GLÜCK Industrie-Elektronik GmbH

Zeichnungsnummer:	505014.001.7	**Blatt: 3**		**Rev.:**	*V1.5*
Zeichnungstitel:	***BLE UART Protocol***			**Datum:**	*25.01.2021*
Bezeichnung:	*Bownce A-Sample*			**Geändert:**	*07.10.2022*
Halbfabrikats/ Fertiggeräte Nr.:	*505014.001*				
Firma:	*Bownce*				
Kundenartikel Nr./ Bez.:	*n.a.*				
Sachbearbeiter:	*ThG*				
Geprüft:	*S.H.*				

Zeichnungsnummer: 505014.001.7.B3;V1.5 Stand 07.10.2022

Änderungshistorie

Datum	Version	Art der Änderung
25.01.21	1.0	Erstellt
19.03.21	1.0	Publish V1.0
23.03.21	1.1	Added Byte [10]: transmission counter to 2.1 Data Frame
21.04.21 26.07.21	1.2	Data Frame for Anchor specified Data Frame for Ball extended (2.2)
17.11.21	1.3	Data Frame for Ball extended (Swing period)
29.11.21	1.4	Offline session handling implemented
07.10.2022	1.5	Calculation of rope-tension fixed

Inhaltsverzeichnis

0 Abstract

This document describes the BLE UART Protocoll used in bownce ball and anchor.

1 General format

1.1 General format of payload frames

From Ball/Anchor to APP or from APP to Ball/Anchor

Start of frame	Payload Byte 0	Payload Byte 1	Payload Byte 2	Payload Byte 3	Payload Byte 4	Payload Byte 5	….	Payload Byte 18	End of frame

Start of frame (SOF): depends on payload

End of frame (EOF) : 0x2C (‘)

1.2 General format of command frames

From APP to Ball/Anchor

Start of frame	Command Byte	End of frame

In case the command triggers action on the device that does not result in responding with data frames, the device responds with ACK or NACK

(Such commands may be used in integration testing and/or during EOL testing)

ACK Frame:

The ACK Frame is a single Byte containing **0x40 ("@")**

NACK Frame

The NACK Frame is a single Byte containing **0x41**

2 Specific frame format

2.1 Data frame Anchor

The Data frame is used to transmit data from anchor to app

SOF Byte[0] : 0x23 „#“

Payload Byte [1]: tension [0]
Payload Byte [2]: tension [1]
Payload Byte [3]: battery level [0]
Payload Byte [4]: transmission counter [0]

EOF Byte[5] : 0x27

Where

Rope-tension in [g] = (tension [0] * 255) + tension [1]

Battery level in 100mv Increments

2.2 Data frame Ball

The Data frame is used to transmit data from ball to app

SOF Byte[0] : 0x23 „#“

Payload Byte [1]: timestamp [0]
Payload Byte [2]: timestamp [1]
Payload Byte [3]: timestamp [2]
Payload Byte [4]: hitcounter [0]
Payload Byte [5]: hitcounter [1]
Payload Byte [6]: swing time [0]
Payload Byte [7]: swing time [1]
Payload Byte [8]: consecutive hits [0]
Payload Byte [9]: force [0]
Payload Byte [10]: battery level [0]
Payload Byte [11]: Accuracy [0]
Payload Byte [12]: Flags [0]
Payload Byte [13]: transmission counter [0]
EOF Byte[14] : 0x27

Where

Timestamp = (timestamp [0] * 65536) + (timestamp [1] * 256) + timestamp [2]

Hitcounter = (hitcounter [0] * 256) + hitcounter [1]

consecutive hits = (consecutive hits[0])

swing time = (swing time [0] * 256) + swing time [1] in milliseconds

Flags:

Bit 0 : 1 → cheating detected, 0 → no cheating detected

Bit 1: 1 → direct hit detected , 0 → indirect hit detected

Bit 2: 1 → swing time is valid, 0 → swing time is not valid

(Since we need at least one free swing to measure the swing time, swing time may be invalid in case there were no free swings during the session)

2.3 Info frame

The „info frame“ is used to transmit informational data such as SW Version number from ball and/or anchor tot he app. The info frame is not send periodically by the ball/anchor, but can be requested by the APP.

SOF Byte[0] : 0x24 „$“

Payload Byte [1]: TBD

Payload Byte [2]: TBD

Payload Byte [3]: TBD

…. TBD ….

Payload Byte [19]: TBD

EOF Byte[20] : 0x27

2.4 Upload frame

The „Upload frame“ can be used to by the APP to send data to the ball/anchor device e.g. the device name.

SOF Byte[0] : 0x7B „{“

Payload Byte [1]: payload ID (0x01 → Device Name following)

Payload Byte [2]:

…

Payload Byte [N]: last byte of payload

EOF Byte[N+1] : 0x27

The receiving device respondes with **ACK** or **NACK**

2.5 Offline bowncing

2.5.1 Request offline session information

The offline session information frame can be used from the app to request the information about the number of stored offline sessions.

SOF Byte[0] : 0x7C „|“

Payload Byte [1]: 0x02

EOF Byte[3] : 0x27

The receiving device respondes with the offline info frame:

SOF Byte[0] : 0x25

Payload Byte [1]: number of stored offline bowncing sessions

EOF Byte[20] : 0x27

2.5.2 Request offline session

SOF Byte[0] : 0x7C „|“

Payload Byte [1]: 0x03

Payload Byte [2]: session id to transmit

EOF Byte[3] : 0x27

The receiving device respondes with the session frame:

SOF Byte[0] : 0x26
Payload Byte [1]: session ID
Payload Byte [2]: session ID
Payload Byte [3]: session ID
Payload Byte [4]: session ID
Payload Byte [5]: hitcounter [0]
Payload Byte [6]: hitcounter [1]
Payload Byte [7]: average accuracy
Payload Byte [8]: average hitpower
Payload Byte [9]: consecutive hits
Payload Byte [10]: min. accuracy
Payload Byte [11]: max. accuracy
Payload Byte [12]: min. hitpower
Payload Byte [13]: max. hitpower
Payload Byte [14]: swing time [0]
Payload Byte [15]: swing time [1]
EOF Byte[16] : 0x27

Where:
Hitcounter = (hitcounter [0] * 256) + hitcounter [1]
consecutive hits = (consecutive hits[0])
swing time = (swing time [0] * 256) + swing time [1] in milliseconds

3 Specific command format

3.1 Request info frame

This command can be used by the APP to resquest the info frame from ball or anchor

SOF Byte[0] : 0x7C „|“

Payload Byte [1]: 0x00

EOF Byte[3] : 0x27

The receiving device respondes with the info frame. No ACK or NACK will be sent!

3.2 Reset to factory settings

SOF Byte[0] : 0x7C „|“

Payload Byte [1]: 0x01

EOF Byte[3] : 0x27

The receiving device respondes with **ACK** or **NACK**