CS144
An Introduction to Computer Networks

Physical Links

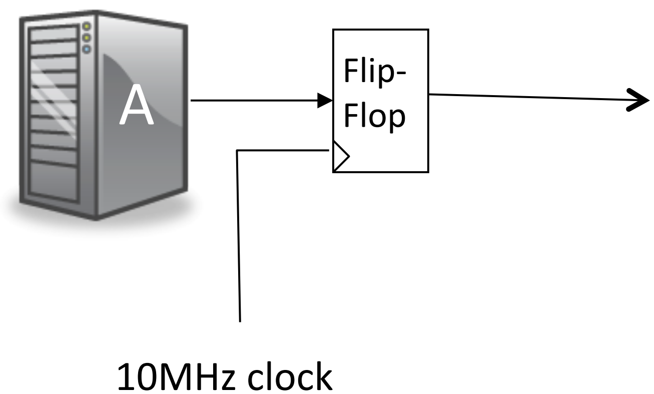
Clocks and Clock Recovery

Nick McKeown
Professor of Electrical Engineering
and Computer Science, Stanford University

Outline

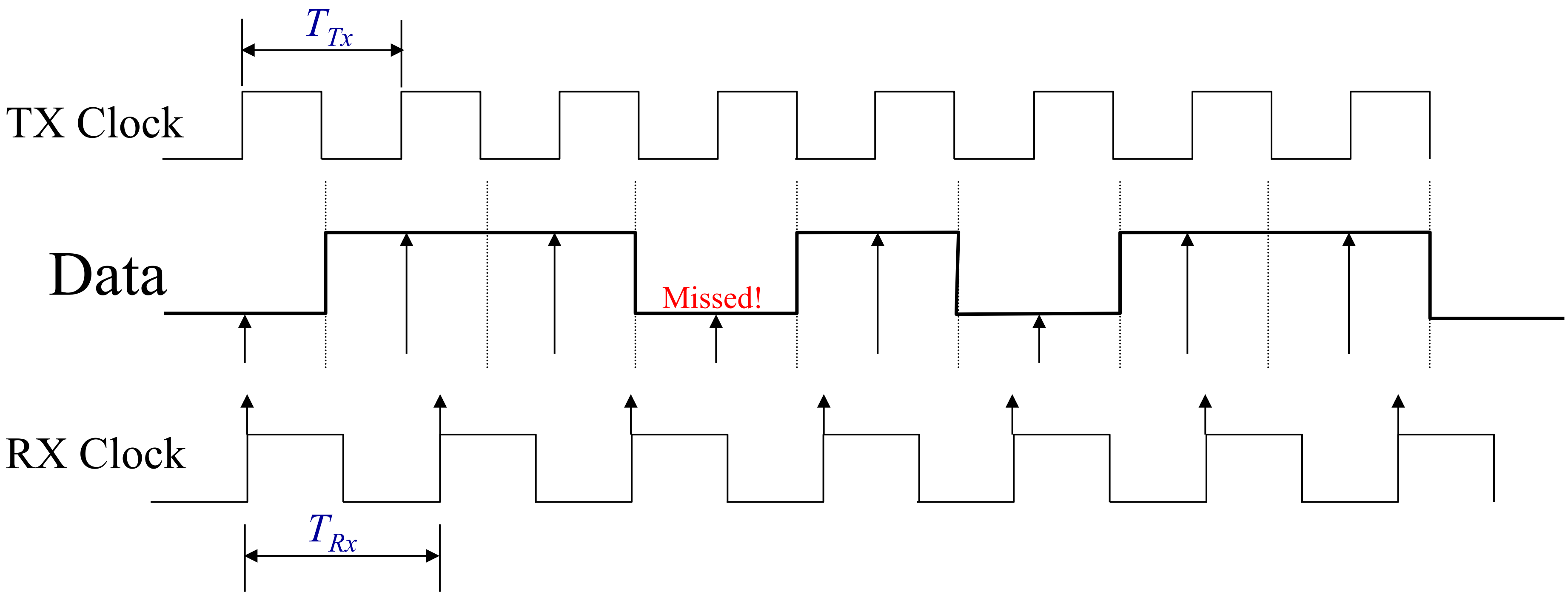
- Data is transmitted using a “clock”
- The receiver needs to know when to sample the arriving data
- Asynchronous communications
- Synchronous communications
  - Encoding the clock with the data
  - Recovering the clock
  - Getting the data into the receiver’s clock “domain”

Data is transmitted using a clock



If we don't know the sender's clock

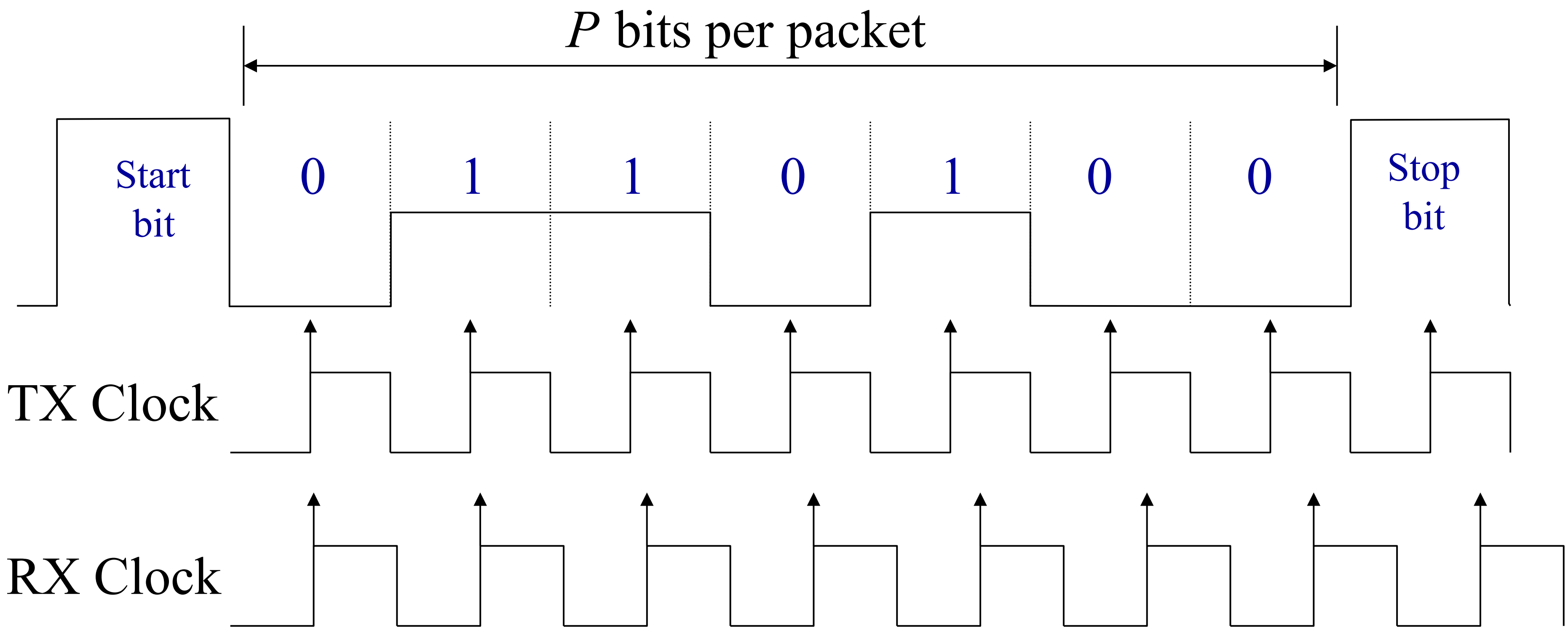
If we don't know the sender's clock



If the RX clock is p ppm <u>*slower*</u> *than the TX clock, then:* $T_{Rx} = T_{Tx}(1+10^{-6}p)$.

After $\dfrac{0.5}{10^{-6}p}$ *bit times, the RX clock will miss a bit.*

Asynchronous communication

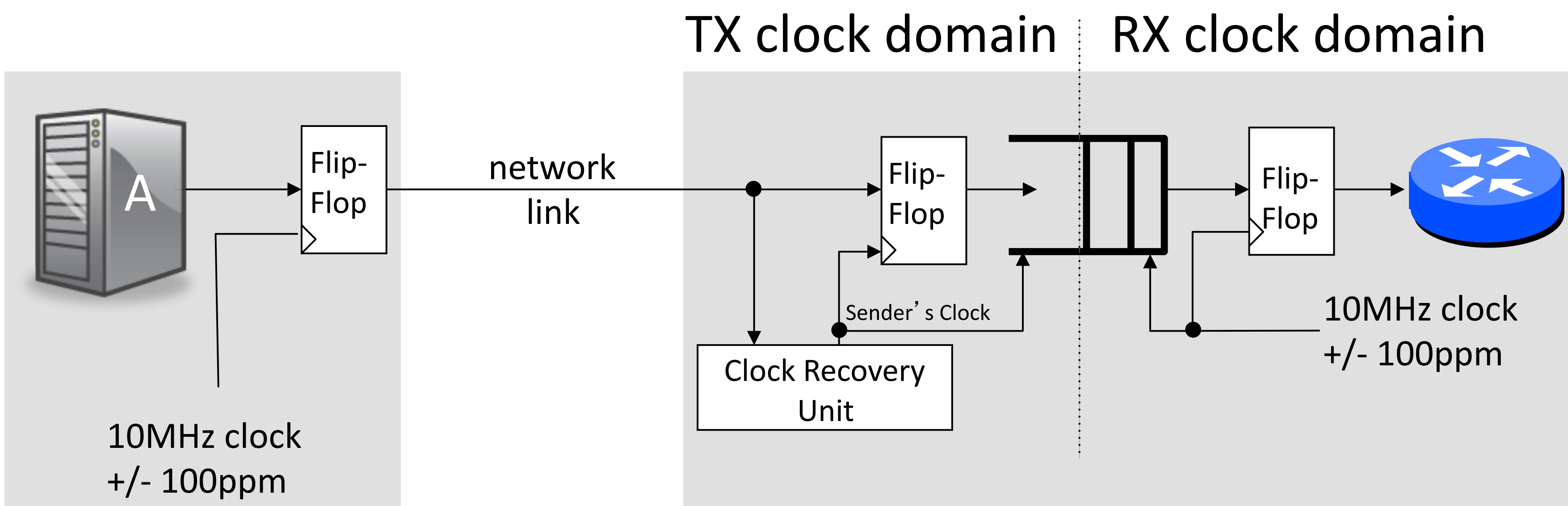


Asynchronous communications sometimes used for links with short packets e.g. IR remote control, serial links.

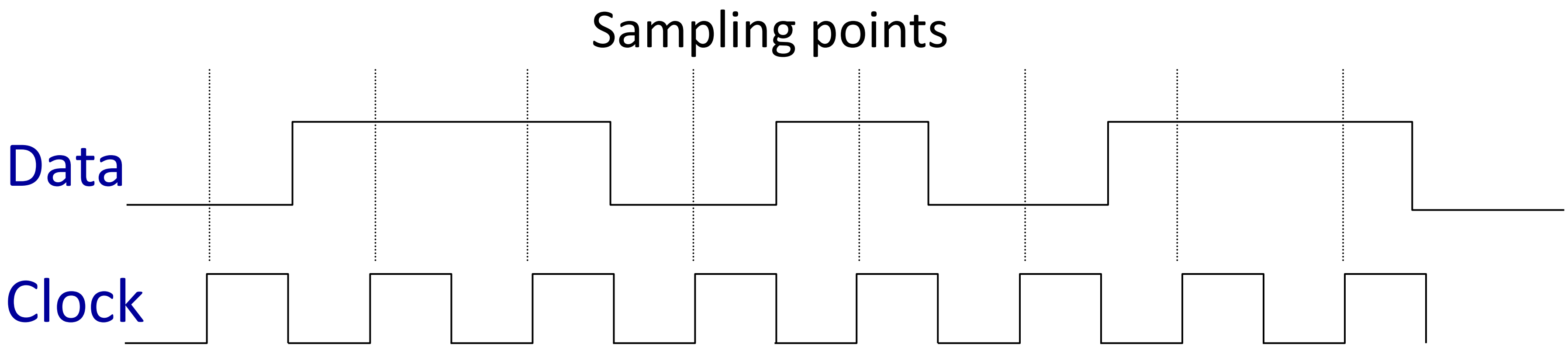
Outline

- Data is transmitted using a “clock”
- The receiver needs to know when to sample the arriving data
- Asynchronous communications
- Synchronous communications
  - Encoding the clock with the data
  - Recovering the clock
  - Getting the data into the receiver’s clock “domain”

Synchronous Communication



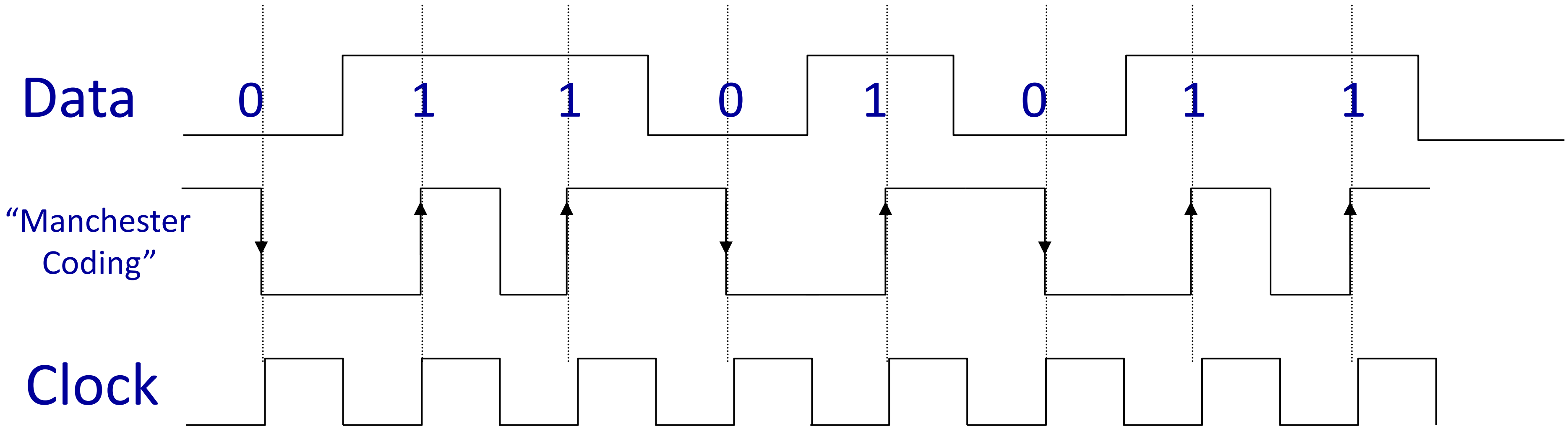
Encoding for clock recovery



If the clock is not sent separately, the data stream must have sufficient transitions so that the receiver can determine the clock.

Encoding for clock recovery

Example #1: 10Mb/s Ethernet



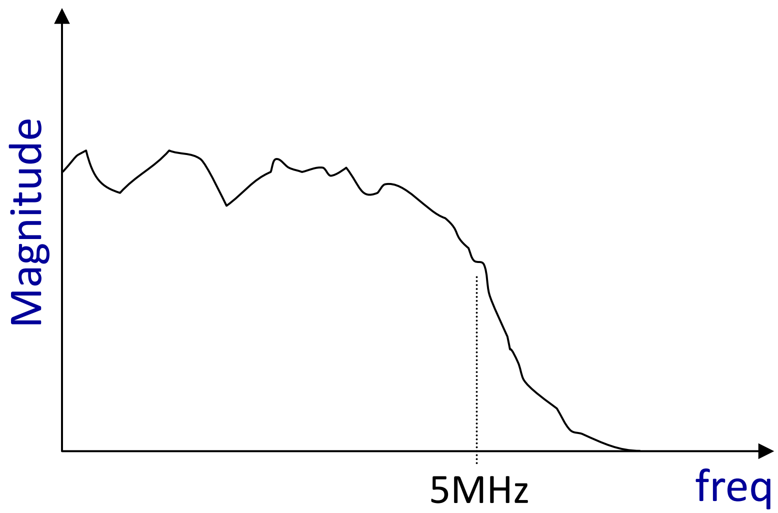
Advantages of Manchester encoding:

- Guarantees one transition per bit period.
- Ensures d.c. balance (i.e. equal numbers of hi and lo).

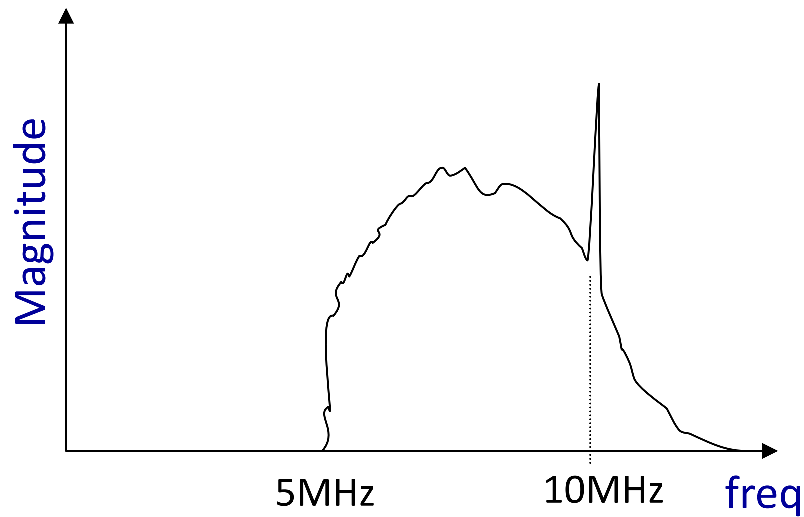
Disadvantages

- Doubles bandwidth needed in the worst case.

Frequency Spectrum for 10Mb/s Ethernet



Without Manchester coding



With Manchester coding

Encoding for clock recovery

Example #2: 4b5b encoding

4-bit data	**5-bit code**
0000	11110
0001	01001
0010	10100
…	…

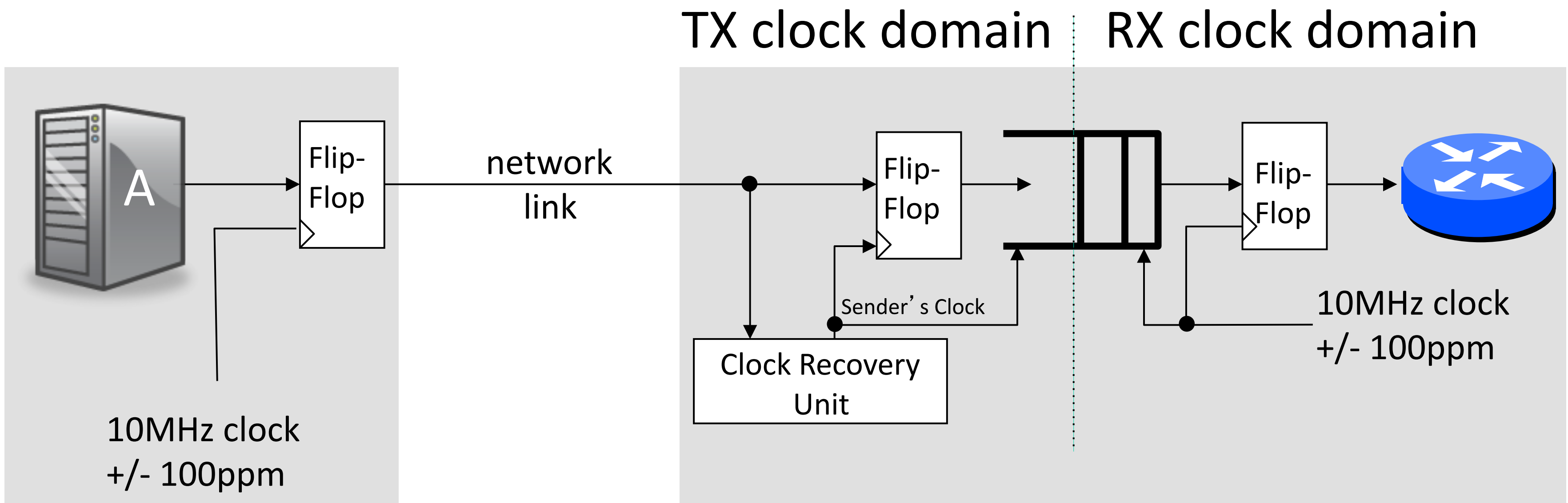
Advantages of 4b/5b encoding:

- More bandwidth efficient (only 25% overhead).
- Allows extra codes to be used for control information.

Disadvantages

- Fewer transitions makes clock recovery a little harder.

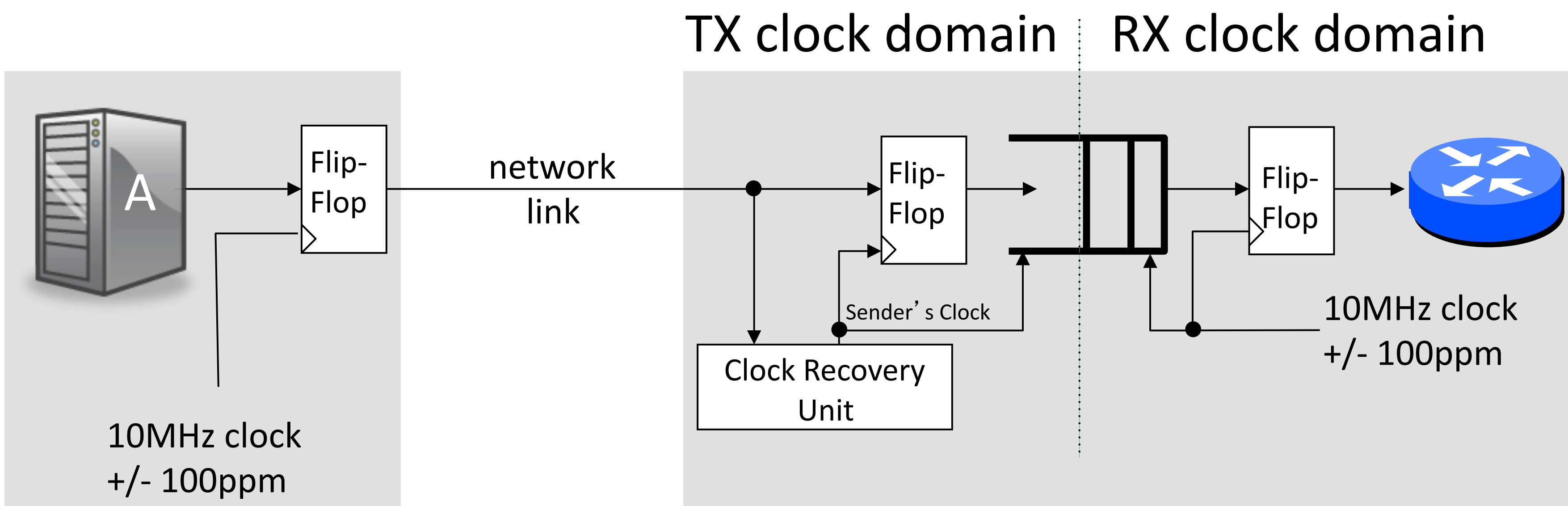
Summary



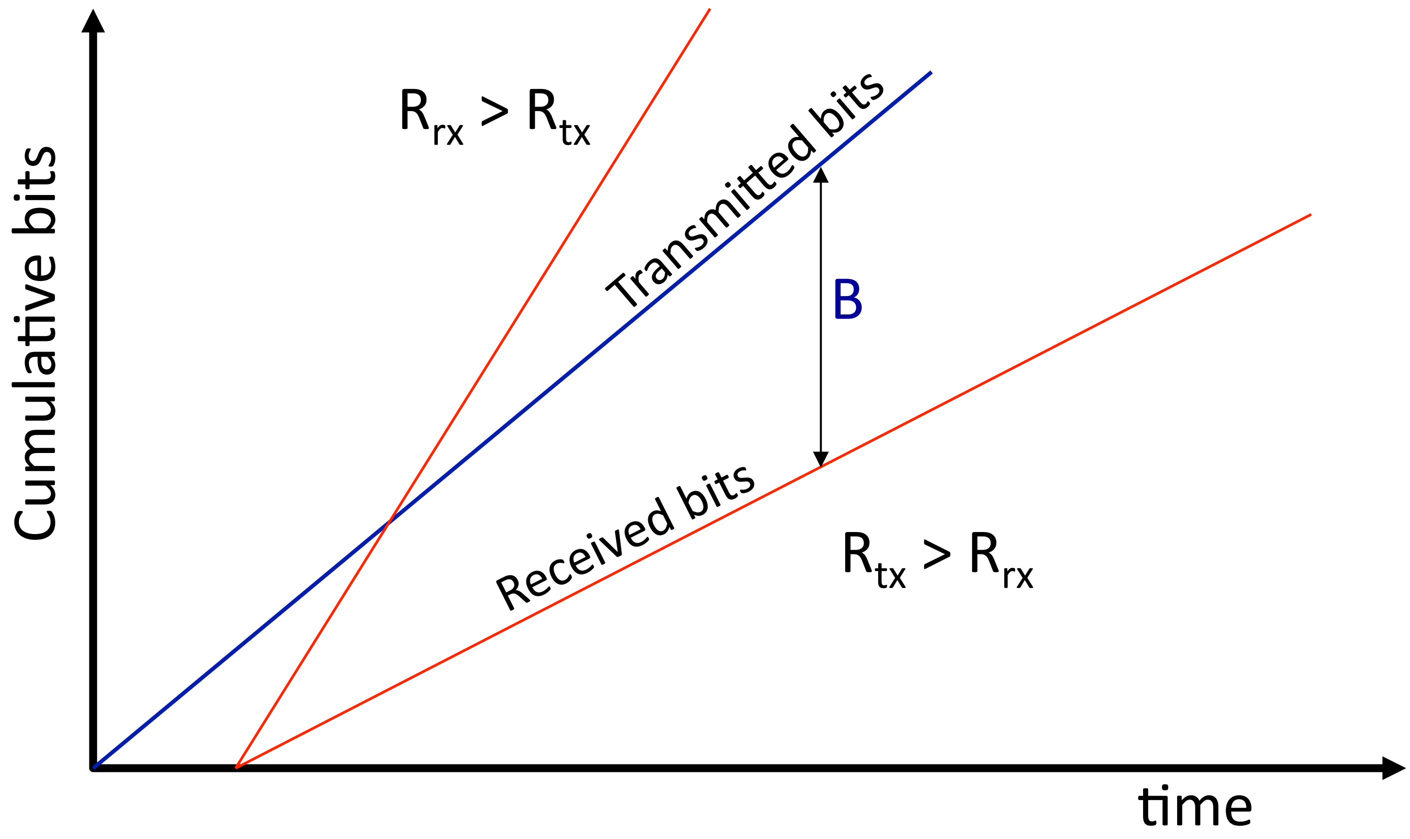
TX clock domain
RX clock domain
A
Flip-Flop
network link
Flip-Flop
Flip-Flop
Sender's Clock
Clock Recovery Unit
10MHz clock +/- 100ppm
10MHz clock +/- 100ppm

<end>

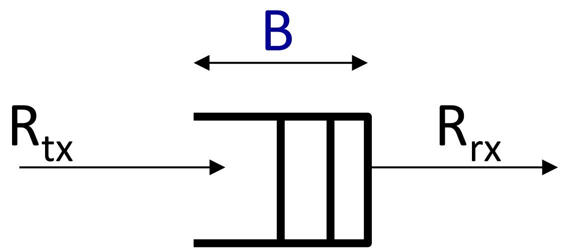
Elasticity Buffer



Sizing an elasticity buffer

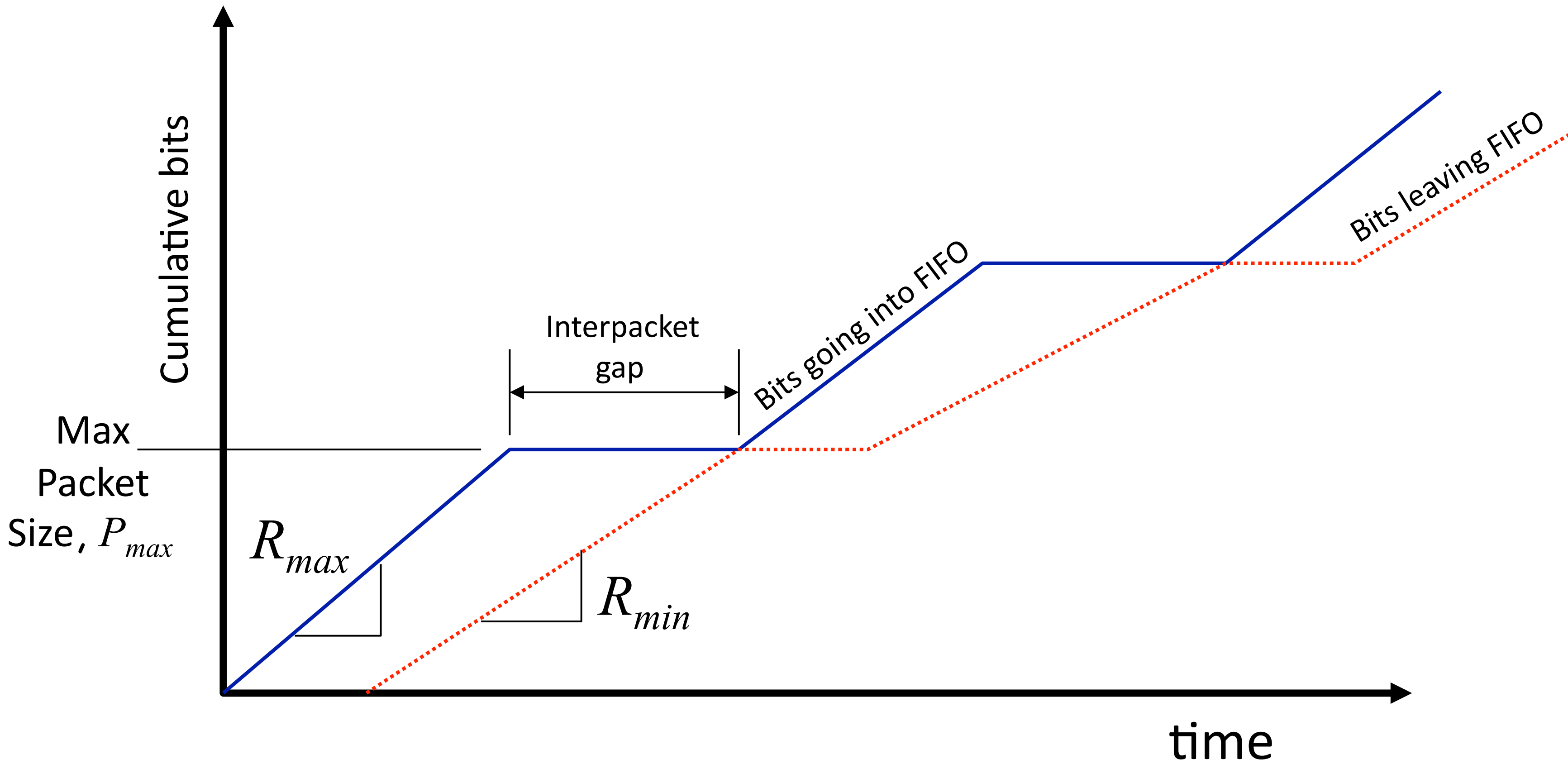


Sizing an elasticity buffer

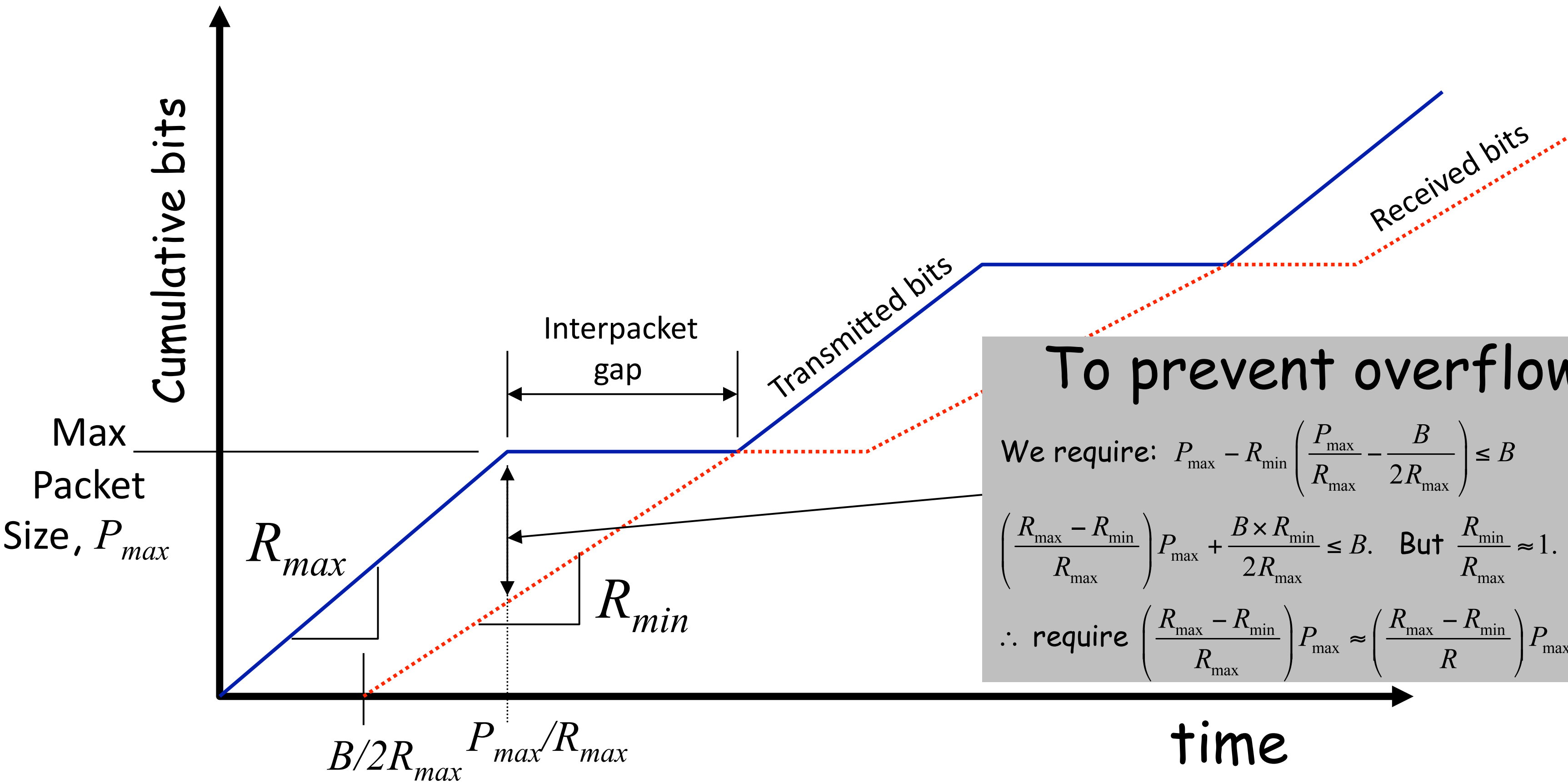


1. Hold buffer nominally at B/2.
   - At start of new packet, allow buffer to fill to B/2.
   - Or, make sure buffer drains to B/2 before new packet.
2. Size buffer so that it does not overflow or underflow before packet completes.
3. ($R_{tx} > R_{rx}$): Given inter packet gap, size B/2 for no overflow.
4. ($R_{rx} > R_{tx}$): Given max length packet, pick B/2 for no underflow.

Preventing overflow



Preventing overflow



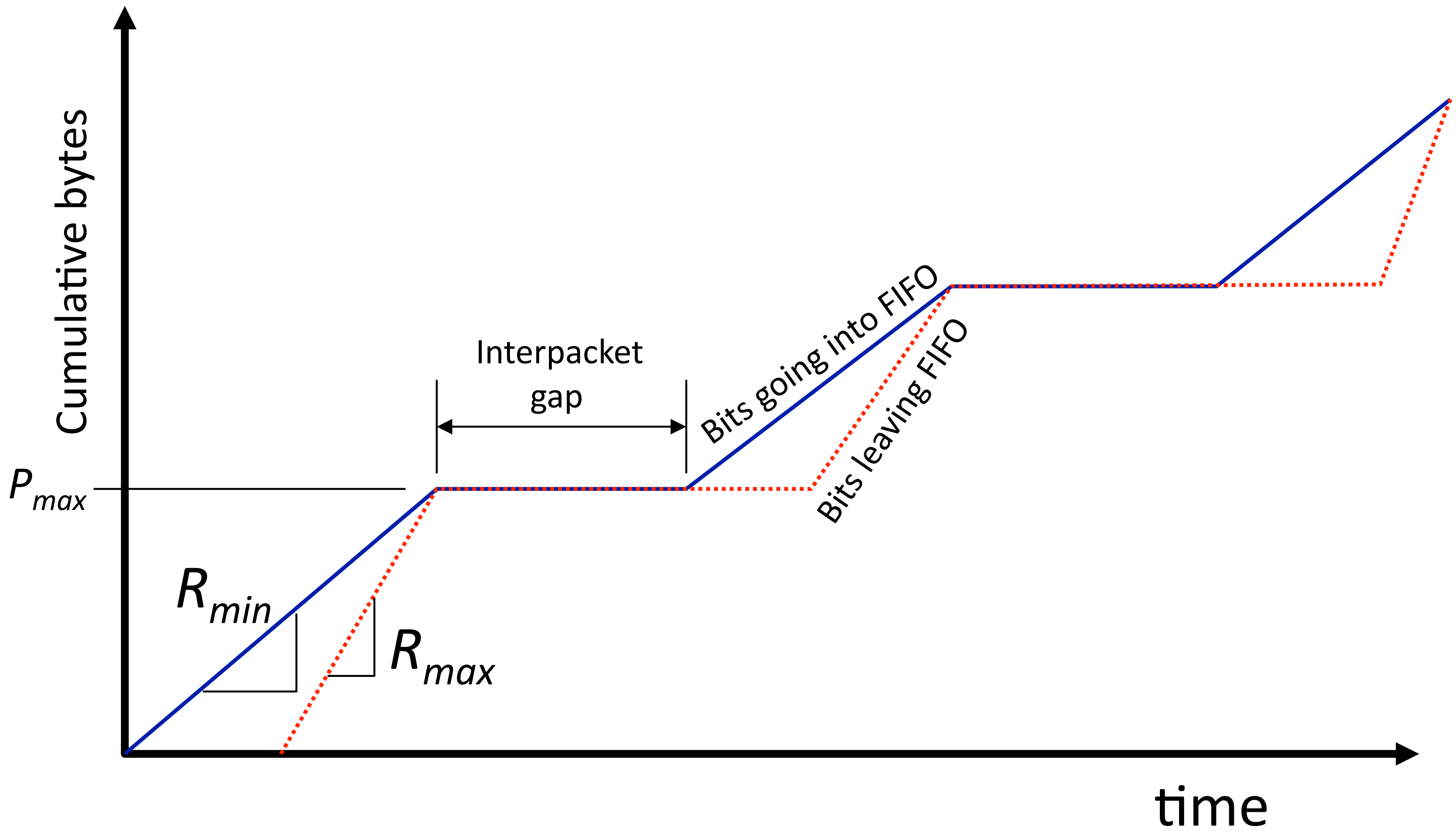
To prevent overflow:

We require: $P_{max} - R_{min}\left(\frac{P_{max}}{R_{max}} - \frac{B}{2R_{max}}\right) \le B$

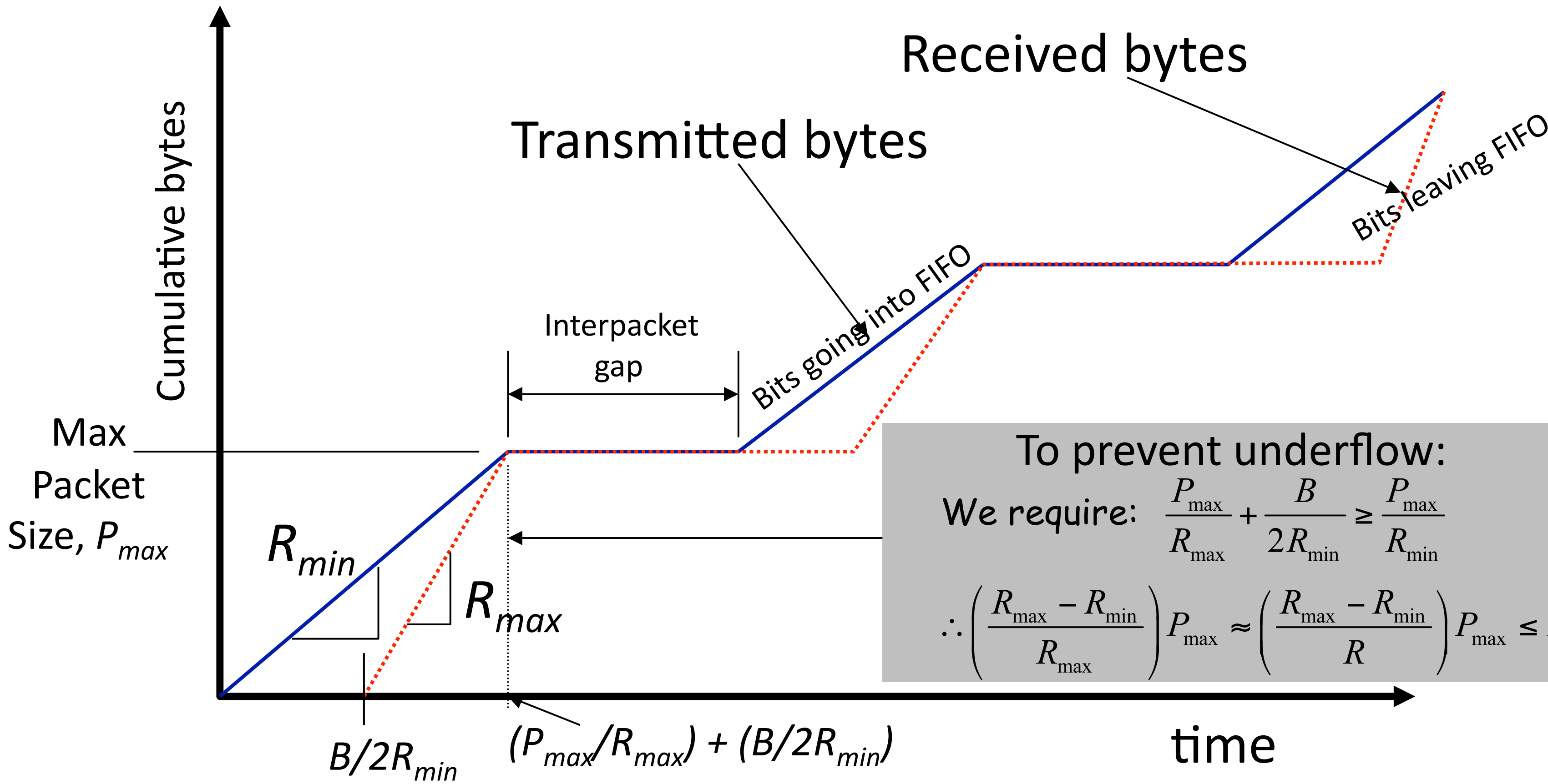
$\left(\frac{R_{max} - R_{min}}{R_{max}}\right) P_{max} + \frac{B \times R_{min}}{2R_{max}} \le B.$ But $\frac{R_{min}}{R_{max}} \approx 1.$

$\therefore$ require $\left(\frac{R_{max} - R_{min}}{R_{max}}\right) P_{max} \approx \left(\frac{R_{max} - R_{min}}{R}\right) P_{max} \le B/2$

Preventing underflow



Preventing underflow



To prevent underflow:

We require: $$\frac{P_{\max}}{R_{\max}} + \frac{B}{2R_{\min}} \geq \frac{P_{\max}}{R_{\min}}$$

$$\therefore \left(\frac{R_{\max} - R_{\min}}{R_{\max}}\right) P_{\max} \approx \left(\frac{R_{\max} - R_{\min}}{R}\right) P_{\max} \leq B/2$$

Sizing an elasticity buffer

Example

Maximum packet size 4500bytes

Clock tolerance +/- 100ppm

$$\left(\frac{R_{max} - R_{min}}{R}\right) = 200 \times 10^{-6}$$

$$\therefore B \geq 2(4500 \times 8 \times 200 \times 10^{-6}) = 14 bits$$

Therefore,

1. Buffer larger than 14 bits
2. Wait for at least 7 bits before draining buffer
3. Inter-packet gap at least 7 bits