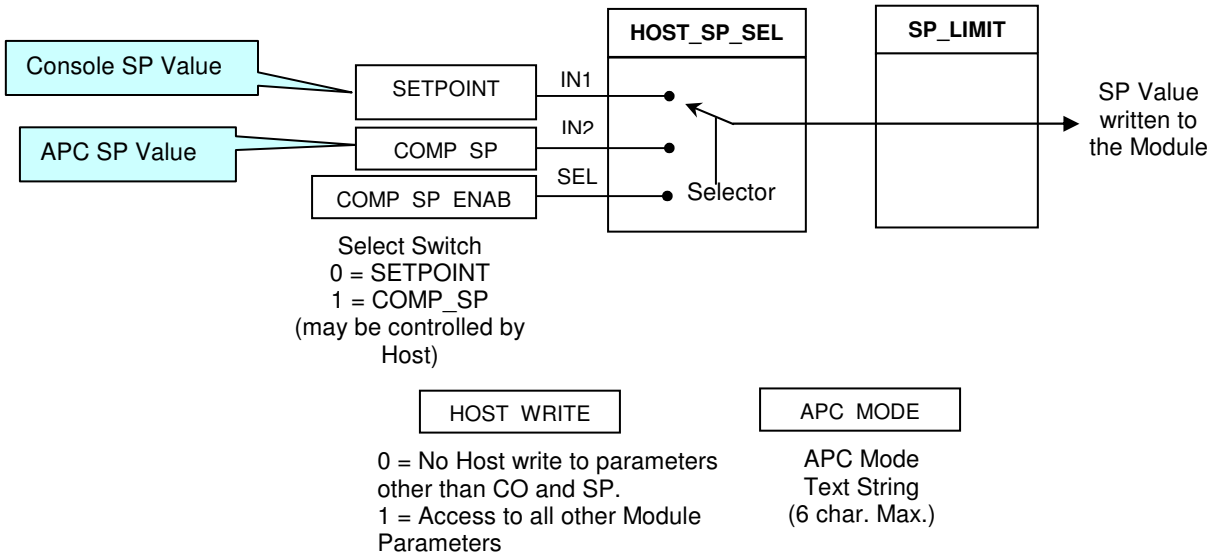
Module Architectural Overview

Shown below are basic depictions of DeltaV control modules changes that facilitate Host access. Module "Logic Block" scripting insures bumpless transfer between "Console" and "Computer" written values.

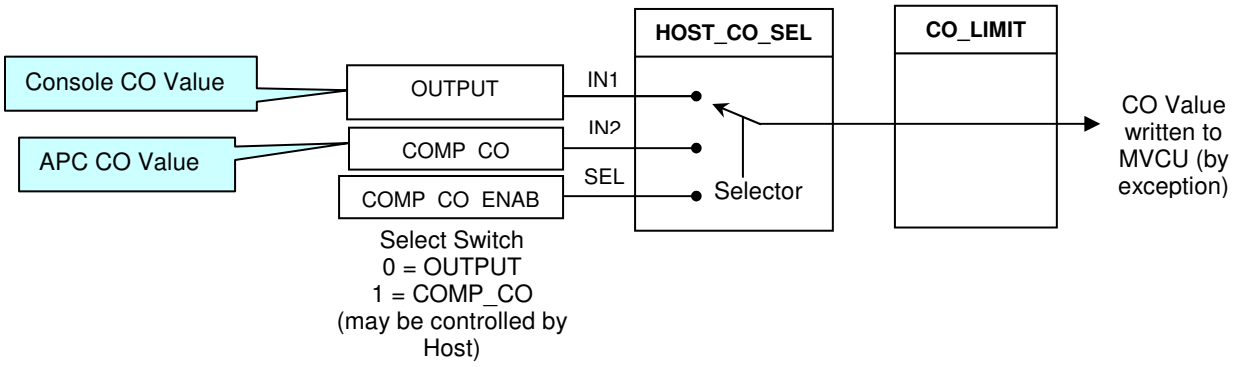
Analog and Floating Node Module
(Output Control)



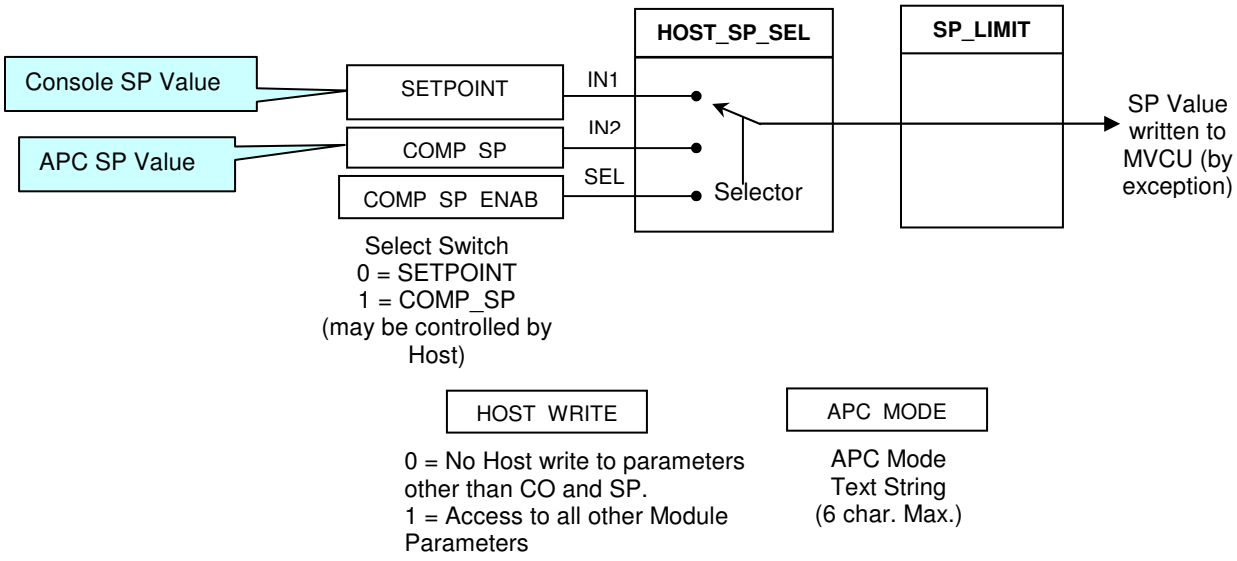
Analog and Floating Node Module
(Setpoint Control)



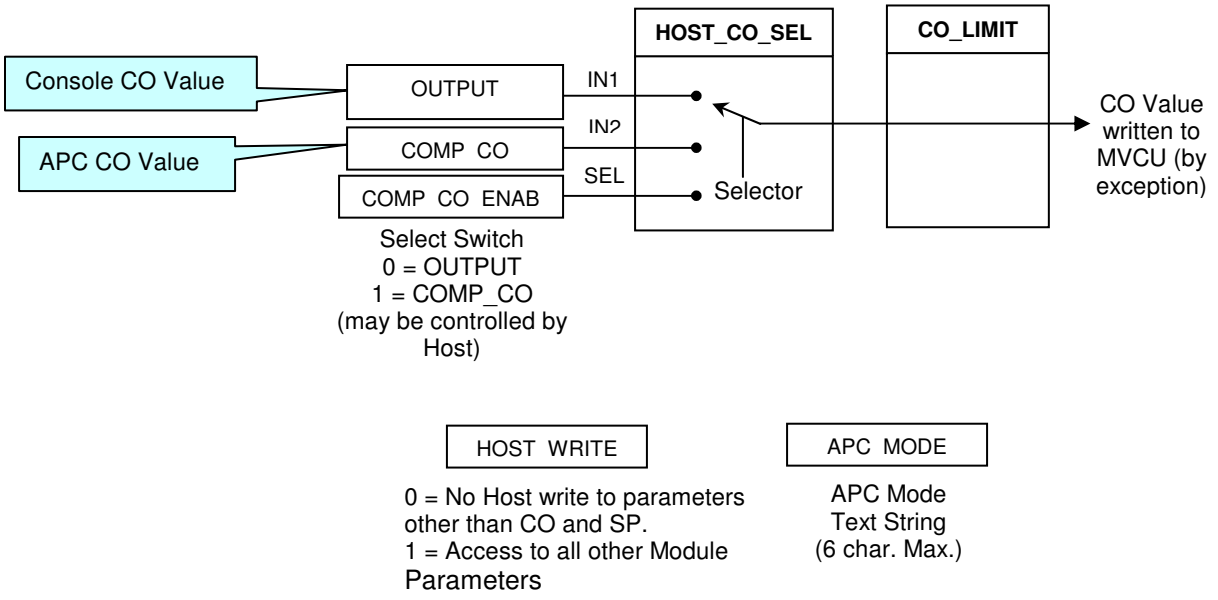
PID Module
(Output Control)



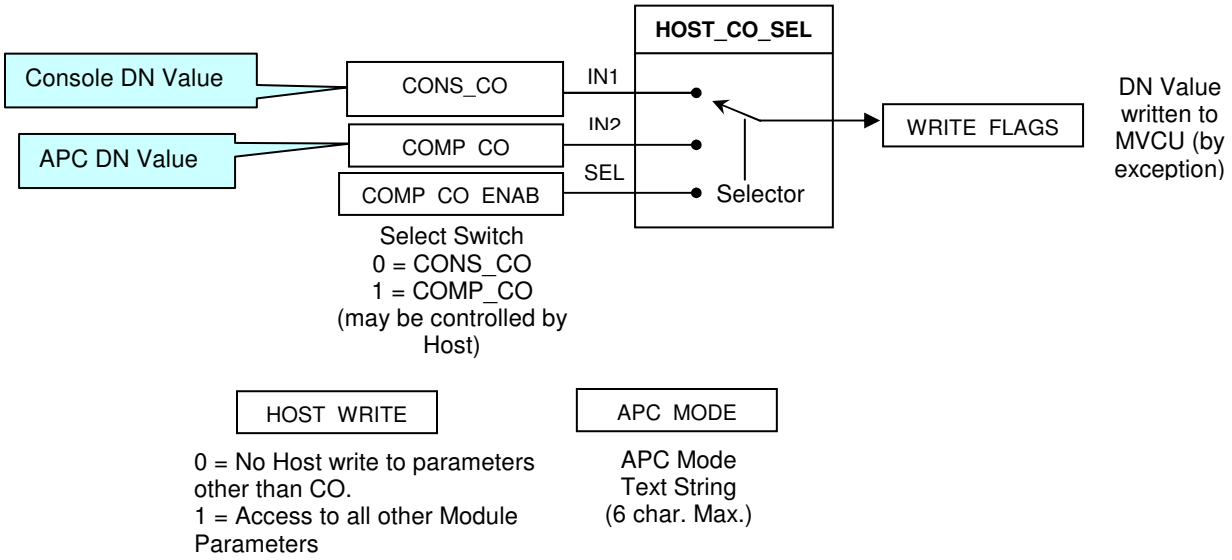
PID Module
(Setpoint Control)



Auto/Man Function and Manual Loader Modules
(Output Control)



Digital Node Module
(Output Control)



Host Access Properties

Depending upon data type, the following parameters may be used to control “Host Access”.

Data Type	HOST_WRITE	COMP_CO	COMP_SP	Setpoint (Deviation Alarms)
Analog Input Block	X			X
Analog Output Block	X			X
Analog Node Block	X	X	X	X
Floating Node Block	X	X	X	X
PID	X	X	X	X
Ratio Station	X			X
Auto/Manual Station	X	X	X	X
Manual Loader	X	X		X
Totalizer	X		X	X
Digital Input Block	X			
Digital Output Block	X			
Digital Node Block	X	X		
Dual In	X			
Timer	X			
Counter	X			
Actuator	X			

Host Read/Write Parameters and Variables

Tabulated below are the parameter references that may allow OPC data exchange with the DeltaV to RMV Interface. Note that the module name (Tagname) precedes all of the parameter references contained in these tables.

Analog Input and Output Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Block Input (PV)	AI1/OUT.CV		Eng. Units	NO
Block Setpoint (SP)	SP_LIMIT/OUT.CV	SETPOINT/OUT.CV	Used for Deviation Alarms	NO
Block Output (CO)	OUTPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV		0=Auto 1=Manual	YES by Operator
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
HI_HI Alarm Level	ALM1/HI_HI_LIM.CV	ALM1/HI_HI_LIM.CV	Eng. Units	NO
HI Alarm Level	ALM1/HI_LIM.CV	ALM1/HI_LIM.CV	Eng. Units	NO
LO Alarm Level	ALM1/LO_LIM.CV	ALM1/LO_LIM.CV	Eng. Units	NO
LO_LO Alarm Level	ALM1/LO_LO_LIM.CV	ALM1/LO_LO_LIM.CV	Eng. Units	NO
Deviation Alarm HI Limit	DV_HI_LIM.CV	DV_HI_LIM.CV	Eng. Units	NO
Deviation Alarm LO Limit	DV_LO_LIM.CV	DV_LO_LIM.CV	Eng. Units	NO
HI_HI Alarm Actual	ALM1/HI_HI_ACT.CV		0=No Alarm 1=Alarm	NO
HI Alarm Actual	ALM1/HI_ACT.CV		0=No Alarm 1=Alarm	NO
LO Alarm Actual	ALM1/LO_ACT.CV		0=No Alarm 1=Alarm	NO
LO_LO Alarm Actual	ALM1/LO_LO_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation LO Alarm Actual	ALM1/DV_HI_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation HI Alarm Actual	ALM1/DV_LO_ACT.CV		0=No Alarm 1=Alarm	NO
HI_HI Alarm Enable	HI_HI_ALM.ENAB	HI_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
HI Alarm Enable	HI_ALM.ENAB	HI_ALM.ENAB	0=Disabled 1=Enabled	NO
LO Alarm Enable	LO_ALM.ENAB	LO_ALM.ENAB	0=Disabled 1=Enabled	NO
LO_LO Alarm Enable	LO_LO_ALM.ENAB	LO_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm HI Enable	DV_HI_ALM.ENAB	DV_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm LO Enable	DV_LO_ALM.ENAB	DV_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Setpoint LO Control Limit	SP_LO_LIM.CV	SP_LO_LIM.CV	Eng. Units	NO
Setpoint HI Control Limit	SP_HI_LIM.CV	SP_HI_LIM.CV	Eng. Units	NO
Output LO Control Limit	CO_LO_LIM.CV	CO_LO_LIM.CV	Eng. Units	NO
Output HI Control Limit	CO_HI_LIM.CV	CO_HI_LIM.CV	Eng. Units	NO

Analog and Floating Node Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Block Input (PV)	AI1/OUT.CV		Eng. Units	NO
Block Setpoint (SP)	SP_LIMIT/OUT.CV		Used for Deviation Alarms	NO
Block Output (CO)	OUTPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV	BLOCK_FLAGS.CV	0=Auto 1=Manual	YES by Operator or Host
Host CO Write Enable	COMP_CO_ENAB.CV	COMP_CO_ENAB.CV	0=Console CO 1=Computer CO	NO
Host Written CO	COMP_CO.CV	COMP_CO.CV	Eng. Units	YES if Enabled
Host SP Write Enable	COMP_SP_ENAB.CV	COMP_SP_ENAB.CV	0=Console SP 1=Computer SP	NO
Host Written SP	COMP_SP.CV	COMP_SP.CV	Eng. Units	NO
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
APC Control Mode	APC_MODE.CV	APC_MODE.CV	Text String (6 characters)	NO
HI_HI Alarm Level	ALM1/HI_HI_LIM.CV	ALM1/HI_HI_LIM.CV	Eng. Units	NO
HI Alarm Level	ALM1/HI_LIM.CV	ALM1/HI_LIM.CV	Eng. Units	NO
LO Alarm Level	ALM1/LO_LIM.CV	ALM1/LO_LIM.CV	Eng. Units	NO
LO_LO Alarm Level	ALM1/LO_LO_LIM.CV	ALM1/LO_LO_LIM.CV	Eng. Units	NO
Deviation Alarm HI Limit	DV_HI_LIM.CV	DV_HI_LIM.CV	Eng. Units	NO
Deviation Alarm LO Limit	DV_LO_LIM.CV	DV_LO_LIM.CV	Eng. Units	NO
HI_HI Alarm Actual	ALM1/HI_HI_ACT.CV		0=No Alarm 1=Alarm	NO
HI Alarm Actual	ALM1/HI_ACT.CV		0=No Alarm 1=Alarm	NO
LO Alarm Actual	ALM1/LO_ACT.CV		0=No Alarm 1=Alarm	NO
LO_LO Alarm Actual	ALM1/LO_LO_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation LO Alarm Actual	ALM1/DV_HI_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation HI Alarm Actual	ALM1/DV_LO_ACT.CV		0=No Alarm 1=Alarm	NO
HI_HI Alarm Enable	HI_HI_ALM.ENAB	HI_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
HI Alarm Enable	HI_ALM.ENAB	HI_ALM.ENAB	0=Disabled 1=Enabled	NO
LO Alarm Enable	LO_ALM.ENAB	LO_ALM.ENAB	0=Disabled 1=Enabled	NO
LO_LO Alarm Enable	LO_LO_ALM.ENAB	LO_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm HI Enable	DV_HI_ALM.ENAB	DV_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm LO Enable	DV_LO_ALM.ENAB	DV_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Setpoint LO Control Limit	SP_LO_LIM.CV	SP_LO_LIM.CV	Eng. Units	NO
Setpoint HI Control Limit	SP_HI_LIM.CV	SP_HI_LIM.CV	Eng. Units	NO
Output LO Control Limit	CO_LO_LIM.CV	CO_LO_LIM.CV	Eng. Units	NO
Output HI Control Limit	CO_HI_LIM.CV	CO_HI_LIM.CV	Eng. Units	NO

PID Function Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
PID Input (PV)	AI1/OUT.CV		Eng. Units	NO
PID Setpoint (SP)	SP_INPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Block Output (CO)	CO_INPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV	FUNCTION_FLAGS3.CV	0=Auto 1=Manual	YES by Operator or Host
Cons/Casc Mode	SP_SOURCE.CV	FUNCTION_FLAGS1.CV	0=Auto (read) 1=Manual (read) 2=Console (write) 3=Cascade (write)	YES by Operator or Host
Host CO Write Enable	COMP_CO_ENAB.CV	COMP_CO_ENAB.CV	0=Console CO 1=Computer CO	NO
Host Written CO	COMP_CO.CV	COMP_CO.CV	Eng. Units	YES if Enabled
Host SP Write Enable	COMP_SP_ENAB.CV	COMP_SP_ENAB.CV	0=Console SP 1=Computer SP	NO
Host Written SP	COMP_SP.CV	COMP_SP.CV	Eng. Units	YES if Enabled
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
APC Control Mode	APC_MODE.CV	APC_MODE.CV	Text String (6 characters)	NO
HI_HI Alarm Level	ALM1/HI_HI_LIM.CV	ALM1/HI_HI_LIM.CV	Eng. Units	NO
HI Alarm Level	ALM1/HI_LIM.CV	ALM1/HI_LIM.CV	Eng. Units	NO
LO Alarm Level	ALM1/LO_LIM.CV	ALM1/LO_LIM.CV	Eng. Units	NO
LO_LO Alarm Level	ALM1/LO_LO_LIM.CV	ALM1/LO_LO_LIM.CV	Eng. Units	NO
Deviation Alarm HI Limit	DV_HI_LIM.CV	DV_HI_LIM.CV	Eng. Units	NO
Deviation Alarm LO Limit	DV_LO_LIM.CV	DV_LO_LIM.CV	Eng. Units	NO
HI_HI Alarm Actual	ALM1/HI_HI_ACT.CV		0=No Alarm 1=Alarm	NO
HI Alarm Actual	ALM1/HI_ACT.CV		0=No Alarm 1=Alarm	NO
LO Alarm Actual	ALM1/LO_ACT.CV		0=No Alarm 1=Alarm	NO
LO_LO Alarm Actual	ALM1/LO_LO_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation LO Alarm Actual	ALM1/DV_HI_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation HI Alarm Actual	ALM1/DV_LO_ACT.CV		0=No Alarm 1=Alarm	NO
HI_HI Alarm Enable	HI_HI_ALM.ENAB	HI_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
HI Alarm Enable	HI_ALM.ENAB	HI_ALM.ENAB	0=Disabled 1=Enabled	NO
LO Alarm Enable	LO_ALM.ENAB	LO_ALM.ENAB	0=Disabled 1=Enabled	NO
LO_LO Alarm Enable	LO_LO_ALM.ENAB	LO_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm HI Enable	DV_HI_ALM.ENAB	DV_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm LO Enable	DV_LO_ALM.ENAB	DV_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Setpoint LO Control Limit	SP_LO_LIM.CV	SP_LO_LIM.CV	Eng. Units	NO

Setpoint HI Control Limit	SP_HI_LIM.CV	SP_HI_LIM.CV	Eng. Units	NO
Output LO Control Limit	CO_LO_LIM.CV	CO_LO_LIM.CV	Eng. Units	NO
Output HI Control Limit	CO_HI_LIM.CV	CO_HI_LIM.CV	Eng. Units	NO

Auto/Manual and Manual Loader Function Blocks

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Function Block Input (PV)	AI1/OUT.CV		Eng. Units	NO
Function Block Setpoint (SP)	SP_INPUT_SCALE/OUT.CV	SETPOINT/OUT.CV	Used for Deviation Alarms	NO
Function Block Output (CO)	CO_INPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV	FUNCTION_FLAGS3.CV	0=Auto 1=Manual	YES by Operator or Host
Host CO Write Enable	COMP_CO_ENAB.CV	COMP_CO_ENAB.CV	0=Console CO 1=Computer CO	NO
Host Written CO	COMP_CO.CV	COMP_CO.CV	Eng. Units	YES if Enabled
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
APC Control Mode	APC_MODE.CV	APC_MODE.CV	Text String (6 characters)	NO
HI_HI Alarm Level	ALM1/HI_HI_LIM.CV	ALM1/HI_HI_LIM.CV	Eng. Units	NO
HI Alarm Level	ALM1/HI_LIM.CV	ALM1/HI_LIM.CV	Eng. Units	NO
LO Alarm Level	ALM1/LO_LIM.CV	ALM1/LO_LIM.CV	Eng. Units	NO
LO_LO Alarm Level	ALM1/LO_LO_LIM.CV	ALM1/LO_LO_LIM.CV	Eng. Units	NO
Deviation Alarm HI Limit	DV_HI_LIM.CV	DV_HI_LIM.CV	Eng. Units	NO
Deviation Alarm LO Limit	DV_LO_LIM.CV	DV_LO_LIM.CV	Eng. Units	NO
HI_HI Alarm Actual	ALM1/HI_HI_ACT.CV		0=No Alarm 1=Alarm	NO
HI Alarm Actual	ALM1/HI_ACT.CV		0=No Alarm 1=Alarm	NO
LO Alarm Actual	ALM1/LO_ACT.CV		0=No Alarm 1=Alarm	NO
LO_LO Alarm Actual	ALM1/LO_LO_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation LO Alarm Actual	ALM1/DV_HI_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation HI Alarm Actual	ALM1/DV_LO_ACT.CV		0=No Alarm 1=Alarm	NO
HI_HI Alarm Enable	HI_HI_ALM.ENAB	HI_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
HI Alarm Enable	HI_ALM.ENAB	HI_ALM.ENAB	0=Disabled 1=Enabled	NO
LO Alarm Enable	LO_ALM.ENAB	LO_ALM.ENAB	0=Disabled 1=Enabled	NO
LO_LO Alarm Enable	LO_LO_ALM.ENAB	LO_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm HI Enable	DV_HI_ALM.ENAB	DV_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm LO Enable	DV_LO_ALM.ENAB	DV_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Setpoint LO Control Limit	SP_LO_LIM.CV	SP_LO_LIM.CV	Eng. Units	NO
Setpoint HI Control Limit	SP_HI_LIM.CV	SP_HI_LIM.CV	Eng. Units	NO
Output LO Control Limit	CO_LO_LIM.CV	CO_LO_LIM.CV	Eng. Units	NO
Output HI Control Limit	CO_HI_LIM.CV	CO_HI_LIM.CV	Eng. Units	NO

RATIO Station and Totalizer Function Blocks

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Function Block Input (PV)	AI1/OUT.CV		Eng. Units	NO
Function Block Setpoint (SP)	SP_INPUT_SCALE/OUT.CV	SETPOINT/OUT.CV	Used for Deviation Alarms	YES by Operator
Function Block Output (CO)	CO_INPUT_SCALE/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV		0=Auto 1=Manual	YES by Operator
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
HI_HI Alarm Level	ALM1/HI_HI_LIM.CV	ALM1/HI_HI_LIM.CV	Eng. Units	NO
HI Alarm Level	ALM1/HI_LIM.CV	ALM1/HI_LIM.CV	Eng. Units	NO
LO Alarm Level	ALM1/LO_LIM.CV	ALM1/LO_LIM.CV	Eng. Units	NO
LO_LO Alarm Level	ALM1/LO_LO_LIM.CV	ALM1/LO_LO_LIM.CV	Eng. Units	NO
Deviation Alarm HI Limit	DV_HI_LIM.CV	DV_HI_LIM.CV	Eng. Units	NO
Deviation Alarm LO Limit	DV_LO_LIM.CV	DV_LO_LIM.CV	Eng. Units	NO
HI_HI Alarm Actual	ALM1/HI_HI_ACT.CV		0=No Alarm 1=Alarm	NO
HI Alarm Actual	ALM1/HI_ACT.CV		0=No Alarm 1=Alarm	NO
LO Alarm Actual	ALM1/LO_ACT.CV		0=No Alarm 1=Alarm	NO
LO_LO Alarm Actual	ALM1/LO_LO_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation LO Alarm Actual	ALM1/DV_HI_ACT.CV		0=No Alarm 1=Alarm	NO
Deviation HI Alarm Actual	ALM1/DV_LO_ACT.CV		0=No Alarm 1=Alarm	NO
HI_HI Alarm Enable	HI_HI_ALM.ENAB	HI_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
HI Alarm Enable	HI_ALM.ENAB	HI_ALM.ENAB	0=Disabled 1=Enabled	NO
LO Alarm Enable	LO_ALM.ENAB	LO_ALM.ENAB	0=Disabled 1=Enabled	NO
LO_LO Alarm Enable	LO_LO_ALM.ENAB	LO_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm HI Enable	DV_HI_ALM.ENAB	DV_HI_ALM.ENAB	0=Disabled 1=Enabled	NO
Deviation Alarm LO Enable	DV_LO_ALM.ENAB	DV_LO_ALM.ENAB	0=Disabled 1=Enabled	NO
Setpoint LO Control Limit	SP_LO_LIM.CV	SP_LO_LIM.CV	Eng. Units	NO
Setpoint HI Control Limit	SP_HI_LIM.CV	SP_HI_LIM.CV	Eng. Units	NO
Output LO Control Limit	CO_LO_LIM.CV	CO_LO_LIM.CV	Eng. Units	NO
Output HI Control Limit	CO_HI_LIM.CV	CO_HI_LIM.CV	Eng. Units	NO

Digital Input and Output Blocks

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Digital Block Input	BLOCK_INPUT.CV		0=OFF 1=ON	NO
Digital Block Output	BLOCK_OUTPUT.CV		0=OFF 1=ON	NO
Auto/Man Mode	AUTO_MANUAL.CV		0=Auto 1=Manual	NO
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
ON State Name	ON_NAME.CV	ON_NAME.CV	Text String (6 characters)	NO
OFF State Name	OFF_NAME.CV	OFF_NAME.CV	Text String (6 characters)	NO
ON State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
OFF State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO

Digital Node Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Digital Block Input	BLOCK_INPUT.CV		0=OFF 1=ON	NO
Digital Block Output	BLOCK_OUTPUT.CV	DIGITAL_FLAGS.CV	0=OFF (Read) 1=ON (Read) 2=OFF (Write) 3=ON (Write)	YES by Operator or Host
Auto/Man Mode	AUTO_MANUAL.CV	DIGITAL_FLAGS.CV	0=AUTO (Read) 1=MAN (Read) 0=AUTO (Write) 1=MAN (Write)	YES by Operator or Host
Host CO Write Enable	COMP_CO_ENAB.CV	COMP_CO_ENAB.CV	0=Console CO 1=Computer CO	NO
Host Written CO	COMP_CO.CV	COMP_CO.CV	0=OFF 1=ON	YES if Enabled
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
ON State Name	ON_NAME.CV	ON_NAME.CV	Text String (6 characters)	NO
OFF State Name	OFF_NAME.CV	OFF_NAME.CV	Text String (6 characters)	NO
ON State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
OFF State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO

Dual Digital Input

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Digital Input 1	BLOCK_OUTPUT.CV		0=OFF 1=ON	NO
Digital Input 2	BLOCK_OUTPUT1.CV		0=OFF 1=ON	NO
Auto/Man Mode	AUTO_MANUAL.CV		0=Auto 1=Manual	NO
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
Input 1 State Name	ON_NAME.CV	ON_NAME.CV	Text String (6 characters)	NO
Input 2 State Name	OFF_NAME.CV	OFF_NAME.CV	Text String (6 characters)	NO
Input 1 State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
Input 2 State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO

Timer Function Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Digital Block Input	BLOCK_INPUT.CV		0=OFF 1=ON	NO
Digital Block Output	BLOCK_OUTPUT.CV		0=OFF 1=ON	YES by Operator
Timer Current Value	T_VALUE/OUT.CV		Eng. Units	NO
Timer SP Value	T_SETPOINT/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV		0=AUTO 1=MAN	YES by Operator
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
ON State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
OFF State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO

Counter Function Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Digital Block Input	BLOCK_INPUT.CV		0=OFF 1=ON	NO
Digital Block Output	BLOCK_OUTPUT.CV		0=OFF 1=ON	YES by Operator
Timer Current Value	C_VALUE/OUT.CV		Eng. Units	NO
Timer SP Value	C_SETPOINT/OUT.CV		Eng. Units	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV		0=AUTO 1=MAN	YES by Operator
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
ON State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
OFF State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO

Actuator Function Block

Parameter	Read Reference	Write Reference	Notes	Write to MVCU
Auto Input 1	AUTO_INPUT_1.CV		0=OFF 1=ON	NO
Auto Input 2	AUTO_INPUT_2.CV		0=OFF 1=ON	NO
Confirm Open	CONFIRM_OPEN.CV		0=OFF 1=ON	NO
Confirm Closed	CONFIRM_CLOSED.CV		0=OFF 1=ON	NO
Output 1 (ON)	OUTPUT_1.CV		0=OFF 1=ON	YES by Operator
Output 2 (OFF)	OUTPUT_2.CV		0=OFF 1=ON	YES by Operator
Output 3 (HOLD)	OUTPUT_3.CV		0=OFF 1=ON	YES by Operator
Alarm Output	ALARM_OUTPUT.CV		0=OFF 1=ON	YES by Operator
Auto/Man Mode	AUTO_MANUAL.CV		0=AUTO 1=MAN	YES by Operator
Host Write Enabled	HOST_WRITE.CV		0=Not allowed 1=Allowed	NO
Digital Alarm State	DI1/DISC_ACT.CV		0=No Alarm 1=Alarm	NO
Digital Alarm Level	DI1/DISC_LIM.CV	DI1/DISC_LIM.CV	0=Off State 1=ON State	NO
Digital Alarm Enable	DISC_ALM.ENAB	DISC_ALM.ENAB	0=Disabled 1=Enabled	NO
ON State Name	ON_NAME.CV	ON_NAME.CV	Text String (6 characters)	NO
OFF State Name	OFF_NAME.CV	OFF_NAME.CV	Text String (6 characters)	NO
ON State Color	ON_COLOR.CV	ON_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO
OFF State Color	OFF_COLOR.CV	OFF_COLOR.CV	RED, YELLOW, GREEN, WHITE	NO