

Asynchronously Controlled Frequency Locked Loop

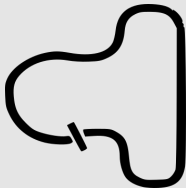
async techniques in a sync system

Suwen Yang, Frankie Y. Liu, Vincent C. Lee

May 9 2016

The Big Picture

Circuit



Rate

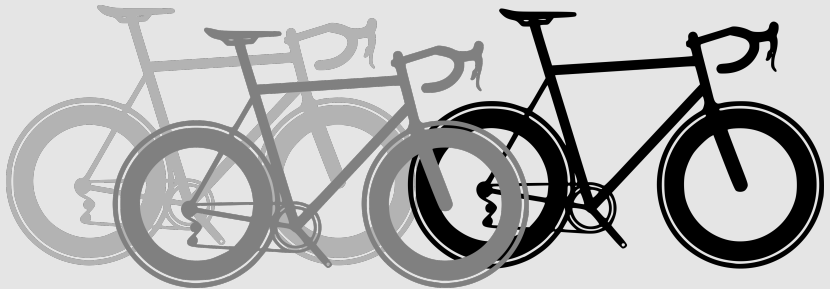


Optimize performance



The Big Picture

Matching speeds



Applications and prior

Apps

- PLL : sep. phase and freq loops, or coarseness
- Dist power grid sync : small freq. variations
- Frequency synthesis : lower loop order
- Capacitance measurement : measure osc. freq

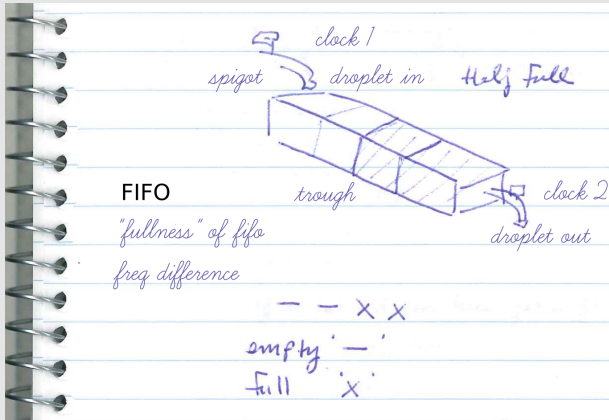
Applications and prior

Prior

- Counters
- Sigma-delta noise shaping modulators
- Frequency-to-voltage conversion
- Second-order integrators

Asynchronous FIFO for frequency detection

Spigots in Minecraft



Symbology

Tic tac toe

Time lapse

"Trick
down" X →

- - - - x x x x

⊗ - - - x x x x

- ⊗ - - - x x x x

- - - ⊗ x x x x

- - - - ⊗ ⊗ x x ← ⊗ "bubble up"

x x x x x x x x

"full"

- - - - - - - -

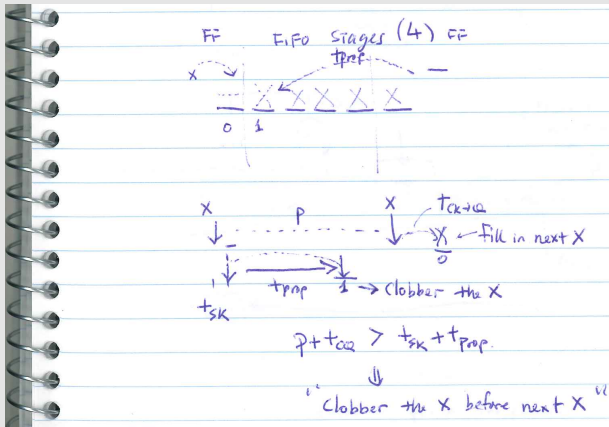
"empty"

- - - - x x x x

"half"

Clobber

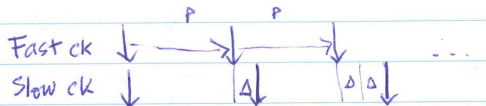
Wack the mole



How long before an update

Check, please

How long for detection



$\frac{P}{\Delta}$ cycles before getting a whole period

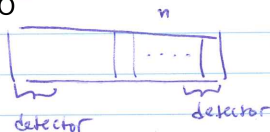
Tradeoff
fewer stages

$$\text{time} : P \cdot \frac{P}{\Delta} = \frac{P^2}{\Delta}$$

+ shorter resol time

- phase insensitivity

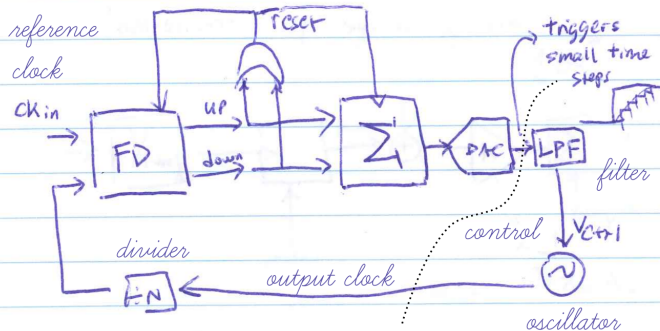
FIFO




$$T_w \propto \frac{n \cdot P^2}{\Delta}$$

FLL with the frequency detector

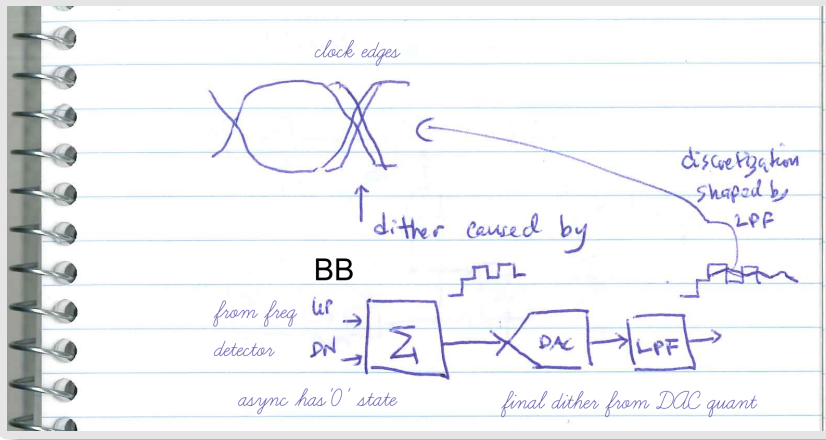
In a system



emulate  $V_{ctrl} \rightarrow$ period
event driven simulation every $\frac{1}{2}$ period the period
in system verilog is updated.

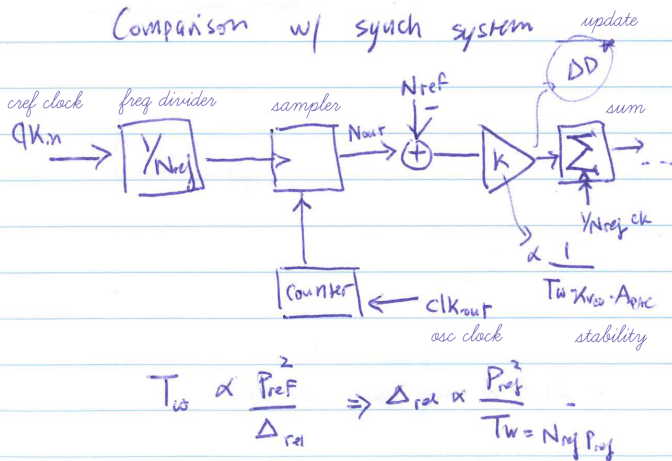
Dither

What is that fuzz



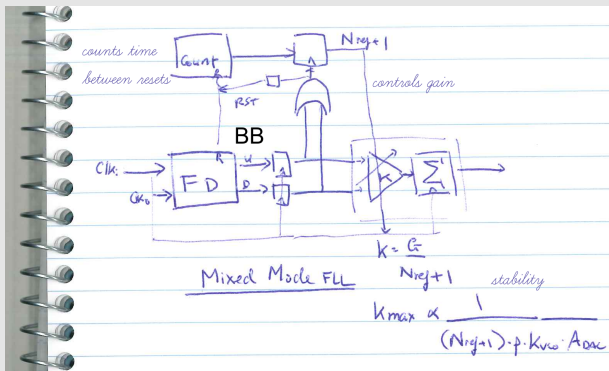
Synchronous counting

By the clock



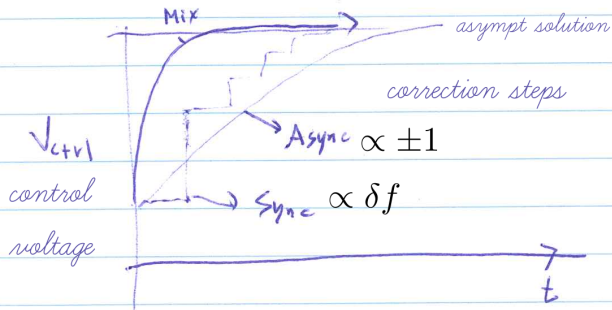
Mixed Mode FLL

Lock faster



Time traces for oscillator control voltage

Show and tell



Key: Time between updates
used to change gain in feedback

Ending

There is no place like home

- Pro: simple design FLL, resolution not set by large number of bits, fixed “space” requirement
- Pro: not sensitive to phase offset between the two signals
- Pro: static jitter set by DAC quantization error
- Pro: fast locking via gear shifting, done in an asynchronous manner, goes with the flow
- Con: adoption of non time-synchronous solution

Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for informational purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Questions ?

Frankie Liu
frankie.liu@oracle.com