

UC20 WinCE USB Driver **User Guide**

UMTS/HSPA Module Series

Rev. UC20_WinCE_USB_Driver_User_Guide_V1.0

Date: 2013-08-12



Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarter:

Quectel Wireless Solutions Co., Ltd.

Room 501, Building 13, No.99, Tianzhou Road, Shanghai, China, 200233

Tel: +86 21 5108 6236

Mail: info@quectel.com

Or our local office, for more information, please visit:

<http://www.quectel.com/support/salesupport.aspx>

For technical support, to report documentation errors, please visit:

<http://www.quectel.com/support/techsupport.aspx>

GENERAL NOTES

QUECTEL OFFERS THIS INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN ARE SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

THIS INFORMATION CONTAINED HERE IS PROPRIETARY TECHNICAL INFORMATION OF QUECTEL CO., LTD. TRANSMITTABLE, REPRODUCTION, DISSEMINATION AND EDITING OF THIS DOCUMENT AS WELL AS UTILIZATION OF THIS CONTENTS ARE FORBIDDEN WITHOUT PERMISSION. OFFENDERS WILL BE HELD LIABLE FOR PAYMENT OF DAMAGES. ALL RIGHTS ARE RESERVED IN THE EVENT OF A PATENT GRANT OR REGISTRATION OF A UTILITY MODEL OR DESIGN.

Copyright © Quectel Wireless Solutions Co., Ltd. 2013. All rights reserved.

About the document

History

| Revision | Date | Author | Description |
|----------|------------|------------|-------------|
| 1.0 | 2013-08-12 | Wythe WANG | Initial |

Contents

| | |
|---|-----------|
| About the document | 2 |
| Contents | 3 |
| Table Index..... | 4 |
| Figure Index | 5 |
| 1 Introduction | 6 |
| 2 USB Interface Descriptor | 7 |
| 2.1. Composite Communication Device Enumeration | 7 |
| 2.1.1. Device Descriptor | 7 |
| 2.1.2. Configuration | 8 |
| 2.1.3. Interface 0 (DM Port)..... | 8 |
| 2.1.4. Interface 1 (NMEA Port) | 9 |
| 2.1.5. Interface 2 (AT Port) | 11 |
| 2.1.6. Interface 3 (Modem Port) | 12 |
| 2.1.7. Interface 4 (NDIS Port)..... | 14 |
| 3 System Integrating..... | 17 |
| 3.1. Introduction of Driver Package | 17 |
| 3.2. Integrate USB Driver Files | 17 |
| 3.2.1. Check System Component | 17 |
| 3.2.2. Copy Files | 17 |
| 3.2.3. Modify the Platform.reg | 18 |
| 3.2.4. Modify the Platform.bib | 18 |
| 3.2.5. Rebuild and Create System Image | 18 |
| 3.3. USB COM Port..... | 18 |
| 4 COM Port for Application | 20 |
| 4.1. Testing AT Commands on AT Port | 20 |
| 4.2. Create PPP Connection on Modem Port | 20 |
| 5 Appendix A Reference..... | 28 |

Table Index

| | |
|--|----|
| TABLE 1: DEVICE DESCRIPTOR..... | 7 |
| TABLE 2: CONFIGURATION DESCRIPTOR\CONFIGURATION..... | 8 |
| TABLE 3: CONFIGURATION DESCRIPTOR\INTERFACE 0..... | 8 |
| TABLE 4: CONFIGURATION DESCRIPTOR\INTERFACE 0\ENDPOINT DESCRIPTOR 1 | 9 |
| TABLE 5: CONFIGURATION DESCRIPTOR\INTERFACE 0\ENDPOINT DESCRIPTOR 2 | 9 |
| TABLE 6: CONFIGURATION DESCRIPTOR\INTERFACE 1..... | 9 |
| TABLE 7: CONFIGURATION DESCRIPTOR\INTERFACE 1\ENDPOINT DESCRIPTOR 1 | 10 |
| TABLE 8: CONFIGURATION DESCRIPTOR\INTERFACE 1\ENDPOINT DESCRIPTOR 2 | 11 |
| TABLE 9: CONFIGURATION DESCRIPTOR\INTERFACE 2..... | 11 |
| TABLE 10: CONFIGURATION DESCRIPTOR\INTERFACE 2\ENDPOINT DESCRIPTOR 1 | 12 |
| TABLE 11: CONFIGURATION DESCRIPTOR\INTERFACE 2\ENDPOINT DESCRIPTOR 2 | 12 |
| TABLE 12: CONFIGURATION DESCRIPTOR\INTERFACE 3 | 12 |
| TABLE 13: CONFIGURATION DESCRIPTOR\INTERFACE 3\ENDPOINT DESCRIPTOR 1 | 13 |
| TABLE 14: CONFIGURATION DESCRIPTOR\INTERFACE 3\ENDPOINT DESCRIPTOR 2 | 13 |
| TABLE 15: CONFIGURATION DESCRIPTOR\INTERFACE 3\ENDPOINT DESCRIPTOR 3 | 14 |
| TABLE 16: CONFIGURATION DESCRIPTOR\INTERFACE 4 | 14 |
| TABLE 17: CONFIGURATION DESCRIPTOR\INTERFACE 4\ENDPOINT DESCRIPTOR 1 | 15 |
| TABLE 18: CONFIGURATION DESCRIPTOR\INTERFACE 4\ENDPOINT DESCRIPTOR 2 | 15 |
| TABLE 19: CONFIGURATION DESCRIPTOR\INTERFACE 4\ENDPOINT DESCRIPTOR 3 | 15 |
| TABLE 20: RELATIONSHIP BETWEEN INTERFACES AND COM DEVICES | 19 |
| TABLE 21: TERMS AND ABBREVIATIONS | 28 |

Figure Index

| | |
|--|----|
| FIGURE 1: USB DRIVER PACKAGE STRUCTURE | 17 |
| FIGURE 2: OPEN CONTROL PANEL | 20 |
| FIGURE 3: CLICK NETWORK AND DIAL-UP CONNECTIONS..... | 21 |
| FIGURE 4: CLICK MAKE NEW CONNECTION | 21 |
| FIGURE 5: MAKE NEW CONNECTION INTERFACE..... | 22 |
| FIGURE 6: MODEM SELECTION INTERFACE | 22 |
| FIGURE 7: SELECT "QUECTEL UC20 HSUSB MODEM" | 23 |
| FIGURE 8: CONNECTION PROPERTIES..... | 23 |
| FIGURE 9: SET APN | 24 |
| FIGURE 10: SET PHONE NUMBER..... | 24 |
| FIGURE 11: ADD NEW CONNECTION OK | 25 |
| FIGURE 12: DIAL-UP CONNECTION INTERFACE | 25 |
| FIGURE 13: ADD USER NAME AND PASSWORD..... | 26 |
| FIGURE 14: BEGIN TO SET-UP PPP CONNECTION | 26 |
| FIGURE 15: PPP CONNECTED | 27 |

1 Introduction

This document mainly introduces how to integrate the USB driver for UC20 module in WinCE 6.0 OS, and how to use the USB port after the USB driver is loaded successfully.

Quectel
Confidential

2 USB Interface Descriptor

This chapter lists the USB interface descriptors for UC20 as composite communication device.

2.1. Composite Communication Device Enumeration

2.1.1. Device Descriptor

Table 1: Device Descriptor

| Name | Value | Dec | Hex |
|--------------------|-------------------------------------|-------|--------|
| bLength | 18 | 18 | 0x12 |
| bDescriptorType | DEVICE | 1 | 0x01 |
| bcdUSB | 2.0 | 512 | 0x0200 |
| bDeviceClass | Class defined at interface level | 0 | 0x00 |
| bDeviceSubClass | Subclass defined at interface level | 0 | 0x00 |
| bDeviceProtocol | None | 0 | 0x00 |
| bMaxPacketSize0 | 64 | 64 | 0x40 |
| idVendor | 0x05c6 | 1478 | 0x05c6 |
| idProduct | 0x9003 | 36867 | 0x9003 |
| bcdDevice | 0.0 | 0 | 0x0000 |
| iManufacturer | 3 | 3 | 0x03 |
| iProduct | 2 | 2 | 0x02 |
| iSerialNumber | 0 | 0 | 0x00 |
| bNumConfigurations | 1 | 1 | 0x01 |

2.1.2. Configuration

Table 2: Configuration Descriptor\Configuration

| Name | Value | Dec | Hex |
|---------------------------|---------------|-----|--------|
| bLength | Valid | 9 | 0x09 |
| bDescriptorType | CONFIGURATION | 2 | 0x02 |
| wTotalLength | 138 bytes | 138 | 0x008A |
| bNumInterfaxe | 5 | 5 | 0x05 |
| bConfigurationValue | 1 | 1 | 0x01 |
| iConfiguration | 1 | 1 | 0x01 |
| bmAttributes | 0xE0 | 224 | 0xE0 |
| bmAttributes.RemoteWakeup | Supported | 1 | 0x01 |
| bmAttributes.SelfPowered | Yes | 1 | 0x01 |
| bmAttributes.Reserved7 | One | 1 | 0x01 |
| bMaxPower | 500 mA | 250 | 0xFA |

2.1.3. Interface 0 (DM Port)

Table 3: Configuration Descriptor\Interface 0

| Name | Value | Dec | Hex |
|-------------------|-----------------|-----|------|
| bLength | Valid | 9 | 0x09 |
| bDescriptorType | INTERFACE | 4 | 0x04 |
| bInterfaceNumber | 0 | 0 | 0x00 |
| bAlternateSetting | 0 | 0 | 0x00 |
| bNumEndpoints | 2 | 2 | 0x02 |
| bInterfaceClass | Vendor-specific | 255 | 0xFF |

| | | | |
|---------------------------|-----------------|-----|------|
| bInterfaceSubClass | Vendor-specific | 255 | 0xFF |
| bInterfaceProtocol | Vendor-specific | 255 | 0xFF |
| iInterface | 0 | 0 | 0x00 |

Table 4: Configuration Descriptor\Interface 0\Endpoint Descriptor 1

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 1 IN ¹ | 129 | 0x81 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

Table 5: Configuration Descriptor\Interface 0\Endpoint Descriptor 2

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 1 OUT ¹ | 1 | 0x01 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

2.1.4. Interface 1 (NMEA Port)

Table 6: Configuration Descriptor\Interface 1

| Name | Value | Dec | Hex |
|---------------------------|-----------------|-----|------|
| bLength | Valid | 9 | 0x09 |
| bDescriptorType | INTERFACE | 4 | 0x04 |
| bInterfaceNumber | 1 | 1 | 0x01 |
| bAlternateSetting | 0 | 0 | 0x00 |
| bNumEndpoints | 2 | 2 | 0x02 |
| bInterfaceClass | Vendor-specific | 255 | 0xFF |
| bInterfaceSubClass | Vendor-specific | 255 | 0xFF |
| bInterfaceProtocol | Vendor-specific | 255 | 0xFF |
| iInterface | 0 | 0 | 0x00 |

Table 7: Configuration Descriptor\Interface 1\Endpoint Descriptor 1

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 2 IN ¹ | 130 | 0x82 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹ Given endpoint address is an example and can be modified.

Table 8: Configuration Descriptor\Interface 1\Endpoint Descriptor 2

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 2 OUT ¹ | 2 | 0x02 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

2.1.5. Interface 2 (AT Port)

Table 9: Configuration Descriptor\Interface 2

| Name | Value | Dec | Hex |
|---------------------------|-----------------|-----|------|
| bLength | Valid | 9 | 0x09 |
| bDescriptorType | INTERFACE | 4 | 0x04 |
| bInterfaceNumber | 2 | 2 | 0x02 |
| bAlternateSetting | 0 | 0 | 0x00 |
| bNumEndpoints | 2 | 2 | 0x02 |
| bInterfaceClass | Vendor-specific | 255 | 0xFF |
| bInterfaceSubClass | Vendor-specific | 255 | 0xFF |
| bInterfaceProtocol | Vendor-specific | 255 | 0xFF |
| iInterface | 0 | 0 | 0x00 |

Table 10: Configuration Descriptor\Interface 2\Endpoint Descriptor 1

| Name | Value | Dec | Hex |
|------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 3 IN ¹ | 131 | 0x83 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

Table 11: Configuration Descriptor\Interface 2\Endpoint Descriptor 2

| Name | Value | Dec | Hex |
|------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 3 OUT ¹ | 3 | 0x03 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

2.1.6. Interface 3 (Modem Port)

Table 12: Configuration Descriptor\Interface 3

| Name | Value | Dec | Hex |
|------------------|-----------|-----|------|
| bLength | Valid | 9 | 0x09 |
| bDescriptorType | INTERFACE | 4 | 0x04 |
| bInterfaceNumber | 3 | 3 | 0x03 |

| | | | |
|---------------------------|-----------------|-----|------|
| bAlternateSetting | 0 | 0 | 0x00 |
| bNumEndpoints | 3 | 3 | 0x03 |
| bInterfaceClass | Vendor-specific | 255 | 0xFF |
| bInterfaceSubClass | Vendor-specific | 255 | 0xFF |
| bInterfaceProtocol | Vendor-specific | 255 | 0xFF |
| iInterface | 0 | 0 | 0x00 |

Table 13: Configuration Descriptor\Interface 3\Endpoint Descriptor 1

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 4 IN ¹ | 132 | 0x84 |
| bmAttributes | Transfer-Types: INTERRUPT | 3 | 0x03 |
| wMaxPacketSize | 64 bytes | 64 | 0x0040 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹ Given endpoint address is an example and can be modified.

Table 14: Configuration Descriptor\Interface 3\Endpoint Descriptor 2

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| BLength | Valid | 7 | 0x07 |
| BDescriptorType | ENDPOINT | 5 | 0x05 |
| BEndpointAddress | 5 IN ¹ | 133 | 0x85 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹ Given endpoint address is an example and can be modified.

Table 15: Configuration Descriptor\Interface 3\Endpoint Descriptor 3

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 4 OUT ¹ | 4 | 0x04 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

2.1.7. Interface 4 (NDIS Port)

Table 16: Configuration Descriptor\Interface 4

| Name | Value | Dec | Hex |
|---------------------------|-----------------|-----|------|
| BLength | Valid | 9 | 0x09 |
| bDescriptorType | INTERFACE | 4 | 0x04 |
| bInterfaceNumber | 4 | 4 | 0x04 |
| bAlternateSetting | 0 | 0 | 0x00 |
| bNumEndpoints | 3 | 3 | 0x03 |
| bInterfaceClass | Vendor-specific | 255 | 0xFF |
| bInterfaceSubClass | Vendor-specific | 255 | 0xFF |
| bInterfaceProtocol | Vendor-specific | 255 | 0xFF |
| iInterface | 0 | 0 | 0x00 |

Table 17: Configuration Descriptor\Interface 4\Endpoint Descriptor 1

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 6 IN ¹ | 134 | 0x86 |
| bmAttributes | Transfer-Types: INTERRUPT | 3 | 0x03 |
| wMaxPacketSize | 64 bytes | 64 | 0x0040 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

Table 18: Configuration Descriptor\Interface 4\Endpoint Descriptor 2

| Name | Value | Dec | Hex |
|-------------------------|-------------------------------------|-----|--------|
| BLength | Valid | 7 | 0x07 |
| BDescriptorType | ENDPOINT | 5 | 0x05 |
| BEndpointAddress | 7 IN ¹ | 135 | 0x87 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

Table 19: Configuration Descriptor\Interface 4\Endpoint Descriptor 3

| Name | Value | Dec | Hex |
|-------------------------|----------------------|-----|------|
| bLength | Valid | 7 | 0x07 |
| bDescriptorType | ENDPOINT | 5 | 0x05 |
| bEndpointAddress | 5 OUT ¹ | 5 | 0x05 |
| bmAttributes | Transfer-Types: BULK | 2 | 0x02 |

| | | | |
|-----------------------|-------------------------------------|-----|--------|
| wMaxPacketSize | 512 bytes | 512 | 0x0200 |
| bInterval | At most one NAK each 32 microframes | 32 | 0x20 |

¹. Given endpoint address is an example and can be modified.

NOTE

The driver package does not support the NDIS interface temporarily.

3 System Integrating

When USB device is connected to the host system, the host system will load USB driver automatically. Therefore, it is strongly recommended that you integrate the USB driver into the development board when you create the WinCE system image.

3.1. Introduction of Driver Package

UC20 provides USB driver package for WinCE which includes BINARY and REG folders. qlril.dll is stored in BINARY folder. qlril.reg, matching with qlril.dll, is stored in REG folder. Please refer to the structure of driver package as below:

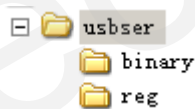


Figure 1: USB Driver Package Structure

3.2. Integrate USB Driver Files

The way of integrating WinCE USB driver mainly introduces how to add the driver BINARY and REG files in your WinCE system. When you start the integration, you should follow the steps as below:

3.2.1. Check System Component

Before integrating driver package for UC20, you should make sure that you have enabled USB Host Support under kernel option in your WinCE system because the running of USB driver depends on this system component.

3.2.2. Copy Files

Please copy the qlusbser.dll and qlusbser.reg in driver package to your BSP project folder, i.e. C:\WINCE600\PLATFORM\<TARGETBSP>\FILES.

NOTE

Please do remember to change <TARGETBSP> as your own BSP directory name.

3.2.3. Modify the Platform.reg

Add the line below at the end of platform.reg.

```
#include "$(_PLATFORMROOT)\<TARGETBSP>\FILES\qlusbser.reg"
```

3.2.4. Modify the Platform.bib

1. For WinCE6.0, add the line below:

```
MODULES
..
..
qlusbser.dll      $(_PLATFORMROOT)\(TARGETBSP)\FILES\qlusbser.dll    NK      SHK
..
..
```

2. For WinCE5.0, add the line below:

```
MODULES
..
..
qlusbser.dll      $(_PLATFORMROOT)\(TARGETBSP)\FILES\qlusbser.dll    NK      SHC
..
..
```

3.2.5. Rebuild and Create System Image

After you have done the four steps above, you need to execute "clean sysgen" command to rebuild your project and create the new system image.

3.3. USB COM Port

Download the new system image to your target board and reboot your WinCE system. For the newly installed system, USB driver will be loaded when you connect UC20 module to the board with USB port.

After the USB driver has been loaded, the driver will register three COM devices to the system device manager. The index of default COM devices which are defined in the qlusbser.reg lists as below:

- COM5
- COM6
- COM7
- COM8

You can use serial port tool to check whether these COM ports are created or not. And the corresponding relations between interface and device name below which had been set in the default REG files are shown as below:

Table 20: Relationship between Interfaces and COM Devices

| INDEX | Interface Name | Device Name |
|-------|-----------------|-------------|
| 0 | DM Interface | COM5 |
| 1 | NMEA Interface | COM6 |
| 2 | AT Interface | COM7 |
| 3 | Modem Interface | COM8 |
| 4 | NDIS Interface | NONE |

NOTES

The index for interface is defined in REG files in the driver package. You must modify the index as your free COM index of your WinCE board.

4 COM Port for Application

For WinCE system, you can send AT commands with USB AT Port and set up the PPP connection with USB Modem Port. In this way, you can enjoy the VOICE CALL or SMS over USB AT Port and surf the Internet over the USB Modem Port.

4.1. Testing AT Commands on AT Port

Open the USB AT Port with the serial debugging tool and send “AT\r\n” to the COM port. If the tool receives the result code “OK”, it proves that the UC20 module is available for system.

4.2. Create PPP Connection on Modem Port

In WinCE, you can make a new PPP connection on “Network and Dial-up Connections” system options. It is simple to set up PPP dial-up over our UC20 module via the new PPP connection. After PPP dial-up connection is established, you can enjoy surfing the Internet. The method of creating PPP connection is given as below:

1. Open and Enter “Control Panel”.

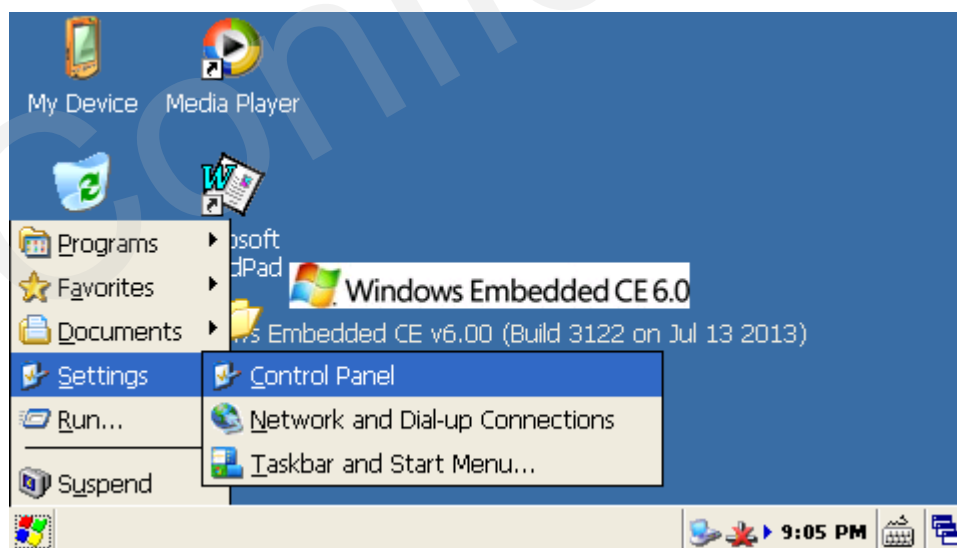


Figure 2: Open Control Panel

2. Double Click “Network and Dial-up Connections”



Figure 3: Click Network and Dial-up Connections

3. Enter below Window Interface:

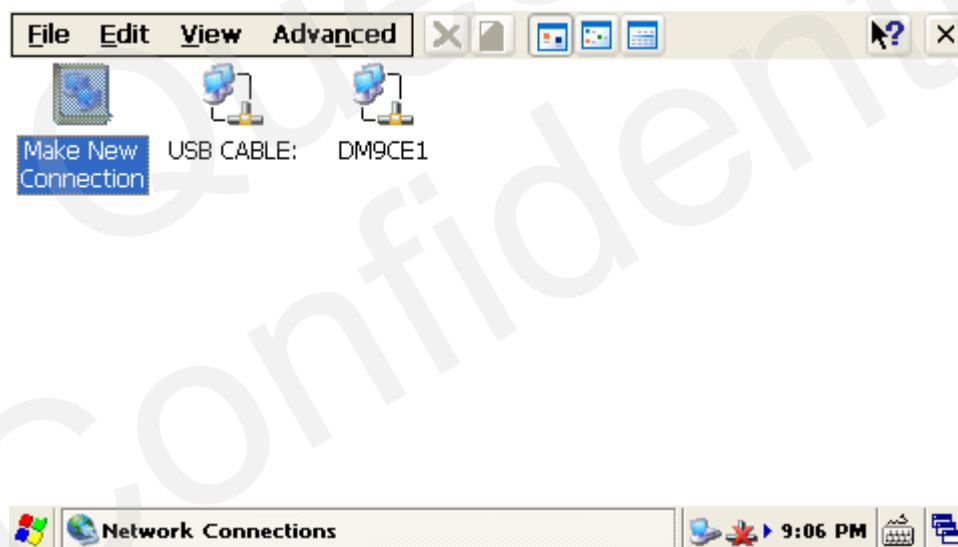


Figure 4: Click Make New Connection

4. Double click the “Make New Connection” and the pop-up box is displayed:

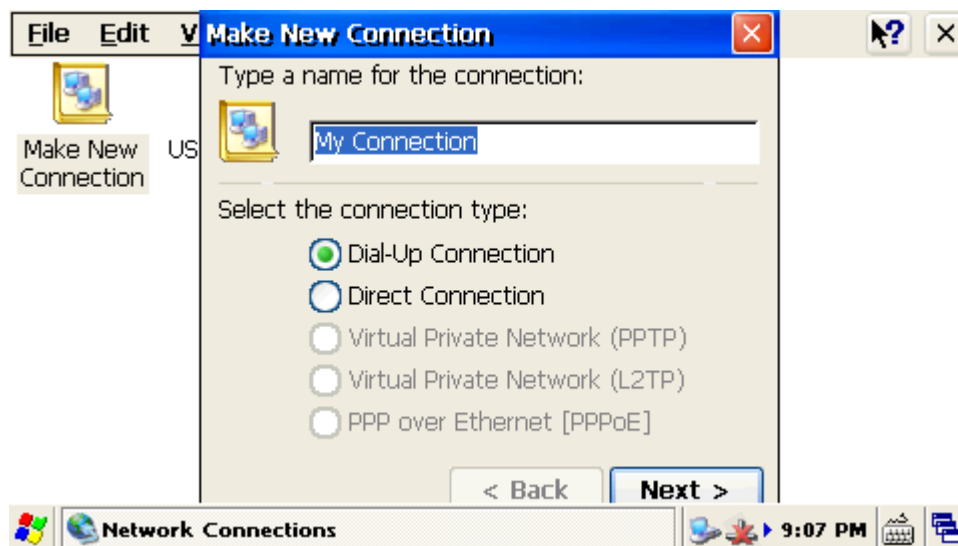


Figure 5: Make New Connection Interface

5. Click the "Next" button:

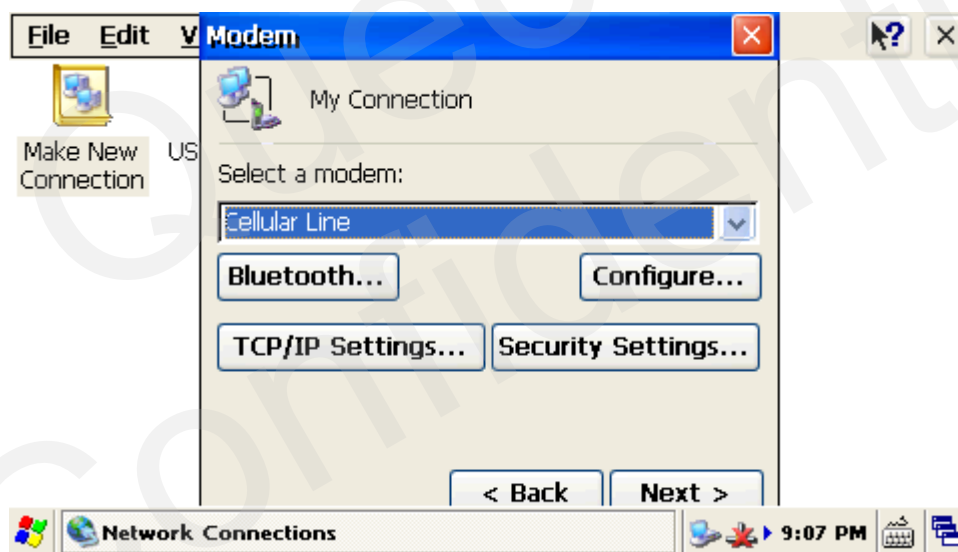


Figure 6: Modem Selection Interface

6. Select the "Quectel UC20 HSUSB Modem" on the pull-down menu:

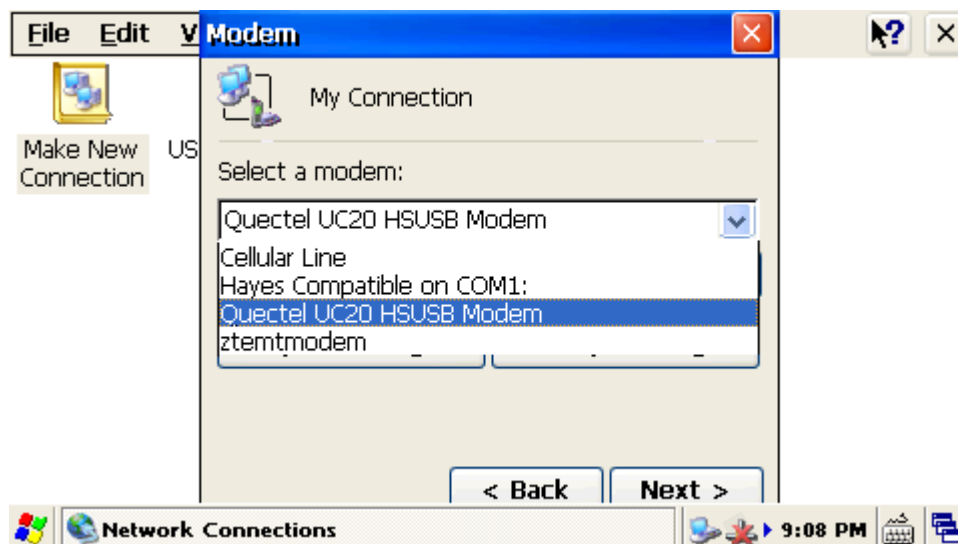


Figure 7: Select “Quectel UC20 HSUSB Modem”

7. Click the “Configure” button:



Figure 8: Connection Properties

8. Select “Call Options” and configure the APN in the “Extra Settings” text-box. For example, you can use SIM Card of CHINA-UNICOM, and you need to set “3gnet” for CHINA-UNICOM to your APN:

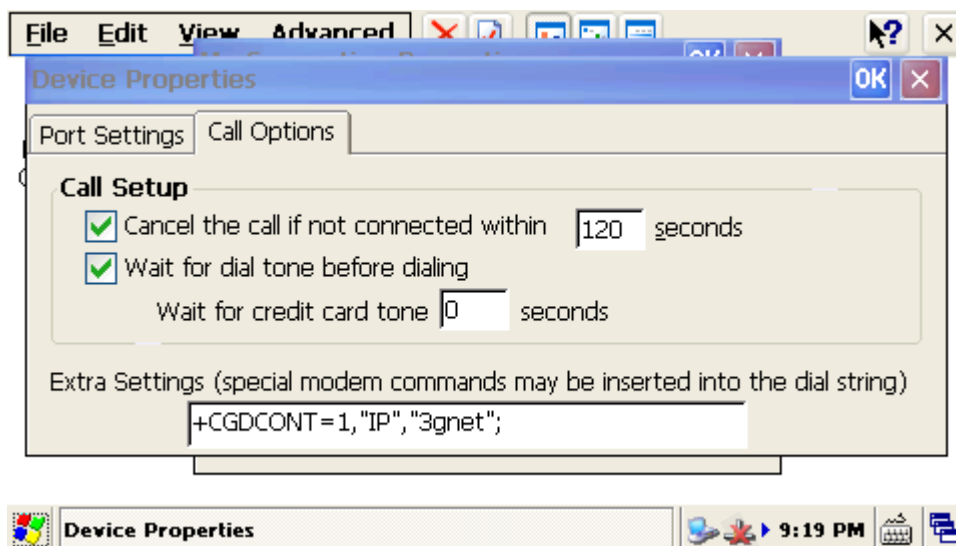


Figure 9: Set APN

9. Click "OK" and click "Next" button. It goes to the interface of phone number configuration. Then, fill in "*99#" the Phone number text-box:

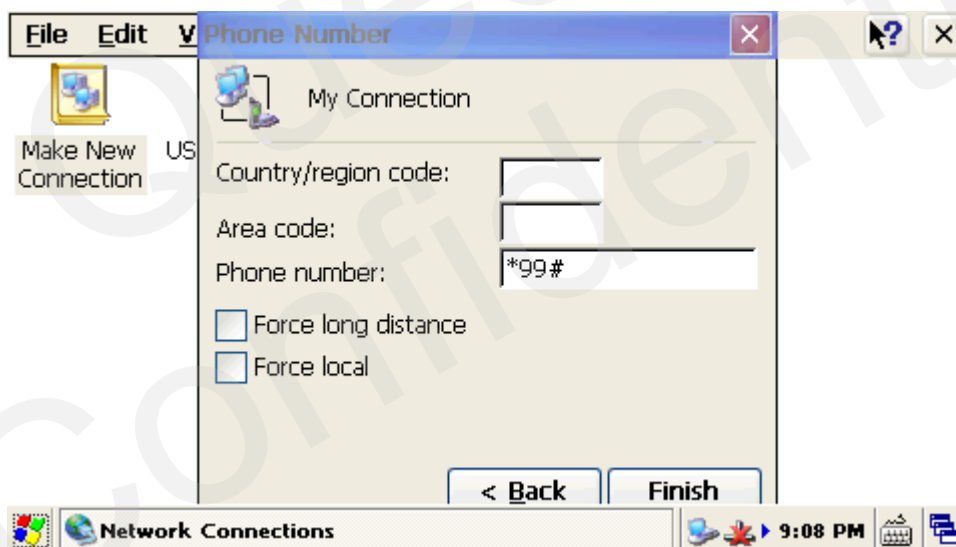


Figure 10: Set Phone number

10. Click "Finish" button and a new icon named "My Connection" will appear in this window:

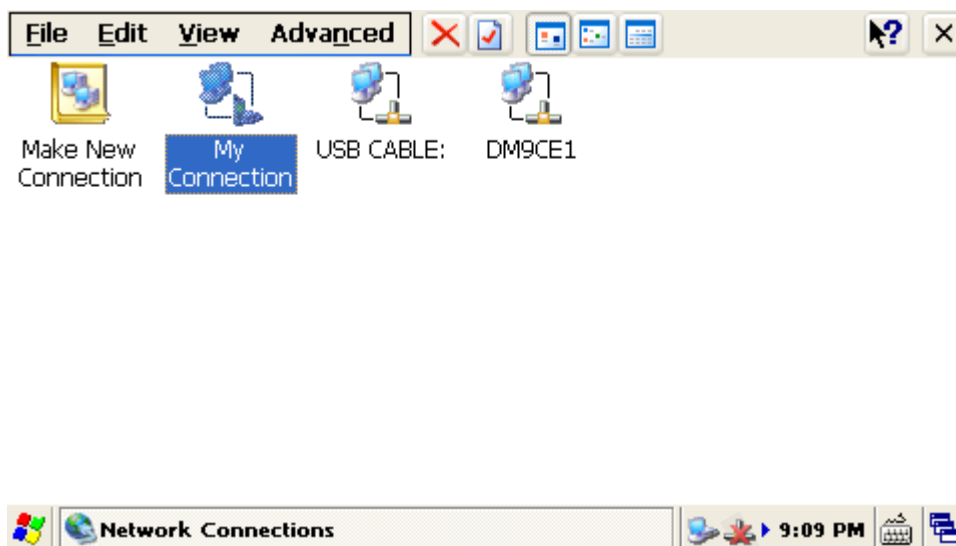


Figure 11: Add New Connection OK

11. Double click “My Connection” icon:

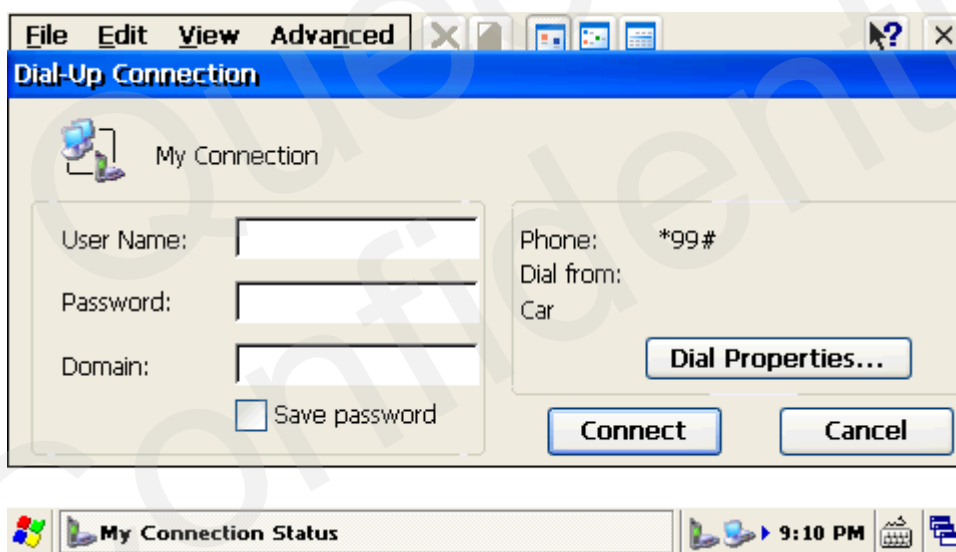


Figure 12: Dial-up Connection Interface

12. Fill in user name and password of PPP dial-up in the text-box. It uses “wap” as User Name and Password here:

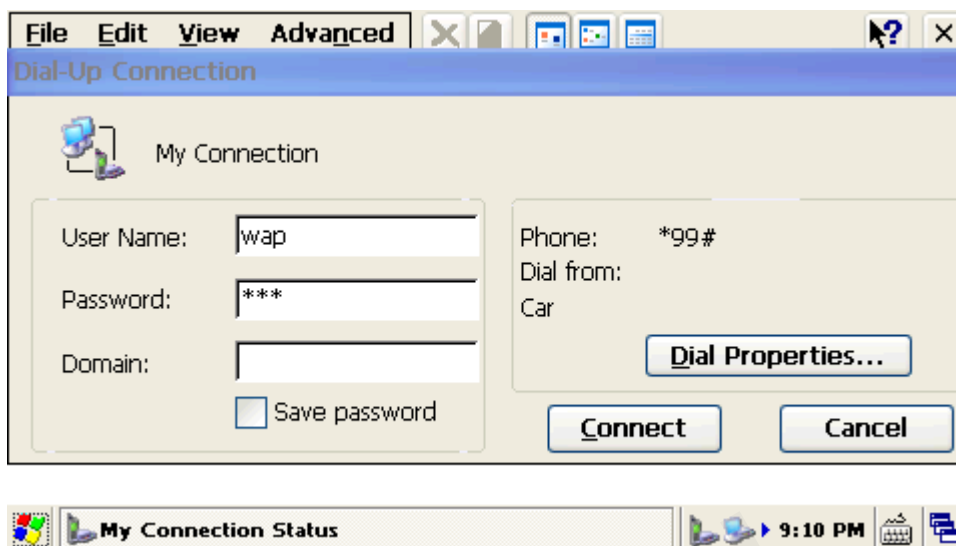


Figure 13: Add User Name and Password

13. Click "Connect" button to set up the PPP connection:

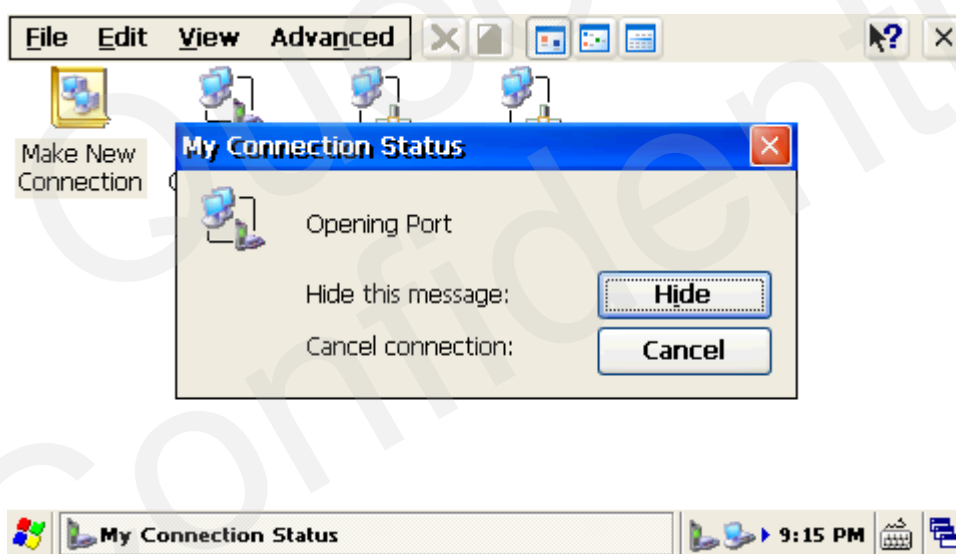


Figure 14: Begin to set-up PPP connection

14. Usually, the PPP dial-up will be connected in several seconds:

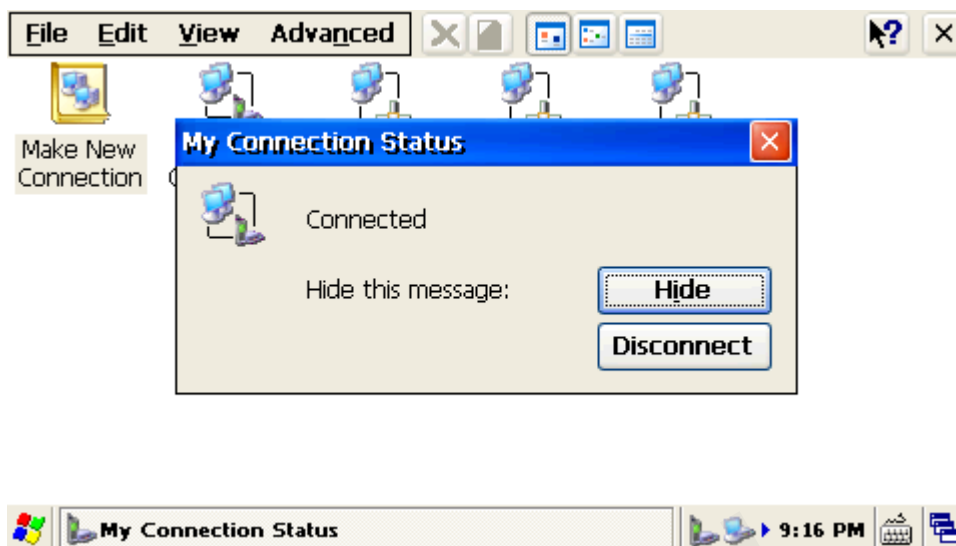


Figure 15: PPP Connected

15. Finally, you can surf the Internet on your WinCE board:



Figure 16: Surf the Internet

5 Appendix A Reference

Table 21: Terms and Abbreviations

| Abbreviation | Description |
|--------------|-----------------------|
| OS | Operating System |
| USB | Universal Serial Bus |
| BSP | Board Support Package |