HEXIWEAR UART MESSAGING API

The serial interface between host processor MK64 and bluetooth processor KW40 supports UART communication using no parity, 8 data bits, and 2 stop bits. Baud rate is 230400 BPS.

Figure 1 shows the packet format for all serial messages exchanged between MCUs:

Start Byte 1 (1 byte)	Start Byte 2 (1 byte)	Packet Type (1 byte)	Data Length (1 byte)	Data (N bytes)	Trailer (1 byte)

Figure 1.

- Start Byte 1 - Always 0x55
- Start Byte 2 - 0xAB if confirmation with OK packet requested (packet with type *packetType_OK*), 0xAA if not
- Packet Type - Message type information (see tables below)
- Data Length - Number of data bytes that follow (maximum length is 23)
- Data - Message data bytes
- Trailer - Always 0x45

Sender can request confirmation of packet previously sent by setting *Start Byte 2* to 0xAB. In such case sender waits maximum *gHostInterface_retransmitTimeout* miliseconds (default value is 1000) before retransmitting previously sent packet. Maximum number of retransmitions is *gHostInterface_retransmitCount* (default value is 3).

Types of packets which could be sent from MK64 MCU is listed in Table 1.

Type	Description
packetType_batteryLevel	**Battery level (percentage).** Data Length = 1 byte
packetType_accel	**Accel measurement for x, y, z coordinate (range +/- 4g).** Data Length = 3 byte
packetType_gyro	**Gyro measurement for x, y, z coordinate (range +/- 256 deg/sec).** Data Length = 3 byte
packetType_magnet	**Magnet measurement for x, y, z coordinate.** Data Length = 3 byte
packetType_ambiLight	**Level of light (percentage).** Data Length = 1 byte
packetType_temperature	**Temperature measurement (Celsius).** Data Length = 2 byte
packetType_humidity	**Humidity measurement (percentage).** Data Length = 2 byte
packetType_pressure	**Pressure measurement (Pascal).** Data Length = 2 byte
packetType_heartRate	**Heart Rate value (beats per minute).** Data Length = 1 byte
packetType_steps	**Number of steps measured by pedometer.** Data Length = 2 byte
packetType_calorie	**Value of kcal.** Data Length = 2 byte

packetType_alertOut	**Used for sending alerts from watch to phone.** Data Length is variable.
packetType_otapKW40Started	**Start OTAP procedure for KW40 MCU.** Data Length = 0 byte
packetType_otapMK64Started	**Start OTAP procedure for MK64 MCU.** Data Length = 0 byte
packetType_tsiGroupToggleActive	**Change active group (pair) of vertical touch sense electrodes.** Data Length = 0 byte
packetType_tsiGroupGetActive	**Request info about active group (pair) of vertical touch sense electrodes.** Data Length = 0 byte
packetType_advModeGet	**Request info about status of advertising process.** Data Length = 0 byte
packetType_advModeToggle	**Change status of advertising process.** Data Length = 0 byte
packetType_appMode	**Device working mode.** Data Length = 1 byte
packetType_linkStateGet	**Request info about status of bluetooth link.** Data Length = 1 byte
packetType_OK	**Send if bluetooth MCU waiting for confirmation of packet sent.** Data Length = 0 byte

Table 1.

Types of packets which could be sent from KW40 bluetooth MCU is listed in Table 2.

Type	Description
packetType_pressUp	**Detected the touch of the up electrode.** Data Length = 0 bytes
packetType_pressDown	**Detected the touch of the down electrode.** Data Length = 0 bytes
packetType_pressLeft	**Detected the touch of the left electrode.** Data Length = 0 bytes
packetType_pressRight	**Detected the touch of the right electrode.** Data Length = 0 bytes
packetType_alertIn	**Used for sending alerts from phone to watch.** Data length is variable.
packetType_passDisplay	**Send passkey to be displayed on watch (for pairing procedure).** Data Length = 3 bytes
packetType_otapCompleted	**Sent when OTAP procedure is finished successfully.** Data Length = 0 bytes
packetType_otapFailed	**Sent when OTAP procedure is finished unsuccessfully.** Data Length = 0 bytes

packetType_tsiGroupSendActive	**Info about active group (pair) of vertical touch sense electrodes.**
	Data Length = 1 byte
	Data Field 0:
	  0 - *Active left pair*
	  1 - *Active right pair*

packetType_advModeSend	**Info about status of advertising process.** Data Length = 1 byte Data Field 0: 0 - *Advertising is disabled*

	1 - *Advertising is enabled*

packetType_linkStateSend	**Info about status of bluetooth link.**
	Data Length = 1 byte
	Data Field 0:
	  0 - *Device is disconnected*
	  1 - *Device is connected*

packetType_notification	**Notification info received from iOS phone, through ANCS service.**
	Data Length = 2 byte
	Data Field 0: ANCS event ID:
	  0 - *Added*
	  1 - *Modified*
	  2 - *Removed*
	Data Field 1: ANCS category ID:
	  6 - *Email*
	  2 - *Missed Call*
	  4 - *Social*

packetType_buildVersion	**Versions of last burned MCU images.** Data Length = 1 byte
packetType_OK	**Send if host MCU waiting for confirmation of packet sent.** Data Length = 0 byte

Table 2.

NOTE:

1) Please see *HEXIWEAR BLUETOOTH SPECIFICATIONS* for detailed explanation of service specific packet types.

2) Header file host_mcu_interface.h contains declaration of hostInterface_packetType_t enum, with assigned value for every type listed above.