Parallel & Distributed Real-Time Systems

Lecture #12

Professor Jan Jonsson

Department of Computer Science and Engineering
Chalmers University of Technology

Administrative issues

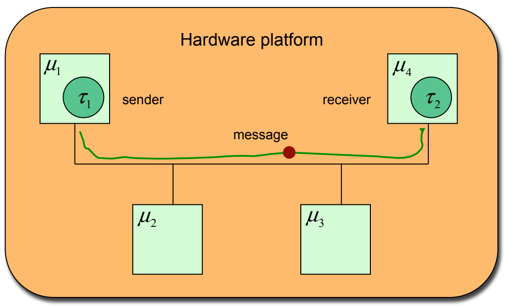
This week's schedule:

- Only one lecture (Monday)
- No lectures or consultation sessions on Thursday
- Presentation time slots on Monday, Wednesday and Thursday

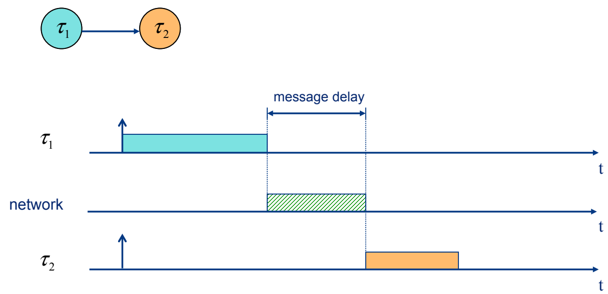
Next week's schedule:

- Guest lecture on Thursday, May 21 @ 08:00
  - Industrial communication networks (Dr. Roger Johansson, Chalmers)
- Regular lecture on Thursday, May 21 @ 10:00
- Consultation sessions on Monday and Friday
- Presentation time slots on Monday and Thursday

Network communication



Network communication



CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Message delay:

- Message delays are caused by the following overheads:
  - Formatting (packetizing) the message
  - Queuing the message, while waiting for access to medium
  - Transmitting the message on the medium
  - Notifying the receiver of message arrival
  - Deformatting (depacketizing) the message

Formatting/deformatting overheads are typically included in the execution time of the sending/receiving task.

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Queuing delay:

- The queuing delay for a task is caused by:
  - Waiting for a corresponding time slot (TTP/C, FlexRay)
  - Waiting for a transmission token (Token Ring, FDDI)
  - Waiting for a contention-free transmission (Ethernet)
  - Waiting for network priority negotiation (CAN)
  - Waiting for removal from priority queue (Switched Ethernet, EDD-D)

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Transmission delay:

- The delay for transmitting the message is a function of:
  - Message length (bits)
  - Data rate (bits/s)

$$t_{\text{frame}} = \frac{N_{\text{frame}}}{R}$$

and

  - Communication distance (m)
  - Signal propagation velocity (m/s)

$$t_{\text{prop}} = \frac{L}{v}$$

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

How is the message transfer scheduled between tasks assigned to different processors?

- Integrated scheduling:
  - Scheduling of tasks and inter-task communication are regarded as comparable operations.
  - Requires compatible dispatching strategies.
- Separated scheduling:
  - Scheduling of tasks and inter-task communication are performed as separate steps.
  - Allows for different dispatching strategies.

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Integrated scheduling:

- Suitable for simple homogeneous systems with known assignment of tasks to processors
- Examples:
  - Time-driven task dispatching + TTP/C network protocol
  - Static-priority task dispatching + CAN protocol
  - Static-priority task dispatching + Token Ring network protocol

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Separated scheduling:

- Suitable for heterogeneous systems or when assignment of tasks to processors is not always known in advance
- Motivation:
  - Transmission delay is zero if communicating tasks are assigned to the same processor
  - Number of communication links that a message traverses may be a function of the assignment (depends on topology and routing strategy)
  - Different communication links may employ different message dispatching policies

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

How is the message transfer synchronized between communicating tasks?

- Asynchronous communication:
  - Sending and reception of messages are performed as independent operations at run-time.
- Synchronous communication:
  - Sending and receiving tasks synchronize their network medium access at run-time.

CHALMERS | UNIVERSITY OF GOTHENBURG

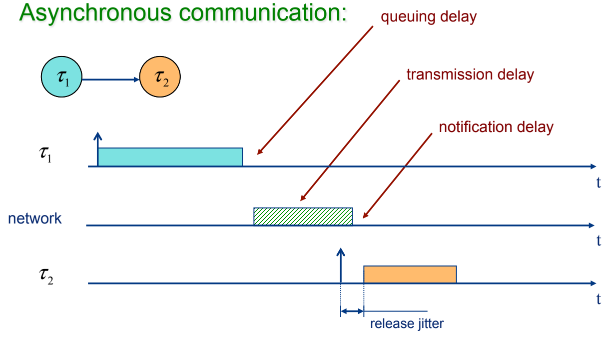
Network communication

Asynchronous communication

- Implementation:
  - Network controller chip administrates message transmission and reception (example: CAN, Ethernet)
  - Interrupt handler notifies the receiver
- Release jitter:
  - Queuing delays (at sender or in multi-hop network switches) and notification delay cause variations in message arrival time
  - Arrival-time variations gives rise to release jitter at receiving task (which may negatively affect schedulability)
  - Release jitter is minimized by using offsets for receiving tasks, or by maintaining message periodicity in multi-hop networks

Network communication

Asynchronous communication:



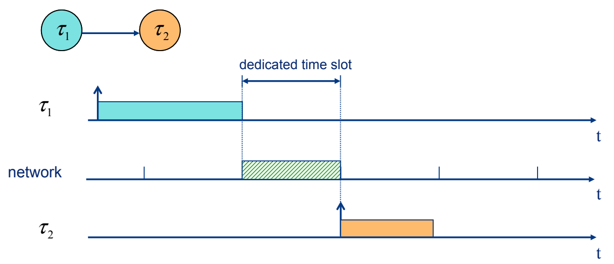
Network communication

Synchronous communication

- Implementation:
  - Network controller chip makes sure message transmission and reception occurs within a dedicated time slot in a TDMA bus network
  - Off-line static scheduling is used for matching the time slot with the execution of sending and receiving tasks
  - Queuing and notification delays can be kept to a minimum by instructing the off-line scheduling algorithm to use jitter minimization as the scheduling objective

Network communication

Synchronous communication:



Network communication

How is the message transfer imposed with a deadline?

- As a separate schedulable entity:
  - Suitable deadline-assignment techniques must be used
  - Worst-case message delay must be known beforehand
- As part of the receiving task:
  - No explicit deadline needed for message transmission
  - May impose release jitter on the receiving task

Network communication

How is the message transferred onto the medium?

- Contention-free communication:
  - Senders need not contend for medium access at run-time
  - Examples: TTP/C, FlexRay, Switched Ethernet
- Token-based communication:
  - Each sender using the medium gets one chance to send its messages, based on a predetermined order
  - Examples: Token Ring, FDDI
- Collision-based communication:
  - Senders may have to contend for the medium at run-time
  - Examples: Ethernet, CAN

CHALMERS | UNIVERSITY OF GOTHENBURG

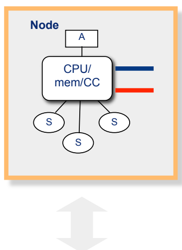
Network communication

Contention-free communication:

- One or more dedicated time slots for each task/processor
  - Shared communication bus
  - Medium access is divided into communication cycles (normally related to task hyper periods to allow for integrated scheduling)
  - Dedicated time slots provide bounded message queuing delays
  - TTP/C, TTCAN ("exclusive mode"), FlexRay ("static segment")
- One sender only for each communication line
  - Point-to-point communication networks with link switches
  - Output and input buffers with deterministic queuing policies in switches provide bounded message queuing delays
  - Switched Ethernet, EDD-D, Network Calculus

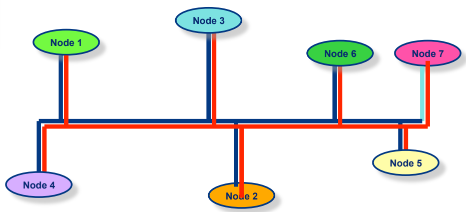
CHALMERS | UNIVERSITY OF GOTHENBURG

The TTCAN protocol



A second controller is required to implement the redundant bus

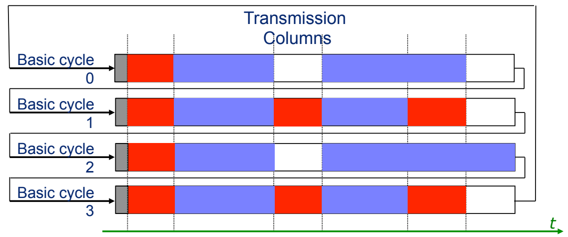
- Based on the CAN protocol
- Bus topology
- Media: twisted pair
- 1Mbit/s



CHALMERS | UNIVERSITY OF GOTHENBURG

The TTCAN protocol

"Exclusive" – guaranteed service
"Arbitration" – guaranteed service (high ID), best effort (low ID)
"Reserved" – for future expansion...

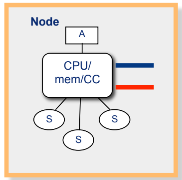


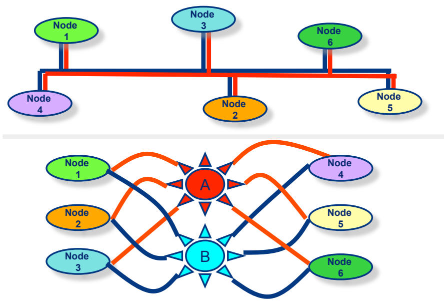
Time is global and measured in *network time units* (NTU's)

CHALMERS | UNIVERSITY OF GOTHENBURG

The TTP/C protocol

- Double channels (one redundant). Bus topology or "star" (optical)
- Media: twisted pair, fibre
- 10 Mbit/s for each channel



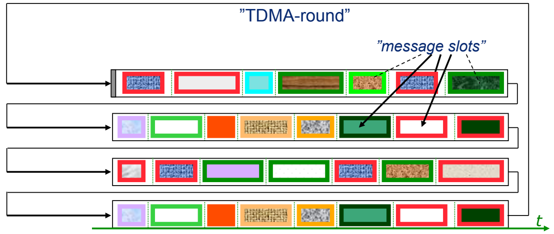


A network is built on either twin buses or twin stars.

CHALMERS | UNIVERSITY OF GOTHENBURG

The TTP/C protocol

All communication is statically scheduled
Guaranteed service

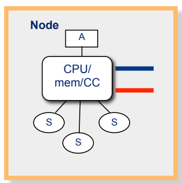


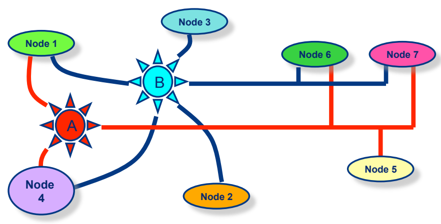
Non-periodic messages have to be fitted into static slots by the application

CHALMERS | UNIVERSITY OF GOTHENBURG

The FlexRay protocol

- Double channels, bus or star (even mixed).
- Media: twisted pair, fibre
- 10 Mbit/s for each channel





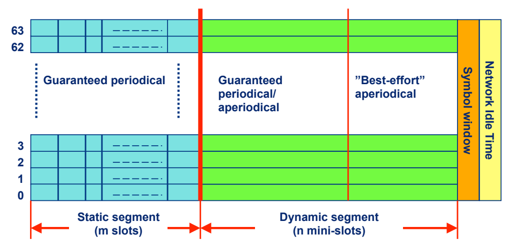
Redundant channel can be used for an alternative schedule

CHALMERS | UNIVERSITY OF GOTHENBURG

The FlexRay protocol

"Static segment" (compare w/ TTCAN "Exclusive")
– guaranteed service

"Dynamic segment" (compare w/ TTCAN "Arbitration")
– guaranteed service (high ID), "best effort" (low ID)



Max 64 nodes on a Flexray network.

CHALMERS | UNIVERSITY OF GOTHENBURG

Network communication

Token-based communication:

- Utilize a <u>token</u> for the arbitration of message transmissions on a shared medium
  - The sender is only allowed to transmit its messages when it possesses the token
  - Message priorities/quotas allows for bounded queuing delays
- Examples:
  - Timed-Token Protocol
  - Token Bus (IEEE 802.4)
  - Token Ring (IEEE 802.5)
  - FDDI (ANSI X3T9.5)

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Timed-Token Protocol: (Malcolm & Zhao, 1994)

- Concepts:
  - By token rotation (TR) we mean that the token has made a complete cycle among all the processor nodes.
  - The token cycle time is the <u>real</u> value of the time taken for TR.
  - The target token-rotation time (TTRT) is an <u>expected</u> value of the time taken for TR.
- Protocol:
  - Every time the token visits a processor node, it is allowed to transmit up to a pre-assigned quota of real-time messages.
  - At token reception, token cycle time is compared against TTRT:
    - if token is late, only real-time messages are transmitted
    - if token is early, non-real-time messages are also transmitted

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Timed-Token Protocol:

A necessary feasibility test:

The deadline of each message transmission must be at least twice the TTRT.

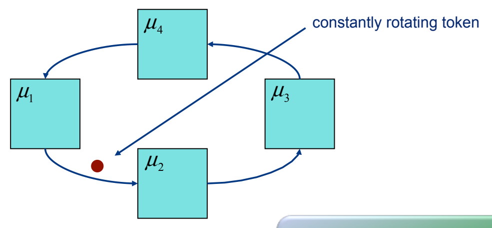
A sufficient feasibility test:

The accumulated transmission quotas should not exceed TTRT minus the overhead for token transmission time.

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Token Ring: (IEEE 802.5)



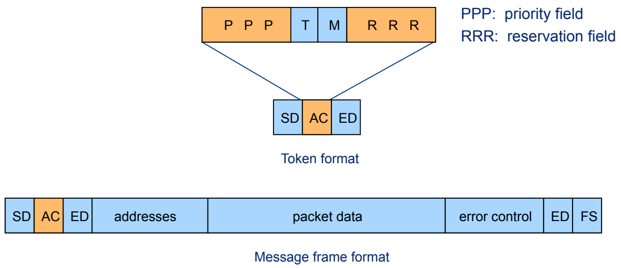
"token walk time": $W_T = (n-1)D_B + L + T_{prop}$

D_B : node delay
L : buffer delay
T_{prop} : ring propagation delay

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Token Ring message frame format:



CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Token Ring protocol:

1. Each node examines RRR of a busy token as it passes and inserts the priority of its pending message only if it is greater than the priority currently in RRR.
2. A node does not grab a "free" token unless the priority of its pending message is at least as high as the priority in PPP. Then the token status is changed to "busy".
3. A transmitting node appends its pending message after the "busy" token and sets RRR appropriately.
4. A transmitting node waits until it receives back the "busy" token before releasing the next "free" token with PPP set to the (possibly) updated RRR.

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Token Ring real-time protocol: (Sathaye & Strosnider, 1994)

The rate-monotonic (RM) scheduling algorithm can be adapted to the Token Ring protocol by assuming a non-preemptive dispatching model.

- Limitations:
  - Messages cannot be interrupted during transmission, which means that message scheduling is non-preemptive.
  - Message headers must be included in message size
  - Notion of highest priority might be outdated since the system is distributed
  - The number of priority bits (3) defined in IEEE 802.5 does not allow for an arbitrary number of priority levels.

CHALMERS | UNIVERSITY OF GOTHENBURG

Token-based communication

Token Ring real-time protocol: (Sathaye & Strosnider, 1994)

A sufficient and necessary feasibility test:

$$\forall i : R_i = t_{sys} + b_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil e_j \le D_i$$

t_{sys} : system overhead defined by the system

b_i : blocking time due to ongoing transmissions

e_j : "execution time" consisting of the following time components

- Capture token when node has highest-priority message pending
- Transmit message
- Transmit subsequent free token

Network communication

Collision-based communication:

- Utilize collision-detect mechanism to determine validity of message transmissions on a shared medium
  - The sender tries to send messages independently of other senders' intention to do so
  - Attempts may be done at any time or when some specific network state occurs
- Examples:
  - Ethernet w/ multiple senders (IEEE 802.3)
  - CAN (ISO 11898)

Collision-based communication

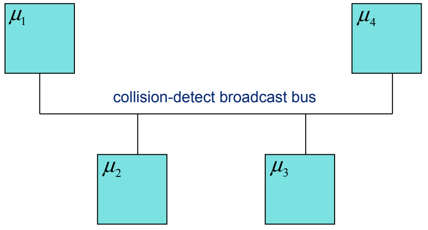
Ethernet protocols w/ multiple senders:

- Senders attempt to send a complete message
- If messages collide, all transmissions are aborted
- After collision, re-transmission is made after a random delay
- Protocol extensions for real-time systems:
  - VTCSMA (Zhao & Ramamritham, 1987)
  - Window Protocol (Zhao, Stankovic & Ramamritham, 1990)

Message queuing delay can in general not be bounded!
Therefore, these protocols do not give any guarantees for meeting imposed message deadlines!

Collision-based communication

Controller Area Network (CAN): (ISO 11898)



Collision-based communication

Controller Area Network (CAN):

- Senders transmit a message header (with an identifier)
- If messages collide, a hardware-supported protocol is used to determine what sender will be allowed to send the rest of the message; transmissions by other senders are aborted

Message queuing delay can be bounded with appropriate identifier assignment!
Therefore, this protocol makes it possible to meet imposed message deadlines!

Collision-based communication

CAN message frame format: (short format)

SOF	11-bit identifier	control	0 - 8 bytes of message data	error control	Ack	EOF

11-bit identifier is used for two purposes:
- assign a priority to the message (low number ⇒ high priority)
- enable receiver to filter messages

Wired-AND:

Each node monitors the bus while transmitting.

If multiple nodes are transmitting simultaneously and one node transmits a '0', then all nodes will see a '0'. If all nodes transmit a '1', then all nodes will see a '1'.

Collision-based communication

CAN protocol: (binary countdown)

1. Each node with a pending message waits until bus is idle.
2. The node begins transmitting the highest-priority message pending on the node. Identifier is transmitted first, in the order of most-significant bit to least-significant bit.
3. If a node transmits a recessive bit ('1') but sees a dominant bit ('0') on the bus, then it stops transmitting since it is not transmitting the highest-priority message in the system.
4. The node that transmits the last bit of its identifier without detecting a bus inconsistency has the highest priority and can start transmitting the body of the message.

Collision-based communication

CAN real-time protocols:

- Protocol #1: (Davis et al., 2007)
  - Any fixed-priority scheduling algorithm can be adapted to the CAN protocol by assuming non-preemptive dispatching.
- Protocol #2: (Zuberi & Shin, 1995)
  - The earliest-deadline-first (EDF) and deadline-monotonic (DM) scheduling algorithms can also be adapted to the CAN protocol by appropriately partitioning the identifier field.

Additional reading:
Study the paper by Davis et al. (2007)