

Muscle Oxygen Data in FIT Files

Date: November 9, 2016

Revision: A

Purpose:

The purpose of this document is to specify standards formats and techniques for storing data from one or more Muscle Oxygen sensors in a single FIT file. The capability to support multiple sensors of the same type has not been exploited to any large extent previously so the FIT structure and many software packages are not yet well adapted to support it. Also, the new developer data format is still very new and does not support all of the same functionality in the same ways as the MO2 Profile data format. This document is intended to provide practical guidance to maximize functionality within these constraints.

Formats:

This document will refer to 2 different formats for storing Muscle Oxygen Data in the Fit file. The format that is specified in the ANT+ Muscle Oxygen Profile document number D00001555 will be referred to as the “MO2 Profile Format”. The format that is used by Garmin Connect IQ devices will be referred to as “Developer Data Format”.

Both formats support storing data from multiple sensors.

Developer Data Format ConnectIQ Setup:

Garmin’s ConnectIQ currently supports only Session, Lap, and Record Messages and these are stored in Developer Data Format. When creating fields in ConnectIQ to store this data, the following parameters can be specified.

- name (string)
- fieldId (number)
- type (type)
- options
 - :count (number)
 - :mesgType (number)
 - :units (string)
 - :nativeNum (number)

Developer Data Format for MO2 Record, Lap, and Session Messages:

This section describes how MO2 data should be formatted from ConnectIQ to ensure the best interoperability of the FIT file data.

- name (string) – The name string can be made of a concatenation of several pieces of information to help identify the data. The following format will be used by Moxy CIQ data Fields and Applications.
 - [CIQ Data Field] + [SmO₂ or THb] + “Sensor” + [serial_number] + “on” + [body_location]
 - [CIQ Data Field] will be “1st”, “2nd”, “3rd”, etc.
 - “SmO₂” will be used for saturated_hemoglobin_percent records and “THb” will be used for total_hemoglobin_conc records
 - [serial_number] will be the ANT+ deviceNumber obtained when the sensor connects
 - [body_location] will be the string associated with the sensor_position value that is set in the App Settings of the CIQ Data Field.
 - An example string would be “1st THb Sensor 317 on Left Quad”
- fieldId (number)
 - The fieldId number will be set according to the following table

Table 1 – Muscle Oxygen Sensor fieldIds

Muscle Oxygen Sensor	saturated_hemoglobin_percent (SmO ₂) Record Data fieldId	total_hemoglobin_conc (THb) Record Data fieldId	Other Record, Lap, or Session Data fieldIds
1st	0	1	2-31
2nd	32	33	34-63
3rd	64	65	66-95
4th	96	97	98-127
5th	128	129	130-159
6th	160	161	162-191
7th	192	193	194-223
8th	224	225	226-255

- type (type) - The data type should be set to DATA_TYPE_FLOAT for all Record, Lap, and Session data.
- options
 - :count (number) – Not used
 - :msgType (number)
 - Record Data - Set to MSG_TYPE_RECORD
 - Lap Data - Set to MSG_TYPE_LAP
 - Session Data - Set to MSG_TYPE_SESSION
 - :units (string)
 - Set to “%” for saturated_hemoglobin_percent
 - Set to “THb” for total_hemoglobin_conc
 - :nativeNum (number) – These values come from the FIT SDK profile.xlsx spreadsheet Activity File Messages section. They are listed for reference in the table below.

Table 2 - :nativeNum Values

Message Type	Message	:nativeNum
Record	total_hemoglobin_conc	54
Record	total_hemoglobin_conc_min	55
Record	total_hemoglobin_conc_max	56
Record	saturated_hemoglobin_percent	57
Record	saturated_hemoglobin_percent_min	58
Record	saturated_hemoglobin_percent_max	59
Lap	avg_total_hemoglobin_conc	84
Lap	min_total_hemoglobin_conc	85
Lap	max_total_hemoglobin_conc	86
Lap	avg_saturated_hemoglobin_percent	87
Lap	min_saturated_hemoglobin_percent	88
Lap	max_saturated_hemoglobin_percent	89
Session	avg_total_hemoglobin_conc	95
Session	min_total_hemoglobin_conc	96
Session	max_total_hemoglobin_conc	97
Session	avg_saturated_hemoglobin_percent	98
Session	min_saturated_hemoglobin_percent	99
Session	max_saturated_hemoglobin_percent	100

Developer Data Format for device_info Messages:

ConnectIQ doesn't specify a method of encoding device_info data in the Developer Data format. However, it is possible to encode this information using a custom format. This section describes a standard method for encoding this data. It is optional to include this information.

This data will be encoded as a byte array to maximize the use of the available space. Each CIQ data field is limited to a total of 32 bytes for all session messages contributed by the data field including session data like avg_total_hemoglobin_conc. It is up to the CIQ App developer to stay below that limit.

This data will be encoded using a custom session message with the parameter as set below.

- name (string) – "CIQ_device_info"
- fieldId (number) – Use a field id number from the "Other Record, Lap, or Session Data fieldIds" in the fieldId table above for the appropriate Muscle Oxygen Sensor.
- type (type) – DATA_TYPE_UINT8
- options
 - :count (number) – Set to the number of Bytes actually used.
 - :mesgType (number) – Set to MESG_TYPE_SESSION
 - :units (string) – Not Used
 - :nativeNum (number) – Not Used

A byte array can be written to the session message with the following information contained in each byte. The Byte array can be made smaller than 14 Bytes by omitting the highest numbered Bytes. However, the Array Elements cannot be reassigned to different fields.

Table 3 – device_info Byte Array

Byte / Array Element	device_info FIT Field	Value
0	device_type	31
1	sensor_position	Used to indicate the location of the sensor on the body using the body_position enum
2	serial_number LSB	ANT+ deviceNumber for the Sensor
3	serial_number 2nd Byte	
4	serial_number 3rd Byte	
5	serial_number MSB	
6	manufacturer LSB	76 for Moxy 98 for BSX
7	manufacturer MSB	
8	battery_status	<div><div>new</div><div>good</div><div>ok</div><div>low</div><div>critical</div><div>charging</div><div>unknown</div><div>1</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div></div>
9	software_version LSB	
10	software_version MSB	
11	hardware_version	
12	product LSB	
13	product MSB	

MO2 Profile Format for Record, Lap, and Session Messages:

The MO2 Profile describes the format for encoding multiple MO2 sensor data in a FIT file. This section adds some clarifications.

If data from multiple MO2 sensors is stored in the FIT file, a separate device_info message with a unique device_index must be created for each MO2 sensor.

Record data from multiple MO2 sensors will be entered as separate record messages even if they have the same time stamp. When multiple MO2 sensors are used, the device_index must be included in the record messages.

Session and Lap information from the multiple MO2 sensors are recorded in an array as opposed to separate messages. This is done so that there will only be one lap message per lap event and one session message per session. The numerical value of the device_index determines which sensor data

gets stored in which elements of these arrays. The device_index=0 is only used for MO2 data when the MO2 sensor writes the FIT file itself. When device_index=0 is used or when no device_index is used the lap and session data is stored in Array Element 0.

When another device writes the MO2 data in the FIT file from one or multiple MO2 sensors, the device_index must be greater than 0. The lap and session data arrays must have element 0 written as invalid (0xFFFF for UINT16). The lap and session data from the sensor with the lowest value of device_index will be stored in array element 1.

Table 4 – MO2 Profile Format device_index and Array Elements for Lap and Session Data

	FIT file is created by MO2 sensor		FIT file created by some other device	
	device_index	Array Element	device_index	Array Element
1st MO2 Sensor	Use device_index = 0 or don't use device_index at all	0	Must use device_index > 0. The lowest device_index will be the 1st MO2 Sensor	1
2nd MO2 Sensor			Must use device_index > than the one used for 1st MO2 sensor	2
3rd MO2 Sensor			Must use device_index > than the one used for 2nd MO2 sensor	3
4th MO2 Sensor			Must use device_index > than the one used for 3rd MO2 sensor	4
5th MO2 Sensor			Must use device_index > than the one used for 4th MO2 sensor	5
6th MO2 Sensor			Must use device_index > than the one used for 5th MO2 sensor	6
7th MO2 Sensor			Must use device_index > than the one used for 6th MO2 sensor	7
8th MO2 Sensor			Must use device_index > than the one used for 7th MO2 sensor	8

Priority – Profile Format vs Developer Data Format:

It is possible for a FIT file to contain both Profile Format and Developer Format Data. An example would be the Garmin 735XT which supports one MO2 sensor natively and also supports CIQ data fields.

If a decoding application identifies the data only by whether it is the 1st, 2nd, 3rd or ... sensor then it's possible that both sensors could be identified as the 1st sensor. It is up to the user to avoid this problem when they set up their device. If they are using the natively supported MO2 sensor and a data field, they should install the "Moxy 2nd Sensor" data field.

However if the user fails to do this correctly and if there is a conflict that the application does not overcome by using the device_info data, the decoding application should give priority to the Profile Format Data.

Priority – More Data than the Decoding Software Supports

If a data decoding software supports fewer MO2 sensors than are present in the FIT file, the software should use the “1st MO2 Sensor” data first and the “2nd MO2 Sensor” data second and so on and omit the higher MO2 Sensors that it doesn’t have support for.