Revision	Date	By	Changes
A	May 2023	SGE	Initial release based on former Xsens Dot BLE Service Specification (XD0506P). Movella legal entity changed. Added proprietary DFU Service UUID. Improved recording message specification. Added configuration message specification.
B	May 2023	MVE	Updated Message Service specification to include requesting information about available filter profiles.
C	July 2023	WWI	Updated Device Report characteristic button callback to include the double and triple button press

Movella™

Document XD0506P, Revision C, July 2023

Movella DOT BLE Service Specification

Bluetooth Service and API specification for Movella DOT

Table of Contents

List of references

1. Software Downloads | Movella.com Movella DOT online resources (Documentation and Software)
2. Dot Server Example | Base Movella DOT Bluetooth example implementation

List of Tables

List of Figures

1 Introduction

This document gives detailed Bluetooth services and characteristics specifications of the Movella DOT sensor. You can refer to this document to build your applications in any platform that supports Bluetooth. Movella offers SDK's as well for many different platforms. Table 1 shows the Bluetooth requirements of the host device.

Table 1: Bluetooth requirements

Bluetooth requirement
• Best performance with BLE 5.2, DLE supported • Compatible with Bluetooth 4.2 and up.

To get started, Movella recommends reading the *Movella DOT User Manual* first to familiarize with the basic functions of the sensor. After that, this document details the Bluetooth interface of the Movella DOT. In addition to this document there is example code available: DOT Server. It is a demonstration built using Node.js in combination with Noble.

Movella also provides Software Development Kits (SDK) for users to interact with Movella DOT sensors. The SDKs eliminate the need to interact with the Bluetooth device directly. Their role is to implement the Bluetooth functionality described in this document. SDKs are available for iOS, Android, Windows and Linux, more can be found in the online resources or on Base.

1.1 Base UUID

All attributes, i.e., services and characteristics of Movella DOT have a UUID that is formatted as a hexadecimal number:

1517**xxxx**-4947-11E9-8646-D663BD873D93

The bold characters are those that differ between the attributes. As such, the short notation of the UUID can be used. Refer to Table 2 for the available services and its UUID.

Table 2: BLE services and characteristics

Service	UUID	Description
Configuration Service	0x1000	Sensor information and configuration settings.
Measurement Service	0x2000	Configuring and receiving data of real-time streaming.
Battery Service	0x3000	Charging status and battery level.
DFU Service	0x4000	Device Firmware Update service. Not for public use.
Message Service	0x7000	Shared service for recording and synchronization functions.

Note – DFU service is not further specified in this document. Firmware update is possible via SDKs or mobile apps.

1.2 Scanning and Filtering

Movella Technologies B.V. is an Adopter Member of Bluetooth SIG. Refer to Table 3 for the Bluetooth company identifier of Movella.

Table 3 Movella Bluetooth company identifier

Company	Decimal	Hexadecimal
Movella Technologies B.V.	2182	0x0886

You can use the company ID together with the Bluetooth advertisement device name ”Movella DOT” to find and connect to the Movella DOT sensors. Movella DOT sensors have MAC address in the D4:22:CD:XX:XX:XX range.

1.3 Endianness

All the members are set in little-endian.

1.4 Firmware Compatibility

All the services and characteristics are applicable to the latest firmware version. Table 4 shows the supported hardware, SDK and App versions regarding the firmware version. It also shows the BLE advertisement device name which is useful when scanning for DOT sensors.

Movella has released the 2nd generation of DOT hereinafter referred to as v2. Refer to the DOT v2 Product Change Notification for the detailed changes of the introduction of the v2. The firmware supports both v1 and v2 sensors unless otherwise mentioned. All functions in this document are applicable to v1 and v2.

Table 4: Firmware compatibility

Firmware	BLE Device Name	Hardware	SDK and Mobile Apps
2.x.x	Movella Dot	Both v1 and v2	
2.4.0	Xsens Dot	Both v1 and v2	v2023.0
2.2.1	Xsens Dot	Both v1 and v2	v2022.0
2.1.0	Xsens Dot	Both v1 and v2	v2021.0
2.0.0	Xsens Dot	Both v1 and v2	v2021.0
1.6.0	Xsens Dot	v1	v2020.4
1.4.0	Xsens Dot	v1	v2020.2
1.3.0 (beta)	Xsens Dot	v1	v2020.1
1.0.0	Xsens Dot	v1	v2020.0, v2020.1, v2020.2

Refer to the firmware changelogs for all the changes between different firmware versions.

Note – It’s always recommended to use the latest firmware, SDK and App versions for new features and improvements.

2 Configuration Service

Configuration service provides sensor information such as Bluetooth identity address, firmware version and serial number, as well as controlling the sensor configurations.

The UUID of this service is **0x1000** and relevant characteristics are given in Table 5

Table 5: Characteristics of the Configuration Service

Characteristic	UUID	Description	Length	Property
Device info	0x1001	Sensor basic information	34	Read
Device control	0x1002	Sensor behavior and configurations	32	Read, Write
Device report	0x1004	Return sensor status or control command results	36	Notify

2.1 Device Info Characteristic

The device info characteristic is a 34-byte read-only data structure with the fields as specified in Table 6.

Table 6: Device info characteristic structure

Field name	Size	Description	Values
MAC Address	6	Bluetooth identity address	BD_ADDR, 6 byte MAC
Version Major	1	Firmware major version	0 ~ 255, uint8
Version Minor	1	Firmware minor version	0 ~ 255, uint8
Version Revision	1	Firmware revision version	0 ~ 255, uint8
Build Year	2	Firmware build year	2019 ~ 2100, uint16
Build Month	1	Firmware build month	1 ~ 12, uint8
Build Date	1	Firmware build date	1 ~ 31, uint8
Build Hour	1	Firmware build hour	0 ~ 23, uint8
Build Minute	1	Firmware build minute	0 ~ 59, uint8
Build Second	1	Firmware build second	0 ~ 59, uint8
SoftDevice version	4	Compatible BLE SoftDevice version	uint32
Serial Number	8	Serial number	uint64
Short Product Code	6	First 6 characters of product code	“XS-T01” - DOT v1 “XS-T02” - DOT v2

2.2 Device Control Characteristic

The device control characteristic is a 32-bytes data structure with the fields as specified in Table 7.

The Device Control Characteristic is used to select output rate and filter profile for measurement and recording. The filter profiles available on the Movella DOT are shown in Table 8. Refer to the *Movella DOT User Manual* for detailed information about the filter profiles.

Table 7: Device control characteristic structure

Field name	Size	Description	Values
Visit Index	1	One-byte bitmask. b0: Identifying b1: Power off b2: Power saving timeout b3: Tag name b4: Output rate b5: Filter profile index	Set the bit to 1 to enable the corresponding function. Set the bit to 0 to ignore.
Identifying	1	Sensor will be identified if it's set to "0x01". The sensor LED will fast blink 8 times and then a short pause in red, lasting for 10 seconds.	0x01: identify the sensor
Power off and power on options	1	Power off – shut down the sensor. Power on options – This setting is only available in DOT v2 sensor. v2 sensor can always be powered on by pressing the power button for 2 seconds. Additionally, it can also be turned on by USB plug in. This function is disabled by default and it can be enabled by setting the b1 to "1".	b0: Set '1' to power off the sensor b1: Set '1' to enable the v2 sensor to be powered on by USB plug in. Set '0' to disable it. b2 – b7: reserved Set b0 to '1' will ignore other bits in this field.
Power saving timeout X (minute)	1	Timeout threshold that sensor goes to power saving mode in advertisement mode. Set to 0 if you want to disable power saving in advertisement mode.	0 ~ 30, default value 10
Power saving timeout X (second)	1		0 ~ 60, default value 0
Power saving	1	Timeout threshold that sensor goes to power saving mode in connection mode.	0 ~ 30, default value 30

timeout Y (minute)		Set to 0 if you want to disable power saving in connection mode.	
Power saving timeout Y (second)	1		0 ~ 60, default value 0
Device Tag length	1	Length of tag name. Ignore the write operation for invalid length.	0 ~ 16
Device Tag	16	Device tag name. Don't use special characters such as '/\:,' in case of potential errors in host applications.	Default value "Movella DOT"
Output rate	2	Data output rate (Hz) for real-time streaming and recording. It cannot be changed after measurement or synchronization start.	Values 1, 4, 10, 12, 15, 20, 30, 60 (Default), 120 (Recording)
Filter profile index	1	Index of the filter profile. Use the index to get or set the active filter profile for real-time streaming and recording. It cannot be changed after measurement or synchronization start.	Refer to Table 8 for the index and information of filter profiles.
Reserved	5	Reserved for future use	0

Note - the bold fields are non-volatile and are saved on the Movella DOT sensor.

Table 8: Filter profile index and information

Index	Name	Name length	Description
0	General	7	Default for general human motion
1	Dynamic	7	For fast and jerky human motion like sprinting.

2.3 Device Report Characteristic

The device report characteristic is a 36-bytes data structure. Based on the command sent from the device control characteristic, or the operations applied on the sensor, the sensor will send out specific report to inform the central device. The following sections list out 3 different types of report. The structure of each report is also specified. Unused bytes are reserved for future use and can be ignored.

2.3.1 Power Off

Sensor will send out this report when it is powered off by following methods:

- Power button is pressed for over 3s.
- Power-off command is received from host as described in Table 7.
- Battery level is lower than 2%.

Table 9: Power off structure

Field name	Size	Description	Value
Type	1	Sensor is powered off.	1
-	35	Unused	-

2.3.2 Power Saving

Sensor will send out this report when entering power saving mode in advertisement or connection mode.

Table 10: Power saving structure

Field name	Size	Description	Value
Type	1	Sensor is in power saving mode.	4
-	35	Unused	-

2.3.3 Button Callback

The button callback is available when the DOT is connected to the host. A report is sent from the device to the host when the power button is pressed. The report consists of the type of button press and a timestamp. There are three types of button presses, single, double and triple clicks. This message format is available in Table 11.

Valid clicks are detected when the time between button down and button up is between 30-1600ms. After releasing the button there is a cooldown period of 800ms. If another valid click is detected in the cooldown period then it is registered as another click in the series. This repeats as well for the triple click. The device will wait for the cooldown period before sending the notification (single and double click). The triple click callback is sent immediately (no cooldown) and a new chain is started. The timestamp is taken from the sensor's local clock and independent of synchronization timestamp. The timestamp relates to the starting point (button down) of the click-chain. Longer presses than 1600ms will lead to different behaviour, such as power down of the device, consult the user manual for more information.

Table 11: Button callback structure

Field name	Size	Description	Values
Type	1	Detected click type	5 – single click 6 – double click 7 – triple click
Length	1	The length of the timestamp	4
Timestamp	4	Timestamp when the button is pressed. Unit is millisecond.	
-	30	Unused	-

3 Measurement Service

Measurement service enables the start and stop of the measurement on the sensor, as well as setting payload modes and receiving the measurement data. It also offers control for the orientation reset functionality.

The UUID of this service is **0x2000** and relevant characteristics are given in Table 12

Table 12: Characteristics of the Measurement Service

Characteristic	UUID	Description	Length	Property
Control	0x2001	Control the start/stop and payload mode of the measurement.	3	Read, Write
Long payload length	0x2002	Return the data of payload modes that have payload length higher than 40 bytes.	63	Notify
Medium payload length	0x2003	Return the data of payload modes that have payload length between 21 to 40 bytes.	40	Notify
Short payload length	0x2004	Return the data of payload modes that have payload length between 0 to 20 bytes.	20	Notify
Magnetic field mapper	0x2005	Reserved, not for public use.	-	-
Orientation reset control	0x2006	Reset or revert the heading.	2	Read, Write
Orientation reset status	0x2007	Heading reset result	1	Read
Orientation reset data	0x2008	Reserved, not for public use.	-	-

3.1 Control Characteristic

The control characteristic is a 3-bytes data structure with the fields as specified in Table 13.

Note - Enable BLE notification on the matching 'Payload' characteristic (0x2002, 0x2003, 0x2004) to get the measurement data for real-time streaming before setting the start action in the Control characteristic (0x2001).

Table 13: Control characteristic structure

Field Name	Size	Description	Values
Type	1	Type of the control target	1: measurement Others are invalid value
Action	1	Start or stop the measurement or get the status of the measurement.	0: stop, or measurement is stopped 1: start, or measurement is started
Payload mode	1	Set the payload mode or get the current payload mode. Is only applied on 'start' action. Based on payload length, each measurement type will use long, medium, or short payload length characteristics for notification. Refer to section 3.2, 3.3 and 3.4 for the data structure of each payload mode.	1: High Fidelity (with mag) [1] 2: Extended (Quaternion) 3: Complete (Quaternion) 4: Orientation (Euler) 5: Orientation (Quaternion) 6: Free acceleration 7: Extended (Euler) 16: Complete (Euler) 17: High Fidelity[2] 18: Delta quantities (with mag) 19: Delta quantities 20: Rate quantities (with mag) 21: Rate quantities 22: Custom mode 1 23: Custom mode 2 24: Custom mode 3 25: Custom mode 4[3] 26: Custom mode 5

Note - the **bold** fields are non-volatile and are saved by the Movella DOT sensor.

[1] High Fidelity (with mag) payload mode can only be parsed through the SDK
[2] High Fidelity payload mode can only be parsed through the SDK
[3] Custom mode 4 payload mode can only be parsed through the SDK as it contains high fidelity inertial data.

3.2 Long Payload Length Characteristic

Long payload length characteristic will return the data of payload mode that has the payload length higher than 40 bytes.

Based on the selected payload mode in control characteristic, the structure of long payload characteristic is specified in Table 14. Refer to Table 17 for the definition and format of each measurement data. The data transmission order is the data structure order listed in the table. Unused bytes are all set to 0x00.

Table 14: Long payload output modes

Payload mode	Total Size	Data structure
Custom mode 4	51	This mode contains timestamp, inertial data in high fidelity mode, quaternion, magnetic field data and status. High fidelity data can only be parsed through the SDK.
Custom mode 5	44	• Timestamp • Quaternions • Acceleration • Angular velocity

3.3 Medium Payload Length Characteristic

Medium payload length characteristic will return the data of payload mode that has the payload length between 21 and 40 bytes.

Based on the selected payload mode in control characteristic, the structure of medium payload characteristic is specified in Table 15. The definition and format of each measurement data type is given in Table 17Table 17. The data transmission order is the data structure order listed in the table. Trailing unused bytes are all set to 0x00.

Table 15: Medium payload output modes

Payload mode	Size	Data structure
Extended (Quaternion)	36	• Timestamp • Quaternion • Free acceleration • Status • Clipping Count Accelerometer • Clipping Count Gyroscope
Complete (Quaternion)	32	• Timestamp • Quaternion • Free acceleration
Extended (Euler)	32	• Timestamp

		• Euler • Free acceleration • Status • Clipping Count Accelerometer • Clipping Count Gyroscope
Complete (Euler)	28	• Timestamp • Euler • Free acceleration
High Fidelity (with mag)	35	This mode contains timestamp, inertial data in high fidelity mode, magnetic field data and status. This can only be parsed by the Movella DOT SDK.
High Fidelity	29	This mode contains timestamp, inertial data in high fidelity mode and status. This can only be parsed by the Movella DOT SDK.
Delta quantities (with mag)	38	• Timestamp • dq • dv • Magnetic field
Delta quantities	32	• Timestamp • dq • dv
Rate quantities (with mag)	34	• Timestamp • Acceleration • Angular velocity • Magnetic field
Rate quantities	28	• Timestamp • Acceleration • Angular velocity
Custom mode 1	40	• Timestamp • Euler • Free acceleration • Angular velocity
Custom mode 2	34	• Timestamp • Euler • Free acceleration • Magnetic field
Custom mode 3	32	• Timestamp • Quaternion • Angular velocity

3.4 Short Payload Length Characteristic

Short payload length characteristic will return the data of payload mode that has the payload length lower than 20 bytes.

Based on the selected payload mode in control characteristic, the structure of long payload characteristic is specified in Table 16. The definition and format of each measurement data type is given in Table 17. The data transmission order is the data structure order listed as below. Unused bytes are all set to 0x00.

Table 16: Short payload output modes

Payload mode	Total Size	Data structure
Orientation (Euler)	16	• Timestamp • Euler
Orientation (Quaternion)	20	• Timestamp • Quaternion
Free acceleration	16	• Timestamp • Free acceleration

3.5 Measurement Data

This section explains the definition and format for each measurement data in medium and short payload length characteristics. Refer to *Movella DOT User Manual* for more details about each data type.

Table 17: Measurement data types

Data	Size	Description	Format
Timestamp	4	Timestamp on the sensor in microseconds.	
Quaternion	16	The orientation expressed as a quaternion.	w,x,y,z, float
Euler angles	12	The orientation expressed as Euler angles, degree.	x,y,z, float
Free acceleration	12	Acceleration in local earth coordinate and the local gravity is deducted, m/s^2	x,y,z, float
dq	16	Orientation change during a time interval.	w,x,y,z, float
dv	12	Velocity change during a time interval, m/s.	x,y,z, float
Acceleration	12	Calibrated acceleration in sensor coordinate, m/s^2.	x,y,z, float
Angular velocity	12	Rate of turn in sensor coordinate, dps.	x,y,z, float
Magnetic field	6	Magnetic field in sensor coordinate, a.u.	x,y,z, fixed point
Status	2	See section 3.5.1.	unsigned short
Clipping Count Accelerometer	1	Amount of clipping accelerometer samples on one or more axes	unsigned integer
Clipping Count Gyroscope	1	Amount of clipping gyroscope samples on one or more axes	unsigned integer

Note – 'float' refers to the IEEE-754 32-bit Floating Point number format.

3.5.1 Status Definition

The status datatype contains metadata about the motion data provided by the Movella DOT sensor.

Data (Bit mask)	Abbr.	Description
0x0001	FL_ClipAccX	Accelerometer is out of range in x-axis
0x0002	FL_ClipAccY	Accelerometer is out of range in y-axis
0x0004	FL_ClipAccZ	Accelerometer is out of range in z-axis
0x0008	FL_ClipGyrX	Gyroscope is out of range in x-axis
0x0010	FL_ClipGyrY	Gyroscope is out of range in y-axis
0x0020	FL_ClipGyrZ	Gyroscope is out of range in z-axis
0x0040	FL_ClipMagX	Magnetometer is out of range in x-axis
0x0080	FL_ClipMagY	Magnetometer is out of range in y-axis
0x0100	FL_ClipMagZ	Magnetometer is out of range in z-axis
0x0200	FL_MagIsNew	Magnetometer data in this packet is new.

For example, if status is 0x0012, it means accelerometer is out of range in y-axis and gyroscope is out of range in y-axis for this data packet. The *Movella DOT User Manual* defines the axes of the Movella DOT hardware.

3.6 Orientation Reset Control Characteristic

Orientation reset allows the user to align the orientation outputs among all sensors and with the object(s) they are connected to. Only heading reset is available for now. The heading reset or revert must be executed during the measurement. Heading reset is maintained between connection/disconnection or different measurements but will be lost after the sensor reboots. After reset the heading, a revert is required before conducting a new reset.

The orientation reset control characteristic a 2-bytes data structure with the fields as specified in Table 18.

Table 18: Orientation reset control structure

Field name	Size	Description	Values
Type	2	Control to reset or revert the heading.	0x0001: Reset heading 0x0007: Revert heading to default 0x0008: Default status

3.7 Orientation Reset Status Characteristic

This read-only characteristic shows the status of orientation reset in the Movella DOT sensor. Heading reset can fail or succeed which is represented with a single byte.

Table 19: Orientation reset status structure

Field name	Size	Description	Values
ResetResult	1	The result of heading reset	0: Fail 1: Success

3.8 Best Practice for Measurement Service

This section describes some best practices to interact with the measurement service of the Movella DOT. Figure 1 describes the steps needed to start measurement and stream motion data to the BLE host. It assumed that the host is connected to the Movella DOT and no other operations are ongoing. Figure 1 shows to first enable notification on one of the measurement data payload characteristics. After that, payload type and start measurement can be written to the control characteristic. Stopping measurement should happen in the reverse order, first stop measurement, then disable notification.

Figure 2 shows the interaction to control the heading reset functionality. This functionality is usable during streaming measurement.

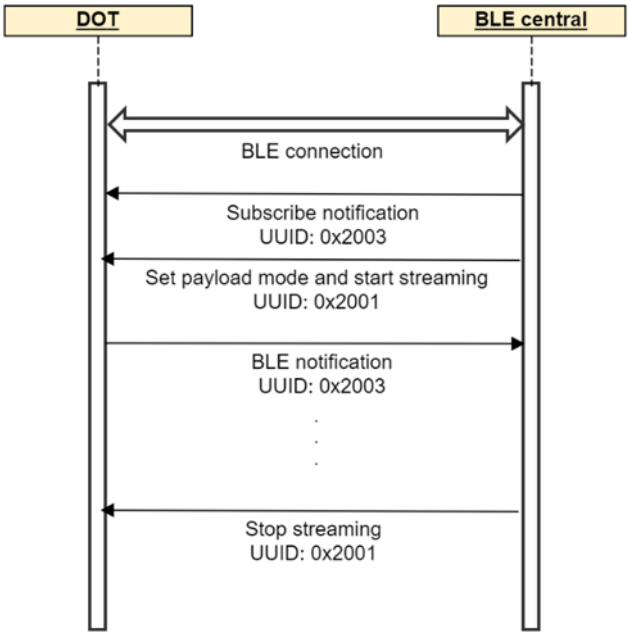

Figure 1: Best practice to start measurement

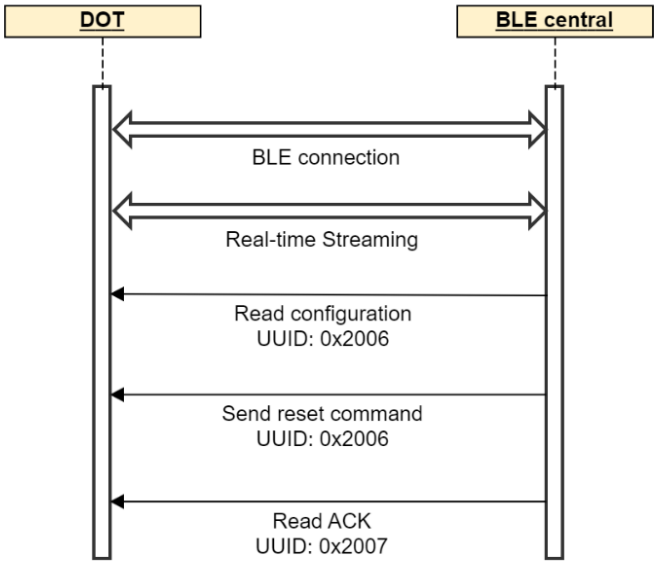

Figure 2: Best practice to perform heading reset

4 Battery Service

This service provides battery information such as battery level and charging status. The sensor will periodically read the fuel gauge to give an up-to-date value. This service will explicitly notify the battery level when battery level percentage changes.

The UUID of this service is **0x3000** and relevant characteristics are given in Table 20.

Table 20: Battery characteristic structure

Characteristic	UUID	Description	Length	Property
Battery	0x3001	Battery level and charging status	2	Read, Notify

4.1 Battery Characteristic

The battery characteristic is a 2-bytes data structure with the fields as specified in Table 21.

Table 21: Battery characteristic structure

Field name	Size	Description	Values
Battery level	1	Battery level in percentage	0~100
Charging status	1	Charging status of the battery	0: Not charging 1: Charging

4.2 Best Practice for Battery Services

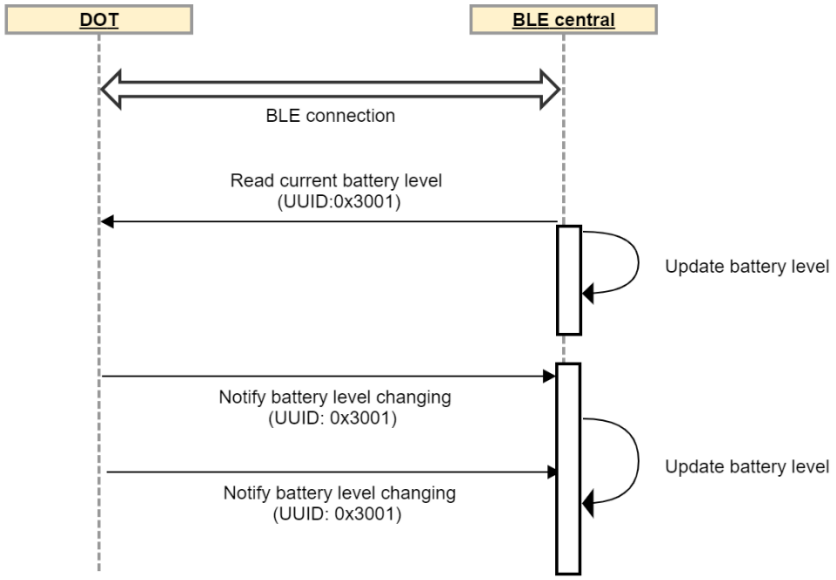

Figure 3: Best practice of battery service

5 Message Service

Message service is a shared service for many different device functions. It handles recording and time synchronization functionality amongst others. The basis is formed by a generic message format and characteristics that allow two-way communication between host and sensor.

The UUID of this service is **0x7000** and its characteristics are given in Table 22. The message service consists of 'synchronous' and 'asynchronous' communication. Synchronous communication is initiated by the host and a Control command is written to the DOT device. The DOT will reply to the synchronous message with a message in the Acknowledge characteristic. This Acknowledge needs to be read by the host. Asynchronous communication occurs when the DOT sends messages via the Notification characteristic. This process is still initiated by the host with a Control command, however the DOT will send one or more asynchronous notification messages. The BLE notify option must be enabled by the host on the Notification characteristic for asynchronous communication to work.

Table 22: Characteristics of the Message Service

Characteristic	UUID	Description	Length	Property
Control	0x7001	Manage control messages	160	Write
Acknowledge	0x7002	Manage acknowledge messages	160	Read
Notification	0x7003	Manage notification message	160	Notify

5.1 Message Structure

The communication via the message service with Movella DOT sensors is done by messages which are built according to a standard structure. The message has a maximum of 157 data bytes.

A DOT message contains the following fields:

MID	LEN	DATA	CHECKSUM

Table 23: Construction of the Message

Field	Field width	Description
MID	1 byte	Message Identifier
LEN	1 byte	Specifies the number of data bytes in the DATA field
DATA	0-157 bytes	Data bytes
CHECKSUM	1 byte	Checksum of message

Message Identifier (MID)
This message field identifies the type of message. It addresses the subsystem responsible for handling the command, or identifies the origin of a notification or acknowledgement.

Table 24: Message ID

MID	Description
0x01	Recording message
0x02	Synchronization message
0x03	Configuration message

Length (LEN)
Specifies the number of data bytes in the DATA field. The valid range is 0-157.

Data (DATA)
This field contains the data bytes. The data is always transmitted in little-endian format.

Checksum
This field is used for communication error-detection. If all message bytes are summed and the lower byte value of the result equals zero, the message is valid and it may be processed. The checksum value of the message should be included in the summation.

5.2 Recording Message

With recording function, sensor data can be stored in internal storage and exported for post process after the measurement. Refer to the *Movella DOT User Manual* for more information about recording function.

MID of recording messages is 0x01. DATA field of recording message contains two fields: recording ID (ReID) and recording DATA (ReDATA).

A recording message contains the following fields:

		DATA		
MID(0x01)	LEN	ReID	ReDATA	CHECKSUM

Recording ID (ReID)
This field identifies different recording messages. For a complete listing of all possible recording messages see section 5.2.2, 5.2.3 and 5.2.4.

Recording DATA (ReDATA)
This field contains the recording data bytes. The interpretation of the recording data bytes is recording message specific, i.e. depending on the ReID value the meaning of the ReDATA bytes is different.

5.2.1 Recording message usage

After sending a recording control message with a certain ReID, check the Acknowledge right after it. Otherwise, this acknowledge will be overwritten by the acknowledge of the next control message.

Some recording message will be replied to with a notification message with a ReID value that is increased. Depending on the message type, the characteristic message can have a data field (no fixed length) or not. If nothing is specified, the data field does not exist.

Some messages have the same ReID and the meaning differs depending on its message service characteristics. For example, the ReID of control message stop recording (StopRecording) is the same as the notification message that recording stopped (RecordingStopped). The difference between the two messages is that they use different message characteristics.

Example
Request the recording status:

Sending message:
GetState = 0x0101 02 FC
Check acknowledge:
Acknowledge = 0x0103 0106 02 F3

Example
Start a 30 min timed recording:

Sending message:
StartRecording = 0x0107 40 DF503B5B0807 E0
Check acknowledge:
Acknowledge = 0x0109 0100 40 DF503B5B0807 DD

Example
Stop a timed recording:

Sending message:
StopRecording = 0x0101 41 BD
Check acknowledge:
Acknowledge = 0x0103 0100 41 BA

Example
Request 1st file information:

Sending message:
RequestFileInfo = 0x0102 6001 9C
Check acknowledge:
Acknowledge = 0x0104 0100 6001 99
Receiving notification:
Notification = 0x0181 61 6F636552 80000000 0001B8C633 ... 00000000 0x0101 62 9C

Example
Select export data with following data quantities:

1. Timestamp
2. Quaternion
3. dq
4. dv
5. Calibrated angular velocity
6. Calibrated acceleration
7. Calibrated mag
8. Status

Sending message:
SelectExportData = 0x0109 74000105 06070809 0A 54
Receiving notification:
Notification = 0x010B0 1000 74000105 06070809 0A 51

Example
Request 7th file data:

Sending message:
RequestFileData = 0x0102700786
Check acknowledge:
Acknowledge = 0x01040100 700783
Receiving notification:
Notification =
0x01557100 00000075 A766A81A 3D643EFF ... 00000000 00000000
0x01557101 00000090 E866A84B A7673E67 ... 00000000 00000000...

5.2.2 Recording control messages

5.2.2.1 GetState

ReID	0x02	Size
ReDATA	n/a	0

Direction	To sensor	
Valid in	Any state	

Request sensor's recording state. Check acknowledge to get the result.

5.2.2.2 EraseFlash

ReID	0x30	Size
ReDATA	EraseUTC	4
Direction	To sensor	
Valid in	Idle state	

Request to clear all the recording data space, other flash space will not be affected. A **StoreFlashInfoDone** notification will be sent to host if recording flash erase is completed.

EraseUTC
The erase start time that contains the timestamp expressed as the UTC time in seconds.

5.2.2.3 StartRecording

ReID	0x40	Size
ReDATA	StartUTC	4
	RecordingTime, unsigned integer	2
Direction	To sensor	
Valid in	Idle state	

Start recording message **RecordingStopped** notification will be sent to the host once the recording stops. Recording will automatically stop in the following situations:

- power button is pressed over 1 second
- time is up for timed recording
- flash memory is over 90%

StartUTC
The recording start time that contains the timestamp expressed as the UTC time in seconds.

RecordingTime
Set RecordingTime to 0xFFFF if you want a recording without timer. Otherwise, set the RecordingTime to a value in second to record for a certain. Don't set a timer that exceeds the maximum recording time. Maximum recording time is based on the data rate as set in the Device Config Characteristic. At 60Hz a DOT v1 is able to record data for 88 min = 5280 seconds, given that the flash is fully cleared. Maximum recording time for DOT v2 at 60Hz is roughly 362 min = 21720 seconds.

5.2.2.4 StopRecording

ReID	0x41	Size

ReDATA	n/a	0
Direction	Both	
Valid in	Recording state	

Stop recording command.

5.2.2.5 RequestRecordingTime

ReID	0x42	Size
ReDATA	n/a	0
Direction	To sensor	
Valid in	Recording state	

Request the recording time since recording started. **RecordingTime** notification will be sent to host with recording time information.

5.2.2.6 RequestFlashInfo

ReID	0x50	Size
ReDATA	n/a	0
Direction	To sensor	
Valid in	Idle state	

Request recording flash information. **ExportFlashInfo** notification will be sent to host with flash information. **ExportFlashInfoDone** notification will be sent if the flash information has been sent completely.

5.2.2.7 RequestFileInfo

ReID	0x60	Size
ReDATA	FileIndex (unsigned)	1
Direction	To sensor	
Valid in	Idle state	

Request recording file information by FileIndex. **ExportFileInfo** notification will be sent to host with the requested recording file information.

ExportFileInfoDone notification will be sent if the file information has been sent completely.

FileIndex
Index of the recording files. Starts from 0x01 and maximum up to 0xFE. You can get total file number and file sizes from **ExportFlashInfo** notification.

5.2.2.8 RequestFileData

ReID	0x70	Size

ReDATA	FileIndex (unsigned)	1
Direction	To sensor	
Valid in	Idle state	

Request recording file data based on FileIndex. Recording file data packets will be sent to host via **ExportFileData** notification. **ExportFileDataDone** notification will be sent if all the file data has been sent.

5.2.2.9 StopExportData

ReID	0x73	Size
ReDATA	n/a	0
Direction	Both	
Valid in	Export recording data	

Use this message to stop data exporting.

5.2.2.10 SelectExportData

ReID	0x74	Size
ReDATA	SelectedData (one byte per data type)	LEN - 1
Direction	To sensor	
Valid in	Idle state	

Configure export data options. Set byte array if you want to export multi-quantites. This message should be sent before RequestFileData. Otherwise, the default data byte array [0x00, 0x04, 0x07, 0x08] will be set.

SelectedData
See Table 25 for the available data quantity and the corresponding value of SelectedData. Refer to chapter 4 in *Movella DOT User Manual* for the meanings of data and section 3.5 for data format.

Table 25: Available data when exporting

Data quantity	SelectedData
Timestamp	0x00
Quaternion	0x01
Euler Angles	0x04
dq	0x05
dv	0x06
Acceleration	0x07
Angular Velocity	0x08
Mag Field	0x09
Status	0x0a
Clipping Count Accelerometer	0x0b
Clipping Count Gyroscope	0x0c

Note - Free acceleration is not provided in this firmware. Refer to this online base article to calculate free acceleration from quaternion and dv.

5.2.2.11 Retransmission

ReID	0x75	Size
ReDATA	RetransDataNumber	4
Direction	To sensor	
Valid in	Export data state	

Retransmit all the data from the RetransDataNumber packet.

RetransDataNumber
Packet Counter of the retransmit data packet.

5.2.3 Recording acknowledge message

ReID	0x01	Size
ReDATA	Ack result (see Table 26)	1
	Control message ReID	1
	Control message ReDATA	LEN-2
Direction	To host	

Acknowledge (ACK) is the receipt of a control message. ReDATA contains the Result in 1 byte and the control message DATA from host to clarify which message the ACK is responding to.

Result
Indicates the receiving status of a control message, or the sensor states when receiving the message.

Table 26: Recording ACK Results

Result	Description	Details
0x00	Success	Control messages write success
0x02	InvalidCmd	Invalid command
0x03	FlashProcessBusy	Flash is occupied by other process
0x06	IdleState	Idle state
0x30	OnErasing	Erasing internal storage
0x40	OnRecording	In recording state
0x50	OnExportFlashInfo	Exporting flash information
0x60	OnExportRecordingFileInfo	Exporting recording file information
0x70	OnExportRecordingFileData	Exporting recording data

Control message ReID
ReID of the control message.

Control message ReDATA
ReDATA of the control message.

5.2.4 Recording notification messages

Notifications of recording control messages which are sent asynchronously. Enable notification on the characteristic to receive these messages.

5.2.4.1 FlashProcessBusy

ReID	0x03	Size
ReDATA	n/a	0
Direction	To host	

Flash is occupied by another process. Wait a while and send the control message again.

5.2.4.2 StoreFlashInfoDone

ReID	0x33	Size
ReDATA	n/a	0
Direction	To host	

Recording flash erase is completed.

5.2.4.3 FlashFull

ReID	0x34	Size
ReDATA	n/a	0
Direction	To host	

Recording flash space is full.

5.2.4.4 InvalidFlashFormat

ReID	0x35	Size
ReDATA	n/a	0
Direction	To host	

Recording flash format is invalid. Current firmware version doesn't support the flash format. Use **EraseFlash** to reset the flash format.

5.2.4.5 RecordingStopped

ReID	0x41	Size
ReDATA	n/a	0
Direction	Both	

Recording stopped.

5.2.4.6 RecordingTime

ReID	0x43	Size
ReDATA	StartUTC	4
	TotalRecordingTime	2
	RemainingRecordingTime	2
Direction	To host	

StartUTC
Recording start UTC and unit is second.

TotalRecordingTime
Total recording time of a timed recording. Unit is second. 0xFFFF for normal (non-timed) recording.

RemainingRecordingTime
Remaining time of a timed recording. Unit is second. 0xFFFF for normal (non-timed) recording.

5.2.4.7 ExportFlashInfo

After the RequestFlashInfo command is received the sensor will send the Flash Info Header (see below) back to the host via the Notification characteristic. The Acknowledge characteristic will contain the status and a copy of the command. After the Flash Info header is sent, the File Indicators will follow asynchronously in the notification characteristic. The export will finish with an ExportFlashInfoDone message. If the recording space is not initialized, the Flash Info Header and File Indicators will be omitted, only ExportFlashInfoDone will be sent. The amount of File Indicators is different for Dot v1 and v2; its size is added to the header in header revision 2.0. File Indicators for empty recording space are not exported.

ReID	0x51	Revision	Size
ReDATA	Header magic number (0x466C6173)	1.0	4
	Header size	1.0	4
	Header revision	1.0	2
	Recording space initialization UTC	1.0	4
	Recording space size	1.0	4
	File indicators size	2.0	4
	Reserved	1.0	106
Direction	To host		

The File Indicators represent a simple recording file system is created to manage the recording files. Following table shows the structure of the recording flash. It consists of a header file and the recording files.

Table 27: Structure of recording flash

Header file (4 kB)	1st recording file	2nd recording file	…

Recording flash information is stored in the Flash Info Header. See Table 28 for the structure of file indicator.

Table 28: Structure of file indicator

Field	Field width	Description
1st file header	1 byte	File header, 0xEE represents a header
1st file data	1 byte per sector	File data, 0xCC means sector has data
2nd file header	1 byte	0xEE indicates next file (header)
2nd file data	1 byte per sector	0xCC CC CC CC for 4 x ~228kB of data
…		
Empty space	1 byte per sector	0xFF indicates unused sector.

Unused bytes in file indicator are set to 0xFF. File Indicators are all 0xFF bytes if there is no header or data in those sectors. If no recording file headers or data are present then no File Indicators will be exported.

You can get the file number and rough file size through file indicator. For example, 0xEECCEECCCCFF…FF means that there are 2 files in the recording flash:

- The first file has a rough size of 232kB (4kB +228kB)
- The second file has a rough size of 460kB (4kB +228kB*2)

This notification may asynchronously send multiple messages based on the actual length of the header file, with each message containing 128 bytes ReDATA. It stops when all File Indicators with actual file data are exported.

If no Flash Info Header is received, and also no File Indicators, then the device will directly send ExportFlashInfoDone. That means the Recording Space Flash Info header structure is invalid. Erase the recording space to reinitialize it. Data cannot be recovered after reinitialization of the file system.

5.2.4.8 ExportFlashInfoDone

ReID	0x52	Size
ReDATA	n/a	0
Direction	To host	

Recording flash information export is completed.

5.2.4.9 ExportFileInfo

ReID	0x61	Revision	Size
ReDATA	Magic number (0x5265636F)	1.0	4
	Header size (128 bytes)	1.0	4
	Revision	1.0	2
	Start recording UTC (from StartRecording)	1.0	4
	Total recording time (from StartRecording)	1.0	2
	Reserved	1.0	112
Direction	To host		

Refer to Table *29* for file structure.

Table 29: Structure of the file

Header space (4 kB)	Data

Recording file information is preserved in header space before the measurement data. The message contains metadata of the recording and delineates which measurement data belongs to which file. ExportFileInfo is separate from ExportFileData.

5.2.4.10 ExportFileInfoDone

ReID	0x62	Size
ReDATA	n/a	0
Direction	To host	

Recording file information export is completed.

5.2.4.11 NoRecordingFile

ReID	0x63	Size
ReDATA	n/a	0
Direction	To host	

There is no recording file (with this FileIndex).

5.2.4.12 ExportFileData

ReID	0x71	Size
ReDATA	DataNumber	4
	ExportedData with measurement data fields determined by SelectExportData	LEN - 5
Direction	To host	

Export the file data based on the FileIndex and SelectedData. Refer to section 3.5 for the format of each data. Each notification contains one data packet with the following ReDATA fields:

DataNumber
Data packet counter, starts from 0.

ExportedData
Data packet base on SelectedData configuration.

5.2.4.13 ExportFileDataDone

ReID	0x72	Size
ReDATA	n/a	0
Direction	To host	

Recording data export is completed.

5.2.4.14 ExportDataStopped

ReID	0x73	Size
ReDATA	n/a	0
Direction	Both	

5.2.4.15 ExportFileDataInvalid

ReID	0x76	Size
ReDATA	DataNumber	4
	ExportedData with measurement data fields determined by SelectExportData	LEN - 5
Direction	To host	

Invalid measurement data due to internal checksum or preamble check fail.

5.2.5 Best practice for recording

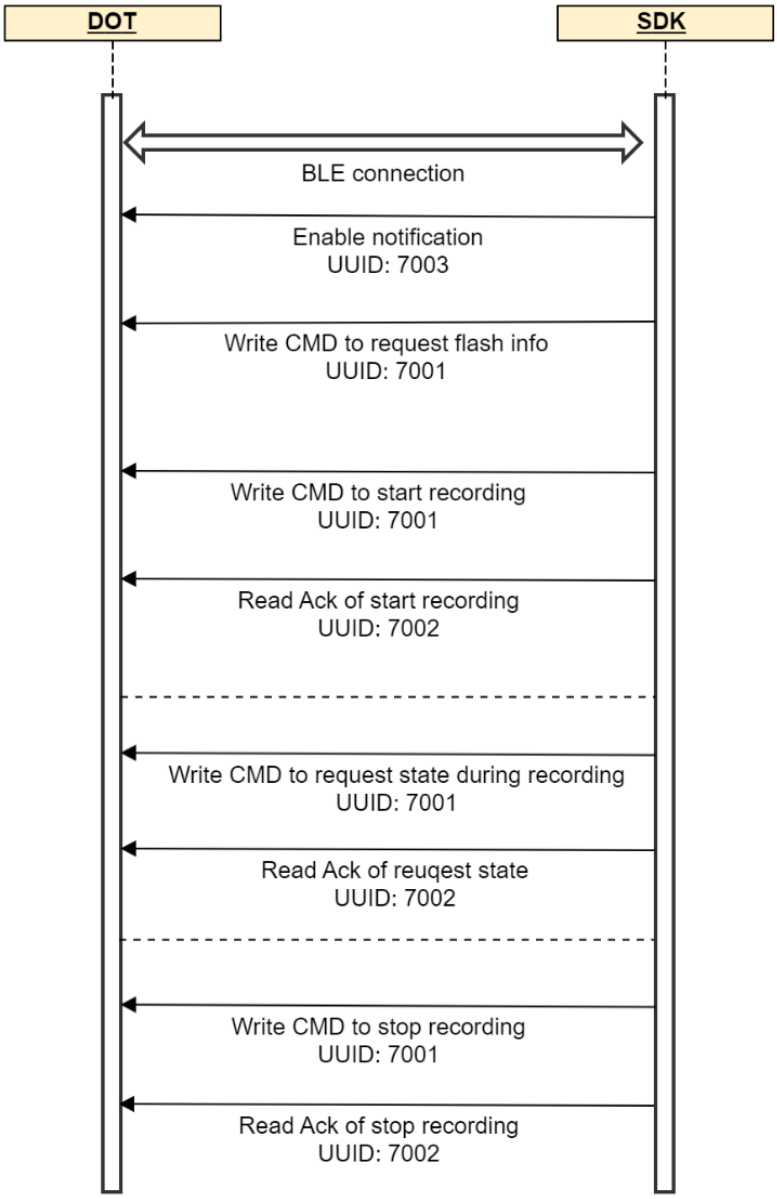

Figure 4: Best practice to start and stop recording

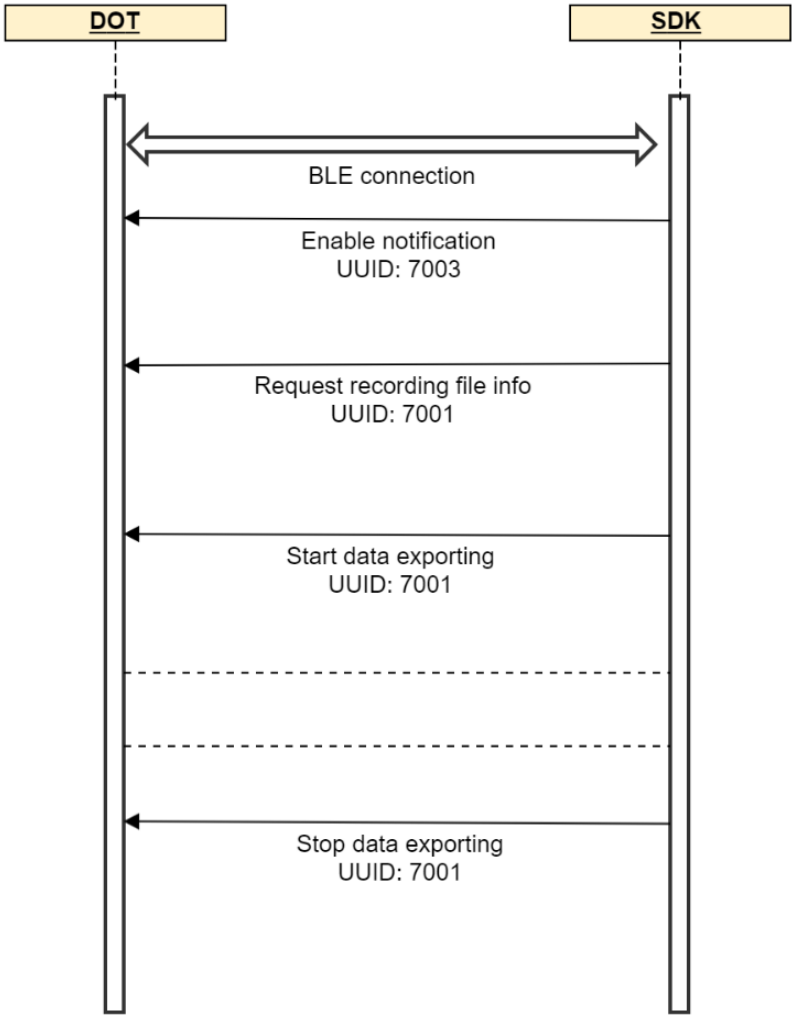

Figure 5: Best practice to export recording files

5.3 Synchronization Message

All sensors can be time-synced with each other to a common sensor time base with the synchronization functionality. Refer to the *Movella DOT User Manual* for more information. Synchronisation is started and controlled via the Message Service.

MID of synchronization messages is 0x02. DATA of synchronization message contains the Sync ID (SyID) and possible Sync DATA (SyDATA). A synchronization message contains the following fields:

		DATA		
MID(0x02)	LEN	SyID	SyDATA	CHECKSUM

Sync ID (SyID)
This field identifies different synchronization messages. For a complete listing of all possible synchronization messages see subsection 5.3.2, 5.3.3 and 5.3.4.

Sync DATA (SyDATA)
This field contains the synchronization data bytes. The interpretation of the synchronization data bytes is synchronization message specific, i.e., depending on the SyID value the meaning of the SyDATA bytes is different.

5.3.1 Synchronization message use cases

Example
Get synchronization status.

Sending message:
GetSyncStatus = 0x0201 08 F5
Get notification:
SyncStatus = 0x0202 5109 A2 or 0x0202 5104 A7

Example
Start synchronization. Root node MAC address is D4:22:CD:AA:BB:CC.

Sending message:
StartSync = 0x0207 01CCBBAACD22D4 02
Check acknowledge:
Acknowledge = 0x0202 0300 F9

Example
Stop synchronization.

Sending message:
StopSync = 0x0201 02 FB
Get notification:
StopSyncResult = 0x0202 5000 AC

5.3.2 Synchronization control messages

5.3.2.1 StartSync

SyID	0x01	Size
SyDATA	Root node MAC address	6
Direction	To sensor	
Valid in	Connection state	

Start the synchronization. Refer to section 5.3.5 for the best practice to start synchronization.

Root node MAC address
Root node can be any of the sync sensors. Like most fields in the protocol it is Little Endian. That means that if the MAC address is AA:BB:CC:DD:EE:FF, then the SyDATA should be 0xFF, 0xEE, 0xDD, 0xCC, 0xBB, 0xAA. See the example use case in section 5.3.1

5.3.2.2 StopSync

SyID	0x02	Size
SyDATA	n/a	1
Direction	To sensor	
Valid in	Synced state	

Stop the synchronization. **StopSyncResult** notification will be sent to let the host know if the StopSync is successful or not.

5.3.2.3 GetSyncStatus

SyID	0x08	Size
SyDATA	n/a	1
Direction	To sensor	
Valid in	Connection state	

Check if the sensor is already synced or not. **SyncStatus** notification will be sent to host with the sync status.

5.3.3 Synchronization acknowledge message

SyID	0x03	Size
SyDATA	Result	1
	Control Message SyID	1
	Control Message SyData	LEN - 2
Direction	To host	

Acknowledge (ACK) is the receipt of a control message. SyDATA contains the Result in 1 byte and the control message DATA from host to clarify which message the ACK is responding to.

Result

Indicates the result of the synchronization after re-connection with the sensor.

Table 30: Synchronization ACK Results

Result	Description	Details
0x00	Success	Synchronization success
0x05	NotEnoughSamples	Sync failed for not enough data samples
0x07	SkewTooLarge	Sync failed for estimated skew too large
0x08	StartingTimingError	Sync failed for start time error
0x09	Unstarted	Sync is not started

Control message SyID
SyID of the control message.

Control message SyDATA
SyDATA of the control message.

5.3.4 Synchronization notification message

Notifications of synchronization control messages. A callback is required to handle notifications.

5.3.4.1 StopSyncResult

SyID	0x50	Size
SyDATA	Result	1
Direction	To host	

Result
The result of stop sync command. 0x00 means success. 0x01 means failed.

5.3.4.2 SyncStatus

SyID	0x51	Size
SyDATA	SyncStatus	1
Direction	To host	

SyncStatus
The sync status of the sensor. 0x04 means synced. 0x09 means un-synced.

5.3.5 Best practice for synchronization

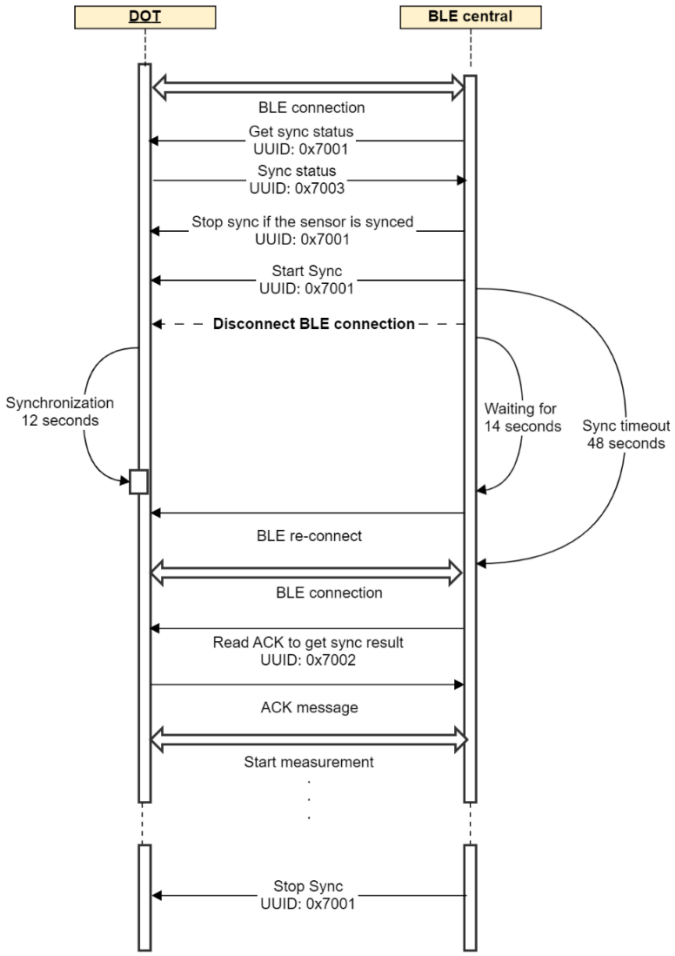

Figure 6: Best practice to start synchronization

Before starting the synchronization, check the synchronization status of the target sensors and make sure they are not synced. Stop the synced sensor before starting a new synchronization to prevent error status.

Set the output rate and filter profile before starting the synchronization. Since the sensor will enter measurement mode right after the sync succeeds so it's not possible to change it after sync.

Disconnect the sensors after sending start sync command to allow scanners to receive the data from the root sensor. It will take about 12 seconds to finish the sync period. Reconnect to the sensors again after 14s and retry if connection fails. Read the acknowledge right after the reconnection to get the synchronization result. If the synchronization is successful for all sensors, then you can start the measurement. If any of the sensors fails in synchronization, you can continue with the successful sensors or stop the synchronization for all the involved sensors and try again.

5.4 Configuration Message

Configuration messages can be used to request device configurations or change them. Configuration messages follow the message structure explained in subsection 5.1. The Message ID (MID) of configuration messages is 0x03. The ConfigID determines what configuration operation is being requested.

A configuration control message contains the following fields.

		DATA[1..n]		
MID(0x03)	LEN	ConfigID	ConfigData	CHECKSUM

ConfigID
This field contains the ID of the requested configuration operation and is one byte long.

ConfigData
This field can contain data to accompany a request to the sensor. Currently only the RequestFilterProfileName message makes use of this field.

There exist two configuration acknowledgement messages. One simple message containing only the requested data in the acknowledgement DATA field. The other acknowledgment DATA format refers to the original command with the ConfigAckID and optional ConfigAckData. Both formats respond with the Configuration MID.

		DATA[0…n]	
MID(0x03)	LEN	Acknowledgement	CHECKSUM

		DATA[1…n]		
MID(0x03)	LEN	ConfigAckID	ConfigAckData	CHECKSUM

5.4.1 Configuration message usage

Example
Revert Dot device to factory settings and end synchronization. Device responds with successful revert acknowledgement.

Send control message:
RevertToFactorySettings = 0x0301 04 F8
Get acknowledge:
Acknowledge = 0x0308 0000000000000000 F5

5.4.2 Configuration control messages

5.4.2.1 RequestMacAddress

ConfigID	0x01	Size
Direction	To sensor	

Command that requests the MAC address that belongs to the Movella DOT. Carries no further payload in the DATA field. Response is big-endian, see section 5.4.3.1

5.4.2.2 RequestTag

ConfigID	0x02	Size
Direction	To sensor	

Command that requests the current Device Tag of the Movella DOT. Carries no further payload in the DATA field.

5.4.2.3 RequestSerialNumber

ConfigID	0x03	Size
Direction	To sensor	

Command that requests the Serial Number of the sensor device. Carries no further payload in the DATA field.

5.4.2.4 RevertToFactorySettings

ConfigID	0x04	Size
Direction	To sensor	

This control message will prompt the device to revert all settings back to factory settings. The factory settings are the default values that Dot devices are shipped with. Recording data will not be cleared. The revert include the following:

- Device Control Characteristic (defaults as in subsection 2.2).
- Measurement Control Characteristic (defaults as in subsection 3.1).
- Magnetic Field Mapping data.
- Synchronization is stopped.

5.4.2.5 RequestFilterProfileCount

ConfigID	0x05	Size
Direction	To sensor	

Command that requests the available number of filter profiles and their corresponding indices as listed in Table 8. Carries no further payload in the DATA field.

5.4.2.6 RequestFilterProfileName

ConfigID	0x06	Size
ConfigData	Filter profile index	1
Direction	To sensor	

Command to request the filter profile name for a certain index obtained from the RequestFilterProfileCount command. ConfigData field contains the index for which to retrieve the name.

5.4.3 Configuration acknowledge messages

5.4.3.1 RequestMacAddress - Acknowledgement

ConfigID	n/a	Size
Acknowledgement	MAC Address as in the Device Info Characteristic.	6
Direction	To host	

Note - This message is returned in big-endian format. A device with Mac AA:BB:CC:DD:EE:FF will have a six-byte return message starting with DATA[0] = 0xAA and DATA[5] as 0xFF.

5.4.3.2 RequestTag - Acknowledgement

ConfigID	n/a	Size
Acknowledgement	Device Tag as configured in the Device Control Characteristic, maximum length is 16.	LEN
Direction	To host	

5.4.3.3 RequestSerialNumber - Acknowledgement

ConfigID	n/a	Size
Acknowledgement	SerialNumber of the device, as in the Device Info Characteristic.	8
Direction	To host	

5.4.3.4 RevertToFactorySettings - Acknowledgement

ConfigID	n/a	Size
Acknowledgement	Settings restore result	1
	MFM data restore result	1
	Reserved	6
Direction	To host	

The revert command is acknowledged by an 8-byte message detailing the successful operation. The ConfigID is not part of the message format. The different bytes correspond to different subsystems being reverted. The device will return 0x00 if the factory settings are successfully reverted. If a failure occurred then the device will return a 0x01.

5.4.3.5 RequestFilterProfileCount - Acknowledgement

ConfigAckID	0x05	Size
ConfigAckData	Count: Number of filter profiles	1
	Indices: List of profiles indices (one byte per index)	LEN-2
Direction	To host	

The acknowledgement contains a list of indices that each represent an available filter profile. The indices correspond to those listed in Table 8. The shorthand names of these filter profiles can be obtained using the RequestFilterProfileName command.

5.4.3.6 RequestFilterProfileName – Acknowledgment

ConfigAckID	0x06	Size
ConfigAckData	Name of requested filter profile as ASCII string	LEN-1
Direction	To host	

The acknowledgement contains the shorthand name for the given index as listed in Table 8. If the index (obtained through the RequestFilterProfileCount message) is unknown it will return “reserved”.

5.4.4 Configuration notification messages

No configuration notification (asynchronous) messages are present in this revision. The use of notification messages is reserved for future use.

5.5 Message Reference Listing

5.5.1 Recording Messages (section 5.2)

MID	ReID	Message	Direction	Description
0x01	0x01	ACK	To host	Acknowledge message
	0x02	GetState	To sensor	Request sensor recording state
	0x03	FlashProcessBusy	To host	Flash is occupied by other process
	0x30	EraseFlash	To sensor	Request to clear all the recording data space
	0x33	StoreFlashInfoDone	To host	Flash information has been updated
	0x34	FlashFull	To host	Recording flash space is full
	0x35	InvalidFlashFormat	To host	Recording flash format is invalid
	0x40	StartRecording	To sensor	Start recording
	0x41	StopRecording	Both	Stop recording or recording stopped
	0x42	RequesetRecordingTime	To sensor	Request recording time
	0x43	RecordingTime	To host	Recording time values
	0x50	RequetFlashInfo	To sensor	Request recording flash information
	0x51	ExportFlashInfo	To host	Export flash information
	0x52	ExportFlashInfoDone	To host	Export flash information done
	0x60	RequestFileInfo	To sensor	Request recording file information by FileIndex
	0x61	ExportFileInfo	To host	Export file information
	0x62	ExportFileInfoDone	To host	Export file information done
	0x63	NoRecordingFile	To host	No recording file (with this FileIndex).
	0x70	RequestFileData	To sensor	Request recording file data based on FileIndex
	0x71	ExportFileData	To host	Export recording file data based on FileIndex
	0x72	ExportFileDataDone	To host	Export file data done
	0x73	StopExportData	Both	Stop export file data or export stopped
	0x74	SelectExportData	To sensor	Configure export data options

	0x75	Retransmission	To sensor	Retransmit all the data from the RetransDataNumber packet
	0x76	ExportFileDataInvalid	To host	Invalid data packet due to internal data checksum or preamble check fail

5.5.2 Synchronization Message (section 5.3)

MID	SyID	Message	Direction	Description
0x02	0x01	StartSync	To sensor	Start synchronization
	0x02	StopSync	To sensor	Stop synchronization
	0x03	ACK	To host	Acknowledge message
	0x08	GetSyncStatus	To sensor	Get synced or un-synced status
	0x50	StopSyncResult	To host	Notification of stop sync result
	0x51	SyncStatus	To host	Notification of sync status

5.5.3 Configuration Messages (section 5.4)

MID	ID/AckID	Message	Direction	Description
0x03	0x01	RequestMAC	To sensor	Get MAC
	n/a	RequestMAC ack	To host	Response with data
	0x02	RequestTag	To sensor	Get Tag
	n/a	RequestTag ack	To host	Response with data
	0x03	RequestSerialNumber	To sensor	Get Serial Number
	n/a	RequestSerialNumber ack	To host	Response with data
	0x04	Revert device	To sensor	Revert device back to factory settings
	n/a	Revert device results	To host	Response with results
	0x05	RequestFilterProfileCount	To sensor	Get number of available filter profiles
	0x05	RequestFilterProfileCountAck	To host	Response with data
	0x06	RequestFilterProfileName	To sensor	Get filter profile name for index
	0x06	RequestFilterProfileNameAck	To host	Response with data