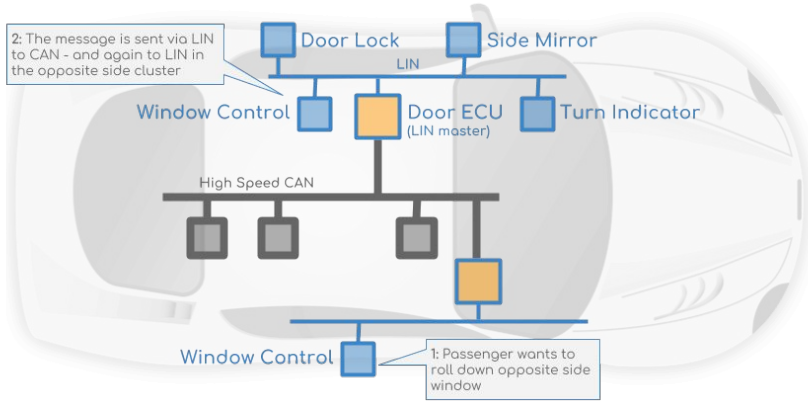

# Local Interconnect Network

Communication Protocols

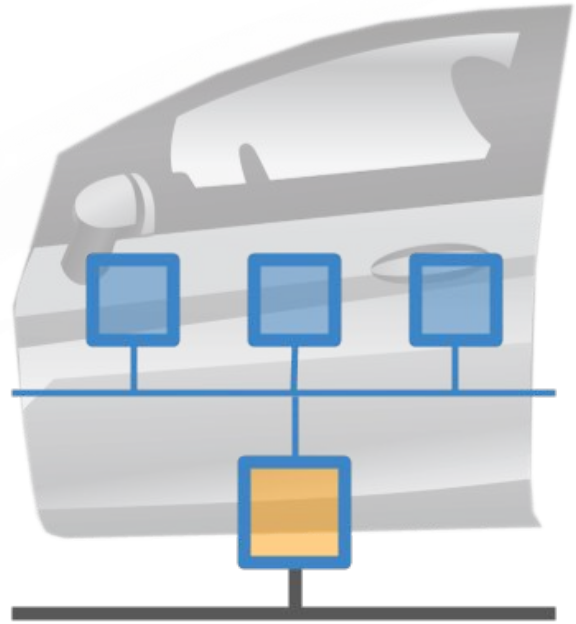
# Local Interconnect Network (LIN)

- Is a synchronous serial bus system
- LIN 1.0 was released by the LIN Consortium in 1999
- LIN 2.2A was released in 2010 and it was standardized by ISO as  ISO-7987 in 2016
- Primarily was developed for a low cost and low speed communication in vehicles
  - But it also is used by other industries.
    - E.g. Home appliances and industrial automation
- Is an alternative to the CAN bus
  - It is used as a subnet to the CAN where
    - High performance and reliability is not required
    - E.g. in vehicles for windows, wipers, air condition etc.



# Local Interconnect Network (LIN - Features)

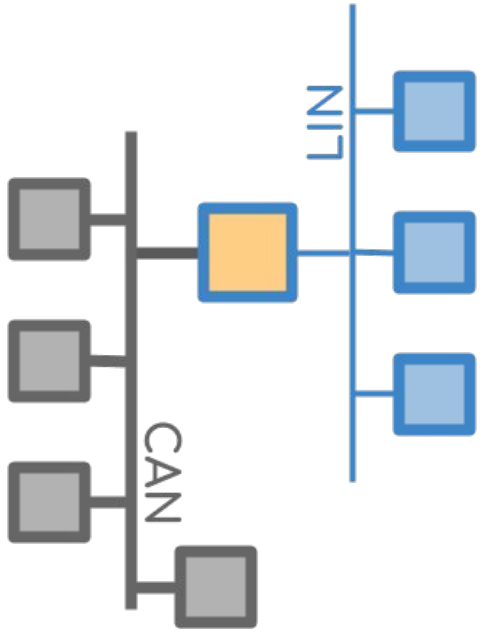
- Low cost and low performance
- A LIN cluster consists of a master and up to 16 slaves
- Uses two wires; data signal line and ground line
  - Supports half-duplex communication
- Transmission speed of LIN is 1 - 20 kbps
- The max bus length is 40 meters
- LIN is a message and byte oriented protocol
  - Variable data length (2, 4, 8 bytes)



- Is a polled bus and supports time triggered scheduling with guaranteed latency time
- LIN supports error detection, checksums, configuration, sleep and wake up modes
- Operating voltage of the bus is 12V

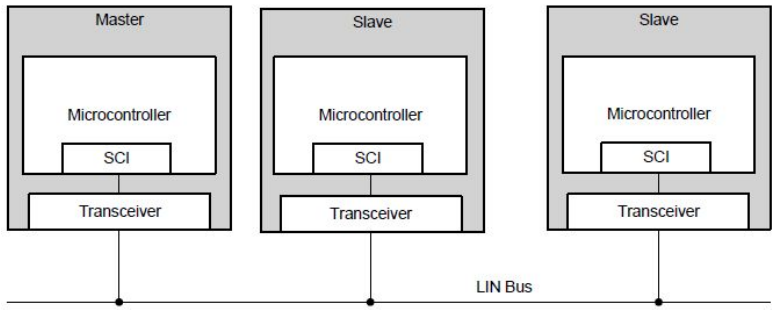
# Local Interconnect Network (LIN and CAN)

- LIN is cheaper
- CAN uses twisted shielded dual wires (5V), while LIN uses a single wire (12V)
- A LIN master typically serves as gateway to the CAN bus
- LIN is deterministic, not event driven (i.e. no bus arbitration)
- LIN clusters have a single master - CAN have multiple
- CAN uses 11 or 29 bit identifiers - LIN uses 6 bit identifiers
- CAN offers up to 1 Mbit/s - LIN max speed is 20 kbit/s
- In both of them “0” is dominant and “1” is recessive
- Both of them use NRZ signal encoding

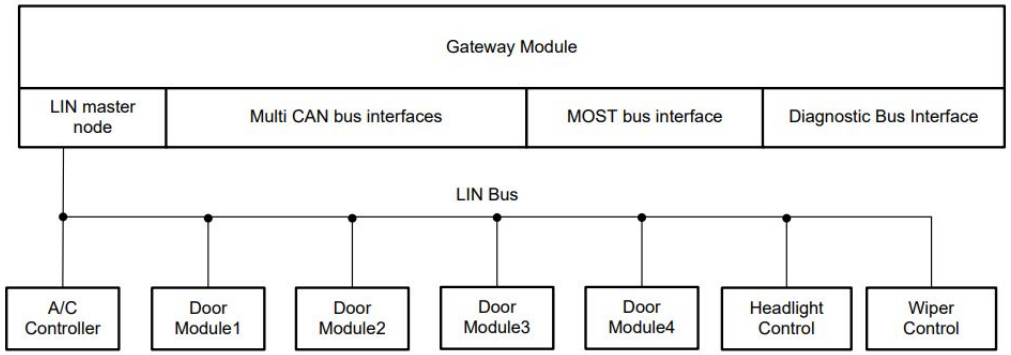


# Local Interconnect Network (LIN - Network Architecture)

- The master is typically a more powerful microcontroller
- The slaves can be less powerful, cheaper microcontrollers.
- The master manages the communication along the bus to each slave.
- A LIN cluster is defined as a number of LIN nodes connected via a physical cable
  - In every cluster there are one master node and up to 16 slave nodes



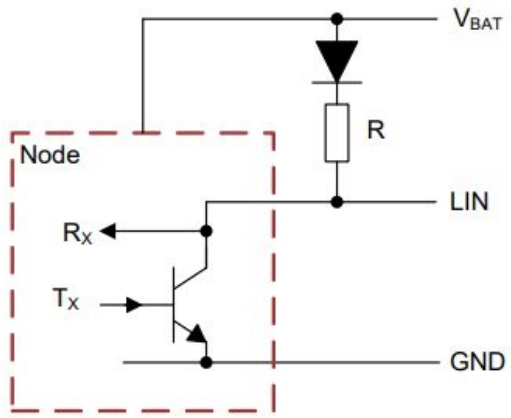

Every node is connected to the bus via a transceiver. The LIN node drives the communication to the transceiver via the Serial Communication Interface (SCI) which is UART in most LIN applications.




The gateway includes the interfaces with all the buses which typically comprise the Vehicle network. The LIN master nose here is configured as part of the gateway module.

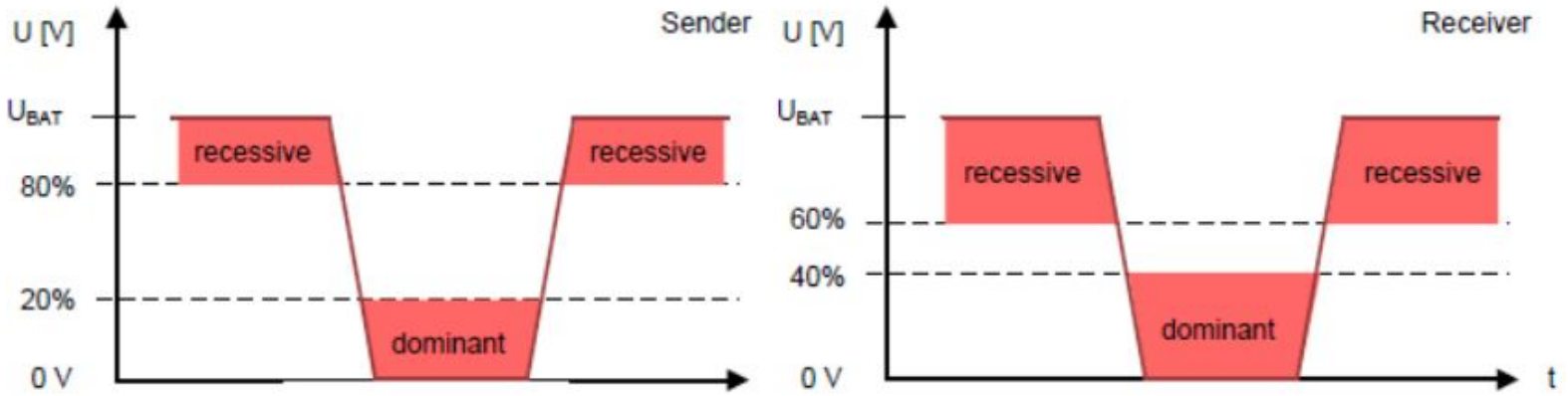
# Local Interconnect Network (LIN - Physical Layer)

- The bus is a pair of wires, data and ground, which
  - Is connected to the nodes
  - Is biased to the battery voltage of the vehicle (+12V)
  - Pull-up resistors is required for the nodes
    - The typical pull-up resistance of a LIN slave is 30 kΩ
      - In all modern transceivers, this is integrated in the IC, so no external pull-up is necessary
    - The master node requires an external line pull-up resistor
      - The typical values are 1 kΩ, 600 Ω and 500 Ω
  - A diode is required in the master node for reverse-polarity protection
  - The LIN bus and transceivers, typically operate at voltages ranging from 9V to 18V
    - Depending on the application; it can go up to 30V



# Local Interconnect Network (LIN - Physical Layer)

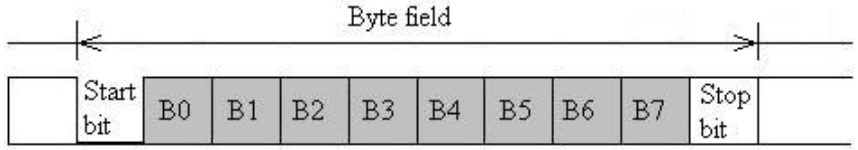
- The sender and receiver have different levels
  - For the sender; High is at least 80% and Low is max 20% of the battery voltage
  - For the receiver; High is at least 60% and Low is max 40% of the battery voltage
  - Due to the difference between the external supply voltage and and the actual bus voltage.
    - Drops in voltage may happen in the cabling, ground shifts and etc.



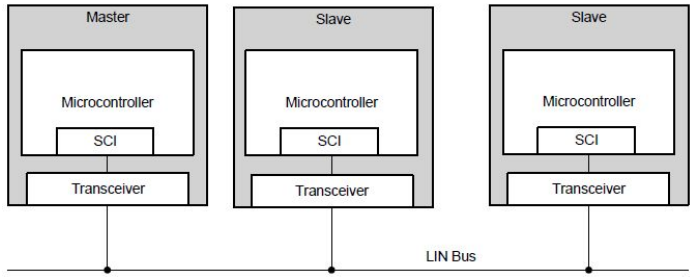

*LIN Bus Threshold Level*

# Local Interconnect Network (LIN - Data Communication)

- Definition of a Byte Field
  - LIN is byte oriented which means that data is sent byte by byte.
  - A byte field contains a start bit (dominant), 8 data bits and a stop bit (recessive)
- Serial Communication Principles
  - The microcontroller transmits bit frames
    - Starting with the dominant start bit in order to synchronizes all receivers on the bus
    - Followed by the LSB bit to the MSB bit and then a stop bit
    - A LIN message is composed of multiple byte field

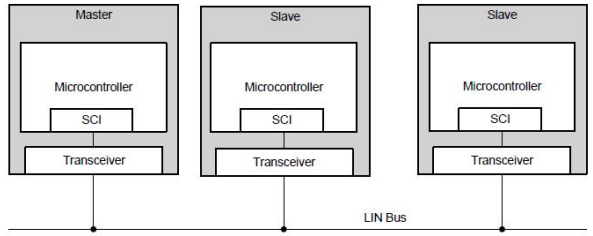



- Master-Slave Principle
  - The master controls all communication on the bus
    - The master node can also act as a slave by replying to its own messages

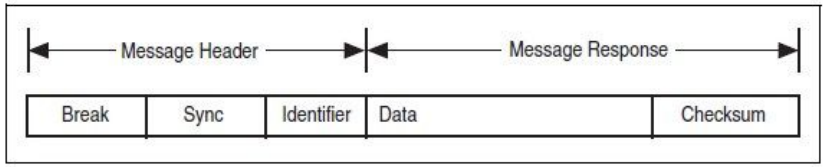


# Local Interconnect Network (LIN - Data Communication)

- Master-Slave Principle ...
  - The slave nodes are unable to communicate with each other
    - They are only capable to response to the master requests.



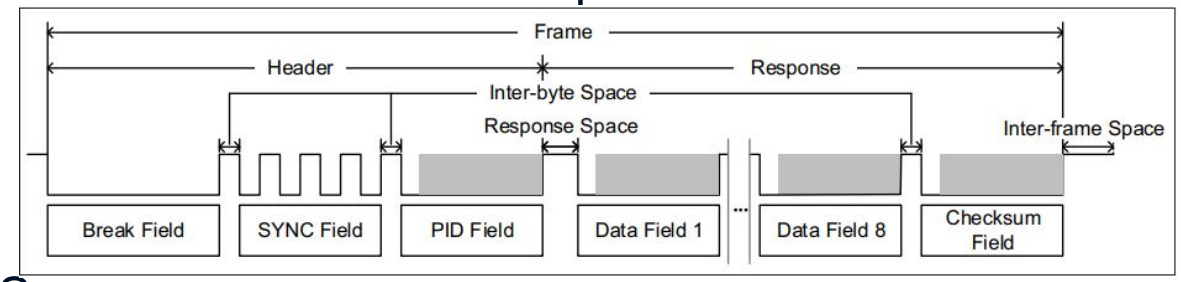

  - The master sends a request to a specific slave as a header (beginning of the frame), and the slave responds to the master as a response frame.
    - There is also a case where the master sends the header and the response, and the slave only listens with no response.




  - In the master node, the developer of the LIN cluster can schedule the messages based on their lengths and avoid collisions on the bus.
    - The master makes sure that the data frames are sent with a right interval
  - A slave can ignore, respond, or just receive the messages
  - Therefore LIN is not used in safety-critical systems (If the master fails, the whole cluster fails).

# Local Interconnect Network (LIN - Frame)

- A frame (message) has a Header and a Response.
- Up to 64 messages can be defined per cluster
- The Header is always transmitted by the master and is divided up into:
  - The Sync Break Field
  - The Sync Field
  - The Protected Identifier (PID) Field



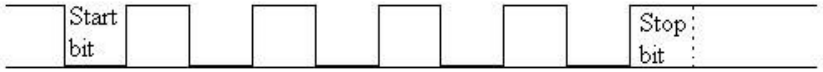

- The optional Inter-byte and Response Spaces
  - Since the low cost LIN slaves are often low performance, delays may occur
  - **Inter-byte Spaces** can be added in order to make delays
  - **The Response Space** allows slave nodes sufficient time to react to the master's header.
  - Sum of all the delay times in a frame shall not exceed **40%** of the frame time.
- The **Inter-frame Space** is the time between the end of a frame and the start of the next frame

# Local Interconnect Network (LIN - Frame Header)

- The Sync Break


  - Is the start of a message and has to be at least 13 dominant bits long including start bit
  - Sync Break ends with a delimiter which should be at least one recessive bit

- The Sync Field




  - This 8 bits predefined value of 0x55 (01010101) is used to synchronize the slaves with the master clock on the falling edges of the signal.

- The Protected Identifier (PID) Field


  - It has 6 bits as the message unique ID and 2 parity bits. The ID also specifies the length of the data
  - Normally one slave is polled by the master at a time and there is no collision risk.
  - The IDs from 0 to 59 are used for unconditional frames.
  - The IDs 60 and 61 are used for diagnostic and configuration frames.
  - The ID 62 is used for user defined frames and ID 63 is reserved.

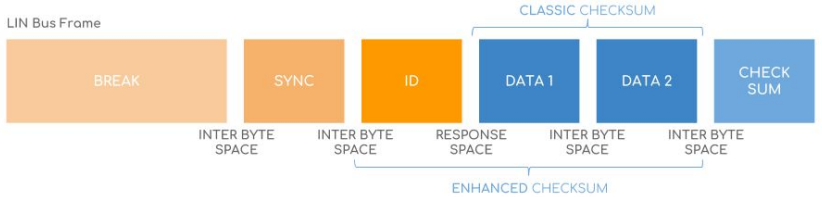
$P0 = ID0 \oplus ID1 \oplus ID2 \oplus ID4$
$P1 = \sim(ID1 \oplus ID3 \oplus ID4 \oplus ID5)$

| ID range | | Frame length |
|---|---|---|
| 0-31 | 0x00 – 0x1f | 2 |
| 32-47 | 0x20 – 0x2f | 4 |
| 48-63 | 0x30 – 0x3f | 8 |

# Local Interconnect Network (LIN - Frame Response)

- The Slave Response Field
  - The response is sent by one of the LIN slaves and is divided into Data and Checksum fields
- Data Field


  - The responding LIN slave may send 2, 4 or 8 data bytes to the LIN bus.
  - The amount of data is fixed by the LIN master (application designer)
- Checksum Field (start bit + 8 bits + stop bit)




  - The 8 bits field ensures the validity of the frame.
  - In the older version of LIN, classic checksum is used(includes only the data field)
  - From LIN 2.1V, an enhanced checksum is used (includes the identifier and the data fields).
  - The checksum is the sum of the bytes.
    - When the sum is greater than 255, then 255 is subtracted.

# Local Interconnect Network (LIN - Frame Types)

- LIN Frame Types
  - In LIN there are 6 frame types. The ID specifies the frame type
    - Unconditional Frames (0 - 59)
    - Event Triggered Frames (0 - 59)
    - Sporadic Frames (0 - 59)
    - Diagnostic Frames (60 - 61)
    - User Defined Frame (62)
      - It is a user defined frame which may contain any type of information
    - Reserved Frame (63)
      - It has been reserved by LIN and must not be used by the user.
  - In practice the majority of communication is done via “Unconditional Frames"
  - All of the frame types have the same basic LIN frame structure.
    - They only differ by timing or content of the data bytes

| ID (dec) | ID (hex) | LIN Frame Type |
|---|---|---|
| 0-59 | 00-3B | Unconditional |
| 0-59 | 00-3B | Event Triggered |
| 0-59 | 00-3B | Sporadic |
| 60-61 | 3C-3D | Diagnostic |
| 62 | 3E | User Defined |
| 63 | 3F | Reserved |

# Local Interconnect Network (LIN - Frame Types)

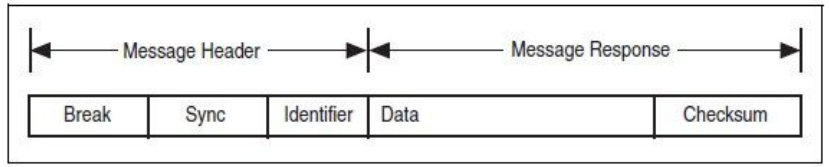
- Unconditional Frame (0 - 59)
  - This is the default form of communication
  - The master sends the header to a specific slave
  - And the relevant slave responds to the master request.
- Event Triggered Frame (0 - 59)
  - The master polls multiple slaves rather than one to save bandwidth
  - A slave only updates its data in an event-triggered frame when the value has changed
  - The slave includes its protected ID in the 1st data byte and its data in the other data bytes
  - If more than one slave want to update data in the frame a collision occurs
    - The master detects the collision(checksum error) and defaults to unconditional frames
    - The master should send unconditional frames to each of the slaves separately

# Local Interconnect Network (LIN - Frame Types)

- Sporadic Frame (0 - 59)
  - This type of frames attempts to provide some dynamic behavior to the LIN.
  - It is only sent by the master if the master knows that a specific slave has updated its data.
  - The master “acts as a slave" and provides the data response to its own header.
    - The master provides information for the slave.
- Diagnostic Frame (60 - 61)
  - Since LIN 2.0, it is possible to read out diagnostic data from master and slave nodes
  - The diagnostic frames are used to find faults in a LIN network
  - The frames always contain 8 data bytes and carry diagnostic or configuration data.
  - ID 60 is used for the master request frame and 61 for the slave response frame
  - The first byte of the data of a diagnostic frame is a NAD (Node Address for Diagnostic)

# Local Interconnect Network (LIN - Error Check & Handling)

- Error detection mechanisms
  - Checking of the Sync Field
    - The slave checks if the clock cycle of the bus can be deduced from Sync Field or not.
    - Sync error occurs if the clock cycle given by the master is out of the tolerance
  - Bit monitoring to check the signal levels
    - The sender compares each sent bit to the bus with the actual bus level.
    - If they are different a bit monitoring error occurs.
    - Bit monitoring error results in termination of transmission
  - Checking the checksum transmitted in the Frame Response
    - Receiver checks the checksum of the arriving data
    - The checksum field can be Classic or Enhanced Checksum
    - A checksum error occurs if the sum of arriving data and checksum is not 0xFF



# Local Interconnect Network (LIN - Error Check & Handling)

- Error detection mechanisms …
  - Checking the parity bits sent in the Frame Header


    - The slave checks the parity bits (P0 and P1) which protects the identifier
    - Parity error occurs if one of the parity bits is not satisfied.
  - Slave Responding Check
    - Checking whether a frame response was sent after a frame header or not
    - Exception: Event Triggered Frame
- Error Handling
  - Error handling is not covered by the LIN specification
  - Error handling must be defined and set up in the implementation of the application.
  - The master executes Status Management.
  - The slaves must communicate their status once per communication cycle.

# Local Interconnect Network (LIN - Error Check & Handling)

- Error Handling …
  - Each slave has its own **status bit** and it should be sent once per sending cycle with an Unconditional Frame and it is evaluated by the master.
  - The status bit is recessive if the slave has detected an error during the last communication cycle. If there is no error, the status bit is transmitted as dominant.
  - If there is an error the master reads the error frame with the error code indicating an error in the response.
- LIN does not require handling of multiple errors within one frame
  - Upon encountering the first error in a frame, the slave aborts processing of the frame
- LIN does not use error counters like CAN

# Local Interconnect Network (LIN - Transmission)

- The bit rate for LIN ranges from 1 to 20 kbps, with a clock tolerance up to ±14%.
  - This 14% value comes from the fact that low-cost oscillators are used
- Synchronization
  - The Sync field is used to synchronize the slaves with the master clock
    - Its value is fixed at “01010101” which is essentially a clock signal at a given frequency
    - This also allows for accurate bit time($T_{bit}$) measurement
- LIN Bus Timing: Because the LIN bus is a polled bus
  - A nominal time slot is allocated to process each frame
    - THeader_Nominal = 34 * Tbit and TResponse_Nominal = 10 * (NData + 1) * Tbit
    - TFrame_Nominal = THeader_Nominal + TResponse_Nominal
  - A maximum time slot  is allocated to process each frame
    - THeader_Maximum = 1.4 * THeader_Nominal and TResponse_Maximum = 1.4 * TResponse_Nominal
    - TFrame_Maximum = THeader_Maximum + TResponse_Maximum = 1.4 * TFrame_Nominal

# Local Interconnect Network (LIN)

- Some useful links
  - LIN 2.2A Specification
  - Introduction to the LIN bus
  - LIN Frame Types
  - LIN BUS EXPLAINED - A SIMPLE INTRO
  - Introduction to the Local Interconnect Network (LIN) Bus
  - Local Interconnect Network (LIN) - Animated Tutorial
  - Local Interconnect Network (LIN) Overview and Training