Anybus .NET Bridge

Streamer Mode

USER MANUAL

SCM-1202-120 1.1 en-US ENGLISH

Connecting Devices™

Table of Contents

Page

1 Preface

1.1 About This Document

The Anybus .NET Bridge has two different modes, Message Mode and Streamer Mode.

This manual describes how to install, configure and use the Anybus .NET Bridge in Streamer Mode .

For additional related documentation and file downloads, please visit the support website at www.anybus.com/support.

1.2 Document History

Revision list		
Version	**Date**	**Description**
1.0	2016-09-22	SCM-1202-098 First release
1.1	2018-03-14	SCM-1202-098 Major rewrite
1.2	2019-04-29	Changed Document ID from SCM-1202-098 to SCM-1202-120 for Streamer Mode and SCM-1202-121 for Message Mode Major rewrite

1.3 Related Documentation

The following documents with network and application development specific information are additional to this user manual.

To download the documents, please visit www.anybus.com/support.

Document	Author
.NET Bridge Startup Guide	HMS

1.4 Document Conventions

Ordered lists are used for instructions that must be carried out in sequence:

1. First do this
2. Then do this

Unordered (bulleted) lists are used for:

- Itemized information
- Instructions that can be carried out in any order

...and for action-result type instructions:

► This action...

→ leads to this result

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

```
Monospaced text 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: *Document Conventions, p. 4*

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

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 Description

2.1 Product Description

The .NET Bridge enables factory-floor data to be presented to .NET software applications.

Data sent from a PLC system can be used in .NET applications for statistics, analysis or maintenance.

Data can also be sent from the .NET environment to the PLC for use in daily operation.

The .NET Bridge is a state-of-the-art IT/OT bridge, easily bridging the gap between the operational technology (OT) and the information technology (IT).

The .NET Bridge can be used with a wide range of use cases, such as simple transfer of KPI values, advanced messages with structured data types and transfer of I/O data for big data.

The .NET Bridge acts as a translator between a function block in a PLC and a .NET object in a computer.

The .NET Bridge has two different modes:

Message Mode

A tag data structure is used and the communication is synchronized with handshake.

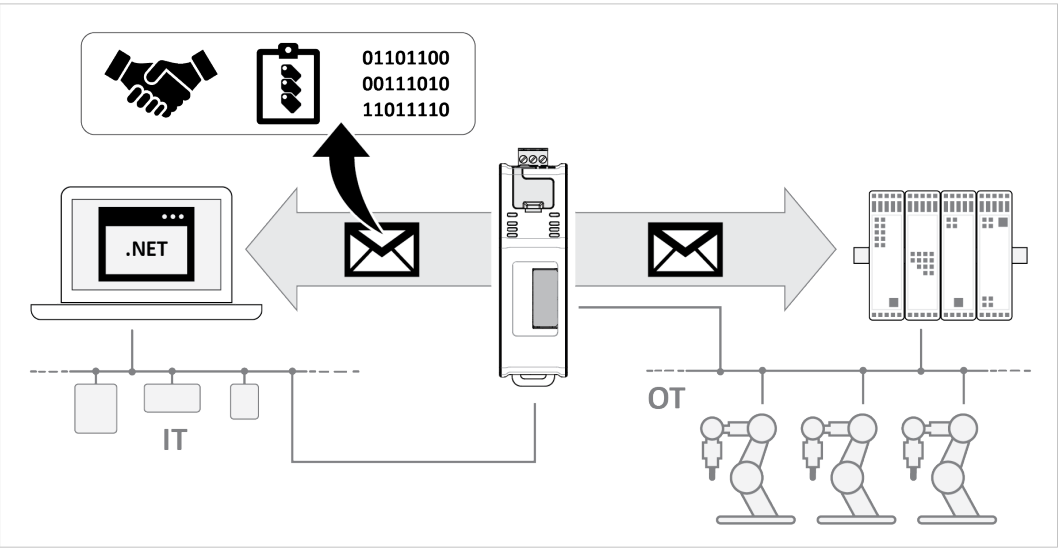


Fig. 1 Message Mode data exchange

Streamer Mode

Raw data is sent between the PLC and the .NET application.

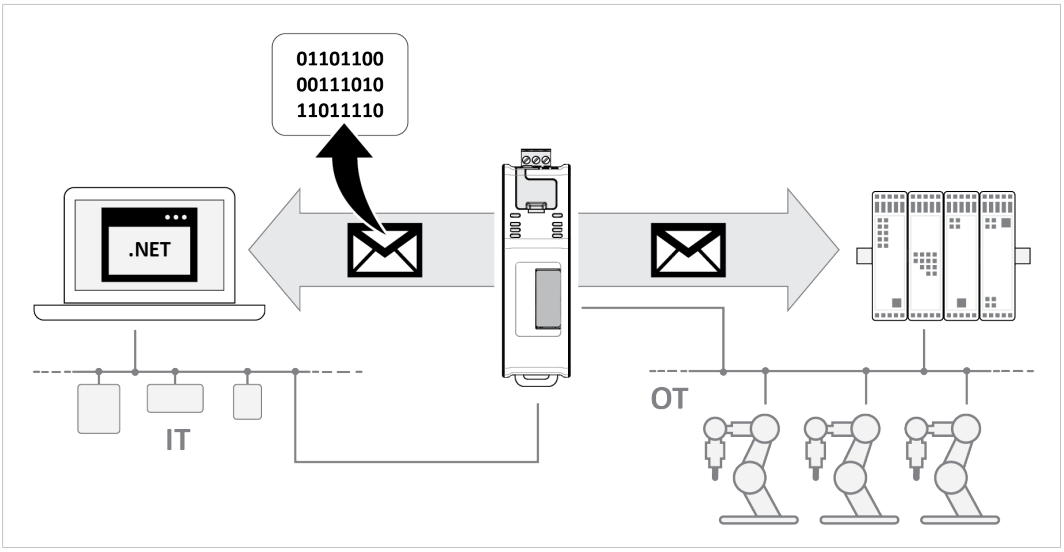


Fig. 2 Streamer Mode data exchange

2.2 Network Security Considerations

The communication between the .NET application and the .NET Bridge is not encrypted, similar to how the communication on the industrial network is not encrypted.

It is recommended to only communicate between the .NET application and the .NET Bridge in a private network.

A virtual private network (VPN) may be used to encrypt communication between the .NET application and the .NET Bridge, if it is needed to communicate over the internet.

3 Preparation

3.1 Prerequisites

3.1.1 Wall Mount Option

> When the .NET Bridge is used in an environment exposed to vibration, increased stability is required.
>
> Use the *Wall-Mount Accessory Kit* and mount the device on a wall instead of a DIN rail.
>
> The *Wall-Mount Accessory Kit* is ordered separately, please visit www.anybus.com for more information.

3.1.2 Required HMS Software

The following HMS software are required:

- Anybus .NET Bridge Setup: Includes Windows-based software needed during the message structure design process and to simulate the .NET application and the PLC application.
- IPconfig: A Windows-based software for configuration of TCP/IP settings in HMS devices.

Download the software applications at www.anybus.com/support.

3.1.3 Required Third Party Software

When developing .NET applications, the following third party software are required:

- Microsoft Excel, or equivalent software that supports the Office Open XML Workbook (xlsx) file format. Needed when using the .NET Bridge Message Mode.
- Microsoft Visual Studio
- Microsoft .NET Framework SDK version 4.5 or later.

On the production computer where the .NET application is installed, only the .NET Framework 4.5 (or later) runtime is required.

.NET Core Runtime is not supported.

3.2 Software Installation

3.2.1 Installing Anybus .NET Bridge Setup

Anybus .NET Bridge Setup software tools are used to generate configuration files and to simulate .NET applications and PLC applications.

Before You Begin

Visit www.anybus.com/support and download the *Anybus .NET Bridge Setup* zip file.

Before starting the installation, it is recommended that you close all other applications.

Procedure

Installing Anybus .NET Bridge Setup software tools:

1. Unzip the *Anybus .NET Bridge Setup* zip file.
2. Double-click the *Anybus .NET Bridge Setup* application file.
3. The Setup Anybus .NET Bridge installer window appears.

   Click **Next** to begin installation.

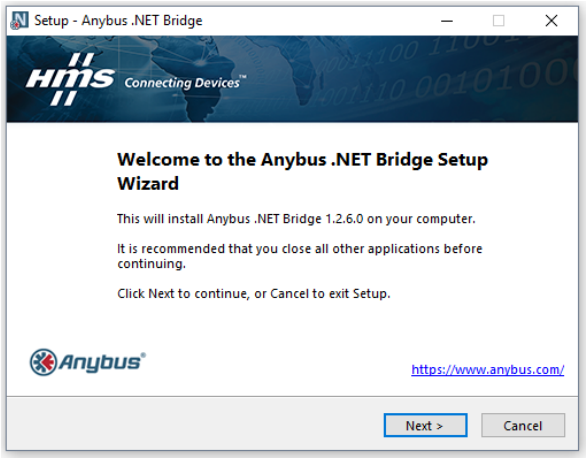


Fig. 3 Anybus .NET Bridge Setup Wizard

4. Follow the prompts in the Anybus .NET Bridge Setup Wizard to complete the installation.
5. Click **Finish**.

Result

→ The following Anybus .NET Bridge Setup software tools are installed on your computer:

- Anybus .NET Bridge Generator
- Anybus .NET Bridge Message Mode PLC Simulator
- Anybus .NET Bridge Message Mode .NET Simulator
- Anybus .NET Bridge Streamer Mode PLC Simulator
- Anybus .NET Bridge Streamer Mode .NET Simulator

3.2.2 Installing IPconfig

IPconfig is used to configure the TCP/IP settings in HMS devices.

Before You Begin

Visit www.anybus.com/support and download the *IPconfig - Utility for module TCP/IP configuration* zip file.

Before starting the installation, it is recommended that you close all other applications.

Procedure

Installing IPconfig:

1. Unzip the *IPconfig - Utility for module TCP/IP configuration* zip file.
2. Double-click the*IPconfig Setup* application file.
3. The Setup IPconfig installer window appears.

   Click **Next** to begin installation.

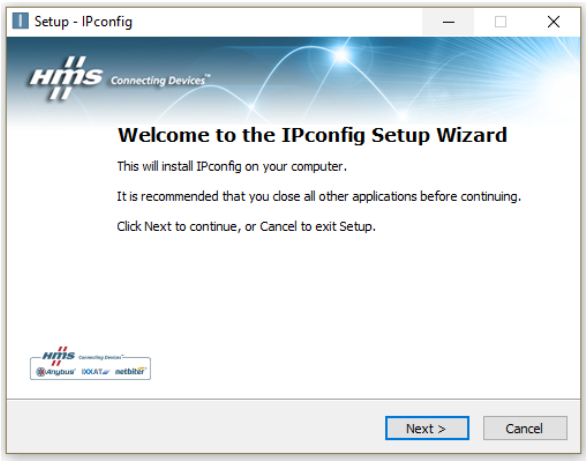


Fig. 4 IPconfig Setup Wizard

4. Follow the prompts in the IPconfig Setup Wizard to complete the installation.
5. Click **Finish**.

Result

→ IPconfig is installed on your computer.

4 Developing the .NET Application

4.1 How the Communication Works

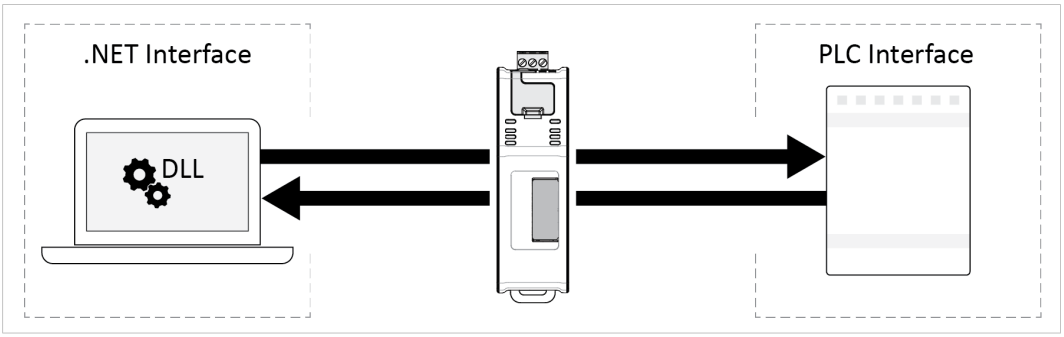


Fig. 5 Communication between the .NET Interface and the PLC Interface

The .NET Bridge transfers all communication sent between a PLC and a .NET application in a PC.

After a message has been sent, no Acknowledgment (ACK) is required before a new message can be sent.

No function block is required in the PLC application.

A DLL file, dynamic-link library, for the .NET Bridge is required in the .NET application.

4.2 .NET Programming

4.2.1 Importing References

Before You Begin

Only Microsoft Visual Studio is supported as the .NET development environment.

Ensure that you have access to the *AnybusNetBridge.dll* file.

The dynamic-link library for the .NET platform that handles the communication with the PLC application.

Visit www.anybus.com/support and download the *AnybusNetBridge.dll* file.

Procedure

1. In Visual Studio, create a .NET Framework project.

Importing the *AnybusNetBridge.dll* file:

2. In the **Solution Explorer**, select the project ► right-click and select **Add** ► **References** ► **Browse**.
3. Browse and select the *AnybusNetBridge.dll* file and click **Add**.
4. Click **OK** to add the file in the project.

4.2.2 Building the .NET Application

Below are some basic steps to getting started with developing .NET applications that are compatible with the .NET Bridge Streamer Mode.

For more information on developing .NET applications for the .NET Bridge Streamer Mode, refer to the code examples, videos and application notes with use cases available at www.anybus.com/support.

```
var dataStreamerBridgeBuilder = new HMS.AnybusNetBridge.DataStreamerBridgeBuilder( bridgeIPAddress );

int netToBridgeDataSendIntervalMilliseconds = 1000;
dataStreamerBridgeBuilder.DataSendInterval = netToBridgeDataSendIntervalMilliseconds;

using( var dataStreamer = dataStreamerBridgeBuilder.ToDataStreamer() )
{
    dataStreamer.ProcessDataSizeRequest += DataStreamer_ProcessDataSizeRequest;
    dataStreamer.BridgeStatusChanged    += DataStreamer_BridgeStatusChanged;
    dataStreamer.NetworkResetReceived   += DataStreamer_NetworkResetReceived;
    dataStreamer.CommunicationException += DataStreamer_CommunicationException;

    dataStreamer.Start();

    var transmitProcessData = Enumerable.Range( 0, TransmitProcessDataSize ).Select( item => Convert.ToByte( item ) ).ToArray();

    while( !cancellationToken.IsCancellationRequested )
    {
        if( dataStreamer.TryGetReceivedProcessData( out IList<byte> receivedProcessData, out int sequenceNumber, out bool isNewData ) )
        {
            if( isNewData )
            {
                Console.WriteLine( $"Received new data with sequence number {sequenceNumber}:" );
                Console.WriteLine( "[" + string.Join( " ", receivedProcessData.Select( item => item.ToString( "X02" ) ) ) + "]" );
            }
            else
            {
                Console.WriteLine( $"Received old data with sequence number {sequenceNumber}." );
            }
        }

        if( updateTransmitDataSignal.WaitOne( 0 ) )
        {
            for( int i = 0; i < transmitProcessData.Length; ++i )
            {
                transmitProcessData[ i ] += 42;
            }

            Console.WriteLine( "Sending new process data:" );
            Console.WriteLine( "[" + string.Join( " ", transmitProcessData.Select( item => item.ToString( "X02" ) ) ) + "]" );

            dataStreamer.SendProcessData( transmitProcessData );
        }
    }

    dataStreamer.Stop();
}
```

Fig. 6 .NET application for Data Streamer, Code Example

1. Create a `dataStreamerBridgeBuilder`, a *bridge builder*.

```
var dataStreamerBridgeBuilder =
new HMS.AnybusNetBridge.DataStreamerBridgeBuilder( bridgeIPAddress );
```

2. Configure the *bridge builder*.

> *There are more properties that can be configured than shown in this example.*

```
int netToBridgeDataSendIntervalMilliseconds = 1000;
dataStreamerBridgeBuilder.DataSendInterval =
netToBridgeDataSendIntervalMilliseconds;
```

3. Create an `IDataStreamer`.

```
var dataStreamer = dataStreamerBridgeBuilder.ToDataStreamer();
```

4. Add *event handlers*.

   This is optional, but recommended.

> *The event handler implementations are not shown for brevity.*

   The .NET application starts listening to various events that may occur.

```
dataStreamer.ProcessDataSizeRequest += DataStreamer_ProcessDataSizeRequest;
dataStreamer.BridgeStatusChanged   += DataStreamer_BridgeStatusChanged;
dataStreamer.NetworkResetReceived  += DataStreamer_NetworkResetReceived;
dataStreamer.CommunicationException += DataStreamer_CommunicationException;
```

5. Connect to the *bridge hardware*.

```
dataStreamer.Start();
```

6. Receive process data from the .NET Bridge hardware.

   Process data sent from the PLC to the .NET Bridge is forwarded to the .NET application.

```
dataStreamer.TryGetReceivedProcessData( out IList<byte> receivedProcessData,
out int sequenceNumber, out bool isNewData );
```

7. Send process data to the .NET Bridge hardware.

   Process data sent from the .NET application to the .NET Bridge is forwarded to the PLC.

```
dataStreamer.SendProcessData( transmitProcessData );
```

8. Disconnect from the .NET Bridge hardware.

```
dataStreamer.Stop();
```

4.3 PLC Simulation

4.3.1 About PLC Simulation

Anybus .NET Bridge Streamer Mode PLC Simulator simulates PLC applications.

The .NET programmer can simulate a PLC application when developing and testing the .NET application.

No access to the PLC application is needed during the simulation.

4.3.2 Setting up the .NET Bridge for PLC Simulation

Before you can start using the .NET Bridge PLC simulation you need to connect the .NET Bridge to the IT network and power.

Before You Begin

Ensure that the .NET Bridge IP settings are configured for the IT network.

For more information, refer to *TCP/IP Configuration, p. 33*.

Procedure

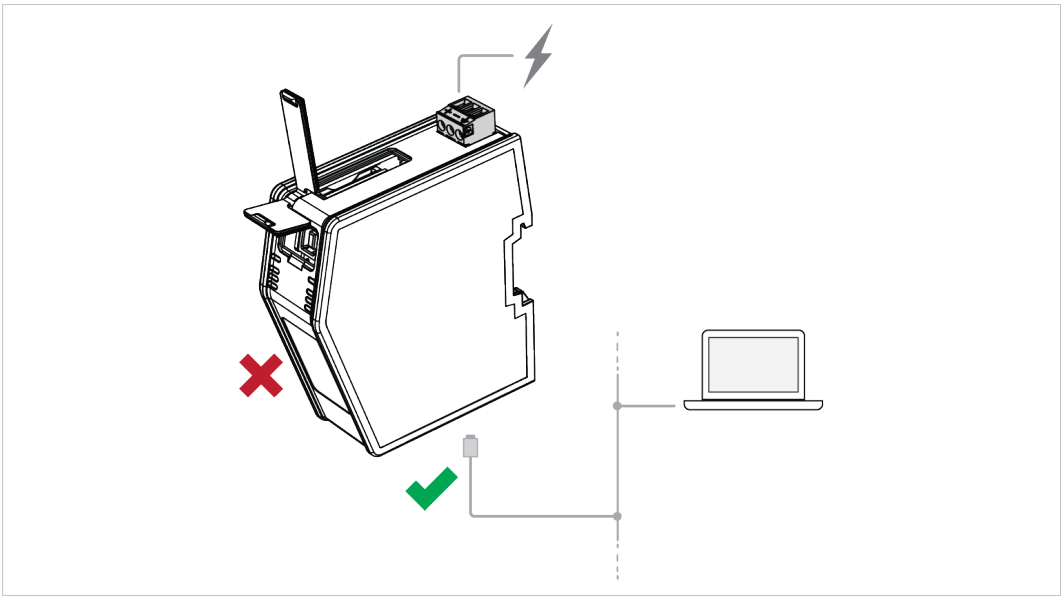


Fig. 7 .NET Bridge connected to IT network and power

1. Connect the .NET Bridge to the IT network via the IT Network Connectors, located at the bottom of the .NET Bridge.
2. Connect the .NET Bridge to power.

   For more information about the Power Connection, refer to *Connecting to Power, p. 28*.

Do not connect the .NET Bridge to the PLC.

LED indication:

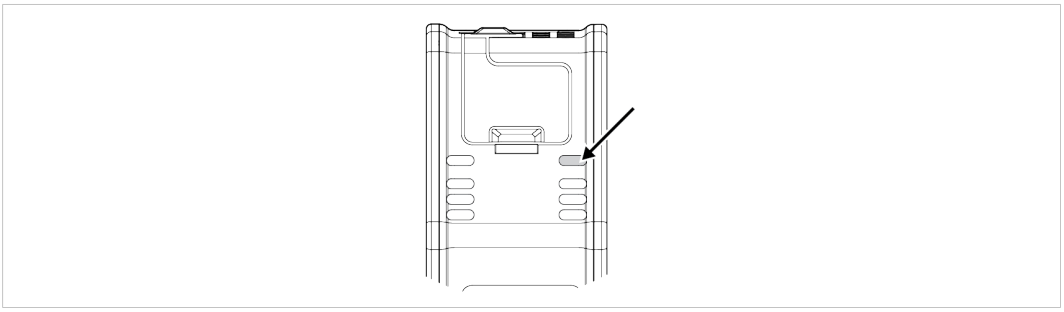


Fig. 8 IT Status LED

3. Check that the *IT Status LED* turn solid green.

4.3.3 Enable PLC Simulation

Before you can start the PLC simulation you must enable PLC simulation.

Before You Begin

Ensure that you have access to the .NET Bridge IP address on the IT network.

Procedure

Enable PLC Simulation:

1. In your web browser, type the .NET Bridge IP address and enter the Anybus .NET Bridge Web Interface start page.
2. In PLC simulation, check the **Enable PLC simulation** checkbox.
3. Click **Store settings**.

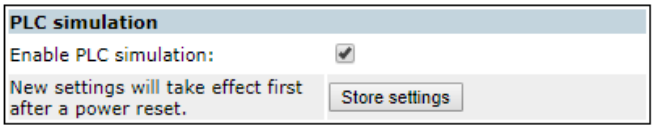


Fig. 9 Enable PLC simulation

4. For changes to take effect, restart the .NET Bridge.

LED indication:

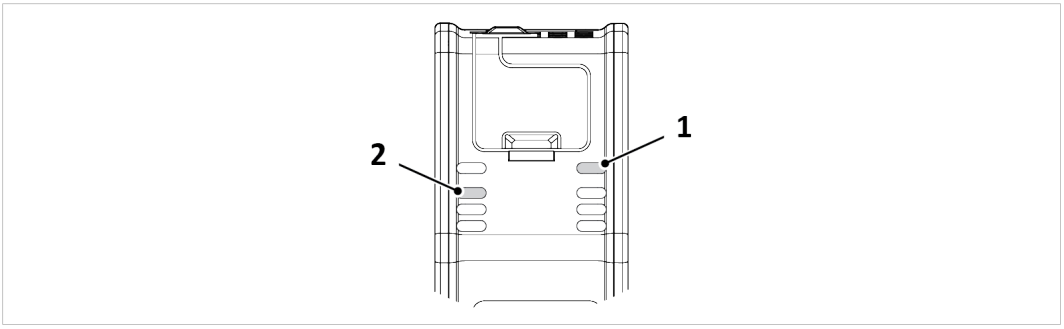


Fig. 10 IT Status LED (1) and SIM PLC Simulation LED (2)

5. Check that the *IT Status LED* (1) is solid green.
6. Check that the *SIM PLC Simulation LED* (2) is blinking green.

   For more information about the LED status indicators, refer to *LED Guide, p. 39*.

To Do Next

Simulation is now enabled, and you can proceed with the PLC simulation by starting the .NET application.

4.3.4 Starting the .NET Application

Before you can start the PLC simulation you must start the .NET application.

Procedure

In Visual Studio, start the .NET application:

1. Click **Start**, to start running the .NET application.

LED indication:

2. Check that the *IT Status LED* (1) is solid green.
3. Check that the *SIM PLC Simulation LED* (2) turn solid green.

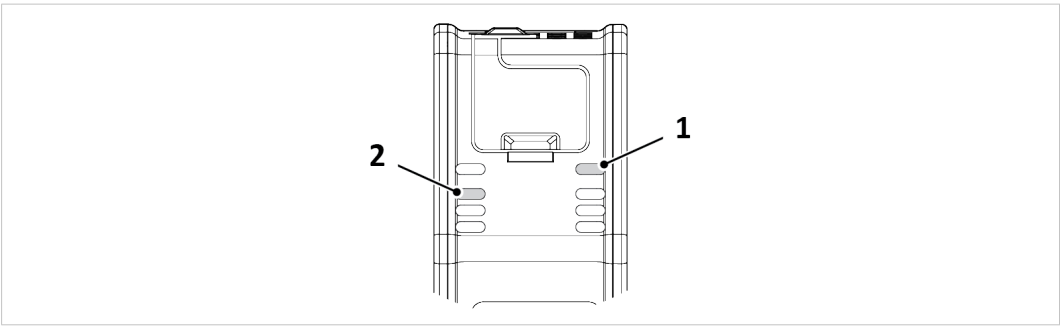


Fig. 11 IT Status LED (1) and SIM PLC Simulation LED (2)

To Do Next

The .NET application is now ready to start sending and receiving messages.

4.3.5 Connect PLC Simulator to .NET Bridge

Before You Begin

Ensure that:

- PLC simulation is enabled in the Anybus .NET Bridge Web Interface.
- you have access to the .NET Bridge IP address, on the IT network.
- the Anybus .NET Bridge Streamer Mode PLC Simulator is installed on your computer.
- the .NET application is running.

Procedure

Connect and Initialize PLC simulation:

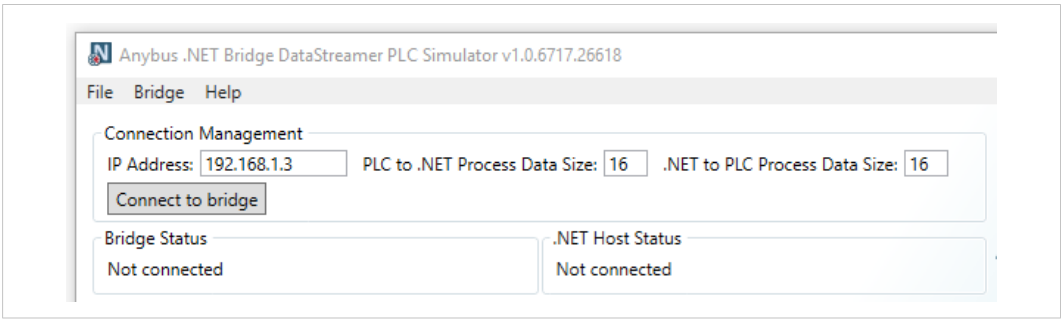


Fig. 12 Connect and Initialize the .NET Bridge

1. Open Anybus .NET Bridge Streamer Mode PLC Simulator.
2. In the Connection Management pane, enter the IP address to the .NET Bridge on the IT network.
3. In the Connection Management pane, enter the Data Size.

   The default data size values are 16 bytes.
4. To connect to the .NET Bridge, click **Connect to bridge**.
5. In the *Bridge Status* pane, verify that the status is *The simulation is enabled*.
6. Check that the .NET application is connected to the simulator:

   In the *.NET Host Status* pane, verify that the status is *A .NET host is connected to the bridge*.

4.3.6 Post and Receive Messages in PLC Simulator

Send messages from the simulated PLC to the .NET application and view the messages received from the .NET application.

Before You Begin

- Ensure that the .NET application is running.

In Anybus .NET Bridge Streamer Mode PLC Simulator, ensure that the:

- Bridge status is *The simulation is enabled*.
- .NET Host Status is *A .NET host is connected to the bridge*.

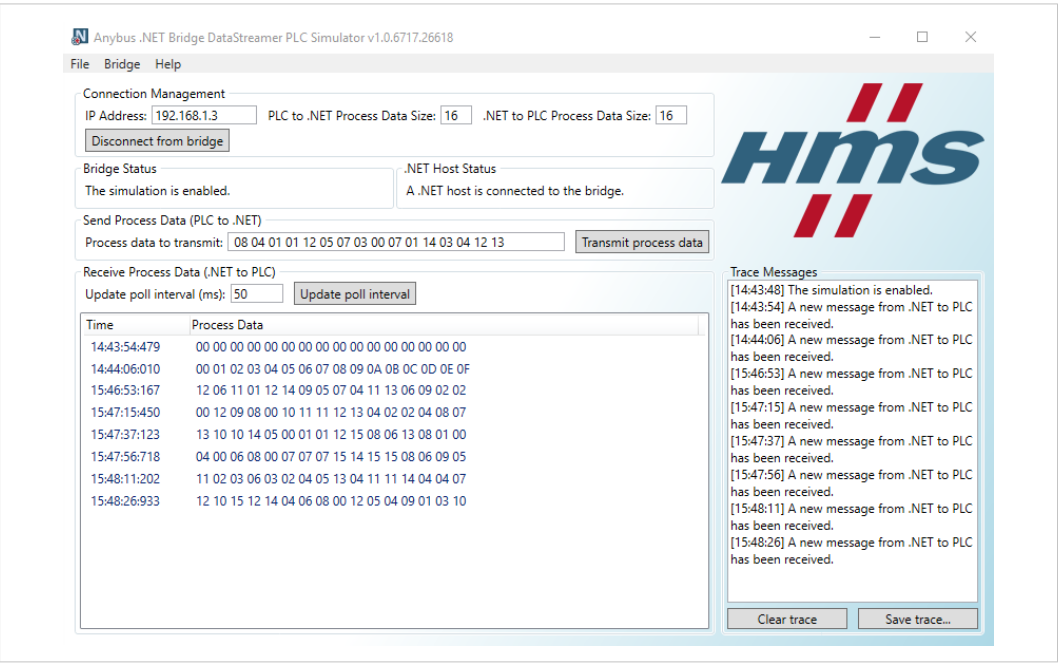


Fig. 13 Example, PLC Simulator main window

Post messages from the simulated PLC

Procedure

In the Send Process Data (PLC to .NET) pane you can send data from the simulated PLC application to the .NET application:

1. Enter the amount of data to transmit in hexadecimal numbers, in the *Process data to transmit* field.
2. To send data, click **Transmit process data**.

Result

→ The message is sent from the simulated PLC to the .NET application.

→ Verify that the message has been received by the .NET application.

Receive message in the simulated PLC

Procedure

In the Receive Process Data (.NET to PLC) pane:

Data sent from the .NET application to the simulated PLC application are listed in the *Receive Process Data (.NET to PLC)* pane.

The *poll interval* is how often the PLC simulator asks the .NET Bridge if it has received new process data from the .NET application.

To update the *poll interva*l, enter a value and press **Update poll interval**.

The default *poll interval* value is 50 ms.

5 Developing the PLC Application

5.1 .NET Simulation Startup

5.1.1 About .NET Simulation

Anybus .NET Bridge Streamer Mode .NET Simulator simulates the .NET application.

The PLC programmer can simulate the .NET application when developing and testing the PLC application.

No access to the .NET application is needed during the simulation.

5.1.2 Setting up the .NET Bridge for .NET Simulation

Before you can start using the .NET simulation you need to connect the .NET Bridge to IT network, PLC and power.

Before You Begin

Ensure that *PLC simulation* is disabled in the Anybus .NET Bridge Web Interface. The *Enable PLC simulation* checkbox must be unchecked.

Ensure that the .NET Bridge IP settings are configured for the IT network and the industrial network.

For more information, refer to *TCP/IP Configuration, p. 33*.

Procedure

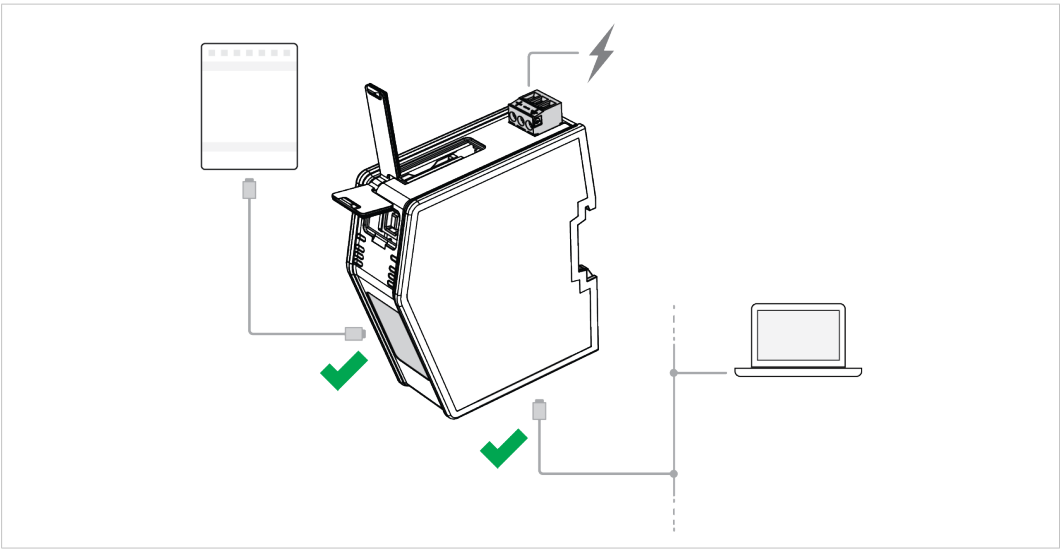


Fig. 14 .NET Bridge connected to IT network and PLC

1. Connect the .NET Bridge to the IT network via the IT Network Connectors, located at the bottom of the .NET Bridge.
2. Connect the .NET Bridge to the PLC via the Industrial Network Connectors, located at the front of the .NET Bridge.
3. Connect the .NET Bridge to power.

   For more information about the Power Connection, refer to *Connecting to Power, p. 28*.

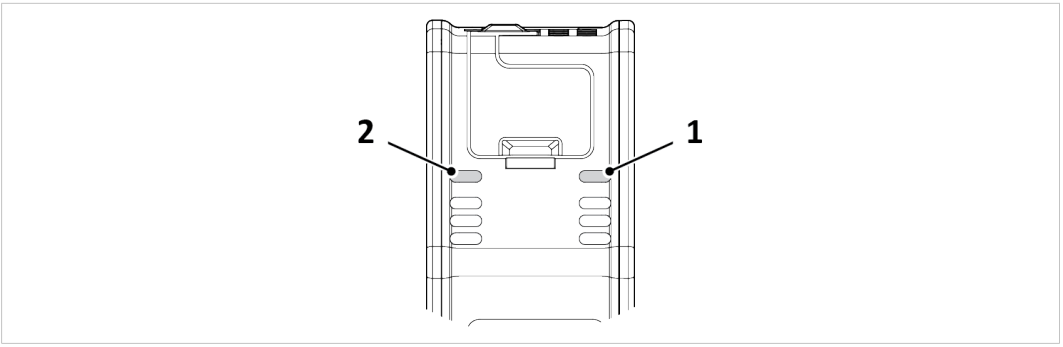


Fig. 15 IT Status LED (1) and OT Status LED (2)

LED indication:

4. Check that the *IT Status LED* (1) turn solid green.
5. Check that the *OT Status LED* (2) turn solid green.

5.1.3 Connect .NET Simulator to .NET Bridge

Connect and initialize the .NET Bridge in Anybus .NET Bridge Streamer Mode .NET Simulator.

Before You Begin

Ensure that:

- you have access to the .NET Bridge IP address, on the IT network.

Procedure

Connect and Initialize .NET simulation:

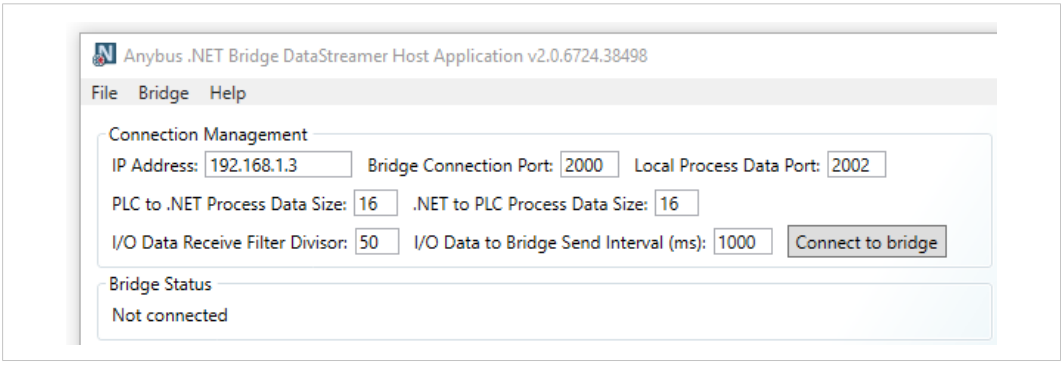


Fig. 16 Connect and Initialize the .NET Bridge

1. Open Anybus .NET Bridge Streamer Mode .NET Simulator.
2. In the Connection Management pane, enter the IP address to the .NET Bridge on the IT network.
3. In the Connection Management pane, enter the Bridge Connection Port, default port is 2000, and Local Process Data Port, the default port is 2002.
4. In the Connection Management pane, enter the Data Size for each direction. The default data size values are 16 bytes.
5. In the Connection Management pane, enter the I/O Data Receive Filter Divisor. The default value is 50.
6. In the Connection Management pane, enter the I/O Data to Bridge Send Interval. The default value is 1000 ms.
7. To connect to the .NET Bridge, click **Connect to bridge**.

8. In *Bridge Status* pane, verify that the status change to *The simulation is enabled*.

5.2 Running the .NET Simulation

5.2.1 Post and Receive Messages in .NET Simulator

Send messages from the simulated .NET application to the PLC application and view the messages received from the PLC application.

Before You Begin

- In Anybus .NET Bridge Streamer Mode .NET Simulator, ensure that the Bridge status is *A .NET host is connected to the bridge*.

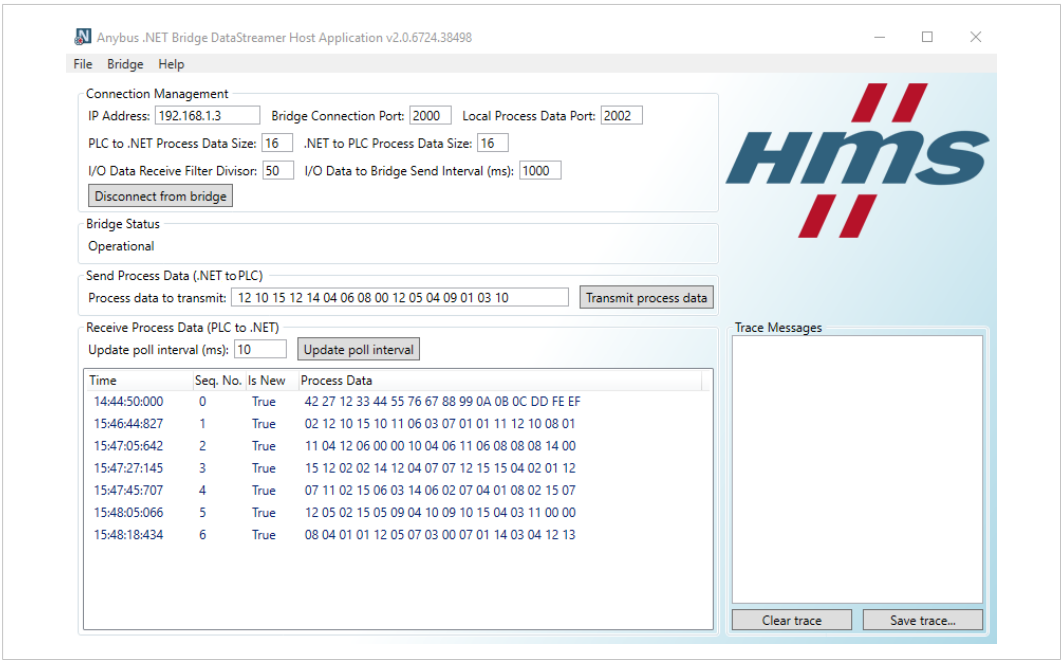


Fig. 17 Example, .NET Bridge Streamer Mode .NET Simulator main window

Post Messages from the simulated .NET Application

Procedure

In the Send Process Data (.NET to PLC) pane you can send data from the simulated the .NET application to the PLC:

1. Enter the amount of data to transmit in hexadecimal numbers, in the *Process data to transmit* field.
2. To send data, click **Transmit process data**.

Result

→ The message is sent from the simulated .NET application to the PLC.

► Open the PLC environment and verify that the message has been received.

Receive Message in simulated .NET application

Procedure

In the *Receive Process Data (PLC to .NET)* pane:

Data sent from the PLC application to the simulated .NET application are listed in the *Receive Process Data (PLC to .NET)* pane.

The *poll interval* is how often the .NET simulator asks the .NET Bridge if it has received new process data from the PLC.

To update the poll interval, enter a value and press **Update poll interval**.

The default poll interval value is 10 ms.

6 Installation

6.1 Product Overview

6.1.1 External Parts

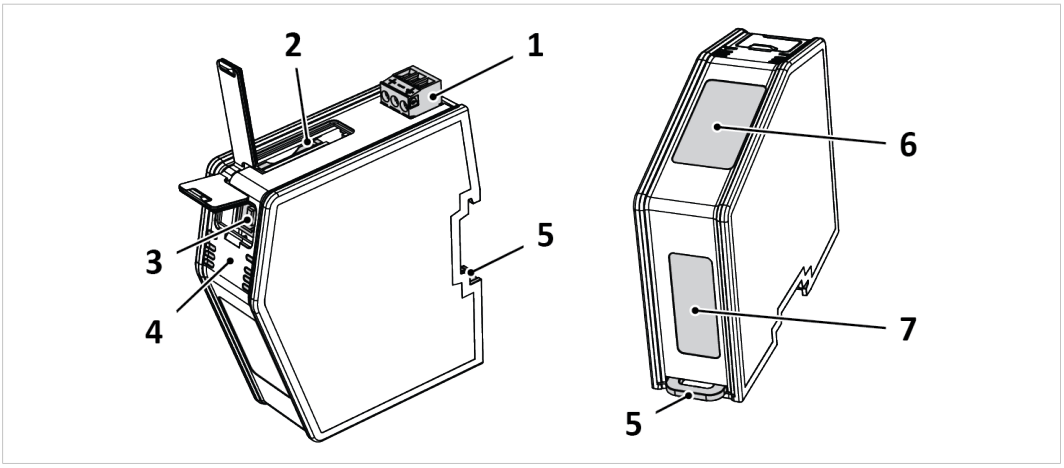


Fig. 18 .NET Bridge external parts

1. **Power Connector**
2. **SD Card Slot** (Currently not in use.)
3. **USB Port**
4. **Status LEDs**
5. **DIN Rail Connector**
6. **Industrial Network Connectors**
7. **IT Network Connectors**

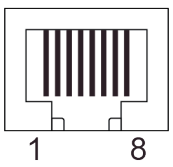
6.1.2 Network Connectors

RJ45 Connector

The RJ45 Connectors located at the front of the .NET Bridge are used to connect the .NET Bridge to an industrial network.

On the .NET Bridge for PROFIBUS there is a D-sub Connector at the front.

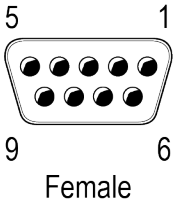
The RJ45 Connectors located at the bottom of the .NET Bridge are used to connect the .NET Bridge to a IT network.

Pin No.	Description	Connector
1	TX+	
2	TX-	
3	RX+	
6	RX-	
4, 5, 7, 8	Not connected	
Housing	Shield	

D-sub Connector

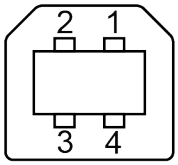
The D-sub Connector is used to connect the .NET Bridge to a PROFIBUS network.

On the .NET Bridge variants for EtherCAT, PROFINET and EtherNet/IP there is an RJ45 Connector at the front.

Pin No.	Description	PROFIBUS Connector DB9F
3	B-line	
4	RTS	
5	GND bus	
6	+5 V bus out	
8	A-line	
1, 2, 7, 9	Not connected	
Housing	PE (Protective Earth)	

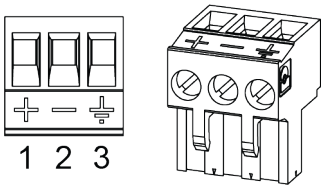
6.1.3 USB Port Type B

The USB Port Type B is used to connect a PC to the .NET Bridge to perform firmware upgrades.

Pin No.	Description	Connector
1	+5 V Input	
2	USBDM (USB communication signals)	
3	USBDP (USB communication signals)	
4	Signal GND	
Housing	Cable shield	

6.1.4 Power Connector

The Power Connector is used to connect the .NET Bridge to power and to Protective Earth (PE).

Pin No.	Description	Connector
1	+24 V DC -15% to +20%	
2	GND	
3	Protective Earth (PE)	

6.2 Mechanical Installation

6.2.1 DIN Rail Mounting Option

The .NET Bridge is designed to be mounted on a DIN rail.

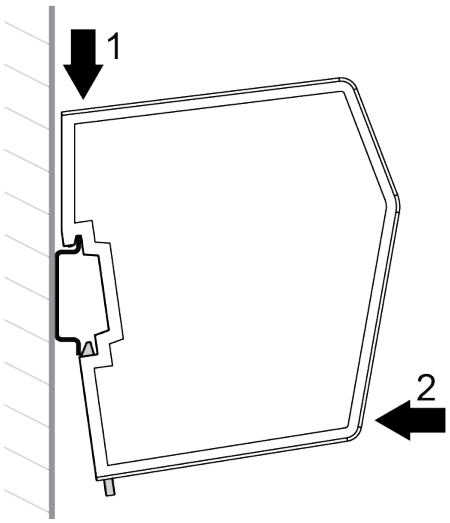
Before You Begin

> **!** The unit must be electrically grounded through the DIN rail for EMC compliance. Make sure that the unit is correctly mounted on the rail and that the rail is properly grounded.

> **!** When the .NET Bridge is used in an environment exposed to vibration, increased stability is required.
>
> Use the *Wall-Mount Accessory Kit* and mount the device on a wall instead of a DIN rail.
>
> The *Wall-Mount Accessory Kit* is ordered separately, please visit www.anybus.com for more information.

Procedure

Mount the .NET Bridge on DIN rail:



1. Hook the .NET Bridge DIN Rail Connector on the DIN rail.
2. Push the .NET Bridge against the DIN rail to make it snap on.

6.2.2 DIN-Rail Demounting

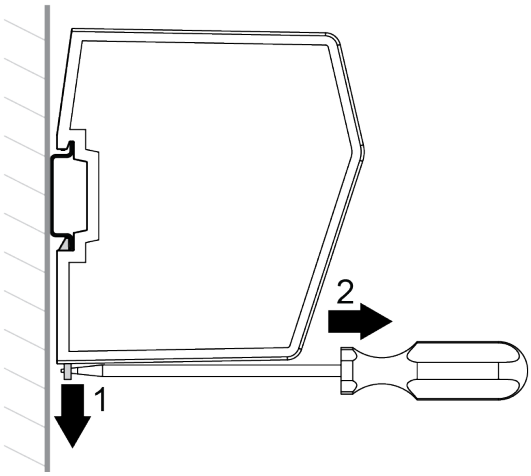
To remove the .NET Bridge from the DIN rail, do the following.

Before You Begin

Have a screwdriver available.

Procedure

Demount the .NET Bridge from the DIN rail:



1. Use the screwdriver to push the DIN Rail Fastening Mechanism down until it locks in the fixed and open position.
2. Unhook the .NET Bridge from the DIN rail.

To Do Next

Do not leave the module with the DIN Rail Fastening Mechanism in the fixed and open position. This may wear the fastening mechanism out and it cannot be used efficiently.

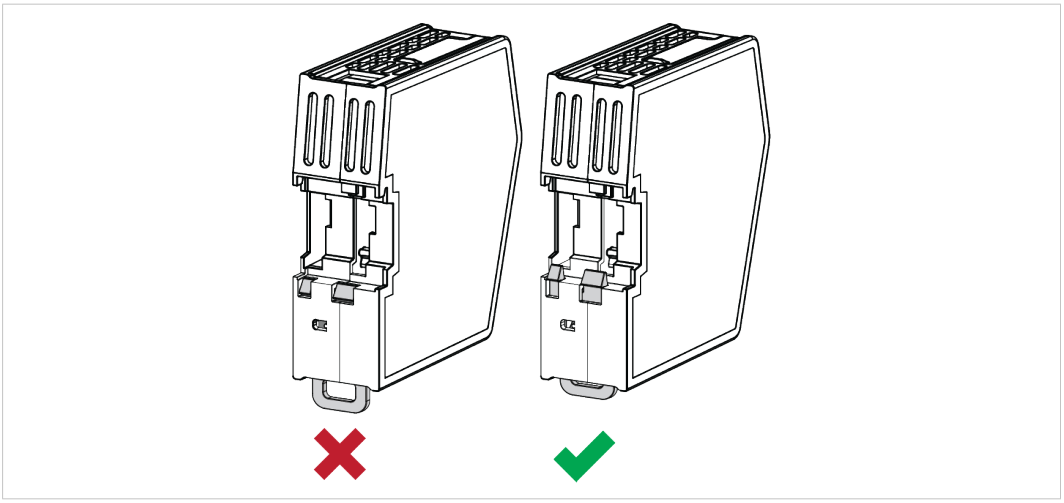


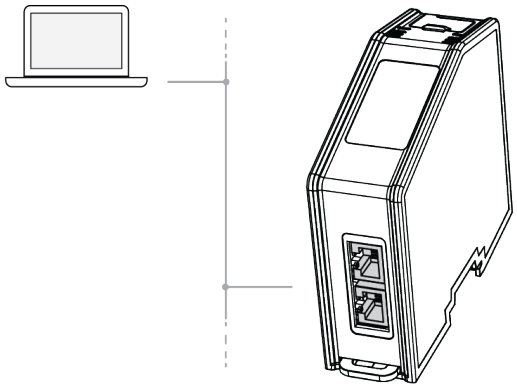
Fig. 19 DIN Rail Fastening Mechanism in open position and closed position

1. After demounting the module, push the DIN Rail Fastening Mechanism into the fixed and closed position.

6.2.3 Connecting to IT Network

Connect the .NET Bridge to the IT network, where the device with the .NET application is installed.

Procedure

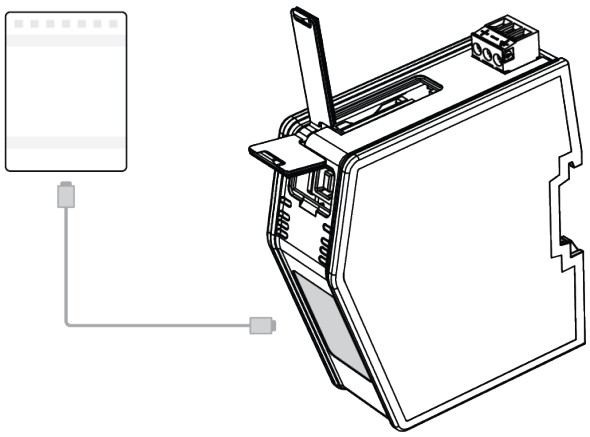


1. Connect the .NET Bridge to the IT network via the dual port switch.

6.2.4 Connecting to Industrial Network

Connect the .NET Bridge to the industrial network, where the PLC is installed.

Procedure



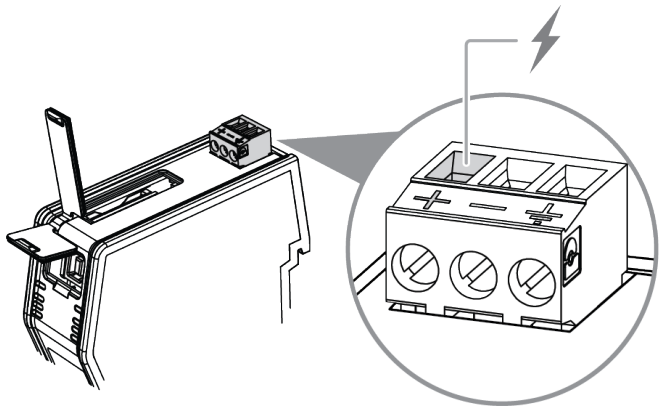
1. Connect the .NET Bridge to the PLC via the network connectors.

6.2.5 Connecting to Power

Connect the .NET Bridge to a power source.

> **!** Connecting power with reverse polarity or using the wrong type of power supply may damage the equipment. Make sure that the power supply is connected correctly and of the recommended type.

Procedure

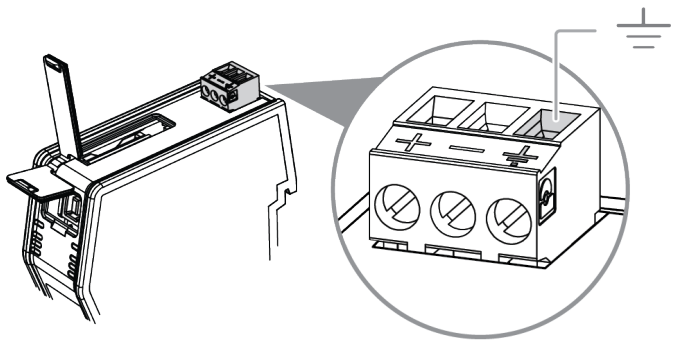


1. Connect the .NET Bridge to a power source via the +24 V DC Power Connector.

6.2.6 Connecting to Protective Earth

> **!** For compliance with EMC standards, the .NET Bridge must be connected to Protective Earth (PE).
>
> When the .NET Bridge is mounted on a wall, connect the device to Protective Earth (PE) via the Protective Earth Connector.
>
> When the .NET Bridge is mounted on a DIN rail, the device is electrically grounded through the DIN rail.

Procedure



1. Connect the .NET Bridge to Protective Earth (PE) via the Protective Earth Connector.

6.3 Network Settings

6.3.1 Configuring the .NET Bridge

The .NET Bridge must be configured before it can be connected to the IT network and the industrial network and before network settings, such as IP addresses, can be changed.

Before You Begin

The .NET Bridge comes with the default IP address *192.168.1.3* for the IT network.

Procedure

Configure the .NET Bridge:

1. Connect the .NET Bridge to the IT network,

   where the PC device with the .NET application, developed for the .NET Bridge, is installed.

2. Start the .NET application.

   → The .NET application connects to the .NET Bridge and configures it.

To Do Next

Continue with the .NET Bridge network settings.

6.3.2 IT Network Settings

The .NET Bridge comes with the default IP address *192.168.1.3* for the IT network.

Assign an IP address to the .NET Bridge, on the IT network where the .NET Bridge is installed.

Use IPconfig to assign the new IP address, refer to *TCP/IP Configuration, p. 33*

6.3.3 Industrial Network Settings

Depending on the industrial network type, assign an IP address or a node address to the .NET Bridge, on the industrial network where the .NET Bridge is installed.

Industrial Network	Settings	Software/Device/Web
EtherNet/IP	Assign an IP address to the .NET Bridge.	IPconfig
PROFINET	Assign an IP address to the .NET Bridge.	IPconfig **Tip:** You can use the PROFINET controller to assign the IP address.
PROFIBUS	Assign a PROFIBUS node address to the .NET Bridge.	Anybus .NET Bridge Web Interface
EtherCAT	Scan the EtherCAT network to locate the .NET Bridge using the EtherCAT PLC. The .NET Bridge is automatically assigned an node address by the PLC.	EtherCAT PLC

For information about IPconfig, refer to *IT Network Settings, p. 29*.

For information about Anybus .NET Bridge Web Interface, refer to *PROFIBUS Node Address Settings, p. 36*.

6.4 Check Operating Status

When the application development and installation are completed, verify that the .NET Bridge is in operation.

Before You Begin

Ensure that the:

- .NET application is installed on the PC device and is running.
- PLC application is downloaded on the PLC and is running.
- .NET Bridge IP settings are configured for the IT network.
- .NET Bridge IP/Node settings are configured for the industrial network.
- .NET Bridge is connected to the IT network and to the industrial network.

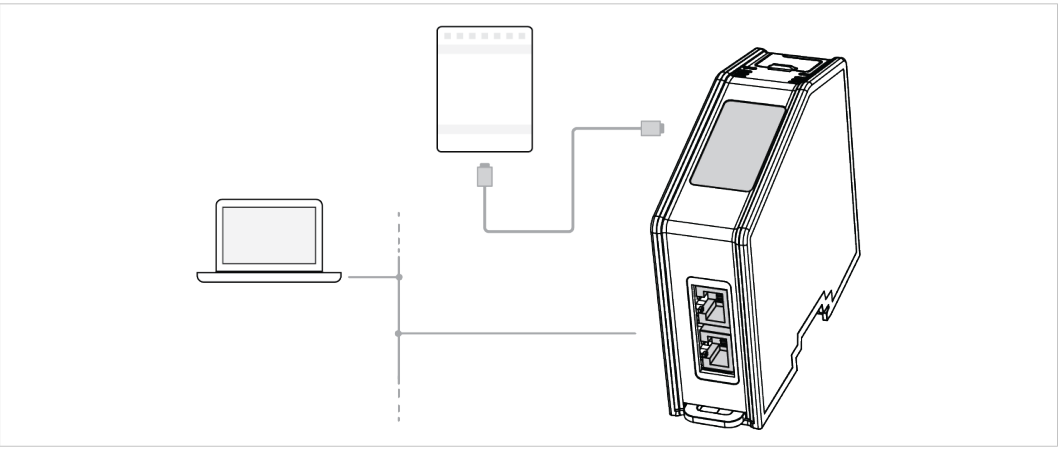


Fig. 20 .NET Bridge connected to IT network and industrial network

- .NET Bridge is connected to power.

Procedure

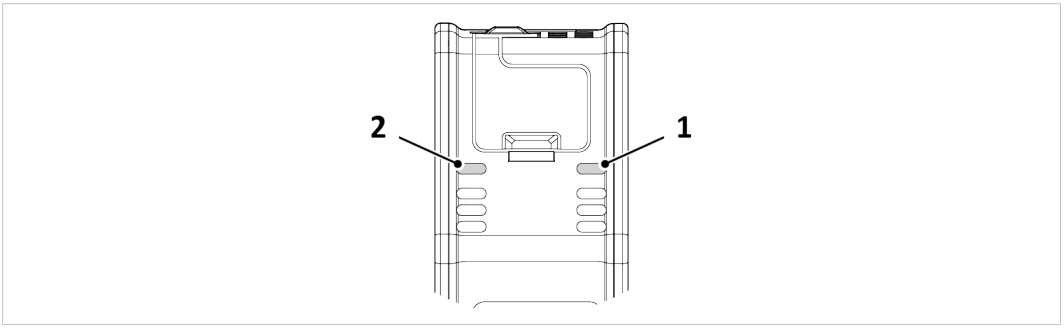


Fig. 21 IT Status LED (1) and OT Status LED (2)

IT network Operating Status

1. When the .NET Bridge is connected to the IT network:
   → The .NET application connects to the .NET Bridge.
   → The IT Status LED (1) start blinking slowly green.

Industrial network Operating Status

2. When the .NET Bridge is connected to the industrial network:
   → The PLC application connects to the .NET Bridge.
   → The IT Status LED (1) and the OT Status LED (2) turn solid green.

For more information about the LED status indicators, refer to *LED Guide, p. 39*

7 Technical Data

7.1 Technical Specifications

Order code	**AB9077–C — Anybus .NET Bridge PROFINET** **AB9078–C — Anybus .NET Bridge EtherNet/IP** **AB9079–C — Anybus .NET Bridge EtherCAT** **AB9071–C — Anybus .NET Bridge PROFIBUS**
Dimensions	110•35•101 mm, 4,33•1,38•3,98"
Weight	160 g, 0,35 lb
Operating temperature	-25 to +70 °C, -13 to +158 °F
Storage temperature	-40 to +85 °C, -13 to +185 °F
Relative Humidity	5-95% noncondensing
Protection class	IP20, NEMA rating 1
Mounting	DIN rail (35•7,5/15) or Wall mount
Current consumption	Typical: 150 mA @ 24 V DC
Power consumption	24 V DC +/- 10%
Certifications	Refer to datasheet at www.anybus.com/support.

A TCP/IP Configuration

A.1 Installing the IPconfig Utility

IPconfig is a Windows-based tool for configuration of TCP/IP settings in HMS devices. The tool will detect all compatible and active HMS devices on the local network.

1. Download IPconfig from www.anybus.com/support.
2. Unpack the contents of the zip archive and run the installer program.

A.2 Scanning for Connected Devices

When IPconfig is started it will automatically scan all available local networks for HMS devices. Detected devices will be listed in the main window. To refresh the list, click on **Scan**.

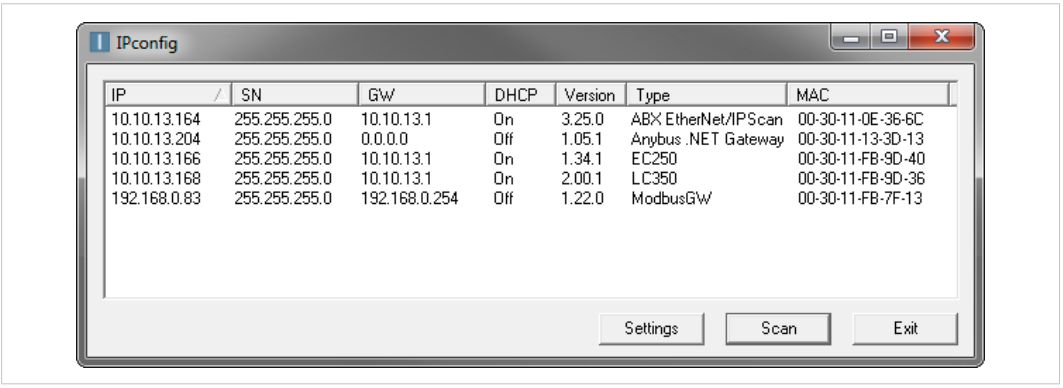


Fig. 22 IPconfig main window

IP	IP address of the device
SN	Subnet mask
GW	Default gateway
DHCP	Automatically managed IP configuration
Version	Firmware version
Type	Product name
MAC	Ethernet MAC address (System ID)

A.3 Ethernet Configuration

To change the IP settings for a device, double-click on the entry in the main window or right-click on it and select **Configuration**.

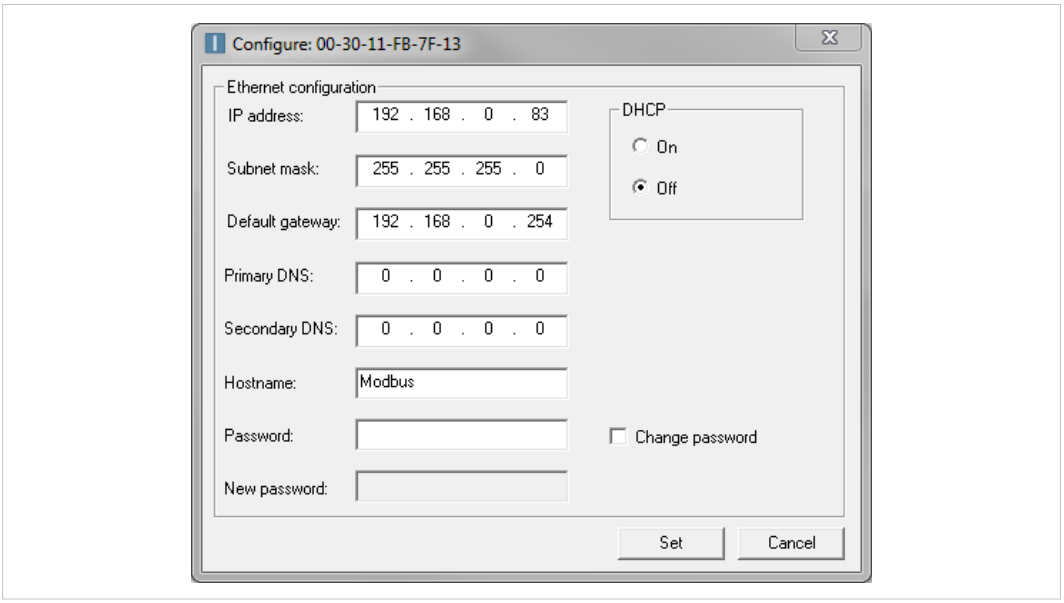


Fig. 23 Ethernet configuration

Enter static IP settings as required, or select DHCP if using dynamic IP addressing.

> Do not enable DHCP if there is no DHCP server available on the network.

You can add a name for the device in the **Hostname** field. Only characters a–z, A–Z, 0–9 and _ (underscore) are allowed.

The default password for changing IP settings is blank (no password). If a password has been set for the device you must enter it to be able to change the settings.

To set a new password, check the **Change password** box and enter the current password in the **Password** field, then enter the new password in the **New password** field.

> For security reasons the default password should always be changed.

Click on **Set** to save the new settings. The device will reboot automatically.

A.4 IPconfig Settings

Additional settings for IPconfig can be accessed by clicking on **Settings**.

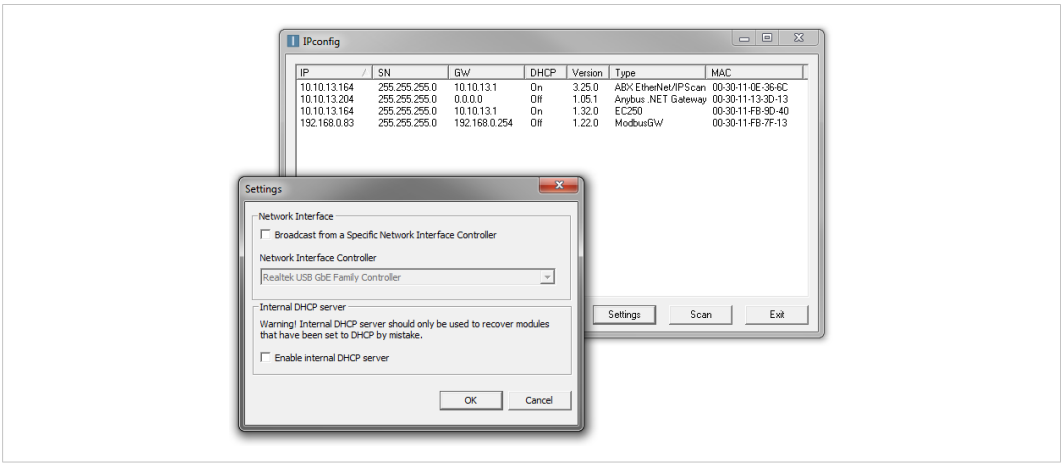


Fig. 24 IPconfig settings

Network Interface

Check this option to select a specific network interface to use when scanning for devices from a computer which has more than one interface. If this option is left unchecked, all available networks will be scanned.

Internal DHCP Server

If a device has been set to use DHCP but there is no DHCP server on the network, the device may not be detected by IPconfig. To recover access to the device an internal DHCP server in IPconfig can be temporarily activated:

1. Click the checkbox for **Internal DHCP Server**, then click **OK**. IPconfig will automatically refresh the scan and list the missing device in the main window.
2. Select the device and configure it to use static IP addressing instead of DHCP.
3. Disable the internal DHCP server.

> **!** Do not enable the internal DHCP server if there is already an active DHCP server on the network.

B PROFIBUS Node Address Settings

Use the Anybus .NET Bridge Web Interface to assign a PROFIBUS node address for the .NET Bridge, on the PROFIBUS network where the .NET Bridge is installed.

Before You Begin

Ensure that the .NET Bridge is connected to the same network as the device that is running the web browser.

Procedure

Assign a PROFIBUS node address:

1. In your web browser, type the .NET Bridge IP address and enter the Anybus .NET Bridge Web Interface start page.

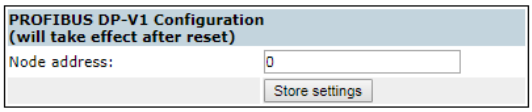


Fig. 25 PROFIBUS DP-V1 Configuration pane

2. In the PROFIBUS DP-V1 Configuration pane, enter the desired PROFIBUS node address.
3. Click **Store settings**.
4. Restart the .NET Bridge.

Result

→ The change take effect after restart.

C .NET Bridge Web Interface

The web interface for the .NET bridge is available via IP address 192.168.1.3, default setting.

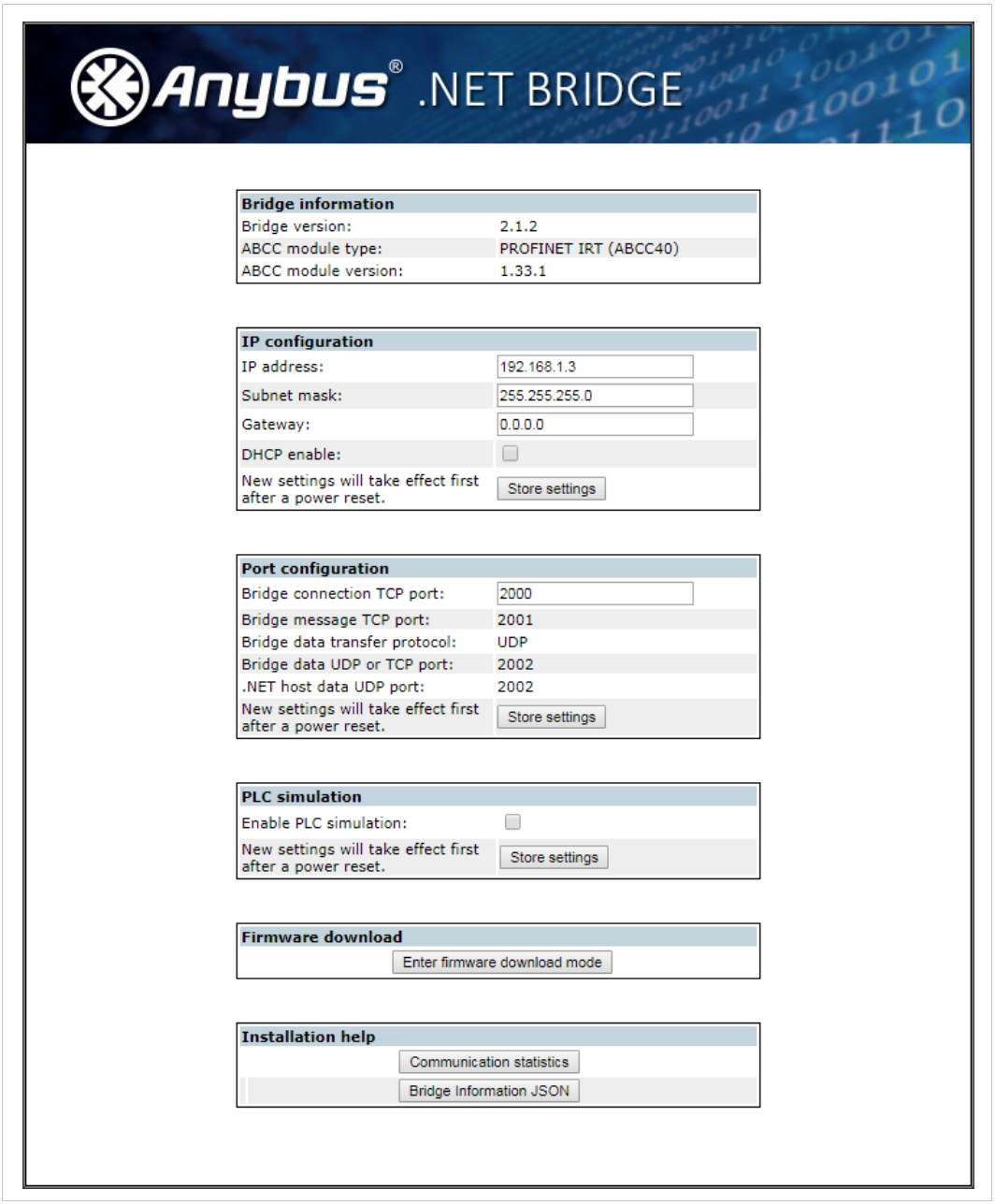


Fig. 26 The Anybus .NET Bridge web interface

The Anybus .NET Bridge Web Interface Functions	
Pane	**Description**
Bridge Information	Show the .NET Bridge version and other information vital for support.
IP configuration	Show current TCP/IP settings. IP configuration is editable.
Port configuration	Current port settings .NET Bridge connection TCP port is editable. .NET Bridge message TCP port and .NET Bridge data TCP/UDP port are configured via the .NET application.
PLC simulation	Checking the **Enable PLC simulation** checkbox enables the .NET Bridge to enter PLC simulation mode. The .NET Bridge must be reset before the changes take effect.
Network specific configuration	Available for certain networks. On PROFIBUS, this is where you set the node address. The .NET Bridge must be reset before the changes take effect
Firmware download	Pressing the **Enter firmware download mode** button makes the .NET Bridge ready for firmware download. The .NET Bridge must be connected via the USB interface. The .NET Bridge can not be accessed any other way, until it has been restarted. Firmware is downloaded using the Firmware Manager II from HMS.
Installation help	The **Communication statistics** button show detailed information about packets and communication.
	The **Bridge information JSON** button show information about the .NET Bridge embedded in a JSON script

The .NET Bridge TCP/IP settings can also be configured in IPconfig. Refer to TCP/IP Configuration, p. 33.

D LED Guide

D.1 LED Description

The .NET Bridge has eight LED status indicators located at the front.

The following tables describes what the different status indications mean.

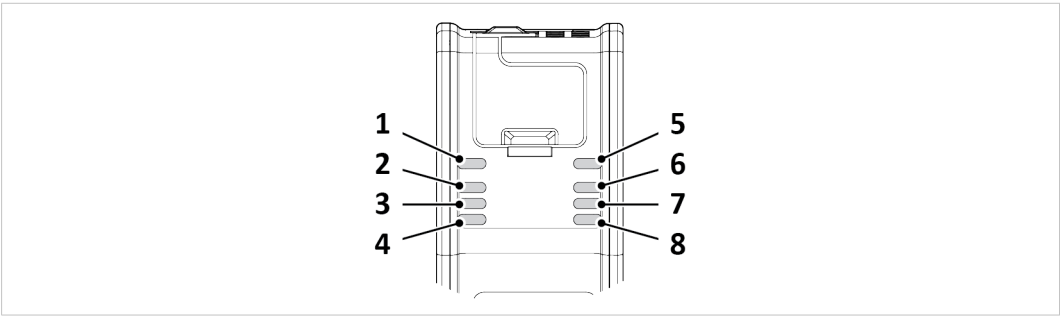


Fig. 27 .NET Bridge LED Status Indicators

General LED Status Indicators:			
LED	**Name**	**Indication**	**Description**
1	OT OT Status	Off	Power off
		Green	Connection to PLC
2	SIM PLC Simulation	Green Green, blinking Off	Simulation started Simulation enabled Simulation mode is off
3, 4	Network specific	-	Refer to information about network specific LED status indicators.
5	IT IT status	Off	Disconnected
		Green, fast blink	Connection to IT network
		Green, slow blink	Pre-operational. Connected to IT and OT network. No I/O data exchange.
		Green	Normal operation. I/O data exchange
		Red, blinking	If this LED and the Power LED both are red, this indicates a fatal error
6	PWR Power	Green	Power on
		Red	If this LED and the IT LED both are red, this indicates a fatal error
7, 8	LA1, LA2 Ethernet Link 1 and 2	Off	No link
		Flashing green	Receiving/transmitting Ethernet packets at 100 Mbit
		Flashing yellow	Activity, receiving/transmitting Ethernet packets at 10 Mbit
		Yellow	Boot up

Status LED indicator (3) and (4) indicates different activities depending on industrial networks.

EtherNet/IP specific LED Status Indicators:

LED	Name	Indication	Description
3	MS Module Status	Off Green Flashing green Red Flashing red	No power Controlled by a scanner in Run state Not configured, or scanner in Idle state Major fault Recoverable fault(s). Module is configured, but stored parameters differ from currently used parameters
4	NS Network Status	Off Green Flashing green Red Flashing red	No power or no IP address Online, connection established Online, no connection established Duplicate IP address, FATAL error on the OT network interface Connection timed out

EtherCAT specific LED Status Indicators:

LED	Name	Indication	Description
3	RUN	Off Green Blinking green Green, one flash Flickering Red	INIT OPERATIONAL PRE-OPERATIONAL SAFE-OPERATIONAL BOOT If RUN and ERR both turn red, this indicates a fatal event. Contact HMS support
4	ERR	Off Blinking red Red Red, one flash Red, two flashes Flickering	No error Invalid configuration Application controller failure. If RUN and ERR both turn red, this indicates a fatal event. Contact HMS support Unsolicited state change Sync manager watchdog timeout Booting error detected

PROFIBUS specific LED Status Indicators:

LED	Name	Indication	Description
3	OP Network Status	Off Green Green, flashing Red, one flash Red, two flashes	Not online Online, data exchange Online, clear Parametrization error PROFIBUS configuration error
4	ST Module Status	Off Green Green, flashing Red	Not initialized Initialized Diagnostic event Fatal error

PROFINET specific LED Status Indicators:

LED	Name	Indication	Description
3	NS Network Status	Off Green Green, one flash Green, three flashes Red Red, one flash Red, two flashes Red, three flashes	No connection Online (RUN): Connection established, IO controller in RUN state Online (STOP): Connection established, IO controller in STOP state Used to identify the slave Fatal error Device name error IP address error Configuration error
4	MS Module Status	Off Green Green, one flash Red Alternating red/green	Not initialized Normal operation Diagnostic event Fatal error Firmware update

D.2 Operation State LED Status

The .NET Bridge operation state is indicated by the LED status indicators.

State	LED Status	Description
Start up	IT: Off OT: Off	The .NET Bridge is connected to power and started but not yet connected to the .NET application or the PLC.
IT connected OT not connected	IT: Fast green blink OT: Off	The .NET Bridge is connected to the .NET application and has initialized the communication interface to the PLC, but there is no communication between the .NET Bridge and the PLC.
Pre-operational	IT: Slow green blink OT: Green	The .NET application and the PLC are both connected to the .NET Bridge. No I/O data exchange.
Operational	IT: Green OT: Green	The system is fully functional. I/O data exchange.
IT Disconnected OT connected	IT: Off OT: Green	The .NET application is disconnected from the .NET Bridge, after exchanging data in Operational state. The PLC side is still active.

E Firmware Update

E.1 Installing the Anybus Firmware Manager II

The Anybus Firmware Manager II is used to handle firmware in HMS devices.

Before You Begin

Visit www.anybus.com/support and download the *Firmware Manager II* zip file.

Before starting the installation, it is recommended that you close all other applications.

Procedure

Installing the Anybus Firmware Manager II:

1. Unzip the Anybus Firmware Manager Setup zip file.
2. Double-click the Firmware Manager Setup file.
3. The Setup Firmware Manager II installer window appears.

   Click **Next** to begin installation.

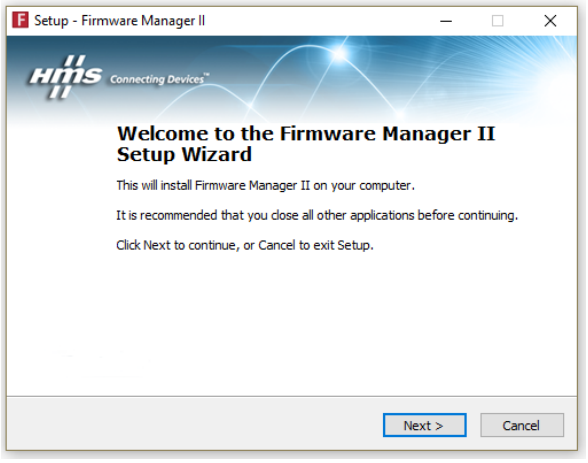


Fig. 28 Anybus Firmware Manager II Setup Wizard

4. Follow the prompts in the Firmware Manager II Setup Wizard to complete the installation.
5. Click **Finish**.

Result

The Anybus Firmware Manager II is installed on your computer.

E.2 .NET Bridge Firmware Update

Before You Begin

- Ensure that Firmware Manager II is installed on your computer.
- Ensure that you have access to the firmware update file.

  How to access firmware update files:

  – Firmware update files can be downloaded at www.anybus.com/support.

Procedure

Update .NET Bridge firmware:

1. Connect the .NET Bridge USB Port to your computer.
2. In your web browser, type the .NET Bridge IP address and enter the Anybus .NET Bridge Web Interface start page.
3. In the Firmware download pane, click **Enter firmware download mode**.

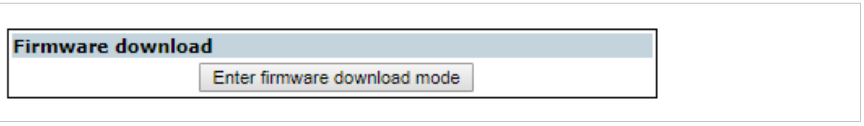


Fig. 29 Enter firmware download mode

   The .NET Bridge enters firmware download mode.

4. Open Firmware Manager II.

   ► Follow the instructions in the built-in help.

E.3 .NET Library Update

Update the .NET application with a new .NET library.

Before You Begin

- For information about Developing the .NET application, refer to *Developing the .NET Application, p. 10*.

Procedure

In Visual Studio:

1. Open the .NET application project.
2. Remove the reference to the current *AnybusNetBridge.dll* assembly.
3. Add a reference to the new *AnybusNetBridge.dll* assembly.
4. Make a new build of the .NET application.

Box 4126
300 04 Halmstad, Sweden

info@hms.se