回路のテンプレ

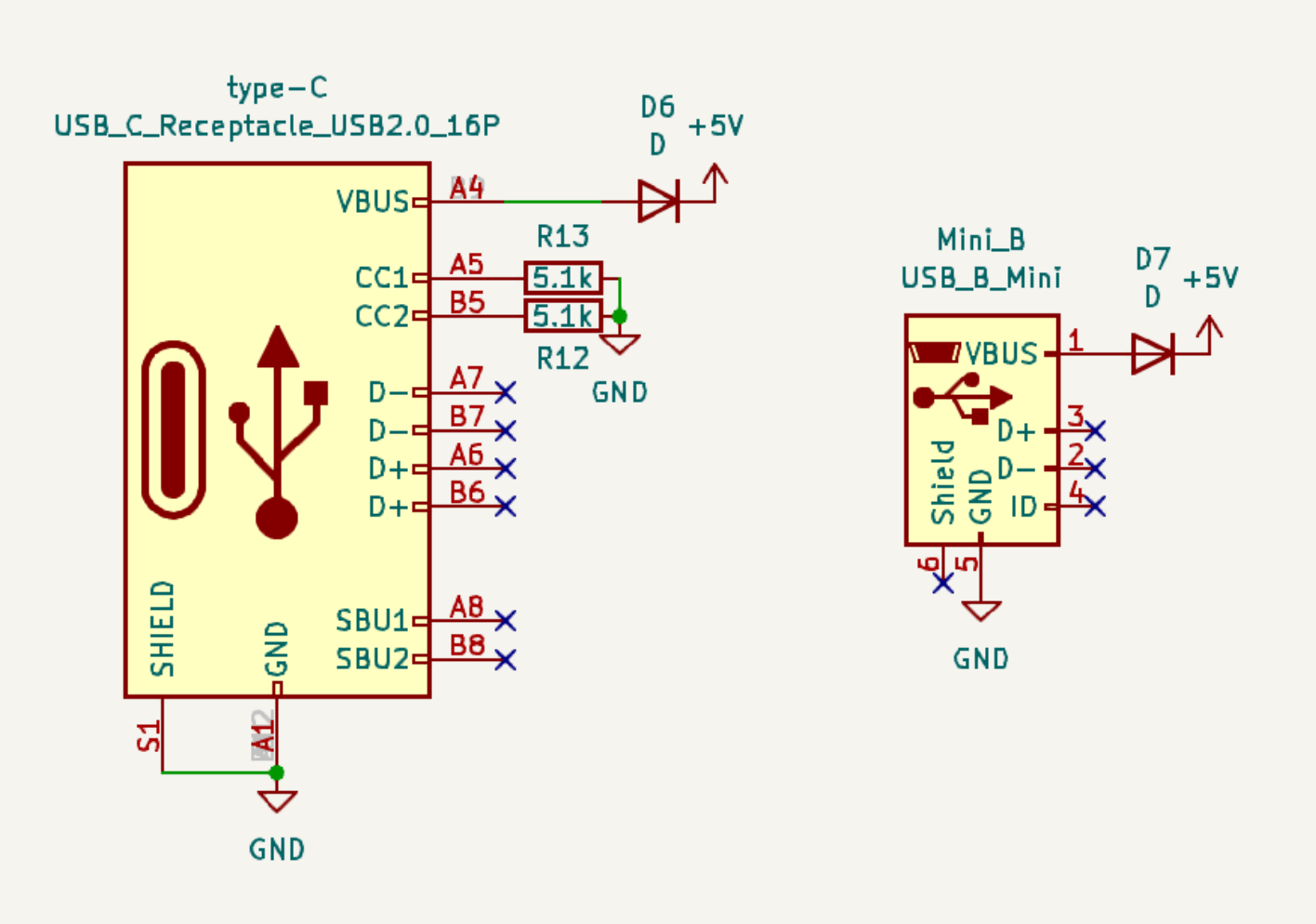
Ver.1

目次

- 電源供給用USB(type-C, mini-B)
- リレー(非常停込み)
- レギュレーター(5V->3.3V)
- STM32
- ST-LINK
- CAN(トランシーバー, コネクター, ロボマス) , 遠隔非常停
- UART
- エンコーダー, Cytron
- LEDたち
- 基板の穴
- ロータリースイッチ

※注
情報は鵜呑みにせず自分で
調べるようにしましょう

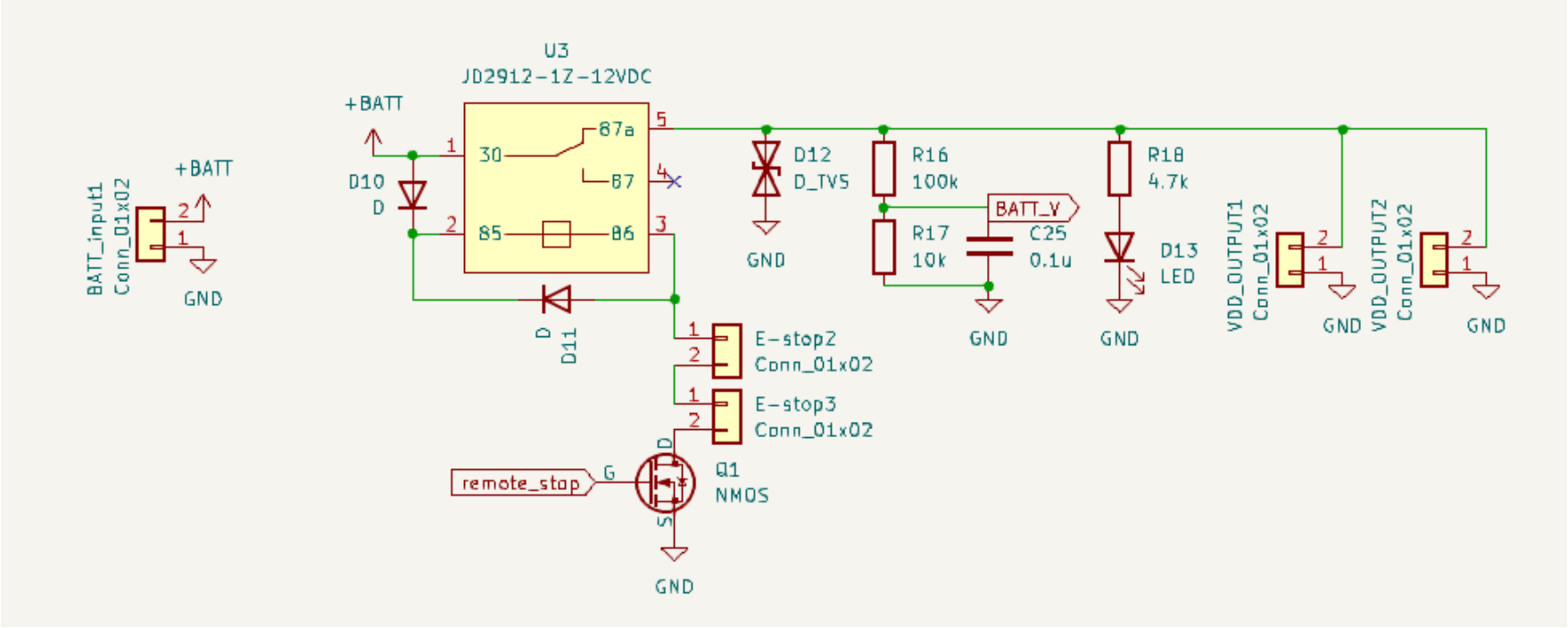
電源供給用USB(type-C, mini-B)



データ通信したかったら
D+,D-につなぐ（調べよう）

リレー

- TVSダイオード(D_TVS)はなくてもよい
- OUTPUTの数は必要数だけ用意
- BATT_input1はリポバッテリーにつなげる
- Remote_stopはマイコンにつなげて遠隔非常停として使う
- BATT_Vもマイコンにつなげて電圧の状態を取得できるようにする



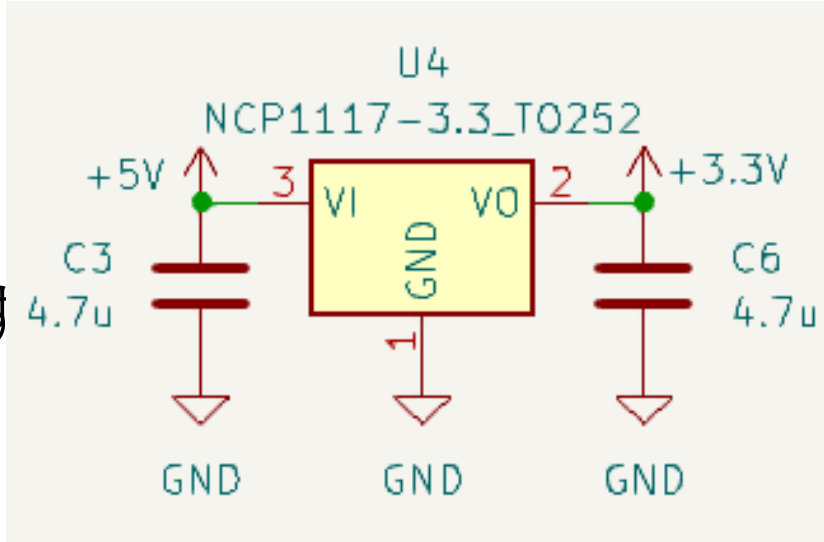
レギュレーター

- 制御電源はモバイルバッテリーから供給している
- モバイルバッテリーは5Vだが、STM32は3.3Vで動かす

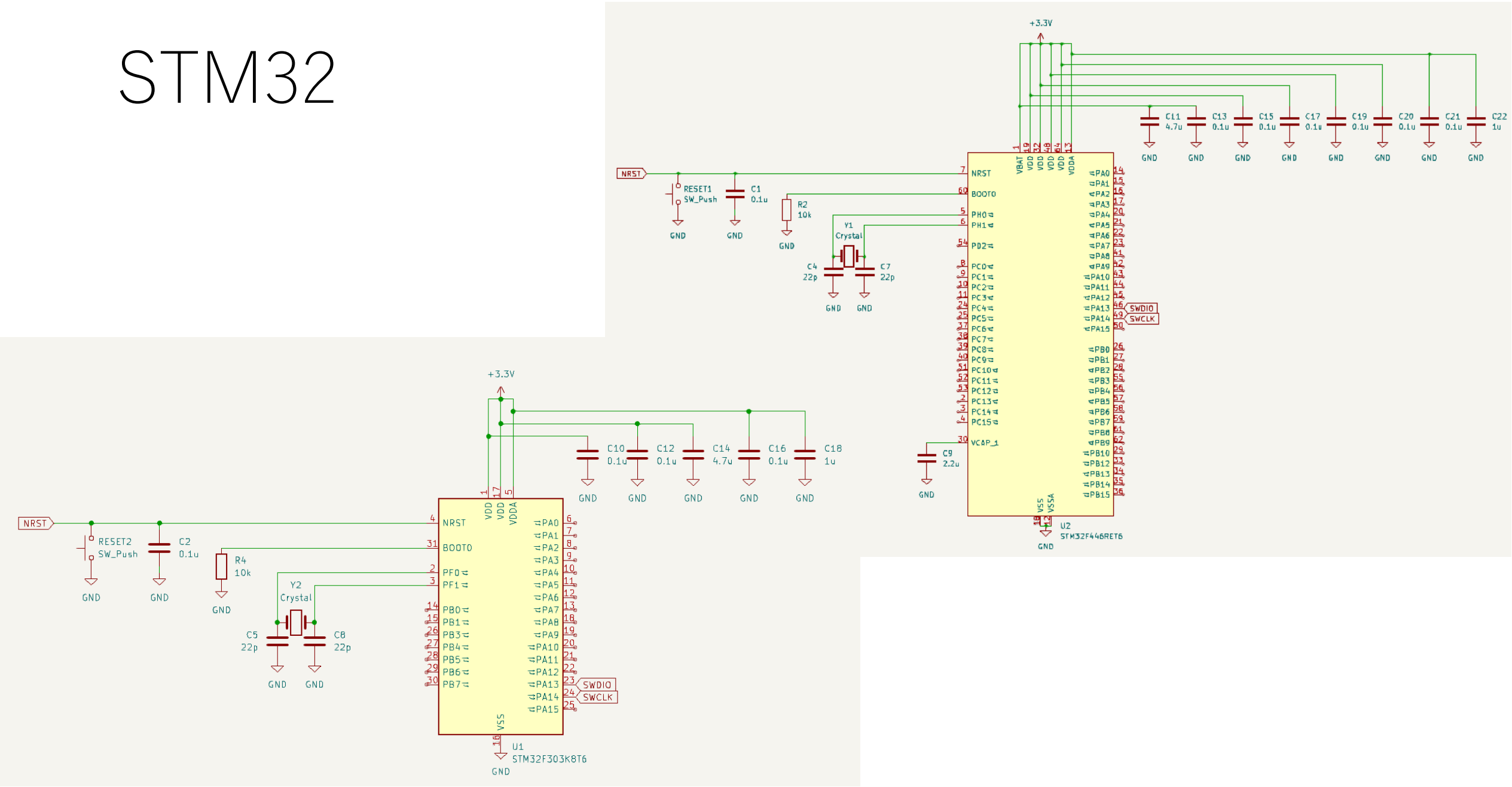
→5Vを3.3Vに降圧したい

- レギュレーターで降圧する
- ロボマスモーターの定格は24V
- マブチモーターの定格は12V

→12Vに降圧するレギュレーターも存在す
（その他5V,8Vなどもある）



STM32

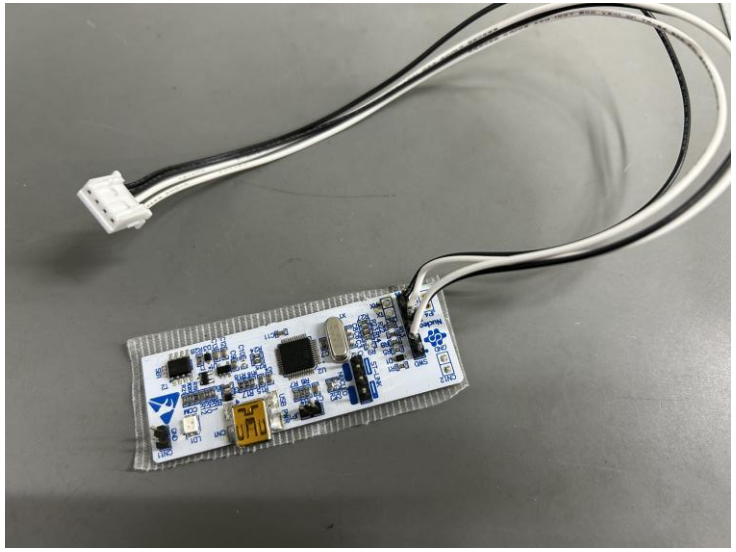

+3.3V
NRST
RESET1
SW_Push
C1
0.1u
R2
10k
Y1
Crystal
C4
22p
C7
22p
GND
C11
4.7u
C13
0.1u
C15
0.1u
C17
0.1u
C19
0.1u
C20
0.1u
C21
0.1u
C22
1u
VBAT
VDD
VDDA
NRST
BOOT0
PH0
PH1
PD2
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
PC8
PC9
PC10
PC11
PC12
PC13
PC14
PC15
VCAP_1
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PA8
PA9
PA10
PA11
PA12
PA13
PA14
PA15
SWDIO
SWCLK
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PB8
PB9
PB10
PB12
PB13
PB14
PB15
C9
2.2u
VSS
VSSA
U2
STM32F446RETB
C10
0.1u
C12
0.1u
C14
4.7u
C16
0.1u
C18
1u
RESET2
SW_Push
C2
0.1u
R4
10k
Y2
Crystal
C5
22p
C8
22p
PF0
PF1
U1
STM32F303K8T6

STM32

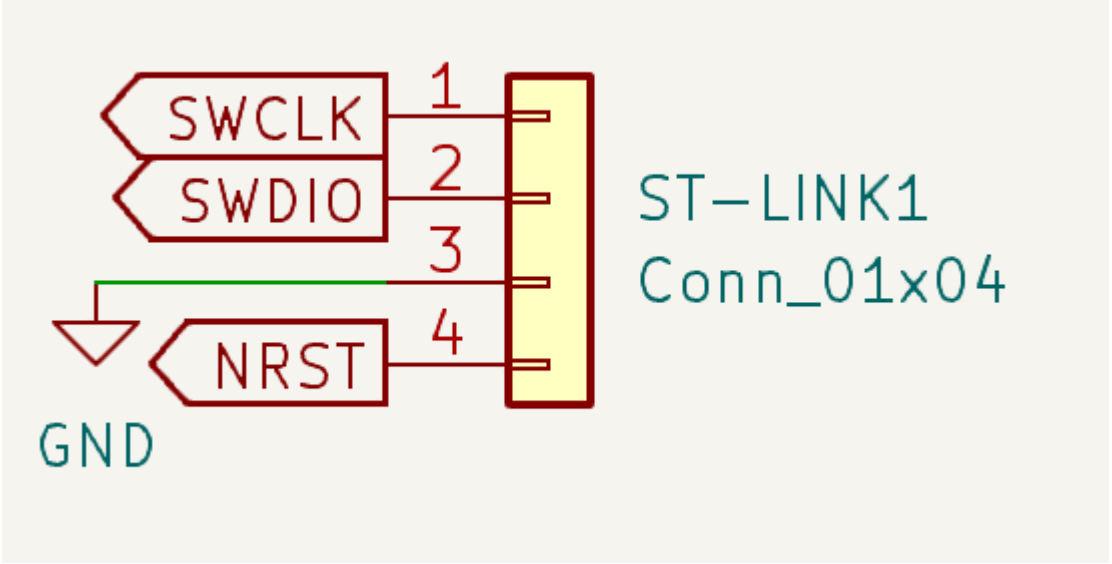
- 電源のコンデンサーは
  VDD:(0.1uF)x1
  VBAT:(0.1uF)x1+(4.7uF)x1
  VDDA:(0.1uF)x1+(1uF)x1
- NRSTはGNDに接続したときに変数をすべてリセットする
- 内部プルアップがあるのでプルアップする必要はない
- NRSTのコンデンサーはチャタリング防止用
- BOOT0の10kΩはSTM32の起動のモードに関係するらしい
  (気にしなくていい)
- VCAPには2.2uFをつなげる
- Crystalはタイマー用
- 困ったらデータシートを見ると良い(STM32〇〇 datasheet)

ST－LINK

- マイコンに書き込むために使う

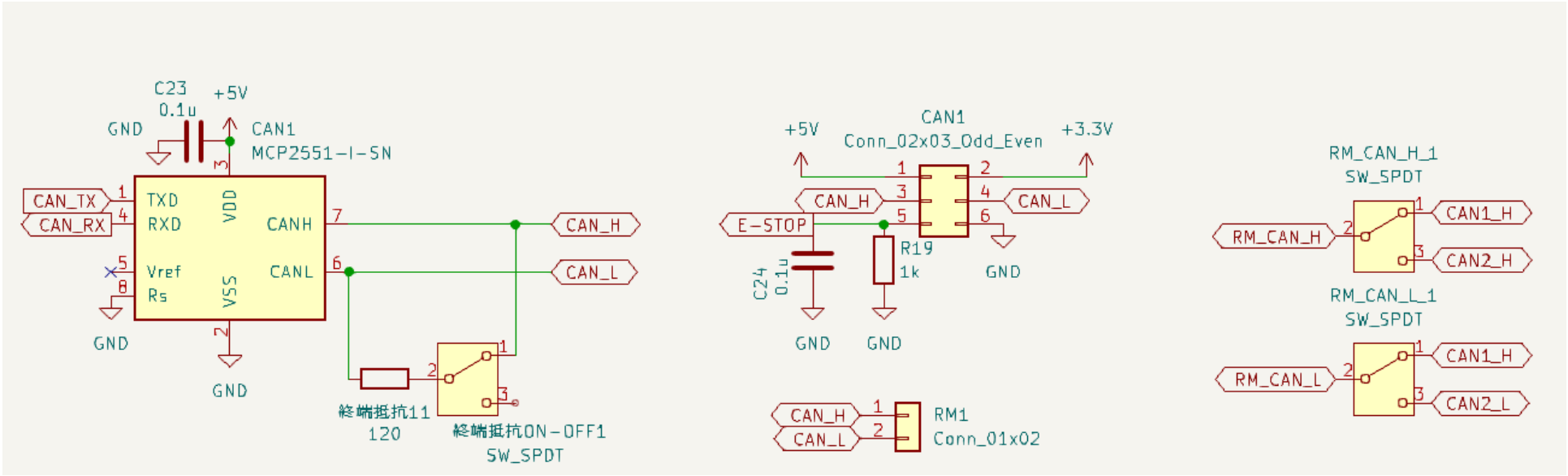


ST-LINK



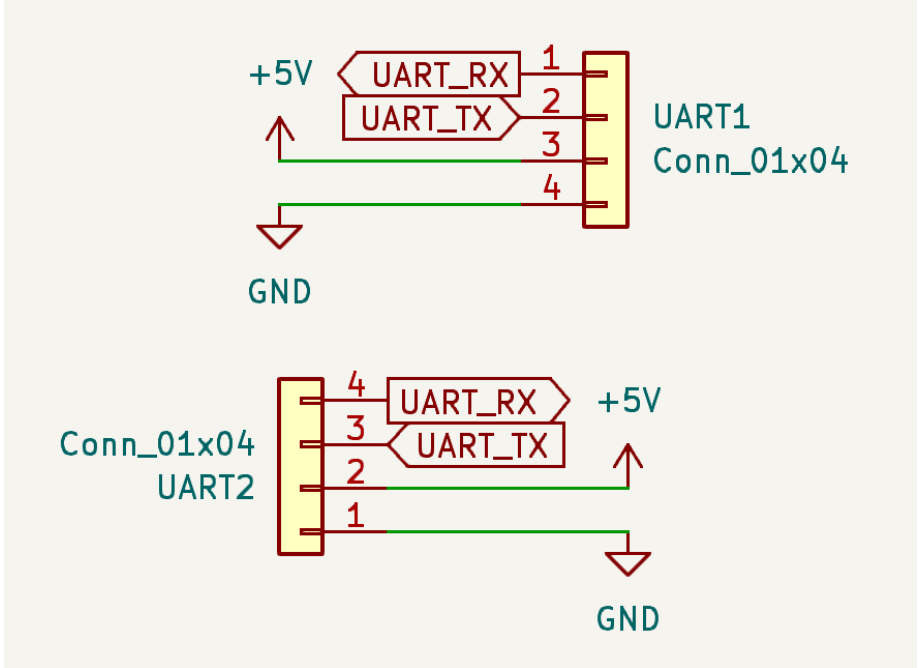
CAN

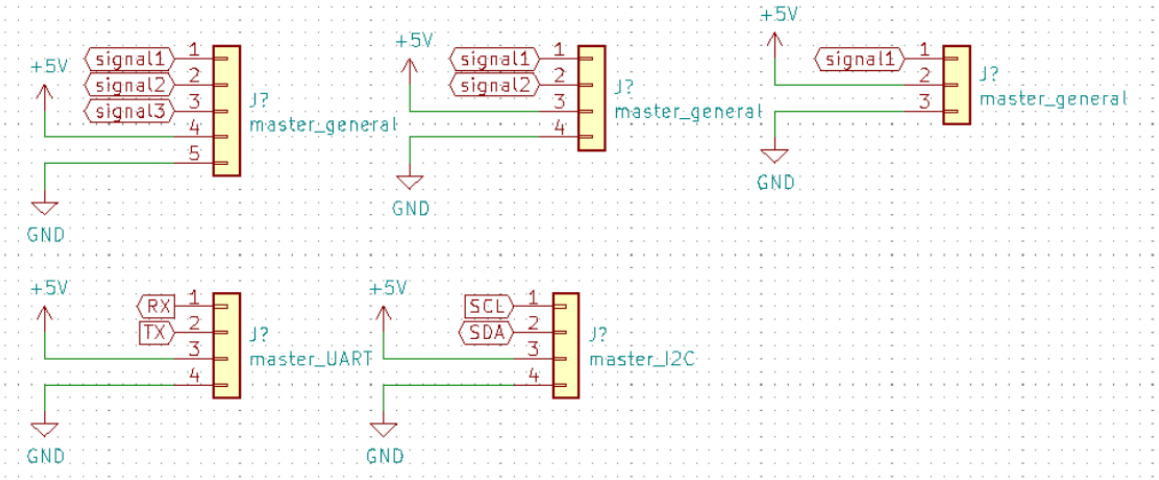
- 終端抵抗はスイッチで切り替えられるようにする
- E-STOPは非常停止スイッチが押されているかをGPIOで伝える
- これで駆動電源の状態を取得し指令を送るのを止めるなどする



UART

- つなげる一方を上, 他方を下のようにする(上の画像)
- 他の汎用的なコネクタも同じ(下の画像)
- 参考用リンク(コネクタについてまとめた記事, 他のコネクタについても載ってるので1回読もう)
- https://github.com/TNCTRobocon/manuals_markdown/blob/main/circuit/roboA_rules/connector_rule.md



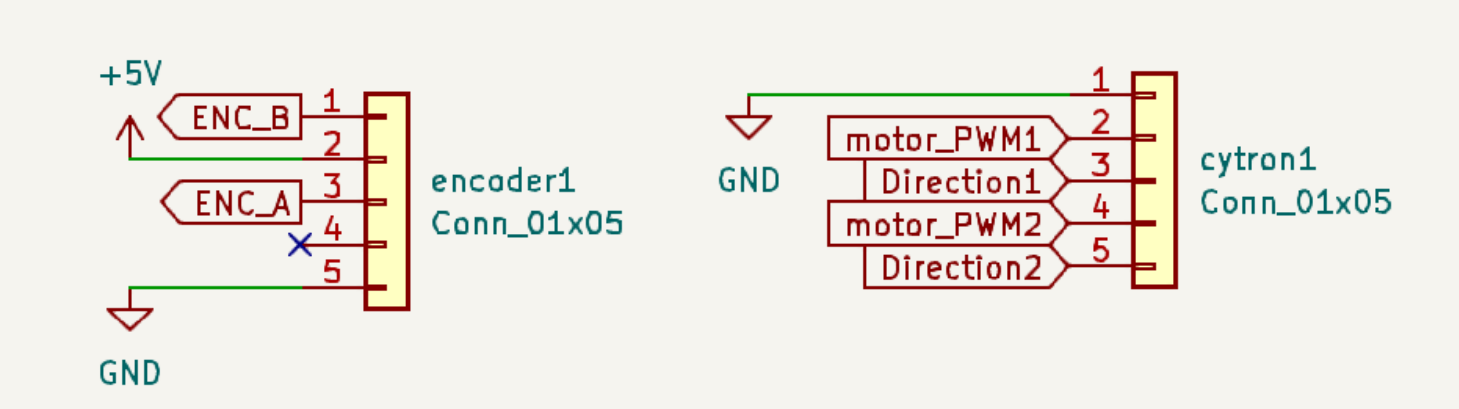


エンコーダー, cytron

- エンコーダーは5VなのでSTM32の耐圧ピンに接続する (ENC_A, ENC_B)
- 耐圧ピンはデータシートに“FT”と書かれている
- cytronはモータードライバーの一種

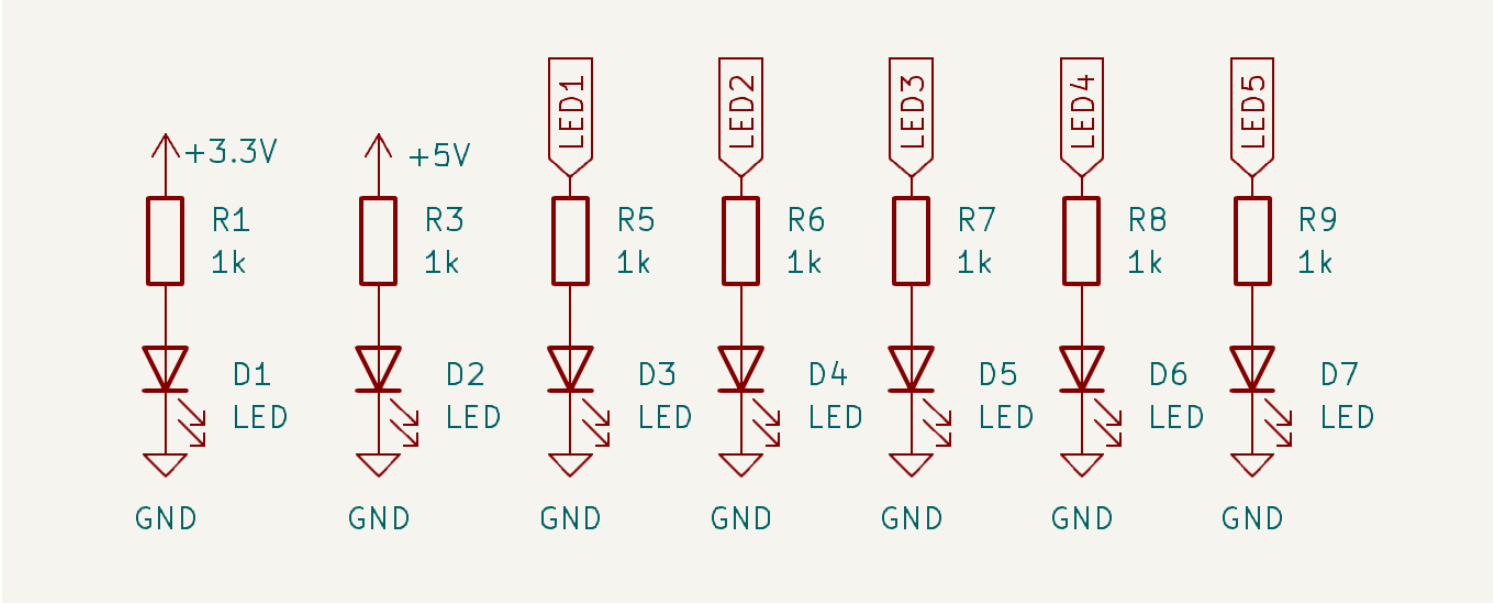
Table 10. STM32F446xx pin and ball descriptions (continued)

Pin number					Pin name (function after reset)	Pin type	I/O structure	Notes	Alternate functions	Additional functions
LQFP64	LQFP100	WLCSP 81	UFBGA144	LQFP144						
-	5	-	B4	5	PE6	I/O	FT	-	TRACED3, TIM9_CH2, SPI4_MOSI, SAI1_SD_A, FMC_A22, DCMI_D7, EVENTOUT	-
1	6	B9	C2	6	VBAT	S	-	-	-	-
2	7	C8	A1	7	PC13	I/O	FT	-	EVENTOUT	TAMP_1/WKUP1
3	8	C9	B1	8	PC14-OSC32_IN(PC14)	I/O	FT	-	EVENTOUT	OSC32_IN
4	9	D9	C1	9	PC15-OSC32_OUT(PC15)	I/O	FT	-	EVENTOUT	OSC32_OUT
-	-	-	C3	10	PF0	I/O	FT	-	I2C2_SDA, FMC_A0, EVENTOUT	-
-	-	-	C4	11	PF1	I/O	FT	-	I2C2_SCL, FMC_A1, EVENTOUT	-
-	-	-	D4	12	PF2	I/O	FT	-	I2C2_SMBA, FMC_A2, EVENTOUT	-
-	-	-	E2	13	PF3	I/O	FT	-	FMC_A3, EVENTOUT	ADC3_IN9
-	-	-	E3	14	PF4	I/O	FT	-	FMC_A4, EVENTOUT	ADC3_IN14
-	-	-	E4	15	PF5	I/O	FT	-	FMC_A5, EVENTOUT	ADC3_IN15
-	10	-	D2	16	VSS	S	-	-	-	-
-	11	-	D3	17	VDD	S	-	-	-	-



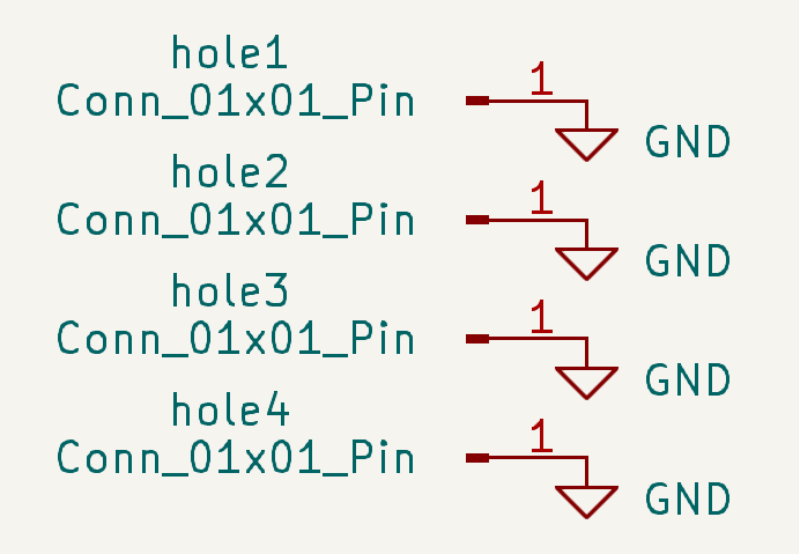
LEDたち

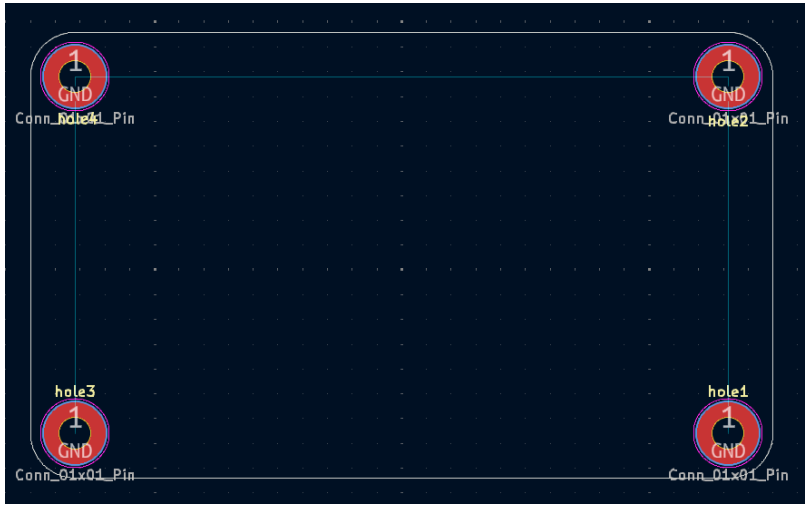
- 回路のどこが動いているのか視覚的に判断できるようにするためにつける
- 3.3V, 5V(電源基板なら+BATT, 出力にも)につける
- 例)5Vは光ってるけど3.3Vは光らない
  →レギュレーターが壊れてる(?)
- LED1~5は必要そうな個数(3個でいい)だけ用意する(マイコンにつなげる)
- プログラムがちゃんと動かない！といったときにデバッグ用に用意する
- 例)CAN通信を受信したら光らせるというような処理をプログラムに入れる



基板の穴

- 実際に基板に配置するときは右の図のようになります
- 穴の間隔は先に決めましょう
  (基板のサイズ, コネクタの配置とともに規格を作ると良い)





ロータリースイッチ

- スイッチの状態をマイコンで読み取ることでプログラムを変えずに設定を変えたりIDを変えたりできる