Head tracker protocol
General information

PRELIMINARY: Subject to change without notice. Last updated 26th June 2019.

Form of messages

The head tracker communicates using MIDI System Exclusive messages:

```
f0  00 21 42  <message type>  [other data]  f7
```

Data addressed to the head tracker will have a type below 0x40; data from it has a type greater than or equal to 0x40.

Another category of messages, beyond the scope of this guide, manage firmware version control and upgrading. They will not interfere.

Instant gratification

All sensors and data output are turned off by default. Most users will simply want a continual output of yaw, pitch, and roll, at around 50Hz. The transactions required to do this, in the same way as the demo software, are as follows:

Reset, turn on all sensors except the magnetometer, and enable 3DOF tracking at 50Hz:

```
f0  00 21 42  00  00 4b 01 01  f7
```

The head tracker will start to provide data. Zero the head tracker (when ready):

```
f0  00 21 42  01  00 01  f7
```

14- and 16-bit numbers

14-bit numbers are two's complement signed and given in big-endian form. Where used, 16-bit numbers are encoded similarly, but as three bytes:

First byte	`0000 00MX`
Second byte	`0XXX XXXX`
Third byte	`0XXX XXXL`

where M is the most-significant bit and L is the least-significant. Converting incoming data to a signed integer is therefore achieved in the following way:

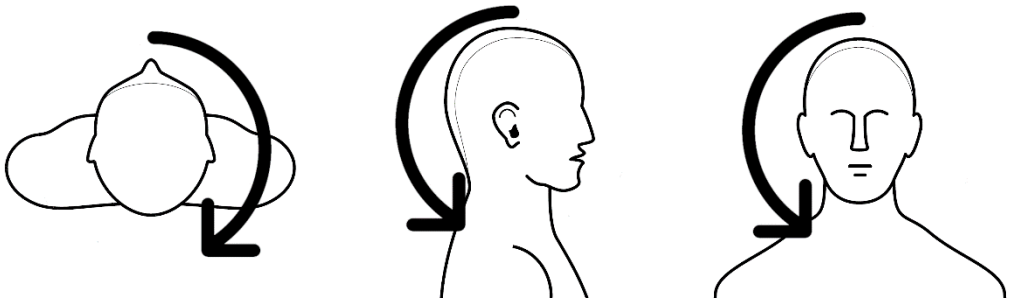
```
int i = (d[0] << 14) + (d[1] << 7) + d[2];
if (i & 0x8000) { i -= 0x10000; }
```

Erratum about the button

There *is* a push-button on the head tracker, but it has been disabled and encased in this version of hardware, as it was made too small to be particularly useful. Messages that concern the button's operation will not do anything for now, but will be activated at a later date.

Tait-Bryan angles

The output of the head tracker is expressed in Tait-Bryan angles. This is the formal name for the class of Euler angle that is expressed as yaw, pitch, and roll. Precise head orientation is obtained by applying the three rotations in the following order. The arrows indicate the direction of a positive rotation.



Axes of rotation are fixed with respect to the head, not the world.

The order is important: 'pitch 45 degrees then roll 45 degrees' places one's head in a different attitude from 'roll 45 degrees then pitch 45 degrees'.

There are a number of ways to develop the necessary transformations on points in space, depending on whether the implementer prefers to work in matrices, quaternions, or directly in three-dimensional trigonometry. (In terms of operational complexity, any approach works out approximately the same.) This page on Wikipedia is a useful touchstone.

Translation

In the head tracker's 6DOF system, a positive X movement moves the listener towards the right-hand wall; a positive Y movement moves them towards the front wall; a positive Z movement moves them towards the ceiling. These movements are fixed relative to the world.

Head tracker protocol
Messages in detail

Message 0 : Configure sensors and processing pipeline

```
f0  00 21 42  00  <parameter> <value> [<parameter> <value> ...]  f7
```

Message 0 / Parameter 0 :
Sensor setup

6	5	4	3	2	1	0
RESET	RATE		FAR-END SENSOR	TOP-END MAG	TOP-END ACC	NEAR-END SENSOR

bit 6 RESET
1 : reset values in other registers to their power-on defaults

bits 5:4 RATE : sensor output rate
00 : 50-52Hz (default)
01 : 25-26Hz
10 : 100-104Hz
11 : reserved (do not use)

bit 3 FAR-END SENSOR
0 : turn off far-end sensor (default)
1 : reset/activate far-end sensor

bit 2 TOP-END MAG
0: turn off top-end magnetometer (default)
1: reset/activate top-end magnetometer sensor

bit 1 TOP-END ACC
0: turn off top-end accelerometer (default)
1: reset/activate top-end accelerometer sensor

bit 0 NEAR-END SENSOR
0 : turn off near-end sensor (default)
1: reset/activate top sensor

Note: writing to this address will always reset the sensors, regardless of its current and intended states.

Message 0 / Parameter 1 :
Data output and formatting

6	5	4	3 2	1 0
NOCAL	0	RAW	FORMAT	TRACKING

bit 6 NOCAL : ignore calibration data
0 : calibrated data is sent / used in the pipeline (default)
1 : calibration data is ignored (used for checking calibration)

bit 4 RAW : output raw sensor data
0 : do not output raw data (default)
1 : output raw data at full sensor rate

bits 3:2 FORMAT : output format for orientation data
00 : Tait-Bryan angles (default)
01-11 : reserved (do not use)

bits 1:0 TRACKING : enable head tracking
00 : head tracking is turned off (default)
01 : enable output in 3DOF mode: yaw/pitch/roll
10 : enable output in 6DOF mode: yaw/pitch/roll/x/y/z
11 : reserved (do not use)

Message 0 / Parameter 2 :
Push-button function

6	5	4	3	2	1	0
BUTTON-MIDI	LONG-PRESS			SHORT-PRESS		

bit 6 BUTTON-MIDI : send MIDI in response to button pushes
0 : do not send button-related MIDI
1: send button-related MIDI (default)

bits 5-3 LONG-PRESS : configure action of a long button press
000 : no action (default)
001 : zero the head tracker
010 : switch the chirality mode between left- and right-ear
011 – 111 : reserved (do not use)

bits 2:0 SHORT-PRESS : configure action of a short push-and-release
000 : no action
001 : zero the head tracker (default)
010 : switch the chirality mode between left- and right-ear
011 – 111 : reserved (do not use)

Message 1 : Control during use

```
f0  00 21 42  01  <parameter> <value> [<parameter> <value> ...]  f7
```

Message 1 / Parameter 0 : Zeroing

6	5	4	3	2	1	0
0	0	0	0	0	0	ZEROING

bit 0 ZEROING : Set zero point
0 : No action
1 : Use most recent stable data as zero (level, straight ahead)

Message 1 / Parameter 1 : Chirality

6	5	4	3	2	1	0
RIGHTEAR-SAVE	0	0	0	0	0	RIGHTEAR

bit 6 RIGHTEAR-SAVE
1 : The RIGHTEAR in this message will be the new power-on default

bit 0 RIGHTEAR
0 : Power cable is positioned over left ear (default)
1 : Power cable is positioned over right ear

Note: this inverts the polarity of pitch and roll rotation, and X and Y translation, while yaw and Z remain the same.

Message 2 : Factory calibration

```
f0  00 21 42  02  <dataset> [<16-bit words of data>]  f7
```

Calibration data is kept in non-volatile storage, so careless use of this call can potentially de-calibrate the head tracker. The 16-bit format is used to convey fixed-point numbers as follows:

dataset	bytes to follow	Meaning
0	15 { x, y, z }	Magnetometer offset, scaled so that 0x8000 = -1.0
1	15 { x, y, z }	Magnetometer scalar, applied after offset. scaled so that 0x400 = 1.0

If data is omitted or the inexact amount is provided, the dataset is read back.

Message 3 : Read or write raw I^2C

```
f0  00 21 42  03  <I2C address high> <I2C address low> [<data or byte count>]  f7
```

Note: you will almost certainly never need to use this call.

All information transmitted in this kind of packet, including the device address and any byte count, is encoded as nibbles. The most significant is given first.

Even slave address: write raw I^2C. Encode slave address, then data byte-by-byte.

```
e.g. write AA 55 to device 50:
f0 00 21 42 03 05 00 0A 0A 05 05 f7
```

Odd slave address: read raw I^2C. Encode slave address, then provide number of bytes to read.

```
e.g. read 4 bytes from device 50:
f0 00 21 42 03 05 01 00 04 f7
```

Message 64 : Head tracker data response

Message 64 / Parameter 0 :
Orientation

```
f0  00 21 42  40 00  <14-bit yaw> <14-bit pitch> <14-bit roll>  f7
```

The 16-bit data is given in radians, such that there are 4 sign/unit bits and 10 fractional bits. This gives a maximum accuracy of 1/1024 radians. For example:

180 degrees → PI radians → 0x0C91

How precise the head tracker actually is can be left as an exercise for the experimenter.

Example: to obtain the data in degrees from the raw 14-bit input:

```
float yaw_degrees = 0.0559529f * yaw_raw;
```

Message 64 / Parameter 1 :
Position

```
f0  00 21 41  40 01  <14-bit x> <14-bit y> <14-bit z>  f7
```

This data is given in metres, such that there is 2 sign/unit bits and 12 fractional bits. This gives a maximum accuracy of 1/4096 metres.

The current design of head tracker is not intended to provide freedom to wander: it is intended for use in a seated position, and the data is filtered to assume this. In normal use, the unit bits will not do much of interest.

Message 65 : Raw sensor data response

```
f0  00 21 42  41  <sensor id> <timestamp>  <16-bit x> <16-bit y> <16-bit z>  f7
```

Sent asynchronously as soon as data is ready. The timestamp increments every millisecond, so it wraps every 128ms. This is frequent enough that there will be no more than one wrap-around between sensor readings.

Sensor ID	Device	Sensor
0	Near-end sensor	accelerometer
1		gyroscope
2	Top sensor	accelerometer
3		magnetometer
4	Far-end sensor	accelerometer
5		gyroscope

Message 66 : Calibration data response

```
f0  00 21 42  42  <dataset> <multiple bytes of data>  f7
```

Sent in response to a Message 2 read request.

Message 67 : I^2C data response

```
f0  00 21 42  43  <nibble-encoded I2C data>  f7
```

Sent in response to a Message 3 read request.

Message 68 : Button event

```
f0  00 21 42  44  <button state>  f7
```

button state	meaning
0	Release after press
1	Press
2	Release after long press
3	Long press