Asynchronous Computing in Sense Amplifier-based Pass Transistor Logic

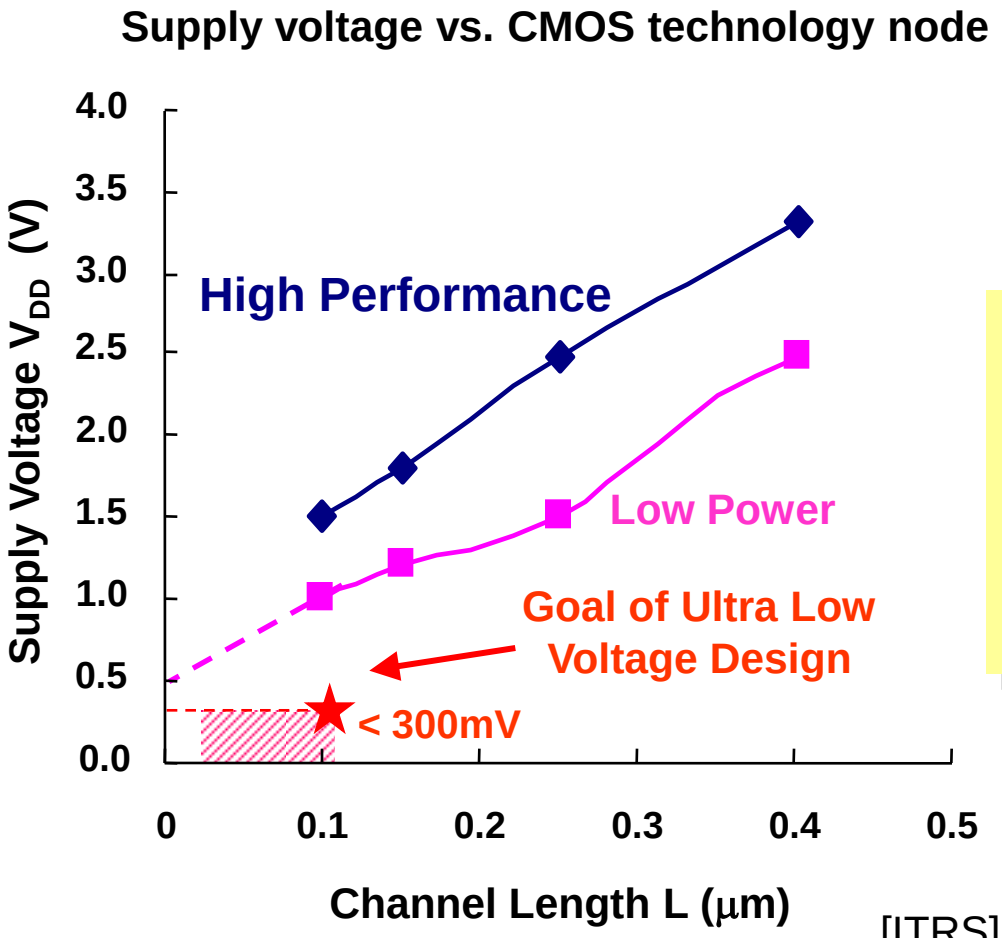
Tsung-Te Liu, Louis P. Alarcón, Matthew D. Pierson, and Jan M. Rabaey

University of California, Berkeley

Outline

- Motivation
- Sense Amplifier-based Pass Transistor Logic (SAPTL)
- Self-timed SAPTL
  - Bundled-data self-timed SAPTL design
  - Dual-rail self-timed SAPTL design
- Simulation results
  - Area
  - Energy-delay
  - Leakage
- Conclusion

Focus: Ultra Low Voltage Design

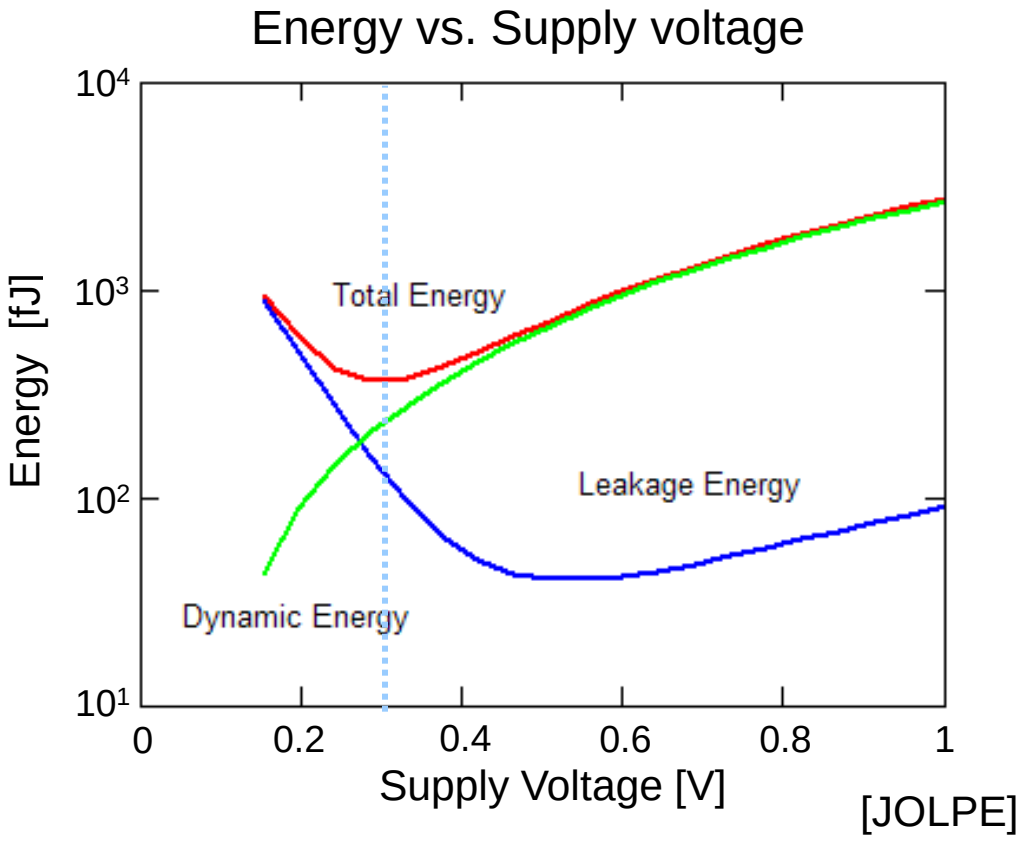


[ITRS]

Voltage scaling:

- Most effective technique for **power reduction**
- But at low supply voltage
  - Leakage dominates
  - Delay is large and variable

Issue-1:
Leakage dominates at low supply voltage

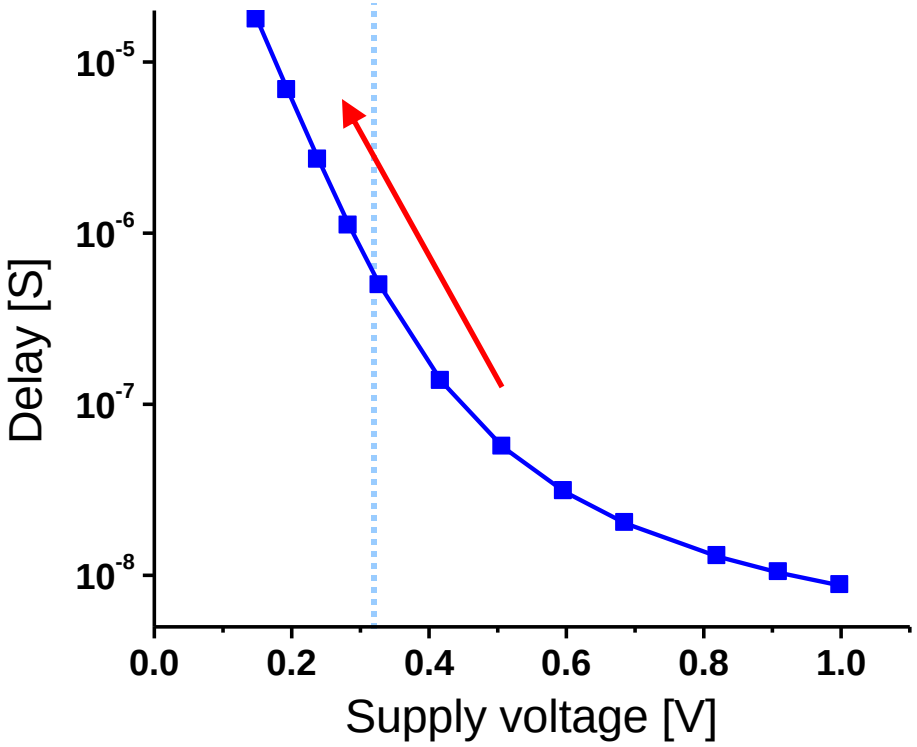


- 203-stage NAND2 ring oscillator
- 90-nm CMOS standard cell

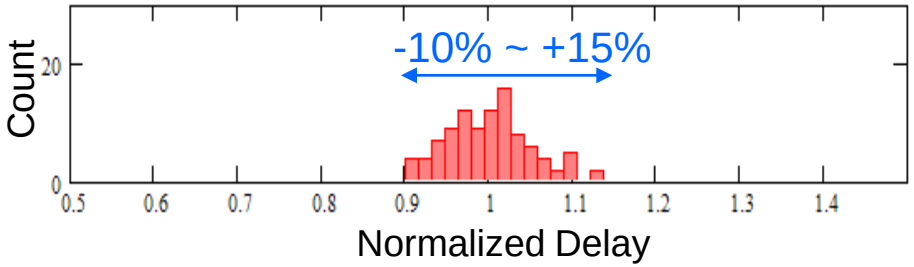
- Leakage dominates the power and energy at low V_{DD}
- Leakage determines standby power for portable devices

Issue-2:
Delay is large & variable at low supply voltage

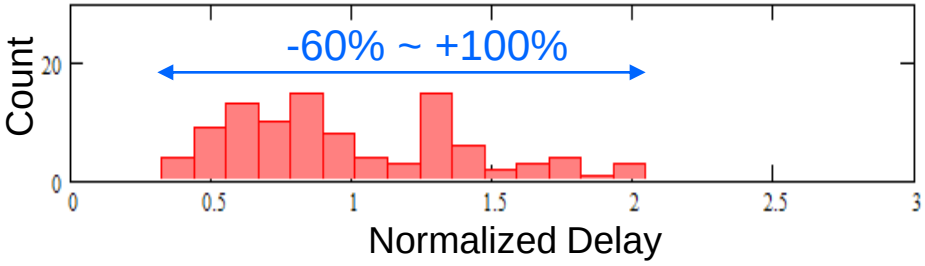
Delay vs. Supply Voltage



Delay spread due to transistor mismatches



Delay spread due to process variations



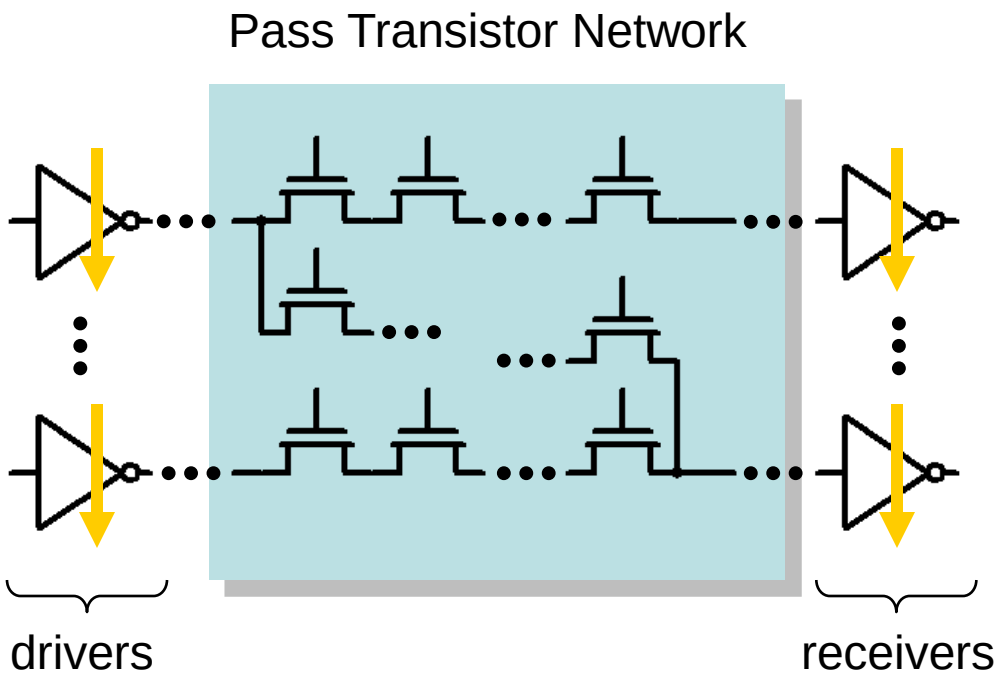
- Supply = 300mV

- Delay increases exponentially at low V_{DD}
- Large delay spread in actual technology environment

Outline

- Motivation
- Sense Amplifier-based Pass Transistor Logic (SAPTL)
- Self-timed SAPTL
  - Bundled-data self-timed SAPTL design
  - Dual-rail self-timed SAPTL design
- Simulation results
  - Area
  - Energy-delay
  - Leakage
- Conclusion

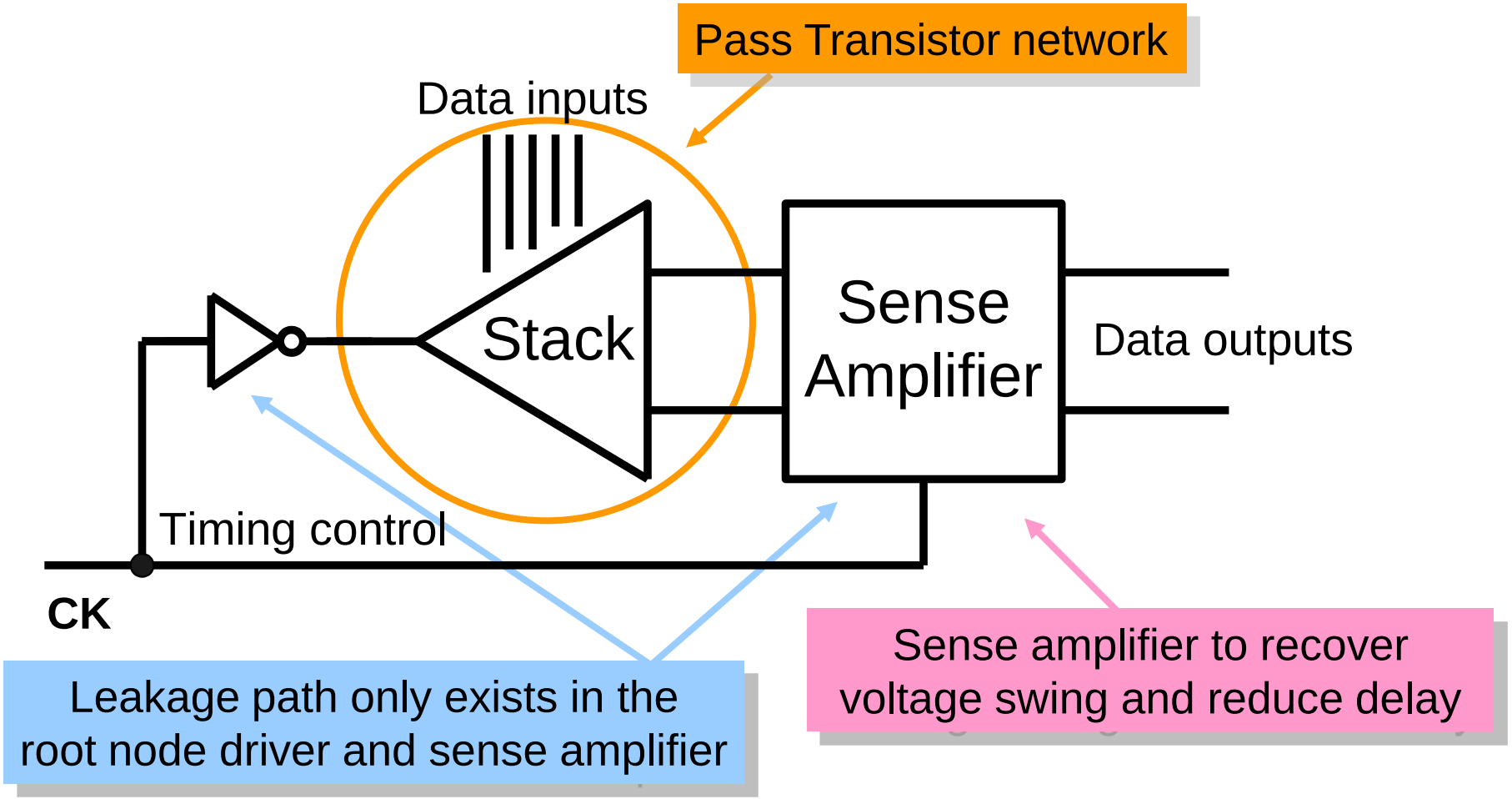
Solution: Use pass transistor logic!



- No VDD and GND connections in the logic path
  - Nearly zero leakage through the logic path
  - Can keep scaling down the threshold voltage

[JOLPE]

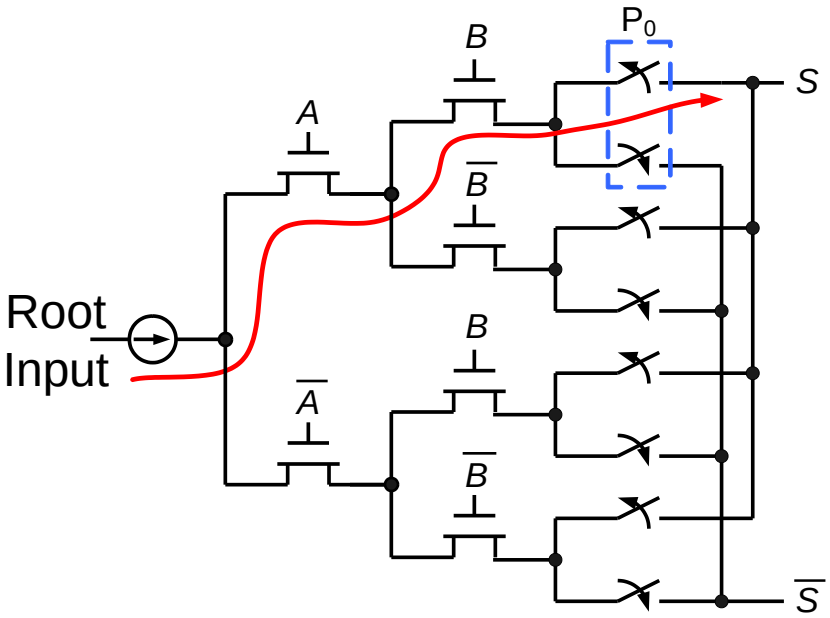
SAPTL:
Sense Amplifier-based Pass Transistor Logic



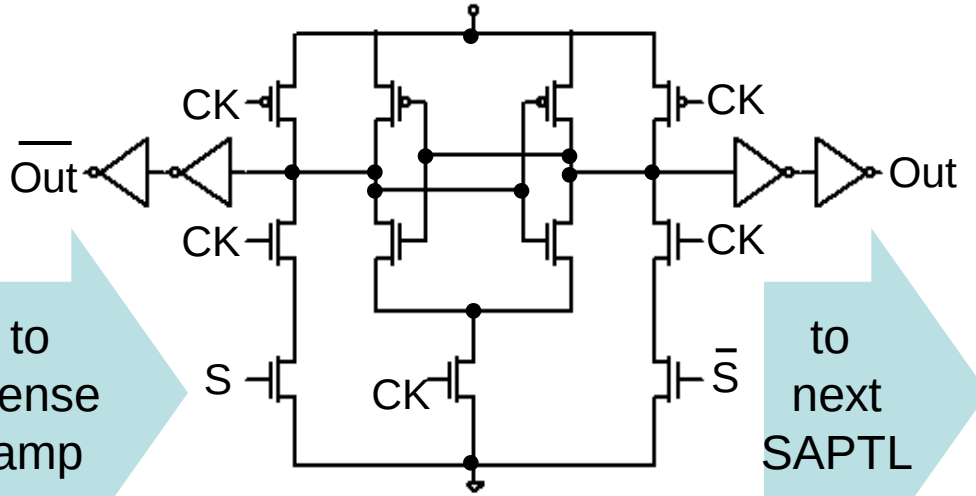
[JOLPE]

SAPTL Computation:
to contain leakage while reducing threshold

Stack



Sense Amplifier

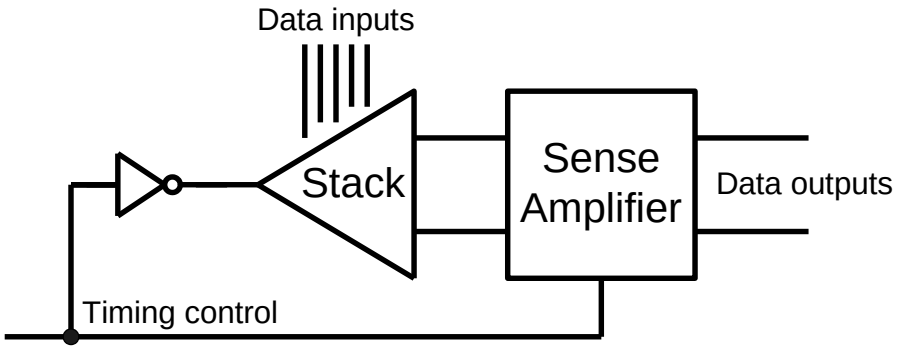


- Current steering
- Works for very low I_{on} to I_{off} ratio
- Could be programmable

- Outputs pre-charge to V_{DD} to reset
- Latch retains the computation results
- Low voltage operation down to 300mV

[JOLPE]

SAPTL Control:
to deal with delay variation & uncertainty



Fact:

- Computation delay varies in actual technology environment
  - I_{on} to I_{off} current ratio varies and changes the stack delay
  - Offset variations in the sense amplifier reduce the stack output margins
- Energy efficiency is function of timing accuracy
  - Turn off the sense amplifier when inactive
- Synchronous timing is too conservative
  - Unable to track local delay variations
  - Additional skew and jitter in the clock distribution

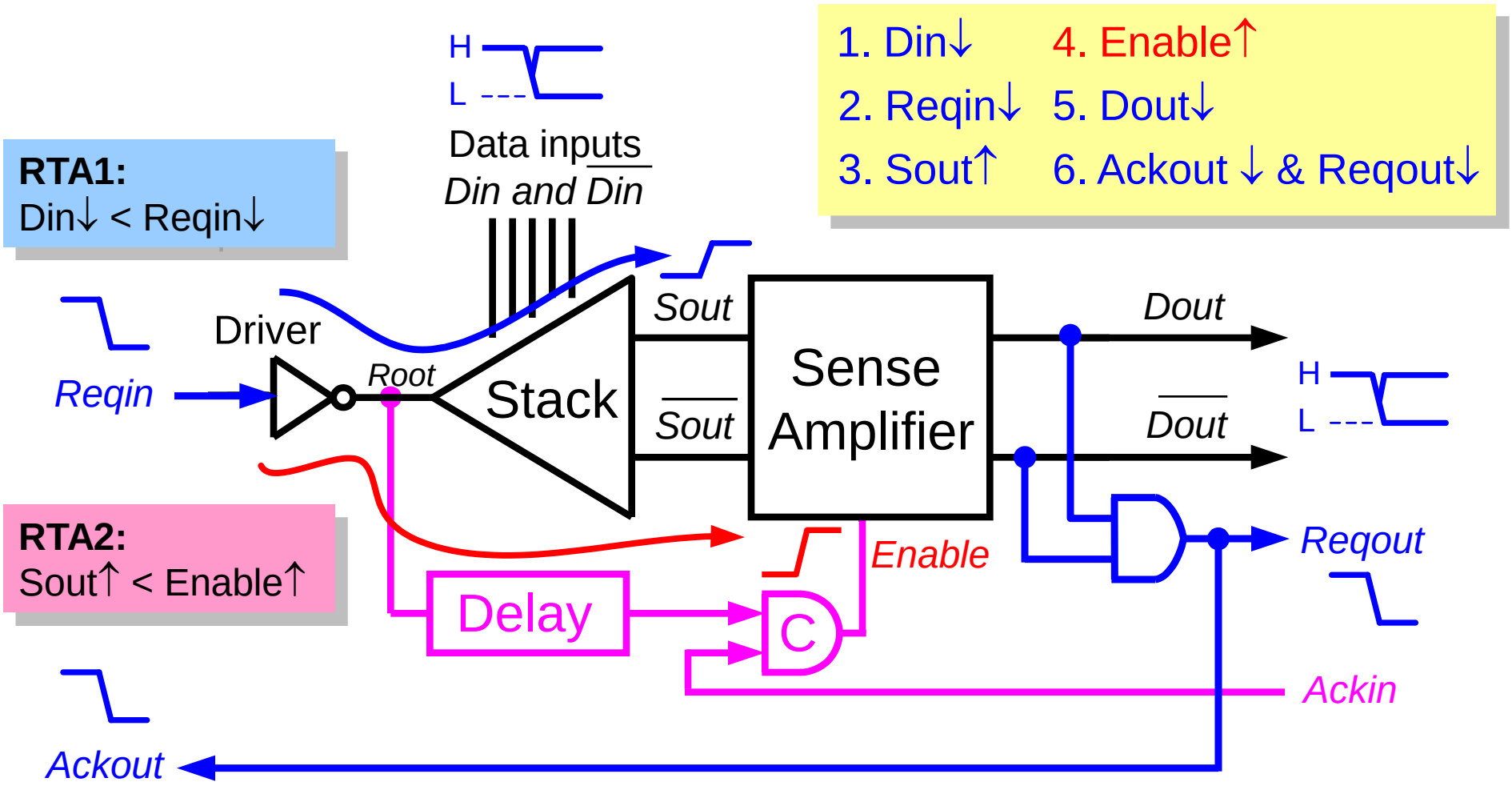
Solution: Use Self-timed control!

- Exploit local timing information
- Low implementation cost in SAPTL

Outline

- Motivation
- Sense Amplifier-based Pass Transistor Logic (SAPTL)
- Self-timed SAPTL
  - Bundled-data self-timed SAPTL design
  - Dual-rail self-timed SAPTL design
- Simulation results
  - Area
  - Energy-delay
  - Leakage
- Conclusion

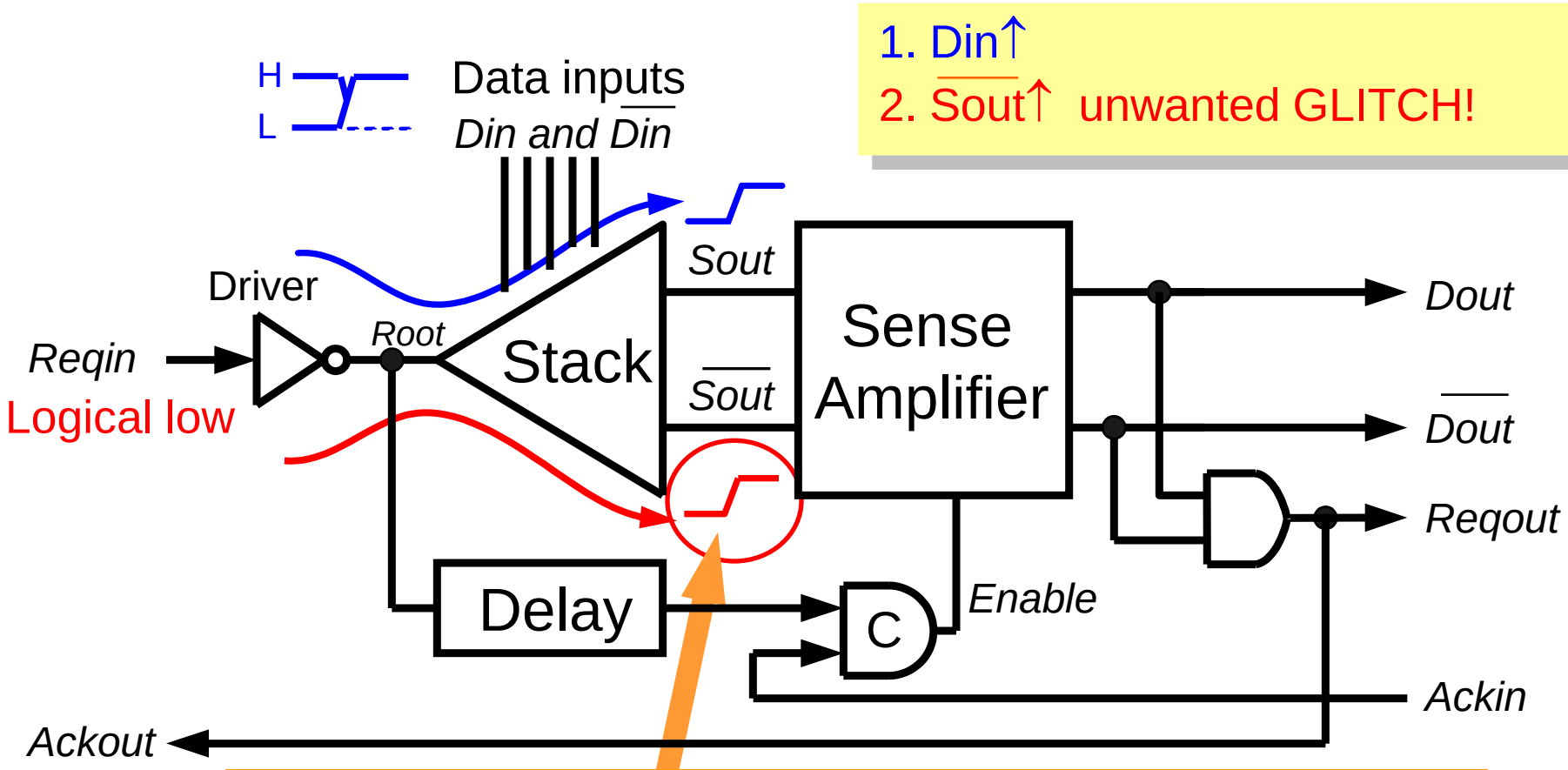
Data Evaluation
in Bundled-Data Self-Timed SAPTL



* RTA : Relative Timing Assumption

Data Reset
in Bundled-Data Self-Timed SAPTL

1. Din↑
2. $\overline{\text{Sout}}$↑ unwanted GLITCH!



Issue: Opposite stack output charges up if *Reqin* is still low
- Degrades the reliability of the self-timed operation
- Increases the energy dissipation

Solution to Avoid Data Reset Glitch

Use different handshake protocols

- Use Din↓ < Reqin↓ for data evaluation
- Use Reqin↑ < Din↑ for data reset

Implementation-1: Late reset

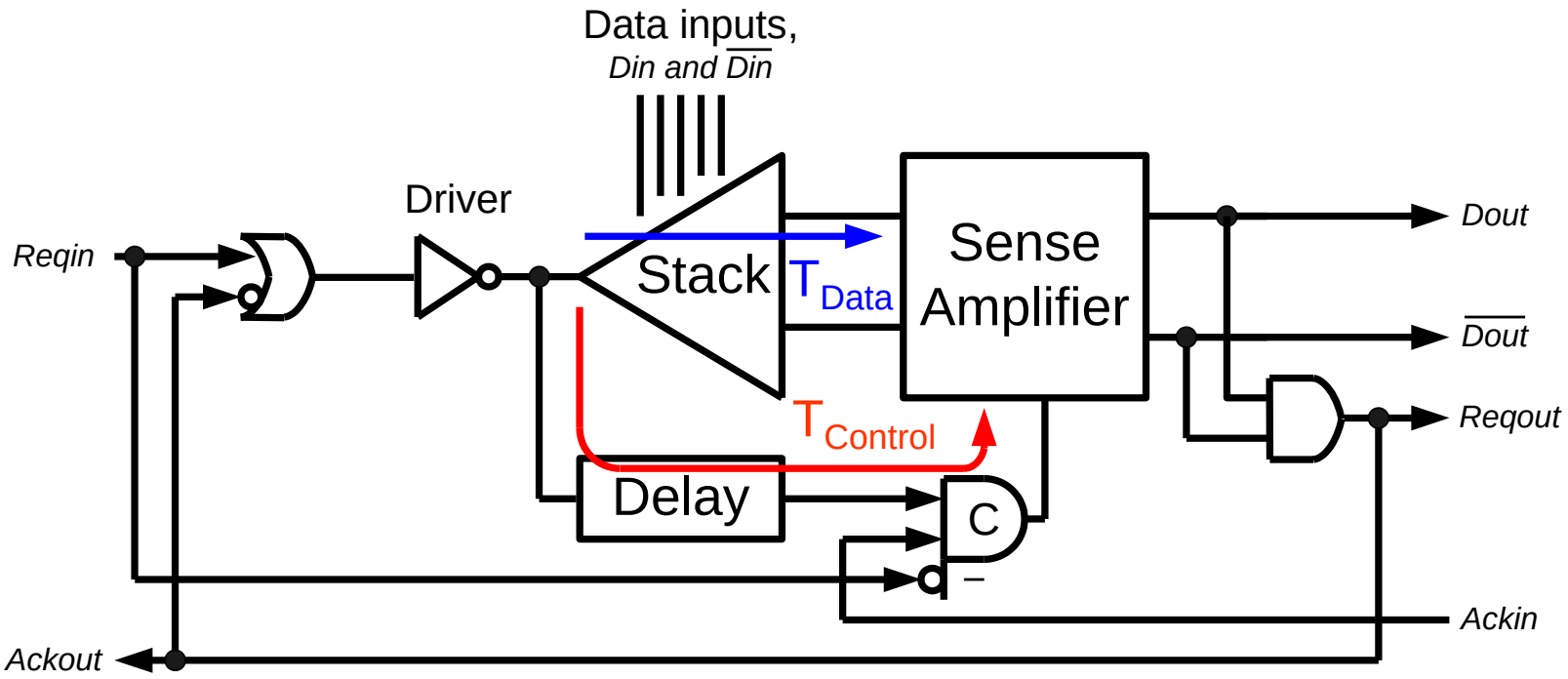
- **Increase data delay:** delay data input signals
- Easy to implement
- But...retards the data reset operation

Implementation-2: Early reset

- **Decrease control delay:** trigger driver before data inputs reset
- Maintains the original data reset latency and throughput
- But…requires an additional relative timing assumption
  ...which is easy to meet

Our choice!

Glitch-Free
Bundled-Data Self-Timed SAPTL

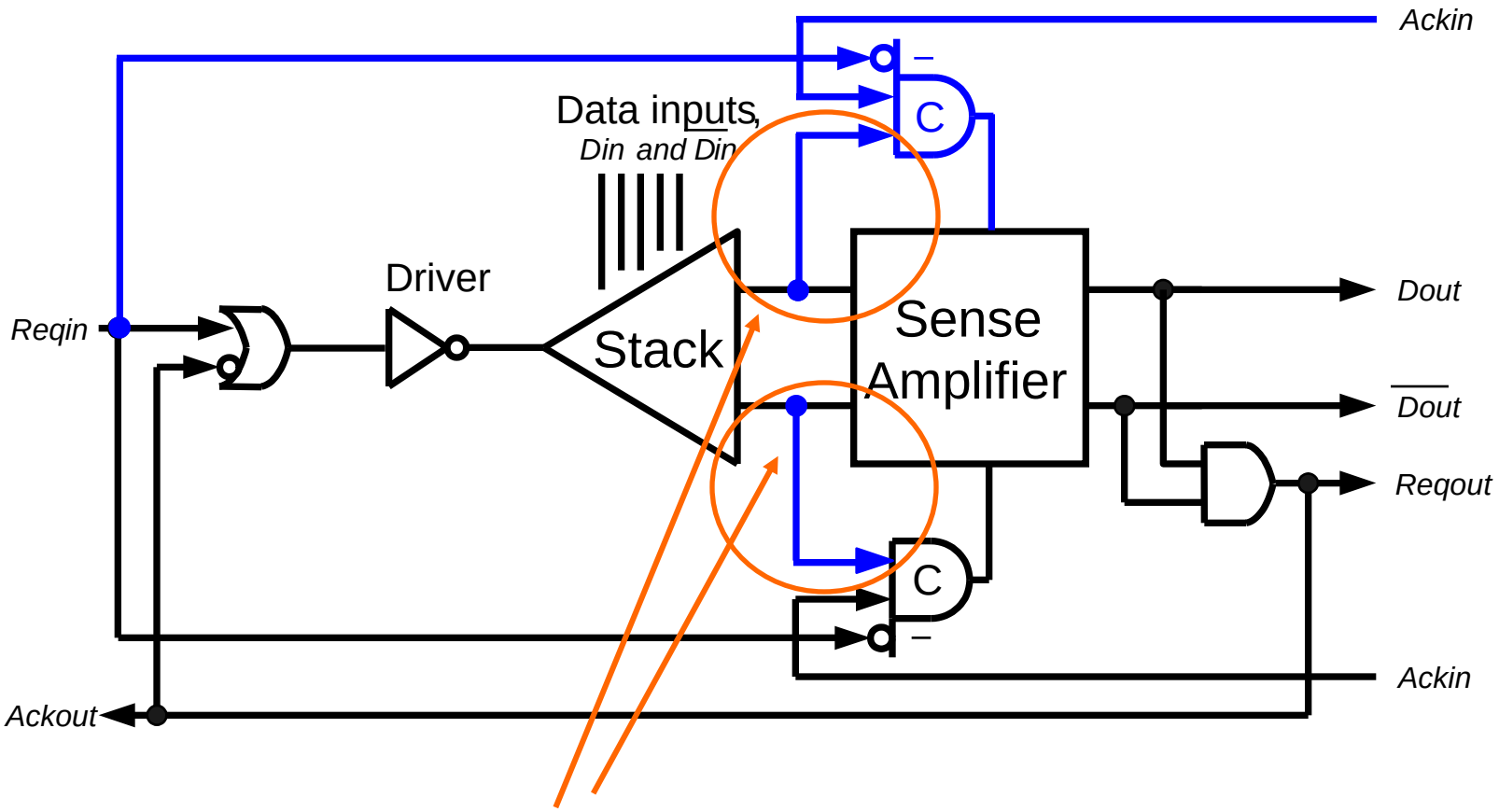


- Replace the global clock signal with local handshake logic
- Delay line controls the local timing
- Still have a delay-matching issue ($T_{Control} > T_{Data}$)

Outline

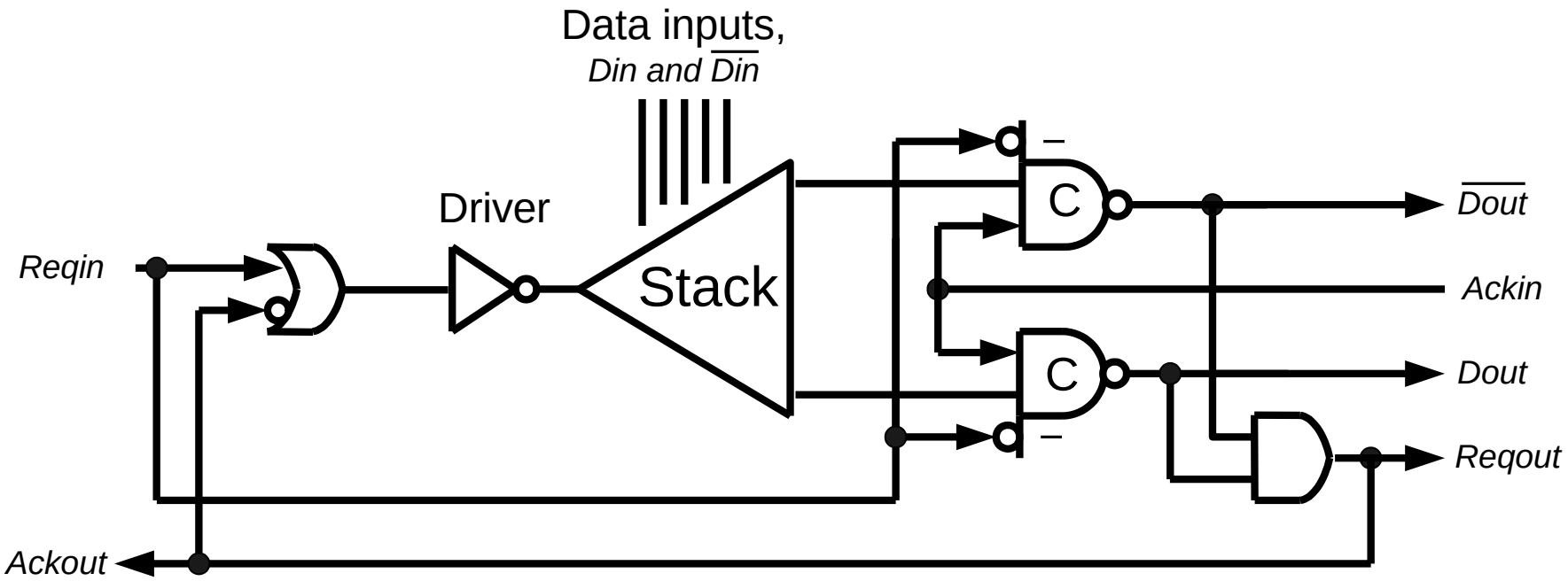
- Motivation
- Sense Amplifier-based Pass Transistor Logic (SAPTL)
- Self-timed SAPTL
  - Bundled-data self-timed SAPTL design
  - Dual-rail self-timed SAPTL design
- Simulation results
  - Area
  - Energy-delay
  - Leakage
- Conclusion

Dual-Rail Self-Timed SAPTL Design



Use stack outputs as trigger signals for sense amplifier

Dual-Rail Self-Timed SAPTL

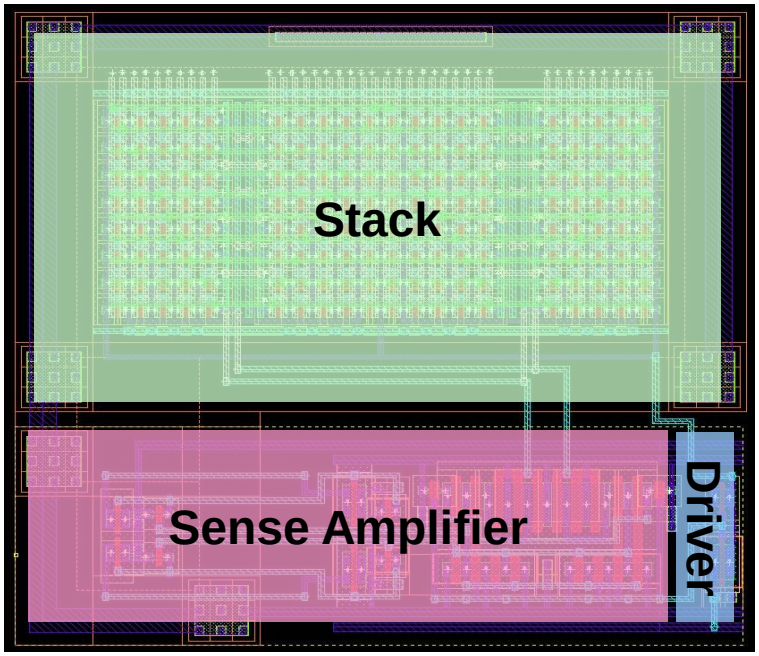


- No delay matching issue anymore
- C-element combines several functions
  - Protocol control
  - Signal recovery
- Hardware complexity similar to the original synchronous SAPTL design

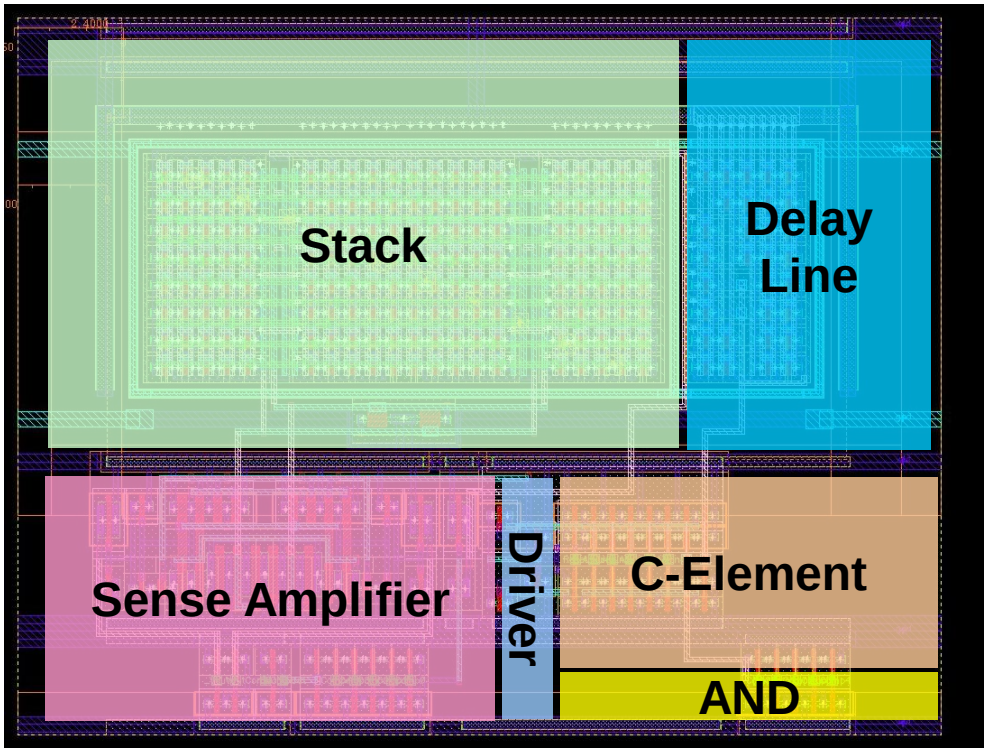
Outline

- Motivation
- Sense Amplifier-based Pass Transistor Logic (SAPTL)
- Self-timed SAPTL
  - Bundled-data self-timed SAPTL design
  - Dual-rail self-timed SAPTL design
- Simulation results
  - Area
  - Energy-delay
  - Leakage
- Conclusion

SAPTL5 Area Comparison in 90-nm CMOS



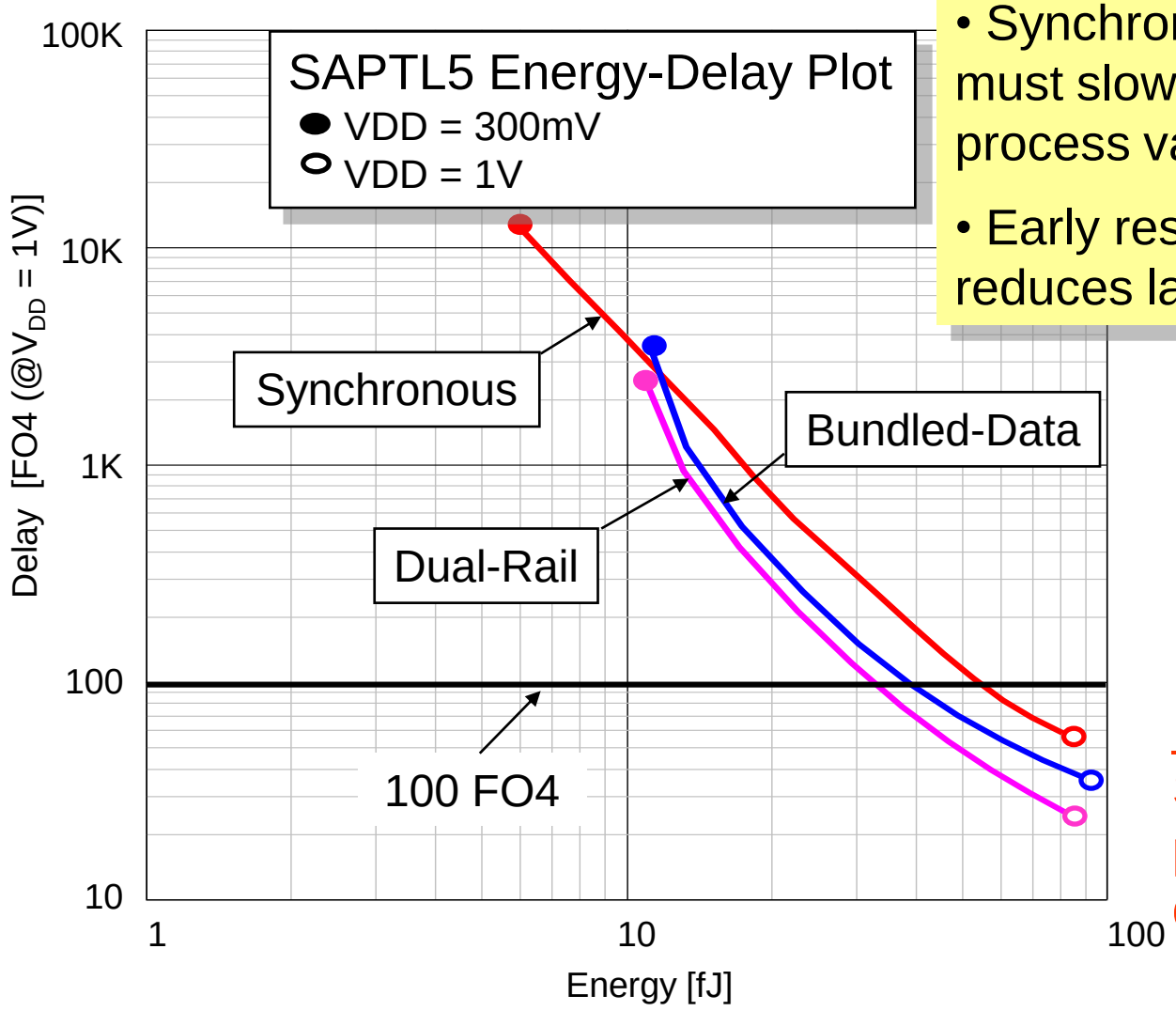
Synchronous SAPTL:
20 μm x 18 μm



Bundled-data SAPTL:
25 μm x 19 μm
(~ 32% larger area)

* **Dual-rail SAPTL:** 20 μm x 21 μm (~ 17% larger than synchronous SAPTL)

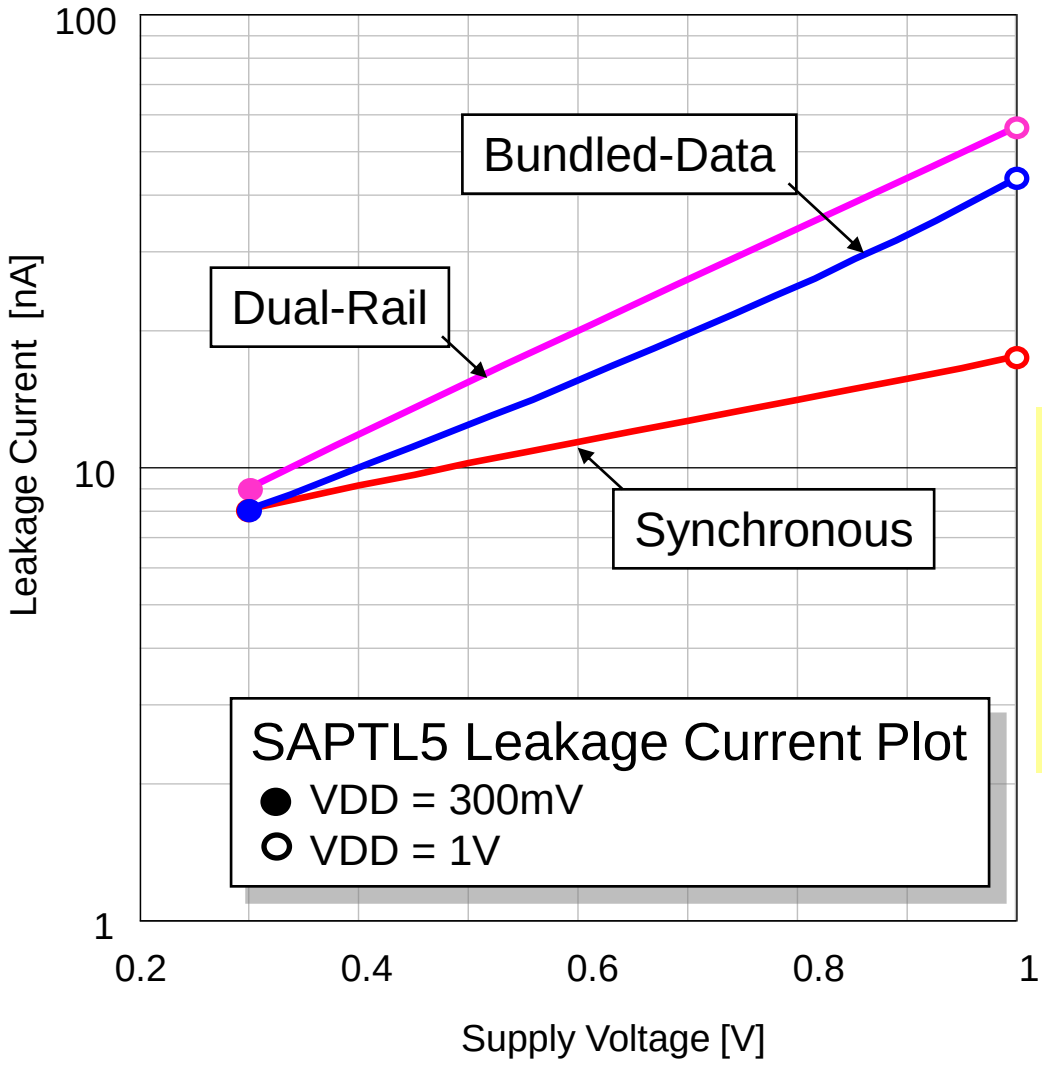
Energy-Delay Simulations



- Synchronous & bundled-data must slow down to account for process variations
- Early reset of self-timed SAPTL reduces latency and energy

Note:
Synchronous SAPTL plot excludes the clock distribution energy

Leakage Simulations



Note:
Synchronous SAPTL plot excludes the clock distribution leakage

- Dual-rail design requires larger C-elements and thus suffers higher leakage
- Leakage difference diminishes with lower supply voltages

Conclusion

- SAPTL is a promising candidate
  for low energy computation at low supply voltage

- Self-timed operation
  improves reliability and energy-delay performance
  without increasing hardware complexity

- Dual-rail self-timed SAPTL
  achieves better energy and speed performance
  in technologies with process variations

Acknowledgement

- Marly Roncken and Ivan Sutherland
- Berkeley Wireless Research Center
- NSF Infrastructure Grant
- STMicroelectronics
- Gigascale Systems Research Center

Thank you!