

GROUND Lite Communications Protocol Specification

Version 1.0.0

B.B.F.M. Verspaandonk, M.Y.A. Wierckx

January 25, 2025

Contents

1 Overview	1
2 Packet Structure	1
Field Descriptions	1
3 Handling magic_number in Content	2
4 Encoding order	2
5 Examples	2
Single Value	2
Array	2
Escaped Magic Number	3

1 Overview

The GROUND (GAIA Radio OUtput Network Delivery) Lite protocol is a power and data efficiency optimized version of the original GROUND protocol. All data is serialized in little-endian format, meaning the least significant byte is sent first. For example, the number `0x1234` would be transmitted as `0x34 0x12`.

2 Packet Structure

A packet comprises the following fields:

#	Field Name	Size
1	<code>magic_number</code>	4 bytes
2	<code>content_type</code>	1 bytes
3	<code>content_size</code>	1 bytes
4	<code>content</code>	<code>content_size</code> bytes

Field Descriptions

magic_number: A constant value `0x67616961` (ASCII for `gaia`) that marks the start of a packet.

content_type: Specifies the type and structure of the data. Its interpreted as follows:

Value	Type	Description
<code>0x01</code>	<code>float GPS_POS[3]</code>	GPS coordinates
<code>0x02</code>	<code>float G_FORCES[3]</code>	G-force measurement
<code>0x03</code>	<code>float ROTATION[3]</code>	Angle measurement

0x04	uint32_t	TIME	Time
0x05	uint32_t	GPS_FIX_AGE	Time in ms since last gps fix
0x06	float	GPS_HDOP	Horizontal Dilution of Precision
0x07	uint8_t	GPS_NUM_OF_SATS	Number of satellites in view
0x08	float	GPS_FAIL_PERCENTAGE	Percentage of GPS checksums failed
0x09	uint16_t	CO2_CONCENTRATION	<i>UNUSED</i> CO ₂ concentration in ppb
0x0A	float	TEMPERATURE	Temperature in °C
0x0B	float	PRESSURE	Pressure in Pa
0x0C	uint16_t	DUST_CONCENTRATION	Dust concentration in $\mu\text{g}/\text{m}^3$
0x0D	float	UV_RADIATION	UV radiation in mW/cm^2
0x0E	uint16_t		Packet number

content_size: The number of bytes in the **content** field.

content: The actual data payload. Its interpretation depends on **content_type**.

3 Handling magic_number in Content

If the **magic_number** sequence 0x67616961 appears in the **content**, it must be escaped by appending a 0x00 byte immediately after. For example:

67 61 69 61 → 67 61 69 61 00

The escape byte contributes to **content_size** but should be removed during packet parsing.

4 Encoding order

The fields in a packet are encoded in the following order:

1. Check if the **magic_number** sequence appears in the **content** field. If so, escape it.
2. Calculate the content size.
3. Add the **magic_number** sequence, **content_type**, **content_size** and **content** fields.
4. Transmit the packet.

5 Examples

Single Value

Packet encoding a single unsigned 16-bit integer with value 4660:

67 61 69 61 0B 04 00 50 7D 44

Breakdown:

```
67 61 69 61 // Magic number
0B          // Content type: Pressure
04          // Content size: 4 bytes
00 50 7D 44 // Content: 1013.25 Pa
```

Array

Packet encoding GPS coordinates (latitude, longitude) as two 64-bit doubles:

67 61 69 61 03 0C 00 00 B4 42 9A 99 16 43 C3 F5 48 40

Breakdown:

```
67 61 69 61          // Magic number
03                   // Content type: Angle measurement (float ROTATION[3])
0C                   // Content size: 12 bytes
00 00 B4 42          // Angle 1: 90.0
9A 99 16 43          // Angle 2: 150.6
C3 F5 48 40          // Angle 3: 3.14
```

Escaped Magic Number

GPS coordinates with a `magic_number` sequence in the content:

```
67 61 69 61 01 0E 67 61 69 61 00 00 00 20 40 67 61 69 61 00
```

Breakdown:

```
67 61 69 61          // Magic Number
01                   // Content type: GPS coordinates (float GPS_POS[3])
0E                   // Content size: 14 bytes
67 61 69 61 00       // Latitude with escaped magic number: 2690693700000000000000 (this isn't a
00 00 20 40          // Longitude: 2.5
67 61 69 61 00       // Altitude with escaped magic number: 2690693700000000000000 (about 28441
```