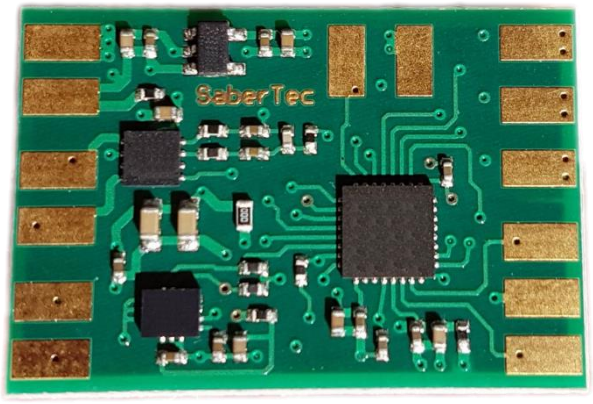
SABERTEC
GERMAN ENGINEERING POWER



Golden Harvest

Manual

SaberTec
Owner: Frederic Folz
Brentanoweg 3
66802 Überherrn
GERMANY

Contact: info@sabertec.net
Website: https://sabertec.net

11/21/2018

Thank you

Thank you for sharing your passion with us! It's only your support that enables us to further develop our technology and continue optimizing it. Without you, we couldn't realize our visions! Back in 2016, we founded SaberTec with the goal to develop a soundboard that can provide the most impressive and realistic saber experience imaginable – two years of hard work later the Golden Harvest was born. Our company's fundamental philosophy is to closely include you, the community, into everything we do – may it be the development of our saber heart, the decision of its name or the further improvement of it. In this sense, we want to invite you to participate in this great project. If you have any ideas or wishes, we would be honored if you would shared them with us! Just write a mail to info@sabertec.net. We are excited to hear from you!

Secret Message from Planet Earth

Will be revealed in future. A mystery awaits you!

Safety notice

As our Golden Harvest board uses electronic parts, it is sensitive to electrostatic discharge. Please handle it with care and make sure to install it correctly. Furthermore, we recommend to use only protected Li-Ion batteries. We are not responsible for any damage that arises from a faulty install or use of the board.

Legal information

Content

Technical specifications and features

The Golden Harvest has the following technical specifications and provides all of the following features:

- tiny dimensions of 31 mm x 21 mm x 4 mm / 1.22 " x 0.82 " x 0.157 "
- powered by 3.7 V (single 18650 Li-ion cell for example)
- supports up to 4 LED channels with up to 4 A each:
  - can drive even QuadCrees
  - capable of delivering up to 16 A in sum (if you need such a high current, please make sure that your battery can provide that and your Golden Harvest is cooled enough)
  - can drive up to 200 accent LEDs (with about 20 mA each) wired in parallel per LED channel
- supports speakers with 4 Ohm and 8 Ohm and with up to 3 W power
- full functionality accessible even if used with only one momentary switch
- can also be used with one momentary and one latching switch as well as with two momentary switches
- Hibernation Mode: allows power saving Deep Sleep for more than 8 months
- microSD card slot
- highly sophisticated motion engine:
  - customizable sensitivity
  - capable of tracing every nuance of your saber's motion
  - detects swings
  - detects clashs
  - detects stabs
  - detects spins
- revolutionary, proprietary sound engine
- 12 bit audio sample resolution
- full-blown sound effect engine:
  - up to 99 different swing sounds
  - up to 99 different clash sounds
  - up to 99 different stab sounds
  - up to 99 different spin sounds
  - up to 99 different lockup sounds
  - up to 99 different blaster block sounds
  - up to 99 different force push sounds
  - up to 99 different boot sounds
  - up to 99 different power on sounds
  - up to 99 different power off sounds
  - hum sound
  - menu sounds
  - up to 99 different sound fonts that contain all these sound effects
- overwhelming light effect engine:
  - full customizability of the effects
  - each LED channel can have completely independent light effects
  - capability of defining unique light effects even for accent LEDs
  - customizability of crystal chamber effects for example
  - pulse effects:
    - conventional pulse

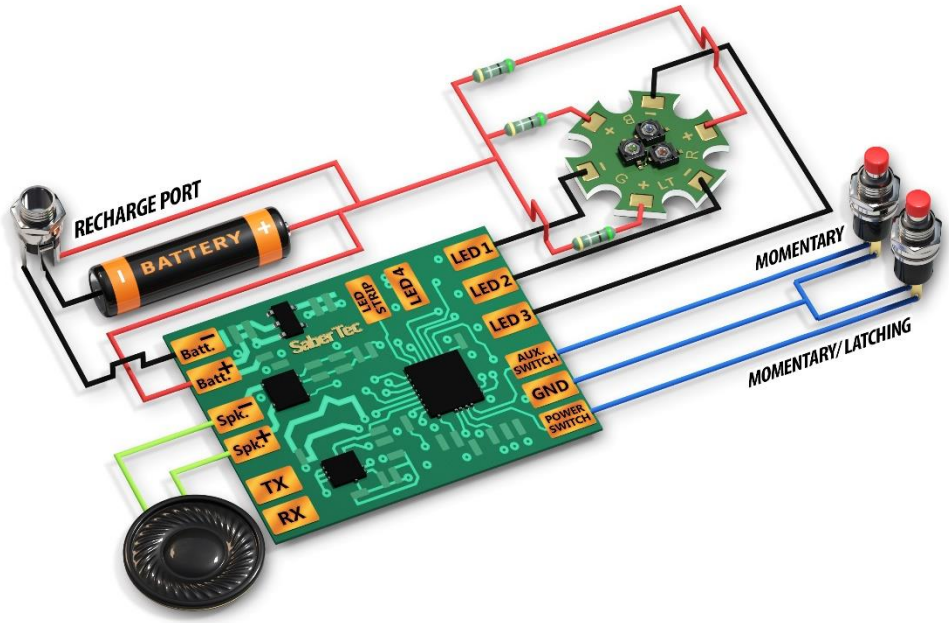
        - speed transition pulse, a.k.a. *"HyperPulse"*
        - intensity transition pulse, a.k.a. *"ÜberPulse"*
        - color transition effects
        - 6 different pulse shapes: sine, impulse, inverted impulse, rising sawtooth, falling sawtooth, rectangular
    - flicker effects:
        - conventional flicker
        - intensity transition, a.k.a. *"PhaseFlicker"*
        - 3 different flicker types: subtractive, additive and mixed
        - flicker synchronization
        - flicker color protection
    - configurable fade out
    - configurable effect duration
    - unique basic effects
    - unique clash effects
    - unique stab effects
    - unique swing effects
    - unique spin effects
    - unique lockup effects
    - unique blaster block effects
    - unique force push effects
    - as there are more than 600 parameters, you can literally invent your own light effects
    - assuming there are 1000 possible values for each parameter (which is a hard under-estimation), you could create 10^{1800} (a one with 1800 zeros) possible effect fonts – that's more than there are atoms in the universe!
    - genious online configuration tool for designing, saving and sharing your light effects: https://sabertec.net/configurator
    - tutorial for the configurator available on https://sabertec.net/downloads
    - also default parameter settings available, so you have already epic light effects even without having to change any of the parameters
    - all light effect parameters are summarized in effect fonts
    - capability of changing the whole light effects on the fly just as changing sound fonts (see below)
- game-changing effect fonts:
    - introduced as an analogue to sound fonts
    - contain all customizable parameters, especially the light effect parameters
    - can define certain effect styles just as sound fonts define certain sound styles
    - you can also have different motion sensitivity in different effect fonts
    - can be changed on the fly
    - up to 99 different effect fonts supported, i.e. you can have up to 99 completely different saber effect styles, not only different colors
    - pairable with sound fonts on the fly as the Golden Harvest remembers with which effect font a sound font was lastly used
    - default effect fonts available on https://sabertec.net/downloads

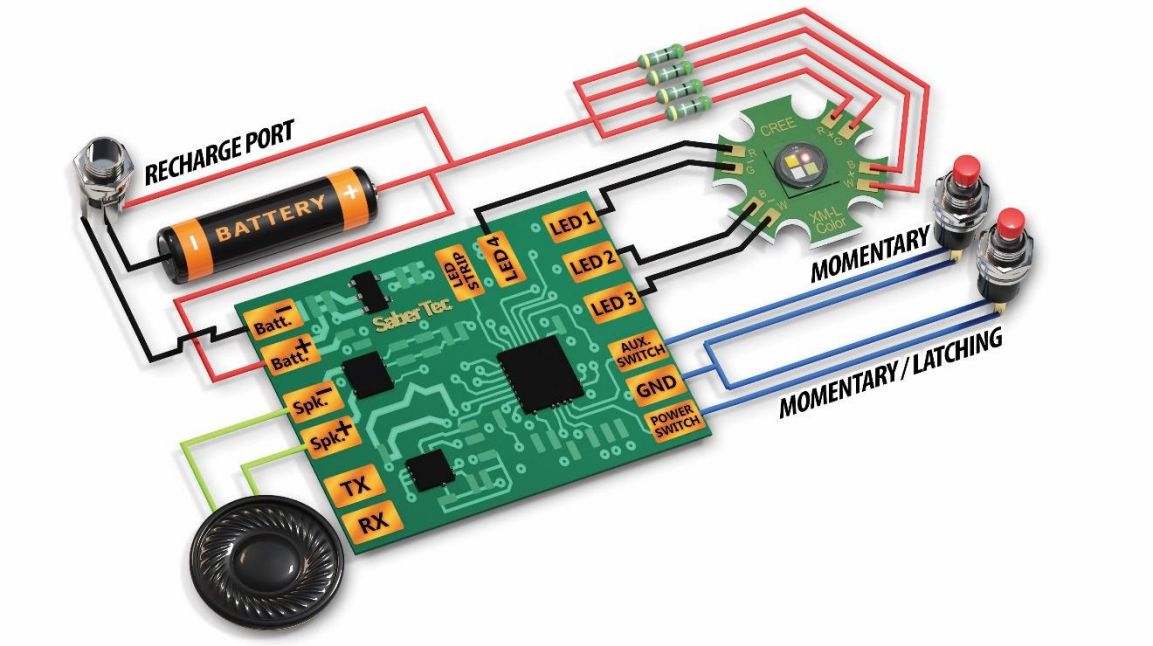
Wiring of the board

After getting in touch with the technical specifications, we'll now come to how to install the Golden Harvest correctly. The general wiring of our board is shown in the pictures below which are wiring examples for different LED setups (there are much more possible than shown though). The first one illustrates the wiring for a LED with three dice, for example a TriCree LED. The second one illustrates the wiring for a LED with four dice as you can even power QuadCrees with our board for example. As you can see, the recharge port and the battery are connected to the pads labelled as "Batt. –" or "Batt. +", respectively, whereas the speaker is connected to the pads labelled as "Spk. –" and "Spk. +". As mentioned above, our board features four different LED channels. These are accessible via the pads labelled as "LED1", "LED2", "LED3" and "LED4" which are connected to the negative of the LED. Please make sure to use appropriate resistors for your LEDs (we don't take any responsibility for possible damage due to using a resistor with too small resistance). To calculate the needed resistivity, have a look at the data sheets of the LEDs you want to use. The four LED channels don't necessarily have to drive your blade LED as you can even drive accent LEDs with the LED channels. If you decide to use only two LED channels for the blade for example, you can use the remaining LED channels to drive a switch LED or crystal chamber LEDs for example. As each LED channel provides up to 4 A, you could even connect up to 200 accent LEDs in parallel to each LED channel (assuming they need 20 mA each). As you'll see in one of the following sections, you can even define custom light effects for the accent LEDs. They can be synchronized to the blade effects, but you can also choose them to be completely independent from the blade effects. This enables you to design independent crystal chamber effects for example.

Another important pad is the "POWER SWITCH" pad to which you connect the power switch as its name already suggests. The power switch can be both a momentary or a latching switch. If you decide to go for a two-switch configuration, you can add a second switch to the "AUX. SWITCH" pad which has to be a momentary switch. The second pin of each switch has to be connected to the "GND" pad. We're currently also working to support LED strips such as Neopixel in future. Most probably, this support will come with a future version of the board. The respective "LED STRIP" pad is already shown in the wiring examples below.

If you want to use a recharge port, please make sure that you connect it correctly as there are different pin layouts possible. When the kill key is pulled, the recharge port has to connect the pad "Batt. –" and the negative of the battery whereas it has to disconnect both when the kill key is inserted.





Hibernation Mode

The Golden Harvest board features also a Hibernation Mode which allows a shelf-time of more than 8 months. The board enters this mode once it wasn't used for 5 minutes while it's powered. This default value can be customized by adjusting the parameter "deep_sleep" which will be described in more detail in the "effect fonts" section below. You can wake the board up from the Hibernation Mode by pressing the power button.

Motion engine

A key feature of our board is a highly sophisticated motion engine that is capable of detecting four different movements: swings, clashs, stabs and spins. Although all of them are detected by a complex algorithm, we tried to reduce the set of parameters for calibrating the motion engine as far as possbile. We were able to limit it to only one sensitivity parameter for each type of motion, apart from spins which intrinsically have two parameters to specify. As it will be described in more detail in the following "effect fonts" section, each type of motion has a threshold parameter that belongs to it. This threshold parameter is a measure for the sensitivity as it specifies how strong a motion has to be in order to detect the respective motion type. The higher the threshold is, the smaller is the sensitivity. In addition to the threshold parameter, spins have a trigger duration parameter. It describes how long the motion has to be stronger than specified by the threshold parameter without any interruption. Virtually, this corresponds to how long you have to rotate your saber without any interruption in order to trigger a spin.

Light and sound effects

Effect fonts

During the development of our Golden Harvest board we focussed in particular on a high customizability of the light and sound effects. Here, especially two aspects were very important to us. On the one hand, our goal was to provide you an overwhelming, realistic experience that let's you dive into another world. On the other hand, the light and sound effects should be versatilely configurable such that you could even create your very own world by giving your lightsaber an unique style. May it be simple, complex or even exotic effects – due to the great variety of our settings, you can realize

almost every effect you can imagine and literally even invent own ones. In the current version of our Golden Harvest board, you can use more than 600 parameters for that which are customizable by using our online configurator. Just unleash your creativity! But even if you're not willing to change many parameters: our Golden Harvest board can also be run without changing any parameter at all as it comes with a default set of parameter values.

Before we give you a detailed description of all the possible settings, we show you how to adjust them. The parameters of all your different light effect styles are summarized into so-called "effect fonts" which we introduced as an analogue to "sound fonts". Each effect font contains all the parameters that define the look and behavior of your saber. One effect font could create a green blade with a strong flickering for example whereas another effect font could create a red, pulsing blade that appears to be unstable due to an ÜberPulse effect. Furthermore, you could even assign a different sound volume or motion sensitivity to different effect fonts for example. And the best thing is: you're also able to change the effect font on the fly while the saber is turned on!

The effect fonts are accessible on your microSD card. They're called "effect[number].txt", whereby [number] enumerates the effect fonts. You can open them with all common text editors, independently of your operating system. So, one possibility of changing the settings of your saber or creating your own light effects is to adjust the respective parameter values in the effect font files manually. This is rather a choice of purists. The other possibility is to use our revolutionizing online configurator to design your custom effects! We're very proud of this configurator as we spent months of hard work and a lot of passion to provide you a flabergasting experience and make your eyes sparkle. In the online configurator, you can adjust all the parameteres of the effect fonts graphically via numerical input fields, multiple-choice fields or sliders. One of the most impressive features of it is a real-time preview of the light effects you're just creating. It shows an animation of how your saber would look like using this effect font. The configurator comes with two different graphic settings as well as two different degrees of complexity. If you're using a desktop computer with an older graphics card or a mobile device, we recommend to use the "fast graphics" option. If you're accessing the configurator with a modern computer with a powerful graphics card, we recommend to use the "fancy graphics" setting. Depending on whether you want to adjust a smaller set of parameters or the full range, you can use the "easy mode" or the "advanced mode", respectively. A more detailed description of our configurator you can find on https://sabertec.net/downloads.

Our configurator is accessible on https://sabertec.net/configurator.

Now, we come to the description of the parameters you can customize manually in the effect fonts or with our configurator. The basic structure of the effect fonts that contain these parameters is shown below. We go through it from top to bottom. First, there are the general settings. These contain all parameters that concern the sound engine, your hardware setup as well as the motion sensitivity. In the following, we give a list of these parameters with a short description. If a parameter defines a time, its values are given in units of 20 ms unless otherwise noted. In this case, a value of 10 would correspond to a time of 200 ms for example.

```
#SaberTec Customizer - Config File
#This config file contains all the parameters defining the behaviour of your saber.
#Here you can customize the light and sound effects as well as the motion detection.



#_____________________________
#
#General Settings
#_____________________________


volume=100

button_off_time=250
button_on_time=10

button_short_time=5
button_long_time=250
button_mode=2

swing_threshold=5000
clash_threshold=3000
stab_threshold=3000
spin_threshold=12000
spin_trigger_duration=300

swing_cooldown=200
clash_cooldown=150
stab_cooldown=150
spin_cooldown=250
blasterblock_protection=10000
forcepush_protection=10000

deep_sleep=300
```

PARAMETER	DESCRIPTION
volume	Defines the loudness of your saber. Ranges from 0 (mute) to 100 (maximum loudness).
button_off_time	Defines how long you have to press the power button to turn your saber off. Recommended value of 250.
button_on_time	Defines how long you have to press the power button to turn your saber on. Recommended value of 10.
button_short_time	In the menu navigation, a short button press triggers certain features (see section „Menu navigation“). This parameter defines how long you have to press a button to be recognized as a short press. Recommended value of 5.
button_long_time	In the menu navigation, a long button press triggers certain features (see section „Menu navigation“). This parameter defines how long you have to press a button to be recognized as a long press. Recommended value of 250.
button_mode	Defines which switch configuration you’re using. A value of 0 corresponds to a single momentary switch, a value of 1 corresponds to one momentary switch and one latching switch and a value of 2 corresponds to two momentary switches.
swing_threshold	Defines the minimum rotation speed of your saber required to trigger a swing. The lower this value, the more sensitive your swing detection is. Recommended values between 3000 and 7000.
clash_threshold	Defines the minimum strength with which you hit an obstacle required to trigger a clash. The lower this value, the more sensitive your clash detection is. Recommeded values between 2000 and 4000.

stab_threshold	Defines the minimum strength with which you perform a stab required to trigger a stab. The lower this value, the more sensitive your stab detection is. Recommended values between 2500 and 6000.
spin_threshold	Defines the minimum rotation speed of your saber required to trigger a spin. The lower this value, the more sensitive your spin detection is. Recommended values between 3000 and 7000.
spin_trigger_duration	Defines how long you have to rotate your saber with the minimum rotation speed defined by „spin_threshold“ without any interruption to trigger a spin. The lower this value, the more sensitive your spin detection is. Recommended values between 300 and 500.
swing_cooldown	Defines how long no other swing can be triggered after a swing has been detected. Can protect swings from a too fast interruption by other swings. Recommended values between 200 and 300.
clash_cooldown	Defines how long no other clash can be triggered after a clash has been detected. Can protect clashs from a too fast interruption by other clashs. Recommended values between 0 and 200.
stab_cooldown	Defines how long no other stab can be triggered after a stab has been detected. Can protect stabs from a too fast interruption by other stabs. Recommended values between 100 and 300.
spin_cooldown	Defines how long no other spin can be triggered after a spin has been detected. Can protect spins from a too fast interruption by other spins. Recommended values between 100 and 300.
blasterblock_protection	Defines how long no motion can be triggered after a blaster block has been triggered. If you want to protect the blaster block completely, set this parameter to a high value. Recommended values between 0 and 10000.
forcepush_protection	Defines how long no motion can be triggered after a force push has been triggered. If you want to protect the force push completely, set this parameter to a high value. Recommended values between 0 and 10000.
deep_sleep	Defines the time after which the board enters the Hibernation Mode if it’s not used used while it’s powered. Values are given in seconds.

These are all parameters that belong to the general settings. Now, let’s consider the light effects which are specified below the parameters of the general settings. The light effects section is separated in 8 different blocks of similar structure. Each of these blocks corresponds to one of the 8 different states the saber can be in: basic, swing, clash, stab, spin, lockup, blaster block or force push. By adjusting the parameters in a certain block, you can customize the light effects of the corresponding saber state. But… wait a minute. This is genious! So, you can even assign special light effects to swings for example! A color change during the swing would be really exotic though, but you could even design a rather smooth light effect change during the swing: for example a weak flicker. So, each time you swing your saber it would flicker, simulating a blade that gets unstable because of sourcing more energy from the diatium power cell and dissipating it. You see – it’s possible to literally invent your own custom light effects!

As the set of parameters that define the light effects is almost the same for all of the states, we consider the basic state as an example in the following. This is the state the saber is in when no motion is triggered. The respective parameters of this state are shown below.

```
#_____________________________
#
#Basic
#_____________________________


led=0,0,1023,1023
accent=0,0,0,1

sync1=1,1,1,1
sync2=0,0,0,0
color_protection=0
flicker_min_intensity=0,0,0,0
flicker_max_intensity=300,300,300,0
flicker_min_delay=1,1,1,1
flicker_max_delay=5,5,5,5
flicker_min_duration=1,1,1,1
flicker_max_duration=1,1,1,1
flicker_period=0,0,0,0
flicker_phase_shift=0,0,0,0
flicker_type=0,0,0,0

pulse_shape=0,0,0,0
pulse_min_intensity=0,0,0,1023
pulse_max_intensity=0,0,0,1023
pulse_intensity_change_period=0,0,0,0
pulse_intensity_phase_shift=0,0,0,0
pulse_min_period=0,0,0,1000
pulse_max_period=0,0,0,1000
pulse_phase_shift=0,0,0,0
pulse_modulation_period=0,0,0,0
pulse_modulation_phase_shift=0,0,0,0
```

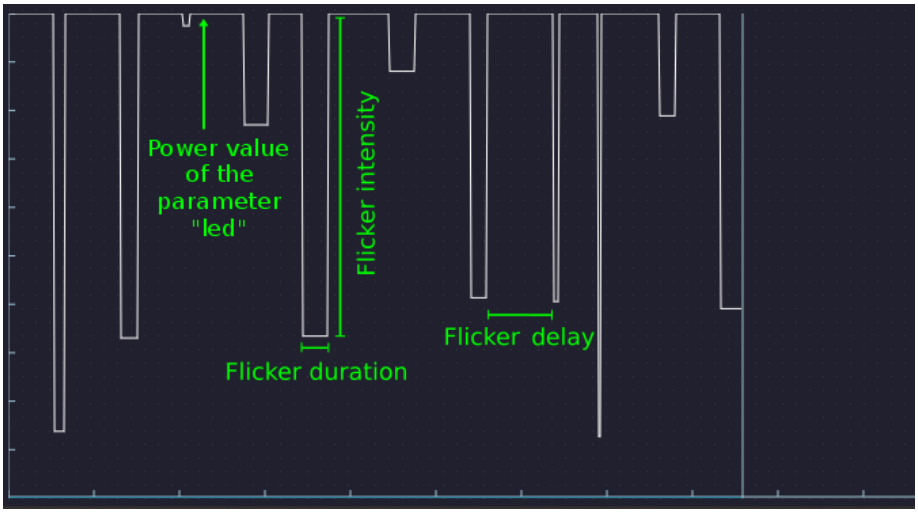
As you can see, the parameters of the basic state consist of three parts: the color part, the flicker part and the pulse part. Almost all parameters have the format "parameter=value1,value2,value3,value4", whereby "value1" is the value of the parameter in LED channel 1, "value2" is the value of the parameter in LED channel 2 and so on. Thus, these parameters can be set for each LED channel independently! This creates an enormous range of different light effects that enables your saber to literally become unique. Let's first consider the color part. In this part, you can define both the color of your blade and whether an LED channel drives a blade LED or an accent LED. One of the fascinating features of our board is that the accent channels are not different from the blade channels with respect to the customizability of the light effects. You can define the full range of light effects also for accent LEDs, completely independet of the blade LEDs if you want! This is a really amazing feature, especially if you want to enlighten a crystal chamber for example. With our board, you can design unique crystal chamber effects, may it be a simple flicker or pulse effect or a more advanced, exotic effect like a color transition. In the following, the two parameters of the color part are described.

PARAMETER	DESCRIPTION
led	Defines the relative power with which the respective LED channels are driven and thus the color of the blade. Ranges from 0 to 1023 for each channel. A value of 0 means that the respective LED channel provides no power, whereas a value of 1023 means that the respective LED channel provides the maximum power. If you set "led=0,300,600,1023" for example, this means that LED channel 1 gets no power, LED channel 2 gets a power value of 300, LED channel 3 gets a power value of 600 and LED channel 4 gets a power value of 1023.
accent	Defines whether an LED channel is used for driving a blade LED or an accent LED. A value of 0 means that the respective LED channel is used for driving a blade LED whereas a value of 1 means that the respective LED channel is used for driving an

accent LED. If you want to use the first three channels for the blade LED and the fourth channel for the lighting a switch for example, you should choose "accent=0,0,0,1". The only difference between blade channels and accent channels is that the latter are already powered when the kill key is pulled, but the saber is not ignited yet.

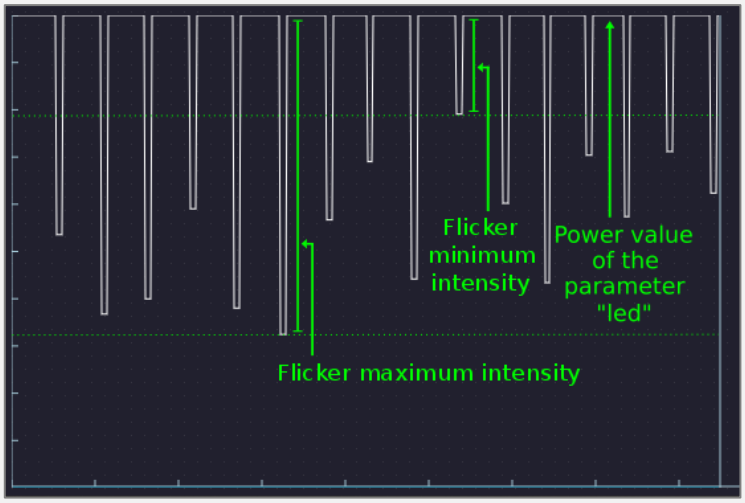
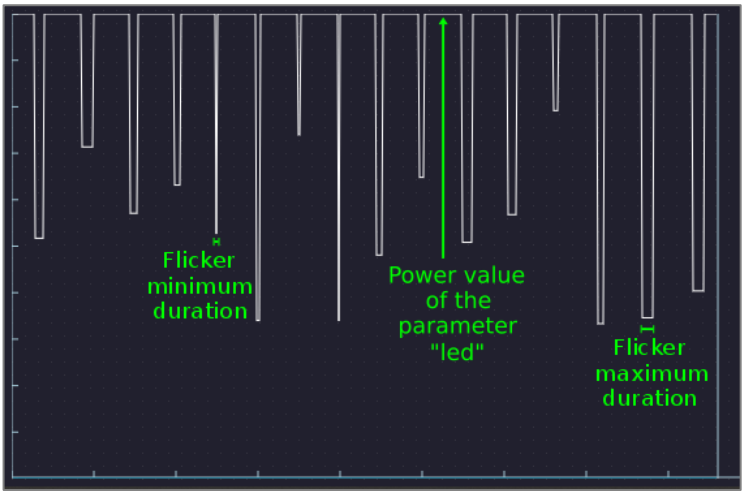
Now, let's turn to the second part. It's called the flicker part as the parameters that it contains define a flicker effect. This effect creates random (but also deterministic if you want) and sudden drops in the power level defined by the parameter "led". It can be characterised by three basic properties: the flicker intensity, the flicker delay and the flicker duration. Hereby, the flicker intensity describes how deep these drops are whereas the flicker delay describes the temporal distance between two successive drops. The flicker duration describes how long these power drops maintain.

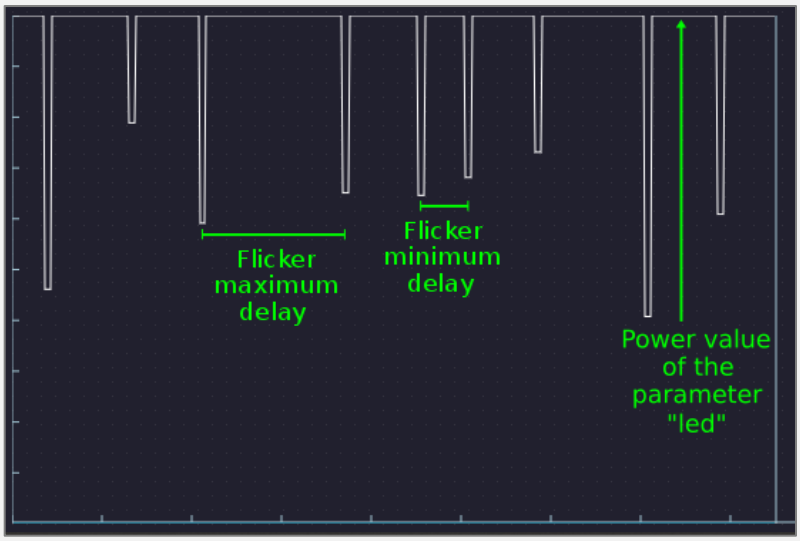
The picture below illustrates these three characteristic properties of the flicker effect graphically. It shows the power an LED gets over time from one LED channel (the term power may not be technically correct here, but it intuitively describes it best). As you can see, the flicker intensity, duration and delay can vary over time which will become clear in the next section.



In the following, we give an overview of the parameters of the flicker part and a short description. The parameters are marked by different colors. These indicate whether they're for users who only want to change a few parameters (black) or for more advanced users who want to unleash the full-blown effect engine (green).

PARAMETER	DESCRIPTION
flicker_min_intensity, flicker_max_intensity	Define the minimum and maximum flicker intensity (see description above). The actual flicker intensity varies randomly over time within the interval between these two values. The higher the flicker intensity, the stronger the flicker effect appears. Each LED channel has its own value. *Example:* *flicker_min_intensity = 200,0,600,200* *flicker_max_intensity = 700,0,600,1023* *In this case, the minimum flicker intensity of channel 1 is 200 whereas the maximum flicker intensity is 700 which means that the actual flicker intensity varies over time within the interval between 200 and 700. In LED channel 2, both*

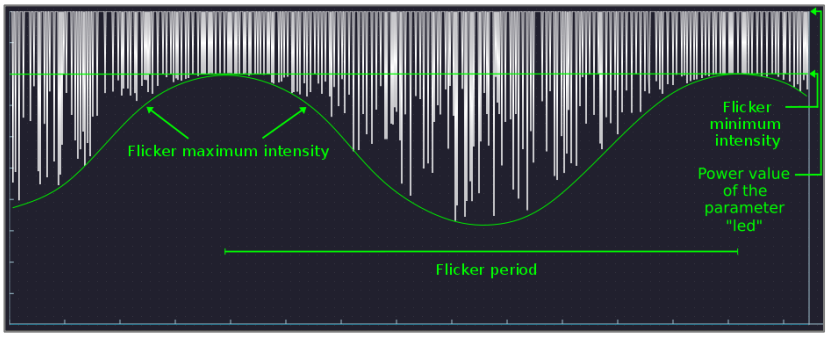
	the minimum and maximum flicker intensity is 0 which corresponds to a disabled flicker effect. The other channels are treated analogously.  The picture above is a screenshot from our online configurator and illustrates the meaning of both the "flicker_min_intensity" and the "flicker_max_intensity" parameter. It shows the change of the LED power over time using a flicker effect with the same intensity parameters as in channel 1 in the previous example. As you can see, the depth of each drop in the LED power varies randomly within the interval between the flicker minimum and maximum intensity, i.e. between 200 and 700 in this case.
flicker_min_duration, flicker_max_duration	Define the minimum and maximum flicker duration (see description above). The actual flicker duration varies randomly over time within the interval between these two values. The smaller the duration, the shorter the power drops are. Each LED channel has ist own value. *Example:* *flicker_min_duration = 1,1,5,1* *flicker_max_duration = 5,1,5,5* *In this case, the minimum flicker duration is 1 in LED channel 1 and the maximum flicker duration is 5. This means that the actual flicker duration varies randomly over time within the interval between 1 and 5. In LED channel 2, both the minimum and maximum flicker duration are 1. This means that every power drop has duration 1. The other channels are treated analogously.* 

	The picture above is a screenshot from our online configurator and illustrates the meaning of both the "flicker_min_duration" and the "flicker_max_duration" parameter. It shows the change of the LED power over time using a flicker effect with the same duration parameters as in channel 1 in the previous example. As you can see, the duration of each drop in the LED power varies randomly within the interval between the flicker minimum and maximum intensity, i.e. between 1 and 5 in this case.
flicker_min_delay, flicker_max_delay	Define the minimum and maximum flicker delay (see description above). The actual flicker delay varies randomly within the interval between these two values. The smaller the flicker delay, the higher the speed of the flicker effect appears. Each LED channel has its own value. *Example:* *flicker_min_delay = 1,1,1,1* *flicker_max_delay = 5,1,5,10* *In this case, the minimum flicker delay is 1 in LED channel 1 and the maximum delay is 5. This means that the actual flicker delay varies randomly over time within the interval between 1 and 5. In LED channel 2, both the minimum and maximum flicker delay are 1. This means that the actual flicker delay is always 1.*  The picture above is a screenshot from our online configurator and illustrates the meaning of both the "flicker_min_delay" and the "flicker_max_delay" parameter. It shows the change of the LED power over time using a flicker effect with the same delay parameters as in channel 1 in the previous example. As you can see, the delay between two successive drops in the LED power varies randomly within the interval between the minimum and maximum flicker delay, i.e. between 10 and 50 in this case. These values are usually too big for a conventional flicker effect and were only chosen that high for a better visualization.
flicker_period, flicker_phase_shift	The *maximum flicker intensity* changes smoothly over time back and forth (sine-like) between "flicker_min_intensity" and "flicker_max_intensity". Thus, the interval between the minimum and maximum flicker intensity changes over time. The parameter "flicker_period" describes how fast or slow this change occurs. The higher the value, the slower the transition happens. Ranges from 0 (no change over time at all) to 65535. When your saber is turned on, the maximum flicker intensity first has the value "flicker_max_intensity", before it decreases to the value "flicker_min_intensity". The parameter "flicker_phase_shift" can shift this

behavior so that the maximum flicker intensity starts with the value "flicker_min_intensity" instead for example. Ranges from 0 (no shift) to 359 (maximum shift). We call the intensity transition effect PhaseFlicker. It can be disabled by setting "flicker_period" to 0. In this case, the maximum flicker intensity hast he value "flicker_max_intensity" and remains constant. We call the intensity transition effect PhaseFlicker. Each LED channel has ist own value.

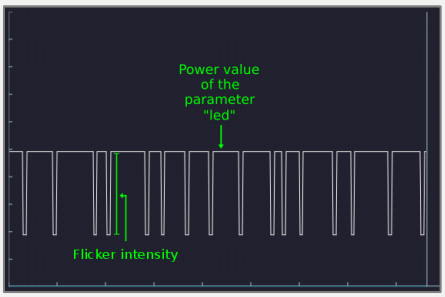
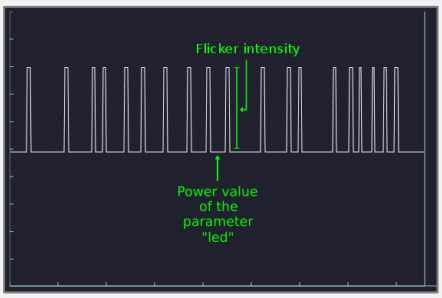
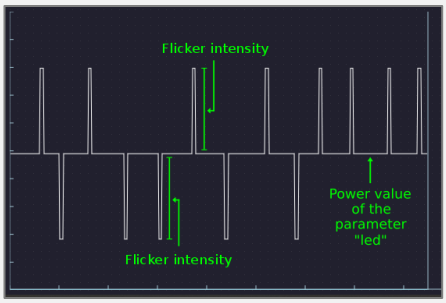
Example:
flicker_period = 600,0,300,1000
flicker_phase_shift = 0,0,180,0

In this case, the flicker period is 600 in LED channel 1 which means that the maximum flicker intensity varies over time between "flicker_min_intensity" and "flicker_maximum_intensity" with a transition duration of 600. This is especially interesting if you choose "flicker_minimum_intensity=0" as in this configuration, your saber would undergo a smooth transition between two phases: no flicker present and flicker present. This could simulate a saber that undergoes a transition between a stable and an unstable state for example. In LED channel 2, there is no smooth change in the flicker intensity over time (the flicker transition effect is disabled). Only the original random changes within the interval between "flicker_min_intensity" and "flicker_max_intensity" remain. The other channels are treated analogously.



The picture above is a screenshot from our online configurator and illustrates the meaning of the "flicker_period" parameter. It shows the change of the LED power over time using a flicker effect with the same "flicker period" parameter value as in channel 1 in the previous example. As you can see, the depth of each drop in the LED power varies randomly within the interval between the flicker minimum and maximum intensity, whereby the latter changes periodically between the parameter values "flicker_min_intensity" and "flicker_max_intensity".

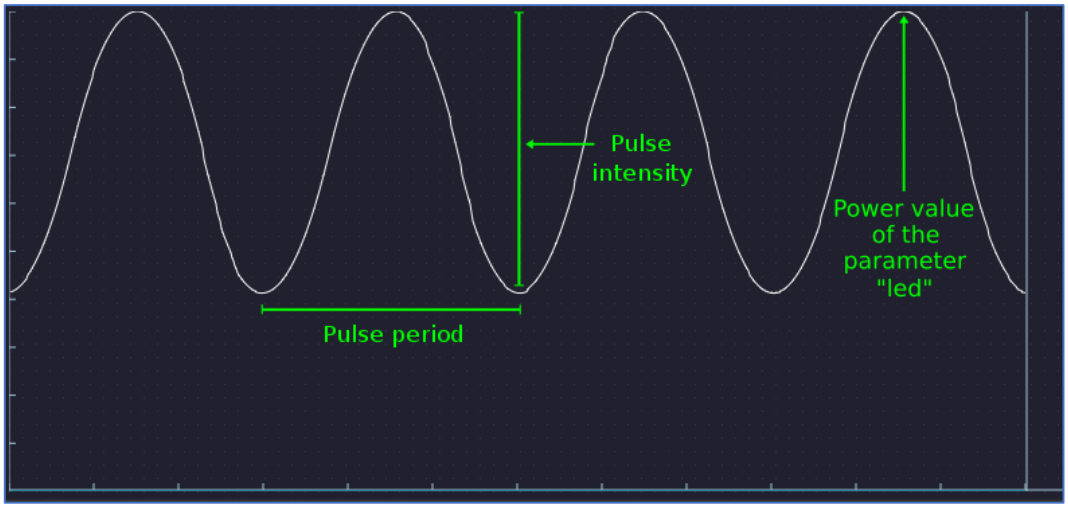
sync1, sync2	As mentioned above, both the flicker duration and delay vary randomly over time between their respective minimum and maximum values defined by the corresponding parameters. This leads to the fact that even if you use the same duration and delay parameter values for different LED channels, the drops in the LED power don't necessarily occur at the same time. By using the "sync1" and "sync2" parameters, you can synchronize the LED channels such that the drops in the LED power always occur at the same time. Both parameters can be 0 or 1 for each channel. All LED channels with a value of 1 are synchronized. *Example:* *sync1=1,0,1,0* *sync2=0,1,0,1*

	In this case, the LED channels 1 and 3 are synchronized as well as the LED channels 2 and 4. Now, you also see why there are two synchronization parameters: to cover all possible combinations, you need two. This can be interesting for a crystal chamber for example. By using the synchronization parameters, you could both synchronize the blade LED channels and the chamber LED channels.
color_protection	Can be used in combination with the synchronization parameters. If you synchronize LED channels, you force the drops in the LED power that the flicker effect creates to occur at the same time. By setting the "color_protection" parameter to 1, you also synchronize the flicker intensity which avoids color fluctuations due to the flicker effect. A value of 0 disables the synchronization of the flicker intensity, which can create impressive color flashes. This parameter has only one value for all LED channels.
flicker_type	Defines the type of the flicker effect. A flicker effect cannot only create sudden power *drops* which is called "subtractive type", but also sudden power *peaks* which is called "additive type". Furthermore, a flicker effect also can mix both types by creating sudden power drops and peaks in a random manner which is called "mixed type". A value of 0 corresponds to the subtractive type, a value of 1 to the additive type whereas a value of 2 corresponds to the mixed type. Each LED channel has its own value. *Example:* *flicker_type = 0,1,0,0* *In this case, channel 1 displays a flicker of type 0 which corresponds to a subtractive type whereas channel 2 displays a flicker of type 1 which corresponds to an additive type. An additional flicker type can be especially interesting as on top clash effect or to enlighten a crystal chamber. By using it, you could configure a "chamber flash" that creates random light bursts. The other channels are treated analogously.* subtractive type (0)  additive type (1)  mixed type (2) 

	The pictures above are screenshots from our online configurator and illustrate the meaning of the "flicker type" parameter. It shows the change of the LED power over time using a subtractive, additive or mixed flicker effect, respectively. As you can see, a subtractive flicker effect creates drops in the LED power whereas an additive one creates peaks and a mixed one creates both drops and peaks. For the additive and the mixed flicker effect, all the other parameters described above have the same meaning as for the subtractive flicker effect.

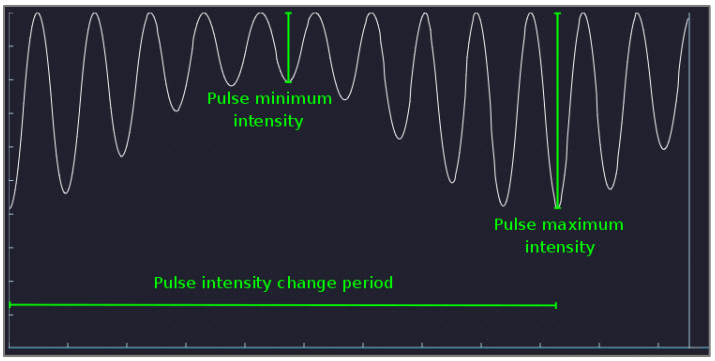
In the following, we'll turn to the pulse part of the basic light effects. It's situated below the flicker part and defines – as its name suggests – a pulse effect. In contrast to the flicker effect, the pulse effect creates periodic and smooth drops (sine-like) in the LED power that is defined by the parameter "led". It can be characterised by two basic properties: the pulse intensity and the pulse period. Hereby, the pulse intensity describes how deep the pulse effect smoothly digs into the LED power. In addition to that, the pulse period describes the duration of one drop. Thus, it is a measure for the pulse speed. The smaller the values of the pulse period, the faster the LED pulses.

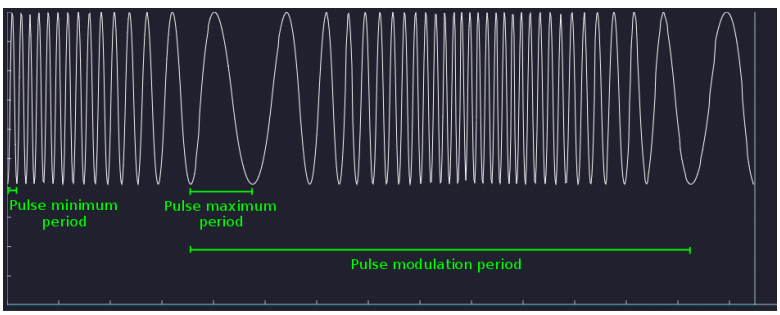
The picture below illustrates these two characteristic properties of the pulse effect graphically. It shows the power an LED gets over time from one LED channel using a pulse effect. Both pulse intensity and period can also vary over time which will become clear in the next section.

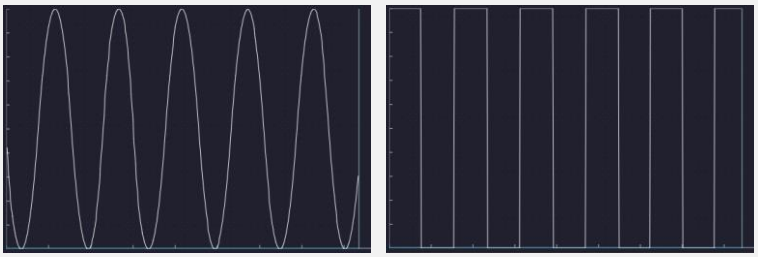
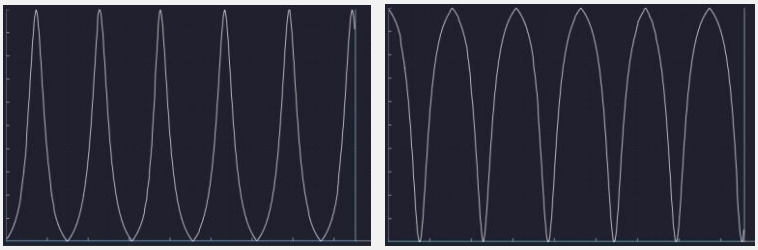
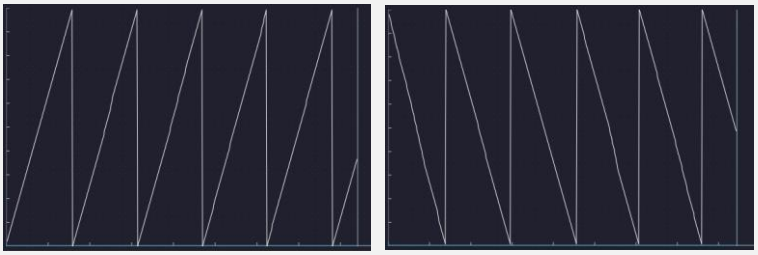


In the following, we give a list of all the parameters of the pulse part and a short description. As for the flicker part, the parameters are marked by different colors. These indicate whether they're for users who only want to change a few parameters (black) or for more advanced users who want to unleash the full-blown effect engine (green).

PARAMETER	DESCRIPTION
pulse_min_intensity, pulse_max_intensity	Define the minimum and maximum pulse intensity. The actual pulse intensity changes over time periodically within the interval between "pulse_min_intensity" and "pulse_max_intensity" (see picture below). We call this intensity change effect *"ÜberPulse"*. It can be disabled by setting "pulse_min_intensity" and "pulse_max_intensity" to the same value. By doing so, one obtains a conventional pulse effect with a constant pulse intensity. Each LED channel has its own value. *Example:* *pulse_min_intensity=200,300,0,0* *pulse_max_intensity=600,300,1023,0*

	In this case, LED channel 2 shows a pulse effect with a minimum intensity of 200 and a maximum intensity of 600. Thus, drops in the LED power are periodically created with a depth that changes over time repetitively back and forth within the interval between 200 and 600. By that, you can create a pulse effect that starts almost invisible and gets stronger and stronger over time for example. LED channel 1 shows a pulse effect with an intensity of 300 which means that the pulse effect periodically digs into the LED power down to a depth of 300. The other LED channels are treated analogously.
pulse_intensity_change_period, pulse_intensity_phase_shift	The parameter "pulse_intensity_change_period" defines how long the periodic change of the pulse intensity from the minimum to the maximum value and back again takes (see picture below). Thus, it is a measure for how fast the transition between a pulse with intensity "pulse_min_intensity" and one with intensity "pulse_max_intensity" occurs. The smaller the value of "pulse_intensity_change_period", the faster the transition occurs. If you turn your saber on and if you have set a pulse effect with an intensity transition, the pulse will begin with its maximum intensity, before changing to its minimum intensity (see picture below). You can change this behavior by adjusting the parameter "pulse_intensity_phase_shift". By changing its value, you can shift the transition such that the pulse begins with its minimum intensity instead of its maximum intensity for example. Each LED channel has its own value. *Example:* *pulse_intensity_change_period=600,0,1000,0* *pulse_intensity_phase_shift=0,0,180,0* *In this case, LED channel 1 shows a pulse intensity change with a period of 600. In LED channel 2, the pulse intensity change is disabled as a value of 0 means that the pulse intensity doesn't change over time. Its constant value is "pulse_maximum_intensity". The other LED channels are treated analogously.*  The picture above is a screenshot from our online configurator and illustrates the meaning of the "pulse_min_intensity", "pulse_max_intensity" and "pulse_intensity_change_period" parameter. It shows the change of the LED power over time using a pulse effect with the same parameter value as in channel 1 in the previous two examples. As you can see, the actual pulse intensity, i.e the depth of each drop in the LED power varies smoothly within the interval between the minimum and maximum pulse intensity.

pulse_min_period, pulse_max_period	Define the minimum and maximum period of the pulse effect. The actual period of the pulse effect changes periodically within the interval between these two values. As mentioned above, the pulse period is a measure for the pulse speed. The smaller the pulse period, the higher the pulse speed. Thus, you can create a pulse whose speed changes periodically over time. We call this speed transition effect HyperPulse. You can disable it by setting both parameters "pulse_min_period" and "pulse_max_period" to the same value. This creates a conventional pulse effect. Each LED channel has its own value. *Example:* *pulse_min_period=100,100,1000,1000* *pulse_max_period=100,1000,1000,1000* *In this case, the pulse period is constant at a value of 100 in LED channel 1 as both minimum and maximum period have the same value. LED channel 2 shows a pulse with a period that changes back and forth within the interval between 100 and 1000. Thus, you get a pulse that periodically decreases and increases its speed. This can create an amazing look as it simulates a blade that undergoes a transition between a calm and an aggressive phase. The other LED channels are treated analogously.*
pulse_modulation_period, pulse_modulation_phase_shift	The parameter "pulse_modulation_period" defines how long the periodic change between the minimum and maximum period, i.e. the maximum and minimum speed takes. Thus, it is a measure for how fast this speed transition occurs. The smaller the value of "pulse_modulation_period", the faster the transition occurs. If you turn your saber on and if you have set a pulse effect with speed transition, it starts with its minimum period, before continously changing to its maximum period (see picture below). Each LED channel has its own value. *Example:* *pulse_modulation_period=1000,10000,0,0* *pulse_modulation_phase_shift=0,180,0,0* *In this case, the actual pulse period changes within the interval between "pulse_minimum_period" and "pulse_maximum_period" with a transition duration of 1000 in LED channel 1. LED channel 2 shows a pulse with a speed transition duration of 10000 and a phase shift of 180. This means that the pulse starts with ist maximum period instead of its minimum period. The other LED channels are treated analogously.* 

	The picture above is a screenshot from our online configurator and illustrates the meaning of the "pulse_min_period", "pulse_max_period" and the "pulse_modulation_period" parameter. It shows the change of the LED power over time using a pulse effect with the same parameter values as in channel 1 in the previous two examples. As you can see, the pulse speed varies smoothly within the interval between the minimum and maximum pulse speed.
pulse_shape	Defines the shape of the periodic, smooth drops into the LED power that the pulse effect creates. A value of 0 corresponds to a sine shape (which was used in the pictures above), a value of 1 to an impulse shape, a value of 2 to an inverted impulse shape, a value of 3 to a rising sawtooth shape, a value of 4 to a falling sawtooth shape and a value of 5 to a rectangular shape (see the picture below). Each LED channel has its own value. *Example:* *pulse_shape=0,1,0,0* *In this case, the pulse effect in LED channel 1 creates sine shaped periodic, smooth drops in the LED power. LED channel 2 shows a pulse effect that creates impulse shaped periodic, smooth drops in the LED power. The other LED channels are treated analogously.* sine shape (0) rectangular shape (5)  impulse shape (1) inverted impulse shape (2)  rising sawtooth (3) falling sawtooth (4)  The picture above is a screenshot from our online configurator and illustrates the meaning of the "pulse_shape" parameter. It shows the

	change of the LED power over time using a pulse effect with sine, impulse, inverted impulse, rising sawtooth, falling sawtooth or rectangular shape, respectively. For the pulse shapes different than sine, all the other parameters described above have the same meaning as for the sine shaped pulse effect.

So far, these are all parameters that define the light effects of the basic state. As mentioned above, the other states, namely clash, stab, swing, spin, lockup, blaster block and force push have exactly the same parameters. Thus, they can be customized in the same manner as the basic state. For all of these additional states, except from lockup, there are even two more parameters each in comparison to the basic state: effect duration and fade out. Let's have a look at the clash state for example. All the clash parameters are shown below.

```
#_____________________________
#
#Clash
#_____________________________


cl_led=1023,0,1023,1023

cl_sync1=1,1,1,1
cl_sync2=0,0,0,0
cl_color_protection=0
cl_flicker_min_intensity=1023,1023,1023,0
cl_flicker_max_intensity=1023,1023,1023,0
cl_flicker_min_delay=1,1,1,1
cl_flicker_max_delay=1,1,1,1
cl_flicker_min_duration=1,1,1,1
cl_flicker_max_duration=1,1,1,1
cl_flicker_period=0,0,0,0
cl_flicker_phase_shift=0,0,0,0
cl_flicker_type=0,0,0,0

cl_pulse_shape=0,0,0,0
cl_pulse_min_intensity=0,0,0,1023
cl_pulse_max_intensity=0,0,0,1023
cl_pulse_intensity_change_period=0,0,0,0
cl_pulse_intensity_phase_shift=0,0,0,0
cl_pulse_min_period=0,0,0,1000
cl_pulse_max_period=0,0,0,1000
cl_pulse_phase_shift=0,0,0,0
cl_pulse_modulation_period=0,0,0,0
cl_pulse_modulation_phase_shift=0,0,0,0

cl_duration=100
cl_fade_out=60
```

As you can see, you have the same degree of customizability just as in the basic state, whereby the parameters here have the additional prefix "cl_" which indicates that they belong to the clash state. In addition to these settings you also have the parameters "cl_duration" and "cl_fade_out". The parameter "cl_duration" describes how long the clash light effects are shown, given relatively to the length of the clash sounds. It ranges from 0 to 100, which corresponds to a duration of 0 % of the clash sounds (disabled clash light effects) or a duration of 100 % of the clash sounds (light effects are shown until the end of the clash sounds is reached), respectively. This is an amazingly powerful customization parameter. On the one hand, you could create light effects that are very short like an epic, sudden flash when a clash is triggered for example. On the other hand, you could create light effects that maintain very long which could simulate a blade that is strongly affected by the clash and takes a long time to stabilize again. Here, you have again the full freedom in designing your very own, unique light effects as you could also choose values of 70 % for example or any other value within the respective

range. The parameter "cl_fade_out" forms a congenial feature together with the clash duration. When a clash is triggered and the clash light effects occur, they start to smoothly fade out to the basic light effects. This is done by mixing continuously more and more of the basic effects to the clash effects. The parameter "cl_fade_out" describes how fast this transition between the clash and basic effects occurs. It is given relatively to the duration of the clash light effects and ranges from 0 to 100. A value of 0 corresponds to a duration of 0 % of the clash light effects duration (disabled fade out, the clash effects are immediately interrupted by the basic effects when they end). A value of 100 corresponds to a duration of 100 % of the clash light effects duration (smoothest transition possible). Using the fade put parameter, you can create flabbergasting effects. If you have a red blade with a white LED flashing on clash and if you set a fade out for example, the blade color smoothly changes from white over white-red to red again as the blade stabilizes after the clash.

The other states, namely, stab, swing, spin, blaster block and force push also have a duration and fade out parameter with the very same functionality as in the clash state each. The only thing that is different between these states is the prefix of the parameters.

Sound fonts

Just as effect fonts define the light effects of your saber, sound fonts define its sound effects. Each sound font contains a set of sound files for all effects our board supports. Thus, they contain swing sounds, clash sounds, stab sounds, spin sounds, lockup sounds, blaster block sounds, force push sounds, boot sounds, power sounds, menu sounds and a hum sound. Our board supports up to 99 different sound files of each of these sound effects. In case you have more than one sound file for a sound effect, it's chosen randomly which sound file is played when the respective effect is triggered. So, you see there is a huge variety of different sound files that are summarized to a sound font. Sound fonts usually differ with respect to their style and the atmosphere they create. Whereas some sound fonts could give you a Jedi-feeling, others may contain Sith-like sounds. The sound fonts are saved on the microSD card of your saber, each of them as an own folder of the format "sound[number]", whereby [number] enumerates them (see the next section). If you want to add a sound font to your saber, just create a new folder of the format mentioned above with a number that isn't in use yet and put the sound files in there. In order ensure that the sound fonts work properly, please make sure that your microSD card is formatted in the FAT32 file system. The microSD card that comes with your Golden Harvest is already formatted in this file system. There are many sound fonts available online. Make sure to visit www.saberfont.com, which is the biggest source for sound fonts. You can use all pre-mixed sound fonts, even without additional renaming of the sound files. Most of the sound fonts on saberfont.com are pre-mixed. You can get a list of all of them on http://www.saberfont.com/.

Default sound font package

Our Golden Harvest board comes with a set of default sound fonts that are already saved on the microSD card. Currently, this package contains the following sound fonts (listed in randomized order):

- **"Balance" by LordBlako Saber Fonts:**
  A powerful font belonging to a character who tries to bring balance between the light and the dark side of the force. The default menu sounds of our Golden Harvest board were also provided by LordBlako Saber Fonts. Check out more of the genious sound fonts by LordBlako Saber Fonts on http://www.saberfont.com/Lord-Blako_c_15.html
- **"Psy-Borg" by CrystalSoniX:**
  This mind blowing font belongs to the Psy-Borg who is the ultimate melding of man and machine. With his human skeleton, nervous system and mind (sort of) intact, the rest resembles (and may in fact be) Jensaarai warrior's armour: reflecting his totem beast the acklay, while also similar to the Japanese Samurai of our own universe. This sound font

contains also custom menu sounds. Check out more of the amazing work of CrystalSoniX on http://www.saberfont.com/Crystal-Sonix-_c_33.html.

- **"Guardian" by LDN Sabers:**
  This is the strong font of a a Jedi who survived Order 66. He was forced to construct a new type of lightsaber, which was a double bladed wooden pike to be able to hide from the empire. Check out more of the epic fonts by LDN Sabers on http://www.saberfont.com/LDN-Sabers_c_79.html.
- **"Daddy Issues" by Fonts by Fourzze:**
  A faithful recreation of the hero sound of "Star Wars: The Return of the Jedi". This sound font is also used in our online configurator. Check out more of the mind-blowing fonts by Fonts by Fourzze on http://www.saberfont.com/Fonts-by-Fourzze_c_40.html.

Guys, thank you so much for supporting our Golden Harvest project by providing these amazing sound fonts!

The default sound font package will be extended soon. Stay tuned! If you purchase a Golden Harvest board now, you'll get the sound fonts that'll be added for free of course!

Sound file requirements

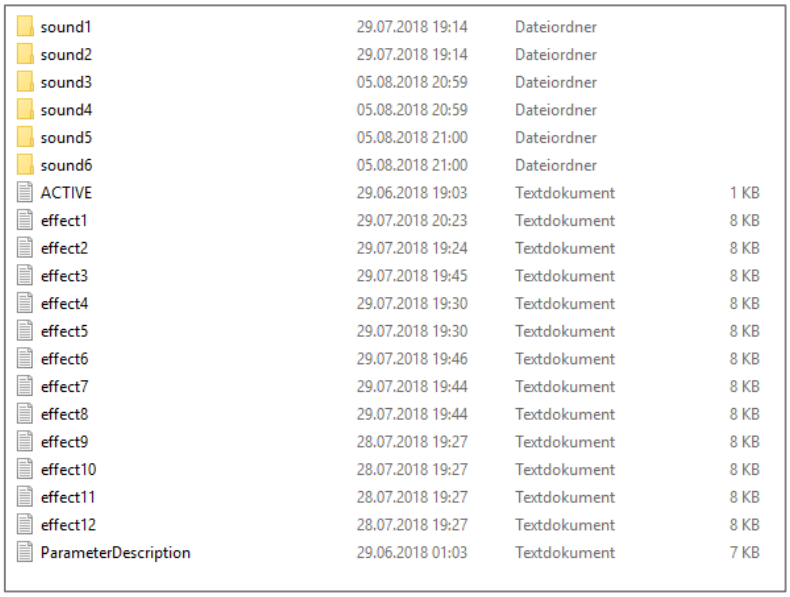
Our board recognizes all pre-mixed sound fonts from saberfont.com even without additional renaming required. Nevertheless, we recommend to use our own naming system which is given in the following. If you want to design your own sound font, please make sure that the sound files have a sampling rate of 22,050 kHz, a resolution of 16 bit and all sounds are mixed with the hum, at least at the beginning and the end. The following sound types are used by our Golden Harvest board:

- font.wav:
  identification sound of the font
- boot[number].wav:
  sound played after the kill key is pulled
- pwron[number].wav:
  ignition sound of your saber
- pwroff[number].wav
  retraction sound of your saber
- hum[number].wav:
  sound of your saber when no motion is triggered
- clash[number].wav:
  sound of your saber when triggering a clash
- stab[number].wav:
  sound of your saber when triggering a stab
- swing[number].wav:
  sound of your saber when triggering a swing
- spin[number].wav:
  sound of your saber when triggering a spin
- lock[number].wav:
  sound of your saber when triggering lockup
- blast[number].wav:
  sound of your saber when triggering a blaster effect
- force[number].wav:
  sound of your saber when triggering a force effect

Hereby, the sound files are enumerated contrinuously by [number]. There are up to 99 different files possible for each sound type which are selected randomly when triggered. Our Golden Harvest board uses default files for the menu sounds, so you don't need to add them to your sound font. If you want to add them in order to get underline the overall atmosphere your sound font creates though, you would have to create the two files "change.wav" and "save.wav". The first is played when the effect font is changed and the latter is played when the effect font menu is leaved.

Folder structure on the microSD card

As we described both effect fonts and sound fonts in the sections above, let's have a look now at the folder structure on the microSD card of your Golden Harvest board. It is shown in the picture below. There, you can see the folders with the name "sound[number]" which contain the different sound fonts as described in the previous section. In this example, there are 6 sound fonts installed, but as mentioned above, our board supports even up to 99 different sound fonts. These would then be enumerated from 1 to 99. Below the sound font folders, you see a text file called "ACTIVE.txt". This file contains the information about which sound font you used last time, so that when you turn the saber on again, this sound font will be automatically chosen. What you cannot see here is that the sound font folders contain also a file called "ACTIVE.txt". That one contains the information about which effect font you used last time. Thus, you can combine both sound fonts and effect fonts on the fly and the pairing will be saved. This is a great feature as you can assign a red blade to a dark font and a blue or green blade to a light font for example. Further down in the folder structure, you find text files called "effect[number].txt", whereby [number] enumerates them. These files contain the different effect fonts. In this example, there are 12 effect fonts installed, but as mentioned above, our board supports even up to 99 different effect fonts. These would then be enumerated from 1 to 99. Below the effect fonts, there's another file called "ParameterDescription.txt". As its name already suggests, it contains a short description of all the parameters you can customize in the effect fonts. For a more detailed description, see the section "effect fonts".



sound1	29.07.2018 19:14	Dateiordner	
sound2	29.07.2018 19:14	Dateiordner	
sound3	05.08.2018 20:59	Dateiordner	
sound4	05.08.2018 20:59	Dateiordner	
sound5	05.08.2018 21:00	Dateiordner	
sound6	05.08.2018 21:00	Dateiordner	
ACTIVE	29.06.2018 19:03	Textdokument	1 KB
effect1	29.07.2018 20:23	Textdokument	8 KB
effect2	29.07.2018 19:24	Textdokument	8 KB
effect3	29.07.2018 19:45	Textdokument	8 KB
effect4	29.07.2018 19:30	Textdokument	8 KB
effect5	29.07.2018 19:30	Textdokument	8 KB
effect6	29.07.2018 19:46	Textdokument	8 KB
effect7	29.07.2018 19:44	Textdokument	8 KB
effect8	29.07.2018 19:44	Textdokument	8 KB
effect9	28.07.2018 19:27	Textdokument	8 KB
effect10	28.07.2018 19:27	Textdokument	8 KB
effect11	28.07.2018 19:27	Textdokument	8 KB
effect12	28.07.2018 19:27	Textdokument	8 KB
ParameterDescription	29.06.2018 01:03	Textdokument	7 KB

Menu navigation

Navigation using only one momentary switch

As mentioned above, all of the features of our board are accessible even with using only one momentary switch. With it, you can change the effect fonts and the sound fonts as well as trigger other functions by the commands given in the following.

FUNCTION	TRIGGERING WITH ONE MOMENTARY SWITCH
POWER ON	Press for "button_on_time" (default: 10 ms)
ENTER EFFECT MENU	Long press, followed by short press
∟ CHANGE EFFECT FONT	∟ Short press
SAVE AND LEAVE EFFECT MENU	Long press
CHANGE SOUND FONT	Long press before igniting the blade
LOCKUP	Short press, followed by holding the switch
BLASTER BLOCK	Short press
FORCE PUSH	Two fast short presses (like a double click)
MUTE	Holding the switch during pulling of the kill key
POWER OFF	Press for "button_off_time" (default: 1000 ms)
WAKE UP FROM HIBERNATION MODE	Short press

You can customize the menu navigation in the effect fonts. There, the exact time how long you have to hold the switch to trigger a power on, short press, long press or power off, respectively, is specified. See the section "effect fonts" for a more detailed description.

Navigation using one momentary and one latching switch

Alternatively to using only one momentary switch, you can also use two switches: one momentary and one latching switch. With two switches, you can change the effect fonts and the sound fonts as well as trigger other functions by the commands given in the following. Here, both short press and long press refer to the momentary switch.

FUNCTION	TRIGGERING WITH ONE MOMENTARY SWITCH
POWER ON	Turn on latching switch
ENTER EFFECT MENU	Long press, followed by short press
∟ CHANGE EFFECT FONT	∟ Short press
SAVE AND LEAVE EFFECT MENU	Long press
CHANGE SOUND FONT	Long press before igniting the blade
LOCKUP	Short press, followed by holding the switch
BLASTER BLOCK	Short press
FORCE PUSH	Two fast short presses (like a double click)
MUTE	Holding the switch during pulling of the kill key
POWER OFF	Turn off latching switch
WAKE UP FROM HIBERNATION MODE	Short press

You can customize the menu navigation in the effect fonts. There, the exact time how long you have to hold the switch to trigger a power on, short press, long press or power off, respectively, is specified. See the section "effect fonts" for a more detailed description.

Navigation using two momentary switches

You can also use two momentary switches. With them, you can change the effect fonts and the sound fonts as well as trigger other functions by the commands given in the following. The two switches are reffered to as control and power switch according to the wiring examples in the section "Wiring of the board".

FUNCTION	TRIGGERING WITH TWO SWITCHES
POWER ON	Short press on power switch
CHANGE EFFECT FONT	Two fast presses on power switch
CHANGE SOUND FONT	Short press on control switch before the blade is ignited
LOCKUP	Holding control switch
BLASTER BLOCK	Short press on control switch
FORCE PUSH	Two fast presses on control switch
MUTE	Holding of control switch during power on
POWER OFF	Long Press on power switch
WAKE UP FROM HIBERNATION MODE	Short press on power switch

You can customize the menu navigation in the effect fonts. There, the exact time how long you have to hold the switch to trigger a power on, short press, long press or power off, respectively, is specified. See the section "effect fonts" for a more detailed description.