Honeywell

Experion PKS
Bristol Babcock OpenBSI Interface Reference

EPDOC-XXX9-en-431A
February 2015

Release 431

Honeywell

Document	Release	Issue	Date
EPDOC-XXX9-en-431A	431	0	February 2015

Disclaimer

Contents

Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers 5
Devices supported by the Bristol Babcock OpenBSI interface 7
Other documentation for Bristol Babcock OpenBSI 8
Bristol Babcock OpenBSI-specific terms 9
Architectures for Bristol Babcock OpenBSI 10
Installing OpenBSI tools on the Server 10
Communication settings for Bristol Babcock OpenBSI 11
Configuring the Bristol Babcock node 12
Configuring the network address 12
Defining the network 12
Changing the network address 12
Changing OpenBSI configuration 14
Bristol Babcock OpenBSI channel and controller reference 15
Main properties for a Bristol Babcock OpenBSI channel 16
Main properties for a Bristol Babcock OpenBSI controller 17
Bristol Babcock OpenBSI controller types 20
Optimizing Bristol Babcock OpenBSI scanning performance 21
Reporting by exception 21
Understanding polling 21
Bristol Babcock OpenBSI scan packets 21
Bristol Babcock OpenBSI points reference 23
Defining a Bristol Babcock OpenBSI address for a point parameter 24
Address syntax for Bristol Babcock OpenBSI controllers 24
Point addresses supported by Bristol Babcock OpenBSI 25
Bristol Babcock OpenBSI EFM meter templates reference 29
Main properties for a Bristol Babcock OpenBSI meter template 30
Configuration Log properties for a Bristol Babcock OpenBSI meter template 32
Interval Log properties for a Bristol Babcock OpenBSI meter template 33
Daily Log properties for a Bristol Babcock OpenBSI meter template 34
Alarm and Event properties for a Bristol Babcock OpenBSI meter template 35
Ultrasonic Log properties for a Bristol Babcock OpenBSI meter template 36
Composition Log properties for a Bristol Babcock OpenBSI meter template 37
Gas Quality Log properties for a Bristol Babcock OpenBSI meter template 38
Data Export properties for a Bristol Babcock OpenBSI meter template 39
Bristol Babcock OpenBSI Application Programming Interface 41
Troubleshooting Bristol Babcock OpenBSI issues 43
Troubleshooting Bristol Babcock OpenBSI point configuration errors 44
Testing Bristol Babcock OpenBSI communications with the server 45
Troubleshooting Bristol Babcock OpenBSI scanning errors 46
Confirming resolution of Bristol Babcock OpenBSI point addresses 47
Notices 49
Documentation feedback 50
How to report a security vulnerability 51

Support 52
Training classes 53

Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers

This reference describes how to set up, configure, and test Bristol Babcock OpenBSI controller communications with the server.

Revision history

Revision	Date	Description
A	February 2015	Initial release of document.

How to use this guide

The following steps show the order in which the controller interface should be configured. Complete each step before starting the next.

Steps for connecting and configuring a Bristol Babcock OpenBSI controller.

Steps	Go to
Connect and set up the Bristol Babcock controller according to the instructions in the controller user manual.	
Install OpenBSI tools on the server	Installing OpenBSI tools on the server
Configure Bristol Babcock nodes using OpenBSI utilities	Configuring the Bristol Babcock node
Use Quick Builder to define channels	• Bristol Babcock OpenBSI channel and controller reference • *Quick Builder User's Guide*
Use Quick Builder to define controllers	• Bristol Babcock OpenBSI channel and controller reference • *Quick Builder User's Guide*
Download channel and controller definitions to the server	*Quick Builder User's Guide*
Enable channels and test communications	Testing Bristol Babcock OpenBSI communications with the server
Troubleshoot communication errors	Troubleshooting Bristol Babcock OpenBSI scanning errors
Use Quick Builder to define points	Defining a Bristol Babcock OpenBSI address for a point parameter

Related topics

"Devices supported by the Bristol Babcock OpenBSI interface" on page 7
"Other documentation for Bristol Babcock OpenBSI" on page 8
"Bristol Babcock OpenBSI-specific terms" on page 9
"Architectures for Bristol Babcock OpenBSI" on page 10
"Communication settings for Bristol Babcock OpenBSI" on page 11

“Configuring the Bristol Babcock node” on page 12
“Changing OpenBSI configuration” on page 14
“Installing OpenBSI tools on the Server” on page 10
“Bristol Babcock OpenBSI channel and controller reference” on page 15
“Testing Bristol Babcock OpenBSI communications with the server” on page 45
“Troubleshooting Bristol Babcock OpenBSI scanning errors” on page 46
“Defining a Bristol Babcock OpenBSI address for a point parameter” on page 24

Devices supported by the Bristol Babcock OpenBSI interface

Generally, the server supports any device that communicates via the *Bristol Babcock Synchronous Asynchronous* (BSAP) communication protocol and fits in the OpenBSI architecture.

The server supports the following Bristol Babcock models:

- DPC 3330
- ControlWave (signal name limited to 20 characters)

Consult the *Software Change Notice* or *Software Release Bulletin* for supported versions of the OpenBSI API.

Attention

The ControlWave RTU contains a setting **RDB - Name in ACCOL format** (signal _USE_ACCOL_NAME) that which will translate all of the ControlWave signal names to a legacy naming format.

For example, *@GV._TIME_007* becomes *#TIME.007*.

If this flag is set for the ControlWave, then all points built for these signals should also use the legacy naming format.

Other documentation for Bristol Babcock OpenBSI

The following document describes the address space of all the devices supported by this interface in great detail. It also provides information about the physical configuration of the controllers. Reading this document is also useful when installing the interface.

- *DPC 3330 Instruction Manual*

The following documents might also be useful when installing and configuring the DPC 3330:

- *Network 3000 Communication Application Programmer's Reference*
- *Bristol Babcock ACCOL Reference and OpenBSI Utilities Binders*
- *ACCOL II Interactive Compiler (AIC) Manual*
- *ACCOL II Reference Manual*

From Honeywell:

- *Software Installation User's Guide*
- *Hardware and Point Build Reference*

Bristol Babcock OpenBSI-specific terms

ACCOL	Advanced Communications & Control Oriented Language
API	Application Programming Interface
BB	Bristol Babcock
BSI	Bristol Systems Interface
LocalView	Bristol Utility for configuring RTU over a serial connection
NetView	Bristol Utility for configuring RTU over an Ethernet connection
NRT	Node Routing Table. Configuration table for network topology.
PPP	Point to Point Protocol
RBE	Report By Exception
RDB	Remote DataBase
RTU	Remote Terminal Unit

Architectures for Bristol Babcock OpenBSI

The following architectures are possible for Bristol Babcock OpenBSI:

- Ethernet using Internet Protocol
- PPP using Internet Protocol
- Serial connection (direct cable)
- Serial connection (Terminal Server)

Only Internet Protocol has been qualified for a redundant server configuration. Serial connections are only supported for non-redundant servers.

The RS-562 connection is between a serial port on the server and the serial interface on the Bristol Babcock device. The BSAP protocol is fully described in the Network 3000 Communications Application Programmer's Reference.

The OpenBSI interface requires that all connections are defined and configured with the OpenBSI tools. These connections are transparent to the server. See Bristol Babcock OpenBSI Utilities for details on configuring communications.

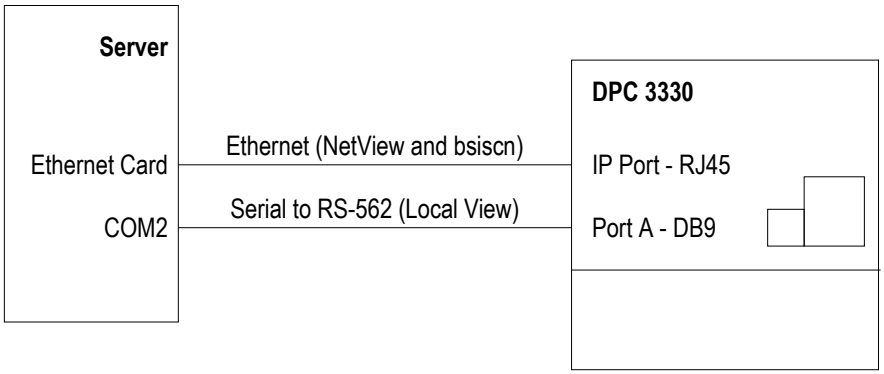

Figure 1: Bristol Babcock OpenBSI Interface System Architecture

Installing OpenBSI tools on the Server

Some OpenBSI Tools are required to run on the server platform. These are:

- LocalView (configures network address)
- Workbench (produces ACCOL load file)
- BSAP to IP Setup (maps a Terminal Server port to a local COM port)
- NetView (configures RTU, produces NRT, downloads ACCOL load)
- DataView (reads signal values from RTU)

Install these tools in *\OpenBSI*. OpenBSI tools can be started from the Windows Explorer or a command prompt from any working folder, for example *\OpenBSI\NetView*, *\OpenBSI\DataView*. Opening a saved configuration file will start the appropriate tool.

The OpenBSI tools must be used to configure all Bristol Babcock hardware. Once this is done, use Quick Builder to configure the server hardware and points.

Related topics

“Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers” on page 5

Communication settings for Bristol Babcock OpenBSI

LocalView uses a local cable connection to the DPC Port on the Bristol Babcock node. “Figure 2: RS-562 connection using D9 to D9” shows the pinouts for an RS-562 connection to a server-end D9 serial port. The DPC 3330 uses a RS-562 9-pin D connector.

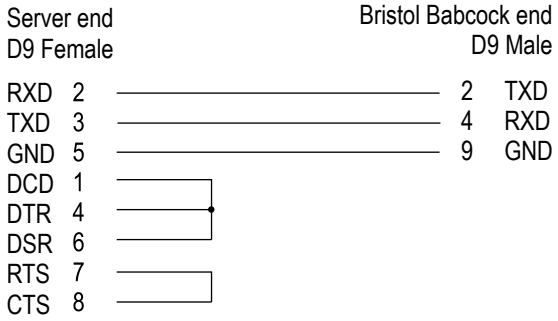

Figure 2: RS-562 connection using D9 to D9

“Figure 3: RS-562 connection using D9 to D25” shows the pinouts for an RS-562 connection to a server-end D25 serial port.

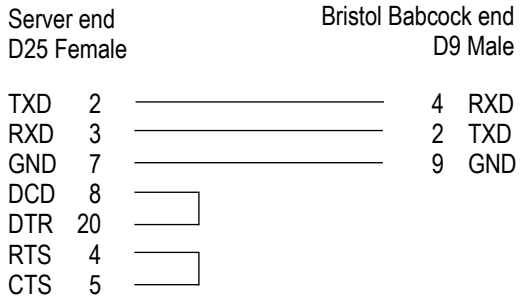

Figure 3: RS-562 connection using D9 to D25

Configuring the Bristol Babcock node

Related topics

“Configuring the network address” on page 12
“Defining the network” on page 12
“Changing the network address” on page 12

Configuring the network address

A bootstrap process is required to configure the network address for the Bristol Babcock node. This is achieved by using **LocalView** to set up the communication ports. This is done before you download an ACCOL load to the Bristol Babcock device. See the OpenBSI Utilities manuals for instructions on using **LocalView**.

To configure and download a Bristol Babcock node

1. Use **Workbench** to create the load.
2. Use **NetView** to create the node configuration. Set the load file to that created in step 1.
3. Use **LocalView** to configure the communication ports. Set the node configuration to that created in step 2. Once the network address configuration has been saved to the node, the node must be reset.
4. Use **NetView** for further configuration, and download the load file to the node.

> **Attention**
> Step 3 is the bootstrap process.

Defining the network

Once the network address for the node has been configured and downloaded, and the node has been reset, **NetView** can be used for additional configuration tasks.

NetView contains the network topology, and generates the Node Routing Table (NRT). Every IP node in the network can be configured from a single instance of **NetView**.

> **Attention**
> **LocalView** requires a direct cable connection to the Bristol Babcock node. **NetView** can connect over a standard Ethernet network, and can be used in conjunction with other OpenBSI utilities.

Changing the network address

NetView can be used to change the communication port details of an existing configured RTU, including the RTU’s IP configuration. See the OpenBSI Utilities manuals for details of using **NetView**.

To change the network configuration of a Bristol Babcock node

1. Start **NetView**, sign in for configuration changes, select the RTU, and select **RTU** > **RTU Configuration Parameters** from the right-mouse button menu.
2. Click **Load from RTU** to upload the current RTU communication port configuration.
3. Change the communication configuration as desired.
4. Click **Save to RTU** to send the new configuration to the RTU.
5. Click **Close**.
6. In **NetView**, select the RTU, right-click to display the floating menu and then select **Properties**.

7 Change the Primary and/or Secondary IP Address as required, click **OK**.

8 Reset the RTU. When the RTU has completed its startup sequence, use **NetView** to download its load file again.

Attention

If the IP address of the RTU is changed, the change will only take effect when the RTU is reset. The RTU properties in **NetView** must be changed before downloading the load file again.

Changing OpenBSI configuration

Using NetView to change OpenBSI configuration can only occur when the Experion server is stopped.

If you have redundant servers, use the following procedure to change OpenBSI configuration.

Prerequisites

- Server A is running as the primary server.
- Server A and Server B are synchronized.

To change OpenBSI configuration on redundant servers

1. Stop Server B.
2. Using NetView, change the OpenBSI configuration on Server B.
3. Close NetView to allow scan tasks to connect to OpenBSI.
4. Start Server B.
5. Synchronize the servers.
6. Manually failover from Server A to Server B.
   Server B is now the primary server.
7. Repeat steps 1 to 5 on Server A.
8. Synchronize the servers.
9. (Optional) Manually failover from Server B to Server A.
   Server A is now the primary server.

Related topics

“Main properties for a Bristol Babcock OpenBSI controller” on page 17

Bristol Babcock OpenBSI channel and controller reference

This section describes the configuration and addressing information specific to Bristol Babcock OpenBSI channels and controllers.

In addition to the information contained in this reference, and for help to build channels and controllers, see the section titled "Building controllers or channels" in the *Quick Builder User's Guide*.

Related topics

“Main properties for a Bristol Babcock OpenBSI channel” on page 16
“Main properties for a Bristol Babcock OpenBSI controller” on page 17
“Bristol Babcock OpenBSI controller types” on page 20
“Optimizing Bristol Babcock OpenBSI scanning performance” on page 21
“Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers” on page 5

Main properties for a Bristol Babcock OpenBSI channel

The Main tab defines the basic properties for a Bristol Babcock OpenBSI channel.

For information about how to create a channel, see the topic titled "Building controllers and channels" in the *Quick Builder User's Guide*.

Property	Description
Name	The unique name of the channel. A maximum of *10* alphanumeric characters (no spaces or double quotes). Note: In Station displays, underscore characters (_) appear as spaces.
Description	(Optional) A description of the channel. A maximum of *132* alphanumeric characters, including spaces.
Associated Asset	The Tag Name of the Asset to be associated with the alarm group.
Marginal Alarm Limit	The communications alarm marginal limit at which the channel is declared to be marginal. When this limit is reached, a high priority alarm is generated. To change the priority of the alarm system wide, see the topic titled "Configuring system alarm priorities" in the *Server and Client Configuration Guide*. To change the priority of the alarm for one channel, see the topic titled "About configuring custom system alarm priorities for an individual channel or controller" in the *Server and Client Configuration Guide*. A channel barometer monitors the total number of requests and the number of times the controller did not respond or response was incorrect. The barometer increments by two or more, depending on the error, and decrements for each good call. To calculate an acceptable marginal alarm limit, use the formula: Square root of the number of controllers on the channel × Marginal Alarm Limit defined on those controllers (Normally, you specify the same value for all controllers on a channel). For example, if there are 9 controllers on the channel and their Marginal Alarm Limit is set to *25*, the value would be *3* (which is the square root of 9) *x 25 = 75*.
Fail Alarm Limit	The communications alarm fail limit at which the channel is declared to have failed. When this barometer limit is reached, an urgent alarm is generated. To change the priority of the alarm system wide, see the topic titled "Configuring system alarm priorities" in the *Server and Client Configuration Guide*. To change the priority of the alarm for one channel, see the topic titled "About configuring custom system alarm priorities for an individual channel or controller" in the *Server and Client Configuration Guide*. Set this to double the value specified for the channel Marginal Alarm Limit.
Exception Poll Period	The default value is *5* seconds. All controllers will be put on an exception poll for the scan period closest to this value. See the topic titled "Optimizing Bristol Babcock OpenBSI scanning performance" for more information.
Item Type	The type of channel specified when this item was created.
Last Modified	The date and time the channel properties were modified.
Last Downloaded	The date and time the channel was last downloaded to the server.
Item Number	The unique item number currently assigned to this channel, in the format *CHNcc*, where *cc* is the channel number. You can change the item number if you need to match your current server database configuration. The number must be between *01* and the maximum number of channels allowed for your system. For more information about setting the maximum value, see the topic titled "Adjusting sizing of non-licensed items" in the *Supplementary Installation Tasks Guide*.

Related topics

"Optimizing Bristol Babcock OpenBSI scanning performance" on page 21

Main properties for a Bristol Babcock OpenBSI controller

The Main tab defines the basic properties for a Bristol Babcock OpenBSI controller.

For information about how to create a controller, see the topic titled "Building controllers and channels" in the *Quick Builder User's Guide*.

Attention

Whenever you add a new Bristol Babcock OpenBSI controller to the network, you need to change the OpenBSI configuration using NetView. See the topic titled "Changing OpenBSI configuration" for more information.

Property	Description
Name	The unique name of the controller. A maximum of *10* alphanumeric characters (no spaces or double quotes). Note: In Station displays, underscore characters (_) appear as spaces. For LAN–connected controllers, the name must not contain underscore (_) characters. This name is used to look up the controller in NetView if you do not specify a **Node Name**.
Description	(Optional) A description of the controller. A maximum of *132* alphanumeric characters, including spaces.
Associated Asset	The Tag Name of the Asset to be associated with the alarm group.
Channel Name	The name of the channel on which the controller communicates with the server. (You must have already defined a channel for it to appear in this list.)
Marginal Alarm Limit	The communications alarm marginal limit at which the controller is declared to be marginal. When this limit is reached, a high priority alarm is generated. To change the priority of the alarm system wide, see the topic titled "Configuring system alarm priorities" in the *Server and Client Configuration Guide*. To change the priority of the alarm for one controller, see the topic titled "About configuring custom system alarm priorities for an individual channel or controller" in the *Server and Client Configuration Guide*. A controller barometer monitors the total number of requests and the number of times the controller did not respond or response was incorrect. The barometer increments by two or more, depending on the error, and decrements for each good call. The default value is *25*.
Fail Alarm Limit	The communications alarm fail limit at which the controller is declared to have failed. When this barometer limit is reached, an urgent alarm is generated. To change the priority of the alarm system wide, see the topic titled "Configuring system alarm priorities" in the *Server and Client Configuration Guide*. To change the priority of the alarm for one controller, see the topic titled "About configuring custom system alarm priorities for an individual channel or controller" in the *Server and Client Configuration Guide*. Set this to double the value specified for the controller Marginal Alarm Limit. The default is *50*.
Dynamic Scanning Fastest Scan Period	Select the **Dynamic Scanning** check box to enable dynamic scanning of all point parameters on this controller. The default setting for this check box is selected. Define the fastest possible scan period (in seconds) that dynamic scanning will scan point parameters on this controller. The default is *15* seconds. The dynamic scanning period does not affect the static scanning rate for a parameter. For example, if the scanning rate for a parameter is 10 seconds, and the dynamic scanning rate for the controller is 15 seconds, the parameter will still be scanned at a period of 10 seconds.

Property	Description
Controller Type	The controller type, either: • SignalsOnly • ArraysOnly • ControlWave See the topic titled "Bristol Babcock OpenBSI controller types" for more information.
Node Name	Sixteen character node name assigned via the Bristol Babcock NetView software program.
Transaction Timeout	Transaction table timeout in seconds. Default is *60*. This timeout value is the length of time that the server will wait for an RDB response. It should be set to the worst case RDB round trip time between the server and the controller. This will be affected by poll periods of intermediate nodes, baud, and so on.
Sync Enable	Enables daily time synchronization of the controller to the server. The default is *Enable*.
Sync Time	Synchronization time for the Controller. Default is zero. This value is the number of minutes after midnight that time synchronization is triggered.
RDB Security Level	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. RDB security level. Default setting is *F*. Not normally changed.
RBE Scan Rate	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. RBE scan period in tenths of seconds. Default is *300*.
RBE Scan Slice	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. RBE scan slice parameter. Default is *1*.
RBE Stop Xmit	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. RBE stop transmit count. Default is *10*.
RBE Timeout	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. RBE timeout between successive wait_init messages in tenths of seconds. Default is *600*.
RBE Inhibit	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. Check to Inhibit use of Bristol Babcock report by exception messages.
Alarm Inhibit	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. Check to Inhibit use of Bristol Babcock alarm messages.
SOE Inhibit	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. Check to Inhibit insertion of Bristol Babcock alarms into the Sequence of Events file.
No RBE Init Reports	This property is only visible if SignalArray or ControlWave is selected as the Controller Type. Flag to skip RBE initialization. RDB initialization is then required to resolve MSD to point addresses.
Item Type	The type of controller specified when this item was created.
Last Modified	The date and time the controller properties were modified.
Last Downloaded	The date and time the controller was last downloaded to the server.

Property	Description
Item Number	The unique item number currently assigned to this controller, in the format *RTUnnnnn*. You can change the item number if you need to match your current server database configuration. The number must be between *01* and the maximum number of controllers allowed for your system. For more information about setting the maximum value, see the topic titled "Adjusting sizing of non-licensed items" in the *Supplementary Installation Tasks Guide*.

Related topics

“Changing OpenBSI configuration” on page 14
“Bristol Babcock OpenBSI controller types” on page 20

Bristol Babcock OpenBSI controller types

Data within a Bristol Babcock OpenBSI device is divided into two types: *signals* and *arrays*. Separate controllers must be defined in the server to address signal data and array data.

If you are collecting EFM data, you can collect from arrays using a SignalsOnly or ControlWave controller type. You do not need to use an ArraysOnly controller type. When configuring EFM, the controller selection determines the address types for the Controller sourced configuration log fields. EFM can also collect from archive and audit data types.

Each controller type has addressing limitations as indicated:

Controller Type	Server Addressable Range
SignalsOnly	Up to 1,485 signals per server controller
ArraysOnly	Up to 3,192 array elements per server controller
ControlWave	Up to 1,485 signals per server controller

If this array addressing capability is not adequate, extra server controllers can be configured to access the array data.

Attention
Do not define more than one server signal controller per physical Bristol Babcock controller. (Physical controllers are configured as RTUs in NetView.)

Related topics

“Main properties for a Bristol Babcock OpenBSI controller” on page 17

Optimizing Bristol Babcock OpenBSI scanning performance

All point parameters that reference Bristol Babcock OpenBSI data should be assigned a scan period that is appropriate to the point scanning strategy. Refer to the topic titled "Scanning and standard points" in the *Server and Client Configuration Guide* for information about scan periods.

Take into account network structure when assigning scan periods, especially for limited-bandwidth IP networks, such as satellite systems.

Periodic scanning and dynamic scanning use the Remote Database (RDB) mechanism for retrieving data.

Scan rates for auxiliary parameters must be no faster than 60 seconds.

The scan packets that have been built can be listed by using the utility **lisscn** (list scan). Listing scan packets helps verify the scanning strategy.

For more information about **lisscn**, see the section titled "Command Reference" in the *Server and Client Configuration Guide*.

Related topics

“Main properties for a Bristol Babcock OpenBSI channel” on page 16

Reporting by exception

If the node supports this feature, it can be used to reduce traffic on the network.

Signals can be configured on RBE to report when the value changes by a specified deadband value. The RBE messages have the same priority as Remote Database (RDB) messages. Therefore, any critical signals should be configured to alarm. Alarms have the highest priority.

Attention

- Avoid configuring status signals to use the alarm system as well as the RBE system. This is unnecessary duplication.
- The most efficient scanning system is to have points mainly on RDB and RBE scanning, with RTU Alarms only being used for critical conditions.

Understanding polling

To receive data from a Bristol Babcock node it must be polled. This makes the frequency of polling very significant in terms of speed of data movement around a Bristol Babcock OpenBSI network.

The server will, by default, poll all controllers every 5 seconds. Each poll can receive a reply from the OpenBSI service intended for any RTU on the network.

This polling is complemented by:

- An additional poll that is issued for each acquisition request. (That is, if a response is expected, a poll is also issued.)
- An immediate poll if NAK is received from the node. (NAK means that the node needs polling.)
- Continuous polling (up to a limit; default 8) while data remains in the node.

Bristol Babcock OpenBSI scan packets

Four types of scan packet are built for an Bristol Babcock interface:

Scan packet type	Description
Hardware Diagnostic	One scan per controller every 60 seconds to verify communications integrity with the controller.
Exception Poll	Controllers are polled at the poll period.
Periodic Data Acquisition Scan	One per scan packet (arrays and RTU signal lists only).
Periodic List Data Acquisition Scan	One per scan packet (signals only).

A Bristol Babcock scan packet may consist of up to:

- 38 signals
- 59 array addresses
- 1 signal list

To be in a scan packet, points must reside:

For	
Arrays	• In the one server controller • Have the same scan period • Data must reside in consecutive elements of the array (in rows)
Signals	• In the one server controller • Have the same scan period
Signal lists	• In the one server controller (one packet per signal list)

The scan packets that have been built can be listed by using the utility **lisscn** (list scan). Listing scan packets helps verify the scanning strategy.

For more information about **lisscn**, see the section titled "Command Reference" in the *Server and Client Configuration Guide*.

Defining Report by Exception Parameters

If Report By Exception is being used for any controller then the initialization data must be specified in Quick Builder. These parameters are always downloaded to the node, and will always override any parameters set when the load was built.

CAUTION
Be careful not to set RBE_PERIOD too low initially as the communications system may be flooded with reports if there are rapidly changing signals on RBE. Leave at the default value until the network is stable.

Bristol Babcock OpenBSI points reference

This section describes how to configure points for a Bristol Babcock OpenBSI controller using Quick Builder.

In addition to the information contained in this reference, and for help to build points, see the section titled "Building and configuring points" in the *Quick Builder User's Guide*.

Related topics

"Defining a Bristol Babcock OpenBSI address for a point parameter" on page 24

Defining a Bristol Babcock OpenBSI address for a point parameter

Related topics

"Address syntax for Bristol Babcock OpenBSI controllers" on page 24
"Point addresses supported by Bristol Babcock OpenBSI" on page 25
"Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers" on page 5
"Testing Bristol Babcock OpenBSI communications with the server" on page 45

Address syntax for Bristol Babcock OpenBSI controllers

For source and destination addresses the format for a Bristol Babcock OpenBSI controller address is:

ControllerName Address

Part	Description
ControllerName	The name of the Bristol Babcock OpenBSI controller.
Address	The address in the controller where the value is recorded. See the section below titled "Address syntax" for more information.

If you would like help with the address, you can use the Address Builder. To display the Address Builder, click ... next to **Address**.

Address syntax

The format for the address is:

Address

Part	Description
Address	The address of the point parameter. See the sections below titled "Signal address" and "Array address."

Signal address

For points referencing signals, definitions are currently done within Quick Builder. The address syntax for a point addressing a signal is:

SignalName [USER_LIST=*UserList*]

Part	Description
SignalName	The signal name, which must be 1 to 20 alphanumeric characters. ControlWave signals may need to be pre-pended with *@GV.* to be read properly from the controller.
UserList	The number of the BB RTU Signal List which contains this *SignalName*. A scan packet will be created to read the entire Signal List. *UserList* is a number between 1 and 255. Every point referencing the *SignalName* should have the *UserList* definition included.

Configuring alarms for signal data

Alarms are used to update point parameters and are the quickest way of transmitting data over the network. Alarms can only be configured for points built on signal data.

Use alarms to:

- Update point parameters.
- Quickly transmit data over the network.
- Reduce periodic scanning.
- Store as events in order to retain Bristol Babcock alarm timestamp information. (The Sequence of Events (SOE) file is used for this purpose.)

Attention

Frequent alarms do impact the periodic scanning of the system, and can cause increased network traffic. Alarms should not be used for signals that are changing frequently.

Reducing periodic scanning with alarms

Place a status point parameter on a slow scan period (300 seconds) and configure the signal in the node to alarm on a change of state.

Retaining timestamp information

An event report entry includes the timestamp received in the alarm report to a resolution of one millisecond. If an alarm value is used to update a process variable, and this causes an alarm to occur, the alarm will be time stamped with the system time not the time in the alarm report. The difference between these timestamps could be significant depending on the network architecture, hence the need to retain the "real" alarm time in the SOE file.

See the *Server and Client Configuration Guide* for more information on SOE configuration and SOE reports.

Attention

- A point reference for the alarm must exist if an SOE entry is to be produced.
- Alarms must be enabled for the point if an SOE entry is to be produced.
- The "extended alarm format" (that has signal inhibit and alarm inhibit information) is not supported.

Array address

For points referencing arrays, definitions are done completely within Quick Builder. The address syntax for a point addressing an array is:

ArrayNumber Row Column

where:

Part	Description
ArrayNumber	The array number—must be in the range 1 to 255.
Row	The row number—must be in the range 1 to 255.
Column	The column number—must be in the range 1 to 255.

Related topics

"Configuration Log properties for a Bristol Babcock OpenBSI meter template" on page 32

Point addresses supported by Bristol Babcock OpenBSI

Valid point addresses specific to the Bristol Babcock OpenBSI interface that can be defined for the server point types are shown. Note the difference between signal and array addressing.

Address	Controller Type Signals	Controller Type Arrays
Accumulator Point		
PVSOURCE	Y	Y
Analog Point		
AxDESTIN	Y	Y
AxSOURCE	Y	Y
MDDESTIN	Y	
MDSOURCE		
OPDESTIN	Y	Y
OPSOURCE	Y	Y
PVSOURCE	Y	
SPDESTIN	Y	
SPSOURCE	Y	
Status Point		
OPDESTIN	Y	Y
OPSOURCE	Y	Y
MDDESTIN	Y	
MDSOURCE		
PVSOURCE	Y	Y

Data format

Bristol Babcock OpenBSI analog values are all stored as single precision (4 byte) IEEE floating point values.

Bristol Babcock OpenBSI logical values are all one-bit.

String signals are not supported by the Bristol Babcock OpenBSI interface as the server does not support the string data type.

The correlation between Bristol Babcock OpenBSI signal types and server point types is given in the following table.

Signal Type	Server Point Type	Format	Range
Analog	Analog	IEEEFP (4 byte)	single precision floating point
Analog	Accumulator	IEEEFP (4 byte)	
Logical	Status (one bit)	Unpacked (1 byte = 1 val)	off = 0x00, on = 0x01
Packed Logical	Not supported	Packed (1 byte = 8 val)	off = 0, on = 1
String	Not supported	NULL terminated (65 char)	

MODE support

Attention
Do not enable MD reverse on points connected to Bristol Babcock controllers and Bristol Babcock OpenBSI controllers.

The modes MAN-LSP, MAN-RSP, AUTO-LSP, and AUTO-RSP are supported.

The MAN mode will correspond to "Manual Enable" and the AUTO mode to "Manual Inhibit" in ACCOL terminology. Similarly, LSP and RSP will correspond to Control Enable and Control Inhibit.

The mode will be obtained from OPSOURCE address.

The MDDESTIN entry must have the same address as OPSOURCE and OPDESTIN. MDPERIOD is not required.

The Manual Inhibit/Enable and Control Inhibit/Enable will be tracked via the mode parameter. The following table gives the correlation between the controller and server modes.

Controller Mode	Server Mode	Default Server Mode Acronym
Manual inhibit/control inhibit	AUTO-RSP	CASC
Manual inhibit/control enable	AUTO-LSP	AUTO
Manual enable/control inhibit	MAN-RSP	MAN-RSP
Manual enable/control enable	MAN-LSP	MAN

Bristol Babcock OpenBSI EFM meter templates reference

This section describes the configuration information specific to meter templates on Bristol Babcock OpenBSI controllers. See the "Building Electronic Flow Measurement (EFM)" section of the Quick Builder User's Guide for information about how to build EFM meter templates and equipment.

Related topics

"Main properties for a Bristol Babcock OpenBSI meter template" on page 30
"Configuration Log properties for a Bristol Babcock OpenBSI meter template" on page 32
"Interval Log properties for a Bristol Babcock OpenBSI meter template" on page 33
"Daily Log properties for a Bristol Babcock OpenBSI meter template" on page 34
"Alarm and Event properties for a Bristol Babcock OpenBSI meter template" on page 35
"Ultrasonic Log properties for a Bristol Babcock OpenBSI meter template" on page 36
"Composition Log properties for a Bristol Babcock OpenBSI meter template" on page 37
"Gas Quality Log properties for a Bristol Babcock OpenBSI meter template" on page 38
"Data Export properties for a Bristol Babcock OpenBSI meter template" on page 39

Main properties for a Bristol Babcock OpenBSI meter template

This topic lists the Main tab settings for a meter template on a Bristol Babcock OpenBSI controller. Different flow computers require different settings. The **Interval Log**, **Daily Log**, **Alarm and Event Log**, **Configuration Record Log**, **Ultrasonic Log**, **Composition Log**, and **Gas Quality Log** logs are all optional. However, you would configure at least one, if not all, of them. The flow computer vendor documentation should provide addressing information for addressing the logs, with following information provided for immediate reference for Bristol Babcock OpenBSI devices.

See the topic titled "Main properties for an EFM meter template" in the *Quick Builder User's Guide* for information about each field on this tab.

To make configuring meter templates an easier task, samples of Bristol Babcock OpenBSI meter templates are included in Experion. See the topic titled "Managing EFM meter templates" in the *Quick Builder User's Guide* for information about how to import sample Bristol Babcock OpenBSI meter templates.

Data structures

Table 1: Data structure types

Data structure	Description
Archive	References the archive file in the controller. Specified by archive number only. It does not require a pointer. The EFM log fields are always returned in the order: DateAndTime, Archive sequence number, Global sequence number, Data values. The rollover is 65,535, as records are fetched by sequence number, NOT by the size of the archive.
Wrap-Around Array	Requires a signal whose value is the last written row number, 1 as first row. You must specify the number of rows is required.
Push-Down Array	No signal pointer is required. Rows are always read downward from *Row 1* until the first field in the array matches the last stored value.

Alarm and Event Log

The Alarm and Event log is always retrieved from the audit file. Events in the audit file in the controller are deleted after collection.

While many meters can be configured on a controller, only one meter should be configured to collect the Alarm and Event log.

Bristol Babcock OpenBSI flow computers

The following table shows sample settings for a Bristol Babcock OpenBSI meter template. The sample is for an Ultrasonic Meter template, so it does not collect/export Orifice plate configuration signals for the Configuration log.

Table 2: Sample Ultrasonic Meter template settings for a Bristol Babcock OpenBSI meter template

Property	Sample setting
Interval Log	
Type	Archive File
Number	52

Property	Sample setting
Pointer	
Rollover	65535
Daily Log	Disabled
Alarm and Event Log	Enabled
Ultrasonic Log	
Type	Archive File
Number	231
Pointer	
Rollover	65535
Composition Log	Disabled
Gas Quality Log	Disabled

Configuration Log properties for a Bristol Babcock OpenBSI meter template

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, configuration log properties are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

If the configuration requires compliance with API21.1 then the configuration properties configured here need to include at least those listed in the API21.1 standard.

'Controller' source properties should use the address syntax as defined in the topic titled "Address syntax for Bristol Babcock OpenBSI controllers" and may include *USER_LIST* reference, which will cause the *USER_LIST* to be requested at time of the configuration log request.

See the topic titled "Configuration Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add configuration properties to this tab.

Related topics

“Address syntax for Bristol Babcock OpenBSI controllers” on page 24

Interval Log properties for a Bristol Babcock OpenBSI meter template

The **Interval Log** tab appears only when the interval log is enabled on the **Main** tab.

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the properties you collect for the Interval Log are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

Data fields in arrays should always be specified as REAL data type.

When configuring log collection from archives, the following fields are returned by the controller before the archive record data. These fields need to be included in the log configuration:

- *DateAndTime*: BYTE[4]
- *Archive Sequence Number*: INT2
- *Global Sequence Number*: INT2

See the topic titled "Interval Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add interval properties to this tab.

Daily Log properties for a Bristol Babcock OpenBSI meter template

The **Daily Log** tab appears only when the daily log is enabled on the **Main** tab.

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the properties you collect for the Daily Log are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

When configuring log collection from archives, the following fields are returned by the controller before the archive record data. These fields need to be included in the log configuration:

- *DateAndTime*: BYTE[4]
- *Archive Sequence Number*: INT2
- *Global Sequence Number*: INT2

See the topic titled "Daily Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add properties to this tab.

Alarm and Event properties for a Bristol Babcock OpenBSI meter template

The **Alarm and Event** tab appears only when the alarm and event log is enabled on the **Main** tab.

The audit file collection is a fixed format for all Bristol Babcock OpenBSI controllers. Use the following table to configure the Alarm and Event log.

Field Name	Data Type	CFX 5 and CFX 7 Mapping
Type	BYTE[1]	Type
Signal	VARTEXT	Name Text
DateAndTime	BYTE[5]	Time Stamp
New Value	REAL	New Value
Old Value	REAL	Old Value
State0 Text	TEXT[6]	State0 Text
State1 Text	TEXT[6]	State1 Text

See the topic titled "Alarm and Event properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add properties to this tab.

Ultrasonic Log properties for a Bristol Babcock OpenBSI meter template

The **Ultrasonic Log** tab appears only when the ultrasonic log is enabled on the **Main** tab.

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the properties you collect for the Ultrasonic Log are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

When configuring log collection from archives, the following fields are returned by the controller before the archive record data. These fields need to be included in the log configuration:

- *DateAndTime*: BYTE[4]
- *Archive Sequence Number*: INT2
- *Global Sequence Number*: INT2

See the topic titled "Ultrasonic Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add properties to this tab.

Composition Log properties for a Bristol Babcock OpenBSI meter template

The **Composition Log** tab appears only when the composition log is enabled on the **Main** tab.

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the properties you collect for the Composition Log are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

When configuring log collection from archives, the following fields are returned by the controller before the archive record data. These fields need to be included in the log configuration:

- *DateAndTime*: BYTE[4]
- *Archive Sequence Number*: INT2
- *Global Sequence Number*: INT2

See the topic titled "Composition Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add properties to this tab.

Gas Quality Log properties for a Bristol Babcock OpenBSI meter template

The **Gas Quality Log** tab appears only when the gas quality log is enabled on the **Main** tab.

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the properties you collect for the Gas Quality Log are specific to the flow computer. See the vendor documentation for the properties supported by the flow computer.

When configuring log collection from archives, the following fields are returned by the controller before the archive record data. These fields need to be included in the log configuration:

- *DateAndTime*: BYTE[4]
- *Archive Sequence Number*: INT2
- *Global Sequence Number*: INT2

See the topic titled "Gas Quality Log properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to add properties to this tab.

Data Export properties for a Bristol Babcock OpenBSI meter template

When creating meter templates for flow computers on Bristol Babcock OpenBSI controllers, the data export formats you specify depend on the gas measurement system receiving the exported EFM data.

See the topic titled "Data Export properties for an EFM meter template" in the *Quick Builder User's Guide* for information on how to configure data export formats on this tab.

Bristol Babcock OpenBSI Application Programming Interface

The Bristol Babcock OpenBSI interface provides its own Application Programming Interface (API) for use by third-party applications. This allows the user to write an application to request and receive data from a Bristol Babcock OpenBSI controller, without having to implement the communication overhead.

C Synopsis

```
#include "userbsi.h"
int c_userbsi(iTimeout, iMailSlot, iDestChn, szDestRtu,
    ucDestMex, iReqSize, bufReq, piRespSize, bufResp)
int iTimeout
int iMailSlot
int iDestChn
char *szDestRtu
unsigned char ucDestMex
int iReqSize
void *bufReq
int *piRespSize
void *bufResp
```

Argument	Description
iTimeout	(in) the amount of time to wait for a reply before returning an error (in seconds).
iMailSlot	(in) the mail slot number to use to make the request.
iDestChn	(in) the number of the channel to use to send the message to the controller.
szDestRtu	(in) the name of the RTU to which the message will be sent. This name does not need to correspond to a controller.
ucDestMex	(in) the message exchange to which the message will be sent.
iReqSize	(in) the size of the request buffer, in bytes.
bufReq	(in) the request buffer.
piRespSize	(in/out) the size of the response buffer, in bytes.
bufResp	(in/out) the response buffer.

Description

USERBSI sends a user-created buffer to an RTU, and returns the RTU response buffer. It will return when a response is received from the RTU, or when the timeout period has expired. It is recommended that only five API calls are made at any one time, to avoid impacting normal server operation.

Diagnostics

Upon successful completion, 0 is returned.

Otherwise, -1 is returned.

Internal errors and diagnostics when processing the user message will be written to the server log.

See *bsi_apitst.c* for examples of creating BSAP requests and decoding BSAP responses.

Troubleshooting Bristol Babcock OpenBSI issues

This section describes troubleshooting tasks for Bristol Babcock OpenBSI that you can perform either on the server or from any Station.

Related topics

“Troubleshooting Bristol Babcock OpenBSI point configuration errors” on page 44
“Testing Bristol Babcock OpenBSI communications with the server” on page 45
“Troubleshooting Bristol Babcock OpenBSI scanning errors” on page 46
“Confirming resolution of Bristol Babcock OpenBSI point addresses” on page 47

Troubleshooting Bristol Babcock OpenBSI point configuration errors

When downloading to the server the errors specific to the Bristol Babcock OpenBSI are:

Error	Reason
Too many fields	Check file for unexpected field.
Controller type not supported	Check that hardware was built correctly.
Signal name is too long	Check the full signal name is less than or equal to 20 characters.
Invalid list number	Check the list number is in the range 1 to 255.
Invalid Bristol Babcock array number	Check the array number is in the range 1 to 255.
Invalid Bristol Babcock array row number	Check the row number is in the range 1 to 255.
Invalid Bristol Babcock array column number	Check the column number is in the range 1 to 255.

Testing Bristol Babcock OpenBSI communications with the server

You use the Bristol Babcock OpenBSI test utility, **bsitst**, to test communications between the server and the Bristol Babcock OpenBSI controller after you have downloaded channel and controller definitions to the server database.

Prerequisites

- Set up the controller.
- Connect all cables.
- Define the controller and channel in Quick Builder.
- Download the Quick Builder definitions to the server, without errors.
- Ensure the channel is out of service.

Attention
If you want the test utility to receive RBE and Alarm messages, shut down the server.

To run the bsitst utility

1. Open a Command Prompt window.
2. Type **bsitst** and then press Enter.
3. Follow the directions as prompted.
   The device Ethernet or serial interface transmit and receive LEDs will flash during transactions.

   If any errors are encountered, review the channel and controller definitions in Quick Builder.

Results

After you verify that the server is communicating with the Bristol Babcock controllers you can configure points. See the topic titled "Defining a Bristol Babcock OpenBSI address for a point parameter."

Related topics

“Troubleshooting Bristol Babcock OpenBSI scanning errors” on page 46
“Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers” on page 5
“Defining a Bristol Babcock OpenBSI address for a point parameter” on page 24

Troubleshooting Bristol Babcock OpenBSI scanning errors

If the Point Detail display for a point shows a bad value (indicated by inverse video), the point might built with an address that is syntactically valid but not configured in the controller.

The controller responds to a request to read this address with an A-NAK message. Note that NAK messages received in response to a read request by the server do not contribute to the error count of the controller or channel.

NAK messages received in response to a write request by the server result in an operator message being sent to the Station as well as contributing to the controller's error count.

Any NAK messages received are printed in the server error log, which can be viewed via the Server Diagnostics Program Group.

If NAK messages occur, check the relevant point address to ensure there are no references to addresses in the controller that are not actually configured within the controller.

Related topics

“Testing Bristol Babcock OpenBSI communications with the server” on page 45
“Planning considerations for installing and configuring Bristol Babcock OpenBSI controllers” on page 5

Confirming resolution of Bristol Babcock OpenBSI point addresses

The server maintains a list of internal addresses that correspond to signal names. When periodic scanning or Reporting by Exception is initialized, signal names are used until the addresses have been determined. From then on, all communication messages will use internal addresses to reduce the message size.

You can view the list of internal addresses using the **padinfo** utility. For each signal name on the specified controller, there are four columns of four digits following the name, for example, *0000 0000 0000 0000*. Once the signal name has been resolved from the controller, the first column will display the internal address, for example, *02de 0000 0000 0000*.

If messages are being logged on the server about unknown signal names, use the **padinfo** utility to determine which signal names to check.

Attention

If a signal name is configured as an *OPDESTIN* or *SPDESTIN* only (that is, not a *SOURCE* address), then its internal address will not be resolved, and the column will remain as *0000*.

If the load changes, it is necessary to re-initialize the signal addresses. This is done automatically, but can be forced using the **padrst** utility that will prompt you for the controller you wish to re-initialize.

For information about using the **padinfo** and **padrst** utilities, see the *Configuration Guide* for your server.

Notices

Trademarks

Experion®, PlantScape®, SafeBrowse®, TotalPlant®, and TDC 3000® are registered trademarks of Honeywell International, Inc.

OneWireless™ is a trademark of Honeywell International, Inc.

Other trademarks

Microsoft and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Trademarks that appear in this document are used only to the benefit of the trademark owner, with no intention of trademark infringement.

Third-party licenses

This product may contain or be derived from materials, including software, of third parties. The third party materials may be subject to licenses, notices, restrictions and obligations imposed by the licensor. The licenses, notices, restrictions and obligations, if any, may be found in the materials accompanying the product, in the documents or files accompanying such third party materials, in a file named third_party_licenses on the media containing the product, or at http://www.honeywell.com/ps/thirdpartylicenses.

Documentation feedback

You can find the most up-to-date documents on the Honeywell Process Solutions support website at:

http://www.honeywellprocess.com/support

If you have comments about Honeywell Process Solutions documentation, send your feedback to:

hpsdocs@honeywell.com

Use this email address to provide feedback, or to report errors and omissions in the documentation. For immediate help with a technical problem, contact your local Honeywell Process Solutions Customer Contact Center (CCC) or Honeywell Technical Assistance Center (TAC) listed in the “Support and other contacts” section of this document.

How to report a security vulnerability

For the purpose of submission, a security vulnerability is defined as a software defect or weakness that can be exploited to reduce the operational or security capabilities of the software.

Honeywell investigates all reports of security vulnerabilities affecting Honeywell products and services.

To report a potential security vulnerability against any Honeywell product, please follow the instructions at:

https://honeywell.com/pages/vulnerabilityreporting.aspx

Submit the requested information to Honeywell using one of the following methods:

- Send an email to security@honeywell.com.

  or

- Contact your local Honeywell Process Solutions Customer Contact Center (CCC) or Honeywell Technical Assistance Center (TAC) listed in the “Support and other contacts” section of this document.

Support

For support, contact your local Honeywell Process Solutions Customer Contact Center (CCC). To find your local CCC visit the website, https://www.honeywellprocess.com/en-US/contact-us/customer-support-contacts/Pages/default.aspx.

Training classes

Honeywell holds technical training classes on Experion PKS. These classes are taught by experts in the field of process control systems. For more information about these classes, contact your Honeywell representative, or see http://www.automationcollege.com.

Index

A

ACCOL tools 10
additional definitions for signal controller type 20
address syntax 24, 25
addressing
  Bristol Babcock OpenBSI controllers 24
Alarm and Event tab
  meter templates 35
alarm inhibit 17
application programming interface (API) 41
architectures
  Bristol Babcock OpenBSI 10

B

Bristol Babcock OpenBSI controllers
  ACCOL tools 10
  address syntax 24, 25
  addressing 24
  alarm inhibit 17
  application programming interface (API) 41
  architectures 10
  bsitst utility 45
  channels 15, 16
  communication settings 11
  configuring 5, 14
  controller type 20
  controllers 15
  data format 25
  devices supported 7
  documentation 8
  Electronic Flow Measurement (EFM) 29
  exception poll period 16
  fail alarm limit 16, 17
  getting started 5
  marginal alarm limit 16, 17
  meter templates 29, 30, 32–39
  modes 25
  NAK messages 46
  NetView 14, 20
  network address 12
  no RBE init reports 17
  node name 17
  OpenBSI configuration 14
  OpenBSI tools 10
  optimizing scanning performance 21
  padinfo 47
  padrst 47
  points 23–25
  polling 21
  RBE inhibit 17
  RBE scan rate 17
  RBE scan slice 17
  RBE stop xmit 17
  RBE timeout 17
  RDB security level 17
  reporting by exception 21
  scanning performance 21
  SOE inhibit 17
  supported architectures 10
  supported devices 7
  sync enable 17
  sync time 17
  terms 9
  testing communications 45
  transaction timeout 17
  troubleshooting 43–47
  writing applications 41
BSAP protocol 7, 10
bsitst utility 45

C

channels
  Bristol Babcock OpenBSI channel 16
  Bristol Babcock OpenBSI controllers 15
  reference 15
commands and utilities
  bsitst 45
  lisscn 21
  padinfo 47
  padrst 47
communications
  Bristol Babcock OpenBSI controllers 45
    testing
      Bristol Babcock OpenBSI controllers 45
Composition Log tab
  meter templates 37
Configuration Log tab
  meter templates 32
configuring
  Bristol Babcock OpenBSI controllers 5
  meter templates 32
  OpenBSI 14
controllers
  application programming interface (API) 41
  Bristol Babcock OpenBSI controllers 5, 7–12, 14, 15, 20, 21, 24, 41
    configuring
      Bristol Babcock OpenBSI controllers 5
  controller type 20
  reference 15

type 20

D

Daily Log tab
meter templates 34
Data Export tab
meter templates 39
data formats
Bristol Babcock OpenBSI controllers 25
devices supported 7
diagnostics
Bristol Babcock OpenBSI controllers 45
documentation
Bristol Babcock OpenBSI 8

E

Electronic Flow Measurement (EFM)
Bristol Babcock OpenBSI 29
meter templates 29, 30, 32–39
exception poll period 16

F

fail alarm limit 16, 17

G

Gas Quality Log tab
meter templates 38

I

Interval Log tab
meter templates 33

L

lisscn utility 21

M

Main tab
Bristol Babcock OpenBSI channel 16
Bristol Babcock OpenBSI controller 17
Bristol Babcock OpenBSI meter template 30
marginal alarm limit 16, 17
meters
meter templates 30, 32–39

N

NetView 14, 20
no RBE init reports 17
node name 17

P

padinfo utility 47
padrst utility 47
points
address syntax 24, 25
Bristol Babcock OpenBSI controllers 23–25, 44
configuration errors 44
reference 23
troubleshooting 44

R

RBE inhibit 17
RBE scan rate 17
RBE scan slice 17
RBE stop xmit 17
RBE timeout 17
RDB security level 17
reporting by exception 21
RS-562 10

S

scanning
errors 46, 47
optimizing performance 21
troubleshooting 46, 47
SOE inhibit 17
supported devices 7
sync enable 17
sync time 17

T

testing communications
Bristol Babcock OpenBSI controllers 45
transaction timeout 17
troubleshooting
Bristol Babcock OpenBSI controllers 43, 44, 46, 47
NAK messages 46
scanning errors 46, 47

U

Ultrasonic Log tab
meter templates 36
utilities
bsitst 45
lisscn 21
padinfo 47
padrst 47