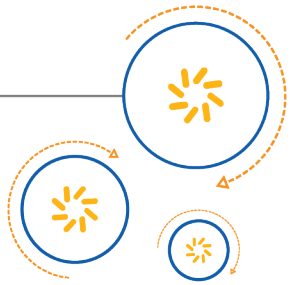
Qualcomm Technologies, Inc.



QMI UIMRMT 1.2 for MPSS.JO.1.2

QMI User Identity Module Remote Spec

80-NV304-56 A

February 3, 2016

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

Revision History

Revision	Date	Description
A	Feb 2016	Initial release. Created from 80-NH952-56 AC. Updates for this revision include version minor version 1 and minor version 2. Updated: • Optional Error Cause for Card Error Event TLV (Section 3.4.1) • Sections 3.5.3, 3.8.2, and 3.9.2 • QMI_UIMRMT Fundamental Concepts (Section 2.4) • Related Documents (Section A.1) Added new TLVs: • Response Timeout (Section 3.8.1) • Voltage Class (Section 3.8.1) • Power-down Mode (Section 3.9.1)

Contents

List of Tables

1 Introduction

1.1 Purpose

This specification documents Major Version 1 of the Qualcomm Messaging Interface (QMI) for UIM Remote (QMI_UIMRMT).

The QMI_UIMRMT service allows modem access to a UIM that is not directly connected to the modem, but is accessible via the QMI interface.

1.2 Scope

This document is intended for QMI clients that connect a Qualcomm MSM™ device with a UIM via the QMI_UIMRMT.

This document provides the following details about QMI_UIMRMT:

- Theory of operation – Chapter 2 provides the theory of operation of QMI_UIMRMT. The chapter includes messaging conventions, assigned QMI service type, fundamental service concepts, and state variables related to the service.
- Message formats, syntax, and semantics – Chapter 3 provides the specific syntax and semantics of messages included in this version of the QMI_UIMRMT specification.

1.3 Conventions

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

1.4 Technical Assistance

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

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

2 Theory of Operation

2.1 Generalized QMI Service Compliance

The QMI_UIMRMT service complies with the generalized QMI service specification, including the rules for messages, indications and responses, byte ordering, arbitration, constants, result, and error code values described in 80-VB816-1. Extensions to the generalized QMI service theory of operation are noted in subsequent sections of this chapter.

2.2 UIMRMT Service Type

UIMRMT is assigned QMI service type 0x32.

2.3 Message Definition Template

2.3.1 Response Message Result TLV

This Type-Length-Value (TLV) is present in all Response messages defined in this document. It is not present in the Indication messages.

Name	Version introduced	Version last modified
Result Code	Corresponding response's *Version introduced*	Corresponding response's *Version last modified*

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x02			1	Result Code
Length	4			2	
Value	→	uint16	qmi_result	2	Result code • QMI_RESULT_SUCCESS • QMI_RESULT_FAILURE
		uint16	qmi_error	2	Error code – Possible error code values are described in the error codes section of each message definition

2.4 QMI_UIMRMT Fundamental Concepts

The QMI_UIMRMT service enables a Qualcomm MSM device to access a remote UIM by providing the following features:

- Ability to receive card events and Answer to Reset (ATR) of the card
- Card power-up, power-down, and warm reset
- Raw Application Protocol Data Unit (APDU) transmission to the card

A control point can register to make a connection and card available to the QMI_UIMRMT service, which in turn provides notifications to the modem. For USB-UICC support, refer to 80-NN611-1.

For Bluetooth SAP (BT-SAP) client mode support, refer to 80-NC254-75.

2.5 Service State Variables

2.5.1 Shared State Variables

No QMI_UIMRMT state variables are shared across control points.

3 QMI_UIM_REMOTE Messages

Table 3-1 QMI_UIM_REMOTE messages

Command	ID	Description
QMI_UIM_REMOTE_GET_SUPPORTED_MSGS	0x001E	Queries the set of messages implemented by the currently running software.
QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS	0x001F	Queries the fields supported for a single command as implemented by the currently running software.
QMI_UIM_REMOTE_RESET	0x0020	Resets the service state variables of the requesting control point.
QMI_UIM_REMOTE_EVENT	0x0021	Notifies the service of remote UIM events.
QMI_UIM_REMOTE_APDU	0x0022	Exchanges the APDU with the remote card.
QMI_UIM_REMOTE_APDU_IND	0x0022 indication	Indication to the control point to transmit an APDU to the card.
QMI_UIM_REMOTE_CONNECT_IND	0x0023	Indication to the control point to establish a connection with the card.
QMI_UIM_REMOTE_DISCONNECT_IND	0x0024	Indication to the control point to tear down the connection with the card.
QMI_UIM_REMOTE_CARD_POWER_UP_IND	0x0025	Indication to the control point to power up the card.
QMI_UIM_REMOTE_CARD_POWER_DOWN_IND	0x0026	Indication to the control point to power down the card.
QMI_UIM_REMOTE_CARD_RESET_IND	0x0027	Indication to the control point to reset the card.

3.1 QMI_UIM_REMOTE_GET_SUPPORTED_MSGS

Queries the set of messages implemented by the currently running software.

UIM_REMOTE message ID

0x001E

Version introduced

Major - 1, Minor - 0

3.1.1 Request - QMI_UIM_REMOTE_GET_SUPPORTED_MSGS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.1.2 Response - QMI_UIM_REMOTE_GET_SUPPORTED_MSGS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Common version introduced	Common version last modified
Result Code	1.6	1.7

Optional TLVs

Name	Common version introduced	Common version last modified
List of Supported Messages	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	List of Supported Messages
Length	Var			2	
Value	→	uint16	supported_msgs_len	2	Number of sets of the following elements: • supported_msgs
		uint8	supported_msgs	Var	This array of uint8 is a bitmask where each bit represents a message ID, i.e., starting with the LSB, bit 0 represents message ID 0, bit 1 represents message ID 1, etc. The bit is set to 1 if the message is supported; otherwise, it is set to zero. For example, if a service supports exactly four messages with IDs 0, 1, 30, and 31 (decimal), the array (in hexadecimal) is 4 bytes [03 00 00 c0].

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_INFO_UNAVAILABLE	Information is not available

3.1.3 Description of QMI_UIM_REMOTE_GET_SUPPORTED_MSGS REQ/RESP

This command queries the set of messages implemented by the currently running software. This may be a subset of the messages defined in this revision of the service.

3.2 QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS

Queries the fields supported for a single command as implemented by the currently running software.

UIM_REMOTE message ID

0x001F

Version introduced

Major - 1, Minor - 0

3.2.1 Request - QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Common version introduced	Common version last modified
Service Message ID	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Service Message ID
Length	2			2	
Value	→	uint16	msg_id	2	ID of the command for which the supported fields are requested.

Optional TLVs

None

3.2.2 Response - QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Common version introduced	Common version last modified
Result Code	1.6	1.7

Optional TLVs

Name	Common version introduced	Common version last modified
List of Supported Request Fields	1.6	1.6
List of Supported Response Fields	1.6	1.6
List of Supported Indication Fields	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	List of Supported Request Fields
Length	Var			2	
Value	→	uint8	request_fields_len	1	Number of sets of the following elements: • request_fields
		uint8	request_fields	Var	This field describes which optional field IDs are supported in the QMI request. The array of uint8 is a bitmask where each bit represents a field (TLV) ID. Because fields 0 to 15 (decimal) are mandatory by definition, the first bit represents field ID 16. Starting with the LSB, bit 0 represents field ID 16, bit 1 represents field ID 17, etc. The bit is set to 1 if the field ID is supported; otherwise, it is set to zero. For example, if a service supports exactly four fields with IDs 16, 17, 30, and 31 (decimal), the array (in hexadecimal) is 2 bytes [03 c0].
Type	0x11			1	List of Supported Response Fields
Length	Var			2	
Value	→	uint8	response_fields_len	1	Number of sets of the following elements: • response_fields

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	response_fields	Var	This field describes which optional field IDs are supported in the QMI response. Its format is the same as request_fields.
Type	0x12			1	List of Supported Indication Fields
Length	Var			2	
Value	→	uint8	indication_fields_len	1	Number of sets of the following elements: • indication_fields
		uint8	indication_fields	Var	This field describes which optional field IDs are supported in the QMI indication. Its format is the same as request_fields.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_REQUESTED_NUM_UNSUPPORTED	Requested message ID is not supported by the currently running software
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_INFO_UNAVAILABLE	Information is not available

3.2.3 Description of QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS REQ/RESP

This command queries the fields supported for a single command as implemented by the currently running software.

If the request, response, or indication is supported for the given message ID, the corresponding optional array is included in QMI_UIM_REMOTE_GET_SUPPORTED_FIELDS_RESP, even if the message does not contain any optional fields. This enables the client to distinguish this case from one where the service does not support the request, response, or indication.

Examples are:

- If the specified message ID is not supported by the service, the response has qmi_result = QMI_RESULT_FAILURE and qmi_error = QMI_ERR_REQUESTED_NUM_UNSUPPORTED.
- If the specified message ID is an empty message, the response has qmi_result = QMI_RESULT_SUCCESS and qmi_error = QMI_ERR_NONE. None of the optional arrays are included.
- If the specified message ID supports the request with 0 optional fields, the response with 3 optional fields (16, 17, and 18 decimal), and does not support an indication, the response has the following:
  - qmi_result = QMI_RESULT_SUCCESS
  - qmi_error = QMI_ERR_NONE

– request_fields array is included with length zero

– response_fields array is included with length 1 value [07]

– indication_fields array is not included

Trailing zero bytes are omitted from the response. For example, if the message defines 20 different fields but the response only contains 16 bits, the client is to assume the last four fields are not supported.

3.3 QMI_UIM_REMOTE_RESET

Resets the service state variables of the requesting control point.

UIM_REMOTE message ID

0x0020

Version introduced

Major - 1, Minor - 0

3.3.1 Request - QMI_UIM_REMOTE_RESET_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.3.2 Response - QMI_UIM_REMOTE_RESET_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Result Code	1.0	1.0

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission

3.3.3 Description of QMI_UIM_REMOTE_RESET REQ/RESP

This command resets the issuing control point's state kept by the service. This is the equivalent to closing the service and reopening it. Because it is performed as a single operation, the client ID of the requesting control point does not change.

The control point's state variables change to their default values before the response is issued.

3.4 QMI_UIM_REMOTE_EVENT

Notifies the service of remote UIM events.

UIM_REMOTE message ID

0x0021

Version introduced

Major - 1, Minor - 0

3.4.1 Request - QMI_UIM_REMOTE_EVENT_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
UIM Remote Event Information	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM Remote Event Information
Length	8			2	
Value	→	enum	event	4	Event type received from the card. Values: • UIM_REMOTE_CONNECTION_ UNAVAILABLE (0x0) – Connection is unavailable • UIM_REMOTE_CONNECTION_ AVAILABLE (0x1) – Connection is available • UIM_REMOTE_CARD_INSERTED (0x2) – Card is inserted • UIM_REMOTE_CARD_REMOVED (0x3) – Card was removed • UIM_REMOTE_CARD_ERROR (0x4) – Card error • UIM_REMOTE_CARD_RESET (0x5) – Card reset • UIM_REMOTE_CARD_WAKEUP (0x6) – Card wake-up

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum	slot	4	Card slot for the event type received. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

Name	Version introduced	Version last modified
UIM Remote Answer to Reset Bytes	1.0	1.0
UIM Remote Wakeup Support	1.0	1.0
Error Cause for Card Error Event	1.0	1.2

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	UIM Remote Answer to Reset Bytes
Length	Var			2	
Value	→	uint8	atr_len	1	Number of sets of the following elements: • atr
		uint8	atr	Var	Answer to reset.
Type	0x11			1	UIM Remote Wakeup Support
Length	1			2	
Value	→	boolean	wakeup_support	1	Indicates whether the UIM Remote supports the wake-up property.
Type	0x12			1	Error Cause for Card Error Event
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum	error_cause	4	Indicates the cause of error for a card error event. • UIM_REMOTE_CARD_ERROR_ UNKNOWN_ERROR (0x0) – Unknown error • UIM_REMOTE_CARD_ERROR_ NO_LINK_ESTABLISHED (0x1) – No link was established • UIM_REMOTE_CARD_ERROR_ COMMAND_TIMEOUT (0x2) – Command timeout • UIM_REMOTE_CARD_ERROR_ DUE_TO_POWER_DOWN (0x3) – Error due to a card power down • UIM_REMOTE_CARD_ERROR_ DUE_TO_POWER_DOWN_TELECOM (0x4) – Error due to a telecom power down

3.4.2 Response - QMI_UIM_REMOTE_EVENT_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Result Code	1.0	1.0

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission

3.4.3 Description of QMI_UIM_REMOTE_EVENT REQ/RESP

This command notifies the service about the availability of a connection to a remote UIM. Once a connection is established, the command notifies the service about card events. Possible events for a specific slot include:

- Connection available or unavailable
- Card inserted and the associated ATR
- Card removed
- Card error

3.5 QMI_UIM_REMOTE_APDU

Exchanges the APDU with the remote card.

UIM_REMOTE message ID

0x0022

Version introduced

Major - 1, Minor - 0

3.5.1 Request - QMI_UIM_REMOTE_APDU_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Status of APDU Transaction	1.0	1.0
Card Slot	1.0	1.0
APDU ID	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Status of APDU Transaction
Length	2			2	
Value	→	enum16	apdu_status	2	APDU status will be either QMI_RESULT_SUCCESS or QMI_RESULT_FAILURE.
Type	0x02			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3
Type	0x03			1	APDU ID
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint32	apdu_id	4	Identifier for a command and response APDU pair.

Optional TLVs

Name	Version introduced	Version last modified
Response APDU Information	1.0	1.0
Response APDU	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Response APDU Information
Length	8			2	
Value	→	uint32	total_response_apdu_size	4	Total response APDU size for the transaction.
		uint32	response_apdu_ segment_offset	4	Offset of the APDU segment in the message.
Type	0x11			1	Response APDU
Length	Var			2	
Value	→	uint16	response_apdu_segment_len	2	Number of sets of the following elements: • response_apdu_segment
		uint8	response_apdu_segment	Var	APDU returned from the control point.

3.5.2 Response - QMI_UIM_REMOTE_APDU_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Result Code	1.0	1.0

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point, or the message was corrupted during transmission

3.5.3 Description of QMI_UIM_REMOTE_APDU REQ/RESP

This command relays the card response for a previously sent command APDU.

If the response length is greater than the maximum response APDU length supported per message, the control point divides the response into multiple chunks and sends each chunk in a different message, providing the offset of the chunk. This allows the service to reconstitute the entire response.

3.5.4 Indication - QMI_UIM_REMOTE_APDU_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0
APDU ID	1.0	1.0
Command APDU	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3
Type	0x02			1	APDU ID
Length	4			2	
Value	→	uint32	apdu_id	4	Identifier for a command and response APDU pair.
Type	0x03			1	Command APDU
Length	Var			2	
Value	→	uint16	command_apdu_len	2	Number of sets of the following elements: • command_apdu
		uint8	command_apdu	Var	APDU request sent to a control point.

Optional TLVs

None

3.5.5 Description of QMI_UIM_REMOTE_APDU_IND

The control point receives this indication when the service wants to transmit an APDU to the card on a specific slot.

3.6 QMI_UIM_REMOTE_CONNECT_IND

Indication to the control point to establish a connection with the card.

UIM_REMOTE message ID

0x0023

Version introduced

Major - 1, Minor - 0

3.6.1 Indication - QMI_UIM_REMOTE_CONNECT_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

None

3.6.2 Description of QMI_UIM_REMOTE_CONNECT_IND

The control point receives this indication when the service wants to establish a connection with the card on a specific slot and power it up.

3.7 QMI_UIM_REMOTE_DISCONNECT_IND

Indication to the control point to tear down the connection with the card.

UIM_REMOTE message ID

0x0024

Version introduced

Major - 1, Minor - 0

3.7.1 Indication - QMI_UIM_REMOTE_DISCONNECT_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

None

3.7.2 Description of QMI_UIM_REMOTE_DISCONNECT_IND

The control point receives this indication when the service wants to tear down a connection with the card on a specific slot.

3.8 QMI_UIM_REMOTE_CARD_POWER_UP_IND

Indication to the control point to power up the card.

UIM_REMOTE message ID

0x0025

Version introduced

Major - 1, Minor - 0

3.8.1 Indication - QMI_UIM_REMOTE_CARD_POWER_UP_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

Name	Version introduced	Version last modified
Response Timeout	1.2	1.2
Voltage Class	1.2	1.2

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Response Timeout
Length	4			2	
Value	→	uint32	response_timeout	4	Response timeout in milliseconds.
Type	0x11			1	Voltage Class
Length	4			2	
Value	→	enum	voltage_class	4	Voltage class. Values: • UIM_REMOTE_VOLTAGE_CLASS_C_LOW (0x0) – VOLTAGE_CLASS_C_LOW • UIM_REMOTE_VOLTAGE_CLASS_C (0x1) – VOLTAGE_CLASS_C • UIM_REMOTE_VOLTAGE_CLASS_C_HIGH (0x2) – VOLTAGE_CLASS_C_HIGH • UIM_REMOTE_VOLTAGE_CLASS_B_LOW (0x3) – VOLTAGE_CLASS_B_LOW • UIM_REMOTE_VOLTAGE_CLASS_B (0x4) – VOLTAGE_CLASS_B • UIM_REMOTE_VOLTAGE_CLASS_B_HIGH (0x5) – VOLTAGE_CLASS_B_HIGH All other values are reserved for future use.

3.8.2 Description of QMI_UIM_REMOTE_CARD_POWER_UP_IND

The control point receives this indication when the service wants to power up the card on a specific slot.

The uim_remote_voltage_class_enum enumeration for the Voltage Class TLV is defined to have common node values among the control point, modem, PMIC, and RPM for Qualcomm proprietary APIs.

Voltage Class B values:

- UIM_REMOTE_VOLTAGE_CLASS_B_HIGH – 3.05 Volts
- UIM_REMOTE_VOLTAGE_CLASS_B – 3.0 Volts
- UIM_REMOTE_VOLTAGE_CLASS_B_LOW – 2.85 Volts

Voltage Class C values:

- UIM_REMOTE_VOLTAGE_CLASS_C_HIGH – 1.9 Volts
- UIM_REMOTE_VOLTAGE_CLASS_C – 1.8 Volts
- UIM_REMOTE_VOLTAGE_CLASS_C_LOW – 1.7 Volts

3.9 QMI_UIM_REMOTE_CARD_POWER_DOWN_IND

Indication to the control point to power down the card.

UIM_REMOTE message ID

0x0026

Version introduced

Major - 1, Minor - 0

3.9.1 Indication - QMI_UIM_REMOTE_CARD_POWER_DOWN_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

Name	Version introduced	Version last modified
Power-down Mode	1.1	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Power-down Mode
Length	4			2	
Value	→	enum	mode	4	Power-down mode. Values: • UIM_REMOTE_POWER_DOWN_TELECOM_INTERFACE (0x0) – Power down the telecom • UIM_REMOTE_POWER_DOWN_CARD (0x1) – Power down the card All other values are reserved for future use.

3.9.2 Description of QMI_UIM_REMOTE_CARD_POWER_DOWN_IND

The control point receives this indication when the service wants to power down the card on a specific slot.

Power-down mode indicates the type of power down, e.g., power down of the entire card or only the telecom interface. This TLV is applicable to limited scenarios and can be ignored in other cases.

3.10 QMI_UIM_REMOTE_CARD_RESET_IND

Indication to the control point to reset the card.

UIM_REMOTE message ID

0x0027

Version introduced

Major - 1, Minor - 0

3.10.1 Indication - QMI_UIM_REMOTE_CARD_RESET_IND

Message type

Indication

Sender

Service

Indication scope

Unicast (per control point)

Mandatory TLVs

Name	Version introduced	Version last modified
Card Slot	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Card Slot
Length	4			2	
Value	→	enum	slot	4	Slot type. Values: • UIM_REMOTE_SLOT_NOT_ APPLICABLE (0x0) – Not applicable • UIM_REMOTE_SLOT_1 (0x1) – Slot 1 • UIM_REMOTE_SLOT_2 (0x2) – Slot 2 • UIM_REMOTE_SLOT_3 (0x3) – Slot 3

Optional TLVs

None

3.10.2 Description of QMI_UIM_REMOTE_CARD_RESET_IND

The control point receives this indication when the service wants to warm reset the card on a specific slot.

A References

A.1 Related Documents

Title	Number
Qualcomm Technologies	
QMI Client API Interface Specification	80-N1123-1
QMI Common Service Interface API Interface Specification	80-N1123-2
Qualcomm Messaging Interface (QMI) Architecture	80-VB816-1
Presentation: Bluetooth® SAP Client Mode Overview	80-NC254-75
Presentation: USB UICC Overview	80-NN611-1

A.2 Acronyms and Terms

Acronym or term	Definition
ATR	answer to reset
APDU	application protocol data unit
QMI	Qualcomm messaging interface
TLV	type-length-value
UIM	user identity module