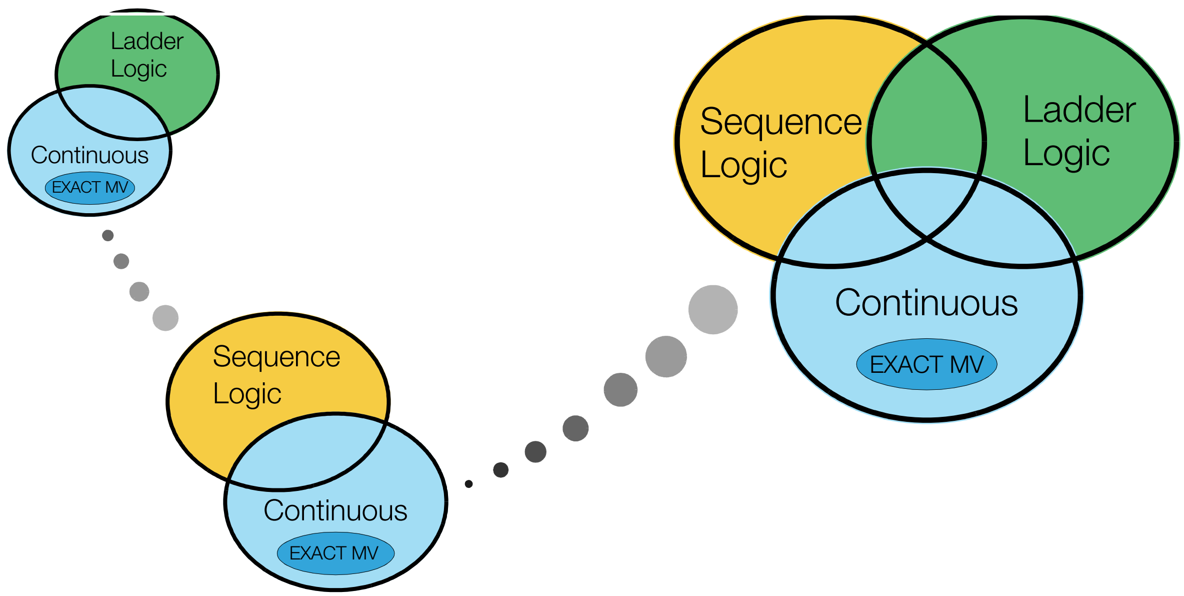
**Foxboro Evo™**
**Process Automation**
**System**

**Product Specifications**

**PSS 31S-3B3 B3**

# Field Control Processor 280 (CP280) Integrated Control Software




*The Foxboro Evo™ Process Automation System's Integrated Control Software provides for the integration of continuous, ladder logic and sequential control in a Foxboro Evo control strategy.*

## FEATURES

- Integration of continuous, sequential, and ladder logic control domains for ease of implementing complex control strategies for a wide variety of process control applications
- Comprehensive set of continuous control blocks (algorithms) for input/output, valve/motor control, regulatory control, computation, logic, and alarming
- Comprehensive set of ladder logic instructions for designing modular solutions to logic control problems in familiar, easy-to-use relay ladder symbols
- High-level sequence programming language provides a comprehensive set of sequence logic instructions for developing sequential, feedback-oriented applications at equipment control level
- Integration of control blocks within groups for ease of configuring control strategies
- EXACT® MV advanced control algorithms provide adaptive feedback and feedforward PID control capability with pre-tuning functions
- Configuration of continuous block types, ladder logic and sequential blocks via the following flexible, intuitive, user interfaces:

- Foxboro Evo™ Control Editors, including the Block Configurator, Programmable Logic Block (PLB) Ladder Logic Editor, the Sequence Block HLBL Editor, and the Sequence Block SFC Editor
- The I/A Series® Configuration Component (IACC) software
- Integrated Control Configurator

- Reuse of block configurations from one controller to another.

## OVERVIEW

The Foxboro Evo Integrated Control Software provides a selection of continuous, sequential, and ladder logic control domains. Each domain can be used independently or integrated to meet specific application requirements.

The control software provides the user with a comprehensive control strategy and input/output implementation for on/off control, timing, regulatory and feedback applications. Implemented within a block and parameter structure, the control software provides the base for the integration of continuous, ladder logic and sequential control with programmable logic controller functionality.

The software control strategies allow you to mix and match continuous and Programmable Logic Block (PLB) capabilities with sequencing in the same control scheme.

The process control domains – continuous, sequential and batch, and ladder logic – execute in any Foxboro control processor. The control software running in a control processor provides the interface to the Fieldbus subsystem and other I/O subsystems through I/O blocks. These I/O blocks interface to the specific applications executing in the Fieldbus Modules (FBMs).

Access to the FBMs and associated devices (for example, field sensors, actuators, Intelligent Field Devices) is through the control software in the Field Control Processor 280 (FCP280).

An optional fault-tolerant control processor configuration, where two identical control processors (Primary and Shadow) are running in parallel, provides enhanced reliability to the control process.

For more information on the FCP280, refer to *Field Control Processor 280 (FCP280)* (PSS 31H-1B11 B3).

## EXACT Multivariable Control (EXACT MV)

The Foxboro Evo EXACT Multivariable Controller (EXACT MV) with Adaptive Multivariable Control provides feedforward tuning of up to four variables, plus tuning of the feedback controller. On the low end, it provides superior control of a single difficult to control loop with multiple load upsets. On the high end, multiple blocks can interconnect interactive signals among the loops to improve control for up to five interacting control loops in a five by five cross coupled scheme.

At the core of the EXACT MV is an advanced control block, PIDA, for advanced Proportional-Integral-Derivative control. However, the formula for adaptive multivariable capability resides in the feedback tuning extender block, FBTUNE, and the feedforward tuning block, FFTUNE.

## CONTROL SOFTWARE CONCEPTS

Figure 1 shows the integration of continuous control, ladder logic, and sequential control in a single control processor. A block is a member of a set of algorithms that perform a certain control task. The Foxboro Evo control blocks also support user-defined arithmetic and Boolean logic functions, alarm detection and reporting, and shared variables, which are global variables that allow data to be shared across applications within the Foxboro Evo Process Automation System.

The control software also provides communication interface blocks called Equipment Control Blocks (ECBs). ECBs are located in the control processor and provide I/O and control information related to Migration products, FBMs, Field Device System Integrators (FDSIs) and their related devices (such as actuators, sensors, and Intelligent Field Devices, including third-party field I/O devices).

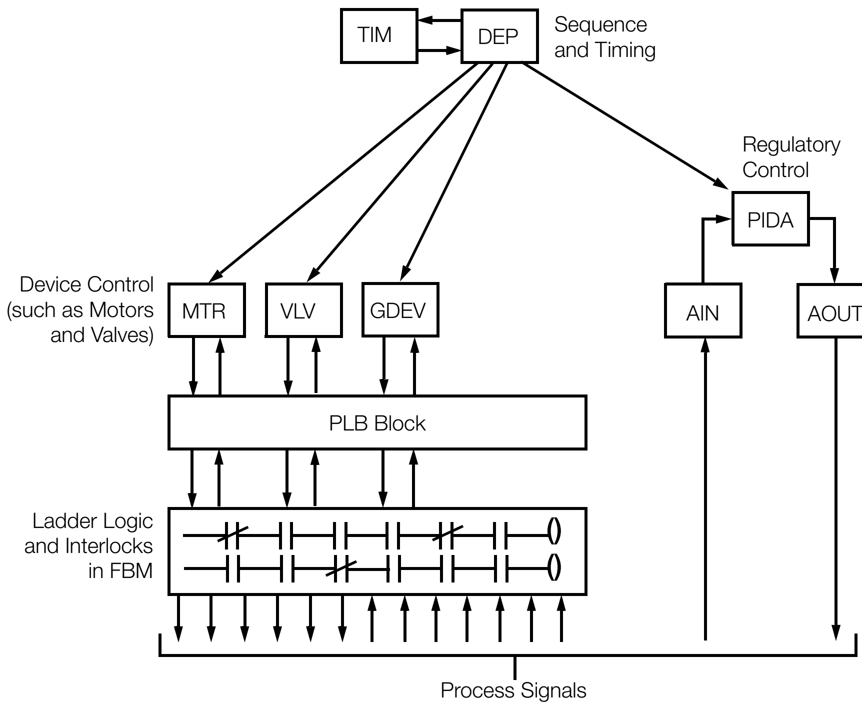



*Figure 1. Total Integration of Control Software in a Single Control Processor*

## Block Processing Speed

A high-priority task scheduled by the operating system in the control processor processes the control blocks every Block Processing Cycle (BPC), as shown in Figure 2. Block processing speed is configurable from 50 ms to 60 minutes. For configuration specifications, refer to specific FBM product specification sheets.

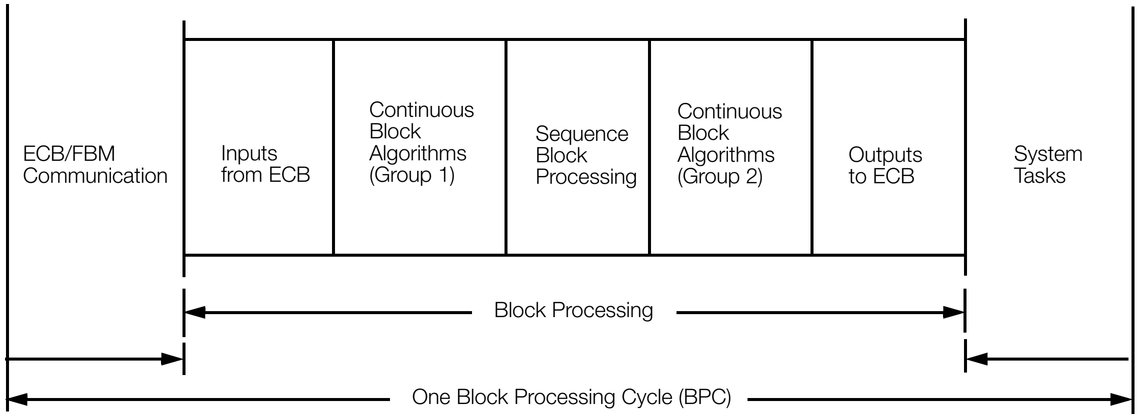



*Figure 2. Block Processing Cycle (BPC)*

At the beginning of block processing, I/O data is read from each Fieldbus Module, for both input and output channels. Data is read at the ECB scan period specified for that module. However, data cannot be read faster than the BPC. At the end of block processing, process outputs are written to each of the Fieldbus Modules.

### Station Block

Each control processor contains a station block automatically created to provide global data on control processor system functions. The station block contains information such as:

- Continuous block processing load
- Total CP load
- Basic processing cycles and CP overruns
- Number of bytes of dynamic free memory
- Number of peer-to-peer connections
- Sequence processing load
- I/O scan load
- OM scan load
- Idle time
- Station alarm groups
- Configuration security option.

Previous CPs only supported one I/O channel to an HDLC fieldbus (which provides communication to a chain of Fieldbus Modules) directly, or used external hardware, such as the FEM100 (discussed in PSS 21H-2Y14 B4) to support up to four I/O channels. However, the FCP280 supports four I/O channels directly, without the use of such external hardware.

Every HDLC fieldbus' channel requires a software driver in the CP called a *Primary ECB*. Thus, the FCP280's Compound has four Primary ECBs - one for each channel - through which the station block directs all communications. Each of its FBM's ECBs must be assigned to the Primary ECB for the channel on which their FBM resides via a new parameter added to them called "CHAN".

## CONTROL CONFIGURATION CONCEPTS

The Foxboro Evo control blocks are configured from the Process Engineer's Environment, which provides access to the Block Configurator, Integrated Control Configurator (ICC), or I/A Series Configuration Component (IACC).

The Block Configurator, ICC, or IACC provide for the configuration of Equipment Control Blocks associated with Fieldbus Modules. These blocks provide the means of communicating the control information among the control processors, Fieldbus Modules, actuators, sensors, and Intelligent Field Devices.

Configuration of continuous block types, ladder logic and sequential blocks is accomplished through the Block Configurator, ICC, or IACC. During the configuration process, the configurator recognizes the various block types chosen by the user and is able to identify their unique control domain association. For additional information on:

- Block Configurator, refer to PSS 31S-10B3 B3
- ICC, refer to PSS 21S-3A3 B3
- IACC, refer to PSS 21S-2B5 B4.

Block configurations can be reused from controller to controller by importing them into the configuration environment. In this way, the plant intellectual property can be preserved and leveraged.

The Intelligent field devices (IFDs) are configured using either the Intelligent Field Device Configurator (IFDC) or the Foxboro Evo maintenance tool-set compliant with the Field Device Tool (FDT) 1.2 Standard. For more information on the IFDC, refer to PSS 21S-8A3 B3. For more information on the FDT compliant tool-set, refer to PSS 21S-8A5 B3.

## CONTINUOUS CONTROL CONCEPTS

To use a cascade and feedforward control strategy as an example, you might build the strategy by selecting the following blocks:

- Three Analog Input (AIN) blocks
- One advanced Proportional/Integral/Derivative (PIDA) control block
- One Proportional/Integral/Derivative (PID) control block
- One Analog Output (AOUT) block
- One Feedback Tuner Extender (FBTUNE) for PIDA
- One Feedforward Tuner Extender (FFTUNE) for PIDA.

In the cascade control strategy shown in Figure 3, each block requires a unique name. This can be any combination of up to 12 characters, for example, FLWTRNS100.

FBTUNE and FFTUNE extender blocks may be linked to the PIDA blocks to provide feedback and feedforward adaptive tuning.

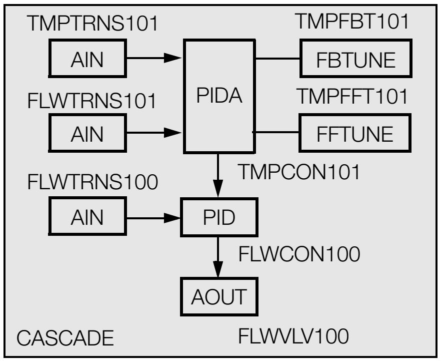



*Figure 3. Example of a Cascade Control Strategy*

The feedforward input may be used to anticipate the effect of a measured load or an interacting controller. Refer to *EXACT Multivariable Control (EXACT MV)* (PSS 21S-3A2 B3) for additional information.

The control processor must be sized so that it does not exceed any of the following limits:

- Maximum number of FBMs allowed
- Maximum throughput of the Fieldbus
- Available memory to store the control blocks
- Available memory to store the Object Manager (OM) scanner databases
- Available time for processing control blocks and ECBs.

**NOTE**

> The control processor should be sized following the guidelines in *Field Control Processor 280 (FCP280) Sizing Guidelines and Excel Workbook* (B0700FY).

Any block can be connected to any other block anywhere in the system. Compound names must be unique within the system. Block names must be unique only within a Compound.

### Input/Output Blocks

I/O blocks provide the interface to the process inputs and outputs, (for example, sensors and valves). The I/O blocks relate logical names, such as FLWTRNS100, to physical hardware point addresses, identified by the device ID and point number.

All references to I/O points from control configuration, displays, data logging, and so on use a logical name instead of the physical address.

### Equipment Control Blocks (ECBs)

The software interface centered about the Equipment Control Block (ECB) is the communications network between the Foxboro Control Processor and the process instrumentation. ECBs vary depending on the Fieldbus Module (FBM) and its application. The list of ECBs is located in Table 1 on page 11.

FBMs read the I/O data from the process instrumentation on a per FBM basis when the Block Processing Cycle begins. The FBM then conditions (for example, digitizes and normalizes) the data where necessary, and stores the data and the status into its ECB. When the control blocks are processed, the Control Processor uses newly retrieved data and generates new outputs for the appropriate ECB.

Some ECB(s) also provide an extended communications interface for a direct interchange between the Foxboro Evo Process Automation System and field devices using various digital protocols. The field devices incorporate intelligence at the hardware device level (signal conditioning and process control) and use the ECBs in the FCP280 to provide the information to the control blocks.

## ALARMING

The control blocks provide a comprehensive set of I/O alarm indicators and messages including:

- Bad input and output point alarming
- High/low absolute alarming of the analog measurement, input or output signal
- Deviation alarming of the setpoint minus measurement error
- Contact state change alarming
- Priority level (1 to 5) indication of alarms
- Alarm type indication of the highest-priority active alarm for the block

When an alarm occurs, the Alarm Manager notifies the operator via messages on process displays and alarm printers, alarms horns and annunciator keyboards. High/low absolute and deviation alarm limits, alarm enable/disable options, and other block alarm parameters are set via the Foxboro Evo Control Editors, IACC, or ICC configurators.

## LADDER LOGIC CONCEPTS

Ladder logic allows you to design modular solutions to logic control problems in familiar, easy-to-use relay ladder symbols. Figure 4 shows a sample ladder diagram.

By itself, ladder logic performs simple relay-type operations. Ladder logic used in conjunction with a control processor's continuous and sequential control blocks can implement sophisticated control strategies.

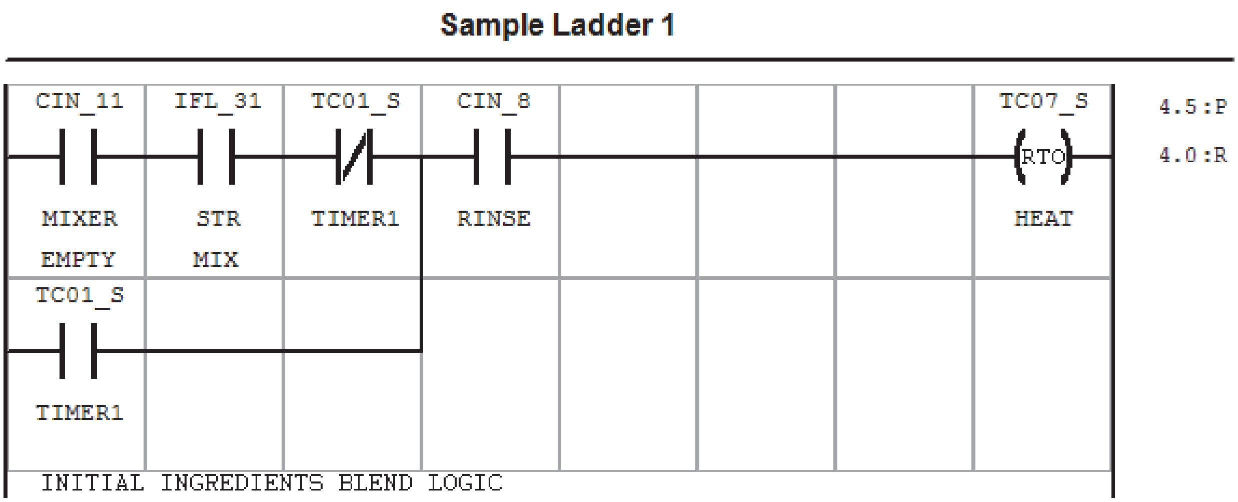



*Figure 4. Sample Ladder Diagram*

A Programmable Logic Block (PLB) provides connections between a ladder diagram and user tasks, other blocks, and other ladder diagrams. Connection is made through PLB input and output parameters. These parameters map to user-configured external flag references within a ladder diagram division called a segment.

PLB Editor software allows you to construct a ladder diagram in segments, check for syntax errors, and produce a printed copy for documentation. You can compile the ladder diagram and install the code in a Fieldbus Module or save the ladders for later use. You can develop a library of ladder diagrams and retrieve (copy) segments for inclusion in other ladder diagrams.

Using the PLB View, you can view the state of the ladder logic associated with the PLB. Using the PLB Monitor, you can monitor the ladder logic and force contacts and coils on or off to verify correct operation of the logic under simulated process conditions.

The PLB Detail Display, automatically created after ladder logic configuration, allows you to monitor the status of ladder logic contacts, timers, counters, and coils through a ladder diagram display, which uses industry standard symbols, or through graphic displays that you create.

## SEQUENTIAL CONTROL CONCEPTS

Sequential control is one of the control domains of the Integrated Control Software. Sequence, continuous, and ladder logic blocks can be combined within a single compound structure (see Figure 5).

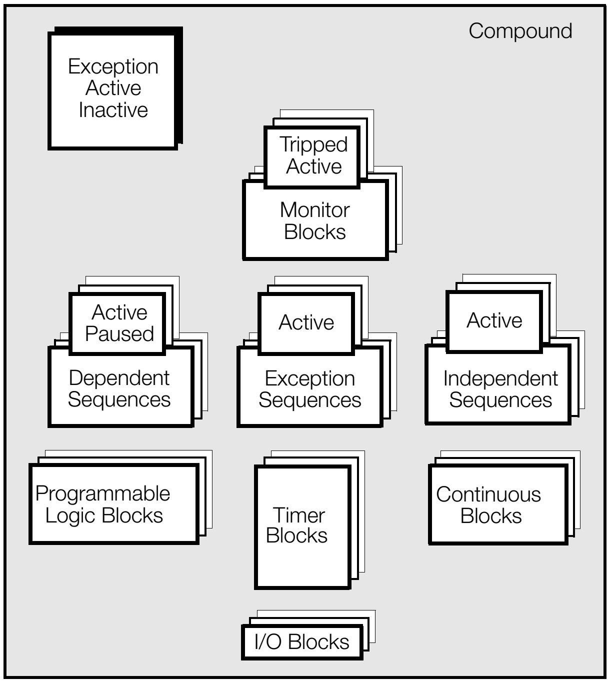



*Figure 5. Example of Compound Contents*

Sequence blocks contain logic that supervises the control loops. The logic regulates such things as:

- Pressure control
- Temperature control
- Agitator control
- Ingredient fills
- Gas control.

Sequential control is a series of equipment control level activities such as filling a tank, blending its contents, and draining the tank.

In the continuous control domain, control blocks have fixed algorithms, a fixed number of parameters, and fixed properties. Additionally, continuous block algorithms can refer only to their own parameters.

In the sequential control domain, on the other hand, you can build your own algorithms using a logical set of parameters within the control compound/block structure. Sequence block algorithms can also read and write other compound/block parameters directly.

A high-level language and a parameter set provide the necessary components to configure Sequence blocks. When built, these blocks can be combined with continuous and/or ladder logic blocks in a compound.

## Sequential Control Block Classes

There are three classes of blocks: Sequence, Monitor, and Timer.

| | |
|---|---|
| Sequence blocks (DEP, IND, EXC) | Manipulate block parameters, or shared variables.<br>Activate other Sequence blocks and Monitor blocks.<br>Send messages to historians. |
| Monitor blocks (MON) | Monitor up to 16 process conditions independently (parameter values and Boolean expressions).<br>Activate other Sequence blocks. |
| Timer blocks (TIM) | Keep track of time while control strategies are executing. |

## Block States

The block states are Inactive, Active, Paused, or Tripped.

The Inactive state means that a Sequence block is not executing statements or that the Monitor block is not evaluating conditions.

The Active state means that a Sequence block is executing statements or that the Monitor block is evaluating conditions.

The Paused state means that a Dependent block is in a suspended condition. Dependent blocks pause whenever an Exception block in the same compound becomes active. The Dependent block becomes active again when the Exception blocks complete their execution. The Exception and Independent blocks are never paused. The Independent blocks do not affect the execution of other Sequence blocks, and other Sequence blocks do not affect the operation of Independent blocks.

The Monitor block has a Tripped state (one of its conditions is true). Therefore, a sequence is activated by the Monitor block. If a Monitor block activates another block because one of its conditions has become true, it enters the Tripped state until the other block is finished processing.

## Processing Sequence Blocks

Sequence blocks can run in conjunction with continuous blocks, ladder logic blocks, and each other in that:

- Sequences may be Active concurrently.
- Monitor blocks may be Active at the same time as Sequence blocks.

Timing is an independent feature and can run in conjunction with other blocks. The timer block provides four (4) independent timer functions.

## Sequence Language

The Sequence block language is a structured language programming language focused on control applications. The language includes logic flow control statements as well as Boolean and arithmetic functions.

The language statements do not operate the I/O directly. Rather, they make connections between their own parameters and I/O block parameters. They can write the I/O block parameters within continuous, ladder logic, or other Sequence blocks.

### Logic Flow Control Statements

These statements determine the flow of control. They may select groups of statements to be executed, skip them, execute them repetitively, or delay their execution. They are:

```
IF. . .THEN. . .ELSEIF. . .ELSE. . .ENDIF
FOR. . .TO. . .DO. . .ENDFOR
REPEAT. . .UNTIL
WHILE. . .DO. . .ENDWHILE
EXITLOOP
GOTO
WAIT. . .TIME
WAIT. . .UNTIL CONDITION
EXIT
RETRY
```

### Data Operation Statements

There are two kinds of statements that can manipulate data: the Assignment statement and the Procedural statement.

The Assignment statement replaces the current value of some object with a new value that results from evaluation of an expression.

The Procedural statements are:

| | |
|---|---|
| ACTIVATE | Activates a Sequence block or a Monitor block. |
| ABORT | Aborts an active Sequence block or Monitor block. |
| START_TIMER | Starts timers at current value or select value. |
| STOP_TIMER | Stops timers. |
| ACTCASES | Manipulates activity of the 16 Monitor block cases. |
| SENDMSG | Initiates a message from executing sequence logic. It can address any object that acts like a logical device, such as historians or annunciator keys. It can also assign a message to a string parameter. |
| SEND CONF | Sends messages interactively to logical devices or objects that act like logical devices. A message received confirmation is expected. |

## OPERATOR INTERFACE INTEGRATION

For each block in the user's control strategy (including ECBs), a Block Detail Display with a standard block faceplate is available allowing the user immediate access to block parameters and functions via the FoxView™ window. For more information on FoxView software, refer to PSS 21S-2B8 B4.

You can also use the FoxDraw™ application to create custom process displays. For more information on FoxDraw™ software, refer to PSS 21S-2B3 B4.

## I/A SERIES CONTROL BLOCKS

The Foxboro Evo control blocks supported by the FCP280 are listed in Table 1.

**Table 1. Foxboro Evo Control Blocks**

| Input/Output | |
|---|---|
| AI | Analog Input block connects to an AI function block in an H1 FOUNDATION fieldbus™ device via an FBM228 to receive a single analog input for a Foxboro Evo control strategy. |
| AIN | Analog Input block supports a single input channel from a Fieldbus Module. |
| AINR | Redundant Analog Input block supports a redundant input from redundant Fieldbus Modules supporting either a single transmitter or redundant transmitters. |
| AO | Analog Output block connects to an AO function block in an H1 FOUNDATION fieldbus device via an FBM228 to send a single analog output from a Foxboro Evo control strategy to the device. |
| AOUT | Analog Output block provides auto/manual with bias function and supports a single output point for an analog type Fieldbus Module. |
| AOUTR | Redundant Analog Output supports a selected redundant output to redundant Fieldbus Modules. |
| BIN | Binary Input block receives a single binary input from an external device for input to a Foxboro control processor via a Distributed Control Interface (DCI). |
| BINR | Redundant Binary Input block selects a single binary input from redundant external sources for input to a Foxboro control processor via a Distributed Control Interface (DCI). The source of the input can be either two or three redundant inputs from the same device or different devices. |
| BOUT | Binary Output block sends a single binary output from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). |
| BOUTR | Redundant Binary Output block sends a digital output from a Foxboro control processor to up to three devices via a Distributed Control Interface (DCI). |
| CIN | Contact Input block supports a single input point for a digital input type Fieldbus Module. |
| CINR | Redundant Contact Input block receives redundant contact inputs from a pair of Fieldbus Modules to select and alarm a single digital input for the Foxboro Evo control strategy. |
| COUT | Contact Output block supports a single output for a digital type Fieldbus Module. |
| COUTR | Redundant Contact Output block is a digital contact output block that provides the Foxboro Evo control strategy with optionally pulsed output capability for a single digital point directed to a pair of redundant Fieldbus Modules. |

Table 1. Foxboro Evo Control Blocks (Continued)

| | |
|---|---|
| DI | Digital Input block connects to a DI function block in an H1 FOUNDATION fieldbus device via an FBM228 to receive a single digital input for a Foxboro Evo control strategy. |
| DO | Digital Output block connects to a DO function block in an H1 FOUNDATION fieldbus device via an FBM228 to send a single digital output from a Foxboro Evo control strategy to the device. |
| EVENT | Event block provides messages reporting the sequencing of state-change events detected in a contact input Fieldbus Module. |
| IIN | Integer Input block receives a single integer input from an external device for input to a Foxboro control processor via a Distributed Control Interface (DCI). |
| IINR | The Redundant Integer Input block is a Distributed Control Interface (DCI) block that enables the control strategy to read a single integer value selected from multiple points in the same or different field devices. |
| IOUT | Integer Output block sends a single integer output from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). |
| MAI | The Multiple Analog Input block enables the control strategy to read up to eight analog values from an MAI device function block operating in a FOUNDATION fieldbus™ H1 device. |
| MAIN | Multiple Analog Input block supports up to 8 inputs from a Fieldbus Module and an internal channel for a temperature reference sensor. |
| MAO | The Multiple Analog Output block enables the control strategy to write up to eight analog values to an MAO device function block operating in a FOUNDATION fieldbus™ H1 device. |
| MCIN | Multiple Contact Input block supports up to 32 contact inputs from digital input type Fieldbus Modules. |
| MCOUT | Multiple Contact Output block supports up to 16 digital outputs for a digital type Fieldbus Module. |
| MDI | FOUNDATION™ fieldbus Multiple Discrete Input supports up to eight inputs from a FOUNDATION fieldbus H1 device via an FBM228. |
| MDO | FOUNDATION™ fieldbus Multiple Discrete Output supports up to eight outputs from a FOUNDATION fieldbus H1 device via an FBM228. |
| PAKIN | Packed Input block receives a 32-bit discrete data input from an external device for input to a Foxboro control processor via a Distributed Control Interface (DCI). |
| PAKINR | The Redundant Packed Input block is a Distributed Control Interface (DCI) block that reads up to 32 contiguous bits from an external device. The source of the value may be specified as either two or three redundant points in the same device or different devices. |

Table 1.  Foxboro Evo Control Blocks (Continued)

| | |
|---|---|
| PAKOUT | Packed Input block sends a 32-bit discrete data output from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). |
| RIN | Real Input block receives a single real input from an external device for input to a Foxboro control processor via a Distributed Control Interface (DCI). |
| RINR | Redundant Real Input block selects a single binary input from redundant external sources for input to a Foxboro control processor via a Distributed Control Interface (DCI). The source of the input can be either two or three redundant inputs from the same device or different devices. |
| ROUT | Real Output block sends a single real output from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). |
| ROUTR | Redundant Real Output block sends redundant real outputs from a Foxboro control processor to two or three external devices via a Distributed Control Interface (DCI). |
| STRIN | String Input block receives a string input from an external device for input to a Foxboro control processor via a Distributed Control Interface (DCI). |
| STROUT | String Output block sends a string output from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). |
| **Device Control** | |
| GDEV | General Device block provides Open/Close control of motor, or air, operated valves, and Run/Stop control of 2-wire or 3-wire motor circuits. |
| MOVLV | Motor-Operated Valve block operates two related output contacts, which open/close a motor-operated valve on an incremental basis. |
| MTR | Motor Controller block performs both 2-wire and 3-wire motor control functions. |
| PLSOUT | Pulse Output block sends on/off or start/stop type commands through two momentary pulsed outputs from a Foxboro control processor to an external device via a Distributed Control Interface (DCI). Typically, these commands are sent to a latching function in a device, with one command used for the set input of the latch and the other used for the reset input. |
| VLV | Valve on/off controller block operates two related output contacts, which open or close a solenoid valve. |
| **Regulatory Control**[a] | |
| BIAS | Bias block produces an output that is the sum of the two input values, MEAS and BIAS, each of which can be scaled independently. |

Table 1. Foxboro Evo Control Blocks (Continued)

| | |
|---|---|
| RATIO | Ratio block computes an output that is the scaled multiplication of a measurement input with a ratio set-point input. |
| DGAP | Differential Gap block provides optional bi-state or tri-state on/off control of two Boolean outputs. |
| LIM | Limiter block provides a position and velocity limiter. |
| OUTSEL | Output Select block allows the selection of one of two inputs from upstream blocks to be used as output to the process. |
| PID | Proportional, Integral, Derivative block provides functions of a traditional, interacting, 3-term controller. |
| PIDA | Advanced Proportional, Integral, Derivative block provides functions of an EXACT MV multi-variable controller and is used in conjunction with the feedforward and feedback tuning blocks. |
| PIDFF | FOUNDATION™ fieldbus Proportional, Integral, Derivative block integrates the corresponding FOUNDATION fieldbus H1 device PID function block into the Foxboro Evo Process Automation System. |
| FFTUNE | Feedforward Tuning block connects to the PIDA block as an extension block for feedforward control loops. |
| FBTUNE | Feedback Tuning block connects to the PIDA block as an extension block for performing enhanced EXACT MV tuning for feedback control loops. |
| PIDE | PID with EXACT Tuning block provides PID with the EXACT self-tuning algorithm. |
| PIDX | PID Extended block adds to the PID block: a sampled-data control option to use with sampling type instruments; TRACK capability that forces the output to track an independent track input; an optional non-linear gain element for Ph control; a batch option that provides preloadable integral bias for batch control. |
| PIDXE | PID Extended with EXACT Tuning block combines the functionality of PIDX and PIDE. |
| PTC | Proportional Time Controller block performs the functions of a proportional-time on/off controller. |
| UNIVFF | Universal FOUNDATION™ fieldbus connects to a resource, transducer or function block operating in a FOUNDATION fieldbus H1 device. |
| **Selection, Ramping, Dynamic Compensation** | |
| DTIME | Dead Time block delays the input a variable time interval before making it available at the output. |

**Table 1. Foxboro Evo Control Blocks (Continued)**

| | |
|---|---|
| LLAG | Lead/Lag block compensates signal value by making output dynamically lead or lag the input. |
| RAMP | Ramp block performs a multi-segment ramp sequence; up to 5 segments may be used. |
| SIGSEL | Signal Selector block examines up to 8 inputs and produces an output dependent upon a relational selection option. |
| SWCH | Switch Position Selector block selects either of two independent inputs. |
| **Computation, Logic, and Conversion** | |
| ACCUM | Accumulator block accumulates a real input signal and scales it to produce a real output quantity. |
| CALC | Calculator block provides up to 50 sequentially-executed arithmetic and logical operations. Has the capability of a programmable scientific pocket calculator. |
| CALCA | Advanced Calculation block adds dual-operand efficiency to many mathematical and logical calculation operations. |
| CHARC | Characterizer block converts a real input to a real output using a table lookup of 20 piecewise linear conversion segments. |
| LOGIC | Logic block provides logic and timer functions. |
| MATH | Mathematics block provides a set of mathematics functions for specialized control needs. |
| PATT | Pattern Matching block provides matching capability for 16-bit patterns. |
| STATE | State block outputs selected 16-bit patterns. |
| **Alarm** | |
| ALMPRI | Alarm Priority Change block dynamically reassigns the specified priority of an alarm point. |
| BLNALM | Boolean Alarm block provides independent state-change alarm messages for 8 Boolean-type inputs. |
| MEALM | Measurement Alarm block provides an alarm message for one measurement input with limit indicators for Intelligent Field Devices: high-low absolute alarming, rate-of-change alarming, and high-high/low-low alarming. |
| MSG | Message Generator block provides a state change message for each of eight inputs from Intelligent Field Devices. |
| PATALM | Pattern Alarm block compares the relationship of up to 8 Boolean inputs to up to 8 unique user-specified patterns. |

**Table 1. Foxboro Evo Control Blocks (Continued)**

| | |
|---|---|
| REALM | Real Alarm block supports 3 types of alarming: high/low absolute alarming on the measurement, rate-of-change alarming on the measurement, high/low deviation alarm on the measurement/set point difference. |
| STALM | State Alarm block provides state alarming for event changes received from an Intelligent Field Device. |
| **Sequence Blocks** All Sequence blocks (IND, DEP, and EXC) provide sequential control for regulatory feedback applications at the equipment control level. | |
| DEP | Dependent Sequence block is used for normal sequence logic to define sequence of events, activate/deactivate other Sequence blocks, activate/deactivate Monitor (MON) blocks or individually monitor cases of a MON block, control timers in Timer (TIM) blocks, access any shared variable/parameter of any block in the system. Pauses when EXC blocks in the same compound are active. |
| EXC | Exception Sequence block is used to handle abnormal events. Exception blocks are normally activated by monitor blocks. |
| IND | Independent block is used to provide the same functions as the Dependent block; however, the IND block does not pause as DEP does when any EXC blocks in the same compound are active. |
| MON | Monitor block provides the capability of monitoring process conditions. |
| TIM | Timer block contains four individual timers that can be run by a Sequence block (IND, DEP, or EXC) to time sequence activities. |
| **Data Storage** | |
| REAL | Real Data Variable block provides the capability of storing a real data value for use by other control blocks. |
| BOOL | Boolean Data Variable block provides the capability of storing a Boolean data value for use by other control blocks. |
| LONG | Long Integer Data Variable block provides the capability of storing a long integer data variable for use by other control blocks. |
| STRING | String Data Variable block provides the capability of storing a configurable and settable string data variable for use by other control blocks. |
| PACK | Packed Long Data Variable block provides the capability of storing a packed long data variable for use by other control blocks. |

**Table 1. Foxboro Evo Control Blocks (Continued)**

| **Equipment Control Blocks** | |
|---|---|
| ECB1 | Analog Input. |
| ECB2 | Analog Input & Analog Output. |
| ECB4 | Pulse Input. |
| ECB5 | Digital In, Sustained/Momentary, Digital Out. |
| ECB6 | Sequence of Events Input[b]. |
| ECB7 | Digital In & Pulse Count Input. |
| ECB8 | Ladder Logic. |
| ECB9 | Remote/Manual Station (Analog I/O, Digital I/O). |
| ECB11/ ECBP | Primary ECB 1-4 for the FCP280 - named PRIMARY_ECB, PRIMARY_ECB2, PRIMARY_ECB3, and PRIMARY_ECB4. |
| ECB18 | Intelligent Transmitter ECB provides data and configuration information directly from Intelligent Transmitters as output to control blocks. |
| ECB38R | Intelligent Transmitter Input, Analog Output, Dual Baud Rate, Redundant I/O. |
| ECB41 | Analog Input FBC. |
| ECB42 | Digital Input FBC. |
| ECB43 | Analog Output FBC. |
| ECB44 | Digital Output FBC. |
| ECB45 | SPECTRUM™ Migration Integrator. |
| ECB46 | Digital Input/Digital Output FBC. |
| ECB47 | FBP10 Interface to Cluster I/O (Parent). |
| ECB48 | SPECTRUM Migration Integrator, Non-Redundant. |
| ECB48R | SPECTRUM Migration Integrator, Redundant. |
| ECB49 | SPECTRUM Migration Integrator (FIO). |
| ECB50 | SPECTRUM Migration Integrator (UFM). |
| ECB51 | SPECTRUM Migration Integrator (UIO). |
| ECB53 | Analog Output. |
| ECB73 | FoxCom™ ECB |
| ECB74 | Intelligent Positioner ECB provides data and configuration information directly from Intelligent Positioners as output to control blocks. |

**Table 1. Foxboro Evo Control Blocks (Continued)**

| | |
|---|---|
| ECB200 | Parent ECB for non-redundant DCI-based FBM types. |
| ECB201 | Field devices connected to DCI-based FBM types. |
| ECB202 | Parent ECB for redundant DCI-based FBM types. |

(a) MVC/MVL blocks for Connoisseur support are not included in the FCP280 images.
(b) ECB6 is only used with the pre-Version 8.0 Sequence of Events package.

## RELATED PRODUCT SPECIFICATION SHEETS

For reference purposes, Table 2 lists the Product Specification Sheets (PSSes) for additional hardware and software elements in the DIN rail mounted subsystem.

**Table 2. Related Product Specification Sheets**

| PSS Number | Title |
|---|---|
| PSS 31H-1B11 B3 | Field Control Processor 280 (FCP280) |
| PSS 21H-2W1 B3 | DIN Rail Mounted Subsystem Overview |
| PSS 21H-2W1 B4 | 100 Series Fieldbus Module Upgrade Subsystem Overview |
| PSS 21S-1C2 B3 | Time Synchronization Overview |
| PSS 21S-2B3 B4 | FoxDraw Display Builder and Configurator |
| PSS 21S-2B5 B4 | I/A Series Configuration Component (IACC) Engineering Package for Windows 7 |
| PSS 21S-2B8 B4 | FoxView Software |
| PSS 21S-3A2 B3 | EXACT Multivariable Control (EXACT MV) |
| PSS 21S-8A3 B3 | Intelligent Field Device Configurator FoxCom and HART® Protocols |
| PSS 21S-8A5 B3 | Overview of HART Device Manager and FoxCom Device Manager |
| PSS 31S-10B3 B3 | Control Editors |
| PSS 31S-10B8 B3 | Logic Block Editor and Troubleshooting Tool |

Invensys
10900 Equity Drive
Houston, TX 77041
United States of America
http://invensys.com

Global Customer Support
Inside U.S.: 1-866-746-6477
Outside U.S.: 1-508-549-2424 or contact
your local Invensys representative.
Website: https://support.ips.invensys.com



MB 031 0414