

# How to Implement LAN Bypass for V3200 Series

Moxa Technical Support Team

[support@moxa.com](mailto:support@moxa.com)

## Contents

<b>1</b>	<b>Introduction</b> .....	<b>2</b>
<b>2</b>	<b>LAN Bypass Generations</b> .....	<b>2</b>
<b>3</b>	<b>LAN Bypass States</b> .....	<b>3</b>
<b>4</b>	<b>LAN Bypass Utility</b> .....	<b>3</b>
<b>4.1</b>	<b>System Prerequisites</b> .....	<b>3</b>
<b>4.2</b>	<b>Running the LAN Bypass Utility</b> .....	<b>4</b>
<b>4.3</b>	<b>LAN Bypass Modes</b> .....	<b>6</b>
<b>4.4</b>	<b>Checking the Firmware Version of the MCU</b> .....	<b>12</b>
<b>5</b>	<b>LAN Bypass APIs</b> .....	<b>13</b>
<b>5.1</b>	<b>RelayMode</b> .....	<b>13</b>
<b>5.2</b>	<b>WDTResetMode</b> .....	<b>15</b>
<b>5.3</b>	<b>WDTRelayMode</b> .....	<b>17</b>
<b>5.4</b>	<b>AppWDTMode</b> .....	<b>19</b>
<b>5.5</b>	<b>AppWDTKick</b> .....	<b>21</b>
<b>5.6</b>	<b>AppWDTResetMode</b> .....	<b>22</b>
<b>5.7</b>	<b>AppWDTRelayMode</b> .....	<b>24</b>
<b>5.8</b>	<b>PowerOffRelayMode</b> .....	<b>26</b>
<b>5.9</b>	<b>MCUFirmwareVersion</b> .....	<b>28</b>
<b>6</b>	<b>Conclusion</b> .....	<b>28</b>

Copyright © 2024 Moxa Inc.

Released on Feb 1, 2024

### About Moxa

Moxa is a leading provider of edge connectivity, industrial computing, and network infrastructure solutions for enabling connectivity for the Industrial Internet of Things. With 35 years of industry experience, Moxa has connected more than 82 million devices worldwide and has a distribution and service network that reaches customers in more than 80 countries. Moxa delivers lasting business value by empowering industry with reliable networks and sincere service for industrial communications infrastructures. Information about Moxa's solutions is available at [www.moxa.com](http://www.moxa.com).

### How to Contact Moxa

Tel: 1-714-528-6777

Fax: 1-714-528-6778

**MOXA**<sup>®</sup>

## 1 Introduction

System stability and availability is a key challenge for system integrators and users alike. This means effective asset management is more crucial than ever before. Asset management requires continuous status monitoring and the flexibility to take well-informed actions when devices malfunction or sometime even before that as part of preventative measures. Seamless data transmission is required to be able to effectively reap the benefits of digitalization. LAN bypass functions can take advantage of redundancy and other functions in the network to ensure seamless data transmission.

## 2 LAN Bypass Generations

### 1<sup>st</sup> Generation LAN Bypass

Usually controlled by a hardware jumper and hence provides a single mechanism (LAN bypass enabled or disabled) while the system is powered on. The hardware jumper is set manually.

### 2<sup>nd</sup> Generation LAN Bypass

Instead of just a hardware jumper to enable and disable LAN bypass, it is also possible to remotely change the latch relay link of bypass pairs. Another benefit is that you can set up each bypass pair with different settings (enable or disable).

### 3<sup>rd</sup> Generation LAN Bypass

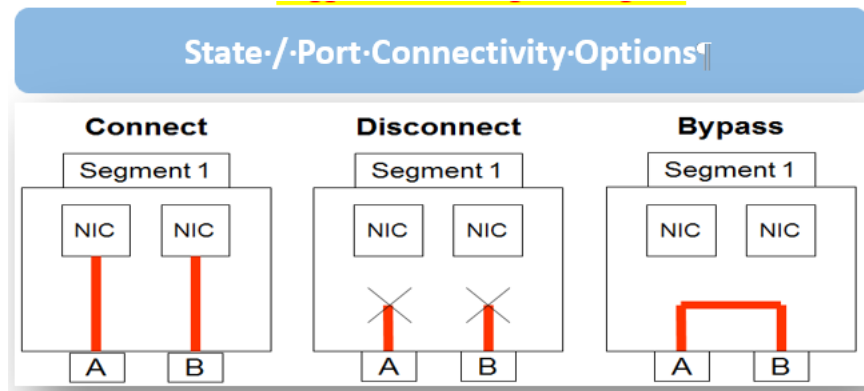
Uses a microcontroller, providing a more flexible bypass functionality for different system states (powered on or powered off).

This document describes the 3<sup>rd</sup> Generation LAN bypass set up and related SW APIs in the V3200 Series computers.

### 3 LAN Bypass States

Advanced LAN bypass mechanisms use direct low-latency connections between the NIC(s) and the LAN bypass MCU(s). The connections can have the following three states:

- **Connect:** A and B IO ports are connected to the NICs and is data transmitted through system normally
- **Disconnect:** A and B IO ports are neither connected to the NICs nor to each other, which means that data packets are blocked.
- **Bypass:** A and B IO ports are connected to each other to keep transmitting data without interruption even when a system device crashes or encounters a cyberattack.



- 
- Note**
- The default action setting for all options is "Connect", which means LAN Bypass is disabled.
  - A and B IO ports are the LAN 7 and LAN 8 ports on the V3200 series computer.
- 

## 4 LAN Bypass Utility

The V3200 Series computers come with a LAN bypass utility.

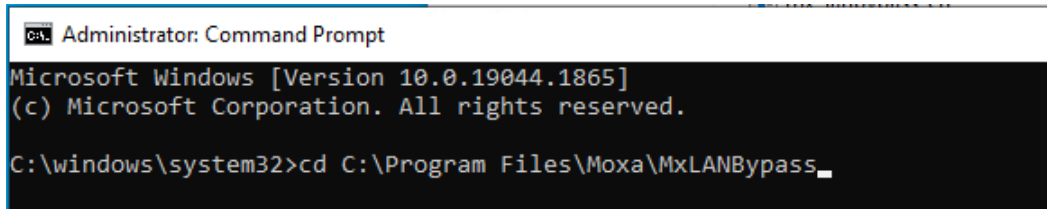
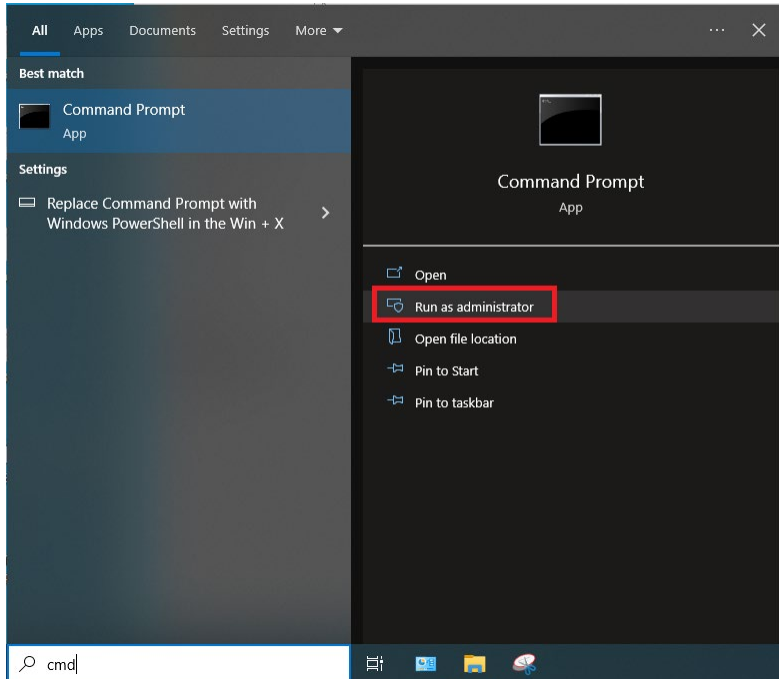
### 4.1 System Prerequisites

- Windows 10
- .Net Framework 4.8
- Microsoft Visual C++ 2015-2022 Redistributable x64

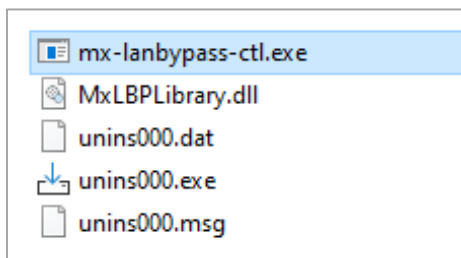
## 4.2 Running the LAN Bypass Utility

To run the utility, do the following:

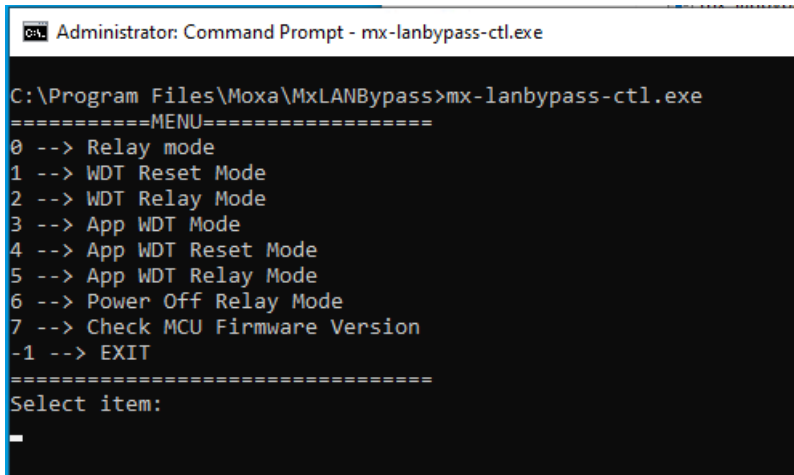
1. Open the command prompt as Administrator and type the following to access the folder: **C:\ProgramFiles\Moxa\MxLANBypass\**



2. Run the **mx-lanbypass-ctl.exe** from the **C:\Program Files\Moxa\MxLANBypass\** folder.



3. Type in the number corresponding to the function that you want to run.  
Refer to the LAN Bypass Modes section for a detailed description of the functions.



```
Administrator: Command Prompt - mx-lanbypass-ctl.exe

C:\Program Files\Moxa\MxLANBypass>mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
_
```

## 4.3 LAN Bypass Modes

### 4.3.1 Relay Mode

Use this mode to set the connection type between the target LANs.

After typing 0 in the prompt for **Relay mode**, select one of the following options:

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
0

Current Relay Mode is Disconnect.

=====
0 --> [Setting] Connect
1 --> [Setting] Disconnect
2 --> [Setting] ByPass
-1 --> Back to menu
=====
Select item:
2
Set OK.

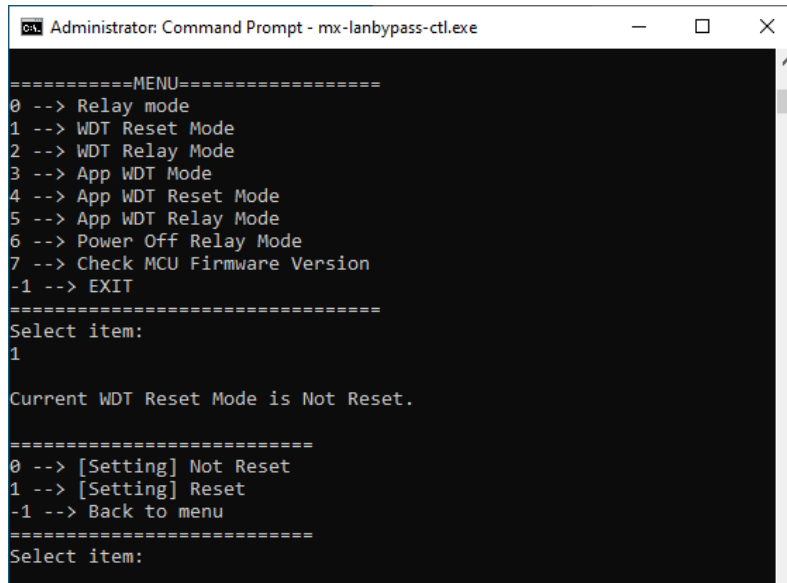
Current Relay Mode is By Pass.
```

### 4.3.2 WDT Reset Mode

Use this mode to indicate if the system needs to be reset after the watchdog is triggered.

After typing 1 for **WDT Reset Mode**, select one of the following options:

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog is triggered
1	Reset	The system will reset when the watchdog is triggered



```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
1

Current WDT Reset Mode is Not Reset.

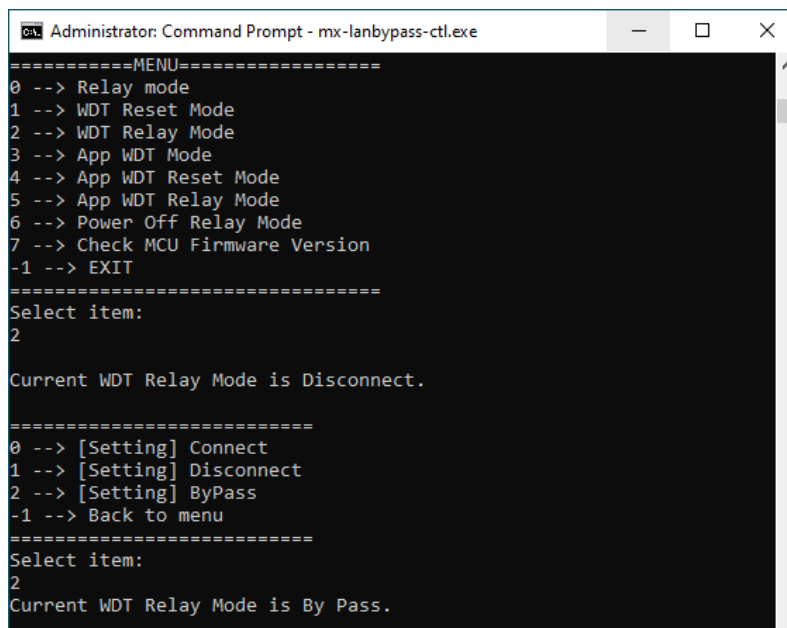
=====
0 --> [Setting] Not Reset
1 --> [Setting] Reset
-1 --> Back to menu
=====
Select item:
```

### 4.3.3 WDT Relay Mode

Use this mode to indicate the relay mode to switch to after the watchdog is triggered.

After typing 2 for **WDT Relay Mode**, select one of the following options:

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device



```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
2

Current WDT Relay Mode is Disconnect.

=====
0 --> [Setting] Connect
1 --> [Setting] Disconnect
2 --> [Setting] ByPass
-1 --> Back to menu
=====
Select item:
2
Current WDT Relay Mode is By Pass.
```



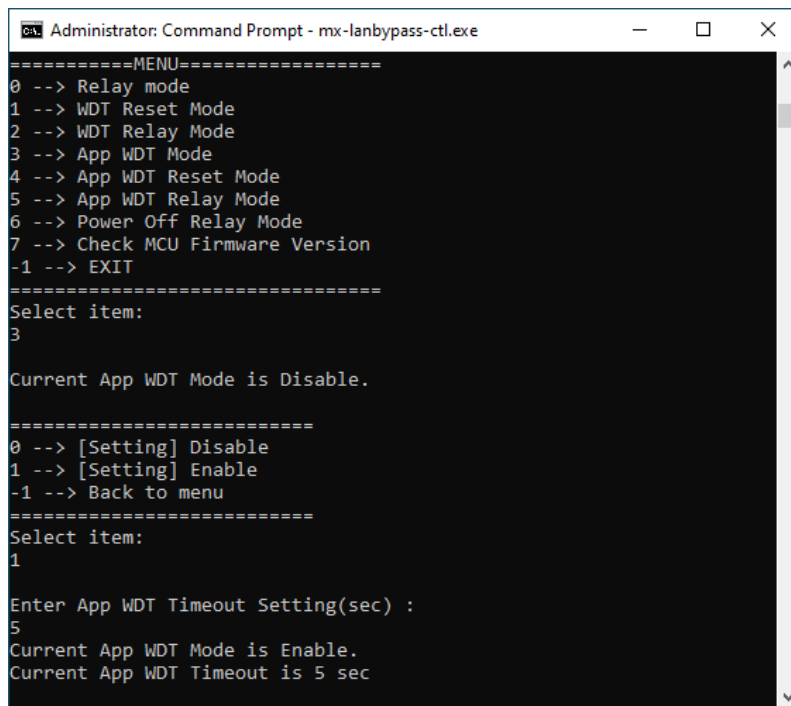
### 4.3.4 App WDT Mode

Use this mode to enable or disable the watchdog application. Activating the watchdog function is key to creating a trigger to activate LAN bypass when your application encounters issues or is unresponsive.

After typing 3 for **App WDT Mode**, select one of the following options:

Index	Mode	Description
0	Disable	Disable the app watchdog function
1	Enable	Enable the app watchdog function

When you select **Enable**, you must enter an **App WDT timeout** (5 to 15 sec) to set the interval for sending a watchdog “kick” the MCU.

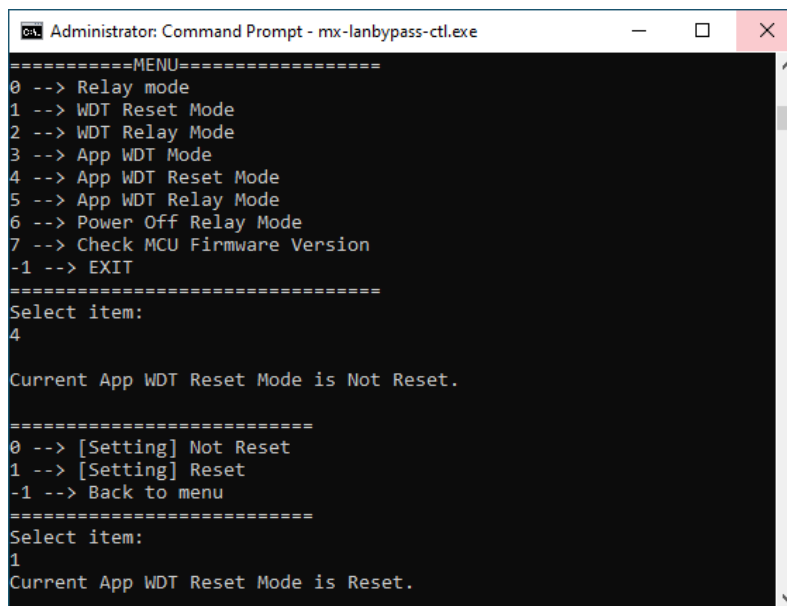


### 4.3.5 App WDT Reset Mode

Use this mode to indicate if the system needs to be reset after the app watchdog is triggered.

After typing 4 for **App WDT Reset Mode**, select one of the following options:

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog is triggered
1	Reset	The system will reset when the watchdog is triggered



```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
4

Current App WDT Reset Mode is Not Reset.

=====
0 --> [Setting] Not Reset
1 --> [Setting] Reset
-1 --> Back to menu
=====
Select item:
1

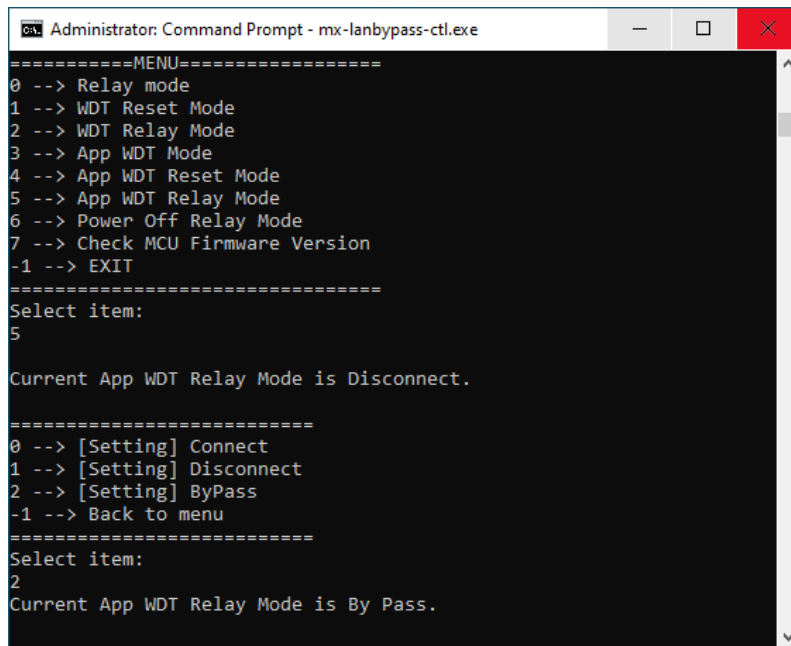
Current App WDT Reset Mode is Reset.
```

### 4.3.6 App WDT Relay Mode

Use this mode to indicate the relay mode to switch to after the app watchdog is triggered.

After typing 5 for **App WDT Relay Mode**, select one of the following options:

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device



```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
5

Current App WDT Relay Mode is Disconnect.

=====
0 --> [Setting] Connect
1 --> [Setting] Disconnect
2 --> [Setting] ByPass
-1 --> Back to menu
=====
Select item:
2

Current App WDT Relay Mode is By Pass.
```

### 4.3.7 Power Off Relay Mode

Use this mode to indicate the relay mode to switch to after the system is powered off.

After typing 6 for **Power Off Relay Mode**, select one of the following options:

Index	Mode	Description
0	Disconnect	Target LANs disconnected
1	Bypass	Target LANs are physically connected without passing through the network device

```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
6

Current Power Off Relay Mode is Disconnect.

=====
0 --> [Setting] Disconnect
1 --> [Setting] ByPass
-1 --> Back to menu
=====
Select item:
1

Current Power Off Relay Mode is By Pass.
```

### 4.4 Checking the Firmware Version of the MCU

Use this option to check the firmware version on the MCU. Type 7 in the prompt to select **Check MCU Firmware Version**. The firmware version information is displayed at the prompt.

```
Administrator: Command Prompt - mx-lanbypass-ctl.exe
=====MENU=====
0 --> Relay mode
1 --> WDT Reset Mode
2 --> WDT Relay Mode
3 --> App WDT Mode
4 --> App WDT Reset Mode
5 --> App WDT Relay Mode
6 --> Power Off Relay Mode
7 --> Check MCU Firmware Version
-1 --> EXIT
=====
Select item:
7

MCU Firmware Version is 1.0.0[S08]
```

## 5 LAN Bypass APIs

The V3200 Series computers also support LAN bypass APIs that can help you perform the same functions as the LAN Bypass utility. The API functions are described in detail in the following sections.

### 5.1 RelayMode

Use these to check and control the connection mode of the target LANs using a relay:

**GetRelayMode**

**SetRelayMode**

#### 5.1.1 GetRelayMode

##### Syntax

```
int GetRelayMode(out int relayMode)
```

##### Description

Gets the relay mode from the MCU.

##### Parameters

**relayMode** The relay mode; called using a reference variable.

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

##### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

##### Requirements

Name	Items
DLL	MxLBPLibrary.dll

### 5.1.2 SetRelayMode

#### Syntax

```
int SetRelayMode(int relayMode)
```

#### Description

Sets the relay mode.

#### Parameters

**relayMode** The relay mode; called using a reference variable.

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.2 WDTResetMode

Use these to indicate whether the system needs to be reset after the watchdog is triggered:

**GetWDTResetMode**

**SetWDTResetMode**

### 5.2.1 GetWDTResetMode

#### Syntax

```
int GetWDTResetMode(out int WDTResetMode)
```

#### Description

Gets the watchdog reset mode from the MCU.

#### Parameters

**WDTResetMode** The watchdog reset mode; called using a reference variable.

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog is triggered
1	Reset	The system will reset when the watchdog is triggered

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.2.2 SetWDTResetMode

### Syntax

```
int SetWDTResetMode(int WDTResetMode)
```

### Description

Sets the watchdog reset mode.

### Parameters

**WDTResetMode** The watchdog reset mode; called using a reference variable.

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog is triggered
1	Reset	The system will reset when the watchdog is triggered

### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

### Requirements

Name	Items
DLL	MxLBPLibrary.dll



## 5.3 WDTRelayMode

Use these to indicate which relay mode to switch to after the watchdog is triggered:

`GetWDTRelayMode`

`SetWDTRelayMode`

### 5.3.1 GetWDTRelayMode

#### Syntax

```
int GetWDTRelayMode(out int WDTRelayMode)
```

#### Description

Gets the watchdog relay mode from the MCU.

#### Parameters

**WDTRelayMode** The watchdog relay mode; called using a reference variable.

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

### 5.3.2 SetWDTRelayMode

#### Syntax

```
int SetWDTRelayMode(int WDTRelayMode)
```

#### Description

Sets the watchdog relay mode.

#### Parameters

**WDTRelayMode** The watchdog relay mode; called using a reference variable.

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.4 AppWDTMode

Use these to determine whether to enable the watchdog application:

**GetAppWDTMode**

**SetAppWDTMode**

An **AppWDTKick** needs to be sent to the MCU within the timeout to detect if the application is running properly. If there are errors in the application and the MCU does not receive the **AppWDTKick** within the timeout, the watchdog application is triggered.

### 5.4.1 GetAppWDTMode

#### Syntax

```
int GetAppWDTMode(out int appWDTMode, out int timeout)
```

#### Description

Gets the mode of the watchdog application from the MCU.

#### Parameters

**appWDTMode** The watchdog application mode; called using a reference variable.

Index	Mode	Description
0	Disable	Disable the watchdog application
1	Enable	Enable the watchdog application

**Timeout** The timeout in seconds (specify a value between 5 and 15)

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.4.2 SetAppWDTMode

### Syntax

```
int SetAppWDTMode(int appWDTMode, int timeout)
```

### Description

Sets app watchdog mode.

### Parameters

**appWDTMode** The watchdog application mode; called using a reference variable.

Index	Mode	Description
0	Disable	Disable the watchdog application
1	Enable	Enable the watchdog application

**Timeout** The timeout in seconds (specify a value between 5 and 15)

### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.5 AppWDTKick

An **AppWDTKick** is sent to the MCU to determine if the watchdog application is running properly.

**SendAppWDTKick**

### 5.5.1 SendAppWDTKick

#### Syntax

```
int SendAppWDTKick()
```

#### Description

Sends an AppWDTKick to the MCU.

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.6 AppWDTRResetMode

Use these to indicate if the system needs to be reset after the watchdog application is triggered:

`GetAppWDTRResetMode`

`SetAppWDTRResetMode`

### 5.6.1 GetAppWDTRResetMode

#### Syntax

```
int GetAppWDTRResetMode(out int appWDTRResetMode)
```

#### Description

Gets the watchdog application reset mode from the MCU.

#### Parameters

`appWDTRResetMode` The watchdog application reset mode; called using a reference variable.

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog application is triggered
1	Reset	The system will reset when the watchdog application is triggered

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.6.2 SetAppWDTRetCode

### Syntax

```
int SetAppWDTRetCode(int appWDTRetCode)
```

### Description

Sets the watchdog application reset mode in the MCU.

### Parameters

<b>appWDTRetCode</b>	The watchdog application reset mode; called using a reference variable.
----------------------	---

Index	Mode	Description
0	Not Reset	The system will not reset when the watchdog application is triggered
1	Reset	The system will reset when the watchdog application is triggered

### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.7 AppWDTRelayMode

Use these to indicate the relay mode to switch to after the watchdog application is triggered:

`GetAppWDTRelayMode`

`SetAppWDTRelayMode`

### 5.7.1 GetAppWDTRelayMode

#### Syntax

```
int GetAppWDTRelayMode(out int appWDTRelayMode)
```

#### Description

Gets the watchdog application relay mode from the MCU.

#### Parameters

<code>appWDTRelayMode</code>	The relay mode of watchdog application; called using a reference variable.
------------------------------	--

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll



## 5.7.2 SetAppWDTRelayMode

### Syntax

```
int SetAppWDTRelayMode(int appWDTRelayMode)
```

### Description

Sets the watchdog application relay mode in the MCU.

### Parameters

**appWDTRelayMode** The relay mode of watchdog application; called using a reference variable.

Index	Mode	Description
0	Connect	Target LANs connected
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.8 PowerOffRelayMode

Use these to indicate the relay mode to switch to after the system is powered off:

`GetPowerOffRelayMode`

`SetPowerOffRelayMode`

### 5.8.1 GetPowerOffRelayMode

#### Syntax

```
int GetPowerOffRelayMode(out int powerOffRelayMode)
```

#### Description

Gets the power off relay mode from the MCU.

#### Parameters

**powerOffRelayMode** The power off relay mode; called using a reference variable.

Index	Mode	Description
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.8.2 SetPowerOffRelayMode

### Syntax

```
int SetPowerOffRelayMode(int powerOffRelayMode)
```

### Description

Sets the power off relay mode in the MCU.

### Parameters

**powerOffRelayMode** The power off relay mode; called using a reference variable.

Index	Mode	Description
1	Disconnect	Target LANs disconnected
2	Bypass	Target LANs are physically connected without passing through the network device

### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 5.9 MCUFirmwareVersion

Use this to get the firmware version of the MCU:

`GetMCUFirmwareVersion`

### 5.9.1 GetMCUFirmwareVersion

#### Syntax

```
int GetMCUFirmwareVersion(out string version)
```

#### Description

Gets the firmware version from the MCU.

#### Parameters

**version** The string containing the MCU firmware version (e.g., 1.0.0[S00]); called using a reference variable.

#### Return Value

If the operation is completed successfully, the return value is 0. If the operation fails, the return value is 1.

#### Requirements

Name	Items
DLL	MxLBPLibrary.dll

## 6 Conclusion

3rd generation bypass functions have become indispensable in LAN bypass control. Even when your system/application shuts down unexpectedly, LAN Bypass functions can be configured for uninterrupted data transmission in critical applications to maximize network uptime. These functions play a critical role when a device acts as a security gateway between different network segments.