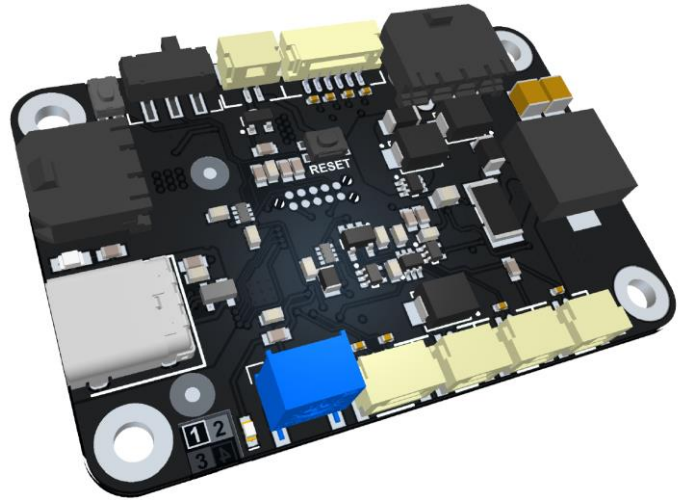


# ETH zürich IlluminationBoard V2 REV2 & 3

## Features

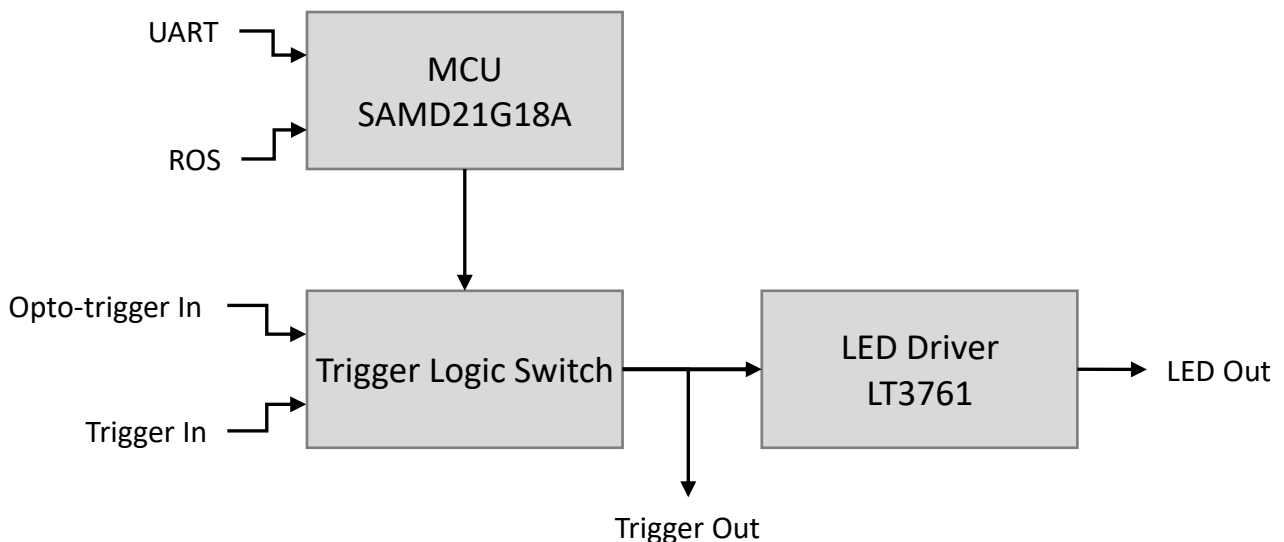
- 21V – 28V supply
- External trigger inputs
  - o Opto-isolated
  - o Non-isolated
- Trigger output for synching multiple boards
- Manual brightness change with potentiometer
- Fan output (controllable)
- Support for LEDPCB (high-power LED board with temperature measurement capability)
- Easy reprogramming via USB



## Description

Strong LED Driver with external trigger capability. IlluminationBoard can be configured to work with ROS or other environments.

## Block Diagram



## Contents

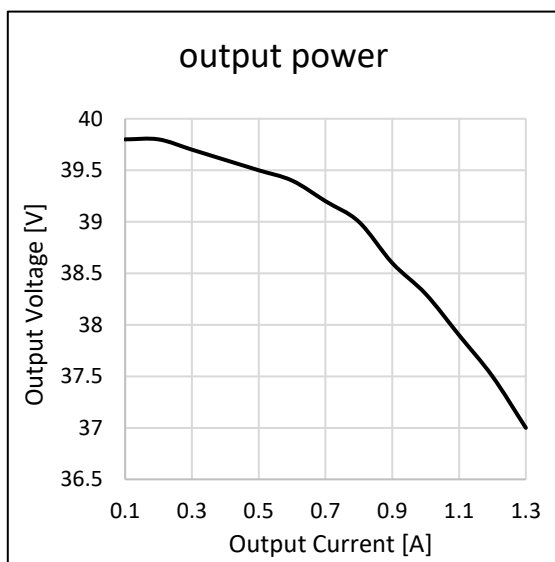
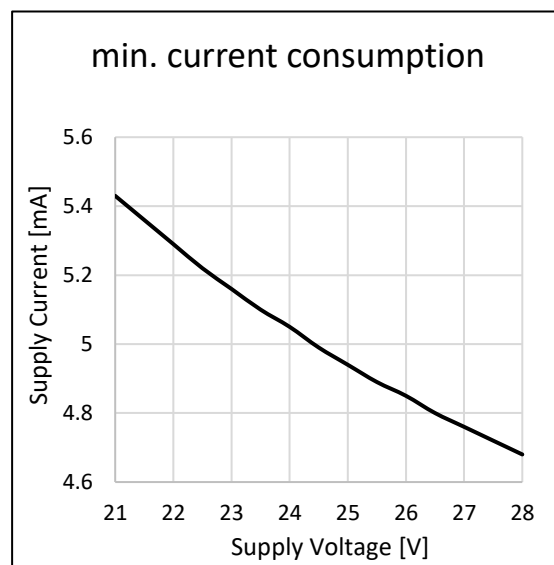
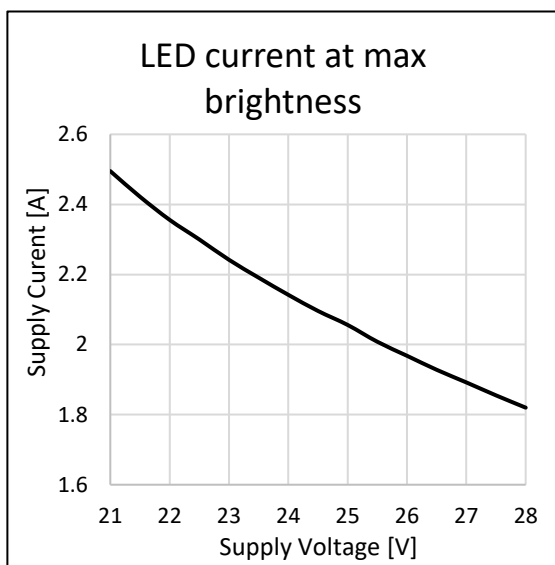
<b>1. Electrical Characteristics</b>	<b>3</b>
1.1. Curves	4
<b>2. Firmware</b>	<b>5</b>
2.1. ROS	5
<b>3. Dimensions</b>	<b>6</b>
<b>4. Connectors</b>	<b>7</b>
4.1. Pinout	8
<b>5. Revisions</b>	<b>9</b>
5.1. Rev 1	9
5.2. Rev 2 & 3	9

# 1. Electrical Characteristics

(U<sub>SUPPLY</sub> = 24V unless otherwise noted)

Characteristics	Measurement			Unit
	Min	Typ	Max	
Operating Supply Voltage	21	24	28	V
Maximum Supply Current (LED full brightness)	2.15	2.16	2.18	A
Maximum Board Temperature (LED full brightness, Ambient temperature: 25.5°C)			92	°C
<b>External Trigger Input non-isolated</b>				
Input Voltage	0	3.3	5.5	V
High-level input voltage	1.8			V
Low-level input voltage			1.3	V
<b>External Trigger Input opto-isolated</b>				
Input Voltage	0	5	30	V
High-level input voltage	1.3			V
Low-level input voltage			1.2	V

## 1.1. Curves



## 2. Firmware

The latest firmware for IlluminationBoard V2 can be found [here on GitHub](#). To program the firmware onto your IlluminationBoard, you need to flash the Arduino MKRZERO bootloader first. It is recommended to do so with Atmel Studio.

After flashing the bootloader, you can program the firmware via the USB Type-C port<sup>1</sup>.

### 2.1. ROS

The ROS version of the IlluminationBoard firmware runs with Arduino roserial.

Communication is done via three rostopics: *pwm* (UInt8), *trigger* (Bool), *config* (UInt8).

Topic	Description	Parameters
pwm	Sets pwm duty cycle of LED. Automatically sets dutycycle source to ros.	std::msgs/UInt8 dutycycle range: 0-255
trigger	Triggers the LED on or off	std::msgs/Bool trigger <b>0</b> : LED off <b>1</b> : LED on
config	Configures the trigger input and pwm dutycycle source	<b>Bit 0:</b> - <b>0</b> : External trigger source - <b>1</b> : Internal trigger source  <b>Bit 1:</b> - <b>0</b> : use ros as dutycycle source - <b>1</b> : use potentiometer as source for dutycycle

Following steps have been used to connect to and communicate with the IlluminationBoard via ROS:

1. Start a roscore service

```
roscore
```

2. Use rosrn to connect to the IlluminationBoard<sup>2</sup>:

```
roslaunch roserial_python serial_node.py /dev/ttyACM0
```

3. To send a message over ROSSerial use one of the following commands:

- a. pwm:

```
rostopic pub Illuminationboard/pwm std_msgs/UInt8 [0...255]
```

- b. trigger:

```
rostopic pub Illuminationboard/trigger std_msgs/Bool [false, true]
```

- c. config:

```
rostopic pub Illuminationboard/config std_msgs/UInt8 [0...3]
```

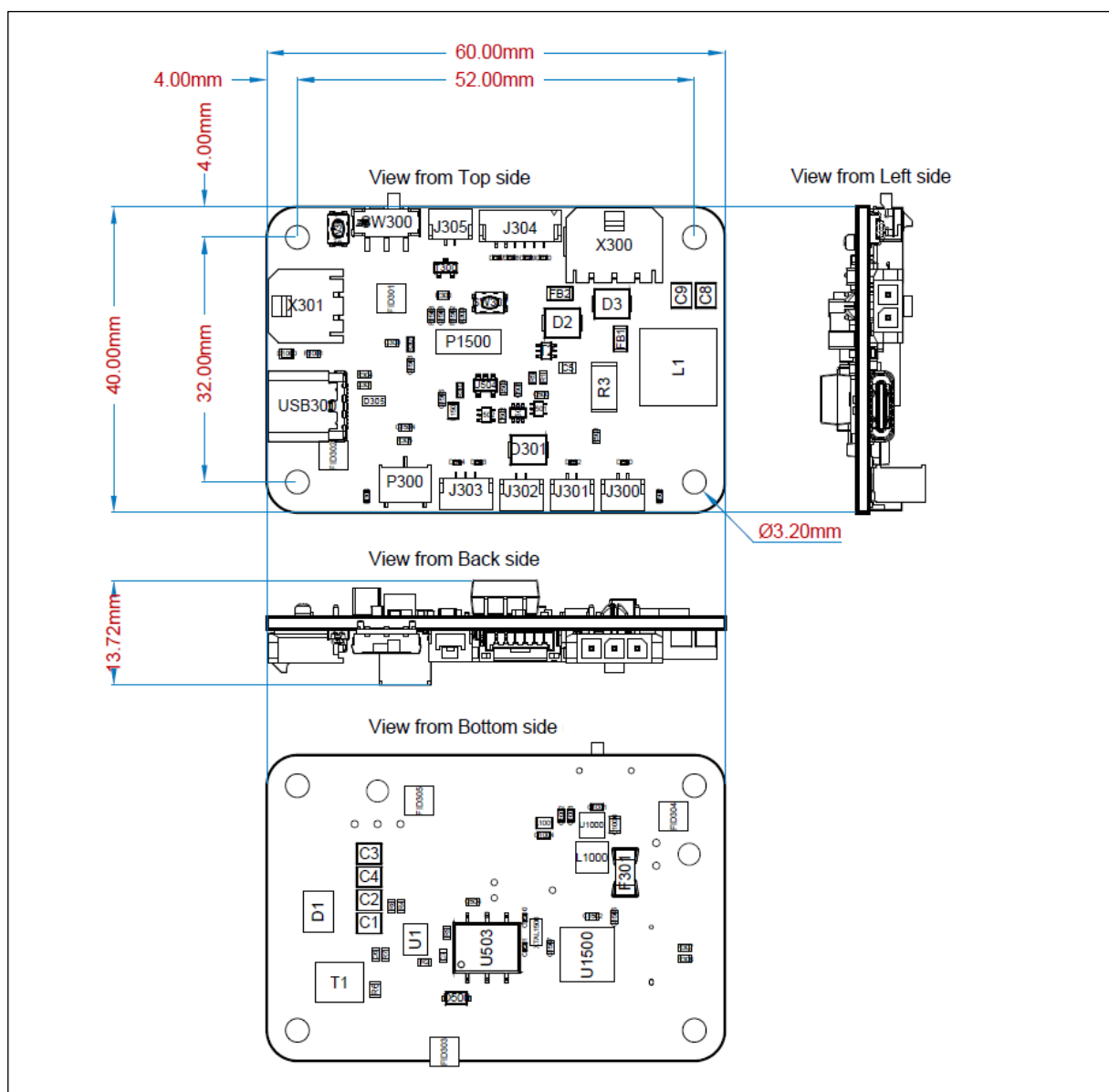
Tested on ROS Noetic with Ubuntu 20.04.1 LTS.

For further information about ROSSerial, visit [wiki.ros.org](http://wiki.ros.org).

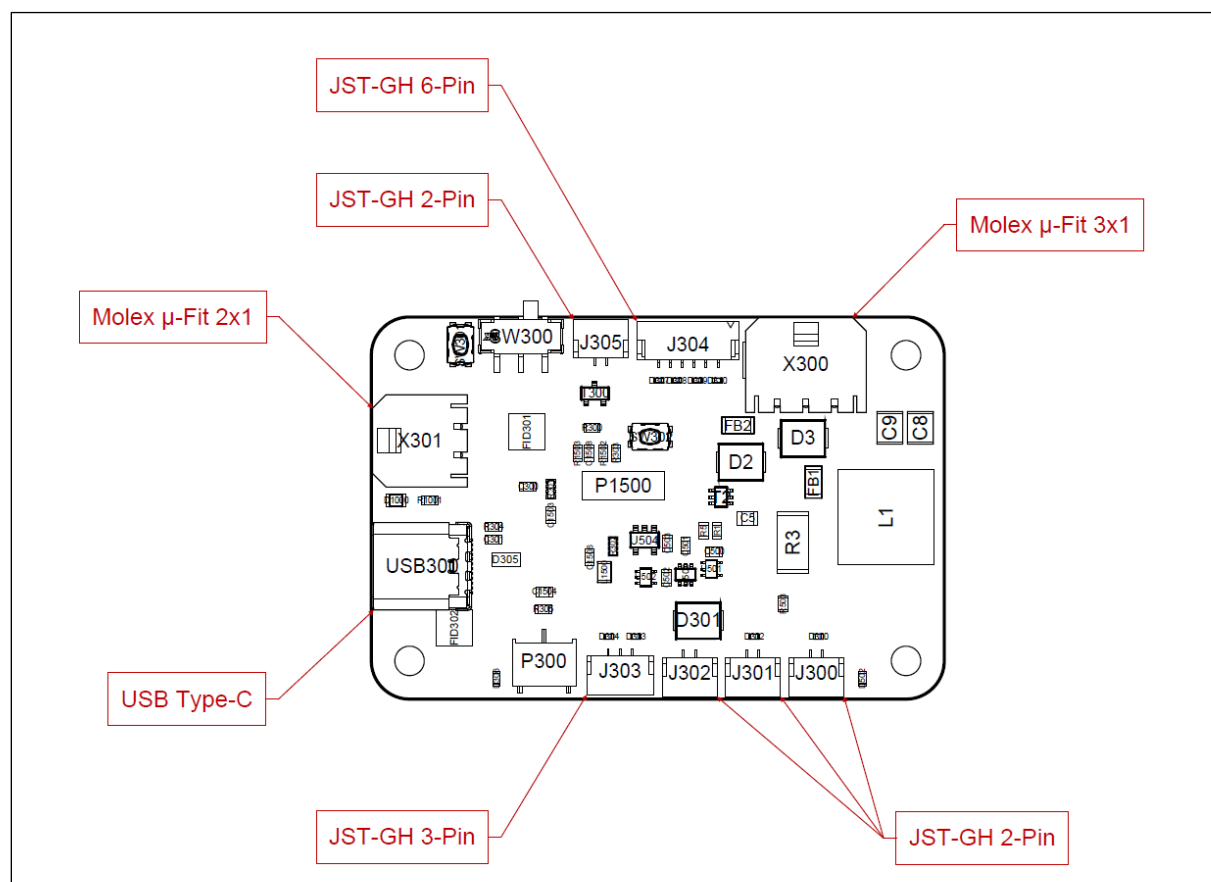
<sup>1</sup> IlluminationBoard CAN NOT be powered by USB only, needs external supply (24V)

<sup>2</sup> The specified /dev/tty USB connection may vary.

### 3. Dimensions



## 4. Connectors

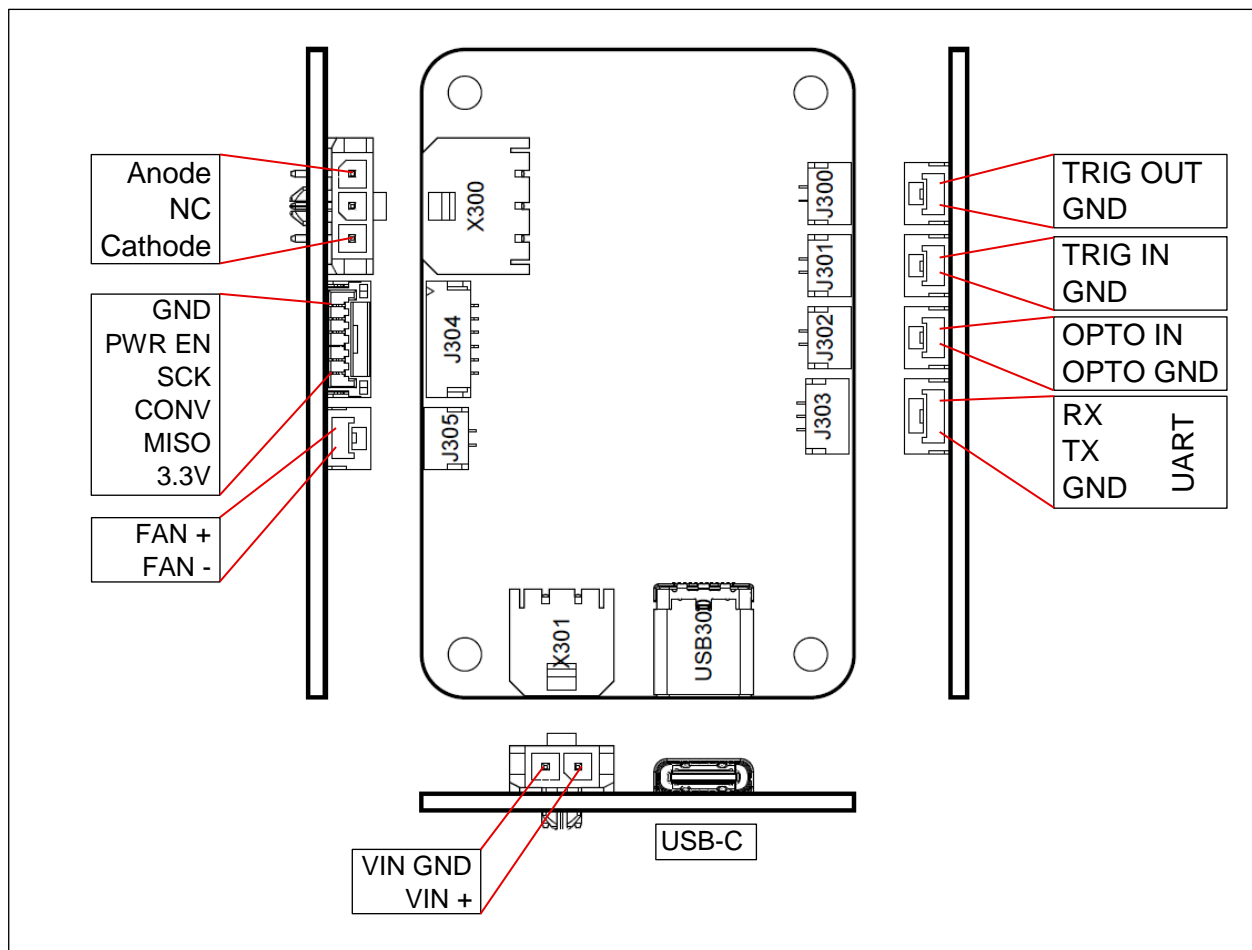


Name	Manufacturer Part Number	Supplier Part Number <sup>1</sup>
JST-GH 2-Pin	GHR-02V-S	455-1592-ND
JST-GH 3-Pin	GHR-03V-S	455-1593-ND
JST-GH 6-Pin	GHR-06V-S	455-1596-ND
Molex μ-Fit 2x1	0436450200	WM1845-ND
Molex μ-Fit 3x1	0436450300	WM1846-ND

These part numbers refer to the corresponding counterparts of the connectors.

<sup>1</sup> Digi-Key Electronics

## 4.1. Pinout





## 5. Revisions

### 5.1. Rev 1

IlluminationBoard V2 Rev 1 uses a different pinout on the MCU, thus it needs a special Arduino package. This package can be found on the [GitHub page](#) and needs to be installed in the Arduino IDE.

Furthermore, revision 1 has some overheating issues and the TEMP SENS connector has a swapped pinout.

### 5.2. Rev 2 & 3

Revision 2 and 3 are practically the same board, only difference being the removal of a diode, addition of some missing Tracks on the PCB and different TVS diodes on the TEMP SENS as well as the UART pins.

Since revision 1, there have been some more changes:

- Improved heat dissipation
  - o Changed inductor
  - o Changed FET
- Fan control
- Relocation of TEMP SENS connector on the PCB (closer to LED OUT)
- Swapped position of USB and VIN components (USB Type-C connector and 2x1 Molex Micro-Fit connector)
- Adjusted placement of some more components
- Mirrored TEMP SENS pinout for a 1-to-1 connection to the LEDPCB
- Adjusted MCU pinout to MKRZERO package
- Smaller potentiometer
- Added standard Arduino MKRZERO LED for debugging
- Added fuse