

123456

A

IN-OUT

File: inout.kicad_sch

B

PowerSupply

File: ps.kicad_sch

C

controller Sheet

File: controller.kicad_sch

D

MH1MH2MH3MH4

GND

Appmind

Sheet: /

File: RB2AMPController.kicad_sch

Title: Radioberry Amp Controller

Size: A4

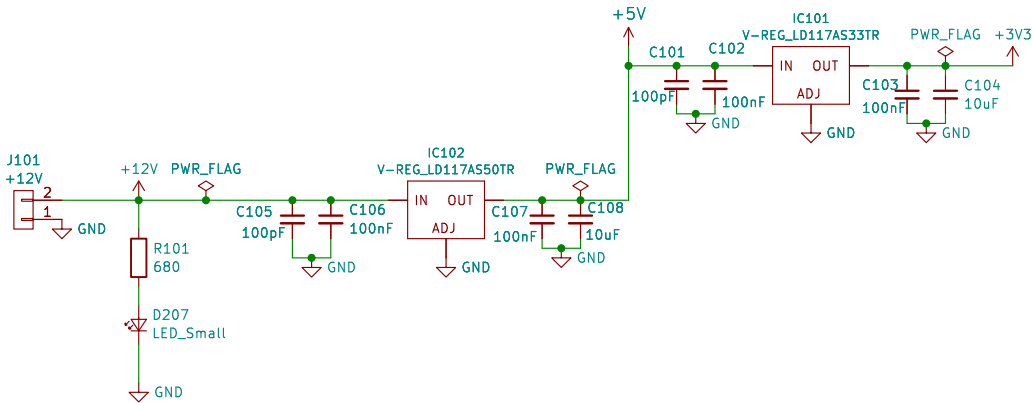
Date: 2023-10-10

Rev: beta 0.1

KiCad E.D.A. kicad 7.0.7

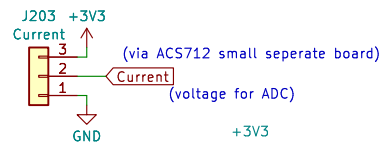
Id: 1/4

12V Power Supply



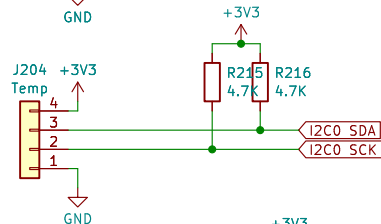
Sheet: /PowerSupply/ File: ps.kicad_sch		
Title:		
Size: A4	Date:	Rev:
KiCad E.D.A. kicad 7.0.7		Id: 2/4

Current

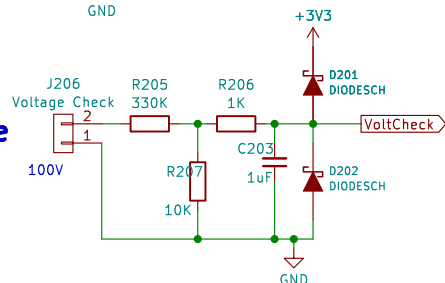


Temp.

i2c

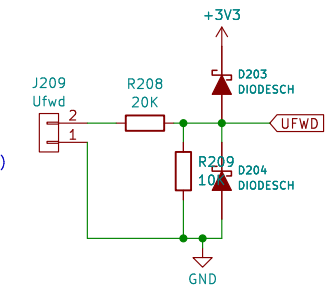


Voltage Check



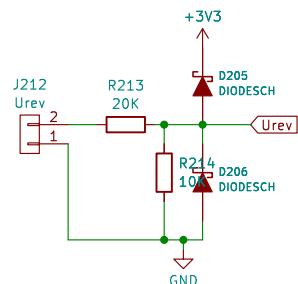
Ufwd

(From LPF board)

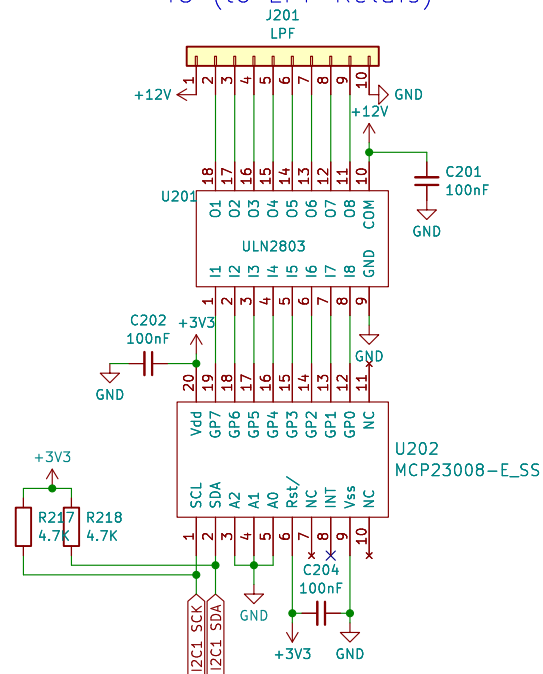


Urev

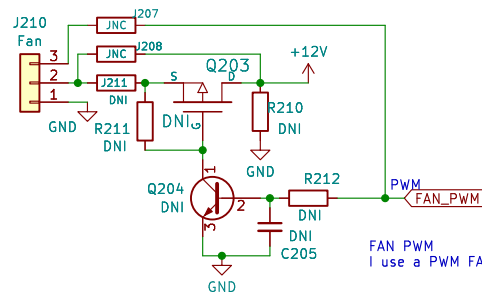
(From LPF board)



IO (to LPF Relais)

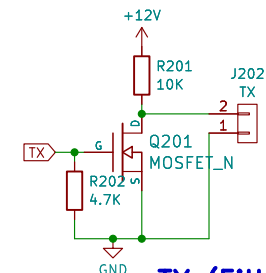


Fan

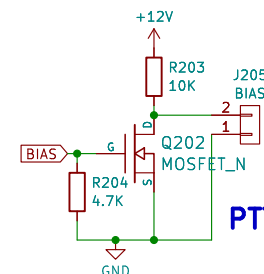


FAN PWM
I use a PWM FAN; so no need for the other components

TX (Filterboard)



PTT PA (BIAS)



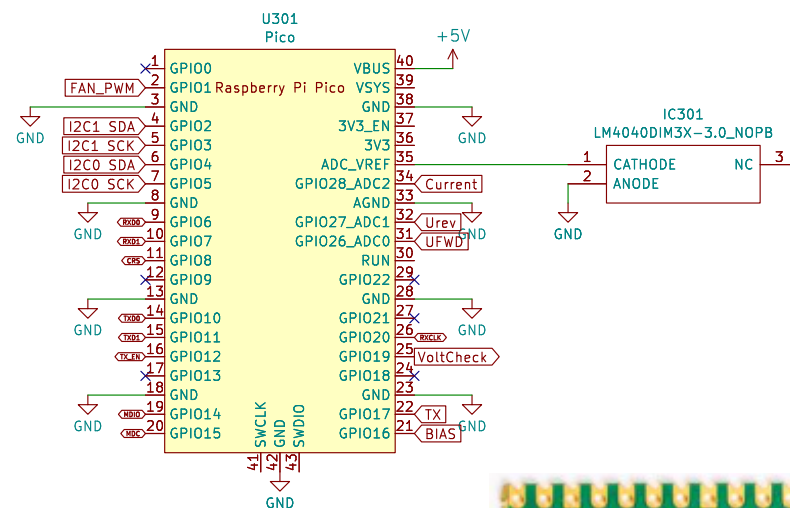
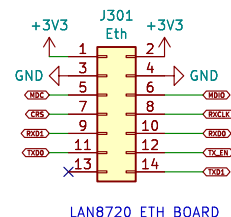
Sheet: /IN-OUT/
File: inout.kicad_sch

Title:

Size: A4
KiCad E.D.A. kicad 7.0.7

Date:

Rev:
Id: 3/4



Radioberry Amplifier Controller

Using a Raspberry PI 2040 Pico module. Adding an Ethernet Module with the lwip software library a tcp server is created.

The radioberry firmware implements a tcpip client and is able to exchange info.

The data from the SDR program controls the PA and the info from the PA is presented to the operator.

Sheet: /controller Sheet/
File: controller.kicad_sch

Title:

Size: A4
KiCad E.D.A. kicad 7.0.7

Date:

Rev:

Id: 4/4