# CANbridge NT 420

## USER MANUAL

**4.01.0331.20004 1.0 ENGLISH**


Connecting Devices™

# Important User Information

## Liability

Every care has been taken in the preparation of this document. Please inform HMS Industrial Networks of any inaccuracies or omissions. The data and illustrations found in this document are not binding. We, HMS Industrial Networks, reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be considered as a commitment by HMS Industrial Networks. HMS Industrial Networks assumes no responsibility for any errors that may appear in this document.

There are many applications of this product. Those responsible for the use of this device must ensure that all the necessary steps have been taken to verify that the applications meet all performance and safety requirements including any applicable laws, regulations, codes, and standards.

HMS Industrial Networks will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features, timing, or functional side effects found outside the documented scope of this product. The effects caused by any direct or indirect use of such aspects of the product are undefined, and may include e.g. compatibility issues and stability issues.

The examples and illustrations in this document are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular implementation, HMS Industrial Networks cannot assume responsibility for actual use based on these examples and illustrations.

## Intellectual Property Rights

HMS Industrial Networks has intellectual property rights relating to technology embodied in the product described in this document. These intellectual property rights may include patents and pending patent applications in the USA and other countries.

# Table of Contents

Page

This page intentionally left blank

# 1 User Guide

Please read the manual carefully. Make sure you fully understand the manual before using the product.

## 1.1 Related Documents

| Document | Author |
|---|---|
| Installation Manual VCI Driver | HMS |
| | |

## 1.2 Document History

| Version | Date | Description |
|---|---|---|
| 1.0 | February 2017 | First release |
| | | |

## 1.3 Conventions

Instructions and results are structured as follows:

- ► instruction 1
- ► instruction 2
  - ➡ result 1
  - ➡ result 2

Lists are structured as follows:

- item 1
- item 2

**Bold typeface** indicates interactive parts such as connectors and switches on the hardware, or menus and buttons in a graphical user interface.

```
This font is used to indicate program code and other
kinds of data input/output such as configuration scripts.
```

This is a cross-reference within this document: *Conventions, p. 4*

This is an external link (URL): www.hms-networks.com

Safety advice is structured as follows:

> **!** Cause of the hazard!
>
> Consequences of not taking remediate action.
>
> How to avoid the hazard.

Safety signs and signalwords are used dependent on the level of the hazard.

*This is additional information which may facilitate installation and/or operation.*

> **!** This instruction must be followed to avoid a risk of reduced functionality and/or damage to the equipment, or to avoid a network security risk.

> **Caution**
> This instruction must be followed to avoid a risk of personal injury.

> **WARNING**
> This instruction must be followed to avoid a risk of death or serious injury.

# 2 Safety and Dangers

## 2.1 Information on EMC

> **!** Risk of interference to radio and television if used in office or home environment!
>
> Use exclusively included accessories.
>
> Make sure shield of interface is connected with device plug and plug on other side.
>
> Use exclusively shielded cables.

## 2.2 General Safety Notes

- ► Protect product from moisture and humidity.
- ► Protect product from too high or too low temperature (see *Technical Data, p. 28*).
- ► Protect product from fire.
- ► Don't throw, drop or try to bend the product.
- ► Don't paint the product.
- ► Don't modify or disassemble the product. Service must be carried out by HMS Industrial Networks.
- ► Don't use modified products.
- ► Store products in dry and dust-free place.

# 3 Scope of Delivery

Included in scope of delivery:

- CANbridge NT 420
- 1x power connector
- 4x CAN connector
- User Manual *CANbridge NT 420*
- Installation Manual *VCI Driver*
- CD with CAN-Gateway Configurator and VCI driver
- Mini USB cable

# 4 Features

The CANbridge NT 420 is a universal, intelligent CAN topology component, that allows the coupling of four Classic CAN or two Classic CAN and two CAN FD networks, also with different bit rates or frame formats. CAN messages are received by one network and then transmitted in the other network, according to filter and implementation rules.

The CANbridge NT 420 application firmware includes a filtering mechanism based on CAN identifiers. As basis serves a filter list, which contains CAN identifiers. Selected CAN messages can be forwarded and others discarded. The CANbridge NT 420 provides message filtering for all CAN connections.

Features and highlights:

- 4x CAN connections, ISO 11989-2 (terminal adapters)
- 1x mini USB 2.0 port, high-speed
- Configuration via USB
- With the included CAN-Gateway Configurator a configuration can be created, modified, written to and read from the target device via USB connection.
- Star Coupler and Bridge setup possible
- Freely configurable baud-rates
- Automatic baud-rate detection
- CAN ID filtering, configured via the configuration tool IXXAT CAN-Gateway Configurator
- Bridging between Classic CAN and CAN FD networks

# 5 Installation

## 5.1 Installing Software

### 5.1.1 Installing the Driver

For the operation of the CAN-Gateway Configurator a VCI driver version 3.5 or higher is needed.

**Windows**

► Install VCI driver (see installation manual *VCI Driver*).

### 5.1.2 Installing the CAN-Gateway Configurator

► Insert CD-ROM in CD drive.

► Run *CanGWconfig_xy.exe*.

➡ Hardware wizard starts automatically.

► Follow instructions in installation program.

*It is possible to download the CAN-Gateway Configurator from www.ixxat.com.*

## 5.2 Connectors

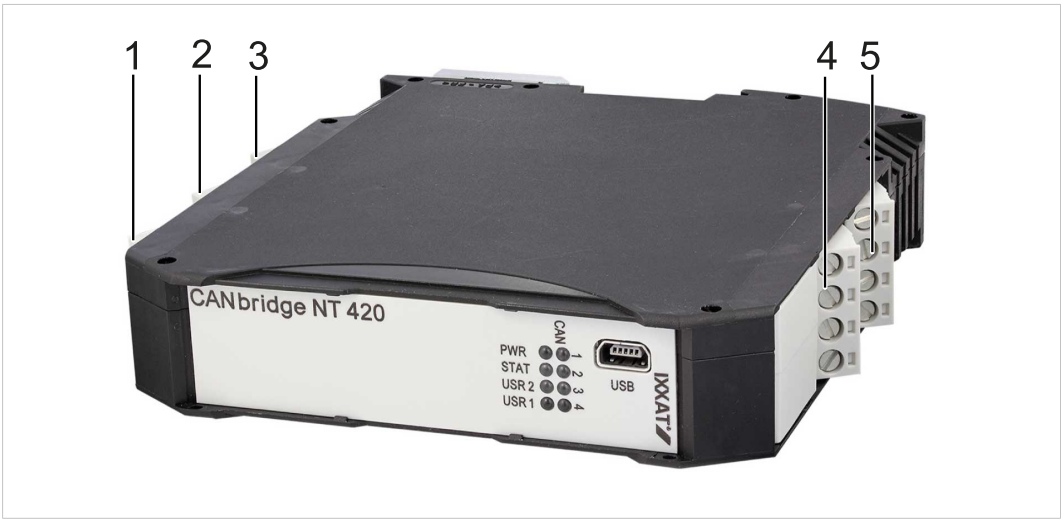



**Fig. 1 Connectors**

| | |
|---|---|
| 1 | CAN 1 |
| 2 | CAN 2 |
| 3 | Power connector |
| 4 | CAN 3 |
| 5 | CAN 4 |

- ► Make sure that the cross-sectional area of cable is equal to or larger than 0.14 mm² resp. AWG 25.
- ► To remove the connector use screwdriver or similar tool.
- ► Connect cables.
- ► Plug connector into housing.

The shield of the CAN connector is connected to the device ground and the PE on the back of the device (DIN rail) via a 1 MΩ resistor and a 10 nF capacitor. To achieve highest interference immunity, ground the shield of the CAN cable.

### 5.2.1 Power Connector


**Fig. 2 Power connector**

**Pin Allocation of Power Connector**

| Pin no. | Signal |
|---|---|
| 1 | V+ (+9 V to +32 V DC) |
| 2 | V- |
| 3 | - |
| 4 | - |

### 5.2.2 CAN Connectors


**Fig. 3 CAN connector**

**Pin allocation of CAN connector CAN 1 to CAN 4**

| Pin no. | Signal |
|---|---|
| 1 | CAN-High |
| 2 | CAN-Low |
| 3 | CAN-GND |
| 4 | Shield |

# 6 Configuration

## 6.1 Basic Configuration

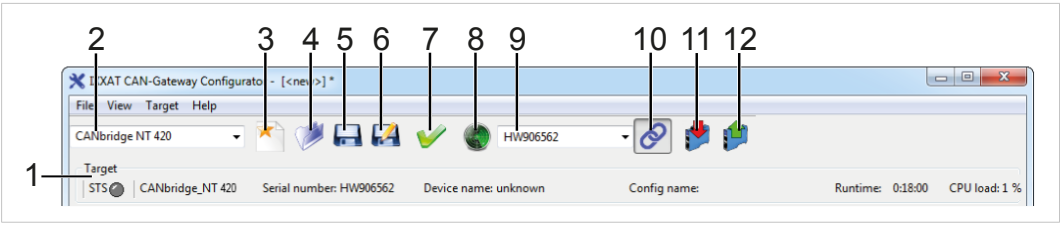



**Fig. 4 CAN-Gateway Configurator**

| | |
|---|---|
| 1 | Information about target device, **STS** (status) reflects Power LED of device |
| 2 | Drop-down list **Select device type** |
| 3 | Button **New** |
| 4 | Button **Open** |
| 5 | Button **Save** |
| 6 | Button **Save as** |
| 7 | Button **Verify** |
| 8 | Button **Scan** |
| 9 | Combo box **Target device** |
| 10 | Button **Connect** |
| 11 | Button **Write to** |
| 12 | Button **Read from** |

- ► Connect a device to the host computer via Mini USB cable.
- ► Start IXXAT CAN-Gateway Configurator.
- ► Select CANbridge NT 420 in drop-down list **Select device type** (**2**).
- ► Click button **Scan** (**8**).
- ► Select device in combo box **Target device** (**9**).
- ► Click button **Connect** (**10**) to connect the selected device.
  - ➡ Most recently used project is opened.
  - ➡ If device is connected, **STS** (**1**) is green flashing.
  - ➡ Information about target device are shown (**1**).
- ► To create a new project file, click button **New** (**3**).

  or
- ► To change the current configuration of the device, click button **Read from** (**12**) and save the configuration.

  or
- ► To open an existing configuration file (not from CAN-Gateway Configurator), open menu **File** and select **Import legacy configuration**.
- ► Configure device as Star Coupler (see *Star Coupler Configuration, p. 10*) or as bridge (see *Bridge Configuration, p. 13*).
- ► To write configuration to device, click button **Write to** (**11**).

## 6.2 Star Coupler Configuration

In Star Coupler setup all messages are transmitted unchanged to the other port and all four CAN ports operate in Classic CAN mode. Filters or CAN-ID modifications are not possible.

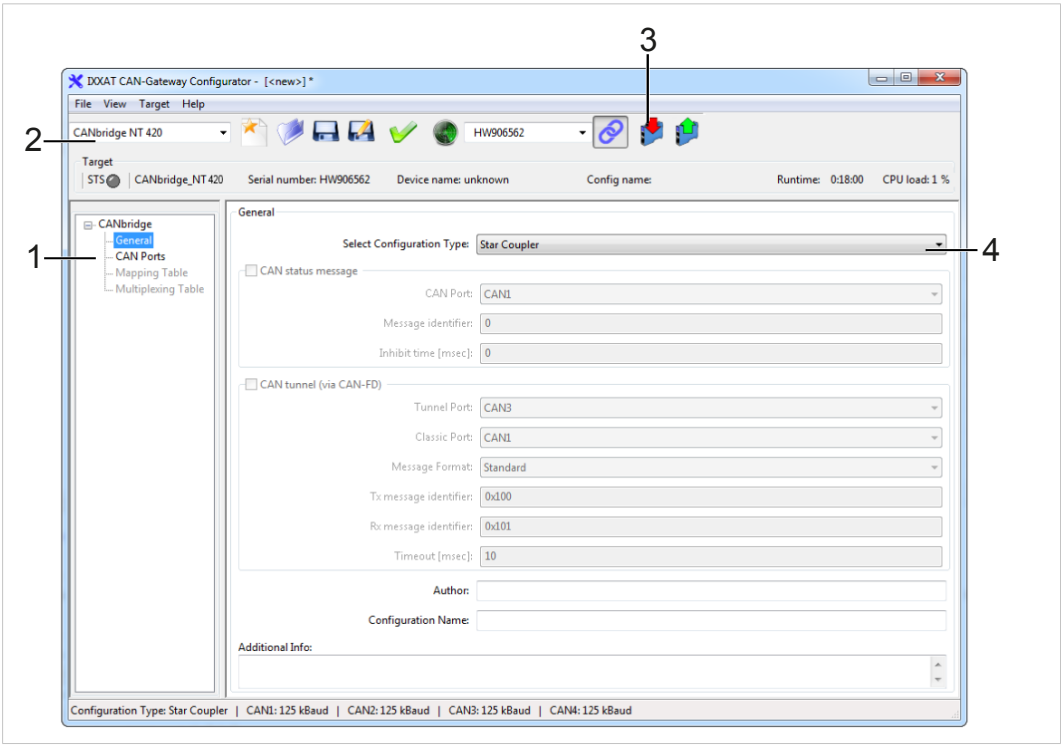



**Fig. 5 Star coupler configuration**

*It is possible to add information about the configuration in fields **Author**, **Configuration Name** and **Additional Info**.*

- ► Make sure that correct device type is selected (**2**).
- ► Select **General** in configuration tree (**1**).
- ► In drop-down list **Select configuration type** (**4**) select **Star Coupler**.
  - ➡ Mapping table is deactivated.
  - ➡ All messages are transmitted unchanged.
- ► Set baud-rate for CAN ports 1 to 4 (see *Baud-Rate Settings, p. 11*).
- ► To write configuration to device, click button **Write to** (**3**).

### 6.2.1 Baud-Rate Settings

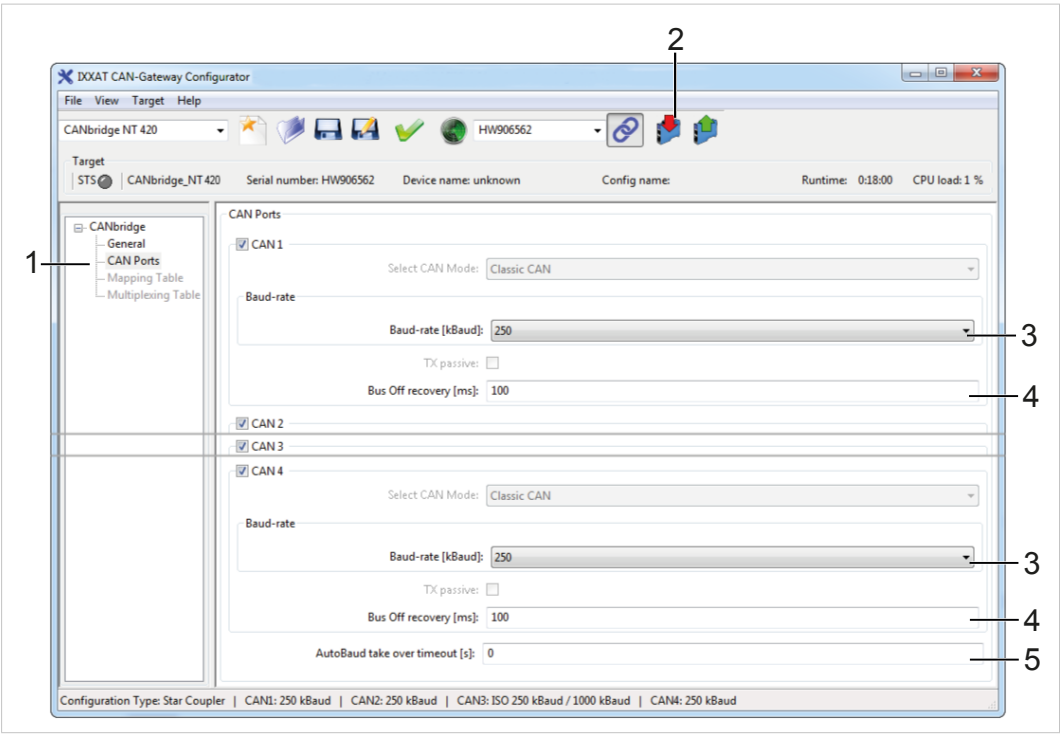



**Fig. 6 Baud-rate settings**

► Select **CAN Ports** in configuration tree (**1**).

➡ Form to set baud-rate appears on right side.

► Configure baud-rate for each port in drop-down lists **Baud-rate** (**3**).

Setting the baud-rate is possible in different ways:

- predefined CiA baud-rate (listed in drop-down list)
- setting with bit timing register (see *Setting with Bit Timing Register, p. 18*)
- automatic baud-rate detection (see *AutoBaud Take Over Timeout, p. 12*)

It is possible to configure for each CAN port wether the CAN controller restarts automatically after a bus off.

► Set time in milliseconds for a restart of the CAN controller after a bus off for each CAN port in fields **Bus Off recovery** (**4**).

► Choose value between 1000 and 60000 milliseconds (value 0 means no recovery).

► To write configuration to device, click button **Write to** (**2**).

### AutoBaud Take Over Timeout

> **!** Automatic baud-rate detection is exclusively possible if at least two nodes per segment are active.

*Setting of an AutoBaud take over timeout is exclusively possible in Star Coupler configuration.*

In Star Coupler configuration it is possible to set a timeout for the automatic baud-rate take over. When a baud-rate is detected on one bus the timeout time starts:

- If no baud-rate is detected on any other bus, the detected baud-rate is set on all other busses, when the timeout time expires.
- If the same baud-rate is detected on several busses, the baud-rate is set on all other busses, when the timeout time expires.
- If several different baud-rates are detected, there is no Autobaud take over.

When baud-rate is adopted, transmitting is attempted. If an error occurs during transmission, the device is switched off. The devices changes to state *Operational* when baudrate is set on every bus.

In the standard configuration no timeout is set. The device remains in automatic baud-rate detection until baud-rate is detected on both busses.

► In drop-down list **Baud-rate** (**3**) select **automatic baud-rate detection**.

► To set a timeout for the automatic baud-rate take over, specify time in field **AutoBaud take over timeout** (**5**).

For information about automatic baudrate detection sequence indicated by LEDs see *Automatic Baud-rate Detection Sequence, p. 26*.

## 6.3 Bridge Configuration

By using the mapping table and the multiplexing table the Bridge setup allows the free configuration of the transmission of CAN messages between the CAN ports, also between Classic CAN and CAN FD.

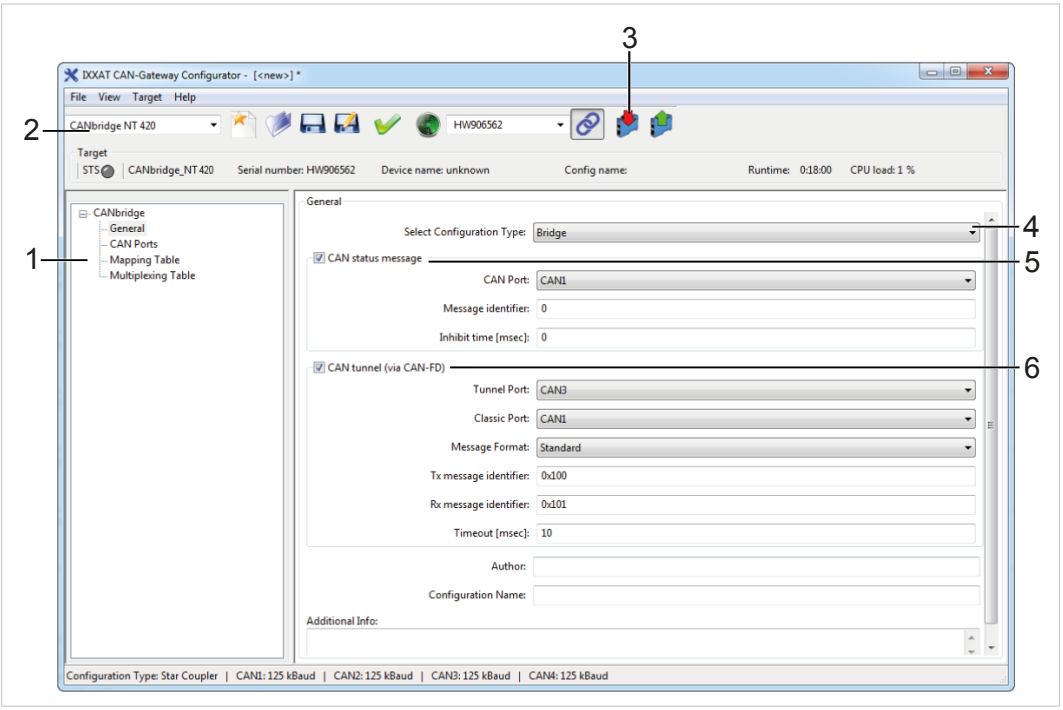



**Fig. 7 Bridge configuration**

- ► Make sure, that correct device type is selected (**2**).
- ► Specify the following settings:
  - – general settings (see *General Settings, p. 13*)
  - – baud-rates (see *Baud-Rate Settings, p. 16*)
  - – mapping table (see *Configuring the Mapping Table, p. 20*)
  - – multiplexing table (see *Configuring the Multiplexing Table, p. 24*)
- ► To write configuration to device, click button **Write to** (**3**).

### 6.3.1 General Settings

*It is possible to add information about the configuration in fields **Author**, **Configuration Name** and **Additional Info**.*

- ► Make sure that correct device type is selected (**2**).
- ► Select **General** in configuration tree (**1**).
- ► In drop-down list **Select configuration type** (**4**) select **Bridge**.

## CAN Status Message (5)

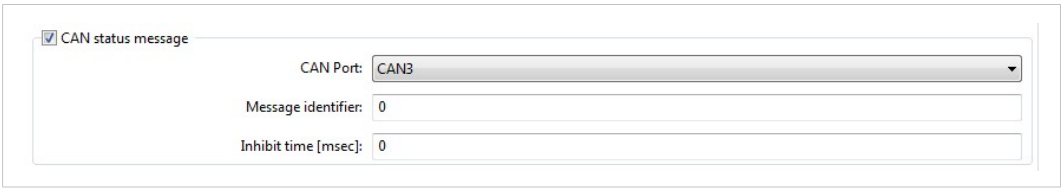



**Fig. 8 CAN status message settings**

It is possible to activate a CAN status message, in which the error state of all CAN ports is transmitted. With every change of the error state of a CAN port a CAN status message is transmitted.

**Data field of a CAN status message**

| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 |
|---|---|---|---|---|---|---|---|
| CAN 1 | CAN 2 | CAN 3 | CAN 4 | R | R | R | R |

The error state of the CAN ports is in the first 4 data bytes of the CAN status message. Byte 1 is the first data byte after the CAN identifier. The remaining 4 data bytes are reserved.

A CAN status byte can be of one of the following states:

- Error Active (normal state), 0x00
- Error Passive, 0x01
- Bus Off, 0x02
- Not Available, 0xFF

> **!** If hexadecimal values are used, they must begin with 0x.
>
> Example: 0x55

► To transmit CAN status messages, activate check box **CAN status message**.

► In drop-down list **CAN Port** select transmitting port for CAN status message.

► In field **Message identifier** enter identifier in decimal values.

► In field **Inhibit time** specify the minimum time in msec between the transmitting of status messages, if several errors occur.

## CAN Tunnel (6)

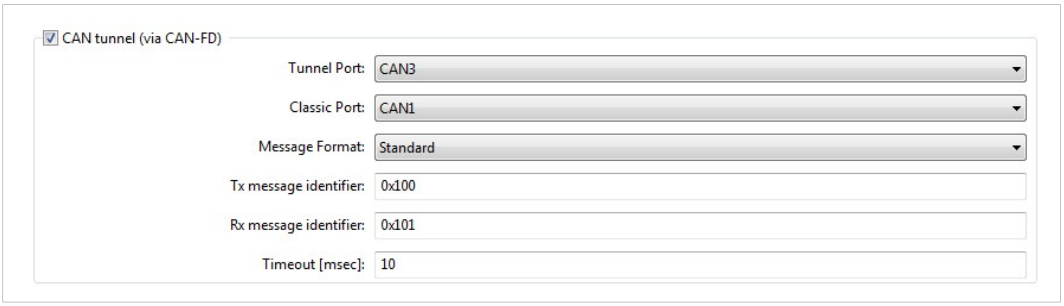



**Fig. 9 CAN tunnel settings**

With two CANbridge NT 420 devices it is possible to transmit messages between two Classic CAN networks via a CAN FD network (CAN tunnel). Only two identifiers are necessary for the CAN FD network, via these two CAN FD messages all Classic CAN messages are transferred. The busload on the tunnel can be reduced due to the usage of the maximum length of 64 bytes. The Tx message identifier of the first device must be configured to Rx messages identifier of the second device and vice versa.

- ► To activate a CAN tunnel via CAN FD between two devices, activate check box **CAN tunnel**.
- ► In drop-down list **Tunnel Port** select transmitting port for CAN FD messages.
- ► In drop-down list **Classic Port** select receiving port for CAN FD messages.
- ► In field **Tx message identifier** enter identifier of message to be transmitted in decimal values.
- ► In field **Rx message identifier** enter identifier of received message in decimal values.
- ► In field **timeout** specify the timeout in msec.

> **!** If hexadecimal values are used, they must begin with 0x.
>
> Example: 0x55

- ► Configure second device.
- ► Make sure, that **Tx message identifier** of first device matches **Rx message identifier** of second device.
- ► Make sure, that **Rx message identifier** of first device matches **Tx message identifier** of second device.

## 6.3.2 Baud-Rate Settings

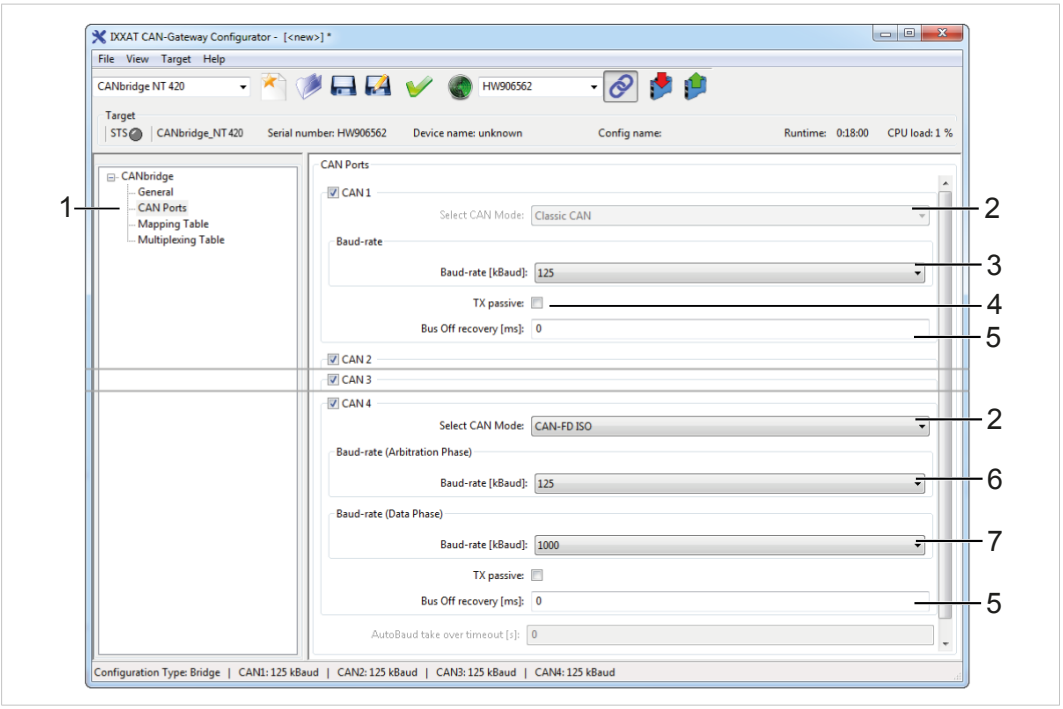



**Fig. 10 Baud-rate Settings Bridge**

► Select **CAN Ports** (**1**) in configuration tree.

➡ Form to set baud-rate for CAN 1 to CAN 4 appears on right side.

### CAN Mode (2)

CAN 1 and CAN 2 are Classic CAN channels.

For CAN 3 and CAN 4 different CAN modes can be selected:

- Classic CAN
- ISO CAN FD
- Non-ISO CAN FD

*CAN FD does not support autobaud detection.*

► Select CAN mode in drop-down list **Select CAN Mode** (**2**).

► In CAN FD mode configure baud-rate for **Arbitration Phase** (**6**) and baud-rate for **Data Phase** (**7**).

### Arbitration Phase and Data Phase

CAN FD uses two baud-rates, one for the arbitration phase, which is limited to the maximum of Classic CAN (1000 kBit/s) and one for the data phase (up to 8 MBit/s).

### Baud-Rate (3)

*CAN FD does not support autobaud detection.*

► Configure baud-rate for each port in drop-down lists **Baud-rate** (**3/6/7**).

► Use predefined baud-rates listed in drop-down list.

or

► Set baud-rate with bit timing register (see *Setting with Bit Timing Register, p. 18*).

### TX Passive Mode (4)

If a CAN port is in TX passive mode, it acts exclusively as listener. It receives messages, but does not transmit messages, nor affect the communication (neither acknowledgement bit nor error frames are generated).

► To set port in TX passive mode, activate check box **TX passive mode** (**4**).

### Bus Off Recovery (5)

It is possible to configure for each CAN port wether the CAN controller restarts automatically after a bus off.

► Set time in milliseconds for a restart of the CAN controller after a bus off for each CAN port in fields **Bus Off recovery** (**5**).

► Choose value between 1000 and 60000 milliseconds (value 0 means no recovery).

## Setting with Bit Timing Register

*HMS Industrial Networks recommends to use the predefined standard baud-rates. If user defined baud-rates are used, make sure that the entered values are valid.*

If the baud-rate is set with the bit timing register of the controller, baud-rates that are not defined by CiA can be used.

The clock frequency of the CAN module applied for the calculation of the baudrate is 36 MHz resp. 80 MHz.

Formula for calculation of baudrate:

CAN 1 and CAN 2:

- baudrate [kBaud] = 36000 / ((TSEG1 + TSEG2 +1) * Prescaler)

CAN 3 and CAN 4 (CAN FD):

- baudrate [kBaud] = 80000 / ((TSEG1 + TSEG2 +1) * Prescaler)

► For user defined baud-rates select **user defined via register values** in drop-down list (**2**).

► Set values for **Prescaler**, **SJW**, **TSEG1** and **TSEG2**.

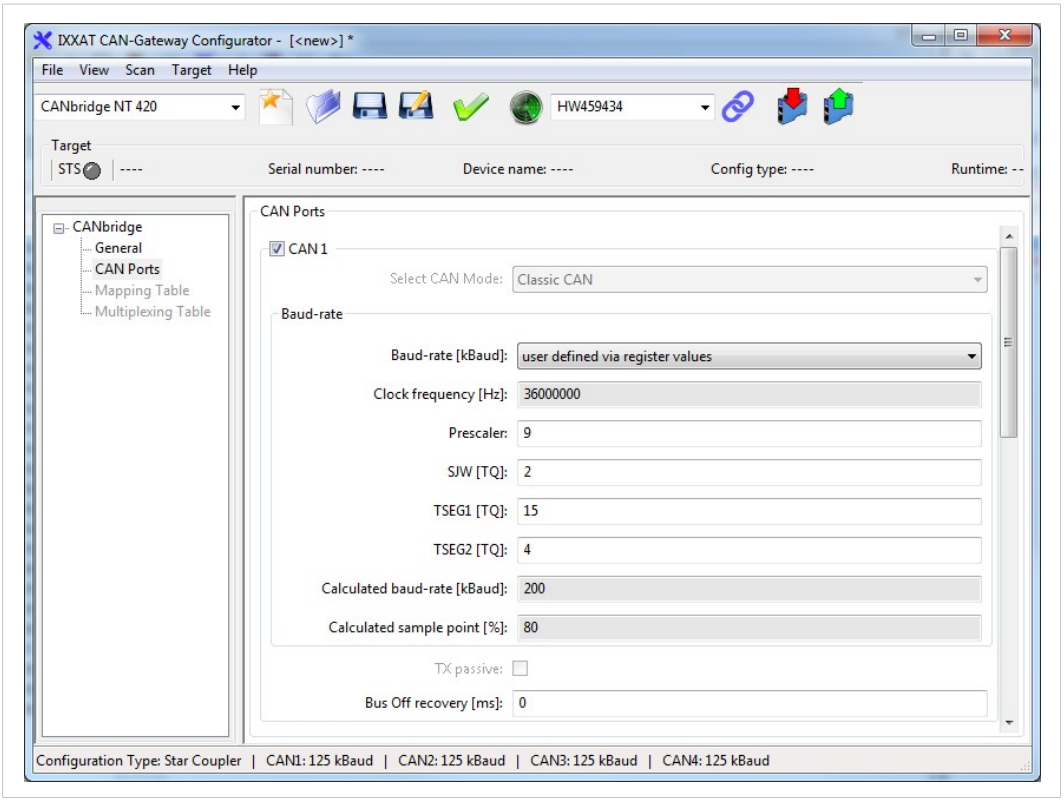



**Fig. 11 Bit time register Classic CAN**

### Setting Recommendations for CAN FD

*HMS Industrial Networks recommends to use in all connected nodes the same bit timing settings.*

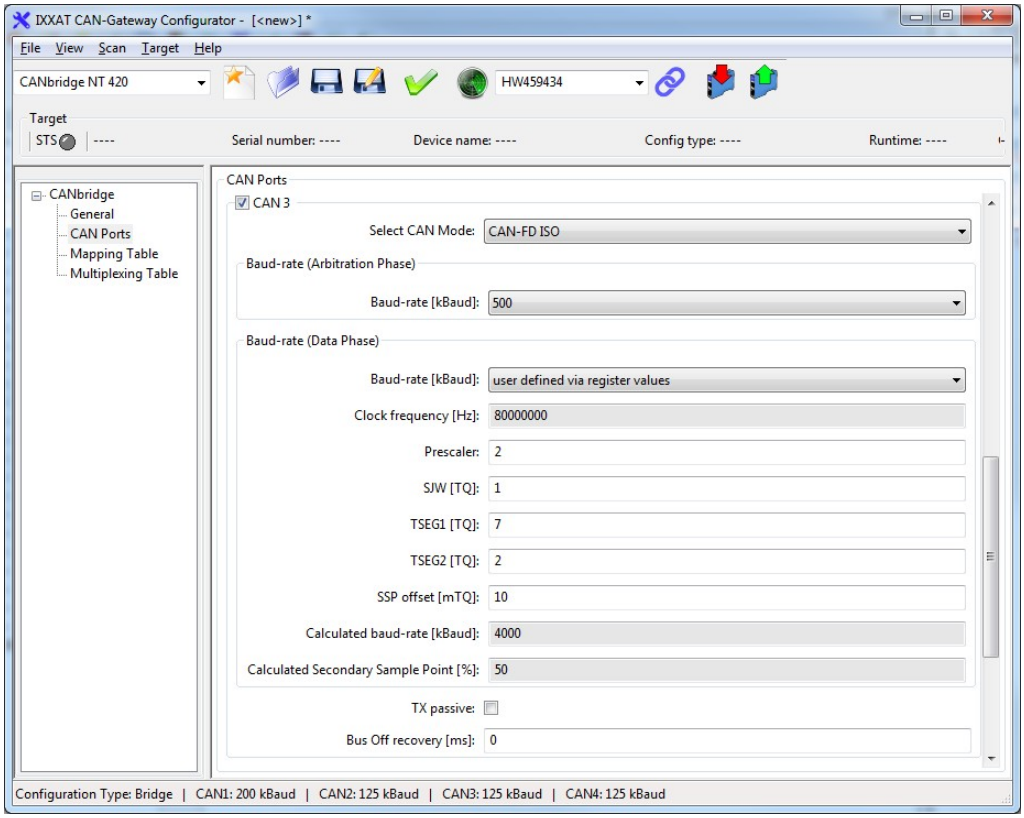



**Fig. 12 Bit time register CAN FD**

Observe the following recommendations:

- Set arbitration and data phase prescaler as low as possible.
- Configure the same arbitration sample point for all CAN nodes.
- Configure the same data phase primary sample point for all CAN nodes.
- Set SJW for arbitration phase as large as possible.
- Set SJW for data phase as large as required by used oscillator (clock source).

To provide optimal compatibility with other CAN FD devices and stable communication with the network, make sure that the calculated secondary sample point is about 50 %. To achieve this use the following formula to set SPP offset:

- SSP = ((TSEG1 + TSEG2 +1) * Prescaler) / 2

## 6.3.3 Configuring the Mapping Table

The IXXAT CAN-Gateway Configurator allows free routing configurations. Individual messages or message groups can be mapped from and to each CAN port.

The route through the CANbridge NT 420 always starts at the receiving CAN controller (message source) and ends at the transmitting CAN controller (message destination).

*It is possible to configure the mapping table offline.*

*It is possible to load a .csv file of an existing mapping table. The rows and columns must be the same as in the integrated mapping table. An example is included on the delivery CD.*

### Entry limitations

The size of the mapping table is limited.

Maximal possible entries:

- maximal 512 rows in total
- limitations of extended format:
  - maximal 256 identifier entries per CAN
  - maximal 8 mask/value entries per CAN

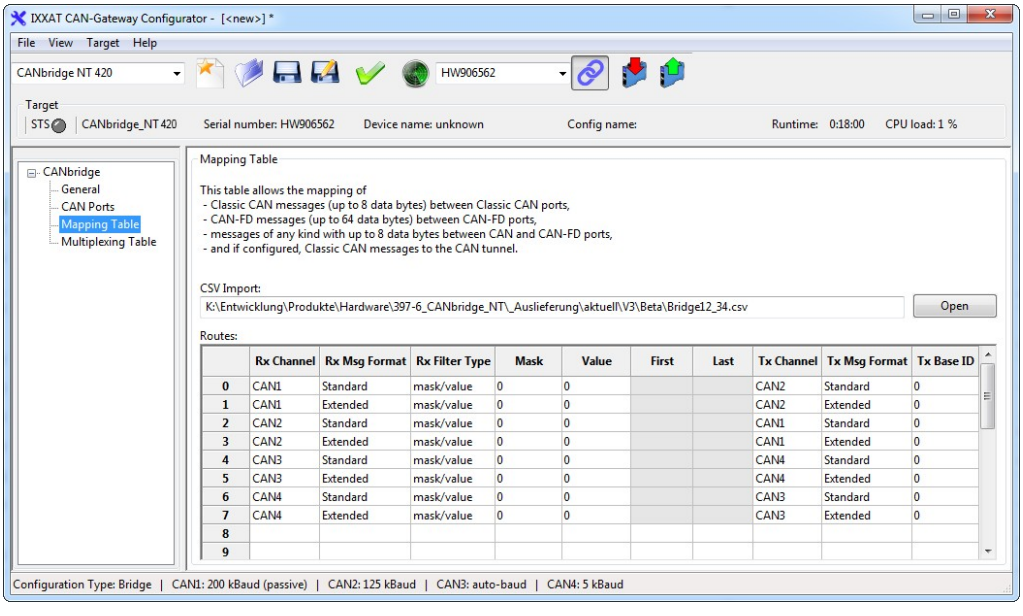



**Fig. 13 Mapping table**

► Select **Mapping Table** in configuration tree.

➡ Form to specify mapping table appears on right side.

*If new mapping tables are loaded, already available mapping table entries will be deleted.*

► To load a mapping table click button **Open** (**2**).

➡ Window **Select a File** is opened.

or

► Configure routing for each group of CAN messages.

> ! In extended format a range filter is not possible.

> ! If hexadecimal values are used, they must begin with 0x.
> Example: 0x55

► To mark a row right-click on the left number column.

► Select a row.

► To open context menu left-click.

► Click on cell to edit cell content.

➡ Drop-down list is opened.

► When a row is completely defined, click button **Verify**.

➡ In status window status, error and warnings are showed.

**Possible Entries**

| Column | Possible entries |
|---|---|
| Rx Channel | CAN1, CAN2, CAN 3, CAN 4 |
| Rx Msg Format | Standard, Extended<br>Defines format in which messages are transmitted, standard (11 bit identifiers) or extended (29 bit identifier). |
| Rx Filter Type | Identifier, mask/value, range,<br>filter range is exclusively possible in standard format. |
| Mask | Used with mask/value filter: defines which bits of an identifier are relevant for the filter and which are not relevant, see *Mask/Value Filter, p. 22*.<br>Decimal and hexadecimal values possible. |
| Value/Identifier | With mask/value filter: defines the values for the filter relevant bits (as defined in **Mask**), see *Mask/Value Filter, p. 22*<br>With identifier filter: defines the identifier<br>Decimal and hexadecimal values possible. |
| First | First value of range (decimal and hexadecimal values possible) |
| Last | Last value of range (decimal and hexadecimal values possible) |
| Tx Channel | CAN1, CAN2, CAN 3, CAN 4, Tunnel |
| Tx Message Format | Standard, Extended<br>Defines format in which messages are received, standard (11 bit identifiers) or extended (29 bit identifier). |
| Tx Base ID | With range and mask/value filter: specifies the transmit identifiers to which the received identifiers that passed the filter are mapped.<br>With identifier filter: specifies the transmit identifier |

### Examples Tx Base ID

The defined Rx identifiers pass the filter. These valid messages are then mapped to the transmit messages starting at the message identifier set in TX Base ID.

| Filter type | Rx | Tx Base ID | Transmitted Identifier |
|---|---|---|---|
| Range | First: 0x100<br>Last: 0x200 | 0x300 | 0x300–0x400 |
| Mask/Value | Mask: 0x700<br>Value: 0x100 | 0x200 | 0x200–0x2FF |
| Identifier | 0x123 | 0x456 | 0x456 |

## Mask/Value Filter

With the mask/value filter (available for either 11 bit or 29 bit identifiers) possible valid identifiers based on bit masks can be defined.

Binary representation of mask:

- binary positions with value 1 are relevant for the filter
- binary positions with value 0 are not relevant for the filter

Binary representation of value:

- Defines the values for the positions that are marked as relevant (1) in mask.
- Values in positions that are marked as not relevant (0) in mask are ignored.

The following formula expresses the same condition under which an identifier passes the filter:

- if (value & mask) == (identifier & mask) then identifier is valid

### Examples

**11 bit identifier**

| | hex | bin |
|---|---|---|
| **Identifier** | 0x700 | 0111:0000:0000 |
| **Mask** | 0x700 | 0111:0000:0000 |
| **Value** | 0x700 | 0111:0000:0000 |
| **Result** | 0x700 | 0111:0000:0000 |
| | Any identifier between 0x700 and 0x7FF passes the filter, as only the first 3 bits of the mask are marked as relevant. | |

**29 bit identifier**

| | hex | bin |
|---|---|---|
| **Identifier** | 0x10003344 | 0001:0000:0000:0000:0011:0011:0100:0100 |
| **Mask** | 0x1F00FFFF | 0001:1111:0000:0000:1111:1111:1111:1111 |
| **Value** | 0x10003344 | 0001:0000:0000:0000:0011:0011:0100:0100 |
| **Result** | 0x10003344 | 0001:0000:0000:0000:0011:0011:0100:0100 |
| | Any identifier between 0x10003344 and 0x10FF3344 passes the filter. | |

**Mask/value filter**

| Value | Mask | Valid message identifiers which pass the filter |
|---|---|---|
| 0x100 | 0x7FF | 0x100 |
| 0x100 | 0x700 | 0x100–0x1FF |
| 0x000 | 0x000 | 0x000–0x7FF |

## Mapping Table Example 1

The following mapping table is an example of a bridge, that allows all messages (standard and extended) to pass:

- from CAN 1 to CAN 2 and vice versa
- from CAN 3 to CAN 4 and vice versa

Routes:

| | Rx Channel | Rx Msg Format | Rx Filter Type | Mask | Value | First | Last | Tx Channel | Tx Msg Format | Tx Base ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | CAN1 | Standard | mask/value | 0 | 0 | | | CAN2 | Standard | 0 |
| 1 | CAN1 | Extended | mask/value | 0 | 0 | | | CAN2 | Extended | 0 |
| 2 | CAN2 | Standard | mask/value | 0 | 0 | | | CAN1 | Standard | 0 |
| 3 | CAN2 | Extended | mask/value | 0 | 0 | | | CAN1 | Extended | 0 |
| 4 | CAN3 | Standard | mask/value | 0 | 0 | | | CAN4 | Standard | 0 |
| 5 | CAN3 | Extended | mask/value | 0 | 0 | | | CAN4 | Extended | 0 |
| 6 | CAN4 | Standard | mask/value | 0 | 0 | | | CAN3 | Standard | 0 |
| 7 | CAN4 | Extended | mask/value | 0 | 0 | | | CAN3 | Extended | 0 |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | |
| 10 | | | | | | | | | | |

**Fig. 14 Example 1 mapping table**

## Mapping Table Example 2

The following mapping table is an example of a star coupler, that allows all messages (standard and extended) to pass from every CAN port to every CAN port.

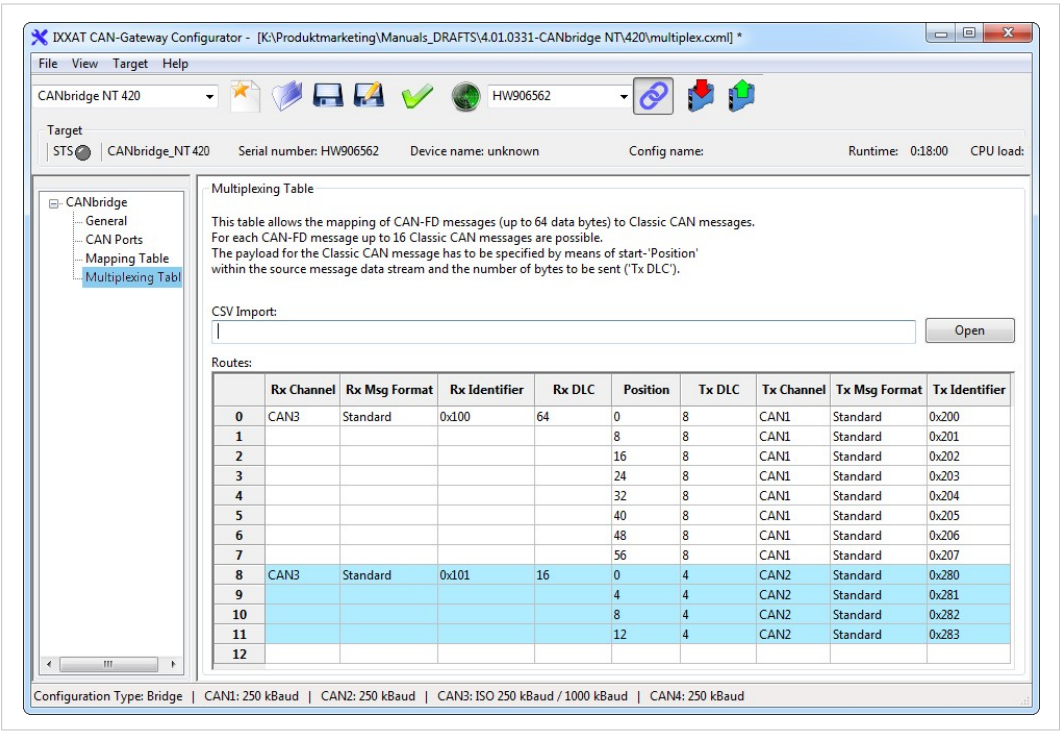
| | Rx Channel | Rx Msg Format | Rx Filter Type | Mask | Value | First | Last | Tx Channel | Tx Msg Format | Tx Base ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | CAN1 | Standard | mask/value | 0 | 0 | | | CAN2 | Standard | 0 |
| 1 | CAN1 | Standard | mask/value | 0 | 0 | | | CAN3 | Standard | 0 |
| 2 | CAN1 | Standard | mask/value | 0 | 0 | | | CAN4 | Standard | 0 |
| 3 | CAN1 | Extended | mask/value | 0 | 0 | | | CAN2 | Extended | 0 |
| 4 | CAN1 | Extended | mask/value | 0 | 0 | | | CAN3 | Extended | 0 |
| 5 | CAN1 | Extended | mask/value | 0 | 0 | | | CAN4 | Extended | 0 |
| 6 | CAN2 | Standard | mask/value | 0 | 0 | | | CAN1 | Standard | 0 |
| 7 | CAN2 | Standard | mask/value | 0 | 0 | | | CAN3 | Standard | 0 |
| 8 | CAN2 | Standard | mask/value | 0 | 0 | | | CAN4 | Standard | 0 |
| 9 | CAN2 | Extended | mask/value | 0 | 0 | | | CAN1 | Extended | 0 |
| 10 | CAN2 | Extended | mask/value | 0 | 0 | | | CAN3 | Extended | 0 |
| 11 | CAN2 | Extended | mask/value | 0 | 0 | | | CAN4 | Extended | 0 |
| 12 | CAN3 | Standard | mask/value | 0 | 0 | | | CAN1 | Standard | 0 |
| 13 | CAN3 | Standard | mask/value | 0 | 0 | | | CAN2 | Standard | 0 |
| 14 | CAN3 | Standard | mask/value | 0 | 0 | | | CAN4 | Standard | 0 |
| 15 | CAN3 | Extended | mask/value | 0 | 0 | | | CAN1 | Extended | 0 |
| 16 | CAN3 | Extended | mask/value | 0 | 0 | | | CAN2 | Extended | 0 |
| 17 | CAN3 | Extended | mask/value | 0 | 0 | | | CAN4 | Extended | 0 |
| 18 | CAN4 | Standard | mask/value | 0 | 0 | | | CAN1 | Standard | 0 |
| 19 | CAN4 | Standard | mask/value | 0 | 0 | | | CAN2 | Standard | 0 |
| 21 | CAN4 | Standard | mask/value | 0 | 0 | | | CAN3 | Standard | 0 |
| 22 | CAN4 | Extended | mask/value | 0 | 0 | | | CAN1 | Extended | 0 |
| 23 | CAN4 | Extended | mask/value | 0 | 0 | | | CAN2 | Extended | 0 |
| 24 | CAN4 | Extended | mask/value | 0 | 0 | | | CAN3 | Extended | 0 |
| 25 | | | | | | | | | | |

**Fig. 15 Example 2 mapping table**

*To allow all messages to pass from every CAN port to every CAN port is also possible by using the range filter. Use range settings from value 0 to the value of the maximum ID (depending on wether 11 bit or 29 bit identifiers are used).*

## 6.3.4 Configuring the Multiplexing Table

The multiplexing table allows to divide CAN FD messages of up to 64 data bytes in Classic CAN messages of up to 8 data bytes.




| | Rx Channel | Rx Msg Format | Rx Identifier | Rx DLC | Position | Tx DLC | Tx Channel | Tx Msg Format | Tx Identifier |
|---|---|---|---|---|---|---|---|---|---|
| 0 | CAN3 | Standard | 0x100 | 64 | 0 | 8 | CAN1 | Standard | 0x200 |
| 1 | | | | | 8 | 8 | CAN1 | Standard | 0x201 |
| 2 | | | | | 16 | 8 | CAN1 | Standard | 0x202 |
| 3 | | | | | 24 | 8 | CAN1 | Standard | 0x203 |
| 4 | | | | | 32 | 8 | CAN1 | Standard | 0x204 |
| 5 | | | | | 40 | 8 | CAN1 | Standard | 0x205 |
| 6 | | | | | 48 | 8 | CAN1 | Standard | 0x206 |
| 7 | | | | | 56 | 8 | CAN1 | Standard | 0x207 |
| 8 | CAN3 | Standard | 0x101 | 16 | 0 | 4 | CAN2 | Standard | 0x280 |
| 9 | | | | | 4 | 4 | CAN2 | Standard | 0x281 |
| 10 | | | | | 8 | 4 | CAN2 | Standard | 0x282 |
| 11 | | | | | 12 | 4 | CAN2 | Standard | 0x283 |
| 12 | | | | | | | | | |

**Fig. 16 Example multiplexing table**

### Possible Entries

| Column | Possible entries |
|---|---|
| Rx Channel | CAN 3, CAN 4 |
| Rx Msg Format | Standard, Extended<br>Defines format in which messages are transmitted, standard (11 bit identifiers) or extended (29 bit identifier). |
| Rx Identifier | Identifier of CAN FD message to be divided |
| Rx DLC | Number of data bytes of CAN FD message to be divided |
| Position | Starting position within source message data stream |
| Tx DLC | Number of data bytes of Classic CAN message to be transmitted (up to 8 data bytes) |
| Tx Channel | CAN 1, CAN 2, CAN 3 (Classic CAN) and CAN 4 (Classic CAN) |
| Tx Msg Format | Standard, Extended<br>Defines format in which messages are received, standard (11 bit identifiers) or extended (29 bit identifier). |
| Tx Identifier | Identifier of received Classic CAN message |

► Configure the multiplexing table (handling see *Configuring the Mapping Table, p. 20*).

# 6.4 Further Settings

## 6.4.1 Reset to Factory Settings

It is possible to reset a connected device to factory settings.

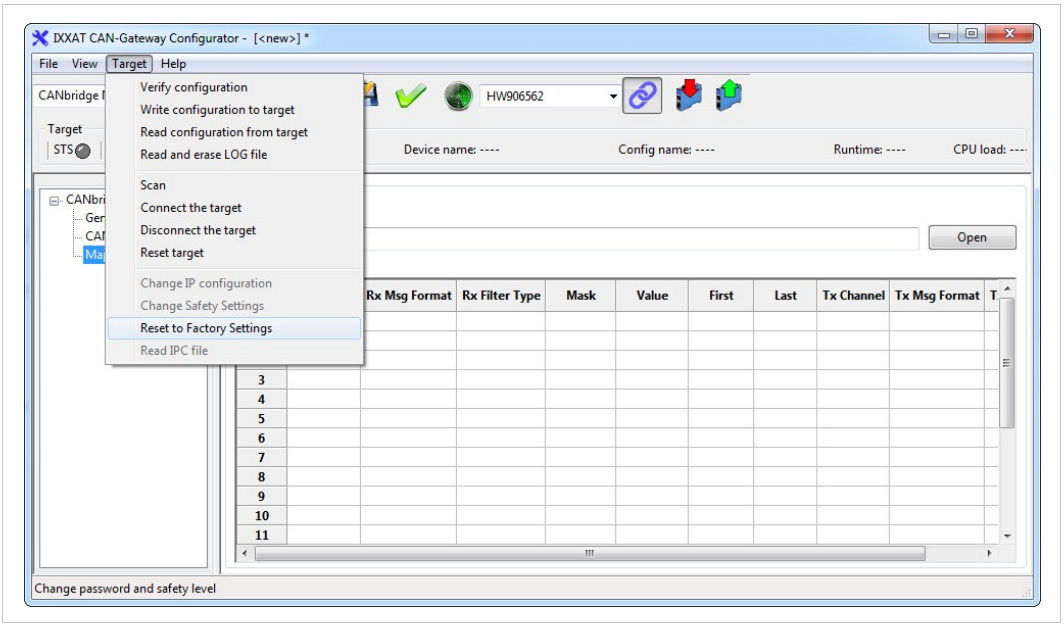
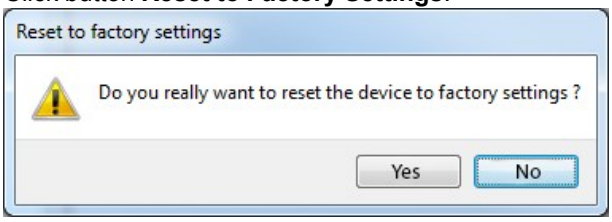



**Fig. 17 Menu Reset to Factory Settings**

► Make sure that device is connected via USB.

► Open menu **Target**.

► Click button **Reset to Factory Settings**.




► Click button **Yes** to confirm the reset.

## 6.4.2 Creating Log File

A log file can be used to detect errors, for example for the IXXAT support.

► Open menu **Target** and select **Read and erase LOG file**.

➡ .txt-file is created.

➡ Window **Save** is opened.

► Save file.

# 7 Operation

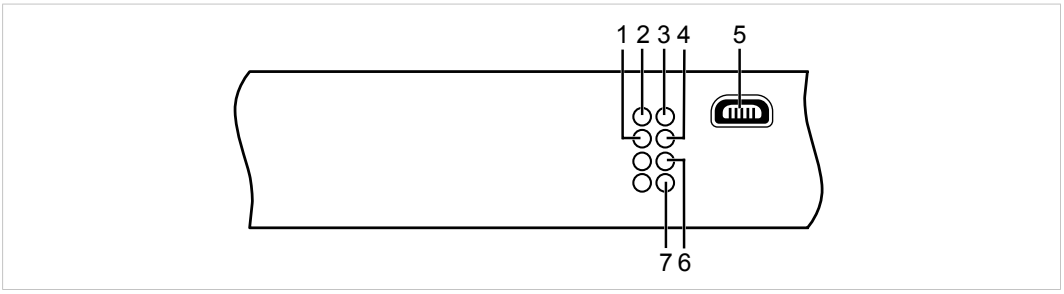



**Fig. 18 Ports and LEDs**

| | |
|---|---|
| 1 | Status LED |
| 2 | Power LED |
| 3 | CAN 1 LED |
| 4 | CAN 2 LED |
| 5 | Mini USB port |
| 6 | CAN 3 LED |
| 7 | CAN 4 LED |

## 7.1 Automatic Baud-rate Detection Sequence

> **!** Automatic baud-rate detection is exclusively possible in Star Coupler configuration.

During the automatic baudrate detection CAN 1 LED (**3**), CAN 2 LED (**4**), CAN 3 LED (**6**), CAN 4 LED (**7**) and Status LED (**1**) indicate the status.

| Status | LED CAN 1 | LED CAN 2/3/4 | Status LED |
|---|---|---|---|
| Automatic baudrate detection active on both channels | Orange flashing | Orange flashing | Green and orange flashing |
| Baudrate detected on CAN 1, baudrate detection on CAN 2 active | Off | Orange flashing | Green and orange flashing |
| Baudrate on CAN 2 detected or adopted from CAN 1, communication present | Green flashing | Green flashing | Green flashing |

## 7.2 Mini USB Port

Provided to connect the device for configuration.

# 7.3 Indicators

## 7.3.1 Status LED

Indicates the device status.

| LED state | Description | Comments |
|---|---|---|
| Off | Device not ready | No firmware, application firmware not started |
| Green flashing (1 Hz) | Application firmware started | Device in *operational* state |
| Red/green flashing | Configuration file error | Rewriting of configuration to device necessary |
| Green/orange flashing | Device in *configuring* state | - |
| Orange flashing (1 Hz) | Automatic baud-rate detection | Ongoing automatic baud-rate detection |
| Red flashing | Device error | Application signals a device error, no configuration or error in configuration |

## 7.3.2 Power LED

Indicates the status of the power supply.

| LED state | Description | Comments |
|---|---|---|
| Off | No power | Possible cause:<br>• Device not connected to power supply<br>• Fuse of device damaged<br>• Internal power supply damaged<br>• Power supply not sufficient |
| Green | Power | Device fully functional |

## 7.3.3 CAN LED

CAN LED 1 to 4 each indicates the status of the corresponding CAN interface.

| LED state | Description | Comments |
|---|---|---|
| Off | No CAN communication | No connection to CAN |
| Orange flashing | See *Automatic Baud-rate Detection Sequence, p. 26* | - |
| Green flashing | CAN communication | Baud-rate is detected, LED is triggered with each CAN message |
| Green | CAN communication | Baud-rate of second bus is detected, device in *operational* state, no messages on CAN bus |
| Red flashing | CAN communication, CAN controller in *error* state | CAN controller in *Error Warning* or *Error Passive* state, reception/transmission of CAN messages possible |
| Red | Bus off | CAN controller is in *Bus Off* state, no CAN communication possible |

# 8 Technical Data

| | |
|---|---|
| Dimensions | 114.5 x 99 x 22.5 mm |
| Weight | Approx. 150 g |
| Operating temperature | -40 °C to +85 °C |
| Storage temperature | -40 °C to +85 °C |
| Power supply | 9 V to 36 V DC |
| Current consumption | Typically 110 mA (at 24 V input voltage) |
| Housing material | Polyamide |
| Galvanic isolation | 1 kV for 1 sec |
| CAN transceiver | Texas Instruments SN65HVD251P |
| Max. number of bus nodes | 120 |
| CAN bus termination resistor | None |
| CAN baud-rates | Classic CAN: 5 to 1000 kBaud<br>CAN FD: 5 to 8000 kBaud |
| Protection class | IP20 |

# 9 Support/Return Hardware

Observe the following information in the support area on www.ixxat.com:

- information about products
- FAQ lists
- installation notes
- updated product versions
- updates

## 9.1 Support

► For problems or support with the product request support at www.ixxat.com/support.

► If required use support phone contacts on www.ixxat.com.

## 9.2 Return Hardware

► Fill in the form for warranty claims and repair on www.ixxat.com.

► Print out the Product Return Number (PRN resp. RMA).

► Pack product in a physically- and ESD-safe way, use original packaging if possible.

► Enclose PRN number.

► Observe further notes on www.ixxat.com.

► Return hardware.

# 10 Disposal

► Dispose of product according to national laws and regulations.

► Observe further notes about disposal of products on www.ixxat.com.

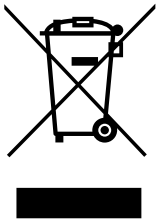
This page intentionally left blank

# A Regulatory Compliance

## A.1 EMC Compliance (CE)

The product is in compliance with the Electromagnetic Compatibility Directive. More information and the Declaration of Conformity is found at www.ixxat.com.

## A.2 Disposal and recycling



You must dispose of this product properly according to local laws and regulations. Because this product contains electronic components, it must be disposed of separately from household waste. When this product reaches its end of life, contact local authorities to learn about disposal and recycling options, or simply drop it off at your local HMS office or return it to HMS.

For more information, see www.hms-networks.com.

Box 4126
300 04 Halmstad, Sweden

info@hms.se
4.01.0331.20004 1.0.3277 / 2017-02-09 11:27