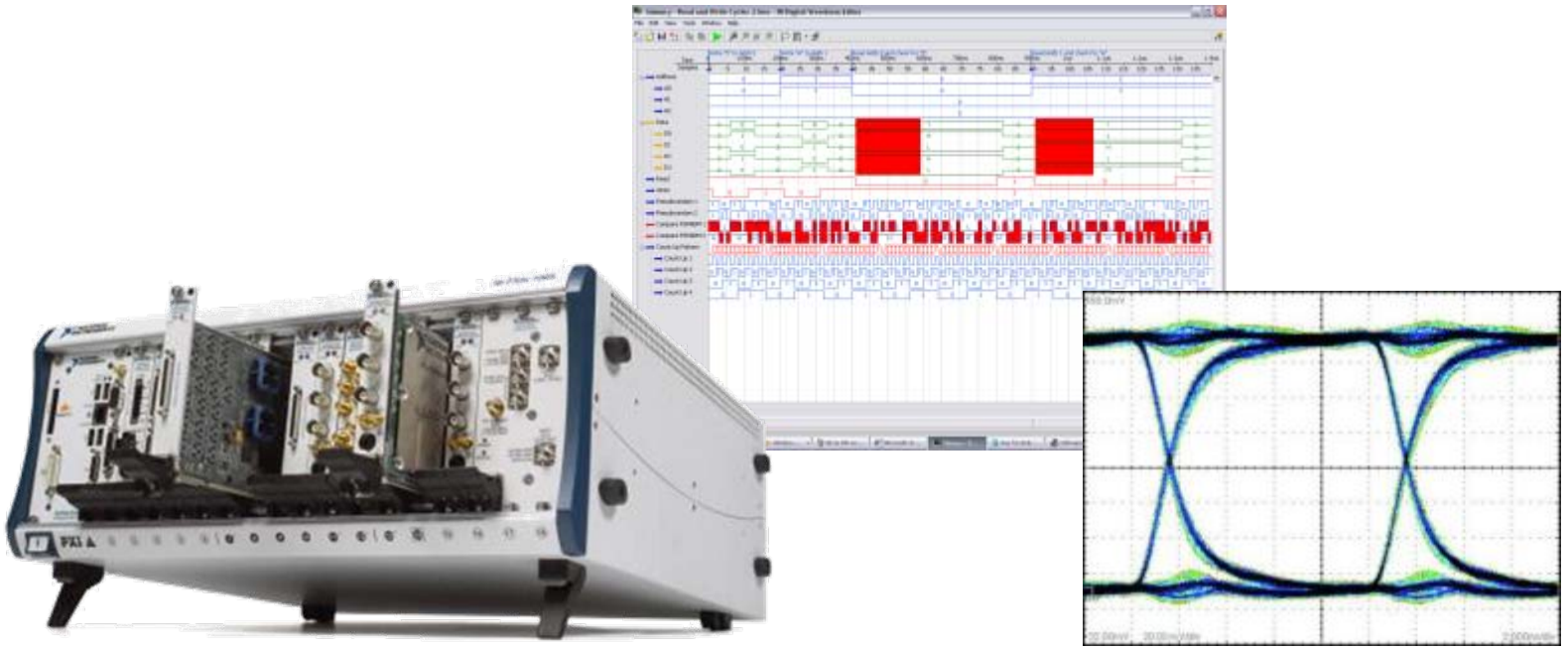
Oslo - Stockholm - Utrecht - Brussels - Copenhagen - Helsinki

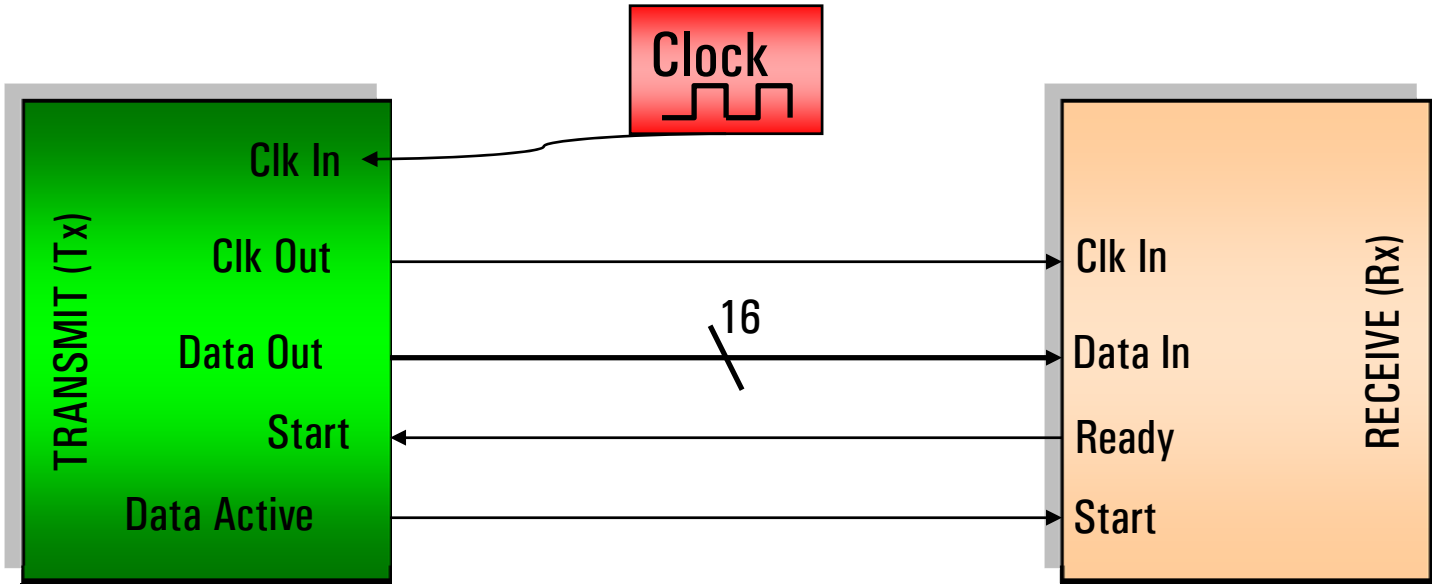
ni.com/nidays

NATIONAL INSTRUMENTS

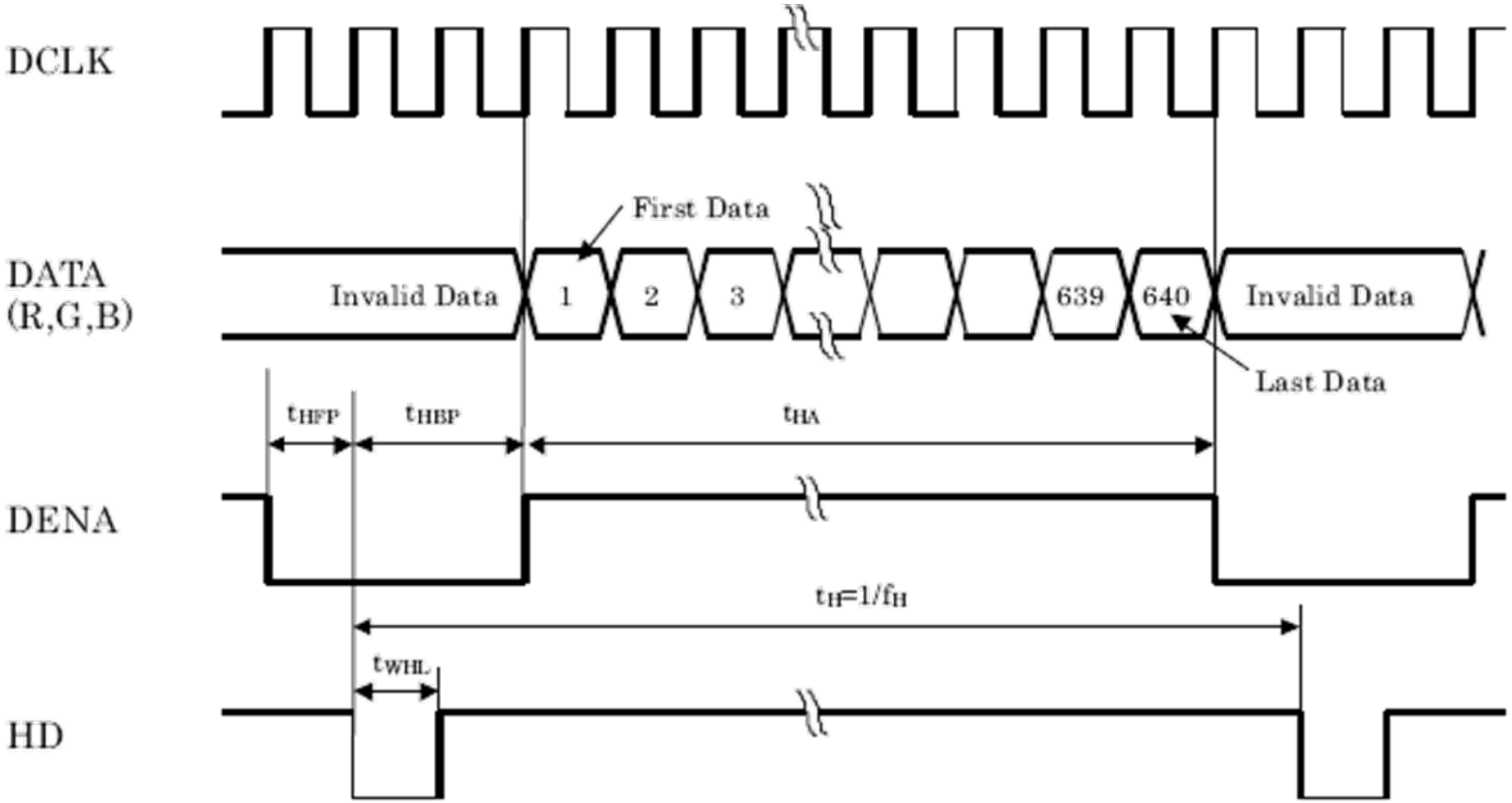
Digital Interfacing and Test Fundamentals



Common Conventions – Block Diagrams

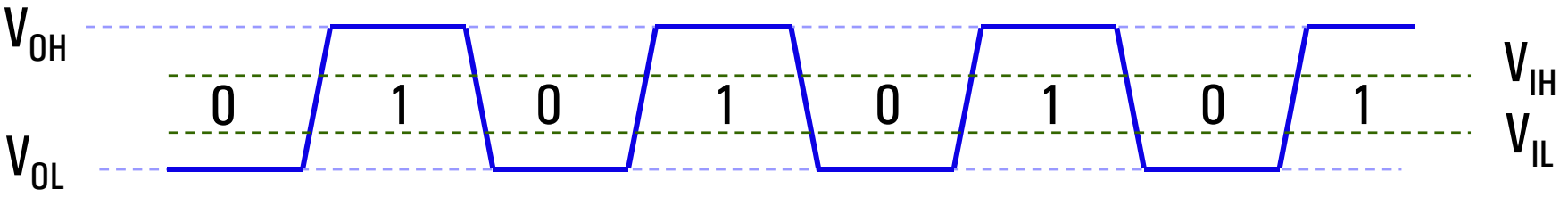


Common Conventions – Timing Diagrams

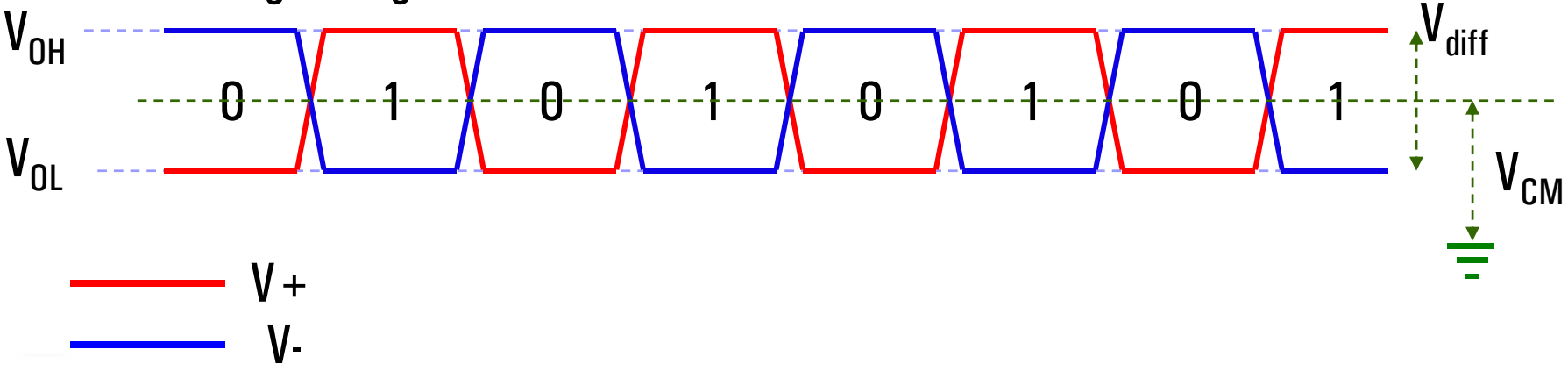


Voltage Definitions

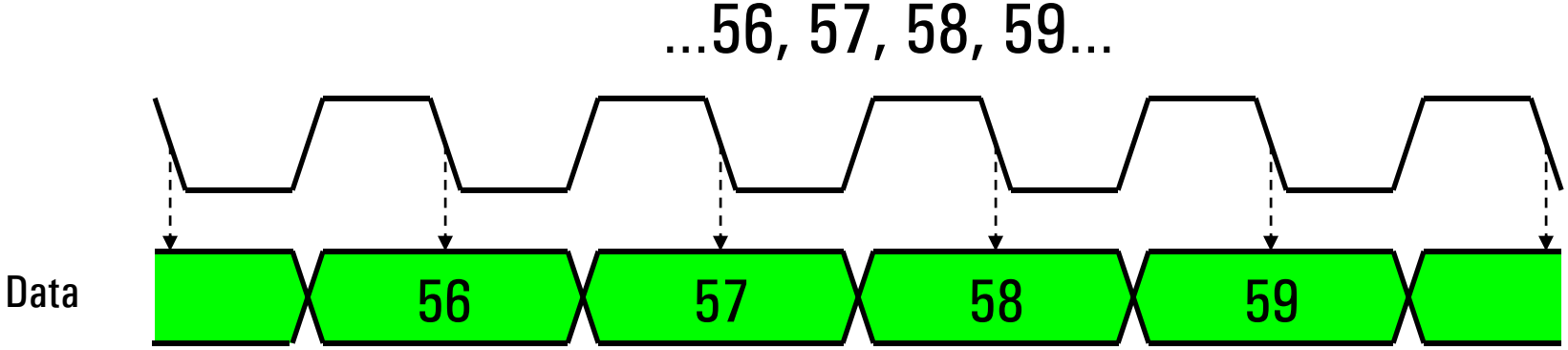
Single-ended digital signals (TTL, CMOS, RS232...)



Differential digital signals (LVDS, ECL, PECL, RS422...)

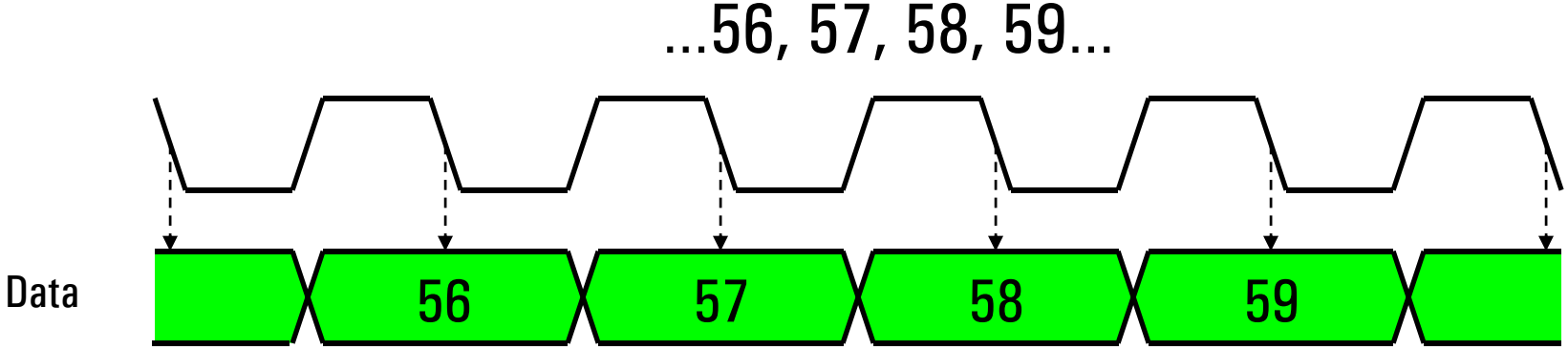


Timing and Digital Data Transfer



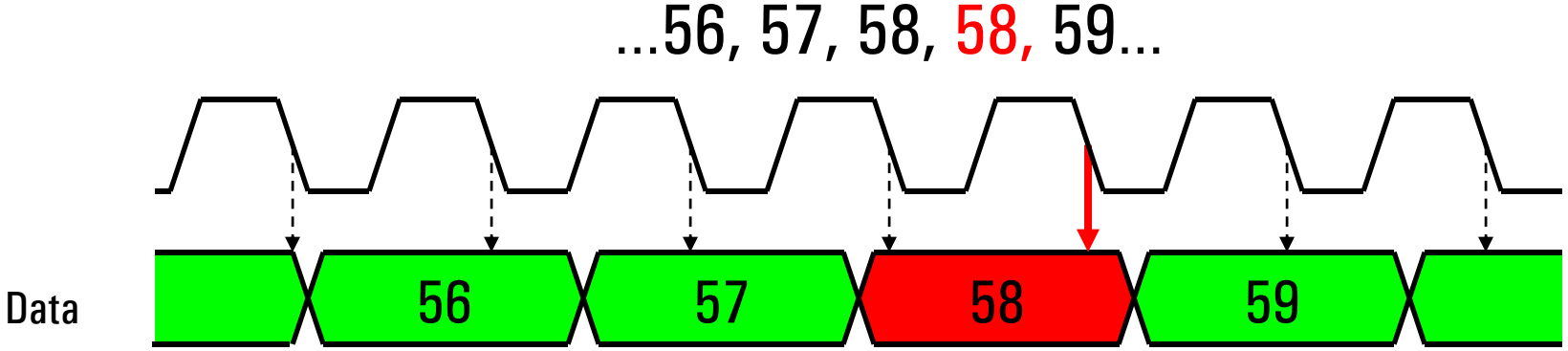
- Understanding timing (i.e., WHEN to sample and WHEN to generate) is crucial

Timing and Digital Data Transfer



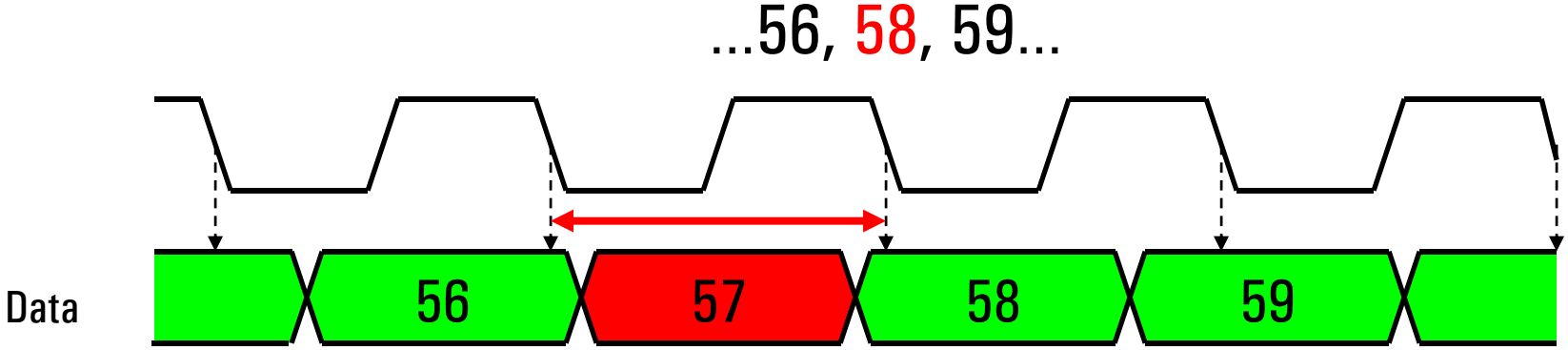
- Violate timing and you could ...

Timing and Digital Data Transfer



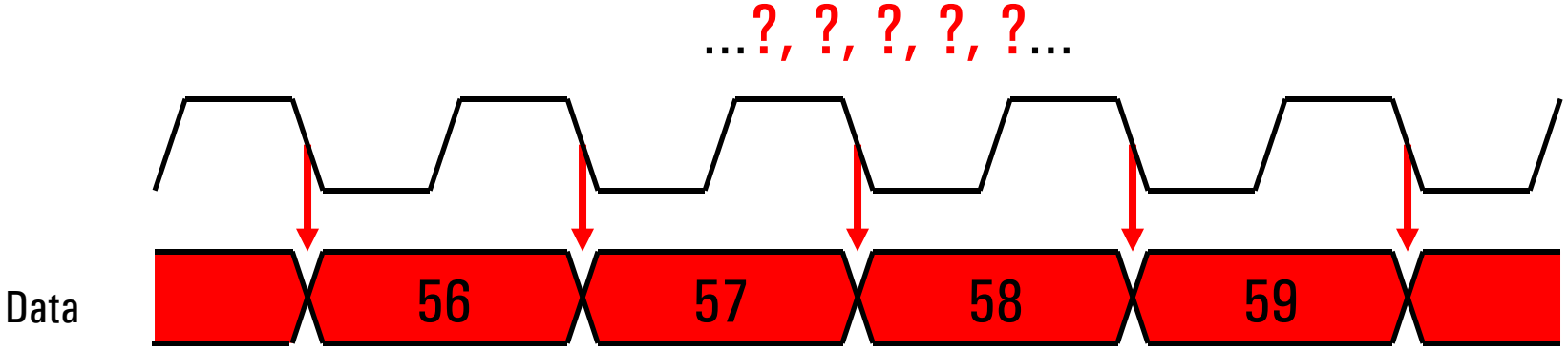
- Violate timing and you could ...
  - Sample the same data twice

Timing and Digital Data Transfer



- Violate timing and you could ...
  - Sample the same data twice
  - Miss a sample

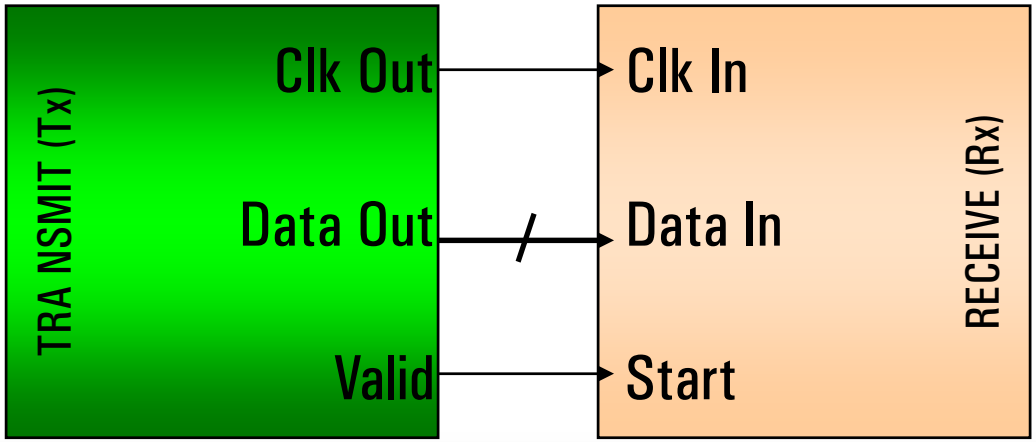
Timing and Digital Data Transfer



- Violate timing and you could ...
  - Sample the same data twice
  - Miss a sample
  - Sample during the transition

Digital Data Transfer Timing Configurations

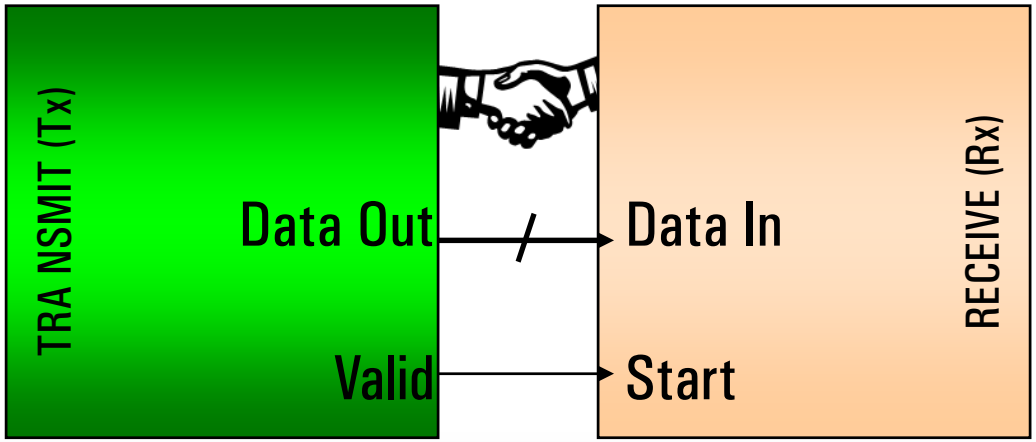
- **Synchronous** Transfer
  - Data sent at constant rate using shared periodic clock



Digital Data Transfer Timing Configurations

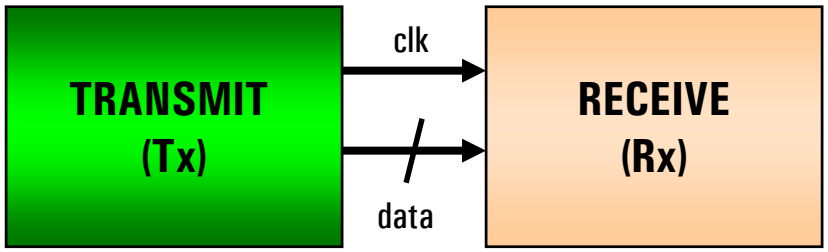
- **Synchronous** Transfer
  - Data sent at constant rate using shared periodic clock
- **Asynchronous** Transfer (i.e., Handshaking)
  - Data sent upon request using handshake signals

    Tx and Rx still have internal clocks, they just don't share them



Synchronous Data Transfer

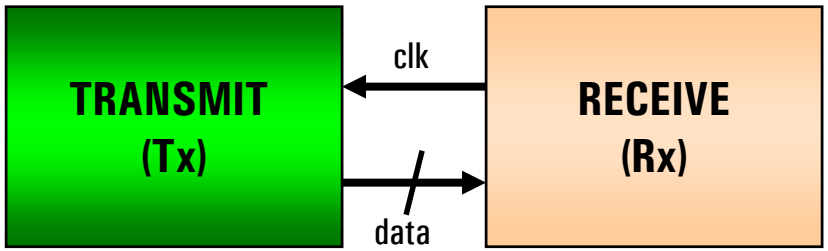
- Common clock synchronizes communication
- Clock is generated by:
  - Tx → Rx (Source-Synchronous)
  - Rx → Tx, or external to both (System-Synchronous)

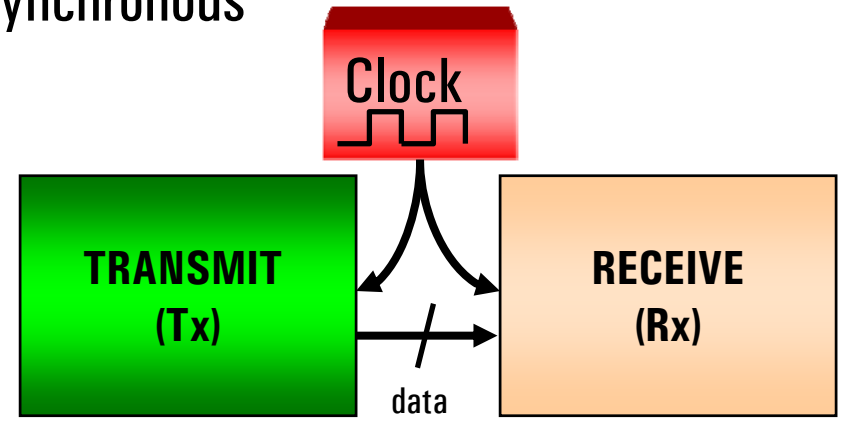


Source
Synchronous

System
Synchronous

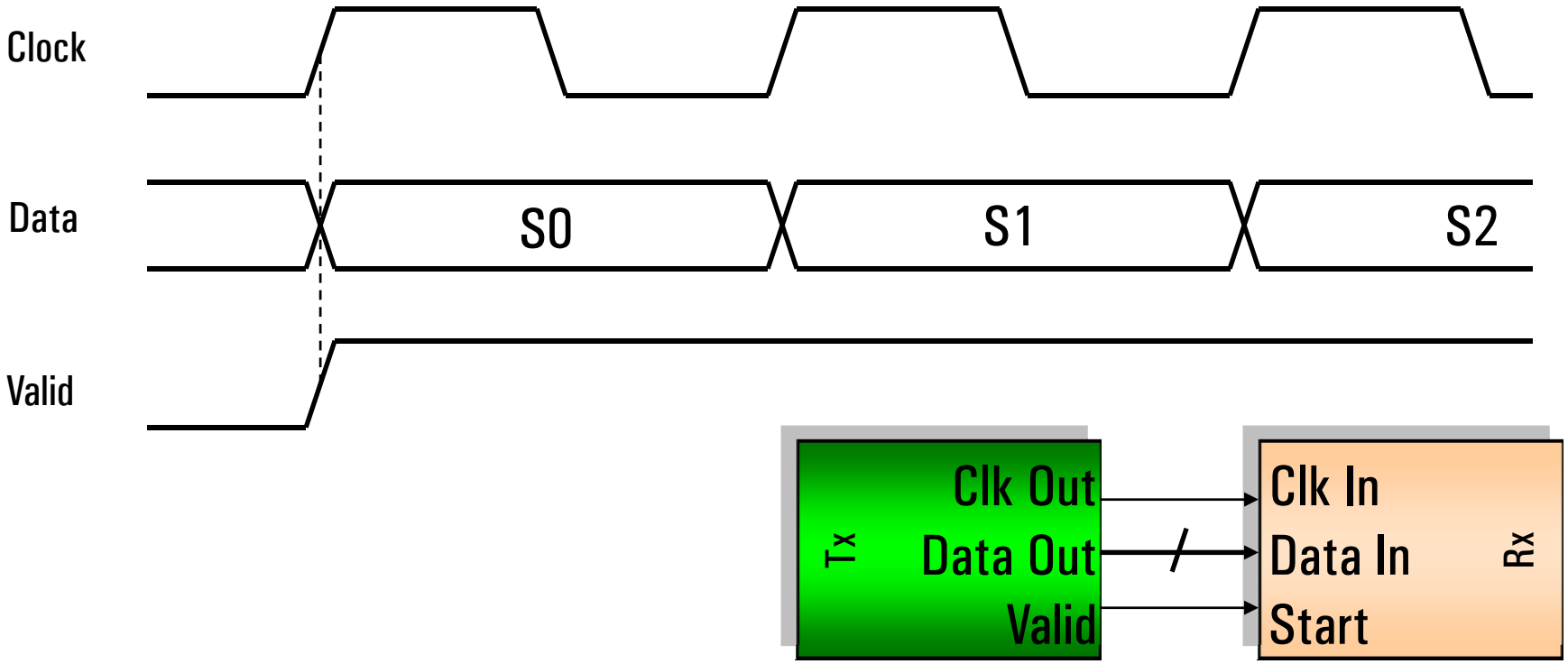
OR



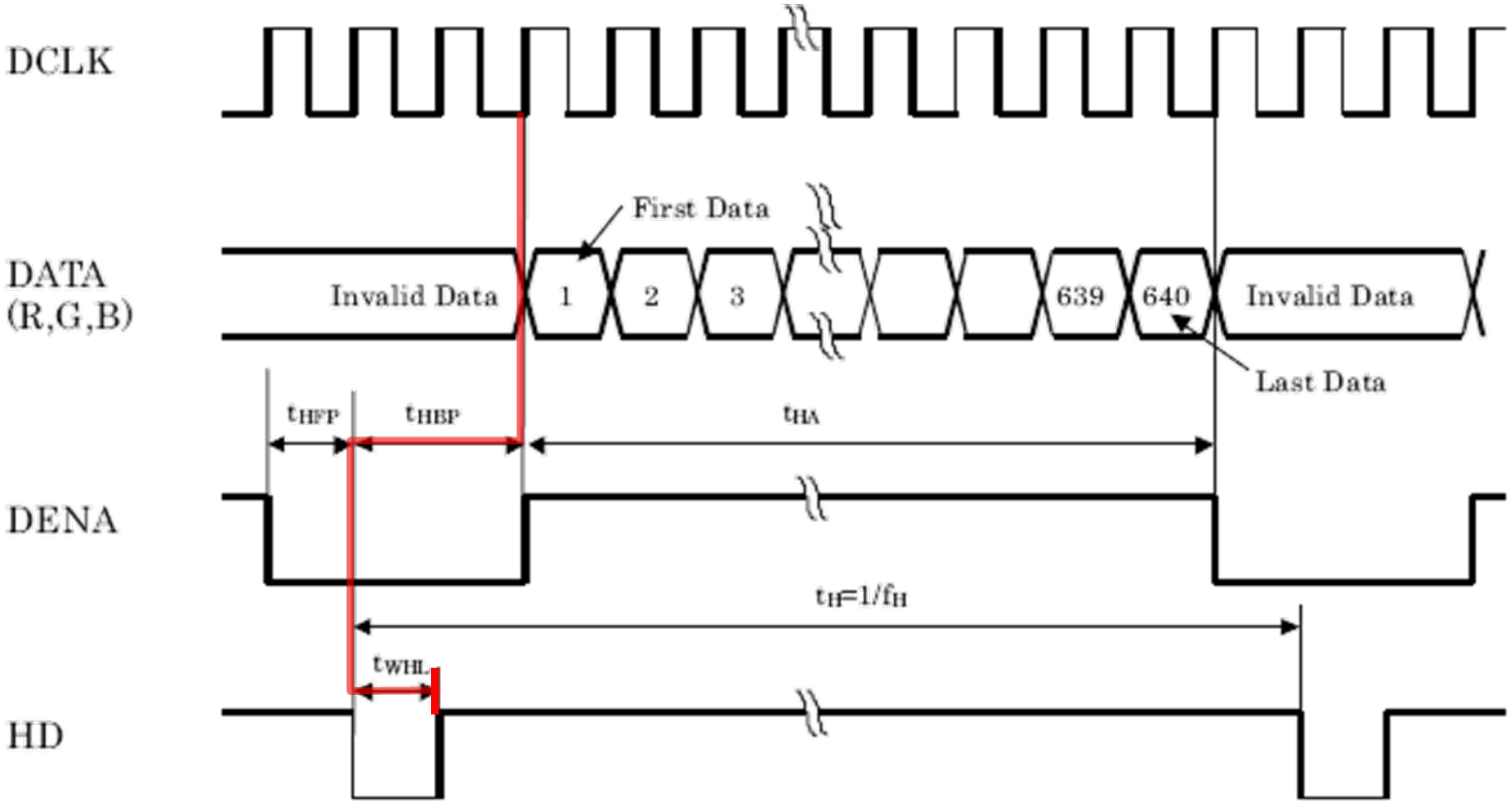


Synchronous Data Transfer

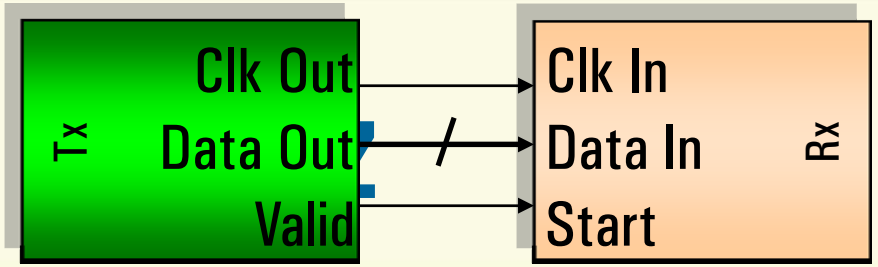
- Clock is periodic
- Data and control signals generated relative to clock edge



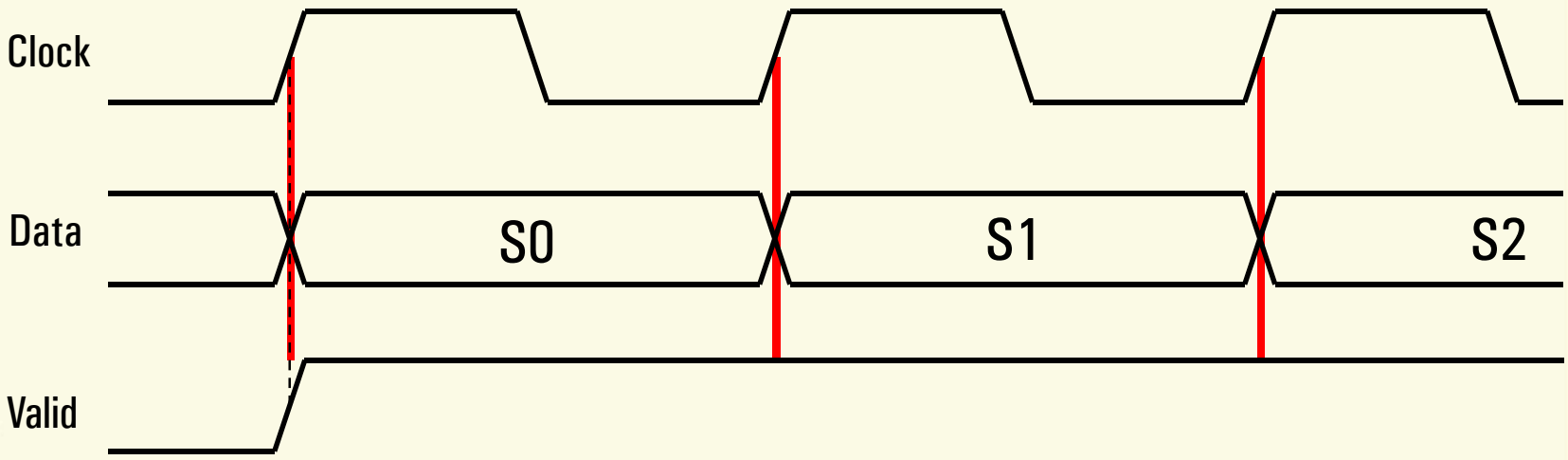
Synchronous Data Transfer: Timing Diagram

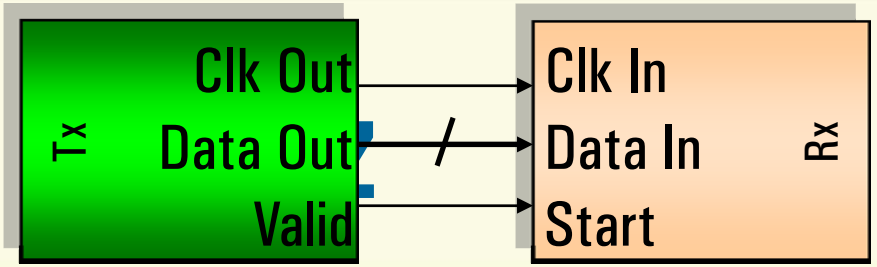


All timing can be traced back to the clock ...



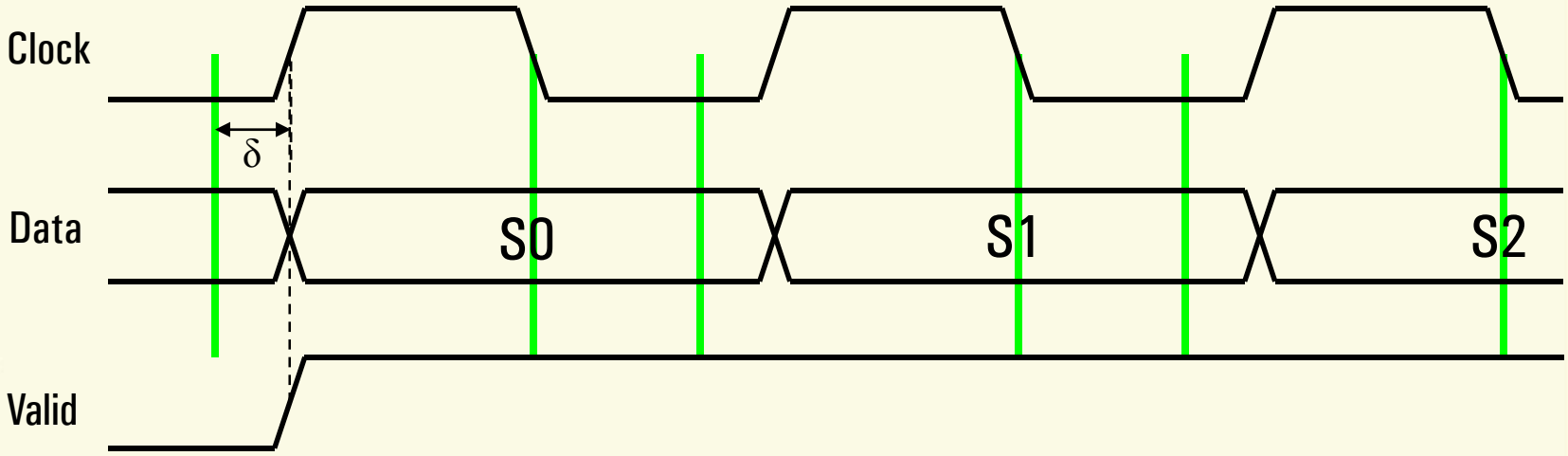
- **Q:** Rx is configured to sample on the rising edge of **Clock**. Will this work?
- **A:** No. Rx will sample while **Data** and **Valid** are unstable.





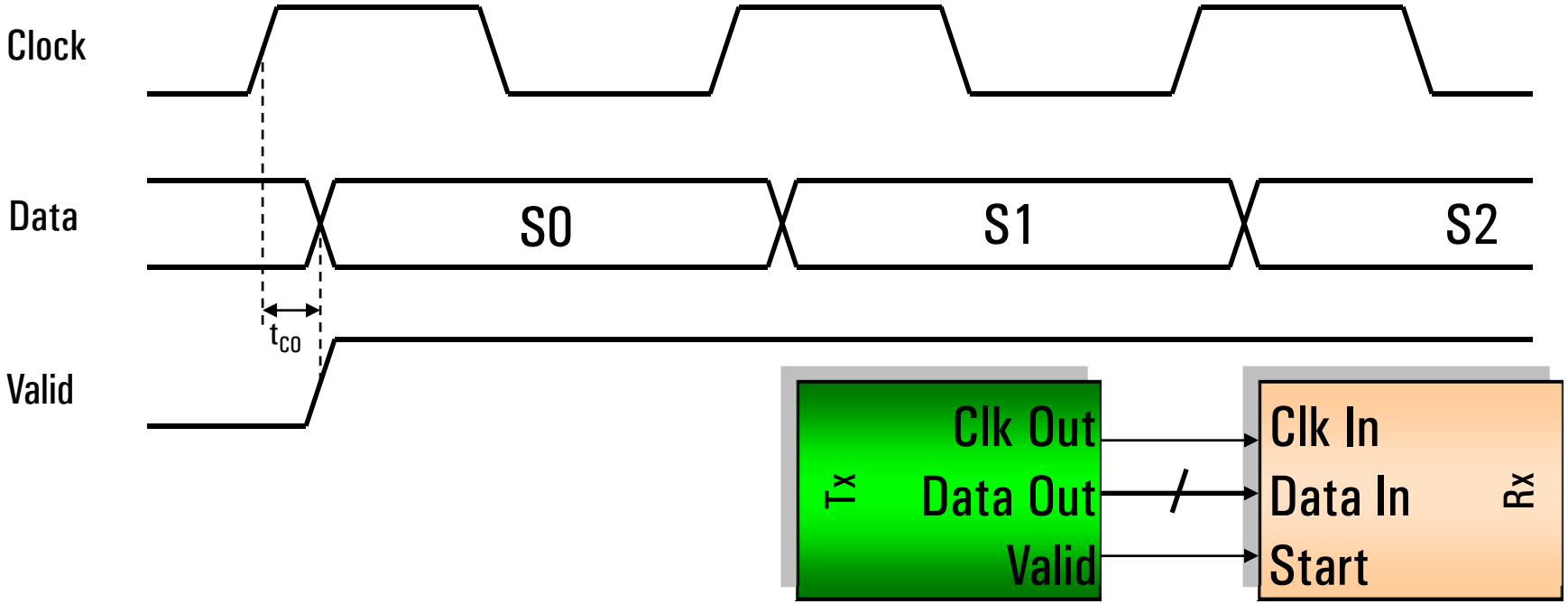
- **Q:** What *Rx* change allows correct sampling of **Data** and **Valid**?
- **A:** Sample on falling edge (or custom delay)

- **Q:** What *Tx* change allows correct sampling of **Data** and **Valid**?
- **A:** Shift **Clock** relative to **Data** and **Valid**
  (or shift **Data** and **Valid** signals relative to **Clock**)



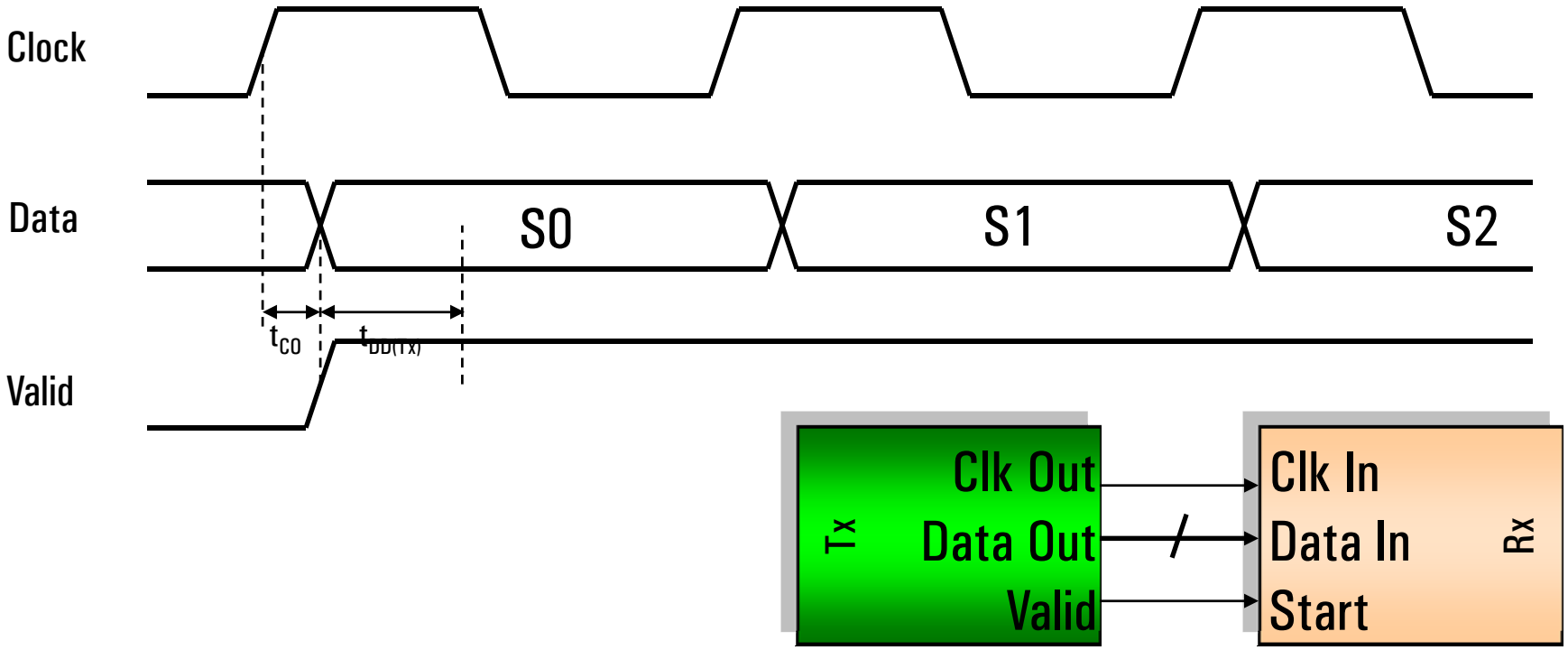
Synchronous Data Transfer

- "Clock to Out" time (t_{CO})
  - Tx Spec – Time between rising edge of clock and transition of data
  - Automatically puts rising and falling edges when Data/Valid are stable



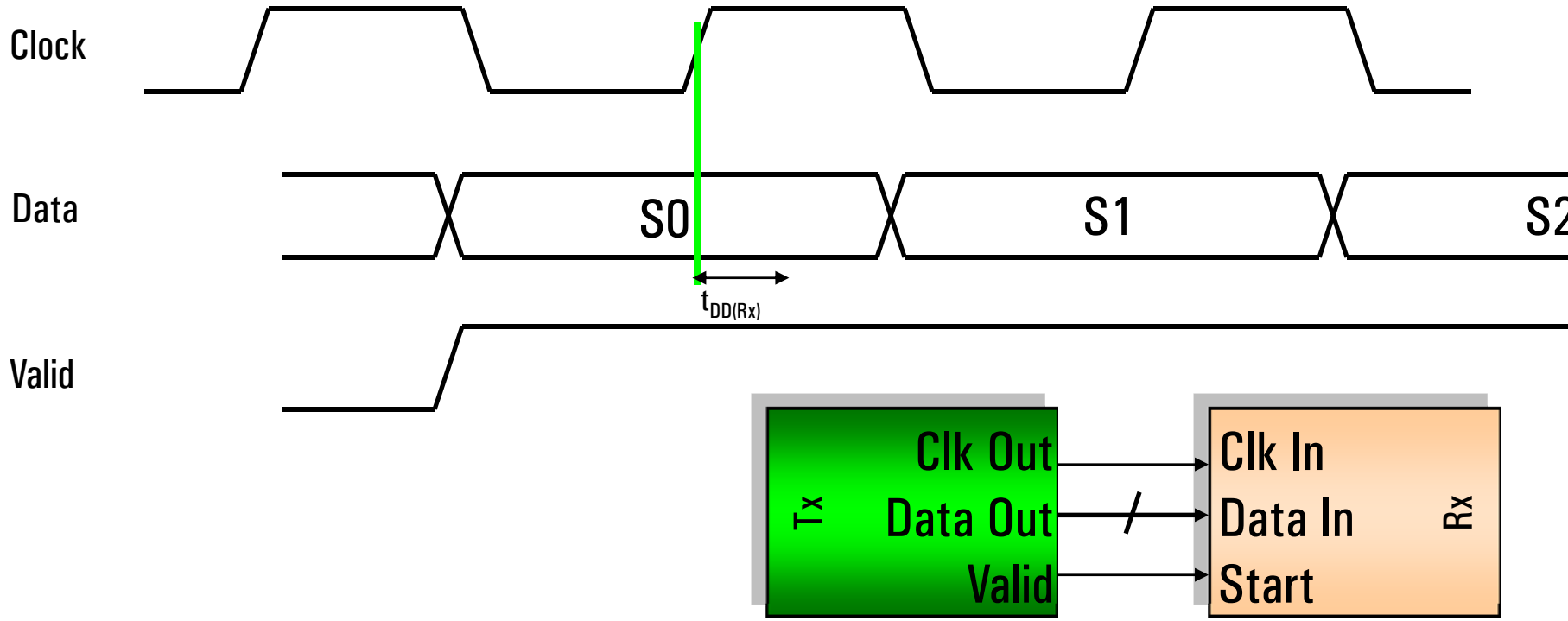
Synchronous Data Transfer

- Data Delay - *Generation*
  - Custom placement of generation or transmit (Tx) data
  - $t_{DD(Tx)}$ is additive to t_{CO}



Synchronous Data Transfer

- Data Delay – *Acquisition*
  - Delayed sampling or acquisition of data for receive (Rx)



Data Delay on NI Digital Products

- Available only on SMC products – NI 654x, 655x, 656x
  - One value for generation, one for acquisition *per board*
  - Can choose to apply delay for gen/acq *per channel*
- Fine resolution
  - Up to 1/256 of clock frequency (40 ps at 100 MHz)
- Clock frequency must be more than 25 MHz

NI 6541/42

NI 6551/52

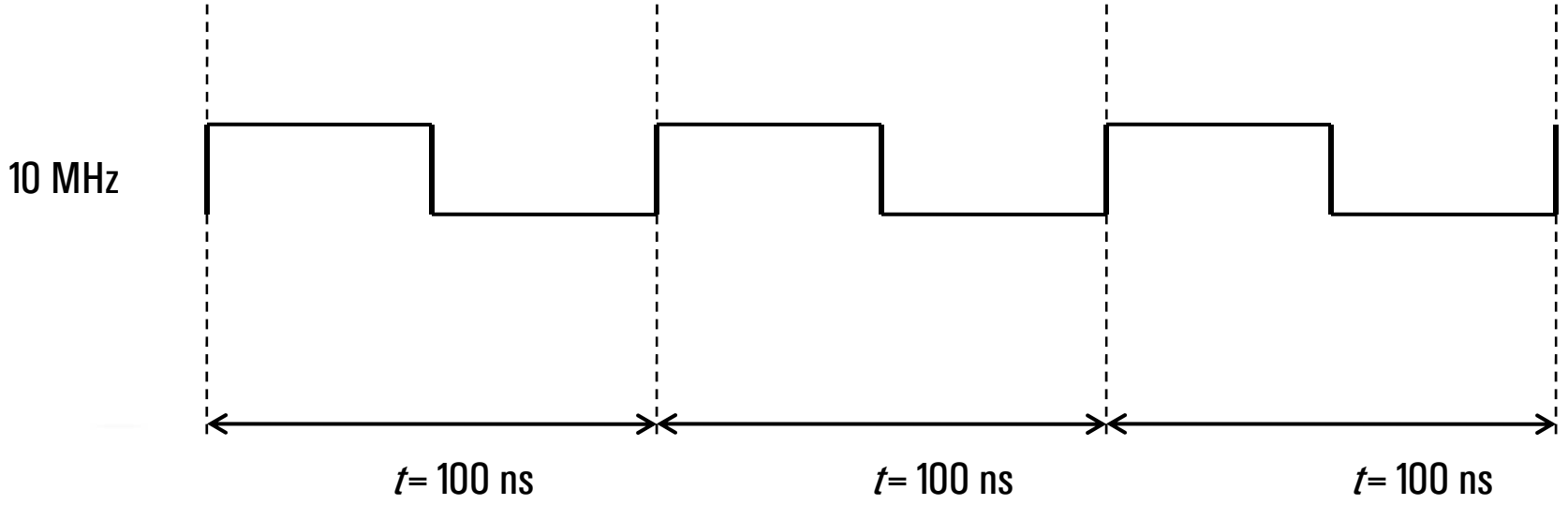
NI 6561/62

Timing – Oversampling

- Instead of running at the exact speed, run faster!
  - 2X, 4X, 8X oversampling generates finer resolution

Example

- 10 MHz sampling gives 1/freq = 100 ns edge placement

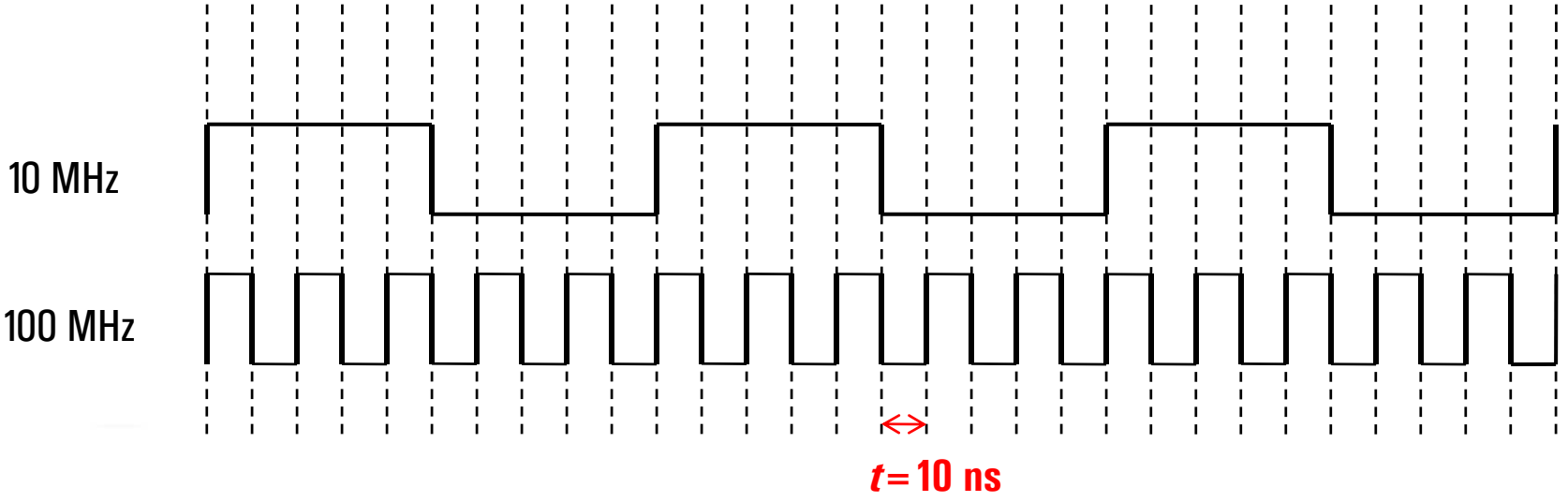


Timing – Oversampling

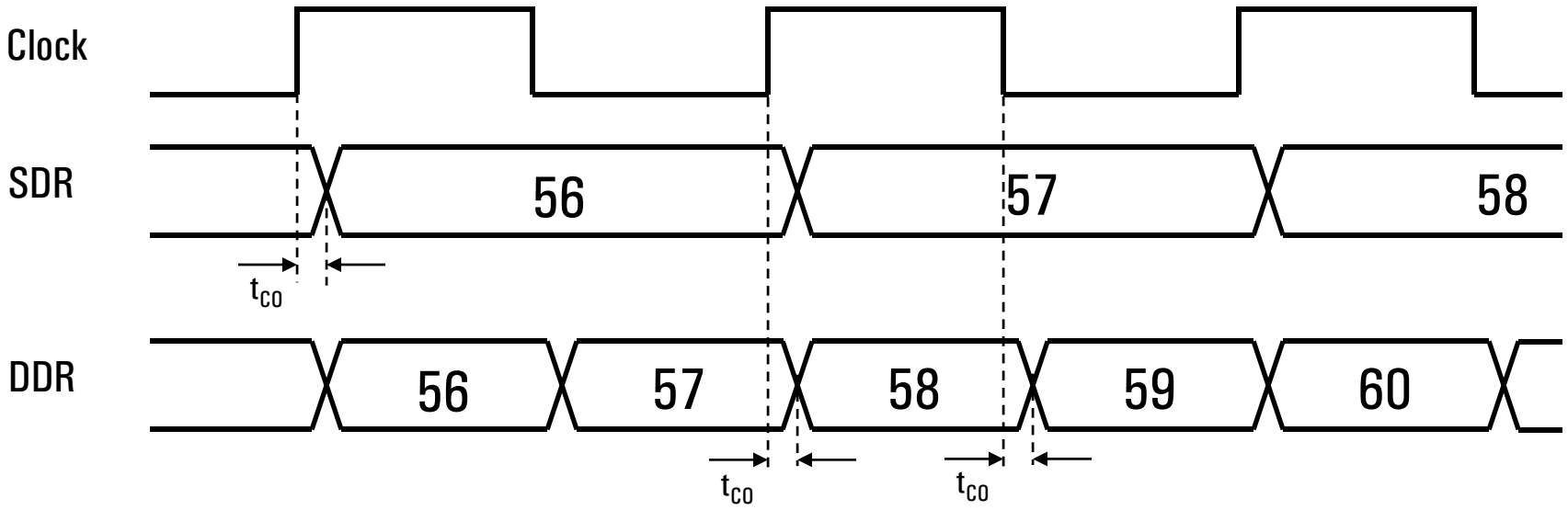
- Instead of running at the exact speed, run faster!
  - 2X, 4X, 8X oversampling generates finer resolution

Example

- *100 MHz* sampling gives 1/freq = ***10 ns*** edge placement



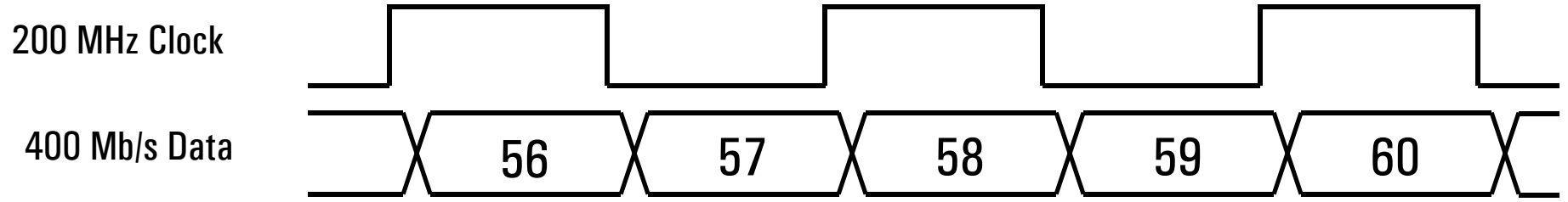
Synchronous Data Transfer – SDR vs. DDR



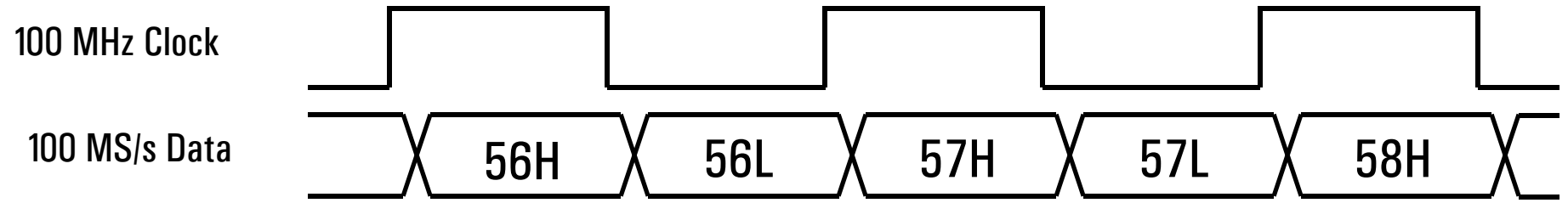
- SDR – Single Data Rate (one sample per period)
- DDR – Double Data Rate (two samples per period)
  - Feature on NI 656x LVDS products to provide up to 400 Mb/s

Synchronous Data Transfer: Why Use DDR?

- Allows higher-speed transfers with slower designs

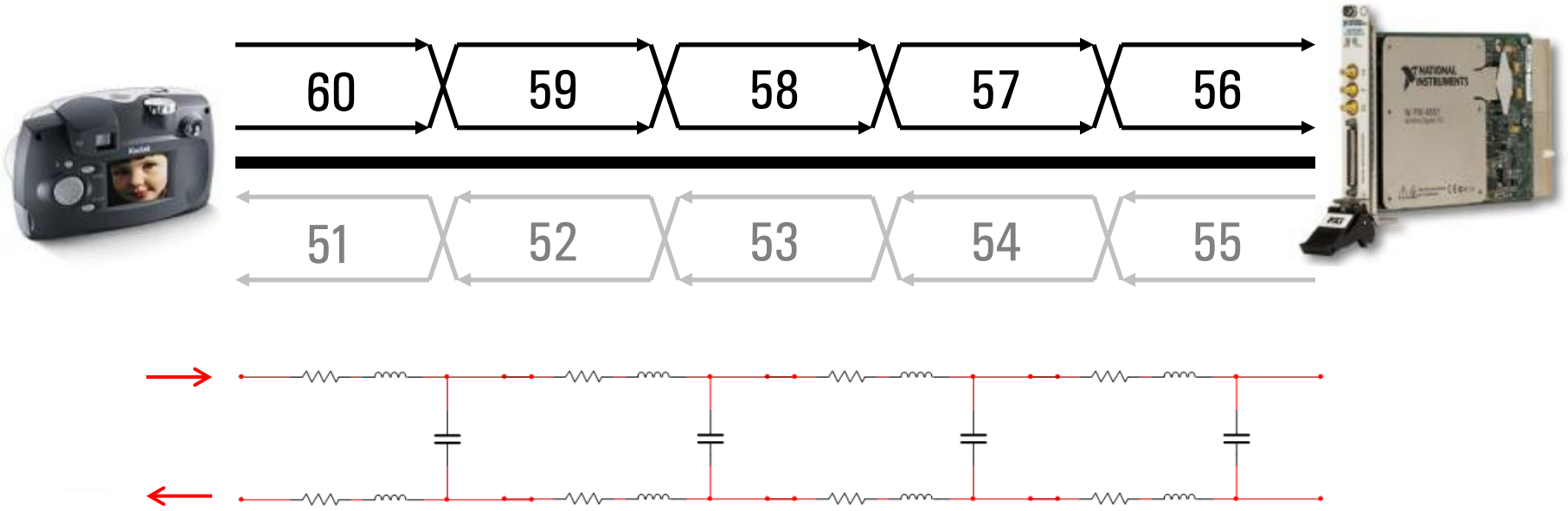


- Allows smaller IC package
  - e.g., send 32-bit samples at 100 MS/s with 100 MHz clock
    - Option 1: Send SDR on 32-pin bus
    - Option 2: Send DDR on 16-pin bus

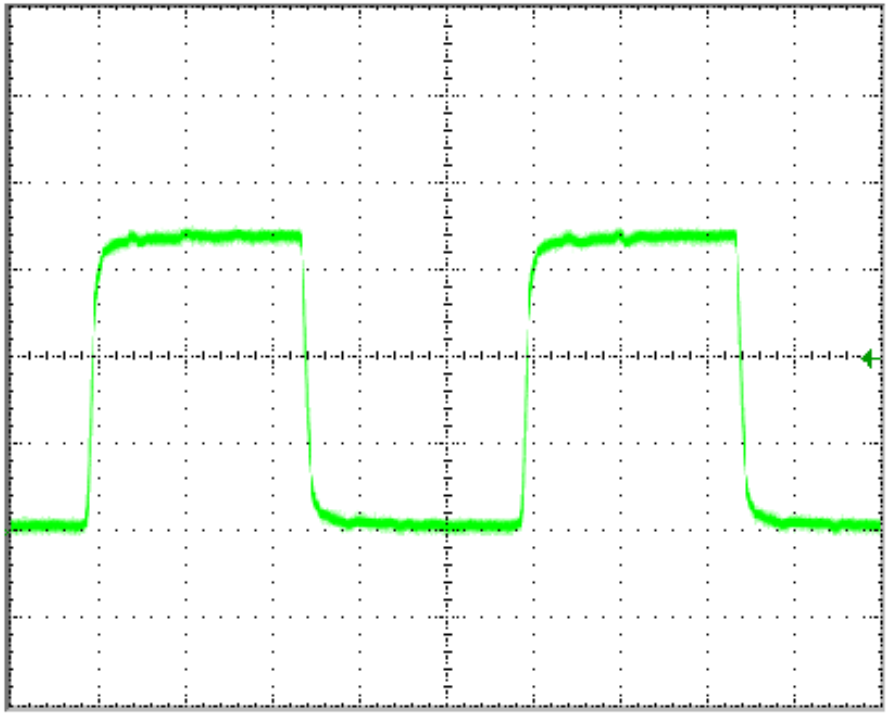


Probing and Termination

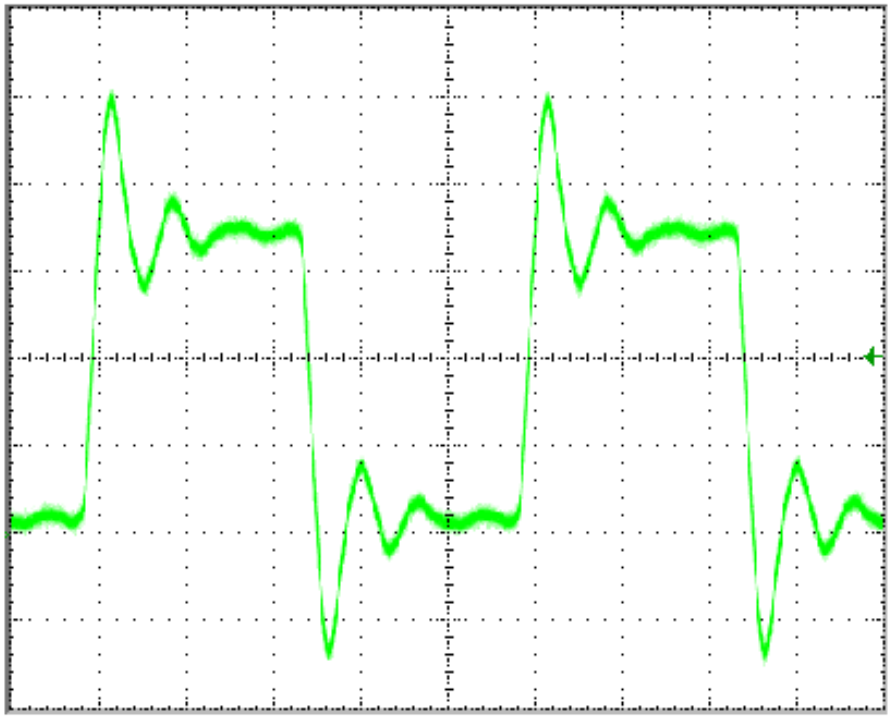
- High-speed digital communication – a cable is not *just* a cable
  - Multiple samples can be in the cable simultaneously
  - Without impedance matching, samples are reflected back
  - Cable's characteristic impedance is part of the circuit



Signal Termination

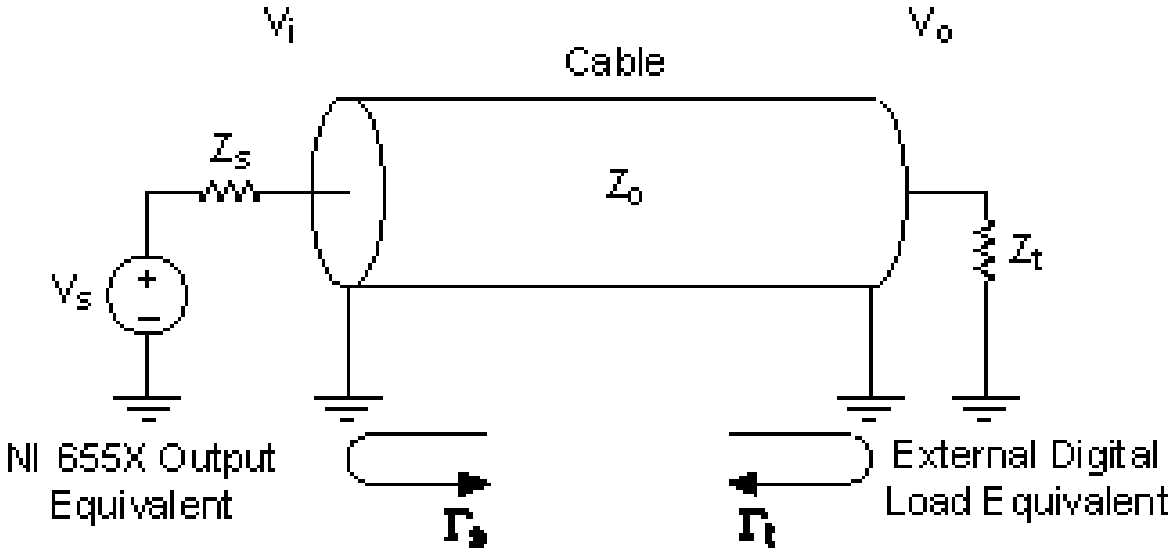


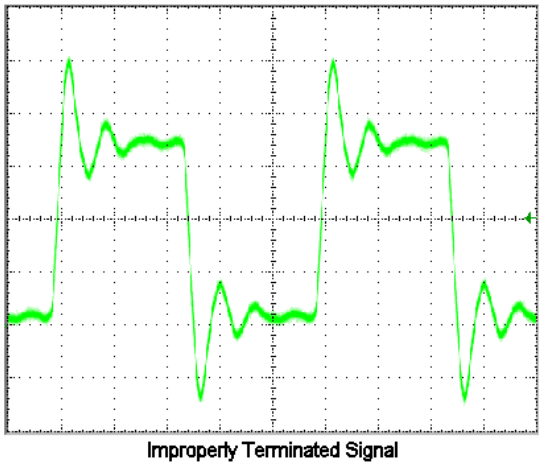
Properly Terminated Signal



Improperly Terminated Signal

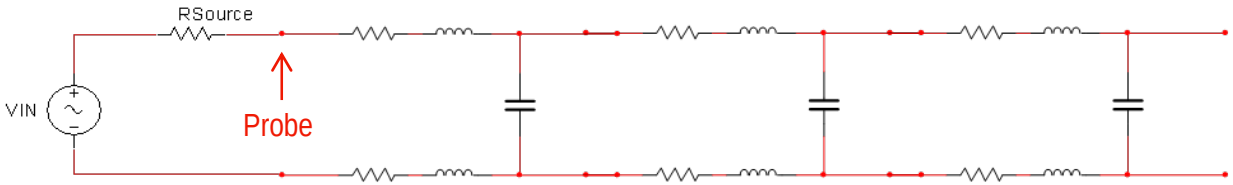
Signal Termination

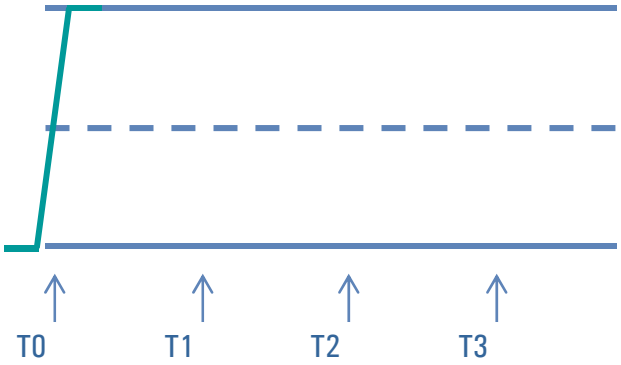




1. Mismatches in system resistances/impedance cause reflections
2. Reflections are the leading cause of poor signal integrity
3. To maximize signal integrity, minimize mismatches in system impedances

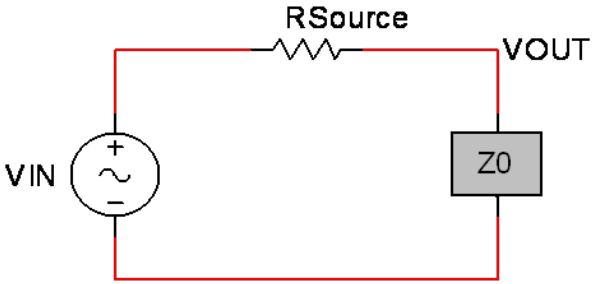
Probing and Termination





Probing and Termination

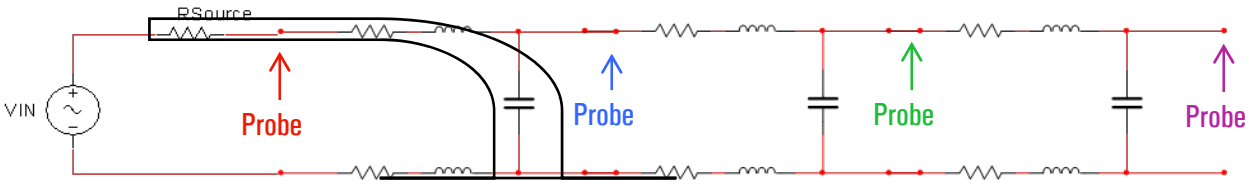
- Edge encounters voltage divider
  - Between source impedance and “characteristic” (AC) impedance

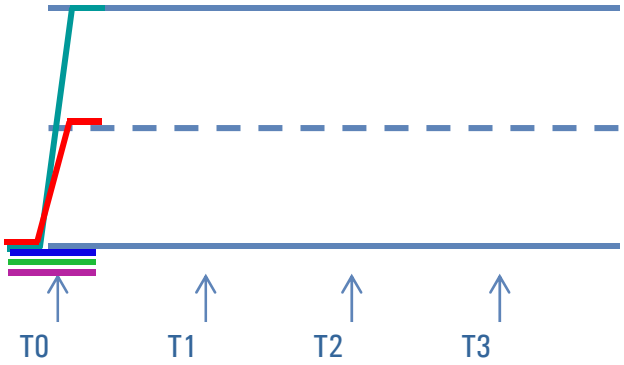


$$V_{OUT} = V_{IN} * \frac{Z_0}{R_S + Z_0}$$

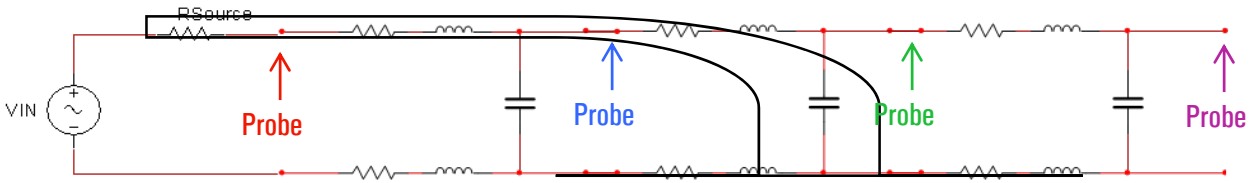
- This is purely an AC phenomenon
  - DC steady state does not see the reactive components of Z_0

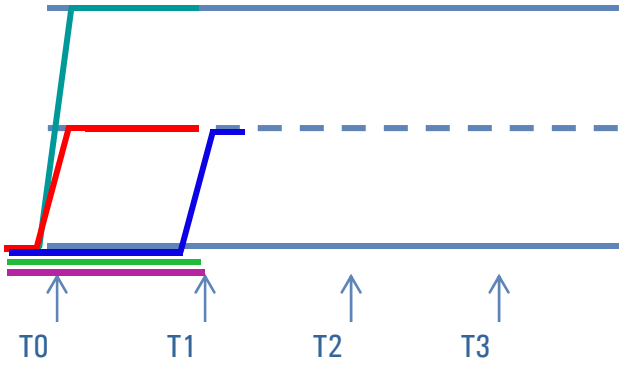
Probing and Termination



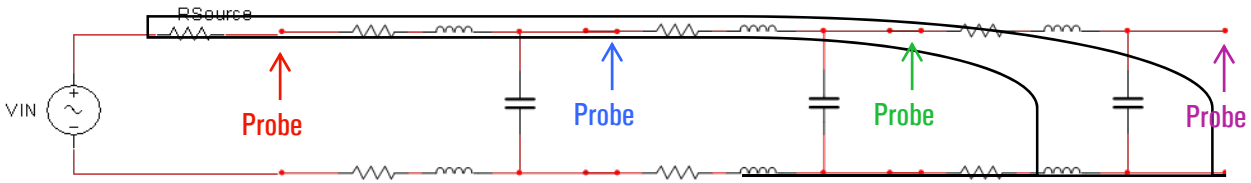


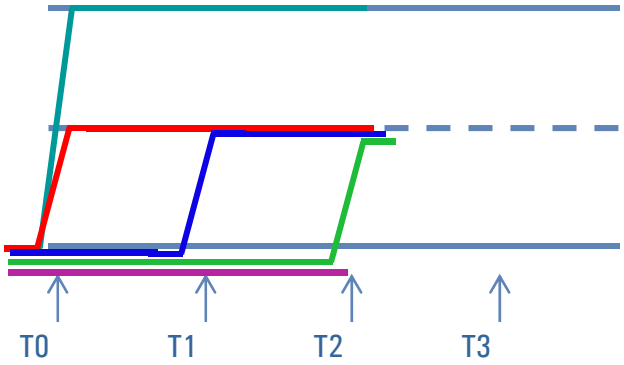
Probing and Termination



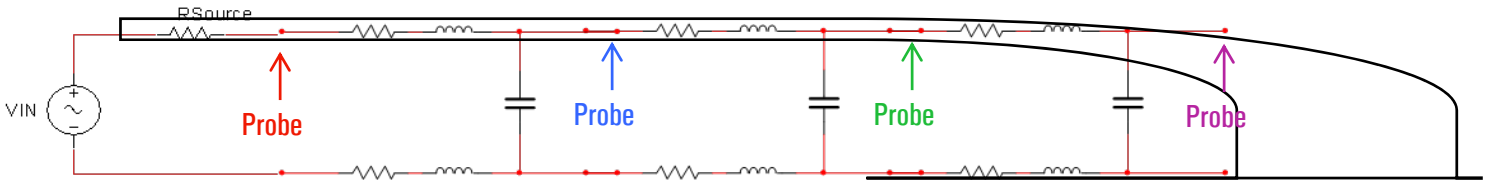


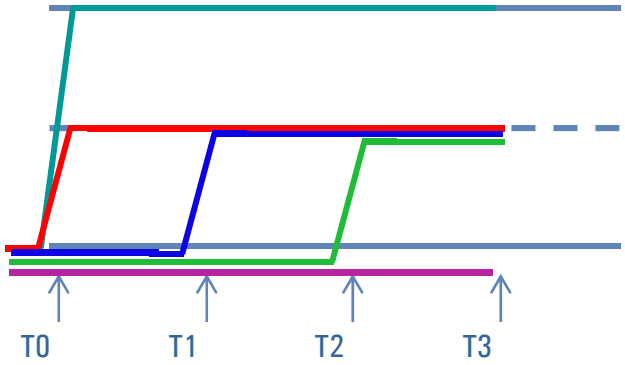
Probing and Termination



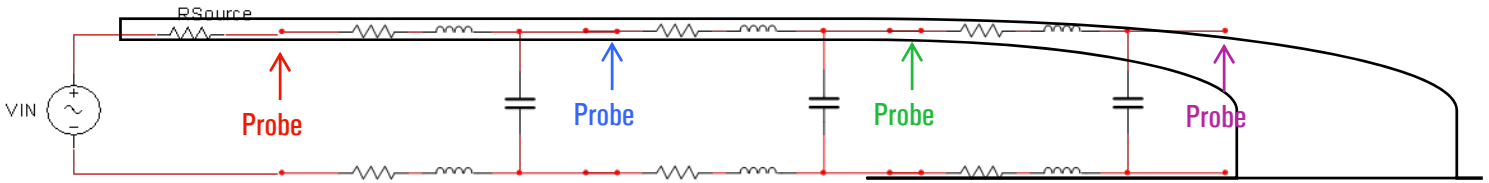


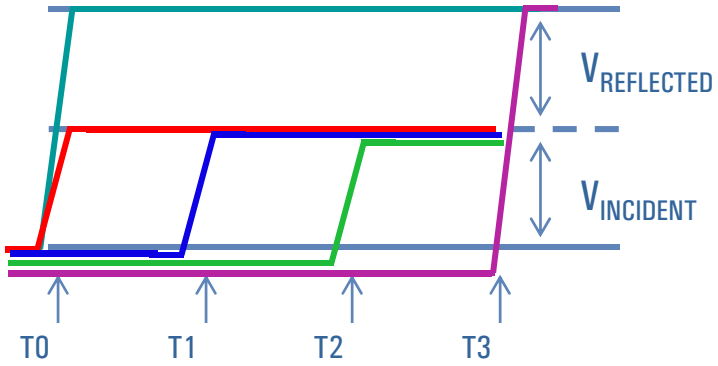
Probing and Termination



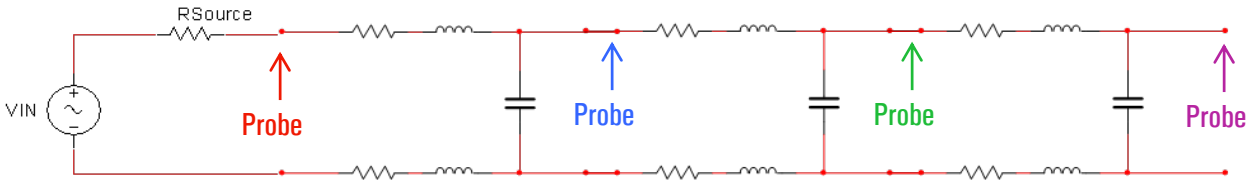


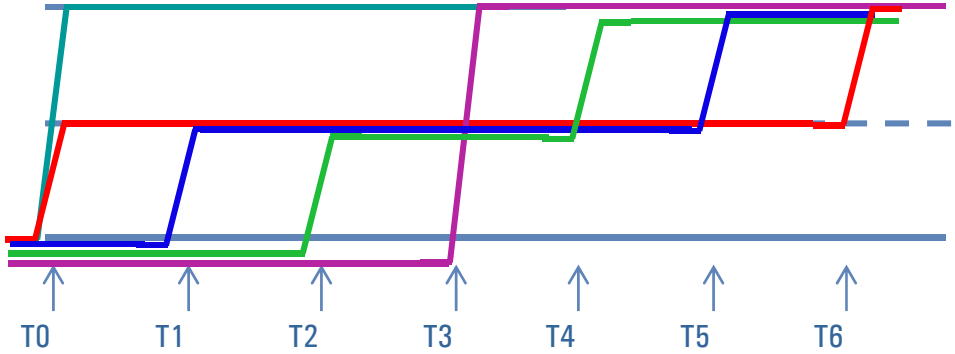
Probing and Termination





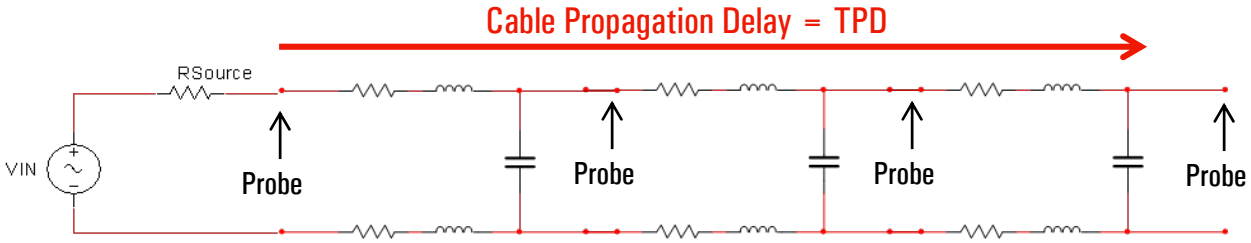
Probing and Termination

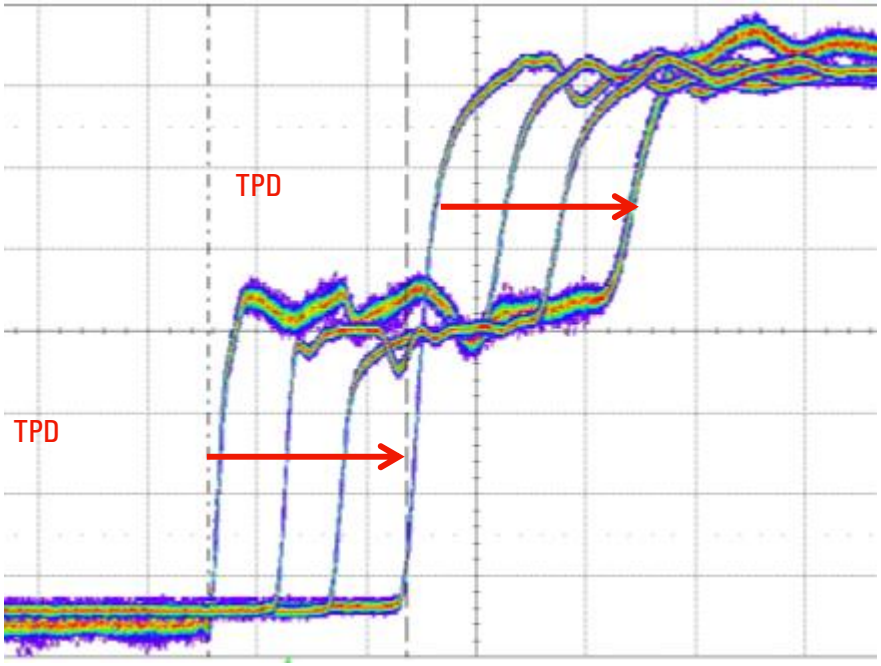




Probe as close to the receiver as possible!!!

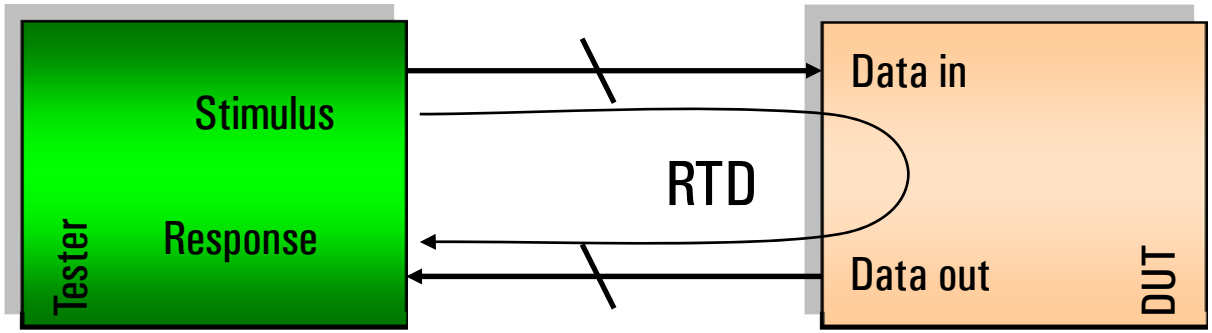
Probing and Termination

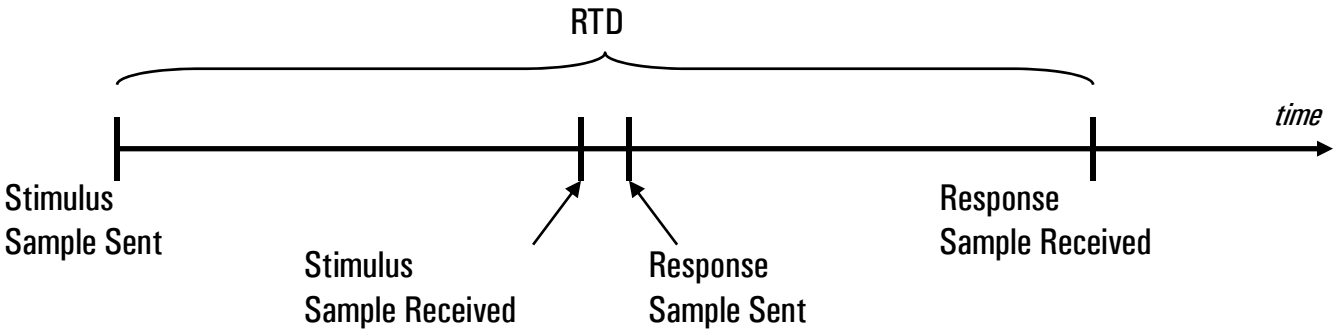




Compensating for Round-Trip Delay

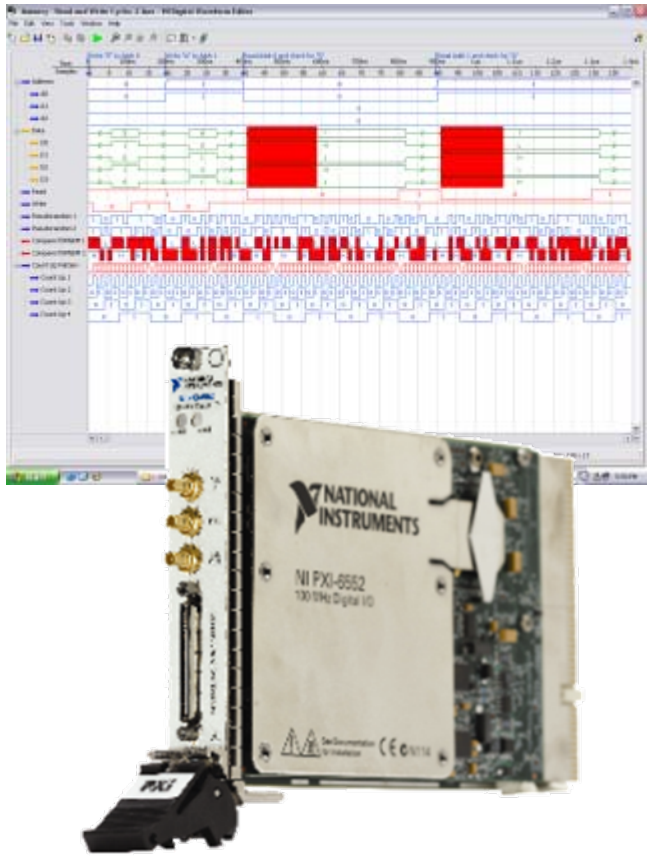
- Round-Trip Delay (RTD)
  - Time for stimulus to reach DUT and response to return
    - Includes cable length and any DUT propagation delay





NI Digital Waveform Editor 3.0

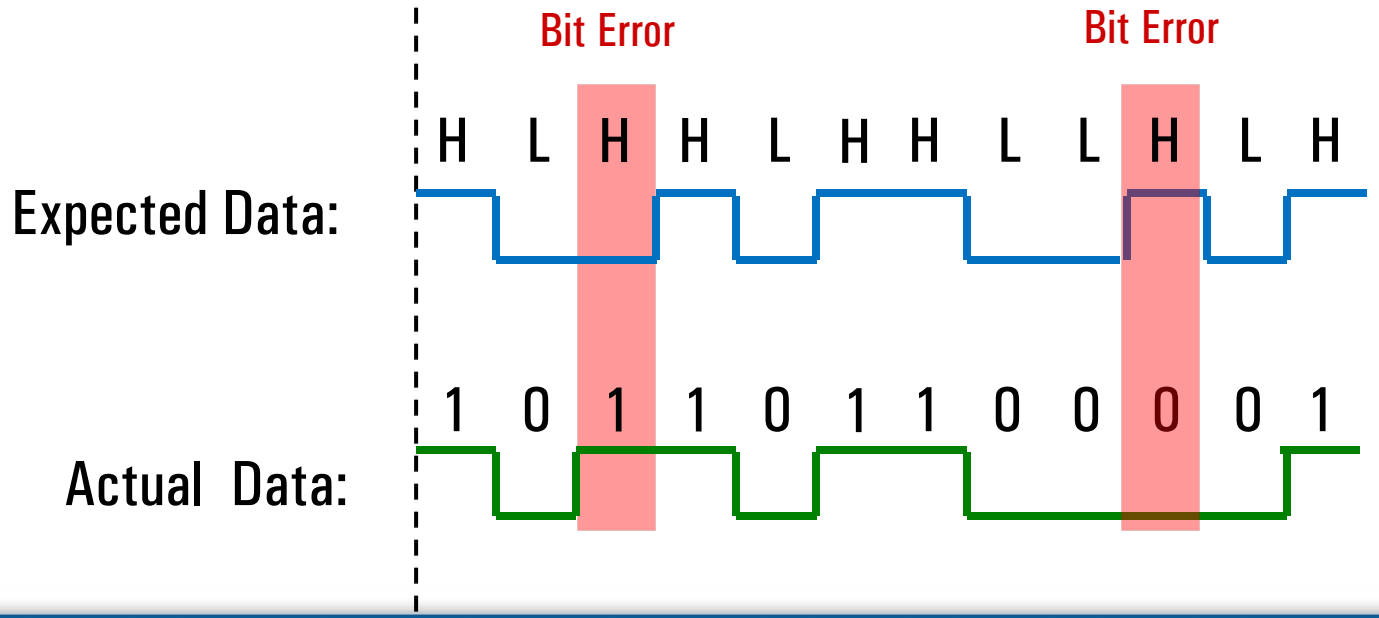
- Design digital waveforms
  - Drive (generate) - 0, 1, and Z (tri-state)
  - Compare (acquire) - H, L, and X
- Import VCD or ASCII files from VHDL
- Built-in patterns (Count up/down, PRBS, etc.)
- Create buses, add bookmarks, and measure timing with cursors
- Play waveforms with NI 654x, NI 655x, or NI 656x digital devices
- Highlight bit errors with NI 655x with hardware compare



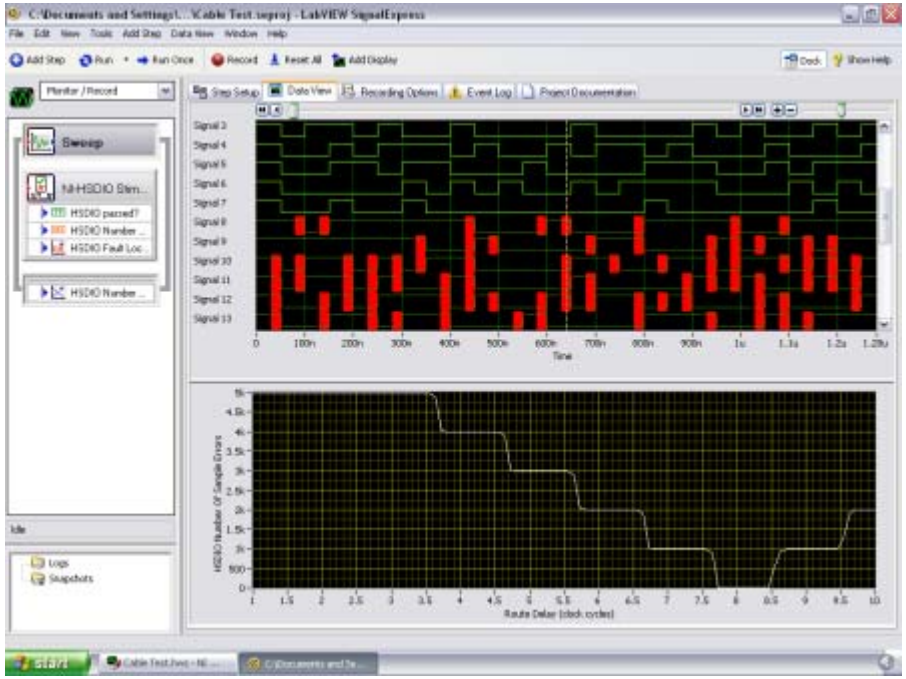
NI PXI-6552 and Digital Waveform Editor

Hardware Bit Comparison in Real Time

- Compare the value of expected data with actual response
- Define/load expected values prior to running (H, L, and X)
- Compare acquired bits against expected values in memory
- Errors show up in software API immediately



Demo: Measuring Cable Delay



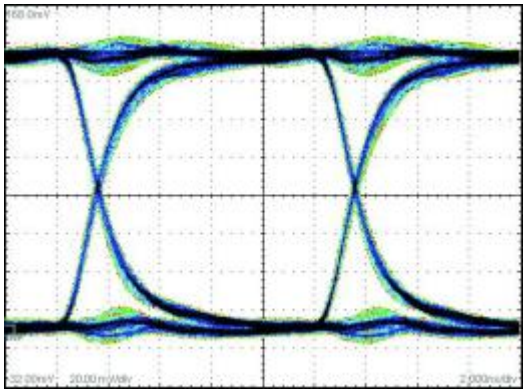
NI Digital Waveform Editor 3.0

NI LabVIEW SignalExpress 2.0

NI PXI-6552

For More Information …

- **High-Speed Digital Fundamentals**
  - Collection of white papers, demos, etc.
  - ni.com/hsdio



- **NI High-Speed Digital I/O Products**
  - Selectable voltages ranges
  - Programmable voltages ranges
  - Differential voltages ranges
  - PCI, PXI, PCI Express, PXI Express, and PCMCIA