Sharkoon Skiller MECH SGK3 USB control protocol

Kacper Donat <kacper@kadet.net>

1 Basic info

Property	**Value**
VendorID	0c45
ProductID	8513
Control Interface Index	1
Endianness	Big Endian

2 Protocol

Commands are submited to keybord via USB control transfer, state updates are send by keyboard via USB interrupts. After every command, keyboard should send state update.

Every packet has length of 64 bytes, and consist of 8 bytes header followed by payload, which can be up to 56 bytes long. It's not possible to split packet into two, instead we should use addressing feature described later.

0	1	2	3	4	5	6	7
04h	checksum		cmd	len	addr		00h
payload up to 56 bytes							

2.1 Header

Every packet starts with magic byte `04h`, probably used to determine control packets. Every value is written in Big Endian manner, ie. LOW byte commes first. Header is ended with one null byte used for padding.

2.1.1 checksum

Magic value is followed by 2-byte long checksum. Checksum calculation method is really trivial - it's just a sum of all bytes after checksum.

$$\texttt{checksum} = \sum_{i=3}^{64} \texttt{Byte}_i \tag{1}$$

2.1.2 command

Command is 1-byte long identifier, which determines payload kind and action taken by keyboard. Commands can be used to update keyboard state, or to read state.

Command	Value	Description
SK_CMD_DISABLE_LED	01h	Disables key backlight, takes some time.
SK_CMD_ENABLE_LED	02h	Enables key backlight, takes some time.
SK_CMD_READ_???	03h	Reads some state.
SK_CMD_READ_???	05h	Reads some state.
SK_CMD_SET_PROP	06h	Changes keyboard mode or mode settings.
SK_CMD_MAP_KEYS	08h	Writes key mapping to controller.
SK_CMD_WRITE_MACRO	0Ah	Saves macros in controller memory.
SK_CMD_SET_COLOR	11h	Changes LED RGB values in key given keys, or key range.

All commands will be described more precisely in later sections.

2.1.3 len

Length of payload, up to 56.

2.1.4 addr

Address of changed value/register, command specific, if not needed should be set to `0000h`.

3 commands

3.1 `SK_CMD_DISABLE_LED` (`01h`)

This command disables all LEDs[1] on keyboard. It's useful when we need to change few values and don't want to constantly blink diodes.

3.2 `SK_CMD_ENABLE_LED` (`02h`)

This command re-enables all LEDs on keyboard. It's useful when we need to change few values and don't want to constantly blink diodes. Both operations are quite long to execute, they can took about 1s.

3.3 `SK_CMD_SET_PROP` (`06h`)

This command is used to change color mode and other settings like brighteness, pulsation direction, polling rate etc.

3.3.1 Addressing

Changed property is referenced via `addr`.

Property	Address	Description
SK_PROP_MODE	00h	Changes backlight program (pulsing, permamant, custom, etc.)
SK_PROP_BRIGHTNESS	01h	Sets brihgtness level
SK_PROP_SPEED	02h	Sets animation speed
SK_PROP_DIRECTION	03h	Sets animation direction
SK_PROP_RGB	04h	LGBT promotion
SK_PROP_COLOR	05h	Base color
SK_PROP_POLLING	0Fh	Polling rate

[1] And maybe other things, hard to test.

3.3.2 Payload

SK_PROP_MODE (00h) Payload is 1-byte long mode identifier, possible modes are listed in table below.

Identifier	Value	Name
SK_MODE_WAVE	01h	Wave
SK_MODE_DRIFT	02h	Drift
SK_MODE_SWIRL	03h	Swirl
SK_MODE_CHANGE	04h	color change
SK_MODE_PULSATING	05h	pulsating
SK_MODE_PERMAMENT	06h	permament
SK_MODE_EXPLOSION	07h	Explosion
SK_MODE_TRIGGER	08h	Trigger
SK_MODE_BURST	09h	Gamma Ray Burst
SK_MODE_CHAOS	0Ah	Chaos
SK_MODE_COSMIC	0Bh	Cosmic
SK_MODE_GRADIENT	0Ch	Gradient
SK_MODE_TIDE	0Dh	Tide
SK_MODE_LIGHTING	0Eh	Ball Lighting
SK_MODE_MATRIX	0Fh	Matrix
SK_MODE_RICOCHET	10h	Ricochet
SK_MODE_RIPPLE	12h	Ripple
SK_MODE_CUSTOM	14h	Custom

SK_PROP_BRIGHTNESS (01h) Payload is 1-byte from range 00h-05h, where 00h means no light at all and 05h is the brightest value.

SK_PROP_SPEED (02h) Payload is 1-byte from range 00h-05h, where 00h is the slowest setting and 05h is the fastest one.

SK_PROP_DIRECTION (03h) Payload 1-byte value desciring direction, 00 for left to right/top to bottom and ffh for reverse.

SK_PROP_RGB (04h) Payload is 1-byte long flag, which defines if we should see rainbow (01h) or not (00h)

SK_PROP_COLOR (05h) Payload is 3-byte RGB value describing color used by mode, detailed structure is exactly the same as in 3.4.2.

SK_PROP_POLLING (0Fh) Payload is 1-byte long value identifing used polling rate.

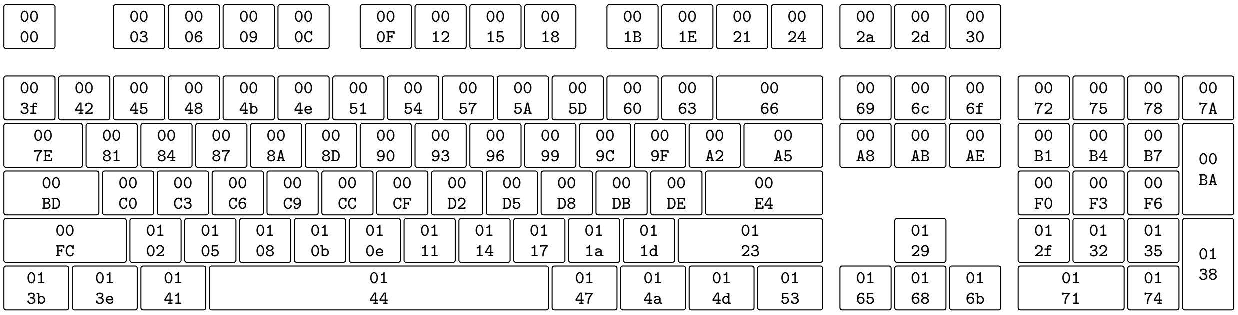
Identifier	Value	Polling rate
SK_POLL_125HZ	00h	125 Hz
SK_POLL_250HZ	01h	250 Hz
SK_POLL_500HZ	02h	500 Hz
SK_POLL_1000HZ	03h	1000 Hz

3.4 SK_CMD_SET_COLOR (11h)

This command can be used to change color of 1 or few LEDs at once. This operation takes less than 100ms, and refreshes all diodes (in simpler words - they blink). This command only makes sens weh in SK_MODE_CUSTOM.

3.4.1 Addressing

The addr header value identifies starting address of specific LED in keyboard. Rule of thumb is that keys are addressed from left to right, top to bottom staring with Esc addressed 0000h. Key address map can be seen below.



3.4.2 Payload

Payload of this command consist of consecutive RGB values, 1-byte per channel, we could describe this struct like that:

```
typedef struct {
    struct {
        uint8_t r;
        uint8_t g;
        uint8_t b;
    } color[SK_MAX_PAYLOAD / 3];
} skillerctl_payload_color_t;
```

3.4.3 Examples

```
0000 04 13 01 11 03 00 00 01 ff 00 00 00 00 00 00 00
0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 1: setting color of Esc to red (ff0000h)

```
0000 04 13 01 11 09 00 00 01 ff ff ff ff ff ff ff ff
0010 ff 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 2: setting color of Esc, F1, F2 to white (ffffffh)