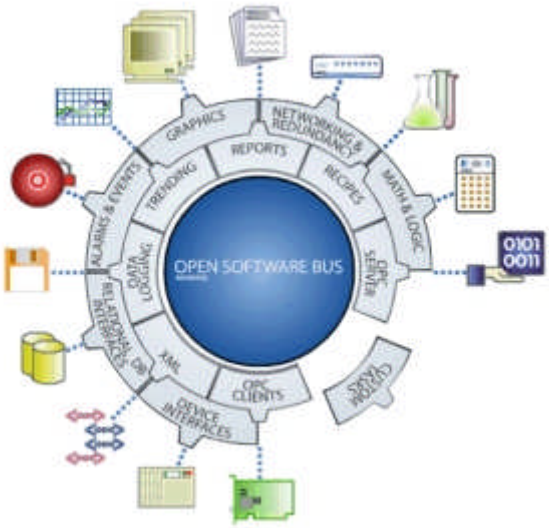
SIEMENS



FactoryLink Valmet Damaticx CIS Protocol Driver

for

FactoryLink 7.5.x

FINTsys

Version 2.0
Printed: Tuesday, October 07, 2008

Table of Contents

1. INTRODUCTION 5

1.1. SCOPE OF THIS DOCUMENT 5

2. CONTENTS OF SHIPMENT 7

3. INSTALLATION 9

3.1. INSTALLATION OF THE FACTORYLINK SOFTWARE 9
3.2. INSTALLATION OF THE SERIAL PORT 11
3.3. INSTALLATION OF THE PROTECTION 12
3.3.1. The DeltaLink option file 12
3.3.2. Demo installation 12

4. PRINCIPLE 13

4.1. THE TRANSLATOR RAP DRIVER PRINCIPLE 13
4.2. VALMET DAMATIC COMMUNICATION 14
4.2.1. Datasets 14
4.2.2. Sets 14
4.2.3. Valmet Damatic CIS Communication 14
4.3. FACTORYLINK DOMAIN SELECTION 15
4.4. VALMET DAMATIC CLASSIC TAG ADDRESS FORMAT 16

5. CONFIGURATION TABLES 17

5.1. MAILBOX DEFINITION 19
5.2. DECODER DEFINITION 20
5.3. DEVICE DEFINITION 21
5.4. DATASET DEFINITION 24
5.5. SET DEFINITION 26
5.6. ADDITIONAL REMARKS 29
5.6.1. Size restrictions 29
5.6.2. Errors in set definitions 29
5.6.3. Translator tag definitions 29

APPENDIX A. THE DELTA.OPT FILE 33

APPENDIX B. VALMET PROTOCOL DRIVER ERROR CODES 35

APPENDIX C. MESSAGES 37

1. Introduction

Thank you for buying this driver! We hope you will enjoy using this product.

1.1. Scope of this document

This manual is written for a technician who is familiar with both the FactoryLink® IV software and the Valmet Damatic computer and it's CIS interface. This document can be used both as a training manual as well as a reference manual.

> ***Note:*** *Please check the contents of the shipment with the list as described in the next chapter.*

The first section of this manual deals with the installation of hardware and software in your FactoryLink workstation. This part is split into a platform independent and a platform specific part (currently Windows'95 and Windows-NT are supported). Please read carefully through both parts to make sure both hardware and software are installed correctly.

The second part explains the operation principles of the communication with the Valmet Damatic computer and the Decoder. Here all terms and definitions are explained to the reader. It explains for example a term like "Dataset".

The third part explains the exact tables associated with this driver. This part is useful only to FactoryLink programmers and can be used as a reference. This part is also an example of how to use this driver with the Decoder.

The last part are the appendices which contain summarised data.

RLD Automation
Van Sonsbeeckstraat 11
5344 JB Oss
The Netherlands

tel. +31(0) 412 655 990
fax +31(0) 412 655 991
e-mail : support@rldautomation.eu

This page is left blank intentionally.

2. Contents of shipment

Please check the package you received with the checklist below. Should there be an item missing contact DeltaLink bv to correct the problem. There is a limit of 90 days after shipment to report problems!

This package includes the following:

❶ 1 diskette labelled "DeltaLink Valmet Damatic CIS Protocol Driver"
Files:
\AC\VALMET.AC
\BIN\VALMET.EXE
\CTGEN\VALMET.CTG
\KEY\DE\VAL_BUS.KEY
\KEY\DE\VAL_TYPE.KEY
\KEY\EN\VAL_BUS.KEY
\KEY\EN\VAL_TYPE.KEY
\KEY\FR\VAL_BUS.KEY
\KEY\FR\VAL_TYPE.KEY
\MSG\DE\VALMET.HLP
\MSG\DE\VALMET.TXT
\MSG\DE\VALM_AC.TXT
\MSG\EN\VALMET.HLP
\MSG\EN\VALMET.TXT
\MSG\EN\VALM_AC.TXT
\MSG\FR\VALMET.HLP
\MSG\FR\VALMET.TXT
\MSG\FR\VALM_AC.TXT
\OPT\DELTA.OPT
\INSTALL.BAT
\INST_SEQ.EXE
\FLBUILD.ID
\FLXMEDIA
\UPDATE.EXE
\INSTALL.$$$

❷ 1 DeltaLink authorization sequence for a VALMET option.

❸ This manual (Which seems to be present).

You should also have:

☑ A IMX based translator e.g. I/O Translator.

☑ A Valmet Damatic computer with a corresponding CIS interface.

☑ A computer with at least one free serial communication port to connect the FactoryLink station and the Valmet Damatic computer.

This page is left blank intentionally.

3. Installation

3.1. Installation of the FactoryLink software.

To install the FactoryLink task and its related tables please follow the following steps.

Before installing

Before installing the Valmet Damatic CIS driver on the system FactoryLink must have been installed error free. It is very important that all the environment settings are made for the FactoryLink system such as the *FLINK*, *FLOPT* etc.

First:

Copy the files from the "DeltaLink Valmet Damatic CIS driver" diskette to the appropriate directories. This will be automatically done by running the install utility placed on the installation diskette.

Follow the next procedure:

```
a:\↵
install↵
```

Second:

After you installed the software you need to activate the tables in the FactoryLink Configuration Manager (FLCM). The installation automatically appends the *valmet.ac* entry into the *{FLINK}/AC/titles* file[1]. The place of this entry is also the place where the option appears in the FLCM Main Menu. Therefore check the validity of the entry and move it to the place where you want to appear it in the Configuration manager. The entry must match the *valmet.ac* entry in the following table:

```
file: {FLINK}/ac/titles:

...
windhdr.ac
decoder.ac
valmet.ac
spool.ac
...
```

Third:

To make sure all the Configuration Tables (CT's) are generated after a change, the install utility automatically adds the *Valmet* entry at the end of the *{FLINK}/ctgen/ctlist* file. The place of this entry is not important. Check if this entry has the same format as in the next table:

```
file: {FLINK}\ctgen\ctlist:

...
rp: rptovr rpthdr
valmet: val_pd_t val_pd_x val_pd_y val_pd_t val_pd_z
decoder: decoderm decoderp decoderd decodert
timer: itimer etimer
...
```

[1] {FLINK} is the working directory for the FactoryLink programs.

Fourth:

To enable the help fields in the Valmet Damatic CIS tables type the following command:

```
{FLINK}\BIN\mkhelp↵
```

Fifth:

The Valmet Damatic CIS task must be entered in the FactoryLink Configuration Manager (FLCM) System Configuration table. An entry of an existing task which will not be used at run-time can be overwritten or a new entry can be made with as a minimum the following data:

Task Name	*Description*	....	*Executable File*
VALMET	***DeltaLink MCI Valmet Protocol driver***		***bin/valmet.exe***

The *Task Name* and *Executable File* name are fixed and should not be altered by the user.

This completes the installation of the FactoryLink (software) parts of the Valmet Damatic CIS driver.

3.2. Installation of the serial port

It is necessary to have a serial port on your system which can be used by the protocol driver for communicating with the Valmet Damatic CIS system(s). Refer to the manual(s) of your system for installing/setting up a serial port. Make sure everything is installed correctly, and additional device drivers, when needed, are loaded. Connect the serial port to the Valmet Damatic CIS system(s), connector layouts are described in the manuals for your system and the Valmet system. Refer to *Appendix B* for information about setting the serial communication parameters used by the Valmet Damatic driver.

3.3. Installation of the protection

The Valmet Damatic CIS protocol driver is protected via the DeltaLink option file. This file contains authorization sequence codes for DeltaLink modules. The protection is linked to the serial number of the FactoryLink package.

3.3.1. The DeltaLink option file

The installation media of the Valmet Damatic CIS driver contains an option file, named ‘delta.opt’, in the 'opt' directory. This file contains the unique authorization sequence which enables the Valmet Damatic CIS driver to run. The install utility automatically copies the authorization sequence into the {FLOPT}/delta.opt file. It is also possible to enter the authorization sequence manually into the {FLOPT}/delta.opt file. For more information on the delta.opt file refer to *Appendix A*.

Note that the task will only run on the FactoryLink system with the same serial number. The *delta.opt* file on the installation diskette contains, for reference, the serial number of FactoryLink.

3.3.2. Demo installation

It is possible to install the Valmet Damatic driver without an authorization code. This will be done from a normal installation media. In this case the task will start up but only runs in so called 'demo' mode. This means that the driver runs only for a limited period of time (five hours). After this period has expired the task will shutdown and can not be restarted before the complete FactoryLink system has been restarted.

After installation of this demo version an authorization code can be ordered and installed which enables the task to run without the time and restart limitation. The authorization code must be entered manually in the {FLOPT}/delta.opt file. For more information on entering the authorization code in the delta.opt file refer to *Appendix A*.

The limitations of a demo version of the Valmet Damatic driver are:

- five hours of consecutive run-time
- not restartable (FactoryLink must be restarted)

4. Principle

4.1. The translator RAP Driver principle

RAPD stands for Rapid Application Protocol Driver. The RAPD principle was adopted so that protocol drivers can be easily and rapidly configured for a FactoryLink application. RAPD is based on the Intertask Mail Exchange Standard or IMX, which defines a way for a protocol driver task to communicate with an I/O Translator task (e.g. I/O Translator, high speed logger). The RAPD system consists of a protocol driver which communicates with external devices (RTUS, PLC's, etc.) and a translator which controls data storage (going to and coming from a protocol driver) in the FactoryLink real-time database. All data collected by the protocol driver is referenced as contiguous blocks or ranges within the device. This enables communications between the driver and a device to be very efficient. All data is referenced between the driver and the translator in terms of datasets. Datasets, described in the next section, define memory regions or locations of data within a device.
The protocol driver and the IOX communicate with one another via FactoryLink mailbox tags, according to the IMX standard. Every task (translator and protocol drivers) has its own mailbox, so for full communication between a translator and a protocol driver a mailbox database element for every task has to be defined. The IMX standard is especially designed for the following situation. To use one translator and several protocol drivers. For example the translator together with the Valmet Damatic protocol driver and the Modbus Plus protocol driver. Aside from storage duties, the translator provides data conversions (i.e. analog, IEEE conversions, etc.) for I/O data to/from a protocol driver.

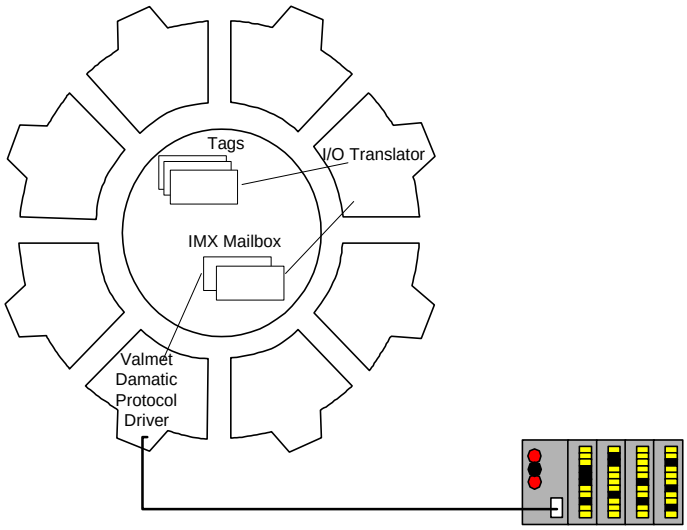


Figure 4.1.1 The RAPD principle.

4.2. Valmet Damatic Communication

4.2.1. Datasets

A dataset is a (non-contiguous) area of data in the I/O-device. The complete area can be read with one command (trigger). The decoder takes care of mapping the I/O-device-data to/ FactoryLink TAGs (including conversions).

Note: The Valmet driver does not support writing to datasets.

4.2.2. Sets

A set is the definition of the contents of a dataset. In other words, a set defines which I/O-points in the Valmet Damatic computer should be accessed when reading a data set. Each entry in a set defines one single I/O-point. The size of a basic-element of a set depends on the type of data that remains in the Valmet Damatic computer.

4.2.3. Valmet Damatic CIS Communication

Data exchange between a FactoryLink workstation and the Valmet Damatic computer is performed by using the CIS interface of the Valmet Damatic computer. The CIS interface allows a serial point-to-point communication. Through the CIS interface the protocol driver will read (not write) data from the Valmet. Communication is always initiated by the protocol driver.

4.3. FactoryLink domain selection

The standard domain for the ValmetDamatic protocol driver is the SHARED domain. The protocol driver communicates with a dedicated piece of hardware, there fore only one task should be able to access the hardware. If only one program accesses the hardware, the task should be located in the shared domain and therefore started by the shared runtime-manager.

Important: The protocol driver and the translator must be in the same domain (either SHARED or USER).

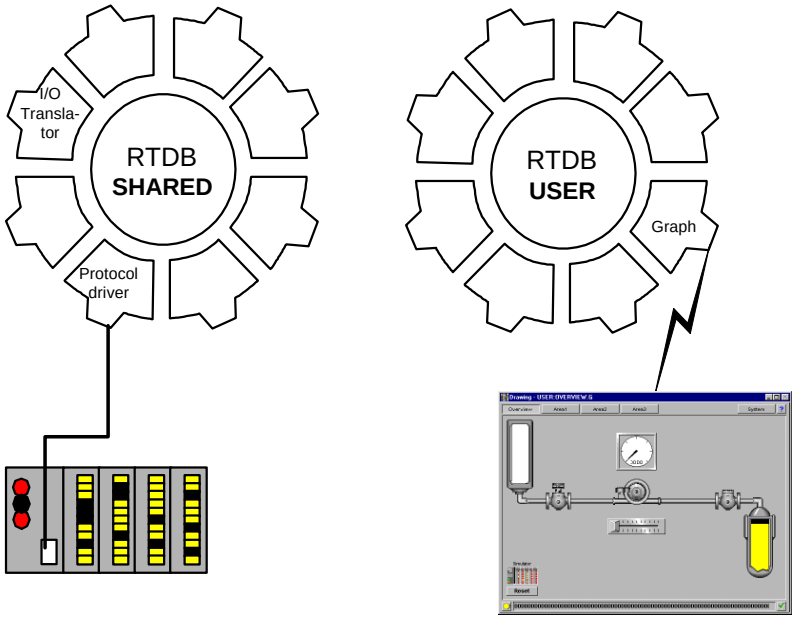


Figure 4.3.1 Standard domain selection.

4.4. Valmet Damatic Classic Tag Address Format

Internally, the Valmet GCU addresses its points by a combination of Bus Number (1-15), Process Station's Number (1-14), Block (0-7), Signal Group (0-3) and Signal Number (0-3).

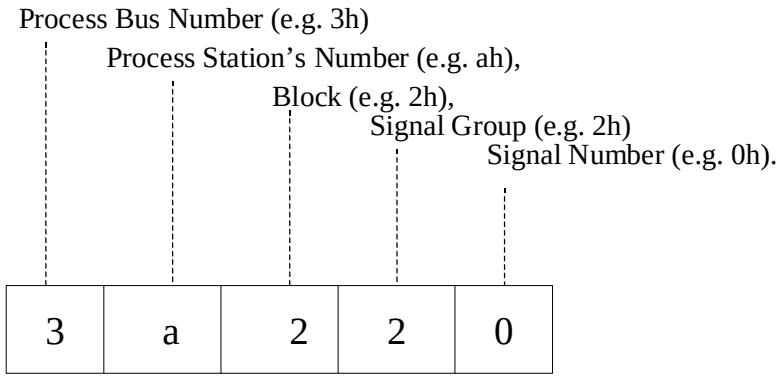


Figure 4.4.1 Classic tag address format.

5. Configuration tables

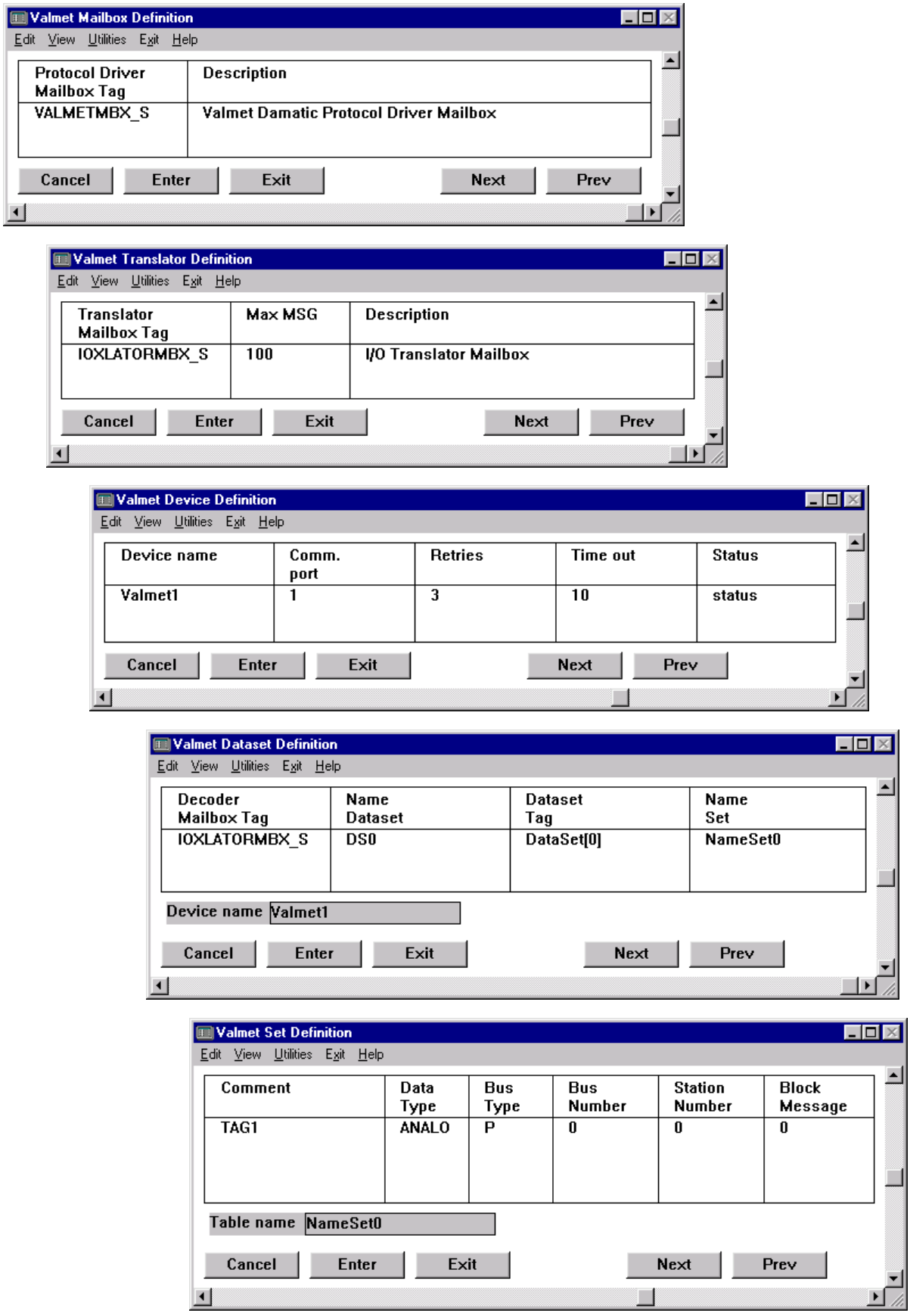


Figure 5.0.1 Valmet confgurarion panels.

In the Configuration Manager Main Menu, select **Valmet Protocol Driver**. Five tables appear, with the titles of all panels visible for direct access. To access a specific panel position the cursor on a visible area and press the left mouse-button, or use the Next/Prev buttons.

Note: For general information about entering data in FactoryLink configuration tables, refer to the FactoryLink Fundamentals Manual.

5.1. Mailbox definition

From the display of all the panels, select the *Valmet mailbox definition* panel.

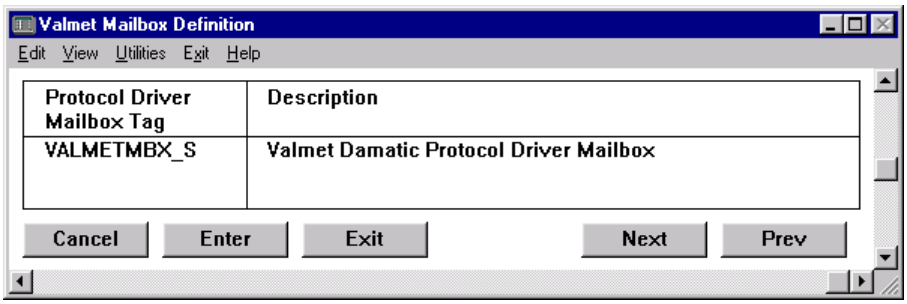


Figure 5.1.1 DeltaLink Valmet definition panel.

The DeltaLink Valmet mailbox definition panel allows the user to initialise one Mailbox Tag for the Valmet protocol driver. Only one Mailbox is needed for full performance of MCI. Specify the following information.

- **Protocol Driver Mailbox Tag**
  Tag name of the Valmet protocol mailbox element that the developer defines, to be referenced by a decoder task (**N.B.** MCI must be supported by the decoder task). The decoder task uses this mailbox to send requests to the Valmet protocol driver.

  valid entry: Required.
  description: Standard FactoryLink tag name of type MAILBOX.

- **Description**
  Description of the Valmet protocol mailbox element defined by the developer.

  valid entry: Output only.

5.2. Decoder definition

From the display of all the panels, select the *DeltaLink Valmet Decoder definition* panel.

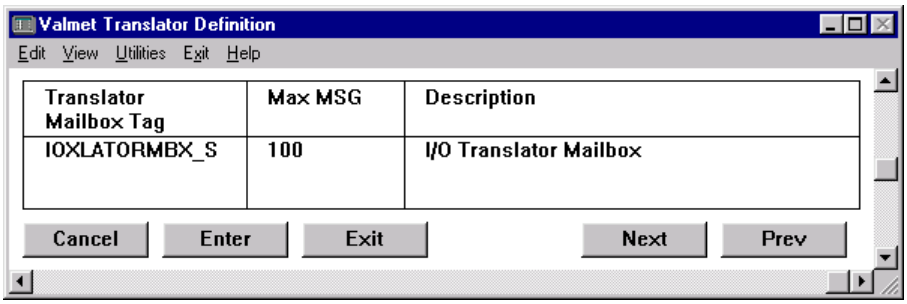


Figure 5.2.1 DeltaLink DAMATIC CIS Decoder definition.

The DeltaLink Valmet decoder definition panel allows the user to specify one or more decoder(s). Every Mailbox Tag specifies a different decoder. Only one Mailbox Tag, for every decoder, is needed for full performance of MCI. Specify the following information.

- **Decoder Mailbox Tag**
  Tag name of a decoder mailbox element, to be referenced by the Valmet protocol driver task (**N.B.** MCI must be supported by the decoder task). The decoder task uses this mailbox to receive data from the Valmet protocol driver.

  valid entry: Required.
  description: Standard FactoryLink tag name of type MAILBOX.

- **Max MSG**
  The maximum number of requests for the protocol driver, which can be queued in the Mailbox Tag. Recommended is a value of 100 messages. The number of messages is limited by the size of an integer value (9999), and practically by the amount of available memory. The memory needed for a request depends on the size of the configured datasets.

  valid entry: Required/default: 100.
  valid entries: 1 .. 9999.

- **Description**
  Description of the decoder mailbox element defined by the developer.

  valid entry: Output only.

5.3. Device definition

From the display of all the panels, select the *DeltaLink Valmet Device definition* panel.

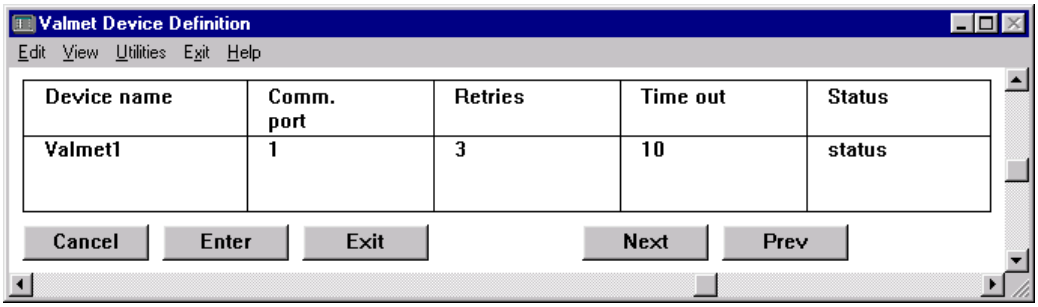


Figure 5.3.1 DeltaLink Valmet Device definition.

The DeltaLink Valmet Device definition panel allows the user to specify logical devices for the communication. Each logical device uses exclusively a serial communication port. Laos, it is possible to specify tags that enable the programmer to enable/disable reading and writing to the physical device.
Specify the following information.

- **Device Name**
  Logical name assigned by the user to represent a particular communication device.

  valid entry: Required.
  description: Alphanumeric string of up to 16 characters.

- **Com port**
  The number of the serial port through which the device must be accessed.

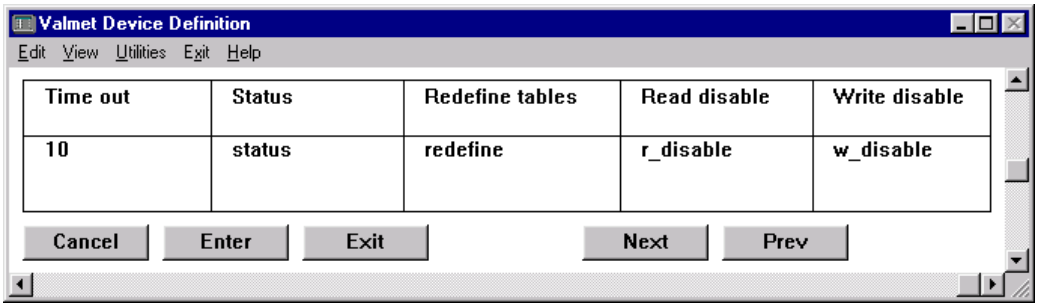
  valid entry: Required/default: 1
  description: Any valid number larger than 1 identifying an existing serial port.

- **Retries**
  The number of retries that the protocol driver will execute when the Valmet computer does not respond or responds negatively to a request.

  valid entry: Required/default: 3.
  description: Any valid number.

- **Time out**
  The time out value for the protocol driver, specified in seconds. The protocol driver will wait for the full response to be received from Valmet during the specified number of seconds. If there is no (valid) response within the time out period the driver will retry.

  valid entry: Required/default: 5.
  description: Any valid number larger than 0.



- **Status**
  Real-time database element written to by the Valmet protocol driver task to indicate the status of a communication command. The status element can be referenced by any task to handle communication error situations. For a detailed list of communication errors see *Appendix B*.

  valid entry: Optional.
  description: Standard FactoryLink tag name of type ANALOG.

- **Redefine Tables**
  Real-time database element forced to ON by the user to trigger a rebuild of the set definitions in the Valmet computer. The rebuild is performed in case the database element has the value ON and the change flags are ON.
  Used in case the Valmet computer has powered on or has lost the definitions downloaded by the driver through some other cause.

  valid entry: Optional.
  description: Standard FactoryLink tag name of type DIGITAL.

- **Read disable**
  Real-time digital database element used to enable/disable read commands for the logical device. Read commands are enabled in case there is no tag defined, or the status of the digital tag is OFF. Read commands are disabled if the status of the tag is ON.

  valid entry: Optional.
  description: Standard FactoryLink tag name of type DIGITAL.

- **Write disable**
  Real-time digital database element used to enable/disable write commands for the logical device. Write commands are enabled in case there is no tag defined, or the status of the digital tag is OFF. Fetch commands are disabled if the status of the tag is ON.

  valid entry: Optional.
  description: Standard FactoryLink tag name of type DIGITAL.

  Note: This tag is only there for future purposes and is currently not used.

5.4. Dataset definition

From the display of all the panels, select the *DeltaLink Valmet Dataset definition* panel.

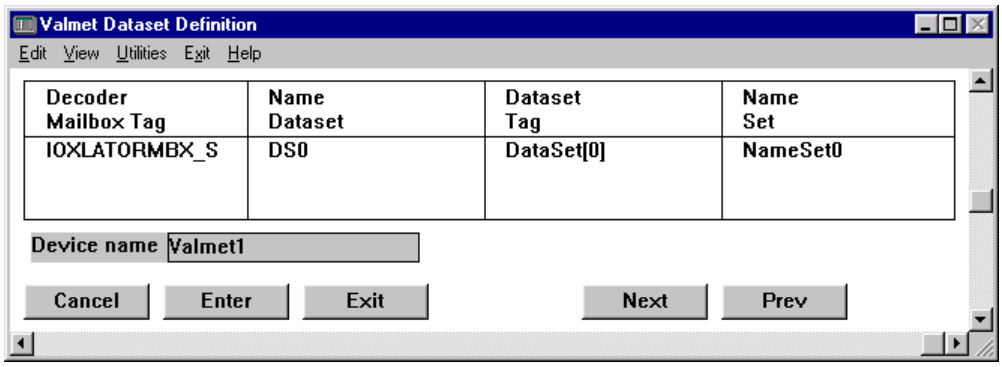


Figure 5.4.1 DeltaLink Valmet Dataset definition.

The DeltaLink Valmet Dataset definition panel allows the user to specify datasets for a particular logical device. Different logical device names are selected in the panel *DeltaLink Valmet Device definition*. The name of the logical device is displayed in the field *Device Name*, at the bottom of the table. Specify the following information.

- **Decoder Mailbox Tag**
  Tag name of a decoder mailbox element, to be referenced by the Valmet protocol driver task (**N.B.** MCI must be supported by the decoder task). The decoder task uses this mailbox to receive data from the Valmet protocol driver.

  valid entry: Required.
  description: Standard FactoryLink tag name of type MAILBOX.

- **Name Dataset**
  Unique name for the data set. The name is used internally in the decoder and protocol driver for activating a data fetch from the external device. A dataset is referenced, internally by the decoder and the protocol driver, by the name. Therefore the name specified in this field must be unique.

  valid entry: Required.
  description: Standard FactoryLink tag name of type DIGITAL.

- **Dataset Tag**
  Tag name representing a (unique) logical name for a data set. The tag is used internally in the decoder and protocol driver for activating a data fetch from the external device. A dataset is referenced, internally by the decoder and the protocol driver, by the tag name. Therefore the tag specified in this field must be unique.

  valid entry: Required.
  description: Standard FactoryLink tag name of type DIGITAL.

- **Name Set**
  Unique name for the set that defines the actual contents of this data set. This name is used to link to the set definition panels. The name is used internally in the protocol driver. Therefore the name specified in this field must be unique.

  valid entry: Required.
  description: Alphanumeric string of up to 18 characters

5.5. Set definition

From the display of all the panels, select the *DeltaLink Valmet Set definition* panel.

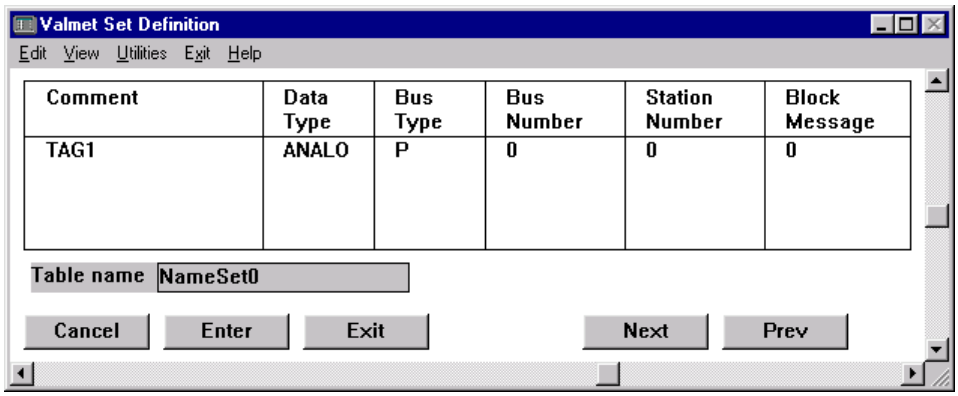


Figure 5.4.1 DeltaLink Valmet Set definition.

The DeltaLink Valmet Set definition panel allows the user to specify the actual contents of a datasets for a particular logical device. Different datasets names are selected in the panel *DeltaLink Valmet Dataset definition*. The name of the set is device is displayed in the field Table *Name*, at the bottom of the table.

Specify the following information (for details on the meaning the user is also referred to the Valmet Damatic manuals):

- **Comment**
  Text to be filled in by the user for documentation purposes. The text can be used for example to store the corresponding 'tagname' within Valmet or to make it easier to define the panels in the decoder (e.g. by inserting here a sequence number).
  The text is not used by the protocol driver.

  valid entry: Optional.
  description: Alphanumeric string up to 20 characters.

- **Data Type**

  Type definition of the value in the Valmet Damatic computer.

  ANALOG refers
  valid entry: Required/default:A.
  description:

  There are 3 types possible: ANALOG (or A or ANA) , COUNTER (or C or COU) , DIGITAL. (or D or DIG).
  ANALOG specifies an analog 16 bit value for example the level of a tank.
  COUNTER specifies a 32 bit counter value.
  DIGITAL specifies to fetch a number of bits (digitals) together with associated status information. The number of bits that is actually fetched depends on the address that is specified. (See further on).

- **Bus Type**

The type of bus to access.

valid entry: Required/default:P.
description:

Two bus types are possible: PROCESS (or P) and CONTROL (or C). PROCESS specifies to access the process bus. CONTROL specifies to access the control bus.

- **Bus number**

Specifies the number of the bus to access.

valid entry: Required/default:0.
description: 0..7

- **Station Number**

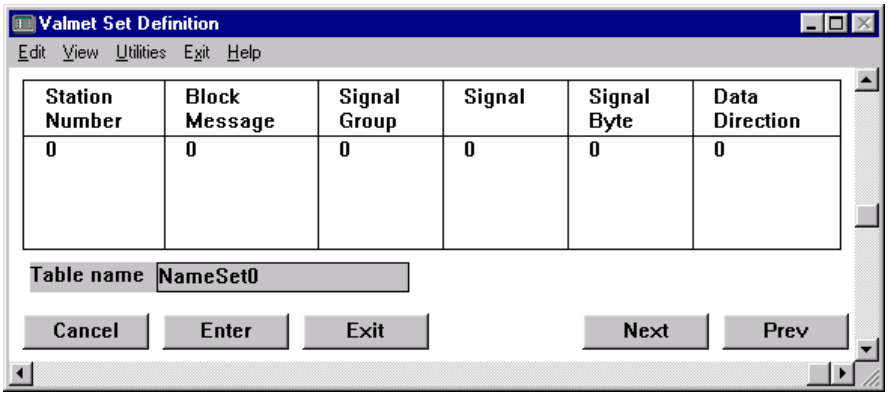
Specifies the number of the station to access.

valid entry: Required/default:0.
description: 0..15

- **Block Message**

Specifies the number of the block to access within the station.

valid entry: Required/default:0.
description: 0..7



- **Signal group**

Specifies the number of the group to access within the block message.

valid entry: Required/default:0.
description: 0..3

- **Signal**

Specifies the signal to access within the group.

valid entry: Required/default:0.
description: 0..3. Also:

In case a COUNTER value is accessed, the signal number should be 0 or 2.

- **Signal byte**

  Specifies (in case of a digital signal only) the byte to access.

  valid entry: Required/default:0
  description: 0..1. Also:

  For non-digital signals this value must be 0.
  For digital signals: if 0 is filled the lower byte is accessed., if 1 is specified the higher byte is accessed.

- **Data direction**

  Specifies the data direction.

  valid entry: Required/default:0.
  description: 0..15. Also:

  In case data from a process station is accessed, a 0 must be specified. Otherwise 1 value greater than 0 must be specified. The number then specifies the control room station to access.

- **Data direction**

  valid entry: Required.
  description: Alphanumeric string up to 20 characters.

5.6. Additional remarks

5.6.1. Size restrictions

For an external system to be able to access values in the Valmet Damatic computer, the external system must carry out a two-stage action.

- First, the external system must define the values that need to be accessed. The definition must be sent to the Valmet computer;
- Second, data can be read by accessing the definitions that have been previously sent.

The protocol driver takes care of sending the definition for a dataset to Valmet the first time a dataset is accessed after start-up of the driver of the first after a 'table redefine' is requested (see the description of the Device panel).

Due to protocol limitations, a dataset can access at most 32 values. Counter values are twice as large and are counted double. So one can define in a dataset at most 32 analog/digital values or at most 16 counter values. If counter values and non-counter values are mixed the number of values in a dataset ranges between 16 and 32.

Due to limitations on the size of the definition table in the Valmet computer, each device can define at most 200 values in all the sets that belong to the device. Counter values are twice as large and are counted double. So one can define in all datasets of a device at most 200 analog/digital values or at most 100 counter values. If counter values and non-counter values are mixed the number of values in a device ranges between 100 and 200.

If the user makes a device and/or dataset definition that violates these constraints, the driver will report the error and exit immediately.

5.6.2. Errors in set definitions

In case the user defines an Valmet address in a set that does not exist or defines an address that does not match the defined datatype (for example specifying the datatype DIGITAl in the protocol driver while the value is actaully an analog value) results in Valmet returning 2 bytes (or 4 in case of a counter) filled with the hexadecimal value FF. A corresponding converted (!) value will be displayed by FactoryLink.

5.6.3. Translator tag definitions

To access the values read from the Valmet device, the user needs to define Decoder panels accordingly. For the Valmet driver some special precautions must be taken. These precautions will be described below.

Due to the nature of the Valmet Datamatic CIS interface it is not possible to use absolute addressing. Relative addressing must be used.

The address field must be filled as follows:

- fill 0 to access the fault code that Valmet always sends back.
- the address of the entry in the Decoder dataset that matches the entry <n> in the corresponding protocol driver dataset is computed from the address of entry <n-1> by adding 1 to the previous address if entry <n-1> is analog or digital and by adding 2 to the previous address if entry <n-1> is a counter.

Analog values

For analog values the Valmet computer sends, apart from the value itself, also status-like information related to the value. To access 'status' information and store it in a digital tag, you must specify the same address as for the analog value itself and use a bit field as follows:

Controller output value

- fill 0 to access the fault code for a controller output value
- fill 1 to access the auto/manual status for a controller output value

Measurement result

- fill 0 to access the O.K. status (value 0) or fault in process connection error (value 1) for a measurement result.
- fill 0 to access the range overflow error (value 1 if overflow).

Digital values

For every digital value defined in the Valmet set, the Valmet computer sends back 8 bits (depending on the signal byte) and, depending on the type of value in Valmet, some status information. To access 'bit' and 'error' information and store it in a digital tag, you must specify the address as explained above and use a bit field as follows (the titles specify the type of value that is accessed in Valmet):

Information of 4 controlled selector switches

- fill for selector <i> (i=0..3) <8+2*i> to access the open (value 0) or closed (value 1) position of the switch.
- fill for selector <i> (i=0..3) <9+2*i> to access whether the switch is in intermediate position (value 1) or not (value 0).
- fill 0 or 1 to access general error information for this area.

Information of 8 switches

- fill for selector <i> (i=0..7) <8+i> to access the open (value 0) or closed (value 1) position of the switch.
- fill 0 or 1 to access general error information for this area.

Information about controller mode

- fill 0 to access whether the output indicates continous control (value 0) or indicates the need to change control (value 1).
- fill 1 to access whether the controller is on tape 1 (value 0) or on tape 2 (value 1).
- fill 2 to access whether the block operates as a controller (value 0) or as a integrator (value 1).
- fill 3 to access whether a computer set value exists (value 1)or not (value 0)
- fill 4 to access whether a remote set value exists (value 1)or not (value 0)
- fill 5 to access whether Damatic controls the setpoint (value 0) or whether the computer controls the setpoint (value 1).
- fill 6 to access whether the controller is in local mode (value 0) or remote mode (value 1).
- fill 7 to access whether the controller is in manual mode (value 0) or auto mode (value 1).

Conversion types

In the decoder you must also specify a conversion to access the requested (analog) value. Fill the conversion type as follows:

To access the analog value itself, specify ANALOG.
To access the lower alarm limit , specify ANALOG.
To access the upper alarm limit , specify ANALOG.
To access the T_I or K_N parameter, specify ANALOG.
To access the K_p or T_D parameter, specify ANALOG.

This page is left blank intentionally.

Appendix A. The delta.opt file

In order to run the Valmet Damatic CIS driver permanently with full functionality either an authorisation sequence or a protection hardware key with the Valmet Damatic CIS option is needed. This appendix describes the authorisation sequence together with the option file.

The authorisation sequence of every independent DeltaLink module must reside in the *{FLOPT}/delta.opt* file in order to take effect. In case the Valmet Damatic CIS driver has been ordered with an authorisation sequence then this sequence resides on the install diskette (in the *opt**delta.opt* file) and will be automatically added to the *{FLOPT}/opt file* with the installation. In case the Valmet Damatic CIS driver has been previously installed from a demo diskette and the authorisation sequence has been purchased later then the sequence must be added manually to the *{FLOPT}\delta.opt* file.

The format of the delta.opt file consists of two parts. The first part is the comment header. This part remains always at the beginning of the file. Every line of the comment part starts with an '*' character. The second part contains the authorisation sequences of every independent DeltaLink module. Every line must hold one sequence code and must apply to a strict format.

An example of the option file looks like this:

```
*
* Copyright 1994 DeltaLink bv. All Rights Reserved
*
* DeltaLink bv
* Saffierborch 6c
* NL-5241 LN Rosmalen
* The Netherlands
* Tel: (int) 31 4192 20025
* Fax: (int) 31 4192 15451
*
*
* FactoryLink Serial Number: 12345OS2
*
* DeltaLink module option:
*
* MODULE = taskname of always 8 characters with trailing spaces
* .      = <space>
* X      = authorisation code supplied by DeltaLink
*
* MODULE..XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX
VALMET  2D6F E047 2534 036E 1215 EC80 92F8 1EFC C4C0
DECODER  B623 C712 D3A7 A08D 8F96 9F14 017B 7D7C F073
```

In case a full installation diskette with authorisation diskette has been ordered then the header in the option file on the diskette contains the serial number of FactoryLink. The Valmet Damatic CIS driver will run only on the FactoryLink package with this serial number. If the serial number is not listed in the *{FLOPT}\delta.opt* file due to a previous demo installation then this number can be added in the header of this file.

The authorisation code must exactly match the format as listed in the header. If this is not the case the module will not recognise the authorisation sequence and start up in demo mode.

The format of an authorisation sequence line is as follows:

MODULE<s><s>XXXX<s>XXXX<s>XXXX<s>XXXX<s>XXXX<s>XXXX<s>XXXX<s>XXXX<s>XXXX<CR><LF>

The **MODULE** field contains the DeltaLink module name in this case Valmet. This field must always be 8 characters long. If the module name is shorter then 8 characters then the name must be filled out with spaces to 8 characters.

After the MODULE field one space must be entered;

After the space field 9 records must be specified with the authorisation code. One record is build up of one leading space (ASCII 0x20) and four sequence codes. The sequence codes must be entered exactly as specified by DeltaLink.

After the authorisation code records a carriage return (ASCII 0x13) and linefeed (ASCII 0x10) must follow.

There may be no empty lines between the specification of more than one module. To add an authorisation sequence a normal editor can be used. If all modules with the right authorisation codes are specified according to the format described above then the modules will start with full functionality.

Appendix B. Valmet protocol driver error codes

The error code is returned to the user in a user-defined status tag. These error codes will also be printed with the message of the DeltaLink Valmet protocol driver in the run-time manager. The errors can be generated from different parts of the driver which will be listed here:

Mailbox Communication Interface errors

Error #	Description
100	Bad message type
101	Message with dataset control tag not found in queue
102	No messages available to query
103	Bad receive mailbox tag
104	Bad mailbox send tag
105	Bad dataset control tag

Valmet errors

Error #	Description
350	A read was requested while reading was disabled.
351	A write was requested while writing was disabled.
352	The com port could not be opened.
353	An incorrect BCC was returned by Valmet.
354	An invalid type of answer was returned by Valmet.
355	An invalid length was returned by Valmet.
356	An invalid number of words was returned by Valmet.
357	A time out occurred before a valid or expected message was returned.
358	Extraneous characters followed the checksum byte.
369	The message returned by Valmet had an unexpected composition.

FactoryLink errors

Error #	Description
401	Internal error
402	Out of memory
403	Operating system error
404	Initialisation not successful
405	Initialisation not successful
406	Incorrect function
407	Incorrect argument
408	Incorrect data
409	Bad tag
410	Null pointer assignment
411	Change flag not set
412	Procedure table full
413	Bad procedure name
414	Bad user name
415	Bad option
416	Incorrect checksum
417	No options
418	No key
419	Bad key
420	No port available
421	Port busy
422	FL already active
423	No lock
424	Lock failed
425	Lock expired
426	Wait failed
427	Termination flag set
428	Q-size to big

429	Q-size changed
430	No tag list
431	Tag list changed
432	Wakeup failed
433	No signals
434	Signalled
435	Not a mailbox
436	No messages
437	Access denied
438	Attribute failure
439	Invalid attribute
440	Attribute not defined
441	Application exists
442	RTDB does not exist
443	No task bit
444	Not a lite task

Appendix C. Messages

If an error condition occurs in the protocol driver task during run-time mode, a message to that effect will appear on the runtime manager graphics screen to the right of "Valmet". The error messages that may be displayed are as follows:

DeltaLink protection key missing

The protocol driver task was unable to detect a proper functioning key. This is a fatal error. Install the key and/or software correctly, power up a printer connected to the key or set the printer online.

Can't open CT file

The protocol driver task was unable to open the configuration table file, generally because it does not exist. This is a fatal error.

No triggers defined

The protocol driver configuration table file does not contain an expected trigger. This is a fatal error.

No Datasets defined

The protocol driver configuration table file does not contain a valid dataset definition. This is a fatal error.

No Decoders defined

The protocol driver configuration table file does not contain a valid decoder definition. This is a fatal error.

Unknown decoder table %s record %d type %d

The protocol driver configuration table file contains an unknown decoder mailbox tag, for the specified table, record number and type of dataset. This is a fatal error.

Reading CT index

An error occurred during reading a Configuration Table index, normally this means the CT-file is corrupted. This is a fatal error.

Reading CT header

An error occurred during reading a Configuration Table header, normally this means the CT-file is corrupted. This is a fatal error.

Reading CT record

An error occurred during reading a Configuration Table record, normally this means the CT-file is corrupted. This is a fatal error.

Out of RAM

There is not enough RAM memory to load the complete configuration and/or task.

Invalid tag number

The protocol driver encountered an invalid TAG number. This is a fatal error.

Bad tag %s %s %d

The protocol driver encountered an incorrect TAG definition. The device, functionality and the error number are displayed.

Reading data %s %s, %d

The protocol driver encountered an error reading a TAG. The device, functionality and the error number are displayed.

Mbx query %s mbx, %d

The protocol driver encountered an error reading a mailbox TAG. The name of the TAG and the error number are displayed.

Change read %s %s, %d

The protocol driver encountered an error during a change read of a TAG. The device, functionality and the error number are displayed.

Forced write %s, %d

The protocol driver encountered an error during forced writing a TAG. The device and the error number are displayed.

MCI init %d

MCI initialisation error, internal error of the protocol driver task. Specified is the error number.

MCI maximum msg in Decoder MBX %d reached

MCI error, the maximum number of allowed messages in a decoder mailbox tag is reached. Specified is the record number of the decoder mailbox.

MCI initialising PD queue %d

MCI initialisation error, internal error of the protocol driver task. Specified is the error number.

MCI receive from PD mailbox %d

MCI error, internal error of the protocol driver task. Specified is the error number.

MCI write to send mailbox %d

MCI error, internal error of the protocol driver task. Specified is the error number.

MCI search control tag %d,%d device %s %d

MCI error, internal error of the protocol driver task, a previously registrated command in the mailbox could not be found. Specified are the dataset control tag, device name and the error number.

MCI get number of messages %s

MCI error, an error occurred during reading a mailbox tag element. Specified is the error number.

MCI identify dataset by number not supported %d

MCI error, decoder tried to identify a dataset by a number. This is not supported in this version of the protocol driver. Specified is the dataset number.

MCI received msg couldn't be converted to dataset

MCI error, a received message in the mailbox tag element is not recognised as a valid read or write command.

MCI msg adjust %s on ds %d:%d: %d

MCI error, a received message in the mailbox tag element could not be interpreted as a valid read or write command for the external device. Specified are the device name, dataset control tag and the error number.

MCI unknown msg type %d

MCI error, internal error of the protocol driver task. Specified is the number of an unrecognised command.

MCI Bad MSG Control TAG table %s record %d type %d

MCI error, internal error of the protocol driver task. Specified are table number, record number and type number.

MCI Duplicate MSG Ctrl TAG table %s record %d type %d

MCI error, internal error of the protocol driver task. Specified are table number, record number and type number.

MCI Bad Mailbox TAG table %s record %d type %d

MCI error, internal error of the protocol driver task. Specified are table number, record number and type number.

MCI DS for %s function not found, MsgCtrl %d

MCI error, internal error of the protocol driver task, missing dataset for the task. Specified are function and the message control TAG.

Com init comm layer %d

An error occurred during initialising the communication layer Specified is the error number.

Device %s. Error while communicating %s

Status message for the specified device and function. The meaning of the error number can be found in the previous appendix.

Com building device

An error occurred during building the communication datablock for device. This is a fatal error.

Internal error

Internal error of the protocol driver task, contact DeltaLink Benelux for support.

Too many entries in set %s. Number is %d.

The set with the given name violated the size restrictions for a set. The number of entries in the set is given (where counters count double.)

Too many entries in device %s. Number is %d.

The device with the given name violated the size restrictions for a device. The number of entries in the device is given (where counters count double.)

Internal error

Internal error of the protocol driver task, contact DeltaLink Benelux for support.

