MICRONAS

Application Note

HAL® 1820, HAL 24xy, HAL 28xy, HAL 36xy, HAL 38xy

Application Board HAL-APB V1.x

3D HAL® by Micronas   vario HAL® by Micronas

Edition Oct. 8, 2012
APN000055_003EN

License Note

HAL36xy and HAL38xy use licenses of Fraunhofer Institute for Integrated Circuits IIS.

Contents

Release Note: **Revision bars indicate significant changes to the previous edition.**

1. Introduction

1.1. General Information

The hardware and software description in this document is valid for the Application Board HAL-APB V1.x.

Fig. 1–1: Application Board HAL-APB V1.x

1.2. Introduction

The Application Board HAL-APB V1.x (HAL-APB) is an board for programming the Micronas Hall-effect sensor families with analog and digital output formats. The board is equipped with a Micronas Flash micro controller CDC 3207G. It provides an application software supporting a command interface for the communication with a PC. This allows the implementation of specific PC software for engineering purposes or in-line calibration. The HAL-APB can be ordered with a housing or as a PCB version.

In the case of a housing, an additional extension board with two sockets for the connection of up to two Hall sensors (depending on the sensor type) is supplied.

Two versions of the Application Board HAL-APB are in use: version 1.3 and the updated version 1.5. Both versions are free to be used in laboratories for engineering purposes.

Note: For usage in the production, board version 1.5 is mandatory.

1.2.1. Supported HAL Sensors

The HAL-APB supports the sensors listed in Table 1–1.

Table 1–1: Supported sensors

Sensor	Remark
HAL 1820	Linear sensor with analog output
HAL 242x	Linear sensor with analog output
HAL 2810	Linear sensor with LIN 2.0 Interface
HAL 283x	Linear sensor with SENT Interface
HAL 2850	Linear sensor with fast PWM output
HAL 3625	Direct angle sensor with analog output
HAL 3675	Direct angle sensor with PWM output
HAL 385x	2D position sensor with analog output
HAL 387x	2D position sensor with PWM output

Please refer to the corresponding Programming Guides Application Notes for detailed information on the sensors listed or contact the Application Support Sensors (support_sensor@micronas.com).

1.2.2. Sensor-specific PC Software

Micronas GmbH provides easy-to-use PC software (LabView) for each supported sensor.

1.3. Board Block Diagram

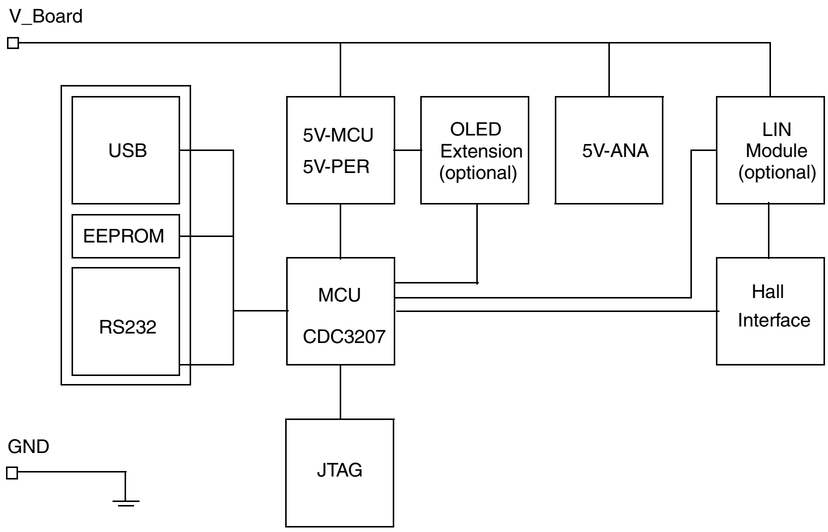


Fig. 1–2: HAL-APB block diagram

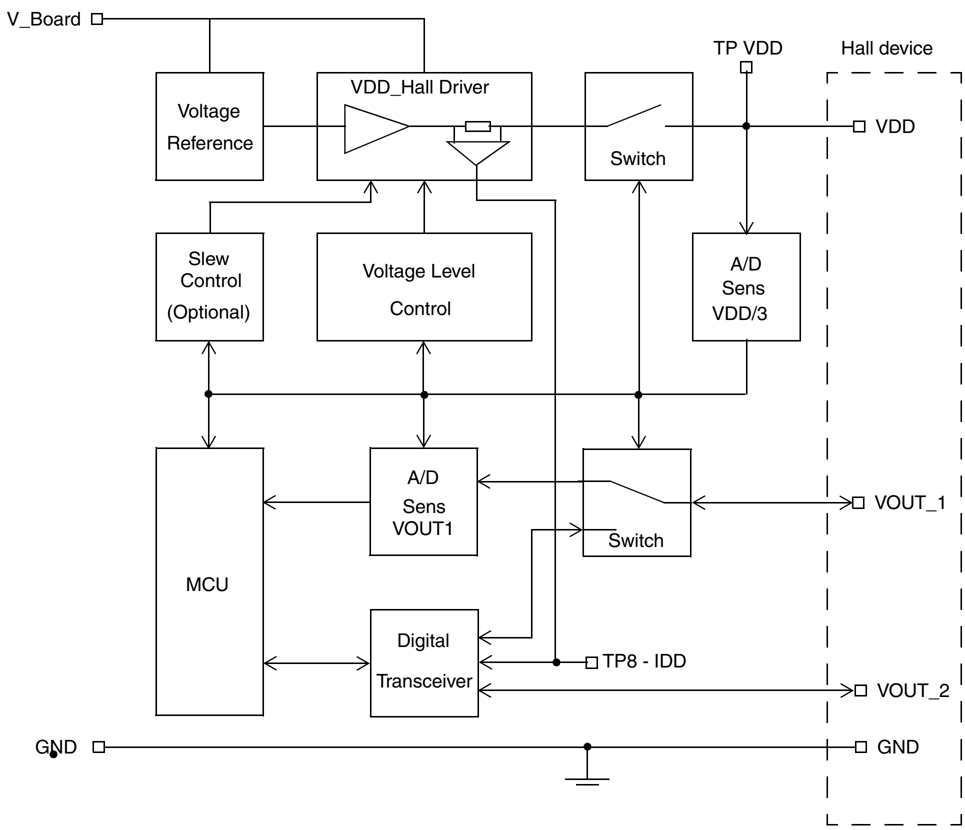


Fig. 1–3: Detailed view of HAL Interface

2. Getting started

2.1. First Steps

2.1.1. Check HAL-APB V1.x

– Connect the HAL-APB to the supply voltage.
– Check if the power-on self-test was passed successfully. (ERROR LED is switched off after power on. Exception: LIN mode)

Board Supply

The HAL-APB requires a stabilized power supply. For this purpose, either when using it without housing, the connector X2 (DC jack) or the terminal beneath it can be used.

Power-On Self-Test

The HAL-APB firmware provides a power-on self-test. The self-test is started after connecting the board supply. During the self-test, the status LEDs including the Error LED, will flash.

In case of a detected error, the ERROR LED remains illuminated after the self-test. In LIN mode the ERROR LED is switched on as long as the Vsupply of the sensor is not set to 12 V.

2.1.2. Check Communication with PC and Hall Sensor Connection

Connect a Hall sensor with the HAL-APB.

(a) directly into the socket HAL 1

or

(b) into one of the sockets of the HAL-APB extension board (housing version).

Note: For the first communication check, we recommend using the sensor specific Programming Environment LabView software provided by Micronas for the specific HAL sensor.

you can also

– set up a Hyperterminal connection (see Section 6 on page 18)
– switch Vdd on using the “vho1” command (see Section 7.2 on page 21).
– try to read out a register (see chapter of the used sensor type).

3. Board Configuration

3.1. Jumper Settings

For changing between LIN-Bus and Biphase-M communication, jumpers need to be set differently. For non-housed (optional) application boards it may be necessary to switch jumper for USB/RS-232 connection. The following pictures show how to set the jumpers correctly.

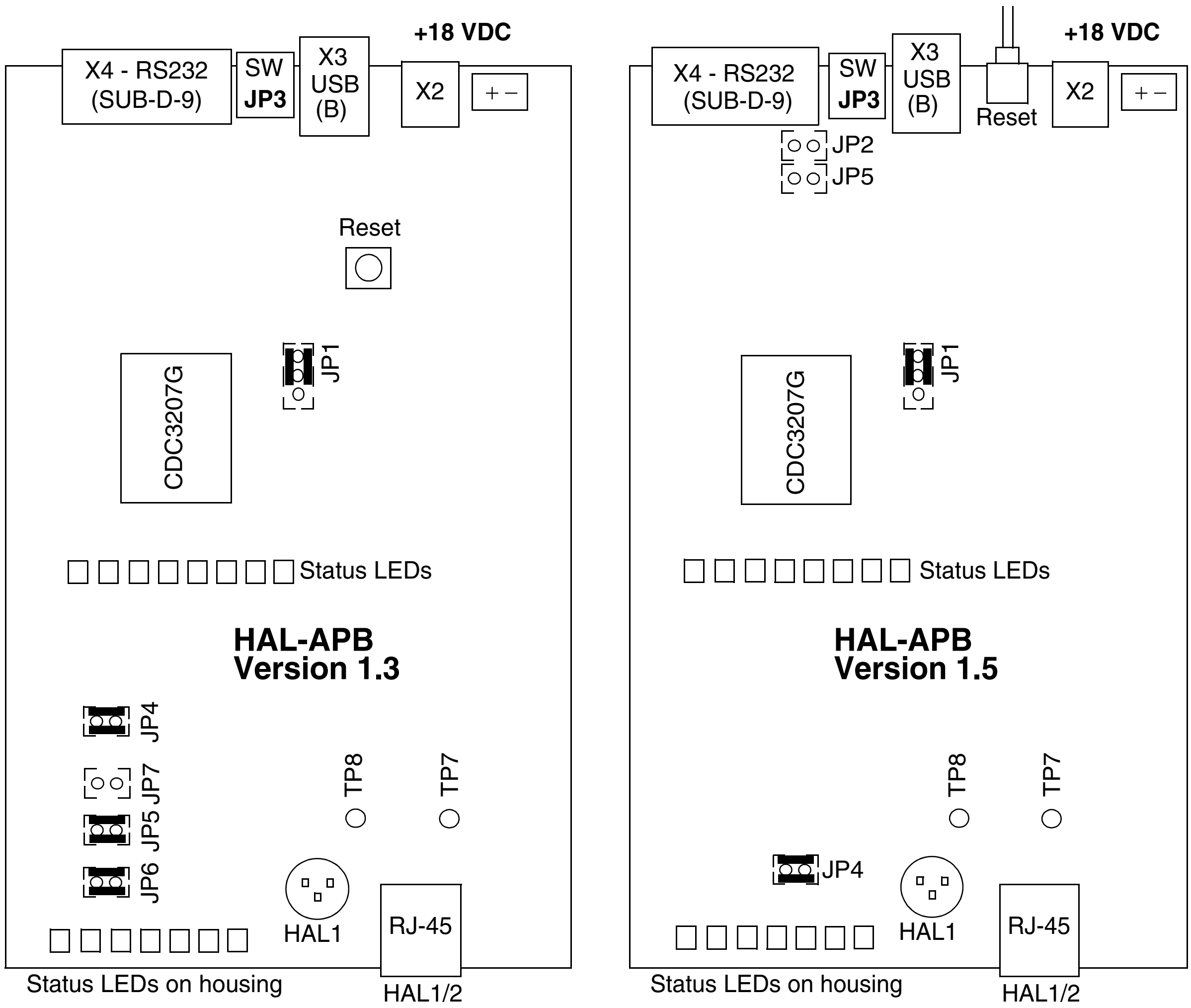


Fig. 3–1: Jumper settings HAL-APB V1.x

The default jumper position Pos1 is indicated by black bars in Fig. 3–1.

Note: For board versions higher than v1.3, no manual adjustment is required. The HAL-APB firmware automatically detects the appropriate protocol to be used for the Hall sensor.

Table 3–2: Jumper settings

Jumper	Setting	Function
JP1	pos1 (default) pos2	debug normal operation
JP2	**V1.5** open (default) close	communication via **RS232** manual MCU reset during firmware flash automatic MCU reset during firmware flash
	V1.3 pos1 (default) pos2	 normal operation reserved
JP3	close (default) open	USB RS-232
JP4	close (default) open	VDD_Hall equals GND when Vsup is switched off VDD_Hall is floating when Vsup is switched off
JP5	**V1.5** open (default) close	communication via **USB** manual MCU reset during firmware flash automatic MCU reset during firmware flash
	V1.3 close (default) open	 normal operation only for LIN Bus applications
JP6 (only V1.3)	close (default) open	normal operation only for LIN Bus applications
JP7 (only V1.3)	open (default) close	normal operation only for LIN bus applications

Note: JP7 must not be set in combination with JP4 and JP5 (only V1.3)

Table 3–3: Board LED description

LED Name	Function
ERROR	On, in case of communication error
READY	On, after power-on of board
PCCOM	On, in case of communication between PC and HAL-APB
HAL_RD	Telegram on VOUT
HAL_PR	reserved
HAL_WR	Telegram high level on Hall VDD
HAL_ON	Hall VDD on

3.2. HAL Interface Connector

Depending on the sensor type, up to two sensors can be connected to the board. For this purpose, a 6-pin connector HAL1/2 is provided. Alternatively, one Hall Sensor can be inserted in the 3-pin socket HAL beneath the connector HAL1/2 (only available for boards without housing).

The following pin's are connected in parallel Pin No. 1(V_{SUP}Sensor1) and 4 (V_{SUP}Sensor2) and Pin No. 2 (Common Sensor GND) and Pin No. 5 (Common Sensor GND). The male plug (modular RJ-12, OST (MMJ) coding) corresponding to the fawn connector HAL1/2 can be ordered from every electronics store. The pinning of the interface is described in Table 3–4.

Table 3–4: Pinning of the HAL interface HAL1/2

Pin No.	Description
1	Sensor input V_{SUP} Sensor 1
2	Common Sensor GND
3	Sensor output $V_{OUT/DIO}$ Sensor 1
4	Sensor input V_{SUP} Sensor 2
5	Common Sensor GND
6	Sensor output $V_{OUT/DIO}$ Sensor 2

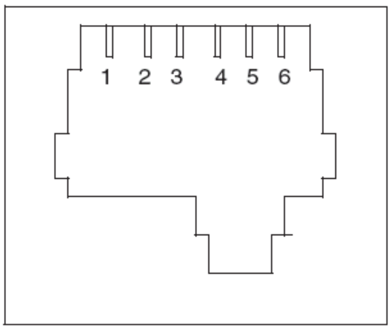


Fig. 3–2: Modular connector HAL1/2, front view

3.3. Firmware update

The procedure for a firmware update of the HAL-APB V1.x is provided in the Application Note "Firmware Update HAL-APB V1.x"

4. Specification

4.1. Recommended Operating Conditions

All voltages are referenced to GND (-VB pin at X1 = GND at X2)

Table 4–5: Board conditions

Symbol	Parameter	Connector	Limit Values			Unit	Test Conditions
			Min.	Typ.	Max.		
I_{SUP}	Supply Current	X2	-	180	-	mA	
V_{SUP}	Supply Voltage	X2	16	18	20	V	
C_L	Load Capacitance	HAL1/2	-	-	100	nF	

4.2. Recommended Wiring

We recommend connecting the application to the board using shielded wires.

In order to minimize the risk of electromagnetic disturbances, the cable should be as short as possible.

Note: Especially in noisy environments beneath power switches, electromagnetic actuators, and the like, EMI-compliant layout of the wiring is mandatory.

For recommended cable parameters, please refer to Table 4–6.

4.3. Maintenance and Calibration

We recommend sending the programmer board back to the supplier for maintenance and calibration of the voltage levels after one year of operation.

The Hall programmer board must not be maintained or repaired by the customer. In case of any problems or defects, please contact your supplier.

WARNING: Do not modify any part of the Hall programmer board V 1.x, nor readjust any trimming potentiometer. Otherwise, the board may be damaged, the sensor programming may be insufficient, and the reliability of the sensor reduced.

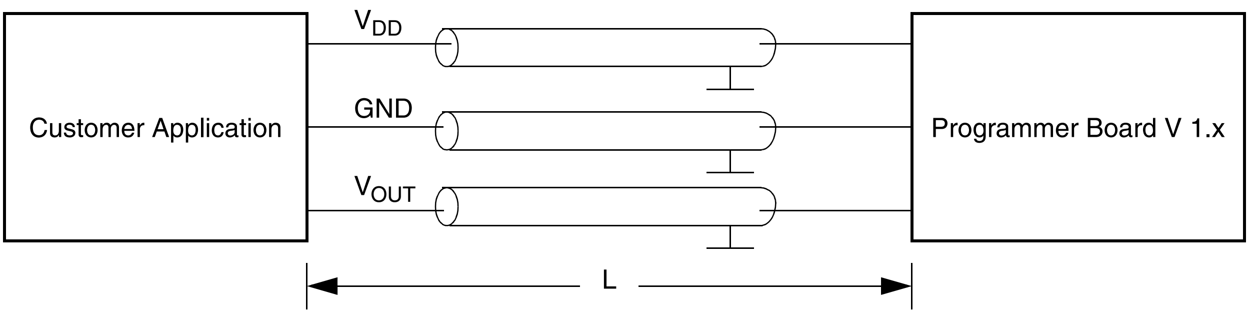


Fig. 4–3: Recommended wiring - schematic sketch

Table 4–6: Recommended cable parameters

Symbol	Parameter	Min.	Typ.	Max.	Unit	Conditions
R_0	Ohmic Resistance per Wire	–	1	5	Ω	$I \leq 10$ mA
C_0	Capacitance	–	80	120	pF	
Z	Impedance	–	50	–	Ω	
L	Length	–	–	1	m	

4.4. Characteristics

All voltages are referenced to GND (-VB pin at X1 = GND at X2)

Table 4–7: Board characteristics

Symbol	Parameter	Connector	Limit Values			Unit	Test Conditions
			Min.	Typ.	Max.		
I_{SUP_HAL}	Output Load Current	HAL1/2	-	-	40	mA	Supply current per device
V_{OUT_HAL}	Output Voltage of Hall Device	HAL1/2	0 0	-	5 18	V V	Standard configuration (default) LIN configuration only!
$V_{SUP\_HAL\_NORM}$	NORM Level of HAL Supply Voltage	HAL 1/2	4.9	5	5.1	V	
$V_{SUP\_HAL\_LOW}$	LOW Level of HAL Supply Voltage	HAL 1/2	5.8	6.0	6.6	V	
$V_{SUP\_HAL\_HIGH}$	HIGH Level of HAL Supply Voltage	HAL 1/2	6.8	7.3	7.8	V	

Note: The voltage levels are trimmed by the manufacturer. If any of the levels listed are found to be outside the specification limits, please contact the manufacturer or the Application Support Sensors Team.

The RS232 cable should be a standard serial cable. Also called straight cable.

5. USB Driver Installation

Note: When using the serial cable you do not need to install this drivers. They are only necessary for connecting the Application Board HAL-APB V1.x via USB cable to the PC.

5.1. Installing the USB VCP Drivers

Plug in the Application Board HAL-APB V1.x (Power supply also connected) into a spare USB port and plug in the power supply.

Windows 7 will automatically search latest driver if the PC is connected to the internet. If there are problems with the installation follow the application note: “AN_119_FTDI_Drivers_Installation_Guide_for_Windows7.pdf”

The application note: “AN_104_FTDI_Drivers_Installation_Guide_for_WindowsXP.pdf” can be used to install the driver on a windows xp system.

The application note can be either found on the Micronas Service Portal (https://service.micronas.com/workgroups/) or on the FTDI homepage.

Note: Sometimes the installer repeats the whole procedure. If this happens please do the same as explained above again.

6. Board Functions

6.1. Serial Command Interpreter

This board provides a serial command interpreter for the interaction with a PC, connected via USB or RS232.

The serial communication protocol applies a software handshake:

– The PC acts as a master, the HAL-APB V1.x as slave,

– The HAL-APB V1.x responds to each master **COMMAND** frame with a **RESPONSE** frame.

6.1.1. Serial Interface Configuration

When using a hyperterminal communication please set the following parameters.

Table 6–8: parameter settings of serial interface

Parameter	**Value**
Bits per second	38400
Data bits	8
Parity	Even
Stop bits	1
Flow control	none

6.1.2. Definition of the COMMAND Frame

The command frame is of variable length. There are basically two types of commands:

1. for board configuration
2. for communication with connected Hall device

The command string has to end with <CR> (ASCII character 0x0D), optionally with <CR><LF> (ASCII characters 0x0D, 0x0A).

6.1.3. Definition of the RESPONSE Frame

The **response** frame consists of 7...10 characters plus 1 finishing <LF>

<ST>:<R9><R8>....<R2><R1><R0> <LF>

ST is non-zero in case of errors (see Table 6–9)

The Rx-characters contain the received data depending on the command (see device-dependent command lists in section 9, 10,...).

6.1.4. Analog Measurements

Its also possible to measure analog voltages, as the HAL_VDD or the HAL_VOUT with the ADC of the HAL-APB. The HAL_OUT is only correctly measurable when HAL_VDD equals 5 V.

Example

ftvdl0 (set VDD to 5 V)
ftana1 (measure HAL_VDD)
ftana2 (measure HAL_VOUT)

V_{DD} = DATA / 1024 x 3 x 5V

V_{OUT} = DATA / 1024 x 5V

DATA is measured by ftana command as explained in Table 7–11.

6.1.5. Error Codes

Table 6–9: Error codes

STATUS	Error
0	no error
1	acknowledge error
2	2'nd Acknowledge error
3	invalid command for selected Mode
4	PID in running table cannot be modified (LIN)
5	LIN communication Error
6	LIN interface connection Error
7	no PWM (at PWM Duty Cycle read command)
8	reserved
9	reserved
10 (0xA)	reserved
11 (0xB)	reserved
12 (0xC)	reserved
13 (0xD)	data read error
14 (0xE)	invalid command parameter
15 (0xF)	invalid command

7. Board Mode Settings

7.1. Board Operation Modes

In order to meet the different requirements of the various Hall devices, the board can be run in different operation modes. When a particular device is used, the corresponding board mode has to be selected first. The mode list can be displayed by sending the board command “?m”.

Table 7–10: Board modes

Mode	Description
8	HAL 2810 – LIN Mode
9	HAL 283x/50 – Biphase via DIO- Pin
A	HAL 1820 – Biphase via V_{SUP} Pin HAL 24xy – Biphase via V_{SUP} Pin HAL 3625 – Biphase via V_{SUP} Pin HAL 3675 – Biphase via V_{SUP} Pin HAL 38xy – Biphase via V_{SUP} Pin
C	HAL 24xy – Biphase via OUT - Pin HAL 3625 – Biphase via OUT - Pin HAL 3675 – Biphase via OUT - Pin HAL 38xy – Biphase via OUT - Pin

7.2. Board Configuration Commands

The board configuration commands shall be used to

– select the board mode
– set/read configuration data like the bit time or firmware version
– control the power supply V_{DD_HAL} of the connected sensor

Table 7–11: Board configuration

Action	Command	Parameter	Remarks
get firmware version	?v	return <ST>:*[Version]*	firmware release version Example => ?v <= 0:v2.32
show available board modest	?m	-	returns a list of board modes available

Table 7–11: Board configuration

Action	Command	Parameter	Remarks
set board mode	sm**N**	**N** see Table 7–10 on page 21 for details return value: <ST>:0000**N**	switch board to device specific mode Example => smA <= 0:0000A
Switch V_{SUP_HAL} on	vho1	return value: <ST>:00001	supply voltage on (default 6 V; see voltage levels for details) Example => vho1 <= 0:00001
Switch V_{SUP_HAL} off	vho0	return value: <ST>:00000	supply voltage off => vho0 <= 0:00000
For factory tests only			
set low voltage level	ftvdl**X**	**X** = 0: V_{SUP}= 5V **X** = 1: V_{SUP} = 6V	normal V_{SUP} low level for Biphase-M protocol, if programming via V_{SUP}-Pin
set high voltage level	ftvdh**X**	X = 0: V_{SUP} = 5V X = 1: V_{SUP} = 7.5V (if ftvdl0) X = 1: V_{SUP} = 8V (if ftvdl1)	high level for Biphase-M protocol, if programming via V_{SUP}-Pin
set programming voltage	ftvdp**X**	X = 0: V_{SUP}= 5V (if ftvdl0 and ftvdh0) X = 1: V_{SUP} = 12V (if ftvdl0 and ftvdh0) X = 1: V_{SUP} = 12.5V (if ftvdl1 and ftvdh0) X = 1: V_{SUP} = 14.5V (if ftvdl0 and ftvdh1) X = 1: V_{SUP} = 15V (if ftvdl1 and ftvdh1)	for LIN-Mode
force output voltage	ftsme**X**	**X** = 1: force output voltage to high state X = 0: release output	
switch output pull-up resistor on/off	ftpon**X**	X = 0: pull-up off X = 1: pull-up on	for open-drain devices you need the pull-up resistor
select I/O channel	ftses**N**	N = 1 or 2 return value: <ST>:0000N	N = 1 HAL1 N = 2 HAL 2 only possible in combination with programming via OUT-Pin.
set Bit time	sbt**BT**	BT = 000A... 0D48 (10us...3.4ms) return value: <ST>:00000	bit time in µs as 4-digit hex No. (defaul=0x03E8)
measure V_{SUP}	ftana1	ftana1 "\n" return value: <ST>:000000	measure the supply voltage with HAL-APB's ADC Example => ftana1 <= 0:00177 0x00177 = 375(dec) * 15 / 1024 = 5.49 V

Table 7–11: Board configuration

Action	Command	Parameter	Remarks
measure VOUT_HAL	ftana2	ftana1 "\n" return value: <ST>:000000	measure the analog output voltage of sensor 1 with HAL-APB's ADC Example => ftana2 <= 0:00200 0x00200 = 512(dec) * 5 / 1024 = 2.5 V only for Sensor 1 available
read PWM-Period and Pulse width[1)]	pr**N**	**N** = 0 / 1 return value: <ST>:<P4><P3><P2><P1><P0><W4><W3><W2><W1><W0>	**N** trigger on falling/rising PWM edge <ST> board status <P> 5 digit Period <W> 5 digit Pulse width *Example* => pr1 (OP-bit=0) <= <ST>:013AE00A00 - Conversion of PWM-Period: 0x013AE = 5038_{dec} / 10000 = 0,5ms - Conversion of Pulse width: 0x00A00 = 2560_{dec} / 10000 = 0,26ms - Calculation of Duty-Cycle: 2560/ 5038 = 50,8%
get Bit time	?bt	return value: <ST>:**BT**	BT as a 5-digit hex No.
get last acknowledge time	?ack	return value: <ST>:0x**ACK**	ACK as a 4-digit hex No. returns the width of last acknowledge pulse
[1)]Available with firmware versions greater then 2.32. With firmware versions less then 2.32 this command is only available in mode 9. <ST> = Board Status character see from Section 6.1.5 on page 20 onwards for details			

8. HAL 1820

The HAL 1820 is a universal magnetic field sensor with a linear output, based on the Hall effect. Magnetic field, perpendicular to the branded side of the sensor provides a output voltage direct proportional to the applied magnetic flux through the Hall plate and proportional to the supply voltage (ratiometric behavior). Details on features and specification are described in the data sheet.

8.1. Programming interface

The sensor is programmed via supply voltage modulation. After detecting a command, the sensor reads or writes the memory and answers with a digital signal on the output pin.

A logical “0“ is coded as no level change within the bit time. A logical “1“ is coded as a level change of typically 50% of the bit time. After each bit, a level change occurs (see Fig. 8–1).

The serial telegram is used to transmit the EEPROM content, error codes and digital values of the magnetic field from and to the sensor.

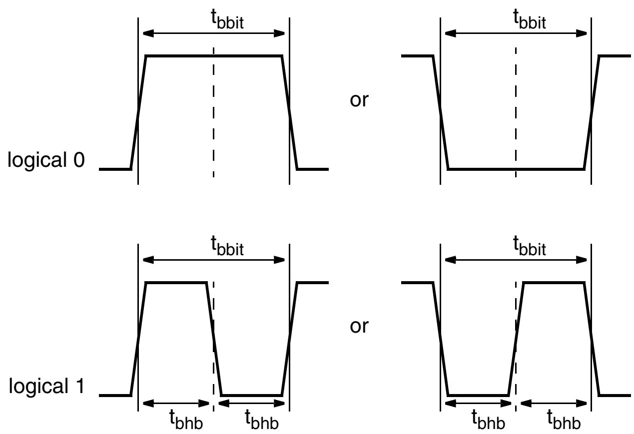


Fig. 8–1: Definition of logical 0 and 1 bit

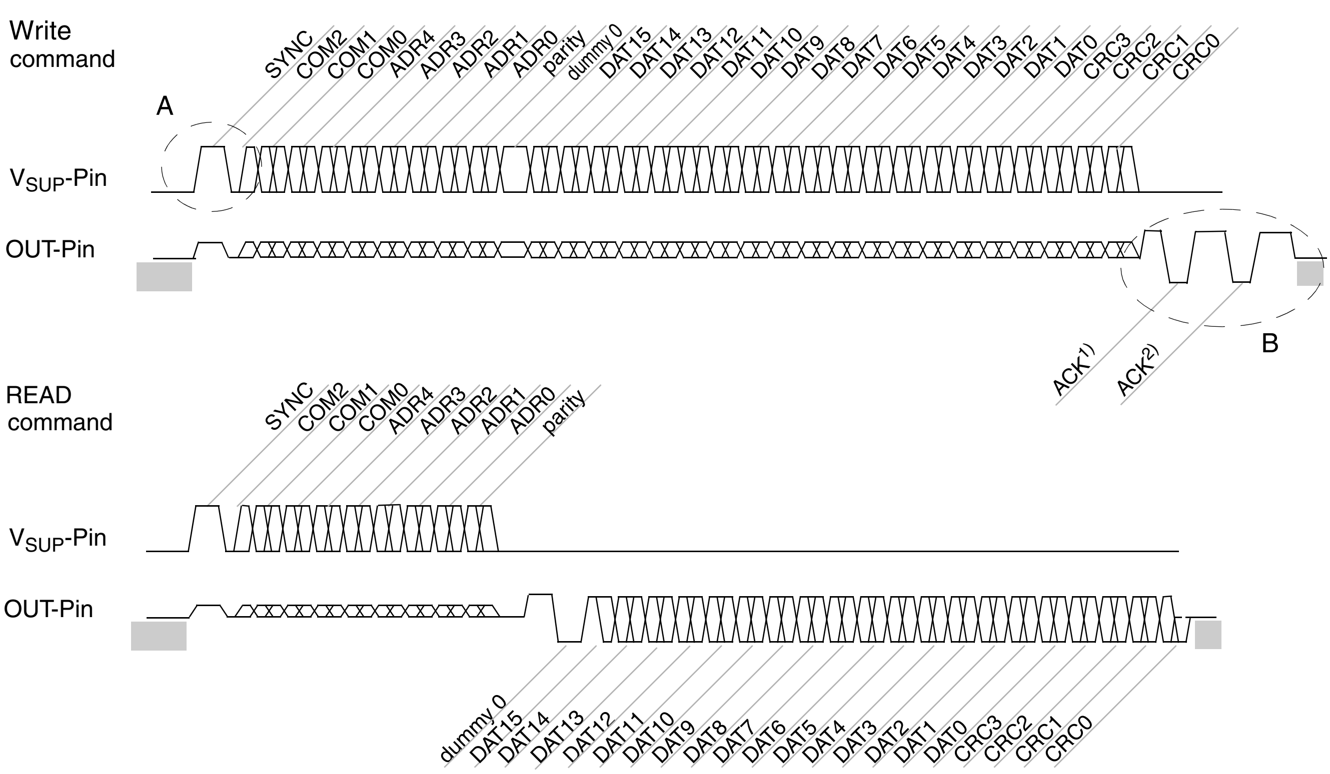
8.2. Command Structures of Protocol

COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

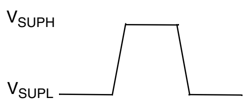
SYNC: start bit (always 0)
DAT: data bit
CRC: CRC bit

ACK: acknowledge
: Analog Output

Communication via V_{SUP}-Pin (Biphase-In = V_{SUP}-Pin / Biphase-Out = OUT-Pin)

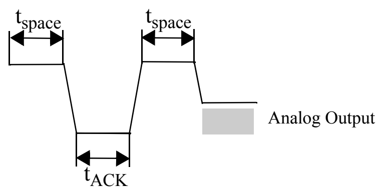


Detail A



Detail B

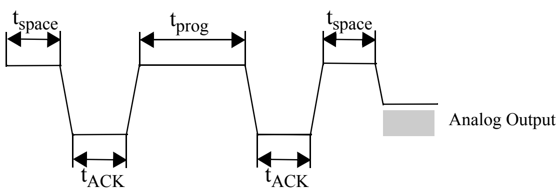
OUT-Pin



[1)]One Acknowledge if a register is written (except the NVPROG register)

Detail B

OUT-Pin



[2)]2nd Acknowledge only if the NVPROG register is written and the erasing or programming was successful

8.3. Telegram Parameters

Table 8–12: Telegram parameters for programming via V_{SUP}-Pin (all voltages are referenced to GND)

Symbol	Parameter	Pin No.	Limit Values			Unit	Comment
			Min.	Typ.	Max.		
$V_{SUPProgram}$	V_{SUP} Voltage for EEPROM & NVRAM programming (during Programming)	1	5.7	6	6.5	V	
Biphase frame characteristic of the host							
t_{h_bbit}	Host biphase bit time		972	1024	1075	µs	
t_{h_bhb}	Host biphase half bit time		0.45	0.5	0.55	t_{h_bbit}	
t_{h_bifsp}	Host biphase interframe space		3			t_{h_bbit}	
	Slew rate		-	2	-	V/µs	
Biphase frame characteristic of the Sensor							
t_{s_bbit}	Sensor biphase bit time		820	1024	1225	µs	
t_{s_bhb}	Sensor biphase half bit time		-	0.5	-	t_{s_bbit}	
t_{s_bifsp}	Sensor biphase interframe space		2	-	-	t_{s_bbit}	
t_{s_bresp}	Sensor biphase response time		1	-	5	t_{s_bbit}	
	Slew rate		-	2	-	V/µs	
Detail A							
V_{SUPL}	Supply Voltage for Low Level during Programming through Sensor V_{SUP} Pin	1	5.8	6.3	6.6	V	
V_{SUPH}	Supply Voltage for High Level during Programming through Sensor V_{SUP} Pin	1	6.8	7.3	7.8	V	
Detail B							
t_{space}	Time before and after acknowledge		-	1	-	t_{s_bbit}	
t_{ACK}	Acknowledge time		-	1	-	t_{s_bbit}	
t_{prog}	Programming time		-	1	-	ms	

8.4. Available sensor commands

The sensor supports 2 commands which provide read and write access to the NVRAM and register. Then mentioned commands allows for example to read the hall value, and to program the NVRAM.

The used write data frame and read data frame contains of 5 address bits.

In case of a unknown command, the sensor does neither transmit an acknowledge nor a body.

Table 8–13: Available Commands

Command	COM b[2:0]	frame type	ADR b[4:0]	DAT b[15:0] (RD/WD)
Read	1	read	offset address (0 to 31)	data read from address = ADR
Write	6	write	offset address (0 to 31)	data which is written to address = ADR

8.4.1. Read

The read telegram uses the read data frame. The sensor transmits the data of the address (A b[4:0]) after the header has been successful received and the address is permitted. Otherwise, the sensor does not respond.

8.4.2. Write

The write telegram uses the write data frame. The sensor saves the received address to the calculated effective address and transmits an acknowledge after the header and body has been successful received and the effective address is permitted. Otherwise, the command is discarded and the sensor transmits no acknowledge.

A write telegram is also discarded while NVRAM programming.

8.4.3. Protocol Error Handling

The sensor is detecting and logging protocol errors and command errors. The command errors are specified in section 1.

The sensor is detecting following communication errors

– invalid parity

– invalid checksum

– command error
A command error occurs when the command is either unknown or the execution has failed.

In case of an error, the sensor transmits no acknowledge nor a body. If the protocol is understood an ACK is sent no matter if the command can/may be processed

8.4.4. Data check

To allow data transmission in rough environments, two separate check mechanisms are implemented.

The command bits and the address bits are followed by a common parity bit as per description

8.4.5. CRC

The data bits are always followed by 4 CRC bits. For all commands but read the CRC result is calculated of all protocol bits, including command, address, parity and data bits.

For read command, the CRC result is calculated of dummy bit and data bits only (16).

The polynomial for the CRC calculation is always X^4+X+1.

In case of correct command detection (parity, CRC and command address if applicable), ACK is sent.

Disrupted transfers can be retried by the master.

8.4.6. Parity check

For the command and address bits, an “odd” parity check is used. In the case of an even number of “1”s, the parity bit has to be “1”. In the case of on odd number of “1”s, the parity bit has to be “0”. With the parity bit, global parity is always even.

8.5. HAL 1820 - Board commands

Note: For general board commands see Table 7–11 on page 21

Table 8–14: HAL 1820 - Board commands

Action	Command	Parameter	Data
write data to address	xxw**STR**	**STR** = <A1><A0><D3><D2><D1> <D0><CRC> return value: <ST>:<R5><R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* write C000 (hex) into register 8 => xxw08C0008 <= 0:000000[1)] <= 2:000000[2)]
read data from address	xxr**STR**	**STR** = <ADR1><ADR0> Return value: <ST>:<R3><R2><R1><R0><CRC>	<A> address as 2-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* read address 8 => xxr08 <= 0:C000B

1) response for firmware versions less then 2.32
2) response for fimware versions greater or equal then 2.32

8.6. Locking of the Sensor

For reliability in service, it is mandatory to set the LOCK bit after final adjustment and programming.

The success of the LOCK process should be checked by reading the status of the LOCK bit after locking.

It is also mandatory to check the status of the received data or to read/check the status of the DIAGNOSIS register after each store sequence to verify if the programming of the sensor was successful. VER, PER and OVP (bits <2:0>) should be 0 after each set/erase command. NVE (bit 11) should be 0 only after enough succesive set or erase command (after 10 erase or 10 set).

Electro-static discharges (ESD) may disturb the supply voltage during programming. Please take precautions against ESD.

For the programming during product development and also for production purposes, a programming tool including hardware and software is available on request. It is recommended to use the Micronas tool kit for an easy product development. It is also recommanded for production to always first program the “0” (erase command) the the “1” (set command).

Note: It is possible to read the registers of the HAL 1820 after locking. Changing of the memory after locking is not possible.

9. HAL24xy

The HAL 24xy is a universal magnetic field sensor with a linear output, based on the Hall effect. Magnetic field, perpendicular to the branded side of the sensor provides a output voltage direct proportional to the applied magnetic flux through the Hall plate and proportional to the supply voltage (ratiometric behaviour). Details on features and specification are described in the data sheet.

9.1. Programming interface

The sensor can be programmed via supply voltage modulation or via output voltage modulation. The default mode is the programming via the output voltage modulation. The sensor answers with a modulation of the output voltage.

A logical "0" is coded as no level change within the bit time. A logical "1" is coded as a level change of typically 50% of the bit time. After each bit, a level change occurs (see Fig. 9–1).

The serial telegram is used to transmit the EEPROM content, error codes and digital values of the magnetic field from and to the sensor.

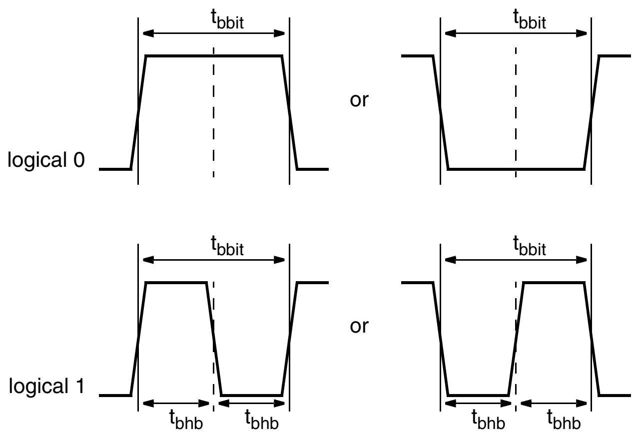


Fig. 9–1: Definition of logical 0 and 1 bit

9.2. Command Structure of Protocol for communication via V_{SUP}

command structure

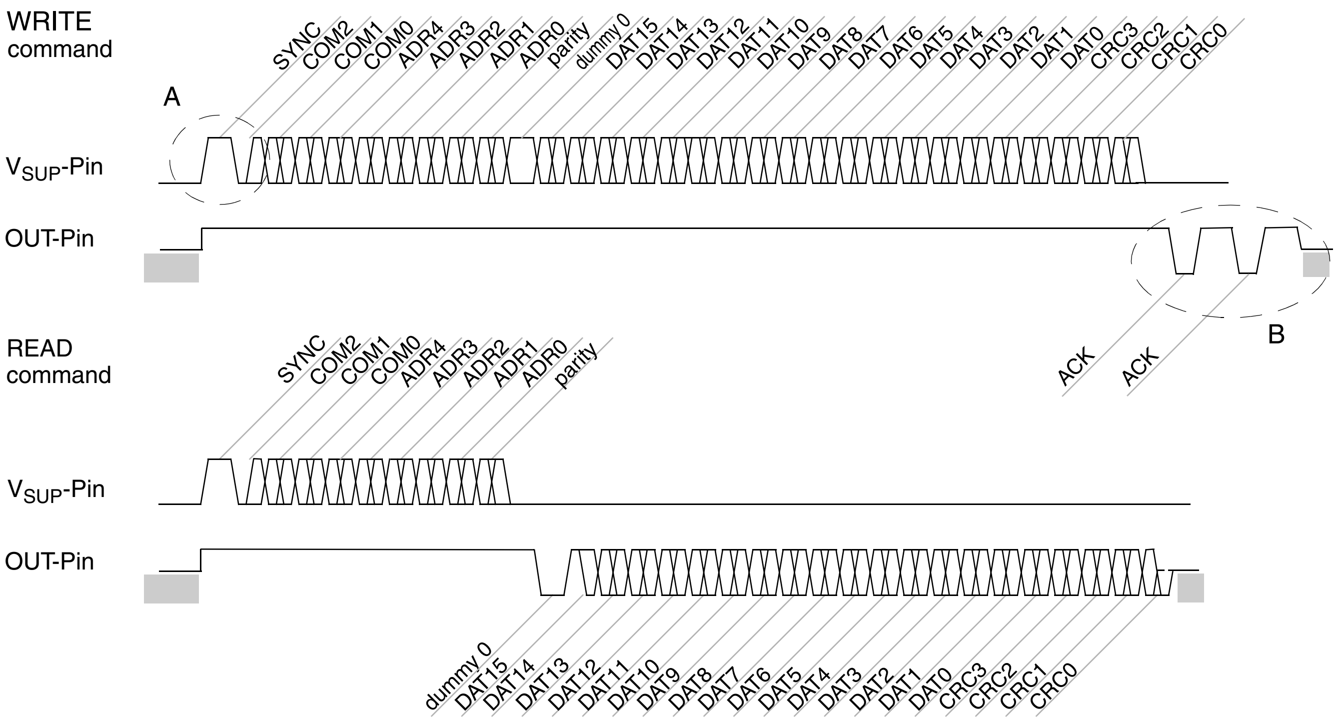
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

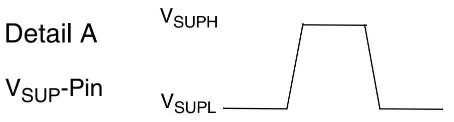
SYNC: start bit (always 0)
DAT: data bit

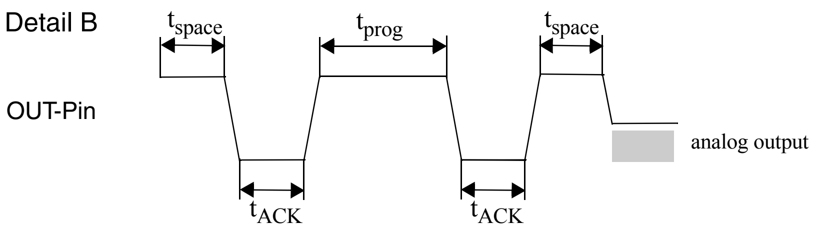
CRC: CRC bit

ACK: acknowledge
: Analog output

Communication via V_{SUP}-Pin (Biphase-In = V_{SUP} / Biphase-Out = OUT-Pin)







9.3. Command Structure of Protocol for communication via OUT-Pin

command structure

COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

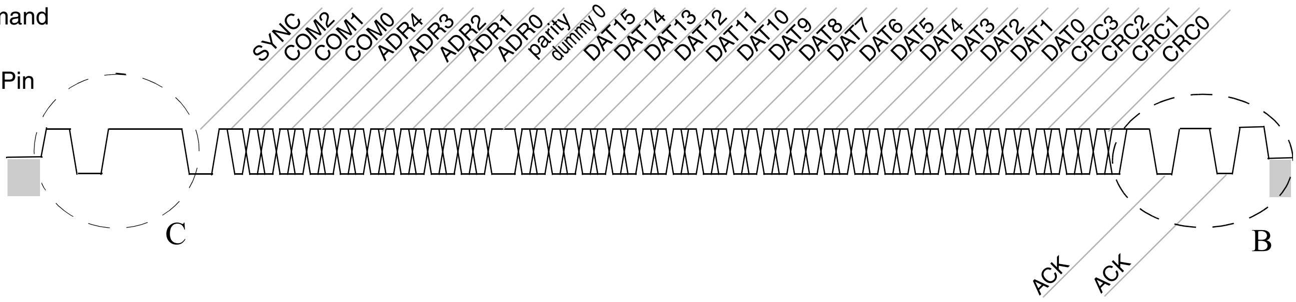
SYNC: start bit (always 0)
DAT: data bit
CRC: CRC bit

ACK: acknowledge
: Analog output

Communication via Out-Pin / Bidirectional on Out-Pin

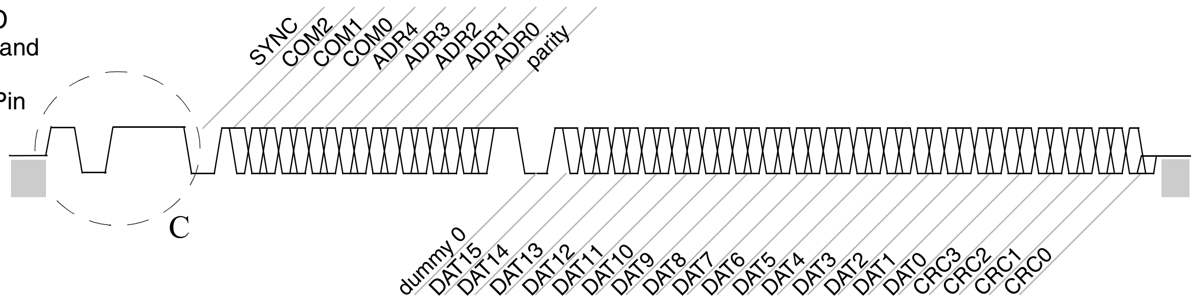
WRITE command

OUT-Pin



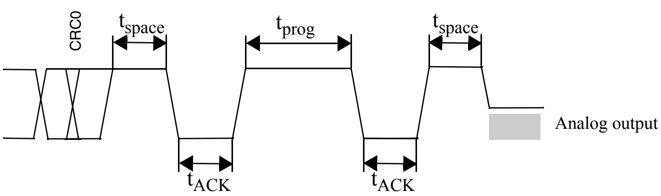
READ command

OUT-Pin



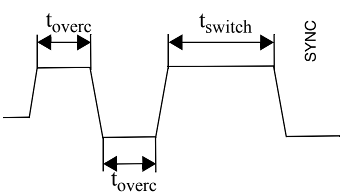
Detail B

OUT-Pin



Detail C

OUT-Pin



9.4. Telegram Parameter

Symbol	Parameter	Pin No.	Limit Values			Unit	Comment
			Min.	Typ.	Max.		
$V_{SUPProgram}$	V_{SUP} Voltage for EEPROM & NVRAM programming (during Programming)	1	5.7	6	6.5	V	
Biphase frame characteristic of the host							
t_{h_bbit}	Host biphase bit time		972	1024	1075	µs	
t_{h_bhb}	Host biphase half bit time		0.45	0.5	0.55	t_{h_bbit}	
t_{h_bifsp}	Host biphase interframe space		3			t_{h_bbit}	
	Slew rate		-	2	-	V/µs	
Biphase frame characteristic of the Sensor							
t_{s_bbit}	Sensor biphase bit time		820	1024	1225	µs	
t_{s_bhb}	Sensor biphase half bit time		-	0.5	-	t_{s_bbit}	
t_{s_bifsp}	Sensor biphase interframe space		2	-	-	t_{s_bbit}	
t_{s_bresp}	Sensor biphase response time		1	-	5	t_{s_bbit}	
	Slew rate		-	2	-	V/µs	
Detail A (programming via V_{SUP})							
V_{SUPL}	Supply Voltage for Low Level during Programming through Sensor V_{SUP} Pin	1	5.8	6.3	6.6	V	
V_{SUPH}	Supply Voltage for High Level during Programming through Sensor V_{SUP} Pin	1	6.8	7.3	7.8	V	
Detail B							
t_{space}	Time before and after acknowledge	3	-	1	-	t_{s_bbit}	
t_{ACK}	Acknowledge time	3	-	1	-	t_{s_bbit}	
t_{prog}	Programming time	3	-	4.5	-	ms	
Detail C (programming via OUT-Pin)							
t_{overc}	Over current pulse duration	3	-	2	-	ms	
t_{switch}	Time to switch sensor from application mode into programming mode	3	-	4	-	ms	
I_{oth}	Over current threshold	3		+/-10		mA	

Note: Only with programming via OUT-Pin is it possible to connect and program two sensors to the same V_{SUP} and same GND line.

9.5. Available sensor commands

The sensor supports 3 commands which provide read and write access to the whole memory (NVRAM;ROM, RAM, register).

The used write data frame and read data frame contains of 5 address bits only. A set base address command which defines a base address, expands the accessible address range to 8 bit.

In case of a unknown command, the sensor does neither transmit an acknowledge nor a body.

Table 9–15: Available sensor commands

Command	COM b[2:0]	frame type	ADR b[4:0]	DAT b[15:0] (RD/WD)
Read	1	read	offset address (0 to 31)	data read from address = ADR
Set base address	3	write	don’t care	base address 0,1,2,3
Write	6	write	offset address (0 to 31)	data which is written to address = ADR

9.5.1. Set base address

The set base address telegram functions as preparation for the write telegram and the read telegram. It uses the write data frame. Bit [15:2] are don’t care bit 0 and bit1 are concatenated to the address.The sensor transmits an acknowledge if a communication error has not been detected.

9.5.2. Read

The read telegram uses the read data frame. The sensor transmits the data of the effective address after the header has been successful received and the effective address is permitted. Otherwise, the sensor does not respond.

The effective address is calculated by the base address plus offset address. The offset address is defined by the address bits of the header (A b[4:0]).

9.5.3. Write

The write telegram uses the write data frame. The sensor saves the received address to the calculated effective address and transmits an acknowledge after the header and body has been successful received and the effective address is permitted. Otherwise, the command is discarded and the sensor transmits no acknowledge.

A write telegram is also discarded while EEPROM programming. During the NVPROM programming sequence, a write command is discarded.

9.5.4. CRC

The data bits are always followed by 4 CRC bits. For all commands but read the CRC result is calculated of all protocol bits, including command, address, parity and data bits.

For read command, the CRC result is calculated of data bits only (16).

The polynomial for the CRC calculation is always X^4+X+1.

In case of correct command detection (parity, CRC and command address if applicable), an ACK is sent as an answer.

9.5.5. Parity check

For the command and address bits, an “odd” parity check is used. In the case of an even number of “1”s, the parity bit has to be “1”. In the case of an odd number of “1”s, the parity bit has to be “0”.

9.5.6. Protocol Error Handling

The sensor is detecting protocol errors and command errors.

The sensor is detecting following communication errors:

– invalid parity
– invalid checksum
– command error
  A command error occurs when the command is either unknown or the execution has failed.

9.6. HAL 24xy - Board commands

Table 9–16: HAL 24xy commands

Action	Command	Parameter	Remarks
set base address	xxsb**STR**	**STR** = <A1><A0><D3><D2><D1> <D0><CRC> return value: <ST>:<R5><R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* set base address 1 => xxsb000001d <= 0:000000
write data	xxw**STR**	**STR** = <A1><A0><D3><D2><D1> <D0><CRC> return value: <ST>:<R5><R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* write 3333 to address b => xxw0b33333 <= 0:000000
read data	xxr**STR**	**STR** = <A1><A0> return value: <ST>:<R3><R2><R1><R0><CRC>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* read address b => xxr0b <= 0:3333E

9.7. Locking of the Sensor

For reliability in service, it is mandatory to set the LOCK bit after final adjustment and programming.

The success of the LOCK process should be checked by reading the status of the LOCK bit after locking.

It is also mandatory to check the acknowledges of the sensor or to read/check the status of the PROG register after each store sequence to verify if the programming of the sensor was successful.

Electro-static discharges (ESD) may disturb the supply voltage during programming. Please take precautions against ESD.

For the programming during product development and also for production purposes, a programming tool including hardware and software is available on request. It is recommended to use the Micronas tool kit for an easy product development.

Note: It is not possible to write or to read a register after locking.

10. HAL 2810 – Board Commands

The HAL 28xy family consists of members with different digital interfaces like LIN, PWM and SENT. All members within this family can be programmed without any additional programming pin. Programming is done via LIN frames or Biphase-M telegrams depending on the family member.

10.1. LIN Interface

Note: For the LIN2.0 Mode, the hardware configuration of the HAL-APB V1.x must be changed. For details (see Section 3 on page 10).

The LIN Interface on the HAL-APB V1.x is intended to be used with one HAL 2810 sensor as a point to point communication. The firmware comprises a master driver for LIN protocol 2.0. A slave mode is not implemented.

10.1.1. LIN Interface Mode Configuration

Note: In case of LIN configuration the bus voltage level is equal to V_{Board} (normally 18 V).

For using this mode correct the V_{DD_HAL} has to be set to 12V too. This is done by sending the two commands

=> ftvdp1(setting programming voltage = 12V)

=> vho1(switching on V_{DD_HAL})

10.1.2. Schedule Tables

Different schedule tables allow the use of unconditional frames and diagnostic/configuration frames. It is possible to switch between 4 schedule tables, where tables 1 to 3 handle unconditional frames and table 4 handles configuration frames. The PIDs given in the table are the default settings in the HAL-APB V1.x firmware schedule table settings. The PIDs of the unconditional frames in the schedule tables 1 to 3 can be changed in the test terminal.

This can be only done while the corresponding table is not scheduled! The number of response bytes is set to a fixed value for each frame inside a schedule table. If a PID is changed accordingly, this length is applied to the new PID. Changed PIDs are lost after powering off the HAL-APB V1.x. Each schedule table includes only two LIN frames! The scheduling time between frames is set to 20 ms (fixed).

Note: To apply more user friendly behavior, especially for the saving of special PID settings, the calling application software should implement the preservation of the PID settings.

Table 10–17: Schedule Tables unconditional frames of the HAL-APB V1.x

Table Number	Description
ID_Table_1	Default scheduling table after startup. It comprises a set address to prepare a data access (W/nR=0) and a Read 2 bytes frame. The address is arbitrary. the default scheduling prevents the sensor from going into sleep. Frame1: PID = 0x03 (set address), 3 bytes Frame2: PID = 0xc4 (Read 2 bytes), 2 bytes
ID_Table_2	This table prepares an address to read from (W/nR=0) and performs a 4 byte read from there. Frame1: PID = 0x03 (set address), 3 bytes Frame 2: PID = 0x85, (Read 4 bytes), 4 bytes
ID_Table_3	This table comprises a frame to write a byte to an address (W/nR=1). A second frame includes only a dummy 4 byte read from this address. Frame1: PID = 0x03 (set address), 3 bytes Frame2: PID = 0x85 (Read 4 bytes), 4 bytes

Table 10–18: Schedule table configuration frames of the HAL-APB V1.x

Table Number	Description
ID_Table_4	This table contains configuration frames to schedule. The two frames are a request (PID60) and a response (PID61) frame. the data bytes to be used with the request frame can be set by the user. Frame1: PID60 = 0x3C, 8bytes Frame2: PID61 =0x7D, 8bytes

10.1.3. Error Handling

A global error flag is set if a LIN communication error, e.g. a shorted bus or a disconnected slave has been detected. There is no certain fault confinement implemented but only this global error information. The last error information can be kept alive or reset.

10.2. HAL 2810 (LIN2.0) – Board commands

Note: For general board commands see Table 7–11 on page 21

Table 10–19: HAL 2810 commands

Action	Command	Parameter	Remarks
switch schedule table	lsstab**N**	**N** = 1...4 return value: <ST>:<R7><R6><R5><R4><R3><R2><R1><R0>	**N** schedule table No. <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 8-digit hex No. *Example* *=> lsstab1* *<= <ST>:00000000*
set address = Prepare Data Access	lwpa**STR**	**STR** = <A3><A2><A1><A0> return value: <ST>:<R7><R6><R5><R4><R3><R2><R1><R0>	<A> address as 4-digit hex No. <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 8-digit hex No. *Example* *=> lwpa*3047 *=> <ST>:00000000*
set address = Write Byte	lwa**STR**	**STR** = <A3><A2><A1><A0><D1><D0> return value: <ST>:<R7><R6><R5><R4><R3><R2><R1><R0>	<A> address as 4-digit hex No. <D> data as 2-digit hex No <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 8-digit hex No. *Example* *=> lwa304711* *=> <ST>:00000000*
send single shot of schedule table 3	lsos	return value: <ST>:<R7><R6><R5><R4><R3><R2><R1><R0>	<R> received data as 8-digit hex No. <ST> board status (see Table 6–9 on page 20 for details) *Example* *=> lsos* *<= <ST>:00000000*
read last data	lrd	return value: <ST>:<R7><R6><R5><R4><R3><R2><R1><R0>	<ST> board status (see Table 6–9 on page 20 for details) <R> received data as 8-digit hex No. if lsstab1: <R3>... <R0> if lsstab2: <R7>... <R0>

10.3. Locking of the Sensor

A lock bit directly affects the sensor hardware:

1. Any further write access to the corresponding EEPROM area is blocked.

2. The EEPROM cells are permanently connected to their RAM layer.

If the lock bit C0LOC is not set the configuration of the DSP may be controlled by the data stored in the RAM layer only. The customer must verify and (if necessary) refresh the configuration data periodically.

Note: It is mandatory to lock the DSP CONIFIG EEPROM when the sensors are used for qualification tests and in field applications.

If the lock bit C1LOC is not set the configuration of the LIN bus may be controlled by the data stored in the RAM layer only. The customer must verify and (if necessary) refresh the configuration data periodically.

Note: Micronas recommends to lock the COM CONFIG area when the sensors are used for qualification tests and in field applications.

The lock bit does not restrict the read access to the memory. Any permitted address can be read independent of the lock bit.

– Before setting a lock bit (C0LOC or C1LOC) verify the register contents of the corresponding EEPROM area.

– Check for the effectiveness of the lock bit after locking.

This can be done by a write attempt to one of the EEPROM registers.

Note: The lock mechanism gets active with the next reset after setting the lock bit. Once the lock mechanism is active the corresponding EEPROM area cannot be reprogrammed. In particular the lock bits C0LOC, C1LOC, MLOC cannot be cleared.

11. HAL 283x / HAL 2850

The HAL 283x/HAL2850 are universal magnetic field sensors with SENT/PWM output, based on the Hall effect. Magnetic field, perpendicular to the branded side of the sensor provides a output signal direct proportional to the applied magnetic flux through the Hall plate. Details on features and specification are described in the data sheet.

11.1. Programming interface

The sensor can be programmed via output voltage modulation. The sensor answers with a modulation of the output voltage.

A logical “0“ is coded as no level change within the bit time. A logical “1“ is coded as a level change of typically 50% of the bit time. After each bit, a level change occurs (see Fig. 11–1).

The serial telegram is used to transmit the EEPROM content, error codes and digital values of the magnetic field from and to the sensor.

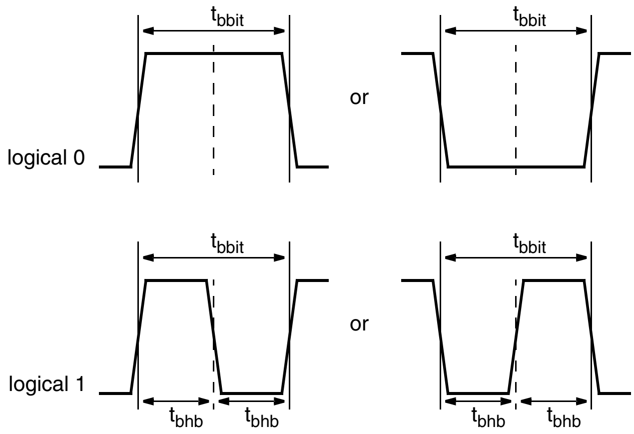


Fig. 11–1: Definition of logical 0 and 1 bit

11.2. Command Structure of Protocol for communication via OUT-Pin

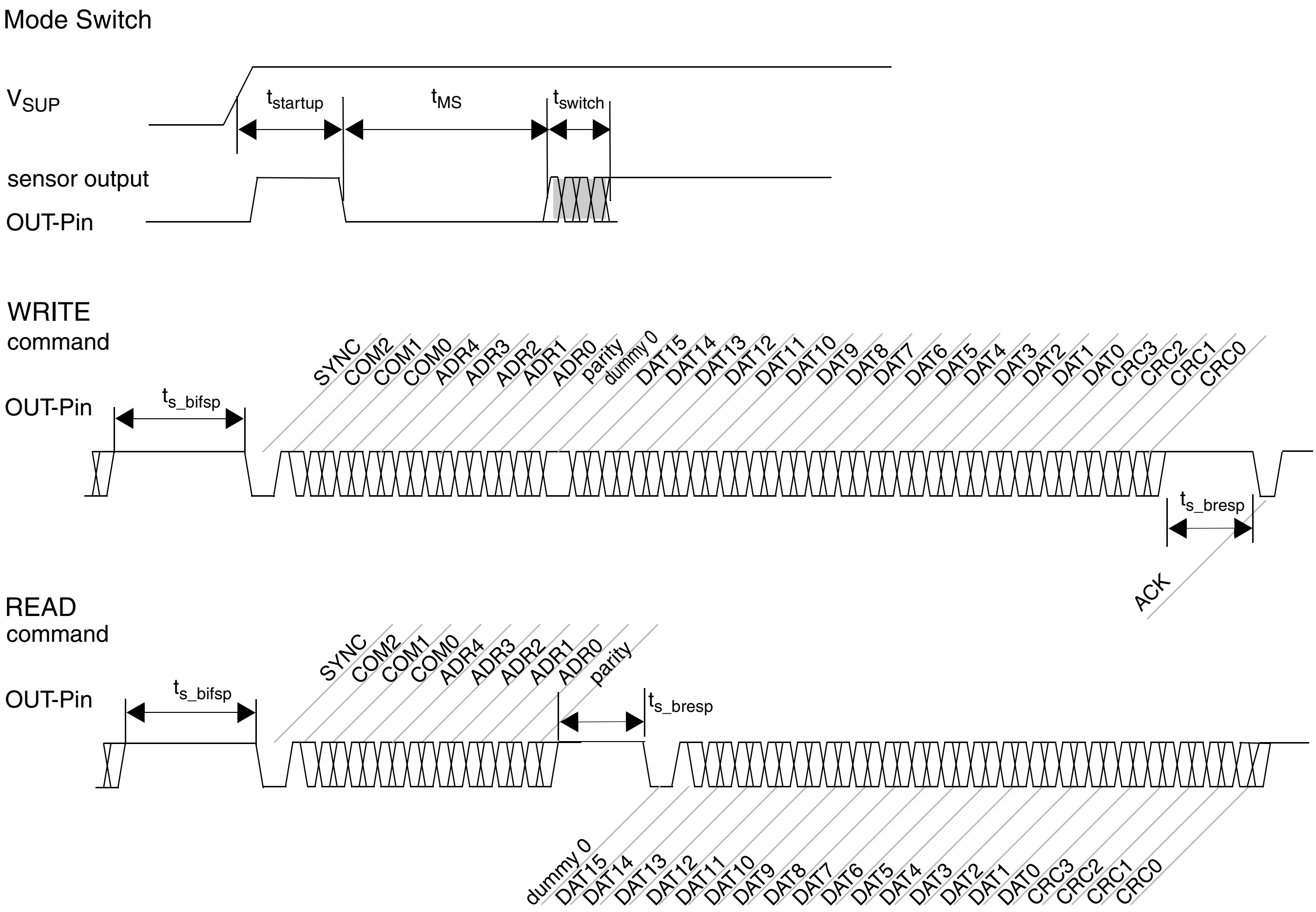
command structure

COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

SYNC: start bit (always 0)
DAT: data bit
CRC: CRC bit

ACK: acknowledge
: digital Output

Communication via Out-Pin / Bidirectional on Out-Pin



11.3. Telegram Parameters

Symbol	Parameter	Pin No.	Limit Values			Unit	Comment
			Min.	Typ.	Max.		
$V_{SUPProgram}$	V_{SUP} Voltage for EEPROM programming (during Programming)	1	5.5	-	18	V	
$t_{startup}$	Start up time **(HAL2850)**	3	-	-	8 [1)]	ms	PERIOD = 0 (0.5 ms)
					9 [1)]	ms	PERIOD = 1 (1 ms)
					10 [1)]	ms	PERIOD = 3 (2 ms)
					10 [1)]	ms	PERIOD = 7 (4 ms)
					20 [1)]	ms	PERIOD = 11 (8 ms)
					40 [1)]	ms	PERIOD = 15 (16 ms)
					80 [1)]	ms	PERIOD = 19 (32 ms)
	Start up time **(HAL283x)**				1[2)]	ms	
t_{MS}	Mode switch	3	130	-	560	ms	
t_{switch}			0		12	ms	
Biphase frame characteristic of the host							
t_{h_bbit}	Host biphase bit time		972	1024	1075	µs	
t_{h_bhb}	Host biphase half bit time		0.45	0.5	0.55	t_{h_bbit}	
t_{h_bifsp}	Host biphase interframe space		3			t_{h_bbit}	
	Slew rate		-	2	-	V/µs	
Biphase frame characteristic of the Sensor							
t_{s_bbit}	Sensor biphase bit time		820	1024	1225	µs	
t_{s_bhb}	Sensor biphase half bit time		-	0.5	-	t_{s_bbit}	
t_{s_bifsp}	Sensor biphase interframe space		2	-	-	t_{s_bbit}	
t_{s_bresp}	Sensor biphase response time		1	-	5	t_{s_bbit}	
	Slew rate		-	2	-	V/µs	

[1)]For POST= 0, 10 ms must be added when POST=1 and 3, 32 ms must be added when POST=2
2)For POST= 0, 10 ms must be added when POST=1, 2 or 3, 7 ms must be added

11.4. Available sensor commands

The sensor supports 5 commands which provide read and write access to the whole memory (ROM, RAM, register). Then mentioned commands allows for example to read the hall value, the temperature value, and to program the EEPROM.

The used write data frame and read data frame contains of 5 address bits only. A "set base address" command which defines a base address, expands the accessible address range to 16 bit.

In case of a unknown command, the sensor does neither transmit an acknowledge nor a body.

Table 11–20: Supported commands

Command	COM b[2:0]	frame type	A b[4:0]	DAT b[15:8] (RD/WD)	DAT b[7:0] (RD/WD)
Read with absolute address	0	read	absolute address (0 to 31, byte aligned)	data read from address = A + 1	data read from address = A
Read with base address	1	read	address offset (0 to 31, byte aligned)	data read from address = base address + A+ 1	data read from address = base address + A
Set base address	3	write	don’t care	base address (16 bit, byte aligned)	
Write byte with base address	5	write	address offset (0 to 31, byte aligned)	don’t care	data which is written to address = base address + A
Write word (16-bit) with base address	6	write	address offset (0 to 31, byte aligned)	data which is written to address = base address + A +1	data which is written to address = base address + A

11.4.1. Set base address

The “set base address” telegram functions as preparation for write telegrams and the "read with base address" telegram. It uses the write data frame. The base address is defined by DAT[15:0]. The sensor transmits an acknowledge if a communication error has not been detected.

11.4.2. Read with absolute address

The "read with absolute address" telegram uses the read data frame. The sensor transmits the data of the effective address after the header has been successful received and the effective address is permitted. Otherwise, the sensor does not respond.

The effective address is defined by the address bits of the header (A b[4:0]). Thus, this telegram can be used for reading the lower 32 byte only.

11.4.3. Read with base address

The "read with base address" telegram uses the read data frame. The sensor transmits the data of the effective address after the header has been successful received and the effective address is permitted. Otherwise, the sensor does not respond. The effective address is calculated by the base address plus offset address. The offset address is defined by the address bits of the header (A b[4:0]).

11.4.4. Write byte with base address

The "write byte with base address" telegram uses the write data frame. The sensor saves the received byte (DAT b[7:0]) to the calculated effective address and transmits an acknowledge after the header and body has been successful received and the effective address is permitted. Otherwise, the command is discarded and the sensor transmits no acknowledge.

A "write byte with base address" telegram is also discarded while EEPROM programming.

11.4.5. Write word with base address

This telegram is similar to the "write byte with base address" telegram. Unlike the "write byte with base address" telegram, this telegram is used for writing 16-bit data to the effective address and the effective address+1.

11.4.6. Special Cases

After reset, the IC does not execute a "read with base address", "write byte with base address" or "write word with base address" command till a "set base address" telegram has been received.

During the EEPROM programming sequence (clear and set), a "write byte with base address" command or "write word with base address" command is discarded. It is recommended to pause the communication while the clear and set sequence is operated.

11.4.7. Protocol Error Handling

The sensor is detecting and logging protocol errors and command errors.

The sensor is detecting following communication errors:

- frame error
A frame error can occur due to a too short interframe space, a non valid start bit ("0"), a non valid dummy bit "0", to less bits of a frame (time-out), or invalid pulse widths.

- invalid parity

- invalid checksum

- command error
A command error occurs when the command is either unknown or the execution has failed.

In case of an error, the sensor neither transmits an acknowledge nor a body.

11.5. HAL 283x / HAL 2850 – Board commands

Note: For general board commands see Table 7.2 on page 21

Table 11–21: HAL 283x / HAL 2850 commands

Action	Command	Parameter	Remarks
read PWM-Period and Pulse width (only HAL 2850)	pr**N**	**N** = 0 / 1 return value: <ST>:<P3><P2><P1><P0><W3><W2><W1><W0>	**N** trigger on falling/rising PWM edge <ST> board status (see Table 6–9 on page 20 for details) <P> 4digit Period <W>4digit Pulse width *Example* => pr1 (OP-bit=0) <= <ST>:13AE0A00 - Conversion of PWM-Period: 0x13AE = 5038_{dec} / 10000 = 0,5ms - Conversion of Pulse width: 0x0A00 = 2560_{dec} / 10000 = 0,26ms - Calculation of Duty-Cycle: 2560/ 5038 = 50,8%
mode switch	pcms	return value: return value: <ST>:<R4><R3><R2><R1><R0>	switch HAL 283x/HAL 2850 to Biphase-M programming mode <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 5-digit hex No.
read from page 0 returns data read from address to address+1	pxr0**STR**	**STR** = <A1><A0> return value: <ST>:<R3><R2><R1><R0><CRC>	<A> address as 2-digit hex No. <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. <CRC> checksum as 1-digit hex No. *Example* *=> pxr002* *=> <ST>:0FFB2*
set base address	pxsb**STR**	**STR** = <A3><A2><A1><A0><CRC> return value: <ST>:<R4><R3><R2><R1><R0>	<A> base address as 4-digit hex No. <CRC> checksum <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 5-digit hex No. *Example* *=> pxsb30006* *=> <ST>:000000*

Table 11–21: HAL 283x / HAL 2850 commands

Action	Command	Parameter	Remarks
read byte with base address returns data from address to address+1	pxrb**STR**	**STR** = <A1><A0> return value: <ST>:<R3><R2><R1><R0><CRC>	<A> address as 2-digit hex No. <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. <CRC> checksum *Example* *=> pxrb00* *<= <ST>:D4537*
write byte with base address	pxwb**STR**	**STR** = <A1><A0><D1><D0><CRC> return value: <ST>:<R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 2-digit hex No. <CRC> checksum <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 5-digit hex No. *Example* *=> pxwb00000* *<= <ST>:000000*
write word with base address	pxww**STR**	**STR** = <A1><A0><D3><D2><D1><D0><CRC> return value: <ST>:<R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum <ST> board status (see Table 6–9 on page 20 for details) <R> received data as 5-digit hex No.

11.6. Locking of the Sensor

For reliability in service, it is mandatory to set the LOCK bit after final adjustment and programming.

The success of the LOCK process should be checked by reading the status of the LOCK bit after locking.

It is also mandatory to check the status bit of the received data or read/check the EEPCTRL register after each store sequence to verify if the programming of the sensor was successful.

Electro-static discharges (ESD) may disturb the supply voltage during programming. Please take precautions against ESD.

For the programming during product development and also for production purposes, a programming tool including hardware and software is available on request. It is recommended to use the Micronas tool kit for an easy product development.

Note: It is not possible to write or to read a register after locking.

12. HAL 3625, HAL 3675, HAL 385x and HAL 387x

The HAL 36x5, HAL 385x and HAL 387x are members of a new generation of Hall-effect sensors with vertical hall plate technology. With the new vertical Hall technology it is possible to directly measure rotation angles in a range of 0° to 360° and linear movements with simple magnetic circuits.Details on features and specification are described in the data sheet.

12.1. Programming interface

The sensor can be programmed via supply voltage modulation or via output voltage modulation. The default mode is the programming via the output voltage modulation. The sensor answers with a modulation of the output voltage.

A logical “0“ is coded as no level change within the bit time. A logical “1“ is coded as a level change of typically 50% of the bit time. After each bit, a level change occurs (see Fig. 12–1).

The serial telegram is used to transmit the EEPROM content, error codes and digital values of the magnetic field from and to the sensor.

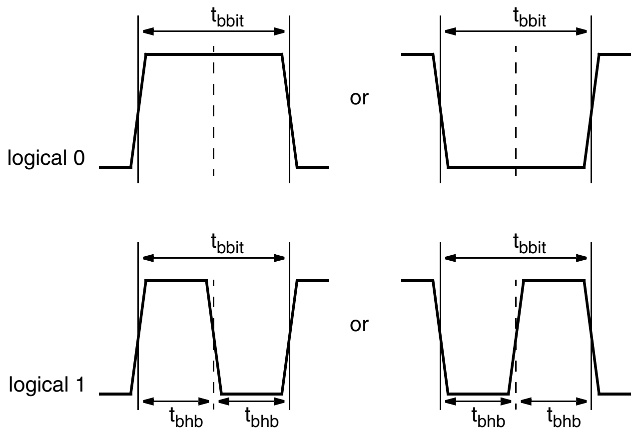


Fig. 12–1: Definition of logical 0 and 1 bit

12.2. Command Structure of Protocol for communication via V_{SUP}

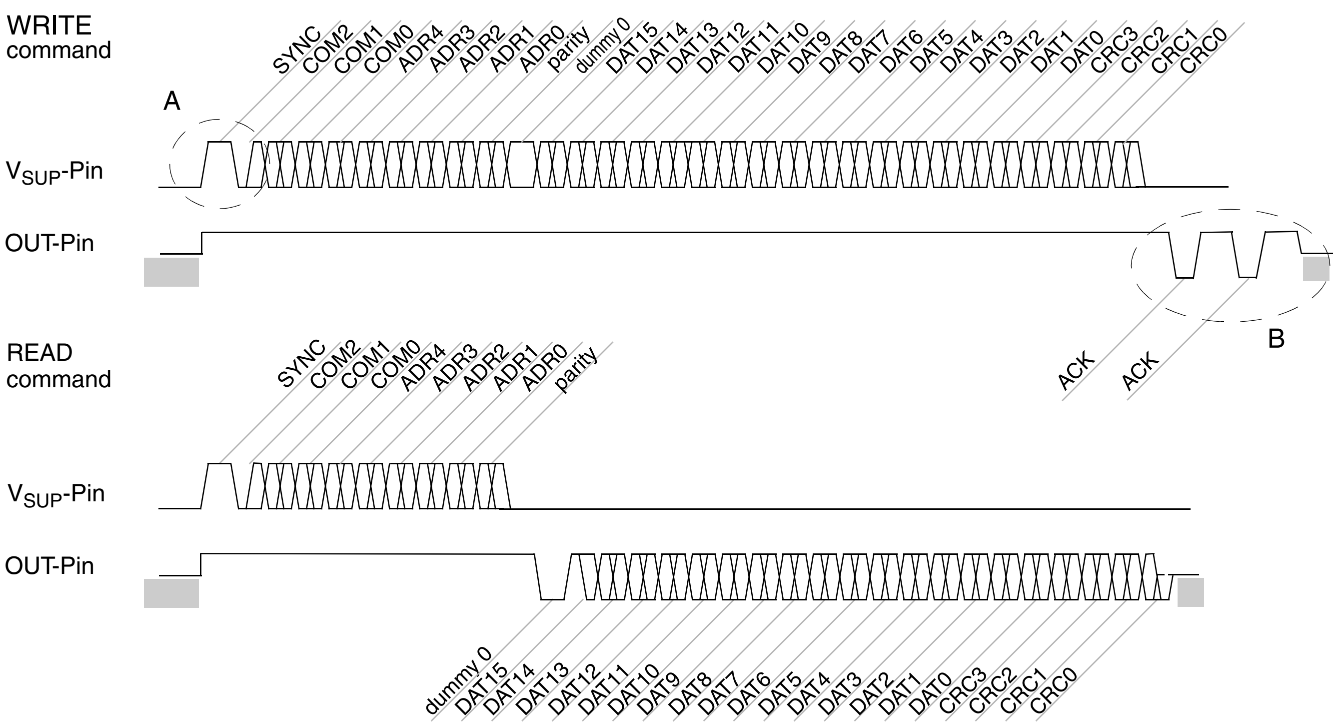
command structure

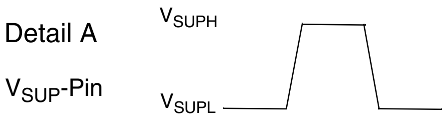
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

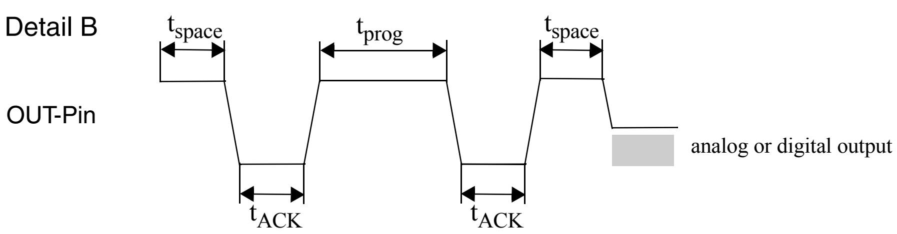
SYNC: start bit (always 0)
DAT: data bit
CRC: CRC bit

ACK: acknowledge
(grey box): analog or digital output

Communication via V_{SUP}-Pin (Biphase-In = V_{SUP} / Biphase-Out = OUT-Pin)







12.3. Telegram Parameters

Symbol	Parameter	Pin No.	Limit Values			Unit	Comment
			Min.	Typ.	Max.		
$V_{SUPProgram}$	V_{SUP} Voltage for EEPROM & NVRAM programming (during Programming)	1	5.7	6	6.5	V	
Biphase frame characteristic of the host							
t_{h_bbit}	Host biphase bit time		972	1024	1075	µs	
t_{h_bhb}	Host biphase half bit time		0.45	0.5	0.55	t_{h_bbit}	
t_{h_bifsp}	Host biphase interframe space		3			t_{h_bbit}	
	Slew rate		-	2	-	V/µs	
Biphase frame characteristic of the Sensor							
t_{s_bbit}	Sensor biphase bit time		820	1024	1225	µs	
t_{s_bhb}	Sensor biphase half bit time		-	0.5	-	t_{s_bbit}	
t_{s_bifsp}	Sensor biphase interframe space		2	-	-	t_{s_bbit}	
t_{s_bresp}	Sensor biphase response time		1	-	5	t_{s_bbit}	
	Slew rate		-	2	-	V/µs	
Detail A							
V_{SUPL}	Supply Voltage for Low Level during Programming through Sensor V_{SUP} Pin	1	5.8	6.3	6.6	V	
V_{SUPH}	Supply Voltage for High Level during Programming through Sensor V_{SUP} Pin	1	6.8	7.3	7.8	V	
Detail B							
t_{space}	Time before and after acknowledge		-	1	-	t_{s_bbit}	
t_{ACK}	Acknowledge time		-	1	-	t_{s_bbit}	
t_{prog}	Programming time		-	4.5	-	ms	

12.4. Command Structure of Protocol for communication via OUT-Pin

command structure

COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

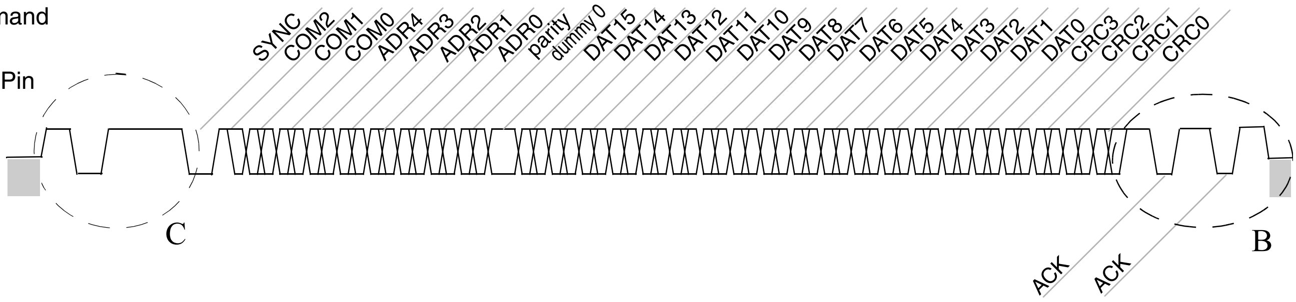
SYNC: start bit (always 0)
DAT: data bit
CRC: CRC bit

ACK: acknowledge
: Analog or digital output

Communication via Out-Pin / Bidirectional on Out-Pin

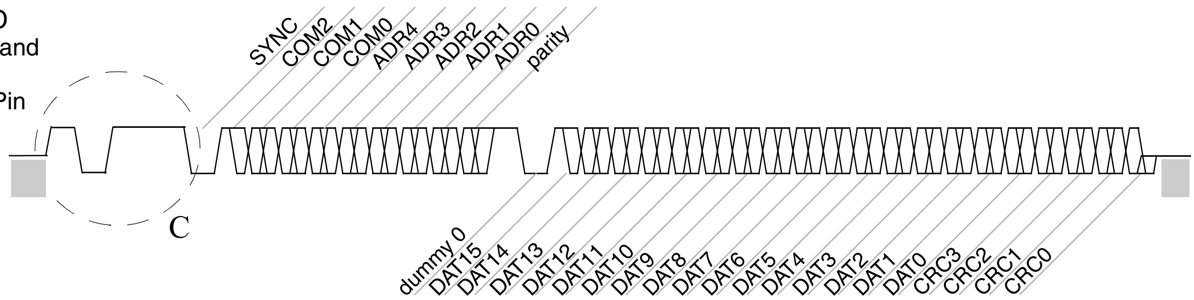
WRITE command

OUT-Pin



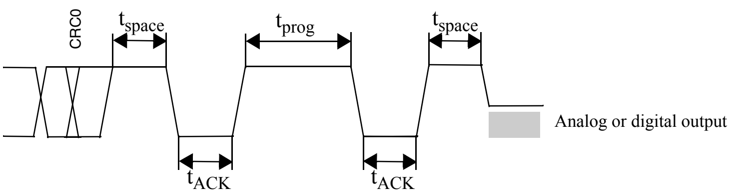
READ command

OUT-Pin



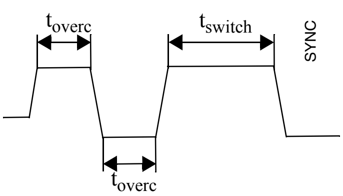
Detail B

OUT-Pin



Detail C

OUT-Pin



12.5. Telegram Parameter

Symbol	Parameter	Pin No.	Limit Values			Unit	Comment
			Min.	Typ.	Max.		
$V_{SUPProgram}$	V_{SUP} Voltage for EEPROM & NVRAM programming (during Programming)	1	5.7	6	6.5	V	
Biphase frame characteristic of the host							
t_{h_bbit}	Host biphase bit time		972	1024	1075	µs	
t_{h_bhb}	Host biphase half bit time		0.45	0.5	0.55	t_{h_bbit}	
t_{h_bifsp}	Host biphase interframe space		3			t_{h_bbit}	
	Slew rate		-	2	-	V/µs	
Biphase frame characteristic of the Sensor							
t_{s_bbit}	Sensor biphase bit time		820	1024	1225	µs	
t_{s_bhb}	Sensor biphase half bit time		-	0.5	-	t_{s_bbit}	
t_{s_bifsp}	Sensor biphase interframe space		2	-	-	t_{s_bbit}	
t_{s_bresp}	Sensor biphase response time		1	-	5	t_{s_bbit}	
	Slew rate		-	2	-	V/µs	
Detail A (programming via V_{SUP})							
V_{SUPL}	Supply Voltage for Low Level during Programming through Sensor V_{SUP} Pin	1	5.8	6.3	6.6	V	
V_{SUPH}	Supply Voltage for High Level during Programming through Sensor V_{SUP} Pin	1	6.8	7.3	7.8	V	
Detail B							
t_{space}	Time before and after acknowl-edge	3	-	1	-	t_{s_bbit}	
t_{ACK}	Acknowledge time	3	-	1	-	t_{s_bbit}	
t_{prog}	Programming time	3	-	4.5	-	ms	
Detail C (programming via OUT-Pin)							
t_{overc}	Over current pulse duration	3	-	2	-	ms	
t_{switch}	Time to switch sensor from application mode into programming mode	3	-	4	-	ms	
I_{oth}	Over current threshold	3		+/-10		mA	

Note: Only with programming via OUT-Pin is it possible to connect and program two sensors to the same V_{SUP} and same GND line.

12.6. Available sensor commands

The sensor supports 3 commands which provide read and write access to the whole memory (NVRAM;ROM, RAM, register).

The used write data frame and read data frame contains of 5 address bits only. A set base address command which defines a base address, expands the accessible address range to 8 bit.

In case of a unknown command, the sensor does neither transmit an acknowledge nor a body.

Table 12–22: Available commands

Command	COM b[2:0]	frame type	ADR b[4:0]	DAT b[15:0] (RD/WD)
Read	1	read	offset address (0 to 31)	data read from address = ADR
Set base address	3	write	don’t care	base address 0,1,2,3
Write	6	write	offset address (0 to 31)	data which is written to address = ADR

12.6.1. Set base address

The set base address telegram functions as preparation for the write telegram and the read telegram. It uses the write data frame. Bit [15:2] are don’t care bit 0 and bit1 are concatenated to the address.The sensor transmits an acknowledge if a communication error has not been detected.

12.6.2. Read

The read telegram uses the read data frame. The sensor transmits the data of the effective address after the header has been successful received and the effective address is permitted. Otherwise, the sensor does not respond.

The effective address is calculated by the base address plus offset address. The offset address is defined by the address bits of the header (A b[4:0]).

12.6.3. Write

The write telegram uses the write data frame. The sensor saves the received address to the calculated effective address and transmits an acknowledge after the header and body has been successful received and the effective address is permitted. Otherwise, the command is discarded and the sensor transmits no acknowledge.

A write telegram is also discarded while EEPROM programming. During the NVPROM programming sequence, a write command is discarded.

12.6.4. CRC

The data bits are always followed by 4 CRC bits. For all commands but read the CRC result is calculated of all protocol bits, including command, address, parity and data bits.

For read command, the CRC result is calculated of data bits only (16).

The polynomial for the CRC calculation is always X^4+X+1.

In case of correct command detection (parity, CRC and command address if applicable), an ACK is sent as an answer.

12.6.5. Parity check

For the command and address bits, an “odd” parity check is used. In the case of an even number of “1”s, the parity bit has to be “1”. In the case of an odd number of “1”s, the parity bit has to be “0”.

12.6.6. Protocol Error Handling

The sensor is detecting protocol errors and command errors.

The sensor is detecting following communication errors:

– invalid parity

– invalid checksum

– command error
A command error occurs when the command is either unknown or the execution has failed.

12.7. HAL 3625, HAL 3675, HAL 385x and HAL 385x– Board commands

Table 12–23: HAL 3625, HAL 3675, HAL 385x and HAL 387x board commands

Action	Command	Address	Data
set base address	xxsb**STR**	**STR** = <A1><A0><D3><D2><D1> <D0><CRC> return value: <ST>:<R5><R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* set base address 1 => xxsb000001d <= 0:000000
write data	xxw**STR**	**STR** = <A1><A0><D3><D2><D1> <D0><CRC> return value: <ST>:<R5><R4><R3><R2><R1><R0>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* write 37B7 to address 8 => xxw0837B76 <= 0:000000
read data	xxr**STR**	**STR** = <A1><A0> return value: <ST>:<R4><R3><R2><R1><CRC>	<A> address as 2-digit hex No. <D> data as 4-digit hex No. <CRC> checksum as 1-digit hex No. <ST> = Status of Board (see Table 6–9 on page 20 for details) <R> received data as 4-digit hex No. *Example:* read address 8 => xxr08 <= 0:37B75

12.8. Locking of the Sensor

For reliability in service, it is mandatory to set the LOCK bit after final adjustment and programming.

The success of the LOCK process should be checked by reading the status of the LOCK bit after locking.

It is also mandatory to check the acknowledges of the sensor or to read/check the status of the PROG register after each store sequence to verify if the programming of the sensor was successful. VER, PER should be 0 after each set/erase command. NVE should be 0 after set/erase command of the NVRAM.

Electro-static discharges (ESD) may disturb the supply voltage during programming. Please take precautions against ESD.

For the programming during product development and also for production purposes, a programming tool including hardware and software is available on request. It is recommended to use the Micronas tool kit for an easy product development.

Note: It is not possible to write or to read a register after locking.

13. Application Note History

1. HAL 1820, HAL 28xy, HAL 3625 Application Board HAL-APB V1.x, April, 16, 2010; APN000055_001EN. First release of the application note.
2. HAL 1820, HAL 28xy, HAL 3625 Application Board HAL-APB V1.x, March 24, 2011; APN000055_002EN. Second release of the application note.
   Major Changes:

– HAL-APB version 1.5 added

3. HAL 1820, HAL 28xy, HAL 3625, HAL 3675, HAL 385x and HAL 387x Application Board HAL-APB V1.x, Oct. 8, 2012, APN000055_003EN. Third release of the application note.
   Major Changes:

– Section 4.2 Recommended Wiring added.

– Section 4.3 Maintenance and Calibration added.

– Section 5 USB Driver Installation changed.

Micronas GmbH
Hans-Bunte-Strasse 19 · D-79108 Freiburg · P.O. Box 840 · D-79008 Freiburg, Germany
Tel. +49-761-517-0 · Fax +49-761-517-2174 · E-mail: docservice@micronas.com · Internet: www.micronas.com