Zephyr
technology

BioHarness Bluetooth Comms Link Specification

Document History

Version	Date	Description
1.0	26th May 2010	Extracted from General Comms Link v1.66
1.1	31st May 2010	Updated after review.
1.2	16th Mar 2011	Periodic update rate of Accelerometer packet corrected in 0x1E message. Removed Blood Pressure & SpO2 and removed reference to canine monitoring from description of General Data Packet in message 0x20 and from Appendix B. Added note describing change from ACK to ETX for messages 0x20, 0x21, 0x22, 0x24 & 0x25. Also added note about change to 1000ms update rate for General Packet. Added sections for new messages: Extended Data Packet (0x28), Accelerometer 100mg Data Packet (0x2A), Set Algorithm Config (0xB6), Get Algorithm Config (0xB7), Set Extended Data Packet Transmit State (0xB8), Set BioHarness User Config Item (0xB9) and Set Accelerometer 100mg Data Packet Transmit State (0xBC). Updated Set BioHarness User Config (0xA6) and Get BioHarness User Config (0xA7) to include configuration of logging format. Changed posture range in all messages that include posture from +/-90 to +/-180. Extended Accelerometer / Activity range from 3.3 to 16g. Corrected Breathing Wave Amp range & units in Appendix B.
1.3	21st Nov 2011	Update for BioH 3.0

Document Notes

All numbers in this document are written in decimal, except hexadecimal numbers which are prefixed by '0x'. For example 5436 is decimal, while 0x5436 is hexadecimal.

This document describes messages applying to BioHarness Bluetooth versions 2.0 and 3.0.

As there are hardware differences between the two devices, not all messages are applicable to both devices. Specificity of messages is indicated in Appendix A – Overview of Messages at the end of this document.

v2.0

v3.0

Zephyr technology

DOCUMENT NOTES 3
1. REFERENCES 8
2. ABBREVIATIONS 8
3. INTRODUCTION 8
4. COMMUNICATIONS OVERVIEW 9
4.1. Basic Message Format 9
4.1.1. STX 9
4.1.2. Msg ID 9
4.1.3. DLC 9
4.1.4. Data Payload 10
4.1.5. CRC 10
4.1.6. ETX 11
4.1.7. ACK/NAK 11
4.2. Reception Timeout 11
4.3. Transmission Mediums 11
4.3.1. Direct Links 11
4.3.2. Wireless Links 12
5. STANDARD MESSAGES 13
5.1. MSG:0x01 - Read Logging Data 14
5.1.1. Request Message 14
5.1.2. Response Message 15
5.2. MSG:0x02 - Delete Log File 16
5.2.1. Request Message 16
5.2.2. Response Message 16
5.3. MSG:0x07 - Set RTC Date/Time 17
5.3.1. Request Message 17
5.3.2. Response Message 17
5.4. MSG:0x08 - Get RTC Date/Time 18
5.4.1. Request Message 18
5.4.2. Response Message 18
5.5. MSG:0x09 - Get Boot Software Version 19
5.5.1. Request Message 19
5.5.2. Response Message 19
5.6. MSG:0x0A - Get Application Software Version 20
5.6.1. Request Message 20
5.6.2. Response Message 20
5.7. MSG:0x0B - Get Serial Number 21
5.7.1. Request Message 21
5.7.2. Response Message 21
5.8. MSG:0x0C - Get Hardware Part Number 22
5.8.1. Request Message 22
5.8.2. Response Message 22
5.9. MSG:0x0D - Get Bootloader Part Number 23
5.9.1. Request Message 23
5.9.2. Response Message 23
5.10. MSG:0x0E - Get Application Part Number 24
5.10.1. Request Message 24
5.10.2. Response Message 24
5.11. MSG:0x10 - Set Network ID 25
5.11.1. Request Message 25
5.11.2. Response Message 25

5.12. MSG:0x11 - Get Network ID 26
5.12.1. Request Message 26
5.12.2. Response Message 26
5.13. MSG:0x12 - Get Unit MAC Address 27
5.13.1. Request Message 27
5.13.2. Response Message 27
5.14. MSG:0x14 - Set General Data Packet Transmit State 28
5.14.1. Request Message 28
5.14.2. Response Message 28
5.15. MSG:0x15 - Set Breathing Waveform Packet Transmit State 29
5.15.1. Request Message 29
5.15.2. Response Message 29
5.16. MSG:0x16 - Set ECG Waveform Packet Transmit State 30
5.16.1. Request Message 30
5.16.2. Response Message 30
5.17. MSG:0x17 - Get Unit Bluetooth Friendly Name 31
5.17.1. Request Message 31
5.17.2. Response Message 31
5.18. MSG:0x19 - Set R to R Data Packet Transmit State 32
5.18.1. Request Message 32
5.18.2. Response Message 32
5.19. MSG:0x1E - Set Accelerometer Packet Transmit State 33
5.19.1. Request Message 33
5.19.2. Response Message 33
5.20. MSG:0x9B - Set ROG Settings 34
5.20.1. Request Message 34
5.20.2. Response Message 34
5.21. MSG:0x9C - Get ROG Settings 35
5.21.1. Request Message 35
5.21.2. Response Message 35
5.22. MSG:0xA2 - Set Bluetooth User Config 36
5.22.1. Request Message 36
5.22.2. Response Message 36
5.23. MSG:0xA3 - Get Bluetooth User Config 37
5.23.1. Request Message 37
5.23.2. Response Message 37
5.24. MSG:0XA4 - Set BT Link Config 38
5.24.1. Request Message 38
5.24.2. Response Message 38
5.25. MSG:0xA5 - Get BT Link Config 39
5.25.1. Request Message 39
5.25.2. Response Message 39
5.26. MSG:0xA6 - Set BioHarness User Config 40
5.26.1. Request Message 40
5.26.2. Response Message 41
5.27. MSG 0xA7 - Get BioHarness User Config 42
5.27.1. Request Message 42
5.27.2. Response Message 42
5.28. MSG:0xAC – Get Battery Status 43
5.28.1. Request Message 43
5.28.2. Response Message 43
5.29. MSG:0x1F – Reboot Unit 44
5.29.1. Request Message 44
5.29.2. Response Message 44

5.30. MSG:0xB0 – Bluetooth Peripheral Message 45
5.30.1. Request Message 45
5.30.2. Response Message 46
5.31. MSG:0xB3 – Reset Configuration 47
5.31.1. Request Message 47
5.31.2. Response Message 47
5.32. MSG:0xB4 - Set Accelerometer Axis Mapping 48
5.32.1. Request Message 48
5.32.2. Response Message 48
5.33. MSG 0xB5 - Get Accelerometer Axis Mapping 49
5.33.1. Request Message 49
5.33.2. Response Message 49
5.34. MSG:0xB6 - Set Algorithm Config 50
5.34.1. Request Message 50
5.34.2. Response Message 51
5.35. MSG 0xB7 - Get Algorithm Config 52
5.35.1. Request Message 52
5.35.2. Response Message 52
5.36. MSG:0xB8 - Set Extended Data Packet Transmit State 53
5.36.1. Request Message 53
5.36.2. Response Message 53
5.37. MSG:0xB9 - Set BioHarness User Config Item 54
5.37.1. Request Message 54
5.37.2. Response Message 54
5.38. MSG:0xBC - Set Accelerometer 100mg Packet Transmit State 55
5.38.1. Request Message 55
5.38.2. Response Message 55
5.39. MSG:0xBD - Set Summary Data Packet Update Rate 56
5.39.1. Request Message 56
5.39.2. Response Message 56
5.40. MSG:0xBE - Set Subject Info Settings 57
5.40.1. Request Message 57
5.40.2. Response Message 60
5.41. MSG:0xBF - Get Subject Info Settings 61
5.41.1. Request Message 61
5.41.2. Response Message 61
5.42. MSG:0xD0 - Set Remote MAC Address & PIN 62
5.42.1. Request Message 62
5.42.2. Response Message 62
5.43. MSG:0xD1 - Get Remote MAC Address & PIN 63
5.43.1. Request Message 63
5.43.2. Response Message 63
5.44. MSG:0xD4 - Get Remote Device Description 64
5.44.1. Request Message 64
5.44.2. Response Message 64
5.45. MSG:0xD5 - Get Supported Log Formats 65
5.45.1. Request Message 65
5.45.2. Response Message 65
6. PERIODIC DATA PACKETS 66
6.1. MSG:0x20 - General Data Packet 66
6.2. MSG:0x21 - Breathing Waveform Packet 70
6.2.1. Packing Format 70
6.3. MSG:0x22 - ECG Waveform Packet 71
6.3.1. Packing Format 71

6.4. MSG:0x23 - Lifesign 72
6.5. MSG:0x24 - R to R Packet 73
6.5.1. Packing Format 73
6.6. MSG:0x25 - Accelerometer Packet 74
6.6.1. Packing Format 75
6.7. MSG:0x27 – Bluetooth Device Data Packet 76
6.8. MSG:0x28 – Extended Data Packet 77
6.9. MSG:0x2A – Accelerometer 100mg Packet 79
6.1. MSG:0x2B - Summary Data Packet 80
6.1.1. Version 2 80
6.2. MSG:0x2C – Event Packet 85
7. FIXED MESSAGE TYPES 86
7.1. NAK Response 86
8. APPENDIX A – OVERVIEW OF MESSAGES 87

1. References

Ref #	ID	Description
[1]		Event Messaging System

2. Abbreviations

Abbreviation	Description
ACK	**ACK**nowlege
ASCII	**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange
BPM	**B**eats **P**er **M**inute
CRC	**C**yclic **R**edundancy **C**heck
DLC	**D**ata **L**ength **C**ode
ECG	**E**lectro**C**ardio **G**raph
ETX	**E**nd of **T**ext
ID	**Id**entifier
LS	**L**east **S**ignificant
MAC	**M**edia **A**ccess **C**ontrol
MS	**M**ost **S**ignificant
NAK	**N**egative **AC**K**n**owledge
PC	**P**ersonal **C**omputer
RF	**R**adio **F**requency
STX	**S**tart of **T**ext
SPP	**Serial Port Profile**
UART	**U**niversal **A**synchronous **R**eceiver **T**ransmitter
VCP	**Virtual Com Port**

3. Introduction

This document specifies the protocol used to enable communications between a Zephyr Bluetooth BioHarness and other (remote) units. The protocol enables communications to a PC via USB VCP or via Bluetooth SPP. The protocol is half-duplex and UART-based in nature, relying on a simple request/response format. The protocol is defined, together with the specification of each message and describes the data used within each packet.

4. Communications Overview

The serial communication is request/response based, with the local or remote unit acting as master in the data transfer. The delays between the request and response messages are dependent on the message type; some can reply instantly e.g. “get version”, whereas others take longer e.g. “delete log file”; the delay essentially depends on the work the unit is requested to perform. The protocol only accommodates a point-to-point communications transfer, with no addressing used. The unit replies to the request message with one of two basic message types. If the request message is found to be valid, the response is in the form of an ACK with required data included; if found to be invalid, the response is in the form of a NAK message and no data is inserted into the message.
There is an exception to the rule where ‘periodic’ messages are transferred between units. In this case, no responses are required.

4.1. Basic Message Format

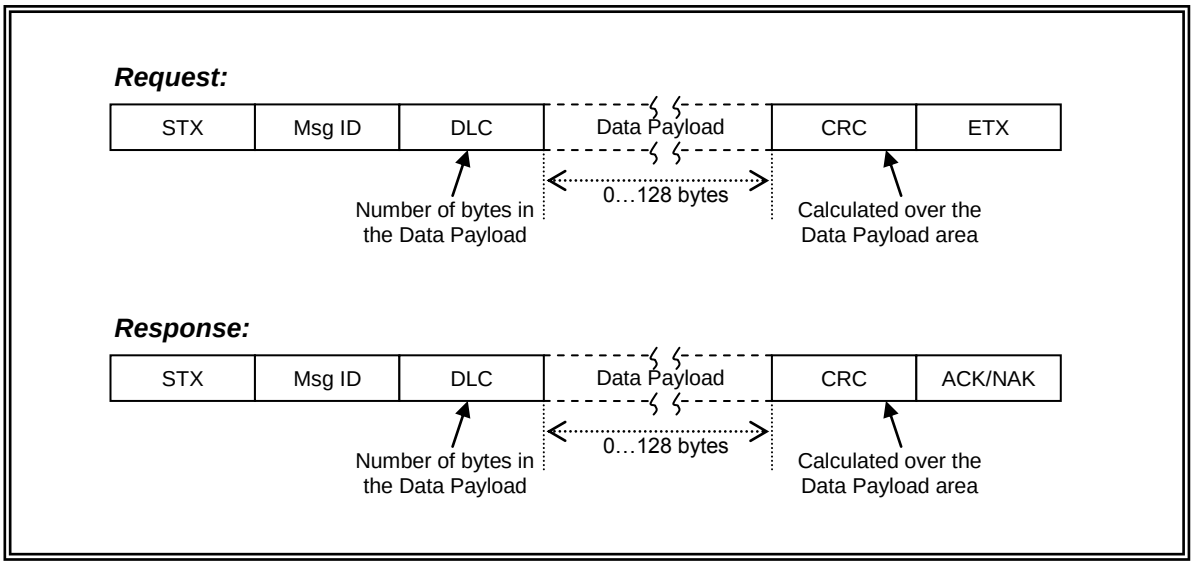

Figure 4-1 ***Basic Message Format***

4.1.1. STX

The STX field (Start of Text) is a standard ASCII control character (0x02) and denotes the start of the message. Although the protocol does not guarantee this value will not appear again within a message, it gives some delimiting to the message and therefore a start character to search for when receiving data.

4.1.2. Msg ID

The Message ID uniquely identifies each message and is in binary format. A response message uses the same Message ID as the corresponding request message.

4.1.3. DLC

The Data Length Code is used to specify the number of bytes within the data payload field of the message. Valid values range between zero and 128 (inclusive).

4.1.4. Data Payload

This field contains the actual data sent between the local and remote units and can contain anywhere between zero and 128 bytes of data. The number of bytes in this field is dictated by the DLC field.

4.1.5. CRC

An 8-bit CRC is used and has a polynomial of 0x8C (CRC-8), with the accumulator being initialised to zero before the CRC calculation begins. The following source code indicates the required implementation:

crc: The current CRC.
Ch: Value to add to the CRC calculation.

```
Void crc8PushByte(uint8_t *crc, uint8_t ch)
{
   uint8_t i;

   *crc = *crc ^ ch;
   for (i=0; i<8; i++)
   {
      if (*crc & 1)
      {
         *crc = (*crc >> 1) ^0x8C;
      }
      else
      {
         *crc = (*crc >> 1);
      }
   }
}
```

pcrc: Pointer to running CRC to update (set this to something before first call).
Block: Pointer to the.block of data to push through.
Count: The number of bytes to push.
Return: The computed CRC (result also updated in pcrc if that is non-NULL).

```
Uint8_t crc8PushBlock(uint8_t *pcrc, uint8_t *block, uint16_t count)
{
   uint8_t crc = pcrc ? *pcrc : 0;

   for (; count>0; --count,block++)
   {
      crc8PushByte(&crc, *block);
   }
   if (pcrc) *pcrc = crc;
   return crc;
}
```

For example, if the CRC has to be calculated over a block of 10 bytes of data (starting from the start of the buffer) the following function call could be used:

```
crc = crc8PushBlock( NULL, &dataBuffer[0], 10 );
```

4.1.6. ETX

The ETX field (End of Text) is a standard ASCII control character (0x03) and denotes the end of the message. Note that although this is the last field within the request message, the response message has no ETX value, instead using an ACK/NAK in its place.

4.1.7. ACK/NAK

This field is only present in the response message and indicates whether the unit found errors within the request message or couldn't perform the requested operation. If all data within the request was found to be valid and the unit could perform the requested operation, the response will have an ACK appended (standard ASCII control character, value = 0x06); otherwise a NAK is appended (standard ASCII control character, value = 0x15) – see section "Fixed Message Types" for more details on the NAK packet.

4.2. Reception Timeout

Although a general reception timeout is not utilized within the link processing, an inter-byte timeout is required (for direct links, e.g. to a PC), with the timeout period of 100ms being adequate. If there has been no data received within this period, reception within the unit should be reset, forcing a search for the beginning of a message again (the STX character). Although the response sent by the unit depends on the requested operation, it is recommended that a fixed timeout is used by the remote unit of 500ms (for direct PC links).

4.3. Transmission Mediums

The protocol can be used to transfer data over either a direct link to a PC via USB or via Bluetooth.

4.3.1. Direct Links

Given the example of a direct link to a PC via a virtual serial port via USB, the protocol can be used without being encapsulated in any other protocols.

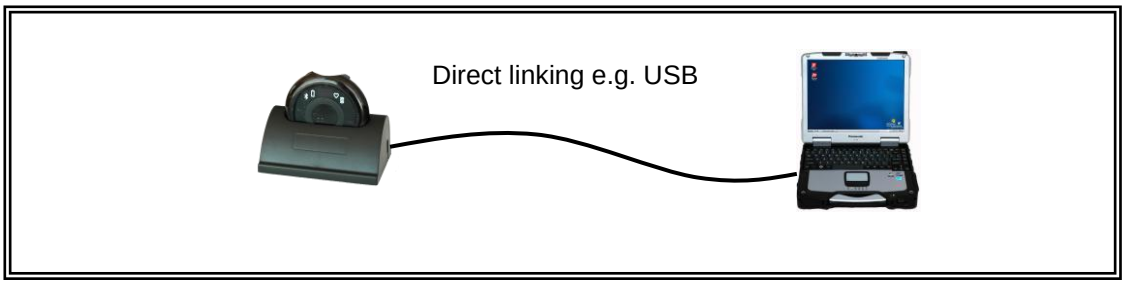

Figure 4-2 Direct Communications Example

In the figure shown above, a direct link between a PC and a Bio Harness can be implemented and requires no modification to the protocol described in this document. The link is point-to-point and therefore no unit addressing is required. Because the link is direct, the only delays between the request and response messages will be due to the amount of processing the receiver has to perform.

4.3.2. Wireless Links

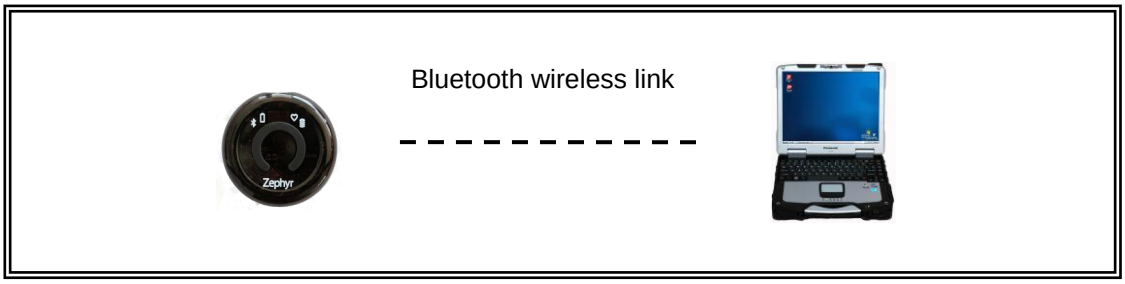

Figure 4-3 Wireless Communications Example

The figure above shows a wireless link using Bluetooth. As Bluetooth provides direct links between units on a PAN (Personal Area Network), the packets sent are generally point-to-point and therefore again there is no requirement to use addressing. The inherent addressing scheme of Bluetooth takes care of it for us. There may be several messages wrapped within one Bluetooth packet and therefore this must be taken into account.

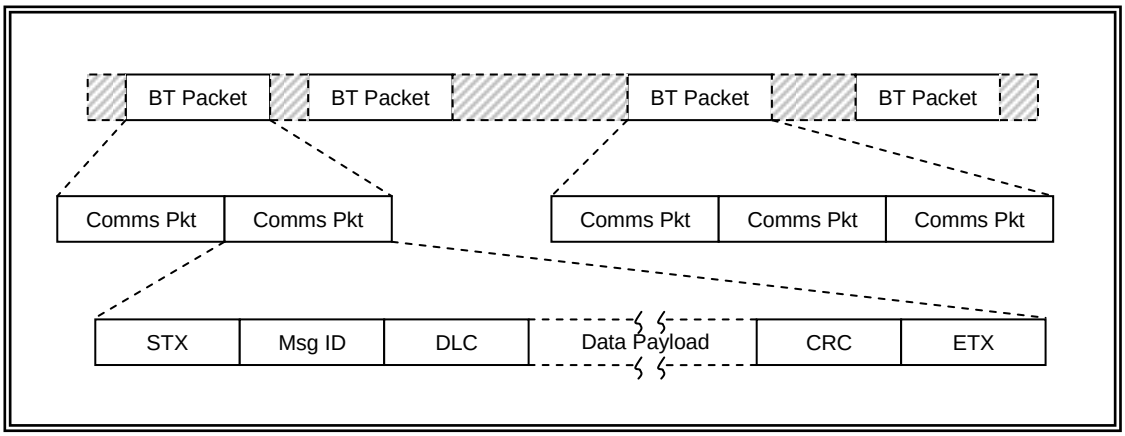

Figure 4-4 RF Packet Encapsulation

The figure above shows that when sending standard communications packets over a Bluetooth link, the possibility must be taken into account that multiple communication packets (specified in this document) may be concatenated into one Bluetooth packet and sent to the remote unit. Therefore the message parsing method employed by the receiver must be able to extract one or several communications packets from within a Bluetooth packet.

5. Standard Messages

This section specifies request/response messages that are implemented in the BioHarness, valid range of values and approximate response timeouts. Both the request and response formats are specified.

5.1. MSG:0x01 - Read Logging Data

5.1.1. Request Message

This message requests logged data from the Bio Harness. The offset within the log and the number of bytes to be read is specified.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x01								Msg ID
2	6								DLC
3	File Number								Payload
4	Offset (LS Byte)								
5									
6									
7	Offset (MS Byte)								
8	Number of bytes to read								
9	CRC								CRC
10	ETX								ETX

Table 5-1 ***Request: Read Data from the Log***

- The File Number is used to reference a particular log file in the log. At present, only one log file is stored, but there may be a need to use more than one file in the future. At present however, the value zero can be used in this field.
- The Offset is a 32-bit binary value in Little Endian format. The LS byte is located first within the message. The minimum value is zero and the maximum is one less than the file size. If a request is made to read data beyond the end of the file, a NAK response will be issued.
- The number of bytes to read must be either 1 or a multiple of 2. The minimum value is 1 and the maximum is 128. Odd values other than 1 are not permitted.

5.1.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x01								Msg ID
2	Number of bytes read								DLC
3	Data read from the Log								Payload
3+DLC	CRC								CRC
4+DLC	ACK								ACK/NAK

Table 5-2 ***Response: Read Data from the Log***

- The number of bytes read (DLC) is the same as the requested number of bytes in the request message.
- The data payload could be anywhere between 1 and 128 bytes in length.
- An ACK is appended to the message.
- If the data couldn't be read from the Log for any reason, a NAK is sent in response.

5.2. MSG:0x02 - Delete Log File

5.2.1. Request Message

This message requests that the Bio Harness deletes a log file.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x02								Msg ID
2	1								DLC
3	File Number								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-3 ***Request: Delete Log File***

- The File Number is used to reference a particular log file in the log. At present, only one log file is stored, but we may use more in the future. At the moment, the value zero can be used in this field.

5.2.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x02								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-4 ***Response: Delete Log File***

- The response simply indicates to the PC that the log file has been deleted.
- If the file couldn't be deleted, a NAK is sent.

5.3. MSG:0x07 - Set RTC Date/Time

5.3.1. Request Message

This message sets the date and time within the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x07								Msg ID
2	7								DLC
3	Day								Payload
4	Month								
5	Year (LS Byte)								
6	Year (MS Byte)								
7	Hours								
8	Minutes								
9	Seconds								
10	CRC								CRC
11	ETX								ETX

Table 5-5 ***Request: Set RTC Date/Time***

- The Day field is the day of the month, valid values are 1…31 inclusive.
- The Month field is the month of the year, valid values are 1…12 inclusive.
- Year is in Little Endian format (LS byte first), valid values are 2000…2099 inclusive.
- Hours is the hour of the day, valid values are 0…23 inclusive.
- Minutes is the minute of the hour, valid values are 0…59 inclusive.
- Seconds is the second of the minute, valid values are 0…59 inclusive.
- If any of the payload values are found to be invalid, a NAK is sent back to the PC.

5.3.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x07								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-6 ***Response: Set RTC Date/Time***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and the RTC in the remote unit was updated.

5.4. MSG:0x08 - Get RTC Date/Time

5.4.1. Request Message

This message requests the current date and time within the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x08								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-7 ***Request: Get RTC Date/Time***

- There is no data payload within the message as it is simply a command to get the current date and time from the remote unit.

5.4.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x08								Msg ID
2	7								DLC
3	Day								Payload
4	Month								
5	Year (LS Byte)								
6	Year (MS Byte)								
7	Hours								
8	Minutes								
9	Seconds								
10	CRC								CRC
11	ACK								ACK/NAK

Table 5-8 ***Response: Get RTC Date/Time***

- The Day field is the day of the month, valid values are 1…31 inclusive.
- The Month field is the month of the year, valid values are 1…12 inclusive.
- Year is in Little Endian format (LS byte first), valid values are 2000…2099 inclusive.
- Hours is the hour of the day, valid values are 0…23 inclusive.
- Minutes is the minute of the hour, valid values are 0…59 inclusive.
- Seconds is the second of the minute, valid values are 0…59 inclusive.

5.5. MSG:0x09 - Get Boot Software Version

5.5.1. Request Message

This message requests the Bootloader software version from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x09								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-9 ***Request: Get Boot Software Version***

- There is no data payload within the message as it is simply a command to get the Boot Software Version from the remote unit.

5.5.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x09								Msg ID
2	8								DLC
3	Major Version (LS Byte)								Payload
4	Major Version (MS Byte)								
5	Minor Version (LS Byte)								
6	Minor Version (MS Byte)								
7	Reserved (LS Byte)								
8	Reserved (MS Byte)								
9	Build Number (LS Byte)								
10	Build Number (MS Byte)								
11	CRC								CRC
12	ACK								ACK/NAK

Table 5-10 ***Response: Get Boot Software Version***

- The Major Version field is in binary and in Little Endian format (LS Byte first).
- The Minor Version field is in binary and in Little Endian format (LS Byte first).
- The Reserved field is not currently used and should be set to zero.
- The Build Number is in binary and in Little Format format (LS Byte first).

5.6. MSG:0x0A - Get Application Software Version

5.6.1. Request Message

This message requests the Application software version from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0A								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-11 ***Request: Get Application Software Version***

- There is no data payload within the message as it is simply a command to get the Application Software Version from the remote unit.

5.6.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0A								Msg ID
2	8								DLC
3	Major Version (LS Byte)								Payload
4	Major Version (MS Byte)								
5	Minor Version (LS Byte)								
6	Minor Version (MS Byte)								
7	Reserved (LS Byte)								
8	Reserved (MS Byte)								
9	Build Number (LS Byte)								
10	Build Number (MS Byte)								
11	CRC								CRC
12	ACK								ACK/NAK

Table 5-12 ***Response: Get Application Software Version***

- The Major Version field is in binary and in Little Endian format (LS Byte first).
- The Minor Version field is in binary and in Little Endian format (LS Byte first).
- The Reserved field is not currently used and should be set to zero.
- The Build Number is in binary and in Little Endian format (LS Byte first).

5.7. MSG:0x0B - Get Serial Number

5.7.1. Request Message

This message requests the Serial Number from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0B								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-13 ***Request: Get Serial Number***

- There is no data payload within the message as it is simply a command to get the Serial Number from the remote unit.

5.7.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0B								Msg ID
2	12								DLC
3	Serial Number String (ASCII) – 1st Byte								Payload
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14	Serial Number String (ASCII) – 12th Byte								
15	CRC								CRC
16	ACK								ACK/NAK

Table 5-14 ***Response: Get Serial Number***

- The Serial Number string is in ASCII and is 12 characters in length e.g. “BHT123456789”.
- There is no NULL terminator at the end of the string; thus the receiving application must append a NULL to the string data.
- If the serial number is less than 12 bytes in length, the end of the string is padded with spaces.

5.8. MSG:0x0C - Get Hardware Part Number

5.8.1. Request Message

This message requests the Hardware Part Number from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0C								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-15 ***Request: Get Hardware Part Number***

- There is no data payload within the message as it is simply a command to get the Hardware Part Number from the remote unit.

5.8.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0C								Msg ID
2	12								DLC
3	Hardware Part Number String (ASCII) – 1st Byte								Payload
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14	Hardware Part Number String (ASCII) – 12th Byte								
15	CRC								CRC
16	ACK								ACK/NAK

Table 5-16 ***Response: Get Hardware Part Number***

- The Hardware Part Number string is in ASCII and is 12 characters in length e.g. “9900.0085v1a”.
- There is no NULL terminator at the end of the string; thus the receiving application must append a NULL at the end of the string data.
- If the part number is less than 12 bytes in length, the end of the string is padded with spaces.

5.9. MSG:0x0D - Get Bootloader Part Number

5.9.1. Request Message

This message requests the Boot Part Number from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0D								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-17 ***Request: Get Bootloader Part Number***

- There is no data payload within the message as it is simply a command to get the Boot Part Number from the remote unit.

5.9.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0D								Msg ID
2	12								DLC
3	Boot Part Number String (ASCII) – 1st Byte								Payload
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14	Boot Part Number String (ASCII) – 12th Byte								
15	CRC								CRC
16	ACK								ACK/NAK

Table 5-18 ***Response: Get Bootloader Part Number***

- The Bootloader Part Number string is in ASCII and is 12 characters in length e.g. “9500.0001”.
- There is no NULL terminator at the end of the string; thus the receiving application must append a NULL to the end of the string data.
- If the part number is less than 12 bytes in length, the end of the string is padded with spaces.

5.10. MSG:0x0E - Get Application Part Number

5.10.1. Request Message

This message requests the Application Part Number from the remote unit.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0E								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-19 ***Request: Get Application Part Number***

- There is no data payload within the message as it is simply a command to get the Application Part Number from the remote unit.

5.10.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x0E								Msg ID
2	12								DLC
3	Application Part Number String (ASCII) – 1st Byte								Payload
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14	Application Part Number String (ASCII) – 12th Byte								
15	CRC								CRC
16	ACK								ACK/NAK

Table 5-20 ***Response: Get Application Part Number***

- The Application Part Number string is in ASCII and is 12 characters in length e.g. “9500.0002”.
- There is no NULL terminator at the end of the string; thus the receiving application must append a NULL to the end of the string data.
- If the part number is less than 12 bytes in length, the end of the string is padded with spaces.

5.11. MSG:0x10 - Set Network ID

5.11.1. Request Message

This message sets the Bluetooth Network ID within the device. Note that the Network ID makes up part of the Bluetooth Friendly name:

"<Unit Type> <Network ID>"

e.g.

"BH John Smith 5"

Contains a unit type of **BH** (Bio Harness) and a Network ID of **John Smith 5**.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x10								Msg ID
2	Number of characters in string.								DLC
3	Bluetooth Network ID (ASCII) – 1st Byte								Payload
:									
:									
:									
:									
:									
DLC+3	Bluetooth Network ID (ASCII) – Last Byte								
DLC+4	CRC								CRC
DLC+5	ETX								ETX

Table 5-21 ***Request: Set Network ID***

- The Network ID string is in ASCII.
- The string has a minimum of 2 a maximum of 29 characters in length.
- There is no NULL terminator at the end of the string.

5.11.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x10								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-22 ***Response: Set Network ID***

- There is no data payload within the message as the packet merely acknowledges that the remote unit accepted the new Bluetooth Network ID.

5.12. MSG:0x11 - Get Network ID

5.12.1. Request Message

This message gets the Bluetooth Network ID from the device. Note that the Network ID makes up part of the Bluetooth Friendly name:

"**<Unit Type> <Network ID>**"

e.g.

"**BH John Smith 5**"

Contains a unit type of **BH** (Bio Harness) and a Network ID of **John Smith 5**.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x11								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-23 ***Request: Get Network ID***

- There is no data payload within the message as it is simply a command to get the Bluetooth Network ID from the remote unit.

5.12.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x11								Msg ID
2	Number of characters in string.								DLC
3	Bluetooth Network ID (ASCII) – 1st Byte								Payload
:									
:									
:									
:									
:									
DLC+3	Bluetooth Network ID (ASCII) – Last Byte								
DLC+4	CRC								CRC
DLC+5	ACK								ACK/NAK

Table 5-24 ***Response: Get Network ID***

- The Bluetooth string is in ASCII.
- The string has a minimum of 2 a maximum of 29 characters in length.
- There is no NULL terminator at the end of the string.

5.13. MSG:0x12 - Get Unit MAC Address

5.13.1. Request Message

This message gets the Bluetooth MAC address from the device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x12								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-25 ***Request: Get Unit MAC Address***

- There is no data payload within the message as it is simply a command to get the Bluetooth MAC Address from the device.

5.13.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x12								Msg ID
2	17								DLC
3	Bluetooth MAC Address String (ASCII) – 1st Byte								Payload
:									
:									
:									
:									
:									
19	Bluetooth MAC Address String (ASCII) – 17th Byte								
20	CRC								CRC
21	ACK								ACK/NAK

Table 5-26 ***Response: Get Unit MAC Address***

- The MAC Address string is in ASCII.
- The string has a fixed number of characters (17) e.g. "**00:07:80:82:7A:61**".
- There is no NULL terminator at the end of the string and therefore it is the responsibility of the receiving application to append a NULL to the string data.

5.14. MSG:0x14 - Set General Data Packet Transmit State

5.14.1. Request Message

This message requests that the Bio Harness enables or disables the Bluetooth transmission of the periodic General Data Packet

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x14								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-27 ***Request: Set General Data Packet Transmit State***

- This command enables (payload = 1) or disables (payload = 0) the General Data Packet transmission. When enabled, the general data packet is transmitted periodically (every 1.008 seconds).

5.14.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x14								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-28 ***Response: Set General Data Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.15. MSG:0x15 - Set Breathing Waveform Packet Transmit State

5.15.1. Request Message

This message requests that the Bio Harness enables or disables the Bluetooth transmission of the periodic Breathing Waveform Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x15								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-29 ***Request: Set Breathing Waveform Packet Transmit State***

- This command enables (payload = 1) or disables (payload = 0) the Breathing Waveform Packet transmission. When enabled, the packet is transmitted periodically (every 1.008 seconds).

5.15.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x15								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-30 ***Response: Set Breathing Waveform Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.16. MSG:0x16 - Set ECG Waveform Packet Transmit State

5.16.1. Request Message

This message requests that the Bio Harness enables or disables the Bluetooth transmission of the periodic ECG Waveform Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x16								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-31 ***Request: Set ECG Waveform Packet Transmit State***

- This command enables (payload = 1) of disables (payload = 0) the ECG Waveform Packet transmission. When enabled, the packet is transmitted periodically (every 252ms).

5.16.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x16								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-32 ***Response: Set ECG Waveform Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.17. MSG:0x17 - Get Unit Bluetooth Friendly Name

5.17.1. Request Message

This message gets the Bluetooth 'Friendly Name' from the device. The Bluetooth name consists of the Unit Type and the Network ID:

"**<Unit Type> <Network ID>**"

e.g. Friendly Name:

"**BH John Smith 5**"

Contains a unit type of **BH** (Bio Harness) and a Network ID of **John Smith 5**.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x17								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-33 ***Request: Get Bluetooth Friendly Name***

- There is no data payload within the message as it is simply a command to get the Bluetooth Friendly Name from the remote unit.

5.17.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x17								Msg ID
2	Number of characters in string.								DLC
3	Bluetooth Friendly Name (ASCII) – 1st Byte								Payload
:									
:									
:									
:									
:									
DLC+3	Bluetooth Friendly Name (ASCII) – Last Byte								
DLC+4	CRC								CRC
DLC+5	ACK								ACK/NAK

Table 5-34 ***Response: Get Bluetooth Friendly Name***

- The Bluetooth string is in ASCII.
- The string has a minimum of 4 a maximum of 32 characters in length.
- There is no NULL terminator at the end of the string.

5.18. MSG:0x19 - Set R to R Data Packet Transmit State

5.18.1. Request Message

This message requests that the Bio Harness enables or disables the Bluetooth transmission of the periodic R to R Data Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x19								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-35 ***Request: Set R to R Data Packet Transmit State***

- This command enables (payload = 1) of disables (payload = 0) the R to R Data Packet transmission. When enabled, the packet is transmitted periodically (every 1008ms).

5.18.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x19								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-36 ***Response: Set R to R Waveform Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.19. MSG:0x1E - Set Accelerometer Packet Transmit State

5.19.1. Request Message

This message requests that the Bio Harness enables or disables the Bluetooth transmission of the periodic Accelerometer Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x1E								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-37 ***Request: Set Accelerometer Packet Transmit State***

- This command enables (payload = 1) of disables (payload = 0) the Accelerometer Packet transmission. When enabled, the packet is transmitted periodically (every 400ms).

5.19.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x1E								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-38 ***Response: Set Accelerometer Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.20. MSG:0x9B - Set ROG Settings

This message is applicable to devices configured to use version 1 of Zephyr's ROG algorithm – up to Bluetooth firmware release v2.3.2.0.

5.20.1. Request Message

This message sets the ROG thresholds that the device uses for the ROG algorithm.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x9B								Msg ID
2	16								DLC
3	Heart Rate High (0…240) – LS Byte								Payload
4	Heart Rate High (0…240) – MS Byte								
5	Heart Rate Low (0…240) – LS Byte								
6	Heart Rate Low (0…240) – MS Byte								
7	Respiration Rate High (0…70) – LS Byte								
8	Respiration Rate High (0…70) – MS Byte								
9	Respiration Rate Low (0…70) – LS Byte								
10	Respiration Rate Low (0…70) – MS Byte								
11	Activity High (0…16) – LS Byte								
12	Activity High (0…16) – MS Byte								
13	Activity Low (0…16) – LS Byte								
14	Activity Low (0…16) – MS Byte								
15	Green to Orange Time – LS Byte								
16	Green to Orange Time – MS Byte								
17	Orange to Red Time – LS Byte								
18	Orange to Red Time – MS Byte								
19	CRC								CRC
20	ETX								ETX

Table 5-39 ***Request: Set ROG Settings***

- The data payload has the following specifications:
  - Heart Rate: 0…240 with 1 unit resolution e.g. 132 = 132 BPM
  - Respiration Rate: 0…70 with 0.1 unit resolution e.g. 173 = 17.3 BPM
  - Activity: 0…16 with 0.01 unit resolution e.g. 256 = 2.56 VMU
  - The times are in seconds.

5.20.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x9B								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-40 ***Response: Set ROG Settings***

- There is no data payload within the message as it simply acknowledges that the ROG Settings were stored.

5.21. MSG:0x9C - Get ROG Settings

This message is applicable to devices configured to use version 1 of Zephyr's ROG algorithm – up to Bluetooth firmware release v2.3.2.0.

5.21.1. Request Message

This message gets the ROG thresholds that the device uses for the ROG algorithm.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x9C								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-41 ***Request: Get ROG Settings***

- There is no data payload within the message as it is simply a command to get the ROG Settings from the device.

5.21.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x9C								Msg ID
2	16								DLC
3	Heart Rate High (0…240) – LS Byte								Payload
4	Heart Rate High (0…240) – MS Byte								
5	Heart Rate Low (0…240) – LS Byte								
6	Heart Rate Low (0…240) – MS Byte								
7	Respiration Rate High (0…70) – LS Byte								
8	Respiration Rate High (0…70) – MS Byte								
9	Respiration Rate Low (0…70) – LS Byte								
10	Respiration Rate Low (0…70) – MS Byte								
11	Activity High (0…16) – LS Byte								
12	Activity High (0…16) – MS Byte								
13	Activity Low (0…16) – LS Byte								
14	Activity Low (0…16) – MS Byte								
15	Green to Orange Time – LS Byte								
16	Green to Orange Time – MS Byte								
17	Orange to Red Time – LS Byte								
18	Orange to Red Time – MS Byte								
19	CRC								CRC
20	ACK								ACK/NAK

Table 5-42 ***Response: Get ROG Settings***

- The data payload has the following specifications:
  - Heart Rate: 0…240 with 1 unit resolution e.g. 132 = 132 BPM
  - Respiration Rate: 0…70 with 0.1 unit resolution e.g. 173 = 17.3 BPM
  - Activity: 0…16 with 0.01 unit resolution e.g. 256 = 2.56 VMU
  - The times are in seconds.

5.22. MSG:0xA2 - Set Bluetooth User Config

5.22.1. Request Message

This message sets the user configurable Bluetooth settings.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA2								Msg ID
2	Payload Length								DLC
3	UC1- Device Discoverable – LS Byte								Payload
4	UC1 -Device Discoverable – MS Byte								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-43 ***Request: Set Bluetooth User Config***

- Payload Length
  - Length of data within message.
  - Length is 2 in this example but can grow up to 30 bytes when more Bluetooth settings are configurable.
- UCx
  - User Config Settings (UC1 – UC15)
  - UC1 Device Discoverable
    - 0 Device not visible in inquiry.
    - 1 Device visible in inquiry
  - UC2 – UC15 Reserved for future use.

5.22.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA2								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-44 ***Response: Set Bluetooth User Config***

- There is no data payload within the message as it simply acknowledges that the Bluetooth Settings were stored.

5.23. MSG:0xA3 - Get Bluetooth User Config

5.23.1. Request Message

This message requests the user configurable Bluetooth settings.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA3								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-45 ***Request: Get Bluetooth User Config***

- There is no data payload within the message as it is simply a command to get the user configurable Bluetooth settings from the device.

5.23.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA3								Msg ID
2	Payload Length								DLC
3	UC1- Device Discoverable – LS Byte								Payload
4	UC1 -Device Discoverable – MS Byte								
DLC+3	CRC								CRC
DLC+4	ACK/NAK								ACK/NAK

Table 5-46 ***Response: Get Bluetooth User Config***

- Payload Length
  - Length of data within message.
  - Length is 2 in this example but can grow up to 30 bytes when more Bluetooth settings are configurable.
- UCx
  - User Config Settings (UC1 – UC15)
  - UC1 Device Discoverable
    - 0 Device not visible in inquiry.
    - 1 Device visible in inquiry
  - UC2 – UC15 Reserved for future use

5.24. MSG:0XA4 - Set BT Link Config

5.24.1. Request Message

This message sets the timeout & lifesign periods for a Bluetooth link.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA4								Msg ID
2	4								DLC
3	Link Timeout (ms) – LS Byte								Payload
4	Link Timeout (ms) – MS Byte								
5	Lifesign Period (ms) – LS Byte								
6	Lifesign Period (ms) – MS Byte								
7	CRC								CRC
8	ETX								ETX

Table 5-47 ***Request: Set BT Link Config***

- Link Timeout
  - Link closed if no data received over link within this time in ms.
  - Set to 0 if no timeout required.
- Lifesign Period
  - Time in ms between sending Lifesign messages to remote device.
  - Set to 0 to never send Lifesign messages.

5.24.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA4								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-48 ***Response: Set BT Link Config***

- There is no data payload within the message as it simply acknowledges that the BT Link Settings were stored.

5.25. MSG:0xA5 - Get BT Link Config

5.25.1. Request Message

This message requests the timeout & lifesign periods for a Bluetooth link.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA5								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-49 ***Request: Get BT Link Config***

- There is no data payload within the message as it is simply a command to get the BT Link Settings from the device.

5.25.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA5								Msg ID
2	Payload Length								DLC
3	Link Timeout (ms) – LS Byte								Payload
4	Link Timeout (ms) – MS Byte								
5	Lifesign Period (ms) – LS Byte								
6	Lifesign Period (ms) – MS Byte								
7	CRC								CRC
8	ACK/NAK								ACK/NAK

Table 5-50 ***Response: Get BT Link Config***

- Link Timeout
  - Link closed if no data received over link within this time in ms.
  - Set to 0 if no timeout required.
- Lifesign Period
  - Time in ms between sending Lifesign messages to remote device.
  - Set to 0 to never send Lifesign messages.

5.26. MSG:0xA6 - Set BioHarness User Config

5.26.1. Request Message

This message sets the BioHarness user configurable settings.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA6								Msg ID
2	Payload Length								DLC
3	Log Enable								Payload
4	Bluetooth Enable								
5	Button Enable When Worn								
6	ECG Polarity								
7	Logging Format								
8	Team System Enable								
9	802.15.4 Enable								
10	LED Enable								
11	Audio Enable								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-51 ***Request: Set BioHarness User Config***

- Payload Length is 9 in this example but can grow up to 30 bytes when more BioHarness user settings are configurable.
- The 'Get Supported Log Formats' (0xD5) message should be used to determine which log formats can be set. If that message is not supported by the device, log formats 0-2 may be set as listed below.

Configuration Item		Value	Description
Num.	Name		
0	Log Enable	0	Disable logging of physiological data
		1	Enable logging of physiological data
1	Bluetooth Enable	0	Disable Bluetooth communications
		1	Enable Bluetooth communications
2	Button Enable when worn	0	Button disabled when worn (pressing has no effect)
		1	Button enabled when worn
3	ECG Polarity	0	Normal Polarity
		1	Reversed Polarity
4	Logging Format - use 'Get Supported Log Formats' (0xD5) message	0	General Logging Format
		1	General Logging Format with ECG Data
		2	General Logging Format with Accelerometer Data
		3-254	Use 'Get Supported Log Formats' (0xD5) message
5	Team System Enable	0	Team System Mode Disabled
		1	Team System Mode Enabled
6	802.15.4 Enable	0	Disable 802.15.4 communications
		1	Enable 802.15.4 communications
7	LED Enable	0	Disable LEDs
		1	Enable LEDs
8	Audio Enable	0	Disable audio feedback
		1	Enable audio feedback

Table 5-52 **BioHarness User Configuration Items**

5.26.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA6								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-53 ***Response: Set BioHarness User Config***

- There is no data payload within the message as it simply acknowledges that the BioHarness user Settings were stored.

5.27. MSG 0xA7 - Get BioHarness User Config

5.27.1. Request Message

This message requests the BioHarness user configurable settings.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA7								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-54 ***Request: Get BioHarness User Config***

- There is no data payload within the message as it is simply a command to get the BioHarness User Settings from the device.

5.27.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xA7								Msg ID
2	Payload Length								DLC
3	Log Enable								Payload
4	RF Enable								
5	Button Enable When Worn								
6	ECG Polarity								
7	Logging Format								
8	Team System Enable								
9	802.15.4 Enable								
10	LED Enable								
11	Audio Enable								
DLC+3	CRC								CRC
DLC+4	ACK/NAK								ACK/NAK

Table 5-55 ***Response: Get BioHarness User Config***

- Payload Length is 9 in this example but can grow up to 30 bytes when more BioHarness user settings are configurable.
- See Table 5- for a description of the configuration items in this message. A value of 255 can be returned for a configuration item to indicate that it is not supported by the device. This allows any configuration tool to determine which configuration is supported by which device.

5.28. MSG:0xAC – Get Battery Status

5.28.1. Request Message

This message requests the current battery voltage in millivolts and the battery charge as a percentage of full charge.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xAC								Msg ID
2	0								DLC
4	CRC								CRC
5	ETX								ETX

Table 5-56 ***Request: Get Battery Status***

5.28.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xAC								Msg ID
2	3								DLC
3	Battery Voltage (mv) – LS Byte								Payload
4	Battery Voltage (mv) – MS Byte								
5	Battery Charge (%)								
6	CRC								CRC
7	ACK								ACK/NAK

Table 5-57 ***Response: Get Battery Status***

- Current Battery voltage in millivolts (0…max voltage).
- Current Battery Charge in % (0…100).
- For a BioHarness 3.0, this message is valid when the device is *not* in its charge cradle (otherwise the *charging* voltage is returned, which is invariably equivalent to 100%)

5.29. MSG:0x1F – Reboot Unit

5.29.1. Request Message

This message requests that the remote unit performs a reboot. As soon as the unit has sent the acknowledgement to the PC, it performs reboot via a watchdog timer reset.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x1F								Msg ID
2	7								DLC
3	Verification String (ASCII) – 1st Byte								Payload
4									
5									
6									
7									
8									
9	Verification String (ASCII) – 7th Byte								
10	CRC								CRC
11	ETX								ETX

Table 5-58 ***Request: Reboot Unit***

- The payload should contain a verification string "**ZReBoot**". If the data in the payload received by the unit does not match the string (case sensitive), a NAK is returned as the response.

5.29.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x1F								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-59 ***Response: Reboot Unit***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the unit will now perform a reboot.
- After the response message has been sent to the PC, the unit reboots after approximately 125ms. The PC will have to wait a further 5 seconds before the unit can continue communications (unless Boot mode communications are required).

5.30. MSG:0xB0 – Bluetooth Peripheral Message

5.30.1. Request Message

This message requests that the remote unit performs the requested action on the specified Bluetooth device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB0								Msg ID
2	DLC								DLC
3	Sequence ID								Payload
4	Requested Action								
5	Bluetooth MAC Address - 1st Byte								
:	:								
10	Bluetooth MAC Address - 6th Byte								
:	Action Parameters								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-60 ***Request: Bluetooth Peripheral Message***

- The Sequence ID is incremented by the sender for every message sent.
- The Bluetooth MAC address is a 6-byte MAC address in binary representation used to identify the device.
- The action Parameters field depends on the Requested Action, see 5.62 below.

Action		Parameters		
Name	**Code**	**Byte**	**Description**	**Notes**
disconnect from device	0x00	none		
connect to device	0x01	0	Device Manufacturer Code (see Table 5-)	0 .. 255
		1	Device Type Code (see Table 5-)	0 .. 255
		2,3	Sampling period (see Table 5-)	
		4,5	Logging period (see Table 5-)	
		6	Bluetooth PIN (ASCII) - 1st Byte	optional
		:	:	
		n	Bluetooth PIN (ASCII) - nth Byte	
update configuration	0x02	0,1	Sampling period (see Table 5-)	
		2,3	Logging period (see Table 5-)	

Table 5-61 ***Actions and their Parameters***

- The Bluetooth PIN is an optional, variable length array of ASCII characters which is not null-terminated, its length can be determined from the DLC of the message.
- If the update configuration action is sent, the device will reset its sampling cycle and send the next sample immediately.
- If the device is unable to sample measurements at the rate specified by the sampling period, then it will send samples at the highest possible rate.

Manufacturer		Device	
Name	**Code**	**Name**	**Code**
Zephyr Technology	0x01	BioHarness	0x01
Nonin Medical	0x02	Nonin 9560 Onyx II SpO2 sensor	0x01
Mytech Technology	0x03	Mytech HPL-108 blood pressure sensor	0x01

Table 5-62 ***Device Manufacturers and Types***

Byte/Bit	7	6	5	4	3	2	1	0
0	Unit (see Table 5-)				time period value (bits 11 .. 8)			
1	time period value (bits 7 .. 0)							

Table 5-63 ***Sampling and Logging Period***

code	unit
0	milliseconds
1	seconds
2 .. 15	reserved for future use

Table 5-64 ***Unit Codes***

5.30.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB0								Msg ID
2	1								DLC
3	Sequence ID								Payload
4	CRC								CRC
5	ACK/NAK								ACK/NAK

Table 5-65 ***Response: Bluetooth Peripheral Message***

The response message will contain a NAK, if:

- The specified device/manufacturer combination is not known/supported
- Request to connect to a device, but the maximum number of simultaneous connections to the same type of device has already been reached
- Request to close a non-existing connection
- Requested action is not known

Otherwise an ACK response is returned.

The Sequence ID must be identical to the one received in the Request message.

5.31. MSG:0xB3 – Reset Configuration

5.31.1. Request Message

This message is used to reset the device configuration according to the specified mode. This command may also cause the device to be restarted.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB3								Msg ID
2	1								DLC
3	Mode								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-66 ***Request: Reset Configuration***

The Mode field determines how and what aspect of the configuration is reset, according to the following table:

Mode	Description
0	Reset all configuration to factory defaults
1	Reset all configuration except calibration data
2-255	Reserved for future use

5.31.2. Response Message

An ACK response indicates that the command is understood and shall be carried out shortly. If the command or the specified mode is not supported by the command, or for any reason the command cannot be performed, a NAK response is returned.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB3								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-67 ***Response: Reset Configuration***

5.32. MSG:0xB4 - Set Accelerometer Axis Mapping

5.32.1. Request Message

This message sets the axis mapping information for the accelerometer. Any accelerometer axis can be mapped to any other axis as well as being inverted to allow the device to be used in a number of different orientations (e.g. worn on the front, worn on the side, upside down, etc.).

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB4								Msg ID
2	3								DLC
3	X-axis mapping (see Table 5-)								Payload
4	Y-axis mapping (see Table 5-)								
5	Z-axis mapping (see Table 5-)								
6	CRC								CRC
7	ETX								ETX

Table 5-68 ***Request: Set Accelerometer Axis Mapping***

Mapping data is provided for each axis to determine which axis it should be mapped to and whether it should be inverted. For example if a BioHarness is used on the right side facing backwards, then the mapping is as follows (with negative sign showing inversion):
(-X) → (Y), (-Y) → (Z), (Z) → (X), encoded according to Table 5-.

5.32.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB4								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-69 ***Response: Set Accelerometer Offset***

- There is no data payload within the message as it simply acknowledges that the Accelerometer Axis Mapping Settings were stored.
- If the Axis Index is illegal, a NAK is returned.

Byte/Bit	7	6	5	4	3	2	1	0
0	invert	Not used (should be set to 0)					Axis Index	

Table 5-70 **Axis Mapping Data**

Mapping Data	Description
Invert	0 = axis not inverted, 1 = axis inverted
Axis Index	Index of axis to map to: 0 = X Axis, 1 = Y Axis, 2 = Z Axis, 3 = illegal

5.33. MSG 0xB5 - Get Accelerometer Axis Mapping

5.33.1. Request Message

This message reads the accelerometer axis mapping.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB5								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 5-71 ***Request: Get Accelerometer Axis Mapping***

- There is no data payload within the message as it is simply a command to get the BioHarness User Settings from the device.

5.33.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB5								Msg ID
2	Payload Length								DLC
3	X-axis mapping (see Table 5-)								Payload
4	Y-axis mapping (see Table 5-)								
5	Z-axis mapping (see Table 5-)								
6	CRC								CRC
7	ACK/NAK								ACK/NAK

Table 5-72 ***Response: Get Accelerometer Axis Mapping***

5.34. MSG:0xB6 - Set Algorithm Config

5.34.1. Request Message

This message sets the Configuration for extended data algorithms.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB6								Msg ID
2	DLC								DLC
3	Data Algorithm Identifier								Payload
4	Configuration Data – 1st byte								
:	:								
:	:								
DLC+2	Configuration Data nth byte								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-73 ***Request: Set Algorithm Config***

- Length of message is variable (depending on algorithm type).
- Length of message for a given data type may increase in the future to accommodate additional configuration options for that algorithm if needed, therefore implementations should not fail if the DLC is greater than the expected value.

7	6	5	4	3	2	1	0
unused		**TXEN**	**Type (0 .. 31)**				
Type	Identifies the type of algorithm to configure						
TXEN	Transmit state of data packets for this algorithm (1=enabled, 0=disabled)						

Table 5-74 **Algorithm Identifier**

Type		Configuration Data		
Name	**Code**	**Byte**	**Description**	**Notes**
Heart Rate Recovery	0x00	0	Heart Rate Max	bpm
		1	Activity Running Threshold	0.01g
		2	Activity High Jogging Threshold	0.01g
		3	Activity Low Jogging Threshold	0.01g
		4	Activity High Walking Threshold	0.01g
		5	Activity Low Walking Threshold	0.01g
		6	Activity High Idle Threshold	0.01g
Orthostatic Hypotension	0x01	*No configuration for this algorithm*		
Vertical Jump	0x02	*No configuration for this algorithm*		
Forty Yard Dash	0x03	*No configuration for this algorithm*		

Table 5-75 **Algorithm Configuration**

5.34.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB6								Msg ID
2	1								DLC
3	Algorithm Identifier (see Table 5-)								Payload
4	CRC								CRC
5	ACK/NAK								ACK/NAK

Table 5-76 ***Response: Set Algorithm Config***

- If the specified algorithm is supported, an ACK is returned to acknowledge that the configuration was received.
- The response message contains a copy of the algorithm identifier that was sent in the request message.

5.35. MSG 0xB7 - Get Algorithm Config

5.35.1. Request Message

This message reads the Configuration for extended data algorithms.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB7								Msg ID
2	1								DLC
3	0	0	0	Algorithm Type (0 .. 31)					Payload
4	CRC								CRC
5	ETX								ETX

Table 5-77 ***Request: Get Algorithm Config***

5.35.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB7								Msg ID
2	DLC								DLC
3	Algorithm Identifier (see Table 5-)								
4	Configuration Data – 1st byte								Payload
:	:								
:	:								
DLC+2	Configuration Data nth byte								
DLC+3	CRC								CRC
DLC+4	ACK/NAK								ACK/NAK

Table 5-78 ***Response: Get Algorithm Config***

- Length of response message is variable (depending on algorithm type).
- The format of configuration data is specific to each data type. See Table 5- for details.

5.36. MSG:0xB8 - Set Extended Data Packet Transmit State

5.36.1. Request Message

This message sets the Transmit State for Extended Data Packets.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB8								Msg ID
2	4								DLC
3	Bit-List of extended packets 0 to 7 to enable (LS Byte)								Payload
4	Bit-List of extended packets 8 to 15 to enable								
5	Bit-List of extended packets 16 to 23 to enable								
6	Bit-List of extended packets 24 to 31 to enable (MS Byte)								
7	CRC								CRC
8	ETX								ETX

Table 5-79 ***Request: Set Extended Data Packet Transmit State***

The specified bit list determines which data packets are allowed to be transmitted. Bits 0 to 31 represent Extended Data Types 0 to 31. For each bit set, the corresponding type of extended data is enabled, if the bit is cleared, it is disabled.

5.36.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB8								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-80 ***Response: Set Extended Data Packet Transmit State***

- There is no data payload within the message as it simply acknowledges that the Extended Data Packet Transmit States were set.

5.37. MSG:0xB9 - Set BioHarness User Config Item

5.37.1. Request Message

This message sets the specified BioHarness User Configuration Item to the value provided.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB9								Msg ID
2	2								DLC
3	Number of Configuration Item (see Table 5-)								Payload
4	Value to set for the Configuration Item (see Table 5-)								
7	CRC								CRC
8	ETX								ETX

Table 5-81 *Request:* Set BioHarness User Config Item

See Table 5- for a description of the configuration items which can be set.

5.37.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xB9								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK/NAK								ACK/NAK

Table 5-82 *Response:* Set BioHarness User Config Item

- There is no data payload within the message as it simply acknowledges that the configuration item was set to the given value.
- If the configuration item is not supported or the value cannot be set, NAK is returned.

5.38. MSG:0xBC - Set Accelerometer 100mg Packet Transmit State

5.38.1. Request Message

This message requests that the BioHarness enables or disables the Bluetooth transmission of the periodic Accelerometer 100mg Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBC								Msg ID
2	1								DLC
3	Transmission State								Payload
4	CRC								CRC
5	ETX								ETX

Table 5-83 ***Request: Set Accelerometer 100mg Packet Transmit State***

- This command enables (payload = 1) of disables (payload = 0) the Accelerometer 100mg Packet transmission. When enabled, the packet is transmitted periodically (every 400ms).

5.38.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBC								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-84 ***Response: Set Accelerometer 100mg Packet Transmit State***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the transmission state to the requested mode.

5.39. MSG:0xBD - Set Summary Data Packet Update Rate

5.39.1. Request Message

This message requests that the Bio Harness sets the Bluetooth transmission period of the Summary Data Packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBD								Msg ID
2	2								DLC
3	Packet Update Period – LS Byte								Payload
	Packet Update Period – MS Byte								
4	CRC								CRC
5	ETX								ETX

Table 5-85 ***Request: Set Summary Data Packet Update Rate***

- This command sets the transmission update rate of the Summary Data Packet in seconds. If set to 0, transmission of the packet is disabled.

5.39.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBD								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-86 ***Response: Set Summary Data Packet Update Rate***

- There is no data payload within the message as the packet merely acknowledges that the data was accepted and that the remote unit has set the update rate to the specified value.

5.40. MSG:0xBE - Set Subject Info Settings

5.40.1. Request Message

This message sets the subject info settings that the device uses for the ROG2 algorithm and possibly other algorithms.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBE								Msg ID
2	variable								DLC
3	Type				Version				Payload
4	Data Byte 0								
:	:								
:	Data Byte n								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-87 ***Request: Set Subject Info Settings***

- DLC: The DLC of the message will vary depending on how much configuration data is provided. The DLC of the message should be checked and a NAK returned if incorrect.
- Type: Defines the type of configuration data present in the message, can be one of:
  - 0 = Partial Subject Info (short version – all other items are reset to default)
  - 1 = Full Subject Info (long version)
  - 2-15 = Reserved for future use
- Version: Defines the version of configuration data for the particular type. This allows the data format of each type of configuration item to be changed in the future. For example it may be required to later add new configurable parameters or increase the range or resolution of certain parameters which can be achieved by defining a new version of a particular type. The firmware implementation must check the version number and return a NAK response if it does not support a particular version.
- Data Bytes: The format of the configuration data depends on the Type and Version field and is detailed in Table 5- below. Data items are always stored LSB first.

Type	Version	Data Offset	Size (bytes)	Range	Unit	Description
0 Partial Subject Info	1	0	1	0..240	bpm	Heart Rate High Idle Red Limit
		1	1	0..240	bpm	Heart Rate High Idle Orange Limit
		2	1	0..240	bpm	Heart Rate High Active Red Limit
		3	1	0..240	bpm	Heart Rate High Active Orange Limit
		4	1	0..240	bpm	Heart Rate Max
		5	2	0..70	0.1 bpm	Breathing High Idle Red Limit
		7	2	0..70	0.1 bpm	Breathing High Idle Orange Limit
		9	2	0..70	0.1 bpm	Breathing High Active Red Limit
		11	2	0..70	0.1 bpm	Breathing High Active Orange Limit

		13	2	0..70	0.1 bpm	Breathing VT Threshold
0 Partial Subject Info	2	0	1	-	years	Age
		1	1	0..10	Unit less	Fitness Level
		2	1	0..240	bpm	Heart Rate High Idle Red Limit
		3	1	0..240	bpm	Heart Rate High Idle Orange Limit
		4	1	0..240	bpm	Heart Rate High Active Red Limit
		5	1	0..240	bpm	Heart Rate High Active Orange Limit
		6	1	0..240	bpm	Heart Rate Max
		7	2	0..70	0.1 bpm	Breathing High Idle Red Limit
		9	2	0..70	0.1 bpm	Breathing High Idle Orange Limit
		11	2	0..70	0.1 bpm	Breathing High Active Red Limit
		13	2	0..70	0.1 bpm	Breathing High Active Orange Limit
		15	2	0..70	0.1 bpm	Breathing VT Threshold
1 Full Subject Info	1	0	1	-	years	Age
		1	1	0..240	bpm	Heart Rate High Idle Red Limit
		2	1	0..240	bpm	Heart Rate High Idle Orange Limit
		3	1	0..240	bpm	Heart Rate High Active Red Limit
		4	1	0..240	bpm	Heart Rate High Active Orange Limit
		5	1	0..240	bpm	Heart Rate Low Idle Red Limit
		6	1	0..240	bpm	Heart Rate Low Idle Orange Limit
		7	1	0..240	bpm	Heart Rate Low Active Red Limit
		8	1	0..240	bpm	Heart Rate Low Active Orange Limit
		9	1	0..240	bpm	Heart Rate Max
		10	2	0..70	0.1 bpm	Breathing High Idle Red Limit
		12	2	0..70	0.1 bpm	Breathing High Idle Orange Limit
		14	2	0..70	0.1 bpm	Breathing High Active Red Limit
		16	2	0..70	0.1 bpm	Breathing High Active Orange Limit
		18	2	0..70	0.1 bpm	Breathing Low Idle Red Limit
		20	2	0..70	0.1 bpm	Breathing Low Idle Orange Limit
		22	2	0..70	0.1 bpm	Breathing Low Active Red Limit
		24	2	0..70	0.1 bpm	Breathing Low Active Orange Limit
		26	2	0..70	0.1 bpm	Breathing VT Threshold
		28	1	0..255	100 mg	Run Activity Threshold
		29	1	0..255	100 mg	High Jog Activity Threshold
		30	1	0..255	100 mg	Low Jog Activity Threshold
		31	1	0..255	100 mg	High Walk Activity Threshold
		32	1	0..255	100 mg	Low Walk Activity Threshold
		33	1	0..255	100 mg	High Idle Activity Threshold
		34	2	-180..+180	degrees	Max Supine Threshold
		36	2	-180..+180	degrees	Backwards Inverted Threshold
		38	2	-180..+180	degrees	Max Recline Threshold
		40	2	-180..+180	degrees	Min Supine Threshold
		42	2	-180..+180	degrees	Backwards Upright Threshold
		44	2	-180..+180	degrees	Min Recline Threshold
		46	2	-180..+180	degrees	Min Incline Threshold
		48	2	-180..+180	degrees	Forwards Upright Threshold
		50	2	-180..+180	degrees	Min Prone Threshold
		52	2	-180..+180	degrees	Max Incline Threshold
		54	2	-180..+180	degrees	Forwards Inverted Threshold
		56	2	-180..+180	degrees	Max Prone Threshold
1	2	0	1	-	years	Age

Full Subject Info		1	1	0..10	Unit less	Fitness Level
		2	1	0..240	bpm	Heart Rate High Idle Red Limit
		3	1	0..240	bpm	Heart Rate High Idle Orange Limit
		4	1	0..240	bpm	Heart Rate High Active Red Limit
		5	1	0..240	bpm	Heart Rate High Active Orange Limit
		6	1	0..240	bpm	Heart Rate Low Idle Red Limit
		7	1	0..240	bpm	Heart Rate Low Idle Orange Limit
		8	1	0..240	bpm	Heart Rate Low Active Red Limit
		9	1	0..240	bpm	Heart Rate Low Active Orange Limit
		10	1	0..240	bpm	Heart Rate Max
		11	2	0..70	0.1 bpm	Breathing High Idle Red Limit
		13	2	0..70	0.1 bpm	Breathing High Idle Orange Limit
		15	2	0..70	0.1 bpm	Breathing High Active Red Limit
		17	2	0..70	0.1 bpm	Breathing High Active Orange Limit
		19	2	0..70	0.1 bpm	Breathing Low Idle Red Limit
		21	2	0..70	0.1 bpm	Breathing Low Idle Orange Limit
		23	2	0..70	0.1 bpm	Breathing Low Active Red Limit
		25	2	0..70	0.1 bpm	Breathing Low Active Orange Limit
		27	2	0..70	0.1 bpm	Breathing VT Threshold
		20	1	0..255	100 mg	Run Activity Threshold
		30	1	0..255	100 mg	High Jog Activity Threshold
		31	1	0..255	100 mg	Low Jog Activity Threshold
		32	1	0..255	100 mg	High Walk Activity Threshold
		33	1	0..255	100 mg	Low Walk Activity Threshold
		34	1	0..255	100 mg	High Idle Activity Threshold
		35	2	-180..+180	degrees	Max Supine Threshold
		37	2	-180..+180	degrees	Backwards Inverted Threshold
		39	2	-180..+180	degrees	Max Recline Threshold
		41	2	-180..+180	degrees	Min Supine Threshold
		43	2	-180..+180	degrees	Backwards Upright Threshold
		45	2	-180..+180	degrees	Min Recline Threshold
		47	2	-180..+180	degrees	Min Incline Threshold
		49	2	-180..+180	degrees	Forwards Upright Threshold
		51	2	-180..+180	degrees	Min Prone Threshold
		53	2	-180..+180	degrees	Max Incline Threshold
		55	2	-180..+180	degrees	Forwards Inverted Threshold
		57	2	-180..+180	degrees	Max Prone Threshold
2-15 unused						

Table 5-88: Subject Info Data Formats

5.40.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBE								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-89 ***Response: Set Subject Info Settings***

- There is no data payload within the message as it simply acknowledges that the Subject Info Settings were stored.

5.41. MSG:0xBF - Get Subject Info Settings

5.41.1. Request Message

This message gets the subject info settings that the device uses for the ROG2 algorithm and possibly other algorithms.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBF								Msg ID
2	0								DLC
3	Type				unused				Payload
4	CRC								CRC
5	ETX								ETX

Table 5-90 ***Request: Get Subject Info Settings***

- Type allows the sender of the request to specify the type of configuration data to read. The current implementation only supports “Full Subject Info” type (1). All other types will return a NAK.
- Unused – These bits are currently not used and must be set to 0. In the future this field may be used to allow a specific version of an info structure to be requested.

5.41.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xBF								Msg ID
2	variable								DLC
3	Type				Version				Payload
4	Data Byte 0								
:	:								
:	Data Byte n								
DLC+3	CRC								CRC
DLC+4	ACK								ACK

Table 5-91 ***Response: Get Subject Info Settings***

- Type/Version byte identifies the format of configuration data.
- Data is sent for the given Type/Version according t87.
- If configuration data cannot be returned or the implementation does not support this configuration message, a NAK message is returned with no payload.

5.42. MSG:0xD0 - Set Remote MAC Address & PIN

5.42.1. Request Message

This message can be used to set multiple Bluetooth MAC address and PIN codes of remote devices that should be connected to. For Example, this message could be used to configure an RID to connect to a GPS/3DL device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD0								Msg ID
2	DLC								DLC
3	Remote Bluetooth Device Number								Payload
4	Bluetooth MAC address String (ASCII) – 1st Byte								
:	:								
20	Bluetooth MAC address String (ASCII) – 17th Byte								
21	Bluetooth PIN (ASCII) - 1st Byte								
:	:								
DLC+2	Bluetooth PIN (ASCII) - nth Byte								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-92 ***Request: Set Remote MAC Address & PIN***

- The Remote Bluetooth Device Number allows multiple Remote MAC address & PIN numbers to be stored in the device. Remote Device Number 0 is reserved so if the device only supports connecting to one remote device then this number should be 1.
- The MAC Address string and PIN are both in ASCII.
- The MAC Address has a fixed number of characters (17) e.g. "**01:23:45:67:89:AB**".
- There is no NULL terminator at the end of the string and therefore it is the responsibility of the receiving application to append a NULL to the string data.
- The Bluetooth PIN is an optional, variable length array of ASCII characters which is not null-terminated, its length can be determined from the DLC of the message.

5.42.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD0								Msg ID
2	0								DLC
3	CRC								CRC
4	ACK								ACK/NAK

Table 5-93 ***Response: Set Remote MAC Address & PIN***

- There is no data payload within the message as it simply acknowledges that the Remote MAC Address & PIN were stored.

5.43. MSG:0xD1 - Get Remote MAC Address & PIN

5.43.1. Request Message

This message returns the Bluetooth MAC address & PIN code of the requested remote Bluetooth Device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD1								Msg ID
2	1								DLC
3	Remote Bluetooth Device Number								Payload
4	CRC								CRC
5	ACK								ACK/NAK

Table 5-94 ***Request: Get Remote MAC Address & PIN***

- The Remote Bluetooth Device Number specifies which Remote MAC address & PIN should be returned. Remote Device Number 0 is reserved so if the device only supports connecting to one remote device then this number should be 1.

5.43.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD1								Msg ID
2	DLC								DLC
3	Remote Bluetooth Device Number								Payload
4	Bluetooth MAC address String (ASCII) – 1st Byte								
:	:								
20	Bluetooth MAC address String (ASCII) – 17th Byte								
21	Bluetooth PIN (ASCII) - 1st Byte								
:	:								
DLC+2	Bluetooth PIN (ASCII) - nth Byte								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 5-95 ***Response: Get Remote MAC Address & PIN***

- The MAC Address string and PIN are both in ASCII.
- The MAC Address has a fixed number of characters (17) e.g. "**00:07:80:82:7A:61**".
- There is no NULL terminator at the end of the string and therefore it is the responsibility of the receiving application to append a NULL to the string data.
- The Bluetooth PIN is an optional, variable length array of ASCII characters which is not null-terminated, its length can be determined from the DLC of the message.

5.44. MSG:0xD4 - Get Remote Device Description

5.44.1. Request Message

This message returns the Description of the requested remote Bluetooth Device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD4								Msg ID
2	1								DLC
3	Remote Bluetooth Device Number								Payload
4	CRC								CRC
5	ACK								ACK/NAK

Table 5-96 ***Request: Get Remote Device Description***

- The Remote Bluetooth Device Number specifies which Remote Device Description should be returned. Remote Device Number 0 is reserved so if the device only supports connecting to one remote device then this number should be 1.

5.44.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD4								Msg ID
2	DLC								DLC
	Remote Bluetooth Device Number								Payload
3	Remote Device Description (ASCII) – 1st Byte								
:									
:									
:									
:									
:									
DLC+3	Remote Device Description (ASCII) – Last Byte								
DLC+4	CRC								CRC
DLC+5	ACK								ACK/NAK

Table 5-97 ***Response: Get Remote Device Description***

- The Remote Device Description is sent as an ASCII string.
- There is no NULL terminator at the end of the string.
- The string has a maximum of 32 characters in length.
- If the remote Bluetooth Device Number is not supported by a device, a NAK message will be returned.

5.45. MSG:0xD5 - Get Supported Log Formats

5.45.1. Request Message

This message returns the list of log formats supported by the device.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD5								Msg ID
2	1								DLC
3	Log File Number								Payload
4	CRC								CRC
5	ACK								ACK/NAK

Table 5-98 ***Request: Get Supported Log Formats***

- The Log File Number specifies which log file to get the supported log formats for. This number is normally set to 0.

5.45.2. Response Message

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0xD5								Msg ID
2	DLC								DLC
3	Log File Number								Payload
4	Log Format 0 – LS Byte								
5	Log Format 0 – MS Byte								
:	Log Format 1 – LS Byte								
:	Log Format 1 – MS Byte								
:	:								
:	:								
DLC+2	Log Format N – LS Byte								
DLC+3	Log Format N – MS Byte								
DLC+4	CRC								CRC
DLC+5	ACK								ACK/NAK

Table 5-99 ***Response: Get Supported Log Formats***

- The Log File Number indicates which log file listed supported log formats are for.
- For every supported log format the log format version number is returned as a 4-digit BCD number. For example version 0.0.1.2 is represented as 0x0012.
- A NAK message will be returned if the device does not support getting supported log formats for the specified log file number

6. Periodic Data Packets

This section specifies packets which do not require acknowledgement and/or are sent on a periodic basis.

6.1. MSG:0x20 - General Data Packet

This message contains the General Data transmitted by the Bio Harness. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x20								Msg ID
2	53								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Heart Rate (0…240) – LS Byte								
13	Heart Rate (0…240) – MS Byte								
14	Respiration Rate (0…70) – LS Byte								
15	Respiration Rate (0…70) – MS Byte								
16	Skin Temperature (0…60) – LS Byte								
17	Skin Temperature (0…60) – MS Byte								
18	Posture (-180…180) – LS Byte								
19	Posture (-180…180) – MS Byte								
20	VMU (0…16) – LS Byte								
21	VMU(0…16) – MS Byte								
22	Peak Acceleration (0…16) – LS Byte								
23	Peak Acceleration (0…16) – MS Byte								
24	Battery Voltage								
26	Breathing Wave Amplitude								
28	ECG Amplitude†								
30	ECG Noise†								
32	Vertical Axis Acceleration Min								
34	Vertical Axis Acceleration Peak								
36	Lateral Axis Acceleration Min								
38	Lateral Axis Acceleration Peak								
40	Sagittal Axis Acceleration Min								
42	Sagittal Axis Acceleration Peak								
44	Zephyr System Channel								
46	GSR								
48	*unused*								
50	*unused*								
52	ROG								
53	ALARM								
54	Battery Status(see 6.2 below)								
55	Button/Worn(see 6.2 below)								
56	CRC								CRC
57	ETX								ETX

Table 6-1 ***Message: General Data Packet***

- Bluetooth BioHarness implementations of this message prior to v2.2.0.0 terminated this packet with an ACK instead of an ETX, Therefore it is recommended that any application which receives this message should handle both types of termination.

Also prior to this version, the periodic update rate of this packet is 1008ms instead of 1000ms

- Once enabled (by the “Set General Data Packet Transmission State” message), the packet is sent periodically to a remote unit (every 1000ms). General Data packet content details
- The physiological data contained in the packet is specified as follows:
    - Heart Rate: 0…240 with 1 unit resolution e.g. 132 = 132 BPM
        - Invalid Heart Rate: 65535 BPM
    - Respiration Rate: 0…70 with 0.1 unit resolution e.g. 173 = 17.3 BPM
        - Invalid Respiration Rate: 6553.5 BPM
    - Skin Temperature: 0…60 with 0.1 unit resolution e.g. 357 = 35.7 °C
        - Invalid Skin Temperature: -3276.8°C
    - Posture: -180…180 with 1 unit resolution e.g. -25 = -25
        - Invalid Posture: -32768°
    - Activity: 0…16 with 0.01 unit resolution e.g. 256 = 2.56 VMU
        - Invalid Activity: 655.35 VMU
    - Peak Acceleration: 0…16 with 0.01 unit resolution e.g. 679 = 6.79g
        - Invalid Peak Acceleration: 655.35g
    - Breathing Wave Amplitude: 0 – 65535 with resolution 1 LSB
        - Invalid Breathing Wave Amplitude : 65535
    - ECG Amplitude: 0…0.05 with 0.000001 unit resolution e.g 2376 = 0.002376V
        - Invalid ECG Amplitude: 0.065535V
    - ECG Noise: 0…0.05 with 0.000001 unit resolution e.g 1245 = 0.001245V
        - Invalid ECG Noise: 0.065535V
    - Acceleration Min: -16…16 with 0.01 unit resolution e.g. -83 = -0.83g
        - Invalid Acceleration Min: -327.68g
    - Acceleration Peak: -16…16 with 0.01 unit resolution e.g. 1225 = 12.25g
        - Invalid Acceleration Peak: -327.68g
    - GSR: 0…65534 with 1 unit resolution e.g. 10 = 10 nanoSiemens
        - Invalid GSR: 65535
    - ROG: (See Summary Data Packet description)
        - Invalid ROG: 0

- The following parameters describe the status of the device and of the connection over which the packet is sent:
    - Battery Voltage: 0…4.2 with 0.001 unit resolution e.g. 3904 = 3.904V
        - Invalid Battery Voltage: 65.535V
- All data items from byte index 12 (Heart Rate) to 54 (LED Status) are 16-bit signed numbers and are LS byte first.
- The sequence number increments by 1 for each transmission.
- If a particular parameter is not supported by the device, then an invalid value will be returned

Byte 55 below

15	14	13	12	11	10	9	8
PMWS	UIBS	BHHSL	BHESC	Unused			

Byte 54 below

7	6	5	4	3	2	1	0
Unused	BPFC						

PMWS Bit 15 Physiological Monitor Worn Status
0 Not worn by user
1 Worn by user

UIBS Bit 14 User Interface Button Status
0 Button not pressed
1 Button Pressed

BHHSL Bit 13 BioHarness Heart-Rate Signal Low
0 BioHarness Heart-Rate Signal Quality is acceptable
1 BioHarness Heart-Rate Signal Quality Low (HR coasting)

BHESC Bit 12 BioHarness External Sensors Connected
0 BioHarness has no external sensors connected
1 BioHarness has external sensors connected

Unused Bits 11-7 Unused

BPFC Bits 6-0 Battery status as Percentage of Full Charge
0 0 % (Battery Cut-off Voltage)
….
100 100% (Battery Fully Charged)

6.2. MSG:0x21 - Breathing Waveform Packet

This message contains the Breathing Waveform data transmitted by the Bio Harness. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x21								Msg ID
2	32								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Breathing Waveform Data (18 Samples) – see “Packing Format”								
35	CRC								CRC
36	ETX								ETX

Table 6-2 ***Message: Breathing Waveform Packet***

Note: Bluetooth BioHarness implementations of this message prior to v2.2.0.0 terminated this packet with an ACK instead of an ETX, Therefore it is recommended that any application which receives this message should handle both types of termination.

- Once enabled (by the “Set Breathing Waveform Packet Transmission State” message), the packet is sent periodically to a remote unit (every 1.008 seconds).
- Each Breathing Waveform sample is separated in time by 56ms.

6.2.1. Packing Format

Each Breathing Waveform sample is bit-packed into the message to minimise the amount of space used. The data is packed in the following format:

Byte/Bit	7	6	5	4	3	2	1	0
0								Bit 0
1							Bit 0	Bit 9
2						Bit 0	Bit 9	
3					Bit 0	Bit 9		
4	Bit 9							
5	As Byte 0 (pattern repeats every 5 bytes).							

Table 6-3 ***Packing Format for Breathing Waveform Data.***

- Because each sample is 10-bits in length, the data is bit-packed into the message to conserve space.

6.3. MSG:0x22 - ECG Waveform Packet

This message contains the ECG Waveform data transmitted by the Bio Harness. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x22								Msg ID
2	88								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	ECG Waveform Data (63 Samples) – see “Packing Format”								
91	CRC								CRC
92	ETX								ETX

Table 6-4 ***Message: ECG Waveform Packet***

Note: Bluetooth BioHarness implementations of this message prior to v2.2.0.0 terminated this packet with an ACK instead of an ETX, Therefore it is recommended that any application which receives this message should handle both types of termination.

- Once enabled (by the “Set ECG Waveform Packet Transmission State” message), the packet is sent periodically to a remote unit (every 252ms). Each ECG Waveform sample is 4ms later than the previous one.

6.3.1. Packing Format

Each ECG Waveform sample is bit-packed into the message to minimise the amount of space used in the message. The data is packed in the following format:

Byte/Bit	7	6	5	4	3	2	1	0
0								Bit 0
1						Bit 0	Bit 9	
2					Bit 0	Bit 9		
3			Bit 0	Bit 9				
4	Bit 9							
5	As Byte 0 (pattern repeats every 5 bytes).							

Table 6-5 ***Packing Format for ECG Waveform Data.***

- Because each sample is 10-bits in length, the data is bit-packed into the message to conserve space.

6.4. MSG:0x23 - Lifesign

This message can be sent from either side of a Bluetooth link to indicate to the remote unit that the link is still active (otherwise a timeout will occur).

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x23								Msg ID
2	0								DLC
3	CRC								CRC
4	ETX								ETX

Table 6-6 ***Message: Lifesign***

- The message has no data content.

6.5. MSG:0x24 - R to R Packet

This message contains the R to R data transmitted by the Bio Harness. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x24								Msg ID
2	45								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	R to R Data (18 Samples) – see “Packing Format”								
48	CRC								CRC
49	ETX								ETX

Table 6-7 ***Message: R to R Packet***

Note: Bluetooth BioHarness implementations of this message prior to v2.2.0.0 terminated this packet with an ACK instead of an ETX, Therefore it is recommended that any application which receives this message should handle both types of termination.

- Once enabled (by the “Set R to R Packet Transmission State” message), the packet is sent periodically to a remote unit (every 1.008 seconds).
- Each R to R sample is separated in time by 56ms.

6.5.1. Packing Format

The R to R samples are packed in the following format:

Byte/Bit	7	6	5	4	3	2	1	0
0	R to R Sample 0 (LS Byte)							
1	R to R Sample 0 (MS Byte)							
2	R to R Sample 1 (LS Byte)							
3	R to R Sample 1 (MS Byte)							
4	R to R Sample 2 (LS Byte)							
5	R to R Sample 2 (MS Byte)							
34	R to R Sample 17 (LS Byte)							
35	R to R Sample 17 (MS Byte)							

Table 6-8 ***Packing Format for R to R Data.***

- Each sample is 16-bits in length so the data is packed into 2 bytes (LS Byte first).

6.6. MSG:0x25 - Accelerometer Packet

This message contains the Accelerometer data transmitted by the Bio Harness. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x25								Msg ID
2	84								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Accelerometer Data (20 Sample Sets) – see “Packing Format”								
87	CRC								CRC
88	ETX								ETX

Table 6-9 ***Message: Accelerometer Packet***

Note: Bluetooth BioHarness implementations of this message prior to v2.2.0.0 terminated this packet with an ACK instead of an ETX, Therefore it is recommended that any application which receives this message should handle both types of termination.

- Once enabled (by the “Set Accelerometer Packet Transmission State” message), the packet is sent periodically to a remote unit (every 400ms). Each set of XYZ accelerometer samples is 20ms later than the previous one.

6.6.1. Packing Format

Each set of Accelerometer samples is bit-packed into the message to minimise the amount of space used in the message. The data is packed in the following format:

Byte/Bit	7	6	5	4	3	2	1	0
0								X-Bit 0
1						Y-Bit 0	X-Bit 9	
2				Z-Bit 0	Y-Bit 9			
3		X-Bit 0	Z-Bit 9					
4	X-Bit 9							
5								Y-Bit 0
6						Z-Bit 0	Y-Bit 9	
7				X-Bit 0	Z-Bit 9			
8		Y-Bit 0	X-Bit 9					
9	Y-Bit 9							
10								Z-Bit 0
11						X-Bit 0	Z-Bit 9	
12				Y-Bit 0	X-Bit 9			
13		Z-Bit 0	Y-Bit 9					
14	Z-Bit 9							
15	As Byte 0 (pattern repeats every 15 bytes).							

Table 6-10 ***Packing Format for Accelerometer Data.***

- Because each sample is 10-bits in length, the data is bit-packed into the message to conserve space.
- Note: the X/Y/Z axes correspond to the diagram below. The arrows indicate the direction for positive data values.

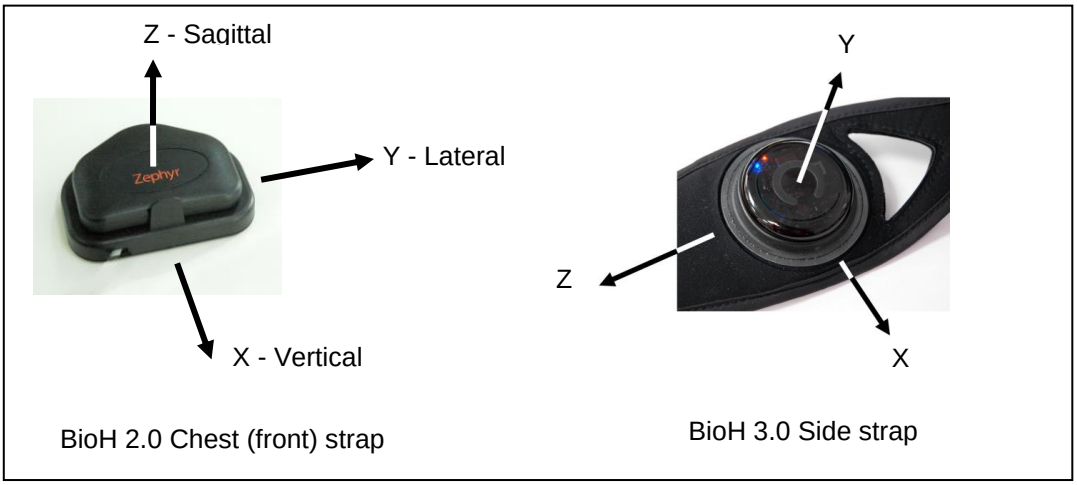

Fig. 6.11

6.7. MSG:0x27 – Bluetooth Device Data Packet

This message acts as a wrapper for data from third party Bluetoothdevices thus allowing the data to be sent between zephyr devices. A typical use is for the BioHarness to send this message containing data from a 3rd party Bluetooth device connected to the Bio Harness.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x27								Msg ID
2	DLC								DLC
3	Sequence Number (0…255)								Payload
4	Device Manufacturer Code (see Table 5-)								
5	Device Type Code (see Table 5-)								
6	Bluetooth MAC Address - 1st Byte								
:	:								
11	Bluetooth MAC Address - 6th Byte								
12	Status								
13	Bluetooth Device Data								
:	:								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 6-11 ***Message: Bluetooth peripheral Data Packet***

- The sequence number is incremented for every message sent
- The Bluetooth MAC address is a 6-byte MAC address in binary representation used to identify the device.
- **Status –** Indicates the status of the measurement
- **Bluetooth Device Data –** Data from the third party device. The length and contents of the data is specific to the type of device that the data originates from.

7	6	5	4	3	2	1	0
Unused				BDMS			

Table 6-12 Status Bits

Unused Bits 7-4 Unused

BDMS Bits 0-3 Bluetooth Device Measurement Status

0	Valid Measurement
1	Unreliable Measurement
2	Unable To Connect
3	Authentication Error
4	Communication Error
5	Failed Measurement
6-15	Reserved for future use

6.8. MSG:0x28 – Extended Data Packet

This message can be used to send additional information not contained in the general data packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x28								Msg ID
2	DLC								DLC
3	Packet Identifier								Payload
4	Sequence Number (0 .. 255)								
5	Packet Data								
:	:								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 6-13 ***Message: Extended Data Packet***

- **Packet Identifier:**

7	6	5	4	3	2	1	0
unused		**OWPKT**	**Data Type (0 .. 31)**				

Data Type Identifies the type of data this packet contains
OWPKT **O**ver-**W**riting **P**ac**k**e**t** (If set, packet should not be queued by RIDs)

- **Sequence Number** – message count, incremented separately for each data type
- **Packet Data –** Data encoded according to the Data Type, see below

Data Type		Packet Data		
Name	**Code**	**Byte**	**Description**	**Notes**
Heart Rate Recovery	0x00	0	Time Stamp (ms of day) LS Byte	
		1	:	
		2	:	
		3	Time Stamp (ms of day) MS Byte	
		4	Data Valid Flags (*See Table 6-15*)	
		5	HRR_{peak} – Peak Heart Rate after exercise	0 .. 240 bpm
		6	HRR_{30} – Heart Rate 30s after exercise	0 .. 240 bpm
		7	HRR_{60} – Heart Rate 60s after exercise	0 .. 240 bpm
		8	HRR_{120} – Heart Rate 120s after exercise	0 .. 240 bpm
		9	HRR_{180} – Heart Rate 180s after exercise	0 .. 240 bpm
Orthostatic Hypotension	0x01	0	Data Valid Flags (*See Table 6-16*)	
		1	Average HR over 60s period before standing	0 .. 240 bpm
		2	Instantaneous HR just before standing	0 .. 240 bpm
		3	Peak HR during first 15 seconds of standing	0 .. 240 bpm
		4	Average HR over 60s period after standing	0 .. 240 bpm
Vertical Jump Test	0x02	0	Time spent airborne - LS Byte	ms
		1	Time spent airborne - MS Byte	
		2	Peak Acceleration during Jump - LS Byte	10mg units
		3	Peak Acceleration during Jump - MS Byte	
Forty Yard Dash test	0x03	0	Peak Activity during Dash - LS Byte	10mg units
		1	Peak Activity during Dash - MS Byte	

Table 6-14 Extended Packet Data Types

- The Time Stamp for Heart Rate Recovery data is the same for all packets sent during one HRR test, so that the receiver can identify which HRR data packets belong to the same test, and when HRR data from the next HRR test is received.

7	6	5	4	3	2	1	0
Unused			HRR180	HRR120	HRR60	HRR30	HRRPK

HRR Data Valid Flags indicate which HR values in the packet are valid (1=valid):

HRRPK **H**eart **R**ate **R**ecovery **Pe**a**k** data valid
HRR30 **H**eart **R**ate **R**ecovery **30**s data valid
HRR60 **H**eart **R**ate **R**ecovery **60**s data valid
HRR120 **H**eart **R**ate **R**ecovery **120**s data valid
HRR180 **H**eart **R**ate **R**ecovery **180**s data valid

Table 6-15 HRR Data Valid Flags

7	6	5	4	3	2	1	0
Unused				HRLAV	HRLI	HRSPK	HRSAV

Orthostatic HR Data Valid Flags indicate which HR values in the packet are valid (1=valid):

HRLAV **H**eart **R**ate **L**ying **Av**erage (over 60s before standing) data valid
HRLI **H**eart **R**ate **L**ying **I**nstantaneous (before standing) data valid
HRSPK **H**eart **R**ate **S**tand **Pe**a**k** (15s after standing) data valid
HRSAV **H**eart **R**ate **S**tand **Av**erage (over 60s after standing) data valid

Table 6-16 Orthostatic HR Data Valid Flags

6.9. MSG:0x2A – Accelerometer 100mg Packet

This message contains 3-axis Accelerometer data transmitted in 100mg units. Once enabled, the packet is transmitted periodically.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x2A								Msg ID
2	84								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Accelerometer Data (20 Sample Sets) – see 6.6.1 for packing format								
87	CRC								CRC
88	ETX								ETX

Table 6-17 ***Message: Accelerometer 100mg Packet***

- Each sample is a signed 10-bit value in 100mg units. i.e. a value of 23 equates to +2.3g and a value of -11 equates to -1.1g.
- Once enabled (by the “Set Accelerometer 100mg Packet Transmit State” message), the packet is sent periodically to a remote unit (every 400ms). Each set of XYZ accelerometer samples is 20ms later than the previous one.

6.1. MSG:0x2B - Summary Data Packet

This message contains Summary Data which once enabled, is transmitted periodically at the rate specified by the remote device.

6.1.1. Version 2

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x2B								Msg ID
2	71								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Version Number = 2								
13	Heart Rate (0…240) – LS Byte								
14	Heart Rate (0…240) – MS Byte								
15	Respiration Rate (0…70) – LS Byte								
16	Respiration Rate (0…70) – MS Byte								
17	Skin Temperature (0…60) – LS Byte								
18	Skin Temperature (0…60) – MS Byte								
19	Posture (-180…180) – LS Byte								
20	Posture (-180…180) – MS Byte								
21	Activity (0…16) – LS Byte								
22	Activity(0…16) – MS Byte								
23	Peak Acceleration (0…16) – LS Byte								
24	Peak Acceleration (0…16) – MS Byte								
25	Battery Voltage - LS Byte								
26	Battery Voltage - MS Byte								
27	Battery Level (0…100)								
28	Breathing Wave Amplitude – LS Byte								
29	Breathing Wave Amplitude – MS Byte								
30	Breathing Wave Noise– LS Byte								
31	Breathing Wave Noise– MS Byte								
32	Breathing Rate Confidence (0…100)								
33	ECG Amplitude – LS Byte								
34	ECG Amplitude – MS Byte								
35	ECG Noise – LS Byte								
36	ECG Noise – MS Byte								
37	Heart Rate Confidence (0…100)								
38	Heart Rate Variability – LS Byte								
39	Heart Rate Variability – MS Byte								
40	System Confidence (0…100)								
41	GSR – LS Byte								
42	GSR – MS Byte								
43	ROG – LS Byte								

44	ROG – MS Byte	
45	Vertical Axis Acceleration Min – LS Byte	
46	Vertical Axis Acceleration Min – MS Byte	
47	Vertical Axis Acceleration Peak – LS Byte	
48	Vertical Axis Acceleration Peak – MS Byte	
49	Lateral Axis Acceleration Min – LS Byte	
50	Lateral Axis Acceleration Min – MS Byte	
51	Lateral Axis Acceleration Peak – LS Byte	
52	Lateral Axis Acceleration Peak – MS Byte	
53	Sagittal Axis Acceleration Min – LS Byte	
54	Sagittal Axis Acceleration Min – MS Byte	
55	Sagittal Axis Acceleration Peak – LS Byte	
56	Sagittal Axis Acceleration Peak – MS Byte	
57	Device Internal Temp – LS Byte	
58	Device Internal Temp – MS Byte	
59	Status Info LS Byte	
60	Status Info MS Byte	
61	Link Quality	
62	RSSI	
63	Tx Power	
64	Estimated Core Temperature- LS Byte	
65	Estimated Core Temperature- MS Byte	
66	Auxiliary ADC Channel 1 – LS Byte	
67	Auxiliary ADC Channel 1 – MS Byte	
68	Auxiliary ADC Channel 2 – LS Byte	
69	Auxiliary ADC Channel 2 – MS Byte	
70	Auxiliary ADC Channel 3 – LS Byte	
71	Auxiliary ADC Channel 3 – MS Byte	
72	Reserved – LS Byte	
73	Reserved – MS Byte	
74	CRC	CRC
75	ETX	ETX

.

Table 6-18 ***Message: Summary Data Packet (Version 2)***

- The length of the data packet is indicated by the DLC field. Different versions of the packet may have different lengths. Therefore to maintain compatibility it is important to read the DLC so that the CRC & ETX fields are correctly located within the message.
- The version number is in the range 1…255. If there are any parameters that are only supported or have a different meaning in some versions of the message, these are detailed in this document.
- The update period of the packet is set by the MSG:0xBD - Set Summary Data Packet Update Rate. This update rate is also used as the epoch over which data within the packet is calculated. This only relates to Activity, Peak Acceleration & min + peak accelerations for each axis. All other data within the packet is the latest value for that particular parameter at the time of transmission.
- The sequence number is incremented for every message sent.
- For each parameter in the message, there is a value that indicates no valid data which is used if the parameter is not supported by a particular device or just not available at a particular time. Therefore if the invalid value is received for a particular

parameter, this value should not be displayed as the value but instead it should be indicated that there is no data available for this parameter. The invalid values are described below.

- The physiological data contained in the packet is specified as follows:
  - Heart Rate: 0…240 with 1 unit resolution e.g. 132 = 132 BPM
    - Invalid Heart Rate: 65535 BPM
  - Respiration Rate: 0…70 with 0.1 unit resolution e.g. 173 = 17.3 BPM
    - Invalid Respiration Rate: 6553.5 BPM
  - Skin Temperature: 0…60 with 0.1 unit resolution e.g. 357 = 35.7 °C
    - Invalid Skin Temperature: -3276.8°C
  - Posture: -180…180 with 1 unit resolution e.g. -25 = -25
    - Invalid Posture: -32768°
  - Activity: 0…16 with 0.01 unit resolution e.g. 256 = 2.56 VMU
    - Invalid Activity: 655.35 VMU
  - Peak Acceleration: 0…16 with 0.01 unit resolution e.g. 679 = 6.79g
    - Invalid Peak Acceleration: 655.35g
  - Breathing Wave Amplitude:
    - Invalid Breathing Wave Amplitude : 65535
  - Breathing Wave Noise:
    - Invalid Breathing Wave Noise: 65535
  - Breathing Rate Confidence: 0…100 with 1 unit resolution e.g. 95 = 95%
    - Invalid Breathing Rate Confidence: 255
  - ECG Amplitude: 0…0.05 with 0.000001 unit resolution e.g 2376 = 0.002376V
    - Invalid ECG Amplitude: 0.065535V
  - ECG Noise: 0…0.05 with 0.000001 unit resolution e.g 1245 = 0.001245V
    - Invalid ECG Noise: 0.065535V
  - Breathing Rate Confidence: 0…100 with 1 unit resolution e.g. 95 = 95%
    - Invalid Breathing Rate Confidence: 255
  - Heart Rate Confidence: 0…100 with 1 unit resolution e.g. 97 = 97%
    - Invalid Heart Rate Confidence: 255
  - Heart Rate Variability:
    - Invalid Heart Rate Variability: 65535
  - System Confidence: 0…100 with 1 unit resolution e.g. 83 = 83%
    - Invalid System Confidence: 255
  - GSR: 0…65534 with 1 unit resolution e.g. 10 = 10 nanoSiemens
    - Invalid GSR: 65535
  - ROG: (See *Table 6-20*)
    - Invalid ROG: 0
  - Acceleration Min: -16…16 with 0.01 unit resolution e.g. -83 = -0.83g
    - Invalid Acceleration Min: -327.68g
  - Acceleration Peak: -16…16 with 0.01 unit resolution e.g. 1225 = 12.25g
    - Invalid Acceleration Peak: -327.68g
  - Estimated Core Temperature:33^0 C to 41^0 C with 0.1^0 C resolution e.g. 386 = 38.6^0 C
    - Invalid Core Temperature: 6553.5^0 C
  - Auxiliary ADC Channels: possible range 0…65534, actual range/unit is implementation-specific
    - Invalid Auxiliary ADC Channels: 65535

- The following parameters describe the status of the device and of the connection over which the packet is sent:
  - Battery Voltage: 0…4.2 with 0.001 unit resolution e.g. 3904 = 3.904V
    - Invalid Battery Voltage: 65.535V

- Battery Level: 0…100 with 1 unit resolution e.g. 65 = 65%
  - Invalid Battery Level: 255
- Device Internal Temp: 0…100 with 0.1 unit resolution e.g. 612 = 61.2°C
  - Invalid Device Internal Temp: -3276.8°C
- Status Info: (See Table 6-21)
  - Invalid Status Info: 0
- Link Quality: 0…254, unitless number where higher number = better quality
  - Invalid Link Quality: 255
- RSSI: -127…127 with 1 unit resolution e.g. -5 = -5dB
  - Invalid RSSI: -128
- Tx Power: -30…20 with 1 unit resolution e.g. 10 = +10dBm
  - Invalid Tx Power: -128dBm

7	6	5	4	3	2	1	0
Time (Bits 0-4)					Status		

Table 6-19 ROG LS Byte

15	14	13	12	11	10	9	8
Time (Bits 5-12)							

Table 6-20 ROG MS Byte

Status	Bits	ROG Status	
	0-2	000	Invalid
		001	Green
		010	Orange
		011	Red
		1xx	Reserved
Time	Bits	Time in current state	
	2-15	0 – 8192 seconds	

7	6	5	4	3	2	1	0
POUF	STUF	RRUF	HRUF	NFTG	BPDF	DWDL	

Table 6-21 *Status Info LS Byte*

15	14	13	12	11	10	9	8
Unused					ECTUF	HRVUF	ACUF

Table 6-22 Status Info MS Byte

DWDL	Bits	Device Worn Detection Level	
	0-1	00	Full Confidence Device is Worn
		01	High Confidence Device is Worn
		10	Low Confidence Device is Worn
		11	No Confidence Device is Worn

BPDF	Bit 2	Button Press Detection Flag 0 No Button Press Detected 1 Button Press Detected
NFTG	Bit 3	Not Fitted To Garment 0 Device Fitted To Garment 1 Device Not Fitted To Garment
HRUF	Bit 4	Heart Rate Unreliable Flag 0 Reported Heart Rate is Reliable 1 Reported Heart Rate is Unreliable
RRUF	Bit 5	Respiration Rate Unreliable Flag 0 Reported Respiration Rate is Reliable 1 Reported Respiration Rate is Unreliable
STUF	Bit 6	Skin Temperature Unreliable Flag 0 Reported Skin Temperature is Reliable 1 Reported Skin Temperature is Unreliable
POUF	Bit 7	Posture Unreliable Flag 0 Reported Posture is Reliable 1 Reported Posture is Unreliable
ACUF	Bit 8	Activity Unreliable Flag 0 Reported Activity is Reliable 1 Reported Activity is Unreliable
HRVUF	Bit 9	Heart Rate Variability Unreliable Flag 0 Reported Heart Rate Variability is Reliable 1 Reported Heart Rate Variability is Unreliable
ECTUF	Bit 10	Estimated Core Temperature Unreliable Flag 0 Reported Estimated Core Temperature is Reliable 1 Reported Estimated Core Temperature is Unreliable
Unused	Bits 15-11	Unused

6.2. MSG:0x2C – Event Packet

This message is used for transmitting events in a data packet.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	0x2C								Msg ID
2	DLC								DLC
3	Sequence Number (0…255)								Payload
4	Timestamp – Year (LS Byte)								
5	Timestamp – Year (MS Byte)								
6	Timestamp – Month								
7	Timestamp – Day								
8	Timestamp – Milliseconds of day (LS Byte)								
9	:								
10	:								
11	Timestamp – Milliseconds of day (MS Byte)								
12	Event Code LS Byte								
13	Event Code MS Byte								
:	Optional Event-Specific Data First Byte								
:	:								
:	Option Event-Specific Last Byte								
DLC+3	CRC								CRC
DLC+4	ETX								ETX

Table 6-23 ***Message: Event Packet***

- The sequence number is incremented for every message sent
- The Time Stamp indicates when the event happened, rather than when it was transmitted
- The DLC should be used to determine if any event-specific data exists
- For the Event Codes and event-specific data, refer to **Error! Reference source not found.**

7. Fixed Message Types

7.1. NAK Response

This message is sent back to the request unit when either:

- The DLC value in the received request message didn't match the number of bytes in the data payload field.
- The received CRC and the calculated CRC didn't match.
- The ETX was not received/the value in the ETX field was incorrect.
- The data within the received payload was invalid e.g. RTC day was > 31.
- A failure to process the command occurred e.g. the Log could not be read.

Byte/Bit	7	6	5	4	3	2	1	0	Field
0	STX								STX
1	Message ID								Msg ID
2	0								DLC
3	CRC								CRC
4	NAK								ACK/NAK

Table 7-1 ***Response: NAK***

- The value in the Msg ID field is the same as the Msg ID in the request message.
- There is no data payload within the message as the packet merely indicates that the request failed.

8. Appendix A – Overview of Messages

Do to hardware differences between BioH v2.0 and v3.0, some messages are only supported by on version of the device.

Msg ID	Message	Description	BH 2.0	BH 3.0
0x01	Read Logging Data	Retrieve logging data	•	•
0x02	Delete Logging File	Delete logging file	•	•
0x07	Set RTC	Set the Date/Time	•	•
0x08	Get RTC	Get the Date/Time used	•	•
0x09	Get Bootloader Version Number	Get Firmware Boot Version	•	•
0x0A	Get Application Version Number	Get Firmware App Version	•	•
0x0B	Get Unit Serial Number	Get Firmware Serial Number	•	•
0x0C	Get Hardware Part Number	Get Firmware HW Part Number	•	•
0x0D	Get Bootloader Part Number	Get Firmware Boot Number	•	•
0x0E	Get Application Part Number	Get Firmware App Number	•	•
0x10	Set Network ID	Set the Bluetooth Network ID	•	•
0x11	Get Network ID	Get the Bluetooth Network ID	•	•
0x12	Get Unit MAC Address	Get the Bluetooth MAC address	•	•
0x14	Set General Data Packet Transmit State	Enable/Disable packet	•	•
0x15	Set Breathing Waveform Pkt Transmit State	Enable/Disable packet	•	•
0x16	Set ECG Waveform Packet Transmit State	Enable/Disable packet	•	•
0x17	Get Unit Bluetooth Name	Get Bluetooth Friendly Name	•	•
0x19	Set R to R Data Packet Transmit State	Enable/Disable packet	•	•
0x1E	Set Accelerometer Packet Transmit State	Enable/Disable packet	•	•
0x1F	Reboot Unit	Restart Code	•	•
0x20	General Data (streaming) Packet	No ACK required	•	•
0x21	Breathing Waveform (streaming) Packet	No ACK required	•	•
0x22	ECG Waveform (streaming) Packet	No ACK required	•	•
0x23	Lifesign Packet	No ACK required	•	•
0x24	R to R Data (streaming) Packet	No ACK required	•	•
0x25	Accelerometer Data (streaming) Packet	No ACK required	•	•
0x27	Bluetooth peripheral Data Packet	No ACK required	•	•
0x28	Extended Data Packet	No ACK required	•	•
0x2A	Accelerometer 100mg Data (streaming) Packet	No ACK required	•	•

0x2B	Summary Data (streaming) Packet	No ACK required	•	•
0x2C	Event Packet	No ACK required	•	•
0x9B	Set ROG Settings	Set the ROG algorithm thresholds	•	•
0x9C	Get ROG Settings	Get the ROG algorithm thresholds	•	•
0xA2	Set Bluetooth User Config	Set Bluetooth Module config	•	•
0xA3	Get Bluetooth User Config	Get Bluetooth Module config	•	•
0xA4	Set BT Link Config	Set Bluetooth Link config	•	•
0xA5	Get BT Link Config	Get Bluetooth Link config	•	•
0xA6	Set BioHarness User Config	Set BioHarness User Config	•	•
0xA7	Get BioHarness User Config	Get BioHarness User Config	•	•
0xAC	Get Battery Status	Current Battery Voltage & Charge	•	•
0xB0	BlueTooth Peripheral Message	For handling Bluetooth peripherals	•	•
0xB3	Reset Configuration	Resets device configuration	•	•
0xB4	Set Accelerometer Axis Mapping	Set mapping of accelerometer axes	•	•
0xB5	Get Accelerometer Axis Mapping	Get mapping of accelerometer axes	•	•
0xB6	Set Algorithm Config	Set configuration of an algorithm	•	•
0xB7	Get Algorithm Config	Get configuration of an algorithm	•	•
0xB8	Set Extended Data Packet Transmit State	Set transmit state of extended data packets	•	•
0xB9	Set BioHarness User Config Item	Set BioHarness User Config Item	•	•
0xBC	Set Accelerometer 100mg Data Packet Transmit State	Enable/Disable packet	•	•
0xBD	Set Summary Data Packet Update Rate	Set Packet Update period	•	•
0xBE	Set Subject Info Settings	Set configuration for subject	•	•
0xBF	Get Subject Info Settings	Get configuration for subject	•	•
0xD0	Set Remote MAC Address & PIN	Set Remote Device to call		•
0xD1	Get Remote MAC Address & PIN	Get Remote Device to call		•
0xD4	Get Remote Device Description	Get Description of Device to call		•
0xD5	Get Supported Log Formats	Get list of supported log formats		•

Table 8-1 Message Overview