Local Interconnect Network

Article Talk

Read

文△ 10 languages ∨

Edit View history Tools >

Contents [hide]

(Top)

Network topology

Overview

LIN hardware

→ LIN protocol

Header

Frame types

Response Slave node position detection (SNPD) or autoaddressing

XWDC auto-addressing procedure Bus shunt method (BSM)

BSM auto-addressing procedure LIN advantages

Applications Addressing

External links

See also References

Extra wire daisy chain (XWDC)

LIN is a broadcast serial network comprising 16 nodes (one master and typically up to 15 slaves). [2][3][4][5] All messages are initiated by the master with at most one slave replying to a given message identifier. The master node can also act as a slave by

Network topology [edit]

standardized in the ISO 17987 series.

From Wikipedia, the free encyclopedia

replying to its own messages. Because all communications are initiated by the master it is not necessary to implement a collision detection. [6]

(DC-LIN) transceiver. LIN over DC power line (DC-LIN) was standardized as ISO/AWI 17987-8.^[1]

The master and slaves are typically microcontrollers, but may be implemented in specialized hardware or ASICs in order to save cost, space, or power. Current uses combine the low-cost efficiency of LIN and simple sensors to create small networks. These sub-systems can be connected by back-bone-

LIN (Local Interconnect Network) is a serial network protocol used for communication between components in vehicles. It is a single wire, serial

network protocol that supports communications up to 19.2 Kbit/s at a bus length of 40 meters. The need for a cheap serial network arose as the

technologies and the facilities implemented in the car grew, while the CAN bus was too expensive to implement for every component in the car.

In the late 1990s, the LIN Consortium was founded by five automakers (BMW, Volkswagen Group, Audi, Volvo Cars, Mercedes-Benz), with the

CAN in Automation has been appointed by the ISO Technical Management Board (TMB) as the Registration Authority for the LIN Supplier ID

technologies supplied (networking and hardware expertise) from Volcano Automotive Group and Motorola. The first fully implemented version of the

new LIN specification (LIN version 1.3) was published in November 2002. In September 2003, version 2.0 was introduced to expand capabilities and

make provisions for additional diagnostics features. LIN may be used also over the vehicle's battery power-line with a special LIN over DC power line

European car manufacturers started using different serial communication technologies, which led to compatibility problems.

network (i.e. CAN in cars).[/] Overview [edit]

Automotive Group) and Freescale (Formerly Motorola, now NXP). The first fully implemented version of the new LIN specification was published in November 2002 as LIN version 1.3. In September 2003 version 2.0 was introduced to expand configuration capabilities and make provisions for significant additional diagnostics features and tool interfaces. The protocol's main features are listed below:

• Single master, up to 16 slaves (i.e. no bus arbitration). This is the value recommended by the LIN Consortium to achieve deterministic time response.[8] Slave Node Position Detection (SNPD) allows node address assignment after power-up^[9] • Single wire communications up to 19.2 kbit/s @ 40 meter bus length. [8][10] In the LIN specification 2.2, [9] the speed up to 20 kbit/s.

Guaranteed latency times.

Configuration flexibility.

Detection of defective nodes.

Variable length of data frame (2, 4 and 8 byte).

 Multi-cast reception with time synchronization, without crystals or ceramic resonators. Data checksum and error detection.

Operating voltage of 12 V.^[8]

 Low cost silicon implementation based on standard UART/SCI hardware. Enabler for hierarchical networks.

6. **Reserved frame.** These shall not be used in a LIN 2.0 cluster. Their identifier is 63 (0x3F).

starting the communication and not as a standard data byte with all values zero (hexadecimal 0x00).

Data is transferred across the bus in fixed form messages of selectable lengths. The master task transmits a header that consists of a break signal followed by synchronization and identifier fields. The slaves respond with a data frame that consists of between 2, 4 and 8 data bytes plus 3 bytes of

control information.[9]

existing CAN network leading to hierarchical networks within cars.

LIN message frame [edit]

A message contains the following fields: [9] Synchronization break Synchronization byte Identifier byte

Frame types [edit]

1. **Unconditional frame.** These always carry signals and their identifiers are in the range 0 to 59 (0x00 to 0x3b). All subscribers of the unconditional frame shall receive the frame and make it available to the application (assuming no errors were detected).

2. **Event-triggered frame.** The purpose of this is to increase the responsiveness of the LIN cluster without assigning too much of the bus

Data bytes

Checksum byte

bandwidth to the polling of multiple slave nodes with seldom occurring events. The first data byte of the carried unconditional frame shall be equal to a protected identifier assigned to an event-triggered frame. A slave shall reply with an associated unconditional frame only if its data

LIN hardware [edit]

(0x3C), called master request frame, or 61(0x3D), called slave response frame. Before generating the header of a diagnostic frame, the master task asks its diagnostic module if it shall be sent or if the bus shall be silent. The slave tasks publish and subscribe to the response according to their diagnostic module. 5. **User-defined frame.** These can carry any kind of information. Their identifier is 62 (0x3E). The header of a user-defined frame is always transmitted when a frame slot allocated to the frame is processed

value has changed. If none of the slave tasks responds to the header the rest of the frame slot is silent and the header is ignored. If more than

one slave task responds to the header in the same frame slot a collision will occur, and the master has to resolve the collision by requesting all

transceiver and works as a pure LIN node without added functionality. As LIN Slave nodes should be as cheap as possible, they may generate their internal clocks by using RC oscillators instead of crystal oscillators

The LIN specification was designed to allow very cheap hardware-nodes being used within a network. It is a low-cost, single-wire network based on

microcontroller generates all needed LIN data (protocol ...) (partly) by software and is connected to the LIN network via a LIN transceiver (simply

speaking, a level shifter with some add-ons). Working as a LIN node is only part of the possible functionality. The LIN hardware may include this

ISO 9141.[11] In today's car networking topologies, microcontrollers with either UART capability or dedicated LIN hardware are used. The

The LIN-Master uses one or more predefined scheduling tables to start the sending and receiving to the LIN bus. These scheduling tables contain at least the relative timing, where the message sending is initiated. One LIN Frame consists of the two parts **header** and **response**. The header is always

sent by the LIN Master, while the response is sent by either one dedicated LIN-Slave or the LIN master itself. Transmitted data within the LIN is transmitted serially as eight bit data bytes with one start bit, one stop-bit, and no parity (break field does not have a start bit and stop bit). Bit rates vary within the range of 1 kbit/s to 20 kbit/s. Data on the bus is divided into recessive (logical HIGH) and dominant

(logical LOW). The time normally is considered by the LIN Masters stable clock source, the smallest entity is one bit time (52 μ s @ 19.2 kbit/s).

Two bus states — Sleep-mode and active — are used within the LIN protocol. While data is on the bus, all LIN-nodes are requested to be in active

state. After a specified timeout, the nodes enter Sleep mode and will be released back to active state by a WAKEUP frame. This frame may be sent by

internal software application. After all nodes are awakened, the Master continues to schedule the next Identifier. Header [edit] The header consists of five parts: BREAK: The BREAK field is used to activate all attached LIN slaves to listen to the following parts of the header. It consists of one start bit and several

dominant bits. The length is at least 11-bit times; standard use as of today are 13-bit times, and therefore differs from the basic data format. This is

used to ensure that listening LIN nodes with a main-clock differing from the set bus baud rate in specified ranges will detect the BREAK as the frame

SYNC: The SYNC is a standard data format byte with a value of hexadecimal 0x55. LIN slaves running on RC oscillator will use the distance between

a fixed amount of rising and falling edges to measure the current bit time on the bus (the master's time normal) and to recalculate the internal baud

INTER BYTE SPACE: Inter Byte Space is used to adjust for bus jitter. It is an optional component within the LIN specification. If enabled, then all LIN nodes must be prepared to deal with it.

Response [edit]

Restrictions:

Reserved

2. bus_out

Shunt resistor

2 Pull-up mode

3 Current source mode

4. Differential amplifier

Analog to digital converter

BSM auto-addressing procedure [edit]

1.2 The bus current is measured, Ioffset

2.2 The bus current is measured, I_{PU}

SNPD Method

All auto-addressing slaves must be in one line

Standard slaves can be connected in any way

0xFF

Extra wire daisy chain (XWDC) [edit]

SNPD Method ID

the bus, it may be called Tx-identifier.

and one between every Data byte in the payload. IDENTIFIER: The IDENTIFIER defines one action to be fulfilled by one or several of the attached LIN slave nodes. The network designer has to ensure the fault-free functionality in the design phase (one slave is allowed to send data to the bus in one frame time).

If the identifier causes one physical LIN slave to send the response, the identifier may be called a Rx-identifier. If the master's slave task sends data to

RESPONSE SPACE: Response Space is the time between the IDENTIFIER field and the first Data byte which starts the LIN RESPONSE part of the LIN frame. When a particular LIN frame is transmitted completely, Header + Response, by the LIN MASTER, the LIN MASTER will use the full RESPONSE SPACE TIME to calculate when to send the response after sending the header. If the response part of the LIN frame is coming from a

physically different SLAVE NODE, then each node (master & slave) will utilize 50% of the Response Space time in their timeout calculations.

CHECKSUM: There are two checksum-models available within LIN - The first is the checksum including the data bytes only (specification up to Version 1.3), the second one includes the identifier in addition (Version 2.0+). The used checksum model is pre-defined by the application designer.

These methods allow the detection of the position of slave nodes on the LIN bus and allow the assignment of a unique node address (NAD).^[12]

Allows similar or the same devices to be connected on the bus without end of line programming or connector pin programming.

• The output of the first node, D₂, is connected to the input, D₁ of the second node, and so on resulting in a daisy chain.

Bus shunt method 0x02 Elmos Semiconductor 0x03 **TBD** Reserved

TBD

• The first SNPD node input D1 is either set to GND or connected to the output of the master.

Each slave node has to provide two extra pins, one input, D_1 , and one output, D_2 .

Company

Slave node position detection (SNPD) or autoaddressing [edit]

3. Comparator referenced to V_{bat}/2 XWDC auto-addressing procedure [edit] At the start of the procedure no SNPD devices have a NAD assigned 1 First auto-addressing LIN message 1.1 All outputs (D₂) are set to a high level, all pull-downs are turned off 1.2 The first SNPD node is selected. It is identified by having the input D₁ low (hardwired).

2.3 The detected node turns on the pull-down at the output D₂

Bus shunt method (BSM) [edit] Each slave node has two LIN pins

2.2 The selected node takes the address from the LIN configuration message

2.4 Steps 2.1-2.4 are repeated until all slave nodes are assigned an address

3 All pull-ups and pull-downs are turned off completing the addressing procedure

At the start of the procedure, none of the SNPD devices have a NAD assigned. The autoaddressing routine is performed during the sync field. The sync field is broken into three phases:

2.1 Pull-ups are turned on and current sources remain off

2.3 Nodes with $\Delta I = I_{PU}-I_{offset} < 1$ mA are "selected"

1. The standard pull-up must be switchable

2. Switchable 2 mA current source from V_{bat}

3.1 Selected nodes switch current source on and others switch pull-ups off 3.2 Bus current is measured, I_{CS} 3.3 Node with $\Delta I = I_{CS} - I_{offset} < 1$ mA is detected as the last node

 Harness reduction More reliable vehicles Extension easy to implement.

No protocol license fee required

Cheaper than CAN and other communications buses

LIN advantages [edit]

Components available

Easy to use

Roof

Door

Illumination

See also [edit]

Steering wheel

Seat position motors, occupant sensors, control panel Seat Engine Sensors, small motors, cooling fan motors Grille Grille shutter

Addressing [edit] Addressing in LIN is achieved with a NAD (Node ADdress) that is part of the PID (protected identifier). NAD values are on 7bits, so in the range 1 to

5. A "The LIN Short Story" m 6. ^ "Lin Concept" . LIN Overview. LIN Administration. Archived from the

V •T •E

V •T •E

This page was last edited on 15 May 2023, at 08:18 (UTC).

External links [edit]

A free online LIN checksum calculator ☑

original

on 5 October 2011. Retrieved 28 October 2011. 7. A "Target Applications" 2. LIN Overview. LIN Administration. Archived from the original

on 5 October 2011. Retrieved 28 October 2011.

Detection (SNPD)" LIN Consortium

 it is not longer available, because the latest LIN specification (2.2A) is being transcribed to the ISO (International Organization for

12. ^ Anand Gopalan, Akeem Whitehead. "Automatic Slave Node Position

8. ^ a b c "Clemson Vehicular Electronics Laboratory: AUTOMOTIVE

2009-01-14. 090114 cvel.clemson.edu

at the Wayback Machine

090114 interfacebus.com

Machine

BUSES" ☑. Archived from the original ☑ on 2012-04-14. Retrieved

9. A a b c d e LIN Specification Package Rev. 2.2a Archived 2 2008-04-26

10. ^ "LIN Bus Description, Automotive Bus, Local Interconnect Network" ☑.

11. ^ LIN Technical Overview 2 Archived 2 2011-07-19 at the Wayback

LIN Bus Explained - A Simple Intro ☑

Category: Serial buses

The LIN bus is an inexpensive serial communications protocol, which effectively supports remote application within a car's network. It is particularly intended for mechatronic nodes in distributed automotive applications, but is equally suited to industrial applications. It is intended to complement the In the late 1990s the Local Interconnect Network (LIN) Consortium was founded by five European automakers, Mentor Graphics (Formerly Volcano

associated unconditional frames before requesting the event-triggered frame again. 3. Sporadic frame. This frame is transmitted by the master as required, so a collision cannot occur. The header of a sporadic frame shall only be sent in its associated frame slot when the master task knows that a signal carried in the frame has been updated. The publisher of the sporadic frame shall always provide the response to the header. 4. **Diagnostic frame.** These always carry diagnostic or configuration data and they always contain eight data bytes. The identifier is either 60

(quartz or a ceramic). To ensure the baud rate-stability within one LIN frame, the SYNC field within the header is used. LIN protocol [edit]

any node requesting activity on the bus, either the LIN Master following its internal schedule, or one of the attached LIN Slaves being activated by its

There is an Inter Byte Space between the BREAK and SYNC field, one between the SYNC and IDENTIFIER, one between the payload and Checksum

The response is sent by one of the attached LIN slave **tasks** and is divided into data and checksum.^[9] DATA: The responding slave may send zero to eight data bytes to the bus. The amount of data is fixed by the application designer and mirrors data relevant for the application which the LIN slave runs in.

0x01 NXP (formerly Philips) Extra wire daisy chain **TBD** Reserved 0x04

1. Switchable resistive pull-up to V_{bat} 2. Pull-down to GND

Each configuration pin D_x (x=1-2) has additional circuitry to aid in the position detection.

1.3 The selected node takes the address from the LIN configuration message 1.4 The detected node turns on the pull-down at the output D₂ 2 Subsequent auto-addressing LIN messages

1. bus_in

Each slave node needs some additional circuitry compared to the standard LIN circuitry to aid in the position detection.

2.1 The first non addressed SNPD node is selected. It is identified by having the input D_1 low (D_2 of previous node).

1 Offset current measurement 1.1 All outputs pull-ups and current sources are switched off

LIN is not a full replacement of the CAN bus. But the LIN bus is a good alternative wherever low costs are essential and speed/bandwidth is not

important. Typically, it is used within sub-systems that are not critical to vehicle performance or safety - some examples are given below.

Cruise control, wiper, turning light, climate control, radio, wheel lock

Vehicle trim enhancement, sill plates illuminated with RGB LED

3.4 Current sources are switched off and pull-ups are switched on 3.5 The last node will accept the address contained in the LIN configuration message This technique is covered by the patents EP 1490772 B1 and US 7091876.

Applications [edit] Application segments Specific LIN application examples

Sensor, light sensor, light control, sun roof

Climate Small motors, control panel Mirror, central ECU, mirror switch, window lift, seat control switch, door lock

127 (0x7F) and it is a composition of supplier ID, function ID and variant ID. You can obtain a supplier ID by contacting CAN in Automation that is the authority responsible for the assignment of such identifiers.

2. ^ Mary Tamar Tan, Brian Bailey, Han Lin. "Microchip AN2059: LIN Basics

and Implementation of the MCC LIN Stack Library on 8-Bit PIC

4. ^ Steve Winder. "Power Supplies for LED Driving" \(\mathbb{L}\). p. 284

 List of network buses LIN-CP References [edit]

3. A "ATAN0049: Two-wire LIN Networking" ...

1. ^ . "ISO/AWI 17987-8" ₺.

Microcontrollers" m.

Standardization) as part of the process to be accepted as ISO standard ISO 17987 Part 1-7. LIN Specification v2.2A - archived

• LIN Supplier ID Registration Authority 🗗 Registration Authority for the LIN Supplier ID standardized in the ISO 17987 series Article about a free open hardware/software implementation of the LIN protocol

show

[show]

Technical and de facto standards for wired computer buses **Automation protocols**

Wikimedia Foundation, Inc., a non-profit organization. Privacy policy About Wikipedia Disclaimers Contact Wikipedia Mobile view Developers Statistics Cookie statement

Text is available under the Creative Commons Attribution-ShareAlike License 3.0; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the