XVGU Communication Driver

Driver for USB Communication with
EZ Tower Light devices

Contents

Introduction

The XVGU driver enables communication between BLUE Open Studio (or BOS, for short) system and EZ Tower Light using USB cable connection, according to the specifications discussed in this publication.

This publication was designed to help you install, configure, and execute the XVGU driver to enable communication with EZ Tower Light devices. The information in this publication is organized as follows:

- **Introduction**: Provides an overview of the XVGU driver documentation.
- **General Information**: Provides information needed to identify all the required components (hardware and software) used to implement communication between BOS and the XVGU driver.
- **Installing the Driver**: Explains how to install the XVGU driver.
- **Configuring the Driver**: Explains how to configure the communication driver.
- **Executing the Driver**: Explains how to execute the driver to verify that you have installed and configured the driver correctly.
- **Troubleshooting**: Lists the most common error codes for this protocol.
- **Sample Application**: Explains how to use a sample application to test the driver configuration.
- **Revision History**: Provides a log of all modifications made to the driver and the documentation.

Notes:

- This document assumes that you have read the "Development Environment" chapter in the product's *Technical Reference Manual*.
- This document also assumes that you are familiar with the Windows environment.
  If you are unfamiliar with Windows, we suggest using the **Help** feature (available from the Windows desktop **Start** menu) as you work through this guide.

General Information

This chapter explains how to identify all the hardware and software components used to implement communication between the XVGU driver and EZ Tower Light.

The information is organized into the following sections:

- Device Characteristics
- Link Characteristics
- Driver Characteristics

Device Characteristics

This driver has been tested successfully with the following devices:

- **Manufacturer**: Schneider Electric
- **Compatible Equipment**: EZ Tower Light devices PFXZCETWW*, PFXZCETWHA*, XVGU3SHA* and XVGU3SW*.
- **Programmer Software**: Not required.
- **OS Device Driver :** Install the device driver (MCHPWinUSBDevice_v2.inf) – see Appendix A

Link Characteristics

To establish communication, you must use links with the following specifications:

- **Device Communication Port**: USB Port

Driver Characteristics

The XVGU driver is composed of the following files:

- **`XVGU.INI`**: Internal driver file. *You must not modify this file*.
- **`XVGU.MSG`**: Internal driver file containing error messages for each error code. *You must not modify this file*.
- **`XVGU.PDF`**: Document providing detailed information about the XVGU driver.
- **`XVGU.DLL`**: Compiled driver.

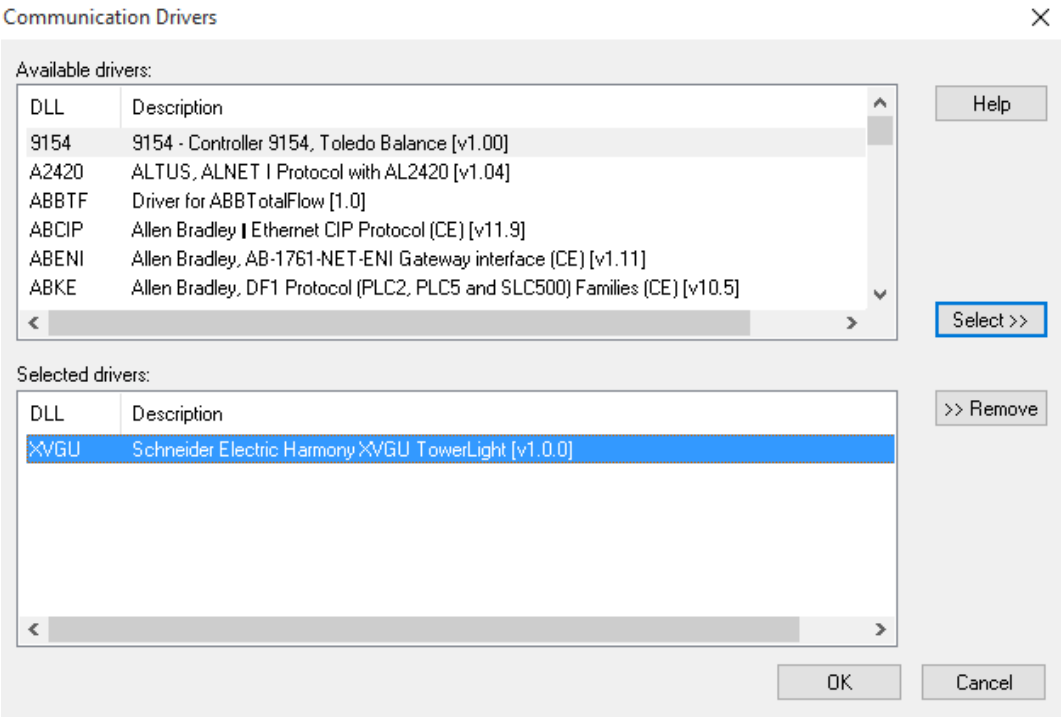
> **Notes:**
>
> The XVGU driver requires the **`LibXVGUTowerLight.DLL`**, which is automatically installed along with the driver at the **`/DRV/API`** sub-folder from BOS.
>
> Moreover, in addition to the aforementioned driver files, you must install the operating system (OS) device driver (MCHPWinUSBDevice_v2.inf) – see Appendix A for more information.

Installing the Driver

When you install BOS, all of the communication drivers are installed automatically.
You must select the driver that is appropriate for the application you are using.

Perform the following steps to select the driver from within the application:

1. Open BOS from the **Start** menu.
2. From the BOS main menu bar, select **File** →**Open Project** to open your application.
3. Select **Insert** →**Driver** from the main menu bar to open the *Communication Drivers* dialog.
4. Select the **XVGU** driver from the *Available Drivers* list, and then click the **Select** button:



Communication Drivers Dialog

5. When the **XVGU** driver displays in the *Selected Drivers* list, click the **OK** button to close the dialog.

> ➲ **Caution:**
> For safety reasons, you must use caution when installing the physical hardware. Consult the hardware manufacturer's documentation for specific installation instructions.

Configuring the Driver

After opening BOS and selecting the XVGU driver, you must configure the driver. The XVGU driver's configuration is done in a part:

- Defining communication tags and controls in the Communication tables or *Driver* worksheet

  Worksheets are divided into two sections, a *Header* and a *Body*. The fields contained in these two sections are standard for all communications drivers — except the **Station**, **Header** and **Address** fields, which are driver-specific. This document explains how to configure the **Station**, **Header** and **Address** fields only.

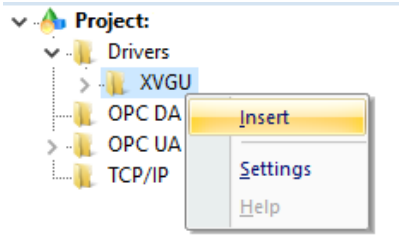
> **Notes:**
> For a detailed description of the BOS *Standard* and *MAIN* Driver Worksheets, and information about configuring the standard fields, review the product's *Technical Reference Manual*.

Configuring the Driver Worksheet

This section explains how to configure a *Standard Driver Worksheet* (or Communication table) to associate application tags with the TowerLight addresses. You can configure multiple *Driver* worksheets — each of which is divided into a *Header* section and a *Body* section.

Use the following steps to create a new *Standard Driver* worksheet:

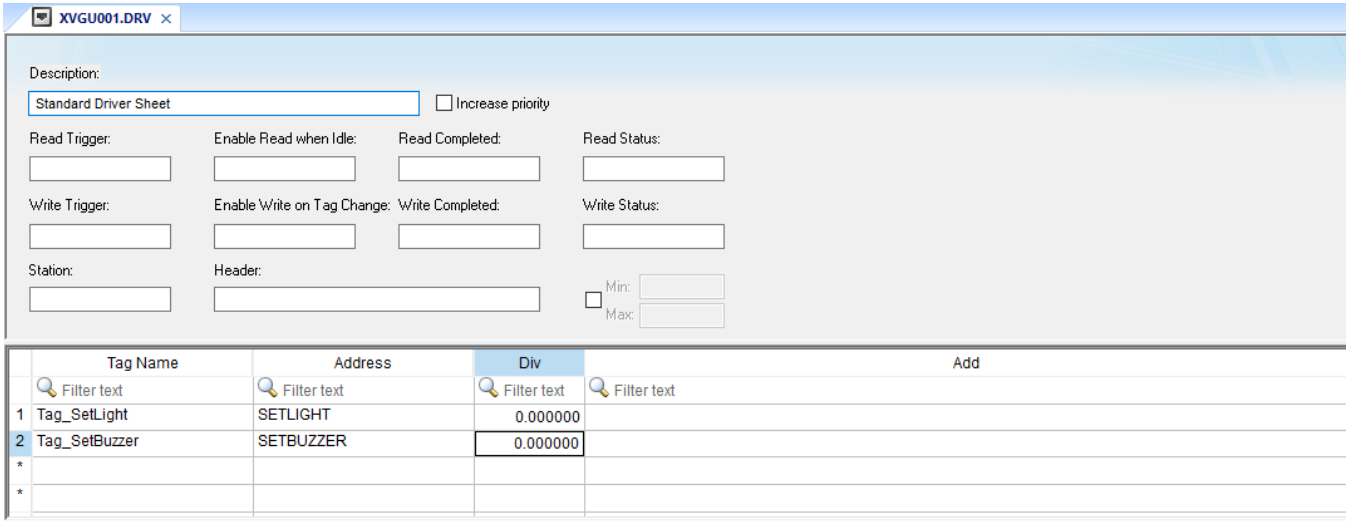
1. From the BOS development environment, select the **Comm** tab, located below the *Workspace* pane.
2. In the *Workspace* pane, expand the *Drivers* folder, and right-click the *XVGU* subfolder.
3. When the pop-up menu displays, select the **Insert** option:



Inserting a New Worksheet

> **Note:**
> To optimize communication and ensure better system performance, you must tie the tags in different driver worksheets to the events that trigger communication between each tag group and the period in which each tag group must be read or written. Also, we recommend configuring the communication addresses in sequential blocks to improve performance.

The *XVGU.drv* dialog displays (similar to the following figure):



XVGU Driver Worksheet

In general, all parameters on the *Driver* worksheet (except the **Station**, **Header** and **Address** fields) are standard for all communication drivers, and they will not be discussed in this publication. For detailed information about configuring the standard parameters, consult the BOS *Technical Reference Manual*.

4. Use the following information to complete the **Station**, **Header**, and **Address** fields on this worksheet:

Station: this field is not used for XVGU driver. It must be in blank.

Header: this field is not used for XVGU driver. It must be in blank.

Address field: Use the information provided in the following table to associate each tag to its respective function that will change a Tower Light configuration.

Type the tag from your application database into the **Tag Name** column. This tag will set the device's behavior through the address. The address must comply with the following syntax:

`<Function>` (for example: **`SETLIGHT`**)

Where:

- **`<Function>:`** defines the command that will be set on the device.

The following table show all the address possibilities.

Sample Address Configuration		
Header Field	**Address Field**	**Description**
	SETLIGHT	Set a specified layer's color and mode.
	SETLIGHTTOP	Set the top layer's color and mode.
	SETLIGHTMIDDLE	Set the middle layer's color and mode.
	SETLIGHTBOTTOM	Set the bottom layer's color and mode.
	SETBUZZER	Set a buzzer with a specified tone, volume and pattern.
	CONFIGPATTERN	Write customized patterns into the ROM of the TowerLight device.
	RUNPATTERN	Run a saved target pattern.
	RESETALL	Turn off all lights and buzzer.

Tag value format

When the driver addresses are set, the respective tags value need to have some parameters configured. These will be sent to the TowerLight. In the following, it will be briefly explained how each address works and its parameters.

SETLIGHT

Set the light status of any layer. The tag value format for this address is:

<Layer>:<Color>:<Mode>

Where:

<Layer>: BOTTOM (0), MIDDLE (1) or TOP (2).
<Color>: RED (0), YELLOW (1), GREEN (2), BLUE, (3) or ORANGE (4).
<Mode>: OFF (0), ON (1), BLINK (2) or FLASH (3).

SETLIGHTBOTTOM, SETLIGHTMIDDLE and SETLIGHTTOP.

If the address chosen is equal to SETLIGHTBOTTOM, SETLIGHTMIDDLE or SETLIGHTTOP, we have these configuration on the tag value:

<Color>:<Mode>

Where:

<Color>: RED (0), YELLOW (1), GREEN (2), BLUE, (3) or ORANGE (4).
<Mode>: OFF (0), ON (1), BLINK (2) or FLASH (3).

The meaning of these addresses is that the layer is already set in the command call.

SETBUZZER

Set the buzzer sound. The tag value format for this address is:

<Tone>:<Volume>:<BuzzerPattern>

Where:

<Tone>: HIGH (0) or LOW (1).
<Volume>: MAXIMUM (0), MIDDLE (1) or MINIMUM (2).
<BuzzerPattern>: OFF (0), PATTERN1 (1), PATTERN2 (2), PATTERN3 (3) or PATTERN4 (4).

CONFIGPATTERN

Defines a pattern to operate TowerLight that can be saved in the device. The tag value format for this address is:

<PatternNumber>:<BottomColor>:<BottomMode>: <MiddleColor>:<MiddleMode>:<TopColor>:<TopMode>:
<Tone>:<Volume>:<BuzzerPattern>

Where:

<PatternNumber>: 0 – 7.
<BottomColor >: RED (0), YELLOW (1), GREEN (2), BLUE, (3) or ORANGE (4).
<BottomMode >: OFF (0), ON (1), BLINK (2) or FLASH (3).
<MiddleColor >: RED (0), YELLOW (1), GREEN (2), BLUE, (3) or ORANGE (4).
<MiddleMode >: OFF (0), ON (1), BLINK (2) or FLASH (3).
<TopColor >: RED (0), YELLOW (1), GREEN (2), BLUE, (3) or ORANGE (4).
<TopMode >: OFF (0), ON (1), BLINK (2) or FLASH (3).
<Tone>: HIGH (0) or LOW (1).
<Volume>: MAXIMUM (0), MIDDLE (1) or MINIMUM (2).
<BuzzerPattern>: OFF (0), PATTERN1 (1), PATTERN2 (2), PATTERN3 (3) or PATTERN4 (4).

RUNPATTERN

Run a saved pattern to operate TowerLight. The tag value format for this address is:

<PatternNumber>

Where:

<PatternNumber>: 0 – 7.

RESETALL

Turn off lights and buzzer when value changes between 0 and 1.

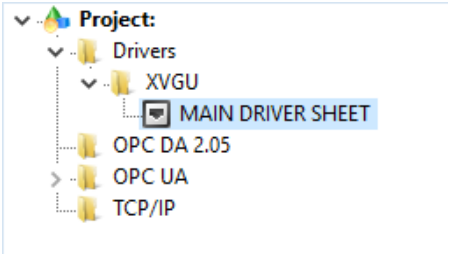
> **Note:**
> This driver only supports Write actions and does not support Read actions. Please configure the *Action* field on the *Main Driver Sheet* to be *Write* for all the items. The fields *Read Trigger*, *Enable Read when Idle, Read Completed*, *Read Status* must be left blank on the *Standard Driver Worksheets*.

Examples of Tag Values in Standard Driver Sheet

Address Field	Tag Values
SETLIGHT	TOP:RED:ON
	MIDDLE:ORANGE:FLASH
SETLIGHTTOP	RED:ON
	1:2
	GREEN:0
SETLIGHTMIDDLE	ORANGE:FLASH
SETLIGHTBOTTOM	BLUE:OFF
	YELLOW:BLINK
SETBUZZER	LOW:MINIMUM:PATTERN1
	HIGH:MAXIMUM:OFF
CONFIGPATTERN	1:RED:ON:GREEN:ON:ORANGE:FLASH:LOW:MINIMUM:PATTERN1
	1:0:0:0:0:0:0:0:0:0
RUNPATTERN	0,1,2,3,4,5,6,7
RESETALL	Tag change between (for example 0 or 1

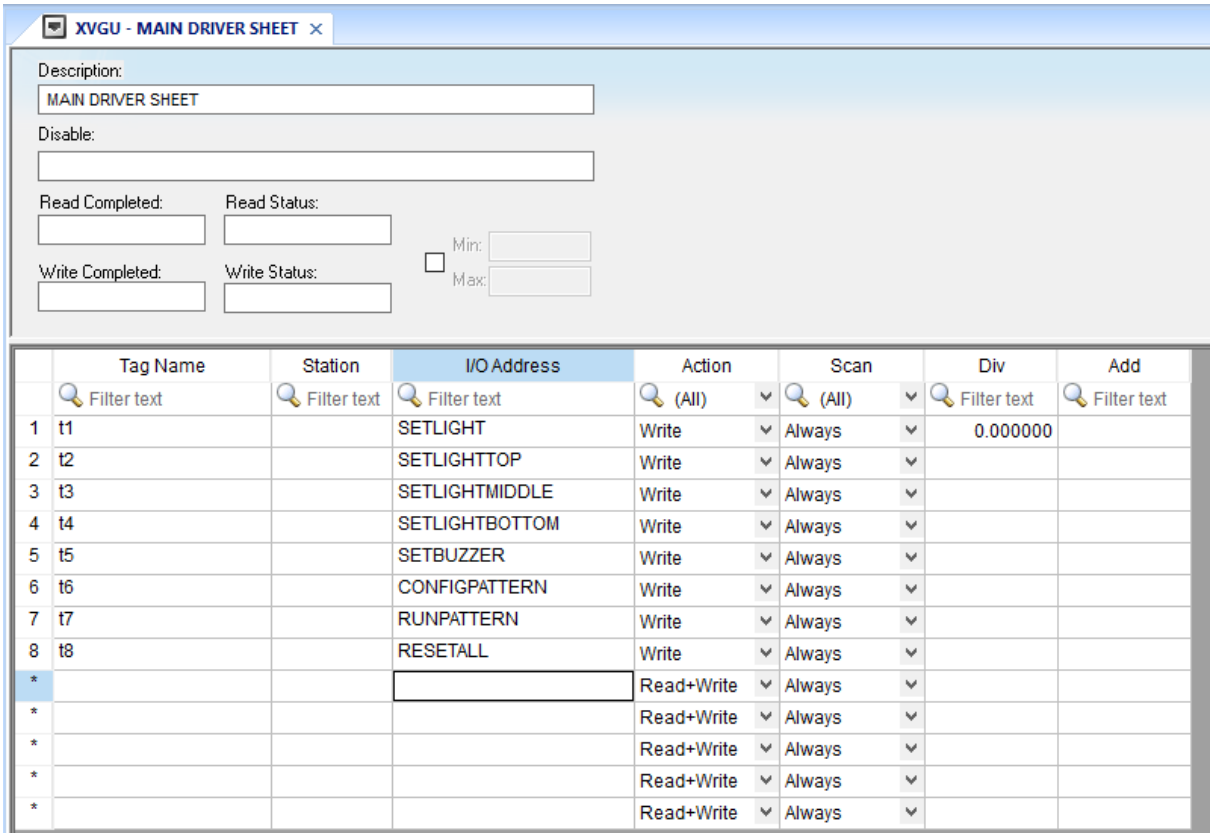
Main Driver Sheet (MDS)

When the driver is inserted into the application, the *MAIN DRIVER SHEET* is automatically added to the driver folder.



Main Driver Sheet

The MAIN DRIVER SHEET provides a simple way to associate BOS tags to addresses in the TowerLight. Most of the MAIN DRIVER SHEET entries are standard for any driver. Refer to the BOS *Technical Reference Manual* about the configuration of the standard fields. The fields that require specific syntax for this driver are described below:



	Tag Name	Station	I/O Address	Action	Scan	Div	Add
	Filter text	Filter text	Filter text	(All)	(All)	Filter text	Filter text
1	t1		SETLIGHT	Write	Always	0.000000	
2	t2		SETLIGHTTOP	Write	Always		
3	t3		SETLIGHTMIDDLE	Write	Always		
4	t4		SETLIGHTBOTTOM	Write	Always		
5	t5		SETBUZZER	Write	Always		
6	t6		CONFIGPATTERN	Write	Always		
7	t7		RUNPATTERN	Write	Always		
8	t8		RESETALL	Write	Always		
*				Read+Write	Always		
*				Read+Write	Always		
*				Read+Write	Always		
*				Read+Write	Always		
*				Read+Write	Always		

Main Driver Sheet

- **Station:** this field is not used for XVGU driver. It must be in blank.
- **I/O Address:** Please see *Address Field* in the *Standard Driver Sheet* section.

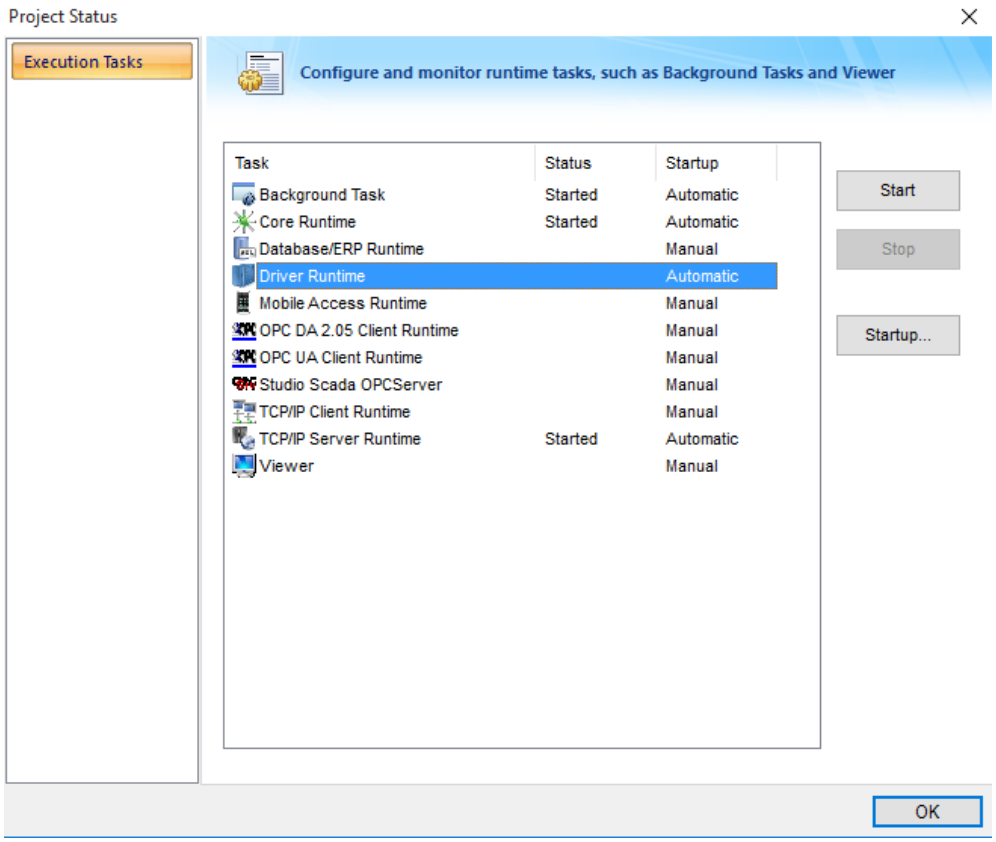
> **Note:**
> This driver only supports Write actions and does not support Read actions. Please configure the *Action* field on the *Main Driver Sheet* to be *Write* for all the items. The fields *Read Trigger*, *Enable Read when Idle, Read Completed*, *Read Status* must be left blank on the *Standard Driver Worksheets*.

Executing the Driver

After adding the XVGU driver to a project, BOS sets the project to execute the driver automatically when you start the run-time environment.

To verify that the driver run-time task is enabled and will start correctly, perform the following steps:

1. Select **Project** → **Status** from the main menu bar.
   The *Project Status* dialog displays:



Project Status Dialog

2. Verify that the *Driver Runtime* task is set to **Automatic**.

If the setting is correct, click **OK** to close the dialog.

If the **Driver Runtime** task is set to **Manual**, select the **Driver Runtime** line. When the **Startup** button becomes active, click the button to toggle the *Startup* mode to **Automatic**.

3. Click **OK** to close the *Project Status* dialog.
4. Start the application to run the driver.

Troubleshooting

If the XVGU driver fails to communicate with the device, the tag you configured for the **Read Status** or **Write Status** fields will receive an error code. Use this error code and the following table to identify what kind of failure occurred.

Error Code	Description
0	TowerLight is working correctly
1	TowerLight library was not found
2	TowerLight library is not valid
3	Function not found in TowerLight
4	Connection to TowerLight failed
5	Address is write-only. Read operation is not allowed
6	Invalid pattern number
7	Invalid layer
8	Invalid mode
9	Invalid color
10	Invalid tone
11	Invalid volume
12	Invalid buzzer pattern
13	Invalid pattern definition
14	Invalid number of parameters

➪ **Tip:**

You can verify communication status using the BOS development environment *Output* window (*LogWin* module). To establish an event log for **Field Read Commands**, **Field Write Commands**, and **Protocol Analyzer**, right-click in the *Output* window. When the pop-up menu displays, select the option to set the log events.

If you must contact us for technical support, please have the following information available:

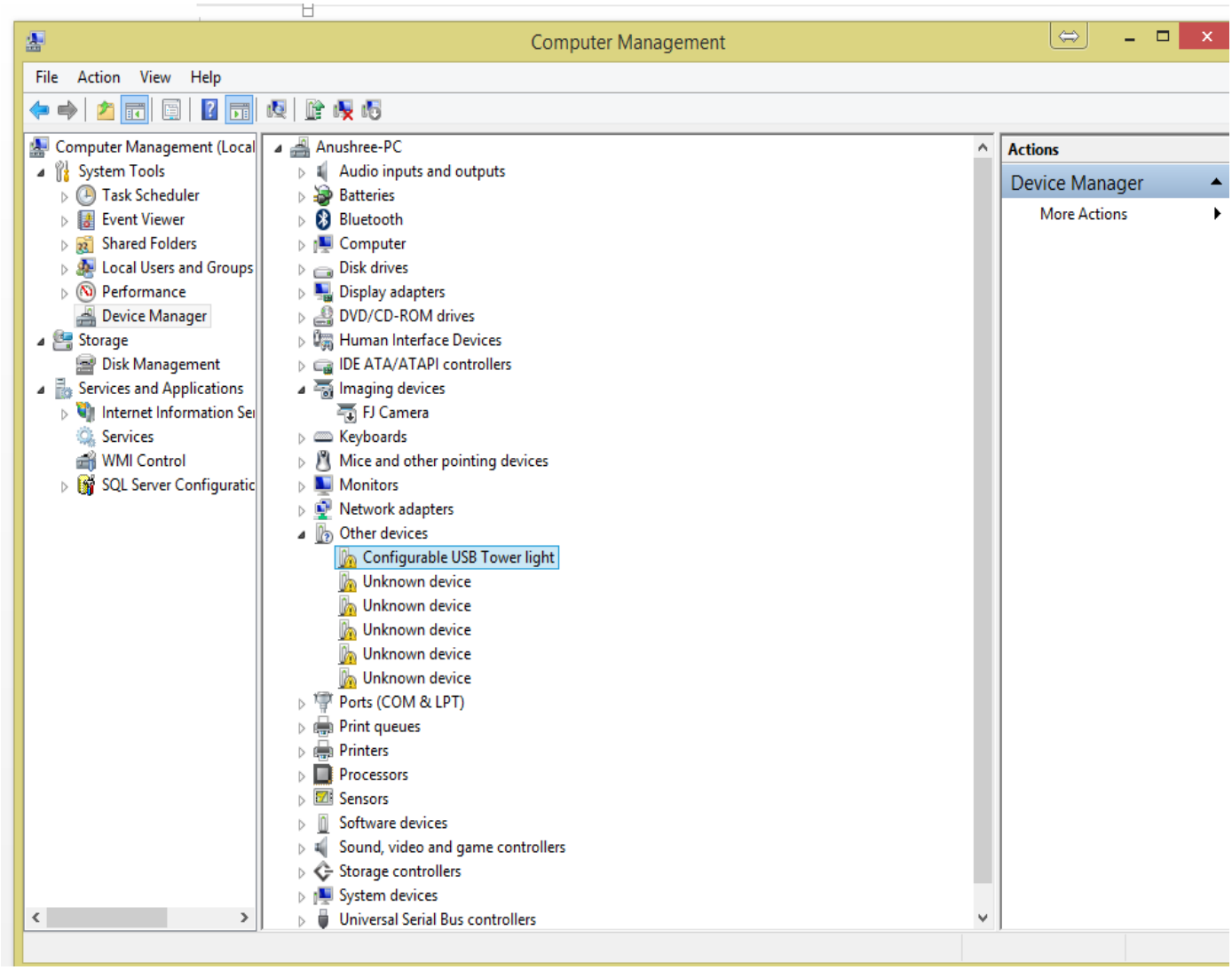
- **Operating system** (type and version): To find this information, select **Tools** → **System Information**.
- **Project Information**: To find this information, select **Help** → **Support Information.**
- **Driver version** and **communication log**: Displays in the BOS *Output* window when the driver is running.
- **Device model** and **boards**: Consult the hardware manufacturer's documentation for this information.

Appendix – A : Installing the Device Driver

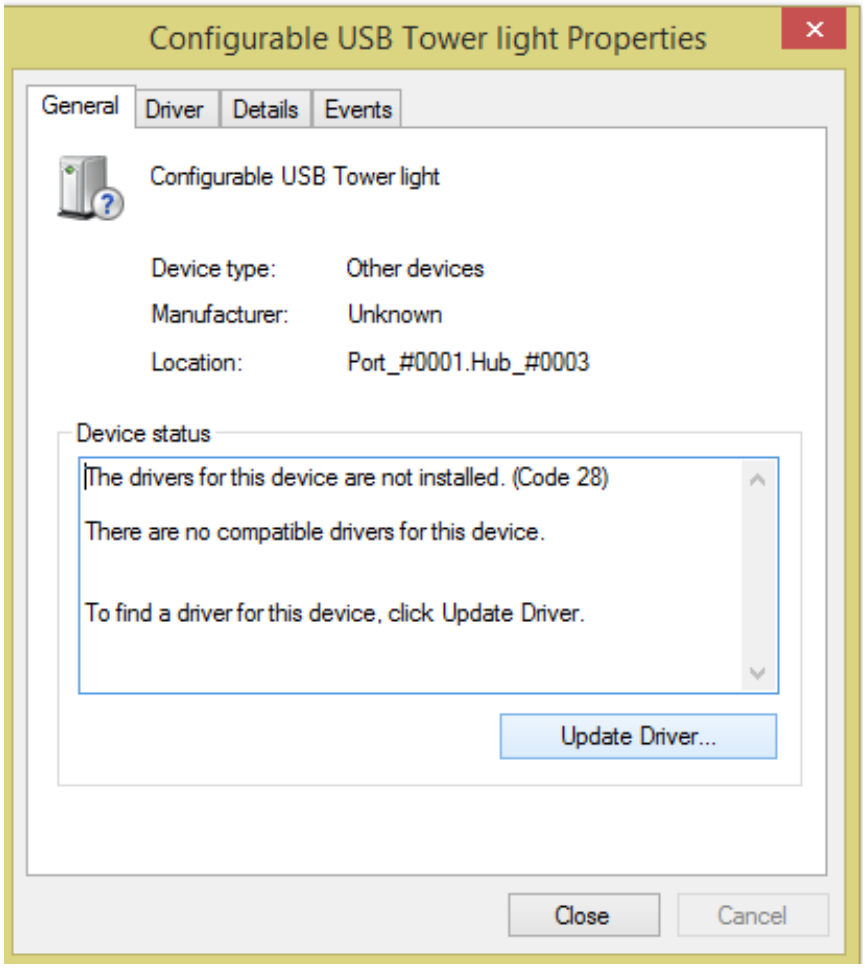
This section describes how to install and configure the device driver (MCHPWinUSBDevice_v2.inf) for the Tower Light device. You can download the device driver from our support site at http://www.pro-face.com/trans/en/manual/1038.html. The drivers are already installed on the SP5000 Series Open Box.

1. Connect the USB cable of the device to PC.

2. If the "found new Hardware Wizard" starts, select "no, not this time", and then click Next.
(The wizard does not appear if older drivers already installed.)
If so, remove the device listed in the Device Manager "the port" field.
By running the "scan for hardware changes" and is recognized as a new device.

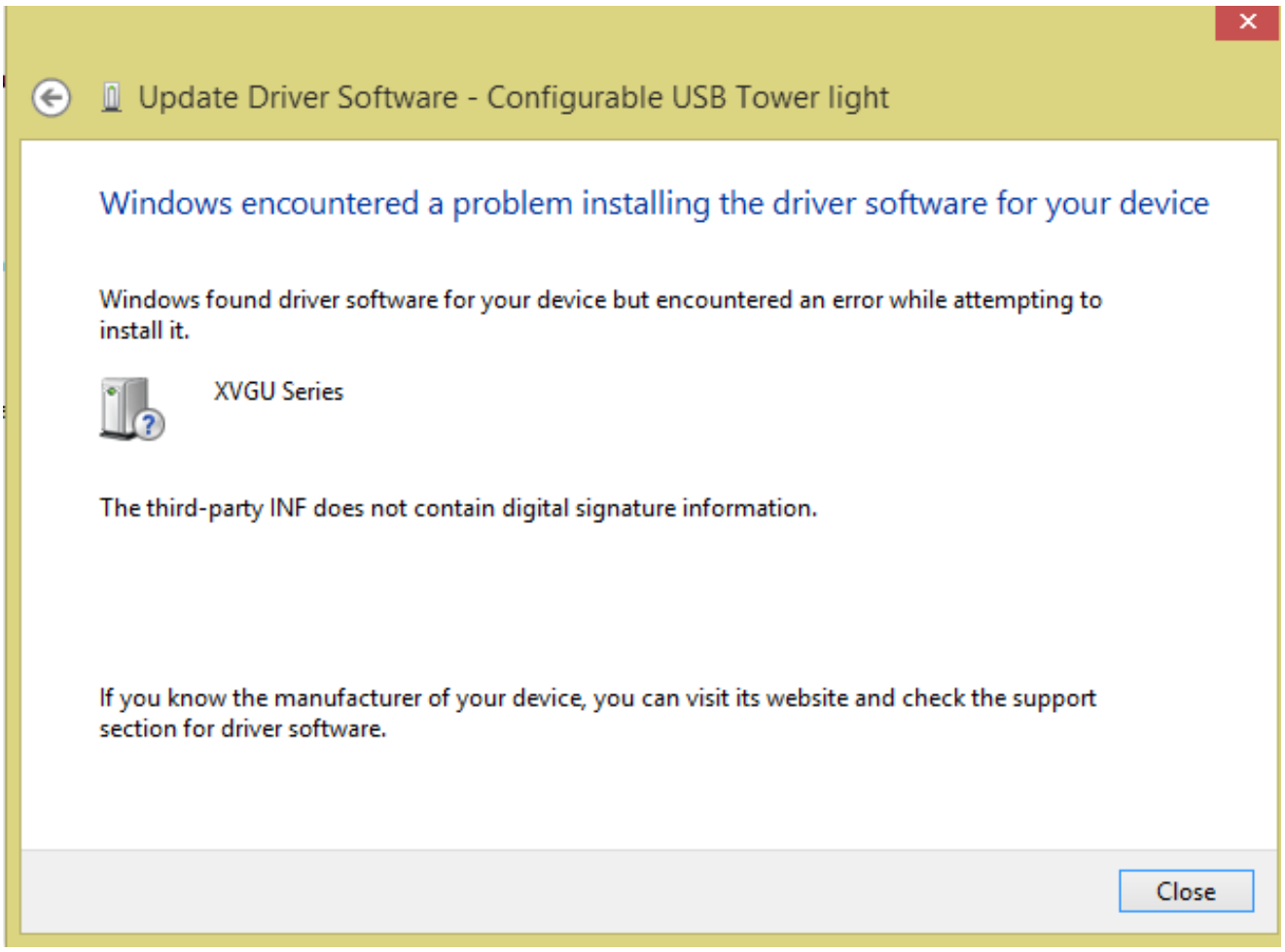


On right clicking the "Configurable USB Tower Light" -> Select Update Driver



3. Select "Browse my computer for driver software" and point to the folder with the driver (MCHPWinUSBDevice_v2.inf) file.

You may see this error message



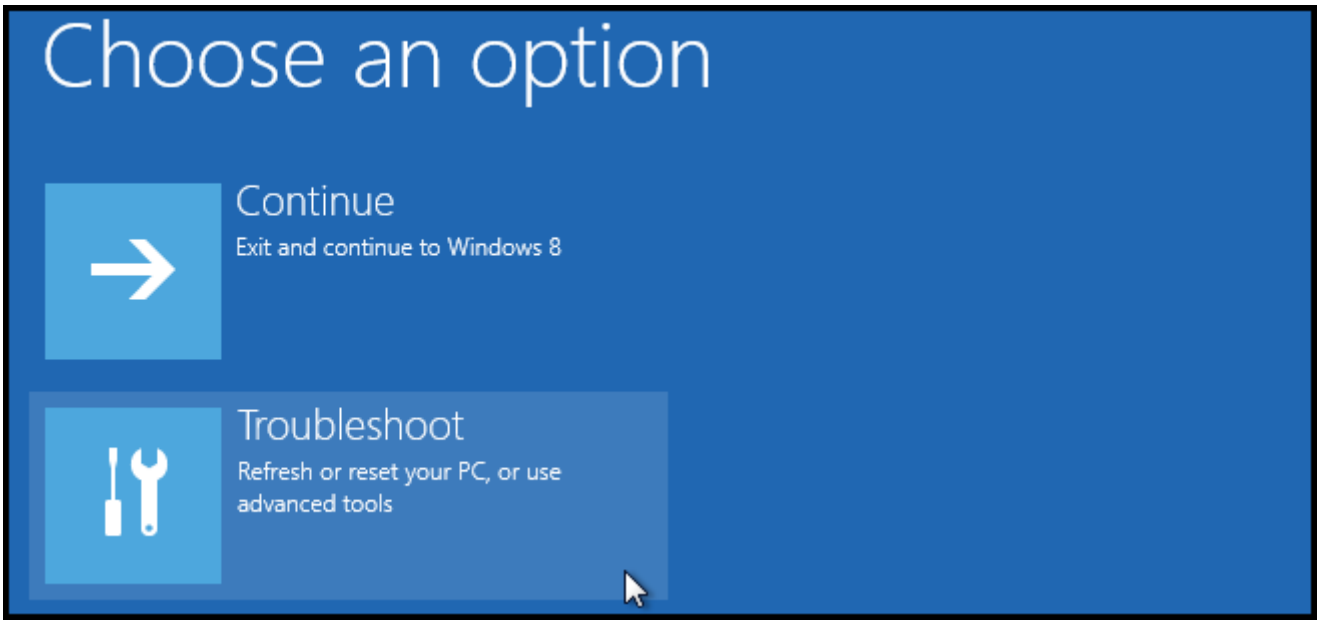
4. To troubleshoot the problem, you must disable driver signature verification by going to the Troubleshooting options from the boot manager.

Follow these sub-steps:

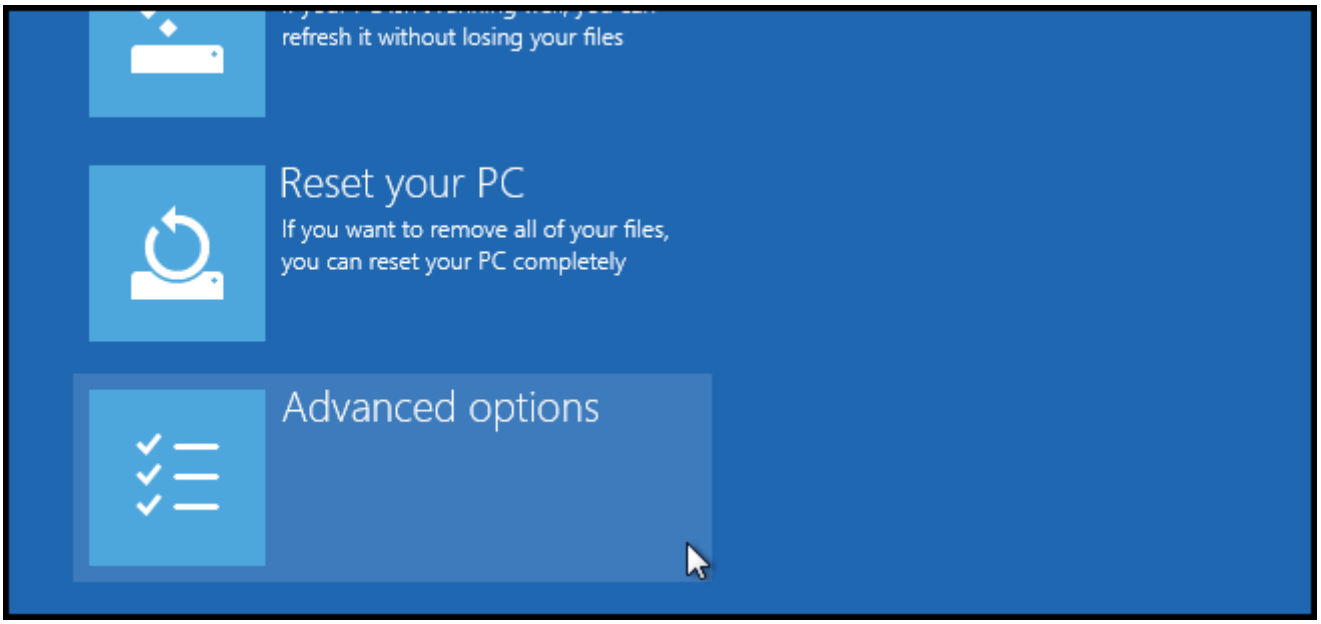
a. Select Restart from the power options menu (on Windows 8 that's under Charms or on the login screen, and in Windows 10 it's on the Start Menu).

b. Hold down the Ctrl +Alt + Shift keys while you click Restart.

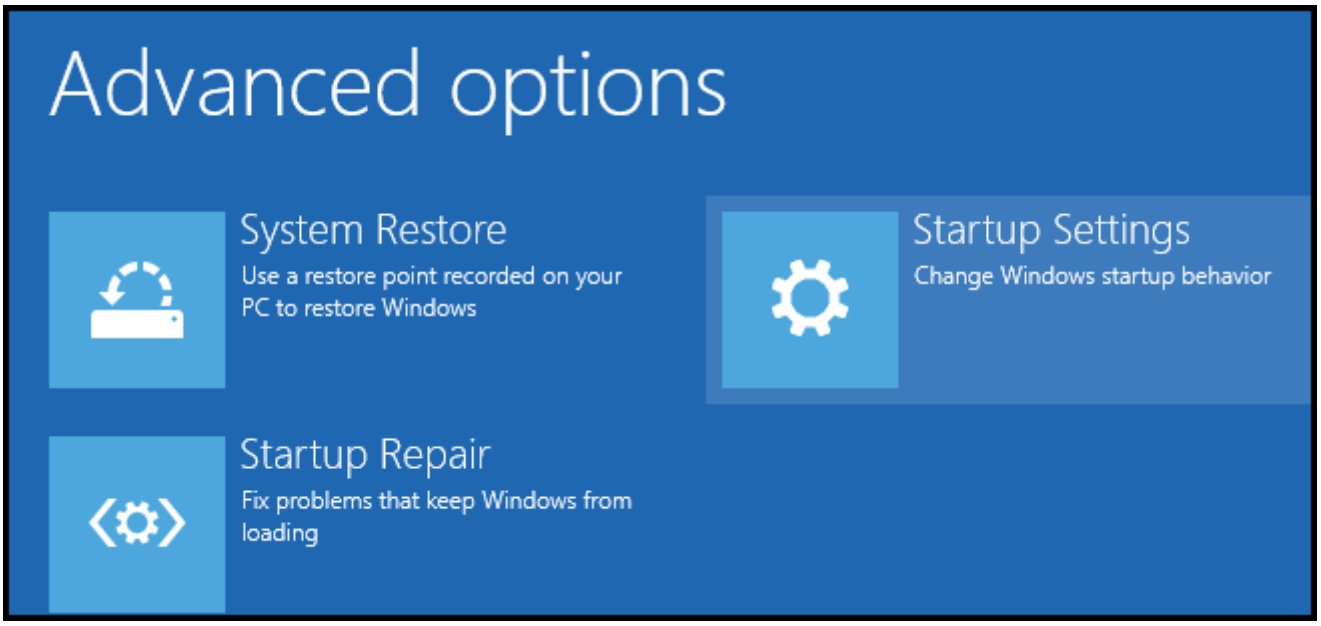
c. Once your computer has rebooted choose the Troubleshoot option.



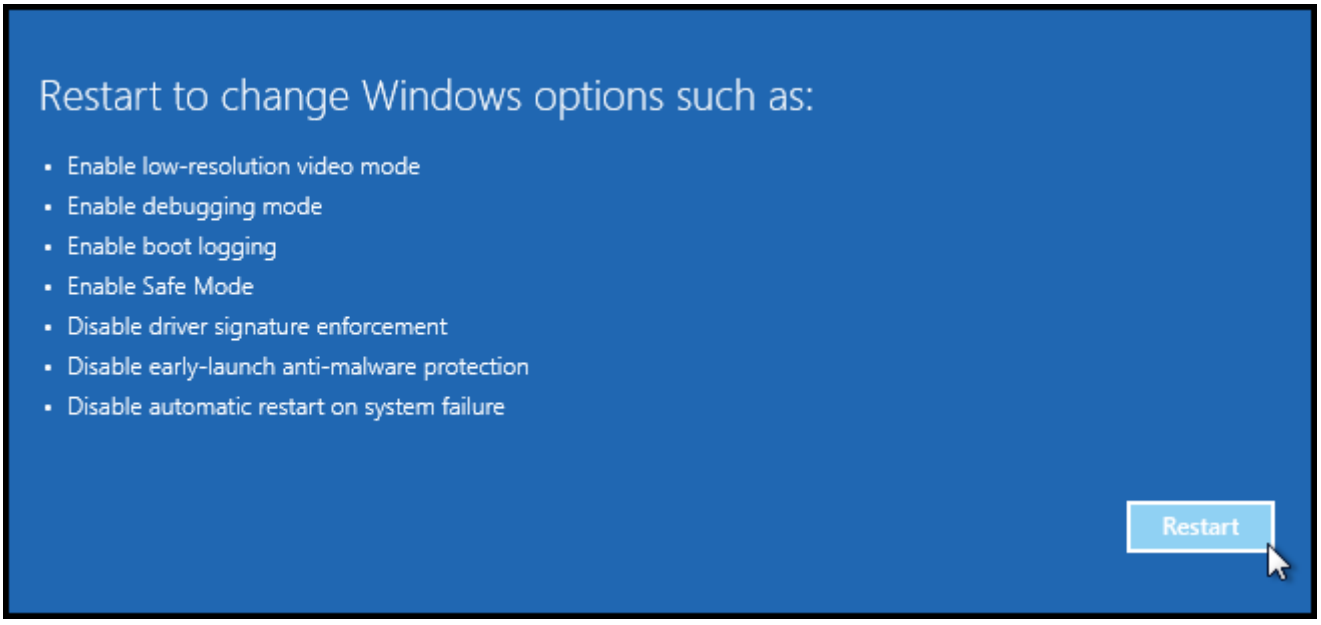
d. Then head into Advanced options.



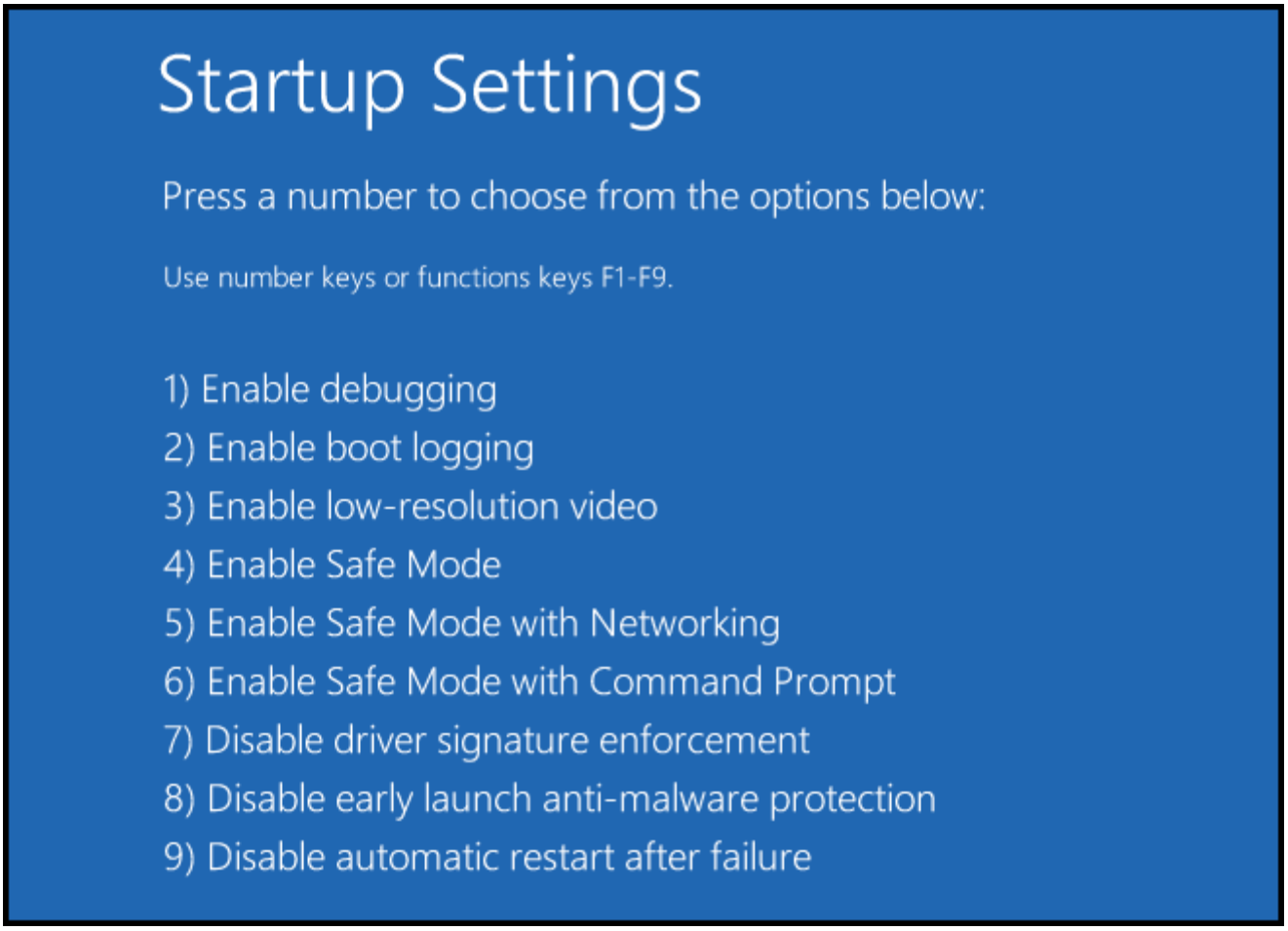
e. Then Startup Settings.



f. Click Restart



g. When you see the window below: Click F7



After following all these steps, the driver can be installed by repeating the instructions from Steps 1 through 3.

5. Click Install the driver anyway, if a warning message pops up after step 3.

6. The driver will now be installed and a message will be shown that the installation was successful.