Ultra low power integer-N ADPLL

Master's thesis project - meeting 7

Cole Nielsen
Department of Electronic Systems, NTNU

28 February 2020 (calendar week 9)

Overview

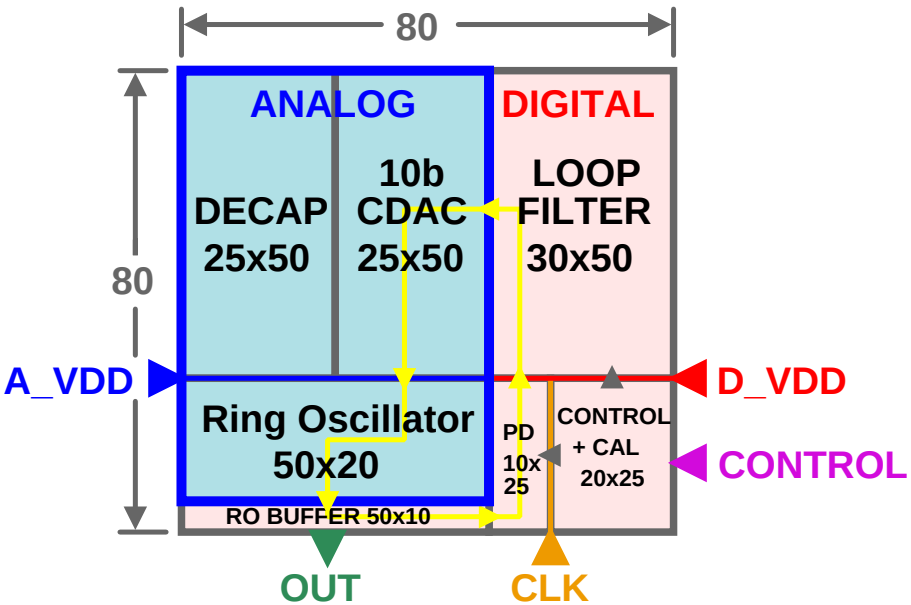
For this week...

1. PLL floorplan
2. Loop filter V2
   - Area/pin location aligned with floor plan.
   - Simplified DSP operation.
   - Changeable filter coefficients.
3. Spur estimate.

PLL Floorplan

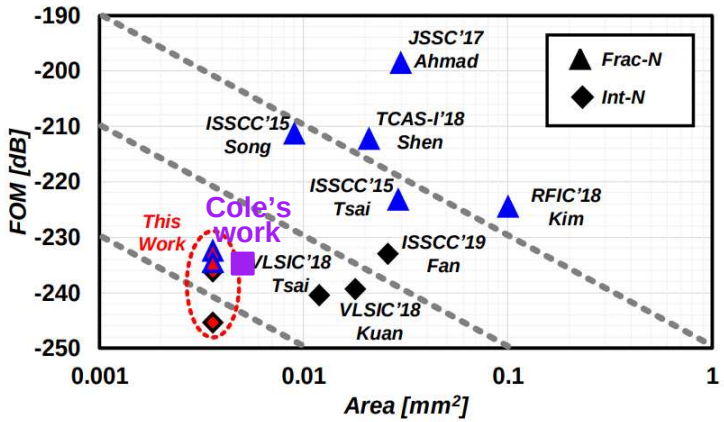
Loop filter

- Full PLL floorplan necessary to make reasonable placement of pins and to constrain size of loop filter.
- **Total area = 0.0064 mm^2** (80 μm x 80 μm)
  - **Active area = 0.00515 mm^2** (minus DECAP)
- Loop filter architecture simplified to reduce area into 30 μm x 50 μm.
- 10b CDAC utilizing minimum 1μm x 1μm APMOM caps.
  - Between 1.96-4.72 fF per unit cap (2-4.8 pF total), depending how many metal layers used.
  - Depends on arangement, however should fit in either 25 μm x 50 μm or perhaps 30 μm x 50 μm.
- Include dedicated decap for analog to help with noise?? 25 μm x 50 μm will yield ca. 30 pF.
- Seperate power for analog/digital
  - Possibly 0.5 for digital and 0.8 for analog? Higher supply is better for supply rejection with VCO...



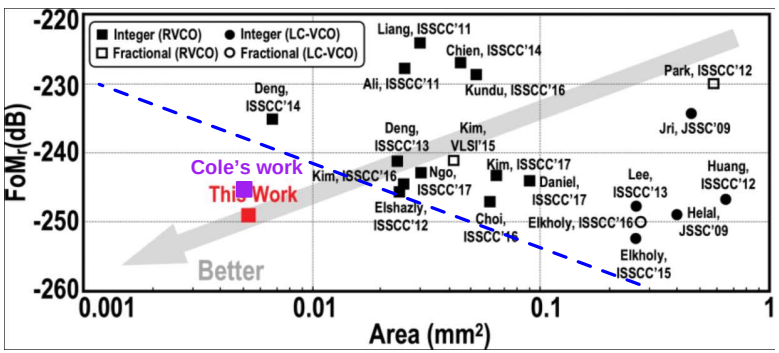
Renewed area comparison.

5nm FinFET state of art



— Sythesized PLL in TSMC 5nm FinFET [3] that *has* been fabricated, published in SSCL (2020). Area is 0.0036 mm^2, or 50 μm x 72 μm.

MDLL State of art



— Figure from 2019 JSSC paper [2] on a MDLL. Area is 0.0056 mm^2 in 28nm technology.

— FOM_r here is with normalization to a 200 MHz reference oscillator.

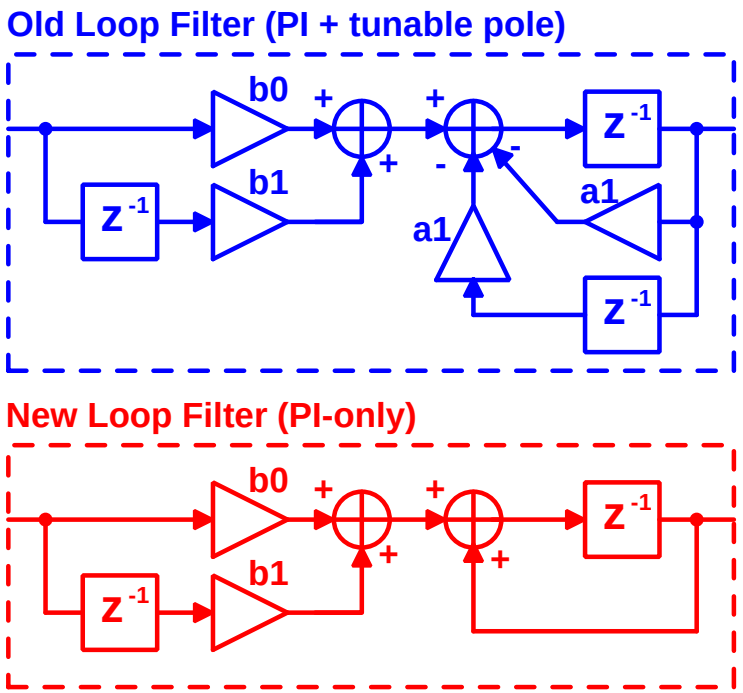
— This work uses higher power (1.45 mW), and higher reference (200 MHz), which perhaps explains better FOM.

Verdict: Competitive in area with state of art (not revolutionary though).

New loop filter.

PI-only with changeable filter coefficients.

- Original architecture with 2 feedforward/2 feedback filter coefficients becomes large (ca. 60 μm x 60 μm).
  - Implemented fixed pole at zero, and one each of tunable pole/zeros.
  - With N bit word size, requires 4x NxN array multiplier.
- **Solution:** reduce to only two feedforward programmable filter coefficients. **This is a classical PI-controller**.
  - Stability is straightforward, PLL dynamics simpler to model.
  - Uncompensated zero results in inherent peaking of phase noise spectrum.
- **Also implemented:** On-the-fly changeable filter coefficients to allow for gear switching of the PLL.



New loop filter.

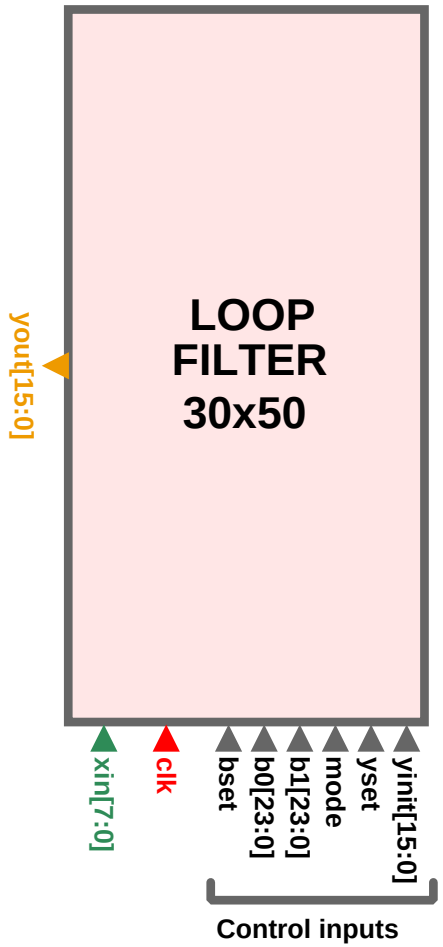
PI-only with changeable filter coefficients.

— Current filter optimization yields the following minimum resolution requirements:

Mode	Sign	Integer	Fractional	Sum
Counter	1	4	5	10
BBPD	1	4	11	16

— Selection of the input word size xin has been made to equal the counter resolution (8 bit). Accounting for a sign bit this equates to 7 integer bits.

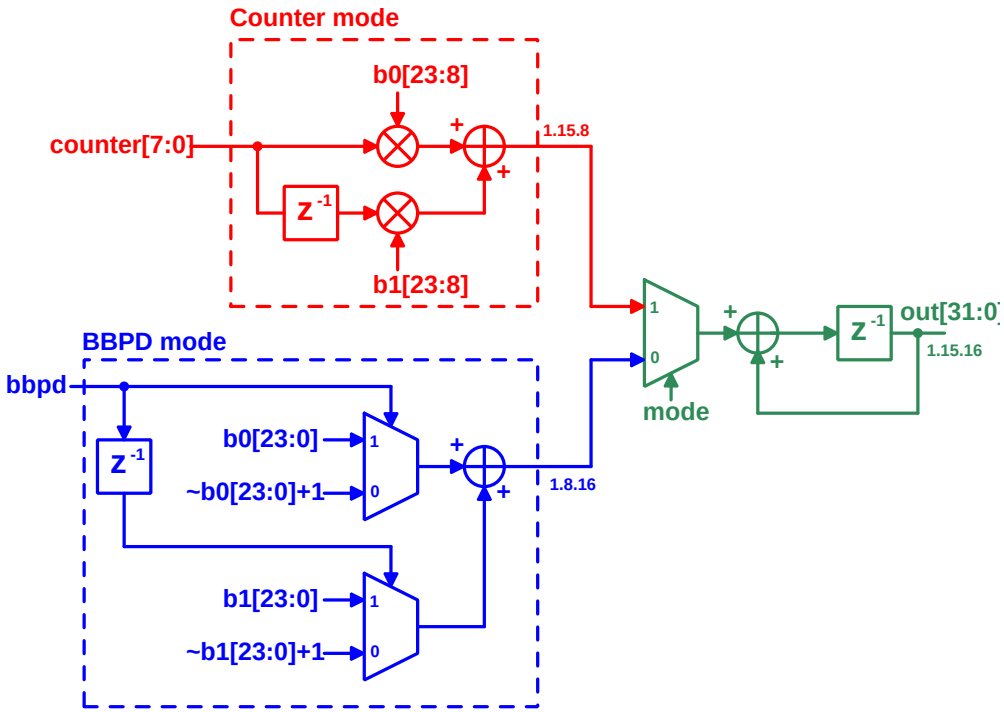
— **Also implemented:** On-the-fly changeable decimal location to allow for gear switching of the PLL.



New loop filter.

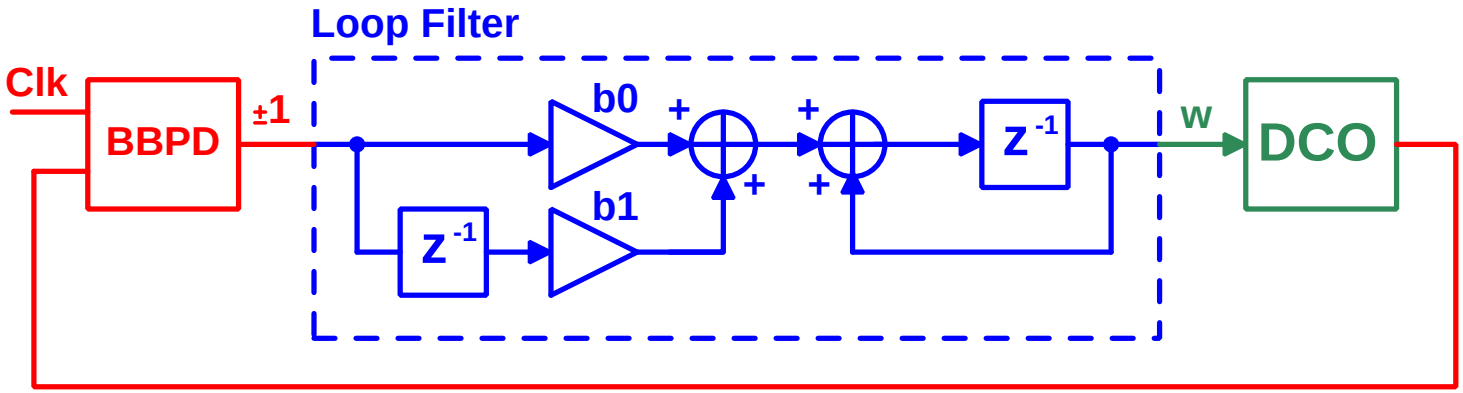
New split architecture.

- Now use split datapath for counter and BBPD modes.
- Reducing to only two feed forward coefficients allows for removal multipliers for BBPD use.
  - Now implemented with muxes.
  - Lower power/complexity, can use larger word for filter coefficients (24 bit) with little penalty, ie. adder complexity is $\mathcal{O}(n \log n)$, vs $\mathcal{O}(n^2)$ for multipliers.



BBPD-PLL resolution jitter

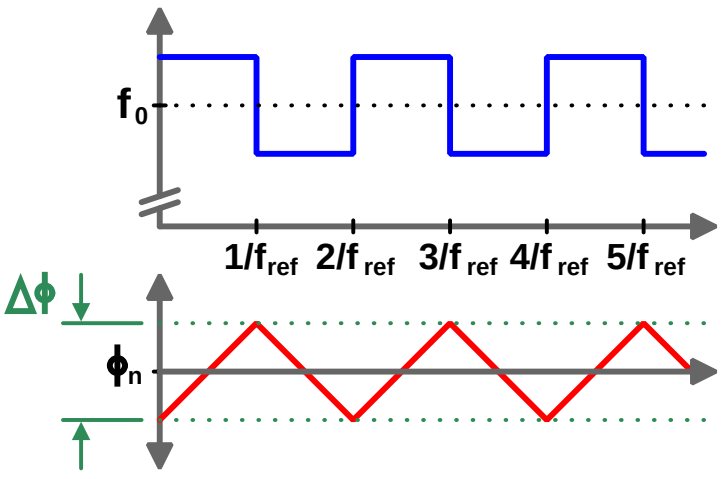
Resolution/phase quantization effects

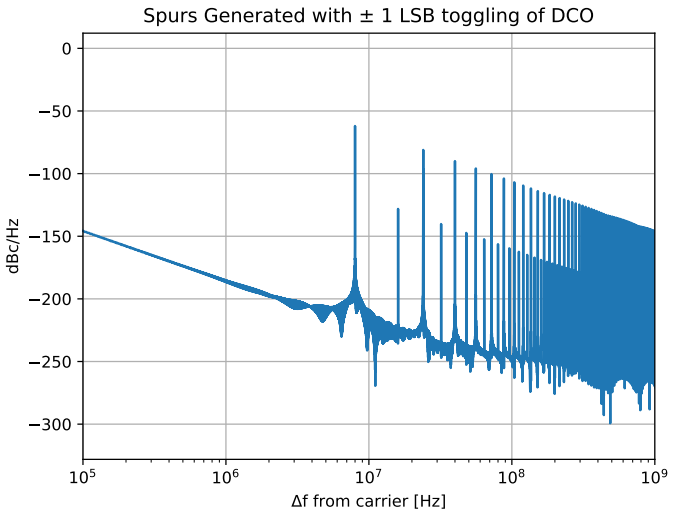


- Output of BBPD is quantized $\pm$ 1. With PI loop filter architecture, there are only 4 possible values that the output **w** can increment by: $\lfloor b_0 + b_1 \rfloor$, $\lfloor b_0 - b_1 \rfloor$, $\lfloor -b_0 + b_1 \rfloor$, $\lfloor -b_0 - b_1 \rfloor$.
- In steady state, with the BBPD outputting a worst case sequence of +1/-1/+1/-1... , the output will toggle betwwen $\lfloor b_0 - b_1 \rfloor$ and $\lfloor -b_0 + b_1 \rfloor$.
  - Current optimization yields b_0 = 8.899856, b_1 = -8.301153
  - Thus output **w** will increment by +17, -18, +17, -18 ...
  - Essentially output will make large-ish jumps in frequency every reference cycle

BBPD-PLL resolution jitter

Worst case estimate (for spurs)





- Worst case input sequence results in square wave frequency modulation, which in the phase domain creates a cyclo-stationary triangle wave with period $f_{ref}/2$. This creates **SPURS** at $f_{ref}/2$.
- In general, $f_{ref} >> K_{DCO}|b_0 - b_1|$ (frequency deviation), so spurs are not generated at the deviation frequencies.
- With 1 LSB deviation per ref. cycle, a -62 dBc spur (SSB) is expected $f_{ref}/2$.
  - Under my current b_0, b_1, this spur will be -37 dBc.

BBPD-PLL resolution jitter

Total phase noise estimate from resolution jitter.

— **Cyclostationary (+1, -1, +1, -1, ...):**

- Under my system parameters, the total phase noise power from resolution effects is -34 dBc, essentially all of which is in the first spur at $f_{ref}/2$.

$$\Delta\Phi = \frac{2\pi|b_0 - b_1|K_{DCO}}{f_{ref}} \tag{1}$$

$$\sigma_{\Phi rj} = \frac{\pi|b_0 - b_1|K_{DCO}}{\sqrt{3}f_{ref}} \tag{2}$$

— **General note:** the phase noise from this source must $<<$ than target SNR for PLL application. Can use these relations to find limits for maximum K_{DCO} before resolution jitter is dominant.

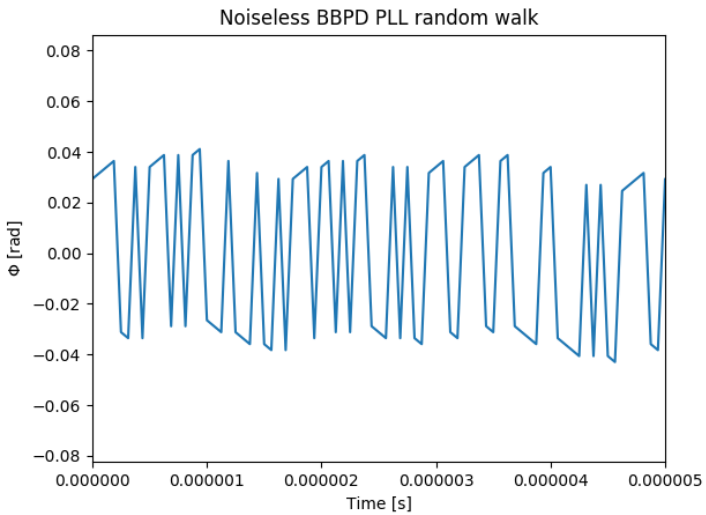
BBPD-PLL resolution jitter

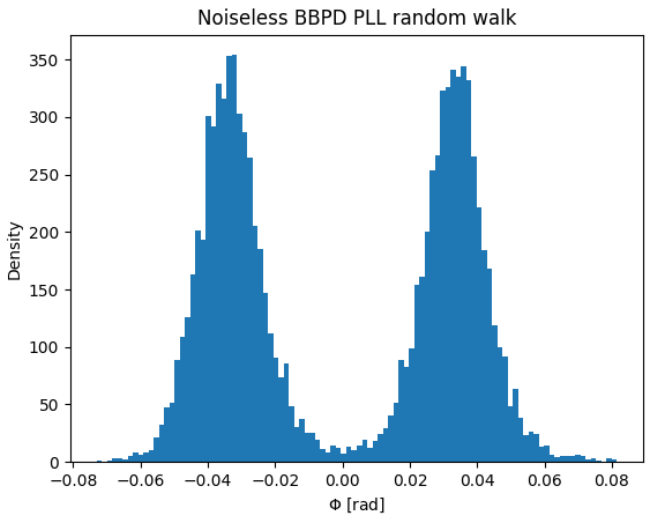
Total phase noise estimate from resolution jitter.

— **Average case:**

- All transitions equally likely out of BBPD.
- Under my system parameters, the total phase noise power from resolution effects is -29.4 dBc (simulated).
- Phase noise is bimodal. The total RMS phase noise power is approximately equal to the absolute value of the distribution means.

$$\mu = \pm \frac{\pi |b_0 - b_1| K_{DCO}}{f_{ref}}, \qquad \sigma_{\Phi rj} \approx |\mu| \tag{3}$$





BBPD Gain.

Additional nonlinearity to an already nonlinear thing...

— Found an interesting dissertation on BBPD PLLs [3], which shows that the linearized BBPD gain model I have been using ($K_{BBPD} = 2/\sqrt{2\pi}\sigma_{\Phi_n}$) is not totally correct.
  - Due to BBPF-PLL resolution jitter, varies depending if resolution jitter or if random noise is the dominant component.

— Need to account for this in my filter optimization code...

— An approximation, with random/uncorrelated noise with $\sigma_{\Phi uc}$, and resolution jitter $\sigma_{\Phi rj}$ (based on variable substitution of [3]'s theory):

$$K_{BBPD} = \frac{1}{\sqrt{2\pi}\sigma_{\Phi uc}} \left[1 + e^{-\frac{1}{2}\left(\frac{\sigma_{\Phi rj}}{\sigma_{\Phi uc}}\right)^2} \right] \quad (4)$$

$$\sigma_{\Phi rj} \approx \frac{\pi |b_0 - b_1| K_{DCO}}{f_{ref}} \quad (5)$$

— Currently, I should be well into the random noise dominated regime.

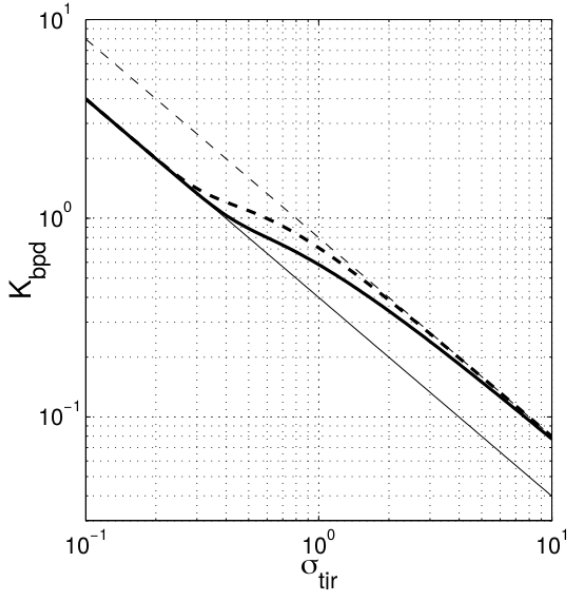


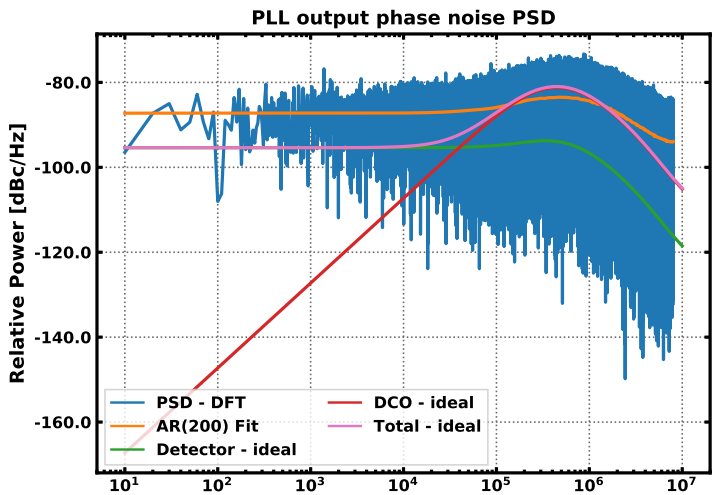
Figure 4.5: Plot of K_{bpd} versus σ_{tjr} (gaussian jitter, $N\beta K_T$ normalized to 1): K_{bpd} computed with a 101 states Markov Chain (thick solid), K_{bpd} approximated with a 3 states Markov Chain (thick dashed), asymptote $K_{bpd} = 1/(\sqrt{2\pi}\sigma_{t_{jr}})$ (thin solid), asymptote $K_{bpd} = 2/(\sqrt{2\pi}\sigma_{t_{jr}})$ (thin dashed).

Filter optimization

Need to add

1. Modeling of BBPD jitter (extracted from circuit simulation).
2. Account for noise from limit cycle/phase resolution effects.
   - Adjust K_{BBPD} accordingly
3. Aliasing/folding of high frequency phase noise into bandwidth of f_{ref} due to sampled nature
   - Impossible to avoid...

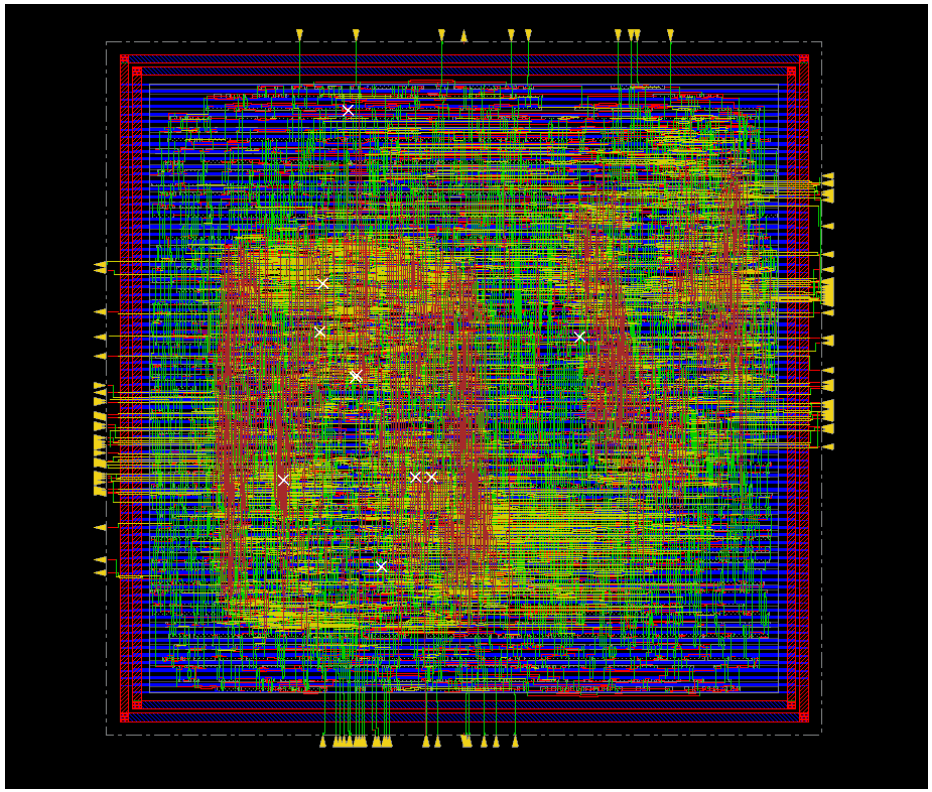
Can maybe get better correlation between filter optimization results and discrete time simulation.



Synthesis/place and route

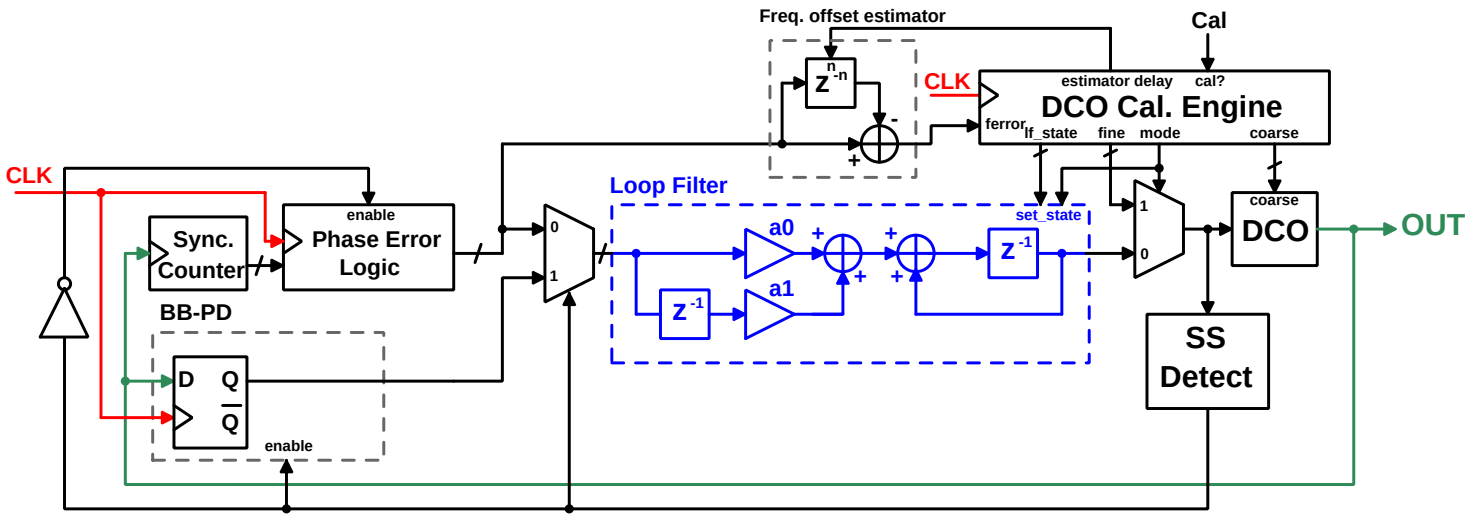
Loop filter layout.

— First iteration at loop filter layout... still need to extract the circuit/LVS etc...



Architecture

Block Diagram



Power Targets (revised)

(Divider not necessary)

DCO	Phase detector	Digital (LF)	Other	SUM
50 μW	10 μW	10 μW	**0** ~~≤ 5~~ μW	$\leq$ **70** ~~100~~ μW

Specification

System Performance Targets

Parameter	Value	Unit	Notes
Frequency	2.4-2.4835	GHz	2.4G ISM Band
Ref. frequency	16	MHz	Yields 6 channels
Power	$\leq$ **70** ~~75~~ μW	μW	Minimize!
FSK BER	$\leq$ 1e-2		GFSK* with f_{dev}=$\pm$250 KHz
CNR	$>$ 20	dBc	Yields **-235** dB FOM_{jitter} ideally
Initial Lock Time	$\leq$ 10	μs	Upon cold start
Re-lock Time	$\leq$ 5	μs	Coming out of standby, f_{error} $<$ **1 MHz**
Lock Δf tolerance	100	kHz	
FOM_{jitter}	$\leq$ -230	dB	**For state of art in size/power**
Area	$<$ 0.01	mm^2	

* Using BT=0.3, 1 MSymbols/s, 4 demodulated symbols averaged per bit to yield 250 kbps.

Specification

Component-level specs

Parameter	Value	Unit
Counter range	256 steps	coverage of 150-155
Divider ratio	150-155	(For non-counter based)
~~**TDC resolution**~~	~~$\geq$ 155~~	~~steps/reference cycle~~
DCO gain K_{DCO}	10^4	Hz/LSB
DCO tuning range	10	MHz
DCO DAC resolution	10	bit
DCO Phase noise	$<$ -80	dBc/Hz @ $\Delta f = 10^6$ Hz, f_c = 2.448 GHz
DCO Power	$\leq$ 50	μW
Digital filter word resolution	$\leq$ 16	bits (power grows as $\mathcal{O}(n^2)$)
BB-PD jitter	$\leq$ 12	ps_{rms}

Time plan (pt. 1)

Week #	Dates	Tasks	Outcomes
4	20.1 - 26.1	Finalize high level modeling	Component level specification
5	27.1 - 2.2	Establish test bench in Virtuoso	With ideal PLL implementation
6	3.2 - 9.2	Schem. design: phase detector	TDC - flash and counter based
7	10.2 - 16.2	Schem. design: phase detector	Bang-bang phase detector
8	17.2 - 23.2	RTL, synthesis, place&route	Digital loop filter
9	24.2 - 1.3	RTL, synthesis, place&route	Digital loop filter
10	2.3 - 8.3	Schem. design: oscillator	Ring DCO
11	9.3 - 15.3	Layout: oscillator	
12	16.3 - 22.3	CDAC	Schem+layout
13	23.3 - 29.3	Calibration	RTL/schem. for calibration
14	30.3 - 5.4	Flex week - schem. design	Finalize schematic level design
15	6.4 - 12.4	**Easter**	-
16	13.4 - 19.4	Layout	Phase detector
17	20.4 - 26.4	Layout	Oscillator

Legend: **Done** **Current** **Revised**

Time plan (pt. 2)

Week #	Dates	Tasks	Outcomes
18	27.4 - 3.5	Layout	Divider/calibration
19	4.5 - 10.5	Layout	Finalization/system integration
20	11.5 - 17.5	Flex week (layout) OR yield improvement	Depending on progress
21	18.5 - 24.5	**Report writing**	
22	25.5 - 31.5	**Report writing**	
23	1.6 - 7.6	**Report writing**	**Deadline 8.6**

Legend: **Done** **Current** **Revised**

References

[1] Liu, B., Li, Z., Fu, X., Shirane, A., Kurosu, H., Nakane, Y., Masaki, S., Okada, K., Zhang, Y., Qiu, J., Huang, H., Sun, Z., Xu, D., Zhang, H., Wang, Y. and Pang, J. (2020). A Fully-Synthesizable Fractional-N Injection-Locked PLL for Digital Clocking with Triangle/Sawtooth Spread-Spectrum Modulation Capability in 5 nm CMOS. IEEE Solid-State Circuits Letters, pp.1-1.

[2] Yang, S., Yin, J., Mak, P. and Martins, R. (2019). A 0.0056-mm2 -249-dB-FoM All-Digital MDLL Using a Block-Sharing Offset-Free Frequency-Tracking Loop and Dual Multiplexed-Ring VCOs. IEEE Journal of Solid-State Circuits, 54(1), pp.88-98.

[3] N. Da Dalt, "Linearized analysis of a digital bang-bang PLL and its validity limits applied to jitter transfer and jitter generation", IEEE Trans. Circuits Syst. I, vol. 55, pp. 3663-3675, Dec. 2008.