

Distributed embedded systems

- characteristics, examples,
trends and challenges



Martin Törngren

Embedded Control Systems group
Mechatronics
KTH

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

1

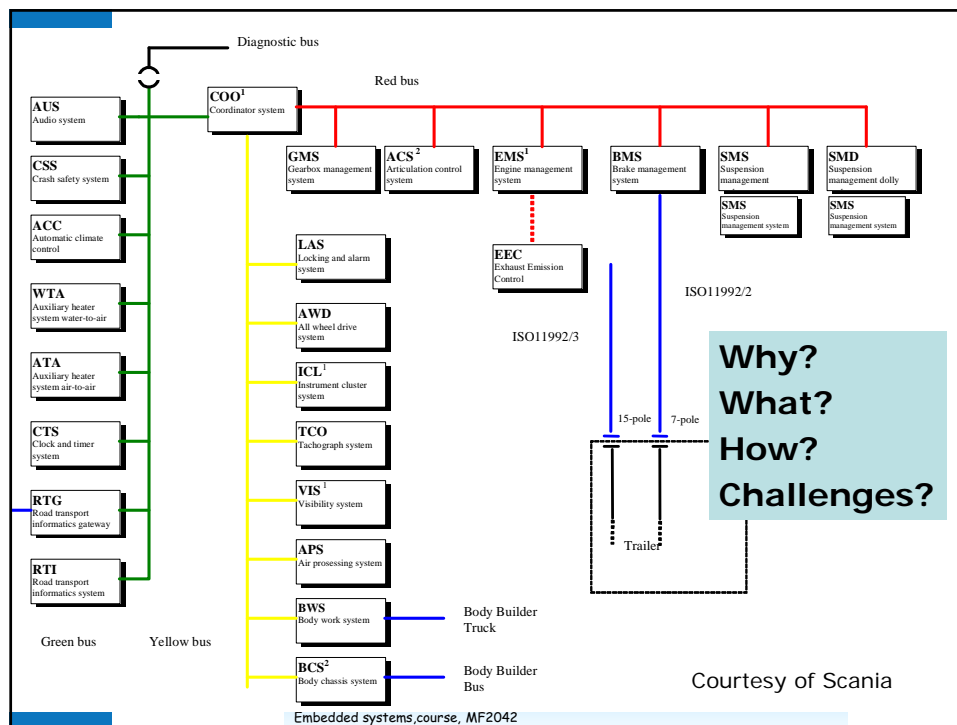
Outline

- Recap – why, what and how of distributed systems
 - Characteristics, Advantages, Problems
- Design issues for distributed embedded systems
 - Systems and application examples
 - Trends and research



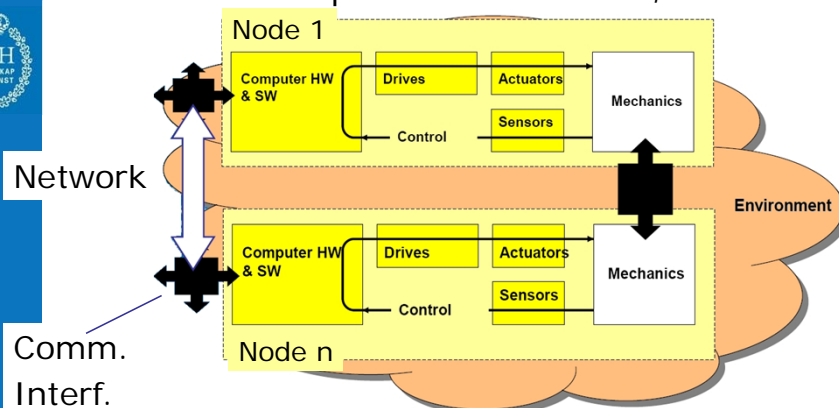
Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

2



What Is A Distributed System?

- A set of nodes connected by the network, cooperating to achieve a common goal
 - Node: a μC + I/O + communication interface
 - One or multiple networks: wired, wireless





Analogy

- Setup phase:
 - Communication model/pattern
 - Physical level; logical level; reliability
 - Broadcast vs. Point-to-point mode
- Design and implementation
 - Functions ~ the phrases
 - Allocated to hardware elements – "nodes"
 - Timing and global clock
 - Access to the same com. network/scheduling
 - Application control logic - centralized

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

5



Communication models

<u>OSI model oriented</u>	<u>RT model</u>
General purpose	Dedicated
Little apriori knowledge	Often static
Point to point logical connection	Multicast
Sender & receiver addressing	Source addressing
Positive ack retransmission	Datagram
E.g. TCP/IP!	E.g. CAN!

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

6

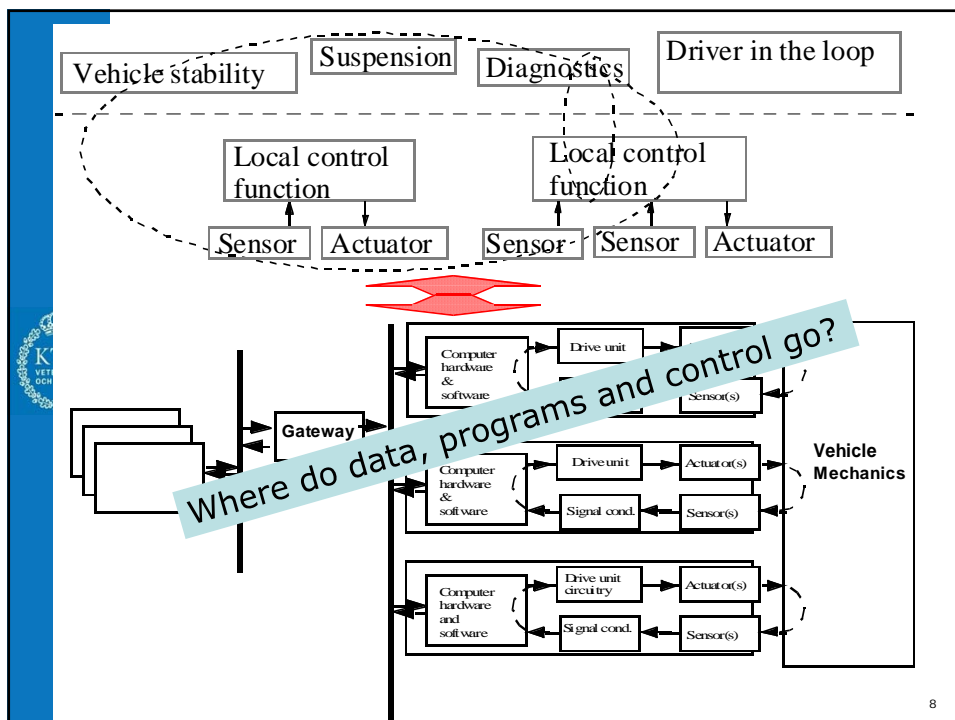


Views of distributed systems

- Software allocation and roles of nodes
- Application program view
- Driver view
- Logical topology
- Physical layout
- Cables and wires
- Physical vs. Logical communication
 - Protocols

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

7



8



Application programmer view

```
msg[0]=0;  
...  
msg[6]=0;  
msg[7]=50;
```

Application program interface

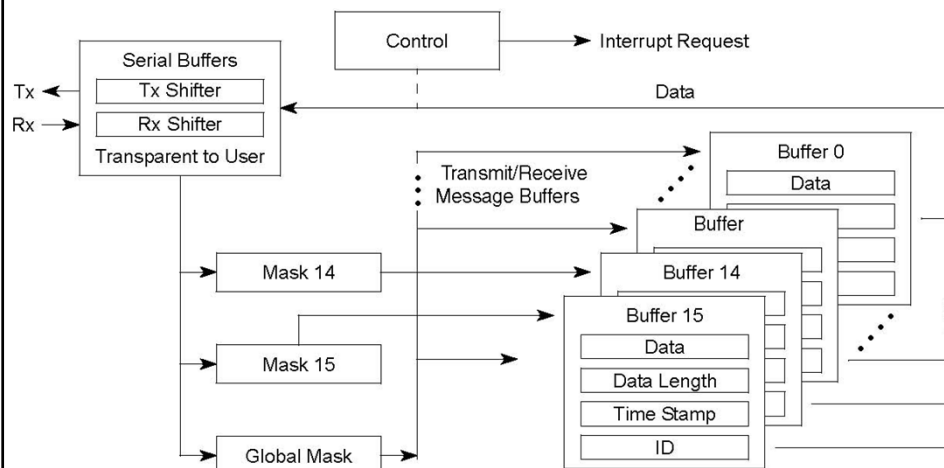
```
CANSendMsg( 0, 0x0cfe6cee, msg, 8, 0 );  
delay_ms(1000);  
dip204_clear_display();  
...
```

Layered structure
- abstraction!

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

9

Driver view illustrated through Coldfire CAN

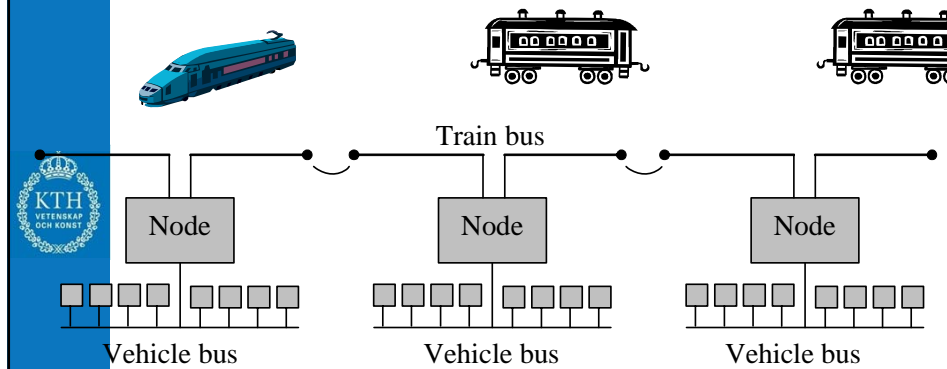


Courtesy of Freescale

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

10

Logical hardware topology: trains

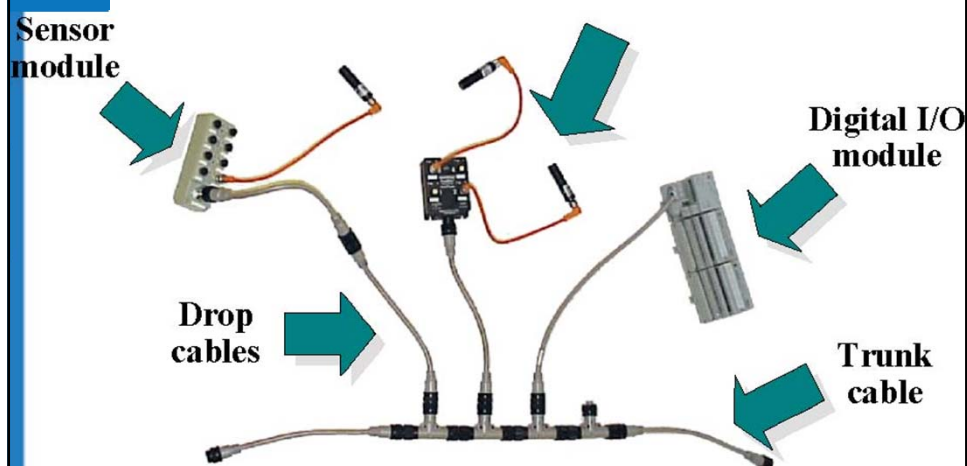


Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

Courtesy of Bombardier

11

Physical layout

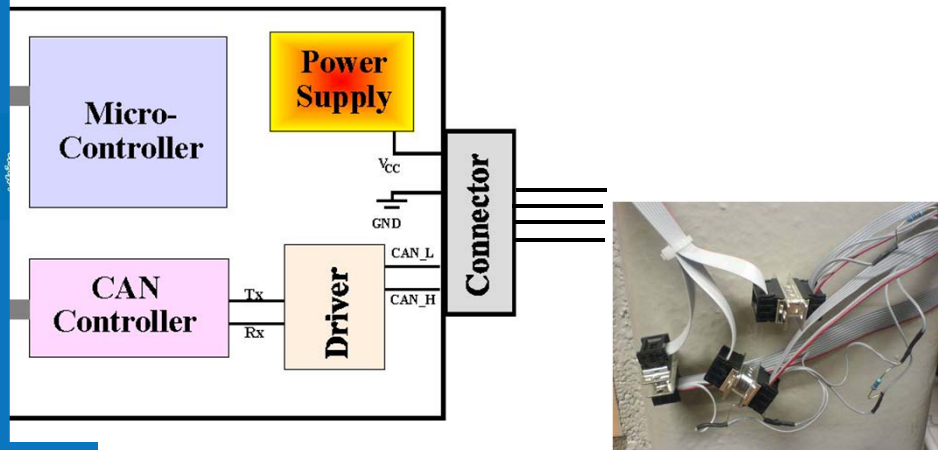


Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

Courtesy of Kvaser

12

Signal and cabling view



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

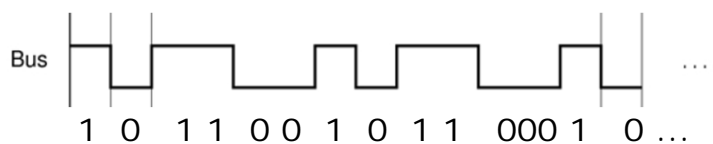
13

Data transmission over CAN: Application, datalink and physical level views

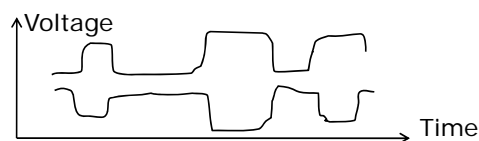
`CANSendMsg(0, 0x0cfe6cee, msg, 8, 0);`

And levels inbetween!

- Driver software



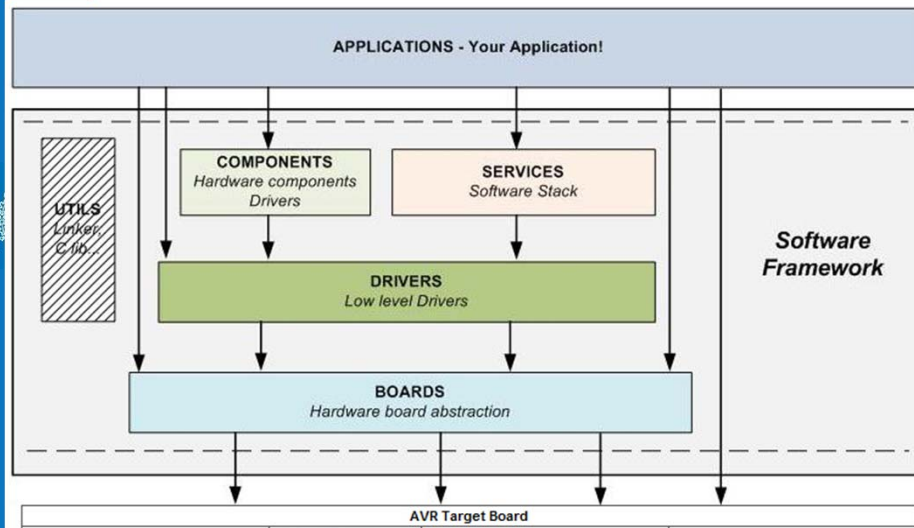
Diff.
Transmission
(Rs422)



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

14

Layering



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

15

Views of distributed systems

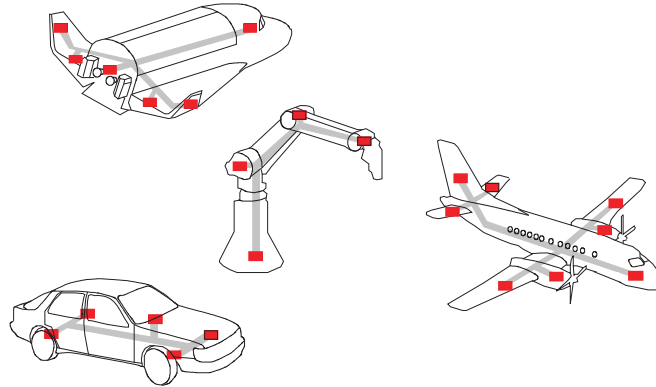
- Software allocation and roles of nodes
- Application program view
- Driver view
- Logical topology
- Physical layout
- Cables and wires
- Physical vs. Logical communication
 - Protocols



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

16

Why distributed systems?

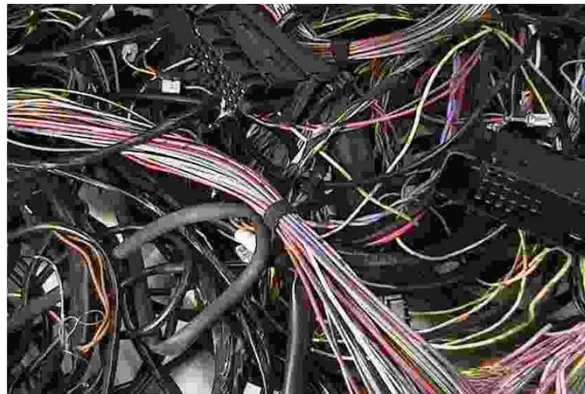


Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

17

Why not Analog Cabling?

- Traditional method: a μC is connected to distributed sensors and actuators by dedicated analog cables
- An example: 1900 wires, 2.2km long



Embedded systems, course, MF2042

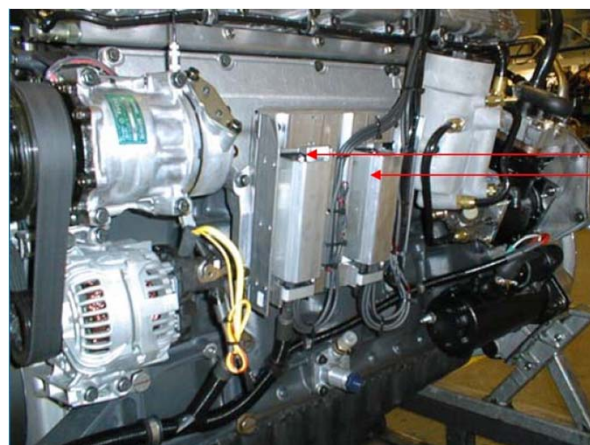
18

Reducing Cables with Network

- Digital signals – robustness against noise
 - Increase data integrity
- Reduce the cost of cables, connectors
- Increase system reliability (fewer connectors)
- Shared resources and information:
 - Sensors, actuators, computations, ...
- Modularity
- Performance
- Local intelligence

A Mechatronic Module

- Scania diesel engine and controller



ECU connectors
on top of the ECU

Interfaces:
- Mechanical
Connections
- CAN

Many advantages!!! What are the problems???

- (partial) failures
- Delays
- Consistency
- Organizational challenges
- How to deal with distributed functions?
- How to verify?



Distributed systems – failure modes

Nodes

- Omission
 - Fail-silent
 - Signalled failure
- Value
 - Wrong data sent
- Timing
 - E.g. too early, too late
- Interference
 - "Babbling idiot"
- Commission
 - Unintended message

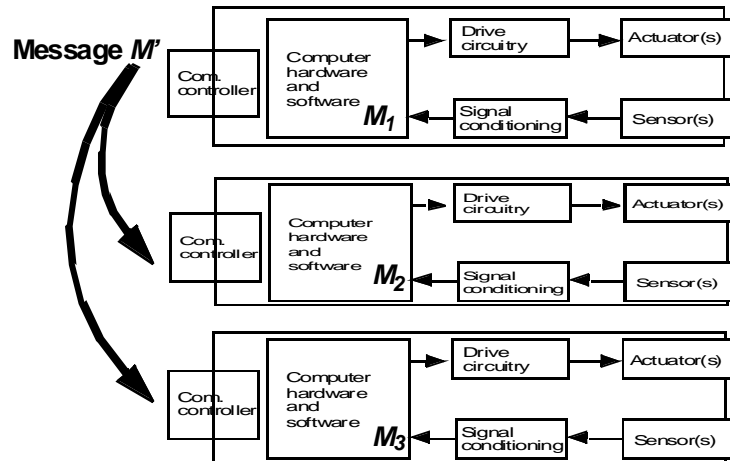
Network

- Omission
 - Transient or permanent
 - Permanent failure – special case: Network becomes partitioned!
- Value
 - Alteration of value
- Timing
 - Network failure causes delay/different delays
- Interference
- Commission

Attributes: Duration, Detectability, Symmetry



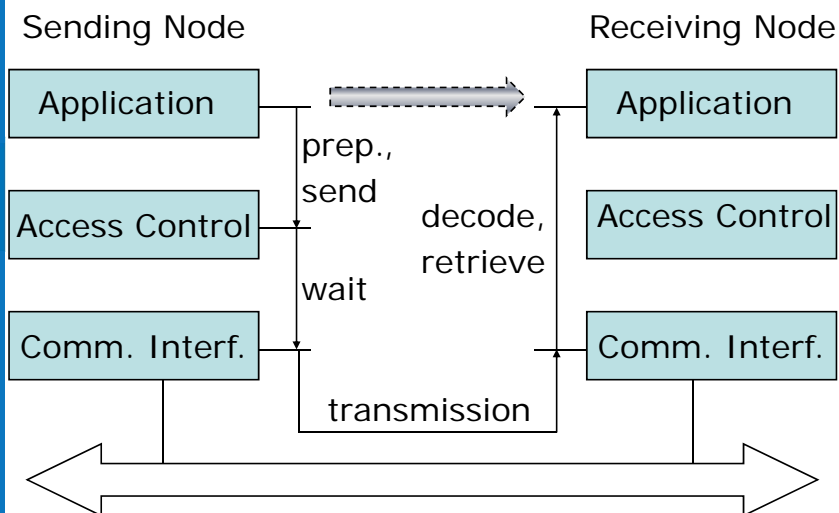
Distributed state variables, $\{M_1, M_2, M_3\}$
 \Rightarrow the system state; is it consistent?



When is the state required to be consistent?
 Consistency in the presence of faults?
 Trade-offs \rightarrow compare voice vs. file transfer!

23

Time delays in distributed systems

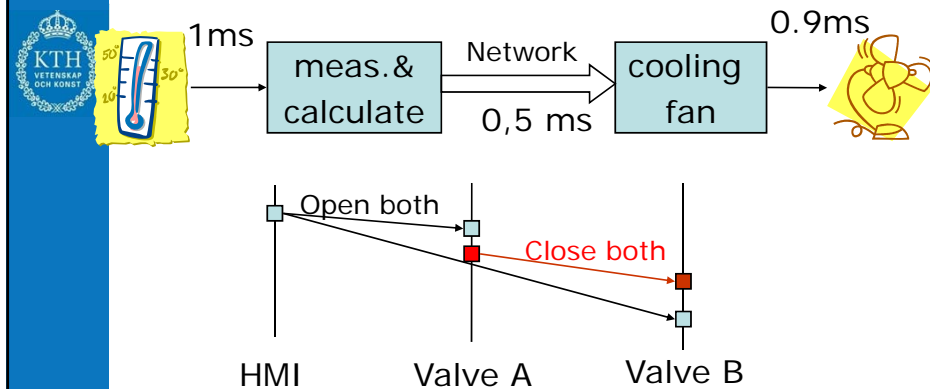


Martin Törnqvist, Sept. 2010 - Distributed systems
 Embedded systems course, MF2042

24

Delays and event ordering

- Incorrect event order possible



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

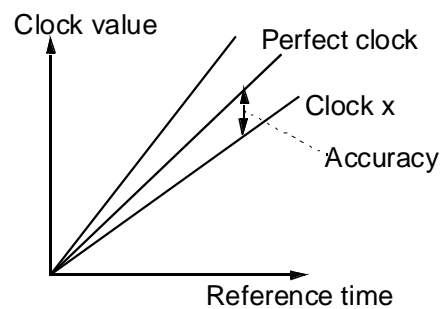
25

Clock synchronization

Local unsynchronized clocks:

- Granularity
- Drift rate, e.g. 10^{-7} s/s
- Clock drift rate \sim temp. & voltage applied to the crystal resonator

Free running clocks will drift apart



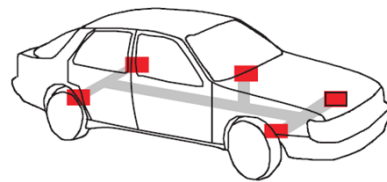
Clock synchronization

- Establishing an approximate global time within a system
- Internal clock synchronization \rightarrow precision of the global clock
 \rightarrow Master/slave vs. distributed clock synchronization
- Synchronization to an external time reference

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

26

Why are there so many types of networks, technologies and protocols??



Controller area network

CanOpen, Volcano, J1939, ...

LIN, Most, Flexray

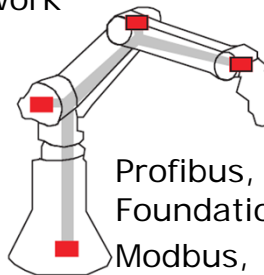
Train Communication Network

Ethernet, TCP/IP

MIL1553B

Wireless, short range
Cellular

BACnet, Lonworks



Profibus, Foundation Fieldbus, Modbus, ...

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

27

Outline

- Recap – why, what and how of distributed systems
 - Characteristics, Advantages, Problems
- Design issues for distributed embedded systems
 - Systems and application examples
 - Trends and research

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

28



Design issues

- Centralized or distributed system?
- Which type of distributed system?
 - Logical topology and physical layout
 - Decentralization of data, programs & control
 - Which communication protocol
- How to design and dimension them?
 - **The importance of the system architecture!**
 - ➔ Cost vs. Reliability vs. Performance vs. ...
 - Delays, synchronization and failures
 - Verification?
 - Who does what?

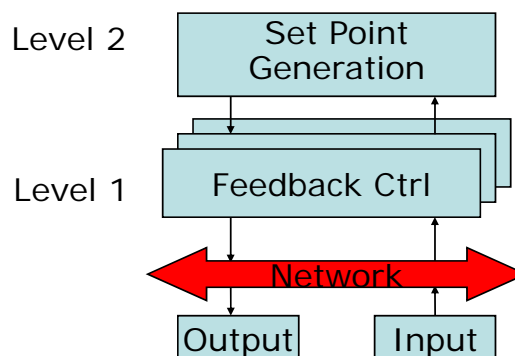
Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

29



Distributed I/O

- Common in process control
- Additional time-delay in control loop
- Care has to be taken to synchronize sensor inputs and multiple outputs

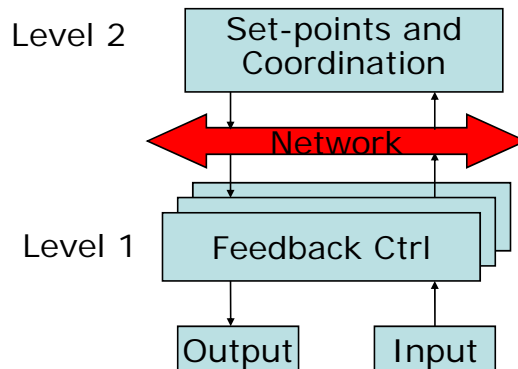


Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

30

Decentralized local control

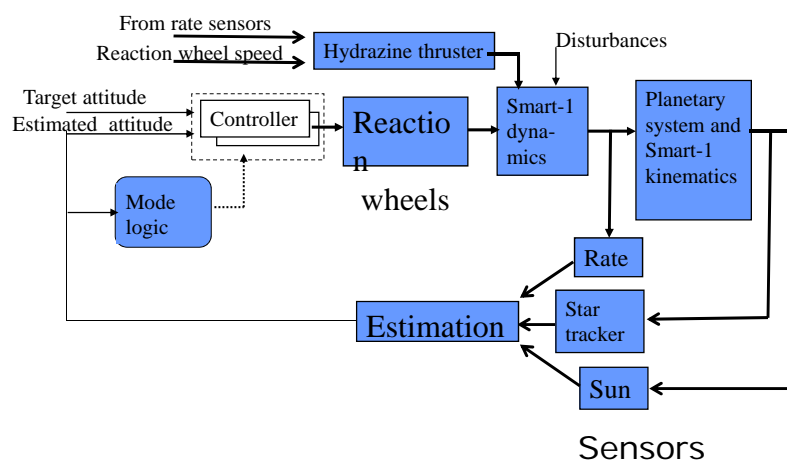
- Common in machine control and automotive
- Need for synchronization of level 1 loops with coordinator



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

31

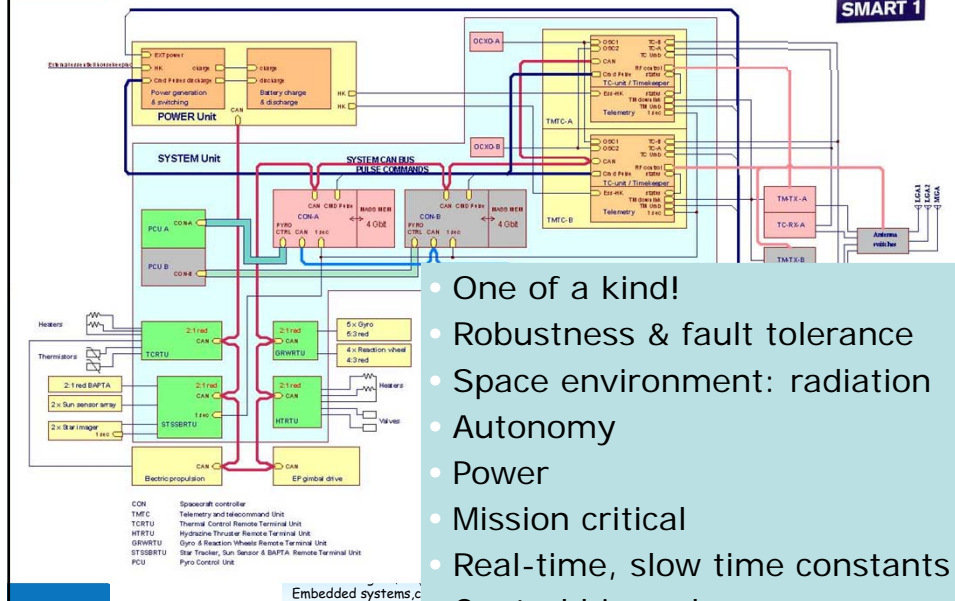
Attitude control system in the science mode of operation



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

32

Smart-1 block diagram



- One of a kind!
- Robustness & fault tolerance
- Space environment: radiation
- Autonomy
- Power
- Mission critical
- Real-time, slow time constants

Main on-board computer

TCS695E single chip ERC32 processor, 20 MHz clock

EEPROM 2 Mbyte EDAC protected


SRAM 3 Mbytes EDAC protected, needs SW scrubbing cycle < 1 minute,

MassMemory 512 Mbyte, EDAC protected, HW scrubbing

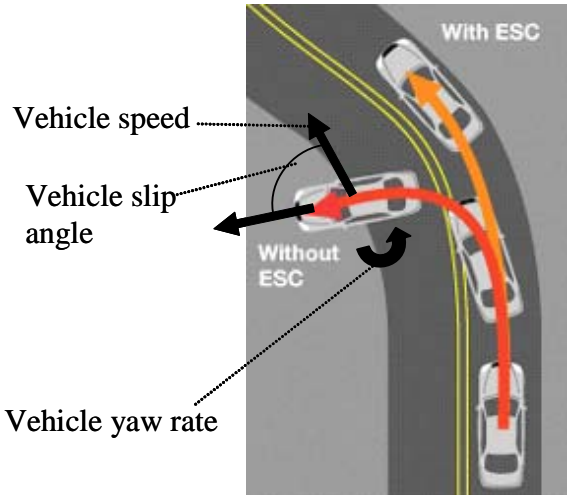
CAN controllers, 2 kBytes FIFO in reception path
→ Purchased VHDL code; radiation tolerant FPGA

Watchdog, 2 s timeout, can be disabled per pulse telecommand

JTAG I/F for EEPROM software upload and board HW checkout



Example application: stability control

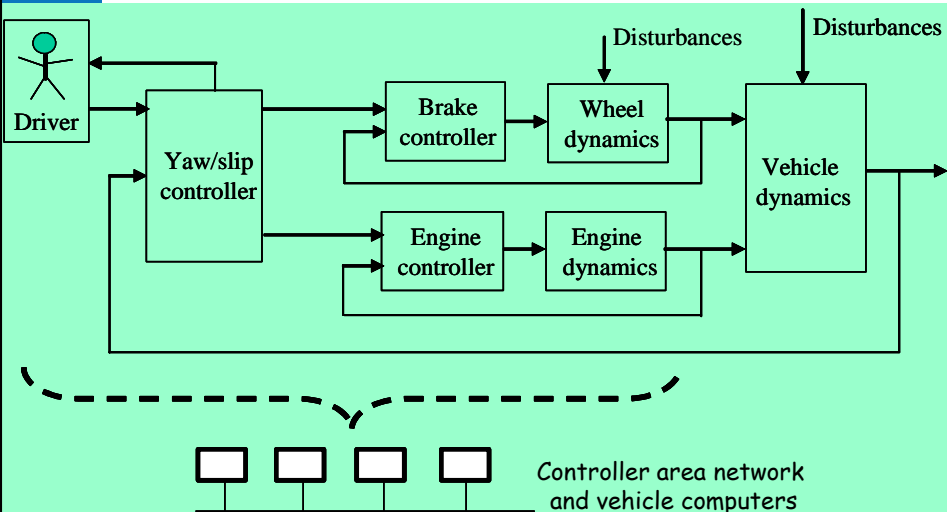


Source: ESC education - www.esceducation.org

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

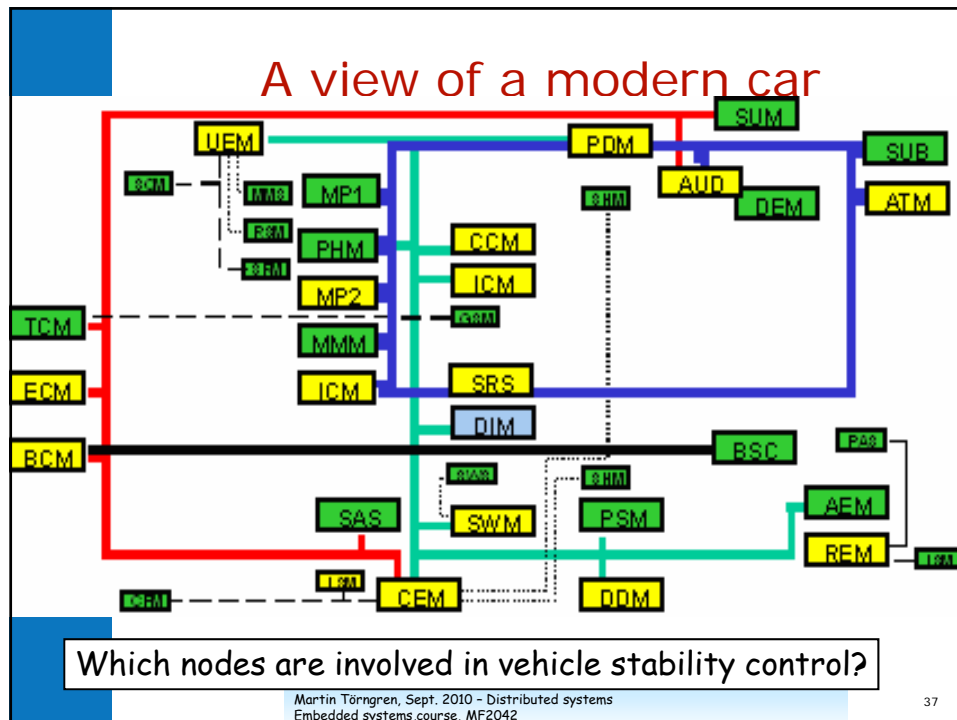
35

Example application: vehicle stability control



Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

36



Distributed systems perspective

- 70's – Distributed control in process automation
- 80's – Avionics
- 90's – Automotive
- External connectivity and wireless

Telecom; cellular telephony
Computer communication; Internet;
TCP/IP, wireless
Short range wireless communication;
Bluetooth, Zigbee, ...

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

38

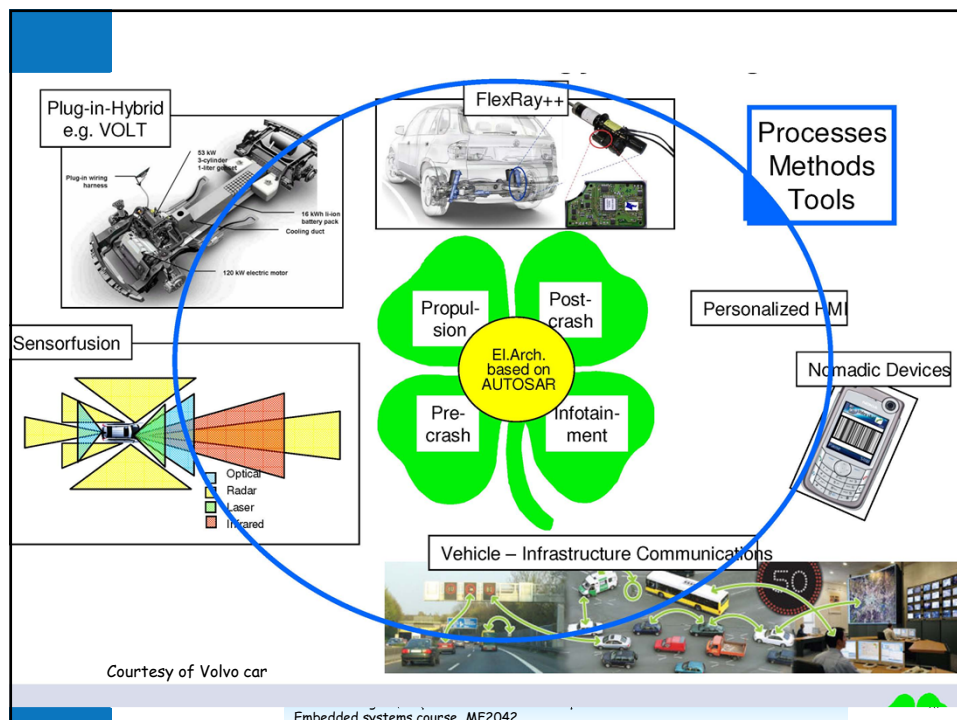
Trends

- Connectivity
- New (distributed) functions
- Convergence
- Standardized platforms
 - Middleware – for example Autosar
 - Protocols
- Product lines and model-based development
 - Reuse of SW and HW components
- Evolving electronics; multicore



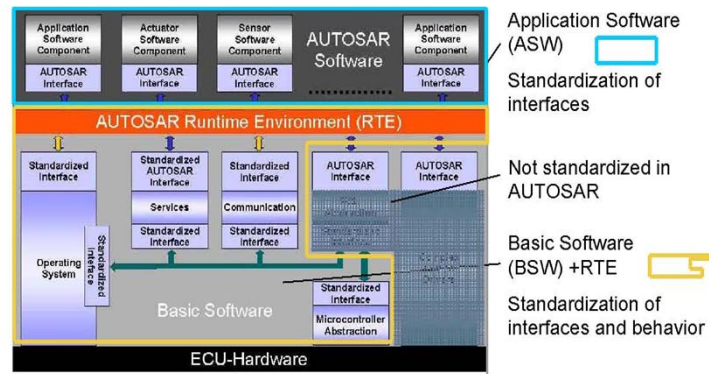
Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

39



Embedded systems, course, MF2042

Autosar – Automotive SW architecture



Autosar also defines information exchange

Objectives: Basic SW: Decoupling of Hardware and Application Software
Application SW: Relocation / Reuse of SW-Components between ECUs

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

41

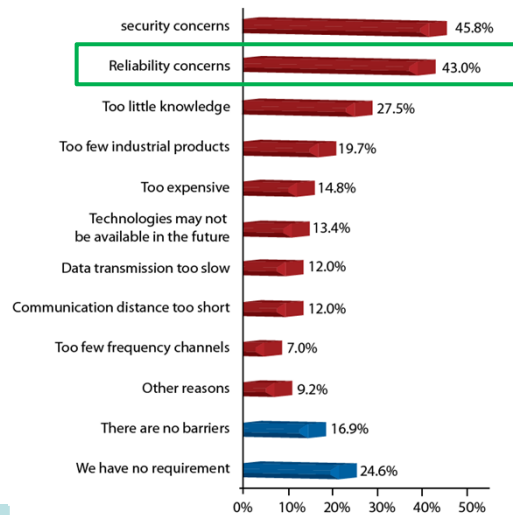
Benefits of wireless networking in industrial control

- Cost (wiring and installation)
- Flexibility
 - Less physical design limitations
 - More mobile equipment
 - Faster commissioning and reconfiguration
- Reliability
 - No cable wear and tear
 - No connector failure



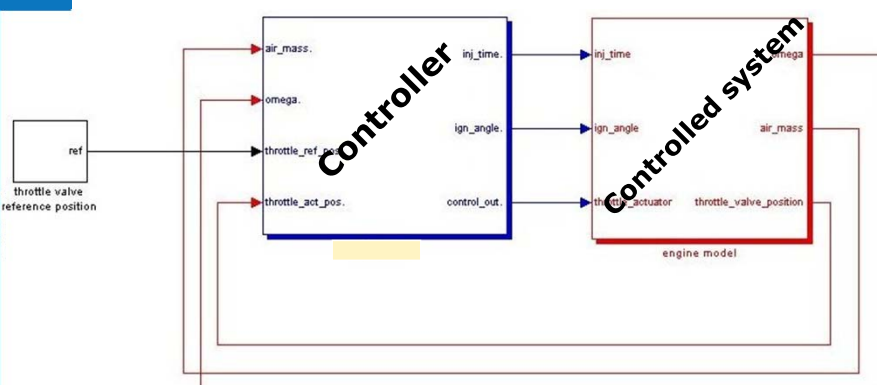
Courtesy of Karl-Henrik Johansson
Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems, course, MF2042

Barriers against wireless networking in industrial control



"Market pulse: Wireless in industrial systems: cautious enthusiasm", Industrial Embedded Systems, Winter 2006

Prototyping and different hardware



Hardware in the loop simulation

The diagram illustrates a Hardware in the Loop (HiL) simulation setup. On the left, a block labeled 'ref' (throttle valve reference position) provides a signal to the 'throttle_ref_pos.' input of the 'ECUcode' block. The 'ECUcode' block, which represents the engine control software, has several other inputs: 'air_mass.', 'omega.', and 'throttle_act_pos.'. It also has outputs: 'inj_time.', 'ign_angle.', and 'control_out.'. A thick black arrow points from the 'ECUcode' block to a physical 'ECU' (Electronic Control Unit) hardware component. The 'ECU' is connected to a 'dSPACE Simulator' hardware component. The 'dSPACE Simulator' is a rack-mounted unit with multiple slots, some of which contain modules. The 'ECU' and 'dSPACE Simulator' are connected via a bidirectional communication link, represented by a double-headed arrow. The 'dSPACE Simulator' also has a connection to the 'ref' block, completing the feedback loop.

Martin Törngren, Sept. 2010 - Distributed systems
Embedded systems course, MF2042

45

Concept – for a network test

The diagram illustrates a network test concept for an overall model. At the top center is the **Central unit**, which is connected to a **Control computer** (monitor and keyboard) on the left via a dashed line. To the right of the central unit is a box representing the **Driving cockpit** (showing a steering wheel and dashboard) and a **customer** (showing a car icon). Below the central unit, three server racks are shown, each connected to the central unit. The left rack is labeled **I/O-Unit engine** and has a circuit board icon. The middle rack is labeled **I/O-Unit ESP / xxx** and has two circuit board icons. The right rack is labeled **I/O-Unit transmission** and has a circuit board icon. The entire setup is labeled **Overall model**. Three blue dots are at the bottom right.

Control computer

Central unit

Driving cockpit

customer

Overall model

I/O-Unit engine

I/O-Unit ESP / xxx

I/O-Unit transmission

I/O - program

Martin Törngren, Sept. 2010 - Distributed systems Embedded systems course, MF2042

46

Summary

- Why, what & how of distributed systems
 - Remember the exercise!
- Design issues for distributed embedded systems
 - Architecture and verification
 - Dealing with complexity
 - Trends:
 - Increasing connectivity and convergence
 - Standardization

