TRUE CIRCUITS INC™ | *timing is everything*

Overcoming Timing Closure Issues in Wide Interface DDR, HBM and ONFI Subsystems

Brian Gardner
V.P. of Business Development
True Circuits, Inc.

December 14, 2017

Parallel Memory Interfaces

DDR

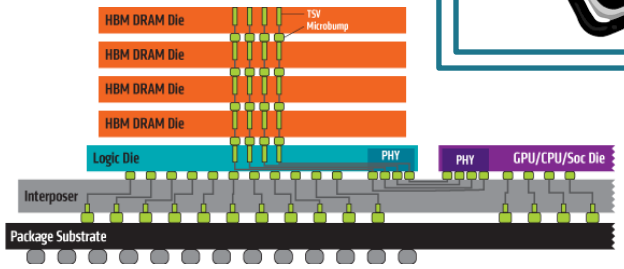
speed: 3200 Mbps+
wide: 72 bits ++

server

LPDDR

speed: 4266 Mbps+
wide: 32/64 bits

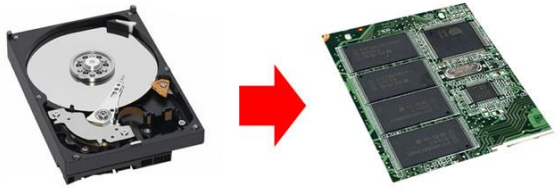
cell
tablet
+++



HBM

speed: 2000 Mbps+
wide: 1024 bits, 8 ch

graphics
(A.I.)



HDD SSD

ONFI

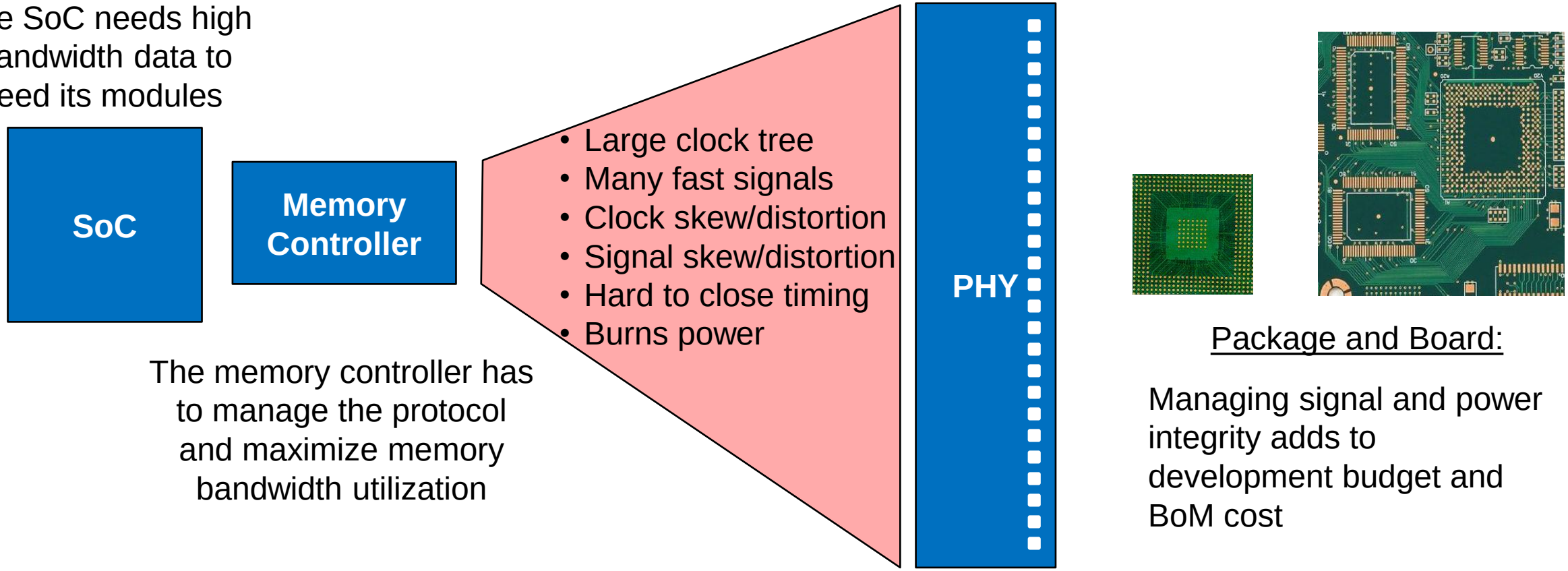
speed: 800 MT/s
wide: 64 bits ++ (SSD)

SSD
++

TRUE CIRCUITS INC.

Parallel Interface PHY Challenges

The SoC needs high bandwidth data to feed its modules



The memory controller has to manage the protocol and maximize memory bandwidth utilization

Package and Board:

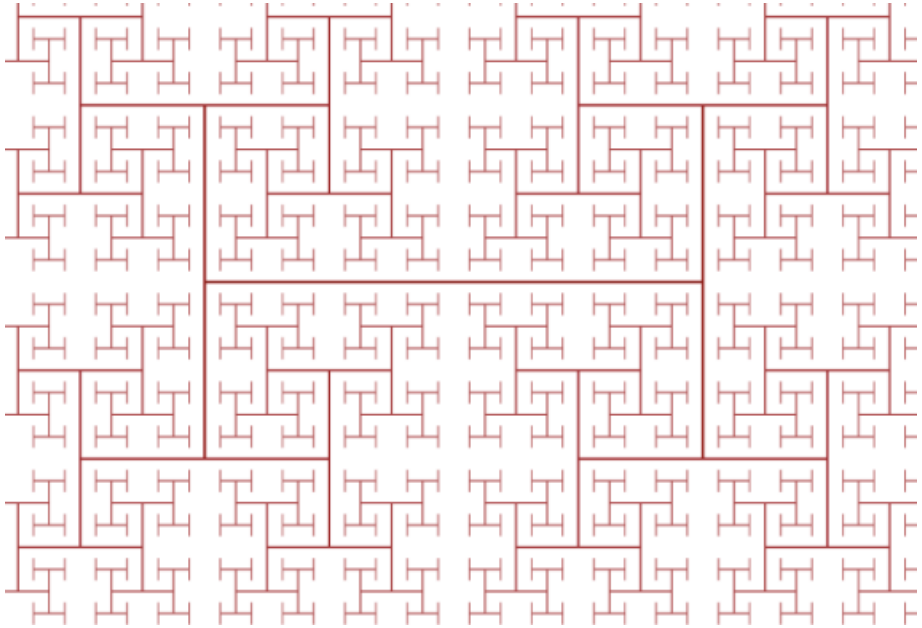
Managing signal and power integrity adds to development budget and BoM cost

Large number of pins, big area and high power.
Use TSMC process options effectively.

TRUE CIRCUITS INC.

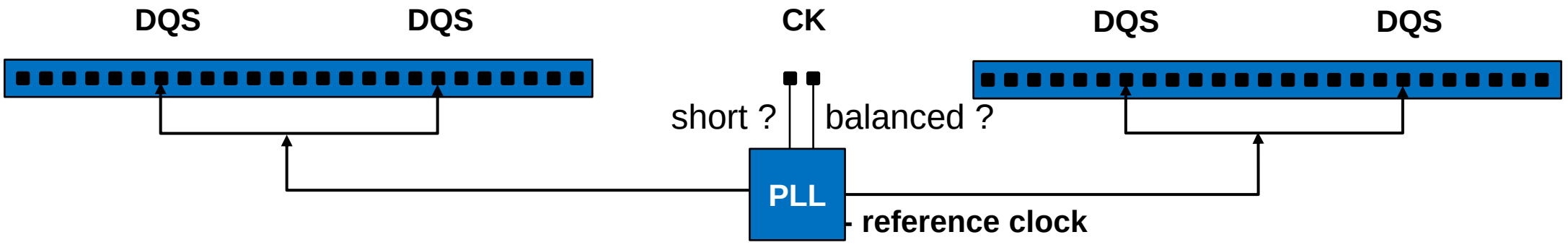
Large Synthesized Clock Tree Issues

- Clock tree may encompass memory controller and PHY
  - Memory controller has high computation complexity
  - PHY has long clock runs
  - TSMC has Vt choices that can help
- Clock uncertainty due to
  - Skew and loading differences
  - On chip variation (OCV)
  - Duty cycle distortion
  - Signal integrity and crosstalk
- To get better clock matching
  - Utilize TSMC's many metal stack options
  - Increase power, spend more time



TRUE CIRCUITS INC.

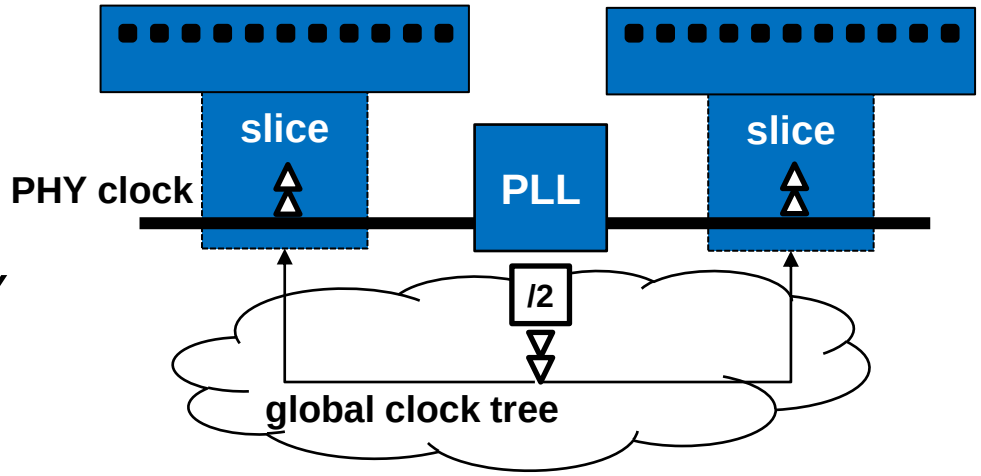
Memory Clock Issues

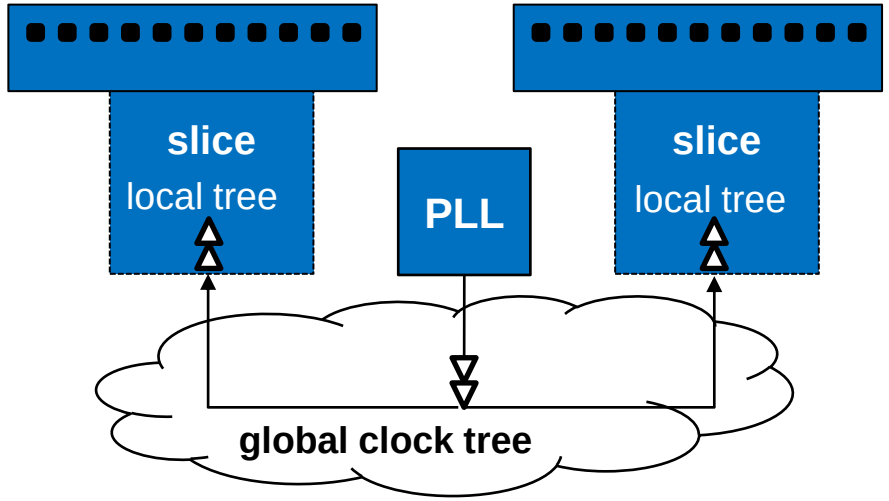


- Clock must be routed long distances, with little distortion (DCD, jitter)
- CK has a tight jitter and duty-cycle distortion spec (at memory pin)
- DQS and CK must be aligned at the memory pin
- Tradeoff:
  - Do you keep the clock path to CK short?
  - Do you balance the CK and DQS clock paths?
- The spec for the reference clock may be hard to meet (speed, jitter)

PHY Timing Tradeoffs

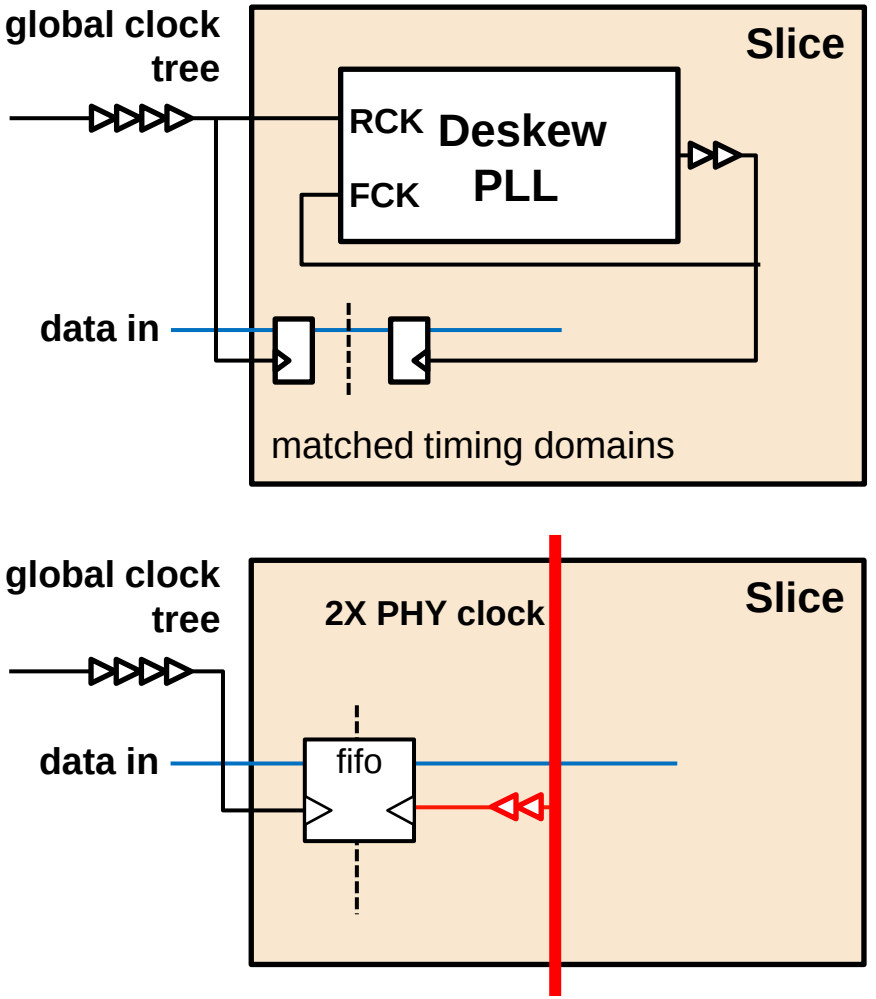
- PLL output sent straight to slices
  - Shielded PHY clock routed to slices
  - Slower, synthesized global clock tree for memory controller and signals to/from PHY
  - Timing domain crossing at slice between global clock and PHY clock
- Slices run on synthesized, global clock
  - Sole clock to slice from global clock tree
  - Timing domain crossing between global clock tree and clock tree local to the slice
- Clock crossing might also include a frequency change (2:1, 4:1)





Solutions for Timing Domain Crossings

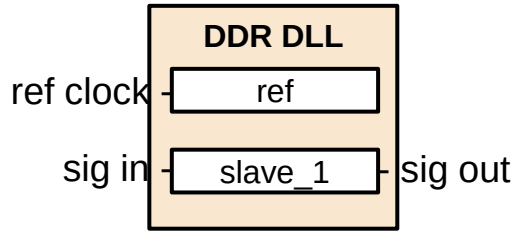
- Deskew PLL
  - Eliminates clock domain crossing at slice
  - Aligns local clock tree to PHY clock
  - Produces a low jitter clock with 50% duty cycle to create write DQS and DQ
- FIFO
  - Separate input and output clocks
  - Very easy to change rate (2:1, 4:1)
- Either way, assure jitter is common to DQ and DQS (not in the DQ-DQS budget)
  - DQs and DQS driven from common point



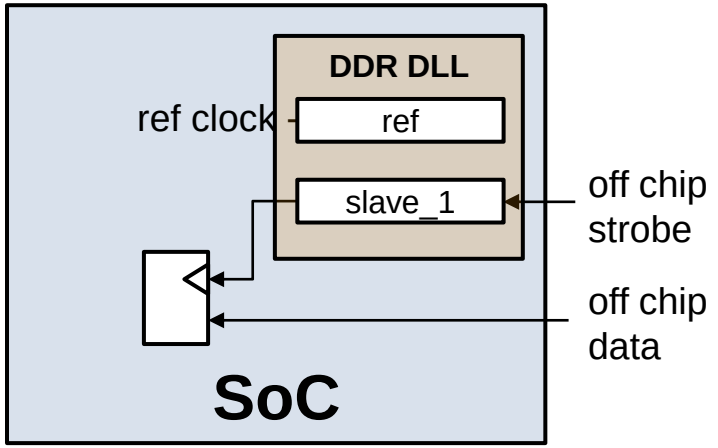
Solutions for Creating Delays Using the DLL

- PHYs need PVT-compensated, fine resolution delays
- Master/slave DLL
  - Creates a programmable delay, constant over PVT
  - Uses a reference clock
  - Uses analog circuitry for smooth control, fine resolution
  - Configurable number of slaves
- Examples of using DLL slaves
  - Delay DQ-aligned read strobe into center of data eye
  - Delay write DQS to center of data eye
  - Delay DQS to align with CK at memory device
  - Many other tuning/training possibilities

The DLL

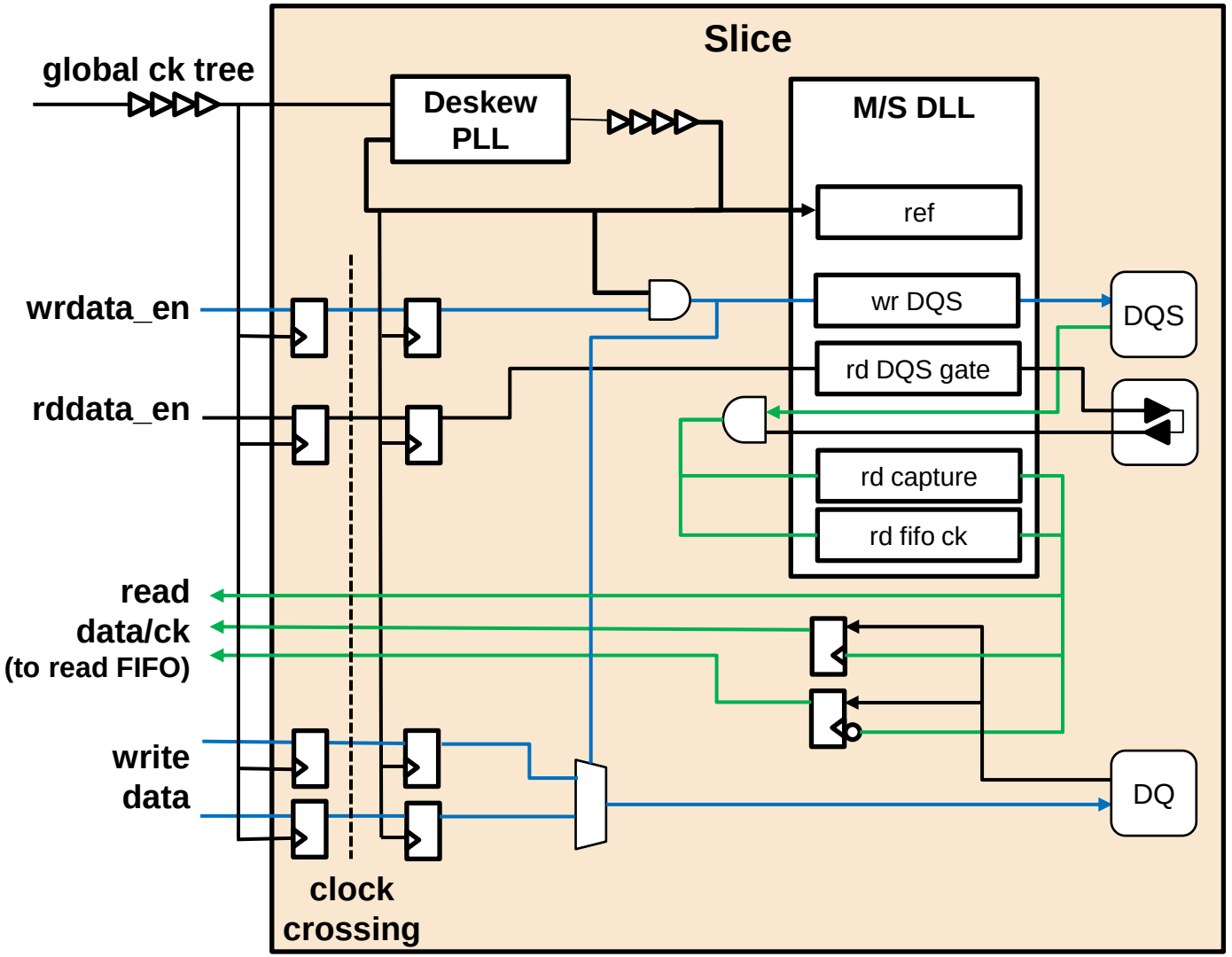


Use of the DLL



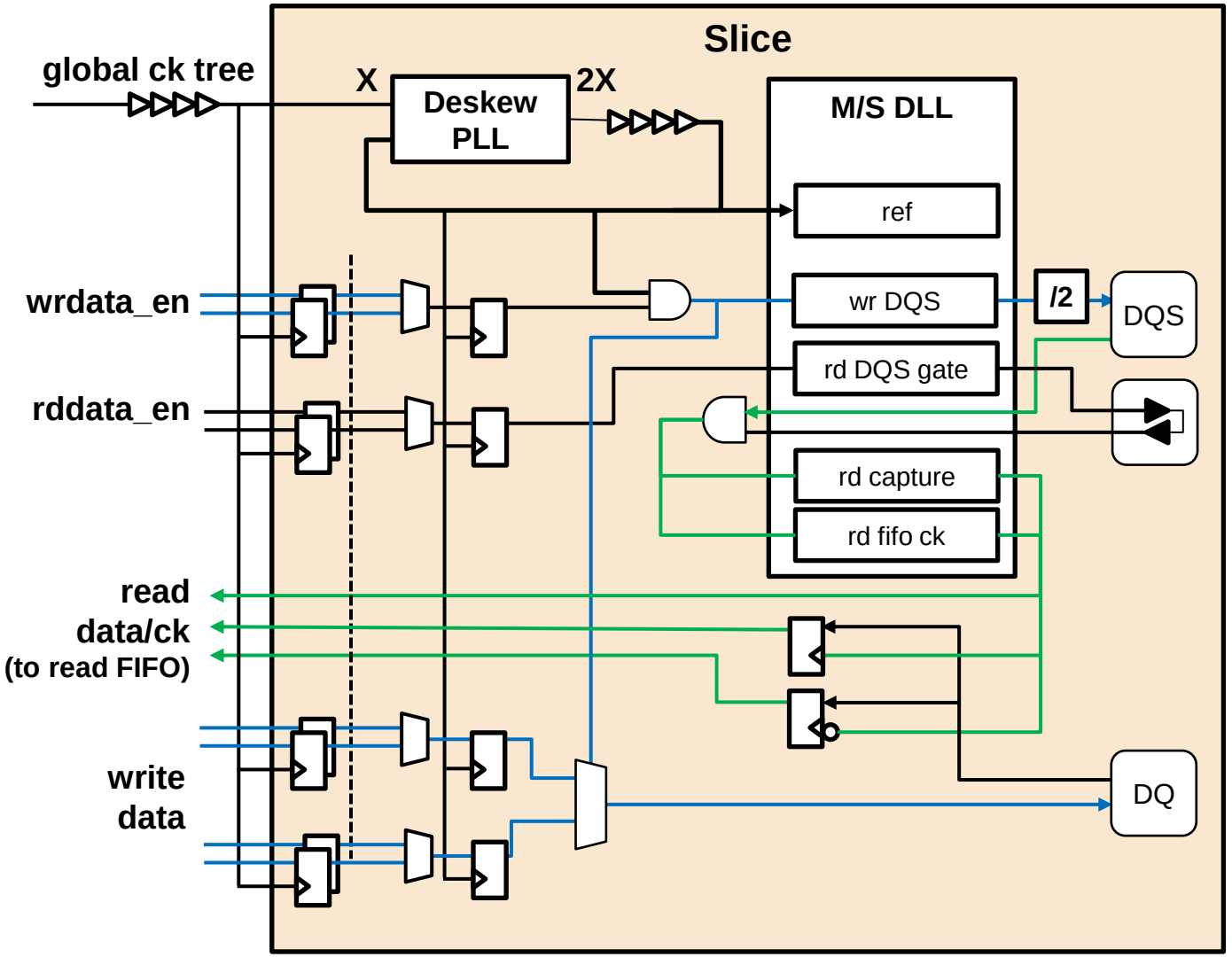
The Total Picture

- Deskew PLL
  - Aligns global and local clock domains
  - Removes high frequency jitter and improves duty-cycle distortion
- Master-Slave DLL
  - Centers write DQS strobe in write data eye
  - Adjusts rddata_en to gate read DQS
  - Centers read DQS in data eye
  - Creates a source-synchronous strobe for read data to the FIFO
- Other additional slaves
  - Deskew per pin or group
  - Train per chip select



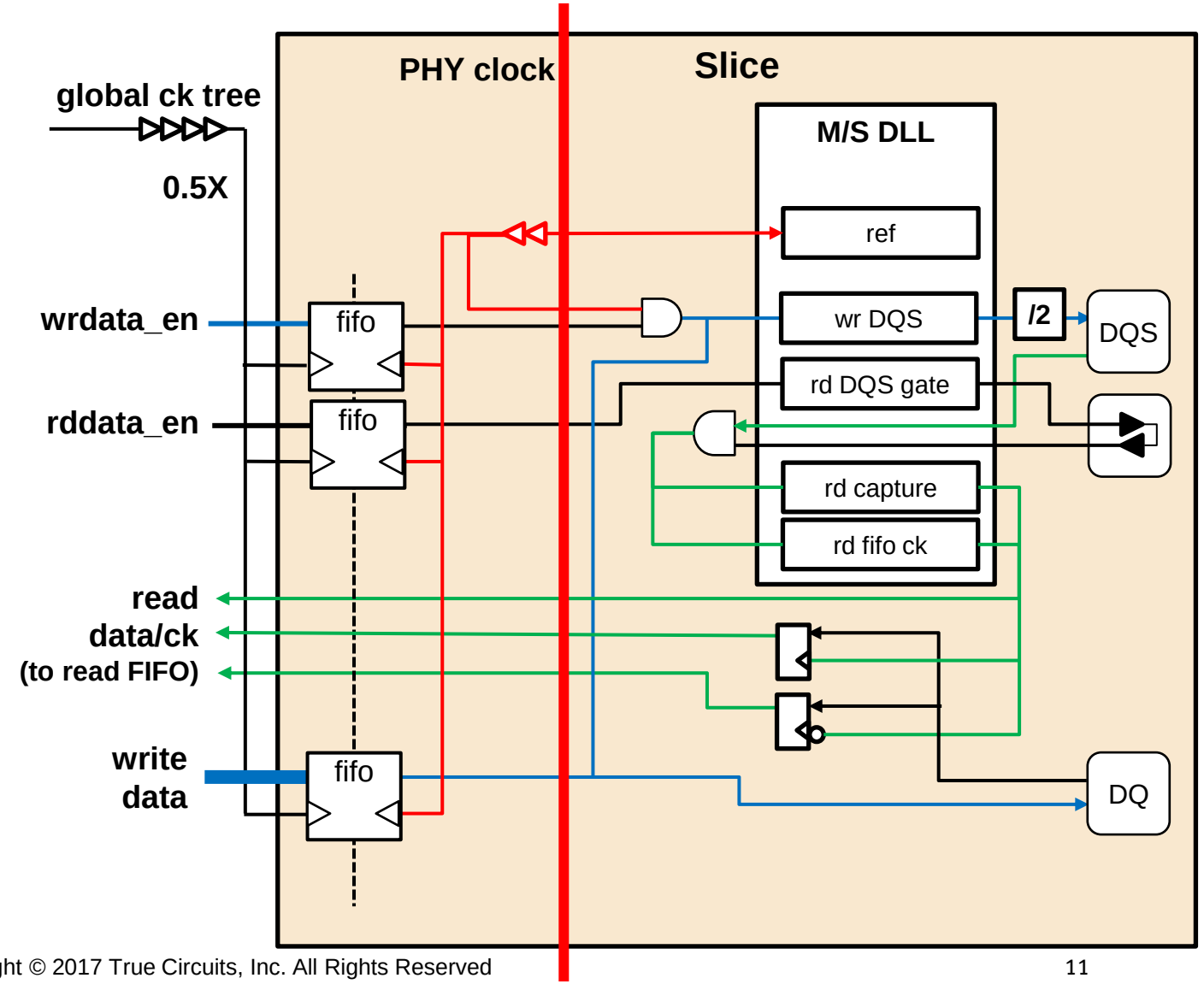
Design Choices: Using a Deskew PLL to Multiply

- Deskew PLL
  - Multiplies clock by 2
- Can run global clock at 0.5X
  - With PHY clock at 1X
  - Slows write data and read/write enable, allowing easier timing closure
- Or, run slice with 2X clock
  - While global clock at 1X
  - Divide write DQS at pin to reduce duty-cycle distortion
- Or, multiply by 4 and do both



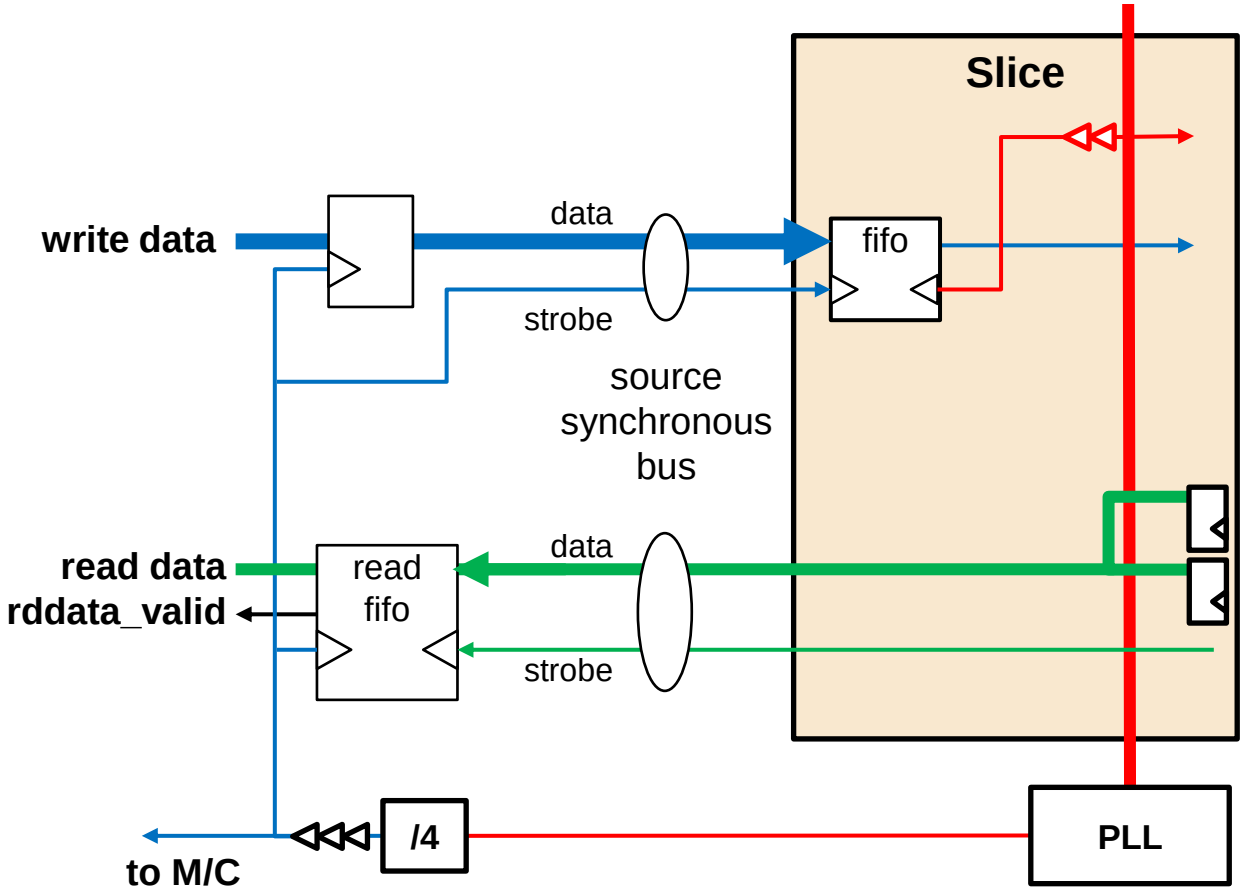
Design Choices: Using FIFOs

- PHY clock is routed separately
  - Minimizes jitter and DCD to slice
  - Shielding reduces cross-talk
  - Faster clock possible than with a global tree
- FIFO
  - Use to cross between global clock and PHY clock for write data and write/read enables
  - Also used on the read side along with read_valid signal
  - Allows clock ratios (2:1, 4:1)
- Must insure FIFOs stay in sync with one another and CA slice



Design Choices: Using Source Synchronous Signals

- Source synchronous bus
  - Data and strobe routed together and matched
  - Replaces large, synthesized global clock and data pipeline
  - Easy timing closure; simple PnR
- FIFOs
  - Used to cross timing domains
- Other signals “multi-cycle path”
  - Tools use loose setup/hold constraints



More Design Choices: Looking at the TCI DDR PHY

- Deskew every DQ pin using DLL and DLL-like circuits
- Use a regulated PHY clock tree to reduce jitter
- Add a built-in, low jitter PLL for the PHY clock that can use any reference clock, and optionally adds spread spectrum
- Sample read data at the switch point to allow continuous tuning of data eye and internal voltage reference
- Add loopback testing for the full write path out and the read path back
- Add circuitry to measure flight times and switch point jitter

Summary

- Wide, parallel interfaces have many challenges
- There are many solutions and design choices
- TCI has the right PLL, DLL and DDR PHY IP and the timing architecture expertise to help you succeed

Thank you!