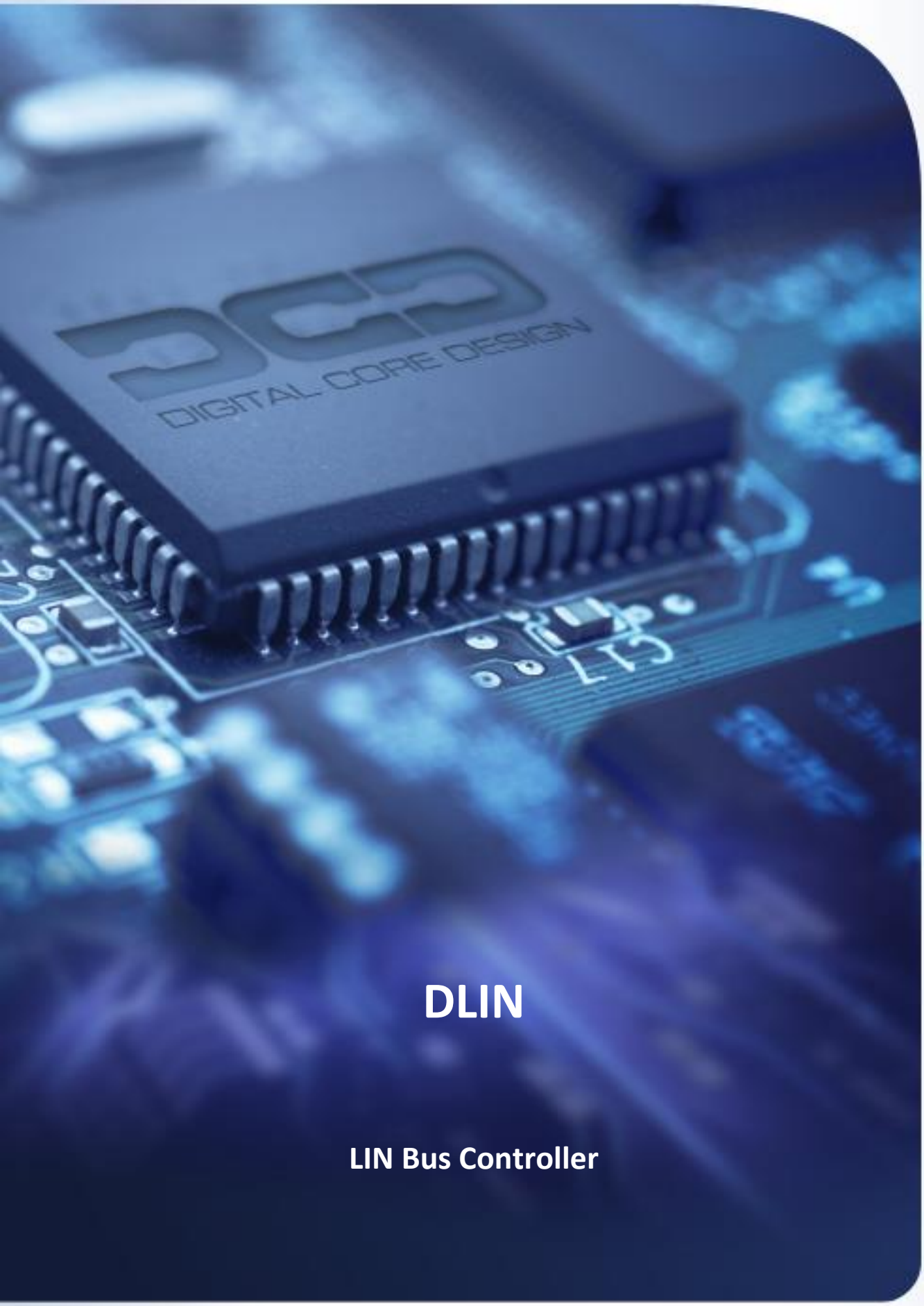
DIGITAL CORE DESIGN



HDL CORE SPECIFICATION

DLIN

LIN Bus Controller

ver. 1.22_zda

Contents

Figures

Tables

1. OVERVIEW

1.1. INTRODUCTION

The DLIN is a standalone controller for the Local Interconnect Network (LIN) used in automotive and industrial applications. DLIN conforms to LIN 2.1 specification and optionally compatibility with LIN 1.3. The DLIN allows serial transmission between 1kbit/s and 20 kbit/s.

1.2. DOCUMENT STRUCTURE

Document contains brief description of DLIN core functionality. This manual is intended for design engineers who are planning to use the DLIN HDL core in conjunction with simulation and synthesis tools.

1.3. LIN OVERVIEW

LIN (Local Interconnect Network) is a serial communication protocol, which was created to provide a cost efficient bus communication. The LIN specification is developed by LIN consortium (**http://www.lin-subbus.org**). The LIN standard includes the specification of the transmission medium, the interface between development tools, the transmission protocol and the interfaces for software programming. LIN has been created to decrement cost of automotive networks, it has been replaced the most expensive CAN in simple application (sensors or actuators). The LIN device can be implemented as a master or as a slave node.

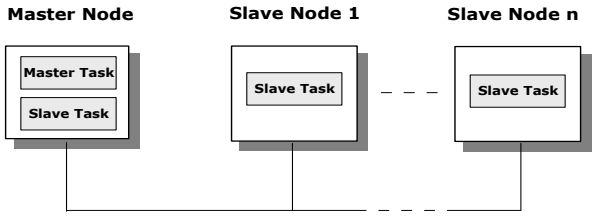


Figure 1. LIN network connection

Transmission is initiated by Master Node which sends the data frame to Slave Nodes (max. 15) throat one wire bus.

2. DELIVERABLES

- Source code:
  - VHDL Source Code or/and
  - VERILOG Source Code or/and
  - FPGA Netlist
- VHDL & VERILOG test bench environment
  - Active-HDL automatic simulation macros
  - ModelSim automatic simulation macros
  - Tests with reference responses
- Technical documentation
  - Installation notes
  - HDL core specification
  - Datasheet
- Synthesis scripts
- Example application
- Technical support
  - IP Core implementation support
  - 3 months maintenance
    - Delivery of the IP Core and documentation updates, minor and major versions changes
    - Implementation support and consulting
    - Phone & email support

3. FEATURES

3.1. KEY FEATURES

- Conforms to LIN 2.2 and LIN 1.3 specifications
- Automatic LIN Header handling
- Automatic Re-synchronization
- Data rate between 1Kbit/s and 20 Kbit/s
- Master and Slave work mode
- Time-out detection
- Extended error detection
- “Break-in-data” support
- Wake-up detection
- Go to Sleep detection
- Automatic Bit Rate detection (in slave mode)

3.2. CONFIGURATION

Some key parameters of DLIN controller are configured by setting constants in package:

PARAMETR NAME	DESCRIPTION
RST_ACTIVE	Reset active level
CS_ACTIVE	Chip select active level
WR_RD_ACTIVE	Write and read active level

Table 1. DLIN generic parameters description

The LIN configuration provides possibility to set active level of reset, chip select, write and read in the core. This configuration is enabled by constants RST_ACTIVE, CS_ACTIVE and WR_RD_ACTIVE. Set those parameters to ‘1’ force active level high and to ‘0’ to force active level low. Default setting:

```
constant RST_ACTIVE      :STD_LOGIC:='1'; --VHDL
constant CS_ACTIVE       :STD_LOGIC:='0'; --VHDL
constant WR_RD_ACTIVE    :STD_LOGIC:='0'; --VHDL
```

4. DLIN HARDWARE DESCRIPTION

4.1. DLIN SYMBOL

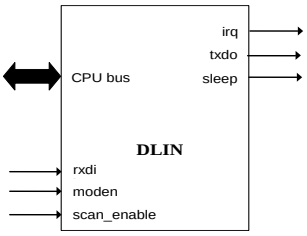


Figure 2. DLIN symbol

4.2. PINS DESCRIPTION

Pin functionality is described in the following table. All pins are one directional. There are no three state output pins and internal signals.

PIN	TYPE	ACTIVE	DESCRIPTION
moden	input	high	Global module enable
scan_enable	input	high	Enable scan mode when '1'. Should be '0' for normal work
rxdi	input	-	LIN receive data
irq	output	High	Interrupt signal
txdo	output	-	LIN transmit data
sleep	output	High	Sleep mode indicator

Table 2. DLIN pins description

The CPU interface is described in chapers below. Active pin depends on selected interface type.

4.3. APB CPU INTERFACE

The interface below is active when ***APB_IF*** is defined. Read/Write timings are compatible with Advanced Peripherals Bus standard.

`define APB_IF

PIN	TYPE	ACTIVE	DESCRIPTION
pclk	input	-	Global clock (CLK period source)
presetn	input	low	Global reset
psel	input	high	Peripheral select - used for read/write strobes
penable	input	high	Peripheral enable - used for read/write strobes
pwdata (31:0)	input	-	Peripheral write data bus
paddr (31:0)	input	-	Peripheral address
pstrb(3:0)	input	high	Peripheral strobe - select byte(s) to write on PWDATA bus
pwrite	input	high	Peripheral write when HIGH else read
prdata(31:0)	output	-	Peripheral read data bus
pready	input	high	Peripheral ready pin

Table 3. DLIN CPU bus - APB interface pins description

Note: When the APB interface is selected, the byte access to DLIN device is only possible.

DLIN read and write APB timings are shown in figures below.

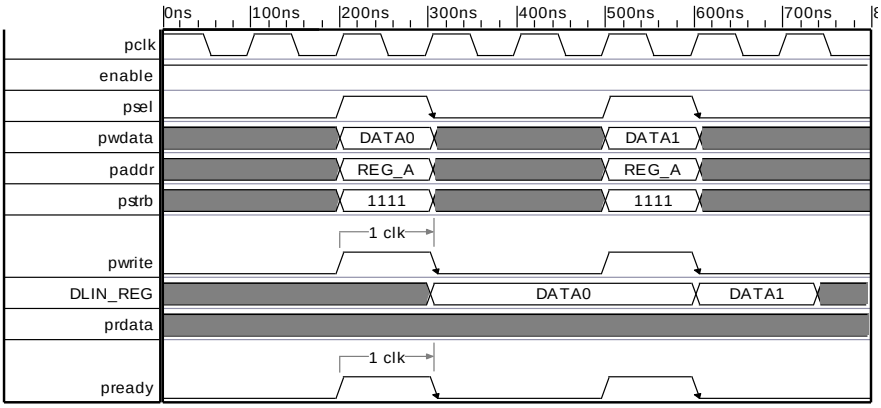


Figure 3. APB data write to DLIN registers timing

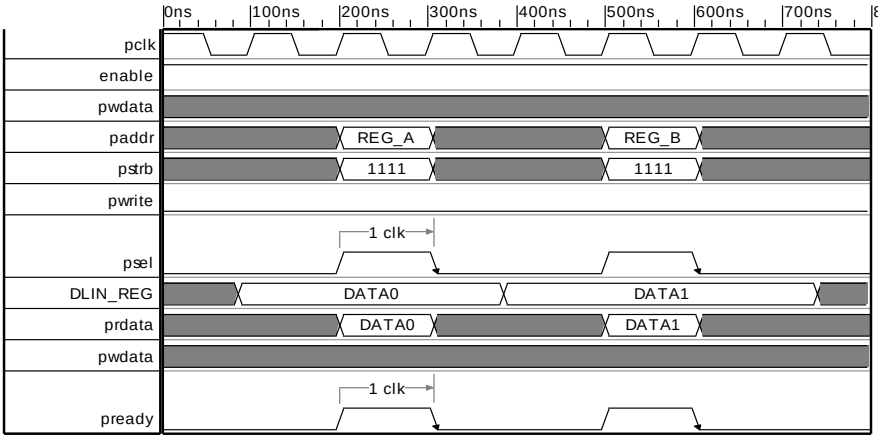


Figure 4. APB data read from DLIN registers timing

Notes:

REG_x - register address
DATAx - written data into register
DLIN_REG - read/written register inside DLIN

4.4. DEFAULT INTERFACE

The interface below is active no other is defined.

PIN	TYPE	ACTIVE	DESCRIPTION
clk	input	-	Global clock (CLK period source)
rst	input	High	Global reset
cs	input	Low	Chip select
rd	input	Low	Read data strobe
wr	input	Low	Write data strobe
addr(3:0)	input	-	Address bus
datai(7:0)	input	-	Input data bus
datao(7:0)	input	-	Output data bus

Table 4. DLIN CPU bus - APB interface pins description

DLIN read and write timings are shown in figures below.

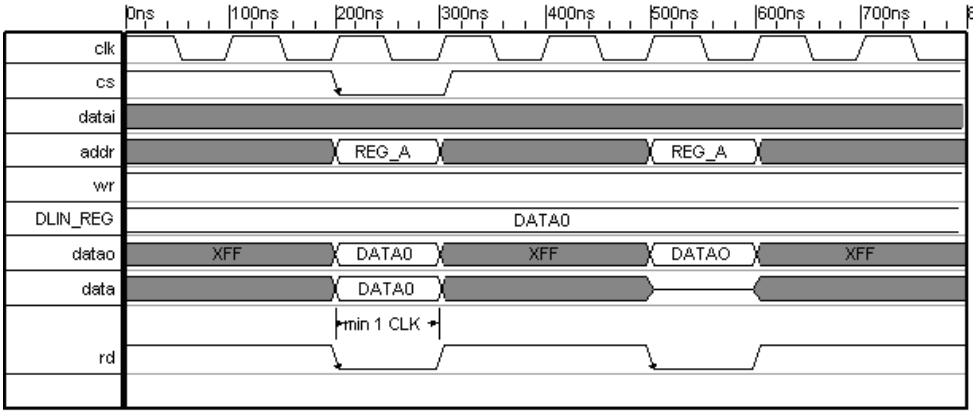


Figure 5. Data read from DLIN registers timing

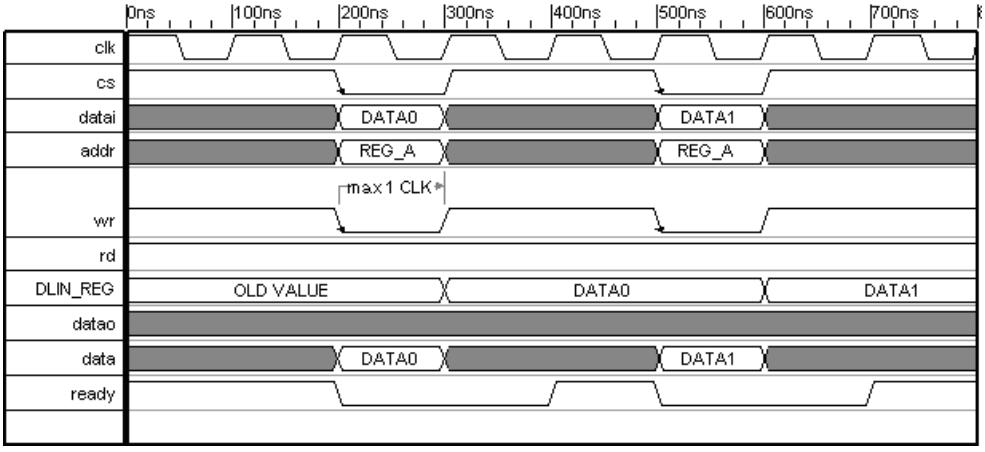


Figure 6. Data write to DLIN registers timing

Notes:

REG_x - register address
DATAx - written data into register
DLIN_REG - read/written register inside DLIN

4.5. BLOCK DIAGRAM

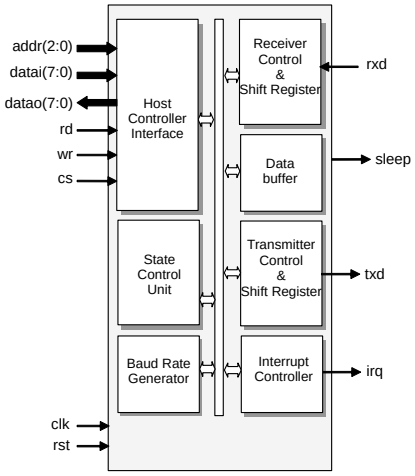


Figure 7. DLIN block diagram

4.5.1. BAUD RATE GENERATOR

The DLIN contains a programmable 15 bit baud generator which divides clock input by a divisor in the range between 1 and (2^{15}–1). The formula for the baud rate is:

$$BR = \frac{f}{(16 \cdot Divisov)}$$

- **Divisor - DLH:DLL** baud rate registers

Divisor value depends on two registers, called divisor latches ***DLL*** and ***DLH***, which store the divisor in a 15-bit binary format (the 16-th bit in ***DLH*** register response for synchronization). These divisor latches must be loaded during initialization of the DLIN in order to ensure desired operation of the baud generator. When either of the divisor latches is loaded, a 15-bit baud counter is also loaded on the ***CLK*** rising edge following the write to ***DLL*** or ***DLH*** to prevent long counts on initial load.

4.5.2. INTERRUPT CONTROLLER

As shown in figure below there is six flags which can be sources of interrupt. Five of those flags are information flag and one is error flag. The information flags are:

- LID flag - is set then header has been successfully received
- RDY flag - is set then last command has been finished
- ABORT flag - when is set is indicates that abort command has been executed
- SLEEP flag - is set then DLIN decode ***go to sleep*** command or when LIN bus inactivity has been detected
- WAKEUP flag - is set when ***wake-up*** signal has been detected on LIN bus

The error flag is logical 'OR' of all bits in ***LER*** register. There are seven causes of error:

- Framing error
- Bit error
- Parity error
- Checksum error
- Frame timeout error
- Synchronization error
- Overrun error
- Wake-up error

There all flags can generate interrupt if the corresponding enable bit is set in ***LIE*** register. When DLIN module detect some error is break currently executed command and enter to error state. Within this state it not execute any command or detect any transaction on the bus until ***LSR, LER*** and ***LBS*** registers will be read. Reading those registers at the same time reset the interrupt output.

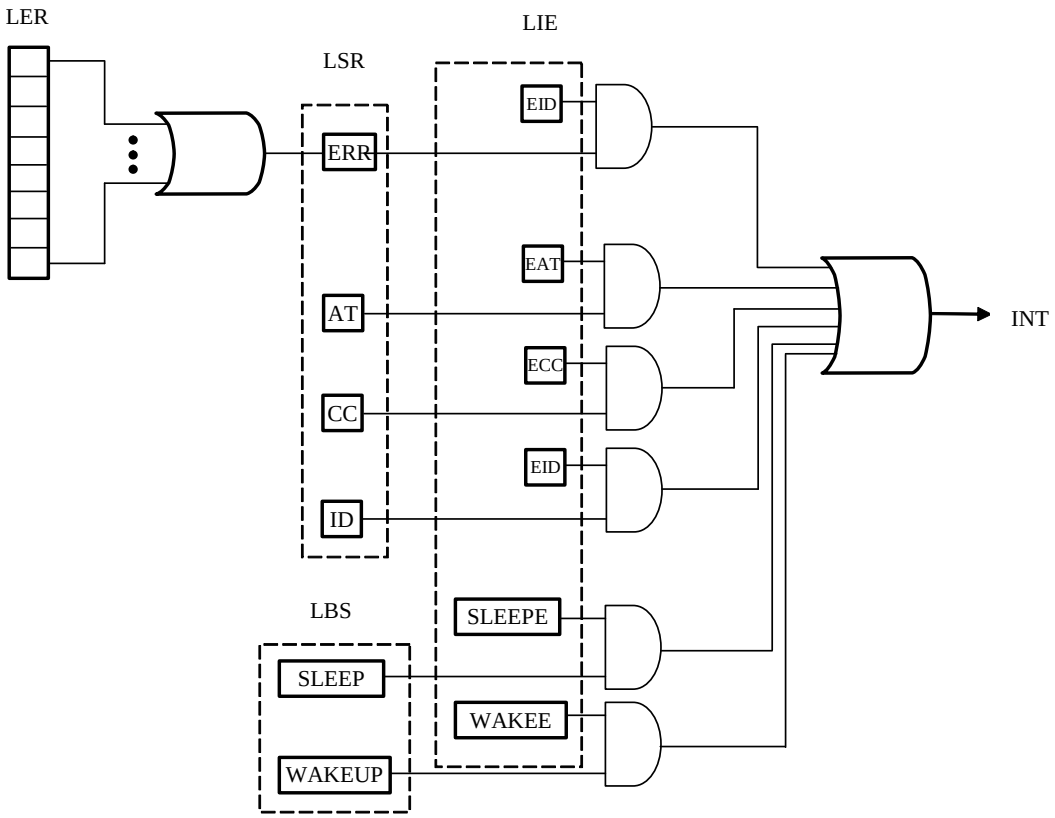


Figure 8. Interrupt controller

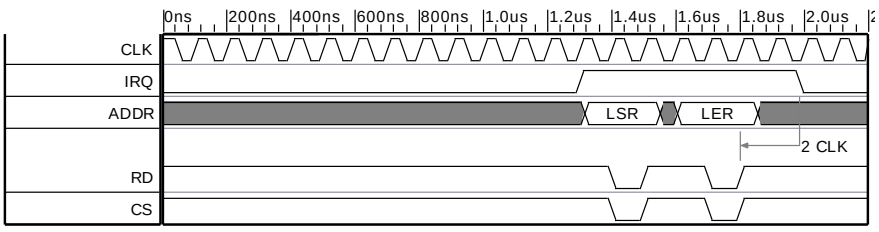


Figure 9. Erase interrupt timing

4.6. DLIN REGISTERS

DLIN includes internal registers which are used to configure the kind and mode of transmission.

REGISTER	Description	Address
LBUF	Data buffer register	0x00
LSEL	Select register	0x01
LID	ID register	0x02
LER	Error register	0x03
LIE	Interrupt enable register	0x04
LSR	Status register	0x05
LCR	Control register	0x05
DLL	Baud rate register	0x06
DLH	Baud rate register	0x07
HDRL	Header Delay Low register	0x08
HDRH	Header Delay High register	0x09
HDP	Header Delay Prescaler	0x0A
LBS	Lin Bus Status register	0x0B
WURT	Wake-Up Response Timeout register	0x0C
IDTL	Idle Detection Time (low)register	0x0D
IDTH	Idle Detection Time (high) register	0x0E

Table 5. DLIN registers

4.6.1. DATA BUFFER

The LBUF is used to write or read data frame destined to send or received via LIN bus. The data buffer in DLIN is designed as FIFO, where ***INDEX*** field is the address pointer to the required data byte. The data byte can be read or written through ***LBUF*** register. Depending of ***AINC*** bit in ***LSEL*** register the data pointer can be automatically incremented after each access to ***LBUF***. The first byte in the LIN frame is stored at the ***INDEX=0*** and the last one is stored at the ***INDEX=7***. In log mode, it is possible to access the ninth byte of FIFO, which is in this case the checksum byte. When DLIN executes RX response or TX response command it starts from first byte od data buffer (***index=0***).

LBUF (0x00)

7	6	5	4	3	2	1	0	Reset
LBUF.7	LBUF.6	LBUF.5	LBUF.4	LBUF.3	LBUF.2	LBUF.1	LBUF.0	0x00

Figure 10. LIN Data buffer

4.6.2. LSEL REGISTER

LSEL register is used to configure the FIFO memory: Auto increment index of FIFO memory after set a value into FIFO; INDEX bits are used to select used to select the place of FIFO index where the data will be setting.

LSEL (0x01)

7	6	5	4	3	2	1	0	Reset
AINC	-	-	-	INDEX				0x00

Figure 11. LIN Select register

AINC — Auto increment FIFO index after access
- 0 - Auto increment mode disabled
- 1 - Auto increment mode enabled

INDEX — Location of the LIN response data byte into the FIFO data buffer.

4.6.3. LID REGISTER

This register contains ID which is sent in header frame in master mode. In case of slave mode this register contains received ID.

LID (0x02)

7	6	5	4	3	2	1	0	Reset
-	-	ID						0x00

Figure 12. LIN Identifier register

ID — Indicates LIN bus identifier.
If LCR[5:2] are "1111" then bits LID[5:4] indicates number of data in message frame:
- 00 – 2 bytes
- 01 – 2 bytes
- 10 – 4 bytes
- 11 – 8 bytes

4.6.4. LIN ERROR REGISTER

LIN Error Register is read only register which provides information about reason of occurred error. All flags are automatically set when corresponding event occurs and cleared by reading this register. The ***WAKEER*** bit is used only when DLIN is in slave mode. It set when master does not respond to wake-up signal within time specified by ***WURT*** register. The second case where this bit is set is when the response counter is running (after sending the wake-up signal) and new data has been written to ***WURT*** register. If DLIN is in master mode, this bit is not changed. The ***OVER*** bit is set when current command is steel in progress when then new command has been written to ***LCR*** register. The ***SYNCER*** bit is the another bit which is use in slave mode only. This is set when the synchronizaction field measurement comes out after the specified range. The ***TOVER*** indicates that frame was not completed within maximum time lenght or received header not complete within maximum time length. The ***CHKSER*** indicates error checksum calculation in received frame. The received header with incorrect pariry bits is indicated by set PER bit. When DLIN transmit frame it monitores bit value on the bus, if received bit is not the same as transmitted bit the ***BITER*** flag is set. The ***FER*** bit is set when DLIN does not detect proper value of stop bit (recessive on LIN bus).

LER (0x03)

7	6	5	4	3	2	1	0	Reset
WAKEER	**OVER**	**SYNCER**	**TOVER**	**CHKSER**	**PER**	**BITER**	**FER**	**0x00**

Figure 13. LIN error register

WAKEER			Wake up error flag (slave node only).
	0	-	No wake-up error has been detected.
	1	-	Wake-up response from master node not has been received.
OVER			Overrun error flag. This flag indicates that new command was entered while RDY flag is low.
	0	-	No overrun error has been detected.
	1	-	Overrun error has been detected.
SYNCER			Synchronization error flag. (slave mode only)
	0	-	No error with the SYNC field has been detected
	1	-	Edges of SYNC field are outside of the maximum tolerance
TOVER			Frame time out error flag.
	0	-	No timeout error has been detected.
	1	-	Timeout error has been detected.
CHKSER			Checksum error flag. This flag indicates that inverted modulo-256 sum of all received data bytes added to the checksum does not result in 0xFF
	0	-	No checksum error has been detected.
	1	-	Checksum error has been detected.
PER			Parity error flag. This flag indicates that received header did not have correct Protected Identifier field.
	0	-	No parity error has been detected.
	1	-	Parity error has been detected in received header
BITER			Bit error flag. This flag indicates that bit value monitored on bus is different from the bit value that is sent
	0	-	No bit error has been detected.
	1	-	Bit error has been detected.
FER			Framing error flag. This flag indicates that the received character did not have a valid Stop bit.
	0	-	No framing error has been detected.
	1	-	Framing error has been detected.

4.6.5. INTERRUPT ENABLE REGISTER

This register enables the four types of DLIN interrupts. Each interrupt can individually activate the interrupt ***IRQ*** output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 5 of the LIN Interrupt Enable register. Similarly, setting bits of the ***LIE*** register to logic 1, enables the selected interrupt(s). Figure below shows the contents of the LIE details on each bit follow.

LIE (0x04)

7	6	5	4	3	2	1	0	Reset
-	-	WAKEE	SLEEPE	ABORTE	ERRE	RDYE	LIDE	0x00

Figure 14. LIN Interrupt Enable register

WAKEE			Enable wakeup detection interrupt
	0		Wake-up interrupt disable
	1		Wake-up interrupt enable
SLEEPE			Sleep condition interrupt.
	0		Sleep interrupt disable
	1		Sleep interrupt enable
ABORTE			Enable ABORT Interrupt
	0	-	Abort command interrupt disabled
	1	-	Abort command interrupt enabled
ERRE			Enable Error Interrupt
	0	-	Error interrupt disabled
	1	-	Error interrupt enabled
RDYE			Enable ready interrupt
	0	-	Ready interrupt disabled
	1	-	Ready interrupt enabled
LIDE			Enable header receive interrupt. (slave mode only)
	0	-	Header received interrupt disabled.
	1	-	Header received interrupt enabled.

4.6.6. LIN STATUS REGISTER

LIN Status Register is a read only register which provides DLIN status information to the host CPU. All flags are automatically set when corresponding event occurs. When ***ERR*** bit is set it means that some errors were encountered. To find the source of error is read from the ***LER*** register (see the next section). Bit LID (LSR[0]) is cleared upon reading LID register. ERR bit (LSR[2]) is cleared when the ***LER*** register is read.

LSR (0x05)

7	6	5	4	3	2	1	0	Reset
LOG SIZE				ABORT	ERR	RDY	LID	0x02

Figure 15. LIN status register

LOG SIZE		-	Detected size of received message , available only when log mode is enable. In others cases return 0000
ABORT			Abort command flag. Abort transfer command is executed, while BUSY flag is set.
	0	-	no abort command
	1	-	current command execution has been aborted
ERR			Error flag.
	0	-	no error
	1	-	indicates transmission error
RDY			Ready flag. This bit is set after command executed.
	0	-	current command is in the progress
	1	-	indicates idle state
LID			Header receive flag. (slave mode only)
	0	-	no header received
	1	-	Complete header has been received.

4.6.7. LCR REGISTER

Control Register is used for controlling transmission frame. It is write only register, reading of this address return contents of status register.

LCR (0x05)

7	6	5	4	3	2	1	0	Reset
MODE	**LIN13**	**SIZE**				**CMD**		**0x00**

Figure 16. LIN control register

MODE — Master/slave mode select
- 0 - Slave
- 1 - Master

LIN13 — Checksum selector
- 0 - Enhanced checksum, compatible with LIN 2.1 specification
- 1 - Classic checksum, compatible with LIN 1.3 specification

SIZE — LIN 2.1 mode:
Length of message :
0000 – 0 data bytes
0001 – 1 data byte
0010 – 2 data bytes
0011 – 3 data bytes
0100 – 4 data bytes
0101 – 5 data bytes
0110 – 6 data bytes
0111 – 7 data bytes
1000 – 8 data bytes
1001 – 1011 reserved
1100 = Automatic Bit rate detection
1101 – wake-up mode
1110 – switching to log mode
1111 – switching frame length indicator to LID[5:4]

CMD — DLIN commands:
00 – Abort command
01 – Send header command (master mode only) when SIZE[3:0]!="1101"
01 – Send wake-up command when SIZE[3:0]=="1101"
01 – Send Automatic Bit Rate detection command when SIZE[3:0]=="1100"
10 – TX response command
11 – RX response command

4.6.8. DLL AND DLH REGISTER

The sync bit is used for enabling synchronization process (Slave mode only). Synchronization process uses ***SYNC*** field in transmitted header to synchronize slave data with master clock. It is based on time measurement between falling edges of the pattern. After finished synchronization ***SYNC*** bit is automatically clear by hardware.

DLL (0x06)

7	6	5	4	3	2	1	0	Reset
DLL.7	**DLL.6**	**DLL.5**	**DLL.4**	**DLL.3**	**DLL.2**	**DLL.1**	**DLL.0**	**0x00**

Figure 17. Data baud rate register (low)

DLL — Baud rate divider least significant bits.

DLH (0x07)

7	6	5	4	3	2	1	0	Reset
SYNC	**DLH.6**	**DLH.5**	**DLH.4**	**DLH.3**	**DLH.2**	**DLH.1**	**DLH.0**	**0x00**

Figure 18. Data baud rate register (high)

SYNC — Synchronization mode (slave mode only)
- 0 - Synchronization mode disabled
- 1 - Synchronization mode enabled

This bit is automatically cleared after detection of the synchronization field

DLH — [6:0] Baud rate divider most significant bits.

DLL register with DLH register form the 15 - bits baud rate register that is used for dividing the clock frequency. (see details in section 4.3.2)

4.6.9. HDR REGISTER

The ***HDR*** it is 16-bit register which specifies additional delay between toggle Send Header command, by write to ***LSR*** register, and start Frame Header sending on LIN bus. After write to ***LSR*** register value "xxxxxx10" the value from ***HDR*** register will be copied to internal counter. This counter will be decremented when the prescaler counter reaches zero. When the ***HDR*** counter reaches 0 value, DLIN starts sending Frame Header. Writing new value to ***HDR*** register during countdown has no effect until current command is not finished. If the ***HDR*** register value is 0 and Send Header command is toggle, the Frame Header will be send immediately.

HDRL (0x08)

7	6	5	4	3	2	1	0	Reset
HDR.7	HDR.6	HDR.5	HDR.4	HDR.3	HDR.2	HDR.1	HDR.0	0x00

Figure 19. Header Delay register (low)

HDRH (0x09)

7	6	5	4	3	2	1	0	Reset
HDR.15	HDR.14	HDR.13	HDR.12	HDR.11	HDR.10	HDR.9	HDR.8	0x00

Figure 20. Header Delay register (high)

4.6.10. HDP REGISTER

The HDP it is 8-bit prescaler register which specifies interval of decrementing ***HDR*** register. Content of this register is decremented in every clock cycles when Send Header command is toggle. When counter reach zero the ***HDR*** register is decremented. Write new value to ***HDP*** register during countdown has no effect until current command will not finish.

HDP (0x0A)

7	6	5	4	3	2	1	0	Reset
-	-	HDP.5	HDP.4	HDP.3	HDP.2	HDP.1	HDP.0	0x00

Figure 21. Header Delay Prescaler register

4.6.11. LBR REGISTER

The ***LBR*** it is 8-bit registers which contains information about LIN bus status. The ***SLEEPM*** bit is the only bit which can by set/clear by cpu host. Access to this bit is allowed only no command is executed. This bit can be automatically clear when wake-up signal is detected on LIN bus. The detection of wake-up signal is enabled only when ***SLEEPM*** bit is set. The ***WAKEUP*** bit is set when the wake-up signal is detected on LIN bus. The ***SLEEP*** bit indicates detection of sleep mode. This bit is set in one of two cases. When master send go to sleep command and when ***IDT*** timer detects inactivity of LIN bus (if ***IDTRUN*** bit is set). When one of this continion occur the ***SLEEP*** bit is set and interrupt request in generated. The detection if sleep mode does not entry DLIN in low-power mode. It inform the application aboat sleep mode event. Read from this register automatically clear ***WAKEUP*** bit and ***SLEEP*** bit.

LBR (0x0B)

7	6	5	4	3	2	1	0	Reset
-	-				SLEEP	WAKEUP	SLEEPM	0x00

Figure 22. LIN bus statu register

SLEEP		Sleep mode detection
	0 -	No Sleep condition occurred
	1 -	Sleep condition occur
WAKEUP		Wake-up signal detection
	0 -	No wake-up signal
	1 -	Wake-up signal detected
SLEEPM		Sleep mode.
	0 -	DLIN is normal mode
	1 -	DLIN is in sleep mode

4.6.12. WURT REGISTER

The register contains 8 - bit value witch is compared with wake-up response timer value in waiting for response state. When DLIN work as slave and it sent the wake-up signal it reset and run wake-up response timer, this timer is increment at every bit time tick divided by 32. When counted value and WURT value are equal and no response detected then WUER bit is set and interrupt request is generated.

WURT (0x0C)

7	6	5	4	3	2	1	0	Reset
WURT[7..0]								0x00

Figure 23. Wake-up response time register

4.6.13. IDTL AND ITDH REGISTER

The register contains 13- bit value which is compared with idle detection timer value. This timer is reset and run when dominant to recessive transition is detected, is increment at every bit time tick divided by 32. To enable counting the IDTRUN bit must be set, in this situaltion write to IDT[12:0] is locked. In order to save new value to IDT register, first clear IDTRUN bit and then write new value to IDT[12:0].

IDTL (0x0D)

7	6	5	4	3	2	1	0	Reset
IDT[7..0]								0xFF

Figure 24. IDT low register

IDTH (0x0E)

7	6	5	4	3	2	1	0	Reset
IDTRUN			IDT[12:8]					0x1F

Figure 25. IDT high register

5. FUNCTIONALITY

5.1. RECEIVER CONTROL

The receiver unit in the DLIN starts when the falling edge on ***RXD*** input is detected. After that the ***RXD*** input is sample every 16*baud cycles as it is shown in figure below. When the last data bit is sampled, the stop bit is sampled at the middle of the bit. If the it is the ***CRC*** field of frame or ***PID*** frame of header the end of receive is indicated (in the middle of bit). Note that: LIN bus specification requires that response-space must be nonnegative, in this case application must wait a minimum half of the bit time before start sending data.

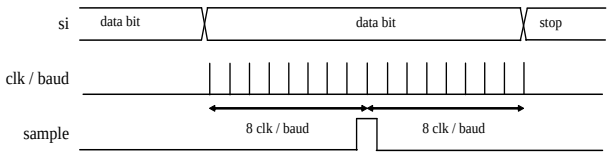


Figure 26. Sample clock cycle

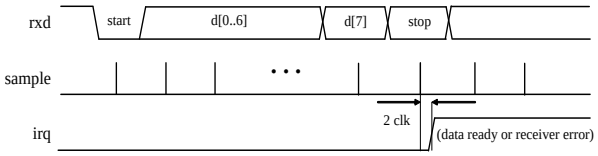


Figure 27. Receiver timing

5.2. LOG MODE

The DLIN core in LOG mode is very similar to receive state. The main difference is that it has unknown frame size. To switch the DLIN to log mode, set “xx111011” to ***LCR*** register, in response of header received. After that, the DLIN is configured to receive data with maximum frame timeout. When data is incoming, is stores in FIFO and data counter increments. When maximum timeout is reached, end of command is signalized by setting RDY bit in ***LSR*** register. When polling mode is used to check finishing of operation, only LSR register should be read. Reading LER register in polling mode checking loop will cause the log mode to work incorrectly. Number of received data is stored in the ***LSR[7:4]*** register and frame data is saved in FIFO. FIFO can hold 8 bytes of frame data plus one byte of frame CRC. After the log mode comand has been finished, the LER register should be read to prepare DLIN to receive next command, regardless of whether the ERR bit in LSR register was set or not.

5.3. HEADER DELAY

The DLIN core is equipped with a counter which inserts delay between triggering the Send Header command and its execution. The Header delay module consists of two counters : prescaler counter and main counter. The prescaler counter specifies time intervals of the main counter decrement. The Header delay counter starts counting down when the write "1xxxxx01" value to ***LCR*** register occur. After that, contents of ***HDP*** register loads to prescaler counter and contents of ***HDR*** register are loaded to main header delay counter. The prescaler counter starts to operate, it is decrement in every clock cycle until reaches zero. In this moment, value is reloaded again and overflow signal is generated. This signal causes the main counter decrement, when reaches zero run transmitting Frame Header on LIN bus. This feature is available for master mode only and value written to HDR register must be greater than zero.

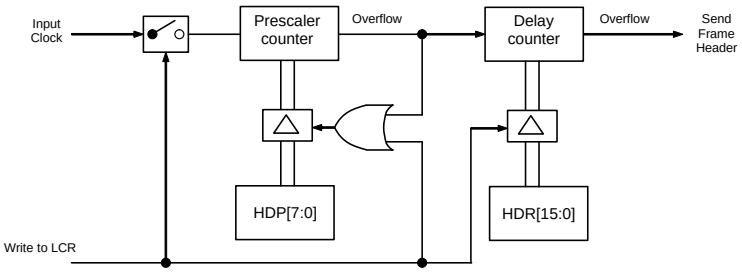


Figure 28. Header Delay block diagram

5.4. SLAVE SYNCHRONIZATION

Solid communication via LIN bus required bit rate tolerance be within certain limits. Each transmission is initiated by the master node, it causes the slave nodes must synchonized to the master baud rate. To meet this requirement the master, at the begining of the frame, sending synchronization byte following the break field. The data of the synchronization byte is equal 0x55 and is composed of five falling edges that can be used to adjust to refenece baud rate.

For master node the deviation from nominal bit rate must be less than 0.5%. The slave node there are two different accuracies : <14% for the reception of break and sync byte, <1,5% for reception or transmitting of remainder of the LIN frame. These two requires are illustrated figure below.

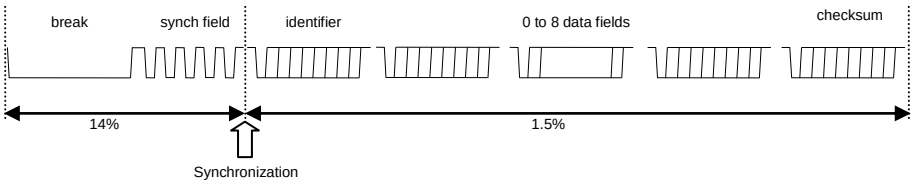


Figure 29. Slave node baudrate tolerances

The synchronization byte is used as a fixed reference clock and it is measured by dedicated timer clocked by local clock source. To enable measurement of the synchronization field and resynchronization in the slave node, the ***SYNC*** bit in the ***DLH*** register must be set. The synchronization process starts after at the successful detection of break field. Then the DLIN waits for start bit, when this occurs the edges counter and duration timer are cleared. At the next falling edge timer start counting until it is count four falling edges, which stop the measurement and new clock divisor value has been calculated. If the measured value is within the specified tolerance range(±14%) registers ***DLL*** and ***DLH*** are updated. In the others cases calculated value is rejected and the ***SYNCER*** bit is set. When the stop bit of sychronization field will be detected the ***SYNC*** bit is automatically cleared regardless of the result.

When ***SYNC*** bit in ***DLH*** register is not set in slave node, in this case the synchronization field is used to check deviation from nominal baud rate generated by master node. Then measured value is beyond specified range (±1.5%) the ***SYNCER*** is set.

5.5. BREAK - IN - DATA

The LIN bus specificatin requres that LIN controller can detect the BREAK/SYNC field sequence at any moment, even if DLIN expect response data byte field. In the case when BREAK/SYNC field sequence happens, the currently executed command is aborted and the process of receiving a new frame beguns.

- On slave node, an error is generated, depending of current executed command it can be framming error or bit error, which should be serviced before new header will be completly received.
- On master node, first firmware must abort current executing command by calling ABORT, which is signalized by set ABORT bit in LSR register. After that apply the Send header command, which start transmitting new LIN header.

5.6. SLEEP MODE

The LIN bus specification definies a sleep mode to reduce the system power consumtion. The master node use master request frame (ID=0x3C) to brodcast go to sleep command. The sleep mode request must be transmitted as a normal transmit message, send header with ID=0x3C, after that, send 8 bytes data which shown on figure below.

Data1	Data2	Data3	Data4	Data5	Data6	Data7	Data8
0x00	**0xFF**	**0xFF**	**0xFF**	**0xFF**	**0xFF**	**0xFF**	**0xFF**

Figure 30. Go to sleep command

The frame identifier 0x3C is a diagnostic frame identifier and is always use classic checksum calculation method. The LIN slave node must decode the go to sleep command from message identyfier and data bytes. After that, DLIN set ***SLEEP*** bit in ***LBS*** register and interrupt is generated (if ***SLEEPE*** bit is set). The secund condition to set ***SLEEP*** bit is detects LIN bus inactivite. It is necessery to determine the time after which the LIN bus is recognize as idle. In specification this time is between 4 to 10 second. The formula for idle detection timeout is :

$$IDT = \frac{tov}{(bt \cdot 32)}$$

where:

tov - is the timer timeout

bt - is the bit time set in by baudrate divisor

For example if your baudrare is 19200 b/s the bit time is 52 us. When the idle state must be detected after 4 s the 2500d (0x9C4) should be written to IDT register.

When ***SLEEP*** bit is set the application should put DLIN into the Sleep mode by set ***SLEEPM*** bit.

The entry DLIN into sleep mode not set unit into low-power mode. It only inform application about some event. The behaviore in this event is application.

5.7. WAKE - UP

The wake-up signal can be send by master or slave node to tesrminates the Sleep mode of the LIN bus. To send wake-up signal in master mode value "1x110101" must be written to ***LCR*** register. After that, DLIN start transmiting wake-up signal and set ***RDY*** bit in ***LSR*** register when it done, the interrupt request is generated.

If the DLIN is in the Sleep mode it monitors the LIN bus to detects wake-up signal. Then it occurs the ***WAKEUP*** bit in ***LBS*** register is set, ***SLEEPM*** bit is clear and interrupt request is generated.

In the case when DLIN is in slave mode and sending wake-up signal the "0x110101" must be written to ***LCR*** register, after successful transmission, the ***RDY*** bit is set in ***LCR*** register and interrupt request is generated. In the same moment the wake-up response timer starts counting, if the master node not response in time specified by this timer the ***WAKEER*** bit is set in ***LER*** register and interrupt is generated (when ***WAKEE*** bit in ***LIE*** register is set).

In the specification response time is defined between 150ms - 250ms. To configure this timeout value the WUR register must be set according to the formula below:

$$WURT = \frac{tov}{(bt \cdot 32)}$$

where:

tov - is the timer timeout

bt - is the bit time set in by baudrate divisor

5.8. AUTOMATIC BIT RATE DETECTION

The DLIN core is equipped with a module for detecting the master bit rate. To switch the DLIN to automatic Bit Rate Detection mode, set "0x110001" to ***LCR*** register, DLIN must work as slave. After that, the DLIN begins to monitor the LIN bus to detect the BREAK filed and SYNCH field. Based on the time measurement, between first and last falling edge on SYNC field, a new bit rate value is detected. This new value of baudrate generator is loaded to registers ***DLL*** and ***DLH***. Then DLIN is waiting for the remaining part of LIN header (it's waiting on the PID field). After successful reception the LIN header, the interrupt is generated and the updated value of ***DLL*** and ***DLH*** registers can be read by software. When DLIN enters to Automatic Bit Rate Detection mode, it remains in it until correct LIN header has been detected and writing any new command (except ABORT command) to LCR registers, causes an overrun error.

Typical operation

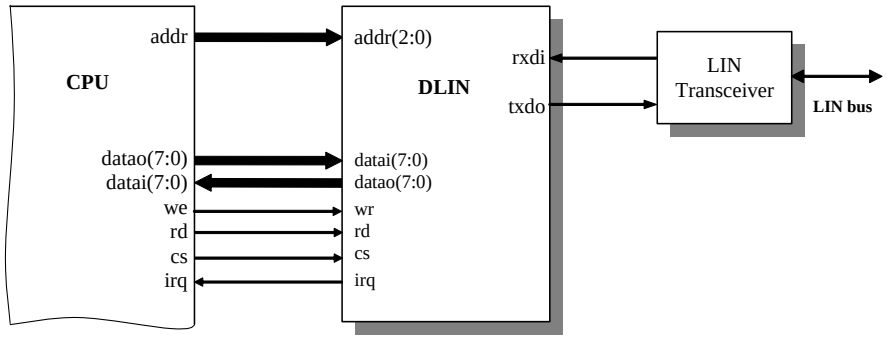


Figure 31. Typical application

The DLIN has simple and easy to use CPU/BUS interface. Reset (*rst*) active level can be modified by generic constant ***RST_ACTIVE*** in the DLIN package. Chip select (*cs*) is a slave select signal and must be asserted during read and write transfers, its active level can be modified by generic constant: ***CS_ACTIVE***. An address is selected with *addr* lines. *Datai* and *datao* are input and output data buses. *Rd* and *wr* control bus access. These are activated by signals defined in constant: ***WR_RD_ACTIVE*** and must be valid for at least one clock cycle during transfers. The DLIN is capable of generating interrupts to processor. It is accomplished by signal *int* (active high).

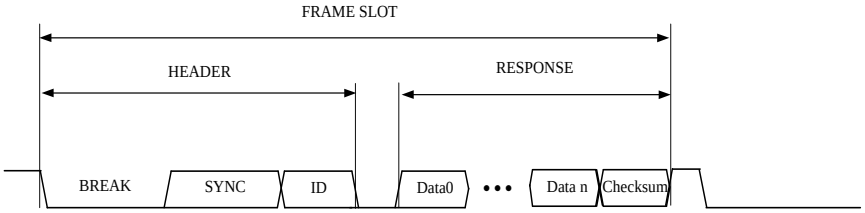


Figure 32. Receiver timing

The header consists of BREAK, SYNC and ID fields. BREAK field consists of 10 to 26 bit times of dominant value. SYNC field is the 0x55 value (synchronization process is described in section 4.5.4). The ID field includes frame identification number that was written into LID register during header detection. Response consists of data and checksum fields. CHECKSUM field include enhanced (2.1 LIN version) or classic (1.3 LIN version) checksum.

- Enhanced checksum

$$Checksum = 255 - [(\sum_{0}^{n} data\,n) + ID] + [((\sum_{0}^{n} data\,n) + ID) \gg 8]$$

- Classic checksum

$$Checksum = 255 - [(\sum_{0}^{n} data\,n) + ((\sum_{0}^{n} data\,n) \gg 8)]$$

5.9. SLAVE MODE OPERATION

This section presents data transfer of DLIN device in Slave mode.

5.9.1. SLAVE RECEIVE THE HEADER

Slave cannot initialize the transmission; it must wait for the Header sent by Master.

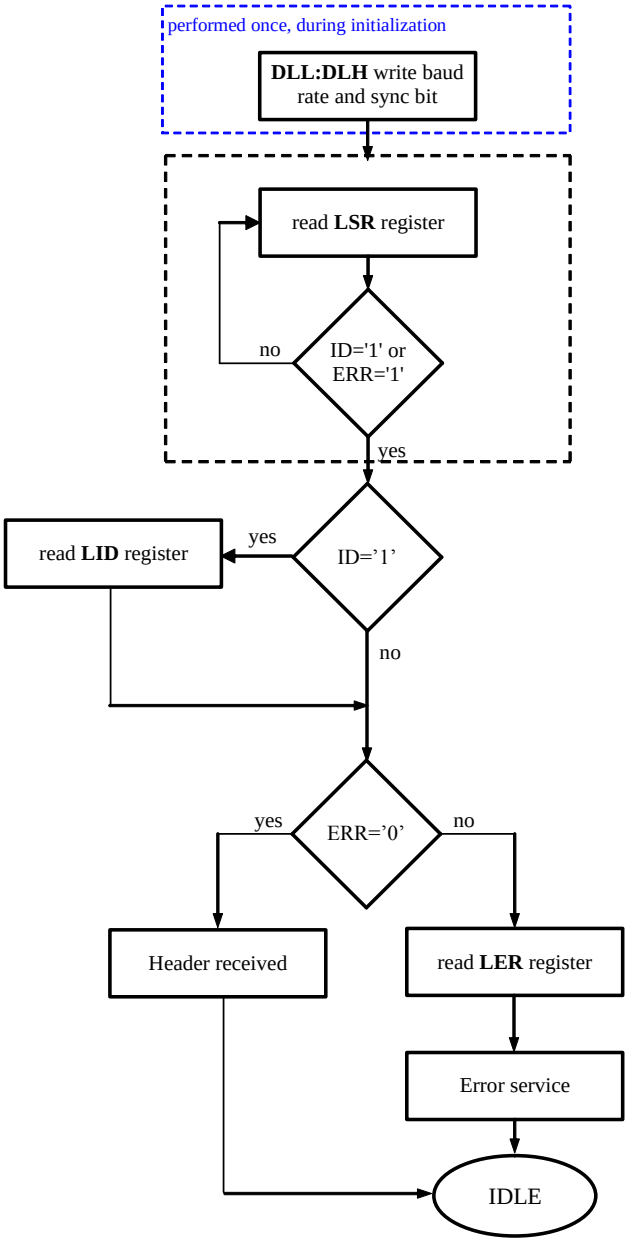


Figure 33.Slave header received flowchart

After configuration the baud rate value by DLL and DLH register writing, read the LSR register and check the ID bit value. When ID is equal '1' the Header is received and when ERR bit is set, the LER register should be read to establish the cause of error.

5.9.2. SLAVE RX RESPONSE

After reading the Header frame Slave device may be configured to receiving the data bytes.

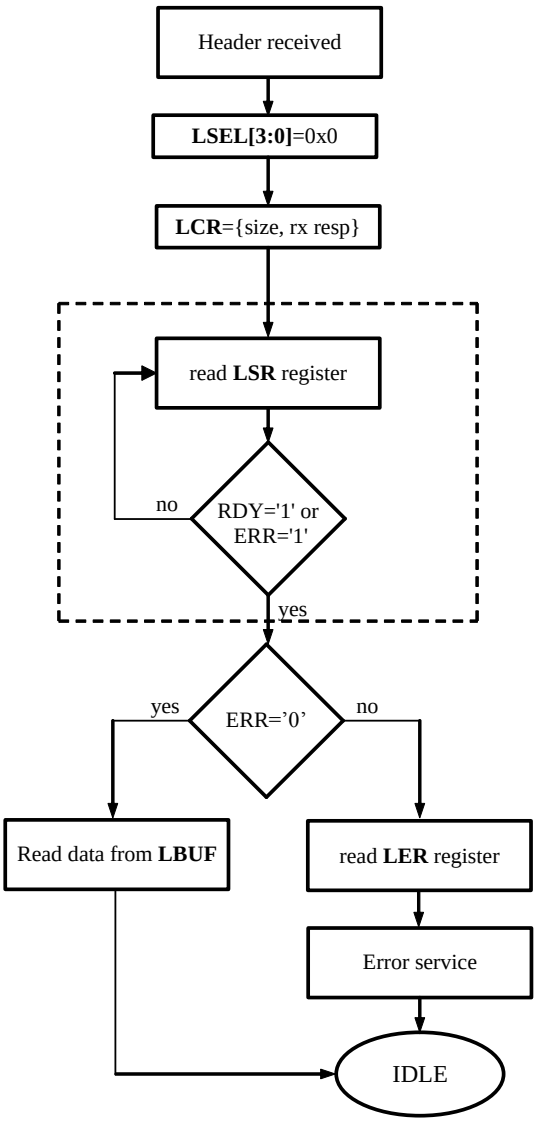


Figure 34.Slave RX response flowchart

5.9.3. SLAVE TX RESPONSE

After reading the Header frame Slave device may be configured to transmitting the data bytes.

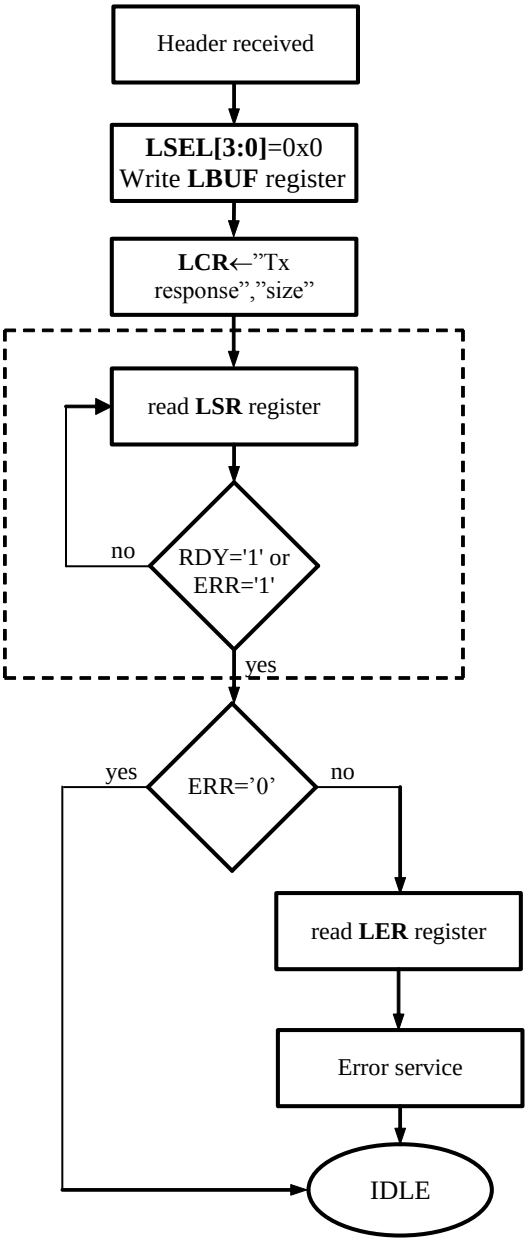


Figure 35.Slave TX response flowchart

5.9.4. LOG MODE

After reading the Header frame Slave device may be configured to receiving data frame, without knowing frame size. Note, that after log mode comand has been finished, the LER register should be read to prepare DLIN to receive next command, regardless of whether the ERR bit in LSR register was set or not.

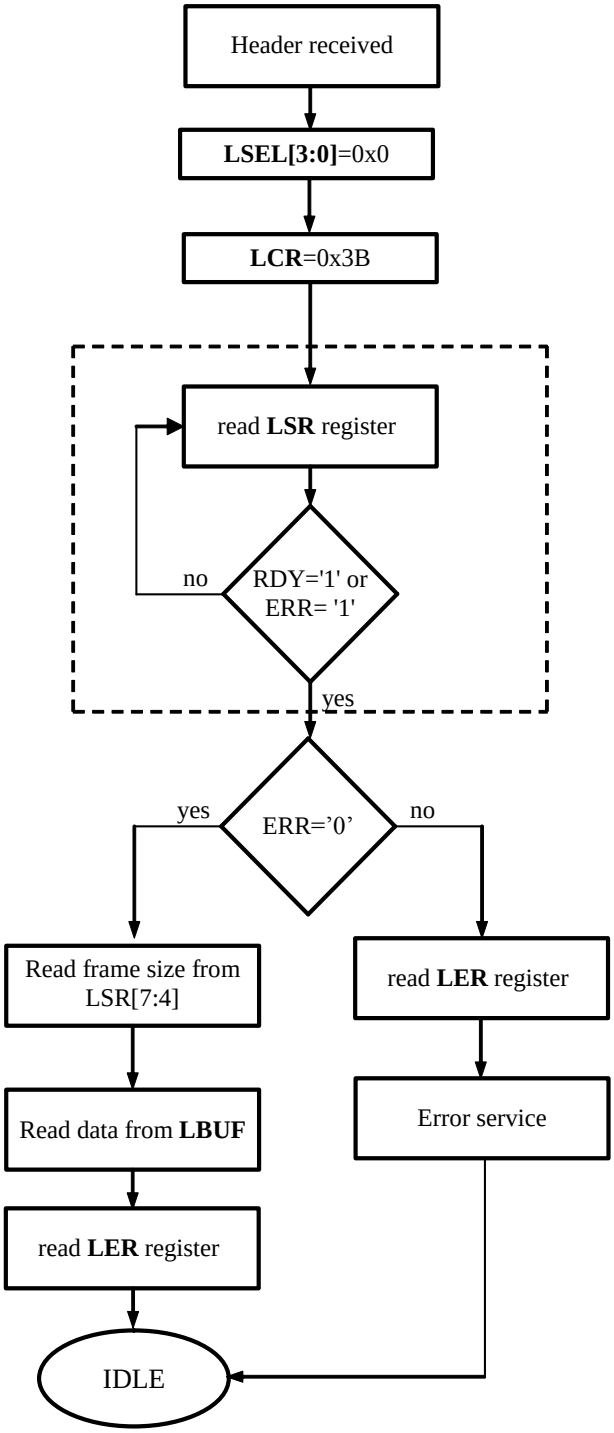


Figure 36.Slave Log mode flowchart

5.9.5. SLAVE MULTI-FRAME OPERATION

Typically, behavior of the slave determines the valid value of ID frame:

- Ignore the subsequent data transmission
- Switch to receive mode and listen to data transmitted from another node
- Switch to send mode and send data in response to the header

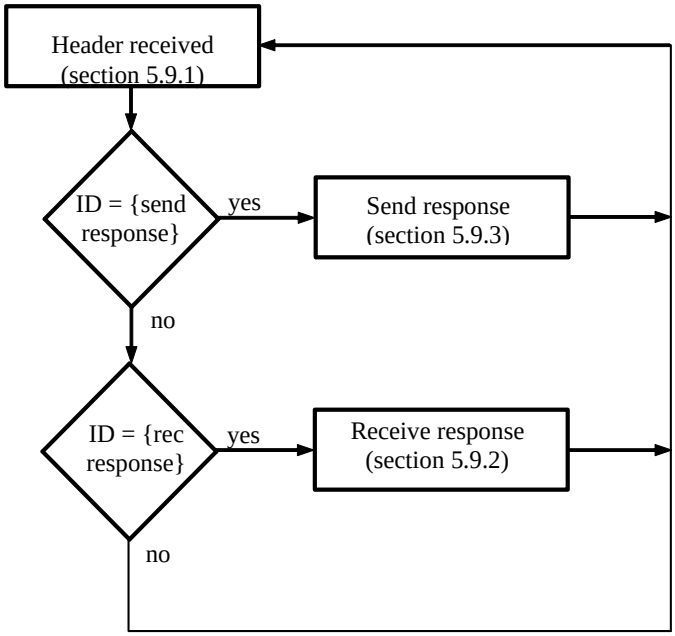


Figure 37.Slave multi frame operation flowchart

5.10. MASTER MODE OPERATION

5.10.1. SEND HEADER

To initialize the transmission Master device must send the Header frame.

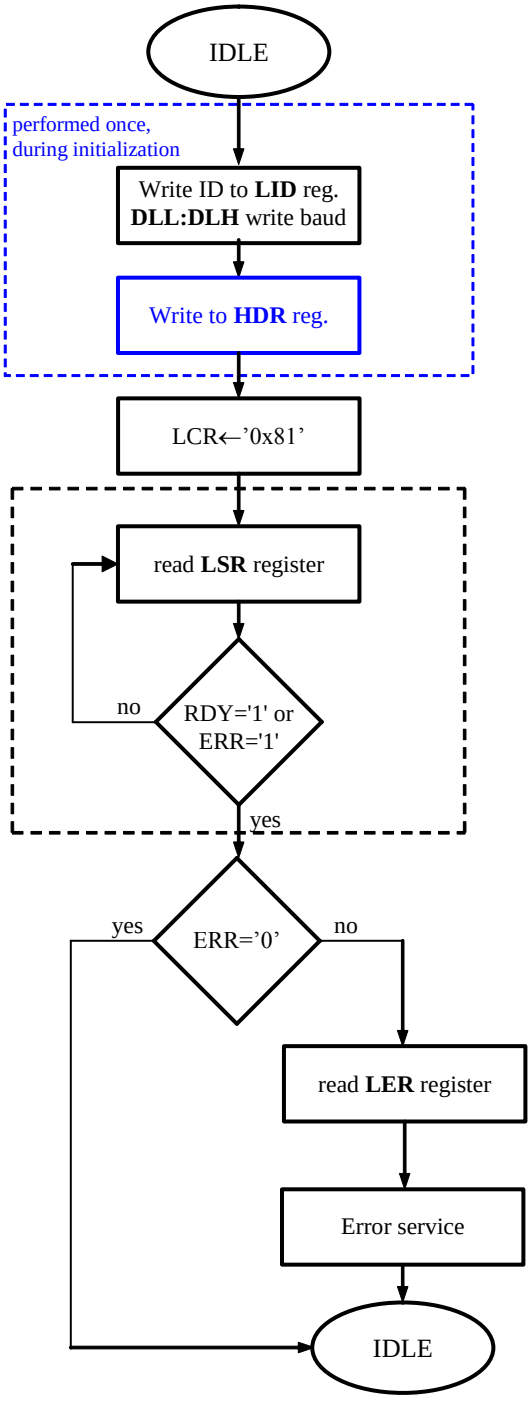


Figure 38.Master header send flowchart

5.10.2. SEND RESPONSE

When the master wants to send some data to slave, first it should send the header and then send data, as shown at the flowchart below.

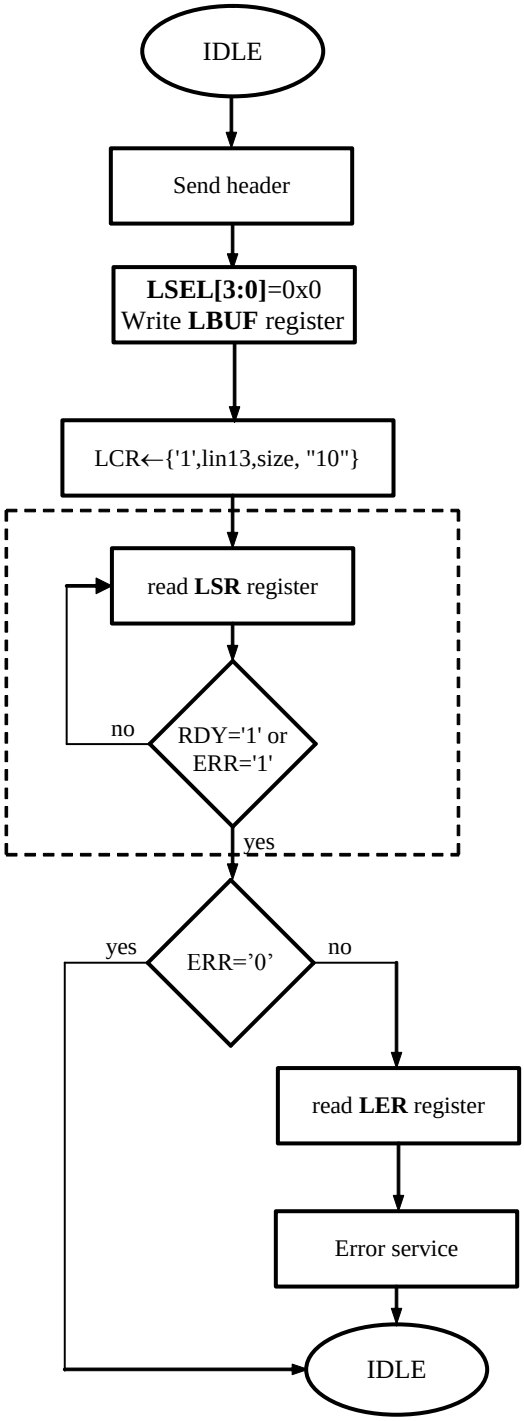


Figure 39.Master send response flowchart

5.10.3. RECEIVE RESPONSE

When the master wants to receive data from slave, first it should send the header, then switch to receiver and wait for data from slave.

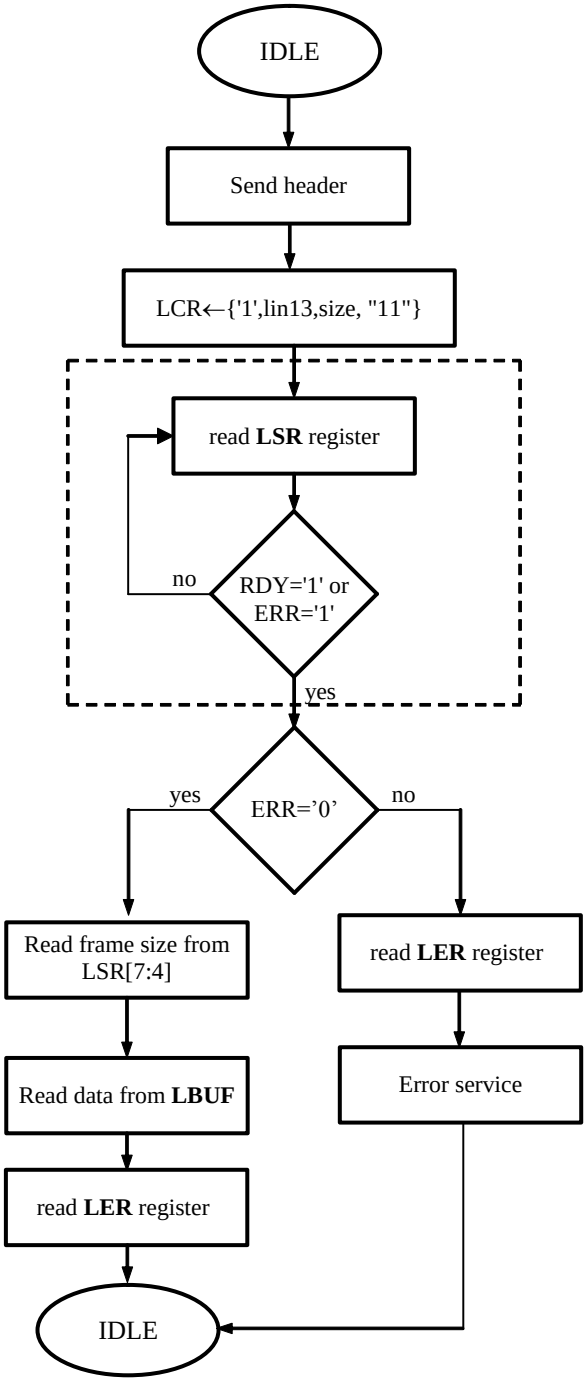


Figure 40.Master receive response flowchart

6. REVISION HISTORY

- 1.00 - The first release
- 1.01 - Parity bits has been interchanged
- 1.02 - Checksum calculate has been fixed
- 1.06 - Variable frame length in LIN1.3 mode has been added
- 1.07 - Parity bits have been added to checksum calculation in LIN 2.1
- 1.08 - Switching between LIN2.1 and LIN1.3 mode has been changed.
  - VFL bit has been removed. Frame size depends LCR[5:2] register
  - LIN1.3 bit has been changed to CLASSCHK and it switching between classic and enhanced checksum calculation
- 1.09 – The LCR registry address has been corrected
- 1.10
  - LOG mode has been added
  - BUSY bit has been changed to RDY and functionality has been inverted
  - Data write timing has been fixed
- 1.14
  - The order of TX response command in the Figure 19 has been changed
- 1.15
  - Header delay register and counter were added
- 1.17 - Sleep mode and wake-up detection has been added
- 1.18 - scan_enable and moden pins have been added on DLIN symbol
- 1.19 - APB interface has been added
- 1.20 Automatic Bit rate detection has been added
- 1.21 Block diagrams for Slave RX response and LOG mode have been corrected, bit LID clearing method has been changed
- 1.22 Flowcharts have been added

7. ORDERING INFORMATION AND SUPPORT

Headquarters:
Wroclawska 94, 41-902 Bytom, POLAND
e-mail: info@dcd.pl
tel.: +48 32 282 82 66
fax: +48 32 282 74 37

Distributors:
Please check http://dcd.pl/sales