AUTOMOTIVE SPECIALIZATION PROGRAM

LIN

Edgar Pérez-Navarro

Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

History and Expectation

Growth expectations

– 3 … 10 LIN nodes per vehicle

– 1.2 billion LIN nodes per year world wide

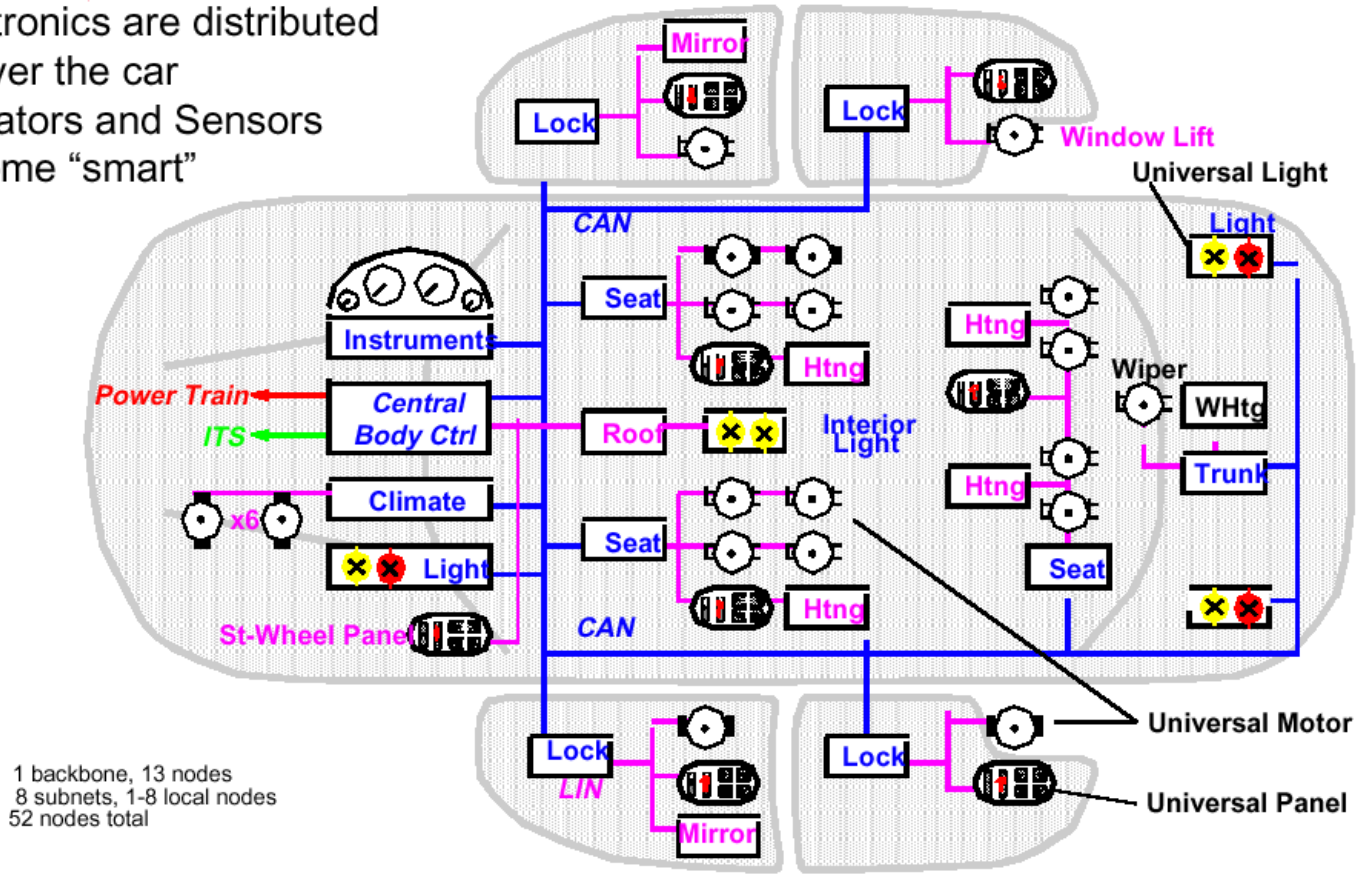
2001 - **first application under development start in car production**

First specification draft released **Jul. 1999**

1998 - **LIN Consortium is initiated as a workgroup (5 car manufactures + 1 semiconductor supplier + 1 tool provider)**

Typical example for LIN in automobiles (1)

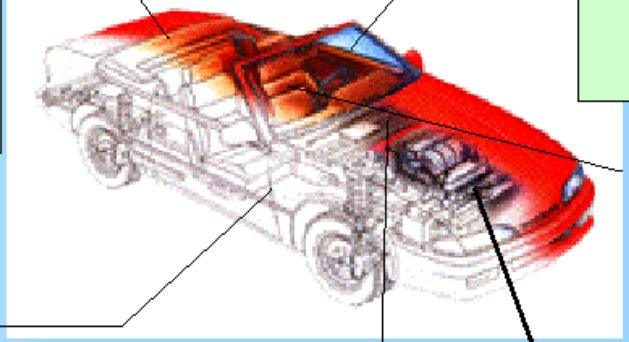
- Electronics are distributed all over the car
- Actuators and Sensors become “smart”



1 backbone, 13 nodes
8 subnets, 1-8 local nodes
52 nodes total

Typical example for LIN in automobiles (2)

Roof:
(high amount of wiring)
Rain Sensor, Light Sensor, Light Control, Sun Roof …
(Rain Sensor needs to be interrogated every 10-20ms)



Steering Wheel:
(very many controls are going to be positioned on the steering wheel)
Cruise Control, Wiper, Turning Light, …
Optional: Climate Control, Radio, Telephone, etc.

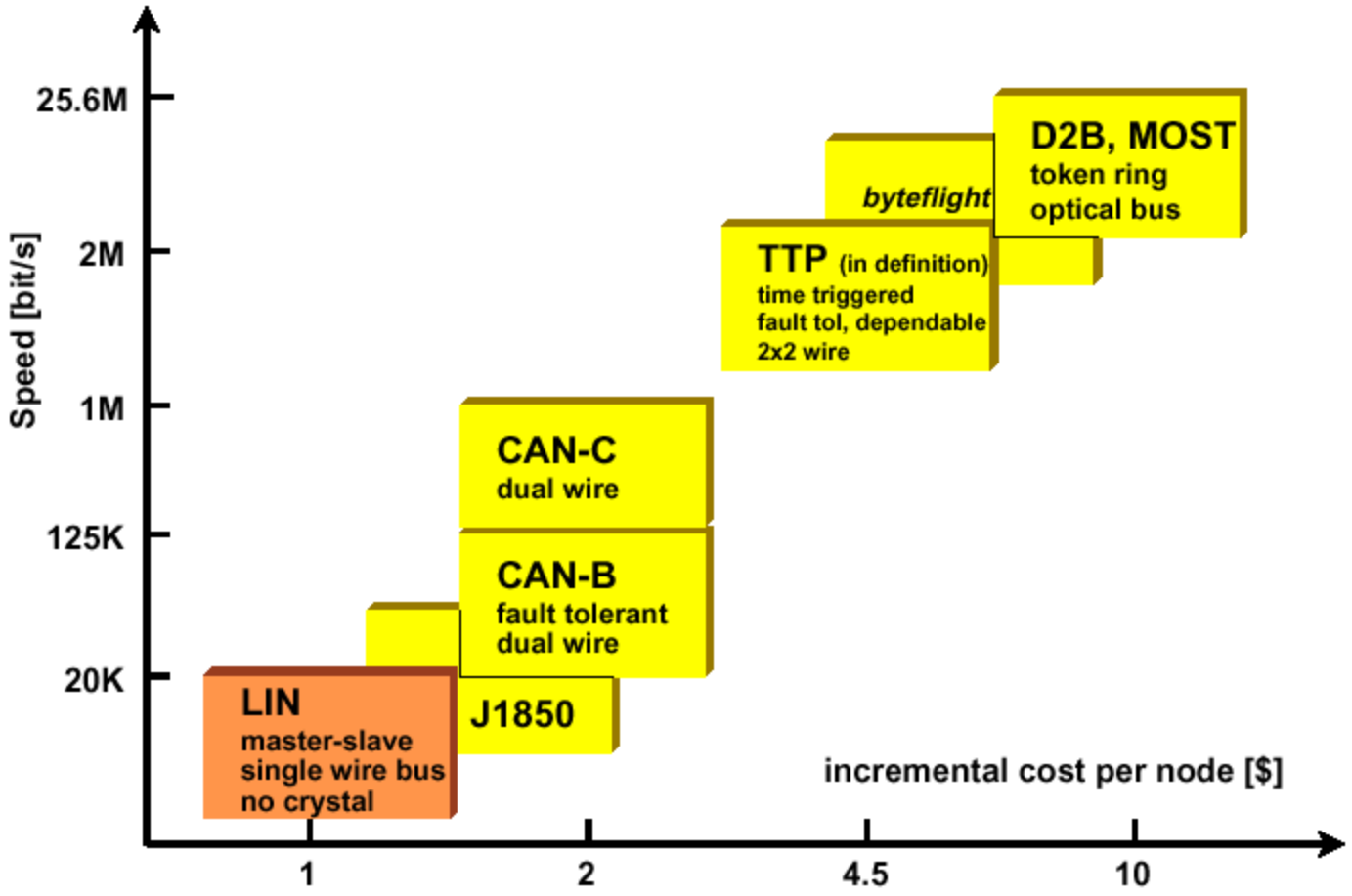
Seat:
many Seat Position Motors,
Occupancy Sensor,
Control Panel

•Door:
Mirror,Central ECU,
Mirror Switch, Window Lift,
Seat Control Switch,
Door Lock, etc.

Climate:
many Small Motors
Control Panel

Engine:
Sensors
Small Motors

Automotive Bus Systems



Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Technical Features(1)

- **Sub-Bus as a extension to CAN to provide connection to local network clusters**

- **Low cost single-wire implementation**
  (cheaper than CAN but does not have the same reliability level as CAN)

- **Low cost silicon implementation based on common UART/SCI interface hardware**
  (almost any microcontroller has necessary hardware on chip)

- **Single Master / Multiple Slave concept**

(no Arbitration is necessary)

Technical Features(2)

- **Self synchronization without quartz or ceramics resonators in slave node**
  (significant cost reduction of hardware platform)
- **Speed up to 20kbit/s**

limited by the EMI of single wire transmission. Recommended Bit Rates:

Slow: 2400 bit/sec

Medium: 9600bit/sec

Fast: 19200 bit/sec

- **Guarantee of latency times for signal transmission**
- **Hot plug-in / plug-out**

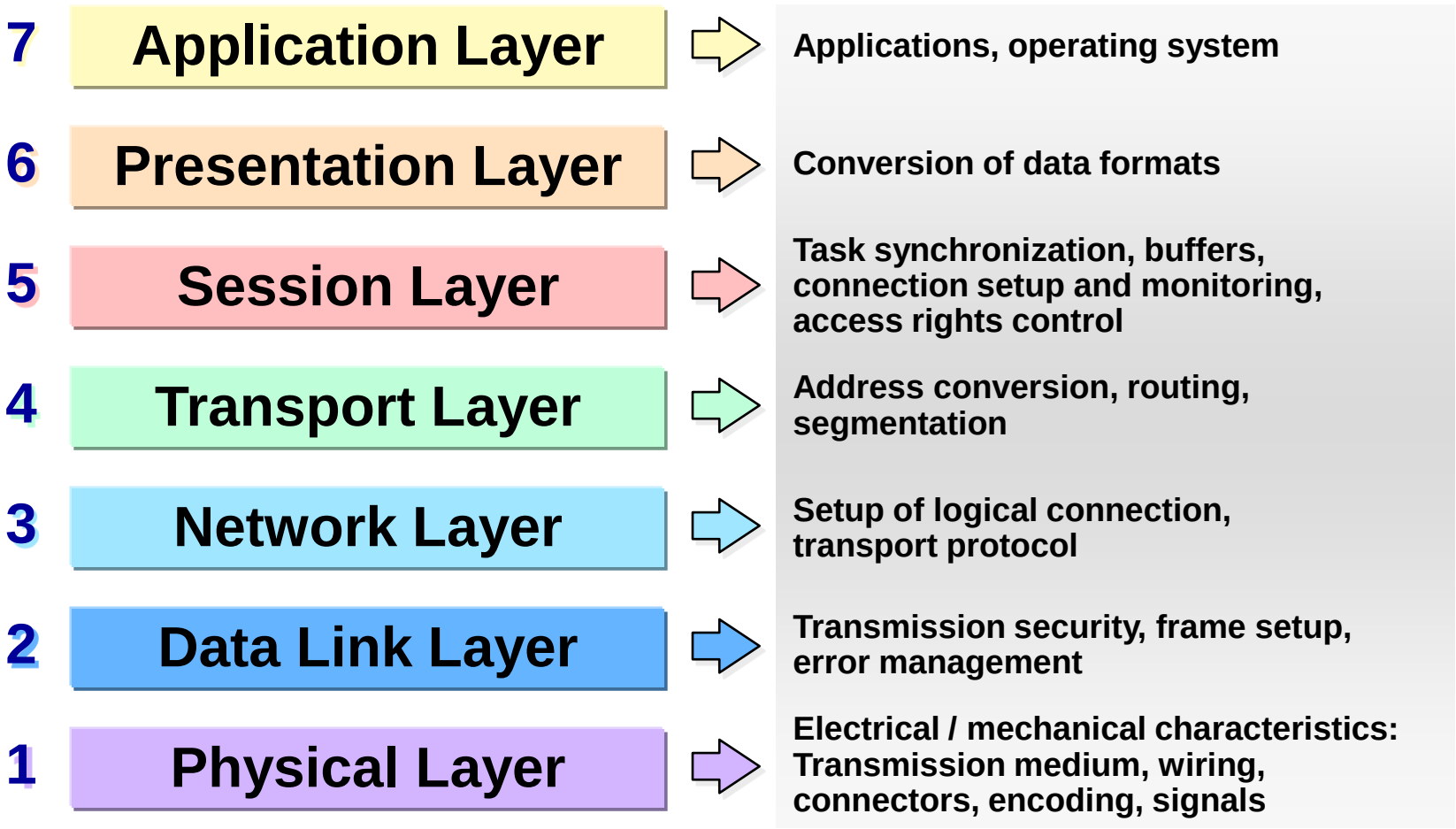
Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

The ISO/OSI reference model and LIN (1)

7	Application Layer	All
6	Presentation Layer	People
5	Session Layer	Seem
4	Transport Layer	To
3	Network Layer	Need
2	Data Link Layer	Data
1	Physical Layer	Processing


7
Application Layer
Applications, operating system
6
Presentation Layer
Conversion of data formats
5
Session Layer
Task synchronization, buffers, connection setup and monitoring, access rights control
4
Transport Layer
Address conversion, routing, segmentation
3
Network Layer
Setup of logical connection, transport protocol
2
Data Link Layer
Transmission security, frame setup, error management
1
Physical Layer
Electrical / mechanical characteristics: Transmission medium, wiring, connectors, encoding, signals

The ISO/OSI reference model and LIN (3)

2 **Data Link Layer**

LLC- Logical Link Control

is concerning with Message Filtering and Recovery Management

→ **Acceptance Filtering, Recovery Management, Time Base Synchronization, Message Validation**

MAC - Medium Access Control

is supervised by a management entity called Fault Confinement

→ **Data Encapsulation/Decapsulation, Error Detection, Error Signaling, Serialization/Deserialization**

1 **Physical Layer** → **Bit Timing, Bit synchronization, Line Driver/Receiver**

Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

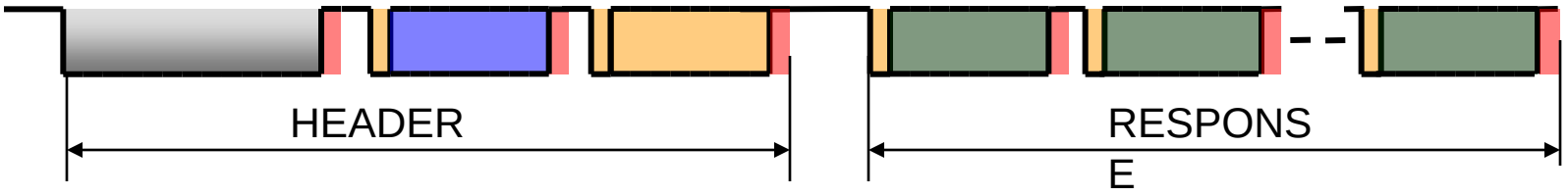
Frames (1)

- **Frame: “Envelope” for transmission data**

- **3 different frame types :**
  - **Message Frames:** used for regular data transmission
  - **Command Frames:** used for software updates, network configuration, and diagnostic purposes
  - **Extended Frames:** are reserved to allow the embedding of user-defined message formats and future LIN formats into the current LIN protocol without violating the current LIN specification

Frames (2)

Frame: Formats



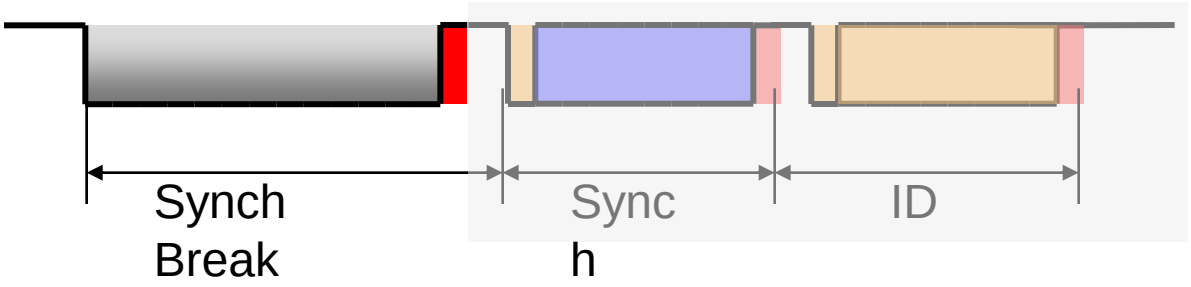
Byte Fields Formats: transmission with LSB first

Overview

LIN Local Interconnect Network Training

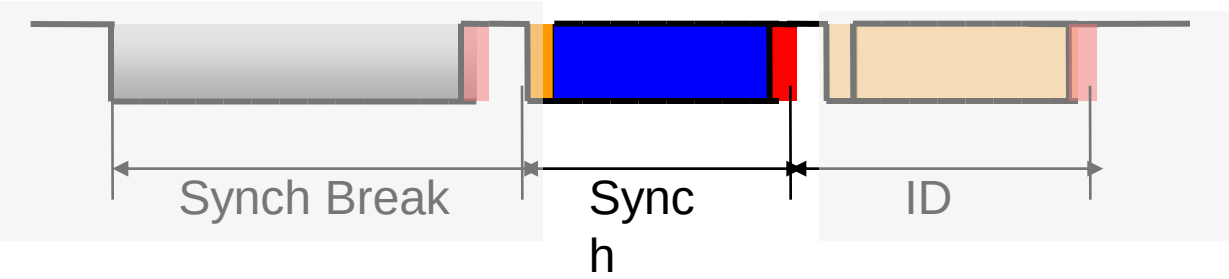
- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Message Frames: Header(1)



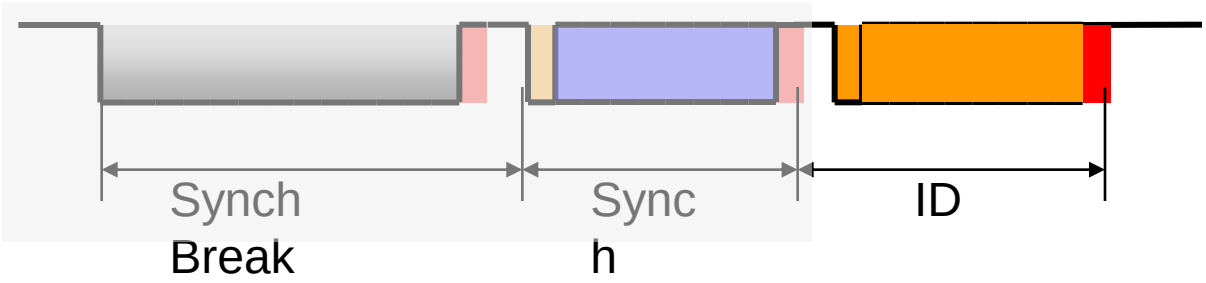
- **Synchronization break field**
  - **identify the beginning of a message frame**
  - **consists of 2 parts:**
    - “dominant” bus value with a minimum duration of $\mathbf{T_{SYNBRK}}$ (13 bits)
    - “recessive” bus value with a minimum duration of $\mathbf{T_{SYNDEL}}$ (1bit)

Message Frames: Header(2)

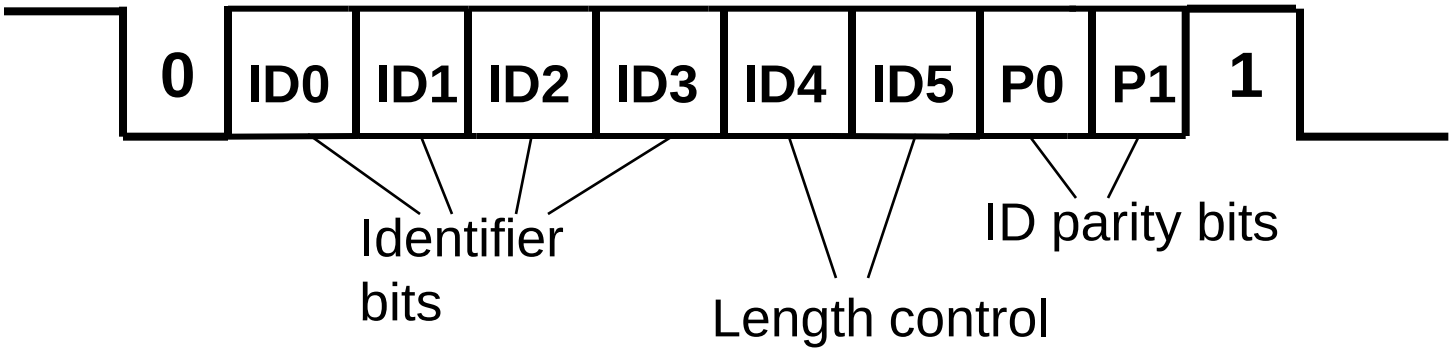


- **Synchronization field**
  - **contains the information for clock synchronization**
  - **consists of pattern: “0x55”**

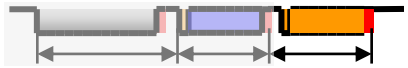
Message Frames: Header(3)



- **Identifier field: contains the content and length of message**



Message Frames: Header(4)



- **ID0 - ID5 IDENTIFIER ⇒ 64 (2^6) identifiers divided in four subsets of 16 identifiers with 2, 4, 8 data fields.**

ID5	ID4	N_{DATA} (number of data fields) [byte]
0	0	2
0	1	2
1	0	4
1	1	8

- **PO - P1 are the parity check bits of identifier**

$$P0 = ID0 \oplus ID1 \otimes ID2 \oplus ID4 \text{ (even parity)}$$

$$P1 = \overline{ID1 \oplus ID3 \otimes ID4 \oplus ID5} \text{ (odd parity)}$$

Message Frames: Response Fields

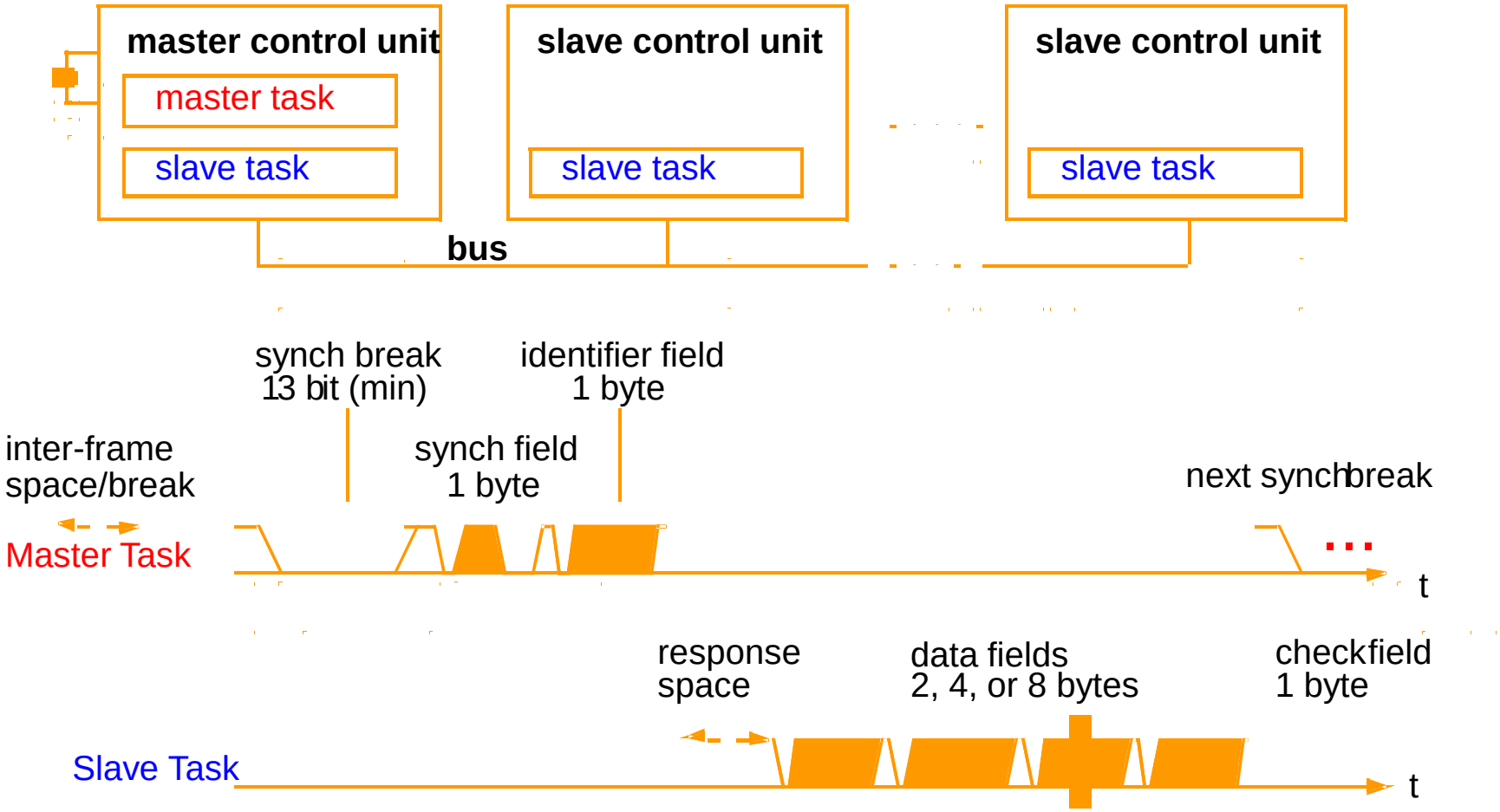
- Contains the data field and the checksum
  - DATA field consists of BYTE fields
  - CHECKSUM field contains the inverted modulo-256 sum over all data bytes calculated by “add with carry”

Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Communication Concept of LIN(1)



Communication Concept of LIN(2)

Master Task

- **Has control the whole Bus and Protocol:**
  - **sends the header of a message**
  - **monitors data bytes and check sum byte and evaluates them on consistency**
  - **receives WakeUpBreak from a slave node when the bus is inactive and they request some action**
  - **serves as a reference with it's clock base (stable clock necessary)**

Communication Concept of LIN(3)

Slave Task

- Is one of 2-16 members on a bus and receives or transmits data when the appropriate ID is sent by the master:
  - waits for Synch Break
  - synchronize on Synch Byte
  - snoops for ID. According to ID slave determine what to do: receive data, transmit data or do nothing
  - when transmitting sends 2, 4, or 8 data bytes and the checksum byte

Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Command and Extended Frames: Command Frames

- are used to broadcast general command requests for service purposes from the master to all bus participants:
  - ID = 0x3C(0x3C) "Master request frame" to send commands and data from the master to the slave node
  - ID = 0x3D(0x7F) "Slave response frame" that triggers one node to send data to the master node
- first data byte of a command frame containing the values from 0x00 to 0x7F are reserved:
  - SLEEP MODE command - used to broadcast the sleep mode to all bus nodes: ID = 0x3C and DATA(0)=0x00

Command and Extended Frames: Extended Frames

- are reserved to allow the embedding of user-defined message formats and future LIN formats into LIN protocol without violating the current LIN specification:
  - ID = 0x3E(0xFE) user defined extended frame
  - ID = 0x3F(0xBF) future LIN extension

- the identifier can be followed by an arbitrary number of LIN bytes fieldc

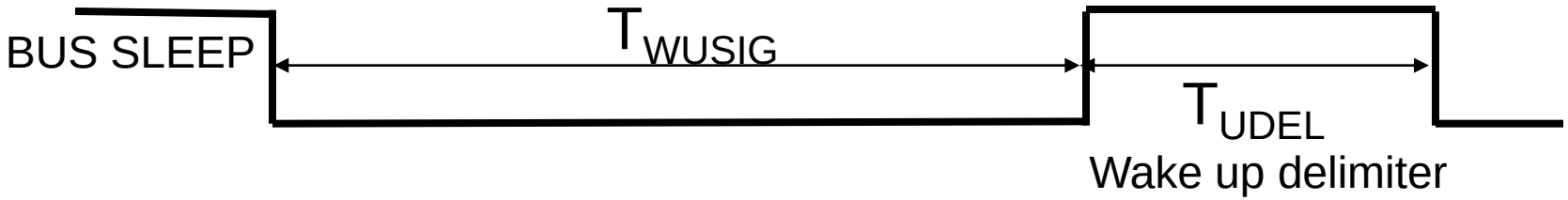
Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Sleep mode and wake-up signal (1)

- Sleep mode on the bus is determine by:
  - a sleep command frame
  - a time-out (T_{TIME_OUT} = 25000 T_{BIT}) on slave nodes in case the sleep command message was corrupted
- Sleep mode on the bus can be terminated by any node (slave task) by sending a Wake-Up Signal (detected by the master as a valid character 0x80, 0xC0, 0x00):c



Sleep mode and wake-up signal (2)

- **Wake up procedure:**
  - **After detecting wake-up signal all nodes run through the Start-up procedure and wait for the master task to send a synch break field followed by a synch field**
  - **If no synch field is detected before "Time out after wake up" ($T_{TOBRK} \leq 128\ T_{BIT}$) a new wake-up signal is issued by the node requesting first wake-up.**
  - **This sequence is issued not more than 3 times then the transmission is suspended for a "Time out after three breaks"($T_{T3BRK} \geq 5000\ T_{BIT}$)**
  - **The re-transmission of the wake-up signal has to be decided by the application**

Overview

LIN Local Interconnect Network Training

- History and Expectation
- Technical Features
- The ISO/OSI reference model and LIN
- Frames
- Message Frames
- Communication concept of LIN
- Command and Extended Frames
- Sleep mode and wake-up signal
- Error and Exception handling

Error and exception handling

- **Possible error types:**
  - **Bit Error**
  - **Checksum Error**
  - **Identifier Parity Error**
  - **Slave Not Responding Error**

Error and exception handling(1)

Bit Errors

- **Error description**

  The bit actually appearing on the bus is different that the one transmitted

- **Method of detection**

  Sending unit monitors the bus while transmitting. A BIT_ERROR has to be detected at that bit time.

- **Fault Confinement**

  This error is detected by - master task in master

  - slave task in slave

  while reading back its own transmission

Error and exception handling(2)

Checksum Errors

➢ Error description

The inverted modulo-256 sum over all received data bytes does not match with the receive checksum byte

➢ Method of detection

The sum of the inverted modulo-256 sum over all received data bytes and the checksum byte does not result in 0xFF

➢ Fault Confinement

This error is detected by - slave task in master when expecting or reading data from the bus

- slave task in slave while reading from the bus

Error and exception handling(3)

Identifier-Parity Errors

➢ Error description

The parity identifier bits does not match with the correct calculated values

➢ Method of detection

☝ **Typical LIN slave application do not distinguish between an unknown but valid identifier and a corrupted identifier**

➢ Fault Confinement

This error is detected by - master task in master while reading back its own transmission

- slave task in slave while reading from the bus

Error and exception handling(4)

Slave-Not-Responding Errors

- **Error description**

  The message frame is not fully completed within maximum length

- **Method of detection**

  A slave task waits the entire message upon transmission of the new header

- **Fault Confinement**

  This error is detected by - slave task in master when expecting or reading data from the bus

  - slave task in slave while reading from the bus only when a slave expects a message from another slave

Error and exception handling(5)

Inconsistent-Synch-Field Errors

- **Error description**

  Synch field is different than the pattern 0x55

- **Method of detection**

  Slave task detects edges of Synch field outside the given tolerance

- **Fault Confinement**

  This error is detected only by slave task in slave

Error and exception handling(6)

Causes for message errors

- Local Disturbance of Ground Potential
- Local Disturbance of Supply Voltage
- Global Electric Disturbance of the Bus Signal
- Unsynchronized Time Base