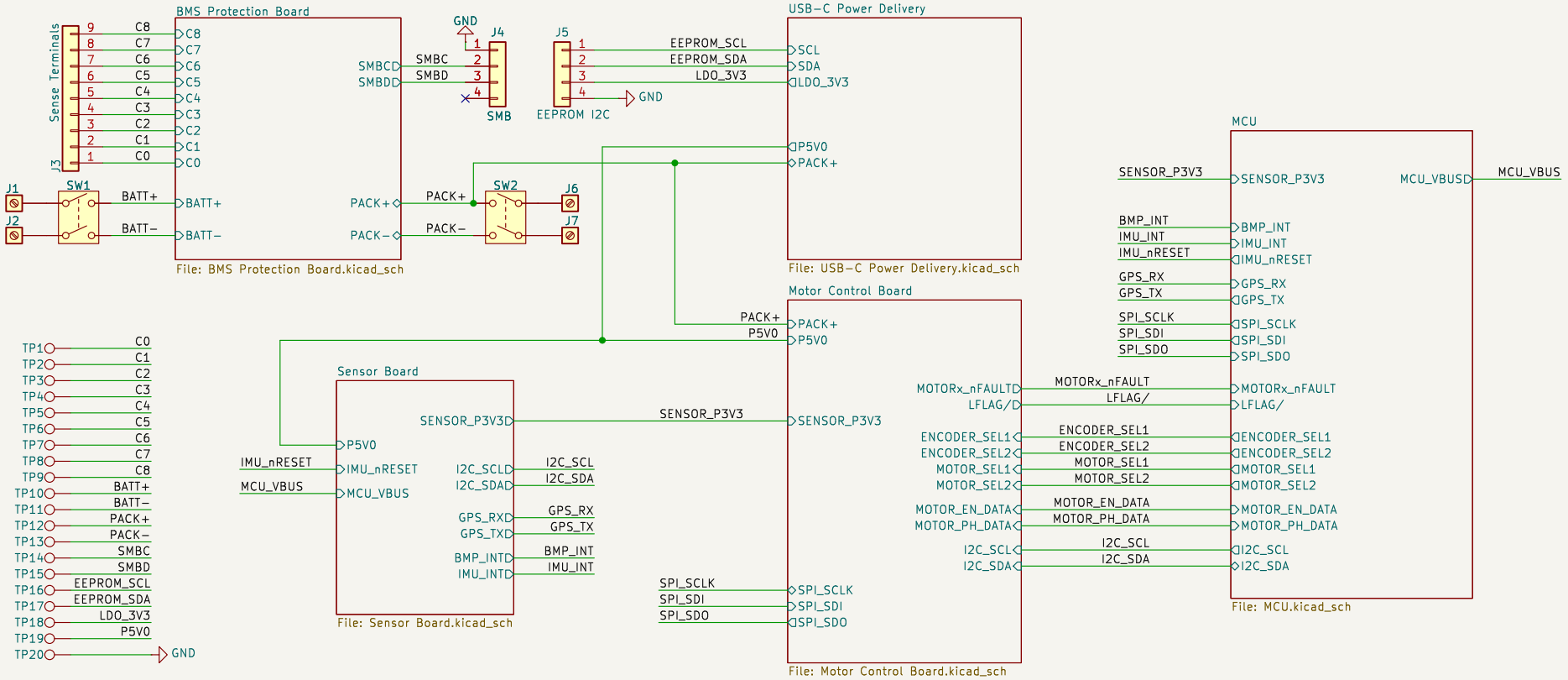
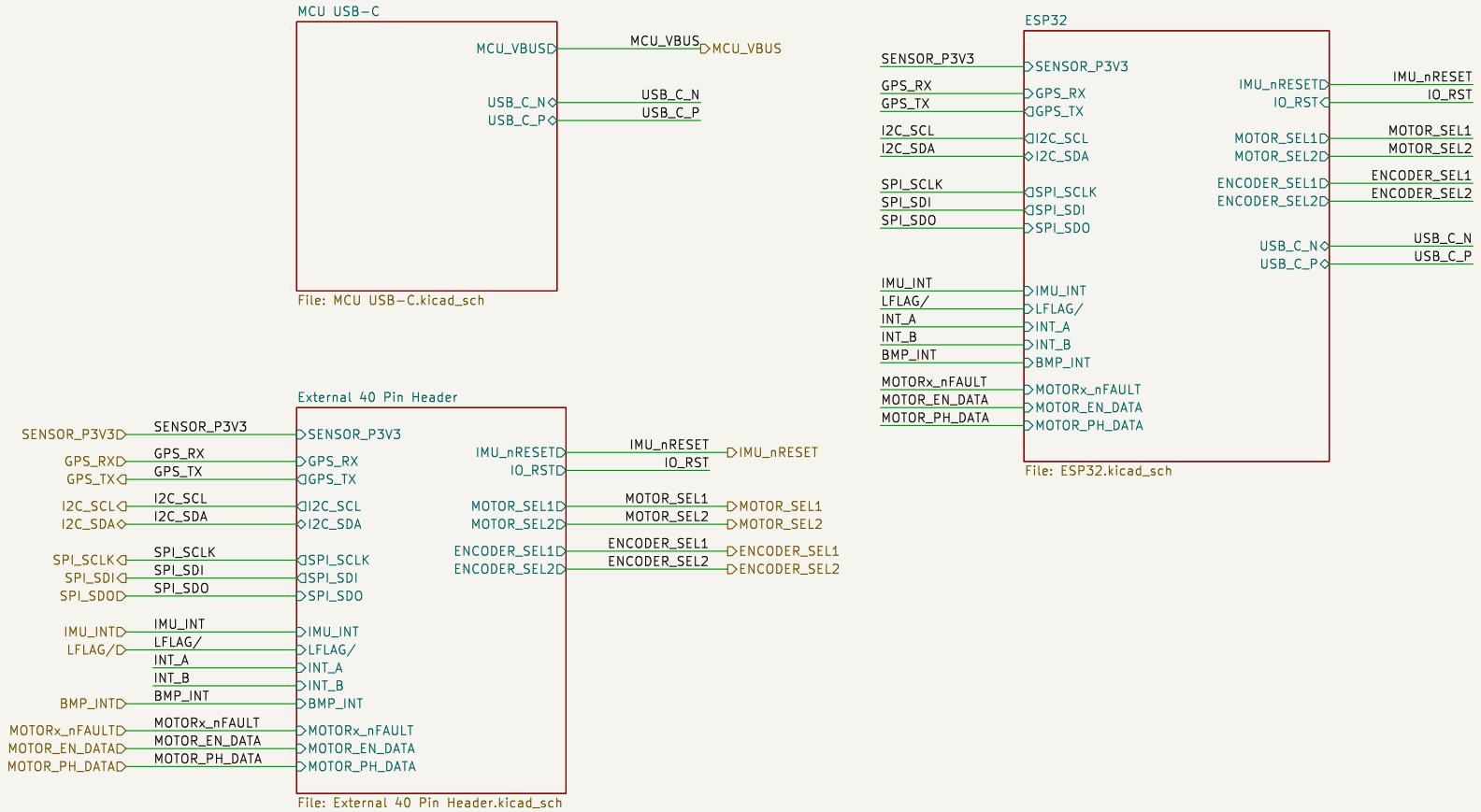


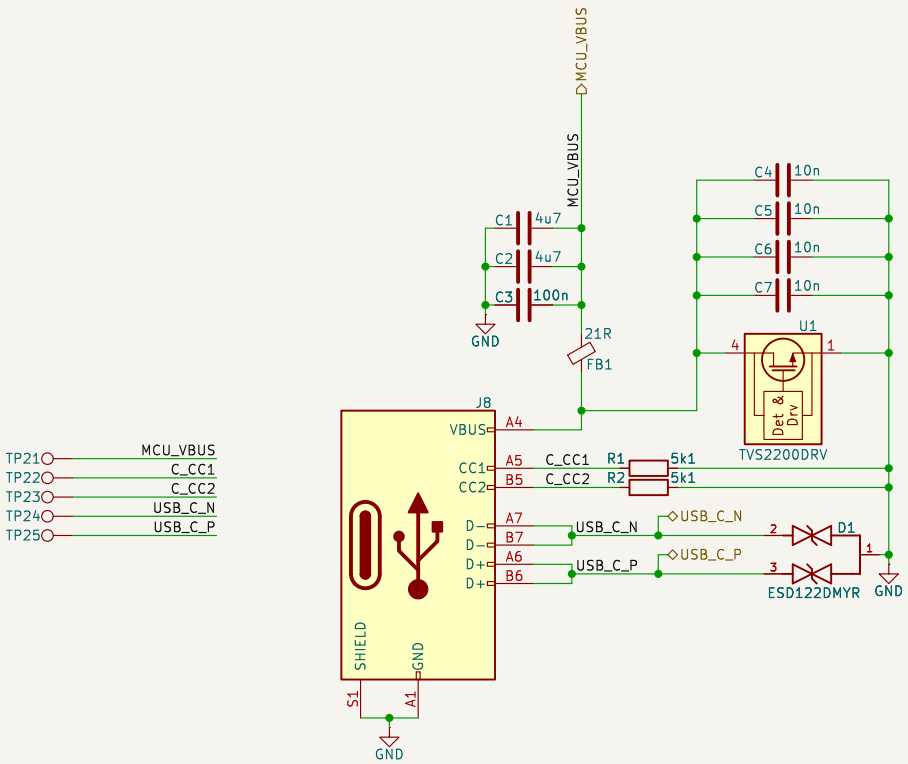
Sheet: /		
File: STAR.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 1/38



Sheet: /MCU/
File: MCU.kicad_sch

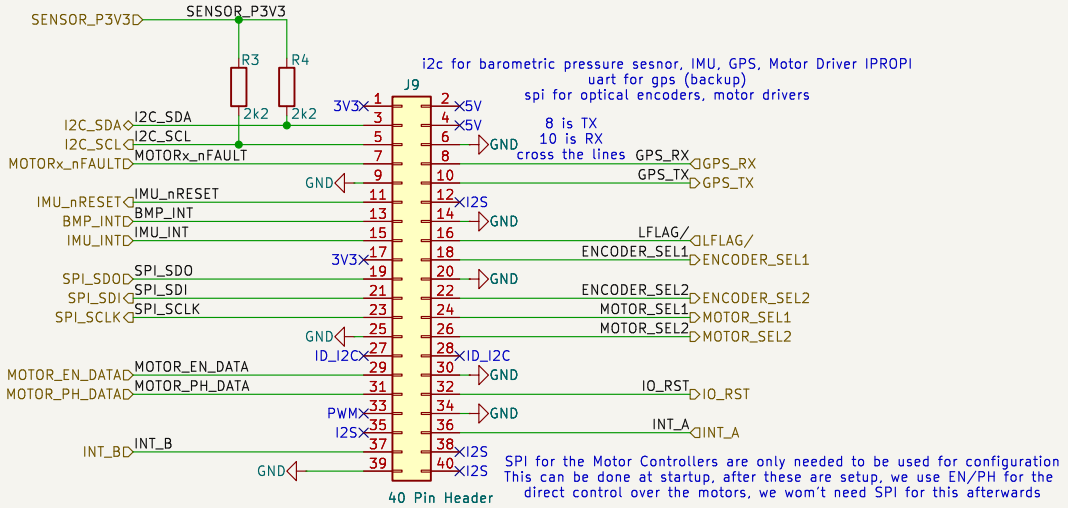
Title:

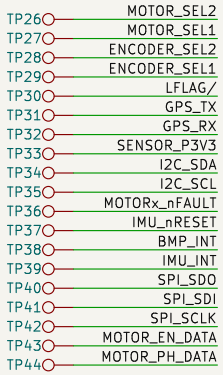
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 2/38



Sheet: /MCU/MCU USB−C/		
File: MCU USB−C.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 3/38

If we are using an external board (TDAA4VM J721EXSKG01EVM) 40 pin header

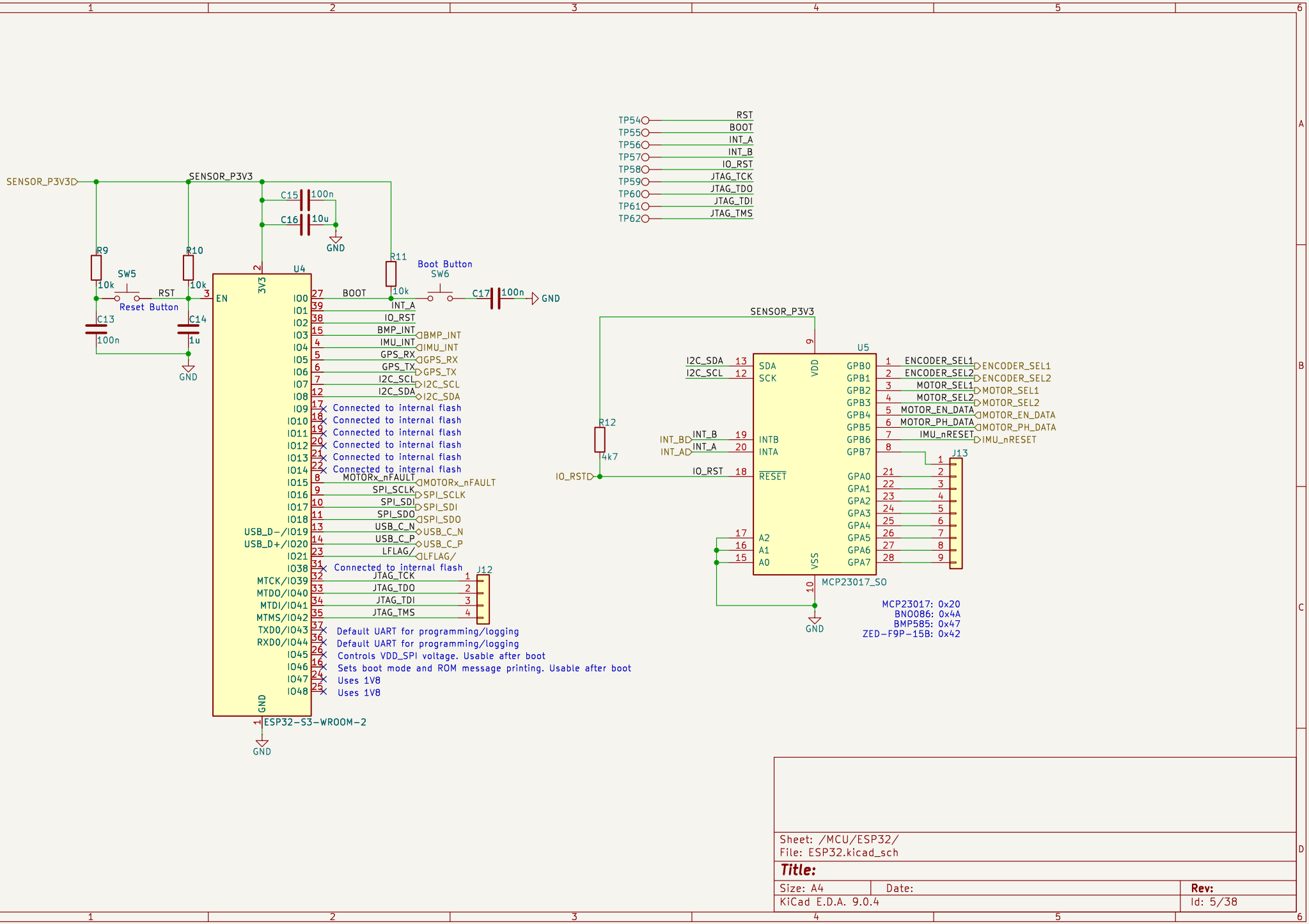


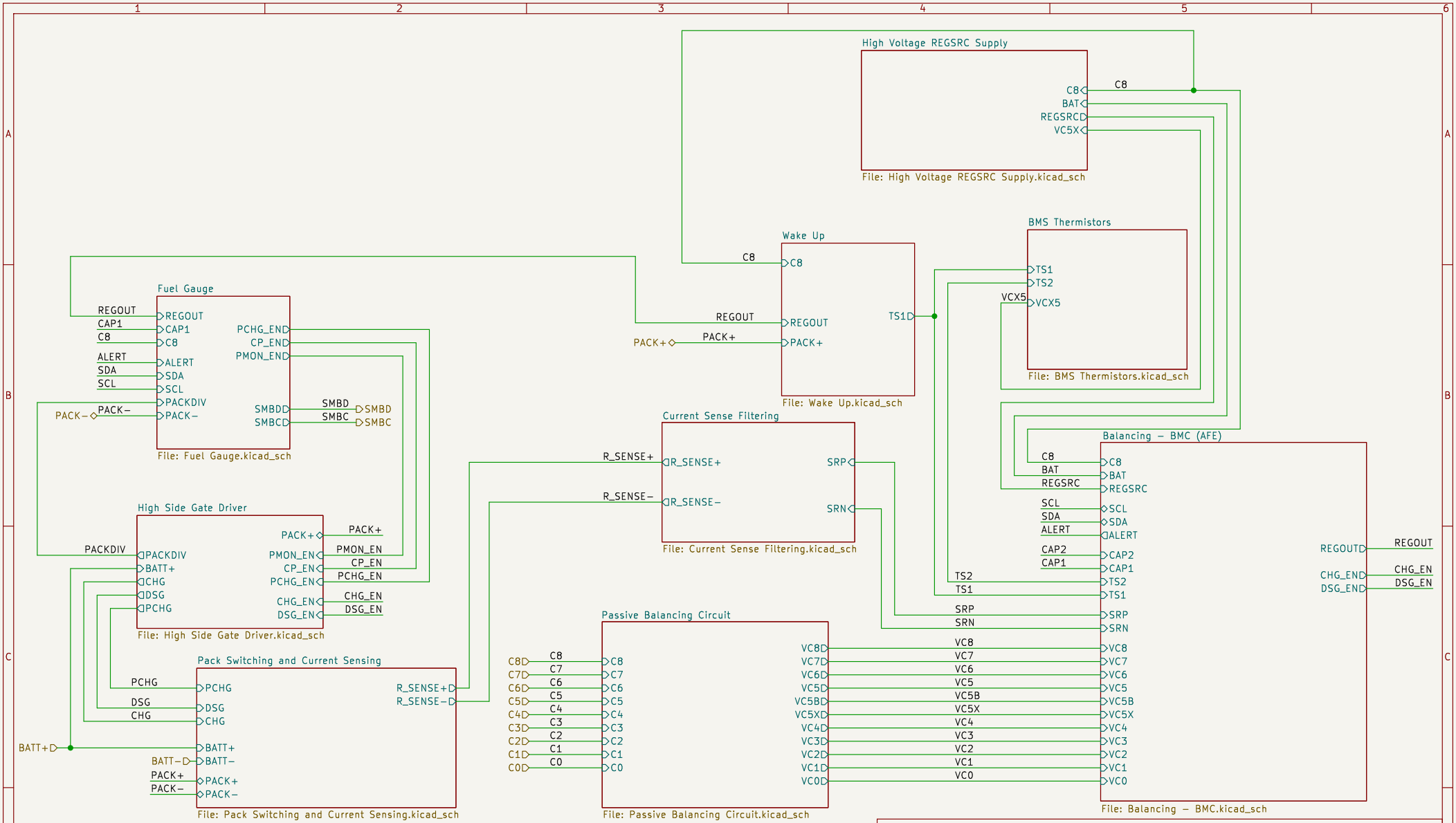


Sheet: /MCU/External 40 Pin Header/
File: External 40 Pin Header.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 4/38

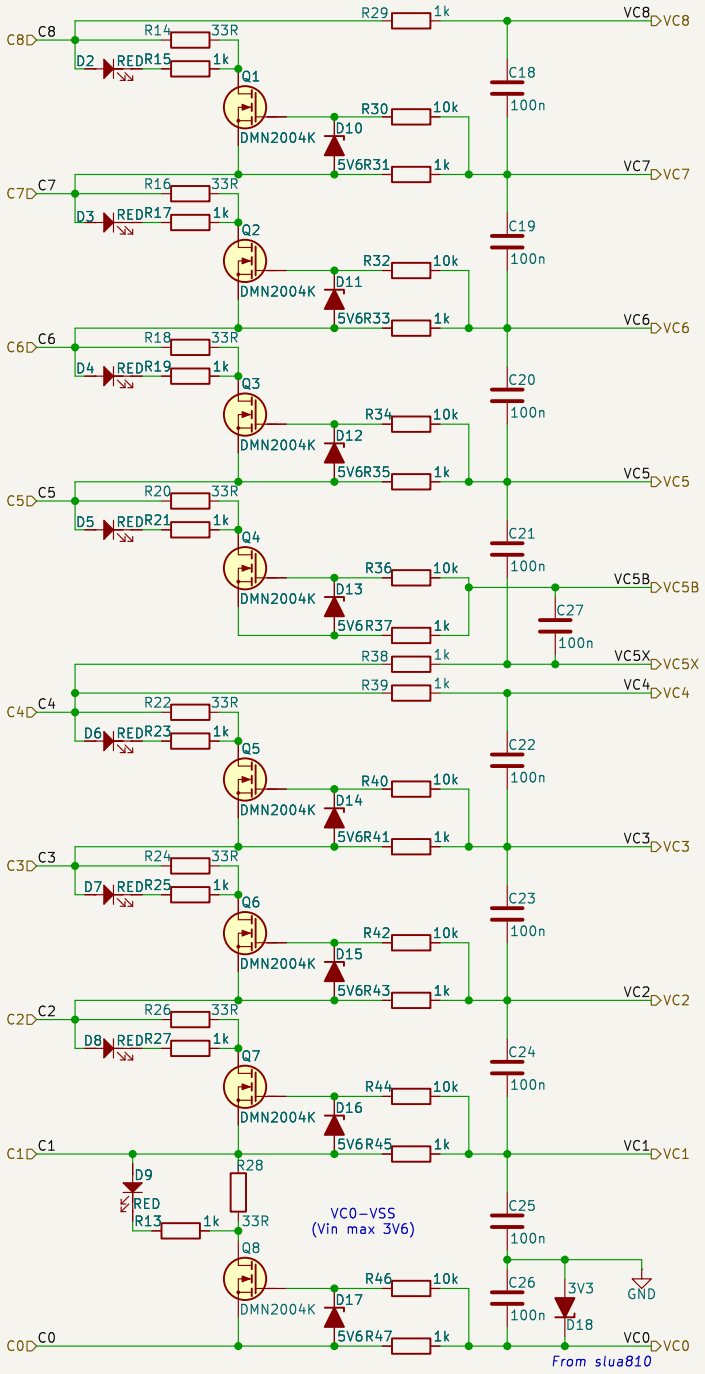

SENSOR_P3V3
C15 100n
C16 10u
GND
R9 10k
R10 10k
SW5
RST
Reset Button
C13 100n
C14 1u
R11 10k
Boot Button
SW6
C17 100n
U4
EN
3V3
IO0 BOOT
IO1 INT_A
IO2 IO_RST
IO3 BMP_INT
IO4 IMU_INT
IO5 GPS_RX
IO6 GPS_TX
IO7 I2C_SCL
IO8 I2C_SDA
IO9 Connected to internal flash
IO10 Connected to internal flash
IO11 Connected to internal flash
IO12 Connected to internal flash
IO13 Connected to internal flash
IO14 Connected to internal flash
IO15 MOTORx_nFAULT
IO16 SPI_SCLK
IO17 SPI_SDI
IO18 SPI_SDO
USB_D-/IO19 USB_C_N
USB_D+/IO20 USB_C_P
IO21 LFLAG/
IO38 Connected to internal flash
MTCK/IO39 JTAG_TCK
MTDO/IO40 JTAG_TDO
MTDI/IO41 JTAG_TDI
MTMS/IO42 JTAG_TMS
TXD0/IO43 Default UART for programming/logging
RXD0/IO44 Default UART for programming/logging
IO45 Controls VDD_SPI voltage. Usable after boot
IO46 Sets boot mode and ROM message printing. Usable after boot
IO47 Uses 1V8
IO48 Uses 1V8
GND
ESP32-S3-WROOM-2
J12
TP54 RST
TP55 BOOT
TP56 INT_A
TP57 INT_B
TP58 IO_RST
TP59 JTAG_TCK
TP60 JTAG_TDO
TP61 JTAG_TDI
TP62 JTAG_TMS
SENSOR_P3V3
R12 4k7
IO_RST
U5
I2C_SDA
I2C_SCL
SDA
SCK
VDD
INT_B
INT_A
INTB
INTA
RESET
A2
A1
A0
VSS
GPB0 ENCODER_SEL1
GPB1 ENCODER_SEL2
GPB2 MOTOR_SEL1
GPB3 MOTOR_SEL2
GPB4 MOTOR_EN_DATA
GPB5 MOTOR_PH_DATA
GPB6 IMU_nRESET
GPB7
GPA0
GPA1
GPA2
GPA3
GPA4
GPA5
GPA6
GPA7
MCP23017_SO
J13
MCP23017: 0x20
BNO086: 0x4A
BMP585: 0x47
ZED-F9P-15B: 0x42
Sheet: /MCU/ESP32/
File: ESP32.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 5/38

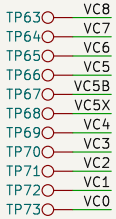


BMS Protection Board for 8S Battery Pack
In/Out: min voltage: 25.6V, nominal voltage: 29.6V, max voltage: 33.6V
Output: 50A continuous discharge current
Min Each Cell set to 3.2V, Max Each Cell set to 4.2V

Sheet: /BMS Protection Board/		
File: BMS Protection Board.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 7/38

Passive Balancing Circuit

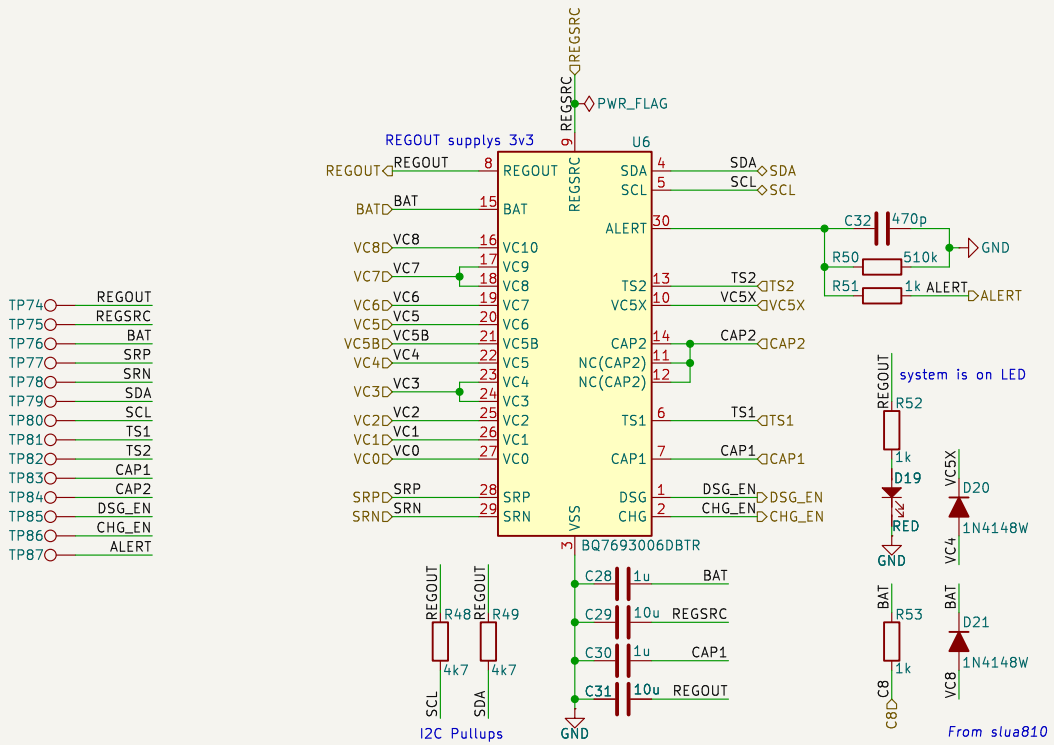




Sheet: /BMS Protection Board/Passive Balancing Circuit/
File: Passive Balancing Circuit.kicad_sch

Title:

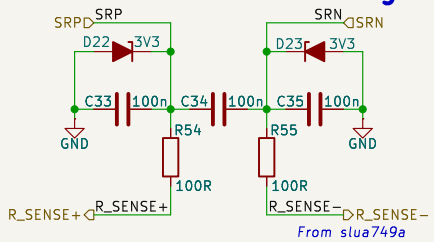
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 8/38

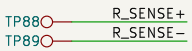


Balancing & BMC (AFE)

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: 9/38

Current Sense Filtering

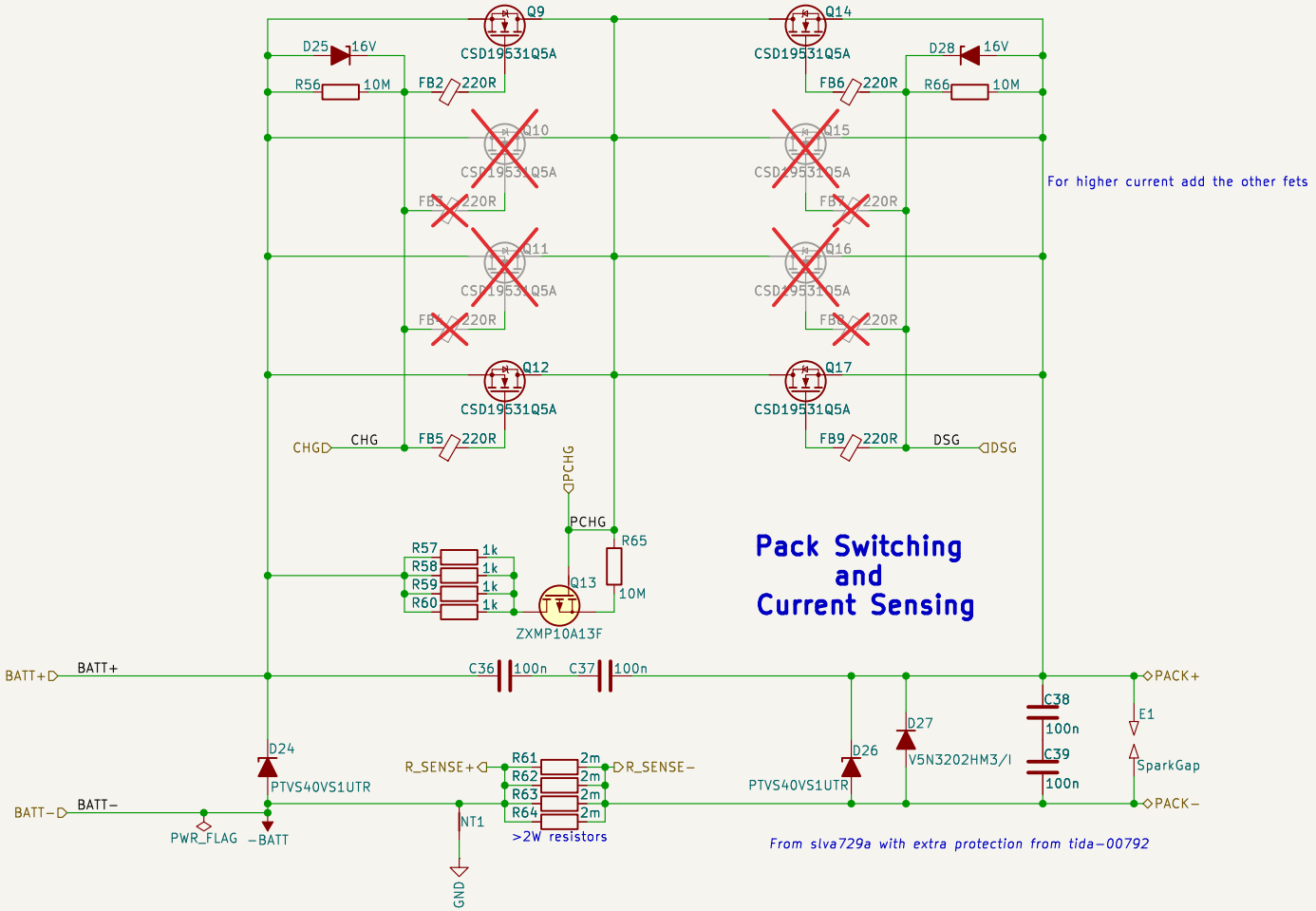




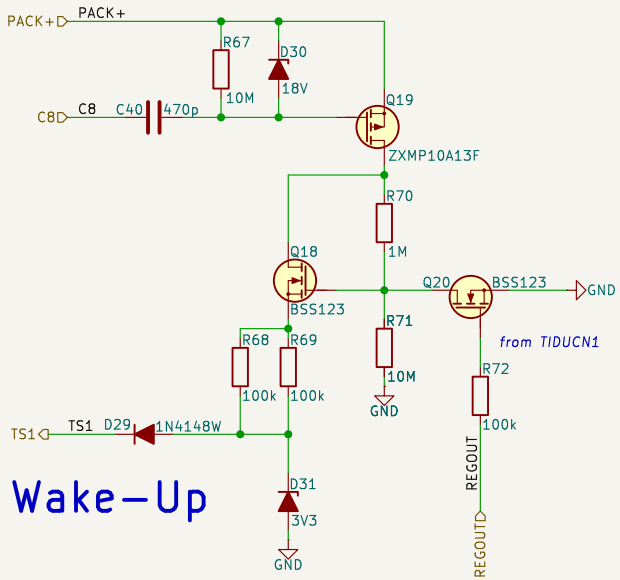
Sheet: /BMS Protection Board/Current Sense Filtering/
File: Current Sense Filtering.kicad_sch

Title:

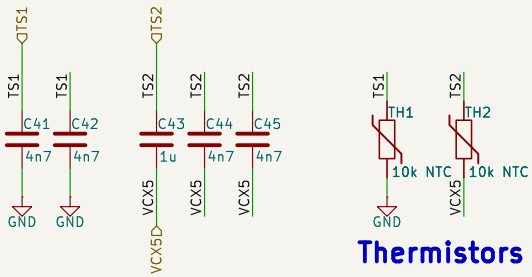
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 10/38



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: 11/38

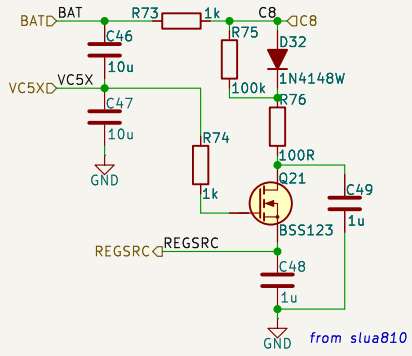


Sheet: /BMS Protection Board/Wake Up/		
File: Wake Up.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 12/38



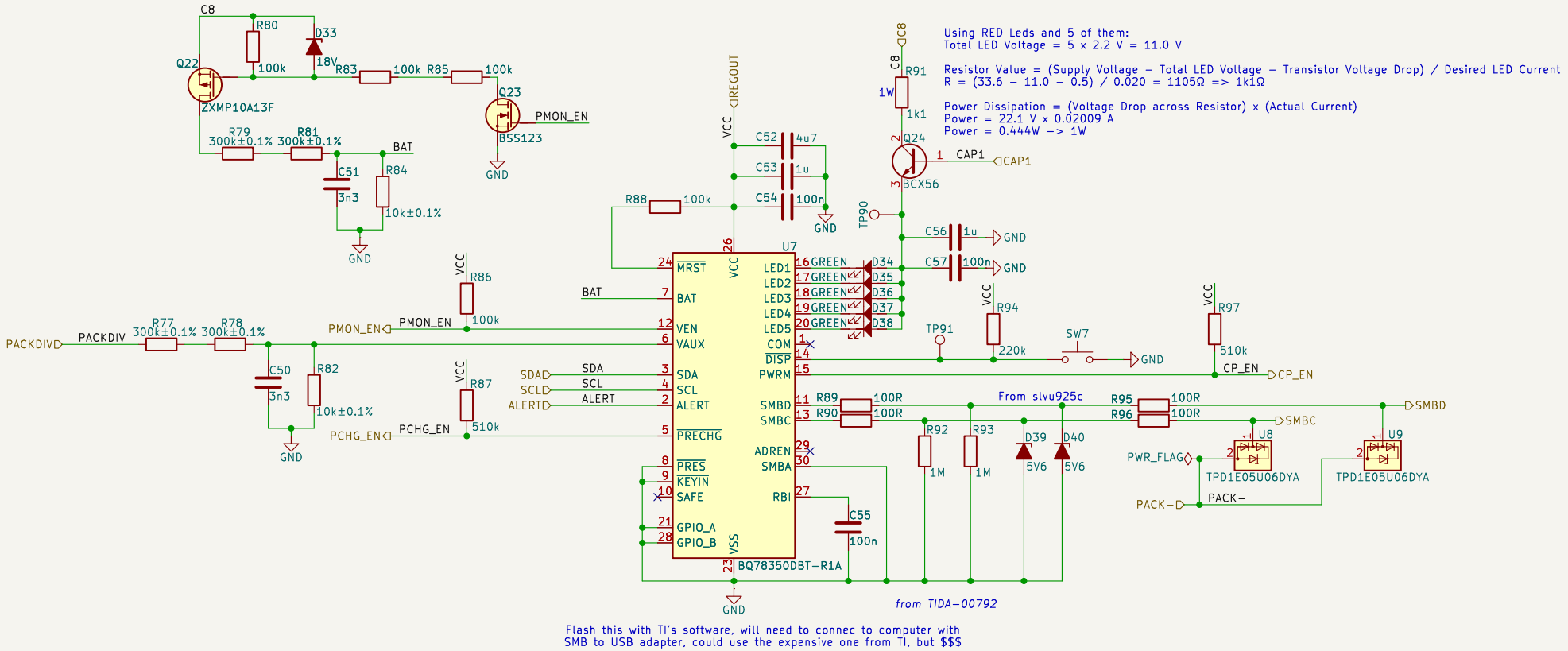
Sheet: /BMS Protection Board/BMS Thermistors/		
File: BMS Thermistors.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 13/38

High Voltage REGSRC Supply



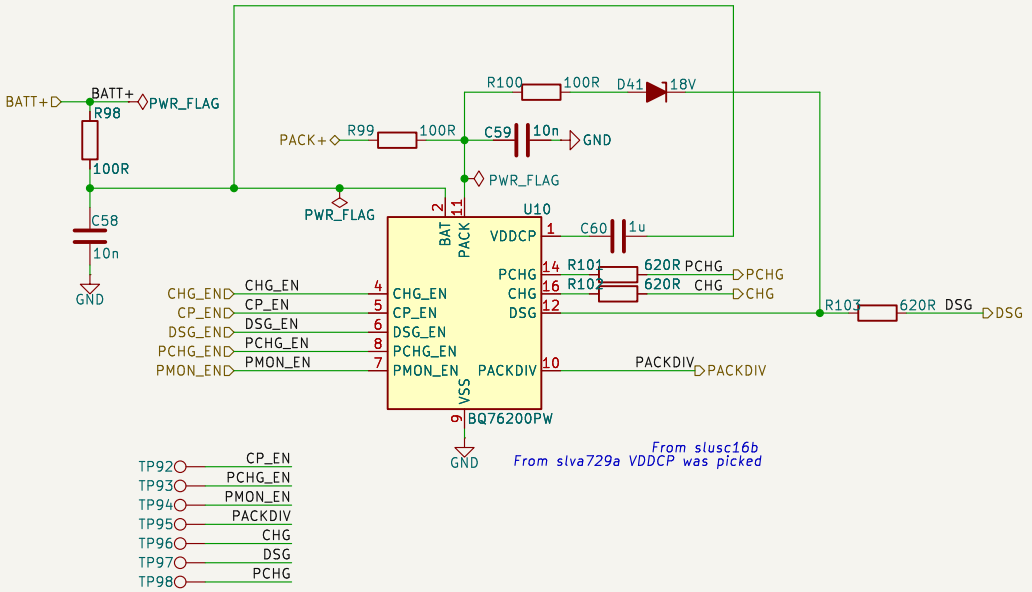
Sheet: /BMS Protection Board/High Voltage REGSRC Supply/		
File: High Voltage REGSRC Supply.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 14/38

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: 15/38

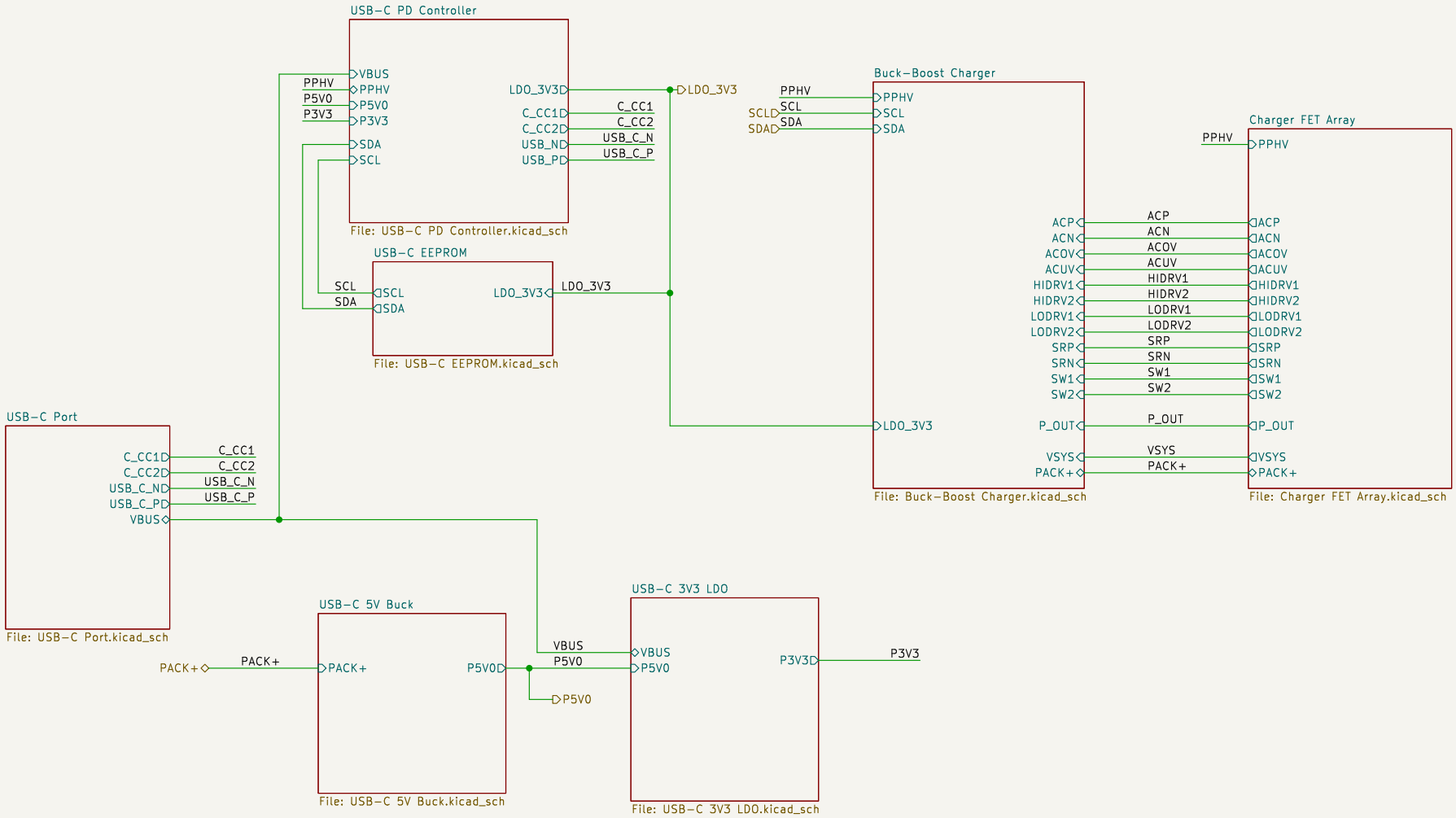
High-Side Gate Driver



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: 16/38



Design is based off of PMP41062

The Correct Flashing Process

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.

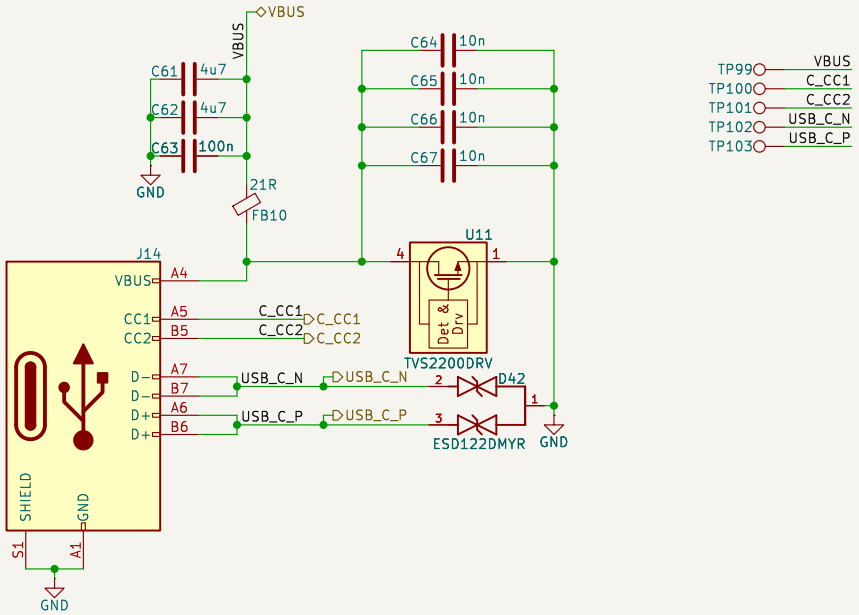
Assemble the Final Circuit: Once the EEPROM is successfully flashed, connect it to the I2Cc (the controller port) of the TPS25751.

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:

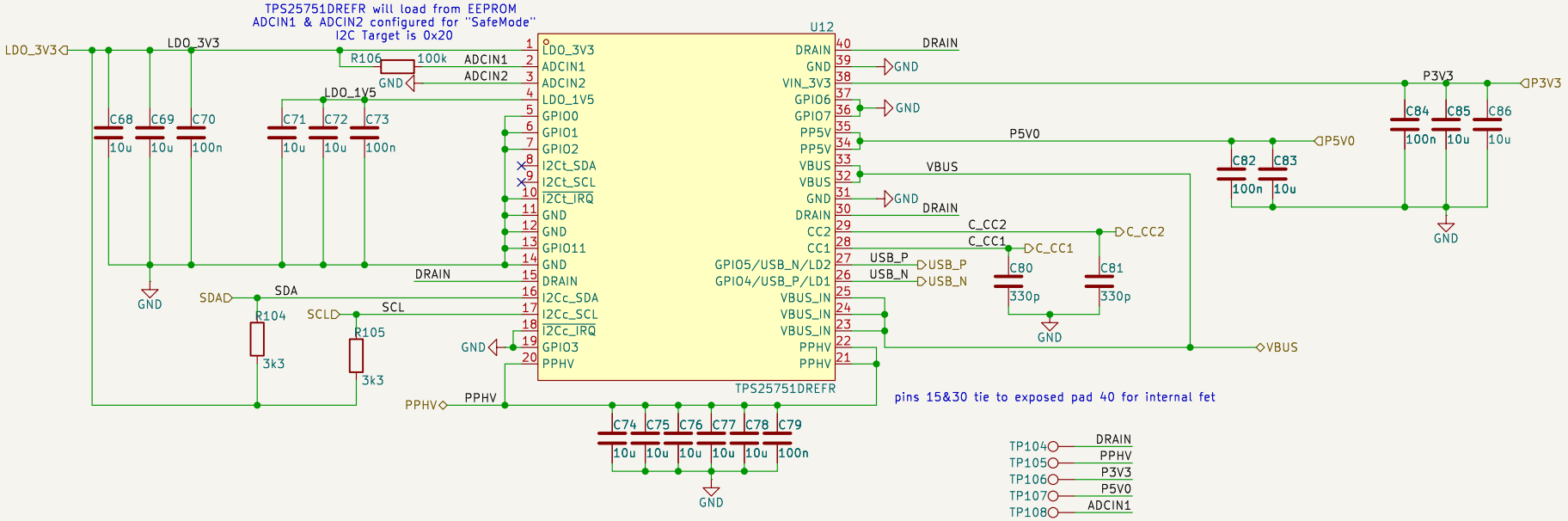
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 17/38



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: 18/38



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

Title:

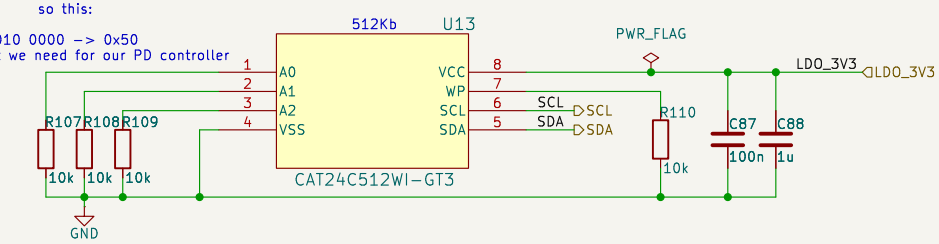
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 19/38

1010 ????
GND=0

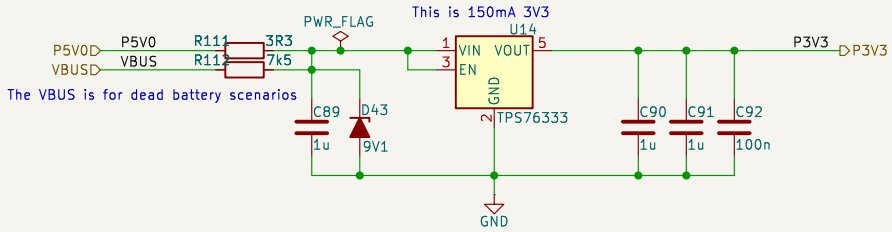
so this:

1010 0000 -> 0x50
which is what we need for our PD controller

Flash this with the bin taken from TI's website. Use I2C to USB adapter.
Flash this when the system is off (not powered), only flash/power this.



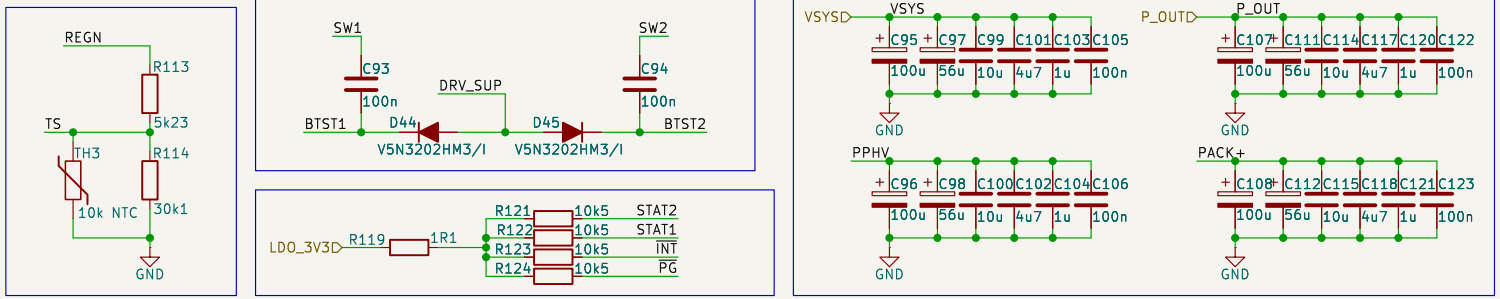
Sheet: /USB-C Power Delivery/USB-C EEPROM/		
File: USB-C EEPROM.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 20/38



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: 21/38



these should be precision resistors

U15 BQ25756RRVR

PWR_FLAG

"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+ R116 249k±0.1% R117 11k96±0.1%

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

R115 11k36±0.1% R118 2k49±0.1% R120 100k±0.1% R125 0R

VAC VAC BTST1 BTST2 DRV_SUP HIDRV1 SW1 LODRV1 HIDRV2 SW2 LODRV2 REGN INT STAT1 STAT2 PG SCL SDA EXP PGND PGND NC NC NC

PPHV C109 4u7 C113 1u C116 1u C119 100n

C110 4u7

GND

TP109 VSYS
TP110 P_OUT
TP111 ACP
TP112 ACN
TP113 SRP
TP114 SRN
TP115 FB
TP116 FBG
TP117 TS
TP118 ICHG
TP119 ILIM_HIZ
TP120 ACUV
TP121 ACOV
TP122 FSW_SYNC
TP123 CE
TP124 BTST1
TP125 BTST2
TP126 DRV_SUP
TP127 HIDRV1
TP128 SW1
TP129 LODRV1
TP130 HIDRV2
TP131 SW2
TP132 LODRV2
TP133 REGN
TP134 INT
TP135 STAT1
TP136 STAT2
TP137 SCL
TP138 SDA
TP139 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 = 50AkΩ / 4.4A = 11.36kΩ

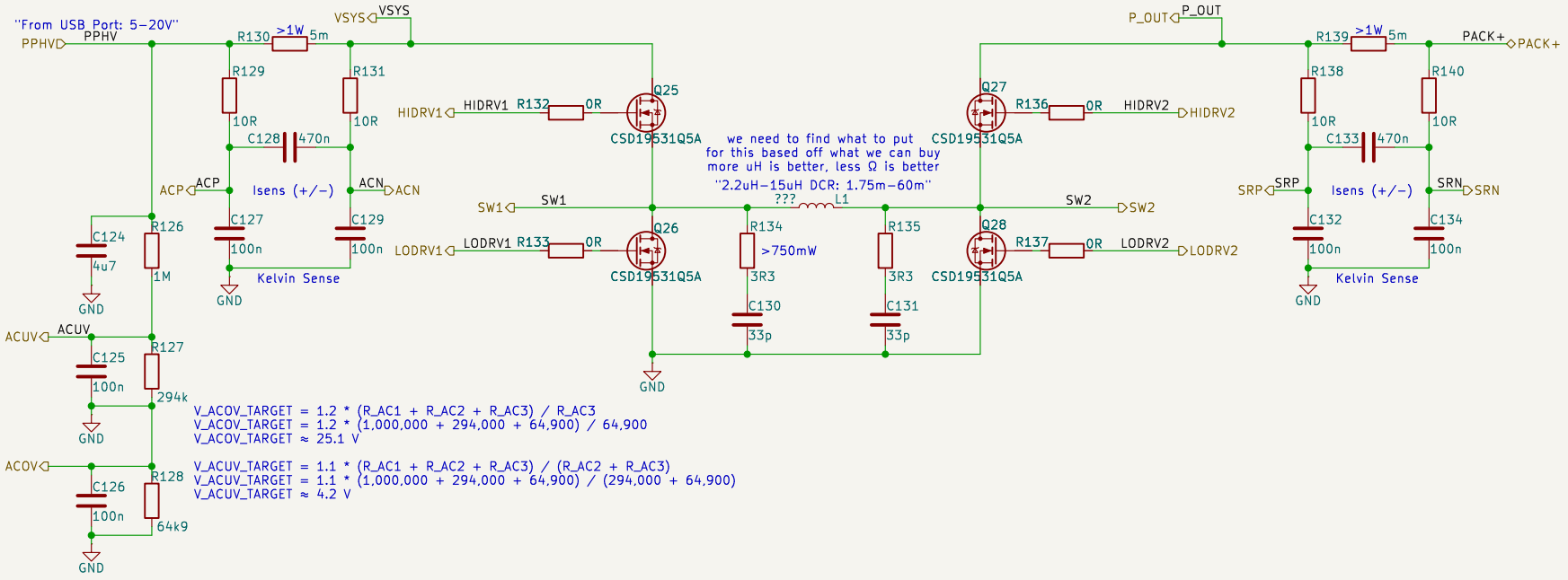
Since this is 100W, 20V/5A,
a safe upper will be set for 8A

using 5mΩ sense resistor:
R = 20AkΩ / 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: 22/38



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: 23/38

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

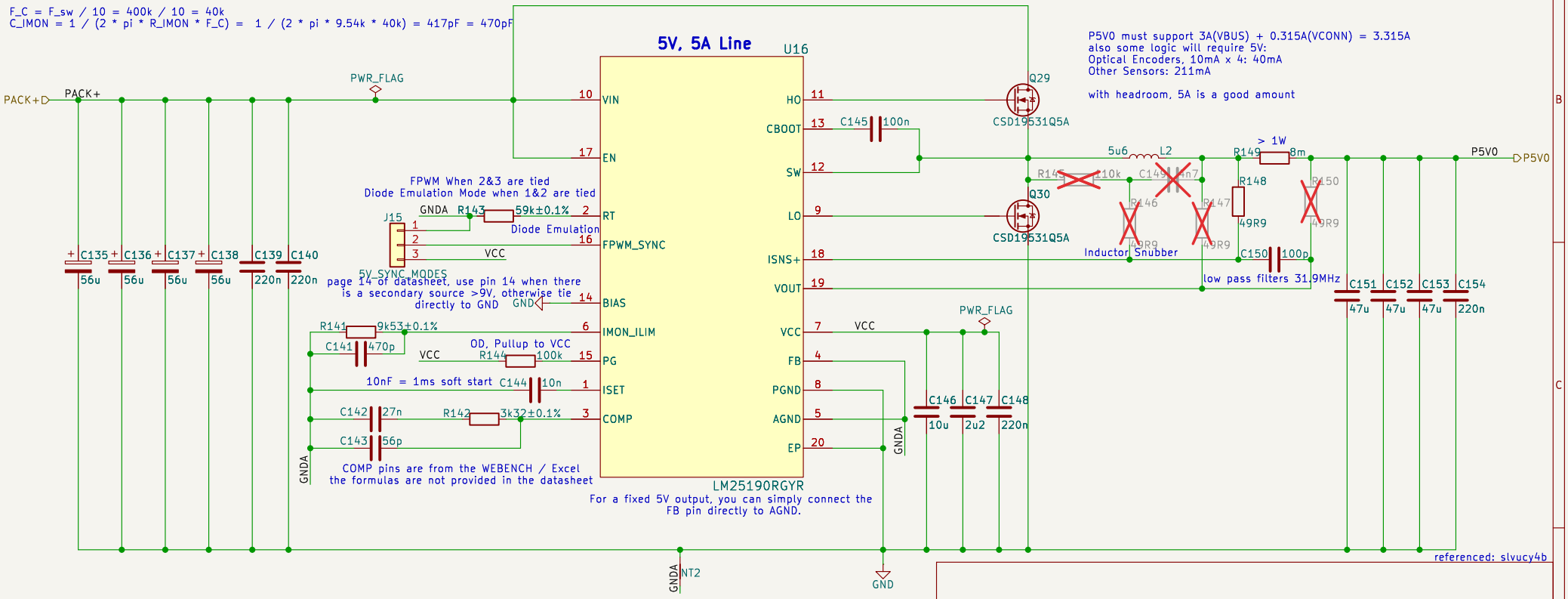
I_CO(RMS) = Delta_I_L / 12^(1/2) = 0.55A
*** The Capacitor Bank must be rated for > 0.55A for its ripple current

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

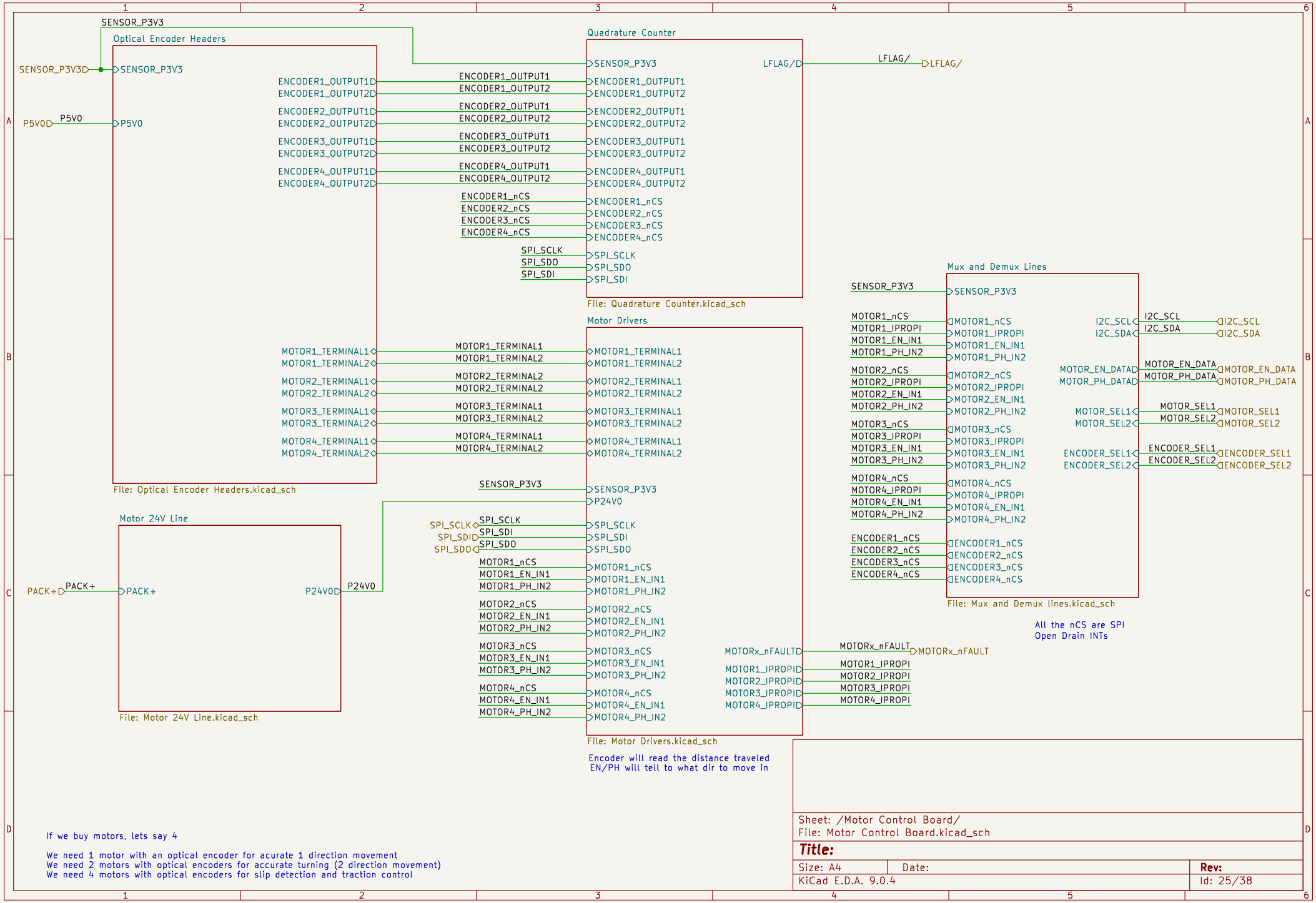
Soft Start:
* From datasheet: I_ISET (internal) is 10uA
* Delta_V = 1V (voltage ramp from 0 to 1)
I = Delta_Q / Delta_T , C = Delta_Q / Delta_V , Delta_T = (C * Delta_V) / I
T_SSCC = (C_ISET * Delta_V) / I_ISET = (10n * 1) / 10u = 1ms

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: 24/38


SENSOR_P3V3
Optical Encoder Headers
SENSOR_P3V3
P5V0
ENCODER1_OUTPUT1
ENCODER1_OUTPUT2
ENCODER2_OUTPUT1
ENCODER2_OUTPUT2
ENCODER3_OUTPUT1
ENCODER3_OUTPUT2
ENCODER4_OUTPUT1
ENCODER4_OUTPUT2
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/
ENCODER1_nCS
ENCODER2_nCS
ENCODER3_nCS
ENCODER4_nCS
SPI_SCLK
SPI_SDO
SPI_SDI
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
Encoder will read the distance traveled
EN/PH will tell to what dir to move in
Motor 24V Line
PACK+
File: Motor 24V Line.kicad_sch
Mux and Demux Lines
I2C_SCL
I2C_SDA
MOTOR_EN_DATA
MOTOR_PH_DATA
MOTOR_SEL1
MOTOR_SEL2
ENCODER_SEL1
ENCODER_SEL2
File: Mux and Demux lines.kicad_sch
All the nCS are SPI
Open Drain INTs
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
Sheet: /Motor Control Board/
File: Motor Control Board.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 25/38

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)

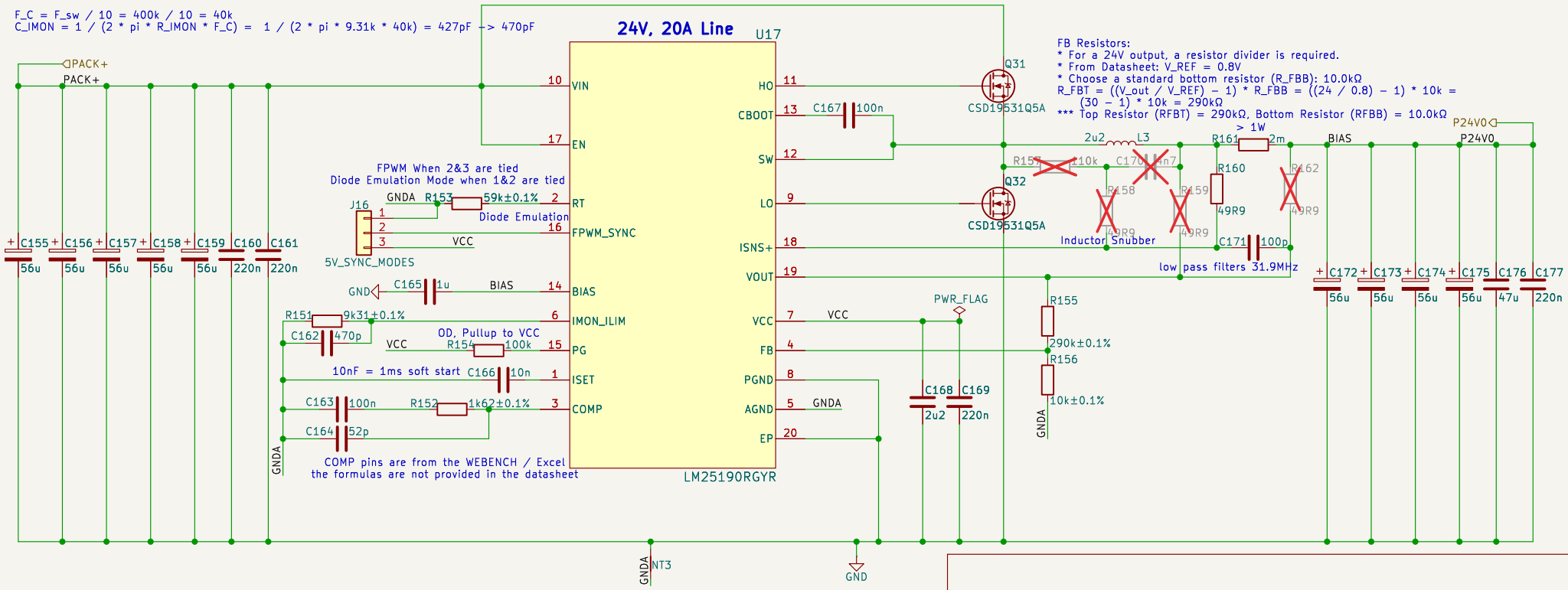
F_C = F_sw / 10 = 400k / 10 = 40k
C_IMON = 1 / (2 * pi * R_IMON * F_C) = 1 / (2 * pi * 9.31k * 40k) = 427pF -> 470pF

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

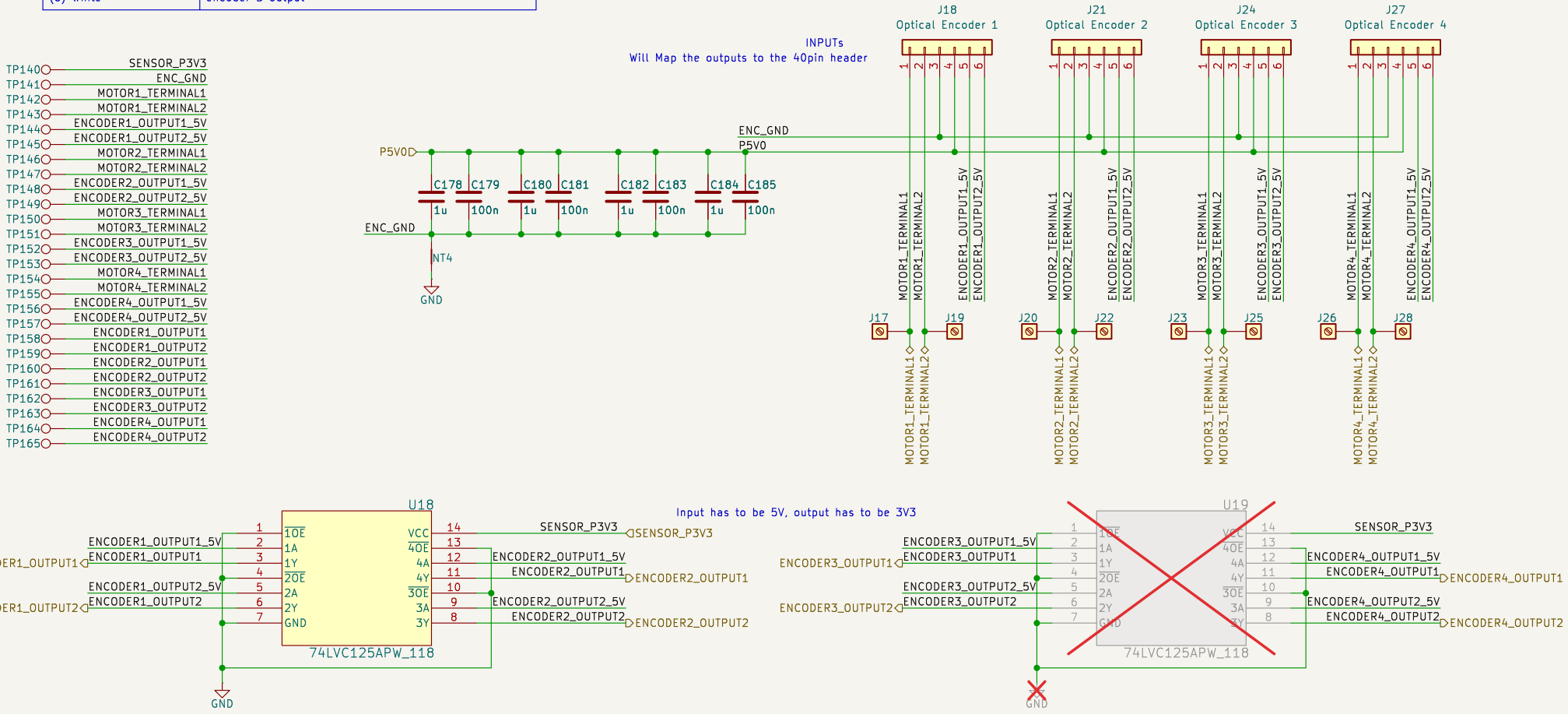
Cin:
Dmax = Vout / Vin(MIN) = 24 / 25.6 = 0.938
Delta_I_L(@Vin_MIN) = 24 / (2.2u * 400k) * (1 - 24 / 25.6) = 1.7A
I_CIN(RMS) = SQRT(D * ((1 - D) * I_out^2 + Delta_I_L^2 / 12)) = SQRT(0.938 * ((1 - 0.938) * 20^2 + 1.7^2 / 12)) = 4.88A(RMS)
Delta_V_Supply = 1% * 25.6V = 0.256V = 0.25V
C_in >= (I_out *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



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: 26/38

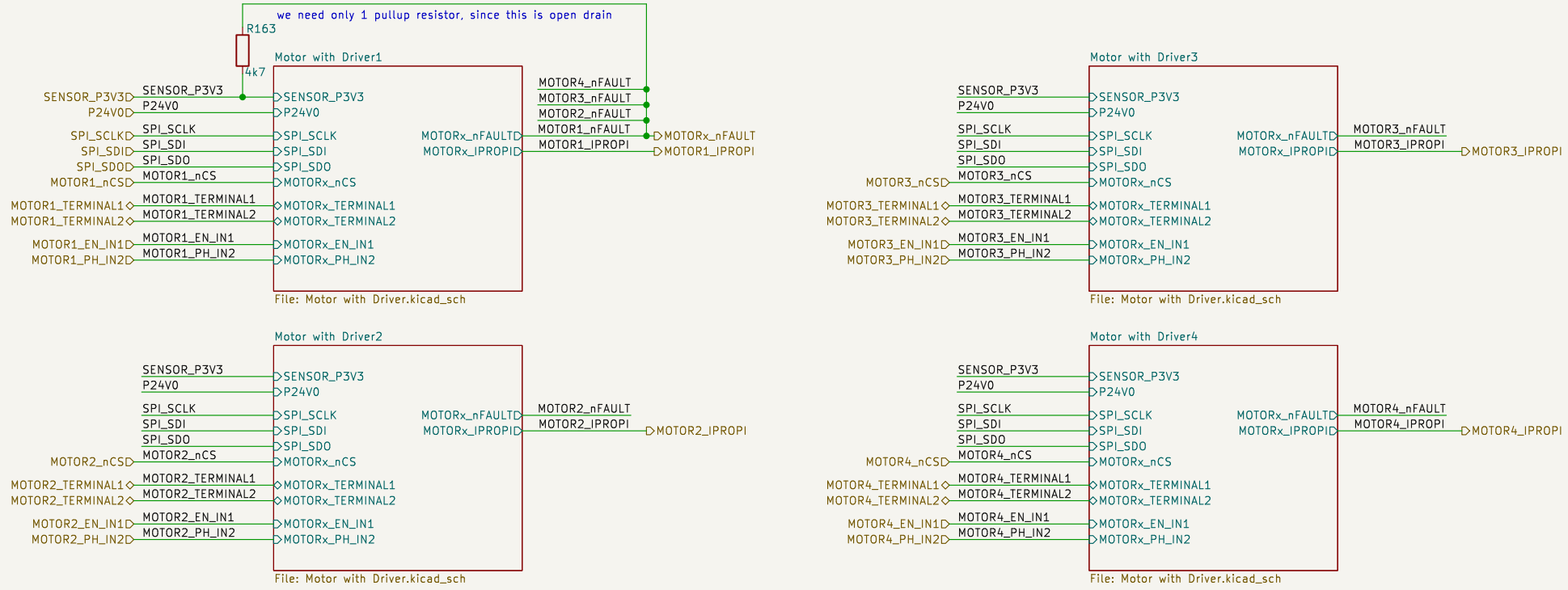
(1) Red	motor power (connects to one motor terminal)
(2) Black	motor power (connects to the other motor terminal)
(3) Green	encoder GND
(4) Blue	encoder Vcc (3.5 – 20 V)
(5) Yellow	encoder A output
(6) White	encoder B output



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: 27/38



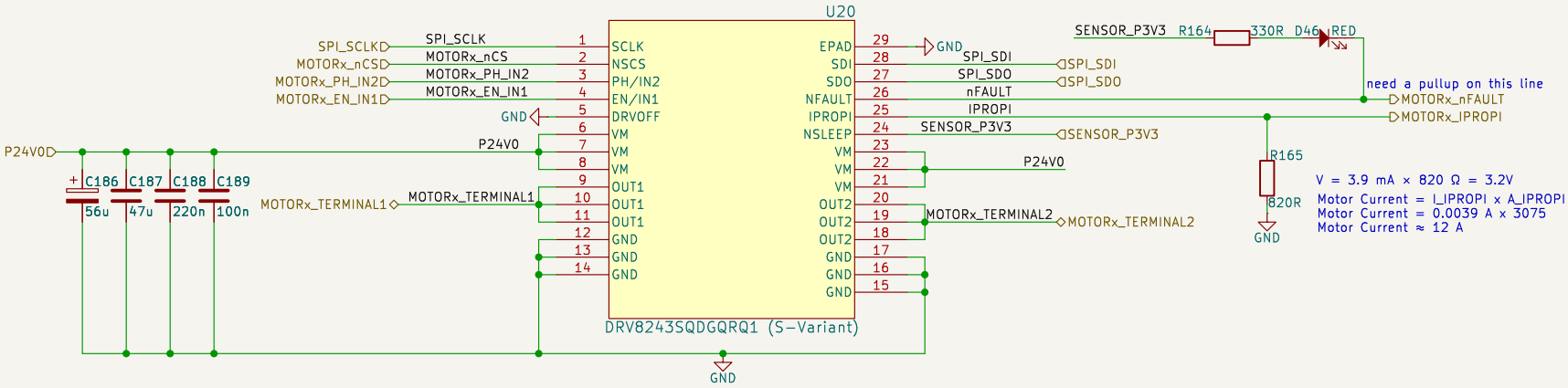
Sheet: /Motor Control Board/Motor Drivers/
File: Motor Drivers.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 28/38

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.



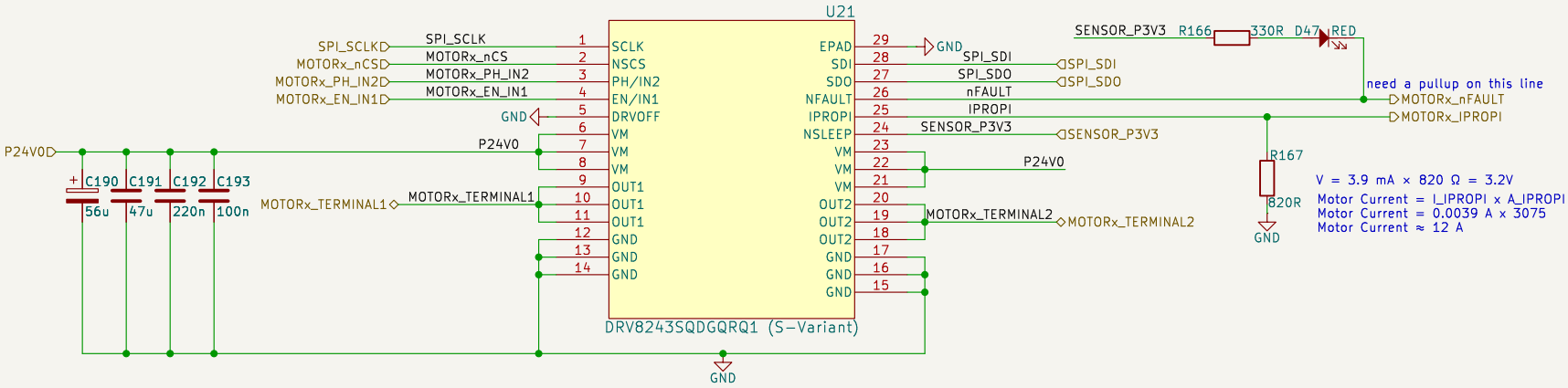
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver1/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 29/38

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.



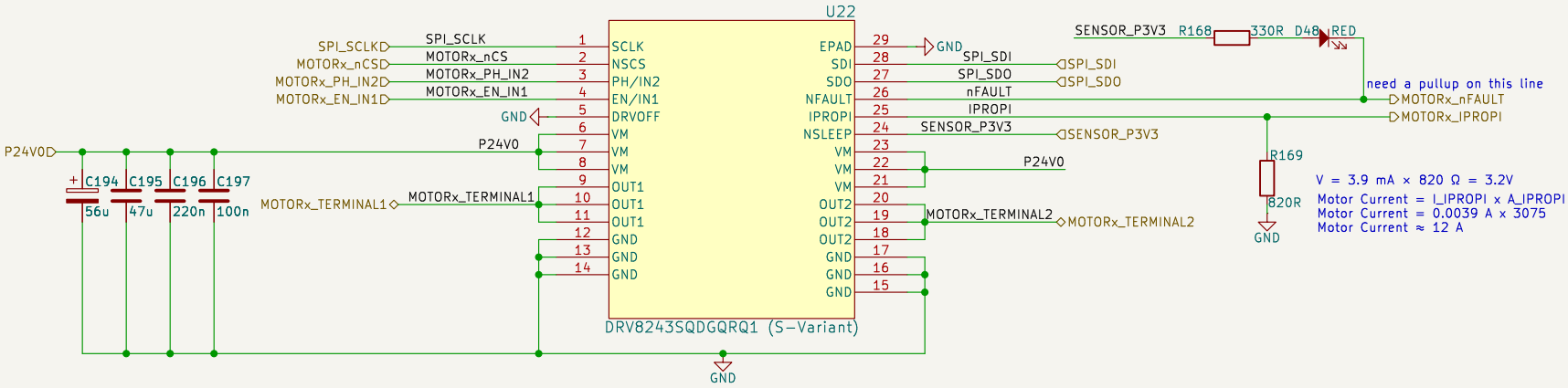
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver2/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 30/38

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.



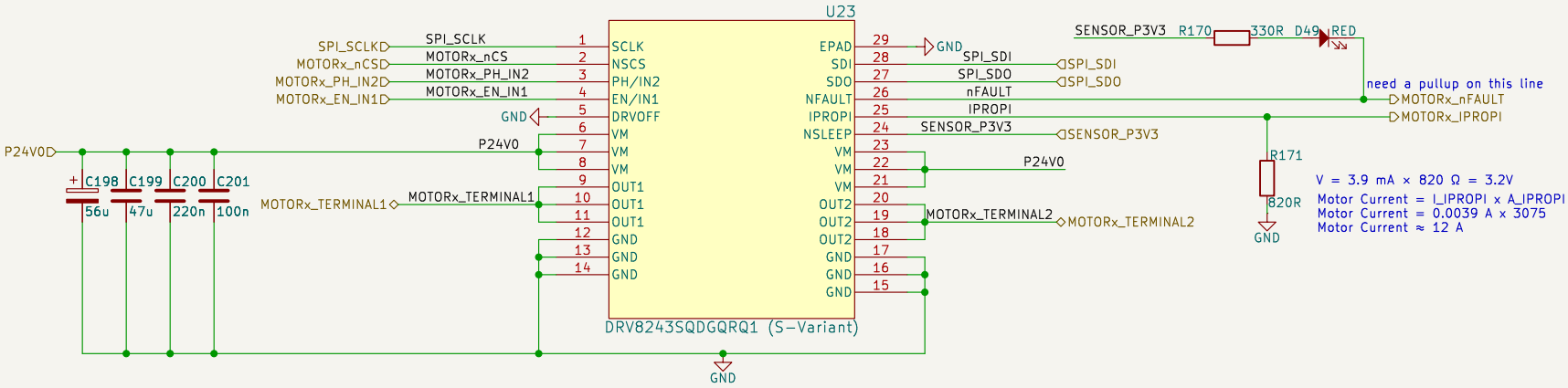
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver3/
File: Motor with Driver.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 31/38

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

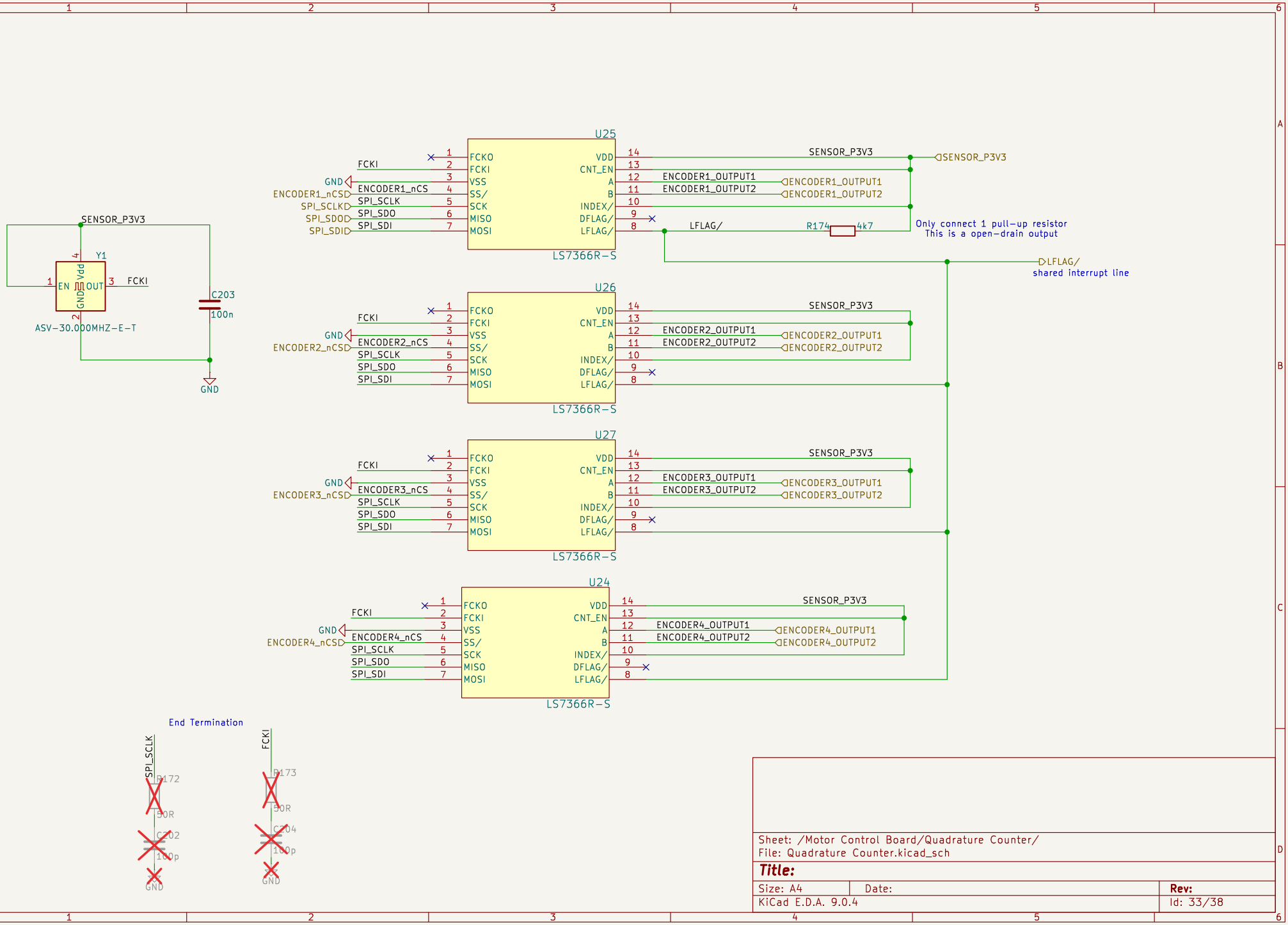
For the SPI (S) variant, the maximum is 10 MHz, but you should reduce it to 8 MHz if you have a 20 pF load on the SDO line.

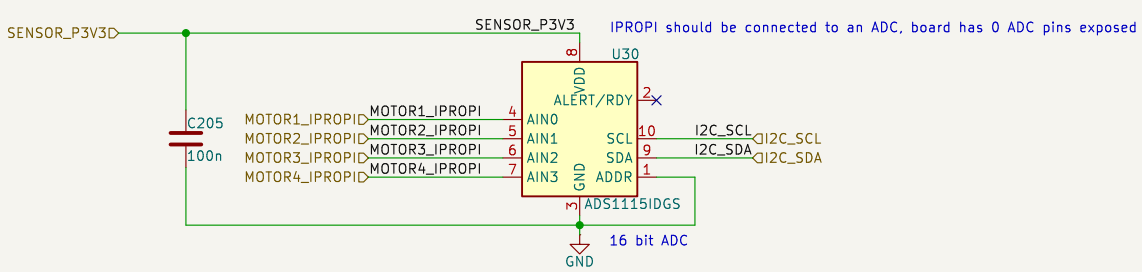


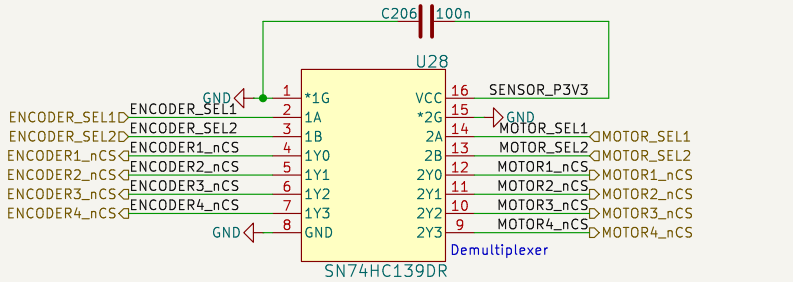
Sheet: /Motor Control Board/Motor Drivers/Motor with Driver4/
File: Motor with Driver.kicad_sch

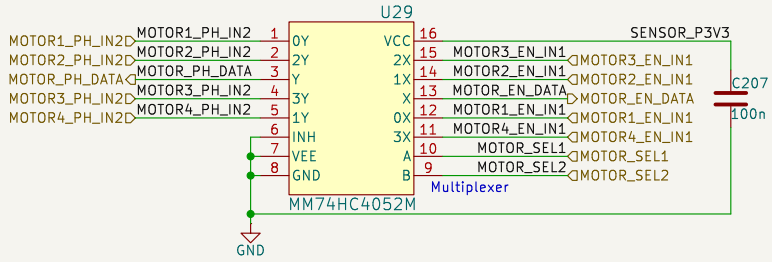
Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 32/38


SENSOR_P3V3
Y1
EN
GND
Vdd
OUT
FCKI
ASV-30.000MHZ-E-T
C203
100n
GND
U25
FCKO
FCKI
VSS
SS/
SCK
MISO
MOSI
VDD
CNT_EN
A
B
INDEX/
DFLAG/
LFLAG/
ENCODER1_nCS
SPI_SCLK
SPI_SDO
SPI_SDI
ENCODER1_OUTPUT1
ENCODER1_OUTPUT2
LFLAG/
R174
4k7
Only connect 1 pull-up resistor
This is a open-drain output
LFLAG/
shared interrupt line
LS7366R-S
U26
ENCODER2_nCS
ENCODER2_OUTPUT1
ENCODER2_OUTPUT2
U27
ENCODER3_nCS
ENCODER3_OUTPUT1
ENCODER3_OUTPUT2
U24
ENCODER4_nCS
ENCODER4_OUTPUT1
ENCODER4_OUTPUT2
End Termination
SPI_SCLK
R172
50R
C202
100p
GND
FCKI
R173
50R
C204
100p
GND
Sheet: /Motor Control Board/Quadrature Counter/
File: Quadrature Counter.kicad_sch
Title:
Size: A4
Date:
Rev:
KiCad E.D.A. 9.0.4
Id: 33/38



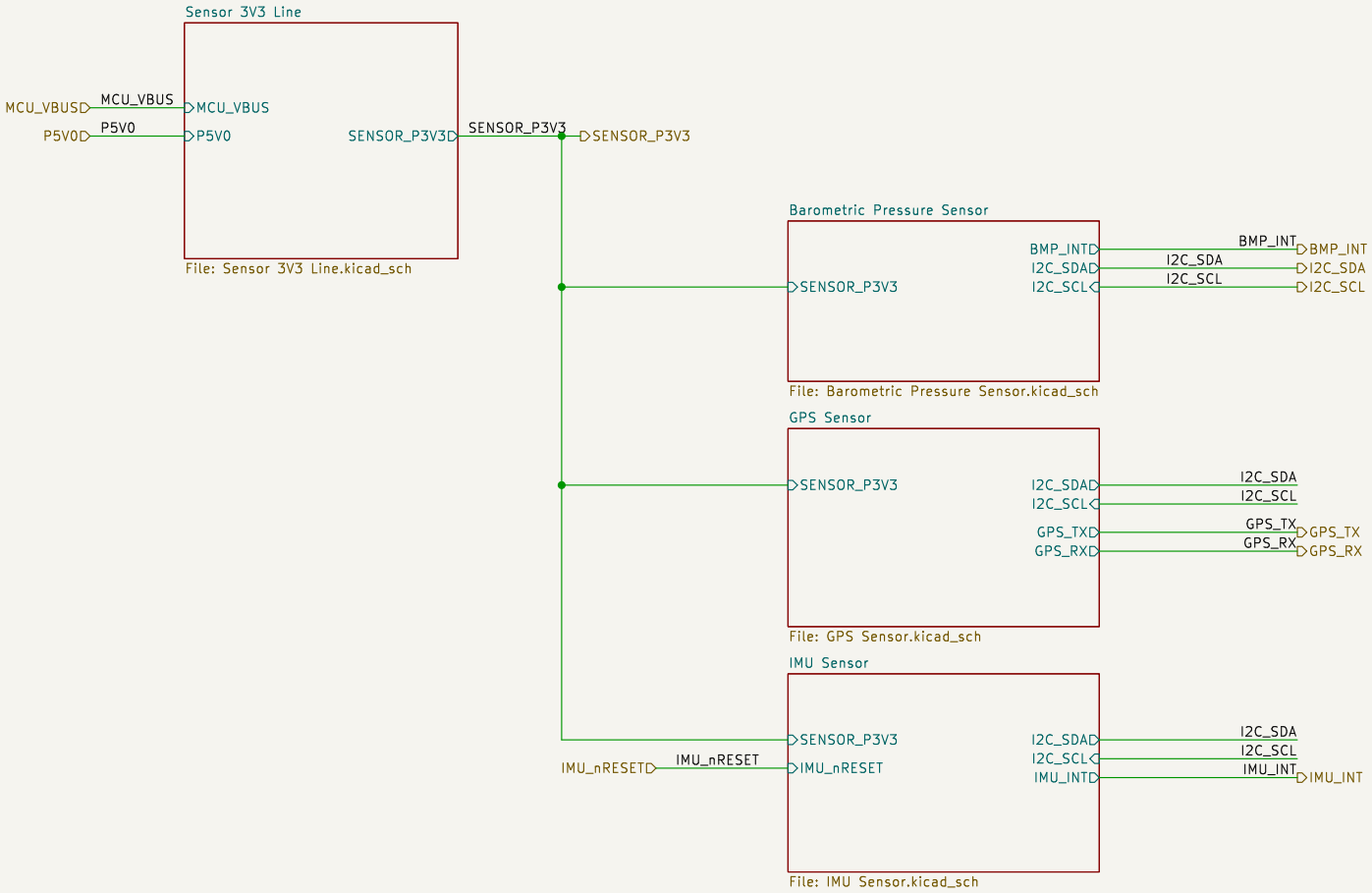




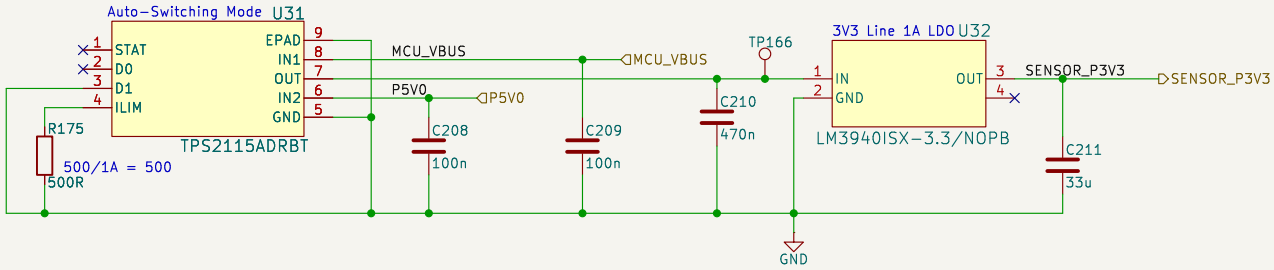
Sheet: /Motor Control Board/Mux and Demux Lines/
File: Mux and Demux lines.kicad_sch

Title:

Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 34/38



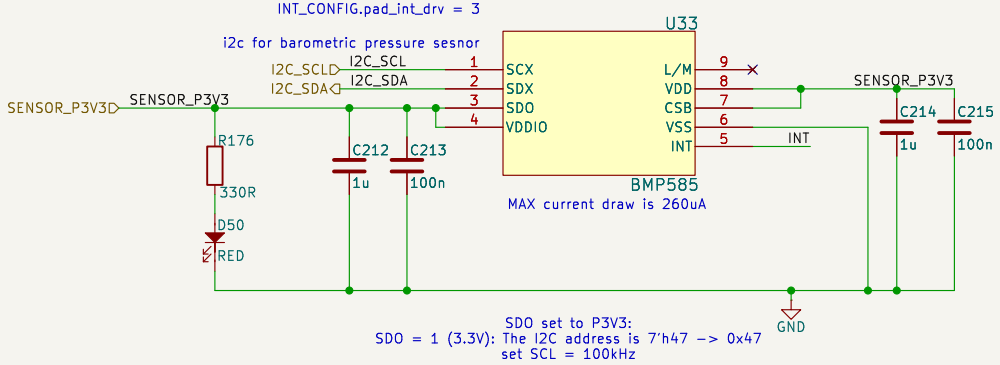
Sheet: /Sensor Board/		
File: Sensor Board.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 35/38

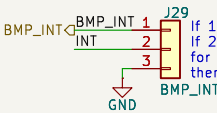


Sheet: /Sensor Board/Sensor 3V3 Line/		
File: Sensor 3V3 Line.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 36/38

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).

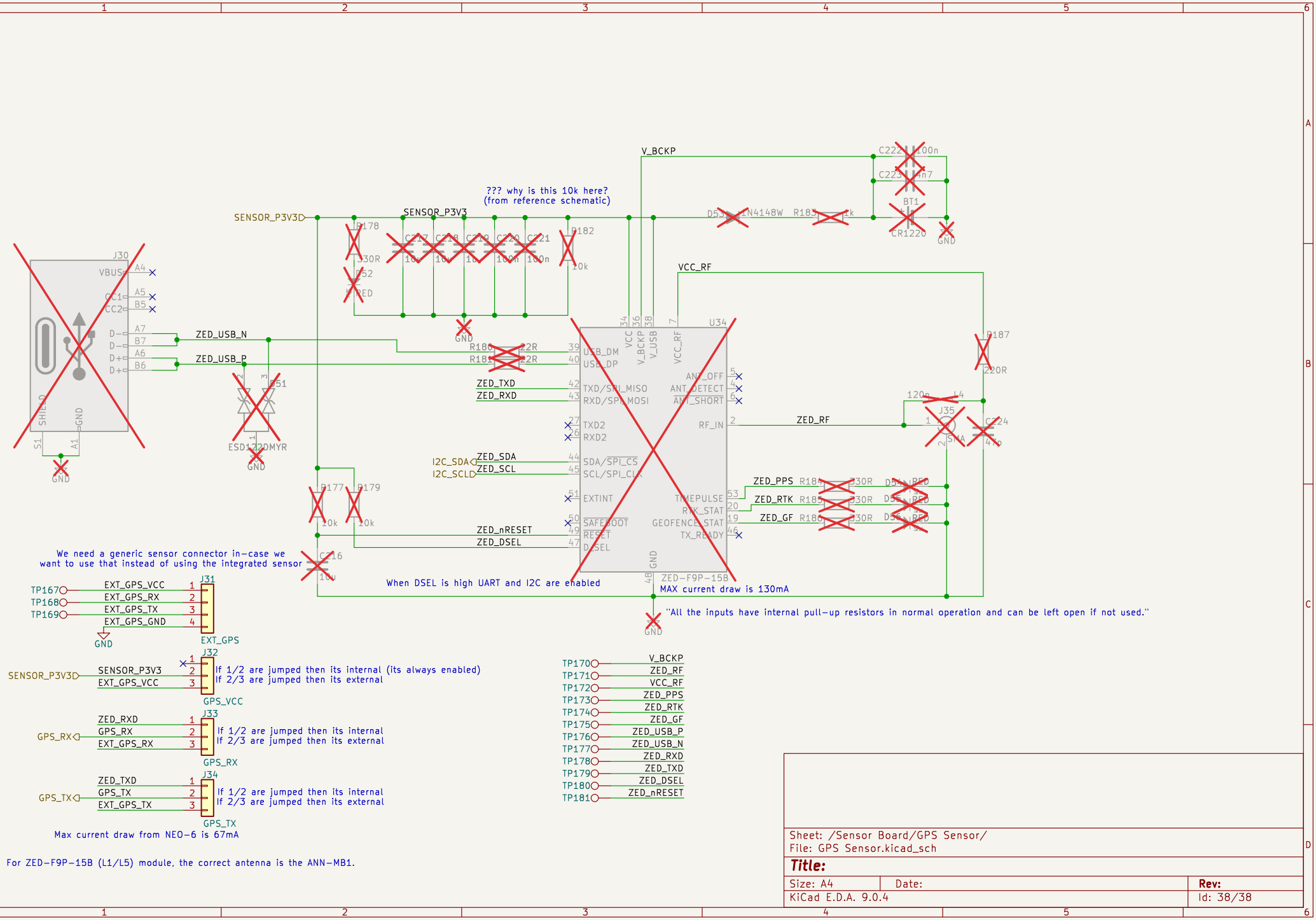
Datasheet sets the IRQ to 0 on start up, so never turn it on
PAGE49: 0x14 on reset is set to 0x35
PAGE49: <int_en 2bits> <int_od 2bits> <int_pol 2bits> <int_mode 2bits LSB>
PAGE51: 8.5, setup for 0x14, <0> <1> <0> <1>
this means: int_mode = latched, int_pol = active low, int_od = open_drain, int_en =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

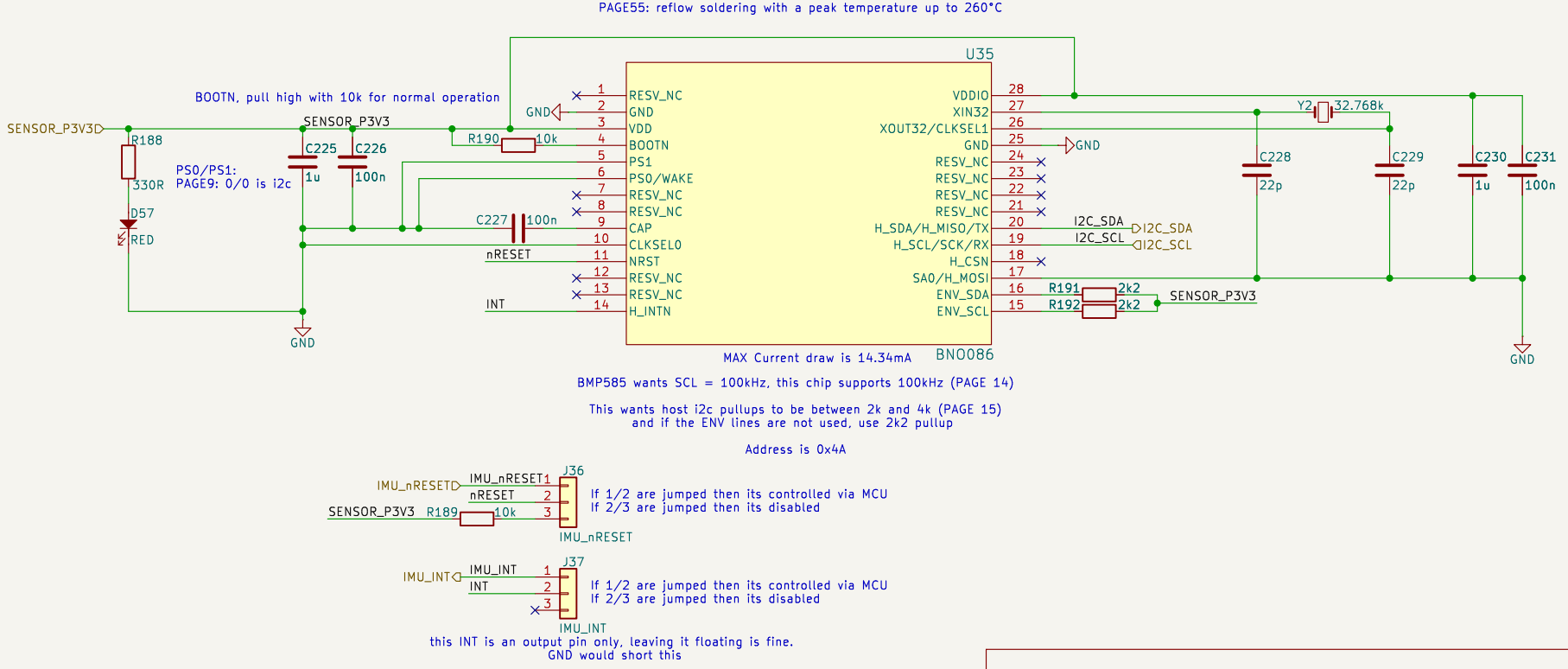




If 1/2 are jumped then its enabled
If 2/3 are jumped then its disabled ---- could be dangerous, read above
for more info on ^^^ we are safe on reset, but if we program it wrong
then we can break the chip

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: 37/38


??? why is this 10k here?
(from reference schematic)
SENSOR_P3V3
V_BCKP
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
EXT_GPS
If 1/2 are jumped then its internal (its always enabled)
If 2/3 are jumped then its external
GPS_VCC
GPS_RX
GPS_TX
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: 38/38



Sheet: /Sensor Board/IMU Sensor/		
File: IMU Sensor.kicad_sch		
Title:		
Size: A4	Date:	**Rev:**
KiCad E.D.A. 9.0.4		Id: 39/38