MOTOTRBO™

Text Messaging Protocol Specification

Version 01.04

9/12/2011

COPYRIGHTS

The enclosed documents and ideas embodied herein are the proprietary information of Motorola Solutions. Any dissemination or disclosure of such violates Motorola Solutions' intellectual property rights. Motorola Solutions reserves all rights to all actions arising there under.

Motorola Solutions disclaims any liability for any use of the specification. Motorola Solutions limits all warranties to the extent allowed by law. Furthermore, Motorola Solutions reserves the right to change this specification at any time without any prior notification. And there is no guarantee that such changes will be backwards compatible with previous version of the specification.

REVISION HISTORY

Version	Date	Section	Page	Lines	Description
01.03	06/24/2011	3.3.1 3.3.2 3.3.3 3.3.4	13,15 15 17 18,19	228-231, 249-254 264-268, 272-276 288-292, 299-303 329-333, 343-346	Propagated “message size” and “address byte” field requirements from “Message Structure” section to each of the sub-sections beneath “Message Definitions”. **(CCMPD01524698)**
01.04	08/17/2011	1.4	7	86	Removed reference to deprecated document (USB Configuration Guide) **(CCMPD01492794)**

TABLE OF CONTENTS

1 Introduction

1.1 Overview

This document describes the specification of the Text Messaging ADK and is based upon the specific features and functionality implemented for the MOTOTRBO radio. This includes detail on the format of the Text Messaging protocol as well as the functionality of the protocol commands and responses. The Text Messaging protocol is an application layer protocol and is “language-independent.” Developers may implement this protocol in any programming language which supports bit stream manipulation. Both the MOTOTRBO portable and mobile radios support the Text Messaging protocol.

1.2 Terminology

Application Layer (OSI Layer 7) - this layer supports application processes.

Big Endian - an order in which the "big end" (most significant value in the sequence) is stored first (at the lowest storage address). For example, in a big-endian computer, the two bytes required for the hexadecimal number 4F52 would be stored as 4F52 in storage (if 4F is stored at storage address 1000, for example, 52 will be at address 1001).

Broadcast - an unsolicited message that is sent to one or more recipients.

Capacity Plus - A digital trunking system configuration accommodating an increased subscriber base and increased volumes of data

Confirmed Delivery Channel - any communications channel that provides confirmation to the sender when the receiver receives its data correctly.

Data Link Layer (OSI Layer 2) - this layer encodes and decodes the data packets. It furnishes transmission protocol knowledge.

Little Endian - an order in which the "little end" (least significant value in the sequence) is stored first. For example, in a little-endian computer, the two bytes required for the hexadecimal number 4F52 would be stored as 524F (52 at address 1000, 4F at 1001).

Network Layer (OSI Layer 3) - this layer provides routing and forwarding services. It creates logical paths, known as virtual circuits, for transmitting data from node to node.

Open System Interconnection (OSI) – a model which defines a networking framework for implementing protocols in seven layers. Control is passed from one layer to the next, starting at the top layer in one station, and proceeding to the bottom layer, over the channel to the next station, and back up the hierarchy.

Packet - a block of transmitted data.

Physical Layer (OSI Layer 1) - this layer conveys the bit stream (i.e. electrical impulse, light or radio signal) through the network at the electrical and mechanical level. It provides the physical means of sending and receiving data.

Presentation Layer (OSI Layer 6) - this layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems.

Reply - a message that is sent in response to a Request message.

Request - a message that expects an immediate reply

Reliable Channel - any communications channel that provides a mechanism to detect and optionally correct an error in data received over the communications channel. Some errors may not be possible to detect or correct. A reliable channel simply provides more reliability above and beyond an unreliable communications channel.

Response - a message that is sent as result of a previous message. A response could be a reply or broadcast message

Session Layer (OSI Layer 5) - this layer establishes, manages and terminates connections between applications.

Transport Layer (OSI Layer 4) - this layer provides transfer of data between hosts.

Unreliable Channel - a communication channel that provides no means for a receiver to detect communication errors.

1.3 Assumptions

It is assumed that the reader of this documentation has the following domain knowledge:

- Principles of two-way radio communications
- Procedural or Object-Oriented Programming
- Transmission Control Protocol / User Datagram Protocol (TCP/UDP)
- Internet Protocol (IP)
- Universal Serial Bus (USB)

The following domain knowledge is considered beneficial, but is not required:

- Open Systems Interconnection (OSI) Model

1.4 References

[1] MOTOTRBO Text Messaging ADK Guide
[2] MOTOTRBO Data Services Overview

2 General Description of Protocol

2.1 Introduction

The MOTOTRBO Text Messaging protocol is defined to provide service status and delivery functions for short messaging between MOTOTRBO radios. Delivery of text messages can be 1-to-1 or 1-to-many radios.

2.2 Protocol Stack

The following diagram shows the protocol stack for Text Messaging for the PC-attached MOTOTRBO radio as well as some other MOTOTRBO subscriber unit in the Radio Network.

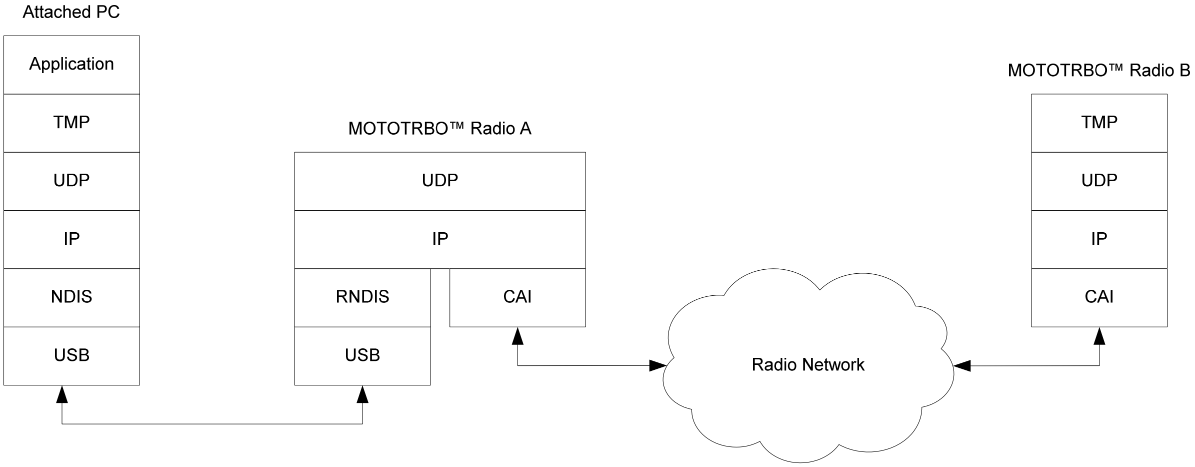


Figure 1 – Example Text Messaging Protocol Stack

2.3 Message Types

In the Text Messaging protocol, the messages are classified into two types:

- Control Message – a message sent to a MOTOTRBO subscriber unit indicating the status of the Text Messaging service
- User Message – a message, containing text, sent from one MOTOTRBO subscriber unit to another or to a group

And, for these two types of messages, the transaction falls into one of two categories:

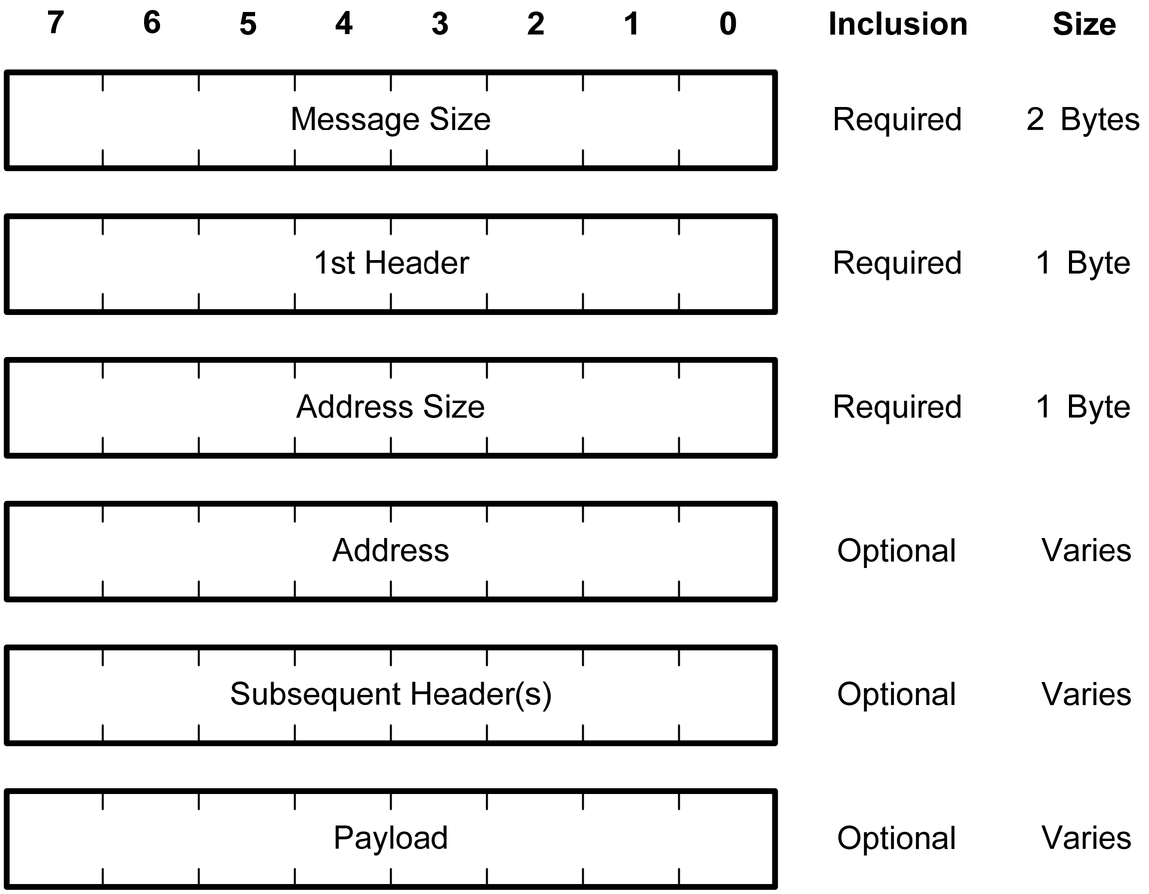
- Confirmed Delivery – the receipt of a message is acknowledged with a response indicating success or failure
- Unconfirmed Delivery – the receipt of a message is not acknowledged and a response is not expected

In general, all control messages are confirmed for delivery, meaning the recipient of the control message acknowledges the sent message. For user messages, only messages intended for one recipient are confirmed for delivery. Acknowledgement to a user message intended for a group of MOTOTRBO subscriber units is not expected and should not be sent.

3 Specific Protocol Messages & Formats

3.1 Message Structure

The message type as well as other message characteristics is indicated by the contents of the message's header. The structure of a message is described in Figure 2. Please see reference [1] for more information.



7 6 5 4 3 2 1 0	Inclusion	Size
Message Size	Required	2 Bytes
1st Header	Required	1 Byte
Address Size	Required	1 Byte
Address	Optional	Varies
Subsequent Header(s)	Optional	Varies
Payload	Optional	Varies

Figure 2 – Text Messaging Protocol Message Structure

The fields are defined as follows:

1. Message Size (Always Present) – This indicates the number of bytes to follow. The size DOES NOT include the two Message Size bytes.
2. First Header (Always Present) – This header includes basic information such as the packet data unit (PDU) type.
3. Address Size (Always Present) – This field indicates the size of the address field that follows in bytes. If the Address field is not used, then Address Size must be cleared to 0.
4. Address (Present if Address Size is not zero) – This field indicates the application layer address which is needed when the destination is outside of the radio network.

The actual encoding of the address field is left to the actual product to define. Examples of the address field are: an IP address, an alphanumeric text string, a simple number, etc. If the destination is a MOTOTRBO radio, then the radio ID must be encoded in UCS2-LE format.

5. Subsequent Header(s) (Depends on PDU Type) – Includes additional information specific to the PDU Type.
6. Payload (Depends on PDU Type) – This includes any PDU specific payload information.

7	6	5	4	3	2	1	0
Ext	Ack	Res	Ctrl	PDU Type			

Figure 3 – 1st Header Bit Definitions

Header Byte 1 (Always Present):

- Extension Bit (Ext) – If this bit is set, there are additional, optional header byte(s) that further define the message.
  - o Required headers are not specified by the use of this bit but are implied in the definition of the PDU.
  - o If an optional header is used, then the PDU must include all optional headers that occur before that header.
- Acknowledgement Required – The use of this bit depends upon the PDU itself. For the purposes of this bit, the PDU is considered an acknowledgement PDU or a non-acknowledgement PDU.
  - o Non-Acknowledgement PDU - If this bit is set, it indicates that the initiator is requesting an acknowledgement to this PDU.
  - o Acknowledgement PDU – If this bit is clear, this acknowledge indicates a success of the respective request. If this bit is set, this acknowledgement indicates the failure of the request.
- Reserved (Res) – This bit is reserved for future use.
- Control/User Bit (Ctrl) – If this bit is set, it indicates that this is a control message for the protocol. The protocol layer uses control messages to exchange information with its peer. If this bit is clear, this is user data (i.e. text message).
- PDU Type – Defines the PDU Type. Types are associated with the control/user setting. For example, there may be a PDU Type 0 for control and a PDU Type 0 for user data (see section 3.2 for more information).

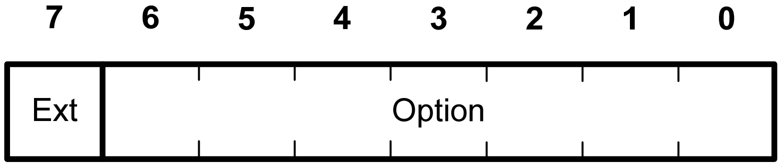


Figure 4 – Nth Header Bit Definitions

Header Byte N (Optional, N>1, Dependent on byte N-1's Ext setting):

- Extension Bit (Ext) – If this bit is set, there is another optional byte that follows that further defines the type (the 2nd defined header byte should follow the extension).
- Option – These bits contain control or data values in context with the PDU type. This bit space may be subdivided into smaller fields representing different options.

Additional Information:

- For payload encoded in UCS2-LE (which is the Little Endian format), two bytes will be used for encoding each character, for example the character "h" in the payload will be shown as "0x68 0x00". However, the message size will always be stored as a big endian value. For example, if the message size is 140, it shall be shown as "0x00 0x8C".
- If payload data is required, the payload structure must follow a defined format.
- Header extension bytes that are optional must follow the defined order. If optional header byte (3) is required, then header byte (2) must be present.
- Any PDU types that are not understood should be ignored.

3.2 PDU Types

The Text Messaging protocol PDU types for control messages and user messages are defined below. Please note the following definitions for these tables: ACK – indicates whether an acknowledgement is required for this PDU Type; 2nd Header – indicates if a subsequent header is required for this PDU Type.

Control Message PDU Types				
PDU Type	**Name**	**ACK**	**2nd Header**	**Notes**
%0000	TMS Service Availability	Y	Y	Sent on service startup.
%0001 - %1110	Reserved	–	N/A	
%1111	TMS Acknowledgement	N	Optional	Additional message bytes depending on request message.

Table 1 – Control Message PDU Types

User Message PDU Types				
PDU Type	**Name**	**ACK**	**2nd Header**	**Notes**
%0000	Simple Text Message	Y	Y	Payload will contain the message.
%0001 - %1111	Reserved	–	N/A	

Table 2 – User Message PDU Types

3.3 Message Definitions

This section defines the Text Messaging protocol messages that are sent and received between multiples devices. The messages provide two basic operations: delivery of a text message to a target device(s) or receipt acknowledgement of a text message from a target device. Exceptions to these basic operations are noted.

3.3.1 TMS Service Availability

This message is used to announce Text Messaging Service availability on the system by the TMS Server. An optional header can be included to indicate the capability of the device that is generating the service availability. If a device header is not included, the default is a TMS Server (Internal Capability). The address of the device should be determined from the source address of the received datagram. This address should be stored for use by the radio.

This message must only be sent by a TMS application providing the server functionality described by the “Device” field.

NOTE

For Capacity Plus the TMS Service Availability is optional. The TMS server IP address is CPS configurable.

Message Size

Required 2-byte field indicating the number of bytes to follow in the message. The message size does not include the two Message Size bytes. Refer to Section **3.1**

Headers

1st Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	Varies	Set to 1 if desirable to describe TMS capability.
6	Acknowledgement	1	Set to 1 to request acknowledgement.
5	Reserved	0	Reserved for future use
4	Control / User	1	Set to 1 to indicate Control message type.
3:0	PDU Type	%0000	Type code for this message.

2nd Header Byte (Optional)			
Bits	**Field**	**Value**	**Notes**
7	Extension	0	Clear to 0; no other optional headers to follow.

6:3	Reserved	0	Reserved for future use.
2:0	Device	Varies	0x00 – Limited Capability 0x01 – Internal Capability 0x02 – External Capability 0x03 – Full Capability

Device definitions:

- Limited Capability – Basic text messaging capability is enabled (i.e. to default devices such as dispatchers) only.
- Internal Capability – Text messaging capability to devices on the Radio Network is supported (i.e. other subscribers) only.
- External Capability – Text messaging capability to devices on the Customer Enterprise Network is supported (i.e. email addresses, external networks, etc.) only.
- Full Capability – Indicates the device is capable of delivering messages to default devices, Radio Network devices, and Customer Enterprise Network devices.

Address Size

Required 1-byte field. If the address field is unused, the field is cleared to “0”. Refer to Section **3.1**.

Payload

None.

3.3.2 TMS Service Availability Acknowledgement

This message is used to positively acknowledge a service availability message.

This message is only sent by MOTOTRBO radios and any TMS applications that function as a peer to any subscriber unit within the Radio Network.

Message Size

Required 2-byte field indicating the number of bytes to follow in the message. The message size does not include the two Message Size bytes. Refer to Section **3.1**

Headers

1st Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	0	Clear to 0; no other optional headers to follow.
6	Acknowledgement	0	Clear to 0 to indicate successful acknowledgement.
5	Reserved	0	Reserved for future use
4	Control / User	1	Set to 1 to indicate Control message type.
3:0	PDU Type	%1111	Type code for this message.

Address Size

Required 1-byte field. If the address field is unused, the field is cleared to “0”. Refer to Section **3.1**.

Payload

None.

3.3.3 TMS Acknowledgement

This message is used to positively acknowledge a text message.

This message is sent by all units within the Radio Network when appropriate.

Message Size

Required 2-byte field indicating the number of bytes to follow in the message. The message size does not include the two Message Size bytes. Refer to Section **3.1**

Headers

1st Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	Varies	Set to 1 to include the sequence number
6	Acknowledgement	0	Clear to 0 to indicate successful acknowledgement.
5	Reserved	0	Reserved for future use
4	Control / User	1	Set to 1 to indicate Control message type
3:0	PDU Type	%1111	Type code for this message.

2nd Header Byte (Optional)			
Bits	**Field**	**Value**	**Notes**
7	Extension	Varies	Set to 1 if size of Sequence Number field overflows.
6:5	Reserved	0	Reserved for future use
4:0	Sequence Number	Varies	Remaining bits hold the sequence number for this message. This matches the originating message sequence number.

3rd Header Byte (Optional)			
Bits	**Field**	**Value**	**Notes**
7	Extension	0	Clear to 0: no other optional headers to follow.
6:5	Sequence Number Extension	Varies	MSB of sequence number (if overflow needed).
4:0	Reserved	0	Reserved for future use

Address Size

Required 1-byte field. If the address field is unused, the field is cleared to “0”. Refer to Section **3.1**.

Payload

None.

3.3.4 Simple Text Message

This message is used to send basic text messages.

In a message inbound to the TMS PC-based application (Server), the Address field will contain the destination address of the target subscriber. In a message outbound from the TMS Server, the Address field will contain the source address of the initiating subscriber.

There may be a need in certain cases for products to have the capability to control whether an acknowledgment to a user message is required. For example, with a message being sent to a group of radios, the message may be sent configured to not generate an acknowledgement to limit possible channel contention.

NOTE

The sequence number in the 2nd Header Byte is 7 bits in length. Bits 0-4 shall be stored in the 2nd Header Byte itself, while the remaining bits 5 and 6 (MSB of the sequence number) shall be stored in the 3rd Header Byte. If the value of the sequence number used is less than 31 (decimal), then bits 5 and 6 shall be filled with the value 0.

This message is sent by all units within the Radio Network when appropriate.

Message Size
Required 2-byte field indicating the number of bytes to follow in the message. The message size does not include the two Message Size bytes. Refer to Section **3.1**

Headers

1st Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	Varies	Set to 1 to include the sequence number.
6	Acknowledgement	Varies	Set to 1 to request acknowledgement.
5	Reserved	1	Reserved for future use
4	Control / User	0	Clear to 0 to indicate User message type.
3:0	PDU Type	%0000	Type code for this message.

2nd Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	Varies	Set to 1 if size of Sequence Number overflows or to indicate Encoding.
6:5	Reserved	0	Reserved for future use
4:0	Sequence Number	Varies	Remaining bits hold the sequence number for this message.

3rd Header Byte (Required)			
Bits	**Field**	**Value**	**Notes**
7	Extension	0	Clear to 0; no other optional headers to follow.
6:5	Sequence Number Extension	Varies	MSB of sequence number (if overflow needed).
4:0	Encoding	0x04	Indicates the encoding of the non-size portions of the payload (name field). Currently only supports: 0x04 – for UCS2-LE encoding

Address Size

Required 1-byte field. If the address field is unused, the field is cleared to "0". Refer to Section **3.1**.

Payload

7	6	5	4	3	2	1	0	Inclusion	Size
Text Message								Required	Varies

MOTOROLA SOLUTIONS

6880309T47