

# Windows 10 IoT Enterprise LTSC 2021 (21H2) User Manual

---

Version 1.0, June 2022

[www.moxa.com/products](http://www.moxa.com/products)

**MOXA**<sup>®</sup>

© 2022 Moxa Inc. All rights reserved.

# Windows 10 IoT Enterprise LTSC 2021 (21H2) User Manual

The software described in this manual is furnished under a license agreement and may be used only in accordance with the terms of that agreement.

## Copyright Notice

© 2022 Moxa Inc. All rights reserved.

## Trademarks

The MOXA logo is a registered trademark of Moxa Inc.  
All other trademarks or registered marks in this manual belong to their respective manufacturers.

## Disclaimer

- Information in this document is subject to change without notice and does not represent a commitment on the part of Moxa.
- Moxa provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Moxa reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.
- Information provided in this manual is intended to be accurate and reliable. However, Moxa assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.
- This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

## Technical Support Contact Information

[www.moxa.com/support](http://www.moxa.com/support)

# Table of Contents

<b>1. Introduction</b>	<b>5</b>
Applicable Series	5
Moxa Windows	5
<b>2. System Initialization</b>	<b>6</b>
Initializing User Settings	6
Initializing the System	10
<b>3. BitLocker</b>	<b>11</b>
Enabling the BitLocker	11
Disabling the BitLocker	14
<b>4. Configuring RAID</b>	<b>17</b>
Changing the RAID Mode	17
Creating a RAID Disk in BIOS	20
Replacing a Disk	24
Removing a RAID Volume From the BIOS	27
<b>5. Teaming</b>	<b>29</b>
Creating an Intel® Net Team	29
Modifying an Intel® Net Team Member	32
Adding an Intel® Net Team Member	32
Removing an Intel® Net Team Member	34
Removing an Intel® Net Team	36
<b>6. Unified Write Filter</b>	<b>37</b>
Turning on UWF on a Running PC	37
Installing UWF Using WMI	39
<b>7. Moxa IO Controller Utility</b>	<b>40</b>
Setting the DIO Status	40
Setting the UART Mode	41
Setting the Relay Status	42
Setting the LED Status	43
Setting the SIM Status	44
Setting PCIE Slot Power Status	45
Setting PCIE Reset Pin Status	46
Setting M.2 B Key Socket 5G Module Power Status	47
<b>8. Moxa Serial Interface Utility</b>	<b>48</b>
Setting the Serial Port Mode	48
<b>9. IO Control API</b>	<b>50</b>
Downloading the API	50
Supported Series	51
mxdgio	52
GetDinStatus	52
GetDoutStatus	53
SetDoutStatus	53
mxsp	54
GetUartMode	54
SetUartMode	55
mxrelay	56
GetRelayData	56
SetRelayData	57
mxled	58
GetLedData	58
SetLedData	59
mxsim	60
GetSIMSlot	60
SetSIMSlot	61
mxpcie	62
GetPCIESlotStatus	62
SetPCIESlotStatus	63
SetPCIESlotStatusWithReset	64
mxpciereset	65

GetRESETSlotStatus .....	65
SetRESETSlotStatus .....	66
mx5Gpower .....	67
GetPowerStatus.....	67
SetPowerStatus .....	68
ResetModule .....	69
mxwdg.....	70
mxwdg_open .....	70
mxwdg_refresh .....	70
mxwdg_close.....	71
<b>10. Windows Recovery.....</b>	<b>72</b>
Preparing the USB device .....	72
Booting From the USB Disk.....	75
System Image Backup.....	76
Restoring the System From a Backup .....	80

# 1. Introduction

---

This Windows 10 IoT Enterprise LTSC 2021 (21H2) user manual is applicable to Moxa's x86-based computers listed below and covers the complete set of instructions for these series. Detailed instructions on configuring advanced settings are covered in following chapters of the manual. Before referring to sections in these chapters, confirm that the hardware specification of your computer model supports the functions/settings covered therein.

## Applicable Series

- **DA-681C Series**
- **DA-682C Series**
- **DA-820C Series**
- **V2403C Series**
- **V2406C Series**
- **MC-1200 Series**
- **MC-3201 Series**

## Moxa Windows

Moxa computers are integrated with Windows drivers and I/O controller utilities based on the Microsoft Windows up-to-date version so that you can use the most compatible hardware-software combination in your application field.

## 2. System Initialization

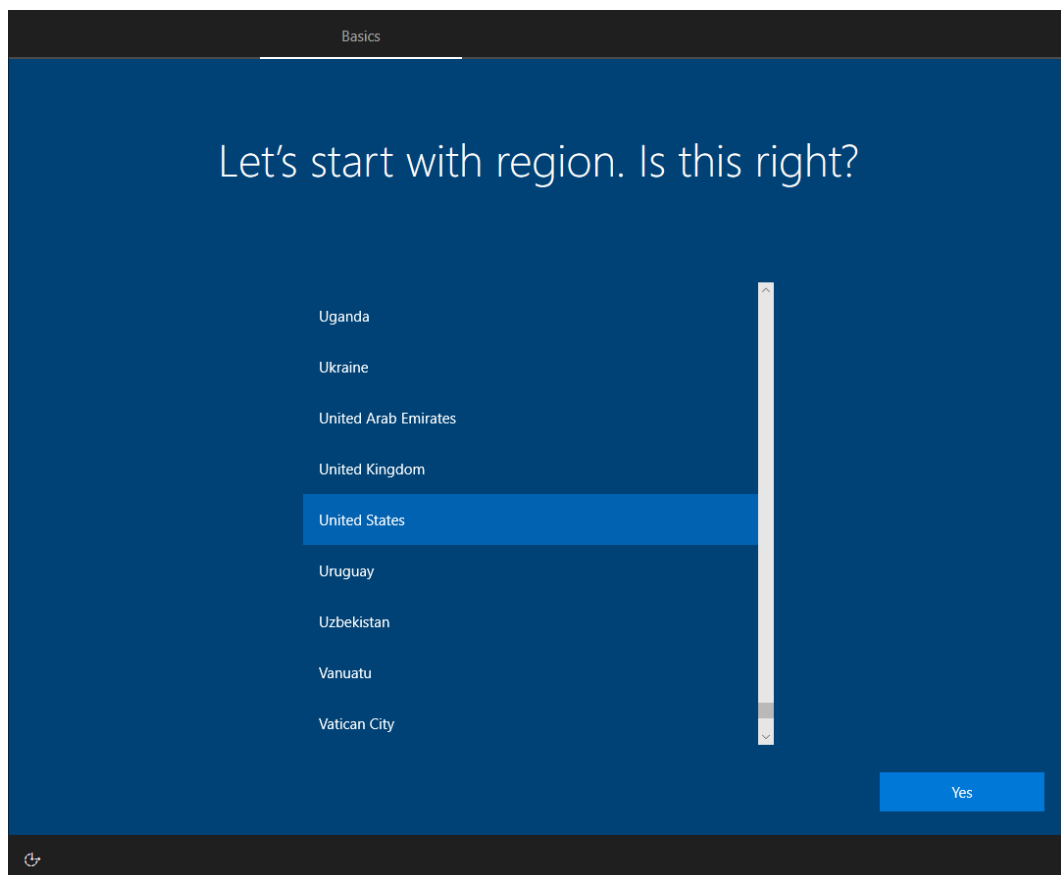
---

In this chapter, we describe how to initialize the system settings when you boot up the computer for the first time. When you turn on the computer, you will see the Windows Out of Box Experience (OOBE) wizard. OOBE consists of a series of screens that require customers to accept the license agreement, connect to the internet, log in with or sign up for a Microsoft Account, and share information with the OEM.

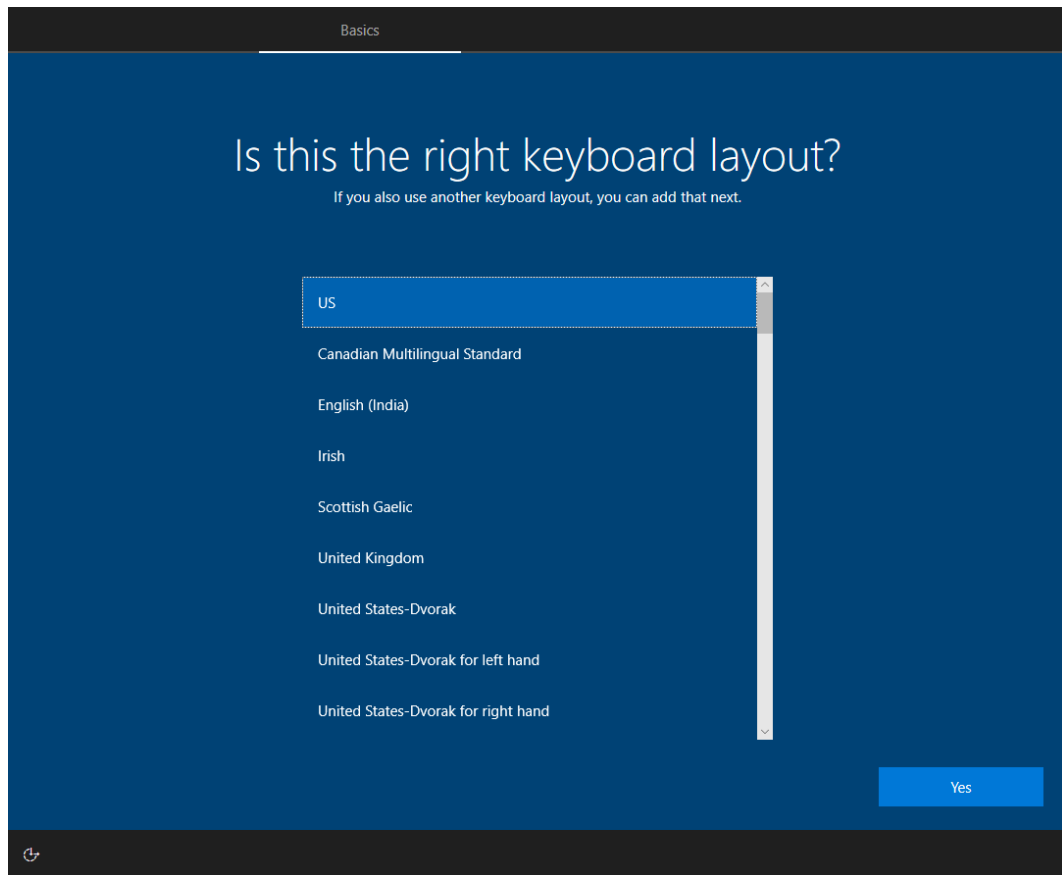
### Initializing User Settings

The following is a non-exhaustive list of OOBE screens that you will see in the order that they are listed here:

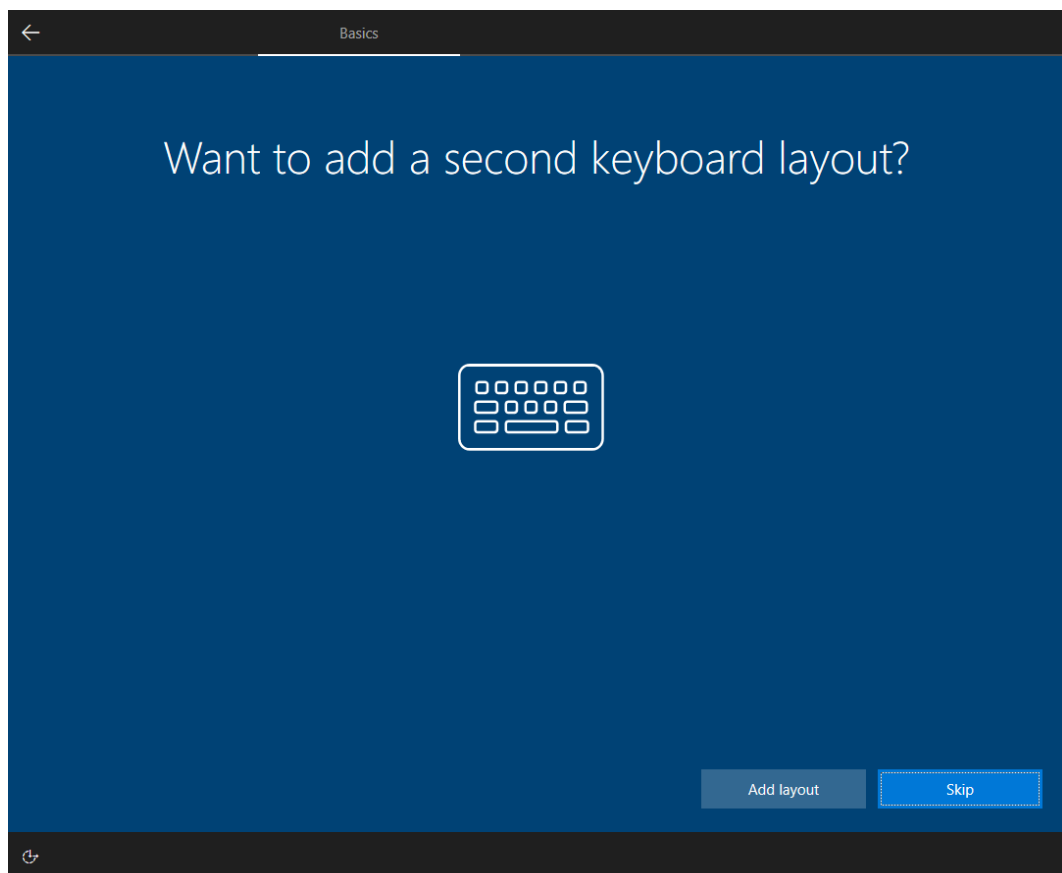
1. Select a region.



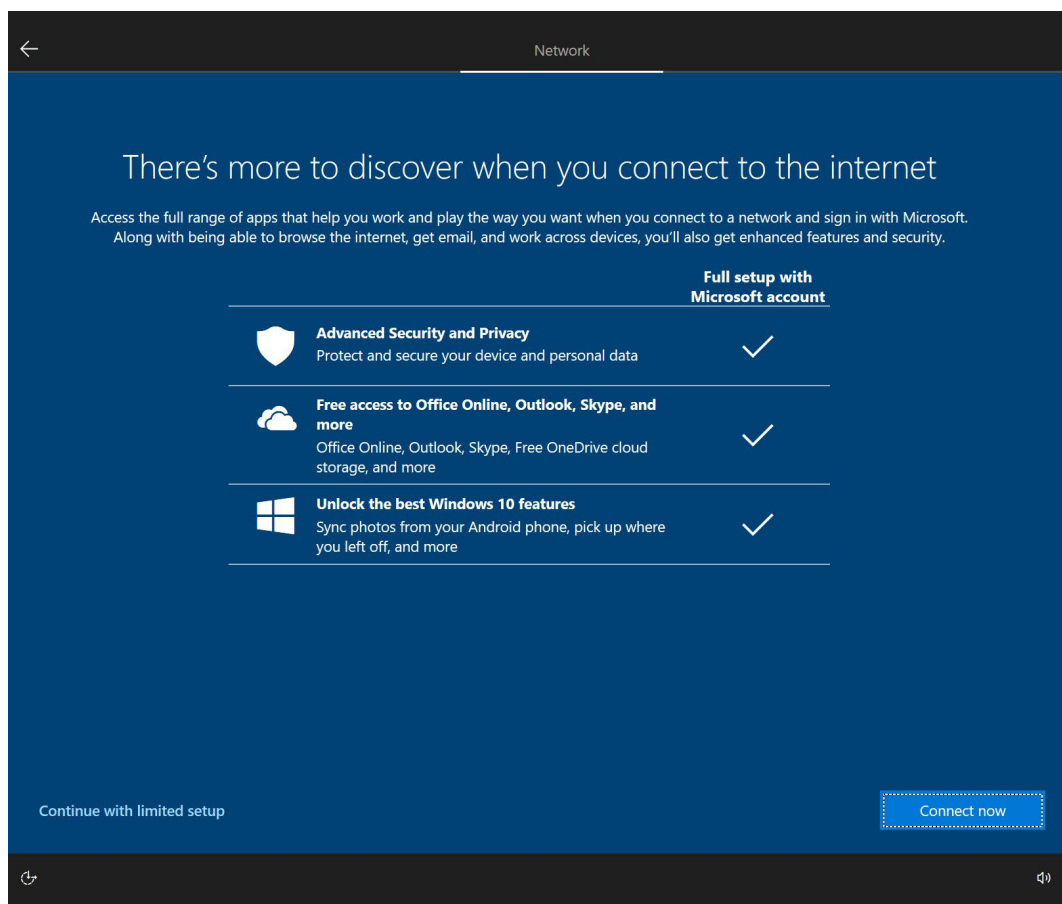
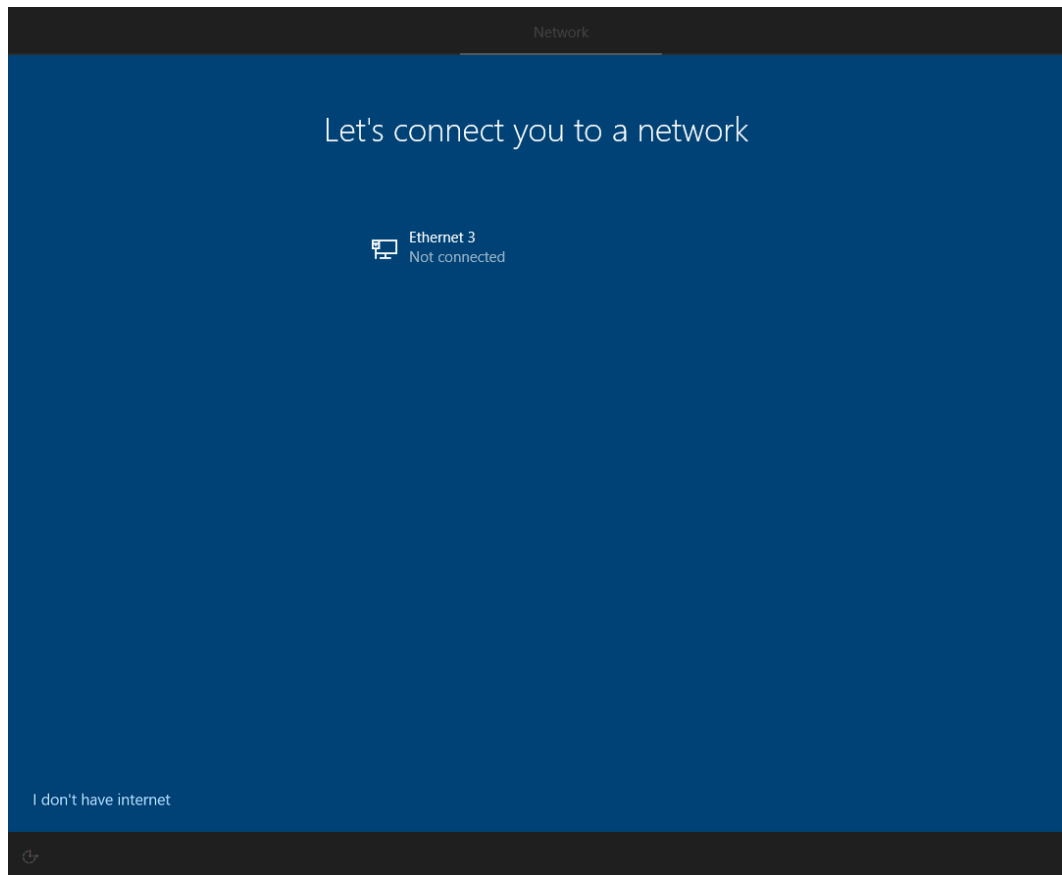
2. Select a keyboard.



3. Select a second keyboard.

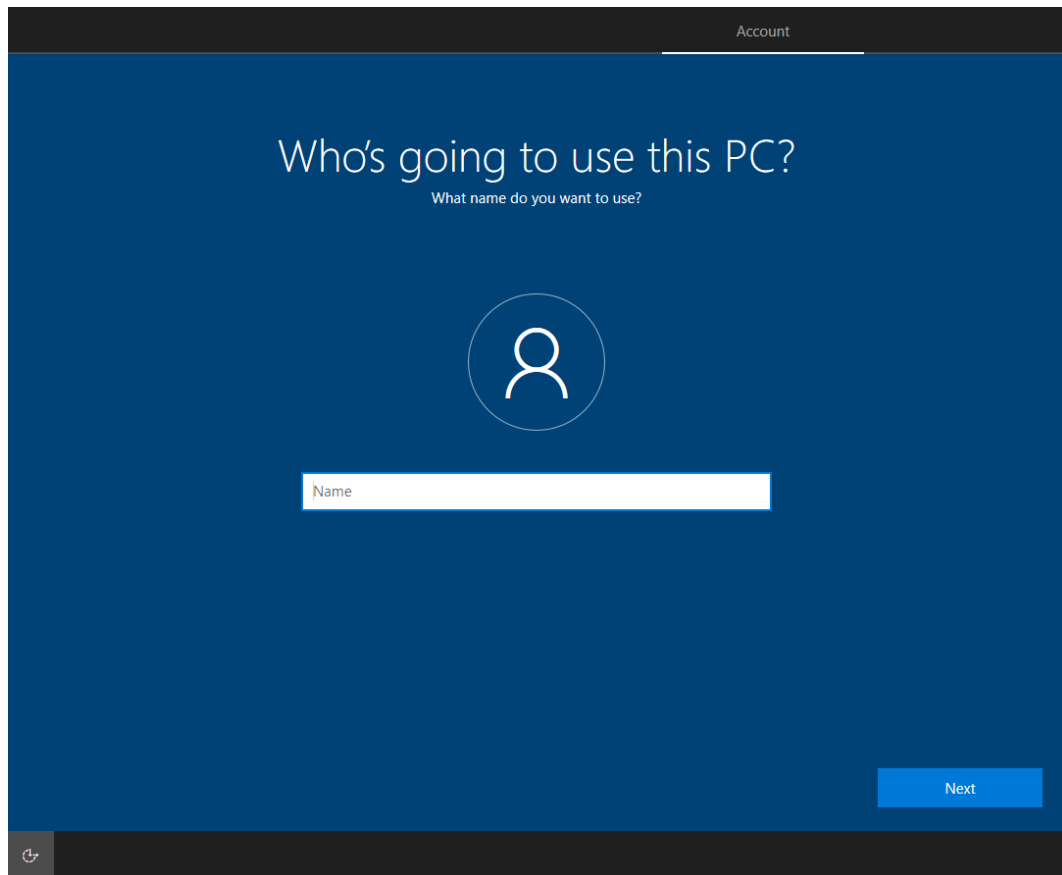


4. Connect to a network or continue with limited setup.

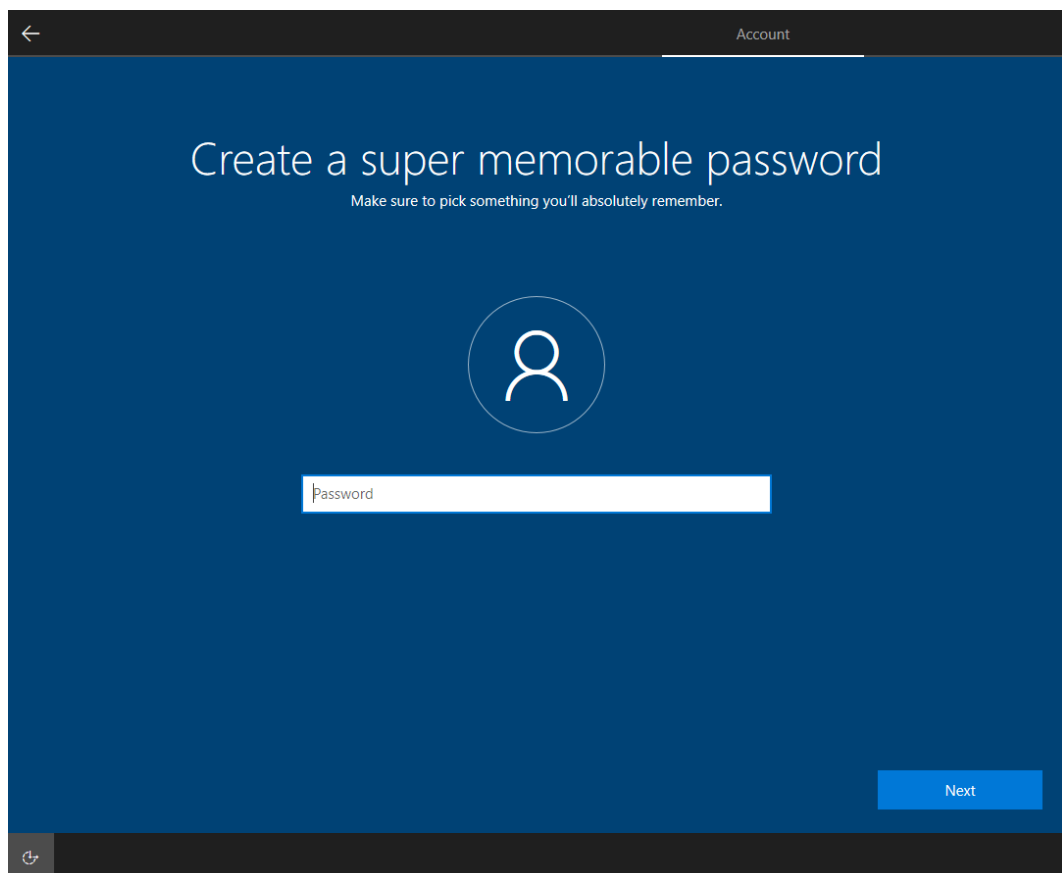




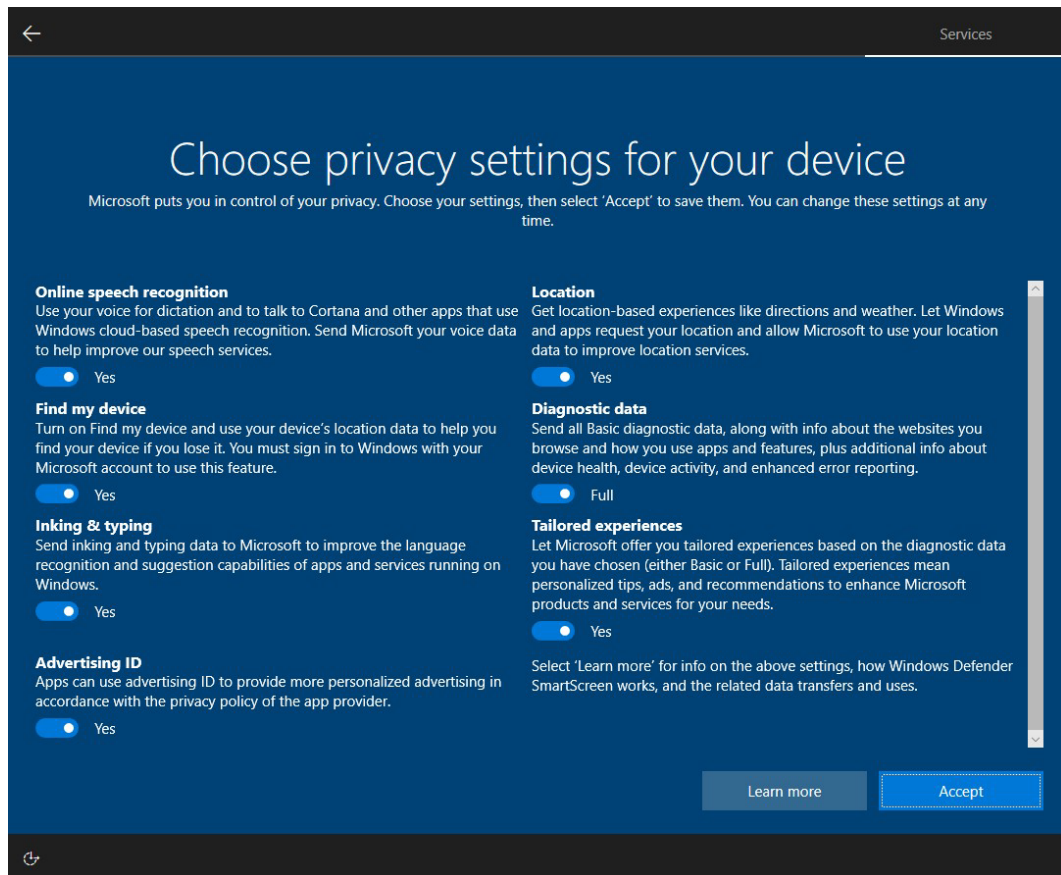
5. Sign in to or create a local account or a Microsoft account (MSA).



6. Set a password.



7. Choose your privacy settings.



## Initializing the System

After the OOBE settings, you will be redirected to the device desktop of the device. Wait until the process is complete.

```
SetupComplete
C:\Windows\system32>start/wait c:\windows\system32\SetOEMModel.exe
C:\Windows\system32>start/wait c:\windows\system32\SortNetName.exe
C:\Windows\system32>reg delete "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run" /v "SetupComplete" /f
The operation completed successfully.
```

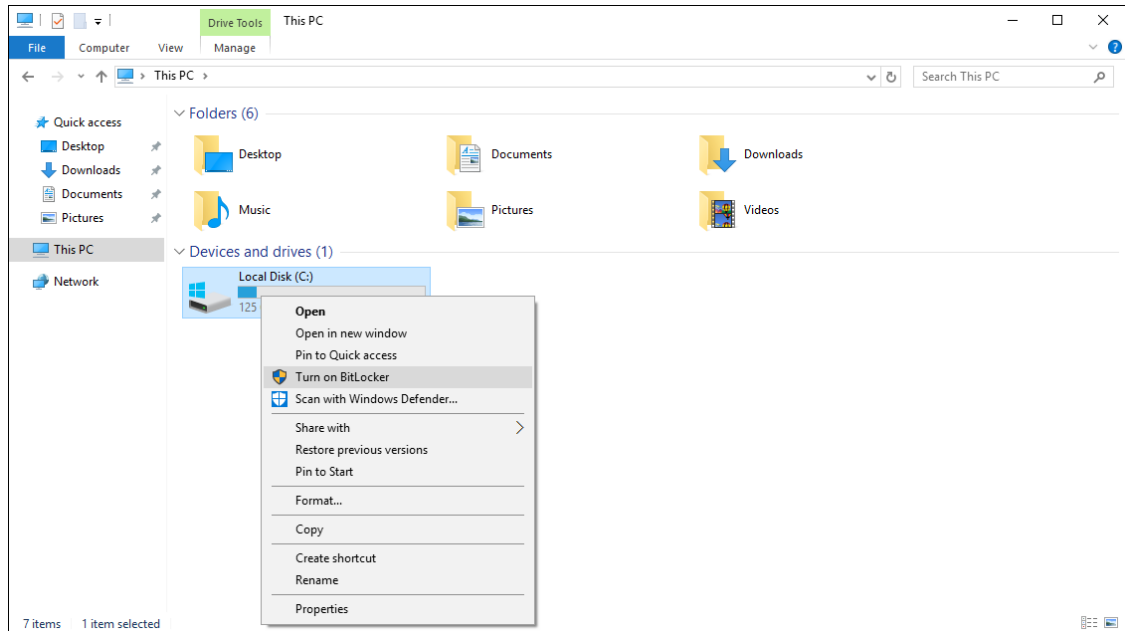
After the process is completed, the system initialization is done. The device will reboot, and the new settings will take effect after the system restarts.

# 3. BitLocker

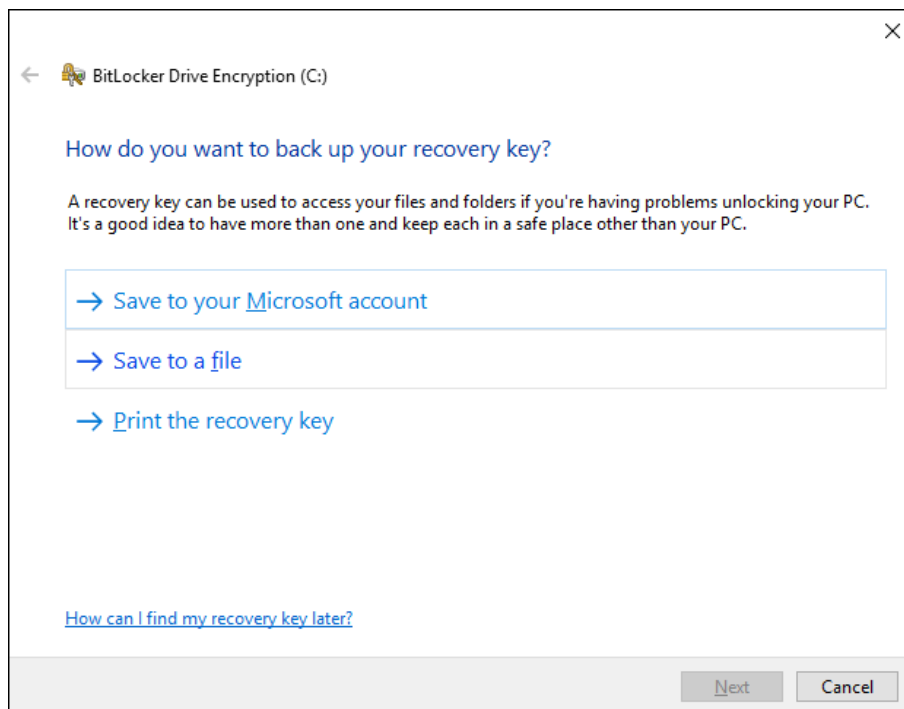
This chapter describes the BitLocker setup process.

## Enabling the BitLocker

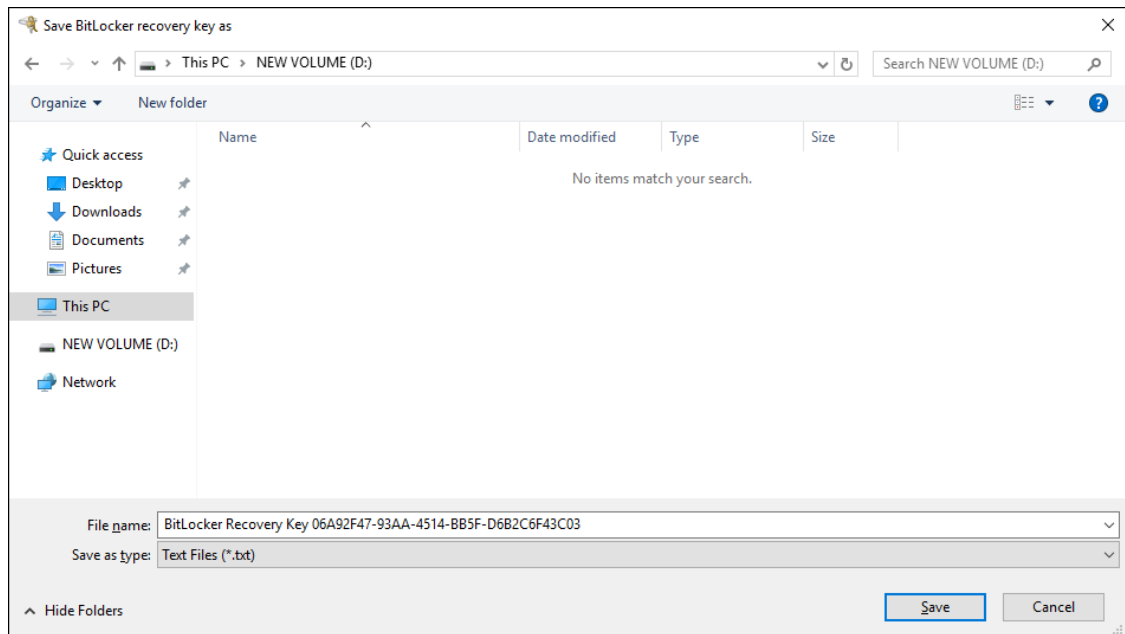
1. In the **Windows Devices and drives**, right-click on the drive and select **Turn on BitLocker**.



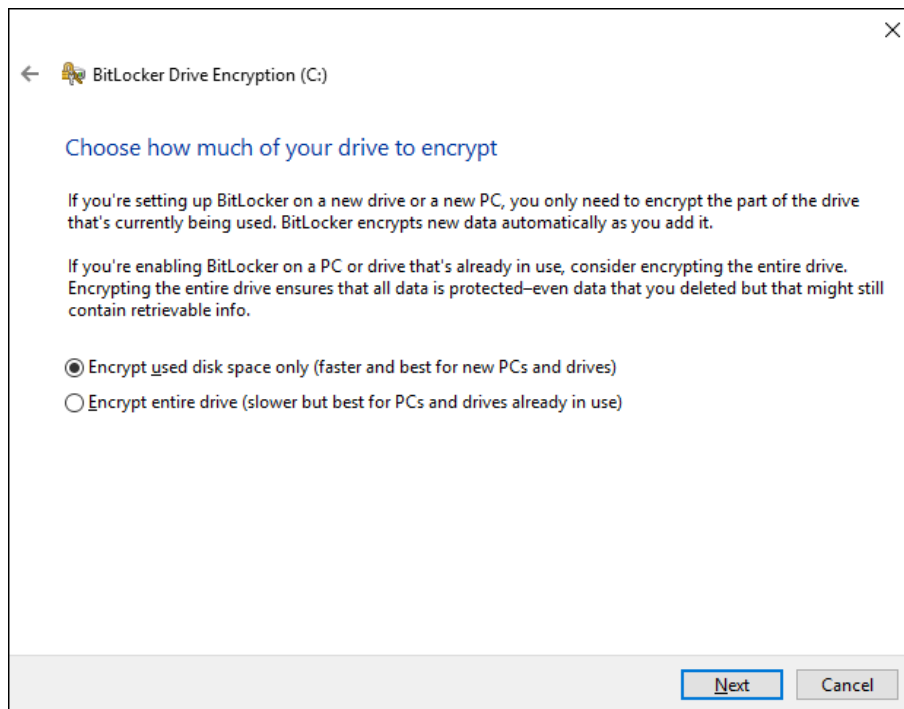
2. Select an option to back up the recovery key. For example, select **Save to a file**.

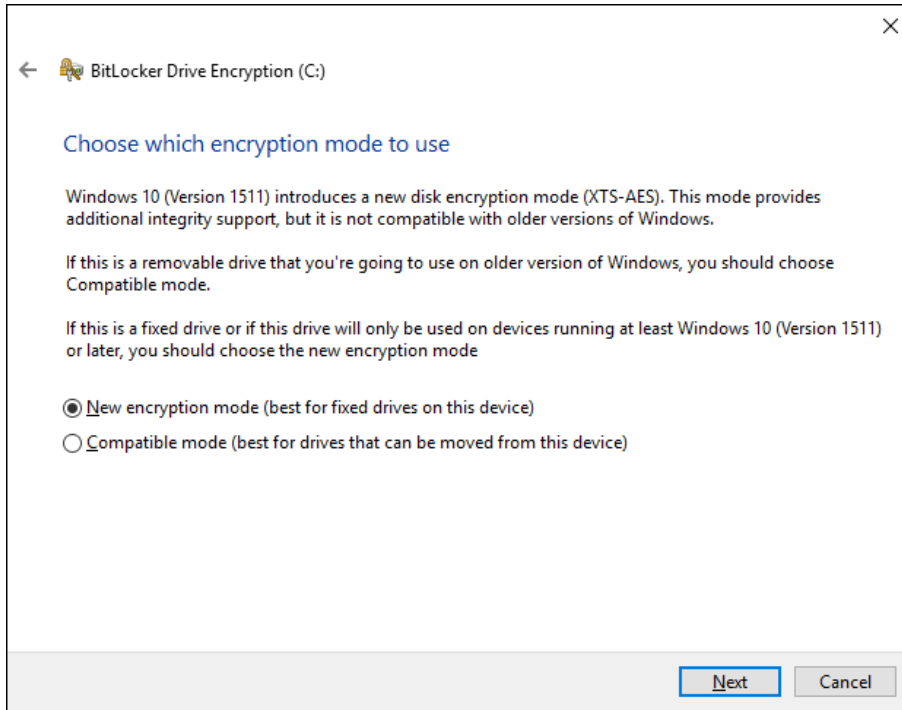


3. Select the path to store the file in.

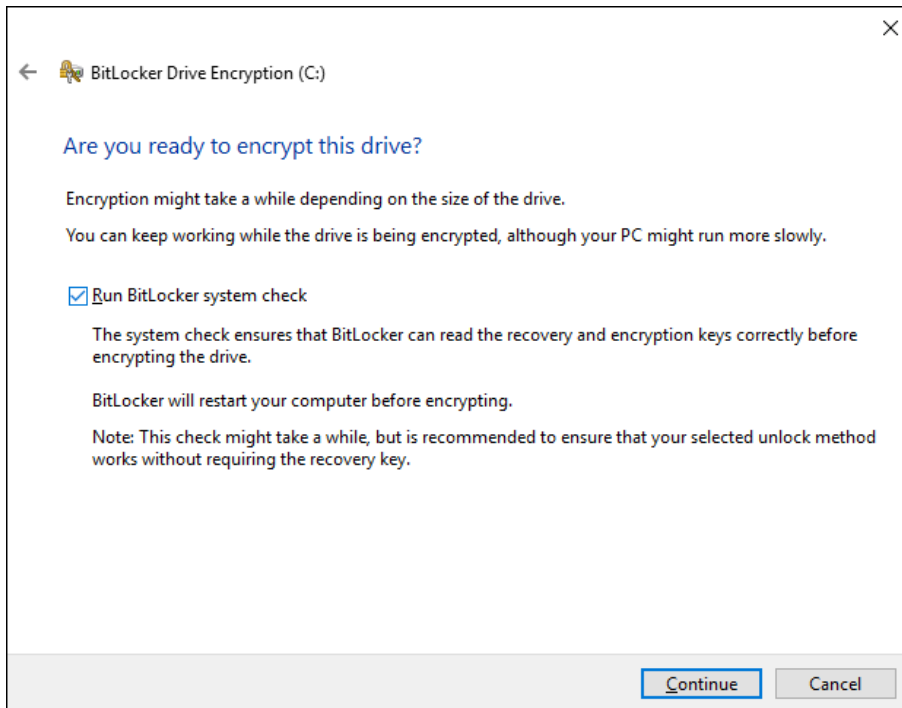


4. Follow the onscreen instructions to specify the drive encryption options.

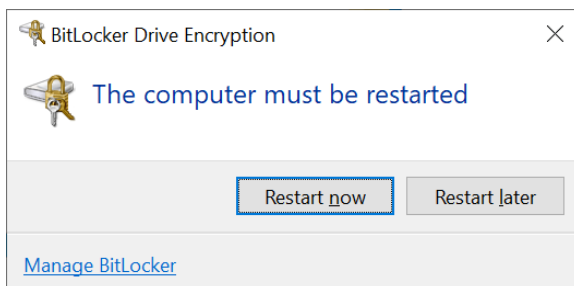




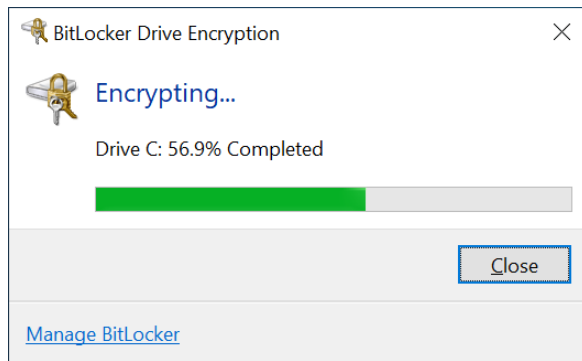
5. Click **Continue**.



6. Restart the computer.

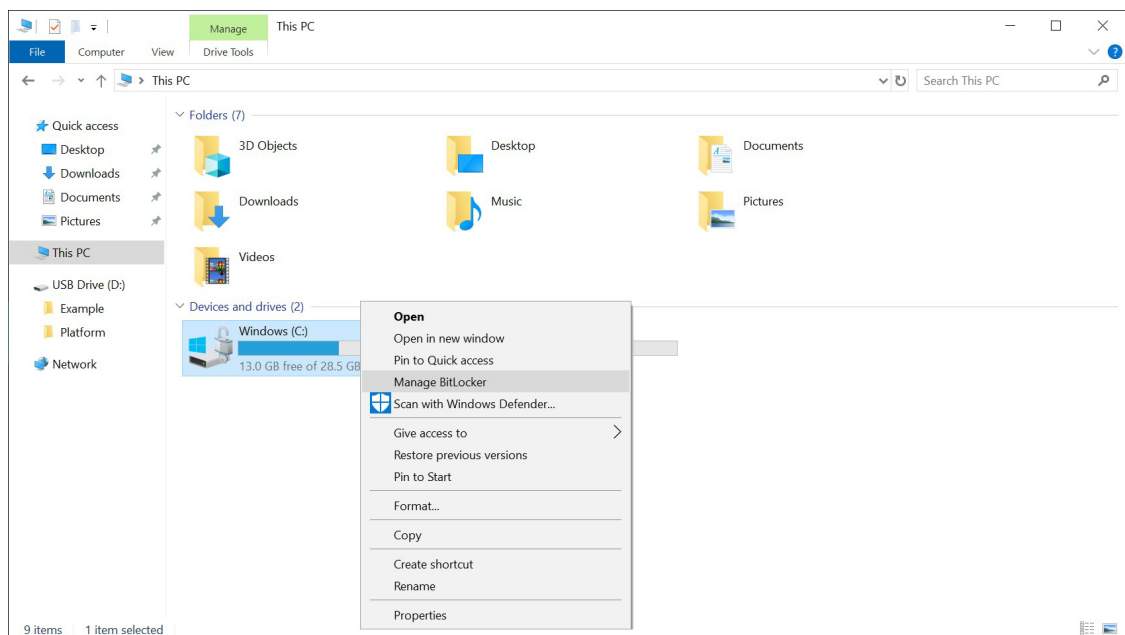


7. Wait for the encryption process to complete and then click **Close**.

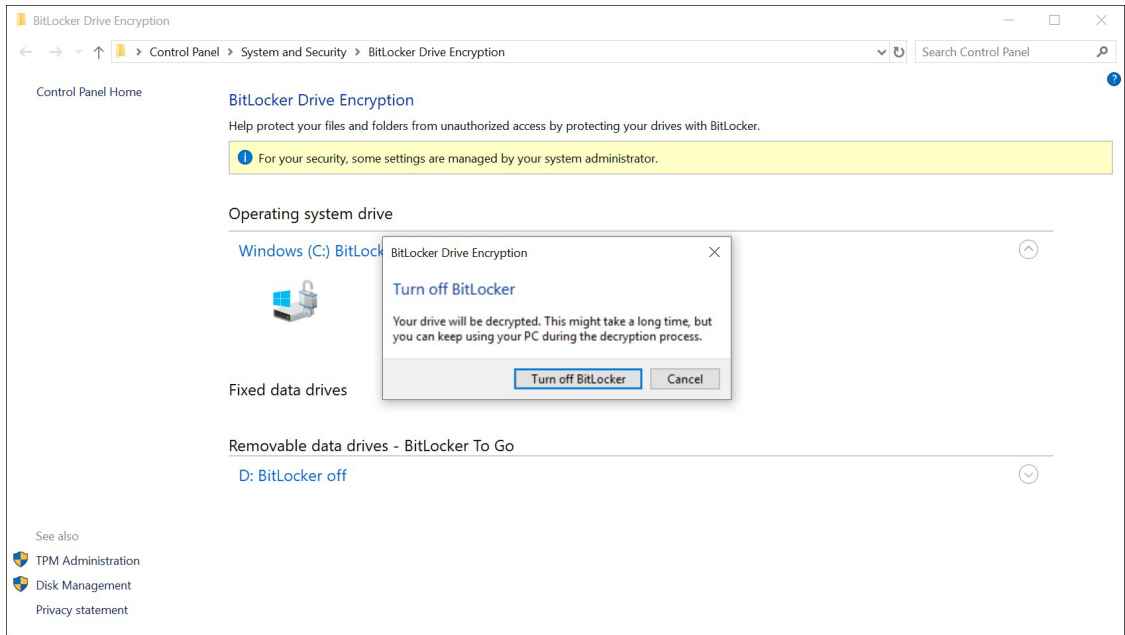


## Disabling the BitLocker

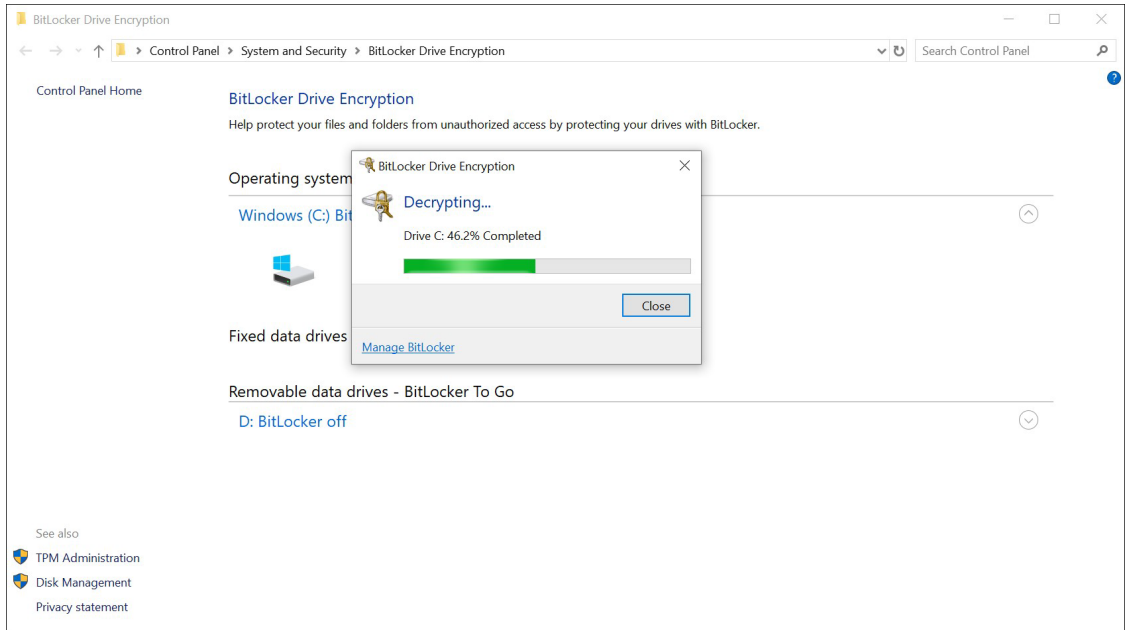
1. In the **Windows Devices and drives**, right-click on the drive and select **Manage BitLocker**.

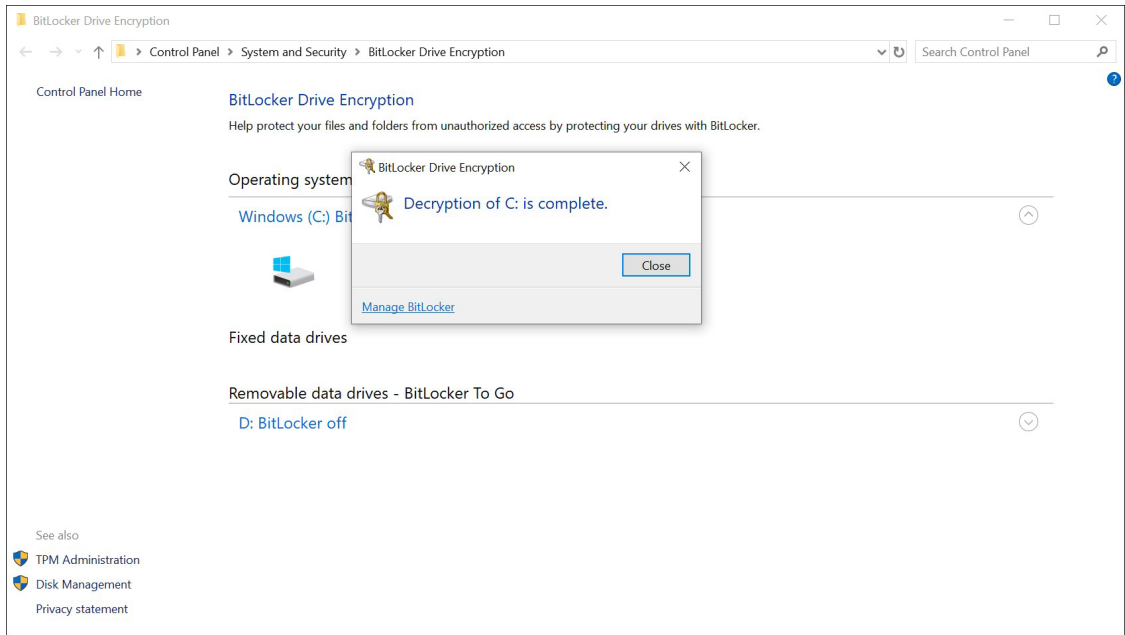


2. Click on **Turn off BitLocker**.

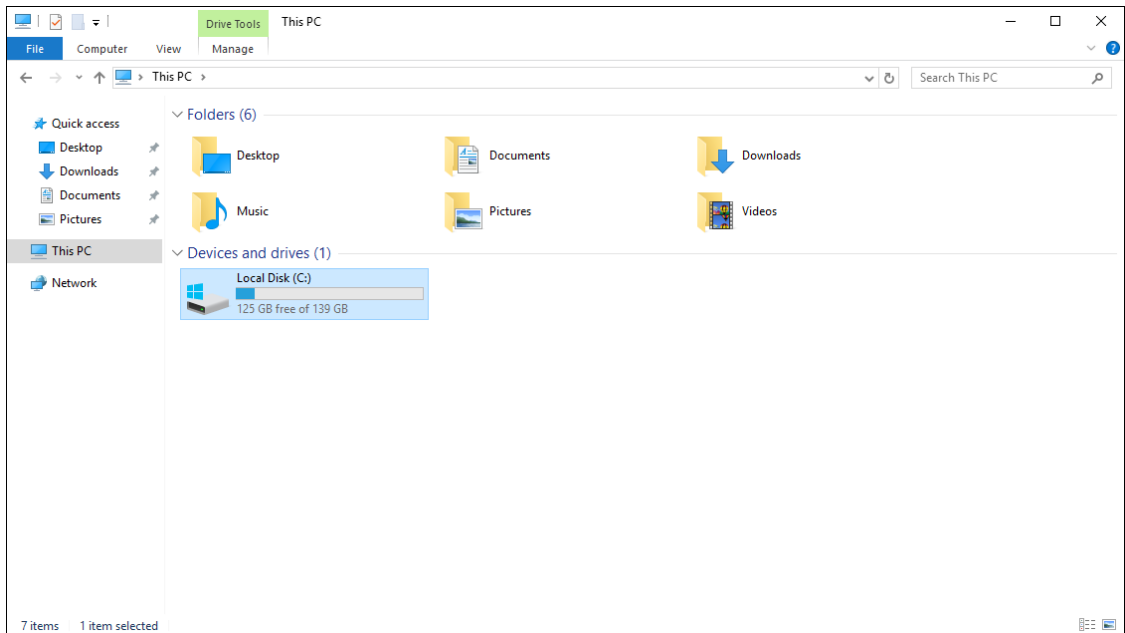


3. Wait for the decryption process to complete and click **Close** to exit the program.





4. Check the disk status after the decryption process is completed.



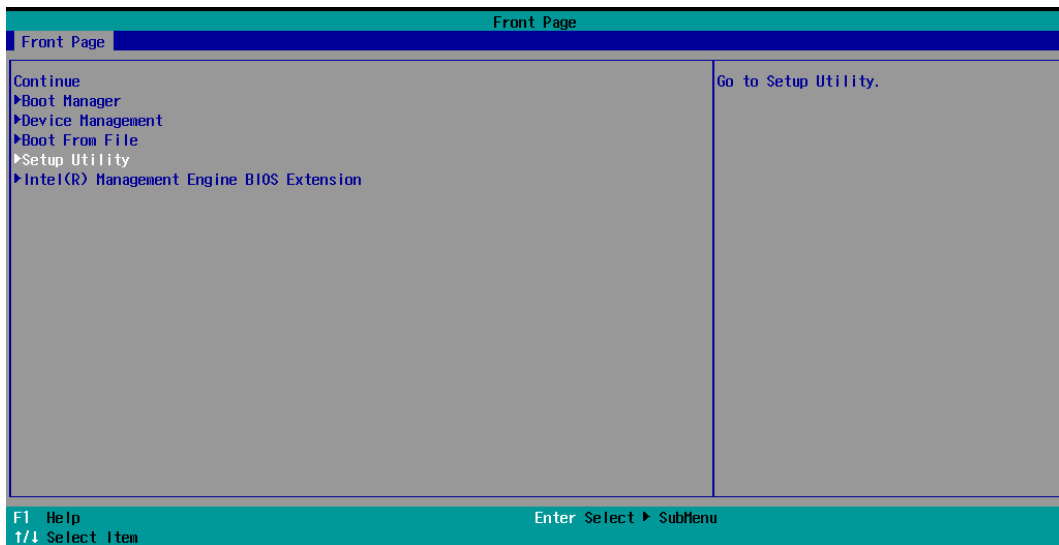


# 4. Configuring RAID

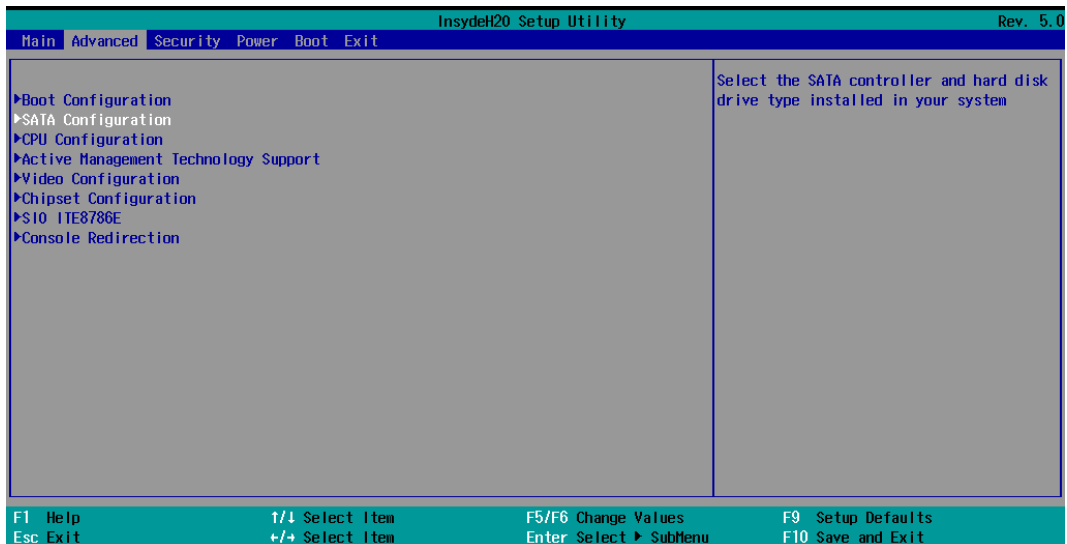
This chapter describes the setup process for RAID.

## Changing the RAID Mode

1. Power on the computer and press **F2** to enter the BIOS menu.
2. Select the **Setup Utility** option.

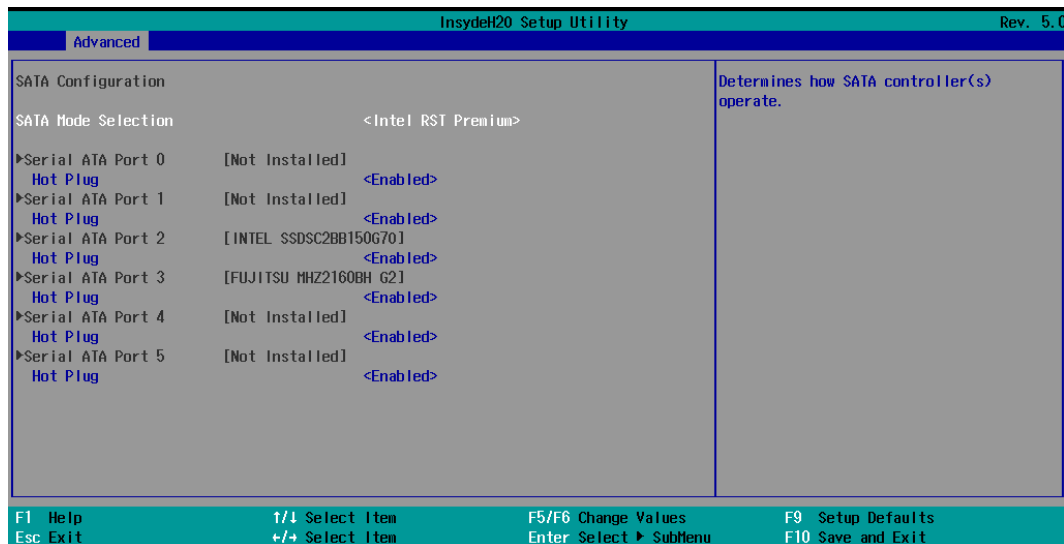
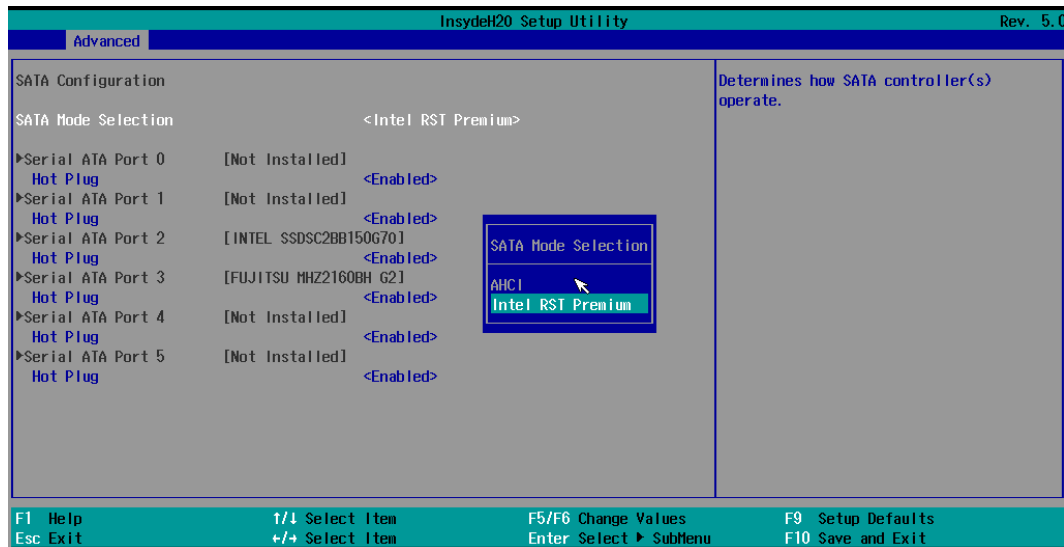


3. Select the **SATA Configuration** option.



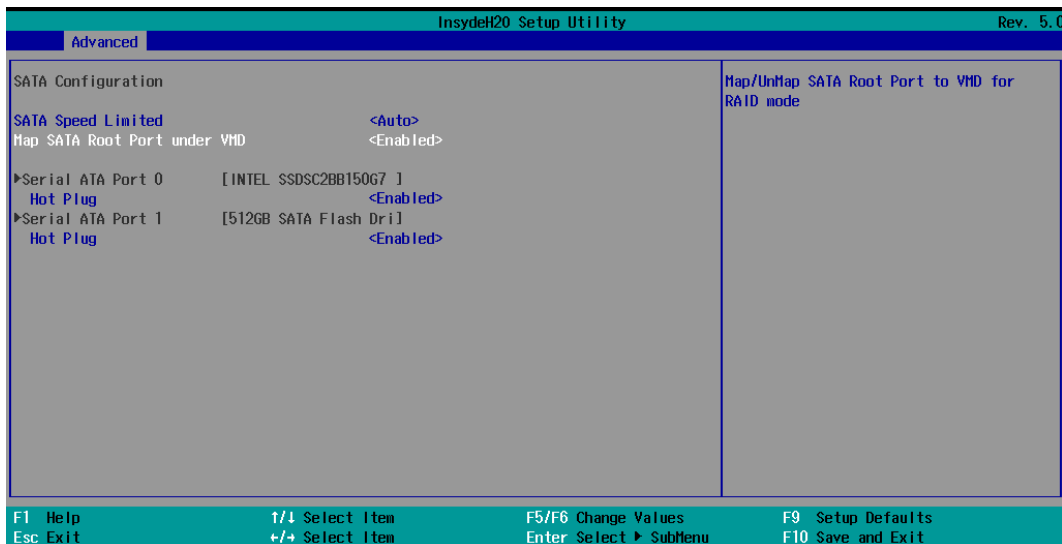
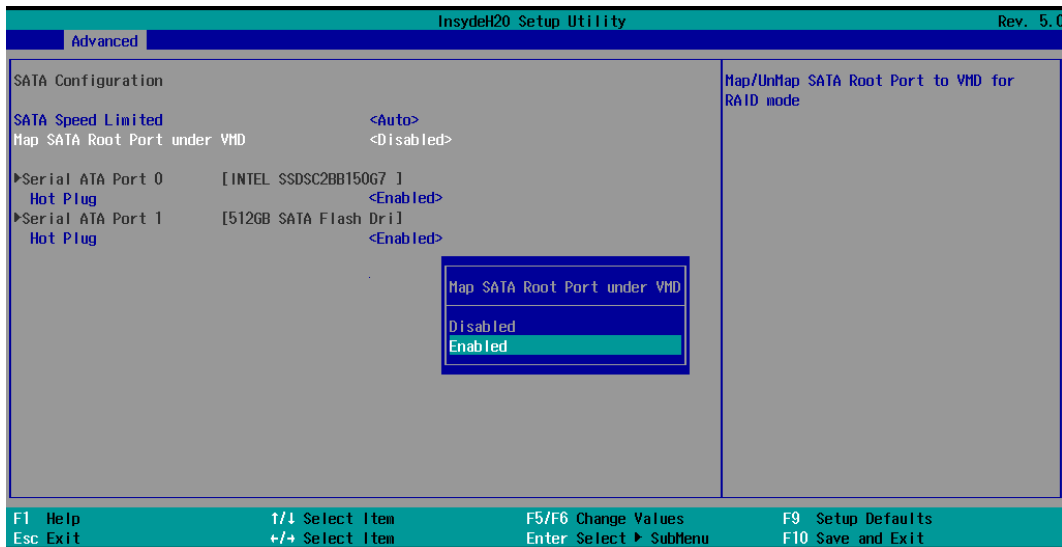
- If your device processor is older than **11th Gen Intel® Core™ Processor (Intel® Tiger-Lake)**, select the **SATA Mode Selection** followed by the **Intel RST Premium** option.

Enable the **Hot Plug** function on all ports.

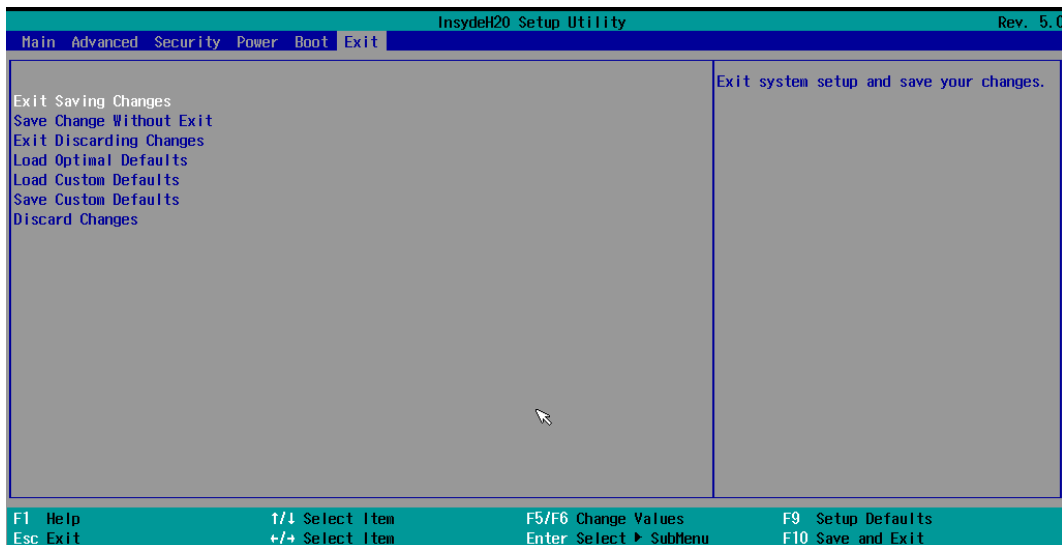


- If your device CPU is **11th Gen Intel® Core™ Processors (Intel® Tiger-Lake)** or newer processor generation, select the **Map SATA Root Port under VMD** and **Enable** this option.

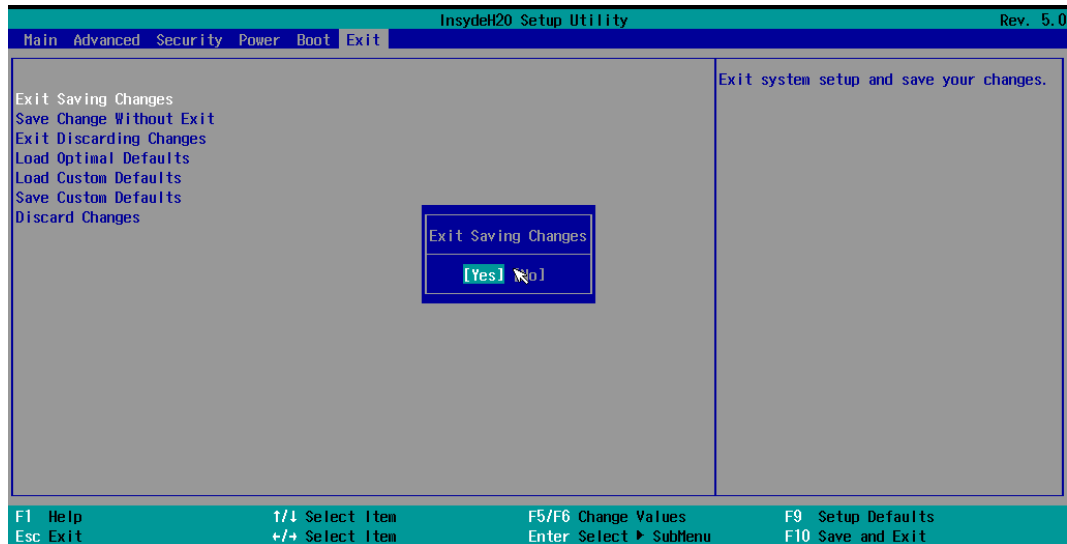
Enable the **Hot Plug** function on all ports.



- Press **F10** to save the settings and then press **ESC** to return to the main page.

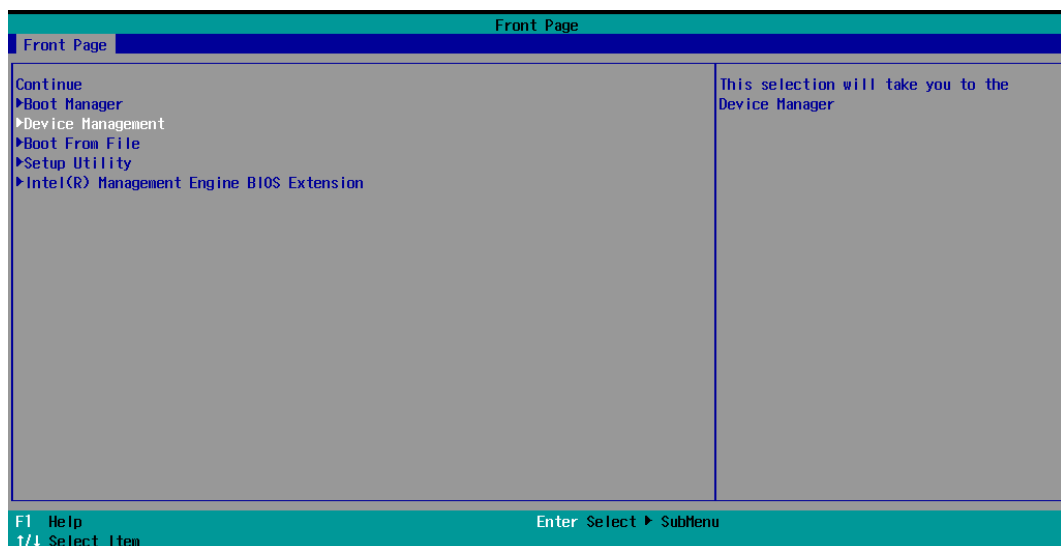


7. Select **Exit Saving Changes**, and then select **Yes** to save the settings.

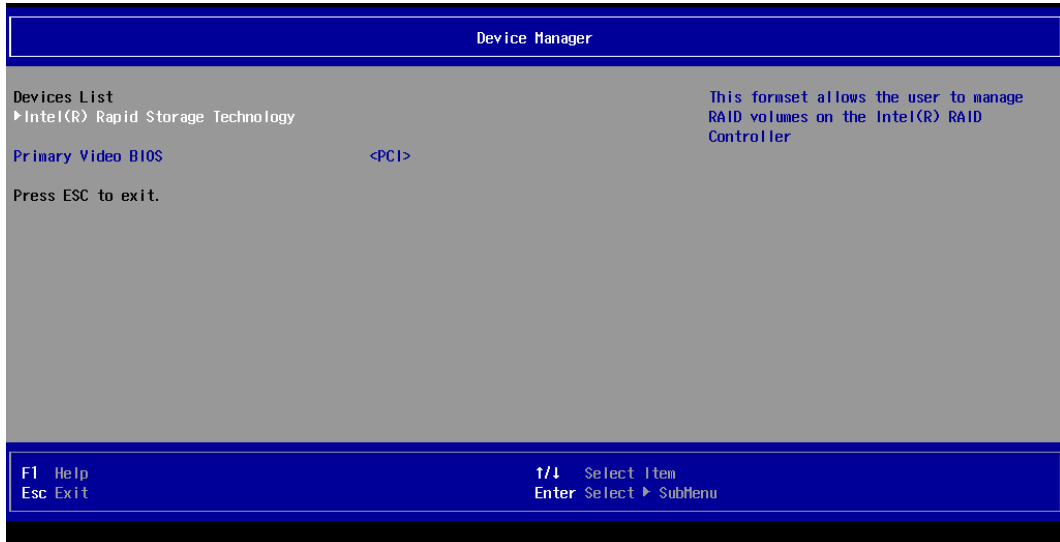


## Creating a RAID Disk in BIOS

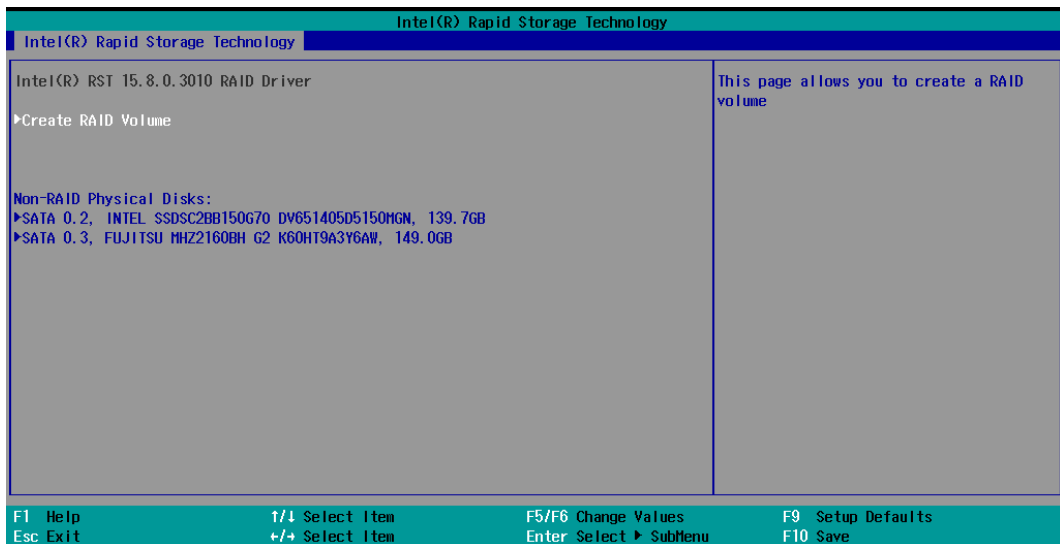
1. Power on the computer and press **F2** to enter the BIOS menu.
2. Select the **Device Management** option.



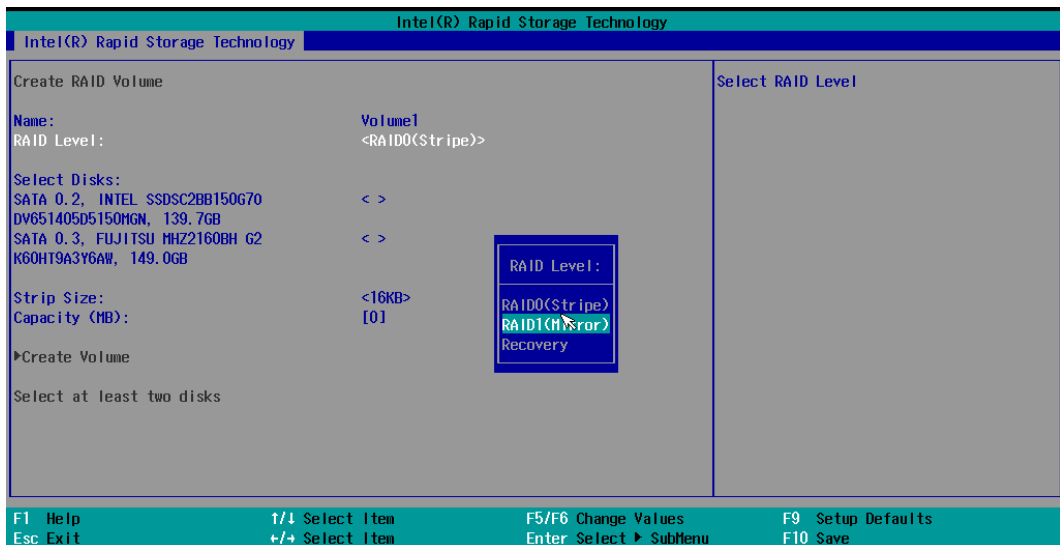
3. Select **Intel® Rapid Storage Technology**.

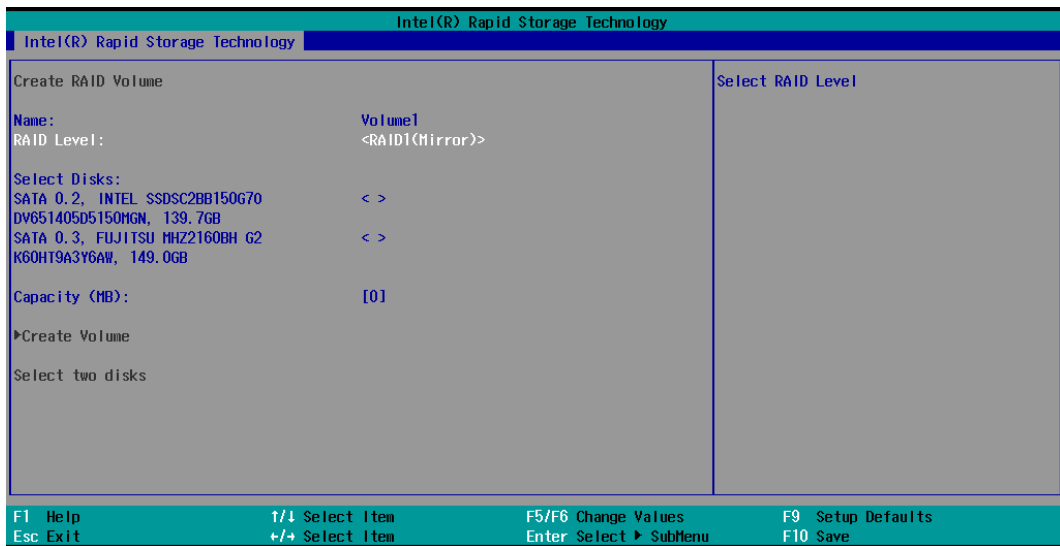


4. Select **Create RAID Volume**.

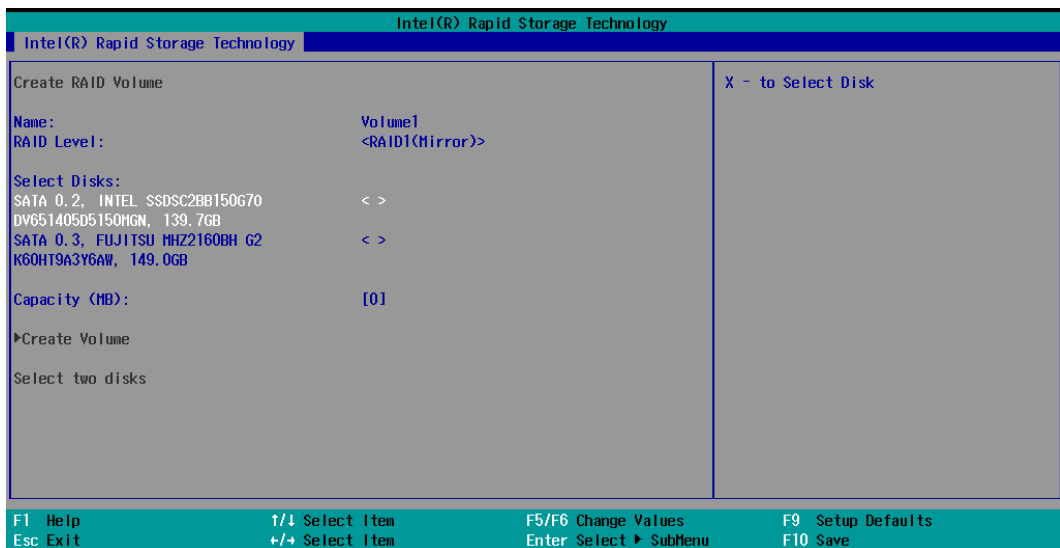


5. Select the **RAID Level** option and then press **Enter** to select the raid level; for example, **RAID1(Mirror)**.

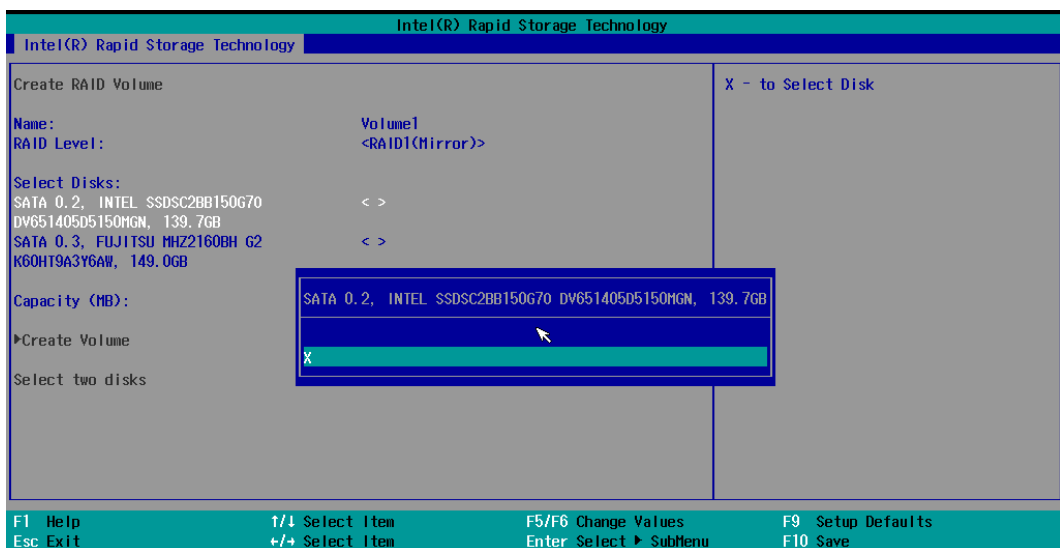




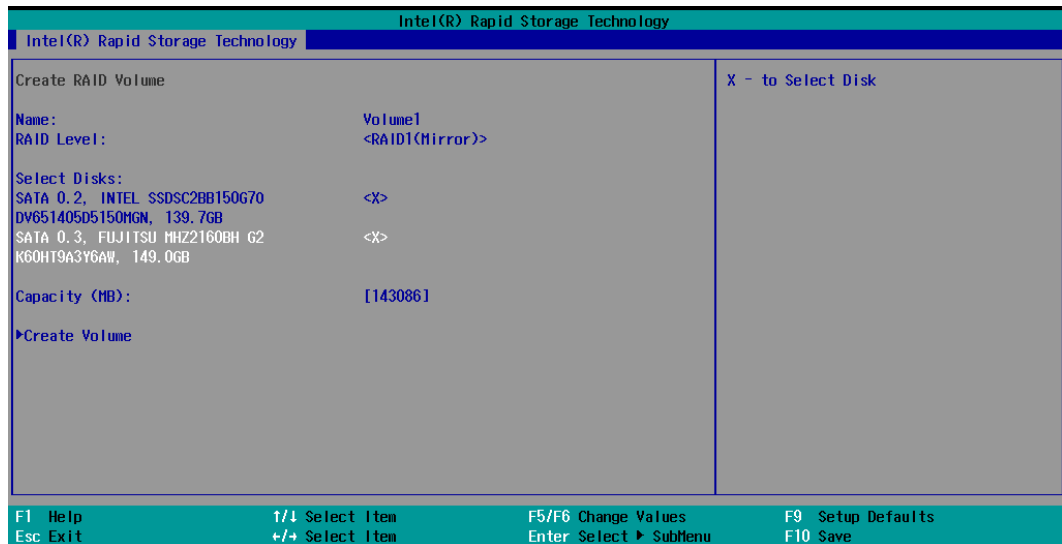
6. Select the target disk.



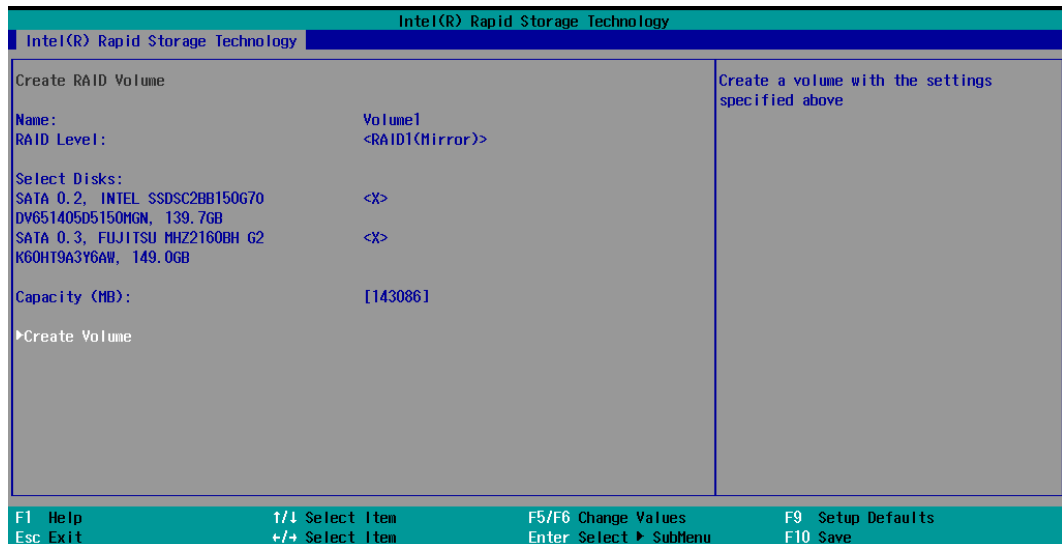
7. Enter **X** and then press **Enter**.



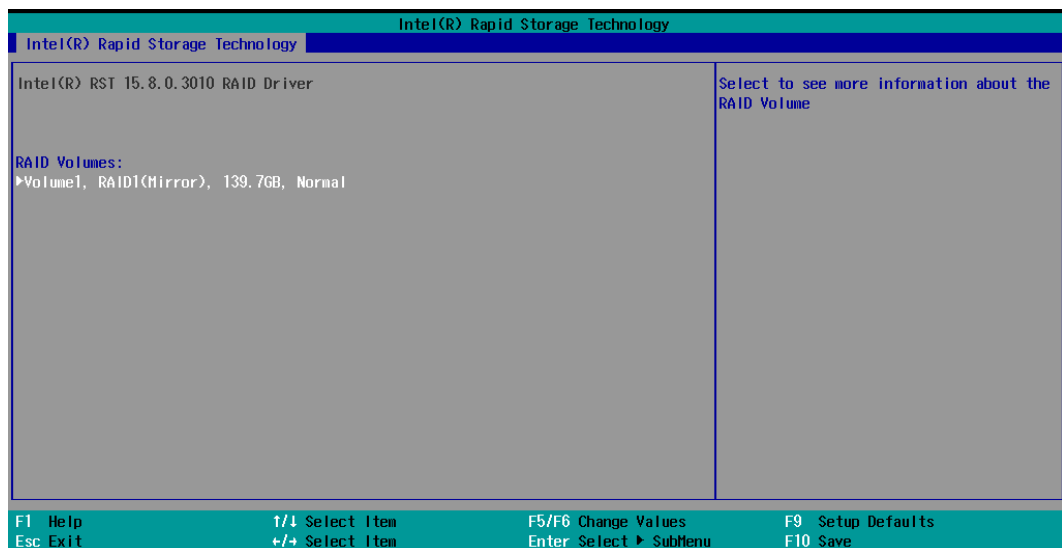
- The disk is now marked with an **X** next to it to indicate the selection.



- Select the **Create Volume** option.



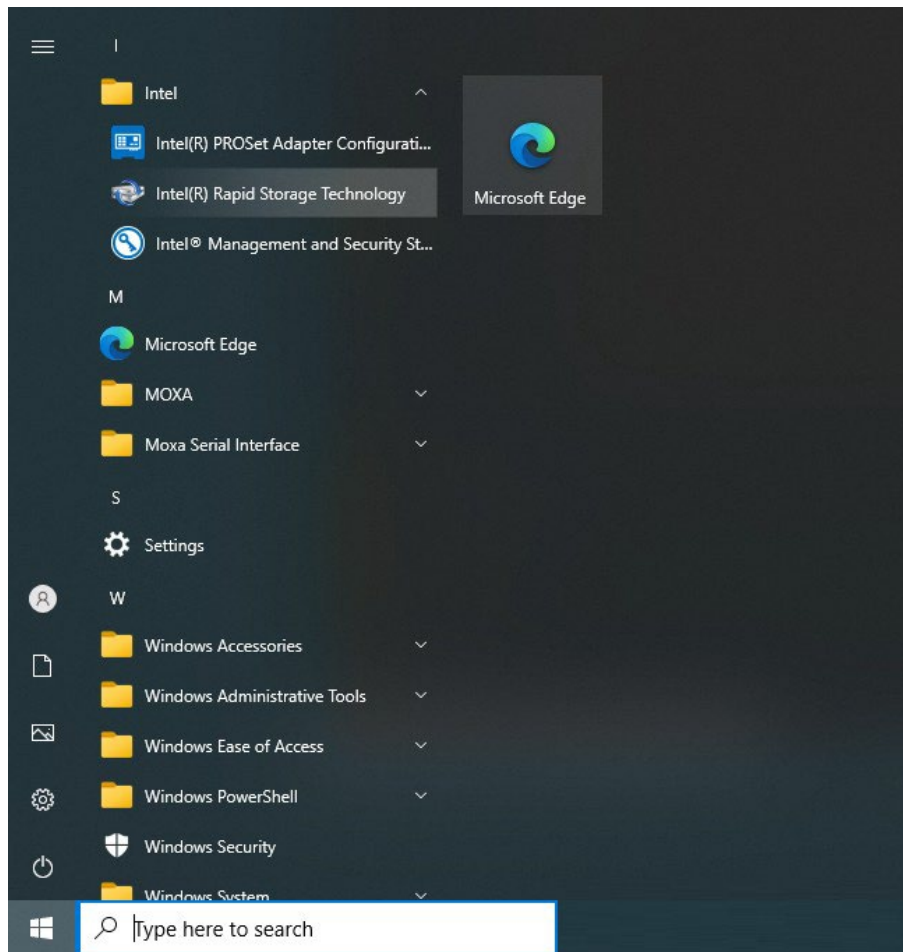
- A RAID volume is created based on the settings specified.



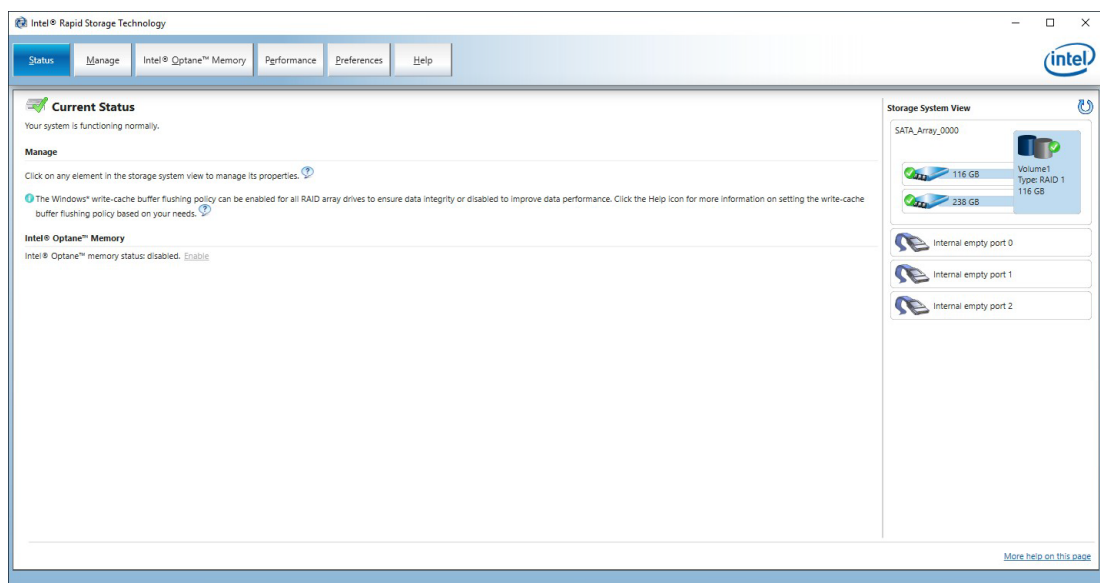
- Press **F10** to save the settings.

# Replacing a Disk

1. Run **Intel® Rapid Storage Technology** from the Windows Start menu.

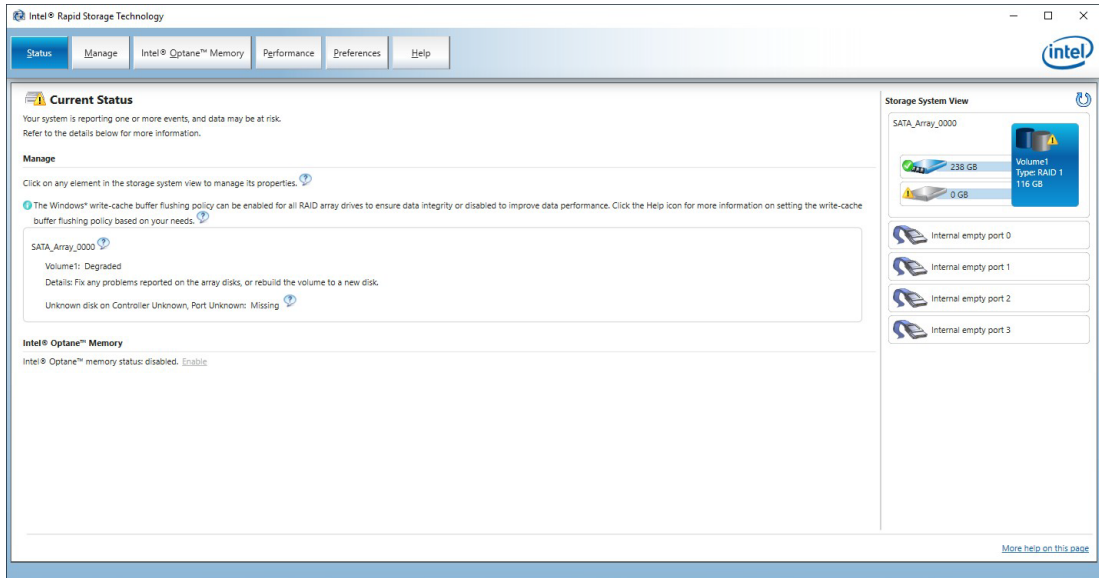


2. Physically unplug the target SSD.

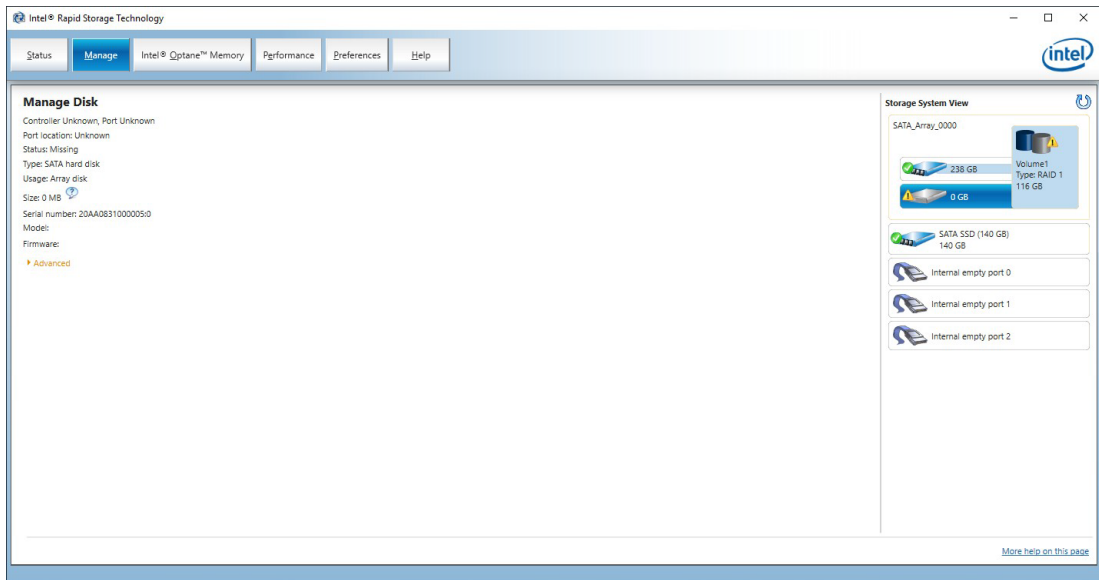


The status of the SSD that is unplugged will change from green (check mark) to yellow (exclamation mark) as shown below:

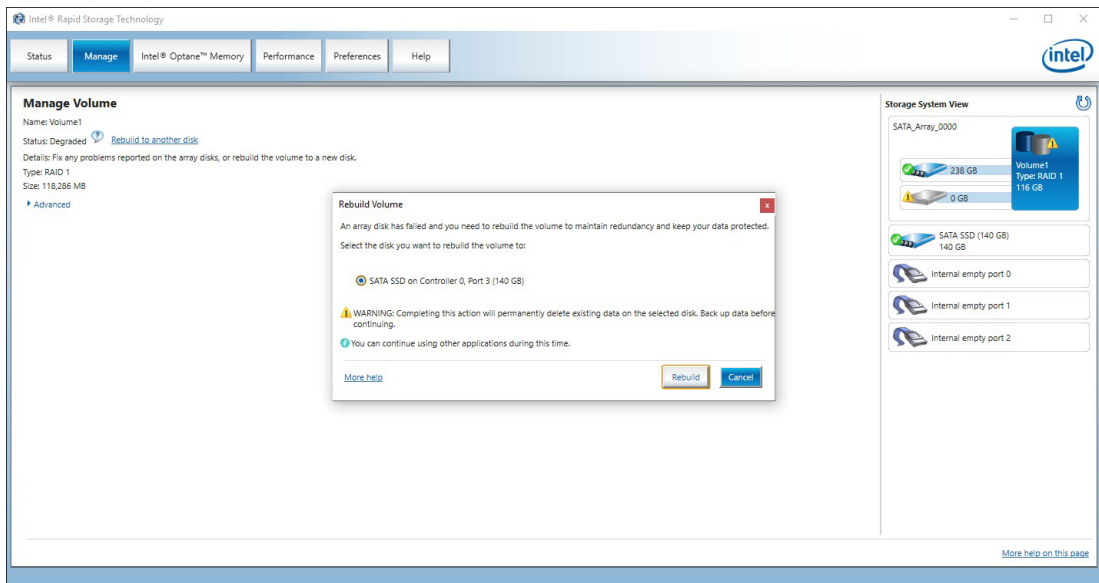
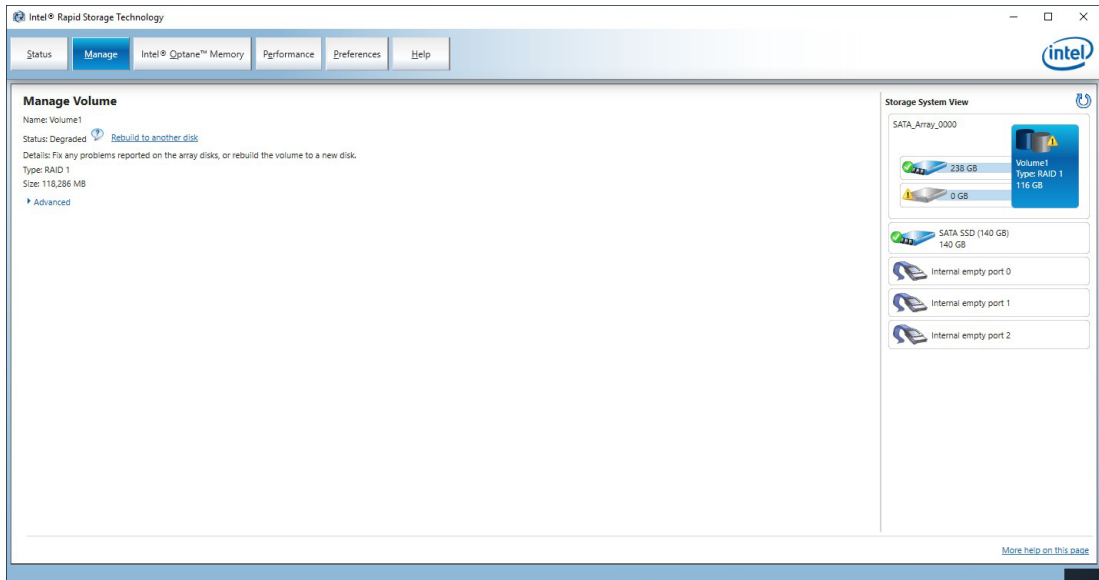




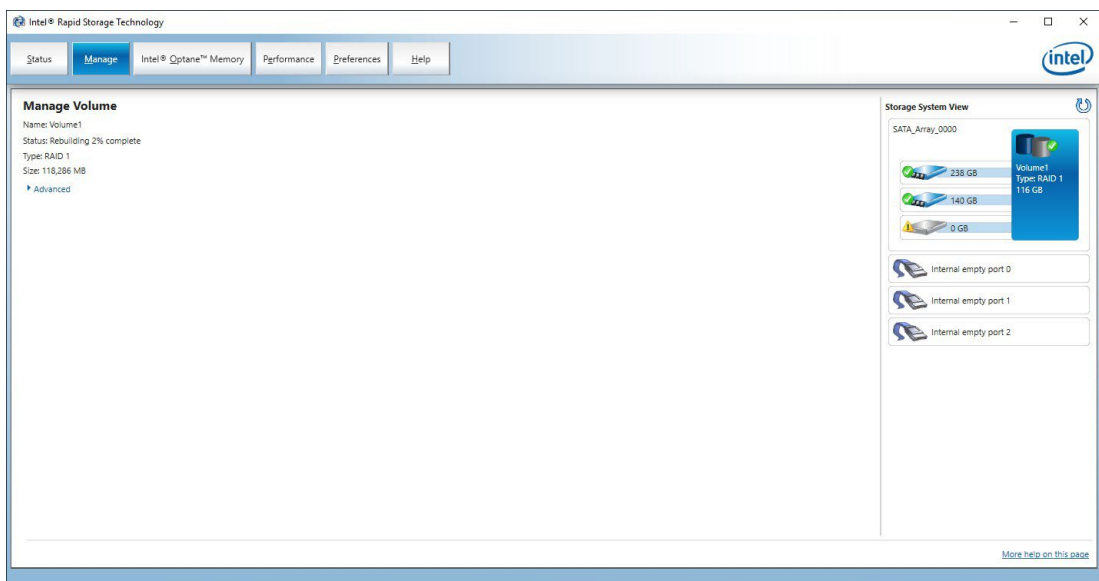
3. Install the new SSD.
4. Click on the **Manage** tab.



5. Select the new SSD and click **Rebuild**.

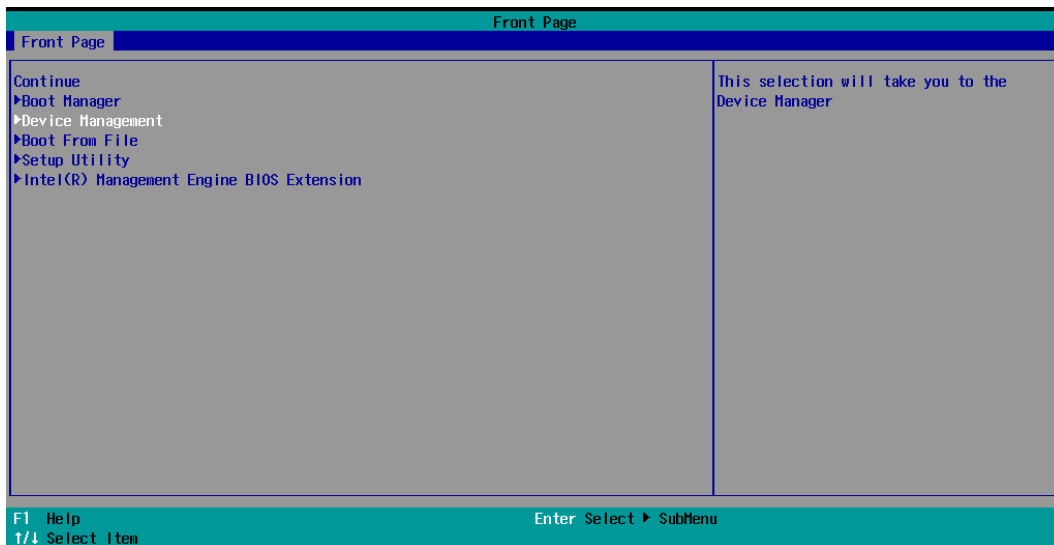


6. Wait for the rebuild process to complete.

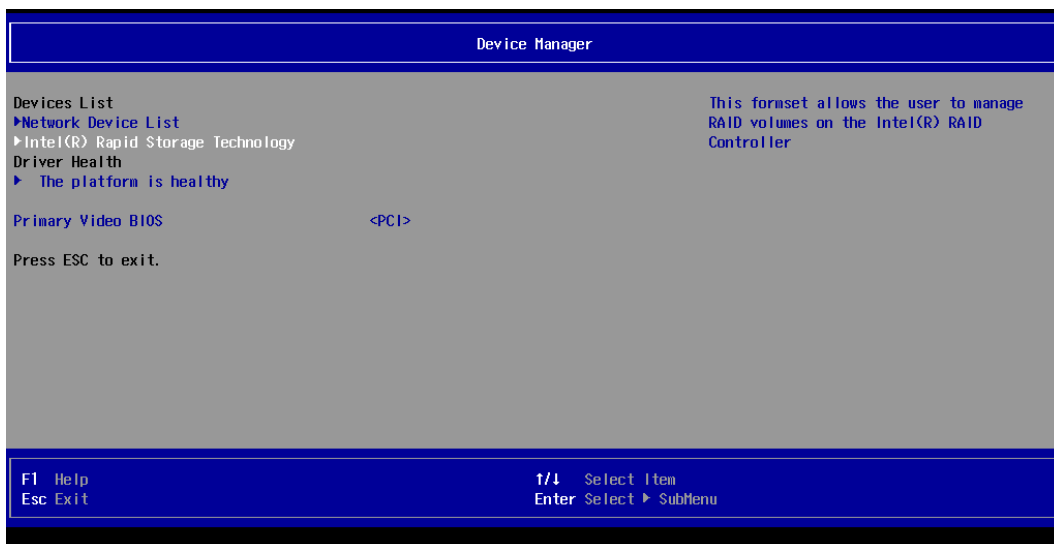


# Removing a RAID Volume From the BIOS

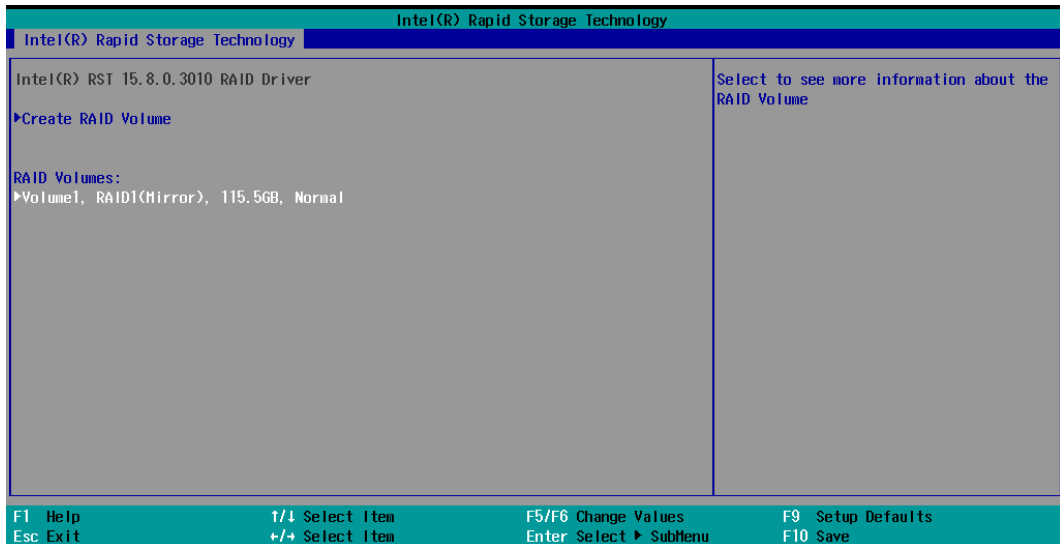
1. Power on the computer and press **F2** to enter the BIOS menu.
2. Select **Device Management**.



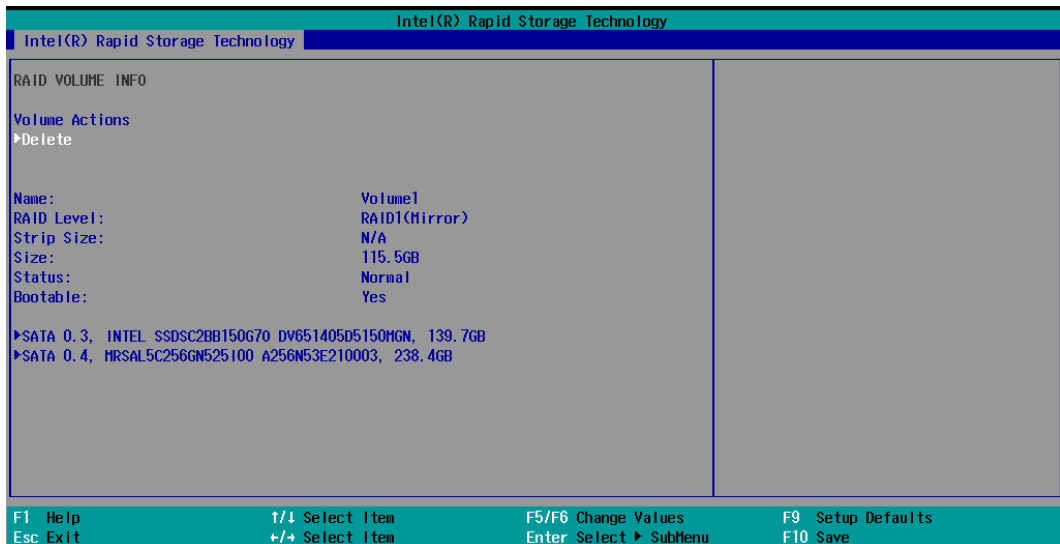
3. Select the **Intel® Rapid Storage Technology** option.



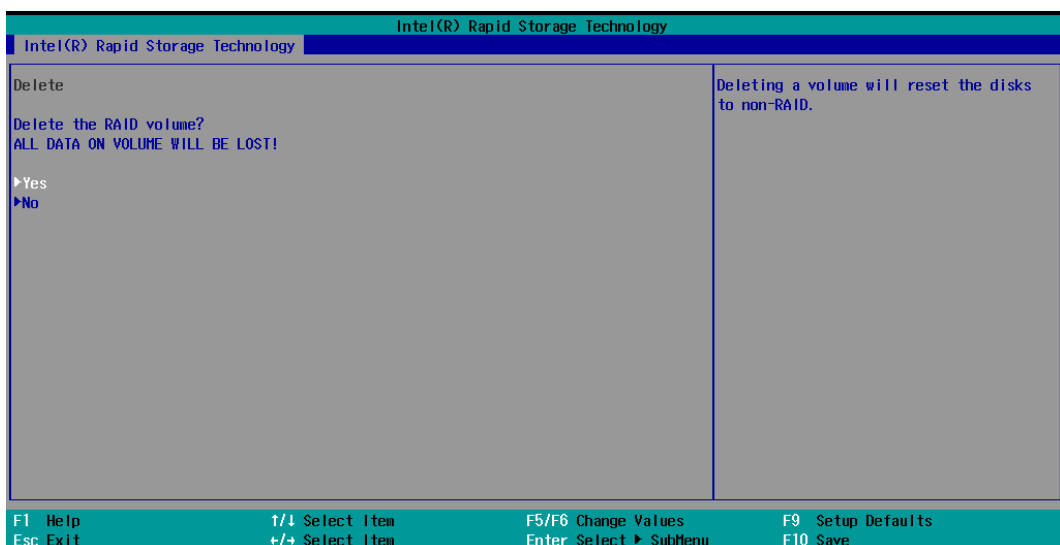
- Select the RAID volume that you want to remove.



- Select **Delete** and then press **Enter**.



- Select **Yes** to confirm and then press **Enter**.



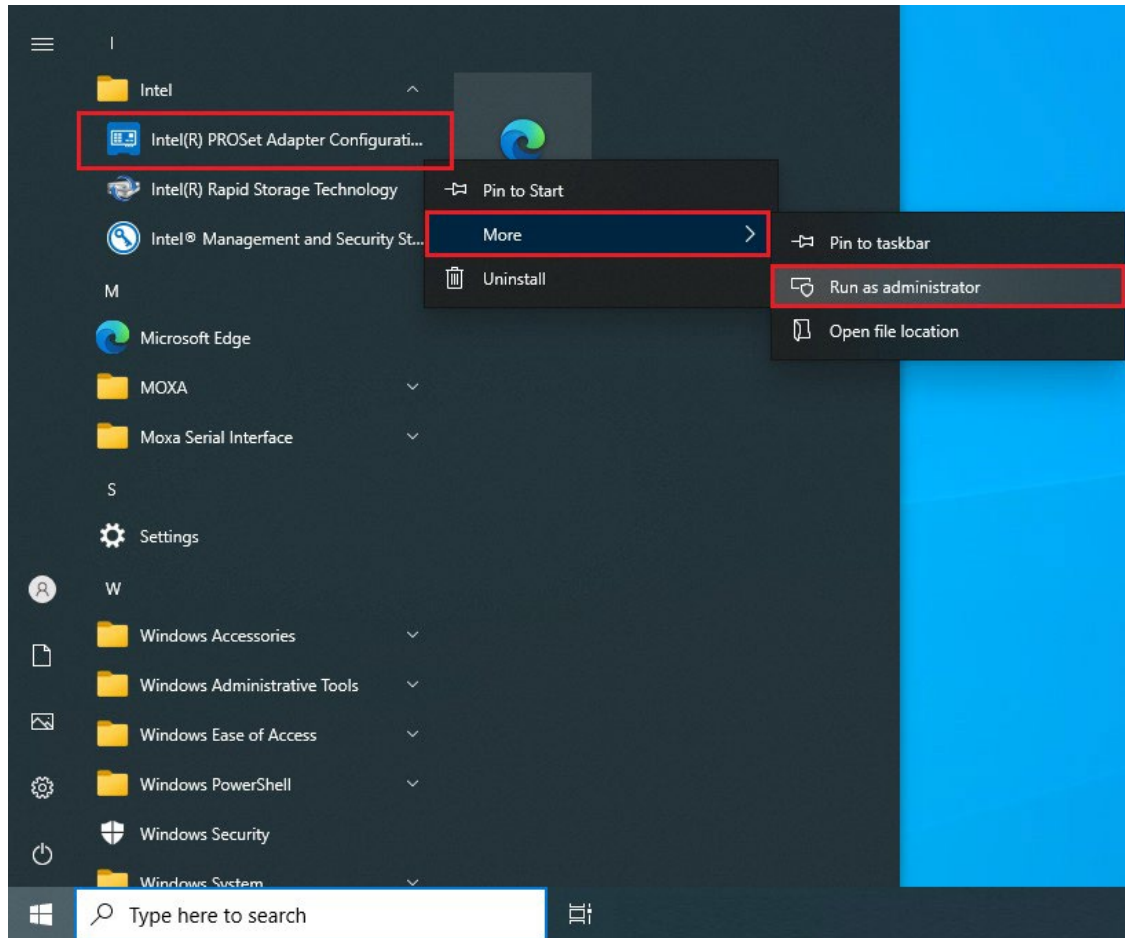
- Press **F10** to save the settings.

# 5. Teaming

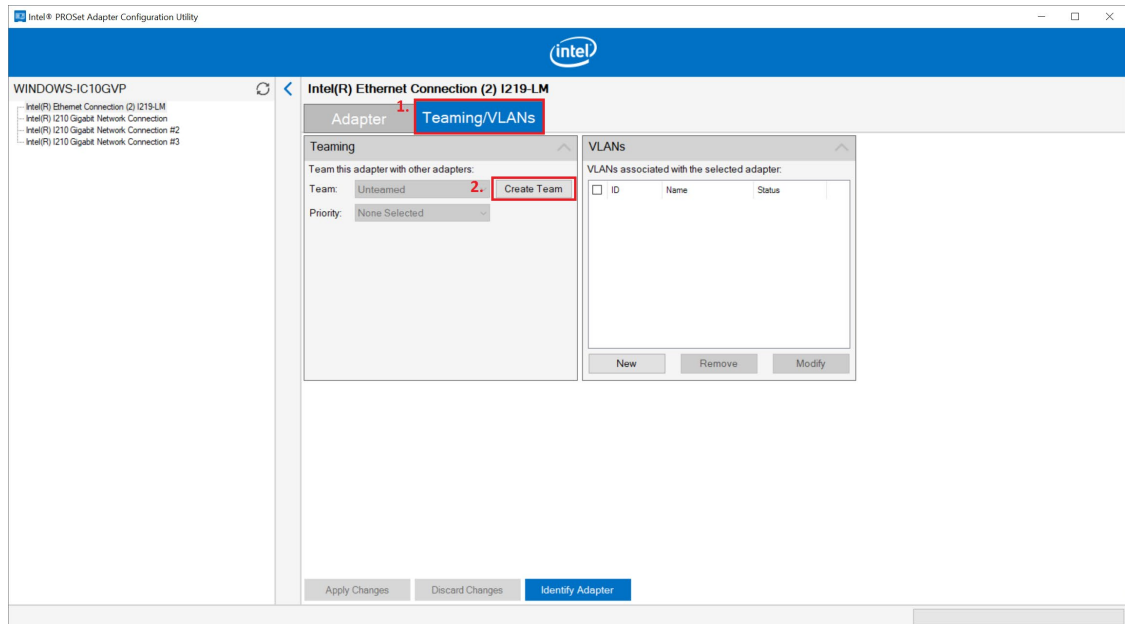
This chapter describes the setup process for the Intel® Teaming function.

## Creating an Intel® Net Team

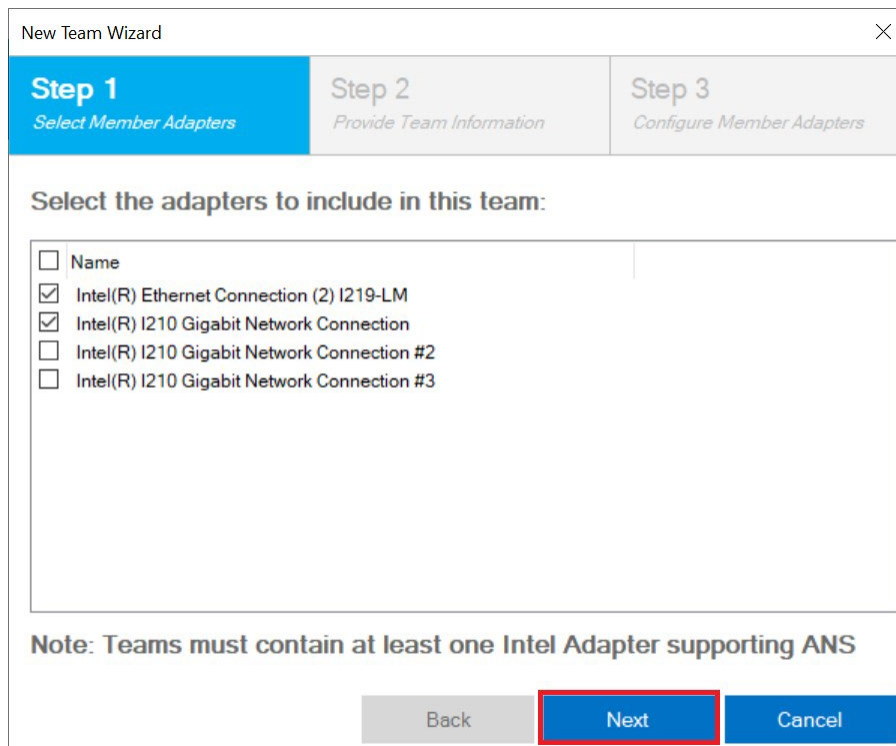
1. Run the **Intel® PROSet Adapter Configuration Utility** as administrator.



2. In the **Teaming/VLANs** tab, click **Create Team**.



3. Select the adapter to include in this team and click **Next**. An Intel ANS team can contain a maximum of eight members.



4. Name the team and select a team type. Click **Next** to continue.

New Team Wizard

Step 1  
*Select Member Adapters*

**Step 2**  
*Provide Team Information*

Step 3  
*Configure Member Adapters*

Please name the team and select a type:

Name:

Type:

- Adapter Fault Tolerance
- Adaptive Load Balancing
- IEEE 802.3ad Dynamic Link Aggregation
- Static Link Aggregation
- Switch Fault Tolerance

Note: All fields are required

Back Next Cancel



## NOTE

You cannot use an Intel AMT enabled adapter in a Dynamic Link Aggregation (DLA) team. You cannot use an Intel AMT enabled adapter in a Static Link Aggregation (SLA) team.

5. Select the primary and secondary adapters for the team and click **Finish** to create an Intel net team.

New Team Wizard

Step 1  
*Select Member Adapters*

Step 2  
*Provide Team Information*

**Step 3**  
*Configure Member Adapters*

Select Primary and Secondary Adapters for the team

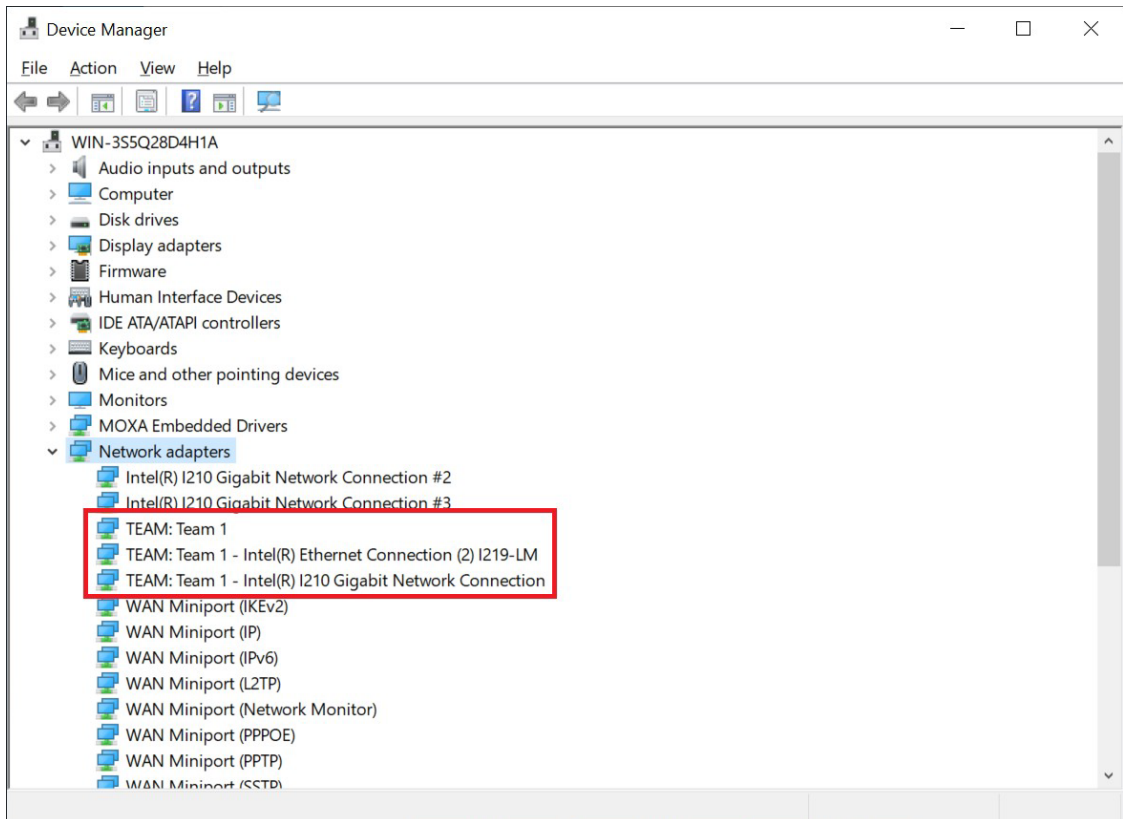
Primary Adapter:

Secondary Adapter:

- None Selected
- Intel(R) Ethernet Connection (2) I219-LM
- Intel(R) I210 Gigabit Network Connection

Back Finish Cancel

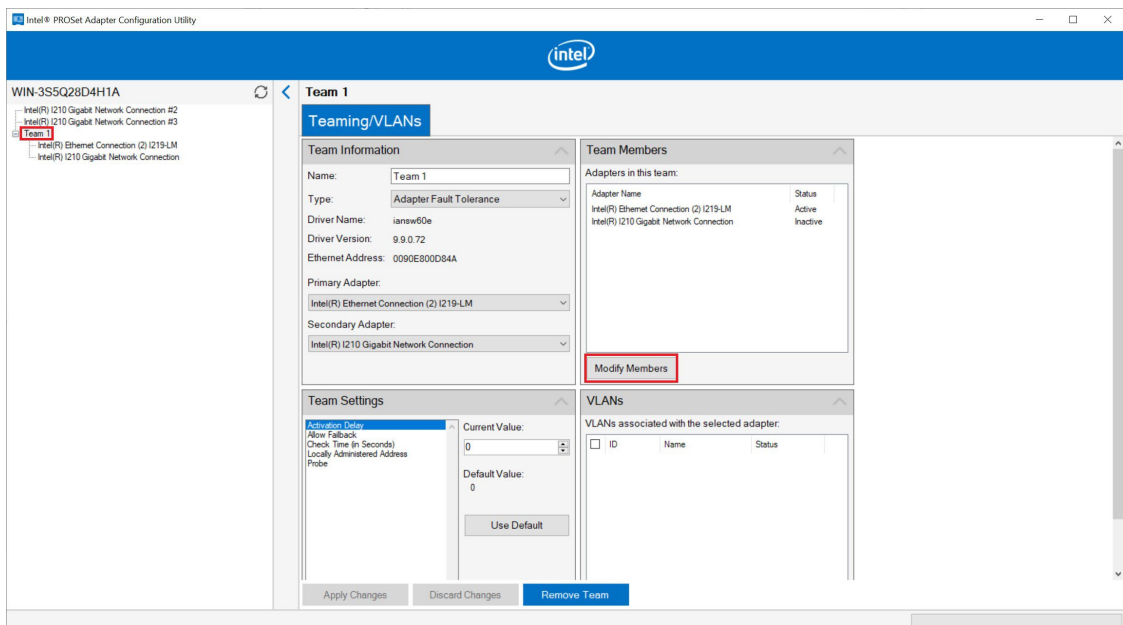
6. Check the **Network adapters** in the **Windows Device Manager**.



## Modifying an Intel® Net Team Member

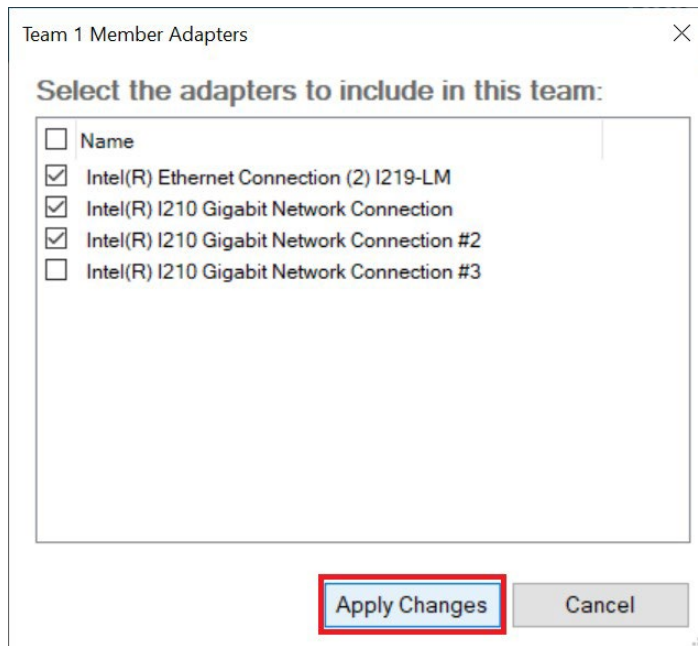
### Adding an Intel® Net Team Member

1. Select an Intel net team and click **Modify Members**.

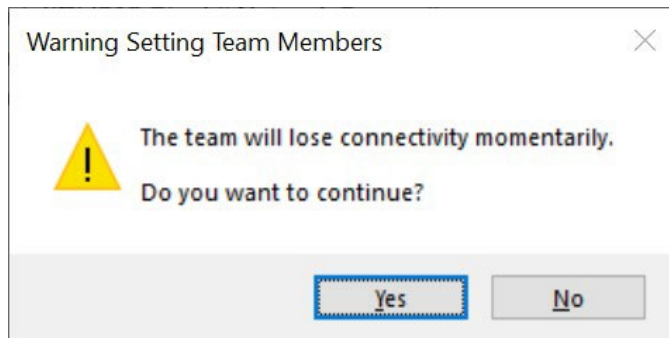




2. Select the adapters to include in this team and click **Apply Changes**.



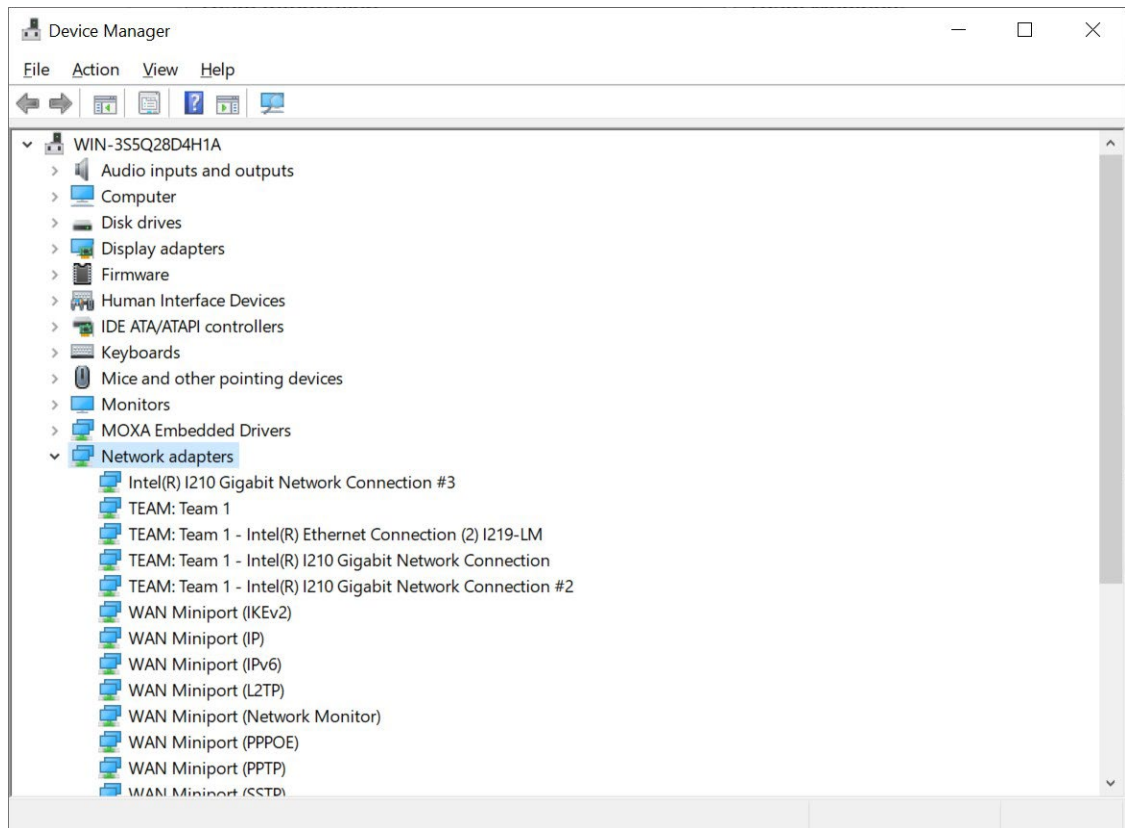
3. Click **Yes** to continue.



## NOTE

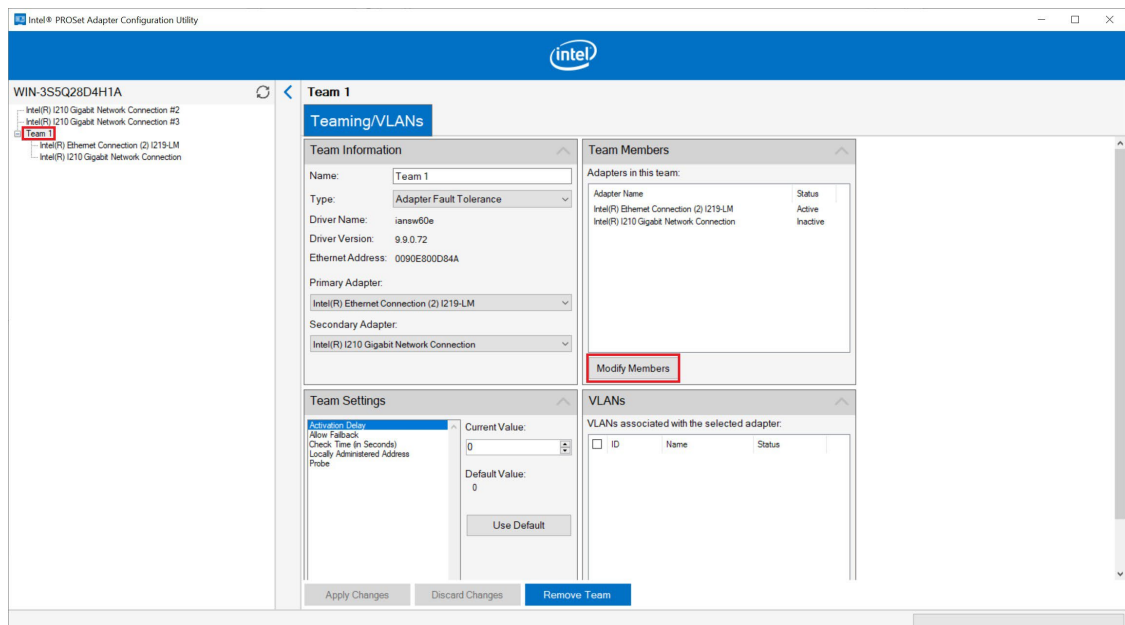
Modifying team members will cause the members to momentarily lose connectivity.

4. Check the **Network adapters** in the **Windows Device Manager**.

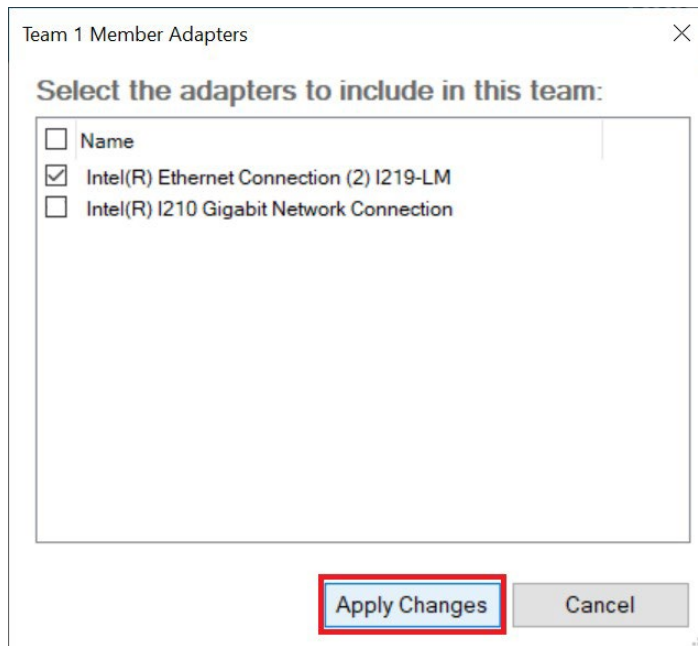


## Removing an Intel® Net Team Member

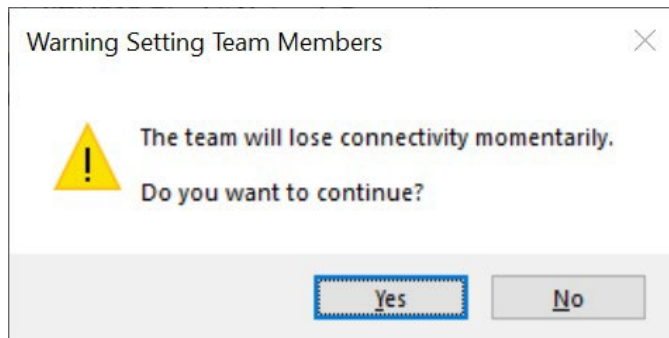
1. Select an Intel net team and click **Modify Members**.



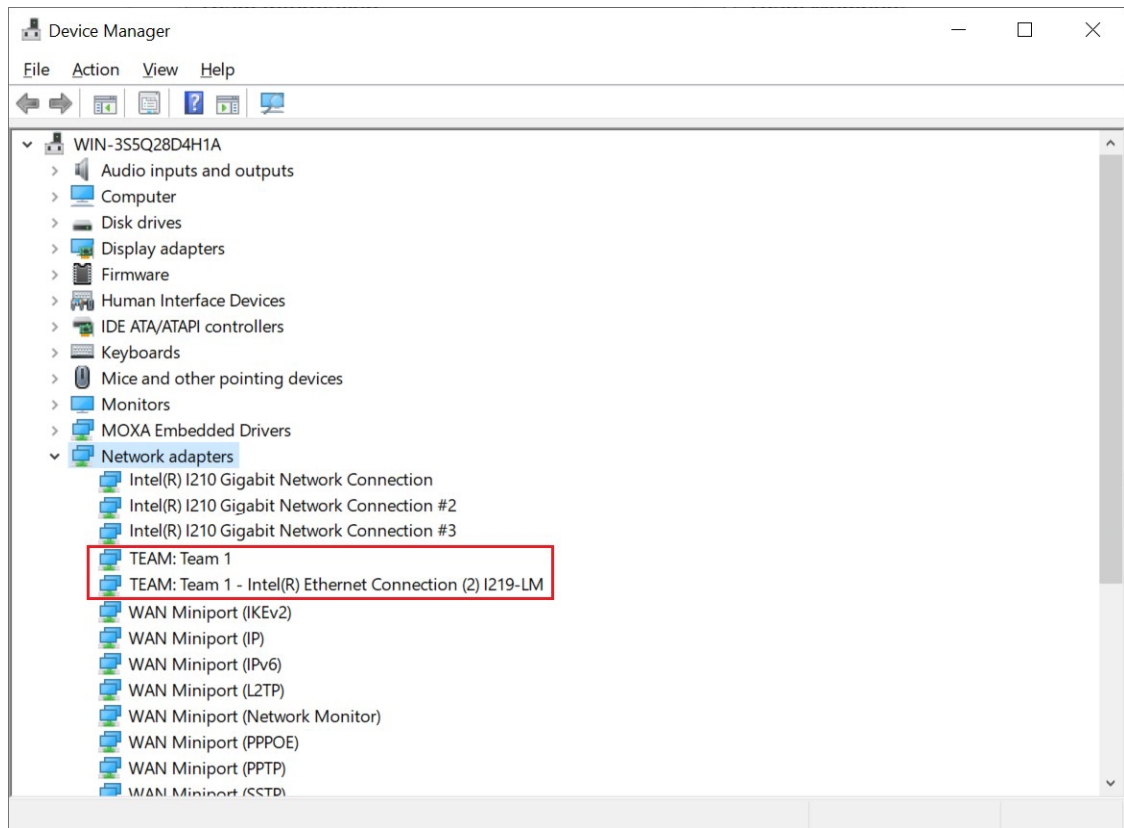
2. Uncheck the adapters you want to remove in this team and click **Apply Changes**.



3. Click **Yes** to continue.

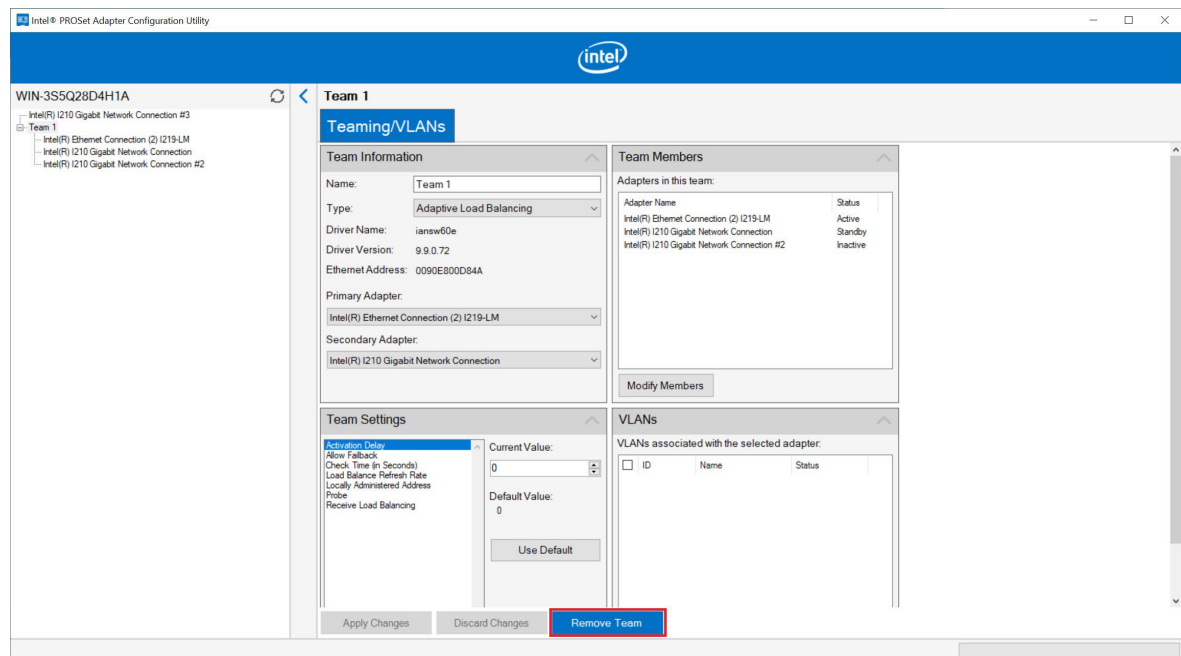


4. Check the **Network adapters** in the **Windows Device Manager**.



## Removing an Intel® Net Team

Select an Intel net team and click **Remove Team** to remove the specified Intel ANS team.



## 6. Unified Write Filter

This chapter describes how to use the Unified the Write Filter (UWF).

To use the UWF, you must first install the feature and enable it; the default is disable.

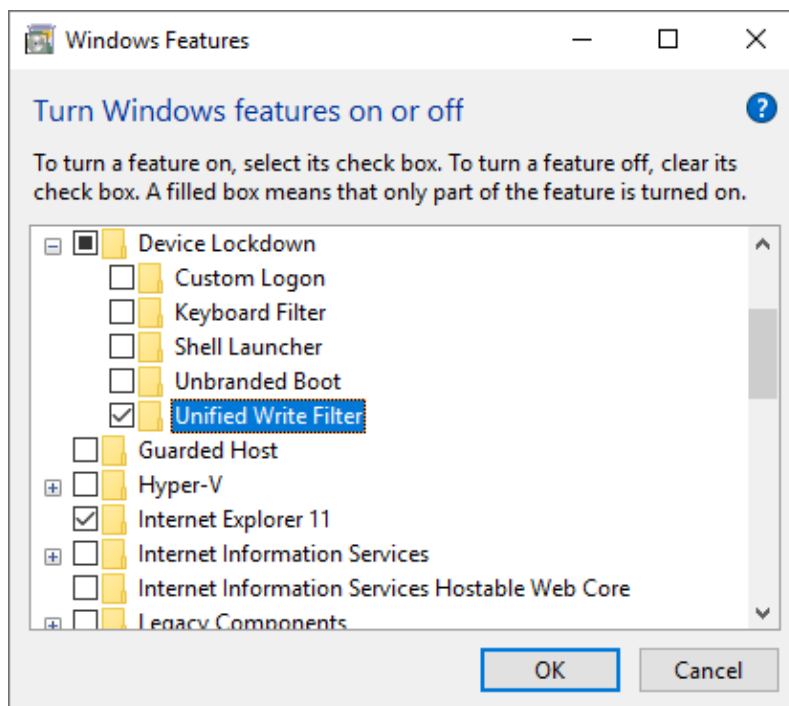
The first time you enable UWF on your device, UWF makes the following changes to your system to improve its performance:

- **Paging files are disabled.**
- **System restore is disabled.**
- **SuperFetch is disabled.**
- **File indexing service is turned off.**
- **Fast boot is disabled.**
- **Defragmentation service is turned off.**
- **BCD setting bootstatuspolicy is set to ignoreallfailures.**

After UWF is enabled, you can select a drive that you want to protect and start using UWF. UWF can help you manage PCs and devices remotely using WMI.

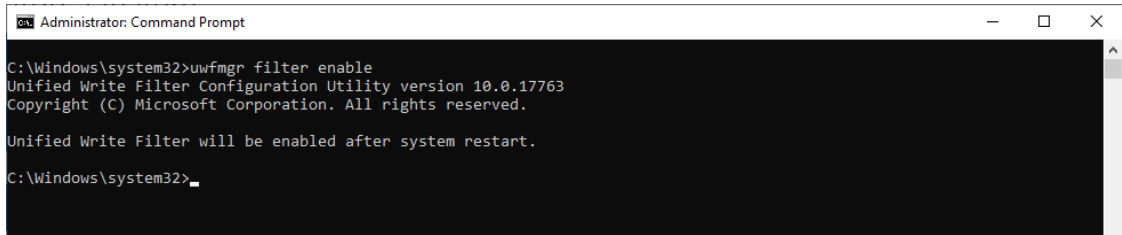
### Turning on UWF on a Running PC

1. Install UWF.
  - a. In the Windows **Start** window, type **Turn Windows features on or off**.
  - b. Open the **Windows Features** window and expand the **Device Lockdown** node.
  - c. Select **Unified Write Filter** and click **OK**.
  - d. Windows searches for the required files and displays a progress bar. Once the files are found, Windows applies the changes. When the changes are complete, a message to this effect is displayed.
  - e. Click **Close**.



2. Enable the following filter as an Administrator:

### cmd uwmgr filter enable



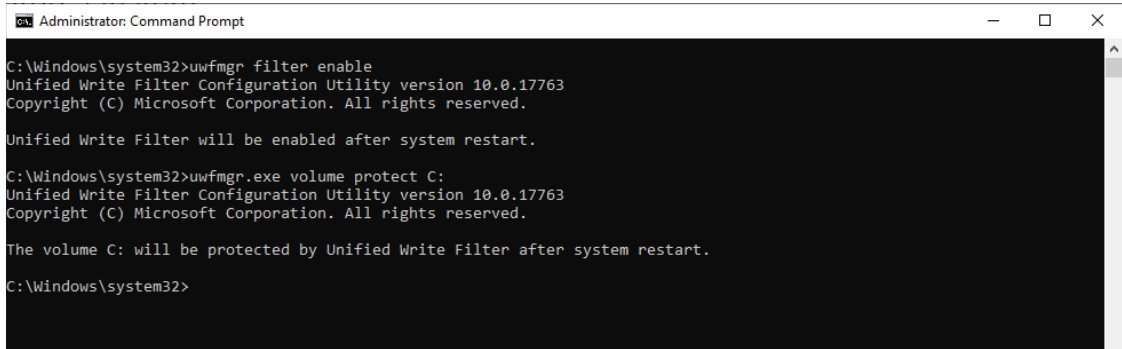
```
Administrator: Command Prompt
C:\Windows\system32>uwmgr filter enable
Unified Write Filter Configuration Utility version 10.0.17763
Copyright (C) Microsoft Corporation. All rights reserved.

Unified Write Filter will be enabled after system restart.

C:\Windows\system32>
```

3. Enable write protection for a drive:

### cmd uwmgr.exe volume protect C:



```
Administrator: Command Prompt
C:\Windows\system32>uwmgr filter enable
Unified Write Filter Configuration Utility version 10.0.17763
Copyright (C) Microsoft Corporation. All rights reserved.

Unified Write Filter will be enabled after system restart.

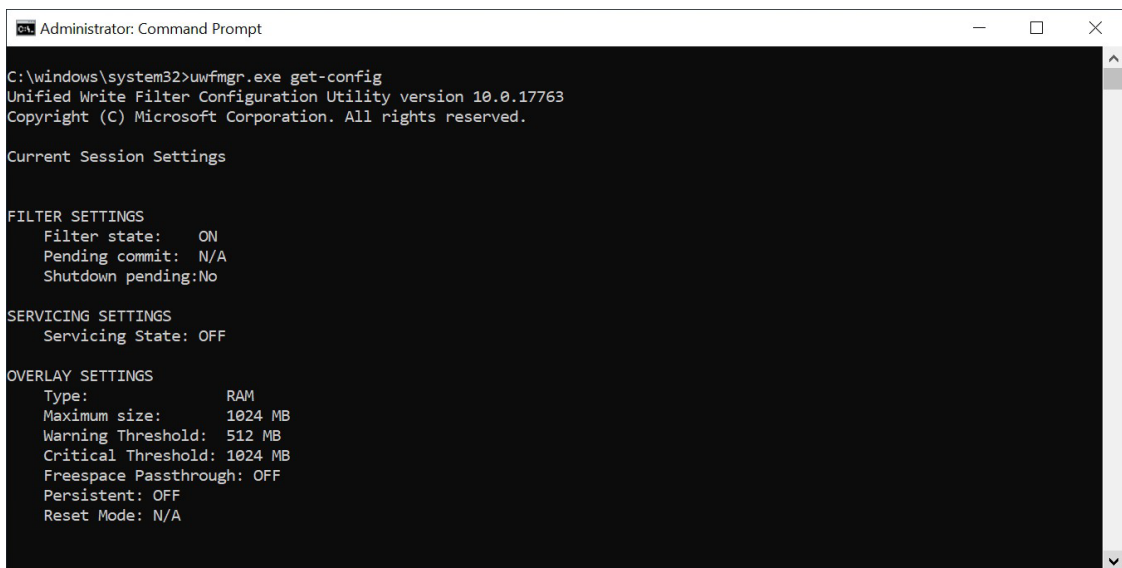
C:\Windows\system32>uwmgr.exe volume protect C:
Unified Write Filter Configuration Utility version 10.0.17763
Copyright (C) Microsoft Corporation. All rights reserved.

The volume C: will be protected by Unified Write Filter after system restart.

C:\Windows\system32>
```

4. Restart your computer.
5. Confirm that UWF is running:

### cmd uwmgr.exe get-config



```
Administrator: Command Prompt
C:\Windows\system32>uwmgr.exe get-config
Unified Write Filter Configuration Utility version 10.0.17763
Copyright (C) Microsoft Corporation. All rights reserved.

Current Session Settings

FILTER SETTINGS
  Filter state:    ON
  Pending commit: N/A
  Shutdown pending:No

SERVICING SETTINGS
  Servicing State: OFF

OVERLAY SETTINGS
  Type:           RAM
  Maximum size:   1024 MB
  Warning Threshold: 512 MB
  Critical Threshold: 1024 MB
  Freespace Passthrough: OFF
  Persistent:     OFF
  Reset Mode:     N/A
```

# Installing UWF Using WMI

If you have already installed Windows on your computer and you do not want to use a provisioning package, you can configure UWF by using Windows Management Instrumentation (WMI) providers. To turn on UWF using WMI, use the **UWF\_Filter** function, specifically the **UWF\_Filter.Enable** method in one of the following ways:

- Use the WMI providers directly in a PowerShell script
- Use the WMI providers directly in an application
- Use the command line tool, uwfmgr.exe



## NOTE

You must restart your computer after you turn on or turn off UWF for the changes to take effect.

You can also change the settings after you turn on UWF. For example, you can move the page file location to an unprotected volume and re-enable paging files.



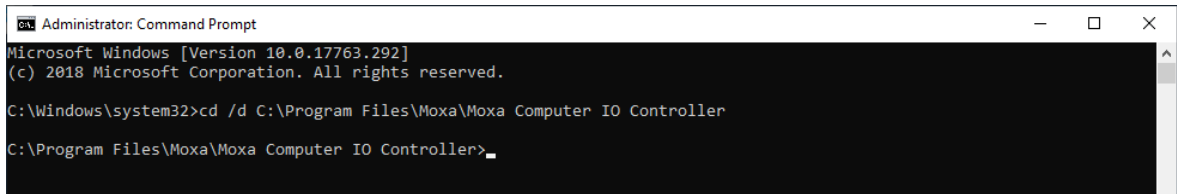
## IMPORTANT!

If you add UWF to your image by using SMI settings in the unattend.xml file, turning on UWF only sets the bootstatuspolicy BCD setting and turns off the defragmentation service. You must manually turn off the other features and services if you want to increase the performance of UWF.

After the device is restarted, UWF maintains configuration settings for the current session in a registry. UWF automatically excludes these registry entries from its filter. Static configuration changes do not take effect until after a device restart; the changes are saved in registry entries for use in the next session. Dynamic configuration changes occur immediately and persist after a device restart.

# 7. Moxa IO Controller Utility

This chapter describes how to use the Moxa IO Controller utility.



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.292]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd /d C:\Program Files\Moxa\Moxa Computer IO Controller

C:\Program Files\Moxa\Moxa Computer IO Controller>_
```

To use the Moxa IO Controller utility, first install the utility and enable the utility to configure the DIO, UART mode, and SIM settings. After the installation process is complete, run the Windows command prompt as an Administrator and change the path to C:\Program Files\Moxa\Moxa Computer IO Controller.

## Setting the DIO Status

Type the **mx-dio-ctl --help** command to see the instructions on using this utility and follow them to get or set the DIO status.



### IMPORTANT!

The DIN and DOUT indices start at 0. Even though the console output starts at 1, the indices still start at 0.



```
Administrator: Command Prompt

C:\Program Files\Moxa\Moxa IO Controller>mx-dio-ctl.exe --help
mx-dio-ctl 1.2.2201.10100
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get value from DIN port 1:
  mx-dio-ctl -i 1
Get value from DOUT port 1:
  mx-dio-ctl -o 1
Set DOUT port 2 value to HIGH:
  mx-dio-ctl -m 1 -o 2

-i      (Group: Index) -i <#DIN index> (Start from 0)
-o      (Group: Index) -o <#DOUT index> (Start from 0)
-m      -m <status>
        0 --> LOW
        1 --> HIGH

--help  Display this help screen.
--version Display version information.

C:\Program Files\Moxa\Moxa IO Controller>_
```



## Example:

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa Computer IO Controller>mx-dio-ctl -i 0
DIN port 0 status: 1

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-dio-ctl -o 0
DOUT port 0 status: 1

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-dio-ctl -o 0 -m 0
DOUT port 0 status: 0

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-dio-ctl -i 0
DIN port 0 status: 0

C:\Program Files\Moxa\Moxa Computer IO Controller>_
```

## Setting the UART Mode

Type the **mx-uart-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the UART mode.



### IMPORTANT!

The UART index starts from **0**. Even though the console output starts at 1, the index still starts at 0.

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-dio-ctl -i 0
DIN port 0 status: 1

C:\Program Files\Moxa\Moxa IO Controller>mx-uart-ctl.exe --help
mx-uart-ctl 1.4.2201.10000
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get uart mode from port 2:
  mx-uart-ctl -p 2
Set port 1 to mode RS-422:
  mx-uart-ctl -m 2 -p 1

  -p          Required. -p <#port index> (Start from 0)

  -m          -m <#uart mode>
              0 --> set to RS232 mode
              1 --> set to RS485-2W mode
              2 --> set to RS422/RS485-4W mode

  --help     Display this help screen.
  --version  Display version information.

C:\Program Files\Moxa\Moxa IO Controller>_
```

## Example:

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa Computer IO Controller>mx-uart-ctl -p 0
Current uart mode is RS232 interface.

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-uart-ctl -p 0 -m 1
Set OK.

Current uart mode is RS485-2W interface.

C:\Program Files\Moxa\Moxa Computer IO Controller>_
```

# Setting the Relay Status

Type the **mx-relay-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the relay status.



## IMPORTANT!

The relay index starts from **0**. Even though the console output starts at 1, the index still starts at 0.

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-relay-ctl.exe --help
mx-relay-ctl 1.0.1905.0
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get value from relay index 1:
  mx-relay-ctl -i 1
Turn on relay index 2:
  mx-relay-ctl -i 2 -m 1

-i      Required. -i <#Relay index> (Start from 0)
-m      -m <status>
        0 --> turn off
        1 --> turn on

--help  Display this help screen.
--version Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa Computer IO Controller>mx-relay-ctl -i 0
Relay index 0 data: 1

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-relay-ctl -i 0 -m 0
Relay index 0 data: 0

C:\Program Files\Moxa\Moxa Computer IO Controller>
```

# Setting the LED Status

Type the **mx-led-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the LED status.



## IMPORTANT!

The relay index starts from **0**. Even though the console output starts at 1, the index still starts at 0.

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-led-ctl.exe --help
mx-led-ctl 1.0.1905.0
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get value from LED index 1:
  mx-led-ctl -i 1
Turn on LED index 2:
  mx-led-ctl -i 2 -m 1
Set LED index 3 to blink mode:
  mx-led-ctl -i 3 -m 2

-i      Required. -i <#LED index> (Start from 0)

-m      -m <status>
        0 --> led off
        1 --> led on
        2 --> led blink

--help  Display this help screen.

--version  Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:

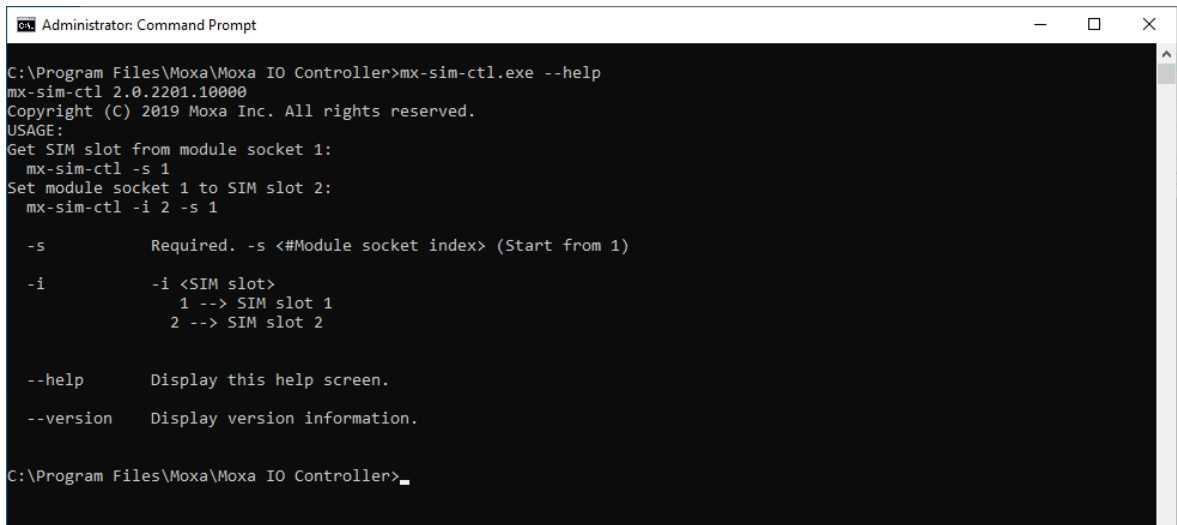
```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa Computer IO Controller>mx-led-ctl -i 0
LED index 0 data: 0

C:\Program Files\Moxa\Moxa Computer IO Controller>mx-led-ctl -i 0 -m 1
LED index 0 data: 1

C:\Program Files\Moxa\Moxa Computer IO Controller>
```

# Setting the SIM Status

Type the **mx-sim-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the status of the SIM card slot.



```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-sim-ctl.exe --help
mx-sim-ctl 2.0.2201.10000
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get SIM slot from module socket 1:
  mx-sim-ctl -s 1
Set module socket 1 to SIM slot 2:
  mx-sim-ctl -i 2 -s 1

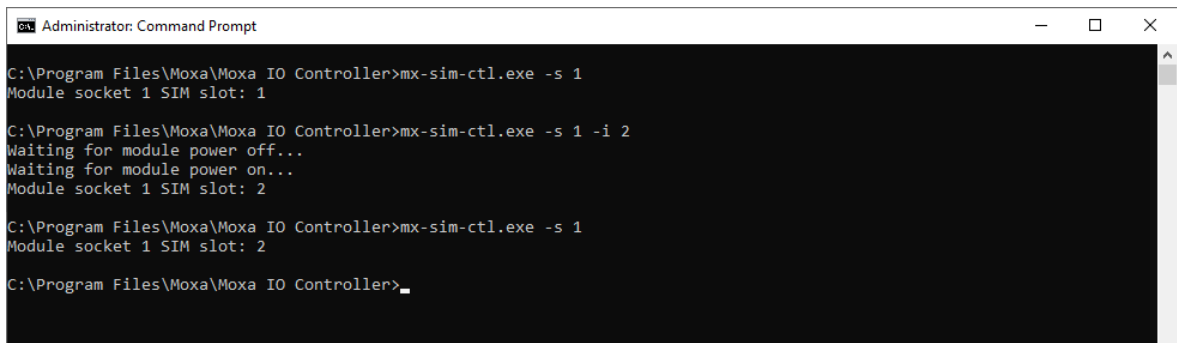
-s          Required. -s <#Module socket index> (Start from 1)

-i          -i <SIM slot>
            1 --> SIM slot 1
            2 --> SIM slot 2

--help     Display this help screen.
--version  Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:



```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-sim-ctl.exe -s 1
Module socket 1 SIM slot: 1

C:\Program Files\Moxa\Moxa IO Controller>mx-sim-ctl.exe -s 1 -i 2
Waiting for module power off...
Waiting for module power on...
Module socket 1 SIM slot: 2

C:\Program Files\Moxa\Moxa IO Controller>mx-sim-ctl.exe -s 1
Module socket 1 SIM slot: 2

C:\Program Files\Moxa\Moxa IO Controller>
```

# Setting PCIE Slot Power Status

Type the **mx-pcie-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the status of the PCIE slot power.



## IMPORTANT!

The PCIE Slot index starts from **0**. Even though the console output starts at 1, the index still starts at 0.

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-pcie-ctl.exe --help
mx-pcie-ctl 1.2.1907.10000
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Get pcie power status from pcie slot 1:
  mx-pcie-ctl -i 1
Set pcie slot 1 power on:
  mx-pcie-ctl -i 1 -m 1
Set pcie slot 1 power on with delay Time 200ms:
  mx-pcie-ctl -i 1 -m 1 -t 200

-i      Required. -i <#PCIE Slot index> (Start from 0)

-m      -m <pcie power status>
        0 --> PCIE power off
        1 --> PCIE power on

-t      -t <#PCIE Reset Delay time(ms)>

--help  Display this help screen.

--version Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-pcie-ctl.exe -i 0
PCIE slot 0 power status: 1

C:\Program Files\Moxa\Moxa IO Controller>mx-pcie-ctl.exe -i 0 -m 0
PCIE slot 0 power status: 0

C:\Program Files\Moxa\Moxa IO Controller>
```

# Setting PCIE Reset Pin Status

Type the **mx-pciereset-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the PCIE reset pin status and delay time.



## IMPORTANT!

The PCIE reset pin index starts from **0**. Even though the console output starts at 1, the index still starts at 0.

```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-pciereset-ctl.exe --help
mx-reset-ctl 2.0.2203.10000
Copyright (C) 2019 Moxa Inc. All rights reserved.
USAGE:
Reset PCIE slot 1 :
  mx-pciereset-ctl -i 1
Reset PCIE slot 1 DelayTime 200ms:
  mx-pciereset-ctl -i 1 -t 200

-i      Required. -i <#PCIE Reset Slot index> (Start from 0)
-t      -t <#PCIE Reset Delay time(ms)>
--help  Display this help screen.
--version Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:

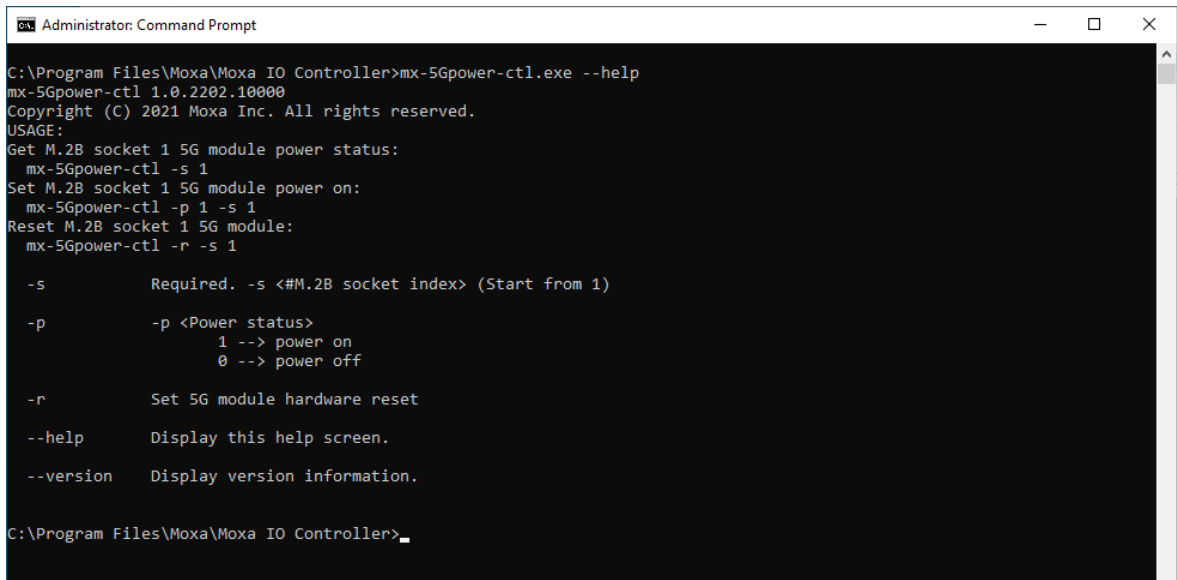
```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-pciereset-ctl.exe -i 0
PCIE slot 0 reset status: 1

C:\Program Files\Moxa\Moxa IO Controller>mx-pciereset-ctl.exe -i 0 -t 200
PCIE slot 0 reset status: 1

C:\Program Files\Moxa\Moxa IO Controller>
```

# Setting M.2 B Key Socket 5G Module Power Status

Type the **mx-5Gpower-ctl --help** command to see instructions on using this utility and follow the onscreen instructions to get or set the status of the M.2 B Key socket power.



```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-5Gpower-ctl.exe --help
mx-5Gpower-ctl 1.0.2202.10000
Copyright (C) 2021 Moxa Inc. All rights reserved.
USAGE:
Get M.2B socket 1 5G module power status:
  mx-5Gpower-ctl -s 1
Set M.2B socket 1 5G module power on:
  mx-5Gpower-ctl -p 1 -s 1
Reset M.2B socket 1 5G module:
  mx-5Gpower-ctl -r -s 1

-s      Required. -s <#M.2B socket index> (Start from 1)

-p      -p <Power status>
        1 --> power on
        0 --> power off

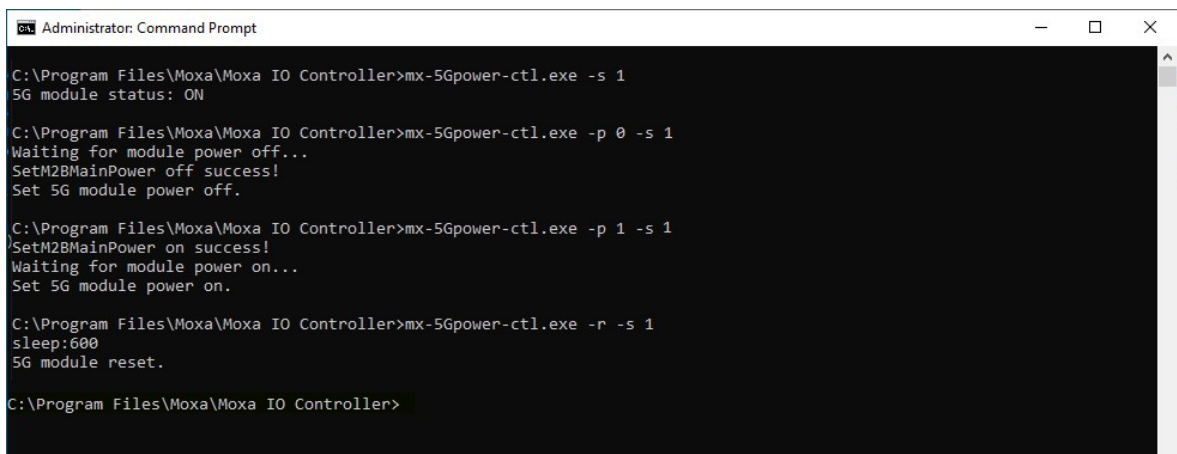
-r      Set 5G module hardware reset

--help  Display this help screen.

--version  Display version information.

C:\Program Files\Moxa\Moxa IO Controller>
```

## Example:



```
Administrator: Command Prompt
C:\Program Files\Moxa\Moxa IO Controller>mx-5Gpower-ctl.exe -s 1
5G module status: ON

C:\Program Files\Moxa\Moxa IO Controller>mx-5Gpower-ctl.exe -p 0 -s 1
Waiting for module power off...
SetM2BMainPower off success!
Set 5G module power off.

C:\Program Files\Moxa\Moxa IO Controller>mx-5Gpower-ctl.exe -p 1 -s 1
SetM2BMainPower on success!
Waiting for module power on...
Set 5G module power on.

C:\Program Files\Moxa\Moxa IO Controller>mx-5Gpower-ctl.exe -r -s 1
sleep:600
5G module reset.

C:\Program Files\Moxa\Moxa IO Controller>
```

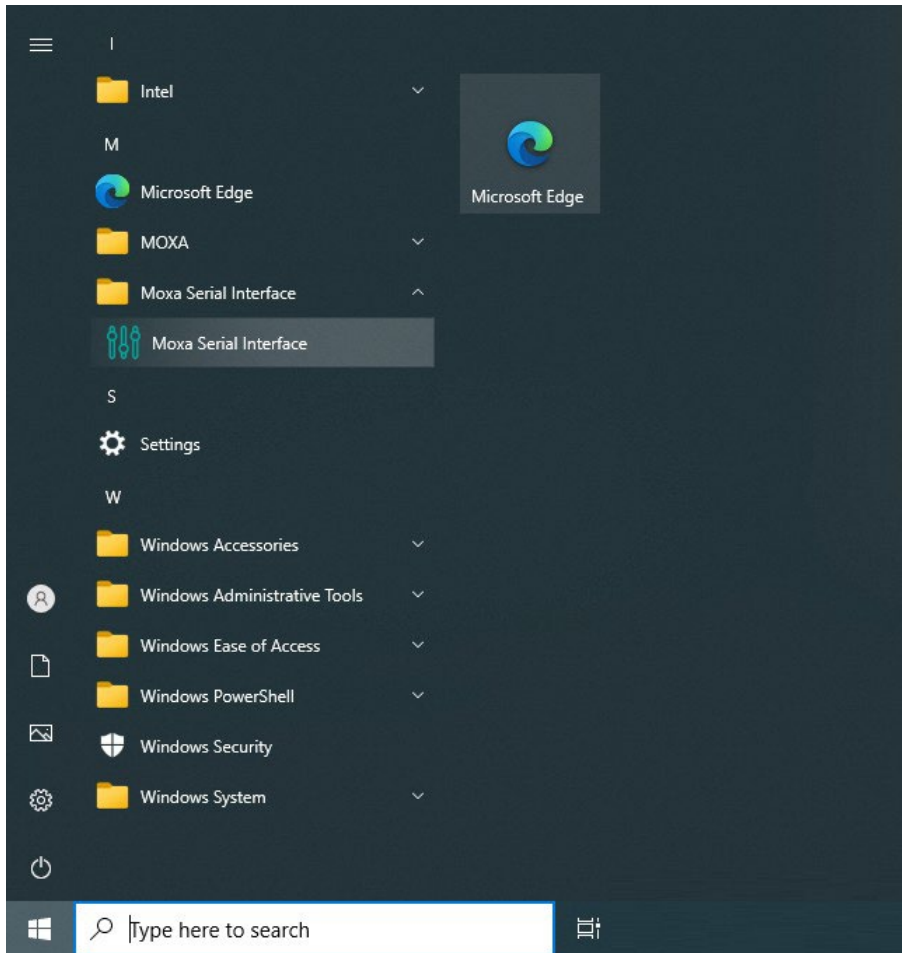
# 8. Moxa Serial Interface Utility

---

In this chapter, we describe how to use the Moxa Serial Interface utility to set the UART mode in your computer's serial interface.

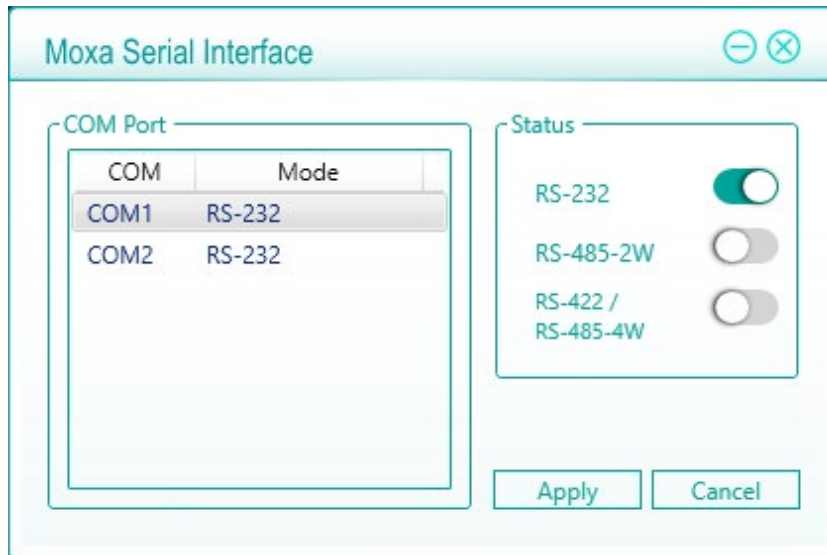
## Setting the Serial Port Mode

1. Install the Moxa Serial Interface utility.
2. From the Windows Start menu, run the **Moxa Serial Interface utility**.





3. Select the target COM port and UART mode and click **Apply** to save the settings.

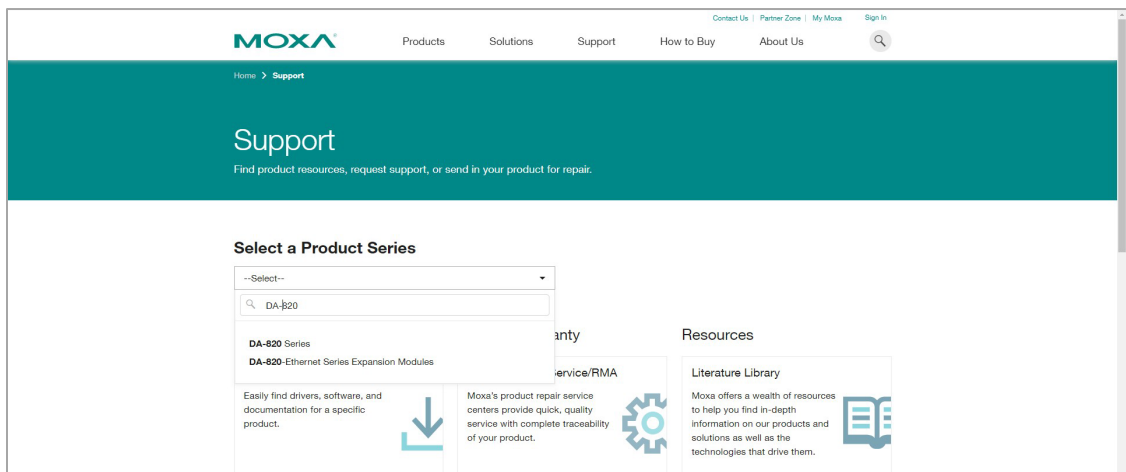


# 9. IO Control API

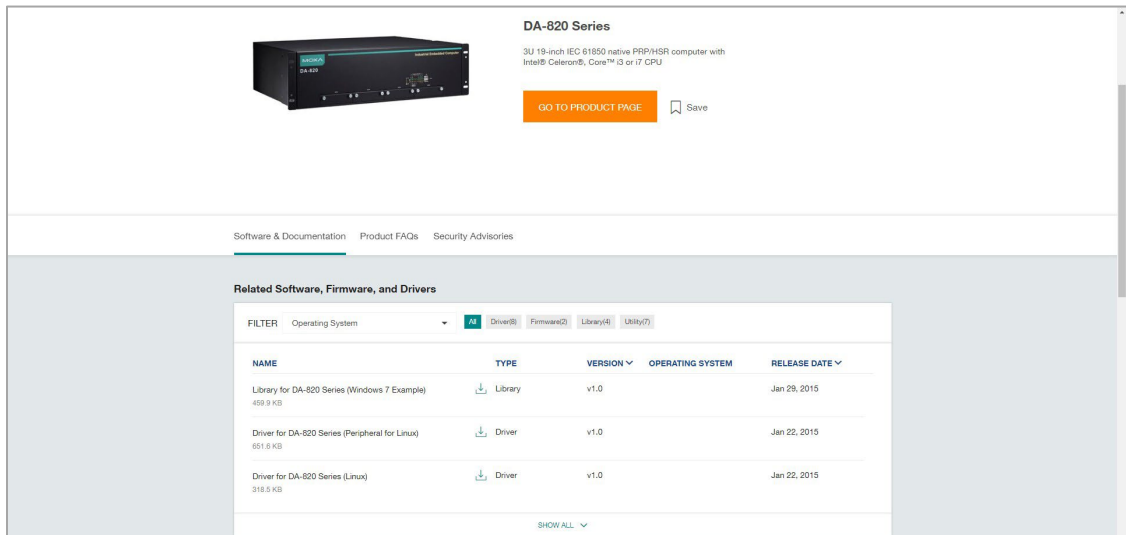
This chapter describes how to use the IO Control API.

## Downloading the API

1. Access the Moxa support page: <https://www.moxa.com/en/support>
2. Select the product series (e.g., DA-820).



3. Download the related files.



# Supported Series

## **mxdgio**

- DA-681C Series
- DA-682C Series
- DA-820C Series
- V2403C Series
- V2406C Series
- MC-3201 Series

## **mxsp**

- DA-681C Series
- DA-682C Series
- DA-820C Series
- V2403C Series
- V2406C Series
- MC-1200 Series
- MC-3201 Series

## **mxrelay**

- DA-681C Series
- DA-682C Series
- DA-820C Series

## **mxled**

- DA-681C Series
- DA-682C Series
- DA-820C Series
- V2403C Series

## **mxsim**

- V2403C Series
- V2406C Series
- MC-1200 Series
- MC-3201 Series

## **mxpcie**

- V2403C Series
- V2406C Series
- MC-1200 Series
- MC-3201 Series

## **mxpciereset**

- MC-3201 Series

## **mx5Gpower**

- MC-3201 Series

## mxwdg

- DA-681C Series
- DA-682C Series
- DA-820C Series
- V2403C Series
- V2406C Series
- MC-1200 Series
- MC-3201 Series

## mxdgio

The **mxdgio** library operates on the digital I/Os and consists of the following:

- **GetDinStatus**
- **GetDoutStatus**
- **SetDoutStatus**

## GetDinStatus

### **Syntax**

```
int GetDinStatus(int port);
```

### **Description**

Gets the status of a digital input port.

### **Parameters**

*port*: The index of the digital input port; starts at 0.

### **Return Value**

The status of the digital input port; 0 for low and 1 for high.

### **Error codes**

The following error codes can be retrieved by the **DIO\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxdgio library initialization failed. Cannot open the json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### **Requirements**

Name	Items
Header	mxdgio.h
Library	mxdgio.lib
DLL	mxdgio.dll
Profile	MxdgioProfile [ <i>ModelName</i> ].json

# GetDoutStatus

## Syntax

```
int GetDoutStatus(int port);
```

## Description

Gets the status of a digital output port.

## Parameters

*port*: The index of the digital output port; starts at 0.

## Return Value

The status of the digital output port; 0 for low and 1 for high.

## Error codes

The following error codes can be retrieved by the **DIO\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxdbgio library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

## Requirements

Name	Items
Header	mxdbgio.h
Library	mxdbgio.lib
DLL	mxdbgio.dll
Profile	MxdgioProfile [ModelName].json

# SetDoutStatus

## Syntax

```
int SetDoutStatus(int port, int status);
```

## Description

Sets the status of a digital output port.

## Parameters

*port*: The index of the digital output port; starts at 0.

*status*: The status of the digital output port; 0 for low and 1 for high.

## Return Value

Returns the value 0 if the digital output status is successfully set.

## Error codes

The following error codes can be retrieved by the **DIO\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxdbgio library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## Requirements

Name	Items
Header	mxdbgio.h
Library	mxdbgio.lib
DLL	mxdbgio.dll
Profile	MxdgioProfile [ModelName].json

# mxsp

The **mxsp** library operates on the serial port and consists of the following:

- **GetUartMode**
- **SetUartMode**

## GetUartMode

### Syntax

```
int GetUartMode(int port);
```

### Description

Gets the status of the UART port.

### Parameters

*port*: The index of the UART port; starts at 0.

### Return Value

The mode of a UART interface; 0 for RS-232, 1 for RS-485-2W, and 2 for RS-422/RS-485-4W.

### Error codes

The following error codes can be retrieved by the **UART\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxsp library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxsp.h
Library	mxsp.lib
DLL	mxsp.dll
Profile	MxspProfile [Mode/Name].json

# SetUartMode

## **Syntax**

```
int SetUartMode(int port, int mode);
```

## **Description**

Sets the status of the UART port.

## **Parameters**

*port*: The index of the UART port; starts at 0.

*mode*: The mode of a UART interface; 0 for RS-232, 1 for RS-485-2w, and 2 for RS-422/RS-485-4W.

## **Return Value**

Returns 0 if the UART mode is successfully set.

## **Error codes**

The following error codes can be retrieved by the **UART\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxsp library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxsp.h
Library	mxsp.lib
DLL	mxsp.dll
Profile	MxspProfile [Mode/Name].json

# mxrelay

The **mxrelay** library operates on the relay output and consists of the following:

- **GetRelayData**
- **SetRelayData**

## GetRelayData

### Syntax

```
int GetRelayData(int port);
```

### Description

Gets the status of the relay output port.

### Parameters

*port*: The index of the relay output port; starts at 0.

### Return Value

The status of a relay output port; 0 for OFF, 1 for ON.

### Error codes

The following error codes can be retrieved by the **RELAY\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxrelay library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxrelay.h
Library	mxrelay.lib
DLL	mxrelay.dll
Profile	MxrelayProfile [ModelName].json



# SetRelayData

## **Syntax**

```
int SetRelayData(int port, int status);
```

## **Description**

Sets the status of the relay output port.

## **Parameters**

*port*: The index of the relay output port; starts at 0. *status*: The status of a relay output; 0 for OFF, 1 for ON.

## **Return Value**

Returns 0 if the status of the relay output is successfully set.

## **Error codes**

The following error codes can be retrieved by the **RELAY\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxrelay library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxrelay.h
Library	mxrelay.lib
DLL	mxrelay.dll
Profile	MxrelayProfile [ModelName].json

# mxled

The **mxled** library operates on the relay output and consists of the following:

- **GetLedData**
- **SetLedData**

## GetLedData

### Syntax

```
int GetLedData(int port);
```

### Description

Gets the status of the LED port.

### Parameters

*port*: The index of the LED port; starts at 0.

### Return Value

The status of a LED port; 0 for OFF, 1 for ON.

### Error codes

The following error codes can be retrieved by the **LED\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxled library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxled.h
Library	mxled.lib
DLL	mxled.dll
Profile	MxledProfile [ <i>ModelName</i> ].json

# SetLedData

## **Syntax**

```
int SetLedData(int port, int status);
```

## **Description**

Sets the status of the LED port.

## **Parameters**

*port*: The index of the LED port; starts at 0.

*status*: The status of the LED; 0 for OFF, 1 for ON, and 2 for blinking.

## **Return Value**

Returns 0 if the LED status is set successfully.

## **Error codes**

The following error codes can be retrieved by the **LED\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxled library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxled.h
Library	mxled.lib
DLL	mxled.dll
Profile	MxledProfile [ModelName].json

# mxsim

The **mxsim** library operates on the relay output and consists of the following:

- **GetSIMSlot**
- **SetSIMSlot**

## GetSIMSlot

### Syntax

```
int GetSIMSlot(int port);
```

### Description

Gets the SIM card slot index number.

### Parameters

*port*: The index of the SIM card slot; starts at 0.

### Return Value

The SIM card slot number; 1 for SIM 1 slot, 2 for SIM 2 slot.

### Error codes

The following error codes can be retrieved by the **SIM\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxsim library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxsim.h
Library	mxsim.lib
DLL	mxsim.dll
Profile	MxsimProfile [ <i>ModelName</i> ].json

# SetSIMSlot

## **Syntax**

```
int SetSIMSlot(int port, int slot);
```

## **Description**

Sets the SIM card slot.

## **Parameters**

*port*: The index of the SIM card slot; starts at 0.

*slot*: The SIM card slot of a SIM number; 0 is SIM 1 slot, 1 is SIM 2 slot.

## **Return Value**

Returns 0 if the SIM card slot of the SIM number is successfully set.

## **Error codes**

The following error codes can be retrieved by the **SIM\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxsim library initialization failed. Cannot open the json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxsim.h
Library	mxsim.lib
DLL	mxsim.dll
Profile	MxsimProfile [ModelName].json

# mxpcie

The **mxpcie** library operates on the relay output and consists of the following:

- **GetPCIESlotStatus**
- **SetPCIESlotStatus**
- **SetPCIESlotStatus**

## GetPCIESlotStatus

### Syntax

```
int GetPCIESlotStatus(int port);
```

### Description

Gets the PCIE slot power status.

### Parameters

*port*: The index of the PCIE slot; starts at 0.

### Return Value

The status of a PCIE slot power; 0 for OFF, 1 for ON.

### Error codes

The following error codes can be retrieved by the **PCIE\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxpcie library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxpcie.h
Library	mxpcie.lib
DLL	mxpcie.dll
Profile	MxpcieProfile [ <i>ModelName</i> ].json

# SetPCIESlotStatus

## **Syntax**

```
int SetPCIESlotStatus(int port, int status);
```

## **Description**

Sets the PCIE slot power status.

## **Parameters**

*port*: The index of the PCIE slot; starts at 0.

*status*: The status of the PCIE slot power; 0 for OFF, 1 for ON.

## **Return Value**

Returns 0 if the PCIE slot power is successfully set.

## **Error codes**

The following error codes can be retrieved by the **PCIE\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxpcie library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxpcie.h
Library	mxpcie.lib
DLL	mxpcie.dll
Profile	MxpcieProfile [ <i>ModelName</i> ].json

# SetPCIESlotStatusWithReset

## **Syntax**

```
int SetPCIESlotStatusWithReset(int port, int status, int time);
```

## **Description**

Sets the PCIE slot power status and PCIE slot reset pin turn ON and OFF.

## **Parameters**

*port*: The index of the PCIE slot; starts at 0.

*status*: The status of the PCIE slot power and PICE reset pin; 0 for OFF, 1 for ON.

*time*: The delay time between PCIE slot reset pin turn ON and OFF.

## **Return Value**

Returns 0 if the PCIE slot power and PCIE reset pin are successfully set.

## **Error codes**

The following error codes can be retrieved by the **PCIE\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxpcie library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxpcie.h
Library	mxpcie.lib
DLL	mxpcie.dll
Profile	MxpcieProfile [ <i>ModelName</i> ].json



# mxpciereset

The **mxpciereset** library operates on the relay output and consists of the following:

- **GetRESETSlotStatus**
- **SetRESETSlotStatus**

## GetRESETSlotStatus

### Syntax

### Description

Gets the PCIE slot reset pin status.

### Parameters

*port*: The index of the PCIE slot; starts at 0.

### Return Value

The status of a PCIE slot reset pin; 0 for OFF, 1 for ON.

### Error codes

The following error codes can be retrieved by the **RESET\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxpciereset library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.

### Requirements

Name	Items
Header	mxpciereset.h
Library	mxpciereset.lib
DLL	mxpciereset.dll
Profile	MxpcieresetProfile [ <i>ModelName</i> ].json

# SetRESETSlotStatus

## **Syntax**

```
int SetRESETSlotStatus(int port, int time);
```

## **Description**

Sets the PCIE slot reset pin ON/OFF cycle and delay time.

## **Parameters**

*port*: The index of the PCIE slot; starts at 0.

*time*: The delay time between PCIE slot reset pin turn ON and OFF.

## **Return Value**

Returns 0 if the PCIE slot reset pin is successfully set.

## **Error codes**

The following error codes can be retrieved by the **RESET\_STATUS** function.

Name	Value	Meaning
LIB_INITIALIZE_FAIL	-1	The mxpcierreset library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-2	Target port index is out of range.
SET_STATUS_ERR	-3	Status setting failed or is defined with a bad format.

## **Requirements**

Name	Items
Header	mxpcierreset.h
Library	mxpcierreset.lib
DLL	mxpcierreset.dll
Profile	MxpcierresetProfile [ <i>ModelName</i> ].json

# mx5Gpower

The **mx5Gpower** library operates on the relay output and consists of the following:

- **GetPowerStatus**
- **SetPowerStatus**
- **ResetModule**

## GetPowerStatus

### Syntax

```
int GetPowerStatus(int port);
```

### Description

Gets the 5G module power status.

### Parameters

*port*: The index of the M.2 B key slot; starts at 0.

### Return Value

The status of a 5G module power; 0 for OFF, 1 for ON.

### Error codes

The following error codes can be retrieved by the **RETURN\_STATUS** function.

Name	Value	Meaning
ACTION_SUCCESS	0	Action success.
ACTION_ERROR	-1	Action error.
LIB_INITIALIZE_FAIL	-2	The mx5Gpower library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-3	Target port index is out of range.
SET_STATUS_ERR	-4	Status setting failed or is defined with a bad format.
INVALID_MODULE	-5	Invalid module.
ACTION_TIMEOUT	-6	Action timeout.

### Requirements

Name	Items
Header	mx5Gpower.h
Library	mx5Gpower.lib
DLL	mx5Gpower.dll
Profile	Mx5GpowerProfile [ <i>ModelName</i> ].json

# SetPowerStatus

## **Syntax**

```
int SetPowerStatus(int port, int status);
```

## **Description**

Sets the 5G module power status.

## **Parameters**

*port*: The index of the M.2 B key; starts at 0.

*status*: The status of a 5G module power; 0 for OFF, 1 for ON.

## **Return Value**

Returns 0 if the 5G module power status is successfully set.

## **Error codes**

The following error codes can be retrieved by the **RETURN\_STATUS** function.

Name	Value	Meaning
ACTION_SUCCESS	0	Action success.
ACTION_ERROR	-1	Action error.
LIB_INITIALIZE_FAIL	-2	The mx5Gpower library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-3	Target port index is out of range.
SET_STATUS_ERR	-4	Status setting failed or is defined with a bad format.
INVALID_MODULE	-5	Invalid module.
ACTION_TIMEOUT	-6	Action timeout.

## **Requirements**

Name	Items
Header	mx5Gpower.h
Library	mx5Gpower.lib
DLL	mx5Gpower.dll
Profile	Mx5GpowerProfile [ <i>ModelName</i> ].json

# ResetModule

## **Syntax**

```
int ResetModule(int port);
```

## **Description**

Resets the 5G module power status. Run the 5G module power cycle.

## **Parameters**

*port*: The index of the M.2 B key; starts at 0.

## **Return Value**

Returns 0 if the 5G module power status is successfully set.

## **Error codes**

Name	Value	Meaning
ACTION_SUCCESS	0	Action success.
ACTION_ERROR	-1	Action error.
LIB_INITIALIZE_FAIL	-2	The mx5Gpower library initialization failed. Cannot open json profile.
PORT_OUTOF_INDEX	-3	Target port index is out of range.
SET_STATUS_ERR	-4	Status setting failed or is defined with a bad format.
INVALID_MODULE	-5	Invalid module.
ACTION_TIMEOUT	-6	Action timeout.

## **Requirements**

Name	Items
Header	mx5Gpower.h
Library	mx5Gpower.lib
DLL	mx5Gpower.dll
Profile	Mx5GpowerProfile [ <i>ModelName</i> ].json

# mxwdg

The **mxwdg** library operates on the relay output and consists of the following:

- **mxwdg\_open**
- **mxwdg\_refresh**
- **mxwdg\_close**

## mxwdg\_open

### **Syntax**

```
PVOID mxwdg_open(unsigned long time);
```

### **Description**

Initializes the watchdog timer.

### **Parameters**

*time*: The interval at which the watchdog timer is refreshed; the unit is seconds.

### **Return Value**

Returns the pointer to the watchdog handle; returns -1 on failure to initialize the watchdog timer.

### **Requirements**

Name	Items
Header	mxwdg.h
Library	mxwdg.lib
DLL	mxwdg.dll

## mxwdg\_refresh

### **Syntax**

```
int mxwdg_refresh(PVOID fd);
```

### **Description**

Refreshes the watchdog timer.

### **Parameters**

*fd*: The handle of the watchdog timer.

### **Return Value**

Returns 0 on success; otherwise, the function has failed.

### **Requirements**

Name	Items
Header	mxwdg.h
Library	mxwdg.lib
DLL	mxwdg.dll

# mxwdg\_close

## **Syntax**

```
void mxwdg_close (PVOID fd) ;
```

## **Description**

Disables the watchdog timer.

## **Parameters**

*fd*: The handle of the watchdog timer.

## **Return Value**

This function does not return a value.

## **Requirements**

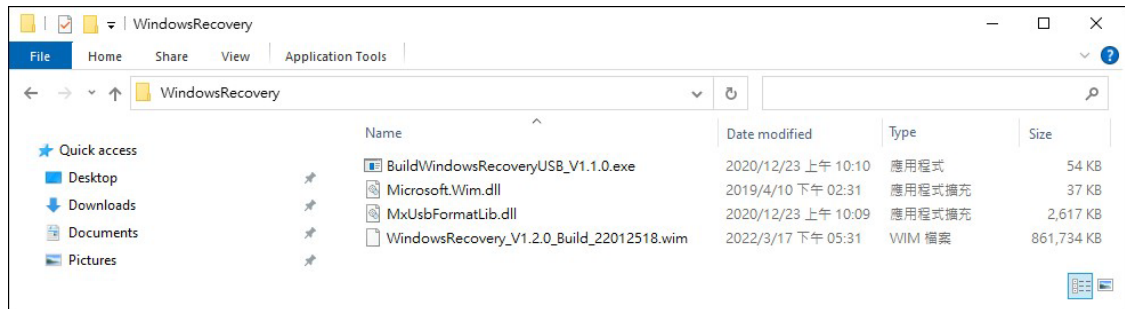
Name	Items
Header	mxwdg.h
Library	mxwdg.lib
DLL	mxwdg.dll

# 10. Windows Recovery

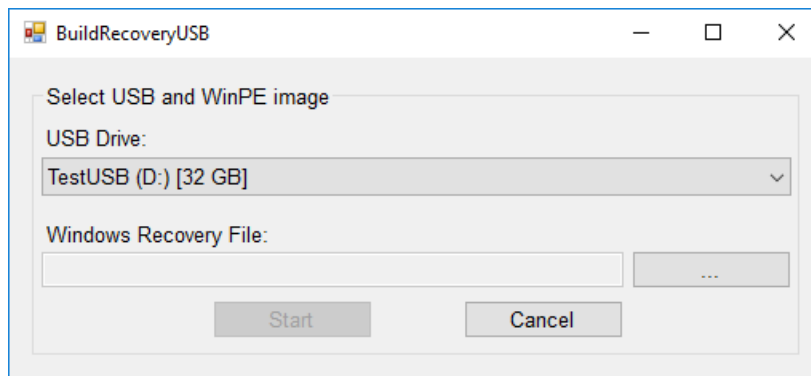
This chapter describes the setup process of the Windows Recovery function.

## Preparing the USB device

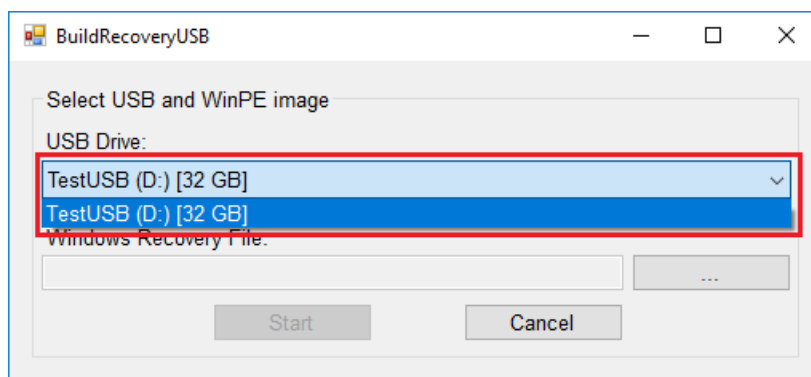
1. Contact a Moxa technical staff and get the required file.



2. Run the **BuildWindowsRecoveryUSB\_V1.1.0.exe**.

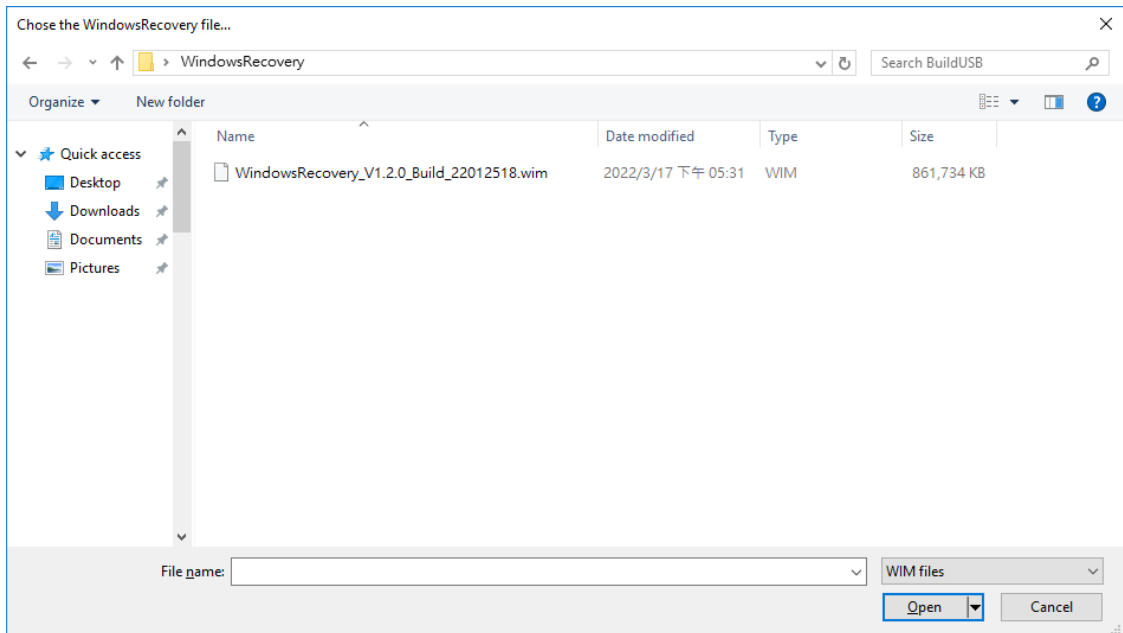
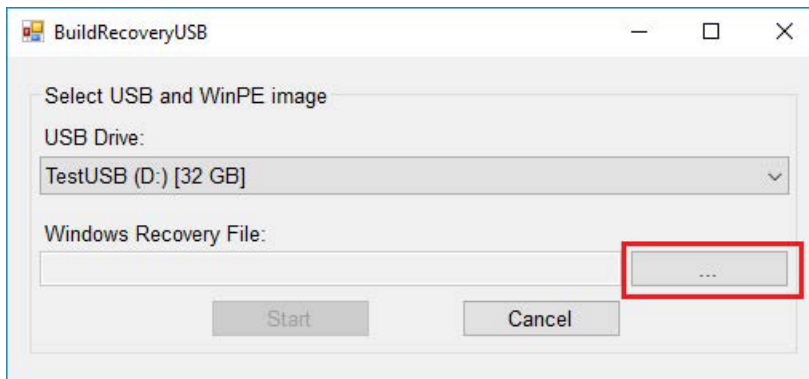


3. Select the USB drive to format.

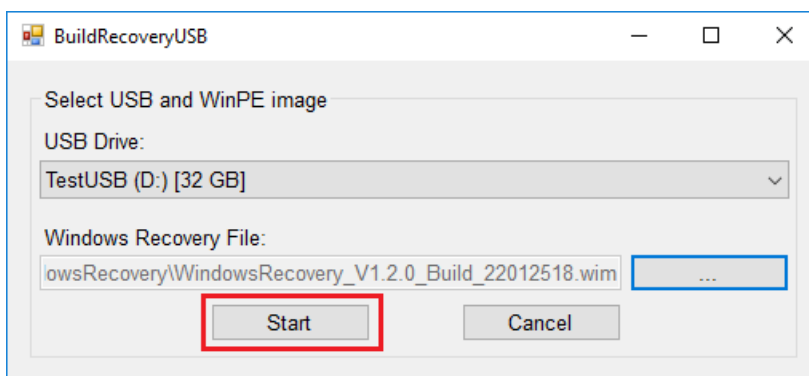




4. Click ... to select .wim file from the folder.

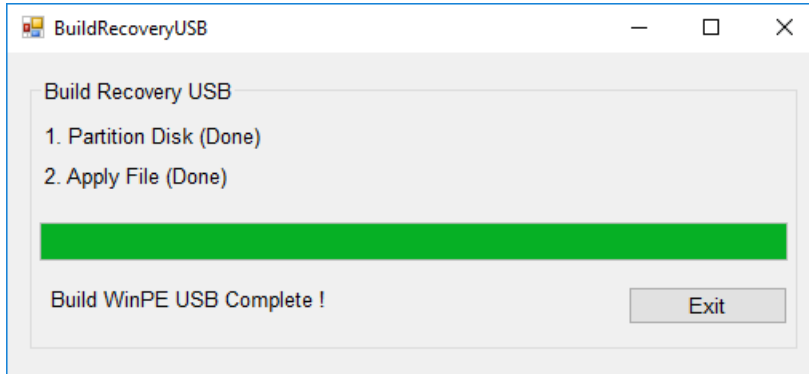


5. Click **Start** and make sure the selected USB can be formatted. Click **Yes** to start creating the recovery USB.

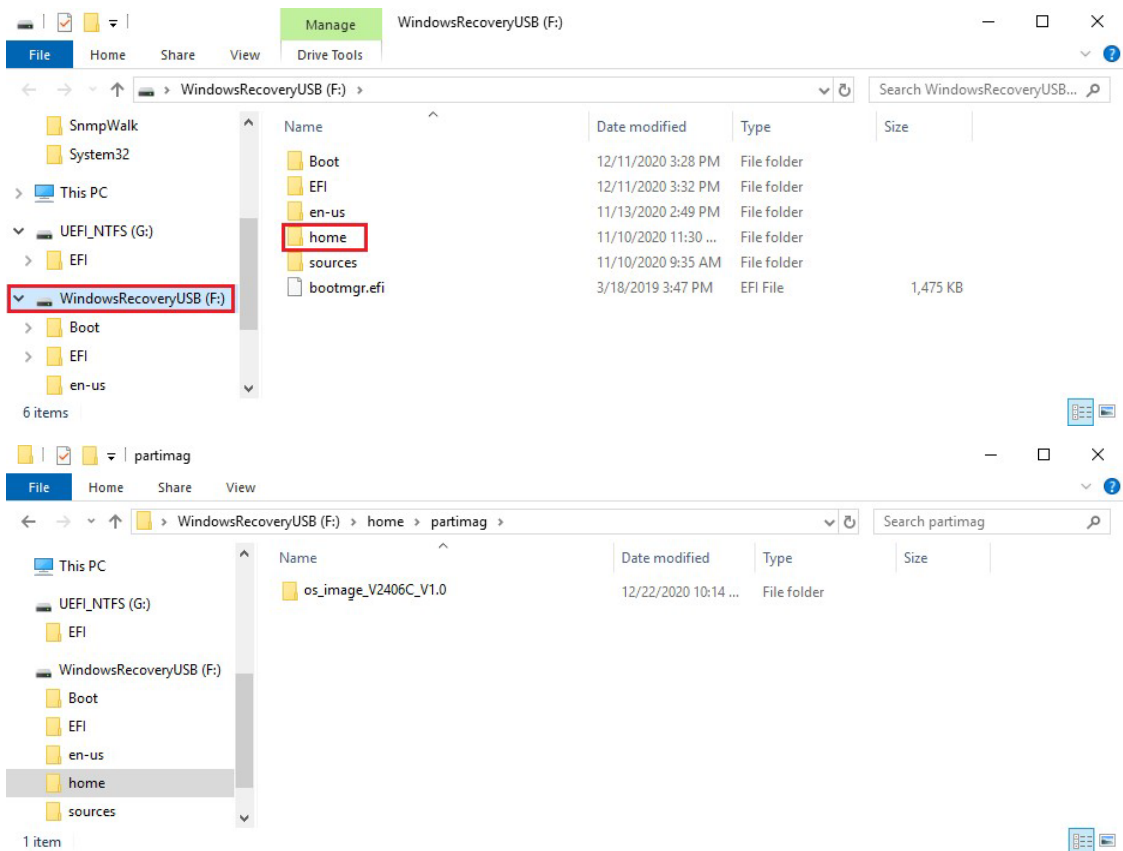


- Wait for the process finished, the program will format the USB device and create a UEFI bootable volume and a WinPE volume.

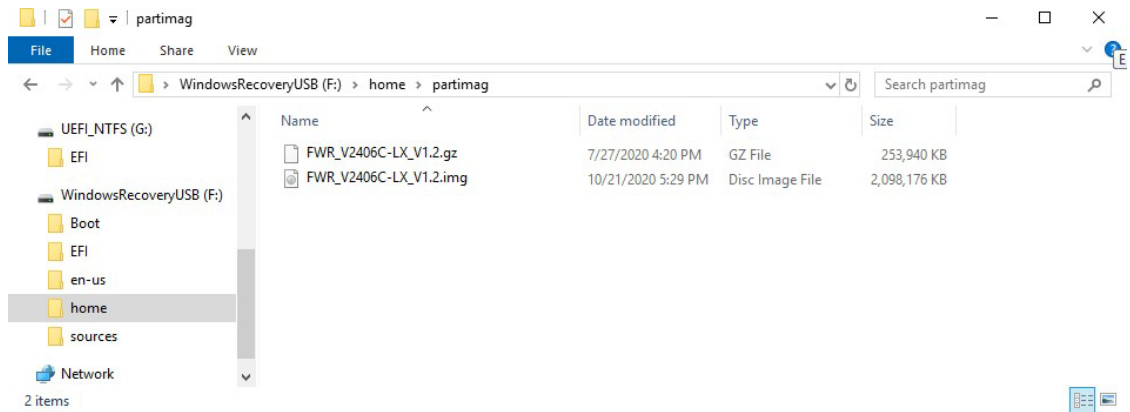
You may see some additional windows about folder information; do not close these. You can close the windows after the process finishes.



- To create a recovery USB disk with the Windows 10 image, copy the **os\_image\_ModelName** directory to the **\home\partimag** folder in the USB drive.

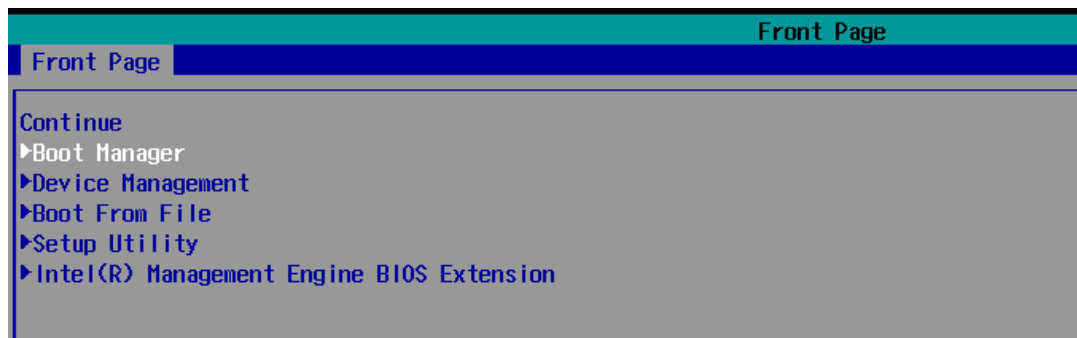


- To create a recovery USB disk with a Linux image, copy the **ModelName.gz** or **ModelName.img** image file to the **\home\partimag** folder in the USB drive.

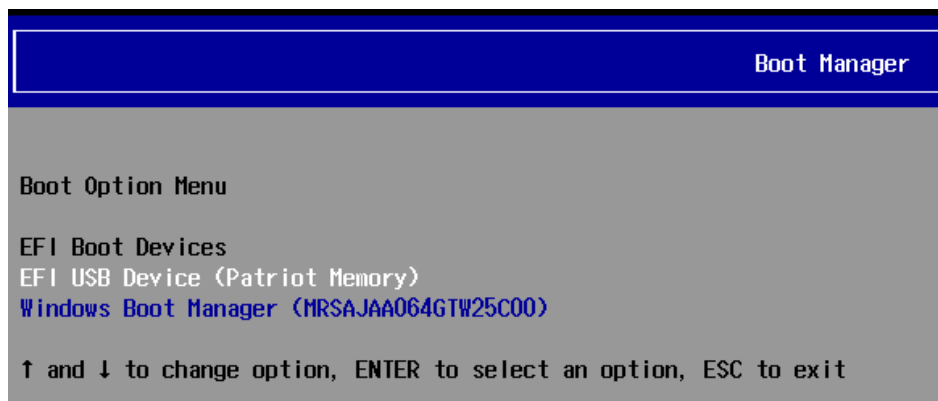


## Booting From the USB Disk

- Turn on the computer and press **F2** when you hear the beep sound to enter the BIOS setup menu.
- Select **Boot Manager** and press **Enter** to continue.



- Select the **EFI USB Device** on the computer and press **Enter** to continue to boot from the USB device.



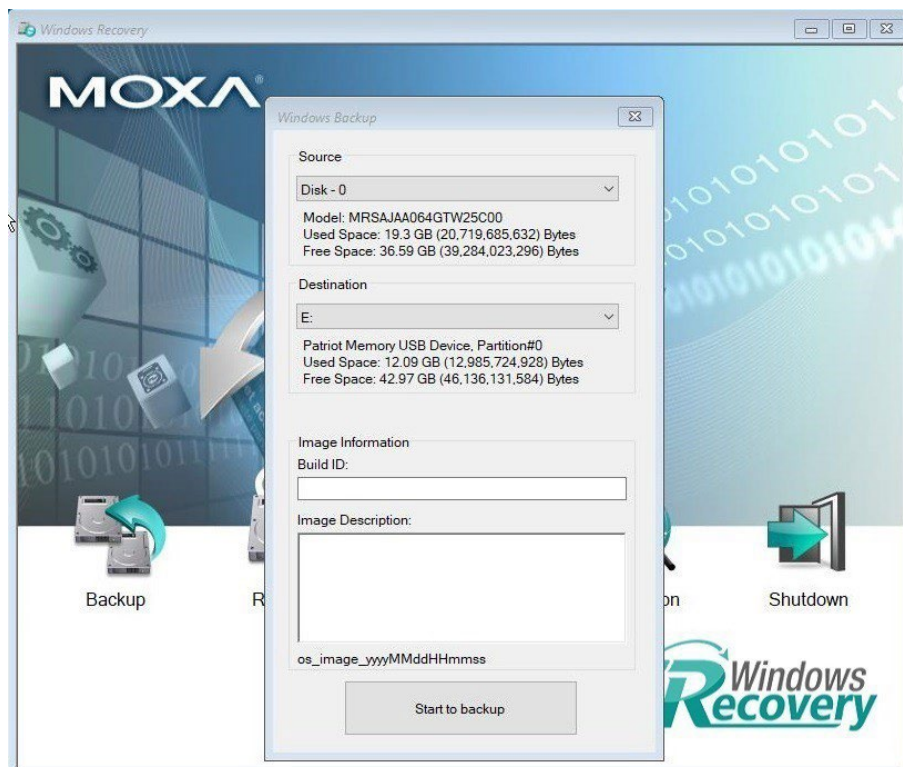
# System Image Backup

To back up the image from the USB disk, run **Windows Preinstallation Environment(WinPE)** and the **Windows Recovery utility** will display. Follow these steps.

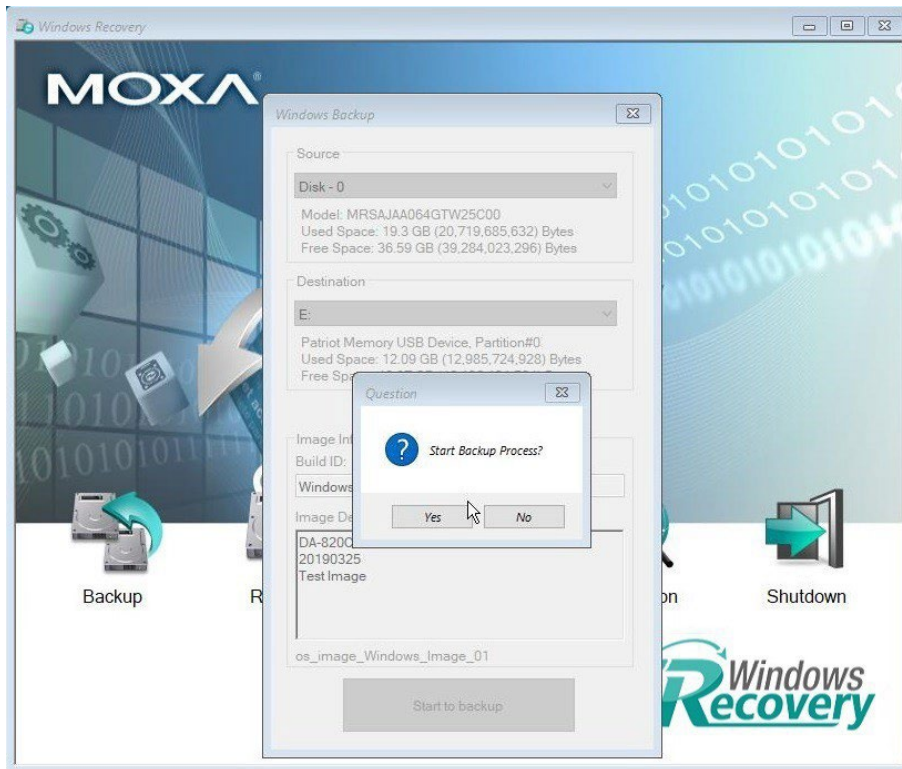
1. Click **Backup**.



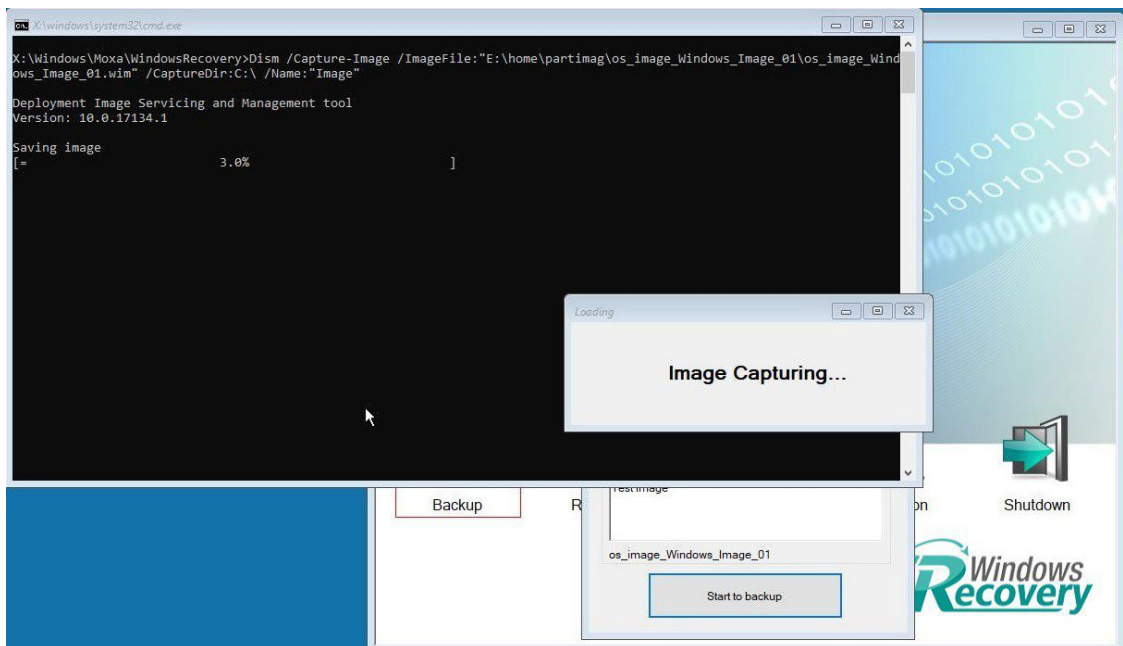
2. Select the **Source disk** to backup and **Destination USB** to store the OS image, also give an image name and description. Click **Start to backup**.



3. Click **Yes** to continue.



4. Wait for the backup process to complete.



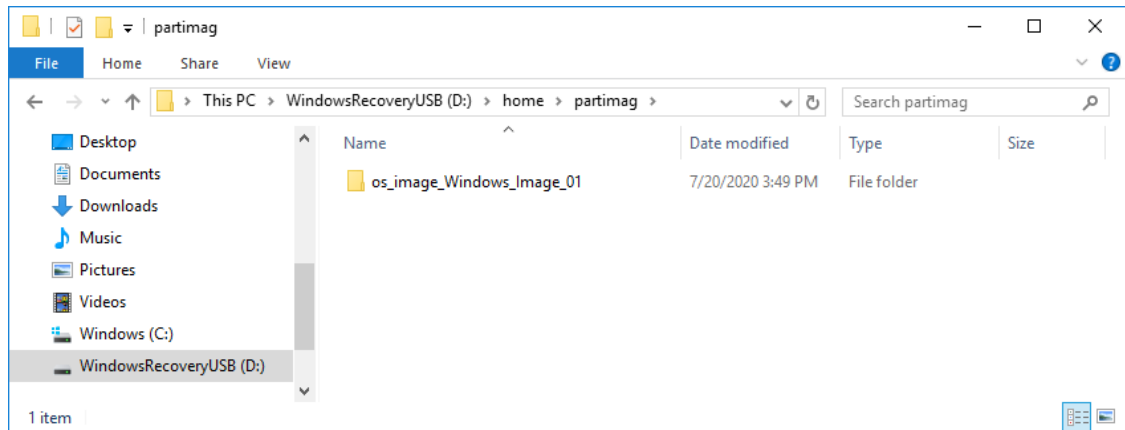
5. When the process is completed, click **OK**.



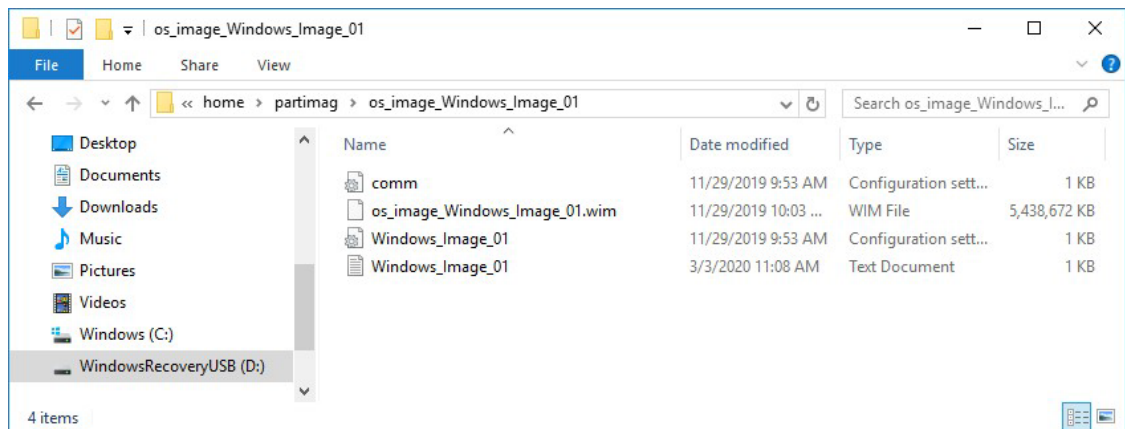
6. Click **OK**, the computer will shut down.



7. The OS image will be saved to the USB disk at **home\partimag**.



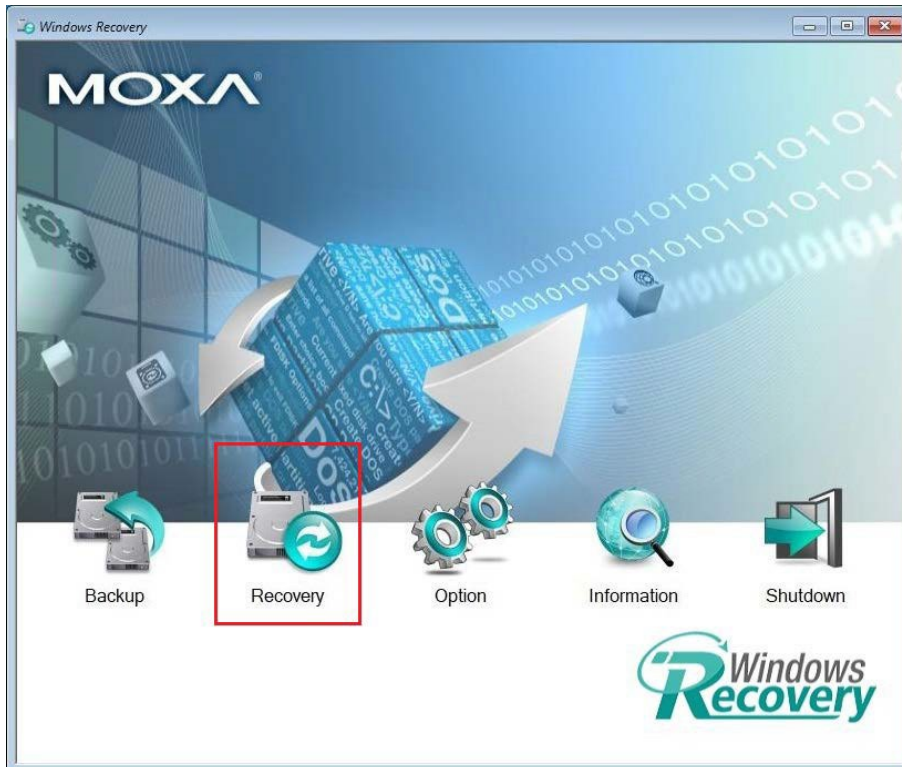
8. In the **os\_image** folder you can view the backup information and the image files.



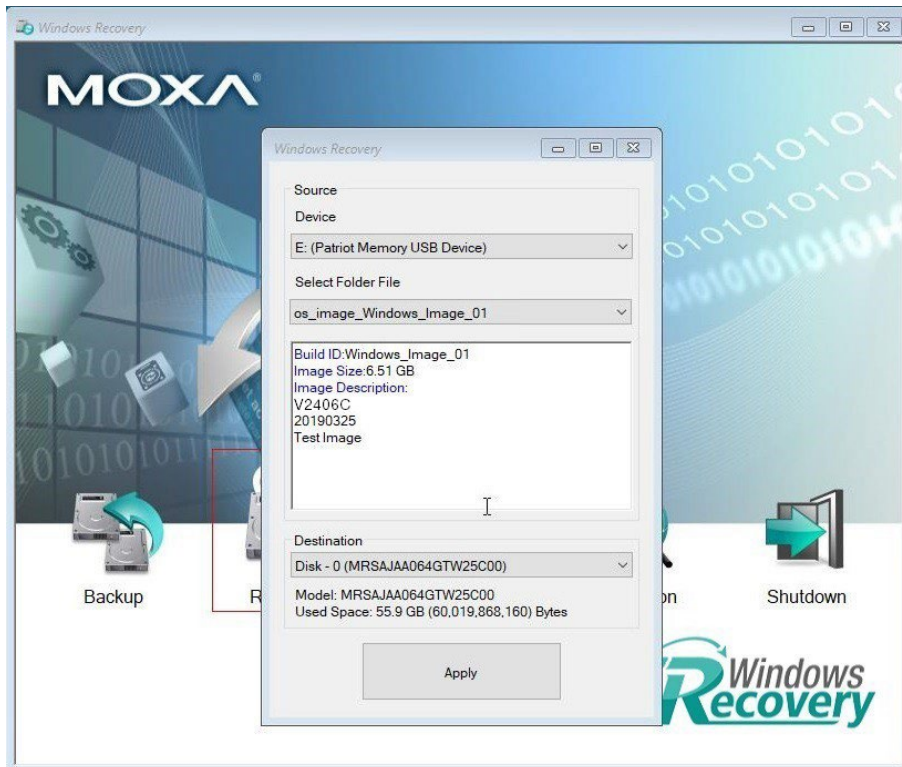
# Restoring the System From a Backup

To restore the image, run the **Windows Preinstallation Environment(WinPE)** and the **Windows Recovery utility** will display. Follow these steps.

1. Click **Recovery**.

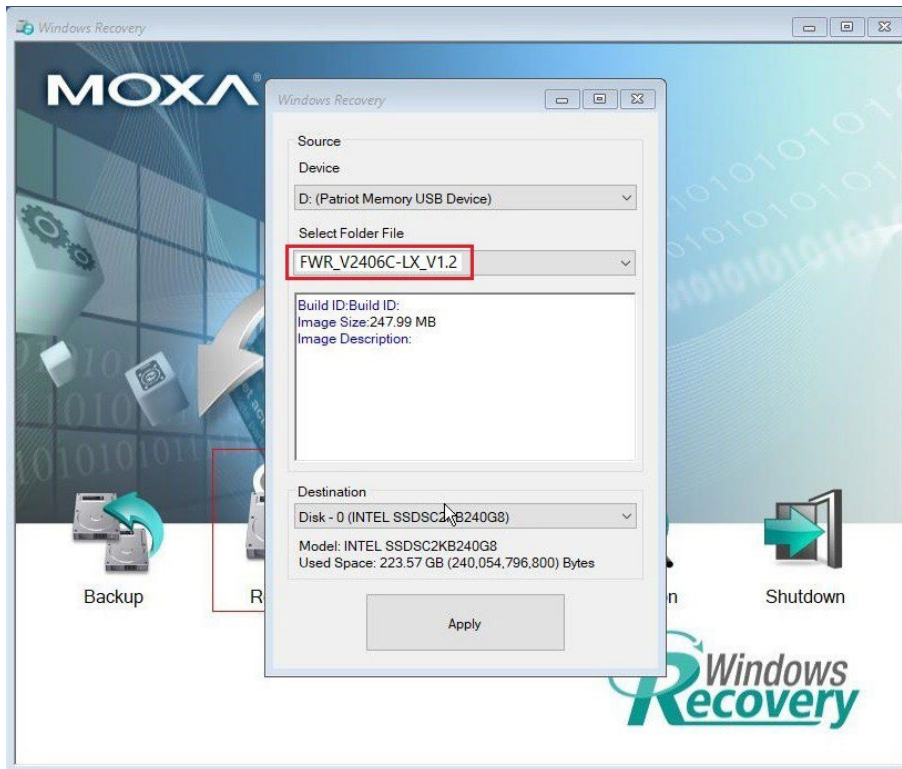


2. Select the **Source USB Device, Image Folder File** and check the image information, select the **Destination Disk** to restore. Click **Apply**.





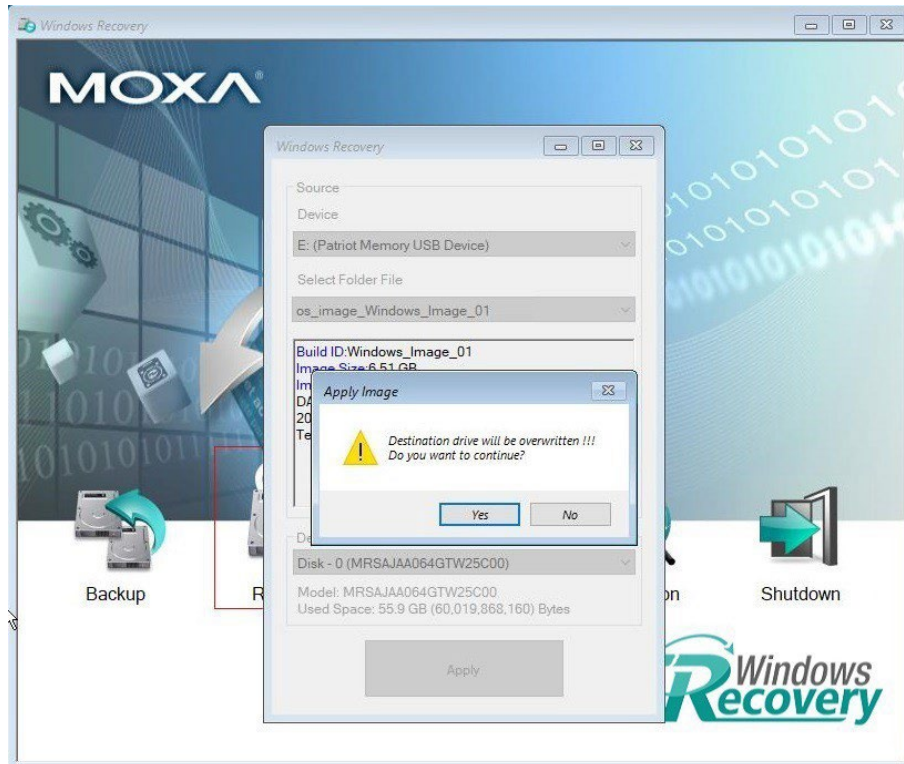
3. If you are using Linux image, select Linux image to restore.



4. Click **Yes** to continue the process.



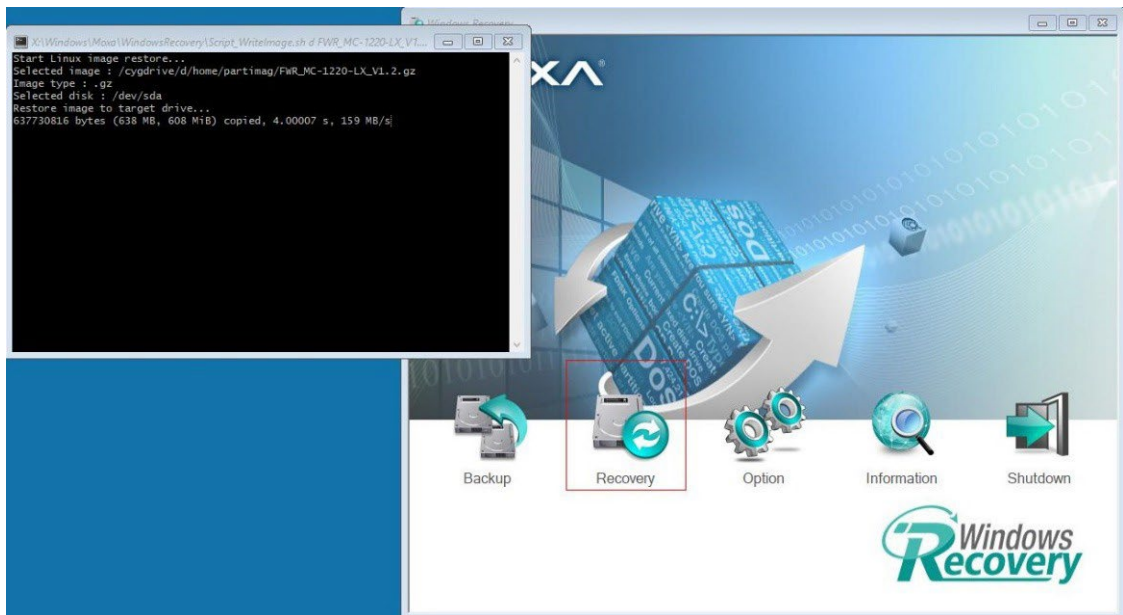
5. Click **Yes** to overwrite the destination drive.



6. Wait for the process to complete.



7. If you restore the Linux image, a new window will appear. Wait for the process to complete.



8. Click **OK**.



9. Click **OK**, the system will shut down and restart.



## NOTE

When you restart the computer, you will need to wait about 5 minutes for the computer to go through two cycles of the reboot process. The system configuration files will be initiated during the first boot-up process. Do not turn off or shut down the computer while the system is restarting.