Hardware interfaces and protocols of data exchange with Marvelmind devices

Version 2026.02.24
Valid for firmware v7.000 and newer

www.marvelmind.com

Table of contents

1. Connection to Marvelmind devices....4
   1.1 UART and other interfaces for Super-Beacon....5
   1.2 UART and other interfaces for beacon Mini-RX....6
   1.3. UART and other interfaces for beacon Mini-TX-2....7
   1.4. UART and other interfaces for Modem HW v5.1....8
   1.5. UART and other interfaces for Super-Modem....9
   1.6. UART and other interfaces for Modem HW v4.9....10
   1.7. UART and other interfaces for Industrial-TX, Industrial-RX, Industrial Super-Beacon...11
   1.8. UART and SPI interfaces for beacon HW v4.9....12
   1.9. UART and SPI interfaces for beacon HW v4.5....13
2. Protocols of communication via UART....14
   2.1 'Marvelmind' protocol for streaming....14
   2.2 Protocol of reading/writing data from/to user device....41
   2.3 NMEA0183 communication protocol....45
3. Protocols of communication via USB (virtual UART)....54
   3.1 'Marvelmind' protocol for streaming....54
   3.2 Protocol of reading/writing data from/to user device....55
   3.3 NMEA0183 communication protocol....56
   3.4 Protocol of data exchange with modem via USB interface....57
4. Protocols of communication via RS-485....84
   4.1 'Marvelmind' protocol for streaming....84
   4.2 Protocol of reading/writing data from/to user device....85
   4.3 NMEA0183 communication protocol....86
5. Protocols of communication via SPI....87
   5.1 Packet with hedgehog location....87
   5.2 Other data via SPI....88
6. Protocols of communication via I^2C....89
   6.1 Compass emulation for drones with PX4....89
   6.2 Other data via I^2C....90
7. Protocols of communication via UDP (Wi-Fi)....91
   7.1 Packet with hedgehog location....92
   7.1.1. Packet with hedgehog location with real-time timestamps (firmware v7.200+)....94
   7.2. Packet with stationary beacons locations....95
   7.3. Packet with raw IMU data....96
   7.3.1. Packet with raw IMU data with real-time timestamps (firmware v7.200+)....97
   7.4. Packet with raw distances data....98

7.4.1. Packet with raw distances data with real-time timestamps (firmware v7.200+).....99
7.5. Packet with IMU fusion data.....100
7.5.1. Packet with IMU fusion data with real-time timestamps (firmware v7.200+).....101
7.6. Packet with telemetry data.....102
7.7. Packet with quality and extended location data.....103
7.8. Packet with telemetry of all beacons.....104
7.9. NMEA0183 protocol.....105
8. Protocols of communication via CAN.....106
8.1. 'Marvelmind' protocol of streaming.....107
8.2. NMEA0183 communication protocol.....108
9. Format of dashboard csv log file.....109
9.1. Format of csv log file (dashboard version V7.000+).....110
9.2. Previous format of csv log (dashboard before V7.000 or modem HW v4.9).....124
10. Marvelmind API.....125
10.1. Installation for Windows.....126
10.2. Installation for Linux.....127
10.3. Check connection to API.....128
10.4. Marvelmind API library description.....129
10.5. Description of C example for Marvelmind API.....187
10.6. Device types.....195
11. Sending user data from/to user devices.....196
12. Contacts.....198
Appendix 1. Calculating CRC-16.....199
Appendix 2. Format of error reply from modem.....200

1. Connection to Marvelmind devices

For communication with Marvelmind devices (modem or mobile beacon (hedgehog)), it shall be connected to an external device (robot, copter, AGV, etc.) via any of the following interfaces:

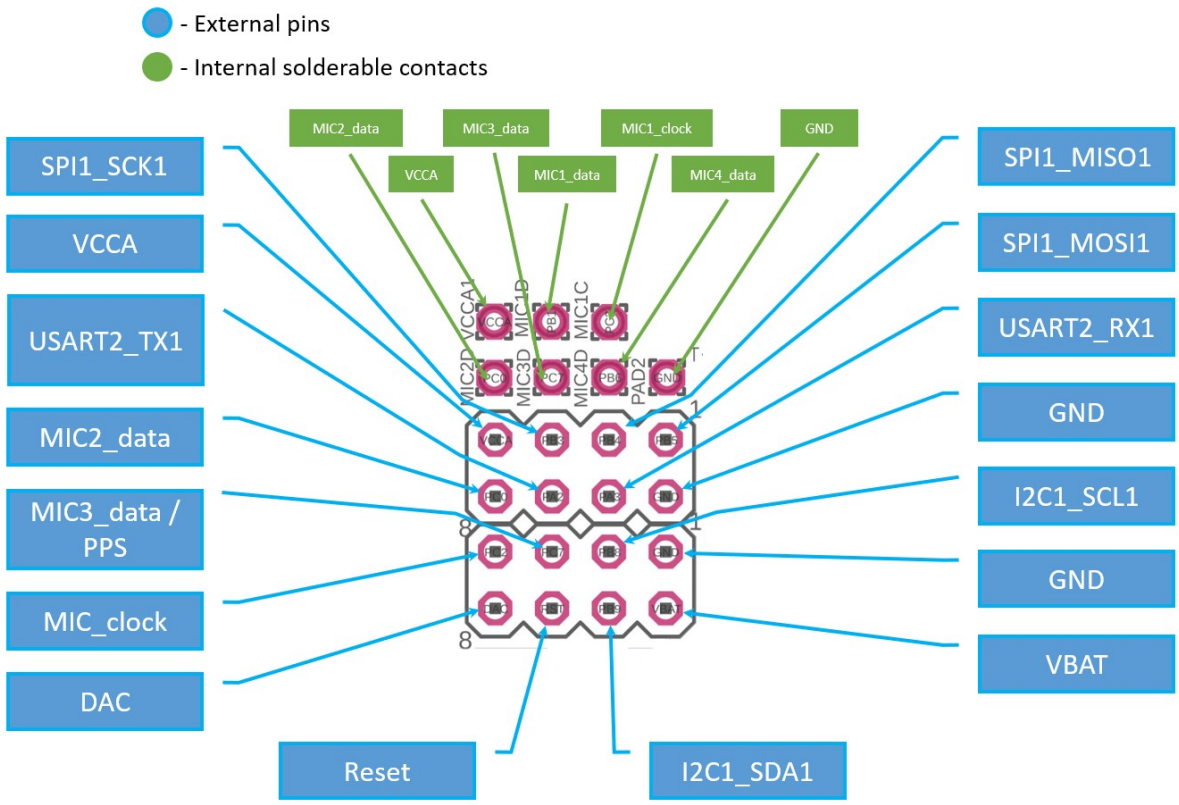
1. Connect to USB-host as an USB device of CDC class (virtual COM port in Windows, ttyACM or ttyUSB in Linux). In the Windows, it requires driver - the same driver as for modem. In Linux, the driver is not required, since the required driver is integrated into Linux kernel. Because real RS-232 is not used in the interface, parameters of serial port opened on the host (baudrate, number of bits, parity, etc) can be any.
2. Connect to UART – 2 wires soldering to pins for streaming or 3 wires for bidirectional communication required. See the picture of hardware interface below. Logic level of UART transmitter is CMOS 3.3V. Default baudrate is 500 kbps, configurable from the Dashboard from following list: 4.8, 9.6, 19.2, 38.4, 57.6, 115.2, 500 kbps. Format of data: 8 bit, no parity, 1 stop bit.
3. Connect to SPI. Marvelmind device acts as SPI slave device. Parameters of SPI: SPI mode 0, MSB inside each byte transmits first. Connection was tested on SCK speed up to 8 MHz. Be careful to provide quality wiring connections on high speeds (more than 500 kHz).
4. Connect to RS-485 (for Super-Modem or Industrial Super-Beacon only).
5. Connect to I2C (for Super-Beacon only).
6. Connect to UDP via Wi-Fi (for Super-Modem) or any network connection (for Dashboard).
7. Connect to CAN (for Industrial Super-Beacon, for Super-Modem supplied by request).

Connection settings summary:

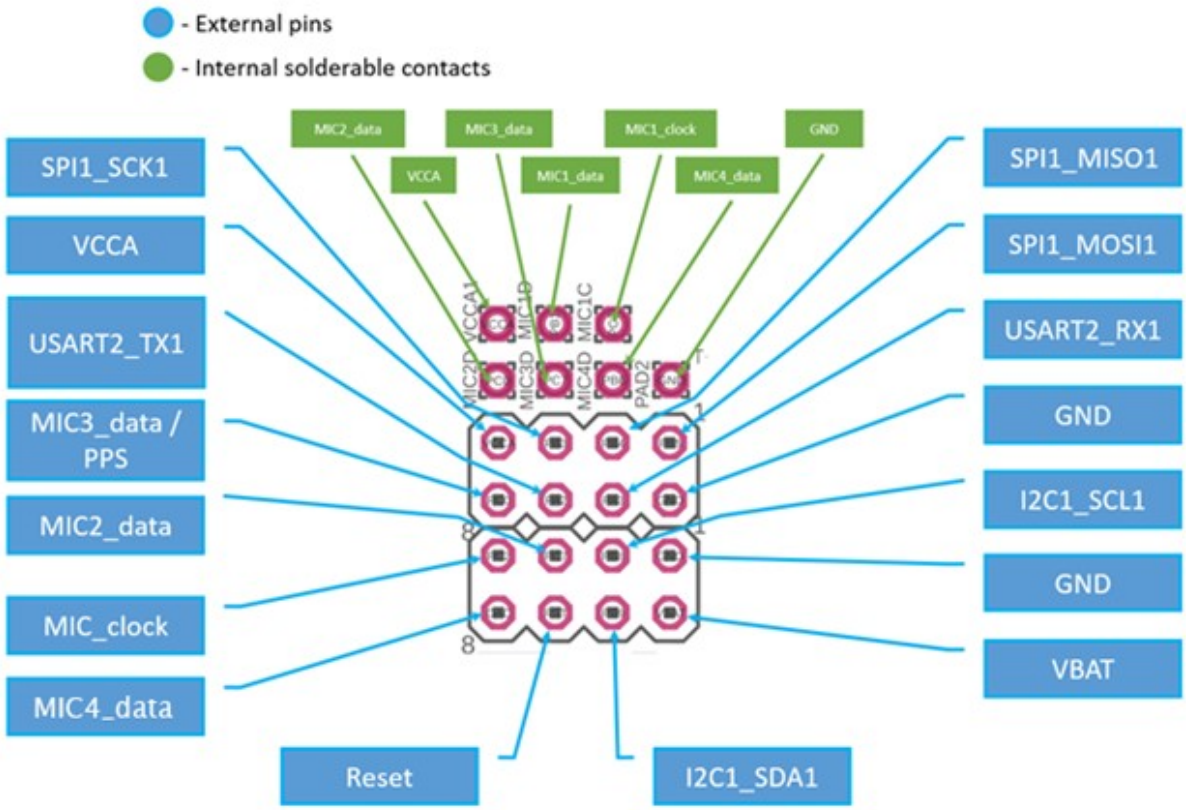
Interface	Bitrate	Other settings
USB (virtual UART)	UART bitrate is not applicable, serial port speed can be set to any value. Data transmitting via USB full speed (12 Mbit/s)	UART settings not applicable
UART	4.8, 9.6, 19.2, 38.4, 57.6, 115.2, 500 Kbit/s Can be selected in dashboard	8 bits of data, 1 stop bit, no parity
SPI	Tested up to 8 Mbit/s	SPI Mode 0
RS-485	4.8, 9.6, 19.2, 38.4, 57.6, 115.2, 500 Kbit/s Can be selected in dashboard (same as for UART)	8 bits of data, 1 stop bit, no parity
I2C	Up to 400 Kbit/s	
UDP	According to network connection speed	
CAN	125 Kbit/s	Standard frame

1.1 UART and other interfaces for Super-Beacon

4x4 pinout for Super-Beacon:



4x4 pinout for Super-Beacon-2 and Super-Beacon-3:



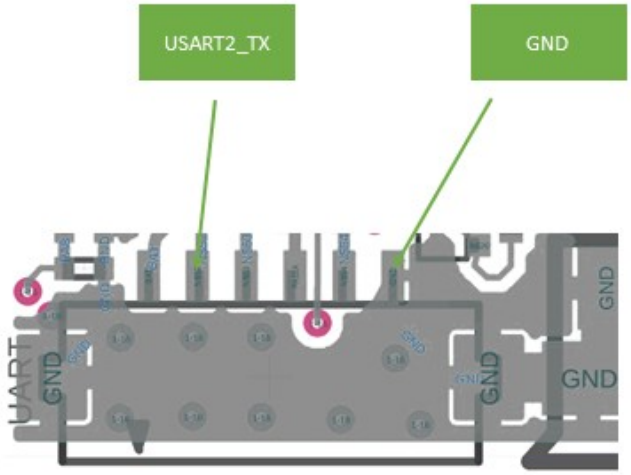
Marvelmind robotics

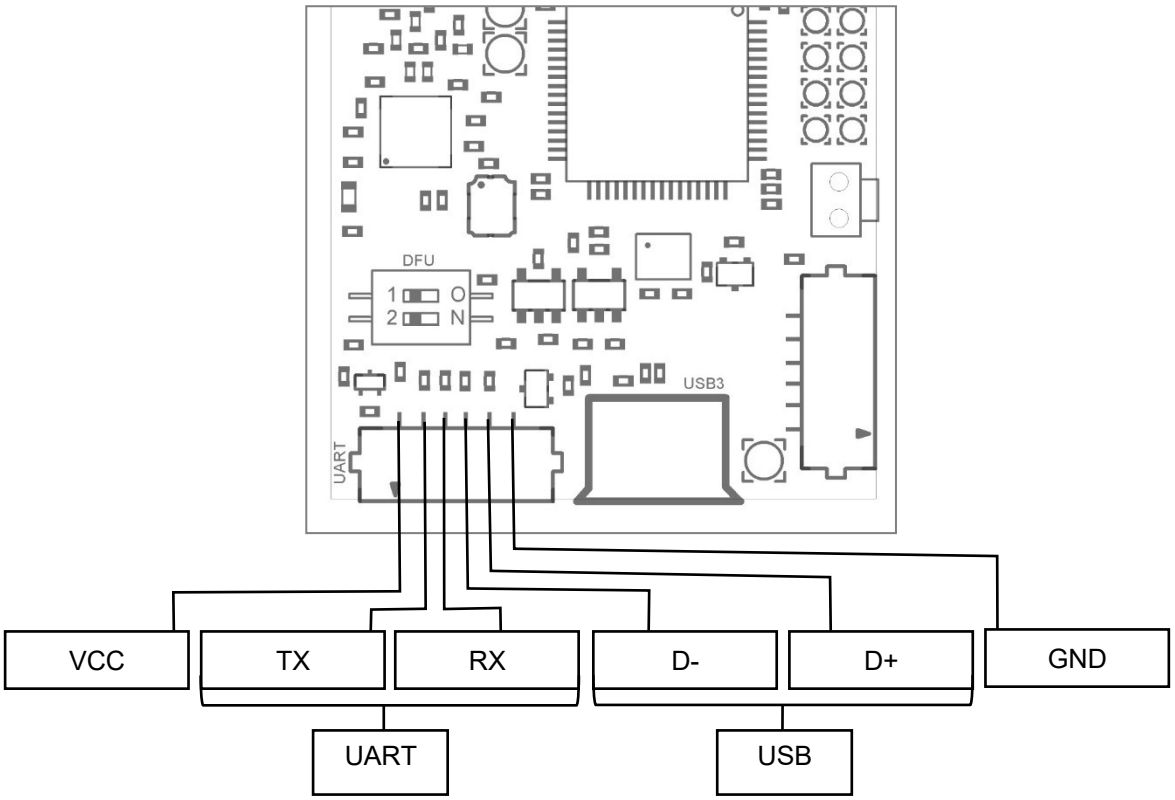
1.2 UART and other interfaces for beacon Mini-RX

Use it only if you sure that you can solder it correctlyDo not forget to turn off the beacon with DIP-switchesIf you solder bad and kill the beacon, Marvelmind team won't be responsible for it

To get UART data streaming from beacon Mini-RX, you must solder to the pins on the board.

- Internal solderable contacts



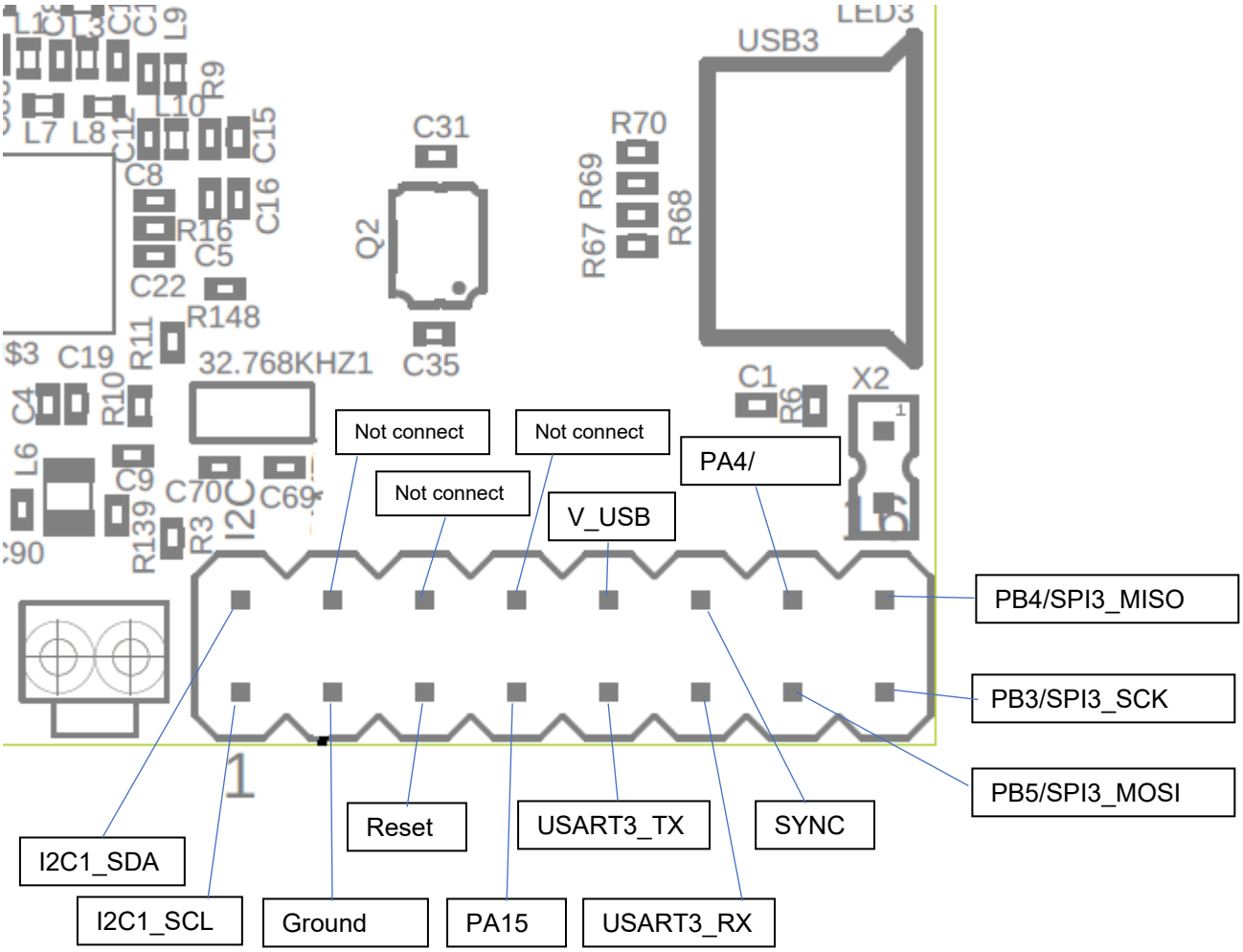


1.3. UART and other interfaces for beacon Mini-TX-2

Beacon Mini-TX-2 has a connector with the same pinout as beacon Mini-RX.

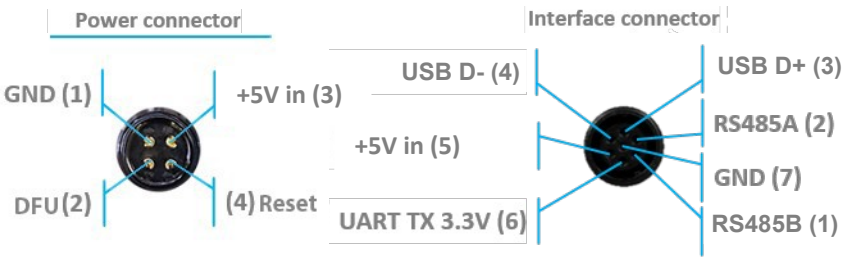
The UART cable can be used for connection to Mini-TX-2.

1.4. UART and other interfaces for Modem HW v5.1



Marvelmind robotics

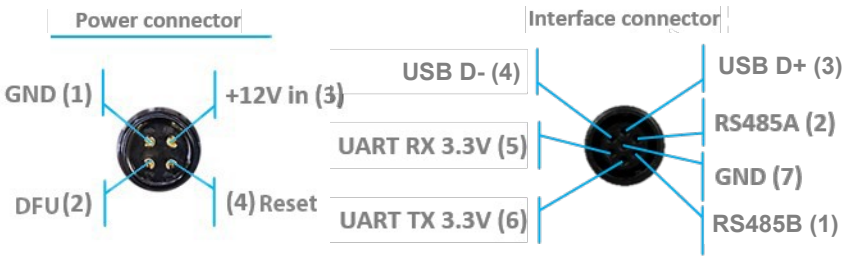
1.5. Super-Modem 的 UART 及其他接口



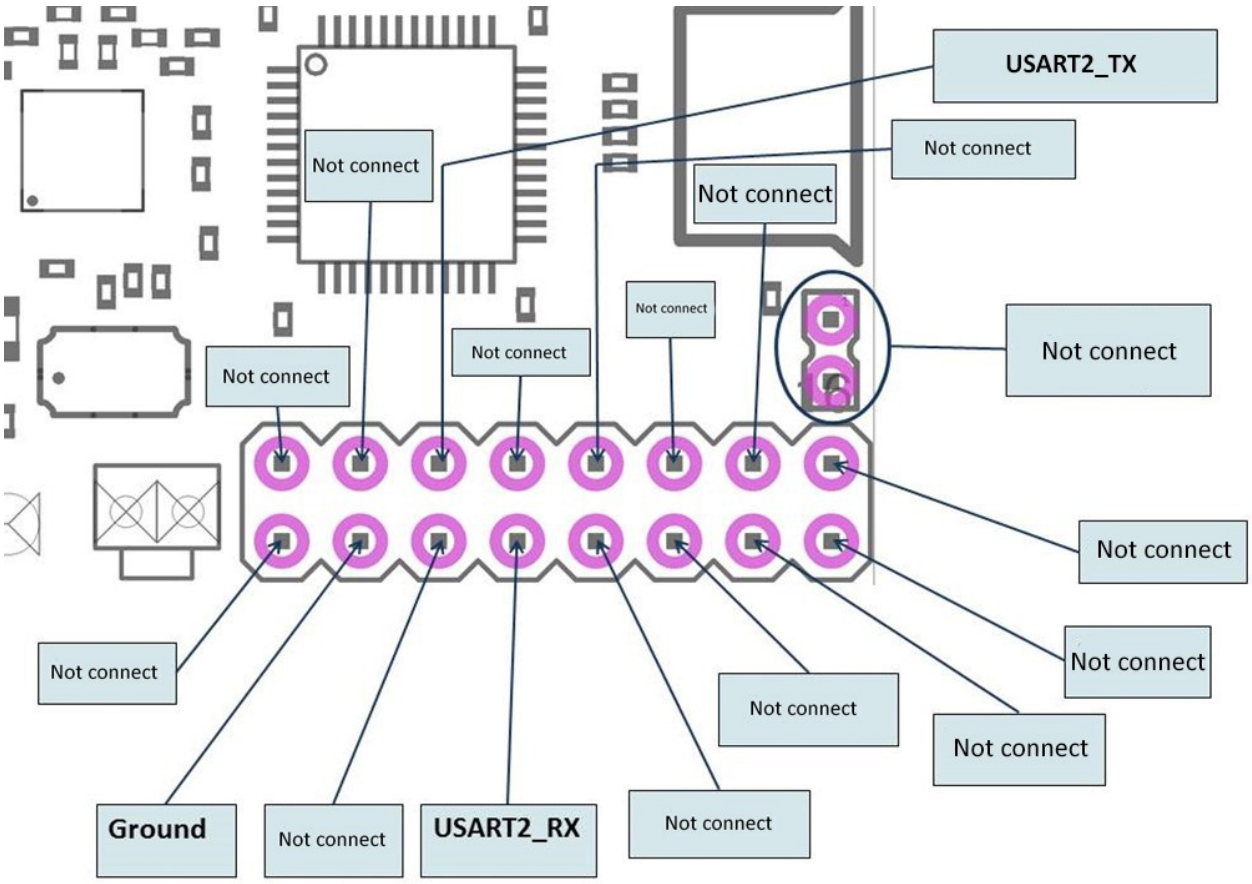
此外,Super-Modem 还包含板载 Wi-Fi 接口。Wi-Fi 连接的配置详见 UDP 章节。

- 新版 Super-Modem(2023 年 6 月起)仅支持 +5V 电源供电。请勿为此版本使用 +12V 电源转换器,否则会烧毁信标!

-



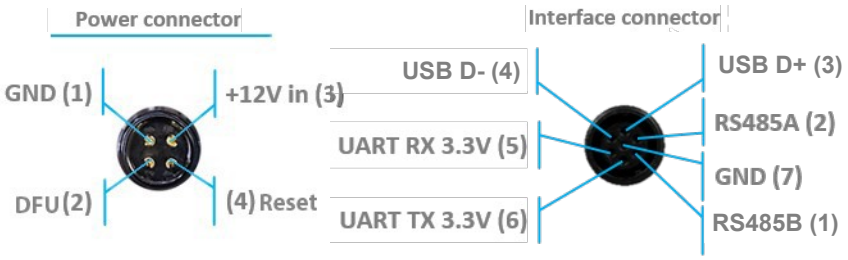
1.6. Modem HW v4.9 的 UART 和其他接口



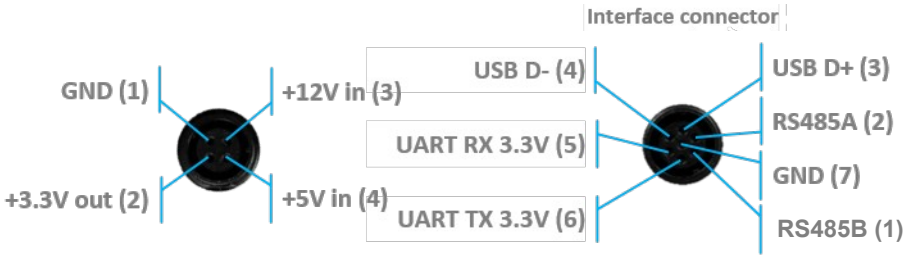
Marvelmind robotics

1.7. Industrial-TX、Industrial-RX、Industrial Super-Beacon 的 UART 和其他接口

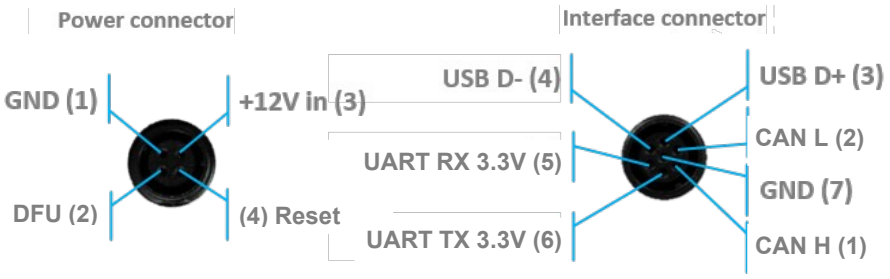
- Industrial-TX、Industrial-RX、Industrial Super-Beacon 的版本 2 和版本 3(自 2022 年 6 月起)仅支持 +5V 电源供电。请勿为此版本使用 +12V 电源转换器,否则将烧毁信标!
- 此版本没有 UART RX,但现在可以将接口连接器用作电源供电。



RS485 修改版引脚定义(2019 年 9 月之前)

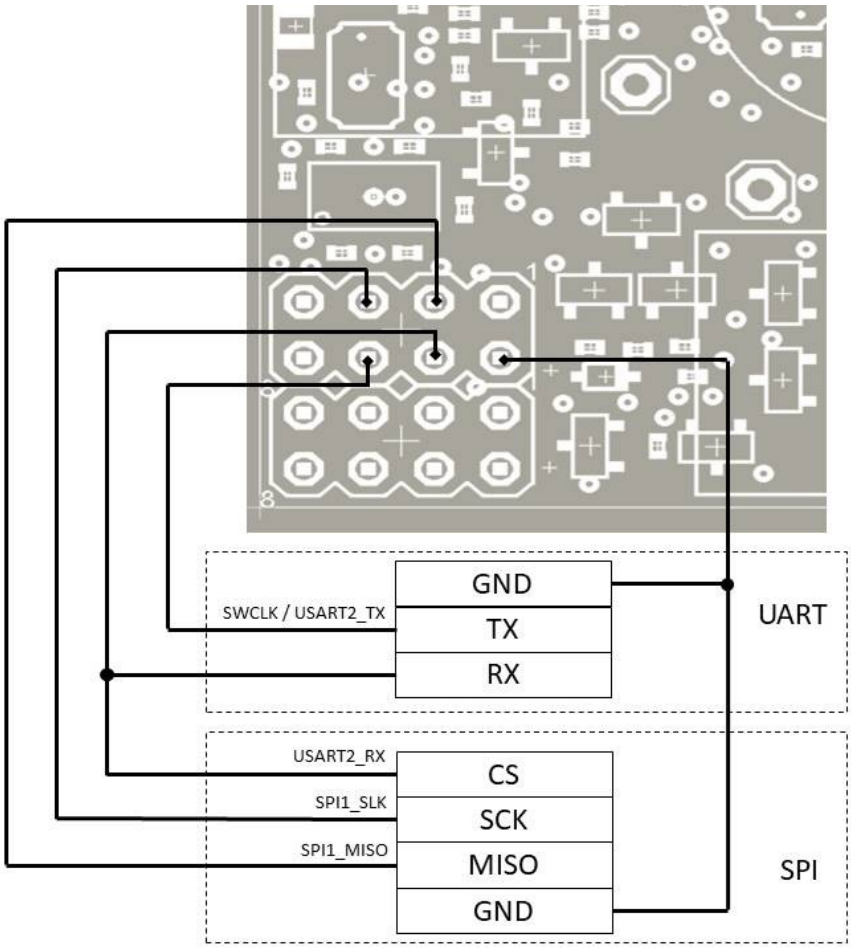


CAN 改装引脚定义



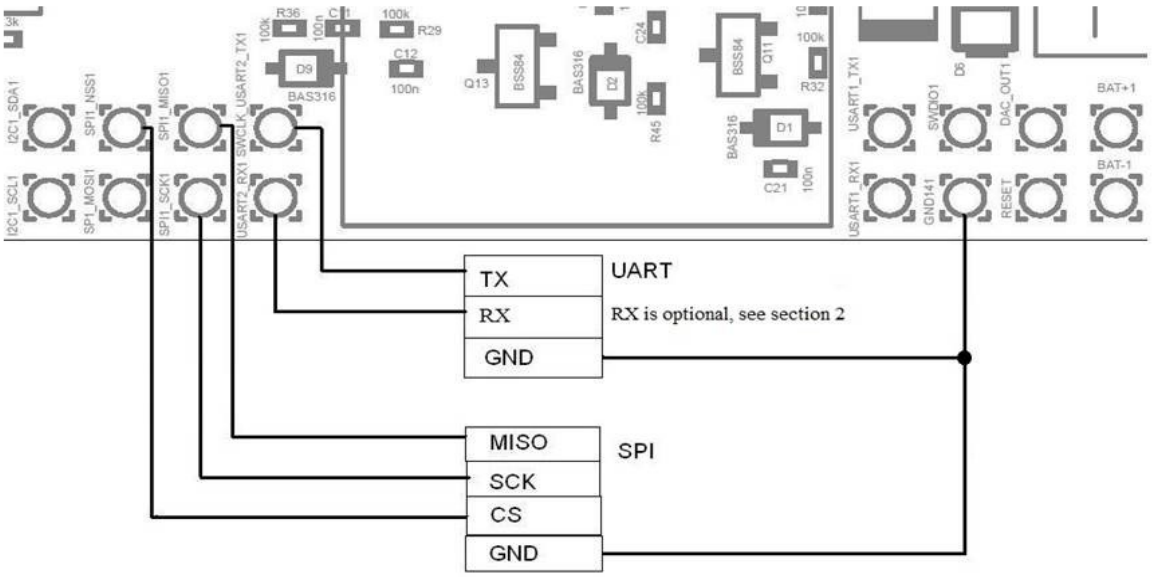
Marvelmind robotics

1.8. 信标 HW v4.9 的 UART 和 SPI 接口



注意：如您所见，UART RX 和 SPI CS 使用同一共享引脚。该引脚的功能（UART 接收器、SPI 片选或其他）可在 Dashboard 的 "Interfaces" 部分通过参数 "PA15 pin function" 进行选择。

1.9. 信标 HW v4.5 的 UART 和 SPI 接口



Marvelmind robotics

2. 通过 UART 的通信协议

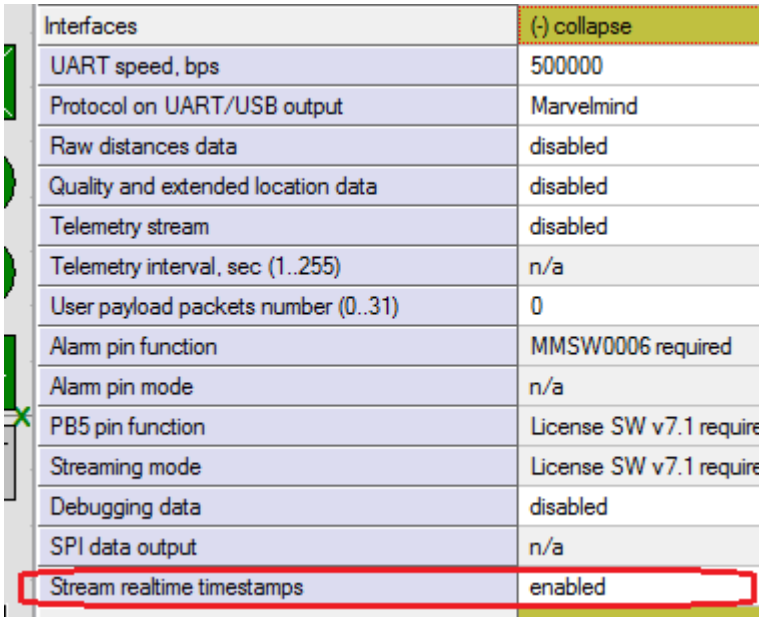
2.1 用于数据流的"Marvelmind"协议

所有数据流数据包具有相同的通用结构：

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	详见说明
4	1	uint8_t	传输数据的字节数	N
5	N	N 字节	根据数据字段代码的有效载荷数据	
5+N	2	uint16_t	CRC-16（见附录 1）	

从软件版本 v7.200 开始，默认启用实时时间戳。这意味着输出的数据包为 0x0081、0x0083、0x0084、0x0085，而不是对应的 0x0011、0x0003、0x0004、0x0005 数据包。

如果您需要使用带有本地时间戳的旧流格式以兼容较旧的软件，可以在 Dashboard 的设备设置中禁用此选项：



Interfaces	(-) collapse
UART speed, bps	500000
Protocol on UART/USB output	Marvelmind
Raw distances data	disabled
Quality and extended location data	disabled
Telemetry stream	disabled
Telemetry interval, sec (1..255)	n/a
User payload packets number (0..31)	0
Alarm pin function	MMSW0006 required
Alarm pin mode	n/a
PB5 pin function	License SW v7.1 require
Streaming mode	License SW v7.1 require
Debugging data	disabled
SPI data output	n/a
Stream realtime timestamps	enabled

2.1.1 Hedgehog 坐标数据包

每次测量新坐标或测量失败时都会传输此数据包。

2.1.1.1 带有移动信标 mm 分辨率坐标和实时时间戳的数据包（固件 V7.200+）

支持的硬件：

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持

Super-Modem：支持

Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持

Mini-TX：当前 HW 版本不支持

Mini-TX-2：通过 UART 线缆支持

Modem HW4.9：支持

Beacon HW4.9：支持

Beacon HW4.5：支持

请参阅有关时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0081
4	1	uint8_t	传输数据的字节数	N
5	8	int64_t	时间戳 – Unix 时间 - 自 1970.01.01 00:00:00 起的毫秒数。 由所有设备与 modem 及 dashboard 同步的时间。	
13	4	int32_t	信标的 X 坐标，mm	
17	4	int32_t	信标的 Y 坐标，mm	
21	4	int32_t	Z 坐标，信标高度，mm	
25	1	uint8_t	标志位字节： Bit 0：1 - 坐标不可用。不应使用 X、Y、Z 字段中的数据。 Bit 1：时间戳单位指示器（参见备注） Bit 2：1 - 用户按钮被按下（V5.23+） Bit 3：1 - 数据可上传至用户设备，参见第 2 节（V5.34+） Bit 4：1 - 希望从用户设备下载数据，参见第 2 节（V5.34+） Bit 5：1 – 第二个用户按钮被按下（V5.74+） Bit 6：1 – 数据属于另一台 hedgehog（非发送此数据包的同一台） Bit 7：– 1 – 超出地理围栏区域	
26	1	uint8_t	hedgehog 的地址	
27	2	uint16_t	Bit 0...11：hedgehog 对在 XY 平面内的方位，单位为十分之一度（0...3600） Bit 12：1 – 坐标为信标对的中心点坐标；0 – 坐标为指定 hedgehog 的坐标 Bit 13：1 - 方位不适用	

Marvelmind robotics

			Bit 14…15：保留（0）	
29	2	uint16_t	从超声波发射到当前时间所经过的时间，单位为毫秒（V5.88+）	
31	M=N-26		可选数据 – 见列表	
31+M	2	uint16_t	CRC-16（见附录 1）	

移动信标位置数据包中的可选数据可以包含以下结构：

- 速度数据（7 字节）。需要在 Dashboard 中移动信标设置的接口部分启用

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	数据字段代码 = 1 表示速度矢量	**1**
1	2	int16_t	沿 X 方向的速度，mm/sec	
3	2	int16_t	沿 Y 方向的速度，mm/sec	
5	2	int16_t	沿 Z 方向的速度，mm/sec	

2.1.1.2 包含移动信标厘米级分辨率坐标的数据包

已废弃的数据包，由 0x0081 替代。有关此废弃数据包的详细信息请查看此处。

2.1.1.3 包含移动信标毫米级分辨率坐标的数据包

已废弃的数据包，由 0x0081 替代。有关此废弃数据包的详细信息请查看此处。

2.1.2 固定信标坐标数据包

当地图被冻结时传输此数据包，每 10 sec 重复一次。

2.1.2.1 包含固定信标毫米级分辨率坐标的数据包，数据代码 0x0012（固件 V5.35+）

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：支持通过 UART 电缆使用
Mini-TX：当前硬件版本不支持
Mini-TX-2：支持通过 UART 电缆使用
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0012
4	1	uint8_t	传输数据的字节数	1+N*14
5	1	uint8_t	数据包中的信标数量	N
6	1	N*14 字节	N 个信标的数据	
6+N*14	2	uint16_t	CRC-16（参见附录 1）	

N 个信标中每个信标的数据结构格式：

偏移量	大小（字节）	类型	描述
0	1	uint8_t	信标地址
1	4	int32_t	信标的 X 坐标，mm
5	4	int32_t	信标的 Y 坐标，mm
9	4	int32_t	Z 坐标，信标高度，mm
13	1	uint8_t	位 0：1 = 位置不可用 位 1…7：保留

Marvelmind robotics

2.1.2.2 含固定信标厘米级分辨率坐标的数据包，数据代码 0x0002。

已废弃的数据包，已由 0x0012 替代。废弃数据包的详细信息请参见此处。

2.1.3 惯性传感器原始数据包

当有新的惯性传感器数据可用时，发送此数据包。

2.1.3.1 带实时时间戳的惯性传感器原始数据包，数据代码 0x0083（固件 V7.200+）

当有新的惯性传感器数据可用时，发送此数据包。

支持的硬件：

Super-Beacon：支持，100 Hz（如启用"Raw inertial sensors data"）
Industrial Super-Beacon：支持，100 Hz（如启用"Raw inertial sensors data"）
Modem HW5.1：支持，系统刷新率（如启用"IMU via modem"）
Super-Modem：支持，系统刷新率（如启用"IMU via modem"）
Mini-RX（Badge、Helmet 等）：支持，100 Hz（如启用"Raw inertial sensors data"）
通过 UART 电缆
Mini-TX：当前 HW 版本不支持
Mini-TX-2：支持，100 Hz（如启用"Raw inertial sensors data"）
Modem HW4.9：支持，系统更新速率（如启用"IMU via modem"）
Beacon HW4.9：支持，100 Hz（如启用"Raw inertial sensors data"）
Beacon HW4.5：支持，100 Hz（如启用"Raw inertial sensors data"）

请参阅有关时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中数据的代码	0x0083
4	1	uint8_t	传输数据的字节数	
5	36		数据包（见下文）	
41	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	说明	值
0	2	int16_t	加速度计，X 轴，1 mg/LSB	
2	2	int16_t	加速度计，Y 轴，1 mg/LSB	
4	2	int16_t	加速度计，Z 轴，1 mg/LSB	
6	2	int16_t	陀螺仪，X 轴，0.0175 dps/LSB	
8	2	int16_t	陀螺仪，Y 轴，0.0175 dps/LSB	
10	2	int16_t	陀螺仪，Z 轴，0.0175 dps/LSB	
12	2	int16_t	罗盘，X 轴，1100 LSB/Gauss	
14	2	int16_t	罗盘，Y 轴，1100 LSB/Gauss	
16	2	int16_t	罗盘，Z 轴，980 LSB/Gauss	
18	1	uint8_t	信标地址	
19	5	5 字节	保留 (0)	
24	8	int64_t	时间戳 – Unix 时间 - 从 1970.01.01	

			00:00:00 起的毫秒数。 由所有设备与 Modem 及 Dashboard 同步的时间。	
32	1	uint8_t	标志位： 位 0： 1 = 加速度计数据不可用 位 1： 1 = 陀螺仪数据不可用 Bit 2： 1 = Compass 数据不可用 Bit 3…7 – 保留 (0)	
33	3	3 字节	保留	

注意：Compass 数据仅适用于配备 IMU 的 HW v4.9 beacon。

2.1.3.2 原始惯性传感器数据包，数据代码 0x0003

过时数据包，已被 0x0083 替代。过时数据包的详细信息请查看此处。

2.1.4 原始距离数据包

2.1.4.1 带实时时间戳的原始距离数据包，数据代码 0x0084（固件 V7.200+）

此数据包在每次测量新坐标或坐标测量失败时，紧随移动 beacon 坐标数据包之后发送。仅在设置中 'Interfaces' 部分启用 "raw distances data" 选项时可用。

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 线缆支持
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

请参阅关于时间戳的说明。

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0084
4	1	uint8_t	传输数据的字节数	
5	36		数据包（见下文）	
41	2	uint16_t	CRC-16（参见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Hedgehog 地址	
1	6		距离项 1	
7	6		距离项 2	
13	6		距离项 3	
19	6		距离项 4	
25	8	int64_t	时间戳 – beacon 超声波发射的 unix 时间，从 1970.01.01 00:00:00 开始的毫秒数。 时间，通过 Modem 和 Dashboard 在所有设备间同步。	
29	2	uint16_t	从超声波发射到当前时刻所经过的时间，单位为毫秒（V5.89+）	
31	1	uint8_t	保留	

距离项格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址（如该项未填则为 0）	
1	4	uint32_t	到信标的距离，mm	
5	1	uint8_t	第 0 位：1 = 距离不适用 第 1…7 位：保留（0）	

Marvelmind robotics

2.1.4.2 带候选数据和实时时间戳的原始距离数据包，数据代码 0x0094（固件 V8.431+）

该数据包在每次测量到新坐标或测量失败时发送，紧随移动信标坐标数据包之后。

仅当在设置的 'Interfaces' 部分启用 "extra raw distances data" 选项时可用。

支持的硬件：

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：按需提供

Super-Modem：按需提供

Mini-RX（Badge、Helmet 等）：通过 UART 电缆支持

Mini-TX：不支持

Mini-TX-2：按需提供

Modem HW4.9：不支持

Beacon HW4.9：不支持

Beacon HW4.5：不支持

请参阅有关时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0094
4	1	uint8_t	传输数据的字节数	
5	80		数据包（见下文）	
85	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	Hedgehog 地址	
1	17		距离项 1	
18	17		距离项 2	
35	17		距离项 3	
52	17		距离项 4	
69	8	int64_t	时间戳 – 信标超声波发射的 unix 时间，自 1970.01.01 00:00:00 起的毫秒数。 由所有设备与 modem 和 dashboard 同步的时间。	
77	2	uint16_t	从超声波发射到当前时刻所经过的时间，单位毫秒（V5.89+）	
79	1	uint8_t	保留	

距离项的格式

偏移	大小（字	类型	描述	数值

Marvelmind robotics

量	节)			
0	1	uint8_t	信标地址（若项未填充则为 0）	
1	1	uint8_t	距离候选值数量	
2	4	uint32_t	到信标的第一个距离候选值，单位 mm	
6	1	uint8_t	第一候选的质量，%	
7	4	uint32_t	到信标距离的第二候选，mm	
11	1	uint8_t	第二候选的质量，%	
12	4	uint32_t	到信标距离的第三候选，mm	
16	1	uint8_t	第三候选的质量，%	

2.1.4.3 原始距离数据包，数据代码 0x0004

已过时的数据包，由 0x0084 替代。有关此过时数据包的详细信息请见此处。

2.1.5 已处理 IMU 数据包

当有新的惯性传感器数据可用时发送此数据包。

2.1.5.1 带实时时间戳的已处理 IMU 数据包，数据代码 0x0085（固件 V7.200+）

当有新的惯性传感器数据可用时发送此数据包。

支持的硬件：

Super-Beacon：支持，100 Hz（若启用 "Processed IMU data"）

Industrial Super-Beacon：支持，100 Hz（若启用 "Processed IMU data"）

Modem HW5.1：支持，系统更新速率（若启用 "IMU via modem"）

Super-Modem：支持，系统更新速率（若启用 "IMU via modem"）

Mini-RX（Badge、Helmet 等）：支持，100 Hz（若启用"Processed IMU data"）
使用 Mini-Rx 的 UART 电缆

Mini-TX：当前 HW 版本不支持

Mini-TX-2：支持，100 Hz（若启用"Processed IMU data"）

Modem HW4.9：支持，系统更新速率（若启用"IMU via modem"）

Beacon HW4.9：支持，100 Hz（若启用"Processed IMU data"）

Beacon HW4.5：支持，100 Hz（若启用"Processed IMU data"）

请参阅关于时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0085
4	1	uint8_t	传输数据的字节数	
5	46		数据包（见下文）	
51	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	说明	数值
0	4	int32_t	信标坐标 X（融合），mm	
4	4	int32_t	信标坐标 Y（融合），mm	
8	4	int32_t	信标坐标 Z（融合），mm	
12	2	int16_t	旋转四元数的 W 分量（角度）	
14	2	int16_t	旋转四元数的 X 分量（角度）	
16	2	int16_t	旋转四元数的 Y 分量（角度）	
18	2	int16_t	旋转四元数的 Z 分量（角度）	
20	2	int16_t	信标的 X 方向速度（融合），mm/s	
22	2	int16_t	信标的 Y 方向速度（融合），mm/s	
24	2	int16_t	信标的 Z 方向速度（融合），mm/s	
26	2	int16_t	信标的 X 方向加速度，mm/s2	
28	2	int16_t	信标的 Y 方向加速度，mm/s2	
30	2	int16_t	信标的 Z 方向加速度，mm/s2	
32	1	uint8_t	信标地址	
33	1	1 字节	保留（0）	

Marvelmind robotics

34	8	int64_t	时间戳 – Unix 时间 - 自 1970.01.01 00:00:00 起的毫秒数。 时间，由所有设备与 Modem 和 Dashboard 同步。	
42	1	uint8_t	标志位： 位 0：1 = 位置数据不可用 位 1：1 = 四元数数据不可用 位 2：1 = 速度数据不可用 位 3：1 = 加速度数据不可用	
43	3	3 字节	保留（0）	

注意：四元数归一化至 10000 值。

2.1.5.2 已处理 IMU 数据包，数据代码 0x0005

已弃用数据包，由 0x0085 替代。有关已弃用数据包的详细信息，请参阅此处。

2.1.6 遥测数据包（数据代码 0x0006）

如果在设置的 "Interfaces" 部分中启用了 "Telemetry stream" 选项，则该数据包将在位置更新后发送。

支持的硬件：

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持（固件 V7.000+）

Super-Modem：支持（固件 V7.000+）

Mini-RX（Badge、Helmet 等）：通过 UART 电缆支持

Mini-TX：当前 HW 版本不支持

Mini-TX-2：通过 UART 电缆支持

Modem HW4.9：不支持

Beacon HW4.9：支持

Beacon HW4.5：支持

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0006
4	1	uint8_t	传输数据的字节数	
5	16		数据包（见下文）	
21	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	数值
0	2	uint16_t	电池电压，mV	
2	1	int8_t	RSSI，dBm	
3	1	uint8_t	信标地址	
4	12		保留 (0)	

Marvelmind robotics

2.1.7 质量及扩展位置数据包（数据代码 0x0007）

如果在设置的"Interfaces"部分启用了"Quality and extended location data"选项，则该数据包将在位置更新后传输。

支持的硬件：

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持

Super-Modem：支持

Mini-RX（Badge、Helmet 等）：需配合 UART 电缆使用

Mini-TX：当前 HW 版本不支持

Mini-TX-2：需配合 UART 电缆使用

Modem HW4.9：支持（仅质量字段）

Beacon HW4.9：支持（仅质量字段）

Beacon HW4.5：支持（仅质量字段）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0007
4	1	uint8_t	传输数据的字节数	
5	16		数据包（见下文）	
21	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	设备地址	
1	1	uint8_t	定位质量，%	
2	1	uint8_t	0 = 无地理围栏区域报警 1...255 - 地理围栏区域索引 此字段需要 MMSW0005 许可证。	
3	13		保留 (0)	

Marvelmind robotics

2.1.8 已停用功能

此处描述的是旧版软件中使用的过时数据包，默认情况下已被其他数据包取代。

2.1.8.1 带有移动信标厘米级分辨率坐标的数据包

过时数据包。已被 0x0081 取代。

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 线缆支持
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中数据的代码	0x0001
4	1	uint8_t	传输数据的字节数	0x10
5	4	uint32_t	时间戳 – 信标超声波发射的内部时间，单位为毫秒，从最近一次唤醒事件开始计算。参见注释。	
9	2	int16_t	信标的 X 坐标，单位 cm	
11	2	int16_t	信标的 Y 坐标，单位 cm	
13	2	int16_t	Z 坐标，信标高度，单位 cm	
15	1	uint8_t	标志位字节： 位 0：1 - 坐标不可用。不应使用 X、Y、Z 字段的数据。 位 1：时间戳单位指示符（参见注释） 位 2：1 - 用户按钮被按下（V5.23+） 位 3：1 - 数据可上传至用户设备，参见第 2 节（V5.34+） 位 4：1 - 需要从用户设备下载数据，参见第 2 节（V5.34+） 位 5：1 – 第二个用户按钮被按下（V5.74+） 位 6：1 – 数据用于另一个 hedgehog（非发送该数据包的同一个） 位 7：– 保留（0）	
16	1	uint8_t	hedgehog 地址	
17	2	uint16_t	位 0...11：hedgehog 对在 XY 平面内的方位，单位为分度（0...3600） 位 12：1 – 坐标为信标对中心位置；0 – 坐标为指定信标位置 位 13：1 - 方位不适用	

			位 14…15：保留（0）	
19	2	uint16_t	从发射超声波到当前时间所经过的时间，单位为毫秒（V5.88+）	
21	2	uint16_t	CRC-16（见附录 1）	

2.1.8.2 含移动信标毫米级分辨率坐标的数据包

已过时的数据包。已由 0x0081 替代。

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-Rx（Badge、Helmet 等）：通过 UART 电缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 电缆支持
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

请参阅关于时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0011
4	1	uint8_t	传输数据的字节数	N
5	4	uint32_t	时间戳——信标超声波发射的内部时间，以毫秒为单位，从最近一次唤醒事件开始计时。参见注释。	
9	4	int32_t	信标的 X 坐标，单位 mm	
13	4	int32_t	信标的 Y 坐标，单位 mm	
17	4	int32_t	Z 坐标，信标高度，单位 mm	
21	1	uint8_t	标志字节： Bit 0: 1 - 坐标不可用。不应使用 X、Y、Z 字段中的数据。 Bit 1: 时间戳单位指示符（参见注释） Bit 2: 1 - 用户按钮被按下（V5.23+） Bit 3: 1 - 数据可上传至用户设备，参见第 2 节（V5.34+） Bit 4: 1 - 希望从用户设备下载数据，参见第 2 节（V5.34+） Bit 5: 1 – 第二个用户按钮被按下（V5.74+） Bit 6: 1 – 数据用于另一个 hedgehog（并非发送此数据包的同一个） Bit 7: – 1 – 超出地理围栏区域	
22	1	uint8_t	hedgehog 地址	
23	2	uint16_t	位 0...11：hedgehog 对在 XY 平面内的方位，单位为 0.1 度（0...3600） 位 12：1 – 坐标为信标对中心的坐标；0 – 坐标为指定 hedgehog 的坐标	

			位 13：1 - 方位不适用 位 14…15：保留（0）	
25	2	uint16_t	从超声波发射到当前时刻所经过的时间，单位为毫秒（V5.88+）	
27	M= N-22		可选数据 – 参见列表	
27+M	2	uint16_t	CRC-16（参见附录 1）	

注意：对于 V5.20 之前的固件版本，时间戳以 1/64 秒为单位，时间戳单位指示位（标志字节的位 1）为 0。对于 5.20 及更高版本，时间戳以毫秒为单位，时间戳单位指示位为 1。

2.1.8.3 以厘米分辨率表示的固定信标坐标数据包，数据代码为 0x0002。

已弃用的数据包。已由 0x0012 替代。

支持的硬件:

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：通过 UART 电缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 电缆支持
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

偏移量	大小（字节)	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0002
4	1	uint8_t	传输数据的字节数	1+N*8
5	1	uint8_t	数据包中信标的数量	N
6	1	N*8 字节	N 个信标的数据	
6+N*8	2	uint16_t	CRC-16(参见附录 1)	

N 个信标中每个信标的数据结构格式:

偏移量	大小(字节)	类型	说明
0	1	uint8_t	信标地址
1	2	int16_t	信标的 X 坐标，cm
3	2	int16_t	信标的 Y 坐标，cm
5	2	int16_t	Z 坐标，信标的高度，cm
7	1	uint8_t	保留 (0)

2.1.8.4 原始惯性传感器数据包，数据代码 0x0003

已废弃数据包。由 0x0083 替代。

支持的硬件:

Super-Beacon：支持，100 Hz（若启用"Raw inertial sensors data"）
Industrial Super-Beacon：支持，100 Hz（若启用"Raw inertial sensors data"）
Modem HW5.1：支持，系统更新速率（若启用"IMU via modem"）
Super-Modem：支持，系统更新速率（若启用"IMU via modem"）
Mini-RX（Badge、Helmet 等）：支持，100 Hz（若启用"Raw inertial sensors data"）
使用 UART 电缆
Mini-TX：当前 HW 版本不支持
Mini-TX-2：支持，100 Hz（若启用"Raw inertial sensors data"）
Modem HW4.9：支持，系统更新速率（若启用"IMU via modem"）
Beacon HW4.9：支持，100 Hz（若启用"Raw inertial sensors data"）
Beacon HW4.5：支持，100 Hz（若启用"Raw inertial sensors data"）

请参阅关于时间戳的说明。

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中数据代码	0x0003
4	1	uint8_t	传输数据的字节数	
5	32		数据包（见下方）	
37	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	2	int16_t	加速度计，X 轴，1 mg/LSB	
2	2	int16_t	加速度计，Y 轴，1 mg/LSB	
4	2	int16_t	加速度计，Z 轴，1 mg/LSB	
6	2	int16_t	陀螺仪，X 轴，0.0175 dps/LSB	
8	2	int16_t	陀螺仪，Y 轴，0.0175 dps/LSB	
10	2	int16_t	陀螺仪，Z 轴，0.0175 dps/LSB	
12	2	int16_t	罗盘,X 轴,1100 LSB/Gauss	
14	2	int16_t	罗盘,Y 轴,1100 LSB/Gauss	
16	2	int16_t	罗盘,Z 轴,980 LSB/Gauss	
18	1	uint8_t	信标地址	
19	5	5 字节	保留 (0)	
24	4	uint32_t	时间戳,ms	
28	1	uint8_t	标志位: Bit 0:1 = 加速度计数据不可用 Bit 1:1 = 陀螺仪数据不可用	

			Bit 2:1 = 罗盘数据不可用 Bit 3...7 – 保留 (0)	
29	3	3 字节	保留	

注意:罗盘数据仅适用于带 IMU 的 HW v4.9 信标。

2.1.8.5 原始距离数据包，数据代码 0x0004

已废弃的数据包。由 0x0084 替代。

支持的硬件:

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：通过 UART 电缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 电缆支持
Modem HW4.9：支持
Beacon HW4.9：支持
Beacon HW4.5：支持

请参阅关于时间戳的说明。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0004
4	1	uint8_t	传输的数据字节数	
5	32		数据包（见下文）	
37	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	hedgehog 地址	
1	6		距离项 1	
7	6		距离项 2	
13	6		距离项 3	
19	6		距离项 4	
25	4	uint32_t	时间戳 — 信标超声波发射的内部时间，单位为毫秒，从最近一次唤醒事件起计算（V5.89+）。	

29	2	uint16_t	从超声波发射到当前时间所经过的时间，单位为毫秒（V5.89+）	
31	1	uint8_t	保留	

距离项格式

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	信标地址（若该项未填充则为 0）	
1	4	uint32_t	到信标的距离，mm	
5	1	uint8_t	位 0：1 = 距离不适用 位 1…7：保留（0）	

Marvelmind robotics

2.1.8.6 处理后的 IMU 数据包，数据代码 0x0005

已废弃数据包。由 0x0085 替代。

支持的硬件：

Super-Beacon：支持角度，100 Hz（如果启用"Processed IMU data"）
Industrial Super-Beacon：支持角度，100 Hz（如果启用"Processed IMU data"）
Modem HW5.1：支持，系统更新率（如果启用"IMU via modem"）
Super-Modem：支持，系统更新率（如果启用"IMU via modem"）
Mini-RX（Badge 等）：支持角度，100 Hz（如果启用"Processed IMU data"）
配合 Mini-Rx 的 UART 线缆
Mini-TX：当前 HW 版本不支持
Mini-TX-2：支持角度，100 Hz（如果启用"Processed IMU data"）
Modem HW4.9：支持，系统更新率（如果启用"IMU via modem"）
Beacon HW4.9：支持，100 Hz（如果启用"Processed IMU data"）
Beacon HW4.5：支持，100 Hz（如果启用"Processed IMU data"）

请参阅关于时间戳的说明。

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0005
4	1	uint8_t	传输数据的字节数	
5	42		数据包（见下文）	
47	2	uint16_t	CRC-16（见附录 1）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	4	int32_t	信标 X 坐标（融合），mm	
4	4	int32_t	信标 Y 坐标（融合），mm	
8	4	int32_t	信标 Z 坐标（融合），mm	
12	2	int16_t	旋转四元数 W 分量（角度）	
14	2	int16_t	旋转四元数 X 分量（角度）	
16	2	int16_t	旋转四元数 Y 分量（角度）	
18	2	int16_t	旋转四元数 Z 分量（角度）	
20	2	int16_t	信标 X 方向速度（融合），mm/s	
22	2	int16_t	信标的 Y 轴速度（融合），mm/s	
24	2	int16_t	信标的 Z 轴速度（融合），mm/s	
26	2	int16_t	信标的 X 轴加速度，mm/s2	
28	2	int16_t	信标的 Y 轴加速度，mm/s2	
30	2	int16_t	信标的 Z 轴加速度，mm/s2	
32	1	uint8_t	信标地址	
33	1	1 字节	保留（0）	

34	4	uint32_t	时间戳，ms	
38	1	uint8_t	标志位： Bit 0： 1 = 位置数据不可用 Bit 1： 1 = 四元数数据不可用 Bit 2： 1 = 速度数据不可用 Bit 3： 1 = 加速度数据不可用 Bit 4…7 – 保留 (0)	
39	3	3 字节	保留 (0)	

注意：四元数已归一化为 10000 值

2.2 从用户设备读取数据/向用户设备写入数据的协议

2.2.1 从用户设备发送数据

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 线缆支持
Modem HW4.9：不支持
Beacon HW4.9：支持
Beacon HW4.5：支持

如果用户设备需要通过 Marvelmind 系统传输数据，应发送以下帧：

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0x00
1	1	uint8_t	数据包类型	0x49
2	2	uint16_t	数据包中的数据代码	0x0200
4	1	uint8_t	传输数据的字节数	N
5	N	N 字节	有效载荷数据	
5+N	2	uint16_t	CRC-16（见附录 1）	

数据将通过无线电按照 Dashboard 中 Hedgehog 设置里"Interfaces"部分定义的"User payload data size"大小分块传输至 Modem。这些数据块的发送速率等于 Hedgehog 的更新速率。Hedgehog 中的缓冲区大小为 128 字节。请注意这一点以避免缓冲区溢出。

2.2.2 向用户设备写入数据

该数据包由 Marvelmind 设备（Modem 或移动信标）传输至用户设备。

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x4a
2	2	uint16_t	数据包中的数据代码	0x0200... 0x02ff
4	1	uint8_t	传输数据的字节数	N
5	N	N 字节	有效载荷数据	
5+N	2	uint16_t	CRC-16(参见附录 1)	

对于此命令,数据代码 0x200 至 0x2ff 为保留值。

如果用户设备成功处理了该请求,则应按以下格式发送响应:

偏移量	大小(字节)	类型	描述	值
0	1	uint8_t	Hedgehog 地址(可从流传输的 0x0001 或 0x0011 数据包中获取)	
1	1	uint8_t	数据包类型	0x4a
2	2	uint16_t	数据包中的数据代码	0x0200... 0x02ff
4	2	uint16_t	CRC-16（见附录 1）	

如果用户设备未能处理请求，则以以下格式发送响应：

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	Hedgehog 地址（从流式传输的 0x0001 数据包获取）	
1	1	uint8_t	数据包类型	0xca
2	2	uint16_t	请求数据的代码	0x0200... 0x02ff
4	1	uint8_t	错误代码（见注释）	1
5	2	uint16_t	CRC-16（见附录 1）	

以下章节描述了具体的数据写入请求。

注意：如果用户设备无法处理来自 hedgehog 的请求，应返回带有以下错误代码之一的回复：

1 - 请求中"数据包类型"字段未知
2 - 请求中"数据代码"字段未知
3 - 请求中的有效载荷数据不正确
6 - 设备正忙，目前无法获取请求的数据

Marvelmind robotics

2.2.2.1 写入移动路径请求

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：按需支持
Super-Modem：按需支持
Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持
Mini-TX：当前硬件版本不支持
Mini-TX-2：通过 UART 电缆支持
Modem HW4.9：不支持
Beacon HW4.9：支持
Beacon HW4.5：支持

此数据包包含一条基本移动命令。Marvelmind 设备依次发送路径中所有基本移动命令。

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x4a
2	2	uint16_t	数据包中数据代码	0x201
4	1	uint8_t	传输数据的字节数	0x0c
5	12	12 字节	负载数据	
17	2	uint16_t	CRC-16（参见附录 1）	

负载数据格式：

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	基本运动类型： 0 - 向前移动 1 - 向后移动 2 - 向右旋转（顺时针） 3 - 向左旋转（逆时针） 4 - 暂停 5 - 从头开始重复程序 6 - 移动到指定点 7 - 设置速度	
1	1	uint8_t	此基本运动的索引 （0 为第一个）	
2	1	uint8_t	基本运动的总数	
3	2	int16_t	运动参数： 类型 0；1 - 运动距离，cm 类型 2；3 - 旋转角度，度 类型 4：暂停时间，ms 类型 6：X 目标坐标，cm 类型 7：速度，%	
5	2	int16_t	运动参数： 类型 6：Y 目标坐标，cm	

7	2	int16_t	运动参数： 类型 6：Z 目标坐标，cm	
9	3	3 字节	保留（0）	

2.2.2.2 写入区域请求

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：按需支持
Super-Modem：按需支持
Mini-RX（Badge、Helmet 等）：通过 UART 线缆支持
Mini-TX：当前 HW 版本不支持
Mini-TX-2：通过 UART 线缆支持
Modem HW4.9：不支持
Beacon HW4.9：支持
Beacon HW4.5：支持

此数据包包含区域列表序列中的一项。Marvelmind 设备会依次发送区域列表的所有命令。

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	目标地址	0xff
1	1	uint8_t	数据包类型	0x4a
2	2	uint16_t	数据包中的数据代码	0x202
4	1	uint8_t	传输数据的字节数	0x25
5	37	37 字节	有效载荷数据	
42	2	uint16_t	CRC-16（参见附录 1）	

有效载荷数据格式：

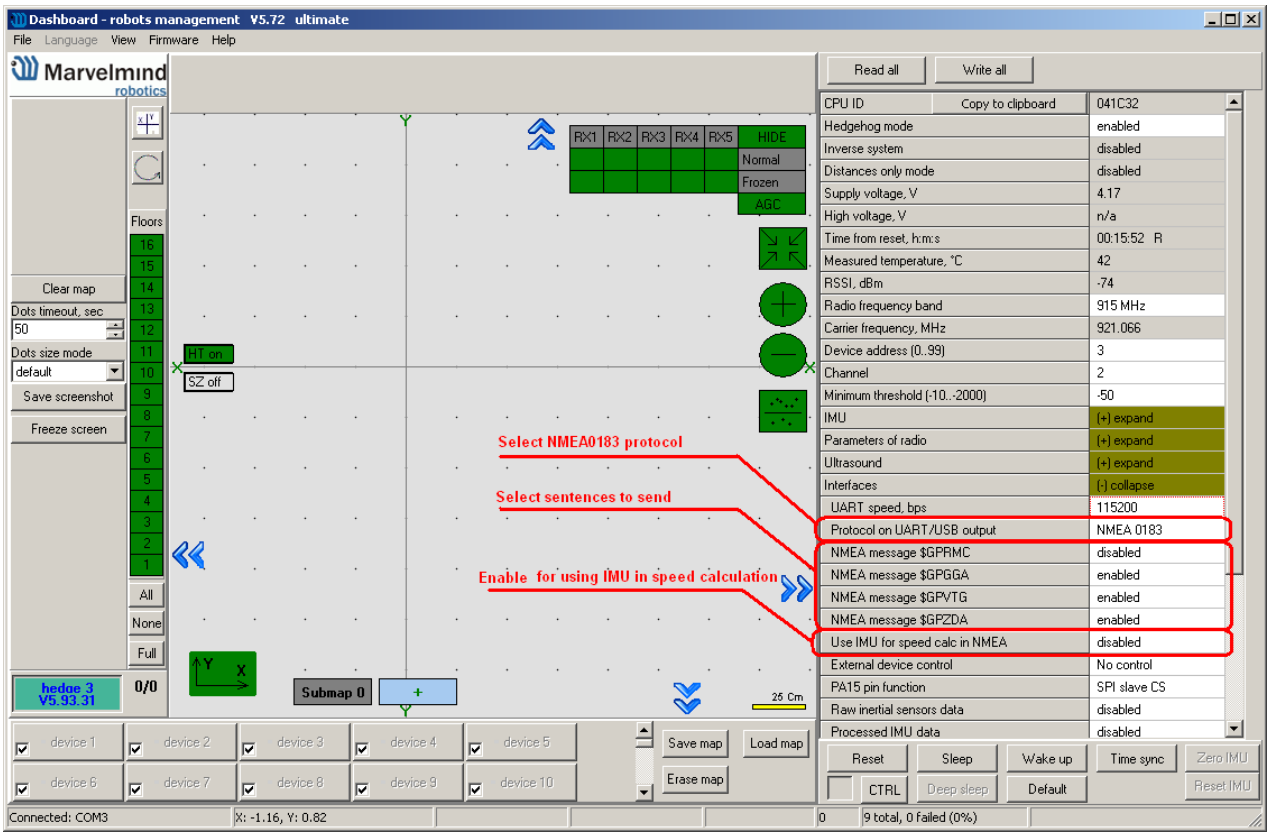
偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	区域索引	
1	1	uint8_t	区域多边形中的点数 (N)	
2	1	uint8_t	本数据包中第一个点的索引：M=0...N-1	
3	1	uint8_t	标志位： 位 0：1 = 非服务区域 位 1：1 = 非行驶区域 位 2：1 = 反向区域 位 3：1 = 激活区域 位 4...7：保留 (0)	
4	1	uint8_t	区域数量	
5	32	4x8 字节	区域多边形最多 4 个点（见下文）	

有效载荷数据格式：

偏移量	大小（字节）	类型	描述	数值
0	4	int32_t	点的 X 坐标，单位 mm	
4	4	int32_t	点的 Y 坐标，单位 mm	

2.3 NMEA0183 通信协议

移动信标可通过 UART 和 USB（虚拟 UART）接口输出部分 NMEA0183 语句。需要在设备中通过 Dashboard 启用 NMEA 协议，如下图所示：



每次接收到更新的位置时，设备都会发送所有已启用的消息。

要从移动信标（Hedgehog）获取 NMEA 数据，需通过以下任一接口将其连接到外部设备（机器人、飞行器、AGV 等）：

1. 作为 CDC 类 USB 设备连接到 USB 主机（在 Windows 中为虚拟 COM 端口，在 Linux 中为 ttyUSB 或 ttyACM）。在 Windows 中需要驱动程序——与 Modem 所用驱动相同。在 Linux 中大多数情况下不需要驱动程序，因为所需驱动已集成到 Linux 内核中。由于该接口并未使用真正的 RS-232，因此主机上打开串口的参数（波特率、位数、校验位等）可以任意设置。
2. 连接到 Hedgehog 上的 UART——需要焊接 2 根线到引脚。请参见下图的信标接口。要输出位置数据，只需连接 2 根线即可：GND 和 USART2_TX。UART 发送端的逻辑电平为 CMOS 3.3V。默认波特率为 500 kbps，可从 Dashboard 配置（参见上图中的参数 "UART speed, bps"），可选值包括：4.8、9.6、19.2、38.4、57.6、115.2、500 kbps。数据格式：8 位、无校验位、1 位停止位。

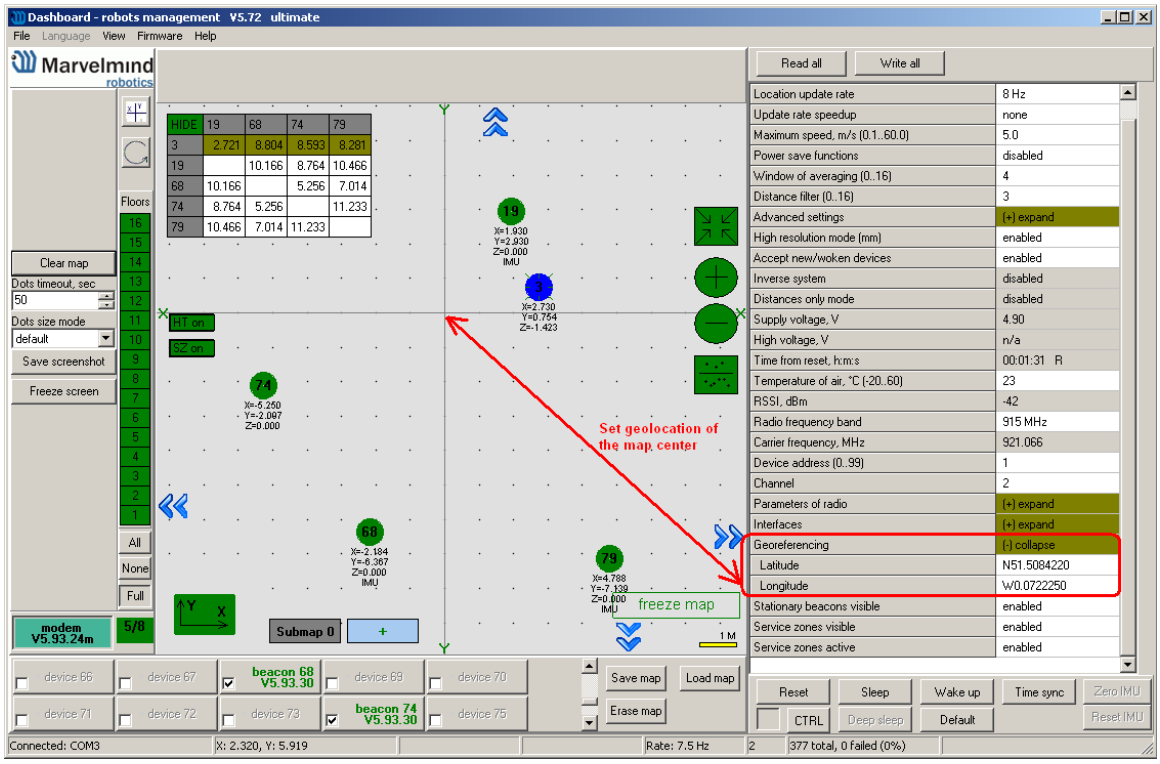
13.11.

2.3.1 坐标转换的通用约定

Marvelmind 系统以直角笛卡尔坐标系（X、Y、Z）的形式测量位置，其中 Z 在大多数情况下表示高度。转换为 GPS 坐标时采用以下约定：

- Z 轴方向向上，Z 坐标表示海拔高度；
- Y 轴指向北方，因此 Y 表示纬度；
- X 轴指向东方,因此 X 表示经度;
- 点 (X= 0, Y= 0) 的 GPS 坐标按照地理参考点确定(默认为:北纬 0°,西经 0°);

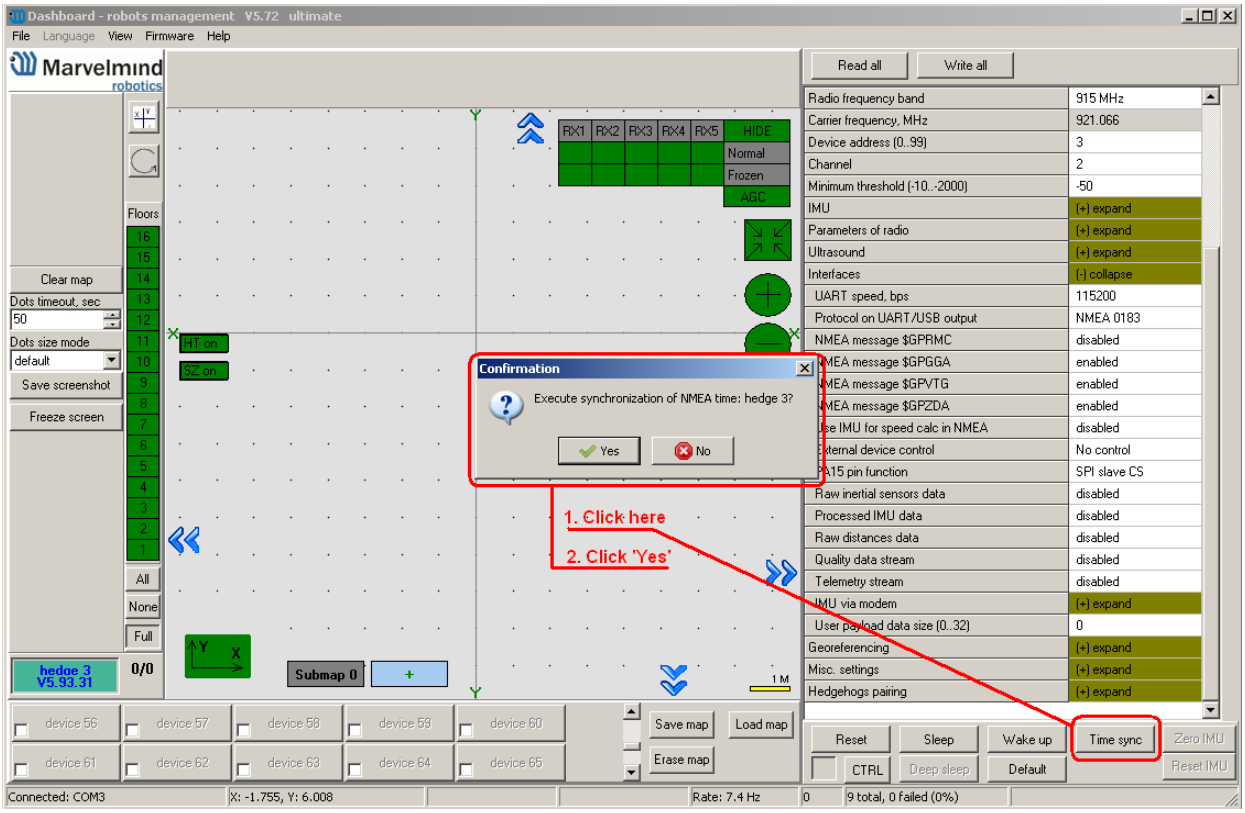
地理参考坐标可按照屏幕截图所示方式设置:



GPS 坐标根据指定的地理参考点和 WGS-84 地球模型计算。更详细地,Lat= Lat_ref + y*9.013373 其中 Lat - 纬度,微度 Lat_ref - 地理参考纬度,微度 y - Marvelmind 系统中的 y 坐标,米 Long= Long_ref + x*8.98315/cos(Lat_ref/1000000)Long - 经度,微度 Long_ref - 地理参考经度,微度 Lat_ref - 地理参考纬度,微度 x - Marvelmind 系统中的 x 坐标,米

2.3.2 时间的一般约定

上电后,移动信标从 2016.08.01 00:00:00 开始计时。用户可按以下屏幕截图所示与计算机时钟同步时间。



2.3.3 "NMEA0183" 消息实现说明

NMEA 0183 消息为 ASCII 编码文本帧,由若干部分组成,各部分以逗号分隔,并以行结束符终止。
在行结束符之前,每条消息以 '*' 符号结尾,后跟两个按 NMEA 0183 标准计算的校验和字符。
NMEA 0183 消息的每个部分代表某个特定参数。
以下是所有支持的消息及参数字段的说明。
消息格式采用 NMEA 0183 标准 3.01 版本,2002 年 1 月 1 日发布。

1. $GPRMC - 推荐的最小专用 GNSS 数据

支持的硬件:

Super-Beacon:支持

Industrial Super-Beacon:支持

Modem HW5.1:支持(从 SW V7.000 开始)

Super-Modem:支持(从 SW V7.000 开始)

Mini-RX(Badge、Helmet 等):通过 UART 电缆支持

Mini-TX:当前 HW 版本不支持

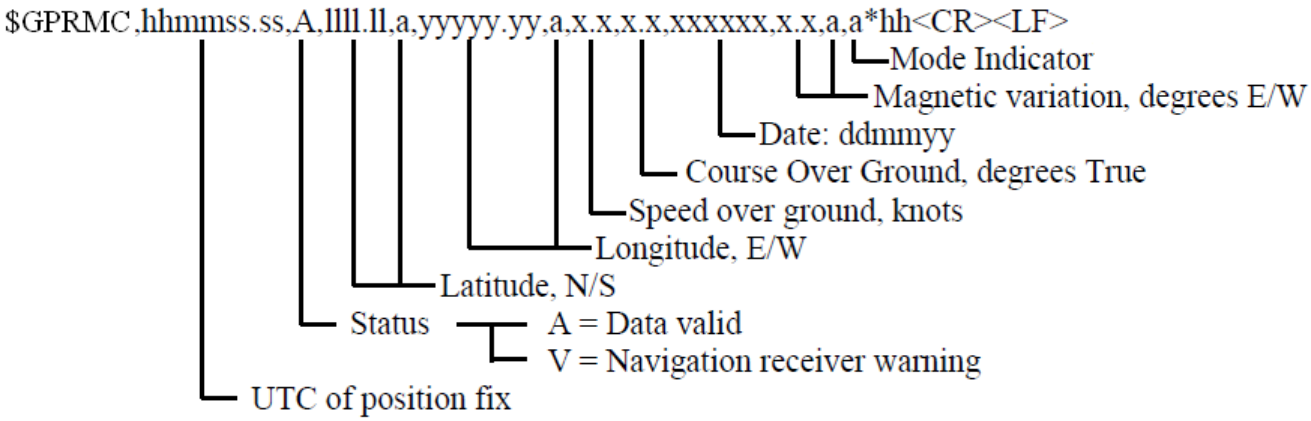
Mini-TX-2:通过 UART 电缆支持

Modem HW4.9：不支持

Beacon HW4.9：支持

Beacon HW4.5：支持

来自 NMEA 0183 标准的通用格式:



字段实现说明:

1.1. '$GPRMC' – 消息类型标识

1.2. 'hhmmss.ss' – UTC 定位时间

根据通用约定，时间从默认的 2016.01.01 开始计数，或与计算机时钟同步。

1.3. 'A' – 状态

如果最后一次位置更新成功，则发送 'A' 值
如果最后一次位置更新中发生任何错误，则发送 'V' 值

1.4. 'llll.llllll,a' – 纬度，N/S

根据通用约定（见上文），纬度对应于相对于地理参考位置的 Y 坐标。纬度以分的 6 位十进制小数形式表示，分辨率不低于 2 mm，

1.5. 'yyyyy.yyyyyy,a' – 经度，E/W

根据通用约定（见上文），经度对应于相对于地理参考位置的 X 坐标。经度以分的 6 位十进制小数形式表示，分辨率不低于 2 mm。

1.6. 'x.x' – 对地速度，节

Marvelmind robotics

Marvelmind 系统测量坐标；速度根据坐标的动态变化并应用一定的滤波计算得出。可选地，可使用 IMU 融合进行速度计算。

1.7. 'xxxxxx' - 日期：ddmmyy

根据通用约定，时间从默认的 2016.01.01 开始计数，或与计算机时钟同步。

1.8. 'x.x,a' - 磁偏角

此参数值始终为空字段。

1.9. 'a' - 模式指示符

如果上次位置更新成功，则发送 'A' 值（自主模式）
如果上次位置更新中发生任何错误，则发送 'N' 值（数据无效）

2. $GPGGA - 全球定位系统定位数据

支持的硬件：

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持（从 SW V7.000 起）

Super-Modem：支持（从 SW V7.000 起）

Mini-RX（Badge、Helmet 等）：使用 Mini-Rx 的 UART 电缆支持

Mini-TX：当前 HW 版本不支持

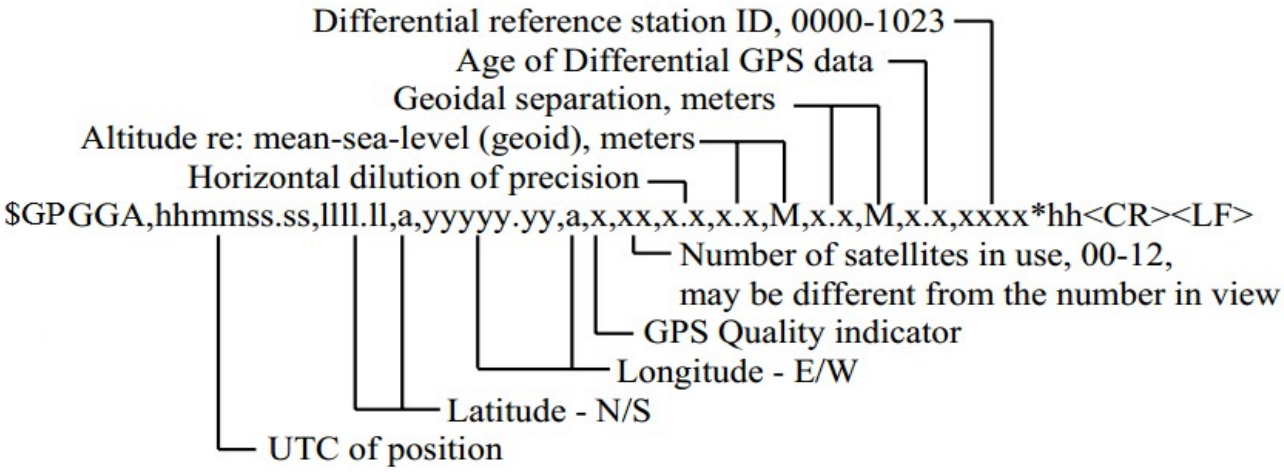
Mini-TX-2：支持

Modem HW4.9：不支持

Beacon HW4.9：支持

Beacon HW4.5：支持

NMEA 0183 标准的通用格式：



字段实现说明：

2.1. '$GPGGA' – 消息类型标识

2.2. 'hhmmss.ss' – UTC 定位时间

根据通用约定,时间从默认的 2016.01.01 开始计算,或与计算机时钟同步。

2.3. 'llll.llllll,a' – 纬度,N/S

根据通用约定(见上文),纬度对应于相对地理参考位置的 Y 坐标。纬度以分的小数部分的 6 位数字表示,分辨率不低于 2 mm。

2.4. 'yyyyy.yyyyyy,a' – 经度,E/W

根据通用约定(见上文),经度对应于相对地理参考位置的 X 坐标。经度以分的小数部分的 6 位数字表示,分辨率不低于 2 mm。

2.5. 'x' – GPS 质量指示器
如果上一次位置更新成功,则发送 '1'(GPS SPS 模式,定位有效)值

如果上一次位置更新发生任何错误,则发送 '0'(定位不可用或无效)值

2.6. 'xx' – 使用的卫星数量
当前实现中始终为 '08'。

2.7. 'x.x' – 水平精度因子
当前实现中始终为 '1.2'。

2.8. 'x.x, M' – 相对于平均海平面(大地水准面)的海拔,单位为米
根据通用约定,此值对应于 Z 坐标。

2.9. 'x.x, M' – 大地水准面分离,单位为米
始终传输 '0.0, M' 值。

2.10. 'x.x' – 差分 GPS 数据的年龄
此参数值始终为空字段,不使用 DGPS。

2.11. 'xxxx' – 差分参考站 ID
此参数值始终为空字段。

3. $GPVTG - 对地航向与对地速度

支持的硬件:

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持（自 SW V7.000 起）

Super-Modem：支持（自 SW V7.000 起）

Mini-RX（Badge、Helmet 等）：通过 Mini-Rx 的 UART 线缆支持

Mini-TX：当前 HW 版本不支持

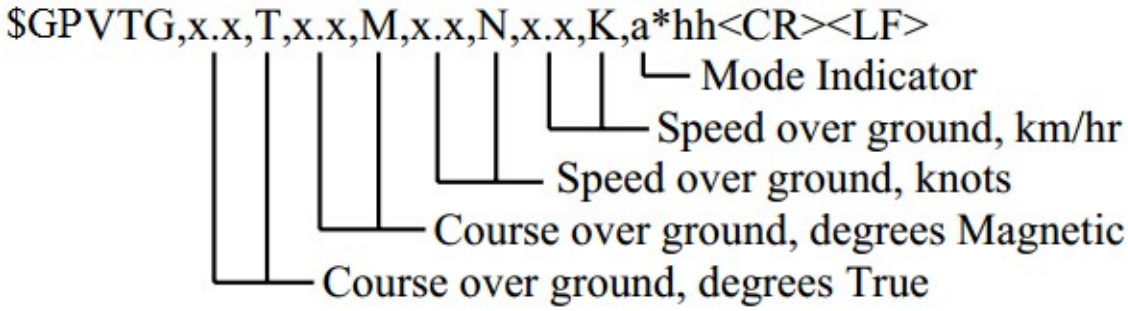
Mini-TX-2：支持

Modem HW4.9：不支持

Beacon HW4.9：支持

Beacon HW4.5：支持

NMEA 0183 标准的通用格式:



字段实现说明:

3.1. '$GPVTG' — 消息类型标识

3.2. 'x.x, T' — 对地航向，真北方向度数
根据 NMEA 标准，航向是速度矢量与指向北方向之间的夹角。如上文通用约定所示，Y 轴被视为指向北的方向。

3.3. 'x.x, M' — 对地航向，磁北方向度数
在当前实现中，磁航向与真航向相同。

3.4. 'x.x, N' — 对地速度，节

Marvelmind 系统测量坐标;速度通过坐标的动态变化并应用一定的滤波计算得出。可选地,它可以使用 IMU 融合进行速度计算。

3.5. 'x.x, K' – 对地速度,km/hr

这是以另一种单位表示的相同速度

3.6. 'a' – 模式指示符

如果最近一次位置更新成功,则发送 'A' 值(自主模式)
如果最近一次位置更新出现任何错误,则发送 'N' 值(数据无效)

4. $GPZDA – 时间和日期

支持的硬件:

Super-Beacon:支持

Industrial Super-Beacon:支持

Modem HW5.1:支持(从 SW V7.000 开始)

Super-Modem:支持(从 SW V7.000 开始)

Mini-RX(Badge、Helmet 等):通过 UART 电缆支持

Mini-TX:当前 HW 版本不支持

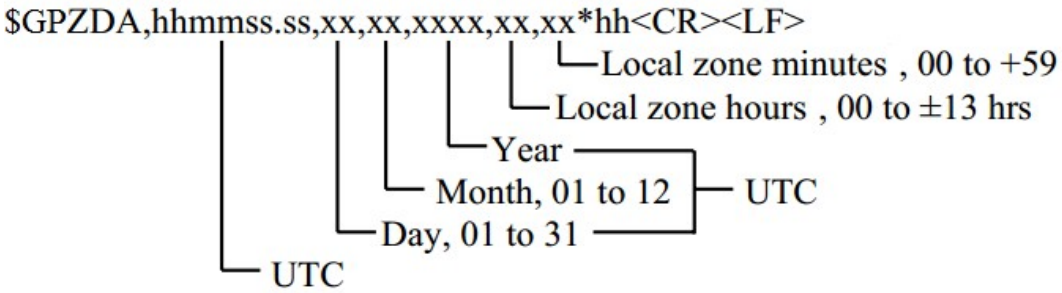
Mini-TX-2:通过 UART 电缆支持

Modem HW4.9:不支持

Beacon HW4.9:支持

Beacon HW4.5:支持

NMEA 0183 标准的通用格式:



根据一般约定,时间从默认的 2016.01.01 开始计数,或与计算机时钟同步。

字段实现说明:

4.1. '$GPZDA' – 消息类型标识

4.1. 'hhmmss.ss' – UTC

时间（时、分、秒）。

4.2. 'xx' – 日，01 至 31

日。

4.3. 'xx' – 月，01 至 12

月。

4.4. 'xxxx' – 年

年。

4.4. 'xx – 本地时区小时数

本地时区始终为 "00" 小时。

4.5. 'xx – 本地时区分钟数

本地时区始终为 "00" 分钟。

5. $GPHDT – 航向

支持的硬件:

Super-Beacon：支持

Industrial Super-Beacon：支持

Modem HW5.1：支持（从 SW V7.000 开始）

Super-Modem：支持（从 SW V7.000 开始）

Mini-RX(Badge、Helmet 等):通过 UART 电缆支持

Mini-TX:不支持

Mini-TX-2:通过 UART 电缆支持

Modem HW4.9:不支持

Beacon HW4.9:不支持

Beacon HW4.5:不支持

注意:启用此数据包的流式传输需要 MMSW0002 许可证。

NMEA 0183 标准的通用格式:

```
$GPHDT,x.x,T*hh<CR><LF>
       └┴─Heading, degrees True
```

5.1. '$GPHDT' – 消息类型标识

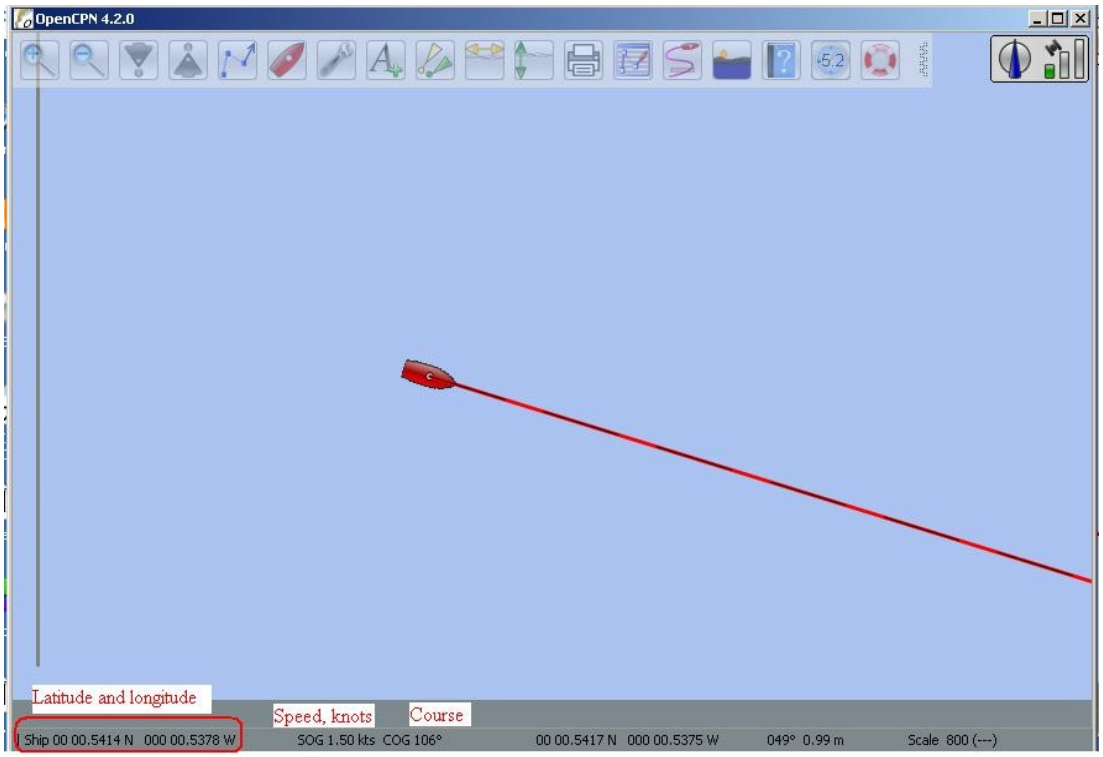
5.2. 'x.x, T' – 航向角,真北方向的度数

这是通过使用配对信标或配对麦克风功能并与陀螺仪融合计算得出的航向角。

Marvelmind robotics

2.3.4 NMEA 数据接收示例

下一个截图是从通过 USB(虚拟 COM 端口)连接到运行在 MS Windows 计算机上的 OpenCPN 软件的移动信标接收到的数据示例。



3. 通过 USB(虚拟 UART)的通信协议

3.1 用于流式传输的 "Marvelmind" 协议

UART 相应章节中描述的所有数据包也可通过 USB(虚拟 UART)获取。

注意:这些数据也可用于 mini-TX 和 mini-RX,无需 "UART Cable for Mini-Rx"。

如果 Marvelmind 设备根据此协议接收到任何请求,流式传输将终止 5 秒。

3.2 从用户设备读取/写入数据的协议

UART 相应章节中描述的所有数据包也可通过 USB(虚拟 UART)获取。

请注意，这些数据也适用于 mini-TX 以及不带"UART Cable for Mini-Rx"的 mini-RX。

3.3 NMEA0183 通信协议

UART 对应章节中描述的所有数据包也可通过 USB（虚拟 UART）获取。

请注意，这些数据也适用于 mini-TX 以及不带"UART Cable for Mini-Rx"的 mini-RX。

如果 Marvelmind 设备收到任何符合此协议的请求，数据流将中断 5 秒。

3.4 通过 USB 接口与 Modem 进行数据交换的协议

该协议由 Dashboard 软件和下一章所述的 Marvelmind API 使用。

Modem 作为 CDC 类 USB 设备连接到 USB 主机（在 Windows 中为虚拟 COM 端口，在 Linux 中为 ttyUSB 或 ttyACM）。

由于该接口未使用真实的 RS-232，因此主机上设置的串口参数（波特率、位数、奇偶校验等）可以是任意值

数据为二进制格式

通过 USB 连接的设备的"网络地址"为 0xff

多字节数值从低字节开始传输（小端格式）

3.4.1 读取最新坐标数据包（固件 V5.13+）

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x4110
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（见附录 1）	0xc004

应答帧格式（从 modem 到主机）

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	modem 的地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	传输数据的字节数	0x64
3	100 (0x64)	100 字节	数据结构（见下文）	
103	2	uint16_t	CRC-16（见附录 1）	

错误响应的格式见附录 2。

数据字段格式（100 字节）

偏移量	大小（字节）	说明
0	96 (6*16)	Modem 接收到的最近六个坐标结构（见下文）
96	1	标志字节： 位 0...1：保留 位 2：1 = 用户数据可用（见第 12 节） 位 3...7：保留
97	3	保留

坐标结构格式（16 字节）

偏移量	大小（字节）	描述
0	1	设备地址
1	4	坐标 X，mm (int32_t)
5	4	坐标 Y，mm (int32_t)

Marvelmind robotics

9	4	坐标 Z，mm (int32_t)
13	1	标志字节： Bit 0：1 – 无有效坐标（Dashboard 中红色模式） Bit 1：1 – 冻结地图上的临时移动信标（蓝色模式） Bit 2：1 – 该信标用于 Hedgehog 定位
14	2	保留 (0)

3.4.2 读取/写入 Modem 配置

3.4.2.1 读取 Modem 配置（固件 V5.30+）

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x5000
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（参见附录 1）	0x0550

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	数据传输字节数	0x30
3	0x30	结构体	数据结构（见下文）	
0x33	2	uint16_t	CRC-16（见附录 1）	

错误回复的格式在附录 2 中进行了描述。

Marvelmind robotics

3.4.2.2 写入 Modem 配置

警告！要写入 Modem 配置，必须先读取配置，设置以下章节中描述的数据字段，然后再写入。请勿更改结构体中的任何其他字节，否则可能会降低 Modem 的工作性能。

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 的地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0x5000
4	2	uint16_t	访问模式	0x0000
6	1	uint8_t	数据传输字节数	0x30
7	0x30	结构体	数据结构（见下文）	
0x37	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 的地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据代码	0x5000
4	2	uint16_t	保留	
6	2	uint16_t	CRC-16（参见附录 1）	

错误回复的格式在附录 2 中描述。

Marvelmind robotics

3.4.2.3 Modem 配置数据结构

数据结构的许多字段未作说明。请勿更改这些字段！它们用于通过 Dashboard 程序调整系统；未经授权的更改可能会影响 modem 的正常工作

偏移量	大小（字节）	类型	描述
0	20	20 字节	未说明
20	1	int8_t	空气温度设置 Vt（有符号）。 Temperature is (Vt+23) °C
21	1	uint8_t	地图坐标为 X=0, Y=0 的信标地址
22	4	4 字节	未说明
26	1	uint8_t	地图坐标为 X>0, Y=0 的信标地址
27	1	uint8_t	地图坐标中 Y>0 的信标地址
28	1	uint8_t	控制标志： 位 0：未说明 位 1：1 - 启用移动信标运动滤波 位 2：未说明 位 3：1 - 高分辨率模式（输出坐标以 mm 代替 cm） 位 4：未说明 Bit 5: 1 = 镜像所有地图 Bit 6: 1 = 省电模式（仅当所有子地图均被冻结时，省电模式才生效） Bit 7: 未说明
29	2	2 字节	未说明
31	1	uint8_t	N，决定获取 hedgehog 坐标的最大频率 F(N)= 2^(N-1) Hz，N= 0…4， F(5)= 12 Hz，F(6)= 16 Hz，F(7)= 16+（最大值）
32	16	16 字节	未说明

Marvelmind robotics

3.4.3 读取/写入子地图配置

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

3.4.3.1 读取子地图配置（固件 V5.30+）

请求帧格式（从主机到 modem）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x60XX，其中 XX 为子地图编号
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	数据传输字节数	0x50 (80)
3	80	结构体	数据结构（见下文）	
83	2	uint16_t	CRC-16（见附录 1）	

错误回复的格式在附录 2 中描述。

Marvelmind robotics

3.4.3.2 写入子地图配置（固件 V5.30+）

警告！要写入子地图配置，您必须先读取配置，设置以下章节中描述的数据字段，然后再写入。请勿更改结构中的任何其他字节，否则可能会降低 modem 的工作性能

请求帧格式（从主机到 modem）

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中数据的代码	0x60XX,其中 XX 为子地图编号
4	2	uint16_t	访问模式	0x0000
6	1	uint8_t	数据传输字节数	0x50 (80)
7	80	structure	数据结构(见下文)	
87	2	uint16_t	CRC-16(见附录 1)	

应答帧格式(从 Modem 到主机)

偏移量	大小(字节)	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据代码	0x5000
4	2	uint16_t	保留	
6	2	uint16_t	CRC-16（见附录 1）	

错误回复格式在附录 2 中描述。

Marvelmind robotics

3.4.3.3 子地图配置数据结构

数据结构的许多字段未作说明。请勿更改这些字段！它们用于通过 Dashboard 程序调整系统；未经授权的更改可能导致 modem 工作性能下降

偏移量	大小（字节）	类型	说明
0	1	uint8_t	用于构建子地图的起始信标地址
1	1	uint8_t	控制字: Bit 0:1 — 子地图被冻结（冻结子地图） Bit 1:1 — 信标高于 Hedgehog Bit 2...4：未说明 Bit 5:1 — 镜像子地图 Bit 6...7：未说明
2	1	uint8_t	距离限制： Bit 0...6：手动限制距离（如果 bit 7 = 1） Bit 7：0 — 自动限制，1 = 手动
3	13	13 字节	未说明
16	2	int16_t	子地图的 X 偏移，cm
18	2	int16_t	子地图的 Y 偏移，cm
20	2	uint16_t	子地图的旋转角度，百分之一度
22	58	58 字节	未说明

Marvelmind robotics

3.4.4 设备休眠/唤醒

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机发送至 modem）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0xb006
4	2	uint16_t	访问模式	唤醒时：0x0002 其他：0x0001
6	1	uint8_t	数据传输字节数	0x08
7	1	uint8_t	密码，字节 0	0x2d
8	1	uint8_t	密码，字节 1	0x94
9	1	uint8_t	密码，字节 2	0x5e
10	1	uint8_t	密码，字节 3	0x81
11	1	uint8_t	命令： 0 – 标准睡眠 1 – 深度睡眠（仅通过 HW 复位唤醒） 2 – 从标准睡眠唤醒 3…255 - 保留	0…2
12	3	3 字节	保留	
15	2	uint16_t	CRC-16（参见附录 1）	

唤醒命令的应答帧格式（从 Modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据代码	0xb006
4	2	uint16_t	保留	
6	2	uint16_t	CRC-16（见附录）	

休眠命令应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型（modem 回复）	0x7f
2	2	uint16_t	数据代码	0xb006
4	2	uint16_t	保留	
6	2	uint16_t	字节 0…5 的 CRC-16（参见附录 1）	
8	1	uint8_t	设备地址	0x01…0xfe
9	1	uint8_t	数据包类型	0x10
10	2	uint16_t	数据代码	0xb006
12	2	uint16_t	保留	
14	2	uint16_t	字节 8…13 的 CRC-16（见附录）	

错误回复的格式在附录 2 中描述。

3.4.5 设置设备地址

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 modem）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中数据代码	0x0101
4	2	uint16_t	访问模式	0x0000
6	1	uint8_t	数据传输字节数	0x02
7	1	uint8_t	数据项代码 (地址)	0x00
8	1	uint8_t	设备的新地址	
9	2	uint16_t	CRC-16 (参见附录 1)	

3.4.6 读取测量的原始距离

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX (Badge、Helmet 等)：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

此命令可通过两种模式访问：

- 数据代码为 0x4000 — 读取最近八个距离。应答帧包含自请求时刻起最近测量的 8 个距离
- 数据代码为 0x4001 — 逐帧读取所有距离。每次后续请求的应答帧包含接下来已保存的 8 个测量距离。当所有距离表传输完毕后，将从头开始

请求帧格式（从主机到 Modem）

偏移	大小（字节）	类型	说明	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x4000 或 0x4001
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	传输数据字节数	0x28
3	40 (0x28)	40 字节	数据结构（见下文）	
43	2	uint16_t	CRC-16（见附录 1）	

错误回复的格式见附录 2。

数据字段格式（40 字节）

偏移量	大小（字节）	描述
0	32 (8*4)	八个原始距离结构（见下文）
32	8	保留

距离结构格式（4 字节）

偏移量	大小（字节）	描述
0	1	超声波接收器地址
1	1	超声波发射器地址

2	2	设备之间的测量距离，mm (uint16_t)

3.4.7 读取信标状态（固件 V5.33+）

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中数据的代码	0x0003
4	2	uint16_t	访问模式	0x0002
6	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 Modem 到主机）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	设备地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	数据传输的字节数	0x20
3	32	32 字节	数据结构（见下文）	
35	2	uint16_t	CRC-16（参见附录 1）	

错误回复格式在附录 2 中描述。

数据字段格式：

偏移量	大小（字节）	类型	描述
0	4	uint32_t	自复位或唤醒以来的工作时间（秒）
4	1	uint8_t	R，无线电 RSSI 寄存器值（接收信号强度指示）。 若 R>128，则 RSSI (dBm) = ((R-256)/2) – 74 若 R<=128，则 RSSI (dBm) = (R/2) - 74
5	1	uint8_t	未说明
6	1	int8_t	Measured temperature Vt (signed). Temperature is (Vt+23) °C
7	2	uint16_t	Bit 0…11：电源电压，mV Bit 12…13：未说明 Bit 14：1：低电量，设备将在短时间后进入睡眠模式 Bit 15：1：电量极低，设备将在短时间后进入深度睡眠模式
9	23	23 字节	未说明

3.4.8 Marvelmind 机器人控制命令

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

3.4.8.1 机器人控制命令

请求帧格式（从主机到 modem）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	机器人地址	0x01...0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0x1000
4	2	uint16_t	访问模式	0x0001
6	1	uint8_t	数据传输字节数	0x10
7	16 (0x10) 字节	uint8_t	机器人控制数据（见下文）	
23	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（由 modem 发送至主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型（modem 回复）	0x7f
2	2	uint16_t	数据代码	0x1000
4	2	uint16_t	保留	
6	2	uint16_t	字节 0...5 的 CRC-16（见附录 1）	
8	1	uint8_t	机器人地址	0x01...0xfe
9	1	uint8_t	数据包类型（机器人回复）	0x10
10	2	uint16_t	数据代码	0x1000
12	2	uint16_t	保留	
14	2	uint16_t	字节 8...13 的 CRC-16 校验(见附录 1)	

错误响应的格式详见附录 2。

机器人控制数据的格式:

偏移量	大 小 (字节)	类型	描述
0	1	uint8_t	控制模式: 0 - 无控制(等待模式) 1 - 电机动力控制 2 - 速度控制 3 - 写入运动程序

Marvelmind robotics

			4 - 暂停运动程序 5 - 暂停后继续运动
1	1	uint8_t	操作代码: 0 - 向前移动 1 - 向后移动 2 - 顺时针旋转 3 - 逆时针旋转 4 - 暂停指定时间（用于模式 3） 5 - 从开始处重复运动程序（用于模式 3） 6 - 按坐标移动到指定点（用于模式 3） 7 - 设置运动速度（用于模式 3）
2	1	uint8_t	控制字节 1： 对于模式 1：电机开启功率，% 对于模式 2：运动速度，% 对于模式 3：程序步骤编号（从零开始）
3	2	Int16_t	程序数据（模式 3）： 操作码 0 或 1：运动距离，cm 操作码 2 或 3：旋转角度，度 操作码 4：暂停时间，ms 操作码 6：运动目标的 X 坐标，cm 操作码 7：运动速度，%
5	1	uint8_t	对于模式 3：程序中的总步数。
6	2	int16_t	程序附加数据（模式 3）： 操作代码 6：移动目标的 Y 坐标，单位 cm
8	2	int16_t	操作代码 6：移动目标的 Z 坐标，单位 cm
10	6	6 字节	保留（0）

关于此复杂命令的一些说明。
机器人控制结构的字节 0 中指定了三种主要的机器人控制模式：

- 功率控制（模式 1）
- 速度控制（模式 2）
- 按程序移动（模式 3）

模式 1 和模式 2 通常用于测试目的。在模式 1 下，机器人以所选的电机功率向前、向后移动，或向左、向右旋转。在模式 2 下，机器人执行相同动作，但会调节功率以保持所选的速度。功率或速度在结构的字节 2 中设置，移动类型在字节 1 中设置。

模式 4 和模式 5 是特殊命令，用于在程序执行过程中暂停移动以及暂停后继续移动。

沿复杂轨迹移动的主要模式是模式 3。
它允许向机器人编程一系列基本动作，这些动作的组合构成运动轨迹。该序列的每一项都应通过此类型的一个命令发送。每个命令应在机器人控制结构的字节 2 中包含当前步骤编号，在字节 5 中包含总步数。

在机器人控制结构的字节 1 中，指定了基本移动的类型。基本移动的参数在"程序数据"字段（字节 3...4）和"程序附加数据"字段（字节 6...7）中指定。

因此，可用的基本动作如下：

- 按给定距离向前移动;
- 按给定距离向后移动;
- 按给定角度顺时针旋转;
- 按给定角度逆时针旋转;
- 暂停指定时间;
- 从第一项重新启动运动程序（用于循环运动）；
- 在 Marvelmind 导航系统坐标中移动到指定点 (X, Y);

- 更改移动速度。

机器人在接收到基元序列后开始执行程序。程序执行完毕后，机器人停止。但如果程序包含操作码为 5 的项（从头重复），则程序会循环执行，直到接收到停止命令或上传新程序为止。

3.4.8.2 停止机器人

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	机器人地址	0x01…0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0x403
4	2	uint16_t	访问模式	0x0001
6	1	uint8_t	数据传输字节数	0x04
7	4 字节	4 字节	保留 (0)	0
11	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型（modem 回复）	0x7f
2	2	uint16_t	数据代码	0x403
4	2	uint16_t	保留	
6	2	uint16_t	字节 0…5 的 CRC-16 校验（参见附录 1）	
8	1	uint8_t	机器人地址	0x01…0xfe
9	1	uint8_t	数据包类型（机器人回复）	0x10
10	2	uint16_t	数据代码	0x403
12	2	uint16_t	保留	
14	2	uint16_t	字节 8…13 的 CRC-16(见附录 1)	

错误回复的格式在附录 2 中描述。

此命令仅用于终止任何机器人运动或运动程序的执行。机器人停止并等待新的命令。

Marvelmind robotics

3.4.9 读取/写入设备控制设置（固件 V6.01+）

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9:不支持
Beacon HW4.5:不支持

3.4.9.1 读取设备控制设置

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	设备地址（beacon/modem）	0x01...0xfe 或 0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x1201
4	2	uint16_t	访问模式	0x0001
6	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x03 表示请求已发送至 modem 0x7f 表示请求已发送至 beacon
2	1	uint8_t	数据传输字节数	0x10
3	16	结构体	数据结构（参见第 9.3 节）。 仅当请求发送至 modem (0xff) 时相关	
11	2	uint16_t	字节 0...10 的 CRC-16（参见附录 1）	
如果请求发送至 beacon，将接收以下数据				
13	1	uint8_t	设备地址	0x01...0xfe
14	1	uint8_t	数据包类型	0x03
15	1	uint8_t	数据传输字节数	0x08/0x10
16	16	结构体	数据结构（见下文）。	
32	2	uint16_t	字节 12...22 的 CRC-16（参见附录 1）	

错误回复的格式在附录 2 中描述。

Marvelmind robotics

3.4.9.2 写入设备控制设置

请求帧格式（从主机到 modem）

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	设备地址（beacon/modem）	0x01…0xfe 或 0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中数据的代码	0x1201
4	2	uint16_t	访问模式	0x0001
6	1	uint8_t	数据传输的字节数	0x10
7	16	结构体	数据结构（见下文）	
23	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x03 如果请求发送至 modem 0x7f 如果请求发送至 beacon
2	2	uint16_t	数据代码	0x1201
4	2	uint16_t	保留	
6	2	uint16_t	字节 0…5 的 CRC-16 校验值（参见附录 1）	
若请求发送至信标（地址为 0x01…0xfe），将接收到以下数据				
8	1	uint8_t	设备地址	0x01…0xfe
9	1	uint8_t	数据包类型	0x10
10	2	uint16_t	数据代码	0x1201
12	2	uint16_t	保留	
14	2	uint16_t	字节 8…13 的 CRC-16 校验值（参见附录 1）	

错误回复的格式在附录 2 中描述。

Marvelmind robotics

3.4.9.3 控制设置有效负载数据的格式

偏移量	大小（字节）	类型	描述
0	1	uint8_t	标志位： 位 0...5：未作说明，应始终为零！ 位 6：0 - 固定信标模式，1 - Hedgehog 模式 位 7：保留（0）
1	1	uint8_t	UART 波特率设置： 0：500000 bps（默认值） 1：4800 bps 2：9600 bps 3：19200 bps 4：38400 bps 5：57600 bps 6：115200 bps 7...255：保留
2	1	uint8_t	保留 (0)
3	1	uint8_t	位 0...3：无线电配置文件： 0: 38.4 kbps 1: 150 kbps 2: 500 kbps 3...7: 保留 位 4...6：无线电频段： 0: 433 MHz 1: 868 MHz 2: 915 MHz 3: 315 MHz 4...7: 保留 位 7：保留
4	1	uint8_t	UART/USB 输出类型： 0: Marvelmind 协议 1: NMEA0183
5	1	uint8_t	NMEA0183 模式下发送的 NMEA 帧掩码： Bit 0：1 - 发送 $GPRMC 帧 Bit 1：1 - 发送 $GPGGA 帧 Bit 2：1 - 发送 $GPVTG 帧 Bit 3：1 - 发送 $GPZDA 帧 Bit 4...7：保留（0）
6	1	uint8_t	从此 Hedgehog 发送到 Modem 的用户负载数据字节数（0...32）
7	1	uint8_t	"IMU via modem" 模式下发送到 Modem 的 IMU 数据掩码： Bit 0：IMU 融合位置 Bit 1：四元数 Bit 2：速度 Bit 3：加速度 Bit 4：原始加速度计数据 Bit 5：原始陀螺仪数据 Bit 6：原始罗盘数据 Bit 7：0 = 发送 IMU 融合数据，1 = 发送原始 IMU 数据

Marvelmind robotics

8	1	uint8_t	Bit 0...6：遥测数据流间隔（0 = 无数据流） 位 7：保留（0）
9	1	uint8_t	位 0：使用 IMU 进行速度计算 位 1...7 – 保留（0）
10	6	6 字节	保留（0）

警告！如果您在通过无线电连接的信标上更改无线电配置文件，无线电连接将会丢失。如果需要切换配置文件，请先逐个切换所有信标的无线电配置文件，然后再切换 Modem 的无线电配置文件。几秒钟后，所有信标应在新的无线电配置文件下可用。

3.4.10 读取网络中的设备列表（固件 V6.01+）

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x31xx，其中 xx 为设备组编号
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 Modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 的地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	传输数据的字节数	0x72
3	1	uint8_t	网络中的设备总数 (K)	
4	112	112 字节	0...16 个网络中设备信息的结构体，详见下方描述	
116	1	uint8_t	保留	0x00
117	2	uint16_t	CRC-16（参见附录 1）	

错误回复格式详见附录 2。

网络中设备数据的格式（7 字节）

偏移量	大小（字节）	说明

Marvelmind robotics

0	1	设备地址（0x01...0xfe）
1	1	固件主版本号
2	1	固件次版本号
3	1	位 0...5：设备类型： 10：轮式机器人 12：履带式机器人 22：Beacon HW V4.5 23： Beacon HW V4.5 （hedgehog 模式） 24：Modem（HW V4.5/4.9） 30：Beacon HW V4.9 31： Beacon HW V4.9 （hedgehog 模式） 32：Mini-RX beacon 36：Mini TX beacon（HW V5.07） 37：Industrial-TX beacon 41：Industrial-RX beacon 42：Super-Beacon 43：Super-Beacon（Hedgehog 模式） 44：Industrial Super-Beacon 45：Industrial Super-Beacon （Hedgehog 模式） Bit 6：1 - 存在多个使用该地址的设备 Bit 7：1 - 休眠模式
4	1	固件的第二次要版本号

3.4.11 固件的读取版本号

支持的硬件:

Super-Beacon：支持
Industrial Super-Beacon：支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：支持
Mini-TX：支持
Mini-TX-2：支持
Modem HW4.9：支持
Beacon HW4.9:支持
Beacon HW4.5:支持

请求帧格式(从主机到 modem)

偏移量	大小(字节)	类型	说明	值
0	1	uint8_t	设备地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0xfe00
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 modem 发送到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	传输数据的字节数	0x08
3	1	uint8_t	固件次版本号	
4	1	uint8_t	固件主版本号	
5	3	3 字节	保留	
8	1	uint8_t	设备类型 ID	
9	2	uint16_t	保留	
11	2	uint16_t	CRC-16（见附录 1）	

错误回复格式详见附录 2。

Marvelmind robotics

3.4.12 读取用户数据

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	2	uint16_t	数据包中的数据代码	0x0004
4	2	uint16_t	访问模式	0x0000
6	2	uint16_t	CRC-16（参见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x03
2	1	uint8_t	传输数据字节数	0x84
3	1	uint8_t	用户数据总大小	
4	3	3 字节	保留 (0)	
7	128	uint8_t	来自 Hedgehog 的用户数据	
135	2	uint16_t	CRC-16 (参见附录 1)	

错误回复的格式在附录 2 中描述。

来自 Hedgehog 的用户数据是具有以下结构的记录序列:

偏移量	大小 (字节)	类型	描述	值
0	1	uint8_t	Hedgehog 的地址	H
1	1	uint8_t	来自 Hedgehog 的用户数据字节数	M
2	M	uint8_t	来自 hedgehog H 的 M 字节数据	

3.4.13 写入手动设备位置

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机到 Modem）

偏移量	大小（字节）	类型	描述	数值
0	1	uint8_t	信标地址	0x01...0xfe
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0x5003
4	2	uint16_t	访问模式	0x0002
6	1	uint8_t	数据传输字节数	0x20
7	32	结构体	数据结构（见下文）	
39	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Modem 的地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据代码	0x5003
4	2	uint16_t	保留	
6	2	uint16_t	CRC-16（参见附录 1）	

错误回复的格式详见附录 2。

数据结构格式:

偏移量	大小（字节）	类型	说明	值
0	4	int32_t	X 坐标，mm	
4	4	int32_t	Y 坐标，mm	
8	4	int32_t	Z 坐标，mm	
12	1	uint8_t	未说明	0xff
13	4	int32_t	未说明	0
17	4	int32_t	未说明	0
21	4	int32_t	未说明	0
25	1	uint8_t	未说明	0x02
26	6	6 字节	保留	0

3.4.14 写入信标之间的手动距离

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

请求帧格式（从主机发送到 modem）

偏移	大小（字节）	类型	描述	值
0	1	uint8_t	modem 的地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据包中的数据代码	0x4003
4	2	uint16_t	访问模式	0x0000
6	1	uint8_t	数据传输的字节数	0x10
7	16	结构	数据结构(见下文)	
23	2	uint16_t	CRC-16（见附录 1）	

应答帧格式（从 modem 到主机）

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	modem 地址	0xff
1	1	uint8_t	数据包类型	0x10
2	2	uint16_t	数据代码	0x4003
4	2	uint16_t	保留	
6	2	uint16_t	CRC-16（见附录 1）	

错误回复的格式在附录 2 中描述。

数据结构格式：

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	第一个信标的地址	
1	1	uint8_t	第二个信标的地址	
2	4	uint32_t	信标之间的距离，单位 mm	
6	10	10 字节	保留	0

4. 通过 RS-485 进行通信的协议

4.1 用于数据流的 "Marvelmind" 协议

支持的硬件:

Super-Beacon：不支持
Industrial Super-Beacon：支持
Modem HW5.1:不支持
Super-Modem:支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX:不支持
Mini-TX-2:不支持
Modem HW4.9:不支持
Beacon HW4.9：不支持
Beacon HW4.5:不支持

UART 对应章节中描述的所有数据包也可通过 RS-485 使用。

请注意，这些数据仅适用于 Super-Modem 和 Industrial Super-Beacon，因为它们板载了 RS-485 硬件。

4.2 从用户设备读取/写入数据的协议

支持的硬件:

Super-Beacon：不支持

Industrial Super-Beacon:按需提供

Modem HW5.1:不支持

Super-Modem:按需提供

Mini-RX（Badge、Helmet 等）：不支持

Mini-TX:不支持

Mini-TX-2:不支持

Modem HW4.9:不支持

Beacon HW4.9：不支持

Beacon HW4.5：不支持

UART 相应章节中描述的所有数据包均可按需实现。

请注意，这些数据仅适用于 Super-Modem 和 Industrial Super-Beacon，因为它们板载 RS-485 硬件。

4.3 NMEA0183 通信协议

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：支持
Modem HW5.1：不支持
Super-Modem：支持（从 SW V7.000 开始）
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

UART 相应章节中描述的所有数据包也可通过 RS-485 获取。

请注意，这些数据仅适用于 Super-Modem 和 Industrial Super-Beacon，因为它们板载 RS-485 硬件。

5. 通过 SPI 通信的协议

5.1 包含 hedgehog 位置的数据包

支持的硬件:

Super-Beacon：支持
Industrial Super-Beacon：不支持
Modem HW5.1：按需支持
Super-Modem：不支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：支持
Beacon HW4.5：支持

Super-Beacon、Beacon HW4.9 和 Beacon HW4.5 可作为 SPI 从设备工作，并支持读取包含 Hedgehog 位置数据的数据包。Modem HW5.1 具有硬件 SPI 支持，软件支持可按需添加。

5.2 通过 SPI 传输的其他数据

支持的硬件:

Super-Beacon：按需支持
Industrial Super-Beacon：不支持
Modem HW5.1：按需支持
Super-Modem：不支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

对于 Super-Beacon 和 modem HW5.1，可根据需求添加第 2 章中描述的其他数据包的支持

6. 通过 I2C 通信的协议

6.1 用于搭载 PX4 的无人机的罗盘仿真

支持的硬件：

Super-Beacon：支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：不支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

配对的 Super-Beacon 可作为更稳定、更精确的罗盘，通过 I2C 连接到 PX4。为此，您需要购买 MMSW0003 许可证。

6.2 通过 I2C 传输的其他数据

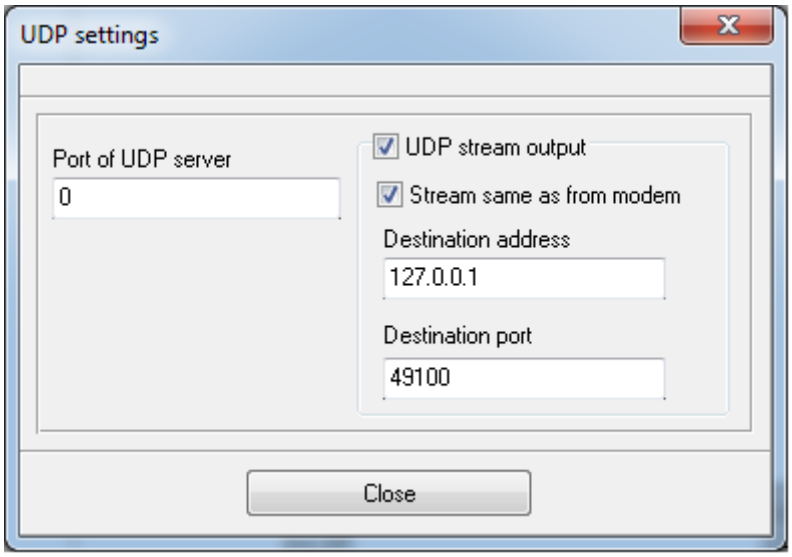
支持的硬件：

Super-Beacon：按需支持
Industrial Super-Beacon：不支持
Modem HW5.1：按需支持
Super-Modem：不支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持

对于 Super-Beacon 和 Modem HW5.1，可按需添加对第 2 章所述其他数据包的支持

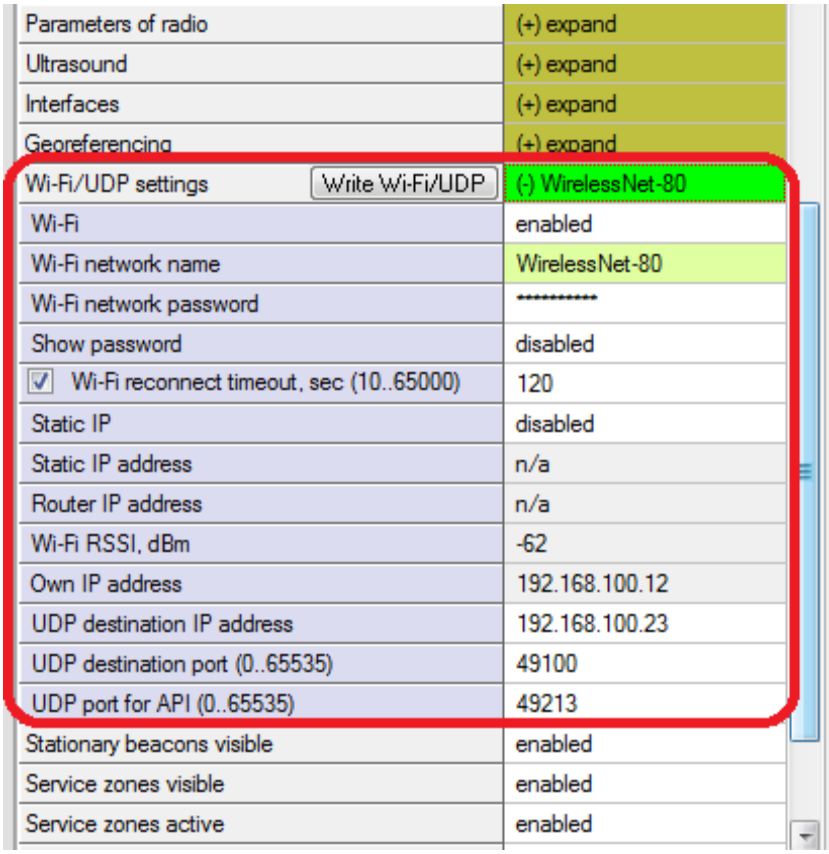
7. 通过 UDP (Wi-Fi) 的通信协议

Dashboard 软件可以通过运行该 Dashboard 的 PC 的网络接口经 UDP 传输数据。目标 IP 地址/端口可通过菜单 File/UDP settings 进行调整：



Super-Modem 内置 Wi-Fi，能够流式传输移动信标的位置及下文所述的其他数据。

可通过 Dashboard 中的 Super-Modem 设置来配置 WiFi 网络和 UDP 流传输：



7.1 Hedgehog 位置数据包

支持的硬件/软件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小	类型	描述	值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0011
4	1	uint8_t	数据大小（字节）	N
5	4	uint32_t	时间戳——从 Dashboard/Super-Modem 运行开始到接收坐标时刻的时间（毫秒）	
9	4	int32_t	信标的 X 坐标，mm	
13	4	int32_t	信标的 Y 坐标，mm	
17	4	int32_t	信标的 Z 坐标，mm	
21	1	uint8_t	标志字节： 位 0：1 - 坐标不可用。不应使用 X、Y、Z 字段中的数据。 位 1：时间戳单位指示符（见注释） 位 2...6：保留（0） 位 7：– 1 – 超出地理围栏区域	
22	1	uint8_t	保留（0）	
23	2	uint16_t	位 0...11：Hedgehog 对在 XY 平面内的方向，单位为十分之一度（0...3600） 位 12：1 – 坐标为信标对的中心点；0 – 坐标为指定的 Hedgehog 位 13...15：保留（0）	
25	2	保留（0）		
27	M=N-22		可选数据字段 – 见列表	
27+M	2	保留（0）		

注意：对于 V6.290 之前的 Dashboard 和 Super-Modem 版本，时间戳单位为 1/64 秒，时间戳单位指示符（标志字节的位 1）为 0。对于 V6.290 及更高版本，时间戳单位为毫秒，时间戳单位指示符为 1。

移动信标位置数据包中的可选数据可包含以下结构：

- 速度数据（7 字节）。应在 Dashboard 中移动信标设置的接口部分启用

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	数据字段代码 = 1 表示速度向量	**1**
1	2	int16_t	X 方向速度，mm/sec	
3	2	int16_t	Y 方向速度，mm/sec	
5	2	int16_t	Z 方向速度，mm/sec	

7.1.1. 带有实时时间戳的 Hedgehog 位置数据包（固件 v7.200+）

支持的硬件/软件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小	类型	描述	值
0	1	uint8_t	Beacon 地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0081
4	1	uint8_t	数据大小（字节）	N
5	8	int64_t	时间戳 – 信标超声波发射的 unix 时间，从 1970.01.01 00:00:00 起的毫秒数。 时间由所有设备、Modem 和 Dashboard 同步。	
13	4	int32_t	信标的 X 坐标，mm	
17	4	int32_t	信标的 Y 坐标，mm	
21	4	int32_t	信标的 Z 坐标，mm	
25	1	uint8_t	标志字节： 位 0：1 - 坐标不可用。不应使用 X、Y、Z 字段中的数据。 位 1：时间戳单位指示符（见注释） 位 2...6：保留（0） 位 7：1 – 超出地理围栏区域	
26	1	uint8_t	保留（0）	
27	2	uint16_t	位 0...11：XY 平面内 Hedgehog 对的方向，十分之一度（0...3600） 位 12：1 – 坐标为信标对中心坐标；0 – 坐标为指定 Hedgehog 的坐标 位 13...15:保留 (0)	
29	2	保留 (0)		
31	M=N-26		可选数据字段 – 参见列表	
31+M	2	保留 (0)		

7.2. 含固定信标位置的数据包

支持的硬件/软件:

Super-Beacon:不支持
Industrial Super-Beacon:不支持
Modem HW5.1:不支持
Super-Modem:支持
Mini-RX(Badge、Helmet 等):不支持
Mini-TX:不支持
Mini-TX-2:不支持
Modem HW4.9:不支持
Beacon HW4.9:不支持
Beacon HW4.5:不支持
Dashboard 软件:支持

数据包格式

偏移	大小（字节）	类型	描述	数值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0012
4	1	uint8_t	传输数据的字节数	1+N*14
5	1	uint8_t	数据包中的信标数量	N
6	1	N*14 字节	N 个 beacon 的数据	

每个 beacon 的数据结构格式（共 N 个）：

偏移量	大小（字节）	类型	说明
0	1	uint8_t	beacon 的地址
1	4	int32_t	beacon 的 X 坐标，mm
5	4	int32_t	beacon 的 Y 坐标，mm
9	4	int32_t	beacon 的 Z 坐标，mm
13	1	uint8_t	保留（0）

7.3. 原始 IMU 数据数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中数据代码	0x0003
4	1	uint8_t	传输数据字节数	
5	2	int16_t	加速度计，X 轴，1 mg/LSB	
7	2	int16_t	加速度计，Y 轴，1 mg/LSB	
9	2	int16_t	加速度计，Z 轴，1 mg/LSB	
11	2	int16_t	陀螺仪，X 轴，0.0175 dps/LSB	
13	2	int16_t	陀螺仪，Y 轴，0.0175 dps/LSB	
15	2	int16_t	陀螺仪，Z 轴，0.0175 dps/LSB	
17	2	int16_t	罗盘，X 轴，1100 LSB/Gauss	
19	2	int16_t	罗盘，Y 轴，1100 LSB/Gauss	
21	2	int16_t	罗盘，Z 轴，980 LSB/Gauss	
23	1	uint8_t	信标地址	
24	5	5 字节	保留 (0)	
29	4	uint32_t	时间戳，ms	
33	8	8 字节	保留	

注意：罗盘数据仅适用于带 IMU 的 HW v4.9 信标。

7.3.1. 带实时时间戳的原始 IMU 数据包（固件 v7.200+）

支持的硬件/软件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	说明	数值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0083
4	1	uint8_t	传输数据的字节数	
5	2	int16_t	加速度计，X 轴，1 mg/LSB	
7	2	int16_t	加速度计，Y 轴，1 mg/LSB	
9	2	int16_t	加速度计，Z 轴，1 mg/LSB	
11	2	int16_t	陀螺仪，X 轴，0.0175 dps/LSB	
13	2	int16_t	陀螺仪，Y 轴，0.0175 dps/LSB	
15	2	int16_t	陀螺仪，Z 轴，0.0175 dps/LSB	
17	2	int16_t	电子罗盘，X 轴，1100 LSB/Gauss	
19	2	int16_t	电子罗盘，Y 轴，1100 LSB/Gauss	
21	2	int16_t	罗盘，Z 轴，980 LSB/Gauss	
23	1	uint8_t	Beacon 的地址	
24	5	5 字节	保留 (0)	
29	8	int64_t	时间戳 – unix 时间，自 1970.01.01 00:00:00 起的毫秒数。 所有设备与 Modem 和 Dashboard 同步的时间。	
37	8	8 字节	保留	

注意：罗盘数据仅适用于配备 IMU 的 HW v4.9 Beacon。

7.4. 原始距离数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0004
4	1	uint8_t	传输数据的字节数	
5	32		数据包（见下文）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	hedgehog 地址	
1	6		距离项 1	
7	6		距离项 2	
13	6		距离项 3	
19	6		距离项 4	
25	4	uint32_t	时间戳 – beacon 超声波发射的内部时间，单位为毫秒，从最近一次唤醒事件的时刻起计（V5.89+）。	
29	2	uint16_t	从超声波发射到当前时间所经过的时间，单位毫秒（V5.89+）	
31	1	uint8_t	保留	

距离项格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址（若该项未填充则为 0）	
1	4	uint32_t	到信标的距离，mm	
5	1	uint8_t	保留（0）	

Marvelmind robotics

7.4.1. 带有实时时间戳的原始距离数据数据包（固件 v7.200+）

支持的硬件/软件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请查看有关时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0084
4	1	uint8_t	传输数据的字节数	
5	32		数据包（见下文）	

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	Hedgehog 地址	
1	6		距离项 1	
7	6		距离项 2	
13	6		距离项 3	
19	6		距离项 4	
25	8	int64_t	时间戳 – 信标超声波发射的 unix 时间，自 1970.01.01 00:00:00 起的毫秒数。 所有设备、modem 和 dashboard 之间同步的时间。	
33	2	uint16_t	从超声波发射到当前时间所经过的时间，单位为毫秒（V5.89+）	
35	1	uint8_t	保留	

距离项目格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址（若该项未填充则为 0）	
1	4	uint32_t	到信标的距离，单位为 mm	
5	1	uint8_t	保留（0）	

7.5. 带有 IMU 融合数据的数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅有关时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0005
4	1	uint8_t	传输数据的字节数	
5	4	int32_t	beacon 的 X 坐标（融合），mm	
9	4	int32_t	beacon 的 Y 坐标（融合），mm	
13	4	int32_t	beacon 的 Z 坐标（融合），mm	
17	2	int16_t	旋转四元数的 W 分量	
19	2	int16_t	旋转四元数的 X 分量	
21	2	int16_t	旋转四元数的 Y 分量	
23	2	int16_t	旋转四元数的 Z 分量	
25	2	int16_t	信标的 X 方向速度（融合），mm/s	
27	2	int16_t	信标的 Y 方向速度（融合），mm/s	
29	2	int16_t	信标的 Z 方向速度（融合），mm/s	
31	2	int16_t	信标的 X 方向加速度，mm/s2	
33	2	int16_t	信标的 Y 方向加速度，mm/s2	
35	2	int16_t	信标的 Z 方向加速度，mm/s2	
37	1	uint8_t	信标地址	
38	1	1 字节	保留（0）	
39	4	uint32_t	时间戳，ms	
43	4	4 字节	保留 (0)	

注意：四元数已归一化为 10000 值。

Marvelmind robotics

7.5.1. 带有实时时间戳的 IMU 融合数据包（固件 v7.200+）

支持的硬件/软件：

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

请参阅关于时间戳的说明。

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0085
4	1	uint8_t	传输数据的字节数	
5	4	int32_t	信标的 X 坐标(融合),mm	
9	4	int32_t	信标的 Y 坐标(融合),mm	
13	4	int32_t	信标的 Z 坐标(融合),mm	
17	2	int16_t	旋转四元数的 W 分量	
19	2	int16_t	旋转四元数的 X 分量	
21	2	int16_t	旋转四元数的 Y 分量	
23	2	int16_t	旋转四元数的 Z 分量	
25	2	int16_t	信标的 X 方向速度（融合），mm/s	
27	2	int16_t	信标的 Y 方向速度（融合），mm/s	
29	2	int16_t	信标的 Z 方向速度（融合），mm/s	
31	2	int16_t	信标的 X 方向加速度，mm/s2	
33	2	int16_t	信标的 Y 方向加速度，mm/s2	
35	2	int16_t	信标的 Z 方向加速度，mm/s2	
37	1	uint8_t	信标地址	
38	1	1 字节	保留 (0)	
39	8	int64_t	时间戳 – unix 时间，自 1970.01.01 00:00:00 起的毫秒数。 时间，由所有设备与 Modem 和 Dashboard 同步	
47	4	4 字节	保留 (0)	

注意：四元数归一化至 10000 值。

7.6. 遥测数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0006
4	1	uint8_t	传输数据的字节数	
5	2	uint16_t	电池电压，mV	
7	1	int8_t	RSSI，dBm	
8	13		保留 (0)	

7.7. 包含质量和扩展位置数据的数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	
1	1	uint8_t	数据包类型	0x47
2	2	uint16_t	数据包中的数据代码	0x0007
4	1	uint8_t	传输的数据字节数	
5	1	uint8_t	设备地址	
6	1	uint8_t	定位质量，%	
7	1	uint8_t	0 = 无地理围栏区域警报 1...255 — 地理围栏区域索引 此字段需要 MMSW0005 许可证。	
8	13		保留 (0)	

7.8. 所有 beacon 的遥测数据包

支持的硬件/软件:

Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持（SSM 固件中）
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：按需

数据包格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	地址	0xff
1	1	uint8_t	数据包类型	0x48
2	2	uint16_t	数据包中的数据代码	0x2001
4	2	uint16_t	传输数据的字节数	
6	N*10		N 个信标的遥测数据（见下表）	

信标遥测项格式

偏移量	大小（字节）	类型	描述	值
0	1	uint8_t	信标地址	
1	2	uint16_t	电源电压，mV	
3	1	int8_t	RSSI，dBm	
4	4	uint32_t	距上次数据更新经过的时间，秒	
8	2	uint16_t	保留 (0)	

7.9. NMEA0183 协议

支持的硬件/软件:

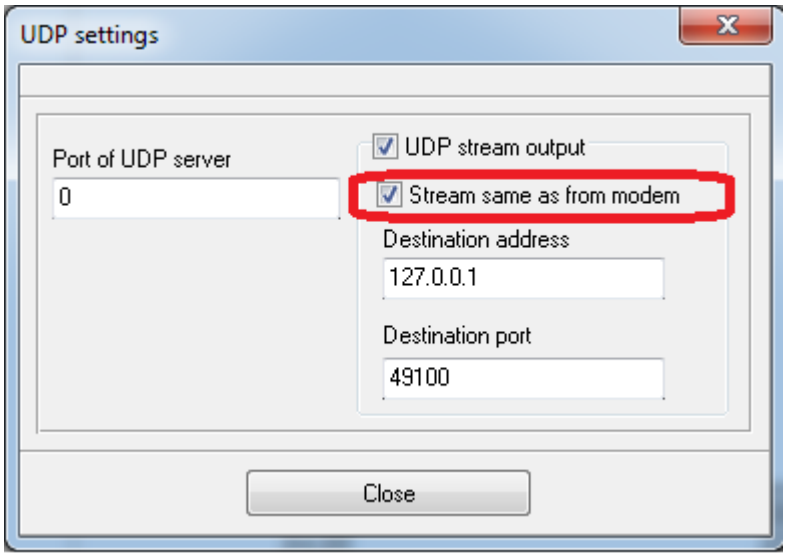
Super-Beacon：不支持
Industrial Super-Beacon：不支持
Modem HW5.1：不支持
Super-Modem：支持
Mini-RX（Badge、Helmet 等）：不支持
Mini-TX：不支持
Mini-TX-2：不支持
Modem HW4.9：不支持
Beacon HW4.9：不支持
Beacon HW4.5：不支持
Dashboard 软件：支持

数据包格式

偏移量	大小（字节）	类型	描述	值
0	N	N 字节	NMEA0183 消息（详见此处说明）	
N	1	uint8_t	移动信标地址	

注意事项:

- 如果在 Dashboard 中为 Super-Modem 的设置界面的 interfaces 部分选择了 NMEA0183 协议，Super-Modem 将通过 UDP 流式传输 NMEA0183 消息
- 如果在 Dashboard 中为 modem 的设置界面的 interfaces 部分选择了 NMEA0183 协议，并且在 Dashboard 的 Settings/UDP Settings 菜单中选择了下方截图所示的选项，Dashboard 将流式传输 NMEA0183 消息：



8. 通过 CAN 进行通信的协议

支持的硬件：

Super-Beacon：不支持
Industrial Super-Beacon：支持
Modem HW5.1:不支持
Super-Modem:按需提供
Mini-RX (Badge、Helmet 等):不支持
Mini-TX:不支持
Mini-TX-2:不支持
Modem HW4.9:不支持
Beacon HW4.9:不支持
Beacon HW4.5:不支持

可根据需求在 Super-Modem 和 Industrial Super-Beacon 中安装 CAN 硬件支持。如果安装了 CAN，则 RS-485 将不可用。

CAN 参数：

波特率：125 kbps。

帧格式：标准。

8.1. "Marvelmind" 流传输协议

在关于 UART 流传输的相应章节中描述的数据包也通过 CAN 传输，CAN 帧 ID 为 0x10。每个 CAN 帧可包含 1 到 8 字节的数据。数据字节数在 CAN 帧的 DLC 字段中指定。

数据以原始流形式传输，因此 CAN 帧可能包含一个数据包的结尾和下一个数据包的开头。用户应接收多个 CAN 帧，将其数据字段放入某个缓冲区，并按照与从 UART 接收数据相同的方式进行处理。

8.2. NMEA0183 通信协议

在关于 UART 流传输的相应章节中描述的数据包也通过 CAN 传输，CAN 帧 ID 为 0x11。每个 CAN 帧可包含 1 到 8 字节的数据。数据字节数在 CAN 帧的 DLC 字段中指定。

数据以原始流形式传输，因此 CAN 帧可能包含一个数据包的结尾和下一个数据包的开头。用户应接收多个 CAN 帧，将其数据字段放入某个缓冲区，并以相同方式进行处理。

9. Dashboard csv 日志文件格式

Dashboard 将固定信标和移动信标的位置以及其他数据存储到位于 dashboard 目录中 "log" 文件夹下的 csv 日志文件中。从 V7.000 版本开始，日志格式已发生变化。之前的格式仅保留用于 modem HW v4.9。

9.1. csv 日志文件格式（dashboard 版本 V7.000+）

在 dashboard V7.000+ 版本的 csv 日志文件中，每个事件作为一行 CSV 记录到日志中，不同的事件对应不同的行格式。同时，所有类型行的起始部分是相同的。

以下是 csv 日志文件中若干行的示例：

```
T2021_11_04__173001_581,user,41,17,14,4.675,2.714,0.250,2,975,100
T2021_11_04__173001_581,user,41,17,15,4.665,2.708,0.250,2,975,114
T2021_11_04__173001_581,user,41,17,26,4.073,1.987,0.250,2,3462,128
T2021_11_04__173001_581,user,41,17,27,4.075,1.987,0.250,2,3462,141
T2021_11_04__173001_581,user,41,17,28,3.588,1.979,0.250,2,3496,155
T2021_11_04__173001_581,user,41,17,29,3.592,1.978,0.250,2,3496,169
T2021_11_04__173001_701,user,43,15,nl
T2021_11_04__173001_728,user,43,27,nl
T2021_11_04__173001_756,user,43,29,nl
```

行的公共部分包括前 3 个字段：

"T2021_11_04__173001_581" —— 本行数据的时间戳：2021.11.04, 17:30:01.581；
"user" —— 用户名（为将来保留）。在未来版本中，dashboard 将支持用户登录；
"41" —— 行类型 ID。不同的行类型在后续字段中有不同的格式。

数据字段中有一些常见的特殊代码：

"nl" —— 无许可证（no license）。需要某种许可证才能填充此字段；
"na" —— 不适用（not applicable）。该字段没有相关数据。例如，如果移动信标未能成功定位，则 X、Y、Z 坐标字段将包含 "na"。

以下章节包含不同类型行的说明。

Marvelmind robotics

9.1.1. 行类型 ID 01 – 链接到地图文件

当地图文件被自动保存或用户按下"Save map"按钮时，会记录此行。

行字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	01 - 行类型 ID（链接到地图文件）
3	此时保存的地图文件名称

9.1.2. 行类型 ID 41 – Marvelmind 协议流记录

当"Interfaces"部分中的 modem 协议设置为"Marvelmind"时，会记录此行。

Marvelmind 协议有不同类型的记录，它们对应日志文件中的不同行，在以下小节中描述。

9.1.2.1. Hedgehog 位置 (41 17) 或 (41 129)

行字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流）
3	17 (0x0011) – hedgehog 位置的数据代码 129 (0x0081) – hedgehog 位置数据代码（带实时时间戳流）
4	Hedgehog 地址
5	Hedgehog X 坐标，单位：米
6	Hedgehog Y 坐标，单位：米
7	Hedgehog Z 坐标，单位：米
8	标志位： Bit 0：1 - 坐标不可用。不应使用 X、Y、Z 字段中的数据。 Bit 1…6：保留 Bit 7：– 1 – 超出地理围栏区域
9	偏航角与标志位： Bit 0…11：hedgehog 对的偏航角，单位：十分之一度（0…3600） Bit 12：1 – 坐标为信标对的中心位置；0 – 坐标为指定 hedgehog 的位置
10	时间偏移，单位：ms。从超声波发射到计算出本行中位置所经过的时间

9.1.2.2. 固定信标位置 (41 18)

该行字段:

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流）
3	18 (0x0012) – 固定信标位置的数据代码
4	固定信标地址
5	信标 X 坐标,单位:米
6	信标 Y 坐标,单位:米
7	信标 Z 坐标,单位:米
8	保留字段

9.1.2.3. 从 hedgehog 到固定信标的原始距离 (41 4) 或 (41 132)

该行的字段:

N	字段说明
0	时间戳(通用字段)
1	用户名(通用字段)
2	41 - 行类型 ID(Marvelmind 协议流)
3	4 (0x0004) – 原始距离的数据代码 132 (0x0084) – 原始距离的数据代码(带实时时间戳流)
4	Hedgehog 地址
5	N – 该行中的距离数量
6	N 个距离子记录(2*N 个字段),见下方
6+N*2+1	时间偏移,单位:ms。从超声波发射到距离测量所经过的时间

距离子记录的字段:

0	固定信标地址
1	到固定信标的距离

Marvelmind robotics

9.1.2.4. 原始 IMU 数据 (41 3) 或 (41 131)

此行需要 MMSW0005 许可证。

行的字段:

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流式传输）
3	3 (0x0003) – 原始 IMU 数据的数据代码 131 (0x0083) – 原始 IMU 数据的数据代码（已为 Hedgehog 启用实时时间戳）
4	Hedgehog 地址
5	加速度计，X 轴，1 mg/LSB
6	加速度计，Y 轴，1 mg/LSB
7	加速度计，Z 轴，1 mg/LSB
8	陀螺仪，X 轴，0.0175 dps/LSB
9	陀螺仪，Y 轴，0.0175 dps/LSB
10	陀螺仪，Z 轴，0.0175 dps/LSB
11	罗盘，X 轴，1100 LSB/Gauss
12	罗盘，Y 轴，1100 LSB/Gauss
13	罗盘，Z 轴，980 LSB/Gauss

9.1.2.5. IMU 融合数据 (41 5) 或 (41 133)

此行需要 MMSW0005 许可证。

该行的字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流传输）
3	5 (0x0005) – IMU 融合数据的数据代码 133 (0x0085) – IMU 融合数据的数据代码（hedgehog 启用实时时间戳）
4	hedgehog 地址
5	信标 X 坐标（融合），米
6	信标 Y 坐标（融合），米
7	信标 Z 坐标（融合），米
8	旋转四元数的 W 字段
9	旋转四元数的 X 字段
10	旋转四元数的 Y 字段
11	旋转四元数的 Z 字段
12	beacon 的 X 方向速度（融合），mm/s
13	beacon 的 Y 方向速度（融合），mm/s
14	beacon 的 Z 方向速度（融合），mm/s
15	beacon 的 X 方向加速度，mm/s2
16	beacon 的 Y 方向加速度，mm/s2
17	beacon 的 Z 方向加速度，mm/s2

9.1.2.6. 遥测数据 (41 6)

行字段:

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流式传输）
3	6 (0x0006) – 遥测数据的数据代码
4	beacon 的地址
5	电源电压，V
6	RSSI，dBm

9.1.2.7. 质量及扩展位置数据 (41 7)

行字段:

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	41 - 行类型 ID（Marvelmind 协议流式传输）
3	7 (0x0007) – 质量和扩展位置数据的数据代码
4	Hedgehog 的地址
5	位置质量，%
6	地理围栏区域编号（此字段需要 MMSW0005 许可证）

9.1.3. 行类型 ID 42 – NMEA0183 流式记录

此行需要 MMSW0005 许可证。

当"Interfaces"部分中的 modem 协议设置为"NMEA0183"时，会记录此行。

行字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	42 - 行类型 ID（NMEA0183 协议流式传输）
3	Hedgehog 的地址
4、5 等	字段顺序遵循 NMEA0183 格式（NMEA0183 记录同样采用逗号分隔值格式）

9.1.4. 行类型 ID 43 – 通过 hedgehog 传输的用户有效负载数据

此行需要 MMSW0005 许可证。

如果 hedgehog 在设置的接口部分启用了非零的有效负载数据大小，且用户设备通过 hedgehog 的 USB 或 UART 传输任何数据，则记录此行。

此外，某些 Marvelmind 设备也支持有效负载数据，例如 robots v100 和 Boxie。

该行的字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	43 - 行类型 ID（用户有效负载）
3	hedgehog 的地址
4、5 等	有效负载数据的逗号分隔字节序列（每个字段为 1 字节）

9.1.4.1. Robot v100 的有效负载遥测数据

该行的通用格式对应于用户有效负载数据格式。

数据字节（从该行的第四个字段开始）构成具有以下描述格式的数据记录。

多字节值从低字节开始排列（小端格式）。

Robot v100 遥测记录 N3：

偏移量	大小（字节）	类型	描述	值
0	2	uint16_t	记录 ID	0x3003
2	2*12		12 个激光雷达的距离（每个激光雷达 2 字节） 每个激光雷达数据采用以下格式： Bit 0...11 – 激光雷达测得的距离，单位 mm Bit 12...15 – 距离测量状态 Status = 0 – 距离已测量 Status <> 0 – 距离未测量	
26	1	uint8_t	激光雷达总体状态： Bit0: 1 – 激光雷达读取成功 0 – 激光雷达读取失败 Bit 1...7 – 保留（0）	
27	1	uint8_t	机器人状态： 0：机器人正常停止 1：机器人因报警而停止 2：机器人正在自主移动 3：机器人正在充电	
28	1	uint8_t	RV - 机器人电池电压。	

			V= (RV/10) + 20 伏	
29	2	int16_t	机器人供电电流，x10 mA 如果数值为负，则表示机器人电池正以此电流充电。	N
31	1	uint8_t	剩余电池容量，%	
32	2	uint16_t	机器人供电功率，瓦	
34	2	uint16_t	左轮速度，mm/s	
36	2	uint16_t	右轮速度，mm/s	
37	1	uint8_t	左电机功率，%	
38	1	uint8_t	右电机功率，%	
39	2	uint16_t	期望速度，mm/s	
41	2	int16_t	机器人 X 坐标，cm	
43	1	uint8_t	状态标志： Bit 0...3 – 保留 Bit 4：1 – 超声波跟踪错误 Bit 5...7 - 保留	
44	2	int16_t	机器人 Y 坐标，cm	

机器人 v100 遥测记录 N4：

偏移量	大小（字节）	类型	描述	值
0	2	uint16_t	记录 ID	0x3004
2	2*12		12 个激光雷达的距离数据（每个激光雷达 2 字节） 每个激光雷达数据的格式如下： Bit 0...11 – 激光雷达测得的距离，单位 mm Bit 12...15 – 距离测量状态 Status = 0 – 距离已测量 Status <> 0 – 距离未测量	
26	1	uint8_t	激光雷达总体状态： Bit0: 1 – 激光雷达读取成功 0 – 激光雷达读取失败 Bit 1...7 – 保留（0）	
27	1	uint8_t	机器人状态： 0：机器人正常停止 1：机器人因报警而停止 2：机器人正在自主移动 3：机器人正在充电	
28	1	uint8_t	运动程序中当前项目的索引	
29	1	uint8_t	运动程序中的项目总数	
30	1	uint8_t	保留	
31	1	uint8_t	LiDAR 检测到的障碍物方向： 0 – 无 1 – 前方 2 – 左侧 3 - 右侧	
32	1	uint8_t	LiDAR 最小报警距离，x2cm	
33	1	uint8_t	引起报警的当前 LiDAR 测量距离，x2cm	
34	3		保留	
36	2	int16_t	机器人 X 位置，cm	

Marvelmind robotics

38	2	int16_t	机器人 Y 位置，cm	

9.1.4.2. Robot Boxie 的有效载荷遥测数据

该行的通用格式对应于用户有效载荷数据格式。

数据字节（从该行的第四个字段开始）构成数据记录，其格式如下所述。

多字节值从低字节开始放置（小端格式）。

Robot Boxie 遥测记录 N1：

偏移	大小（字节）	类型	描述	值
0	2	uint16_t	记录 ID	0x3101
2	2*12		12 个 LiDAR 测得的距离（每个 LiDAR 2 字节） 每个 LiDAR 数据具有以下格式： 位 0...11 – 该 LiDAR 测得的距离，mm Bit 12...15 – 距离测量状态 Status = 0 – 已测量距离 Status <> 0 – 未测量距离	
26	1	uint8_t	激光雷达总体状态： Bit0：1 – 激光雷达读取成功 0 – 激光雷达读取失败 Bit 1...7 – 保留（0）	
27	1	uint8_t	机器人状态： 0：机器人正常停止 1：机器人因报警而停止 2：机器人自主移动中 3：机器人充电中	
28	1	uint8_t	RV - 机器人电池电压，x100 mV。 例如，值 118 表示 11.8V	
29	2	int16_t	机器人供电电流，x10 mA 例如，值 123 表示 1.230 A	
31	1	uint8_t	保留	
32	1	uint8_t	左电机功率, %	
33	1	uint8_t	右电机功率, %	
34	2	uint16_t	左轮速度, mm/s	
36	2	uint16_t	右轮速度, mm/s	
37	2	int16_t	左电机里程计已行驶路径, cm	
39	2	int16_t	右电机里程计已行驶路径, cm	
41	2	int16_t	机器人 X 坐标, cm	
43	1	uint8_t	状态标志: Bit 0...1 – 保留 Bit 2: 1 – 运动程序正在执行 位 2: 1 – 运动已暂停 位 4: 1 – 超声波跟踪错误 位 5...7 - 保留	
44	2	int16_t	机器人 Y 坐标，cm	

Marvelmind robotics

机器人 Boxie 遥测记录 N3：

偏移量	大小（字节）	类型	描述	值
0	2	uint16_t	记录 ID	0x3103
2	2	int16_t	角度控制 PID 调节器的当前 "P" 值	
4	2	int16_t	角度控制 PID 调节器的当前 "I" 值	
6	2	int16_t	角度控制 PID 调节器的当前 "D" 值	
8	2	int16_t	使用 EKF 滤波器计算的机器人 X 坐标，单位：cm	
10	2	int16_t	使用 EKF 滤波器计算的机器人 Y 坐标，单位：cm	
12	2	int16_t	保留	
13	1	uint8_t	激光雷达综合状态： Bit0：1 – 激光雷达读取成功 0 – 激光雷达读取失败 Bit 1...7 – 保留 (0)	
14	1	uint8_t	机器人状态： 0：机器人正常停止 1：机器人因任何报警而停止 2：机器人自主移动中 3：机器人充电中	
15	1	uint8_t	当前移动步骤索引（第一个路径点为 0，第二个为 1，依此类推）	
16	1	uint8_t	当前程序中的总运动步数	
17	1	uint8_t	运动标志： 位 0：1 – "永久运行"选项 位 1...7 - 保留	
18	1	uint8_t	0 – 无激光雷达报警 1...12 - 触发报警的激光雷达索引	
19	1	uint8_t	激光雷达最小报警距离，x2cm	
20	1	uint8_t	触发报警的当前激光雷达测量距离，x2cm	
21	2	int16_t	机器人上配对信标的角度，度	
23	1	uint8_t	机器人期望速度（用户设置），cm/s	
24	2	int16_t	机器人 X 坐标，cm	
26	2	int16_t	机器人 Y 坐标，cm	
28	2	int16_t	保留	
30	2	int16_t	保留	
32	2	int16_t	机器人期望方向与当前朝向之间的偏差角度，单位：度	
34	1	uint8_t	保留	
35	2	int16_t	融合后的机器人朝向角，单位：度	
36	5	5 字节	保留	

9.1.5. 行类型 ID 44 – dashboard 实时播放器位置

此行需要 MMSW0005 许可证。

如果启用了实时播放器，则会为 hedgehog 记录此行。实时播放器提供 100 Hz 的位置数据。

该行的字段：

N	字段说明
0	时间戳（通用字段）
1	用户名（通用字段）
2	44 - 行类型 ID（实时播放器位置）
3	Hedgehog 的地址
4	保留字段
5	Hedgehog X 坐标，单位：米
6	Hedgehog Y 坐标，单位：米
7	Hedgehog Z 坐标，单位：米

Marvelmind robotics

9.1.6. 行类型 ID 55 – 信标简短遥测

该行包含从 SSM（Super Super-Modem）接收到的信标遥测数据。

行的字段：

N	字段描述
0	时间戳（通用字段）
1	用户名（通用字段）
2	55 - 行类型 ID（简短遥测）
3	信标的地址
4	信标电源电压低字节：V0
5	信标电源电压高字节：V1 信标电源电压为 V= V0+V1*256 mV
6	采用二进制补码表示的 RSSI：R0 如果 R0 小于 128，则 RSSI= R0 dBm 如果 (R0 >= 128) RSSI= R0-256 dBm
7	保留
8	保留

Marvelmind robotics

9.2. csv 日志的旧格式(V7.000 之前的 Dashboard 或 modem HW v4.9)

下图展示了日志文件的旧格式:

Format of CSV file recorded by dashboard

```
1608733078625,0,2911360,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078656,31,2911391,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078656,0,2911391,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078671,15,2911406,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078671,0,2911406,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078687,16,2911422,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078687,0,2911422,60,0.805,1.160,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078843,156,2911578,60,0.806,1.159,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078843,0,2911578,60,0.806,1.159,1.000,50,1.784,58,1.519,255,0,0,0,0,
1608733078859,16,2911594,60,0.806,1.159,1.000,50,1.778,58,1.528,255,0,0,0,0,
1608733078859,0,2911594,60,0.806,1.159,1.000,50,1.778,58,1.528,255,0,0,0,0,
1608733078890,31,2911625,60,0.807,1.159,1.000,50,1.778,58,1.528,255,0,0,0,0,
1608733078890,0,2911625,60,0.807,1.159,1.000,50,1.778,58,1.528,255,0,0,0,0,
```

Case specific parameters: robot state, some users telemetry or so on

32-bit status word.
Bit 0: 1 = geofencing zone alarm
Bit 1...31: reserved

Coordinates of hedgehog: X,Y,Z (meters)

Addresses of stationary beacons and distances from hedgehog to them (meters)

Separator. 255 means end of raw distances list

Address of hedgehog

Time from running dashboard, ms

Time from previous record, ms

Time from 1 january 1970 (Unix time), ms

Marvelmind robotics

10. Marvelmind API

Marvelmind API 库由 Marvelmind Dashboard 软件使用,并为用户软件提供接口。 API 以动态链接库 (DLL) 形式提供给 MS Windows,以共享库形式提供给 Linux(x86 和 ARM 平台)。 API 通过 USB(虚拟串口)连接到 modem,并实现与 modem 的通信协议。

除了 API 库外,软件包还包含 C 语言示例软件,该示例软件用于测试 API,并包含所有 API 函数的调用。

该示例可作为开发用户软件的基础,也可作为将 API 库接口(文件 'marvelmind_api.c')移植到其他编程语言的基础。

已测试的环境:

1. MS Windows 10; CPU: Intel Core i5
2. Ubuntu 20.04; CPU: Intel Core i5
3. Raspbian (2018-11-13-raspbian-stretch-full); 平台: Raspberry Pi 3 Model B+

10.1. Windows 安装

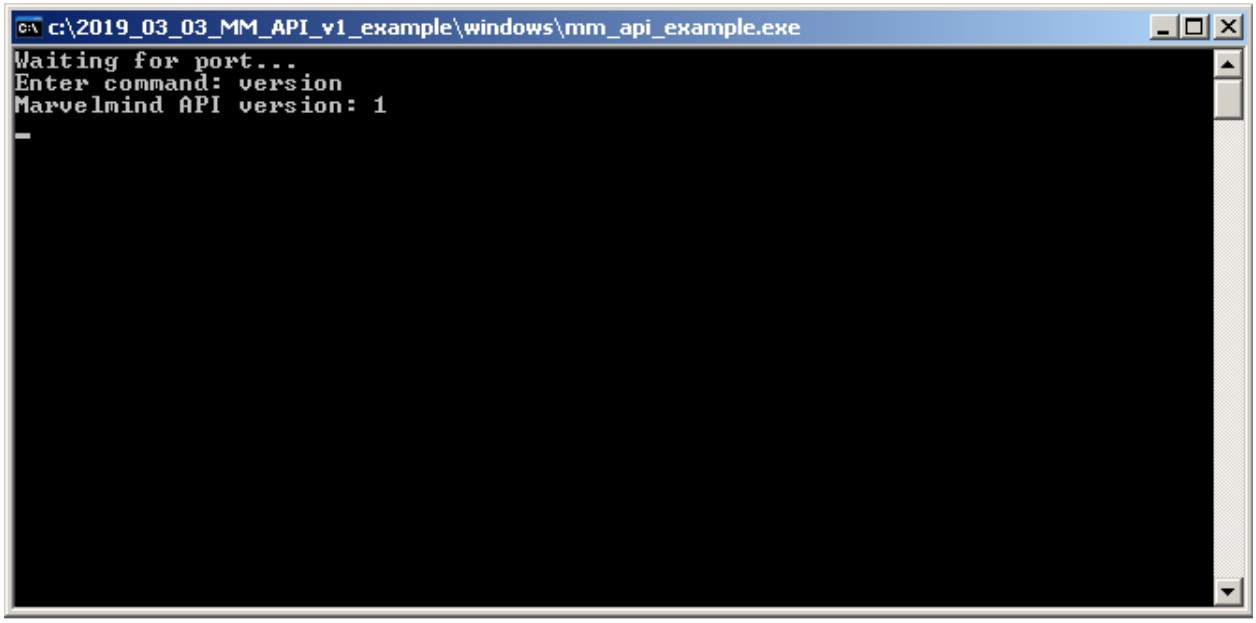
- 下载 Marvelmind API 软件包。将 Dashboard API 和示例软件复制到您将用于该程序的目录。由于 Windows 版本的示例附带预构建的可执行文件,您可以立即从 API 软件包中的 'windows' 目录运行 'mm_api_example.exe'。

10.2. Linux 安装

- 下载 Marvelmind API 软件包。将 Dashboard API 复制到您将用于该程序的目录。请注意,Linux 版本提供两种硬件平台版本：x86(大多数基于 Intel 或 AMD CPU 的笔记本电脑)和 arm(例如 Raspberry PI 等单板计算机)
- 将与您的平台对应的库 libdashapi.so 复制到目录 /usr/local/lib,方法是在包含 libdashapi.so 的目录中打开终端并执行命令 sudo cp libdashapi.so /usr/local/lib。之后,在终端中执行 sudo ldconfig。
- 您可能需要通过将用户添加到 dialout 组来为用户授予访问串口的权限：
  - 在终端中执行：sudo adduser $USER dialout
  - 在目录 /etc/udev/rules.d 中添加文件 "99-tty.rules",内容如下:
    #Marvelmind 串口规则
    KERNEL=="ttyACM0",GROUP="dialout",MODE="666"
- 构建示例软件 — 在软件包附带的 'source' 目录中打开终端,执行 'make all'
- 在终端中输入 './mm_api_example' 运行示例

10.3. 检查与 API 的连接

运行示例软件后,在终端中按 "空格" 键,输入命令 'version' 并按回车。如果示例软件打印出 API 的版本,则表明其可与 API 库通信。



10.4. Marvelmind API 库说明

API 以 MS Windows 的动态链接库 (DLL) 和 Linux(x86 和 ARM 平台)的共享库形式提供。该库包含一组函数,用于通过连接到计算机 USB 端口的 Modem 对 Marvelmind 系统进行监控和控制。本节文档包含所有这些函数的说明。

为了更好地兼容不同的编程语言,大多数复杂数据结构通过无类型内存指针传递。函数说明中包含每个数据字段在内存池中的偏移量。在示例软件的 'marvelmind_api.c' 文件中,您可以看到内存池与 C 结构体字段之间的数据传输实现。

说明中的参数类型以 C 语法显示。以下是这些类型的描述:

类型	大 小 (字节)	说明
bool	1	布尔类型。零表示假,非零表示真
uint8_t	1	无符号整数值,0…255
int8_t	1	二进制补码格式的有符号整数值,-128…127
uint16_t	2	无符号整数值，0…65535
int16_t	2	二进制补码格式的有符号整数值，-32768…32767
uint32_t	4	无符号整数值，0…4294967295
int32_t	4	二进制补码格式的有符号整数值， -2147483648…2147483647
void *	4/8	内存指针（内存地址）。 32 位平台为 4 字节，64 位平台为 8 字节。

每个函数描述都包含该函数可用的一组 API 版本。新的 API 版本将支持 Marvelmind 系统中新功能对应的更多函数。目前并非 Dashboard 中的所有功能都可通过 API 使用，如果您需要更多的 API 函数，请通过 info@marvelmind.com 联系我们。

支持的函数列表：

函数	API 版本	所需许可证
获取 Marvelmind API 库的版本	V1+	无
获取最近一次错误	V6+	无
尝试打开串行端口	V1+	无
按指定名称尝试打开串行端口	V2+	无
尝试打开 UDP 端口	V9+	无
关闭串行端口	V1+	无
获取 Marvelmind 设备的版本和 CPU ID	V1+	无
获取设备列表	V1+	无
唤醒设备	V1+	无
使设备进入睡眠	V1+	无
从信标获取遥测数据	V1+	无
获取最新位置数据	V1+	无
获取最新位置数据（含角度）	V3+	无
设置信标位置	V3+	MMSW0005
设置信标之间的距离	V4+	MMSW0005
获取最新原始距离数据	V1+	无
获取 hedgehog 的高度	V4+	无
设置 hedgehog 的高度	V4+	MMSW0005
获取子地图中固定信标的高度	V4+	无
设置子地图中固定信标的高度	V4+	MMSW0005
获取位置更新速率设置	V1+	无
设置位置更新速率	V1+	MMSW0005
添加子地图	V1+	MMSW0005
删除子地图	V1+	MMSW0005
冻结子地图	V1+	MMSW0005
解冻子地图	V1+	MMSW0005
获取子地图设置	V1+	无
设置子地图设置	V1+	MMSW0005
冻结地图	V4+	MMSW0005
解冻地图	V4+	MMSW0005
获取信标的超声波设置	V1+	无
设置信标的超声波设置	V1+	MMSW0005
擦除地图	V1+	MMSW0005
将设备重置为默认设置	V1+	MMSW0005
将信标连接到坐标轴	V2+	MMSW0005
读取 modem 的配置内存转储	V3+	MMSW0005
写入 modem 的配置内存转储	V3+	MMSW0005
从 modem 获取空气温度设置	V3+	无
在 modem 中设置空气温度	V3+	无
设备软件复位	V3+	无
获取信标实时播放器设置	V6+	无
设置信标实时播放器设置	V6+	MMSW0005
获取地理参照设置	V6+	无
设置地理参照设置	V6+	MMSW0005
获取位置更新模式	V6+	无
设置位置更新模式	V6+	MMSW0005
更新位置命令	V6+	MMSW0005
设置信标的地理围栏报警状态	V9+	MMSW0005

		MMSW0006
发送通用用户负载数据	V9+	MMSW0005
获取通用用户负载数据	V9+	MMSW0005
发送手动距离测量命令	V9+	MMSW0011
从 modem 获取流数据	V9+	无
检查设备类型是否为 modem	V1+	无
检查设备类型是否为固定信标	V1+	无
检查设备类型是否为 hedgehog	V1+	无

10.4.1. 获取 Marvelmind API 库的版本

读取 API 库的版本。用于确保所需函数在此版本的库中可用。

函数名称：mm_api_version
C 语言声明：bool mm_api_version(void *pdata);
适用的 API 版本：V1+

许可证要求：无

返回值：

类型	说明
bool	true – 函数执行成功 false – 执行出错

参数：

类型	说明
void *	指向待填充数据的指针

通过指针返回的数据结构。

类型	说明
uint32_t	API 库的版本

Marvelmind robotics

10.4.2. 获取最近一次错误

读取 API 库最近一次操作的状态，以区分错误原因。

函数名称：mm_get_last_error
C 语言声明：bool mm_get_last_error(void *pdata);
适用的 API 版本：V6+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功 false – 执行出错

参数：

类型	描述
void *	指向待填充数据的指针

通过指针返回的数据结构。

类型	描述
uint32_t	上一次操作的状态： 0：操作成功执行 1：通信错误 2：打开串口错误 3：需要许可证

Marvelmind robotics

10.4.3. 打开串口

打开通过 USB（虚拟串口）连接 Marvelmind 设备（modem 或信标）的端口。您无需指定串口名称，因为 API 会搜索所有串口并检查其是否对应 Marvelmind 设备。

函数名称：mm_open_port
C 语言中的声明：bool mm_open_port ();
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功，端口已打开 false – 执行出错

参数：无

Marvelmind robotics

10.4.4. 按指定名称打开串行端口

打开 Marvelmind 设备（modem 或信标）通过 USB（虚拟串行端口）连接的端口。函数尝试打开指定名称的端口。

函数名称：mm_open_port_by_name
C 语言声明：bool mm_open_port_by_name(void *pdata);
可用的 API 版本：V2+

需要许可证：无

返回值：

类型	说明
bool	true – 函数执行成功，端口已打开 false – 执行出错

参数：

类型	说明
void *	指向串口名称的指针——以零结尾的 ASCII 字符序列 (ASCIIZ)

Marvelmind robotics

10.4.4.1. 打开 UDP 端口

允许通过 UDP 而非 USB 与 Super-Modem 建立通信。

函数名称：mm_open_port_udp
C 语言声明：bool mm_open_port_udp(void *pdata);
适用的 API 版本：V9+

所需许可证：无

返回值：

类型	说明
bool	true – 函数成功执行，UDP 端口已打开 false – 执行出错

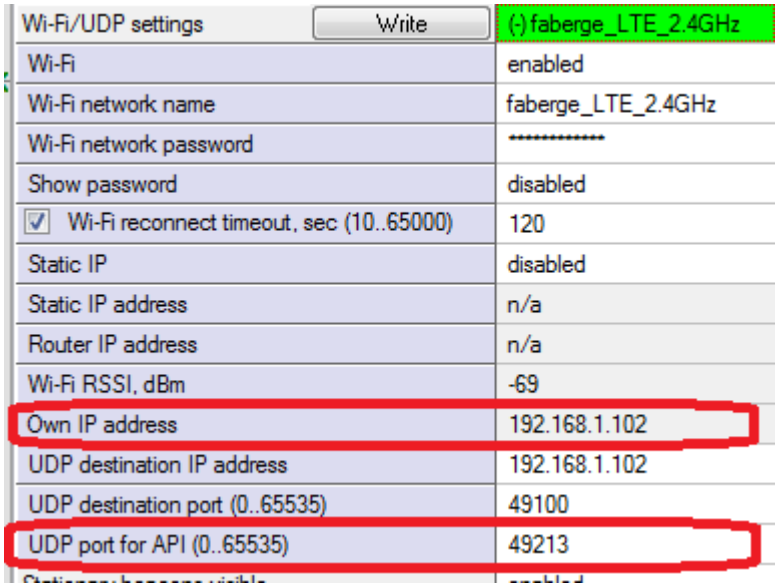
参数：

类型	说明
void *	指向 UDP 设置结构体的指针（见下文）

指针所指向的数据结构：

类型	描述
uint16_t	连接的 UDP 端口
uint16_t	通信超时时间，ms
uint16_t	保留
最多 255 字节	IP 地址——以零结尾的 ASCII 字符序列（ASCIIZ）

IP 地址和 UDP 端口应与 Super-Modem 的设置相对应（见下方截图）。



10.4.5. 关闭串行端口

如果端口先前已由 mm_open_port 函数打开，则关闭该端口。
函数名称：mm_close_port
C 语言中的声明：bool mm_close_port ();
适用的 API 版本：V1+

许可要求：无

返回值：

类型	描述
bool	true – 函数执行成功，端口已关闭 false – 执行出错

参数：无

Marvelmind robotics

10.4.6. 获取 Marvelmind 设备的版本和 CPU ID

读取版本和 CPU ID。版本包括固件版本和设备硬件类型的相关信息。CPU ID 是设备单品的唯一 ID。

函数名称：mm_get_device_version_and_id
C 语言中的声明：bool mm_get_device_version_and_id (uint8_t address, void *pdata);

适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已获取版本和 CPU ID 数据 false – 执行出错

参数：

类型	描述
uint8_t	Marvelmind 设备地址（1…254）
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
uint8_t	固件主版本号（示例：对于版本 V6.07a 为 "6"）
uint8_t	固件次版本号（示例：对于版本 V6.07a 为 "7"）
uint8_t	固件第二次版本号（示例：对于版本 V6.07a 为 "1"）
uint8_t	设备类型 ID（参见附录）。
uint8_t	固件选项（待定）。
uint32_t	CPU ID。以十六进制打印此值可得到 Dashboard 中及设备贴纸上所示形式的 CPU ID。

10.4.7. 获取设备列表

读取 Modem 已知的 Marvelmind 设备列表。该列表包含通过无线电连接到 Modem 网络的所有设备，包括休眠设备。

函数名称：mm_get_devices_list
C 语言声明：bool mm_get_devices_list (void *pdata);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功，已获取设备列表 false – 执行出错

参数：

类型	说明
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	说明
uint8_t	列表中后续设备的数量（N）
N*9 字节	N 个设备结构体的序列，详见下表

列表中每个设备的结构：

类型	说明
uint8_t	设备地址
bool	true = 地址重复 - 发现多个设备使用相同地址 false = 地址未重复
bool	true = 设备处于休眠状态 false = 设备未处于休眠状态
uint8_t	固件主版本号（示例："6"，对应版本 V6.07a）
uint8_t	固件次版本号（示例："7"，对应版本 V6.07a）
uint8_t	固件第二次版本号（示例："1"，对应版本 V6.07a）
uint8_t	设备类型 ID（见附录）。
uint8_t	固件选项（待定）。
uint8_t	标志位： Bit 0: 1 – 设备连接完成 – 设备已确认连接 0 – 等待设备确认（类似 Dashboard 中的"Connecting...")。 Bit 1...7 - 待定

10.4.8. 唤醒设备

向指定设备发送唤醒命令。如果已发送唤醒命令且该设备存在，该设备将在几秒钟内连接到 Modem，并出现在设备列表中。

函数名称：mm_wake_device
C 语言中的声明：bool mm_wake_device (uint8_t address);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数成功执行，已发送唤醒命令 false – 执行出错

参数：

类型	描述
uint8_t	1…254 - 要唤醒的 Marvelmind 设备地址 0 – 唤醒所有设备

10.4.9. 使设备进入休眠

使现有设备进入休眠。

函数名称：mm_send_to_sleep_device
C 语言声明：bool mm_send_to_sleep_device (uint8_t address);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已发送休眠命令 false – 执行出错

参数：

类型	描述
uint8_t	1…254 - 需进入休眠的 Marvelmind 设备地址 0 – 使所有设备进入休眠

10.4.10. 从信标获取遥测数据

读取 Marvelmind 信标的遥测数据。

函数名称：mm_get_beacon_telemetry
C 语言声明：bool mm_get_beacon_telemetry (uint8_t address, void *pdata);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已获取遥测数据 false – 执行出错

参数：

类型	描述
uint8_t	Marvelmind 信标地址 (1…254)
void *	指向待填充数据的指针

通过指针返回的数据结构:

类型	描述
uint32_t	信标工作时间（秒）（从复位或唤醒开始计时）。
int8_t	RSSI，dBm – 无线信号强度
int8_t	测量温度，°C
uint16_t	电源电压，mV
16 字节	保留 (0)

10.4.11. 获取最新位置数据

从 Modem 读取最新更新的坐标数据包。如果可用，还读取用户有效载荷数据。

函数名称: mm_get_last_locations
C 语言声明: bool mm_get_last_locations(void *pdata);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功，已获取位置数据 false – 执行出错

参数：

类型	说明
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	说明
18*6 字节	6 个 18 字节的数据结构，包含最新更新的坐标，详见下表
bool	true – 有新的原始距离数据可读取
5 字节	待定
uint8_t	用户有效载荷数据大小 (M)
M 字节	用户有效载荷数据

每个位置数据项的结构：

类型	描述
uint8_t	设备地址 (1...254) 0 - 此数据项未填充
uint8_t	头索引 (待定)
int32_t	X 坐标，mm
int32_t	Y 坐标，mm
int32_t	Z 坐标，mm
uint8_t	状态标志（TBD）
uint8_t	定位质量，0...100%
uint8_t	TBD
uint8_t	TBD

Marvelmind robotics

10.4.12. 获取最新位置数据（含角度）

从 Modem 读取最新更新的坐标数据包（对于配对信标含角度）。如有可用，也会读取用户负载数据。

函数名称：mm_get_last_locations2
C 语言声明：bool mm_get_last_locations2(void *pdata);
可用的 API 版本：V3+
所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已获取位置数据 false – 执行错误

参数：

类型	描述
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
20*6 字节	6 个 20 字节数据结构，包含最近更新的坐标，见下表
bool	true – 有新的原始距离数据可读取
5 字节	待定
uint8_t	用户载荷数据大小 (M)
M 字节	用户载荷数据

每个位置数据项的结构：

类型	描述
uint8_t	设备地址 (1…254) 0 - 此数据项未填充
uint8_t	头索引 (TBD)
int32_t	X 坐标，mm
int32_t	Y 坐标，mm
int32_t	Z 坐标，mm
uint8_t	状态标志 (TBD)
uint8_t	定位质量，0…100%
uint8_t	TBD
uint8_t	TBD
uint16_t	位 0…11 – 旋转角度，单位为 1/10 度（如启用配对信标功能） 位 12 – 1 = 角度不可用 位 13…15 - 保留

10.4.13. 设置信标位置

手动设置指定信标的位置。

函数名称：mm_set_beacon_location
C 语言声明：bool mm_set_beacon_location (uint8_t address, void *pdata);
适用的 API 版本：V3+
所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，位置已更新 false – 执行错误

参数：

类型	描述
uint8_t	信标地址
void *	指向位置数据缓冲区的指针

指针所指向的数据结构（应在函数调用前填充）：

类型	描述
int32_t	信标的新 X 坐标，mm
int32_t	信标的新 Y 坐标，mm
int32_t	信标的新 Z 坐标，mm

10.4.14. 设置信标之间的距离

手动设置信标之间的距离。

函数名称：mm_set_beacons_distance
C 语言声明：bool mm_set_beacons_distance (void *pdata);
适用的 API 版本：V4+
所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数成功执行，距离已写入 false – 执行出错

参数：

类型	描述
void *	指向距离数据缓冲区的指针

指针所指向的数据结构（应在函数调用前填写）：

类型	描述
uint8_t	第一个信标的地址
uint8_t	第二个信标的地址
int32_t	信标之间的距离，单位 mm

10.4.15. 获取最新的原始距离数据

从 Modem 读取最新更新的原始距离数据包。

函数名称：mm_get_last_distances
C 语言声明：bool mm_get_last_distances(void *pdata);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功，已获取原始距离数据 false – 执行出错

参数：

类型	说明
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	说明
uint8_t	原始距离数据项的数量 (N)。 每次请求的最大原始距离数为 16：N<=16
9*N 字节	N 个 9 字节数据结构，包含最近的原始距离，参见下表

每个原始距离数据项的结构：

类型	描述
uint8_t	超声波 RX 设备地址 (1…254) 0 - 此数据项未填充
uint8_t	RX 探头索引 (TBD)
uint8_t	超声波 TX 设备地址 (1…254) 0 - 此数据项未填充
uint8_t	TX 探头索引 (TBD)
uint32_t	从 TX 设备到 RX 设备的距离，mm
uint8_t	TBD

Marvelmind robotics

10.4.16. 获取 hedgehog 的高度

返回移动信标（hedgehog）的高度。

函数名称：mm_get_hedge_height
C 语言声明：bool mm_get_hedge_height (uint8_t address, void *pdata);
适用的 API 版本：V4+
所需许可证：无

返回值：

类型	说明
bool	true – 函数成功执行，已返回高度 false – 执行出错

参数：

类型	说明
uint8_t	hedgehog 的地址
void *	指向高度数据缓冲区的指针

指针所指数据的结构：

类型	描述
int32_t	Hedgehog 的高度，mm

Marvelmind robotics

10.4.17. 设置 Hedgehog 的高度

设置移动信标（Hedgehog）的高度。

函数名：mm_set_hedge_height
C 语言声明：bool mm_set_hedge_height (uint8_t address, void *pdata);
可用于 API 版本：V4+
需要许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数成功执行，高度已更改 false – 执行错误

参数：

类型	描述
uint8_t	hedgehog 的地址
void *	指向存放高度数据缓冲区的指针

指针所指的数据结构（应在函数调用前填充）：

类型	描述
int32_t	hedgehog 的高度，mm

10.4.18. 获取子地图中固定信标的高度

返回子地图中固定信标的高度。

函数名：mm_get_beacon_height
C 语言声明：bool mm_get_beacon_height (uint8_t address, void *pdata);
适用的 API 版本：V4+
所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，返回高度 false – 执行出错

参数：

类型	说明
uint8_t	信标地址
void *	指向高度数据缓冲区的指针

指针所指向的数据结构：

类型	说明
uint8_t	子地图 ID，应在函数调用前填写
int32_t	信标高度，mm

Marvelmind robotics

10.4.19. 设置子地图中固定信标的高度

设置子地图中固定信标的高度。

函数名：mm_set_beacon_height
C 语言声明：bool mm_set_beacon_height (uint8_t address, void *pdata);
适用的 API 版本：V4+
所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，高度已更改 false – 执行出错

参数：

类型	描述
uint8_t	信标地址
void *	指向高度数据缓冲区的指针

指针所指向的数据结构（应在函数调用前填充）：

类型	描述
uint8_t	子地图 ID
int32_t	信标高度，单位 mm

10.4.20. 获取位置更新速率设置

从 modem 读取位置更新速率设置。

函数名称：mm_get_update_rate_setting
C 语言中的声明：bool mm_get_update_rate_setting (void *pdata);

适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已获取更新速率 false – 执行出错

参数：

类型	描述
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
uint32_t	位置更新速率设置，单位为 mHz。因此，1 Hz 返回 1000，16 Hz 返回 16000，0.05 Hz 模式返回 50。

Marvelmind robotics

10.4.21. 设置位置更新速率

将位置更新速率设置写入 Modem。

函数名称：mm_set_update_rate_setting
C 语言声明：bool mm_set_update_rate_setting (void *pdata);

适用的 API 版本：V1+

所需许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行，更新速率已更改 false – 执行出错

参数：

类型	说明
void *	数据指针

指针所指数据的结构（应在调用函数之前填充）：

类型	描述
uint32_t	位置更新速率设置，单位为 mHz。因此，1 Hz 返回 1000，16 Hz 返回 16000，0.05 Hz 模式返回 50。系统将从以下系列中选择最接近指定更新速率的值：0.05 Hz、0.1 Hz、0.2 Hz、0.5 Hz、1 Hz、2 Hz、4 Hz、8 Hz、12 Hz、16 Hz、16+Hz。

Marvelmind robotics

10.4.22. 添加子地图

添加新的子地图。

函数名称：mm_add_submap
C 语言声明：bool mm_add_submap (uint8_t submapId);
适用的 API 版本：V1+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，已添加子地图 false – 执行出错

参数：

类型	描述
uint8_t	要添加的子地图 ID (0...254)

Marvelmind robotics

10.4.23. 删除子地图

删除现有子地图。

函数名称：mm_delete_submap
C 语言声明：bool mm_delete_submap (uint8_t submapId);
适用的 API 版本：V1+

所需许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行，子地图已被删除 false – 执行错误

参数：

类型	说明
uint8_t	要删除的子地图 ID (0…254)

Marvelmind robotics

10.4.24. 冻结子地图

冻结子地图。

函数名称：mm_freeze_submap
C 语言声明：bool mm_freeze_submap (uint8_t submapId);
适用的 API 版本：V1+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，子地图已冻结 false – 执行出错

参数：

类型	描述
uint8_t	要冻结的子地图 ID（0…254）

Marvelmind robotics

10.4.25. 解冻子地图

解冻子地图。

函数名称：mm_unfreeze_submap
C 语言声明：bool mm_unfreeze_submap (uint8_t submapId);
适用的 API 版本：V1+

需要许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，子地图已解冻 false – 执行出错

参数：

类型	描述
uint8_t	要解冻的子地图 ID（0...254）

Marvelmind robotics

10.4.26. 获取子地图设置

从 Modem 读取子地图设置。

函数名称：mm_get_submap_settings
C 语言声明：bool mm_get_submap_settings (uint8_t submapId , void *pdata);
适用的 API 版本：V1+

需要许可证：无

返回值：

类型	说明
bool	true – 函数成功执行，子地图设置已获取 false – 执行错误

参数：

类型	说明
uint8_t	子地图 ID (0...254)
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	说明
uint8_t	起始信标三边定位
uint8_t	起始信标组，信标 1
uint8_t	起始信标组，信标 2
uint8_t	初始信标组，信标 3
uint8_t	初始信标组，信标 4
bool	true = 启用 3D 导航
bool	true = 子地图仅用于 Z 坐标
bool	true = 手动限制距离 false = 自动限制距离
uint8_t	最大距离，单位米（用于手动限制距离）
int16_t	子地图 X 偏移，cm
int16_t	子地图 Y 偏移，cm
int16_t	子地图 Z 偏移，cm
uint16_t	子地图旋转，单位为百分之一度
int16_t	平面旋转四元数，W（四元数已归一化为 10000）
int16_t	平面旋转四元数，X
int16_t	平面旋转四元数，Y
int16_t	平面旋转四元数，Z
int16_t	服务区厚度，单位为 cm
int16_t	2D 模式下 Hedge 的高度
bool	true = 子地图已冻结
bool	true = 子地图已锁定
bool	true = 固定信标高于移动信标
bool	true = 子地图被镜像
4 字节	子地图中信标的地址列表（0 = 无）
8 字节	附近子地图的 ID 列表（255 = 无）
uint8_t	服务区多边形点的数量 (P)
P*4 字节	服务区多边形点结构列表（见下文）

Marvelmind robotics

服务区多边形点的结构：

类型	描述
int16_t	X，cm
int16_t	Y，cm

Marvelmind robotics

10.4.27. 设置子地图参数

将子地图参数写入 modem。

函数名称：mm_set_submap_settings
C 语言声明：bool mm_set_submap_settings (uint8_t submapId , void *pdata);
适用的 API 版本：V1+

需要许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行，子地图设置已更改 false – 执行出错

参数：

类型	说明
uint8_t	子地图 ID（0…254）
void *	指向待写入数据的指针（参见 'get submap settings' 函数）。

Marvelmind robotics

10.4.28. 冻结地图

冻结子地图。

函数名称：mm_freeze_map
C 语言声明：bool mm_freeze_map ();
适用的 API 版本：V4+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，地图已冻结 false – 执行出错

10.4.29. 解冻地图

冻结子地图。

函数名：mm_unfreeze_map
C 语言声明：bool mm_freeze_map ();
适用的 API 版本：V4+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，地图已解冻 false – 执行出错

Marvelmind robotics

10.4.30. 获取信标的超声波设置

从指定信标读取超声波设置。

函数名称：mm_get_ultrasound_settings
C 语言声明：bool mm_get_ultrasound_settings (uint8_t address , void *pdata);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功，已成功获取超声波设置 false – 执行出错

参数：

类型	描述
uint8_t	信标地址 (1…254)
void *	指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
uint16_t	超声波 TX 频率(不适用于仅 DSP RX 的 beacon)
uint8_t	TX 周期数(不适用于仅 DSP RX 的 beacon)
bool	true = RX 使用 AGC false = RX 使用手动增益
uint16_t	手动增益值(0…4000)
bool	true = 在正常模式下启用传感器 RX1
bool	true = 在正常模式下启用传感器 RX2
bool	true = 在正常模式下启用传感器 RX3
bool	true = 在正常模式下启用传感器 RX4
bool	true = 在正常模式下启用传感器 RX5
bool	true = 传感器 RX1 在冻结模式下启用
bool	true = 传感器 RX2 在冻结模式下启用
bool	true = 传感器 RX3 在冻结模式下启用
bool	true = 传感器 RX4 在冻结模式下启用
bool	true = 传感器 RX5 在冻结模式下启用
uint8_t	DSP RX 滤波器索引（仅适用于 DSP beacon） 0 = 19 kHz 1 = 25 kHz 2 = 31 kHz 3 = 37 kHz 4 = 45 kHz

10.4.31. 设置 beacon 的超声波参数

将超声波参数写入指定的 beacon。

函数名称：mm_set_ultrasound_settings
C 语言声明:bool mm_set_ultrasound_settings (uint8_t address, void *pdata);
适用的 API 版本:V1+

所需许可证:MMSW0005

返回值:

类型	描述
bool	true – 函数成功执行,超声波设置已更改 false – 执行出错

参数:

类型	描述
uint8_t	信标地址(1...254)
void *	指向待写入数据的指针(参见"get ultrasonic settings"函数)。

Marvelmind robotics

10.4.32. 擦除地图

擦除 Modem 中的地图 – 删除所有子地图(submap 0 除外),将 submap 0 重置为初始状态,从网络中移除所有已连接的信标。

函数名称:mm_erase_map
C 语言声明:bool mm_erase_map ();
适用的 API 版本：V1+

所需许可证：MMSW0005

返回值:

类型	描述
bool	true – 函数执行成功，地图已清除 false – 执行出错

参数：无

Marvelmind robotics

10.4.33. 将设备重置为默认设置

将设备重置为默认设置（无线电、超声波等）。

函数名称：mm_set_default_settings
C 语言声明：bool mm_set_default_settings (uint8_t address);
适用的 API 版本：V1+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，设备已重置为默认设置 false – 执行错误

参数：

类型	描述
uint8_t	设备地址（1...254） 255 – 将通过 USB 连接的设备重置为默认值

Marvelmind robotics

10.4.34. 将信标连接到坐标轴

平移地图，使所选信标位于坐标轴上。

函数名称：mm_beacons_to_axes
C 语言声明：bool mm_beacons_to_axes (uint8_t address_0, uint8_t address_x, uint8_t address_y);
可用的 API 版本：V2+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数成功执行，地图已平移 false – 执行错误

参数：

类型	描述
uint8_t	address_0 – 应位于中心位置的信标地址（X=0, Y=0）
uint8_t	address_x – 应位于 X 轴上的信标地址（Y= 0）
uint8_t	address_y – 应位于 Y 轴正方向的信标地址（Y>0）

Marvelmind robotics

10.4.35. 读取 modem 配置存储器的转储数据

读取 modem 配置存储器的转储数据。可用于保存 modem 的设置和已存储的地图。

函数名称：mm_read_flash_dump
C 语言声明：bool mm_read_flash_dump(uint32_t offset, uint32_t size, void *pdata);

适用的 API 版本：V3+
所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功，转储数据已读取 false – 执行错误

参数:

类型	说明
uint32_t	offset – 相对于配置存储器起始位置的偏移量,单位为字节
uint32_t	size – 要读取的数据大小,单位为字节
void *	pdata – 指向用户接收数据缓冲区的指针

10.4.36. 写入 modem 配置存储器的转储数据

将数据转储写入 modem 的配置存储器。可用于恢复 modem 的设置和地图。

函数名称:mm_write_flash_dump

C 语言声明:bool mm_write_flash_dump(uint32_t offset, uint32_t size, void *pdata);

适用的 API 版本:V3+

所需许可证:MMSW0005

返回值:

类型	说明
bool	true – 函数执行成功,转储已写入 false – 执行出错

参数:

类型	描述
uint32_t	offset – 相对于配置存储器起始位置的偏移量，单位为字节 为保证正确操作，偏移量应按 4096 字节页对齐（取值为 0、4096、8192 等）。
uint32_t	size – 要写入的数据大小，单位为字节
void *	pdata – 指向用户数据缓冲区的指针

注意：写入配置后，建议对 modem 执行软件复位（mm_reset_device(255)）以应用新设置，防止被覆盖。

Marvelmind robotics

10.4.37. 设备重启（软复位）

对指定设备执行软件复位。

函数名称：mm_reset_device

C 语言声明：bool mm_reset_device (uint8_t address);

适用 API 版本：V3+

所需许可：无

返回值：

类型	描述
bool	true – 函数执行成功，设备正在重置 false – 执行出错

参数：

类型	描述
uint8_t	设备地址（1…254） 255 – 对通过 USB 连接的设备进行软件重置

Marvelmind robotics

10.4.38. 从 Modem 读取空气温度设置

从 Modem 读取空气温度设置（以摄氏度为单位）。

函数名称：mm_get_air_temperature

C 语言声明：bool mm_get_air_temperature (void *pdata);

适用的 API 版本：V3+

需要许可证：无

返回值：

类型	描述
bool	true – 函数成功执行，返回温度值 false – 执行出错

通过 pdata 指针返回的数据结构：

类型	说明
int8_t	空气温度，摄氏度

Marvelmind robotics

10.4.39. 将空气温度设置写入 Modem

在 Modem 中设置空气温度（摄氏度）。

函数名称：mm_set_air_temperature

C 语言声明：bool mm_set_air_temperature (void *pdata);

适用的 API 版本：V3+

所需许可：无

返回值：

类型	说明
bool	true – 函数成功执行，温度已写入 false – 执行出错

用户应通过 pdata 指针提供的数据结构：

类型	描述
int8_t	空气温度，摄氏度

Marvelmind robotics

10.4.40. 获取信标实时播放器设置

读取信标的实时播放器设置。

函数名称：mm_get_realtime_player_settings

C 语言声明：bool mm_get_realtime_player_settings (uint8_t address, void *pdata);

适用的 API 版本：V6+

所需许可证：无

返回值：

类型	描述
bool	true – 函数成功执行 false – 执行出错

参数：

类型	描述
uint8_t	address - 信标地址 (1…254)
void *	pdata - 指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
bool	true = 实时播放器已启用
uint8_t	实时播放器待处理的前向采样数
uint8_t	实时播放器待处理的后向采样数
uint8_t	保留 (0)
uint8_t	保留 (0)

Marvelmind robotics

10.4.41. 设置信标实时播放器参数

为信标设置实时播放器参数。

函数名称：mm_set_realtime_player_settings

C 语言声明：bool mm_set_realtime_player_settings (uint8_t address, void *pdata);

适用的 API 版本：V6+

所需许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功 false – 执行错误

参数：

类型	描述
uint8_t	address - 信标地址 (1...254)
void *	pdata - 指向要写入数据的指针（参见"获取信标实时播放器设置"函数）

Marvelmind robotics

10.4.42. 获取地理参考设置

读取地理参考设置（Marvelmind 地图中点 (X=0 ,Y=0) 的地理位置）。

函数名称：mm_get_georeferencing_settings

C 中的声明：bool mm_get_georeferencing_settings (void *pdata);

适用的 API 版本：V6+

所需许可证：无

返回值：

类型	描述
bool	true – 函数成功执行 false – 执行错误

参数：

类型	描述
void *	pdata - 指向待填充数据的指针

通过指针返回的数据结构：

类型	描述
int32_t	纬度，x10-7 度
int32_t	经度，x10-7 度

Marvelmind robotics

10.4.43. 设置地理参考设置

设置地理参考设置（Marvelmind 地图中点 (X=0, Y=0) 的地理位置）。

函数名称：mm_set_georeferencing_settings

C 语言声明：bool mm_set_georeferencing_settings (void *pdata);

适用的 API 版本：V6+

所需许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行 false – 执行出错

参数：

类型	说明
void *	pdata - 指向要写入数据的指针（参见"获取地理参考设置"函数）

Marvelmind robotics

10.4.44. 获取位置更新模式

读取移动信标当前的位置更新模式。

函数名称：mm_get_update_position_mode

C 语言声明：bool mm_get_update_position_mode (void *pdata);

适用的 API 版本：V6+

所需许可证：无

返回值：

类型	说明
bool	true – 函数执行成功 false – 执行出错

参数：

类型	说明
void *	pdata - 指向待填充数据的指针

通过指针返回的数据结构：

类型	说明
uint8_t	移动信标位置更新模式： 0 – 自动更新位置（默认模式） 1 – 按用户请求在下一个更新周期更新位置 2 – 根据用户请求立即更新位置
7 字节	保留 (0)

Marvelmind robotics

10.4.45. 设置位置更新模式

设置移动信标的位置更新模式。

函数名称：mm_set_update_position_mode

C 语言声明：bool mm_set_update_position_mode (void *pdata);

适用的 API 版本：V6+

需要许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功 false – 执行出错

参数：

类型	描述
void *	pdata - 指向待写入数据的指针（参见函数"Get mode of updating positions"）

Marvelmind robotics

10.4.46. 更新位置命令

发送命令以更新移动信标的位置（如果更新模式不是自动的）。

函数名：mm_set_update_position_command

C 语言声明：bool mm_set_update_position_command (void *pdata);

可用的 API 版本：V6+

所需许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行 false – 执行出错

参数：

类型	说明
void *	pdata - 指向待写入数据的指针

指针指向的数据结构：

类型	说明
8 字节	保留 (0)

Marvelmind robotics

10.4.47. 设置信标的地理围栏报警状态

发送命令以设置信标报警引脚的报警状态（用于 Super-Beacon）。

引脚的报警状态可通过 Dashboard 设置中 "Interfaces" 部分的 "Alarm pin mode" 选项指定（需激活 MMSW0006 许可证）。

函数名称：mm_set_alarm_state

C 语言声明：bool mm_set_alarm_state (uint8_t address, void *pdata);

适用的 API 版本：V9+

所需许可证：MMSW0005、 MMSW0006

返回值：

类型	描述
bool	true – 函数执行成功 false – 执行出错

参数：

类型	描述
uint8_t	address - 信标地址 (1...254)
void *	pdata - 指向要写入数据的指针

指针所指数据的结构：

类型	描述
uint8_t	报警引脚模式： 0 – 引脚根据地理围栏状态自动控制 1 – 引脚手动控制 – 无报警状态 2 – 引脚手动控制 – 报警状态
uint8_t	地理围栏区域索引 – beacon 在报警状态下将输出的地理围栏区域编号
6 字节	保留 (0)

Marvelmind robotics

10.4.48. 发送通用用户负载数据

发送通用用户负载数据。如果 API 连接到 modem，数据将通过指定移动 beacon 的 UART/USB 端口传输。如果 API 连接到移动 beacon，数据将通过 modem 的 UART/USB 端口传输。接收到的数据可在远端通过接收 API 函数、Arduino 示例、ROS 及其他软件获取。

函数名称：mm_send_user_payload_data

C 语言声明：bool mm_send_user_payload_data (uint8_t address, void *pdata);

适用的 API 版本：V9+

需要许可证：MMSW0005

返回值：

类型	描述
bool	true – 函数执行成功 false – 执行错误

参数：

类型	描述
uint8_t	address - 信标地址（1...254），当 API 连接到 modem 时 当 API 连接到信标时不适用
void *	pdata - 指向待写入数据的指针

指针所指向的数据结构：

类型	描述
uint8_t	待传输数据的大小
256 字节	待传输的通用数据缓冲区

Marvelmind robotics

10.4.49. 获取通用用户有效载荷数据

接收由传输 API 函数、Arduino、ROS 或其他用户软件发送的通用用户有效载荷数据。如果 API 连接到 modem，此函数可以接收通过移动信标的 UART/USB 端口传输的数据。如果 API 连接到移动信标，此函数可以接收通过 modem 传输的数据。

函数名称：mm_get_user_payload_data

C 语言声明：bool mm_get_user_payload_data (void *pdata);

可用的 API 版本：V9+

需要许可证：MMSW0005

返回值：

类型	说明
bool	true – 函数成功执行 false – 执行出错

参数：

类型	说明
void *	pdata - 指向待接收数据的指针

通过指针接收的数据结构：

类型	描述
uint8_t	address – 信标的地址
int64_t	数据传输的时间戳 – 从 1970 年 1 月 1 日起的毫秒数(Unix 时间)
uint8_t	待传输数据的大小
256 字节	接收数据的缓冲区

Marvelmind robotics

10.4.50. 发送手动距离测量命令

发送从指定信标到系统中其他信标的距离测量命令。在当前版本的软件中,该功能在IA(反向架构)中受支持。

函数名称:mm_send_distances_measurement_command

C 中的声明:bool mm_send_distances_measurement_command (void *pdata);

适用的 API 版本:V9+

所需许可证:MMSW0011

返回值:

类型	描述
bool	true – 函数成功执行 false – 执行错误

参数:

类型	描述
void *	pdata - 指向要发送数据的指针

指针所指数据的结构:

类型	描述
uint8_t	模式: 0 – 自动 1 – 手动（通过此命令）
uint8_t	信标地址
uint32_t	最大测量距离，mm
8 字节	保留

Marvelmind robotics

10.4.51. 从 Modem 获取流数据

以先前描述的格式读取 modem 的流数据。

函数名称：mm_get_stream_data

C 语言声明：bool mm_get_stream_data (void *pdata);

适用 API 版本：V9+

所需许可证：无

返回值：

类型	描述
bool	true – 函数执行成功 false – 执行出错

参数：

类型	描述
void *	pdata - 指向要发送数据的指针

指针所指数据的结构：

类型	描述
uint8_t	此回复中的数据流记录数量（0...16）
138*16 字节	16 条数据流记录，每条 138 字节（见下文）
8 字节	保留

数据流记录的结构：

类型	说明
uint8_t	记录大小，字节
uint8_t	记录类型。与 Dashboard 日志文件中的 'line type' 取值相同。 例如，41 表示 Marvelmind 协议数据
8 字节	保留
128 字节	数据流记录数据

Marvelmind robotics

10.4.52. 检查设备类型是否为 Modem

检查指定的设备类型是否对应于 Modem。

函数名称：mm_device_is_modem
C 语言声明：bool mm_device_is_modem (uint8_t deviceType);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	说明
bool	true – 指定的设备类型对应于 Modem

参数：

类型	说明
uint8_t	要检查的设备类型

Marvelmind robotics

10.4.53. 检查设备类型是否为固定信标

检查指定的设备类型是否对应于固定信标。

函数名称：mm_device_is_beacon
C 语言声明：bool mm_device_is_beacon (uint8_t deviceType);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 指定的设备类型对应静态信标

参数：

类型	描述
uint8_t	要检查的设备类型

Marvelmind robotics

10.4.54. 检查设备类型是否为 hedgehog

检查指定的设备类型是否对应 hedgehog。

函数名称：mm_device_is_hedgehog
C 语言声明：bool mm_device_is_hedgehog (uint8_t deviceType);
适用的 API 版本：V1+

所需许可证：无

返回值：

类型	描述
bool	true – 指定的设备类型对应 hedgehog

参数:

类型	说明
uint8_t	待检查的设备类型

Marvelmind robotics

10.5. Marvelmind API 的 C 示例说明

C 示例用于测试 Marvelmind API，也可作为构建用户应用程序的基础。

该 C 示例为控制台应用程序。已在以下平台上进行测试：

- CPU：Intel Core 2 Duo，操作系统：MS Windows XP；
- CPU：Intel Core i5，操作系统：Linux Ubuntu 16.04；
- Raspberry Pi 3 Model B+，操作系统：Raspbian (2018-11-13-raspbian-stretch-full)

在 Windows 平台上，该示例使用 CodeBlocks IDE 构建，因此示例中包含 CodeBlocks 项目文件。

在 Linux 平台上，该示例使用 make 工具构建，因此示例中包含相应的 makefile。

该示例包含以下模块：

文件名	说明
main.c	包含 main() 函数的模块。调用示例函数并实现简单的命令行界面。
marvelmind_example.c marvelmind_example.h	marvelmindStart() – 示例的初始化 marvelmindFinish() – 在完成 API 工作后调用 marvelmindCycle() – 由主循环频繁调用 此外，该模块还包含若干用于处理用户输入命令的函数。
marvelmind_api.c marvelmind_api.h	marvelmindAPILoad() – 加载 API 库 marvelmindAPIFree() – 释放 API 库所占用的内存 与 API 库通信的所有函数。
marvelmind_devices.c marvelmind_devices.h	支持通过调用"get devices list"命令从 Modem 获取的信标列表。每个信标包含其位置数据以及与其他信标的距离数据。
marvelmind_pos.c marvelmind_pos.h	读取最新的位置数据和最新的原始距离数据。并在设备列表中更新这些数据。
marvelmind_utils.c marvelmind_utils.h	其他模块使用的一些辅助函数。

示例的工作原理：

1. 尝试打开串口直到成功
2. 端口打开后，程序会读取通过 USB 连接的设备版本。如果是 Modem，程序将继续执行下一步
3. 连接到 Modem 后，程序以 1 Hz 的频率读取设备列表。该设备列表与当前存储在 marvelmind_devices.c 模块中的列表进行比较，如检测到任何更改，则更新 marvelmind_devices.c 中的列表。所有更改都会在控制台中打印
4. 连接到 Modem 后，程序以 20 Hz 的频率读取最新位置数据。如果设置了新原始距离数据标志，程序将读取最新原始距离。程序将位置和距离与 marvelmind_devices.c 中设备列表的数据进行比较，如有变化则更新数据。所有更改的数据都会在控制台中打印
5. 如果程序连续 10 次无法获取最新位置数据，则关闭端口并返回步骤 1——尝试再次打开端口。当 Modem 断开 USB 连接后又重新连接时，需要重新打开端口
6. 如果用户按下"空格"键，程序将显示"Enter command:"消息并等待用户命令。大多数 API 函数通过用户命令调用，详见下文

用户命令：

如果用户在程序运行时按下"空格"键，程序将显示消息"Enter command:"。用户应在键盘上输入命令并按回车键。

下表包含所有用户命令的格式：

命令组	说明

Marvelmind robotics

API 版本	命令格式: **version** 操作: 打印 API 库的版本
退出程序	命令格式: **quit** 操作: 结束程序执行
休眠/唤醒	命令格式: **wake <address>** 操作: 执行唤醒命令。 示例: **wake 5 - 发送命令以唤醒设备 5** **wake 0 - 发送命令以唤醒所有设备**
	命令格式: **sleep <address>** 操作: 执行进入休眠命令。 示例: **sleep 5 - 使设备 5 进入休眠** **sleep 0 - 使所有设备进入休眠**
默认	命令格式: **default <address>** 操作: 执行恢复默认设置命令。 示例: **default 5 - 为设备 5 设置默认设置**
读取遥测数据	命令格式: **tele <address>** 操作: 读取并打印信标的遥测数据。 示例: **tele 5 - 读取并打印信标 5 的遥测数据**
子地图命令	命令格式: **submap add <submapId>** 操作: 执行命令以添加指定子地图 ID 的子地图。 示例: **submap add 1 - 添加子地图 1**
	命令格式: **submap delete <submapId>** 操作: 执行命令以删除指定子地图 ID 的子地图。 示例: **submap delete 1 - 删除子地图 1**
	命令格式: **submap freeze <submapId>** 操作: 执行命令以冻结指定子地图 ID 的子地图。 示例:

Marvelmind robotics

	submap freeze 0 - **冻结子地图 0**
	命令格式: **submap unfreeze <submapId>** 操作: 执行命令以解冻指定子地图 ID 的子地图。 示例: **submap unfreeze 0** - **解冻子地图 0**
	命令格式: **submap get <submapId>** 操作: 执行命令以获取指定子地图 ID 的子地图设置。 示例: **submap get 0** - **获取并打印子地图 0 的设置**
	命令格式: **submap testset <submapId>** 操作: 执行命令以设置指定 submap ID 的子地图设置。程序将写入一些预定义的设置用于测试该命令;请参见示例代码。 示例: **submap testset 0** - **修改子地图 0 的设置**
地图命令	命令格式: **map erase** 操作: 执行擦除地图命令。 示例: **map erase** - **擦除 modem 中的地图**
	命令格式: **map freeze** 操作: 执行冻结地图命令。 示例: **map freeze** - **冻结地图**
	命令格式: **map unfreeze** 操作: 执行解冻地图命令。 示例: **map unfreeze** - **解冻地图**
更新频率命令	命令格式: **rate get** 操作: 执行读取更新频率设置命令。 示例: **rate get** - **读取并打印更新频率设置**
	命令格式:

	rate set <value> 操作： 执行更改更新频率设置命令。数值单位为 Hz 示例： **rate set 0.5 - 设置更新频率为 0.5 Hz**
超声波命令	命令格式： **usound get <address>** 操作： 对指定信标执行读取超声波设置。 示例： **usound get 5 - 读取并打印信标 5 的超声波设置**
	命令格式： **usound testset <address>** 操作： 对指定信标执行写入超声波设置。程序会写入一些预定义的设置以测试该命令；请参阅示例代码。 示例： **usound testset 5 - 修改信标 5 的超声波设置**
将信标连接到坐标轴命令	命令格式： **axes <address_0> <address_x> <address_y>** 操作： 执行将信标连接到坐标轴命令。 示例： **axes 3 4 5 - 将信标 3 设置为 X=0, Y=0；信标 4 沿 X 轴 (Y=0)；信标 5 位于 X 轴之上 (Y>0)**
从 Modem 读取配置内存转储	命令格式： **read_dump <offset> <size>** 操作： 执行读取 modem 配置存储器转储命令。 示例： **read_dump 0 1000 - 从配置存储器起始位置读取前 1000 字节**
向 modem 写入配置存储器测试转储	命令格式： **write_dump_test <offset> <size>** 操作： 执行向 modem 配置存储器写入转储命令。 示例： **write_dump_test 0 1000 - 使用测试模式填充配置存储器起始位置的前 1000 字节**

Marvelmind robotics

设备软件复位	命令格式： **reset <address>** 操作： 执行软件复位命令。 示例： **reset 255 - 对通过 USB 连接的设备执行软件复位**
空气温度命令	命令格式： **temperature get** 操作： 从 Modem 读取空气温度设置 示例： **temperature get 读取并打印超声波空气温度设置**
	命令格式： **temperature set <value>** 操作： 向 Modem 写入空气温度设置 示例： **temperature set 30 将空气温度设置为 30 摄氏度**
设置 Beacon 位置	命令格式： **setloc <address> <X > <Y > <Z>** 操作： 执行设置 Beacon 位置命令。X、Y、Z 为坐标，单位为米。 示例： **setloc 12 1.51 3.45 2.0 - 将 Beacon 12 的位置设置为 X= 1.51 m, Y= 3.45 m, Z= 2.0 m**
设置 Beacon 之间的距离	命令格式： **setdist <address1> <address2 > <distance>** 操作： 执行设置信标之间距离的命令。Address1 和 address2 是信标的地址。Distance 是以米为单位的距离。 示例： **setdist 12 13 16.5 - 将信标 12 和 13 之间的距离设置为 16.5 米**
高度命令	命令格式： **height_h get <address>** 操作： 执行获取 hedgehog 高度的命令。Address 是 hedgehog 的地址。 示例： **height_h get 15 - 读取并打印 hedgehog 15 的高度**
	命令格式： **height_h set <address> <height>** 操作：

Marvelmind robotics

	执行设置 hedgehog 高度的命令。Address 是 hedgehog 的地址。Height 以米为单位。 示例： **height_h set 15 2.5 - 将 hedgehog 15 的高度设置为 2.5 米**
	命令格式： **height_b get <address> <submap_id>** 操作： 执行获取固定信标高度命令。Address 为信标的地址。Submap_id 为信标所属子地图的 ID。 示例： **height_b get 12 0 - 读取并打印子地图 0 中固定信标 12 的高度。**
	命令格式： **height_b set <address> <submap_id> <height>** 操作： 执行设置固定信标高度命令。Address 为 hedgehog 的地址。Submap_id 为信标所属子地图的 ID。高度单位为米。 示例： **height_b set 12 0 5.1 - 将子地图 0 中信标 12 的高度设置为 5.1 米**
实时播放器命令	命令格式： **rtp get <address>** 操作： 执行获取实时播放器设置命令。Address 为信标的地址。 示例： **rtp get 15 - 读取并打印信标 15 的实时播放器设置**
	命令格式： **rtp testset <address>** 操作： 执行设置实时播放器设置命令。Address 为信标地址。程序会写入一些预定义的设置以测试该命令；请参阅示例代码。 示例： **rtp testset 15 - 为 15 号信标设置测试实时播放器设置**
地理参考命令	命令格式： **georef get** 操作： 执行获取地理参考设置命令。 示例： **georef get - 读取并打印地理参考设置**
	命令格式： **georef set <latitude> <longitude>** 操作： 执行设置地理参考设置命令。 示例： **georef set 10 20 – 写入地理参考：纬度 10 度，经度 20 度**
更新模式命令	命令格式： **update_mode get** 操作：

Marvelmind robotics

	执行获取位置更新模式命令。 示例： **update_mode get - 读取并打印位置更新模式**
	命令格式： **update_mode set <mode>** 操作： 执行设置位置更新模式命令。 示例： **update_mode set 0 - 设置为位置自动更新模式**
	命令格式： **update** 操作： 执行更新位置命令。 示例： **update - 根据当前模式更新移动信标的位置**
设置地理围栏警报状态	命令格式： **alarm <address> <mode> <zone>** 操作： 执行设置地理围栏报警状态命令。 示例： **alarm 10 2 5 - 在编号为 10 的 beacon 上设置地理围栏编号为 5 的报警信号**
用户载荷命令	命令格式： **payload read <address>** 操作： 执行获取用户载荷数据命令。 示例： **payload read - 从任意 beacon/modem 读取用户载荷** 示例： **payload read 10 - 从编号为 10 的 beacon 读取用户载荷**
	命令格式： **payload write <address>** 操作： 执行发送用户载荷数据命令。 示例： **payload write 10 - 将测试载荷数据写入编号为 10 的 beacon** 测试模式为从 100 开始的 40 字节数据：100,101,..., 139
手动距离测量命令	命令格式： **distance <manual/auto> <address> <max distance>** 操作： 执行手动距离测量命令。 示例： **distance manual 10 5 – 测量信标 10 到其他信标的距离，最大距离 5 米** 示例： **distance manual 10 – 测量信标 10 到其他信标的距离，最大距离 30 米（默认）** 示例： **distance auto - 返回自动距离测量模式**

Marvelmind robotics

10.6. 设备类型

以下是特定设备的"设备类型 ID"值列表：

设备类型 ID	设备描述
22	Beacon HW V4.5
23	Beacon HW V4.5（hedgehog 模式）
24	Modem HW V4.9
30	Beacon HW V4.9
31	Beacon HW V4.9（hedgehog 模式）
32	Beacon Mini-RX
36	Beacon Mini-TX
37	Beacon-TX-IP67
41	Beacon industrial-RX
42	Super-Beacon
43	Super-Beacon（hedgehog 模式）
44	Industrial Super-Beacon
45	Industrial Super-Beacon（hedgehog 模式）
46	Super-Modem
48	Modem HW V5.1

您可以通过设备列表和读取设备版本命令获取设备类型 ID。

11. 从/向用户设备发送用户数据

Marvelmind 支持通过 Marvelmind 系统传输用户数据的多种方式：

- 通过 Modem 的 UART 或 USB 发送数据，并通过移动信标的 UART 或 USB 接收
- 通过移动信标的 UART 或 USB 发送数据，并通过 Modem 的 UART 或 USB 接收。

Super-Modem 还支持通过 UDP 发送和接收用户数据。

数据传输协议已在本文档前面的章节中进行了说明：

- 向用户设备发送数据以及从用户设备接收数据的协议；
- 用于发送和接收数据的 API 函数。

Marvelmind 提供了多种通信软件示例：

示例	Arduino（UART）	PC / Raspberry Pi（USB）				PC / Raspberry Pi（UDP Super-Modem）
		API	C	Python	ROS/ROS2	C 示例
用户设备 beacon←	+	+	+	+	+	不适用
Modem 用户设备→	+	+	+	+	+	+
用户设备 beacon→	+	+	-	-	+	不适用
Modem 用户设备←	+	+	-	-	+	+

示例完整列表：

- 用于发送和接收用户数据的 Arduino 示例位于 Marvelmind 软件包中的 '01_Common_Indoor_positioning_SW/ 06_Examples/ arduino' 文件夹。'hedgehog_sample_uart_user_data_receive_v2' 用于接收用户数据，'hedgehog_sample_uart_user_data_send_v2' 用于发送用户数据。
- API 通信示例位于 Marvelmind 软件包中的 '01_Common_Indoor_positioning_SW/ 05_API/example_source'（源代码）和 '01_Common_Indoor_positioning_SW/ 05_API/example_bin_win32'（Windows 二进制文件）文件夹。可按照本文档所述调用数据发送或接收功能。
- 用于接收流式数据的 C 示例位于 Marvelmind 软件包中的 '01_Common_Indoor_positioning_SW/ 06_Examples/ c' 文件夹。该示例也可在 GitHub 的代码仓库中获取。此示例仅打印从移动 beacon 或 modem 接收到的所有数据，包括用户数据。
- 用于接收流数据的 Python 示例位于 Marvelmind 软件包的 '01_Common_Indoor_positioning_SW/ 06_Examples/ python' 文件夹中。该示例也可在 GitHub 仓库中获取。此示例仅打印从移动信标或 Modem 接收到的所有数据，包括用户数据。
- 用于接收流数据的 ROS 包示例位于 Marvelmind 软件包的 '01_Common_Indoor_positioning_SW/ 06_Examples/ ROS' 文件夹中。该软件包也可在仓库中获取。该 ROS 包允许通过 API 接收用户数据和发送用户数据。详情请参阅文档。

- 用于接收流数据的 ROS-2 包示例位于 Marvelmind 软件包的 '01_Common_Indoor_positioning_SW/ 06_Examples/ ROS2' 文件夹中。该软件包也可在仓库中获取。该 ROS 包允许通过 API 接收用户数据和发送用户数据。详情请参阅文档。
- 用于通过 UDP 接收数据的 C 示例位于 Marvelmind 软件包的 '01_Common_Indoor_positioning_SW/ 06_Examples/ c' 文件夹中。此示例仅打印通过 UDP 从 Super-Modem 或 Dashboard 接收到的所有数据，包括用户数据。如果使用 API 通过 UDP（而非 USB）连接到 Super-Modem，则可通过 API 经由 UDP 发送用户数据。

12. 联系方式

如需其他支持，请将您的问题发送至 info@marvelmind.com

附录 1. CRC-16 计算

校验和采用 CRC-16。N 字节帧的最后两个字节填充 CRC-16，应用于帧的前 (N-2) 个字节。要校验数据，您可以对整个 N 字节帧应用 CRC-16，结果值应为零。

以下是该算法的 'C' 语言实现：

```
typedef uint16_t ModbusCrc;

typedef union {
	uint16_t w;
	struct{
		uint8_t lo;
		uint8_t hi;
	} b;
	uint8_t bs[2];
} Bytes;

static ModbusCrc modbusCalcCrc(const void *buf, uint16_t length)
{
	uint8_t *arr = (uint8_t *)buf;
	Bytes crc;

	crc.w = 0xffff;

	while(length--){
		char i;
		bool odd;

		crc.b.lo ^= *arr++;
		for(i = 0; i< 8; i++){
			odd = crc.w& 0x01;
			crc.w>>= 1;
			if (odd)
				crc.w ^= 0xa001;
		}
	}
	return (ModbusCrc) crc.w;
}
```


Marvelmind robotics

附录 2. Modem 错误回复格式

错误帧格式（从 Modem 到主机）

偏移量	大小（字节）	类型	说明	值
0	1	uint8_t	Modem 地址	0xff
1	1	uint8_t	数据包类型	
2	1	uint8_t	错误代码	
3	2	uint16_t	CRC-16（参见附录 1）	

错误数据包的类型是请求帧的数据包类型加上最高位。例如，如果请求的数据包类型为 0x03，则错误数据包的类型将为 0x83。

错误代码可能为以下之一：

1 – 请求中的数据包类型未知
2 – 请求中的数据代码未知
3 – 请求的数据字段错误
6 – 设备忙
10 – 来自远程设备的错误消息
11 – 远程设备回复超时

Marvelmind robotics