

Experion PKS  
**METTLER TOLEDO Integration Reference**

EPDOCXX69-en-431A  
February 2015

**Release 431**

Document	Release	Issue	Date
EPDOCXX69-en-431A	431	0	February 2015

## Disclaimer

This document contains Honeywell proprietary information. Information contained herein is to be used solely for the purpose submitted, and no part of this document or its contents shall be reproduced, published, or disclosed to a third party without the express permission of Honeywell International Sàrl.

While this information is presented in good faith and believed to be accurate, Honeywell disclaims the implied warranties of merchantability and fitness for a purpose and makes no express warranties except as may be stated in its written agreement with and for its customer.

In no event is Honeywell liable to anyone for any direct, special, or consequential damages. The information and specifications in this document are subject to change without notice.

Copyright 2015 - Honeywell International Sàrl

# Contents

<b>1 About This Document .....</b>	<b>5</b>
<b>2 METTLER TOLEDO integration overview .....</b>	<b>7</b>
2.1 Introduction .....	8
2.1.1 Q.iMPACT integration products .....	8
2.1.2 JAGXTREME integration blocks .....	8
<b>3 METTLER TOLEDO Q.iMPACT integration .....</b>	<b>9</b>
3.1 Overview of Q.iMPACT in Experion .....	10
3.1.1 Q.iMPACT terminal support .....	10
3.2 Architecture with QIPACTERM and QIPACCHAN blocks .....	11
3.2.1 Experion function block architecture .....	11
3.2.2 Data flow model .....	11
3.2.3 QIPACTERM block .....	12
3.2.4 QIPACCHAN block .....	12
3.3 Q.iMPACT integration setup .....	14
3.3.1 Software environment requirements .....	14
3.3.2 System Performance, Capacity, and Topology .....	14
3.3.3 Configuration tools .....	14
3.3.4 Update ControlNet settings .....	14
3.3.5 Hardware installation .....	15
3.3.6 Q.iMPACT terminal configuration .....	15
3.4 Configure a QIPACTERM block .....	16
3.4.1 Physical location .....	16
3.4.2 Q.iMPACT terminal configuration .....	16
3.4.3 QIPACTERM configuration loading .....	17
3.4.4 QIPACTERM block parameters .....	17
3.5 Configure a QIPACCHAN block .....	18
3.5.1 Assign the QIPACCHAN block .....	18
3.5.2 QIPACCHAN block parameters .....	18
3.6 Communication and fault behaviors .....	19
3.7 Q.iMPACT operation and notifications .....	20
3.7.1 Input processing .....	20
3.7.2 Status processing .....	20
3.7.3 Output processing .....	20
3.7.4 Notifications .....	20
3.8 QIPACTERM block parameters .....	22
3.8.1 QIPACTERM parameters .....	22
3.9 QIPACCHAN block parameters .....	24
3.9.1 QIPACCHAN parameters .....	24
3.9.2 QIPACCHAN input parameters .....	24
3.9.3 QIPACCHAN material transfer status parameters .....	26
3.9.4 QIPACCHAN material transfer interface parameters .....	26
3.9.5 QIPACTERM communications parameters .....	27
<b>4 METTLER TOLEDO JAGXTREME integration .....</b>	<b>29</b>
4.1 Overview of JAGXTREME in Experion .....	30
4.1.1 JAGXTREME terminal support .....	30

4.2 Architecture with JAGXTERM and JAGXCCHAN blocks .....	31
4.2.1 Experion function block architecture .....	31
4.2.2 Data flow model .....	31
4.2.3 JAGXTERM block .....	32
4.2.4 JAGXCHAN block .....	32
4.3 JAGXTREME integration setup .....	34
4.3.1 Software environment requirements .....	34
4.3.2 System performance, capacity, and topology specifications .....	34
4.3.3 Configuration tools .....	34
4.3.4 Update ControlNet settings .....	34
4.4 Configure a JAGXTERM block .....	35
4.4.1 Physical location .....	35
4.4.2 JAGXTREME terminal configuration .....	36
4.4.3 JAGXTERM configuration loading .....	36
4.4.4 JAGXTERM block parameters .....	36
4.5 Configure a JAGXCHAN block .....	37
4.5.1 Assign the JAGXCHAN block .....	37
4.5.2 JAGXCHAN block parameters .....	37
4.6 Communication and fault behaviors .....	38
4.7 JAGXTERM operation and notifications .....	39
4.7.1 Input processing .....	39
4.7.2 Status processing .....	39
4.7.3 Notifications .....	39
4.8 JAGXTERM parameters .....	40
4.9 JAGXCHAN parameters .....	41
4.9.1 JAGXCHAN input parameters .....	41
4.9.2 JAGXCHAN command parameters .....	42

# 1 About This Document

This document is intended to provide the information that is necessary for users of the METTLER TOLEDO Q.iMPACT and JAGXTREME integration tools available from Honeywell for the integration of these products with Experion Process Knowledge System (PKS).

## Revision history

Revision	Date	Description
A	February 2015	Initial release of document.

## Who should use this manual

This manual is intended for developers and implementers who are creating and using process control solutions that involve the METTLER TOLEDO Q.iMPACT and/or JAGXTREME (JX) terminals and related scale and flow meter products. Users should be familiar with the appropriate METTLER TOLEDO products. They should also be familiar with the Experion system in general and with the Control Builder application in particular.

## METTLER TOLEDO integration prerequisites

Intimate knowledge of METTLER TOLEDO devices is required to properly configure them. Detailed information can be found at the supplier's website at .

The configuration of the METTLER TOLEDO subsystem must be complete, functional, and documented before the integration to Experion can be completed. In addition, the METTLER TOLEDO system must have a ControlNet interface card.

## Organization of this document

This document is divided into two separate sections covering the following integration products:

- “METTLER TOLEDO Q.iMPACT integration” on page 9
- “METTLER TOLEDO JAGXTREME integration” on page 29

These products are provided by Honeywell to support the integration of the METTLER TOLEDO Q.iMPACT and/or JAGXTREME terminals with the Experion system.

The Component Control Libraries (CCLs) for these products are included with the Experion Control Builder application beginning with release 210. You must purchase a license in order to use the METTLER TOLEDO integration components. Q.iMPACT and JAGXTREME integration products are licensed separately.

## Terms and acronyms

Term/Acronym	Definition
Assembly	A device-defined data structure used to communicate between an Experion system and a METTLER TOLEDO device.
Association	Refers to the mapping of I/O channel function blocks to I/O module function blocks, which interface the physical I/O device's data.

Term/Acronym	Definition
CB	Control Builder
CCL	Control Component Library
CEE	Control Execution Environment
CM	Control Module
FB	Function Block. A control object in the Experion system that performs a specific control action.
IOC	I/O Channel FB. The function block representing a single channel of an I/O device. IOC function blocks are used in control building to allow development of control strategies that are independent of underlying I/O devices. Prior to load, each IOC function block is associated with an I/O device.
I/O Manager	Name for subsystem that runs in the C200 controller that is responsible for establishing, monitoring, closing, and re-establishing communications connections to I/O devices.
IOM	I/O Module FB. The C200 Controller resident function block that works with IOC function blocks to interface I/O device data to the Control Execution Environment (CEE).
IO Module Device	Refers to the actual hardware card or equipment.
IOC	I/O Channel FB
IOFB	IO Function Block (IOC and IOM)
IOM	IO Module FB
JX	JAGXTREME
Load	Experion activity that results in the function block being loaded from the System Repository to the controller. This operation initiates connection to the IO module device.
MT	METTLER TOLEDO
NaN	Not a Number (IEEE Floating Point representation)
PAC	Predictive Adaptive Control
PCM	Process Control Module. The Honeywell module that combined with an ICP back plane, one or more ControlNet Interface modules (and optional I/O devices) constitute a controller.
PV	Process Value
SCM	Sequential Control Module. The SCM 'sequences' the equipment module through a series of distinct states to accomplish a process task (such as heat, cool, agitate, charge, discharge, etc.). The SCM may be used to perform a stand-alone sequencing operation or for batch control in presence of the Level-3 and Level-4 above.
Tagged block	Function block, which has a system global name, such as IOM, CM and SCO.

## 2 METTLER TOLEDO integration overview

### **Related topics**

“Introduction” on page 8

---

## 2.1 Introduction

The Q.iMPACT matroller (material transfer controller) and the JAGXTREME scale terminals are integral parts of a larger batch control system. They operate within a tightly integrated subsystem with 'open' connectivity.

METTLER TOLEDO and Honeywell have collaborated to deliver load-and-go integration that operates on standard communication protocols and system I/O to deliver robust and cost effective solutions. Process data, information, status, and diagnostics contained in the Q.iMPACT matroller and JAGXTREME terminals are available to the host system with the exception of configuration, which is done using the terminal front panels, and a standalone PC/Workstation (QiMPACT only). Experion function blocks provide all the communication, diagnostics and display capabilities required to deliver seamless plug-and-play integration. The standard Experion CNI card provides connectivity via standard ControlNet protocol. This integrated solution provides customers with tighter material transfer control, decreased batch cycle time, and improved production quality.

### 2.1.1 Q.iMPACT integration products

The Q.iMPACT integration products available from Honeywell consist of two function blocks:

- “QIPACTERM block” on page 12: An I/O module function block that assembles up to 24 channels of input data from a Q.iMPACT terminal, and forwards commands from the system to the terminal.
- “QIPACCHAN block” on page 12: A channel function block that works with the QIPACTERM block to provide data from the terminal devices to a control strategy, and accepts commands from a control strategy and passes them to the terminal devices.

### 2.1.2 JAGXTREME integration blocks

The JAGXTREME integration products available from Honeywell consist of two function blocks:

- “JAGXTERM block” on page 32: An I/O module function block that assembles up to four channels of input data from a JAGXTREME (JX) terminal, and forwards commands from the system to the terminal.
- “JAGXCHAN block” on page 32: A channel function block that works with the JAGXTERM block to provide data from the terminal devices to a control strategy, and accepts commands from a control strategy and passes them to the terminal devices.



## 3 METTLER TOLEDO Q.iMPACT integration

### Related topics

- “Overview of Q.iMPACT in Experion” on page 10
- “Architecture with QIPACTERM and QIPACCHAN blocks” on page 11
- “Q.iMPACT integration setup” on page 14
- “Configure a QIPACTERM block” on page 16
- “Configure a QIPACCHAN block” on page 18
- “Communication and fault behaviors” on page 19
- “Q.iMPACT operation and notifications” on page 20
- “QIPACTERM block parameters” on page 22
- “QIPACCHAN block parameters” on page 18

---

## 3.1 Overview of Q.iMPACT in Experion

The information contained in this section is intended to provide a brief overview of the technology involved with integrating ControlNet-connected METTLER TOLEDO Q.iMPACT terminals into the Experion system.

### 3.1.1 Q.iMPACT terminal support

Q.iMPACT terminals communicate over ControlNet using Predictive Adaptive Control (PAC) messaging. This messaging protocol was developed by METTLER TOLEDO to support the PAC technology implemented within the Q.iMPACT.

As part of this PAC functionality, the concept of clustering Q.iMPACT terminals is also implemented. Clustering allows for any ControlNet connected Q.iMPACT terminal to provide data to a host system for any instrument connected to any Q.iMPACT terminal in the cluster. Clustered Q.iMPACT terminals communicate with one another via a separate Ethernet network. The Q.iMPACT cluster can have up to 20 Q.iMPACT terminals and supports up to a maximum of 200 channels. Each channel of a Q.iMPACT cluster represents one connected instrument. The instruments are either load cells (scales) or flow meters. Any ControlNet-connected Q.iMPACT terminal can provide up to 24 channels of data to the C200 controller. Any ControlNet-connected terminal can provide channel data from any channel in the cluster regardless of which terminal the instrument is physically connected.

Each Q.iMPACT terminal has three option slots, each of which can accept a ControlNet interface card, a load cell interface card, or a flow meter interface card. Each load cell interface card supports one or two load cells. Each flow meter interface card supports up to four flow meters. Therefore, a ControlNet connected Q.iMPACT terminal can support up to four load cells, or eight flow meters, or two load cells and four flow meters.

Q.iMPACT terminals that are not directly connected to ControlNet can still be part of the cluster. Their channel data is communicated to the ControlNet connected terminals via peer-to-peer communications over the Ethernet link. These Q.iMPACT terminals have all three-option slots available for instrument option cards. However, any Q.iMPACT terminal can still have a maximum of four connected load cells. Therefore, these terminals can support up to 4 load cells and 4 flow meters, or 12 flow meters.

In order to provide an easy-to-use integration solution, two Experion function blocks are provided. The set consists of an IOM (I/O Module) function block and a Channel function block. The functionality of these function blocks and the parameter set supported by each are described in subsequent sections.

Each Experion C200 CNI card can support four Q.iMPACT terminals. Each C200 controller can in turn support four CNI cards. Therefore, one C200 controller can support a total of 16 Q.iMPACT terminals. Since each ControlNet-connected Q.iMPACT can communicate up to 24 channels of data to the C200, 16 Q.iMPACTs is more than enough to relay all the data for the maximum number of channels (200) in a Q.iMPACT cluster (16 Q.iMPACTs x 24 Channels = 384 Channels).

## 3.2 Architecture with QIPACTERM and QIPACCHAN blocks

The following diagram illustrates the topology of the integration solution.

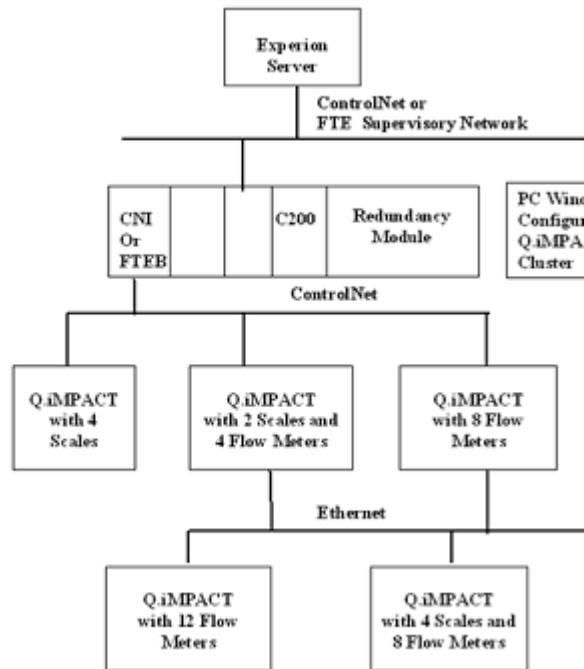


Figure 1: Q.iMPACT Integration Solution Topology

### Topology diagram notes

- Only one C200 controller can be connected to the Q.iMPACT cluster ControlNet. In addition, the Q.iMPACT cluster must reside on an I/O ControlNet segment that is physically isolated from unscheduled I/O communications.
- Single ControlNet media is depicted; however, dual ControlNet media configurations are permitted.
- A non-redundant controller configuration is depicted; however, redundant configurations are also permitted.

### 3.2.1 Experion function block architecture

The JAGXTREME integration solution includes two Experion function blocks in the form of a Control Component Library (CCL). This provides a communication path from the JAGXTREME terminals, including a set of function blocks that allow other Experion blocks to utilize input/output data from the terminal connected scales and flow meters.

### 3.2.2 Data flow model

The following diagram depicts the data flow through the various components and blocks.

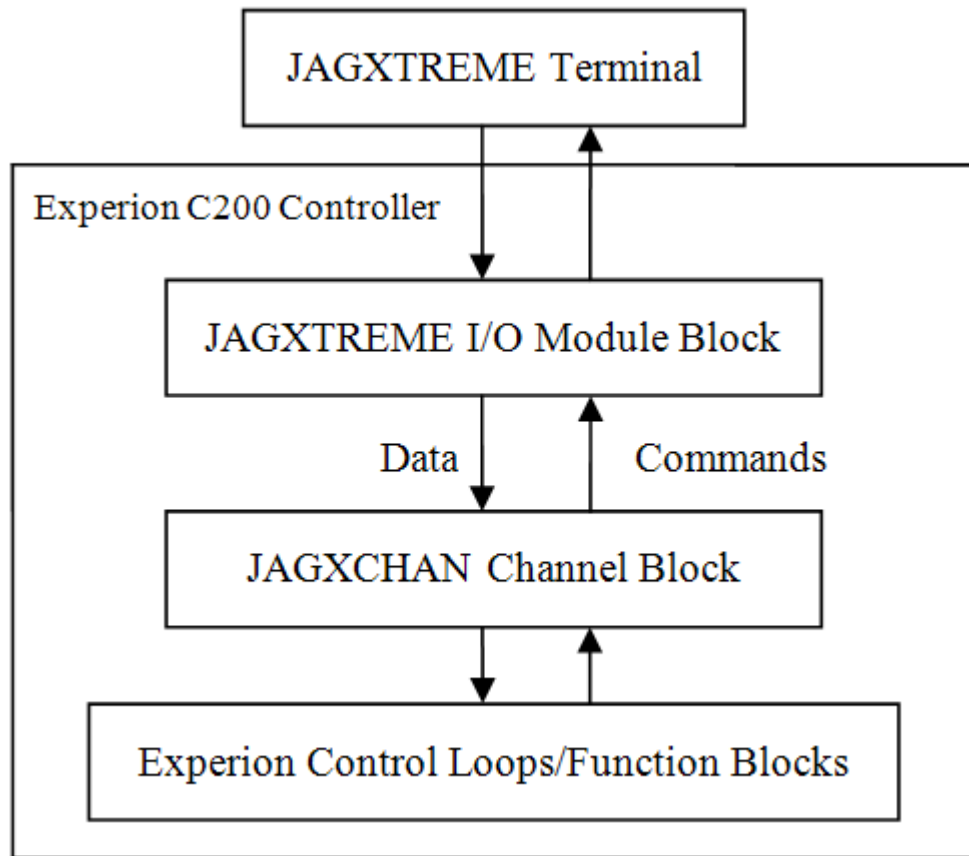


Figure 2: JAGXTREME Integration Solution Data Flow

### 3.2.3 QIPACTERM block

The METTLER TOLEDO Q.iMPACT terminal IOM function block represents a connected Q.iMPACT terminal. It resides in the I/O heading on the Experion Control Builder Project and Monitoring tabs. The Q.iMPACT QIPACTERM block supports the PAC input data assembly. The input assembly provides up to 24 channels of input data from a Q.iMPACT terminal. The channel information can be for any instrument connected to any terminal in the Q.iMPACT cluster.

The channel information provided by a Q.iMPACT terminal and its location within the input assembly is configured via the terminal's web server using a web browser. The input data for each channel consists of channel number, two words of status information, three weight and rate values, and two time values. This information is then fetched by the channel block to be used in a control strategy. The terminal blocks also receive data from the control strategy consisting of a command word and optional information depending on the command being issued. This data is sent to the terminal.

### 3.2.4 QIPACCHAN block

The QIPACCHAN channel function block fetches real-time data from the QIPACTERM block for use in control strategies. The input data consists of the information within the Q.iMPACT status words, and the analog weight, rate, and time parameters. This block also provides parameters for initiating and monitoring material transfer commands.

The QIPACCHAN function block executes as a dependent block at the base period of the containing control module. The block fetches the most recent input data from the QIPACTERM IOM function block for all 24 channels that can be returned in the input assembly from the associated Q.iMPACT. These channels can be any

of the 200 channels that can exist in a Q.iMPACT cluster. Therefore, the block allows all parameters except the CHANNEL parameter to be accessed using the Q.iMPACT channel number [1..200]. Thus, control applications can be configured using the real channel numbers as they exist in the Q.iMPACT cluster. The block takes care of finding the data for the requested channel within the input assembly. If the requested channel is not part of the input assembly, an error is returned.

The QIPACCHAN block also provides parameters used to initiate and monitor material transfer functions. The material transfer interface parameters are used to set up and initiate material transfers. The control application should set up all of the supporting data for a transfer such as material path and target weight first. When ready, it should then write the appropriate command value to the COMMAND parameter. An appropriate message will be assembled and sent to the Q.iMPACT for the specified command. Some of the material transfer status parameters provide feedback on the progress of a material transfer and are periodically updated while a transfer is in progress. Others provide results data after the material transfer is complete.

There can be a material transfer running on each Q.iMPACT cluster channel simultaneously. However, there can only be one material transfer running on a particular channel.

## 3.3 Q.iMPACT integration setup

### Related topics

- “Software environment requirements” on page 14
- “System Performance, Capacity, and Topology” on page 14
- “Configuration tools” on page 14
- “Update ControlNet settings” on page 14
- “Hardware installation” on page 15
- “Q.iMPACT terminal configuration” on page 15

### 3.3.1 Software environment requirements

The METTLER TOLEDO JAGXTREME Interface CCL is included in the Control Builder application, release 210 and later.

Users must purchase a license to be able to use the JAGXTREME integration products.

### 3.3.2 System Performance, Capacity, and Topology

The following specifications apply to the system topology.

- The C200 will support 16 terminals. Since each Q.iMPACT terminal uses one connection, each terminal has the weight of one I/O Module.
- The maximum number of terminals per downlink CNI is four. Therefore, four CNI modules are needed to support 16 Q.iMPACT terminals.
- Each physical I/O ControlNet segment can only support either scheduled or unscheduled communications but not both. Therefore, downlink CNIs can share a physical ControlNet segment only if the I/O associated with each downlink CNI uses the same scheduled or unscheduled communications.

### 3.3.3 Configuration tools

Each of the affected hardware components in the Experion/MT JAGXTREME architecture require the use of specific configuration tools as listed in the table below:

**Table 1: JAGXTREME Architecture Configuration Tools**

Hardware Component	Configuration tool
JAGXTREME Terminals	Front panel
C200 / CEE	Experion Control Builder
CNI	Experion Network Tools Utility

### 3.3.4 Update ControlNet settings

Using the Experion PKS Network Tools Application, the ControlNet's Keeper module must be configured as follows:

Maximum Scheduled Nodes (SMax): The highest MAC Address of all modules on the I/O ControlNet segment.

For information on using the Network Tools Utility, refer to 'NetworkTools' in the *Control Hardware Troubleshooting and Maintenance Guide*.

### 3.3.5 Hardware installation

Refer to Experion documentation for installation and configuration of the C200 and ControlNet hardware.

### 3.3.6 Q.iMPACT terminal configuration

The IOM function block currently provides no direct processing of Q.iMPACT terminal device configuration information. The Q.iMPACT terminals must be configured using their web server feature via a PC based web browser.

## 3.4 Configure a QIPACTERM block

The following table describes various configuration and operation scenarios.

**Table 2: Configuration/Operation Scenarios for QIPACTERM I/O Module**

Operation	Discussion
Configure (instantiate) and load	<p>Same as configuration and load of other Experion I/O modules.</p> <ul style="list-style-type: none"> <li>When a new module is chosen from the <b>Library</b> tab, the templates are found in the Q.IMPACT library.</li> <li>When a new module is chosen from the menu, the templates are found under <b>File &gt; New &gt; I/O Modules &gt; QIMPACT &gt; QIPACTERM - Mettler Toledo Q.iIMPACT</b>.</li> </ul>
Delete	Same as deletion of other Experion I/O modules. When a QIPACTERM I/O module block is deleted, the I/O module function block closes the connection to the terminal, and the icon for the terminal block is removed from the <b>Monitor</b> tab in CB.
Reload	Same as reload of other Experion I/O modules.
Upload	Same as upload of other Experion I/O modules.
Update	Same as update of other Experion I/O modules.
Checkpoint and Restore	Same as checkpoint and restore of other Experion I/O modules.
Inactivation	<p>Produces the following results:</p> <ul style="list-style-type: none"> <li>The icon representing the module in the <b>Monitor</b> tab becomes blue.</li> <li>The <b>Detail Display</b> displays the INACTIVE state.</li> <li>There should be no fault alarms in Experion server (communication with the module is not disrupted).</li> </ul>
Inactivation of CEE containing QIPACTERM I/O modules	There should be no fault alarms in Experion server (communication with the module is not disrupted).

### Related topics

“Physical location” on page 16

“Q.iIMPACT terminal configuration” on page 15

“QIPACTERM configuration loading” on page 17

“QIPACTERM block parameters” on page 17

### 3.4.1 Physical location

As with other I/O module function blocks, the location of the physical JX terminal is specified through the configuration of the following three parameters:

- IOMSLOT**: Unused since the JX terminal sits directly on ControlNet and *not* in a 1756 chassis.
- ULCNBMAC**: MAC address of the JX terminal.
- DLCNBSLOT**: Slot number of the downlink CNI in the controller chassis.

### 3.4.2 Q.iIMPACT terminal configuration

The IOM function block currently provides no direct processing of Q.iIMPACT terminal device configuration information. The Q.iIMPACT terminals must be configured using their web server feature via a PC based web browser.



### 3.4.3 QIPACTERM configuration loading

Configuring (instantiating) a Q.IMPACTERM I/O module is the same procedure as for any other Experion system I/O module. You can drag and drop the module type (template) from the library tree into a CEE. Alternatively, you can do a **File > New > I/O Modules > QIMPACT > QIPACTERM - Mettler Toledo Q.iMPACT**. In the latter case, the instance will be created under the Unassigned section of the Library tree and you will have to use the CB assignment function to assign it to the CEE.

Loading the I/O module is done with standard CB functionality.

### 3.4.4 QIPACTERM block parameters

The QIPACTERM block supports all of the parameters typically associated with I/O module function blocks in the Experion system. In addition, it supports parameters that are specific to the QIPACTERM block itself.

In the Control Builder Parameter Reference, the "QIPACTERM parameters" section lists the supported parameters and their descriptions, with the parameter names linked to the detailed definitions.

## 3.5 Configure a QIPACCHAN block

The QIPACCHAN function block is configured in the same manner as other CB function blocks. You instantiate the block in a CM, assign the CM to the CEE, load, open the properties form for the QIPACCHAN block and enter values for parameters as required.

### 3.5.1 Assign the QIPACCHAN block

As with other Experion system I/O channel function blocks, the QIPACCHAN channel block must be associated with a specific I/O module block. The QIPACCHAN channel block can only be associated with the QIPACTERM block.

To perform this association, open the properties form for the QIPACCHAN block. On the **Main** tab, select **Mettler Toledo QiMPACT** in the **Module Type** list, select the desired QIPACTERM block in the **Module Name** list, and click **Assign Channel Block**.

Channel Block to IO Module Assignment

Module Name: QIPACTERM\_134    Module Type: Mettler Toledo Q.iMPACT

Assigned to Module . . . . . QIPACTERM\_134  
Assigned to Channel . . . . . 0

Channel Number	Channel Name
0	CM_136.QIPACCHANA

Assign Channel Block

Select an empty channel in the list box at right and press the "Assign Channel Block" button above.

Figure 3: Assigning the QIPACCHAN Block

### 3.5.2 QIPACCHAN block parameters

#### Related topics

- “QIPACCHAN parameters” on page 24
- “QIPACCHAN input parameters” on page 24
- “QIPACCHAN material transfer status parameters” on page 26
- “QIPACCHAN material transfer interface parameters” on page 26
- “QIPACTERM communications parameters” on page 27

## 3.6 Communication and fault behaviors

Table 3: Communication and Fault Behaviors

Fault	Behavior
Switchover	<p>Switchover while connected to a JX terminal produces the following behavior:</p> <ul style="list-style-type: none"> <li>• No loss of connection to the JX terminal</li> <li>• No communication fault alarms are generated for the module as a result of the switchover</li> </ul>
Cable fault	<p>A cable fault in a single media system or a dual cable fault in a dual media system results in the following behavior:</p> <ul style="list-style-type: none"> <li>• Communication to the JX terminal is lost <ul style="list-style-type: none"> <li>– The CB Monitor tree icon for the terminal block appears RED (if activated)</li> <li>– <b>Comm Failure</b> is displayed on the Detail Display of the <i>connected</i> terminal</li> <li>– Communication Failure alarms are generated for connected terminal I/O module blocks</li> </ul> </li> <li>• JAGXTERM function blocks ... <ul style="list-style-type: none"> <li>– Stop writing data to the terminal as long as the connection is bad</li> <li>– Clear all fault indications</li> <li>– Parameters go to fail-safe values</li> </ul> </li> </ul>
Cable fault cleared	<p>Restoration of the cable connection results in the following behavior:</p> <ul style="list-style-type: none"> <li>• Communication to the JX terminal is restored.</li> <li>• The Monitor tree icons for the terminal I/O module block appear green (connected and active).</li> <li>• <b>OK</b> is displayed on the Detail Display of the <i>connected</i> terminal.</li> <li>• Associated alarms return to normal for the <i>connected</i> terminal.</li> </ul>

## 3.7 Q.iMPACT operation and notifications

### Related topics

“Input processing” on page 20

“Status processing” on page 20

“Output processing” on page 20

“Notifications” on page 20

### 3.7.1 Input processing

- The JAGXTERM function block executes as an independent block at the base period of the containing CEE.
- The remainder of the processing described here only occurs if the function block is in the **ACTIVE** state.

The status of the communication path to the device is checked.

- If the communication path is **not** intact:
  - A notification is generated.
  - For the JAGXTERM function blocks: **NAN** is supplied for the values of **FEEDWEIGHT**, **GROSSWEIGHT**, and **RATE** values.
  - For Q.iMPACT IOM blocks: **0** is supplied for the Floating-Point Value (FPVALUERAW).
  - All **Boolean** parameters are set to **FALSE**.
- If the communication path **is** intact:
  - The most recent input data is obtained from the JX terminal.
  - This input data is assigned to the corresponding function block parameters.

### 3.7.2 Status processing

This block processes block-specific fault and alarm data.

The block processes communication fault data. A communications fault is generated if the JAGXTERM function block loses communication with its associated JX terminal.

### 3.7.3 Output processing

Output processing consists of the IOM function block sending commands it has received to the device. The output processing operations revolve around:

- Checking the communications path from the C200 to the device.
- For Q.iMPACT IOM blocks: sending the **CHANNEL**, **SEQUENCENUM**, **MATERIALPATH**, and **COMMAND** values to the target terminal. Depending on the **COMMAND**, the **GROUPNUMBER**, **OVERLAPNUM**, and **TARGETWEIGHT** values may also be sent.

### 3.7.4 Notifications

The following table lists the notifications that can be generated by this block.

**Table 4: Notifications for the JAGXTERM Block**

Type	Condition	Notification Message
Communication fault	The associated JAGXTERM function block loses communication with the JX terminal.	Communication Error
JX instrument fault	A fault condition occurred on the associated instrument.	JX Instrument Fault
JX instrument alarm	An alarm condition (usually a warning) occurred on the associated instrument.	JX Instrument Alarm

## 3.8 QIPACTERM block parameters

### 3.8.1 QIPACTERM parameters

The following table lists the names and descriptions of the QIPACTERM block parameters. The parameter names are linked to the detailed descriptions of the parameters in the *Control Builder Parameter Reference*.

**Table 5: QIPACTERM Block Parameters**

Parameter Name	Description
CHANNEL[1..24]	Channel number
COMINTEGRITY[1..24]	Communications data integrity
DATAOK[1..24]	Instrument data integrity OK
OVERCAPACITY[1..24]	Scale over capacity
UNDERZERO[1..24]	Scale under zero
SCALEMOTION[1..24]	Scale motion
CYCLEACTIVE[1..24]	Material transfer cycle active
FCE_OUTPUT[1..24]	Final control element output
AWAITINGACK[1..24]	Waiting for controller to acknowledge last material transfer/hand add complete
FEEDTYPE[1..24]	Feed type
MANUALMODE[1..24]	Manual-not auto-mode
GROSSWEIGHT[1..24]	Gross weight
FEEDOVERRIDE[1..24]	Feed override active
FEEDFAILED[1..24]	Feed failed
COMMERROR[1..24]	Communication error (within Q.iMPACT cluster)
WGTUNSTABLE[1..24]	Device stability warning
VERYUNSTABLE[1..24]	Very unstable device
ERRATICFLOW[1..24]	Too high or too low flow
3TIMESFLOW[1..24]	Three times average flow
RATEALARM[1..24]	Fast feed rate alarm
WAITOVLPREQ[1..24]	Wait for all overlap requests
DELAYPRIMARY[1..24]	Waiting to start primary overlapped feed
PRIMOVERLAP[1..24]	Primary overlapped feed in progress
SECOVERLAP[1..24]	Secondary overlapped feed in progress
FEEDWEIGHT[1..24]	Feed weight
GROSSWGTFEED[1..24]	Gross weight feed
RATE[1..24]	Rate of change of weight
SLOWSTEPTIMR[1..24]	Time until slow step timer expires in seconds
TIMETOFINISH[1..24]	Estimated time to complete in seconds
STATUS_1[1..24]	Status 1 byte
STATUS_2[1..24]	Status 2 Word
CBADREADS	Consecutive bad reads

Parameter Name	Description
TBADREADS	Total bad reads

## 3.9 QIPACCHAN block parameters

### 3.9.1 QIPACCHAN parameters

There are four categories of QIPACCHAN block parameters:

- “QIPACCHAN input parameters” on page 24
- “QIPACCHAN material transfer status parameters” on page 26
- “QIPACCHAN material transfer interface parameters” on page 26
- “QIPACCHAN communications parameters” on page 27

The following tables list the names and descriptions of the four categories of QIPACCHAN block parameters. The parameter names are linked to the detailed descriptions of the parameters in the *Control Builder Parameter Reference*.

Note that most of the array parameters are followed by a corresponding scalar parameter with similar spelling and that begins with the letter 'V.' For example, the COMINTEGRITY[1..200] parameter is followed by the VCOMINTEGRITY parameter. These 'V' (for 'Visual') parameters are displayed on the **Channel Data** tab of the block property form. These 'V' parameters allow you to view the values of their associated array parameters for a selected channel, which is specified by putting the desired channel number in the VSEL\_CHANNEL parameter.

### 3.9.2 QIPACCHAN input parameters

Table 6: QIPACCHAN Block Input Parameters

Parameter Name	Description
CHANNEL[1..24]	Q.iMPACT input assembly channel numbers
VSEL_CHANNEL	Selected channel
COMINTEGRITY[1..200]	Communications data integrity
VCOMINTEGRITY	Communications data integrity
DATAOK[1..200]	Instrument data integrity OK
VDATAOK	Instrument data integrity OK
OVERCAPACITY[1..200]	Scale over capacity
VOVERCAPACITY	Scale over capacity
UNDERZERO[1..200]	Scale under zero
VUNDERZERO	Scale under zero
SCALEMOTION[1..200]	Scale motion
VSCALEMOTION	Scale motion
CYCLEACTIVE[1..200]	Material transfer cycle active
VCYCLEACTIVE	Material transfer cycle active
FCE_OUTPUT[1..200]	Final control element output
VFCE_OUTPUT	Final control element output
AWAITINGACK[1..200]	Waiting for controller to acknowledge last material transfer/hand add complete
VAWAITINGACK	Waiting for controller to acknowledge last material transfer/hand add complete
FEEDTYPE[1..200]	Feed type



Parameter Name	Description
VFEEDTYPE	Feed type
MANUALMODE[1..200]	Manual-not-auto-mode
VMANUALMODE	Manual-not-auto-mode
GROSSWGTFEED[1..200]	Gross weight feed
VGROSSWGTFED	Gross weight feed
FEEDOVERRIDE[1..200]	Feed override active
VFEEDOVERRDE	Feed override active
FEEDFAILED[1..200]	Feed failed
VFEEDFAILED	Feed failed
COMMERROR[1..200]	Communication error (within Q.iMPACT cluster)
VCOMMERROR	Communication error (within Q.iMPACT cluster)
WAITSTABLE[1..200]	Device stability warning
VWGTUNSTABLE	Device stability warning
VERYUNSTABLE[1..200]	Very unstable device
VVERYUNSTBLE	Very unstable device
ERRATICFLOW[1..200]	Too high or too low flow
VERRATICFLOW	Too high or too low flow
3TIMESFLOW[1..200]	Three times average flow
V3TIMESFLOW	Three times average flow
RATEALARM[1..200]	Fast feed rate alarm
VRATEALARM	Fast feed rate alarm
WAITOVLPREQ[1..200]	Wait for all overlap requests
VWAITOVLPREQ	Wait for all overlap requests
DELAYPRIMARY[1..200]	Waiting to start primary overlapped feed
VDELAYPRIMRY	Waiting to start primary overlapped feed
PRIMOVERLAP[1..200]	Primary overlapped feed in progress
VPRIMOVERLAP	Primary overlapped feed in progress
SECOVERLAP[1..200]	Secondary overlapped feed in progress
VSECOVERLAP	Secondary overlapped feed in progress
FEEDWEIGHT[1..200]	Feed weight
VFEEDWEIGHT	Feed weight
GROSSWEIGHT[1..200]	Gross weight
VGROSSWEIGHT	Gross weight
RATE[1..200]	Rate of change of weight
VRATE	Rate of change of weight
SLOWSTEPTMR[1..200]	Time until slow step timer expires in seconds
VSLOWSTEPTMR	Time until slow step timer expires in seconds
TIMETOFINISH[1..200]	Estimated time to complete in seconds
VTIMETOFINSH	Estimated time to complete in seconds
MSGSEQNUMBER[1..200]	Message sequence number

Parameter Name	Description
VMSGSEQNUMBR	Message sequence number (this parameter is not displayed on the Channel Data tab)

### 3.9.3 QIPACCHAN material transfer status parameters

Table 7: QIPACCHAN Block Material Transfer Parameters

Parameter Name	Description
MATERIALPATH[1..200]	Material path index
VMATERIALPTH	Material path index
COMMAND[1..200]	Command number
VCOMMAND	Command number
COMMANDSTATS[1..200]	Command status from Q.iMPACT
VCOMMANDSTAT	Command status from Q.iMPACT
MXFERSTATUS[1..200]	Material transfer status from Q.iMPACT
VMXFERSTATUS	Material transfer status from Q.iMPACT
MXFERSTATQ[1..200]	Material transfer status qualifiers integer representation
VMXFERSTATQ	Material transfer status qualifiers integer representation
OVERTOLRNCE[1..200]	Over tolerance
VOVERTOLRNCE	Over tolerance
UNDERTOLRNCE[1..200]	Under tolerance
VUNDERTOLRNC	Under tolerance
POWERFAIL[1..200]	Power failure
VPOWERFAIL	Power failure
DELIVEREDWGT[1..200]	Delivered weight
VDELIVEREDWT	Delivered weight
TARGETERROR[1..200]	Deviation error from target weight
VTARGETERROR	Deviation error from target weight
AVGFLOWRATE[1..200]	Average flow rate
VAVGFLOWRATE	Average flow rate
AVGSPILL[1..200]	Average spill
VAVGSPILL	Average spill

### 3.9.4 QIPACCHAN material transfer interface parameters

Table 8: QIPACCHAN Block Material Transfer Interface Parameters

Parameter Name	Description
VMXCHANNEL	Channel number
MXMATPATH[1..200]	Material path index
VMXMATPATH	Material path index
MXCOMMAND[1..200]	Command number
VMXCOMMAND	Command number

Parameter Name	Description
MXGROUPNUM[1..200]	Group number for start material transfer command
VMXGROUPNUM	Group number for start material transfer command
MXOVLAPNUM[1..200]	Number of overlapping secondary feeds
VMXOVLAPNUM	Number of overlapping secondary feeds
MXTARGETWGT[1..200]	Target weight
VMXTARGETWGT	Target weight
MXPOSTOLER[1..200]	Tolerance +
VMXPOSTOLER	Tolerance +
MXNEGTOLER[1..200]	Tolerance -
VMXNEGTOLER	Tolerance -
MXTRANID[1..200]	Material transfer ID
VMXTRANID	Material transfer ID

### 3.9.5 QIPACTERM communications parameters

Table 9: QIPACTERM Block Communications Parameters

Parameter Name	Description
CNISLOT	CNI slot number
QIMPACTMAC	QiIMPACT CNET MAC address
LASTSTAT	Status last transaction
LASTEXTSTAT	Ext. status last transaction
RETRYSCURR	Retries last transaction
RETRYSTOTAL	Retries total
COMMSTATUS	Communications status



## 4 METTLER TOLEDO JAGXTREME integration

### Related topics

- “Overview of JAGXTREME in Experion” on page 30
- “Architecture with JAGXTERM and JAGXCCHAN blocks” on page 31
- “JAGXTREME integration setup” on page 34
- “Configure a JAGXTERM block” on page 35
- “Configure a JAGXCHAN block” on page 37
- “Communication and fault behaviors” on page 19
- “JAGXTERM operation and notifications” on page 39
- “JAGXTERM parameters” on page 40
- “JAGXCHAN parameters” on page 41

---

## 4.1 Overview of JAGXTREME in Experion

The information contained in this section is intended to provide a brief overview of the technology involved with integrating ControlNet-connected METTLER TOLEDO JAGXTREME (JX) terminals with the Experion system.

### 4.1.1 JAGXTREME terminal support

JAGXTREME terminals can be configured to communicate over ControlNet using one of these three types of messaging: Integer Assembly, Extended Integer Assembly, and Floating-point Assembly messaging. Experion supports only the Floating-point Assembly format. This format includes all of the functionality of the integer formats and provides for greater accuracy of the values communicated.

The communications protocol is bidirectional. This allows the C200 controller to issue cyclic commands to the JX terminals as well as receive cyclic data from them. Each JX terminal cyclically communicates an input assembly message to the C200 controller that contains up to four scale slots of data. The number of slots in each message that contain applicable data depends on the number of devices (scales and flow meters) connected, and on the configuration of the JX terminal. Each JX terminal has three option slots that can accept a ControlNet interface card, a load cell interface card, or a flow meter interface card. Each load cell interface card supports one or two load cells. Each flow meter interface card supports up to four flow meters.

In order to provide an easy-to-use integration solution, two function blocks have been implemented. The first is the IOM (I/O Module) function block. This block is responsible for establishing, maintaining, and monitoring communications with a JX terminal. The user must configure an IOM function block for each JX terminal. The second block type is the channel block. It makes the data received from the JX terminal available to controller applications, and handles commands directed to the JX terminals from controller applications. The user must configure a channel function block for each JX terminal.

Calculations indicate that a CNI card in an Experion C200 chassis can support eight JX terminals.

## 4.2 Architecture with JAGXTERM and JAGXCCHAN blocks

The following diagram illustrates the topology of the integration solution.

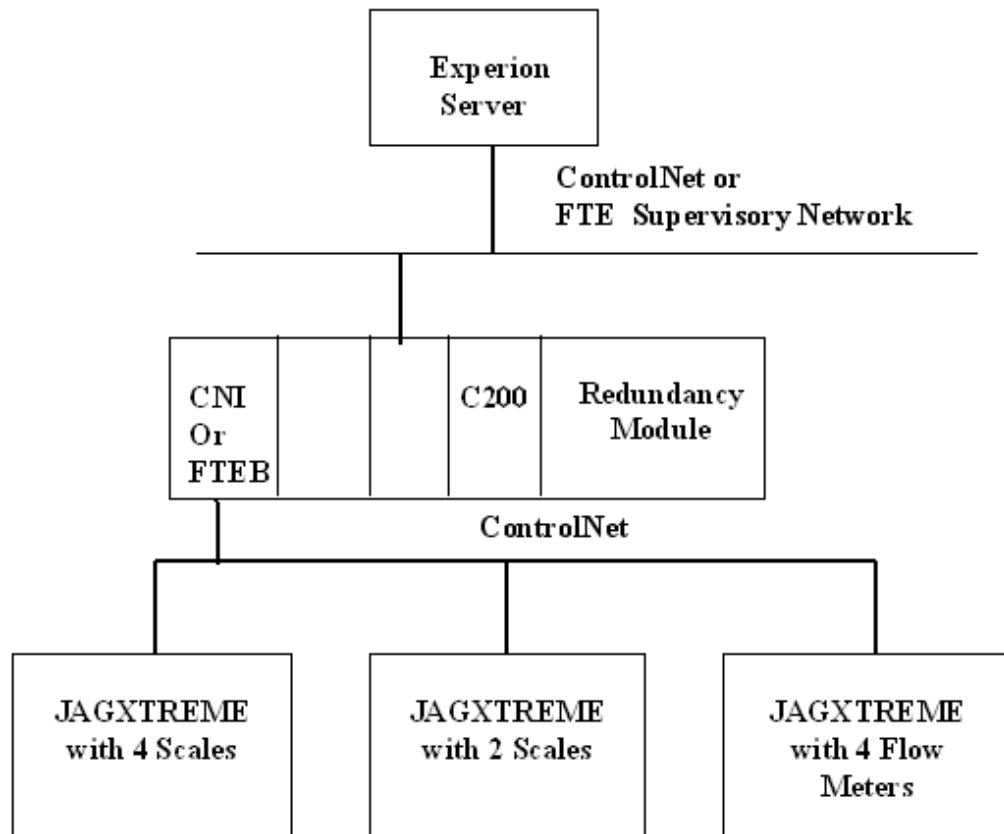


Figure 4: JAGXTREME Integration Solution Topology

### Topology diagram notes

- Only one Experion C200 controller can be connected to the JX ControlNet. In addition, the JX terminals must reside on an I/O ControlNet segment that is physically isolated from unscheduled I/O communications.
- Single ControlNet media is depicted; however, dual ControlNet media configurations are permitted.
- A non-redundant controller configuration is depicted; however, redundant configurations are also permitted.

### 4.2.1 Experion function block architecture

The JAGXTREME integration solution includes two Experion function blocks in the form of a Control Component Library (CCL). This provides a communication path from the JAGXTREME terminals, including a set of function blocks that allow other Experion blocks to utilize input/output data from the terminal connected scales and flow meters.

### 4.2.2 Data flow model

The following diagram depicts the data flow through the various components and blocks.

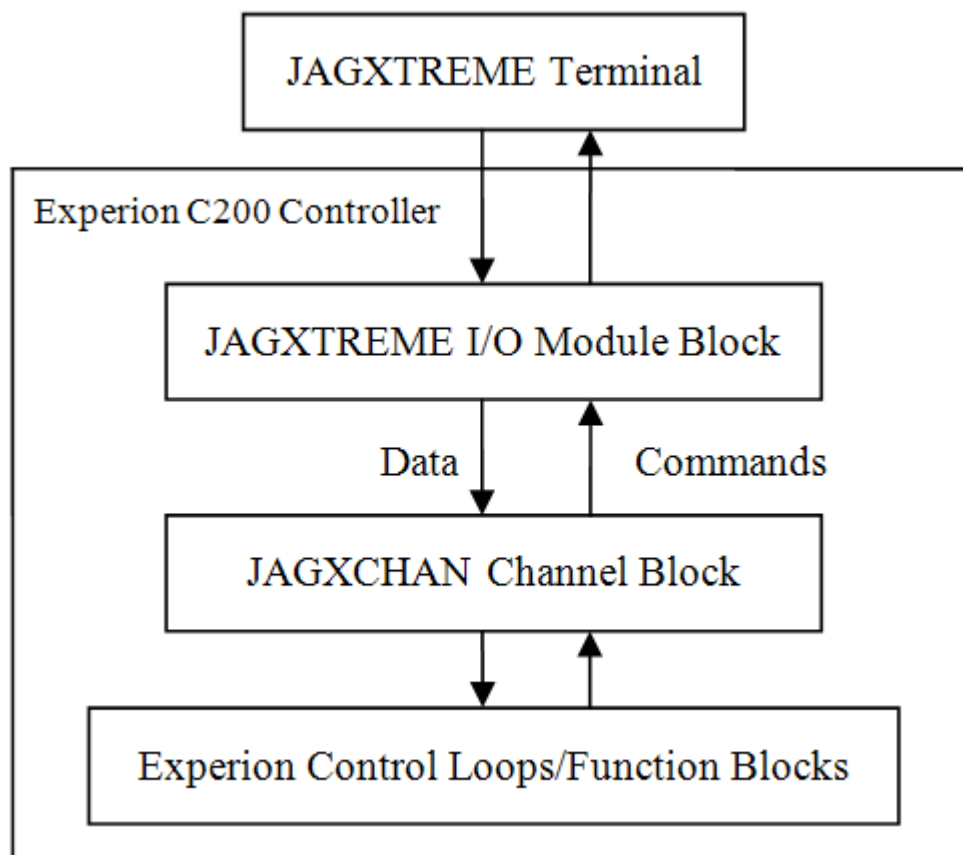


Figure 5: JAGXTREME Integration Solution Data Flow

### 4.2.3 JAGXTERM block

The JAGXTERM block represents the associated JAGXTREME hardware device. From the user perspective, this block is similar to an I/O module and resides in the I/O heading on the Experion PKS Control Builder *Project* and *Monitoring* tabs.

The JAGXTERM IOM block supports the floating-point input data assembly. This input assembly provides up to four channels of input data from a JX terminal. The channel information can be for any instrument connected to the terminal. The channel information provided by a JX terminal and its location within the input assembly is configured via the terminal's front panel and by commands that can be issued from the Experion C200 controller. The input data for each channel consists of the floating-point indicator byte, the floating-point value, and the status word. The floating-point indicator specifies which of 32 different values is represented by the floating-point value. This information is then fetched by the JAGXCHAN block and is made available to control strategies within the Experion C200 controller.

### 4.2.4 JAGXCHAN block

The JAGXCHAN function block fetches real-time data from an associated JAGXTERM function block, parses that data into parameters immediately usable by control strategies, and provides a path for control functions to issue commands to the JX terminals.

The input data consists of information from the floating-point indicator byte, the status word, and the floating-point value.

As with other Experion PKS I/O channel function blocks, the JAGXCHAN block must be associated with a specific I/O module block. The JAGXCHAN channel block can only be associated with the JAGXTERM block.



The JAGXCHAN function block executes as a dependent block at the base period of the containing CM. The block fetches the most recent input data from its associated JAGXTERM IOM function block every execution.

The JAGXTERM IOM function block executes as an independent block at the base period of the containing CEE if the function block is in the ACTIVE state. The data for all four-scale slots is extracted from the input assembly and inserted into the appropriate parameters. Individual Boolean values and enumerations are made available via specific parameter names to control strategies.

This interface does not support JX input rotation. However, a user application could be written to send and monitor the appropriate commands to implement this feature. This interface merely makes the current data being provided available through its parameter set.

Command processing consists of the JAGXCHAN block forwarding commands it has received to its associated JX terminal. The parameters CMDREADY, COMMAND, and CMDFPVALUE are provided for this purpose as described in the next section. To issue a command, the control strategy or user must follow the following steps:

1. Check that the CMDREADY parameter is TRUE indicating that there is no command in progress for the scale.
2. Insert a value into the CMDFPVALUE parameter if required for the command being issued.
3. Write the appropriate command value to the COMMAND parameter. Writing to the COMMAND parameter triggers the interface to insert the new command information to the periodic output assembly sent to the JX terminal. The CMDREADY parameter will be FALSE for the short period of time it takes to insert the new data into the output assembly.

## 4.3 JAGXTREME integration setup

### Related topics

“Software environment requirements” on page 14

“System performance, capacity, and topology specifications” on page 34

“Configuration tools” on page 14

“Update ControlNet settings” on page 14

### 4.3.1 Software environment requirements

The METTLER TOLEDO JAGXTREME Interface CCL is included in the Control Builder application, release 210 and later.

Users must purchase a license to be able to use the JAGXTREME integration products.

### 4.3.2 System performance, capacity, and topology specifications

The specifications below are based on extrapolated maximums. Actual maximums could be different and should be verified by the user by appropriate testing.

1. The C200 will support 32 JX terminals. Since each JX terminal uses one connection, each terminal has the weight of one I/O Module. This must be considered during system configuration.
2. The maximum number of JX terminals per downlink CNI is eight. Therefore, a total of four CNI modules are needed to support 32 JX terminals.
3. Each physical I/O ControlNet segment can support either scheduled or unscheduled communications but not both. Therefore, downlink CNIs can share a physical ControlNet segment only if the I/O associated with each downlink CNI uses the same scheduled or unscheduled communications.

### 4.3.3 Configuration tools

Each of the affected hardware components in the Experion/MT JAGXTREME architecture require the use of specific configuration tools as listed in the table below:

**Table 10: JAGXTREME Architecture Configuration Tools**

Hardware Component	Configuration tool
JAGXTREME Terminals	Front panel
C200 / CEE	Experion Control Builder
CNI	Experion Network Tools Utility

### 4.3.4 Update ControlNet settings

Using the Experion PKS Network Tools Application, the ControlNet's Keeper module must be configured as follows:

Maximum Scheduled Nodes (SMax): The highest MAC Address of all modules on the I/O ControlNet segment.

For information on using the Network Tools Utility, refer to 'NetworkTools' in the *Control Hardware Troubleshooting and Maintenance Guide*.

## 4.4 Configure a JAGXTERM block

The following table lists the various configuration and operation scenarios.

**Table 11: Configuration and Operation Scenarios**

Operation	Discussion
Configure (instantiate) and load	Configuration and load of the JAGXTREME I/O Module is the same as configuration and load of other Experion I/O modules. <ul style="list-style-type: none"> <li>When a new module is chosen from the <b>Library</b> tab, the templates are found in the JAGXTREME library.</li> <li>When a new module is chosen from the menu, the templates are found under <b>File &gt; New &gt; I/O Modules &gt; JAGXTREME &gt; JAGXTERM - Mettler Toledo - JAGXTREME</b>.</li> </ul>
Delete	Deletion of the JAGXTERM and JAGXCHAN blocks is the same as deletion of other Experion blocks. When a JAGXTERM I/O module block is deleted, the I/O module function block closes the connection to the terminal, and the icon for the terminal block is removed from the <b>Monitor</b> tab in CB.
Reload	Reload of the JAGXTERM and JAGXCHAN blocks is the same as reload of other Experion blocks.
Upload	Upload of the JAGXTERM and JAGXCHAN blocks is the same as upload of other Experion blocks.
Update	Update of the JAGXTERM and JAGXCHAN blocks is the same as update of other Experion blocks.
Checkpoint and Restore	Checkpoint and restore of the JAGXTERM and JAGXCHAN blocks is the same as checkpoint and restore of other Experion blocks.
Inactivation of JAGXTERM I/O module	Produces the following results: <ul style="list-style-type: none"> <li>The icon representing the module in the <b>Monitor</b> tab becomes blue.</li> <li>The <b>Detail Display</b> displays the INACTIVE state.</li> <li>There should be no fault alarms in Experion server (communication with the module is not disrupted).</li> </ul>
Inactivation of a CEE containing JAGXTREME I/O modules	There should be no fault alarms in Experion server (communication with the module is not disrupted).

### Related topics

“Physical location” on page 16

“JAGXTREME terminal configuration” on page 36

“JAGXTERM configuration loading” on page 36

“JAGXTERM block parameters” on page 36

### 4.4.1 Physical location

As with other I/O module function blocks, the location of the physical JX terminal is specified through the configuration of the following three parameters:

- IOMSLLOT**: Unused since the JX terminal sits directly on ControlNet and *not* in a 1756 chassis.
- ULCNBMAC**: MAC address of the JX terminal.
- DLCNBSLOT**: Slot number of the downlink CNI in the controller chassis.

## 4.4.2 JAGXTREME terminal configuration

Refer to METTLER TOLEDO documentation for information about configuring the JAGXTREME terminal. The JAGXTREME terminal is configured from the front panel.

## 4.4.3 JAGXTERM configuration loading

Configuring (instantiating) a JAGXTREME I/O module is the same procedure as for any other Experion system I/O module. You can drag and drop the module type (template) from the library tree into a CEE. Alternatively, you can do a **File > New > I/O Modules > JAGXTREME > JAGXTERM - Mettler Toledo - JAGXTREME**. In the latter case, the instance will be created under the Unassigned section of the Library tree and you will have to use the CB assignment function to assign it to the CEE.

Loading the I/O module is done with standard CB functionality.

## 4.4.4 JAGXTERM block parameters

The JAGXTERM block supports all of the parameters typically associated with I/O module function blocks, in the Experion system. In addition, there are parameters specific to the JAGXTERM block.

The parameters specific to the JAGXTERM function block are provided as debugging aids. The three values returned in the input assembly for each of the four scale slots are presented as received. The floating-point indicator byte (FPINDRAW) and the status word (STATUSRAW) are presented as integers. The floating-point value (FPVALUERAW) is displayed as a floating-point number. The parameters on this block are not meant to be used by control strategies. Those defined as part of the JAGXCHAN block should be used. It contains many more parameters that provide the current value of the returned data in a more detailed format. For instance, each status bit in the status word is presented as a separate BOOLEAN parameter.

The section '“JAGXTERM parameters” on page 40' lists the supported parameters and their descriptions, with the parameter names linked to the detailed definitions in the Control Builder Parameter Reference.

## 4.5 Configure a JAGXCHAN block

The JAGXCHAN function block is configured in the same manner as other CB function blocks. You instantiate the block in a CM, assign the CM to the CEE, load, open the properties form for the JAGXCHAN block and enter values for parameters as required.

**Related topics**

“Assign the JAGXCHAN block” on page 37

“JAGXCHAN block parameters” on page 37

### 4.5.1 Assign the JAGXCHAN block

As with other Experion system I/O channel function blocks, the JAGXCHAN channel block must be associated with a specific I/O module block. The JAGXCHAN channel block can only be associated with the JAGXTERM block.

To perform this association, open the properties form for the JAGXCHAN block. On the **Main** tab, select **Mettler Toledo JAGXTREME** in the **Module Type** list, select the desired JAGXTERM block in the **Module Name** list, and click **Assign Channel Block**.

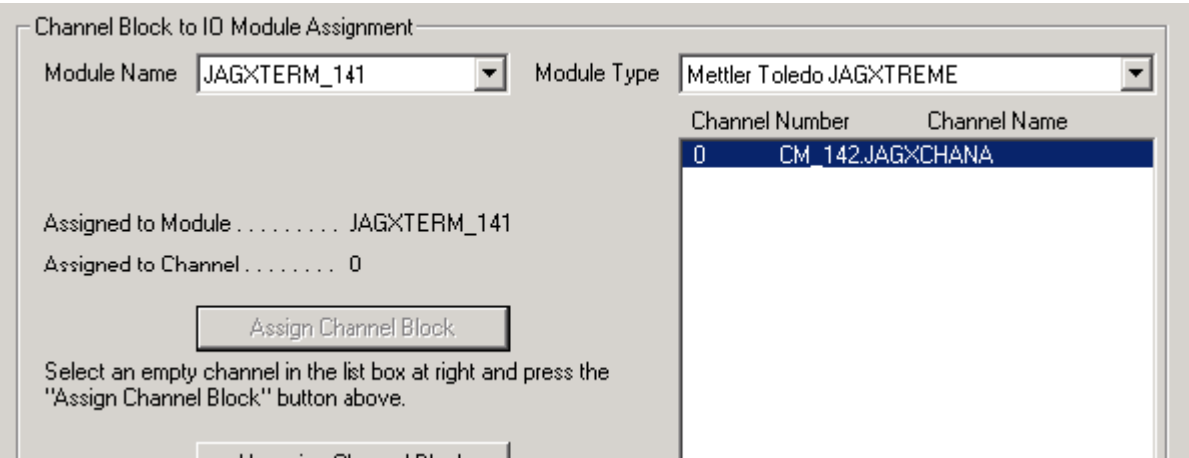


Figure 6: Assigning the JAGXCHAN Block

### 4.5.2 JAGXCHAN block parameters

The JAGXCHAN input channel block supports all of the parameters typically associated with I/O channel function blocks in the Experion system. In addition, it supports parameters that are specific to the QIPACCHAN channel block itself.

In the *Control Builder Parameter Reference*, the sections '“JAGXCHAN input parameters” on page 41' and '“JAGXCHAN command parameters” on page 42' list the supported parameter names and their descriptions, with the parameter names linked to the detailed definitions.

## 4.6 Communication and fault behaviors

Table 12: Communication and Fault Behaviors

Fault	Behavior
Switchover	<p>Switchover while connected to a JX terminal produces the following behavior:</p> <ul style="list-style-type: none"> <li>No loss of connection to the JX terminal</li> <li>No communication fault alarms are generated for the module as a result of the switchover</li> </ul>
Cable fault	<p>A cable fault in a single media system or a dual cable fault in a dual media system results in the following behavior:</p> <ul style="list-style-type: none"> <li>Communication to the JX terminal is lost <ul style="list-style-type: none"> <li>The CB Monitor tree icon for the terminal block appears RED (if activated)</li> <li><b>Comm Failure</b> is displayed on the Detail Display of the <i>connected</i> terminal</li> <li>Communication Failure alarms are generated for connected terminal I/O module blocks</li> </ul> </li> <li>JAGXTERM function blocks ... <ul style="list-style-type: none"> <li>Stop writing data to the terminal as long as the connection is bad</li> <li>Clear all fault indications</li> <li>Parameters go to fail-safe values</li> </ul> </li> </ul>
Cable fault cleared	<p>Restoration of the cable connection results in the following behavior:</p> <ul style="list-style-type: none"> <li>Communication to the JX terminal is restored.</li> <li>The Monitor tree icons for the terminal I/O module block appear green (connected and active).</li> <li><b>OK</b> is displayed on the Detail Display of the <i>connected</i> terminal.</li> <li>Associated alarms return to normal for the <i>connected</i> terminal.</li> </ul>

## 4.7 JAGXTERM operation and notifications

### Related topics

“Input processing” on page 20

“Status processing” on page 20

“Notifications” on page 20

### 4.7.1 Input processing

- The JAGXTERM function block executes as an independent block at the base period of the containing CEE.
- The remainder of the processing described here only occurs if the function block is in the **ACTIVE** state.

The status of the communication path to the device is checked.

- If the communication path is **not** intact:
  - A notification is generated.
  - For the JAGXTERM function blocks: **NAN** is supplied for the values of **FEEDWEIGHT**, **GROSSWEIGHT**, and **RATE** values.
  - For Q.iMPACT IOM blocks: **0** is supplied for the Floating-Point Value (FPVALUERAW).
  - All **Boolean** parameters are set to **FALSE**.
- If the communication path **is** intact:
  - The most recent input data is obtained from the JX terminal.
  - This input data is assigned to the corresponding function block parameters.

### 4.7.2 Status processing

This block processes block-specific fault and alarm data.

The block processes communication fault data. A communications fault is generated if the JAGXTERM function block loses communication with its associated JX terminal.

### 4.7.3 Notifications

The following table lists the notifications that can be generated by this block.

**Table 13: Notifications for the JAGXTERM Block**

Type	Condition	Notification Message
Communication fault	The associated JAGXTERM function block loses communication with the JX terminal.	Communication Error
JX instrument fault	A fault condition occurred on the associated instrument.	JX Instrument Fault
JX instrument alarm	An alarm condition (usually a warning) occurred on the associated instrument.	JX Instrument Alarm

# 4.8 JAGXTERM parameters

The following table lists the names and descriptions of the JAGXTERM block parameters. The parameter names are linked to the detailed descriptions of the parameters in the *Control Builder Parameter Reference*.

Table 14: JAGXTERM Block Parameters

Parameter Name	Description
<a href="#">FPINDRAW[1..4]</a>	Floating-point indicator
<a href="#">FPVALUERAW[1..4]</a>	Floating-point value
<a href="#">STATUSRAW[1..4]</a>	Status word



## 4.9 JAGXCHAN parameters

There are two categories of JAGXCHAN block parameters:

- “JAGXCHAN input parameters” on page 41
- “JAGXCHAN command parameters” on page 42

The following tables list the names and descriptions of the three categories of JAGXCHAN block parameters. The parameter names are linked to the detailed descriptions of the parameters in the *Control Builder Parameter Reference*.

### Related topics

“JAGXCHAN input parameters” on page 41

“JAGXCHAN command parameters” on page 42

### 4.9.1 JAGXCHAN input parameters

Table 15: JAGXCHAN Block Input Parameters

Parameter name	Description
FPINDICATOR[1..4]	Floating-point input indicator
INTEGRITY1[1..4]	Data integrity bit 1
COMMANDACK[1..4]	Command acknowledge
FPVALUE[1..4]	Floating-point value
SP1FEEDING[1..4]	Set point 1 feeding
SP2FEEDING[1..4]	Set point 2 feeding
SP1FASTFEED[1..4]	Set point 1 fast feeding
SP2FASTFEED[1..4]	Set point 2 fast feeding
SP1TOLERANCE[1..4]	Set point 1 in tolerance
SCALESELECTD[1..4]	Scale selected (weight on local display)
JAGBASIC1[1..4]	JagBasic custom bit 1
JAGBASIC2[1..4]	JagBasic custom bit 2
ESCKEYSTROKE[1..4]	Escape keystroke
DISCRTINPUT1[1..4]	Discrete input 1
DISCRTINPUT2[1..4]	Discrete input 2
DISCRTINPUT3[1..4]	Discrete input 3
MOTIONBIT[1..4]	Motion bit
NETMODE[1..4]	Net weight mode
INTEGRITY2[1..4]	Data integrity bit 2
DATAOK[1..4]	Data OK/Error indication

4.9.2 JAGXCHAN command parameters

Table 16: JAGXCHAN Block Command Parameters

Parameter name	Description
COMMAND	Scale command
CMDFPVALUE	Floating-point value
CMDREADY	Command interface ready flag