



GETTING STARTED WITH FREEDSP-AURORA

DIY



REVISION HISTORY

Revision	Description	Date
v1.0	Initial Version	28 Nov 2018
v1.1	Hardware release	04 May 2019



Revision History	2
About freeDSP-aurora	4
Important Information	6
Features	7
Overviews	8
Block Diagram	8
Connectors	9
Optional Input and Output Add-Ons	11
How To Get freeDSP-aurora Up And Running	12
Get Everything Needed	12
Solder the Board	13
Programming the ESP32	14
Programming the XMOS XE216-512-TQ128	17
Upload a DSP Plugin	24
Appendix	25
Part List	25
Assembly Print	32
Schematics	33



ABOUT FREEDSP-AURORA

The freeDSP-aurora is a cost-effective real-time audio signal processing solution for researchers and the do-it-yourself community. It is a bare circuit board that can be incorporated into your own projects. It comes with no housing. Easy assembling and simple programmability are the main focus. It is based on Analog Devices' ADAU1452 DSP chip in bundle with the free graphical development environment SigmaStudio. The programming model is function-block based – comparable to other graphical programming languages like PureData or Max/MSP. Many prebuilt blocks (e.g., filters, compressors, effects, or logic) can be placed in the signal path via drag and drop. If the included libraries do not have the functions needed, low-level blocks, such as multipliers and delays, can be wired together to create custom algorithms. For more information please refer to the Analog Devices website.

FreeDSP-aurora offers a wide range of DSP processing options and interface controls with easy programmability. It can be used in various audio applications, e.g.:

- Room compensation / system equalization
- Digital crossovers in active loudspeaker concepts
- Multiband dynamics processing
- Delay compensation / phase shift
- Bass enhancement
- Subwoofer integration
- Advanced instrument audio effect units
- Stereo image widening
- ...

A XMOS XE216-512-TQ128 MCU is used to expose an USB Audio Class 2 compliant interface to a host computer running macOS, Linux or Windows 10. The boards provides 8 audio input and 8 audio output channels to the host computer. Additionally an ADAT input/output and a Wordclock input/output is provided by the XMOS MCU.



The ESP32 MCU controls the operation of the DSP. Furthermore, it provides WiFi and Bluetooth connectivity and handles peripherals like rotary encoder, display, temperature sensor, PWM controlled fan and IR sensor.

As part of the project an open source software is published that controls the operation of the freeDSP-aurora from macOS, Windows or iOS. Via the control software the user can access all parameters of the uploaded DSP plugin. Due to the open source licenses users can modify the control software for supporting their own DSP plugins.

The plans and software for the freeDSP-aurora board are published under a Creative Commons Attribution ShareAlike 4.0 International (CC BY-SA 4.0) license, which allows the unrestricted use and modification of the module. This means that experienced users can make their own version of the board, extending it and improving it, as long as they credit freeDSP and auverdion and release their designs under the same license.

The freeDSP brand and freeDSP logo are copyright of Sebastian Merchel and Ludwig Kormann and cannot be used without formal permission.

The auverdion brand is copyright of Raphael Knoop and cannot be used without formal permission.

This getting started guide is published under the same CC license.



IMPORTANT INFORMATION

The freeDSP-aurora board might generate signals that may damage your audio equipment. Please read and understand this manual before starting to work with your board. Adjust all hardware settings and configure your software before connecting any audio equipment to freeDSP-aurora. Always start with low volume on your amplifier and slowly increase the level to reduce the risk of damaging your audio system.

freeDSP-aurora is provided to you ‘as is’. We make no express or implied warranties whatsoever with respect to its functionality, operability, or use, including, without limitation, any implied warranties of merchantability, fitness for a particular purpose, or infringement. We expressly disclaim any liability whatsoever for any direct, indirect, consequential, incidental or special damages, including, without limitation, lost revenues, lost profits, losses resulting from business interruption or loss of data regardless of the form of action or legal theory under which the liability may be asserted, even if advised of the possibility or likelihood of such damages. Features and specifications might change without prior notice.

Please keep in mind that freeDSP-aurora is an open-source spare-time project. Because freeDSP-aurora is very flexible, many applications are possible. Questions and new ideas can be discussed online with other DIYers. Please use the *Digital Line Level* subforum @ diyAudio.com or the *Elektronik* subforum @ www.diy-hifi-forum.eu to connect with other people working with freeDSP-aurora. Please create individual threads for your topics only if you cannot find your issue in the existing threads. Some questions can be answered by carefully reading this manual. We cannot provide individual support via email. Thank you for your understanding!

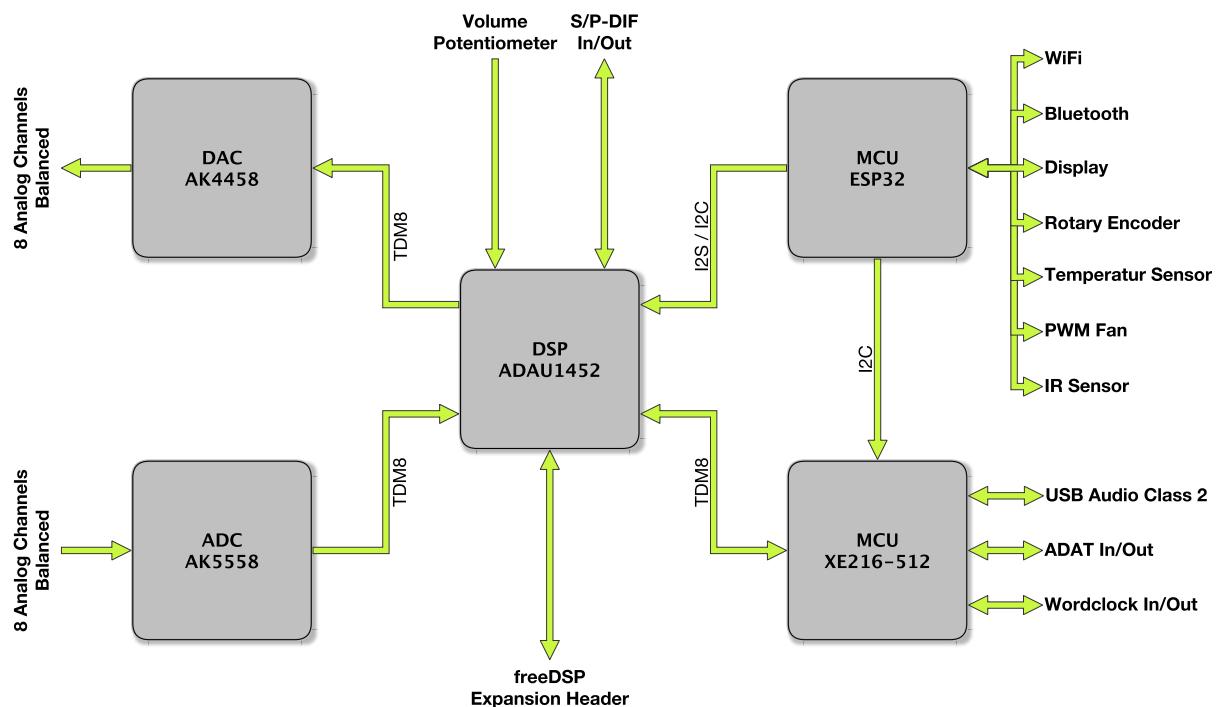


FEATURES

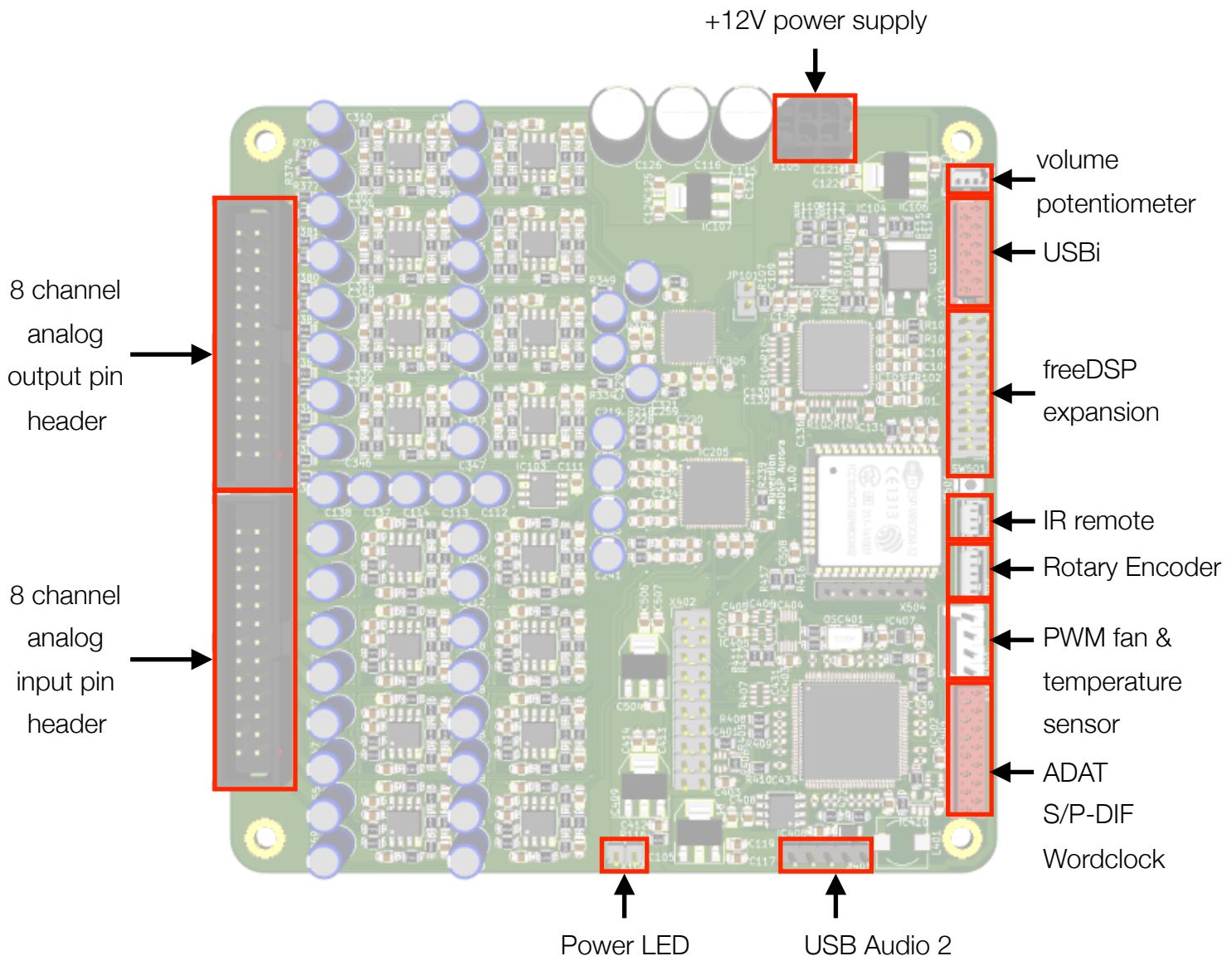
- Analog Devices ADAU1452, 294.912 MHz, 32-bit SigmaDSP
 - 6144 SIMD instructions per sample @ 48kHz fs
 - 40kWords of data RAM
 - 800ms digital audio delay pool @ 48kHz fs
 - 8 stereo ASRCs with 139dB DNR
- XMOS XE216-512-TQ128 for multichannel bidirectional audio streaming
- ESP32 for WiFi or Bluetooth control
- AKM AK4458 32bit-DAC
- AKM AK5558 32bit-ADC
- Supporting sample rates between 44.1kHz and 192kHz
- 8 analog balanced input channels, +6dBu
- 8 analog balanced output channels, +6dBu
- S/P-DIF input and output
- ADAT input and output
- Wordclock input and output
- Support for display, rotary encoder, volume potentiometer, temperature sensor, PWM controlled fan, IR sensor
- One freeDSP expansion header
- USB Audio Class 2 Bidirectional streaming with 8 channels in and 8 channels out, full-duplex. Works with ASIO driver under Windows 10 and driverless under macOS and Linux.
- Realtime control software for Windows, macOS, Linux, iOS connecting by WiFi available under an open source license
- Board dimensions: 110mm x 110mm

OVERVIEWS

BLOCK DIAGRAM



CONNECTORS





Analog Audio Connections

Analog audio input connections can be made on pin header X201. The audio inputs are designed for balanced operation at +6dBu maximum input level. Unbalanced sources can be connected as well by grounding the negative input. In the latter case you may have to change the gain of the input stage to improve the signal to noise ratio. Alternatively you may want to use a conversion circuit if your audio sources use a different connection (e.g. unbalanced) or have another level.

Analog audio output connections can be made on pin header X301. The audio outputs are designed for balanced operation at +6dBu maximum output level. In case of unbalanced sinks leave the negative output unconnected. In the latter case you may have to change the gain of the output stage to improve the signal to noise ratio. Alternatively you want to use a conversion circuit if your audio sinks use a different connection (e.g. unbalanced) or have another level.

When making audio connections, make sure that your equipment is powered off to avoid damage.

The gain of the analog audio input and output stage can only be changed by changing the resistors on the board to another value.

Digital Audio Connections

On connector X101 you can make your digital audio connections like S/P-DIF input and output and ADAT input and output. Wordclock input and output can be connected to X101, too.

FreeDSP Expansion Header

X102 is the an expansion header for additional input and output boards. The pinout complies with the I2S expansion header specification of the freeDSP project. If you want to connect a I2C display please connect it to this header, too.

Fan and Temperature Sensor

On connector X501 you can connect a PWM controlled fan. Please use the Sense pin to connect a temperature sensor (e.g. NTC).

Rotary Encoder

On connector X502 you can connect a rotary encoder with or without a push button.



IR Sensor

On connector X503 you can connect an infra red receiving diode.

USB Connection

Your host computer connects on the pin header X401. Please use a common off-the-shelf cable assemblies (typically used in computer hardware). Always confirm the pin-out with the manufacturer, or you could easily cause damage to your computer or freeDSP-aurora. Usually the GND-pins 4 and 5 can be identified by a black wire, but you can never be sure unless you checked the manufacturer's specifications of the connector.

FreeDSP-aurora was designed to be class compliant with UAC2. Thus, on macOS and Linux you don't need to install any driver. Windows 10 comes now with a UAC2 driver as well but you may have to install additional stuff like the free software asio4all and your audio software needs to support ASIO. Please note, that ASIO is only needed if you want to use the 8 input channels for recording audio. If you just want to use freeDSP-aurora to stream audio data to your audio equipment, ASIO is not needed and you can skip the asio4all installation.

Power LED

On this connector you can connect a LED to show the power on/off status.

Power Supply

freeDSP-aurora needs a power supply of +12V on X105. Attention: Apply power to the board only after all connections have been made and you double-checked everything.

OPTIONAL INPUT AND OUTPUT ADD-ONS

FreeDSP-aurora was designed to support as many applications as possible. Therefore, all inputs and outputs are on pin headers or ribbon cable connectors. This way user can adapt the front-ends to their needs. Some add-on boards will be available for this project. These boards will cover the most common used input output configurations, e.g. active multi-way loudspeaker or an 8 channel loudspeaker management or buttons and displays for user interaction. Please check the github repository and/or the website for information about the add-on boards.



HOW TO GET FREEDSP-AURORA UP AND RUNNING

The steps in this chapter are only required if you want to build your freeDSP-aurora from scratch or if you want to initially program or update firmwares. If you just want to upload another DSP schematic you can directly jump to chapter **Upload a DSP schematic**.

GET EVERYTHING NEEDED

You will need a soldering iron with a fine tip plus some soldering experience to assemble the surface mounted and through hole components.

Additionally you need:

- an USB2Serial converter like a FTx232 module, e.g. https://www.ftdichip.com/Products/Modules/DevelopmentModules.htm#FT2232H_Mini or similar
- optionally a XTAG debug adapter <https://www.xmos.com/support/boards?product=19480> if you want to use the features provided by the XMOS device
- optionally an USBi adapter from Analog Devices (EVAL-ADUSB2EBZ) or the freeUSBi programmer (see the freeDSP website) for programming the DSP if you want to program the DSP through SigmaStudio

Order the freeDSP-aurora kit. Sometimes centralized buying of PCB and all parts is offered on the forums. Please keep in mind that freeDSP-aurora is a spare-time project. It may take some time until a new batch of boards and/or kits is offered.

- (a) Alternatively, order the PCB via www.tindie.com. You will also need to order all electronic parts. The components can be ordered via DigiKey (or other distributors). This might be the fastest option.
- (b) Alternatively, you can manufacture your own circuit board or modify and extend the original design. This might be the most flexible option, but needs more expertise (and money) to get up and running.
- (c) Alternatively, manufacture the printed circuit board and order all parts yourself. You might want to locally organize centralized buying and board production together with some friends. You can find the necessary KiCAD files of the board on the freeDSP website



www.freeDSP.cc. You will also need to order all electronic parts. The components can be ordered via DigiKey (or other distributors).

SOLDER THE BOARD

You should start with soldering the most difficult part: The ICs. On youtube you can find tutorials how to solder the QFPs packages, e.g.

<https://www.youtube.com/watch?v=YUryJOAiPa4>

and how to solder the QFNs packages, e.g.

<https://www.youtube.com/watch?v=BvhE16vBfX4>

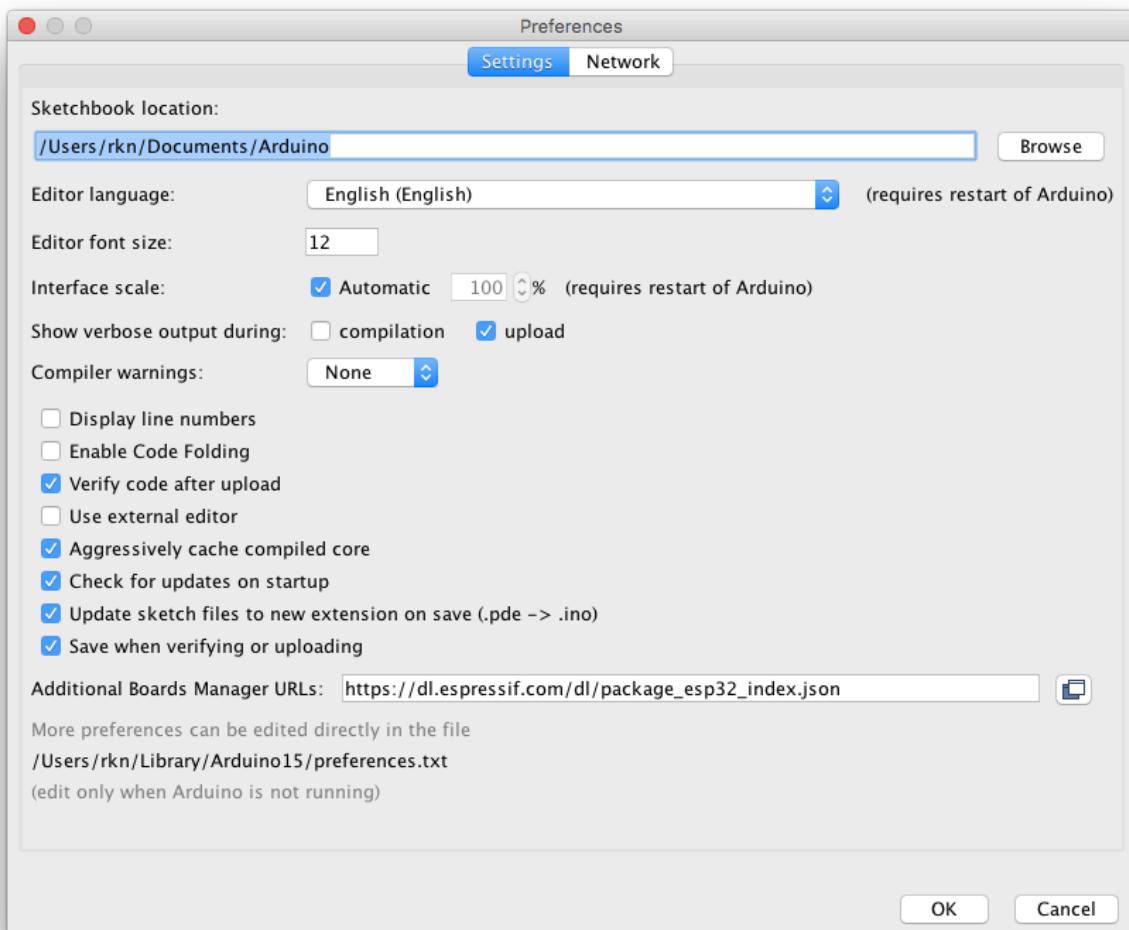
Then continue with all the other SMDs and finally solder the THT components. Always start with the components with lowest height.

PROGRAMMING THE ESP32

The ESP32 MCU on the board is needed to make parameter changes at runtime, save them and to upload new schematics to the DSP.

To program the ESP32 with the latest freeDSP-aurora firmware you have to do few steps:

1. Download and install the Arduino IDE from <https://www.arduino.cc> without charge.
2. Start the Arduino IDE and open the Preferences dialog and insert https://dl.espressif.com/dl/package_esp32_index.json for the *Additional Boards Manager URLs* and click OK. Then close the Arduino IDE.





3. Install the ESP32 toolchain by following the install instructions for your operating system described here:
<https://github.com/espressif/arduino-esp32#installation-instructions>
4. After download was completed and you restarted Arduino IDE you should see several ESP32-boards under *Tools->Board*. Please select *ESP Dev Module*.
5. Now connect your USB2Serial converter to X504: Connect TXD (pin 3), RXD (pin 4) and GND (pin 6) with the corresponding pins on your USB2Serial converter.
6. Please double check the pinout and the correct wiring. Then connect your USB2Serial converter to a USB port on your computer.
7. Select your USB2Serial converter under *Tools->Port* and open the Serial Monitor of Arduino IDE.
8. If you now power up the freeDSP-aurora board you should see the bootlog of the ESP32. If not, please double check your wiring. Perhaps RXD and TXD need to be swapped.
9. You are now ready to program the latest firmware on your ESP32.
10. Connect pin 2 with pin 1 of X504. If you now press and release button SW501 you should see a message in the Serial Monitor that the ESP32 is now waiting for a program download.
11. Open the file aurora.ino from the freeDSP-aurora repository.
12. Click *Sketch->Click/Verify*. The Arduino IDE should now build the firmware. After success click *Sketch->Upload* to upload the firmware to the ESP32 on your freeDSP-aurora board.
Please note: If the upload fails very often you did not set the ESP32 into programming mode as described in step 10.
13. After a successful programming of the firmware and a power off/on cycle of the freeDSP-aurora you should now be able to see your board in the list of available devices when you open the Wifi control panel of your computer.
14. At any time you can make changes to the ESP32 firmware and upload it to the freeDSP-aurora board by repeating steps 4 to 13.

After your first firmware upload you can also use the Over-The-Air (OTA) firmware upload to program the ESP32 over Wifi without having a USB2Serial converter connected. Please follow the instruction in the following link:



<https://www.az-delivery.de/blogs/azdelivery-blog-fur-arduino-und-raspberry-pi/ota-over-the-air-esp-programmieren-uber-wlan-entwurf?ls=en>

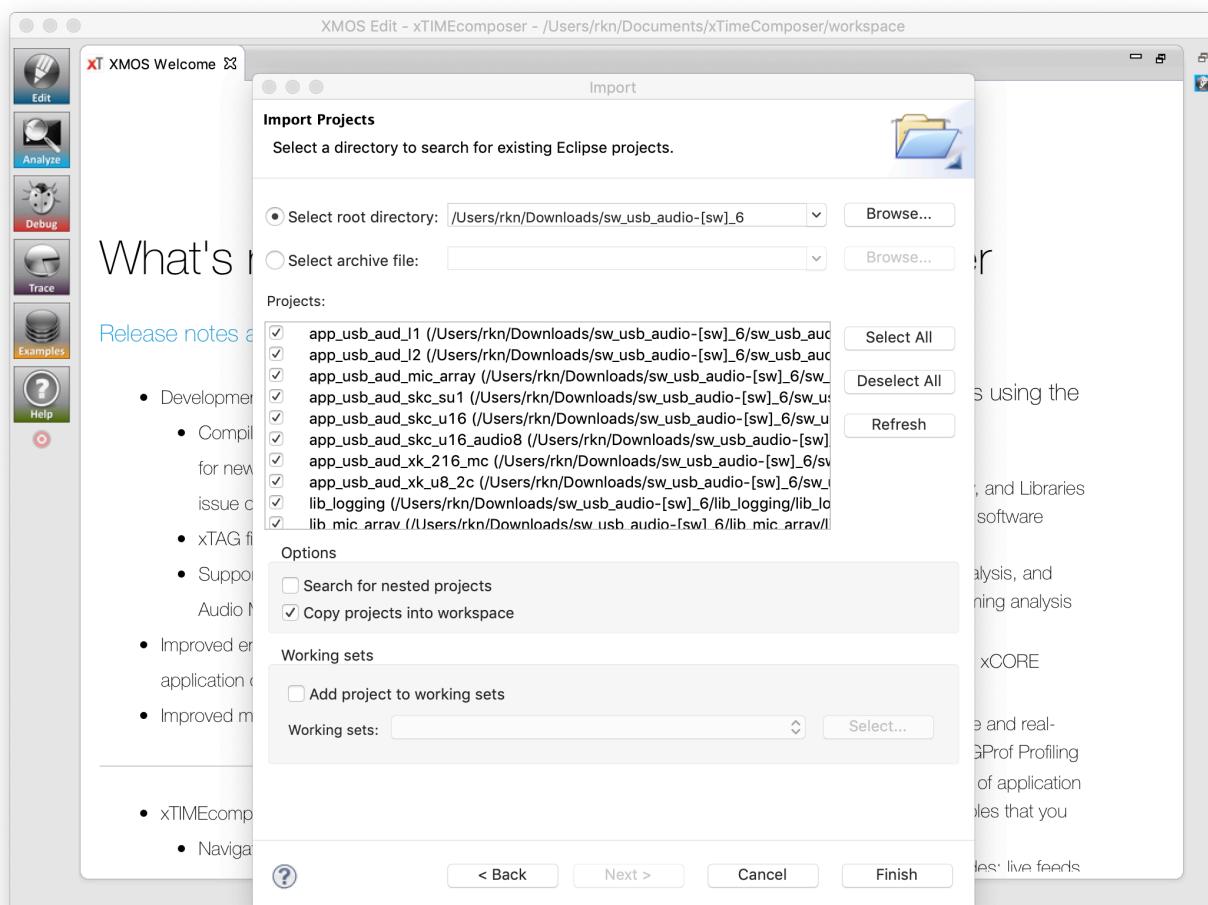
The OTA update method can only be used after the ESP32 has successfully connected to a local Wifi and your computer is connected to the same network. It does not work stable when you connect to the ESP32 by an adhoc connection.

If you want to use a third party firmware please make sure that the firmware supports the OTA firmware upload.

PROGRAMMING THE XMOS XE216-512-TQ128

If you want to use the features ADAT In/Out or Wordclock In/Out or USB Audio Class 2, you need to program the XE216-512-TQ128 to get the features working:

1. Download xTimeComposer Studio from <https://www.xmos.com/support/tools> without charge.
2. Download the XMOS USB Audio 2.0 Device Software Version: 6.15.2rc1 source code from <http://www.xmos.com/published/usbaudiodevice-software?version=all> without charge and unpack the archive (if not done automatically).
3. To import the framework into your workspace open *File->Import General->Existing Project into workspace* and click *Next*.
4. Select the directory where you have unpacked the downloaded framework and check *Copy projects into workspace*.



5. Click *Finish*.

6. In xTIMEcomposer Studio copy project app_usb_aud_xk_216_mc and rename the copy to app_freeDSP-aurora.
7. Copy the files from <git repository>/SOURCES/XMOS/src/core to <workspace>/app_freeDSP-aurora/src/core. Replace existing files with the files from the git repository.
8. Open the file audiohw.xc in xTIMEcomposer Studio and change to function AudioHwConfig to this:

```

void AudioHwConfig(unsigned samFreq, unsigned mClk, chanend ? c_codec, unsigned dsdMode,
                    unsigned sampRes_DAC, unsigned sampRes_ADC)
{
    unsigned char data[1] = {0};

    /* Set master clock select appropriately */
#if defined(USE_FRACTIONAL_N)
    /* Configure external fractional-n clock multiplier for
300Hz -> mClkFreq */
    PllMult(mClk, PLL_SYNC_FREQ);
#endif
    /* Allow some time for mclk to lock and MCLK to stabilise -
this is important to avoid glitches at start of stream */
    {
        timer t;
        unsigned time;
        t :> time;
        t when timerafter(time+AUDIO_PLL_LOCK_DELAY) :> void;
    }

#if defined(USE_FRACTIONAL_N)
    while(1)
    {

```

```
    /* Read Unlock Indicator in PLL as sanity check... */
    CS2100_REGREAD(CS2100_DEVICE_CONTROL, data);
    if (!(data[0] & 0x80))
    {
        break;
    }
}

#else
    if (mClk == MCLK_441)
    {
        set_gpio(P_GPIO_MCLK_FSEL, 0);
    }
    else
    {
        set_gpio(P_GPIO_MCLK_FSEL, 1); //mClk = MCLK_48
    }

    /* Allow MCLK to settle */
    wait_us(20000);
#endif

return;
}
```

9. Open the file customdefines.h in xTIMEcomposer Studio and browse to the USB descriptor definitions and add/change the product and vendor string describing your product, e.g.

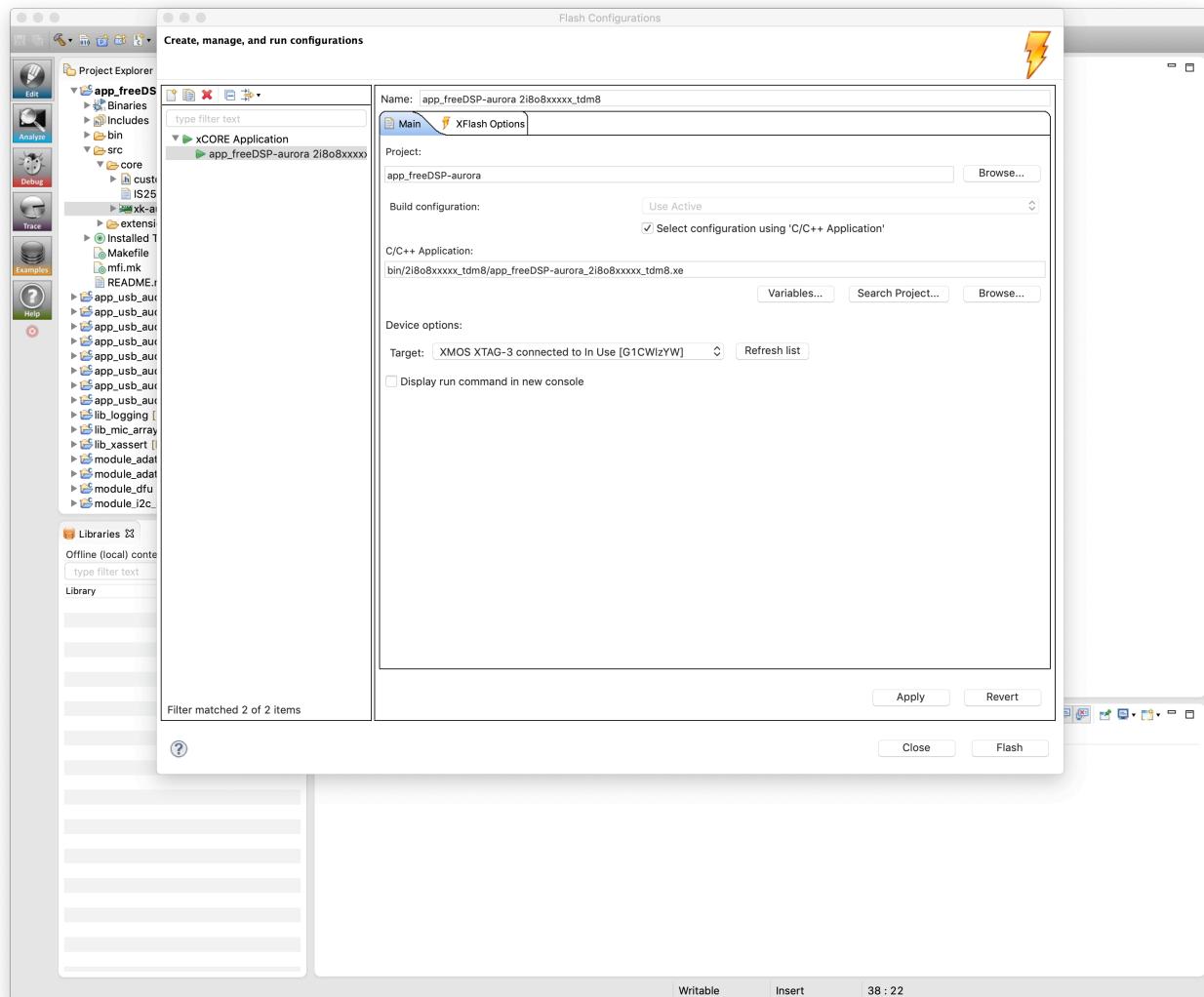
```
#define PRODUCT_STR      "freeDSP-aurora"
#define VENDOR_STR       "auverdion"
```

10. Change VENDOR_ID, PID_AUDIO_1 and PID_AUDIO_2 to your own VID and PID.
11. Open the file Makefile in xTIMEcomposer Studio search for the line starting with XCC_FLAGS_2i808xxxxx_tdm8¹ and make sure that it looks like this:

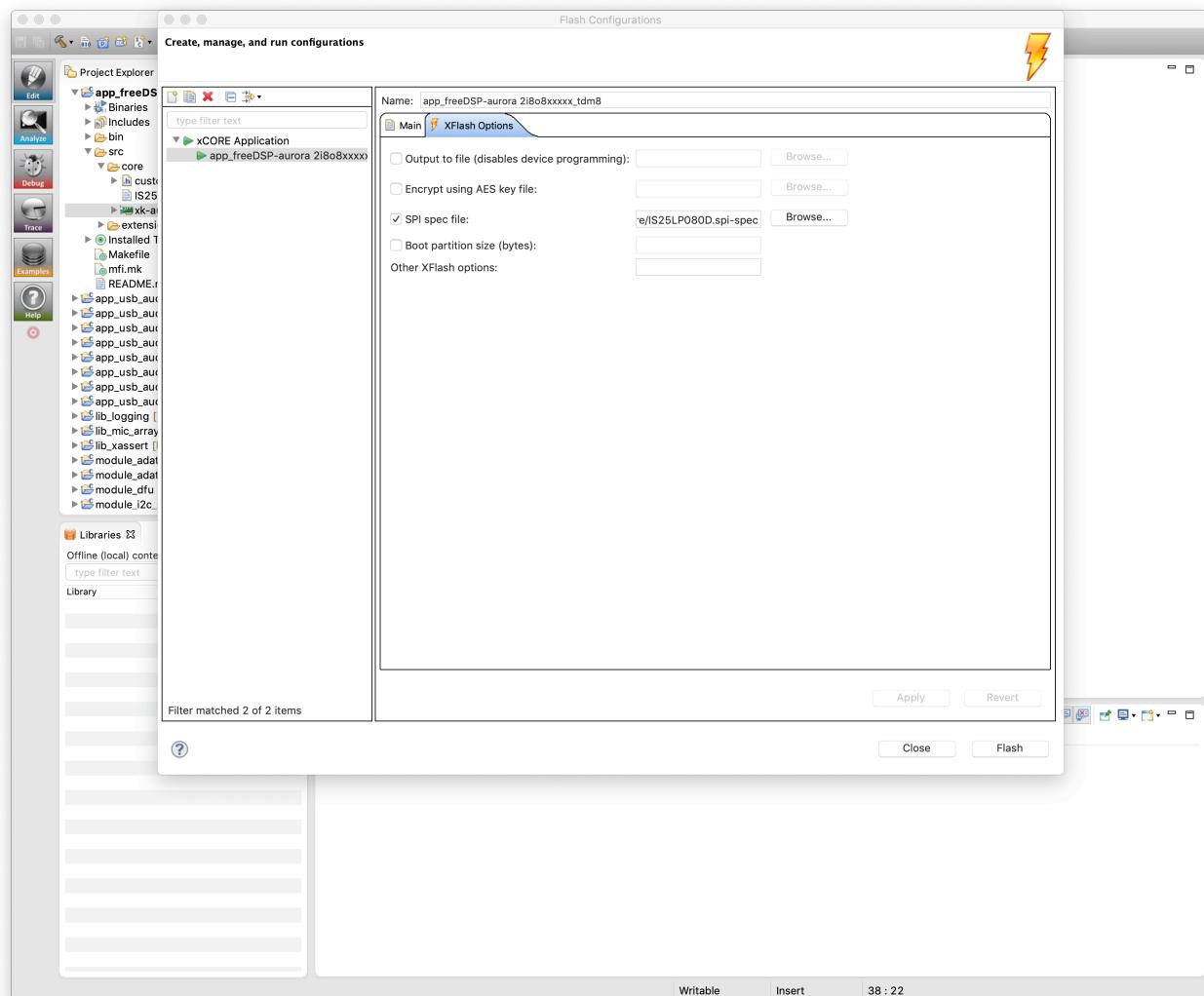
```
XCC_FLAGS_2i808xxxxx_tdm8 = $(BUILD_FLAGS) -DI2S_CHANS_DAC=8 \
-DI2S_CHANS_ADC=8 -DNUM_USB_CHAN_OUT=8 -DNUM_USB_CHAN_IN=8 \
-DMIDI=0 -DSPDIF_TX=0 -DSPDIF_RX=0 -DADAT_TX=0 -DADAT_RX=0 \
-DDSD_CHANS_DAC=0 -DI2S_MODE_TDM=1 -DMIN_FREQ=48000 \
-DMAX_FREQ=48000
```

12. Select the current build configuration by selecting the configuration you have edited in step 11. E.g. *Project->Build Configurations->Set Active->2i808xxxxx_tdm8*
13. Now it is time to click *Project->Build all* to build your firmware.
14. Open *Run->Flash Configurations*
15. Select xCORE Application and click on *New* button
16. Make sure that your settings look like the following screenshot. Only the target identifier string may differ.

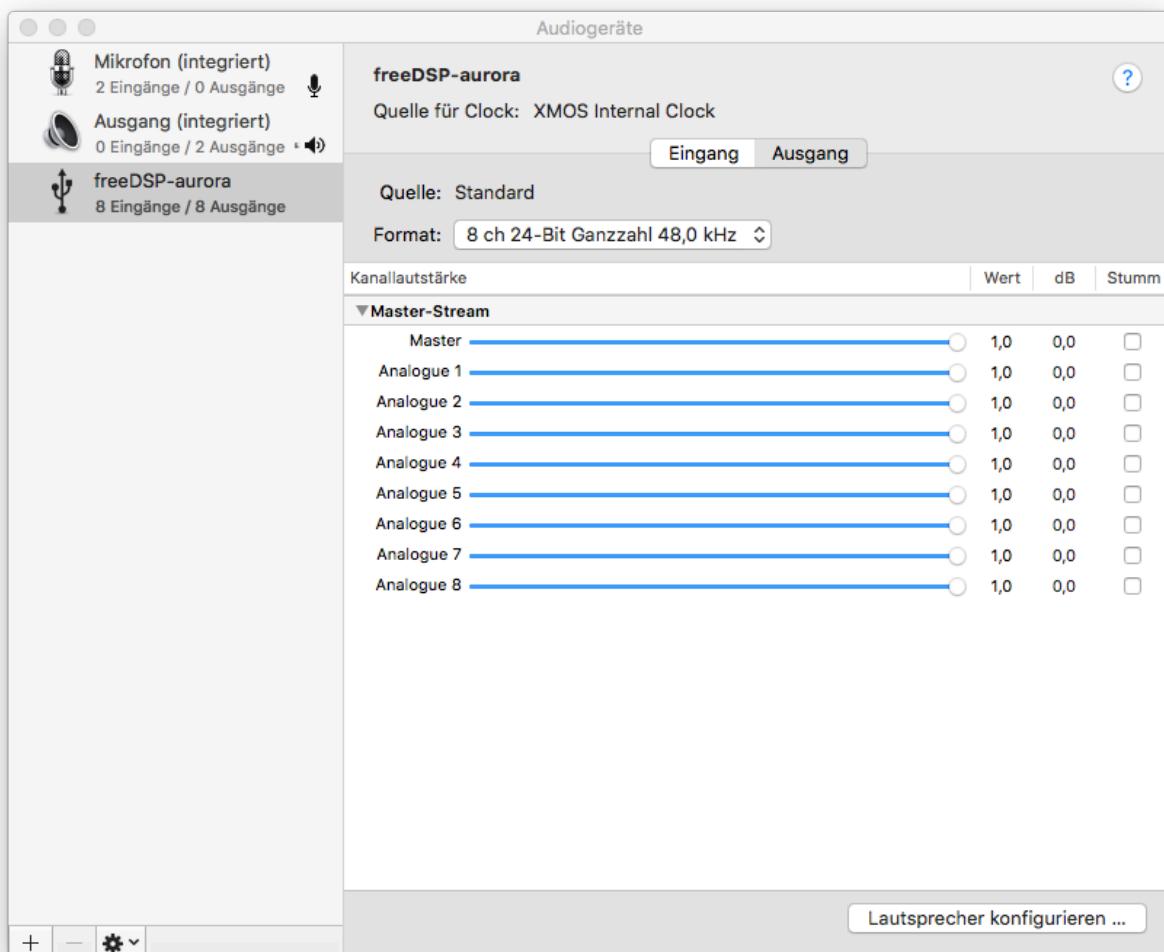
¹ You may want to choose another configuration depending on your required features, eg. ADAT



17. Switch to *XFlash Options* tab and set the check mark on SPI spec file and select the spi-spec file you have copied from the git repository.



18. Click on *Apply*
19. Click on *Flash* to upload the firmware to your board.
20. After success you should remove the xTAG and connect your freeDSP-aurora to your computer on the USB-B connector. It should now be enumerated by the operating system and e.g. for macOS being listed in the Audio-MIDI-Setup:





UPLOAD A DSP PLUGIN

FreeDSP-aurora is designed to run different DSP plugins. A DSP plugin is a firmware uploaded to the SigmaDSP. Each plugin can describe another functionality for the DSP. Normally DSP plugins are uploaded with auverdionControl app. Please refer to the auverdionControl manual for details about uploading a DSP plugin.

By updating auverdionControl to the latest version you will always get access to all DSP plugins, that are freely available.

Alternatively, you can upload a DSP plugin with an USBi programmer either from Analog Devices or with an freeUSBi. By connecting an USBi programmer you can directly communicate with the DSP from SigmaStudio and upload programs. This is very convenient for writing and testing your own DSP plugins.

Connect your USBi programmer to X104. The procedure is the same as in any other freeDSP based on ADAU1452. Please refer to manuals of other freeDSPs or the EVAL-ADAU1452MINIZ User Guide (http://www.analog.com/media/en/technical-documentation/user-guides/EVAL-ADAU1452MINIZ_User_Guide.pdf) for details about programming the DSP. SigmaStudio can be downloaded from http://www.analog.com/en/design-center/processors-and-dsp/evaluation-and-development-software/ss_sigst_02.html with no charge.

The most easiest way to store your new DSP plugin non-volatile is to add it to auverdionControl and adapt the GUI to control the parameters. Please refer to the source code of auverdionControl for details on this.

Alternatively, you can populate IC102 with 25AA1024T-I/SM EEPROM and remove the jumper JP101. You need to remove the firmware upload function from the ESP32 program and reprogram the ESP32, too. This approach is for advanced users only.

APPENDIX

PART LIST

Reference	Qty	Value1	Value2	Manufacturer	pn
C103	1	5n60	10% 50V X7R	Yageo	CC0805KRX7R9 BB562
C101, C104, C111, C117, C119, C120, C122, C123, C125, C127, C132, C259, C260, C337, C360, C409, C410, C412, C414, C415, C502, C507, C508	23	10u0	20% 25V X5R	Murata Electronics North America	GRM21BR61E10 6MA73L
C102, C105, C109, C110, C118, C121, C124, C128, C129, C130, C131, C133, C134, C135, C136, C220, C230, C234, C242, C249, C261, C262, C264, C273, C274, C275, C276, C277, C278, C279, C280, C320, C328, C336, C344, C352, C361, C362, C363, C364, C365, C366, C367, C368, C369, C370, C371, C372, C373, C374, C375, C376, C377, C402, C403, C404, C405, C406, C407, C408, C411, C413, C501, C503, C506	65	100n	5% 50V X7R	KEMET	C0805C104K5R ACTU
C106	1	150p	5% 50V C0G	KEMET	C0805C151J5G ACTU
C107, C108	2	22p0	5% 50V C0G/ NPO	KEMET	C0805C220J5G ACTU

Reference	Qty	Value1	Value2	Manufacturer	pn
C112, C113, C114, C137, C138, C203, C204, C211, C212, C217, C218, C219, C227, C228, C229, C233, C237, C238, C241, C247, C248, C255, C256, C269, C270, C301, C302, C309, C310, C314, C315, C321, C324, C325, C329, C330, C331, C340, C341, C345, C346, C347, C353, C356, C357	45	100u	20% 25V	Nichicon	UVY1E101MDD
C115, C116, C126	3	330u	20% 25V	Wurth Electronics Inc.	860010474012
C201, C210, C225, C226, C240, C252, C267, C268	8	1n00	5% 50V PPS	Panasonic Electronic Components	ECH- U1H102JX5
C205, C206, C213, C214, C221, C222, C231, C232, C236, C243, C244, C246, C257, C258, C271, C272, C281, C282, C283, C284, C285, C286, C287, C288	24	100p	5% 50V C0G/ NP0	KEMET	C0805C101J5G ACTU
C202, C207, C208, C209, C215, C216, C223, C224, C235, C239, C245, C251, C253, C254, C265, C266	16	220p	2% 50V PPS	Panasonic Electronic Components	ECH- U1H221GX5
C250	1	4u70	20% 25V X5R	Murata Electronics North America	GRM21BR61E47 5MA12L
C263, C401	2	10n0	10% 50V X7R	Yageo	CC0805KRX7R9 BB103
C305, C306, C318, C319, C334, C335, C350, C351	8	27p0	5% 50V C0G/ NP0	KEMET	C0805C270J5G ACTU
C307, C308, C322, C323, C338, C339, C354, C355	8	3n90	5% 16V PPS	Panasonic Electronic Components	ECH- U1C392JX5

Reference	Qty	Value1	Value2	Manufacturer	pn
C313	1	1u00	10% 16V X7R	Yageo	CC0805KX7R7 BB105
C303, C304, C311, C312, C316, C317, C326, C327, C332, C333, C342, C343, C348, C349, C358, C359	16	470p	2% 50V PPS	Panasonic Electronic Components	ECH- U1H471GX5
C416, C504	2	22u0	20% 10V X5R	Murata Electronics North America	GRM21BR61A22 6ME51L
C417, C418, C419, C420, C421, C422, C423, C424, C425, C426, C427, C428, C429, C430, C431, C432, C433, C434, C435, C436, C437, C438, C439, C440, C441	25	100n	10% 10V X7R	Murata Electronics North America	GRM155R71A10 4KA01D
D401	1	MBR120VLSFT	Schottky 20V 1 A	ON Semiconducto r	MBR120VLSFT3 G
FER101, FER102	2	600R	@100MHz 600mA	Murata Electronics North America	BLM21AG601SN 1D
IC101	1	ADAU1452	-	Analog Devices Inc.	ADAU1452WBCP Z
IC102	1	DNP	1Mbit	Microchip Technology	25AA1024T-I/ SM
IC103	1	TLE2426ID	Spannungsrefe renz	Texas Instruments	TLE2426IDR
IC104	1	ADM811TARTZ	Spannungsmoni tor	Analog Devices Inc.	ADM811TARTZ- REEL
IC105, IC107	2	LM1117MP-5. 0	LDO	Texas Instruments	LM1117MP-5.0 /NOPB
IC106, IC409, IC502	3	NCP1117LPST 33	LDO	ON Semiconducto r	NCP1117LPST3 3T3G

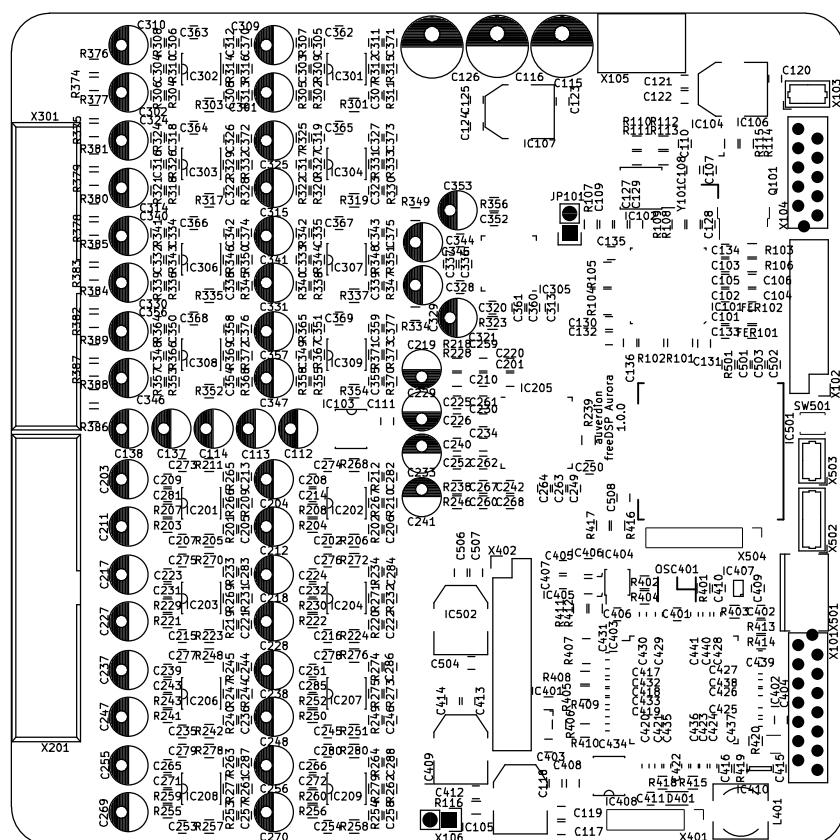
Reference	Qty	Value1	Value2	Manufacturer	pn
IC201, IC202, IC203, IC204, IC206, IC207, IC208, IC209, IC301, IC302, IC303, IC304, IC306, IC307, IC308, IC309	16	OPA1652AIDR	OpAmp	Texas Instruments	OPA1652AIDR
IC205	1	AK5558VN	ADC	AKM Semiconductor Inc.	AK5558VN
IC305	1	AK4458VN	DAC	AKM Semiconductor Inc.	AK4458VN
IC401	1	NC7WZ07	Noninverting Buffer	ON Semiconductor	NC7WZ07P6X
IC402	1	NC7SZ175	D-Type Flip-Flop with Asynchronous Clear	Texas Instruments	SN74LVC1G175DCKR
IC403	1	XE216-512-TQ128	xCore	XMOS	XE216-512-TQ128-C20
IC404	1	CS2100CP	Fractional-N Clock Multiplier	Cirrus Logic Inc.	CS2100CP-CZZ
IC405, IC406	2	NC7SZ157	2-Input Non-Inverting Multiplexer	ON Semiconductor	NC7SZ157P6X
IC407	1	TLV70025	2.5V, 200mA	Texas Instruments	TLV70025DCKR
IC408	1	IS25LP080D	8Mb	ISSI, Integrated Silicon Solution Inc	IS25LP080D-JNLE
IC410	1	TLV62565	1.5A	Texas Instruments	TLV62565DBVR
IC501	1	ESP32-WROOM-32U	Module	Espressif Systems	ESP32-WROOM-32U
JP101	1	Jumper	-	Wurth Electronics Inc.	61300211121
L401	1	2u20	30% 4.5A 26 mOhm	Murata Power Solutions Inc.	45222C

Reference	Qty	Value1	Value2	Manufacturer	pn
OSC401	1	24M0	-	Abracon LLC	ASFL1-24.000 MHZ-EC-T
Q101	1	STD2805	PNP 60V 5A 150MHz 15W	STMicroelectronics	STD2805T4
R101, R102, R104, R105, R407	5	33R0	5% 62,5mW	Yageo	TC164- JR-0733RL
R103, R114, R410	3	1K00	1% 0.125W Thick	Yageo	RC0805FR-071 KL
R106	1	4K32	1% 0.125W Thick	Yageo	RC0805FR-074 K32L
R108	1	100R	1% 0.125W Thick	Yageo	RC0805FR-071 00RL
R109, R239, R402, R404, R408, R409, R411, R412	8	33R0	1% 0.125W Thick	Yageo	RC0805FR-073 3RL
R110, R111, R416, R417	4	2K00	1% 0.125W Thick	Yageo	RC0805FR-072 KL
R107, R112, R113, R401, R405, R406, R413, R414, R501	9	10K0	1% 0.125W Thick	Yageo	RC0805FR-071 0KL
R115, R374, R375, R376, R377, R378, R379, R380, R381, R382, R383, R384, R385, R386, R387, R388, R389, R418	18	100K	1% 0.125W Thick	Yageo	RC0805FR-071 00KL
R116	1	560R	1% 0.125W Thick	Yageo	RC0805FR-075 60RL
R203, R204, R207, R208, R221, R222, R229, R230, R241, R243, R250, R252, R255, R256, R259, R260	16	10K0	1% 0.25W Thin	Stackpole Electronics Inc.	RNCP0805FTD1 0K0
R201, R202, R209, R210, R219, R220, R231, R232, R240, R244, R249, R253, R254, R261, R262, R273	16	6K80	0.5% 0.125W Thin	Panasonic Electronic Components	ERA-6AED682V

Reference	Qty	Value1	Value2	Manufacturer	pn
R205, R206, R211, R223, R224, R242, R248, R251, R257, R258, R268, R270, R272, R276, R278, R280	16	10R0	1% 0.25W Thin	Stackpole Electronics Inc.	RNCP0805FTD1 0R0
R218, R228, R238, R246, R323, R334, R349, R356	8	20R0	1% 0.125W Thick	Yageo	RC0805FR-072 0RL
R212, R233, R234, R245, R247, R263, R264, R265, R266, R267, R269, R271, R274, R275, R277, R279	16	2K20	0.5% 0.125W Thin	Panasonic Electronic Components	ERA-6AED222V
R301, R303, R311, R313, R317, R319, R328, R330, R335, R337, R345, R347, R352, R354, R368, R370	16	4K42	0.5% 0.1W Thin	Susumu	RR1220P-4421 -D-M
R302, R304, R315, R316, R318, R320, R332, R333, R336, R338, R350, R351, R353, R355, R372, R373	16	3K40	0.5% 0.1W Thin	Susumu	RR1220P-3401 -D-M
R305, R306, R312, R314, R321, R322, R329, R331, R339, R340, R346, R348, R357, R358, R369, R371	16	150R	1% 0.25W Thin	Stackpole Electronics Inc.	RNCP0805FTD1 50R
R307, R308, R309, R310, R324, R325, R326, R327, R341, R342, R343, R344, R364, R365, R366, R367	16	4K70	1% 0.125W Thin	Yageo	RT0805FRE074 K7L
R403	1	4R70	1% 0.125W Thick	Yageo	RC0805FR-074 R7L
R415	1	43R2	1% 0.125W Thick	Panasonic Electronic Components	ERJ-6ENF43R2 V
R419	1	80K6	1% 0.125W Thick	Yageo	RC0805FR-078 0K6L
R420	1	120K	1% 0.125W Thick	Yageo	RC0805FR-071 20KL

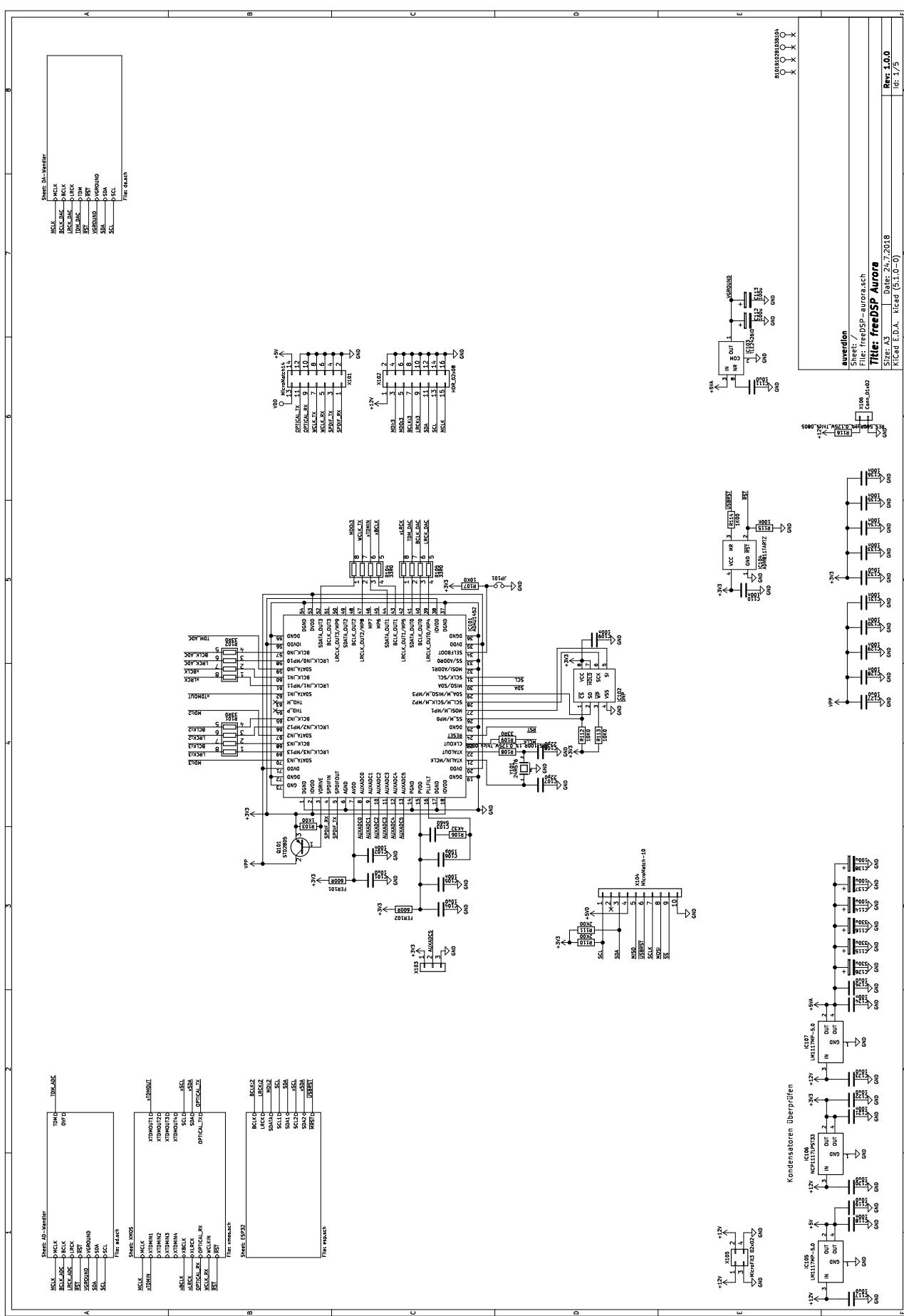
Reference	Qty	Value1	Value2	Manufacturer	pn
SW501	1	B3U-1000P	SWITCH TACTILE SPST- NO 0.05A 12V	Omron Electronics Inc-EMC Div	B3U-1000P
X101	1	MicroMatch-14	-	TE Connectivity AMP Connectors	1-215079-4
X102	1	HDR_02x08	-	Sullins Connector Solutions	PRPC008DAAN-RC
X103, X503	2	PB-01x03	-	Molex, LLC	530470310
X104	1	MicroMatch-10	-	TE Connectivity AMP Connectors	1-215079-0
X105	1	MicroFit3 02x02	-	Molex, LLC	430450428
X106	1	HDR-01x02	-	Wurth Electronics Inc.	61300211121
X201, X301	2	IDC26	-	Bud Industries	BC-32677
X401	1	HDR-01x05	-	Sullins Connector Solutions	PRPC005SAAN-RC
X402	1	HDR-02x10	-	Sullins Connector Solutions	PRPC010DAAN-RC
X501	1	KK254-01x04	-	Molex, LLC	22232041
X502	1	PB-01x05	-	Molex, LLC	530470510
X504	1	HDR-01x06	-	Sullins Connector Solutions	PRPC006SAAN-RC
Y101	1	24M576	-	EPSON	FA-238 24.5760MB-K3
	1	RF ANT 2.4GHZ	WHIP TILT CABLE	PulseLarsen Antennas	W1049B050

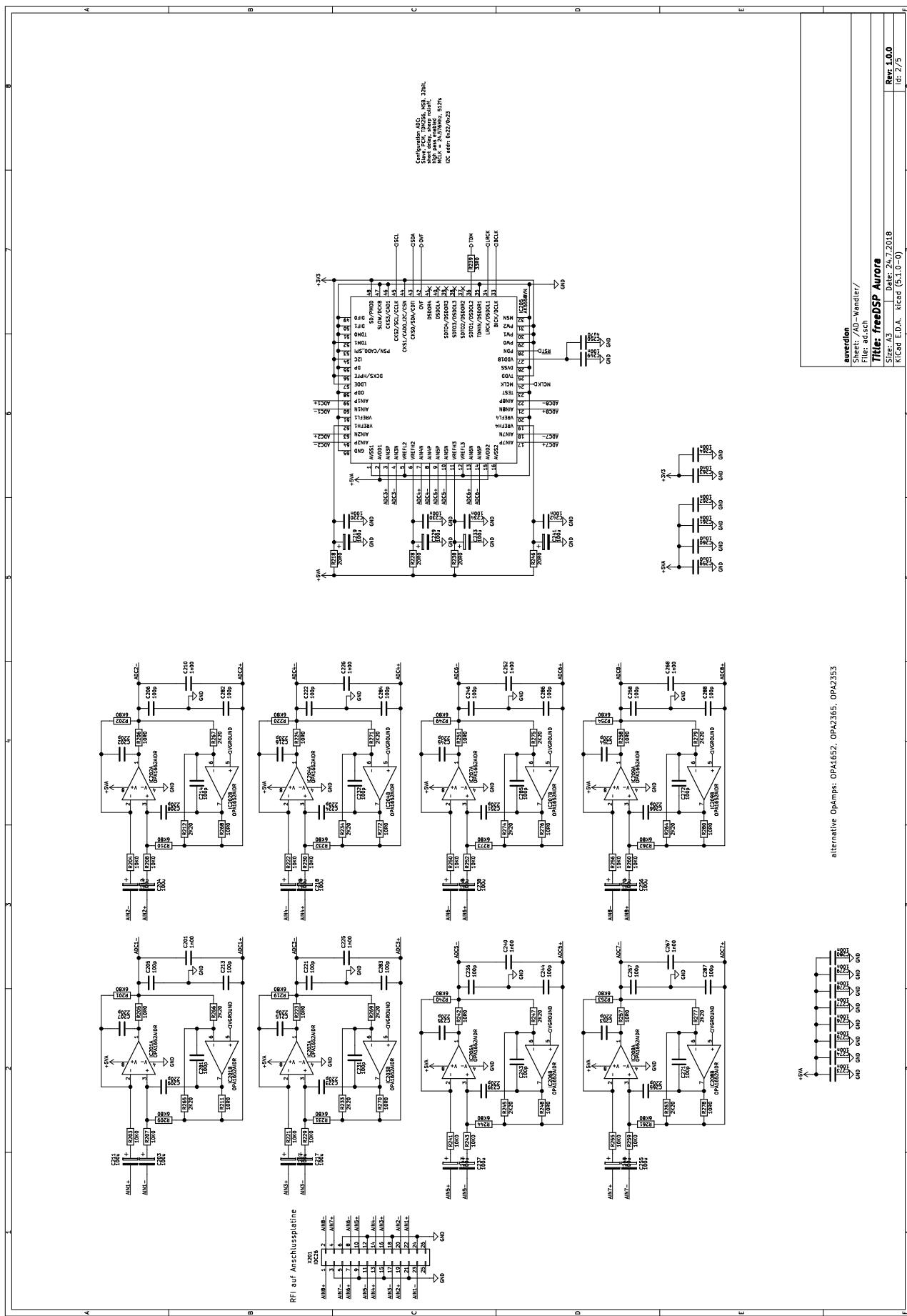
ASSEMBLY PRINT

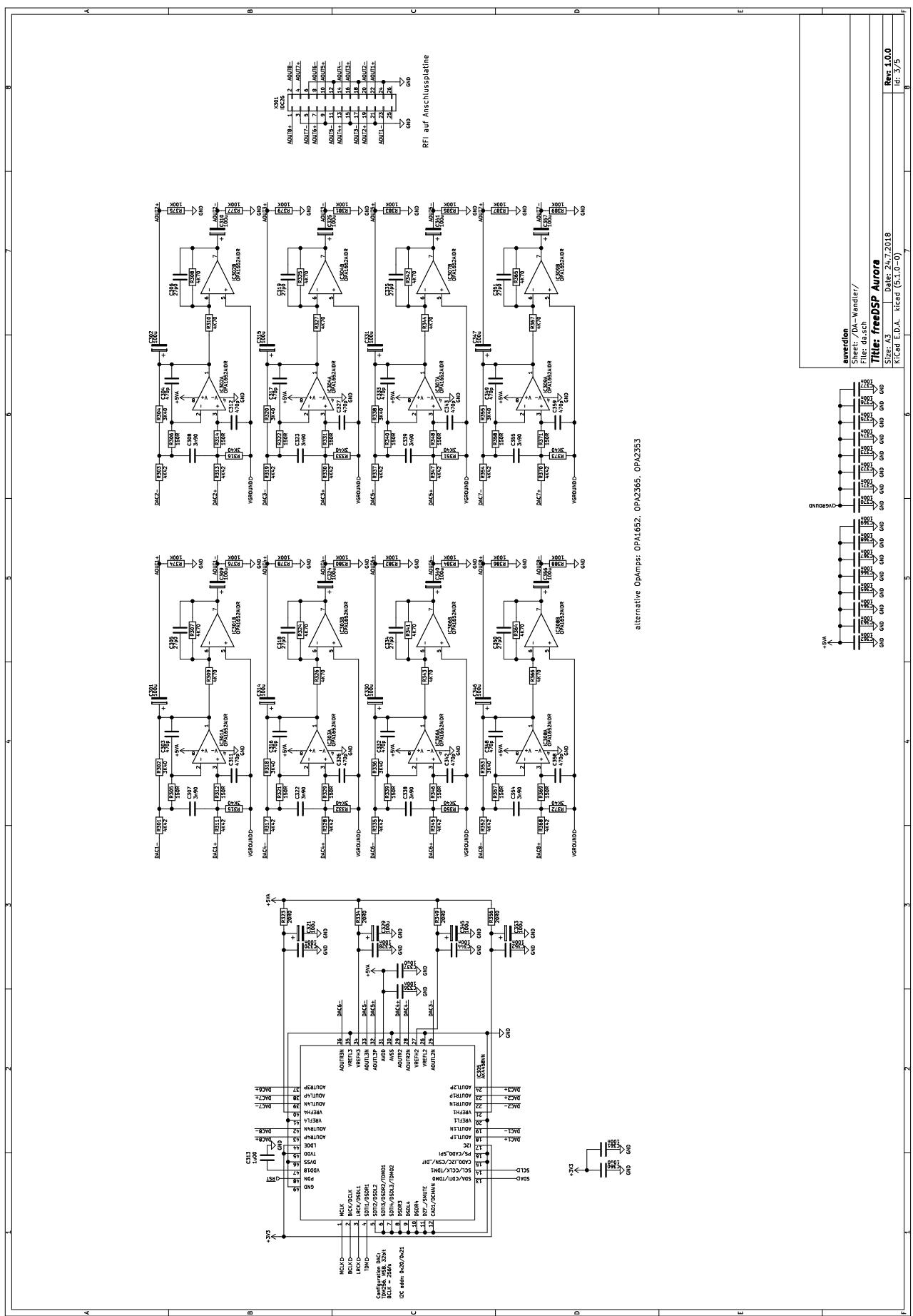


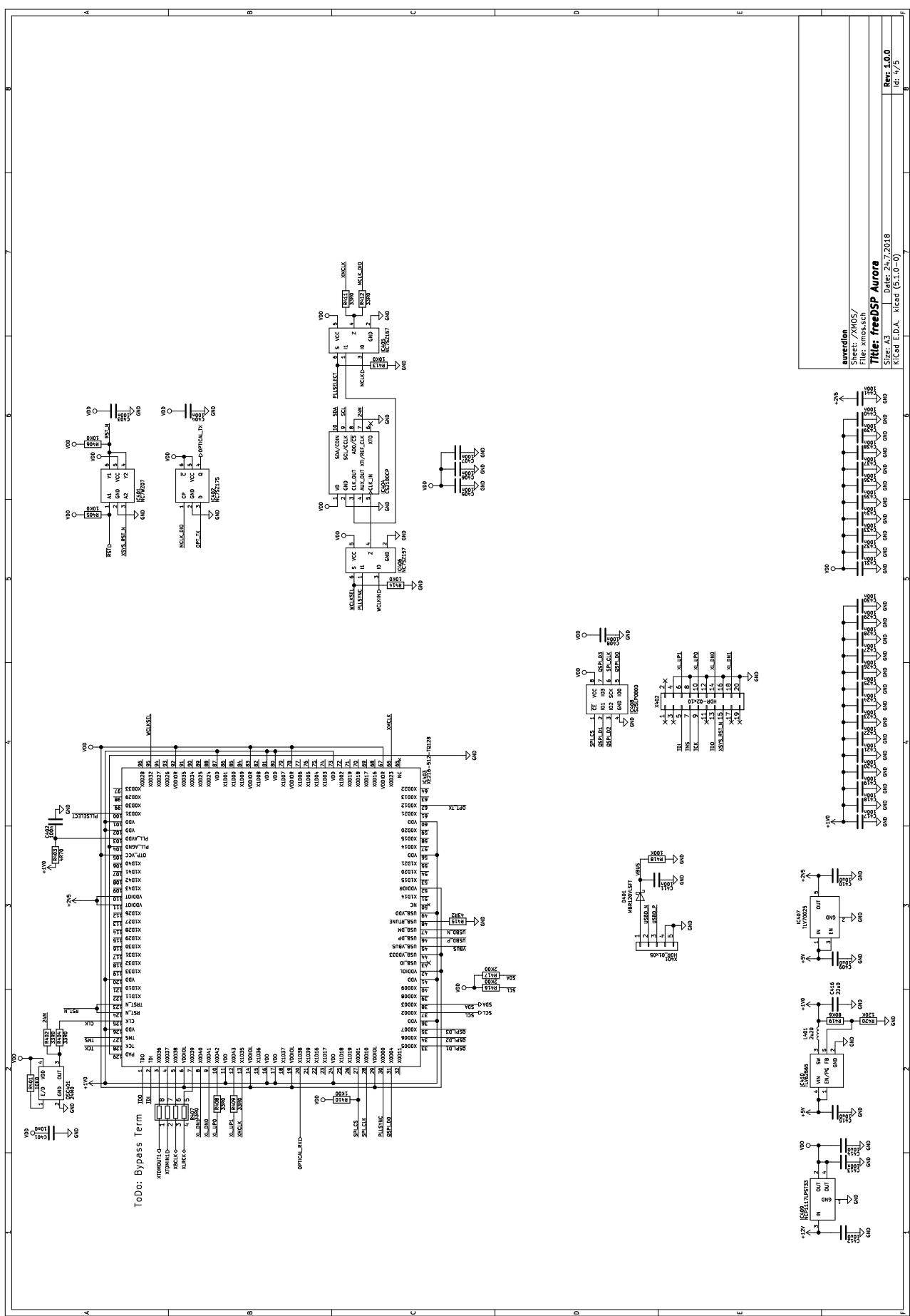


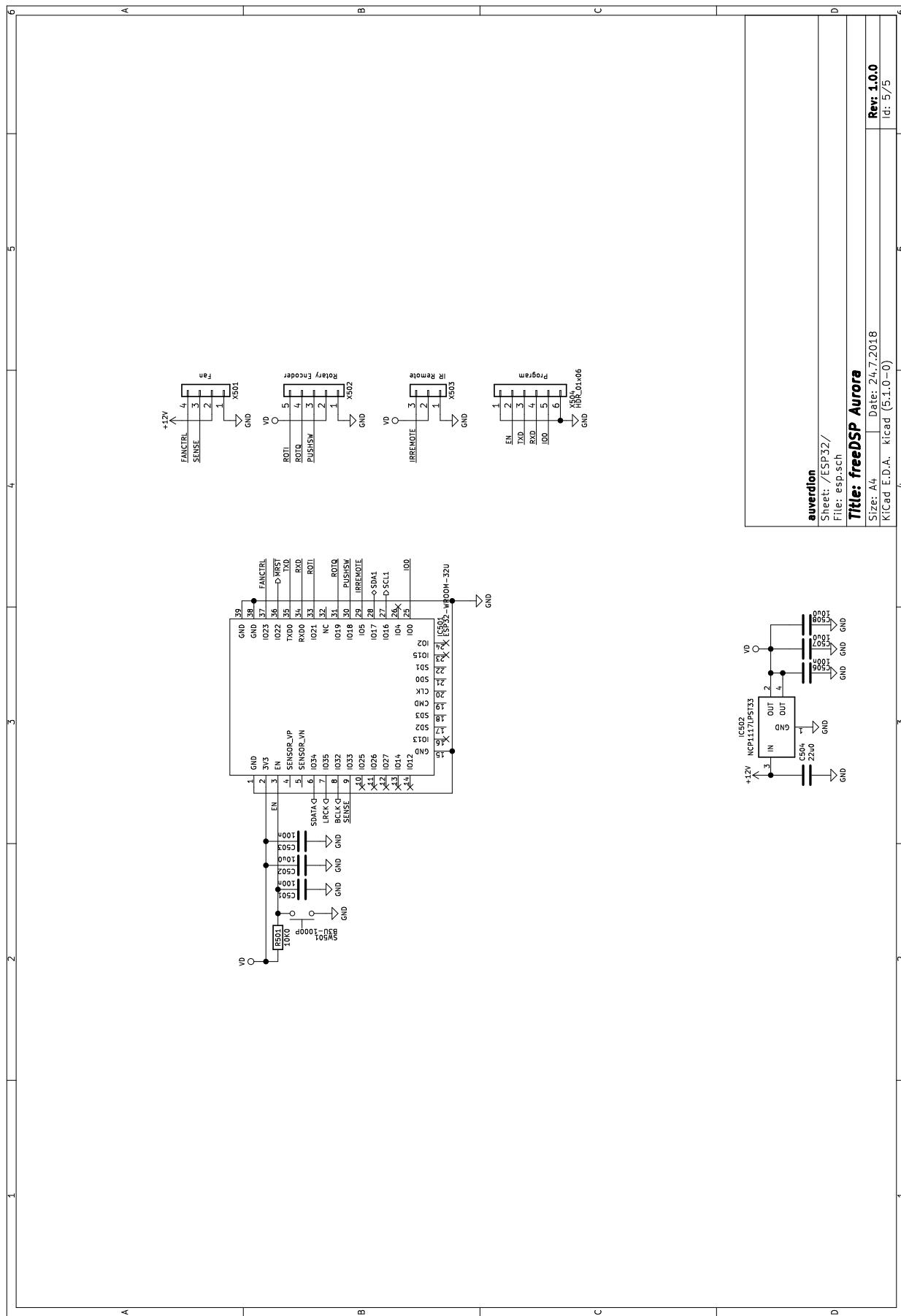
SCHEMATICS













Please visit <https://github.com/freeDSP/freeDSP-AURORA> for updates, bugfixes and new DSP plugins.

Enjoy your freeDSP-aurora and build something awesome!