LIN
Extra Wire Daisy Chain Slave Node Position Detection

Revision 1.0

DISCLAIMER

This specification as released by the LIN Consortium is intended for the purpose of information only and is provided on an "AS IS" basis only and cannot be the basis for any claims. The LIN Consortium will not be liable for any use of this Specification. The unauthorized use, e.g. copying, displaying or other use of any content from this document is a violation of the law and intellectual property rights.

The LIN Consortium and its members make no representation or assurance that the standard can be practiced without infringing the intellectual property rights of members of the LIN consortium or of third parties. Each user of this standard (whether or not a member of the LIN consortium) bears its own responsibility to determine if its implementation infringes intellectual property rights of third parties, and each user is responsible for acquiring any patent or other intellectual property rights it may require to produce its products.

The LIN Consortium disclaims any and all warranties with respect to the subject matter of this standard, which is expressly delivered AS IS."

REVISION HISTORY

Issue	Date	Remark
Revision 1.0	2008-12-10	1st Release

TABLE OF CONTENTS

1 SCOPE

This document is intended to describe one method for the detection of the position of a particular slave node in a LIN network with equal built slaves. This does not limit the use of position detection to the method described here.

This document covers the method Extra Wire Daisy Chain.

1.1 REFERENCES

[1] LIN Specification Package, Revision 2.1, 2006-11-24

[2] Electromagnetic compatibility (EMC)- Part 4-2: Testing and measurement techniques - Electrostatic discharge immunity test, IEC 61000-4-2: 1995

2 REQUIREMENTS

The specified methods must provide a means to assign a slave node with a unique node address within the particular LIN network, which can be used to configure the nodes according to LIN 2.1.

Any Slave Node Position Detection method should not violate the LIN Specification. In case an SNPD method violates the LIN Specification, these violations are described in the following chapters with the respective method descriptions. The behavior is described in the chapter “Limitations in Use” of the respective method description.

3 EXTRA WIRE DAISY CHAIN

3.1 CONTENTS OF THIS CHAPTER

This Chapter describes the Extra Wire Daisy Chain (XWDC) method and how it is being applied.

3.2 PRINCIPLE

Each slave node (SNPD node) has to provide two extra pins, one input pin D1 and one output pin D2 (Figure 3-1). The input of the first SNPD node is either set to low level (e.g. by connecting it to ground) or alternatively connected via a configuration wire to an output of the master. Another configuration wire is routed from the output D2 of the first SNPD node to the input D1 of the second and so on, resulting in a daisy chain.

In the beginning, the outputs of all SNPD nodes feature a high level. In case the NAD of a SNPD node is not configured and its input D1 is on low level, the SNPD node is selected and the node address configuration messages via the LIN bus are addressed to this SNPD node. After finishing the configuration, the SNPD node switches its output D2 to low level, which selects the second SNPD node, and so on. This way a SNPD node can signal to the next SNPD node in line that the next configuration messages apply to it.

The configuration can also work in two directions when inputs and outputs are implemented as bidirectional I/O. In order to configure the daisy chain in reverse direction, inputs D1 and outputs D2 are exchanged.

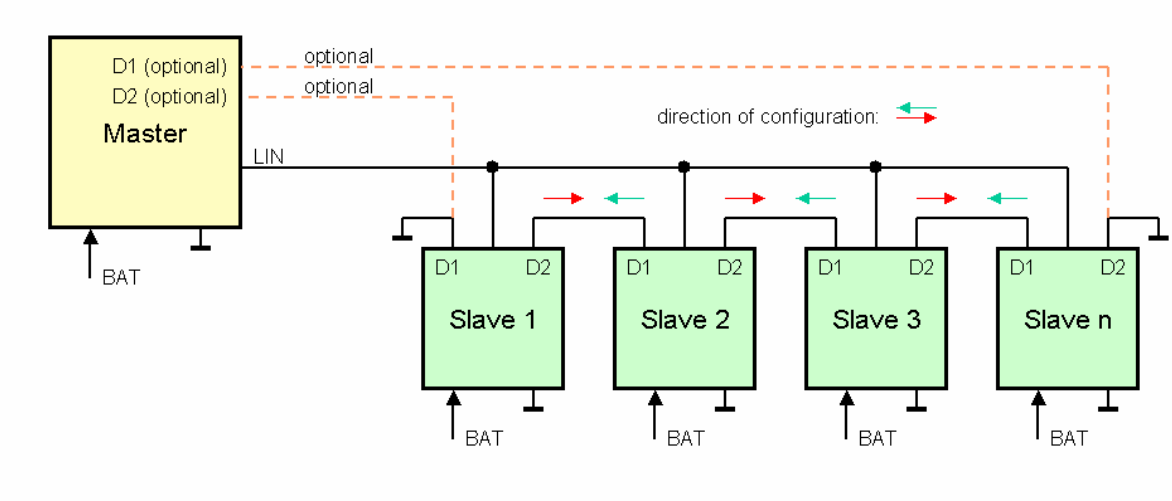

Figure 3-1: Block diagram of the Extra Wire Daisy Chain method

3.3 PHYSICAL LAYER

The principle circuit diagram of the Extra Wire Daisy Chain is shown in Figure 3-2.

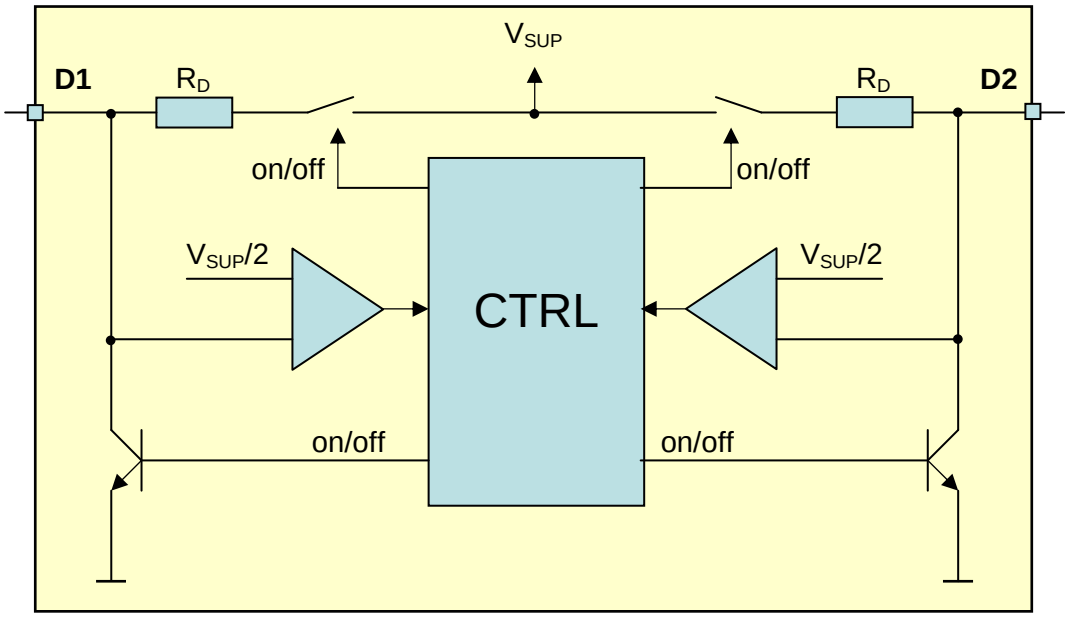

Figure 3-2: Principle circuit diagram of the Extra Wire Daisy Chain method

Switching on/off the pull-up resistors and drivers is controlled during the configuration process via the Extra Wire Daisy Chain sub functions (see section 3.4). Before starting the configuration and in case the configuration has finished, all pull up resistors and drivers shall be switched off.

Note

In case of a LIN communication time-out all slaves shall switch off their pull up resistors and drivers.

All voltages are defined with respect to ground; positive currents flow into the IC; **values are given for V_{SUP} operation range;** unless otherwise specified.

no.	symbol	parameter	min.	typ.	max.	unit	conditions
4.1.1	$V_{SUP\_NON\_OP}$	voltage range within which the device is not destroyed	-0.3		40	V	
4.1.2	V_{D_ESD}	ESD protection	2k[5]			V	IEC 61000-4-2: 1995 [2]
4.1.3	V_{SUP}	supply voltage operation range	7		18	V	
4.1.4	$I_{D\_NO\_GND}$[1]	daisy chain pin current when loss of ground	-3,6		1	mA	$0V < V_D < 18V$; $V_{GND} = V_{SUP} = 12V$
4.1.5	$I_{D\_NO\_BAT}$	daisy chain pin current when loss of battery			90	µA	$0V < V_D < 18V$; $V_{SUP} = V_{GND} = 0V$
4.1.6	I_{D_LEAK}[1]	daisy chain pin leakage current	-20		20	µA	$0V < V_D < 18V$; $V_{SUP} = 12V$
4.1.7	I_{D_LIM}[2,4]	daisy chain pin low-level output short circuit current	10		200	mA	$V_{SUP} = 18V$; $V_D = 18V$;
4.1.8	V_{D_ON}[2,4]	daisy chain pin voltage drop			0.2	V_{SUP}	$R_{LOAD} = R_D \| R_D = 2.5k\Omega$
4.1.9	R_D[3]	daisy chain pin pull-up	5	10	30	kΩ	
4.1.10	C_D	capacitance of daisy chain node			1	nF	
4.1.11	C_{DC}	total capacitance of daisy chain connection			8	nF	
4.1.12	V_{DC_LOW}	daisy chain low state			0.4	V_{SUP}	
4.1.13	V_{DC_HIGH}	daisy chain high state	0.6			V_{SUP}	
4.1.14	t_{pd_Sample}	Propagation delay: time from end of LIN frame until sampling of Daisy Chain input level			50	µs	See Figure 3-3
4.1.15	t_{pd_Level}	Propagation delay: time from end of LIN frame until Daisy Chain output level is turned high/low	100 [6]		$t_{diff\_Checksums} - t_{pd\_Settle}$ (Min)	µs	See Figure 3-3
4.1.16	t_{pd_Settle}	Propagation delay: time before end of LIN frame	100 [6]		$t_{diff\_Checksums} - t_{pd\_Level}$ (Min)	µs	See Figure 3-3

Table 3-1: Electrical Parameters of the Extra Wire Daisy Chain method

Notes:

1. Pull-up resistor is switched off
2. Pull-up resistor is switched on
3. Pull-up resistor must be switched off if Extra Wire Daisy Chain method is finished
4. The Extra Wire Daisy Chain-driver shall withstand a short-circuit to V_{SUP} = 18V
5. The required ESD level for automotive applications can be up to ± 8 kV at the connectors of the ECU
6. $t_{diff\_Checksums}$ = time between end of two directly following LIN frames, minimum and maximum values according to the LIN specification

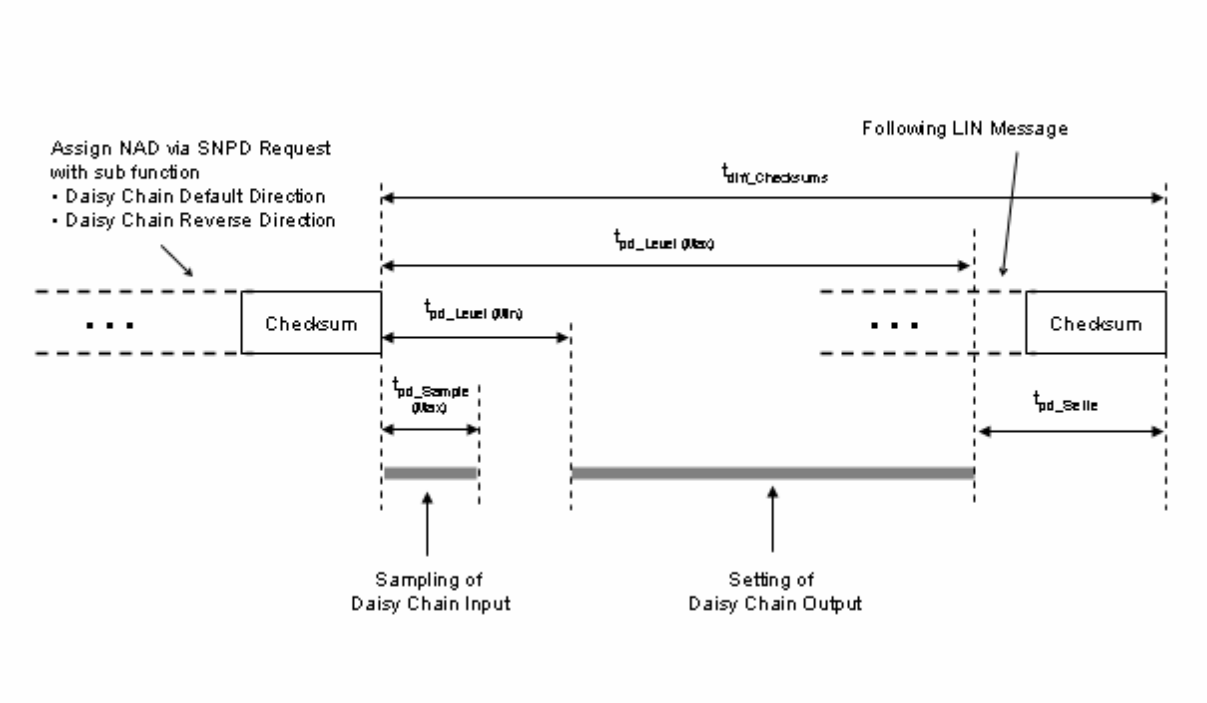

Figure 3-3: Timing of the Extra Wire Signal

3.4 SUB FUNCTIONS

The SNPD sub function IDs for the Extra Wire Daisy Chain method are summarized in Table 3-2.

SNPD sub function	SNPD sub function ID
Daisy Chain Initialization	0x01
Daisy Chain Default Direction (D1 input, D2 output)	0x02
Daisy Chain Reverse Direction (D2 input, D1 output)	0x03
Daisy Chain Finished	0x04

Table 3-2: SNPD sub function IDs of the Extra Wire Daisy Chain method

Realizing the Extra Wire Daisy Chain method, all SNPD sub functions are mandatory, except of the sub function Daisy Chain Reverse Direction (SNPD Sub Function ID 0x03).

Note

There is no SNPD response to the Daisy Chain Initialization and Daisy Chain Finished requests. The optional responses to the Daisy Chain Default Direction and Daisy Chain Reverse Direction requests shall only be provided by the SNPD node the requests were addressed for.

3.5 CONFIGURATION FLOW

The configuration flow is shown in Figure 3-4. Beginning with the Daisy Chain Initialization, all SNPD nodes switch their pull up resistors on and their output drivers off. Afterwards, the SNPD nodes are successively configured from the first SNPD node to the last. After receiving an Assign NAD via SNPD request with sub function Daisy Chain Default Direction, all slaves which NAD is not configured sample their input D1 (Figure 3-2). In case the input is on low level, the corresponding request is addressed for this node. After successful NAD assignment, the respective SNPD node sets its output D2 to low level (Figure 3-2), thus selecting the next SNPD node.

Between sending Assign NAD via SNPD requests with any XWDC sub function, the master can optionally send other LIN commands and thus e.g. continue configuration of the before selected SNPD Node and/or get SNPD responses. The configuration process ends by sending the sub function Daisy Chain Finished. After receiving this message, all SNPD nodes switch their pull up resistors and output drivers off.

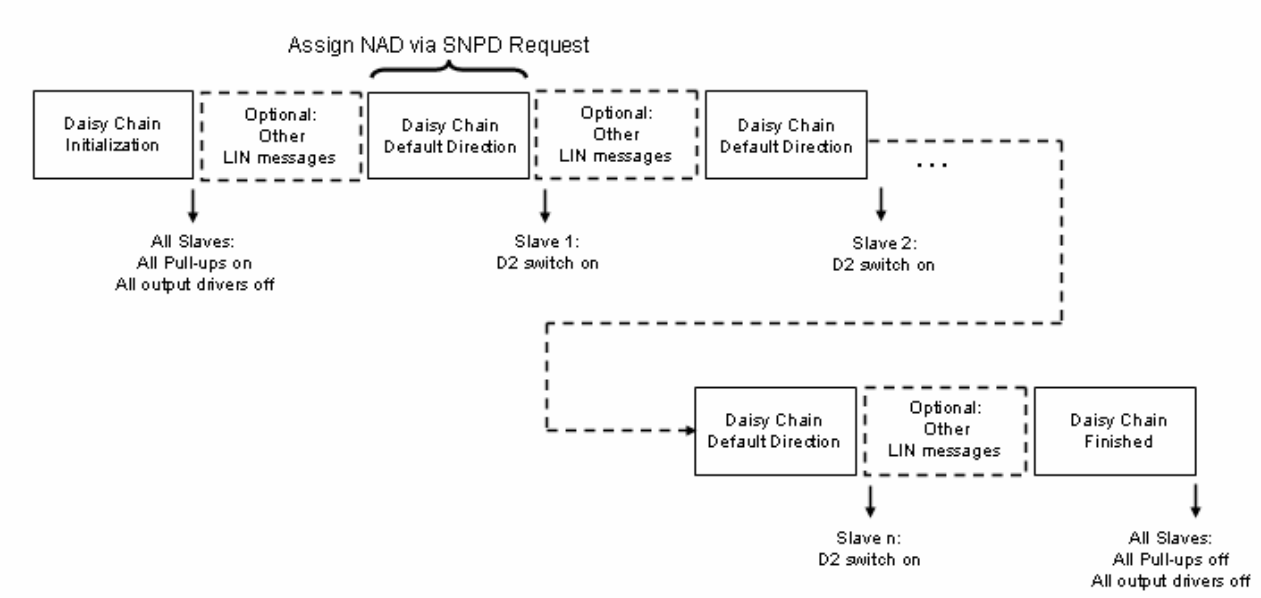

Figure 3-4: Configuration flow of the Extra Wire Daisy Chain method

A change to the reverse configuration direction is possible after each LIN command. In this case the master sends the Assign NAD via SNPD request with SNPD sub function Daisy Chain Reverse Direction, thus starting the configuration with the last slave n in the row.

Note:

SNPD nodes that have already been successfully configured after receiving a Daisy Chain Initialization request shall not participate again at the configuration process before a new Daisy Chain Initialization request is performed.

3.5.1 Daisy Chain Setup Flow in Detail (default direction)

Daisy Chain Initialization

Assign NAD via SNPD Request

Header 0x3C	+	NAD	PCI	SID	D1	D2	D3	D4	D5
		Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	unused
		0x7f	0x06	0xb5	0xff	0x7f	0x01	0x01	0xff

All SNPD Slaves:
All Extra Wire Daisy Chain pull-ups on
All Extra Wire Daisy Chain output drivers off

Optional: other (standard) LIN Messages

Assign NAD to slave 1 (default direction)

Assign NAD via SNPD Request

Header 0x3C	+	NAD	PCI	SID	D1	D2	D3	D4	D5
		Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
		0x7f	0x06	0xb5	0xff	0x7f	0x02	0x01	New NAD for Slave 1

1) All XWDC slaves set direction to default direction (D1 input; D2 output).
2) Slave 1 accepts now New NAD because it is “selected” (input is low).
3) Slave 1 sets output to dominant (low). Thus slave 2 becomes selected.

Optional: other (standard) LIN Messages

Assign NAD to slave 2 (default direction)

Assign NAD via SNPD Request

Header 0x3C	+	NAD	PCI	SID	D1	D2	D3	D4	D5
		Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
		0x7f	0x06	0xb5	0xff	0x7f	0x02	0x01	New NAD for Slave 2

1) All XWDC slaves set direction to default direction (D1 input; D2 output).
2) Slave 2 accepts now New NAD because it is “selected” (input is low).
3) Slave 2 sets output to dominant (low). Thus slave 3 becomes selected.

Optional: other (standard) LIN Messages

...

Assign NAD to slave n (default direction)

Assign NAD via SNPD Request

Header 0x3C	+	NAD	PCI	SID	D1	D2	D3	D4	D5
		Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
		0x7f	0x06	0xb5	0xff	0x7f	0x02	0x01	New NAD for Slave n

1) All XWDC slaves set direction to default direction (D1 input; D2 output).
2) Slave n accepts now New NAD because it is “selected” (input is low).
3) Slave n sets output to dominant (low).

Optional: other (standard) LIN Messages

Daisy Chain Finished

Assign NAD via SNPD Request

Header 0x3C +

NAD	PCI	SID	D1	D2	D3	D4	D5
Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	unused
0x7f	0x06	0xb5	0xff	0x7f	0x04	0x01	0xff

All XWDC slaves:
All pull-ups off.
All output drivers off.

3.5.2 Daisy Chain Setup Flow in Detail (reverse direction)

Daisy Chain Initialization

Assign NAD via SNPD Request

Header 0x3C +

NAD	PCI	SID	D1	D2	D3	D4	D5
Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	unused
0x7f	0x06	0xb5	0xff	0x7f	0x01	0x01	0xff

All SNPD Slaves:
All Extra Wire Daisy Chain pull-ups on
All Extra Wire Daisy Chain output drivers off

Optional: other (standard) LIN Messages

Assign NAD to slave n (reverse direction)

Assign NAD via SNPD Request

Header 0x3C +

NAD	PCI	SID	D1	D2	D3	D4	D5
Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
0x7f	0x06	0xb5	0xff	0x7f	0x03	0x01	New NAD for Slave n

1) All XWDC slaves set direction to reverse direction (D1 output; D2 input).
2) Slave n accepts now New NAD because it is “selected” (input is low).
3) Slave n sets output to dominant (low). Thus slave n-1 becomes selected.

Optional: other (standard) LIN Messages

Assign NAD to slave n-1 (reverse direction)

Assign NAD via SNPD Request

Header 0x3C +

NAD	PCI	SID	D1	D2	D3	D4	D5
Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
0x7f	0x06	0xb5	0xff	0x7f	0x03	0x01	New NAD for Slave n-1

1) All XWDC slaves set direction to reverse direction (D1 output; D2 input).
2) Slave n-1 accepts now New NAD because it is “selected” (input is low).
3) Slave n-1 sets output to dominant (low). Thus slave n-2 becomes selected.

Optional: other (standard) LIN Messages

...

Assign NAD to slave 1 (default direction)

Assign NAD via SNPD Request

Header 0x3C +

NAD	PCI	SID	D1	D2	D3	D4	D5
Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	New NAD
0x7f	0x06	0xb5	0xff	0x7f	0x03	0x01	New NAD for Slave 1

1) All XWDC slaves set direction to reverse direction (D1 output; D2 input).
2) Slave 1 accepts now New NAD because it is “selected” (input is low).
3) Slave 1 sets output to dominant (low).

Optional: other (standard) LIN Messages

Daisy Chain Finished

Assign NAD via SNPD Request

Header 0x3C	+	NAD	PCI	SID	D1	D2	D3	D4	D5
		Initial NAD			Supplier ID LSB	Supplier ID MSB	Function ID LSB	Function ID MSB	Unused
		0x7f	0x06	0xb5	0xff	0x7f	0x04	0x01	0xff

All XWDC slaves:
All pull-ups off.
All output drivers off.

3.5.3 Optional SNPD Response

Optional: other (standard) LIN Messages

SNPD Response of last assigned Frame

Positive Assign NAD via SNPD Response

Header 0x3D	+	NAD	PCI	RSID	D1	D2	D3	D4	D5
					Unused				
		0x7f	0x01	0xf6	0xff	0xff	0xff	0xff	0xff

The last SNPD addressed slave responds on this header when the last 0x3C-Command was an “Assign NAD via SNPD” request for the XWDC method.

3.6 LIMITATIONS IN USE

Since the Extra Wire Daisy Chain method works completely independent from the LIN Physical Layer and Protocol Layer the method does not impact the LIN Standard and therefore has no limitations.