BLENDICS

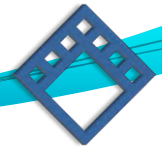
# Synchronizers And Data Flip-Flops are Different


Jerome Cox
Blendics Inc.
St. Louis, Missouri, USA
jcox@blendics.com

George Engel
Southern Illinois University
Edwardsville, Illinois, USA
gengel@siue.edu

David Zar
Blendics Inc.
St. Louis, Missouri, USA
dzar@blendics.com

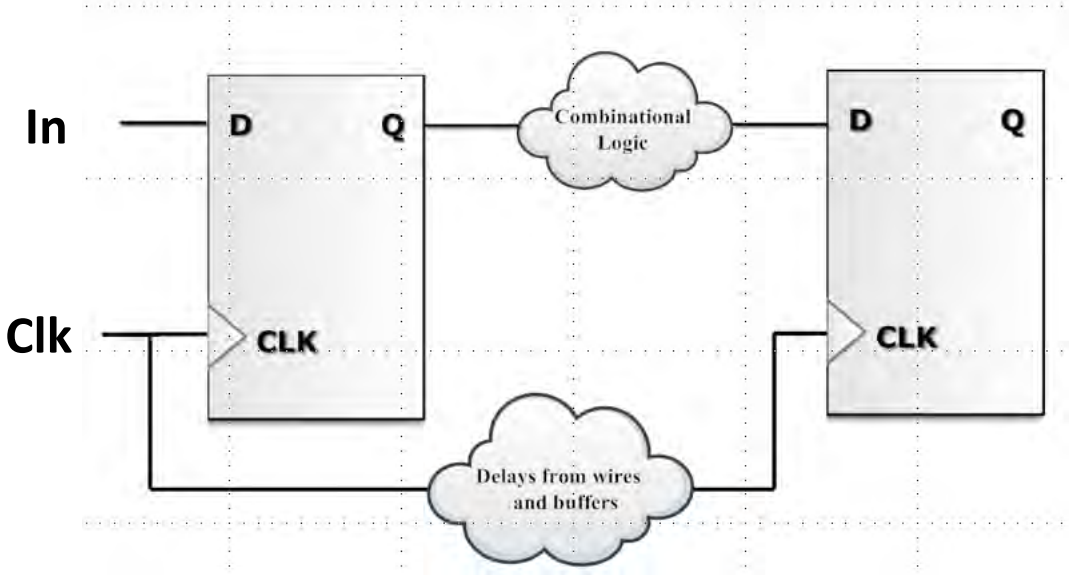
Ian W. Jones
Oracle Labs
Redwood Shores, California, USA
ian.w.jones@oracle.com



# Outline

- Data vs. Synchronizer Flip-Flop
- Flip-Flop Hazards and Mitigations
- Benchmark Synchronizer FF Design
- Performance of Benchmark Synchronizer FF
- Using MetaACE_LTD for Analysis
- Summary

# Data Flip-Flops Vs. Synchronizer Flip-Flops



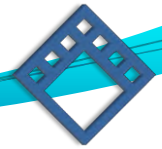

Different performance characteristics to optimize based on FF use:

| | $t_{pd}$ | $t_{su}$ | $t_h$ | $\tau$ | $T_w$ |
|---|---|---|---|---|---|
| Data FF | minimize | minimize | 0 | - | - |
| Synchronizer FF | - | - | 0 | minimize | minimize |

# Flip-Flop Hazards

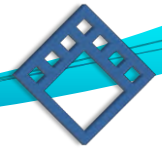
- Types
  - Uncertainty in transition timing
  - Clock/data skew
  - Uncertainty in logic level
- Hazards Mitigated In
  - Data FF
    - $t_{clk} > t_{su} + t_{pd} + t_{cl\text{-}d} + t_k$
  - Synchronizer
    - Mean Time Between Failure (MTBF)

$$MTBF = 1/\Pr(failure) = \frac{e^{t_s/\tau}}{T_w f_c f_d}$$



# Use of Data and Synchronizer Flip-Flops

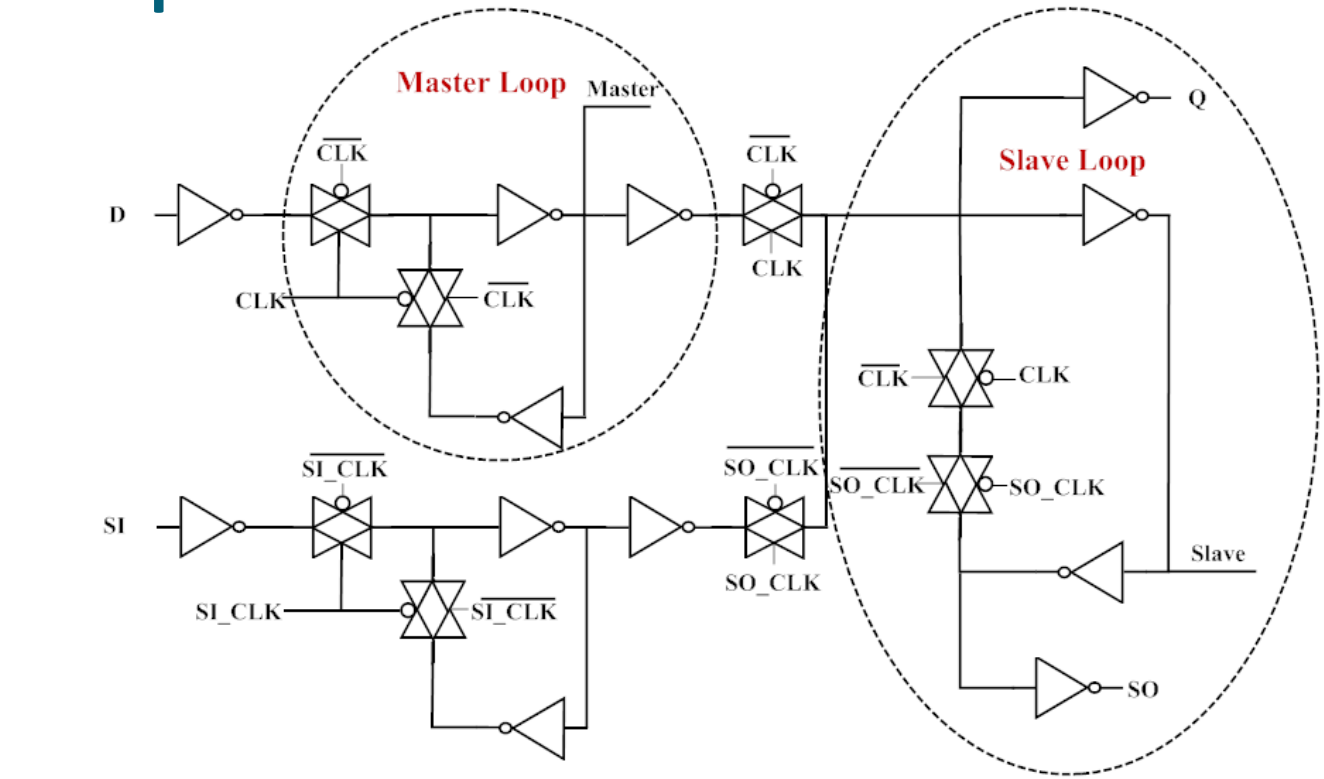
- Data Flip-Flop
  - Temporary storage of data
    - Prevent data values from corruption during a clock cycle
    - Hold data values for multiple clock cycles
  - Deterministic cycle-to-cycle operation
    - Implies large setup/hold times
- Synchronizer Flip-Flop
  - Minimize *Pr(failure)*
    - Data/clock may arrive at any time which may cause a setup/hold violation at a following data flip-flop
  - Preserve data transition sequence
    - No guarantee of deterministic cycle-to-cycle timing



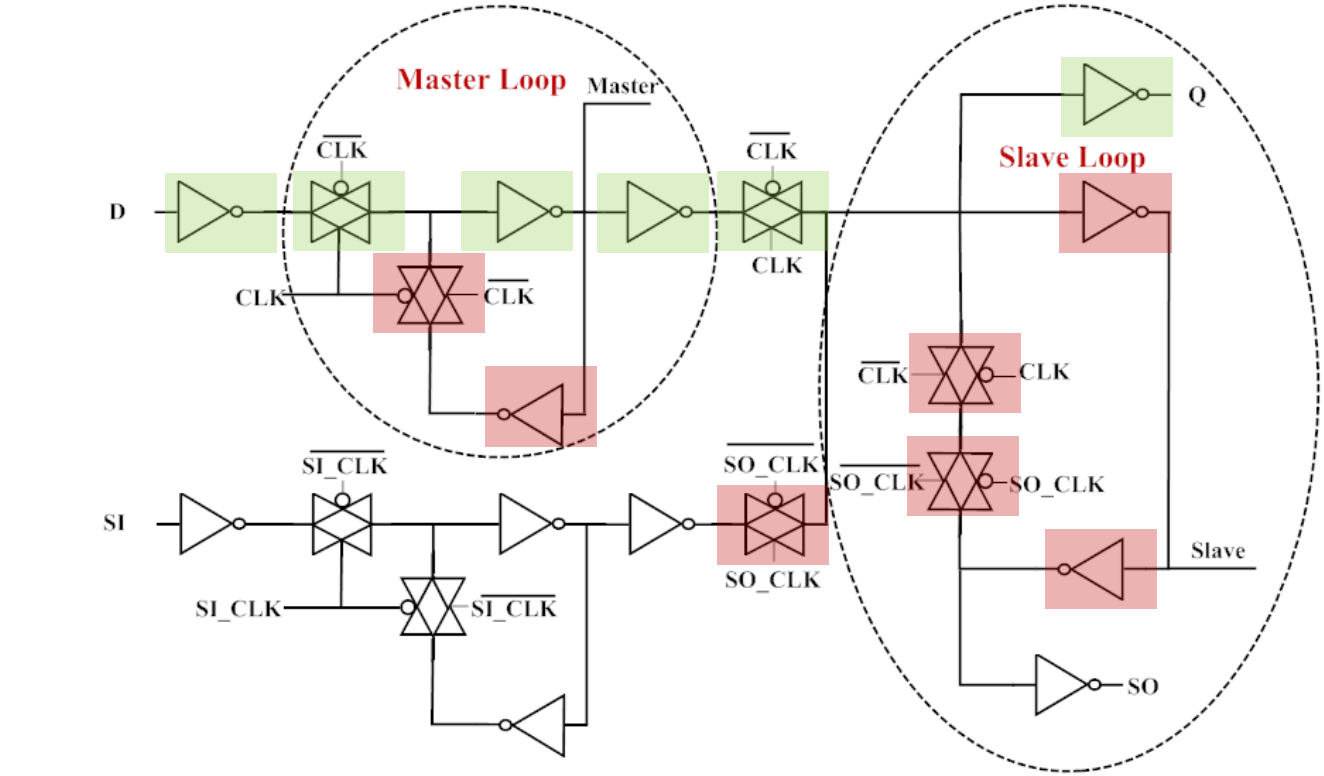
# Benchmark Synchronizer FF

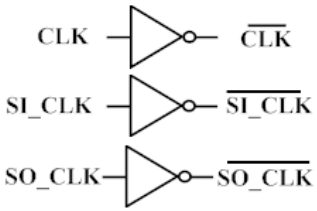
- Synchronizer FFs are not the same as data FFs
  - Show how to design a good FF for use as a synchronizer.
- Metastability/synchronizer design is not well-understood by many engineers and their managers
  - An example circuit will make it easier for engineers to understand good synchronizer design and the pitfalls they are likely to encounter when trying to estimate metastability-related MTBF rates.
- Metastability related failures are likely to increase as process variability increases.
  - Time to get better and understanding the risks and design options.

# Synchronizer Cell Candidate: A Data Flip-Flop with Scan Chain

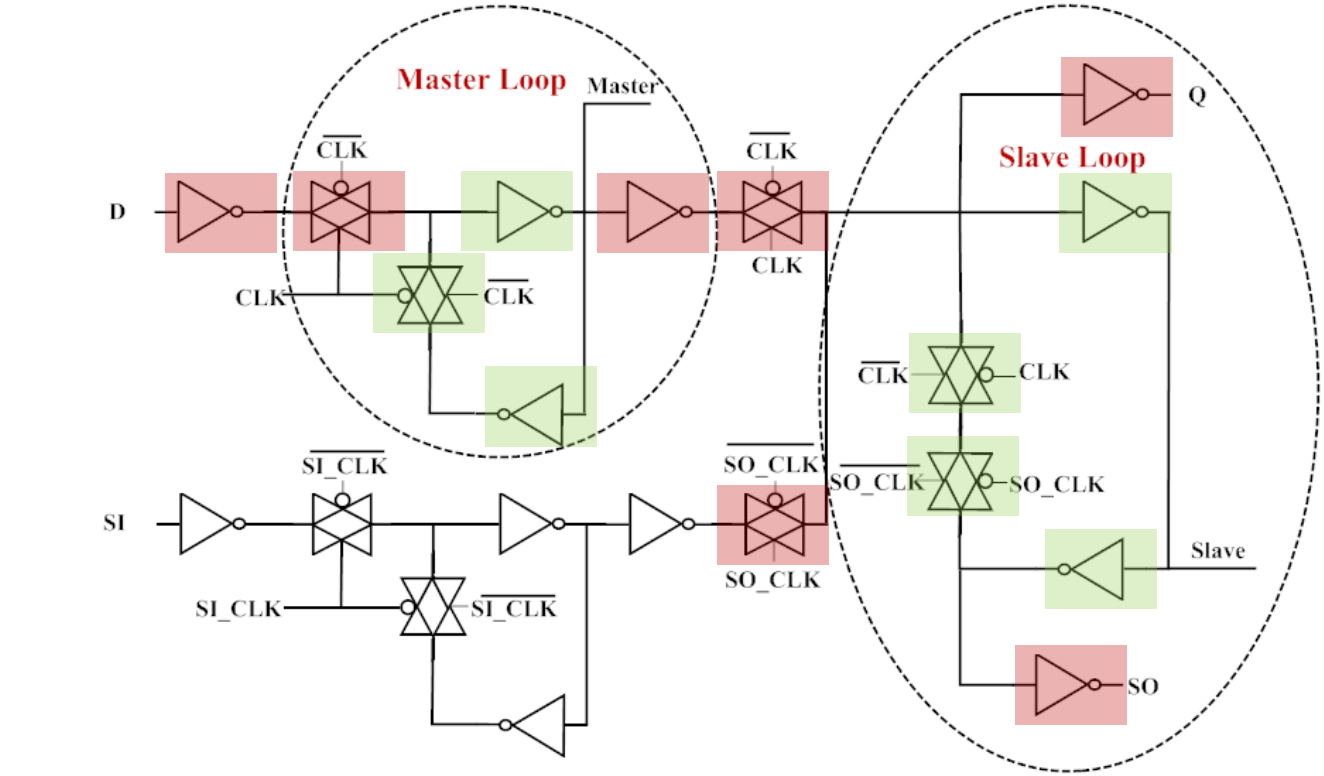


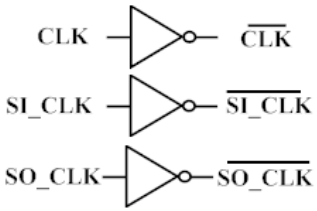
# Optimize For Data Use



# Optimize for Synchronizer Use



# Gain-Bandwidth Product and $\tau$

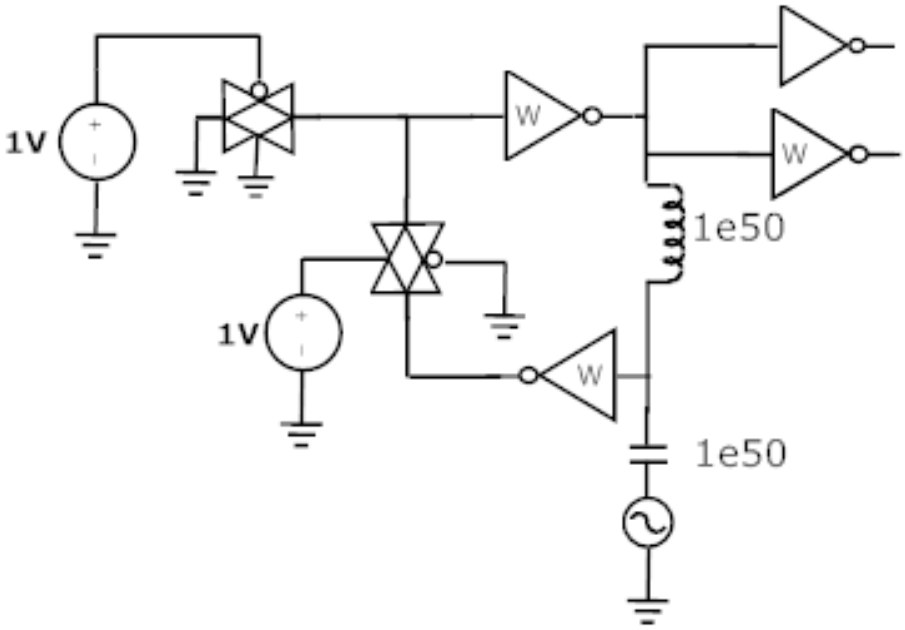
- Converting a data FF into a synchronizer FF requires maximizing the gain-bandwidth product (GBW) of the regenerative loops in the master and slave latches.
- Gain-Bandwidth Product and $\tau$ are inversely proportional

$$\tau = \left[r\sqrt{A_{DC}}\right]\frac{1}{2\pi GBW}, r = \frac{C_{L2}}{C_{L1}+C_{L2}}$$

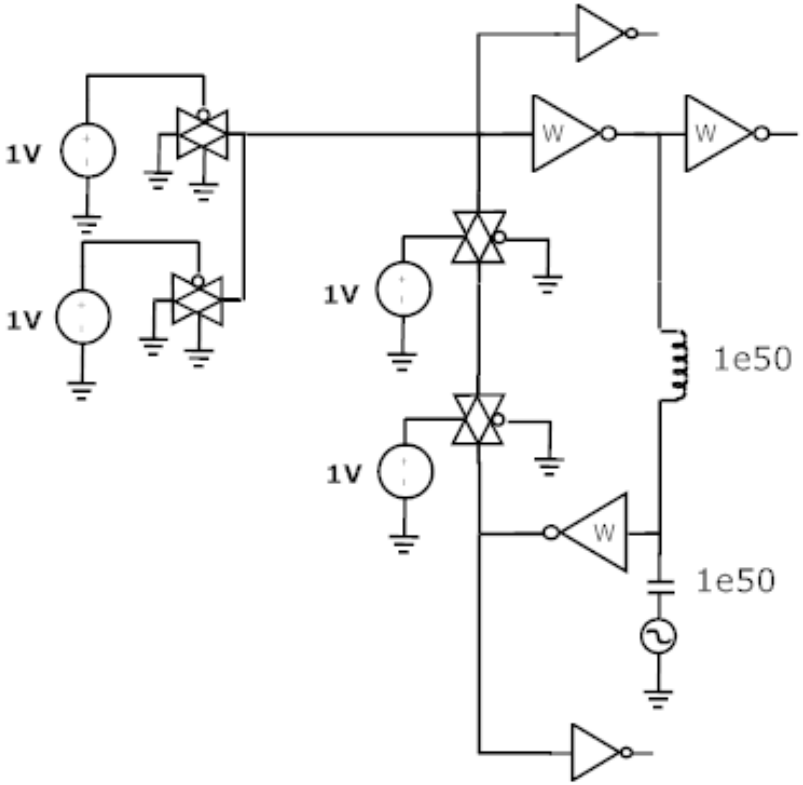
- This relation is used to size the devices in the regenerative loops
- Through use of a small signal analysis, a near-optimum size for loop devices can be determined. This method gives engineers a good method to design a high-quality synchronizer.

# AC Analysis to Maximize GBW

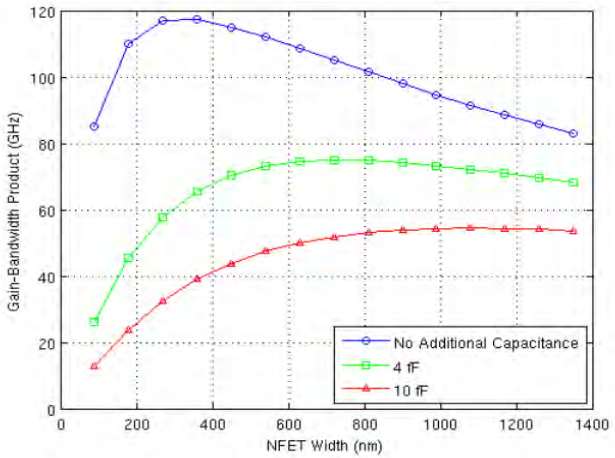
MASTER



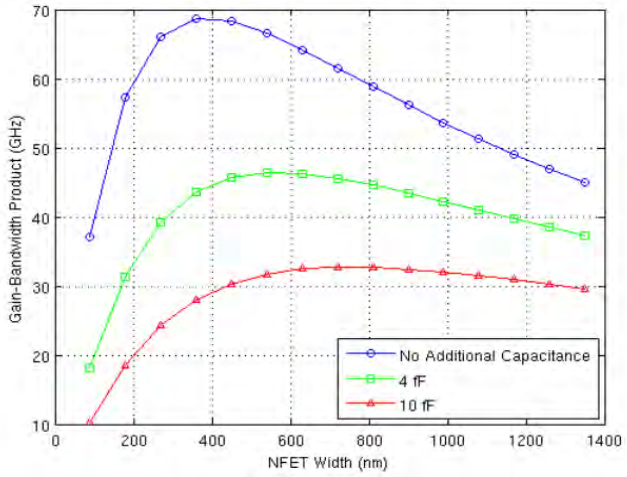

SLAVE



# GBW as a Function of Device Width




*GBW Vs. Device Width: Master*



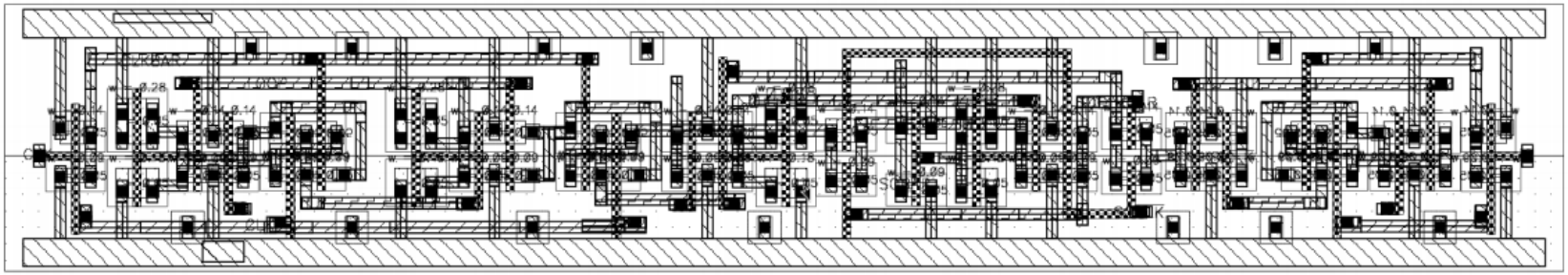

*GBW Vs. Device Width: Slave*

| NFET Device Width | Master | Slave |
|---|---|---|
| 90 nm | 85 GHz | 37 GHz |
| 180 nm | 110 GHz | 57 GHz |
| 270 nm | 117 GHz | 66 GHz |
| 360 nm | 117 GHz | 69 GHz |
| 450 nm | 115 GHz | 68 GHz |

| | Master | Slave |
|---|---|---|
| No capacitance | 117.1 GHz | 66 GHz |
| 1 fF | 93.75 GHz | 56.4 GHz |
| 2 fF | 77.8 GHz | 49.2 GHz |
| 4 fF | 57.8 GHz | 39.2 GHz |
| 10 fF | 32.4 GHz | 24.2 GHz |

*At NFET Width =270 um*

# Physical Layout and Performance



- Area: 1.73 μm x 9.6 μm, 15.65 $\mu m^2$
- Performance:

| | Dff, VTG (ps) | VTG (ps) | VTL (ps) |
|---|---|---|---|
| $\tau_M$ | 19 | 14 | 10 |
| $\tau_S$ | 55 | 31 | 19 |
| $\tau_{eff}$ | 28 | 19 | 13 |

$$\tau_{eff} = \left(\frac{\alpha}{\tau_m} + \frac{1-\alpha}{\tau_s}\right)^{-1}$$

α is the duty cycle for the master

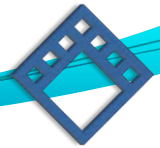
# Summary

- Data and Synchronizer FFs are not the same things.
- A typical data FF circuit can be converted to a good synchronizer circuit.
  - The method uses an AC analysis to optimize the GBW of the cascaded inverters in the critical regenerative loops.
  - Using low Vt devices will also improve GBW.
- The benchmark synchronizer FF design is available to the public as well as *MetaACE_LTD* for analyzing metastability performance.

# Questions

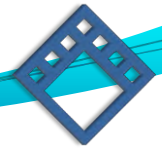
Where do I get the benchmark synchronizer?
http://blendics.com/a-public-synchronizer/

Where do I get *MetaACE_LTD*?
http://blendics.com/forums/



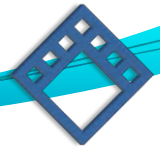
# PVT Tolerant Synchronizer Design

- From equations developed through formal sensitivity analysis the following recommendations can be made for a PVT variation tolerant design:
  - Use the highest supply voltage possible,
  - Use the lowest, available threshold transistors in regenerative loops,
  - Use minimum length FETs in the loops since high-field effects reduce sensitivity,
  - Use transistor widths no wider than necessary since wider devices can force the FETs out of strong inversion, and
  - If available, choose a synchronizer topology with metastable voltages that are insensitive to supply voltage changes.



# FreePDK45

- Purposely non-manufacturable 45 nm “process”
- Predictive HSPICE models from ASU
- PDK (Process Design Kit) from NCSU for Cadence IC6 toolset
- Used by researchers to explore device performance and design flows in deep sub-micron processes
- Three threshold voltages available
  - VTL → Low threshold → High-speed
  - VTG → Normal threshold → General-purpose
  - VTH → High threshold → Low-power



# *MetaACE_LTD*

- MetaACE_LTD is a free version of *MetaACE* which is a tool that allows easy determination of parameters needed for analysis of MTBF.
- Automates hundreds of simulations, tabulates results and computes τ and Tw.
- Sweeps voltage and temperature for corner analysis.
- Supports any circuit where SPICE circuit and transistor-level models are available.
- Can be run from the command line with configuration files for inclusion into a verification/extraction flow.
- Has been verified against silicon across a range of voltage and temperature.