Redes de Comunicação em Ambientes Industriais
Aula 13

Luís Almeida lda@det.ua.pt
Paulo Pedreiras pedreiras@det.ua.pt

Electronic Systems Lab-IEETA / DET
Universidade de Aveiro
Aveiro, Portugal

In the last episode ... (1)

✓ CANopen

- ✓ CAN High Layer Protocol
- ✓ Object-oriented Modeling of Device and Network
- ✓ Interoperability between Devices
- ✓ Interchangeability of Devices
- ✓ Off-the-shelf Plug-and-play Capability
- ✓ Off-the-shelf Configuration and Analysis Tools
- ✓ Standardized Communication Services
  - ✓ Peer-to-peer communication
  - ✓ Segmented Data Transfer
- ✓ Network and Node Configuration
- ✓ Network and Node Error Handling

In the last episode … (2)

- ✓ **CANopen**
  - ✓ Protocols:
    - ✓ Process Data Object (PDO) Protocol
    - ✓ Service Data Object (SDO) Protocols
    - ✓ Synchronization (SYNC) Protocol
    - ✓ Emergency (EMCY) Protocol
    - ✓ Network Management Protocols:
      - NMT Message Protocol
      - Boot-Up Protocol
      - Error Control Protocol

Ethernet

standards.ieee.org/getieee802/802.3.html

Ethernet

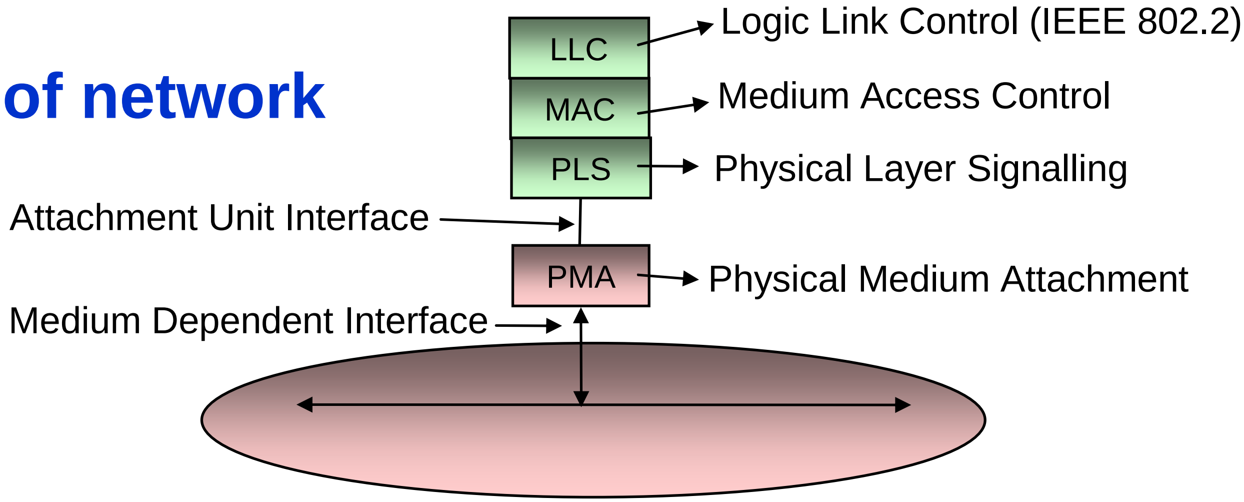
- **Created in the mid 70s** (!) by Robert Metcalfe at the Xerox Palo Alto Research Center.
- Aimed initially at **sharing** expensive **computing peripherals** in office environment (particularly, it was developed to connect a printer to a computer)
- From then on, it became extremely popular, being used in many domains beyond its original scope. Particularly in **industrial systems**, it became the most serious candidate for the unifying communications protocol, from the plant floor to the management.
- Standardized in the mid 80s as **IEEE 802.3**

Ethernet

- **Multi-master, broadcast**, serial **bus** (initially) or **star** (currently)
  - Bus: ☺ simpler cabling / ☹ faults in any point may cause complete communication disruption; fault tracing difficult
  - Star: ☹ cabling more complex and expensive / ☺ more fault tolerant, easier and faster fault source tracing
- Synchronous transmission with Manchester bit coding
- Transmission rate of **10, 100Mbit/s, 1** and **10Gbit/s**
- Max. number of nodes is **1024** (normally limited by a lower number of ports of the networking equipment)
- Max. link length is 100m (100Mbit/s, UTP cat5 cabling)
- Max. of 2 hubs between any 2 nodes (100Mbit/s)
- 2 architectures: **shared** (hubs), **segmented** (switches)
- Addressing classes: unicast, multicast and broadcast

Ethernet

- ✓ **Architecture of network interface**



- ✓ **Frame structure**

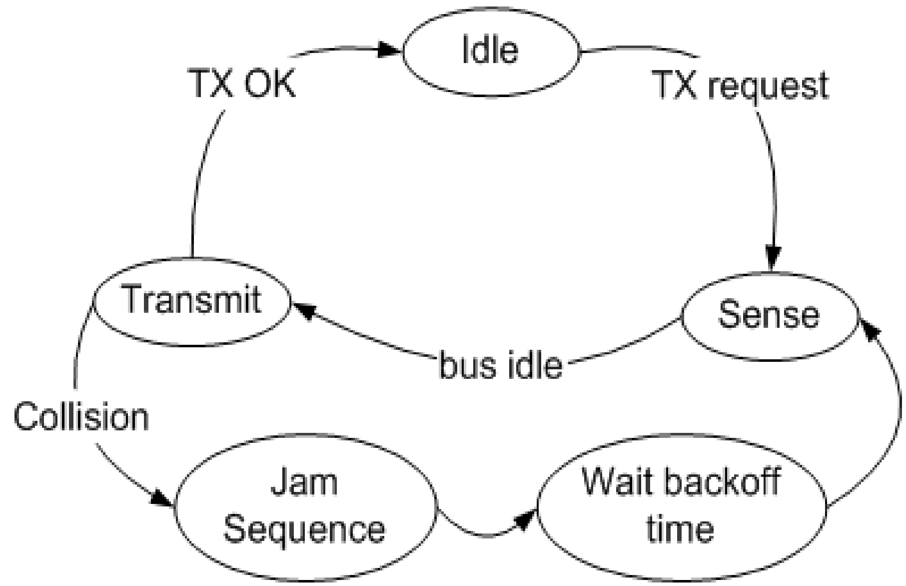
Dst. Addr	Src. Addr	Type/Len	LLC	SNAP	Data	CRC
6	6	2	3	5	38-1492 46-1500	4

LLC defined in IEEE802.2

LLC: Logic Link Control (Destination & Source Service Access Point + Control)

SNAP: Sub-Network Access Protocol (used for IP packets)

Ethernet



CSMA/CD (simplified)

- **CSMA/CD** non-deterministic arbitration (used in shared Ethernet, only)
- **1-persistent** transmission (transmits with 100% probability as soon as the medium is considered free)
- Collisions can occur during the interval of one ***slot*** after start of transmission (512 bits)
- When a collision is detected a **jamming** signal is sent (32 bits long)
- Frames vary between 64 (min) and 1518 (max) octets
  physically add 7+1 octets for preamble & SOF and 96 bit times for IFS

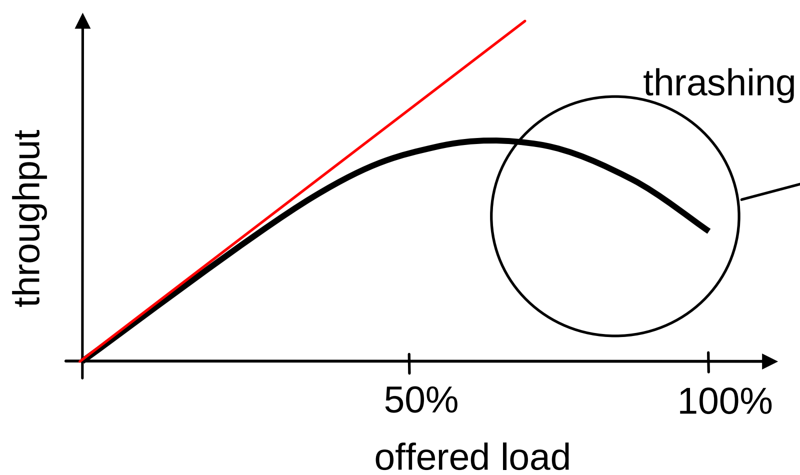
Ethernet

- Back-off and Retry mechanism:
  BEB = Binary Exponential Backoff
- Retry instant in number of slots is randomly chosen (uniform distribution) within randomization window.
- Randomization window starts with [0, 2) slots and duplicates every retry until [0,1024) slots.
  Maximum of 16 retries.
- Performance degrades after ~60% utilization
  - For higher utilization it is prone to **thrashing** caused by chained collisions (**not suited for real-time behavior**)

Ethernet

✓ **Thrashing effect**

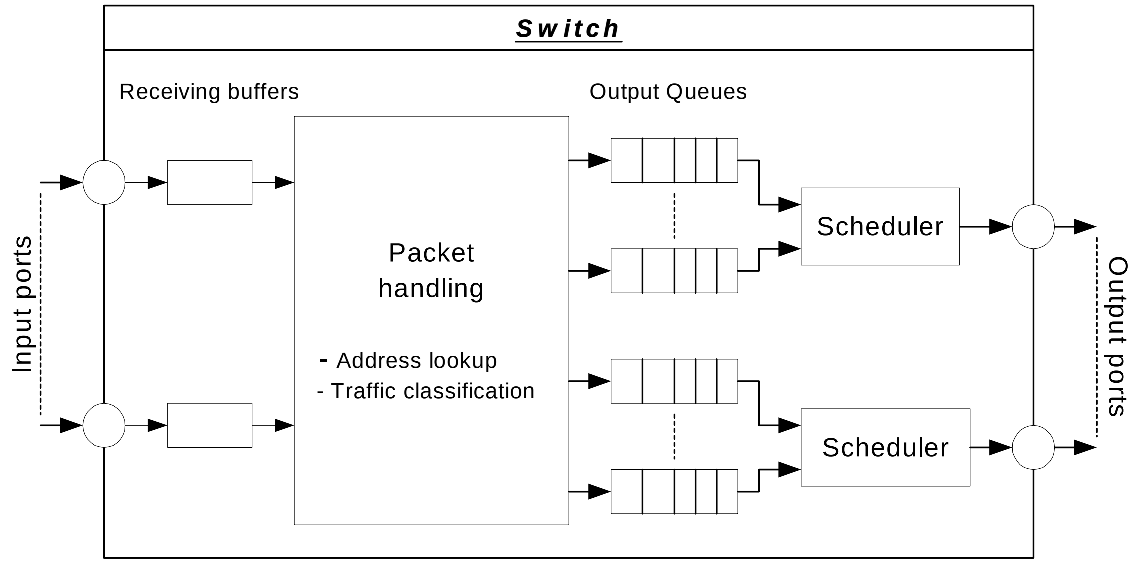
As the offered load increases,
the actual load handled by the network (throughput) decreases



Maximum network-induced delays can be very long with high packet-drop rates

Switched Ethernet

✓ Ethernet switches



- ✓ Nowadays the most common solution (even for general purpose networks)
- ✓ No collisions; nodes see a private collision domain (micro-segmented architecture)
- ✓ Messages directed to a busy output port are temporarily stored in the switch memory
- ✓ Up to 8 priorities

Switched Ethernet

- **Using switches**
  - Became the most common solution
    - Current switches are *wire-speed* (non-blocking)
    - 802.1D – possibly with multiple priority queues (802.1p)
    - 802.1Q – Virtual LANs
  - **Not perfect !**
    - Priority inversions in queues (normally FIFO)
    - Mutual interference through shared memory and CPU
    - Additional forwarding delay (with jitter caused by address table look up, address learning, flooding)
    - Delays vary with switch technology and internal traffic handling algorithms

Why Real-Time Ethernet

- ✓ **Motivations for using Ethernet**
  in application domains beyond its original one
  e.g. factory automation, large embedded systems
  (Decotignie, 2001)
  - ✓ Cheap wrt other high speed technologies
  - ✓ Widely available
  - ✓ Scalable tx rate
  - ✓ High bandwidth
  - ✓ Easy integration with office networks
    (important for logging, management and multi-level integration, e.g. CIM)
  - ✓ IP stacks widely available
  - ✓ Well known and mature technology

Why not Real-Time Ethernet

- **Downside of using Ethernet**
  in application domains beyond its original one
  e.g. factory automation, large embedded systems
  (Decotignie, 2001)
  - Connection costs higher than traditional fieldbuses
    (PHY, transformer, MAC)
  - High communication overhead
    (for short data items)
  - High computing power requirements
    (for efficient bandwidth usage)
  - Typical star topology not always adequate
    (may lead to extra long cabling)
  - Existing protocol stacks (mainly IP) not optimized for real-time operation

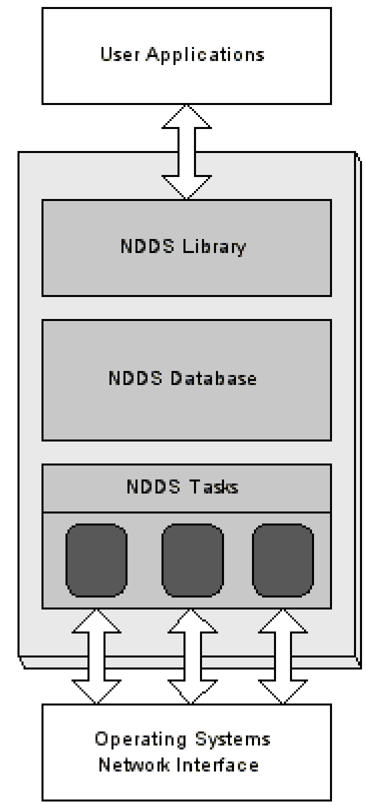
Real-Time Ethernet

- ✓ **Making Ethernet real-time**
  - ✓ Keep **network load low** (~1%, e.g. NDDS) !
  - ✓ **Avoid bursts** (e.g. traffic smoothing)
  - ✓ **Modify the back-off and retry** mechanism (e.g. CSMA/DCR, Virtual-time, windows, EQuB)
  - ✓ **Control transmission** instants (e.g. token-passing (RETHER, RT-EP), master-slave (FTT-Ethernet, ETHERNET-Powerlink), TDMA (PROFINET),…)
  - ✓ Micro-segmented **Switched Ethernet** with admission and transmission control

Brief review of RT Ethernet techniques

- CSMA/CD based.

  E.g. ORTE, Network Data Delivery Service (NDDS)
  - Statistical guarantees based on limitation of the bandwidth utilization
  - Publisher/subscriber cooperation model
  - NDDS database: publishers and subscribers metadata
  - NDDS database is shared among all nodes (holistic view)
  - NDDS Library: set of functions accessible to the application to
    - System set-up, Update/get variable's values, etc.
  - NDDS tasks
    - send and receive publication updates, NDDS database updates, etc.

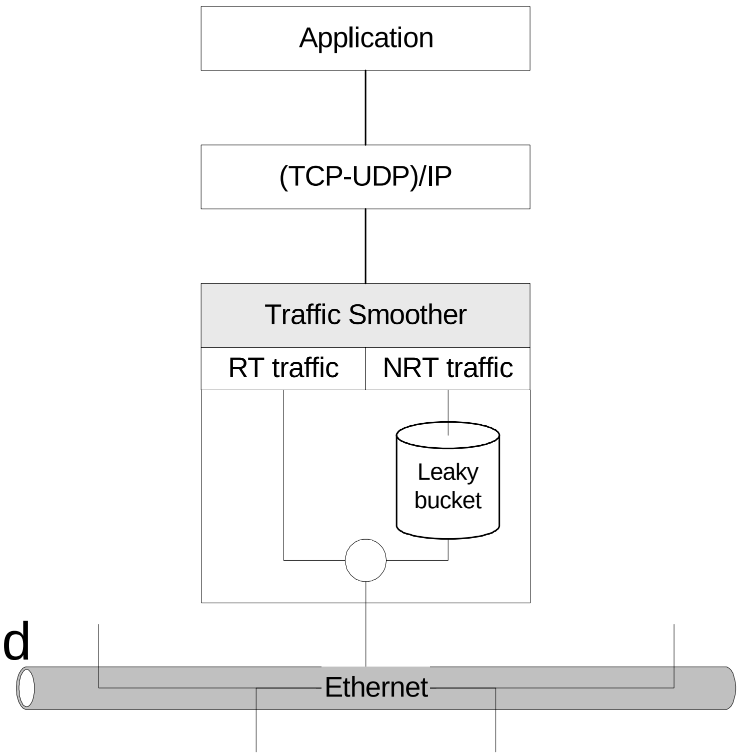


Brief review of RT Ethernet techniques

- CSMA/CD based.

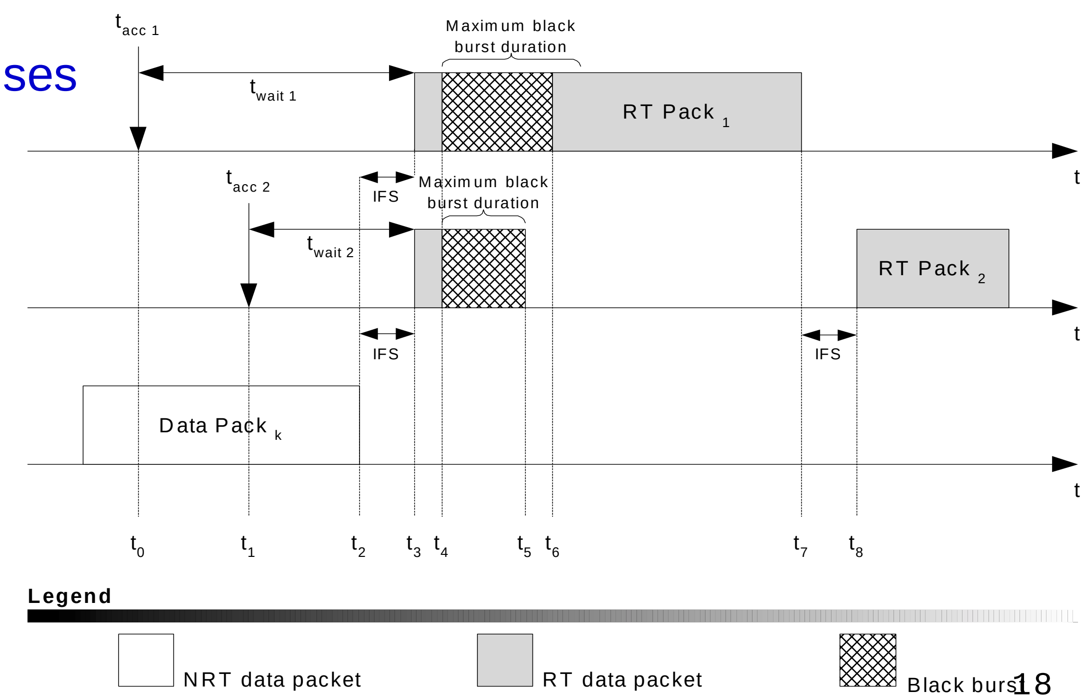
  E.g. Traffic-smoothing

  - Statistical guarantees based on limitation of the traffic bandwidth and burstiness
  - Real-time traffic transmitted immediately
  - Non-real-time traffic load and burstiness controlled using a shaper (leacky bucket)
  - Adaptive techniques recently proposed
    - Shaper parameters addapted according to the instantaneous communication characteristics (load and number of collisions)



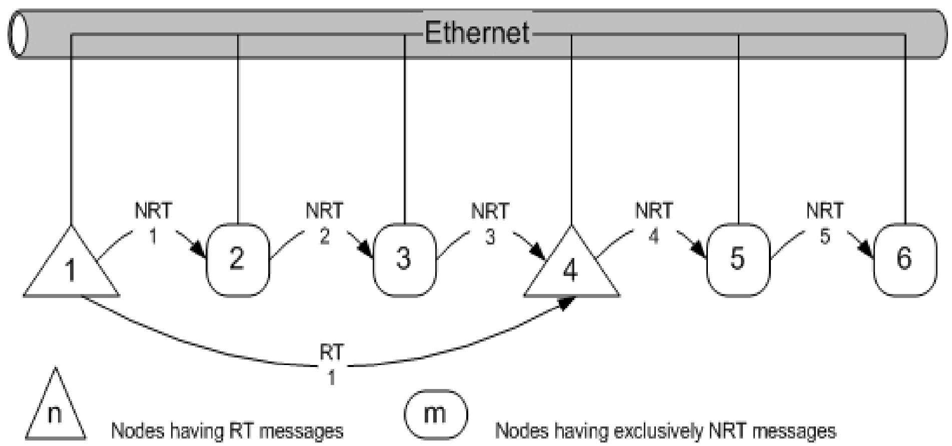
Brief review of RT Ethernet techniques

- Modified CSMA/CD protocols
  E.g. CSMA/DCR, Virtual-time, EQuB
  - Upon collisons RT nodes send black bursts (i.e. non-standard <jam signals) with length dependent on their priority (e.g. waiting time)
  - Bus state continuosly monitored. When a node senses no contention initiates the data transmission immediately
  - If at the end of its black burst a node senses a jamm signal in the bus reliquishes the attempt



Brief review of RT Ethernet techniques

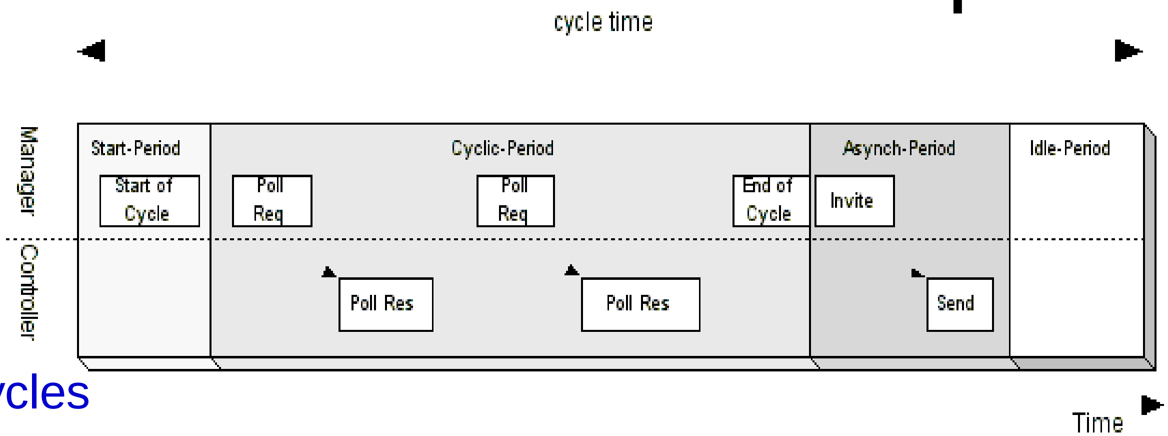
- Token passing
  E.g. RT-EP, RETHER



  - Real-time data assumed as periodic
  - Bus time divided in cycles of fixed duration
  - Access to the bus regulated by the token
  - First all RT nodes are visited
  - Until the end of the cycle the token visits NRT sources
  - E.g.
    - cycle *i* {1 – 4 – 1 – 2 – 3 – 4 – 5 – 6}
    - cycle *i*+1 {1 – 4 – 1 – 2},
    - cycle *i*+2 {1 – 4 – 1 – 2 – 3 – 4}
  - Explicit RT requests, online admission control

Brief review of RT Ethernet techniques

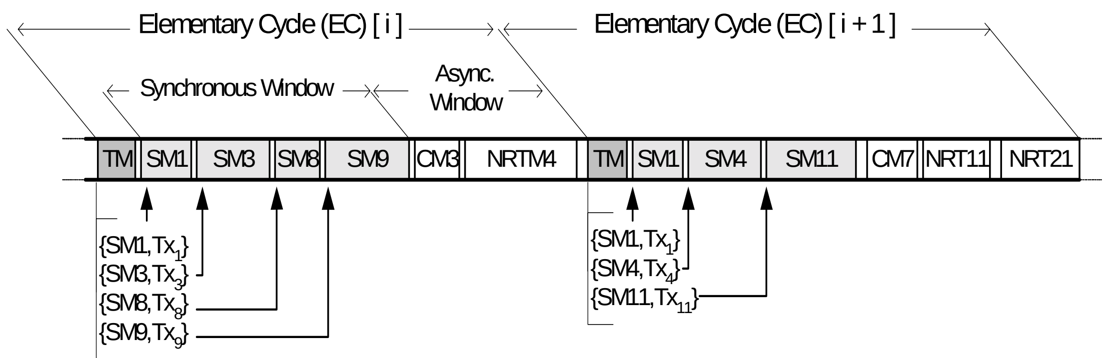
- ✓ Master/Slave
  E.g. ETHERNET Powerlink



- ✓ Fixed duration cycles
- ✓ Four phases (Start, Cyclic, Asynchronous and Idle)
- ✓ Periodic (isochronous) and event (asynchronous) data exchanges
- ✓ Network Manager (NM) (Master) coordinates the access to the bus
- ✓ Powerlink controllers (Slaves) transmit only after explicit request of the NM
- ✓ Different implementations allowed
  - ✓ Software
  - ✓ Dedicated CPU
  - ✓ Hardware
- ✓ Hub (high-speed) and switch architectures allowed

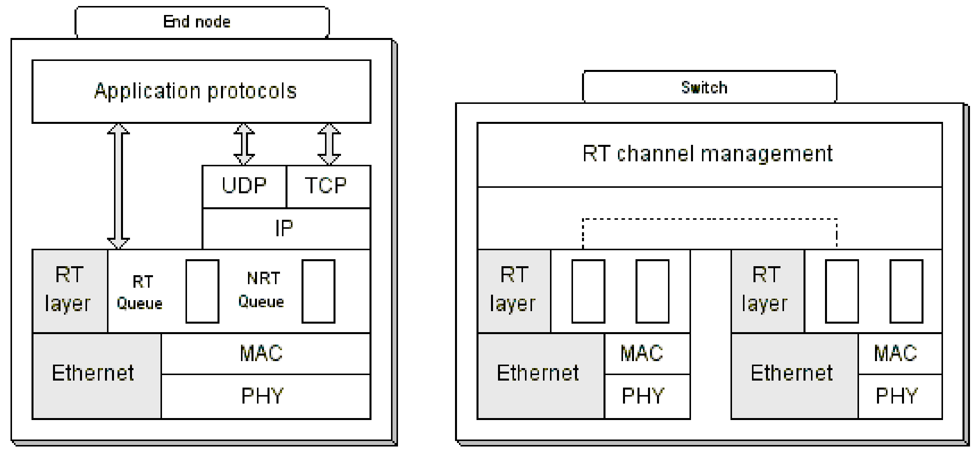
Brief review of RT Ethernet techniques

- Master/Slave
  E.g. FTT-Ethernet



  - Fixed duration cycles (Elementary Cycle / EC)
  - Two phases (Synchronous and Asynchronous windows)
  - Periodic (isochronous) and event (asynchronous) data exchanges
  - Master/Multislave . Master coordinates the access to the bus via a so-called trigger message, which contains the EC-schedule
  - Slaves transmit only after explicit request of the Master
  - Online scheduling
  - Online admission control
  - Arbitrary scheduling policies supported
  - Hub and switch architectures allowed

Brief review of RT Ethernet techniques



- Switched Ethernet
  E.g. Ethernet/IP,
  (modified) EDF switch
- Real-time and non-real-time traffic
- RT traffic scheduled according to EDF and subject to admission control
- Based on RT layer (RT-l) both in nodes and (modified) Ethernet switches
- RT-l
  - RT connection set-up
  - Admission control (both up and down links)
  - Deadline partitioning between the up and downlink
  - Message transmission and reception

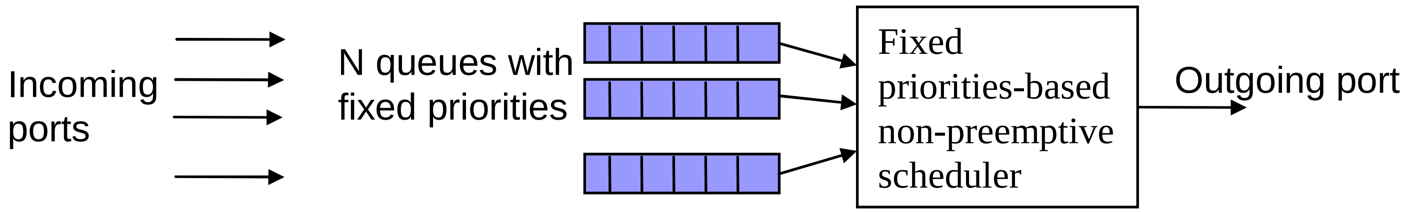
Real-time Ethernet

- ✓ **Schedulability analysis**
  - ✓ **<u>Shared Ethernet</u>:** depends on the specific overlay protocol
    - ✓ Master-Slave, TDMA, Token-passing, Traffic smoothing
  - ✓ **<u>Switched Ethernet</u>:** depends on queuing policies and switch internals
    - ✓ There are models that can be used directly, including for multiple priority queues and chained switches (e.g. the Network Calculus)
  - ✓ For **high bit rates** and especially **IP communication** the delays in the **protocol stacks** become **more important** than the network-induced delays!

Real-time Ethernet

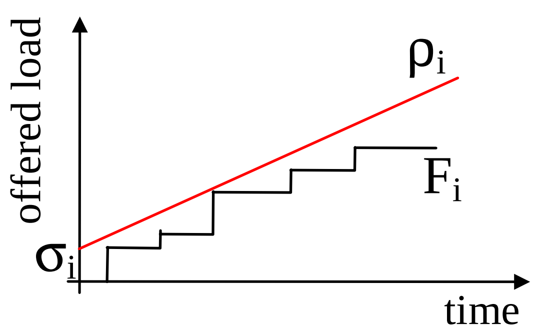
- ✓ **Schedulability analysis**
  (switched Ethernet)

  - ✓ Requires knowledge about the incoming traffic
  - ✓ Typical analysis for fixed-priorities:
    - ✓ **Network Calculus** (Cruz, 1991)
      also known as the (σ,ρ) model
    - ✓ **Response-time analysis**
      (similar to CAN as previously shown)



Real-time Ethernet

- **Schedulability analysis**
  (switched Ethernet – Network Calculus)
  - Cumulative arrival at queue *i* (F_i) upper bounded by (σ_i,ρ_i): $F(t) - F(s) \leq \sigma_i + \rho_i * (t-s) \quad \forall_{0\leq s\leq t}$
  - Decreasing priorities with *i*
  - Channel capacity *c*
  - Upper bound on delay at queue i D_i



$$D_i = \frac{\sum_{j=1}^{i} \sigma_j + \max_{i+1\leq j\leq N} (C_j)}{c - \sum_{j=1}^{i-1} \rho_j} \quad , \quad c \geq \sum_{j=1}^{N} \rho_j$$

Stability condition (for $c \geq \sum_{j=1}^{N} \rho_j$)

Real-time Ethernet

- ✓ **Schedulability analysis**
  (switched Ethernet – Network Calculus)

  - ✓ One advantage of the network calculus is that is allows determining a bound to the burstiness of the outgoing traffic (σ'_i) and consequently to the **buffer requirements** of that flow

$$\sigma'_i = \sigma_i + \rho_i * \frac{\sum_{j=1}^{i-1} \sigma_j + \max_{i+1 \le j \le N} (C_j)}{c - \sum_{j=1}^{i-1} \rho_j}$$

Summary

- ✓ Ethernet frame structure (minimum data length is 46 octets)
- ✓ Bus and star topologies
- ✓ CSMA/CD operation
  - ✓ Non-deterministic collision-resolution mechanism
- ✓ Shared vs Switched Ethernet
- ✓ Why (not) Ethernet for Real-Time applications
- ✓ (Very!) Brief review of Ethernet-based RT protocols
  - ✓ CSMA/CD based: NDDS, Traffic Smoothing
  - ✓ Modified CSMA/CD: EQuB
  - ✓ Token-passing: Rether
  - ✓ Master/Slave: ETHERNET Powerlink, FTT-Ethernet (the best ☺)
  - ✓ Switched Ethernet: EDF switch
- ✓ Analitical tools
  - ✓ Network calculus