

Teensy bat detector building instructions.

Cor Berrevoet's version of Frank DD4WH's bat detector idea, PCB design and some optimization by Edwin PE1PWF

Board version 0.2b 20-01-2020

Visual representation of the received sounds.

3 listening modes:

-(Auto) Heterodyne,

-Time Division

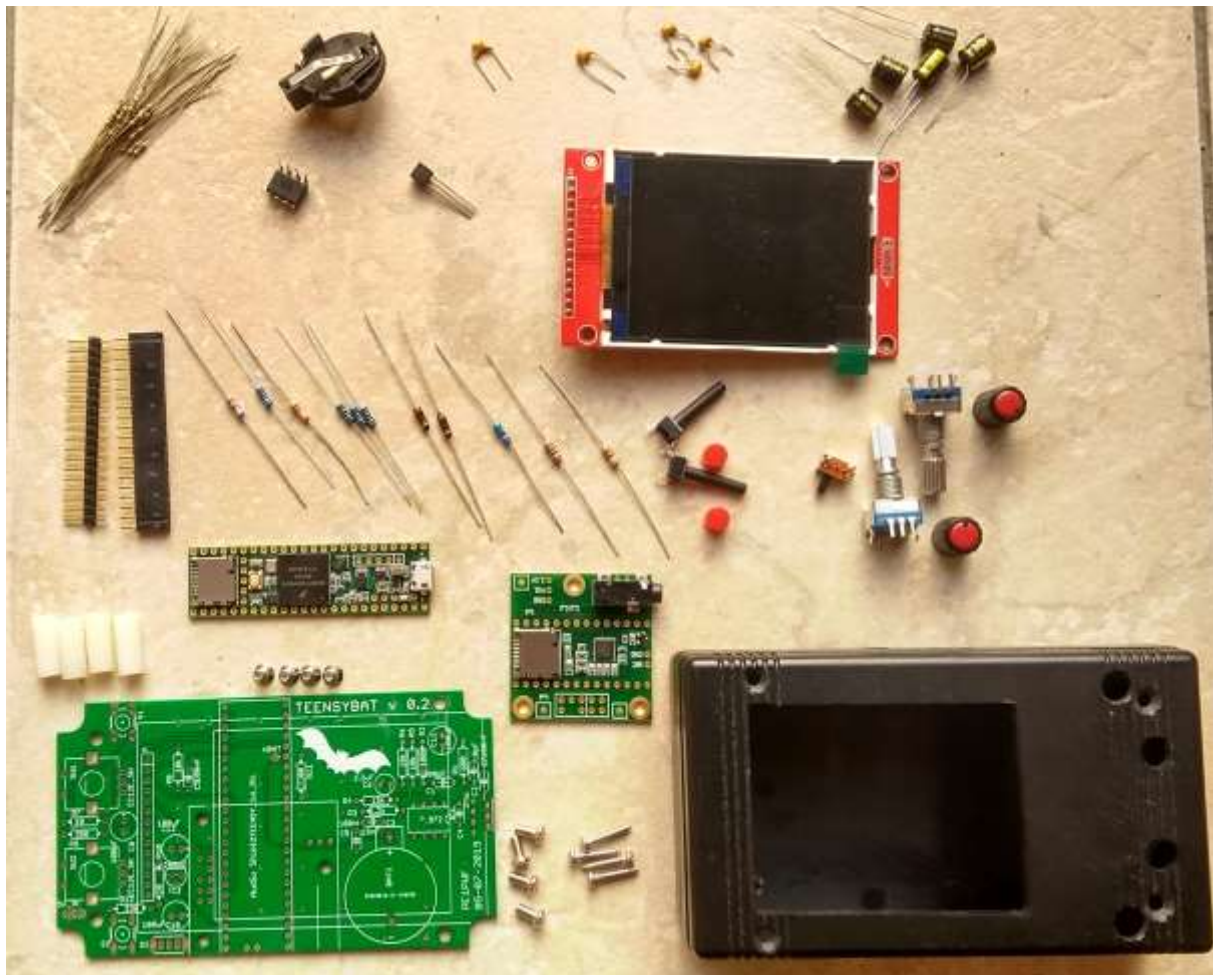
-Time Expansion

High recording sample rates, up to 384kHz

Check <https://forum.pjrc.com/threads/38988-Bat-detector> for more information on the project, code and discussion.



Edwin, 13-08-2020



Almost all parts needed. Not in this picture: CR2032 coin cell and MEMS microphone. In this V0.2 version board I use an TL962 IC instead of the transistor. (new boards have smd pads for is as well)

Before we start building be aware that two important noise reduction tricks were used. I suggest you do the same. The first thing is a separate 3.3Volts regulator, the output is used for the audio board and microphone and amplifier. Be sure you do not connect 3.3V of the teensy to the audioboard or mainboard. The writing of the SD card creates noise on the 3.3V circuit of the teensy. A separate 3.3V regulator is now mounted on the V0.2 board.

The other trick is that we use 150 or 100 ohm resistors in every data line coming from the Teensy we will use small 1/8Watt resistors instead of headers. If you insist one using headers I suggest you build a transistor amplifier instead of the opamp amplifier. The opamp easily picks up the data signals.

Parts list:

Part Value

5V_IN Header connector or wires soldered on pads
BAT1 Battery holder for CR2032 like Multicomp CH25-2032LF, ebay 4894462099247 + CR2032
C1 4.7nF 5.08mm pitch
C2, C4, C7 not used
C3 15nF 5.08mm pitch
C5, C9 100nF 5.08mm pitch
C6, C8, C10, C11, C12 100uF, 10V low esr electrolytic capacitor 2.54 mm pitch
IC1 TL972
IC2 LP2950Z 100 mA, Low Power Low Dropout Voltage Regulator
J1 2.8" 240x320 SPI TFT LCD ILI9341 5V/3.3V (red pcb, 14+4pin) (eBay usually has bad packaging)
R1 not used
R2 270 ohm for ICS / 390 ohm for SPU microphone choose best value to match mic impedance
R3 1k
R4 18k
R5, R6, R8 10k
R7, R10 4.7r
R9 4.7k / 47r Display backlight, unfortunately diplays differ a lot. (4.7k for newer displays)
R11 47r
R12 150k / 560r Display backlight for night mode. (150k for newer display)
S1, S2 6x6 tactile switch 18-20mm high plus small button (6x6mm)
S3 5mm Horizontal Slide Switch SPDT 1P2T 3 Terminals PCB Panel Latching (optional day/night)
SW1, SW2 Rotary encoder with switch EC11 EC12 with 20mm shaft (plus small knob 9.8*12mm)
U1 Teensy 3.6 module WITHOUT HEADERS!
U2 Teensy 3 Audioshield
U3 MEMSMICrophone like SPU0410LR5H-QB or ICS-40730 (like ELV "Mikrofon MEMS1 kit 68-15 14 56")
VBAT Extra header pin (total header pins 1x 16*1 male 1x 20*1 female should be enough)

Additional parts

26x 1/8 watt 150 ohms resistors (used to connect teensy data lines)
Project Box 116x68x36mm Black, search this size on eBay, AliExpress, Amazon
(5 ribs on each end of the top of the box) an alternative could be RND 455-00087
4x 15mm high M3 standoff (brass, plastic your choice), with a touch display 16mm is better. (or remove the touch panel)
8x M3 screws l=8mm (or 4 screws an 4 nuts depending on the used standoffs)
1x small flat power bank, you can put this into the box, or use it externally.

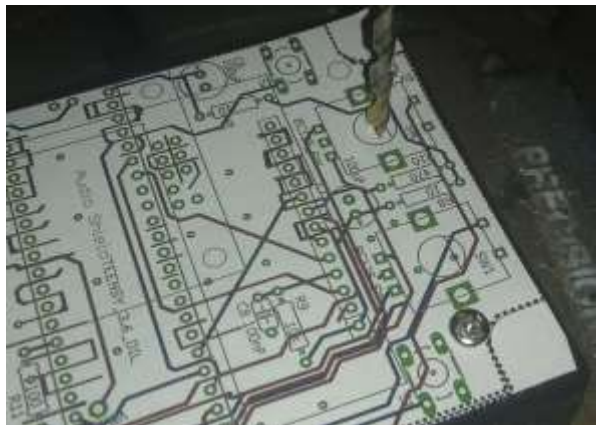
Power supply: for power supply I use a power bank that I built into the enclosure. Please note you want the PCB from the power bank as far as possible from the microphone. (to fit my power bank I needed to take it apart)

Microphone: If you mount the microphone in the enclosure, be sure to make a good seal to the wall of the enclosure. Another option is to allow the microphone to be outside of the enclosure. Prevent creating resonance chambers around the microphone. Best results however are with a microphone inside the enclosure with a good seal to the wall.

Before mounting parts we could use the PCB as template for drilling holes to fix the PCB, first we need to remove all the low supports from the box by cutting or drilling.



After cutting away the supports we do not need, we can use the PCB to determine the four mounting holes.



After drilling the four holes we use the PCB image as template for drilling the other holes. For the encoders I used 7mm and for the tactile switches 5.5mm the display window is slightly left of the centre!

The holes for the headphones, micro USB and SD-bard should be measured after building since these differ a little by mounting distance.

On the last page of this document you will also see a drawing of the dimensions and placed of the holes to be cut out.

You need to buy the Teensy 3.6 without the headers and mount these “header” pins yourself. All 26 data pins should be 150 or 100 ohm resistor for noise suppression.

On youtube you can find a video showing how to solder/assemble the boards. I suggest you watch it or at least some pars of it. Mounting the “header resistors” is a bit tricky.

<https://www.youtube.com/watch?v=mgacnVbqvVM>

“Teensybat build v0.2 pcb”

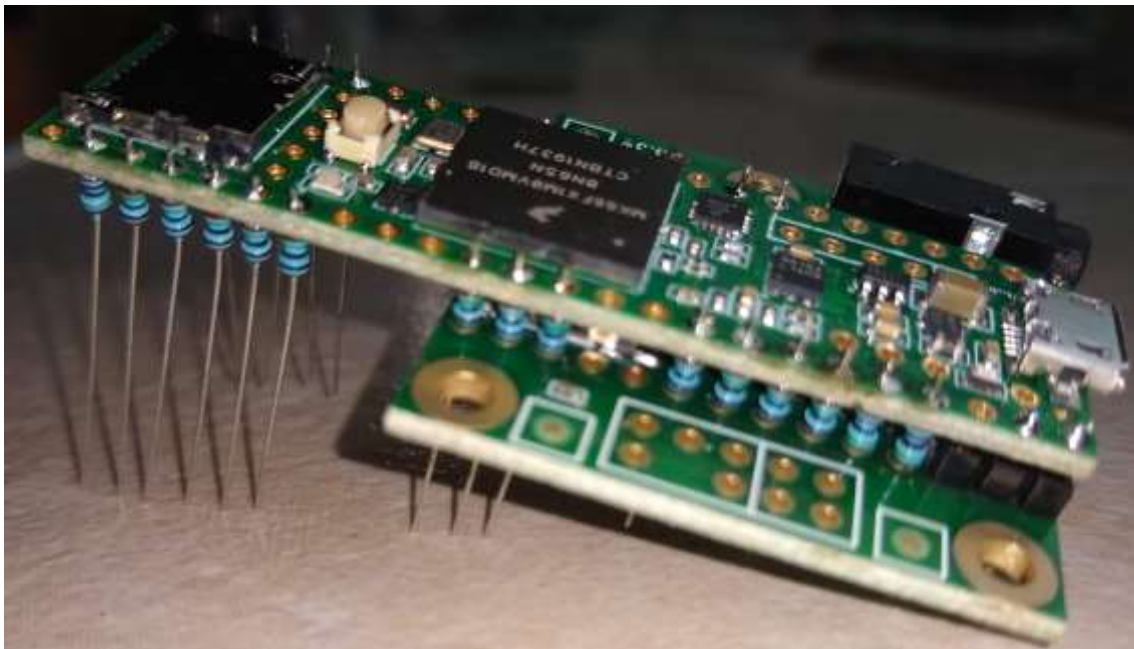
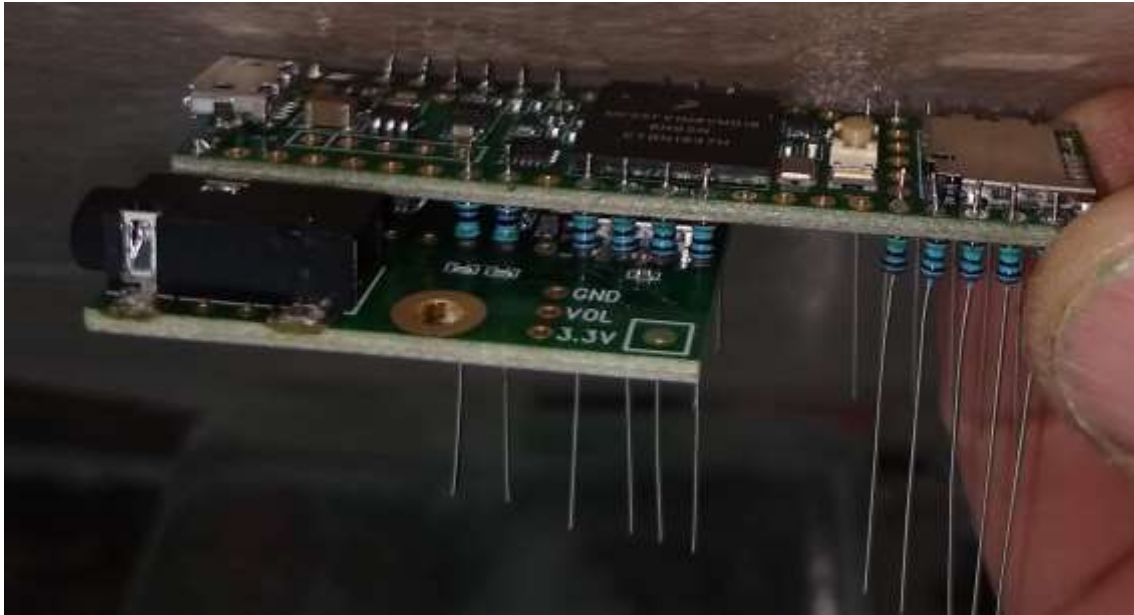
First we start soldering the 26 "header resistors to the teensy:

Clip off one wire on the resistors to about 4mm from the body, I put them in the board and solder them one-by-one. After the resistor is soldered I heat up the solder again pushing the resistor against the PCB making sure it is nice and straight.

I use the diagram on the next page to determine the position of a resistor.

After all these small 1/8 watt resistors are placed cut off the other ends of the wires so that the length is uneven. This makes putting the boards together much easier.

After cutting the wires, I place the remaining header pins in the audioboard and the mount the teensy on the audioboard like in the picture below.



Be sure you have the headers and resistors properly soldered to the audioboard before you proceed. This picture shows you where to put the resistor and where to put the header pins. Please keep in mind the MIC and 3.3V – MEMS MIC pins only connect the audioboard to the mainboard and not to the teensy.

Since the teensy comes quite close to the display board I do rip off the SD card connector, we do not need it. You can see that in the pictures below. Please notice the Q1 component on the newer boards.

As it turns out these have another kind of backlight control.

R9 and R12 on the Teensybat board are used for backlight. If you have no Q1 on the display board R9=47 Ohm, and R12=560 Ohm. If your display has Q1, please use R9=4k7 and R12=150k

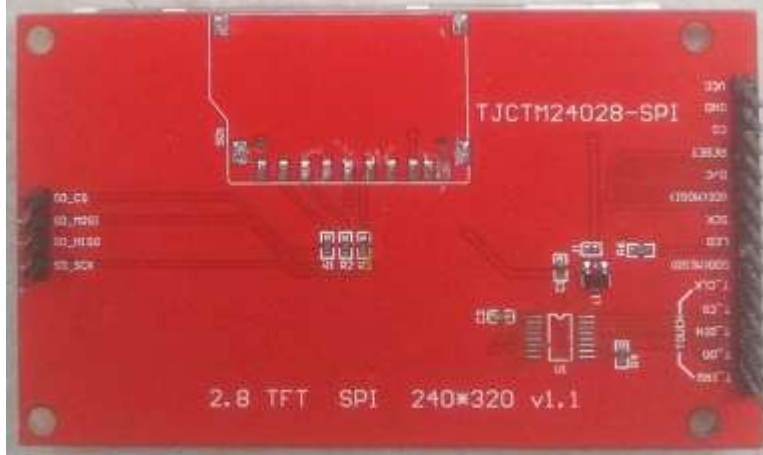


All these 3 boards are sold as the same model.

This one I bought a few weeks ago. You can see Q1 is on this board.



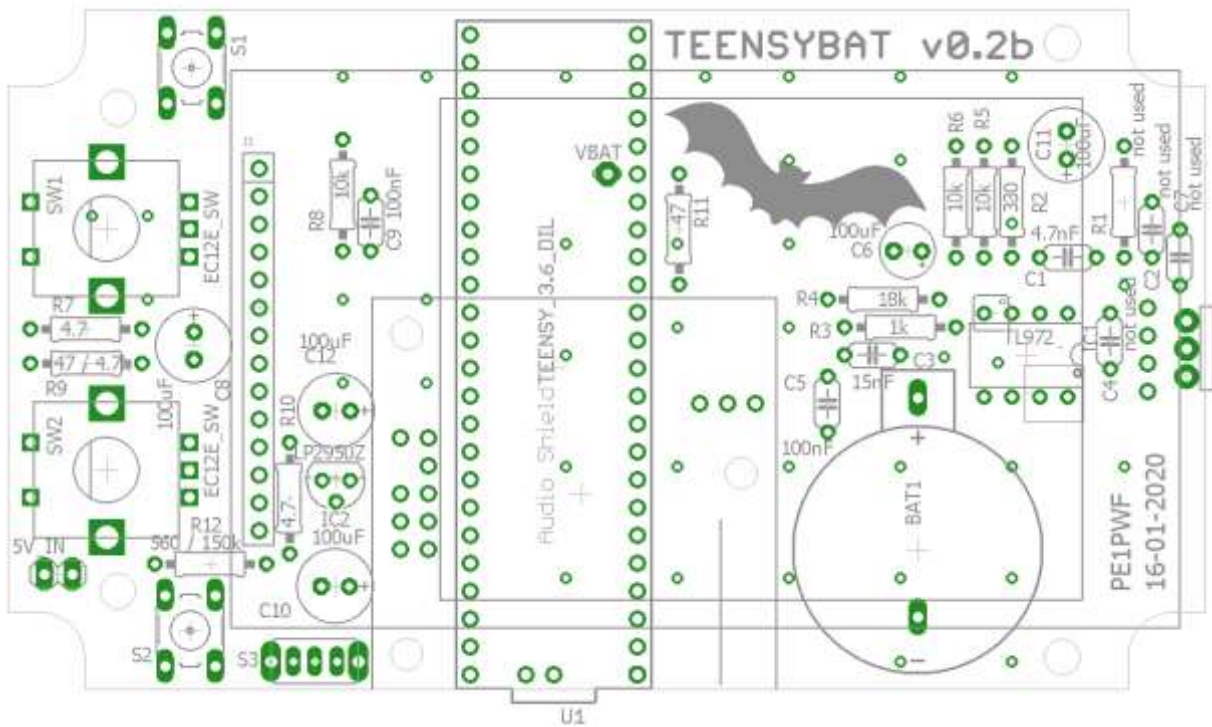
This is my latest purchase, the silkscreen suddenly shows CE marking and the layout of the tracks is a little different. Q1 is also on this board.



This is one of the older boards I have. The backlight is connected directly to the LED pin via a 5.6 Ohm resistor.

Use 47 Ohm for daylight conditions and 560 Ohm for night time in series.

This is the board and it's component layout



All parts are mounted on the top side of the board. The display is mounted on header sockets so there is some space to mount parts under the display. And we can remove it if we need to replace the clock-battery. Board revisions 0.2a and 0.2b can hold SMD versions of the op-amp IC.

It is important to have a good seal on the microphone. The microphone PCB needs to touch the wall of the housing. I normally add a piece of sealing foam.



The hole in the enclosure should ideally have 0 mm all thickness. For this I uses a lager drill to make a countersink hole (118 degree cone) after drilling a 2mm microphone port. This 0 mm wall thickness and good seal between the housing and microphone ensure the best audio quality.



Installing the software.

Since there are people having problems compiling the code we will load the already compiled HEX file.

To install the software we need a compiler program like Arduino IDE. I know there are other options but Arduino IDE is the most common.

<https://www.arduino.cc/>

To be able to use teensy in Arduino we should also install Teensyduino

<https://www.pjrc.com/teensy/teensyduino.html>

The project files from Cor Berrevoets contain other updated files.

Download the zip here https://github.com/CorBer/teensy_batdetector

Or my google drive

<https://drive.google.com/drive/folders/1NRtWXN9gGVnbPbqapPHgUCFOQDgEV1q?usp=sharing>

We now only use the precompiled .hex files.

In windows we open: C:\Program Files (x86)\Arduino\hardware\tools\Teensy.exe



We select the right hex file under File "Open HEX File"

The program tells us to press the button on the teensy to load the program.

One can also run

C:\Program Files (x86)\Arduino\hardware\tools\Teensy_reboot.exe instead of pushing the button.

The Teensy reboots and loads the new software.

Testing.

With the display in spectrum mode you should see a whistle as a strong peak on the left of the display. Rubbing your hands together near the microphone you should see a wider signal near about 20-30Khz.

Pointing the detector towards a switching power supply, like your phone charger or computer/laptop power supply or a TV should give some peaks near 30-60Khz.

You could also dangle a keyring with some keys in front of the microphone. These give a loud noise over just about the whole spectrum. If you dangle your keys close to the microphone you can set the gain to 0 and still have strong signals.

TE gives the best sound, of course you can also use HT or TD.

Using the bat detector.

The left push button standard switched the display modes.

The right push button selects different listening modes. FD, Auto HTD, Auto_TE, PASS, HTD

Both encoders can have similar functions like for setting frequency gain or volume.

The encoders have a push and a turn function. Push to enter a selection, turn to change the value.

For recording and playback you should use the left encoder.

To record, set the Button L function to record now the a push on the left button will start recording, another push will stop the recording.

To play, select Play, wait a few seconds, select the file you want to hear and press the left pushbutton. Pressing the left button again stops playing. During playing the right encoder can be used to set the playback rate.

For recording ultrasound, set the recording rate to a high value like 281 (default) or 352.

Just experiment a little, I do expect new features in the code so I will not try and write a complete user manual at this moment.

A full user manual is also available on my google drive.

The recordings are WAV or RAW format. You can use these in programs like Audacity.

Use file, Import. Raw data to import the recorded file. Or simply open the .WAV file.

For RAW select:

Signed 16bit PCM

Normal byte order

1 channel (mono)

And be sure you set the right Sample rate like 281000 or 352000 Hz like what you used when recording.

You can't play and listen this file right away. You have to slow it down, but it is wise to save this file first as a wav file so we can use it in other programs. Use file export WAV.

Audacity allows us to view the file as spectrogram, standard it is waveform, but spectrogram allows us to visualize the bat calls. This enables us to determine frequency, call length and time in between the calls. This information can help us to determine the species.

Click the downward pointing arrow ∨, left of the waveform.

Select spectrogram and select the same button again to set spectrogram settings. Change the maximum frequency to at least 100000 Hz.

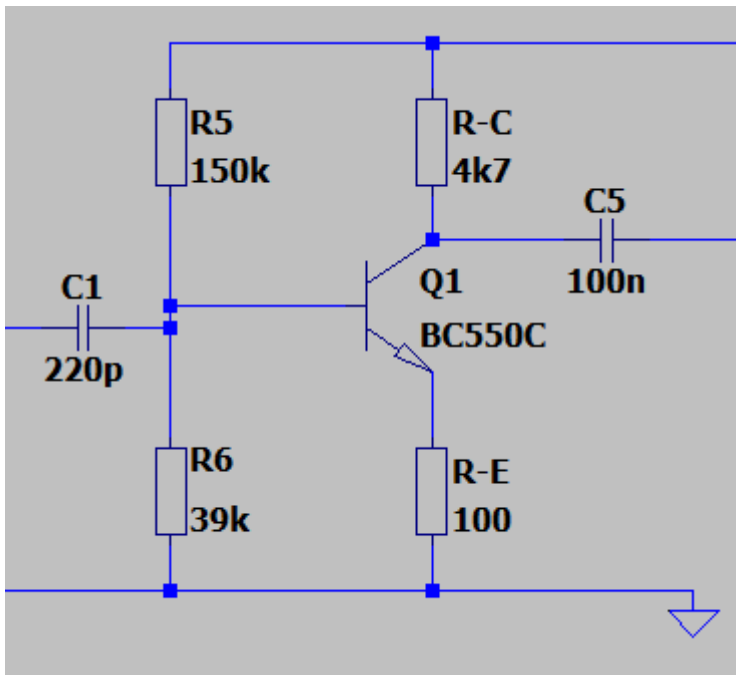
In this same menu you can also find "ratio" of you set the ratio to something like 44100, the playback will be slowed down and you can listen to the bat calls because the frequency is slowed to something we can hear.

Another nice tool to try is Bat Explorer. Use the 1.11 version since that one is free.

https://downloads.batlogger.com/be/BatExplorer_v1.11.4.0.exe

By importing the wav file, we saved in audacity we can use this tool to determine the time between, length and frequency of the calls to determine the species.

For the ones that are looking for a transistor amplifier, try this.....



C1 is a small value of only 220pF, this will reduce the amplification of low frequency sounds.

Since the opamp tends to pick up datanoise and the transistor is not that sensitive, it is best to use a transistor if you do not want to mount the Teensy on resistors.

So if you are using headers, use this transistor amplifier.



Mount it like this.

R5, R6 C1 and C5 are in the original positions.

R-C and R-E mount in holes of the opamp and C4, The transistor is also mounted in holes of the opamp.

Make sure you use a wire for R2.

Why do we want a microphone amplifier?

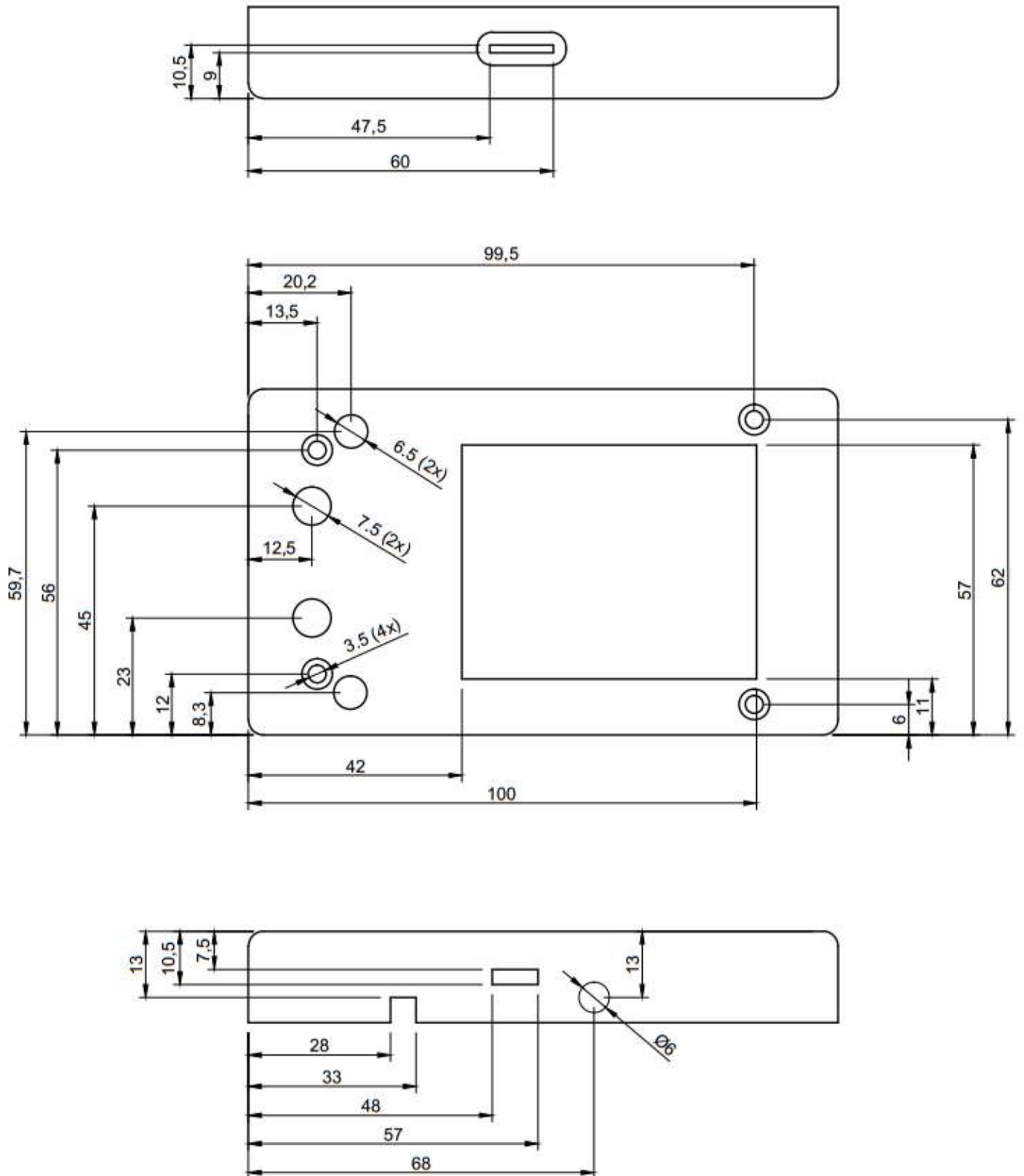
You could also connect the microphone directly to C5.

The amplifier however attenuates the low frequencies and amplifies the higher frequencies where the microphone seems to drop in sensitivity.

The Teensy audio board also seems noisy on high gain, the amplifier is much less noisy so that is why we prefer to do some amplification.

The Opamp is a bit more versatile and lower noise than the transistor amplifier. The transistor however picks up much less datanoise.

This picture shows just about the place to create the holes for knobs, display, audio and USB connector and a slot for the SD card. For mounting I like countersunk screws so that is why you see larger holes around the 3.5mm holes. If you also chamfer the edges around the SD cars slot, you will be able to take out the micro SD card quite easily.



! heights can depend on the length of the "header" resistors used !

Power bank.

I do like to build in a power bank. An “ultra slim” power bank like this one just fits in the enclosure.

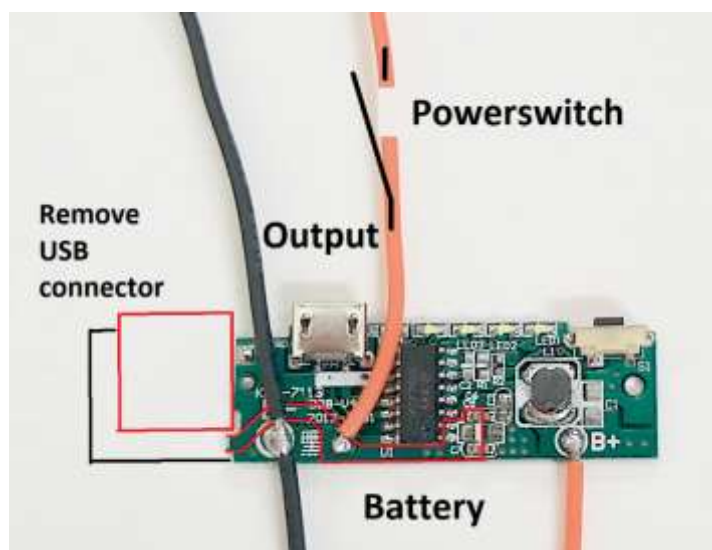


The are 4000 or 4400mAh power banks but you can see the being sold as 5000, 8000, 10000mAh power banks on Aliexpress, Banggood, Ebay or other online stores. You could also power the detector via the micro USB connector on the teensy.

I break off the output connector and glue the pcb and battery in the enclosure. I solder a wire to the track that was connected on the USB-A output connector

Via a small power switch I connect the output of the PCB to the +5v input of the detector.

I drill a few holes and small slot in the enclosure so I can charge the battery and see the charges state (4) LED's.

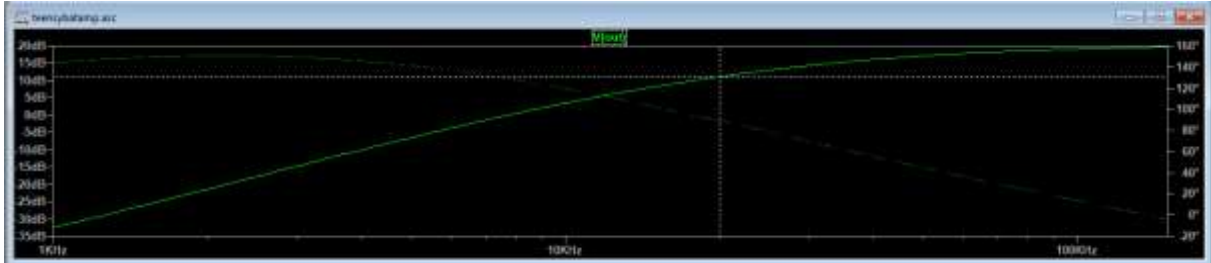


Modifications.

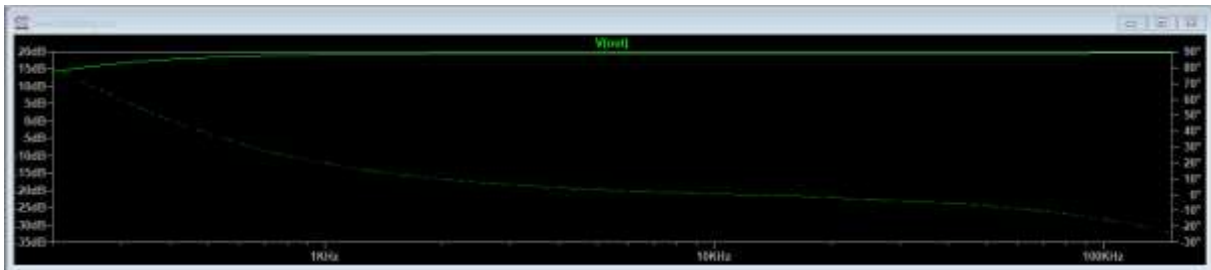
I can imagine you might want some other features which can easily be altered by choosing some other component values.

Microphone amplifier:

The microphone amplifier made to attenuate low frequency sounds and amplify higher ultrasound frequencies more than the lower ultrasound frequencies. This helps to keep voices out of the recordings and compensates for the lower sensitivity of the microphone on higher frequencies.



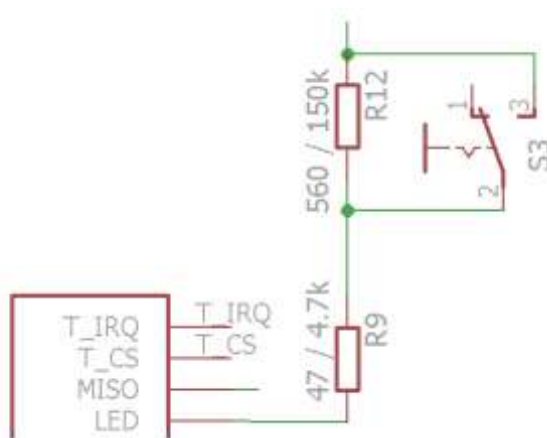
If you would like to record audible sounds, you only need to change both C1 and C3 to 1uF the microphone amplification will be close to 20db over the whole range from 0.1-200kHz

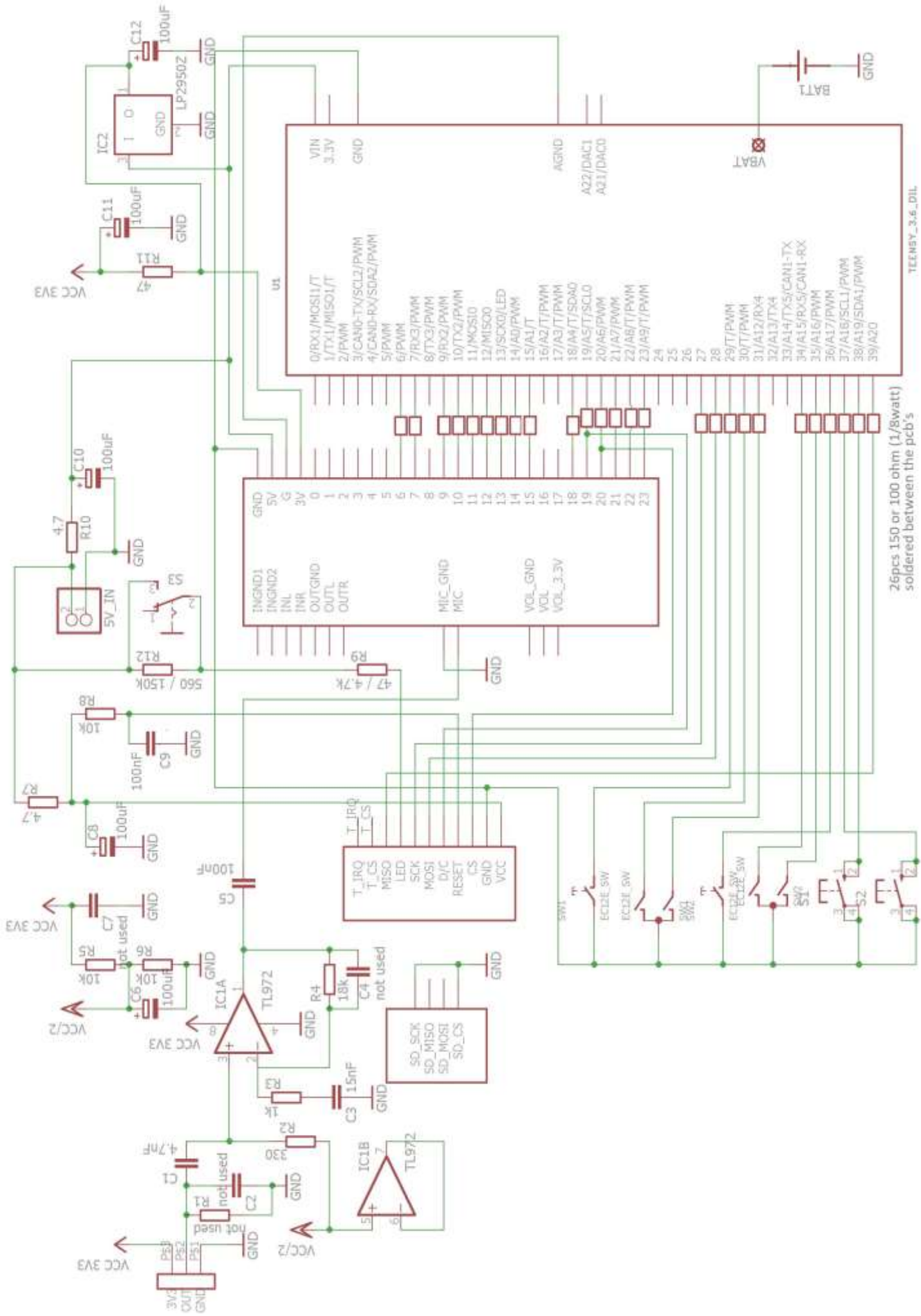


Another thing you might want to have is a switch to turn the display off.

If you want to use auto recording you probably want to save on battery power and do not want to attract anyone with the display backlight on. We can modify the display backlight circuit for that.

The dim-mode usually is 150k+4.7k in series. Bright is just 4.7 if you remove R12 you can change display to Bright or off. If you do not want bright and off, but dim and off, put 150K in the position of R9 and no resistor on the R15 position.





26pcs 150 or 100 ohm (1/8watt) soldered between the pcb's