

Contents

1	Introduction	3
1.1	Intent	3
1.2	How to Use this Document	3
1.3	Overview of Operation	4
2	Circuit Description	5
2.1	Power	6
2.1.1	Overall Layout	6
2.1.2	Barrel Jack / Power Input	6
2.1.3	TPS5420D Buck Converter	6
2.1.4	L7980 Buck Converter	8
2.2	Top Level Schematic	14
2.2.1	KT2520K 40MHz TXCO	14
2.2.2	NC7S04 Inverter	14
2.2.3	NB3N551 1:4 Clock Buffer	14
2.3	TX	15
2.3.1	ADF4158 Frequency Synthesizer	15
2.3.2	HMC431LP4RF VCO	23
2.3.3	SE2567L Power Amplifier	23
2.3.4	Notes	23
2.4	USB	25
2.4.1	FT2232H	25
2.4.2	93LC46B 1k EEPROM	25
2.5	FPGA	27
2.5.1	XC7A15T-FTG256 Xilinx FPGA	27
2.6	ADC	36
2.7	Mixer	40
2.7.1	ADL5802	40
2.8	IF	41
2.9	RX1 & RX2	42
3	PCB	43
3.1	General Layout	43
3.2	Component Layout Considerations	43
3.2.1	Power Input	43
3.2.2	LTC2292 ADC	44
3.2.3	RX1 / RX2	44
3.3	RF Impedance Matching	44

4	Antennas	53
5	Links	54
6	Full Schematic	55

Chapter 1

Introduction

1.1 Intent

This document explains, in a great amount of detail, the design, operation and layout of an FMCW radar PCB. The radar was originally designed by Henrik Forstén and is described [in his blog post](#). His source code is hosted on GitHub [here](#). I have made this document in an effort to understand his design and to build it myself. While most of the design remains his work, I have made several modifications for deprecated parts or other design choices. I have also made an effort to update the schematic/footprints to use the default KiCad schematic and footprint libraries where possible. The design is a work in progress and will likely be updated through several different versions in order to make improvements and add functionality. I also hope to make the design more my own as it is modified and updated. This document will be kept up-to-date with those changes.

1.2 How to Use this Document

This document contains several sections. The first is an in-depth description of the circuit and the function of each component in it. Each section corresponds to a schematic sheet in the design. Below each section header is a hyperlink to a PDF version of that sheet of the schematic. The hyperlink is labeled “schematic” and is italicized. The full schematic in PDF form is contained at the end of this document. I have opted for this format instead of embedding diagrams of individual components throughout the document, which I believe is more aesthetically pleasing and is easier to maintain as the schematic is updated. Each section is then further subdivided into subsections composed of the individual components in the schematic sheet. They provide a detailed description of the component’s operation and how it interacts with other components on the PCB. I also frequently include a table listing the full pinout of the component, which includes the purpose of each pin and its connections in the schematic.

The following section describes the layout of the PCB in detail. It contains a subsection describing overall layout structure, including ground planes. I’ve also included a subsection for specific component layout considerations. I’ve left out considerations that are generally good practice and are thus applicable to most digital circuits. For instance, I’ve omitted justifications for bypassing unless the component’s datasheet recommends a specific layout that differs from the default recommendations. Lastly, I’ve included a section for impedance matching.

The last section (prior to the full schematic) contains a selection of useful hyperlinks.

1.3 Overview of Operation

This design uses one antenna for transmitting and two for reception.

Chapter 2

Circuit Description

2.1 Power

schematic

Component	MFN	Links
Ferrite bead	CIC21P101NE	Datasheet , Digi-Key
Buck Converter	TPS5420D	Datasheet , Digi-Key

2.1.1 Overall Layout

A barrel jack is used to feed a 12V input to the board which is then fed into several buck converters that each output different voltages for use by linear regulators. The linear regulators then produce the final voltage levels used by the various digital ICs on the PCB. An LED is connected to the output of one of the converters and is used to indicate that power is being administered to the board. Switching converters are used in several cases as intermediaries between the 12V input and linear regulators in order to increase energy efficiency and the amount of current that can be drawn by the ICs. Using linear regulators downstream from the switching converters produces less noise in the power supply lines. For the 10V power supply the switching converter is unnecessary and a linear regulator is used directly.

2.1.2 Barrel Jack / Power Input

A ferrite bead pi filter is placed at the output of the barrel jack connection. The ferrite bead used is specifically designed for high current and power line noise suppression. The barrel connector is a switched jack, but we do not use a battery on this PCB so the 3rd pin is grounded.

2.1.3 TPS5420D Buck Converter

The design uses 2 buck converters to bring the 12V input down to 5.6V and 3.6V intermediates. The block diagram for the TPS5420D is shown in Figure 2.1.

Downstream Current Draw

Table 2.2: The current draw of components downstream from the 5.6V buck converter. Where the datasheet is ambiguous, I've assumed worst-case current draw.

MFN	I _Q	I _{max}	Notes
ADL5802	170mA	300mA	5V
ADF4158	312.5μA	5mA	5V (VP)
SE5004L	300mA	800mA	5V (VCC1, VCC2, VCC3)
Total	470mA	1.105A	-

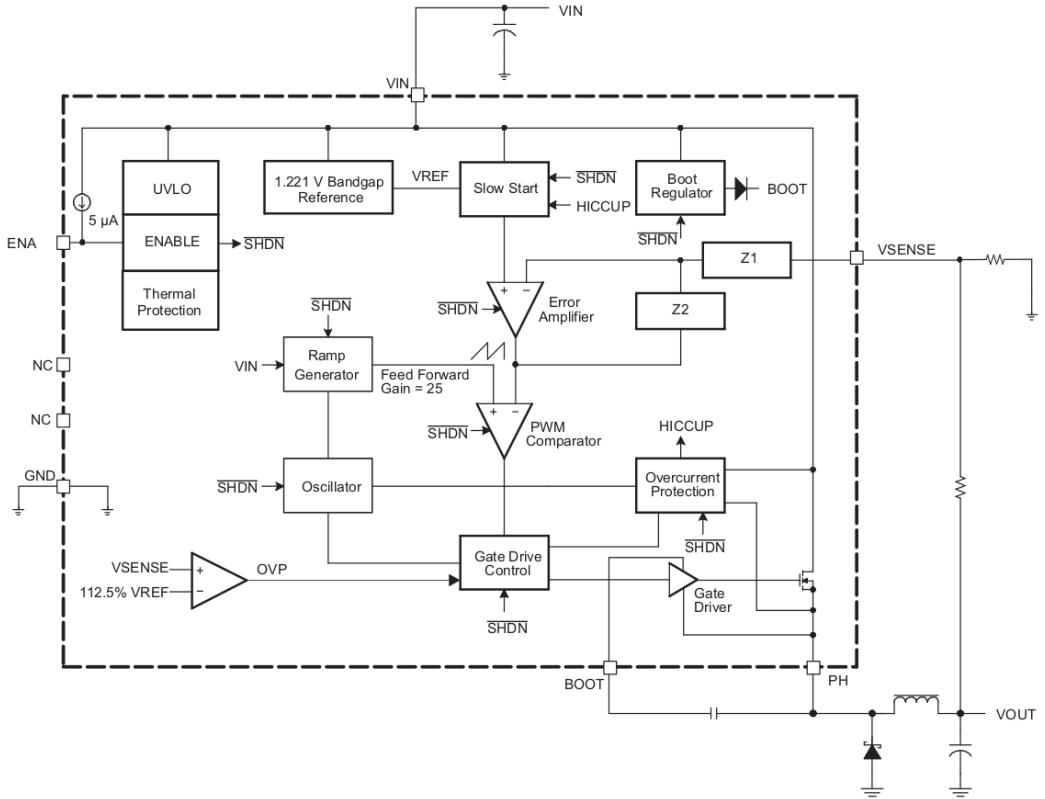


Figure 2.1: TPS5420D Block Diagram

Table 2.3: Components downstream from the 3.6V buck converter.

MFN	I _Q	I _{max}	Notes
KT2520K	-	2mA	1.8V
NC7S04	1 μ A	12.5mA	3.3V
NB3N551	-	40mA	3.3V
XC7A15T-FTG256	97mA	277mA	1V (VCCINT, VCCBRAM)
XC7A15T-FTG256	22mA	87mA	1.8V (VCCADC, VCCAUX)
XC7A15T-FTG256	1mA	201mA	3.3V (VCC0)
W25Q32JV	10 μ A	25mA	3.3V
FT2232H	510 μ A	280mA	3.3V (VPHY, VPPLL, VCORE, VCCIO)
93LC46B	-	2mA	3.3V
LTC2292	5mA	95mA	3.3V (OVDD), 3V (VDD)
ADA4940-2	4.2mA	5.52mA	3.3V
ADF4158	-	32mA	3.3V (AVDD, DVDD)
HMC431LP4	19mA	27mA	3V
TRF37A73	250 μ A	130mA	3V
SKY65404	20mA	36mA	3V (VENABLE, VCC)

Table 2.3: Components downstream from the 3.6V buck converter.

MFN	I_Q	I_{max}	Notes
Total	169mA	1.252A	-

Pinout

FiXme Fatal: What is the point of BOOT?

Table 2.4: TPS5420D pinout.

Pin	Description
BOOT	
VSENSE	Feedback voltage for the regulator. This compares the divided output with a 1.221V reference.
ENA	Enable pin. This can be left floating to enable the device.
VIN	Input supply voltage.
PH	Output voltage.

Component Selection

We will choose an inductor such that it satisfies the following 3 requirements:

1. The inductance is given by Equation 2.1.
2. The current rating is 2x the maximum load current.
3. The self-resonant frequency is at least 10x the switching frequency.

$$L_{min} = \frac{T}{2} \frac{V_{out}}{I_{out(min)}} \left(1 - \frac{V_{out}}{V_{in}} \right) \quad (2.1)$$

The requirements are given in Table 2.1.3. All of these requirements are satisfied by the [SRR1210A-330M](#) 33 μ H ferrite bead by Bourns.

Table 2.5: TPS5420D inductor requirements. I've assumed an input ripple of 300mV.

Buck Vout	Lmin	Current Rating	SRF min
5.6V	6.5 μ H	2.21A	5MHz
3.6V	15.1 μ H	2.50A	5MHz

2.1.4 L7980 Buck Converter

The L7980's block diagram is shown in Figure 2.2 and its pin connections are described in Table 2.1.4.

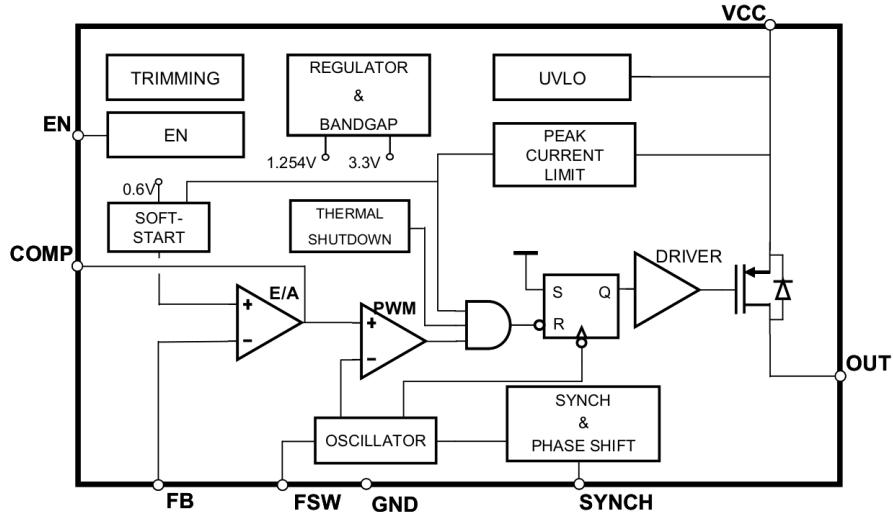


Figure 2.2: L7980 buck converter block diagram.

Table 2.6: L7980 pin connections.

Pin	Description
OUT	Regulated output voltage.
SYNCH	Synchronization pin that can be used when multiple L7980s are used on the same board. This pin serves a dual purpose: it allows an L7980 with a higher switching frequency (set with FSW) to synchronize an L7980 with a lower switching frequency and shifts its phase by half a period. That is, the “master” (the L7980 with a higher frequency) device makes its internal frequency available on this pin but phase shifted. Now, the lower frequency converters have that higher frequency at a phase shift. This reduces noise through destructive interference. This can also be used with an externally supplied frequency, but we don’t do that here.
EN	Active-high enable pin.
COMP	The error amplifier output which is used for loop frequency compensation. More on this below.
FB	The feedback input. This is fed into the inverting input of an error amplifier against a 0.6V reference. Therefore, we connect our output with a voltage divider to this pin such that it is 0.6V when we have the desired output voltage.
FSW	A resistor can be connected between this pin and ground to increase the converter’s switching frequency. See Equation 2.2. Leaving the pin free-floating sets the frequency at 250kHz, which is its minimum.
GND	Ground.
VCC	The unregulated DC input voltage. In our case, 12V from the barrel jack.

The radar contains 3 L7980s that output 5.6V, 3.6V and 1V. The two higher voltages are fed to linear regulators for further down-regulation and noise-filtering. The last is fed directly as one of the input voltages to the FPGA.

Switching Frequency

The equation relating the resistor value at FSW to the switching frequency is:

$$R_{SW} = \frac{28.5 \times 10^9}{F_{SW} - 250 \times 10^3} - 3.23 \times 10^3 \quad (2.2)$$

The 5.6V output buck converter uses a resistor of $59\text{k}\Omega$, which corresponds to a switching frequency of 708kHz. This converter exports its internal frequency to the 3.6V output converter, so although that converter uses a $68\text{k}\Omega$ resistor, it also switches at 708kHz. The 1V output converter does not have its synchronization pin tied to the other two so its frequency (determined by the $68\text{k}\Omega$ resistor) is 650kHz.

Inductance Value

The benefit of a higher switching frequency is a lower minimum inductance value, which decreases the cost of a circuit ($L_{min} \propto 1/F_{SW}$). To select an inductor value, we use Equation 2.3.

$$\Delta I_L = \frac{V_{IN} - V_{OUT}}{L} T_{ON} \quad (2.3)$$

The duty cycle for buck converters is given by Equation 2.4. Since $V_{IN} = 12V$ and $V_{OUT} = 5.6V$, $D = 47\%$. We already found that the switching frequency is 708kHz, which gives a total period of $1.41\mu\text{s}$ and therefore $T_{ON} = 0.659\mu\text{s}$. The inductor value used is $33\mu\text{H}$ which gives an inductor current ripple of 128mA. This 5.6V output is used to power an ADL5802 mixer (§ 2.7.1), which has a quiescent current of 220mA. Since half the inductor ripple is less than the mixer's minimum current draw, the inductor will stay in CCM and the mixer will see a stable output voltage.

$$DV_{IN} = V_{OUT} \quad (2.4)$$

PWM

This device uses a voltage-mode PWM comparator to stabilize the output voltage at its desired level. The block diagram shows how this works. The divided output voltage is fed back into the inverting input of the error amplifier and a 0.6V reference is fed to the non-inverting input. The error amplifier's output is compared against a sawtooth waveform at the inverting input, generated by the built-in oscillator. As long as the error amplifier voltage is greater than that of the sawtooth waveform the flip-flop reset is not triggered and the PMOS transistor is switched on. However, once the sawtooth waveform voltage becomes greater than that of the error amplifier, the active-low reset will be triggered on the flip-flop and the transistor will switch off until the next clock pulse from the internal oscillator. So, when the output voltage goes above its intended level the duty cycle decreases and when it goes below its intended level the duty cycle increases.

Loop Compensation

The COMP pin is used to provide feedback to the error amplifier, which in turn generates the error voltage signal that is compared with the internal sawtooth signal to determine the pulse

modulation that controls the voltage output. There are two different compensation networks that we can use, as described in the datasheet: type II compensation and type III compensation. The datasheet recommends the use of type III compensation for MLCC capacitors given their low ESR. **{START INCOMPLETE}** The actual calculation of these values is tricky and requires knowledge of feedback and op-amps, which in turn requires prerequisite knowledge of transistors. I believe it's probably necessary to complete chapters 1-4 of the Art of Electronics to understand this, so I intend to come back to it. On the upside, this should provide a solid background for DSP **{END INCOMPLETE}**.

The buck converter outputs feed into several different chips, the first of which is an LDO regulator for which an example is shown in Figure 2.3. The reason for attaching an LDO regulator in series with a buck converter may be to reduce the noise of the circuit and increase the efficiency at the expense of board area and cost. Particularly, the **LDO regulator** should be able to smooth out the noise caused by using an inductor with a low value in the buck converter. It is meant to effectively reject noise from the input source over the frequency range of 10Hz to 10MHz. Since the switching frequency of the buck converter is less than 1MHz and consequently the current/voltage ripple should be of the same frequency, the LDO regulator should be sufficient to smooth out the input noise provided, however, that it does not dip below the output voltage of the LDO regulator since the margin for error is small. $V_{IN} - V_{OUT} = 0.6V$ and linear regulators are not able to output a voltage greater than their input voltage. The max dropout of the LDO is 0.2V, which allows the output voltage of the buck converter a deviation of 0.4V.

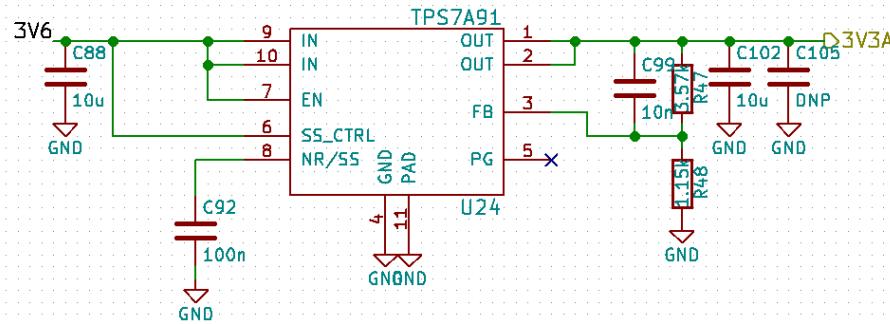


Figure 2.3: A LDO regulator that takes as an input the voltage from one of the buck converters and whose output is directly used to drive the operation of one of the ICs elsewhere in the circuit.

The input voltage is connected to both IN pins with a bypass capacitor of $10 \mu F$, which is the minimum specified by the datasheet. It is additionally connected to the EN pin which activates the LDO; the LDO is active for $V_{EN} \geq V_{IH}$ and disabled for $V_{EN} \leq V_{IL}$ (connecting the same input to IN and EN will make $V_{EN} = V_{IH}$, thus satisfying the first condition). The 2 resistors attached to OUT and FB make a voltage divider that determines the output voltage of the linear regulator, according to Equation 2.5.

$$R_1 = R_2 \left(\frac{V_{OUT}}{V_{REF}} - 1 \right) \quad (2.5)$$

Where $V_{REF} = 0.8V$. Specifically, 1.15k and 3.57k are specified in the datasheet in order to get an output voltage of 3.3V. SS_CTRL is connected to IN instead of GND which increases the soft-startup charging current to $100 \mu A$ from $6.2 \mu A$ and decreases the start-up time, which is given by Equation 2.6.

$$t_{SS} = (V_{REF} \times C_{NR/SS}) / I_{NR/SS} \quad (2.6)$$

$$= 0.8V \times 100 \times 10^{-9}F / (100 \times 10^{-6}A) \quad (2.7)$$

$$= 0.8\text{ms} \quad (2.8)$$

An output capacitor of $10 \mu\text{F}$ is chosen which is the minimum specified by the datasheet. Additionally, a feed-forward capacitor of 10nF is added between the OUT and FB pins to improve the noise and PSRR performance of the voltage regulator. The value is recommended by the datasheet. The $C_{NR/SS}$ capacitor of 100nF is used to create an output RC filter for output noise and is also used to set the soft-start time. A value of between 10nF and $10\mu\text{F}$ is recommended. PAD and GND should both be connected to ground, as indicated by the datasheet. PG indicates whether the output voltage is in a usable state. Since we are not using it, it can be left floating.

Another buck converter output feeds into the input of an ultra low LDO, with a dropout voltage of 40mV as shown in Figure 2.4. R104 acts as a pull-up resistor, pulling the voltage on EN high when PG is not driven. However, when PG is low, EN will be driven low. The bypass pin is left open, although a $1\mu\text{F}$ capacitor could have been placed there to reduce output noise.

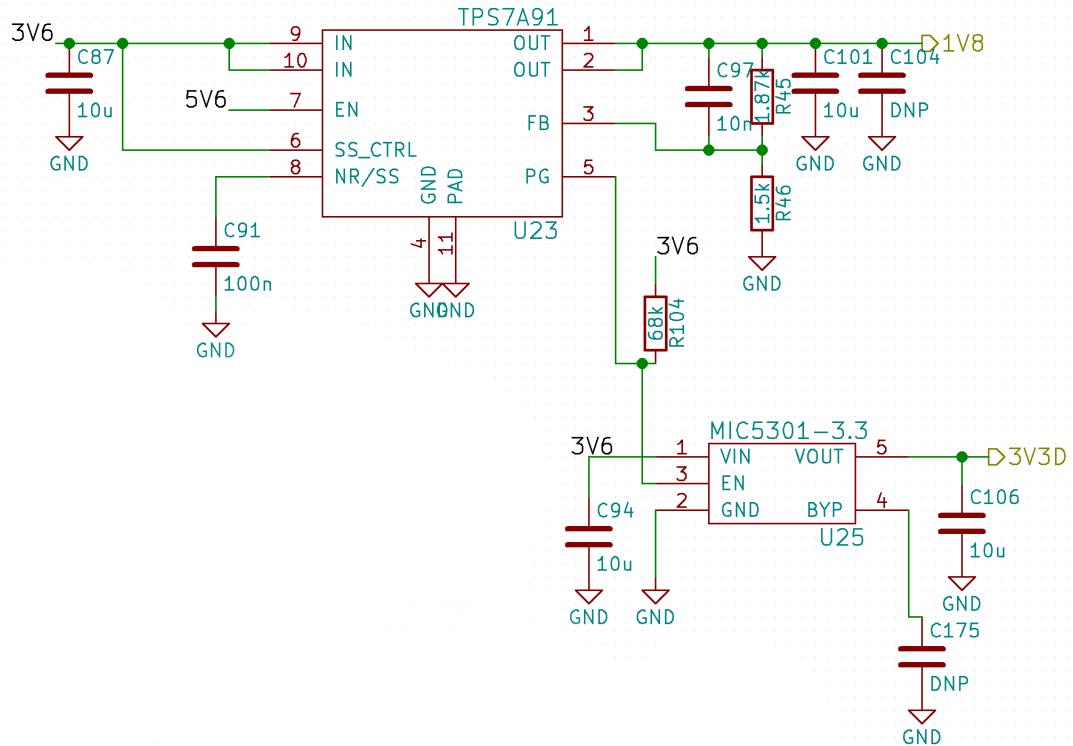


Figure 2.4: An ultra LDO regulator whose output feeds into an FPGA input. It uses the PG pin from another LDO regulator to alternately enable/disable it.

Another output from a buck converter leads to a 3V output LDO regulator with a dropout voltage of 120mV shown in Figure 2.5. All of the pin connections are evident.

The last component on the power sheet is an [LP2985](#) LDO regulator shown in Figure 2.6, that takes the 12V input voltage from the barrel jack and outputs 10V . ON/OFF is an active-low

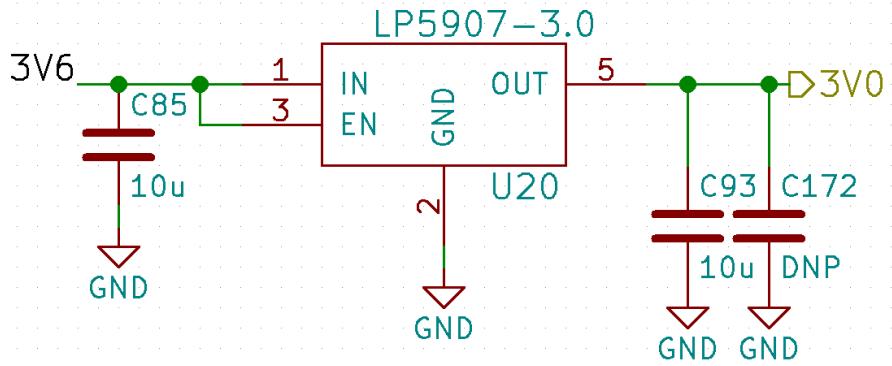


Figure 2.5: The LP5907 is a 3V LDO regulator with a dropout voltage of 120mV.

shutdown pin, so it is tied to V_{IN} . A 10nF capacitor is tied to BYPASS to decrease the output noise.

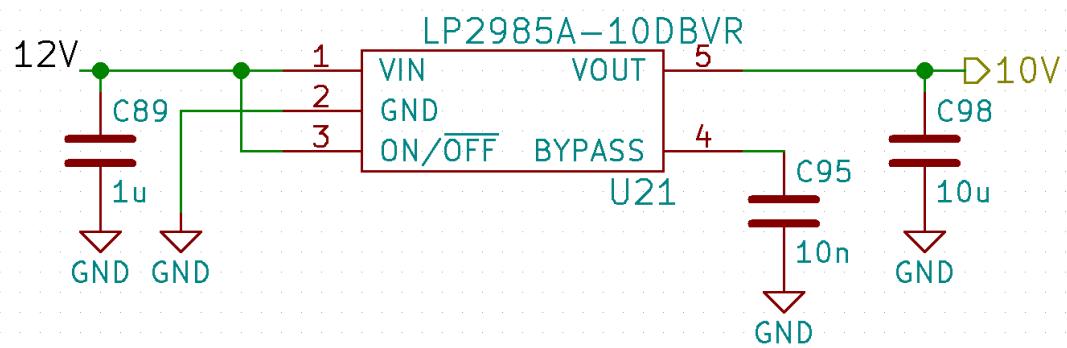


Figure 2.6: The LP2985 LDO regulator.

2.2 Top Level Schematic

schematic

The top level sheet instantiates the various subcomponents of the design. Additionally, it generates a 40MHz clock signal that feeds into a clock buffer and is output to various components, as shown in Figure 2.7. The LP5907 is a voltage regulator that takes a 3.6V input signal generated by a buck converter on the power sheet and outputs a 1.8V signal to power the KT2520K crystal oscillator which generates a 40MHz clock signal. The coupling capacitor attached in series to the output of the crystal oscillator will prevent a DC bias from leaving the crystal oscillator. The Schmitt inverter is used to turn the clipped sine wave output of the crystal oscillator into a square wave. By connecting the output of the Schmitt inverter back to its input via two resistors with a bypass capacitor between them, the input sine wave to the trigger becomes centered around VCC/2, or 1.65V in this case. The capacitor filters out the high frequency signals centers it just at the DC bias level. The 40MHz square wave feeds into the clock fanout buffer NB3N551 which generates 3 clock outputs that oscillate at 40 MHz between about 0V and 3V.

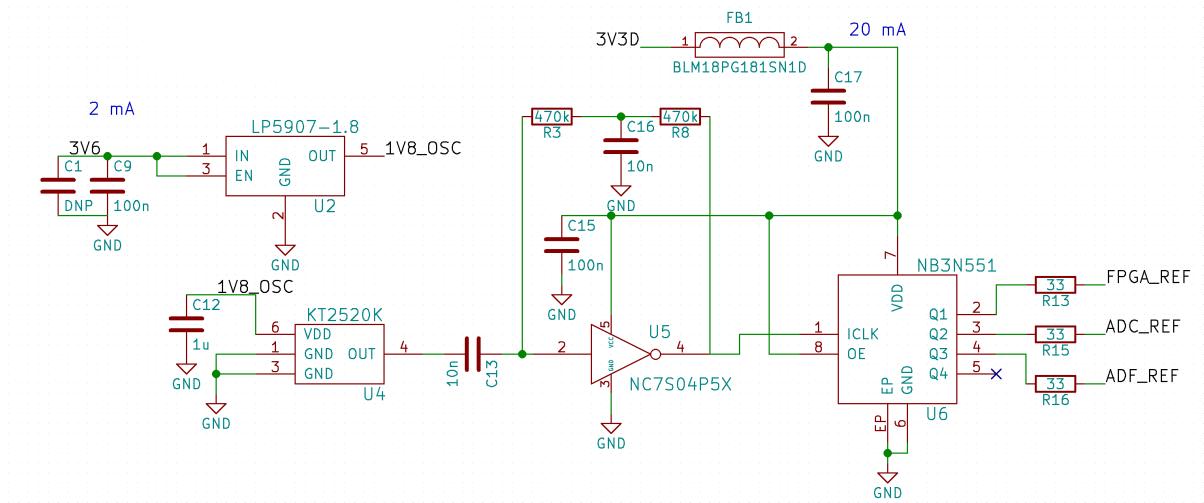


Figure 2.7: A 40MHz clock signal is used to synchronize different components on the board.

2.2.1 KT2520K 40MHz TXCO

2.2.2 NC7S04 Inverter

2.2.3 NB3N551 1:4 Clock Buffer

2.3 TX

schematic

2.3.1 ADF4158 Frequency Synthesizer

Overview

The [ADF4158](#) is a 6.1GHz fractional-n frequency synthesizer. A block diagram describing its functionality is shown in Figure 2.8.

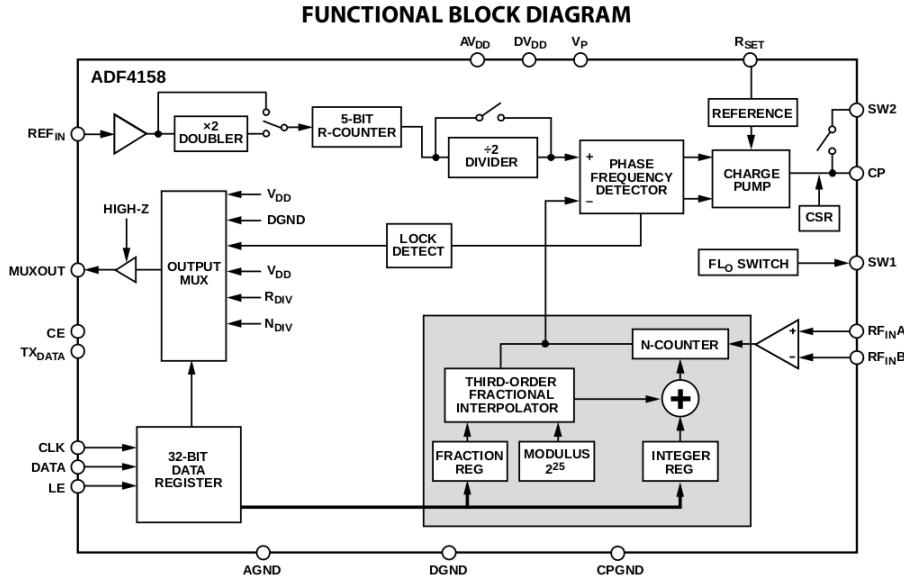


Figure 2.8: ADF4158 block diagram.

The ADF4158 relies on an external VCO, described in § 2.3.2, whose output frequency is given by Equation 2.9. The output resolution is $f_{RES} = f_{PFD}/2^{25}$ (see Equation 2.9 and its corresponding parameter table for an explanation of f_{PFD}). The 2^{25} arises from the FRAC value (set in registers 0 and 1) which is given 25 bits.

$$\begin{aligned} RF_{OUT} &= f_{PFD} \times (INT + (FRAC/2^{25})) \\ f_{PFD} &= REF_{IN} \times [(1 + D) / (R \times (1 + T))] \end{aligned} \quad (2.9)$$

Parameter/Variable	Description
RF _{OUT}	The VCO's output frequency. This is the frequency that's amplified for transmission.
f_{PFD}	The input frequency to the PFD post prescaling. In our case this is 20MHz.
INT	The N counter that has a multiplicative effect on the VCO output frequency.

Parameter/Variable	Description
FRAC	FRAC is the numerator of the fractional number added to INT. This is what distinguishes fractional-n synthesis from integer-n synthesis. It allows greater precision for the VCO output frequency without significantly increasing the prescalers and N counter.
REF _{IN}	The reference input frequency, which in our case is a 40MHz clock signal from the fanout buffer on the top level of the schematic.
D	The doubler bit, which can be 0 or 1. If set to 1, the REF _{IN} frequency is doubled before arriving at the R counter.
R	The input prescaler.
T	The divide- by-2 bit, which can be 0 or 1 and divides the frequency by 2 between the R prescaler and PFD.

Waveform Generation

The ADF4158 is capable of producing several different waveforms. The one used in this design is a sawtooth ramp in frequency as a function of time, shown in Figure 2.9. There are 3 different parameters that determine the shape of a ramp: (1) frequency deviation (the amount the frequency increases at each time step), (2) timeout interval (the amount of time between each time step) and (3) the number of time steps. This is shown diagrammatically in Figure 2.10. The equations governing these parameters are given in Equation 2.10. The number of steps is set directly in register 6. In our configuration $f_{DEV} = 300\text{kHz}$ and Timer = $0.5\mu\text{s}$, which given that the number of steps is equal to 2000 and the starting frequency is 5.3GHz, the sawtooth will ramp from 5.3GHz to 5.9GHz in 1ms. We also use a delay between bursts of 2ms. Equation 2.11 is used to derive this.

$$f_{DEV} = \left(f_{PFD}/2^{25} \right) \times (\text{DEV} \times 2^{\text{DEV_OFFSET}}) \quad (2.10)$$

$$\text{Timer} = \text{CLK}_1 \times \text{CLK}_2 \times (1/f_{PFD})$$

Parameter/Variable	Description
f_{DEV}	The frequency deviation for each frequency jump during ramp.
f_{PFD}	The input frequency to the PFD post prescaling. In our case this is 20MHz.
DEV	A 16-bit value set in register 5.
DEV_OFFSET	A 4-bit word set in register 5.
Timer	The time between each frequency hop.
CLK_1	A 12-bit clock divider set in register 2.
CLK_2	Another 12-bit clock divider set in register 4.

$$\text{Delay} = t_{PFD} \times \text{CLK}_1 \times \text{Delay Start Word} \quad (2.11)$$

Parameter/Variable	Description
Delay	The delay between bursts. This is shown graphically in Figure 2.11.
f_{PFD}	The input frequency to the PFD post prescaling. In our case this is 20MHz.
CLK_1	A 12-bit clock divider set in register 2.
Delay Start Word	A 12-bit value set in register 7.

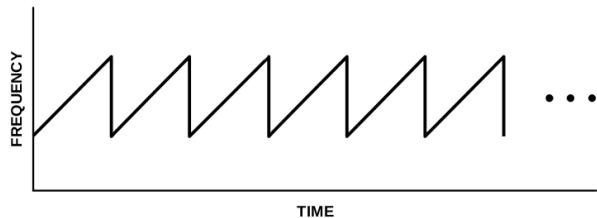


Figure 2.9: Sawtooth ramp.

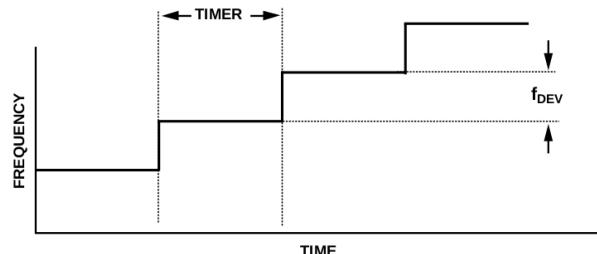


Figure 2.10: Waveform timing.

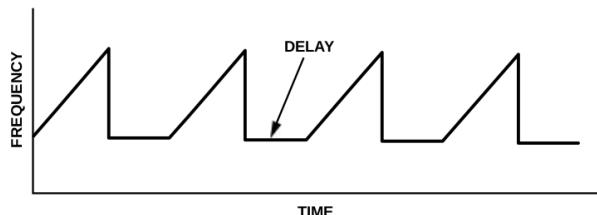


Figure 2.11: Delay between ramps for sawtooth mode.

Configuration Registers

The ADF4158 contains 8 configuration registers. The configurations are shown in the tables below. All reserved bits should be set to 0. The control bits are the least significant 3 bits of each register and are set to the register number for each register (e.g. register 0 is 000 and register 1 is 001). I've left these out of the tables because their values are obvious.

Table 2.10: FRAC/INT REGISTER (R0) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–14	FRAC(MSB)	0	This sets the 12 most significant bits of FRAC. We leave the fractional value at 0.
15–26	INT	265	This is the feedback, or N, counter.
27–30	MUXOUT Control	15	This enables “readback to muxout” which allows interrupting waveform generation and reading back the frequency at the time of interrupt. The functionality is not fully setup on the board(muxout connects to a DNP NMOS) and TX_DATA, which can trigger the interrupt, is set to 0 by the FPGA HDL and left there.
31	Ramp On	1	This bit enables the ramp.

Table 2.11: LSB FRAC REGISTER(R1) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–14	Reserved	0	All reserved bits set to 0.
15–27	FRAC(LSB)	0	This sets the 13 least significant bits of FRAC. We leave the fractional value at 0.
28–31	Reserved	0	All reserved bits set to 0.

Table 2.12: R-DIVIDER REGISTER(R2) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–14	CLK ₁ Divider	10	One of the determinants of the duration of a timestep in waveform generation. See § 2.3.1 for more details.
15–19	R-Counter	1	This 5-bit segment is used to divide the frequency of the reference signal before it enters the PFD. We leave it at 1 and thus do not use it to divide the frequency.
20	Reference Doubler	0	We leave this at 0 and thus do not use the doubler to double the reference signal frequency before input to the PFD. The maximum input frequency for the PFD is 30MHz and so doubling our 20MHz signal (we used the divider to divide the 40MHz signal by 2) would violate this condition.
21	RDIV2	1	Inserts a divide-by-2 toggle flip-flop between the R-counter and PFD. This provides a 50% duty cycle that allows cycle slip reduction to be used which improves lock times.

Table 2.12: R-DIVIDER REGISTER(R2) MAP

Bits	Mnemonic	Value	DESCRIPTION
22	Prescaler	1	The prescaler limits the INT value and through that the maximum frequency to 3GHz. Since we have an INT value of 265 and our maximum frequency is almost double 3GHz we set this to 1.
23	Reserved	0	All reserved bits set to 0.
24–27	Charge Pump Current Setting	0	Sets the current to the minimum value which is 0.31mA and is the level necessary to use cycle slip reduction which we are.
28	CSR Enable	1	Enables cycle slip reduction which provides better lock times.
29–31	Reserved	0	All reserved bits are set to 0.

Table 2.13: FUNCTION REGISTER(R3) MAP

Bits	Mnemonic	Value	DESCRIPTION
3	Counter Reset	0	When this is set to 1, the synthesizer counters are held in reset. For normal operation we set this to 0.
4	Charge Pump Three-State	0	Holds the charge pump in three-state mode if set to 1. For normal operation it must be set to 0.
5	Power-Down	0	Setting this to 1 powers down the device. Setting it to 0 allows normal operation.
6	PD Polarity	1	Set to 1 for positive VCO characteristics. Set to 0 for negative VCO characteristics. Since our VCO outputs a positive voltage signal we set this to 1.
7	LD _P	0	Sets the minimum number of PFD cycles before a lock detect can be set.
8	FSK Enable	0	Disables FSK modulation.
9	PSK Enable	0	Disables PSK modulation.
10-11	Ramp Mode	0	Sets the waveform as a continuous sawtooth.
12-13	Reserved	0	All reserve bits are set to 0.
14	SD Reset	0	Setting this to 0 resets the $\Sigma - \Delta$ modulator on each write to register 0, which is the recommended operation. Setting this to 1 disables resetting the modulator.
15	N SEL	0	When set to 1, this creates an additional delay in setting INT and FRAC which can prevent the PLL overshooting. Since we set INT once and do not update it, this is not necessary and we leave it as 0.
16-31	Reserved	0	All reserve bits are set to 0.

Table 2.14: TEST REGISTER(R4) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–6	Reserved	0	All reserved bits are set to 0.
7–18	CLK ₂ Divider	1	This is used to set the timeout interval in ramp generation. See § 2.3.1 for more information.
19–20	CLK DIV Mode	3	This enables ramp divider mode, which specifies that CLK ₁ and CLK ₂ are used for ramp generation.
21–22	Readback to MUXOUT	3	Confusingly, this has been set to 3, which corresponds to neither of the supported values. Since we don't actually use the MUXOUT, this seems to be fine. If, at some point in the future, we do use the MUXOUT we will probably need to fix this.
23–24	Negative Bleed Current	0	This setting can help improve performance in the dead zone. We've disabled it. Note that this setting and readback to MUXOUT cannot simultaneously be enabled. To understand this setting better refer to AN-1154 Application Note .
25	Reserved	0	All reserved bits are set to 0.
26–30	Σ – Δ modulator mode	0	0 enables this during normal operation. We can set it to 14 when FRAC=0. Even though we've set FRAC to 0, we have left this as 0 which seems strange. It shouldn't cause anything to malfunction, but may cause the ADF4158 to draw more power. We should experiment with setting this to 14 when using the actual board.
31	LE SEL	0	LE is the load enable pin which we use to load data onto the ADF4158's internal registers. Setting this to 0 enables the default operation of using the pin to set LE. Setting it to 1 would synchronize it with the reference signal.

Table 2.15: DEVIATION REGISTER (R5) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–18	Deviation Word	31457	This is used to set the size of successive frequency jumps during ramp. See § 2.3.1 for more information.
19–22	Deviation Offset Word	4	This is also used to set the size of successive frequency jumps during ramp. See § 2.3.1 for more information.
23	Deviation Select	0	Setting this bit to 1 is used for FSK as described on page 28 of the datasheet. Since we do not use FSK we leave this as 0.

Table 2.15: DEVIATION REGISTER (R5) MAP

Bits	Mnemonic	Value	DESCRIPTION
24	Ramp 2 Enable	0	Setting this bit to 1 allows a second ramp with different settings than the first. We only need the first ramp.
25	FSK Ramp Enable	0	Setting this bit to 1 uses FSK. Again, we do not use FSK and so leave this bit as 0.
26–27	Interrupt	0	Sets the type of interrupt used to read the value of INT and FRAC of the ramp at a given moment. We don't read these values and therefore leave this as 0, which corresponds to interrupt off.
28	PAR Ramp	0	Setting this bit to 1 allows a parabolic ramp which we don't use.
29	Tx Ramp CLK	0	Setting this to 0 uses the clock divider (instead of the TX data clock) for clocking the ramp.
30–31	Reserved	0	

Table 2.16: STEP REGISTER (R6) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–22	Step Word	2000	This determines the number of steps in a ramp. To understand this see § 2.3.1 on waveform generation.
23	Step SEL	0	This bit is used when 2 different ramps are needed (for instance with FSK). As we don't need this functionality we leave it off and set this bit to 0.
24–31	Reserved	0	

Table 2.17: DELAY REGISTER (R7) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–14	Delayed Start Word	4000	Sets the ramp start delay. We do not use a start delay, but we do use this to delay between ramps.
15	Delayed Start Enable	0	We do not use a start delay.
16	Delay Clock Select	1	Increases the period of the delay clock by multiplying the period of the PFD clock by CLK ₁ . This creates an effective period of 500ns.
17	Ramp Delay	1	We enable a delay between ramp bursts.
18	Ramp Delay Fast Lock	0	Disables the ramp delay fast lock function.
19–31	Reserved	0	

FIXME I need to finish describing this component along with all other component on this sheet.

Table 2.18: All ADF4158 pin connections.

PIN	Mnemonic	DESCRIPTION
1	CPGND	The charge pump ground, which we set to the same ground as the rest of the circuit.
2, 3	AGND	Prescaler ground; also set to common ground.
4	RF _{IN} B	The complement to the VCO output. We couple this to ground with a small capacitor: in this case, 10pF.
5	RF _{IN} A	The VCO output. This goes through prescaling and then becomes the PLL input signal.
6, 7, 8	AV _{DD}	The positive power supply for the RF part of the PLL. This takes 3.3V.
9	REF _{IN}	The reference input, which is our 40MHz clock signal.
10	DGND	Digital ground. This uses the common ground.
11	SDGND	$\Sigma - \Delta$ modulator ground.
12	TX _{DATA}	This pin can be used when employing FSK or PSK to transmit data. Since we are not using this we connect this pin to the FPGA but just set the value to GND.
13	CE	We must set this pin to high in order to enable the device.
14	CLK	A clock used to synchronize register configuration.
15	DATA	Serial data input to set the configuration registers.
16	LE	This pin is brought high in order to load data into the configuration registers.
17	MUXOUT	This allows internal parameters of the ADF4158 to be read externally. We don't currently use this in our design.
18	SDV _{DD}	Power supply pin for the $\Sigma - \Delta$ modulator. We also set this to 3.3V.
19	DV _{DD}	Power supply pin for the digital circuitry. Also 3.3V.
20, 21	SW1, SW2	FIXME: These pins are used for the fast-locking feature, which we use in our design. The typical topologies for this are shown in Figure 2.12. However, our design does not use either of these topologies.
22	V _P	The charge pump power supply, which we set to 5V.
23	R _{SET}	This pin sets the maximum charge pump current according to the equation $I_{CP\max} = 25.5/R_{SET}$. We use 5.49k Ω for the resistor which gives a charge pump current of 4.6mA.
24	CP	The charge pump output that is amplified and feeds into the VCO. Obviously, because this is a charge pump we need to place a bypass capacitor to ground before the amplifier. Here we use 51pF.
25	EPAD	This is connected to GND.

TLV2172 Operational Amplifier

The [TLV2172](#) op-amp is used to produce a gain of 2 to match the CP output of the ADF4158 to the VTUNE input of the VCO. This is necessary since the charge pump supports a tune voltage of up to 5.5V but the VCO has a tune range of 0V to 10V. A non-inverting amplifier configuration is used to produce the necessary gain.

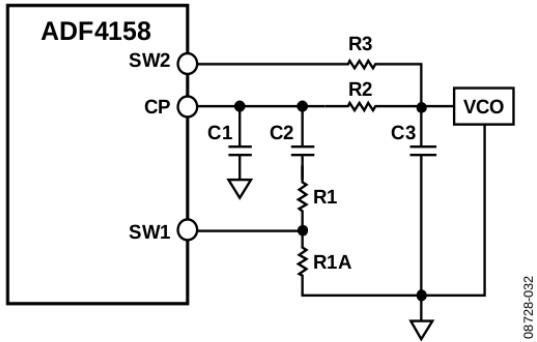


Figure 46. Fast-Lock Loop Filter Topology—Topology 1

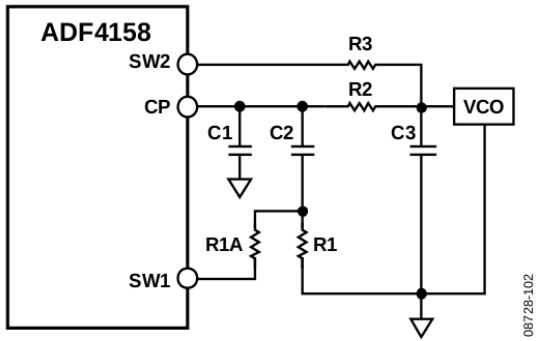


Figure 47. Fast-Lock Loop Filter Topology—Topology 2

Figure 2.12: Fast lock topologies.

2.3.2 HMC431LP4RF VCO

The [HMC431LP4RF](#) is a radio-frequency VCO.

2.3.3 SE2567L Power Amplifier

Fixme Note!

Fixme Note: I've changed this component with the [SE5004L](#), which has higher power output. Everything from this section forward (within the TX section) will need to be changed.

2.3.4 Notes

MGA25203 → SE2567L Substitution

The MGA-25203 power amplifier is now obsolete and there are no alternatives with an identical interface. Therefore the design will need to be modified. The alternative I will use is the [SE2567L](#) which seems to be very similar. There are a few small differences in addition to the pin differences: the original device has a power output of 23dBm in contrast to the new device which has a power output of 21.5dBm when $VCC0,1=3.3V$ and $VCC2,3=4.5V$. I believe this is close enough to not create any issues, especially as they are both quoted as having 30dB gain and are IEEE 802.11-compliant. Additionally, the original device has a load current about twice that of the new device. Since the output powers are relatively similar (and the overview of original design describes a 20dB gain), I don't believe this should be an issue. The original

chip contains BCTRL, BSW, and BSPLY that differ from the new chip. BCTRL regulates the device current and is kept at a constant 2.8V. There is no analogous pin on the new chip. BSW is an enable pin that turns on the device when set to 1.8V. When it is set to 0V, the device is turned off. The original design uses a MOSFET transistor with a drain of 1.8V, source at GND and gate controlled by the FPGA (when the gate is activated the BSW pin gets a voltage of 0V and is disabled). The new chip contains an analogous pin, EN, which enables the power amplifier at the input voltage level 3.3V and disables it at GND. So, to get analogous functionality, we should connect the MOSFET drain to 3.3V instead of 1.8. Everything else should stay the same. BSPLY simply received the input voltage of 3.3V. In the new chip, the pins that are different from the original chip (and not already discussed) are VCC0-3 and DET. DET is a pin that outputs a voltage indicating the power output. We should connect this to a DNP capacitor to GND to be able to detect the output power of the amplifier with an ammeter. VCC0-1 should be connected to the 3.3V power supply. In order to get the additional power output, we should connect VCC2-3 to a 4.5V power supply instead of 3.3V. Unfortunately, it isn't possible to simply repurpose the voltage divider to the BCTRL pin for this since I do not know the internal impedance of the power amplifier IC and in general voltage dividers are bad for supplying power. Additionally, the simple LP5907 voltage divider may not work since it is only rated to 250mA of current, which doesn't leave much room above the 220mA expected by the SE2567L. It seems to make the most sense to use the [TPS7A91](#) which is rated to 1A. It has the downside of requiring additional hardware which will need to be adjusted in the PCB accordingly. To get the correct output voltage of 4.5V, we need to use Equation 2.5. Since the datasheet doesn't specify the resistors for a 4.5V output voltage this takes a bit of art to finding the correct resistors. Ideally, we want resistors with a higher-value so that less power is dissipated, and are commonly used to reduce cost and make them easier to replace if necessary. Lastly, I'd like to keep the values in the ballpark of those specified in the datasheet in case there are any other reasons I'm not thinking of. The values $R_2 = 1.47\text{k}\Omega$ and $R_1 = 6.8\text{k}\Omega$ should work well.

2.4 USB

In order to configure the FPGA we load the configuration bit stream onto the board using a US cable and USB micro receptor on the PCB. The signal is then passed along to an [FT2232H](#) IC that translates the USB signal into JTAG data which it sends to the FPGA using the TCK, TDI, TDO and TMS pins. The schematic for this is shown in Figure [??](#). The FT2232H IC requires several power inputs: 3 VCORE input pins that require a 1.8V input voltage (this comes from VREGOUT, which is a pin that outputs a 1.8V signal from an internal voltage regulator), 1 3.3V VREGIN input (which serves as the input pin to drive the 1.8V regulated output), 4 3.3V VCCIO input pins, and 23.3V inputs (VPLL and VPHY) that are filtered with an LC filter.

FIXME I'm a bit confused as to the operation of the LC filter. It is implemented as a ferrite bead with two parallel capacitors in parallel with the load. This feeds in from an LDO regulator that in turn received its input from a buck converter operating at roughly 700kHz. The frequency at which the ferrite bead is in its resistive state is around 100MHz, well above the switching regulator noise. Maybe the filter is meant to filter out noise other than that from the buck converter. For instance, it is possible that the LDO regulator used after the buck converter (which has a PSRR near 0 around 500kHz and above) is sufficient for filtering the switching noise. The Art of Electronics states that ferrite beads should be placed at different points throughout the design to raise impedance at high frequencies. It is possible that this is the functionality provided by these ferrite beads. OSCI and OSCO are the oscillator input and output, respectively. These must be connected to a 12MHz oscillator with a frequency tolerance less than 30ppm (ours is 10ppm). REF is a current reference that must be connected to a $12\text{k}\Omega$ resistor to ground. DM and DP are the USB data signal minus and plus lines, respectively. TEST should be connected to ground. RESET# (the # indicates an active low pin) is connected to a pull-up resistor to 3.3V so the reset is deasserted whenever the input voltage is at a sufficiently high and stable level. PWREN# is an output that is 0 during normal operation. We don't need it here so we leave it floating. SUSPEND# is similar to PWREN#; it is low when the USB is in suspend mode. This is used as an input to the FPGA. BDBUS0-3 are used for the JTAG interface to configure the FPGA. In order, they are TCK, TDI, TDO and TMS.

2.4.1 FT2232H

FIXME The FT2232H device also communicates with the FPGA using a synchronous FIFO interface as described [here](#). This seems to be the way that data is sent between the FPGA and the host computer although I'm not clear how this works. In any event, it uses the ADBUS0-7 and ACBUS0-7 pins. The ADBUS pins are bidirectional pins that serve as the FPGA side translation of the USB data. In other words, they should allow the FPGA to read USB data from a host computer and send data back to that host computer through a USB cable. The ACBUS lines seem to be used to signal reads/writes/etc. It also seems to require the external EEPROM storage IC, [93LC46B](#) although I'm not sure why this interface requires additional, external memory.

2.4.2 93LC46B 1k EEPROM

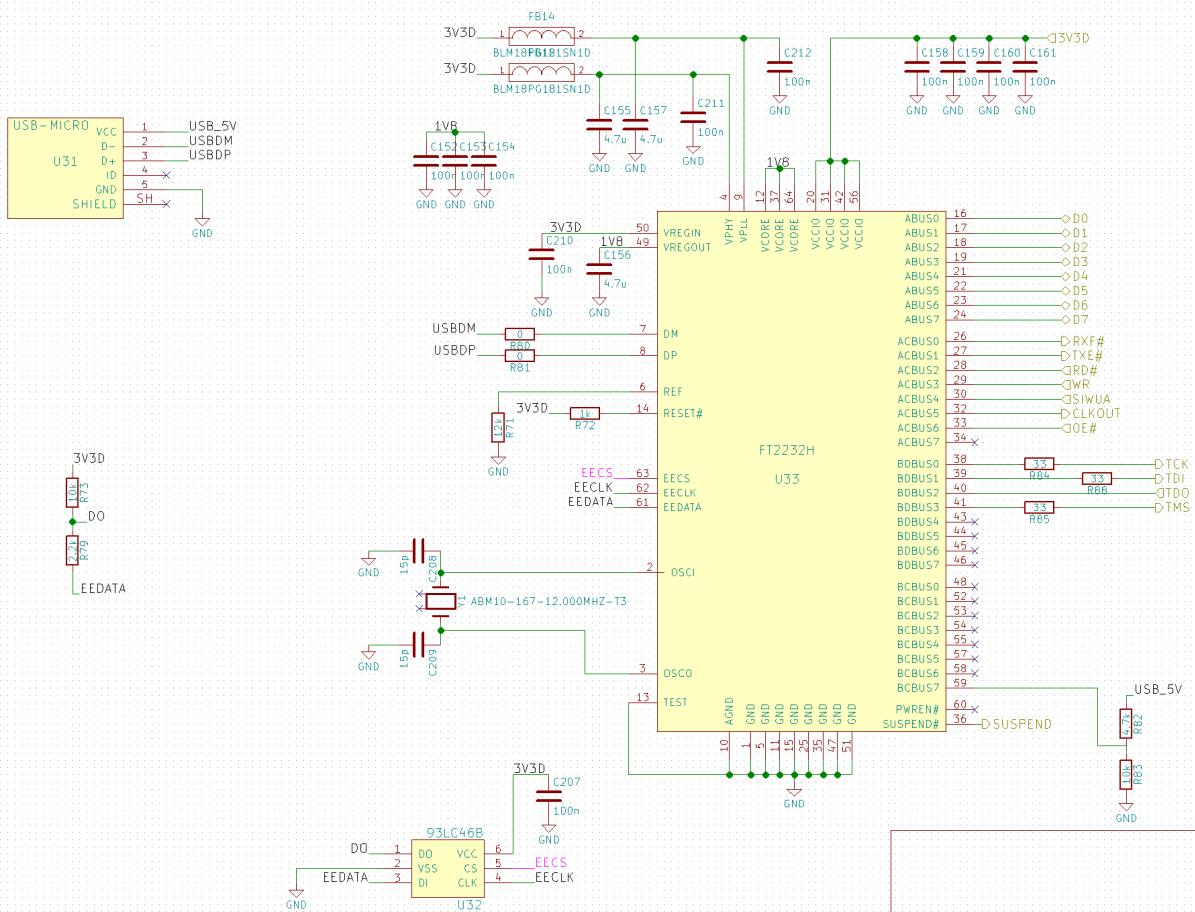


Figure 2.13: The configuration bit stream is loaded onto the FPGA using a USB-JTAG interface.

2.5 FPGA

2.5.1 XC7A15T-FTG256 Xilinx FPGA

There are several different documents for the [Artix-7 FPGA family](#), which is used for this radar. There is a [package-pinout document](#) that, as its name suggests, contains information about the various package options available and the associated pin definitions. There is an [overview sheet](#) that contains very general information about the FPGA. There is a [PCB design guide](#) document that contains recommendations for soldering the FPGA on a PCB. There is a [XADC document](#) which describes the mixed-signal functionality of the FPGA. Finally, there is a [configuration guide](#) that describes the different ways to configure the FPGA with a bitstream. The FPGA pins follow a logical syntax: they can be of the form IO_LXXY_ZZZ_#, IO_XX_ZZZ_#, or a specific name. If they have a specific name, they have a dedicated function described starting on page 26 of the package-pinout document. If they have the general form IO_LXXY_ZZZ_# or IO_XX_ZZZ_#, they can be used for several different dedicated functions, or as general purpose IO pins. L indicates that the pin can be used as a differential pair (differential signaling), where Y=P or N depending on whether the pin is the positive or negative side of the differential pair. XX gives a unique number identifier that can be used to associate the two different pins forming a differential pair. ZZZ represents one or more functions that the pin can be used for in addition to general purpose IO. # indicates the bank number, which separate the pins into one of several different regions. The package used in this design is FTG256C, which contains banks 0, 14, 15, 34 and 35. Bank 0 contains the dedicated configuration pins. Each bank has 4 pairs of clock capable inputs for differential or single ended clocks (there are no global clock pins). The bypass capacitor recommendations are outlined in the PCB design guide. As with any other bypass capacitor design, place the small caps as close to the pins they connect to as possible. The larger caps should also be as close as possible, but give priority to the small caps. The FPGA loads its configuration from an SPI flash memory device, [W25Q32JV](#). The CCLK_0 pin is exported from the FPGA to the flash device to drive its operation and coordinates writes and reads to and from the device. DO is used by the FPGA to perform SPI reads from the flash memory device. It is connected to J14 of the FPGA. DI is connected to J13 of the FPGA and is used by the FPGA to send data to the flash memory device. CS is active low and is used by the FPGA to signal data transmission is about to occur. It is connected to L12 on the FPGA. WP (write protect) and HOLD are both active low pins and are used to prevent the status configuration registers from being written to and can pause the device when multiple devices share the same SPI signal, respectively. They are unused and thus connected to the 3.3V power supply. The device is powered with 3.3V (it supports a range of 2.7V to 3.6V). Table 2.5.1 contains a list of all FPGA pins and their connections.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
A01	GND.
A02	Connected to an external 1x1 pin header that is currently unused by FPGA logic.
A03	Connected to external 2x4 pin header that is currently unused by FPGA logic.
A04	Connected to same pin header as A03. Also unused.
A05	Same.
A06	One of the voltage supplies for bank 35. It uses 3.3V as with all the other voltage supplies to the main banks of the FPGA.
A07	Connected to same pin header as A03. Unused.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
A08	A different 2x4 pin header. Unused.
A09	Connected to the same pin header as A08. Unused.
A10	Floating.
A11	GND.
A12	Connected as an output to the FT2232H USB-JTAG converter. Pulling this low allows the FT2232H device to output data to the FPGA through the ADBUS pins. Driving it high sets the ADBUS pins as inputs. Remember, the ADBUS pins serve as the bidirectional translation between JTAG data and USB data.
A13	SIWU (channel A). This pin is output to the FT2232H device and can be used to optimize USB data transfers (more info in the FT2232H datasheet). It is not currently used by the FPGA logic.
A14	Write output pin to the FT2232H device. When this is driven low, the FPGA writes data to the FT2232H. When it's driven high, the FPGA can perform a read from the FT2232H.
A15	Inputs a 60MHz clock signal that originated at the FT2232H. This clock is used to synchronize all data transfers between the FPGA and the FT2232H.
A16	A 3.3V input to bank 15.
B01	ADC_SHDN1. One of two ADC_SHDN outputs that, together with OEA (B02), controls the shutdown mode selection of the ADC. It is used by the digital FPGA logic. When both SHDN and OE are grounded, the ADC performs normally. When they are both pulled high, the ADC goes into sleep mode. There are two other states but they are not used by the FPGA logic.
B02	ADC_OE1. See pin B01. This is used for channel A.
B03	3.3V input for bank 35.
B04	Floating.
B05	Connected to the same 2x4 pin header as A03. Unused.
B06	Floating.
B07	Connected to the same 2x4 pin header as A03. Unused.
B08	GND.
B09	Connected to the same pin header as A08. Unused.
B10	Connected to the same pin header as A08. Unused.
B11	Floating.
B12	Floating.
B13	3.3V input for bank 15.
B14	RD#. A read active-low output pin to the FT2232H device. When this is driven low, the FPGA reads data from the FT2232H.
B15	TXE#. An active-low input from the FT2232H. FT2232H drives this pin low to signal sending data to the FPGA (i.e. a read for the FPGA).
B16	RXF#. An active-low input from the FT2232H. FT2232H drives this pin low to signal that it should read data. To the FPGA, this signals that it should write data.
C01	Floating.
C02	OF1. An input from the ADC. It is pulled high when an overflow or underflow occurs at channel A. It is currently unused by the FPGA logic.
C03	Floating.
C04	Floating.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
C05	GND.
C06	Floating.
C07	Floating.
C08	Floating.
C09	Floating.
C10	3.3V input to bank 15.
C11	Connected to the same pin header as A08. Unused.
C12	Connected to the same pin header as A08. Unused.
C13	Floating.
C14	Floating.
C15	GND.
C16	FT_SUSPEND. An active-low input to the FPGA. FT223H drives this low when the USB is in suspend mode. It is currently unused by the FPGA logic.
D01	LED. Connects to an LED that is used by the FPGA to signal data processing.
D02	GND.
D03	Floating.
D04	Floating.
D05	Floating.
D06	Floating.
D07	3.3V input to bank 35.
D08	Floating.
D09	Floating.
D10	Floating.
D11	Floating.
D12	GND.
D13	Floating.
D14	Floating.
D15	FT_D7. Bidirectional pin 7/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
D16	FT_D6. Bidirectional pin 6/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
E01	D10. Input pin 10/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
E02	D11. Input pin 11/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
E03	Floating.
E04	3.3V input to bank 35.
E05	Floating.
E06	Floating.
E07	CFGPVS_0. A dedicated pin that is part of the configuration logic of the FPGA. It is used to specify the voltage level used for all banks. Since we use 3.3V, we connect this pin to the same 3.3V voltage level.
E08	CCLK_0. An output pin exported from the FPGA to the flash memory device, W25Q32JV, to drive its operation and coordinate writes and reads to and from the device.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
E09	GND.
E10	VCCBRAM. One of the two power supply pins to the FPGA's internal RAM. It requires 1.0V.
E11	Floating.
E12	Floating.
E13	Floating.
E14	3.3V power supply to bank 15.
E15	FT_D5. Bidirectional pin 5/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
E16	FT_D4. Bidirectional pin 4/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
F01	VCC0_35. 3.3V power supply to bank 35.
F02	D9. Input pin 9/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
F03	Floating.
F04	Floating.
F05	Floating.
F06	GND.
F07	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
F08	VCCBATT_0. Can be used for memory backup. We do not use it so we tie it to GND.
F09	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
F10	GND.
F11	VCCBRAM. One of the two power supply pins to the FPGA's internal RAM. It requires 1.0V.
F12	Floating.
F13	Floating.
F14	FT_D3. Bidirectional pin 3/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
F15	FT_D0. Bidirectional pin 0/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
F16	GND.
G01	D10. Input pin 10/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
G02	D7. Input pin 7/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
G03	GND.
G04	Floating.
G05	Floating.
G06	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
G07	GNDADC_0. The reference voltage for the onchip ADC.
G08	VCCADC_0. A 1.8V power supply used to power the onchip ADC.
G09	GND.
G10	VCCAUX. A 1.8V power supply for auxiliary circuits in the FPGA IC.
G11	Floating.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
G12	Floating.
G13	GND.
G14	Floating.
G15	FT_D2. Bidirectional pin 2/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
G16	FT_D1. Bidirectional pin 1/7 of ADBUS, used to transmit data between the FPGA and FT2232H.
H01	D4. Input pin 4/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
H02	D5. Input pin 5/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
H03	D6. Input pin 6/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
H04	Floating.
H05	Floating.
H06	GND.
H07	VREFN_0. A dedicated pin that is used as a 1.25V reference GND voltage. Tied to GND.
H08	VP_0. A dedicated pin that is used as the XADC differential analog input (positive side). It is left floating and is unused.
H09	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
H10	DONE_0. A bidirectional dedicated pin. It is pulled high when configuration is done. It is connected to a pull-up resistor and an external connector, presumably for debugging purposes.
H11	Floating.
H12	Floating.
H13	Floating.
H14	Floating.
H15	VCC0_15. 3.3V power supply for bank 15.
H16	CARD_DETECT. Connects to the SD card. Currently unused by FPGA logic.
J01	Floating.
J02	VCC0_35. 3.3V power supply for bank 35.
J03	D3. Input pin 3/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
J04	MIX_ENBL. An enable pin that is output to the ADL5802 mixer. It is pulled low to enable the mixer and pulled high to disable it.
J05	Floating.
J06	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
J07	VN_0. A dedicated pin that is used as the XADC differential analog input (negative side). It is tied to GND and is unused.
J08	VREFP_0. A dedicated pin that is used as a 1.25V reference input. It is tied to GND and unused.
J09	GND.
J10	VCCAUX. A 1.8V power supply for auxiliary circuits in the FPGA IC.
J11	GND.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
J12	VCC0_15. 3.3V power supply for bank 15.
J13	SPI_MOSI. Used by the FPGA to send data to the flash memory device.
J14	SPI_DIN. Used by the FPGA to read data from the flash memory device.
J15	Floating.
J16	Floating.
K01	D2. Input pin 2/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
K02	D1. Input pin 1/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
K03	ADF_MUXOUT. Input from the frequency synthesizer. It indicates that a sweep is done.
K04	GND.
K05	Floating.
K06	GND.
K07	DXN_0. The cathode of two temperature-monitoring diode pins. It is not used and is therefore tied to GND.
K08	DXP_0. The anode of two temperature-monitoring diode pins. It is not used and is therefore tied to GND.
K09	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
K10	INIT_B. Indicates initialization of configuration memory. It is pulled high and is unused.
K11	VCCAUX. A 1.8V power supply for auxiliary circuits in the FPGA IC.
K12	Floating.
K13	Floating.
K14	GND.
K15	Floating.
K16	Floating.
L01	GND.
L02	D0. Input pin 0/11 of the ADC's channel A digital outputs. It is used to transmit data from the ADC to FPGA.
L03	Floating.
L04	Floating.
L05	Floating.
L06	VCC0_0. 3.3V power supply for bank 0 (i.e. the bank for dedicated configuration pins).
L07	TCK. An input pin originating at the output of FT2232H and is used to transmit the JTAG clock. It is not used by the FPGA logic.
L08	VCCINT. 1.0V power supply for the internal core logic of the FPGA.
L09	PROGRAM_B. Connected to a pushbutton switch and can be used to perform an asynchronous reset of the configuration logic.
L10	VCCAUX. A 1.8V power supply for auxiliary circuits in the FPGA IC.
L11	GND.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
L12	SPI_CS. An output pin that can be brought low to indicate that a transmission will take place between the FPGA and flash storage device. It is currently left in the high-impedance state in FPGA logic, seemingly indicating that the flash storage device is not yet used.
L13	Floating.
L14	Floating.
L15	PUDC_B. Pulled low, which configures all I/O pins to enable their internal pull-up resistors.
L16	VCC0_14. 3.3V power supply to bank 14.
M01	OF2. An input from the ADC. It is pulled high when an overflow or underflow occurs at channel B. It is currently unused by the FPGA logic.
M02	ADF_DATA. A serial data output pin the frequency synthesizer.
M03	VCC0_34. 3.3V power supply to bank 34.
M04	Floating.
M05	Floating.
M06	Floating.
M07	TMS. Output pin that is fed into the FT223H as a mode select pin. It is unused by the current FPGA logic.
M08	GND.
M09	M0. Along with M1 and M2, this specifies the configuration mode of the FPGA. M[2:0] = 001 which indicates the FPGA acts as a master in an SPI interface. Because a DNP resistor is placed between M0 and 3.3V, it is not actually connected to the power supply. However, since all pull-up resistors should be enabled by pulling PUDC_B low, it should register a logic 1.
M10	M1. See M09.
M11	M2. See M09.
M12	Floating.
M13	VCC0_14. 3.3V power supply to bank 14.
M14	Floating.
M15	Floating.
M16	SD_DAT1. A data line to the SD card reader. Not currently used by the FPGA logic.
N01	ADF_LE. An output that connects to the ADF4158 frequency synthesizer. When it is pulled high, data stored in the ADF4158 shift registers is loaded into one of the 8 latches.
N02	ADC_SHDN2. See B01.
N03	Floating.
N04	Floating.
N05	GND.
N06	Floating.
N07	TDI. JTAG data input from FT2232H. It is not currently used by the FPGA logic.
N08	TDO. JTAG data output to FT2232H. It is not currently used by the FPGA logic.
N09	Floating.
N10	VCC0_14. 3.3V power supply for bank 14.
N11	CLK_REF. An input pin that takes the main 40MHz reference clock used by the FPGA. It is one of the outputs of the clock fanout buffer.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
N12	Floating.
N13	Floating.
N14	SD_CLK. A clock that synchronizes activity with the SD card reader. It is not currently used by the FPGA logic.
N15	GND.
N16	SD_DAT0. A data line to the SD card reader. Not currently used by the FPGA logic.
P01	ADC_OE2. See pin B01. This is used for channel B.
P02	GND.
P03	ADF_TXDATA. Output pin that transmits data to be used by the ADF4158 frequency synthesizer for FSK or PSK transmission. This is unused by the FPGA logic.
P04	Floating.
P05	Floating.
P06	Floating.
P07	VCC0_14. 3.3V power supply for bank 14.
P08	Floating.
P09	Floating.
P10	Floating.
P11	Floating.
P12	GND.
P13	Floating.
P14	Floating.
P15	Floating.
P16	SD_CMD. A pin used to communicate with the SD card reader. Currently unused by the FPGA logic.
R01	ADF_CLK. An output clock used to synchronize the operation of the ADF4158 frequency synthesizer.
R02	Floating.
R03	ADF_CE. An output pin to the ADF4158. When this pin is driven low, it powers down the frequency synthesizer.
R04	VCC0_34. 3.3V power supply for bank 34.
R05	Floating.
R06	Floating.
R07	Floating.
R08	Floating.
R09	GND.
R10	Floating.
R11	Floating.
R12	Floating.
R13	Floating.
R14	VCC0_14. 3.3V power supply for bank 14.
R15	SD_DAT3. A data line to the SD card reader. Not currently used by the FPGA logic.
R16	SD_DAT2. A data line to the SD card reader. Not currently used by the FPGA logic.
T01	VCC0_34. 3.3V power supply for bank 34.

Table 2.19: All FPGA pin connections in alphabetical order.

PIN	DESCRIPTION
T02	PA_OFF. Output pin that connects to the base of a transistor and can be used to enable (high) or disable (low) the operation of the power amplifier, SE2567L.
T03	Floating.
T04	ADF_DONE. Input pin that is unused by the FPGA logic.
T05	Floating.
T06	GND.
T07	Floating.
T08	Floating.
T09	Floating.
T10	Floating.
T11	VCC0_14. 3.3V power supply for bank 14.
T12	Floating.
T13	Floating.
T14	Floating.
T15	Floating.
T16	GND.

PROGRAM_B_0 is connected to a pushbutton switch and can be used to perform an asynchronous reset to the configuration logic. TDI, TDO, TMS, and TCK are used for the JTAG clock, data input, data output, and mode select, respectively. They are connected to the FT2232H IC that translates between USB signals and JTAG signals. They are used to configure the FPGA (which is also stored in the flash memory). M[2:0] specify the configuration mode for the FPGA. In our configuration, M0 is pulled high while the other 2 are pulled low, indicating a Master SPI interface. CFGBVS_0 is used to specify the voltage level used for all banks. Since we use 3.3V, we connect this pin to the same 3.3V. The PUDC_B pin is pulled low, which configures all I/O pins to enable their internal pull-up resistors. **{STARTINCOMPLETE}** The FPGA seems to have a built-in ADC. However, it doesn't appear to be used **{END INCOMPLETE}**. There is an SD card reader connected up to the FPGA, however, it is unused by the current FPGA code. VCCAUX is used for auxiliary circuits and must be 1.8V. VC-CADC_0 is also 1.8V and is used to power the onchip ADC. VCCINT powers the internal core logic of the FPGA and must be 1.0V. VCCBRAM is the power supply for the FPGA internal RAM, which requires 1.0V.

2.6 ADC

The LTC2292 is a 40MHz, 12-bit differential input ADC. The schematic for this device is shown in Figure 2.14. We use this device instead of the built-in FPGA ADCs because those sample at a rate of 1Msps, which is insufficient for our needs. The difference signal received by the ADC and passed to the FPGA has a frequency in the range of kHz to a few MHz, for which a 40MHz sampling rate is significantly more than the Nyquist frequency. The received signal is received on two antennas, which allows for the angle of the received signal to be computed. These are amplified, mixed and then amplified again before feeding into the ADC. By connecting the CLKA, CLKB, and MUX pins together, we multiplex both channels together through the same output pins, D01-D11A. The timing for this multiplexed output is shown in Figure 2.15. Table 2.6 contains a list of all the ADC pins and their connections.

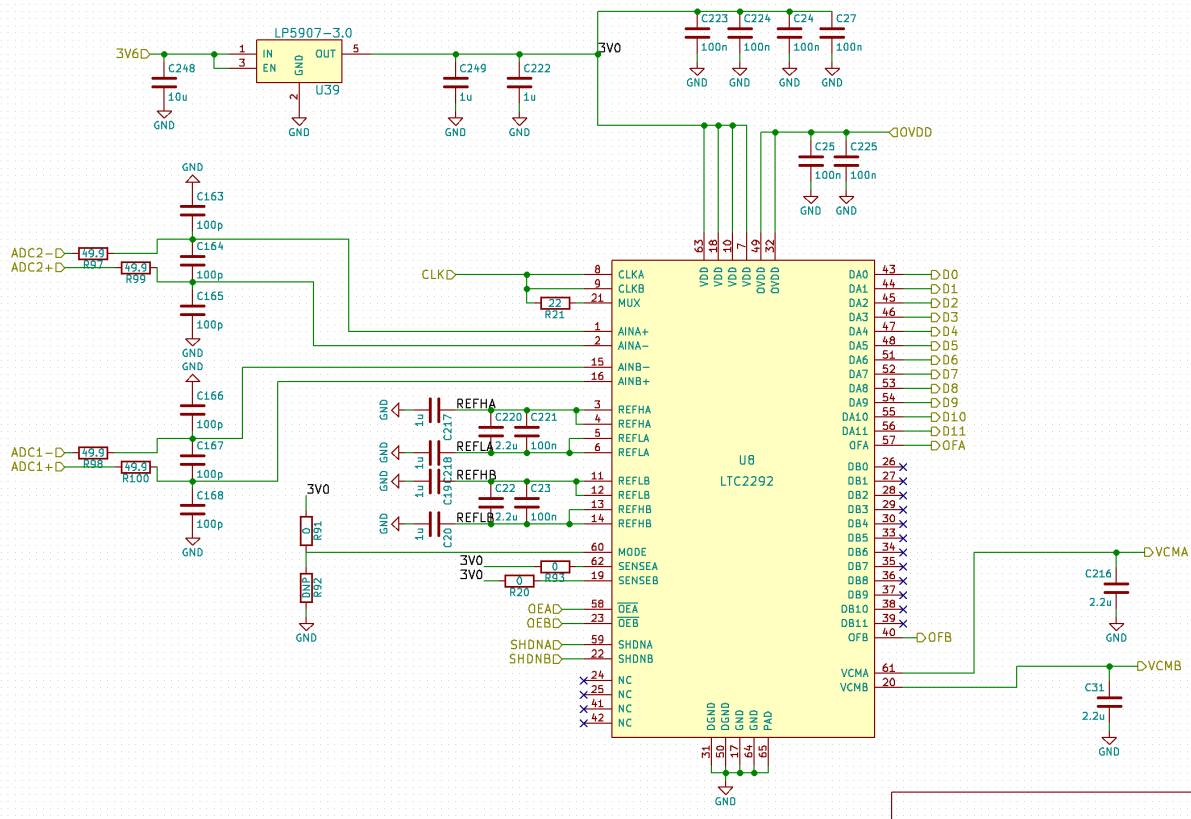


Figure 2.14: The LTC2292ADC schematic.

Table 2.20: All LTC2292 ADC pin connections in logical groupings.

LABELS	PIN #s	DESCRIPTION
VDD	7, 10, 18, 63	The power supply required is 3.0V and each pin requires its own $0.1\mu F$ capacitor. When designing the PCB, ensure that each capacitor is placed as close to its corresponding pin as possible to minimize the trace inductance. The 3.0V signal itself is generated by using an LP5907 LDO regulator that takes in a 3.6V signal.

Table 2.20: All LTC2292 ADC pin connections in logical groupings.

LABELS	PIN #s	DESCRIPTION
OVDD	39, 42	The power supply for the output that the ADC feeds to, which is the FPGA and takes a 3.3V power supply. So, we connect this to the 3.3V power rail and bypass it with two $0.1\mu\text{F}$ capacitors, one for each pin. Again, each should be placed adjacent to their corresponding pins (39 and 42).
CLKA, CLKB, MUX	8, 9, 21	Feeds a 40MHz clock signal to the device and specifies that the digitized channel A and B data should be multiplexed and pass through both output buses A and B. We leave B unconnected, so only bus A matters.
DA0-DA11	43-48, 51-56	The digitized output data that contains the input data from both channels A and B, multiplexed. This is fed into the FPGA for processing.
OFA	57	This pin is driven high when overflow or underflow occurs. Otherwise it is kept low. We export this to the FPGA, although it is not currently used by the FPGA logic.
DB0-DB11	26-30, 33-39	The channel B data bus. Since we multiplex everything through the channel A data bus, this data bus is redundant and therefore left unconnected.
OFB	40	Similar to OFA, but for channel B. This is exported to the FPGA, but is also unused.
VCMA, VCMB	61, 20	A 1.5V signal that is used to set the common-mode voltage of the IF differential amplifiers, whose outputs are sent to this ADC. They are each bypassed to GND with a $2.2\mu\text{F}$ capacitor, which should be placed directly next to their respective pins.
GND, OGND	17, 64, 65, 31, 50	The ADC power ground and output power ground, respectively. These can all be routed to the same ground plane.
NC	24-25, 41-42	No connect.
SHDNA, SHDNB, OEA, OEB	59, 22, 58, 23	These are input pins that are connected to the FPGA. The FPGA logic can ground both SHDNA and OEA to allow channel A to operate normally or bring them both high to put channel A in sleep mode. Channel B works the same way.
SENSEA, SENSEB	62, 19	These are connected to VDD, which specifies that the input voltage range of the differential signals for both channels A and B is $1.5\text{V} \pm 1\text{V}$. 1.5V is the common-mode voltage and the channels allow a 2V range around that.
MODE	60	Connecting mode to VDD specifies the output format as 2s complement and turns off the clock duty stabilizer, which is unnecessary because the input clock has a 50% duty cycle.

Table 2.20: All LTC2292 ADC pin connections in logical groupings.

LABELS	PIN #s	DESCRIPTION
REFHA, REFLA, REFLB, REFHB	3-6, 11-14	These are the high and low reference for channels A and B, respectively. Their connection is specified exactly by the datasheet. It is critical that the $0.1\mu\text{F}$ capacitor is placed as close to the pins as possible.
AINA+, AINA-, AINB-, AINB+	1-2, 15-16	The positive and negative differential inputs for channels A and B. {STARTINCOMPLETE} These lines have a capacitor between the positive and negative analog inputs, as well as capacitors connecting each line to GND. The capacitor value between the lines of $0.1\mu\text{F}$ is different from the suggested value of 12pF and the capacitors connecting the lines to GND are not suggested from the datasheet (see page 18 of the datasheet). Additionally, the resistance value of 49.9Ω differs from the suggested resistance of 25Ω . Lastly, the negative output of the differential amplifier for channel A is feeding into the positive input line {END INCOMPLETE} .

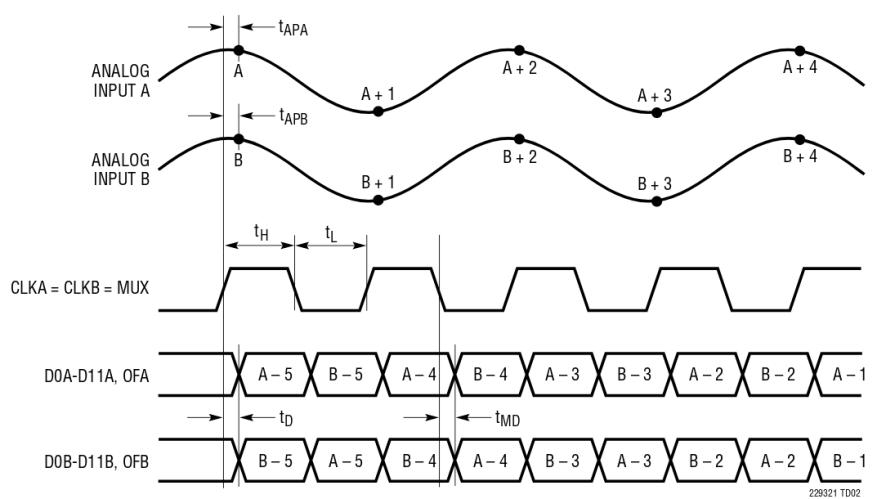


Figure 2.15: Multiplexed digital output bus timing for the LTC2292 ADC.

2.7 Mixer

2.7.1 ADL5802

The mixer schematic shown in Figure 2.16 is composed of three high-frequency baluns that convert two rx and one local oscillator (the original transmitted signal) single-ended signal into differential signals and then feed them into the mixer. The baluns are **5400BL15B050E** and the mixer is **ADL5802**. The mixer outputs the difference of the local oscillator frequency and each R input which ranges from hundreds of kHz to a few MHz.

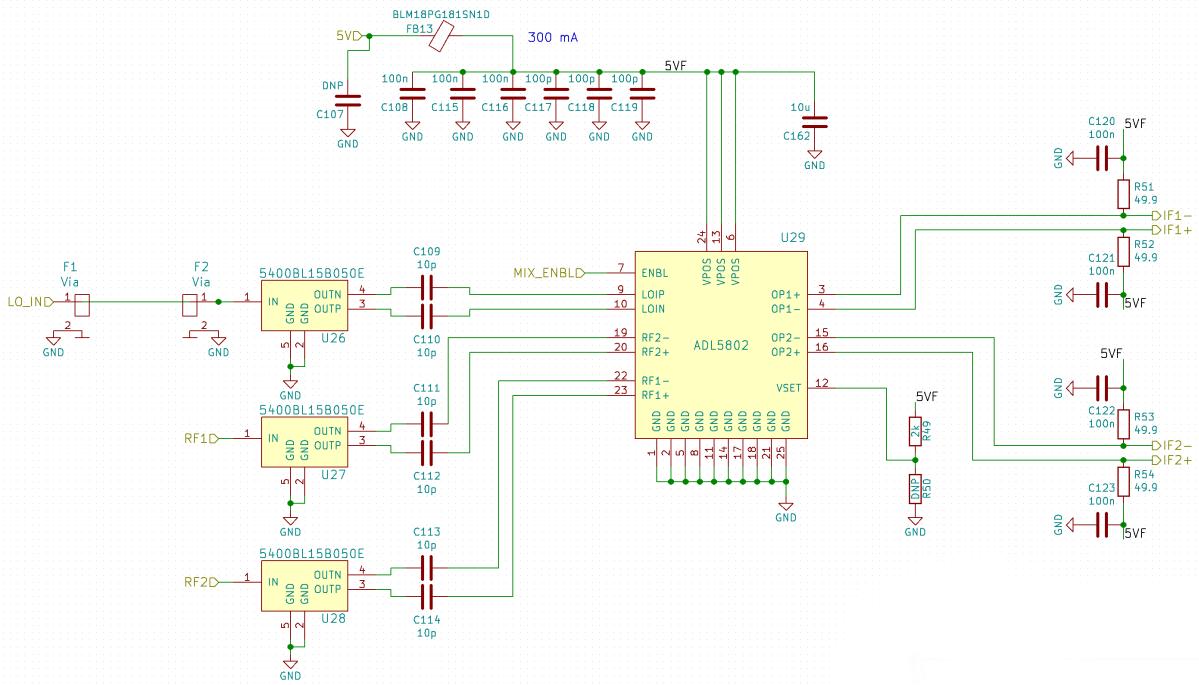


Figure 2.16: The mixer schematic.

The RF and LO input interfaces are designed for a differential input impedance of 50Ω . This is already the differential balanced impedance of the baluns, so we do not need to perform any additional impedance matching at the inputs. The inputs require AC coupling and the datasheet recommends 3pF coupling capacitors placed between the balun outputs and pin inputs. It also shows that the positive balun output of the local oscillator should be hooked up to the positive pin input.

2.8 IF

The IF amplifier, [ADA4940-2](#), receives the output of the mixer, whose frequency is in the range of hundreds of kHz to a few MHz.

2.9 RX1 & RX2

The sheets RX1 and RX2 are identical. They each consist of two amplifiers connected in series, shown in Figure 2.17. The first amplifier is a SKY65404 LNA which operates in the 4.9GHz-5.9GHz range, has a gain of 13dB, a NF of 1dB and a P1dB of -4dBm. It is hooked up exactly as specified in the datasheet, with the addition of a ferrite bead filtering its power supply.

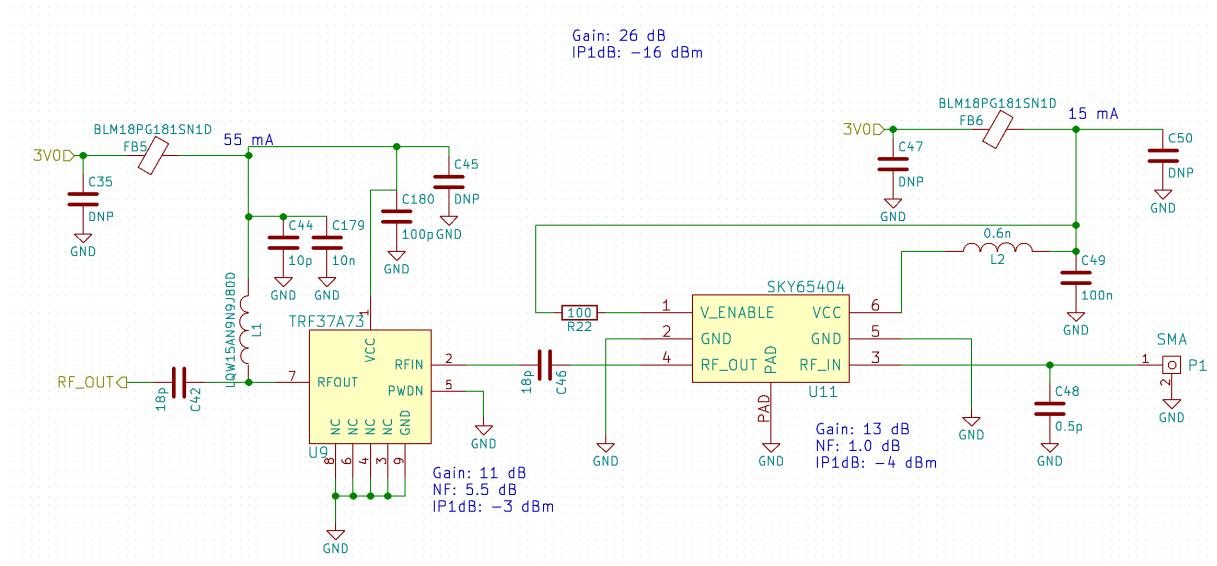


Figure 2.17: One of two RX schematic sheets. Both sheets are identical and consist of two amplifiers connected in series.

The second amplifier is a TRF37A73 RF gain amplifier. It is wired up as recommended in the datasheet, with the addition of a ferrite bead to filter the power supply. Values for the various capacitors and inductor are left out of the datasheet. A 100pF capacitor is used to short high frequency noise to the power supply, which is typical of microwave devices. **{START INCOMPLETE}** I am not sure how the values for the RF bypass capacitors, the inductor, and the DC blocking capacitors was chosen. Additionally, I'm not sure how he arrives at a gain of 26dB, since I've read that gains in series should be additive (and hence 24dB), and I'm not sure where the IP1dB value of -16dBm comes from **{ENDINCOMPLETE}**.

Chapter 3

PCB

3.1 General Layout

The radar is constructed using a 4-layer board and uses both surfaces of the board. The front layer contains nearly all of the board's components while the back layer contains a few surface mount resistors and capacitors. The top layer primarily contains signal traces for top layer SMD components. In particular, it connects many of the signal traces of the ADC to the FPGA, which is placed next to it. Note, however, that most of the signal traces extending from the FPGA only start on the top layer but travel through layer 3. The buck converters and subsequent linear regulators are placed near the top of the board, not necessarily near the components they drive. In addition to the large number of signal traces, the top copper layer contains a large ground plane around the periphery of the board. The 2nd layer is entirely a ground plane. The 3rd layer is largely a ground plane, although it also contains a significant number of traces extending from the FPGA as well as a few other components. The 4th layer primarily contains large power planes for each of the different voltages driving logic in the design. It is worth noting that even though it contains significant power planes, it still has a ground plane that is the largest copper fill zone in this layer. The PCB also has 3 large corner mounting vias with smaller vias placed in a circle in the large via's annular ring. The reason for the small vias is to ensure a continued connection to GND (the mounting vias are connected to GND) even if a screw thread strips too much copper from the main via. Additionally, it helps prevent the PCB from being crushed if too much torque is used to tighten the screw. The 4th corner is occupied by the connections to the DC barrel jack. Power rail traces are made 0.5mm in width whereas signal traces are 0.2mm in width. Grounded vias are placed liberally throughout the design. They have a diameter of 0.46mm and a drill hole size of 0.254mm. I've included several screenshots of the PCB and highlighted important components (Figures 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12).

3.2 Component Layout Considerations

3.2.1 Power Input

The barrel jack used has an inner diameter of 2.1mm and an outer diameter of 5.5mm and should output 12V. Since barrel jack dimensions and voltage output vary widely, special attention should be given to make sure the actual hook up matches these specifications.

The pi filter should be placed as close to the barrel jack connection as possible so as to minimize crosstalk noise with the rest of the board.

3.2.2 LTC2292 ADC

- The LTC2292 contains 4 positive voltage supply pins each requiring a $0.1\mu\text{F}$ bypass capacitor. One capacitor each should be placed directly adjacent to each of the 4 voltage supply pins.
- The device contains 2 input pins for the voltage of the device where the output data is sent. These, similarly require a $0.1\mu\text{F}$ capacitor each. As before they should be placed next to their respective pins, not next to one another.
- VCMA and VCMB each are connected to a $2.2\mu\text{F}$ capacitor to GND. They should be placed as close to the pins as possible.
- REFHA and REFLA (and REFHB and REFLB) have a $0.1\mu\text{F}$ capacitor connected between them. These are the most critical capacitors connected to the ADC. They must be placed 1.5mm away at most, and preferably closer to the pins.

3.2.3 RX1 / RX2

- The 100pF capacitor used to bypass TRF37A73 should be place as close as possible to VCC.

3.3 RF Impedance Matching

Fixme Fatal: I don't think the text below is correct..

I'm keeping it for the value of knowing the trace widths used by the original design. However, the idea that the trace widths are unimportant seems dubious. The PCB calculator built into KiCad should be used for this. Additionally, it is probably worth looking at the following guides: [Maxim: General Layout Guidelines for RF and Mixed-Signal PCBs](#), [Hackaday: Michael Ossmann makes you an RF design hero](#), [Analog](#). I would also look at [this tool](#) for creating a Wilkinson power divider. Also, the trace width for 50Ω resistors seems to have been more like 15mil.

Use [this Mantaro impedance calculator](#) to calculate trace widths. The proper width should be 11.98mil.

The right side of the board houses the RF circuitry (i.e. transmitter, receivers and SMA connectors). The signals are carried to the antennas (a patch-fed horn for transmission and a patch array for reception) via a 50Ω coaxial cable. All RF inputs and outputs for components should match this 50Ω impedance. The original layout does not seem overly concerned with the microstrip transmission line width between RF components. They are kept short and generally have a trace width of about 0.3mm. However, many of them are not even remotely straight and this doesn't seem to be an issue. Where possible the microstrips should be kept short, straight and with an unbroken ground plane beneath them. That seems to be all that is necessary for impedance matching. The patch antennas however will need to be appropriately hooked up in order to match the 50Ω impedance. This should be a relatively straightforward calculation. The

setup Henrik uses seems to be the most logical one. He uses a single patch-fed horn antenna for transmission and a patch array for reception. The horn antenna is made of thin copper. More sophisticated horn antennas require the ability to weld and probably modeling software (e.g. CST).

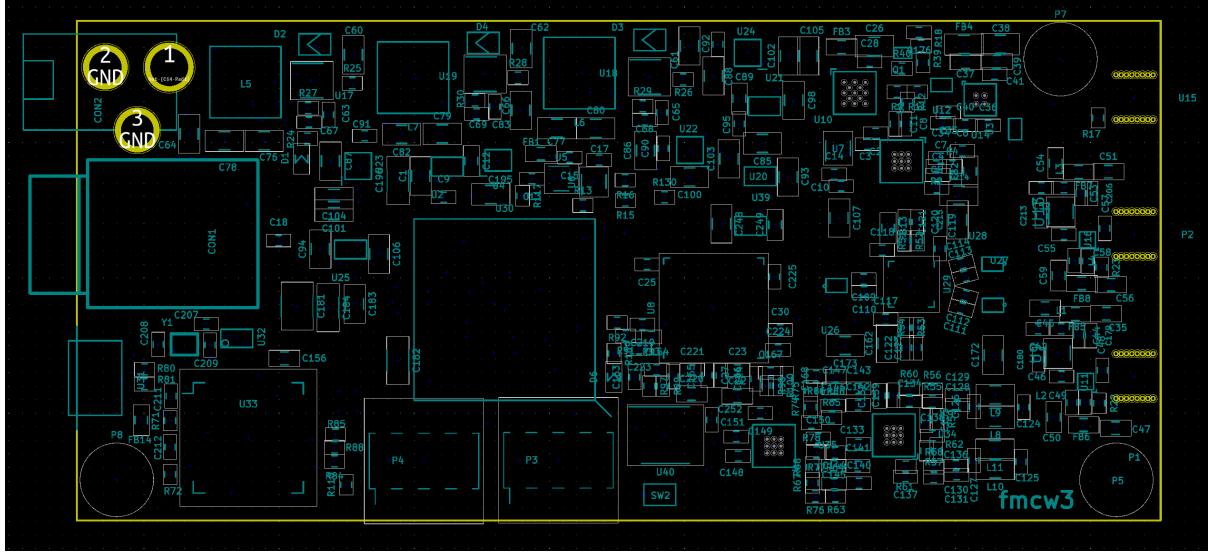


Figure 3.1: Board layout. The top left of the board is where the power source (barrel jack) is connected. The output of the 12V power source is connected to a power plane at the top left-middle of the board, that feeds into the buck converters. Large inductors connected to the buck converters are placed adjacent to their corresponding buck converters. The outputs of the buck converters feed into linear regulators that are mostly placed directly below the buck converters. This is also the region where the main crystal oscillator is placed and its associated fan-out buffer. Transmission circuitry (the frequency synthesizer and some RF amplifiers) are placed at the upper right side of the board near the antenna connectors which are placed on the right side of the board. Circuitry for signal reception are placed along the right side, adjacent to the antenna connectors. The mixer is placed slight inward from here, vertically centered but toward the right side of the board. Below this are intermediate frequency op-amps that feed the mixed signal into an ADC located just above it and to the left (U8). Located just to the left of the IF amplifiers and below the ADC is a flash memory IC (U40) which feeds data to the FPGA (U30), located just to the left of the ADC. Connectors below the FPGA (P3 and P4) can be used to externally monitor the FPGA. In the bottom left of the board is a component to convert USB data to UART data to configure the FPGA as well as a micro USB connection to connect to a host computer. On the left side of the board between the USB connection and barrel jack is a SD card reader that stores data that can be read back out the FPGA.

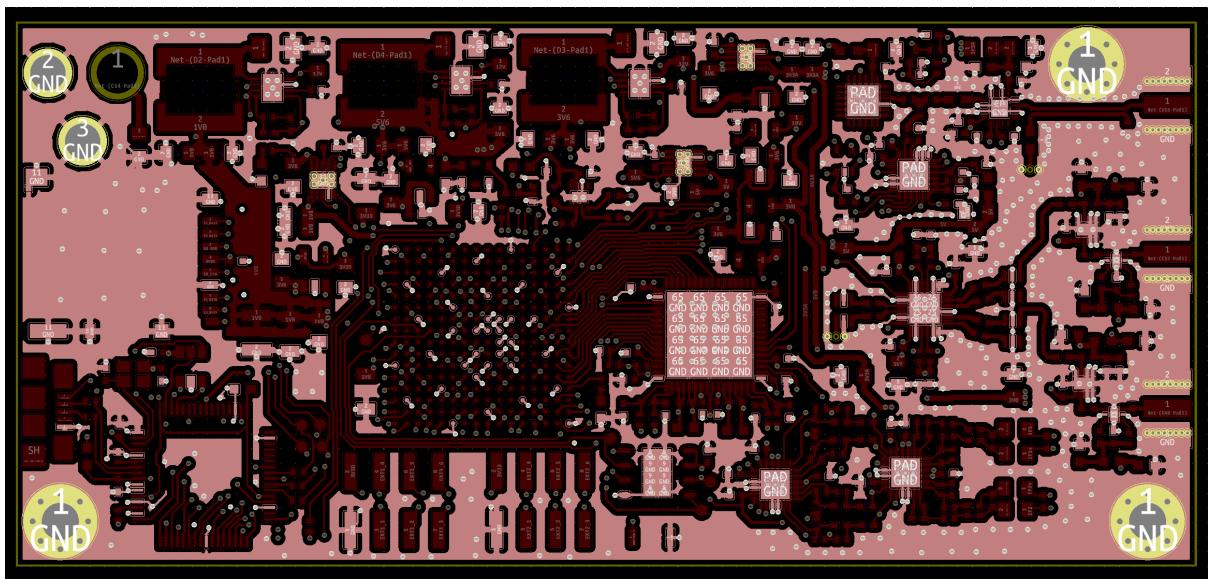


Figure 3.2: A significant portion of layer 1 is a ground plane and the other large part of it is signal traces connecting components. There is also a small 1V power plane toward the upper left not highlighted here.

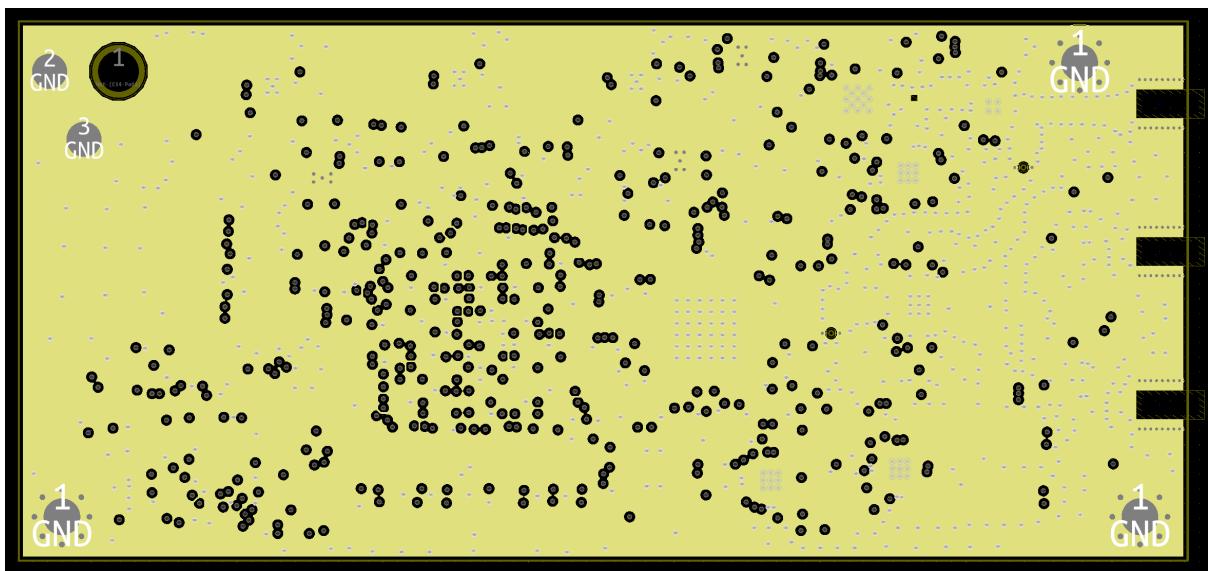


Figure 3.3: All of layer 2 is a ground plane.

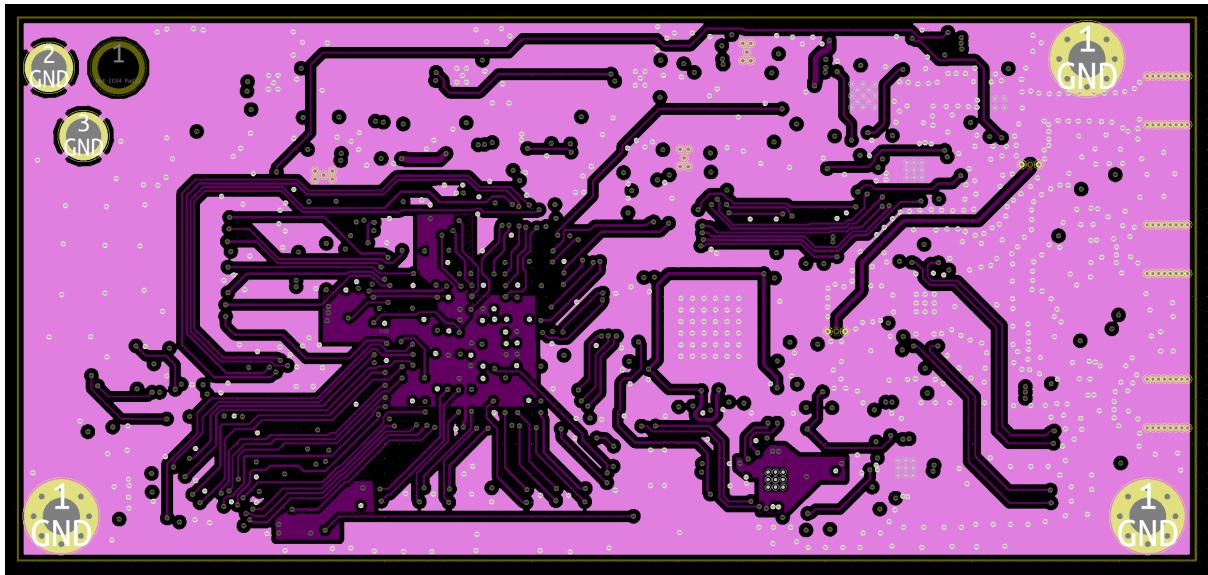


Figure 3.4: Most of layer 3 is a ground plane but there are also a substantial number of trace connecting to the FPGA. It does also contain 1V an 3.3V power planes.

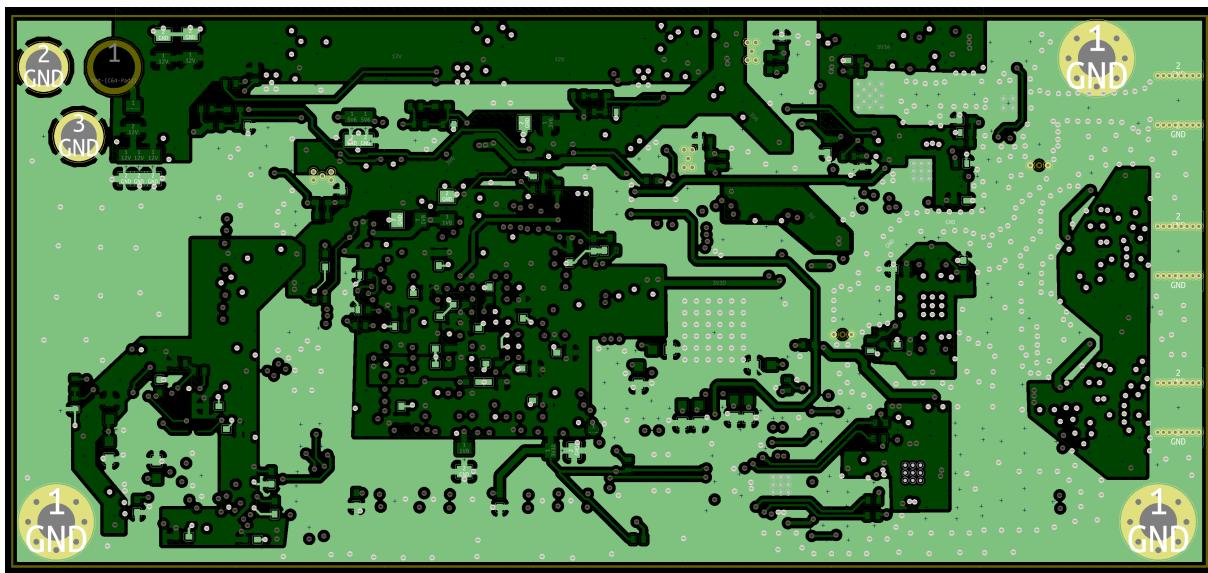


Figure 3.5: The 4th layer of the PCB contains a large ground plane with many vias interspersed.

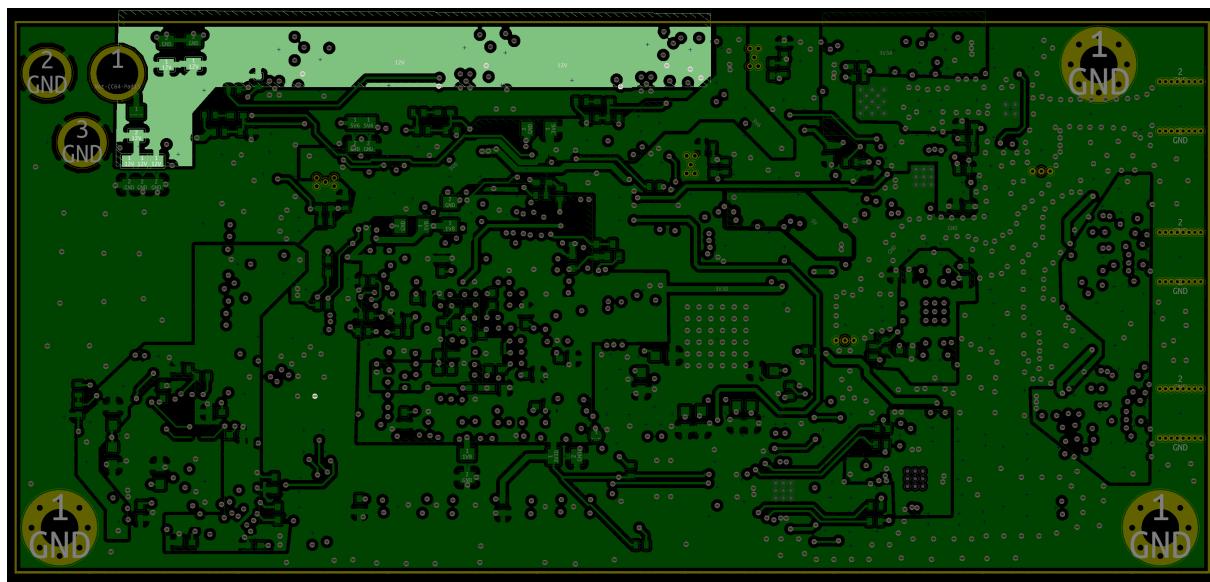


Figure 3.6: The 12V power plane is located at the top of the board adjacent to the D barrel jack and the buck converters it feeds.

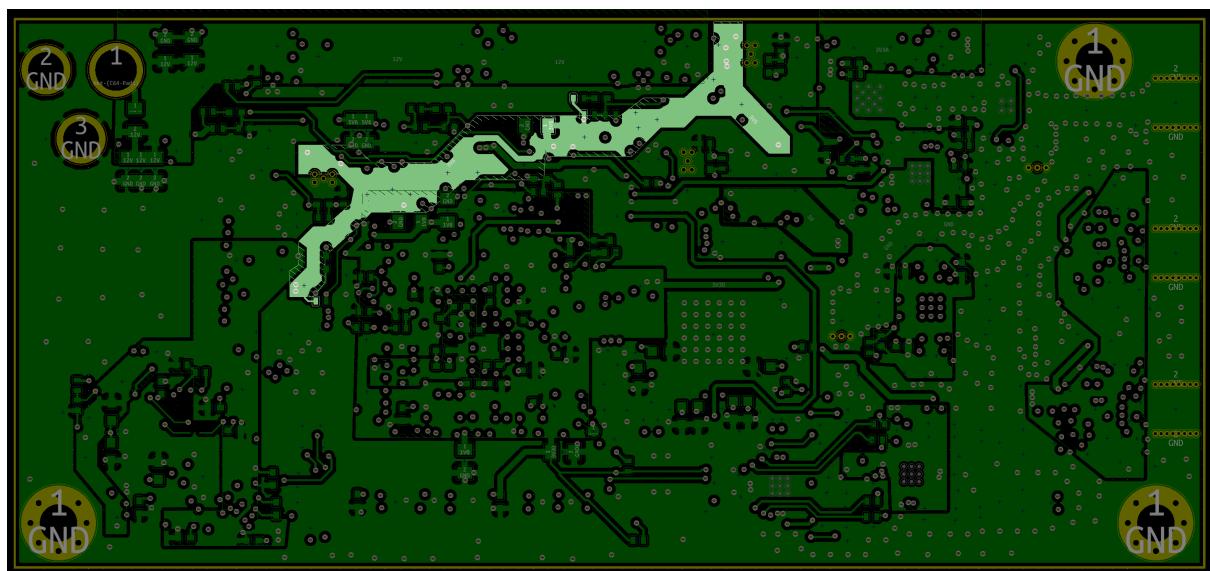


Figure 3.7: There is a 3.6V power plane that is the output of one of the buck converter and is use as the input to several linear regulators that output 1.8V 3.0V and 3.3V.

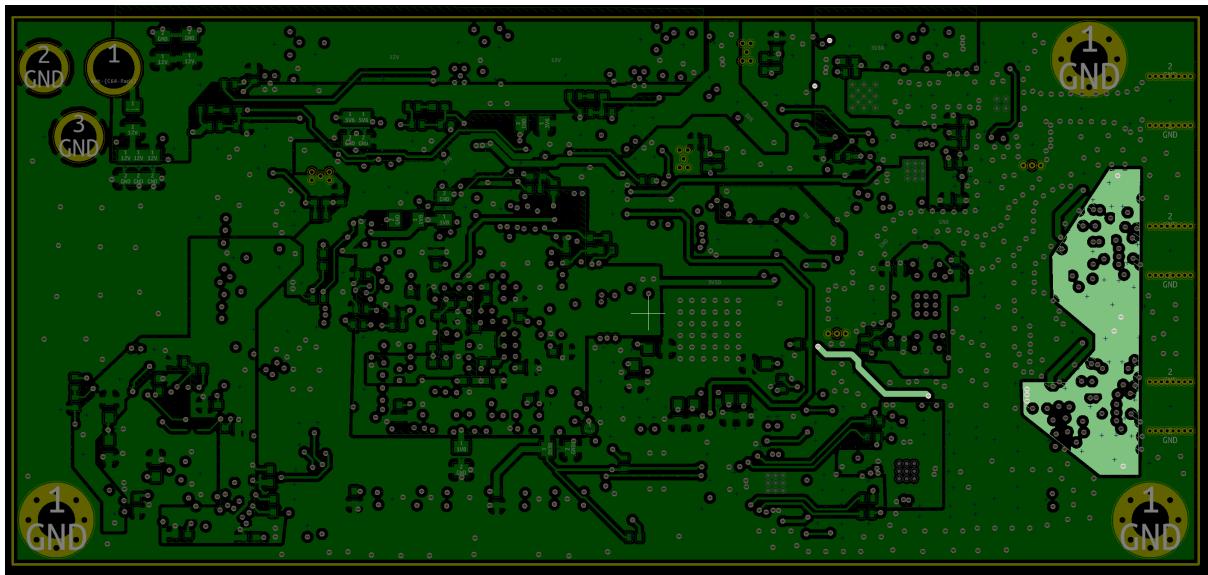


Figure 3.8: A 3V power plane is used to power components near the right side of the board (the right side of the board is where signal transmission and reception occur that amplify receive signals so they can be mixed with the transmitted signal and fed into the ADC before FPGA processing). Notably, the 3V power plane is located near the components but far from the linear regulator that outputs it.

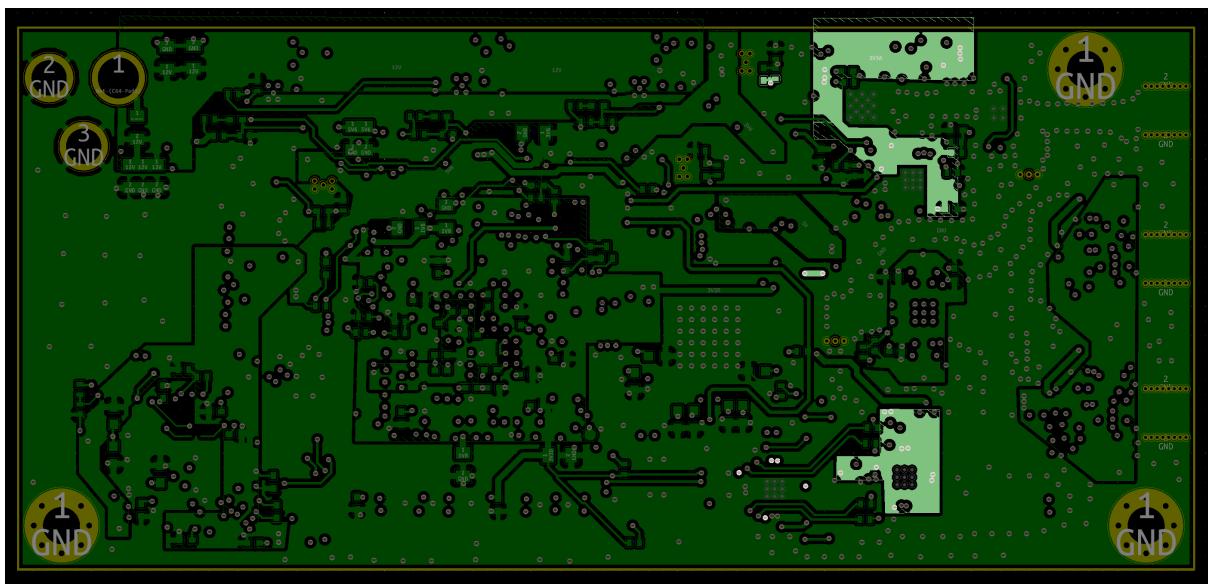


Figure 3.9: A 3.3V power plane is used to power several components for signal transmission, such as a frequency synthesizer, and RF amplifier. These, along with all other reception and transmission circuitry, are located on the right side of the board.

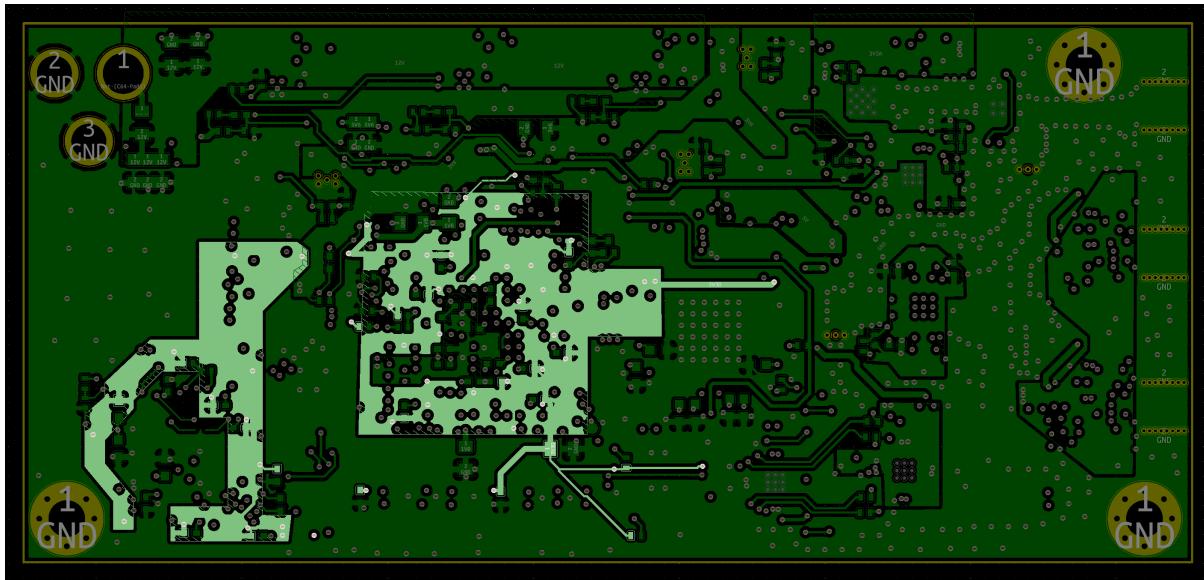


Figure 3.10: Another 3.3V power plane is used as one of the power inputs to the FPGA as well as to a USB-to-UART device (a configuration bit stream is sent from a host computer through a USB cable to the PCB where this device converts it into the proper UART format to be transmitted to the FPGA) and an ADC (the ADC converts the mixed transmitted-received signals to digital and then sends them to the FPGA for processing).

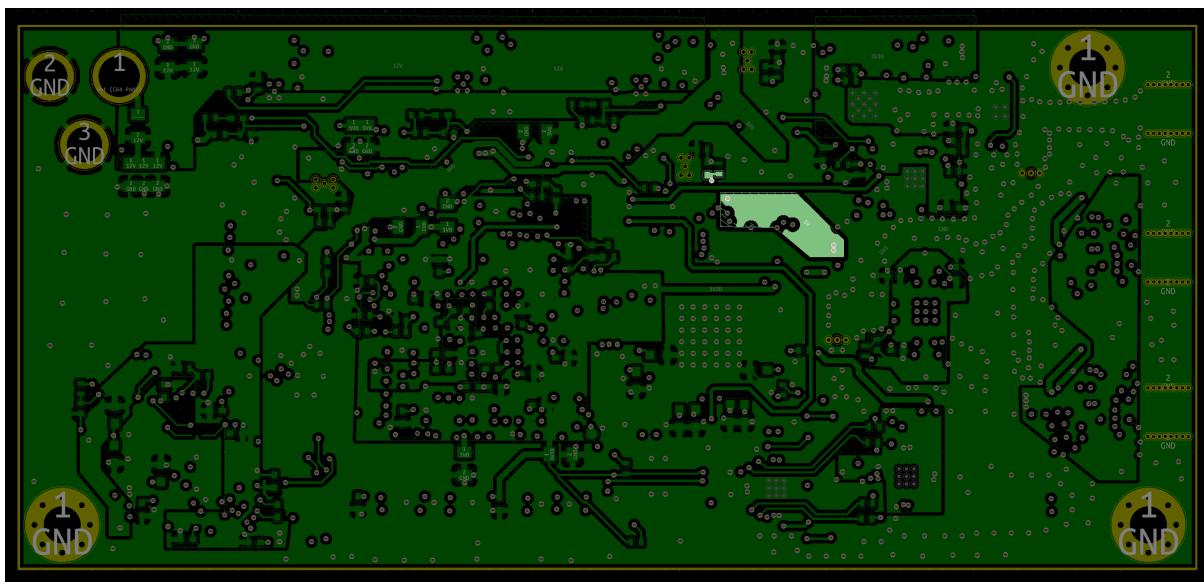


Figure 3.11: A 5V power plane powers one of the inputs to the frequency synthesizer for transmission.

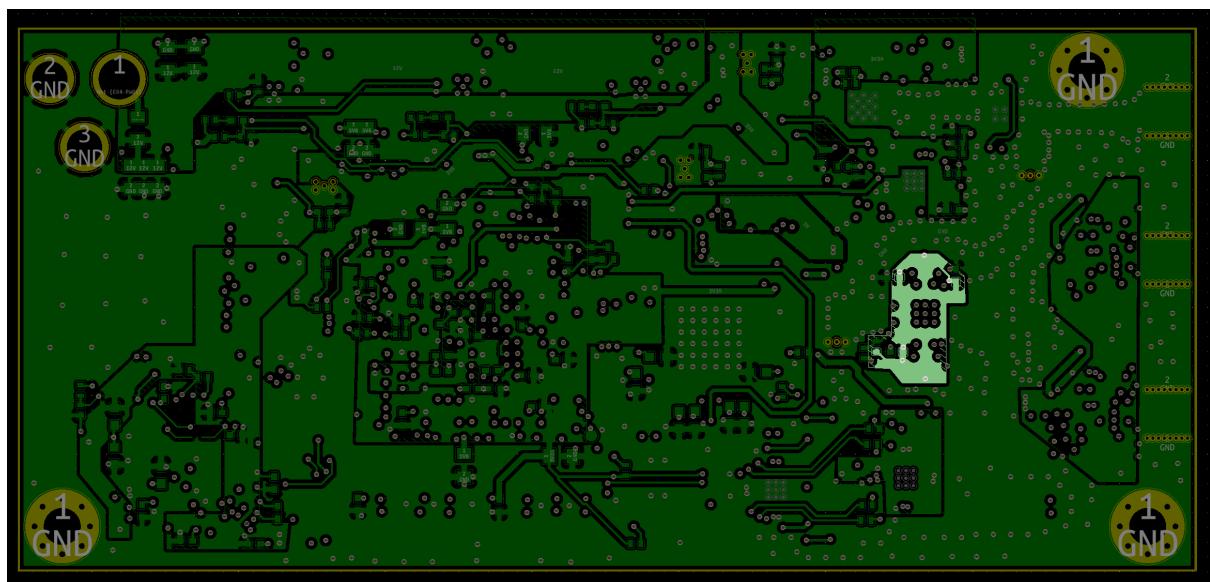


Figure 3.12: Another 5V power plane is used for to power devices after going through a ferrite bead.

Chapter 4

Antennas

There are several promising places to start in learning how to do this. Obviously, the [hforsten blog](#) is a good place. Contextual Electronics also has a [youtube video](#) about building a PCB antenna. Finally, [OpenEMS](#) is apparently a very powerful tool for simulating and designing antennas (among other things RF). This is already installed and can be scripted through Octave. Make sure to look at the tutorials.

One antenna is used for transmission and two are used for reception. It might be possible to achieve the same effect with two overall antennas but the design would need to be significantly altered to do so. The transmission antenna should have high gain and high directivity. This ensures sufficient power for reception, longer range, and less side clutter (i.e. you don't want a wide field of vision, you want this to be mostly directed in a single direction). We use two reception antennas instead of one so that we can perform beamforming. Basically, we mix the signal picked up by each reception antenna individually with the transmitted signal. Each of these mixed products tells us the distance from the object. However, if the object is at a slight angle, the phase difference between the two mixed products tells us the incident angle, so we can localize the distant object.

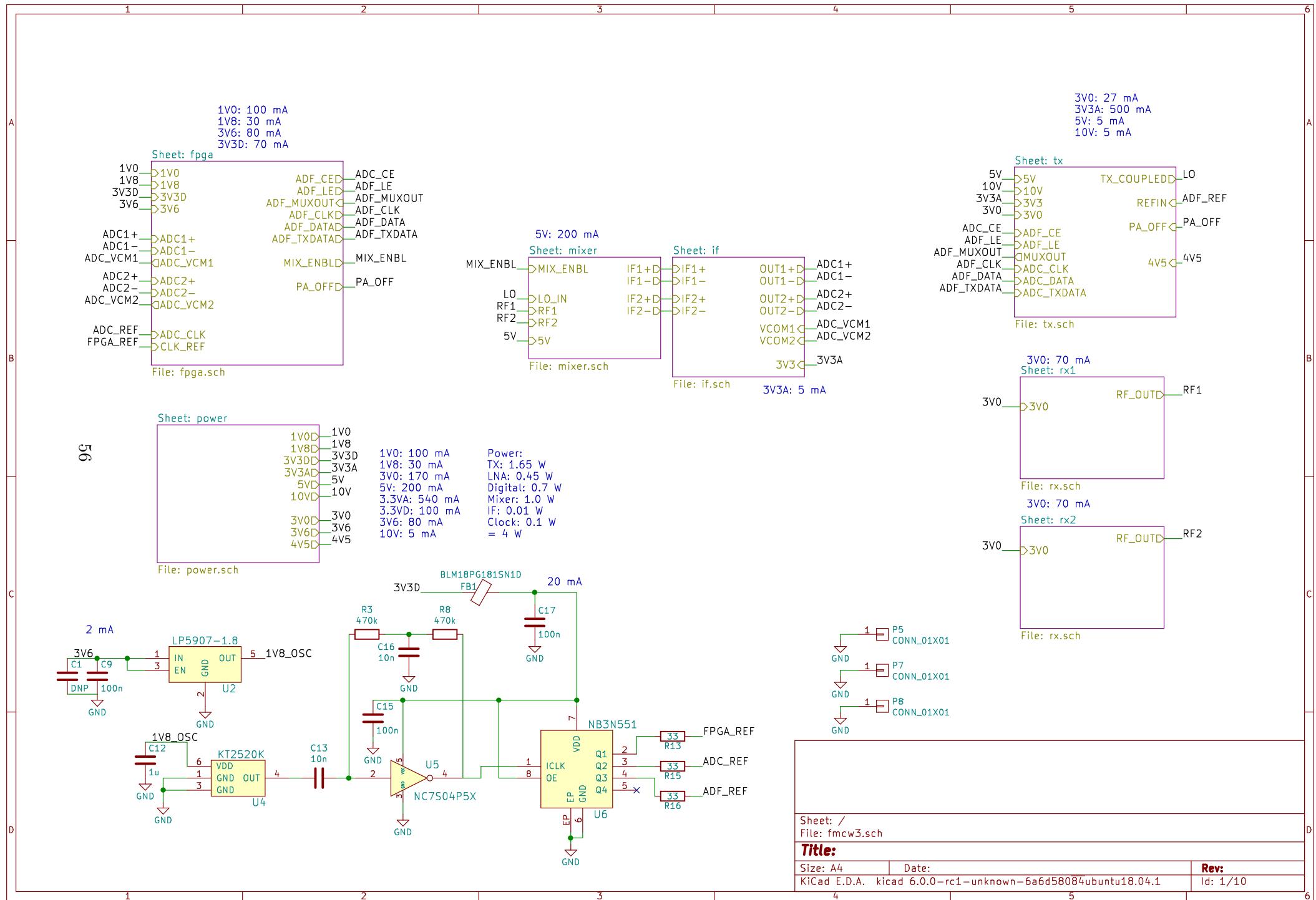
[This antenna](#) looks like it would be good for transmission. [This RF Elements](#) antenna looks even better, but is probably more than I'm willing to pay (it's gain is lower, however). [The Triple Feed Patch Array Antenna](#) is a great deal for the price and is based off a clever open-source design, [here](#). I'm not sure if the gain will be quite enough, however. If you're confused about the operation of that last one, look at the description for it on Antenna Test Lab. They provide excellent information. I'm mostly struggling to find a good patch array antenna for reception. I may have to just build this myself, but I'm worried about the quality using normal FR4.

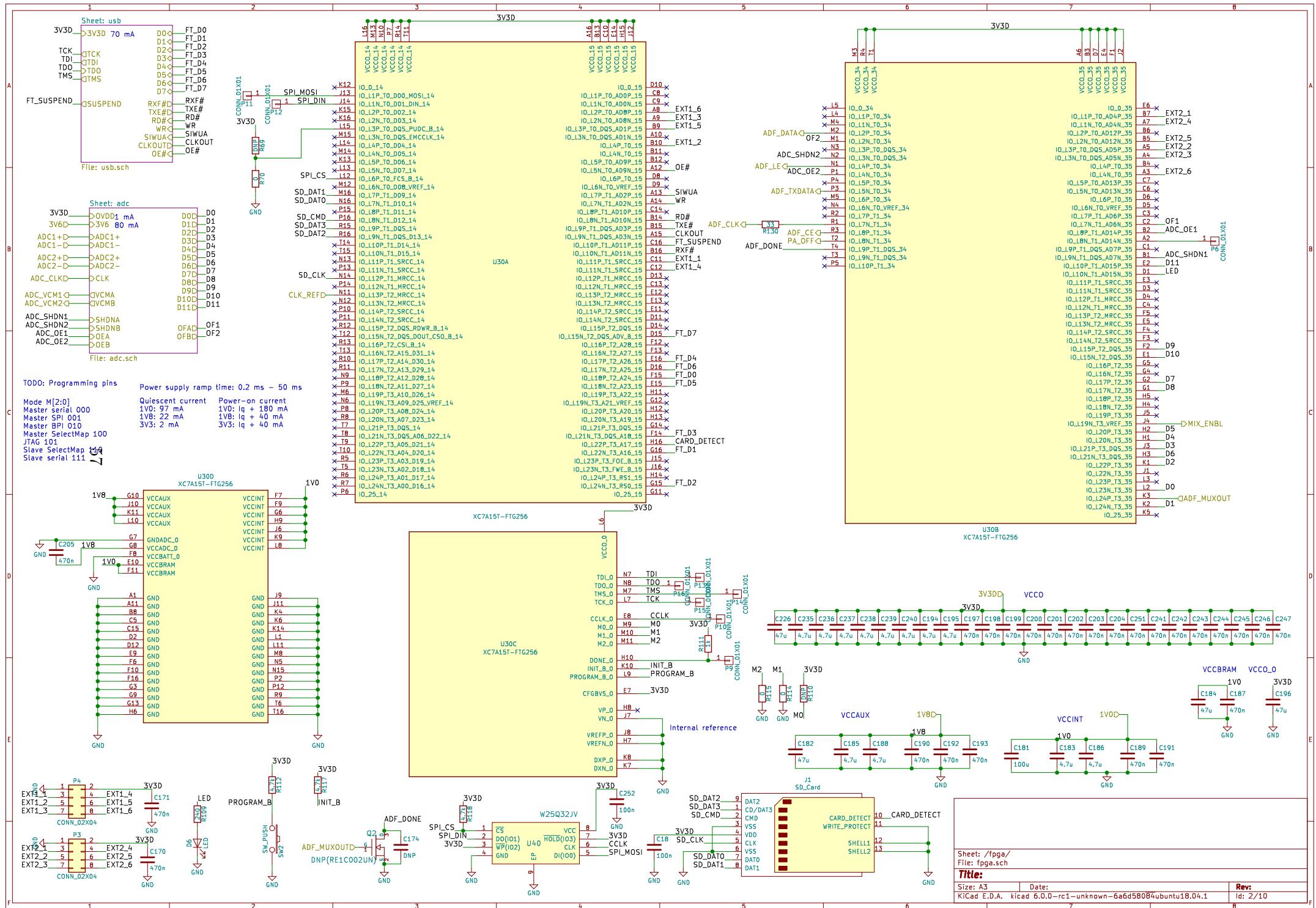
Chapter 5

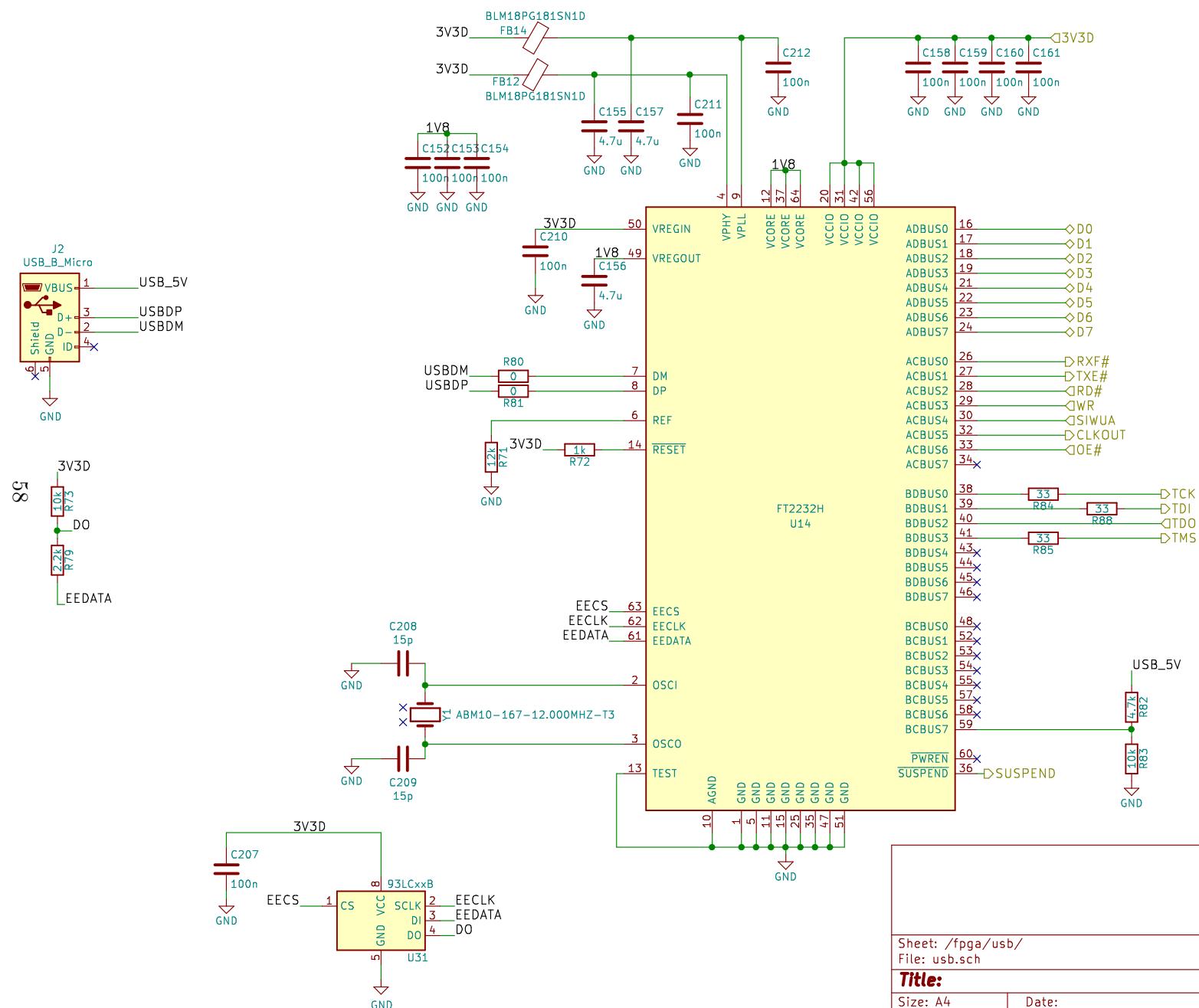
Links

Chapter 6

Full Schematic







Sheet: /fpga/usb/

File: usb.sch

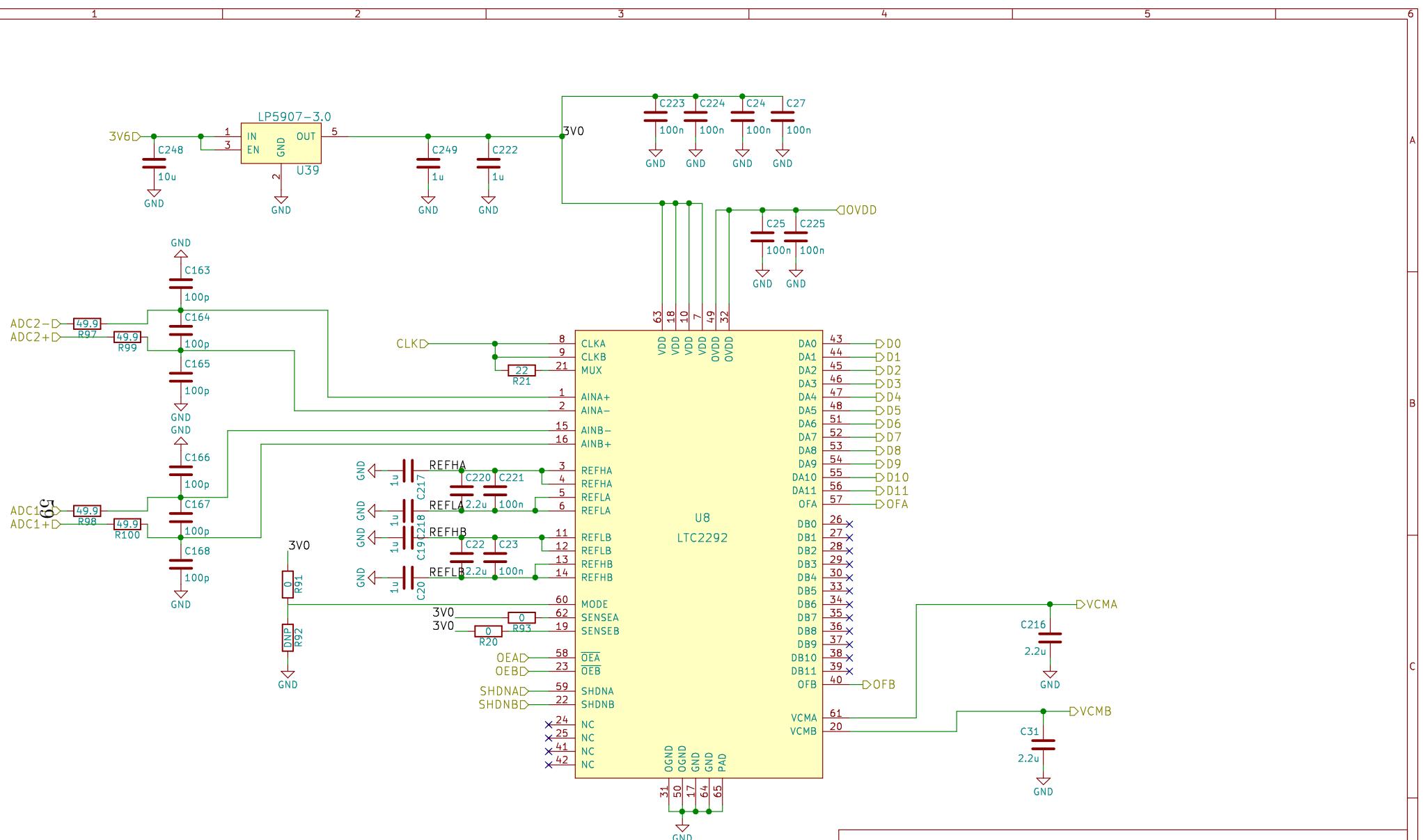
Title:

Size: A4 Date:

KiCad E.D.A. kicad 6.0.0-rc1-unknown-6a6d58084ubuntu18.04.1

Rev:

Id: 3/10



Sheet: /fpga/adc/
File: adc.sch

Title:

Size: A4 Date:
KiCad E.D.A. kicad 6.0.0-rc1-unknown-6a6d58084ubuntu18.04.1

Rev:
Id: 4/10

10V current: 1mA
3.3V current: 600mA
5V current: 1mA

09

BLM18PG181SN1D
FB2

DNP DNP

C215 C214

GND GND

100n

BLM18PG181SN1D
FB3

DNP R40

GND

100n

BLM18PG181SN1D
FB4

DNP R41

GND

100n

BLM18PG181SN1D
FB5

DNP R42

GND

100n

BLM18PG181SN1D
FB6

DNP R43

GND

100n

BLM18PG181SN1D
FB7

DNP R44

GND

100n

BLM18PG181SN1D
FB8

DNP R45

GND

100n

BLM18PG181SN1D
FB9

DNP R46

GND

100n

BLM18PG181SN1D
FB10

DNP R47

GND

100n

BLM18PG181SN1D
FB11

DNP R48

GND

100n

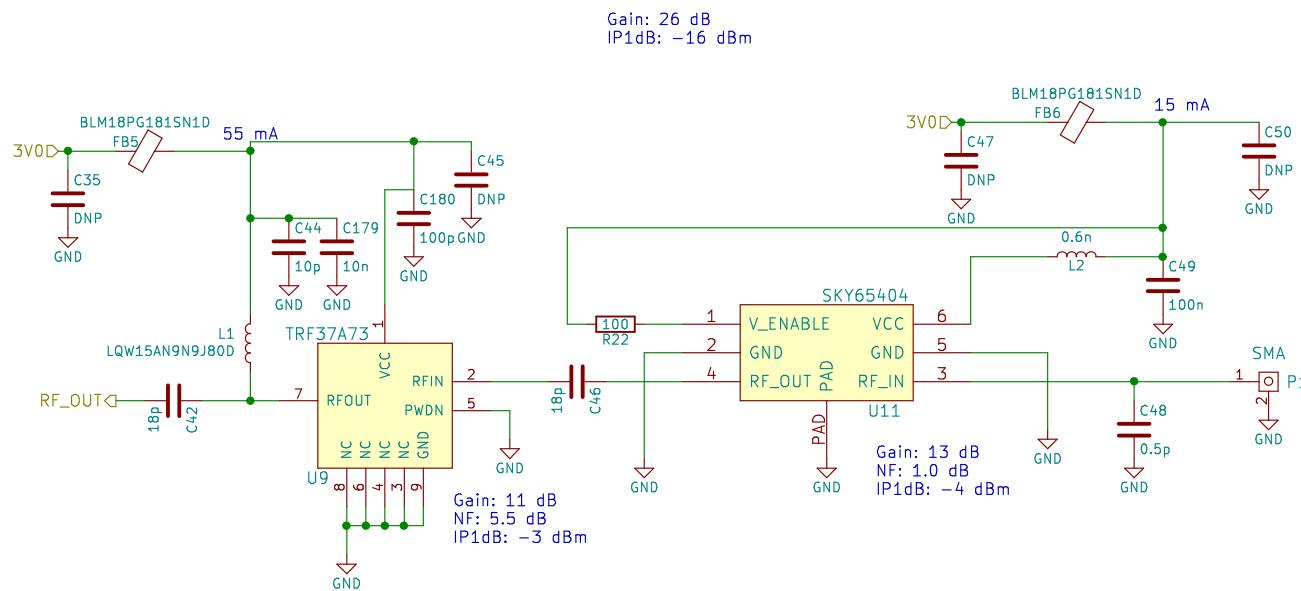
BLM18PG181SN1D
FB12

DNP R49

GND

100n

I9



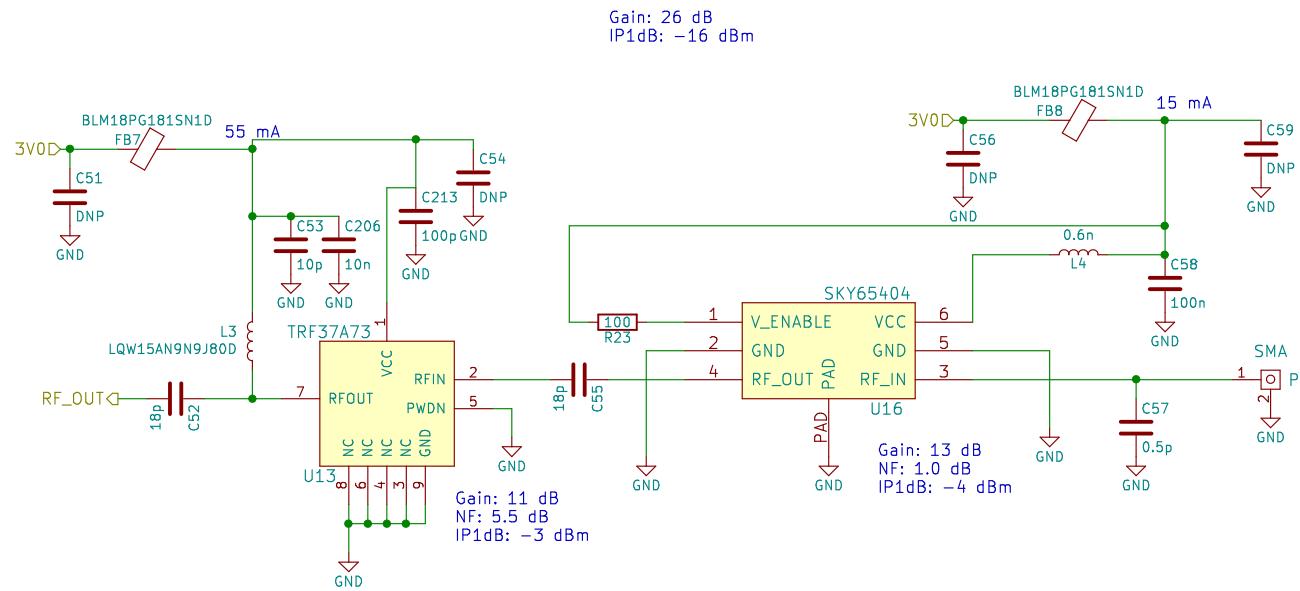
Sheet: /rx1/
File: rx.sch

Title:

Size: A4 Date:
KiCad E.D.A. kicad 6.0.0-rc1-unknown-6a6d58084ubuntu18.04.1

Rev:
Id: 6/10

62

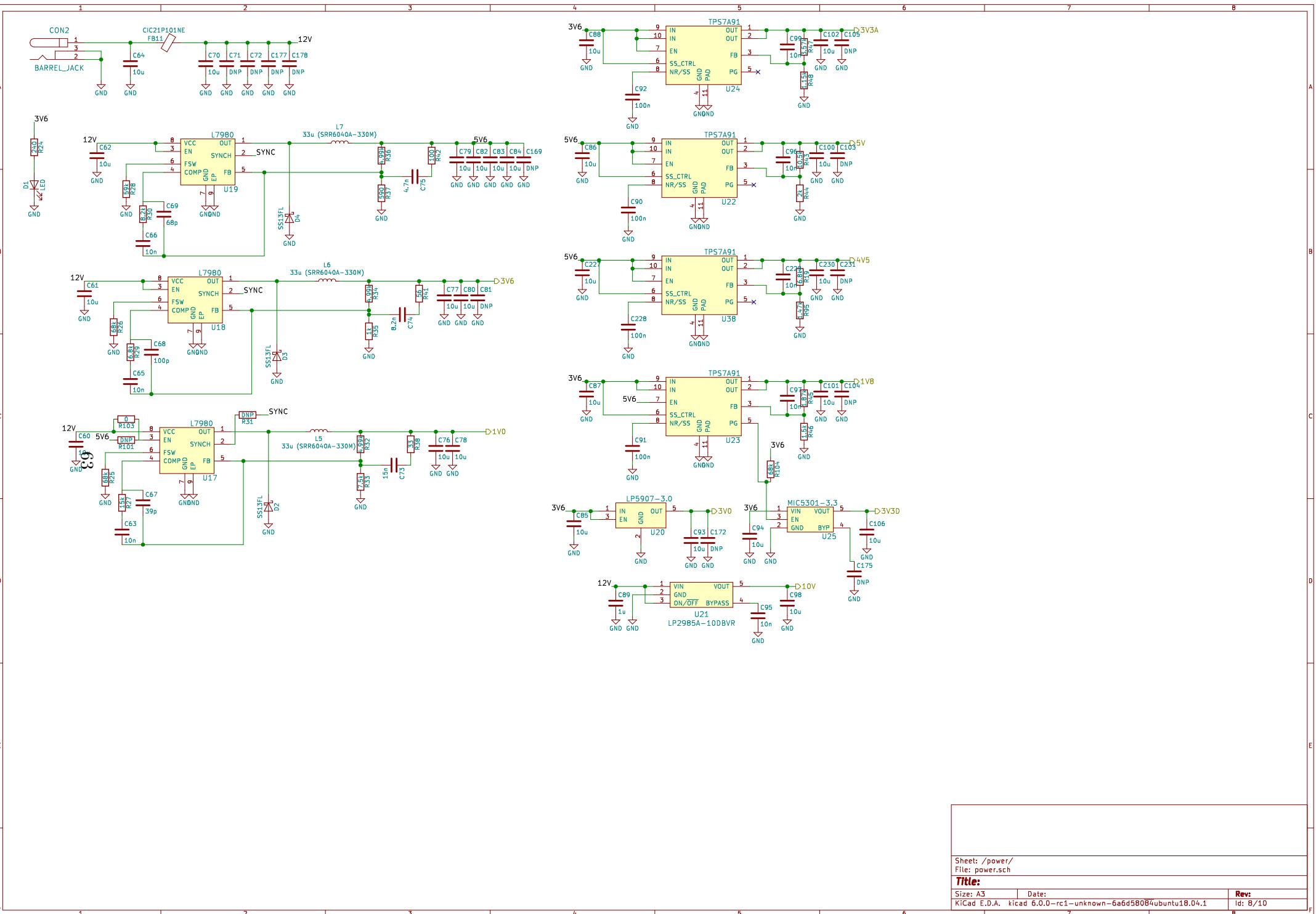


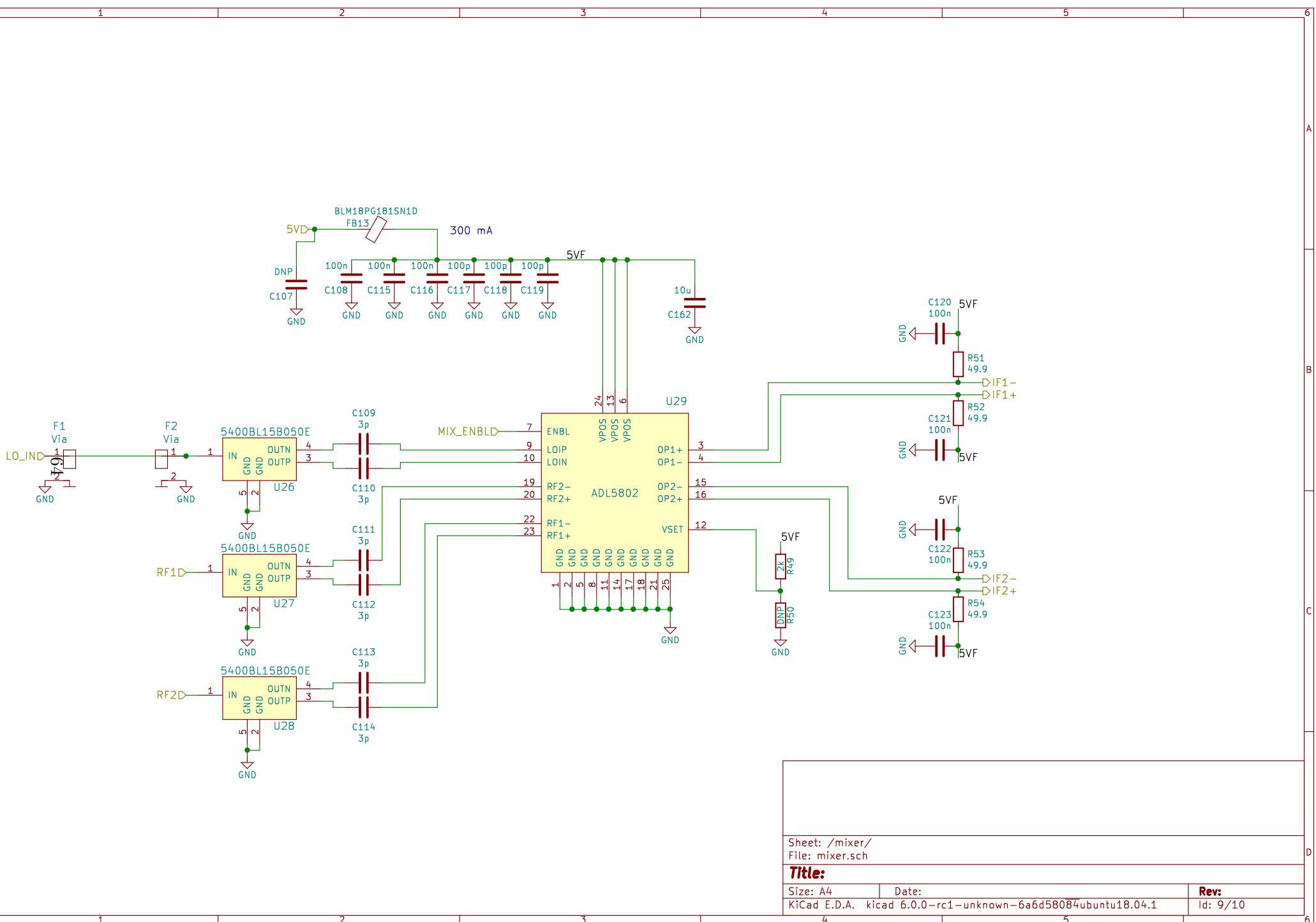
Sheet: /rx2/
File: rx.sch

Title:

Size: A4 Date:
KiCad E.D.A. kicad 6.0.0-rc1-unknown-6a6d58084ubuntu18.04.1

Rev:
Id: 7/10





Sheet: /mixer/
File: mixer.sch

Title:

Size: A4 Date:
KiCad E.D.A. kicad 6.0.0-rc1-unknown-6a6d58084ubuntu18.04.1

Rev:
Id: 9/10

