Abstract
This document explains how to install the USB driver in a Windows Embedded OS.
This document applies to the following products:

<table>
<thead>
<tr>
<th>Product name</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISA-U2 series</td>
</tr>
<tr>
<td>SARA-U2 series</td>
</tr>
<tr>
<td>TOBY-L2 series</td>
</tr>
<tr>
<td>MPCI-L2 series</td>
</tr>
<tr>
<td>TOBY-R2 series</td>
</tr>
<tr>
<td>LARA-R2 series</td>
</tr>
</tbody>
</table>
Contents

Contents ............................................................................................................................................. 3

1 Introduction .................................................................................................................................. 4

2 Rights .......................................................................................................................................... 4

3 Getting started .......................................................................................................................... 5

4 Package content ......................................................................................................................... 6
  4.1 LISA-U2 / SARA-U2 / TOBY-R2 / LARA-R2 series ................................................................. 6
  4.2 TOBY-L2 series ..................................................................................................................... 6

5 Installation ................................................................................................................................... 7
  5.1 LISA-U2 / SARA-U2 / TOBY-R2 / LARA-R2 series ................................................................. 7
  5.1.1 Preliminary support for CDC-ECM function ......................................................................... 7
  5.2 TOBY-L2 series ..................................................................................................................... 8
  5.2.1 SW Requirements .............................................................................................................. 8
  5.2.2 Integration steps ................................................................................................................. 8
  5.2.3 Image built verification ....................................................................................................... 11
  5.2.4 Adding drivers to an already released OS image .................................................................. 11

6 Dial-up networking ..................................................................................................................... 12

7 Debug .......................................................................................................................................... 16
  7.1 Virtual COM ports and AT commands .................................................................................. 16
  7.2 Networking ............................................................................................................................ 17

Related documents ......................................................................................................................... 18

Revision history .............................................................................................................................. 18

Contact ........................................................................................................................................... 19
1 Introduction

This document describes the USB driver installation in a Windows Embedded OS.

This document applies to the u-blox USB driver version 1.55.0.0 for Windows Embedded OS.

This document provides all needed information about installation and usage of the provided software. The package version 1.55.0.0 contains the USB driver for CPU architecture ARMV4I. The following embedded operating systems are supported:

- Windows CE 5.0
- Windows CE 6.0
- Windows Embedded Compact 7
- Windows Embedded Automotive 7
- Windows Mobile 5.0
- Windows Mobile 6.0
- Windows Mobile 6.1
- Windows Mobile 6.5

This software was developed and tested on a BeagleBoard-xM platform based on ARM Cortex-A8's core.

u-blox assumes no responsibility for the software functionality on other devices or MPUs.

The MPCI-L2 series provides the same feature set as the TOBY-L2 series. Therefore “TOBY-L2” also refers to MPCI-L2 series. u-blox USB driver version 1.55.0.0 is not allowed to be used with TOBY-L2 series. A strategy to use the USB driver present in the MS Platform Builder core OS is provided.

u-blox USB driver version 1.55.0.0 does not support Windows Embedded Compact 2013. USB driver present in the MS Platform Builder core OS shall be used.

2 Rights

The customer is granted a limited, non-exclusive license to use the driver without the right to sublicense, only to operate the u-blox hardware. It is, unless authorized by mandatory provisions of law, forbidden to decompile, to detect the source code, to change, reproduce or develop the firmware or software as well as to reproduce, change or reverse engineer the hardware. The customer is not allowed to modify the driver package to make his own developments.
3 Getting started

This section describes the main connections and settings required to getting started with the EVK-U20, EVK-U23, EVK-U26, EVK-U27, EVK-L20, EVK-L21, EVK-R200, EVK-R202, EVK-R203, EVK-R204, EVK-R211, EVK-R220, EVK-R280 and the BeagleBoard-xM platform.

To evaluate the u-blox cellular module with BeagleBoard-xM platform these devices are needed:

- BeagleBoard-xM platform
- Evaluation Kit and an adapter board where the u-blox cellular module is soldered
- Power supply for BeagleBoard-xM and the u-blox cellular evaluation kit
- PC
- Keyboard
- Mouse
- Monitor with HDMI or S-Video port
- 1 or 2 USB cable
- DVI-D cable or S-video cable
- Jack in/out
- SD card (if not provided with BeagleBoard-xM platform)
4 Package content

4.1 LISA-U2 / SARA-U2 / TOBY-R2 / LARA-R2 series

The SW delivery for this platform is made up of several <SDK build> directories, each which corresponds to a specific embedded operating system and CPU architecture, and provide binary drivers; see Table 1.

<table>
<thead>
<tr>
<th>SDK build</th>
<th>Embedded operating system</th>
</tr>
</thead>
<tbody>
<tr>
<td>BeagleBoard-xM SDK (ARMV4I)</td>
<td>BeagleBoard-xM SDK for Windows CE6</td>
</tr>
<tr>
<td>BeagleBoard-xM WEC7 (ARMV4I)</td>
<td>BeagleBoard-xM SDK for Windows EC7</td>
</tr>
<tr>
<td>Colibri600 (ARMV4I)</td>
<td>Colibri600 SDK for Windows CE 6</td>
</tr>
<tr>
<td>Colibri (ARMV4I)</td>
<td>Colibri SDK for Windows CE 5</td>
</tr>
<tr>
<td>STANDARDSDK_500 (ARMV4I)</td>
<td>Windows CE 5</td>
</tr>
<tr>
<td>Windows Mobile 5.0 Pocket PC SDK (ARMV4I)</td>
<td>Windows Mobile 5 and greater</td>
</tr>
<tr>
<td>Windows Mobile 6 Professional SDK (ARMV4I)</td>
<td>Windows CE6</td>
</tr>
</tbody>
</table>

Table 1: SDK build and operating system list

Other architectures are available on request (x86, MIPS, SH).

Each <SDK build> delivery is provided with the following content:
- `usb_ce_install.exe`: Windows CE driver installer
- `usbcdc.cfg`: driver configuration file
- `usbcdc.dll`: dynamic library for COM driver
- `usbcdc_ce_inst.CAB`: compressed installation file (only for BeagleBoard-xM <SDK build>)
- `SimpleComTest.exe`: program to send/receive string to/from COM port of Windows CE OS
- `SimpleComTest_devname`: program to send/receive string to/from \$device\COM port of Windows CE OS

4.2 TOBY-L2 series

The drivers for the TOBY-L2 product series are already present in the MS Platform Builder core OS. Section 5.2 explains how to enable the driver and integrate the required components and the correct registry keys in a new OS design.
5 Installation

5.1 LISA-U2 / SARA-U2 / TOBY-R2 / LARA-R2 series

Drivers are provided in binary format and can be integrated in the final Windows CE/EC image without building the OS image.

1) Extract the provided files from the delivery package: delivery_<version>.zip

2) Copy the files onto the SD card where the BeagleBoard-xM OS image will be loaded at module power on.

3) Insert the SD card into BeagleBoard-xM's SD slot.

4) Connect the keyboard and mouse to the other USB Host port.

5) Connect the monitor to the DVI-D connector or to the S-Video connector.

6) Connect the power supply cable to the BeagleBoard and the USB cable to the USB OTG port, if MS Activesync is needed.

7) After the OS system start-up:
   a. For the default installation, execute usbcd_ce_inst.CAB (usbcdc.dll will be copied and the registry keys set)
   b. For the custom installation, copy the file usbcdc.dll to the \Windows directory on the Beagleboard platform and copy usb_ce_install.exe and usbcdc.cfg to a directory on the Beagleboard platform. Execute usb_ce_install.exe. Now the registry settings are made.

8) Power on the module. Connect the USB Host port of BeagleBoard-xM to the module.

   The usbcdc.cfg file is updated according to the product (in particular the PID changes based on the product).

   For the usage of TOBY-R2 / LARA-R2 series modules with Windows Embedded Compact 2013 see the information provided in section 5.2 (only the part explaining CDC-ACM virtual COM support is relevant).

5.1.1 Preliminary support for CDC-ECM function

An optional USB configuration can be selected by means of the +UUSBCONF AT command (see the u-blox AT Commands Manual [5] for the command description and its applicability).

To correctly map the CDC-ACM and the CDC-ECM interfaces, add the following registry keys (for pld 0x1104):

[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5445_4356\Default\Default\USBCDC_ClassDriver]
"Dll"="usbcdc.dll"
As a consequence of configuring the USB interface, if any other USB device wants to use the `usbser.dll`, it has to be addressed by VID/PID static mapping.

5.2 TOBY-L2 series

The drivers (binary DLL) are not provided as an installer package, because they are already present in the Platform Builder Core OS sources, but they must be enabled before building the image of the targeted OS. The following explanations refer to Windows Embedded Compact 7 (EC7), but they can also be used for Windows CE 6.0 and Windows Embedded Compact 2013.

A complete explanation of the procedure to build a working Windows Embedded Compact OS Design is beyond the scope of this document.

5.2.1 SW Requirements

To build an OS design for BeagleBoard-xM (or equivalent embedded platform) the following parts are needed:

1) Visual Studio 2008 with Service Pack 1
2) Platform Builder plugin for Windows EC7
3) Board Support Package (BSP) for BeagleBoard-xM
4) Make sure Monthly Update January 2013 is installed ([http://support.microsoft.com/kb/2806587](http://support.microsoft.com/kb/2806587))

Requirements change based on OS and board. Here Windows EC7 on BeagleBoard-xM is considered.

5.2.2 Integration steps

The following actions are needed to add the features which will enable RNDIS networking interface and virtual COM ports to the current OS design:

1) From the catalog, add RNDIS support:
   - Core OS > Windows Embedded Compact > Device Drivers > USB > USB Host > USB Class Drivers > USB RNDIS Class Driver (which corresponds to `SYSGEN_ETH_USB_HOST` variable)

2) Inside OS design properties, add Environment variable to enable CDC-ACM virtual COM support:
   - `SYSGEN_USB_SER = 1`

3) Insert the following registry keys to customize OS USB enumeration:

   Windows Embedded CE 6.0:

   ```
   [HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\Default\224]
   "dll"="usb8023.dll"
   [HKEY_LOCAL_MACHINE\usbserDrivers\USB\LoadClients\Default\Default\10\usbser_class]
   ```
"prefix"="COM"
"dll"="usbser.dll"

[HKEY_LOCAL_MACHINE\Drivers\Unimodem\Settings]
"MdmLogFile"=dword:1
  "Blind_Off"="e0"
  "Blind_On"="e0"

[HKEY_LOCAL_MACHINE\ExtModems\U-blox-Modem]
  "Tsp"="Unimodem.dll"
  "Port"="COM2:"
  "DeviceType"=dword:1
  "Order"=dword:0
  "FriendlyName"="U-blox Cellular Modem"

[HKEY_LOCAL_MACHINE\Drivers\USB\ClientDrivers\UbloxFlashLoader]
"dll"="\\Storage card\ubxdummy.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416\Default]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416\Default\Default\UbloxFlashLoader]
"dll"="\\Storage card\ubxdummy.dll"

; Removal of RNDIS association to Function Class 239 (0xf0)
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\239]

Windows Embedded Compact 7
Windows Embedded Compact 2013

[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\Default\224]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\Default\224\usb8023]
"dll"="usb8023.dll"

[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\Default\10]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\Default\10\usbser_class]
"prefix"="COM"
"dll"="usbser.dll"

[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416\Default\Default\usbser_class]
"dll"="usbser.dll"

[HKEY_LOCAL_MACHINE\Drivers\USB\ClientDrivers\usbser_class\Unimodem]
"DeviceType"=dword:00000001
"Tsp"="unimodem.dll"

[HKEY_LOCAL_MACHINE\ExtModems\U-blox-Modem]
  "Port"="COM2:"
  "DeviceType"=dword:1
"FriendlyName"="U-blox Cellular Modem"

[HKEY_LOCAL_MACHINE\ExtModems\U-blox-Modem\Init]
[HKEY_LOCAL_MACHINE\ExtModems\U-blox-Modem\Settings]
"MdmLogFile"=dword:1
  "Blind_Off"=""
  "Blind_On"=""

[HKEY_LOCAL_MACHINE\Drivers\USB\ClientDrivers\UbloxFlashLoader]
"dll"="\\Storage card\\ubxdummy.dll"
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416\Default]
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\5446_4416\Default\Default\UbloxFlashLoader]
"DLL"="\\Storage card\\ubxdummy.dll"

; Removal of RNDIS association to Function Class 239 (0xf0)
[-HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\239]

; Removal of RNDIS association to Function Class 2 (0x02)
[-HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\Default\2]

Figure 1 shows an example of OS design where the two features are enabled:

Figure 1: Example of OS design
5.2.3 Image built verification
After the OS image has been built, follow steps 3) to 8) of section 5.1. With the OS booted, verify that the system recognizes the RNDIS Network interface as USB80231, as shown in the following picture:

![Figure 2: RNDIS network interface recognition](image)

5.2.4 Adding drivers to an already released OS image
If it is not possible at the build time to embed the needed drivers inside the OS image (e.g. because the final device is already in use in the field), then it is possible to extract them from the build system and package them in a suitable format that can be used at run-time stage.

Among the command line tools provided with the installation of the Windows Embedded Compact build system, there is the CAB Wizard tool (https://msdn.microsoft.com/en-us/library/ee504656.aspx). This tool is able to produce a cabinet (.cab) file that can be used to facilitate installation of applications or drivers on the target device. The example below shows the usage of the command:

```
cabwiz.exe "c:\myfile.inf" /err myfile.err /cpu MyARMDevice
```

The input file “myfile.inf“ must be written following the instruction provided here https://msdn.microsoft.com/en-us/library/ee504709.aspx to specify the two dlls (usb8023.dll and usbser.dll) that must be copied to the target and the registry keys presented in section 5.2.2.
6 Dial-up networking

To create a dial-up connection, perform these steps:

1) Open the “Settings > Network and Dial-up Connections” window and start the “Make New Connection wizard”.

![Figure 3: Dial up connection setup](image)

2) Select the Cellular Modem (the textual name can be changed by modifying the registry keys).

![Figure 4: Modem name insertion](image)
3) Configure the module, leaving “Port Settings” unchanged and setting “Calling Options”:
   a. Uncheck “Wait for dial tone before dialing”
   b. Specify “Extra Settings”, adding +cgdcont=1,“IP”, “<apn_of_your_provider>”

![Modem properties setting](image1)

Figure 5: Modem properties setting

4) Go through the next step, setting the phone number *99***1# and click the “Finish” button.

![Modem properties setup (phone number)](image2)

Figure 6: Modem properties setup (phone number)
5) Now the new connection can be opened.

![Image of dial-up connection window]

Figure 7: Opening the dial up connection


![Image of dial-up properties window]

Figure 8: Dial up properties
7) Establish the connection by clicking “Connect” and wait few seconds for the message box, which confirms the “Connected” status.

![Figure 9: Dial up connection establishment](image)

If the ActiveSync connection is in use, discover the TCP/IP default gateway associated with the PC and remove it with the commands “route print” and “route delete 0.0.0.0 192.168.55.100”:

![Figure 10: ActiveSync connection removal](image)
7 Debug

7.1 Virtual COM ports and AT commands

Use the sample application, SimpleComTest, to issue AT commands and test the USB modem connection.

The preliminary configuration can be done using the “Settings“ tab where the port name and speed can be specified. Then open the COM port and chose the “Communication“ tab.

Figure 11: SimpleComTest preliminary configuration

Click the “Send“ button to transmit AT commands to the selected COM port (select radio button to append carriage return or line feed or both at the end of the string).

Figure 12: AT commands transmission on SimpleComTest
7.2 Networking

MS Embedded OS provides a sample tool to trace the network traffic activity (http://msdn.microsoft.com/en-us/library/ee495089%28v=winembedded.70%29.aspx), which can be analyzed offline with Windows Network Monitor or similar products such as Wireshark. This tool is also very helpful to debug PPP issues. The utility is called “netlog” and can be inserted into the OS image through the catalog (with the activation of the environment variable SYSGEN_NETLOG):
Related documents


For regular updates to u-blox documentation and to receive product change notifications register on our homepage.

Revision history

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Name</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>R01</td>
<td>24-Oct-2014</td>
<td>mace</td>
<td>Initial release</td>
</tr>
<tr>
<td>R02</td>
<td>13-May-2015</td>
<td>mace</td>
<td>Added section to describe CAB Wizard tool usage; Extended applicability to MPCI-L2</td>
</tr>
<tr>
<td>R03</td>
<td>04-Aug-2016</td>
<td>mace</td>
<td>Added section 5.1.1 Preliminary support for CDC-ECM function</td>
</tr>
<tr>
<td>R04</td>
<td>19-Sep-2017</td>
<td>mben</td>
<td>Document applicability extended to TOBY-R2 / LARA-R2 series</td>
</tr>
</tbody>
</table>
Contact

For complete contact information visit us at www.u-blox.com

u-blox Offices

North, Central and South America
u-blox America, Inc.
Phone: +1 703 483 3180
E-mail: info_us@u-blox.com
Regional Office West Coast:
Phone: +1 408 573 3640
E-mail: info_us@u-blox.com
Technical Support:
Phone: +1 703 483 3185
E-mail: support_us@u-blox.com

Headquarters
Europe, Middle East, Africa
u-blox AG
Phone: +41 44 722 74 44
E-mail: info@u-blox.com
Support: support@u-blox.com

Documentation Feedback
E-mail: docsupport@u-blox.com

Asia, Australia, Pacific
u-blox Singapore Pte. Ltd.
Phone: +65 6734 3811
E-mail: info_ap@u-blox.com
Support: support_ap@u-blox.com

Regional Office Australia:
Phone: +61 2 8448 2016
E-mail: info_anz@u-blox.com
Support: support_ap@u-blox.com

Regional Office China (Beijing):
Phone: +86 10 68 133 545
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Chongqing):
Phone: +86 23 6815 1588
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shanghai):
Phone: +86 21 6090 4832
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shenzhen):
Phone: +86 755 8627 1083
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office India:
Phone: +91 80 4050 9200
E-mail: info_in@u-blox.com
Support: support_in@u-blox.com

Regional Office Japan (Osaka):
Phone: +81 6 6941 3660
E-mail: infojp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Japan (Tokyo):
Phone: +81 3 5775 3850
E-mail: infojp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Korea:
Phone: +82 2 542 0861
E-mail: info_kr@u-blox.com
Support: support_kr@u-blox.com

Regional Office Taiwan:
Phone: +886 2 2657 1090
E-mail: info_tw@u-blox.com
Support: support_tw@u-blox.com