Morph Device Name: "LE-Morph InfiniConnect"

Service UUID: "00001100-D102-11E1-9B23-00025B00A5A5"

Command Characteristic UUID: "00001101-D102-11E1-9B23-00025B00A5A5"
Response Characteristic UUID: "00001102-D102-11E1-9B23-00025B00A5A5"

Morph BLE COMMANDS

Byte 0	Byte 1	Byte 2								Byte 3								Bytes 4+
-	-	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	-
Vendor ID		Feature ID								PDU Type		PDU Specific ID						Payload

Vendor ID: will be either 0x0a66 or 0x001d
Feature ID: Feature-specific unique identifier
PDU Type

- Command: 00
- Notification: 01
- Response: 10
- Error: 11

PDU Specific ID: Feature-specific command ID. These are only unique within the Feature.
Payload: Optional payload

==========

Get Morph command handler version - 0x00

Command Write: 0x0a660000

Response Notify: 0x0a66010001

This is a command that can be sent to QCC to check that the Morph command handling is running. Payload received should be 0x01

==========

Get Volume - 0x01

Command Write: 0x0a660001

Reponse Notify: 0x0a6601013 0

Volume received above is payload 0x30, or 48 in decimal.
(Possible range of volume is 0-0x7f, or 0-127 in decimal)

==========

Change Volume - 0x02

Command Write: 0x0a6600024 0

Response Notify: (none)

Payload 0x40 is volume to set

==========

Toggle Mic Mute - 0x03

Command Write: 0x0a660003

Response Notify: (none)

==========

Get Mic Mute State - 0x04

Command Write: 0x0a660004

Response Notify: 0x0a6601040 0

Payload 0x00 means Mic Mute OFF (i.e. mic is not muted)
0x01 means Mic Mute ON (i.e. mic is muted)

==========

Start Pairing Mode - 0x05

Command Write: 0x0a660005

Response Notify: (none)

==========

Get is Handset device connected - 0x06

Command Write: 0x0a660006

Response Notify: 0x0a6601060**1**

Payload is 0x00 if no BT device connected, or 0x01 if there is

==========

Get current Handset device connected - 0x07

Command Write: 0x0a660007

Response Notify: 0x0a66010704b167380ab1000000074d6920413100

MAC Address is 6 bytes: 04b167380ab1
BT Profiles Mask is next 4 bytes: 00000007

All characters after BT Profiles Mask are device name in ASCII char, null terminated, ie 0x4d6920413100
This ASCII text is “Mi A1” which is the bluetooth name of a test Handset device
http://string-functions.com/hex-string.aspx

==========

Get battery level - 0x08

Command Write: 0x0a660008

Response Notify: 0x0a660108434b

Response payload 0x434b, first byte is L earbud 0x43, second byte is R earbud 0x4b
Battery level in decimal is a %, i.e. L earbud is 67%, R earbud is 75%

==========

Connect to a (previously paired) Handset - 0x09

Command Write: 0x0a660009**04b167380ab1**00000007

MAC Address is 6 bytes: 04b167380ab1
BT Profiles Mask is next 4 bytes: 00000007

The MAC Address and BT Profiles Mask should have been saved from command 0x07's response

==========

Disconnect All Handsets - 0x0A

Disconnects all currently connected handsets (they will still remain paired)
This will also cause a disconnection of ESP32.

==========

Disconnect Current Handset - 0x0B

Disconnect the Handset that is currently connected as returned by command 0x07
This will not cause a disconnection of ESP32.

Note: QCC needs to disconnect any current connected handset, before an attempt to make a new connection to another handset. If there is still an existing connected handset, a new connection attempt to another handset will fail

==========

Delete Paired Handset - 0x0C

QCC will remove the device from its list of trusted paired devices

Command Write: 0x0a66000C640BD79117DE
MAC Address is 6 bytes: 640BD79117DE

Response: 0x0a66010C01640BD79117DE

Payload is

- 0x00 if the handset was deleted successfully from the paired devices list,
- 0x01 if there was an error and no deletion occurred: because the device is not present on the paired list
- 0x02 if there was an error and no deletion occurred: because the device is in the list but is currently connected so cannot be deleted.

Followed by 6 bytes of the device mac address concerned: 640BD79117DE

==========
Play Tone - 0x0E

Play a single tone through the primary earbud, can be used for our own user notifications. (No tone is played through the secondary earbud).

Command Write: 0x0a66000E01

Payload byte is number of beeps, eg 0x01 above is 1 beep. Send payload 0x02 for 2 beeps, and so on

Response: -

==========
Media Play / Pause toggle - 0x0F

Command Write: 0x0a66000F

Response: -

==========
Get GamingMode state - 0x10

Command Write: 0x0a660010

Response Notify: 0x0a6601100 1

Payload is 0x00 if GamingMode is OFF, or 0x01 if it is ON

==========
Toggle GamingMode state - 0x11

Command Write: 0x0a660011

Response: -

Notes:
User can also triple-tap on the earbud touch sensor to toggle gaming mode.

==========

Get Num Paired Handsets - 0x12

Command Write: 0x0a660012

Response Notify: 0x0a660112060D

Response payload 0x060D, first byte 0x06 is num paired handsets, second byte 0x0D is max paired handsets supported

==========

Get Paired Handsets - 0x13

Retrieve from QCC the MAC addresses of all the devices currently paired

Command Write: 0x0a6600130005

Response: 0x0a66011304b167380ab1640BD79117DE…

Command:
Payload first byte 0x00 is index of first device to get
Payload second byte 0x05 is num of devices to get

Response:
MAC Address of Device 1 is first 6 bytes payload: 04b167380ab1
MAC Address of Device 2 is next 6 bytes payload: 640BD79117DE
And so on..

The payload is always a multiple of 6 bytes (for each device paired)

Important Observations:
2 devices in the list are always the MAC addresses of the L and R earbuds, i.e.:
00 02 5b 00 ff 01
00 02 5b 00 ff 00

Another 2 in the list is the ESP32, i.e.
94 b9 7e 5a 49 02
94 b9 7e 5a 49 01

But these may or may not be the first few devices. They could appear as device 2, 3, and 5, 6 in the list, if there are several paired devices
Device 1 and 4 would be other Handset devices that are paired in this case

==========

Get Primary Earbud is L/R
Command Write: 0x001d0200

Response Notify: 0x001d030001

Payload is 0x00 if LEFT is primary, or 0x01 if RIGHT is primary

Note: user can also identify the primary earbud by playing a tone via command 0x0E. The tone will be played only through the primary earbud (not the secondary)

==========

NOTIFICATIONS

==========

Volume Change

NotificationID - 0x01
Payload - 1 byte - 0x00 to 0x7f - new volume

Example notification received - 0x0a6600814 8
Payload shows new volume is 0x48

==========

Handset Connected - Remote Initiated

- eg clicking on the Windows Laptop bluetooth settings “Connect” button to Morph InfiniConnect

NotificationID - 0x02
Payload - [none]

Example notification received - 0x0a660082

It appears this will be triggered multiple times, probably because each time a new BT profile is connected (HFP, AVRCP, A2DP, which results in profile mask 0x7), after the initial BT transport connection

==========

Handset Disconnected - Remote Initiated

- eg clicking on the Windows Laptop bluetooth settings “Disconnect” button of Morph InfiniConnect

NotificationID - 0x03
Payload - [none]

Example notification received - 0x0a660083

==========

Handset Connection Attempt Complete - Local Initiated

- eg ESP32 Sending GAIA command 0x09 to QCC to connect to MAC address

- Note this notification is that an “attempt” is complete. It may have completed either successfully or unsuccessfully.

NotificationID - 0x04
Payload - [none]

Example notification received - 0x0a660084 - followed by 0x0a660082

This can be tested by using console terminal (i), send raw bytes to send the command to connect to the (previously paired) teamviewer Laptop which has mac address 0x28c6 3f93 667d- 0x0a66000928c63f93667d00000007

This notification received signals that the attempt by QCC to connect to the laptop has completed. This may have either completed successfully or failed. (eg device is not in range, timeout)

After 0x0a660084 is received, the notification 0x0a660082 is observed to be received twice (likely once for BT transport connection, the second for BT profiles connect). This indicates that the connection is successful to the remote device.

Example notification received - 0x0a660084 - followed by 0x0a660083

For case where attempt by QCC to connect to Laptop, i.e. a valid mac address of a device in range (but Laptop had not previously paired the QCC, thus rejects it)

Example notification received - 0x0a660084 - followed by no other notification

For case where attempt by QCC to connect to an invalid mac address (or not in range).
Or QCC did not disconnect from another currently connected handset first, so attempt to connect to a new handset cannot complete.

==========

Handset Disconnection Attempt Complete - Local Initiated

- eg ESP32 Sending GAIA command 0x0B to QCC to disconnect current handset
- Note this notification is that an “attempt” is complete. It may have completed either successfully or unsuccessfully.

NotificationID - 0x05
Payload - [none]

Example notification received - 0x0a660085 - followed by 0x0a660083

For case where QCC attempts to disconnect from current connected handset (using command 0x0a66000B). After disconnection completes, 0x0a660083 is received.

Note if QCC attempts to disconnect all handsets using 0x0a66000A command, no notifications are received because the ESP32 is also disconnected right away.

==========

Mic Mute State Change

NotificationID - 0x06
Payload - 0x01 (new state is muted), 0x00 (new state is NOT muted)

==========