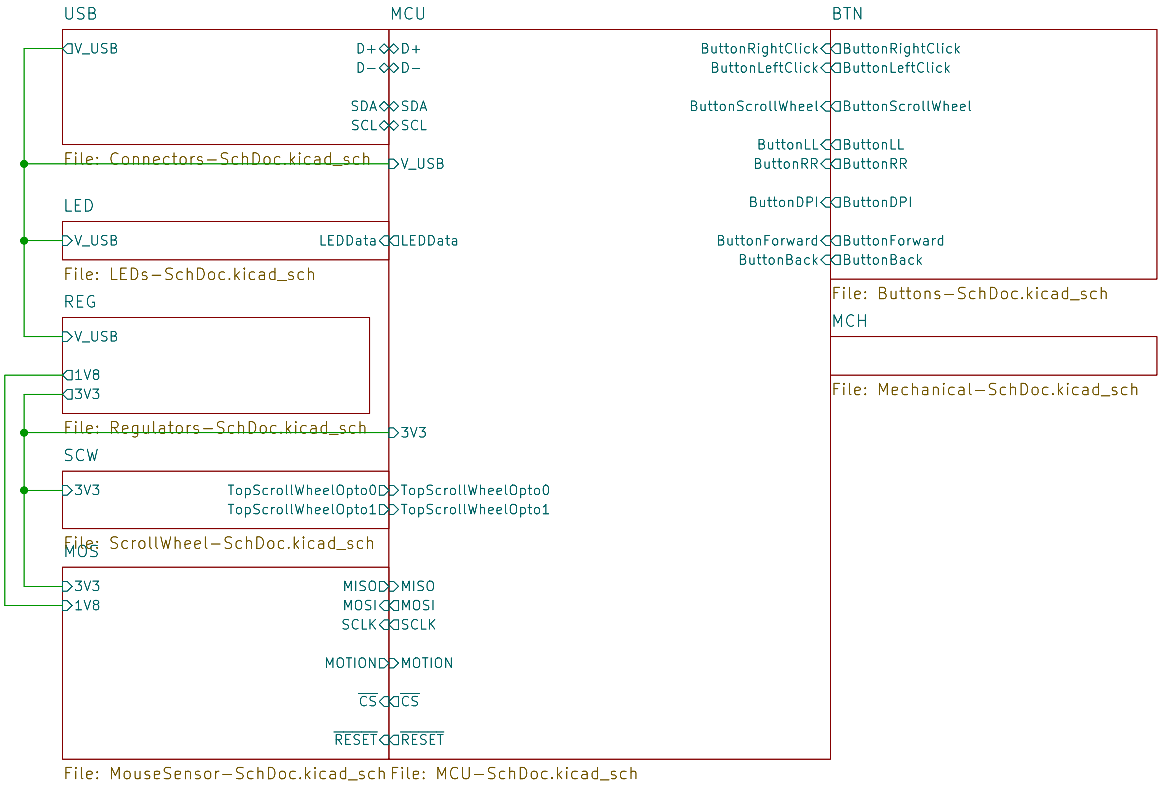
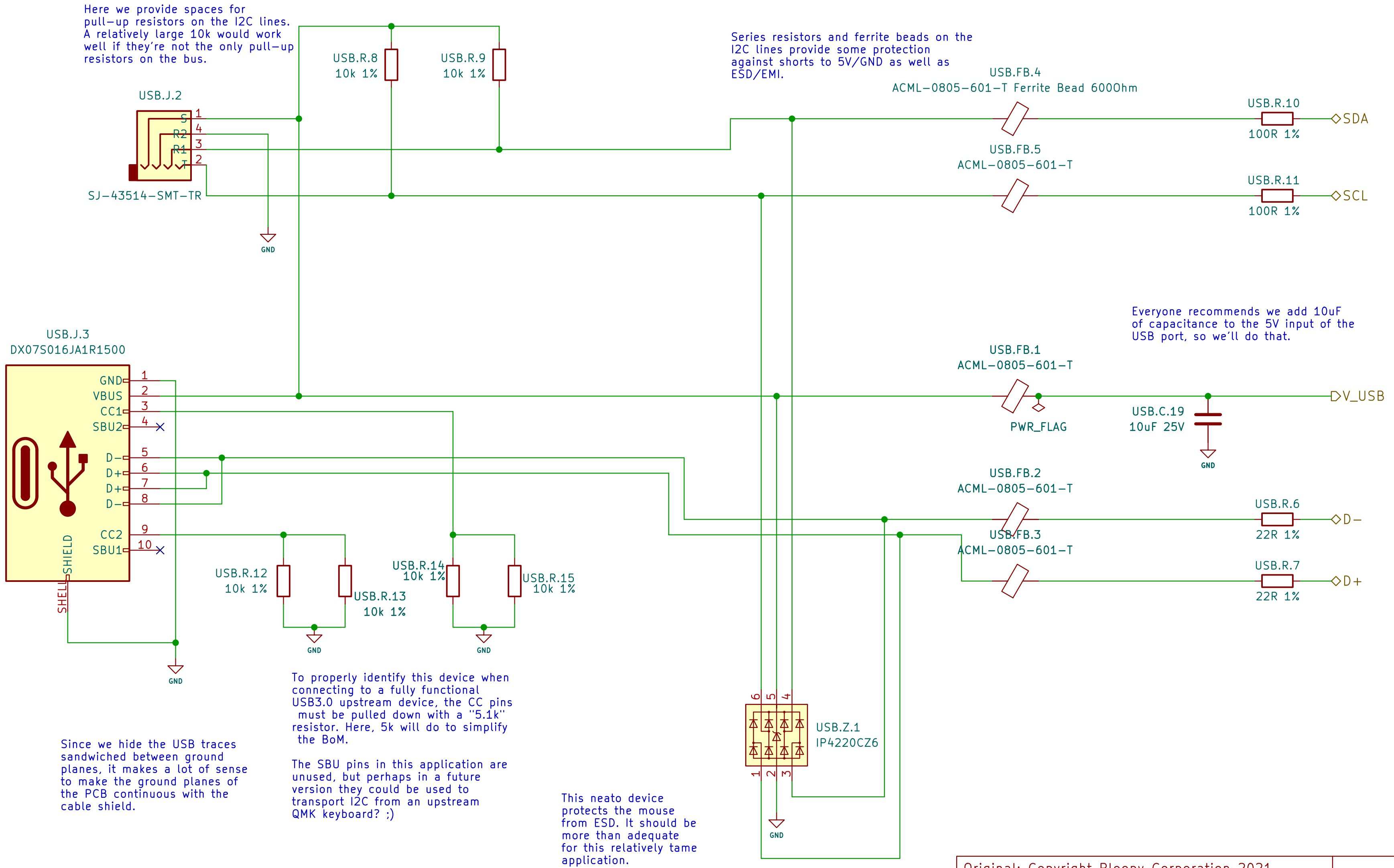


Sheet: /
File: QMKMouse.kicad_sch

Title: QMKMouse-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 1/9

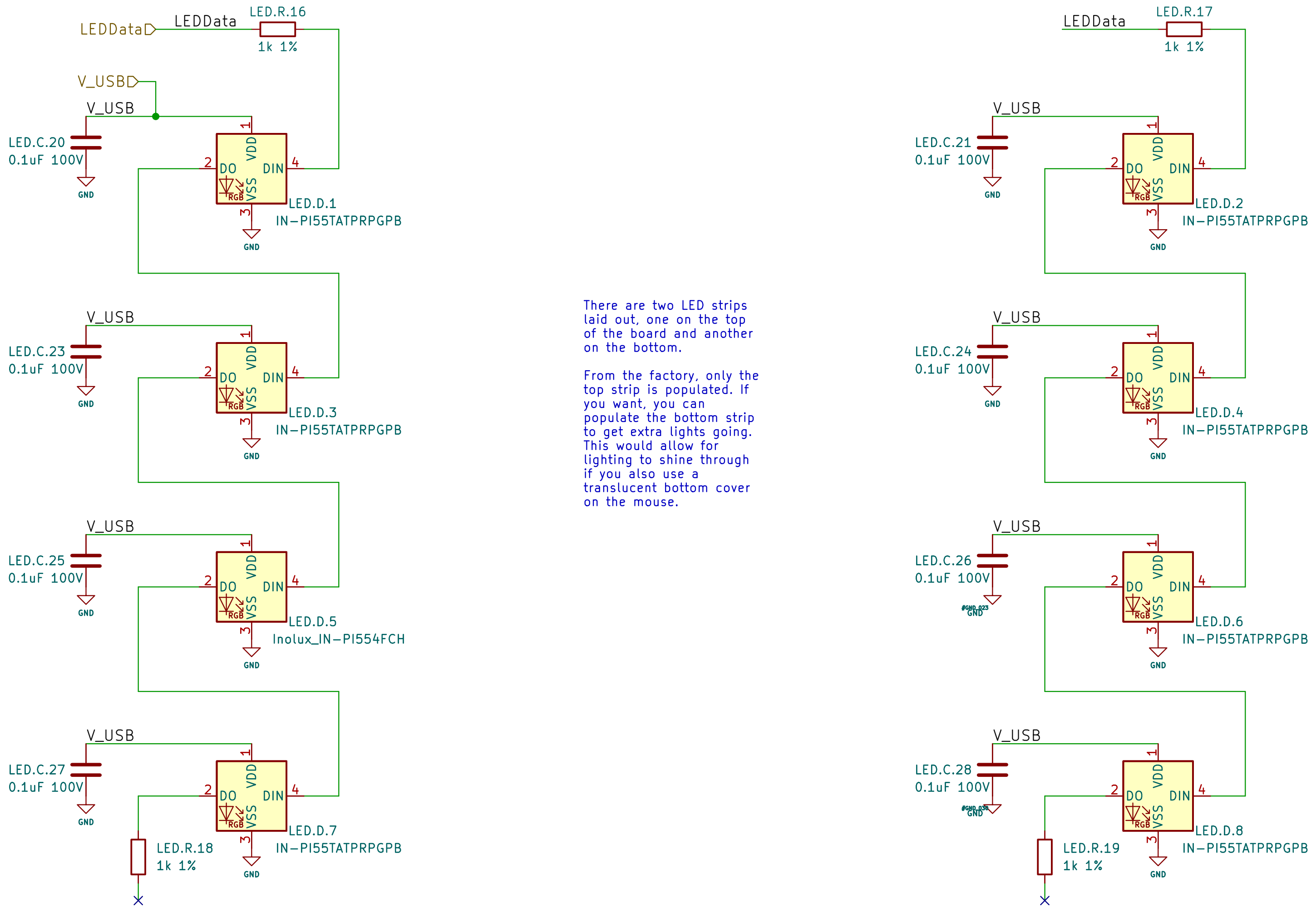


Sheet: /USB/
File: Connectors-SchDoc.kicad_sch

Title: Connectors-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 2/9



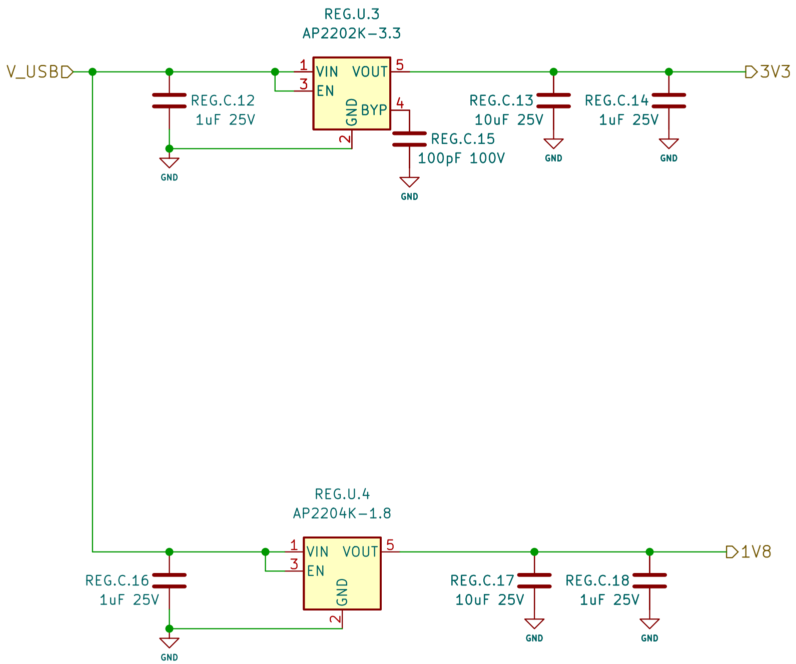
There are two LED strips laid out, one on the top of the board and another on the bottom.

From the factory, only the top strip is populated. If you want, you can populate the bottom strip to get extra lights going. This would allow for lighting to shine through if you also use a translucent bottom cover on the mouse.

Sheet: /LED/
File: LEDs−SchDoc.kicad_sch

Title: LEDs−SchDoc

Size: A4 Date: 17 01 2023 **Rev: 2.001**
KiCad E.D.A. kicad (6.0.10) Id: 3/9

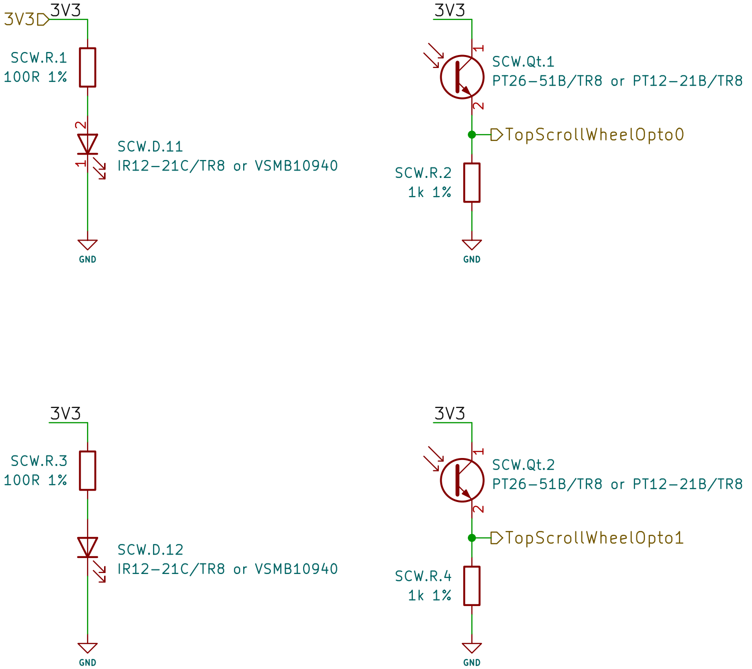


Sheet: /REG/
File: Regulators-SchDoc.kicad_sch

Title: Regulators-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 4/9



The scroll wheel mechanism is a 2-bit optical encoder. The light paths are spaced 1.5 periods over the encoder wheel holes, which should allow for the detection of the rotation direction as well as the rotation itself.

Two IR LEDs are used as light sources; the light from these is obstructed by the encoder wheel built into the scroll wheel mechanism. Two phototransistors are used to detect when the light is blocked/not blocked by the wheel.

Typical LED forward voltage is 1.3V @20mA. To achieve this current level with a 3.3V supply, we'll use a 100R resistor, which is very convenient.

At this power level, the output should be approx 2.3mW/sr.

Since the light path distance is about 12mm, that works out to about 2mW/cm^2, though the actual delivered power will be lower due to the vagaries of the obstructions in the mechanism.

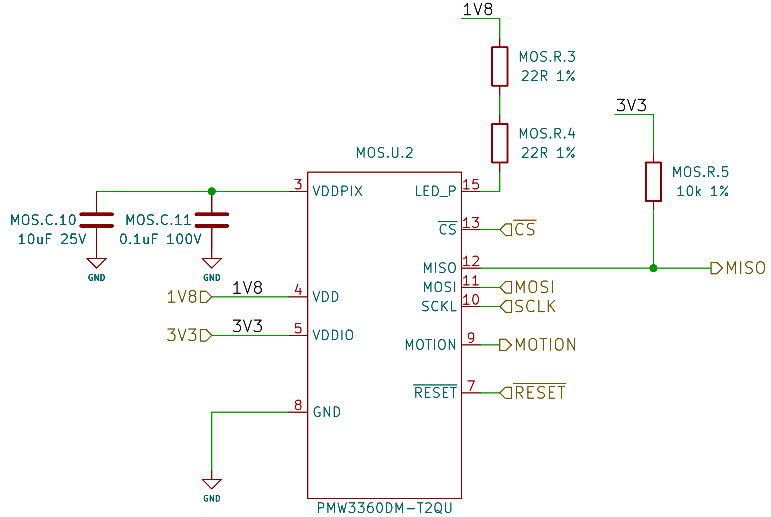
With this amount of light (and accounting for the fact that we'll probably lose a lot of it along the way), we are expecting 1-3mA out of the phototransistor (see figure 6 of the datasheet), so we size the biasing resistor accordingly to produce a useful signal.

Sheet: /SCW/
File: ScrollWheel-SchDoc.kicad_sch

Title: ScrollWheel-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 5/9

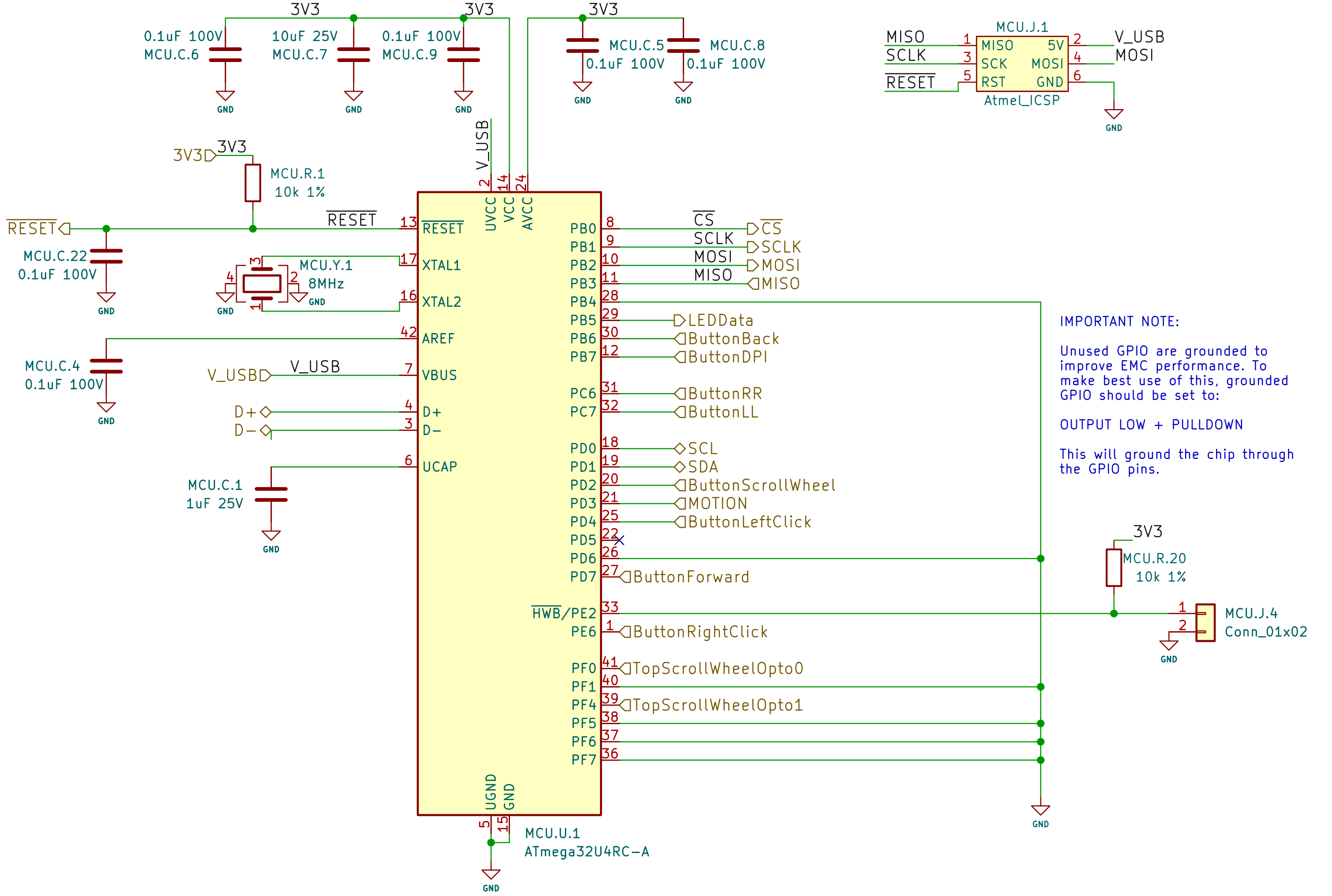


Sheet: /MOS/
File: MouseSensor-SchDoc.kicad_sch

Title: MouseSensor-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 6/9



IMPORTANT NOTE:

Unused GPIO are grounded to improve EMC performance. To make best use of this, grounded GPIO should be set to:

OUTPUT LOW + PULLDOWN

This will ground the chip through the GPIO pins.

The RC variant only means that the internal 8MHz oscillator is enabled by default and that the divider is set so that it results in 1MHz system clock. This is important if we dont supply an external oscillator otherwise we can not programm it.

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7766-8-bit-AVR-ATmega16U4-32U4_Datasheet.pdf#G4.1055626

Sheet: /MCU/
File: MCU-SchDoc.kicad_sch

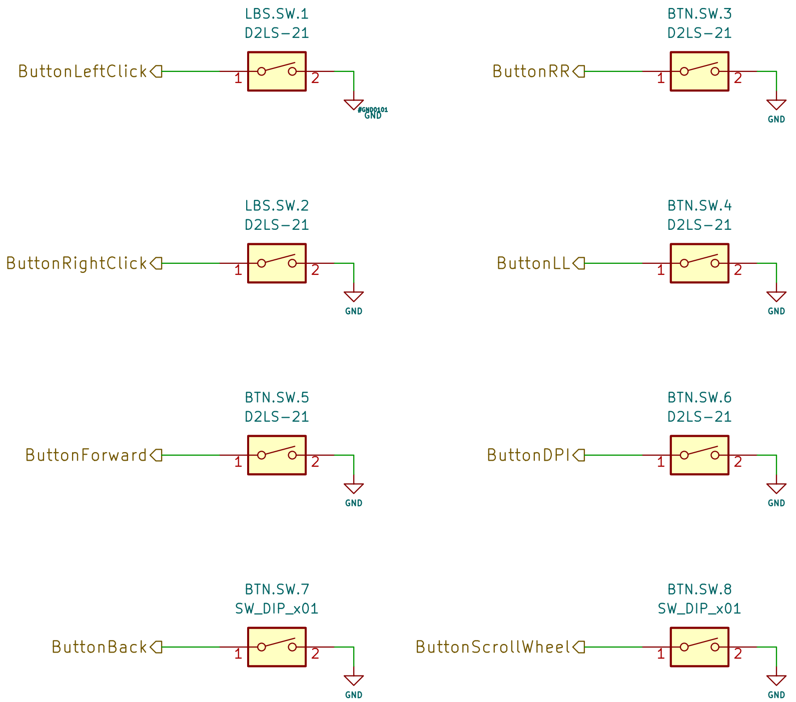
Title: MCU-SchDoc

Size: A4 | Date: 17 01 2023 | **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) | Id: 7/9

These switches can be
found under the various
buttons on the mouse.

Activate the pull-up
resistors on the GPIO
pins attached to these
switches to make them work.

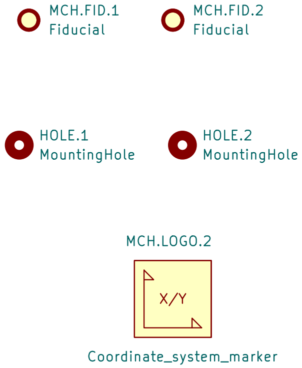


Sheet: /BTN/
File: Buttons-SchDoc.kicad_sch

Title: Buttons-SchDoc

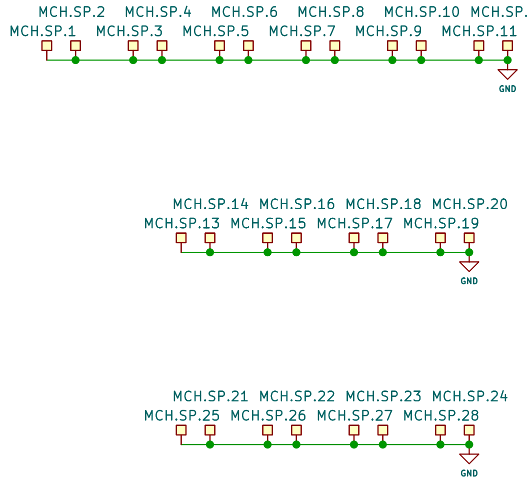
Size: A4	Date: 17 01 2023	**Rev: 2.001**
KiCad E.D.A. kicad (6.0.10)		Id: 8/9

Pick and Place Fiducials



Spark Gaps -- Case

Since the case has gaps in it, we expect ESD to worm its way in via creepage and perhaps other ways. To protect the board from this eventuality, we place spark gaps along the edges.



Sheet: /MCH/
File: Mechanical-SchDoc.kicad_sch

Title: Mechanical-SchDoc

Size: A4 Date: 17 01 2023 **Rev: 2.001**

KiCad E.D.A. kicad (6.0.10) Id: 9/9