

# **ABB Totalflow Driver**

**©2016 Kepware, Inc.**

# Table of Contents

<b>Table of Contents</b> .....	<b>2</b>
ABB Totalflow Driver .....	4
Overview .....	4
<b>Device Setup</b> .....	<b>5</b>
Settings .....	5
EFM Meters .....	8
Trend File Configuration .....	10
Automatic Tag Database Generation .....	12
<b>Data Types Description</b> .....	<b>13</b>
<b>Address Descriptions</b> .....	<b>14</b>
Characteristic Address Descriptions .....	14
DB2 Protocol Address Descriptions .....	18
Trend Address Descriptions .....	19
Statistics Items .....	21
<b>Error Descriptions</b> .....	<b>23</b>
A communication error occurred while <reading/writing> address <tag address> on device <device name>: <verbose communication error> .....	24
A communication error occurred while reading address <tag address> on device <device name>: Device did not respond. ....	24
A communication error occurred while reading address <tag address> on device <device name>: General remote NACK error. ....	24
A communication error occurred while reading address <tag address> on device <device name>: Illegal register read or write. ....	24
A communication error occurred while reading address <tag address> on device <device name>: Invalid data structure. ....	24
A communication error occurred while reading address <tag address> on device <device name>: Received data had a CRC error. ....	25
A communication error occurred while reading address <tag address> on device <device name>: Transaction ID did not match. ....	25
Bit index out of range for register address <tag address> on device <device name>. ....	25
Could not write data for trend file <trend file> on device <device> to disk: <reason>. ....	25
<Device name> - Failed to read EFM pointer file. <Extended error>. ....	26
<Device name> - Failed to read trend pointer file. <Extended error>. ....	26
<Device name> - Failed to write EFM pointer file. <Extended error>. ....	27
<Device name> - Failed to write trend pointer file. <Extended error>. ....	27
Device <device name> is not responding. ....	27
EFM meter <meter name> on device <device name> is invalid. ....	28
EFM type mismatch for meter <meter name> on device <device name>. The meter is configured for <meter type> in the server but it is a <meter type> meter in the device. ....	28
Extra data revision not supported. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>. ....	28
Extra data size is 0 bytes. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>. ....	28
Failed to send communication request for address <tag address> on device <device name>. ....	29

Invalid address for register block <App>.<Array>.<Register X> - <App>.<Array>.<Register Y> on device <device name>. ....	29
Invalid data type <data type> for address <tag address> for device <device name>. ....	29
Register blocking error, verify that data type <data type> is valid for address <tag address> on device <device name>. ....	29
Serialization of EFM data to temporary file <file name> failed. Reason: <file I/O error>. ....	29
Trend file configuration changed for trend file <trend file> on device <device>. Old data moved to <file name>. New data written to <file name>. ....	30
Trend file <trend file name> does not exist on device <device name>. ....	30
Trend upload expected new records for trend file <trend file> on device <device name>, but no records were retrieved. ....	30
Unable to load TCI Toolkit extradata.ini file, EFM feature may not work properly. ....	30
Unable to write to <address> on device <device name>. ....	31
<b>Index</b> .....	<b>32</b>

### ABB Totalflow Driver

---

Help version 1.036

#### CONTENTS

##### [Overview](#)

What is the ABB Totalflow Driver?

##### [Device Setup](#)

How do I configure devices for use with this driver?

##### [Data Types Description](#)

What data types does this driver support?

##### [Address Descriptions](#)

How do I address a data location on an ABB Totalflow device?

##### [Error Descriptions](#)

What error messages are produced by the ABB Totalflow Driver?

### Overview

---

The ABB Totalflow Driver is designed to work with ABB Totalflow devices that support the native DB1 and DB2 Serial Protocols (which are typically used by ABB's flow computers and analyzers). ABB's Totalflow Communication Interface (TCI) toolkit is used to implement the application stack for the driver.

**Important:** EFM functionality is not available in all server versions. To determine whether support is available, refer to the "Server Summary Information" topic located in the server help file.

## Device Setup

### Supported Device Families

The following device families are supported under the "Totalflow FCU" model:

6000 Series MicroFLO  
 6000 XSeriesG4  
 6000 XSeriesG3  
 6000 Series FCU

### Maximum Number of Channels and Devices

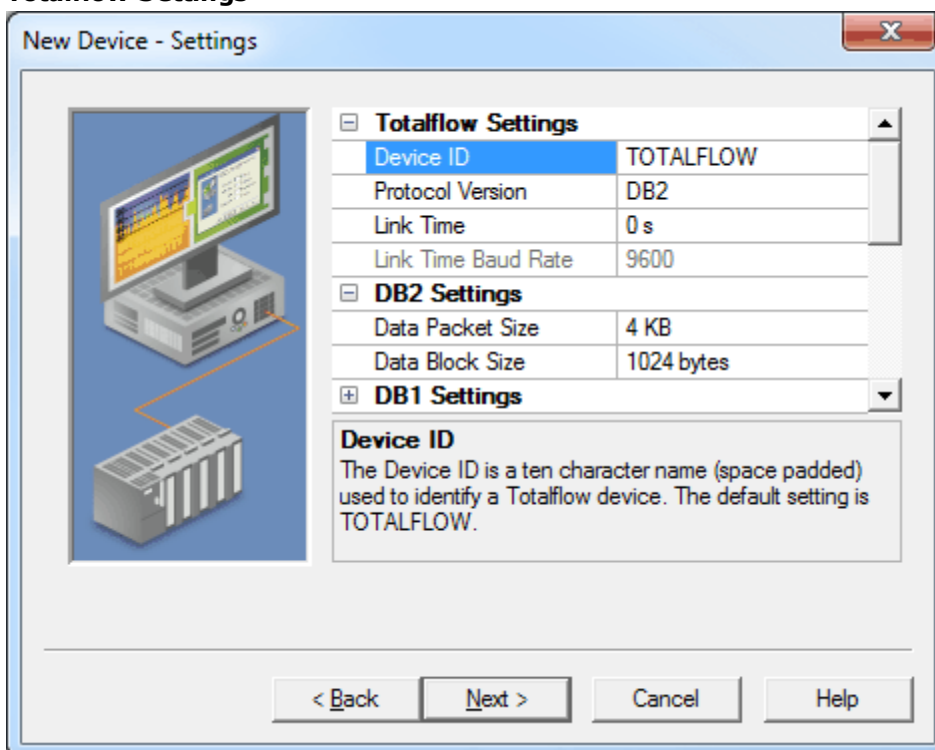
The maximum number of supported channels is 1024. The maximum number of supported devices is 256.

### Ethernet Encapsulation

This driver supports Ethernet Encapsulation, which allows the driver to communicate with serial devices attached to an Ethernet network using a serial-to-Ethernet terminal server. It also allows the driver to directly communicate with a device that is equipped with a TCP/IP port. It may be invoked through the Communications tab in Channel Properties. For more information, refer to the server help file.

## Settings

### Totalflow Settings

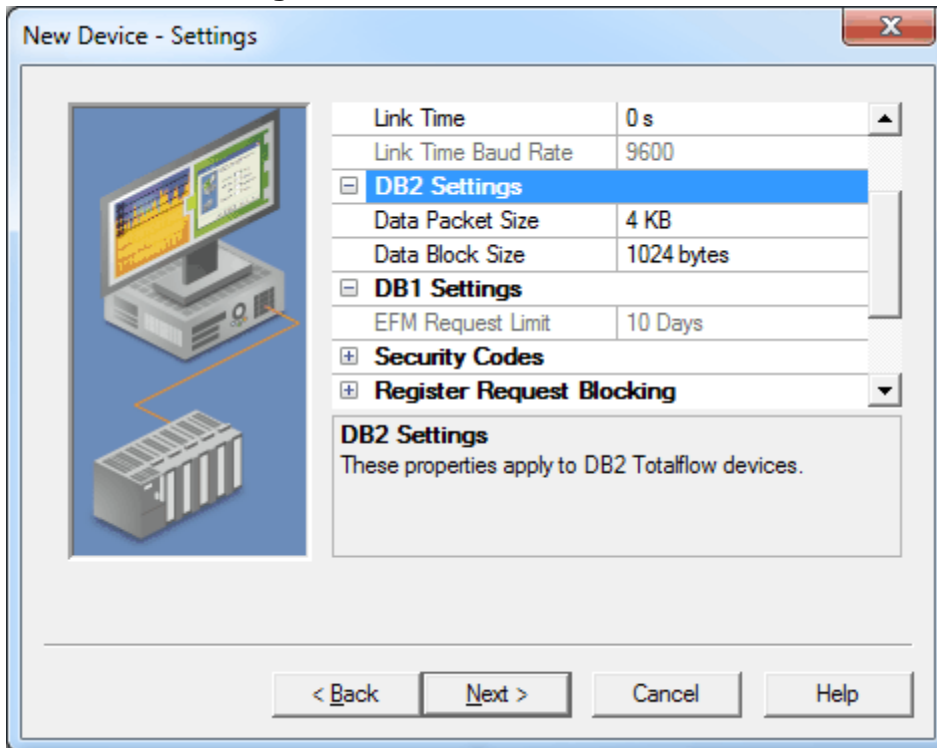


Descriptions of the parameters are as follows:

- **Device ID:** This parameter specifies an ASCII string that is used to identify an ABB Totalflow device. A maximum length of 10 characters is supported. The default setting is "TOTALFLOW".
- **Protocol Version:** This parameter specifies the protocol version to be used for communications. Options include DB1 (not packet-based) and DB2 (packet-based). The default setting is DB2.  
**Note:** DB1 cannot communicate with liquid meters and can only communicate with one meter at a time. DB2 allows up to 20 meters to be created.
- **Link Time:** This parameter specifies the interval that the device opens the receive channel. It is used to determine how many supervisory frames to send so that a remote device's receive channel is on and the request is detected. Options include 0 seconds, 1 second, 2 seconds, and 4 seconds. The default setting is 0 seconds.
- **Link Time Baud Rate:** This parameter is required for the Link Time calculation when communicating with a serial device while Ethernet Encapsulation is enabled. When a COM port is being used, this

parameter is fixed to the COM port's baud rate. The valid range is 300 to 256,000 bits per second. The default setting is 9600 bits per second.

### DB2 and DB1 Settings

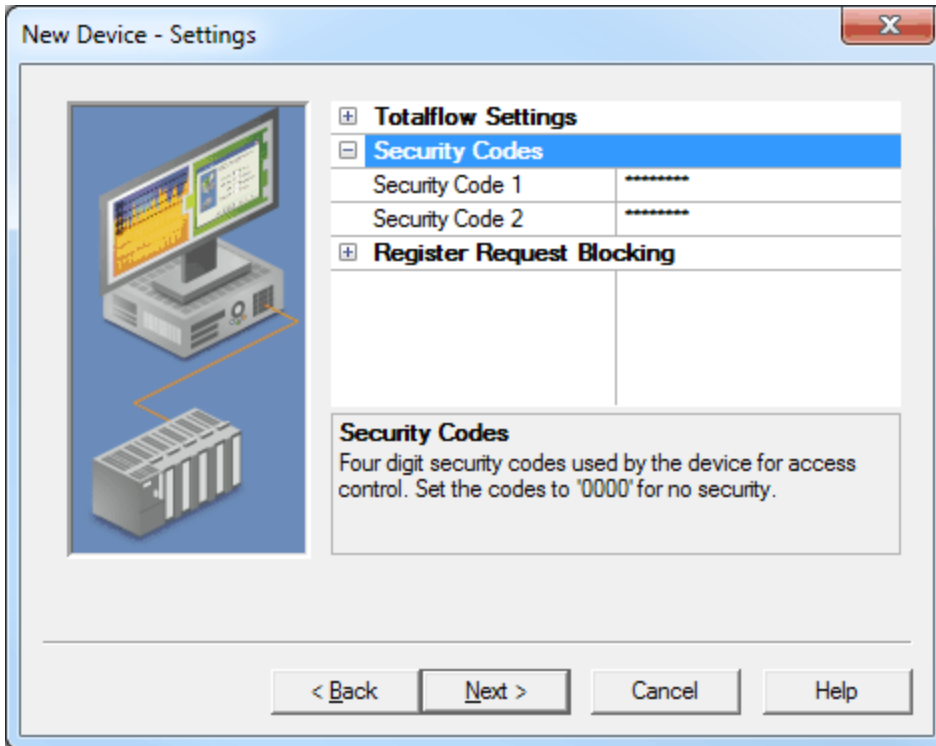


Descriptions of the parameters are as follows:

- **Data Packet Size:** This parameter specifies the size of each remote packet transmitted using packet control. Options include 1 KB, 2 KB, 4 KB, 8 KB, 16 KB, and 32 KB. The default setting is 4 KB. This setting affects EFM and trend uploads only and can be adjusted to optimize communications during uploads.
- **Data Block Size:** This parameter specifies the size of each remote data block transmitted that can be CRC checked. Options include 128 bytes, 256 bytes, 512 bytes, and 1024 bytes. The default setting is 1024 bytes. This setting affects EFM and trend uploads only and can be adjusted to optimize communications during uploads.
- **EFM Request Limit:** This parameter limits the amount of EFM data that is requested to the specified number of days. Options include 3 days, 10 days, and 35 days. The default setting is 10 days.

### Security Codes

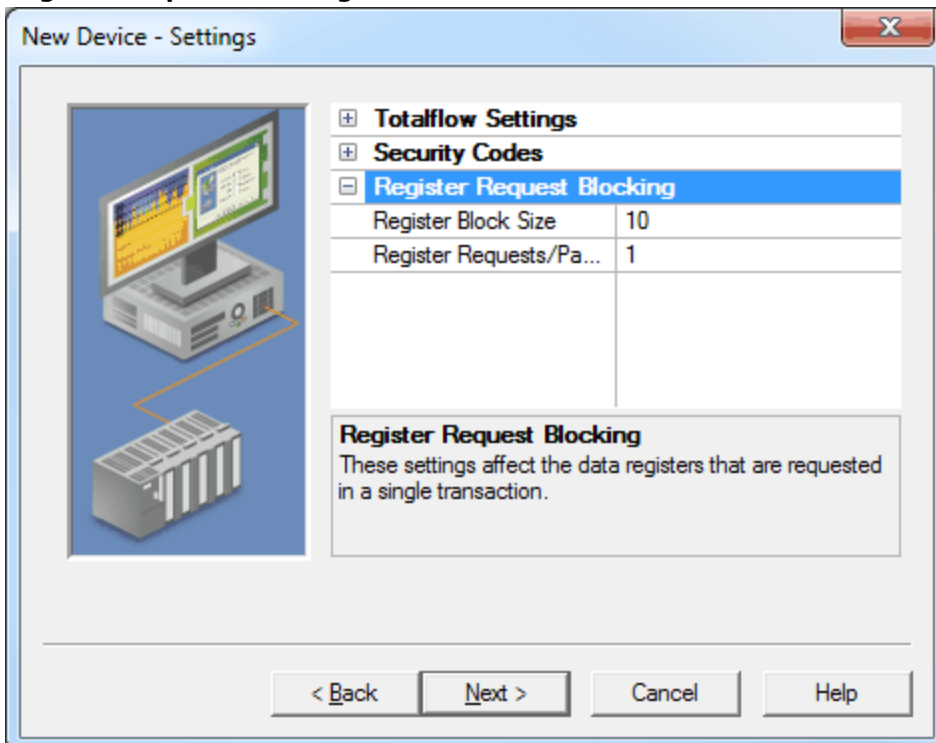
Users can configure security codes in the device to provide communications with additional security. The codes are stored securely.



Descriptions of the parameters are as follows:

- **Security Code 1:** This parameter specifies a four digit code that is used for read-only access. It has a maximum length of 4 characters. The default setting is 0000.
- **Security Code 2:** This parameter specifies a four digit code that is used for read/write access. It has a maximum length of 4 characters. The default setting is 0000.

### Register Request Blocking



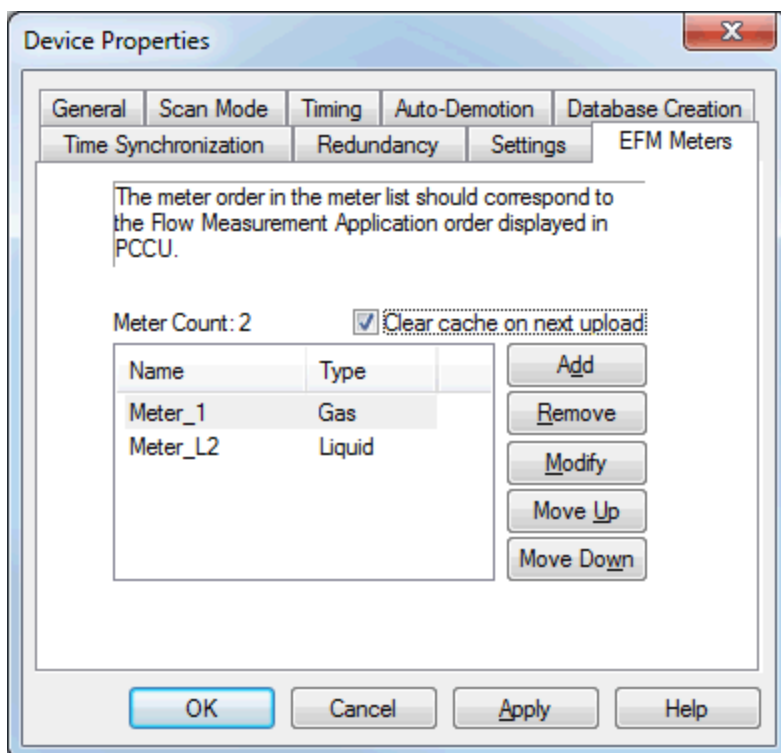
Descriptions of the parameters are as follows:

- **Register Block Size:** This parameter specifies the maximum number of registers that can be read in a single request. The valid range is 1 to 100. The default setting is 10.
- **Register Requests/Packet:** This parameter specifies the maximum number of register blocks that can be read in a single packet. The valid range is 1 to 16. The default setting is 1.

**Note:** These parameters are only available when the selected protocol version is DB2.

## EFM Meters

This dialog is used to add, remove, and modify EFM Meters. The meter order determines the association with the ABB Totalflow Flow Measurement application. This list must coincide with the order of the Flow Applications in ABB's Portable Configuration and Calibration Unit (PCCU) tool. To access these parameters after device creation, right-click on the device and then select **Device Properties | EFM Meters**.



When creating meters, note that DB1 protocol supports only a single meter and DB2 protocol supports a maximum of 20 meters. This does not necessarily reflect the number of meters that can be created in the Totalflow device.

Descriptions of the parameters are as follows:

- **Meter Count:** This parameter displays the number of meters that have been added to the device.
- **EFM Meter List:** This list view displays the meters and their type (gas or liquid) currently supported by the device.
- **Add:** When clicked, this button invokes the Meter Configuration dialog for adding a new meter to the device.
- **Remove:** When clicked, this button removes the selected meter from the EFM Meter List.
- **Modify:** When clicked, this button invokes the Meter Configuration dialog for modifying the selected meter in the EFM Meter List.
- **Move Up:** When clicked, this button moves the selected meter up in the EFM Meter List.
- **Move Down:** When clicked, this button moves the selected meter down in the EFM Meter List.

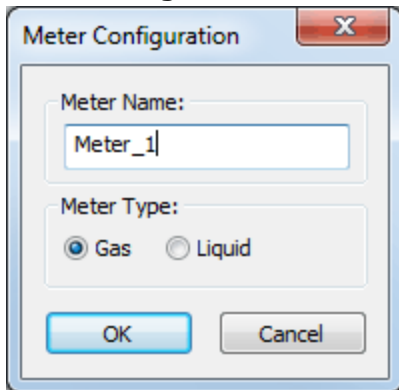
**Note:** Changing a meter's order or removing it from the EFM Meter List causes the EFM cache to be cleared on the next upload. Although this ensures data integrity, users can avoid it by unchecking the **Clear cache on next upload** option before applying any changes. For more information, refer to "Clear Cache on Next Upload" below.

### Clear Cache on Next Upload



Users have the option to clear any cached EFM data from the device during the next upload. This feature also removes pointer files, which are used to track EFM uploads to prevent uploading the same records twice. All EFM data is re-uploaded. Once the cache is cleared, this parameter is automatically disabled. To enable this option, open **Device Properties | EFM Meters**. Then, click **Clear cache on next upload**. The default setting is unchecked.

### Meter Configuration



The screenshot shows a 'Meter Configuration' dialog box. It has a title bar with the text 'Meter Configuration' and a close button (X). The dialog contains two input fields: 'Meter Name:' with the text 'Meter\_1' and 'Meter Type:' with radio buttons for 'Gas' (selected) and 'Liquid'. At the bottom are 'OK' and 'Cancel' buttons.

Description of the parameter is as follows:

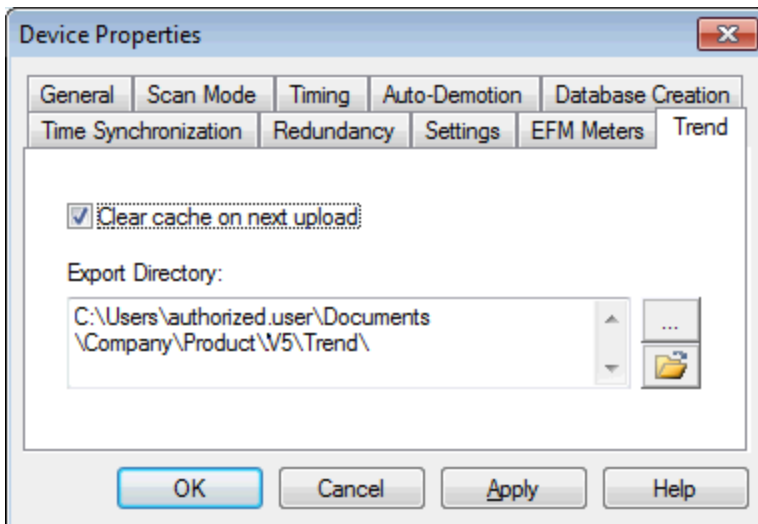
- **Meter Name:** This parameter specifies the meter name. Each meter must be assigned a unique name to request EFM Configuration, Daily History, Hourly History, and Event records from an ABB Totalflow device.
- **Meter Type:** This parameter specifies the meter type (gas or liquid) and must match the application type configured for the meter in PCCU.

**Note:** Liquid meters are not compatible with the DB1 protocol.

## Trend File Configuration

The ABB Totalflow Driver has the ability to upload and export trend data from Totalflow devices via the DB2 protocol. Trend files are exported in comma-separated-value (CSV) file format with each cell contained within double quotations (") and using commas for the field delimiter. This dialog is used to configure settings related to how the server uploads trend files. The CSV file name is the same as defined in the device with the device ID prepended to it. The format is <Device ID>-<Trend File Name>.csv. The device ID should be unique to prevent files from being overwritten.

The following dialog is used to configure settings related to how the server uploads trend files.



To access the dialog box:

1. Selecting the device with trend files in the tree of the Project View.
2. Right-click to open the pop-up menu.
3. Choose **Properties...**
4. Select the **Trend** tab.

Descriptions of the parameters are as follows:

**Export Directory:** Specifies the path to which all trend files are written when uploaded from this device. *See note below for information about the export directory.*

**Browse** Allows the user to navigate to a directory for exported trend files.

**Open Folder** Browses the folder specified in Export Directory.

**Clear Cache on next upload** provides the option to delete the trend cache on the subsequent upload to cause all trend records to be retrieved from the device during the next upload. *See below for more information on trend cache.* Once the cache has been cleared, this option is disabled (unchecked). The default setting is disabled.

**Tip:** The default export directory is Documents\<company>\<product>\V<version\_number>\Trend\ where the bracketed variables are replaced with the local installation information. The directory is created on first upload. The directory can be changed by browsing to a different folder or typing in a new path. If the folder does not exist, it will be created on the next upload.

5. Once editing is complete, click **OK** to apply and close (or **Cancel** to close and discard changes).

### Trend Cache

The trend cache is used to track the timestamp of the latest upload for each trend file. It allows the ability to upload only new records from the device.

**Note:** the trend cache is stored on disk in the form of pointer files so that the cache data is maintained across a server restart. Clearing the trend cache also removes these files from disk.

### Reasons for Export Failure

Trend file uploads and exports can fail for various reasons. Failures during upload from a device are generally caused by device configuration or communication problems noted in the event log (see [Error Descriptions](#) for more information in these cases). Failures during the trend file export can occur in the following ways:

- The file to be written exists and is locked.
- The file to be written is located in a directory for which the server does not have write permissions.
- The server does not have write permissions for the file to be written or the file is read-only.

In these cases, the server cannot export the uploaded data. On the next upload, the same data is uploaded and another export attempted. If problem still exists, the export fails again. All export failures are noted in the event log with a message indicating why the export failed. See [Error Descriptions](#) for more information on these failures.

### Notes on Export Directory

The following restrictions are placed on the export path.

- The path must be either a valid UNC (\\server\share\) or drive letter (C:\path\) path.
- The root of the path must not be a mapped network drive (see below for more information).
- The path must not contain any characters not supported in Windows folder names.
- The path must be less than 256 characters long.  
**Note:** The operating system may place more strict restrictions on the path length than 256. On many Windows operating systems, the maximum path length also includes the file name and extension. Therefore, character space must be reserved when creating the path for all trend file names and extensions to be written to disk.

The driver supports exporting trend CSV files to multiple types of media, including drives with removable storage (flash drives or external hard drives) and network drives. Due to the nature of the server runtime, the driver does not have access to mapped network drives. To export data to a network drive, the path must be specified with the UNC format (\\server\share\) and not a mapped drive letter. Any circumstances in which the media to be exported to is not accessible (cases such as network communication difficulties or removable media removed) data is re-uploaded and another attempt at exporting the data made on the next upload.

## Automatic Tag Database Generation

---

The ABB Totalflow Driver can be configured to automatically generate tags for characteristic items and trend files in the device, as well as tags that indicate the status of a trend upload.

The Characteristic Items, created when the [protocol version](#) is configured for DB1, are described by three functional groups: Last Volume Period (LVP) Tags, Current Measurement Tags, and Device Setup Tags. All tags are read only. These tags are created in the Device Setup and Measurement groups.

Tags related to trend file uploads are created in a tag group at the device level named "Trend". The trend file upload tags created include one tag per file defined in the device, as well *TF\_UploadAll* and *TF\_LastModified* tags. Trend status are also generated: *TS\_UploadingTrend*, *TS\_UploadingTrendFile*, and *TS\_Error* (see [Address Descriptions](#) for more information on these tags).

To generate tags from the device:

1. In the Configuration, select the device for which to generate tags.
2. Right-click and select **Properties...** to open the Device Properties dialog box.
3. Select the Database Creation tab.
4. Click the **Auto Create** button to initiate tag database creation.
5. Click the **Close** button to exit the dialog box.
6. In the Event Log, verify messages confirming successful generation.

**Note:** An automatic tag generation creates the characteristic items regardless of the server's ability to connect to the device. If the server is not connected to the device or communications are lost during the tag generation, the trend file tags are not created and a warning message is posted in the event log.

*For more information about automatic tag generation settings, see the server help file.*

**See Also:**

[Address Descriptions](#)

[Characteristic Address Descriptions](#)

[Trend Address Descriptions](#)

[DB1 and DB2 Settings](#)

**Data Types Description**

---

<b>Data Type</b>	<b>Description</b>
Bool	Single bit
Char	Signed 8-bit value
Byte	Unsigned 8-bit value
Short	Signed 16-bit value
Word	Unsigned 16-bit value
Long	Signed 32-bit value
DWord	Unsigned 32-bit value
Float	32-bit floating point value
String	Null-terminated character array

**Address Descriptions**

ABB Totalflow devices organize data by application, array, and register. Although the meaning and type of data is specific to the application type, a fully-qualified ABB Totalflow address requires all three of these parameters. Its syntax is as follows: `<application #>.<array #>.<register #>`.

Each application is usually designed to perform a specific task, calculation, or function in the ABB Totalflow device. The application numbers are not standardized, meaning that the application number for a Flow Measurement application AGA3-1 can differ from device-to-device. The application numbers are based on the application category and the order in which they are added to the device. The ABB Totalflow Driver follows this addressing convention, supporting the following syntax:

```
<Application>.<Array>.<Register>/<bit>
<Application>.<Array>.<Register>[row][column]
```

**Note:** The valid range for the register identifier fields are 0-255, 0-255, and 0-65535 respectively. The bit index is validated against the specified OPC type. In the event that the register type does not support the specified bit index, the read fails and an appropriate error message is logged once per tag. The product of rows multiplied by columns cannot exceed the maximum register block size of 100. A 1-length row/column is allowed.

For more information, select a link from the list below.

- [Characteristic Address Descriptions](#)
- [DB2 Protocol Address Descriptions](#)
- [Trend Address Descriptions](#)
- [Statistics Items](#)

**Characteristic Address Descriptions**

**Data Blocking**

G1 and G2 devices do not use registers. Data can be requested from a number of categories: the processing is the same for all of them. Data is requested in two blocks based on the method of data request. Characteristics items cannot be requested individually. The driver supports the Current Measurement and/or Device Setup tag categories. Due to blocking limitations, it is recommended that data be retrieved by register for DB2 devices. Only the Characteristic Address Tags are available for DB1 devices (because they do not use a register-based architecture). For more information, refer to [DB2 Protocol Address Descriptions](#).

The Characteristic Items are described by three functional groups: Last Volume Period (LVP) Tags, Current Measurement Tags, and Device Setup Tags. All tags are read only.

**Last Volume Period (LVP) Tags**

ABB Totalflow devices calculate volume in a user-configurable interval between 1 and 60 minutes.

Address	Data Type	Description
LVPALARMS	Byte	Alarms
LVPACF	Float	Accumulated Flow
LVPDP	Float	DP
LVPAP	Float	AP
LVPTF	Float	Temperature
LVPVOLUME	Float	Volume
LVPQV	Float	Qv
LVPIRANGE	Float	Turbine Range
LVPICOUNTS	Float	Turbine Counts
LVPEXT	Float	Extension
LVPCP	Float	Cp
LVPFB	Float	Fb
YSTACFLOW	Float	Yesterday's Low ACF
LVPFR	Float	F(r)
YSTACFHIGH	Float	Yesterday's High ACF
LVPY	Float	Y
YSTACFTOTAL	Float	Yesterday's Total
LVPFPB	Float	Pressure Base Factor

Address	Data Type	Description
LVPFTB	Float	Temperature Base Factor
LVPFTF	Float	Flowing Temperature Factor
LVPFG	Float	Gravity Factor
LVPFPV	Float	Supercompressibility Factor
LVPFA	Float	Auxiliary
LVPFW	Float	Water Vapor Factor
LVPFAUX	Float	Auxiliary Factor
LVPBTU	Float	BTU/SPC
LVPGRAVITY	Float	Specific Gravity
LVPN2	Float	Mole %
LVPCO2	Float	Mole %
LVPH2S	Float	Mole %
LVPH2O	Float	Mole %
LVPHELIUM	Float	Mole %
LVPETHANE	Float	Mole %
LVPETHANE	Float	Mole %
LVPPROPANE	Float	Mole %
LVPNBUTANE	Float	Mole %
LVPIBUTANE	Float	Mole %
LVPNPENTANE	Float	Mole %
LVPIPENTANE	Float	Mole %
LVPNHEXANE	Float	Mole %
LVPNHEPTANE	Float	Mole %
LVPNOCTANE	Float	Mole %
LVPNONANE	Float	Mole %
LVPNDECANE	Float	Mole %
LVPOXYGEN	Float	Mole %
LVPCO	Float	Mole %
LVPHYDROGEN	Float	Mole %
LVPARGON	Float	Mole %
LVPFIP	Float	FIP
LVPS	Float	$F(pv)^2$
LVPEV	Float	Ev
LVPFPM	Float	FPM
LVPCORORIF	Float	Corrected orifice diameter for the effect of temperature
LVPTM	Float	Ftm
LVPCORPIPE	Float	Corrected pipe diameter (ID) for the effect of temperature
LVPPULSECOUNT	Float	Pulse Count
LVPRHOB	Float	Rhob
FLOWPERIOD	Float	Flow Period (seconds)
LVPQM	Float	Qm
LVPCD	Float	Cd

### Current Measurement Tags

Address	Data Type	Description
CURACCVOL	Float	Current Accumulated Volume
CURACCVOLTURBINE	Float	Current Accumulated Volume (Turbine)
CURAP	Float	Current Static Pressure
CURDP	Float	Current Differential Pressure
CURMACF	Float	Current Actual MCF
CURTF	Float	Current Temperature
CURFLOW	Float	Current Flow Rate (SCF)
CURVOLTAGE	Float	Current Voltage
YSTACCVOL	Float	Yesterday's Accumulated Volume
YSTMCFTOTAL	Float	Yesterday's Total MCF

**Device Setup Tags**

Address	Data Type	Description
CONTRACTHOUR	Byte	Contract Hour (0-23)
STREAMID	Uint32	4 Bytes. In ASCII, the Stream ID is <i>iiii-aa-ss</i> .  Bytes 0 & 1: AIU number ( <i>iiii</i> ); 0000 to 9999, with the bytes reversed from Microsoft Byte 2: Analyzer number ( <i>aa</i> ); 1-32 Byte 3: Stream number in the analyzer ( <i>ss</i> ); 1-32
FIXEDAXONERR	Bool	If STREAMATTACHED is 1 and the received analysis is not correct or, if no analysis is received, then:  False = Use last known good analysis True = Use fixed analysis
STREAMATTACHED	Bool	False = Always use fixed analysis True = Attached to AIU
RTDINSTALLED	Bool	False = No True = Yes
USEMEASTEMP	Bool	False = Fixed Temp True = Live
CHECKSEC	Bool	Use Security
MONELORIF	Bool	False = Stainless True = Monel
USEFB	Bool	Use FB in Calculations
USES	Bool	Use $S(F(pv)^2)$ ; Turbine Only
USEFR	Bool	Use Flow Rate
USEFTC	Bool	Use Temperature Correction Factor
USEY	Bool	Use Expansion Factor
USEFPC	Bool	Use Pressure Correction Factor
USEFTB	Bool	Use Temperature Base Factor
USEFPB	Bool	Use Pressure Base Factor
USEFTF	Bool	Use Flowing Temperature Factor
USEFG	Bool	Use Gravity Factor
USEFPV	Bool	Use Super Compressibility Factor
USEFA	Bool	Auxiliary
LCCONTACT	Bool	Charger Low
DPLOWCONTACT	Bool	DP Low Contact
DPHICONTACT	Bool	DP High Contact
APLOWCONTACT	Bool	AP Low Contact
APHICONTACT	Bool	AP High Contact
REMSENSECONTAC	Bool	Use Remote Sense
AUTORESET	Bool	Automatic Reset
VOLSPCONTACT	Bool	Use Volume Set Point
USEFW	Bool	Use Water Vapor Factor
USEFAUX	Bool	Use Auxiliary Factory
FIXEDAP	Float	Fixed AP
USEMEASAP	Float	Use Measured AP
FB	Float	Basic Orifice Factor
ORIFICE	Float	Orifice Diameter (inches)
PIPE	Float	Pipe Diameter (inches)
GRAVITY	Float	Specific Gravity
DPLOWLIM	Float	DP Low Limit
DPHILIM	Float	DP High Limit
APLOWLIM	Float	AP Low Limit
APHILIM	Float	AP High Limit
CO2	Float	Mole % CO2



Address	Data Type	Description
N2	Float	Mole % N2
APLOWCAL	Float	AP Calibration: Low Point
APMIDCAL	Float	AP Calibration: Mid Point
APHICAL	Float	AP Calibration: High Point
DPLowCAL	Float	DP Calibration: Low Point
DPMIDCAL	Float	DP Calibration: Mid Point
DPHICAL	Float	DP Calibration: High Point
ZEROCUTOFF	Float	DP Zero Cutoff
TBASE	Float	Temperature Base
PBASE	Float	Pressure Base
FIXEDTF	Float	Fixed Temperature
TEMPBIAS	Float	Temperature Bias
VISC	Float	Centipose
RSPH	Float	Ratio of Specific Heats
FT	Float	Mass Flow Calibration Temperature Coefficient
FP	Float	Pressure
BTU	Float	BTU/SCF
ACFLOWLIM	Float	Turbine: Low Limit
ACFHILIM	Float	Turbine: High Limit
FAUX	Float	Auxiliary Factory
K	Float	Turbine
CALCUNITS	Byte	Bit Definitions  0 = US Units 1 = IP Units 2 = MT Units 3 = SI Units
ACFLOWLIM	Float	Accumulated Flow Limit
ZMETHOD	Float	Bit Definition  0 = NX19, fixed Ft, Fp 1 = NX19 GCN or GCNM 2 = NX19 GCN 3 = NX19 GCNM 7 = AGA-8 HGCN 8 = AGA-8 GCN Limited Number of High Pressure FCUs 10= AGA-8 GCNM 11= AGA-8 Gross 1992 (Gross 2) 12= AGA-8 Detail 1992
TAPLOCATION	Bool	0 = Downstream 1 = Upstream
TAPTYPE	Bool	0 = Flange 1 = Pipe
VOLCALCMETHOD	Byte	Value Definition  0= No Volume Calculation 1 = AGA-3 1985 2= AGA-3 1992, API 14.3
FIXEDCD	Bool	0 = Fixed 1 = Calculated CD
PIPEREFTEMP	Float	Reference Temperature (F)
ORIFREFTEMP	Float	Orifice Temperature (F)
ZBA	Float	Z of Air at Base Conditions
VOLCALCPER	Uint16	Volume Calculation Period in Seconds
VOLLOGPER	Uint32	Volume Log Period in Seconds
H2S	Float	Mole %
H2O	Float	Mole %

Address	Data Type	Description
HELIUM	Float	Mole %
METHANE	Float	Mole %
ETHANE	Float	Mole %
PROPANE	Float	Mole %
NBUTANE	Float	Mole %
IBUTANE	Float	Mole %
NPENTANE	Float	Mole %
IPENTANE	Float	Mole %
NHEXANE	Float	Mole %
NHEPTANE	Float	Mole %
NOCTANE	Float	Mole %
NNONANE	Float	Mole %
NDECANE	Float	Mole %
OXYGEN	Float	Mole %
CO	Float	Mole %
ORIFEXP	Float	Orifice Plate Coefficient of Expansion
PIPEEXP	Float	Pipe Coefficient of Expansion
BAROP	Float	Barometric Pressure
USECALCCD	Bool	Use Calculated Cd
HYDROGEN	Float	Mole %
ARGON	Float	Mole %
APMIDLOWCAL	Float	AP Calibration: Mid-Low
APMIDHICAL	Float	AP Calibration: Mid-High
DPMIDLOWCAL	Float	DP Calibration: Mid-Low
DPMIDHICAL	Float	DP Calibration: Mid-High
SQRTAVG	Float	Square Root Mean
VOLSP	Float	Volume Set Point
TFLOWLIM	Float	Temperature Low Limit
TFHILIM	Float	Temperature High Limit
FLOWLOWLIM	Float	Flow Low Limit
FLOWHILIM	Float	Flow High Limit
PASSWORDMODE	Bool	0 = No Password Mode 1 = Mode Enabled
FIRSTAX	Bool	At Least One Good Analyzer
PRIMARYELEM	Byte	0= Orifice 1= Turbine
REPORTUNITS	Byte	Report Unite  Bit Definition: 0 = US 1 = IP 2 = MT 3 = SI

**DB2 Protocol Address Descriptions**

DB2 Protocol addressing information applies to G2, G3, and G4 devices.

**Data Blocking**

Due to blocking limitations, it is recommended that data be retrieved by register for DB2 device. A DB2 register request consists of a base address (in the form of <application>.<array>.<register>) and a register count. The registers retrieved in a request are always in the same Totalflow array and have the same data type. For example, if 10 registers are requested beginning at "9.3.0," the response contains data for "9.3.0" through "9.3.9" with a uniform data type. The driver blocks data in the same manner.

Register data is retrieved from the TCI toolkit as an array of bytes. The data type is inferred from the size of the array and the number of elements that were requested. The element size within a data block is fixed based on the largest atomic type (because the data type cannot be accurately deduced until data is retrieved from the device). Encoding the data length provides greater flexibility for the client tag data type. This means that users do not

need to know the Totalflow data type when defining client tags in the driver. Data type mismatches result in truncation. There is no special handling for floating point values. Examples are as follows:

1. A Word tag is defined with an address of "9.3.0". The ABB Totalflow device has this register defined as an int32. The driver reads the tag and sets its quality to Bad because a Word cannot properly represent all int32 values. An error message is posted to the Event Log.
2. A DWord tag is defined with an address of "9.5.0". The ABB Totalflow device has this register defined as a Float. The driver reads the tag, receives the value "1.523," and stores it in the block memory. When the tag is updated, its value is 1069740458 (the binary equivalent of 1.523).

Strings are not blocked by the driver due to their fixed size. A runtime error (including an Event Log message) is posted when a string register is assigned a blockable type. Strings are not converted to any other data type. Furthermore, multiple register requests can be included in a single DB2 transaction. This means that the driver can service multiple block tag reads with a single device request/response. The number of register requests per packet is user-configurable. The driver pools tags based on the following criteria:

- All tags will be read or write.
- All tags must be for the same device.
- The number of tags is limited to the number specified by the user.

### Bit-within-Word Booleans

Bit-within-Word Booleans provide a mechanism for interpreting register data as a bit field. For reads, this involves using a mask to determine the Boolean value for a desired bit (which is specified in the tag's address). In order to write bit values, the driver must perform a Read/Modify/Write operation to ensure that only a single bit is being set.

### Trend Address Descriptions

Trend tags apply to all ABB Totalflow devices that support the DB2 protocol and trend logging.

#### Trend File Tags

Syntax Example: Channel.Device.TF\_UploadAll

Tag	Data Type	Description	Access
TF_<trend_file_name>	Boolean	This tag is used to upload a single file specified by <trend_file_name> where trend_file_name must match the name of a trend file defined in the device. Writing any value to this tag initiates an upload. This tag always reads 0.	Write Only
TF_LastModified	Boolean	This tag is used to upload the trend file that was last modified in the device. This tag reserves the name LastModified; therefore trend files should not be created on the device with this name. Writing any value to this tag initiates an upload. This tag always reads 0.	Write Only
TF_UploadAll	Boolean	This tag is used to upload all trend files on the device. This tag reserves the name UploadAll; therefore trend files should not be created on the device with this name. Writing any value to this tag initiates an upload. This tag always reads 0.	Write Only
TF_UploadLastHours	Word	This tag is defines how many hours of records, from the current time, will be uploaded. It only specifies the number of hours and does not initiate the upload. The value is used only for the next upload initiated by a single trend file tag, last modified tag, or upload all tag and returns to zero when the upload is complete. This tag reserves the name UploadLastHours; therefore trend files should not be created on the device with this name. <b>Notes:</b> <ul style="list-style-type: none"> <li>• The clear cache setting has no effect on uploads that use this tag. Upon completion of an upload that uses this tag, the trend cache is updated with the timestamp of the newest record uploaded.</li> <li>• If an existing CSV file isn't consumed before using this tag, the file could be missing data or have duplicate data.</li> <li>• This tag uses the time zone/DST settings on the Time Synchronization tab in the Device Properties. These</li> </ul>	Read/Write

Tag	Data Type	Description	Access
		settings must match the device's time zone/DST settings for this tag to work properly.	

**Trend File Tag Validation Requirements:**

The following address requirements must be met when creating trend file tags:

- Length (including the "TF\_") must be at least four (4) characters.
- Length (including the "TF\_") must be less than 66 characters. This is due to the 63 character restriction enforced by the TCI toolkit.
- The address must be valid ASCII characters.
- The file name should not contain white space other than the standard space character.
- Prohibited characters are double quotes ("), period (.), and 'at' symbol (@).

**Notes:**

- The file's Scan Status in the device must be set to "On" to upload the trend file from the device.
- These addresses are not case sensitive.

**Trend Status Tags**

Syntax Example: Channel.Device.TS\_Uploading

Tag	Data Type	Description	Access
TS_Uploading	Boolean	This tag indicates whether the driver is currently retrieving a trend file from the device and exporting trend data to disk.	Read Only
TS_UploadingFile	String	This tag indicates the name of the trend file that the driver is currently uploading from the device.	Read Only
TS_ErrorOnLastUpload	Boolean	This tag indicates that an error has occurred during the most recent trend file upload. If an error occurs during a single file or upload all request, this tag is set to TRUE. The tag is reset upon completion of the next successful upload, including uploads that result in no new data.	Read Only
TS_LastUploadStart	Date	This tag indicates the time that the most recent trend file upload began. All times are reported in local time. If no upload has started, the tag reports 01/01/1601 00:00:00.000.	Read Only
TS_LastUploadEnd	Date	This tag indicates the time that the most recent trend file upload finished. All times are reported in local time. If no upload has completed, the tag reports 01/01/1601 00:00:00.000.	Read Only
TS_LastUploadDurationSec	Double	This tag indicates the amount of time (in seconds) the last trend file upload took to complete. This tag is updated on upload completion whether the upload succeeds or fails.	Read Only
TS_ErrorCount	DWord	This tag increments every time a trend file upload fails due to a communication or export error. Writing any value to this tag resets the counter to zero.	Read/Write

**Blocking Trend File Uploads**

When possible, trend files are uploaded in multiple small blocks of records instead of one large block to optimize data throughput over noisy communication links. The Totalflow firmware provides the information required to block uploads for up to 15 trend files via registers (see Notes below). If there are more than 15 trend files configured on a device, some of the trend file uploads will not be uploaded in small blocks. One exception is that all trend file uploads initiated with a non-zero TF\_UploadLastHours value are blocked regardless of the number of trend files configured on the device.

**Notes:**

- The driver records the timestamp of the newest record in the file at the beginning of a trend file upload. If records are being logged at a high rate and the file has wrapped, it is possible that the driver may

upload a few records less than the full trend file size when uploading from a cleared cache state. This is expected behavior as the driver will upload those records on the next upload.

- The following register addresses are used when determining if a trend file upload can be blocked. If a trend file's name is in one of these 15 registers when an upload for the file is initiated, the file's upload is blocked. If the trend file's "Scan Status" is set to "Off" in the device, the trend file's name will not appear in these registers.

<trend app num>.241.0 through <trend app num>.241.14

where <trend app num> is the application slot number where the trend system application is instantiated.

## Statistics Items

Statistical items use data collected through additional diagnostics information, which is not collected by default. To use statistical items, Communication Diagnostics must be enabled. To enable Communication Diagnostics, right-click on the channel in the Project View and click **Properties | Enable Diagnostics**. Alternatively, double-click on the channel and select **Enable Diagnostics**.

### Channel-Level Statistics Items

The syntax for channel-level statistics items is *<channel>.\_Statistics*.

**Note:** Statistics at the channel level are the sum of those same items at the device level.

Item	Data Type	Access	Description
_CommFailures	DWord	Read/Write	The total number of times communication has failed (or has run out of retries).
_ErrorResponses	DWord	Read/Write	The total number of valid error responses received.
_ExpectedResponses	DWord	Read/Write	The total number of expected responses received.
_LastResponseTime	String	Read Only	The time at which the last valid response was received.
_LateData	DWord	Read/Write	The total number of times that a driver tag's data update occurred later than expected (based on the specified scan rate).
_MsgResent	DWord	Read/Write	The total number of messages sent as a retry.
_MsgSent	DWord	Read/Write	The total number of messages sent initially.
_MsgTotal	DWord	Read Only	The total number of messages sent (both _MsgSent + _MsgResent).
_PercentReturn	Float	Read Only	The proportion of expected responses (Received) to initial sends (Sent) as a percentage.
_PercentValid	Float	Read Only	The proportion of total valid responses received (_TotalResponses) to total requests sent (_MsgTotal) as a percentage.
_Reset	Bool	Read/Write	Resets all diagnostic counters. Writing to the _Reset Tag causes all diagnostic counters to be reset at this level.
_RespBadChecksum	DWord	Read/Write	The total number of responses with checksum errors.
_RespTimeouts	DWord	Read/Write	The total number of messages that failed to receive any kind of response.
_RespTruncated	DWord	Read/Write	The total number of messages that received only a partial response.
_TotalResponses	DWord	Read Only	The total number of valid responses

Item	Data Type	Access	Description
			received ( _ErrorResponses + _ExpectedResponses).

Statistical items are not updated in simulation mode (see device general properties).

**Device-Level Statistics Items**

The syntax for device-level statistics items is <channel>.<device>.\_Statistics.

Item	Data Type	Access	Description
_CommFailures	DWord	Read/Write	The total number of times communication has failed (or has run out of retries).
_ErrorResponses	DWord	Read/Write	The total number of valid error responses received.
_ExpectedResponses	DWord	Read/Write	The total number of expected responses received.
_LastResponseTime	String	Read Only	The time at which the last valid response was received.
_LateData	DWord	Read/Write	The total number of times that a driver tag's data update occurred later than expected (based on the specified scan rate).
_MsgResent	DWord	Read/Write	The total number of messages sent as a retry.
_MsgSent	DWord	Read/Write	The total number of messages sent initially.
_MsgTotal	DWord	Read Only	The total number of messages sent (both _MsgSent + _MsgResent).
_PercentReturn	Float	Read Only	The proportion of expected responses (Received) to initial sends (Sent) as a percentage.
_PercentValid	Float	Read Only	The proportion of total valid responses received ( _TotalResponses) to total requests sent ( _MsgTotal) as a percentage.
_Reset	Bool	Read/Write	Resets all diagnostic counters. Writing to the _Reset Tag causes all diagnostic counters to be reset at this level.
_RespBadChecksum	DWord	Read/Write	The total number of responses with checksum errors.
_RespTimeouts	DWord	Read/Write	The total number of messages that failed to receive any kind of response.
_RespTruncated	DWord	Read/Write	The total number of messages that received only a partial response.
_TotalResponses	DWord	Read Only	The total number of valid responses received ( _ErrorResponses + _ExpectedResponses).

Statistical items are not updated in simulation mode (see device general properties).

## Error Descriptions

---

The following messages may be generated. Click on the link for a description of the message.

### Communication Error Messages

[A communication error occurred while <reading/writing> address <tag address> on device <device name>: <verbose communication error>.](#)

[A communication error occurred while reading address <tag address> on device <device name>: Device did not respond.](#)

[A communication error occurred while reading address <tag address> on device <device name>: General remote NACK error.](#)

[A communication error occurred while reading address <tag address> on device <device name>: Illegal register read or write.](#)

[A communication error occurred while reading address <tag address> on device <device name>: Invalid data structure.](#)

[A communication error occurred while reading address <tag address> on device <device name>: Received data had a CRC error.](#)

[A communication error occurred while reading address <tag address> on device <device name>: Transaction ID did not match.](#)

### Device-Specific Error Messages

[Bit index out of range for register address <tag address> on device <device name>.](#)

[Could not write data for trend file <trend file> on device <device> to disk. <reason>.](#)

[<Device name> - Failed to read EFM pointer file. <Extended error>.](#)

[<Device name> - Failed to write EFM pointer file. <Extended error>.](#)

[<Device name> - Failed to read trend pointer file. <Extended error>.](#)

[<Device name> - Failed to write trend pointer file. <Extended error>.](#)

[Device <device name> is not responding.](#)

[EFM type mismatch for meter <meter name> on device <device name>. The meter is configured for <meter type> in the server but it is a <meter type> meter in the device.](#)

[Extra data revision not supported. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>.](#)

[Extra data size is 0 bytes. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>.](#)

[Failed to send communication request for address <tag address> on device <device name>](#)

[Invalid address for register block <App>.<Array>.<Register X> - <App>.<Array>.<Register Y> on device <device name>.](#)

[Invalid data type <data type> for address <tag address> for device <device name>](#)

[Register blocking error, verify that data type <data type> is valid for address <tag address> on device <device name>.](#)

[Serialization of EFM data to temporary file <file name> failed. Reason: <file I/O error>.](#)

[Trend file configuration changed for trend file <trend file> on device <device>. Old data moved to <file name>. New data written to <file name>.](#)

[Trend file <trend file name> does not exist on device <device name>.](#)

[Trend upload expected new records for trend file <trend file> on device <device name>, but no records were retrieved.](#)

[Unable to load TCI Toolkit extradata.ini file, EFM feature may not work properly.](#)

[Unable to write to <address> on device <device name>.](#)

---

**A communication error occurred while <reading/writing> address <tag address> on device <device name>: <verbose communication error>.**

---

**Error Type:**

Warning

**Possible Cause:**

There was an error in communications.

**Solution:**

1. For more information, refer to the verbose error message.
2. Verify the ABB Totalflow device's configuration.

---

**A communication error occurred while reading address <tag address> on device <device name>: Device did not respond.**

---

**Error Type:**

Warning

**Possible Cause:**

The protocol version is not set correctly.

**Solution:**Update the protocol version to match that of the device. *For more information, refer to [Settings](#).*

---

**A communication error occurred while reading address <tag address> on device <device name>: General remote NACK error.**

---

**Error Type:**

Warning

**Possible Cause:**

There is a response mismatch.

**Solution:**

Increase the Request Timeout to allow time for the response to complete.

---

**A communication error occurred while reading address <tag address> on device <device name>: Illegal register read or write.**

---

**Error Type:**

Warning

**Possible Cause:**

The device does not recognize the register address.

**Solution:**

Verify that the register address is valid.

---

**A communication error occurred while reading address <tag address> on device <device name>: Invalid data structure.**

---

**Error Type:**

Warning

**Possible Cause:**

1. The data that was requested is not supported by the destination device.
2. The meter for which data was requested does not exist in the destination device.

**Solution:**



1. Verify that the device supports the DB2 protocol.
2. Verify that, for each meter in the server, a meter also exists on the destination device. Ensure that the order of meters in the server matches the order of meters in the device.

---

**A communication error occurred while reading address <tag address> on device <device name>: Received data had a CRC error.**

---

**Error Type:**

Warning

**Possible Cause:**

1. The response's integrity could not be validated.
2. There is a response mismatch resulting in an invalid CRC check.

**Solution:**

1. Investigate for sources of noise or errors in the transmission media.
2. Increase the Request Timeout to prevent unnecessary request retries.

---

**A communication error occurred while reading address <tag address> on device <device name>: Transaction ID did not match.**

---

**Error Type:**

Warning

**Possible Cause:**

1. The specified tag data type and Totalflow register data type do not match. This prevents the tag from being blocked with other Totalflow registers.
2. There is a response mismatch.

**Solution:**

1. Ensure that the tag data type and the register data type match.
2. Increase the Request Timeout to allow time for the response to complete.

---

**Bit index out of range for register address <tag address> on device <device name>.**

---

**Error Type:**

Warning

**Possible Cause:**

The bit index specified in the tag address is incompatible with the register data type.

**Solution:**

Verify the register data type. Then, adjust the bit index accordingly.

---

**Could not write data for trend file <trend file> on device <device> to disk: <reason>.**

---

**Error Type:**

Warning

**Possible Cause:**

The source of the problem is indicated by the <reason> message. Common <reason> messages:

- Unable to create path <path>.
  - Access denied.
  - Permission denied.
- Unable to open file <file name>.
  - Access denied.
  - Permission denied.
  - No such file or directory.
- Trend file configuration changed, unable to move old data to another file.

**Solution:**

- The path specified for export does not exist and the server was unable to create it. There are two common reasons this can occur:
  - The server does not have permission to the drive or path to create a new folder. Ensure that the server runtime service has permissions for the drive and path to which it must write.
  - The specified path is too long for the current operating system. Choose a shorter path.
- The server was not able to open for read or create the output file.
  - Verify that the file is not locked by another program and is not specified as read only.
  - Verify that the server has permission to read/write to the directory and file.
  - "No such file or directory" appears if the specified path, combined with the output file name, exceeds the system maximum path length. To resolve this problem, use a shorter path to export data.
- When the configuration of the trend file (number of variables, variable names, and units) changes and the trend file already exists on disk, the server attempts to move the old trend data to another file so that the new configuration is not written into the old file with the old configuration. This message is posted if the server is unable to complete this move. Ensure that the file to be moved does not already exist and is not locked, that the disk is not full, and that the server has write permissions in the export directory.

---

**<Device name> - Failed to read EFM pointer file. <Extended error>.**

---

**Error Type:**

Warning

**Extended Error:**

When supplied by the operating system, this describes the file error that occurred.

**Possible Cause:**

1. A permission error was encountered when the EFM pointer cache was read.
2. The EFM pointer cache file is corrupt.

**Solution:**

The ABB Totalflow Driver automatically generates a new EFM pointer file; however, the server re-polls (uploading all EFM data) during the next EFM poll for meters in the device.

**Note:**

*For more information, refer to the extended error.*

---

**<Device name> - Failed to read trend pointer file. <Extended error>.**

---

**Error Type:**

Warning

**Extended Error:**

When supplied by the operating system, this describes the file error that occurred.

**Possible Cause:**

1. A permission error was encountered when the trend pointer cache was read.
2. The trend pointer cache file is corrupt.

**Solution:**

The ABB Totalflow Driver automatically generates a new trend pointer file; however, the server re-polls (uploading all trend data) during the next poll for trend files in the device.

**Note:**

For more information, refer to the extended error.

**<Device name> - Failed to write EFM pointer file. <Extended error>.**

---

**Error Type:**

Warning

**Extended Error:**

When supplied by the operating system, this describes the file error that occurred.

**Possible Cause:**

1. The disk is full.
2. A permission error was encountered when the EFM pointer cache was written.

**Solution:**

The server attempts to update the EFM pointer file periodically, in addition to when the server is shutdown. If the pointer file cannot be written, the server re-polls (uploading all EFM data) during the next EFM poll for meters in the device.

**Note:**

For more information, refer to the extended error.

**<Device name> - Failed to write trend pointer file. <Extended error>.**

---

**Error Type:**

Warning

**Extended Error:**

When supplied by the operating system, this describes the file error that occurred.

**Possible Cause:**

1. The disk is full.
2. A permission error was encountered when the trend pointer cache was written.

**Solution:**

The server attempts to update the trend pointer file periodically, in addition to when the server is shutdown. If the pointer file cannot be written on server shutdown, the next upload after restart begins uploading records with the timestamp from the last successful pointer file write.

**Note:**

For more information, refer to the extended error.

**Device <device name> is not responding.**

---

**Error Type:**

Serious

**Possible Cause:**

1. The serial connection between the device and the host PC is broken.
2. The communications parameters for the serial connection are incorrect.

3. The named device may have been assigned an incorrect device ID.
4. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.

**Solution:**

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified communications parameters match those of the device.
3. Verify that the device ID given to the named device matches that of the actual device.
4. Increase the Request Timeout setting so that the entire response can be handled.

---

**EFM meter <meter name> on device <device name> is invalid.**

---

**Error Type:**

Warning

**Possible Cause:**

The specified meter name or type was modified or the meter was removed while being referenced by an active EFM Exporter meter.

**Solution:**

Update the EFM Exporter's meters to reference the latest EFM meters for the affected device.

---

**EFM type mismatch for meter <meter name> on device <device name>. The meter is configured for <meter type> in the server but it is a <meter type> meter in the device.**

---

**Error Type:**

Warning

**Possible Cause:**

The meter is configured as a liquid meter in the server, but it is a gas meter in the device or vice versa.

**Solution:**

Change the meter type configuration setting in the server to match the meter type in the device.

---

**Extra data revision not supported. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>.**

---

**Error Type:**

Warning

**Possible Cause:**

The device is running a firmware version that is not supported by the driver.

**Solution:**

Verify that the device is running a firmware version that is supported by the driver. Contact Technical Support.

---

**Extra data size is 0 bytes. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size>.**

---

**Error Type:**

Warning

**Possible Cause:**

The device is running a firmware version that is not supported by the driver.

**Solution:**

Verify that the device is running a firmware version that is supported by the driver. Contact Technical Support.

---

**Failed to send communication request for address <tag address> on device <device name>.**


---

**Error Type:**

Warning

**Possible Cause:**

Failed to send the request for the specified tag and device.

**Solution:**

Verify the Totalflow protocol and communication port settings.

---

**Invalid address for register block <App>.<Array>.<Register X> - <App>.<Array>.<Register Y> on device <device name>.**


---

**Error Type:**

Warning

**Possible Cause:**

1. An attempt has been made to reference a nonexistent location in the specified device.
2. There is a response mismatch.

**Solution:**

1. Verify the tags that are assigned to the addresses in the specified range on the device. Then, eliminate ones that reference invalid locations.
2. Increase the Request Timeout to allow time for the response to complete.

---

**Invalid data type <data type> for address <tag address> for device <device name>.**


---

**Error Type:**

Warning

**Possible Cause:**

A tag data type has been selected that cannot properly represent the Totalflow register data type.

**Solution:**

Select a tag data type that matches the Totalflow register data type.

---

**Register blocking error, verify that data type <data type> is valid for address <tag address> on device <device name>.**


---

**Error Type:**

Warning

**Possible Cause:**

The specified tag data type and Totalflow register data type do not match. This prevents the tag from being blocked with other Totalflow registers.

**Solution:**

Ensure that the tag data type and the register data type match.

---

**Serialization of EFM data to temporary file <file name> failed. Reason: <file I/O error>.**


---

**Error Type:**

Warning

**Possible Cause:**

1. The driver was unable to create the specified file directory.
2. The driver was unable to access the specified file.

**Solution:**

1. Verify that the disk has sufficient disk space.
2. Verify user permissions for the specified file directory.

**Trend file configuration changed for trend file <trend file> on device <device>. Old data moved to <file name>. New data written to <file name>.****Error Type:**

Warning

**Possible Cause:**

The configuration of the trend file on the Totalflow device changed after the data was written to disk for that file. Changes detected include the number of records being logged to the file, the order in which they appear, as well as the description or units associated with each variable. Rather than continuing to write data to a file with a header that does not match, the old file is renamed so that the new data can be written to a new file with that name.

**Solution:**

Before modifying trend file configuration in the device, consume and move or delete the old file.

**Trend file <trend file name> does not exist on device <device name>.****Error Type:**

Warning

**Possible Cause:**

The requested trend file is not configured on the device.

**Solution:**

Verify that the requested trend file is configured on the device or correct the file or path.

**Trend upload expected new records for trend file <trend file> on device <device name>, but no records were retrieved.****Error Type:**

Warning

**Possible Cause:**

Records are being logged at a rate that overwrites the entire file before the driver can upload the file.

**Solution:**

Adjust the logging rate and trend file upload interval so that records in the file are not overwritten between uploads.

**Unable to load TCI Toolkit extradata.ini file, EFM feature may not work properly.****Error Type:**

Warning

**Possible Cause:**

The extradata.ini file is corrupt or not located in the same directory as the TCI Toolkit dll (tcidll.dll).

**Solution:**

Verify that a valid extradata.ini file exists in the same directory as the TCI Toolkit dll (tcidll.dll). If the extradata.ini file cannot be found or installed in the directory, contact Technical Support.

**Unable to write to <address> on device <device name>.**

---

**Error Type:**

Serious

**Possible Cause:**

1. The serial connection between the device and the Host PC is broken.
2. The communications parameters for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Device ID.

**Solution:**

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified communications parameters match those of the device.
3. Verify that the Device ID given to the named device matches that of the actual device.

# Index

## A

- A communication error occurred while <reading/writing> address <tag address> on device <device name>: <verbose communication error>. 24
- A communication error occurred while reading address <tag address> on device <device name>. Device did not respond. 24
- A communication error occurred while reading address <tag address> on device <device name>: General remote NACK error. 24
- A communication error occurred while reading address <tag address> on device <device name>: Illegal register read or write. 24
- A communication error occurred while reading address <tag address> on device <device name>: Invalid data structure. 24
- A communication error occurred while reading address <tag address> on device <device name>: Received data had a CRC error. 25
- A communication error occurred while reading address <tag address> on device <device name>: Transaction ID did not match. 25
- Address Descriptions 14
- Automatic Tag Database Generation 12

## B

- Bit index out of range for register address <tag address> on device <device name>. 25

## C

- Characteristic Address Descriptions 14
- Could not write data for trend file <trend file> on device <device> to disk for <reason>. 25

## D

- Data Types Description 13
- DB2 Protocol Address Descriptions 18
- Device <device name> is not responding. 27
- Device Setup 5

## E

- EFM meter <meter name> on device <device name> is invalid. 28
- EFM Meters 8
- EFM type mismatch for meter <meter name> on device <device name>. The meter is configured for <meter type> in the server but it is a <meter type> meter in the device. 28
- Error Descriptions 23
- Extra data revision not supported. The EFM output file for <meter name> on device <device name> will be



missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size> 28

Extra data size is 0 bytes. The EFM output file for <meter name> on device <device name> will be missing some data. Extra data revision = <extra data revision>, extra data size = <extra data size> 28

## F

Failed to read EFM pointer file for <device name>. <Extended error>. 26

Failed to read trend pointer file for <device name>. <Extended error>. 26

Failed to send communication request for address <tag address> on device <device name>. 29

Failed to write EFM pointer file for <device name>. <Extended error>. 27

Failed to write trend pointer file for <device name>. <Extended error>. 27

## H

Help Contents 4

## I

Invalid address for register block <App>.<Array>.<Register X> - <App>.<Array>.<Register Y> on device <device name>. 29

Invalid data type <data type> for address <tag address> for device <device name>. 29

## O

Overview 4

## R

Register blocking error, verify that data type <data type> is valid for address <tag address> on device <device name>. 29

## S

Serialization of EFM data to temporary file <file name> failed. Reason: <file I/O error>. 29

Settings 5

Statistics Items 21

## T

Trend Address Descriptions 19

Trend file <trend file name> does not exist on device <device name>. 30

Trend file configuration changed for trend file <trend file> on device <device>. Old data moved to <file

name>. New data written to <file name>. 30

Trend File Upload Configuration 10

Trend upload expected new records for trend file <trend file> on device <device name>, but no records were retrieved. 30

## **U**

Unable to load TCI Toolkit extradata.ini file, EFM feature may not work properly. 30

Unable to write to <address> on device <device name>. 31