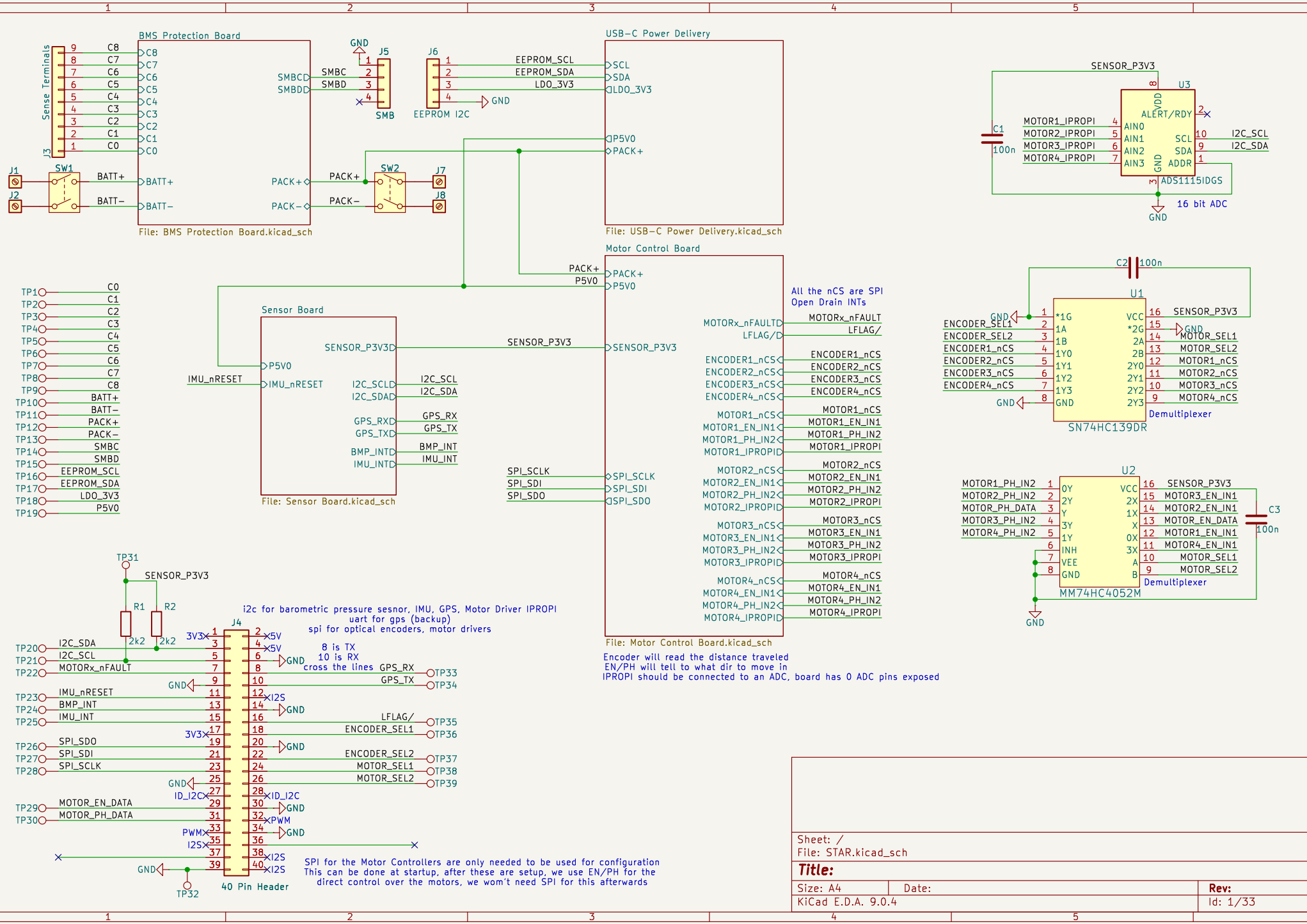
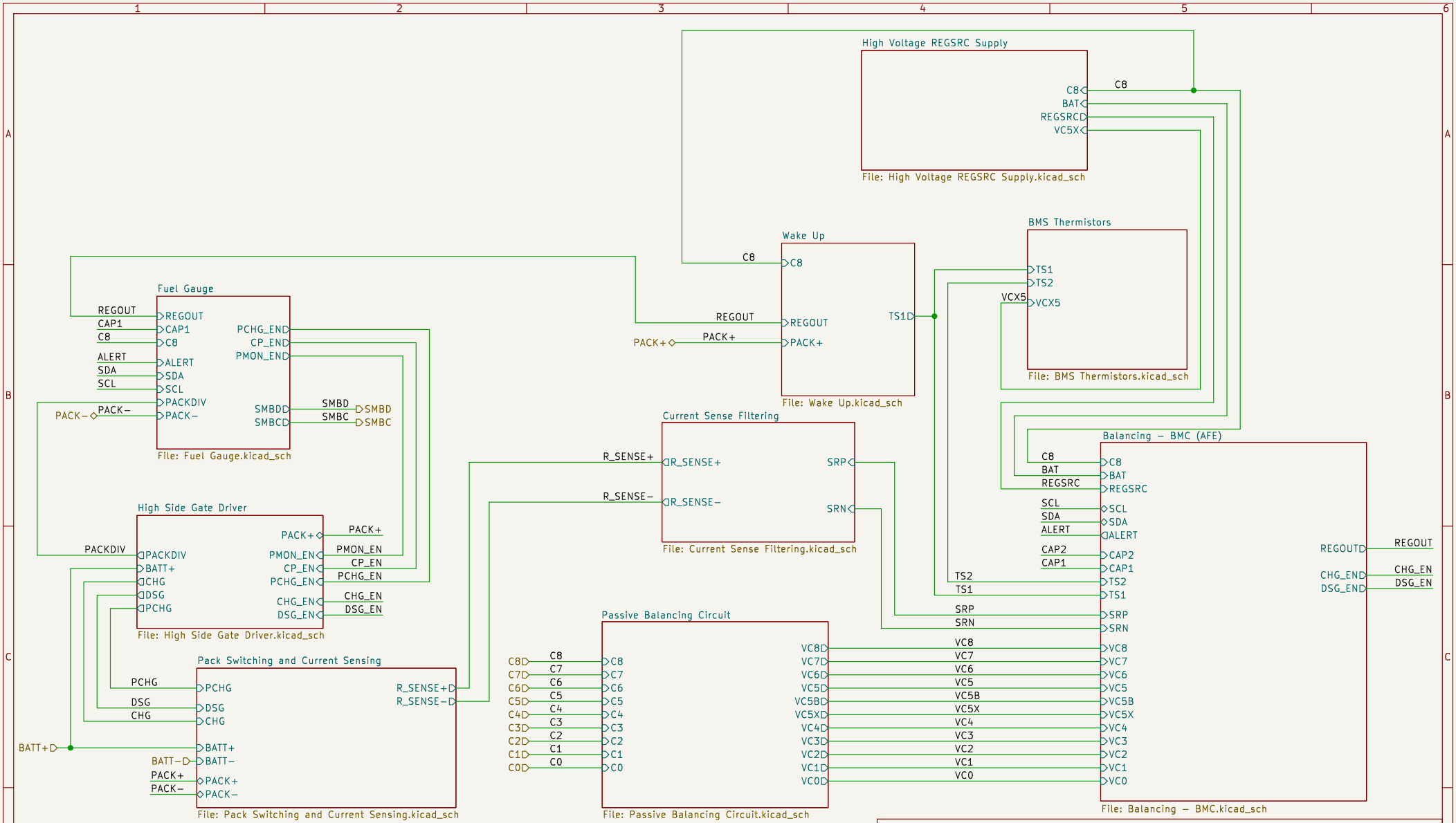

BMS Protection Board
Sense Terminals
J3
C8
C7
C6
C5
C4
C3
C2
C1
C0
SMBC
SMBD
GND
J5
SMB
J6
EEPROM I2C
EEPROM_SCL
EEPROM_SDA
LDO_3V3
GND
J1
J2
SW1
BATT+
BATT−
PACK+
PACK−
SW2
J7
J8
File: BMS Protection Board.kicad_sch
USB−C Power Delivery
SCL
SDA
LDO_3V3
P5V0
PACK+
File: USB−C Power Delivery.kicad_sch
SENSOR_P3V3
U3
VDD
ALERT/RDY
MOTOR1_IPROPI
MOTOR2_IPROPI
MOTOR3_IPROPI
MOTOR4_IPROPI
AIN0
AIN1
AIN2
AIN3
GND
SCL
SDA
ADDR
I2C_SCL
I2C_SDA
C1
100n
ADS1115IDGS
16 bit ADC
TP1
TP2
TP3
TP4
TP5
TP6
TP7
TP8
TP9
TP10
TP11
TP12
TP13
TP14
TP15
TP16
TP17
TP18
TP19
Sensor Board
SENSOR_P3V3
P5V0
IMU_nRESET
I2C_SCL
I2C_SDA
GPS_RX
GPS_TX
BMP_INT
IMU_INT
File: Sensor Board.kicad_sch
Motor Control Board
PACK+
P5V0
SENSOR_P3V3
SPI_SCLK
SPI_SDI
SPI_SDO
All the nCS are SPI
Open Drain INTs
MOTORx_nFAULT
LFLAG/
ENCODER1_nCS
ENCODER2_nCS
ENCODER3_nCS
ENCODER4_nCS
MOTOR1_nCS
MOTOR1_EN_IN1
MOTOR1_PH_IN2
MOTOR1_IPROPI
MOTOR2_nCS
MOTOR2_EN_IN1
MOTOR2_PH_IN2
MOTOR2_IPROPI
MOTOR3_nCS
MOTOR3_EN_IN1
MOTOR3_PH_IN2
MOTOR3_IPROPI
MOTOR4_nCS
MOTOR4_EN_IN1
MOTOR4_PH_IN2
MOTOR4_IPROPI
File: Motor Control Board.kicad_sch
Encoder will read the distance traveled
EN/PH will tell to what dir to move in
IPROPI should be connected to an ADC, board has 0 ADC pins exposed
C2
100n
U1
*1G
*2G
1A
1B
1Y0
1Y1
1Y2
1Y3
2A
2B
2Y0
2Y1
2Y2
2Y3
VCC
ENCODER_SEL1
ENCODER_SEL2
ENCODER1_nCS
ENCODER2_nCS
ENCODER3_nCS
ENCODER4_nCS
MOTOR_SEL1
MOTOR_SEL2
MOTOR1_nCS
MOTOR2_nCS
MOTOR3_nCS
MOTOR4_nCS
Demultiplexer
SN74HC139DR
U2
0Y
2Y
Y
3Y
1Y
INH
VEE
0X
3X
A
B
MOTOR1_PH_IN2
MOTOR2_PH_IN2
MOTOR_PH_DATA
MOTOR3_PH_IN2
MOTOR4_PH_IN2
MOTOR3_EN_IN1
MOTOR2_EN_IN1
MOTOR_EN_DATA
MOTOR1_EN_IN1
MOTOR4_EN_IN1
C3
100n
MM74HC4052M
TP31
R1
R2
2k2
2k2
i2c for barometric pressure sesnor, IMU, GPS, Motor Driver IPROPI
uart for gps (backup)
spi for optical encoders, motor drivers
J4
3V3
5V
I2S
ID_I2C
PWM
8 is TX
10 is RX
cross the lines GPS_RX
GPS_TX
LFLAG/
ENCODER_SEL1
ENCODER_SEL2
MOTOR_SEL1
MOTOR_SEL2
MOTOR_EN_DATA
MOTOR_PH_DATA
TP20
TP21
TP22
TP23
TP24
TP25
TP26
TP27
TP28
TP29
TP30
TP32
TP33
TP34
TP35
TP36
TP37
TP38
TP39
40 Pin Header
SPI for the Motor Controllers are only needed to be used for configuration
This can be done at startup, after these are setup, we use EN/PH for the
direct control over the motors, we wom't need SPI for this afterwards
Sheet: /
File: STAR.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 1/33



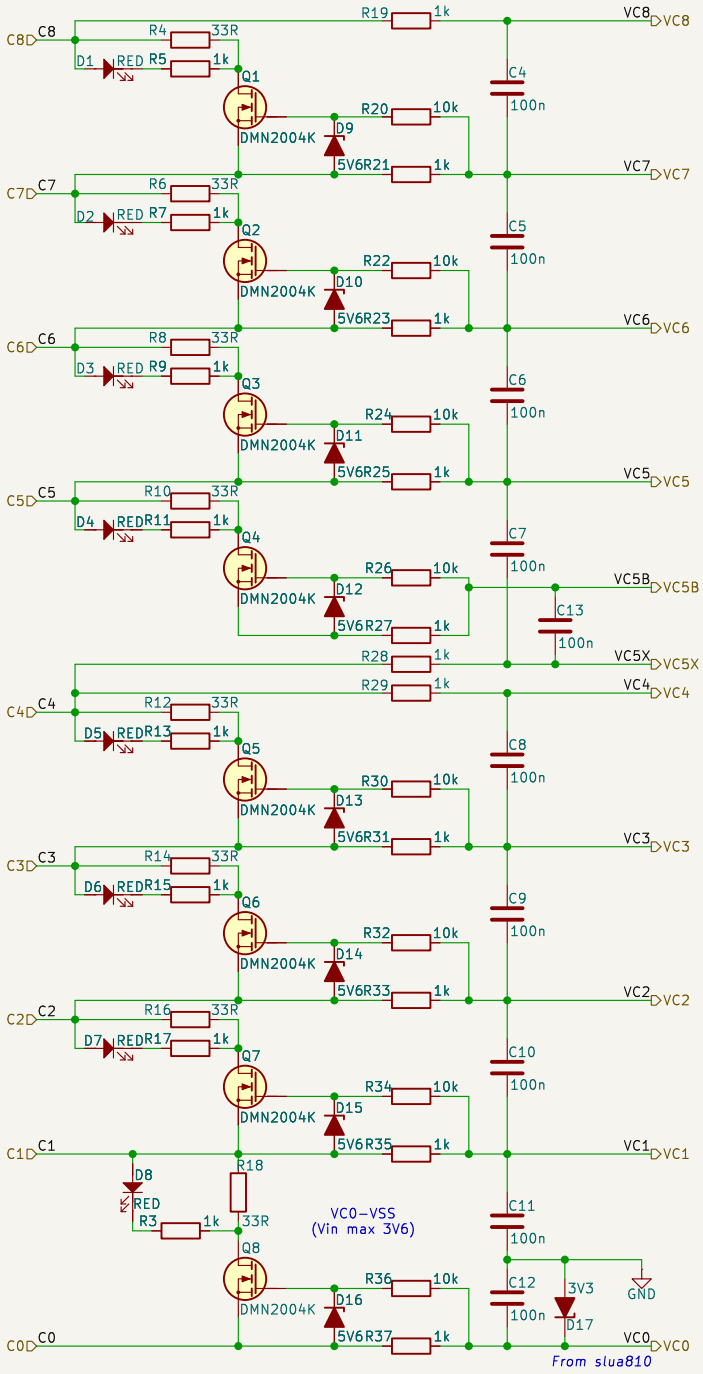
BMS Protection Board for 8S Battery Pack
In/Out: min voltage: 25.6V, nominal voltage: 29.6V, max voltage: 33.6V
Output: 50A continuous discharge current
Min Each Cell set to 3.2V, Max Each Cell set to 4.2V

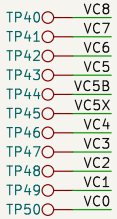
Sheet: /BMS Protection Board/
File: BMS Protection Board.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 2/33

Passive Balancing Circuit

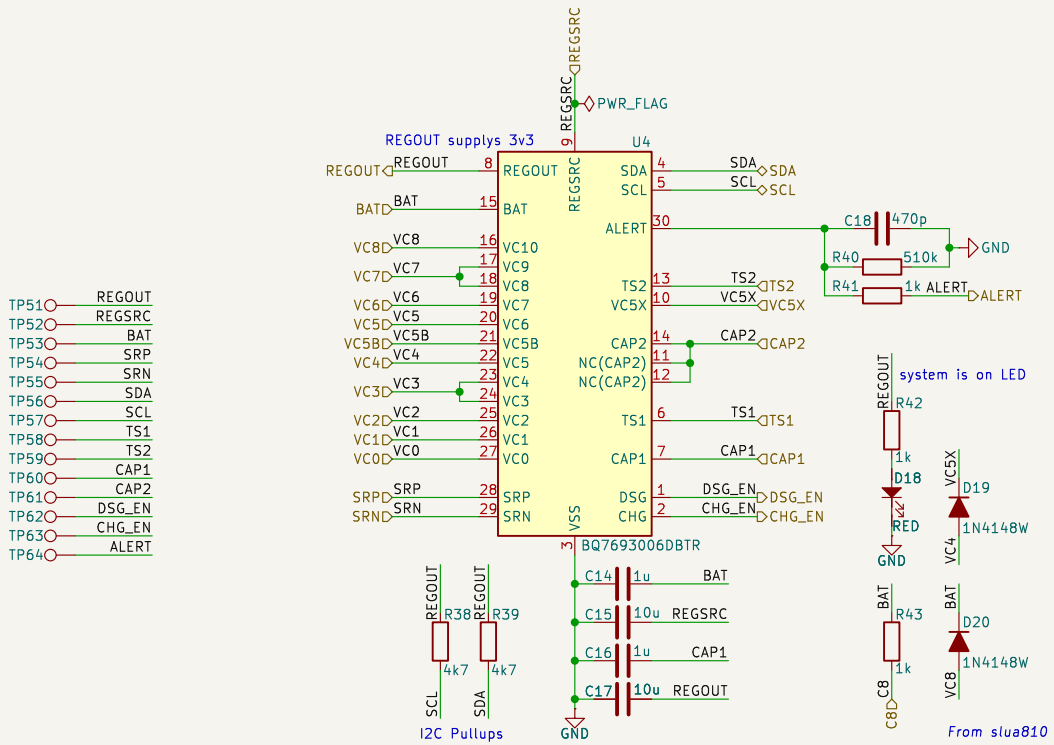




Sheet: /BMS Protection Board/Passive Balancing Circuit/
File: Passive Balancing Circuit.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 3/33



Balancing & BMC (AFE)

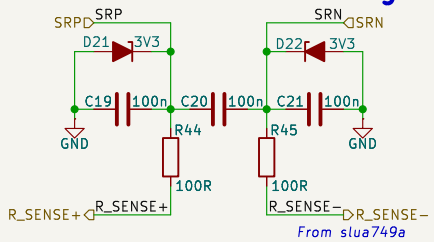
TP51	REGOUT
TP52	REGSRC
TP53	BAT
TP54	SRP
TP55	SRN
TP56	SDA
TP57	SCL
TP58	TS1
TP59	TS2
TP60	CAP1
TP61	CAP2
TP62	DSG_EN
TP63	CHG_EN
TP64	ALERT

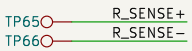
Sheet: /BMS Protection Board/Balancing – BMC (AFE)/
File: Balancing – BMC.kicad_sch

Title:

Size: A4 | Date: | **Rev:**
KiCad E.D.A. 9.0.4 | Id: 4/33

Current Sense Filtering

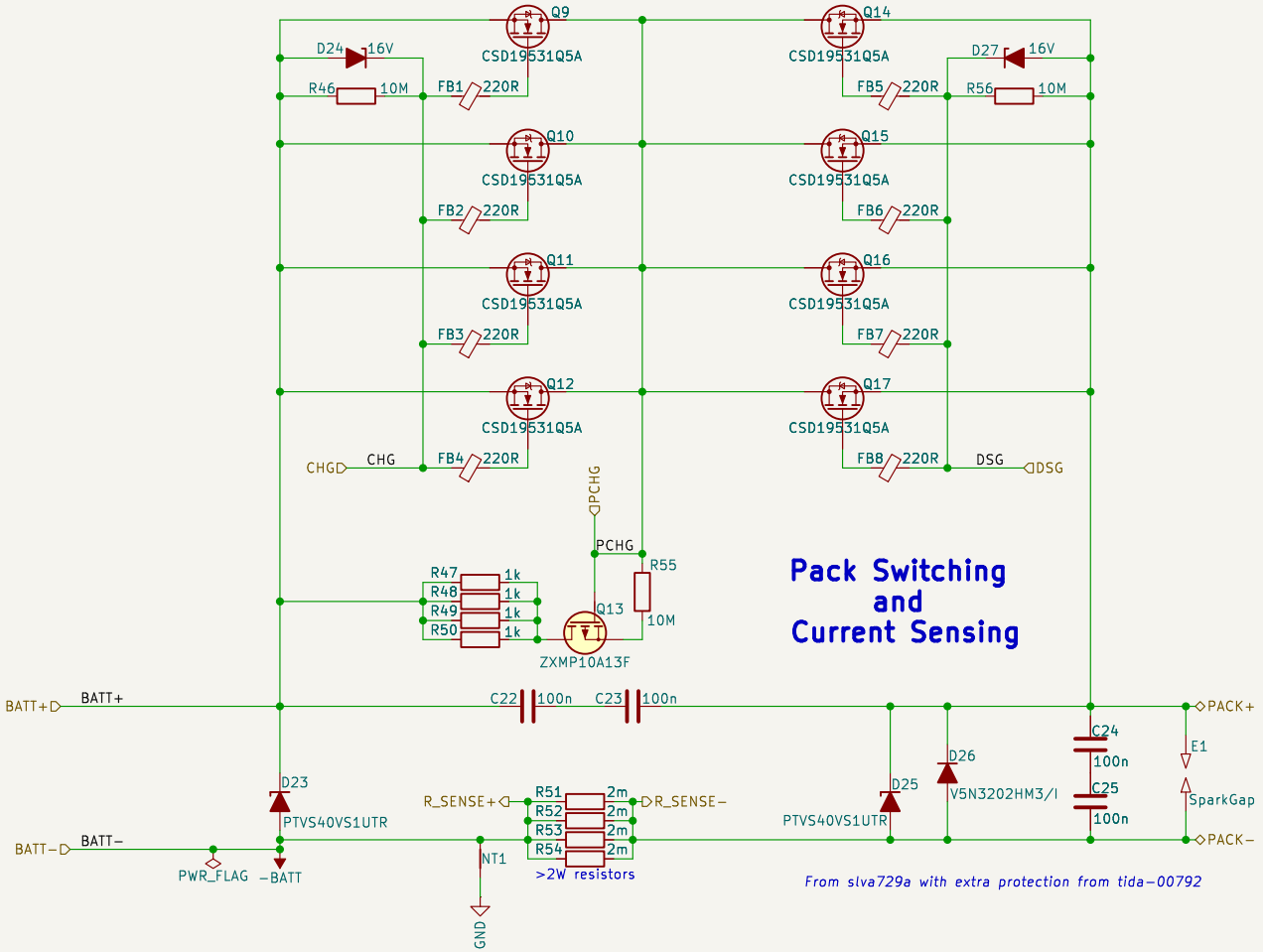




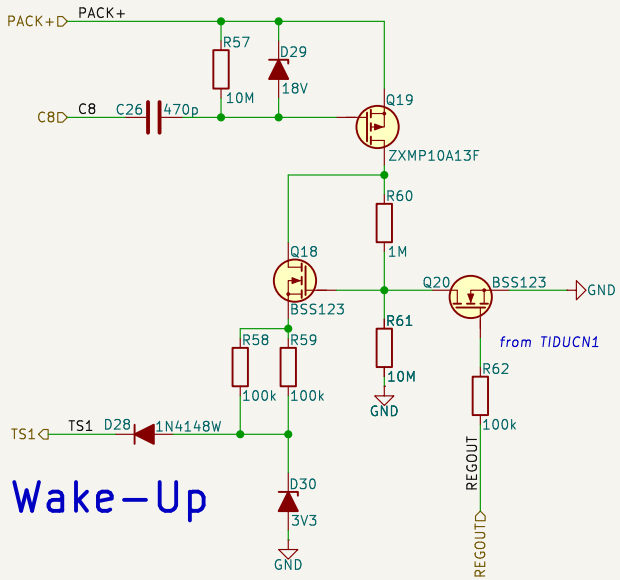
Sheet: /BMS Protection Board/Current Sense Filtering/
File: Current Sense Filtering.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 5/33

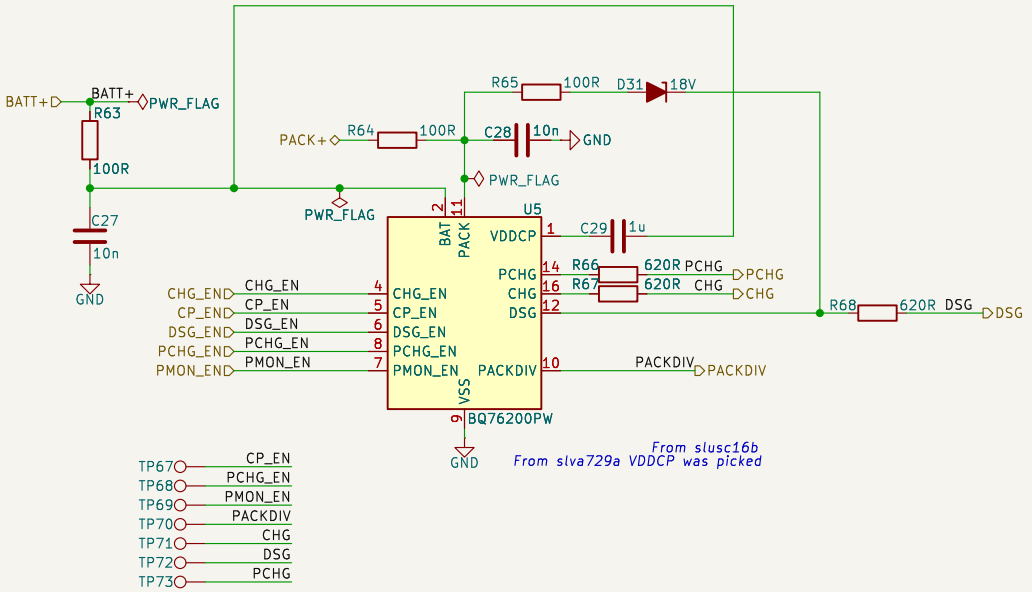


Sheet: /BMS Protection Board/Pack Switching and Current Sensing/		
File: Pack Switching and Current Sensing.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 6/33

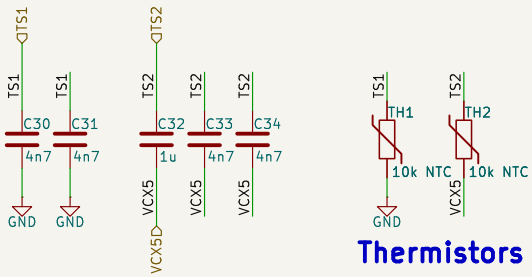


Sheet: /BMS Protection Board/Wake Up/		
File: Wake Up.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 7/33

High-Side Gate Driver

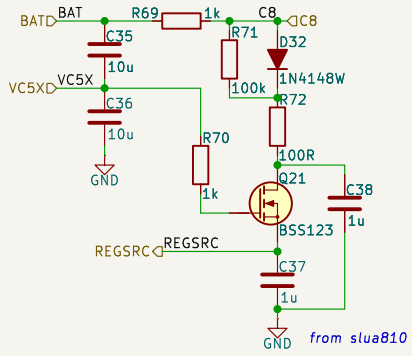


Sheet: /BMS Protection Board/High Side Gate Driver/		
File: High Side Gate Driver.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 8/33



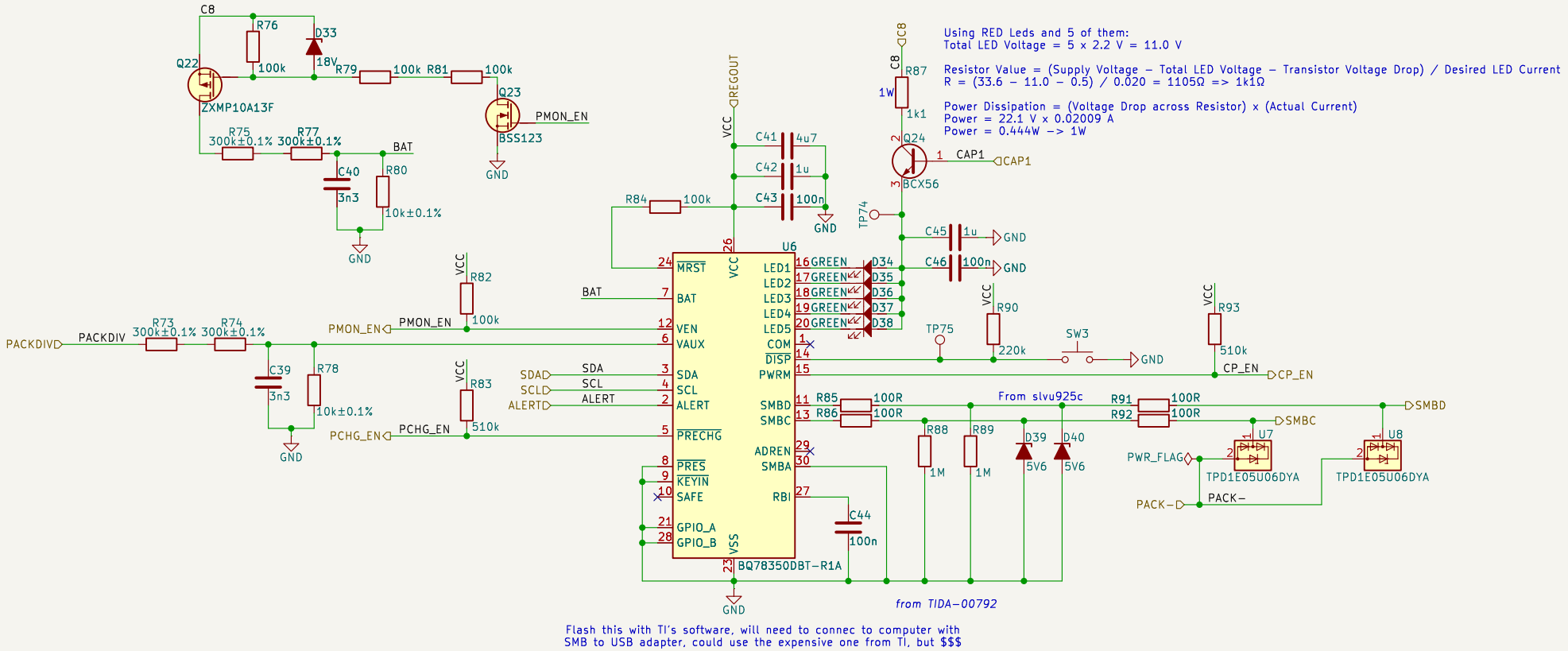
Sheet: /BMS Protection Board/BMS Thermistors/		
File: BMS Thermistors.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 9/33

High Voltage REGSRC Supply



Sheet: /BMS Protection Board/High Voltage REGSRC Supply/		
File: High Voltage REGSRC Supply.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 10/33

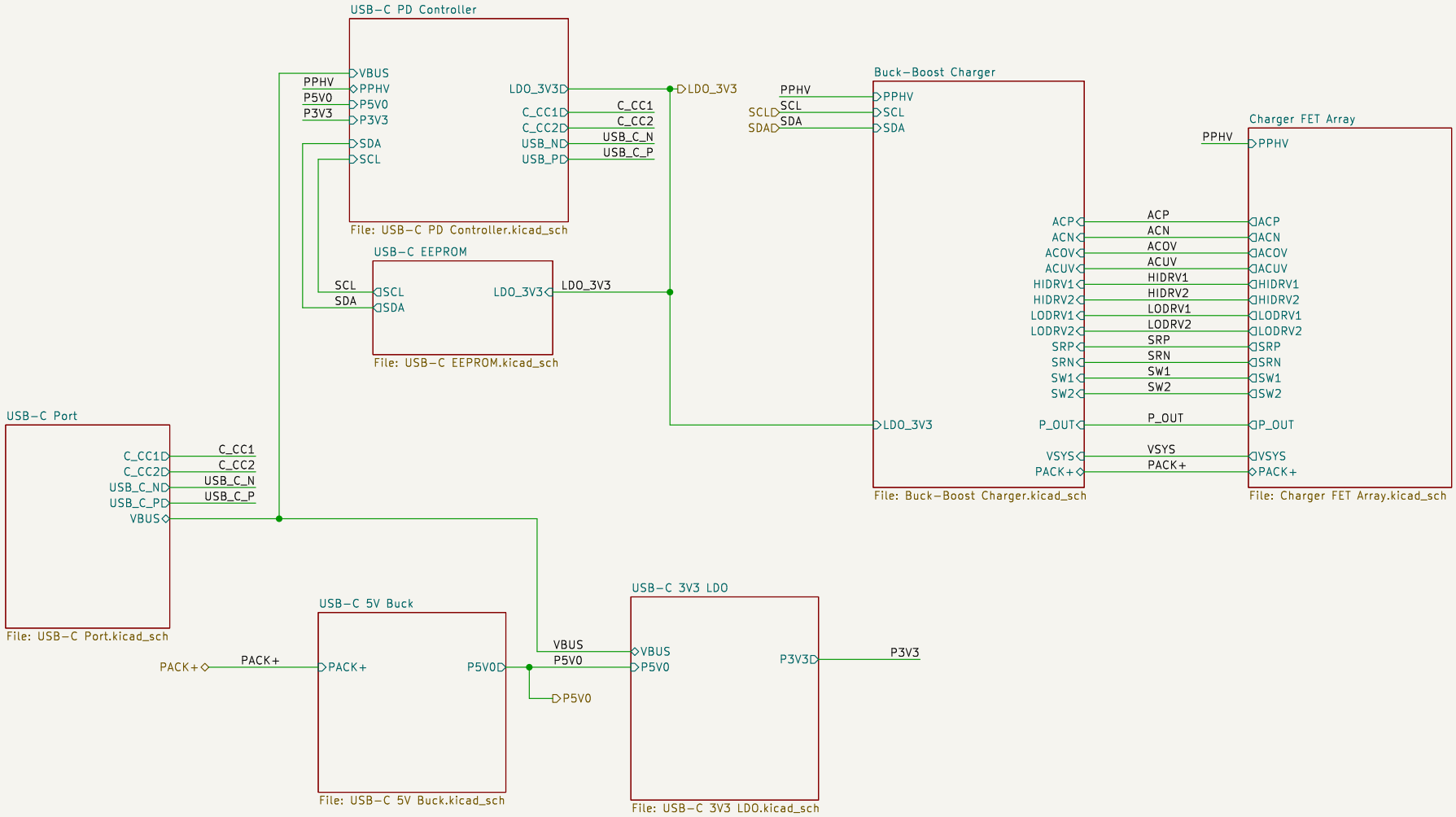
Fuel Gauge



Sheet: /BMS Protection Board/Fuel Gauge/
File: Fuel Gauge.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 11/33



Design is based off of PMP41062

The Correct Flashing Process
Here is the correct workflow:

Generate Your Configuration: Use the TPS2575x Application Customization Tool from Texas Instruments to create the binary file (.bin) that contains all your desired settings.

Program the EEPROM Separately: Connect the CAT24C512 EEPROM to your dedicated flasher or an Arduino. Do this before connecting it to the TPS25751.

Connect power (VCC), ground (GND), SDA, and SCL.

Use the flasher's software to write the .bin file you generated onto the EEPROM.

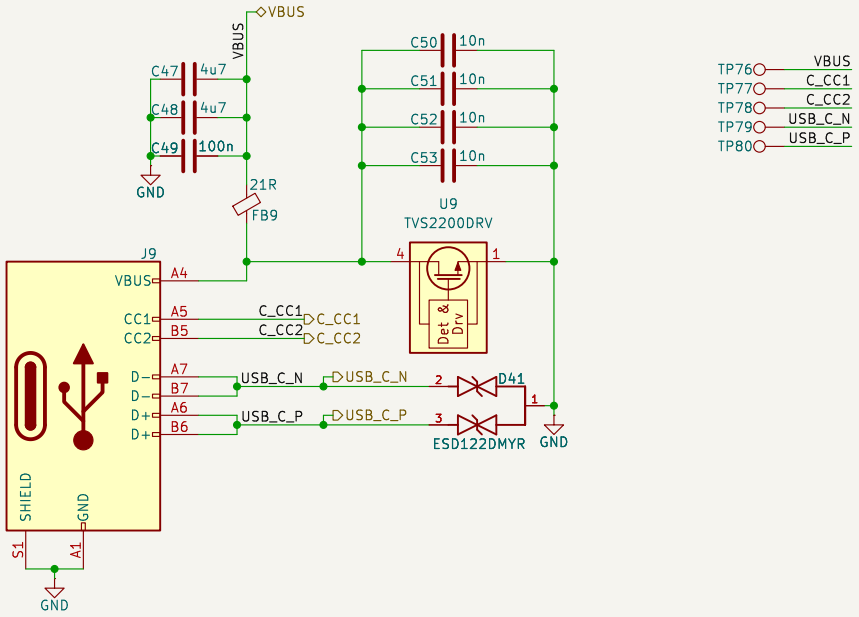
Assemble the Final Circuit: Once the EEPROM is successfully flashed, connect it to the I2Cc (the controller port) of the TPS25751.

Now, when you power on your circuit, the TPS25751 will use its I2Cc port to automatically read the settings you loaded onto the EEPROM and will configure itself correctly.

Sheet: /USB-C Power Delivery/
File: USB-C Power Delivery.kicad_sch

Title:

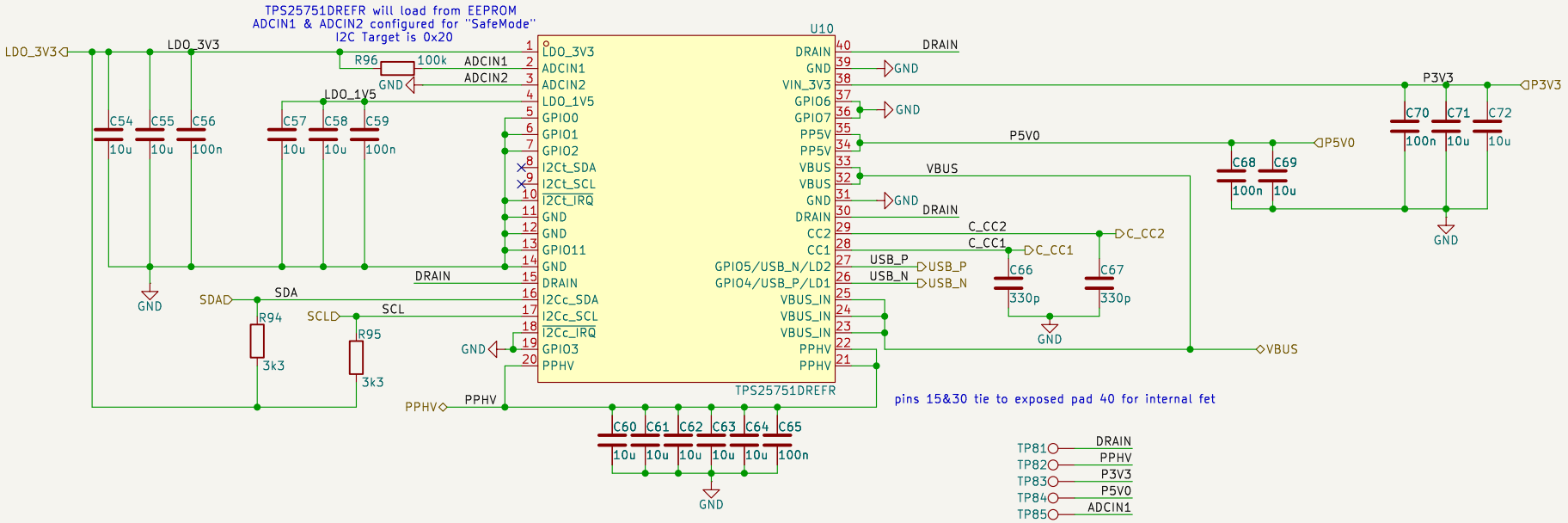
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 12/33



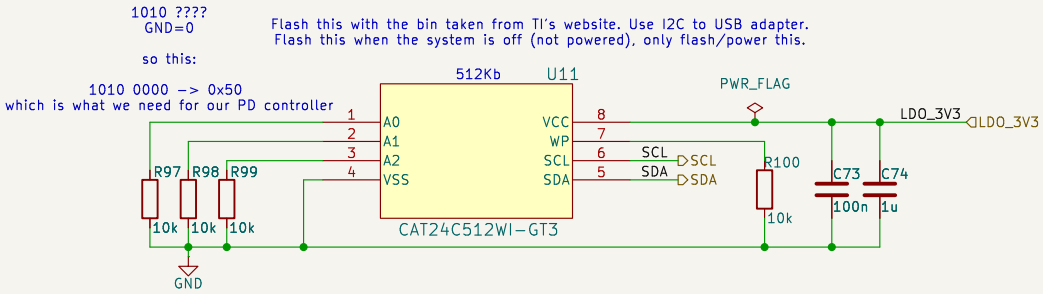
Sheet: /USB-C Power Delivery/USB-C Port/
File: USB-C Port.kicad_sch

Title:

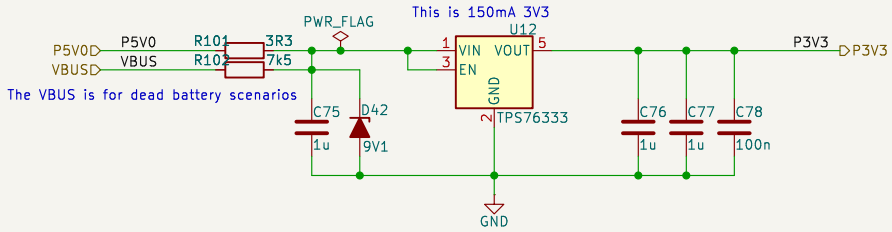
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 13/33



Sheet: /USB-C Power Delivery/USB-C PD Controller/		
File: USB-C PD Controller.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 14/33



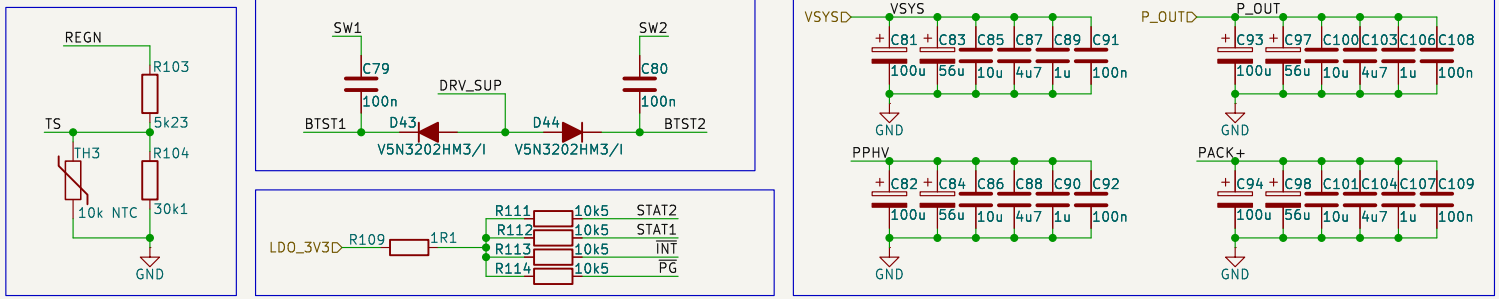
Sheet: /USB-C Power Delivery/USB-C EEPROM/		
File: USB-C EEPROM.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 15/33



Sheet: /USB-C Power Delivery/USB-C 3V3 LDO/
File: USB-C 3V3 LDO.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 16/33



these should be precision resistors

"PIN32 is for reverse mode voltage regulation"

Target Voltage = 8 cells * 4.2 V/cell = 33.6 V
Rtop = 249kΩ (from datasheet's recommendation)
Rbot = Btop * (Vfb / (Vbat_reg − Vfb)) + Rfbg
Vfb: 1.536V from datasheet
Rfbg: 33Ω from datasheet
Rbot = 11.96kΩ

PACK+ R106 249k±0.1%
R107 11k96±0.1%

ACP ACN SRP SRN FB FBG TS ICHG ILIM_HIZ ACUV ACOV FSW_SYNC (Set to 300kHz) CE

R105 11k36±0.1%
R108 2k49±0.1%
R110 100k±0.1%
R115 0R

U13 BQ25756RRVR

PWR_FLAG
PPHV
BTST1 BTST2 DRV_SUP HIDRV1 SW1 LODRV1 HIDRV2 SW2 LODRV2 REGN INT STAT1 STAT2 PG SCL SDA
C95 4u7
C99 1u
C102 1u
C105 100n
C96 4u7
GND

TP86 VSYS
TP87 P_OUT
TP88 ACP
TP89 ACN
TP90 SRP
TP91 SRN
TP92 FB
TP93 FBG
TP94 TS
TP95 ICHG
TP96 ILIM_HIZ
TP97 ACUV
TP98 ACOV
TP99 FSW_SYNC
TP100 CE
TP101 BTST1
TP102 BTST2
TP103 DRV_SUP
TP104 HIDRV1
TP105 SW1
TP106 LODRV1
TP107 HIDRV2
TP108 SW2
TP109 LODRV2
TP110 REGN
TP111 INT
TP112 STAT1
TP113 STAT2
TP114 SCL
TP115 SDA
TP116 PG

For my application:
8S4P setup, each cell is ~2200mAh
4 * 2200mAh = 8.8Ah

Setting charging at 0.5*C
Ichg = 8.8Ah * 0.5 = 4.4A
R = 50Akꭥ / 4.4A = 11.36kΩ

Since this is 100W, 20V/5A,
a safe upper will be set for 8A

using 5mΩ sense resistor:
R = 20Akꭥ / 8A = 2.5kΩ

Sheet: /USB-C Power Delivery/Buck-Boost Charger/
File: Buck-Boost Charger.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 17/33

Switching Frequency = 400kHz
Vin = 25.6V -> 33.6V (29.6V nominal)
Vout = 5V
Iout = 5A
Ripple = 1% * 5V = 50 mV

* From Datasheet: Delta_I_L should be between 30% and 50%:
* If we do 40%, 40% * 5A = 2A
Lo = Vout / (Delta_I_L * F_sw) * (1 - Vout / Vin_nom) = 5 / (2 * 400k) * (1 - 5 / 29.6) = 5.19uH -> 5.6uH

* "The inductor's saturation current rating must be higher than the maximum peak current."
Deltaa_I_L = Vout / (L_o * F_sw) * (1 - Vout / Vin_MAX) = 5 / (5.6u * 400k) * (1 - 5 / 33.6) = 1.89A
* Verify with the peak current
I_LO(PK) = I_o + Delta_I_L / 2 = 5 + 1.89 / 2 = 5.945
*** The inductor selected must have a saturation current rating above 6A

A 20% Saftey margin: 5.95 * 1.2 = 7.14A
The switching frequency is programmed by a single resistor from the RT pin to AGND.
R_RT[kΩ] = ((10^6 / F_sw{kHz}) - 59) / 41 = ((10^6 / 400) - 59) / 41 = 59.54kΩ = 59k

(pin 6)
* From datasheet:
* Vrefl = 1V
* I_IMON_OFFSET = 25uA
* gm_IMON = 2uA/mV
R_IMON = V_refl / ((R_CS * G_mIMON * I_CC) + I_IMON_OFFSET) = 1 / ((8m * 2u * 5) + 25u) = 9.53kΩ

F_C = F_sw / 10 = 400k / 10 = 40k
C_IMON = 1 / (2 * pi * R_IMON * F_C) = 1 / (2 * pi * 9.54k * 40k) = 417pF = 470pF

Current Sense Resistor:
* From Datasheet V_CS-TH = 60mV
Rs = V_CS-TH / (1.2 * I_LO(PK)) = 60m / (1.2 * 5.95) = 8.4mΩ -> 8mΩ is a standard resistor

* Ripple Cout
Cout >= Delta_I_L / (8 * F_sw * Delta_Vout) = 1.9 / (8 * 400k * 50m) = 11.87uF = 11.9uF
* Transient Cout, assuming an acceptable voltage overshoot of 3% (150 mV)
Cout >= (L_o * (Delta_I_Load)^2) / ((Vout + Delta_V_overshoot)^2 - (Vout)^2) = (5.6u * 5^2) / ((5 + 0.15)^2 - 5^2) = 92uF = 100uF

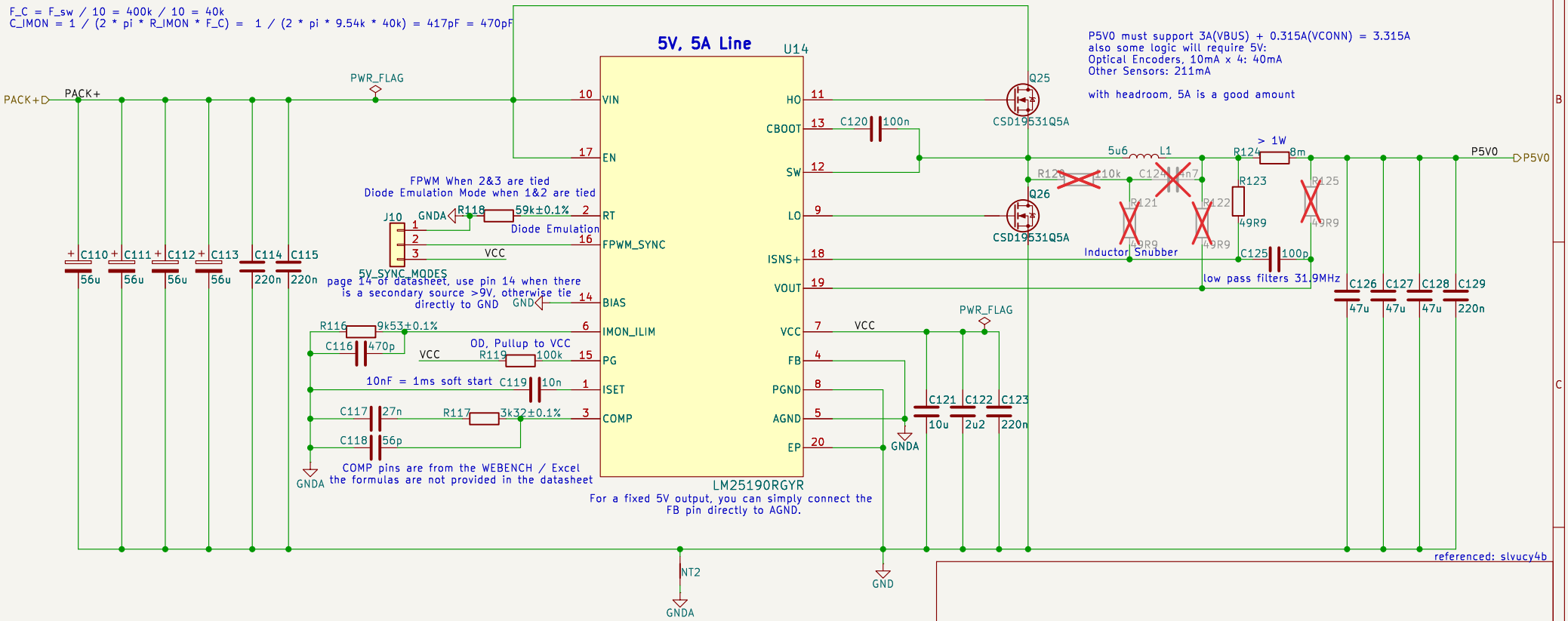
I_CO(RMS) = Delta_I_L / 12^(1/2) = 0.55A
*** The Capacitor Bank must be rated for > 0.55A for its ripple current

Cin:
Dmax = Vout / Vin(MIN) = 5 / 25.6 = 195m
I_CIN(RMS) = SQRT(D * ((1 - D) * (I_out)^2 + (Delta_I_L)^2 / 12) = SQRT(195m * ((1 - 195m) * 5^2 + 1.9^2 / 12) = 1.99 A(RMS)
Delta_V_Supply = 1% * 25.6V = 0.256 = 0.25V
C_in >= (I_out * D * (1 - D)) / (F_sw * Delta_V_Supply) = (5 * 195m * (1 - 195m)) / (400k * 0.25) = 7.85uF

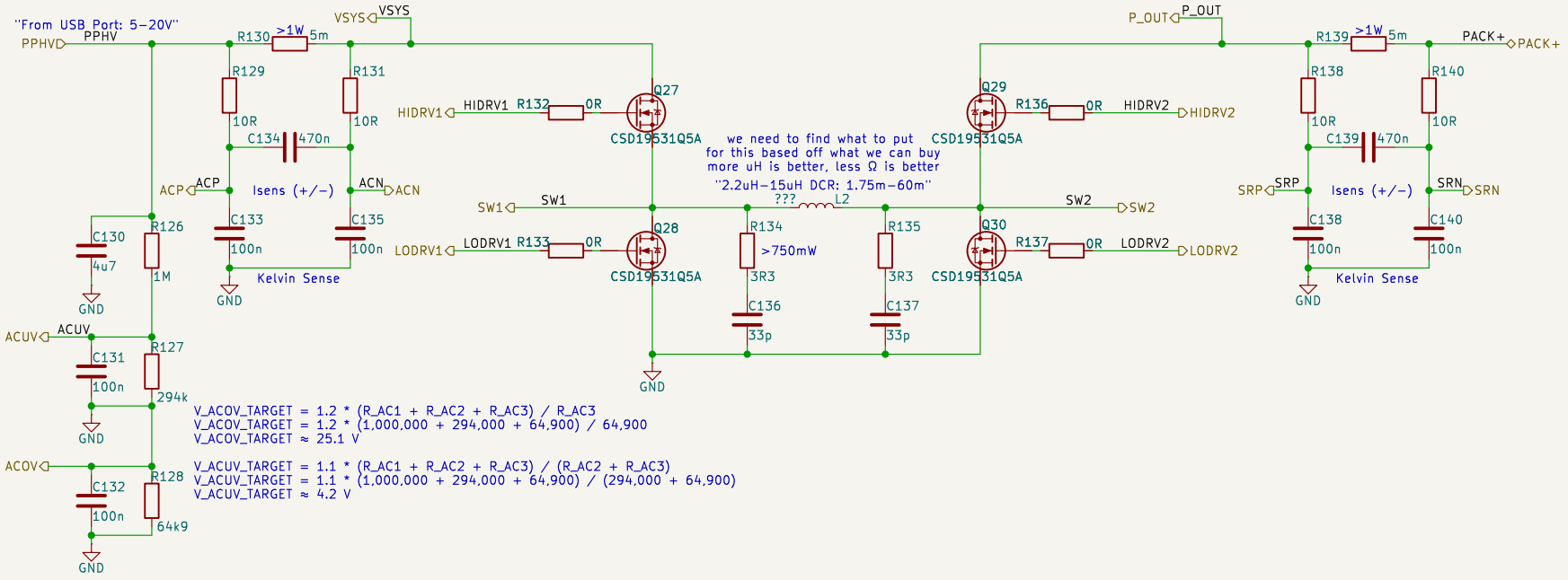
Soft Start:
* From datasheet: I_ISET (internal) is 10uA
* Delta_V = 1V (voltage ramp from 0 to 1)
I = Delta_Q / Delta_T , C = Delta_Q / Delta_V , Delta_T = (C * Delta_V) / I
T_SSCC = (C_ISET * Delta_V) / I_ISET = (10n * 1) / 10u = 1ms

P5V0 must support 3A(VBUS) + 0.315A(VCONN) = 3.315A
also some logic will require 5V:
Optical Encoders, 10mA x 4: 40mA
Other Sensors: 211mA

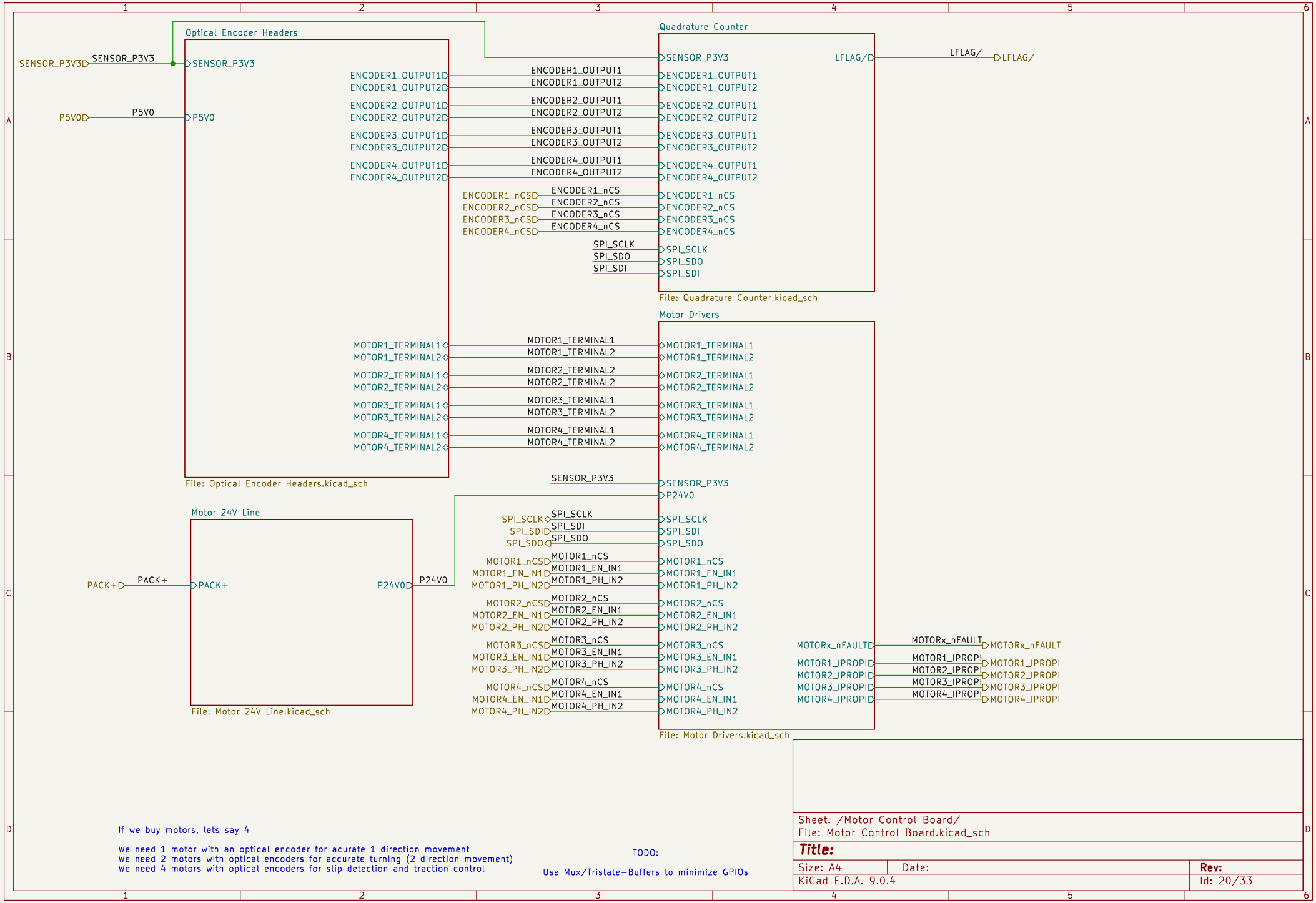
with headroom, 5A is a good amount



Sheet: /USB-C Power Delivery/USB-C 5V Buck/		
File: USB-C 5V Buck.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 18/33



Sheet: /USB-C Power Delivery/Charger FET Array/		
File: Charger FET Array.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 19/33


Optical Encoder Headers
SENSOR_P3V3
P5V0
ENCODER1_OUTPUT1
ENCODER1_OUTPUT2
ENCODER2_OUTPUT1
ENCODER2_OUTPUT2
ENCODER3_OUTPUT1
ENCODER3_OUTPUT2
ENCODER4_OUTPUT1
ENCODER4_OUTPUT2
ENCODER1_nCS
ENCODER2_nCS
ENCODER3_nCS
ENCODER4_nCS
SPI_SCLK
SPI_SDO
SPI_SDI
MOTOR1_TERMINAL1
MOTOR1_TERMINAL2
MOTOR2_TERMINAL1
MOTOR2_TERMINAL2
MOTOR3_TERMINAL1
MOTOR3_TERMINAL2
MOTOR4_TERMINAL1
MOTOR4_TERMINAL2
File: Optical Encoder Headers.kicad_sch
Quadrature Counter
LFLAG/
File: Quadrature Counter.kicad_sch
Motor Drivers
P24V0
MOTOR1_nCS
MOTOR1_EN_IN1
MOTOR1_PH_IN2
MOTOR2_nCS
MOTOR2_EN_IN1
MOTOR2_PH_IN2
MOTOR3_nCS
MOTOR3_EN_IN1
MOTOR3_PH_IN2
MOTOR4_nCS
MOTOR4_EN_IN1
MOTOR4_PH_IN2
MOTORx_nFAULT
MOTOR1_IPROPI
MOTOR2_IPROPI
MOTOR3_IPROPI
MOTOR4_IPROPI
File: Motor Drivers.kicad_sch
Motor 24V Line
PACK+
File: Motor 24V Line.kicad_sch
If we buy motors, lets say 4
We need 1 motor with an optical encoder for acurate 1 direction movement
We need 2 motors with optical encoders for accurate turning (2 direction movement)
We need 4 motors with optical encoders for slip detection and traction control
TODO:
Use Mux/Tristate-Buffers to minimize GPIOs
Sheet: /Motor Control Board/
File: Motor Control Board.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 20/33

Switching Frequency = 400kHz
Vin = 25.6V -> 33.6V (29.6V nominal)
Vout = 24V
Iout = 20A
Ripple = 1% * 24V = 240 mV

* From Datasheet: Delta_I_L should be between 30% and 50%:
* If we do 30%, 30% * 20A = 6A
Lo = Vout / (Delta_I_L * F_sw) * (1 - Vout / Vin_nom) = 24 / (6 * 400k) * (1 - 24 / 29.6) = 1.9uH -> 2.2uH

* "The inductor's saturation current rating must be higher than the maximum peak current."
Delta_I_L = Vout / (L_o * F_sw) * (1 - Vout / Vin_MAX) = 24 / (2.2u * 400k) * (1 - 24 / 33.6) = 7.8A
* Verify with the peak current
I_LO(PK) = I_o + Delta_I_L / 2 = 20 + 7.8 / 2 = 23.9A
*** The inductor selected must have a saturation current rating above 24A

A 20% Saftey margin: 23.9 * 1.2 = 28.7A

The switching frequency is programmed by a single resistor from the RT pin to AGND.
R_RT[kΩ] = ((10^6 / F_sw[kHz]) - 59) / 41 = ((10^6 / 400) - 59) / 41 = 59.54kΩ -> 59kΩ

(pin 6)
* From datasheet:
* Vrefl = 1V
* I_IMON_OFFSET = 25uA
* gm_IMON = 2uA/mV
R_IMON = V_refl / ((R_CS * G_mIMON * I_CC) + I_IMON_OFFSET) = 1 / ((2m * 2u * 20) + 25u) = 9.26kΩ -> 9.31kΩ (standard E96)

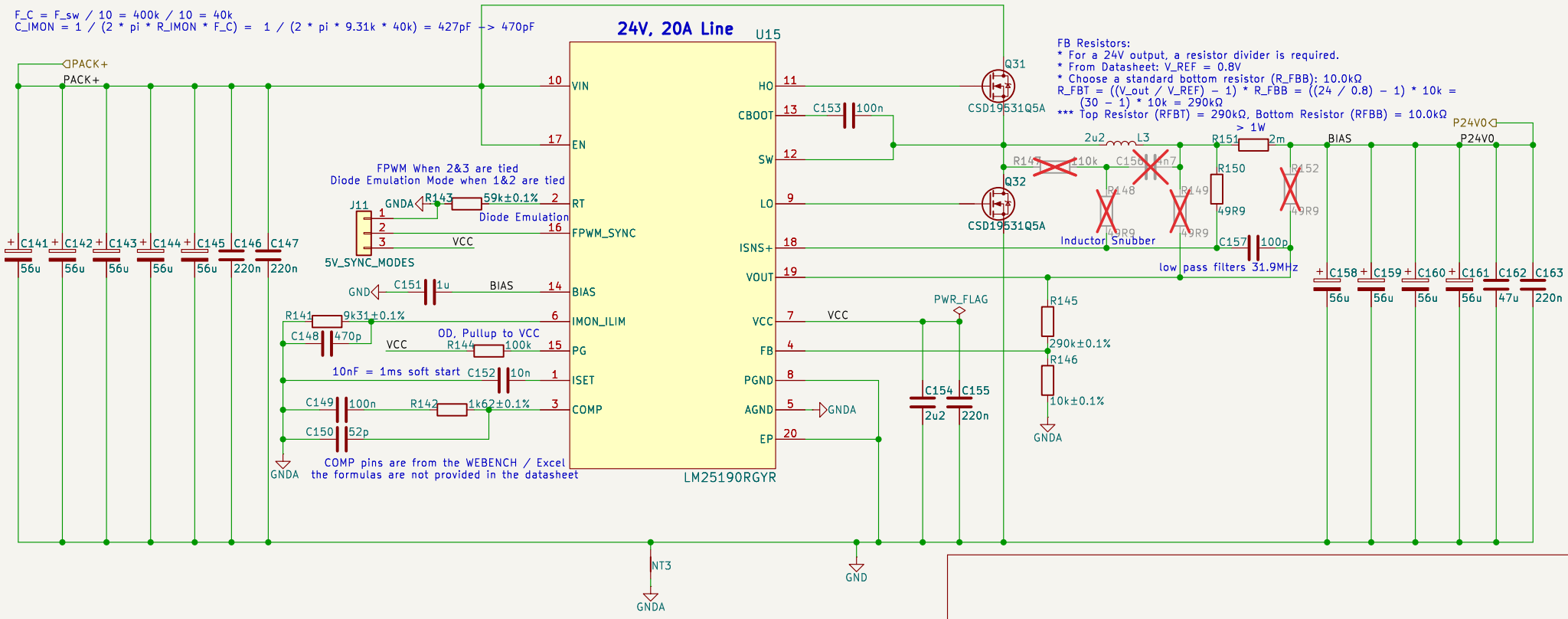
F_C = F_sw / 10 = 400k / 10 = 40k
C_IMON = 1 / (2 * pi * R_IMON * F_C) = 1 / (2 * pi * 9.31k * 40k) = 427pF -> 470pF

Current Sense Resistor:
* From Datasheet V_CS-TH = 60mV
Rs = V_CS-TH / I_PEAKCL = 60m / 28.7A = 2.09mΩ -> 2.0mΩ is a standard resistor

* Cout
Transient response will dominate capacitor selection for a high-current design.
A practical choice for a 20A output would be at least 220uF effective capacitance. -> 220uF
* I_CO(RMS) = Delta_I_L / 12^(1/2) = 7.8 / 3.46 = 2.25A(RMS)
*** The Capacitor Bank must be rated for > 2.25A for its ripple current

Cin:
Dmax = Vout / Vin(MIN) = 24 / 25.6 = 0.938
Delta_I_L(@Vin_MIN) = 24 / (2.2u * 400k) * (1 - 24 / 25.6) = 1.7A
I_CIN(RMS) = SQRT(D * ((1 - D) * I_out^2 + Delta_I_L^2 / 12)) = SQRT(0.938 * ((1 - 0.938) * 20^2 + 1.7^2 / 12)) = 4.88A(RMS)
Delta_V_Supply = 1% * 25.6V = 0.256V = 0.25V
C_in >= (I_out * D * (1 - D)) / (F_sw * Delta_V_Supply) = (20 * 0.938 * (1 - 0.938)) / (400k * 0.25) = 11.6uF -> 22uF effective

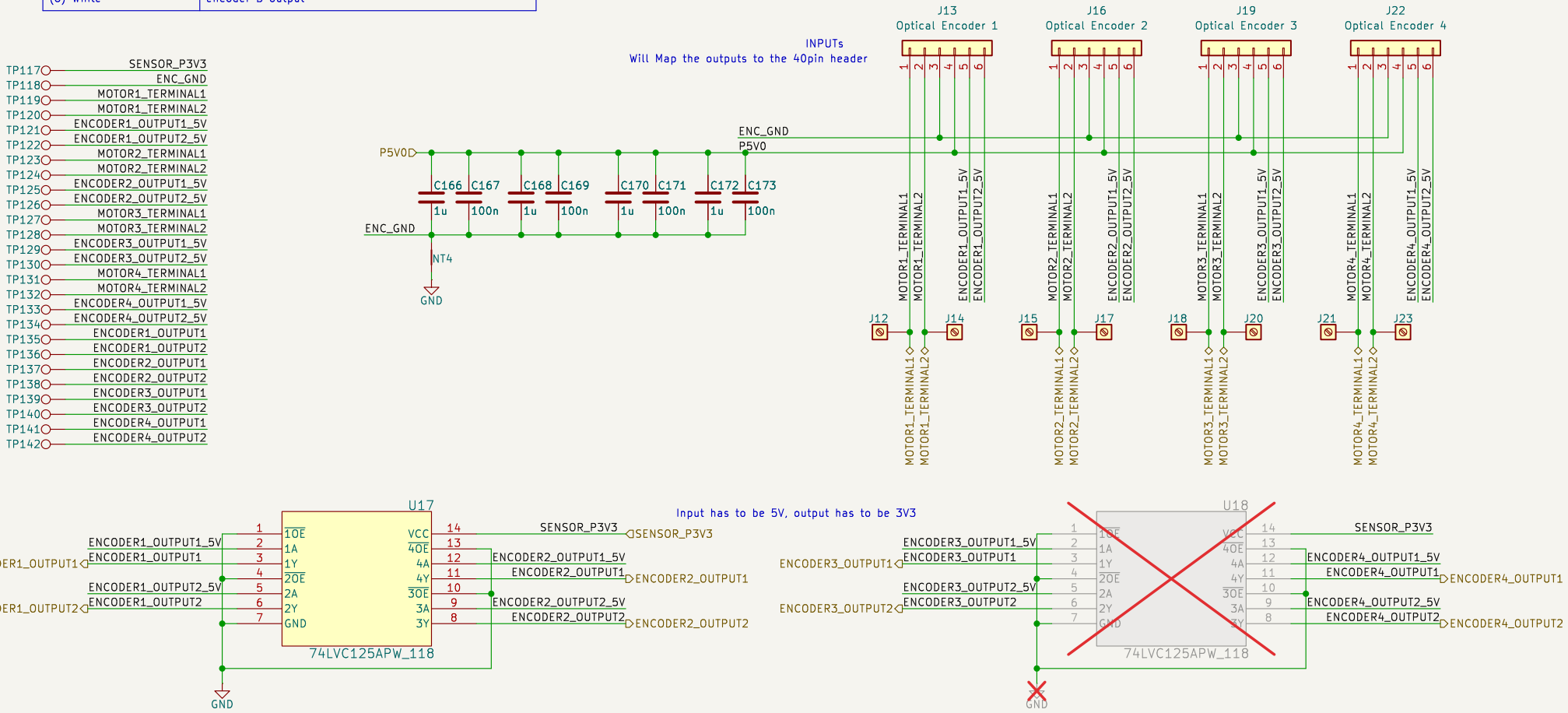
Soft Start:
* From datasheet: I_ISET (internal) is 10uA
* Delta_V = 1V (voltage ramp from 0 to 1)
For a slower start on this high-power line, let's select C_ISET = 22nF
T_SSCC = (C_ISET * Delta_V) / I_ISET = (22n * 1) / 10u = 2.2ms



using global GNDA, is this what we should be doing?

Sheet: /Motor Control Board/Motor 24V Line/		
File: Motor 24V Line.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 22/33

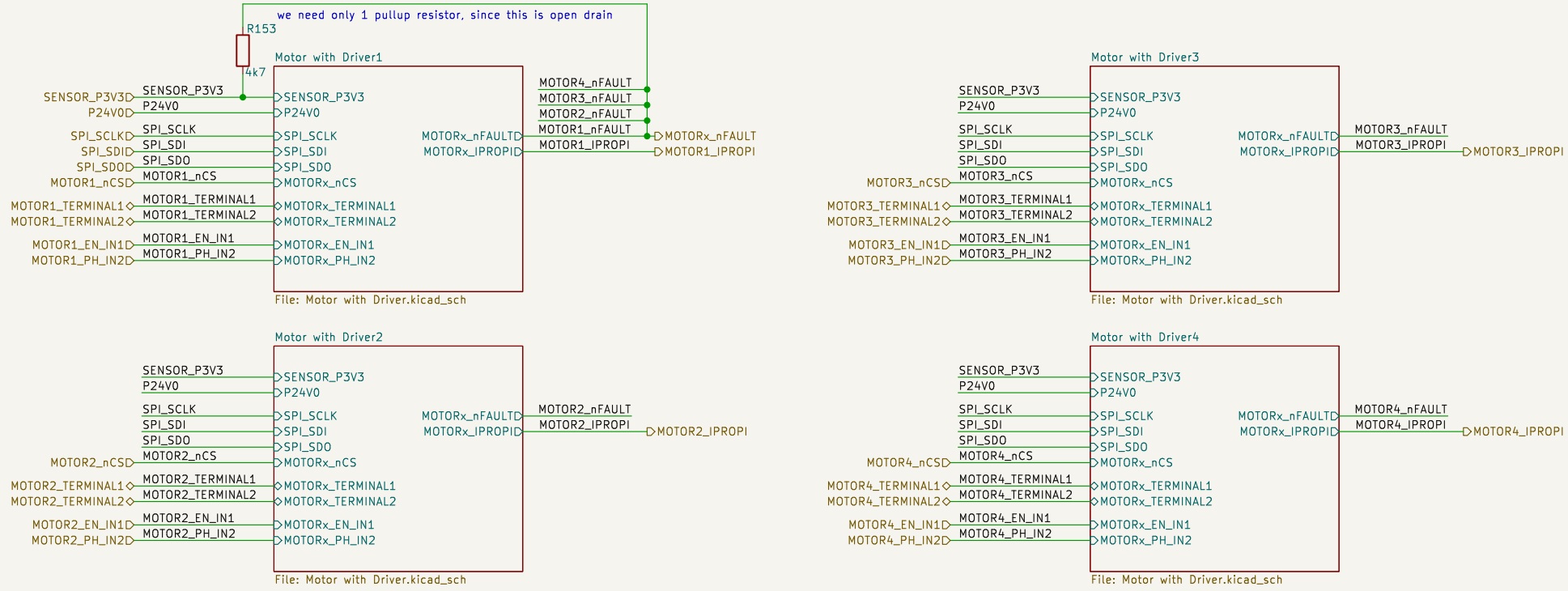
(1) Red	motor power (connects to one motor terminal)
(2) Black	motor power (connects to the other motor terminal)
(3) Green	encoder GND
(4) Blue	encoder Vcc (3.5 – 20 V)
(5) Yellow	encoder A output
(6) White	encoder B output



Sheet: /Motor Control Board/Optical Encoder Headers/
File: Optical Encoder Headers.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 24/33



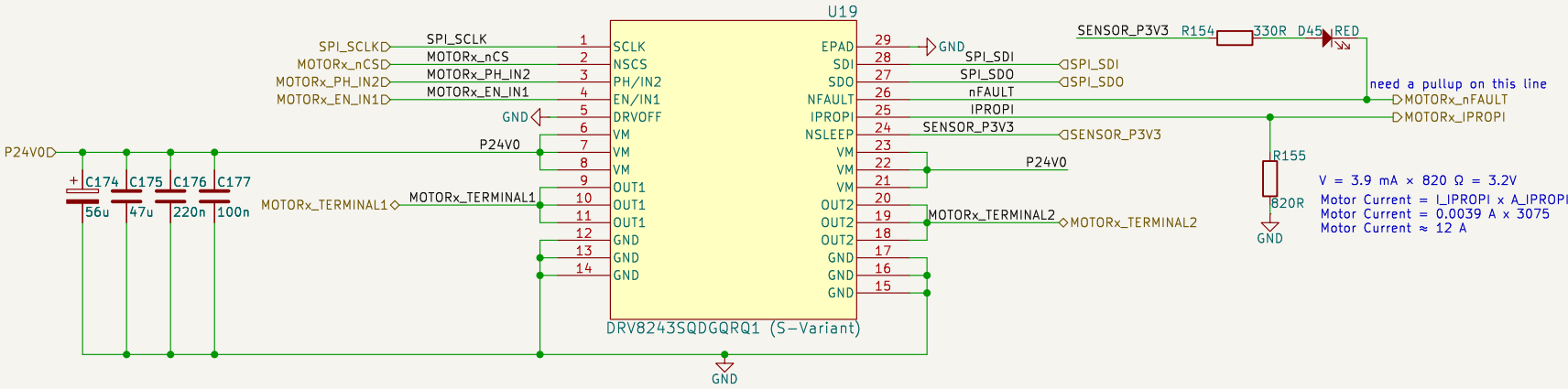
Sheet: /Motor Control Board/Motor Drivers/
File: Motor Drivers.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 25/33

For the SPI (P) variant, you can set the SCLK frequency up to 10 MHz.

For the SPI (S) variant, the maximum is 10 MHz, but you should reduce it to 8 MHz if you have a 20 pF load on the SDO line.



daisy chain these SCLK
put a termination 50Ω resistor and 100pF cap in series to ground

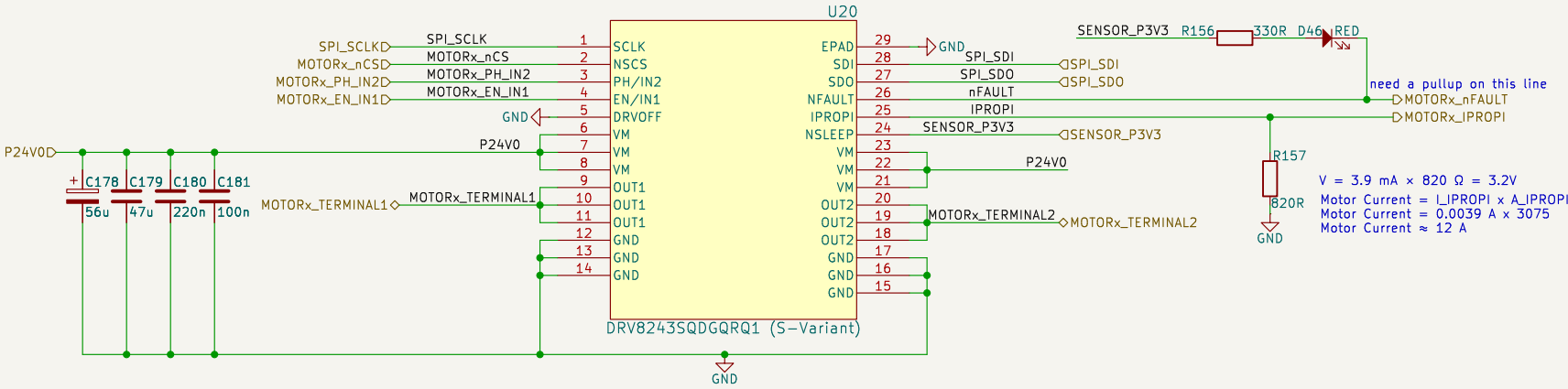
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver1/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 26/33

For the SPI (P) variant, you can set the SCLK frequency up to 10 MHz.

For the SPI (S) variant, the maximum is 10 MHz, but you should reduce it to 8 MHz if you have a 20 pF load on the SDO line.



daisy chain these SCLK
put a termination 50Ω resistor and 100pF cap in series to ground

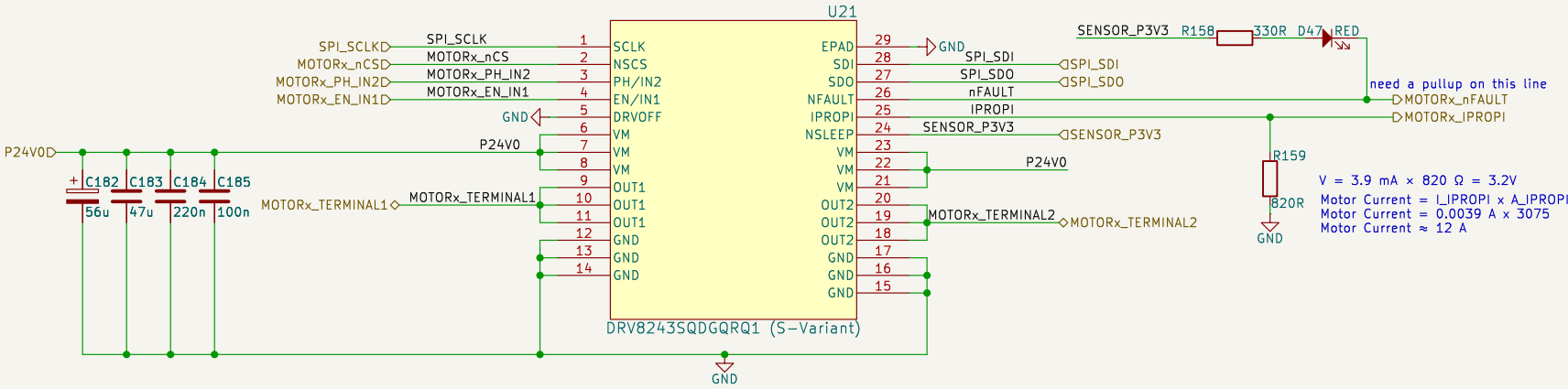
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver2/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 27/33

For the SPI (P) variant, you can set the SCLK frequency up to 10 MHz.

For the SPI (S) variant, the maximum is 10 MHz, but you should reduce it to 8 MHz if you have a 20 pF load on the SDO line.



daisy chain these SCLK
put a termination 50Ω resistor and 100pF cap in series to ground

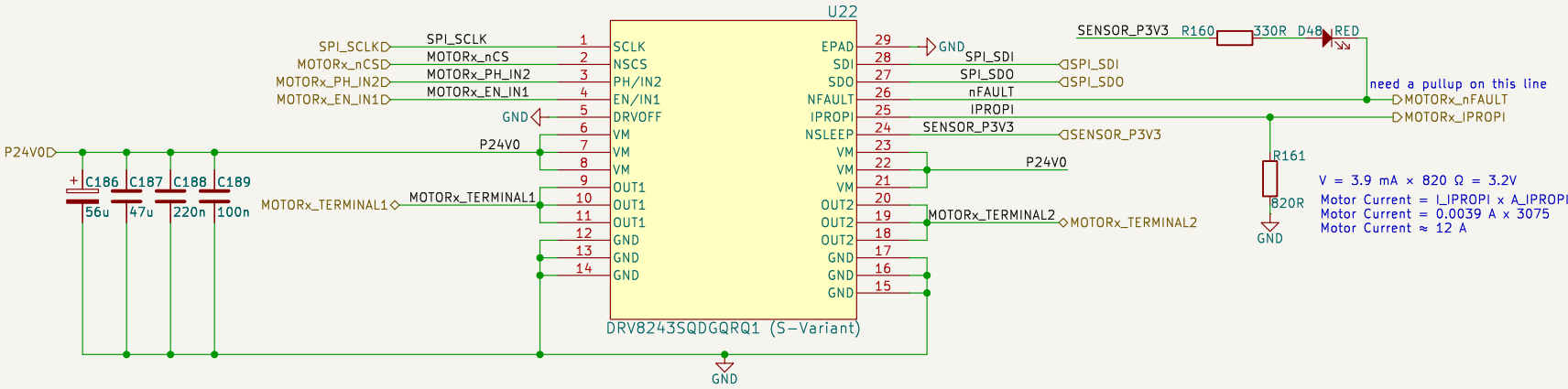
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver3/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 28/33

For the SPI (P) variant, you can set the SCLK frequency up to 10 MHz.

For the SPI (S) variant, the maximum is 10 MHz, but you should reduce it to 8 MHz if you have a 20 pF load on the SDO line.

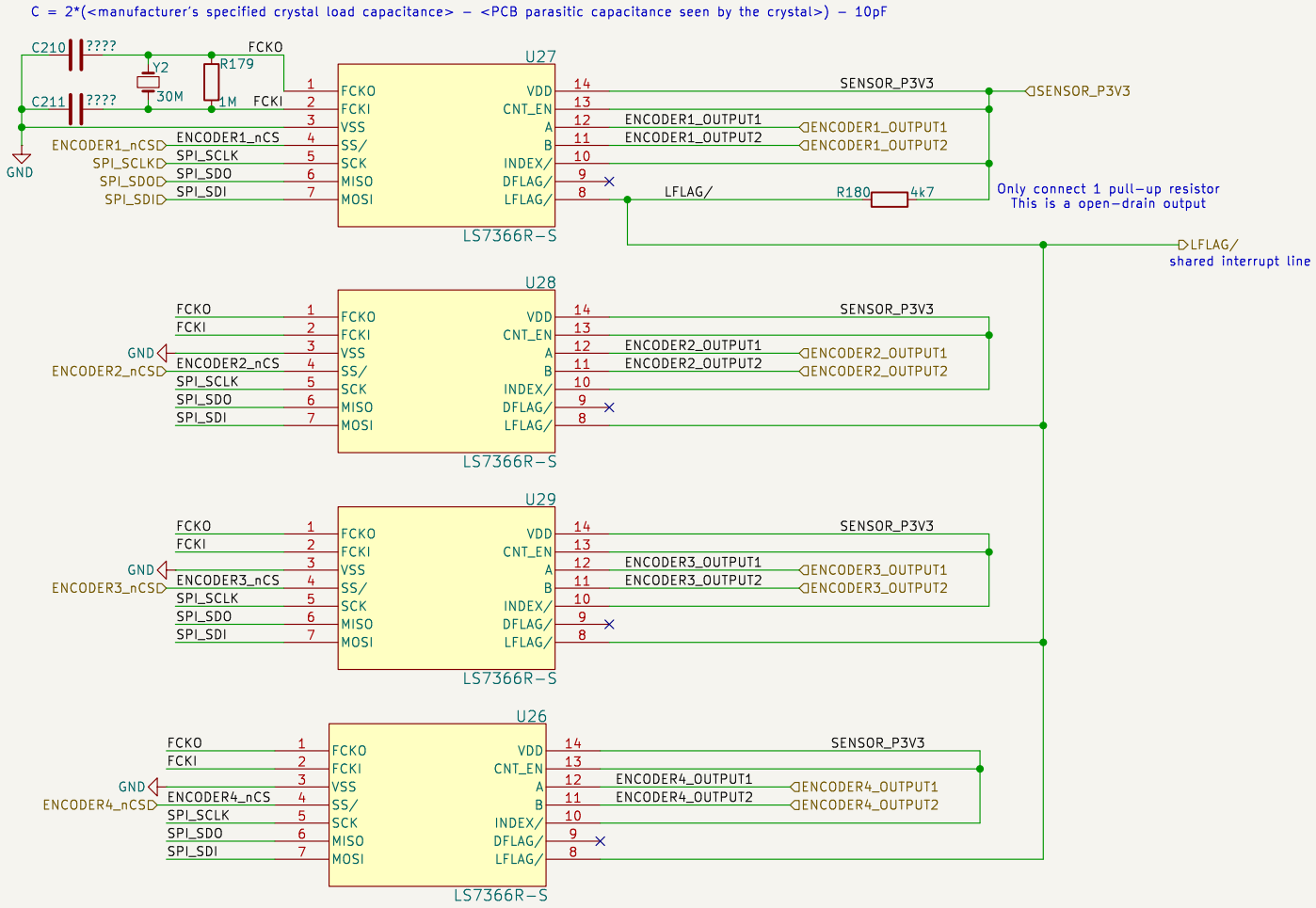


daisy chain these SCLK
put a termination 50Ω resistor and 100pF cap in series to ground

Sheet: /Motor Control Board/Motor Drivers/Motor with Driver4/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 29/33



we might need to have 4 crystals

put an oscillator instead:

. Instead of a crystal the

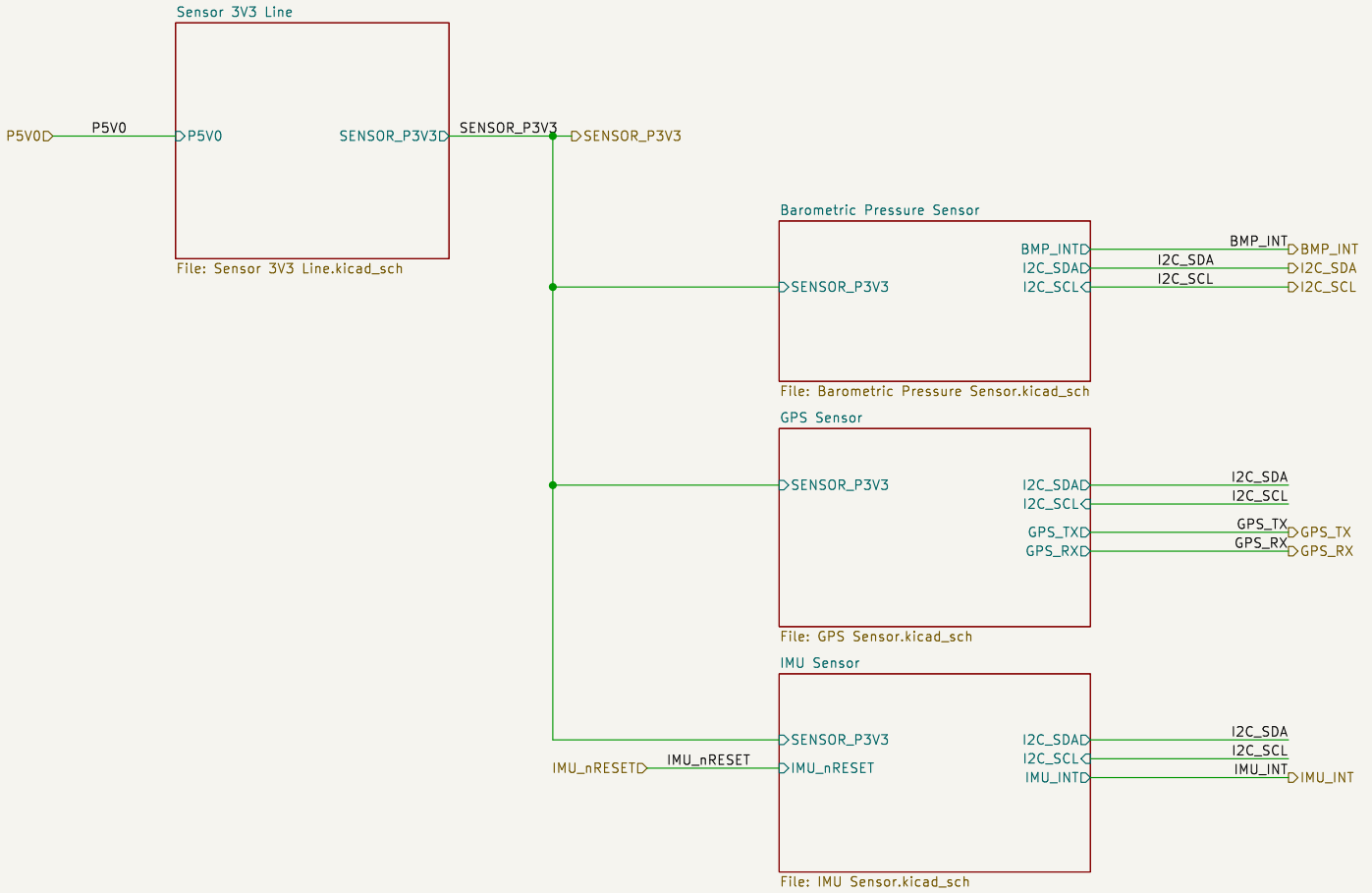
fCKi input may also be driven by an external clock.

put an end or source termination on the external clock or both (put with DNP)

Sheet: /Motor Control Board/Quadrature Counter/
File: Quadrature Counter.kicad_sch

Title:

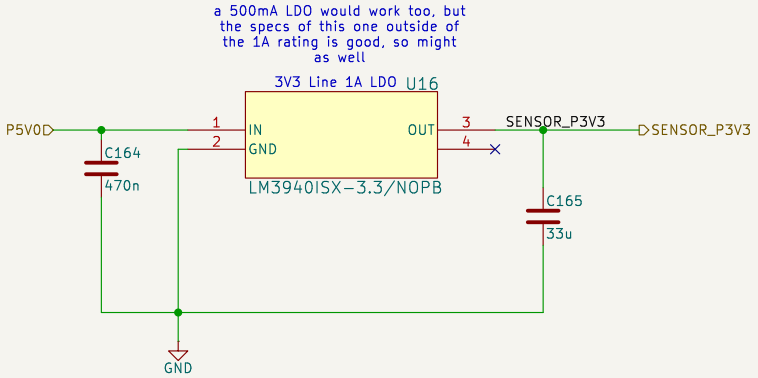
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 33/33



Sheet: /Sensor Board/
File: Sensor Board.kicad_sch

Title:

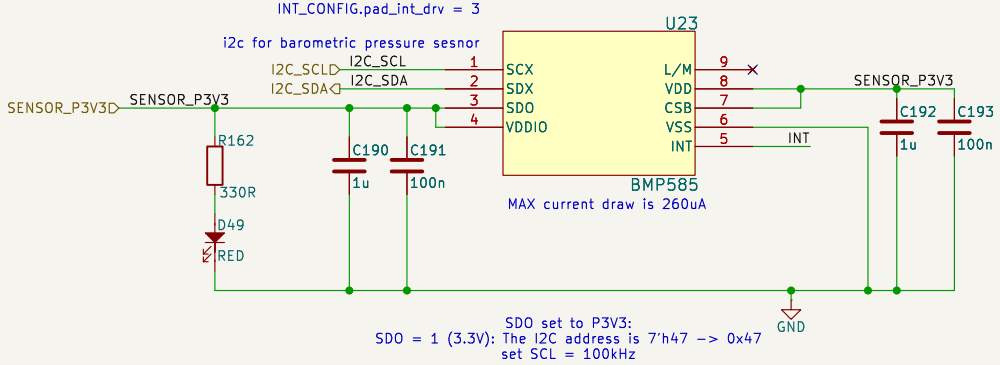
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 21/33

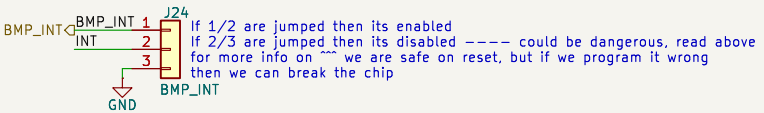


Sheet: /Sensor Board/Sensor 3V3 Line/		
File: Sensor 3V3 Line.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 23/33

Per the datasheet, the unused INT pin is tied to GND to prevent a floating input.
CRITICAL: The interrupt pin must be disabled in software to prevent a short circuit.
Ensure the 'int_en' bit in the INT_CONFIG register (0x14) remains disabled (set to 0).

Datasheet sets the IRQ to 0 on start up, so never turn it on
PAGE49: 0x14 on reset is set to 0x35
PAGE49: <int_en 2bits> <int_od 2bits> <int_pol 2bits> <int_mode 2bits LSB>
PAGE51: 8.5, setup for 0x14, <0> <1> <0> <1>
this means: int_mode = latched, int_pol = active low, int_od = open_drain, int_en = disabled
INT_CONFIG.int_en = 0
INT_CONFIG.od = 1
INT_CONFIG.pol = 0
INT_CONFIG.mode = 1
INT_CONFIG.pad_int_drv = 3

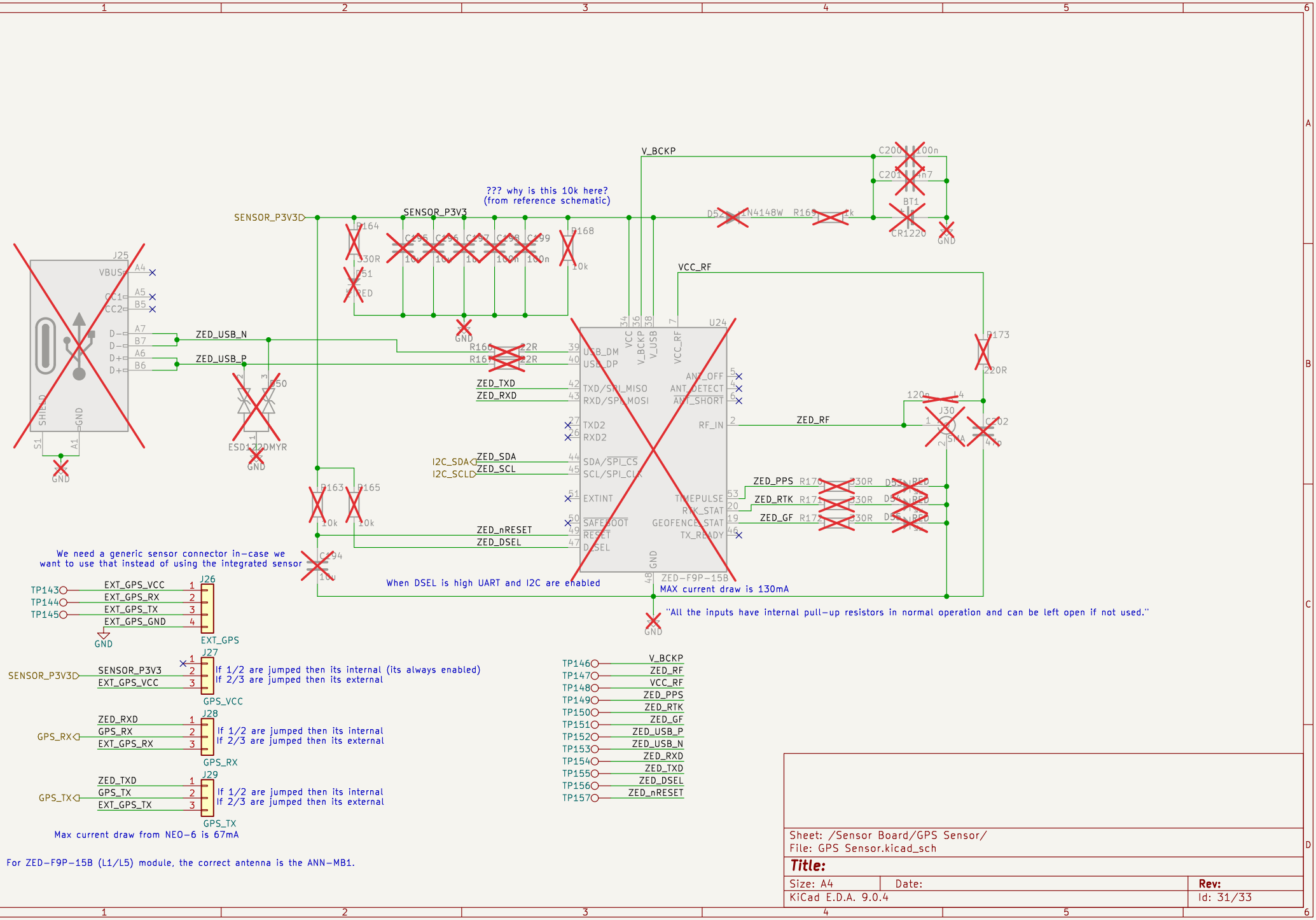


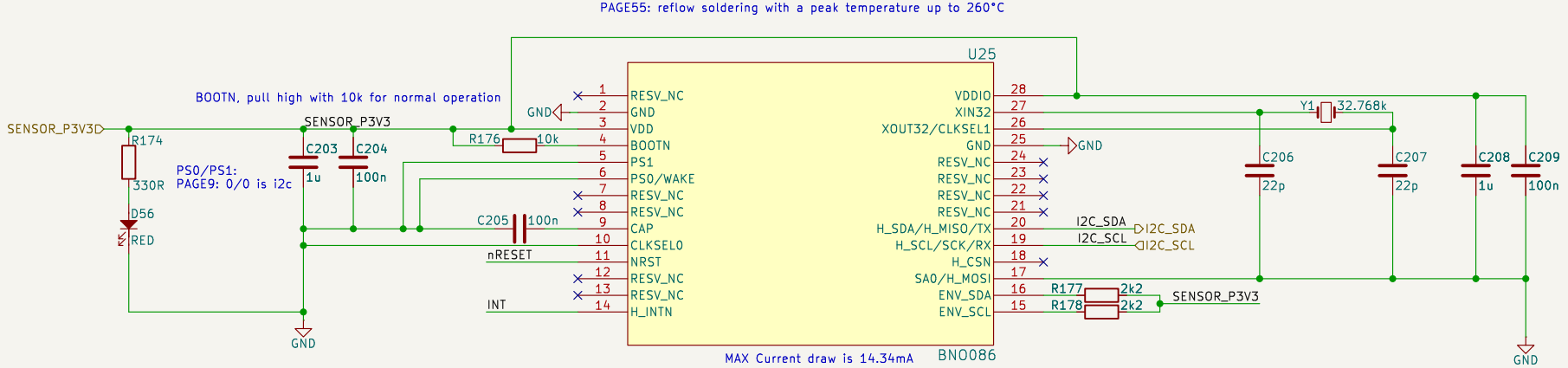


Sheet: /Sensor Board/Barometric Pressure Sensor/
File: Barometric Pressure Sensor.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 30/33


V_BCKP
??? why is this 10k here?
(from reference schematic)
SENSOR_P3V3
VCC_RF
ZED_USB_N
ZED_USB_P
ZED_TXD
ZED_RXD
ZED_SDA
ZED_SCL
ZED_nRESET
ZED_DSEL
ZED_RF
ZED_PPS
ZED_RTK
ZED_GF
I2C_SDA
I2C_SCL
ZED-F9P-15B
MAX current draw is 130mA
When DSEL is high UART and I2C are enabled
"All the inputs have internal pull-up resistors in normal operation and can be left open if not used."
We need a generic sensor connector in-case we want to use that instead of using the integrated sensor
EXT_GPS_VCC
EXT_GPS_RX
EXT_GPS_TX
EXT_GPS_GND
If 1/2 are jumped then its internal (its always enabled)
If 2/3 are jumped then its external
If 1/2 are jumped then its internal
If 2/3 are jumped then its external
If 1/2 are jumped then its internal
If 2/3 are jumped then its external
Max current draw from NEO-6 is 67mA
For ZED-F9P-15B (L1/L5) module, the correct antenna is the ANN-MB1.
Sheet: /Sensor Board/GPS Sensor/
File: GPS Sensor.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 31/33



Sheet: /Sensor Board/IMU Sensor/		
File: IMU Sensor.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 32/33