Introduction

In response to shortages during the Covid−19 pandemic, TU Delft is developing the POxiM transmissive pulse oximeter. A pulse oximeter is a medical instrument which doctors use to monitor the heart rate and blood oxygen saturation of patients during critical stages of and revalidation from Covid−19. POxiM is intended to be fully qualified for use leading up to and after intensive care, thereby freeing existing more stringently qualified pulse oximeters to be used in intensive care. The main design goal is supplyability: 1000s of units should be locally manufacturable even during severe lockdowns. Simplicity and supply chain diversity are important secondary goals in service of this main goal.

POxiM consists of a disposable finger probe and a reusable wrist computer, containining nearly all of the circuitry. This is it technically only the schematic of the wrist computer, it shows all of the circuitry and can be regarded as the main electrical schematic of POxiM. It is also intended to give an overview of the operation of POxiM, motivate detailed design decisions, and to document the derivation of electronic component parameters. This schematic does not aim to justify the system−level and electronic high−level design decisions. Instead, those are motivated in the POxiM report and accompanying system engineering notes respectively.

Apart from this introduction, the top−level sheet briefly explains the working principle of POxiM and gives an block diagram of the physical system. Each subsheet explains the implemantion enclosed therein.

Bibliography

This design was informed by NXP's AN4327 appnote, TI's SLAA655, SLAA274, and SLAA458 appnotes, and TI's TIDA−00311 reference design.

How it works

POxiM is based on the tranmissive pulse oximetry principle. This means that it determines the patient's heart rate and blood oxygen concentration (SPO2) by measuring the portion of red (660nm) and infrared (940nm) light that passes through their finger.

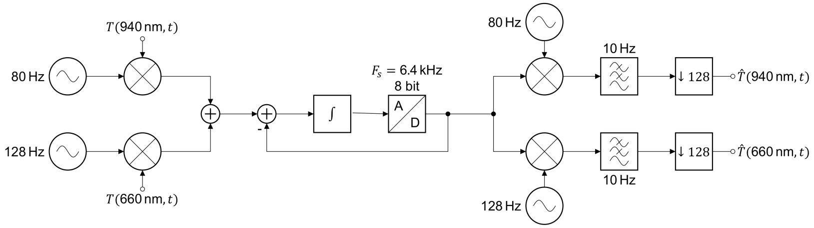
This is enabled by two effects. First, the optical properties of hemoglobin change when it bonds with oxygen, so that blood with a high oxygen saturation has a different absorption ratio between red and infrared light than blood with low oxygen saturation. Second, the pressure variations caused by heartbeats cause blood vessels to expand and contract, thus changing the amount of absorption at both of these wavelengths.

A transmissive pulse oximeter has three main tasks:
- Estimating the transmissivity of 660nm and 940nm light through a finger
- Using this data to estimate the pulse rate and SPO2 of the patient.
- Presenting this data to the patient and/or caretaker.

The latter two are relatively easy. POxiM uses a microcontroller to calculate the heart rate and SPO2 from the transmissivity data using industry standard formulas. This microcontroller also drives the display.

Estimating the transmissivity is significantly more difficult. The information is carried not only in the average DC component of the intensity, but also in the changing AC component of the signal, which has an amplitude of only 0.1% of the DC signal. Hence, measuring the signal requires a large dynamic range.

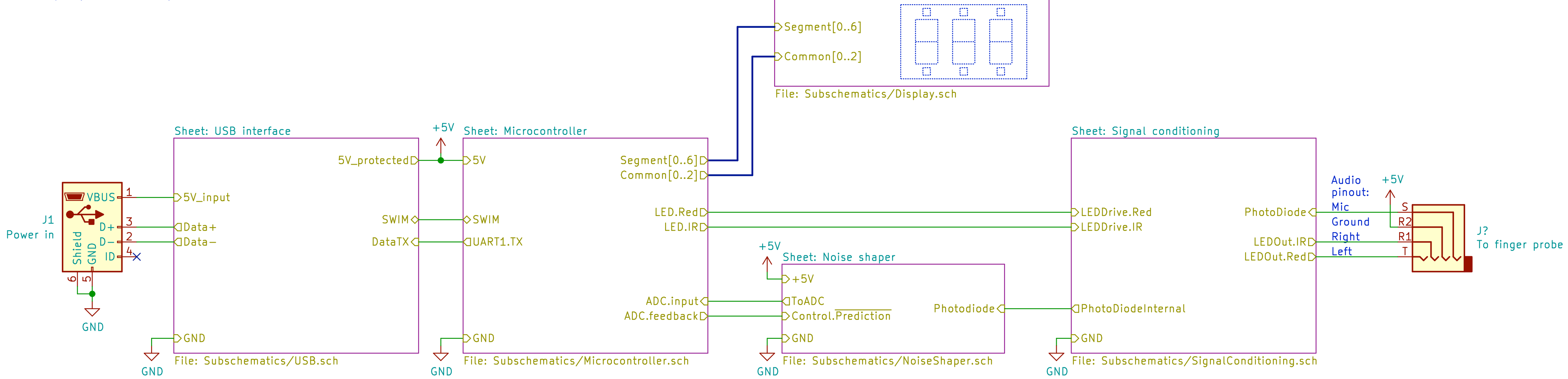
This dynamic range is achieved through oversampling combined with noise shaping, so that the design can be adapted to most microcontrollers. Oversampling spreads the ADC noise over a larger bandwidth frequencies, and noise shaping uses feedback to form a filter which suppresses this noise in the bandwidth of interest, at the expense of higher noise beyond the bandwidth of interest. Digital filters are then used to filter away the noise beyond the bandwidth of interest.



There is only a single optical channel to avoid the cost of optical filters and multiple channels. This introduces a problem, since the channel must be shared between the two wavelengths. Since the noise shaping circuitry is not compatible with sharing the channel in time, the channel is shared in the frequency domain. This is achieved by modulating each wavelength at a unique carrier frequency, and demodulating in software. This is the same principle used in AM radio.

Finally, different skin types require a system dynamic range beyond the dynamic range required to measure the transmissivity. This is achieved by varying the amplitude of the modulating signal to achieve brightness control.

The physical system



Mechanical

H1 MountingHole
H3 MountingHole
H5 AlignmentHole
H2 MountingHole
H4 MountingHole
H6 AlignmentHole

Power budget

There is 100mA available from a single power unit of USB. It is allocated as follows:
- Microcontroller: 5mA
- Readout circuit amplifier: 1mA
- Readout LEDs: 2 · 10mA = 20mA
- Display: 74mA
  - Digits: 7 · 10mA = 70mA
  - Transistor biasing: 4mA

TUDelft
Pulse/Oximeter measurement unit
By Arthur Admiraal & Daan de Groot
POxiM

Sheet: /
File: POxiM−wristcomputer.sch

Title: POxiM wrist computer

Size: A4 | Date: 2020−06−10 | **Rev: A**
KiCad E.D.A. kicad (5.1.0−0) | Id: 1/6

Working principle

This is an integrating differential pulse–code modulation quantiser inspired by a delta–sigma ADC and based directly on the lecture 'Sigma–Delta Modulation' of the TU Delft EE2S31 course on Signal Processing. See also: https://cas.tudelft.nl/Education/courses/ee2s31/

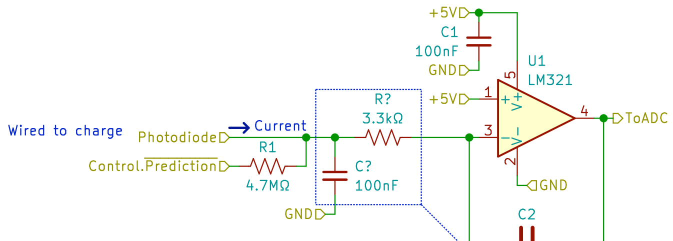
Together with the firmware, the goal of this circuit is to redistribute the ADC noise (both quantisation and otherwise) away from the low frequencies of interest, and towards higher frequencies. This enables a digital low–pass filter to nearly eliminate ADC noise, so that a much better effective resolution can be achieved (from 10 bits to 20 bits).

Dimensioning

As a rule of thumb, noise shaping of order p and filtering to 1/(2^n) of the nyquist bandwidth gives:
$\Delta SQNR = n \cdot (3 + 6p)$ dB
Whereas additional bits of effective resolution give:
$\Delta SQNR = B \cdot 6.02$ dB
Assuming a common 10–bit ADC with a conservative 8 bits of effective resolution, using the first order noise–shaping implemented here and equating for 9 additional bits gives:
$9 \cdot 6.02 \text{ dB} < n \cdot (3 + 6 \cdot 1) \text{ dB}$
$n = 9 \cdot 6.02 / 9 \approx 6.03$
Since sample numbers which are a power of two are easier to work with, we round to n=6. Then the sample rate must be at least 2^6 = 64 times higher than the nyquist frequency. The bandwidth of interest is 10Hz, so the nyquist frequency is 20Hz, and the sample frequency must thus be at least 1.28kHz. However, this is only the sample frequency per channel. Since there are two channels, the system sample rate will be twice as high, or 2.56kHz. We sacrifice a little bandwidth for a nice 16MHz clock division at 2.5kHz.

From measurements, the minimum full scale input current is 1µA. The integration capacitor must be charged up to the full scale ADC voltage with a full scale input current within one period time. Then:
$Q = C \cdot V \Rightarrow C = I \cdot T / V = I / (V \cdot f) = 1\mu A / (2V \cdot 2.5kHz) = 800pF$

Nonidealities which may mess up measurement:
- Capacitor leakage
- Controller noise (voltage & current)
- Resistor noise (voltage & current)
- Stability?
- Bandwidth?
- Aliasing?



86pF

This filter decreases the effect of the jitter in the prediction PWM as a noise source. Furthermore, it limits the amplitude of the fundamental frequency of the PWM fed into the opamp, which decreases nonlinearities in the opamp creating intermodulation distortion and mixing down crosstalk at high frequencies to the band of interest.

As long as the cutoff lies outside the band of interest, the noise shaping there isn't significantly affected. However, the more aggresive the filter, the longer it takes for the filtered prediction to catch up to the photodiode signal. As a result, the gain of the integrator must be reduced to prevent clipping. This does increase the noise in the band of interest.

Things to analyse:
- ! Input noise
  - ! Find out difference between input referred noise and ADC noise
  - ! Find input referred current noise
  - ! Find ADC referred noise
- ! MOSFET threshold voltage
  - ! Find maximum output voltage from MOSFET threshold voltage
  - ! Find whether MOSFET threshold voltage hinders startup
- ! Aliasing?
  - ! Find out the effects of aliasing at input
  - ! Find out the effects of aliasing at ADC
- ! Capacitor leakage
  - ! Linear scaling or additional noise?
- ! Linearity
  - ! Effect of ADC nonlinearity on overall linearity?
- ! Stability
  - ! Small–signal model at operating point
  - ! Find bandwidth
  - ! Find gain and phase margin
  - ! Possible stability mitigations

Input noise
Feedback resistor noise and input referred controller current noise directly impact the input SNR. The resistor noise can be made arbitraily small by using a voltage divider and lower resistor value. The controller voltage noise only adds to the ADC noise since it isn't integrated as the current noise, and is thus noise shaped away.

MOSFET threshold voltage
Maximum output voltage is Vo = Vcc − (Vth + 2 · Vi). At startup, the capacitors have Vc=0V, so the MOSFETs nicely conduct. When the input clips, the capacitors are charged to the maximum output voltage. As soon as the input current drops within the minimum range, the feedback resistor will discharge them.

Aliasing
Noise before the integrator is integrated and then sampled at the ADC sample frequency. Noise after the integrator is directly sampled at the ADC sample frequency. Hence, noise will be aliased down at all points in the system by the ADC sample frequency. However, the limited bandwidth of the integrator will act as an anti–aliasing filter for noise before the integrator. In absence of a dedicated anti–aliasing filter after the integrator, the only limit to aliasing will be the bandwidth of the ADC itself.

Channel bleeding

Capacitor leakage
According to https://www.murata.com/en-eu/support/faqs/products/capacitor/mlcc/char/0039, ceramic capacitors generally have an isolation resistance of >10GΩ, so for voltages <3.3V, this gives currents of I = 3.3V / 10 = 0.33nA

TUDelft
Trades bandwidth to increase resolution
By Arthur Admiraal & Daan de Groot
POxIM

POXIM

Sheet: /Noise shaper/
File: NoiseShaper.sch

Title: Noise shaping

Size: A4	Date: 2020-06-10	**Rev: A**
KiCad E.D.A. kicad (5.1.0-0)		Id: 2/6

All pins are designed to survive three scenarios:
- Hotplugging of the 4 pole 3.5mm jack
- Short-circuits to ground
- Application of line audio

When the 3.5mm jack is hotplugged, the worst that could happen is that outputs short together or to ground. Hence, to satisfy this requirement, the inputs need to handle short-circuits to voltages between 0V and 5V. This also satisfies the short-circuit to ground requirement.

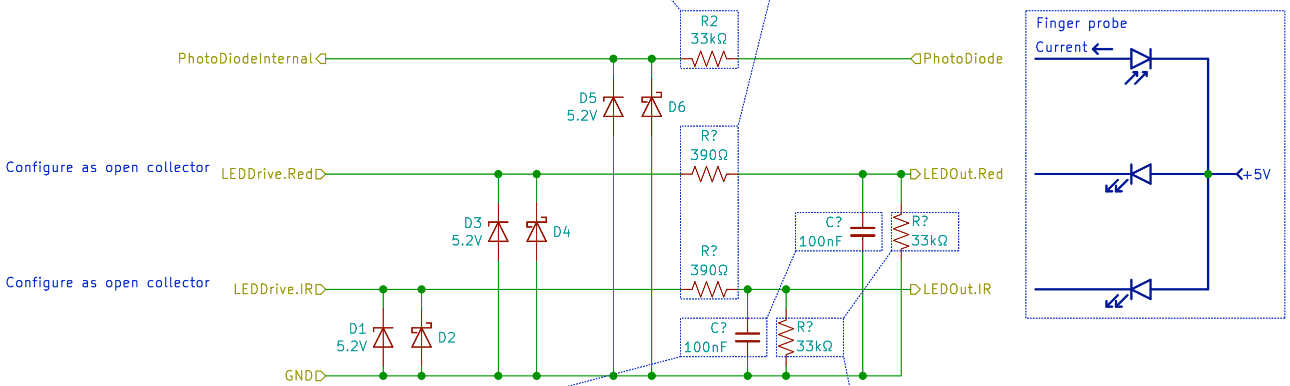
The voltage level of common 3.5mm headphone jacks is <2V in amplitude (see also https://electronics.stackexchange.com/questions/28404/what-is-the-voltage-range-of-a-standard-headphone-jack-from-a-phone). Since the ground terminal is connected to 5V, this means all pins need to handle shorts to voltages between 5V-2V=3V and 5V+2V=7V.

Hence, all three requirements are satisfied if all pins can handle short-circuits to voltages between 0V and 7V.

Must limit the current on short-circuits between 0V and 7V. A current through the microcontroller is attained for a 7V short when the pin is driven low. From the maximum I/O sink current of 20mA:
Rmin = Vs / Imax = 7V / 20mA = 350Ω
Selecting the first E12 value above this minimum gives R = 390Ω.
In this condition, the resistor will dissipate no more than:
P = V² / R = 7V² / 390Ω = 126mW, which is acceptable.

Since the forward voltage over red and infared LEDs is typically between 1V and 2V, the voltage at the cathode of the LEDs will lie between Vs-Vf ∈ 5V - [1, 2]V = [3, 4]V. This is within the protected voltage range. The current will range between:
I = (Vs - Vf) / R ∈ (5V - [1, 2]V) / 390Ω = [7.7, 10.3]mA
Which is sensible.

Limits input current compared on overvoltages and undervoltages. Increases input impedance, but an insignificant amount compared to the source impedance formed by the photodiode.



Filter LED PWM harmonics above:
f = 1 / (2π · R · C) =
= 1 / (2π · 390Ω · 100nF) = 4.1kHz
Prevents sourcing significant power at frequencies where the cable is a good antenna.

Biases LEDs with a small current, such that the voltage over them stays roughly constant. The point of this is to decrease the amplitude of the voltage variations in the cable, so that interference through capacitive coupling is greatly reduced. This decreases crosstalk by ~15dB. Inductive coupling can't be avoided, since the current must be varied to create brightness variations in the LED.

The biasing doesn't work if the LED drive pins force no voltage over de LED when it isn't driven. Hence, they must be configured as open collector.

The bias current is I = (Vs - Vf) / R = (5V - 1.85V) / 33kΩ = 0.1mA. This is comparable to the minimum brightness setting of
Imin = Ifs / N = 20mA / 166 = 0.12mA, so it won't hamper brightness scaling.

TUDelft

Performs filtering, prevents ESD damage

By Arthur Admiraal & Daan de Groot

POxIM

POXiM

Sheet: /Signal conditioning/
File: SignalConditioning.sch

Title: Audio jack signal conditioning

Size: A4 | Date: 2020-05-29 | Rev: A
KiCad E.D.A. kicad (5.1.0-0) | Id: 3/6

Operation

The LEDs are multiplexed in digits. There are three numeric digits, 0 to 2, and a single auxiliary digit. Setting a single common pin high selects the digits with the corresponding number. Setting all the common pins low selects the auxiliary digit. See also the segment map and truth table.

Why buffering?

I/O pins directly drive the segments. Transistors are used as buffers to sink and source the common currents. It would be even simpler to directly use I/O pins, but we found that the maximum average current of $I = I_{o,max} / N_{led} = 20mA / (4 \cdot 7) = 0.71mA$ per LED does not give enough brighness for good readability. Buffering the common currents raises the maximum average current to $I = I_{o,max} / N_{digit} = 20mA / 4 = 5mA$, which is more than the maximum available current through the 100mA fuse.

Power budget

There is 74mA available current for the display. Of this, $7 \cdot 10mA$ is allocated to the LEDs, and 4mA to driving the buffer transistors.

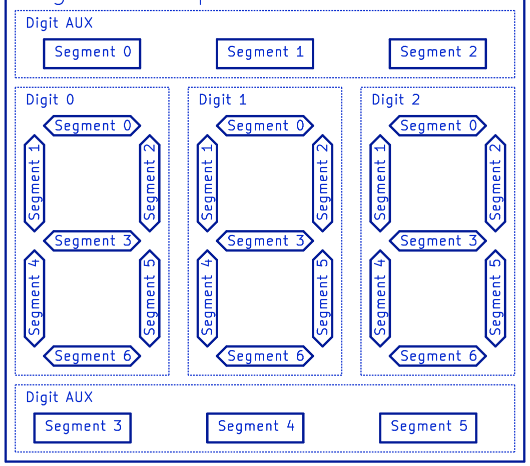
Why bipolar buffers?

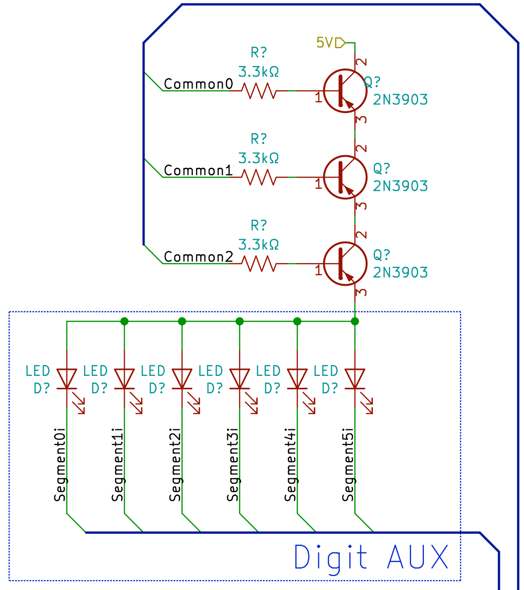
There are many other possibilities to buffering the common currents. MOSFETs would be significantly less power hungry than bipolar transistors. The reason we specified bipolar transistors, is that a circuit topology optimised for bipolar transistors (with base resistors) is compatible with MOSFETs, but a MOSFET-optimised topology (without base resistors) would not be compatible with bipolar transistors, thus restricthing the pool of suitable parts.

Valid states

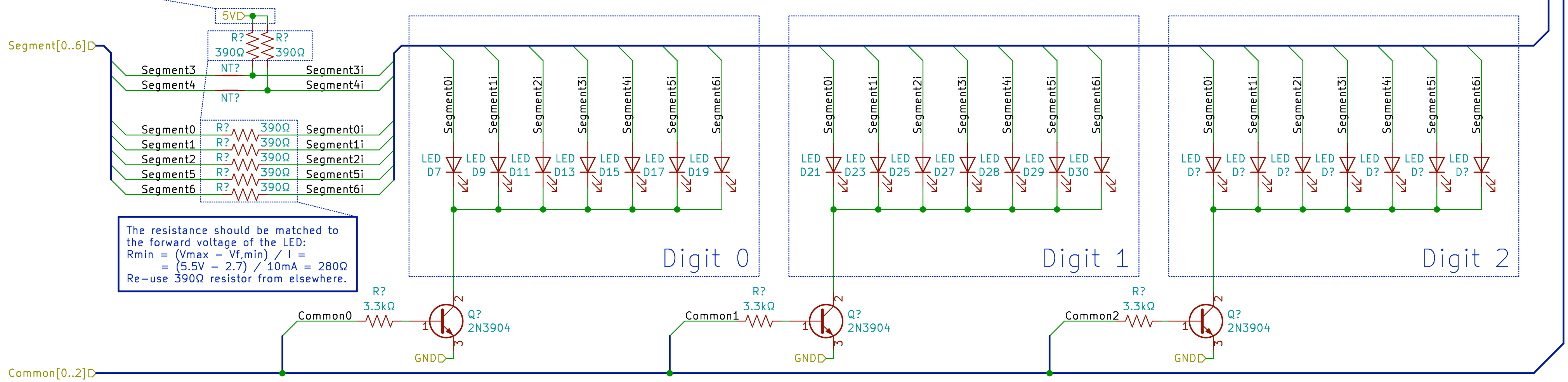
Common 0	Common 1	Common 2	Selected digit
1	0	0	Digit 0
0	1	0	Digit 1
0	0	1	Digit 2
0	0	0	Digit AUX

Segment map





Segment 3 and 4 are open-drain only, so pullups are used. Play close attention to the supply path. It is a potential source of EMI issues, since high-frequency currents flow through here.



The resistance should be matched to the forward voltage of the LED:
$R_{min} = (V_{max} - V_{f,min}) / I = (5.5V - 2.7) / 10mA = 280\Omega$
Re-use 390Ω resistor from elsewhere.

TUDelft

3 7-segment digits and indicator LEDs

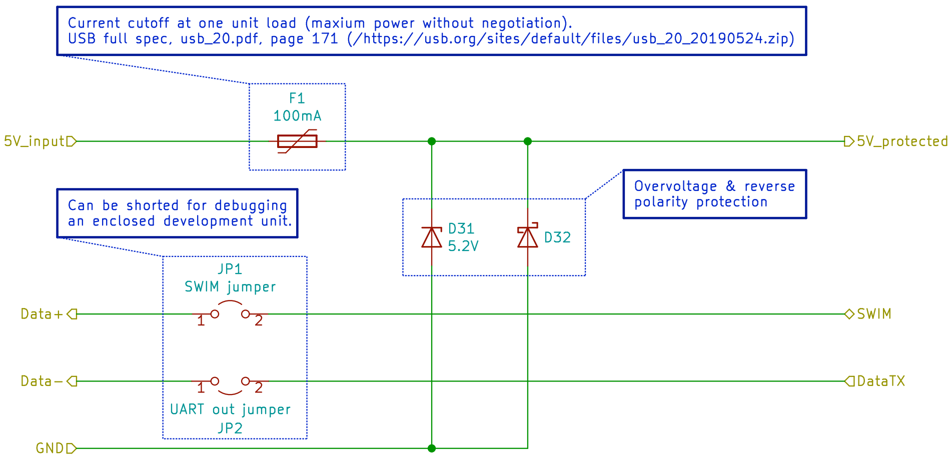
By Arthur Admiraal & Daan de Groot

POxIM

Sheet: /Display/
File: Display.sch

Title: Discrete LED display

Size: A4	Date: 2020-06-13	**Rev: A**
KiCad E.D.A. kicad (5.1.0-0)		Id: 4/6



TUDelft

Prevents excessive voltages and currents

By Arthur Admiraal & Daan de Groot

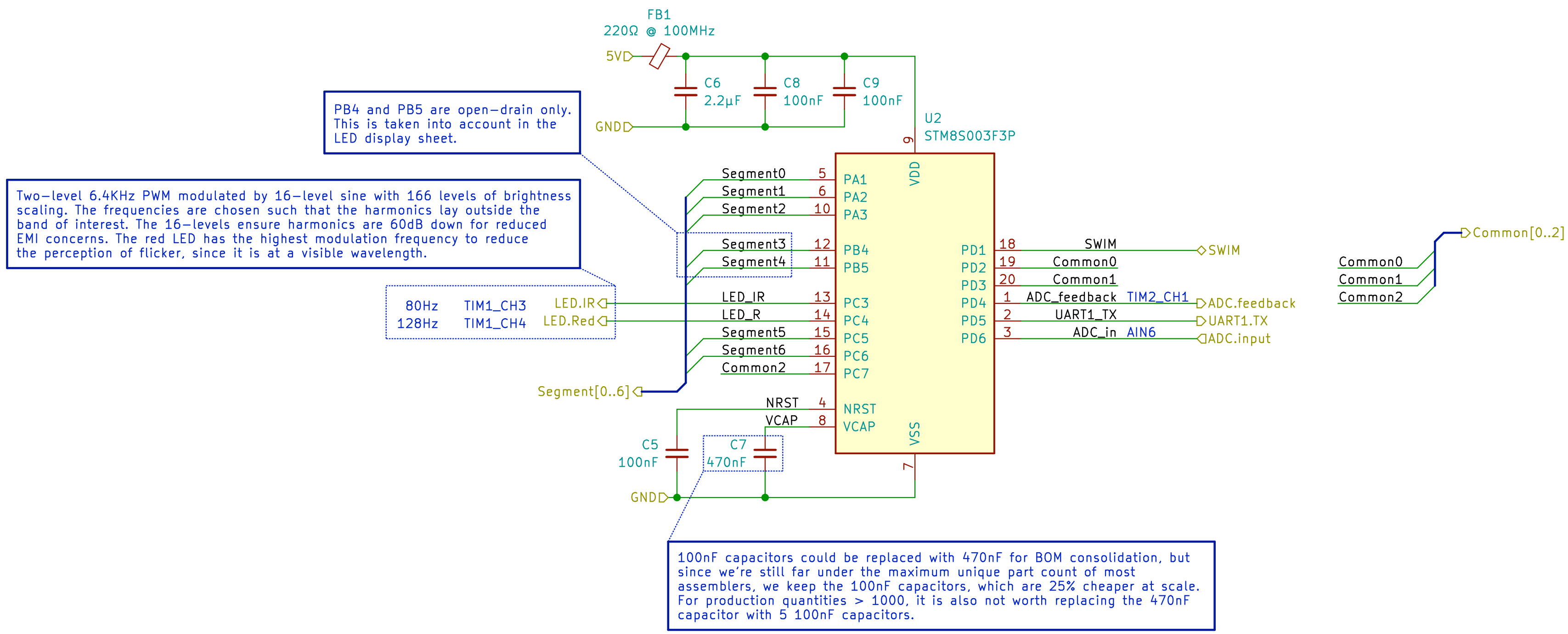
POxIM

Sheet: /USB interface/

File: USB.sch

Title: USB interfacing

Size: A4	Date: 2020-06-10	**Rev: A**
KiCad E.D.A. kicad (5.1.0-0)		Id: 5/6



TU Delft

Performs signal processing and control

By Arthur Admiraal & Daan de Groot

POxIM

POXIM

Sheet: /Microcontroller/

File: Microcontroller.sch

Title: Microcontroller

Size: A4	Date: 2020-06-10	**Rev: A**
KiCad E.D.A. kicad (5.1.0-0)		Id: 6/6