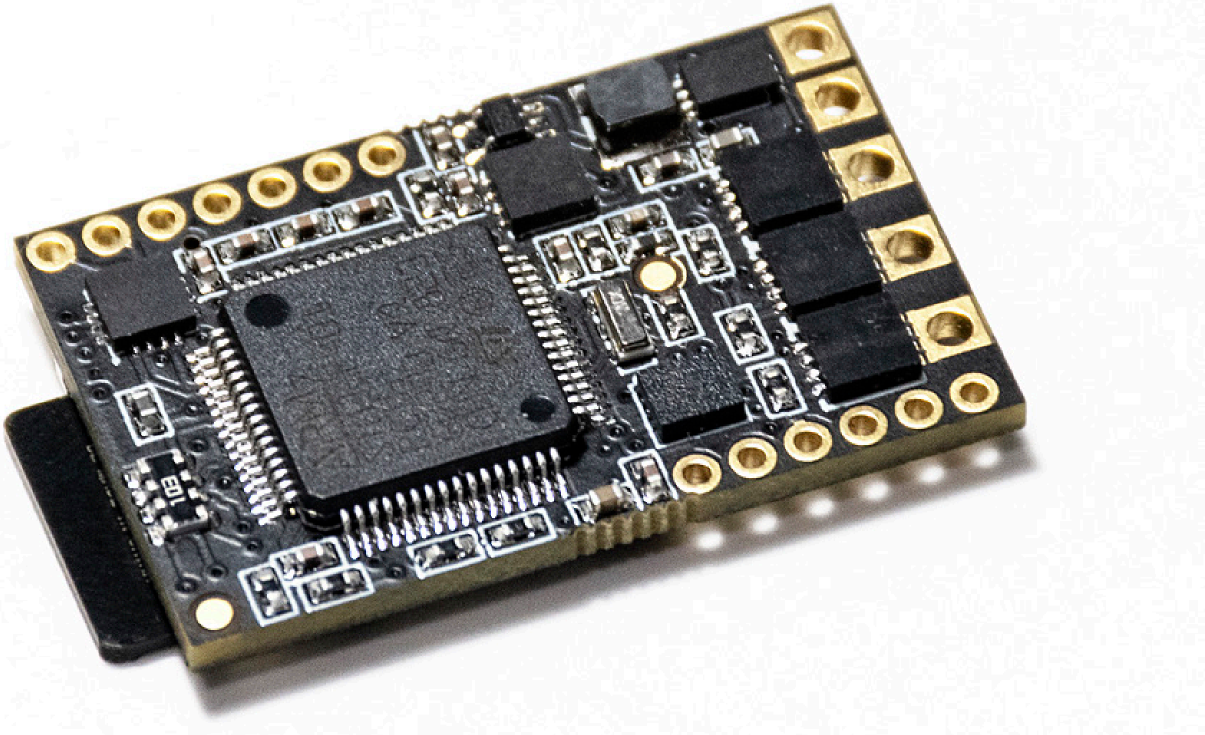
VERSO

Entry-Level saber sound board



User Manual
v2.0

© KR-sabers

2020

Contents

updated:
02.03.2021

FEATURES

Specifications and features:

– Dimensions: 18.9x29.9x4.5mm (+2.4mm longer with micro SD card)
– Power supply: 3-5 Volts, up to 10A per LED channel output; single Li-Ion 3.6-3.7V (low 2.6V, full 4.2V) battery recommended
– 0.55 mA current consumption in Deep Sleep
– 3 high power LED channels
– 1 or 2 buttons control options
– Speaker: 4 ohm or 8 ohm, 2W-5W
– up to 16 sound banks/fonts
– up to 16 blade color profiles
– up to 16 blade effects
– smoothswings + accent swings
– Sound FX (WAV sound files): boot, hum, swing, clash, ingnition, retraction, lockup, blaster blocks, low battery, font name etc.
– Light FX: ramp up, ramp down, fade, blade flickering, flash on clash, fire, unstable, rainbow
– Micro SD card: 4-16Gb Class 4-10 by SanDisk brand recommended
– 3 Watts 5V powered sound amplifier, 16-bit digital output
– 44kHz or 22kHz WAV audio files sample rate
– "SmoothSwing", monophonic, polyphonic sound fonts support
– APA102 (RGB), WS2812B (RGB), SK6812 (RGB, WWA) Neopixel/Pixelblade support
– 1/2/3-color LED stars (Tri-Cree LED modules) support
– Easy and free firmware updates by user via micro-SD card

Demonstration videos:

Link to the demonstration video by KR-sabers
Link to the demonstration video by ShtokCustomWorx

Video tutorials by KR-sabers:

Video tutorials by KR-sabers on YouTube

LED Resistor Calculator

1

HELPFUL LINKS

WHERE TO BUY

Parts links:

VERSO sound boards vendors

- KR-sabers (TheSaberArmory)

Neopixel strips

- Individual Neopixel LEDs
- Neopixel strips/connectors/other supplies (UK) – TheSaberArmory
- Neopixel strips/connectors/other supplies (USA) – TheCustomSaberShop

LEDs and wires

- Tri-Cree high power LEDs (USA) – TheCustomSaberShop
- Tri-Cree high power LEDs (USA) – SaberBay on Etsy
- Tri-Cree high power LEDs (UK) – TheSaberArmory
- Various Accent LEDs (UK) – TheSaberArmory
- PTFE Wires (UK) – TheSaberArmory

Batteries

- Various Batteries (UK) – TheSaberArmory
- Protected KeepPower 18650 15A 3120mAh battery (USA) – SaberBay on Etsy
- Protected KeepPower 18650 7A 3600mAh battery
- Protected KeepPower 18650 10A 3500mAh battery
- Protected KeepPower 18650 15A 3120mAh battery
- Unprotected Vapcell 21700 15A 5000mAh battery – requires external PCM
- Unprotected KeepPower 26650 15A 6000mAh battery – requires external PCM
- 15A Protection Circuit Module (PCM) (aliexpress)
- 18650 Protected Battery holder
- 18650 Protected Battery holder (USA) – SaberBay on Etsy

Recharge ports and Kill Switches

- High Power 1.3mm Recharge Port
- Recharge Ports (UK) – TheSaberArmory
- 2.1mm Recharge port (USA) – SaberBay on Etsy
- High Power Kill Switch (USA) – SaberBay on Etsy
- High Power Kill Switch
- Various Switches (UK) – TheSaberArmory

Blade connectors

- SCW NPXL blade connector (Russia) – ShtokCustomWorx
- SCW NPXL blade connector (USA) – SaberBay on Etsy
- SCW NPXL blade connector (UK) – TheSaberArmory
- TCSS Neopixel blade Pogo connector
- GX16 Neopixel/string blade connectors

Speakers

- Various speakers (UK) – TheSaberArmory
- Various speakers (USA) – TheCustomSaberShop
- 3W speakers

3D-printed chassis links:

- KR-sabers (TheSaberArmory) store
- ShtokCustomWorx on Shapeways
- GOTH-3Designs on Shapeways

VERSO INSTRUCTIONS

HOW TO USE

How to operate the saber with 1 button:

Ignition – Short press whilst OFF
Retraction – Long hold whilst ON
Blaster FX – Short press whilst ON
Lockup FX – Clash then long hold whilst ON
Enter Configuration Menu – Double press whilst ON - Initially, you will enter the first of 3 selections (sub-menus). These 3 'sub-menus' allow you to set the Blade Colour (1), Blade Profile (2) and Sound Font (3). By default, the selections can be navigated by rotating the saber on its axis - like a motorbike throttle. Once you have rotated the saber to your desired selection, follow the 'Confirm Menu' operation below. This will take you to the next sub-menu (Blade Profile), for which you can repeat the aforementioned to make your selection. Repeat for the third sub-menu (Sound Font) and after this confirmation, you will exit the configuration menu.
Confirm Menu – Long press whilst in Menu
Change Preset – Double press whilst blade is OFF
Mute Mode – Long hold for 3 seconds whilst OFF (you will hear a sound (*mute.wav*) to indicate the saber is now in mute mode. Pressing the activation button again will ignite the saber as normal but without any sound. Whilst in mute mode, with the blade off, hold the main activation button for 3 seconds to exit mute mode. You will hear a sound (*unmute.wav*) to indicate the saber has exited mute mode)

How to operate the saber with 2 buttons:

Button #1 – Activation (Act.)
Button #2 – Auxiliary (Aux.)

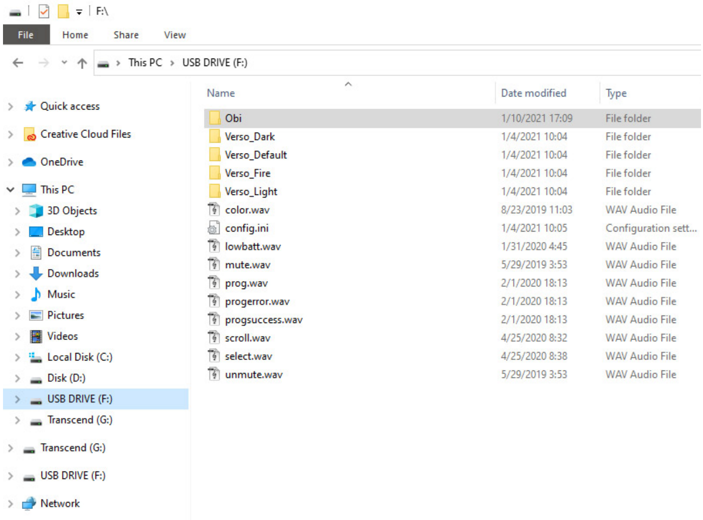
Ignition – Short press Button #1 whilst OFF
Retraction – Long hold Button #1 whilst ON
Blaster FX – Short press Button #2 whilst ON
Lockup FX – Hold Button #2 whilst ON
Enter Configuration Menu – Short press Button #1 and #2 whilst ON - Initially, you will enter the first of 3 selections (sub-menus). These 3 'sub-menus' allow you to set the Blade Colour (1), Blade Profile (2) and Sound Font (3). By default, the selections can be navigated by rotating the saber on its axis - like a motorbike throttle. Once you have rotated the saber to your desired selection, follow the 'Confirm Menu' operation below. This will take you to the next sub-menu (Blade Profile), for which you can repeat the aforementioned to make your selection. Repeat for the third sub-menu (Sound Font) and after this confirmation, you will exit the configuration menu.
Confirm Menu – Short press Button #2 whilst in Menu
Change Preset – Short press Button #2 whilst blade is OFF
Mute Mode – Long hold Button #1 whilst OFF (you will hear a sound (*mute.wav*) to indicate the saber is now in mute mode. Pressing the activation button again will ignite the saber as normal but without any sound. Whilst in mute mode, with the blade off, hold the main activation button for 3 seconds to exit mute mode. You will hear a sound (*unmute.wav*) to indicate the saber has exited mute mode)

How to add sound fonts

YouTube Tutorial: https://youtu.be/jcS4w2eSATE

1) Remove micro-SD card from the VERSO board SD socket and insert it into micro-SD card reader connected to your PC.

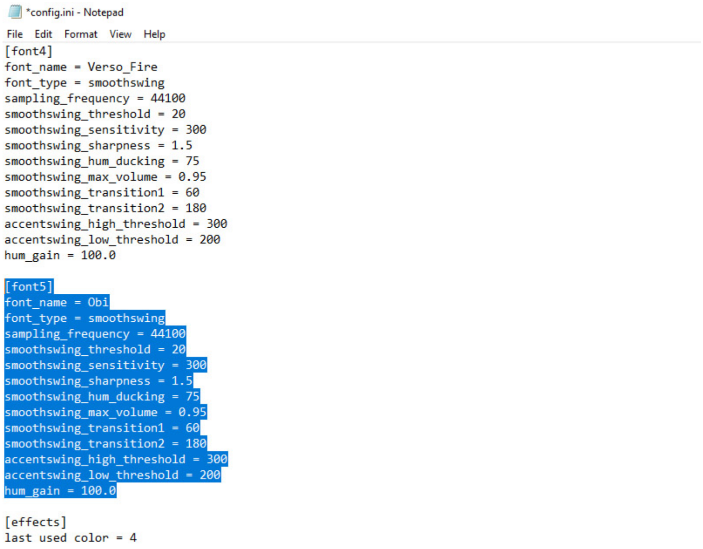
2) Copy and paste the new sound font folder (which includes all sound files formatted for VERSO board) into the micro-SD card root directory (the first directory where you see all other sound font folders). Sound font folder name can be any.



Name	Date modified	Type
Obi	1/10/2021 17:09	File folder
Verso_Dark	1/4/2021 10:04	File folder
Verso_Default	1/4/2021 10:04	File folder
Verso_Fire	1/4/2021 10:04	File folder
Verso_Light	1/4/2021 10:04	File folder
color.wav	8/23/2019 11:03	WAV Audio File
config.ini	1/4/2021 10:05	Configuration sett...
lowbatt.wav	1/31/2020 4:45	WAV Audio File
mute.wav	5/29/2019 3:53	WAV Audio File
prog.wav	2/1/2020 18:13	WAV Audio File
progerror.wav	2/1/2020 18:13	WAV Audio File
progsuccess.wav	2/1/2020 18:13	WAV Audio File
scroll.wav	4/25/2020 8:32	WAV Audio File
select.wav	4/25/2020 8:38	WAV Audio File
unmute.wav	5/29/2019 3:53	WAV Audio File

3) Open *config.ini* file (double-click), scroll down until you see *[font4]* block, select and copy the whole [font4] block and paste it after the *[font4]* block. Rename to *[font5]* and put the new name here: *font_name = Obi* (use the exact name of your new sound font folder). Save the *config.ini* file and close it.

Now you've added a new sound font. You can create maximum 16 font blocks, but you can have as many sound font folders on micro-SD card as you want.



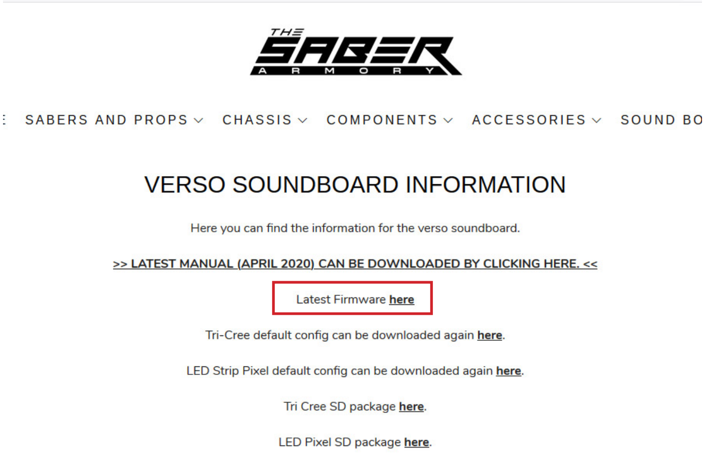
```
[font4]
font_name = Verso_Fire
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0

[font5]
font_name = Obi
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0

[effects]
last used color = 4
```

How to update firmware

1) Download the latest firmware from thesaberarmory.com website.



VERSO SOUNDBOARD INFORMATION

Here you can find the information for the verso soundboard.

>> LATEST MANUAL (APRIL 2020) CAN BE DOWNLOADED BY CLICKING HERE. <<

Latest Firmware **here**

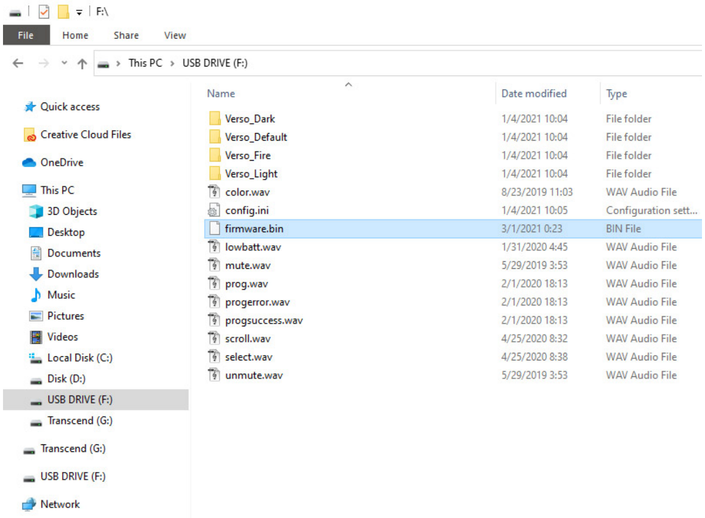
Tri-Cree default config can be downloaded again **here**.

LED Strip Pixel default config can be downloaded again **here**.

Tri Cree SD package **here**.

LED Pixel SD package **here**.

2) Remove the battery from your saber or turn the Kill Switch OFF. Remove micro-SD card from the VERSO board SD socket and insert it into micro-SD card reader connected to your PC.
Copy and paste the downloaded *firmware.bin* file into micro-SD card root directory (the first directory where you see all sound font folders).

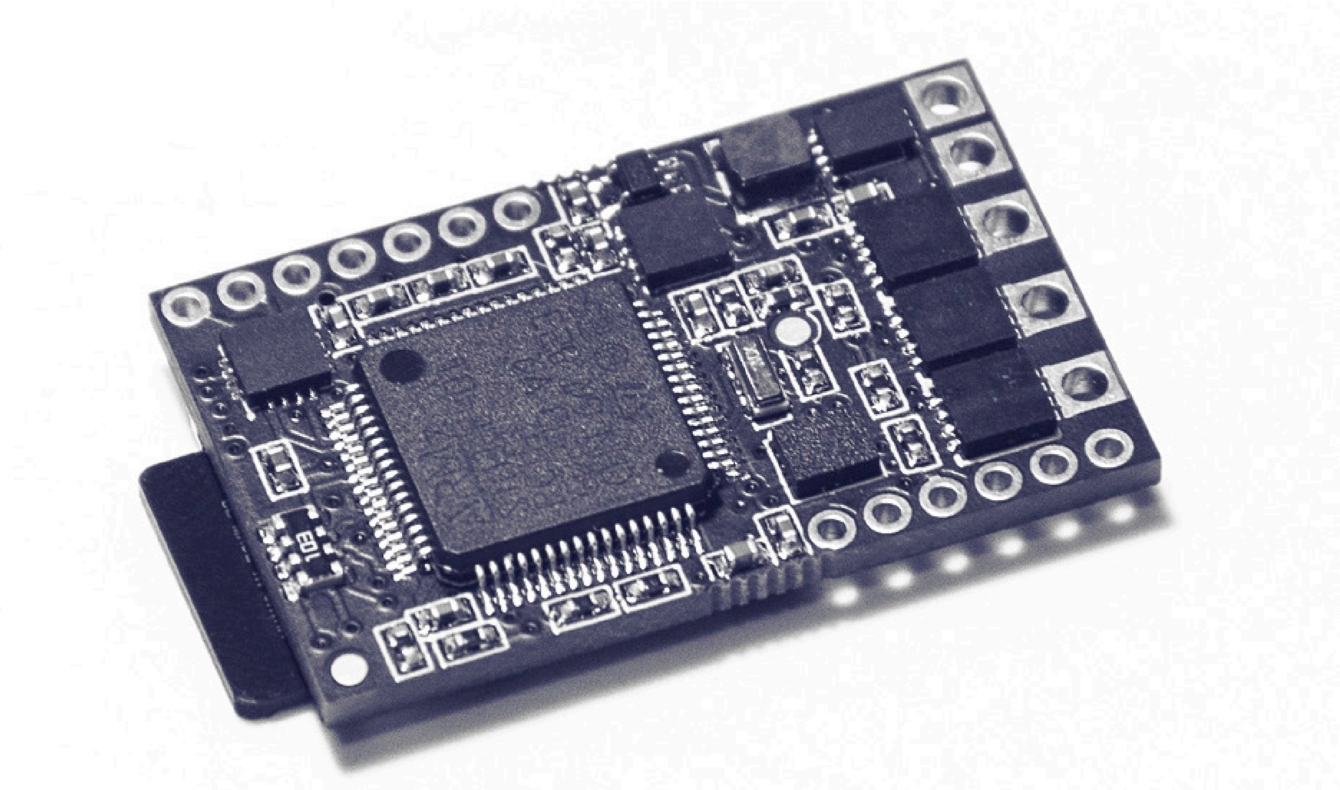


Name	Date modified	Type
Verso_Dark	1/4/2021 10:04	File folder
Verso_Default	1/4/2021 10:04	File folder
Verso_Fire	1/4/2021 10:04	File folder
Verso_Light	1/4/2021 10:04	File folder
color.wav	8/23/2019 11:03	WAV Audio File
config.ini	1/4/2021 10:05	Configuration sett...
firmware.bin	3/1/2021 0:23	BIN File
lowbatt.wav	1/31/2020 4:45	WAV Audio File
mute.wav	5/29/2019 3:53	WAV Audio File
prog.wav	2/1/2020 18:13	WAV Audio File
progerror.wav	2/1/2020 18:13	WAV Audio File
progsuccess.wav	2/1/2020 18:13	WAV Audio File
scroll.wav	4/25/2020 8:32	WAV Audio File
select.wav	4/25/2020 8:38	WAV Audio File
unmute.wav	5/29/2019 3:53	WAV Audio File

3) Remove micro-SD card from the card reader and insert it into the VERSO board. Reboot the board (insert the battery or turn the Kill Switch ON), you will hear firmware update confirmation "*prog.wav*" sound and then a completed update confirmation sound: "*progsuccess.wav*" file or "*progerror.wav*" if there was an error.

Firmware is now updated and the *firmware.bin* file is automatically deleted from the micro-SD card.

Tools and materials

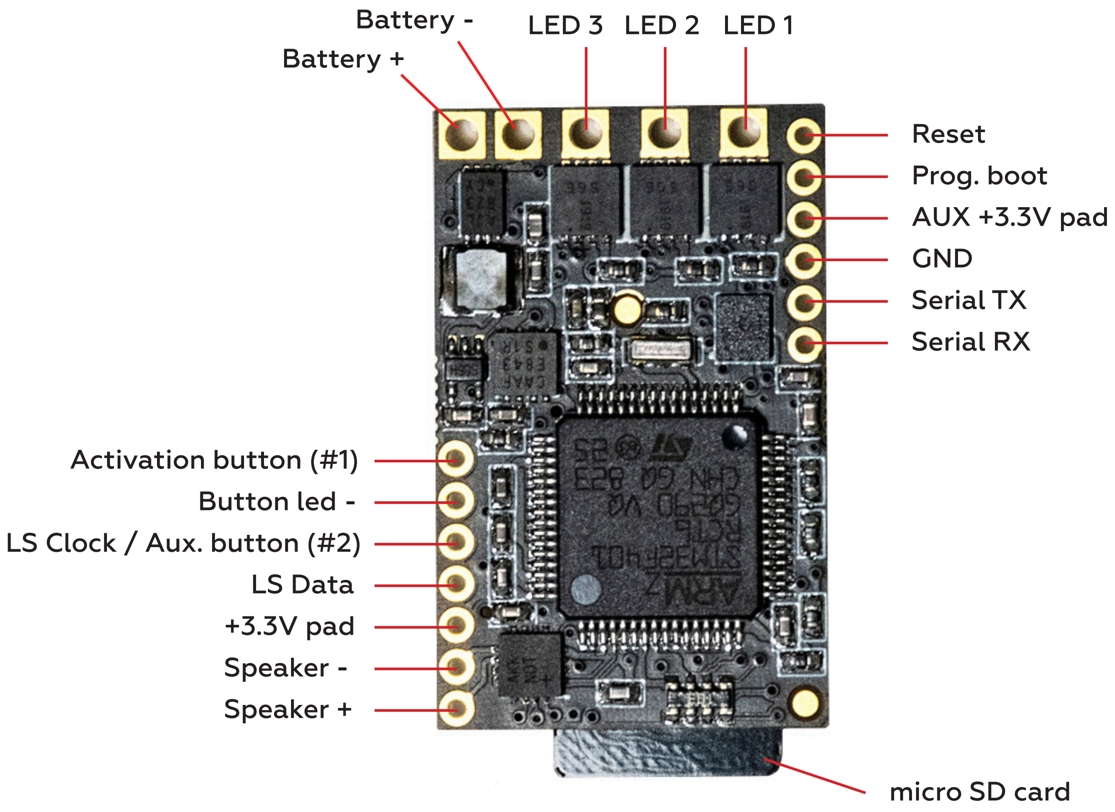


- – VERSO sound board
- – micro SD-card
- – a USB micro SD-card reader (to load sound files from PC to micro SD card)
- – wires of various gauges (32-20 AWG) (PTFE coated copper stranded wires recommended), heat shrink
- – ESD safe soldering station, solder wire, flux etc..
- – pliers, helping hands etc..
- – isopropyl alcohol to clean pads after soldering to make sure no shorts will occur and make the board clean
- – Digital Multimeter (VERY USEFUL!)
- – computer running Windows, Linux or Mac OS with internet access
- – 3.7V Li-Ion Protected rechargeable battery, switches, recharge port, speaker, LEDs, resistors, chassis etc..
- – Smart Li-Ion CC-CV (Constant Current - Constant Voltage mode) battery charger for 3.7V (4.2V) cells

VERSO INSTRUCTIONS

GENERAL INFO

Board pinout



Battery + – battery Positive: from 3 to 5 volt input

Battery - – battery Negative

LED 1, 2, 3 – hooks up to negative side of high power LED (positive side of LED hooks up directly to battery). These pads can handle up to 10 amps each. LED 1 = **Red**, LED 2 = **Green**, LED 3 = **Blue**

Speaker +/- – hooks up to speaker (4 Ohm or 8 Ohm 2W-5W)

+3.3V pad – generated by the board for powering satellite devices like accent leds etc.

LS Data – neopixel strip/led Data signal

LS Clock / Aux. button (#2) – neopixel strip/led Clock signal if required (some neopixel strips/leds have a Clock input, see datasheets), or a second button #2 (Auxiliary)

Button led - – Negative side for the Activation button led return (also can be used for accent led outside the button), controls accent led effects, turns OFF in Deep Sleep

Activation button (#1) – hooks up to a momentary button #1 (polarity doesn't matter)

Reset – used for board programming *(not for user)*

Prog. boot – used for board programming *(not for user)*

AUX +3.3V pad – used for board programming *(not for user)*

GND – ground for satellite devices except high power LEDs

Serial RX, TX – used for board programming *(not for user)*

EDITING PARAMETERS

Configuration file structure

```
[settings]
blade_type = pixel
strip_type = ws2812
pixel_count = 132
crossguard_pixel_count = 0
crossguard_delay = 0
stand_by_after = 30
mute_mode_after = 1000
retraction_after = 500
double_click_time = 250
blaster_after = 200
master_volume = 1.6
button_blink = 1000,1
button_count = 2
lowbatt_thrs = 3100
lowbatt_timebelow = 10
lowbatt_timeabove = 10
lowbatt_interval = 10
emergency_voltage = 3000
emergenct_timeout = 300
clash_sensitivity = 300
selection_wheel = 1,1,1
selection_axis = 1,2
```

General settings block

```
[font1]
font_name = Verso_Dark
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0

[font2]
font_name = Verso_Default
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0

[font3]
font_name = Verso_Light
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0

[font4]
font_name = Verso_Fire
font_type = smoothswing
sampling_frequency = 44100
smoothswing_threshold = 20
smoothswing_sensitivity = 300
smoothswing_sharpness = 1.5
smoothswing_hum_ducking = 75
smoothswing_max_volume = 0.95
smoothswing_transition1 = 60
smoothswing_transition2 = 180
accentswing_high_threshold = 300
accentswing_low_threshold = 200
hum_gain = 100.0
```

Sound fonts blocks
(can create up to 16 fonts blocks)

Copy the whole block, paste after the last one and rename it to the next following number like [font5] etc...

Sound font #4

```
[effects]
last_used_color = 4
last_used_profile = 4
last_used_font = 0
last_used_preset = 0
ignition_time = 250
retraction_time = 350
transition_time = 250
```

Effects block

```
preset1 = 1,1,1
preset2 = 2,2,2
preset3 = 3,3,3
preset4 = 4,4,4
```

User defined presets block
(can create up to 16 presets)

```
preset1 = ...
preset2 = ...
preset3 = ...
preset4 = ...
preset5 = ...
preset6 = ...
preset7 = ...
preset8 = ...
preset9 = ...
preset10 = ...
preset11 = ...
preset12 = ...
preset13 = ...
preset14 = ...
preset15 = ...
preset16 = ...
```

```
blade_color1 = 255,0,0
clash_color1 = 255,255,255
lockup_color1 = 255,255,255
blaster_color1 = 255,255,255

blade_color2 = 0,255,0
clash_color2 = 255,255,255
lockup_color2 = 255,255,255
blaster_color2 = 255,255,2555

blade_color3 = 0,0,255
clash_color3 = 255,255,255
lockup_color3 = 255,255,255
blaster_color3 = 255,255,255

blade_color4 = 255,20,147
clash_color4 = 255,255,255
lockup_color4 = 255,255,255
blaster_color4 = 255,255,255

blade_color5 = 255,0,255
clash_color5 = 255,255,255
lockup_color5 = 255,255,255
blaster_color5 = 255,255,255

blade_color6 = 255,69,0
clash_color6 = 255,255,255
lockup_color6 = 255,255,255
blaster_color6 = 255,255,255

blade_color7 = 0,255,255
clash_color7 = 255,255,255
lockup_color7 = 255,255,255
blaster_color7 = 255,255,255

blade_color8 = 255,255,0
clash_color8 = 255,255,255
lockup_color8 = 255,255,255
blaster_color8 = 255,255,255

blade_color9 = 255,255,255
clash_color9 = 255,255,255
lockup_color9 = 255,255,255
blaster_color9 = 255,255,255
```

Blade colors blocks
(can create up to 16 color blocks)

Copy the whole block, paste after the last one and rename each line to the next following number like:

```
blade_color10 = 255,255,255
clash_color10 = 255,255,255
lockup_color10 = 255,255,255
blaster_color10 = 255,255,255
```

etc...

Color #9

```
flicker_type1 = random
flicker_range1 = 100,100
flicker_frequency1 = 1
clash_type1 = flash
clash_duration1 = 25
clash_frequency1 = 100
lockup_type1 = flash
lockup_frequency1 = 100
lockup_percent1 = 100
lockup_centre1 = 100
blaster_type1 = flash
blaster_duration1 = 30
blaster_frequency1 = 100
blaster_percent1 = 100
blaster_centre1 = 100
blaster_delay1 = 100

flicker_type2 = random
flicker_range2 = 90,100
flicker_frequency2 = 20
clash_type2 = flash
clash_duration2 = 25
clash_frequency2 = 100
lockup_type2 = flash
lockup_frequency2 = 100
lockup_percent2 = 100
lockup_centre2 = 100
blaster_type2 = flash
blaster_duration2 = 30
blaster_frequency2 = 100
blaster_percent2 = 100
blaster_centre2 = 100
blaster_delay2 = 75

flicker_type3 = random
flicker_range3 = 80,100
flicker_frequency3 = 30
clash_type3 = flash
clash_duration3 = 25
clash_frequency3 = 100
lockup_type3 = flash
lockup_frequency3 = 100
lockup_percent3 = 100
lockup_centre3 = 100
blaster_type3 = flash
blaster_duration3 = 30
blaster_frequency3 = 100
blaster_percent3 = 100
blaster_centre3 = 100
blaster_delay3 = 50

flicker_type4 = fire
flicker_range4 = 10,50
flicker_frequency4 = 30
clash_type4 = flash
clash_duration4 = 400
clash_frequency4 = 10
lockup_type4 = static
lockup_frequency4 = 40
lockup_percent4 = 0
lockup_centre4 = 0
blaster_type4 = static
blaster_duration4 = 100
blaster_frequency4 = 2
blaster_percent4 = 0
blaster_centre4 = 0
blaster_delay4 = 50
```

Blade profile blocks
(can create up to 16 blade profiles blocks)

Copy the whole block, paste after the last one and rename each line to the next following number like:

```
flicker_type5 = fire
flicker_range5 = 10,50
flicker_frequency5 = 30
clash_type5 = flash
clash_duration5 = 400
clash_frequency5 = 10
lockup_type5 = static
lockup_frequency5 = 40
lockup_percent5 = 0
lockup_centre5 = 0
blaster_type5 = static
blaster_duration5 = 100
blaster_frequency5 = 2
blaster_percent5 = 0
blaster_centre5 = 0
blaster_delay5 = 50
```

etc...

Blade profile #4

Configuration file

The configuration file is an INI text file (config.ini) placed at the root directory of the micro-SD card. It is subdivided into six sections:

- **Settings**
- **Fonts**
- **Effects**
- **Presets**
- **Blade Colours**
- **Blade Profiles (effects)**

Refer to the config.ini file on micro-SD card as an example.

Settings Section

blade_type – can have one of the following values:

hbled – for high-brightness LEDs
pixel – for addressable LED strips

strip_type – is valid only if *blade_type = pixel* and can have one of the following values:

ws2812 – for WS2812 and SK6812 led (pixel) strips
apa102 – for APA102C led (pixel) strips

pixel_count – is valid only if *blade_type = pixel* and is the number of leds on ONE SIDE of the strip, for example:

```
pixel_count = 144
```

hbledN – this configures each LED output being *N* the number of the LED: 1, 2 or 3, where 1 = RED, 2 = GREEN, 3 = BLUE. It is a series of values defined like:

```
hbledN = <maximum current>, <maximum voltage>, <resistor>, <current point 1>, <voltage point 1>
```

<maximum current> – is the maximum current the LED can withstand (in amperes)
<maximum voltage> – is the voltage at maximum current (in volts)
<resistor> – is the resistor connected to the LED (in Ohms)
<current point 1> – is a current value in amperes, lower than maximum current (usually 0.35A)
<voltage point 1> – is the voltage of the LED at current point 1 (LED voltage at 0.35A)

<current point 1> and *<voltage point 1>* are values used to estimate the current of the LED when the input voltage is higher than maximum voltage. For example, for an LED connected to **LED output 1**, with maximum current = **1**A, maximum voltage = **2.7**V, a **1** Ohm resistor, current point 1 = **0.35**A and voltage point 1 = **2.2**V, the value should be:

```
hbled1 = 1, 2.7, 1, 0.35, 2.2
```

NOTE:

if **hbled1**, **hbled2** or **hbled3** values are missing in the configuration file, then the following values are taken by default (for a Cree XPE2 RGB LED):
hbled1 = 1, 2.65, 1, 0.35, 2.2 – (RED LED)
hbled2 = 1, 3.7, 0, 0.35, 3.2 – (GREEN LED)
hbled3 = 1, 3.4, 0.33, 0.35, 3.1 – (BLUE LED)
The **RED LED** has a 1 Ohm resistor, the **GREEN LED** has no resistor and the **BLUE LED** has a 0.33 Ohm resistor.

crossguard_pixel_count – is valid only if *blade_type = pixel* and is the number of LEDs in each secondary LED strip. Should be 0 if no secondary strip is installed.

crossguard_delay – is valid only if *blade_type = pixel* and is the delay between the end of the ignition/retraction of the main blade and the start of the ignition/retraction of the crossguard blades. A higher number means a longer delay. A value of 100 produces a delay equivalent to 100 pixels ingiting/retracting between the main and crossguard pixels.

EDITING PARAMETERS

Tri-Cree LED	LED 1	LED 2	LED 3
Red/Green/Blue	hbled1 = 1, 2.7, 1, 0.35, 2.2	hbled2 = 1, 3.7, 0, 0.35, 3.2	hbled3 = 1, 3.4, 0.33, 0.35, 3.1
Red/Green/RoyalBlue	hbled1 = 1, 2.7, 1, 0.35, 2.2	hbled2 = 1, 3.7, 0, 0.35, 3.2	hbled3 = 1, 3.4, 0.33, 0.35, 3.1
Red/Red/White	hbled1 = 1, 2.7, 1, 0.35, 2.2	hbled2 = 1, 2.7, 1, 0.35, 2.2	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
Green/Green/White	hbled1 = 1, 3.7, 0, 0.35, 3.2	hbled2 = 1, 3.7, 0, 0.35, 3.2	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
Blue/Blue/White	hbled3 = 1, 3.4, 0.33, 0.35, 3.1	hbled3 = 1, 3.4, 0.33, 0.35, 3.1	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
Amber/Amber/White	hbled1 = 1, 3.3, 0.25, 0.35, 3.05	hbled2 = 1, 3.3, 0.25, 0.35, 3.05	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
DeepRed/DeepRed/White	hbled1 = 1, 2.5, 1.2, 0.35, 2.05	hbled2 = 1, 2.5, 1.2, 0.35, 2.05	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
RedOrange/RedOrange/White	hbled1 = 1, 2.7, 1, 0.35, 2.2	hbled2 = 1, 2.7, 1, 0.35, 2.2	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
Cyan/Cyan/White	hbled1 = 1, 3.4, 0.33, 0.35, 3.1	hbled2 = 1, 3.4, 0.33, 0.35, 3.1	hbled3 = 1, 3.15, 0.5, 0.35, 2.9
Other color combinations also possible, use values from examples above.			

Luxeon Tri-Rebel LED	LED 1	LED 2	LED 3
Red/Green/Blue	hbled1 = 0.7, 2.4, 2, 0.35, 2.1	hbled2 = 1, 3.2, 0.4, 0.35, 2.9	hbled3 = 1, 3.05, 0.7, 0.35, 2.8
Red/Green/RoyalBlue	hbled1 = 0.7, 2.4, 2, 0.35, 2.1	hbled2 = 1, 3.2, 0.4, 0.35, 2.9	hbled3 = 1, 3.05, 0.7, 0.35, 2.8
Red/Red/White	hbled1 = 0.7, 2.4, 2, 0.35, 2.1	hbled2 = 0.7, 2.4, 2, 0.35, 2.1	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
Green/Green/White	hbled1 = 1, 3.2, 0.4, 0.35, 2.9	hbled2 = 1, 3.2, 0.4, 0.35, 2.9	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
Blue/Blue/White	hbled1 = 1, 3.05, 0.7, 0.35, 2.8	hbled2 = 1, 3.05, 0.7, 0.35, 2.8	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
Amber/Amber/White	hbled1 = 0.7, 3.2, 1, 0.35, 3.0	hbled2 = 0.7, 3.2, 1, 0.35, 3.0	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
DeepRed/DeepRed/White	hbled1 = 0.7, 2.4, 2, 0.35, 2.1	hbled2 = 0.7, 2.4, 2, 0.35, 2.1	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
RedOrange/RedOrange/White	hbled1 = 0.7, 2.4, 2, 0.35, 2.1	hbled2 = 0.7, 2.4, 2, 0.35, 2.1	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
Cyan/Cyan/White	hbled1 = 1, 3.2, 0.4, 0.35, 2.9	hbled2 = 1, 3.2, 0.4, 0.35, 2.9	hbled3 = 1, 3.1, 0.5, 0.35, 2.8
Other color combinations also possible, use values from examples above.			

stand_by_after – is a value in seconds of the time to wait without user interaction to enter into stand-by (low power) mode.
mute_mode_after – is a value in milliseconds and represents the time the user has to keep the button pressed to enter and exit the mute mode.
retraction_after – is a value in milliseconds and represents the time the user has to keep the button pressed to produce a retraction.
double_click_time – is a value in milliseconds and represents the time in which two consecutive button clicks are considered a double click (for changing the color of the blade).
blaster_after – defines a value in milliseconds and represents the time the user has to keep the button pressed to produce a blaster effect.
master_volume – defines the master output volume. Is a value between 0 and 2.0 (with decimal point), being 1.0 the default volume without any extra gain.
button_blink – describes the button LED blinking with two values (in milliseconds): the time the LED will stay ON, and the time the LED will stay OFF.
For example, for a sequence of a LED staying ON for 250ms and staying OFF for 750ms, the values will be:

```
button_blink = 250, 750
```

The sum of the two values represent the frequency and duty cycle of the blinking sequence (1 second frequency in the above case, with 25% duty cycle).
If you would prefer no blink, configure to 1000,1.

button_count – the number of buttons installed (1 or 2)
lowbatt_thrs – indicates the low-battery threshold in millivolts. If the battery voltage is lower than this value for the time defined with the *lowbatt_timebelow* parameter, then a low battery condition is declared. If this value is 0, then the low battery detection is disabled.
lowbatt_timebelow – time in seconds with a battery voltage below *lowbatt_thrs* to declare a low battery condition.
lowbatt_timeabove – time in seconds with a battery voltage above *lowbatt_thrs* to exit the low battery condition. This is an optional parameter. Setting this value to zero will make the low battery condition permanent until the next board reset.
lowbatt_interval – WAV file interval time in seconds. If set to 0, the user indication WAV file is played a single time (when the low battery condition is detected).

EDITING PARAMETERS

emergency_voltage – voltage in millivolts, a value between 3000 and 3700. If this parameter is missing or ignored, it takes the default 3500. Please note certain components on the circuit board have a minimum operational voltage of 3V. So please be sure to not leave the *emergency_timeout* parameter for too long. It's default to 180 seconds.
emergency_timeout – time in seconds the voltage is below *emergency_voltage*. If ignored it takes a value of 10 seconds.
When the battery drops below *emergency_voltage* for the time indicated by *emergency_timeout*, then the board will enter automatically in low-power (deep sleep) mode. If the LEDs are ON (strip or HBLED) then a retraction effect is executed before entering in low-power.
clash_sensitivity – is a value between 100 and 500. If this value is 0 or the parameter is not present, it will be set automatically to 300.
Higher value - less sensitive.
selection_wheel – has three values. Each number represents: *blade colour* selection wheel, *blade profile* selection wheel and *font* selection wheel).
0 = disabled, 1 = enabled. This allows you to customise which menus you would like to have available when operating the saber.
selection_axis – two axes which define the selection wheel orientation. Each value can be *1*, *2*, or *3* representing *x*, *y* or *z* accelerometer directions.
For example: *selection_axis = 1,2* would use the *x* and *y* axis so you would cycle the menu like using a motorbike throttle.

Fonts Section

(maximum 16 fonts)

[font#] – the section describing font number N. Each font should have its own section.
font_name – name of the font folder in the SD card.
font_type – describes the type of font. It can be:

> **monophonic** – for a monophonic font
> **polyphonic** – for a polyphonic font
> **smoothswing** – for a smoothswing font

sampling_frequency – is the sampling frequency of all the sounds in a given font. The only two possible values are:

> sampling_frequency = 22050
> sampling_frequency = 44100

smoothswing_threshold – is valid only when font_type = smoothswing, degrees/second needed to register as a swing.
smoothswing_sensitivity – is valid only when font_type = smoothswing, swing sensitivity, degrees of rotations per second required to reach full volume.
smoothswing_sharpness – is valid only when font_type = smoothswing, non-linear swing response, higher values makes it more non-linear.
smoothswing_hum_ducking – is valid only when font_type = smoothswing, how many percent the hum sound will decrease as swing increases.
smoothswing_max_volume – is valid only when font_type = smoothswing, swing volume multiplier defaults to 3x normal volume.
smoothswing_transition1 – is valid only when font_type = smoothswing, length of first transition in degrees.
smoothswing_transition2 – is valid only when font_type = smoothswing, length of second transition in degrees.
accentswing_high_threshold – is valid only when font_type = smoothswing, degrees/second needed to register an accent swing.
A higher value means a faster swing is need to start an accent swing.
accentswing_low_threshold – is valid only when font_type = smoothswing, degrees/second needed to end an accent swing.
A lower value means the accent swing will continue longer as the swing slows down.
hum_gain – is valid only when font_type = smoothswing, percentage to scale the hum sound by increments of 100. A higher value makes the hum louder. A lower value makes the hum quieter. hum_gain = 100.0 keeps the original volume.

Effects Section

last_used_color – do not change. This is set automatically. This allows the board to remember the last used colour before shutting off.
last_used_profile – do not change. This is set automatically. This allows the board to remember the last used profile before shutting off.
last_used_font – do not change. This is set automatically. This allows the board to remember the last used font before shutting off.
last_used_preset – do not change. This is set automatically. This allows the board to remember the last used preset before shutting off.
ignition_time – is the duration in milliseconds of the ignition effect.
retraction_time – is the duration in milliseconds of the retraction effect.
transition_time – is the duration in milliseconds of the transition between blade colors.

Presets Section

(maximum 16 presets)

preset# – number of preset combination of blade colour number (the first number), blade profile number (second number), and font number (third number).
These presets are selected when the blade is retracted.

3 EDITING PARAMETERS

Blade Colour Section

(maximum 16 colours)

blade_color# – this the color of the blade for blade colour N. It's an RGB value.
For example: a Red blade color of RED = 255, GREEN = 0 and BLUE = 0 would be:

```
blade_color# = 255,0,0
```

clash_color# – this the color of the clash for blade colour N. It's an RGB value.
For example: a White clash color of RED = 255, GREEN = 255 and BLUE = 255 would be:

```
clash_color# = 255,255,255
```

lockup_color# – this the color of the lockup for blade colour N. It's an RGB value.
For example: a White lockup color of RED = 255, GREEN = 255 and BLUE = 255 would be:

```
lockup_color# = 255,255,255
```

blaster_color# – this the color of the blaster deflection for blade colour N. It's an RGB value.
For example: a White blaster deflection color of RED = 255, GREEN = 255 and BLUE = 255 would be:

```
blaster_color# = 255,255,255
```

Blade Profile Section

(maximum 16 profiles)

flicker_type# – this is the visual effect on the blade. Valid settings are:

random – produces a flicker effect with changing brightness (see flicker_rangeN below). A traditional lightsaber style.
unstable – an unstable blade effect, like Kylo Ren blade.
fire – a flame effect.
rainbow – for a scrolling rainbow effect. RGB cycle.

flicker_range# – these are two values (in percent) that define the range of available brightness for a flicker effect. For example, for a flicker effect that cycles between 90% brightness up to 100% brightness and back (flicker_type = random), it would be:

```
flicker_range# = 90,100
```

This setting is ignored when *flicker_type# = fire, unstable*, or *rainbow*

flicker_frequency# – this the update frequency of the flicker effect. A higher value produces a faster flicker, a lower value produces a slower flicker.
This setting is ignored when *flicker_type# = fire, unstable*, or *rainbow*

clash_type# – is the type of clash effect. It can be:

static – for a static color with a duration defined by *clash_duration#*
flash – for an intermittent flashing effect with a duration defined by *clash_duration#* and frequency defined by *clash_frequency#*
This setting is ignored when *flicker_type# = fire*

clash_duration# – this is the duration in milliseconds of the clash effect. A higher number produces a longer effect.
clash_frequency# – this is the frequency in Hz of the clash effect. Can be maximum 25. A higher number produces a faster flashing.
This setting is ignored when *flicker_type# = fire*

lockup_type# – is the type of lockup effect. It can be:

static – for a static color
flash – for an intermittent flashing effect with frequency defined by *lockup_frequency#*
This setting is ignored when *flicker_type# = fire*

lockup_frequency# – this is the frequency in Hz of the lockup effect. Can be maximum 25. A higher number produces a faster flashing.
This setting is ignored when *flicker_type# = fire*

lockup_percent# – the percentage of the blade that lights up during the lockup effect. A higher value lights up more of the blade. *lockup_percentN = 0* – lights up a random amount of the blade each time.
This setting is ignored when *flicker_type# = fire*

lockup_centre# – the centre of the block of pixels that light up during the lockup effect, as a percentage of the blade length. A higher number centres the effect further from the hilt. *lockup_centreN = 75* and *lockup_percentN = 50* would light up the half of the blade furthest from the hilt. *lockup_centreN = 0* centres the effect in a random place each time.
This setting is ignored when *flicker_type# = fire*

blaster_type# – is the type of blaster deflection effect. It can be:

static – for a static color with a duration defined by *blaster_duration#*
flash – for an intermittent flashing effect with a duration defined by *blaster_duration#* and frequency defined by *blaster_frequency#*
This setting is ignored when *flicker_type# = fire*

blaster_duration# – this is the duration in milliseconds of the blaster deflection effect. A higher number produces a longer effect.
blaster_frequency# – this is the frequency in Hz of the blaster deflection effect. Can be maximum 25. A higher number produces a faster flashing.
This setting is ignored when *flicker_type# = fire*

blaster_percent# – the percentage of the blade that lights up during the blaster deflection effect. A higher value lights up more of the blade. *blaster_percent# = 0* lights up a random amount of the blade each time.
This setting is ignored when *flicker_type# = fire*

blaster_centre# – the centre of the block of pixels that light up during the blaster deflection effect, as a percentage of the blade length. A higher number centres the effect further from the hilt. *blaster_centreN = 75* and *blaster_percentN = 50* would light up the half of the blade furthest from the hilt. *blaster_centreN = 0* centres the effect in a random place each time.
This setting is ignored when *flicker_type# = fire*

blaster_delay# – the time in milliseconds between the blaster sound effect starting and the deflection sound effect and blaster deflection blade effect starting. A higher number produces a longer delay.

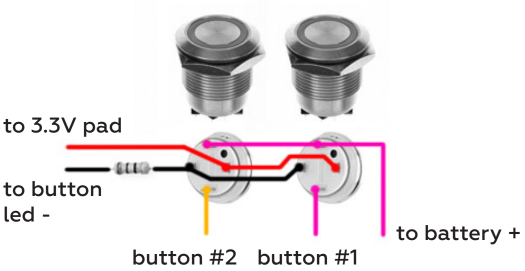
Sound files

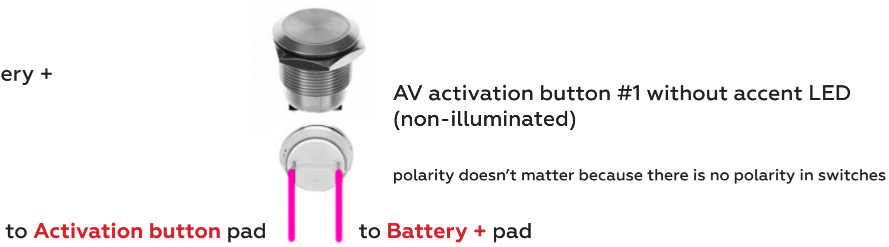
The sound files must be copied to the root directory of the micro-SD card. Each kind of sound must follow this naming convention:

- Boot sounds must be named **boot.wav** or **boot#.wav** (where # is a number) if there are multiple boot files.
- Clash sounds must be named **clash#.wav** (where # is a number) to define multiple clash sounds.
- Swing sounds for monophonic or polyphonic fonts must be named **swing#.wav** (where # is a number) to define multiple swing sounds.
- Swing sounds for smoothswing fonts must be named **swingh#.wav** and **swingl#.wav** (where # is a number).
- The sound file for the ignition must be named **on.wav**. If *on_count* is set to a value greater than 1, then the files must be named **on#.wav**, where # is a number between *1 and on_count*. All files in the specified range must be present in the SD card.
- The sound file for the retraction must be named **off.wav**. If *off_count* is set to a value greater than 1, then the files must be named **off#.wav**, where # is a number between *1 and off_count*. All files in the specified range must be present in the SD card.
- The sound file for the hum must be named **hum.wav**
- The sound file played when changing flicker colors (blade color profiles) must be named **color.wav**
- The sound file played when entering mute mode must be named **mute.wav**
- The sound file played when exiting mute mode must be named **unmute.wav**
- The sound file played for accent swings must be named **aswing#.wav**
- The sound file played for blasters shots must be named **blast#.wav**
- The sound file played for blaster deflection must be named **deflc#.wav**
- The sound file played for lockup must be named **lockup#.wav**
- The sound file played for after lockup sound must be named **endlock#.wav**
- The sound file played when changing a preset must be named **font.wav**
- The sound file played for low battery warning must be named **lowbatt.wav**
- The sound file played for firmware update progress must be named **prog.wav**
- The sound file played for firmware update error must be named **progerror.wav**
- The sound file played for successfully completed firmware update must be named **progsuccess.wav**
- The sound file played when scrolling through blade colours must be named **scroll.wav**
- The sound file played when selecting the blade colour, blade profile or font must be named **select.wav**

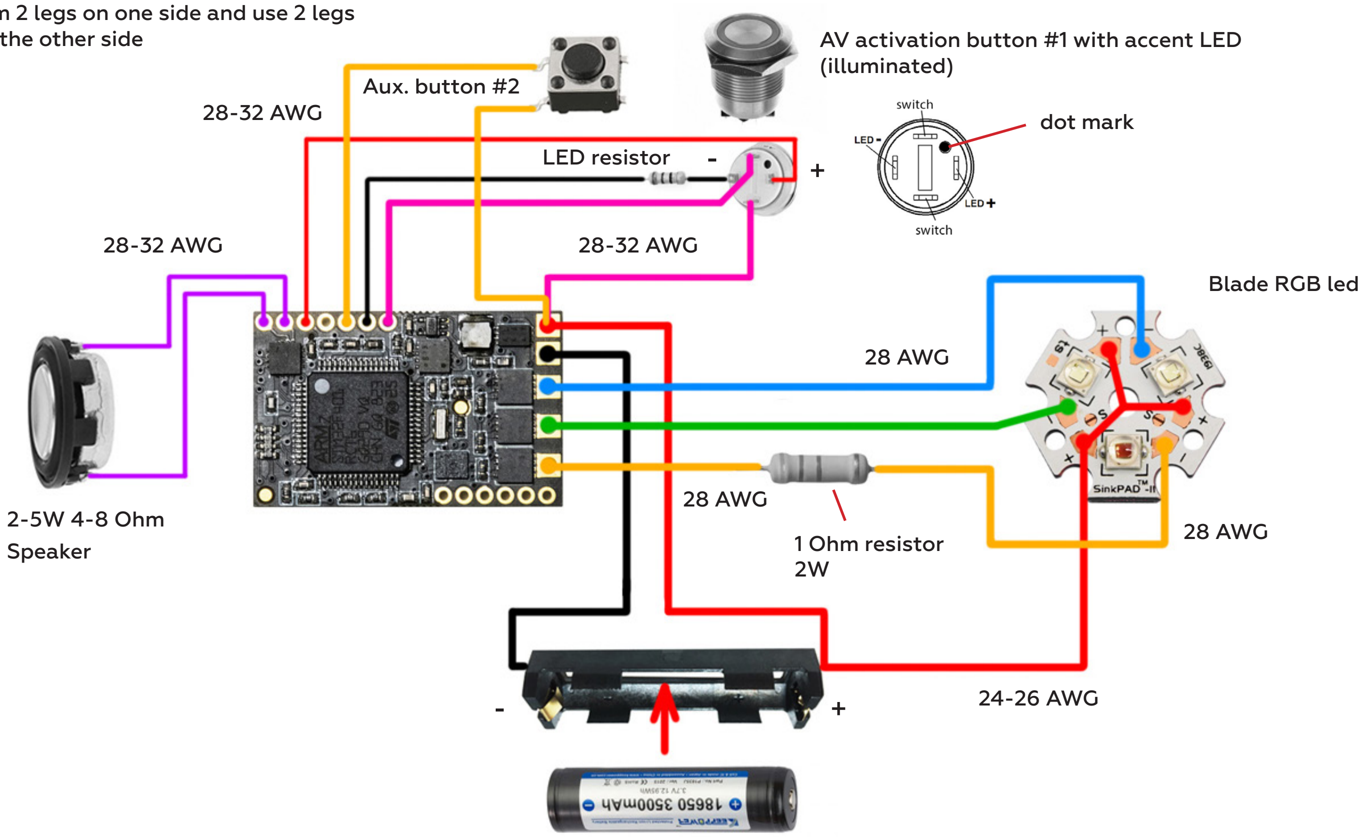
Beginner Tri-Cree wiring diagram (In-hilt LED)

in case 2 illuminated AV switches are used, wire them like this:



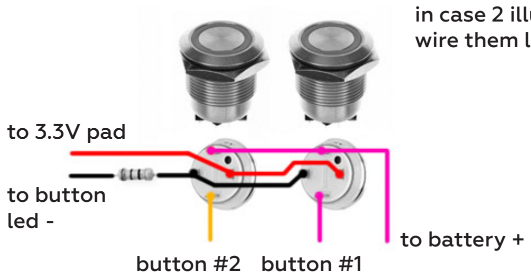

AV activation button #1 without accent LED (non-illuminated)
polarity doesn't matter because there is no polarity in switches
to Activation button pad
to Battery + pad

in case a momentary button with 4 legs is used, trim 2 legs on one side and use 2 legs on the other side

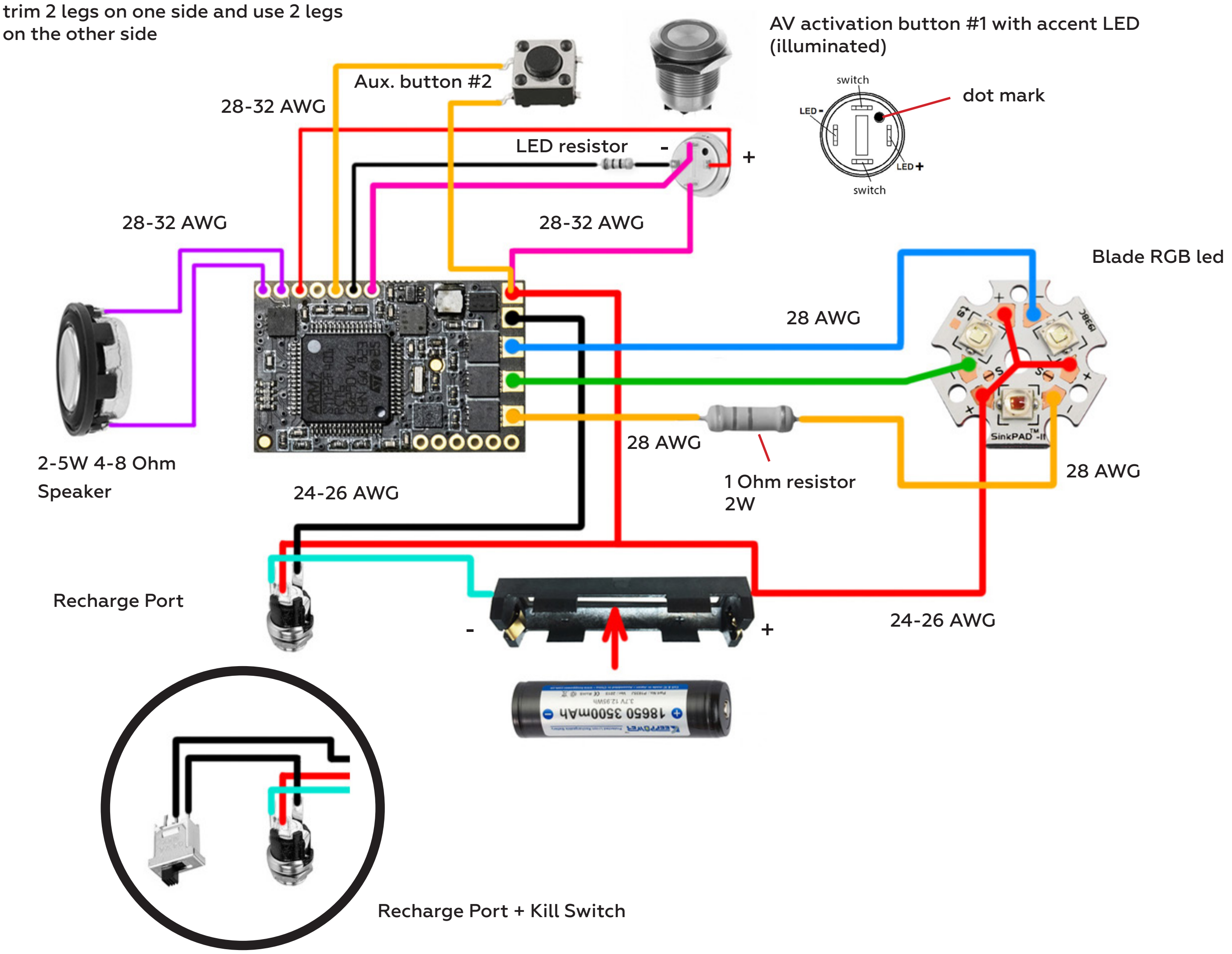


Basic Tri-Cree wiring diagram (In-hilt LED)

in case 2 illuminated AV switches are used, wire them like this:

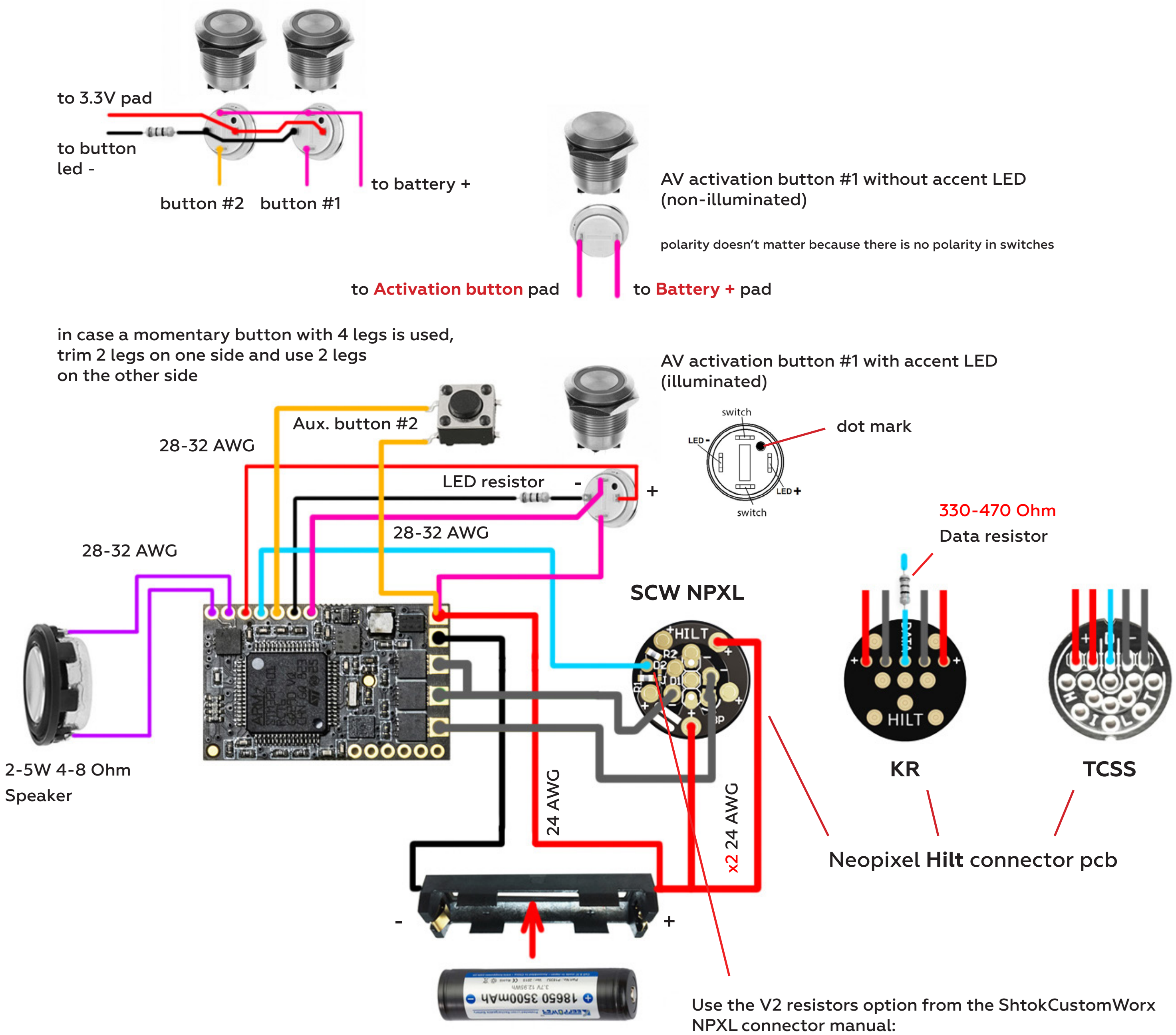


in case a momentary button with 4 legs is used, trim 2 legs on one side and use 2 legs on the other side



Beginner Neopixel/Pixelblade wiring diagram

in case 2 illuminated AV switches are used, wire them like this:



in case a momentary button with 4 legs is used, trim 2 legs on one side and use 2 legs on the other side

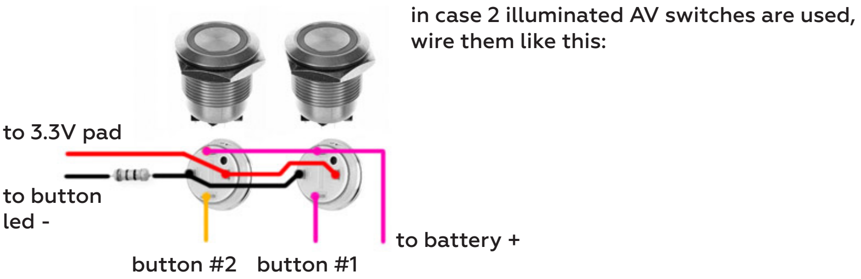
Use the V2 resistors option from the ShtokCustomWorx NPXL connector manual:

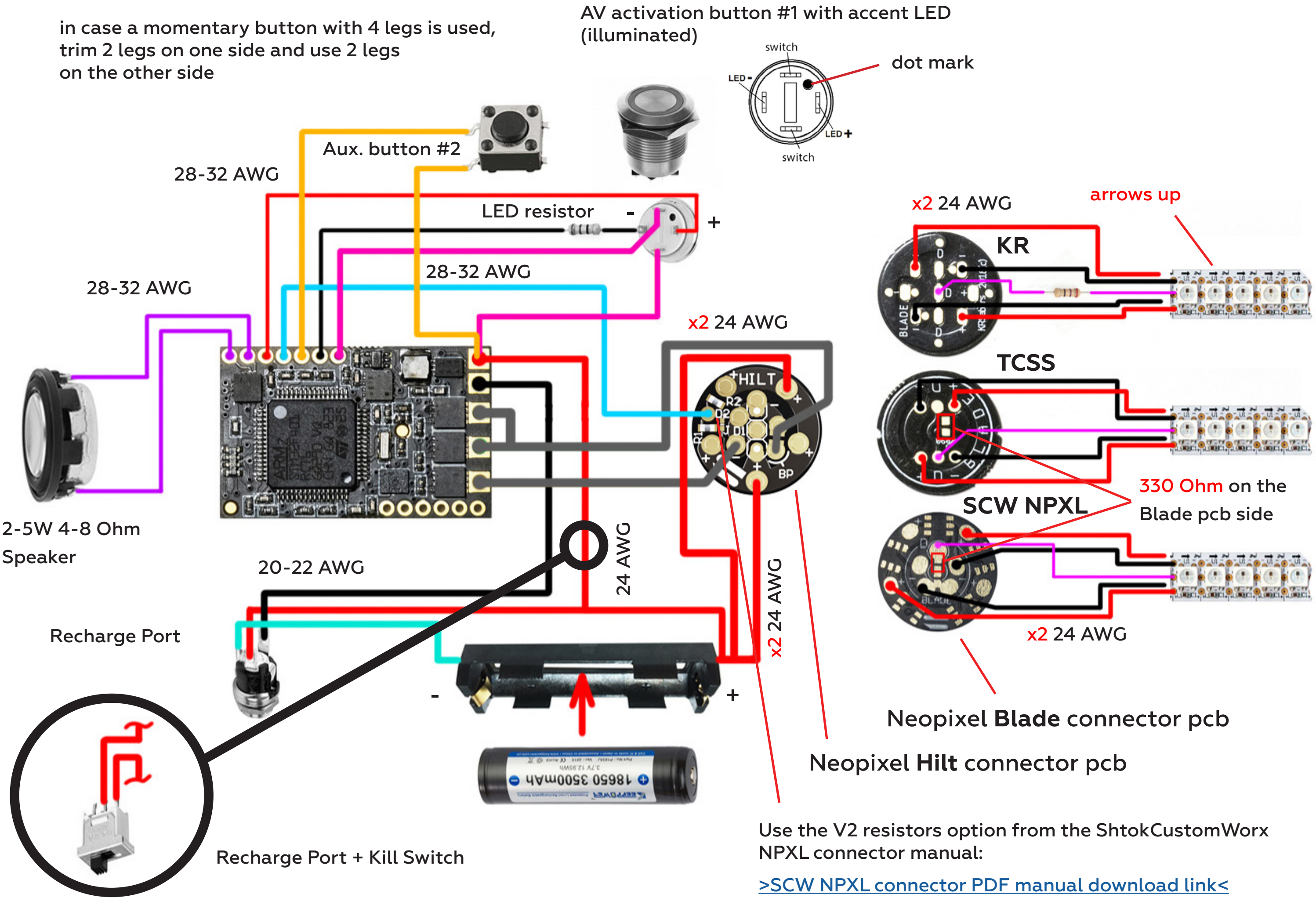
>SCW NPXL connector PDF manual download link<

VERSO INSTRUCTIONS

WIRING DIAGRAMS

Basic Neopixel/Pixelblade wiring diagram **with SCW NPXL connector**



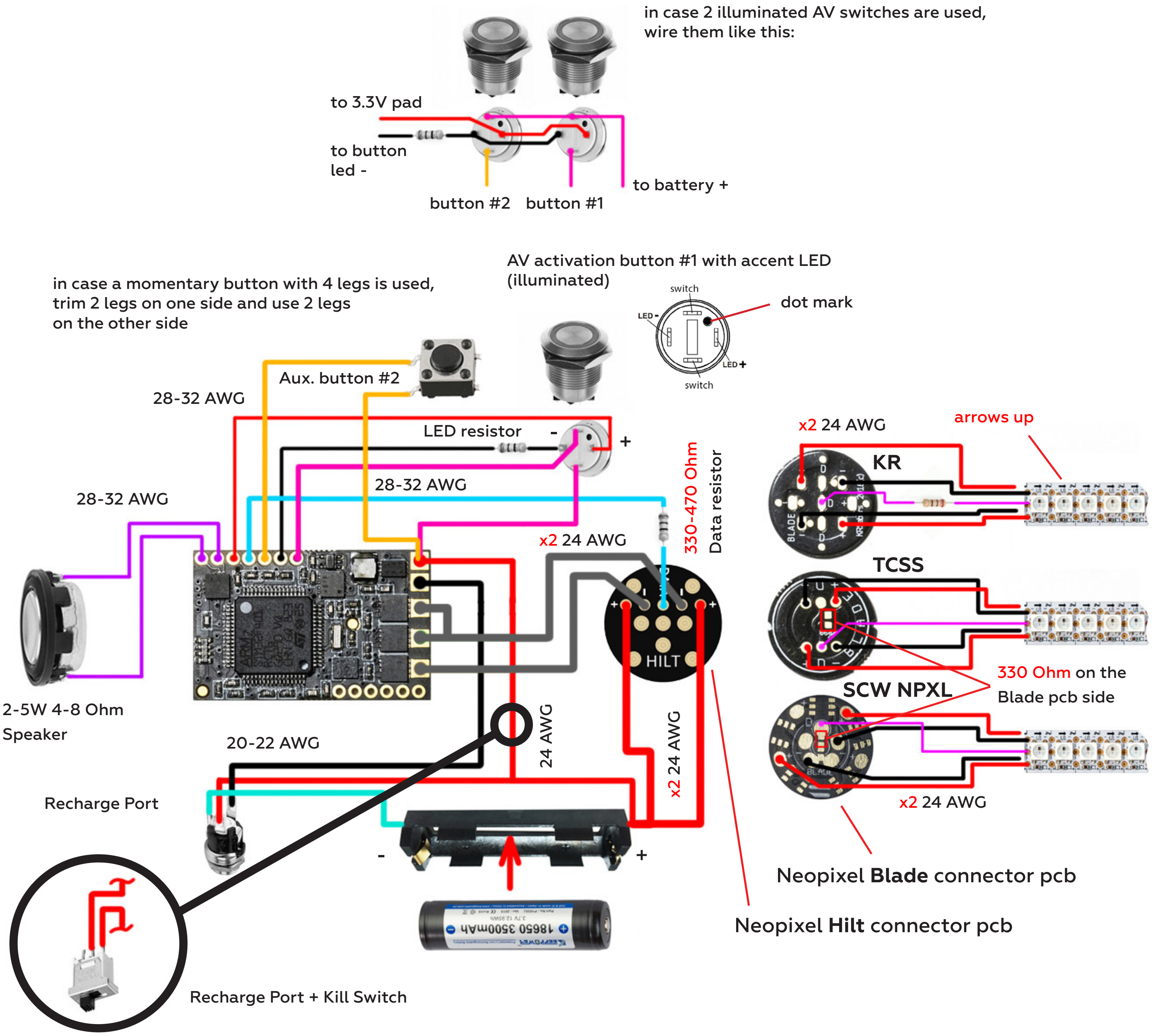


Use the V2 resistors option from the ShtokCustomWorx NPXL connector manual:

>SCW NPXL connector PDF manual download link<

Recommended power wire gauges (22 AWG) are given for 2-strip blade. For 3-strip blade you gonna need at least 20 AWG wires.

Basic Neopixel/Pixelblade wiring diagram
with KR-sabers connector



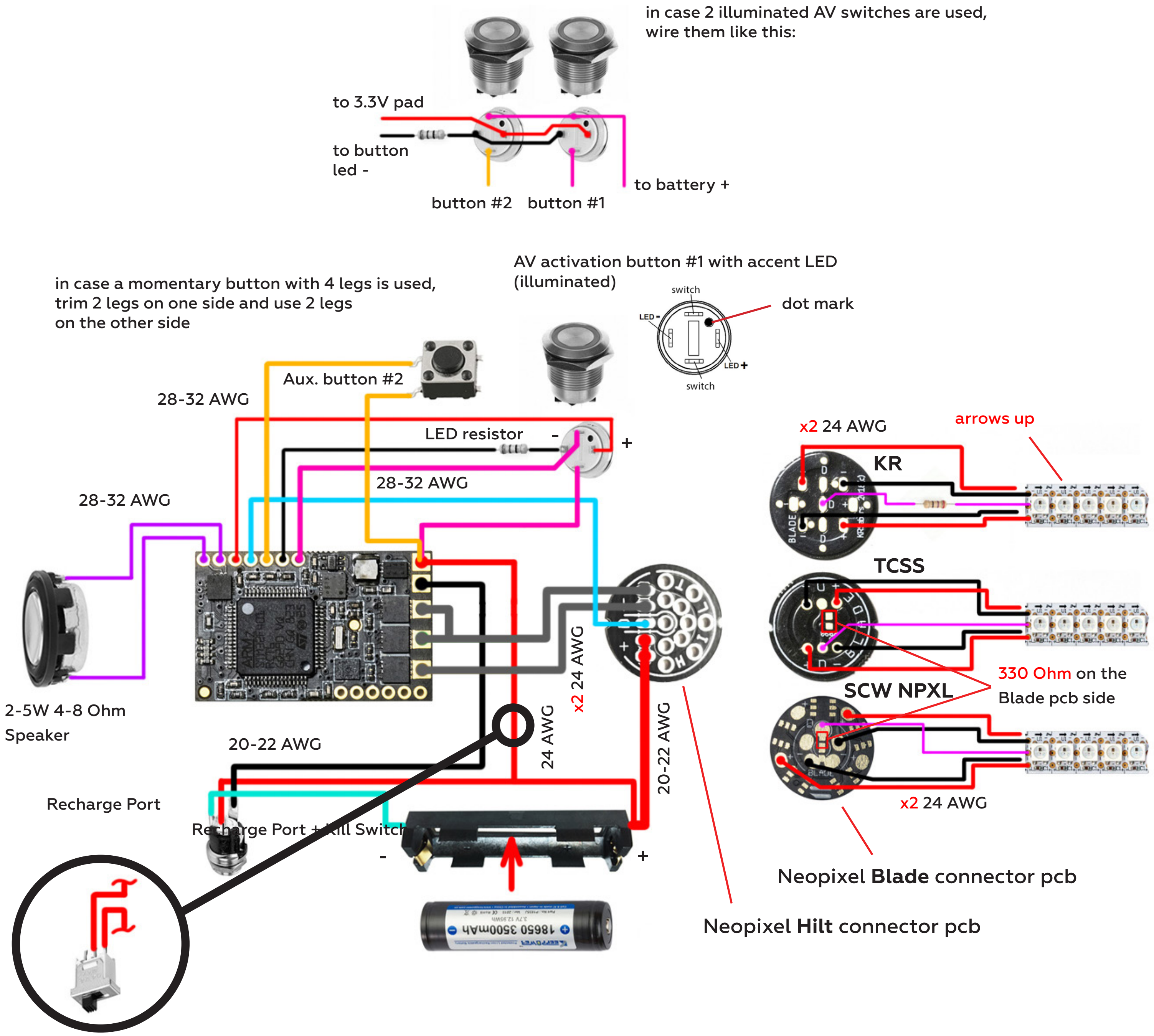
in case 2 illuminated AV switches are used, wire them like this:

in case a momentary button with 4 legs is used, trim 2 legs on one side and use 2 legs on the other side

AV activation button #1 with accent LED (illuminated)

Recommended power wire gauges (22 AWG) are given for 2-strip blade. For 3-strip blade you gonna need at least 20 AWG wires.

Basic Neopixel/Pixelblade wiring diagram **with TCSS connector**



in case 2 illuminated AV switches are used, wire them like this:

in case a momentary button with 4 legs is used, trim 2 legs on one side and use 2 legs on the other side

AV activation button #1 with accent LED (illuminated)

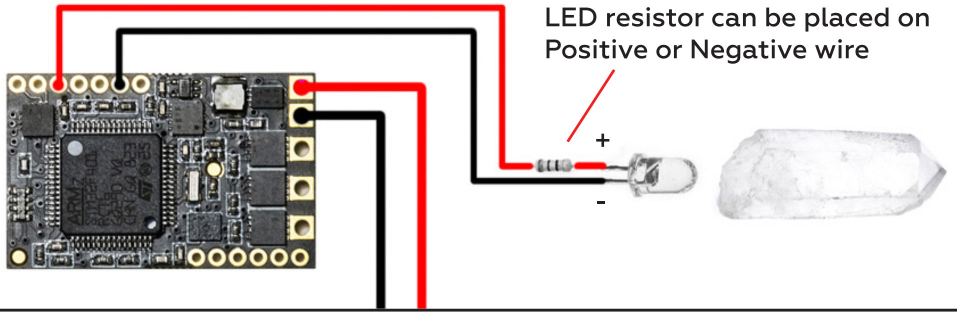
330 Ohm on the Blade pcb side

Recommended power wire gauges (22 AWG) are given for 2-strip blade. For 3-strip blade you gonna need at least 20 AWG wires.

Basic Crystal Chamber LED wiring diagrams

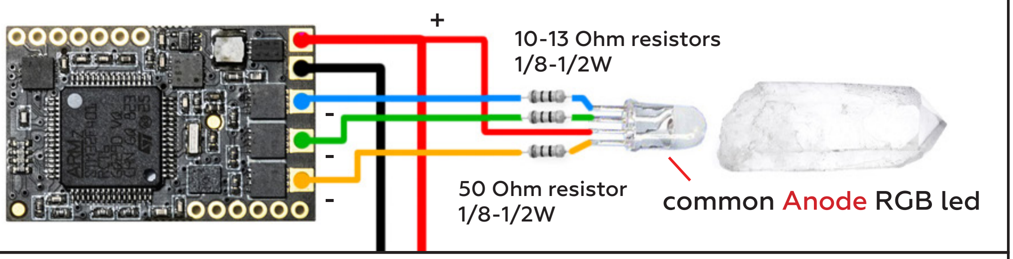
With regular single color accent LED

Using 1 (or multiple) LED/LEDs connected to **+3.3V** and **Button led -** pads.
Crystal LED has it's own "accent" effects.



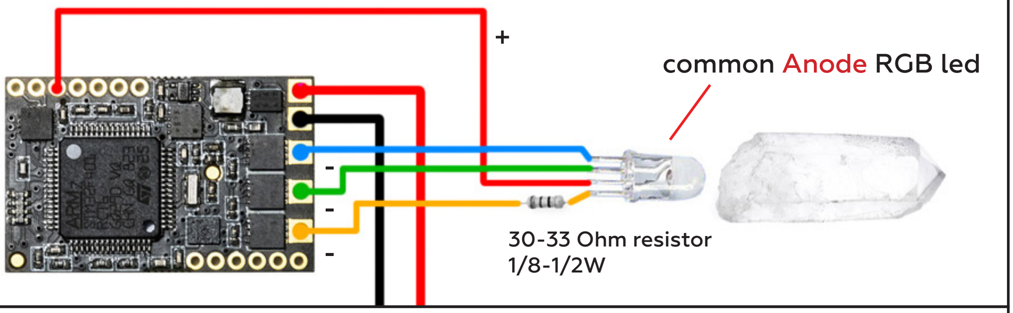
With RGB accent led

Using 1 (or multiple) RGB LED/LEDs connected to **Battery +** and **LED channel 1-3** pads.
Crystal LED mirrors all blade effects.



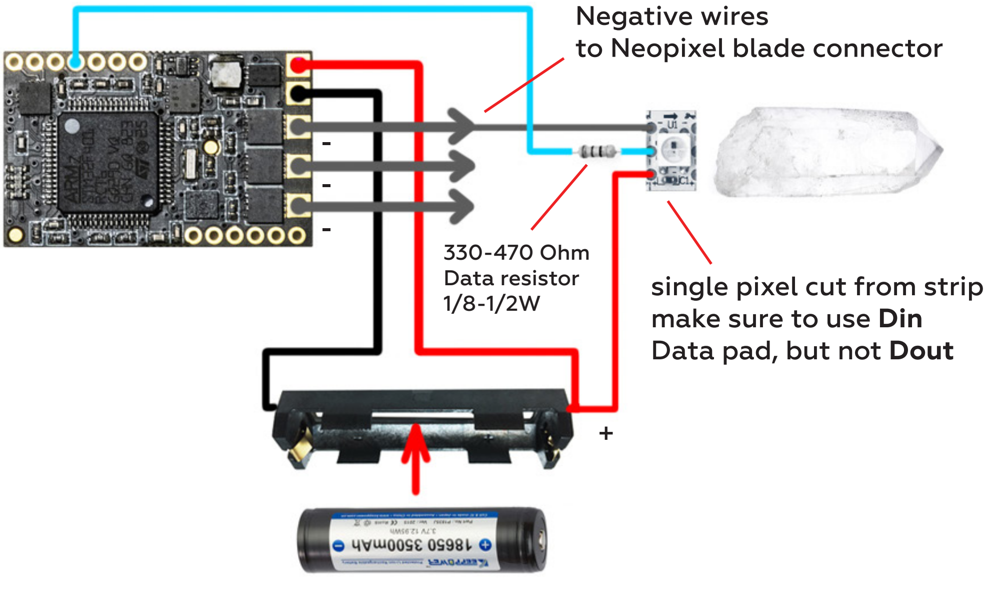
With RGB accent led

Using 1 (or multiple) RGB LED/LEDs connected to **+3.3V** and **LED channel 1-3** pads.
Crystal LED mirrors all blade effects.
NOTE: Green and Blue colors will be slightly dimmer due to 3.3V limit.



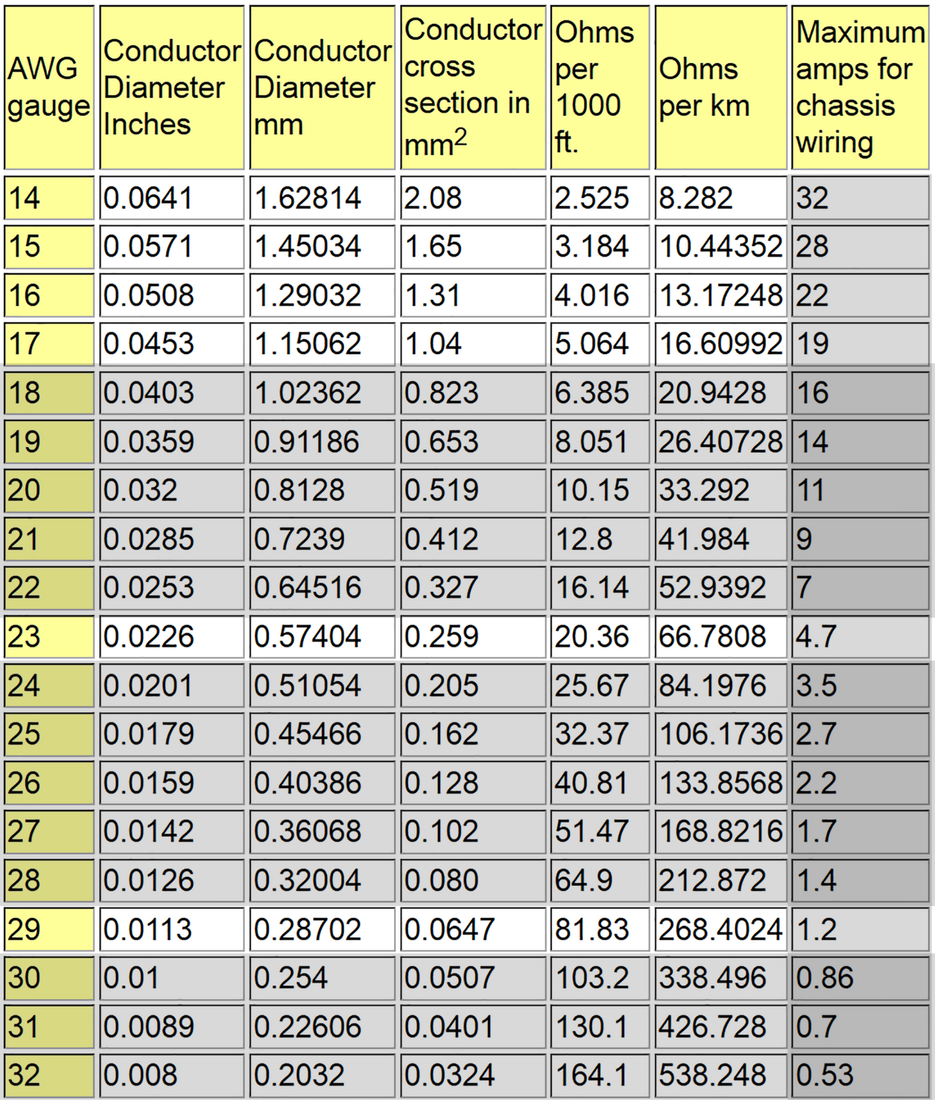
With RGB pixel/pixels

Using 1 (or multiple) pixel/pixels connected to same Positive, Negative and Data wires as the main blade.
Crystal pixel mirrors all blade effects.



WIRE GAUGE GUIDE

Which wire gauge is recommended to use for Positive and Negative power leads for maximum blade brightness efficiency



AWG gauge	Conductor Diameter Inches	Conductor Diameter mm	Conductor cross section in mm^2	Ohms per 1000 ft.	Ohms per km	Maximum amps for chassis wiring
14	0.0641	1.62814	2.08	2.525	8.282	32
15	0.0571	1.45034	1.65	3.184	10.44352	28
16	0.0508	1.29032	1.31	4.016	13.17248	22
17	0.0453	1.15062	1.04	5.064	16.60992	19
18	0.0403	1.02362	0.823	6.385	20.9428	16
19	0.0359	0.91186	0.653	8.051	26.40728	14
20	0.032	0.8128	0.519	10.15	33.292	11
21	0.0285	0.7239	0.412	12.8	41.984	9
22	0.0253	0.64516	0.327	16.14	52.9392	7
23	0.0226	0.57404	0.259	20.36	66.7808	4.7
24	0.0201	0.51054	0.205	25.67	84.1976	3.5
25	0.0179	0.45466	0.162	32.37	106.1736	2.7
26	0.0159	0.40386	0.128	40.81	133.8568	2.2
27	0.0142	0.36068	0.102	51.47	168.8216	1.7
28	0.0126	0.32004	0.080	64.9	212.872	1.4
29	0.0113	0.28702	0.0647	81.83	268.4024	1.2
30	0.01	0.254	0.0507	103.2	338.496	0.86
31	0.0089	0.22606	0.0401	130.1	426.728	0.7
32	0.008	0.2032	0.0324	164.1	538.248	0.53

Chart from PowerStream.com

Neopixel strips
Battery
Recharge Port
Kill Switch

Tri-Cree LED
Battery
Recharge Port
Kill Switch

Everything else

Neopixel strips build (3-17 amperes load)			Tri-Cree LED build (1-4 amperes load)
2-strip	3-strip	4-strip	**28-24 AWG recommended for battery wiring, choose regarding particular build** *30 AWG possible for single 3W Cree LED wiring (one wire per die)*
22 AWG single or 24 AWG dual in parallel	20 AWG single or 23 AWG dual in parallel	18 AWG single or 22 AWG dual in parallel	

For all other components except Neopixel blade strips, high power Tri-Cree LEDs, battery and recharge port/Kill Switch – a 30-32 AWG wire can be used because they are low current circuits (5-500mA) (accent leds, activation and AUX switches, speaker, bluetooth module, RICE port etc.).

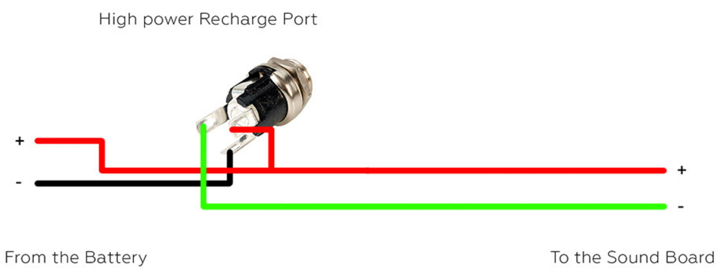
RECHARGE PORTS AND KILL SWITCHES

		3 Amps	5 Amps	6 Amps	7 Amps	8 Amps	11 Amps
	2.1mm Switchcraft 721A Recharge port	OK	OK	OK	OK	OK	96%
	1.3mm Recharge port CUI PJ-013D Martin Beyer 1.3mm Recharge port	OK	OK	OK	98%	97%	95.5%
	Martin Beyer Kill Switch	OK	OK	OK	OK	OK	98%
	1.3mm Recharge port CUI PJ-075DH	OK	OK	OK	97%	95%	94%
	regular cheap 1.3mm Recharge port	75%	melted	melted	melted	melted	melted
	3A Kill Switch CK TS01CQE	OK	OK	OK	OK	OK	98%
	Mini 6pin SMD Slide Switch MSS22D18	OK	OK	70%	melted	melted	melted

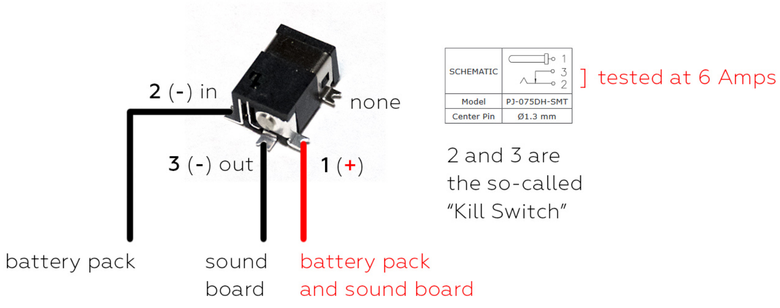
OK — safe to use

1-96% — efficiency (less than 95% not recommended!)

How to wire Recharge Ports



CUI PJ-075DH-SMT
High power 1.3mm recharge port
wiring diagram



NEOPIXEL STRIPS CURRENT DRAW

Neopixel WS2812B/SK6812 strips tested approximate current consumption chart

Tested at 3.7V, 143 leds per strip, at max brightness

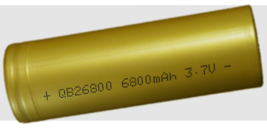
№ of strips	current	1 color without flicker / with flicker	2 colors mixed without flicker / with flicker	3 colors mixed for white without flicker / with flicker
1	Total	2 / 1.9 A	3.6 / 3.3 A	5.2 / 4.9 A
	Per LED	14 / 12.9 mA	12.6 / 11.5 mA	12.1 / 11.4 mA
2	Total	3.7 / 3.5 A	6.9 / 6.4 A	9.9 / 9.3 A
	Per LED	13 / 12.2 mA	12 / 11.1 mA	11.5 / 10.8 mA
3	Total	5.4 / 4.5 A	10.1 / 9.5 A	14.4 / 13.5 A
	Per LED	12.6 / 11.6 mA	11.8 / 11.1 mA	11.2 / 10.5 mA
4	Total	7.1 / 6.7 A	13 / 12.4 A	17.7 / 16.6 A
	Per LED	12.4 / 11.8 mA	11.4 / 10.8 mA	10.3 / 9.7 mA
5	Total	8.8 / 8.4 A	15.7 / 15 A	20.6 / 19.5 A
	Per LED	12.3 / 11.7 mA	11 / 10.5 mA	9.6 / 9.1 mA

RECOMMENDED BATTERIES CHART

SHTOK CUSTOM WORX

Best batteries for sabers

2020

SIZE	BRAND/MODEL	
18350	Keeppower 1200mAh 8A Protected 	Keeppower 1200mAh 10A Unprotected (requires external protection pcb)
14500	Keeppower P1450C2 1000mAh 4A Protected 	Efest IMR14500 V2 650mAh 9A Unprotected (requires external protection pcb)
14650	Efest IMR14650 950mAh 5A Unprotected (requires external protection pcb) 	Keeppower 1100mAh 2-3A Protected
16650	Keeppower 2500mAh 5A Protected 	Sanyo UR16650ZTA 2500mAh 5A Unprotected (requires external protection pcb)
18500	Keeppower P1850J2 2000mAh 4A Protected 	Keeppower IMR18500 1100mAh 10A Unprotected (requires external protection pcb)
18650	Keeppower 3500mAh 10A Protected 	Keeppower 3120mAh 15A Protected
21700	Acebeam 5100mAh 20A Protected 	Keeppower 5000mAh 10A Protected
26650	Keeppower 6000mAh 10-15A Protected 	Keeppower 5500mAh 10A Protected
26800	QueenBattery 6800mAh 30A Unprotected (requires external protection pcb)	

These batteries work for both Tri-Cree and Neopixel setups.

For Tri-Cree LED setup choose the one with highest capacity value (**mAh**), for Neopixel setup choose the one with highest max drain value (**A**).

mAh – *milliamperes per hour: the battery energy capacity rating, means how long the battery will run at a single charge* – ***run time***

A – *Amperes: the battery energy max drain rating, means how much Amperes this battery can provide continuousl* – ***blade brightness performance under high current load***

For Tri-Cree LED setup batteries with 2-3A drain rating are OK (can be higher, but lower are not recommended). For Neopixel setup batteries with 10-15A drain rating are recommended (can be higher, but lower are not recommended). For battery sizes 14500, 14650, 16650, 18350, 18500 it's hard or impossible to find a good capacity with high drain rating, so 5-8A examples from the chart above are best options.

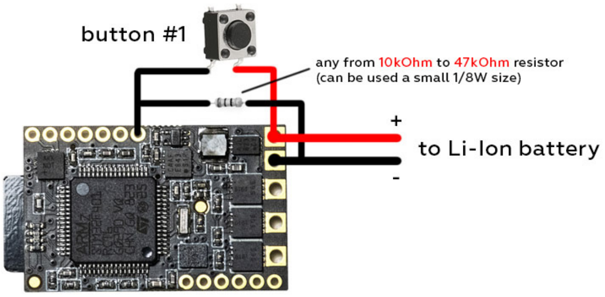
TROUBLESHOOTING

Quick troubleshooting tips

How to solve most common issues

The Activation button #1 continuously loops after booting

– It happens if the user fires internal pull-down resistor due to ESD. Add external **10-47kOhm** pull-down resistor to the switch circuit:


button #1
any from 10kOhm to 47kOhm resistor
(can be used a small 1/8W size)
+
to Li-Ion battery
-

My last color profile is not displaying

– The config.ini files must have a blank line at the end of them. Please be sure to press ENTER after the last color profile has been added ensuring a blank line is at the end of the text file.

The saber will not wake up from Deep Sleep mode

– This is due to the activation switch being wired incorrectly. The Verso has a different wiring method for the switch to other boards on the market. Be sure to study the manual and diagram carefully ensuring you have wired the switch correctly.

I have loaded a new font, but some sounds aren't playing

– Please check you have the proper sound files names and format in your font folder (compare them to the default fonts sound files).

The saber is automatically shutting off

– This is the default Low Battery safety feature working on the board. It will shut the saber off if the battery is below a defined value in the config.ini file. You can disable this feature or adjust the minimum thresholds for activation. Please see Page 10 for more information on this.

What does the sound "Crystal Core Unstable" mean

– This is the Low battery notification sound on the default sound font.

Be careful not to hit the yellow capacitor underneath the board

– This is a tantalum capacitor required for a stable performance of the board.

8 Ohm speakers vs 4 Ohm speakers

– 8 ohm speakers are usually more reliable and can be driven at max volume (=2.0) and gain safely.
– 4 ohm speakers are more sensitive to overpowering, and it's recommedned to use them with half of the max volume setting (=1.0).

TROUBLESHOOTING

Quick troubleshooting tips

How to solve most common issues

TROUBLESHOOTING

Quick troubleshooting tips

How to solve most common issues

5 ABOUT

About this document

Revision history

The table below displays the revision history for the chapters in this manual.

Chapter	Date	Revision	Changes made
All	Jan 2020	1.0	First draft.
- Features - Editing parameters - Wiring diagrams - Recommended batteries - Troubleshooting	Mar 2020	1.1	- added 1-button operation instructions - troubleshooting help - added wiring diagrams for illuminated and non-illuminated AV switches - changes to KR Neopixel connector wiring diagram - changes to "Features" list - changes to "Low Battery" information - added "Emergency Shutdown" section - added Crystal Chamber wiring diagrams - updated "Recommended Batteries" section info
- Features - How To Use - Editing parameters - Wiring diagrams - Troubleshooting	Feb 2021	2.0	- changes to "Features" list - added "How To Use" pages - changes to 1-button and added 2-buttons operation instructions - changes to parameters changing instructions - changes to wiring diagrams (added second button wiring) - added external pull-down resistor wiring diagram in troubleshooting help