



User Manual UMAX140970
Version 1.01
Firmware 1.xx

USER MANUAL

Dual CAN FD to Ethernet Converter

P/N: AX140970

ACRONYMS

A	Ampere
ARP	Address Resolution Protocol
BRP	Bit Rate Pre-scaler
BSD	Berkeley Software Distribution
°C	Celsius (degree)
CAN	Controller Area Network
CE	Conformité Européenne (European Conformity)
CiA	CAN-in-Automation
CIDS	Channel ID Set
CG	Channel Group
EA	Axiomatic Electronic Assistant
EID	Extended Identifier
EFT	Electrical Fast Transient
ESD	Electrostatic Discharge
FD	Flexible Data-Rate
ft	feet
GPL	General Public License
HB	Horizontal Burn
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
ID	Identifier
IEC	International Electrotechnical Commission
IP	Internet Protocol or Ingress Protection (for housing)
ISO	International Organization for Standardization
kbit/s	Kilobits per second
L	Length (for size)
LAN	Local Area Network
LED	Light-Emitting Diode
m	meters
MAC	Media Access Control (address)
Mbit/s	Megabits per second
MCU	Microcontroller Unit
MDIX	Medium Dependent Interface Crossover (MDI-X)
MIT	Massachusetts Institute of Technology
mm	millimetres
PA	Polyamide
PHY	Physical Layer Transceiver (Ethernet chip)
P/N	Part Number
PWR	Power
PoE	Power Over Ethernet
RFC	Request for Comments
RoHS	Restriction of Hazardous Substances
RTOS	Real-Time Operating System
SJW	Synchronization Jump Width
SP	Service Pack or Sample Point
SSP	Software Support Package or Secondary Sample Point
TBD	To be Determined

TCP	Transmission Control Protocol
TDC	Transceiver Delay Compensation
UDP	User Datagram Protocol
UL	Underwriters Laboratories (safety organization)
V	Volt
VDC	Volt Direct Current
W	Width (for size)
WAN	Wide Area Network

TABLE OF CONTENTS

1	INTRODUCTION	6
2	CONVERTER DESCRIPTION.....	7
2.1	Hardware Block Diagram.....	7
2.2	LED Indicators.....	8
2.3	Logical Structure.....	8
2.3.1	Communication Device	9
2.3.1.1	UDP Protocol	10
2.3.1.2	TCP Protocol.....	11
2.3.2	Interconnection Matrix.....	11
2.3.3	Web Server	12
2.3.4	Network Discovery	12
3	CONVERTER CONFIGURATION	13
3.1	Changing Configuration Parameters	14
3.2	Ethernet Configuration.....	16
3.3	CAN Common	17
3.4	Internal Connections.....	18
3.4.1	CAN Frame Routing.....	18
3.5	CAN Port Configuration	19
3.5.1	CAN FD Bit-Timing Recommendations	21
3.5.2	CAN ID Range Filters.....	22
3.5.3	CAN ID Mask Filters.....	23
4	CONVERTER DIAGNOSTICS.....	26
4.1	Health Status.....	27
4.2	Converter Rebooting	27
5	FIRMWARE UPDATE.....	29
5.1	Uploading the New Firmware	29
5.2	Applying the New Firmware.....	30
6	CONVERTER DEPLOYMENT.....	32
6.1	CAN Network Bridging.....	32
6.1.1	Hardware Setup	33
6.1.2	Converter Configuration	34
6.1.2.1	CAN Port Routing Addresses.....	34
6.1.2.2	Server Configuration	34
6.1.2.3	Client Configuration.....	35
6.1.2.4	CAN Port Configuration.....	36
6.1.2.5	Internet Connection.....	36
6.1.2.6	Connecting Several Converters Together	36
7	CONVERTER DISCOVERY	37
8	TECHNICAL SPECIFICATIONS.....	38
8.1	Power Supply	38
8.1.1	Input.....	38
8.1.2	Output	38
8.2	Ethernet Port	38
8.2.1	Ethernet Connector	39
8.3	CAN FD Ports.....	40
8.3.1	CAN Connector 1	40
8.3.2	CAN Connector 2	40
8.4	General Specifications.....	41
8.5	Dimensional Drawing.....	42

9 THIRD PARTY SOFTWARE LICENSE NOTICES43
10 VERSION HISTORY.....47

1 INTRODUCTION

The following user manual describes architecture, functionality, and configuration parameters of the Dual CAN FD to Ethernet Converter. It also contains technical specifications for the converter.

The user should check whether the application firmware installed on the converter is covered by this user manual. It can be done using any web browser connected to the converted over the Ethernet connection, see [Converter Configuration](#) section for more details.

The user manual is valid for application firmware with the same major version number as the user manual. For example, this user manual is valid for any application firmware version 1.xx. Updates specific to the user manual are done by adding letters: A, B, ..., Z to the user manual version number.

2 CONVERTER DESCRIPTION

The Dual CAN FD to Ethernet Converter is a simple device converting Classical CAN or CAN FD frames into UDP or TCP IP datagrams and sending them over the Ethernet network. The device can also convert UDP or TCP datagrams into Classical CAN or CAN FD frames and transmit them through any of the two CAN FD ports.

The converter has two galvanically isolated independent CAN FD ports and one Ethernet port. The CAN FD ports support High-Speed Classical CAN with data rate up to 1Mbit/s or CAN FD with data rate up to 8 Mbit/s. The Fast Ethernet port provides 100Mbit/s data rate.

All types of Classical CAN and CAN FD frames are supported. CAN FD ports can operate either in ISO or in Bosch frame format in the flexible data rate (FD) mode.

The power can be passed through to the CAN port connector. Protection is provided.

The converter contains a web server to setup configuration parameters and monitor the internal state of the converter using a web browser. The user can also update the converter firmware using the web browser.

A simple command-line `AxioDisc.exe` Windows application is provided to locate a converter on the LAN.

To ensure low latency in processing CAN and Ethernet messages, the converter software runs under control of a real-time operating system.

The converter is designed to work on off-road machinery or in a harsh industrial environment with power transients, high humidity, and vibrations.

2.1 Hardware Block Diagram

The converter hardware block diagram is presented in Figure 1.

The converter is powered from the Ethernet connector using dedicated power lines *PWR+* and *PWR-*. The power input is protected from reverse polarity and transients.

The power from the Ethernet connector can be delivered to the CAN connector through a *Power Feed-Through* switch controlled by a microcontroller. The power output *PWR_OUT* has overcurrent, short to battery/ground protection.

The *Ethernet PHY & Magnetics* interface block provides Fast Ethernet connection, whereas *Isolated CAN1* and *Isolated CAN2* blocks provide two independent, individually galvanically isolated, CAN FD physical ports.

A powerful ARM Cortex-M7 based STM32H7 series microcontroller runs IP protocol stack and all Ethernet to CAN conversion logic.

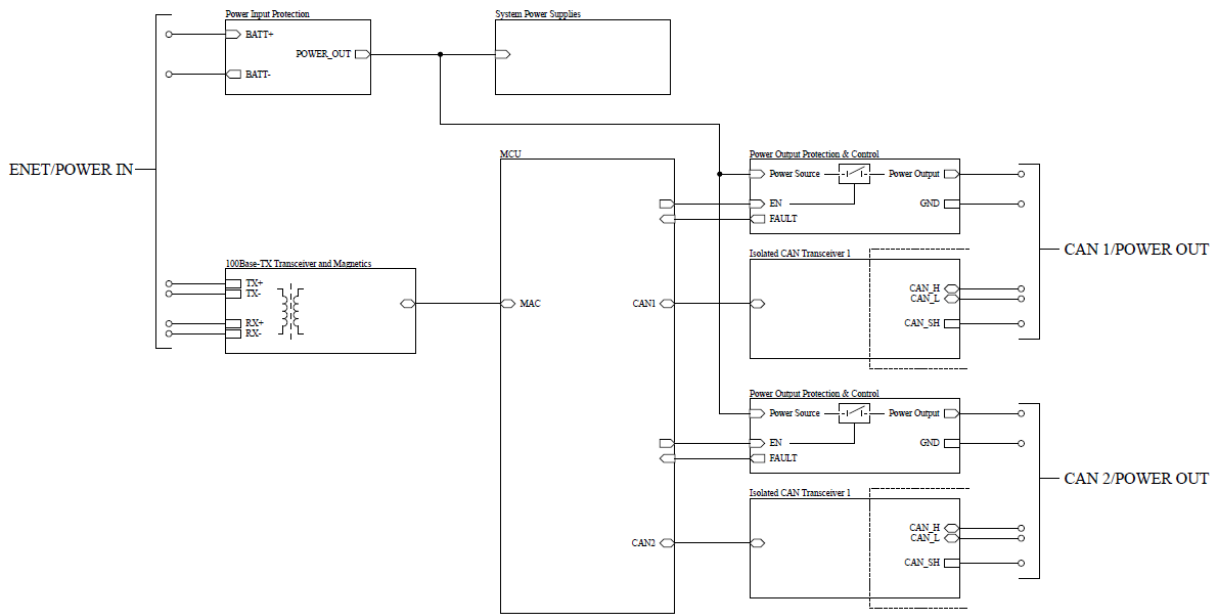


Figure 1. The Converter Hardware Block Diagram

2.2 LED Indicators

There are three green LED indicators on the front panel of the converter. The “POWER”¹ indicator shows whether the unit is powered.

The two Ethernet LED indicators are hardwired to the PHY chip and show the transmission speed “10/100” and the link/activity status “LINK/ACT”, see Table 1:

Table 1. Converter LED Indicators

Name	State	Description
POWER ¹	Off	The converter is not powered
	On	The converter is powered
10/100	Off	Ethernet speed is 10 Mbit/s
	On	Ethernet speed is 100 Mbit/s
LINK/ACT	Off	Ethernet link is down
	On	Ethernet link is up
	Blinking	Ethernet link is up and active

¹“STATUS” on the first version of the unit overlay, see Figure 32.

2.3 Logical Structure

The Dual CAN FD to Ethernet Converter is internally organized as a system of interconnected independent modules. It includes the following internal modules: the *Communication Device*, *Interconnection Matrix*, *Web Server*, and *Network Discovery* module see Figure 2.

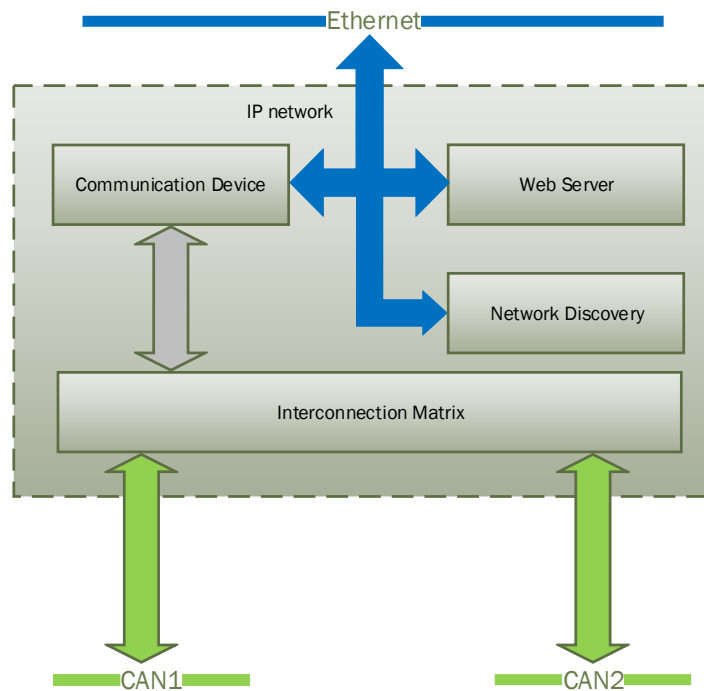


Figure 2. Converter Firmware Architecture

The *Communication Device* is responsible for the protocol conversion between CAN and Ethernet networks.

The *Interconnection Matrix* connects CAN ports with the *Communication Device*, provides CAN-Ethernet mapping and internal connections of the converter CAN ports with each other.

The *Web Server* provides the user interface for device configuration and firmware updates.

The *Network Discovery* module supports a proprietary network discovery protocol to locate the converter IP (internet protocol) address on the LAN, if necessary.

The *Communication Device*, *Web Server*, and the *Network Discovery* module are accessed using IP network interface. They share the same IP address, mask and default gateway settings assigned to the *Communication Device* but use different communication ports and protocols.

2.3.1 Communication Device

The *Communication Device* utilizes a proprietary communication protocol to communicate CAN messages and other auxiliary information over the Ethernet. It supports a client/server communication model. In this model, the *Communication Device* has a primary server role, allowing external clients to establish independent connections with the device.

In addition to the server role, the device can also act as a client, if the *Auto Connect to Remote* configuration parameter is set to *Yes*. In this case, the device will try to establish a connection with a remote server specified by the customer.

The total number of remote connections is limited to 10. When the CAN traffic is extremely high, there could be a situation when this number needs to be reduced, otherwise the connections will become unstable due to limited internal resources of the microcontroller, which are dynamically allocated between open connections.

The *Communication Device* can use either UDP or TCP internet protocol (IP), depending on the value of the *Device Port Type* configuration parameter.

2.3.1.1 UDP Protocol

The UDP protocol is set by default. Since it is a connectionless protocol, one data socket serves all device communication needs, see Figure 3.

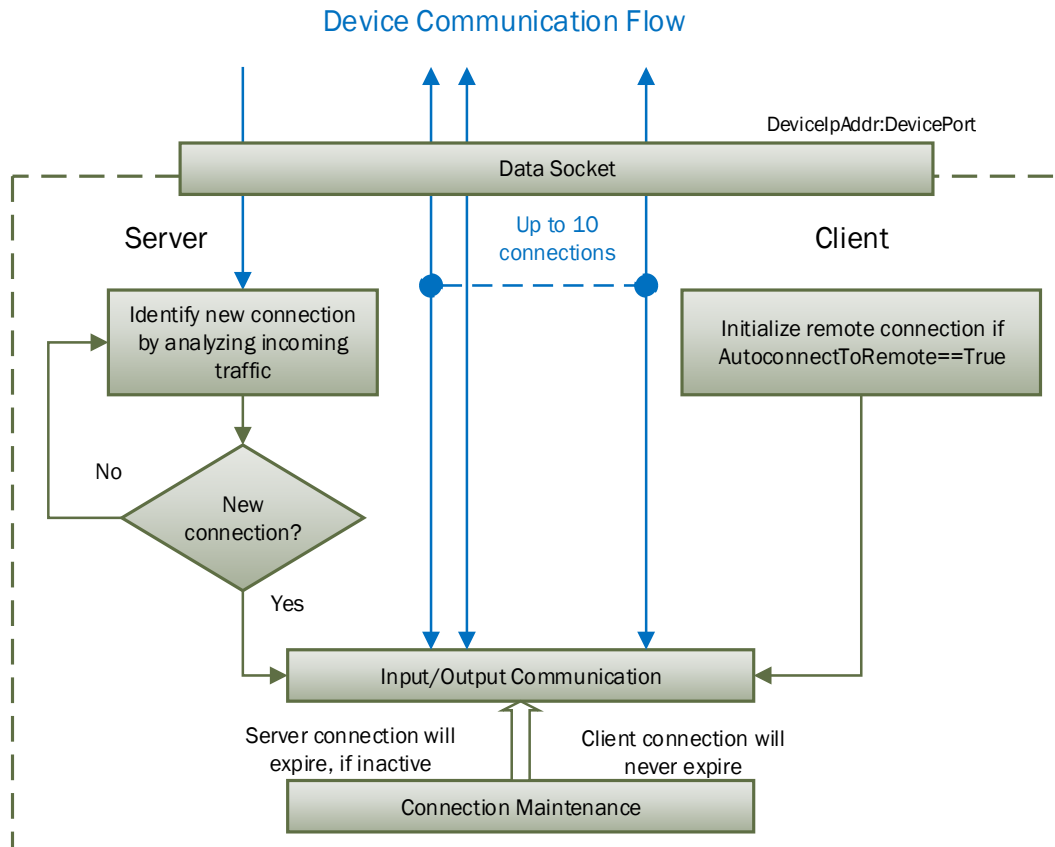


Figure 3. Communication Device. UDP Protocol

All connections with the device are virtual. On the server side, the device analyzes the incoming traffic to check for the new connections. Once a new IPAddress:Port combination is detected, the connection is established and the device starts sending CAN data with Heartbeat messages to the new node.

There are no restrictions on the IP address and port for the incoming connections.

If a client-side is activated by the *Autoconnect to Remote* configuration parameter, the device will automatically start sending CAN data with Heartbeat messages to the remote node on start-up.

To ensure that the device does not send data to not functioning (dead) or disconnected nodes, the server side connections will expire in 10 sec of inactivity, when no valid data is received from the remote node. The client-side connection will never expire.

2.3.1.2 TCP Protocol

When TCP protocol is used, the *Communication Device* opens an individual data socket for each device connection, see: Figure 4.

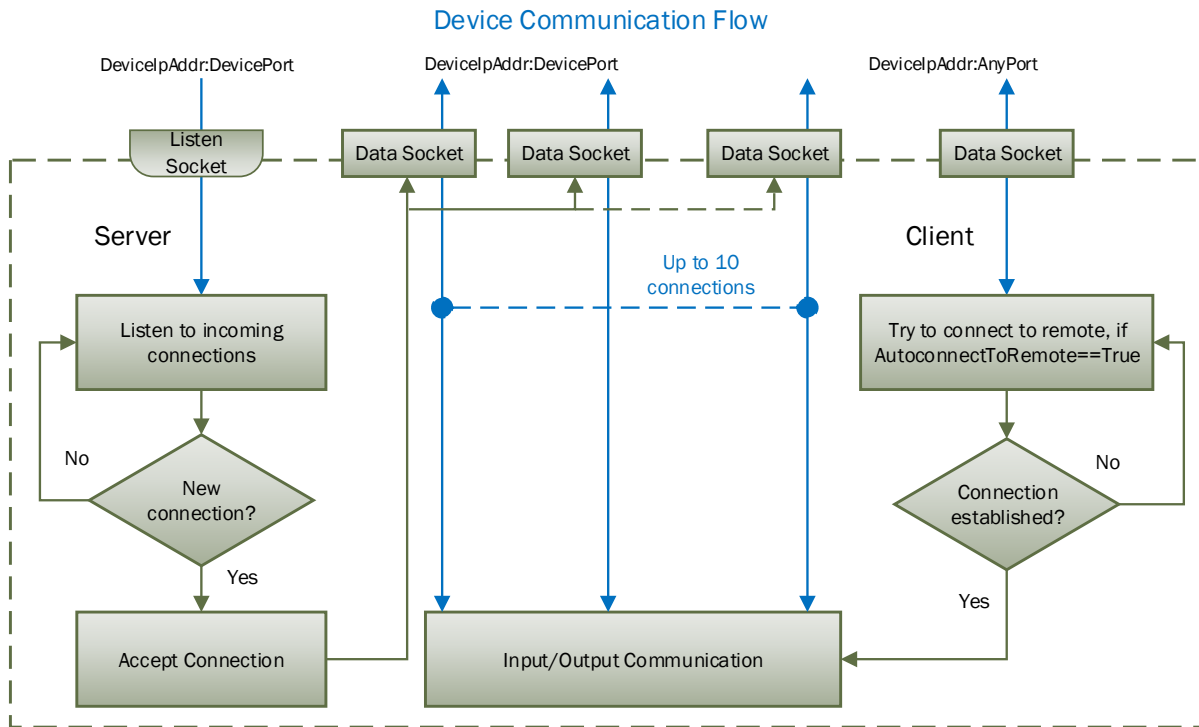


Figure 4. Communication Device. TCP Protocol

The server side opens a listening socket for incoming connections. Once a connection is accepted, a new data socket is created to handle input/output communication with the remote node. There are no restrictions on the IP address and port for the incoming connections, similar to the UDP mode.

On the client side, if *Auto Connect to Remote* is set to *Yes*, a data socket is created for connection with the remote node. A random free port number is assigned to the socket. If the connection drops, the device will try to automatically reconnect with the node to maintain the client connection.

2.3.2 Interconnection Matrix

The *Interconnection Matrix* is responsible for routing CAN frames between the *Communication Device* and CAN ports and between individual CAN ports inside the converter.

For routing CAN frames between the *Communication Device* and CAN ports, the *Interconnection Matrix* uses a special routing address. This address is assigned to CAN ports during CAN-Ethernet mapping and is transmitted over the IP network in outgoing CAN frames. When address of the port and address in the incoming CAN frame

matches, the frame is received by the port, see [CAN Data Routing](#) for further information.

2.3.3 Web Server

The *Web Server* provides a user front-end interface with the converter. It runs a dynamic website that shows: the converter general information, configuration parameters, and the converter real-time diagnostics.

The user can also change configuration parameters and upload the new firmware through this website.

The web browser should support JavaScript.

2.3.4 Network Discovery

The *Network Discovery* module supports a proprietary Axiomatic discovery protocol. It allows to locate a converter with unknown IP address and/or web server port on a LAN using a simple Axiomatic discovery application `AxioDisc.exe`.

3 CONVERTER CONFIGURATION

The converter supports configuration over the internal website running on the device embedded web server. The default *Device IP Address* is *192.168.0.34* and the default *Web Server Port* is *80*. Please, make sure that there are no other devices on this IP address when connecting the converter for the first time to your LAN for configuration.

To connect to the device, run any web browser and point it to the *Device IP Address*. It is not necessary to specify the *Web Server Port* if the web server uses a standard port *80*.

After a successful connection, you will see the device home page, see: Figure 5.

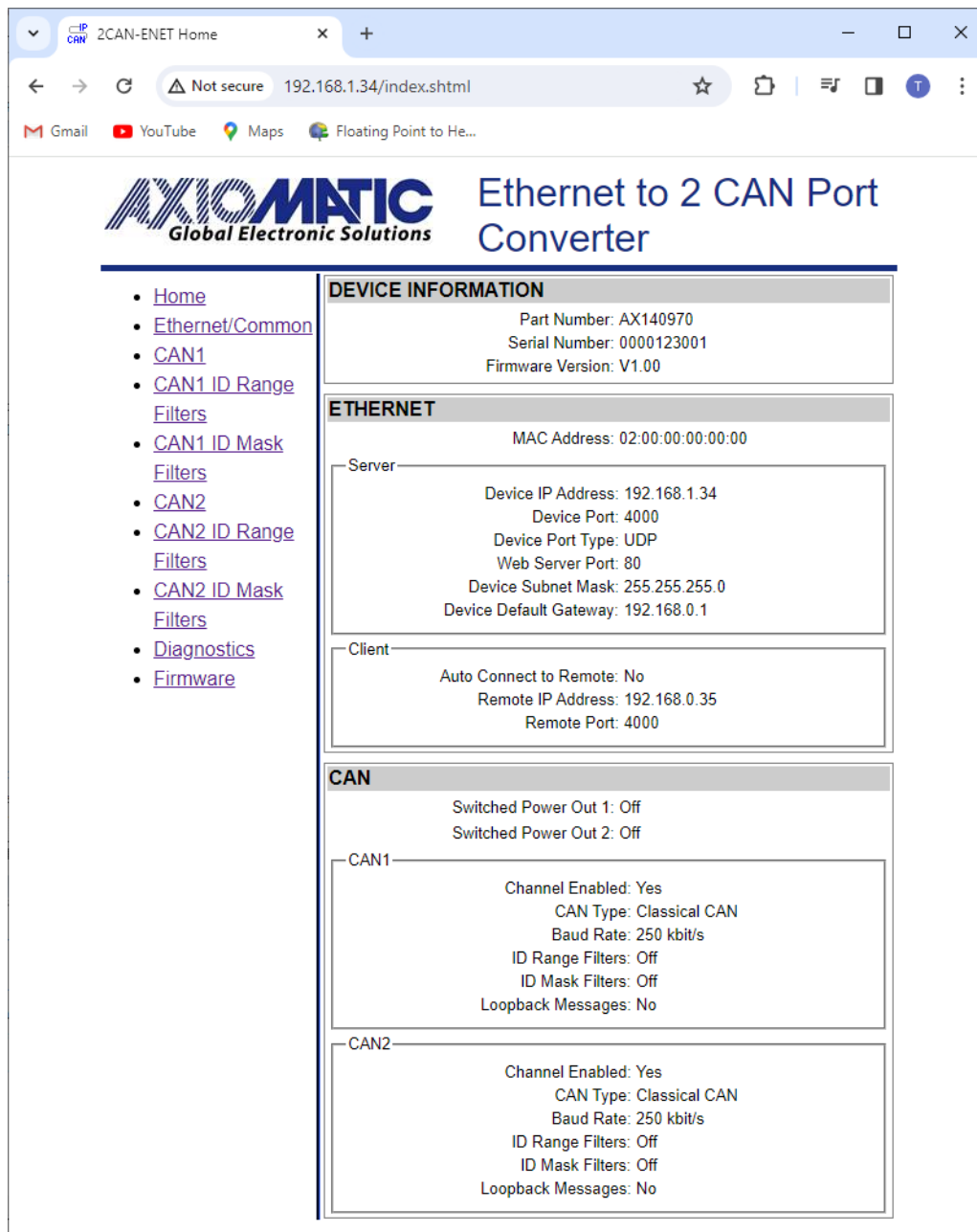


Figure 5. Converter Home Page¹

¹The Firmware Version number may be different from the firmware version described in the manual.

The home page shows the device information, including the converter part number, serial number, and firmware version. It also shows *Ethernet* and *CAN* configuration parameters including the status of CAN ID range and mask input filters.

You will need to allow the site to run JavaScript (this setting is default in the majority of web browsers). If JavaScript is disabled, the website will show a message asking to activate JavaScript at the top of the web page, see Figure 6.

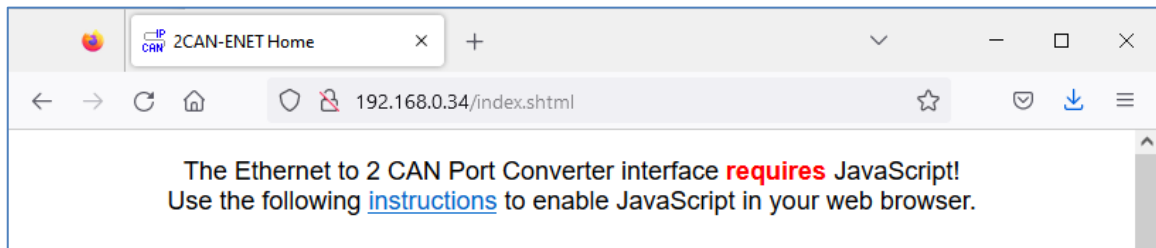


Figure 6. Enable JavaScript Prompt

The *Ethernet* configuration parameters are combined into *Server* and *Client* groups for convenience.

The *Ethernet* and *CAN* configuration parameters have tooltips clarifying their meaning, see Figure 7.

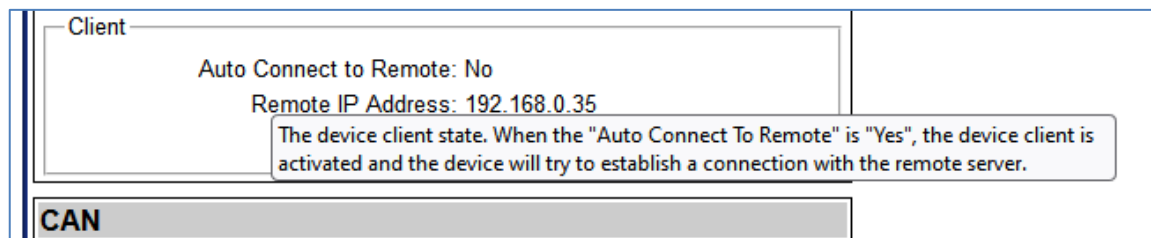


Figure 7. A Tooltip for the "Auto Connect to Remote" Configuration Parameter

3.1 Changing Configuration Parameters

All configuration parameters except CAN port settings can be changed through the *Ethernet/Common* web page. Main CAN settings can be changed through *CAN1(2)* web pages. CAN range and mask filters have their own web pages: *CAN1(2) ID Range Filters* and *CAN1(2) ID Mask Filters*.

The configuration pages can be reached by clicking on their links on the left side of the website, see: Figure 8.

Each configuration web page has fields to enter values of the configuration parameters and three buttons: *Save Settings*, *Discard Settings* and *Set Defaults*, see Figure 8.

The *Save Settings* button will save configuration parameters to non-volatile memory and apply the new settings. The appropriate converter subsystems will be restarted without rebooting of the whole converter. If rebooting is necessary, it is available from the *Diagnostics* page.

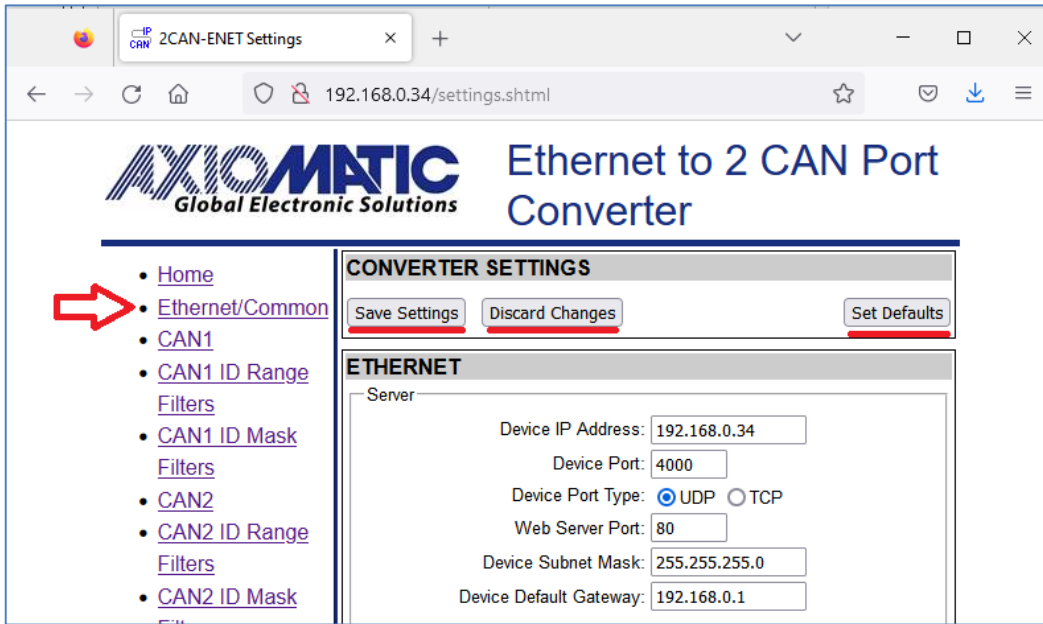


Figure 8. Ethernet/Common Settings Configuration Webpage

The *Discard Settings* button will bring back the original converter settings before editing. In case the user leaves the page without saving, all changes will be also discarded.

The *Set Defaults* button will load the default values of the configuration parameters into the data fields on the configuration page.

The configuration parameters have tooltips for the user's convenience.

When the user presses *Save Settings* button, the web page runs a script to check the validity of the new configuration parameters before uploading them to the web server. For example, the following alert message will be displayed if the user enters the same value for the *Device Port* and the *Web Server Port*, see Figure 9.

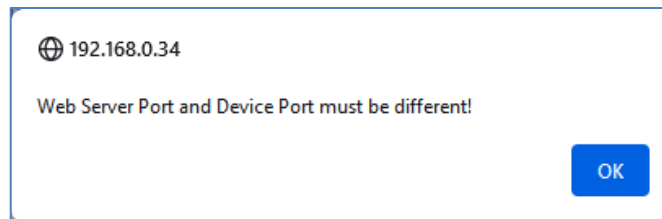


Figure 9. Settings Alert. Error in Configuration Parameters

The website messages should be enabled (not suppressed) in the browser to see this and other feedback messages.

After pressing the *Save Settings* button and saving the configuration parameters, the converter replies with a confirmation message showing the result of the saving operation. For example, if the user has successfully changed an *Ethernet/Common* configuration parameter, the following message will appear, see Figure 10.

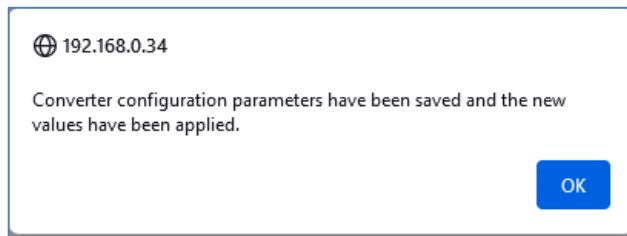


Figure 10. Settings Alert. Configuration Parameters have been Changed Successfully

The new configuration parameters are applied immediately after saving. The converter performs all necessary internal reconfigurations and resets on the fly.

3.2 Ethernet Configuration

All *Ethernet* configuration parameters can be changed through the *Ethernet/Common* webpage, except the *MAC Address*, which is programmed at the factory.

The user-changeable *Ethernet* configuration parameters are presented in Table 2. They statically assign the converter IP network configuration.

The user should avoid using special IP addresses (broadcast, multicast, loopback, etc.) when configuring the *Device IP Address* since this can lead to a permanent loss of communication with the embedded web server.

The *Device Subnet Mask* should have all ones on the left side and all zeros on the right side when converted to the binary presentation as per the Internet document RFC 1878.

When the device client is activated by setting the *Auto Connect to Remote* configuration parameter to *Yes*, the converter will still act as a server accepting connections on the *Device Port* from other clients. This adds versatility to the converter configurations since the same converter can be used together with both: client and server communication nodes.

Table 2. Ethernet Configuration Parameters

Configuration Parameter	Default Value	Range	Description
<i>Device IP Address</i> ¹	192.168.0.34	Any IP address	The converter IP address
<i>Device Port</i>	4000	Any port value except the <i>Web Server Port</i> and the <i>Discovery Protocol Port</i>	The device server port. The device is listening to this port for incoming connections. The Discovery Protocol Port (35100) and the <i>Web Server Port</i> should not be used
<i>Device Port Type</i>	UDP	{UDP, TCP}	Type of the IP protocol used by the device. The device server and client use the same IP protocol
<i>Web Server Port</i>	80	Any port value except the <i>Device Port</i> and the <i>Discovery Protocol Port</i>	The communication port of the embedded web server

Configuration Parameter	Default Value	Range	Description
<i>Device Subnet Mask</i>	255.255.255.0	Any IP address	The converter subnet mask
<i>Device Default Gateway</i> ¹	192.168.0.1	Any IP address	The converter default gateway
<i>Auto Connect to Remote</i>	No	{No, Yes}	The device client state. When the <i>Auto Connect to Remote</i> is <i>Yes</i> , the device client is activated, and the device will try to establish a connection with the remote server
<i>Remote IP Address</i>	192.168.0.35	Any IP address	The remote server IP address. Used by the device client when the <i>Auto Connect to Remote</i> is <i>Yes</i> .
<i>Remote Port</i>	4000	Any port value	The remote server port. Used by the device client when the <i>Auto Connect to Remote</i> is <i>Yes</i> .

¹Common parameter shared with all internal converter modules including the embedded web server. Must be assigned by the network administrator.

The *Remote IP Address* and *Remote Port* are disabled when *Auto Connect to Remote* is set to *No*.

If the user changed the *Device IP Address* or the *Web Server Port*, the website will be automatically re-loaded at the new location, see Figure 11.

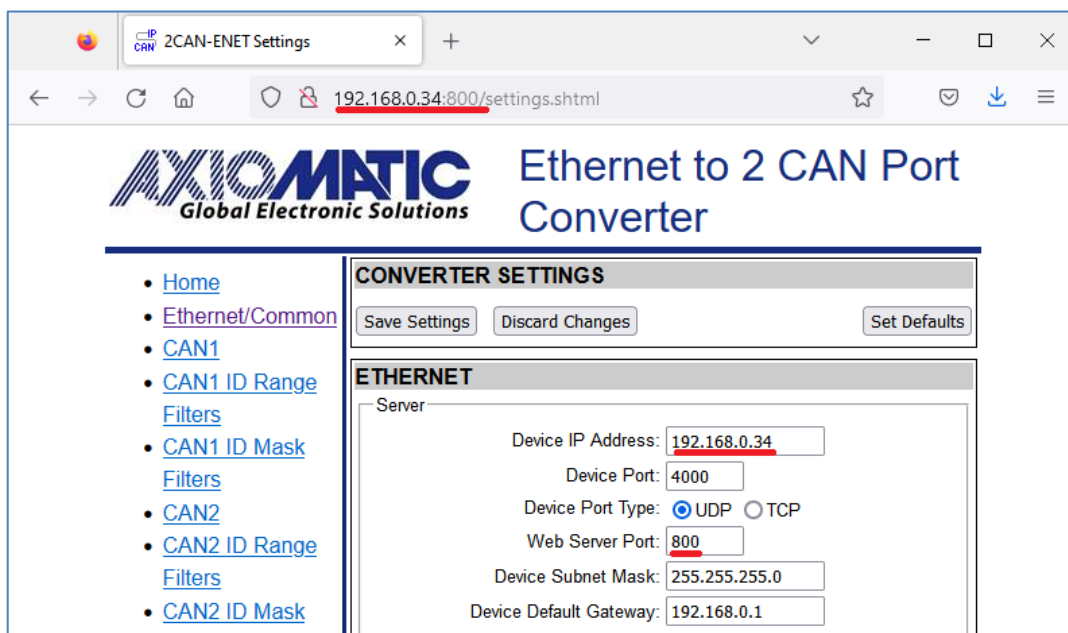


Figure 11. Website Automatic Relocation

3.3 CAN Common

The *CAN Common* settings control the *Power Feed-Through* switch, see Figure 1. The switch can be turned *On* or *Off* from the *Ethernet/Common* webpage.

3.4 Internal Connections

The *Internal Connections* settings control the *Interconnection Matrix* that provides CAN-Ethernet mapping and internal connections between CAN ports.

The CAN-Ethernet mapping defines a routing address for every CAN port in the IP device. It is done by assigning *Group* and *Channel* parameters to each CAN port, see Figure 12.

The screenshot shows a window titled "INTERNAL CONNECTIONS" with a sub-section "CAN-Ethernet Mapping". Underneath, there is a table-like structure with the heading "Ethernet". It has two columns: "Group" and "Channel". For "CAN1", the "Group" is 0 and the "Channel" is 1. For "CAN2", the "Group" is 0 and the "Channel" is 2.

Ethernet	
Group	Channel
CAN1 0	1
CAN2 0	2

Figure 12. CAN-Ethernet Mapping

The internal connection between CAN ports is controlled by the *Connect CAN1 with CAN2* configuration parameter that virtually connects two CAN ports.

The Internal Connections configuration parameters are presented in Table 3.

Table 3. Internal Connections Configuration Parameters

Configuration Parameter	Default Value	Range	Description
<i>Group</i> ¹	0 – CAN1 and CAN2	0...255	<i>CAN Port Channel Group</i> . This parameter is the first part of the <i>CAN Port Address</i> ²
<i>Channel</i> ¹	1 – CAN1 2 – CAN2	1...32	<i>CAN Port Channel Number</i> . This parameter is converted to the <i>CAN Port Channel ID Set</i> ² , which is the second part of the <i>CAN Port Address</i> ²
<i>Connect CAN1 with CAN2</i>	No	{No, Yes}	Internally virtually connects CAN1 and CAN2 ports. When the ports are virtually connected, CAN frames received by one port are sent by the other port

¹ Independently configured for CAN1 and CAN2 ports.

² Described in the [CAN Data Routing](#) sub-section.

3.4.1 CAN Frame Routing

The converter uses a proprietary Axiomatic communication protocol to convert CAN frames into IP datagrams and communicate them over the Ethernet port [1]. This protocol supports CAN frame routing to different CAN ports on the same IP device. This is achieved by using a special routing address that uniquely identifies CAN ports on the device. To allow simultaneously sending CAN frames to different CAN ports, this address is designed the way that it can include a set of individual CAN port IDs.

The CAN routing address is presented by two parameters: *Channel Group (CG)* and *Channel ID Set (CIDS)*: (CG, CIDS). The *Channel Group* defines a set of channels in the *Channel ID Set*. The *Channel ID Set* selects channels, or CAN ports, that will be addressed from this *Channel Group*.

Each Channel ID is presented as a bit (flag) at the channel number position in the *Channel ID Set* field. The *Channel ID Set* can contain up to 32 Channel IDs that are ORed together in the 4-byte *Channel ID Set* field. This allows sending a CAN frame simultaneously to up to 32 CAN ports.

The routing address of the converter CAN ports is defined by the *Group* and *Channel* configuration parameters, see Table 3. They are converted into the *CAN Port Address*. This address is included as *CAN Frame Address* in CAN FD frames sent by the converter through the Ethernet port.

When the *CAN Frame Address* matches the address of a CAN port on another device, the port receives the transmitted CAN frame. This way, CAN ports on different IP devices can be virtually connected together.

Two addresses are considered to match if their *Channel Groups* are equal and there is at least one Channel ID in the *Channel ID Sets* that is selected (set to 1) in both addresses.

The converter CAN ports transmit CAN data with only one CAN port Channel ID in the *CAN Frame Address*. However, it is possible to transmit CAN data with multiple Channel IDs in the *Channel ID Set* field of the *CAN Frame Address* if it is necessary to broadcast CAN data to multiple CAN ports.

It is assumed that Axiomatic CAN converters and PC software supporting the old version of Axiomatic proprietary protocol, that does not have CAN data routing, are using *Group* = 0, *Channel* = 1 combination for their *CAN Port Address*. This is a default setting for CAN1 port.

3.5 CAN Port Configuration

CAN port configuration parameters can be changed through the main webpage *CAN1(2)* and filter webpages *CAN1(2) ID Range Filters*, and *CAN1(2) ID Mask Filters* independently for each CAN channel. If all filters on all filter webpages are disabled, all input CAN messages will be output on the Ethernet network.

The main CAN configuration parameters are available through the *CAN1(2)* webpage, see Table 4.

Table 4. Main CAN Configuration Parameters

Configuration Parameter	Default Value	Range	Description
<i>Channel Enabled</i>	On	{Off, On}	Enables the CAN port. When <i>Channel Enabled</i> is <i>Off</i> , the CAN transceiver is in Standby mode slightly reducing energy consumption of the converter. Communication between other nodes on the CAN bus is not affected when the port is disabled.
<i>CAN Type</i>	Classical CAN	{Classical CAN, ISO CAN FD, Non-ISO CAN FD}	Type of the CAN port. <ul style="list-style-type: none"> Classical CAN refers to High-Speed CAN with up to 1 Mbit/s bit rate and

Configuration Parameter	Default Value	Range	Description
			<p>up to 8-byte data payload per frame, ISO11898-1 (Bosch CAN 2.0A and B).</p> <ul style="list-style-type: none"> • ISO-CAN FD refers to CAN FD frame format according to ISO11898-1. • Non-ISO CAN FD refers to CAN FD frame format defined in Bosch CAN FD Specification V1.0.
<i>Loopback Messages</i>	No	{No, Yes}	Specifies, whether the messages received over the Ethernet and transmitted on the CAN bus, are sent back to the Ethernet. Setting this value to Yes can create an eternal loop when the same messages are bounced between two or more converters. Use with caution.
Classical CAN			
<i>Baud Rate</i>	250 kbit/s	{1000, 800, 666.6(6), 500, 250, 125, 100, 83.3(3), 50, 20, 10} ¹	The Classical CAN baud rate. Not used in CAN FD mode.
CAN FD			
<i>CAN FD Baud Rate</i>	J1939_17 500_5000	{Custom, J2284_4 500_2000, J1939_17 500_5000, J2284_5 500_5000}	<p>Custom CAN FD baud rate or a standard baud rate from the following list:</p> <ul style="list-style-type: none"> • J2284-4:2016-06, 500 kbit/s / 2 Mbit/s • J1939-17:2020-12, 500 kbit/s / 2 Mbit/s • J2284-5:2016-09, 500 kbit/s / 5 Mbit/s
Nominal (Arbitration) Phase			
<i>BRP</i>	2 ³	1...512	<i>Bit Rate Pre-scaler.</i> The value by which the CAN clock frequency, equal to 80MHz, is divided for generating the <i>CAN Time Quantum</i>
<i>Tseg1²</i>	127 ³	1...256	<i>Time Segment 1.</i> Time segment before the sample point minus <i>Synchronization Segment</i> . The <i>Synchronization Segment</i> is always equal to 1
<i>Tseg2²</i>	32 ³	1...128	<i>Time Segment 2.</i> Time segment after the sample point
<i>SJW²</i>	32 ³	1...128	<i>Synchronization Jump Width</i>
Data Phase			
<i>BRP²</i>	2 ³	1...32	<i>Bit Rate Pre-scaler.</i> Bit Rate Pre-scaler. The value by which the CAN clock frequency, equal to 80MHz, is divided for generating the <i>tq</i> – <i>CAN Time Quantum</i>

Configuration Parameter	Default Value	Range	Description
T_{seg1}^2	15 ³	1...32	<i>Time Segment 1.</i> Time segment before the sample point minus <i>Synchronization Segment</i> . The <i>Synchronization Segment</i> is always equal to 1
T_{seg2}^2	4 ³	1...16	<i>Time Segment 2.</i> Time segment after the sample point
SJW^2	4 ³	1...16	<i>Synchronization Jump Width</i>
<i>TxD Delay Compensation</i>	Yes ³	{No, Yes}	<i>Transceiver Delay Compensation (TDC).</i>
<i>TxD Delay Compensation Offset</i>	30 ^{3,4}	0...127	<i>Transceiver Delay Compensation Offset.</i> Defines a position of the <i>Secondary Sample Point (SSP)</i> in <i>Data Phase</i> when the <i>Transceiver Delay Compensation</i> is <i>Yes</i> .

¹ 666.6(6) and 83.3(3) kbit/s are set as 666.667 and 83.333 kbit/s, respectively, in the drop-down menu.

² Set in *CAN Time Quanta, tq*.

³ Set for *Custom Baud Rate*, default value: 250 kbit/s / 2Mbit/s.

⁴ Set in *Minimum Time Quanta, mtq*, which is one period of the CAN clock frequency, equal to 80MHz.

For the Classical CAN, the customer can pick a baud rate from a list of baud rates using the *Baud Rate* configuration parameter.

For the CAN FD, the customer can either use one of the three standard baud rate configurations or use a custom CAN peripheral controller configuration using the *CAN FD Baud Rate* configuration parameter.

In the *Custom CAN* peripheral controller configuration, the customer can independently set bit-timing in nominal (arbitration) and data phases and also apply the transceiver delay compensation (TDC) in the data phase.

The website automatically calculates and displays the *tq* – *Time Quantum, Baud Rate*, and the *Sample Point (SP)* position for both: arbitration and data phases for the selected *CAN FD Baud Rate* configuration. It also shows the *mtq* – *Minimum Time Quantum* and the *SSP* – *Secondary Sample Point* position in the data phase.

Since CAN FD is very critical to CAN bit-timing settings, it is highly recommended to observe the following bit-timing recommendations and use the same settings for all CAN FD nodes on the CAN bus.

3.5.1 CAN FD Bit-Timing Recommendations

The CiA 601-3, *CAN FD Node and System Design Recommendations*, Technical Report, gives the following recommendations:

1. Use BRP_A and BRP_D as low as possible (A and D indexes refer to arbitration and data phases, respectively).
2. Choose SJW_A as large as possible.

3. Choose the highest available CAN clock frequency. The converter CAN clock frequency is already set to the highest recommended 80MHz.
4. Set $BRP_A = BRP_D$, (when not possible set $BRP_A \leq BRP_D$).
5. Configure all CAN nodes to have the same arbitration phase SP and the same data phase SP.
6. Choose SJW_D as large as possible.
7. Enable TDC for data bit rates ≥ 1 Mbit/s.

This document also recommends using the CAN clock frequencies from the following list: {20MHz, 40MHz, 80MHz} to facilitate good interoperability.

When the transceiver delay compensation (TDC) is enabled, the CiA 601-3 recommends setting SSP to 1 mtq earlier inside the received bit compared to the data phase SP inside the transmitted bit.

3.5.2 CAN ID Range Filters

The CAN ID range filters are set through the *CAN1(2) ID Range Filters* configuration webpage for each CAN port, see Figure 13.

The user can independently configure five CAN ID range filters: *Filter 1, Filter 2, ..., Filter 5*.

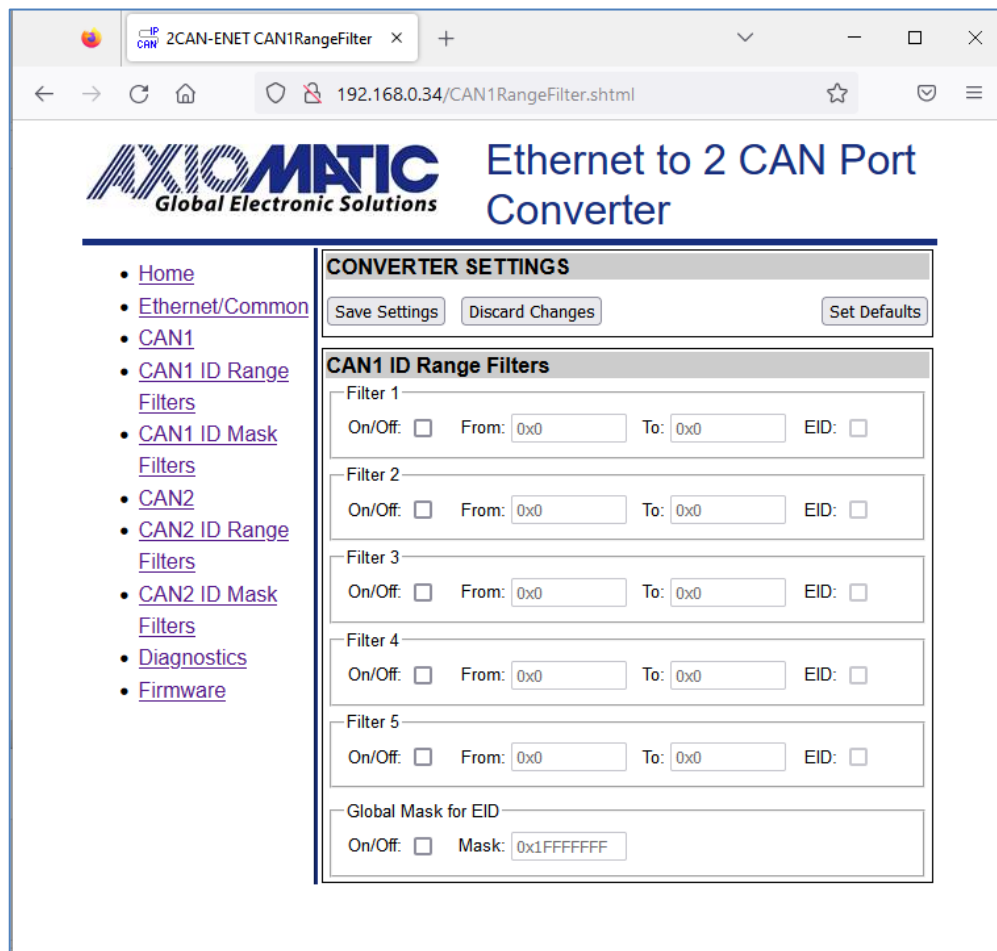


Figure 13. Converter CAN ID Range Filters Page

Once the filter is activated by checking the *On/Off* box, the CAN input frame will pass through to the Ethernet network only if their CAN ID is within the range specified by *From* and *To* configuration parameters.

If $ID_{CAN} \in [ID_{From}; ID_{To}]$, then the CAN frame is accepted.
(1)

Where: ID_{CAN} – CAN frame ID,
 ID_{From} – *From* configuration parameter,
 ID_{To} – *To* configuration parameter.

The *EID* box (*Extended ID* box) defines whether the CAN frame ID is regular or extended.

The user can activate the *Global Mask for EID* configuration parameter through the *On/Off* checkbox at the bottom of the webpage. It will apply a global mask, set by the *Mask* configuration parameter to the CAN frame ID. This Mask will be applied only for Extended ID CAN frames. Regular CAN frames will not be affected by this setting.

$\forall ID_{CAN} \in \mathbf{EID}$, if $(ID_{CAN} \& Mask) \in [ID_{From}; ID_{To}]$, then the CAN frame is accepted.
(2)

Where: \mathbf{EID} – Extended ID CAN frame set,
 $Mask$ – *Mask* configuration parameter.
& – Bitwise AND operator.

This setting is suitable for J1939 filtering.

All CAN ID range filters run in parallel. It is sufficient to satisfy requirements of any active filter to pass the CAN message to the Ethernet network.

If no active filters are defined, it is considered that the CAN ID range filters are disabled, and do not participate in the message filtering process. In this case, *ID Range Filters* are *Off* on the home page, and, if other filters are also disabled, all CAN input messages will be sent to the Ethernet network.

3.5.3 CAN ID Mask Filters

The CAN ID mask filters are set through the *CAN1(2) ID Mask Filter* configuration webpage for each CAN port.

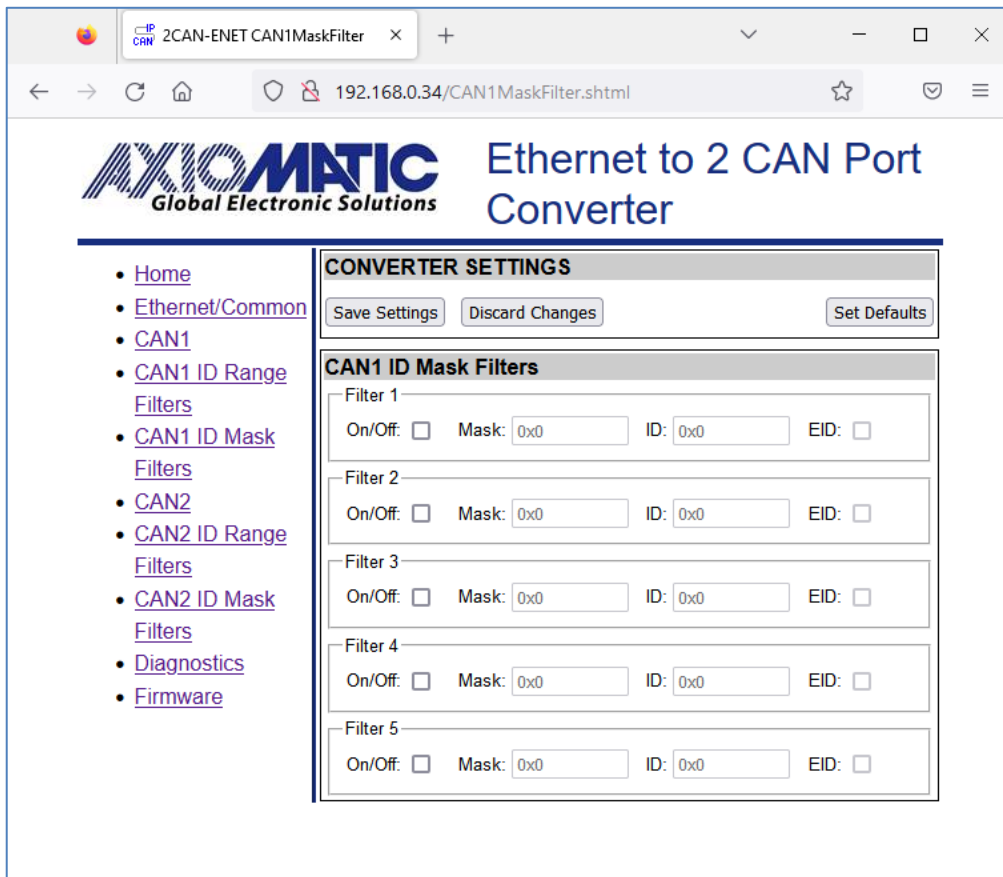


Figure 14. Converter CAN ID Mask Filters Page

There are five independent CAN ID mask filters: *Filter 1*, *Filter 2*, ..., *Filter 5* available to the user.

Once the filter is activated by checking the *On/Off* box, the CAN input frames will pass through the filter to the Ethernet network only if their CAN ID satisfies the following condition:

$$\text{If } ID = ID_{CAN} \& \text{Mask, then the CAN frame is accepted.} \\ (3)$$

Where: ID_{CAN} – CAN frame ID,
 $Mask$ – *Mask* configuration parameter,
 ID – *ID* configuration parameter,
 $\&$ – Bitwise AND operator.

The *EID* box (*Extended ID* box) defines whether the CAN message ID is regular or extended.

All CAN ID mask filters run in parallel the same way as CAN ID range filters. It is sufficient to satisfy requirements of any active filter to send the CAN message to the Ethernet network.

If no active filters are defined, it is considered that the CAN ID mask filters are disabled, and do not participate in the message filtering process. In this case, *ID Mask Filters* are

Off on the home page, and, if other filters are also disabled, all CAN input messages will be sent to the Ethernet network.

4 CONVERTER DIAGNOSTICS

The user can see real-time diagnostic information on the *Diagnostics* page on the converter internal website. The connection to the converter embedded web server is described in the [Converter Configuration](#) section.

To see the *Diagnostics* page, Figure 15, the user should click on the *Diagnostics* link on the left side of the web page.

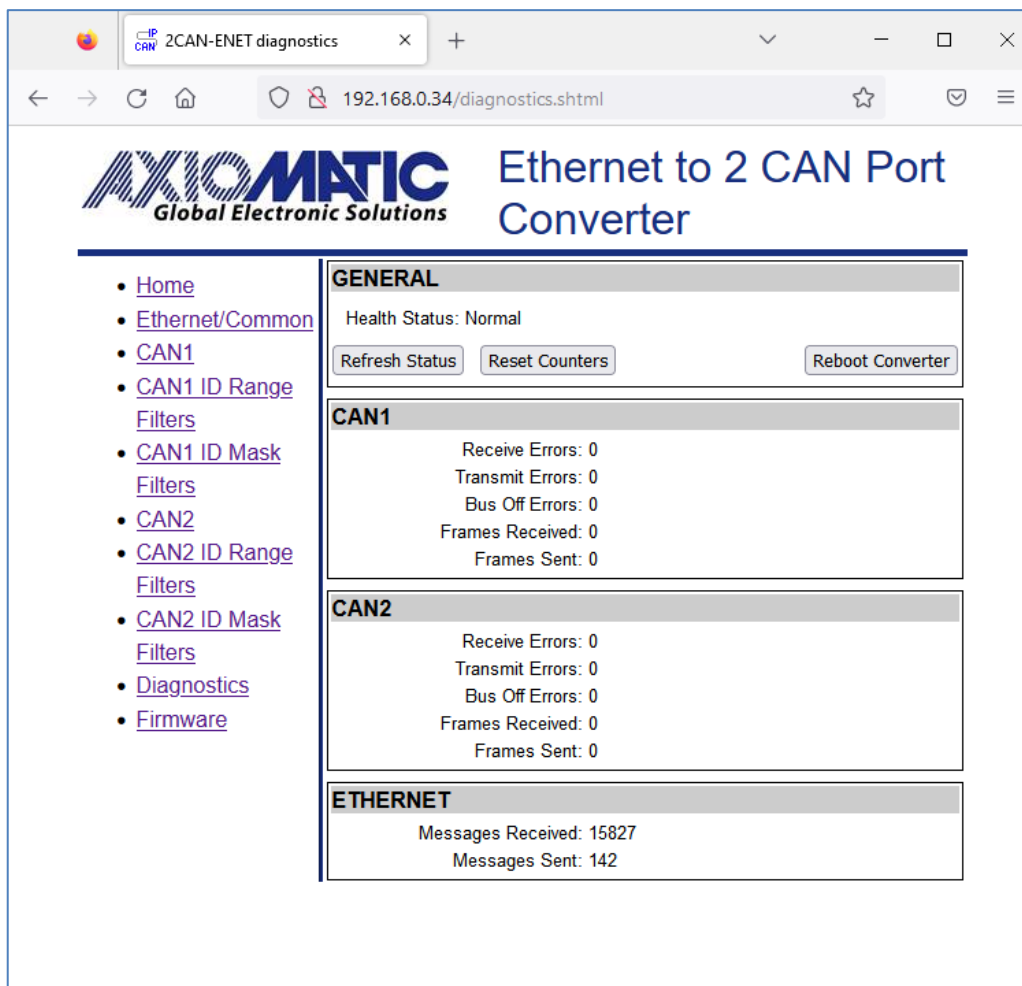


Figure 15. The Converter Diagnostics Page

The *Diagnostics* page shows the *Health Status* of the converter together with the *CAN1(2)* and *Ethernet* statistics.

The user can refresh the values on the page by pressing the *Refresh Status* button or reset the statistic counters by pressing *Reset Counters* button. The *Reboot Converter* button activates the converter rebooting.

The converters do not retain the diagnostic information. All information is lost when the power is shut down.

4.1 Health Status

The converter *Health Status* is an aggregated system run-time parameter calculated on the base of operational statuses of the major device hardware and software components.

The *Health Status* presents the overall operational status of the *Dual CAN FD to Ethernet Converter*, based on the following rules, see Table 5.

Table 5. Health Status

Health Status	Condition
<i>Error</i>	<i>Error</i> is reported when at least one operational status is in the <i>Error</i> state.
<i>Warning</i>	<i>Warning</i> is reported when at least one operational status is in the <i>Warning</i> state and there are no operational statuses in the <i>Error</i> state.
<i>Undefined</i>	<i>Undefined</i> is reported when at least one operational status is in the <i>Undefined</i> state and there are no operational statuses in the <i>Error</i> or <i>Warning</i> state.
<i>Normal</i>	<i>Normal</i> is reported when all operational statuses are in the <i>Normal</i> state.

If the *Health Status* is different from *Normal*, the user will see a verbose message on the *Diagnostics* web page below the *Health Status* describing which operational status is causing a problem, see Figure 16.

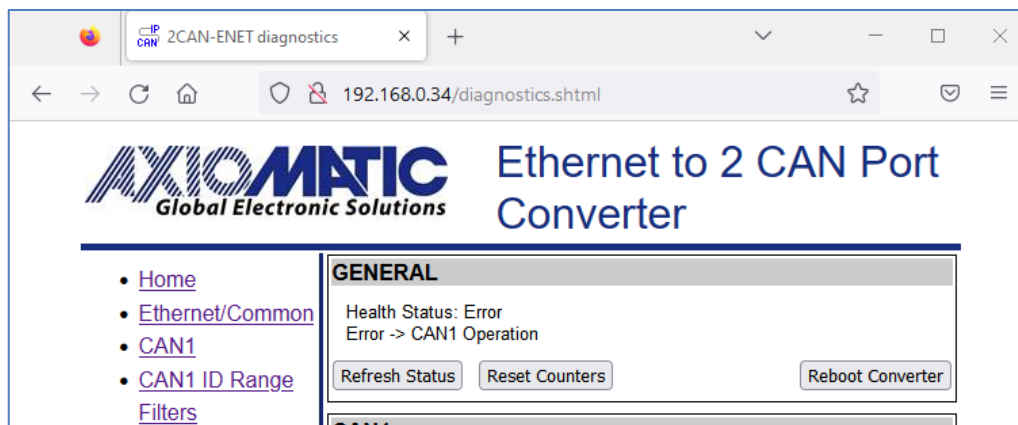


Figure 16. Health Status Message on CAN Error

In case several operational statuses differ from *Normal*, all of them will be shown on the *Diagnostics* page.

The Health Status is also transmitted by the Axiomatic Proprietary protocol [1,3]. Since the *Health Status* format was designed for single channel CAN converters, it will contain CAN status of the CAN port with the routing address: (Group=0, Channel=1) or CAN1 status if no ports use the routing address: (Group=0, Channel=1).

4.2 Converter Rebooting

The user can reboot the converter, when necessary, using the *Reboot Converter* button.

The rebooting operation takes 10 seconds. The user will see the *Reboot* screen with a countdown during this operation, see Figure 17.

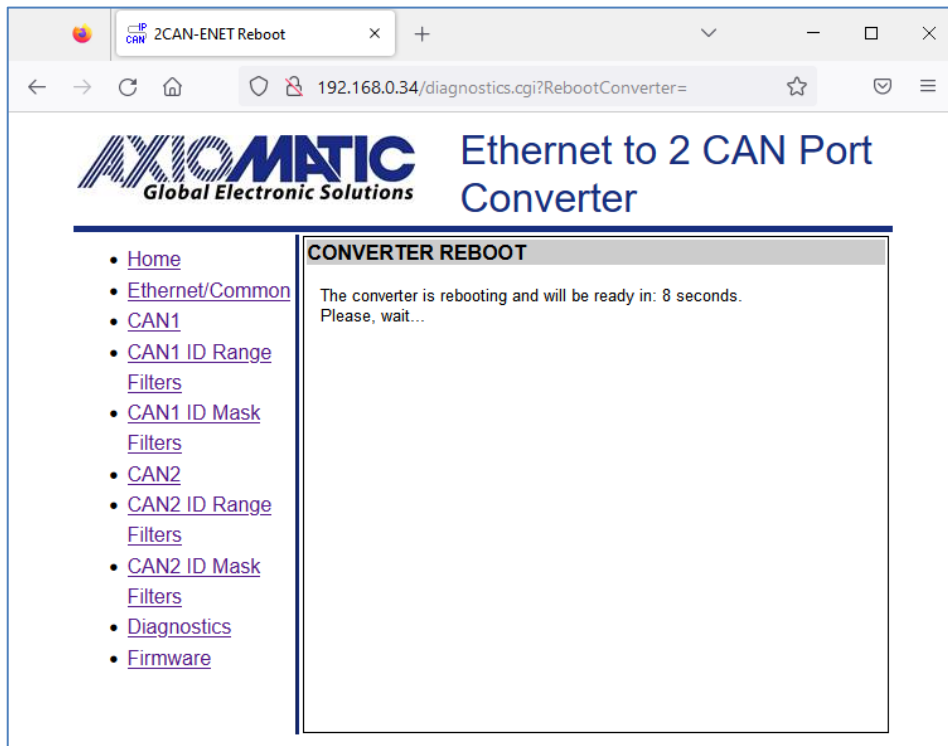


Figure 17. The Converter Reboot Screen

When rebooting is over, the *Diagnostics* page will be reloaded.

5 FIRMWARE UPDATE

The converter firmware can be updated through the converter internal website in the field.

The update procedure is performed in two stages. First, the application firmware is uploaded into the converter internal flash. During this stage, the converter checks the firmware checksum and whether it can be programmed into the unit.

Then, upon the user confirmation, the firmware is programmed into the microcontroller and the unit is restarted. At the end of this process, the user should see the new firmware version number on the converter home page in the browser.

The details of the firmware update are provided below.

5.1 Uploading the New Firmware

To upload the new firmware, the user should activate the *Firmware Uploading* page, Figure 18, by clicking on the *Firmware* link on the left side of the webpage¹.

¹Please note that the *Current Firmware Version* number in the figures below may be different from the firmware version number described in the manual.

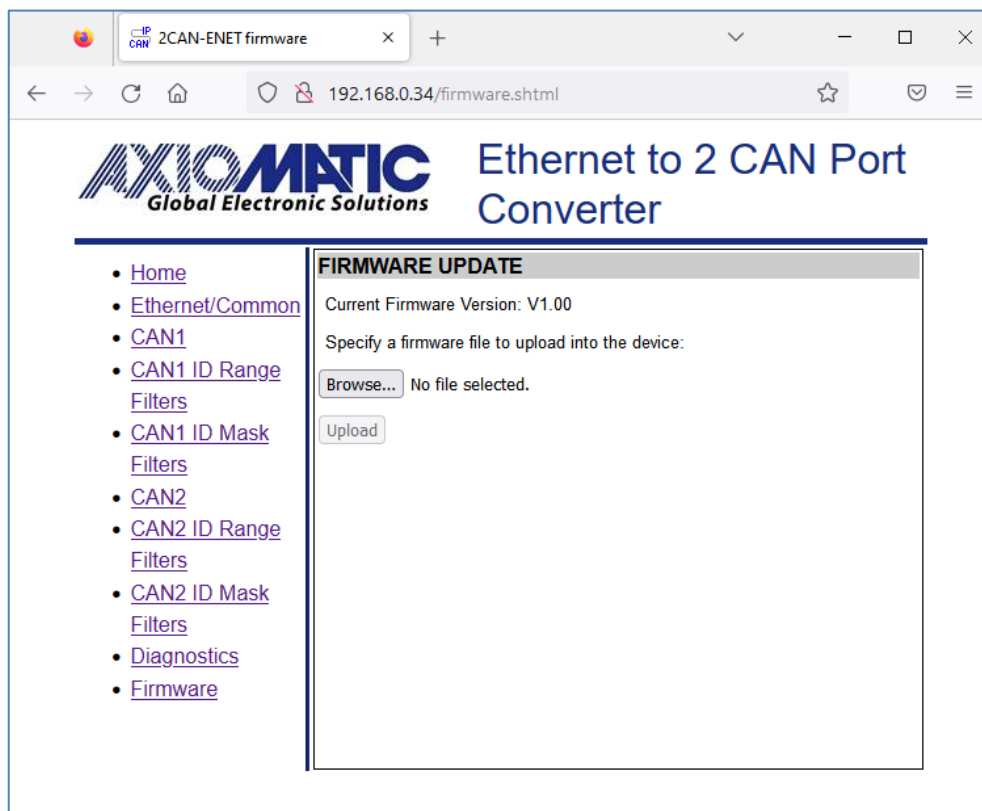


Figure 18. Firmware Uploading Page

Then the user selects the new firmware file using the *Browse...* button.

The firmware file is provided by Axiomatic in a proprietary binary format with extension: .af. The file name should have the following format: AF-23013-X.XX.af, where the UMAX140970

<X.XX> field wildcard reflects the firmware version number. We will use AF-23013-1.00.af file for illustration of the firmware update process in this manual.

When the file is selected, the user should press the *Upload* button. The user will see the dynamic message: “Loading...” in the bottom of the screen and then, if everything is in order, the converter will switch automatically to the *Firmware Update* page.

5.2 Applying the New Firmware

On the *Firmware Update* page, the user will see the new firmware file information, see Figure 19.

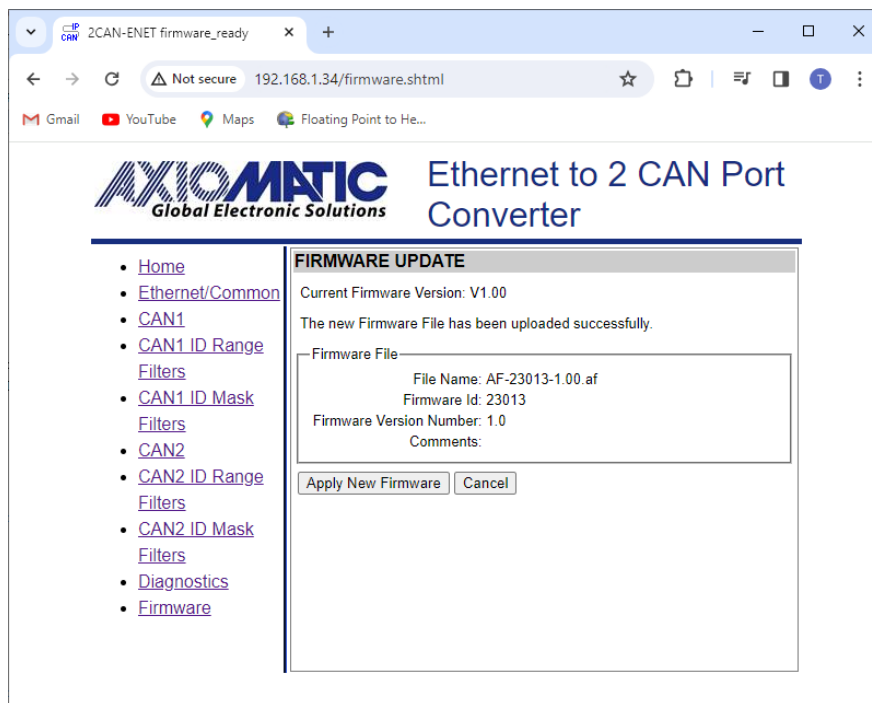


Figure 19. Firmware Update Page

From this point, the user can cancel the firmware update process and keep the old firmware or proceed with flashing the new firmware into the microcontroller by pressing the *Apply New Firmware* button.

When the user presses the *Apply New Firmware* button, the firmware update process is activated, and the *Firmware Upload* page will show the countdown timer, see Figure 20.

The countdown timer is set for 30 seconds necessary to complete the flashing process and reboot the unit.

The converter home page will be displayed after rebooting. The user will see the new application firmware version number in the *Device Information* section on the converter home page¹.

¹In our example, it is the same 1.00 version number since we used the firmware version 1.00 to illustrate the firmware update process.

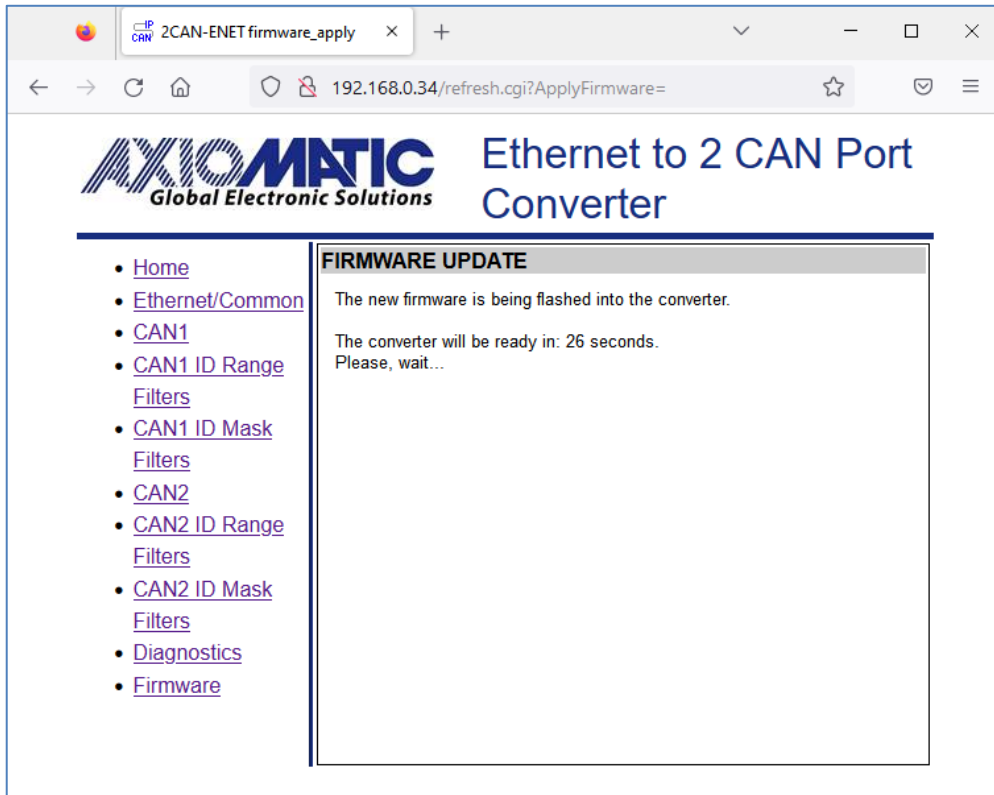


Figure 20. Firmware Update Countdown has been Started

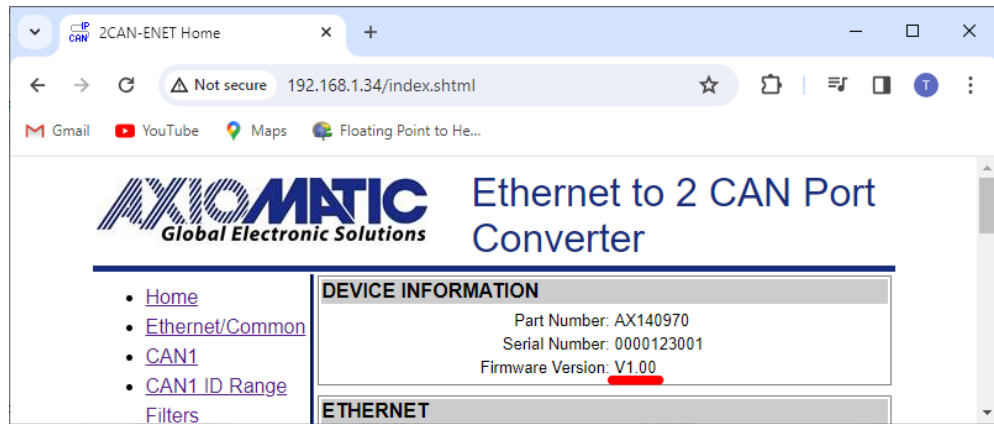


Figure 21. Firmware Version Number After Flashing.

6 CONVERTER DEPLOYMENT

There are two major approaches in using the Dual CAN FD to Ethernet. One is to use the converter on its own as a CAN extender, a bridge or a baud rate converter.

For example, a pair of converters connected through the Ethernet port can bridge together two remote CAN networks. This example can be extended to several CAN networks running at different baud rates in various remote locations, connected together through the Ethernet LAN using the converters.

The dual CAN port of the converter allows local bridging of two CAN networks with different baud rate or connecting a Classical CAN with CAN FD network. When CAN FD frames are sent to the Classical CAN network, their data load is truncated to 8-bytes. All local bridging can be combined with remote bridging over the Ethernet port.

The second approach is to use the converter together with other IP devices that can directly communicate with the converter over the IP network. This approach requires writing a custom software for interfacing with the converter.

Since the converter uses a proprietary communication protocol, Axiomatic provides CAN-ENET Software Support Package (SSP), p/n AX140910, for interfacing with the converter. The SSP is downloadable from www.axiomatic.com, log-in section. Version 3.0.0 or higher that supports CAN FD and internal routing of CAN frames for devices with multiple CAN ports should be used.

The majority of Axiomatic PC software tools support Dual CAN FD to Ethernet converter. They can connect to the CAN bus using this converter the same way as they connect to other Axiomatic Ethernet or Wi-Fi to CAN converters. The Axiomatic Electronic Assistant (EA) can communicate with the converter starting from version 5.11.82.0, and the CAN Assistant – Scope and CAN Assistant – Visual support the converter starting from version 3.0.0.

The CAN port routing address on the Dual CAN FD to Ethernet converter should be set to (Group=0, Channel=1) to communicate with Axiomatic PC software in case the software version does not support the CAN frame routing. The *Converter Type* should be set to *Axiomatic Ethernet-CAN Converter* in the PC software.

The use of the Dual CAN FD to Ethernet converters for bridging CAN networks with or without baud rate conversion is described below. There is no need in custom software for this type of the converter deployment.

6.1 CAN Network Bridging

To bridge two remote CAN networks, the user can simply connect a Dual CAN FD to Ethernet converter to each of the remote CAN network and configure the converters the way that they will talk with each other over the Ethernet port. By using the second CAN port, two pairs of CAN networks can be connected together.

6.1.1 Hardware Setup

The converters should be connected through a local or global IP network, see: Figure 22 and Figure 23.

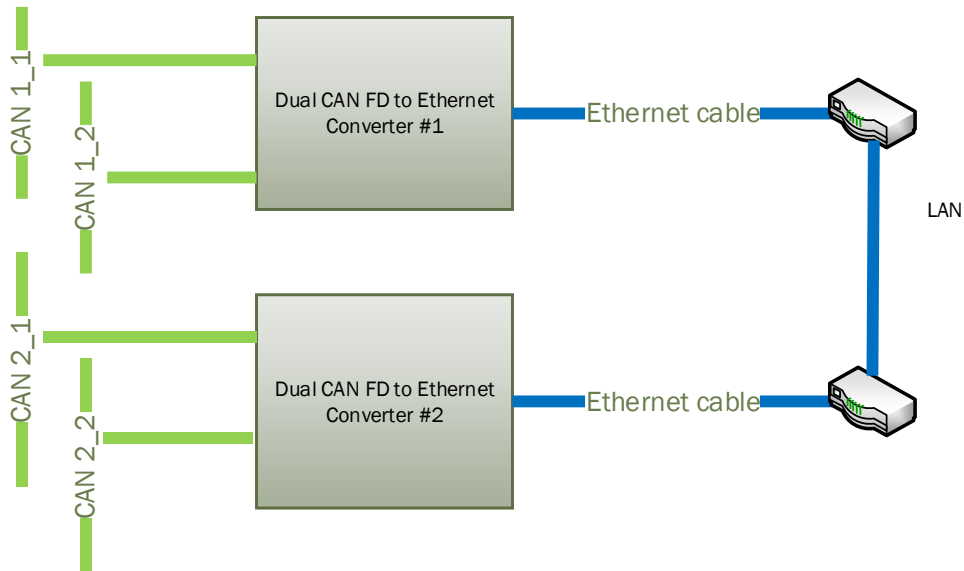


Figure 22. Local Connection through the LAN

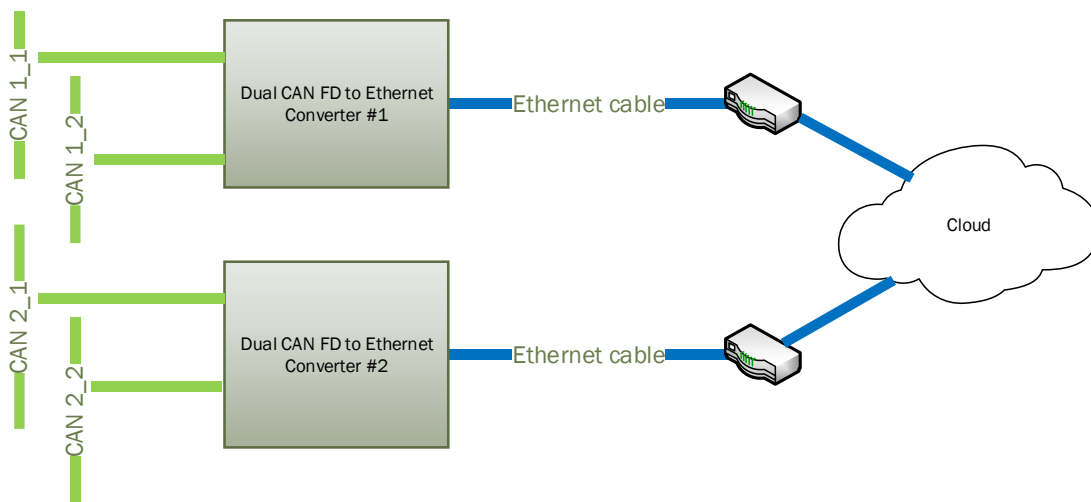


Figure 23. Global Internet Connection through the WAN

In the simplest scenario, two pre-configured converters can be connected by an Ethernet cable, see: Figure 24. Due to the Auto-MDIX feature, both: the straight and crossover cables can be used.

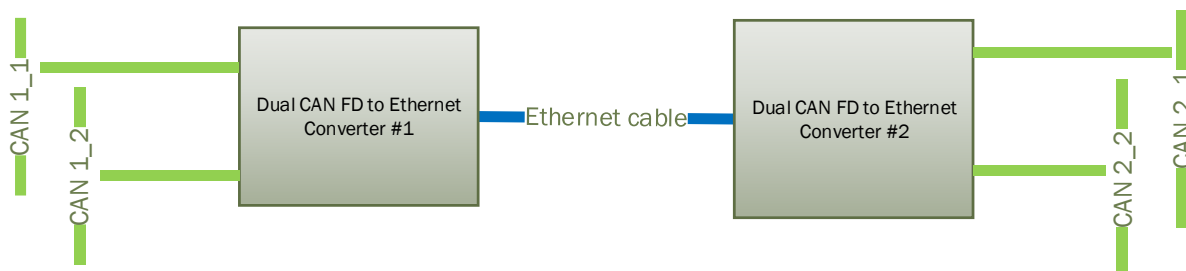


Figure 24. Simple Local Connection

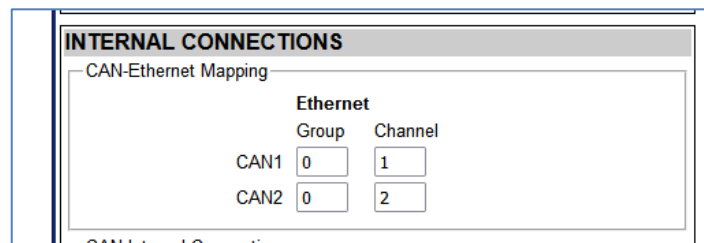
6.1.2 Converter Configuration

After the physical connection is established, the converters should be configured to exchange messages between each other. First, we need to assign routing addresses to CAN ports on both devices. Second, since the converters support client/server communication model, one of the converters should be configured as a server and the other one – as a client. The last step will be individual configuration of CAN ports.

6.1.2.1 CAN Port Routing Addresses

We need to assign the same routing address to CAN ports on different devices if we want them to communicate with each other. In our example, we virtually cross-connect ports on two devices. We connect CAN1 from converter #1 with CAN2 from converter #2 and CAN2 from converter #1 with CAN1 from converter #2.

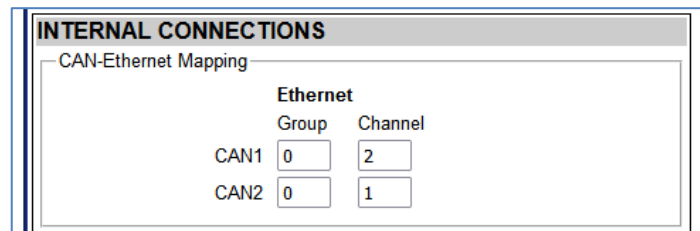
We start from setting two different CAN ports routing addresses on the *Ethernet/Common* webpage, CAN-Ethernet Mapping, for converter #1, see Figure 25.



INTERNAL CONNECTIONS		
CAN-Ethernet Mapping		
Ethernet		
	Group	Channel
CAN1	0	1
CAN2	0	2

Figure 25. CAN-Ethernet Mapping Example. Converter #1

Then we set CAN1 address equal to CAN2 and CAN2 – equal to CAN1 on the *Ethernet/Common* webpage, CAN-Ethernet Mapping, for converter #2, see Figure 26.



INTERNAL CONNECTIONS		
CAN-Ethernet Mapping		
Ethernet		
	Group	Channel
CAN1	0	2
CAN2	0	1

Figure 26. CAN-Ethernet Mapping Example. Converter #2

Now our ports from both devices are virtually connected together.

6.1.2.2 Server Configuration

To configure the converter as a server, first set: *Device IP Address*, *Device Port*, *Device Subnet Mask* and *Device Default Gateway* to the appropriate values received from your network administrator on the *Ethernet/Common* page of the converter #1.

Configure the *Device Port Type* to *UDP* or *TCP* depending on the required message reliability and acceptable delays. Use unreliable but fast UDP when speed is a priority or reliable but slow TCP when message reliability is more important than the speed.

Set *Auto Connect to Remote* to *No* to disable the client side of the converter. The *Remote IP Address* and *Remote Port* can be left untouched since they are not used by the converter when the client mode is disabled. They are grayed on the *Settings* page in this mode.

An example of the converter configuration as a server is presented in Figure 27. This is the converter default configuration.

The screenshot shows the 'ETHERNET' configuration page for Converter #1. It is divided into two sections: 'Server' and 'Client'.
In the 'Server' section, the following fields are visible:
- Device IP Address: 192.168.0.34
- Device Port: 4000
- Device Port Type: UDP TCP
- Web Server Port: 80
- Device Subnet Mask: 255.255.255.0
- Device Default Gateway: 192.168.0.1
In the 'Client' section, the following fields are visible:
- Auto Connect to Remote: No (dropdown menu)
- Remote IP Address: 192.168.0.35 (disabled)
- Remote Port: 4000 (disabled)

Figure 27. The Converter Server Configuration Example. Converter #1

6.1.2.3 Client Configuration

In the client configuration, the user should set *Device IP Address*, *Device Port*, *Device Subnet Mask* and *Device Default Gateway* to the appropriate values received from the network administrator the same way as with the server configuration on the *Ethernet/Common* page of the converter #2.

After that, the user should set the *Device Port Type*, *Remote IP Address* and *Remote Port* to match the settings of the converter in the server mode and activate the client mode by setting the *Auto Connect to Remote* to *Yes*, see Figure 28.

The screenshot shows the 'ETHERNET' configuration page for Converter #2. It is divided into two sections: 'Server' and 'Client'.
In the 'Server' section, the following fields are visible:
- Device IP Address: 192.168.0.35
- Device Port: 4000
- Device Port Type: UDP TCP
- Web Server Port: 80
- Device Subnet Mask: 255.255.255.0
- Device Default Gateway: 192.168.0.1
In the 'Client' section, the following fields are visible:
- Auto Connect to Remote: Yes (dropdown menu)
- Remote IP Address: 192.168.0.34
- Remote Port: 4000

Figure 28. The Converter Client Configuration Example. Converter #2

6.1.2.4 CAN Port Configuration

CAN port configuration can be done similarly on both devices using *CAN1(2)* webpages.

First, enable the channel and set the *CAN Type*. Then, set *Loopback Messages* to *No* to avoid sending the same messages back and forth between ports, which will lead to network saturation and communication failure. For *Classical CAN* ports, set the port *Baud Rate*, see Figure 29.

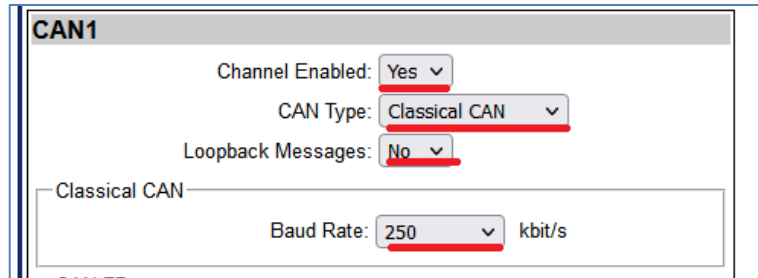


Figure 29. The Converter CAN Port Configuration Example.

Configure the same way all CAN ports on both converters. Now converters are ready to start communication with each other.

6.1.2.5 Internet Connection

When the converters are connected over the internet, the *Remote IP Address* of the client will be a public IP address of the server, not the internal server IP address presented as the *Device IP Address* on Figure 27. The network administrator on the server side will be required to configure port forwarding to open internet access to the converter in the server mode.

6.1.2.6 Connecting Several Converters Together

When the converter is configured as a client, it will still act as a server accepting connections on the *Device Port* from other clients. This adds versatility to the converter configurations since the same converter can be used together with both: client and server communication nodes. As an example, the user can establish an unlimited number of daisy-chain client-server connections, see: Figure 30.

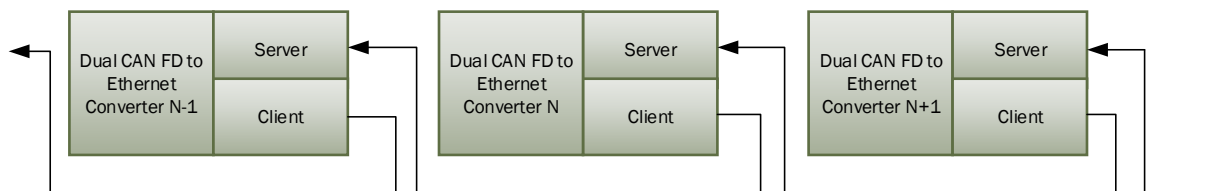
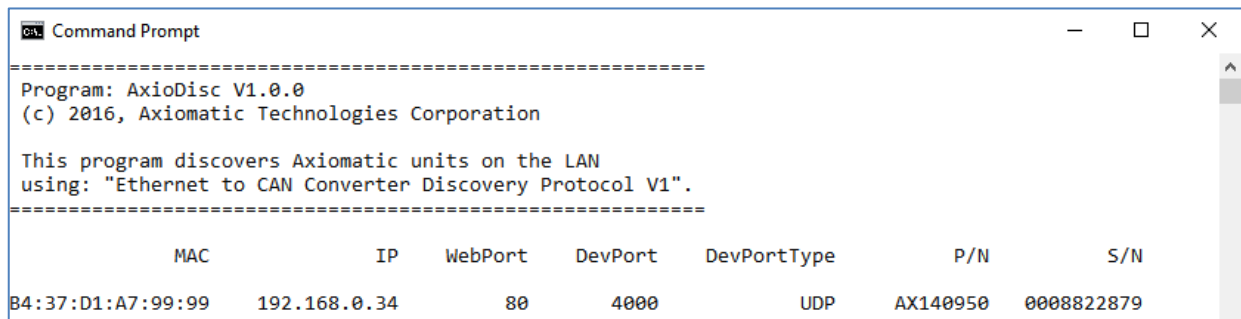


Figure 30. Daisy-Chain Converter Connection

7 CONVERTER DISCOVERY

In case the IP address and/or web server port is unknown or has been lost, the user can recover them using the Axiomatic `AxioDisc.exe` Windows console application. The application uses a proprietary discovery protocol and can recover IP locations of Axiomatic converters on a LAN. The `AxioDisc.exe` application is available upon request.

The application sends a UDP request to the global IP address `255.255.255.255`, port `35100` and waits for the responses from the converters located on the same local network. The converter discovery response includes the unit MAC Address, IP Address, web server port, device port, device port type, the converter part number and serial number.



```
-----
Program: AxioDisc V1.0.0
(c) 2016, Axiomatic Technologies Corporation

This program discovers Axiomatic units on the LAN
using: "Ethernet to CAN Converter Discovery Protocol V1".
-----
      MAC          IP      WebPort  DevPort  DevPortType      P/N      S/N
-----
B4:37:D1:A7:99:99  192.168.0.34      80      4000      UDP      AX140950  0008822879
```

Figure 31. `AxioDisc.exe` Converter Discovery Application

The discovery protocol is also supported by the CAN-ENET Software Support Package, p/n AX140910, and can be used for network discovery and plug-and-play connection to Axiomatic Dual CAN FD to Ethernet converters.

The `AxioDisc.exe` application can run on Windows starting from Win XP SP3. It was tested on Win XP SP3, Win 7 and Win 10. In case the application cannot find standard dlls, the Visual C++ Redistributable for Visual Studio 2015 x86 must be installed on the user's computer from the Microsoft website: <https://www.microsoft.com/en-ca/download/details.aspx?id=48145>

8 TECHNICAL SPECIFICATIONS

8.1 Power Supply

8.1.1 Input

The power supply input is located on the Ethernet connector. It uses automotive battery power¹.

Table 6. Power Supply Input

Parameter	Value	Remarks
Supply Voltage	9...36 VDC	12V, 24V – nominal
Quiescent Current ²	150 mA 75 mA	12V – typical 24V – typical
Protection	Overvoltage, Reverse Polarity Transients/Surge	>38V

¹The power supply is not compatible with the PoE (Power over Ethernet) IEEE 802.3 standard.

²Both CAN ports are On, no CAN traffic. Ethernet is not connected.

8.1.2 Output

The Power Pass-Through supply output is located on the CAN connector.

Table 7. Power Supply Output

Parameter	Value	Remarks
Voltage Output	9...36 VDC	Pass-through voltage from the power supply input
Current Output	0.7A	Maximum pass-through current
Voltage Drop	1.5V	Maximum
Protection	Overcurrent at ~1A with auto-retry Short to Battery/Ground	

8.2 Ethernet Port

Table 8. Ethernet Port Parameters

Parameter	Value	Remarks
Number of Ports	1	Fast Ethernet, ESD, EFT Protected
Port Type	10BASE-T, 100BASE-TX	Auto-configuration and full-duplex supported
MDIX	Auto-MDIX	Auto-crossover to eliminate cabling mismatch
LED Indicators	Speed/Activity	Green LEDs 1. POWER 2. 10/100 LED: Off = 10 Mbit/s On = 100 Mbits/s 3. LINK/ACT LED: Off = No link Solid = Link Blinking = Activity on ethernet

Parameter	Value	Remarks
Protocols	Ethernet IEEE 802.3, IP, ICMP, ARP, UDP, TCP, HTTP, Proprietary ¹	CAN messages are transmitted using a proprietary application protocol running on top of the user selectable UDP or TCP transport protocol [1] The internal web server uses HTTP protocol. The unit supports a proprietary discovery protocol [2]
Server Mode	Up to 10 bi-directional simultaneous connections	Up to 9 connections, if the Client mode is enabled
Client Mode	1 remote bi-directional connection	Auto-connect to a remote server if connection is dropped or temporarily unavailable. Client mode can be disabled
Web server	Provided	Always enabled for converter configuration and diagnostics
Internal Diagnostics	Health Status ¹	Internal health status of the converter is transmitted in heartbeat messages [3]. It is also available from the web server

¹Supported by *CAN-ENET Software Support Package (SSP)*, P/N AX140910, v3.0.0+.

Reference documents describing proprietary protocols and *Health Status* field format are presented below. The documents are available upon request.

Table 9. Reference Documents

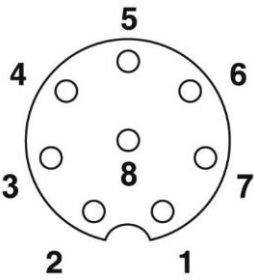
Reference Number	Document Name
[1]	O. Bogush, "Ethernet to CAN Converter Communication Protocol. Document version: 5", Axiomatic Technologies Corporation, December 14, 2022
[2]	O. Bogush, "Ethernet to CAN Converter Discovery Protocol. Document version: 1A", Axiomatic Technologies Corporation, April 5, 2021
[3]	O. Bogush, "Ethernet to CAN Converter Health Status. Document version: 3", Axiomatic Technologies Corporation, April 5, 2021

8.2.1 Ethernet Connector

M12 socket, 8-pin, A-coded, female connector, Phoenix Contact, P/N: 1441817.

Table 10. Ethernet Connector Pinout

PIN #	Description
1	PWR_IN
2	PWR_IN_GND
3	PWR_IN_GND
4	TX_N
5	RX_P
6	TX_P
7	PWR_IN
8	RX_N



Use A-coded mating connectors compliant with IEC 61076-2-101:2012.

The AX070531 Ethernet and Power Cable - 1.7m (5.5 ft.), 8-pin M12 A-coded, Unterminated Leads, Ethernet Jack, can be used for experimenting. The cable is rated for -40 to +75°C.

8.3 CAN FD Ports

Table 11. CAN FD Parameters

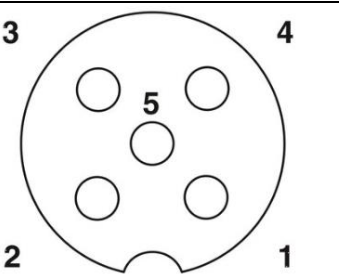
Parameter	Value	Remarks
Number of Ports	2	Individually galvanically Isolated, ESD, EFT protected. Twisted pair, CiA 601-6. Shield connection is provided if shielded cable is used. No internal terminating 120 Ω resistor
Port Type	Classic CAN	High-Speed CAN with up to 1 Mbit/s bit rate and up to 8-byte data payload per frame, ISO11898-1 (Bosch CAN 2.0A and B)
	ISO CAN FD	CAN FD frame format according to ISO11898-1:2015. Up to 1Mbit/s arbitration, up to 8Mbit/s data phase bit rate
	Non-ISO CAN FD	CAN FD frame format according to Bosch CAN FD Specification V1.0. Up to 1Mbit/s arbitration, up to 8Mbit/s data phase bit rate
Baud Rate	1000 kbit/s, 800 kbit/s, 666.6(6) kbit/s, 500 kbit/s, 250 kbit/s, 125 kbit/s, 100 kbit/s, 83.3(3) kbit/s, 50 kbit/s, 20 kbit/s, 10 kbit/s	For Classic CAN only
Frame Filtering	5 CAN ID Range Filters 5 CAN Mask Filters	Per channel, can be disabled. All types of Classic and CAN FD frames are supported.

CAN network requires two 120 Ohm terminating resistors, one on each side of the CAN bus.

8.3.1 CAN Connector 1

M12 socket, 5-pin, A-coded, male connector, Phoenix Contact, P/N: **1441778**.

PIN #	Description
1	CAN1_SHIELD
2	POUT1
3	GND1
4	CAN1_H
5	CAN1_L



Use mating A-coded connectors compliant with IEC 61076-2-101:2012.

AX070532 CAN Cable - 1.5 m (5 ft.), 5-pin M12 A-coded, Unterminated Leads, can be used for experimenting. The cable is rated for -40°C to 105°C.

8.3.2 CAN Connector 2

M12 socket, 5-pin, A-coded, male connector, Phoenix Contact, P/N: **1441778**.

PIN #	Description	
1	CAN2_SHIELD	
2	POUT2	
3	GND2	
4	CAN2_H	
5	CAN2_L	

Use mating A-coded connectors compliant with IEC 61076-2-101:2012.

AX070532 CAN Cable - 1.5 m (5 ft.), 5-pin M12 A-coded, Unterminated Leads, can be used for experimenting. The cable is rated for -40°C to 105°C.

8.4 General Specifications

Table 12. General Specifications

Parameter	Value	Remarks
Operating Temperature	-40°C to 75°C	-40°F to 167°F
Storage Temperature	-40°C to 85°C	-40°F to 185°F
Environmental Protection	IP67	IEC 60529. With mated connectors
Compliance	RoHS Directive	
Enclosure Size and Material	4.21 in x 3.06 in x 1.57 in (106.8 mm x 77.7 mm x 39.9 mm) Injection molded enclosure and cover. Laser welded. PA66, 30% glass fiber reinforced Flammability rating: UL 94 HB	L x W x H including connectors. See dimensional drawing.
Weight	0.25 lb. (0.12 kg)	
Vibration	Pending	
Shock	Pending	

8.5 Dimensional Drawing

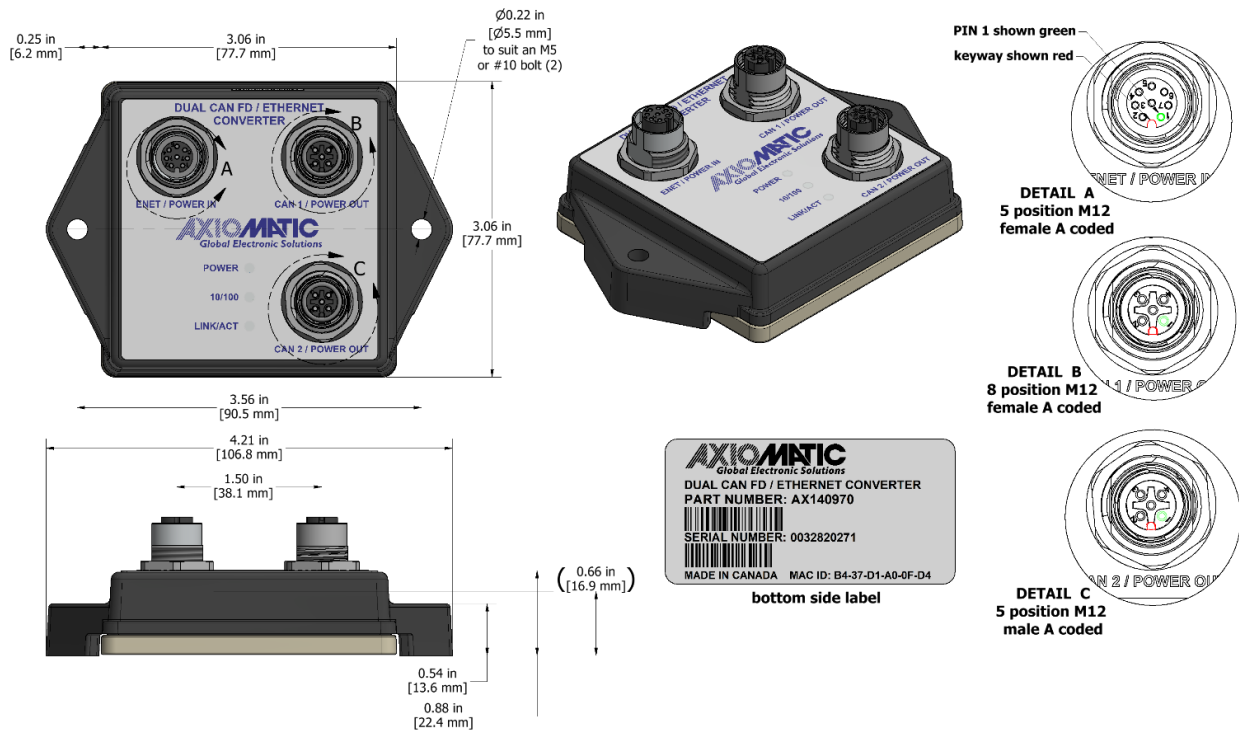


Figure 32. Dimensional Drawing

9 THIRD PARTY SOFTWARE LICENSE NOTICES

This section contains Third Party Software License Notices and/or Additional Terms and Conditions for licensed third-party software components included in the Dual CAN FD to Ethernet Converter firmware.

Table 13. Third Party Software License Notices

Third Party Software	License Notice/Terms
STM32CubeH7	<p>Copyright © 2022 STMicroelectronics. All rights reserved</p> <p>SOFTWARE PACKAGE LICENSE AGREEMENT</p> <p>BY INSTALLING COPYING, DOWNLOADING, ACCESSING OR OTHERWISE USING THIS SOFTWARE PACKAGE OR ANY PART THEREOF (AND THE RELATED DOCUMENTATION) FROM STMICROELECTRONICS INTERNATIONAL N.V, SWISS BRANCH AND/OR ITS AFFILIATED COMPANIES (STMICROELECTRONICS), THE RECIPIENT, ON BEHALF OF HIMSELF OR HERSELF, OR ON BEHALF OF ANY ENTITY BY WHICH SUCH RECIPIENT IS EMPLOYED AND/OR ENGAGED AGREES TO BE BOUND BY THIS SOFTWARE PACKAGE LICENSE AGREEMENT.</p> <p>Under STMicroelectronics' intellectual property rights and subject to applicable licensing terms for any third-party software incorporated in this software package and applicable Open Source Terms (as defined here below), the redistribution, reproduction and use in source and binary forms of the software package or any part thereof, with or without modification, are permitted provided that the following conditions are met:</p> <ol style="list-style-type: none"> 1. Redistribution of source code (modified or not) must retain any copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form, except as embedded into microcontroller or microprocessor device manufactured by or for STMicroelectronics or a software update for such device, must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of STMicroelectronics nor the names of other contributors to this software package may be used to endorse or promote products derived from this software package or part thereof without specific written permission. 4. This software package or any part thereof, including modifications and/or derivative works of this software package, must be used and execute solely and exclusively on or in combination with a microcontroller or a microprocessor devices manufactured by or for STMicroelectronics. 5. No use, reproduction or redistribution of this software package partially or totally may be done in any manner that would subject this software package to any Open Source Terms (as defined below). 6. Some portion of the software package may contain software subject to Open Source Terms (as defined below) applicable for

Third Party Software	License Notice/Terms
	<p>each such portion (“Open Source Software”), as further specified in the software package. Such Open Source Software is supplied under the applicable Open Source Terms and is not subject to the terms and conditions of license hereunder. “Open Source Terms” shall mean any open source license which requires as part of distribution of software that the source code of such software is distributed therewith or otherwise made available, or open source license that substantially complies with the Open Source definition specified at www.opensource.org and any other comparable open source license such as for example GNU General Public License (GPL), Eclipse Public License (EPL), Apache Software License, BSD license and MIT license.</p> <ol style="list-style-type: none"> 7. This software package may also include third party software as expressly specified in the software package subject to specific license terms from such third parties. Such third party software is supplied under such specific license terms and is not subject to the terms and conditions of license hereunder. By installing copying, downloading, accessing or otherwise using this software package, the recipient agrees to be bound by such license terms with regard to such third party software. 8. STMicroelectronics has no obligation to provide any maintenance, support or updates for the software package. 9. The software package is and will remain the exclusive property of STMicroelectronics and its licensors. The recipient will not take any action that jeopardizes STMicroelectronics and its licensors’ proprietary rights or acquire any rights in the software package, except the limited rights specified hereunder. 10. The recipient shall comply with all applicable laws and regulations affecting the use of the software package or any part thereof including any applicable export control law or regulation. 11. Redistribution and use of this software package partially or any part thereof other than as permitted under this license is void and will automatically terminate your rights under this license. <p>THIS SOFTWARE PACKAGE IS PROVIDED BY STMICROELECTRONICS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS, IMPLIED OR STATUTORY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS ARE DISCLAIMED TO THE FULLEST EXTENT PERMITTED BY LAW. IN NO EVENT SHALL STMICROELECTRONICS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE PACKAGE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.</p>

Third Party Software	License Notice/Terms
	EXCEPT AS EXPRESSLY PERMITTED HEREUNDER AND SUBJECT TO THE APPLICABLE LICENSING TERMS FOR ANY THIRD-PARTY SOFTWARE INCORPORATED IN THE SOFTWARE PACKAGE AND OPEN SOURCE TERMS AS APPLICABLE, NO LICENSE OR OTHER RIGHTS, WHETHER EXPRESS OR IMPLIED, ARE GRANTED UNDER ANY PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF STMICROELECTRONICS OR ANY THIRD PARTY.
FreeRTOS V10.5.1	<p>Copyright (C) 2021 Amazon.com, Inc. or its affiliates. All Rights Reserved. MIT License</p> <p>Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:</p> <p>The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.</p> <p>THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.</p>

Third Party Software	License Notice/Terms
LwIP v2.1.2	<p>Copyright (c) 2001-2004 Swedish Institute of Computer Science. All rights reserved.</p> <p>BSD license</p> <p>Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:</p> <ol style="list-style-type: none"> 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission. <p>THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.</p> <p>The license is available at: http://lwip.wikia.com/wiki/License</p>

10 VERSION HISTORY

User Manual Version	Firmware version	Date	Author	Modifications
1.00	1.xx	February 16, 2024	Dmytro Tsebrii	Initial release.
1.01	1.xx	February 20, 2024	M Ejaz	Reformatting Legacy changes Marketing review Updated technical specifications

OUR PRODUCTS

AC/DC Power Supplies
Actuator Controls/Interfaces
Automotive Ethernet Interfaces
Battery Chargers
CAN Controls, Routers, Repeaters
CAN/WiFi, CAN/Bluetooth, Routers
Current/Voltage/PWM Converters
DC/DC Power Converters
Engine Temperature Scanners
Ethernet/CAN Converters,
Gateways, Switches
Fan Drive Controllers
Gateways, CAN/Modbus, RS-232
Gyroscopes, Inclinometers
Hydraulic Valve Controllers
Inclinometers, Triaxial
I/O Controls
LVDT Signal Converters
Machine Controls
Modbus, RS-422, RS-485 Controls
Motor Controls, Inverters
Power Supplies, DC/DC, AC/DC
PWM Signal Converters/Isolators
Resolver Signal Conditioners
Service Tools
Signal Conditioners, Converters
Strain Gauge CAN Controls
Surge Suppressors

OUR COMPANY

Axiomatic provides electronic machine control components to the off-highway, commercial vehicle, electric vehicle, power generator set, material handling, renewable energy and industrial OEM markets. ***We innovate with engineered and off-the-shelf machine controls that add value for our customers.***

QUALITY DESIGN AND MANUFACTURING

We have an ISO9001:2015 registered design/manufacturing facility in Canada.

WARRANTY, APPLICATION APPROVALS/LIMITATIONS

Axiomatic Technologies Corporation reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Users should satisfy themselves that the product is suitable for use in the intended application. All our products carry a limited warranty against defects in material and workmanship. Please refer to our Warranty, Application Approvals/Limitations and Return Materials Process at <https://www.axiomatic.com/service/>.

COMPLIANCE

Product compliance details can be found in the product literature and/or on axiomatic.com. Any inquiries should be sent to sales@axiomatic.com.

SAFE USE

All products should be serviced by Axiomatic. Do not open the product and perform the service yourself.



This product can expose you to chemicals which are known in the State of California, USA to cause cancer and reproductive harm. For more information go to www.P65Warnings.ca.gov.

SERVICE

All products to be returned to Axiomatic require a Return Materials Authorization Number (RMA#) from rma@axiomatic.com. Please provide the following information when requesting an RMA number:

- Serial number, part number
- Runtime hours, description of problem
- Wiring set up diagram, application and other comments as needed

DISPOSAL

Axiomatic products are electronic waste. Please follow your local environmental waste and recycling laws, regulations and policies for safe disposal or recycling of electronic waste.

CONTACTS

Axiomatic Technologies Corporation
1445 Courtneypark Drive E.
Mississauga, ON
CANADA L5T 2E3
TEL: +1 905 602 9270
FAX: +1 905 602 9279
www.axiomatic.com
sales@axiomatic.com

Axiomatic Technologies Oy
Höytämöntie 6
33880 Lempäälä
FINLAND
TEL: +358 103 375 750
www.axiomatic.com
salesfinland@axiomatic.com