4. Medium Access Control

Data link layer protocols, medium access control

Wireless sensor networks

Martin Úbl
ublm@kiv.zcu.cz

2023/24

<DCSE> DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Medium Access Control

- ISO/OSI Layer 2, MAC sub-layer
- interfacing with physical layer
  - we established neighborhood
  - determining, when and for how long we can transfer
  - waking up other stations
- interfacing with network layer
  - waking up the upper stack (upon wakeup request)
  - queuing the application (network/transport) data until the next transmission period
- medium access methods

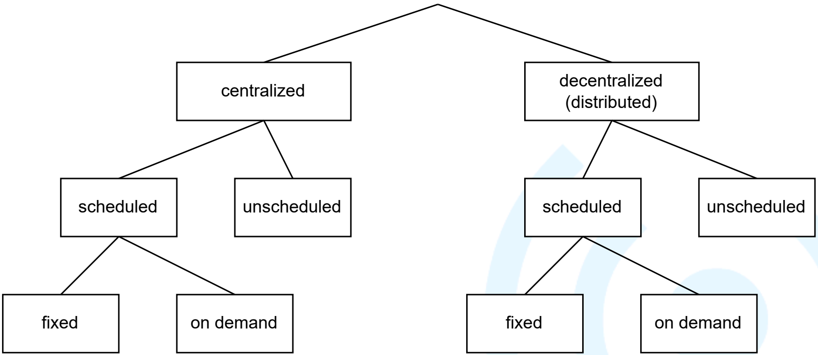
Medium Access Control

Problems

- goals:
  - minimize power consumption
  - minimize delay (depending on application)
  - minimize errors
    - more errors → more retransmissions → higher power consumption
- for now, we will focus on minimizing the power consumption in general
- main problems:
  - superfluous active time (e.g., excessive listening)
  - collisions
  - reception of irrelevant traffic
  - excessive control overhead traffic

Medium Access Control

Categorization



Medium Access Control

Design decisions

- parameters:
  - period of activity (wake up time)
  - period of sleeping (sleep time)
- goal is to minimize wake up time and maximize the sleep time
  - all of it with preserving the maximum transfer effectivity
- two large groups:
  - **synchronous**
  - **asynchronous**

Medium Access Control

Design decisions

- **synchronous**
  - synchronized wake-up plan
  - nodes know, when to wake up to transmit data to a specific node
  - either pre-configured or learned from network operation
- **asynchronous**
  - nodes wake up on their own convenience
  - no synchronization of wake ups
  - nodes synchronize on-demand, when required
    - most likely prior to the data transmission
- at first, let us consider *asynchronous*, as they resemble known scenarios more

Medium Access Control

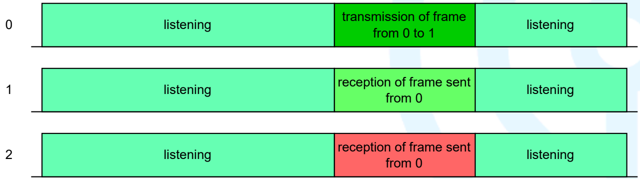
Design decisions

- *asynchronous* protocols
  - we have to find a "rendezvous" point in sender and receiver duty cycles
  - on-demand synchronization
  - **receiver-initiated**
    - receiver periodically transmits a beacon frame
    - indicates, that it is ready to receive transmissions
    - sender waits for the beacon frame and then transmits
  - **sender-initiated**
    - sender asks the receiver for permission to transmit data
    - receiver acknowledges the permission
    - sender then sends the data

Medium Access Control

Naive approach

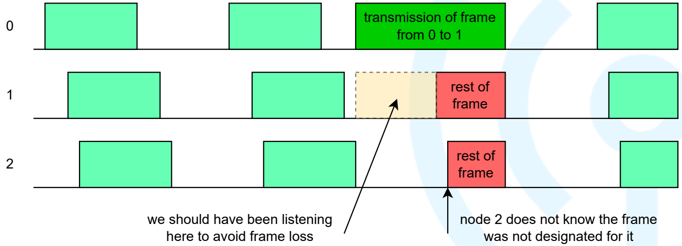
- let us take most naive approach as an example
- wake up time power consumption: 70 mA
- sleep time power consumption: 0.05 mA
- let's ignore the other power consumers for now
- MAC protocol:
  - woken up all the time
  - listening for all frames, receive everything
- average power consumption: 70 mA



Medium Access Control

Naive approach

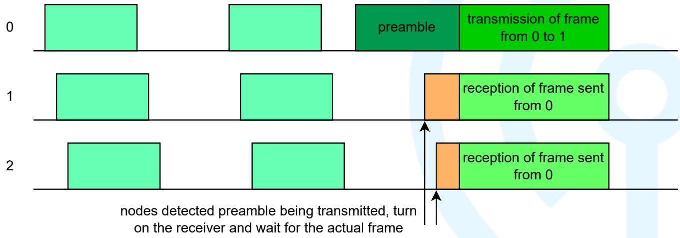
- change the protocol:
  - introduce sleep periods on a regular basis (sleep 50 % of the time)
- problem: what if we wake up to the middle of the frame?
  - the frame is lost and must be retransmitted
- average power consumption: 35 mA
- BUT: some frames are lost!



Medium Access Control

(Not so) naive approach

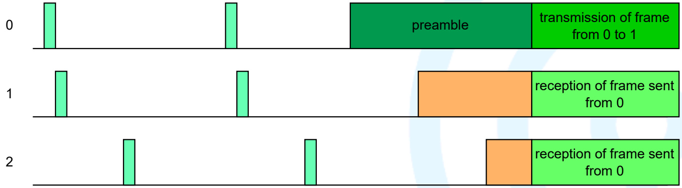
- change the protocol:
  - introduce the *preamble*
    - e.g., a sequence of alternating ones and zeroes
    - keeps the receiver in wake up state
    - preamble is longer, than the sleep time
  - receiver stays in wake up state when detecting preamble
- average power consumption: 35 mA
- frames no longer get lost



Medium Access Control

(Not so) naive approach

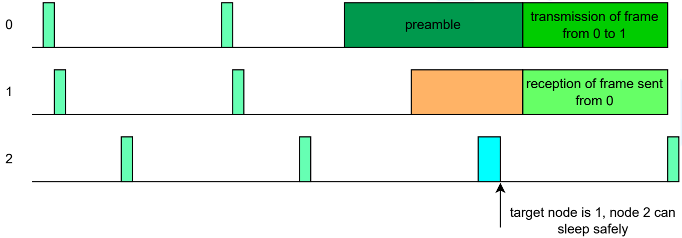
- change the protocol:
  - having the preamble, we can shorten the active listening
  - let us sleep for 95 % of time
- average power consumption: 3 mA
- nodes still "overhear" frames not designated to them



Medium Access Control

(Not so) naive approach

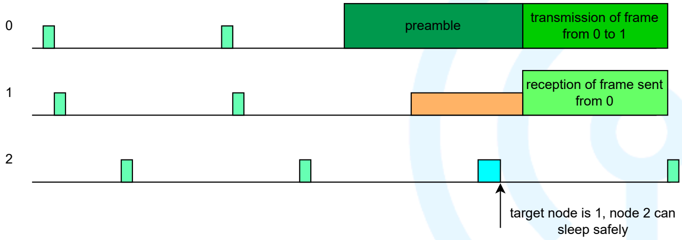
- change the protocol:
  - transmit target address in preamble
    - we have to change the preamble structure to a *strobed* preamble – more on this later
  - allows the receiver to ignore the transmission and reduce active time
- average power consumption: 2.8 mA



Medium Access Control

(Not so) naive approach

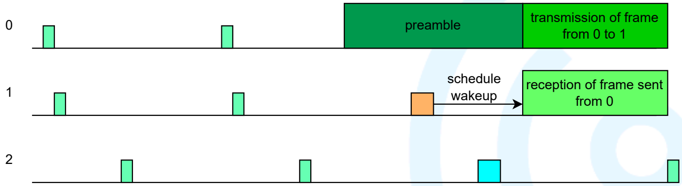
- change the protocol:
  - different physical layer for preamble listening – introduce *low-power listening* (LPL)
    - LPL power consumption: 15 mA (example)
  - LPL radio wakes up the primary radio
- average power consumption: 1 mA



Medium Access Control

(Not so) naive approach

- change the protocol:
  - change the preamble to include time to preamble end
  - receiver (LPL radio) can schedule the wakeup
- average power consumption: 0.8 mA



Medium Access Control

Improving the approach

- we pretty much defined whole scale of practically used MAC protocols
- there are more caveats we will discuss later
- there are some principles we need to discuss first:
  - multiple access to medium
  - preamble
  - low-power listening (LPL)
  - preamble sampling

Medium Access Control

Multiple access

- CSMA – Carrier Sense Multiple Access
- the radio senses the carrier wave (Carrier Sense)
  - before transmission (to avoid collisions)
  - periodically to receive frames
- the radio assumes there are multiple nodes that share a single medium (Multiple Access)
- many variants
  - we know several ones from before
  - CSMA/CD – Collision Detection (e.g., Ethernet)
  - CSMA/CA – Collision Avoidance (e.g., 802.11 / WiFi)

Medium Access Control

Preamble

- preamble
- a pre-defined sequence of ones and zeroes
- has many purposes, we will focus on preambles in WSN to lower power consumption
- optionally has:
  - recipient address
  - time to preamble end
- usually transmitted in reduced mode, e.g.:
  - lower power
  - different modulation (OOK, ...)
  - lower bitrate

Medium Access Control

Preamble

- length of the preamble is protocol-specific
- content is also protocol-specific
  - always contains pre-defined sequence of ones and zeroes
  - additional fields may be included
- outside of WSN, there are many purposes of a preamble
  - to minimize the occurence of collisions (CSMA/CD)
  - to synchronize all recipients
  - etc.

Medium Access Control

Low-power listening

- Low-power listening (LPL)
- turning on the radio in a reduced mode
- often combined with the rest of the principles:
  - preamble – to detect preamble on the carrier wave
  - CSMA – to detect if someone is transmitting data at the moment
- can be implemented by a different radio
  - e.g., on a different frequency, or entirely split from the primary radio logic
  - we already mentioned one such example: Wake-Up Radio (WUR) in 802.11ba

Medium Access Control

Preamble sampling

- Preamble sampling
- the radio sleeps most of the time
- wakes up periodically to detect preamble transmission
  - if no preamble is detected, sleep
  - if a preamble is detected, perform wake-up routine
- wake-up routine
  - depends on the protocol
  - if the preamble contains address, verify, that the address matches
    - if not, sleep; otherwise continue
  - if the preamble contains time to preamble end, schedule wakeup
  - etc.

Medium Access Control

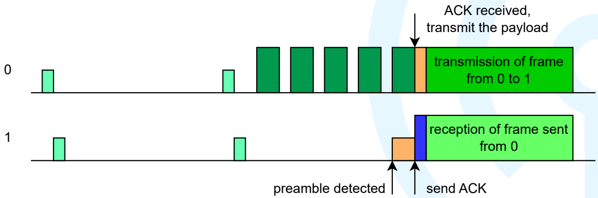
Preamble sampling

- *continuous*
  - once the preamble is detected, radio stays on
  - simple preamble (no additional info)
- *strobed*
  - once the preamble is detected, a short payload is also read
  - preamble consists of the fixed pattern and address
  - may also contain the time to the end

Medium Access Control

Early ACK

- *strobed* preamble allows for early ACKs
- instead of continuously transmitting the preamble, the transmitter introduces blank spaces between
  - transmitter listens for an acknowledgment (ACK) from the recipient
- receiver, once detecting a complete preamble, sends an ACK
- transmitter transmits the payload upon receiving the ACK



Medium Access Control

Early ACK

- Early ACK allows for reducing delays
- the received does not go back to sleep
- instead, transmitter is informed that the received is ready now
- overall time spent on preamble transmission is reduced
  - also saves the power on transmitter side

Medium Access Control

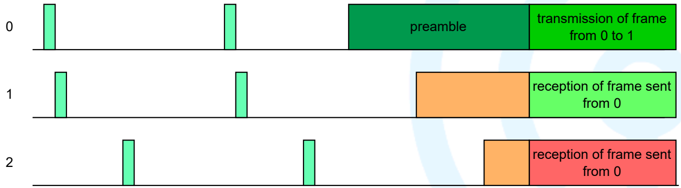
Used protocols

- There are many asynchronous protocols, e.g.:
  - B-MAC, B-MAC+
  - X-MAC
  - Speck MAC
  - RC-MAC, PW-MAC
  - DPS-MAC
  - WiseMAC
  - etc.

Medium Access Control

B-MAC

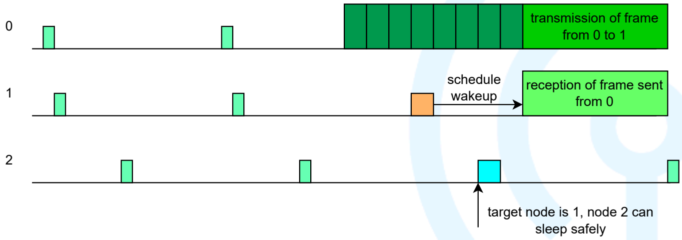
- B-MAC (Berkeley MAC)
- uses:
  - CSMA
  - Low-power listening
  - very short active time for sensing the carrier
  - continuous preamble sampling
- overhearing problem, node waits in active state



Medium Access Control

B-MAC+

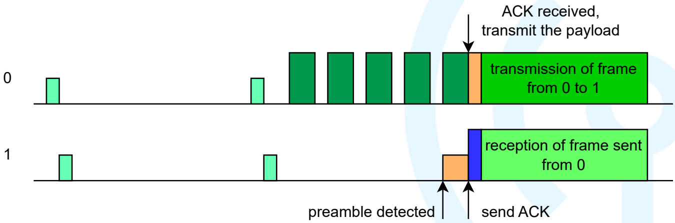
- B-MAC+
- enhancement of B-MAC:
  - strobed preamble with destination address
  - preamble contains number of preamble blocks that follow before payload
- more complex transmitter and receiver code



Medium Access Control

X-MAC

- X-MAC
- enhancement of B-MAC+:
  - strobed preamble with destination address
  - early ACK – transmitter switches to listening mode between preambles
- reduces delays, reduces power consumption on transmitter side



Medium Access Control

SpeckMAC

- SpeckMAC
- two variants:
  - SpeckMAC-B
    - similar to B-MAC+
    - instead of preamble, it uses a designated *wake up packet*
    - broader use than the preamble
  - SpeckMAC-D
    - redundantly retransmits the data packet
    - may be better if data packets are small
    - no need to differentiate between preamble and payload
- SpeckMAC-D was more widely used:
  - simple applications with very small data packets
  - reduced delay
  - simple implementation

Medium Access Control

RC-MAC

- RC-MAC (Receiver-Centric)
- topology-aware protocol
- bases the design on the fact, that most WSN's have a tree topology
- channel access scheduling
  - for heavy-traffic clusters, nodes partially synchronize their wake up plans
  - reduces collisions
- sequential retransmission of lost packets
- improved throughput
- complex code for both ends

Medium Access Control

DPS-MAC

- DPS-MAC (Dual Preamble Sampling)
- extension of X-MAC
- samples the preamble twice during a preamble-sized time window
  - if the channel is free during both sampling, immediately goes back to sleep
  - if the channel is busy during at least one, it powers on the receiver to wait for preamble
- this protocol aims to reduce the idle listening
- idle listening is introduces in X-MAC, as the ACK space can fit into the listening period
  - therefore, receiver may miss the preamble

Medium Access Control

WiseMAC

- WiseMAC
- introduces access-points (APs) with presumably unlimited power supply
- APs learn wake-up schedules of all nodes
- uses long preambles with continuous sampling (sender-initiated)
- AP wakes up just in time to transmit only the required part of preamble
- reduces receiver idle time
- both receiver- and sender-initiated variants exists

Medium Access Control

PW-MAC

- PW-MAC (Predictive-Wakeup)
- receiver-initiated
  - receiver transmits beacons periodically to indicate its availability
  - beacon is transmitted after the data frame to indicate successfull reception (ACK) and to indicate availability
  - no beacon = data were corrupted
- pseudo-random wakeup schedule
  - every node has its PRNG, that "predicts" next wakeup
- this approach can be applied to sender-initiated protocols also

Medium Access Control

STEM

- STEM (Sparse Topology and Energy Management)
- not a MAC protocol by itself
- separate wakeup and data channels
- two variants:
  - STEM-B – sender-initiated beacon protocol, sends beacons on wakeup channel
  - STEM-T – sender-initiated protocol, that sends a busy tone on wakeup channel

Medium Access Control

Synchronous protocols

- textbfsynchronous protocols
- very different from asynchronous
- there exists a (shared) wakeup schedule
- every node knows, when to wake up
- base categories:
  - **centralized** – there exists a central base station (or coordinator) that acts as arbiters of wakeup slots
  - **decentralized** – local information about wakeup schedules on each node

Medium Access Control

Synchronous protocols

- **centralized synchronous protocols**
- base station, that schedules wakeups of all nodes
- assign slots
- several types:
  - bitmap assignment (BMA)
  - self-organizing TDMA protocols (SOTP)
  - event-driven TDMA (ED-TDMA)

Medium Access Control

Synchronous protocols

- **distributed synchronous protocols**
- no central node
- only local information on each node
- examples:
  - self-organizing MACs (SMACS)
  - Power-Aware Clustered TDMA (PAC-TDMA)
  - Distributed Energy-Aware (DEA-MAC)
  - Traffic-Adaptive (TRAMA)

Medium Access Control

Synchronous protocols

- BMA protocols
- bitmap assignment
- two phases:
  1. cluster establishment phase
  2. functional phase
- cluster establishment protocol uses CSMA to elect cluster head based on power level
- functional phase is split into *k* sections of the same length
- each functional phase period contains three periods:
  1. contention
  2. data transfer
  3. idling

Medium Access Control

BMA

1. *contention period*
   - each node turns on the radio
   - in a pre-defined time slot, each node transmits a single-bit flag, if it intends to transmit some data
   - after each node sent its flag, the cluster head prepares a schedule for each node and broadcasts it
   - nodes then goes to sleep or wakes up depending on the schedule
2. *data transfer period*
   - every node transmits in its assigned time slot
   - outside of its slot, it may sleep
3. *idling period*
   - the whole network sleeps for pre-defined time period

Medium Access Control

BMA

- advantages
  - relatively simple hardware
  - effective
  - power-saving
- disadvantages
  - often requires more complex software
  - ineffective for networks with low traffic

Medium Access Control

SOTP

- Self-Organizing TDMA Protocol (SOTP)
- assumptions:
  - base station sees all sensors
  - sensors see only its neighbors
- time split into time frames
- time frames split into time slots
- number of slots is greater than number of nodes in the network

Medium Access Control

SOTP

- time slot types:
  - broadcast slot (BR) – always the first slot in time frame
  - carrier sensing slot (CA) – always second
  - transmitting slot (TX)
  - receiving slot (RX)
  - idling slot (ID)
- each node is in one of the three states:
  - searching (also starting state)
  - synchronized
  - registered

Medium Access Control

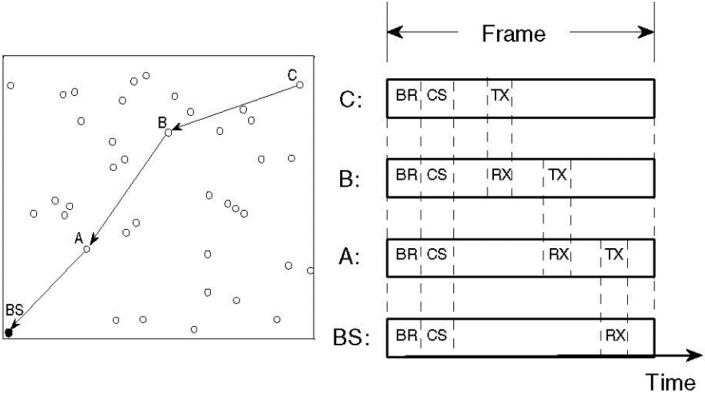
SOTP

- protocol states:
  1. every node starts in *searching* state
  2. base station periodically transmits Slot Allocation Packet (SAP) in BR slot
     - SAP contains allocation of time slots to every registered node in network
  3. node in *searching* state allocates a free slot from SAP and sends Register packet (REG) to base station; the node also selects one of *registered* nodes as its parent; the node changes its state to *synchronized*
  4. base station receives the REG packet and allocates the slot; broadcasts the allocation in next SAP packet
  5. the node receives the SAP packet with its allocation; the node changes its state to *registered*
- TX slots of children is the parents RX slot

Medium Access Control

SOTP

- SOTP slot allocation and relationships



Medium Access Control

SOTP

- advantages
  - effectively reduces delays
  - implements the original TDMA
  - does not require clustering – is self-organizing
- disadvantages
  - base station reaches all nodes
  - data aggregation happens on application layer – must not induce delays on MAC

Medium Access Control

S-MAC

- Sensor-MAC (S-MAC)
- one of the simpler synchronous protocols
- nodes share wake up schedules
- every neighborhood (a cluster of nodes that "see" each other) shares a single schedule
- nodes "know" when to wake up just in time to transmit data
- periodically sleeps and listens (fixed periods)
- *listen* phase is split to SYNC, RTS and CTS subperiods
  - SYNC – part of the time to synchronize schedules
  - RTS – part of the time to request transmission permission
  - CTS – part of the time to respond to RTS

Medium Access Control

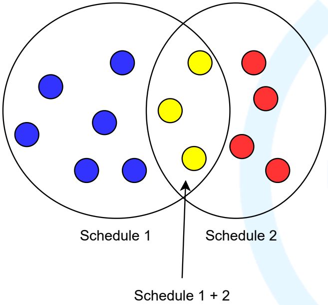
S-MAC

- Sensor-MAC (S-MAC)
- SYNC packet
  - contains node ID
  - contains time to next sleep
- nodes choose to either synchronize with the received schedule, or to retain it as secondary schedule
- if no synchronization occurred yet, it always synchronizes

Medium Access Control

S-MAC

- S-MAC is fairly efficient if there is a single schedule
- when there are more active schedules, there must be a *border* nodes, which acquire multiple wakeup schedules
- *border* nodes gets depleted faster – they wake up more often



Medium Access Control

T-MAC

- T-MAC
- enhances S-MAC
- uses adaptive sleep time
- uses FRTS (Future Request To Send) for message chaining
  - prevents the target node from sleeping – signalizes the future data transfer
  - speeds up the transfer
  - minimizes delays

Medium Access Control

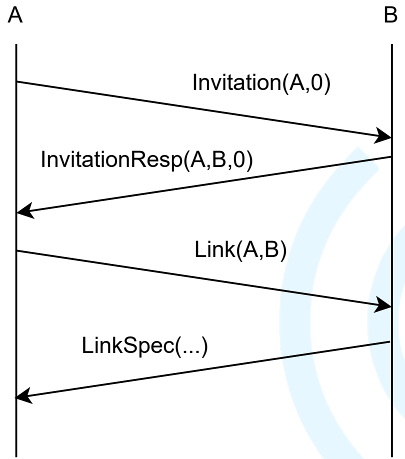
SMACS

- Self-organizing MAC (SMACS)
- distributed protocol
- no central node, nodes organize themselves
- flat topology (no clusters)
- each communication link may use a different frequency
- protocol control packets:
  - Type 1 – invitation – node discovers its neighbors
  - Type 2 – response to invitation (Type 1) packet
  - Type 3 – link – acknowledgment to Type 2 packet
  - Type 4 – link specification – link is formed, this packet exchanges additional info

Medium Access Control

SMACS

- SMACS invitation flow



Medium Access Control

PACT

- Power-Aware Clustered TDMA (PACT)
- passive clustering protocol
- multilevel topologies (tree)
- node roles:
  - cluster head
    - controls the cluster (schedules, ...)
  - gateway
    - exchanges packets between clusters
  - ordinary node
- cluster head and gateway depletes faster – node roles are reassigned once per given period

Medium Access Control

PACT

- *node slot allocation*
  - one TDMA slot assigned to each node in cluster
  - node can transmit to any other node in this time slot
- *link slot allocation*
  - one TDMA slot assigned to each node link (pair of nodes)
  - nodes can directly exchange packets only during this time slot

Medium Access Control

PACT

- PACT time slot consists of:
  - control slots
  - data slots
- every node transmits slot allocation in assigned control slot
- every node learns the schedule
- no collisions during transmission
- every node may go to sleep if it does not wish to transmit as was not of interest in any of foreign slots

Medium Access Control

Conclusion

- there are many MAC protocols
- we went through the most important principles and techniques
- many modern MAC protocols uses a combination of these