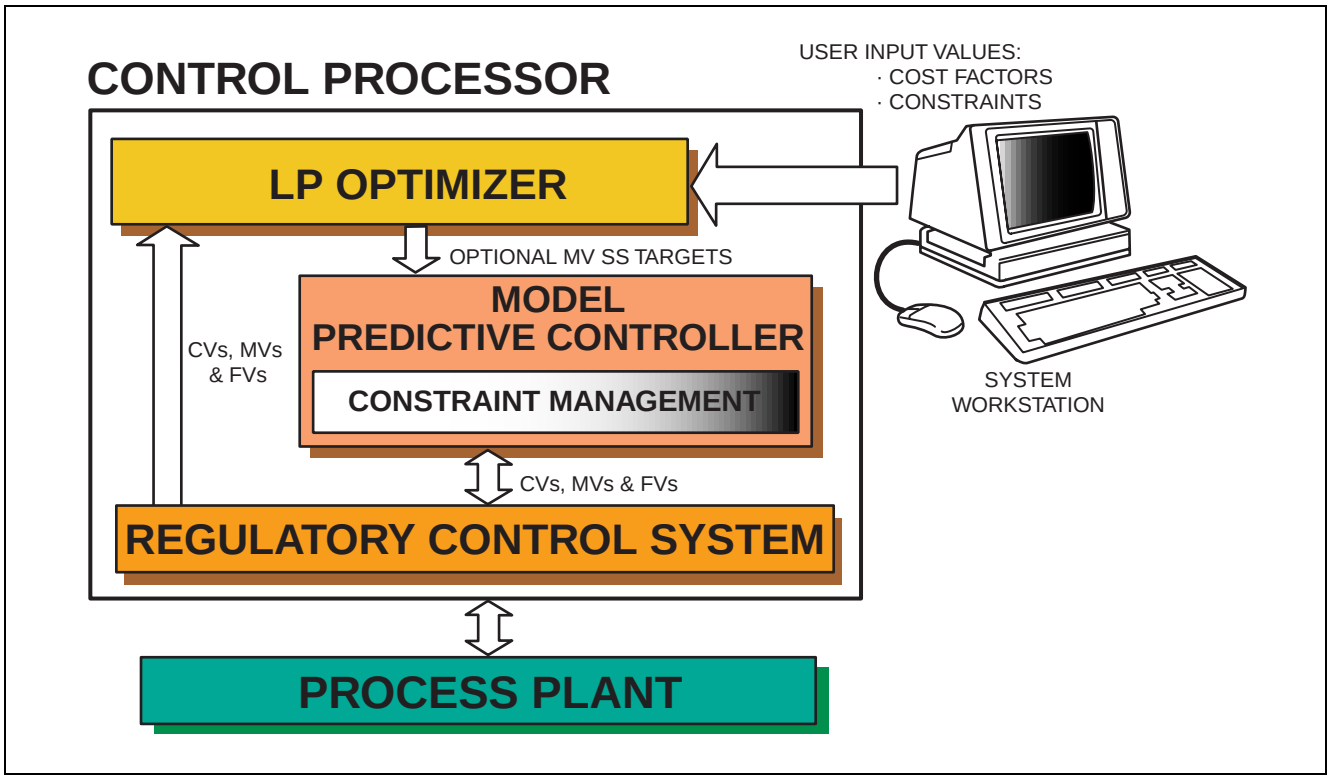
FOXBORO®

PSS 21S-3N5 B4

I/A Series® Software
Embedded Multivariable Controller



The Embedded Multivariable Controller (MVC) is a multivariable control algorithm executing in a Control Processor (CP), such as CP60. This allows you to use the fault-tolerant environment of a CP as the platform for advanced multivariable control schemes. Two block types, the Multivariable Controller (MVC) block and the Multivariable Loop (MVL) block, support an Embedded MVC application running in the CP. These blocks support the multivariable control algorithm and form the interface to other CP blocks. The MVL block gets Controlled Variable (CV) and Feedforward Variable (FV) measurements from blocks such as the AIN and writes the Manipulated Variable (MV) supervisory set points to other regulatory blocks (PIDA, DPIDA, or RATIO).

Model Based Predictive Control in a workstation such as an I/A Series Application Workstation 51E or a high-end Windows NT laptop is the source of the Embedded MVC model which is downloaded as the database for the MVC and MVL blocks. Once created, and then populated by the download database file, the MVC and MVL blocks are free running, needing no additional data from the workstation. Using the MVC and MVL blocks, you can tune and operate the free-standing multivariable controller independent of the workstation environment from which the MVC model was downloaded. The blocks can then be checkpointed, rebooted, and so forth.

The Embedded MVC features:

- Fault-Tolerant Operation – The multivariable controller can run in an optionally fault-tolerant CP.
- Low Measurement Latency – MVC and MVL blocks in the CP run at the Block Processing Cycle (BPC), ensuring that the data presented to the Embedded MVC application is no more than 0.5 seconds old.
- Fast Execution Frequency -– An Embedded MVC application typically runs at a multiple of a 1.0 second interval.
- Minimum Configuration Effort – The Embedded MVC offers you the alternative of going directly from a developed process model to a deployed multivariable controller in a very short period of time. The configuration effort has been greatly reduced, eliminating the configuration of I-signals, readback signals, watchdogs, and other Distributed Control System (DCS) inputs. Many of the connections are implicit or defaulted, reducing the number of configuration details that you must consider.
- Reduced Maintenance Effort – The entire Embedded MVC application is resident within a given Control Processor, reducing the maintenance effort in comparison to a multivariable controller implemented in a workstation and needing support through external processors, networks, and so forth.
- Enhanced Alarming – The Embedded MVC uses the I/A Series block alarm facilities that allow you to customize your alarm requirements.
- I/A Series Object Support – No external Object Management support is required to develop customized screen displays.
- Simulation – Process data is used to generate a linear dynamic model of the process; this process model may then be implemented in CP blocks as a process simulation to help in development and testing of the multivariable controller model.
- Linear Programming (LP) Optimizer – To locate an operating point (a set of steady-state optimum values for the MVs and CVs of the system) that maximizes profit while causing both hard and soft constraints to be respected.

PRODUCT CONCEPT

The Embedded MVC concept employs the Native Connoisseur product to develop and test a multivariable controller (Figure 1). A Connoisseur Toolkit, hosted by a high-end Windows NT laptop computer interfaced to an I/A Series system via an Ethernet link, is used to conduct plant tests, collect data, analyze and develop process models, develop and test controllers, and to perform initial controller commissioning.

After the multivariable controller model development is completed, you can download the model into a CP for permanent deployment, making the MVC application independent of the workstation in which it was developed. The CP provides a fault-tolerant environment for the Embedded MVC application, default displays for the MVC and MVL blocks, and a Secure Supervisory Control (SSC) interface to the existing regulatory controls. With the application supported within the CP, you can incorporate the multivariable controller into Display Manager user graphics and other I/A Series functions.

ENHANCED RELIABILITY

Downloading the multivariable controller application to the Control Processor allows you to take advantage of the fault-tolerant design of the CP. The CP is an optionally fault-tolerant station that performs regulatory, logic, timing, and sequential control together with connected Fieldbus Modules (FBMs). The fault-tolerant CP configuration consists of two modules operating in parallel with two separate connections to the Nodebus and the Fieldbus.

The two Control Processor modules, married together as a fault-tolerant pair, are designed to provide continued operation of the unit in the event of virtually any hardware failure occurring within one module of the pair. Both modules receive and process information simultaneously, and faults are detected by the modules themselves. One of the significant methods of fault detection is comparison of communication messages at the module external interfaces. Upon detection of a fault, self-diagnostics are run by both modules to determine which module is defective. The non-defective module then assumes control without affecting normal system operations.

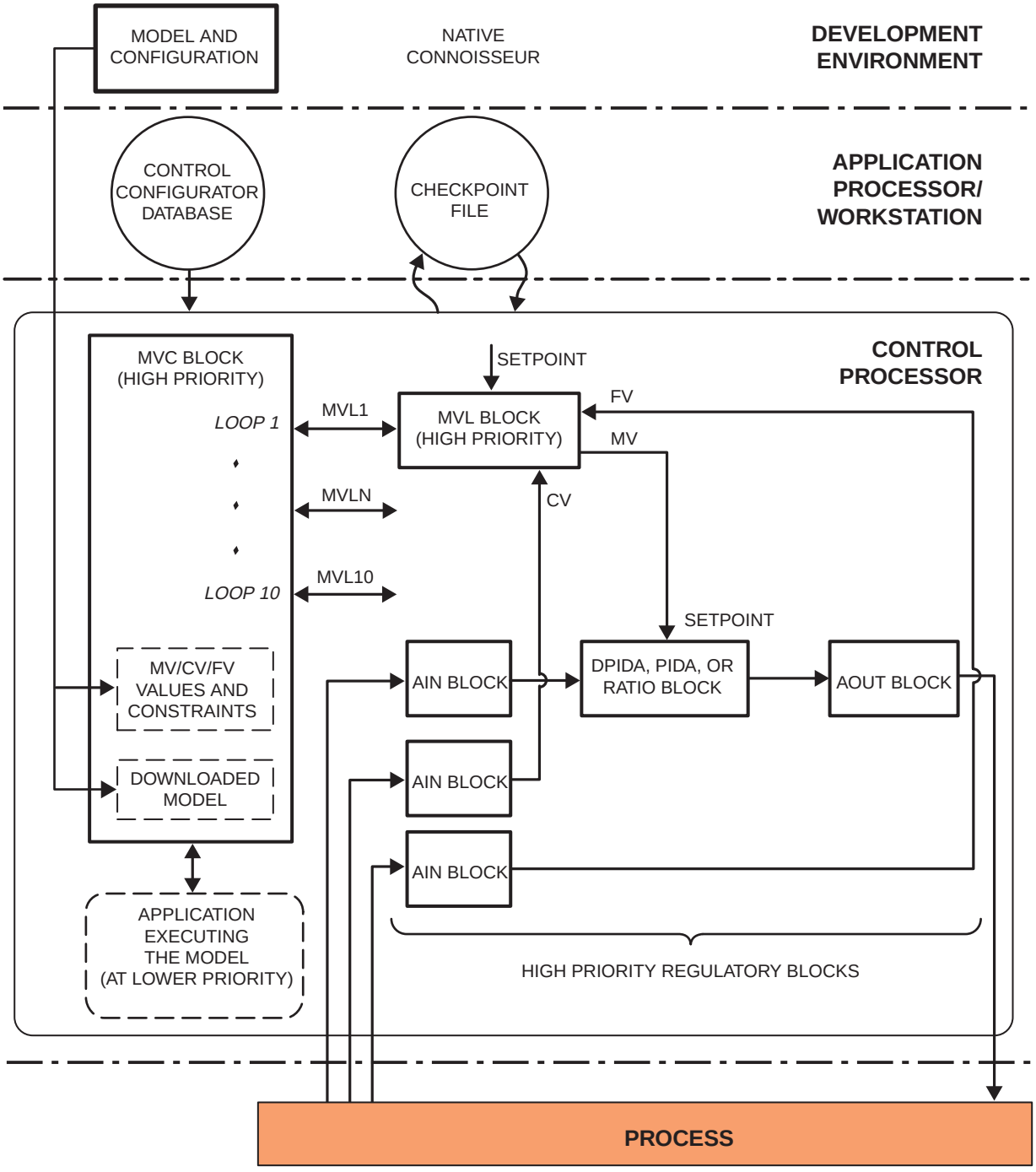


Figure 1. Embedded MVC (Showing One Loop)

SECURE SUPERVISORY CONTROL (SSC) SUPPORT

The Embedded MVC uses the I/A Series SSC functionality to perform group enables, group disables, and the ability to write the supervisory set points of manipulated variable CP regulatory control blocks. Specifically, SSC allows:

- The CP Station block to set a group enable/disable, placing all the manipulated variable regulatory control blocks into the SSC enable/disable state. This allows you to place an MVC controller into its AUTO state without having to change the state of individual regulatory controllers.
- Either the primary or secondary loop of an existing regulatory control scheme to be the supervisory set point. This allows you to interface directly to an existing regulatory control scheme without loop restructuring.
- Automatic fallback without operator intervention and with a range of fallback options should the MVC controller fail and shed its manipulated variables.
- The controller to automatically switch to the fallback state, if the vital MV, CV, or FV fails or is removed from control by the operator.
- The underlying regulatory control blocks that have been enabled to issue an initialization request (one per block) to the supervisory application when a group enable occurs. As a security feature, if an acknowledgement is not received, the blocks do not change into the supervisory state, and the regulatory controls continue normal operation.
- The underlying manipulated variable regulatory control block status flags to be passed up through the regulatory control hierarchy to the Embedded MVC application, providing an indication of the MV saturation state to prevent further MV windup.

OPERATOR DISPLAYS

The Embedded MVC contains default detail displays for the MVC and MVL blocks. These displays allow you get an application running quickly without having to build custom displays.

MVC Operator Display

The MVC Operator display (Figure 2) allows you to:

- Review MVC and MVL status information
- Navigate to other MVL blocks
- Alter the operational state of the controller (AUTOMATIC, WAIT INITIALIZE, READY, OFF)
- Review MVC configuration parameters downloaded from the Native Connoisseur product.

MVL Operator Display

The MVL Operator display (Figure 3) allows you to:

- Review a number of MVL block status parameters
- View MV, CV, and FV trends and configuration overlays
- Acknowledge and view block alarm messages
- View FV value and limits
- View CV set point value and limits
- View MV value, limits, and target
- Change the in-service and out-of-service state of MV, CV, or FV.
- Change values, such as the CV set point, the MV target, and so forth
- Change tuning parameter constants.

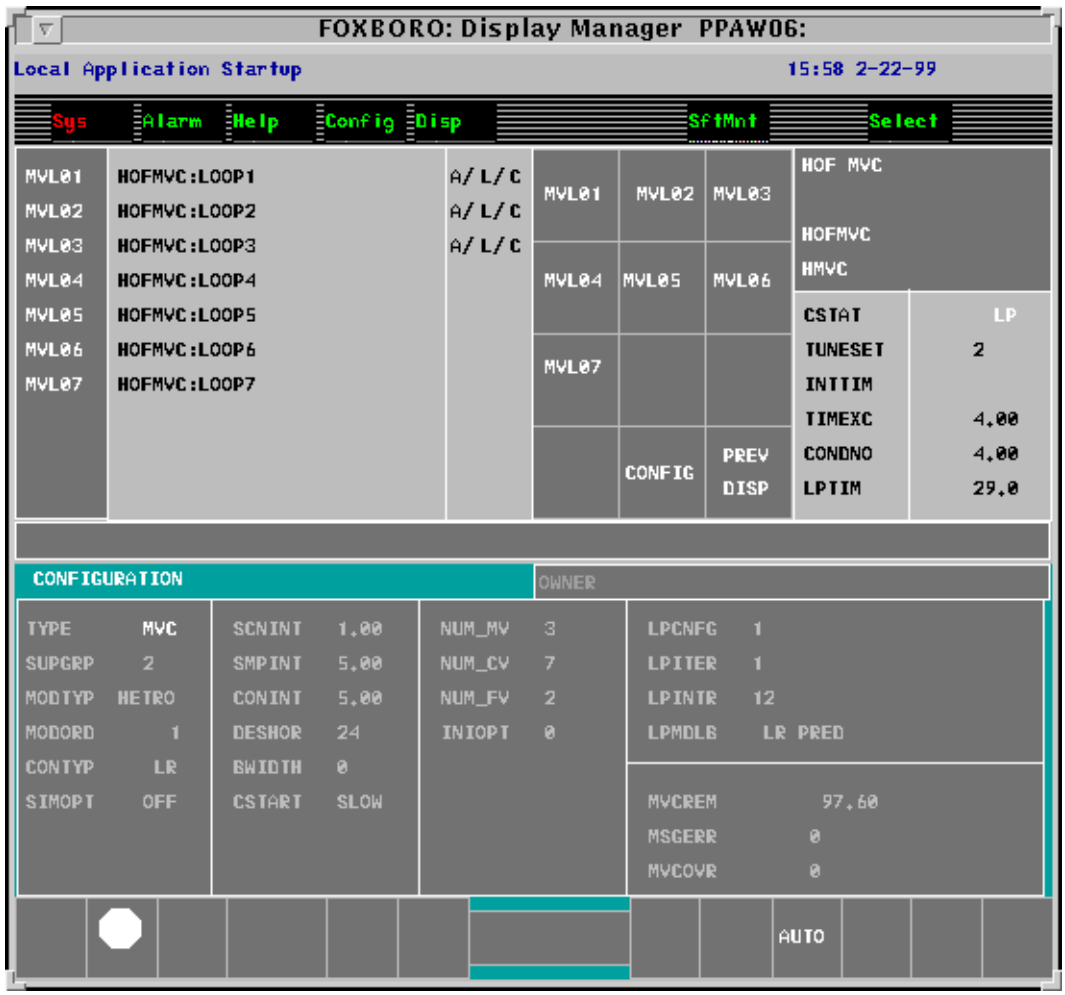


Figure 2. MVC Default Display

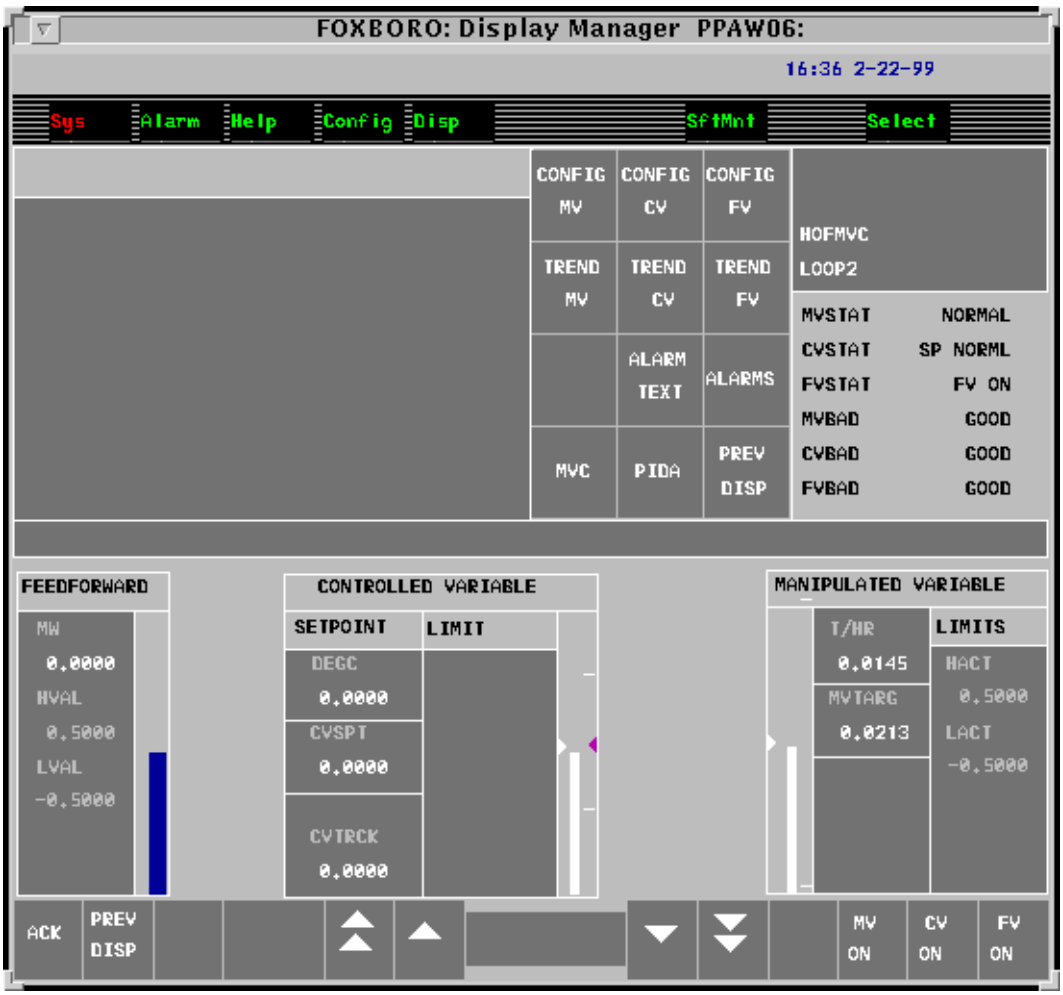


Figure 3. MVL Default Display

PRODUCT CONFIGURATION

The Embedded MVC (Figure 4) is hosted within a Control Processor 60 (or later). The Native Connoisseur instance typically resides in a Solaris-based station (for example, AW51E) or in an Ethernet connected station such as a high-end laptop with a Windows NT operating system for use as a Connoisseur Toolkit.

The platform hosting the Native Connoisseur instance is the source of the Embedded MVC Control Processor database. No particular restrictions are placed upon the location of the Native Connoisseur instance within the topology.

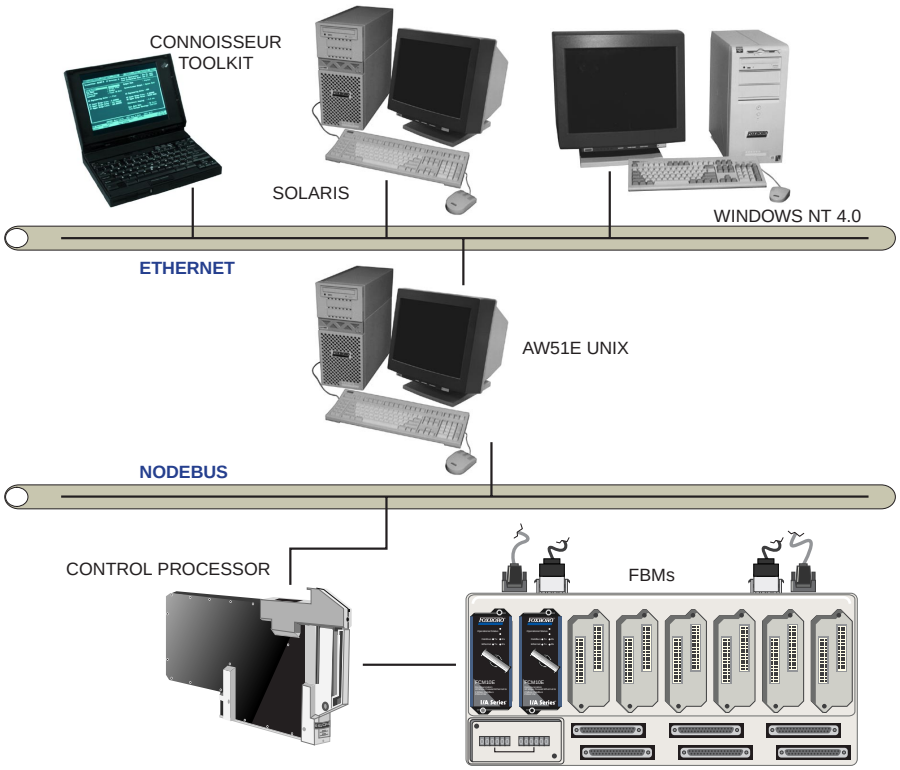


Figure 4. MVC System Configuration

FUNCTIONAL SPECIFICATIONS

Control Processor
Control Processor 60 (CP60), supporting one multivariable controller

Number of Loops per Embedded MVC
Ten Loops - Limited to 10 Connoisseur loops (up to 10 manipulated variables, up to 10 controlled variables, and up to 10 feedforward variables).

Control Blocks
The control blocks that can be used for manipulated variables are the PIDA, DPIDA, and RATIO.

Algorithm Supported
The Long Range algorithm of Connoisseur Version 14 is supported in the Embedded MVC product.

10 Gainsets
The Embedded MVC supports multiple gainsets to improve computation efficiency.

Intersampling
Different sample and control intervals are available, with the same restrictions as imposed by Native Connoisseur product.

MV/CV/FV Out of Service/Vital
Switches are provided to put the manipulated, controlled, or feedforward variable in-service and out-of-service. The reaction of the controller application depends on the state of the variable vital flag.

Messaging
Messaging functionality is similar to that of Connoisseur Version 14, but with additional information to support the download facility.

Secure Supervisory Set Points
The MVs of the Embedded MVC accept secure supervisory set points.

FUNCTIONAL SPECIFICATIONS (Cont.)

Scan and Basic
The Embedded MVC uses a fixed "Basic" interval, with MVC and MVL blocks running at the Control Process Basic Processing Cycle (BPC), with adjustable scan, sample and control intervals.

Linear Programming Optimizer
A Linear Programming facility is included in the Embedded MVC.

Software Release Requirements
I/A Series Version 6.2 or later, and Connoisseur Version 14.01 or later.

Constraint Management
The Embedded MVC supports both MV hard constraint and CV soft constraint management.

Alarming
The Embedded MVC uses the I/A Series alarm facilities that allow you to "customize" the alarms to suit your requirements.

MV, CV, FV "Bad"
In the Embedded MVC the variable status connections are made automatically with no configuration requirements. The reaction of the control application to the "Bad" status is dependent upon the state of the variable vital flag.

CV "Off with Inference" and MV "Off with Feedforward"
As supported in Connoisseur Version 14.

Fast/Slow Initialization
As supported in Connoisseur Version 14.

Controller State Machine
The Embedded MVC controller supports the Connoisseur state machine (OFF, INITIALIZATION, READY, AUTO).

MV Windup/Readback Support
The Embedded MVC provides manipulated variable windup protection. The connections to downstream I/A Series control blocks are made automatically, with no additional user configuration required.

Control Variables Validation
The Control Processor supports the validation mechanism of Connoisseur Version 14. The entry of validation limits is modified to take advantage of I/A Series system features.

Process Model
The Embedded MVC resident controller supports the use of a single model; non-linear neural models are not supported.

Simulation
Control blocks in the CP can be used for process simulation while the multivariable controller is being developed.

Watchdog Timer
No external watchdog timer is required since the Embedded MVC is running in a CP.

Connoisseur Functions
The following Connoisseur functions are not supported in the Embedded MVC: Plant Analysis System (PAS) functions, Pseudo Random Binary Sequences (PRBS) entities and associated data logging, Multiple Signal Types, Director Language, run-time statistics, and "Level" constructs.

The Foxboro Company
33 Commercial Street
Foxboro, Massachusetts 02035-2099
United States of America
http://www.foxboro.com
Inside U.S.: 1-508-543-8750 or 1-888-FOXBORO (1-888-369-2676)
Outside U.S.: Contact your local Foxboro Representative.

MB 021 Printed in U.S.A. 0599

An Invensys company