

# Contents

<b>1</b>	<b>Attribution &amp; Intent</b>	<b>3</b>
<b>2</b>	<b>Overview</b>	<b>4</b>
2.1	Distance Calculation . . . . .	4
2.2	Angle Calculation . . . . .	5
<b>3</b>	<b>Circuit Description</b>	<b>6</b>
3.1	Power . . . . .	7
3.1.1	Overview . . . . .	7
3.1.2	Barrel Jack / Power Input . . . . .	7
3.1.3	TPS5420D Buck Converter . . . . .	7
3.1.4	TPS560200 Buck Converter . . . . .	9
3.1.5	LP2985A-10DBVR Linear Regulator . . . . .	10
3.1.6	TPS7A91 LDO Voltage Regulator . . . . .	12
3.1.7	TPS7A7001DDA LDO Regulator . . . . .	14
3.1.8	LP5907MFX-3.0 . . . . .	15
3.2	Top . . . . .	18
3.3	Overview . . . . .	18
3.3.1	KT2520K 40MHz TXCO . . . . .	18
3.3.2	NC7S04 Inverter . . . . .	18
3.3.3	NB3N551 1:4 Clock Buffer . . . . .	18
3.4	TX . . . . .	19
3.4.1	ADF4158 Frequency Synthesizer . . . . .	19
3.4.2	HMC431LP4RF VCO . . . . .	27
3.4.3	SE2567L Power Amplifier . . . . .	27
3.4.4	Notes . . . . .	27
3.5	USB . . . . .	29
3.5.1	FT2232H . . . . .	29
3.5.2	93LC46B 1k EEPROM . . . . .	29
3.6	FPGA . . . . .	31
3.6.1	Overview . . . . .	31
3.6.2	XC7A15T-FTG256 Xilinx FPGA . . . . .	31
3.6.3	W25Q32JV Flash Memory . . . . .	40
3.7	ADC . . . . .	41
3.7.1	Overview . . . . .	41
3.7.2	LTC2292 . . . . .	41
3.8	Mixer . . . . .	45
3.8.1	ADL5802 . . . . .	45

3.9	IF	46
3.9.1	ADA4940-2 IF Differential Amplifier	46
3.10	RX1 & RX2	47
<b>4</b>	<b>PCB</b>	<b>48</b>
4.1	General Layout	48
4.2	Component Layout Considerations	48
4.2.1	Power Input	48
4.2.2	LTC2292 ADC	49
4.2.3	RX1 / RX2	49
4.3	RF Impedance Matching	49
<b>5</b>	<b>Antennas</b>	<b>58</b>
<b>6</b>	<b>Links</b>	<b>59</b>
<b>7</b>	<b>Full Schematic</b>	<b>60</b>

# Chapter 1

## Attribution & Intent

This design is adapted from [a blog post by Henrik Forstén](#). A large portion of the design and schematics as well as parts of the description are his and should be thus credited. I've made a number of modifications to the schematic and attempted to thoroughly document the radar's operation in an effort to understand it. I hope that the description below will be useful to others who want to understand how a radar can be built from scratch.

# Chapter 2

## Overview

### 2.1 Distance Calculation

The principle of operation of an FMCW radar is shown in Fig. 2.1. A sinusoidal signal that ramps in frequency is transmitted through air via an antenna. The reflected signal (which is just a phase-shifted version of the original signal) is picked up by another antenna (or set of antennas) and modulated with the original signal. The distance to the object can be computed from the result, shown in Eq. 2.1, where  $\Delta f$  is the frequency difference,  $t_{\text{ramp}}$  is the known ramp duration,  $f_{\text{ramp}}$  is the difference between the high and low ramp frequencies,  $c$  is the speed of light and the 2 is present because the time measured is the round-trip time.

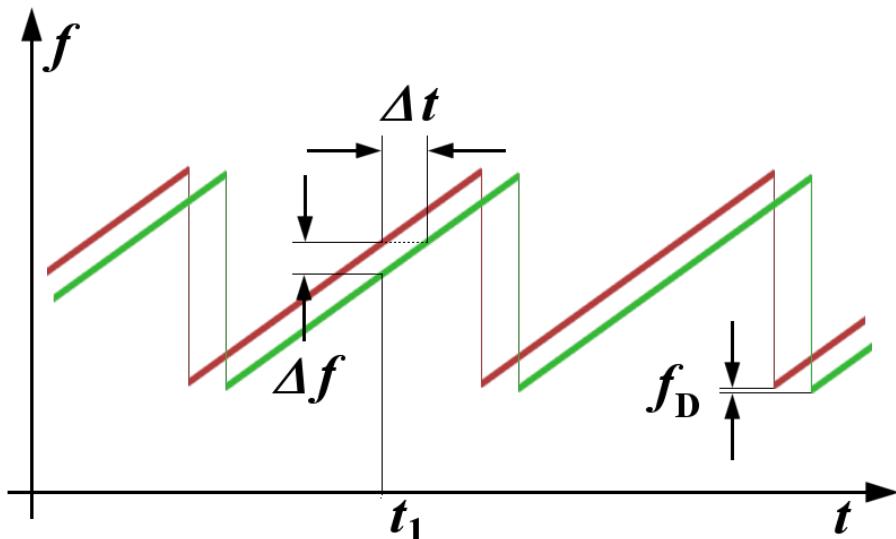


Figure 2.1: FMCW radar operation: a delayed signal is modulated with the original signal and the distance to the remote object can be backed out from the phase difference between the two signals.

$$d = \frac{ct_{\text{ramp}}\Delta f}{2f_{\text{ramp}}} \quad (2.1)$$

To calculate the frequency difference, we first express it in terms of the round-trip time,  $t_d$ :

$$\Delta f = \frac{t_d f_{\text{ramp}}}{t_{\text{ramp}}} \quad (2.2)$$

The transmitted signal during one ramp is:

$$f(t) = \sin(\omega t) \quad (2.3)$$

$$= \sin(2\pi f(t)t) \quad (2.4)$$

$$= \sin\left(2\pi t\left(f_0 + t\frac{f_{\text{ramp}}}{t_{\text{ramp}}}\right)\right) \quad (2.5)$$

The received signal (which is just a phase shifted version of the transmitted signal) is  $f(t - t_d)$ . Using the mixer output we can compute the time delay:

$$m = f(t)f(t - t_d) \quad m \text{ is the mixer output.} \quad (2.6)$$

$$= \sin(\omega t)\sin(\omega(t - t_d)) \quad (2.7)$$

$$= \cos(\omega t_d)/2 \quad ^1 \quad (2.8)$$

$$t_d = \frac{\arccos(2m)}{2\pi\left(f_0 + t\frac{f_{\text{ramp}}}{t_{\text{ramp}}}\right)} \quad (2.9)$$

The distance can be computed with Eq. 2.10.

$$d = \frac{c \arccos(2m)}{4\pi\left(f_0 + t\frac{f_{\text{ramp}}}{t_{\text{ramp}}}\right)} \quad (2.10)$$

## 2.2 Angle Calculation

---

<sup>1</sup>We've used the identity  $\sin \theta \sin \phi = [\cos(\theta - \phi) - \cos(\theta + \phi)]/2$  and the fact that the sum term is outside the IF amplifier pass-band and will be filtered out.

## Chapter 3

# Circuit Description

## 3.1 Power

*schematic*

FIXme Fatal!

**FIXme Fatal:** I think this section needs information about PSRRs, especially for regulators that feed RF circuits.

FIXme Fatal!

**FIXme Fatal:** Another thing that needs clarification is where should the LDO's be placed? Should they be located near the loads they serve or the switching converters? My guess would be near the converters since that decreases the part of the board that is exposed to the switching noise.

### 3.1.1 Overview

A barrel jack is used to feed a 12V input to the board which is then administered to a 10V output linear regulator and two buck converters that output 5.6V and 3.6V. The buck converters in turn pass their voltages on to several LDO voltage regulators for input to the various digital ICs on the board. The benefit of chaining switching converters to linear regulators is greater energy efficiency and better noise suppression than can be achieved by using either alone. An LED indicates when power is administered to the board.

### 3.1.2 Barrel Jack / Power Input

A ferrite bead pi filter is placed at the output of the barrel jack connection. The ferrite bead is rated for  $5.1A$ , which is more than double the absolute max current draw of the radar. The barrel connector is a switched jack, but we do not use a battery on this PCB so the 3rd pin is grounded.

### 3.1.3 TPS5420D Buck Converter

#### Description

The TPS5420D is an internally-compensated buck converter with a fixed switching frequency of 500kHz, whose block diagram is shown in Figure 3.1. The converter first compares the divided voltage output with a 1.221V reference and uses an error amplifier to amplify the difference. It then feeds that difference into a PWM comparator along with a sawtooth ramp waveform. If the PWM comparator outputs a high voltage the switch is turned off, effectively decreasing the duty cycle. Conversely, a low voltage saturates the transistor and increases the duty cycle. This has the effect of converging the output voltage to its set point. The TPS5420 has a ramp time of between 6.6 and 10ms.

#### Pinout

FIXme Fatal!

**FIXme Fatal:** What is the point of BOOT?

Table 3.1: TPS5420D pinout.

#	Pin	Description
1	BOOT	

---

2, 3	NC	No connect.
4	VSENSE	Feedback voltage for the regulator. This compares the divided output with a 1.221V reference.
5	ENA	Enable pin. This can be left floating to enable the device.
6	GND	Ground.
7	VIN	Input supply voltage.
8	PH	Output voltage.

---

## Component Selection

An inductor is chosen such that it satisfies the following 3 requirements:

1. The inductance is given by Equation 3.1.
2. The current rating is at least 2x the maximum load current.
3. The self-resonant frequency is at least 10x the switching frequency.

$$L_{\min} = \frac{T}{2} \frac{V_{\text{out}}}{I_{\text{out(min)}}} \left( 1 - \frac{V_{\text{out}}}{V_{\text{in}}} \right) \quad (3.1)$$

The requirements are given in Table 3.1.3. All of these requirements are satisfied by the [SRR1210A-330M](#) 33 $\mu$ H Bourns ferrite bead.

**Table 3.2: TPS5420D inductor requirements.** I've assumed an input ripple of 300mV.

---

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

---

The input, output and boot capacitors were chosen by the [WEBENCH tool](#). The flyback diode was chosen to support 2A of current.

## Downstream Current Draw

It is important to know both the maximum and minimum current draw of components whose voltage inputs are fed by the output of a buck converter. The minimum current sets a lower bound on the inductance that can be used (see Equation 3.1) since the converter must remain in CCM for proper operation. It is also important to ensure that the buck converter and inductor are rated for the maximum current draw of downstream devices. Saturating the ferrite-core inductor can cause a loss in inductance.

**Table 3.3: The current draw of components downstream from the 5.6V buck converter. When minimum current draw is omitted from the datasheet I've assumed 0A.**

---

MFN	I <sub>Q</sub>	I <sub>max</sub>	Voltage Inputs
ADL5802	170mA	300mA	5V
ADF4158	312.5 $\mu$ A	5mA	5V (VP)

---

<b>SE5004L</b>	300mA	800mA	5V (VCC1, VCC2, VCC3)
Total	470mA	1.105A	-

**Table 3.4: Components downstream from the 3.6V buck converter.**

MFN	I <sub>Q</sub>	I <sub>max</sub>	Voltage Inputs
<b>KT2520K</b>	-	2mA	1.8V
<b>NC7S04</b>	1μA	12.5mA	3.3V
<b>NB3N551</b>	-	40mA	3.3V
<b>XC7A15T-FTG256</b>	97mA	277mA	1V (VCCINT, VCCBRAM)
XC7A15T-FTG256	22mA	87mA	1.8V (VCCADC, VCCAUX)
XC7A15T-FTG256	1mA	201mA	3.3V (VCC0)
<b>W25Q32JV</b>	10μA	25mA	3.3V
<b>FT2232H</b>	510μA	280mA	3.3V (VPHY, VPLL, VCORE, VCCIO)
<b>93LC46B</b>	-	2mA	3.3V
<b>LTC2292</b>	5mA	95mA	3.3V (OVDD), 3V (VDD)
<b>ADA4940-2</b>	4.2mA	5.52mA	3.3V
<b>ADF4158</b>	-	32mA	3.3V (AVDD, DVDD)
<b>HMC431LP4</b>	19mA	27mA	3V
<b>TRF37A73</b>	250μA	130mA	3V
<b>SKY65404</b>	20mA	36mA	3V (VENABLE, VCC)
Total	169mA	1.252A	-

## PCB Layout

The datasheet provides a suggested PCB layout, shown in Figure 3.2.

### 3.1.4 TPS560200 Buck Converter

#### Description

The TPS560200 is an internally-compensated buck converter with a switching frequency of 600kHz and a fixed ramp time of 2ms. It is needed, rather than a linear regulator chained to a TPS5420D, because the FPGA requires its 1V power supply first.

#### Pinout

**Table 3.5: TPS50600 pinout.**

#	Pin	Description
1	EN	Enable pin. Can be floated to unconditionally enable device.
2	GND	Ground.
3	PH	Output voltage.

---

4	VIN	Input voltage.
5	VSENSE	Feedback, compared to a 0.8V reference.

---

## Component Selection

The voltage divider resistors, filter capacitors and input capacitors are specified by the datasheet. The output voltage ripple must be sufficiently small that it falls within the required voltage range of the FPGA's V<sub>CCINT</sub> input. Specifically, it must be less than 100mV. Texas Instruments provides<sup>1</sup> an equation for the voltage ripple of a buck converter, given a small capacitor resistance. I'm further assuming an ESR of 0 since I'm using ceramic capacitors at the output and the current draw is fairly low. The equation is provided in Equation 3.2. The ripple current for a buck converter is given in Equation 3.3. Finally, the remaining inductor requirements are specified in Table 3.1.4. All of these requirements are satisfied by a 10μH inductor, which is used in the datasheet. However, this puts the inductor a bit close to DCM and since I have extra 33μH inductors on hand I'd rather be safe and use one of those.

$$V_{\text{p2p}} = \frac{I_{\text{p2p}}}{8CF_{\text{SW}}} \quad (3.2)$$

$$I_{\text{p2p}} = V_{\text{out}} \frac{1 - D}{LF_{\text{SW}}} \quad (3.3)$$

**Table 3.6: TPS560200 inductor requirements.**

V <sub>out</sub>	L <sub>min</sub>	Current Rating	SRF <sub>min</sub>
1V	7.9μH	554mA	6MHz

## Downstream Current Draw

**Table 3.7: Components downstream from the TPS560200 buck converter.**

MFN	I <sub>Q</sub>	I <sub>max</sub>	Voltage Inputs
XC7A15T-FTG256	97mA	277mA	1V (VCCINT, VCCBRAM)

## PCB Layout Guidelines

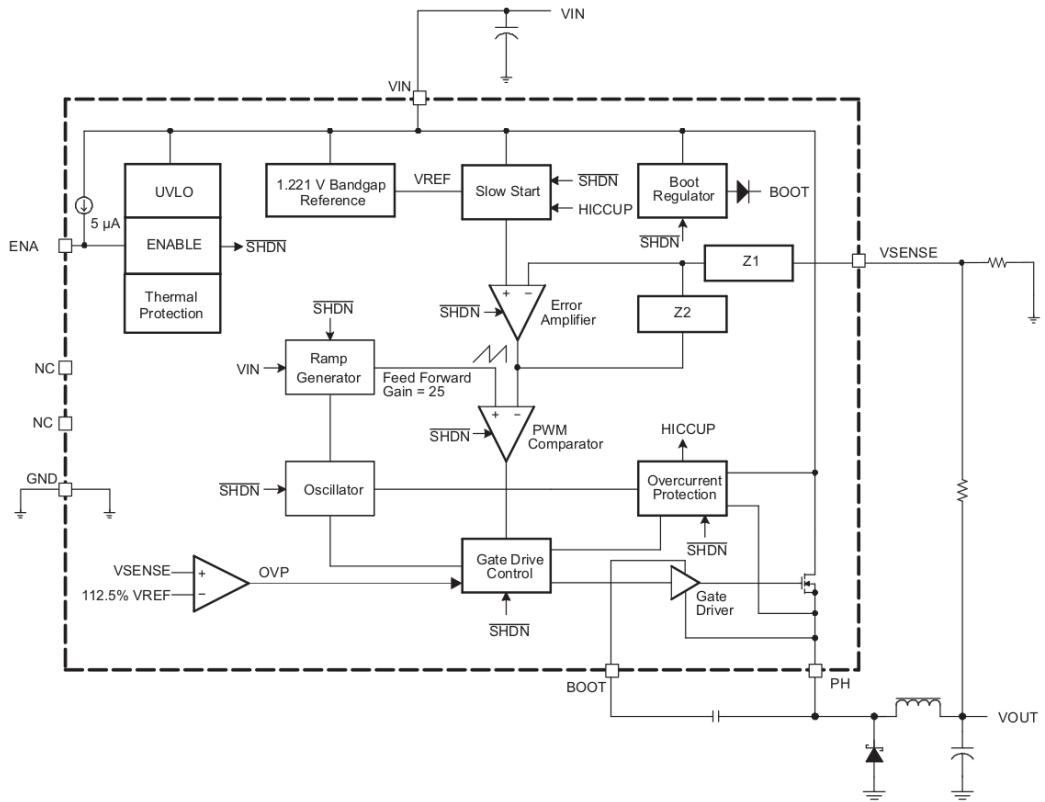
The datasheet provides a recommended layout, shown in Figure 3.3.

### 3.1.5 LP2985A-10DBVR Linear Regulator

#### Description

The LP2985A is a low-dropout fixed-output regulator with a max output current of 150mA. Its functional block diagram is shown in Figure 3.4. It uses an op-amp to compare the voltage-divided output with a 1.23V reference. It then turns a PNP transistor on/off depending on the output voltage level relative to its 10V target.

<sup>1</sup>[www.ti.com/lit/an/slva630a/slva630a.pdf](http://www.ti.com/lit/an/slva630a/slva630a.pdf)



**Figure 3.1: TPS5420D Block Diagram**

## Pinout

**Table 3.8: LP2985A's pin descriptions.**

#	Pin	Description
1	VIN	Voltage supply.
2	GND	Ground.
3	ON/OFF	Active-low shutdown pin. It is tied to VIN since the device should be unconditionally enabled.
4	BYPASS	Connected to a 10nF capacitor to ground to decrease output voltage noise.
5	VOUT	Regulated, voltage output.

## Component Selection

The input capacitor should be at least  $1\mu\text{F}$  and the output capacitor should be at least  $2.2\mu\text{F}$ , but higher values are better at the output. I'm using  $10\mu\text{F}$ .

## Downstream Current Draw

**Table 3.9: Downstream current draw for the 10V linear regulator.**

MFN	I <sub>Q</sub>	I <sub>max</sub>	Voltage Inputs
TLV172DCK	1.6mA	75mA	10V

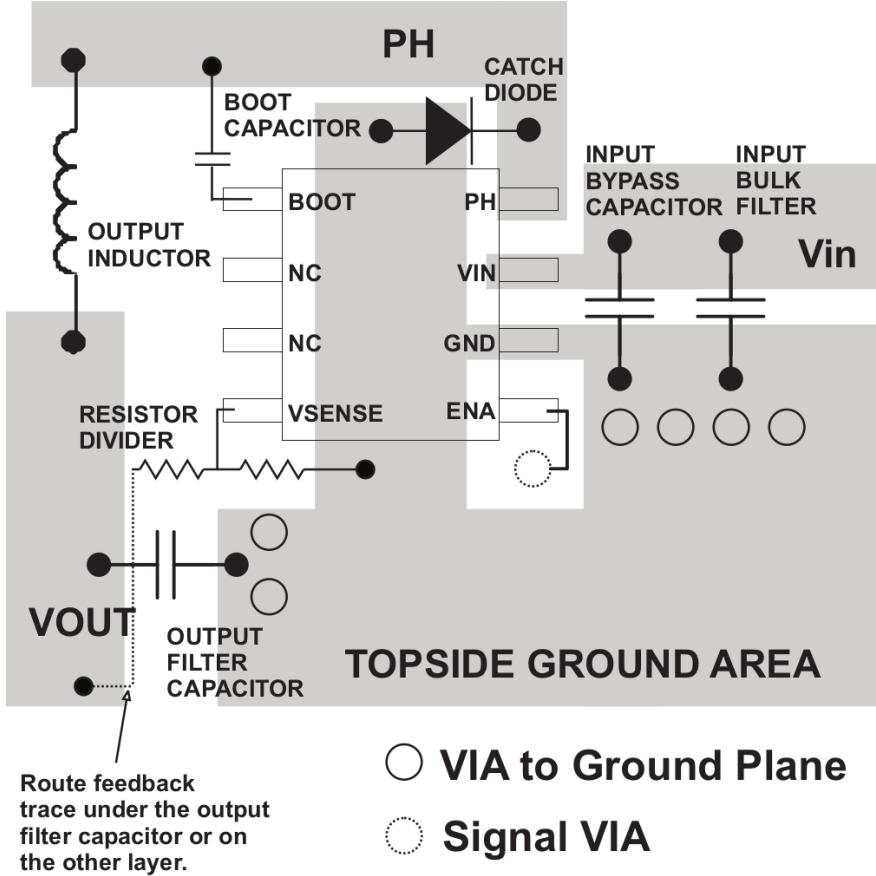


Figure 3.2: TPS5420D suggested PCB layout.

## PCB Layout

The input and output capacitors should be placed within 1cm of the input and output pins and have a low-impedance path to ground.

### 3.1.6 TPS7A91 LDO Voltage Regulator

#### Description

The TPS7A91 is a low-noise, low-dropout and adjustable voltage regulator. It has an RMS output noise of  $4.7\mu\text{V}$ . The regulator works by comparing the feedback voltage, at the inverting input of an error amplifier, with a 0.8V reference voltage at the non-inverting input. The output of the error amplifier is connected to the gate of an n-channel MOSFET whose drain is attached to the input voltage and whose source is attached to the output voltage. This negative feedback creates a stable equilibrium when the output voltage is at its set-point. The functional block diagram is shown in Figure 3.5.

The power-good function of one regulator is used to ensure that the 1.8V supply arrives at the FPGA before the 3.3V supply. The circuit uses an open-drain output with a pull-up resistor. If the output voltage goes below some threshold, the MOSFET will saturate and the PG line will be brought low. Otherwise, it will be pulled high.

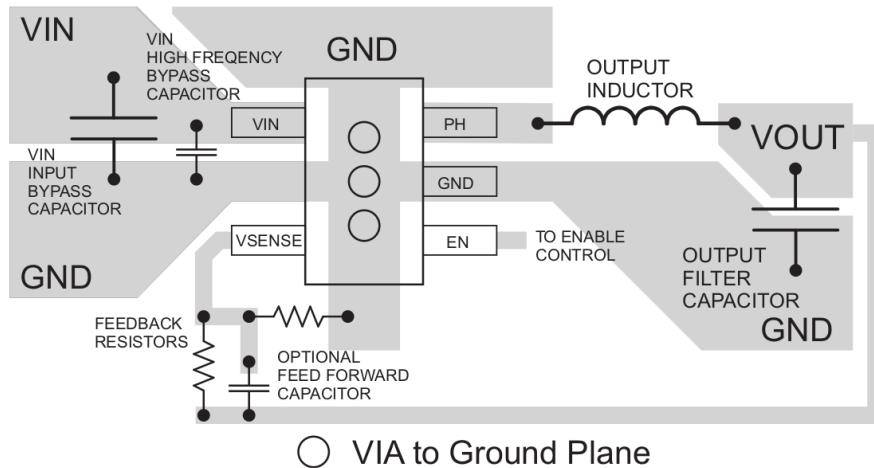


Figure 3.3: TPS560200 recommended PCB layout.

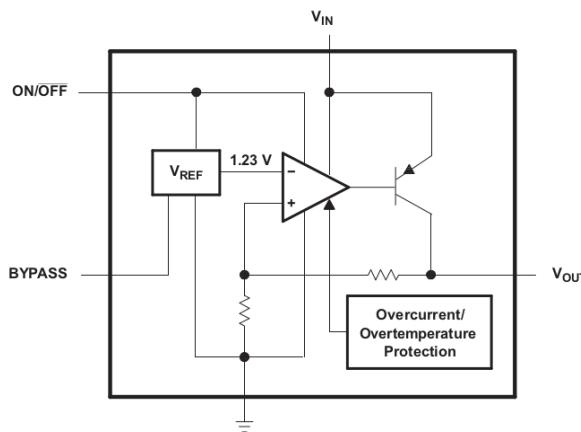


Figure 3.4: The LP2985A-10DBVR LDO regulator block diagram.

## Pinout

Table 3.10: TPS7A91 pinout.

#	Pin	Description
1, 2	OUT	Regulated output voltage. A $10\mu\text{F}$ or greater capacitor should be connected between this pin and ground.
3	FB	Feedback pin. The divided output voltage is compared against a 0.8V reference.
4	GND	Ground.
5	PG	A power good indicator that can signal to downstream devices when voltage is within some range of the desired level. The design doesn't use this and so the pin has been left floating.
6	SS_CTRL	Soft-start control pin. Tying this pin to the input voltage increases the soft-start charging current to $100\mu\text{A}$ , which reduces the startup time. This pin could have also been tied to ground if a slower startup time was desired.
7	EN	Enable. Tying this to the input voltage unconditionally enables this device.

---

8	NR/SS	Noise-reduction pin. A capacitor can be placed between this pin and ground to reduce output noise. This effectively acts like an RC filter. A capacitance between 10nF and 1 $\mu$ F is recommended. I've used 100nF. This pin also limits inrush current.
9, 10	IN	Input voltage pin. A bypass capacitor of 10 $\mu$ F or more is required.

---

## Component Selection

MLCC capacitors are recommended for input and output bypassing due to their low ESR, with preferences given to X5R and X7R types. Additionally, they should be derated by at least 50%. The recommended capacitance is 10 $\mu$ F or greater at the input and output. I've used 10 $\mu$ F at the input and the same at the output, with the option for another capacitor at the output if needed. The 10nF feed-forward capacitor between FB and OUT is recommended by the datasheet to improve noise and PSRR performance. The voltage divider resistors are specifically mentioned in the datasheet to get the appropriate output values.

## PCB Layout

The recommended PCB layout is shown in Figure 3.6.

### 3.1.7 TPS7A7001DDA LDO Regulator

#### Description

The TPS7A7001DDA supports low dropouts up to 2A, which is important since it must accommodate a maximum current draw of 1.1A (see Table 3.1.3).

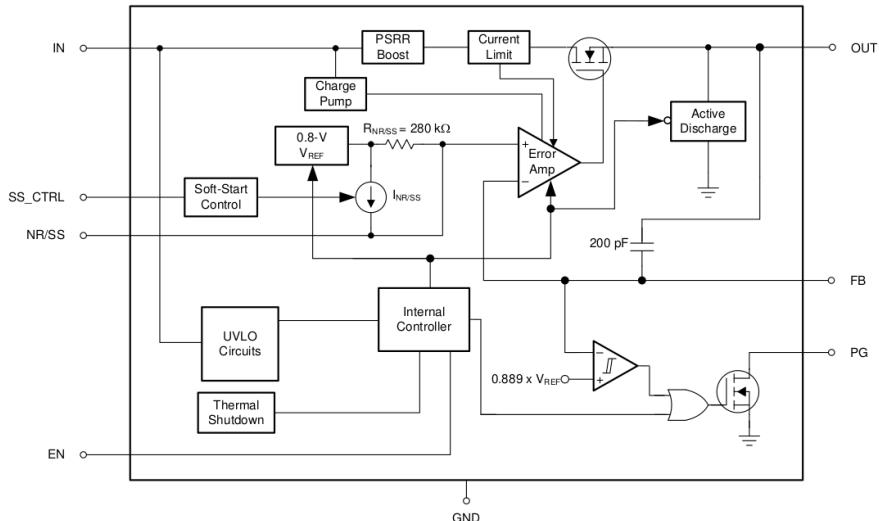
#### Pinout

Table 3.11: TPS7A7001DDA pinout.

#	Pin	Description
1, 4, 5	NC	No connect.
2	EN	Enable input. Connected to IN since unused.
3	IN	Input voltage.
6	OUT	Regulated voltage output.
7	FB	Feedback, compared to 0.5V reference.
8	GND	Ground.
9	EP	Exposed pad. Tied to ground.

#### Component Selection

An input capacitor of 1 – 10 $\mu$ F and with low ESR is recommended. The output capacitor should be ceramic and can be anywhere from 4.7 – 47 $\mu$ F. I've chosen 10 $\mu$ F with the option for an additional capacitor if needed. The voltage divider resistors are specifically recommended by the datasheet.



**Figure 3.5: TP7A91 block diagram.**

## PCB Layout

The recommended PCB layout is provided in [3.7](#).

### 3.1.8 LP5907MFX-3.0

#### Description

The LP5907MFX-3.0 is a low-noise and dropout voltage regulator used to power the VCO and reception-side amplifiers. It has a current rating of 250mA which is more than the maximum current load of the downstream components.

#### Pinout

**Table 3.12: LP5907MFX-3.0 pinout.**

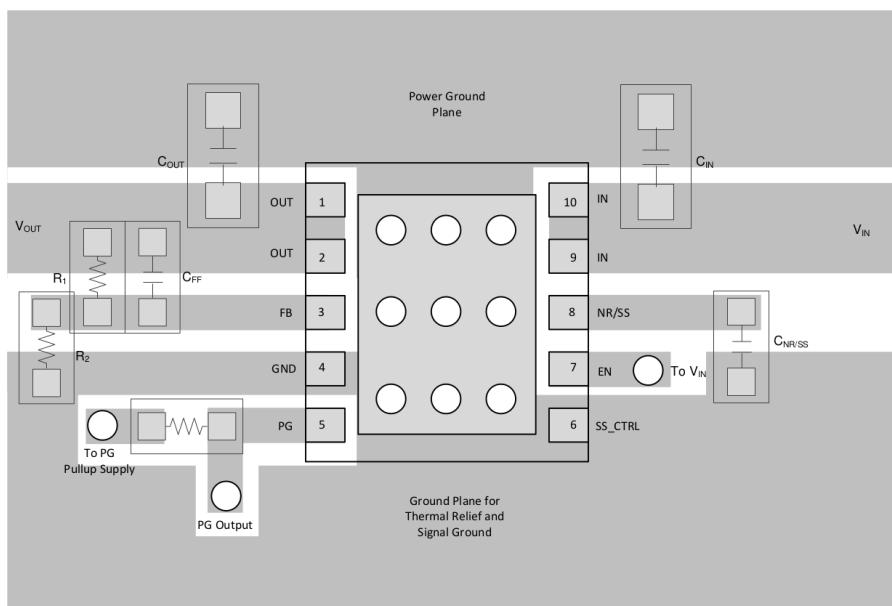
#	Pin	Description
1	IN	Voltage input.
2	GND	Ground.
3	EN	Enable. Tied to IN to keep the device unconditionally on.
4	NC	No connect.
5	OUT	1.8V output.

#### Component Selection

The input and output capacitors should be  $1 - 10\mu\text{F}$ , with the input capacitor at least as large as the output capacitor. I've used  $10\mu\text{F}$  capacitors in this design.

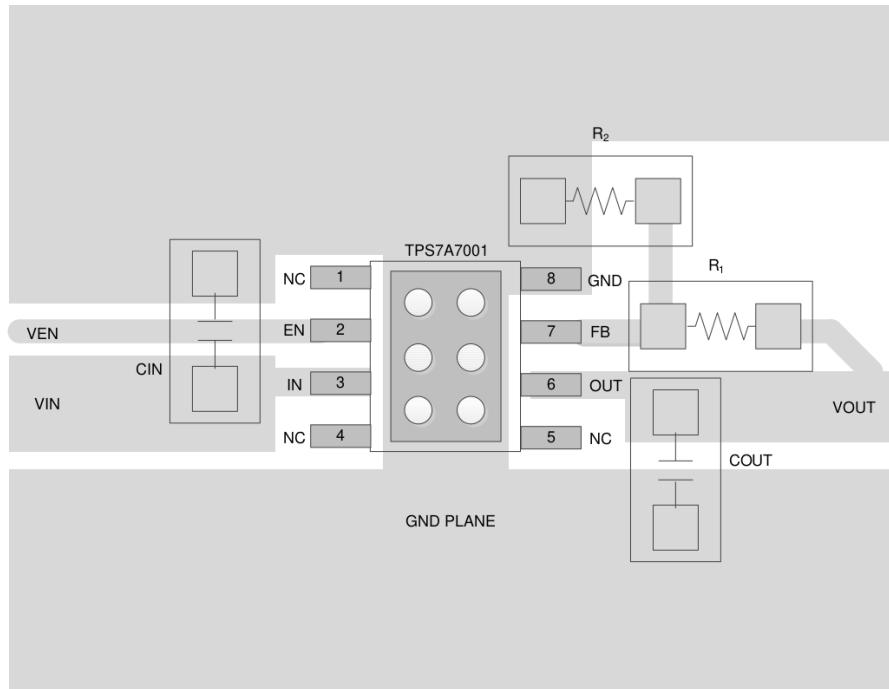
#### PCB Layout

The recommended PCB layout is shown in [Figure 3.8](#).



○ Denotes vias used for application purposes

**Figure 3.6: TPS7A91 recommended PCB layout.**



**Figure 3.7: TPS7A7001DDA recommended PCB layout.**

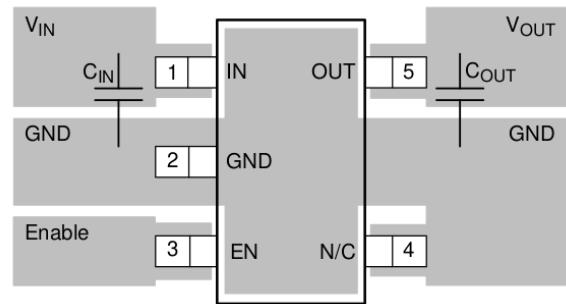


Figure 3.8: LP5907MFX-3.0 recommended PCB layout.

## 3.2 Top

*schematic*

## 3.3 Overview

The top-level schematic has two purposes: it instantiates the various submodules in the design and generates a 40MHz clock signal that is distributed to the **FPGA**, **ADC** and frequency synthesizer.

### 3.3.1 KT2520K 40MHz TXCO

The KT2520K crystal oscillator outputs a 40MHz clipped sine wave. It requires a DC-blocking capacitor  $\geq 1\text{nF}$  at the output since one is not included internally.

### 3.3.2 NC7S04 Inverter

Using a feedback resistor between the inverter's output and input biases the input to  $V_{CC}/2$ , thus keeping the duty cycle near 50%.

### 3.3.3 NB3N551 1:4 Clock Buffer

The clock buffer duplicates the square wave input clock signal into 3 identical clock outputs that are fed to the **FPGA**, **ADC** and frequency synthesizer.

## 3.4 TX

*schematic*

### 3.4.1 ADF4158 Frequency Synthesizer

#### Overview

The [ADF4158](#) is a 6.1GHz fractional-n frequency synthesizer. A block diagram describing its functionality is shown in Figure 3.9.

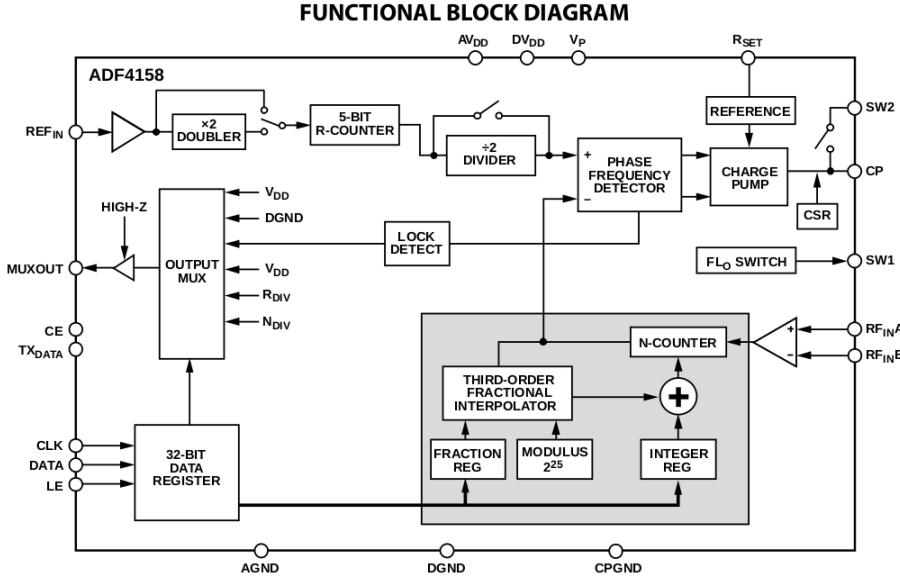


Figure 3.9: ADF4158 block diagram.

The ADF4158 relies on an external VCO, described in § 3.4.2, whose output frequency is given by Equation 3.4. The output resolution is  $f_{\text{RES}} = f_{\text{PFD}}/2^{25}$  (see Equation 3.4 and its corresponding parameter table for an explanation of  $f_{\text{PFD}}$ ). The  $2^{25}$  arises from the FRAC value (set in registers 0 and 1) which is given 25 bits.

$$\begin{aligned} \text{RF}_{\text{OUT}} &= f_{\text{PFD}} \times (\text{INT} + (\text{FRAC}/2^{25})) \\ f_{\text{PFD}} &= \text{REF}_{\text{IN}} \times [(1 + D) / (R \times (1 + T))] \end{aligned} \quad (3.4)$$

---

Parameter/Variable	Description
RF <sub>OUT</sub>	The VCO's output frequency. This is the frequency that's amplified for transmission.
$f_{\text{PFD}}$	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.
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.

Parameter/Variable	Description
REF <sub>IN</sub>	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 <sub>IN</sub> 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 3.10. 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 3.11. The equations governing these parameters are given in Equation 3.5. 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 3.6 is used to derive this.

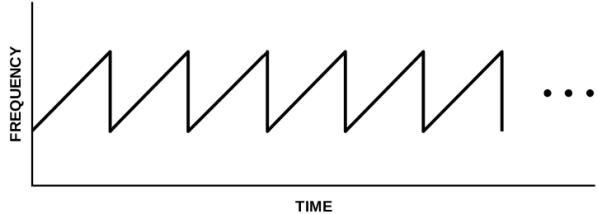
$$f_{DEV} = (f_{PFD}/2^{25}) \times (\text{DEV} \times 2^{\text{DEV\_OFFSET}}) \quad (3.5)$$

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

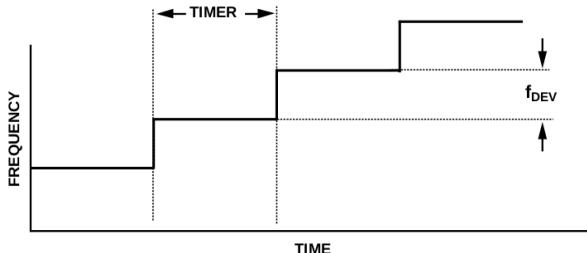
Parameter/Variable	Description
$f_{DEV}$	The frequency deviation for each frequency jump during ramp.
$f_{PFD}$	The input frequency to the PFD post prescaling. In our case this is 20MHz.
DEV	A 16-bit value set in register 5.
DEV_OFFSET	A 4-bit word set in register 5.
Timer	The time between each frequency hop.
CLK <sub>1</sub>	A 12-bit clock divider set in register 2.
CLK <sub>2</sub>	Another 12-bit clock divider set in register 4.

$$\text{Delay} = t_{PFD} \times \text{CLK}_1 \times \text{Delay Start Word} \quad (3.6)$$

Parameter/Variable	Description
Delay	The delay between bursts. This is shown graphically in Figure 3.12.
$f_{PFD}$	The input frequency to the PFD post prescaling. In our case this is 20MHz.
CLK <sub>1</sub>	A 12-bit clock divider set in register 2.
Delay Start Word	A 12-bit value set in register 7.



**Figure 3.10: Sawtooth ramp.**



**Figure 3.11: Waveform timing.**

## 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 3.16: 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 3.17: 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.

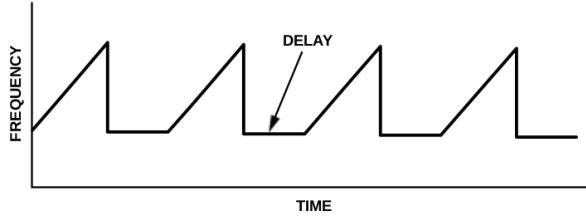


Figure 3.12: Delay between ramps for sawtooth mode.

Table 3.18: R-DIVIDER REGISTER(R2) MAP

Bits	Mnemonic	Value	DESCRIPTION
3–14	CLK <sub>1</sub> Divider	10	One of the determinants of the duration of a timestep in waveform generation. See § 3.4.1 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.
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 3.19: 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.

**Table 3.19: FUNCTION REGISTER(R3) MAP**

Bits	Mnemonic	Value	DESCRIPTION
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	LDP	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 3.20: TEST REGISTER(R4) MAP**

Bits	Mnemonic	Value	DESCRIPTION
3-6	Reserved	0	All reserved bits are set to 0.
7-18	CLK <sub>2</sub> Divider	1	This is used to set the timeout interval in ramp generation. See § 3.4.1 for more information.
19-20	CLK DIV Mode	3	This enables ramp divider mode, which specifies that CLK <sub>1</sub> and CLK <sub>2</sub> 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 <a href="#">AN-1154 Application Note</a> .

**Table 3.20: TEST REGISTER(R4) MAP**

Bits	Mnemonic	Value	DESCRIPTION
25	Reserved	0	All reserved bits are set to 0.
26-30	$\Sigma - \Delta$ 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 3.21: 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 § 3.4.1 for more information.
19-22	Deviation Offset Word	4	This is also used to set the size of successive frequency jumps during ramp. See § 3.4.1 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.
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 3.22: 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 § 3.4.1 on waveform generation.

**Table 3.22: STEP REGISTER (R6) MAP**

Bits	Mnemonic	Value	DESCRIPTION
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 3.23: 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	

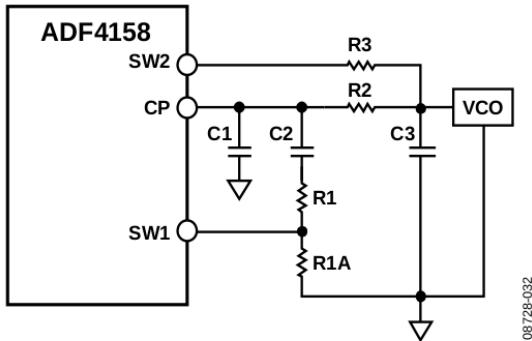
**FIXME** I need to finish describing this component along with all other component on this sheet.

**Table 3.24: All ADF4158 pin connections.**

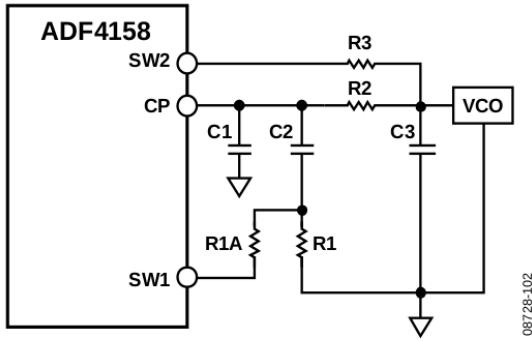
PIN	Mnemonic	DESCRIPTION
1	CPGND	The charge pump ground, which we set to the same ground as the rest of the circuit.
2, 3	AGND	Prescaler ground; also set to common ground.
4	RF <sub>IN</sub> B	The complement to the VCO output. We couple this to ground with a small capacitor: in this case, 10pF.
5	RF <sub>IN</sub> A	The VCO output. This goes through prescaling and then becomes the PLL input signal.
6, 7, 8	AV <sub>DD</sub>	The positive power supply for the RF part of the PLL. This takes 3.3V.
9	REF <sub>IN</sub>	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	TXDATA	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.

**Table 3.24: All ADF4158 pin connections.**

PIN	Mnemonic	DESCRIPTION
18	SDV <sub>DD</sub>	Power supply pin for the $\Sigma - \Delta$ modulator. We also set this to 3.3V.
19	DV <sub>DD</sub>	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 3.13. However, our design does not use either of these topologies.
22	V <sub>P</sub>	The charge pump power supply, which we set to 5V.
23	R <sub>SET</sub>	This pin sets the maximum charge pump current according to the equation $I_{CP\max} = 25.5/R_{SET}$ . We use $5.49k\Omega$ 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.



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



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

**Figure 3.13: Fast lock topologies.**

## 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.

### 3.4.2 HMC431LP4RF VCO

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

### 3.4.3 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.

### 3.4.4 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 ???. 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.

## 3.5 USB

In order to configure the FPGA we load the configuration bit stream onto the board using a USB 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 3.14. 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.

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.

### 3.5.1 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.

### 3.5.2 93LC46B 1k EEPROM

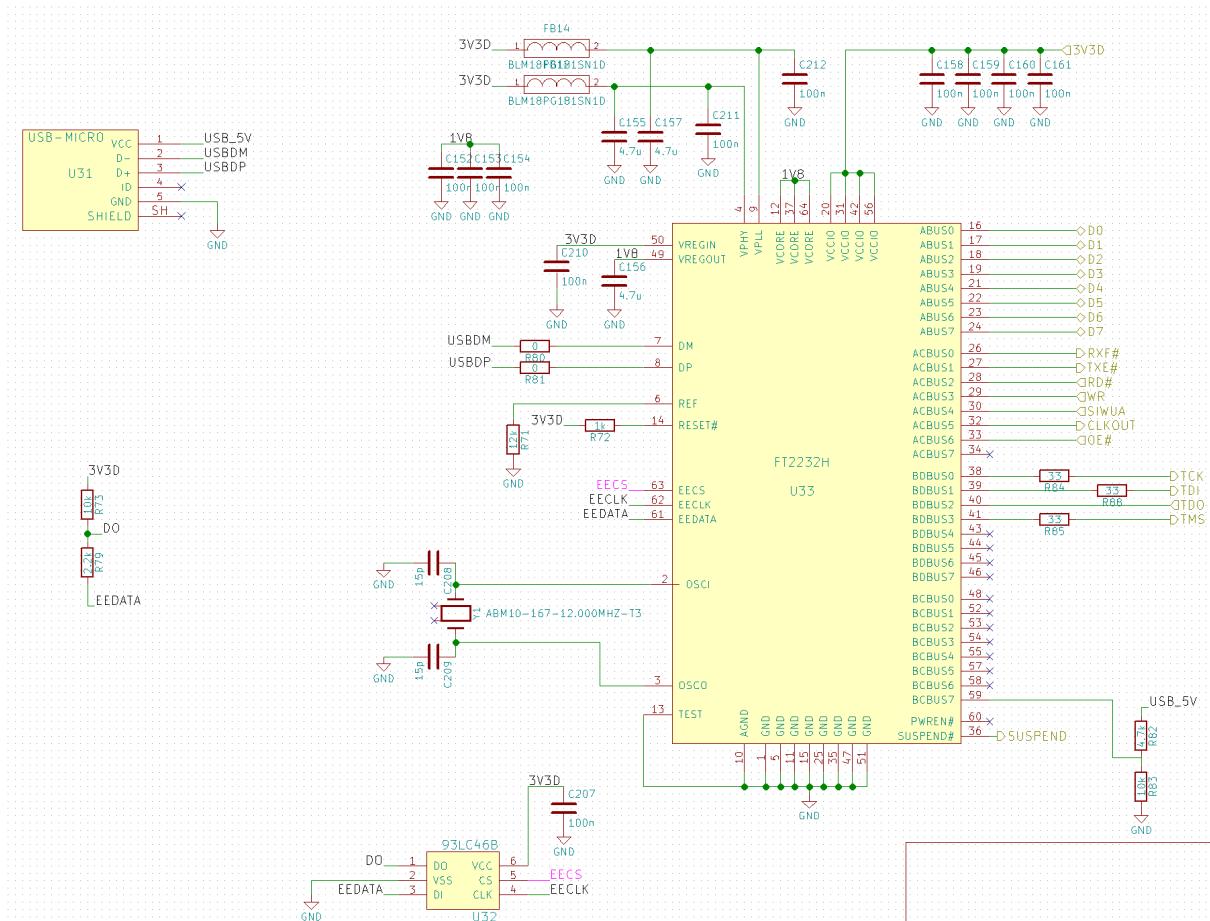


Figure 3.14: The configuration bit stream is loaded onto the FPGA using a USB-JTAG interface.

## 3.6 FPGA

### 3.6.1 Overview

The FPGA has two main roles. It performs digital signal processing on data sent from the ADC and relays the processed data back to a host computer via an [FT2232H](#) which acts as a FIFO-USB interface. Secondly, the FPGA is used to configure other components on the board. In order to program the FPGA, we use a host computer with a JTAG interface via the FT2232H.

The FPGA stores and loads its configuration from an SPI flash memory device, [W25Q32JV](#).

### 3.6.2 XC7A15T-FTG256 Xilinx FPGA

#### Documents

**Table 3.25: Important documents.**

URL	Description
<a href="#">Artix-7 FPGA family</a>	The main page that contains a collection of documents about the 7 series devices.
<a href="#">Package Pinout</a>	The various packages that the device comes in, along with the associated pin definitions.
<a href="#">Overview</a>	Contains general device capability information, such as number of logic cells, DSP blocks, etc.
<a href="#">PCB Design Guide</a>	Contains information about bypassing and other design considerations.
<a href="#">XADC</a>	Contains information about the FPGA's internal ADC.
<a href="#">Configuration Guide</a>	Describes how to configure the FPGA with a bitstream.

#### Power Supply Requirements

The FPGA requires that the VCCINT and VCCBRAM voltages be supplied first, followed by VCCAUX and then by VCCO. Additionally, all this must be done between 0.2 and 50ms. The required voltage levels are displayed in Table 3.6.2.

**Table 3.26: XC7A15T-FTG256 power supply requirements.**

Voltage Pin	Min	Typ	Max
V <sub>CCINT</sub>	0.95	1.00	1.05
V <sub>CCBRAM</sub>	0.95	1.00	1.05
V <sub>CCHAUX</sub>	1.71	1.80	1.89
V <sub>CCO</sub>	1.14	-	3.465

#### Bypassing

The PCB Design Guide above specifies fairly stringent requirements for bypassing and should be adhered to.

## Pinout

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).

**Table 3.27: 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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

**Table 3.27: All FPGA pin connections in alphabetical order.**

PIN	DESCRIPTION
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.
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.

### 3.6.3 W25Q32JV Flash Memory

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 3.6.2 contains a list of all FPGA pins and their connections.

## 3.7 ADC

### 3.7.1 Overview

The ADC is used to digitize signals amplified by the IF [ADA4940-2](#) differential amplifiers before sending them to the [FPGA](#) for processing.

### 3.7.2 LTC2292

#### Description

The [LTC2292](#) is a 40MHz, 12-bit differential input ADC. The schematic for this device is shown in Figure 3.15. 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 3.16. Table 3.7.2 contains a list of all the ADC pins and their connections.

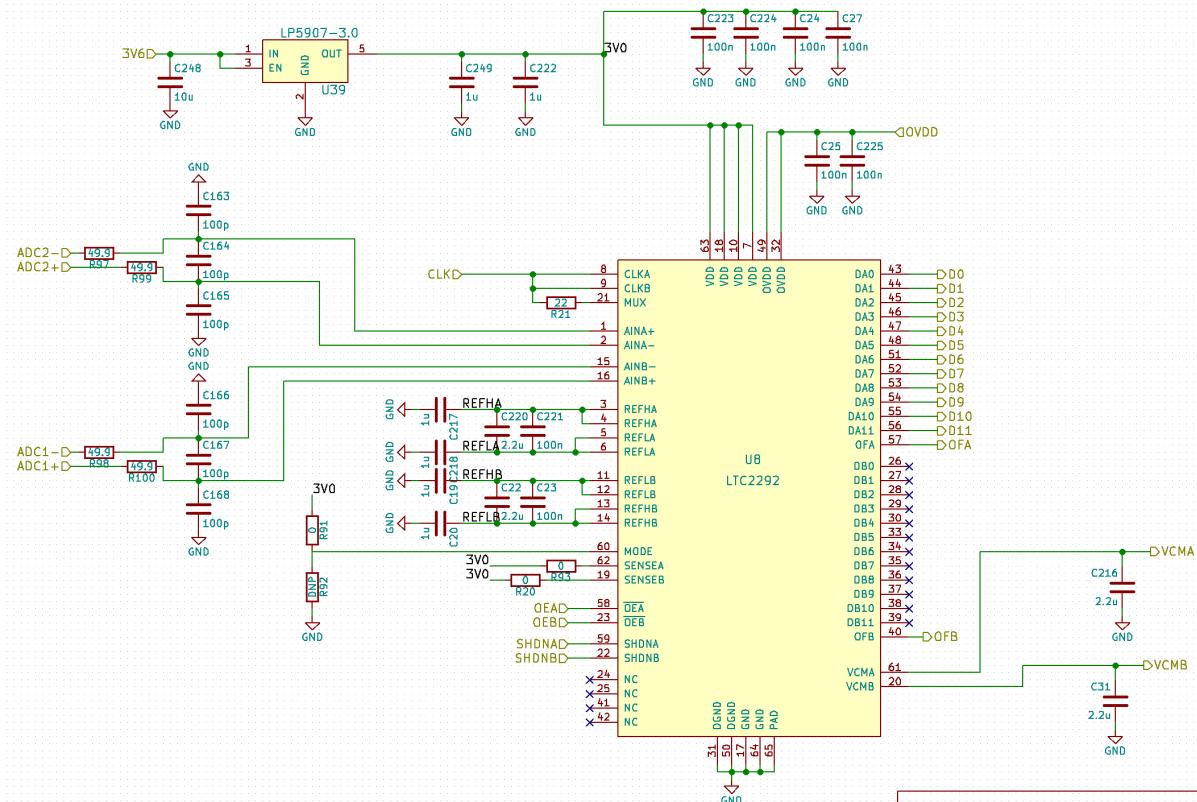


Figure 3.15: The LTC2292ADC schematic.

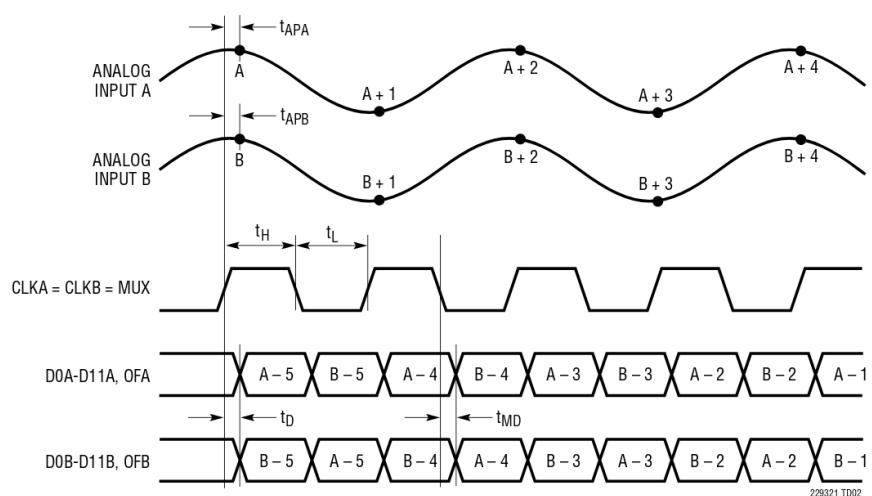
Table 3.28: All LTC2292 ADC pin connections in logical groupings.

LABELS	PIN #s	DESCRIPTION
--------	--------	-------------

VDD	7, 10, 18, 63	The power supply required is 3.0V and each pin requires its own $0.1\mu\text{F}$ 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.
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 $1.5\text{V} \pm 1\text{V}$ . $1.5\text{V}$ 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.
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 $0.1\mu F$ 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. <b>{STARTINCOMPLETE}</b> 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 $0.1\mu F$ 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 $49.9\Omega$ differs from the suggested resistance of $25\Omega$ . Lastly, the negative output of the differential amplifier for channel A is feeding into the positive input line <b>{END INCOMPLETE}</b> .

---

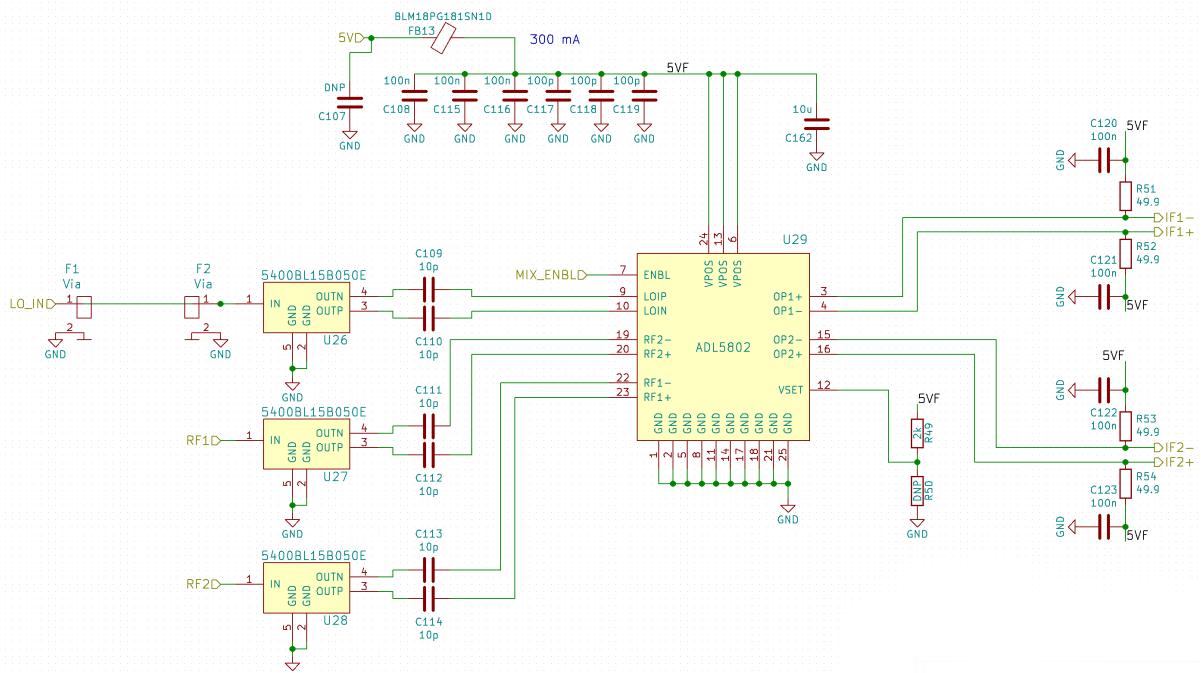


**Figure 3.16: Multiplexed digital output bus timing for the LTC2292 ADC.**

## 3.8 Mixer

### 3.8.1 ADL5802

The mixer schematic shown in Figure 3.17 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 3.17:** The mixer schematic.

The RF and LO input interfaces are designed for a differential input impedance of  $50\Omega$ . 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.

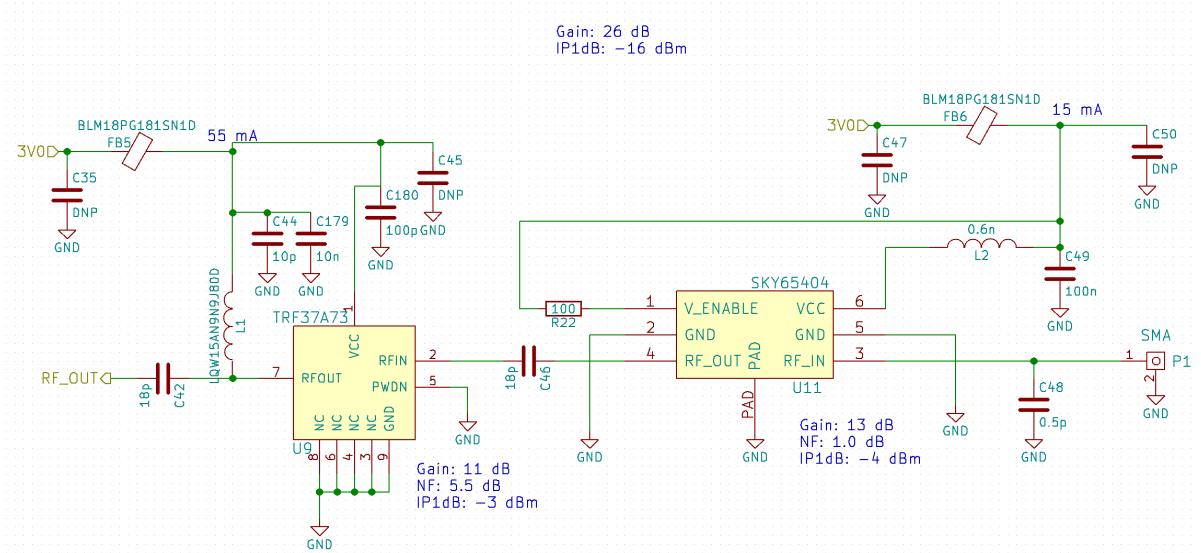
## 3.9 IF

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.

### 3.9.1 ADA4940-2 IF Differential Amplifier

### 3.10 RX1 & RX2

The sheets RX1 and RX2 are identical. They each consist of two amplifiers connected in series, shown in Figure 3.18. 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 3.18:** 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 4

## PCB

### 4.1 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 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12).

### 4.2 Component Layout Considerations

#### 4.2.1 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.

All components should be placed sufficiently close to the TPS5420D buck converter, except for the trace between the voltage divider and VSNS which should be given a sufficiently wide berth from the PH trace.

#### 4.2.2 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.

#### 4.2.3 RX1 / RX2

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

### 4.3 RF Impedance Matching

**Fixme Fatal!**

**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  $50\Omega$  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  $50\Omega$  coaxial cable. All RF inputs and outputs for components should match this  $50\Omega$  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  $50\Omega$  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).

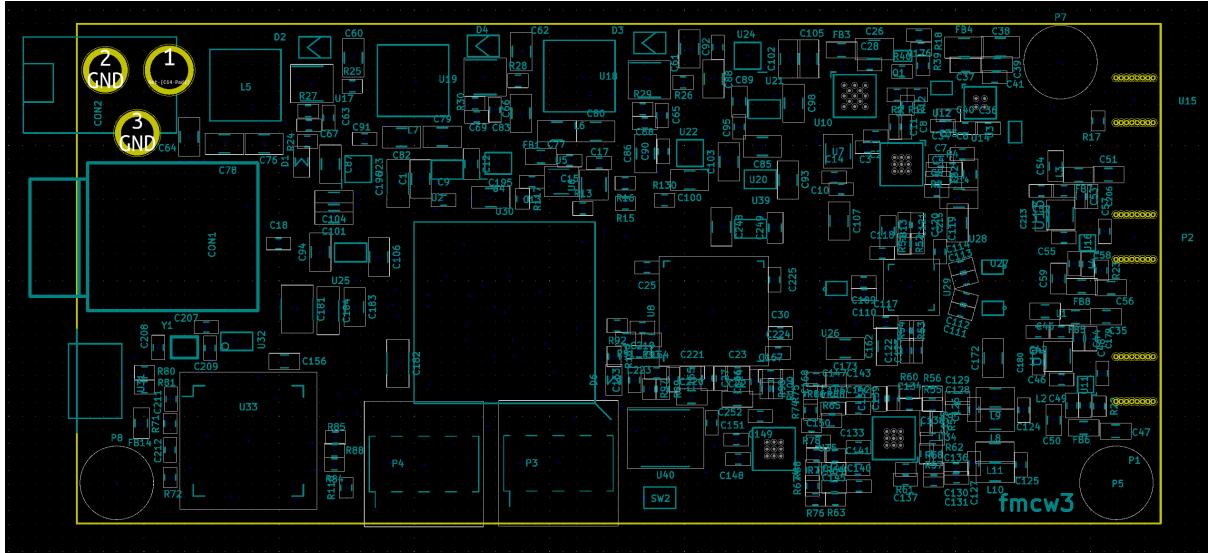


Figure 4.1: Board layout. The top left of the board is where the power source (barrel jack) is connected. The output of the 12V power source is connected to a power plane at the top left-middle of the board, that feeds into the buck converters. Large inductors connected to the buck converters are placed 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.

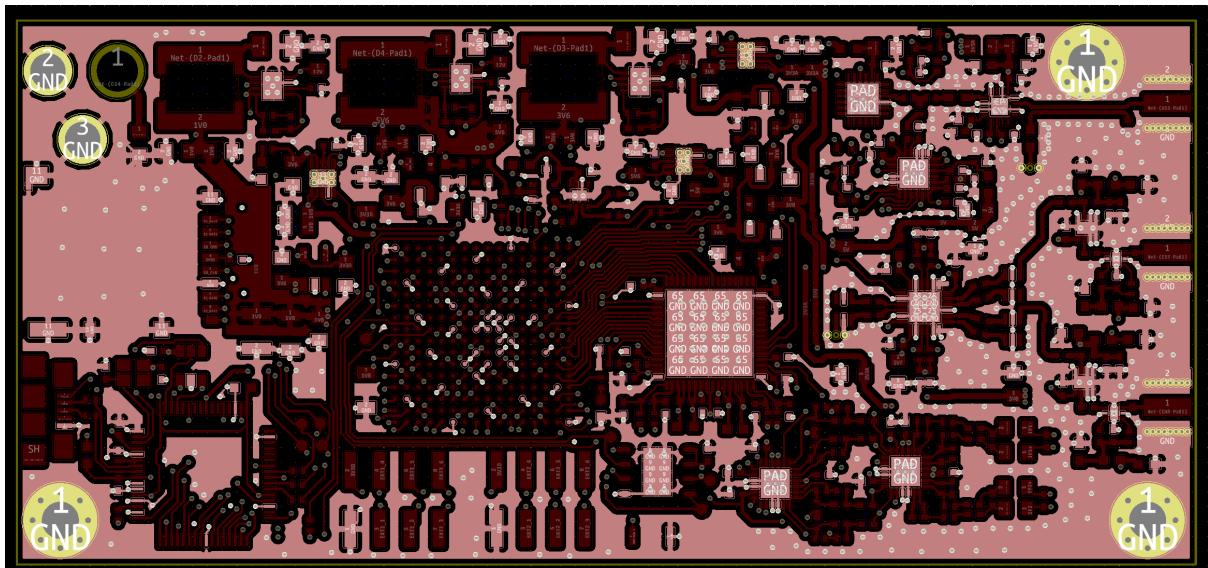


Figure 4.2: A significant portion of layer 1 is a ground plane and the other large part of it is signal traces connecting components. There is also a small 1V power plane toward the upper left not highlighted here.

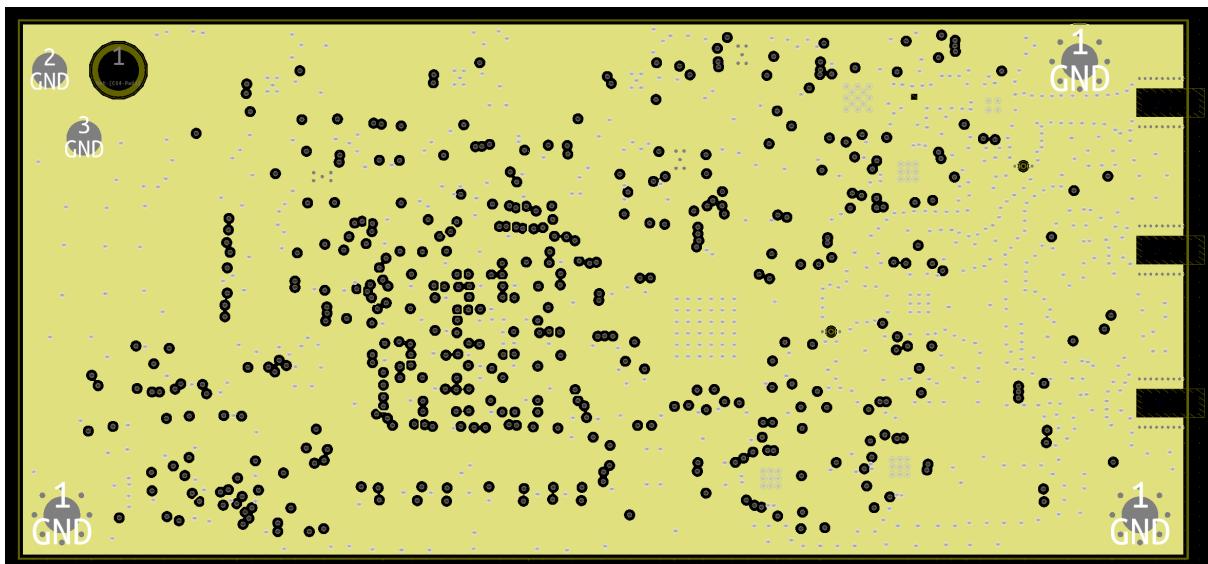


Figure 4.3: All of layer 2 is a ground plane.

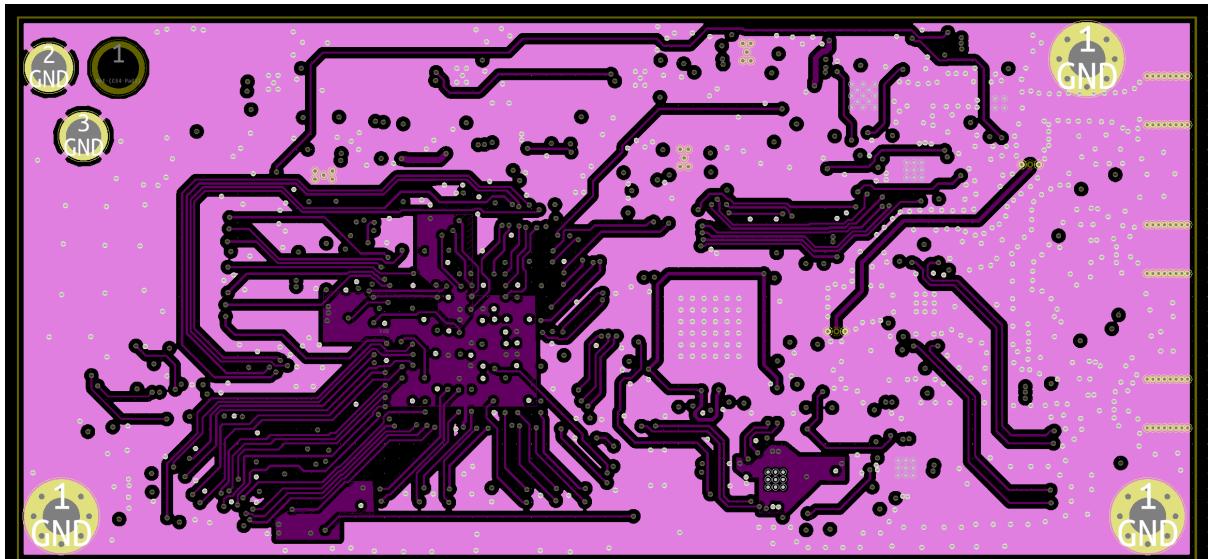


Figure 4.4: Most of layer 3 is a ground plane but there are also a substantial number of trace connecting to the FPGA. It does also contain 1V an 3.3V power planes.

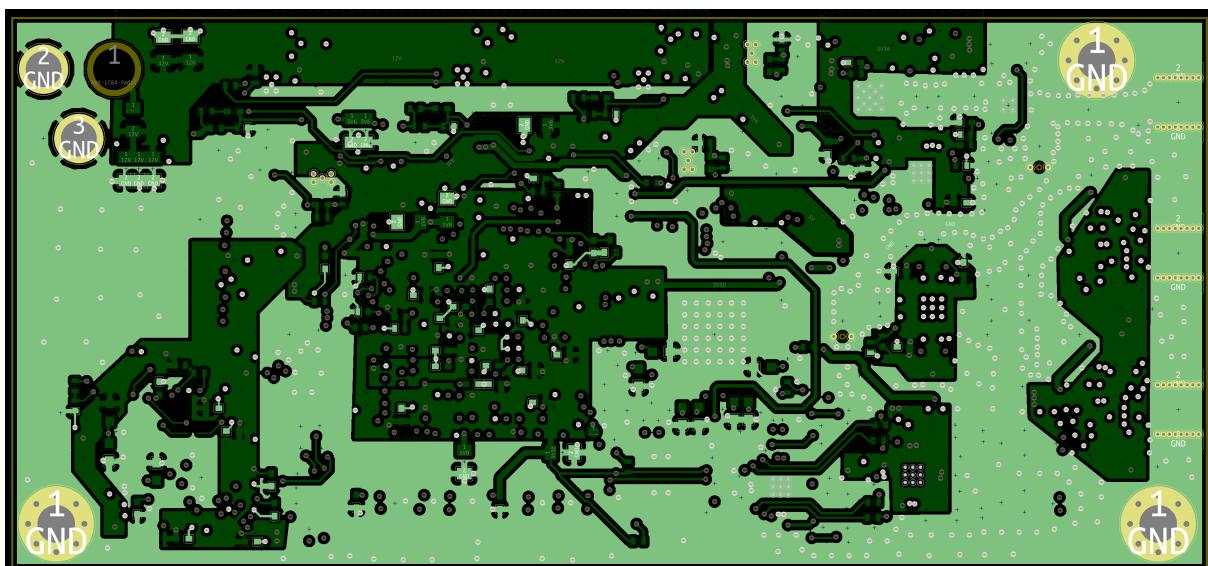


Figure 4.5: The 4th layer of the PCB contains a large ground plane with many vias interspersed.

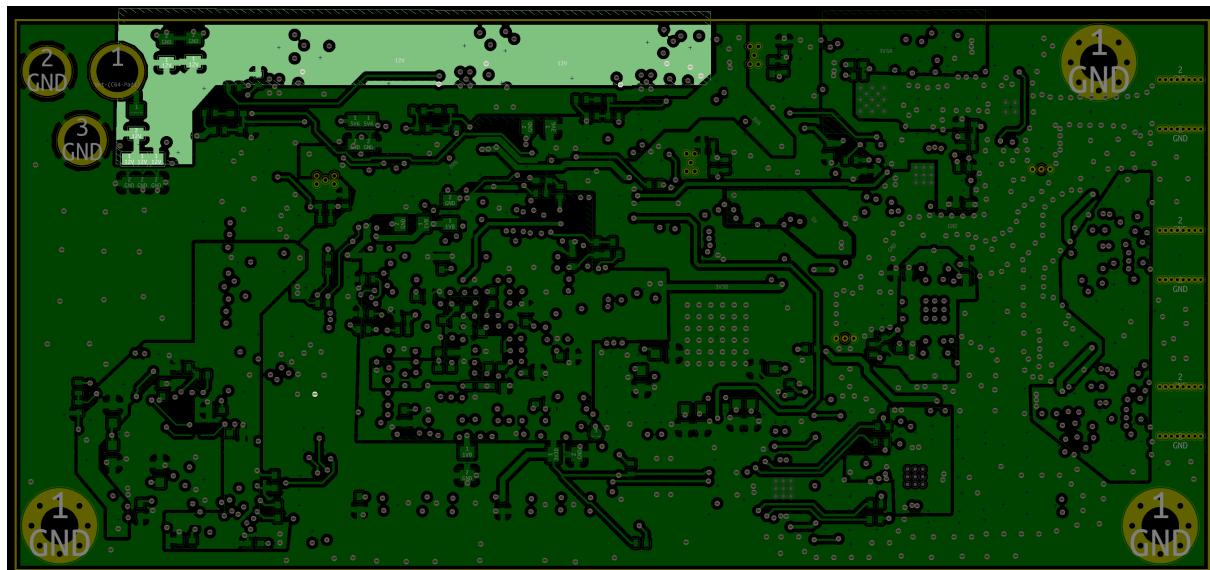


Figure 4.6: The 12V power plane is located at the top of the board adjacent to the D barrel jack and the buck converters it feeds.

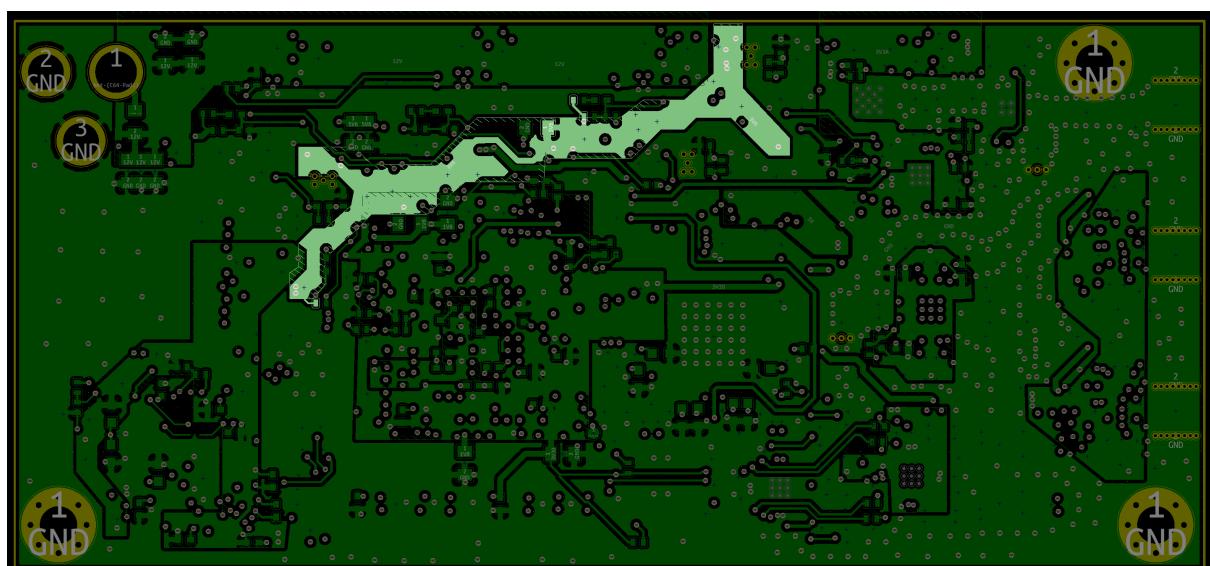


Figure 4.7: There is a 3.6V power plane that is the output of one of the buck converter and is use as the input to several linear regulators that output 1.8V 3.0V and 3.3V.

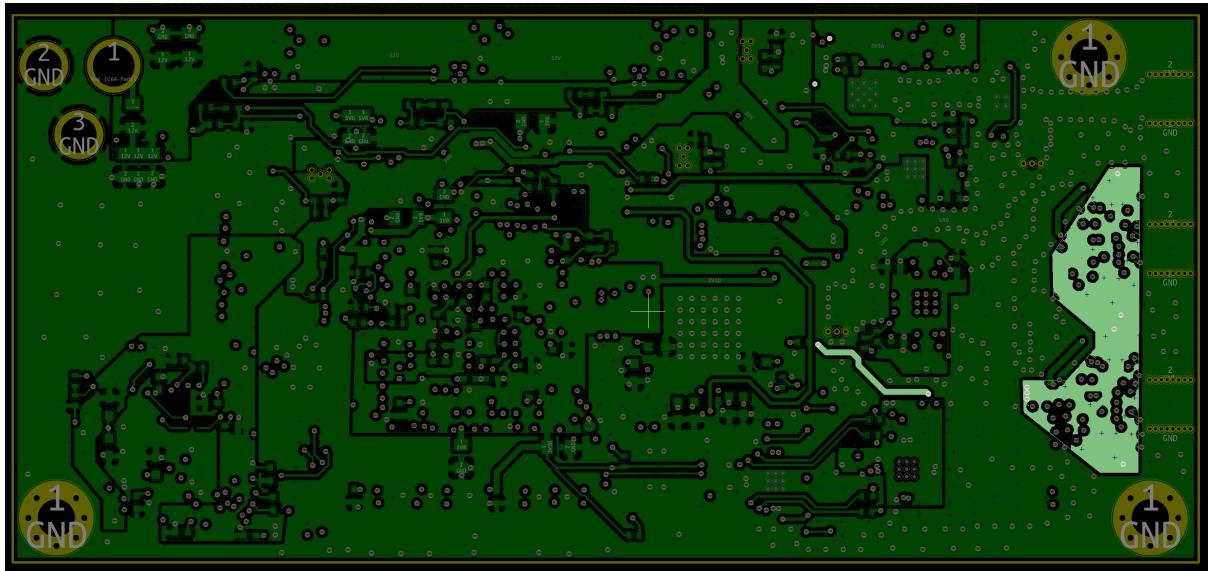


Figure 4.8: A 3V power plane is used to power components near the right side of the board (the right side of the board is where signal transmission and reception occur that amplify receive signals so they can be mixed with the transmitted signal and fed into the ADC before FPGA processing). Notably, the 3V power plane is located near the components but far from the linear regulator that outputs it.

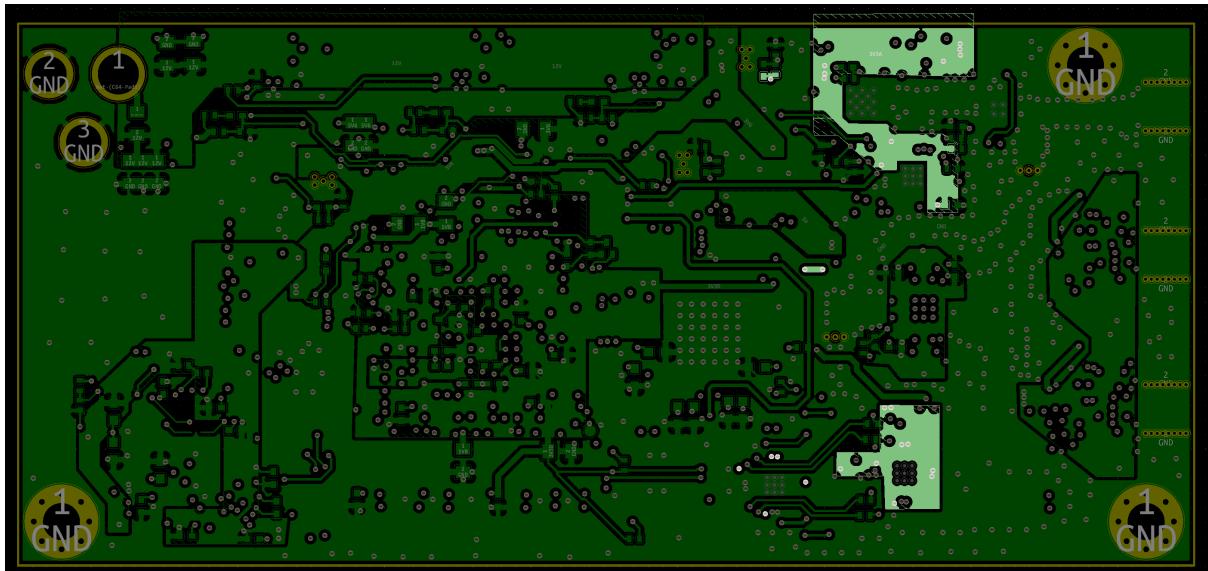


Figure 4.9: A 3.3V power plane is used to power several components for signal transmission, such as a frequency synthesizer, and RF amplifier. These, along with all other reception and transmission circuitry, are located on the right side of the board.

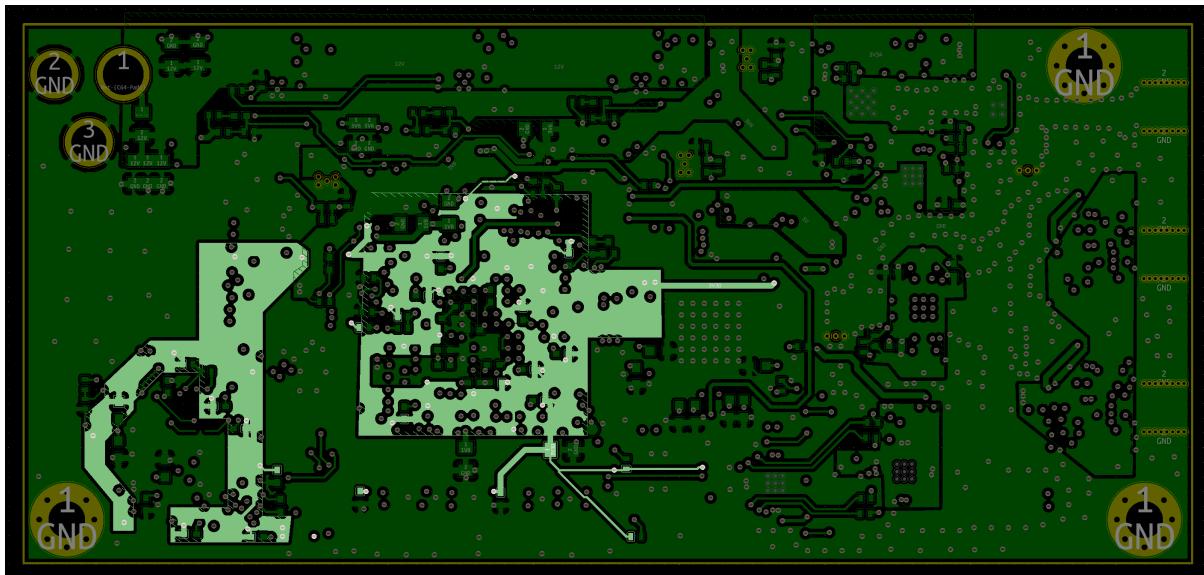


Figure 4.10: Another 3.3V power plane is used as one of the power inputs to the FPGA as well as to a USB-to-UART device (a configuration bit stream is sent from a host computer through a USB cable to the PCB where this device converts it into the proper UART format to be transmitted to the FPGA) and an ADC (the ADC converts the mixed transmitted-received signals to digital and then sends them to the FPGA for processing).

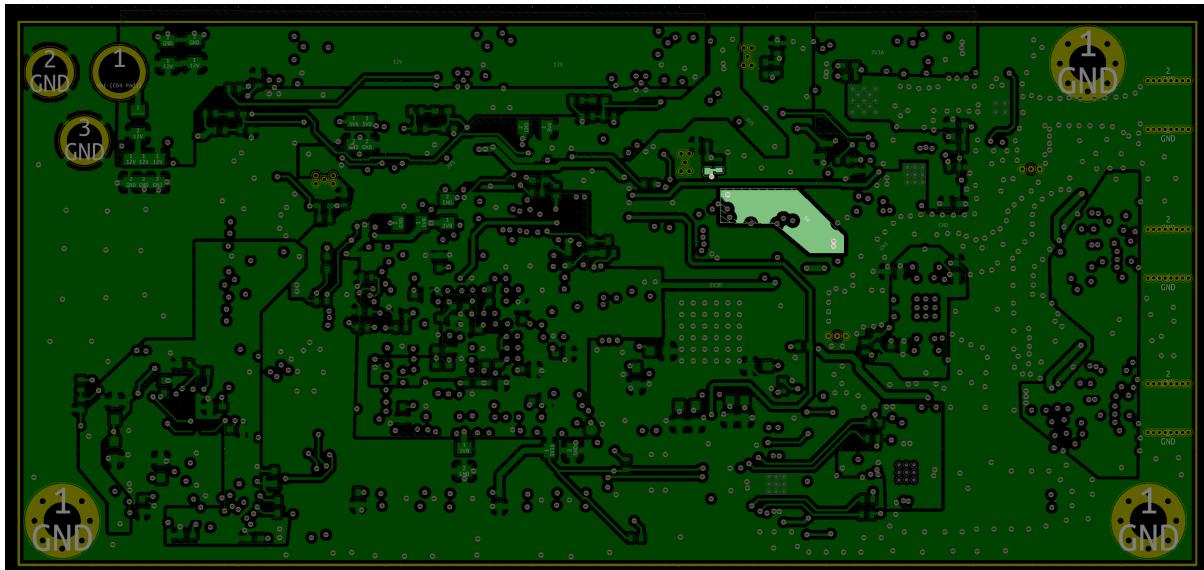


Figure 4.11: A 5V power plane powers one of the inputs to the frequency synthesizer for transmission.

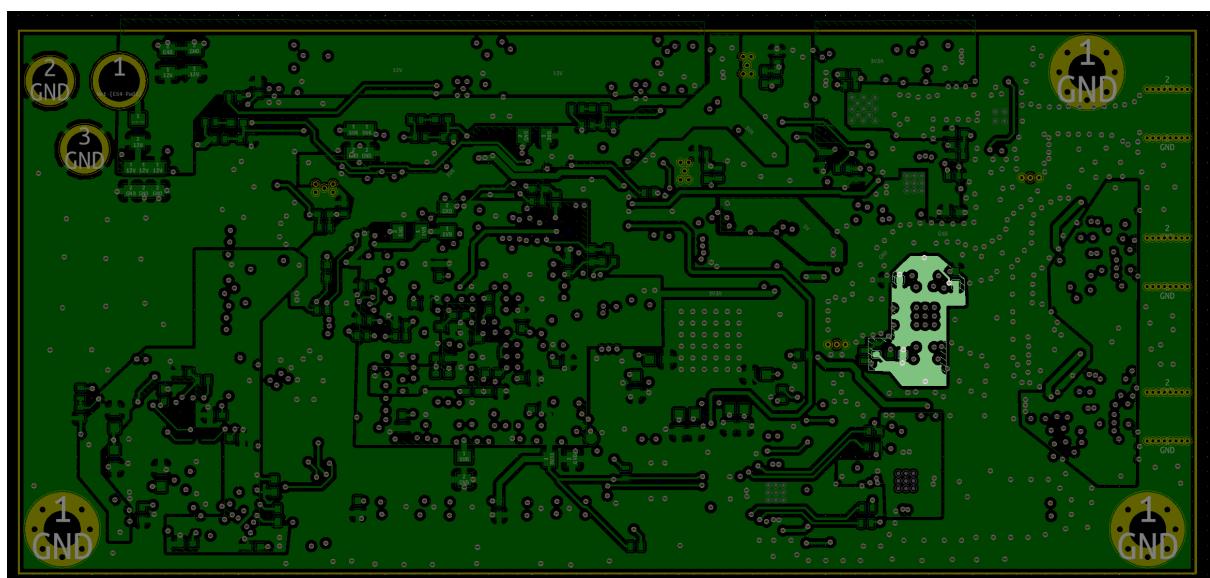


Figure 4.12: Another 5V power plane is used for to power devices after going through a ferrite bead.

# Chapter 5

## 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.

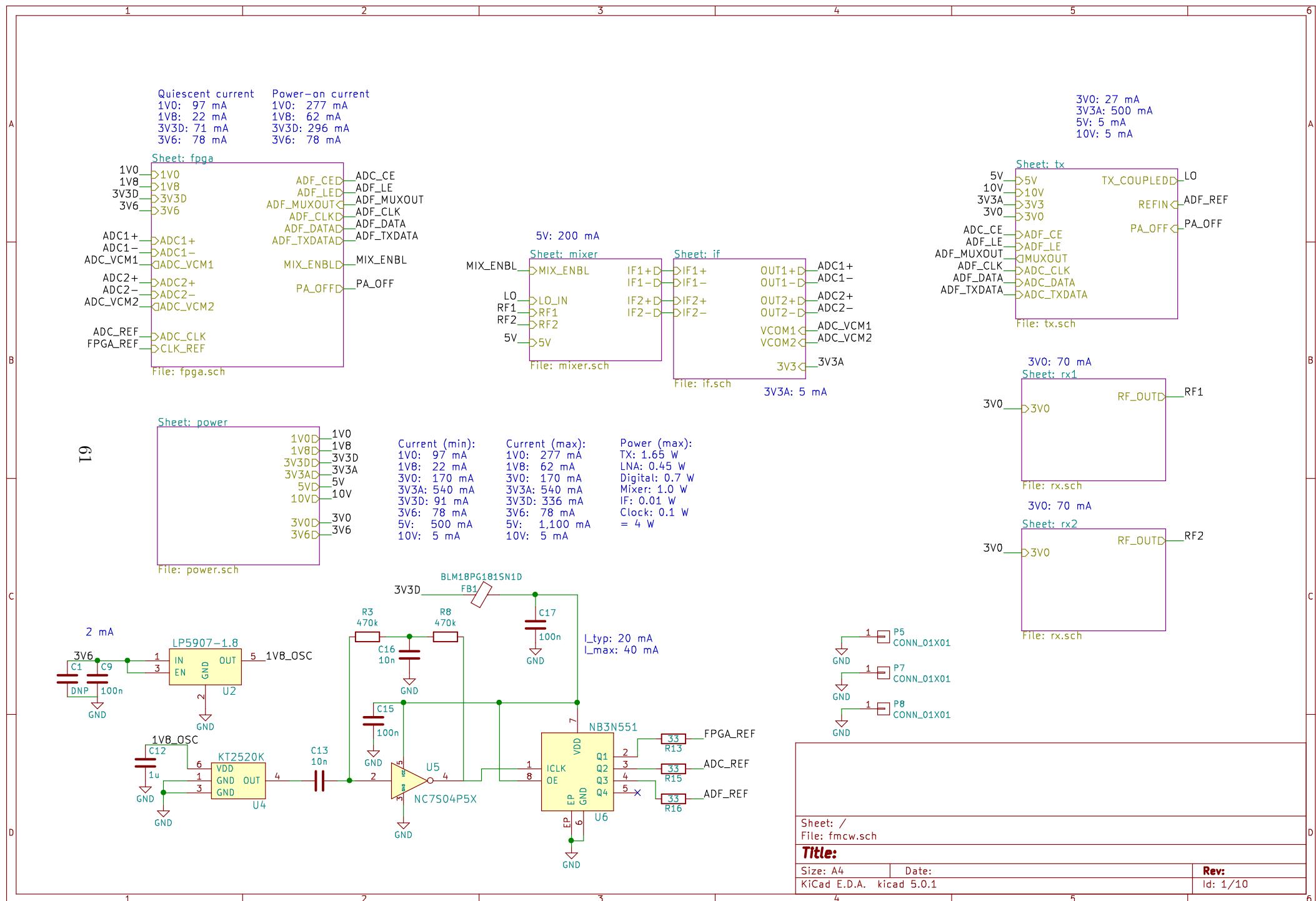
[This antenna](#) looks like it would be good for transmission. [This RF Elements](#) antenna looks even better, but is more than I'm willing to pay (and its gain is lower, however). [The Triple Feed Patch Array Antenna](#) is a great deal for the price and is based off a clever open-source design, [here](#). 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.

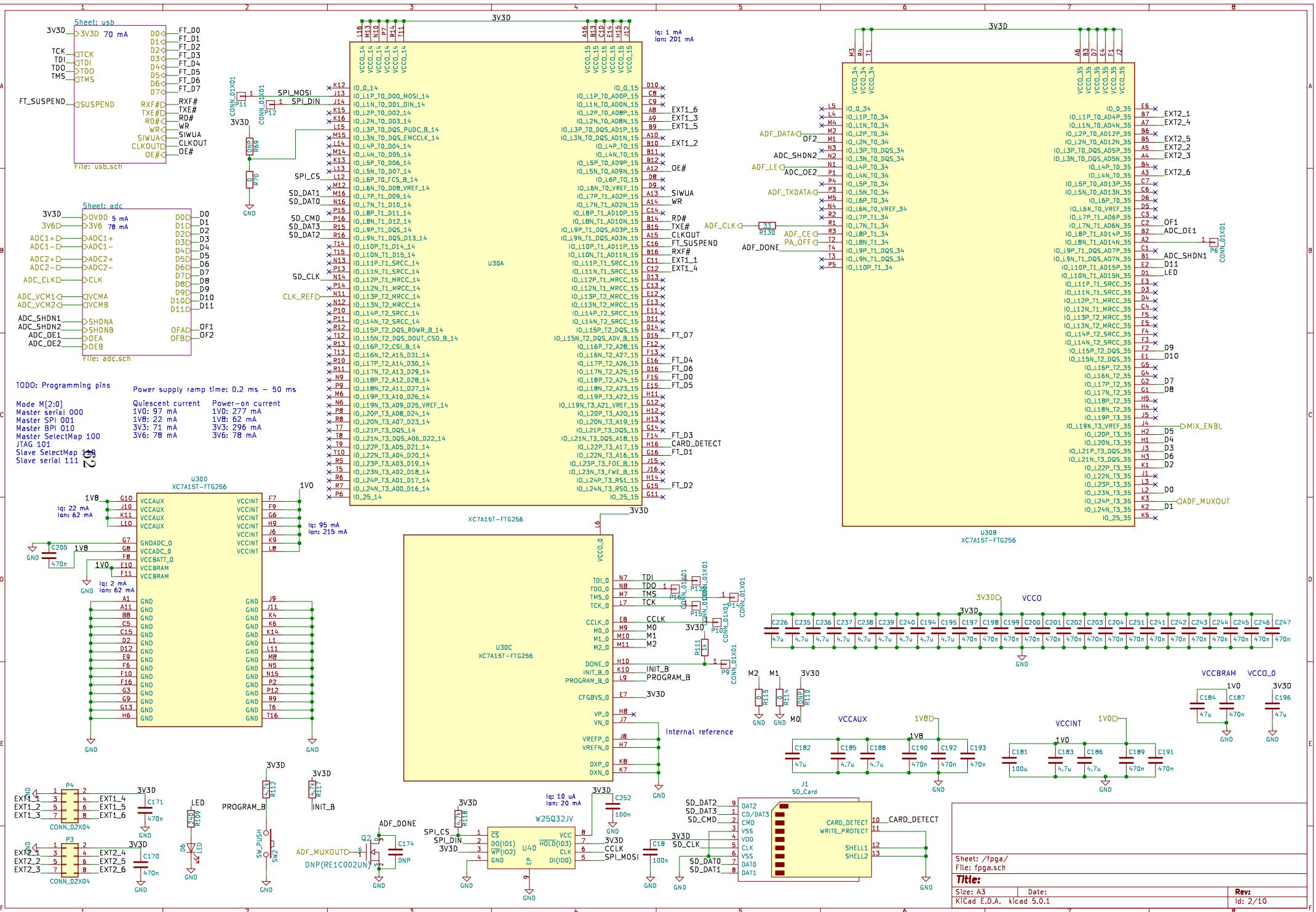
# **Chapter 6**

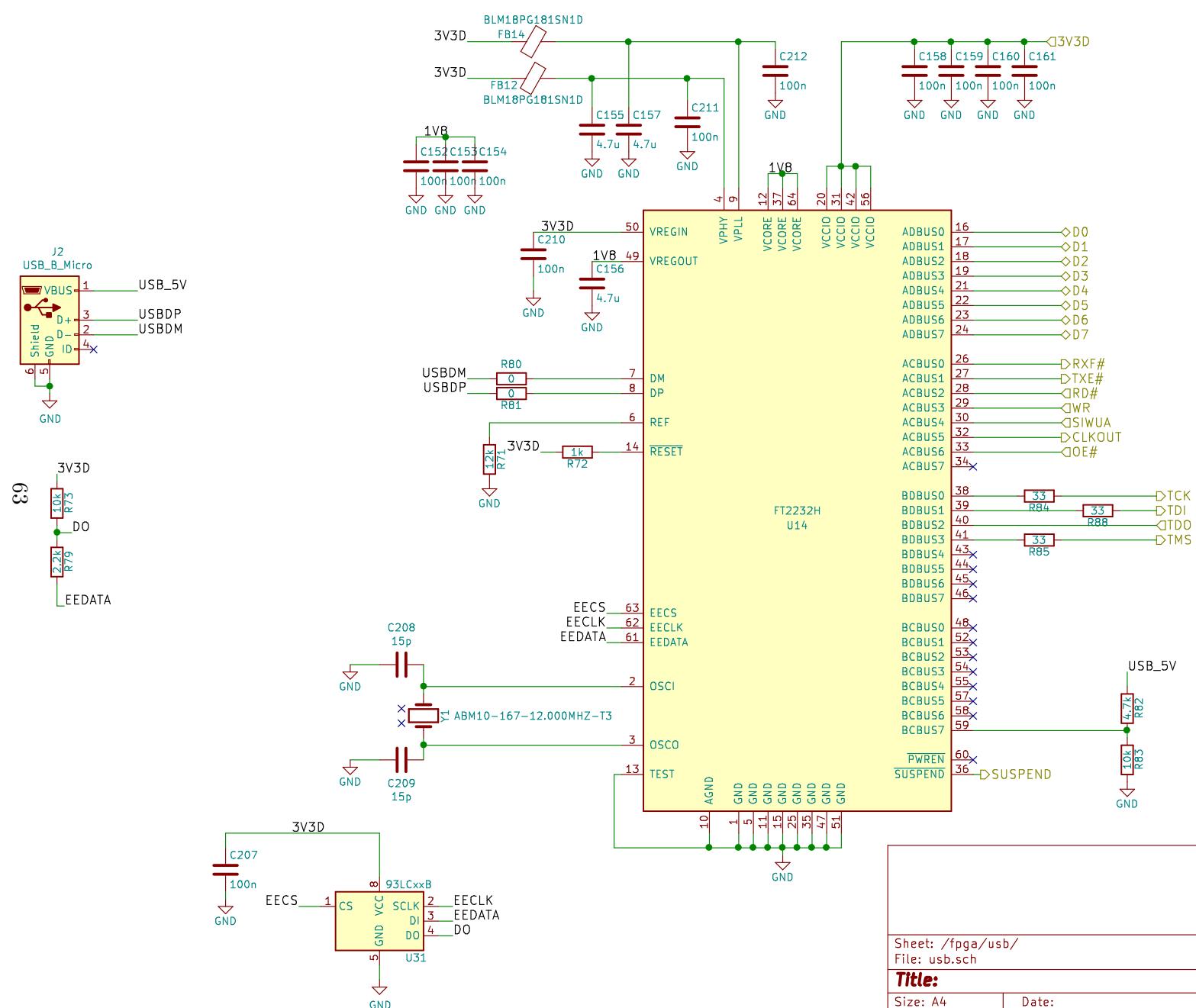
## **Links**

## Chapter 7

### Full Schematic





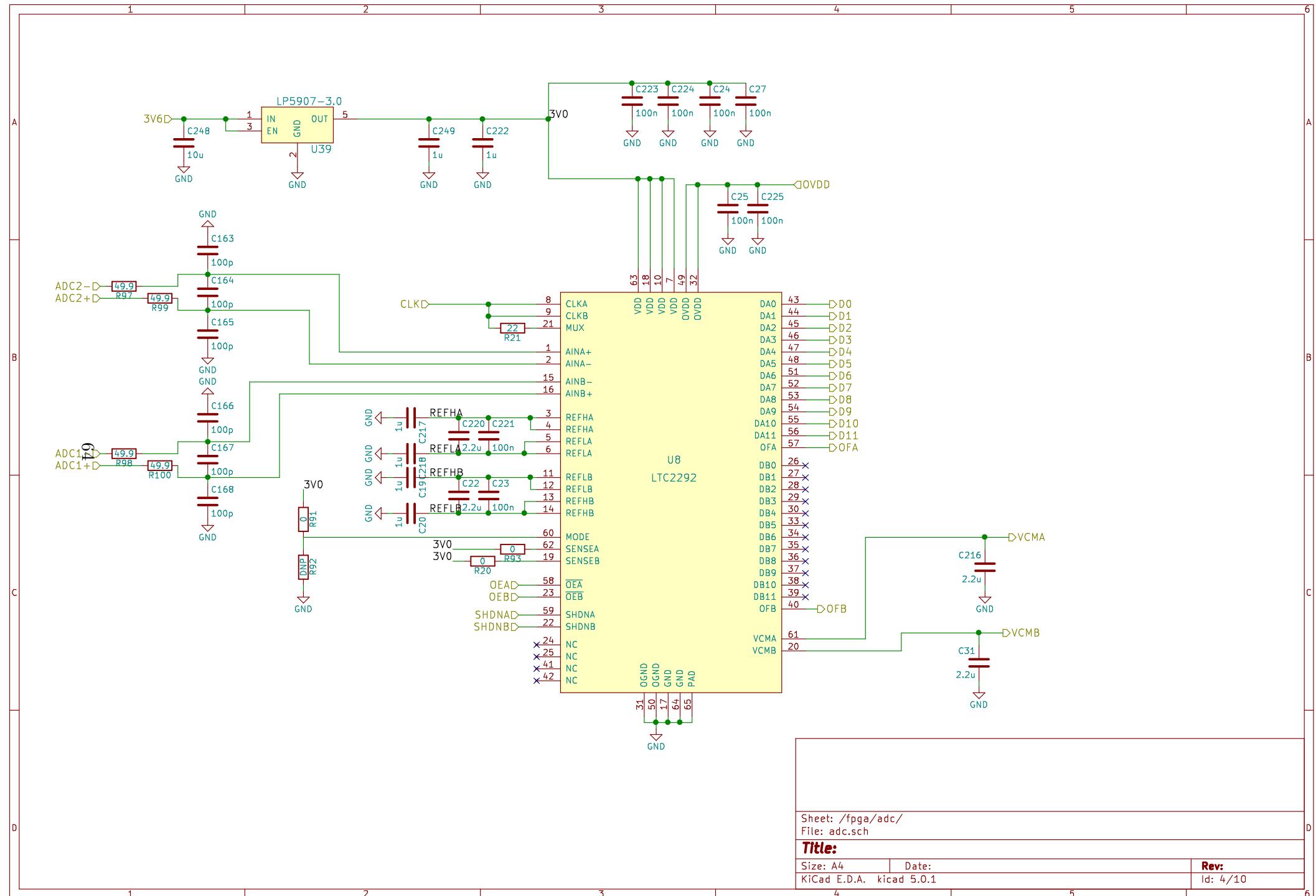


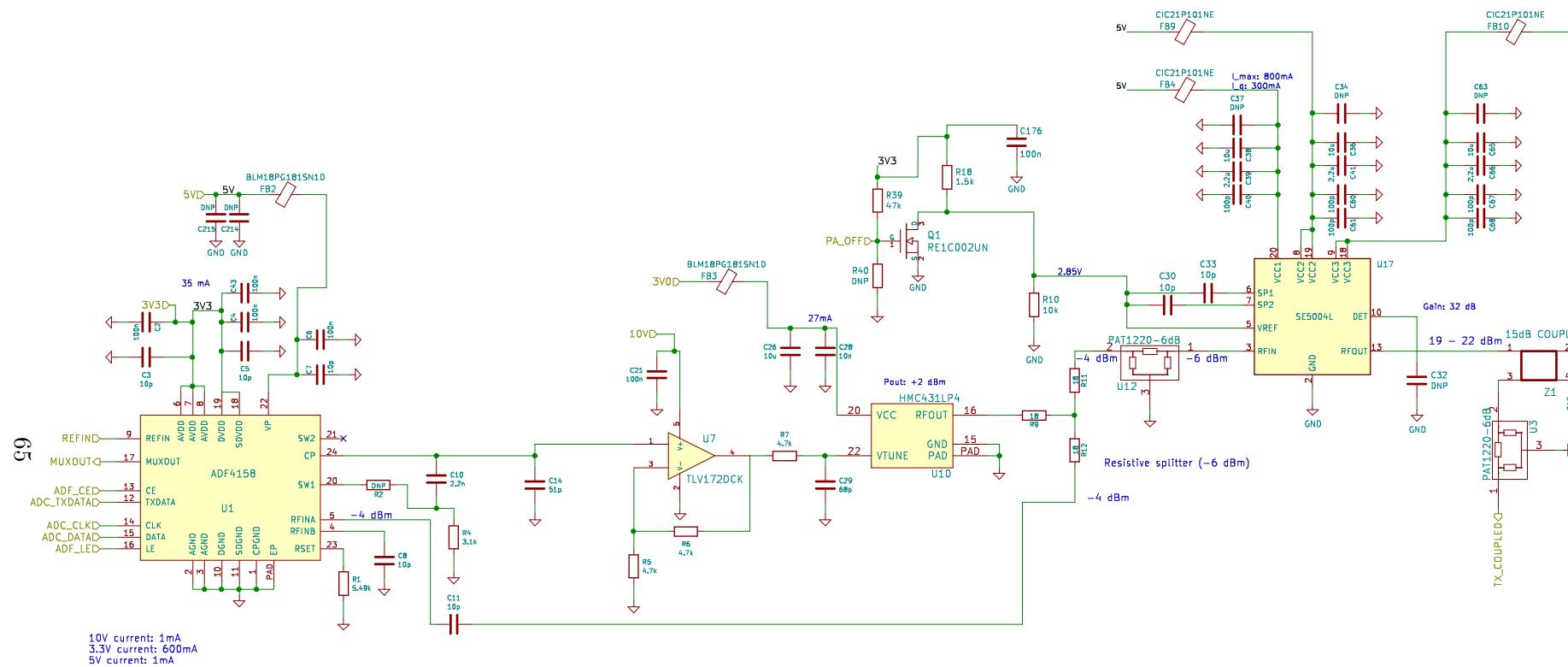
Sheet: /fpga/usb/  
File: usb.sch

Title:

Size: A4 Date:  
KiCad E.D.A. kicad 5.0.1

Rev:  
Id: 3/10



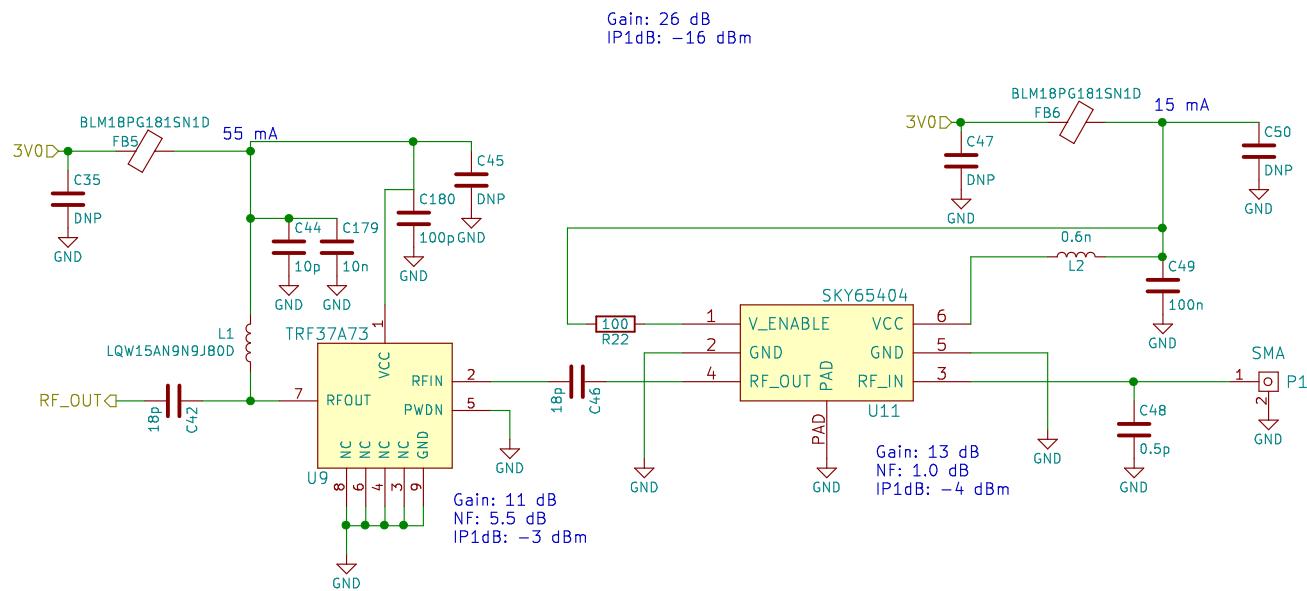


Sheet: ./tx/  
File: tx.sch

Title:

Size: A3	Date:
KiCad E.D.A.	kicad 5.0.1

Rev:
id: 5/10



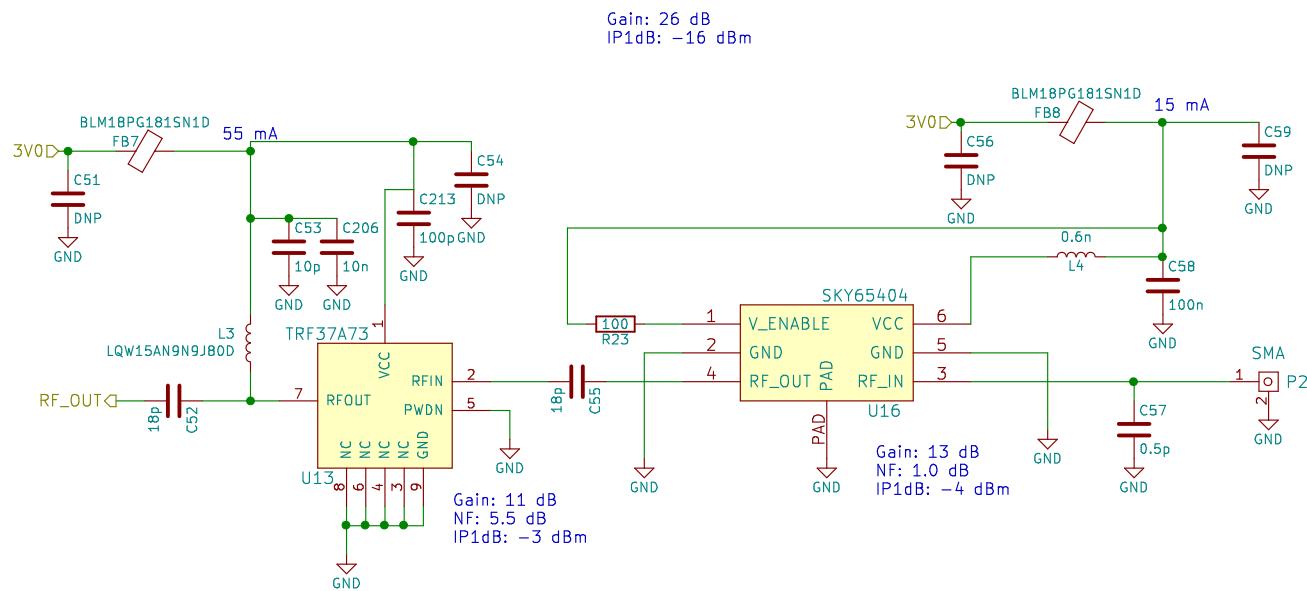
Sheet: /rx1/  
File: rx.sch

**Title:**

Size: A4 Date:  
KiCad E.D.A. kicad 5.0.1

Rev:  
Id: 6/10

L9



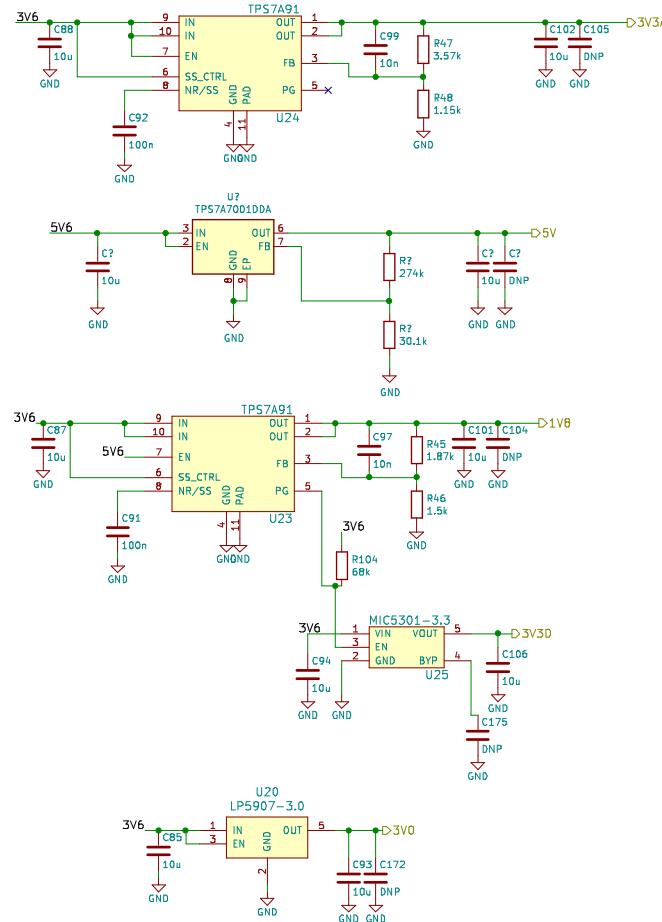
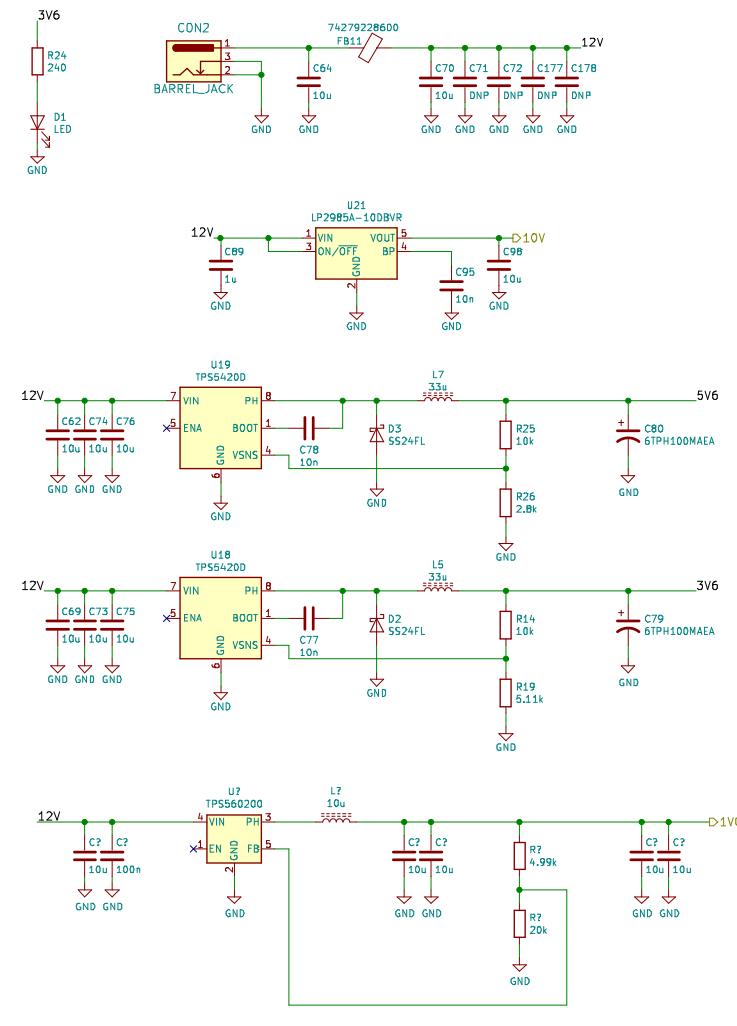
Sheet: /rx2/  
File: rx.sch

**Title:**

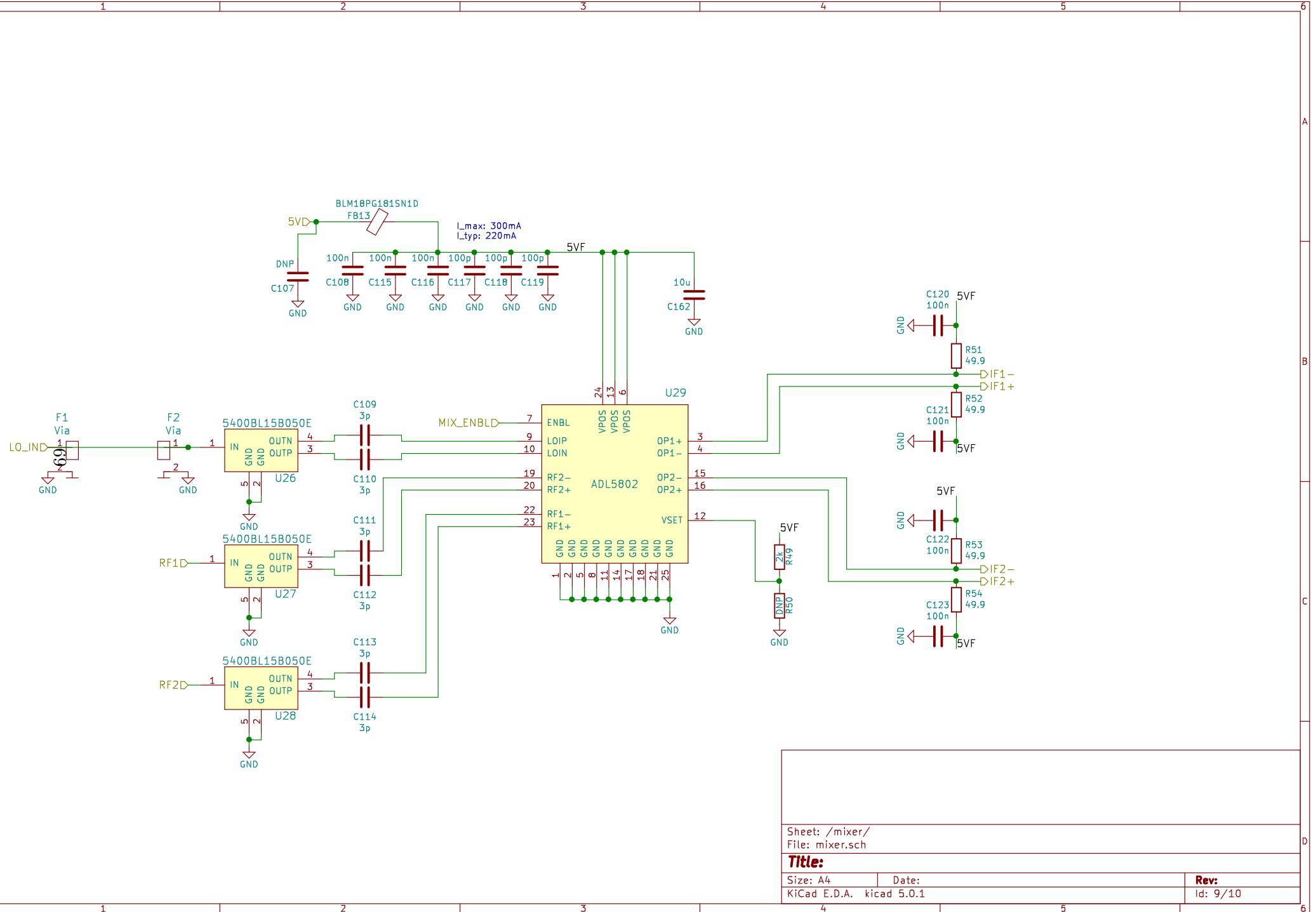
Size: A4 Date:  
KiCad E.D.A. kicad 5.0.1

Rev:  
Id: 7/10

68



Sheet: /power/	
File: power.sch	
<b>Title:</b>	
Size: A3	Date:
KiCad E.D.A. kicad 5.0.1	Rev: 8/10



Sheet: /mixer/  
File: mixer.sch

**Title:**

Size: A4 Date:  
KiCad E.D.A. kicad 5.0.1

Rev:  
Id: 9/10

