The Flooding Time Synchronization Protocol

Miklos Maroti, Branislav Kusy,
Gyula Simon and Akos Ledeczi

Vanderbilt University

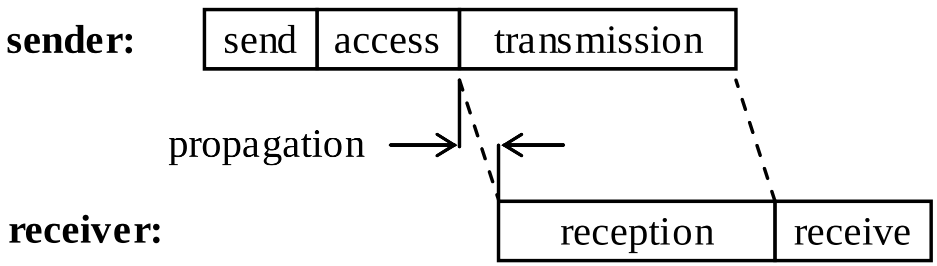
Contributions

- Better understanding of the uncertainties of radio message delivery and new techniques to reduce their effect
- FTSP time stamping (single-hop time synchronization)
  - Uses a single radio message to synchronize sender and receiver(s)
  - The most basic time synchronization primitive
  - 1.4 μs average, 4.2 μs maximum error on the MICA2
  - Implemented on MICA2, MICA2DOT and MICAZ
- FTSP multi-hop time synchronization
  - Synchronizes all nodes to an elected leader
  - Constant network load: 1 message per 30 seconds per node
  - Continuous operation, startup time is network diameter times 60 seconds
  - Fault tolerant: nodes can enter and leave the network, links can fail, nodes can be mobile, topology can change
  - Platform independent: uses the time stamping module

Time synchronization

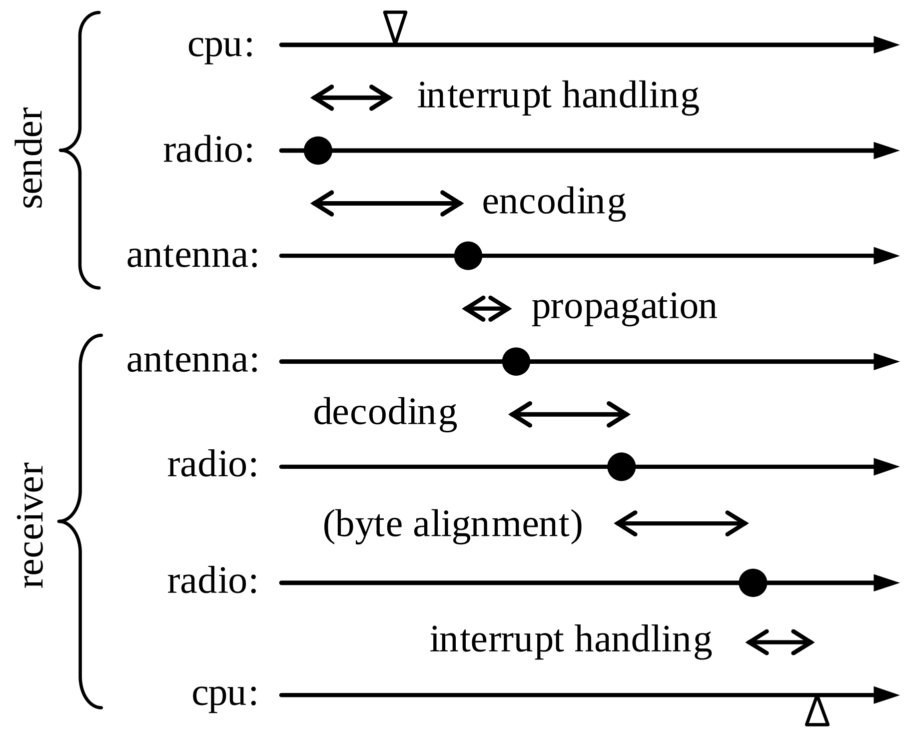
- Need for time synchronization
  - common reference points between neighbors
  - at sensing to obtain a global time-stamp
  - synchronized sensing and actuation
  - time arithmetic in local and global time
- Classification of algorithms
  - **Global time source:** internal vs. external
  - **Lifetime:** on-demand vs. continuous
  - **Scope:** all nodes vs. subsets vs. pairs
  - **Time transformation:** local clock adjustment vs. local and global time pair vs. timescale transformation
  - **Clock drift:** offset vs. skew and offset compensation
  - **Local clock:** CPU (high resolution, no power management, not stable) vs. external crystal
  - **Method:** sender–receiver vs. receiver–receiver
  - **Time stamping:** MAC layer vs. user space
- Previous approaches
  - Network Time Protocol (NTP)
  - Reference Broadcast Synchronization (RBS)
  - Timing-sync Protocol for Sensor Networks (NTSP)
  - Etc.
- Metrics
  - It is NOT only end-to-end accuracy
  - Network load (in messages per seconds per nodes)
  - Start-up time (as the function of the diameter)
  - Fault tolerance
    - nodes entering and leaving
    - incorrect and unstable local clocks
    - topology changes
  - Scalability

Radio message propagation



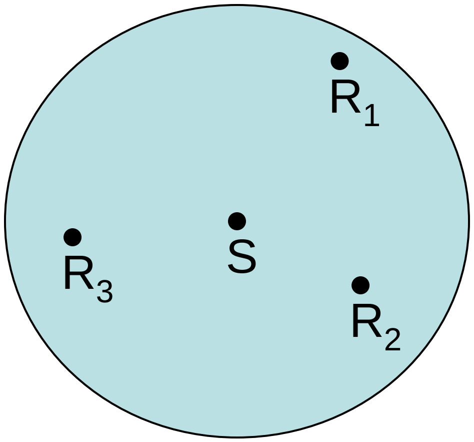
- **byte alignment time:** delay incurred because of different byte alignment of the sender and receiver

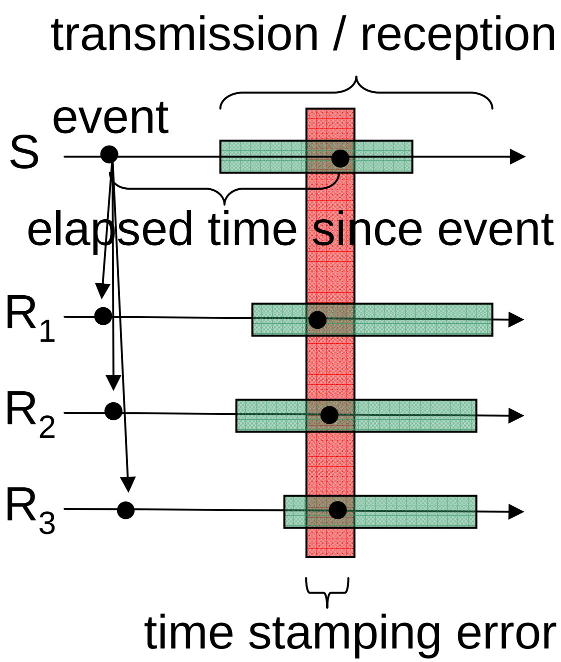
- **access time:** delay incurred waiting for access to the transmit channel
- **interrupt handling time:** The delay between the radio chip raising and the microcontroller responding to an interrupt
- **encoding time:** the time it takes the radio chip to encode and transform a part of the message to electromagnetic waves
- **propagation time:** time it takes for the electromagnetic wave to travel from sender to receiver
- **decoding time:** the time it takes on the receiver side to transform and decode to binary representation



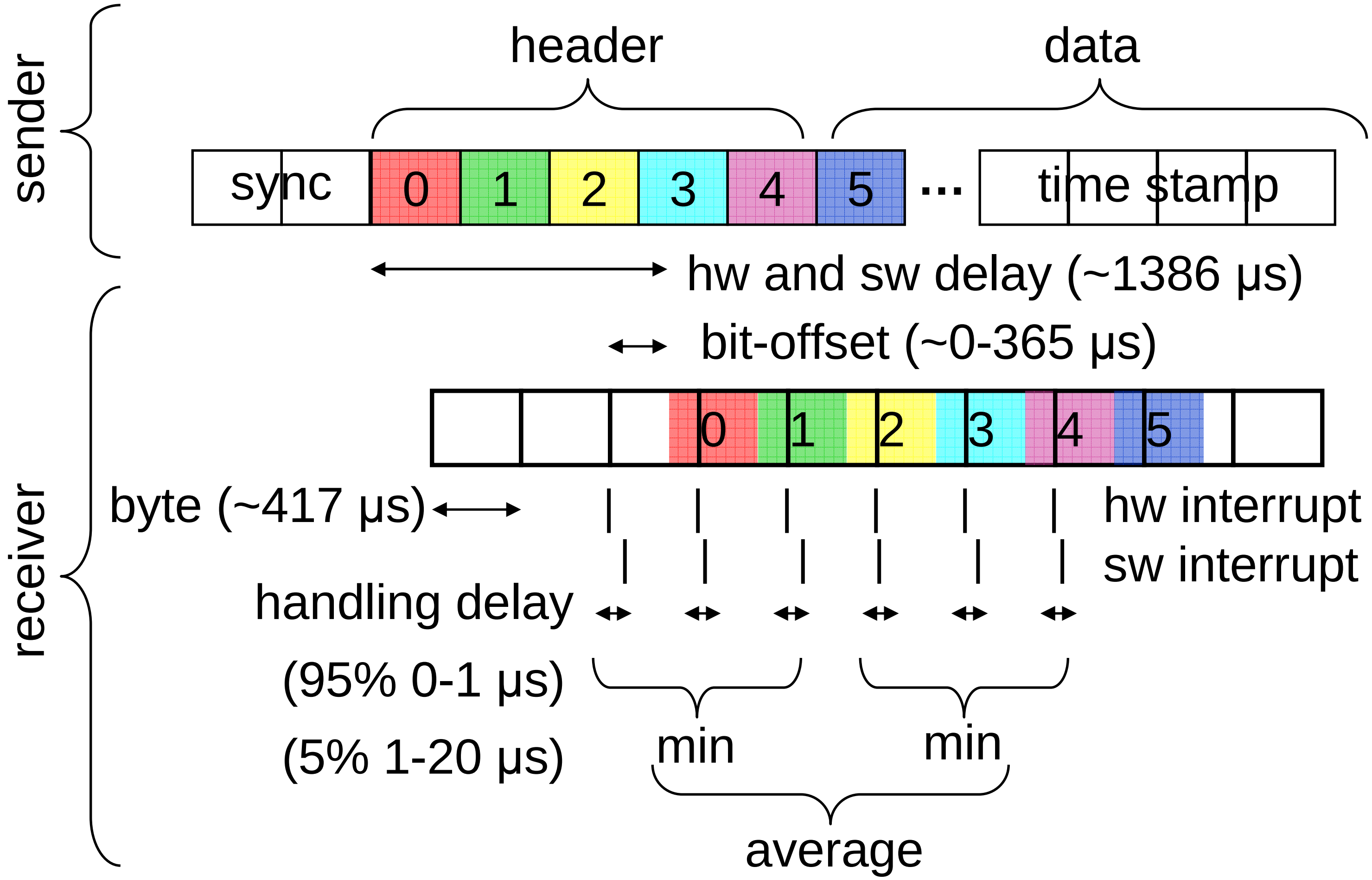
FTSP time stamping

- **Time synchronization primitive:** establishing time reference points between a sender and receiver(s) using a single radio message
  - –Sender obtains timestamp when the message was actually sent in its own local time
  - –The message can contain the local time of the sender at the time of transmission (or the elapsed time since an event)
  - –Receiver obtains timestamp when the message was received in its own local time
- **Algorithm**
  - –Multiple time stamps are made both on the sender and receiver sides at byte boundaries
  - –The time stamps are normalized, and statistically processed and a single time stamp is made both on the sender and receiver sides
  - –The final time stamp on the sender side can be embedded in the message
- **Uses**
  - –time synchronization protocols
  - –acoustic ranging
  - –shooter localization (implicit time synchronization while routing)





Time stamping on MICA2 platform



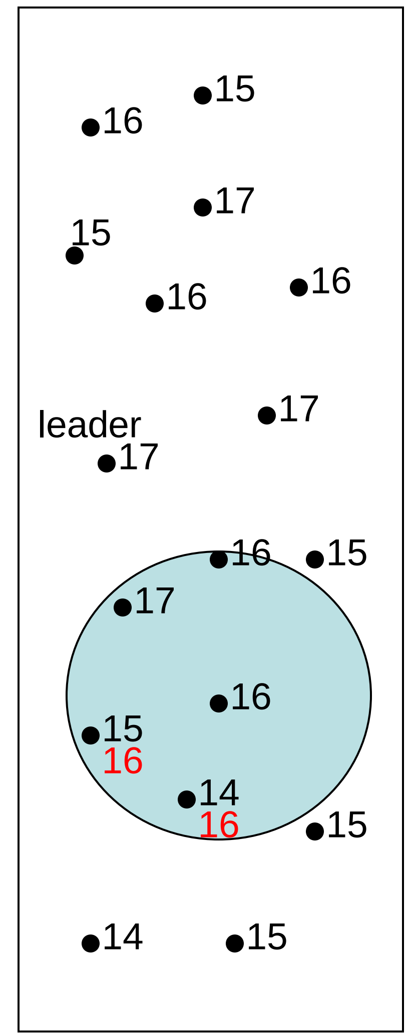
MICA2: 1.4 μs average error,
4.2 μs maximum error

MICA2DOT: 4 μs average,
12 μs maximum error

Limiting factor: the stability of the CPU clock

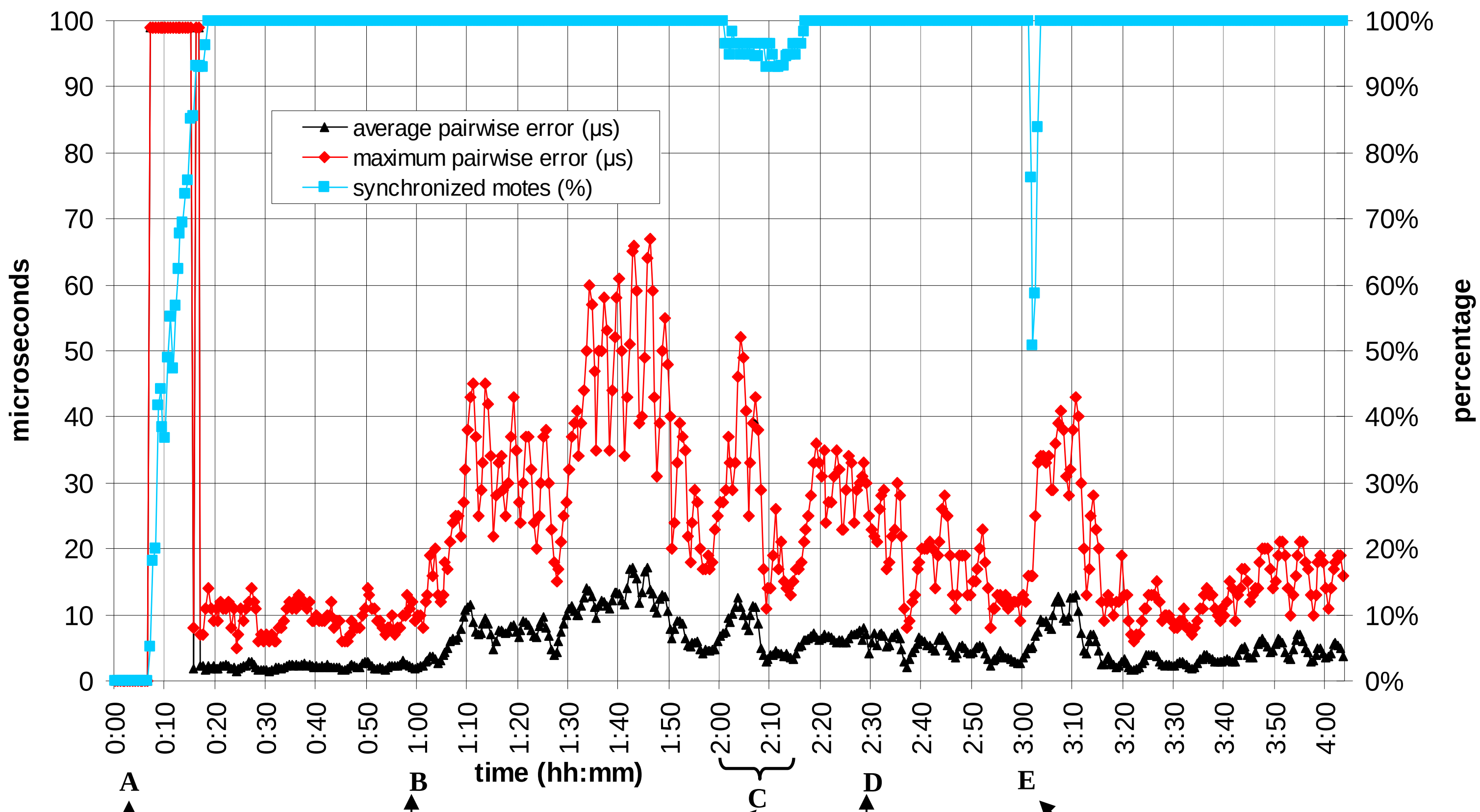
FTSP time synchronization

- **Local and global time**
  - Each node maintains both a local and a global time. Past and future time instances are translated between the two formats
  - Both the clock offset and skew between the local and global clocks are estimated using linear regression
  - Optimized to increase the numerical precision of calculations by normalizing the clock skew to zero (working with skew minus one)
  - Handles local and global clock overflows (uses 32-bit integers)
- **Hierarchy**
  - Global time is synchronized to the local time of an elected leader
  - Utilizes asynchronous diffusion: each node sends one synchronization msg per 30 seconds, constant network load
  - Sequence number, incremented only be the elected leader
  - Continuous operation, startup time is network diameter times 60 seconds
- **Robustness**
  - If leader fails, new leader is elected automatically. The new leader keeps the offset and skew of the old global time
  - Fault tolerant: nodes can enter and leave the network, links can fail, nodes can be mobile, topology can change
- **Platform independent:** uses the time stamping module

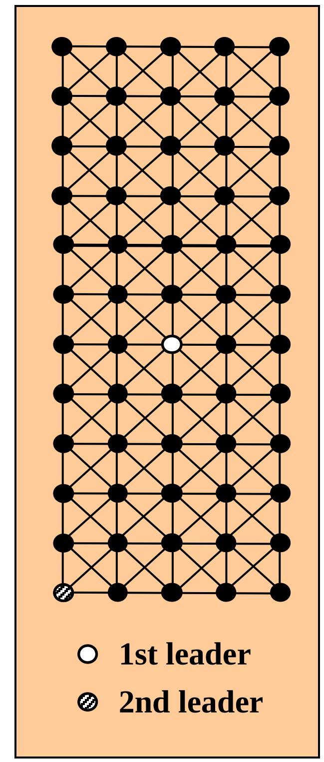


sequence numbers

FTSP experimental evaluation



topology:



all turned on

1st leader turned off

random nodes turned off/on

50% turned off

all turned back on

avg. error: 1.6 µs
max. error: 6.1 µs
per hop

Comparison to RBS and TPSN

- **Reference Broadcast Synchronization (RBS)**
  - Uncompensated delays: propagation, decoding, byte alignment, interrupt handling and receive times
  - Network overhead: 1.5 msgs per synchronization period
  - Hierarchy: clustered with timescale transformation
- **Timing-sync Protocol for Sensor Networks (TPSN)**
  - Uncompensated delays: encoding, decoding, interrupt handling times
  - Network overhead: 2 msgs per synchronization period
  - Hierarchy: spanning tree
- **Flooding Time Synchronization Protocol (FTSP)**
  - Uncompensated delays: propagation time
  - Network overhead: 1 msg per synchronization period
  - Hierarchy: flooding
  - End-to-end accuracy: average 1.6 μs per hop, max 6.1 μs per hop on MICA2
  - Robustness: nodes can enter and leave the network, links can fail, nodes can move
  - Startup: 2x diameter many synchronization periods
- **FTSP time stamping**
  - singe radio message
  - 0 or 4 bytes overhead per msg
  - 1.4 μs average, 4.2 μs maximum error on MICA2

Conclusion

- Time stamping: new time synchronization primitive
  - time synchronization protocols
  - synchronized sensing / actuation
  - power management
- CPU clock cycle precision
  - the effects of discrete time
  - separate local and global times
  - how to sense and actuate with this precision
- Robustness and scalability
  - startup and convergence time
  - scaling to 10000 nodes
  - power management
- One size does not fit all
  - need a spectrum of time synchronization algorithms
  - application specific and integrated solutions