



# ICT Support for Adaptiveness and (Cyber)security in the Smart Grid

## Low-Power Wireless Communication

Olaf Landsiedel

# Devices



# Routing

Smart Meter  
(ZigBee)

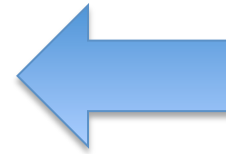
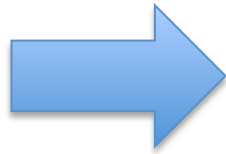
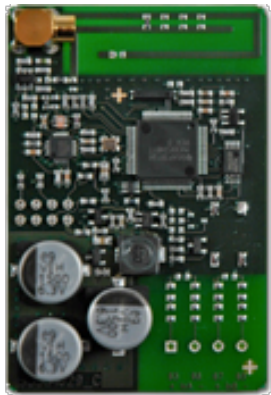


Concentrator  
(ZigBee + 3G)



Why this path? Why no direct communication?

# Lifetime



# Goal for Today

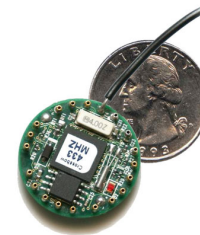
- Devices
  - Low-power wireless communication
  - Or a “wireless sensor node” in general
- Routing in low-power wireless networks
  - From source to sink
- Energy efficiency: often battery driven
  - Enable a life-time of years: allow devices to sleep

Low-Power Wireless

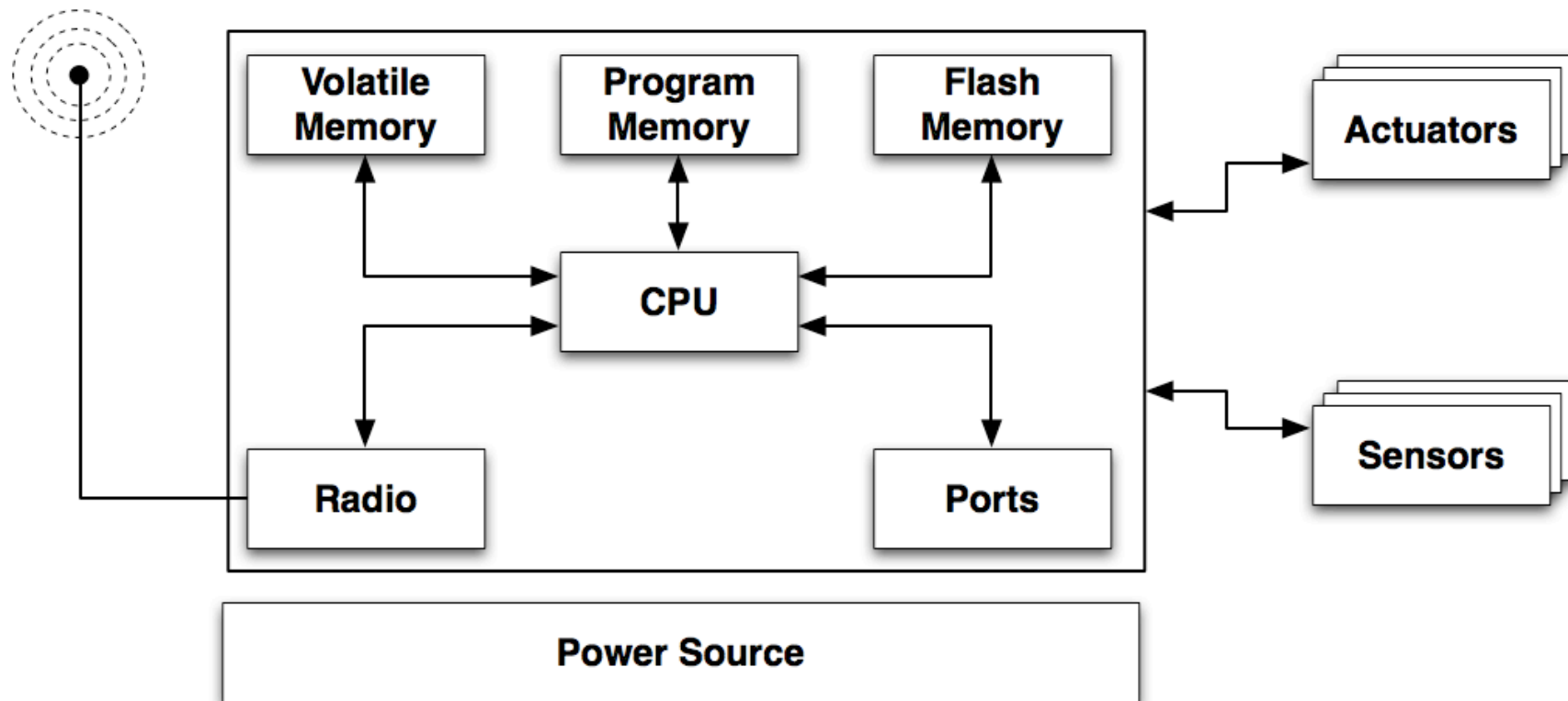
# Devices

# Requirements for such a Device

- Low cost
- High energy efficiency
- Small size



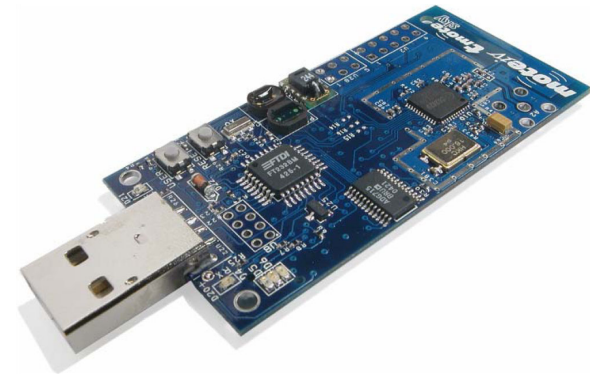
# A Sensor Node (or low-power wireless device)





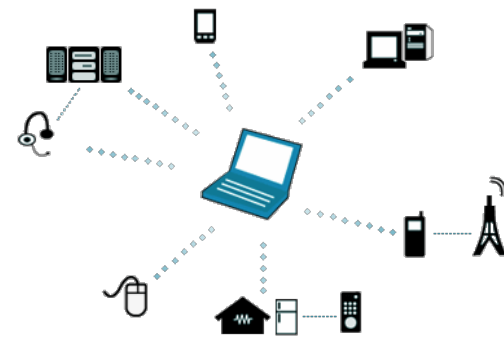
# A Sensor Node (or low-power wireless device)

- TI MSP 430 (16 bit RISC)
  - 8 MHz
  - 10 KB RAM
  - 48 KB code, 1MB flash
- Chipcon CC2420 radio
  - IEEE 802.15.4 compliant
  - 50 m. range indoor,  
250 m. range outdoor
  - bandwidth 250 kbits/s
- On-board antenna



# Why not use WiFi or Bluetooth?

- WiFi/WLAN (IEEE 802.11)
  - Topology: mesh + single hop
  - Throughput: >100 Mbps
  - Power Consumption: ~300mW
- Bluetooth
  - Topology: Single-hop network
    - Master <-> Slave
    - Not good for multi-hop networking
  - Throughput: up to 24 Mbit/s
  - Power Consumption: up to 30mW



# Summary: Device

- Low-Power Hardware
  - Simple Processor
  - Simple, energy-efficient radio
- Low cost, low energy consumption

Low-Power Wireless

# Routing

# Routing

Smart Meter  
(ZigBee)

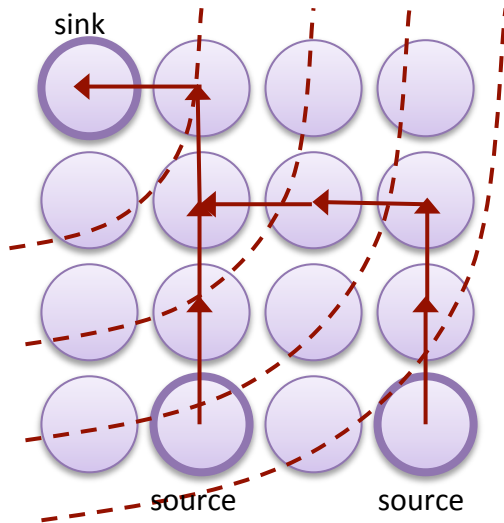


Concentrator  
(ZigBee + 3G)

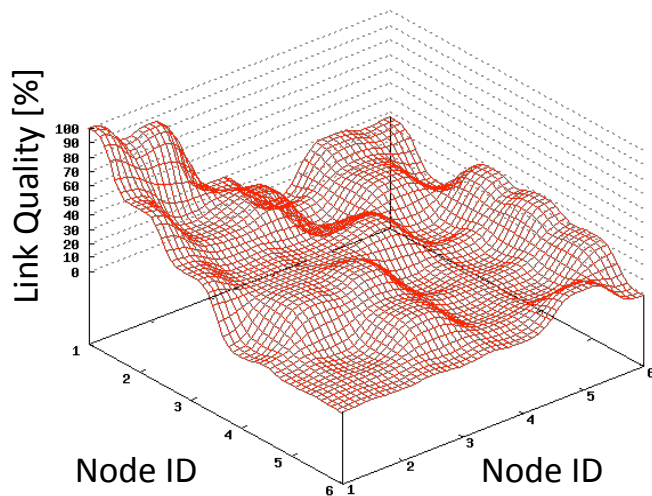


Why this path? Why no direct communication?

# Routing Metrics



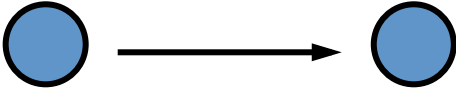
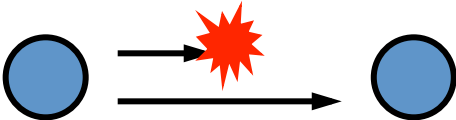
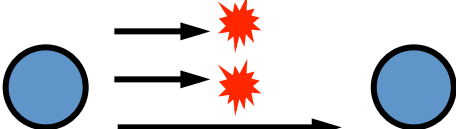
- Path Selection
  - Which path to select?
  - Routing Metric?
    - Minimize Hops?
    - Reliability?



- Wireless Links
  - Highly dynamic

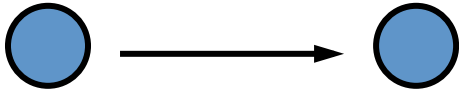
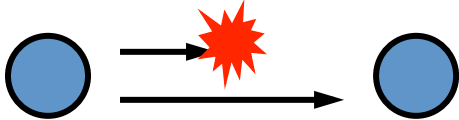
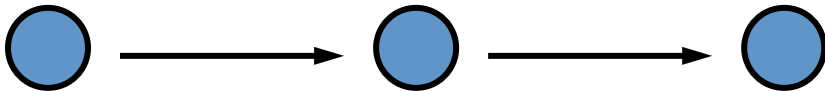
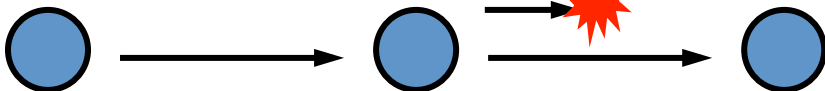
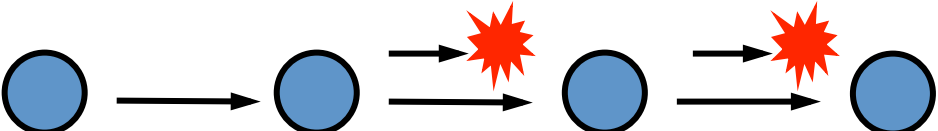
# Routing Metric: ETX

- Goal: Minimize total transmissions per packet
  - Use Metric: Expected Transmission Count (ETX)
    - Measure link over a time to determine ETX
  - Link throughput  $\approx 1 / \text{Link ETX}$

<u>Delivery Ratio</u>		<u>Link ETX</u>	<u>Throughput</u>
100%		1	100%
50%		2	50%
33%		3	33%

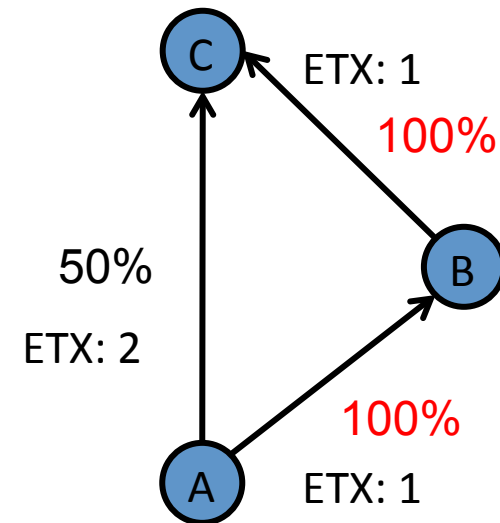
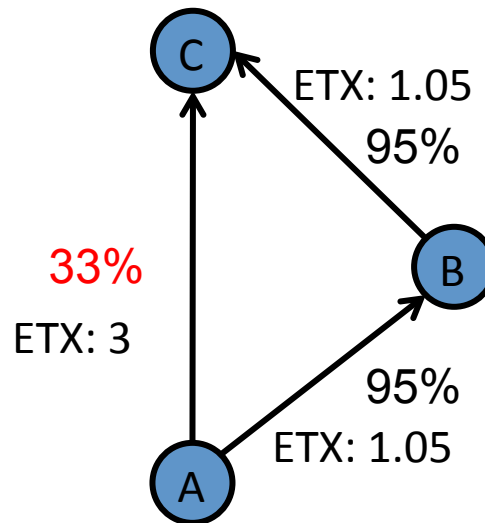
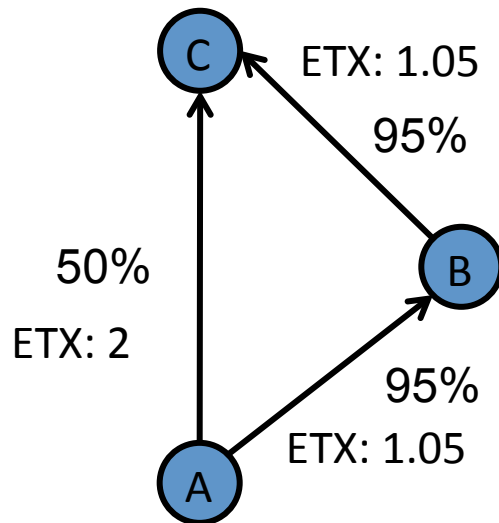
# Route ETX

- Route ETX = Sum of link ETXs
  - Communication is expensive
    - ETX predicts the tx count of a packet -> Reflects energy
  - Route selection:
    - Choose route with lowest route ETX

	<u>Route ETX</u>	<u>Throughput</u>
	1	100%
	2	50%
	2	50%
	3	33%
	5	20%



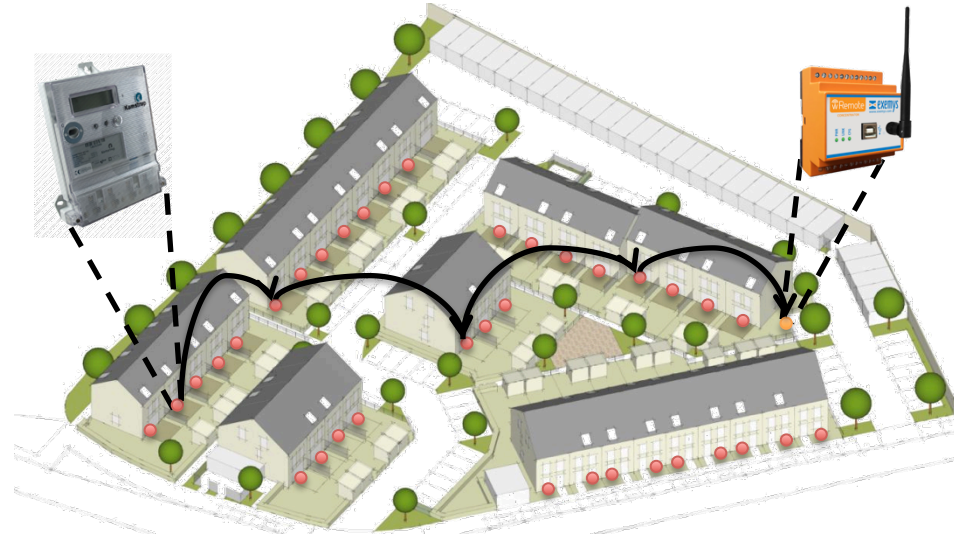
# Question: Which Route is better?



- Which route to take
  - A -> B -> C or A -> B?
  - Example 1
    - A -> B -> C: 2.1 TX
    - A -> B: 2 TX
    - Take A->B

- Example 2
  - A -> B -> C: 2.1 TX
  - A -> B: 3 TX
  - Take A -> B -> C
- Example 3
  - A -> B -> C: 2 TX
  - A -> B: 2 TX
  - Take any

# Summary: Routing Metric



- Expected Transmission Count (ETX)
  - Minimize total number of transmission
  - Good for energy: More transmissions -> more energy
  - Combines hops and reliability into single metric

Low-Power Wireless

# **Sleeping Devices**

# Energy-Efficient MAC

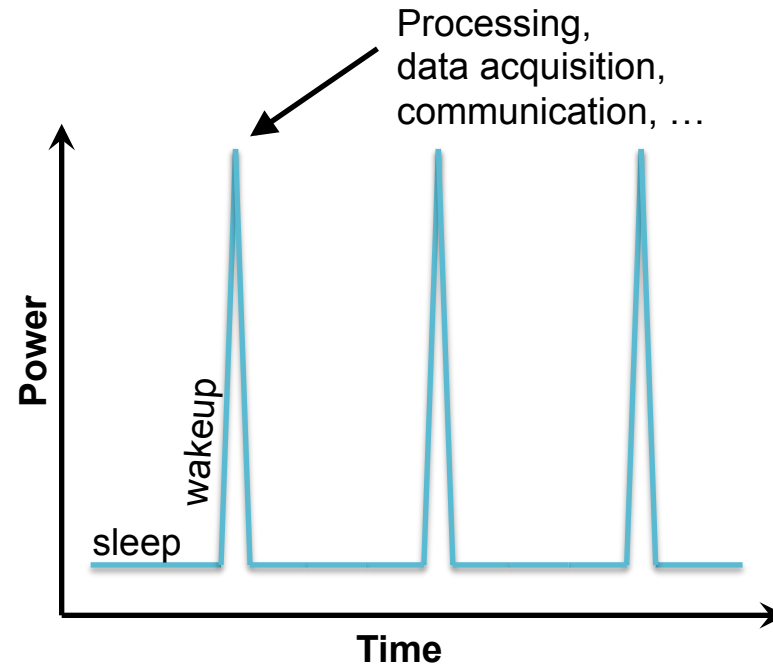
- Targeted life time of WSN
  - Months or years
- Simple back of the envelope calculation:
  - AA battery: About 2000 mAh
  - CC2420 radio:  
19.7mA in RX mode  
(listening to channel)
  - 2000mAh / 19.7mA  
= 101.5 hours  
= 6 days

Operation	Telos
Minimum Voltage	1.8V
Mote Standby (RTC on)	5.1 $\mu$ A
MCU Idle (DCO on)	54.5 $\mu$ A
MCU Active	1.8 mA
MCU + Radio RX	21.8 mA
MCU + Radio TX (0dBm)	19.5 mA
MCU + Flash Read	4.1 mA
MCU + Flash Write	15.1 mA
MCU Wakeup	6 $\mu$ s
Radio Wakeup	580 $\mu$ s

- We want month or years: How?
  - Keep radio off most of the time

# Solution: Duty Cycling

- Duty cycle
  - Wakeup, work, **sleep long**
    - Both: CPU and radio
  - Periodic
    - Data collection
    - Network maintenance
    - *Majority of operation*
  - Triggered events
    - Detection / notification
    - *Occurs infrequently*
      - *But... must be reported quickly and reliably*
  - Sleep:
    - CPU in deep sleep (timers only)
    - Radio off
- Result: Long lifetime
  - Months to years without changing batteries
  - Duty cycle from 0.1% to 1%



**How to communicate with a node sleeping 99% of the time?**

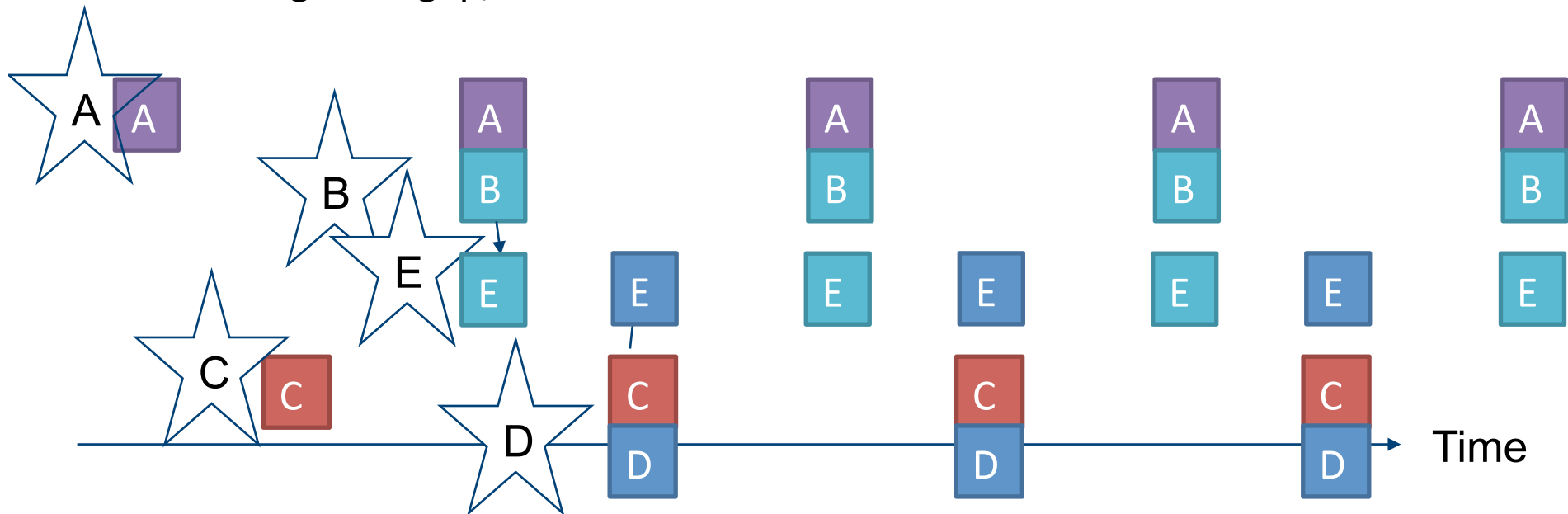
**Synchronous vs. asynchronous wakeups**

# Synchronous Duty Cycling

- Idea:
  - Switch nodes, radios off
  - Ensure that neighboring nodes turn on simultaneously
    - To allow packet exchange (rendezvous)
    - Requires Time Synchronization
    - Called Synchronous duty cyclings
- In wakeup phase
  - Only in these ***active periods***, packet exchanges happen
  - Need to also exchange wakeup schedule between neighbors

# Synchronous Duty Cycling

- Nodes try to pick up schedule synchronization from neighboring nodes
- If no neighbor found, nodes pick some schedule to start with
- If additional nodes join, some node might learn about two different schedules from different nodes
  - “Synchronized islands”
- To bridge this gap, it has to follow both schemes



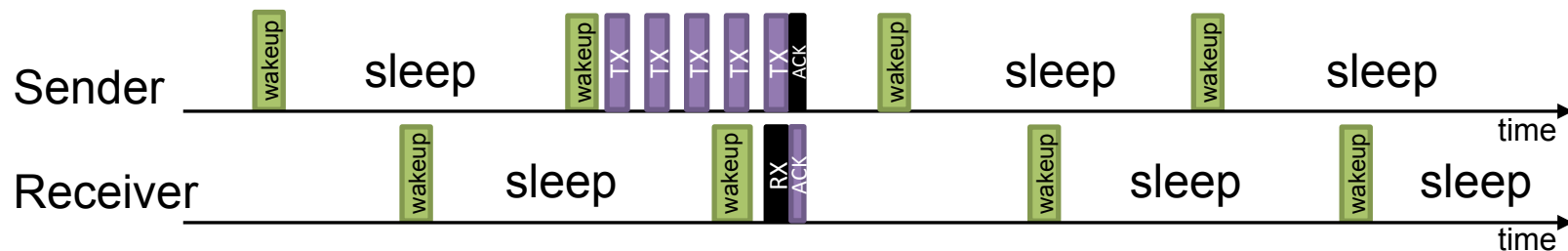
# Synchronous Duty Cycling: Discussion

- Pro: Energy-Efficient
  - A node sleeps most of the time
  - Periodically wake up for short intervals to see if any node is transmitting a packet
- Cons
  - Time sync overhead
    - Account for clock drifts etc.
    - Add guard spaces
  - Some nodes are in multiple “clusters”
    - More wakeups
    - Have higher energy consumption



# Asynchronous Duty Cycling

- So far: Periodic sleeping
  - Need some means to synchronize wake up of nodes
    - Ensure rendezvous between sender and receiver
- Alternative option: Don't try to explicitly synchronize nodes
  - Have receiver sleep and only periodically sample the channel
- Repeat packet until receivers wakes up
  - And acknowledges
  - No Synchronization required! Asynchronous duty cycle



# Asynchronous Duty Cycling: Discussion

- Pro: Energy-Efficient
  - A node sleeps most of the time
  - No need for time sync
  - Periodically wake up for short intervals to see if any node is transmitting a packet
- Cons
  - Transmission are costly
    - Especially when nodes wakeup rarely
  - A single transmissions is repeated many times
    - High channel utilization in this time

# Summary

- Devices: cheap, low-power
  - Low-power wireless
- Routing: Expected Transmission Count (ETX)
  - Account for link dynamics
- Synchronous and asynchronous duty cycling
  - Sleeping devices

Questions?

In part, inspired from / based on slides and figures from Jochen Schiller, Holger Karl, Klaus Wehrle, Kyoung-Don Kang, Leonardo Leiria Fernandes, Joe Polastre, Chenyang Lu, Leo Selavo, Luca Mottola, Adam Dunkels, and many others