

# CANbedded LIN Communication

## Embedded Software Components for Local Interconnect Network

LIN (Local Interconnect Network) is a cost-effective, serial communication system for distributed electronic control units in the motor vehicle. The LIN protocol is specified by the LIN Consortium, of which Vector is an Associate Member.

### Features and Advantages

Increasing software complexity makes it advantageous to implement standardized software components. Vector's know-how acquired in developing and working with other protocols has been utilized throughout the development of LIN components:

- > Standard components based on a portion of the generated code are scalable for application-specific requirements
- > Automatic generation of parameter settings and configurations
- > Runtime efficiency
- > Low demand for ROM and RAM memory
- > Simple interfacing to the application software
- > Compatibility with other Vector products including analysis, simulation, and calibration tools
- > Support of LIN 1.2/1.3/2.0 and LIN 2.1
- > Support of multi-channel LIN configurations

LIN networks are described in the LDF (LIN Description File) format according to the "LIN Configuration Language Specification". This includes all of the information needed to define signals, messages, baudrate, and schedule tables. A configuration tool is used to adapt all LIN components to ECU-specific requirements. The tool imports either LDFs or NCFs (Node Capability File) for parameterization.

Modifications and settings in detail:

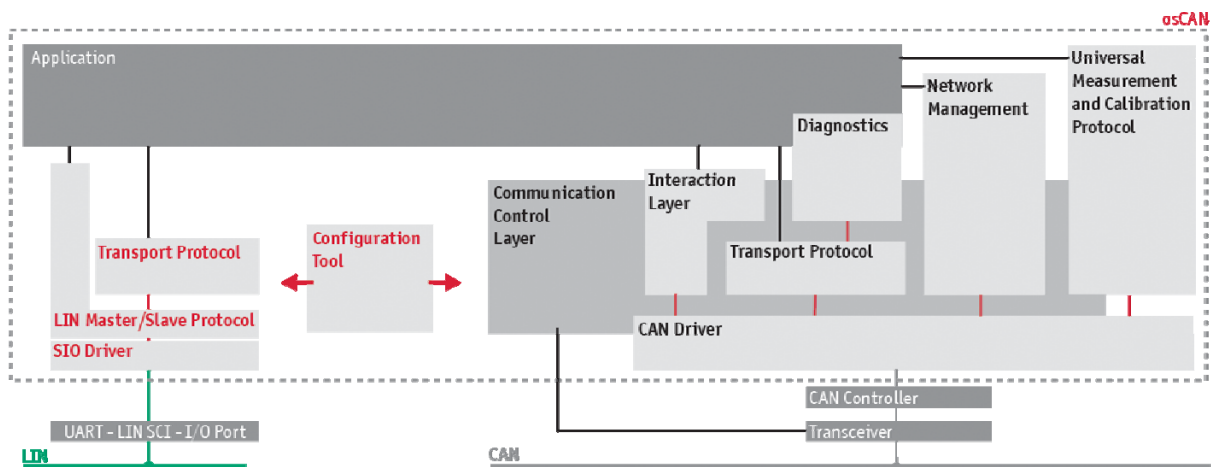
- > Adaptation of ECU-specific parameters to the network design
- > Configuration of the LIN driver to the properties of the specific ECU, e.g. oscillator frequency, UART, etc.
- > Configuration of diagnostic services for LIN Slave ECUs
- > Definition of message and signal callbacks

This makes it possible to create an individually configured LIN stack. The object codes of the LIN software components must be compiled and linked with the application. Vectors LIN simulation and analysis tools are available for integration, analysis, and for test purposes. For physical bus access one of Vector's XLI-Interface products can be purchased; these products serve as universal, serial bus interfaces for the LIN bus. Timing control of the LIN software components is achieved by having tasks called periodically by the application or an operating system. Therefore the LIN driver does not require any auxiliary timers.

### Functions

The fundamental objective of the implementation is to provide an interface that is easy to use and satisfies all applications. The standardized interface offers these features:

- > Initialization of the LIN hardware
- > Configurable baudrate (Recommended: 2400, 9600, and 19200 Baud)
- > Sending of LIN messages (confirmation by flags and callback functions)
- > Receiving of LIN messages (notification by flags and callback functions)



### LIN and CAN Software Architecture

**Availability**

Our LIN software components for motor vehicle ECUs are available for a large number of commonly used microcontrollers. For further information please refer to our home page on the Internet at: [www.canbedded.de/lin](http://www.canbedded.de/lin)

**Training**

As part of our training program, we offer a range of classes and workshops on CANbedded LIN Communication. For more information and the dates of our training courses, please visit our homepage on the Internet at: [www.vector-academy.com](http://www.vector-academy.com)

- > Sleep and wakeup handling
- > Protocol error handling
- > Timeout monitoring of messages
- > Checking of runtime behavior during the development phase (debugging)

**Application Areas**

LIN was developed to round out a collection of automotive protocol standards, but it has been adopted in other areas as well, such as automation engineering. The focus is on cost-effective communication networks for ECU subsystems with low data rates.

LIN therefore supplements existing bus protocols over the long term.

**Hardware Interfaces**

Two software components are needed for LIN communication. One component is a hardware-dependent low-level SIO driver, whose task is to implement abstraction of access to the serial interface. The second component, the LIN driver, is overlaid on this and provides an API to higher-level components. This API satisfies all of the requirements of the current LIN standard.

The driver is a standardized and easy-to-configure software component for communication between master and slave network nodes.

The LIN driver is one of the CANbedded components from Vector; therefore homogeneous integration with CAN components is supported.

This makes it easy to implement CAN-LIN gateways, which are usually implemented as master network nodes.

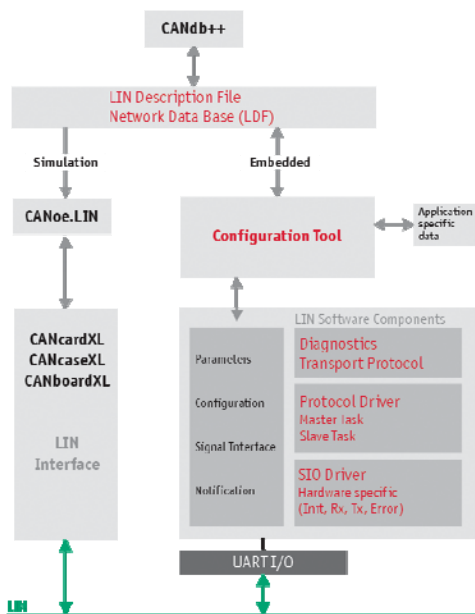
**LIN Master Transport Layer**

Since the master ECU usually has an auxiliary CAN channel, over which diagnostics are performed from a tester, the main task of the LIN master is to pass the received tester requests to the relevant LIN slaves. The LIN Diagnostic Transport Layer (DTL) is used for this; it offers two API types according to the LIN 2.0 and 2.1 specification:

- > DTL Raw is used if the data already exist in segmented form (routing of CAN TP messages on LIN).
- > DTL Cooked accepts assembled data and segments them according to LIN bus requirements (Diagnostic Service Data Units).

**LIN Slave Diagnostics**

LINdiag is a very compact implementation of UDS 14229.3 diagnostics for LIN slave ECUs. It receives requests, routes them to the appropriate service, and constructs the response in interaction with the application. This involves internal management of a diagnostic buffer to monitor for data consistency and overruns. LINdiag needs the LIN Transport Protocol to send and receive requests and responses, and this is included with the product.



Overview of Design, Configuration and Integration Processes

**Further Options**

- > Optional for slave ECUs: Diagnostic components with associated transport protocol
- > Optional for master ECUs: Transport protocol with raw or cooked API
- > Gateway: CAN-LIN and LIN-LIN routing for master ECUs
- > SAE J2602: As a supplemental option, support of the SAE J2602 "Recommended Practice" specification is available for LIN drivers.
- > Flashing of LIN slaves

**Product Components**

The following items are supplied with the product:

- > Configuration tool (executable Windows program)
- > C sources for LIN driver and optional components
- > Sample programs
- > Documentation and operating instructions

**Licenses**

When the same hardware interface and same compiler are used the acquired licenses can be used for all devices of a processor type and any desired projects.

**Supplemental Services**

Vector also offers related project services such as:

- > LIN driver expansions
- > Support of other target hardware
- > Development of higher OEM-specific protocol layers, e.g. diagnostic protocol

Services also include telephone and e-mail hotlines, special workshops with OEM-specific focus, and individualized consultation on software integration with the goal of reducing development time and costs.