ControlWave Designer

ControlWave® Designer, from Emerson Process Management, is a fully IEC 61131-3 compliant programming environment for configuration of continuous and discrete control applications. The advantages offered by IEC 61131-3 have made it the emerging international standard programming environment and is currently employed by most all PLC manufacturers. As a result of its standardization and widespread acceptance, it allows controls engineers the flexibility to choose the most appropriate controller hardware system without the necessity of learning a new programming language for each platform. The flexibility of IEC 61131-3 further allows each manufacturer to augment the basic set of functions with product specific higher level functions to take maximum advantage of each product's distinguishing capabilities, yet strictly adhere to the IEC 61131-3 standards.

ControlWave Designer is based on a modern Windows technology, offering standard functionality of zooming, scrolling, customizable toolbars, drag & drop operations, shortcut manager, and dockable windows. Designer is used to create, edit, compile, debug, document and print, simple as well as very complex process control applications.

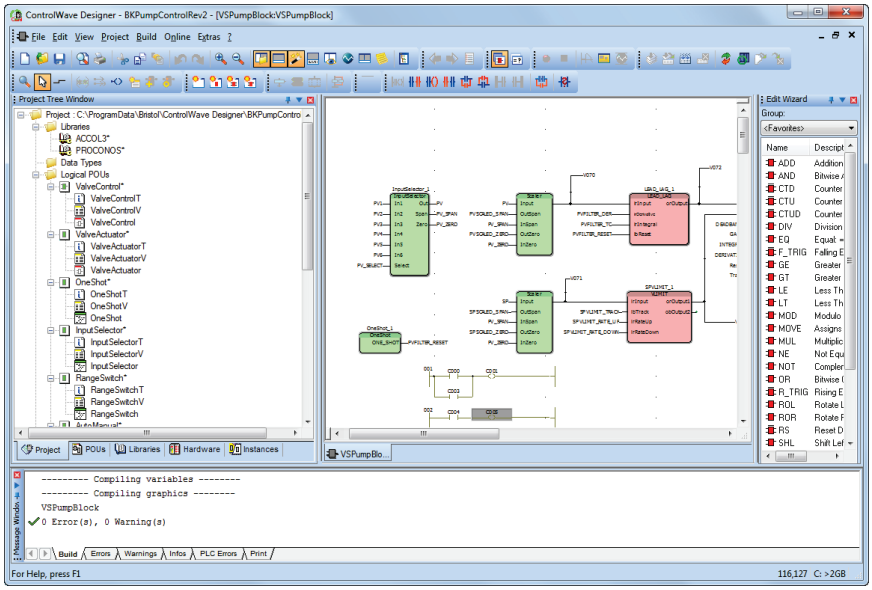
ControlWave Designer includes an extensive library of more than 200 basic IEC 61131-3 functions and function blocks common to many IEC 61131-3 based products.

These include:

- Flip-flops, Counters & Timers
- Ladder diagram functions – coils and contacts, etc.
- Numerical, Arithmetic & Boolean functions – Sine, Cosine, Add, Sub, Square Root, And, Or, etc.
- Selection & Comparison – Min, Max, Greater than, Equal, Less than, etc.
- Type conversions – Integer to Real, Boolean to Word, etc.

ACCOL III Function Block Library Speeds Application Development

In addition to the basic functions and function blocks, ControlWave Designer brings the benefit of over twenty years of SCADA and plant control experience in Emerson's ACCOL III function block library. ACCOL III includes



ControlWave Designer

over sixty function blocks valuable for use in oil and gas, water and wastewater, and process measurement & control applications. Further, ACCOL III is designed to take full advantage of the significant features offered by ControlWave. Briefly, this library includes function blocks for:

- Average, Compare, Totalize
- Scheduling & Sequencing
- PID & Lead/Lag
- AGA gas flow and liquids calculations
- Alarming and Historical Storage
- File handling

Most of the ACCOL III function blocks offer the same functionality available to Network 3000 RTUs, controllers, and flow computers. This greatly simplifies migration to ControlWave for existing Network 3000 users. To further provide forward compatibility, the ACCOL Translator converts most ACCOL functions into ControlWave Designer Structured Text, thus allowing you to leverage previous ACCOL development.

Even without previous ACCOL experience, engineers will appreciate the power of ACCOL III for performing the most common process measurement and control applications.

Alarm System

One of the most important functions of any process automation system is to detect and report alarm conditions. ControlWave Designer with ACCOL III provides the capability of adding any analog or logical variable to an alarm function block to detect, time stamp, store and report analog limit, discrete status, and change of state alarms.

Unlike most other systems, ControlWave detects, time stamps, and stores alarms locally in the controller rather than at the PC level, ensuring that alarms are detected even in the event of a communication failure. Alarms are then reported to the PC once the communication link is re-established. The Alarm Record Queue size is configurable. The default is one hundred alarms. Once full, the oldest alarms will be maintained with time stamp. New alarms will be logged but without a time stamp until some of the older alarm space is freed through reporting. Alarms are time stamped to 1 ms for internal action and reported to the PC with a resolution of 20 ms.

The following alarm blocks and types are supported:

- **Analog Alarm Function Block**
  - Detects High, High-High, Low and Low-Low alarms
  - Independent deadbands are provided for high and low alarms
  - Alarm reports may be enabled or disabled to suppress nuisance alarms
  - Alarm Priority – Event, Operator guide, Non-critical or Critical
  - Alarm descriptor text string
  - Reports single, momentary and multiple alarm conditions since last report
- **Discrete Alarm Function Block**
  - Detects selectable On or Off status
  - Alarm reports may be enabled or disabled to suppress nuisance alarms
  - Alarm Priority – Event, Operator guide, Non-critical or Critical
  - Alarm descriptor text string
  - Reports single, momentary and multiple alarm conditions since last report
- **Change of State Alarm Function Block**
  - Detects discrete change of state from On to Off or from Off to On
  - Alarm reports may be enabled or disabled to suppress nuisance alarms
  - Alarm Priority – Event, Operator guide, Non-critical or Critical
  - Alarm descriptor text string

Reports single, momentary and multiple alarm conditions since last report

Alarms may be transmitted to up to four network connected PCs. Other PCs may obtain the alarm reports from the four declared PCs. At any time the operator may request all current unacknowledged alarms be re-reported. This is particularly useful for PC HMI software products that do not store the system alarm history. In this case the alarm history can be obtained from ControlWave at the operators discretion.

Historical System

Again, most systems rely on the PC to collect and store historical data. But what happens when the data can't be collected due to a communication line failure. ControlWave ensures that important audit and archive data is securely stored, with time stamp, in the controller. To accomplish this, the Designer allows configuration of two special historical function blocks.

The historical data may be logged to either RAM or non-volatile Flash memory. The data is maintained even after a cold start or application downloading.

Archive Function Block

Historical archive files are similar to data arrays, except that each column is directly associated with a particular ControlWave variable, and each column has a descriptive title. The first column of each row contains a time stamp, and the remaining columns in a given row are the variable values or results of calculations based on variable values collected at the time specified by the time stamp. The data

occupying a particular row is referred to as a record; a record index is kept which points to the row containing the current record.

Data storage interval may be 1 minute, 5 minutes, 15 minutes, hourly, or daily. Each file can be up to 64 Kb

OpenBSI Utilities provides facilities to collect this Audit and Archived data, on a scheduled or demand basis, and present them in useful formats including .CSV and ODBC Access formats.

The Audit System is responsible for detecting and logging of selected events and alarms. The Audit system is very application specific. It is often used in applications requiring quality tracking and reporting. Certain alarms, perhaps a high alarm on a pressure measurement, can be recorded in the Audit. Significant events may also be recorded. For example, a maintenance technician might replace a valve or a pump with different operating characteristics.

Audit Function Block

The Audit function blocks allow separate logs for alarms and events. For all alarm signals in the controller, the Audit Function Block generates a one-line message when the alarm signal goes into its alarm state, or when it returns to normal after having been in alarm. An event message is generated for signals that are included in a special signal list. Time stamp resolution for alarm and event message in the Audit storage is 1 ms. Reported alarm resolution is 20 ms. Each log can be up to 64 Kb

Variable Extension Wizard

This utility which runs from within ControlWave Designer allows you to create initialization files (*.INI) which assist in batch configuration of variables within the ControlWave controller.

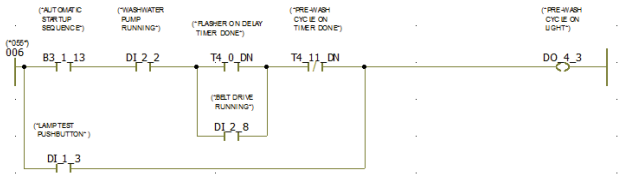
The information in these initialization files is incorporated into the ControlWave project when they are read using either the DB_LOAD or RBE function blocks. The initialization files may be used to:

- **Configure lists** – Identify variables which should be collected via Report by Exception
- **Configure alarms** – Configure descriptive text, ON/OFF text, inhibit/enable flags, or units text

Language Support - Why Settle For Only One?

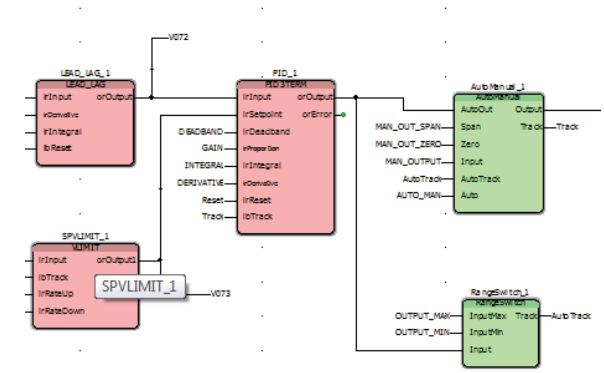
While many IEC 61131-3 programming tools offer only Ladder Diagram or a limited set of languages, ControlWave supports all five languages available in the IEC 61131-3 standard. The first three are graphical programming languages; Ladder Diagram, Sequential Function Chart, and Function Block Diagram. The latter two, Structured Text and Instruction List are text-based languages. Any or all five languages may be used to implement a process control scheme.

Ladder Diagram (LD) employs the elements common to traditional PLCs' such as normally open & normally closed contacts and coils. Ladder is typically used for sequential logic, interlocks and on/off control applications.



Ladder Diagram

Function Block Diagram (FDB) is a graphical programming language that resembles the P&I drawings or circuit diagrams commonly found in process control applications. The graphical representation makes analog control loops visually easy to understand and auto-documents the control scheme.



Function Block Diagram

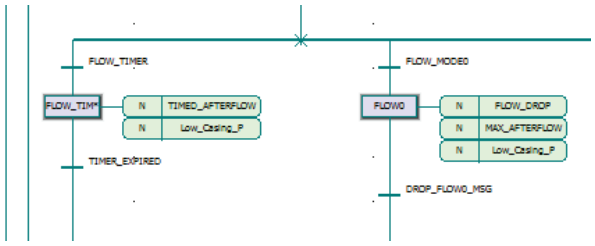
Function Block Diagram programs also allow the use of Ladder Logic in the same program. With Designer, you can even have Ladder objects as inputs to FB and outputs from the FB objects. User Defined Function Blocks can be used in Function Block programs as well.

Structured Text (ST) is a high level text based language containing all the elements of a modern programming language. As well as providing a convenient method of including the library of function blocks and creating custom function blocks, it allows IF-THEN-ELSE and other conditional branching statements.

```
IF (bInitCommList = TRUE) THEN
    bInitCommList := FALSE;

    commissionList[1].iostrTagName    := 'DEV_0001';      (* Tagname in wireless transmitter.  *)
    commissionList[1].ibDecommission  := FALSE;           (* Commission the device when found. *)

    commissionList[2].iostrTagName    := 'DEV_0002';
    commissionList[2].ibDecommission  := FALSE;

    commissionList[3].iostrTagName    := 'DEV_0003';
    commissionList[3].ibDecommission  := FALSE;

    commissionList[4].iostrTagName    := 'DEV_0004';
    commissionList[4].ibDecommission  := FALSE;
                                      (* '12345678901234567890123456789012' *)
    commissionList[5].iostrTagName    := 'LONG DEVICE TAGNAME 32CHARACTERS';
    commissionList[5].ibDecommission  := TRUE;            (* Device is commissioned only after
                                                           * this is changed to FALSE. *)

END_IF;
```

Structured Text

Sequential Function Chart (SFC) allows sequential operations to be programmed in a graphical manner similar to a flow chart. The steps represent the actions, which can be performed in sequence or in parallel, and the transitions represent the conditions that must be completed in order to advance to the next step.



Sequential Function Chart

Instruction List (IL) is also a text-based language similar to assembly language and as such it is somewhat less popular than the other four languages.

```
1    LD   %IX0.2 (* direct variables *)
2    AND  %IX0.3
3    OR   Action_INIT
4    ST   IL_VAR
5
6    LD   Input_IX0_0
7    JMPC MANUAL
8
9    (* Timer FB TON *)
10   LD   Timer_start
11   ST   TON_IL.IN
12   LD   PT_TON_IL
13   ST   TON_IL.PT
14   CAL  TON_IL
15   LD   TON_IL.Q
16   ST   Action_INIT
```

Instruction List

User Defined Function Blocks – Reusable Applications

With ControlWave Designer, once a program containing one or more Functions or Function blocks is created and tested, it can be saved to a User Defined Function Block Library. Then, this new User Defined Function Block (object) can be reused any number of times and can even be transported to other projects. The User Defined Function Blocks have reassignable inputs and outputs so each instance of the function block can have different variables attached to the terminals. Each new project can draw on previous development to significantly reduce implementation time. Engineers have found this object oriented programming concept, available in Designer, to be an extremely valuable tool in application development.

Multitasking Operation

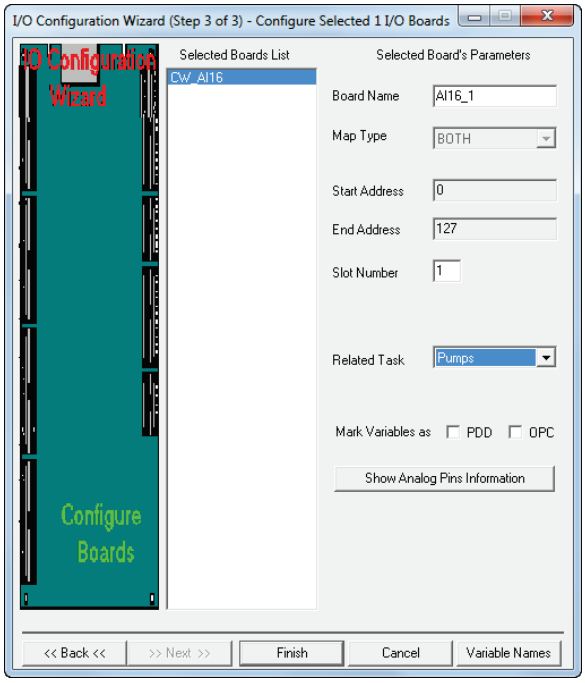
ControlWave runs in a Multitasking environment where program execution can be scheduled and prioritized. While ControlWave scan times are extremely fast, multitasking capability gives you control over critical task execution.

Program Security

For program security, the entire program, all associated variables, and graphics can be zipped into a single file. This project file can then be downloaded, stored in ControlWave memory, and later uploaded for recovery or modification by the Designer software. All variable names, graphics, comments and page layouts are uploaded with the application program so that the uploaded project is identical to the original downloaded file. In addition, the zipped project source is automatically downloaded along with the application executable to ensure synchronization between the source and the executing application program.

I/O Configuration Wizard

The ControlWave Designer I/O Configuration Wizard simplifies the connection between the IEC 61131-3 application program and the physical ControlWave local I/O and Remote I/O modules. While I/O assignment can also be accomplished in the I/O_Configuration section of IEC 61131-3, the ControlWave Designer I/O Configuration Wizard provides a self-prompting point and click menu system to simplify programming and eliminate syntax errors.



I/O Configuration Wizard

System Variable Wizard

Numerous system variables are maintained in ControlWave and the Designer System Variable Wizard gives the user access to them by simply selecting those desired in the Wizard menu.

System variables are used for:

- Control start up and power up execution and control
- Monitoring the efficiency of task execution and start up
- Accessing the system date and time
- Viewing diagnostic information about the communication ports

A system variable is also available to provide a port data line monitor. By viewing the variable, an engineer can view the communication message stream through the port. This is an extremely powerful feature available to those familiar with data line monitor capability.

On-line Program Debug Tools

No matter how experienced the engineer, there is always a need for powerful software debug tools. ControlWave Designer offers a comprehensive set of on-line tools to help you test, analyze and troubleshoot your application program. Real-time data values and program execution are displayed for all five languages.

The Variables Cross Reference list contains all variables, function blocks, actions, transitions, steps, jumps, labels and connectors which are used within the current project. It is a helpful tool for debugging and fault isolation.

The Logic Analyzer is a powerful tool for recording values of variables over a certain time interval. They are displayed graphically in the Logic Analyzer window. All recorded values and settings of the Logic Analyzer are stored automatically with the project.

The Watch Window can be used to collect variables from different worksheets to gain an understanding about how these variables work together. In addition you can use the watch window to force and overwrite variables.

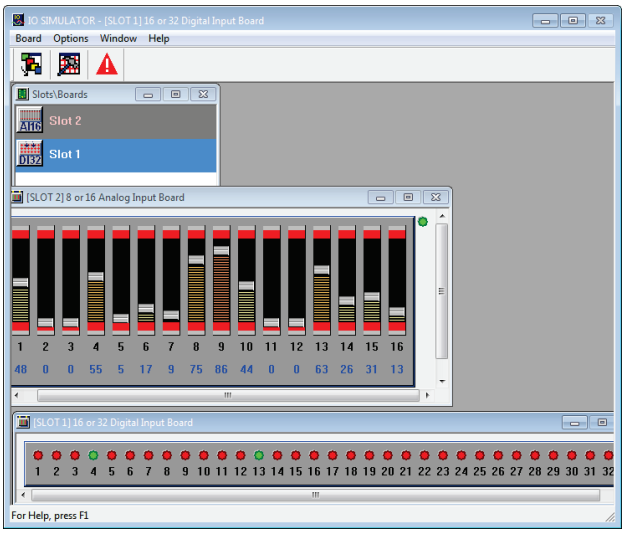
Breakpoint, Single Step and Single Cycle program execution provides a step and trace function that is extremely useful for program debugging. These functions allow you to continue the program execution line by line after a breakpoint has been reached.

Power Flow

For graphical worksheets, IL and ST worksheets, you can switch from the Variable Status to the Address Status with Powerflow and vice versa. Powerflow displays which program parts are actually executed and which ones not. This is useful for debugging worksheets with conditional jumps.

I/O Simulator

The ControlWave Designer I/O Simulator allows the Application load program to be tested on a PC, with simulated analog and digital inputs and outputs.



I/O Simulator

The I/O Simulator utilizes the identical real time operating system used in the ControlWave unit; this allows initial I/O testing and debugging to be performed in a safe, isolated environment, without the need for a running ControlWave unit and process I/O boards.

ObjectServer Database Open Connectivity

Variables in ControlWave Process Automation Controllers can be accessed using ObjectServer software. All variables marked as "CSV" in the variable declaration page are included in an ASCII file generated by the OpenBSI Signal Extractor utility. This means that you only create one database, using ControlWave Designer, then automatically construct the ObjectServer database from that source. Now any OPC compliant client HMI or SCADA application has access to ControlWave variables and alarms. By adhering to open standards, ControlWave simplifies this process and your life.

Specifications

- CPU: 2 GHz Pentium 4 class PC (recommended)
- RAM: 512 MB (Windows XP) or 1 GB (Windows 7)
- Disk Space: 1 GB free space
- Comm: RS 232 or 10/100 MHz Ethernet
- Display: True Color
- Drive: CD-ROM
- Operating System: Windows XP (SP3) or Windows 7

ACCOL III Function Blocks

Function Block	Description
AGA 3, AGA3DENS, AGA3I, AGA3SELECT, AGA3TERM, AGA5, AGA7, AGA8DETAIL, AGA8GROSS, AGA10	Natural gas flow calculations for orifice and turbine measurement complicant with the appropriate American Gas Association specifications.
ALARM	Analog, logical and change of state alarm detection and reporting.
ANOUT	Provides anit-reset windup track and reset feedback to PID controller.
ARCHIVE	Allows variables to be time-stamped and stored in historical Archive files as raw or pre-calculated values.
AUDIT, AUDIT_SELECTED	Provides a history of significant events for selected process variables and alarm variables.
AUTOADJUST	Performs adjusted volume calculations and self-check calculations for an Invensys (Equimeter) Auto-adjust Turbine Meter.
AVERAGER	Computes the time-average and integral of an analog or logical input.
BTI	Allows CW_10, CW_30, and CW_35 controllers with the BBTI board to collect data from the Bristol Teletrans™ Model 3508 Transmitter.
CLIENT/SERVER	Provides peer-to-peer and master/slave communication between ControlWave nodes
COMMAND	Provides a flexible start-up control for a wide range of equipment such as motors, valves or pumps. This Function Block initiates a programmable delay period between an input and output change.
COMPARATOR	Compares two analog inputs to determine selection of an output.
CRC	Allows the user to calculate a Cyclic Redundancy Code (CRC) for a given data array of type USINT. Used to provide CRC checking with the Generic Serial Protocol Function Block.
CUSTOM	Allows the user to configure an Emerson controller to communicate with other manufacturer's programmable logic controllers and networks.
DACCUMULATOR	Allows certain arithmetic operations to be performed on double-precision floating point numbers.
DB_LOAD	Loads variable information from text files. These files are created in the resource directory of the application and downloaded automatically as part of the project. DB_LOAD supports large variable lists and minimizes memory requirements.
DEMUX	Performs a demultiplex operation by sending an input value to a selected location in an output list.
DIAL_CTRL	Allows establishing a communication link between two compatible modems on the public telephone system network. At present only the external modems are supported. The Modem interface is independent and not part of any communication protocol. This allows the DIAL_CTRL FB to perform dialing / modem setup operations independent of the underlying communication protocol, BSAP, Custom, Generic Serial, etc., at the target port. Application has full control over the dial/hang-up and/or modem setup functions.
DIFFERENTIATOR	Calculates the rate of change of an input.
DISPLAY	Used to configure the Bristol 2 button and 25 button keypad and display connected to a controller. This will allow an operator to view/change signal data or be allowed to scroll through lists of signal data based upon their login privileges.

Function Block	Description
ENCODE	This Function Block will do the following: ▪ Convert packed Julian date/time to REAL values. ▪ Convert REAL values to packed Julian date/time. ▪ Convert system date/time to REAL values.
EVP	Calculates the equilibrium vapor pressure for a liquid.
FIELDBUS	Interfaces with FFbus devices via a bridge server.
FILE	Provides program control to open, close, delete, read and write files.
FPV	Computes the super compressibility factor (Fpv) of a gas measured in accordance with the American Gas Association Report No. NX-19.
FUNCTION	Performs a table lookup based on array row and column inputs.
GENERIC_SERIAL	Allows the engineer to write custom serial protocols to third party devices through message read and write.
GPA8173	Converts the mass of natural gas liquids to equivalent liquid volumes at base conditions.
GSV	Converts the gross standard volume for a liquid.
HART	Interface to HART® field devices via serial port or I/O board.
HILOLIMITER	Compares the value of an input to a predefined min and max.
HILOSELECT	Selects the largest and smallest values from a set of inputs.
HWSTI	Honeywell Smart Transmitter Interface.
IEC62591	Allows a ControlWave Micro controller with an IEC62591 Interface module to communicate to IEC62591 wireless devices.
INTEGRATOR	Computes an integral approximation of an input value.
ISO5167	Calculates flow rate for Orifice plates, Nozzles, Venturi tubes, and Venturi-nozzle Primary Devices as specified in the International Organization for Standardization's International Standard ISO 5167-1980 (E), 1980 edition.
LEAD_LAG	Performs a controlled delay on changes based on a given input. Module can also be use for simulation of process lag.
LIQ	Liquids measurements for density 'A' and temperature 'B' tables 5, 6, 23, 24, 53, 54, 59 and 60. Also, F1 and F1M compressibility.
LIST_ELE_NAME	Fetches the variable name for a given list element.
LIST10, 20, 30, 50 & 100	This is a list of up to 10, 20, 30, & 50 input or output variables.
MUX	Performs a multiplex operation by selecting from a group of inputs, and applying that selected input to an output.
PDO	Allows the user to produce pulses of variable width (Pulse Duration Output) at the digital output field wiring terminals.
PID3TERM	Provides a process control algorithm which allows proportional (P), proportional/integral (PI) or proportional/integral/derivative (PID) modes.
PORT_CTRL	Allows the user to manually control the Data Terminal Ready (DTR) output of a selected communications port. Setting DTR to FALSE will disable the selected port and place it in low power mode.
R_INT	Truncates a value to an integer, eliminating any fractional Portion.
R_RND	Rounds an input value up to the next highest integer.
RBE	Report by Exception

Function Block	Description
REDUN_SWITCH	Performs a software controlled fail-over to the Redundant Standby unit in a redundant system.
REG_ARRAY	Allows the user to register arrays for the purpose of retrieval by the external HMI systems.
SCHEDULER	Block is used to equalize the elapsed running time of a number of external devices such as pumps and motors. It provides a device work list that can be scheduled by one of four algorithms that determine the sequence operation.
SEQUENCER	Block is used in applications where a number of operations must be performed on a repetitive or sequential basis such as valve sequencing. The function block accepts a list of up to 255 BOOL inputs and provides output to a corresponding list of BOOL outputs. These inputs and outputs can represent valve patterns.
STEPPER	The Stepper Module performs a series of sequential operations on a number of outputs (1-255). It is particularly useful in applications where external equipment must be activated in sequence for certain fixed time periods, such as during water filter backwash operations.
TCHECK	Provides status checking and data processing for a 3508 TeleTrans™ Transmitter.
TOT_TRND	Totalizes a scaled input value over a set of fixed time intervals.
USERS_ACTIVE	Returns information on all currently signed-in users.
USERS_DEFINED	Allows encrypted access to the security configuration of the ControlWave.
VAR_ATTRIB, VAR_CI	Provides software control of the Manual, Inhibit and Alarm attributes.
VIRT_PORT	Provides serial port expansion through a terminal server.
VLIM	Limits the rate of change of an output.
VMUX	Permits the user to retrieve the value of a variable from a list of variables and modify the output until it equals the input.
XMTR	Provides read/write access to the memory of a TeleTrans Transmitter, or other compatible device.

Headquarters:
Emerson Process Management
Remote Automation Solutions
6005 Rogerdale Road
Houston, TX 77072 U.S.A.
T +1 281 879 2699 | F +1 281 988 4445
www.EmersonProcess.com/Remote

Europe:
Emerson Process Management
Remote Automation Solutions
Unit 8, Waterfront Business Park
Dudley Road, Brierly Hill
Dudley UK DY5 1LX
T +44 1384 487200 | F +44 1384 487258
www.EmersonProcess.com/Remote

North American/Latin America:
Emerson Process Management
Remote Automation Solutions
6005 Rogerdale Road
Houston TX USA 77072
T +1 281 879 2699 | F +1 281 988 4445
www.EmersonProcess.com/Remote

Middle East/Africa:
Emerson Process Management
Remote Automation Solutions
Emerson FZE
P.O. Box 17033
Jebel Ali Free Zone – South 2
Dubai U.A.E.
T +971 4 8118100 | F +971 4 8865465
www.EmersonProcess.com/Remote

Asia-Pacific:
Emerson Process Management
Remote Automation Solutions
1 Pandan Crescent
Singapore 128461
T +65 6777 8211| F +65 6777 0947
www.EmersonProcess.com/Remote