OBERÖSTERREICH

• **Hagenberg** • Linz • Steyr • Wels

Jugglow

API – Application Programming Interface

Version 1.1

October 2015

sapere aude
UNIVERSITY
OF APPLIED SCIENCES
UPPER AUSTRIA

Bildungs LAND
OBERÖSTERREICH

FH OÖ Forschungs & Entwicklungs GmbH

Franz-Fritsch-Strasse 11 / Top 3
4600 Wels
Austria

Research Center Hagenberg
Research Group Embedded Systems

Softwarepark 11
4232 Hagenberg
Austria

Juggglow API

Revision

Rev.	Description
1.0	first release
1.1	pdf bookmarks added BLE UUID of „Celsius Temperature” characteristic was wrong, correct value is 2A1C

Content

1. Ball Control and Ball Control Notification

All Ball Control Commands will be sent via Bluetooth characteristic “Ball Control”, which has a length of 10 bytes. The first byte indicates the command. All commands are defined in the following subsections.

BLE UUID of “Ball Control” characteristic: c75076c0-abbf-11e4-8053-0002a5d5c51b

Table 1: Ball Control Command Template

Byte	Value	Description
1	0x00..0xFF	command identifier
2	0x00..0xFF	data byte 1
3	0x00..0xFF	data byte 2
4	0x00..0xFF	data byte 3
5	0x00..0xFF	data byte 4
6	0x00..0xFF	data byte 5
7	0x00..0xFF	data byte 6
8	0x00..0xFF	data byte 7
9	0x00..0xFF	data byte 8
10	0x00..0xFF	data byte 9

Responses and notifications will be sent via Bluetooth characteristic “Ball Control Notification”, which has a length of 10 bytes.

BLE UUID of “Ball Control Notification” characteristic: f9136034-3b36-4286-8340-570ecd514d35

Table 2: Ball Control Notification Template

Byte	Value	Description
1	0x00..0xFF	command identifier associated to this response
2	0x00..0xFF	data byte 1
3	0x00..0xFF	data byte 2
4	0x00..0xFF	data byte 3
5	0x00..0xFF	data byte 4
6	0x00..0xFF	data byte 5
7	0x00..0xFF	data byte 6
8	0x00..0xFF	data byte 7
9	0x00..0xFF	data byte 8
10	0x00..0xFF	data byte 9

“Ball Control” and “Ball Control Notification” characteristics are part of the Bluetooth custom service “Juggglow” with BLE UUID: 624e957f-cb42-4cd6-bacc-84aeb898f69b.

Table 3: Ball Control Command Overview

Value	Command identifier list
Firmware Commands	
0x00	Idle
0x01	Firmware Shutdown Command
0x02	Firmware Enable Stand-alone Mode Command
0x03	Firmware Check Charging Command / Response
0x04	Firmware Charging Indication Command
0x05	Firmware Set Stop Mode Timing Command
0x06	Firmware Set Standby Mode Timing Command
0x07	Firmware Set Battery Check Intervals Command
0x08	Firmware Set Shut-down Level Command
0x09	Firmware Set Temperature Check Interval Command
0x0A	Firmware Set Critical Temperature Command
0x0B	Firmware Get Configuration
0x0C..0x0F	reserved for future firmware commands
Bluetooth LE Commands	
0x10..0x1F	reserved for future Bluetooth LE commands
Accelerometer Commands	
0x20	Accelerometer Start Recording Values Command
0x21	Accelerometer Stop Recording Values Command
0x22	Accelerometer Get Recording Status Command / Response
0x23..0x2F	reserved for future accelerometer commands
Light Effect Commands	
0x30	Light Effect Off
0x31	Light Effect Stop
0x32	Light Effect Set LED Brightness Command
0x33	Light Effect Run Flash Sequence Command
0x34	Light Effect Set Colors Command
0x35	Light Effect Set Rainbow Command
0x36	Light Effect Set Random Color Command
0x37	Light Effect Set Fading Command
0x38	Light Effect Set Color Timing Command
0x39	Light Effect Set Magic Hands Command
0x3A	Light Effect Set Magic Hands Random Color Command
0x3B	Light Effect Set Color Change on Catch Command
0x3C	Light Effect Set Random Color Change on Catch Command
0x3D.0x5F	reserved for future Light Effect commands
0x60..0xFE	reserved for future use
0xFF	Clear Common Error Flags

1.1. Firmware Commands

This section contains general firmware commands for the stand-alone mode, charging and power settings.

1.1.1. Firmware Shutdown Command

This command sets the Juggglow ball into shutdown state. The Bluetooth communication will be disconnected and the wake-up function via accelerometer will be disabled. It is only possible to reset the ball by connecting a power adapter via USB interface.

Table 4: Firmware Shutdown

Byte	Value	Description
1	0x01	command identifier
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.2. Firmware Enable Stand-Alone Mode Command

The stand-alone mode of the Juggglow balls provides light effects without the use of the Juggglow App. The stand-alone mode can be enabled or disabled with this command. If it is enabled, the light effects of the stand-alone mode can be activated by knocking the ball three times within one second. This mode is enabled by default.

Table 5: Enable Stand-alone Mode

Byte	Value	Description
1	0x02	command identifier
2	0x00..0xFF	0x00: disable stand-alone mode 0x01..0xFF: enable stand-alone mode default: 0x01
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.3. Firmware Check Charging Command

Polls the current charging status. The event “Firmware Check Charging Response” will be generated. It is not necessary to poll the charging status, because the ball will automatically send an event after each change of charging status.

Table 6: Firmware Check Charging Command

Byte	Value	Description
1	0x03	command identifier
2..10	0x00	reserved for future use, value will be ignored

Response: 1.1.4 Firmware Check Charging Response

1.1.4. Firmware Check Charging Response

This notification will be replied if the “Firmware Check Charging Command” was sent or if the charging status has changed. Byte 2 of this response/event indicates if the ball is charging or not.

Table 7: Firmware Charging Indication Response

Byte	Value	Description
1	0x03	response identifier
2	0x00..0x01	0x00: ball is not charging 0x01: ball is charging
3..10	0x00	reserved for future use, value will be ignored

1.1.5. Firmware Charging Indication Command

During charging of the Juggglow balls the LEDs indicate the battery charging level: green: fully charged, yellow: half-charged, orange: nearly discharged, red: discharged. The charging indication can be turned on or off with this command. It is possible to set the interval of the blinking indication. Byte 2 defines the blinking interval in units of 1 sec (1..254 sec).

The charging indication only works if no Bluetooth communication is established.

Table 8: Firmware Charging Indication

Byte	Value	Description
1	0x04	command identifier
2	0x00..0xFF	0x00: LED charging indication permanent on 0xFF: LED charging indication off 0x01..0xFE: LED charging indication with blinking interval in units of 1 sec range: 1..254 sec default: 0x05 (5 sec blinking)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.6. Firmware Set Stop Mode Timing Command

This command sets the lapse of time for entering the stop mode of the microcontroller. A shorter period of time helps to extend the battery runtime of the balls.

Table 9: Firmware Set Stop Mode Timing

Byte	Value	Description
1	0x05	command identifier
2	0x00..0xFF	lapse of time for entering the stop mode in units of 1 sec range: 60..315 sec including 60 sec offset (60 sec + 0..255 sec) default value: 0x00 (60 sec)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.7. Firmware Set Standby Mode Timing Command

This command sets the lapse of time for entering the standby mode of the microcontroller. A shorter period of time helps to extend the battery runtime of the balls.

The timer for the standby mode starts after entering the stop mode.

Table 10: Firmware Set Standby Mode Timing

Byte	Value	Description
1	0x06	command identifier
2	0x00..0xFF	lapse of time for entering the standby mode in units of 1 min range: 1..256 min including 1 min offset (1 min + 0..255 min) default value: 0x1D (30 min)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.8. Firmware Set Battery Check Intervals Command

This command sets the time interval when the Juggglow ball checks the battery level. Byte 2 sets the time interval in running mode and Byte 3 in standby mode.

Table 11: Firmware Set Battery Check Intervals

Byte	Value	Description
1	0x07	command identifier
2	0x00..0xFF	time interval for checking the battery level in running mode in units of 1 sec range: 1..256 sec including 1 sec offset (1 sec + 0..255 sec) default value: 0x01 (2 sec)
3	0x00..0xFF	time interval for checking the battery level in standby mode in units of 6 min range: 6..1440 min including 6 min offset (6 min + 0..239 × 6 min) fix values: 0xEF..0xFF: 1440 min (24 h 00 min) default value: 0x09 (60 min)
4..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.9. Firmware Set Shut-down Level Command

Be careful with this command! You can destroy the battery due to deep discharge if you set the shut-down level to a low value and forget to charge the battery after the ball shuts down. The battery runtime could be increased by setting the shut-down level to a lower value, but it is recommended to keep the shutdown level above the default value to ensure a long battery lifetime.

Table 12: Firmware Set Shut-down Level

Byte	Value	Description
1	0x08	command identifier
2	0x00..0xFF	battery level for shut-down in units of 0.05V range: 0x00..0x0F: 3.1-3.85 V including 3.1 V offset (3.1 V + 0..15 × 0.05 V) fix values: 0x10..0xFF: 3.4V default value: 0x06 (3.4V) *ATTENTION:* Be careful with this setting! You can destroy the battery due to deep discharge.
4..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.10. Firmware Set Temperature Check Interval Command

This command sets the time interval for checking the temperature of the MCU.

Table 13: Firmware Set Temperature Check Interval

Byte	Value	Description
1	0x09	command identifier
2	0x00..0xFF	time interval for checking the temperature of the CPU in units of 1 sec range: 1..256 sec including 1 sec offset (1 sec + 0..255 sec) default value: 0x09 (10 sec)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.11. Firmware Set Critical Temperature Command

This command sets the critical temperature for checking the temperature of the MCU. If the temperature is above the critical value, the MCU will shut down. The Juggglow ball will restart after connecting a power adapter via USB connector. But let the ball cool down at least for half an hour.

Be careful with this command! You can destroy the battery or significantly shorten the battery lifetime due to overheating.

Table 14: Firmware Set Critical Temperature

Byte	Value	Description
1	0x0A	command identifier
2	0x00..0xFF	critical temperature of the CPU in units of 1°C range: 0x00..0x23: 50..85°C including 50°C offset (50°C + 0..35°C) fix values: 0x24..0xFF: 85°C default value: 0x14 (70 °C)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.1.12. Firmware Get Configuration Command

This Command triggers the response of the current firmware configuration of the Juggglow ball. Table 16 lists the configuration response.

Table 15: Firmware Get Configuration

Byte	Value	Description
1	0x0B	command identifier
2..10	0x00	reserved for future use, value will be ignored

Response: 1.1.13 Firmware Get Configuration Response

1.1.13. Firmware Get Configuration Response

This Response will be replied if the “Firmware Get Configuration Command” was sent.

Table 16: Firmware Get Configuration Response

Byte	Value	Description
1	0x0B	command identifier for “Firmware Get Configuration”
2	0x00..0x01	value of Enable Stand-alone Mode
3	0x00..0xFF	value of Charging Indication
4	0x00..0xFF	value of Stop Mode Timing
5	0x00..0xFF	value of Standby Mode Timing
6	0x00..0xFF	value of Battery Check Interval Running
7	0x00..0xEF	value of Battery Check Interval Standby
8	0x00..0x0F	value of Shut-down Level
9	0x00..0xFF	value of Temperature Check Interval
10	0x00..0x23	value of Critical Temperature

1.2. Accelerometer Control Command

1.2.1. Accelerometer Start Recording Values Command

This command starts the recording of the acceleration values. The values will be stored in the external flash memory into the user space. To access the recorded values, you have to read the user space memory with the Flash Control and Flash Data commands on page 27.

Table 17: Accelerometer Start Recording Values

Byte	Value	Description
1	0x20	command identifier
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.2.2. Accelerometer Stop Recording Values Command

This command stops the current recording of the acceleration values.

Table 18: Accelerometer Stop Recording Values

Byte	Value	Description
1	0x21	command identifier
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.2.3. Accelerometer Get Recording Status Command

This command polls the current recording status. The event “Accelerometer Recoding Status Response” will be generated.

Table 19: Accelerometer Get Recording Status Command

Byte	Value	Description
1	0x22	command identifier
2..10	0x00	reserved for future use, value will be ignored

Response: 1.2.4 Accelerometer Recoding Status Response

1.2.4. Accelerometer Recoding Status Response

This response will be generated, when the recording of acceleration values was stopped or the "Accelerometer Get Recording Status" command was set.

Table 20: Accelerometer Recording Status

Byte	Value	Description
1	0x22	command identifier
2	0x00..0x01	0x00: the recording is stopped; 0x01: the recording is running
3..10	0x00	reserved for future use, value will be ignored

1.3. Light Effect Commands

The following commands are used to set LED colors and light effects of the juggling balls.

1.3.1. Light Effect Off Command

This command stops the current running light effect and turns off both LEDs

Table 21: Light Effect Off

Byte	Value	Description
1	0x30	command identifier for "Light Effect Off"
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.2. Light Effect Stop Command

This command stops the current running light effect and the LEDs keep their last color.

Table 22: Light Effect Stop

Byte	Value	Description
1	0x31	command identifier for "Light Effect Stop"
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.3. Light Effect Set LED Brightness Command

This command sets the general brightness of the LEDs for all color effects. The value 0xFF defines full brightness of the LEDs and the value 0x00 turns off the LEDs. Reducing the LED brightness increases the battery runtime of the Juggglow ball.

Table 23: Light Effect Set LED Brightness

Byte	Value	Description
1	0x32	command identifier for "Light Effect Set LED Brightness"
2	0x00..0xFF	LED brightness (range: 0x00..0xFF, default: 0xFF)
3..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.4. Light Effect Run Flash Sequence Command

This command starts the user-defined light sequence from flash memory (user section with 128 kByte).

Byte 2 contains the sequence number, which is defined during download to the flash memory (see section 3 on page 27). If the defined sequence is not stored, no sequence will be started.

Byte 3 contains the start condition of the sequence, which is defined in

Table 25. The start condition can be defined for the beginning and the repeated sequences.

The sequence can be started immediately or with a time offset, which is defined in byte 4.

The sequence could be repeated a maximum of 255 times, which is defined in byte 5.

Table 24: LED Run Flash Sequence

Byte	Value	Description
1	0x33	command identifier for “Light Effect Run Flash Sequence”
2	0x01..0xFF	sequence number
3	0x00..0xFF	sequence start condition (see Table 25)
4	0x00..0xFF	time offset for the start (in units of 1 sec)
5	0x00..0xFF	number of sequence repetitions
6..10	0x00	reserved for future use, value will be ignored

Table 25: Sequence Start Condition

Value	Sequence Start Condition
0x00	beginning: start immediately repetition: start immediately
0x01	beginning: start immediately repetition: start after time offset elapses (set in byte 3)
0x02	beginning: start after time offset elapses (set in byte 3) repetition: start immediately
0x03..0xFF	beginning: start after time offset elapses (set in byte 3) repetition: start after time offset elapses (set in byte 3)

There is no dedicated response to this command.

1.3.5. Light Effect Set Color Command

This command sets the color values (RGB) for the top and bottom LEDs. Set all values to 0xFF to get color white and set all values to 0x00 to turn off the LEDs.

Table 26: Light Effect Set Color

Byte	Value	Description
1	0x34	command identifier for “Light Effect Set Color”
2	0x00..0xFF	value for red color of top LED
3	0x00..0xFF	value for green color of top LED
4	0x00..0xFF	value for blue color of top LED
5	0x00..0xFF	value for red color of bottom LED
6	0x00..0xFF	value for green color of bottom LED
7	0x00..0xFF	value for blue color of bottom LED
8..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.6. Light Effect Set Rainbow Command

This command starts the light effect “Rainbow”. The LED colors smoothly change from red to blue, from blue to green and from green back to red.

Byte 2 defines which LED should be set to the “Rainbow” mode or to the “Inverse Rainbow” mode or should be switched off.

Byte 3 defines the time interval of one single color step, which can be set between 0x01 (1 ms, very fast) and 0xFF (255 ms, very slow). Special value 0x00 defines a time interval less than 1 ms.

Byte 4 defines the color step size. A lower step size results in a smooth color gradient

Table 27: Light Effect Set Rainbow

<table>
<tr><th>Byte</th><th>Value</th><th>Description</th></tr>
<tr><td>1</td><td>0x35</td><td>command identifier for “Light Effect Set Rainbow”</td></tr>
<tr><td>2</td><td>0x00..0xFF</td><td>LED selection value:
<table>
<tr><th>Value</th><th>top LED</th><th>bottom LED</th></tr>
<tr><td>0x00</td><td>Off</td><td>Off</td></tr>
<tr><td>0x01</td><td>Rainbow</td><td>Off</td></tr>
<tr><td>0x02</td><td>Off</td><td>Rainbow</td></tr>
<tr><td>0x03</td><td>Rainbow</td><td>Rainbow</td></tr>
<tr><td>0x04</td><td>Rainbow</td><td>Inverse Rainbow</td></tr>
<tr><td>0x05</td><td>Inverse Rainbow</td><td>Rainbow</td></tr>
<tr><td>0x06</td><td>Inverse Rainbow</td><td>Off</td></tr>
<tr><td>0x07</td><td>Off</td><td>Inverse Rainbow</td></tr>
<tr><td>0x08..0xFF</td><td>Inverse Rainbow</td><td>Inverse Rainbow</td></tr>
</table></td></tr>
<tr><td>3</td><td>0x00..0xFF</td><td>time interval of one color step in units of 1 ms,
range: 1..255 ms,
default: 1 ms,
special value 0x00 defines a time interval &lt; 1 ms</td></tr>
<tr><td>4</td><td>0x00..0xFF</td><td>color step size in units of 1 RGB value
range 1..255 with 1 offset
default: 0x01 = 2 steps,
special value of 0xFF = 0 steps, which stops the effect)</td></tr>
<tr><td>5..10</td><td>0x00</td><td>reserved for future use, value will be ignored</td></tr>
</table>

There is no dedicated response to this command.

1.3.7. Light Effect Set Random Color Command

This command starts a light effect with possibilities for individual timing adjustments. Unlike the "Set Timing" command the color is not fixed, it randomly changes.

Byte 2 defines which LED should be set to the "Random Color" mode or to the "Inverse Random Color" mode or should be switched off.

Byte 3 defines the time interval of the Stroboscope effect, which can be set between 0x00 (100 ms, very fast) and 0xFF (25.6 sec, very slow).

Byte 4 defines the duty cycle of the ON phase of the effect, which can be set between 0x00 (0% on, 100% off) and 0xFF (100% on, 0% off).

Table 28: Light Effect Set Random Color

<table>
<tr><th>Byte</th><th>Value</th><th>Description</th></tr>
<tr><td>1</td><td>0x36</td><td>command identifier for "Light Effect Set Random Color"</td></tr>
<tr><td>2</td><td>0x00..0xFF</td><td>LED selection value:
<table>
<tr><th>Value</th><th>top LED</th><th>bottom LED</th></tr>
<tr><td>0x00</td><td>Off</td><td>Off</td></tr>
<tr><td>0x01</td><td>Random Color</td><td>Off</td></tr>
<tr><td>0x02</td><td>Off</td><td>Random Color</td></tr>
<tr><td>0x03</td><td>Random Color</td><td>Random Color</td></tr>
<tr><td>0x04</td><td>Random Color</td><td>Inverse Random Color</td></tr>
<tr><td>0x05</td><td>Inverse Random Color</td><td>Random Color</td></tr>
<tr><td>0x06</td><td>Inverse Random Color</td><td>Off</td></tr>
<tr><td>0x07</td><td>Off</td><td>Inverse Random Color</td></tr>
<tr><td>0x08..0xFF</td><td>Inverse Random Color</td><td>Inverse Random Color</td></tr>
</table></td></tr>
<tr><td>3</td><td>0x00..0xFF</td><td>time interval of effect in units of 100 ms
range: 100 ms..25.6 sec with 100 ms offset
default: 100 ms = 0x00</td></tr>
<tr><td>4</td><td>0x00..0xFF</td><td>duty cycle of ON phase for effect
0x00 = 0%
0xFF = 100%
default: 0xFF</td></tr>
<tr><td>5..10</td><td>0x00</td><td>reserved for future use, value will be ignored</td></tr>
</table>

There is no dedicated response to this command.

1.3.8. Light Effect Set Fading Command

This light effect changes between two defined colors using a fading effect. It is possible to set this effect for both LEDs or only for the top LED or bottom LED with the LED selection value in byte 2.

Byte 2 defines which LED should be set to the “Fading” mode or to the “Inverse Fading” mode or should be switched off.

Byte 3 defines the time interval of one single color step, which can be set between 0x01 (1 ms, very fast) and 0xFF (255 ms, very slow). Special value 0x00 defines a time interval less than 1 ms.

Byte 4 defines the color step size.

Byte 5 to 7 define the RGB color parameter of first color

Byte 8 to 10 define the RGB color parameter of second color

Table 29: Light Effect Set Fading

<table>
<tr><th>Byte</th><th>Value</th><th>Description</th></tr>
<tr><td>1</td><td>0x37</td><td>command identifier for “Light Effect Set Timing”</td></tr>
<tr><td>2</td><td>0x00..0xFF</td><td>LED selection value:
<table>
<tr><th>Value</th><th>top LED</th><th>bottom LED</th></tr>
<tr><td>0x00</td><td>off</td><td>off</td></tr>
<tr><td>0x01</td><td>Fading</td><td>off</td></tr>
<tr><td>0x02</td><td>off</td><td>Fading</td></tr>
<tr><td>0x03</td><td>Fading</td><td>Fading</td></tr>
<tr><td>0x04</td><td>Fading</td><td>inverse Fading</td></tr>
<tr><td>0x05</td><td>Inverse Fading</td><td>Rainbow</td></tr>
<tr><td>0x06</td><td>inverse Fading</td><td>off</td></tr>
<tr><td>0x07</td><td>off</td><td>inverse Fading</td></tr>
<tr><td>0x08..0xFF</td><td>inverse Fading</td><td>inverse Fading</td></tr>
</table></td></tr>
<tr><td>3</td><td>0x00..0xFF</td><td>time interval of a single color step in units of 1 ms
range: 1..255 ms
default: 1 ms
special value 0x00 defines a time interval &lt; 1 ms</td></tr>
<tr><td>4</td><td>0x00..0xFF</td><td>color step size in units of 1 RGB value
range 1..255 with 1 offset
default: 0x01 = 2 steps,
special value of 0xFF = 0 steps, which changes directly between the defined colors</td></tr>
<tr><td>5</td><td>0x00..0xFF</td><td>first color parameter value for red color of both LEDs</td></tr>
<tr><td>6</td><td>0x00..0xFF</td><td>first color parameter value for green color of both LEDs</td></tr>
<tr><td>7</td><td>0x00..0xFF</td><td>first color parameter value for blue color of both LEDs</td></tr>
<tr><td>8</td><td>0x00..0xFF</td><td>second color parameter value for red color of both LEDs</td></tr>
<tr><td>9</td><td>0x00..0xFF</td><td>second color parameter value for green color of both LEDs</td></tr>
<tr><td>10</td><td>0x00..0xFF</td><td>second color parameter value for blue color of both LEDs</td></tr>
</table>

There is no dedicated response to this command.

1.3.9. Light Effect Set Color Timing Command

This command starts a light effect with possibilities for individual timing adjustments. It is possible to generate *stroboscope*, *blinking*, *flashing* and many other effects.

Byte 2 defines the time interval of this effect, which can be set between 0x00 (10 ms, very fast) and 0xFF (2.56 sec, very slow).

Byte 3 defines the duty cycle of the ON phase for the effect, which can be set between 0x00 (0% on, 100% off) and 0xFF (100% on, 0% off).

Bytes 4 to 9 define the RGB values for the color setting of top and bottom LED. The LEDs can be turned off by setting the RGB values to 0x00.

For example: A white stroboscope effect can be set by sending

0x38	0x0A	0x40	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

Table 30: Light Effect Set Color Timing

Byte	Value	Description
1	0x38	command identifier for “Light Effect Set Timing”
2	0x00..0xFF	time interval of effect in units of 10 ms range: 10 ms..2.56 sec with 10 ms offset default: 10 ms = 0x00
3	0x00..0xFF	duty cycle of ON phase for effect 0x00 = 0% 0xFF = 100% default: 0x20
4	0x00..0xFF	value for red color of top LED
5	0x00..0xFF	value for green color of top LED
6	0x00..0xFF	value for blue color of top LED
7	0x00..0xFF	value for red color of bottom LED
8	0x00..0xFF	value for green color of bottom LED
9	0x00..0xFF	value for blue color of bottom LED
10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.10. Light Effect Set Magic Hands Command

This command starts the “Magic Hands” effect. One color can be set for the catch phase (phase during catch and throw) and another color can be set for the flight phase.

Bytes 2 to 4 define the RGB values for the color setting of both LEDs during catch phase.

Bytes 5 to 7 define the RGB values for the color setting of both LEDs during flight phase.

Table 31: Light Effect Set Magic Hands

Byte	Value	Description
1	0x39	command identifier for “Light Effect Set Magic Hand”
2	0x00..0xFF	value for red color of both LEDs during catch phase
3	0x00..0xFF	value for green color of both LEDs during catch phase
4	0x00..0xFF	value for blue color of both LEDs during catch phase
5	0x00..0xFF	value for red color of both LEDs during flight phase
6	0x00..0xFF	value for green color of both LEDs during flight phase
7	0x00..0xFF	value for blue color of both LEDs during flight phase
8..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.11. Light Effect Set Magic Hands Random Color Command

This command starts the “Magic Hands Random Color” effect. Each ball will turn off the light during the phase between the catch and the throw. That means the lights are turned off as long as the balls are held in the hands during juggling. The color of the ball changes randomly after each catch.

Table 32: Light Effect Set Magic Hands Random Color

Byte	Value	Description
1	0x3A	command identifier for “Light Effect Set Magic Hand Random Color”
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.12. Light Effect Set Color Change on Catch Command

This command starts the “Color Change on Catch” effect. The balls switch between two defined colors after each catch.

Bytes 2 to 4 define the RGB values for the first color of both LEDs.

Bytes 5 to 7 define the RGB values for the second color of both LEDs.

Table 33: Light Effect Set Color Change on Catch

Byte	Value	Description
1	0x3B	command identifier for “Light Effect Set Color Change on Catch”
2	0x00..0xFF	first value for red color of both LEDs
3	0x00..0xFF	first value for green color of both LEDs
4	0x00..0xFF	first value for blue color of both LEDs
5	0x00..0xFF	second value for red color of both LEDs
6	0x00..0xFF	second value for green color of both LEDs
7	0x00..0xFF	second value for blue color of both LEDs
8..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.3.13. Light Effect Set Random Color Change on Catch Command

This command starts the “Color Change on Catch” effect. Each ball will randomly change its color after a catch.

Table 34: Light Effect Set Random Color Change on Catch

Byte	Value	Description
1	0x3C	command identifier for “Light Effect Set Random Color Change on Catch”
2..10	0x00	reserved for future use, value will be ignored

There is no dedicated response to this command.

1.4. Common Errors

If a common error occurs, the microcontroller of the ball sends a “Common Error Response” via Bluetooth notification. The error flags in the microcontroller and the error message stored in the Bluetooth module will be cleared with the command “Clear Common Error Flags”.

1.4.1. Common Error Response

The common error responses are sent via Bluetooth characteristic “Ball Control Notification”.

Table 35: Common Error Response

Byte	Value	Description
1	0xFF	response identifier for “Common Errors”
2	0x00..0xFF	error code
3..10	0x00	don’t care

Table 36: Error Codes

Value	Error Code
0x00	no error / acknowledge of error reset
0x01	command is not defined
0x02..0xFF	reserved for future use

1.4.2. Clear Common Error Flags

This command resets all common error flags in the microcontroller of the ball. It will be sent via Bluetooth characteristic “Ball Control”.

Table 37: Clear Common Error Flags

Byte	Value	Description
1	0xFF	command identifier for “Clear Common Error Flags”
2..10	0x00	don’t care

There is no dedicated response to this command

2. Ball Event Notification

This notification will be sent via Bluetooth characteristic “Ball Event Notification”, which has a length of 10 bytes. The MCU determines the ball events regarding to the accelerometer values. Ball events are shown in Table 39. This notification includes the event counter, which will be increased after each new event change, and the catch counter.

Byte 2 and 3 are reserved for the elapsed time between current and last event. This time duration is indicated in milliseconds.

Byte 4 and 5 are reserved for the event counter.

Byte 6 and 7 are reserved for the catch counter.

BLE UUID of “Ball Event Notification” characteristic: d6d4ef6d-1cef-4aa2-9657-e373d6f697fb

“Ball Event Notification” characteristic is part of the Bluetooth custom service “Juggglow” with BLE UUID: 624e957f-cb42-4cd6-bacc-84aeb898f69b

Table 38: Ball Event Notification

Byte	Value	Description
1	0x00..0x03	ball event type (see Table 39)
2	0x0000.. 0xFFFF	elapsed time between current and last event (in units of 1 ms)
3		
4	0x0000.. 0xFFFF	event counter
5		
6	0x0000.. 0xFFFF	catch counter
7		
8	don’t care	reserved for future use
9	don’t care	reserved for future use
10	don’t care	reserved for future use

Table 39: Ball Event Types

Value	Ball Event Types
0x00	idle
0x01	catch
0x02	throw
0x03	drop
0x04..0xFF	reserved for future use

3. Flash Control and Flash Data

Two Bluetooth characteristics ("Flash Control" and "Flash Data") are defined for the read and write access to the user defined section of the external flash. The user-defined section has a size of 128 kByte.

BLE UUID of "Flash Control" characteristic: 51892c94-c9c7-4b64-9701-d32055c990cd

BLE UUID of "Flash Data" characteristic: 9caf4e31-b2dc-4eb5-a8a0-47daac3e0faa

"Flash Control" and "Flash Data" characteristics are part of the Bluetooth custom service "Juggglow" with BLE UUID: 624e957f-cb42-4cd6-bacc-84aeb898f69b

The software of the ball can write the acceleration values to the external flash (see section 1.2 on page 14). You can read out this information with the command set in this section.

It is also possible to store user-defined color sequences in the external flash. There a two different types of sequences: color and fading sequences.

Table 40: Color Sequence

Byte	Value	Description
1	0xC0	Type: color sequence
2	0x0000.. 0xFFFF	time interval of sequence in units of 1 ms range: 1..65.536 ms, special value 0x0000 defines a time interval < 1 ms
3		
4	0x00..0xFF	value for red color of top LED
5	0x00..0xFF	value for green color of top LED
6	0x00..0xFF	value for blue color of top LED
7	0x00..0xFF	value for red color of bottom LED
8	0x00..0xFF	value for green color of bottom LED
9	0x00..0xFF	value for blue color of bottom LED

Table 41: Fading Sequence

Byte	Value	Description
1	0xF0	Type: fading sequence
2	0x00..0xFF	time interval of color step in units of 1 ms range: 1..255 ms, special value 0x00 defines a time interval < 1 ms
3	0x00..0xFF	color step size in units of 1 RGB value range 1..255 with 1 offset, special value 0xFF = 0 steps, which stops the effect
4	0x00..0xFF	value for red color of top LED
5	0x00..0xFF	value for green color of top LED
6	0x00..0xFF	value for blue color of top LED
7	0x00..0xFF	value for red color of bottom LED
8	0x00..0xFF	value for green color of bottom LED
9	0x00..0xFF	value for blue color of bottom LED

The data is stored in a special binary format in the flash memory, which is described in Table 42.

Table 42: User-space binary header

<table>
<tr><th>Byte</th><th>Value</th><th>Description</th></tr>
<tr><td>1</td><td rowspan="2">0x0000-
0xFFFF</td><td rowspan="2"><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>0xA55A</td><td>Acceleration data</td></tr><tr><td>0x5AA5</td><td>User-defined sequence</td></tr><tr><td>0xFFFF</td><td>Empty</td></tr></table></td></tr>
<tr><td>2</td></tr>
<tr><td>3</td><td rowspan="4">0x00000000-
0xFFFFFFFF</td><td rowspan="4">The user data length without header</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7 –
length+7</td><td>0xXX</td><td>User data (acceleration data or user-defined sequences)</td></tr>
</table>

3.1. Flash Write User-defined Effect Sequence

It is possible to define multiple user-defined effect sequences. When you want to download a user-defined sequence, the binary file must have the following format to work correctly. In the user-space binary header #0 the length value will be ignored. This value will be set in the download sequence of the ball and is only used for internal usage. All other length values must be set to the corresponding sequence sizes (9x number of sequence values). Furthermore a sequence can contain multiple sequence values. A mixed usage is also possible, e.g.: fading and color sequence values in one sequence.

User-space binary header #0	User-space binary header #1	Sequence #1	User-space binary header #2	Sequence #2	…

3.2. Flash Read of Acceleration Values

The format of one acceleration value is described in *Table 43*. Start and stop of recording such sequences is described in *section 1.2*. The upload sequence is described in *section 3.5*.

Table 43: Format of one Acceleration Value

<table>
<tr><th>Byte</th><th>Value</th><th>Description</th></tr>
<tr><td>1</td><td rowspan="4">0x00000000-
0xFFFFFFFF</td><td rowspan="4">Defines the current time of the recorded value</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td><td rowspan="2">0x0000-
0xFFFF</td><td rowspan="2">Calculated acceleration value in g/100</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td rowspan="2">0x0000-
0xFFFF</td><td rowspan="2">Derivation of the acceleration values (-32768 to +32767)</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td><td rowspan="2">0x0000-
0xFFFF</td><td rowspan="2">Time between the current and the old acceleration value in milliseconds</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td><td>0x00-0xFF</td><td>The slope. Possible values are:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>20</td><td>Fast edge fall</td></tr>
<tr><td>70</td><td>Slow edge fall</td></tr>
<tr><td>100</td><td>0G</td></tr>
<tr><td>110</td><td>1G</td></tr>
<tr><td>130</td><td>Slow edge rise</td></tr>
<tr><td>180</td><td>Fast edge rise</td></tr>
<tr><td>255</td><td>Unknown</td></tr>
</table></td></tr>
<tr><td>12</td><td>0x00-0xFF</td><td>Indicates if a new event was detected. 1 .. true, 0 ... false</td></tr>
<tr><td>13</td><td>0x00-0xFF</td><td>The ball event state. Possible values are:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>0</td><td>Free fall</td></tr>
<tr><td>1</td><td>Lay on ground</td></tr>
<tr><td>20</td><td>Knock</td></tr>
<tr><td>100</td><td>Caught</td></tr>
<tr><td>200</td><td>Shock</td></tr>
<tr><td>255</td><td>Unknown</td></tr>
</table></td></tr>
<tr><td>14</td><td>0x00-0x0A</td><td>The ball event type. Possible values are:
<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>0</td><td>idle</td></tr>
<tr><td>1</td><td>catch</td></tr>
<tr><td>2</td><td>throw</td></tr>
<tr><td>3</td><td>drop</td></tr>
<tr><td>4</td><td>shake[1]</td></tr>
<tr><td>5</td><td>light shake[1]</td></tr>
<tr><td>6</td><td>vigorous shake[1]</td></tr>
<tr><td>7</td><td>knocking 2x[1]</td></tr>
<tr><td>8</td><td>knocking 3x</td></tr>
<tr><td>9</td><td>knocking 4x[1]</td></tr>
<tr><td>10</td><td>knocking 5x[1]</td></tr>
</table>
[1] not implemented yet</td></tr>
</table>

3.3. User-defined flash control attribute

Table 44 shows the flash control characteristics. The user application has to send all 10 bytes to store or read the data correctly from or to the flash. In *section 3.4* and *3.5* are the sequences how to use this flash control attribute.

Table 45 displays the possible command identifiers from the first byte in the flash control attribute.

Table 44: User-defined flash control

Byte	Value	Description
1	0x00..0xFF	command identifier / acknowledge / error code (see Table 45)
2	0x00..0xFF	Length in the data characteristics
3	0x00..0xFF	Reserved
4	0x00..0xFF	Reserved
5	0x00000000 - 0xFFFFFFFF	Current file offset
6		
7		
8		
9	0x00..0xFF	Reserved
10	0x00..0xFF	Reserved

Table 45: Command identifier list

Value	Command identifier list
0x00	Succeeded
0x01	Write current data from flash data attribute to the flash with the offset and length in the flash control attribute.
0x02	Writes the internal length to an internal control section. This command must be send at the end of a write procedure.
0x03	Reads from the flash with the offset and length from the flash control attribute and writes the read data to the flash data attribute
0x04..0xFE	reserved for future firmware commands
0xFF	An error occurred during the process

3.4. Download sequence

Figure 1 displays the process how to download a user-defined file. In the first step the data must be set into the user-defined data attribute. Then the current offset and length of the data into the control attribute. After this operation the data will be programmed to the flash memory. So the host application has to wait until a response from the ball was received. Then the host application repeats the write operations until the end of the file is reached. After all data was written, the host application has to write the "write internal length" command and wait for the response from the ball. When the response was received the download sequence is finished.

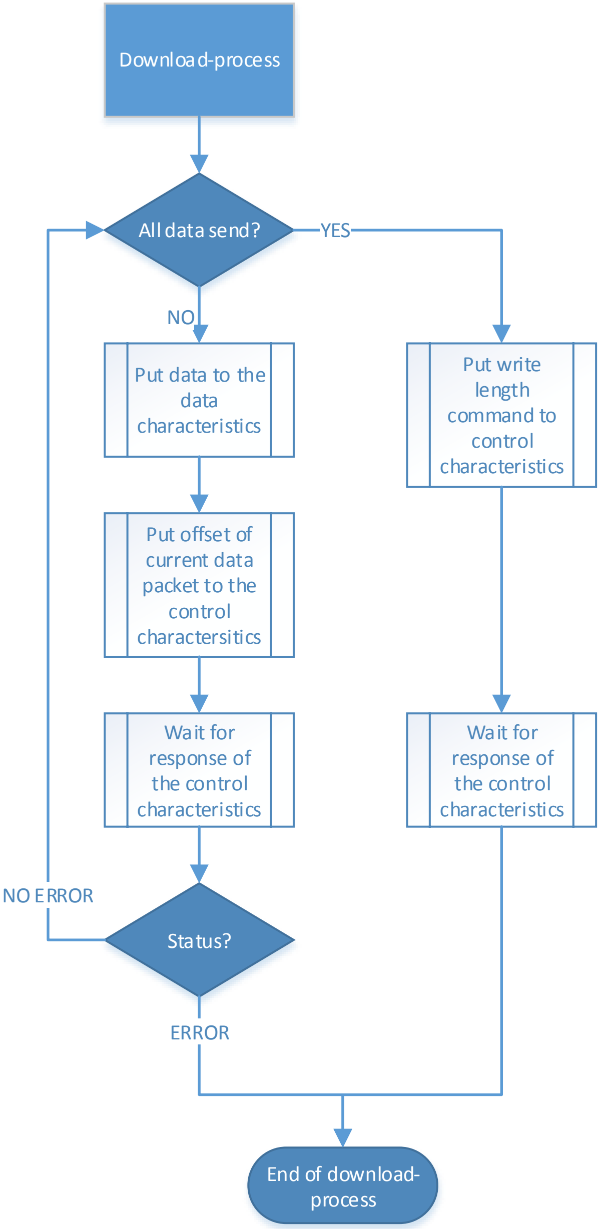
Download-process
All data send?
YES
NO
Put data to the data characteristics
Put offset of current data packet to the control charactersitics
Wait for response of the control characteristics
Status?
NO ERROR
ERROR
Put write length command to control characteristics
Wait for response of the control characteristics
End of download-process

Figure 1: Displays the process how to download a user-defined file

3.5. Upload sequence

Figure 2 displays the upload process for a user-defined file. At the beginning the header is read, where the type of data and length of the block is defined. Table 42 shows the header structure. The header is located on position (offset) 0. If the header type and length are valid, all data will be uploaded in data frames of a maximum size of 20 bytes. To read one data frame the flash control attribute must set the offset and length (maximum of 20) and the "execute read" command. The read operation will be executed and the data will be written into the flash data attribute. When the operation is finished the flash control attribute will be set, which includes the operation result. When a change on this attribute is received the flash operation is finished and the next flash read operation could be started.

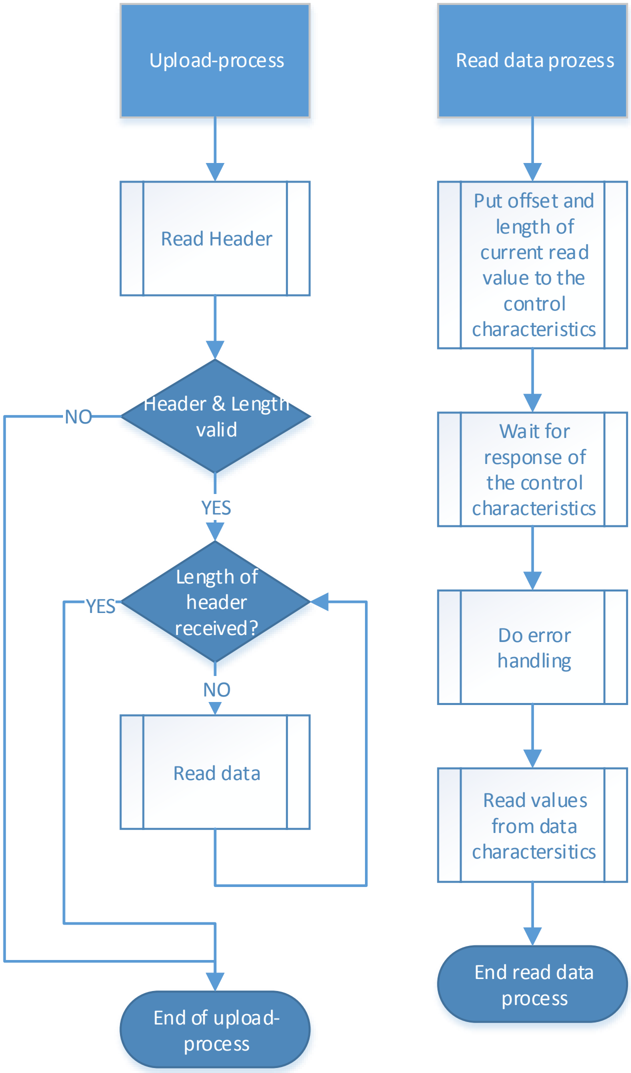
Upload-process
Read Header
Header & Length valid
NO
YES
Length of header received?
YES
NO
Read data
End of upload-process
Read data prozess
Put offset and length of current read value to the control characteristics
Wait for response of the control characteristics
Do error handling
Read values from data charactersitics
End read data process

Figure 2: Displays the process how to upload a user-defined file

4. Battery Service

The “Battery Service” exposes the state of a battery within a device. It contains the “Battery Level” characteristic.

BLE UUID of “Battery Service”: 180F

4.1. Battery Level

The “Battery Level” characteristic will send a notification if the battery level of the Juggglow ball changes its value. The battery charge condition is stored in one byte and represented in percentage with intervals of 10%. The values are displayed in hexadecimal format:

0x00 = 0%, 0x0A = 10%, 0x14 = 20%, 0x1E = 30%, 0x28 = 40%, 0x32 = 50%,
0x3C = 60%, 0x46 = 70%, 0x50 = 80%, 0x5A = 90%, 0x64 = 100%.

A value of 100% represents fully charged while 0% represents completely discharged.

BLE UUID of “Battery Level” characteristic: 2A19

5. Temperature Measurement Service

The “Temperature Measurement Service” exposes the temperature of the microcontroller in the ball. It contains the “Celsius Temperature” characteristic.

BLE UUID of “Temperature Measurement Service”: 1809

5.1. Celsius Temperature

The “Celsius Temperature” characteristic will send a notification if the temperature of the microcontroller changes its value. The temperature is stored in one byte and is represented in degree Celsius (signed 8-bit integer in hexadecimal format: e.g. 0x00 = 0°C, 0x32 = 50°C, 0xFF = -1°C, 0xF6 = -10°C).

BLE UUID of “Celsius Temperature” characteristic: 2A1C

6. Device Information

The device information contains manufacture name, model number, serial number, hardware revision, firmware revision and software revision. Each information is available with its own characteristic.

UUIC of “Device Information” service: 180A

6.1. Manufacture Name String

This characteristic represents the name of the manufacturer of the device: “FH Hagenberg”

BLE UUID of “Manufacture Name String” characteristic: 2A29

6.2. Model Number String

This characteristic represents the model number of the device: “Juggglow”

BLE UUID of “Model Number String” characteristic: 2A24

6.3. Serial Number String

This characteristic represents the serial number for a particular instance of the device.

BLE UUID of “Serial Number String” characteristic: 2A25

6.4. Hardware Revision String

This characteristic represents the hardware revision for the hardware within the device.

BLE UUID of “Hardware Revision String” characteristic: 2A27

6.5. Firmware Revision String

This characteristic represents the firmware revision for the Bluetooth firmware within the device.

BLE UUID of “Firmware Revision String” characteristic: 2A26

6.6. Software Revision String

This characteristic represents the software revision for the microcontroller software within the device.

BLE UUID of “Software Revision String” characteristic: 2A28