# LoRa Peer-to-Peer Protocol

*Farmbeats*

**Document Number 2505F03**
**Revision 5**

**Prepared for Microsoft**
**Date:** 2-Jul-2017, 9:05 AM

## *Document Change Notes*

| Rev | Description | Date | Changed by |
|---|---|---|---|
| 1 | Initial release | 5/11/2017 | John Sawyer |
| 2 | Add Health message, add LoRa configuration details | 5/24/2017 | John Sawyer |
| 3 | Corrected timestamp field information | 5/26/2017 | John Sawyer |
| 4 | Corrected frequency value | 6/20/2017 | John Sawyer |
| 5 | Add two readings message, India radio settings | 7/2/2017 | John Sawyer |

## TABLE OF CONTENTS

# 1.0 INTRODUCTION

This document describes the LoRa radio configuration and the Peer-to-Peer protocol used by the Microsoft Farmbeats project for messages transmitted using the LoRa radio.

# 2.0 REFERENCES

[1] - DOT Series AT Command Reference Guide, Version 2.2, Multi-Tech Systems Inc. (S000643.pdf)

# 3.0 LORA CONFIGURATION

The LoRa radio will be configured to use Peer-to-Peer communications as described in reference document [1]. ACKs at the RF layer will not be enabled, as ACKs at the application layer are being used.

In Peer-to-Peer mode, all LoRa radios in the same network will be assigned the same network address, network session key and data session key. Different networks will have different addresses and keys to prevent cross-communication.

TX Data Rate DR1 will be used. DR1 supports a maximum payload of 53 bytes, which is large enough to support the Farmbeats message sizes. It also uses a slower data rate, to help with long range operation.

The LoRa radio will be configured prior to being deployed, and the network settings saved in permanent storage within the LoRa radio module. Thus, neither the device firmware nor the server app will need to perform any LoRa configuration in the field. The AT command sequence used to configure the LoRa radio will be:

```
AT+NJM=3
AT+ACK=0
AT+NA=4D734662                               [unique per network]
AT+NSK=89ABC89ABC89ABC89ABC89ABC89ABC34     [unique per network]
AT+DSK=45678904567890456789045678904567     [unique per network]
AT+TXDR=DR1
AT+TXF=916500000   [unique per network, between 915500000 and 919700000]
AT&W
ATZ
```

The above commands are entered one time only during factory programming. After each power cycle of the modem, the following command places the modem into Peer-to-Peer mode:

```
AT+SD
```

## 3.1 India Settings

The 868MHz radio module to be used in India will use a different radio configuration than the 915MHz radio used in the US.  The AT command sequence used to configure the LoRa radio will be:

```
AT+NJM=3
AT+ACK=0
AT+NA=4D734662                               [unique per network]
AT+NSK=89ABC89ABC89ABC89ABC89ABC89ABC34      [unique per network]
AT+DSK=45678904567890456789045678904567      [unique per network]
AT+TXDR=DR3
AT+TXP=11
AT+TXF=866300000                             [unique per network]
AT&W
ATZ
```

The above commands are entered one time only during factory programming.  After each power cycle of the modem, the following command places the modem into Peer-to-Peer mode:

```
AT+SD
```

# 4.0 MESSAGE PROTOCOL

## 4.1 SAMPLE DATA

### 4.1.1 Transmission from Farmbeats to Server – One Data Record

This message will send sample data after each set of samples is taken.

| STX | SerNo | MsgType | Timestamp | Data | ETX |
|---|---|---|---|---|---|
| 1 byte | 4 bytes | 1 byte | 4 bytes | 16 bytes | 1 byte |

Where:

STX = Start of Message = 0x02
SerNo = 4-byte Serial Number of Farmbeats device [see Section 4.3.1 for more detail]
MsgType = 0xA1 for Sample Data
Timestamp = 4-byte unix timestamp (little endian) [see Section 4.3.2 for more detail]
Data = 8 2-byte ADC values (little endian) [see Section 4.3.3 for more detail]
ETX = End of Message = 0x03

### 4.1.2 Transmission from Farmbeats to Server – Two Data Records

In the India implementation, the LoRa module imposes a duty cycle between transmissions. This means that the RF retry logic has need to be reworked. To support more robust operations in India, this new “two data records” message has been introduced.

This message will send sample data after each set of samples is taken, if more than one data record is stored in the Farmbeats device.

| STX | SerNo | MsgType | TStamp1 | Data1 | TStamp2 | Data2 | ETX |
|---|---|---|---|---|---|---|---|
| 1 byte | 4 bytes | 1 byte | 4 bytes | 16 bytes | 4 bytes | 16 bytes | 1 byte |

Where:

STX = Start of Message = 0x02
SerNo = 4-byte Serial Number of Farmbeats device [see Section 4.3.1 for more detail]
MsgType = 0xA2 for Two Data Records Sample Data
TStamp1 = 4-byte unix timestamp of most recent record (little endian) [see Section 4.3.2 ]
Data1 = 8 2-byte ADC values of most recent record (little endian) [see Section 4.3.3 ]
TStamp2 = 4-byte unix timestamp of 2nd most recent record (little endian) [see Section 4.3.2 ]
Data2 = 8 2-byte ADC values of 2$^{nd}$ most recent record (little endian) [see Section 4.3.3 ]
ETX = End of Message = 0x03

### 4.1.3 Transmission from Server to Farmbeats

This message will acknowledge the receipt of the sample data from the Farmbeats device. Additionally, this message can update the Farmbeats time/date information, can change the Farmbeats reporting interval, or can reset the Farmbeats device.

| STX | SerNo | MT | TS | Rst | Hlth | Intvl | IntVal | TD | TDVal | ETX |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 byte | 4 bytes | 1 byte | 4 bytes | 1 byte | 1 byte | 1 byte | 2 bytes | 1 byte | 4 bytes | 1 byte |

Where:

STX = Start of Message = 0x02
SerNo = 4-byte Serial Number of Farmbeats device from the sample data message [4.3.1 ]
MT = 0xA2 for Sample Data acknowledgement
TS = 4-byte unix timestamp from the sample data message (little endian) [see Section 4.3.2 ]
Rst = 1 to instruct Farmbeats device to reset, 0 otherwise
Hlth = 1 to request Health Report to be sent, 0 otherwise
Intvl = 1 to instruct Farmbeats device to change interval value
IntVal = new 2-byte interval value (in 10msec intervals) [see Section 4.3.4 for more detail]
TD = 1 to instruct Farmbeats device to update time/date
TDVal = new 4-byte unix timestamp [see Section 4.3.5 for more detail]
ETX = End of Message = 0x03

## 4.2 Health Report

### 4.2.1 Transmission from Farmbeats to Server

This message will send health report when requested by a sample data acknowledgement.

| STX | SerNo | MsgType | Timestamp | Data | ETX |
|---|---|---|---|---|---|
| 1 byte | 4 bytes | 1 byte | 4 bytes | 9 bytes | 1 byte |

Where:

STX = Start of Message = 0x02
SerNo = 4-byte Serial Number of Farmbeats device [see Section 4.3.1 for more detail]
MsgType = 0xA5 for Sample Data
Timestamp = 4-byte unix timestamp (little endian) [see Section 4.3.2 for more detail]
Data = 9 data bytes [see Section 4.3.6 for more detail]
ETX = End of Message = 0x03

### 4.2.2 Transmission from Server to Farmbeats

This message will acknowledge the receipt of the health report from the Farmbeats device.

| STX | SerNo | MsgType | Timestamp | ETX |
|---|---|---|---|---|
| 1 byte | 4 bytes | 1 byte | 4 bytes | 1 byte |

Where:

STX = Start of Message = 0x02
SerNo = 4-byte Serial Number of Farmbeats device from the health report message [4.3.1 ]
MsgType = 0xA6 for Health Report acknowledgement
Timestamp = 4-byte unix timestamp from the health report message (little endian) [4.3.2 ]
ETX = End of Message = 0x03

## 4.3 DATA FIELDS

### 4.3.1 Serial Number Field

4 byte (big endian) data field indicating serial number of Farmbeats device.

Example:

Serial Number: “123J”

Serial Number Field Values: 0x31 0x32 0x33 0x4A

### 4.3.2 Timestamp Field

4 byte (little endian) data field indicating timestamp of message.

Example:

Unix time value 0x02 (1/1/1970 00:00:02)

Timestamp Field Values: 0x02 0x00 0x00 0x00

### 4.3.3 Sample Data Data Field

8 2-byte ADC values, where each ADC value is little endian.

Example:

ADC0 = 0x0CCC, ADC1=0x0100, ADC2 = 0x0456, ADC3-ADC7 = 0x0000

Sample Data Data Field = 0xCC 0x0C   0x00 0x01   0x56 0x04   0x00 0x00 … 0x00 0x00

### 4.3.4 Sample Data Acknowledgement Interval Value Field

16-bit (little endian) Interval value.

Example:

Interval = 600 (0x258)

Sample Data Ack Interval Value Field = 0x58 0x02

### 4.3.5 Sample Data Acknowledgement Time/Date Value Field

4 byte (little endian) data field indicating timestamp.

Example:

Unix time value 0x02 (1/1/1970 12:00:02) ????

Timestamp Field Values: 0x02 0x00 0x00 0x00

### 4.3.6 Health Report Data Field

Data from Health Report structure. 16-bit values are little endian. The structure is defined as:

```
uint8_t powerCycleCount;
uint8_t watchdogResetCount;
uint8_t otherResetCount;
uint16_t loRaResendCount;
uint16_t loRaWatchdogCount;
uint16_t maximumMessageCount;
```

Example:

Power Cycle Count = 1

Watchdog Reset Count = 3

Other Reset Count = 0

LoRa Resend Count = 100 (0x64)

LoRa Watchdog Count = 50 (0x32)

Maximum Message Count = 10 (0x0A)

Health Report Data Field Values: 0x01 0x03 0x00 0x64 0x00 0x32 0x00 0x0A 0x00