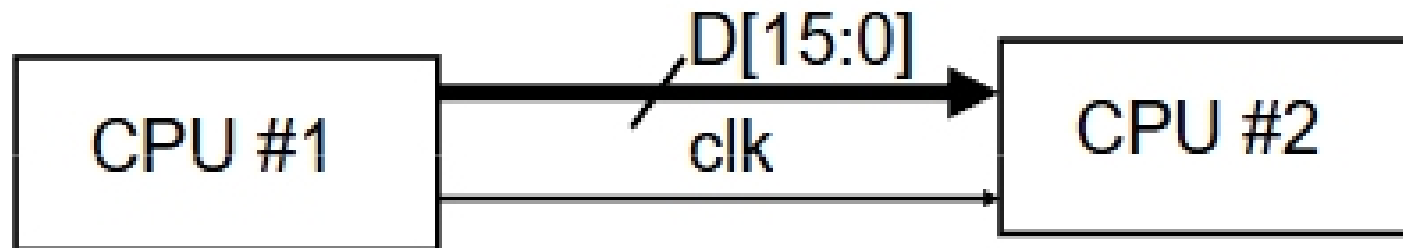


# Parallel IO (Input/Output)

Parallel IO – data sent over a group of parallel wires.  
Typically, a clock is used for synchronization.

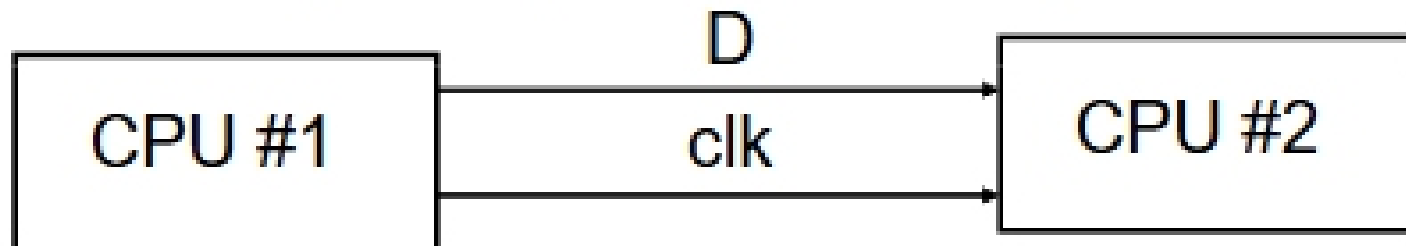


A 16-bit data channel is shown above. If data is transferred each rising clock edge, and clock rate is 300 MHz, then the **data transfer rate (bandwidth)** in bytes/sec is:

$$\begin{aligned} 2 \text{ Bytes/clock period} &= 2\text{Bytes} / (1/300\text{e}06)\text{s} \\ &= 2\text{B} * 300\text{e}06/\text{s} = 600\text{e}06\text{B/s} = 600 \times 10^6 \text{ B/s} \end{aligned}$$

# Serial IO

Serial IO – data sent one bit at a time, over a single wire.  
A clock may or may not be used for synchronization



Question: Assuming one bit is sent each rising clock edge, how fast does the clock have to be achieve  $600 \times 10^6$  B/s?

$$600 \times 10^6 \text{ B/s} = 600 \times 10^6 \text{ B/s} * 8 \text{ bits/1Byte} = 4800 \times 10^6 \text{ b/s}$$

$$1 \text{ bit/clock period} = 4800 \times 10^6 \text{ b/s}$$

$$1 \text{ bit} * \text{Clock Frequency} = 4800 \times 10^6 \text{ b/s}$$

$$\text{Clock Frequency} = 4800 \times 10^6 \text{ Hz} = 4.8 \text{ GHz}$$

# Parallel vs. Serial IO

## Parallel IO Pros/Cons

Pros: Speed, can increase bandwidth by either making data channel wider or increasing clock frequency

Cons: Expensive (wires cost money!). Short distance only – long parallel wire causes crosstalk, data corruption.

## Serial IO Pros/Cons

Pros: Cheap, very few wires needed. Good for long distance (“inches to feet”) interconnect.

Cons: Speed; the fastest serial link will typically have lower bandwidth than the fastest parallel link. However, due to faster integrated circuit technology, new fast serial IO standards (USB2/USB3, Firewire, SATA) have replaced older parallel IO standards.