



LM Direct™
Reference Guide

Version 8.4.e.3
January 2011

Table of Contents

INTRODUCTION	5
ABOUT THIS MANUAL	5
<i>Revision History</i>	5
ABOUT THE READER	8
ABOUT CALAMP	9
SECTION 1— INTRODUCTION.....	10
OVERVIEW	10
COMMUNICATIONS	10
SECTION 2 — MESSAGE STRUCTURES	11
BASIC MESSAGE STRUCTURE	12
<i>Options Header</i>	13
<i>Message Header</i>	15
NULL MESSAGE (MESSAGE TYPE 0)	16
ACKNOWLEDGE MESSAGE (MESSAGE TYPE 1)	17
EVENT REPORT MESSAGE (MESSAGE TYPE 2)	18
<i>ID Report Message (Message Type 3)</i>	22
USER MESSAGE (MESSAGE TYPE 4)	26
APPLICATION MESSAGE (MESSAGE TYPE 5)	30
PARAMETER MESSAGE (MESSAGE TYPE 6)	34
<i>Parameter Update Begin Operation</i>	34
<i>Parameter Write Operation:</i>	34
<i>Parameter Update End Operation</i>	34
<i>Parameter Read Operation:</i>	34
UNIT REQUEST MESSAGE (MESSAGE TYPE 7)	41
LOCATE REPORT MESSAGE (MESSAGE TYPE 8)	43
USER MESSAGE WITH ACCUMULATORS (MESSAGE TYPE 9)	47
MINI EVENT REPORT MESSAGE (MESSAGE TYPE 10)	53
MINI USER MESSAGE (MESSAGE TYPE 11)	56
SECTION 3 — LM DIRECT™ MESSAGE FLOWS	57
OVERVIEW	57
EVENT REPORT MESSAGE FLOW	58
<i>Message Flow</i>	58
<i>Message Contents</i>	59
UNIT REQUEST MESSAGE FLOW	60
<i>Message Flow</i>	60
<i>Message Contents</i>	61
USER MESSAGE FLOW - OUTBOUND	62
<i>Message Flow</i>	62
<i>Message Contents</i>	63
USER MESSAGE FLOW - INBOUND	64
<i>Message Flow</i>	64
<i>Message Contents</i>	64
PARAMETER MESSAGE - READ	66
<i>Message Flow</i>	66
<i>Message Contents</i>	67

PARAMETER MESSAGE - WRITE	68
<i>Message Flow</i>	68
<i>Message Contents</i>	69
MESSAGE FORWARDING	70
<i>Message Flow</i>	70
<i>Message Contents</i>	71
MESSAGE PROXY - OUTBOUND	73
<i>Message Flow</i>	73
<i>Message Contents</i>	74
MESSAGE PROXY - INBOUND	78
<i>Message Flow</i>	78
<i>Message Contents</i>	78
MESSAGE FORWARDING WITH IP LOOKUP	82
<i>Message Flow</i>	82
<i>Message Contents</i>	83
APPENDIX A — USING MOBILE ID.....	85
APPENDIX B — USING AUTHENTICATION	86
MOBILE ID ENABLED EXAMPLE	86
MOBILE ID DISABLED EXAMPLE	86
APPENDIX C — PROCESSING THE OPTIONS HEADER	87
APPENDIX D — ROUTING FIELD AND FORWARDING FIELD DEFINITIONS AND USES	89
APPENDIX E — APPLICATION MESSAGE TYPES AND FORMATS.....	90
TYPES	90
FORMATS	91
<i>IP Request</i>	91
<i>IP Report</i>	91
<i>jPOD™ Vehicle Bus Data</i>	91
<i>Time Sync</i>	93
<i>Download App Messages</i>	93
<i>AT Command</i>	93
<i>Version Report</i>	93
<i>GPS Status Report</i>	93
<i>Message Statistics Report</i>	97
<i>State Report</i>	99
<i>Geo-Zone Action Message</i>	103
<i>Geo-Zone Update Message</i>	105
<i>Capture Message</i>	107
APPENDIX F — GEO-ZONE MESSAGING.....	109
USING GEO-ZONES	109
GEO-ZONE READ & WRITE VERSION REQUESTS	109
GEO-ZONE WRITE REQUESTS	111
GEO-ZONE READ REQUESTS	113

INTRODUCTION

Welcome to the LM Direct™ Reference Manual. This manual is intended to give you detailed information on the usage and operation of the LM Direct™ location messaging interface. This includes message specifications, operational issues, unit configuration and messaging examples.

About This Manual

In order to accurately describe the features and usage of the LM Direct™ interface, this manual has been divided into the following sections.

- Section 1 — Introduction
- Section 2 — Message Structures
- Section 3 — Message Flow
- Section 4 — Appendices

Revision History

Version	Date	Author	Description
0.0	17/07/2003	pherg	Initial Draft
0.1	08/09/2003	pherg	Add routing fields to options header, Added in Msg Type values, completed Param Table in Appendix F.
0.2	09/17/2003	pherg	Added support for Locate Requests and Reports. Added forwarding field to options header. Added Routing and Forwarding Appendix. Reordered the Appendix designations. Fixed BCD digit ordering definitions in ID Report msg.
1.0	10/02/2003	pherg	Release
	10/23/2003	STP	Updated parameter definitions
	10/30/2003	pherg	Added GPS Tracking to Unit Status field Added definition of GPS Status App Msg. Added definition of Msg Statistics App Msg and Unit Request support.
	11/03/2003	STP	Added Authentication Appendix
1.1	11/04/2003	STP	Release
	11/09/2003	pherg	Corrected Param numbering (Appendix F)
	11/10/2003	STP	Updated parameter list
	11/10/2003	STP	Added STATs App Message description
	11/17/2003	STP	Added Message Flow Section
	11/17/2003	STP	Added Forwarding Operation Byte description

Version	Date	Author	Description
1.2	11/17/2003	STP	Release
	10/07/2004	STP	Updated Unit Status Field to reflect Modem Min Test
	10/07/2004	STP	Added new Parameter definitions for 6.1k and 6.1m features
	10/07/2004	STP	Corrected Parameter naming errors
1.3	10/07/2004	STP	Release
	05/19/2005	STP	Update Parameter list for 6.2 and 7.0 code base
	05/19/2005	STP	Added new NAK type
	05/19/2005	STP	Fixed Documentation error for Comm State
	05/19/2005	STP	Added S-Register Appendix
1.4	05/19/2005	STP	Release
	06/23/0225	STP	Update Parameter table definitions
	06/23/2005	STP	Updated S-Register definitions
1.5			RELEASE
	01/06/2005	STP	Updated Parameter table definitions
			Updated S-Register definitions
			Updated Unit Status definition
1.5.1			RELEASE
	02/20/2006	STP	Fixed a Documentation Error with Speed Above and Below Parameter IDs
1.6	04/11/2007	Pherg	Convert doc to CalAmp Updated ID Report with ICC-ID extension
1.6.1	04/20/2007	Pherg	Added Server-to-Mobile sequence numbering Added NAK for duplicate messages Updated Application ID table in ID Report msg Re-pagenated
1.6.2	8/13/2007	Pherg	Added non-zero outbound msg sequence numbers for duplicate msg rejection.
1.6.3	10/19/07	Pherg	Added 'Response Redirection' to prepended header.
1.6.4	11/17/2007	STP	Added Geo-Zone Messaging Appendix Added Geo-Zone Message Definitions (See Appendix E)
1.6.4	11/17/2007	STP	Added User Message With Accumulators Definition (Message Type 9)
1.6.4	11/17/2007	STP	Added State Report Definition
1.6.4	11/17/2007	STP	Removed Parameter and S-Register

Version	Date	Author	Description
			definitions. (See LMU Users Guide)
1.6.5	12/13/2007	Pherg	Added Unit Request 17 to set Zone to current location.
8.2.a.1	3/9/2008	STP	Updated document versioning scheme to match LMU firmware
8.2.g.1	07/02/2008	Pherg	Added Mini Event Report
8.2.g.1	07/22/2008	Pherg	Fixed Param Write Report example
8.2.g.2	08/08/08	Pherg	Added Time Sync App Message Definition
.	05/23/2009	STP	Updated Fix Status definitions
	05/23/2009	STP	Added Parameter Begin Message definition
	05/23/2009	STP	Added Parameter End Message definition
8.3.c.1	05/23/2009	STP	RELEASE
	05/25/2009	STP	Further updates to Fix Status definitions
8.3.c.2			RELEASE
	06/11/2009	STP	Updated 'Only Present when sent by LMU' range for Application Messages
	08/03/2009	STP	Updated PEG Trigger and Action nomenclature for Unit Request messages.
	08/12/2009	STP	Corrected Bit mapping for PEG Flags in App State Application message definition
	09/04/2009	STP	Updated Comm State definition to include Voice and Roaming indications
	09/27/2009	STP	Updated User Message Type to include Routing and ID fields.
8.4.a.1	06/04/2010	Pherg	Added Mini User Message
8.4.b.1	06/04/2010	PHerg	Modified ID Report to add extensions
8.4.c.1	08/16/2010	PHerg	Added Acceleration Report App Message
8.4.3.1	10/02/2010	STP	Removed Acceleration Report App Message
	10/02/2010	STP	Added Capture Report Application Message definition
8.4.3.2	11/01/2011	STP	Updated format to match other CalAmp docs

About The Reader

In order to limit the size and scope of this manual we have made some assumptions about the reader.

- Familiar with basic programming concepts
- Understands IP networking concepts and the UDP protocol
- Familiar with GPS concepts and terminology
- Able to query and set values into an LMU or TetheredLocator using LMU Manager or AT Commands
- Familiar with the features and capabilities of the LMU and TetheredLocator

About CalAmp

CalAmp is a leading provider of wireless communications products that enable anytime/anywhere access to critical information, data and entertainment content. With comprehensive capabilities ranging from product design and development through volume production, CalAmp delivers cost-effective high quality solutions to a broad array of customers and end markets. CalAmp is the leading supplier of Direct Broadcast Satellite (DBS) outdoor customer premise equipment to the U.S. satellite television market. The Company also provides wireless data communication solutions for the telemetry and asset tracking markets, private wireless networks, public safety communications and critical infrastructure and process control applications. For additional information, please visit the Company's website at www.calamp.com.

SECTION 1 – INTRODUCTION

Overview

The LM Direct™ interface was designed as a means for an LMU or TetheredLocator to provide location and status information to an application without requiring the LM Exchange middleware. It also allows the application to read and write the LMU's configuration parameters including the PEG event list and trigger parameters. These applications typically will take one of two forms: Either an in-vehicle application running on a laptop (or PDA) or a management/tracking application on a back-end system.

LM Direct™ provides the following features and capabilities to an application:

- Location Data — Time-of-Fix, Latitude, Longitude, Heading, Speed, Altitude, HDOP, Number of Satellites and Fix Status
- Communications Data — Signal Strength (RSSI), Comm State and Carrier/Operator Identifiers
- LMU Data — Mobile ID, Event Code, Event Index, Event Timestamp, Report Sequence Number, Input States, and Accumulator values.
- PEG-based event reporting
- User messaging between the application and a serial device attached to the LMU
- Full programming of the LMU's PEG functionality
- Full programming of other LMU configuration parameters
- Acknowledgement and retransmission functionality
- Optional logging of event reports and user messages in the LMU when delivery to the application isn't possible

Communications

The LM Direct™ interface in the LMU listens on UDP port 20510 for any requests generated by the application; that is, all outbound messages (i.e., those sent to an LMU) are sent to port 20510. Responses to outbound messages (e.g., ACK/NAK messages) are sent back to the originating IP address and port. Inbound request messages (e.g., Event Reports, User Messages) are sent by the LMU to the IP address and port currently selected from the Inbound Routing List. Typically the application will listen for these messages on port 20500.

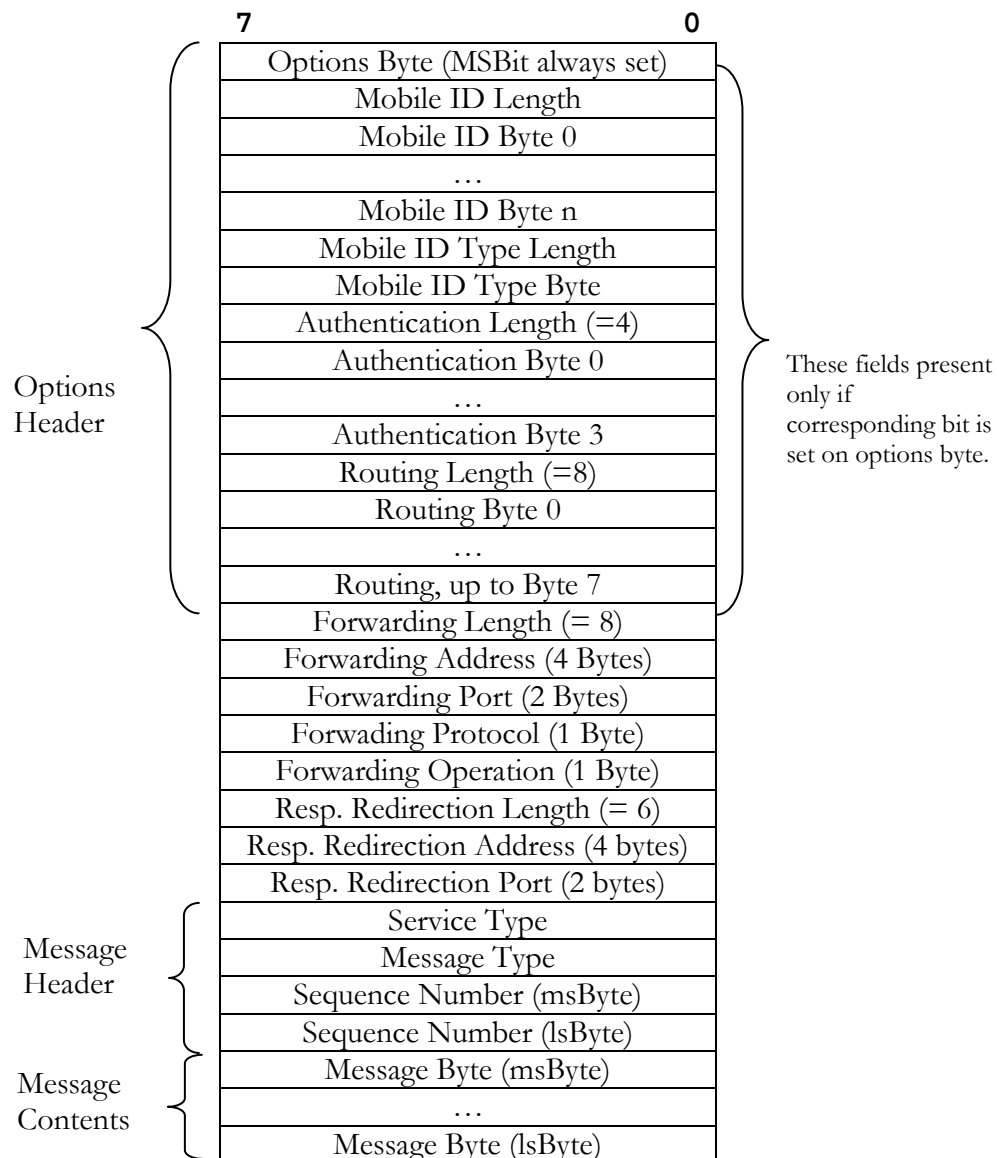
SECTION 2 – MESSAGE STRUCTURES

Messages passed between an application and the LM Direct™ interface have the same basic packet structure. This structure consists of a 20 byte IP header, an 8 byte UDP header, the LM Direct™ header and the LM Direct™ message.

Note that all bytes in multi-byte fields are transmitted in Net Endian format (Big Endian) where the most significant bits are transmitted first. For example, for a 32-bit field, bits 31-24 are transmitted first, 16-23 second, 8-15 third and 0-7 last

Basic Message Structure

The generic message structure of an LM Direct™ message is:



An LM Direct™ message comprises the Options Header, Message Header and Message Contents. The Options byte at the beginning of the Options Header is bit-mapped to indicate which Options Header fields will follow. If the most-significant bit (bit 7) of the first byte of a message is set, the Options Header is present in the message beginning with the Options Byte. Otherwise, the message begins with the Message Header (Service Type is the first byte); that is to say that if the first byte of an LM Direct™ message has the most-significant bit set (bit 7), then the Options Header is present. Likewise, if it is cleared, the Options Header is not included in the message.

Options Header

The Options Header fields used in LM Direct™ are defined as follows:

Options

Header Content Options:

- Bit 0: Mobile ID (0 = disabled, 1 = enabled)
- Bit 1: Mobile ID Type (0 = disabled, 1 = enabled)
- Bit 2: Authentication Word (0 = disabled, 1 = enabled)
- Bit 3: Routing (0 = disabled, 1 = enabled)
- Bit 4: Forwarding (0 = disabled, 1 = enabled)
- Bit 5: Response Redirection (0 = disabled, 1 = enabled)
- Bit 6: Not Used – Reserved for future use
- Bit 7: Always set.

Mobile ID Length

The number of bytes to follow which contain the Mobile ID data

Mobile ID

The Mobile ID of the LMU that either originated the LM Direct™ message or the LMU for which the message is intended. The Mobile ID is made up of numerical digits and is encoded in this field as packed BCD. The most significant digit is placed in the upper four bits of the first byte. In the case where an odd number of digits are used, a 0x0F is used to pad the lower 4 bits of the last byte.

Mobile ID Type Length

The number of bytes to follow which contain the Mobile ID Type data. Note that this value will always be 1.

Mobile ID Type

The type of Mobile ID being used by the LMU:

- 0 – OFF
- 1 – Electronic Serial Number (ESN) of the LMU
- 2 – International Mobile Equipment Identifier (IMEI) or Electronic Identifier (EID) of the wireless modem
- 3 – International Mobile Subscriber Identifier (IMSI) of the SIM card (GSM/GPRS devices only)
- 4 – User Defined Mobile ID
- 5 – Phone Number of the mobile (if available)
- 6 – The current IP Address of the LMU

Authentication Length

The number of bytes to follow which contain the Authentication word; normally this value is 4

Authentication

The Authentication word is used with an outbound message to gain access to the LMU if authentication is enabled.

Routing Length

The number of bytes to follow which contain the Routing field. If present, this value is user defined and can be up to 8.

Routing

The Routing field is up to 8 bytes long and can be used by the Application to store identity information of the originating user for requests made to the LMU from an application subsystem. It is only used in outbound messages. If the Routing field is present in a request received by the LMU, the LMU will include the same field, byte for byte, in the response sent back to the Application. The Application can then route the response back to the originating user based on information received. See Appendix D for further definition and uses of the Routing Field.

Forwarding Length

The number of bytes to follow which contain the Forwarding field. If present, this value is normally 8.

Forwarding

The Forwarding field is 8 bytes long and is used by an LM Direct™ forwarding agent to manage source and destination addresses and ports. It is only used in outbound messages. If the Forwarding field is present in a request received by the LMU, the LMU will include the same field, byte for byte, in the response sent back. The first four bytes of the forwarding field contain the forwarding address. The next two bytes contain the forwarding port. The following byte contains the forwarding protocol (TCP=6, UDP=17). The last byte is for the Forwarding Operation Type. The current operations are Forward (0x00), Proxy (0x01) and Forward with Lookup (0x02). See Appendix D for further definition and uses of the Forwarding field.

Response Redirection Length

The number of bytes in the following Response Redirection field. If present, this value is normally 6.

Response Redirection

The Response Redirection field is 6 bytes long and is used by an LM Direct™ recipient to redirect an acknowledged response back to the specified address and port instead of back to the sender. The first four bytes of the response redirection field contain the redirection address. The next two bytes contain the redirection port.

Message Header

The Message Header fields used in LM Direct™ are defined as follows:

Service Type

- 0 = Unacknowledged Request
- 1 = Acknowledged Request
- 2 = Response to an Acknowledged Request

Message Type

- 0 = Null message
- 1 = ACK/NAK message
- 2 = Event Report message
- 3 = ID Report message
- 4 = User Data message
- 5 = Application Data message
- 6 = Configuration Parameter message
- 7 = Unit Request message
- 8 = Locate Report message
- 9 = User Data with Accumulators message
- 10 = Mini Event Report message
- 11 = Mini User Message

Sequence Number

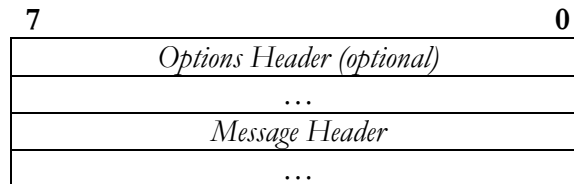
A 16-bit number used to uniquely identify a message. This number shall be initialized to 1 by the LMU on a cold boot and will be incremented in the LMU each time an inbound message is originated by the LMU. The LMU remembers its current Sequence Number during sleep. Eventually the Sequence Number will rollover from 65535 to 1, skipping zero.

The Sequence Number is also used to identify acknowledgements and retransmissions. Responses to Acknowledged Requests shall contain the Sequence Number from the associated Acknowledged Request. A Server receiving from an LMU can also use the Sequence Number as a means to identify whether the received message is an original or a retransmitted copy, thus avoiding processing any duplicate reports from an LMU.

A Server can set the Sequence Number field to zero for all outbound messages sent to the LMU if it is not important that the LMU distinguish an original message from a retransmitted copy. However, if this distinction is necessary, the server can insert a changing non-zero value into the Sequence Number field of each new message sent to the LMU. The LMU will remember the last sequence number it received and will compare it to the non-zero sequence number for the new message. If different, it will process the message normally. If the same, it will not process the message and will return a NAK response with the 'ACK' field set to 7.

Null Message (Message Type 0)

The Null message is an empty message and has no Message Content field. It is primarily used by the LMU which periodically sends it to the Server to keep network firewalls open so the Server can initiate messages to the LMU at any time. Normally the Null Message is sent as an Unacknowledged Request.



Null Message Format

Acknowledge Message (Message Type 1)

Acknowledge messages are used by the LMU or Server to acknowledge or reject requests made over the LM Direct™ interface.

7	0
<i>Options Header (optional)</i>	
...	
<i>Message Header</i>	
...	
Type	
ACK	
spare	
App Version (byte 0)	
App Version (byte 1)	
App Version (byte 2)	

Acknowledge Message Format

Type (1 byte)

This field indicates the type of message being acknowledged (e.g., it will match the ‘Msg Type’ field of the Request message)

ACK (1 byte)

- 0 – ACK (operation successful)
- 1 – NAK (operation failed, no reason)
- 2 – NAK (operation failed, not a supported message type)
- 3 – NAK (operation failed, not a supported operation)
- 4 – NAK (operation failed, unable to pass to serial port)
- 5 – NAK (operation failed, authentication failure)
- 6 – NAK (operation failed, Mobile ID look-up failure)
- 7 – NAK (operation failed, non-zero sequence number same as last received message)

Spare (1 byte)

This field is not used and should be set to zero.

App Version (3 bytes)

This 3-byte is used by the LMU to indicate its three-character application software version identifier (i.e. “60g”). This field should be set to all zeroes when the message is originated by the Server.

Event Report Message (Message Type 2)

Event Report messages are initiated by the LMU and are generated by the LMU's Programmable Event Generator (PEG). They can be either Acknowledged or Unacknowledged Requests. The Server should respond to an Acknowledged Event Report Request with an Acknowledge message. Event reports have the following structure:

15	0
<i>Options Header (optional)</i>	...
...	...
<i>Message Header</i>	...
...	...
Update Time (msByte)	Update Time
Update Time	Update Time (lsByte)
Time of Fix (msByte)	Time of Fix
Time of Fix	Time of Fix (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude (lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Altitude (msByte)	Altitude
Altitude	Altitude (lsByte)
Speed (msByte)	Speed
Speed	Speed (lsByte)
Heading (msByte)	Heading (lsByte)
Satellites	Fix Status
Carrier (msByte)	Carrier (lsByte)
RSSI (msByte)	RSSI (lsByte)
Comm State	HDOP
Inputs	Unit Status
Event Index	Event Code
Accums	spare
Accum 0 (msByte)	Accum 0
Accum 0	Accum 0 (lsByte)
...	...
Accum 15 (msByte)	Accum 15
Accum 15	Accum 15 (lsByte)

Event Report Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Time of Fix (4 bytes)

The last known time of fix from the GPS satellites. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver, measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement

Longitude (4 bytes)

The longitude reading of the GPS receiver, measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement

Altitude (4 bytes)

The altitude reading of the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement

Speed (4 bytes)

The speed as reported by the GPS receiver, measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

Carrier (2 bytes)

The identifier of the Carrier/Operator the wireless modem is currently using

RSSI (2 bytes)

The received signal strength of the wireless modem in dBm. This value is signed in a 2's complement format.

Comm State (1 byte)

The current state of the wireless modem bit mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Voice Call is Active

Bit 5 – Roaming

Bit 6 – Not Used

Bit 7 – Not Used

HDOP (1 byte)

The GPS Horizontal Dilution of Precision - it is a unit-less value reported with a 0.1 lsb.

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Ignition

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Unit Status (1 byte)

Status of key modules within the unit:

Bit 0 – Memory Test results (0=OK, 1=Error)

Bit 1 – GPS Antenna Status (0=OK, 1=Error)

Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)

Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)

Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)

Bit 5 – Unused

Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)

Bit 7 – Unused

Event Index (1 byte)

The index number of the event that generated the report; values should range from 0-74. 255 represents a Real Time PEG Action request.

Event Code (1 byte)

The event code assigned to the report as specified by the event's Action Parameter

Accums (1 byte)

The number of accumulators being reported with the event report

Spare (1 byte)

Not used - Set to 0.

Accum 0 (4 bytes)

The 4-byte value of accumulator 0 if Accums > 0

Accum 1 (4 bytes)

The 4-byte value of accumulator 1 if Accums > 1

Accum 2 (4 bytes)

The 4-byte value of accumulator 2 if Accums > 2

Accum 3 thru Accum 14**Accum 15 (4 bytes)**

The 4-byte value of accumulator 15 if Accums > 15

ID Report Message (Message Type 3)

The ID Report message is sent by the LMU under three scenarios: (1) As a result of a PEG action (Acknowledged or Unacknowledged Request), (2) In response to a Unit Request message (Response) or (3) In response to an SMS “SENDTO” request (Unacknowledged Request). The Server should respond to an Acknowledged ID Report Request with an Acknowledge message. ID Reports have the following structure:

7	0
<i>Options Header (optional)</i>	
...	
<i>Message Header</i>	
...	
Script Version	
Config Version	
App Version (byte 0)	
App Version (byte 1)	
App Version (byte 2)	
Vehicle Class	
Unit Status	
Modem Selection	
Application ID	
Mobile ID Type	
Query ID (byte 0)	
...	
Query ID (byte 3)	
ESN (byte 0)	
...	
ESN (byte 7)	
IMEI (byte 0)	
...	
IMEI (byte 7)	
IMSI (byte 0)	
...	
IMSI (byte 7)	
MIN (byte 0)	
...	
MIN (byte 7)	
ICC-ID (byte 0)	
...	
ICC-ID (byte 9)	
Extension String(s)	

ID Report Message Format

Script Version (1 byte)

A user-defined value stored in S-121 intended to identify the version of the PEG script stored in the LMU

Config Version (1 byte)

A user-defined value stored in S-143 intended to identify the version of the configuration parameters stored in the LMU

App Version (3 bytes)

The LMU's three-character software version identifier (e.g., "60g")

Vehicle Class (1 byte)

A user-defined value stored in S-147 intended to identify the class of the vehicle in which the LMU is installed

Unit Status (1 byte)

Status of key modules within the unit:

Bit 0 – Memory Test results (0=OK, 1=Error)

Bit 1 – GPS Antenna Status (0=OK, 1=Error)

Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)

Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)

Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)

Bit 5 – Unused

Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)

Bit 7 – Unused

Modem Selection (1 byte)

A value stored in S-120 that identifies the LMU's current modem type configuration

Application ID (1 byte)

This field identifies the application (LMU), the communications technology (CDMA, GSM, iDEN, etc.) and the platform (H8). The App ID breakdown is as follows:

App ID:

This field identifies the application program firmware running on the device. Typically it is unique to a device hardware configuration.

Mobile ID Type (1 byte)

A value stored in S-145 that identifies the LMU's current Mobile ID type configuration

Query ID (4 bytes)

A value that identifies the request that initiated the generation of the ID Report message. This is normally only used for SMS initiated ID Reports. This field is zero for all other uses.

ESN (8 bytes, Packed BCD)

The Electronic Serial Number of the LMU in packed BCD form. The most significant digit is placed in the upper four bits of the first byte. Unused trailing digits (4-bits) are filled with all ones (1111).

IMEI (8 bytes, Packed BCD)

The Electronic Serial Number of the modem in the LMU in packed BCD form. The most significant digit is placed in the upper four bits of the first byte. Unused trailing digits (4-bits) are filled with all ones (1111).

IMSI (8 bytes, Packed BCD)

The Subscriber ID of the SIM used in the GSM/GPRS modem in packed BCD form. The most significant digit is placed in the upper four bits of the first byte. Unused trailing digits (4-bits) are filled with all ones (1111).

MIN (8 bytes, Packed BCD)

The phone number of the modem in the LMU (if available) in packed BCD form. The most significant digit is placed in the upper four bits of the first byte. Unused trailing digits (4-bits) are filled with all ones (1111).

ICC-ID (10 bytes, Packed BCD)

The serial number of the SIM used in the GSM/GPRS modem in packed BCD form. The most significant digit is placed in the upper four bits of the first byte. Unused trailing digits (4-bits) are filled with all ones (1111).

Extension Strings(0 to 'n' bytes)

This field is optional and will contain information fields in the form of one or more null-terminated ASCII strings. Each string begins with a tag which indicates the contents of the string. A tag consists of alpha-numeric ASCII characters followed by a colon (':') followed by the data portion of the string. The following are currently defined Extension Strings:

MEID:<CDMA Modem serial number>

PRI:<CDMA Modem PRI version>

Extension Strings are only present in ID Reports when they are necessary to add additional information to the ID Report message.

User Message (Message Type 4)

User Messages are used to pass user data from the Server to the Host serial port on the LMU and vice versa. The Host port must be configured to operate in this mode. User Messages sent by the LMU contain location and status fields that are not present in the User Messages sent by the Server. Note that the maximum number of user data bytes is 848. User Messages have the following structure:

	15	0
	<i>Options Header (optional)</i>	...

	<i>Message Header</i>	...

Present only when sent by LMU.	Update Time (msByte)	Update Time
	Update Time	Update Time (lsByte)
	Time of Fix (msByte)	Time of Fix
	Time of Fix	Time of Fix (lsByte)
	Latitude (msByte)	Latitude
	Latitude	Latitude (lsByte)
	Longitude (msByte)	Longitude
	Longitude	Longitude (lsByte)
	Altitude (msByte)	Altitude
	Altitude	Altitude (lsByte)
	Speed (msByte)	Speed
	Speed	Speed (lsByte)
	Heading (msByte)	Heading (lsByte)
	Satellites	Fix Status
	Carrier (msByte)	Carrier (lsByte)
	RSSI (msByte)	RSSI (lsByte)
	Comm State	HDOP
	Inputs	Unit Status
	User Msg Route	User Msg ID
	User Msg Length (msByte)	User Msg Length (lsByte)
User Msg (msByte)	...	
...	User Msg (lsByte)	

User Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Time of Fix (4 bytes)

The last known time of fix from the GPS satellites. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Altitude (4 bytes)

The altitude reading of the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement.

Speed (4 bytes)

The speed as reported by the GPS receiver measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected.

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

Carrier (2 bytes)

The identifier of the Carrier/Operator the wireless modem is currently using

RSSI (2 bytes)

The received signal strength of the wireless modem in dBm. This value is signed in a 2's complement format.

Comm State (1 byte)

The current state of the wireless modem bit mapped as follows

The current state of the wireless modem bit mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Voice Call is Active

Bit 5 – Roaming

Bit 6 – Not Used

Bit 7 – Not Used

HDOP (1 byte)

The GPS Horizontal Dilution of Precision - it is a unit-less value reported with a 0.1 lsb.

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Ignition

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Unit Status (1 byte)

Status of key modules within the unit:

- Bit 0 – Memory Test results (0=OK, 1=Error)
- Bit 1 – GPS Antenna Status (0=OK, 1=Error)
- Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)
- Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)
- Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)
- Bit 5 – Unused
- Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)
- Bit 7 – Unused

User Msg Route (1 byte)

This field was formerly the upper byte of the User Msg Type field. This new field defines which port the message originated from, or should be sent to: The possible values are:

- 0 – User 0 (Host) Port
- 2 – User 1 (Aux) Port
- 1,3-255 – Reserved for future use

User Msg ID (1 byte)

This field was formerly the lower byte of the User Msg Type field. This new field contains the user defined ID value assigned to the User Message. The value of this field is defined by S-136 or S166 depending on which port originated or is receiving the message.

User Msg Length (2 bytes)

The length, in bytes, of the user message that follows

User Msg (n Bytes)

The contents of the user message

Application Message (Message Type 5)

Application Messages are used to pass LMU application messages to and from the LMU. They provide a subclass of messages used to manage the LMU application. See Appendix E for a description of the possible Application messages types and formats supported by the LMU. Application Messages sent by the LMU contain location and status fields that are not present in the Application Messages sent by the Server. The maximum number of data bytes is 848. Application Messages have the following structure:

	15	0
	<i>Options Header (optional)</i>	...

	<i>Message Header</i>	...

Present only when sent by LMU.	Update Time (msByte)	Update Time
	Update Time	Update Time (lsByte)
	Time of Fix (msByte)	Time of Fix
	Time of Fix	Time of Fix (lsByte)
	Latitude (msByte)	Latitude
	Latitude	Latitude (lsByte)
	Longitude (msByte)	Longitude
	Longitude	Longitude (lsByte)
	Altitude (msByte)	Altitude
	Altitude	Altitude (lsByte)
	Speed (msByte)	Speed
	Speed	Speed (lsByte)
	Heading (msByte)	Heading (lsByte)
	Satellites	Fix Status
	Carrier (msByte)	Carrier (lsByte)
	RSSI (msByte)	RSSI (lsByte)
	Comm State	HDOP
	Inputs	Unit Status
	App Msg Type (msByte)	App Msg Type (lsByte)
	App Msg Length (msByte)	App Msg Length (lsByte)
App Msg (msByte)	...	
...	App Msg (lsByte)	

Application Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Time of Fix (4 bytes)

The last known time of fix from the GPS satellites. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Altitude (4 bytes)

The altitude reading of the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement.

Speed (4 bytes)

The speed as reported by the GPS receiver measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected.

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

Carrier (2 bytes)

The identifier of the Carrier/Operator the wireless modem is currently using

RSSI (2 bytes)

The received signal strength of the wireless modem in dBm. This value is signed in a 2's complement format.

Comm State (1 byte)

The current state of the wireless modem bit mapped as follows

The current state of the wireless modem bit mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Voice Call is Active

Bit 5 – Roaming

Bit 6 – Not Used

Bit 7 – Not Used

HDOP (1 byte)

The GPS Horizontal Dilution of Precision - it is a unit-less value reported with a 0.1 lsb.

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Ignition

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Unit Status (1 byte)

Status of key modules within the unit:

- Bit 0 – Memory Test results (0=OK, 1=Error)
- Bit 1 – GPS Antenna Status (0=OK, 1=Error)
- Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)
- Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)
- Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)
- Bit 5 – Unused
- Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)
- Bit 7 – Unused

App Msg Type (2 bytes)

The pre-defined ‘type’ value of the application message

App Msg Length (2 bytes)

The length, in bytes, of the application message that follows

App Msg (n Bytes)

The contents of the application message

Parameter Message (Message Type 6)

The Parameter message is used by the Server as a request to read or write one or more configuration parameters in the LMU. It is also used as a report by the LMU to report parameter values in response to a read request or as an indication of invalid parameters in response to a write request.

Parameter Update Begin Operation

A parameter update begin operation is used to notify the LMU that a configuration change is about to start. Receipt of this message will cause the PEG Trigger of Update Begin with a Trigger Parameter of 0 to be fired.

Parameter Write Operation:

A parameter write request message consists of one or more Parameter IDs and associated 2-byte length words. Each Parameter ID and length word is followed by one or more sets of values. Each value set consists of a 1-byte index (index=0 for non-array parameters) plus one or more bytes of data. For bit-mapped operations, the data includes a mask followed by the value; non-bit-mapped is just the value. The last Parameter ID in the write request message shall always be a zero and is followed by a length of zero. The LMU will respond with a message that includes any error indications and ending with a Parameter ID of zero and a length of zero. An error indication for an invalid or non-supported Parameter ID is the bad Parameter ID followed by a length of zero. An error indication for an invalid or non-supported Parameter Index is a Parameter ID followed by a length of 1 followed by the invalid Parameter Index.

Parameter Update End Operation

A parameter update end operation is used to notify the LMU that a configuration change has been completed. Receipt of this message will cause the PEG Trigger of Update End with a Trigger Parameter of 0 to be fired.

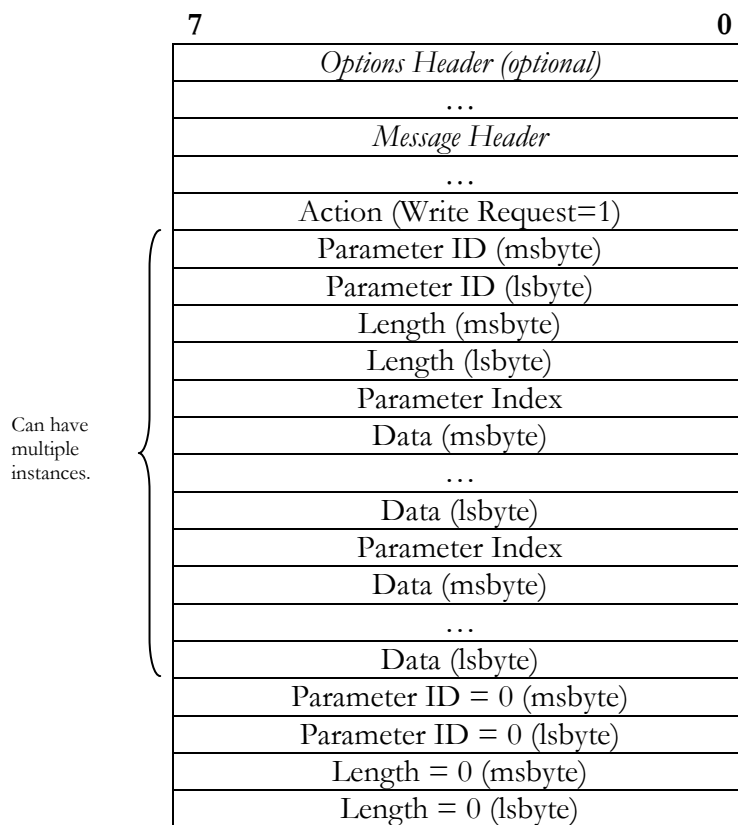
Parameter Read Operation:

A parameter read request message consists of one or more Parameter IDs and associated 2-byte length words. Each Parameter ID and length word is followed by one or more 1-byte parameter indexes (index=0 for non-array parameters). The last Parameter ID in the read request message shall always be a zero followed by a length of zero. The LMU will respond with a message that includes data from only the successful reads. An invalid or unsupported Parameter ID will be ignored and no entry for it will be made in the response message. A valid Parameter ID will result in a response message entry of a Parameter ID followed by a 2-byte length word followed by one or more value sets. Each value set consists of a 1-byte index (index=0 for non-array parameters) plus one or more bytes of data. Each value set corresponds to a requested index for that Parameter ID unless the index is invalid or not supported, in which case it will not be include in the response.

Each parameter request sent by the Server contains an Action byte that specifies whether the operation is a read or a write. Following the Action byte, one or more parameters may be specified. Each parameter is identified by a 2-byte Parameter ID and a 1-byte Index. Some parameters have their own Parameter ID while others share an ID with the others in their list and are thus also

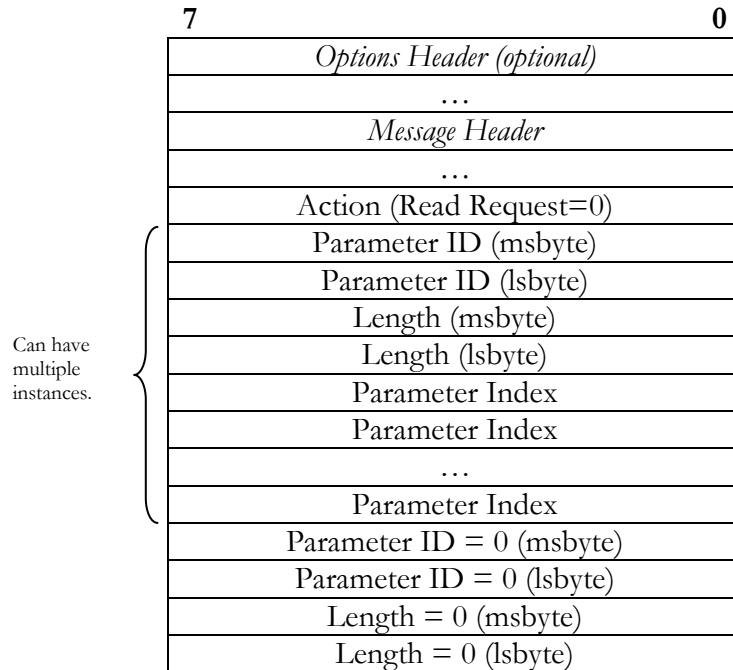
identified by their Index (position within the list). The Index is always 0 for those parameters that are not part of a list.

The format for a parameter request with a write action is shown below. Each Parameter ID is followed by a 2-byte length which specifies how many bytes follow to specify the parameters having this ID. Each individual parameter having this Parameter ID comprises a 1-byte Index and one or more data bytes making up the value of the parameter itself. The number of data bytes is known for each Parameter ID. The list of Parameters is always terminated by a Parameter ID of 0 followed by a length of 0.



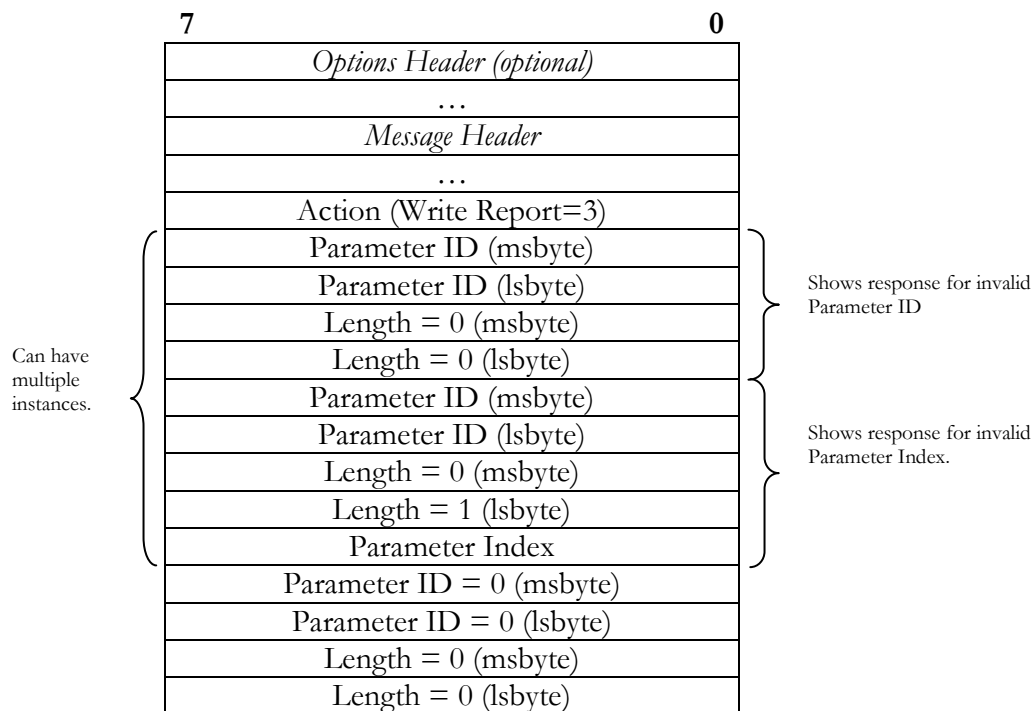
Parameter Write Request Format

The format for a parameter request with a read action is shown below. Each Parameter ID is followed by a 2-byte Length which specifies how many bytes follow to specify the parameters having this ID. Each individual parameter requested having this Parameter ID is identified by a 1-byte Parameter Index. The list of Parameters is always terminated by a Parameter ID of 0 followed by a length of 0.



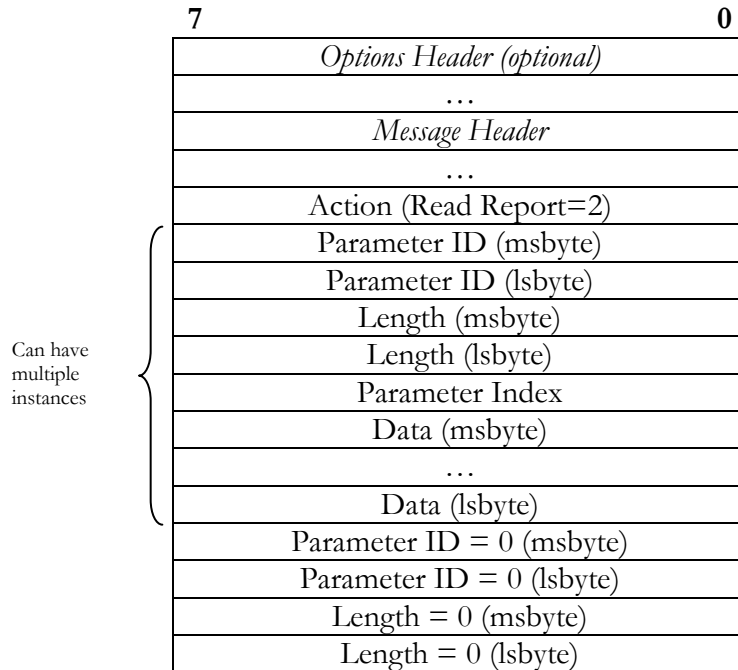
Parameter Read Request Format

The format for a parameter write report is shown below. Each Parameter ID is followed by a 2-byte length which specifies how many bytes follow to specify the parameters having this ID. Each individual parameter having this Parameter ID is comprised by a 1-byte Index and one or more data bytes making up the value of the parameter itself. The number of data bytes is known for each Parameter ID.



Parameter Write Report Format

The format for a parameter read report is shown below. Each Parameter ID is followed by a 2-byte length which specifies how many bytes follow to specify the parameters having this ID. Each individual parameter having this Parameter ID comprises a 1-byte Index and one or more data bytes making up the value of the parameter itself. The number of data bytes is known for each Parameter ID.



Parameter Read Report Format

The format for an Update Begin Request is shown below. This message contains only an action and the 0x0000 Parameter ID and 0x0000 Length values.

7	0
<i>Options Header (optional)</i>	
...	
<i>Message Header</i>	
...	
Action (Update Begin = 2)	
Parameter ID = 0 (msbyte)	
Parameter ID = 0 (lsbyte)	
Length = 0 (msbyte)	
Length = 0 (lsbyte)	

Update Begin Request Format

The format for an Update End Request is shown below. This message contains only an action and the 0x0000 Parameter ID and 0x0000 Length values

7	0
<i>Options Header (optional)</i>	
...	
<i>Message Header</i>	
...	
Action (Update End = 3)	
Parameter ID = 0 (msbyte)	
Parameter ID = 0 (lsbyte)	
Length = 0 (msbyte)	
Length = 0 (lsbyte)	

Update End Request Format

Action (1 byte)

This field is included in a Parameter Request to indicate whether the request is for a write operation or for a read operation:

- 0 – Read
- 1 – Write
- 2 – Update Begin
- 3 – Update End

Parameter ID (2 bytes)

This field holds the two-byte identifier of the parameter type being acted upon. Parameter Messages can contain one or more Param ID field along with associated length and index and value fields as necessary for Reads, Writes and Reports.

Length (2 bytes)

This field immediately follows each Param ID in the message to indicate how many bytes will follow to specify which Param Indexes and associated Param Values (if a write or report operation).

Parameter Index (1 byte)

This field indicates the position of the parameter within the list of parameters having the previous Param ID. Some parameters are the only member of the list so their Param Index is always 0. In a Parameter Read Request, there are two unique Param Index values that have special meaning: (a) A Param Index of 255 indicates to the LMU that it should return the number of members in its list for the Param ID and (b) A Param Index of 254 indicates to the LMU that it should return the values of every member of the list for the Param ID.

Data (1 or more bytes)

This field contains the parameter value and is present in Write Requests and Read Reports. The number of bytes for each Param Value field is dependent upon the Param ID which defines the type of parameter.

Unit Request Message (Message Type 7)

The Unit Request message is used by the Server to request a particular action of the LMU. The Action field defines the nature of the request and the Data8, Data16 and Data32 fields are used for some actions to provide additional information. The format for the Unit Request Message is shown below:

7	0
<i>Options Header (optional)</i>	
...	
<i>Message Header</i>	
...	
Action	
Data8	
Data16 (msbyte)	
Data16 (lsbyte)	
Data32 (msbyte)	
Data32 (2nd byte)	
Data32 (3rd byte)	
Data32 (lsbyte)	

Unit Request Message Format

Action (1 byte)

This field defines the type of unit request being made

Data8 (1 byte)

Used for some Unit Requests as defined in Table below. Otherwise set to 0

Data16 (2 bytes)

Used for some Unit Requests as defined in Table below. Otherwise set to 0

Data32 (4 bytes)

Used for some Unit Requests as defined in Table below. Otherwise set to 0

Unit Request	Action Code	Data8	Data16	Data32
Reboots the LMU	1			
Returns a Version Report in an App Msg	2			
Generates a Real-time PEG Action	3	PEG Action Code	PEG Action Modifier (lower 8-bits)	
Change the authentication password for the LM Direct™ Interface	4			Password
Returns a GPS Status Report in an App Msg	5			
Clears the PEG Event List	6			
Set the Aux Sleep Time. Time=32	7			Time (in seconds)
Generates PEG Special Trigger. Index=8	8	Trigger Modifier		
Returns an ID Report	9			
Returns a Locate Report	10	No. of Accumulators		
Returns a Message Statistics Report in an App Msg	11			
Reset Message Statistics	12			
Clear Error Flags	13			
Force Inbound and Maintenance DNS Lookups	14			
Request State Report	15			
User Flag Update	16	User Flag State	User Flag Mask (8 LSBs)	Special Trigger Number (8 LSBs)
Set Zone to Current Location and Enable Zone	17	Zone #	Zone Hysterisis (meters)	Zone Size (radius in meters)

Unit Request Data Field Usage

Locate Report Message (Message Type 8)

A Locate Report message is sent by the LMU in response to a request by the Application made with a Unit Request message (Action code 10). The number of accumulators reported in this message is specified in the Unit Request by the Application. Locate reports have the following structure:

15	0
<i>Options Header (optional)</i>	...
...	...
<i>Message Header</i>	...
...	...
Update Time (msByte)	Update Time
Update Time	Update Time (lsByte)
Time of Fix (msByte)	Time of Fix
Time of Fix	Time of Fix (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude(lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Altitude (msByte)	Altitude
Altitude	Altitude(lsByte)
Speed (msByte)	Speed
Speed	Speed (lsByte)
Heading (msByte)	Heading (lsByte)
Satellites	Fix Status
Carrier (msByte)	Carrier (lsByte)
RSSI (msByte)	RSSI (lsByte)
Comm State	HDOP
Inputs	Unit Status
Accums	spare
Accum 0(msByte)	Accum 0
Accum 0	Accum 0 (lsByte)
...	...
Accum 15 (msByte)	Accum 15
Accum 15	Accum 15(lsByte)

Locate Report Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Time of Fix (4 bytes)

The last known time of fix from the GPS satellites. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Altitude (4 bytes)

The altitude reading of the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement.

Speed (4 bytes)

The speed as reported by the GPS receiver measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected.

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

Carrier (2 bytes)

The identifier of the Carrier/Operator the wireless modem is currently using

RSSI (2 bytes)

The received signal strength of the wireless modem in dBm. This value is signed in a 2's complement format.

Comm State (1 byte)

The current state of the wireless modem bit mapped as follows

The current state of the wireless modem bit mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Voice Call is Active

Bit 5 – Roaming

Bit 6 – Not Used

Bit 7 – Not Used

HDOP (1 byte)

The GPS Horizontal Dilution of Precision - it is a unit-less value reported with a 0.1 lsb.

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Ignition

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Unit Status (1 byte)

Status of key modules within the unit:

- Bit 0 – Memory Test results (0=OK, 1=Error)
- Bit 1 – GPS Antenna Status (0=OK, 1=Error)
- Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)
- Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)
- Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)
- Bit 5 – Unused
- Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)
- Bit 7 – Unused

Accums (1 byte)

The number of accumulators being reported with the locate report

Spare (1 byte)

Not used. Set to 0

Accum 0 (4 bytes)

The 4-byte value of accumulator 0 if Accums > 0

Accum 1 (4 bytes)

The 4-byte value of accumulator 1 if Accums > 1

Accum 2 (4 bytes)

The 4-byte value of accumulator 2 if Accums > 2

Accum 3 thru Accum 14

Accum 15 (4 bytes)

The 4-byte value of accumulator 15 if Accums > 15

User Message with Accumulators (Message Type 9)

This type of User Message is used to pass user data to the Server from the Host serial port on the LMU including the current Accumulator values. The Host port must be configured to operate in this mode as well as set to use this message type over Type 4. The number of accumulators sent with each User Message is defined by the Inbound & Logged Event Report Contents. Note that the maximum number of user data bytes is 848. Also note that the number of accumulators reported might be automatically reduced for User Messages approaching the 848 byte limit. User Messages have the following structure:

15	0
<i>Options Header (optional)</i>	...
...	...
<i>Message Header</i>	...
...	...
Update Time (msByte)	Update Time
Update Time	Update Time (lsByte)
Time of Fix (msByte)	Time of Fix
Time of Fix	Time of Fix (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude (lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Altitude (msByte)	Altitude
Altitude	Altitude (lsByte)
Speed (msByte)	Speed
Speed	Speed (lsByte)
Heading (msByte)	Heading (lsByte)
Satellites	Fix Status
Carrier (msByte)	Carrier (lsByte)
RSSI (msByte)	RSSI (lsByte)
Comm State	HDOP
Inputs	Unit Status
spare 1 (msByte)	spare 1 (lsByte)
Accums	spare 2
Accum 0 (msByte)	Accum 0
Accum 0	Accum 0 (lsByte)
...	...
Accum N (msByte)	Accum N
Accum N	Accum N (lsByte)
User Msg Route	User Msg ID
User Msg Length (msByte)	User Msg Length (lsByte)
User Msg (msByte)	...
...	User Msg (lsByte)

Present only when sent by LMU.

User Message with Accumulators Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Time of Fix (4 bytes)

The last known time of fix from the GPS satellites. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading of the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Altitude (4 bytes)

The altitude reading of the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement.

Speed (4 bytes)

The speed as reported by the GPS receiver measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected.

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

Carrier (2 bytes)

The identifier of the Carrier/Operator the wireless modem is currently using

RSSI (2 bytes)

The received signal strength of the wireless modem in dBm. This value is signed in a 2's complement format.

Comm State (1 byte)

The current state of the wireless modem bit mapped as follows

The current state of the wireless modem bit mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Voice Call is Active

Bit 5 – Roaming

Bit 6 – Not Used

Bit 7 – Not Used

HDOP (1 byte)

The GPS Horizontal Dilution of Precision - it is a unit-less value reported with a 0.1 lsb.

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Ignition

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Unit Status (1 byte)

Status of key modules within the unit:

- Bit 0 – Memory Test results (0=OK, 1=Error)
- Bit 1 – GPS Antenna Status (0=OK, 1=Error)
- Bit 2 – GPS Receiver Self-Test (0=OK, 1=Error)
- Bit 3 – GPS Receiver Tracking (0=Yes, 1=No)
- Bit 4 – Modem MIN Test results (0 = OK, 1 = Error)
- Bit 5 – Unused
- Bit 6 – GPS Exception Reported (0 = OK, 1 = Error)
- Bit 7 – Unused

Spare 1 (2 byte)

Not used. Set to 0

Accums (1 byte)

The number of accumulators being reported with the locate report

Spare 2 (1 byte)

Not used. Set to 0

Accum 0 (4 bytes)

The 4-byte value of accumulator 0 if Accums > 0

Accum 1 (4 bytes)

The 4-byte value of accumulator 1 if Accums > 1

Accum 2 (4 bytes)

The 4-byte value of accumulator 2 if Accums > 2

Accum 3 thru Accum 14

Accum 15 (4 bytes)

The 4-byte value of accumulator 15 if Accums > 15

User Msg Route (1 byte)

This field was formerly the upper byte of the User Msg Type field. This new field defines which port the message originated from, or should be sent to: The possible values are:

- 0 – User0 (Host) Port
- 2 – User1 (Aux) Port
- 1,3-255 – Reserved for future use

User Msg ID (1 byte)

This field was formerly the lower byte of the User Msg Type field. This new field contains the user defined 'ID' value assigned to the User Message. The value of this field is defined by S-136 or S166 depending on which port originated or is receiving the message.

User Msg Length (2 bytes)

The length, in bytes, of the user message that follows

User Msg (n Bytes)

The contents of the user message

Mini Event Report Message (Message Type 10)

Mini Event Report messages are initiated by the LMU and are generated by the LMU's Programmable Event Generator (PEG). They can be either Acknowledged or Unacknowledged Requests. The Server should respond to an Acknowledged Event Report Request with an Acknowledge message. Mini Event reports have the following structure:

15	0
<i>Options Header (optional)</i>	...
...	...
<i>Message Header</i>	...
...	...
Update Time (msByte)	Update Time
Update Time	Update Time (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude (lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Heading (msByte)	Heading (lsByte)
Speed	FixStat/nSats
Comm/Unit Status	Inputs
Event Code	Accums
Accum 0 (msByte)	Accum 0
Accum 0	Accum 0 (lsByte)
...	...
Accum 15 (msByte)	Accum 15
Accum 15	Accum 15 (lsByte)

Mini Event Report Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading of the GPS receiver, measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement

Longitude (4 bytes)

The longitude reading of the GPS receiver, measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement

Heading (2 bytes)

The heading value reported in degrees from true North

Speed (1 bytes)

The speed as reported by the GPS receiver, measured in kilometers per hour

FixStat/nSats (1 byte)

The current fix status of the GPS receiver and the number of satellites used in the GPS solution bitmapped as follows:

Bit 0 thru Bit 3 – number of satellites (0-15)

Bit 4 – Invalid Time

Bit 5 – Invalid Fix

Bit 6 – Last Known

Bit 7 – Historic

Comm/GPS State (1 byte)

The current state of the wireless modem bit-mapped as follows

Bit 0 – Available

Bit 1 – Network Service

Bit 2 – Data Service

Bit 3 – Connected

Bit 4 – Active Voice Call

Bit 5 - Roaming

The current GPS status is bit-mapped as follows:

Bit 6 – GPS Antenna Status (0=OK, 1=Error)

Bit 7 – GPS Receiver Tracking (0=Yes, 1=No)

Inputs (1 byte)

The current state of the inputs, bit mapped as follows:

Bit 0 – Input 0 (Ignition)

Bit 1 – Input 1

Bit 2 – Input 2

Bit 3 – Input 3

Bit 4 – Input 4

Bit 5 – Input 5

Bit 6 – Input 6

Bit 7 – Input 7

Event Code (1 byte)

The event code assigned to the report as specified by the event's Action Parameter

Accums (1 byte)

The number of accumulators being reported with the event report, 0 to 16.

Accum 0 (4 bytes)

The 4-byte value of accumulator 0 if Accums > 0

Accum 1 (4 bytes)

The 4-byte value of accumulator 1 if Accums > 1

Accum 2 (4 bytes)

The 4-byte value of accumulator 2 if Accums > 2

Accum 3 thru Accum 14

Accum 15 (4 bytes)

The 4-byte value of accumulator 15 if Accums > 15

Mini User Message (Message Type 11)

This type of User Message is used to pass user data to the Server from a serial port on the LMU with minimal overhead. In place of the standard Status fields (Lat, Lon, Speed, etc.) only the Time Tag is included in this message. The LMU must be configured to use this message type over Type 4 or Type 9. Ref. S-Reg 177, bit-0). Note that the maximum number of user data bytes is 848. Mini User Messages have the following structure:

15	0
<i>Options Header (optional)</i>	...
...	...
<i>Message Header</i>	...
...	...
Update Time (msByte)	Update Time
Update Time	Update Time (lsByte)
User Msg Route	User Msg ID
User Msg Length (msByte)	User Msg Length (lsByte)
User Msg (msByte)	...
...	User Msg (lsByte)

Mini User Message Format

Update Time (4 bytes)

The time tag of the message in seconds, referenced from Jan. 1, 1970

User Msg Route (1 byte)

This field was formerly the upper byte of the User Msg Type field. This new field defines which port the message originated from, or should be sent to: The possible values are:

0 – User0 (Host) Port

2 – User1 (Aux) Port

1,3-255 – Reserved for future use

User Msg ID (1 byte)

This field was formerly the lower byte of the User Msg Type field. This new field contains the user defined 'ID' value assigned to the User Message. The value of this field is defined by S-136 or S166 depending on which port originated or is receiving the message.

User Msg Length (2 bytes)

The length, in bytes, of the user message that follows

User Msg (n Bytes)

The contents of the user message

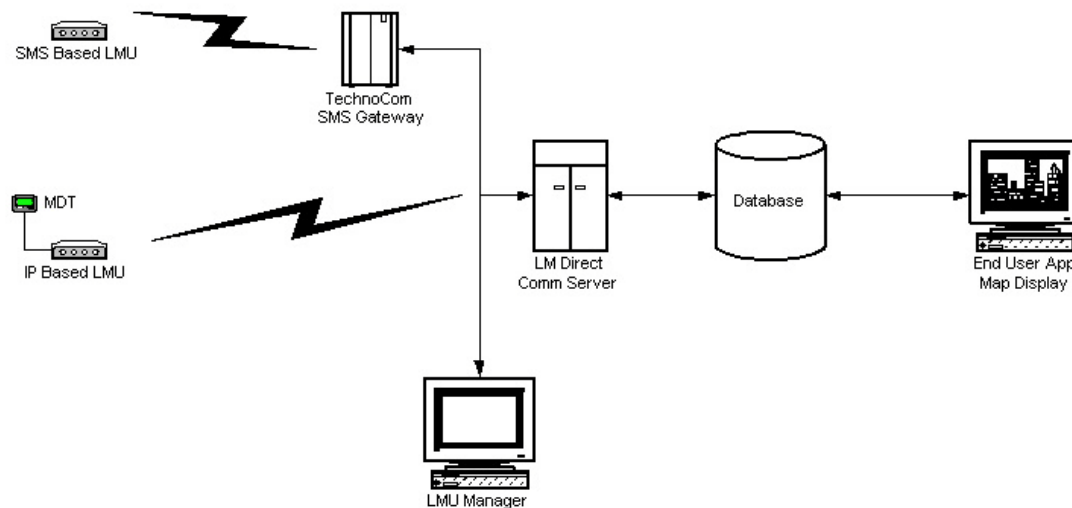
SECTION 3 – LM DIRECT™ MESSAGE FLOWS

Overview

This section describes the flow of various message transactions between the LMU and the backend system. For the purpose of these examples, it is assumed that the complete system comprises the following components:

- An LMU running on an IP based network (Mobile ID 999)
- A Mobile Data Terminal connected to LMU 999
- An LMU running on a Non-IP based network such as SMS (Mobile ID 777, Phone Number:888 555 1010)
- The CalAmp SMS Gateway which translates SMS messages to/from LM Direct™ messages
- A LM Direct™ Comm server
- A database (examples will be shown using SQL commands)
- A end user application/map display

A sample diagram of this system is shown below:



When needed, the examples below will assume the following:

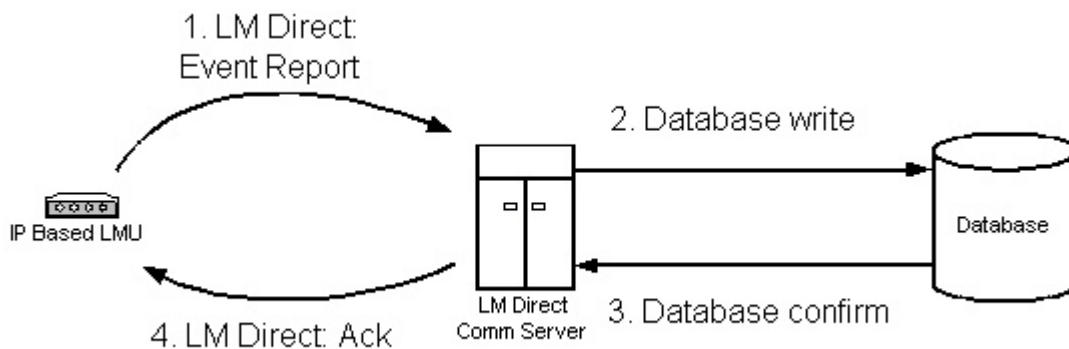
- The current IP address of the IP Based LMU is 166.147.5.5
- The phone number of the SMS based LMU is 888 555 1010
- The phone number of the SMS gateway is 999-555-1010
- The IP Address of the SMS gateway is 63.200.10.10
- The IP address of the LM Direct™ Comm Server is 63.200.10.11
- The IP address of LMU Manager is 63.200.10.12

Event Report Message Flow

Message Flow

This message flow description is centered on the IP based LMU sending Event Reports to the LM Direct™ Comm Server. It centers on the message transactions between the IP Based LMU, the LM Direct™ Comm Server and the Database.

The basic message flow is as follows:



1. The LMU creates an Event Report based on its PEG script and sends it to the LM Direct™ Comm Server
2. The LM Direct™ Comm Server receives, interprets and writes the contents of the Event Report to the database
3. The database confirms the write operation
4. The LM Direct™ Comm Server sends an Acknowledgement message to the LMU. This message is sent only if the Service Type of the event report is 0x01 (e.g., Acknowledged Request).

Message Contents

Shown below are two LM Direct™ messages that could be seen in the above sequence. Fields that are highlighted indicate values that must match between messages.

LM Direct™ Event Report	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x01
Message Type	0x02
Sequence Number	0x000A
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Event Index	0x01
Event Code	0x01
Accums	0x00
Spare	0x00

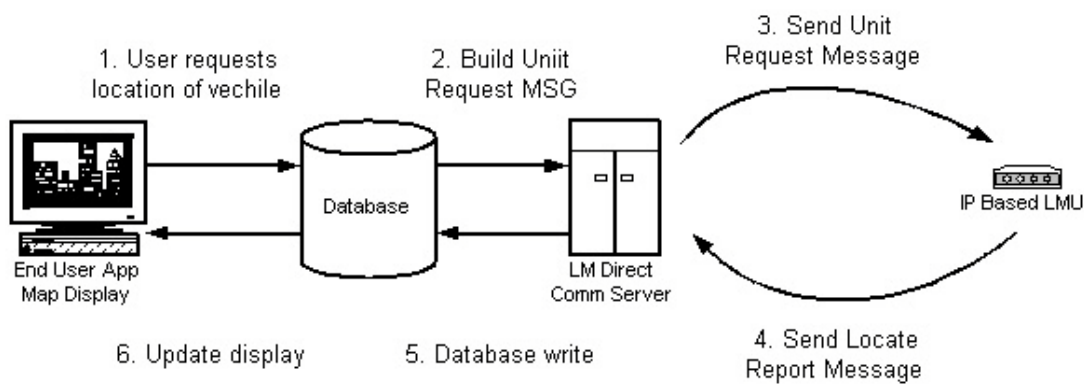
LM Direct™ ACK	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x02
Message Type	0x01
Sequence Number	0x000A
Type	0x02
ACK	0x00
App Version	0x000000

Unit Request Message Flow

Message Flow

This message flow description is centered on the end-user application requesting the current location of the vehicle. In response, the database and LM Direct™ Comm Server should send a Unit Request message with an action Type of 10 (Location Request) to the LMU in question. The LMU will respond with a Locate Report Message.

The basic flow is as follows:



1. Using the map display, the user requests the current location of a vehicle
2. The database and LM Direct™ Comm Server build a Unit Request message with a Locate Request Action
3. The LM Direct™ Comm Server sends the Unit Request message to the LMU based on the last known IP address of the LMU. The database should be set up to store and provide this IP address when needed
4. The LMU returns a Locate Report message
5. The LM Direct™ Comm Server receives, interprets and writes the contents of the Locate Report to the database
6. The Map Display is updated with the current local of the vehicle

Message Contents

Shown below are two LM Direct™ messages that could be seen in the above sequence.

LM Direct™ Unit Request	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x01
Message Type	0x07
Sequence Number	0x0000
Action	0x0A
Data8	0x00
Data16	0x0000
Data32	0x00000000

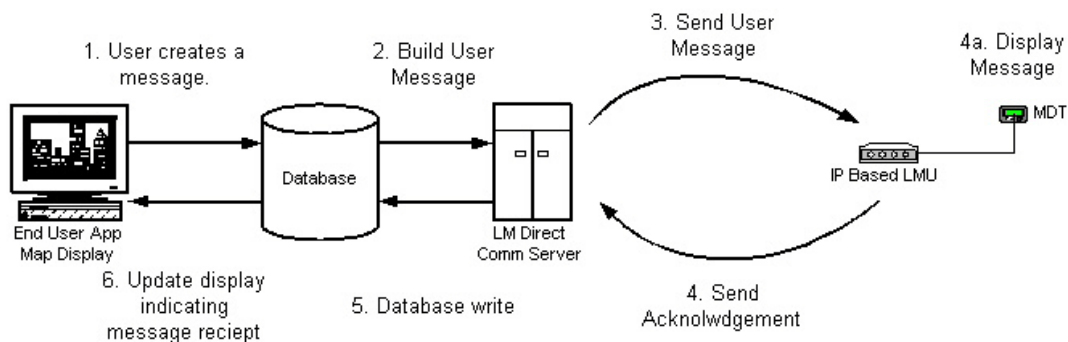
LM Direct™ Locate Report	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

User Message Flow - Outbound

Message Flow

This message flow description is centered on the end-user application attempting to send a message to the MDT attached to the IP based LMU's serial port. In response, the database and LM Direct™ Comm Server should build a User Message and pass it to the LMU in question. The LMU will respond with an Acknowledge Message indicating it attempted to deliver the message via the serial link. This assumes that ATS136 is set to 18 (i.e. 0x12)

The basic flow is as follows:



1. The User creates a message to be sent to the driver of the vehicle via his/her Mobile Data Terminal (e.g., "Please call in")
2. The database and LM Direct™ Comm Server build a User Message message with they payload created by the User.
3. The LM Direct™ Comm Server sends the User Message to the LMU based on the last known IP address of the LMU. The database should be set up to store and provide this IP address when needed
4. The LMU returns an Acknowledgement indicating it received the message. At the same time the LMU will push the user message out its serial port to the MDT. Note: If the MDT is off or not connected this message will be lost.
5. The LM Direct™ Comm Server receives, and interprets the acknowledgement
6. The Map Display is updated to indicate the message was received

Message Contents

Shown below are two LM Direct™ messages that could be seen in the above sequence:

LM Direct™ User Message	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x01
Message Type	0x04
Sequence Number	0x0000
User Message Route	0x00 0x12
User Message ID	
User Message Length	0x000F
User Message	0x506C6561 736C206361 6C6C20696E 0D

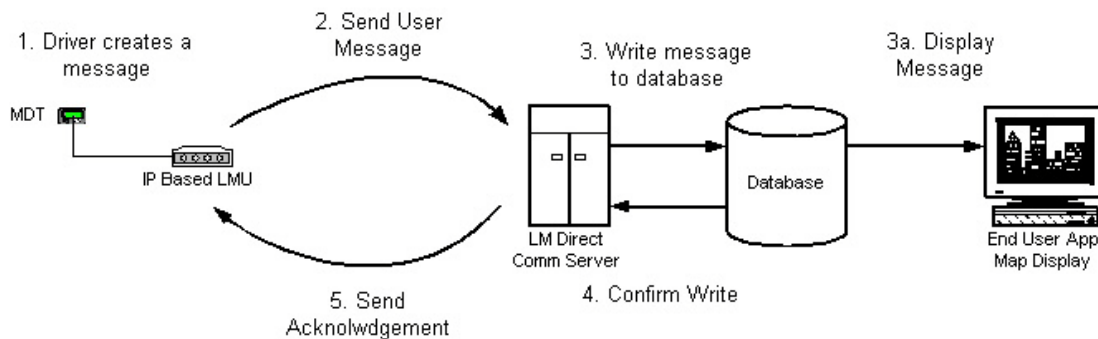
LM Direct™ ACK	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x02
Message Type	0x01
Sequence Number	0x0000
Type	0x04
ACK	0x00
App Version	0x363063

User Message Flow - Inbound

Message Flow

This message flow description is centered on the driver sending a message via his/her MDT. In this case the IP base LMU will create a User Message and attempt to send it to the LM Direct™ Comm Server. Once the message has been successfully pushed into the database, the Comm Server should send an ACK to the LMU indicating it received the message. This assumes that ATS136 is set to 18 (i.e. 0x12)

The basic message flow is shown below:



1. The Driver creates a message on his/her Mobile Data Terminal
2. The LMU packages and delivers this data as a User Message to the LM Direct™ Comm Server
3. The LM Direct™ Comm Server receives, interprets and writes this data to the database. The driver's message is pushed to the End User Application/Map Display
4. The database indicates a successful write operation
5. The LM Direct™ Comm Server sends an Acknowledgement message to the LMU

Message Contents

Shown below are two LM Direct™ messages that could be seen in the above sequence. Highlighted fields must match between messages:

LM Direct™ User Message	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x01
Message Type	0x04
Sequence Number	0x000B
Update Time	0x3FB54B40
Time of Fix	0x3FB54B40
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
User Message Route	0x00
User Message ID	0x12
User Message Length	0x000F
User Message	0x476F696E 67206F6E20 6C756E6368 0D

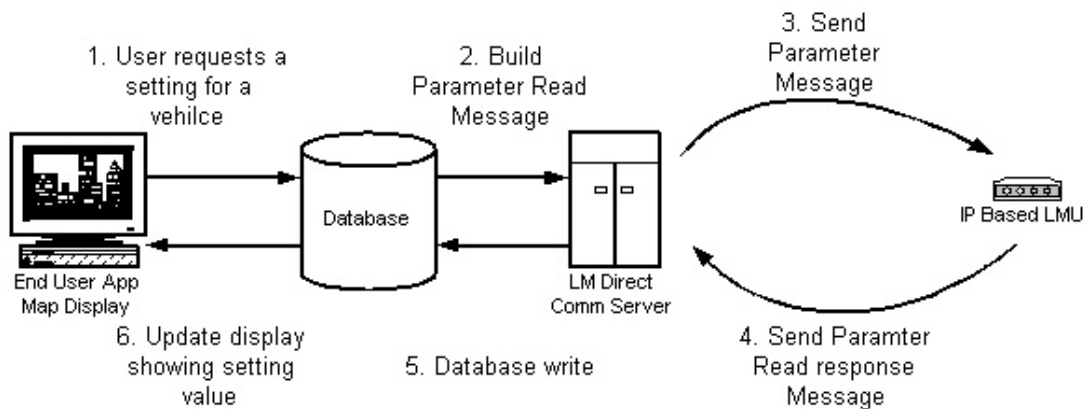
LM Direct™ ACK	
Field	Contents
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x02
Message Type	0x01
Sequence Number	0x000B
Type	0x04
ACK	0x00
App Version	0x000000

Parameter Message - Read

Message Flow

This message flow description is centered on the end-user application attempting to read one of the parameters within an LMU (e.g., the Moving Speed Threshold).

The basic flow is as follows:



1. The user requests to read a specific value (or group of values) from the LMU
2. The Database and LM Direct™ Comm Server build a Parameter Read Request Message
3. The LM Direct™ Comm Server sends the Parameter Message to the LMU based on the last known IP address of the LMU. The database should be set up to store and provide this IP address when needed
4. The LMU sends a Parameter Read Response Message to the Comm Server
5. The Comm server receives, interprets and writes the contents of this message to the database
6. The Map Display is updated with the requested settings

Message Contents

Shown below are two LM Direct™ messages that could be seen in the above sequence. Highlighted fields must match between messages.

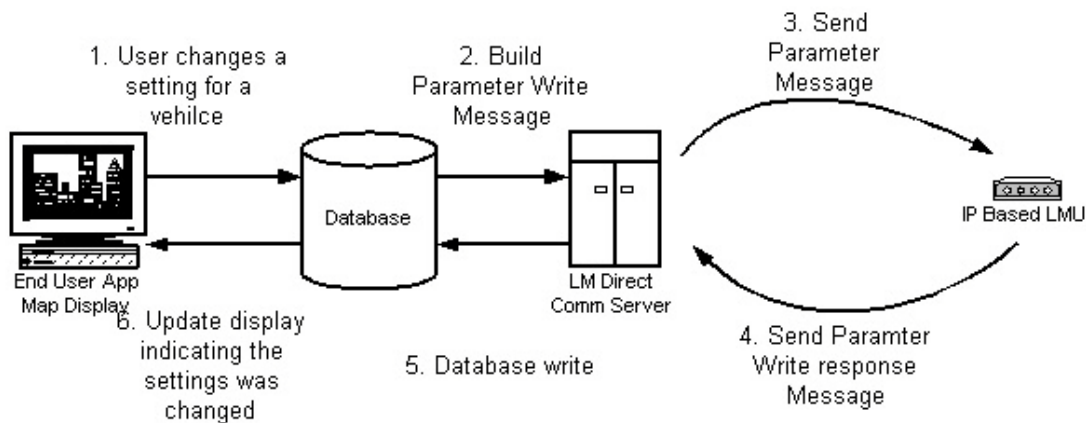
LM Direct™ Parameter Read Request		LM Direct™ Parameter Read Report	
Field	Contents	Field	Contents
Options Byte	0x83	Options Byte	0x83
Mobid ID Length	0x02	Mobid ID Length	0x02
Mobid ID	0xF999	Mobid ID	0xF999
Mobid ID Type Length	0x01	Mobid ID Type Length	0x01
Mobid Type	0x01	Mobid Type	0x01
Service Type	0x01	Service Type	0x02
Message Type	0x06	Message Type	0x06
Sequence Number	0x0000	Sequence Number	0x0000
Action	0x00	Action	0x02
Parameter ID	0x040B	Parameter ID	0x040B
Length	0x0001	Length	0x03
Parameter Index	0x00	Parameter Index	0x00
Parameter ID	0x0000	Data	0x0086
Length	0x0000	Parameter ID	0x0000
		Length	0x0000

Parameter Message - Write

Message Flow

This message flow description is centered on the end-user application attempting to change one of the parameters within an LMU (e.g., the Moving Speed Threshold).

The basic flow is as follows:



1. The user requests to change a specific value (or group of values) for the LMU
2. The Database and LM Direct™ Comm Server build a Parameter Write Request Message
3. The LM Direct™ Comm Server sends the Parameter Message to the LMU based on the last known IP address of the LMU. The database should be set up to store and provide this IP address when needed.
4. The LMU sends a Parameter Write Response Message to the Comm Server
5. The Comm server receives, interprets and writes the contents of this message to the database
6. The Map Display is updated indicating the setting(s) has been changed

Message Contents

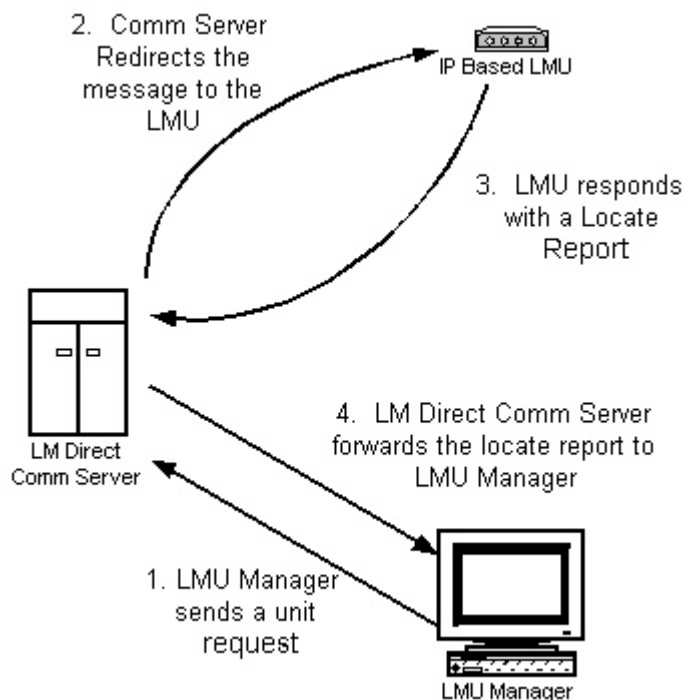
Shown below are two LM Direct™ messages that could be seen in the above sequence. Highlighted fields must match between messages.

LM Direct™ Parameter Write Request		LM Direct™ Parameter Write Report	
Field	Contents	Field	Contents
Options Byte	0x83	Options Byte	0x83
Mobid ID Length	0x02	Mobid ID Length	0x02
Mobid ID	0xF999	Mobid ID	0xF999
Mobid ID Type Length	0x01	Mobid ID Type Length	0x01
Mobid Type	0x01	Mobid Type	0x01
Service Type	0x01	Service Type	0x02
Message Type	0x06	Message Type	0x06
Sequence Number	0x0000	Sequence Number	0x0000
Action	0x01	Action	0x03
Parameter ID	0x040B	Parameter ID	0x0000
Length	0x0003	Length	0x0000
Parameter Index	0x00		
Data	0x00B3		
Parameter ID	0x0000		
Length	0x0000		

Message Forwarding

Message Flow

This message flow description is centered on LMU Manager using the LM Direct™ Comm Server to forward a Unit Request message to an LMU. In this sequence, the Comm server should be able to send the response back to LMU Manager.



1. LMU Manager creates a Unit Request intended for a specific LMU. LMU Manager sends this Unit Request with the forwarding feature enabled to the Inbound Port of the LM Direct™ Comm Server
2. The LM Direct™ Comm Server receives this message and swaps IP address data to send it to the LMU
3. The LMU Responds to the Comm Server with a Locate Report
4. Based on the forwarding information the LM Direct™ Comm Server sends the Locate Report to LMU Manager

Message Contents

Shown below are four LM Direct™ messages that could be seen in the above sequence. Highlighted fields indicate matching values in each message:

- Comm Server IP: 63.200.10.11 = 0x3FC80A0B
- LMU Manager IP: 63.200.10.12 = 0x3FX80A0C
- LMU IP: 166.147.5.5 = 0xA6930505
- Port 20500 = 0x5014
- Port 20510 = 0x501E

LMU Manager to LM Direct™ Unit Request	
Field	Contents
Source Address	0x3FC80A0C
Source Port	0x5014
Destination Address	0x3FC80A0B
Destination Port	5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0xA6930505
Forward Port	0x501E
Forward Protocol Type	0x11
Forward Operation	0x00
Service Type	0x01
Message Type	0x07
Sequence Number	0x0000
Action	0x0A
Data8	0x00
Data16	0x0000
Data32	0x00000000

LM Direct™ to LMU Unit Request	
Field	Contents
Source Address	0x3FC80A0B
Source Port	5014
Destination Address	0xA6930505
Destination Port	0x501E
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0x3FC80A0C
Forward Port	0x5014
Forward Protocol Type	0x11
Forward Operation	0x00
Service Type	0x01
Message Type	0x07
Sequence Number	0x0000
Action	0x0A
Data8	0x00
Data16	0x0000
Data32	0x00000000

LMU to LM Direct™ Locate Report	
Field	Contents
Source Address	0xA6930505
Source Port	0x501E
Destination Address	0x3FC80A0B
Destination Port	5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0x3FC80A0C
Forward Port	0x5014
Forward Protocol Type	0x11
Fwd Operation	0x00
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

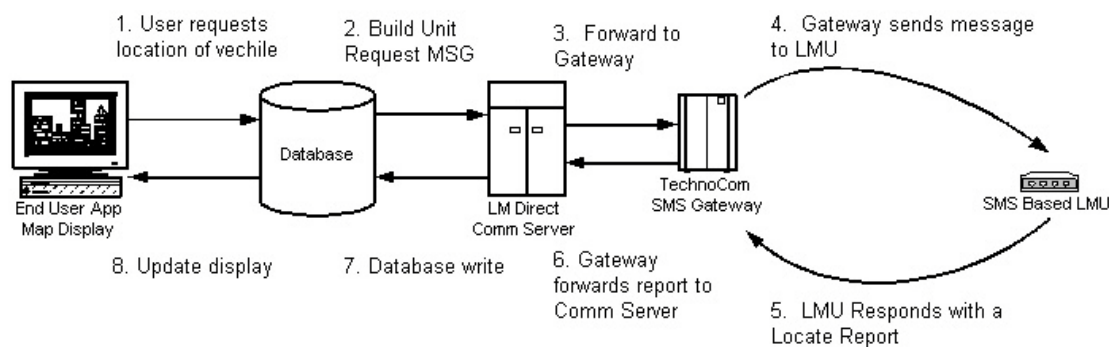
LM Direct™ to LMU Manager Locate Report	
Field	Contents
Source Address	0x3FC80A0B
Source Port	5014
Destination Address	0x3FC80A0C
Destination Port	0x5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0xA6930505
Forward Port	0x501E
Forward Protocol Type	0x11
Fwd Operation	0x00
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

Message Proxy - Outbound

Message Flow

An Outbound Proxy situation is a combination of a normal Unit Request and message forwarding. The Application is attempting to send a Unit Request message to an LMU via the Comm Server; however the LMU is on a non-IP based network (e.g., SMS). The LMU's address must now be mapped to the non-IP based address (i.e., phone number). The proper mapping for an address will be defined in the Specifications for the CalAmp Gateway system. For the purposes of this example, the address is the LMU's phone number and is treated as a 10 digit number. This 10 digit number will be mapped into the Address and Port bytes of the Forwarding Header. It will be padded with 0s to fill all 6 bytes.

The basic message flow is as follows:



1. Using the map display the user requests the current location of a vehicle
2. The database and LM Direct™ Comm Server build a Unit Request message with a Locate Request Action enabling the Proxy feature
3. The LM Direct™ Comm Server sends the Unit Request message to the Gateway adding the LMU's non-IP based address to the Proxy field
4. The Gateway forwards the message to the LMU
5. The LMU responds with a Locate Report
6. The Gateway Receives the Locate Report and forwards it to the Comm Server
7. The Comm server receives, interprets and writes the contents of this message to the database
8. The Map Display is updated indicating the new position of the vehicle

Message Contents

Shown below are four LM Direct™ messages that could be seen in the above sequence. Highlighted fields indicate matching values in each message:

- Comm Server IP: 63.200.10.11 = 0x3FC80A0B
- Gateway IP: 63.200.10.10 = 0x3FX80A0A
- Gateway Phone Number: 9995551010 = 0x000253C80122
- LMU Phone Number: 8885551010 = 0x0002119EBFA2
- Port 20500 = 0x5014
- Port 20510 = 0x501E

LM Direct™ To GatewayUnit Request		Gateway to LMU Unit Request	
Field	Contents	Field	Contents
Source Address	0x3FC80A0C	Source	0x000253C80122
Source Port	0x5014		
Destination Address	0x3FC80A0A	Destination	0x0002119EBFA2
Destination Port	0x501E		
Options Byte	0x93	Options Byte	0x93
Mobid ID Length	0x02	Mobid ID Length	0x02
Mobid ID	0xF999	Mobid ID	0xF999
Mobid ID Type Length	0x01	Mobid ID Type Length	0x01
Mobid Type	0x01	Mobid Type	0x01
Forward Length	0x08	Forward Length	0x08
Proxy	0x0002119EBFA2	Proxy	0x3FC80A0C 0x5014
Forward Protocol Type	0x11	Forward Protocol Type	0x11
Forward Operation	0x00	Forward Operation	0x00
Service Type	0x01	Service Type	0x01
Message Type	0x07	Message Type	0x07
Sequence Number	0x0000	Sequence	0x0000

LM Direct™ To GatewayUnit Request	
Field	Contents
Action	0x0A
Data8	0x00
Data16	0x0000
Data32	0x00000000

Gateway to LMU Unit Request	
Field	Contents
Number	
Action	0x0A
Data8	0x00
Data16	0x0000
Data32	0x00000000

LMU to Gateway Locate Report	
Field	Contents
Source	0x0002119EBFA2
Destination	0x000253C80122
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Proxy Length	0x08
Proxy	0x3FC80A0C 0x5014
Proxy Protocol Type	0x11
Forward Operation	0x00
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002

Gateway to LM Direct™ Locate Report	
Field	Contents
Source Address	0x3FC80A0A
Source Port	501E
Destination Address	0x3FC80A0C
Destination Port	0x5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Proxy Length	0x08
Proxy	0x0002119EBFA2
Proxy Protocol Type	0x11
Forward Operation	0x00
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002

LMU to Gateway Locate Report	
Field	Contents
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

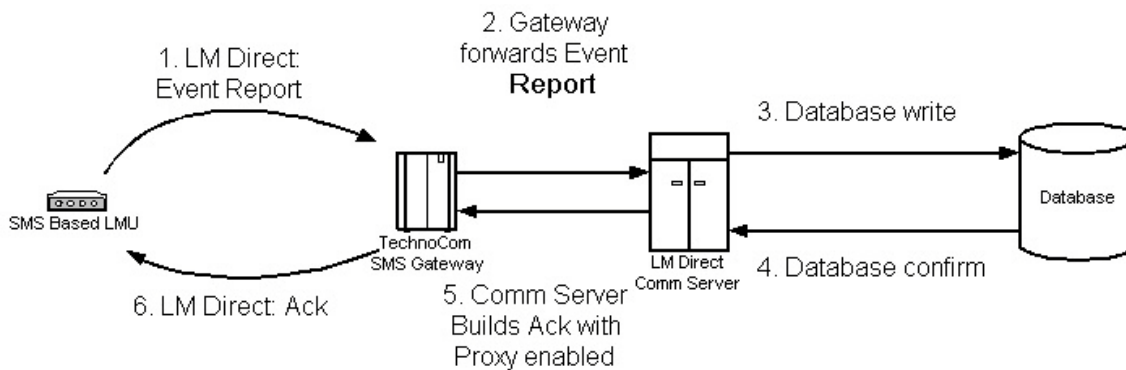
Gateway to LM Direct™ Locate Report	
Field	Contents
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

Message Proxy - Inbound

Message Flow

In an Inbound Proxy situation, a non-IP based LMU is attempting to deliver an Event Report to the LM Direct™ Comm Server via the CalAmp Gateway. This message flow assumes that the LMU is still using an Acknowledged message type so the LM Direct™ Comm Server must send an appropriate response (e.g., an ACK). In order to respond properly, the Comm Server must enable the Proxy settings on the outbound ACK.

The basic message flow is as follows:



1. The LMU creates an Event Report based on its PEG script and sends it to the CalAmp Gateway
2. The Gateway receives the message, adds the appropriate proxy information and forwards it to the Comm Server
3. The LM Direct™ Comm Server receives, interprets and writes the contents of the Event Report to the database
4. The database confirms the write operation
5. The LM Direct™ Comm Server sends an Acknowledgement message to the CalAmp Gateway with the appropriate Proxy information. Remember that this message is sent only if the Service Type of the event report is 0x01 (e.g., Acknowledged Request).
6. The Gateway forwards the ACK to the non-IP based LMU

Message Contents

Shown below are four LM Direct™ messages that could be seen in the above sequence. Highlighted fields indicate matching values in each message:

- Comm Server IP: 63.200.10.11 = 0x3FC80A0B
- Gateway IP: 63.200.10.10 = 0x3FX80A0A
- Gateway Phone Number: 9995551010 = 0x000253C80122
- LMU Phone Number: 8885551010 = 0x0002119EBFA2
- Port 20500 = 0x5014
- Port 20510 = 0x501E

LM Direct™ Event Report	
Field	Contents
Source	0x0002119EBFA2
Destination	0x000253C80122
Options Byte	0x83
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Service Type	0x01
Message Type	0x02
Sequence Number	0x000A
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3

Gateway to LM Direct™ Comm Server Event Report	
Field	Contents
Source Address	0x3FX80A0A
Source Port	0x501E
Destination Address	0x3FC80A0B
Destination Port	0x5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Proxy Length	8
Proxy	0x0002119EBFA2
Proxy Protocol Type	0x11
Forward Operation	0x01
Service Type	0x02
Message Type	0x02
Sequence Number	0x000A

LM Direct™ Event Report	
Field	Contents
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Event Index	0x01
Event Code	0x01
Accums	0x00
Spare	0x00

Gateway to LM Direct™ Comm Server Event Report	
Field	Contents
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA19EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Event Index	0x01
Event Code	0x01
Accums	0x00
Spare	0x00

LM Direct™ to Gateway ACK		LM Direct™ ACK	
Field	Contents	Field	Contents
<i>Source Address</i>	0x3FC80A0B	<i>Source</i>	0x000253C80122
<i>Source Port</i>	0x5014		
<i>Destination Address</i>	0x3FX80A0A	<i>Destination</i>	0x0002119EBFA2
<i>Destination Port</i>	0x501E		
Options Byte	0x93	Options Byte	0x83
Mobid ID Length	0x02	Mobid ID Length	0x02
Mobid ID	0XF999	Mobid ID	0xF999

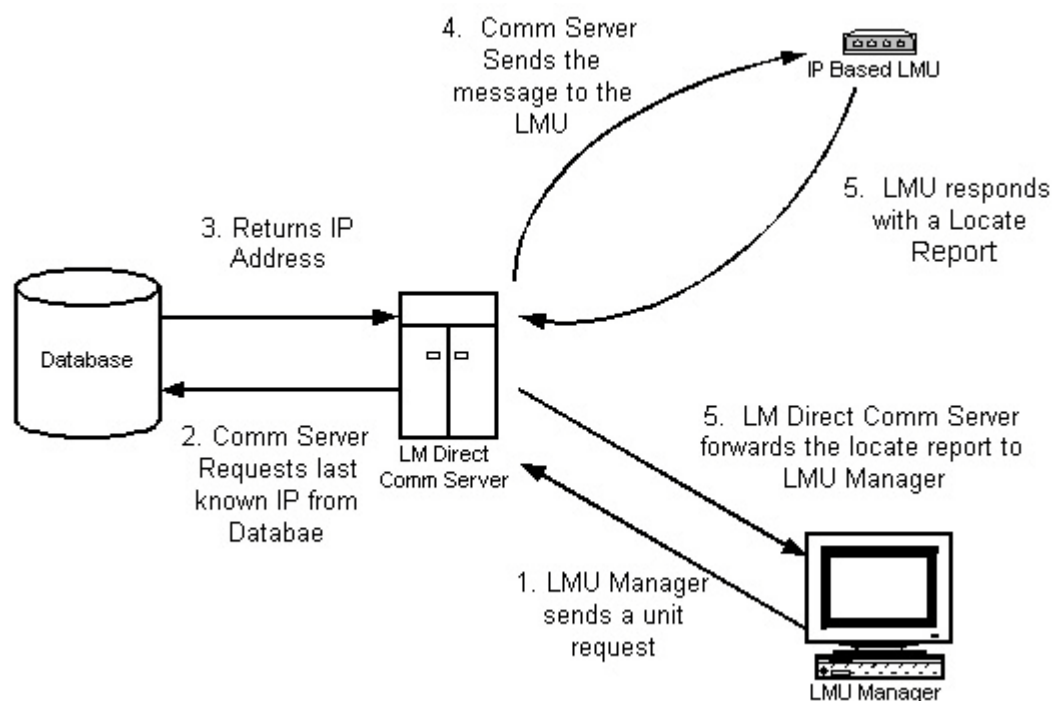
LM Direct™ to Gateway ACK		LM Direct™ ACK	
Field	Contents	Field	Contents
Mobid ID Type Length	0x01	Mobid ID Type Length	0x01
Mobid Type	0x01	Mobid Type	0x01
Proxy Length	8	Service Type	0x02
Proxy	0x0002119EBFA2	Message Type	0x01
Proxy Protocol Type	0x11	Sequence Number	0x000A
Proxy Operation	0x01	Type	0x02
Service Type	0x02	ACK	0x00
Message Type	0x01	App Version	0x000000
Sequence Number	0x000A		
Type	0x02		
ACK	0x00		
App Version	0x000000		

Message Forwarding With IP Lookup

Message Flow

Message forwarding with IP Lookup is similar to the basic Message Forwarding feature. The one key difference is that the LMU Manager program does not know the IP address of the destination LMU ahead of time; it only knows the Mobile ID. This type of request, therefore, requires that the Comm Server query the database for the last known IP address of the LMU in question. Assuming the IP is found, the Comm Server will then forward the message to the LMU. The incoming Locate Report should be handled as a normal forwarding message.

The basic message flow is as follows:



1. LMU Manager creates a Unit Request intended for a specific LMU. LMU Manager sends this Unit Request with the forwarding feature enabled to the Inbound Port of the LM Direct™ Comm Server
2. The LM Direct™ Comm Server receives this message and queries the data base for the IP address of the LMU
3. The database returns an IP address value to the Comm Server
4. The Comm Server sends the Unit Request to the LMU keeping the appropriate forwarding information
5. The LMU Responds to the Comm Server with a Locate Report

- Based on the forwarding information the LM Direct™ Comm Server sends the Locate Report to LMU Manager

Message Contents

Shown below are four LM Direct™ messages that could be seen in the above sequence. Highlighted fields indicate matching values in each message.

- Comm Server IP: 63.200.10.11 = 0x3FC80A0B
- LMU Manager IP: 63.200.10.12 = 0x3FX80A0C
- LMU IP: 166.147.5.5 = 0xA6930505
- Port 20500 = 0x5014
- Port 20510 = 0x501E

LMU Manager to LM Direct™ Unit Request		LM Direct™ to LMU Unit Request	
Field	Contents	Field	Contents
Source Address	0x3FC80A0C	Source Address	0x3FC80A0B
Source Port	0x5014	Source Port	5014
Destination Address	0x3FC80A0B	Destination Address	0xA6930505
Destination Port	5014	Destination Port	0x501E
Options Byte	0x93	Options Byte	0x93
Mobid ID Length	0x02	Mobid ID Length	0x02
Mobid ID	0xF999	Mobid ID	0xF999
Mobid ID Type Length	0x01	Mobid ID Type Length	0x01
Mobid Type	0x01	Mobid Type	0x01
Forward Length	0x08	Forward Length	0x08
Forward Address	0x00000000	Forward Address	0x3FC80A0C
Forward Port	0x501E	Forward Port	0x5014
Forward Protocol Type	0x11	Forward Protocol Type	0x11
Forward Operation	0x02	Forward Operation	0x02
Service Type	0x01	Service Type	0x01
Message Type	0x07	Message Type	0x07
Sequence Number	0x0000	Sequence Number	0x0000
Action	0x0A	Action	0x0A
Data8	0x00	Data8	0x00
Data16	0x0000	Data16	0x0000
Data32	0x00000000	Data32	0x00000000

LMU to LM Direct™ Locate Report	
Field	Contents
Source Address	0xA6930505
Source Port	0x501E
Destination Address	0x3FC80A0B
Destination Port	5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0x3FC80A0C
Forward Port	0x5014
Forward Protocol Type	0x11
Forward Operation	0x02
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

LM Direct™ to LMU Manager Locate Report	
Field	Contents
Source Address	0x3FC80A0B
Source Port	5014
Destination Address	0x3FC80A0C
Destination Port	0x5014
Options Byte	0x93
Mobid ID Length	0x02
Mobid ID	0xF999
Mobid ID Type Length	0x01
Mobid Type	0x01
Forward Length	0x08
Forward Address	0xA6930505
Forward Port	0x501E
Forward Protocol Type	0x11
Forward Operation	0x02
Service Type	0x02
Message Type	0x08
Sequence Number	0x0000
Update Time	0x3FB54B33
Time of Fix	0x3FB54B33
Latitude	0x13B2953C
Longitude	0xBA18EBA3
Altitude	0x00000000
Speed	0x00000002
Heading	0x00D1
Satellites	0x04
Fix Status	0x00
Carrier	0x0004
RSSI	0xFFBF
Comm State	0x0F
HDOP	0x23
Inputs	0x07
Unit Status	0x00
Accums	0x00
Spare	0x00

APPENDIX A – USING MOBILE ID

The Mobile ID and Mobile ID Type fields in the Options Header can be used on both mobile to server messages (e.g., Event Reports) and Server to mobile messages (e.g., Parameter Requests). For wireless networks with dynamic IP addressing schemes, the Mobile ID is required for mobile to Server messages to provide the Server with the Mobile ID of the mobile unit. For Server to mobile messages, Mobile ID is optional, though it is recommended. For each message direction, the Mobile ID serves a slightly different purpose.

On mobile to Server messaging, the Mobile ID in the Options header provides the static identifier (or name) of the unit. On Server to mobile messaging the Mobile ID is used as a filter; that is, the LMU or TetheredLocator will compare its own mobile ID and type with that of the incoming message. If they match, the message is processed; if neither match, the message is discarded. If either of these fields is not present in the Options header, they will not be considered in the match process.

A specific scenario where the Server to mobile filter would be used is as follows:

1. LMU 1 registers to a network and gets IP address 166.10.20.30
2. LMU 1 sends in an event report with MobileID 5555
3. LMU 1 deregisters from the network thereby releasing its IP address
4. LMU 2 registers to the network and gets IP 166.10.20.30
5. The Server software sends a dispatch request to MobileID 5555
6. LMU 2 receives dispatch request for Mobile ID 5555

In this case, if the Mobile ID filter was not in place, LMU 2 would have received instructions meant for LMU 1. Of course, this could cause major problems and confusions in fleet management operations.

APPENDIX B – USING AUTHENTICATION

The Authentication field in the Options Header allows the LMU to reject Server to mobile messages (e.g., Parameter Requests) based on an authentication key. The key is generated by an exclusive or operation between the LM Direct™ Interface's password and a 4 byte number based on the Mobile ID. This 4 byte number is created by taking the lower 9 digits of the Mobile ID.

If Mobile ID reporting has been disabled, the authentication key is generated by an exclusive or between the LM Direct™ Interface password and the current IP address of the LMU. The authentication feature is enabled and disabled based on the LM Direct™ Interface password. Authentication is enabled if the password is equal to or greater than 0x00000001. It is disabled if the password is 0x00000000.

Mobile ID Enabled Example

If the Mobile ID is 2957085360282 and the LM Direct™ Interface Password has been set to 21.12.21.12 then our Authentication key is created as follows:

1. Take the lower 9 digits of the mobile ID
Dec: 085360282
Hex: 0x5167E9A
2. Convert the password to a hexadecimal number
Dotted Dec: 21.12.21.12
Hex: 0x150C150C
3. XOR these two values to get the authentication key
Hex: 0x101A6B96

Mobile ID Disabled Example

If the current IP Address of the LMU is 166.147.64.158 and the LM Direct™ Interface Password has been set to 21.12.21.12 then the Authentication key is created as follows:

1. Take the IP Address of the LMU
Dotted Dec: 166.147.64.158
Hex: 0xA693409E
2. Take the unit's password
Dotted Dec: 21.12.21.12
Hex: 0x150C150C
3. XOR these two values to get the authentication key
Hex: 0x B39F5592

Appendix C — Processing the Options Header

The following pseudo-code snippet illustrates one possible way of processing the Mobile ID information from the Options Header of an LM Direct™ message.

Our basic steps are as follows:

Check the options byte to see if there is header information to process
 Check each bit in the options byte and process the data accordingly

```

/*Check the Options Byte of our incoming packet
 *remember bit 7 will be set if there
 *are options to process
 */
IF (PacketIN->Options & 0x80) THEN
{
/* Assign a Pointer to the top of the packet
 */
  ptrPacket = &PacketIN->Options;
  /* Create an offset to walk through the packet
  */
  packetOffset = 0;
  /* We look at each bit of the Options Byte
  * to determine which values we need to extract.
  .* We will walk through each bit using a left shift
  * bitwise operator
  */
  FOR (Options_Bit = 1; Options_Bit 128; OptionsBit <<= 1)
  {
    /* Check if the options bit has been set.
    */
    IF (PacketIN->Options & Options_Bit) THEN
    {
      /* move our offset to the next byte to process
      */
      packetOffset ++;
      /* Process Mobile ID
      */
    }
  }
  IF (Options_Bit == 1) THEN
  {
    /* BCDtoASCII (Destination, Source, Length)
    */
    BCDtoASCII (MyPacket->MobileID,
      &ptrPacket[packetOffSet+1],
      ptrPacket[PacketOffset])
  }
  ENDIF
  /* Process Mobile ID Type
  */
}
IF (Options_Bit == 2) THEN
{

```

```
        /* NOTE Mobile ID Type Length is
        * always 1 byte.
        */
        MyPacket->MobileIDType =
ptrPacket[packetOffset + 1];
    } ENDIF
    /* Process Authentication field
    */
IF (Options_Bit == 4) THEN
    {
    /* NOTE Authentication Length is
    * always 4 bytes.
    */
        MyPacket->Auth =
ptrPacket[packetOffset + 1];
    } ENDIF
    /* move the Offset past the length of data
    */
    packetOffset += ptrPacket[packetOffset];
    } ENDIF
} END LOOP
} END IF
/* Set index to the beginning of the location data
*/
packetOffset ++
```


APPENDIX D – ROUTING FIELD AND FORWARDING FIELD DEFINITIONS AND USES

The Routing Field can be included in a request made to an LMU to assist the Server or programs in routing the response from the LMU to the appropriate destination. For example, if an Application has several user terminals, the routing field can be used to identify the terminal making the request so that the response returned to the Application can be forwarded to the requesting terminal. The Routing field is optional and is not required for normal, single source application communications with the LMU. It is entirely up to the Application to define the use of the Routing field. Up to 8 bytes are available, which is sufficient to include an IP address and port number as a means of identifying the message source.

The Forwarding field is similar to the Routing field except that it has a specific purpose and formats. Like the Routing field, the Forwarding field included in a request from the Application is returned, byte-for-byte, in the Options header of the response sent back from the LMU. This field has two basic uses, Forward and Proxy.

The Forward Operation is intended for use by a forwarding agent located within the Application that can accept packets from another program such as LMU Manager and forward them to the LMU through the IP/UDP port already in use by the Application. Forwarding is often necessary to circumvent routing issues created by firewalls put in place by the wireless carriers. With these firewalls, typically only messages originated by the address/port destination of an LMU inbound message is permitted to pass through to the LMU. In other words, the only parties that can send to a given LMU are the parties that are currently receiving messages from the LMU. For the Forwarding operation this field comprises four sub fields: A 4-byte address followed by a 2-byte port followed by a 1-byte protocol type followed by a one byte Operation Type.

The Proxy Operation is intended to be used in conjunction with a CalAmp Gateway program. Gateways are programs that will take packets from an IP based network and forward them to LMUs on non-IP based networks (e.g. the 900MHz Point to Multi-Point LMUs). For the Proxy operation this field comprises three sub fields: A 6-byte address followed by a 1-byte protocol type followed by a one byte Operation Type.

APPENDIX E – APPLICATION MESSAGE TYPES AND FORMATS

Application Messages are a class of messages that are used for management of the LMU and for special functions supported by the Server. This section defines the Application Message types and formats. Each of the Application Message formats listed are contained within an Application Message (Message Type 5).

Types

IP Request	10
IP Report	11
jPOD™ –Vehicle Bus Data	30
Time Sync	50
Download ID Report	100
Download Authorization	101
Download Request	102
Download Update	103
Download Complete	104
AT Command	110
Version Report	111
GPS Status Report	112
Message Statistics Report	113
State Report	115
Geo-Zone Action Message	116
Geo-Zone Update Message	117
Capture Report	120

Formats

IP Request

This message is supported by the Server to allow external programs to request the Server provide the IP Address and UDP Port number from which the LMU last reported. The message contains the Mobile ID of the LMU. The Server will respond with an Application Message 'IP Report'. The format for this message is TBD.

IP Report

Sent by the Server in response to receiving an IP Request Application Message from an external program. This message provides the external program the IP Address and UDP Port number from which the requested LMU last reported. The format for this message is TBD.

jPOD™ Vehicle Bus Data

Used by the LMU-4200™ to relay data received from the jPOD™ Vehicle Bus Adapter.

15	0
Map ID	Event Code
Battery Potential (LSByte)	Battery Potential (MSByte)
Engine Coolant Temperature	Engine Speed - RPM (LSByte)
Engine Speed – RPM (MSByte)	Wheel Based Speed (LSByte)
Wheel Based Speed (MSByte)	Accelerator Pedal Position - %
Brake Pedal Position – On/Off	Total Vehicle Distance (LSByte)
Total Vehicle Distance	Total Vehicle Distance
Total Vehicle Distance (MSByte)	Engine Total Fuel Used (LSByte)
Engine Total Fuel Used	Engine Total Fuel Used
Engine Total Fuel Used (MSByte)	Diesel Particulate Filter Status
Reserved	Reserved
DM1	DM1
DM1	DM1
DM1	DM1

jPOD™ App Message Format

Map ID (1 byte)

Indicates a custom mapping definition of parameters to positions in the message so that the vehicle parameters can be extracted correctly at the backend.

In this implementation a value of 1 is selected for Map ID .

All subsequent field definitions are for Map ID 1.

Event Code (1 byte)

A bit mapped field of all the exceptions currently active. When values of certain parameters exceed predefined Thresholds the event code byte is updated according to the bit map definition below and the same 32 byte message with the dump of all the parameters is sent immediately(in addition to the periodic message).

S.No.	Parameter	Default Threshold	Bit set in Event code (LSBit is Bit0) when parameter exceeds threshold
1	Engine Coolant Temperature	100 degree C	Bit1
2	Engine Speed RPM	7000 RPM	Bit 2
3	Vehicle Speed	80 mph	Bit 3
4	Hard Acceleration detection	Acceleration pedal position > 80 %	Bit 4
5	Hard Braking detection	Brake pedal position > 80 %	Bit 5
6	DM1 changed	Not Applicable	Bit 6

J1939 Parameters (26 byte)

The units shown for the following fields are the typical implementation. Users should confirm all measurements with the vehicle manufacturer.

Battery Potential: Vehicle Battery Voltage measured in mV

Engine Coolant Temperature: Measured in Degrees Celsius with a <TBD> lsb

Engine Speed: Measured in RPM with a 1/8 RPM lsb

Wheel Based Speed: Measured in mm/s

Accelerator Position: Measured in percentage where 0 is not depressed with a 0.5% lsb

Brake Pedal Position: Measured as brake on or off

Total Vehicle Distance: Measured in <TBD>

Engine Fuel Total Used: This field is currently not supported

Diesel Particulate Filter Status: <TBD>

Time Sync

Used by the LMU to request time synchronization from the server (Request/Response byte equals 1) and used by the server to send time to the LMU (Request/Response byte equals 2). The ‘Time’ field is a 4-byte UTC (GMT) time (1-sec LSB, 0=12:00AM Jan 1, 1970). The ‘Spare’ fields should be set to 0. The ‘Time’ field is set 0 by the LMU when sending a request.

15	0
Request (1) / Response (2)	Spare
Spare	Spare
Time(msByte)	Time
Time	Time(lsByte)

Time Sync App Message Format

Download App Messages

These messages are reserved for use in downloading a new program into the LMU.

AT Command

Used by Server to send an AT Command to the LMU. Currently, the only AT Command accepted on this interface is a “SENDTO” command. This App Message holds the AT Command string in the ‘Msg’ field. The ‘Msg Length’ field contains the 2-byte length of the AT Command string. The ‘Msg Type’ field should have a value of 110. The LMU will respond with an Acknowledgement Message.

Version Report

Sent by the LMU in response to a Server initiated Unit Request for a Version Report. This App Message holds the version string in the ‘Msg’ field. The ‘Msg Length’ field contains the 2-byte length of the version string. The ‘Msg Type’ field should have a value of 111.

GPS Status Report

Sent by the LMU in response to a Server initiated Unit Request for a GPS Report. This App Message reports the current GPS output information which may be different from what is reported through normal report messages. It also contains the PRN, Azimuth, Elevation, SNR and usage indication for each satellite being tracked by LMU’s GPS receiver. The ‘App Msg Type’ field should have a value of 112 and the ‘App Msg Length’ field should be 124.

15	0
Time of Fix (msByte)	Time of Fix
Time of Fix	Time of Fix (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude(lsByte)
Longitude (msByte)	Longitude

Longitude	Longitude (lsByte)
Altitude (msByte)	Altitude
Altitude	Altitude(lsByte)
Speed (msByte)	Speed
Speed	Speed (lsByte)
Heading (msByte)	Heading (lsByte)
Satellites	Fix Status
HDOP	AGC
Antenna Status	GPS Power

The following fields repeat for each of 16 possible satellites.

PRN	Used
Elevation	SNR
Azimuth(msByte)	Azimuth (lsbyte)

GPS Status App Message Format

Time of Fix (4 bytes)

The time of fix from the GPS receiver. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading from the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading from the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Altitude (4 bytes)

The altitude reading from the GPS receiver measured in centimeters above the WGS-84 Datum, signed 2's complement.

Speed (4 bytes)

The speed as reported by the GPS receiver, measured in centimeters per second

Heading (2 bytes)

The heading value reported in degrees from true North

Satellites (1 byte)

The number of satellites used in the GPS fix solution

Fix Status (1 byte)

The current fix status of the GPS receiver bitmapped as follows:

Bit 0 – Predicted

Bit is set when the position update has a horizontal position accuracy estimate that is less than the Horizontal Position Accuracy Threshold defined in S-Register 142 (and the threshold is non-zero).

Bit 1 – Differentially Corrected

This bit is set when WAAS DGPS is enabled (S-Register 139) and the position has been differentially corrected

Bit 2 – Last Known

This bit is set when the current GPS fix is invalid but a previous fix's value is available.

Bit 3 – Invalid Fix

This bit is set only after a power-up or reset before a valid fix is obtained.

Bit 4 – 2D Fix

This bit is set when 3 or fewer satellites are seen/used in the GPS fix. (i.e. with 3 satellites or less, an altitude value cannot be calculated)

Bit 5 – Historic

This bit is set when the message has been logged by the LMU.

Bit 6 – Invalid Time

This bit is set only after a power-up or reset before a valid time-sync has been obtained.

HDOP (1 byte)

The GPS Horizontal Dilution of Precision: It is a unit-less value reported with a 0.1 lsb.

AGC (1 byte)

The current gain level of the AGC in the GPS receiver - Represented in percent (0-100). 0=not supported

Antenna Status (1 byte)

Electrical status of the GPS Antenna:

0=Active Antenna On and OK

1=Open Circuit

2=Short Circuit

3=Active Antenna off

4=Passive Antenna

5=Unknown

6-255 – Unused

GPS Power (1 byte)

0=GPS Receiver is Off

Nonzero = GPS Receiver is On

The following fields repeat for each of 16 possible satellites:

PRN (1 byte)

Identification number of the satellite, 1-32, 0 if this satellite set of fields is not used

Used (1 byte)

Indicates if this satellite is used in the location fix calculation, ASCII 'U'(0x55)=yes, ASCII <space>(0x20)=no

Elevation (1 byte)

Elevation of satellite above the horizon, degrees 0-90,

SNR (1 byte)

Satellite receive signal strength (C/No) in dBHz, 0-99;

Azimuth (2 bytes)

Azimuth of satellite from true North, degrees 0-359

Message Statistics Report

Sent by the LMU in response to a Server initiated Unit Request for a Message Statistics Report. This App Message reports various message counters in the LMU. The ‘App Msg Type’ field should have a value of 113 and the ‘App Msg Length’ field should be 30 bytes.

15	0
LocReq Count (msByte)	LocReq Count (lsByte)
Received User Msgs (msByte)	Received User Msgs (lsByte)
Inbound Reports (msByte)	Inbound Reports (lsByte)
Sent User Msgs (msByte)	Sent User Msgs (lsByte)
Spare1 (msByte)	Spare 1 (lsByte)
Logged Records (msByte)	Logged Records
Logged Records	Logged Records (lsByte)
Spare2 (msByte)	Spare2
Spare2	Spare2 (lsByte)
Spare3 (msByte)	Spare3
Spare3	Spare3 (lsByte)
Spare4 (msByte)	Spare4
Spare4	Spare4 (lsByte)
Spare5 (msByte)	Spare5
Spare5	Spare5 (lsByte)

Message Statistics App Message Format

LocReq Count (2 bytes)

The number of Locate Requests received by the LMU since the last power-up, wakeup, or stats reset

Received User Msgs (2 bytes)

The number of User Messages received by the LMU since the last power-up, wakeup, or stats reset

Inbound Reports (2 bytes)

The number of Inbound Messages (i.e., Event Reports) sent by the LMU since the last power-up, wakeup or stats reset

Sent User Msgs (2 bytes)

The number of User Messages sent by the LMU since the last power-up, wakeup, or stats reset.

SPARE1 (2 bytes)

This field is reserved for future use

Logged Records (4 bytes)

This field is the current number of Event Reports and User Messages stored in the LMU's log

SPARE2 (4 bytes)

This field is reserved for future use

SPARE3 (4 bytes)

This field is reserved for future use

SPARE4 (4 bytes)

This field is reserved for future use

SPARE5 (4 bytes)

This field is reserved for future use

State Report

15	0
Zone States (msByte)	Zone States
Zone States	Zone States (lsByte)
Environment State (msByte)	Environment State
Environment State	Environment State (lsbyte)
PEG Flags (msByte)	PEG Flags (lsByte)
Input States	Output States
Spare 1 (msByte)	Spare 1 (lsByte)
Spare 2	Accumulator Count
Accumulator 0 (msByte)	Accumulator 0
Accumulator 0	Accumulator 0 (lsByte)
...	...
Accumulator 15 (msByte)	Accumulator 15
Accumulator 15	Accumulator 15 (lsByte)

LMU State Report Format

Zone States (32 byte)

This is a bit mapped field indicating if the LMU is either inside (bit set) or outside (bit cleared) of a defined zone. If the zone is undefined, the bit is also cleared. The mapping is as follows:

Bit 0	Set = Inside Zone 0	Cleared = Outsize Zone 0
Bit 1	Set = Inside Zone 1	Cleared = Outsize Zone 1
Bit 2	Set = Inside Zone 2	Cleared = Outsize Zone 2
Bit 3	Set = Inside Zone 3	Cleared = Outsize Zone 3
...		
Bit 30	Set = Inside Zone 30	Cleared = Outsize Zone 30
Bit 31	Set = Inside Zone 31	Cleared = Outsize Zone 31

Environment State (32 byte)

This is a bit mapped field indicating the environment state of the LMU. The bit mapping is as follows:

Bit 0	Set = PEG Flag 0 set	Cleared = Peg Flag 0 cleared
Bit 1	Set = PEG Flag 1 set	Cleared = Peg Flag 1 cleared
Bit 2	Set = PEG Flag 2 set	Cleared = Peg Flag 2 cleared

Bit 3	Set = PEG Flag 3 set	Cleared = Peg Flag 3 cleared
Bit 4	Set = PEG Flag 4 set	Cleared = Peg Flag 4 cleared
Bit 5	Set = PEG Flag 5 set	Cleared = Peg Flag 5 cleared
Bit 6	Set = PEG Flag 6 set	Cleared = Peg Flag 6 cleared
Bit 7	Set = PEG Flag 7 set	Cleared = Peg Flag 7 cleared
Bit 8	Set = Inside Zone 0	Cleared = Outsize Zone 0
Bit 9	Set = Inside Zone 1	Cleared = Outsize Zone 1
Bit 10	Set = Inside Zone 2	Cleared = Outsize Zone 2
Bit 11	Set = Inside Zone 3	Cleared = Outsize Zone 3
Bit 12	Set = Inside Zone 4	Cleared = Outsize Zone 4
Bit 13	Set = Inside Zone 5	Cleared = Outsize Zone 5
Bit 14:	spare	
Bit 15:	spare	
Bit 16	Set = Above Speed 0	Cleared = Below Speed 0
Bit 17	Set = Above Speed 1	Cleared = Below Speed 1
Bit 18	Set = Above Speed 2	Cleared = Below Speed 2
Bit 19	Set = Above Speed 3	Cleared = Below Speed 3
Bit 20	Set = Day-of-Week 0 True	Cleared = Day-of-Week 0 False
Bit 21	Set = Day-of-Week 1 True	Cleared = Day-of-Week 1 False
Bit 22	Set = Day-of-Week 2 True	Cleared = Day-of-Week 2 False
Bit 23	Set = Day-of-Week 3 True	Cleared = Day-of-Week 3 False
Bit 24	Set = Ignition On	Cleared = Ignition Off
Bit 25	Set = Input 1 High	Cleared = Input 1 Low
Bit 26	Set = Input 2 High	Cleared = Input 2 Low
Bit 27	Set = Input 3 High	Cleared = Input 3 Low
Bit 28	Set = GPS Acquired	Cleared = No GPS
Bit 29	Set = Comm Acquired	Cleared = Comm Lost
Bit 30	Set = Log Active	Cleared = Log Inactive
Bit 31	Set = Moving	Cleared = Not Moving

PEG Flags (2 byte)

This is a bit mapped field indicating the state of each of the available PEG Flags. The bit mapping is as follows:

Bit 0	Set = PEG Flag 0 set	Cleared = Peg Flag 0 cleared
Bit 1	Set = PEG Flag 1 set	Cleared = Peg Flag 1 cleared
Bit 2	Set = PEG Flag 2 set	Cleared = Peg Flag 2 cleared
Bit 3	Set = PEG Flag 3 set	Cleared = Peg Flag 3 cleared
Bit 4	Set = PEG Flag 4 set	Cleared = Peg Flag 4 cleared
Bit 5	Set = PEG Flag 5 set	Cleared = Peg Flag 5 cleared
Bit 6	Set = PEG Flag 6 set	Cleared = Peg Flag 6 cleared
Bit 7	Set = PEG Flag 7 set	Cleared = Peg Flag 7 cleared
Bit 8	Set = PEG Flag 8 set	Cleared = Peg Flag 8 cleared
Bit 9	Set = PEG Flag 9 set	Cleared = Peg Flag 9 cleared
Bit 10	Set = PEG Flag 10 set	Cleared = Peg Flag 10 cleared
Bit 11	Set = PEG Flag 11 set	Cleared = Peg Flag 11 cleared
Bit 12	Set = PEG Flag 12 set	Cleared = Peg Flag 12 cleared
Bit 13	Set = PEG Flag 13 set	Cleared = Peg Flag 13 cleared
Bit 14	Set = PEG Flag 14 set	Cleared = Peg Flag 14 cleared
Bit 15	Set = PEG Flag 15 set	Cleared = Peg Flag 15 cleared

Input States (1 byte)

This is a bit-mapped field indicating the state of each of the LMU's available input lines, including those on the ioPOD. It does not apply to the A/D inputs. The bit mapping is as follows:

Bit 0	Set = Ignition On	Cleared = Ignition Off
Bit 1	Set = Input 1 High	Cleared = Input 1 Low (Or Used as Output)
Bit 2	Set = Input 2 High	Cleared = Input 2 Low (Or Used as Output)
Bit 3	Set = Input 3 High	Cleared = Input 3 Low
Bit 4	Set = Input 4 High	Cleared = Input 4 Low
Bit 5	Set = Input 5 High	Cleared = Input 5 Low
Bit 6	Set = Input 6 High	Cleared = Input 6 Low
Bit 7	Set = Input 7 High	Cleared = Input 7 Low

Outputs (1 bytes)

This is a bit-mapped field indicating the state of each of the LMU's available outputs, including those on the ioPOD. The bit mapping is as follows:

Bit 0	Set = Output 0 Set	Cleared = Output 0 Cleared
Bit 1	Set = Output 1 Set	Cleared = Output 1 Cleared (Or used as Input)
Bit 2	Set = Output 2 Set	Cleared = Output 2 Cleared (Or used as Input)
Bit 3	Set = Output 3 Set	Cleared = Output 3 Cleared
Bit 4	Set = Output 4 Set	Cleared = Output 4 Cleared
Bit 5	Set = Output 5 Set	Cleared = Output 5 Cleared
Bit 6	Set = Output 6 Set	Cleared = Output 6 Cleared
Bit 7:	spare	

Spare 1 (2 bytes)

This field is not used. It should be set to 0

Spare 1 (1 byte)

This field is not used. It should be set to 0

Accumulator Count (1 byte)

The number of accumulators being reported with the event report

Accum 0 (4 bytes)

The 4-byte value of accumulator 0 if Accums > 0

Accum 1 (4 bytes)

The 4-byte value of accumulator 1 if Accums > 1

Accum 2 (4 bytes)

The 4-byte value of accumulator 2 if Accums > 2

Accum 3 thru Accum 14

Accum 15 (4 bytes)

The 4-byte value of accumulator 15 if Accums > 15

Geo-Zone Action Message

15	0
Action Code	Response Code
Block Number	Spare
Start Index (msByte)	Start Index (lsByte)
Record Count (msByte)	Record Count (lsByte)
Major Version	Minor Version

Geo-Zone Action Message Format

Action Code (1 byte)

The value of this field indicates the action being performed (or requested) of the Geo-Zone memory space. The following values are defined:

- 0 – No Action
- 1 – Erase
- 2 – Read Records
- 3 – Update Complete
- 4 – Write Record (response only)
- 5 – Read Version
- 6 – Write Version
- 7 – 255 –Unknown/Reserved

Response Code (1 byte)

The value of this field indicates any error condition reported during one of the Actions. The following values are defined:

- 0 – No Error
- 1 – Write Verification Failure (resend previous request)
- 2 – Write Error
- 3 – Invalid Length on Update Requests or Read Requests
- 4 – Block Error on Update Request or Read Request
- 5 – 255 – Unknown/Reserved

Block Number (1 byte)

The memory block in which the Geo-Zone action is taking place. This value should normally be 0.

Spare (1 byte)

Reserved for future use. Set to 0.

Start Index (2 byte)

This field contains the record index (location) of the first record Requested or 0 for all other Actions

Record Count (2 byte)

The meaning of this field will change depending on the Action code. The actions that contain a value other than 0 are:

Read Version: The Number of records contained in the LMU(response only)

Read Records: The number of records to return in the update message

Major Version (1 byte)

The value of this field contains an unsigned integer value indicating the major version of the Geo-Zone configuration. This field is only valid for action types 6 and 7.

Minor Version (1 byte)

The value of this field contains an unsigned integer value indicating the minor version of the Geo-Zone configuration. This field is only valid for action types 6 and 7.

Geo-Zone Update Message

15	0
Block Number	Record Count
Start Index (msByte)	Start Index (lsByte)
Geo-Zone 0 Type	Geo-Zone 0 ID
Geo-Zone 0 Range (msByte)	Geo-Zone 0 Range (lsByte)
Geo-Zone 0 Latitude (msByte)	Geo-Zone 0 Latitude
Geo-Zone 0 Latitude	Geo-Zone 0 Latitude (lsByte)
Geo-Zone 0 Longitude (msByte)	Geo-Zone 0 Longitude
Geo-Zone 0 Longitude	Geo-Zone 0 Longitude (lsByte)
...	...
Geo-Zone N Type	Geo-Zone N ID
Geo-Zone N Range (msByte)	Geo-Zone N range (lsByte)
Geo-Zone N Latitude (msByte)	Geo-Zone N Latitude
Geo-Zone N Latitude	Geo-Zone N Latitude (lsByte)
Geo-Zone N Longitude (msByte)	Geo-Zone N Longitude
Geo-Zone N Longitude	Geo-Zone N Longitude (lsByte)

Geo-Zone Update Message Format

Block Number (1 byte)

The memory block in which the Geo-Zone action is taking place. This value should normally be 0.

Record Count (1 byte)

This contains the number of records included in this message. This field has a maximum value of 70.

Start Index (2 byte)

This field contains the record index (location) of the first record in this message

Geo-Zone Type (1 byte)

This field contains two values, the type of geo-zone point being defined and the Super Group to which the Geo-Zone point belongs. The type value is contained in bits 0 – 3. The super group ID is contained in bits 4 – 7. The available settings are:

Bits 0-3: Geo-Zone Type

0	reserved
1	PointZone
2	Polygon
3	Route
4-14	spare
15	reserved

Bits 4-7 Super-group ID

0-3	Super Group ID
4-15	reserved

Geo Zone ID (1 byte)

The value of this field is the reference number used for reporting Geo-Zone states within PEG. This field may range from 0 – 255. Note that point for a geo-zone must be grouped together and in order for route and polygon definitions. Zone IDs, must also be grouped under the Geo-Zone Super Group ID.

Geo Zone Range (2 bytes)

The value of this field will change based on the type of Geo-Zone being used. For Point Zones, this value defined the radius of the Point Zone. For Routes it defines the width of the route's corridor (eg the width of the road). This value is not used for Polygon Geo-Zones. The field value can range from 0 – 65535 meters.

Geo-Zone Latitude (4 bytes)

The latitude value of the Geo-Zone point being defined measured in degrees with a 1x10⁻⁷ degree lsb, signed 2's complement

Geo-Zone Longitude (4 bytes)

The longitude of the Geo-Zone point being defined measured in degrees with a 1x10⁻⁷ degree lsb, signed 2's complement

NOTE: A single update message can contain up to 70 Geo-Zone point definitions.

Capture Message

The Capture Message is sent by the LMU in response to the Flush Capture Buffer PEG Action. The Capture Message uses an Application Message Type of 120 and may contain one or more Capture Messages.

15	0
Timestamp (msByte)	Timestamp
Timestamp	Timestamp (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude (lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Accumulator N (msByte)	Accumulator N (lsByte)
Accumulator N+1 (msByte)	Accumulator N+1 (lsByte)
...	
Accumulator N+X (msByte)	Accumulator N+X (lsByte)

Repeats for Each Capture Report in the Capture Message

Capture Report Message Format (2 Byte Accumulators)

15	0
Timestamp (msByte)	Timestamp
Timestamp	Timestamp (lsByte)
Latitude (msByte)	Latitude
Latitude	Latitude (lsByte)
Longitude (msByte)	Longitude
Longitude	Longitude (lsByte)
Accumulator N (msByte)	Accumulator N
Accumulator N	Accumulator N (lsByte)
Accumulator N+1 (msByte)	Accumulator N+1 (lsByte)
Accumulator N+1	Accumulator N+1 (lsByte)
...	
Accumulator N+X (msByte)	Accumulator N+X
Accumulator N+X	Accumulator N+X (lsByte)

Repeats for Each Capture Report in the Capture Message

Capture Report Message Format (4 Byte Accumulators)

Timestamp(4 bytes)

The timestamp from the GPS receiver when the capture report was created. This value is reported in seconds from Jan. 1, 1970

Latitude (4 bytes)

The latitude reading from the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Longitude (4 bytes)

The longitude reading from the GPS receiver measured in degrees with a 1×10^{-7} degree lsb, signed 2's complement.

Accumulator N to Accumulator N+X (2 bytes or 4 bytes)

The value of Accumulator N using either a 2 byte or 4 byte width. The accumulator width is defined by the S-171 setting of the LMU. N and X are defined by the Flush Capture Buffer Action Modifier. Please refer to the PEG Programming Guide for details.

APPENDIX F – GEO-ZONE MESSAGING

Using Geo-Zones

Unlike most configuration items within the LMU, the Geo-Zone definitions are stored in Flash memory. From an external application stand point, this means that a specific sequence of events must be followed in order to read or write geo-zones.

For Polygon definitions, geo-zone point ordering is also important. Points must be listed in either clock-wise or counter clockwise order as you travel around the perimeter of the polygon. The first point definition of the polygon is also used as the last. Polygons points must appear in successive definitions, for instance, if you have a 6 point polygon and a 12 point polygon, the first 6 records must be polygon #1, the next 12 must be polygon #2. Polygon zones cannot share points between each other.

Point zones, since they are stand-alone entities may appear in any order and in any combination. Keep in mind that they cannot appear in the middle of a polygon definition as that would violate the above rule.

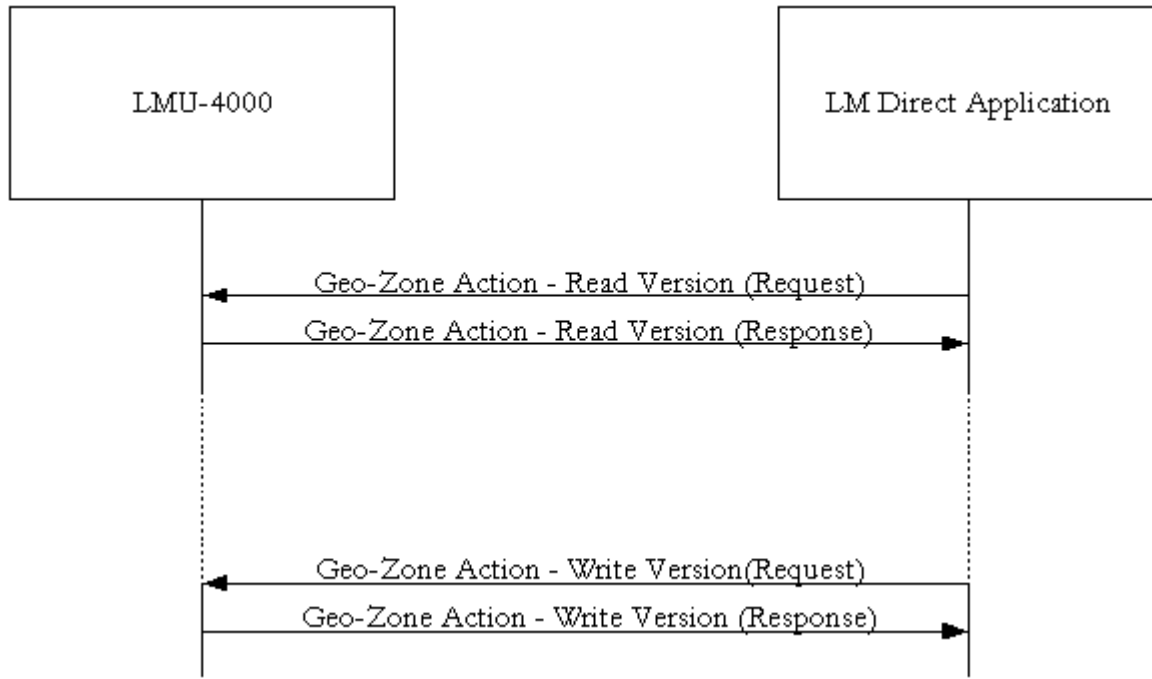
The LMU can store up to 5400 Geo-Zone records.

Geo-Zone Read & Write Version Requests

To read the current version of the Geo-Zone settings, an LM Direct™ Application will send a Geo-Zone action message with the Action Type set to 5 (Read Version Request). The LMU will respond with another Action message containing Action Type 7 as well as populating the Major and Minor version fields.

To write a new version the LM Direct™ App will send a Geo-Zone Action message of type 6 (Write Request) remembering to populate the Major and Minor version fields with the new version information. The LMU will respond with an Action message of type 4 (general response).

Both message flows are diagramed below:

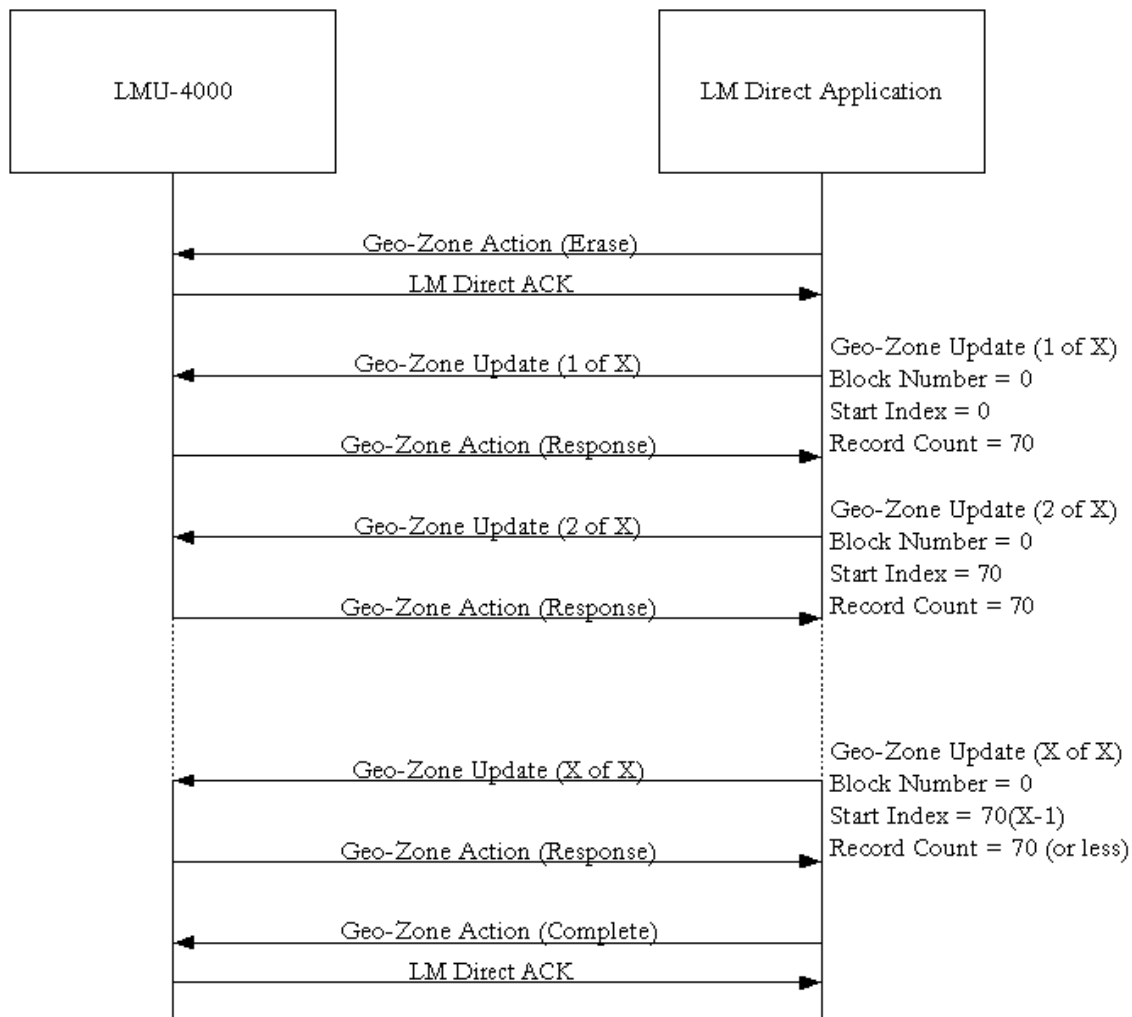


Geo-Zone Read/Write Version Process

Geo-Zone Write Requests

The basic write procedure for geo-zones is as follows.

1. Send an Erase Request
2. Wait for ACK
3. Send a Write Request
4. Wait for Write Complete Report
5. Repeat steps 3&4 until all points have been sent
6. Send a Write Complete Request
7. Wait for ACK



Geo-Zone Write Process

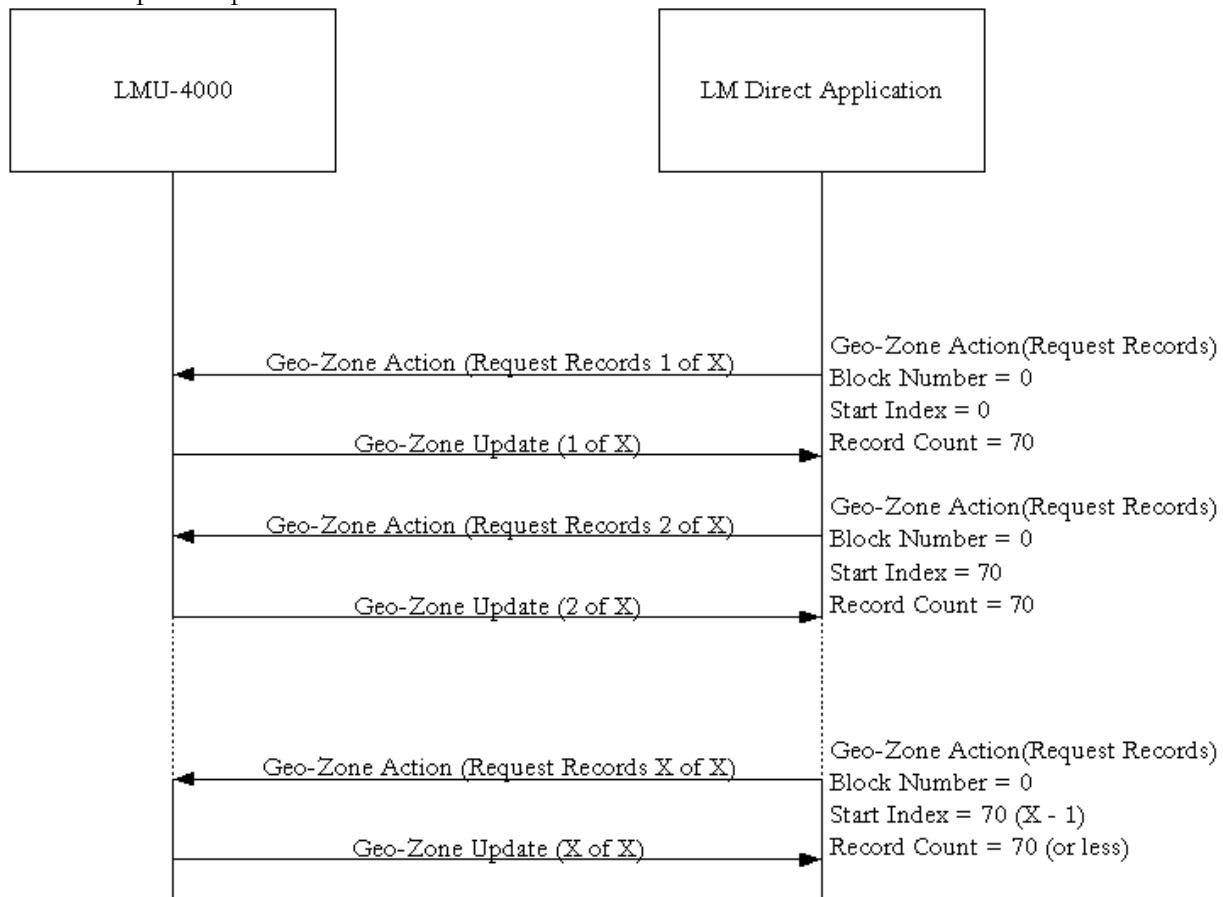
It is important to note that geo-zone processing is stopped when the Erase request is received. It is only restarted when the Write Complete request is received (or the LMU is power cycled).

Application developers must also keep in mind that each write request can hold a maximum of 70 Geo-Zone point definitions.

Geo-Zone Read Requests

The Read process for geo-zones is somewhat simpler since geo-zone processing does not need to be halted.

1. Send a Read Request
2. Wait for Update Message
3. Repeat steps 1&2 until all desired records are read



Geo-Zone Read Process

Again, it is important to note that Update messages will contain a maximum of 70 Geo-Zone. Also note that the LMU will return a Geo-Zone Action message if the read request contains an invalid Start Index, Record Count or Block Number.