Communication in Electronic Systems

Lecture 6: Simple Multiuser Systems

Lecturer: Petar Popovski

TA: Junya Shiraishi, João H. Inacio de Souza

email: petarp@.es.aau.dk

AALBORG UNIVERSITY
DENMARK

Connectivity

Course Overview: Part 2. Communication and Networking

- MM5: Introduction to Communication Systems
- **MM6: Simple Multiuser Systems**
- MM7: Layered System Design. Network Topology and Architecture
- MM8: Networking and Transport Layers
- MM9: Introduction to Security
- MM10: Packets and Digital Modulation
- MM11: Communication waveforms
- MM12: Workshop on modulation and link operation

outline

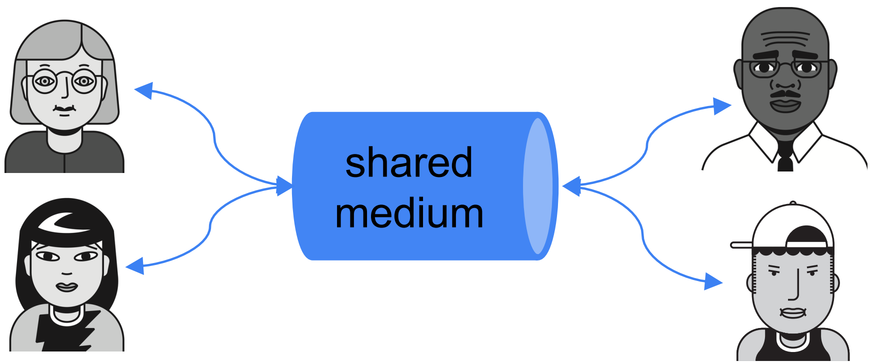
- multiuser communication over a shared medium
- medium access methods
  - scheduling, token based, round-robin, random access
- example: CANbus system

multiuser communication over a shared medium

multiuser communication over a shared medium (1)

- last lecture: point-to-point communication

- shared communication medium
  - wireless, shared ethernet cable

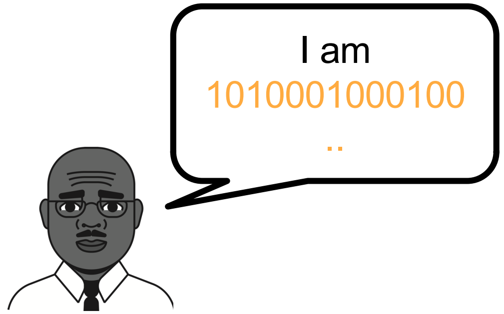


multiuser communication over a shared medium (2)

how to deal with a shared communication medium?

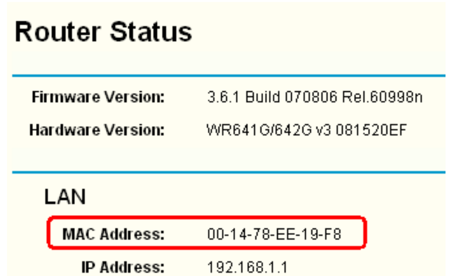
first idea: introduce source/destination addresses

- medium access control (MAC) address



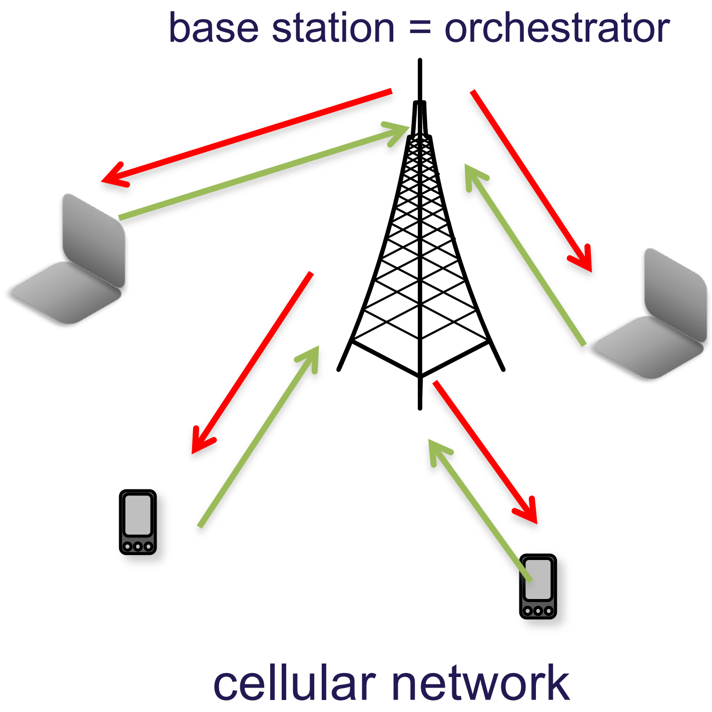
why should we introduce addresses?

- MAC provides flow control and multiplexing for the transmission medium
- MAC controls the way in which a user is making use of the portion of the shared medium



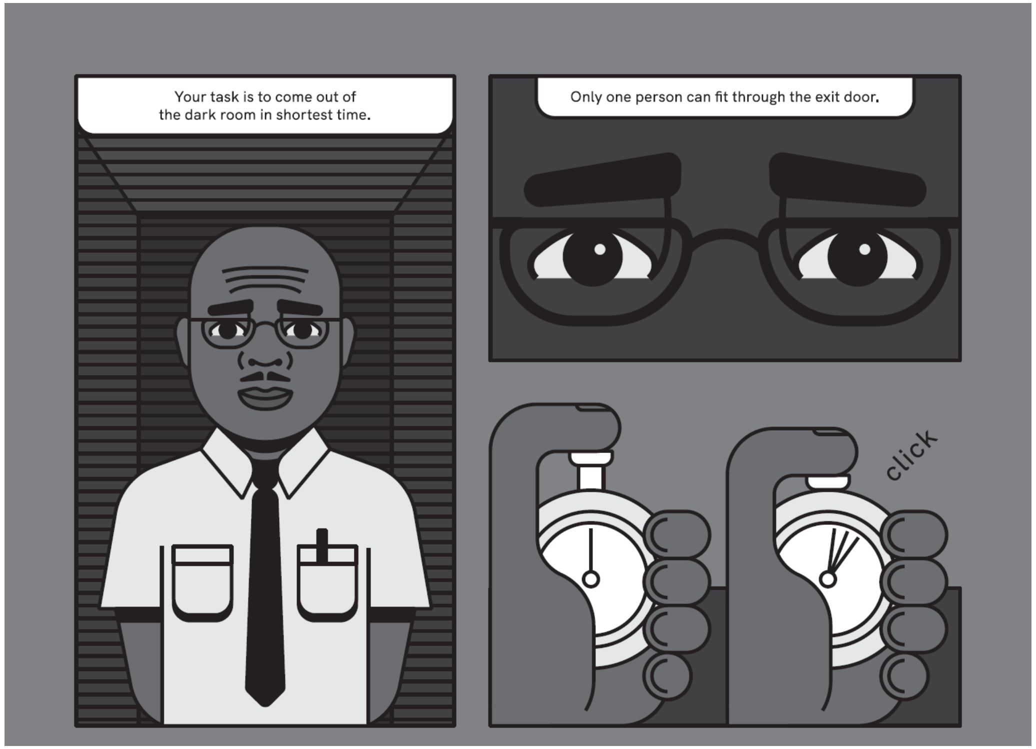
multiuser communication over a shared medium (3)

- downlink transmissions
  - no collisions
- uplink transmissions
  - collisions if no coordination



- Q: how do we deal with the collisions?

the dark room metaphor



1. if the room is not dark, why is the coordination easy?
2. does the coordinator know the names of the people inside the room?
3. is there any other coordination among the people in the room?

medium access methods

1. all users are known?
2. is there any coordination among users?

yes

no

scheduling (reserving)

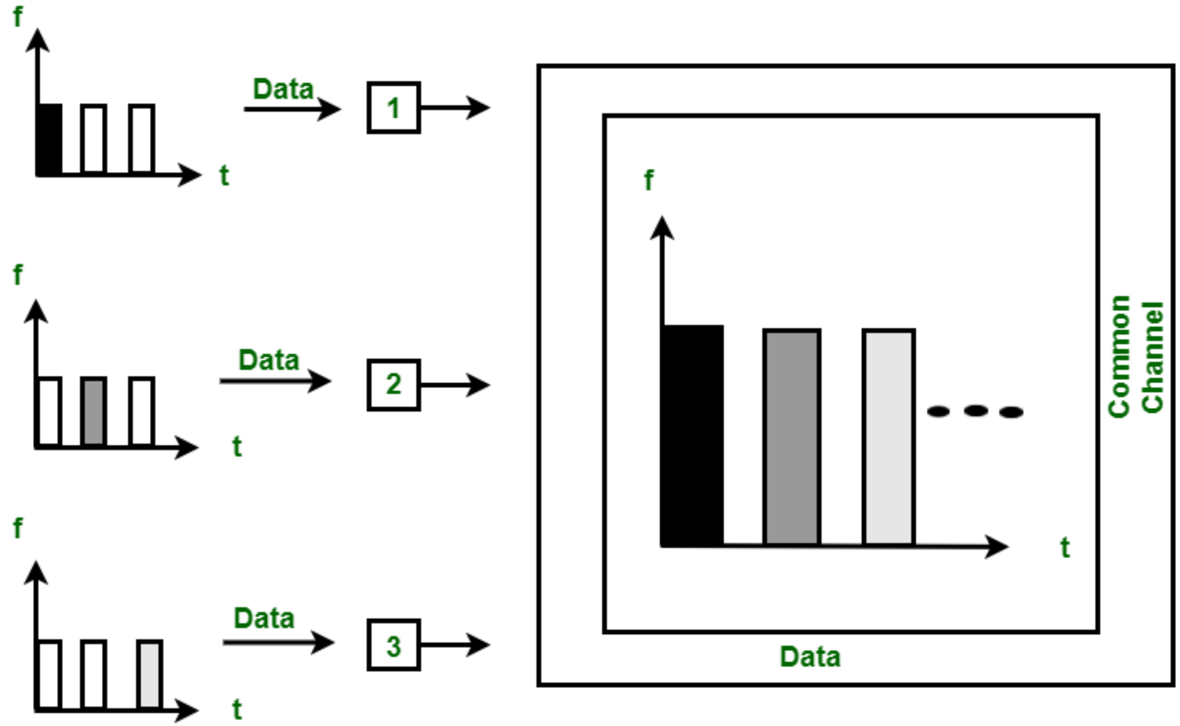
- scheduling: TDMA, FDMA
- token based, round-robin

free-for-all

- random access
  - ALOHA
  - slotted ALOHA
  - carrier sensing

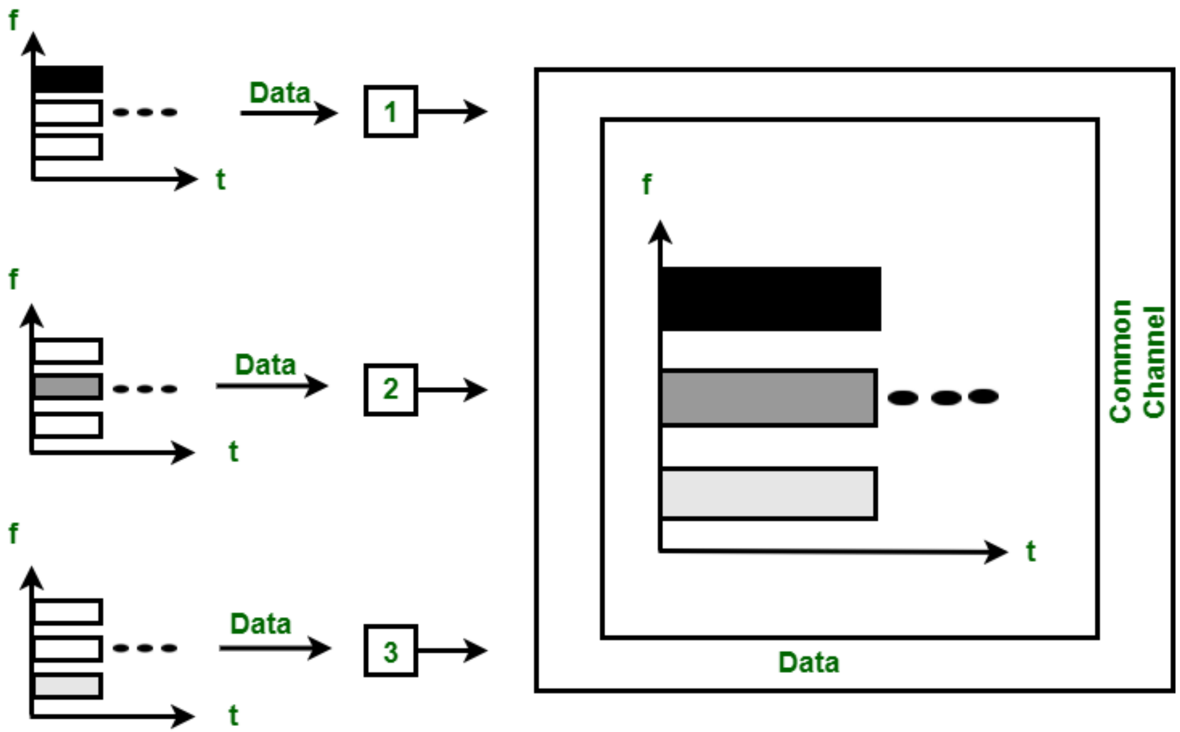
scheduling (1)

- TDMA
  - time-division multiple access



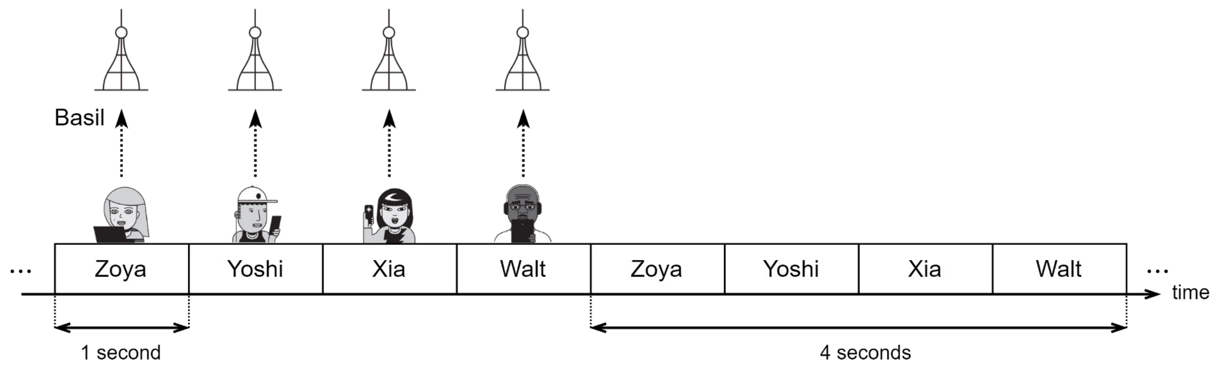
scheduling (2)

- FDMA
  - frequency-division multiple access



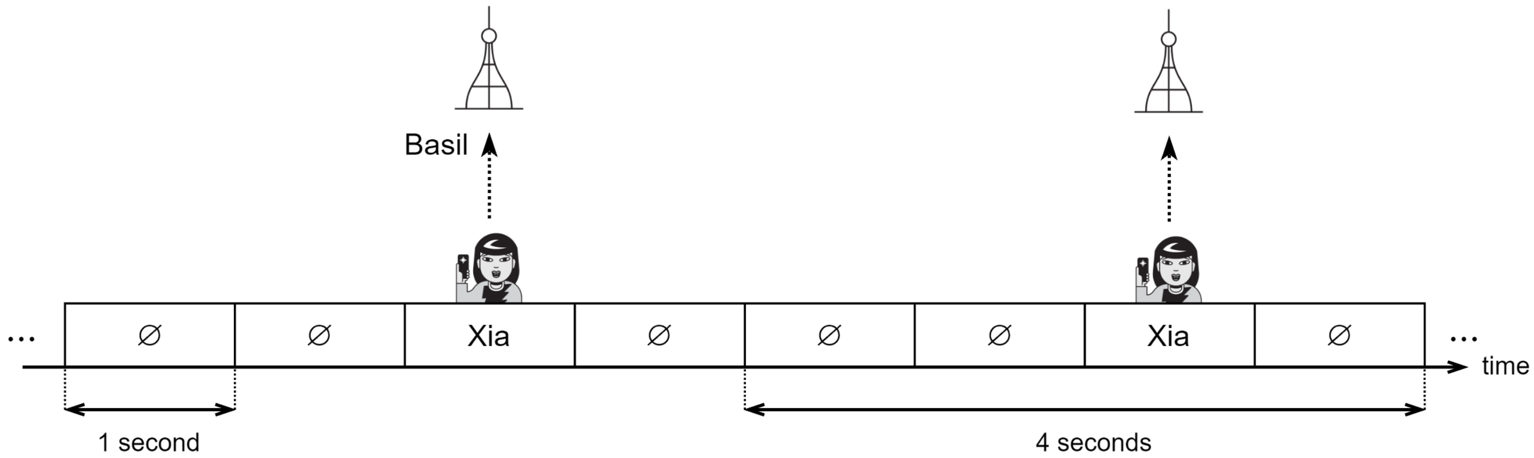
TDMA with periodic reservation (1)

- 4 users send packets to Basil (uplink)
  - packet rate $R = 1 \text{ kbit/s}$
  - one packet contains $1{,}000 \text{ bits}$
- Zoya's effective data rate
  - 2 frames (8 s) to transmit $2{,}000 \text{ bits}$
  - $\frac{2{,}000}{8} = 0.25 \text{ kbit/s} < 1 \text{ kbit/s}$
- system throughput
  - $G = \frac{4{,}000}{4} = 1 \text{ kbit/s}$



TDMA with periodic reservation (2)

- the network traffic has changed
  - only Xia had data packets to send during 10 frames
  - system throughput: $G = \frac{10{,}000}{80} = \color{red}{0.25\ \mathrm{kbit/s}} < 1\ \mathrm{kbit/s}$

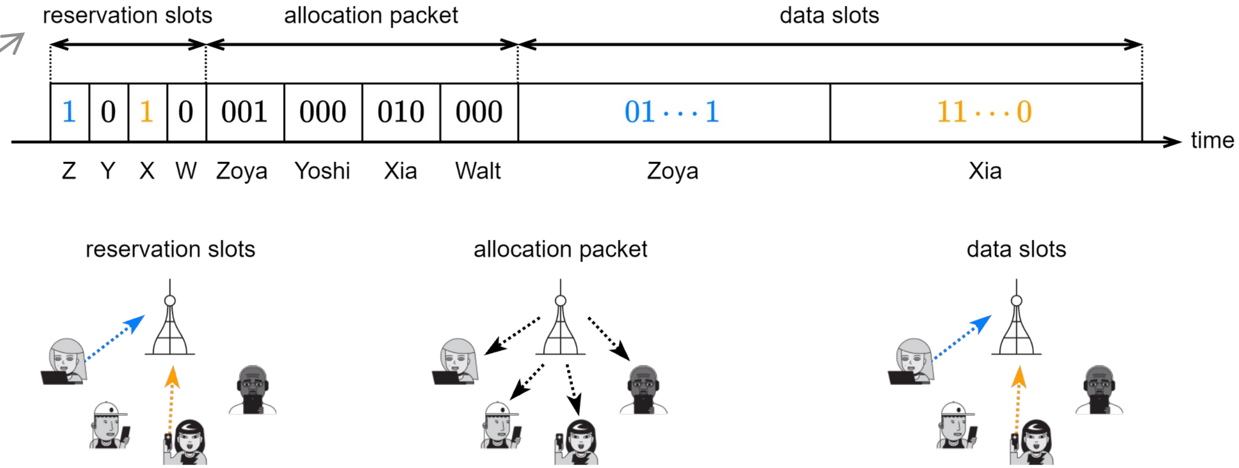


dynamic reservation and the cost of overhead (1)

- 4 users send packets to Basil (uplink)
  - reservation slots (users)
  - allocation packet (Basil)
  - data slots (users)

allocation packet	
000	No data slot reserved
001	1st data slot reserved
010	2nd data slot reserved
	...

reservation slots	
0	User has no packet to transmit
1	User has a packet to transmit

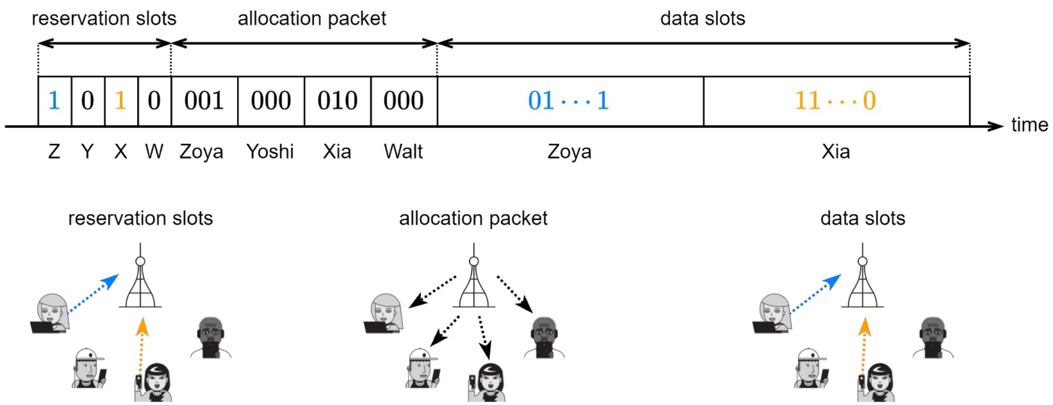


dynamic reservation and the cost of overhead (2)

- system throughput
  - 4 users transmitting / data packet: $1{,}000$ bits / $R = 1$ kbit/s
  - overhead: 16 bits or $\frac{16}{1{,}000} = 0.016$ s
  - throughput: $G = \frac{4{,}000}{0.016+4} = 0.996$ kbit/s ≈ 1 kbit/s

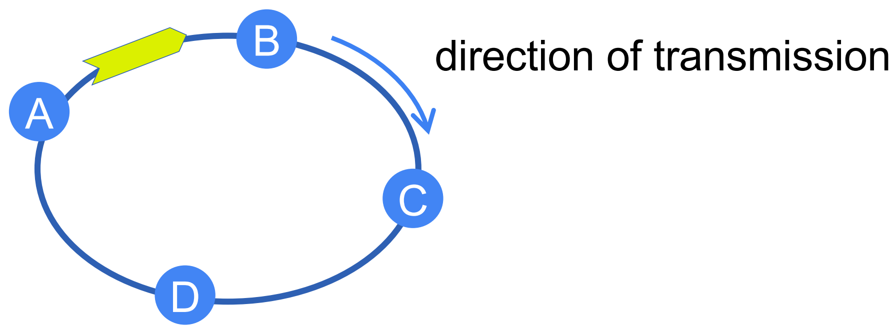
- 1 user transmitting
  - throughput: $G = \frac{1{,}000}{0.016+1} = 0.984$ kbit/s

- 1 sensor transmitting
  - data packet: 4 bits
  - throughput: $G = \frac{4}{0.016+0.004} = 0.2$ kbit/s

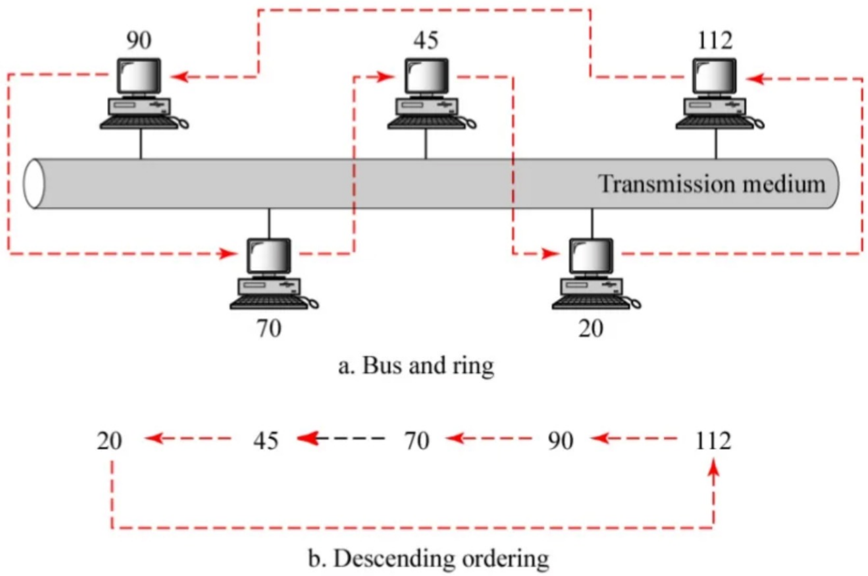


token based (1)

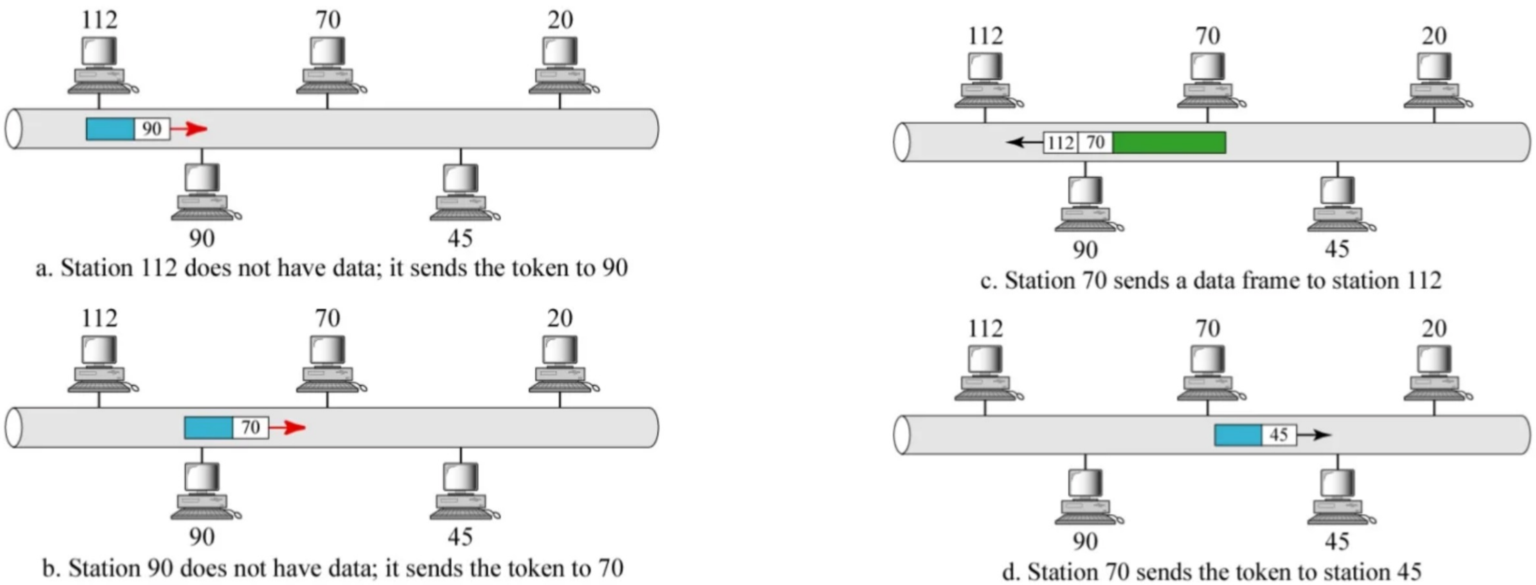
- control token passed from one node to next sequentially
- point-to-point links can be fast
- problems:
  - token overhead
  - latency
  - single point of failure (token)



token based (2)



token based (3)



token based (4)

- under light load – delay is added due to waiting for the token
- under heavy load – ring is “round robin”

advantage:

a) fair access

disadvantages:

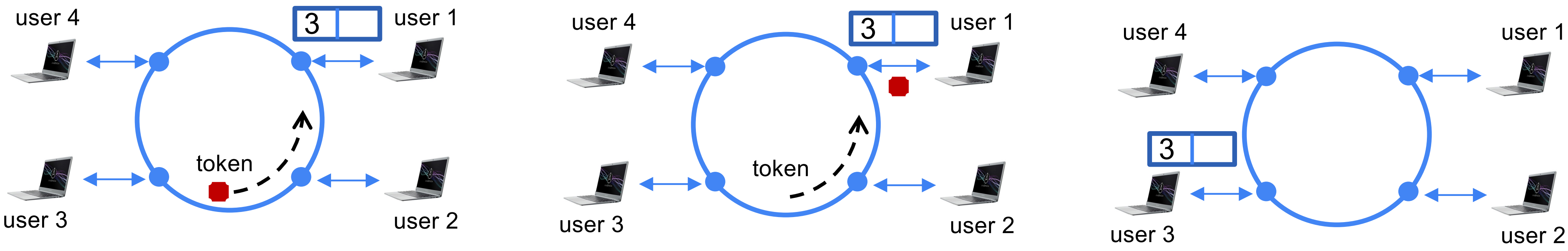
a) ring is sensitive to points of failure

b) added issues due to token maintenance

round-robin

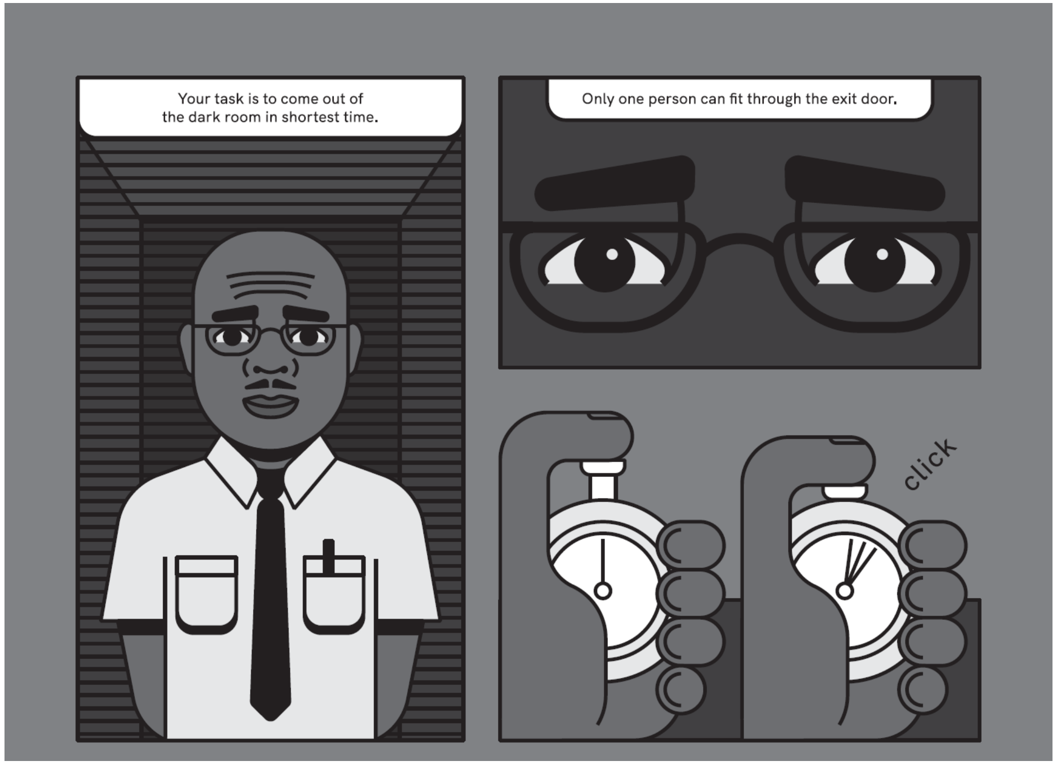
- comes from the round-robin principle
  where each person gets an equal share of something in turns.
- each ready task runs turn by turn only in a cyclic queue for a limited time slice.
  - this algorithm also offers starvation-free execution of processes
- characteristics
  - one of the oldest, fairest, and easiest algorithm
  - round-robin is a pre-emptive algorithm
  - bring the best performance in terms of average response time
  - do not give special priority to more important tasks

round-robin: example



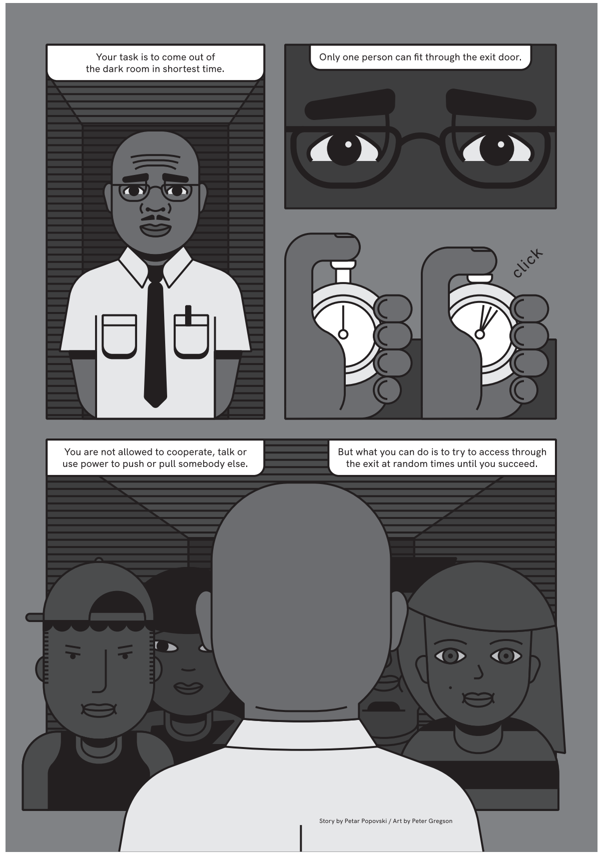
Assume that user 1 needs to transmit a frame to user 3. First, user 1 captures the token and transmits the frame, which circulates through the ring. Since the destination is user 3, user 4 examines the frame and then passes it to the next user. When the frame reaches user 3, user 3 copies the frame. The frame continues the circulation through the ring until user 1 removes the frame and releases a new token.

random access (1)

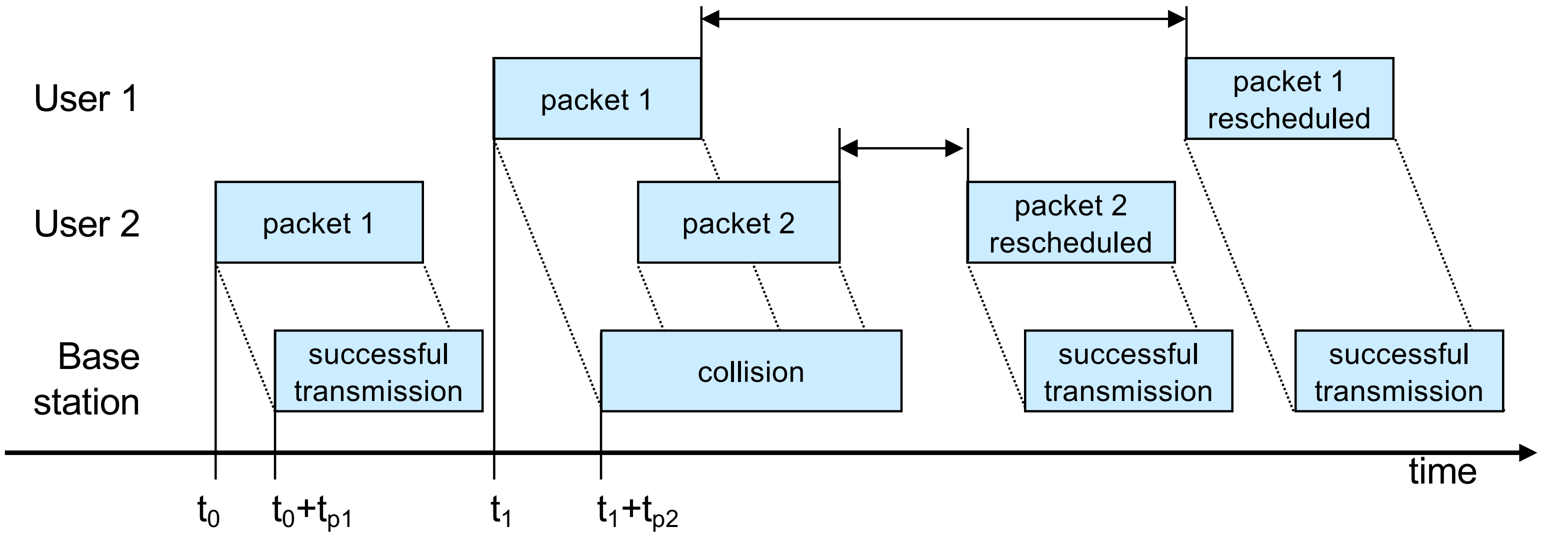


random access (2)

- direct transmission of data messages without prior signaling
- contention/ collisions
- suitable for short messages

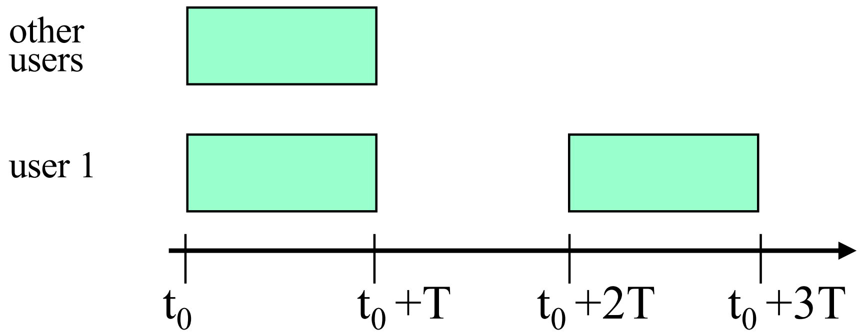


ALOHA



- transmit messages immediately – use entire bandwidth
- in case of collision – delay retransmission for a **random** time interval
- special case of ”stop-and-wait” ARQ (Automatic Repeat Request)

slotted ALOHA



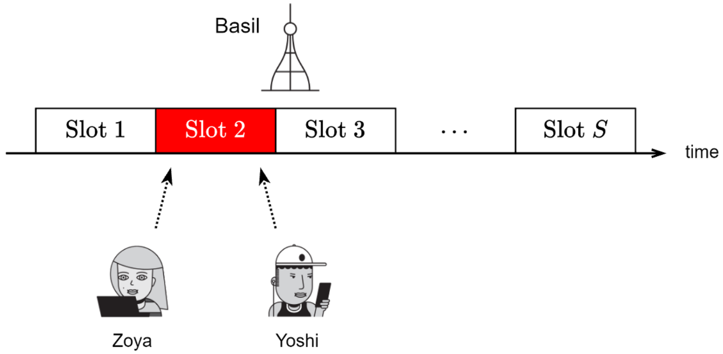
vulnerable period: *T*

throughput

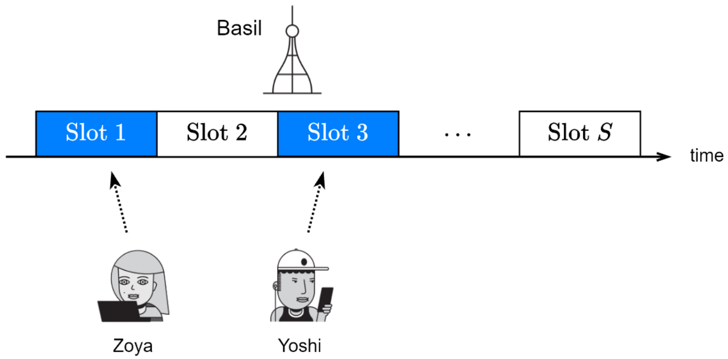
$$S = \lambda T \exp(-\lambda T) = G \exp(-G)$$

framed ALOHA (1)

- 2 users sends packets to Basil
  - S data slots are available
  - Zoya and Yoshi pick a single slot with probability $\frac{1}{S}$



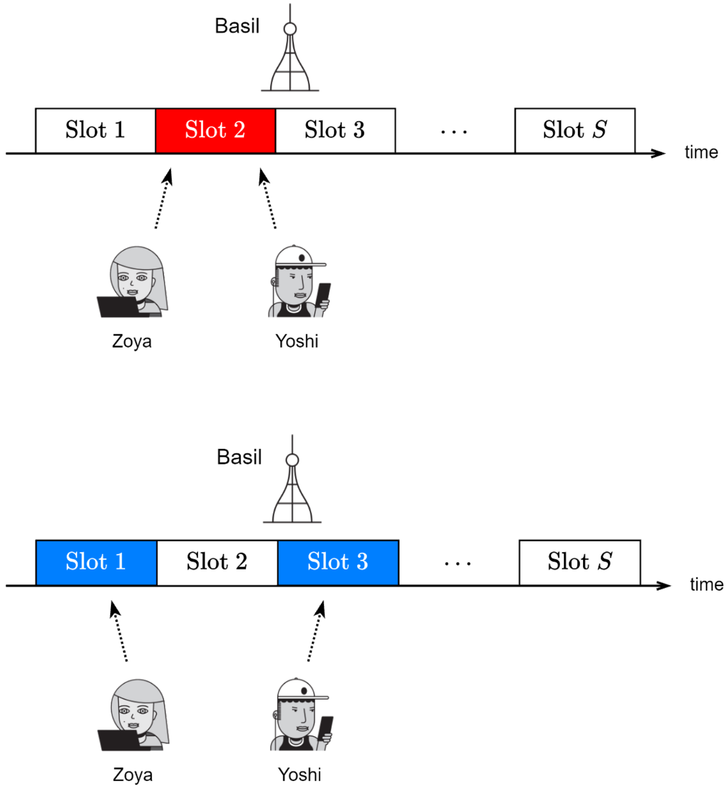
- probability of successful transmission in a packet
  - $P(S) = \frac{1}{S}\left(1 - \frac{1}{S}\right) + \left(1 - \frac{1}{S}\right)\frac{1}{S} = \frac{2}{S}\left(1 - \frac{1}{S}\right)$
  - optimal number of slots: $S^* = 2$
  - $P(S = 2) = \frac{1}{2}$



framed ALOHA (2)

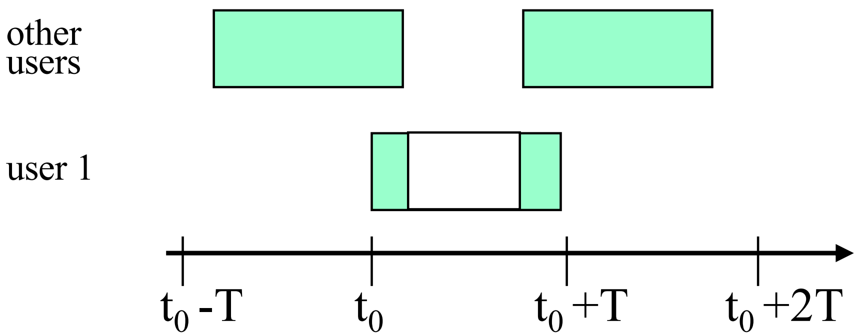
- generalization for K users
  - $P(S) = \frac{K}{S}\left(1 - \frac{1}{S}\right)^{K-1}$
  - optimal number of slots: $S^* = K$
  - $P(S = K) = \left(1 - \frac{1}{K}\right)^{K-1}$

- lower bound for the probability
  - $\lim_{K\to\infty} \left(1 - \frac{1}{K}\right)^{K-1} = e^{-1}$

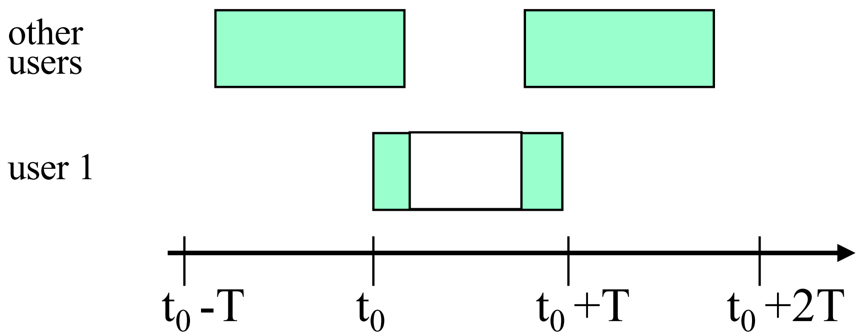


ALOHA analysis (1)

- packet arrival process is modelled as a Poisson process with average arrival rate λ
- λ - arrival rate of new and retransmitted packets



ALOHA analysis (2)



$$\Pr[n] = \frac{(d\lambda)^n}{n!} \exp(-d\lambda)$$

- vulnerable period: *(t-T, t+T)*
- probability that no packet starts *T* seconds before and *T* seconds after the start time of a given packet

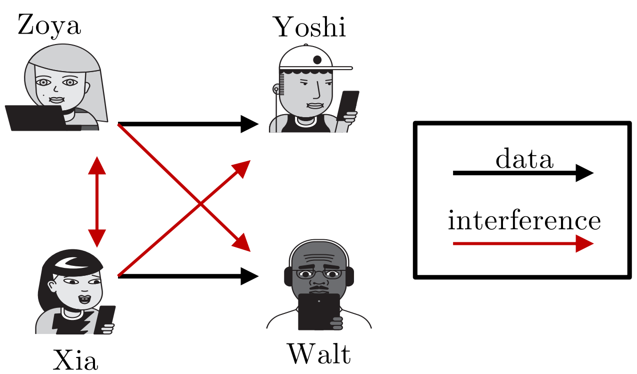
$$\Pr[success] = \Pr[0] = \exp(-2T\lambda)$$

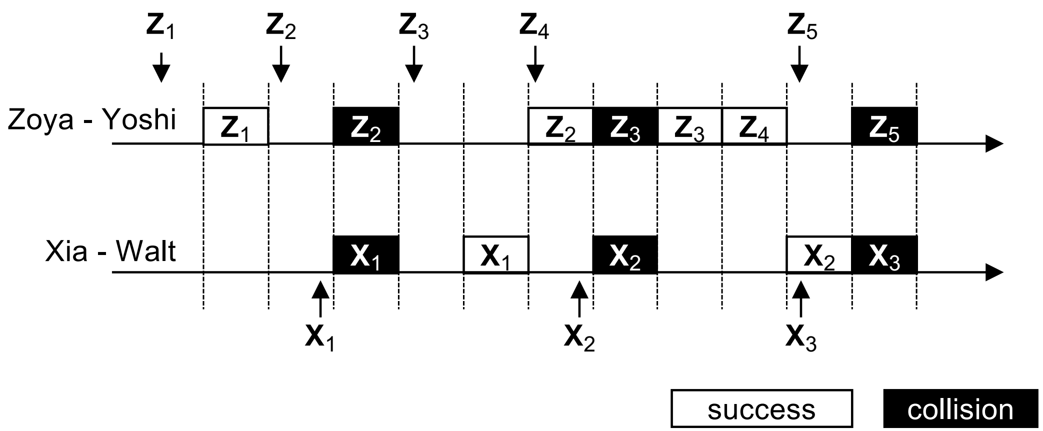
- the throughput is

$$S = \lambda T \exp(-2\lambda T) = G \exp(-2G)$$

carrier sensing (1)

- spectrum sharing with a slotted synchronized structure
  - questionable to assume that two independent systems are a a priori synchronized, but synchronization can occur through the access of a shared medium
  - new packets **always need to wait** at least until the start of a new slot
  - enforced **waiting synchronizes the packets to collide**





carrier sensing (2)

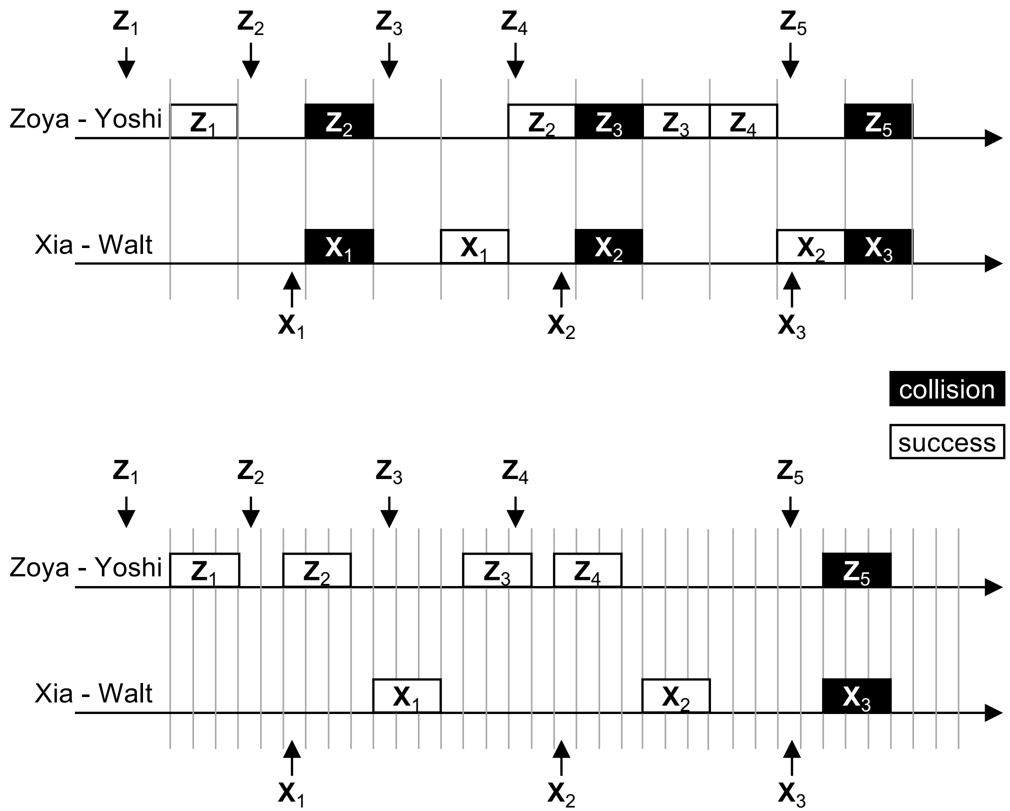
Idea: Make the *idle slots* short (cheap!)

Finer time resolution
preserves asynchronism of arrivals

Devices not currently transmitting
must listen to **sense** if carrier is free

- If the medium is busy,
  device waits for a random time
  =
  countdowns
  a random number of idle minislots

Carrier sensing obviates the need
for fixed-length packet



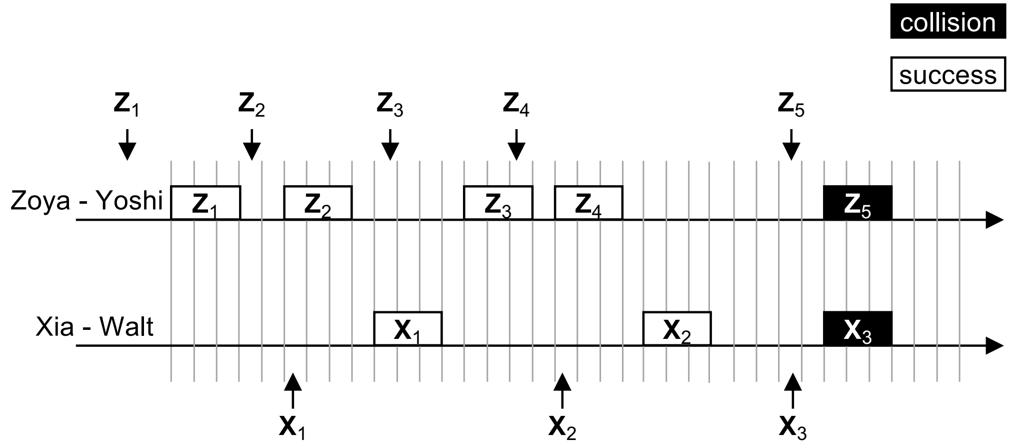
carrier sensing (3)

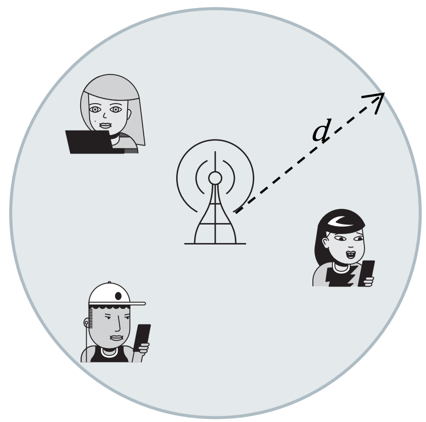
Random number of waiting minislots

- Pausing countdown during sensed carriers to avoid aligning of the access instants after the medium is sensed idle

Choosing the idle slot size

- Carrier should be reliably detectable
- Propagation delay $T_I = \frac{d}{c}$ or $T_I = \frac{2d}{c}$ if the sensing range is larger than the communication range





example
CANbus system

multiuser features of CANbus system (1)

- controller area network (CAN) is a method of serial communication,
  which supports distributed real-time control with a very high level of data integrity
  - used in cars and industry

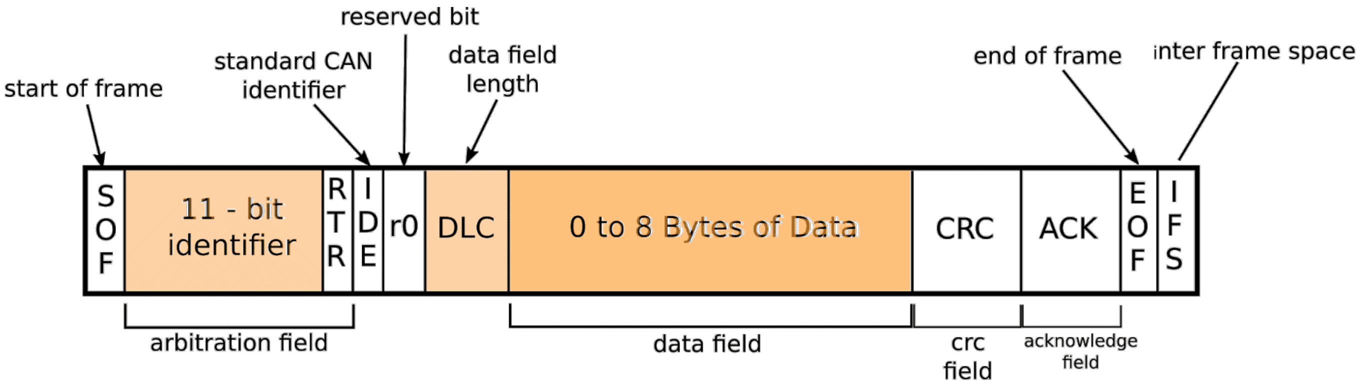
- the bus reduces the wiring connections and
  the overall complexity of the system by connecting all nodes to one bus

multiuser features of CANbus system (2)

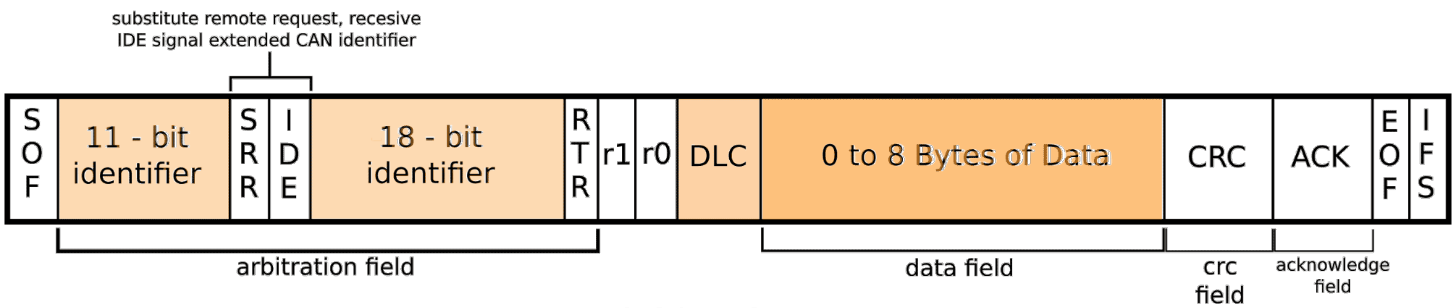
- protocol features
  - error frame handling, when wrong frames are received
  - content based addressing, not message based in the "arbitration ID"
  - priority scheme based on "arbitration ID"
    - lower value indicates higher priority
    - ensures low / deterministic latency for critical messages
      - critical messages like brakes, and fuel input are sent/received quickly
  - supports different data rates

CAN types	low speed	high speed	flexible data rate
max Distance	500 meter	40 meters	10 meters
max data rate	125 kb/s	1 Mb/s	15 Mb/s

multiuser features of CANbus system (3)



Standard CAN data message

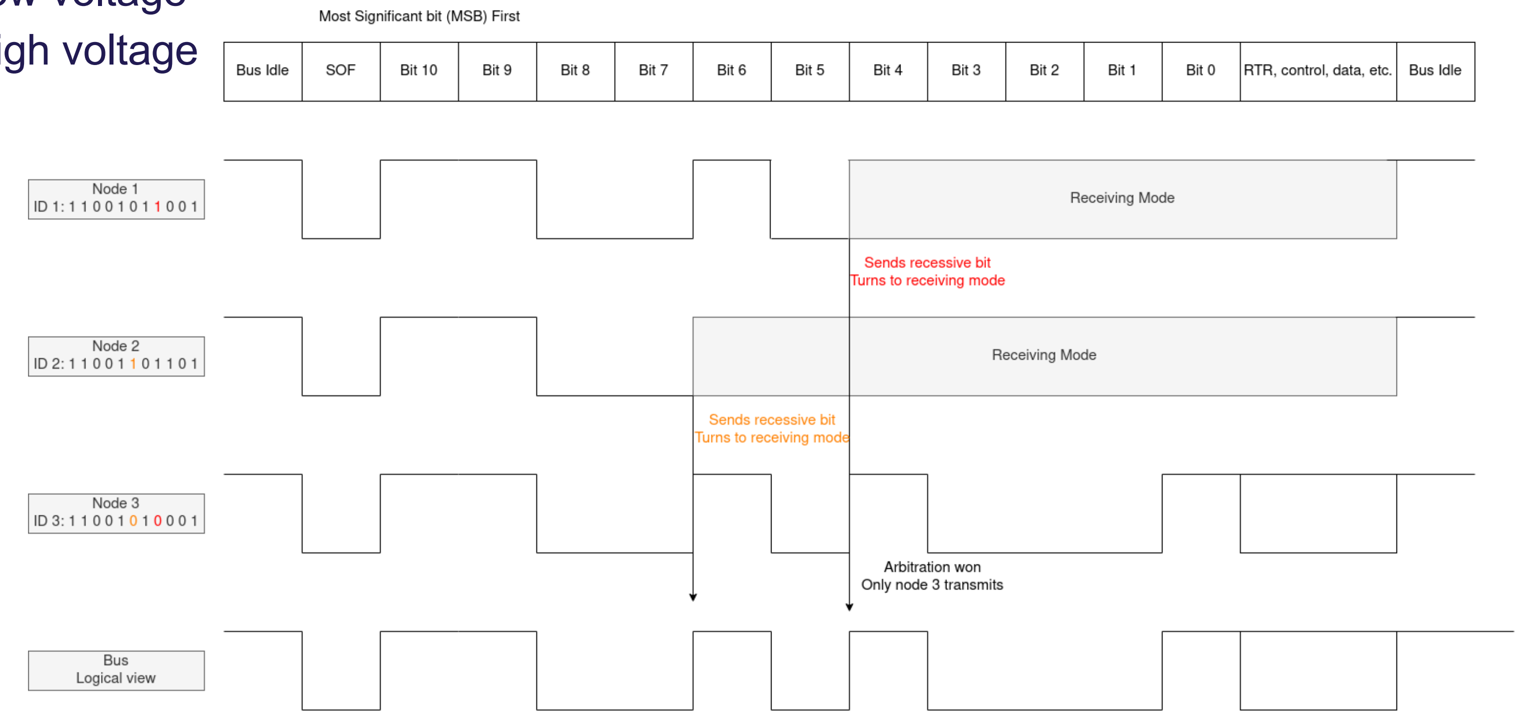


Extended CAN data message

CAN Bus message frame

multiuser features of CANbus system (4)

- arbitration competition occurs when multiple nodes wants to transmit concurrently
  - 1 bit recessive – low voltage
  - 0 bit dominant – high voltage
- example
  - logical view



summary

- we have introduced the basic problem of

  sharing the communication medium among multiple users
- medium access methods
  - scheduling, token based, round-robin, random access
- a simple probabilistic analysis of random access systems
- practical system example: CANbus system