

ENVISIONING • EMPOWERING • EXCELLING



Windows Driver Customization for USB UARTs

Application Note AN-226

Revision History

Document No.	Release Date	Change Description
226ANR00	12/5/18	Initial release.

Table of Contents

Introduction	1
Overview	1
Reference Documentation.....	1
Customized Registry	2
Overview	2
Custom Config.....	2
Fix COM Port By Dev ID	3
Fix COM Port By Location	4
Installing Driver for Customization Features	6
Overview	6
Installation Process	7
Appendix A: Windows Registry	8
Overview	8
How to Open / Import Windows Registry.....	9
Getting Started to Import Registry	9
How to Modify Windows Registry	10
How to Export Windows Registry	11
How to Remove Registry from Windows	12
Appendix B: Registry Setting Sample	13
Appendix C: FAQs	14

List of Figures

Figure 1: Driver Initialization Flow.....	6
Figure 2: Driver Installation Process for Customization.....	7
Figure 3: Default Customized Registry File	9
Figure 4: Windows Registry Editor	9
Figure 5: Example of Modifying the Customized Registry	10
Figure 6: Customized Registry Export Example	11
Figure 7: Customized Registry File Export Example	11
Figure 8: Removing the Customized Registry Example	12

List of Tables

Table 1: Devices Supported	1
Table 2: Custom Configs	2
Table 3: Fix COM Port BY Dev ID	3
Table 4: usbflags	4
Table 5: Fix COM Port By Location	5

Introduction

Overview

This document describes how to use and customize features for the XR21B and XR21V Series USB UARTs. The detailed description helps to easily achieve the customized features. At the time of this writing, the customization features allow for setting fixed COM port numbers. Other customization features may be added at future times. When a specific customization is needed, for example "Fix COM Port Number by Device ID" or "Fix COM Port Number by Location", this Application Note illustrates and gives guidance on how to realize the desired customization on a Windows 7 platform.

Reference Documentation

This Application Note applies to the devices listed in [Table 1](#). Links to the Data Sheets are provided in the Product column.

Table 1: Devices Supported

Product	PID
XR21V1410	0X1410
XR21V1412	0X1412
XR21V1414	0X1414
XR21B1411	0X1411
XR21B1420	0X1420
XR21B1422	0X1422
XR21B1424	0X1424
XR22801	0x1400
XR22802	0x1400, 0X1401
XR22804	0x1400, 0X1401, 0X1402, 0x1403

1. Since Microsoft is eventually going to move to a DCHU driver model for Windows 10, this Application Note is focused on a customized driver in Windows 7, but it is also applicable for Windows 10 with a non-DCHU driver.

Customized Registry

Overview

This section provides a detailed description for all of the supported customization features and each of their related values.

The basic registry structure for the MaxLinear customization consists of a "CustomConfig" key which contains all customization feature flag values, a subkey for each customization feature and all the specific customization feature's related values that need to be set.

2. If one is not familiar with the Windows Registry, please refer to [Appendix "Appendix A: Windows Registry"](#) for Basic Windows registry instructions.

Custom Config

This contains the summary and the switch of all customized features.

Key Location: "HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig"

[Table 2](#) lists all Custom Config related values.

Table 2: Custom Configs

Value Name	Type	Data Range	Description
FixCOMPortByDevID	DWORD	0 - 1	0 disables the feature. 1 enables the feature if FixCOMPortByLocation is disabled. If FixCOMPortByLocation is also enabled, both features will be disabled.
FixCOMPortByLocation	DWORD	0 - 1	0 disables the feature. 1 enables the feature if FixCOMPortByDevID is disabled. If FixCOMPortByDevID is also enabled, both features will be disabled.

Fix COM Port By Dev ID

This feature is used to Fix COM Port Number(s) for specific device(s) by the device ID. Currently up to 8 different devices (see [footnote 3](#)) can be supported simultaneously. This feature is disabled if Fix COM Port By Location is also set.

Key Location:

"HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig\FixCOMPortByDevID"

[Table 3](#) lists all Fix COM Port By Dev ID related values.

Table 3: Fix COM Port BY Dev ID

Value Name	Type	Data Range	Description
TotalDeviceUsed	DWORD	1 - 8	Defines how many devices are applied to the feature. Currently up to 8 devices are supported.
FixCOMDevice X	DWORD	Refer to PID in Table 1	Defines the PID of the device to be used for this feature. The X should be a suffix starting from 1, up to the number defined in <i>TotalDeviceUsed</i> . For this feature, the FixCOM Device X setting should not be duplicated.
FixCOMNum X	DWORD	1 - 255	Defines the starting port number to be assigned to the FixCOM Device X device with the same X value. The X should be a suffix starting from 1, up to the number defined in <i>TotalDeviceUsed</i> . Please note the maximum number also has to be set in the data range.

3. Device number is tallied by PID. Each different PID is viewed as a device. For example, the 2 channel XR22802 has a PID for each separate channel, thus it is viewed as 2 devices when doing customization.

4. This feature is disabled if multiple devices with the same PID are used. In such a case, the Fix COM Port By Location should be used instead.

Fix COM Port By Location

This feature is used to Fix COM Port Number(s) for specific device(s) based on the actual hub location connected. Currently up to 8 different devices can be supported simultaneously. However, there is no limitation on the number of devices connected as long as there is no COM port number collisions. This feature is disabled if Fix COM Port By Dev ID is also set.

For serialized devices (XR21B and XR2280x Series) to be identified by location, an additional usbflags registry has to be set. This allows the USB driver stack to ignore the serial number of the device. However, the usbflags registry only works for the XR21B devices.(see [footnote 5](#)).

The following states how to set the usbflags registry if needed (see [footnote 6](#)); **please note that a system reboot is required after the modification is done in order to have the modification take effect.**

Key Location: "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\usbflags"

In order to enable this feature, all related usbflags values of the target device should be enabled.

[Table 4](#) lists all related values under usbflags.

Table 4: usbflags

Value Name	Type	Data Range	Description
IgnoreHWSerNumvvvvpppp	Binary	0x0000, 0x0001	Indicates whether the USB driver stack must ignore the serial number of the device. 0x0000 indicates this setting is disabled; 0x0001 indicates it is enabled. vvvv is a 4-digit hexadecimal number of the vendor ID. In this case, it should be 0x04E2. pppp is a 4-digit hexadecimal number of the product ID. All supported PIDs can be found in Table 1 .
IgnoreHWSerNumvvvvppppcc	Binary	0x0000, 0x0001	Indicates whether the USB driver stack must ignore the serial number of this device interface. 0x0000 indicates this setting is disabled; 0x0001 indicates it is enabled. vvvv is a 4-digit hexadecimal number of the vendor ID. In this case, it should be 0x04E2. pppp is a 4-digit hexadecimal number of the product ID. All supported PIDs can be found in Table 1 . cc is the interface number. It should be either 00, 02, 04 or 06.

After usbflags are set, if required for the target device, the related configuration settings for Fix COM Port By Location can be set.

Key Location:

"HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig\FixCOMPortByLocation"

[Table 5](#) lists all related values under Fix COM Port By Location.

Table 5: Fix COM Port By Location

Value Name	Type	Data Range	Description
TotalDeviceUsed	DWORD	1 - 8	Defines how many devices are applied. Up to 8 devices are supported.
FixCOMDeviceX	DWORD	Refer to PID in Table 1	Defines the PID of the device to be used. The X should be a suffix starting from 1 up to the number defined in <i>TotalDeviceUsed</i> . For this feature, <i>FixComDeviceX</i> should not be duplicated (see footnote 5).
FixCOMNumX	DWORD	1 - 255	Defines the starting port number to be assigned to the FixCOMDeviceX device with the same X value. The X should be a suffix starting from 1 up to the number defined in <i>TotalDeviceUsed</i> .. Please note the maximum number also has to be set in the data range. <i>It is strongly recommended that this value is set to the same for all devices to avoid possible port number collisions. See footnote 7.</i>

5. This feature is not supported for the XR2280x Series.

6. See [Application Note AN-217](#) for usbflag settings.

7. The port number calculation is based on the assumption that all USB UART devices are connected to the same hub (root hub or downstream hub).

Installing Driver for Customization Features

Overview

This section illustrates the basic mechanism of the driver version V2.6.0.0 and newer and how the registry comes into play when doing customization.

During driver initialization, the driver will lookup the customization registry and check every value. If the feature is enabled and the related values are all valid, the corresponding configuration settings will be set for the driver to perform as a driver with customized features. If none of the features are enabled, or if any related value of an enabled feature is missing or invalid, the customization initialization process will be bypassed or halted. In this case, all customization related configuration will be reset and the driver will perform as a normal release driver.

Figure 1 shows the initialization flow.

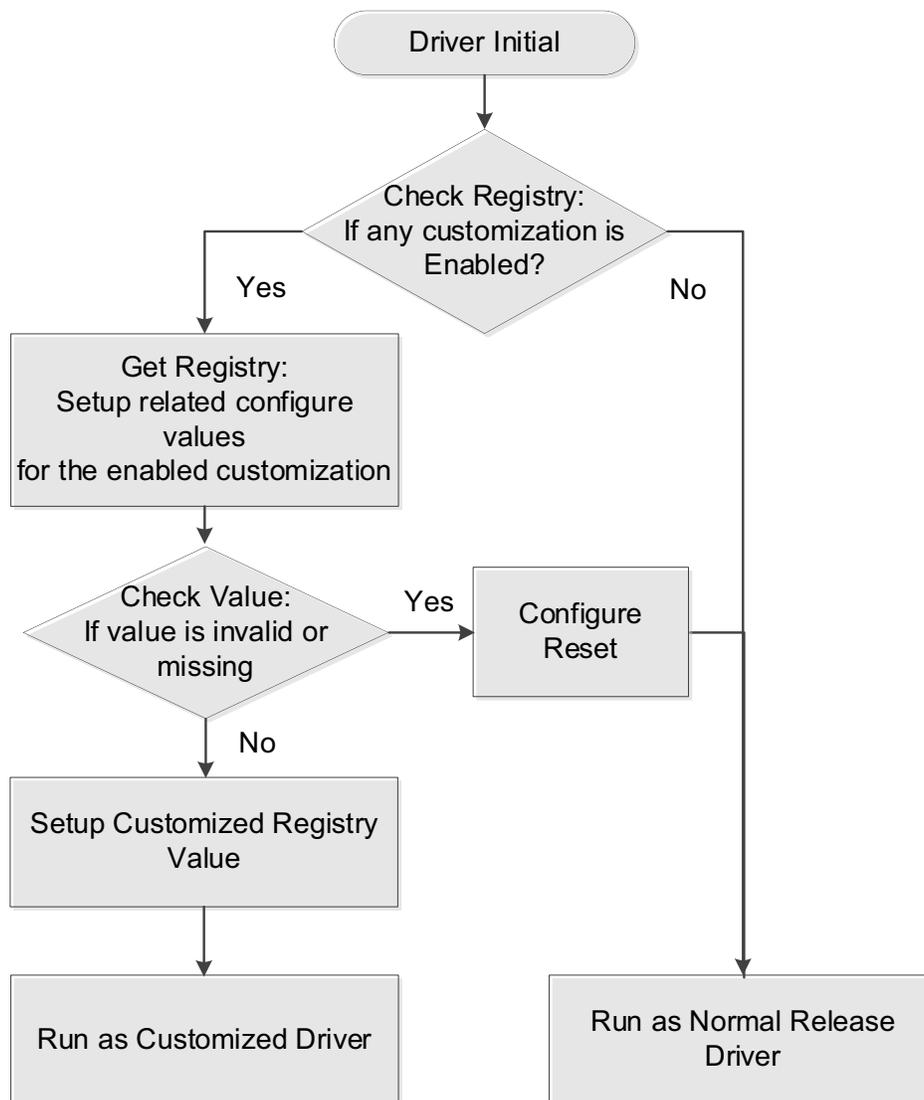


Figure 1: Driver Initialization Flow

Installation Process

This section illustrates the suggested procedure when implementing customization.

In order for the customization to work, the registry settings used for customization has to be modified and applied before installing the driver. After applying the registry setting, a system reboot should be done to make sure the registry is fully applied. Then the driver, version V2.6.0.0 and newer, can be installed. If the customization registry setting is modified correctly, the corresponding customized feature should be able to work.

Figure 2 shows the driver installation process for customization.

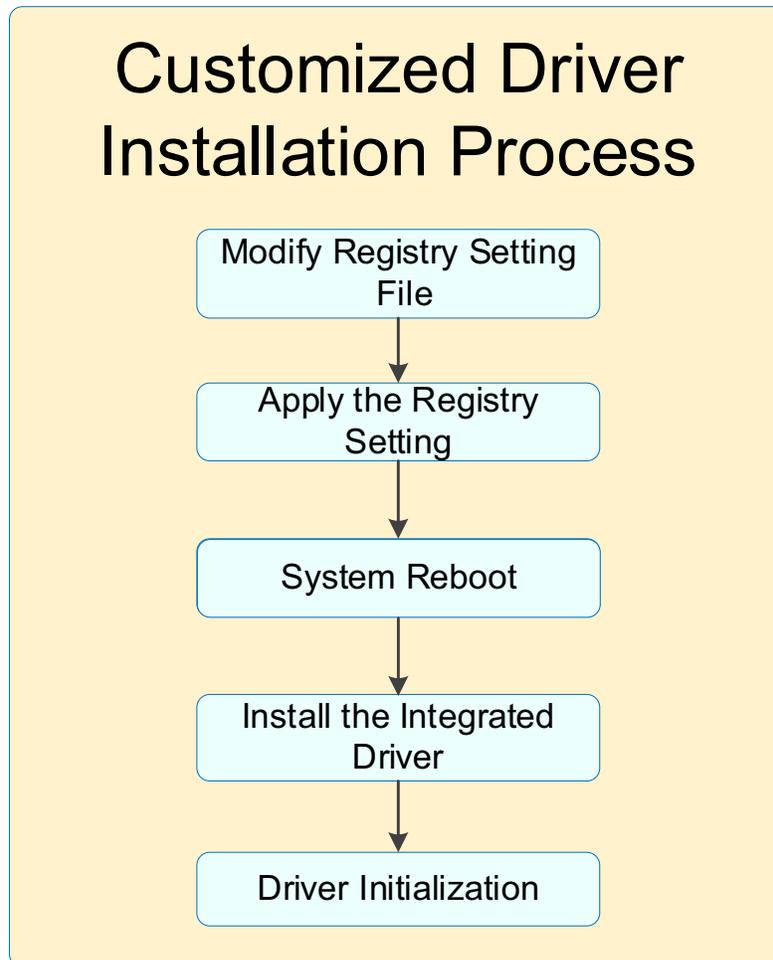


Figure 2: Driver Installation Process for Customization

Appendix A: Windows Registry

Overview

The Windows Registry is a hierarchical database that stores low-level settings for the Microsoft Windows Operating System and for applications using the registry. In simple terms, Windows Registry contains information, settings, options, and other values for programs and hardware installed on all versions of Microsoft Windows Operating Systems. For example, when a program is installed, a new subkey containing settings such as a program's location, its version, and how to start the program, are all added to the Windows Registry.

This section describes how to import, export, modify or remove the registry for customization features. Please be sure you follow each and every step.

8. Before importing, exporting, editing or changing anything in the Microsoft Windows Registry, it is recommended to first backup the registry on your local drive and are familiar with all the Windows registry basics.

How to Open / Import Windows Registry

Getting Started to Import Registry

This sub-section lists the preparation before importing the latest windows driver registry. There are two ways to import the Windows registry. After the imported process is done, it is recommended to reboot the computer to let the registry key take effect for the driver customization.

Solution 1:

Double click the registry file in the Windows 7 platform. Windows 7 will import the contents into the Windows Registry key. [Figure 3](#) shows the registry file name.



Figure 3: Default Customized Registry File

Solution 2:

1. Click the windows "Start button.
2. In the Start Menu, either in the Run Box or the Search box, type "**regedit**" and press Enter.
3. The "Registry Editor" window appears. Select "File" and then "Import" to open the DefaultCustomSample.reg file.

[Figure 4](#) shows the Registry Editor window.

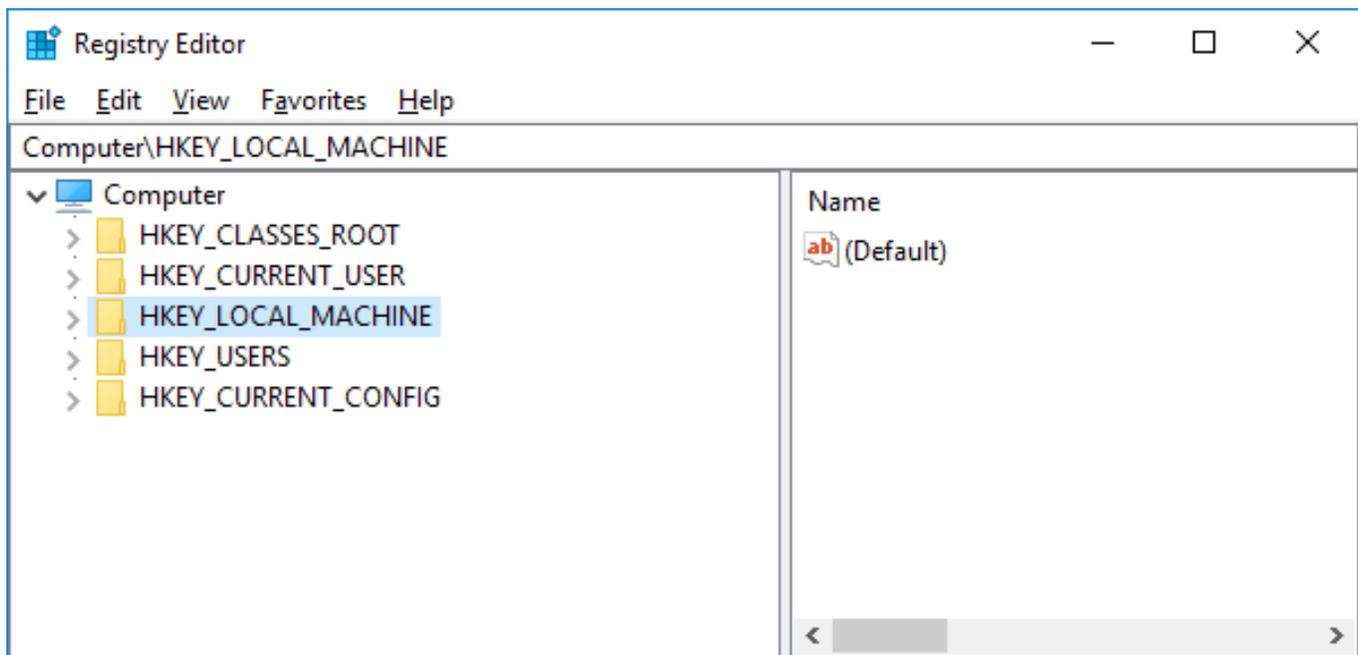


Figure 4: Windows Registry Editor

How to Modify Windows Registry

This section describes how to modify a registry value from Windows. Either double-click or right click on the value name to edit and choose "Modify". A pop-up window appears where the data value can be changed.

Figure 5 illustrates an example of the customized driver key, type and data value. When modifying registry values, it is strongly recommended that the following rules be followed.

- DO NOT modify any registry key names, value names or value types. For example, the value name **FixCOMPortByDevID** corresponds to value type **REG_DWORD**.
- Only modify the data content. For example, the data value of FixCOMPortByDevID is 0x00000001. Note that the REG_DWORD is a hexadecimal value.
- For the valid data range of each customized registry value, see "[Customized Registry](#)"

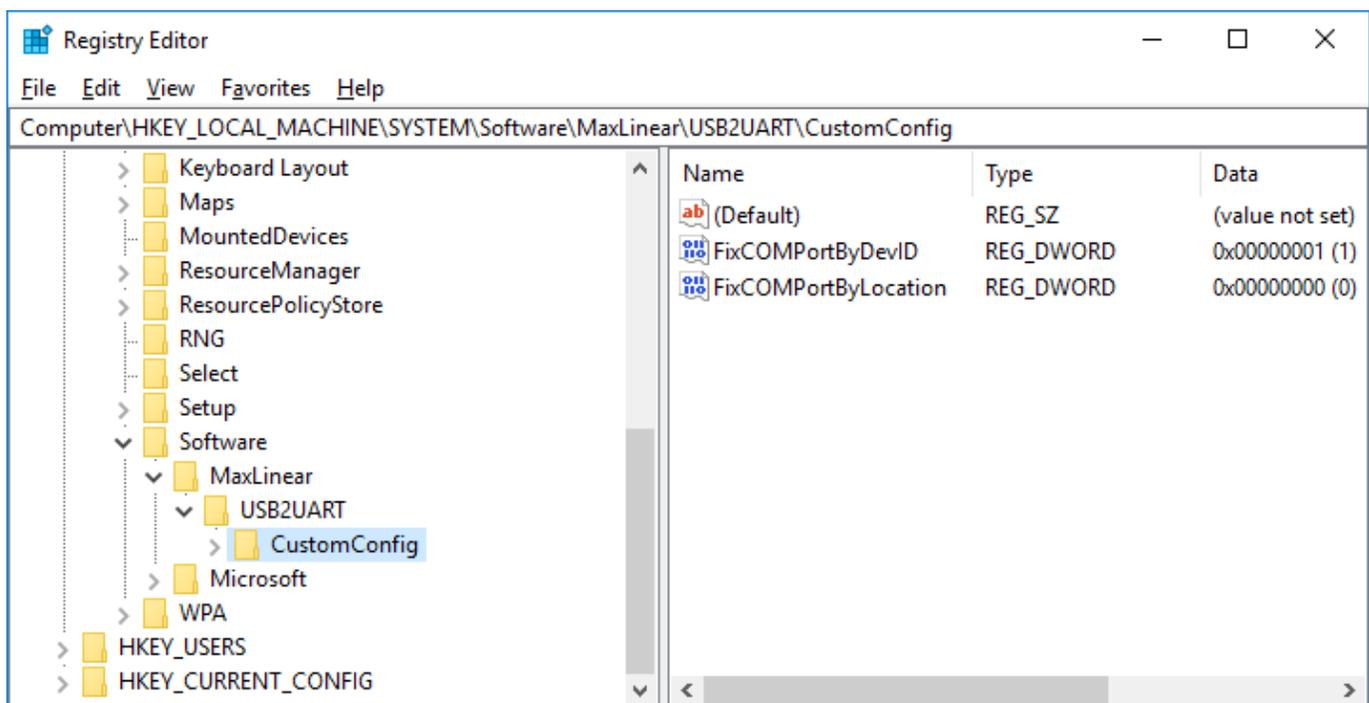


Figure 5: Example of Modifying the Customized Registry

How to Export Windows Registry

This section describes how to export a Windows Registry. If customized registry contents need to be shared, [Figure 6](#) and [Figure 7](#) illustrate how to export and save as a .reg file.

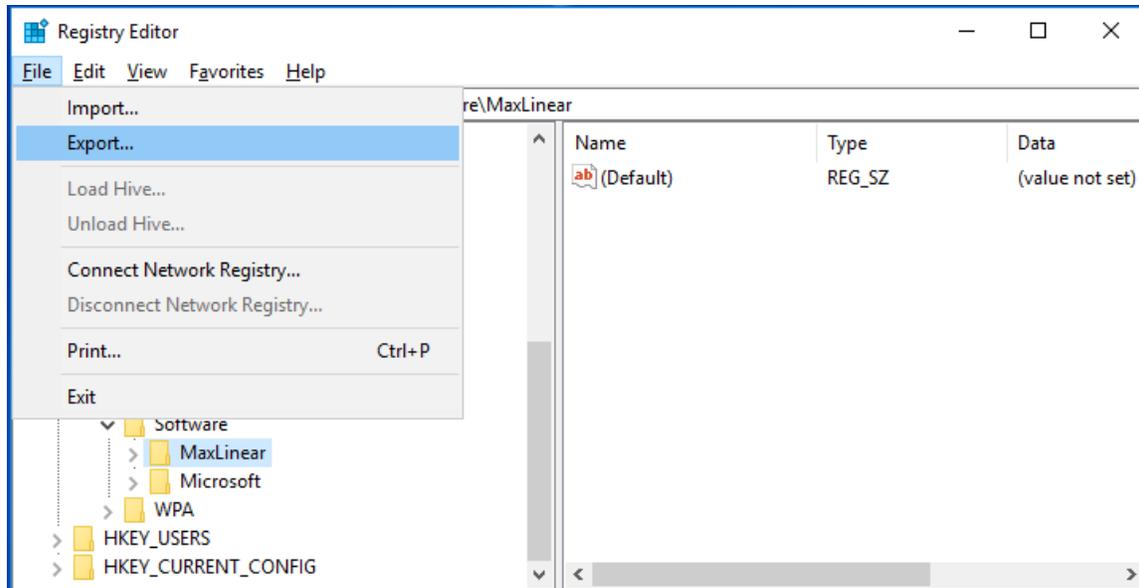


Figure 6: Customized Registry Export Example

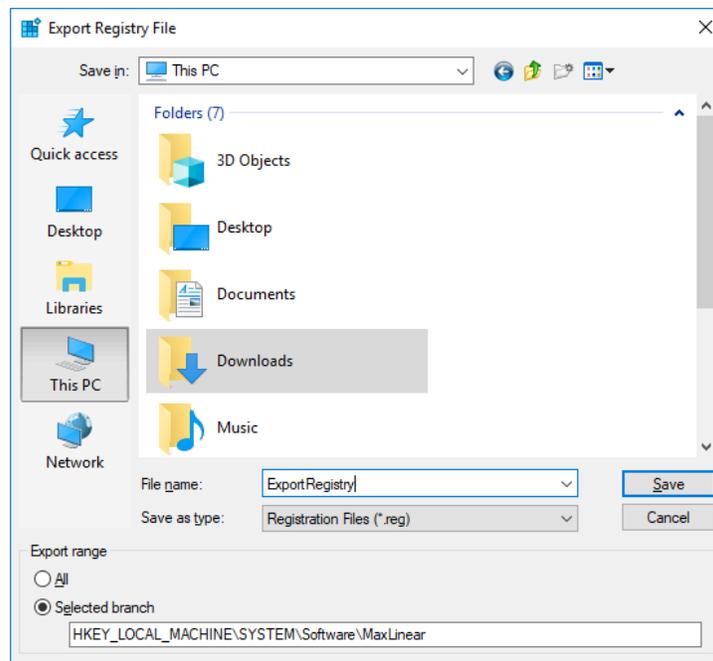


Figure 7: Customized Registry File Export Example

9. The registry exported will only include the subkeys and values starting from the key selected, not the whole registry. If desired, to export the whole customization setting, export from "HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear". (Other related registries might be located elsewhere for specific features).

How to Remove Registry from Windows

This section describes how to remove a registry key from Windows. Here is an example to remove a customized driver's registry key. [Figure 8](#) illustrates removing a customized key from Windows "HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear". Right-click on the "MaxLinear" key and select "Delete" to remove the specific key and all its subkeys.

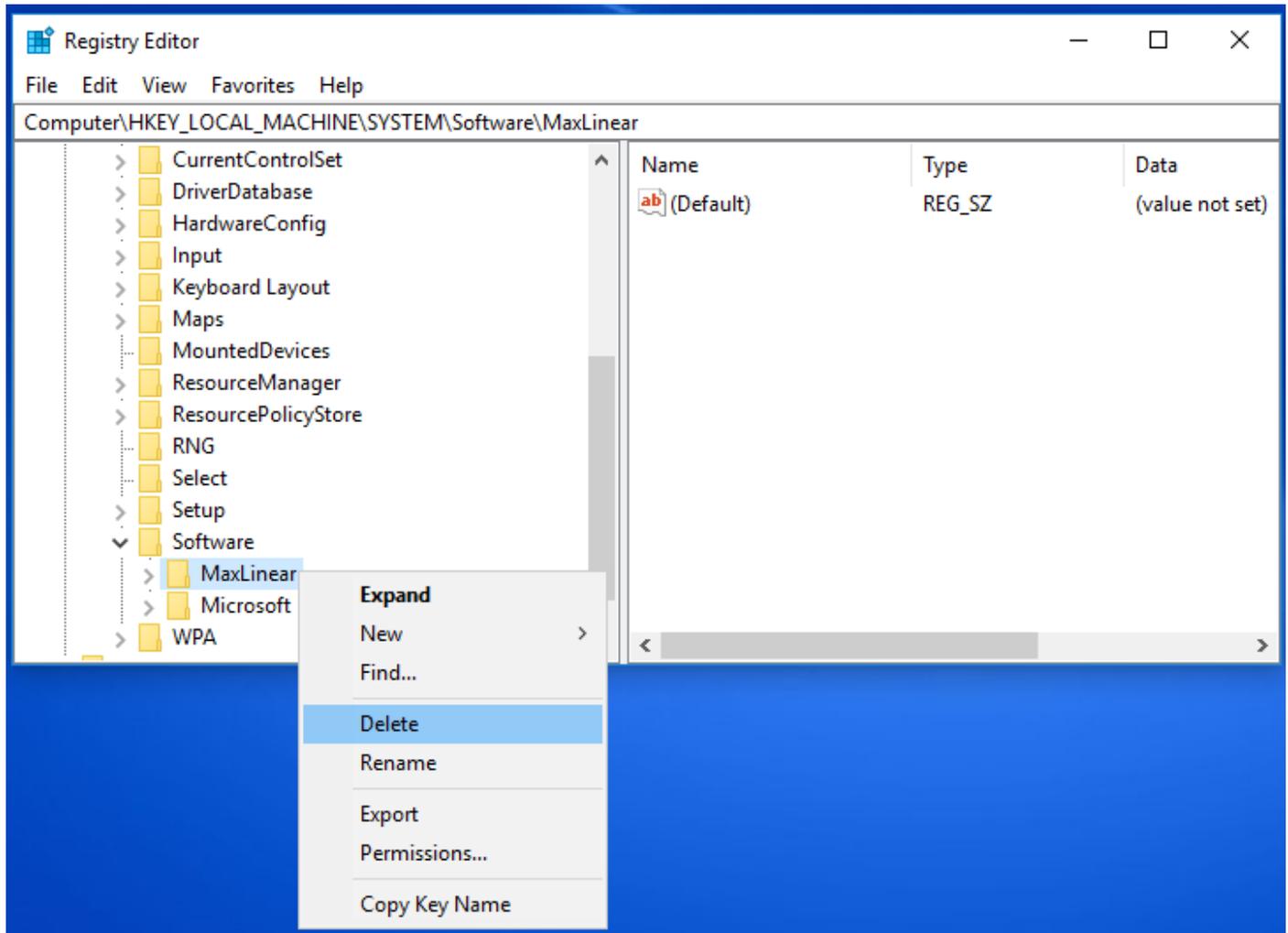


Figure 8: Removing the Customized Registry Example

Appendix B: Registry Setting Sample

File Name: DefaultCustomSample.reg

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\UsbFlags]

"IgnoreHWSerNum04E2141100"=hex:00

"IgnoreHWSerNum04E21411"=hex:00

"IgnoreHWSerNum04E2142000"=hex:00

"IgnoreHWSerNum04E21420"=hex:00

"IgnoreHWSerNum04E2142200"=hex:00

"IgnoreHWSerNum04E2142202"=hex:00

"IgnoreHWSerNum04E21422"=hex:00

"IgnoreHWSerNum04E2142400"=hex:00

"IgnoreHWSerNum04E2142402"=hex:00

"IgnoreHWSerNum04E2142404"=hex:00

"IgnoreHWSerNum04E2142406"=hex:00

"IgnoreHWSerNum04E21424"=hex:00

[HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear]

[HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART]

[HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig]

"FixCOMPortByDevID"=dword:00000001

"FixCOMPortByLocation"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig\FixCOMPortByDevID]

"TotalDeviceUsed"=dword:00000002

"FixCOMDevice1"=dword:00001424

"FixCOMNum1"=dword:00000050

"FixCOMDevice2"=dword:00001411

"FixCOMNum2"=dword:00000064

[HKEY_LOCAL_MACHINE\SYSTEM\Software\MaxLinear\USB2UART\CustomConfig\FixCOMPortByLocation]

"TotalDeviceUsed"=dword:00000004

"FixCOMDevice1"=dword:00001424

"FixCOMNum1"=dword:00000046

"FixCOMDevice2"=dword:00001411

"FixCOMNum2"=dword:00000046

"FixCOMDevice3"=dword:00001422

"FixCOMNum3"=dword:00000046

"FixCOMDevice4"=dword:00001414

"FixCOMNum4"=dword:00000046

Appendix C: FAQs

Q1: What will happen if the registry value type is mismatched with the ones listed in the Application Note?

A1: If the registry value type is set incorrectly, the registry value will not be read correctly. This results in a customization check failure and operation as a normal standard driver. In some rare cases, it might lead to a BSoD (Blue Screen of Death).

Q2: I have disabled the FixCOMPortByDevID feature flag, does this mean that the FixCOMPortByLocation feature will automatically be enabled?

A2: No. Each feature is only enabled when its own specific feature flag is enabled. Disabling a feature does not automatically enable its mutually exclusive features.

Q3: I have enabled the FixCOMPortByLocation feature and the function already seems to be working on XR21V Series devices. Why isn't this function operating correctly on my XR21B device?

A3: For XR21B Series devices, corresponding usbflag registry values have to be set and take effect first before the FixCOMPortByLocation can function correctly. Please refer to [Table 4](#) for details.

Q4: What will happen if usbflags for a XR21V Series device is set for FixCOMPortByLocation?

A4: The purpose of the usbflags registry value set is to have the USB driver stack ignore the Hardware Serial Number of the device. Since XR21V Series devices do not have a Serial Number, setting usbflags for XR21V Series will not have an impact.

Q5: I have already set the usbflags using the FixCOMPortByLocation feature. Why is the port number still assigned by Device ID for XR21B142x on my Windows 7 system?

A5: If the usbflags setting doesn't work in a Windows 7 system, please check if the USB port connected is a USB 3 port. There is a known issue that the usbflags registry setting does not work on USB 3 ports with Windows 7, which is due to an XHCI host driver issue in Windows 7. If the port connected is not a USB 3 port, please check the registry settings.



MaxLinear, Inc.
5966 La Place Court, Suite 100
Carlsbad, CA 92008
760.692.0711 p.
760.444.8598 f.
www.maxlinear.com

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by MaxLinear, Inc. MaxLinear, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced into, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of MaxLinear, Inc.

MaxLinear, Inc. does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless MaxLinear, Inc. receives, in writing, assurances to its satisfaction that: (a) the risk of injury or damage has been minimized; (b) the user assumes all such risks; (c) potential liability of MaxLinear, Inc. is adequately protected under the circumstances.

MaxLinear, Inc. may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from MaxLinear, Inc., the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

MaxLinear, the MaxLinear logo, and any MaxLinear trademarks, MxL, Full-Spectrum Capture, FSC, G.now, AirPHY and the MaxLinear logo are all on the products sold, are all trademarks of MaxLinear, Inc. or one of MaxLinear's subsidiaries in the U.S.A. and other countries. All rights reserved. Other company trademarks and product names appearing herein are the property of their respective owners.