LM Direct in GpsGate

CalAmps LM Direct is UDP interface to allow communication with CalAmp LMU devices without using LM Exchange middleware. The goal of this paper is to describe briefly how to transmit LM Direct information from a GpsGate server, and to do an example with a User Data Message (type 4) packet.

All LM Direct messages have the following parts

> Service Type

> Message Type

> Sequence Number

What follows the Sequence Number is determined by the Message Type.

Service Type is a byte indication of how the LMU should respond. 00 is unacknowledged, 01 means the LMU will acknowledge, and 02 means that the next message is a response to an acknowledgement.

Message type- has 14 options, only 4 of them can go from the server to the LMU

0 = Null message

1 = ACK/NAK message

2 = Event Report message

3 = ID Report message

4 = User Data message *can be sent server to LMU

5 = Application Data message *can be sent server to LMU

6 = Configuration Parameter message *can only be sent server to LMU, used to remotely read, and write parameters

7 = Unit Request message *can only be sent server to LMU, used to do remote actions, already supported by GpsGate

8 = Locate Report message

9 = User Data with Accumulators message

10 = Mini Event Report message

11 = Mini User Data message

12 = Mini Application message
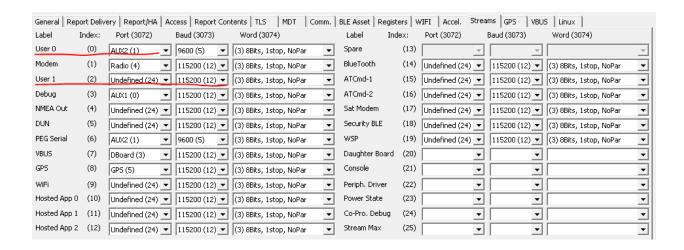
13 = Device Version message (MDT)

14 = Application message with accumulators

Of these, 7 is officially supported.

Sequence type - a 16-bit number that can be used to label a message. For our purposes it is only important so that repeated messages are ignored. There are few instances where this is advantageous. Default is 0000.

User Data Msg (type 4) is used to send data to serial ports attached to LMUs. There are many reasons to do this, like doing turn by turn instructions, sending dispatches, or controlling custom hardware. You are limited to 848 bytes of data. The format for this is the same as the above, plus the following four sections.

User Msg Route – 1 byte defining where the text is to go out at. This is defined in the LMU configuration. 0 is the host port (User 0, or GSD in LMU manager S-Register) and 2 defines the AUX port (User 1 or Aux serial in LMU manager S-Register). This is a little bit confusing because CalAmps documentation and software will use these terms interchangeably. You can also define it in different locations. This is an example of it defined in the Streams tab in LMU Manager Version 8.9.10.7

| General | Report Delivery | Report/HA | Access | Report Contents | TLS | MDT | Comm. | BLE Asset | Registers | WIFI | Accel. | Streams | GPS | VBUS | Linux |

Label	Index:	Port (3072)	Baud (3073)	Word (3074)	Label	Index:	Port (3072)	Baud (3073)	Word (3074)
User 0	(0)	AUX2 (1)	9600 (5)	(3) 8Bits, 1stop, NoPar	Spare	(13)			
Modem	(1)	Radio (4)	115200 (12)	(3) 8Bits, 1stop, NoPar	BlueTooth	(14)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
User 1	(2)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	ATCmd-1	(15)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
Debug	(3)	AUX1 (0)	115200 (12)	(3) 8Bits, 1stop, NoPar	ATCmd-2	(16)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
NMEA Out	(4)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Sat Modem	(17)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
DUN	(5)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Security BLE	(18)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
PEG Serial	(6)	AUX2 (1)	9600 (5)	(3) 8Bits, 1stop, NoPar	WSP	(19)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar
VBUS	(7)	DBoard (3)	115200 (12)	(3) 8Bits, 1stop, NoPar	Daughter Board	(20)			
GPS	(8)	GPS (5)	115200 (12)	(3) 8Bits, 1stop, NoPar	Console	(21)			
WiFi	(9)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Periph. Driver	(22)			
Hosted App 0	(10)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Power State	(23)			
Hosted App 1	(11)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Co-Pro. Debug	(24)			
Hosted App 2	(12)	Undefined (24)	115200 (12)	(3) 8Bits, 1stop, NoPar	Stream Max	(25)			

In this instance User 0 is set to the physical AUX 2 port and User 1 is unused.

This is how it is defined in the S-Registers

S registers 130 to 138 control GSD aka User 0 while Registers 160-169 Control AUX aka User 1. This is not to be confused with the physical AUX ports.

In this example User 0 is on the physical AUX 2 port, so to allow a message type 4 to work we would hook up the device to the LMUs Aux 2 port and define our Message 4 route to be 00 for the GSD/User 0 port.

User Msg ID -must match what is in the 6th Register for that rout (136 for User 0 or 166 for User 1).

This is a dual-purpose Register that gives an Event code for the detection of Seral Messages and as a match to make sure that incoming messages are supposed to be forwarded. Note this is in the Hex equivalent to the Decimal value in the S-Register.

User Message Length – a 2-byte hex indicator of the length of the message in bytes to be

delivered. For instance, the phrase "hello" has 5 bytes of data in ascii so the length would be 0005, while "hello world" has 11 and would be 000B

User Message Contents- The payload to be delivered.

Example. The setup above has User 0 set to Aux2, and an LCD display attached to Aux2 on the LMU. I want to send hello world to the display.

First, I must create the command.

The command will include the following

1. Service Type, I want to unit to acknowledge so I will use type 1. [0x01]
2. Message Type, I want to send out the Aux Port, so User Message (type 4) is selected. [0x04]
3. Sequence Type, I am not sending multiples of this command, so I leave it at 0. [0x00][0x00]
4. User Message Route, I am going to the Aux2 which is User 0. [0x00]
5. User Message ID, I am using User 0 which is GSD in the S-Registers. It is 7 above. [0x07]
6. User Message Length, "hello world" has 11 bytes.[0x00][0x0B]
7. User Message Contents, hellow world in
   hex.[0x68][0x65][0x6C][0x6C][0x6F][0x20][0x77][0x6F][0x72][0x6C][0x64]

When I put it together I get [0x01] [0x04] [0x00][0x00] [0x00] [0x07] [0x00][0x0B]
[0x68][0x65][0x6C][0x6C][0x6F][0x20][0x77][0x6F][0x72][0x6C][0x64]

I would put this command into the GpsGate terminal and send it over. The units display is set up to display any text that is transmitted to it and shows hello world.