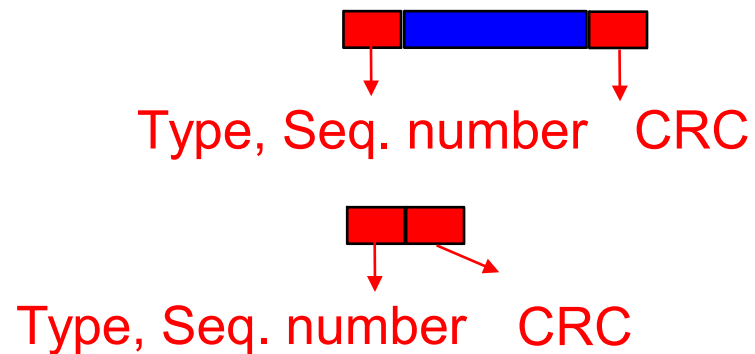
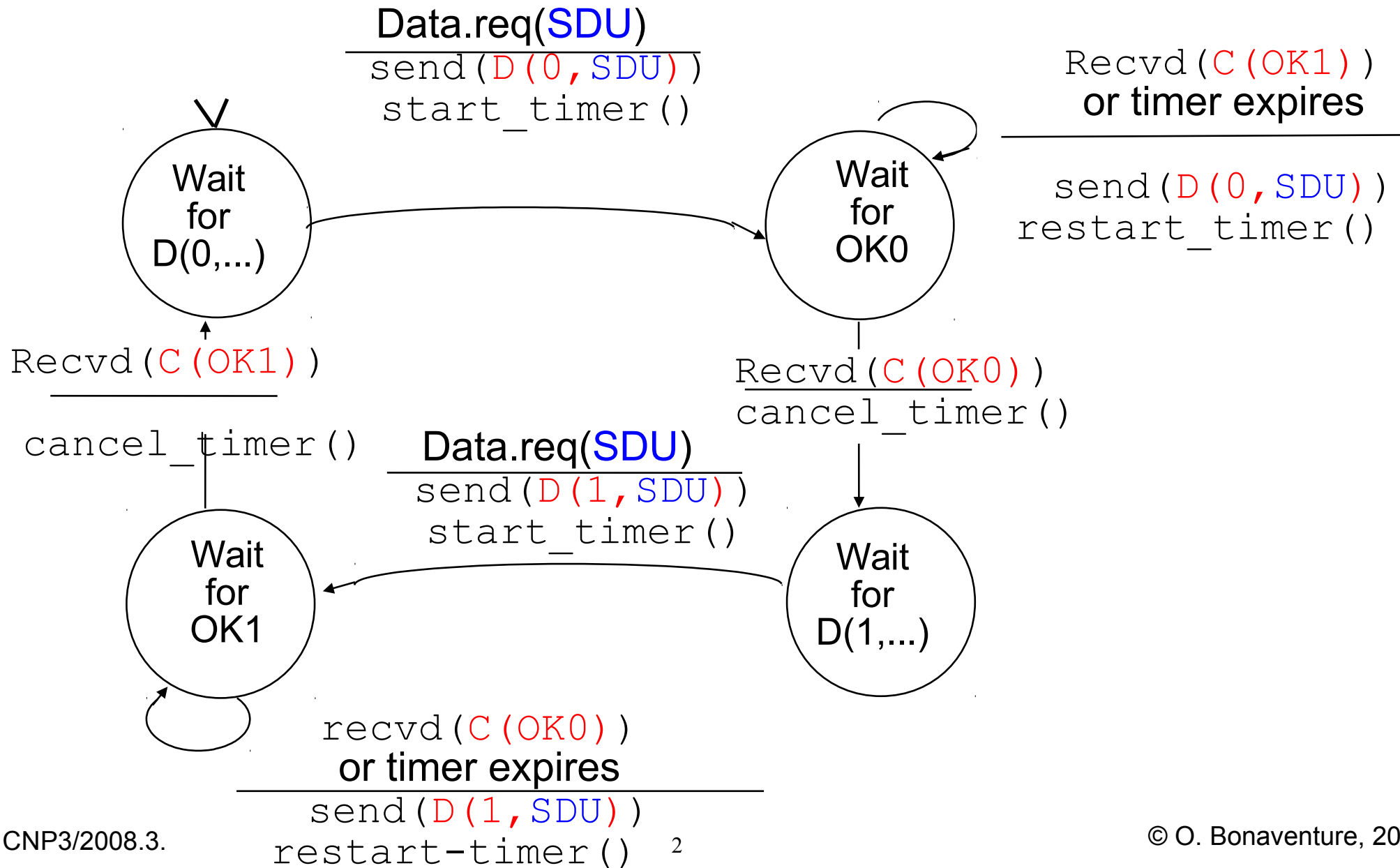


Alternating bit protocol – corrupt messages

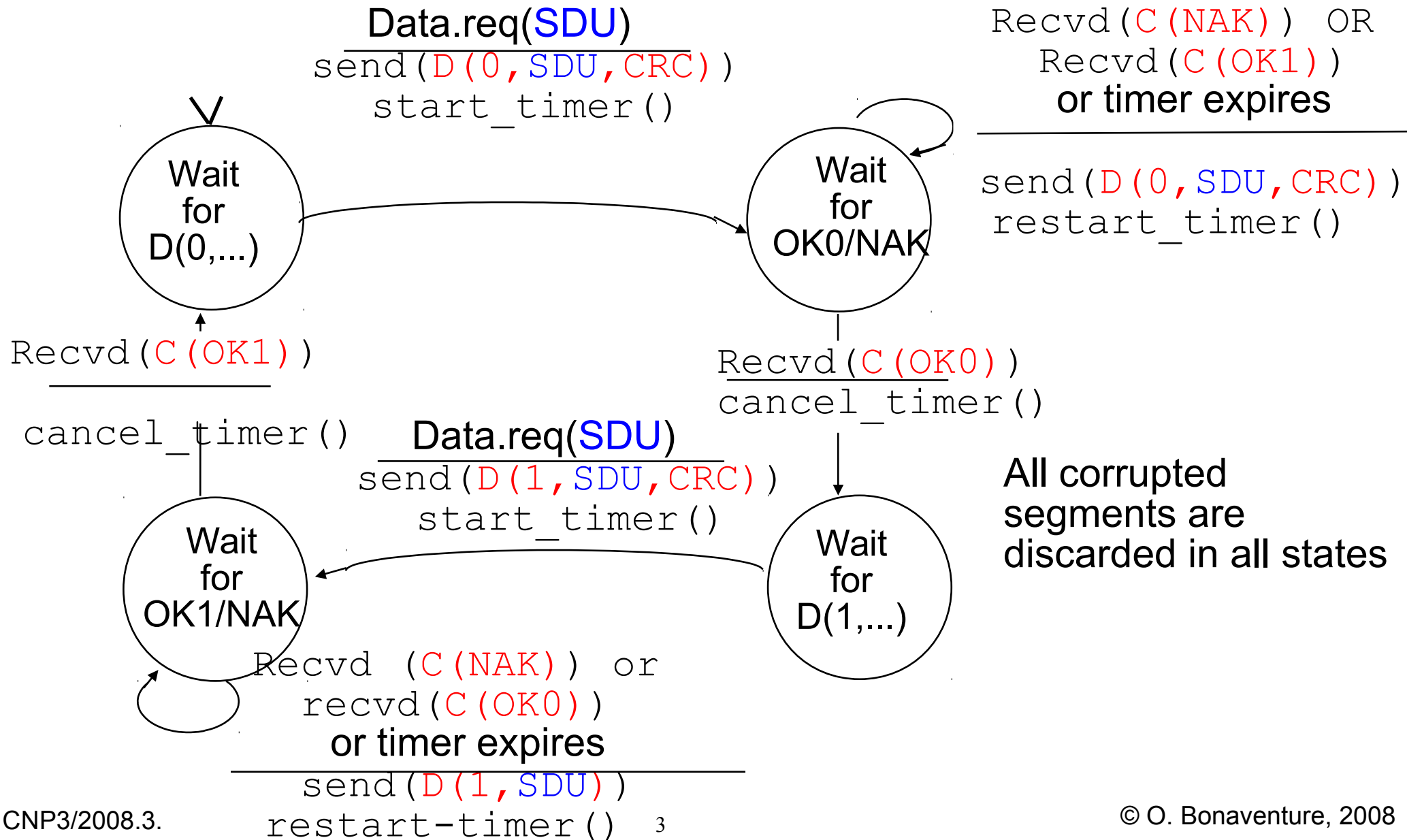
- Contents of each segment
 - Data segments
 - Control segments



Alternating bit protocol : Sender



Alternating bit protocol : Sender + CRC



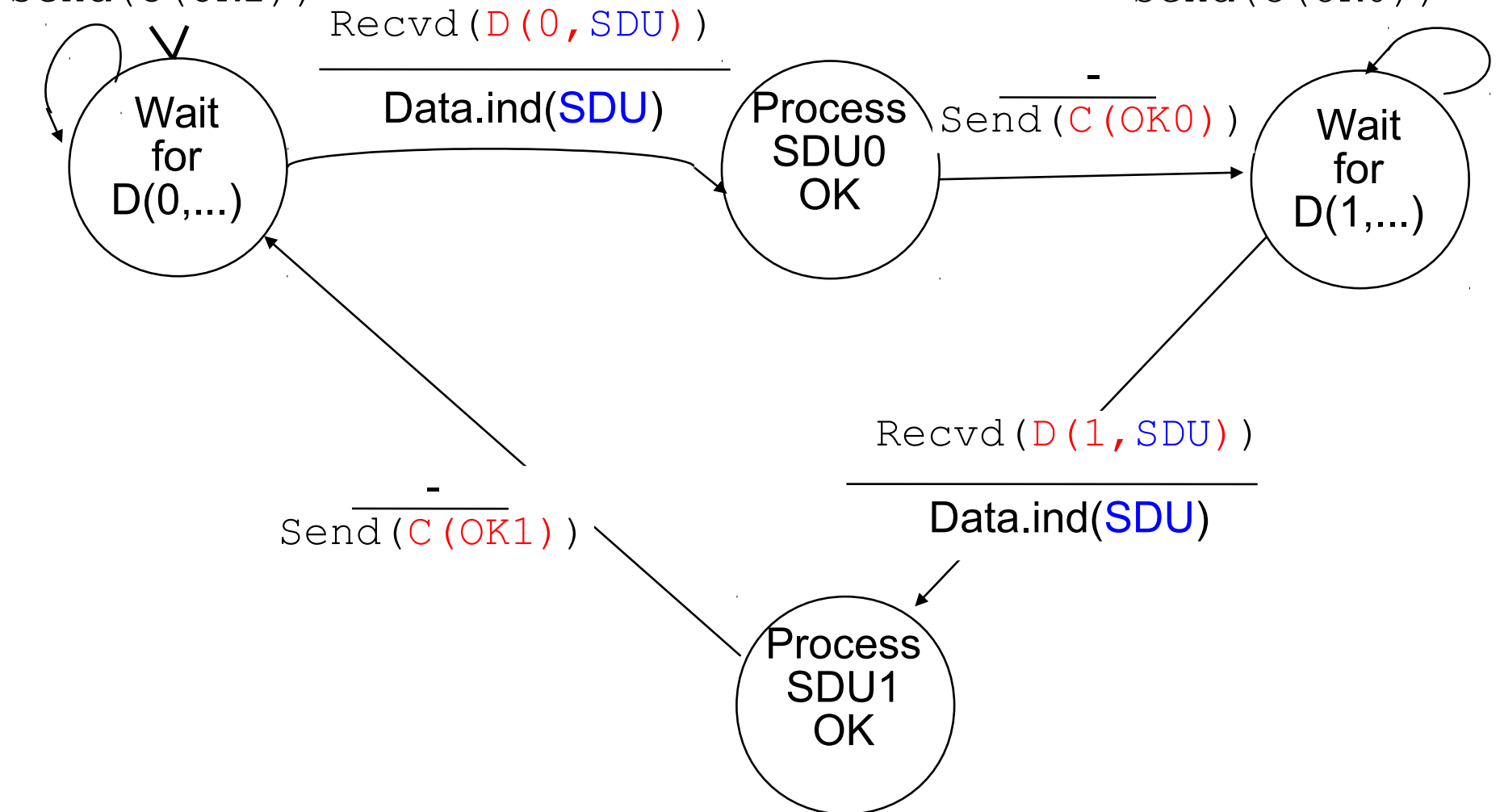
Alternating bit protocol : Receiver

Recvd (D (1, SDU))

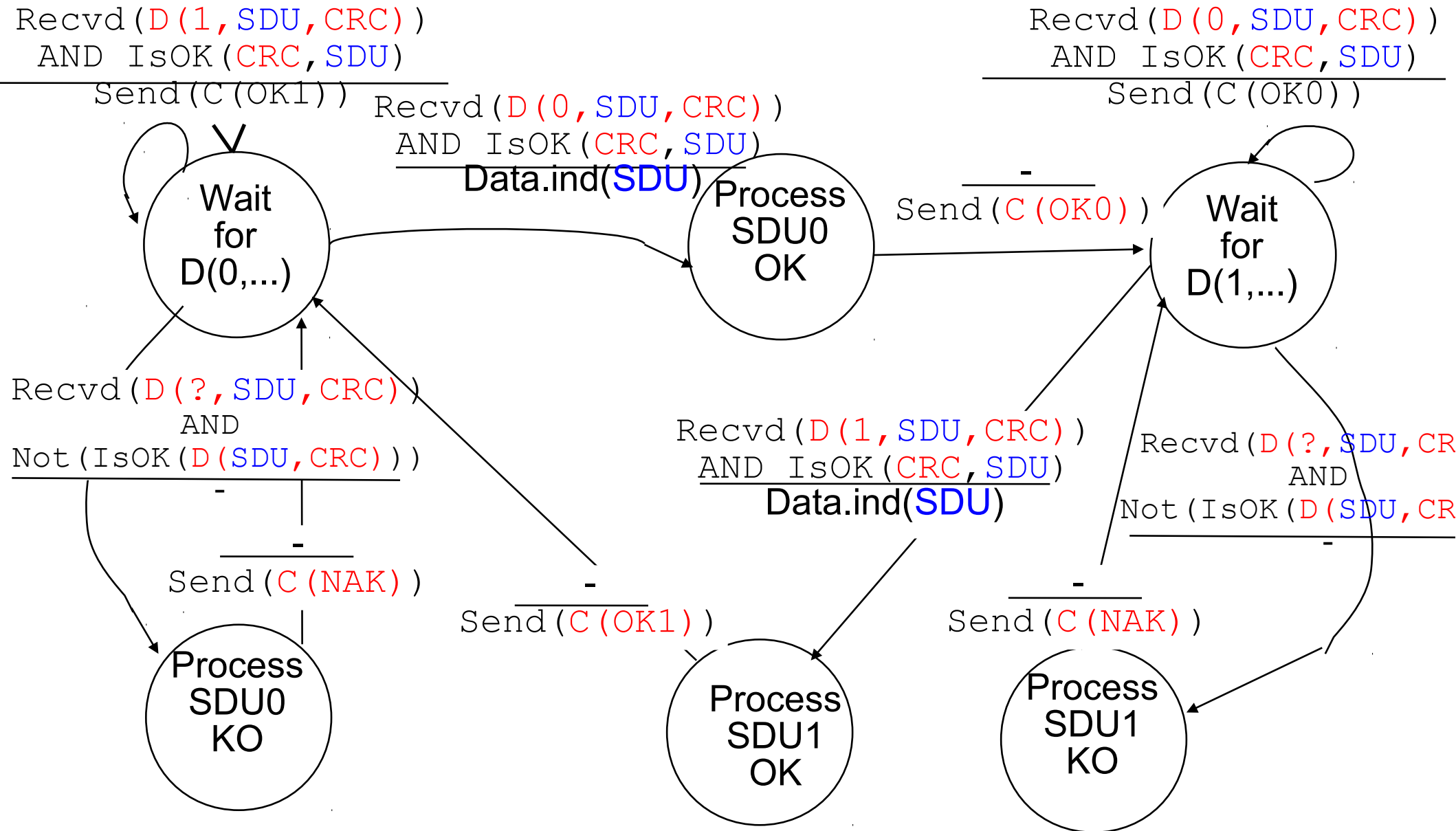
Recvd (D (0, SDU))

Send (C (OK1))

Send (C (OK0))



Alternating bit protocol : Receiver + CRC



A bit of Practice

Amazon provides the S3 storage service where companies and researchers can store lots of information and perform computations on the stored information. Amazon allows users to send files through the Internet, but also by sending hard-disks.

Assume that a 1 Terabyte hard-disk can be delivered within 24 hours to Amazon by courier service.

Assume that hard-disks are sent over a perfect network layer service where the only constraints are :

- The maximum segment size is 1500 bytes,
- The network layer header is 40 bytes (IPv6) or 20 bytes (IPv4).

What is the minimum bandwidth required of this service to match the bandwidth of the courier service over IPv6 ? Over IPv4 ?

What is the goodput of the service ?

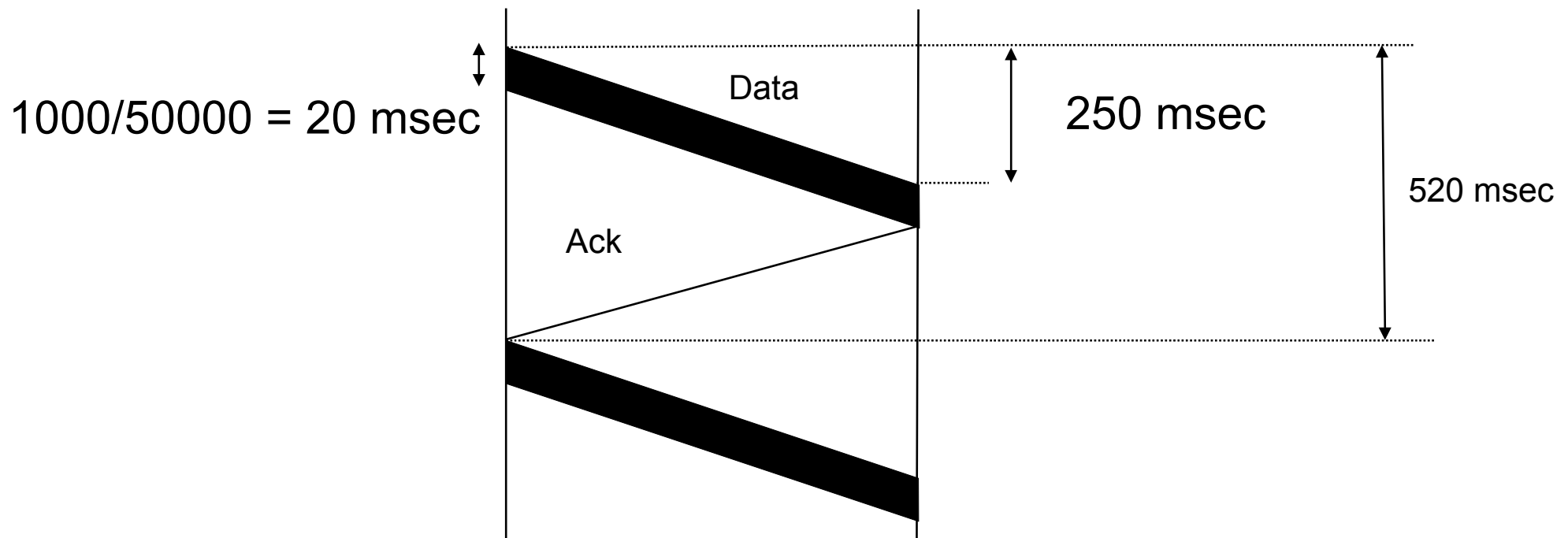
A bit of practice

Several large datacenters operators have announced that they install servers as containers with each container hosting up to 2000 servers. Assuming a container with 2000 servers and each storing 500 GBytes of data, what is the time required to move all the data stored in one container over one 10 Gbps link ?

What is the bandwidth of a truck that needs 10 hours to move one container from one datacenter to another.

Performance of the alternating bit protocol

- What is the performance in **kb/s** of ABP in this case :
 - One-way delay : 250 msec
 - Physical layer throughput : 50 kb/s
 - segment size : 1000 bits



I -> Performance is function of

*bandwidth₈ * round-trip-time*

How to improve the alternating bit protocol ?

- Use a pipeline
- Principle
 - The sender should be allowed to send more than one segment while waiting for an acknowledgement from the receiver

