Vehicle Networks and Control Area Networks (CAN)
Intelligent Transportation Systems

Platooning

Adaptive Gantry Signs & In-vehicle signage

Assisted Driving

Intelligent Portable Infrastructure

Autonomous Driving

Intermodal transportation
ITS Objectives

• Safety:
 ➢ Mitigation of accident severity (passive safety)
 ➢ Prevention of accidents (active safety)
 ➢ Avoidance of hazardous situations (preventive safety)

• Efficiency: Reduction of travel times, fuel consumption, CO2 emission, noise emission

• Infotainment/Comfort: Increasing comfort of driving, Additional information services

• Monetary: Cost reduction (e.g. less sensors, less road infrastructure maintenance); competitive edge
Intra-vehicle communications

- In the past: VW Käfer (1950)
  - 50 m copper wires
  - 0 Electronic Control Units (ECUs)
  - 3860 m copper wires
  - 45 networked Electronic Control Units (ECUs) / 61 ECUs total
  - 11,136 electrical parts in total
  - 3 different bus networks
Vehicle Electronics

- Total complexity of vehicle electronics: Customer demand driven

- Complexity of SW:
  - More integration takes place in SW
  - How do we manage safety, cost, ...?

- Complexity of HW (number of ECUs):
  - No space in vehicle for more ECUs
  - Fewer, larger ECUs more cost effective (Moore's law)

Source: Volvo 2002

Cost of Electronic Embedded systems = \begin{cases} 
1\% & (1980) \\
20\% & (2005) \\
40\% & (2015) 
\end{cases}

ECU: Electronic Control Unit
Intra-vehicle: Degree of Networking

- Electronic fuel injection
- Cruise control
- Central locking system

- Electronic gearbox control
- Electronic air conditioning
- Anti-lock Blocking System (ABS)
- Anti-Slip Control (ASC)

- Navigation system
- RDS/TMC
- Adaptive Cruise Control (ACC)
- Electronic Stability Control (ESC)
- Active Body Control (ABC)
- Airbags
- Park Distance Control

- ACC Stop&Go
- Park Assistant
- Adaptive Headlights
- Night Vision Systems
- Hands-free equipment
- Steer/Brake by Wire
- Lane keeping assistant
- Lane departure Warning
- Personalization
- SW Update
- Force Feedback Pedal
- ...
Network Evolution

- Stand alone ECUs
  - No networking
- Directly connected ECU (partially mesh)
- Star topology with central gateway
- Partitioned bus topology with interconnecting gateway
In-vehicle bus systems

- Volkswagen Golf V

- Engine-CAN (500 kps)
- 2 private CAN (500 kps)
- Instrument-CAN (500 kps)
- Infotainment-CAN (100 kps)
- Convenience-CAN (100 kps)
- 1 Diagnostics-CAN (500 kps)
- 2 LIN-Networks
- K-Wire
CAN History

• Development of CAN mainly driven by Mercedes-Benz for networking of versatile Electronic Control Units (ECUs) with the following requirements:
  ➢ Error-resistance to cope with strong electro-magnet interference
  ➢ Prioritized real-time capabilities with short latency (e.g. for safety critical applications)
  ➢ Fast data rate (Class C network: 125 kbit/s – 1 Mbit/s)
  ➢ Expandability for versatile nodes
  ➢ Cost-effectiveness for wires and nodes

• These requirements also hold for various other application fields in aviation and maritime industry, industrial and home automation, consumer electronics

• Widespread distribution of CAN nowadays
CAN Applications

• Automotive, aviation, space, maritime industry
  - Car, truck, bus; Airplanes; Rockets, space shuttles; Ships
• Medical equipment
  - X-Ray, Electro-Cardiograms (ECG)
• Industrial and home automation
  - Production machines
  - Lifts and escalators
  - Shutter, heating, light control
• Household appliances: washer, dryer
• Consumer electronics: model railway
Number of CAN Nodes (in millions)

Produced by Motorola, Philips, Intel, Infineon, etc.
## CAN in ISO/OSI Reference Model

<table>
<thead>
<tr>
<th>No. of layer</th>
<th>ISO/OSI ref model</th>
<th>CAN protocol specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Application</td>
<td>Application specific</td>
</tr>
<tr>
<td>6</td>
<td>Presentation</td>
<td>Optional: Higher Layer Protocols (HLP)</td>
</tr>
<tr>
<td>5</td>
<td>Session</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Transport</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Network</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Data Link</td>
<td>CAN protocol (with free choice of medium)</td>
</tr>
<tr>
<td>1</td>
<td>Physical</td>
<td></td>
</tr>
</tbody>
</table>
CAN PHY Layer
CAN Hardware

- Bus topology to reduce the number of wires
- Flexible in choosing transmission medium
- Automotive CAN according to ISO 11898-2/3 uses twisted pair with differential voltages on a bus topology (tolerant to single wire disturbance)
- Bus must be terminated with 120 Ω to:
  - remove signal reflections at the end of the bus
  - ensure the bus gets correct DC levels
- Max 30 connected nodes
The signal has to propagate to the most remote node and back again (round trip) before the bit is sampled. Bus length and data rate are correlated.

\[
\text{max. bus length} < \frac{\text{signal velocity} \times \text{nom. bittime}}{2}
\]

<table>
<thead>
<tr>
<th>Data rate (kbit/s)</th>
<th>Max. bus length (m)</th>
<th>Nominal Bit-Time (μs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>40</td>
<td>1</td>
</tr>
<tr>
<td>500</td>
<td>130</td>
<td>2</td>
</tr>
<tr>
<td>250</td>
<td>270</td>
<td>4</td>
</tr>
<tr>
<td>125</td>
<td>530</td>
<td>8</td>
</tr>
<tr>
<td>50</td>
<td>1300</td>
<td>20</td>
</tr>
<tr>
<td>20</td>
<td>3300</td>
<td>50</td>
</tr>
<tr>
<td>10</td>
<td>6700</td>
<td>100</td>
</tr>
</tbody>
</table>

*approximation*
Coding

- Dominant and recessive coding:
  - Dominant: logic “0”
  - Recessive: logic “1”

- If more than one stations send a signal, the bus takes dominant state if at least one station sends a dominant signal. The bus takes recessive state only if both sends a recessive signal.

- Each node transmit and listen at the same time
Synchronization

- No global time source, no dedicated clock signal
- Synchronization by edge detection in data signal
- Bit length known due to uniform clock rate for every node (e.g. 2 μs for 500 kbit/s)
- Hard synchronization with first recessive-to-dominant edge (=dominant Start Of Frame (SOF)bit) after bus idle
- Continuous re-synchronization at every recessive-to-dominant edge transition
Error Detection

- At sender side: monitor bus during transmission
- At receiver side: detect errors by checking whether the bit sequence is in adherence with the bit stuffing rule
CAN Data Link Layer
CAN Frame

- SOF (start-of-frame) bit -- a dominant (logic 0) bit
- Arbitration ID -- identifies the message and indicates the message's priority
  - Standard format: 11 bit ID -> 47~55 bit data frame (+stuff bits)
  - Extended format: 29 bit ID -> 67-75 bit data frame (+stuff bits)
  - Every node on the bus receives all messages and filters according to ID
- IDE (identifier extension) bit -- allows differentiation between standard and extended frames
CAN Frame (cont'd)

- RTR (remote transmission request) bit -- serves to differentiate a remote frame from a data frame. A dominant (logic 0) RTR bit indicates a data frame. A recessive (logic 1) RTR bit indicates a remote frame.

- r0: reserved

- DLC (data length code) -- indicates the number of bytes the data field contains (0-8 bytes requires 4 bit length field)

- Data Field -- contains 0-8 bytes of data

- CRC -- cyclic redundancy check for error detection
  - 15 bit CRC with generator polynomial $x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$
  - 1 bit CRC delimiter: single (always) recessive bit
CAN Frame (cont'd)

• ACK (ACKnowledgement) slot -- The transmitting node checks for the presence of the ACK bit on the bus and reattempts transmission if no acknowledge is detected
  - 1 bit ACK slot: dominant overwriting
  - 1 bit ACK delimiter: single (always) recessive bit
• CAN Signal – an individual piece of data contained within the CAN frame data field, containing up to 8 bytes of data
• End of Frame: 7 recessive bits
• Bit stuffing: sender inserts complementary bit (stuff bit) after 5 successive bits of same polarity.
Contestation-Free MAC

- **CSMA/CR (Collision Resolution):**
  - If two devices transmit simultaneously, the one with smaller arbitration ID gets the higher priority to transmit.
CSMA/CR

- Advantages:
  - Allow different priority
  - High bandwidth utilization

- Limitations?
Error Handling
and
Error Confinement
Error Handling

- Error frames are sent after an error is detected

- Error flag:
  - Active error flag: 6 consecutive dominant bits (breaking the stuffing rule!)
  - Passive error flag: 6 recessive bits (can be squashed by error frames sent by other nodes!)

- Error delimiter: 8 recessive bits

- Majority vote to detect “perpetrator”:
  - Majority of nodes send error frame $\bar{Y}$ transmitter is perpetrator
  - Majority of nodes send no error frame $\bar{Y}_2$ receiver is perpetrator
Error Confinement

• Every node stores two kinds of errors:
  - Transmit error counter (TEC)
  - Receive error counter (REC)

• What a node does if the node is in one of the following states:
  - Error active: Transmission of Active Error Flags (dominant) if error is detected by this node
  - Error passive: Transmission of Passive Error Flags (recessive) if error is detected by this node
  - Bus off: No transmission on the bus
Residual Error Probability

• Example:
  - 1 Bit error every 0.7s
  - Bit rate: 500 kBit/s
  - Operation of 8 hours/day and 365 days/year
  - 1 undetected error in 1000 years