NAVIGIL

Application Protocol Specification

Version 1, Revision 8

Navigil Ltd

This document describes the protocol, formatting and messages used by the TG2 software

Navigil Ltd • Klovinpellontie 1-3, Tower 2 • 02180 Espoo • Finland

GENERAL NOTE

THIS DOCUMENT CONTAINS INFORMATION ON NAVIGIL PRODUCTS. THE SPECIFICATIONS IN THIS DOCUMENT ARE SUBJECT TO CHANGE WITHOUT PRIOR NOTICE. THE RECIPIENT SHALL NOT TRANSFER, COPY, MODIFY, TRANSLATE, REVERSE ENGINEER, CREATE DERIVATIVE WORKS; DISASSEMBLE OR DECOMPILE THE PRODUCT OR OTHERWISE USE THE PRODUCT EXCEPT AS SPECIFICALLY AUTHORIZED. THE PRODUCT AND THIS DOCUMENT ARE PROVIDED ON AN "AS IS" BASIS ONLY AND MAY CONTAIN DEFICIENCIES OR INADEQUACIES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NAVIGIL LTD DISCLAIMS ALL WARRANTIES AND LIABILITIES. THE RECIPIENT UNDERTAKES FOR AN UNLIMITED PERIOD OF TIME TO OBSERVE NON-DISCLOSURE REGUIREMENTS REGARDING ANY INFORMATION AND DATA PROVIDED TO HIM IN THE CONTEXT OF THE DELIVERY OF THE PRODUCT. THIS GENERAL NOTE SHALL BE GOVERNED AND CONSTRUED ACCORDING TO FINNISH LAW.

Table of Contents

DOCUMENT HISTORY

Rev. #	Notes	Date
1	Initial release	2010-05-27
2	Contents of UNIT_REPORT extended	2011-03-19
3	POSITION_REPORT message defined ACKNOWLEDGEMENT, payload definition errors fixed	2011-06-07
4	CONSOLE_DATA message defined New report trigger values added	2011-08-11
5	POSITION_REPORT_2 message definition added Appendix A added for details of checksum calculation	2011-10-17
6	MEASUREMENT_DATA message added	2012-04-10
7	TG2_REPORT added	2012-04-20
8	Added message definitions for DIAGNOSTICS_REPORT, SNAPSHOT4, TRACKING_DATA and MOTION_ALARM Previously unused message header status bits replaced with flags Reason code added to CONN_OPEN payload Sleep time added to CONN_CLOSE payload Added descriptions of encoding schemes Changed formatting	2012-08-14

Complementary reading

The following Navigil reference documents are complementary reading for this document. All operating and firmware related documentation is also available at ***extranet.navigil.com***

Ref. #	Document Name

1 Overview

This document describes the protocol used by the standard applications of Navigil Track&Trace embedded platforms. The communication is performed by exchanging messages of predefined contents. Each message contains a common header part and message specific payload.

The protocol specification itself does not define strict limitations on the communication channel for the messages. The standard applications for the Track&Trace platforms provide implementations for server communication over UDP/IP or TCP/IP. Additionally, there is a possibility to deliver messages over SMS transport with Base64 encoding to avoid problems with binary SMS messages.

1.1 Features

1.1.1 Checksum

CRC-16 checksum is used to verify that the packets are received without errors.

CRC algorithm used with the application protocol is the CCITT CRC-16 with the generator polynomial $x^{16} + x^{12} + x^5 + 1$ and initial value of FFFFh.

See appendix A for examples on calculating the message checksum.

1.1.2 Timestamp

The message header and payloads of certain messages contain an absolute time reference called *timestamp*. All timestamps referred by this document have the same definition unless otherwise stated.

The timestamp used in the protocol is calculated as the number of seconds since 1.1.1970 00.00 UTC.

NOTE: Although the timestamp definition closely resembles the UNIX timestamp definition, they have one essential difference. Unix timestamp is tied to UTC time, i.e., leap seconds are taken into account in Unix time and therefore the Unix time is not incremented for leap seconds. However, as the protocol timestamp is a raw number of seconds since a defined origin, the application protocol timestamp is ahead of the Unix time by the number of UTC leap seconds (25 as of Aug 2012).

1.1.3 Message acknowledgement

As the protocol messages can be delivered over either a reliable or an unreliable connection, the protocol contains an acknowledgement to indicate that a message has been received.

If acknowledgements are used, the sender can store sent packets until the acknowledgement is received. If the acknowledgement is not received within a defined time window, the sender can resend the same packet.

As the packet loss can occur either in uplink or downlink direction, it is possible that the server received a message but the acknowledgement was not delivered back to the mobile unit. In this case the same message will be received by the server twice. This must not be interpreted by the server as an error, but the duplicate message can be discarded by the server, if applicable.

The main purpose of the acknowledgements is to compensate packet loss in an unreliable connection (e.g. UDP). Other types of transmission errors (checksum mismatch, duplicate packet, unrecognized message, etc.) can be indicated to the server as incorrectly received packets in the acknowledgement status code.

All received messages except the acknowledgement message itself must be acknowledged by the receiver. An exception to this rule is the DNA-bit in the message header bit flags (see description below).

1.2 Implementation rules

1.2.1 Byte order

All multibyte fields in the protocol messages are composed using little endian byte order (least significant byte is transmitted first). Although this is in contradiction with the standard IP network byte order (big endian), the byte order selection has been made because of all currently used and planned embedded platforms for Navigil Track&Trace software use little endian as their native byte order.

1.2.2 Byte alignment

All multibyte fields in the messages must be aligned into a byte boundary corresponding the size of the particular field, i.e., 16-bit words into 16-bit offsets from the beginning of the message etc.

If there are unused areas (padding) between the message fields, these must be explicitly defined in the message definitions (section 4).

When constructing the messages, all unused areas must be written as zero.

1.2.3 Decimal values

The messages can only contain integer fields. Where non-integer values must be delivered, they must be represented with fixed point fields (scaled values as integers).

2 Message structure

Each protocol message contains the same message header and a message specific payload definition located directly after the message header.

A message can also contain an optional synchronization preamble of 4 bytes located before the message header. The purpose of the synchronization preamble is to mark a message beginning. This can be utilized in recognizing message boundaries in a protocol receiver when messages are delivered over a stream based connection and it is not known in advance when the boundary is.

The protocol engines in the embedded software may have configuration options to define if the synchronization preamble should be inserted into the messages or not. Any receiver of the application protocol must be capable of detecting the synchronization preamble, if present. Optionally the receiver may also have a configuration setting to discard all messages which do not have the preamble in the beginning of the message.

The synchronization preamble has a fixed value of 2477F5F6h.

A general structure of a protocol message is illustrated in Figure 1.

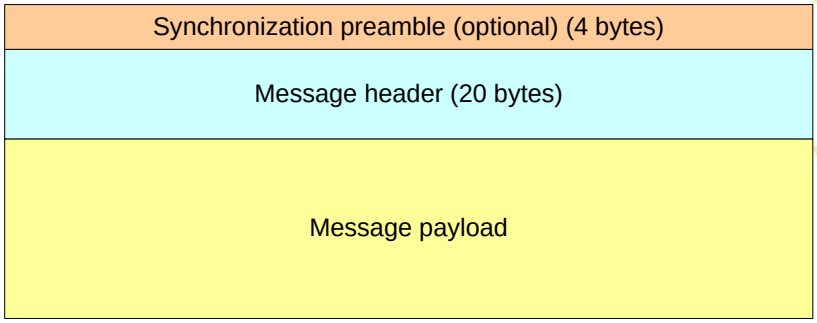


Figure 1: Generic message structure

2.1 Message header

Each protocol message must contain a 20-byte message header, which contains common fields for all messages. The message header is used for identifying the source of the message, verifying its integrity and indicating the type of the message payload.

The contents of the message header are illustrated in Figure 2. Byte offsets refer to the beginning of the message header.

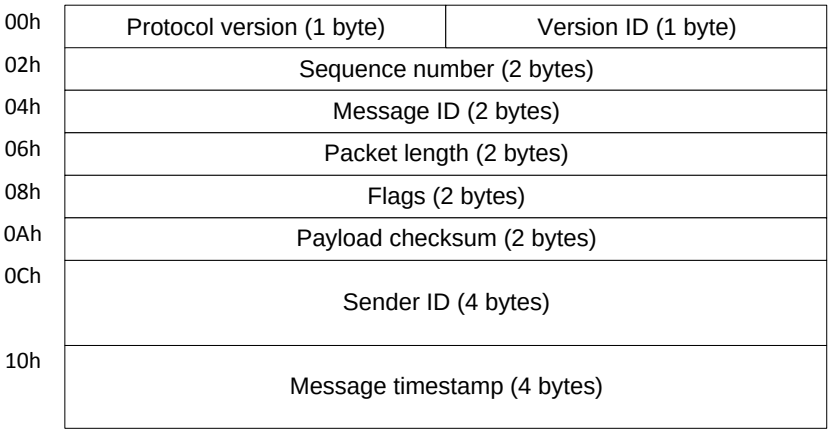
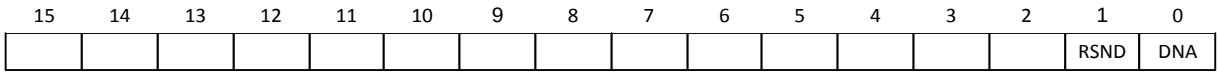


Figure 2: Message header structure

Offset	***Field***	***Description***
00h	Protocol version	Version of the protocol definition. Identifies how rest of the message should be interpreted. This document specifies a protocol version 1.
02h	Sequence number	A 16-bit counter representing the sequence of messages created by the protocol engine. After creation of the message the sequence number counter is incremented by one. Note that the sequence number order can differ from the timestamp order in case a message was created as a snapshot, stored internally during a message trigger preconditioning and some other messages (thus with a bigger timestamp value) were created in between.
04h	Message ID	Identifier of the message type. Message ID values are defined in Table 3.
06h	Packet length	Number of bytes in the entire message structure

		containing the synchronization preamble (if present), message header and message payload.
08h	Flags	Bitflags for carrying additional information. See Table 2 for bit field definitions.
0Ah	Payload checksum	Checksum of the message payload area.
0Ch	Sender ID	Unique identifier of the mobile unit sending the data.
10h	Message timestamp	Timestamp of the time instant when the payload data was created.

Table 1: Message header fields



Bit index	*Field*	*Description*
0	DNA	No Not Acknowledge. This message should not be acknowledged by the received.
1	RSND	Resend. The sender can indicate resending of a previously sent message with this bit. Setting this bit is optional.

Table 2: Message header flags

Message ID	*Message name*
2	ERROR
4	INDICATION
5	CONN_OPEN
6	CONN_CLOSE
7	SYSTEM_REPORT
8	UNIT_REPORT
10	GEOFENCE_ALARM
11	INPUT_ALARM
255	ACKNOWLEDGEMENT

Table 3: Message ID values

2.2 Message payload

The specific payload contents of individual messages are presented in Section 4.

3 Message encoding

In some situations messages may be transferred over such transports where raw binary data is not supported, 100% guaranteed or it is otherwise required to encode the message contents with a special encoding scheme.

A common example of such a scenario is a transport originally intended to carry human readable text only (e.g. SMS). Although SMS PDU structure does support 8-bit binary payload, implementation of the binary SMS payload varies between network operators and generally an application cannot rely that a binary SMS would be transferred to the recipient with the same success rate as normal text messages.

The purpose of the message encoding is to convert the original binary message to a stream of characters allowed in the transport medium. This encoded message can then be transferred to the recipient and decoded into the original binary format and interpreted as a protocol message.

The transport medium may also be limited to even smaller number of allowed characters, e.g., numbers only.

Note that any of the encoding schemes presented below are not intended for compression of the message. Instead, all of them increase the data size and therefore reduce the maximum possible message size, given a certain limit for packet size in the transport medium.

3.1 Encoding principle

A typical encoding scheme uses a set of allowed characters (referred to as “base”) to represent binary data in the original message. One commonly used encoding scheme is Base64, where 64 characters are used as the base of the encoding scheme. When combined into groups of four characters, the group can represent 64^4 distinct values.

As $64^4 = 16777216 = 2^{24} = 256^3$, it can be seen that four digits of Base64 character set can represent any three values of 8-bit binary data. Therefore, Base64 encoding can be defined as encoding three binary values with four Base64 digits. The size ratio is therefore 4/3 = 133%, meaning that each encoded message will take (at least) 133% of the original message size. If the size of the message is not an integer multiple of 3, the size ratio will be slightly higher than 133%, as also the tail of the message (1 or 2 bytes) needs to be encoded with four Base64 characters.

Exactly the same approach can be utilized for other bases, as well. The principle is to find a suitable group size that can represent a given number of 8-bit binary values. The group size should be kept as short as possible to avoid creating long tails and therefore reducing the actual size ratio.

3.1.1 Scheme identification character

Each encoding scheme used for Navigil application protocol messages must contain an identifier of the encoding scheme so that the receiver does not necessarily have to have this information available from other sources. The scheme identification character is the first character in the encoded packet.

This identifier must be selected for each encoding scheme so that the identification character can be carried in the transport medium and it can be distinguished from identifiers of all other schemes as well as from the protocol version number in binary format (version number is the first byte of binary format where no encoding is used).

Depending on the scheme, the identification character may or may not belong to the character set used for actual data encoding.

3.1.2 Message synchronization pattern

For similar reasons as using the synchronization preamble in the binary messages, an encoded message can also contain a special synchronization pattern in the beginning of the encoded packet. This is required for stream based transports where several packets are transferred without explicit knowledge about where the packet boundaries are.

If the encoding identifier character does not belong to the base of the scheme, the identifier alone can be used as the synchronization pattern. Where the entire allowed character set is utilized by the base (e.g. numbers only, Base10), the synchronization pattern is generally of equal length with the group size of encoded characters. In this case the first character of the synchronization pattern is equal to the encoding identifier.

Using the synchronization pattern is optional and can be omitted, when packet boundaries are unambiguously known otherwise.

3.1.3 Encoding tail pattern

When the size of the payload to be encoded is not an integer multiple of the group size of encoded digits, the end of the message contains less bytes than are needed to form a group of bytes to be encoded.

This situation is covered in to different ways, depending on the base. With schemes where characters outside the base are also allowed in the transport medium, special padding characters are used for identifying that the last group is partial. See an example in Base64 scheme.

When characters outside the base cannot be used (e.g., Base10), it is not possible to explicitly indicate a partial group. In this case the raw data group is padded with zeros to form an entire group. The actual size of the raw data is therefore not unambiguously known from the encoded data and must be handled otherwise.

3.2 Base64

Base64 is a commonly used encoding scheme for binary data, when transferred or stored using normal alphanumeric characters only. A common approach is to use uppercase and lowercase letters, numbers and a few common punctuation marks to achieve the base size of 64 characters.

Detailed examples of Base64 can be found in the internet, see e.g. http://en.wikipedia.org/wiki/Base64.

Navigil application protocol encoding with Base64 is identical to the scheme presented in the link above. The index table uses uppercase letters from A-Z, lowercase letters from a-z, numbers from 0-9, a plus sign and a slash (in this order) for values 0-63.

Encoding identifier:	.	(0x2e)
Synchronization pattern:	.?	(0x2e 0x3f)
Padding character:	=	(0x3D)

Index table:

Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

Examples:

Original (hex)	Encoded (with synchronization)
19 18 27 F3 91 73 97 12 98 31 28 93	.?GRgn85FzlxKYMSiT
19 18 27 F3 91 73 97 12 98 31 28	.?GRgn85FzlxKYMSg=
19 18 27 F3 91 73 97 12 98 31	.?GRgn85FzlxKYMQ==

3.3 Base10

Base10 is a “numbers only” encoding scheme designed for representing binary values using only digits 0-9, thus creating a base of 10 distinct characters.

When the base size is not 2^n, the number of distinct values in any exponent of the base size does not usually equal to an integer exponent of 256, meaning that the entire number range of the base is not utilized.

In case of Base10, the selected approach is to represent two 8-bit binary bytes (values 0-65535 when interpreted as a 16-bit integer value) with five digits. Values 65536-99999 are unused by the encoding scheme.

When converting a group of data to be encoded into an integer value, the bytes are interpreted as big endian byte order.

The groups of encoded characters must also have a length a 5 digits, i.e., leading zeros are required for all values less than 10000.

Encoding identifier: 8 (0x38)

Synchronization pattern: 89999 (0x38 0x39 0x39 0x39 0x39)

Examples:

Original (hex)	Encoded (with syncronization)
19 18 27 F3	89999 06424 10227
19 18 27 (equal to 19 18 27 00)	89999 06424 09984

3.4 Base11

Base11 is an improved version of Base10, intended specifically to be used in outbound USSD messages, where a guaranteed character set is number 0-9 and an asterisk sign (hash sign is used as a message terminator and cannot be used in the encoding base).

Base11 follows the idea of Base10, but uses an asterisk (*) to represent a base value 10. Groups of 3 binary bytes are encoded into groups of 7 Base11 characters.

Encoding identifier:	9	(0x39)
Synchronization pattern:	9*99*99	(0x39 0x2A 0x39 0x39 0x2A 0x39 0x39)

Examples:

Original (hex)	**Encoded (without syncronization)**
19 18 28 F3 A2 2E	9 0*23667 9016082
E1 8A 17 FE 18 (equal to E1 8A 17 FE 18	9 8386169 9444124

4 Message definitions

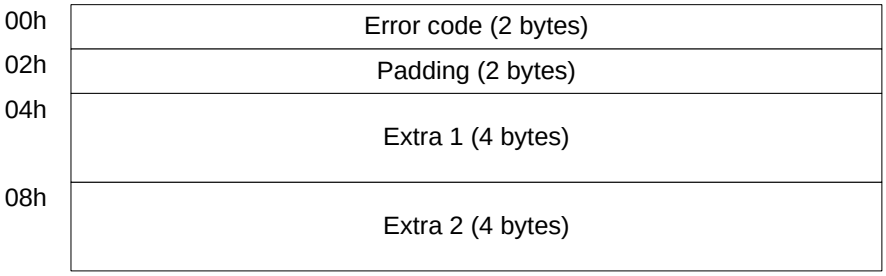
4.1 ERROR (ID 2)

Description Error indication. This message is used to deliver error indications.

Payload size 12 bytes

Support [1]

TG1	TG2
●	●



Offset	*Field*	*Description*
00h	Error code	Identifier of the type of the error. Values are defined in Table 4
04h	Extra 1	An error specific extra code, defined in a separate document.
08h	Extra 2	An error specific extra code, defined in a separate document.

Code	*Description*
1	File download
2	File upload
3	Unknown file transfer
4	OTA update
5	Firmware download
6	Geofence database download
7	Geofence activation file download
8	Eventlog upload

Table 4: Error message codes

[1] This element defines support for the corresponding message in Generation 1 (TG2) and Generation (TG2) products. ● refers to full support in the current version, ○ indicates that the message has been support in earlier versions, but is not supported by the latest available version anymore.

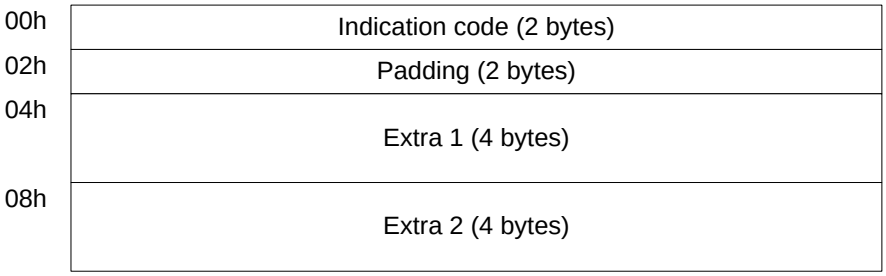
4.2 INDICATION (ID 4)

Description Generic indication message. Similar to the ERROR message but is used for communicating events that are not actual errors.

Payload size 12 bytes

Support

TG1	TG2
•	•



Offset	***Field***	***Description***
00h	Indication code	Identifier of the type of the error. Values are defined in Table 5
04h	Extra 1	An error specific extra code, defined in a separate document.
08h	Extra 2	An error specific extra code, defined in a separate document.

Code	***Description***
1	Safe mode enter
2	Safe mode exit
3	Temperature warning
4	Temperature alarm
5	Firmware downloaded
6	Geofence database downloaded
7	Geofence activation file downloaded
8	Eventlog uploaded
9	File downloaded
10	File uploaded
11	Unknown file transfer
12	Reboot

Table 5: Indication message codes

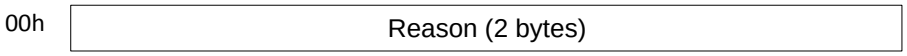
4.3 CONN_OPEN (ID 5)

Description Server handshake message. Indicates that a communication channel between the unit and the server has been opened.

Payload size 2 bytes

Support

TG1	TG2
•	•



Offset	*Field*	*Description*
00h	Reason	Reason for opening the connection. Values are defined in Table 6

0	Reason information not available
10	Timer wakeup
11	Power supply wakeup
12	I/O wakeup
13	Shock sensor wakeup
14	Accelerometer wakeup
30	GPRS error recovery
31	Critical system error
32	Internal reboot
40	External reset

Table 6: Connection open reason codes

4.4 CONN_CLOSE (ID 6)

Description Indicates that the communication channel between the unit and the server has been closed.

Payload size 2 bytes

Support

TG1	TG2
●	●

00h	Reason (2 bytes)
02h	Sleep time (2 bytes)

Offset	***Field***	***Description***
00h	Reason	Reason for closing the connection. Values are defined in Table 7
02h	Sleep time	Estimated sleep time, if asynchronous wakeup trigger do not occur during the sleep period. Zero if the information is not available. Note that the unit may not necessarily reopen the server connection after wakeup, so the sleep time information does not mean the maximum time before the next message from the unit. Units: minutes

1	Server address removed
2	Entering sleep
3	Temperature shutdown
4	Undervoltage shutdown
99	Unknown

Table 7: Connection close reason codes

4.5 SYSTEM_REPORT (ID 7)

Description Contains system level information of the unit. This message is currently used for delivering the version number only.

Payload size 6 bytes

Support

TG1	TG2
•	•

00h	Version major (2 bytes)
02h	Version minor (2 bytes)
04h	Version build (2 bytes)

Offset	*Field*	*Description*
00h	Version major	Major number of the software version.
02h	Version minor	Minor number of the software version.
04h	Version build	Build number of the software version.

4.6 UNIT_REPORT (ID 8)

Description The normal status communication message. Contains the most essential fields representing the current status of the unit.

Payload size 66 bytes

Support

TG1	TG2
•	

00h	Report trigger (2 bytes)
02h	State flags (2 bytes)
04h	Latitude (4 bytes)
08h	Longitude (4 bytes)
0Ch	Altitude (2 bytes)
0Eh	Satellites in fix (2 bytes)
10h	Satellites in track (2 bytes)
12h	GPS antenna state (2 bytes)
14h	Speed (2 bytes)
16h	Direction (2 bytes)
18h	Distance (4 bytes)
1Ch	Delta distance (4 bytes)
20h	Supply voltage (2 bytes)
22h	Battery charger status (2 bytes)
24h	Fix timstamp (4 bytes)
28h	Status flags (2 bytes)
2Ah	Acceleration X (2 bytes)
2Ch	Acceleration Y (2 bytes)
2Eh	Acceleration Z (2 bytes)
30h	GSM MCC (2 bytes)
32h	GSM MNC (2 bytes)
34h	GSM LAC (2 bytes)
36h	GSM CID (2 bytes)
38h	GSM network status (2 bytes)
3Ah	GSM module temperature (2 bytes)
3Ch	I/O status flags (2 bytes)
3Eh	Maximum speed (2 bytes)
40h	Minimum speed (2 bytes)

Offset	*Field*	*Description*
00h	Report trigger	Reason for creating the particular message. Values are defined in Table 8.
02h	State flags	Placeholder for state flags. No flags defined yet, must be written as zeros.
04h	Latitude	Current latitude, unsigned integer. Units: 0.0000001 degrees
08h	Longitude	Current longitude, unsigned integer. Units: 0.0000001 degrees
0Ch	Altitude	Current MSL altitude, unsigned integer. Units: m
0Eh	Satellites in fix	Number of satellites used for calculating the position fix.
10h	Satellites is track	Number of satellites currently tracked by the GPS receiver.
12h	GPS antenna state	State of the GPS antenna. 0 = LNA off 1 = Short circuit 2 = Internal 3 = External
14h	Speed	Current speed, unsigned integer Units: 0.1 m/s
16h	Direction	Current direction, unsigned integer. Units: degrees (0=north, 90=east, 180=south, 270=west)
18h	Distance	Current odometer distance, unsigned integer Units: m
1Ch	Delta distance	Distance travelled from the previous report, unsigned integer Units: m
20h	Supply voltage	Voltage of the power supply of the uTrace PCB, either from an internal battery or an external power supply. Units: mV
22h	Battery charger status	Status of the internal battery charger. 0 = Power disconnected 1 = Charging 2 = Battery full 3 = Charging error
24h	Fix timestamp	Timestamp of the instant when the previous valid fix was calculated.
28h	Status flags	Bit flags for the unit status. Flag positions are defined in Table 9.
2Ah	Acceleration X	Current acceleration along X axis, signed integer. Units: 0.001 G
2Ch	Acceleration Y	Current acceleration along Y axis, signed integer. Units: 0.001 G
2Eh	Acceleration Z	Current acceleration along Z axis, signed integer. Units: 0.001 G
30h	GSM MCC	Mobile Country Code of the current GSM operator
32h	GSM MNC	Mobile Network Code of the current GSM operator
34h	GSM LAC	Location Area Code of the current serving GSM cell

36h	GSM CID	Cell ID of the current serving GSM cell
38h	GSM network status	Status of the GSM network registration. 0 = No Network 1 = Home network 2 = Searching 3 = Registration denied 5 = Roaming network
3Ah	GSM module temperature	Temperature of the GSM module, unsigned integer. Units: degrees C
3Ch	I/O status flags	Bit flags for the unit status. Flag positions are defined in **Table 10**.
3Eh	Maximum speed	Maximum speed since the previous report. Units: km/h
40h	Minimum speed	Minimum speed since the previous report. Units: km/h

1	User interface event
3	Travelled distance
4	Timer
5	Shock sensor
6	GSM temperature alert
7	GSM operator changed
8	GSM cell ID changed
9	Battery low
10	Charger status changed
11	GPS antenna status changed
15	Battery power
16	External power
17	Battery ok
18	Heading changed
19	Accelerometer motion start
20	Accelerometer motion end
21	GPS based trip start
22	GPS based trip end
23	Shock sensor motion start
24	Shock sensor motion end
25	Pinning active
26	Pinning inactive
27	GSM registration status changed
28	User interface action
29	Ignition off
30	Ignition on
31	Crash detected
33	GPS fix lost
34	GPS fix acquired
35	GPS first fix

Table 8: Report trigger codes

0	Trip in progress
1	Accelerometer motion active
2	Shock sensor motion active
13	Accelerometer communication error
14	Temperature warning
15	Battery low

Table 9: Status flag bit positions

0	External CMOS input HIGH
1	External high voltage input HIGH
8	Open drain output CONDUCTIVE

Table 10: I/O flag bit positions

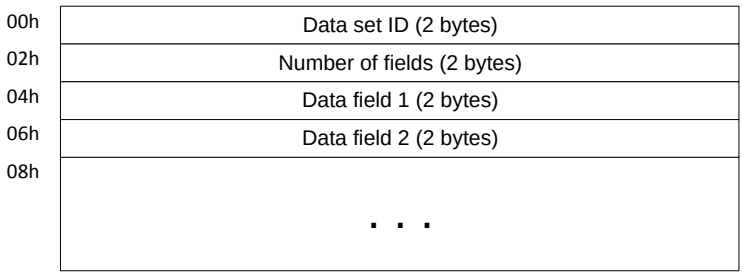
4.7 DIAGNOSTICS_REPORT (ID 9)

Description Generic message structure for carrying diagnostic information. The message payload contains an unlimited number of 16-bit fields for counters, etc. The actual contents of the fields are defined by the data set ID.

Payload size Variable

Support

TG1	TG2



Offset	*Field*	*Description*
00h	Data set ID	Identifier of the data set carried by the message
02h	Number of fields	Number of data fields present in the payload
04h	Data field 1	Data field for carrying the diagnostic information

4.8 GEOFENCE_ALARM (ID 10)

Description Geofence alarm indication.

Payload size 84 bytes

Support

TG1	TG2
●	●

00h	Latitude (4 bytes)
04h	Longitude (4 bytes)
08h	Altitude (2 bytes)
0Ah	Speed (2 bytes)
0Ch	Direction (2 bytes)
0Eh	Alarm type (2 bytes)
10h	Geofence ID (2 bytes)
12h	Group ID (2 bytes)
14h	Name (64 bytes)

Offset	***Field***	***Description***
00h	Latitude	Current latitude, the same definition as in UNIT_REPORT.
04h	Longitude	Current longitude, the same definition as in UNIT_REPORT.
08h	Altitude	Current altitude, the same definition as in UNIT_REPORT.
0Ah	Speed	Current speed, the same definition as in UNIT_REPORT.
0Ch	Direction	Current direction, the same definition as in UNIT_REPORT.
0Eh	Alarm type	Geofence alarm type code. 1 = Geofence IN 2 = Geofence OUT 3 = Geofence group IN 4 = Geofence group OUT
10h	Geofence ID	ID of the alerting geofence, 0 if not applicable.
12h	Group ID	ID of the alerting geofence gorup, 0 if not applicable.
14h	Name	Name of the alerting geofence/group. NULL-terminated 8-bit ASCII. All bytes after the name characters must be written as zeros.

4.9 INPUT_ALARM (ID 11)

Description Indicates an alarm of any of the GPIO inputs.

Payload size 18 bytes

Support

TG1	TG2
●	

00h	Latitude (4 bytes)
04h	Longitude (4 bytes)
08h	Altitude (2 bytes)
0Ah	Speed (2 bytes)
0Ch	Direction (2 bytes)
0Eh	Alarm type (2 bytes)
10h	Input ID (2 bytes)

Offset	***Field***	***Description***
00h	Latitude	Current latitude, the same definition as in UNIT_REPORT.
04h	Longitude	Current longitude, the same definition as in UNIT_REPORT.
08h	Altitude	Current altitude, the same definition as in UNIT_REPORT.
0Ah	Speed	Current speed, the same definition as in UNIT_REPORT.
0Ch	Direction	Current direction, the same definition as in UNIT_REPORT.
0Eh	Alarm type	Input alarm type code. 1 = Fall 2 = Rise 3 = Hold up 4 = Hold down
10h	Input ID	ID of the alerting input. 1 = uTrace evaluation kit button 1 2 = uTrace evaluation kit button 2 3 = uTrace evaluation kit power button 4 = uTrace evaluation kit high voltage input 5 = uTrace evaluation kit CMOS input

4.10 TG2_REPORT (ID 12)

Description Standard snapshot report used by the Generation 2 software. Used in the same way as UNIT_REPORT messages by Gen1 software.

Payload size 64 bytes

Support

TG1	TG2
	O

00h	Report trigger (2 bytes)	
02h	Reserved (1 byte)	GPS assistance age (1 byte)
04h	Fix timestamp (4 bytes)	
08h	Latitude (4 bytes)	
0Ch	Longitude (4 bytes)	
10h	Altitude (2 bytes)	
12h	Satellites in fix (1 byte)	Satellites in track (1 byte)
14h	Speed (2 bytes)	
16h	Direction (2 bytes)	
18h	Distance (4 bytes)	
1Ch	Maximum speed (2 bytes)	
1Eh	Minimum speed (2 bytes)	
20h	VSAUT1 voltage (2 bytes)	
22h	VSAUT2 voltage (2 bytes)	
24h	Solar voltage (2 bytes)	
26h	Battery voltage (2 bytes)	
28h	Status flags (2 bytes)	
2Ah	I/O status flags (2 bytes)	
2Ch	Warning flags (2 bytes)	
2Eh	Alarm flags (2 bytes)	
30h	GSM MCC (2 bytes)	
32h	GSM MNC (2 bytes)	
34h	GSM LAC (2 bytes)	
36h	GSM CID (2 bytes)	
38h	GSM registration status (1 byte)	GSM signal level (1 byte)
3Ah	Temperature (2 bytes)	
3Ch	ADC1 voltage (2 bytes)	
3Eh	ADC2 voltage (2 bytes)	

Offset	*Field*	*Description*
00h	Report trigger	Reason for creating the particular message. Values are defined in Table 8.
03h	GPS assistance age	Age of the current GPS assistance data. Units: days (255 = assistance not available, 254 = more than 253 days)
04h	Fix timestamp	Timestamp of the instant when the previous valid fix was calculated.

08h	Latitude	Current latitude, unsigned integer. Units: 0.0000001 degrees
0Ch	Longitude	Current longitude, unsigned integer. Units: 0.0000001 degrees
10h	Altitude	Current MSL altitude, unsigned integer. Units: m
12h	Satellites in fix	Number of satellites used for calculating the position fix.
13h	Satellites is track	Number of satellites currently tracked by the GPS receiver.
14h	Speed	Current speed, unsigned integer Units: 0.1 m/s
16h	Direction	Current direction, unsigned integer. Units: degrees (0=north, 90=east, 180=south, 270=west)
18h	Distance	Current odometer distance, unsigned integer Units: m
1Ch	Maximum speed	Maximum speed since the previous report. Units: km/h
1Eh	Minimum speed	Minimum speed since the previous report. Units: km/h
20h	VSAUT1 voltage	Voltage in the VSAUT1 power supply input. Units: mV
22h	VSAUT2 voltage	Voltage in the VSAUT2 power supply input. Units: mV
24h	Solar voltage	Voltage in the solar panel input. Units: mV
26h	Battery voltage	Voltage of the internal battery. Units: mV
28h	Status flags	Generic status flags. Bit fields are defined in Figure 3.
2Ah	I/O status flags	I/O status flags. No bit fields current defined.
2Ch	Warning flags	System level warning flags. Bit fields are defined in Figure 4.
2Eh	Alarm flags	System level alarm flags. Bit fields are defined in Figure 5.
30h	GSM MCC	Mobile Country Code of the current GSM operator
32h	GSM MNC	Mobile Network Code of the current GSM operator
34h	GSM LAC	Location Area Code of the current serving GSM cell
36h	GSM CID	Cell ID of the current serving GSM cell
38h	GSM registration status	Status of the GSM network registration. 0 = No Network 1 = Home network 2 = Searching 3 = Registration denied 5 = Roaming network
39h	GSM signal level	Current GSM signal level. Units: dBm
3Ah	Temperature	Current unit temperature. Units: degrees C
3Ch	ADC1 voltage	Voltage in the external ADC input 1. Units: mV
3Eh	ADC2 voltage	Voltage in the external ADC input 2. Units: mV

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCHG	BFUL						EANT	EPWR	SFMD				SHK	ACC	TRP

Figure 3: Status flags

Bit index	*Field*	*Description*
0	TRP	Trip in progress
1	ACC	Accelerometer motion
2	SHK	Shock sensor motion
6	SFMD	Safe mode active
7	EPWR	External power available
8	EANT	External GNSS antenna in use
14	BFUL	Battery full
15	BCHG	Battery charging in progress

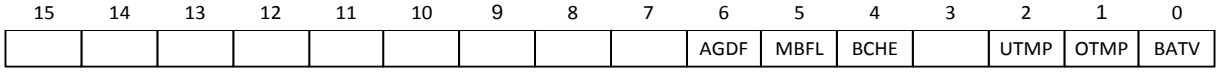


Figure 4: Warning flags

Bit index	*Field*	*Description*
0	BATV	Battery low
1	OTMP	Overtemperature warning
2	UTMP	Undertemperature warning
4	BCHE	Battery charger error
5	MBFL	Message buffer fill level warning
6	AGDF	AGPS data download error

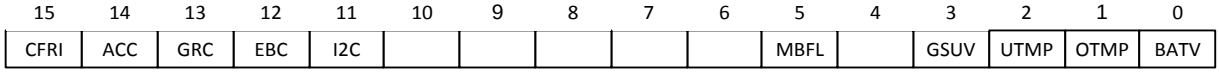


Figure 5: Alarm flags

Bit index	*Field*	*Description*
0	BATV	Battery voltage below undervoltage shutdown level.
1	OTMP	Overtemperature alarm
2	UTMP	Undertemperature alarm
3	GSUV	GSM undervoltage shutdown alarm
5	MBFL	Message buffer full
11	I2C	Generic I2C bus communication alarm
12	EBC	IO expander B communication error
13	GRC	Gyro communication error
14	ACC	Accelerometer communication error
15	CFRI	Configuration file read interrupted

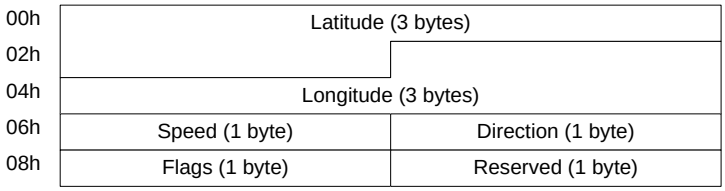
4.11 POSITION_REPORT (ID 13)

Description Position report for periodic reporting. Optimized for small size and contains only the most essential position related fields.

Payload size 10 bytes

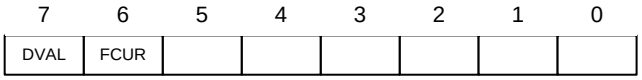
Support

TG1	TG2
●	●



Offset	*Field*	*Description*
00h	Latitude	Current latitude, unsigned integer. Units: 0.00002 degrees
03h	Longitude	Current latitude, unsigned integer. Units: 0.00002 degrees
06h	Speed	Current speed, saturated to 255 km/h Units: km/h
07h	Heading	Current heading Units: 2 degrees
08h	Flags	Status flags, see bit definitions below
09h	Reserved	Reserved for future use, set to zero

Status flags



Offset	*Bit*	*Description*
7	DVAL	Data Valid. Set if the position data in the other payload field is valid (current or old position). If this bit is not set, the position information should not be used in any way.
6	FCUR	Current Fix. Set if the position is the current position when the message is created. If this bit is not set, the position information is the last known good position information.

4.12 CONSOLE_DATA (ID 14)

Description Message for delivering maintenance console data with the protocol messages.

Payload size 88 bytes

Support

TG1	TG2

00h	Data length (2 bytes)	
02h	Set identifier (2 bytes)	
04h	Flags (1 byte)	Reserved (1 byte)
06h	Reserved (2 bytes)	
08h	Data (80 bytes)	

Offset	***Field***	***Description***
00h	Data length	Length of the actual payload in the Data field
02h	Set identifier	Identifier of a data set. In response messages the set identifier is the same as in the original message.
04h	Flags	Status flags, see description below.
08h	Data	Actual console data payload. Not NULL terminated.

Status flags

7	6	5	4	3	2	1	0
RESP							

Offset	***Bit***	***Description***
7	RESP	Response bit. This bit is set if the message contains a response to a received message.

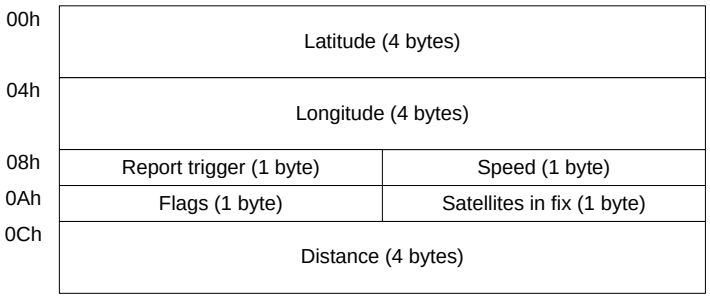
4.13 POSITION_REPORT_2 (ID 15)

Description Similar reduced content report to POSITION_REPORT but with slight differences in the payload contents.

Payload size 16 bytes

Support

TG1	TG2
●	●



Offset	***Field***	***Description***
00h	Latitude	Current latitude, unsigned integer. Units: 0.0000001 degrees
04h	Longitude	Current longitude, unsigned integer. Units: 0.0000001 degrees
08h	Report trigger	Reason for creating the particular message. Values are defined in Table 8.
09h	Speed	Current speed, saturated to 255 km/h Units: km/h
0Ah	Flags	Status flags, identical definitions to the Flags field of POSITION_REPORT.
0Bh	Satellites in fix	Number of satellites used for calculating the position fix.
0Ch	Distance	Current odometer distance, unsigned integer Units: m

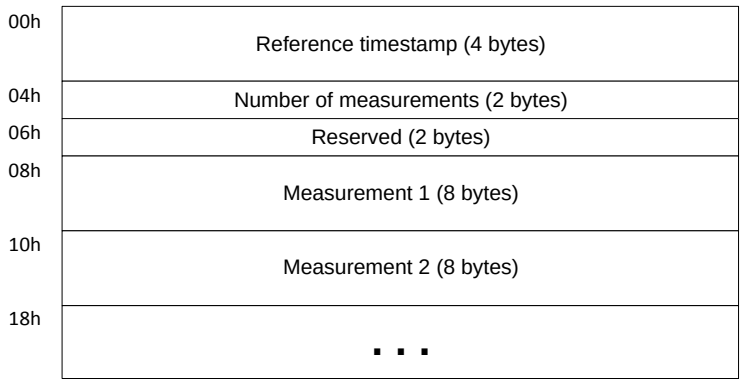
4.14 MEASUREMENT_DATA (ID 16)

Description This message contains generic measurement data values. One message can hold several measurements.

Payload size Variable

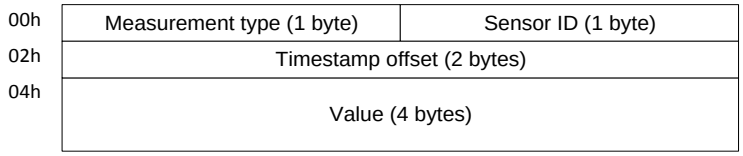
Support

TG1	TG2



Offset	***Field***	***Description***
00h	Reference timestamp	Base timestamp to represent actual measurement timestamps. A reference timestamp approach is used for reducing payload size by using 2-byte timestamp offset fields for the actual measurements. Units: Message timestamp format defined in section 1.1.2.
04h	Number of measurements	Number of measurement items included into the message. The size of the message depends on this value.
	Measurement *n*	*n*th measurement data. See measurement data details below.

Measurement data item:



Offset	***Field***	***Description***
00h	Measurement type	Type of the measurement. Used for proper interpretation of the measurement value. Type identifiers are application specific and are not defined by this document.
01h	Sensor ID	Sensor ID providing the measurement value. Can be used for cases where multiple sensors measure the same quantity.
02h	Timestamp offset	Time offset from the reference timestamp provided in the fixed part of the message. Units: seconds
04h	Value	Actual value of the measurement. 4 bytes of unrestricted data. Interpretation of the measurement value is application dependent based on the measurement type.

4.15 SNAPSHOT4 (ID 17)

Description A snapshot message used for standard reporting by the TD product family.

Payload size 64 bytes

Support

TG1	TG2
	●

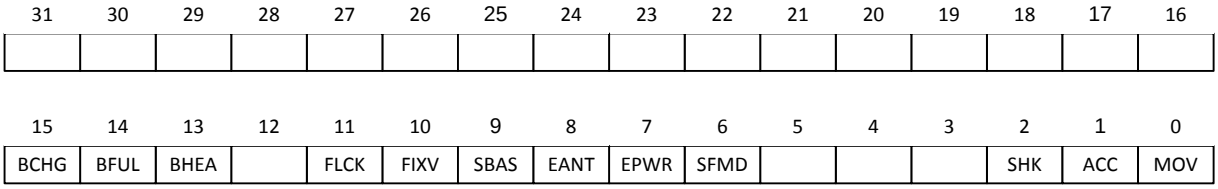
00h	Report trigger (1 byte)	Position fix source (1 byte)
02h	GNSS fix quality (1 byte)	GNSS assistance age (1 byte)
04h	Status flags (4 bytes)	
08h	Fix timestamp (4 bytes)	
0Ch	Latitude (4 bytes)	
10h	Longitude (4 bytes)	
14h	Altitude (2 bytes)	
16h	Speed (2 bytes)	
18h	Direction (2 bytes)	
1Ah	Maximum speed (1 byte)	Maximum speed (1 byte)
1Ch	Distance (4 bytes)	
20h	Supply voltage 1 (1 byte)	Supply voltage 2 (1 byte)
22h	Battery voltage (1 byte)	Temperature (1 byte)
24h	I/O status flags (2 bytes)	
26h	Warning flags (2 bytes)	
28h	Alarm flags (2 bytes)	
2Ah	GSM MCC (2 bytes)	
2Ch	GSM MNC (2 bytes)	
2Eh	GSM LAC (2 bytes)	
30h	GSM CID (2 bytes)	
32h	GSM registration status (1 byte)	GSM signal level (1 byte)
34h	ADC1 voltage (2 bytes)	
36h	ADC2 voltage (2 bytes)	
38h	Geofence (2 bytes)	
3Ah	Distance to geofence (2 bytes)	

Offset	***Field***	***Description***
00h	Report trigger	Reason for creating the particular message. Values are defined in Table 11.
01h	Position fix source	Source of the position information. 1 = GPS 2 = Glonass 11 = GPS & Glonass 20 = GSM Cell ID position

02h	GNSS fix quality	Quality indicator for the represented GNSS fix. Value range: 0-100
03h	GNSS assistance age	Age of the current GNSS assistance data. Units: days (255 = assistance not available, 254 = more than 253 days)
04h	Status flags	Generic status flags. Bit fields are defined in Table 12.
08h	Fix timestamp	Timestamp of the instant when the previous valid fix was calculated.
0Ch	Latitude	Current latitude, unsigned integer. Units: 0.0000001 degrees
10h	Longitude	Current longitude, unsigned integer. Units: 0.0000001 degrees
14h	Altitude	Current MSL altitude, unsigned integer. Units: m
16h	Speed	Current speed, unsigned integer Units: 0.1 m/s
18h	Direction	Current direction, unsigned integer. Units: degrees (0=north, 90=east, 180=south, 270=west)
1Ah	Maximum speed	Maximum speed since the previous report. Units: km/h
1Bh	Minimum speed	Minimum speed since the previous report. Units: km/h
1Ch	Distance	Current odometer reading, unsigned integer Units: m
20h	Supply voltage 1	Voltage in the external power supply input 1. Units: 100 mV, offset 8000 mV
21h	Supply voltage 2	Voltage in the external power supply input 2. Units: 100 mV, offset 8000 mV
22h	Battery voltage	Voltage of the internal battery. Units: 10 mV, offset 2500 mV
23h	Temperature	Current unit temperature. Units: degrees C
24h	I/O status flags	I/O status flags. Bit fields are defined in Table 13.
26h	Warning flags	System level warning flags. Bit fields are defined in Figure 4.
28h	Alarm flags	System level alarm flags. Bit fields are defined in Figure 5.
2Ah	GSM MCC	Mobile Country Code of the current GSM operator
2Ch	GSM MNC	Mobile Network Code of the current GSM operator
2Eh	GSM LAC	Location Area Code of the current serving GSM cell
30h	GSM CID	Cell ID of the current serving GSM cell
32h	GSM registration status	Status of the GSM network registration. 0 = No Network 1 = Home network 2 = Searching 3 = Registration denied 5 = Roaming network
33h	GSM signal level	Current GSM signal level. Units: dBm
34h	ADC1 voltage	Voltage in the external ADC input 1. Units: mV
36h	ADC2 voltage	Voltage in the external ADC input 2. Units: mV
38h	Geofence	ID of the closest / currently active geofence. Zero if data is not available.
3Ah	Distance to geofence	Distance to the geofence indicated by the previous field. Valid only if the geofence ID is nonzero. Units: 0.1 km

1	Timer
2	Distance
3	Heading change
4	Motion start
5	Motion end
6	External power
7	Battery power
8	Ignition on
9	Ignition off
10	User event
11	Battery low
12	Battery ok
13	Geofence ID change
14	Input change

Table 11: Report trigger codes



Bit index	***Field***	***Description***
0	MOV	Movement. Vehicle is currently in motion
1	ACC	Accelerometer motion. The accelerometer is currently seeing mechanical motion.
2	SHK	Shock sensor motion. The shock sensor is currently seeing mechanical motion.
6	SFMD	Safe Mode. The unit is currently in safe mode.
7	EPWR	External power. The unit is currently powered from an external source.
8	EANT	External GNSS antenna. The unit is currently using an external GNSS antenna input.
9	SBAS	Space Based Augmentation System. The current position fix has been calculated using WAAS/EGNOS assistance data.
10	FIXV	Fix Valid. The current position fix is valid.
11	FLCK	Fix locked. The location is currently frozen to previously locked coordinates.
13	BHEA	Battery Heater. The onboard battery heater is currently active.
14	BFUL	Battery Full.
15	BCHG	Battery Charging. The battery charger is currently active.

Table 12: SNAPSHOT4 status flags

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												DIO4	DIO3	DIO2	DIO1

Bit index	*Field*	*Description*
0	DIO1	DIO1 line is currently high
1	DIO2	DIO2 line is currently high
2	DIO3	DIO3 line is currently high
3	DIO4	DIO4 line is currently high

Table 13: SNAPSHOT4 I/O flags

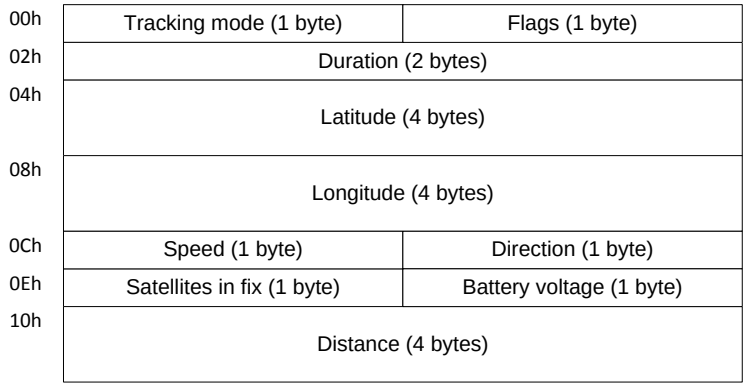
4.16 TRACKING_DATA (ID 18)

Description Real-time tracking data message. Optimized for sending frequent position updates.

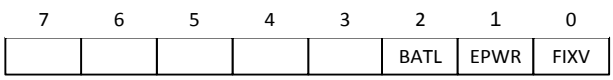
Payload size 18 bytes

Support

TG1	TG2
	●



Offset	*Field*	*Description*
00h	Tracking mode	Tracking mode identifier. 1 = Normal 2 = Emergency tracking
01h	Flags	Generic flags, see Table 14 for bit field definitions.
02h	Duration	Remaining duration of the currently active tracking mode. Units: minutes
04h	Latitude	Current latitude, unsigned integer. Units: 0.0000001 degrees
08h	Longitude	Current longitude, unsigned integer. Units: 0.0000001 degrees
0Ch	Speed	Current speed Units: km/h
0Dh	Direction	Current direction, unsigned integer. Units: 2 degrees (0=north, 90=east, 180=south, 270=west)
0Eh	Satellites in fix	Number of satellites used for calculating the position fix
0Fh	Battery voltage	Current battery voltage Units: 5 mV, offset from 3000 mV (range 3000-4275mV)
10h	Distance	Current odometer reading Units: meters



Bit index	*Field*	*Description*
0	FIXV	Position fix is currently valid
1	EPWR	The unit is currently externally powered
2	BATL	Battery low warning

Table 14: TRACKING_DATA flags

4.17 MOTION_ALARM (ID 19)

Description

This message is used for indicating motion alarms detected by onboard sensors. The message may contain a sensor data log from before and after the actual trigger detection.

The message contains a fixed part describing the alarm and the sensor log format. After the fixed part the actual samples are packed one after each other in recording order. Size of one sample depends on the sample contents (see description below).

Because the sensor data log can be of arbitrary length, the MOTION_ALARM message supports splitting the data set into several messages. By splitting the message large data sets can be delivered without limitations of maximum packet size in the used transport. When sending a data set in multiple messages, the fixed part must be identical in all messages, except the “Message index” field, which indicates the order of messages and “Number of samples”, which defines the number of sample in the particular message.

Payload size

Variable

Support

TG1	TG2

00h	Alarm type (1 byte)	Flags (1 byte)
02h	Data set ID (1 byte)	
04h	Message index (1 byte)	Number of messages (1 byte)
06h	Subsec timestamp (2 bytes)	
08h	Timestamp (4 bytes)	
0Ch	Latitude (4 bytes)	
10h	Longitude (4 bytes)	
14h	Position fix age (1 byte)	Alarm trigger (1 byte)
16h	Alarm trigger value (1 byte)	Alarm trigger threshold (1 byte)
18h	Number of samples (2 bytes)	
1Ah	Sample rate (1 byte)	Sample size (1 byte)
1Ch	Number of preroll samples (2 bytes)	
1Eh	Sample contents (2 bytes)	
20h	Sample 1 (x bytes)	
	Sample 2 (x bytes)	
	. . .	

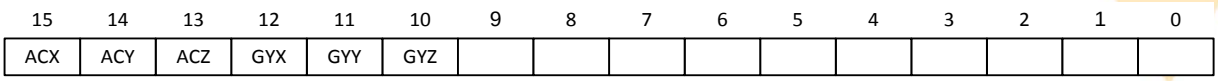
Offset	*Field*	*Description*
00h	Alarm type	Type of the generated alarm 1 = Acceleration alarm
01h	Flags	Generic flags. Defined for future use, no flags currently defined. Must be written as zero.
02h	Data set ID	Identifier of the reported data set. Used for reassembling a long data set from multiple messages.
04h	Message index	Index of the current message in a sequence of multiple messages for the same data set. Indexing starts from 1.
05h	Number of messages	Total number of messages in a sequence of multiple messages for the same data set.
06h	Subsec timestamp	High-precision timestamp representing offset from the integer timestamp value. Units: ms
08h	Timestamp	Timestamp of the detection of the alarm. In the same format as the message timestamp in the message header.
0Ch	Latitude	Latitude when the alarm was detected, unsigned integer. Units: 0.0000001 degrees
10h	Longitude	Latitude when the alarm was detected, unsigned integer. Units: 0.0000001 degrees
14h	Position fix age	Age of the coordinates. Units: seconds 255 = more than 254 seconds
15h	Alarm trigger	Trigger of the current alarm. See Table 15 for value definitions.
16h	Alarm trigger value	Value of the sensor input triggering the alarm. See Table 16 for format definitions
17h	Alarm trigger value threshold	Alarm threshold value for the sensor input triggering the alarm. The same format as the alarm trigger value field.
18h	Number of samples	Number of samples in the particular message. Can be zero, if only the alarm indication is delivered (no sensor log).
1Ah	Sample rate	Sampling rate of the sensor data included into the sensor log. Units: Hz (allowed range 1-100) 240 = 0.1 Hz 241 = 0.5 Hz
1Bh	Sample size	Size of one sample in bytes
1Ch	Number of preroll samples	Number of samples recorded before the alarm trigger
1Eh	Sample contents	Bitmask of different sensor inputs representing the data included into one sample. Individual sensor data fields are packed into the sample data in the order of the bitmask bits (highest index first). See Table 17 for bit field definitions.

Trigger	*Description*
1	Acceleration alarm

Table 15: Alarm trigger codes

Data source	*Value type*	*Value format*
Accelerometer	Measured acceleration	Unsigned integer, units: 0.1g Size: 1 byte
Gyro	Rotation speed	TBD Size: 1 byte

Table 16: Sample data field definitions



Bit index	*Field*	*Description*
15	ACX	Accelerometer X-axis acceleration
14	ACY	Accelerometer Y-axis acceleration
13	ACZ	Accelerometer Z-axis acceleration
12	GYX	Gyro X-axis rotation
11	GYY	Gyro Y-axis rotation
10	GYZ	Gyro Z-axis rotation

Table 17: Sample data content flags

4.18 ACKNOWLEDGEMENT (ID 255)

Description Acknowledgement of a received message.

Payload size 4 bytes

Support

TG1	TG2
●	●

00h	Message reference (2 bytes)
02h	Ack code (2 bytes)

Offset	***Field***	***Description***
00h	Message reference	Sequence number of the message being acknowledged.
02h	Ack code	Acknowledgement code. 0 = OK 1 = Duplicate message 200 = Checksum mismatch 201 = Unrecognized message

5 Appendix A: Checksum calculation

CRC algorithm used with the application protocol is the CCITT CRC-16 with the generator polynomial $x^{16} + x^{12} + x^5 + 1$ and initial value of FFFFh.

Data bytes of the message payload are given to the algorithm in the order in which they are sent to the network.

Some examples of simple data sequences are listed below.

Checksum test sequences	
Input sequence	**Checksum**
00h	E1F0h
00h 00h	1D0Fh
00h 01h 02h 03h	E5F1h
44h 1Dh F7h 81h 5Ah 17h 95h C0h	21BFh

One online calculator which can be used for verifying the checksum implementation is available at

http://www.zorc.breitbandkatze.de/crc.html

Clicking the “CRC-CCITT” changes all parameters of the calculator to correct settings for the application protocol algorithm.

Binary data can be entered into the Data sequence field with a percent sign leading each two-digit hex code of the byte value. For example value 60 (3C in hex) would be entered %3C. Similarly, a longer sequence 45 77 AE 1F w