Project Yaigol

Part 1. Preface

This write-up was inspired by an earlier discussion about rigol DS1000Z series oscilloscopes, when a problem was discovered with the master clock oscillator in that line of scopes. The problem was caused by incorrectly selected PLL (phase locked loop) component values and strange programming of the PLL chip that could not be reasonably explained. Details can be found starting from this post by MarkL here:

http://www.eevblog.com/forum/blog/eevblog-683-rigol-ds1000z-ds2000-oscilloscope-jitter-problems/msg552777/#msg552777

Basically, the PLL was wobbly instead of outputting a stable clock signal. Because the clock drove the ADC chip, the jitter was directly affecting the ADC sampling. MarkL took a few screenshots of the PLL spectrum using his spectrum analyzer and the clock looked plain awful. There was no explanation or justification to such poor PLL clock design other than incompetency of the scope manufacturer and absence of quality control on their production line.

The manufacturer then issued a firmware update that bandaged the problem but did not eliminate it entirely, partially because a proper update required changing the hardware – the PLL loop components on the PCB. The second reason was of the same nature as the one I discovered during this project, and also required replacing the hardware components – more information on this will be provided later in this document.

After that and being still puzzled by the foolishness of this problem in DS1000Z scopes I decided to check my DS2072A scope. I am sure you've already guessed what I found. The answer is yes, same problem with the PLL clock existed in DS2072A. The 1GHz PLL clock was modulated like hell and sometime failed completely, generating plain narrow band noise. I decided to check it a little bit more and see if I can fix the PLL. At the time I did not realize this will become such a sizable investigation which I eventually named the "Project Yaigol".

During my investigation I observed (and you will see it as you read) that the manufacturer of these scopes could not make oscillate the circuits that should oscillate, but was very successful in making oscillate circuits that should not oscillate. For that reason I called that company "the masters of reverse oscillation". To integrate this honorable title into the scope manufacturer's name, the R letter was flipped. The backwards R is written as "Ya" and pronounced "ja" as in yacht or yack. Hence the name of the project: "Project Yaigol".

In this article I will provide information to DS2000 scope owners how to fix the PLL and how to fix other not less stupid problems I found during the investigation. Same as I, you may not at first believe such stupidity can happen. The bad news for you is the problems are caused by fundamental reasons, i.e. not by component tolerances but by bad design decisions as well as by programming that is incoherent with the dependent hardware. The good news is that not much effort is required to perform the fix. Still, it involves SMT work so you have to have appropriate skills and tools to do it. I will be giving instruction in the article as I write and I will summarize it all together in the end, so If you do not want to read the complete article you can jump straight to the last part for a fix guide.

I began looking into it somewhere in April 2015 and it lasted until late June, making notes and taking screenshots. This information has been sitting and waiting since then but I kept postponing writing a post because of other things. I will see if I can do it now and I am going to break it to parts organized by topics. Still may take a couple weeks to complete so please bear with me. Hope this will be to you both entertaining and useful.

Part 2. The PLL

In Yaigol 1000 and 2000 series scopes the 1GHz master clock for the ADC and downstream FPGA is provided by a phase locked loop (PLL) circuit. This part is not meant to be a comprehensive tutorial about PLLs, there is plenty of those on the Internet resources. I will only allude to details when it may help with the context.

In general, a PLL has two programmable divider counters, one is driven by a high frequency voltage controlled oscillator (VCO), the other by a low frequency reference oscillator which is usually an external crystal oscillator. The two counters are programmed in such a way so they produce close output frequencies from two different input frequencies. The counters' outputs are compared by a Phase Frequency Detector (PFD) which produces voltage proportional to phase difference between the two signals. This error correction signal is applied via a low pass filter (called loop filter) to the VCO and changes its frequency so the VCO signal phase aligns (to a certain tolerance) with the reference oscillator signal phase at the PFD input. Because the reference oscillator is typically a high stability source, the output of the PLL also stabilizes in frequency. The PLL achieves a "locked" state, providing the error control loop is properly designed. That includes the counters' coefficients, the PFD comparison frequency, the loop filter component values and some other parameters.

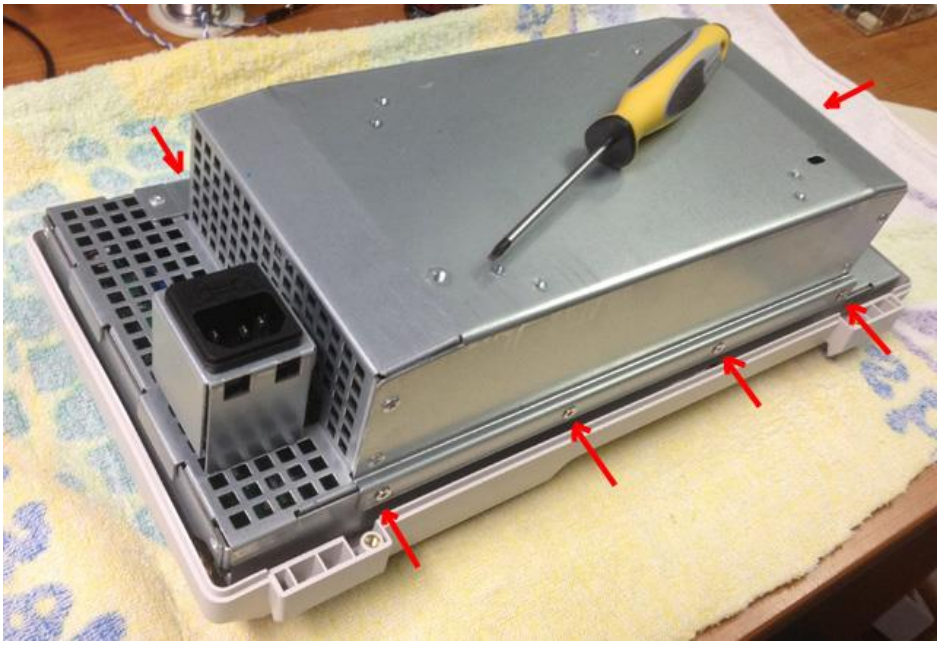
For proper operation a PLL has to lock each time it is powered on or after a change in PLL programming. If it fails to lock, it becomes unstable and the output frequency may either go into an endless loop chasing the reference signal and trying to catch up, which can be seen as wobble or frequency modulation on the screen of a spectrum analyzer, or it can totally wander away and change chaotically or burst into noise.

Yaigol 1000 and 2000 series scopes use ADF4360-7 PLL chip from Analog Devices. Link to the Datasheet:

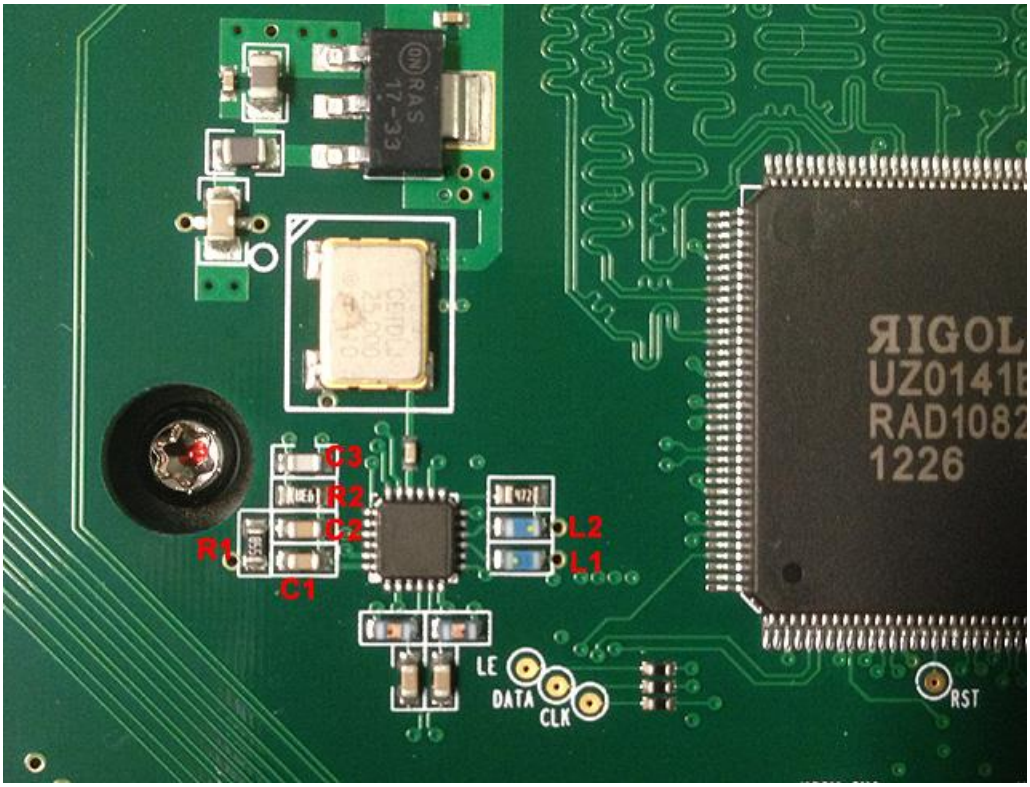
http://www.analog.com/media/en/technical-documentation/data-sheets/ADF4360-8.pdf

Getting access to the PLL required removing the scope back cover. Elsewhere on this forum there is information how to open Yaigol scopes without damaging the warranty sticker. I did not care about the warranty sticker so I just removed it. Location of the screws is shown in the below picture. You will need a Torx size T10 screwdriver with a relatively thin shaft. Not all screwdrivers with changeable blades may work because the two top screws are located under the handle and getting access to them requires holding the handle under a certain angle and the opening is fairly small.

Also the metal screen with the power supply has to be removed. It is held by (if I remember correctly) 8 Torx screws at the top and bottom. Also requires removing the nut holding the BNC Trigger Out connector. Once the back cover is separated a bit, the power supply connectors can be unplugged from the motherboard and the power supply can be put aside. I placed a towel on the desk to protect the scope front panel and controls from been scratched accidentally.



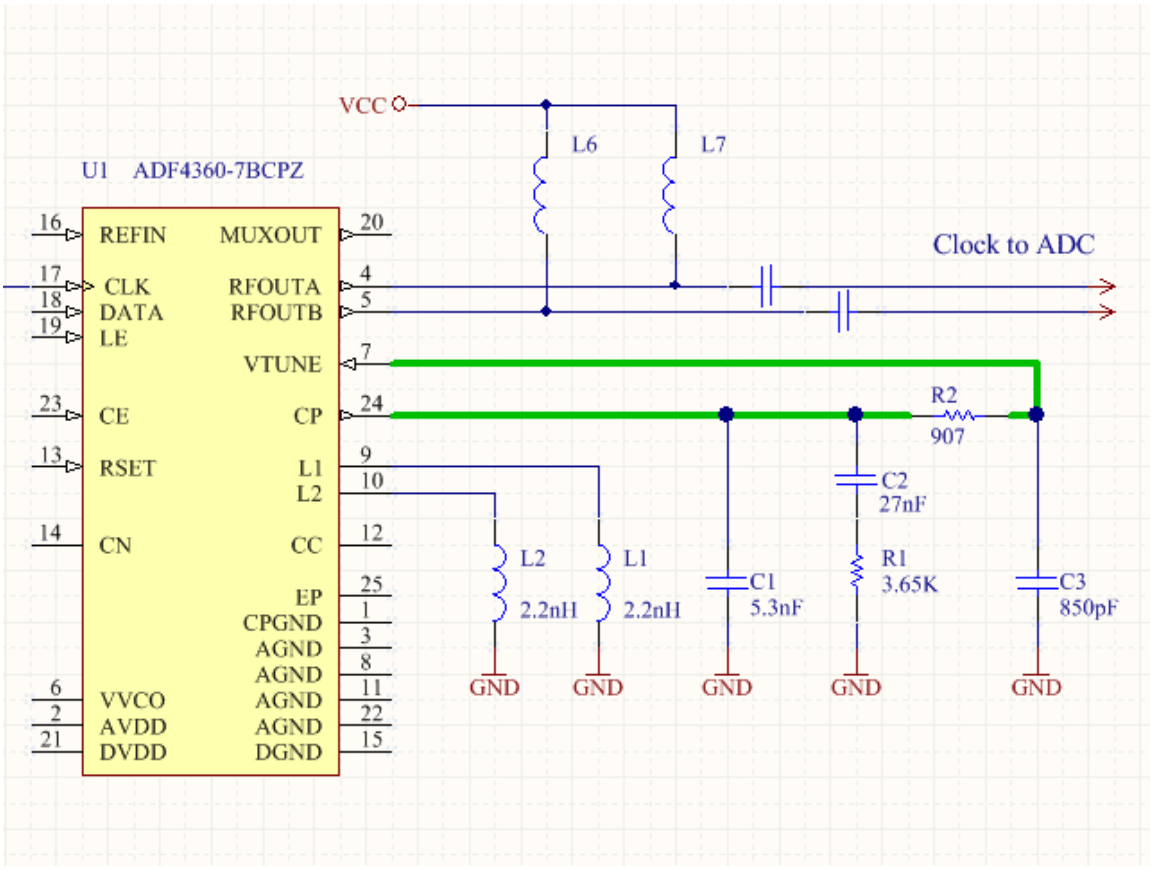
The PLL is located at the mid-left of the board to the left from the ADC chip and above the input stage shielding can. For some reason in DS2000 scopes Yaigol lasered off the PLL chip marking, perhaps in a foolish attempt to protect their ingenious design from been copied. Someone else suggested that they may have done it to hide they used clones of ADI chips. I do not know if this is true. In any case, it was childish and unnecessary because anyone who has ever used ADI 4360 PLLs would instantly recognize it.



Let's see what we got here. The PLL is the smaller chip. To the left of it is the PLL loop filter made of 2 resistors and 3 capacitors. To the right is the two blue-ish color VCO indictors. Above of the PLL is the reference clock oscillator (25MHz) and further above it is the low dropout voltage (LDO) 3.3V regulator NCP1117. Below the PLL is the two VCO loading inductors and dc decoupling capacitors and their bottom pads tracks then disappears into the PCB to drive the ADC which is the bigger chip on the right.

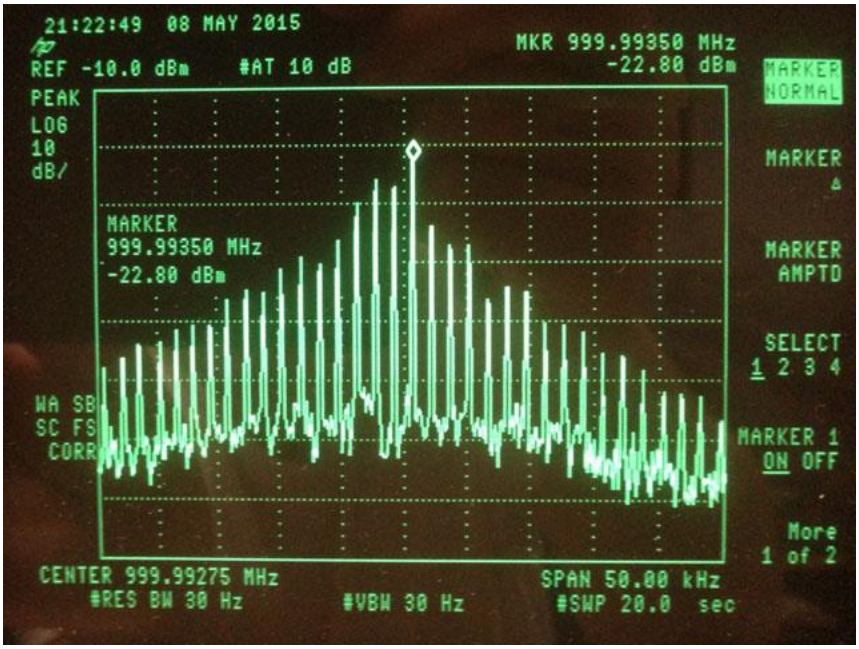
I was relieved to see test pads available for the serial bus, the LE/DATA/CLK right beside the PLL. That was going to probe the PLL much easier. I was also very curious to see what appeared to be three 0401 resistors to the right of the test pads. That made me think that perhaps they connect the PLL bus to the rest of the circuit and if remove them I may fully hijack the PLL bus. As it turned out, that was true and they were not resistors but zero Ohm jumpers. With the jumpers in place I could see on the test pads what I guess was the overall bus activity for whatever ICs connected to it. Removing the jumpers only affected the PLL and not the rest of the bus. That was interesting to see Yaigol added those jumpers, I do not see why that would be needed unless they were not sure about PLL programming.

Here is a related fragment of the schematic with the component values measured. I arbitrarily annotated the components so I can refer to them as I write. The loop filter circuit is highlighted in Green.

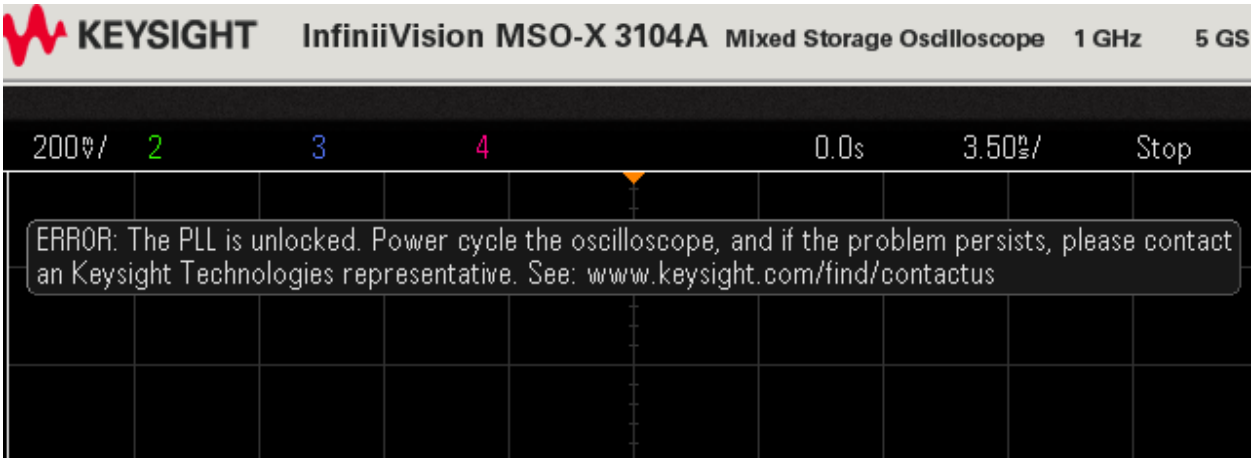


The first thing I did I checked the reference oscillator. It is not shown in the schematic but it drives the CLK pin 17. This is a 7x5 SMT part marked CETDLJ-25.000. I could not find a datasheet for the exact model but there are CEDT brand of oscillators made by Taitien (www.taitien.com). It is a cheap ass oscillator but I found not much wrong with it. Using a hot air tool and a temperature meter I determined the temperature drift was 10 Hz/C, which fit in specification of +/-25ppm -40…+86C given for CEDT model of oscillators. At this time, the oscillator itself was ruled out of the equation.

Next I unsoldered the ADC clock decoupling capacitors and hooked up a spectrum analyzer to L6 and L7 via an RF balun that converted the differential output of the PLL to single ended input of the SA. I have two spectrum analyzers: a HP8594E and HP8565A. I may use in this article screenshots made on either one. Tuning HP8594E SA to 1GHz center frequency revealed a gruesome picture. Boys and girls, let me introduce you DS2072A ADC clock:



So basically DS2072A ADC clock is garbage, same as was in DS1000 that I mentioned in Preface section earlier. Instead of generating a single frequency the DS2072A ADC clock is modulated wobbly junk. Typically that would indicate the PLL is unlocked, i.e. the PLL VCO is not tracking the reference oscillator. Any sane equipment manufacturer would have monitored the PLL and display a warning when PLL is unlocked. Also many if not all would not allow the equipment to operate under such condition. As an example, here is a screenshot done by MarkL on his Agiilent oscilloscope in the DS1000 thread:



Yaigol has nothing in place in either DS1000 or DS2000 scopes to monitor PLL unlock event and the PLL chip lock indicator pin is not even wired on the board, it is left floating. Therefore, there is an army of defective Yaigol oscilloscopes out there that unsuspecting users think they may relay upon. Perhaps it is time to start talking about Yaigol competence?

Behavior of a PLL would depend on selection of the loop filter components that in turn would be defined by the PLL programming. I measured the loop filter components (see the schematic above) but in order

to run them through the simulator with less guessing I also needed to know how Yaigol programmed the PLL. ADI has a simulator tool which is a free software and can be downloaded from here:

http://www.analog.com/en/products/rf-microwave/pll-synth.html

The simulator makes it very easy for even unexperienced person to design a PLL using ADI chips. You enter the requirements, select a chip and it calculates loop filter component values for you. It is easy, all you need is just use it. No one knows ADI PLLs better than ADI themselves. So why you would not use their simulator? Good question for Yaigol, next time you see your Yaigol Yingineer you could ask him.

I hooked up my little cute Sony/Tektronix data analyzer to the PLL bus and captured the programming data loaded to the PLL at power up. I also monitored the chip select LE line for a while and in different scope modes to make sure the PLL is not re-programmed during scope operation. It was not. The PLL was only programmed once at power up. That was good to know as I then could try an external clock source if I wanted.

ADF4360-7 PLL has 3 registers that control the PLL. I captured the programming data and decoded using the PLL Datasheet. Information relevant for our purpose is in the following table:

PLL Register	Programming value at startup	Meaning
R-Counter Latch:	00 03 E9	R counter: 250 (PFD 100kHz)
Control Latch:	49 11 24	Prescaler: 16/17 Charge pump current: 1.56mA Core power level: 10mA MUX Out control: Digital Lock Detect, active High
N-Counter Latch	02 70 42	B Counter: 624 A Counter: 16

First thing that looks suboptimal is a low PFD frequency of 100kHz. The R-Counter divides the reference frequency of 25MHz by 250, which gives 100kHz. ADF4360-7 is specified to max 8MHz PFD frequency. The following basics should be considered when choosing PFD frequency for a design:

If the PLL has to generate changeable frequencies, e.g. as in a multichannel radio, selection of PFD frequency will be determined by the required frequency step separation. Say we want to design a FM receiver with 100kHz channel spacing so each time we change channels our PLL changes its output frequency by 100kHz. To meet this requirement we choose PFD frequency of 100kHz.

PFD compares reference frequency with the VCO frequency divided by N=B*P+A, where P is the Prescaler value, B and A the N-Counter Latch values. The phase noise of the reference oscillator will be multiplied and within the loop bandwidth degrade the synthesizer phase noise by 20logN where N is the VCO divider coefficient. That means to reduce the PLL phase noise within the loop we need to choose smaller division ratio N (choose smaller B counter and A counter). We can do this if we design a fixed frequency PLL. i.e. no requirement to vary PLL frequency. That is the use case we are considering here which is the oscilloscope ADC clock source. How much we could improve the phase noise?

It can be shown that by going with 8MHz vs 100kHz PFD we could improve the synthesizer component of the noise by about 18dB. The other advantage of choosing a higher PFD is that the spurs caused by the PFD will move farther away from the carrier on the frequency scale and become easier to suppress by the loop filter.

The second is the PLL Charge Pump Current of 1.56mA. I do not know why Yaigol would choose that value. ADI characterized the PLL chip in the Datasheet at 2.5mA. When you run a new design in the ADI Tool, the default value is 2.5mA. I always set my designs to use 2.5mA Icp.

The third goes the Core Power Level of 10mA. ADI recommended value is 5mA, and this is what ADI has to say on the Analog Engineering Zone forum:

'====

...In particular check the core power, it is critical that this be set to 5 mA

... changing the core power would have changed the VCO sensitivity, leading to a (possibly) more stable loop filter. But in many cases it may lead to more instability (and now the simulator data is invalid also).

...Using a non-specified VCO core power current changes the VCO vs. frequency characteristic completely

...Only the specified VCO core power current should be used for the ADF4360-x family

...all measurements in characterization, yield analysis and production test are based on 5 mA. Using a different VCO setting will indeed change the VCO characteristics, to characteristics that are largely unknown, except in a few limited cases. If changing the core power 'solved' the problem, my guess is that it simply masked it.

'====

What that means to me is if you program the Core Power Level to anything but 5mA, you are on your own. The formulas in the Datasheet do not apply anymore. The ADI simulator does not have a setting for it, which perhaps means the Simulation results are only valid for 5mA Core Power Level. The Datasheet says in plain English about the Core Power Level: “The recommended setting is 5 mA”. Why would one want to set it to anything different? Next time you speak with your Yaigol Yingineer, you could ask him.

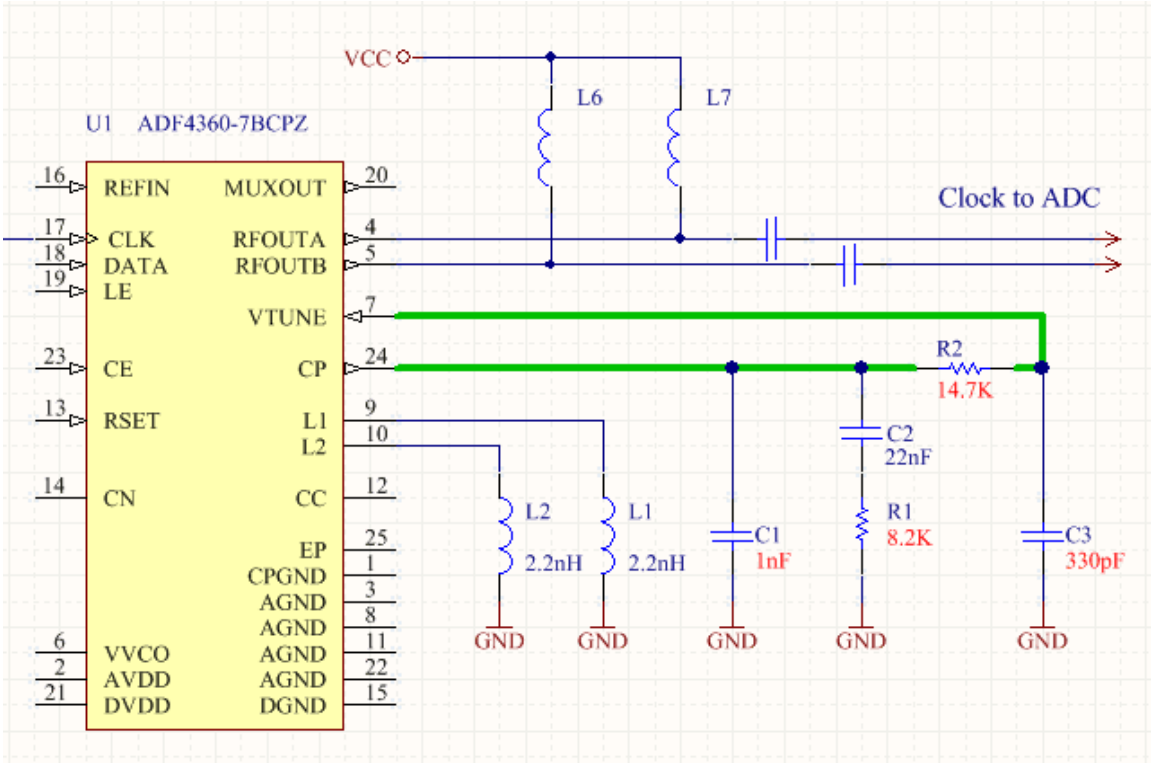
Other parameters that we need to look at are the PLL phase margin and loop filter bandwidth. Punching the measured loop filter component values, PFD frequency and reference clock into the ADI simulator produced the following result:

Loop Filter	CPP_3C
Specify:	Components
Loop Bandwidth	2.04kHz
Phase Margin	37.8 deg
Zero Loc.	1.62kHz
Pole Loc.	8.67kHz
Last Pole	225kHz
C1	5.30nF
R1	3.65k
C2	27.0nF
R2	970
C3	850pF

What is worrying here is a low phase margin value. Low phase margin is an indication of a potentially unstable PLL loop. Different sources recommend phase margin between 40 and 70 degrees. Yaigol value is below the minimum recommended one. The ADI tool recommends 45 degrees as a start value when you do a new design. if I create a new design from scratch (given the required PLL output frequency of 1GHz, reference clock of 25MHz and PFD of 100kHz), the ADI tool produces a different loop filter:

Loop Filter	CPP_3C
Specify:	Components
Loop Bandwidth	4.15kHz
Phase Margin	56.1 deg
Zero Loc.	881 Hz
Pole Loc.	13.4kHz
Last Pole	50.6kHz
C1	1.00nF
R1	8.20k
C2	22.0nF
R2	14.7k
C3	330pF

Note the phase margin has improved and is within the recommended range. I tweaked the component values just a bit to make them standard values. Here is the updated schematic with the new loop filter components marked in Red:


VCC
L6
L7
U1 ADF4360-7BCPZ
16 REFIN
17 CLK
18 DATA
19 LE
23 CE
13 RSET
14 CN
6 VVCO
2 AVDD
21 DVDD
MUXOUT 20
RFOUTA 4
RFOUTB 5
VTUNE 7
CP 24
L1 9
L2 10
CC 12
EP 25
CPGND 1
AGND 3
AGND 8
AGND 11
AGND 22
DGND 15
Clock to ADC
R2
14.7K
C2
22nF
R1
8.2K
C3
330pF
C1
1nF
L2
2.2nH
L1
2.2nH
GND
GND
GND
GND
GND