Adaptive Clocking Techniques for SoC Supply Droop Response in Predictive 7nm CMOS

Dan Fritchman, Wahid Rahman

EE241B Final Project

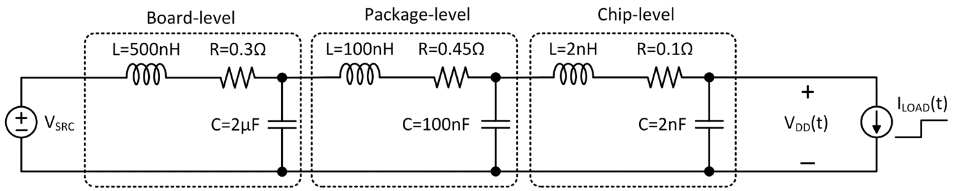
UC Berkeley

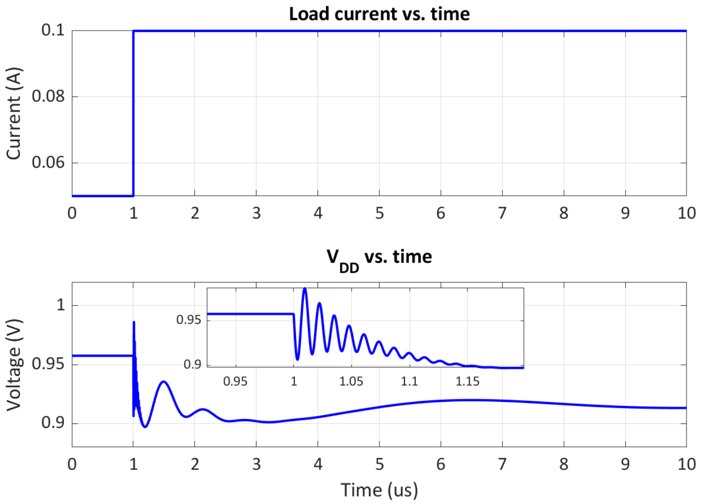
May 4, 2020

Overview

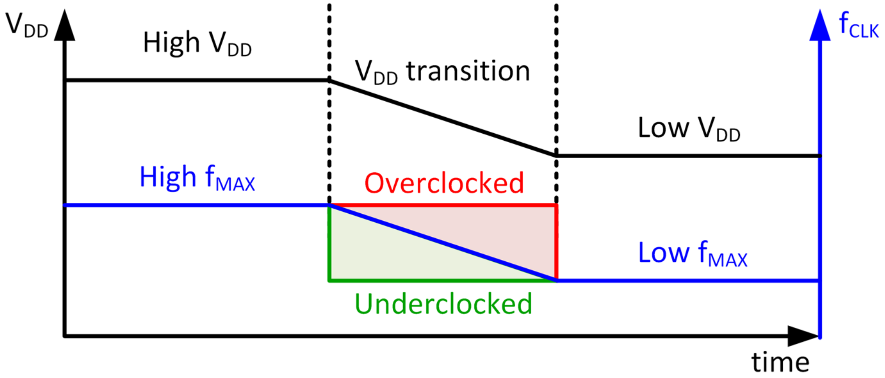
- Motivation
- Adaptive clocking schemes
  - PLL-based adaptive clocking
  - Adaptive Clock Distribution (ACD)
- PLL-based adaptive clocking in ASAP7
- Simulation results
- Summary

Motivation





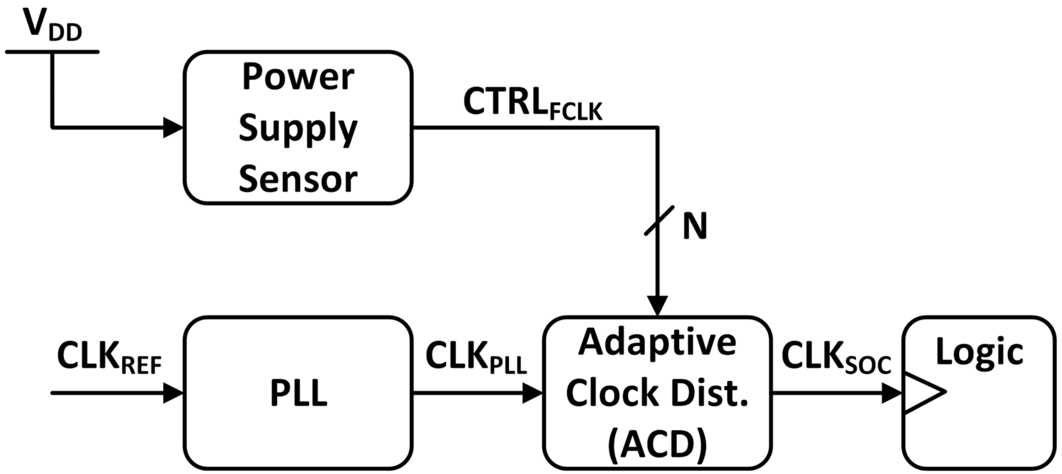
Adapted from T. Hashimoto et al. JSSC 2015



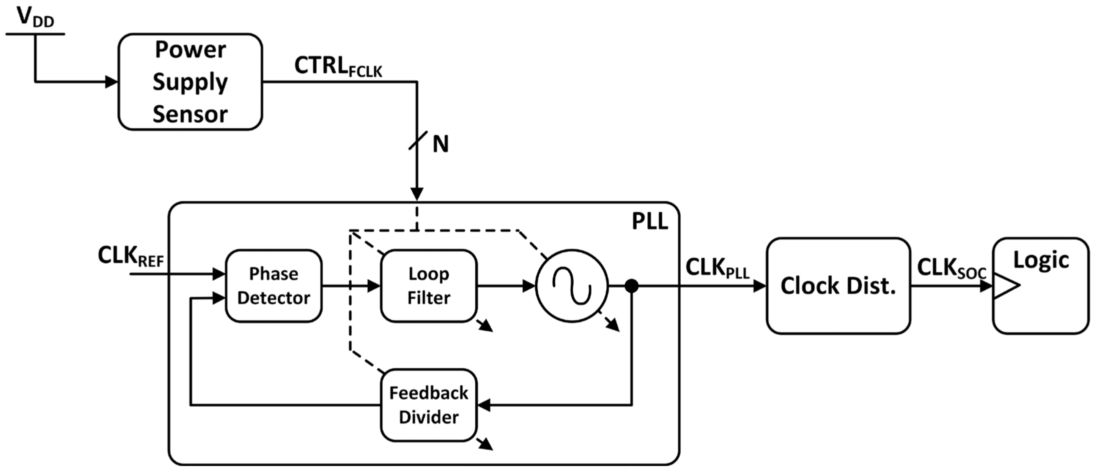
Adapted from F. Ahmad et al. JSSC 2017

- Power management in modern SoCs can enable/disable multiple cores, causing surge currents
- Surge currents cause supply droops through $L\frac{di}{dt}$ ripples from power distribution network
- Objective: Adapt SoC clocking in response to supply droops with minimal guardbanding

Adaptive Clocking Schemes



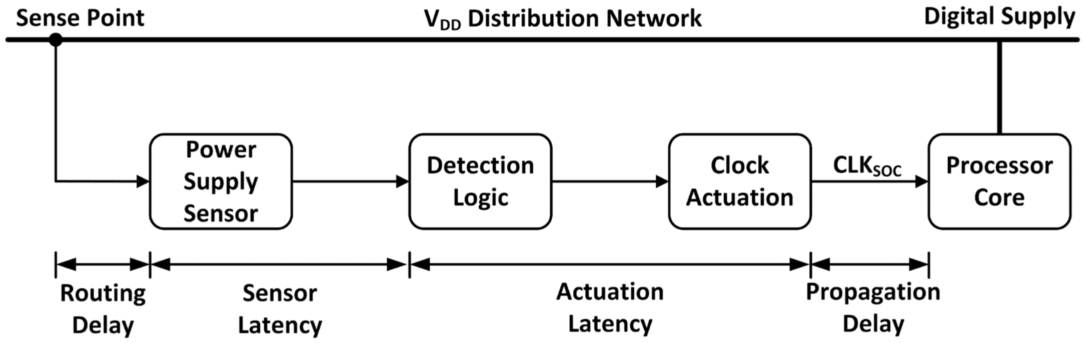
- Power supply sensor detects V_{DD} change
- Clock distribution (post-PLL) adjusts clock frequency by extending period
- E.g. Phase rotators, tunable delay line

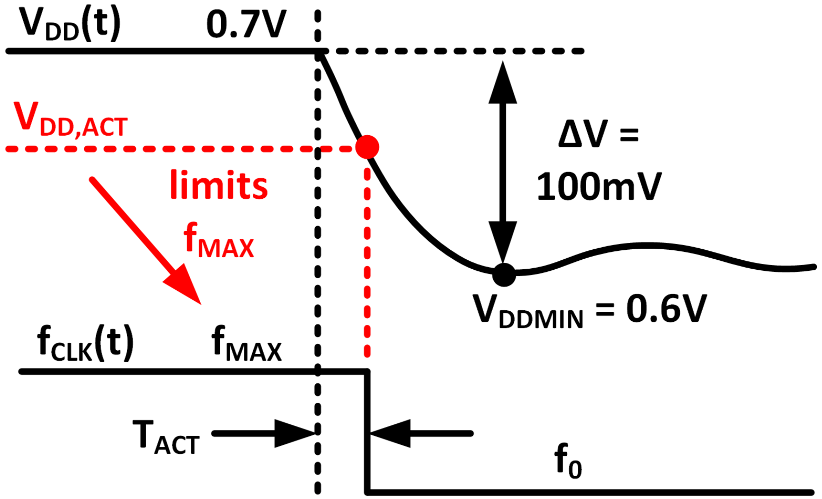


- Power supply sensor detects V_{DD} change
- PLL internally adjusted for frequency
- E.g. direct override of VCO, loop filter, divider

Actuation latency (i.e. adaptation time) is a key metric

Actuation Latency Penalty

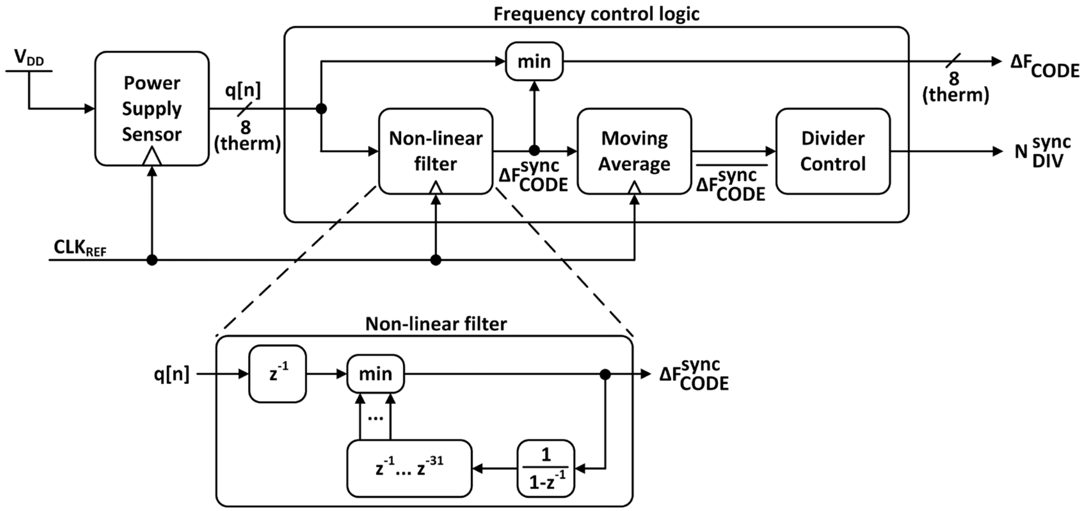


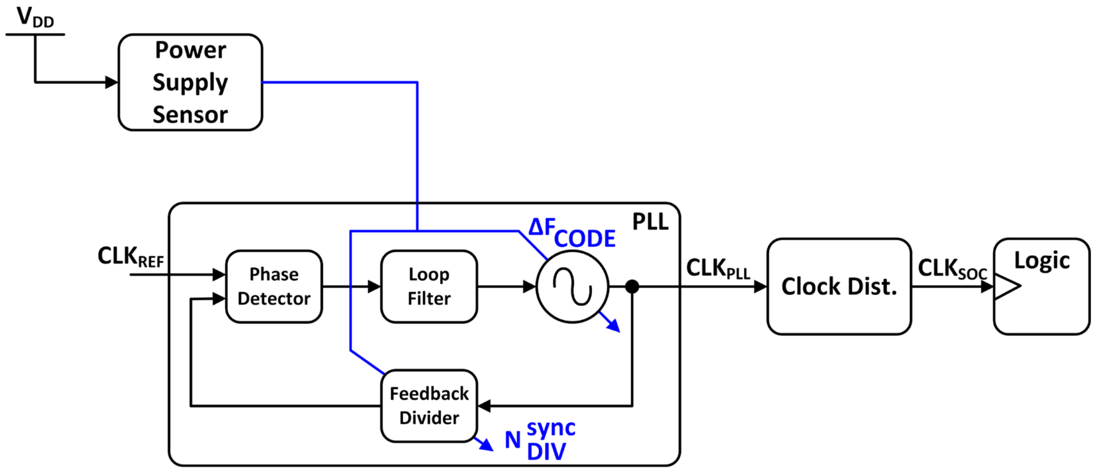


At fixed V_{DD}: Longer T_{ACT} → lower $V_{DD,ACT}$ → lower f_{MAX}

At fixed f_{MAX}: Longer T_{ACT} → lower $V_{DD,ACT}$ → higher V_{MIN}

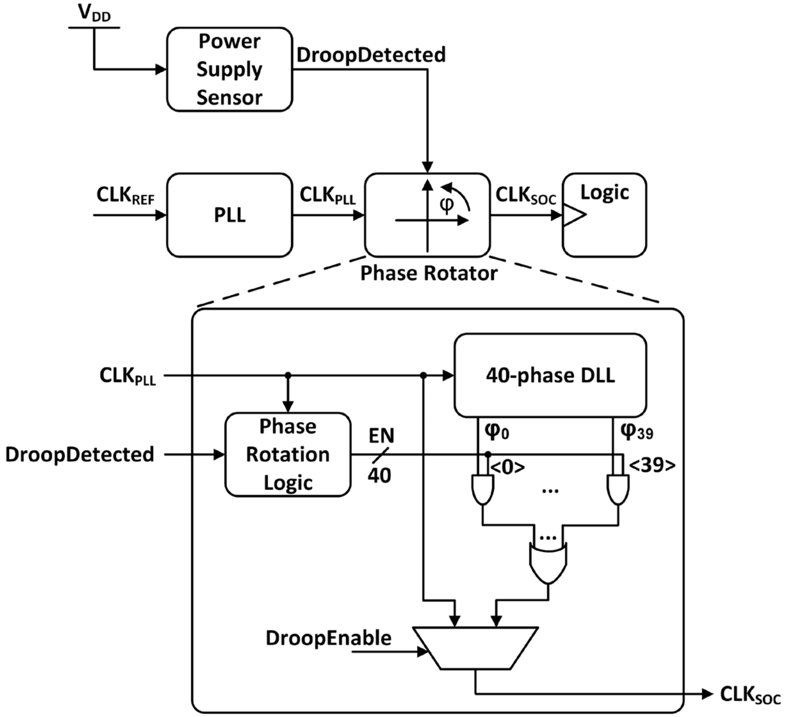
PLL-based adaptive clocking
Hashimoto et al., JSSC 2015 (20nm CMOS)





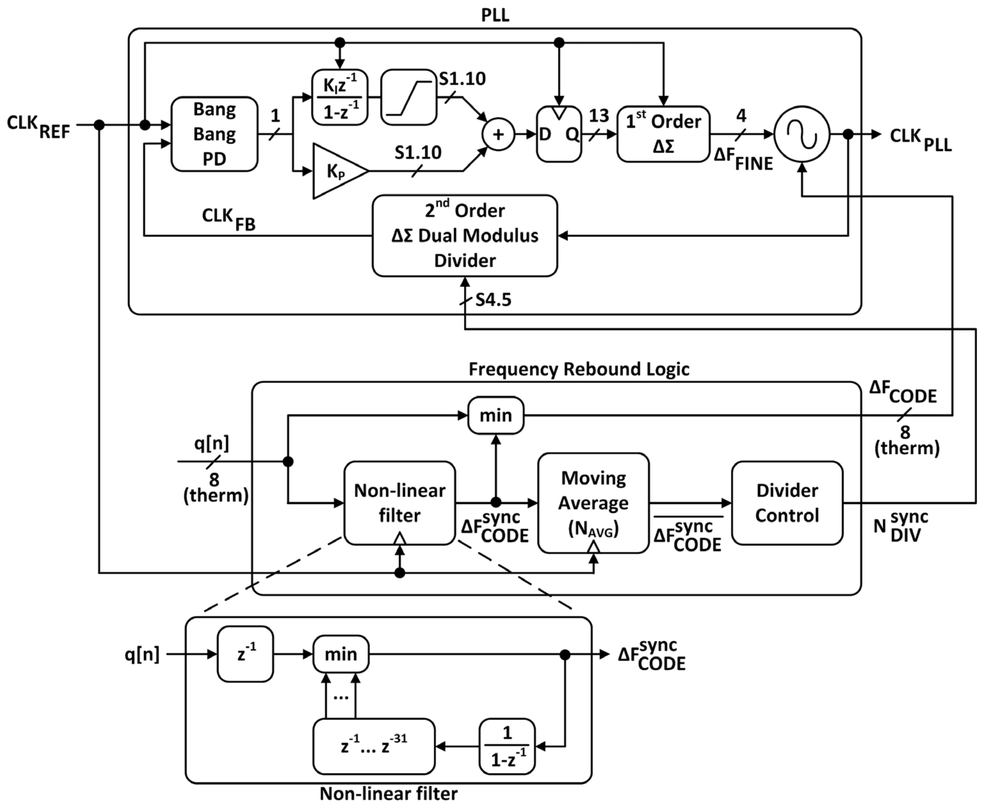
- ΔF_{CODE} immediately adjusts DCO in response to sudden droop
  - Sub-cycle logic latency + thermometer-coded logic (single AND gate)
- N_{DIV}^{SYNC} filters droop through non-linear filter + moving average to track slower droops
  - Helps re-lock PLL to nominal frequency after droop event
- Low-latency logic claimed to improve f_{MAX} by 7.5% (4.65GHz → 5.00GHz) or 5% V_{MIN} reduction at 4.65 GHz

Adaptive clock distribution
Wilcox et al., JSSC 2015 (28nm CMOS)

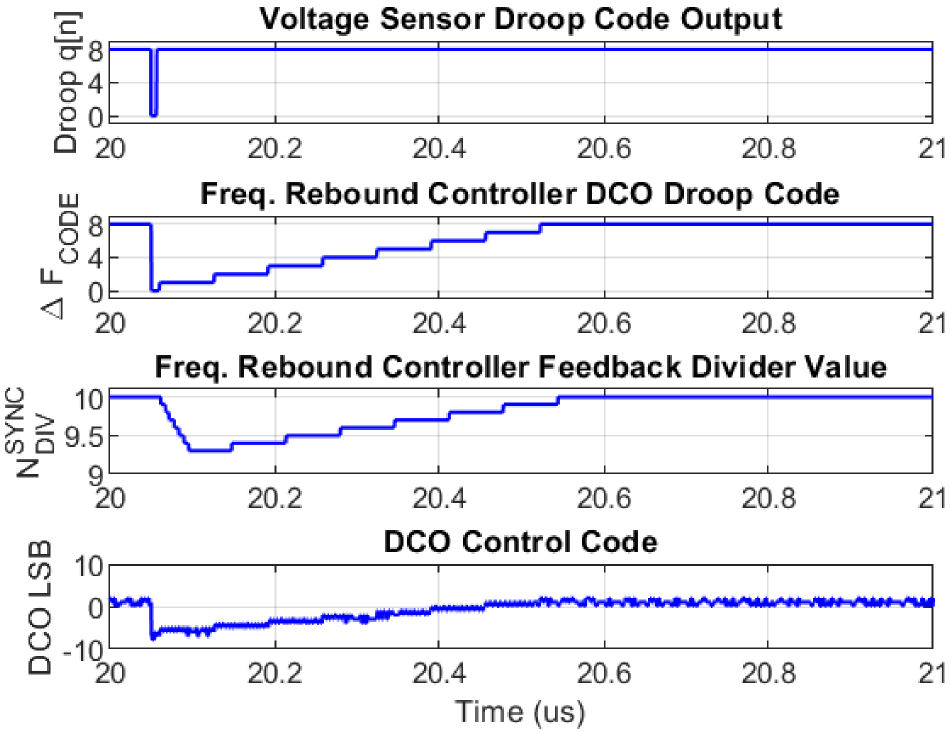


- Phase rotator continuously extends period of CLK_{PLL} to generate lower-frequency CLK_{SOC}
- 40-phase DLL generates desired inputs to phase rotator mux
- Longer actuation latency (3 clock cycles) than Hashimoto et al.
- Up to 6% V_{MIN} reduction reported at 4 GHz

PLL-based adaptive clocking in ASAP7

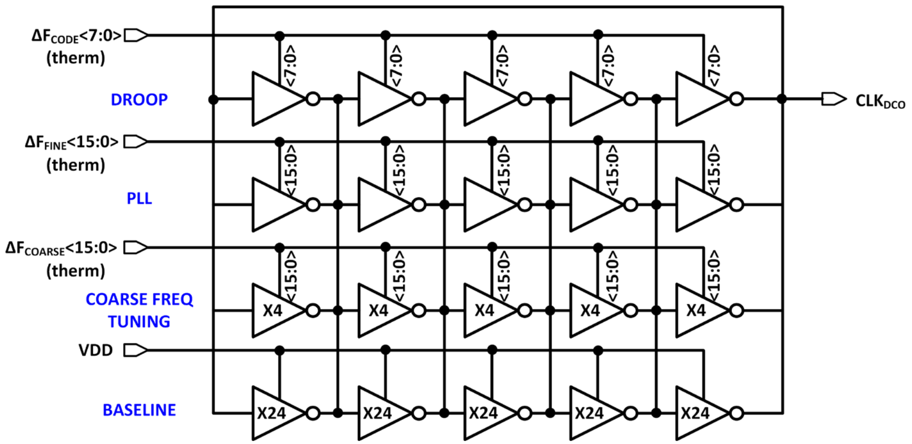


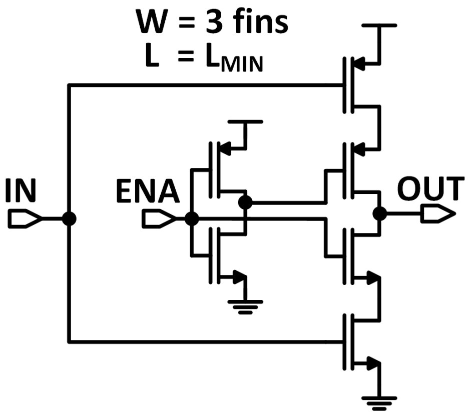
- K_{DCO} = 60MHz/LSB (from PEX)
- $K_P = 2^0$, $K_I = 2^{-9}$
- BW = 810kHz, PM = 79°
- Logic latency = 40ps, DCO latency = 171ps (from PEX)

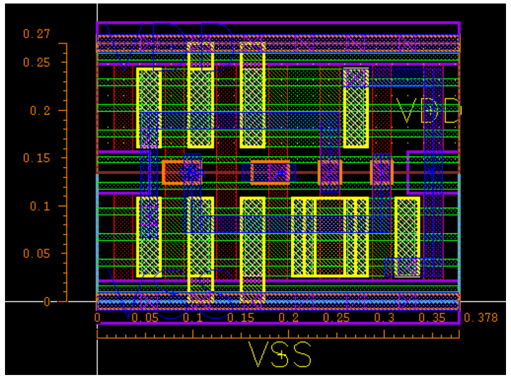


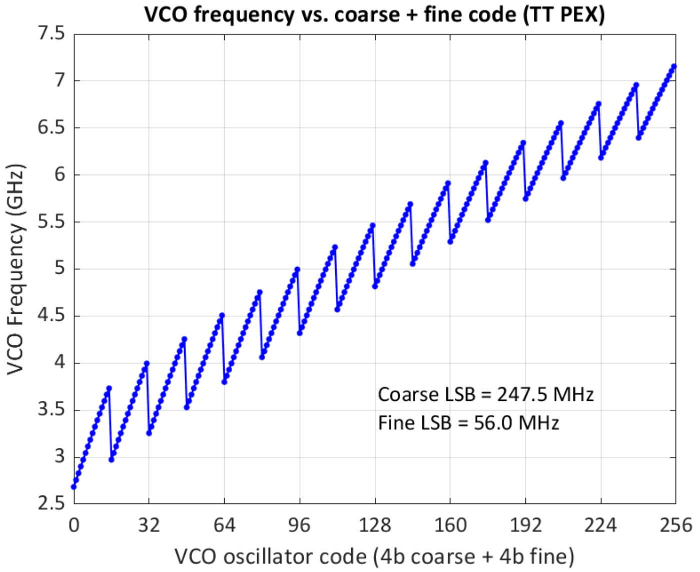
System-level simulation (MATLAB) with critical delays back-annotated from PEX

C2MOS DCO Implementation

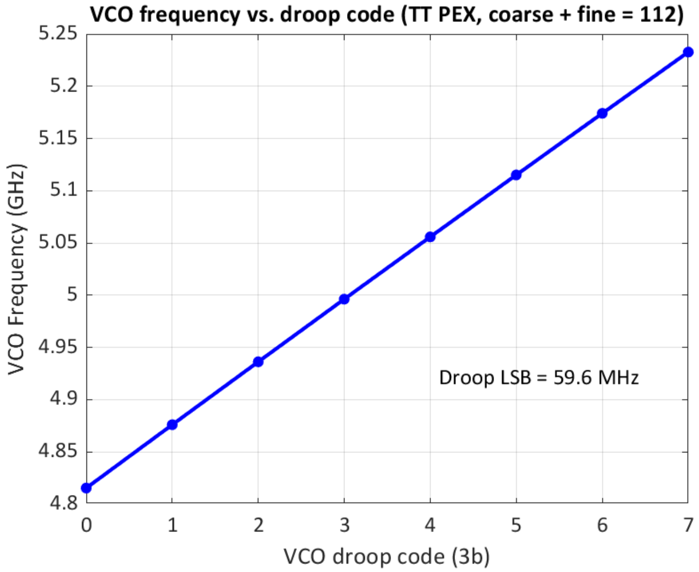






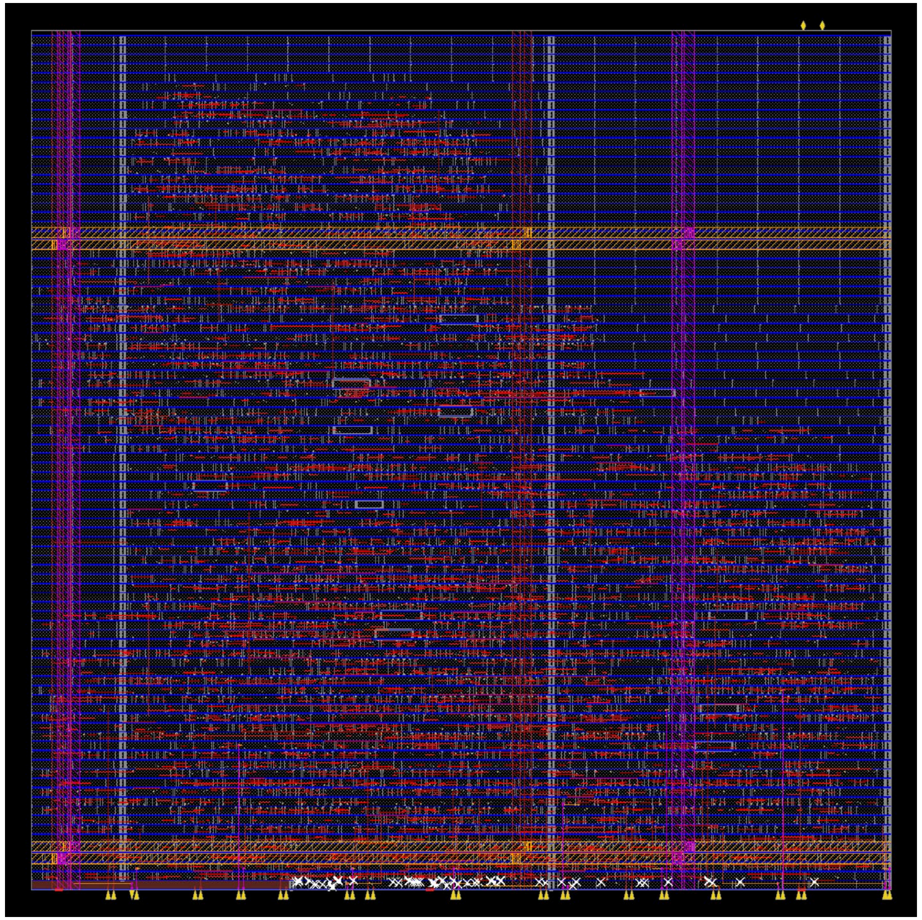


4-bit coarse DAC (initial lock)
4-bit fine DAC (PLL)
3-bit droop DAC

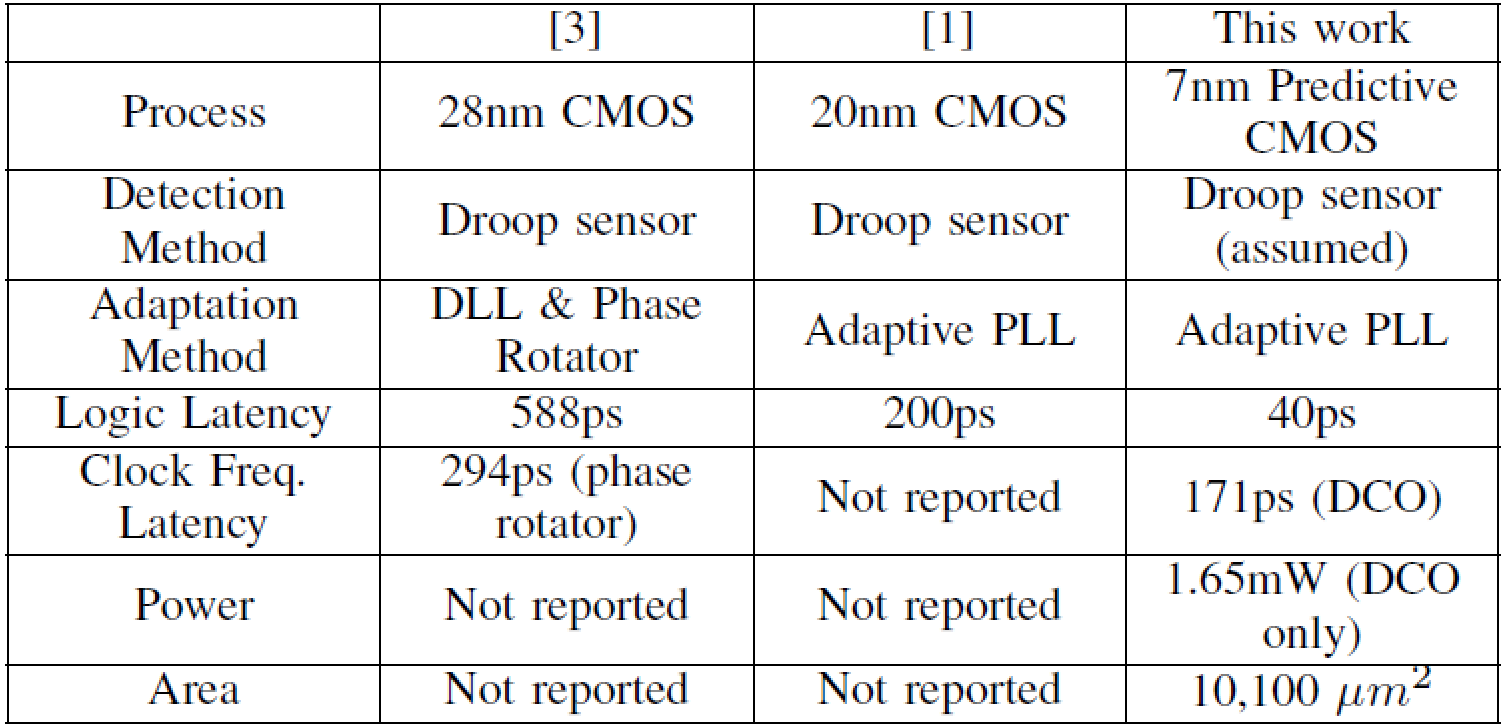


RTL Implementation?

Backend implementation



Comparison to Prior Works



	[3]	[1]	This work
Process	28nm CMOS	20nm CMOS	7nm Predictive CMOS
Detection Method	Droop sensor	Droop sensor	Droop sensor (assumed)
Adaptation Method	DLL & Phase Rotator	Adaptive PLL	Adaptive PLL
Logic Latency	588ps	200ps	40ps
Clock Freq. Latency	294ps (phase rotator)	Not reported	171ps (DCO)
Power	Not reported	Not reported	1.65mW (DCO only)
Area	Not reported	Not reported	10,100 μm^2

Summary