Qualcomm
TECHNOLOGIES, INC

RM Network (RmNet) Interface

80-VT270-1 C

December 23, 2013

Submit technical questions at:
https://support.cdmatech.com/

Confidential and Proprietary – Qualcomm Technologies, Inc.

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Contents

Figures

Tables

Revision history

Revision	Date	Description
A	Jul 2009	Initial release
B	Mar 2010	Added IPv6 support
C	Dec 2013	Updated document title

1 Introduction

1.1 Purpose

This document describes the Rm Network Device (RmNet), which presents a network interface to the TE attached to MSM™ devices, providing it IP connectivity. It essentially provides an NIC-like interface.

1.2 Scope

This document is intended for internal distribution to engineering and product marketing teams involved in the development, integration, and deployment of the RmNet feature, and also for external distribution to connection manager, network driver developers on hosts and PDAs, and other AMSS licensees and wireless carriers.

1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font, e.g., `#include`.

Code variables appear in angle brackets, e.g., `<number>`.

Commands to be entered appear in a different font, e.g., **`copy a:*.* b:`**.

Shading indicates content that has been added or changed in this revision of the document.

1.4 References

Reference documents are listed in Table 1-1. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers may not be sequential.

Table 1-1 Reference documents and standards

Ref.	Document	
Qualcomm Technologies		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Qualcomm MSM™ Interface (QMI) Architecture*	80-VB816-1
Q3	*QMI Wireless Data Service Major Version 1, Minor Version 4 (ISOD)*	80-VB816-5
Q4	*Qualcomm Interface Quality of Service (QMI QoS) Feature Description Document*	80-VF536-2
Q5	*QMI Control Service (QMI_CTL)*	80-VB816-3
Q6	*Qualcomm USB Network Driver for Windows XP User Guide*	80-VB717-1

1.5 Technical assistance

For assistance or clarification on information in this document, submit a case to Qualcomm Technologies, Inc. (QTI) at https://support.cdmatech.com/.

If you do not have access to the CDMATech Support Service website, register for access or send email to support.cdmatech@qti.qualcomm.com.

1.6 Acronyms

For definitions of terms and abbreviations, see [Q1].

2 Overview

The RmNet feature in QTI MSM devices emulates a network interface for the connected TE. This allows for the MSM to behave as a network adapter when attached to a TE and provide packet data connectivity to the attached TE. The term TE is inclusive of all form factors, including devices such as PCs, notebooks, or apps processor.

This NET (i.e., RmNet) device can coexist with other logical devices (e.g., modem, diagnostic, NMEA, etc.).

RmNet relies on a control interface for any control signaling between the TE and MS, e.g., to initiate a data session on demand, send any notifications, etc. Currently, RmNet supports only the Qualcomm MSM Interface (QMI) as the control channel for signaling between the TE and MS. QMI defines the framework and messages for communication between the applications/drivers on the TE and the MS. See [Q2] for details on QMI.

2.1 Supported features

Supported features are:

- IPv4 data connectivity
- Autoconnect mode
- Multi-RmNet
- Raw IP mode (not supported on all products)
- QoS (not supported on all products)
- Interconnects supported – USB and shared memory
- IPv6 data connectivity

3 RmNet Architecture

Figure 3-1 illustrates the various modules involved in the RmNet control path and data path. Currently, RmNet supports only QMI as the control channel. QMI messages are used for control signaling between the TE and the MS. See [Q2] for more details.

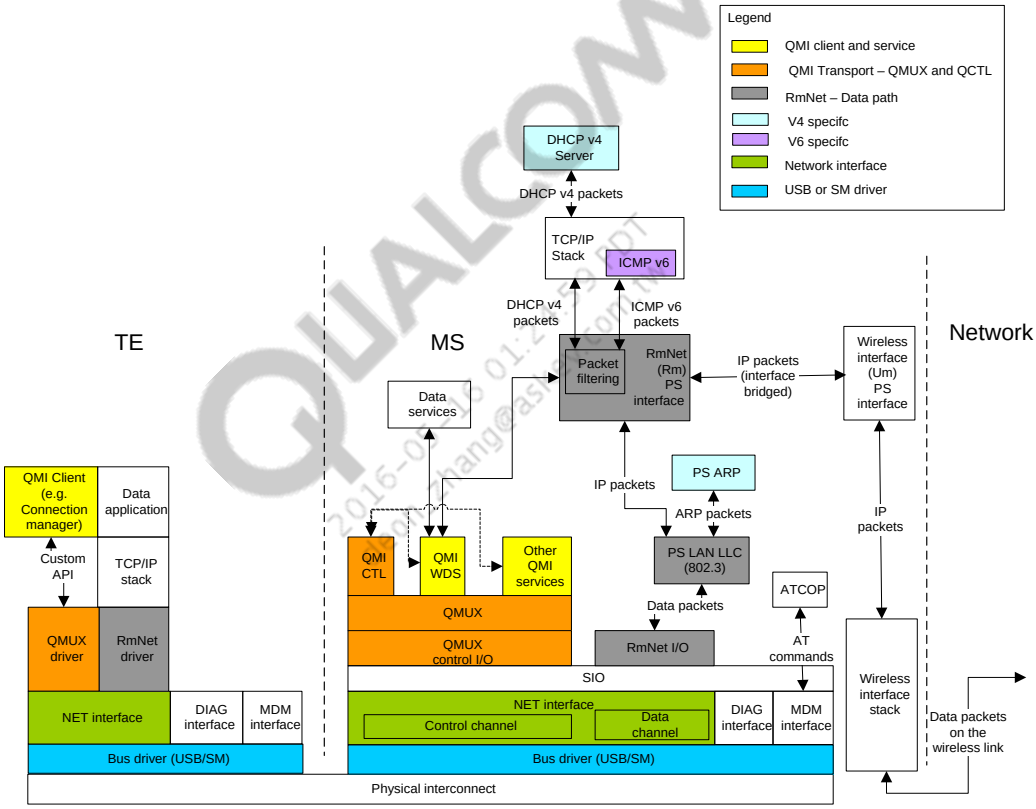

Figure 3-1 RmNet architecture

In the data path, RmNet does not use PPP and hence eliminates the HDLC framing/unframing overhead associated with PPP on the link between the TE and the MS. By default, RmNet uses 802.3 framing. RmNet also supports transferring Raw IP packets (see Section 6.2). When Raw IP mode is used, the LAN LLC and ARP modules are not used.

DHCP v4 server and PS ARP modules are only used for IPv4. The DHCP v4 server is used if the TE is using DHCP for IP address configuration on the TE (see Section 4.2).

ICMPv6 module is used only for IPv6, for neighbor discovery, DAD, and stateless address autoconfiguration.

A single RmNet interface (instance) can support only one of IPv4 or IPv6 at a time. To have both IPv4 and IPv6 data sessions concurrently, each must go over a separate RmNet interface (see Chapter 7 for multiple RmNet instance support).

4 RmNet Data Session Management

This chapter discusses the various scenarios involving data session setup and teardown for RmNet as follows:

- Data session establishment – Describes the steps involved when a TE requests the MS to set up a data session
- TE IP configuration – Describes the procedure used to assign an IP address and other IP configuration to the TE; there are two possible methods:
  - Using standard protocols
    - DHCP – TE uses DHCP procedure to get the IP configuration (for IPv4)
    - Stateless Address Autoconfiguration using ICMPv6 (for IPv6)
  - Using QMI – TE uses QMI messages to get the IP configuration (only for IPv4)
- Data session teardown – Describes the steps involved when a TE requests the MS to tear down a data session
- Autoconnect mode – Describes the Autoconnect mode of operation where a data session is set up or torn down automatically, without an explicit request from the TE
- IP address change – Describes the procedure to notify and convey the new IP configuration to the TE

Components involved in data session establishment/teardown procedures are:

- TE OS stack – The native OS data stack running on the TE
- QMI WDS client(s) – Client(s) of the QMI Wireless Data Service (see [Q3]); sends and receives WDS messages; may be responsible for notifying the TE OS stack of data connectivity and IP configuration using host OS-specific APIs
- QMUX driver – QMUX provides the transport for QMI clients and services (see [Q2]). The client sends and receives QMI messages via an API exposed by the QMUX driver running on the TE.
- MS – The mobile station where the QMI services reside
- Network – The wireless network

4.1 RmNet data session setup

Pre- and post-conditions for RmNet data session are:

- Precondition – The QMI WDS client has successfully obtained a client ID from the MS.
- Postcondition – The MS has obtained the IP configuration from the network.

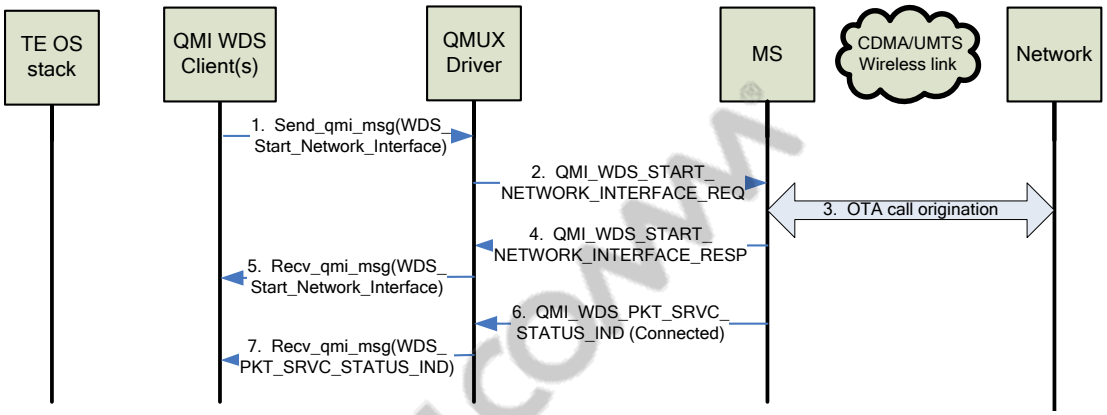

Figure 4-1 RmNet data session setup

The following steps correspond to Figure 4-1:

1. The client on the TE sends a request to start the RmNet data session.
2. The QMUX driver adds the QMUX header to the request before sending it to the MS.
3. On receiving this request, the MS initiates the procedure to bring up the data session OTA.
4. Once the OTA data session is up, the MS sends a response indicating success destined to the client which initiated the request. In case of call origination failure, the MS sends a failure response and the below steps do not occur.
5. The QMUX driver removes the QMUX header and passes the response message to the appropriate client.

NOTE: At this point, a successful response means only that the data session originated successfully over the wireless network. The IP address configuration still must be completed on the TE (see Section 4.2) before data transfer can begin.

6. The MS also sends a broadcast indication to notify that the data session is now connected.
7. The QMUX driver on the TE forwards it to the QMI WDS clients.

4.2 TE IP configuration

4.2.1 Using DHCP for IPv4 address

Pre- and post-conditions for using DHCPv4 are:

- Precondition – The data session must be set up prior to this.
- Postcondition – The TE IP configuration is complete and data packets can be transferred.

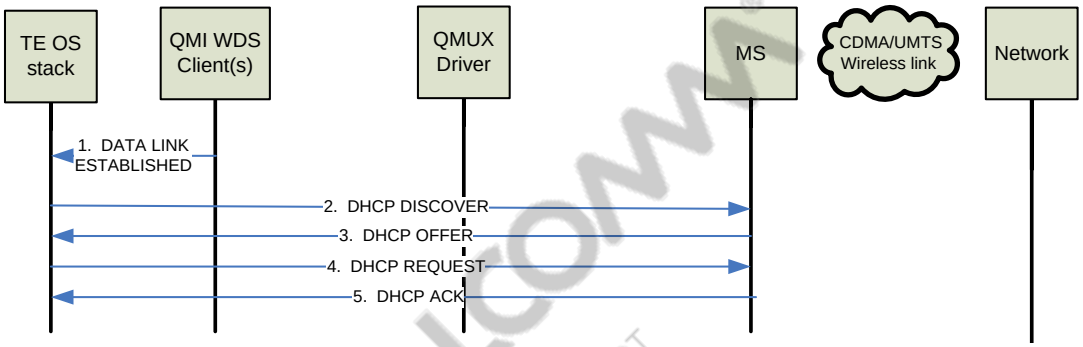

Figure 4-2 TE Ipv4 address configuration using DHCP

The following steps correspond to Figure 4-2:

1. The WDS client notifies the OS running on the TE that the data link is established.

2 to 5. DHCP negotiation occurs between the TE and the MSM to assign an IP address and other IP configuration to the TE. The IP address assigned to the TE is the same as the one assigned to the MS by the network.

4.2.2 Using stateless address autoconfiguration for IPv6 address

Pre- and post-conditions are:

- Precondition – The data session must be set up prior to this.
- Postcondition – The TE IP configuration is complete and data packets can be transferred.

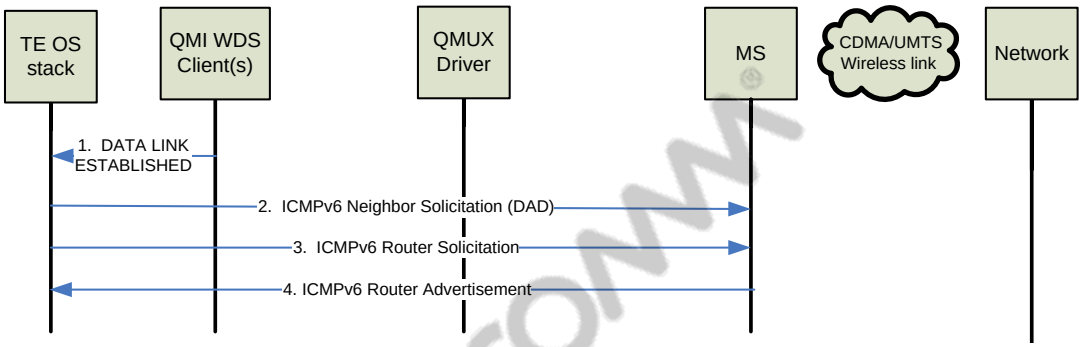

Figure 4-3 TE IPv6 address configuration using stateless autoconfiguration

The following steps correspond to Figure 4-3:

1. The WDS client notifies the OS running on the TE that the data link is established.

2 to 4. The TE must do DAD and stateless address autoconfiguration to configure the address.

NOTE: TE must do DAD for each IID it generates.

4.2.3 Using QMI (only IPv4)

Pre- and post-conditions for using QMI are:

- Precondition – The data session must be set up prior to this.
- Postcondition – The TE IP configuration is complete and data packets can be transferred.

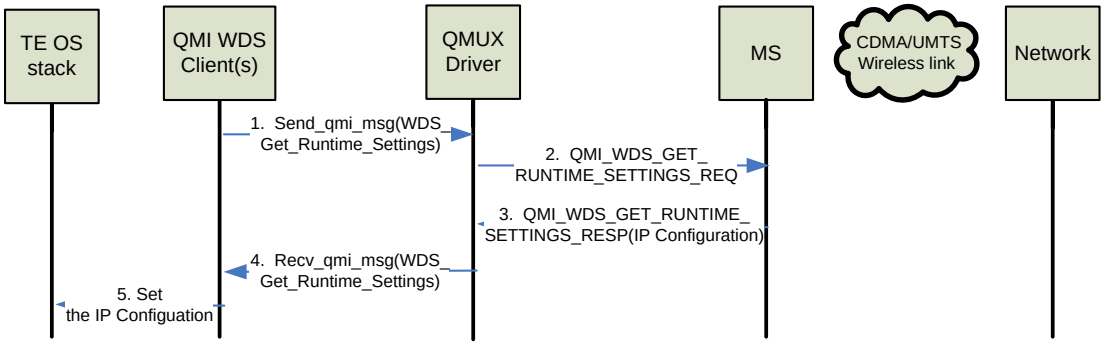

Figure 4-4 TE IPv4 address configuration using QMI

The following steps correspond to Figure 4-4:

1 to 2. The WDS client sends a request to the MS to query the IP configuration to be set on the TE.

3 to 4. The MS responds with the IP address and other configuration. The IP address assigned to the TE is the same as the one assigned by the network.

5. The client then configures that IP address and other configuration parameters on the TE using host-specific APIs.

4.2.4 RmNet guard timer

When the data session is set up with the wireless network, a guard timer is started. The TE is expected to query its IP configuration (as shown in Sections 4.2.1 to 4.2.3) before the expiry of this timer. If the timer expires before the TE queries the IP configuration from the device, the data session is torn down. This is done to save the resources of the wireless network. The value of the guard timer is currently set to 2 min.

4.3 RmNet data session teardown

Pre- and post-conditions for tearing down the RmNet data session are:

- Precondition – The data session had been set up earlier.
- Postcondition – The data session is torn down and the TE loses IP connectivity.

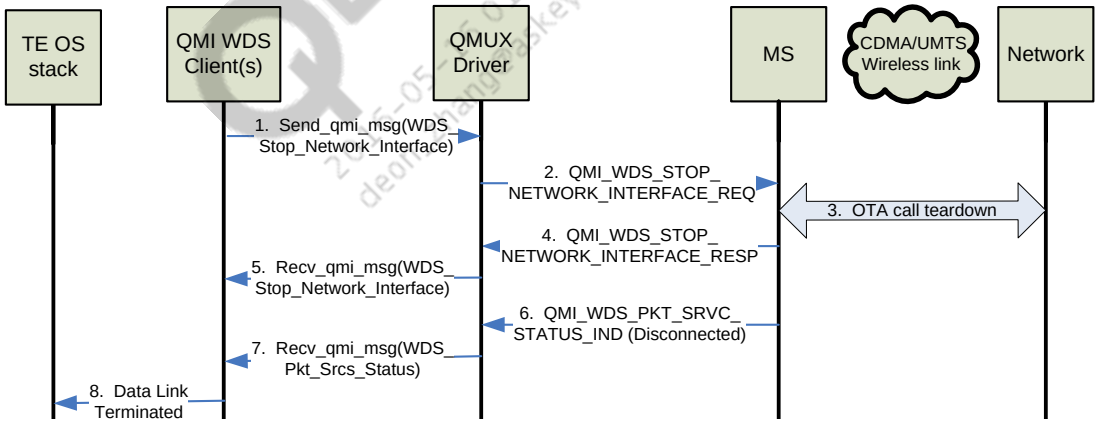

Figure 4-5 Data session teardown

The following steps correspond to Figure 4-5:

1. The client sends a request to stop the RmNet data session.
2. The QMUX driver adds the QMUX header to the message and sends the request to the MS.
3. The MS initiates the call teardown OTA.
4. The MS sends a SUCCESS response destined for the client.
5. The QMUX driver removes the QMUX header and passes the response to the appropriate client.
6. The MS also sends a broadcast indication to notify that the data session is disconnected.

7. The QMUX driver on the TE forwards it to the WDS client.
8. The WDS client will notify the OS running on the TE that the data link was terminated.

4.4 Autoconnect mode

The above sections illustrate the mechanism to manually bring up and tear down a data session. To support Always On operation, RmNet has an Autoconnect mode that, when enabled, brings up the data session automatically when the device is connected to the TE, and tears it down automatically when the device is disconnected from the TE. Autoconnect is available only for RmNet over plug-and-play interconnects such as USB.

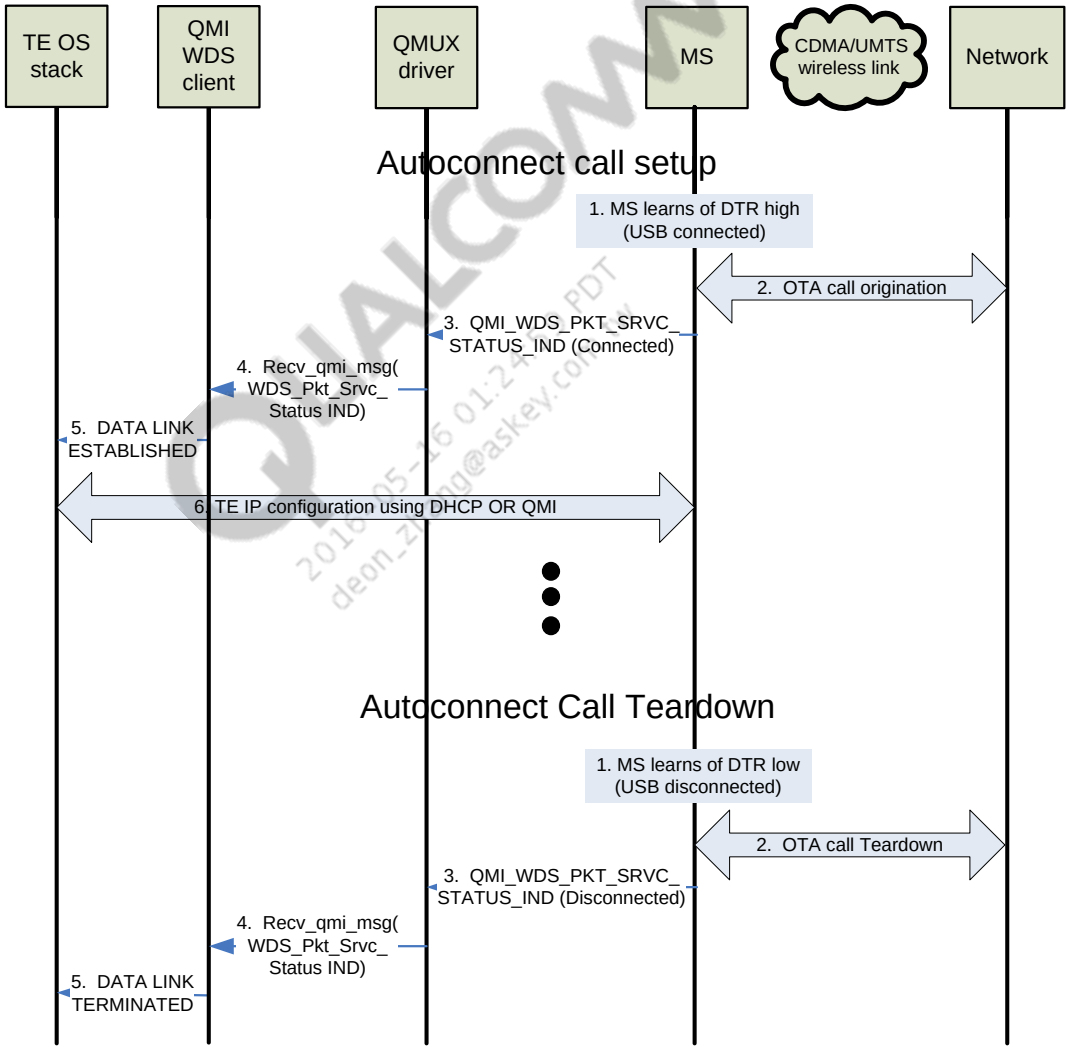

Figure 4-6 Autoconnect mode

The following steps correspond to Figure 4-6.

For Autoconnect data session setup:

1. The MS learns that the USB is connected, e.g., through DTR high.
2. The MS brings up the data session OTA.
3. The MS sends a broadcast QMI WDS indication to notify that the data session is connected.
4. The QMUX driver on the TE forwards it to the WDS client.
5. The WDS client then notifies the OS running on the TE that the data link is established.
6. The IP configuration of the TE is done using DHCP or QMI, as shown in Section 4.2.

For Autoconnect data session teardown:

1. The MS learns that the USB is disconnected, e.g., through DTR low.
2. The MS tears down the data session OTA.
3. The MS sends a broadcast QMI WDS indication to notify that the data session is disconnected.
4. The QMUX driver on the TE forwards it to the WDS client.
5. The WDS client then notifies the OS running on the TE that data connectivity is terminated.

During Autoconnect mode operation, if the data session becomes disconnected for reasons other than USB disconnection, e.g., loss of signal, etc., the RmNet device tries to reestablish the data session using a backoff mechanism. RmNet starts with a timer of 1 sec. Every time a data session establishment attempt fails (rejected by the lower layers of the device or the network), it backs off the timer by a factor of two and then the MS tries again when the timer expires. The timer backoff continues up to a maximum of 2 min, after which RmNet retries every 2 min. During this process, if any attempt is successful, the timer is reset back to the original value of 1 sec.

4.5 IP address change (v4 or v6)

Pre- and post-conditions for changing an IP address are:

- Precondition – The data session is up and the IP address on the TE has been configured.
- Postcondition – The TE was notified of an IP address change and the TE IP address configuration is done again.

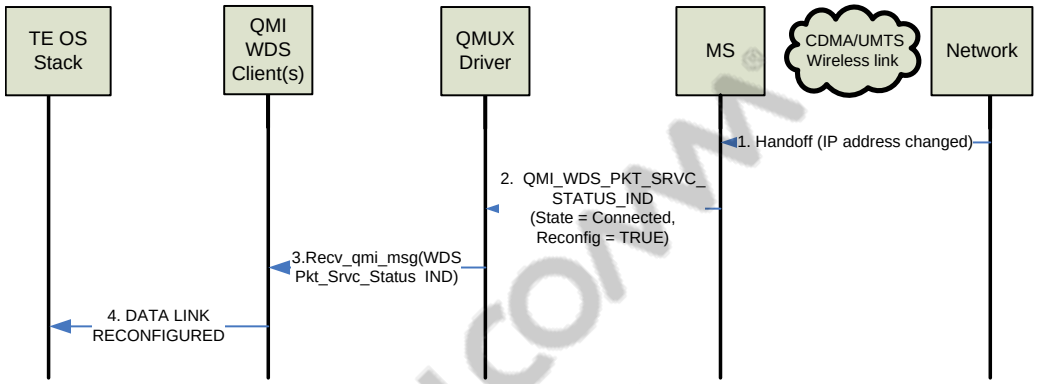

Figure 4-7 IP address change

To change an IP address:

1. Handoff takes place, causing the IP address of the MS to change.
2. The MS sends a broadcast WDS indication to notify that the data session is connected, but reconfiguration is required.
3. The QMUX driver on the TE forwards it to the QMI WDS client.
4. The WDS client notifies the OS running on the TE that the data link IP configuration must be reconfigured.

5 Interconnect Requirements

Currently, RmNet is supported over USB and shared memory interconnects. One or more NET devices (RmNet logical devices) may be supported.

Each logical device must have a separate data and control channel. Channel independence implies that each channel must act as if there were no physical coupling between the communication channels, including (but not limited to) separate Tx and Rx path queuing, independent flow control mechanisms, and independent data transmission scheduling. The interconnection must provide for framing of data packets exchanged, i.e., delineating packet boundaries to the transport protocol, e.g., 802.3.

To support multiple logical devices, the underlying interconnect must provide for a mechanism for multiplexing multiple logical devices over a single physical connection.

6 RmNet Frame Format

This chapter discusses the frame formats supported for the data packets being transmitted over the RmNet interconnect (USB, shared memory).

6.1 802.3 data link layer

The data link layer format is illustrated in Figure 6-1. In this format, IP packets are sent within 802.3 frames. This is the data format used by default.

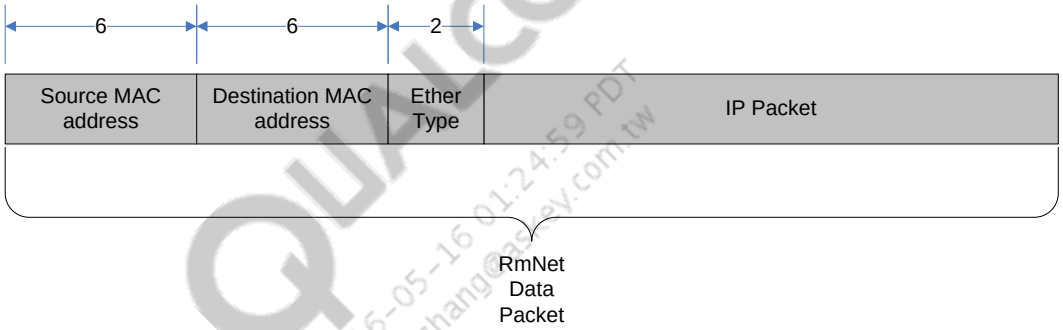

Figure 6-1 802.3 frames

6.2 Raw IP

In this format, raw IP packets are exchanged without any 802.3 header. The QMUX driver on the TE needs to send a QMI_CTL request to the MS to use the raw IP data format. This must be done during power-up initialization before any data session is set up. The TE should assume raw IP data format only after the MS sends a response indicating that Raw IP mode can be used; otherwise, the default is 802.3 mode. See [Q5] for more details.

This raw IP data format is illustrated in Figure 6-2.

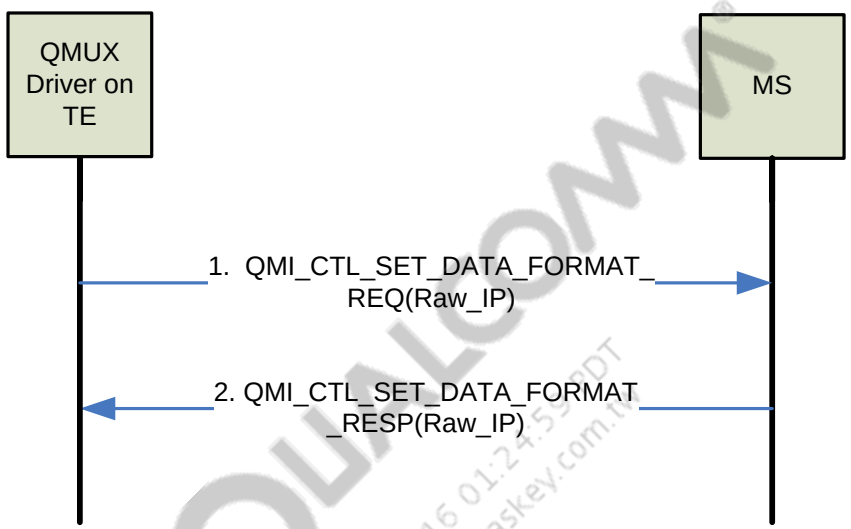

Figure 6-2 Set raw IP data format

6.3 QoS flow header

The QoS flow header data format, illustrated in Figure 6-3, is to be used only when the QMI QoS feature, which provides access to QoS over the wireless network to the applications on the TE, must be used. In this data format, the QoS flow header is added in front of the data packet (802.3 frame or raw IP packet). See [Q4] for more details on QMI QoS.

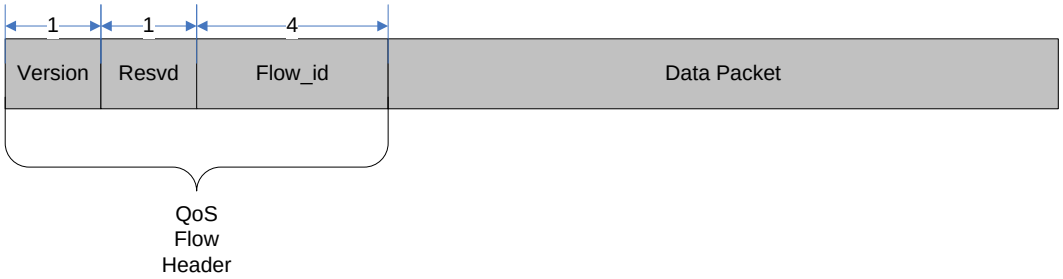

Figure 6-3 QoS flow header

The client side informs the MS about this data format using a QMI_CTL message, as shown in Figure 6-4. This must be done during power-up initialization before any data session is set up. The TE should assume this data format only after it receives a successful response from the MS. See [Q5] for more details.

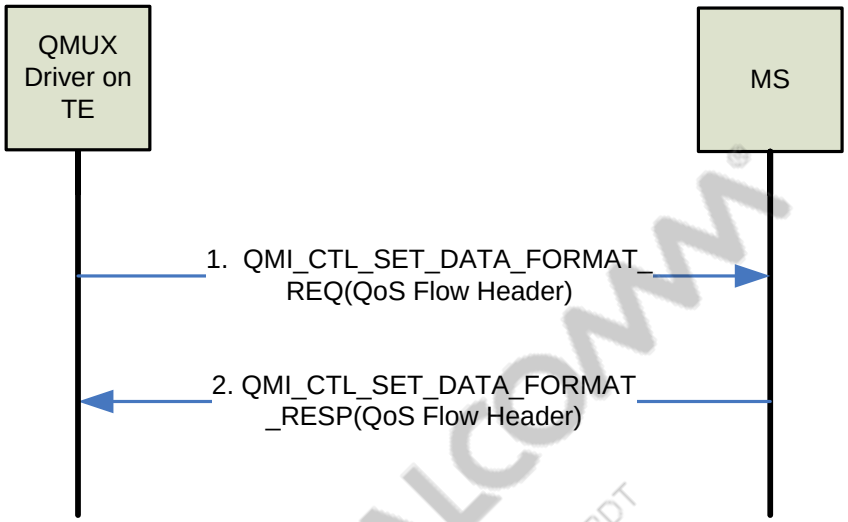

Figure 6-4 Set QoS flow header data format

7 Multi-RmNet

Multi-RmNet refers to supporting multiple logical devices, i.e., NET interfaces. This feature is used to support multiple IP data sessions (each can be either IPv4 or IPv6). A maximum of three RmNet instances can be used simultaneously.

Each RmNet logical device appears as an independent network adapter on the TE, capable of getting its own IP address and transferring data independently Figure 7-1 illustrates this for two RmNet instances. Each instance operates on a separate port and is independent of the other instance.

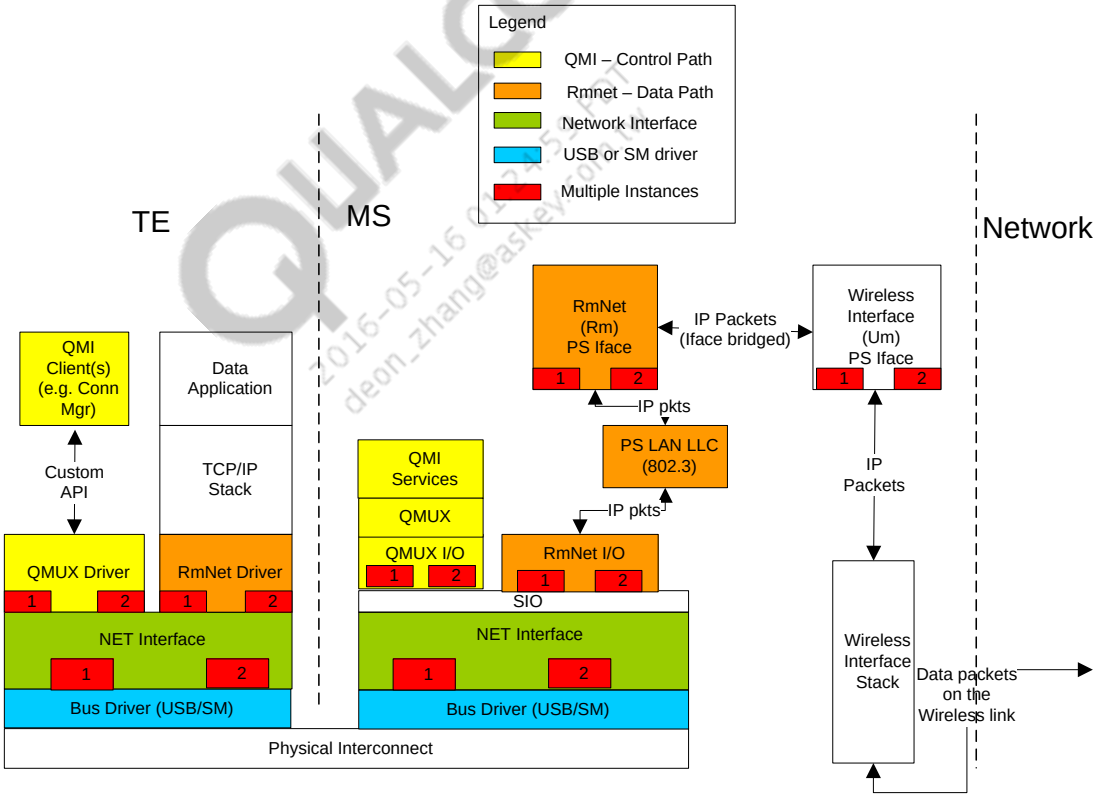

Figure 7-1 Multi-RmNet

8 RmNet Configuration Parameters

Table 8-1 lists the RmNet configuration parameters that are supported. These are applicable only for RmNet over USB.

Table 8-1 RmNet configuration parameters

Configuration parameter	Description	Default value
USB_CURRENT_DEVICE (FS-USB) – NV item 2782 HS_USB_CURRENT_COMPOSITION (HS_USB) - NV item 4526	Determines whether RmNet is part of the USB composition; must be set to a composition value that includes RmNet	0 (RmNet not present)
RMNET_AUTOCONNECT – NV item 3534	Determines whether RmNet Autoconnect is enabled; applicable only for RmNet over USB; consists of two components. an index and an enable/disable setting; the index is used to indicate RmNet instance (currently only one instance, i.e., index 0 is supported); second component can be set to 1 or 0 to enable or disable autoconnect on that particular index	{0, 0} (Autoconnect disabled)