Vehicle Networks

Controller Area Network (CAN)

Univ.-Prof. Dr. Thomas Strang, Dipl.-Inform. Matthias Röckl

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

STI · INNSBRUCK

universität innsbruck

Outline

- Introduction
  - Design requirements
  - History and application fields
  - Recapitulation: Communications basics
- CAN Physical Layer
- CAN Data Link Layer
- CAN Error Handling & Error Confinement

Deutsches Zentrum
für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

Introduction
Requirements for automotive communications

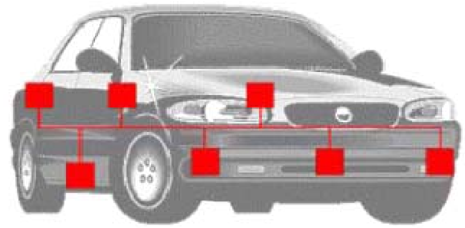
- The development of CAN was mainly driven by Mercedes-Benz for networking of versatile *Electronic Control Units (ECUs)* with the following requirements:
  - **Error-resistance** to cope with strong electro-magnet interference
  - **Prioritized real-time** capabilities with short latency (e.g. for safety 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

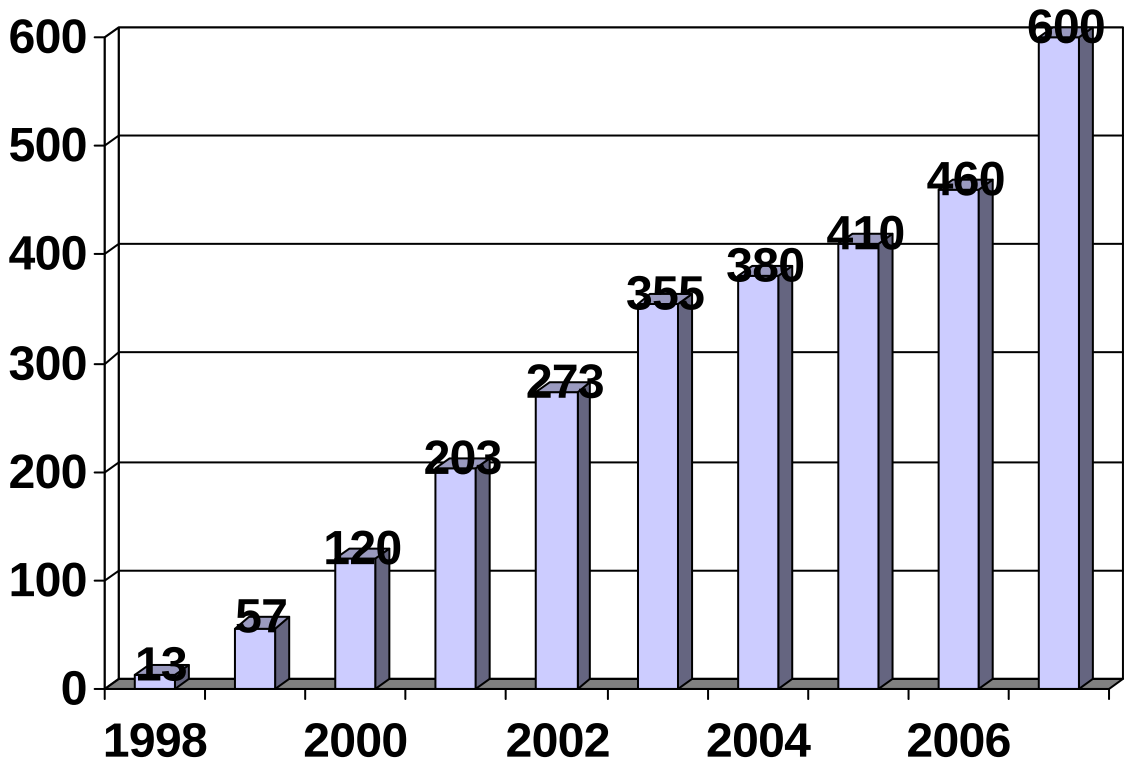
  ➔ Widespread distribution of CAN nowadays

Introduction
Application Fields

- Automotive, aviation, space, maritime industry
  - Cars, trucks, busses
  - 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
  - Washers, Dryers
  - Coffee machines (“Café au CANopen”)
- Consumer electronics
  - Model railway



Introduction
Number of CAN nodes in millions



Produced by Motorola, Philips, Intel, Infineon, etc.

Introduction
History

1985 Start of development of CAN at Robert Bosch GmbH

1986 V1.0 specification of CAN

1991 Specifications of the extended CAN2.0 protocol

- Part 2.0A – 11-bit identifier
- Part 2.0B – 29-bit identifier (extended frame format)

1992 CAN in Automation (CiA) established as the international users and manufacturers group

1993 First car, a Mercedes S-class, equipped with CAN

1994 First standardization at ISO is completed

1998 Development phase of time-triggered CAN (TTCAN) networks

1999 Explosion of CAN-linked equipment in all motor vehicle and industrial applications

Introduction
CAN in ISO/OSI Reference Model

No. of layer	ISO/OSI ref model	CAN protocol specification
7	*Application*	Application specific
6	*Presentation*	Optional: Higher Layer Protocols (HLP)
5	*Session*	
4	*Transport*	
3	*Network*	
2	*Data Link*	**CAN protocol** (with free choice of medium)
1	*Physical*	

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

Introduction
Standards

SAE J2284
High Speed CAN (HSC) For Passenger Vehicle Applications

SAE J1939
Recommended Practise for Control and Communications Network (Class C) on Truck and Bus Applications

ISO 11898 Road vehicles – Controller Area Network (CAN)

Extensions to Data Link Layer

- **ISO 11898-4** Time-triggered CAN
- **ISO 11898-5** relates to high-speed CAN and low-power applications

Data Link Layer

- **ISO 11898-1** Data link layer and physical signalling

According to original Bosch specification CAN2.0A and CAN2.0B

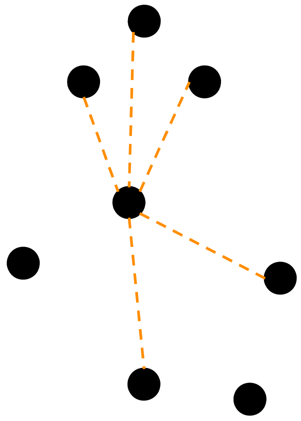
Physical Layer

- **ISO 11898-2** High-speed medium access unit
- **ISO 11898-3** Low-speed fault-tolerant medium-dependent interface

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

Introduction
PHY: Overview

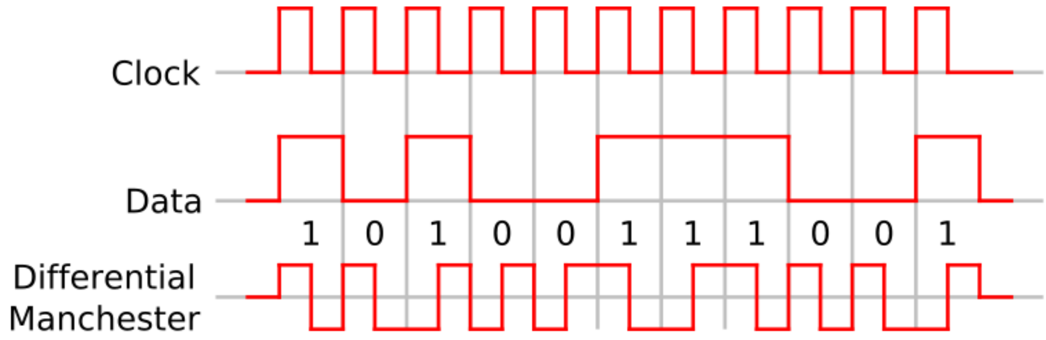
- Responsible for *bit-by-bit* translation of data into/from hardware-specific operations
- Defines electrical, mechanical, and procedural interface to the transmission medium (wire, air, optical fibre etc.)
- Main functions:
  - Line coding: (De-)modulation of the digital signal into/from the amplitude- and time-discrete signal
  - Channel coding: Insertion of redundant information for error detection/correction
  - Synchronisation: Determination of a common time basis



RECAP

Introduction
PHY: Line Coding

- Return-to-zero (RZ) coding (e.g. IrDA Serial Infrared)
- Non-Return-to-Zero (NRZ) coding (e.g. RS-232)
- Manchester coding (e.g. Ethernet)
- 4B3T (e.g. ISDN)



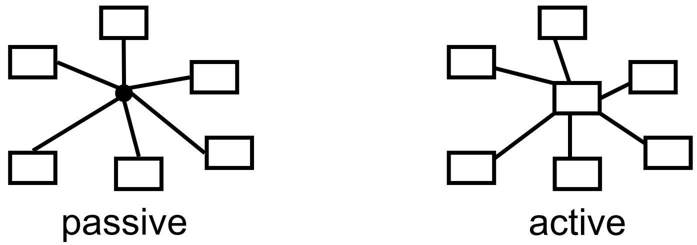
(Unipolar NRZ)

RECAP

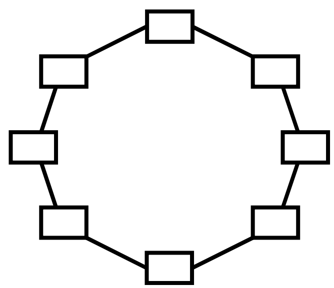
DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

Introduction
PHY: Network topologies

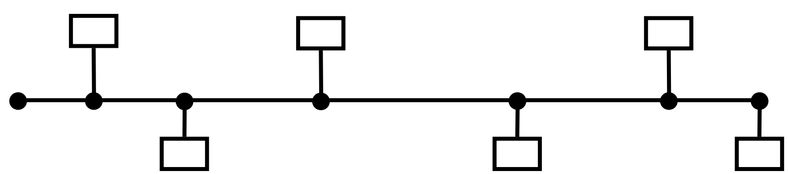
- Star topology



- Ring topology



- Bus topology



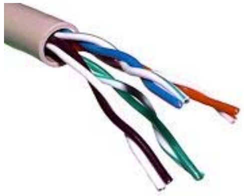
RECAP

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

Introduction
PHY: Wiring

- One-wire: single feed line – return line over ground connection

- Two-wire:
  - Unshielded Twisted pair (UTP)
  - Shielded Twisted pair (STP)

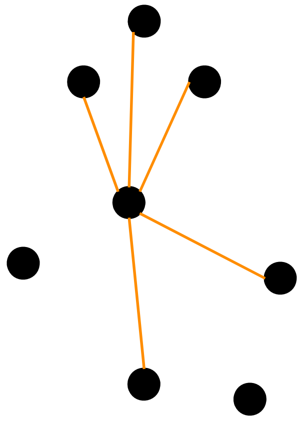


- Coaxial:
  Inner conductor shielded with fine woven wires or metallic foil

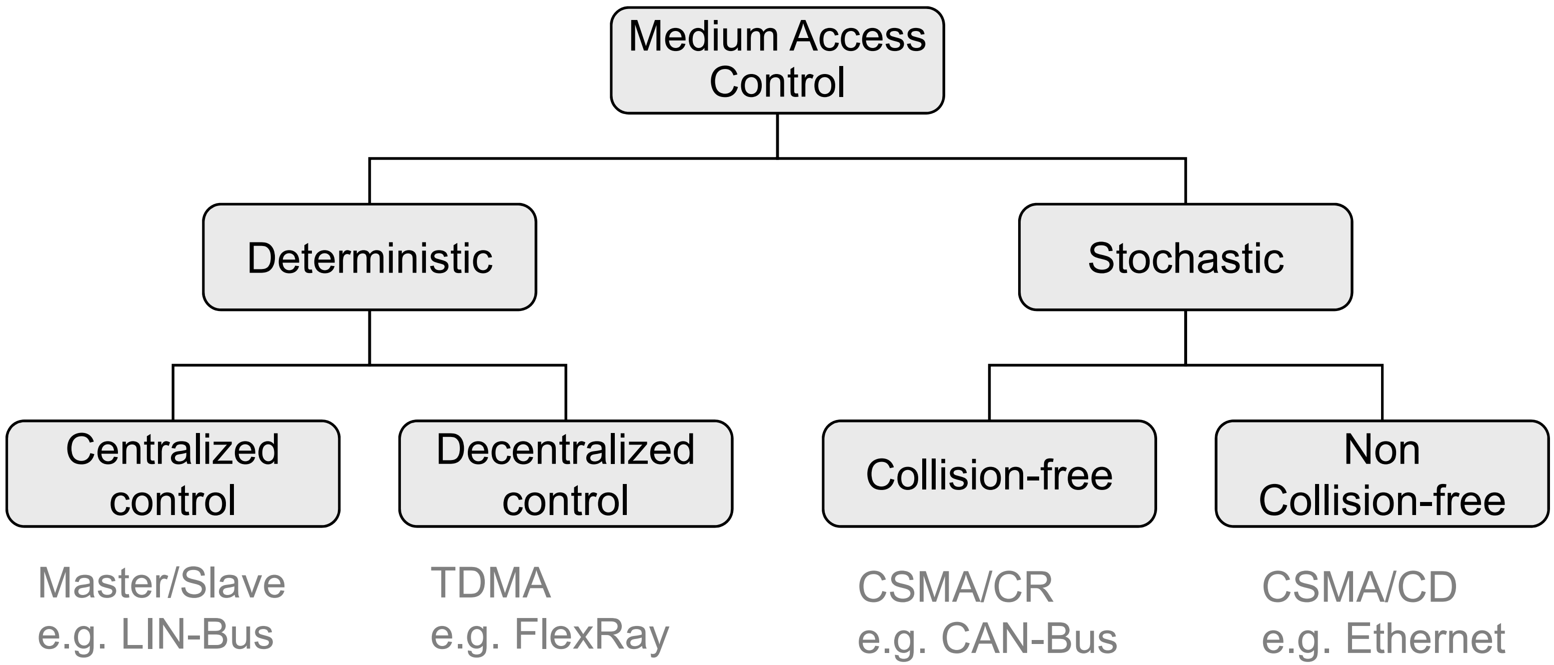
- Optical: Transmission of light through fiber
  - Single-mode fiber (SMF): only single *ray of light* can be carried
  - Multi-mode fiber (MMF): support of more than one propagation mode; reflection on fiber border

Introduction
DL: Overview

- Responsible for *Point-to-Point* / *Point-to-Multipoint* transfer of data
- Main functions:
  - Framing (i.e. sequence of bits)
  - Addressing
  - Error protection/detection/correction
  - Flow control
  - Medium access
- Sublayers:
  - Medium Access Control (MAC)
    e.g. CSMA/CD, CSMA/CA, …
  - Logical link control (LLC)
    e.g. IEEE 802.2



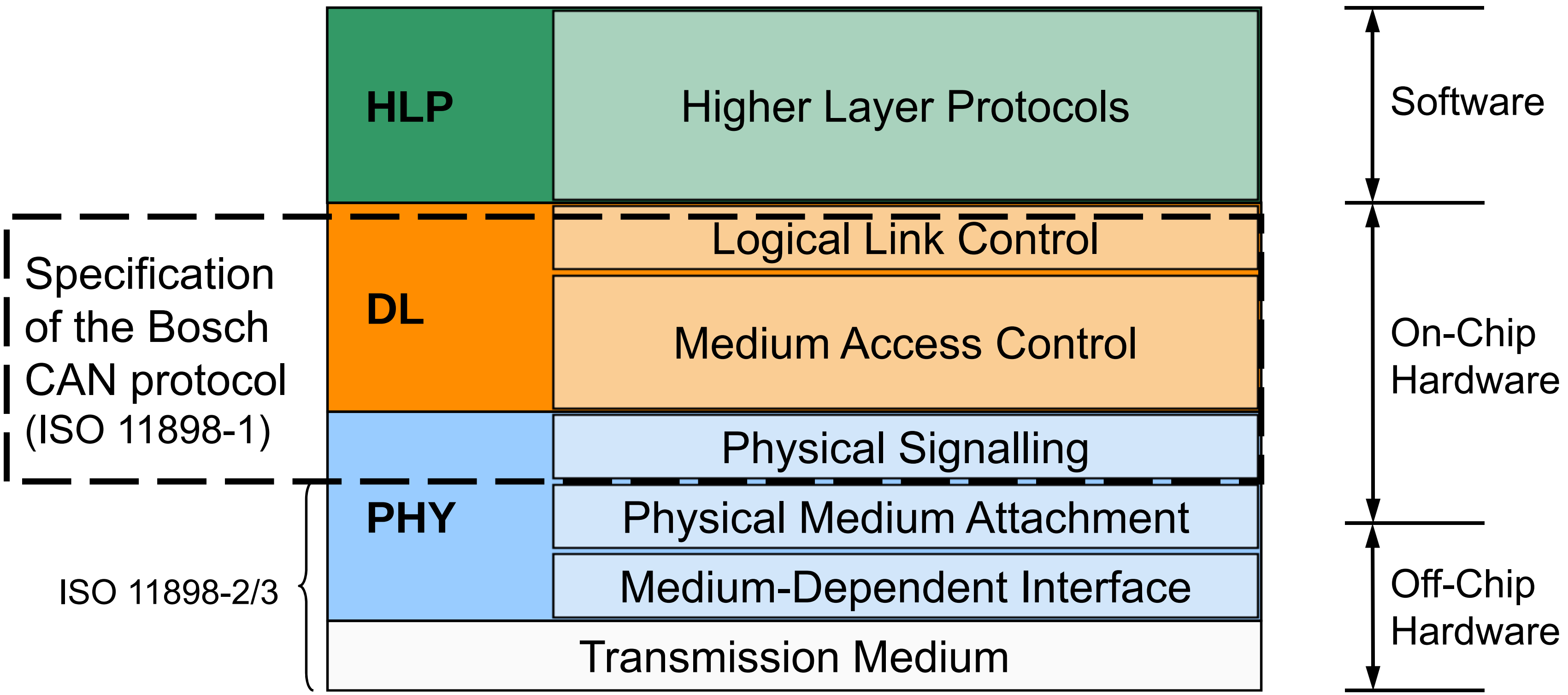
Introduction
DL: Medium Access Control



RECAP

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Protocol Specification



DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

Controller Area Network (CAN)

Physical Layer

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

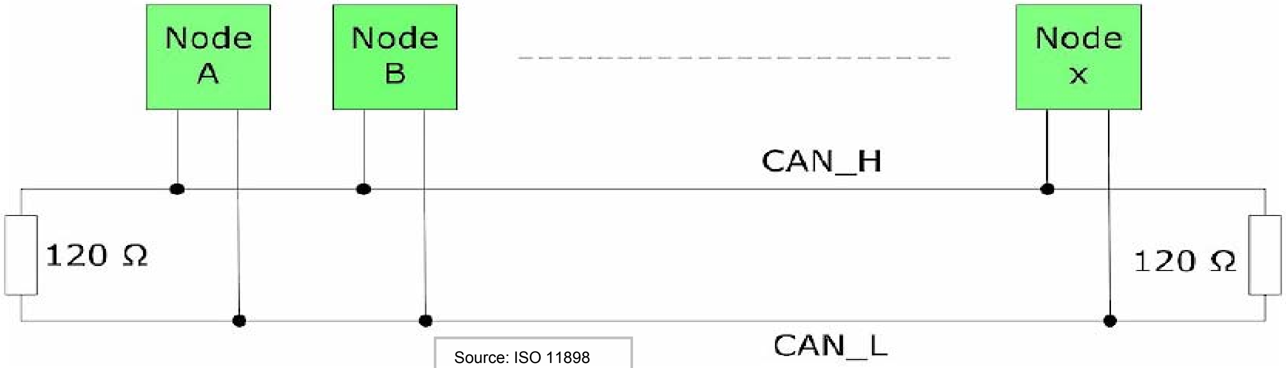
CAN Physical Layer
Overview

- Physical Line Signalling (PLS)
  - Bit Coding, Bit Timing
  - Synchronization
  - Error detection
- Physical Medium Attachment (PMA)
  - Transmitter/Receiver Specification, i.e.
    Characteristics of the command stages and timing stages
- Medium-Dependent Interface (MDI)
  - Bus Cable, Bus Termination Networks, Connectors, etc.

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Physical Layer
Hardware

- *n* nodes would require O(n^2) wires for full meshed networking (up to 100 ECUs in modern cars)
- Bus topology only requires a single bus with *n* branches: O(*n*)
- Pure CAN does not specify 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



Source: ISO 11898

Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Physical Layer
Data rates vs. bus length

- 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
- Signal propagation of information is limited to approximately 0.4-0.7 times the speed of light (≈0.1-0.2 m/ns);
  depends on cable and node impedance
- A nominal bit time can be calculated at each node
- Further factors: transceiver delay, sample time tolerance

$$\boxed{\text{max. bus length} < \frac{\text{signal velocity} * \text{nom. bittime}}{2}}$$

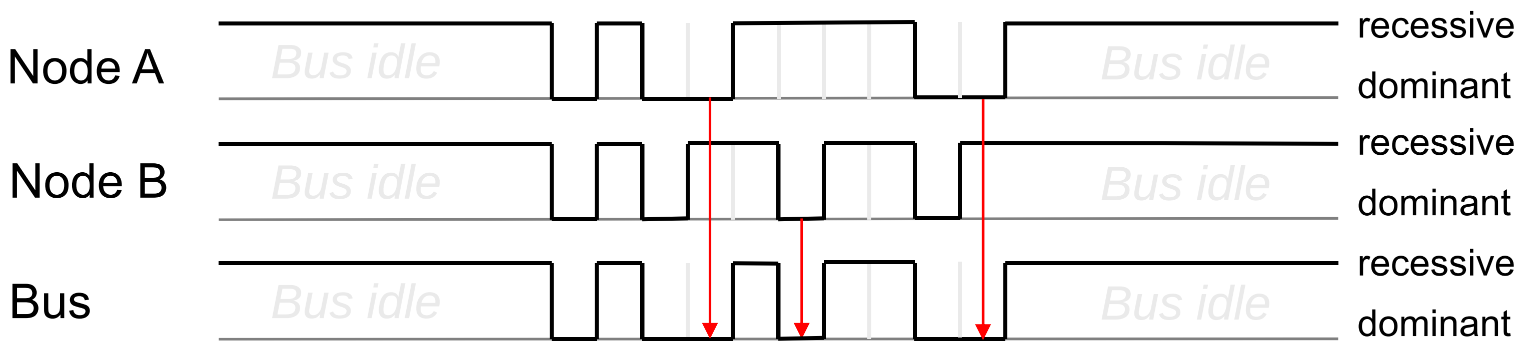
Data rate	Max. bus length*	Nominal Bit-Time
1000 kbit/s	40 m	1µs
500 kbit/s	130 m	2µs
250 kbit/s	270 m	4µs
125 kbit/s	530 m	8µs
50 kbit/s	1300 m	20µs
20 kbit/s	3300 m	50µs
10 kbit/s	6700 m	100µs

*approximation

CAN Physical Layer
Bus access

Bus		Node B	
		dominant	recessive
Node A	dominant	*dominant*	*dominant*
Node A	recessive	*dominant*	*recessive*

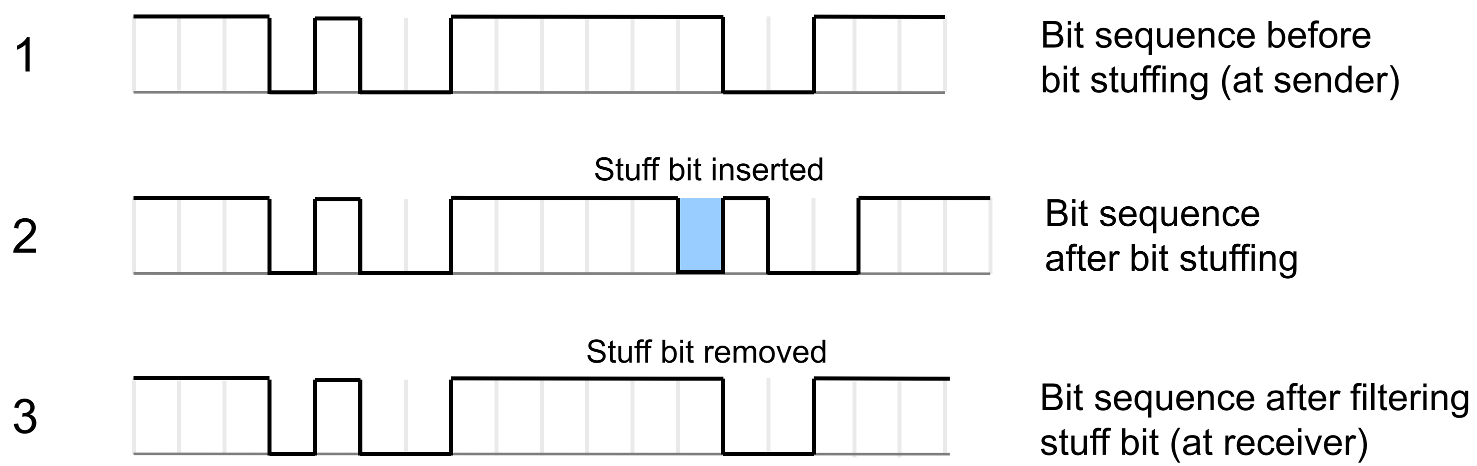
- Dominant and recessive coding:
  - Dominant: logical "0"
  - Recessive: logical "1"
  - If more than one station sends a signal, the whole bus takes the dominant state if at least one station sends a dominant signal. The bus takes the recessive state only when all stations send recessive signals.
- Each node needs to be able to transmit and listen at the same time
  - When a node transmits dominant, it always hears dominant (except for transmission errors)
  - When a node transmits recessive and hears dominant, a bus conflict with another node occurred



DLR
Deutsches Zentrum
für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

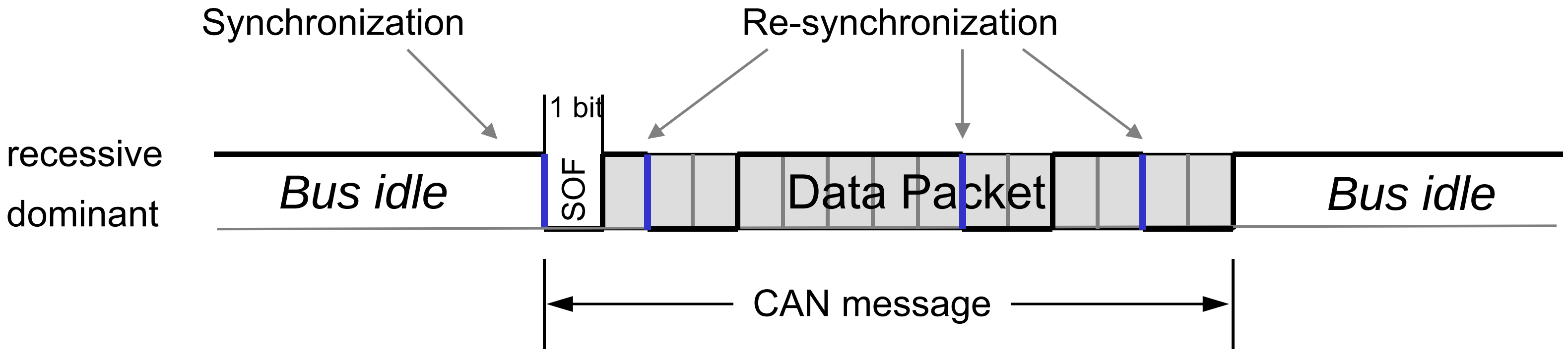
CAN Physical Layer
Line Coding

- Line coding: Non-Return-to-Zero (NRZ) with bit stuffing
  - Standard NRZ has not enough signal edges for synchronisation
    ➔ Bit stuffing: after 5 consecutive bits with same polarity (dominant or recessive), a bit with complementary polarity will be inserted
  - Receiver filters the complementary bit ➔ not visible to higher layers



CAN Physical Layer
Synchronization

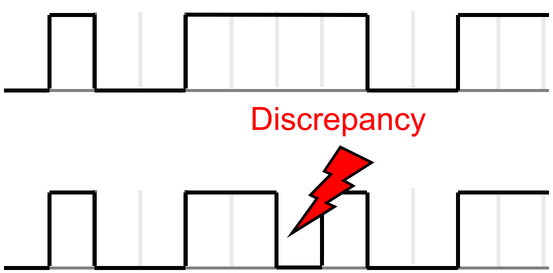
- No global time source, no dedicated clock signal
- Synchronization by edge detection in data signal
- Bit length known due to uniform clock rate for every node (e.g. 2 μs for 500 kbit/s)
- Hard synchronization with first recessive-to-dominant edge (=dominant *Start Of Frame (SOF)* bit) after bus idle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



CAN Physical Layer
Error detection

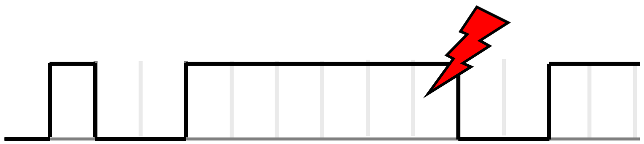
- At sender side:
  - **Bit errors**: Senders monitor the bus during transmission. Discrepancies between sent and monitored bit are interpreted as transmission error
- At receiver side:
  - **Stuff error**: Receiving nodes can detect errors by checking whether the bit sequence is in adherence with the bit stuffing rule

Sender side



Bit error

Receiver side



Stuff error

Controller Area Network (CAN)

Data Link Layer

Deutsches Zentrum
für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

CAN Data Link Layer
Overview

- Sublayer: Medium Access Control (MAC)
  - Message Framing
  - Arbitration
  - Acknowledgement
  - Error detection, error handling
- Sublayer: Logical Link Control (LLC)
  - Flow control
  - Error recovery

Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Data Link Layer
Characteristics

- Short messages assure **short latencies**
  - CAN2.0A (standard format): 11 bit identifier ➔ 47-55 bit data frames (+stuff bits)
  - CAN2.0B (extended format): 29 bit identifier ➔ 67-75 bit data frames (+ stuff bits)
- CAN identifiers do not specify the destination of the message but the content of the message
  ➔ Every node on the bus receives all messages and filters as required
  ➔ **Expandability**
- Contention-free medium access assures **real-time capabilities** (in contrast to Ethernet, WLAN, etc.)
- **Prioritisation** with message identifiers assured by non-destructive arbitration
- Sophisticated error detection, error handling and error confinement assures short- and long-term **error-resistance**

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Data Link Layer
Frame Types

- **Data frame**: Transport of data
  - Periodic or event-driven broadcast of data
    (e.g. temperature, yaw rate)
- **Remote frame**: Request of data
  - Same as Data frame with dominant RTR bit (empty payload)
  - Request for data according to specified identifier
- **Error frame**: Sent when a error is detected
- **Overload frame**: Request of supplementary time interval between preceding and following data frames
- **Interframes**: Separation of data/remote frames from preceding frames

CAN Data Link Layer
CAN2.0A data frame

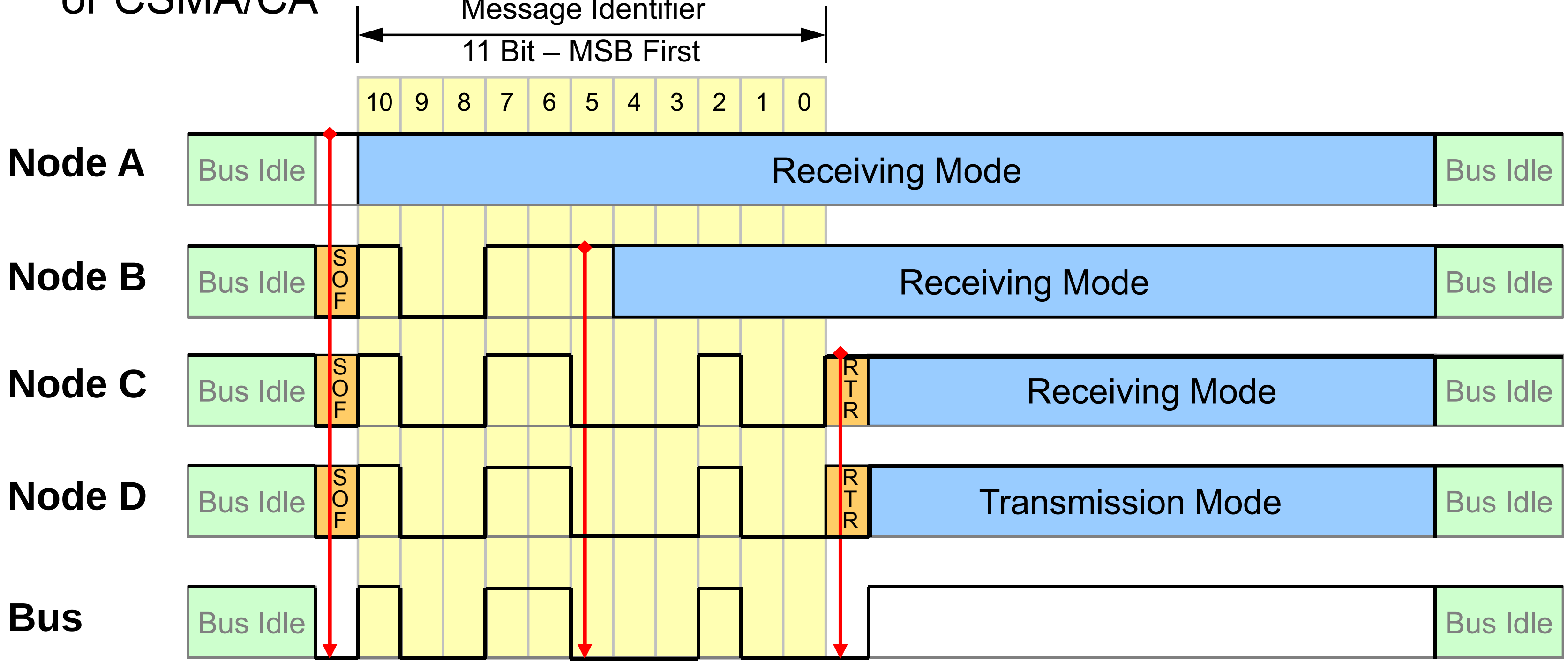
Data Frame

SOF	Arbitration Field	Control Field	Data Field	CRC	ACK	End of Frame
1	11+1	6	0..64	15+1	1+1	7

- SOF (Start of Frame): single dominant bit
- Arbitration Field:
  - 11 bit identifier
  - 1 bit RTR (Remote Transmission Request): dominant for data frames, recessive for remote frames
- CRC (Cyclic Redundancy Check):
  - 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
- ACK (Acknowledgement)
  - 1 bit ACK slot: dominant overwriting
  - 1 bit ACK delimiter: single (always) recessive bit
- End of Frame: 7 recessive bits

CAN Data Link Layer
Arbitration

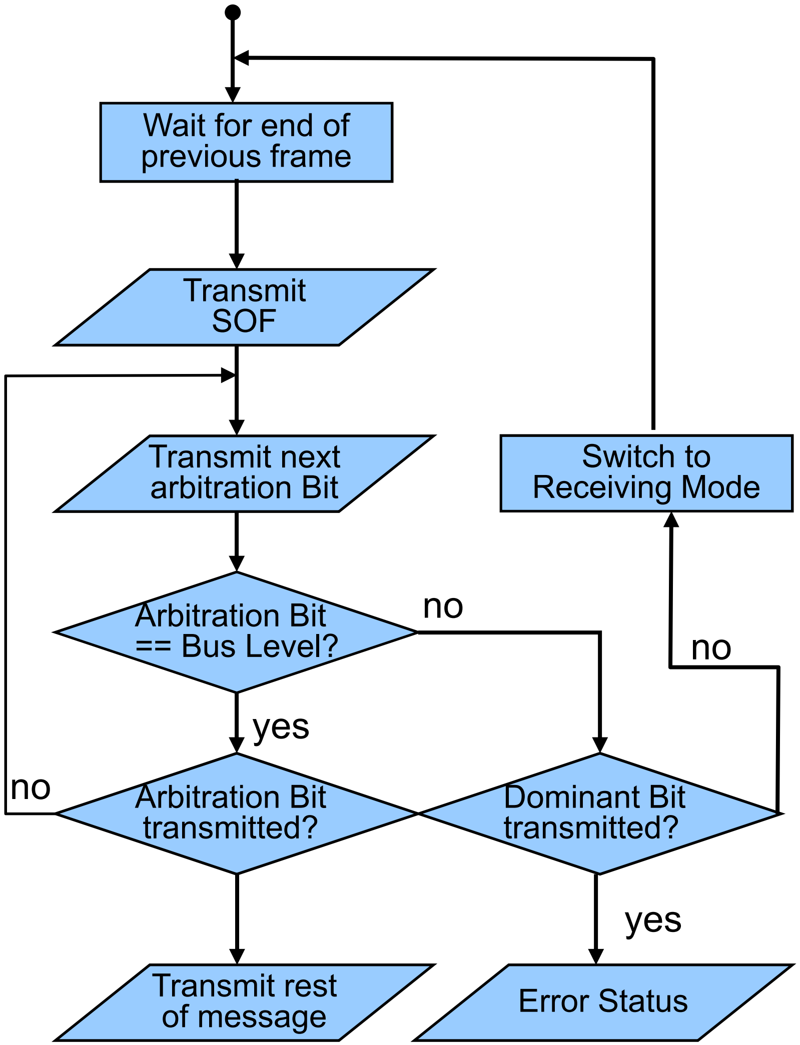
- Bit-wise (bits contained in the identifier)
- Non-destructive CSMA/CR: sender with lowest identifier (=highest priority) wins the arbitration, message is not destroyed as in CSMA/CD or CSMA/CA



DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Data Link Layer
CSMA/CR

- Carrier Sense Multiple Access / Collision Resolution (CSMA/CR)
- Contention-free
- If collision happens, arbitration is used to resolve it
- High bandwidth utilisation (up to 100%)
- Allows prioritisation of messages



DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Data Link Layer
CAN2.0A data frame (Control field)

Control Field

IDE	r0	DLC3	DLC2	DLC1	DLC0
1	1	1	1	1	1

- IDE (IDentifier Extension): Distinction between standard format (CAN2.0A) and extended format (CAN2.0B)
- r0 (reserved)
- DLC3-0 (Data Length Code): size of the data field (0..8 bytes = 9 possibilities requires 4 bit field)

Number of Data Bytes	Data Length Code			
	DLC3	DLC2	DLC1	DLC0
0	dominant	dominant	dominant	dominant
1	dominant	dominant	dominant	recessive
2	dominant	dominant	recessive	dominant
3	dominant	dominant	recessive	recessive
4	dominant	recessive	dominant	dominant
5	dominant	recessive	dominant	recessive
6	dominant	recessive	recessive	dominant
7	dominant	recessive	recessive	recessive
8	recessive	dominant	dominant	dominant

CAN Data Link Layer
Cyclic Redundancy Check

Example message to transmit — Filler — G(x)

```
0011000000101011011000000000000000 : 1100010110011001 = something (not used)
  1100010110011001
  000001010011010010000
       1100010110011001
       01100011000010010
        1100010110011001
        00000011100010110000000
              1100010110011001
              001001110101100100
                1100010110011001
                01011000111111010
                 1100010110011001
                 01110100011000110
                  1100010110011001
                  0010110101011111
```

Remainder used as CRC checksum

$$CAN: G(x) = x^{15} + x^{14} + x^{10} + x^{8} + x^{7} + x^{4} + x^{3} + 1$$
$$= 1100010110011001$$

CAN generator polynomial of degree 15 (requires 16 coefficients to be described, delivers 15 bits of remainder)

DIV mod 2 is equal to XOR

DLR Deutsches Zentrum für Luft- und Raumfahrt e.V. in der Helmholtz-Gemeinschaft

CAN Data Link Layer
Error Detection

- At receiver side:
  - **CRC error**: The calculated CRC by the receiver does not match the CRC contained in the frame
  - **Form error**: Received frame is not in adherence to the fixed frame fields
- At sender side:
  - **Acknowledgement error**: Message has not been acknowledged by a single node - caused by:
    - Transmission error
    - Corrupted ACK field
    - No receivers available
- At other receivers:
  - **Error signalling**: If an error has been detected all other nodes will immediately be informed by an error frame

Controller Area Network (CAN)

Error Handling & Error Confinement

Deutsches Zentrum für Luft- und Raumfahrt e.V.
in der Helmholtz-Gemeinschaft

Error Handling
Error frames

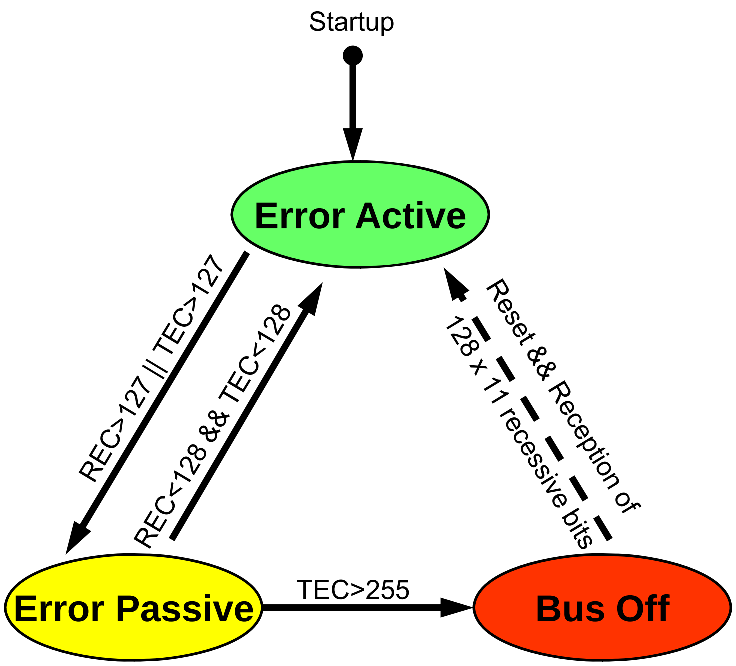
- Error frames are sent immediately after an error has been detected
- Error flag:
  - Active error flag: 6 consecutive dominant bits (breaking the stuffing rule!)
  - Passive error flag: 6 recessive bits (can be squashed by error frames sent by other nodes!)
- Error delimiter: 8 recessive bits

- Majority vote to detect “perpetrator”:
  - Majority of nodes send error frame ➔ transmitter is perpetrator
  - Majority of nodes send no error frame ➔ receiver is perpetrator

Error Handling
Fault confinement

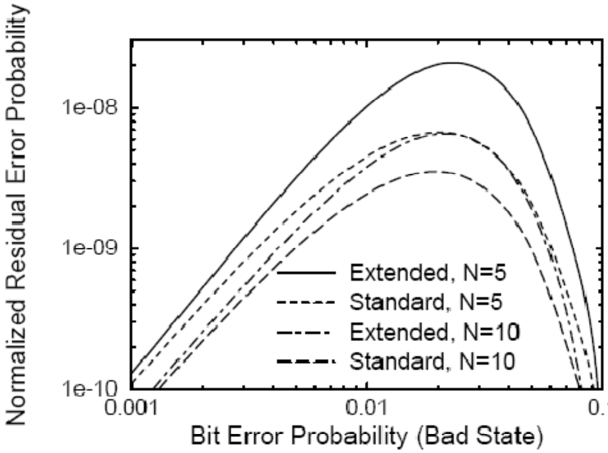
- Every node stores two kinds of errors:
  - Transmit error counter (TEC)
  - Receive error counter (REC)

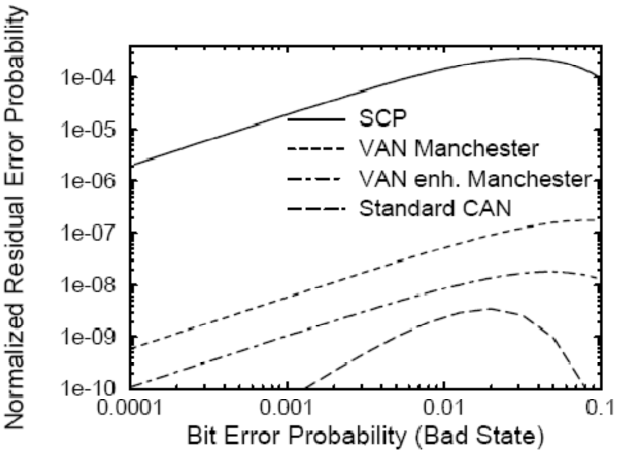
- What a node does if the node is in one of the following states:
  - Error active: Transmission of *Active Error Flags* (dominant) if error is detected by this node
  - Error passive: Transmission of *Passive Error Flags* (recessive) if error is detected by this node
  - Bus off: No transmission on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



Error Handling
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





Source: CiA (1994): Performance of the Error Detection Mechanisms in CAN

Questions

- What were the main design criteria for CAN?
- What ISO OSI layers are defined by Bosch CAN specification (ISO-11898-1)?
- Why does the bus length depend on the data rate?
- How does CAN resolve simultaneous bus accesses?
- What types of errors can be detected by CAN?
- What is bit stuffing good for?
- Why are additional standards such as SAE J2284 and J1939 required for the practical application of CAN?