MICRONAS

Application Note

HAL® 1820, HAL 242x, HAL 36xy, HAL 37xy, HAL 38xy

USB Programming Tool v1.0

3D|HAL® by Micronas vario|HAL® by Micronas

Edition Jan. 20, 2014
APN000093_001EN

Contents

Contents, continued

1. Introduction

1.1. General Information

The hardware and software description in this document is valid for the HAL USB Programming Tool v1.0

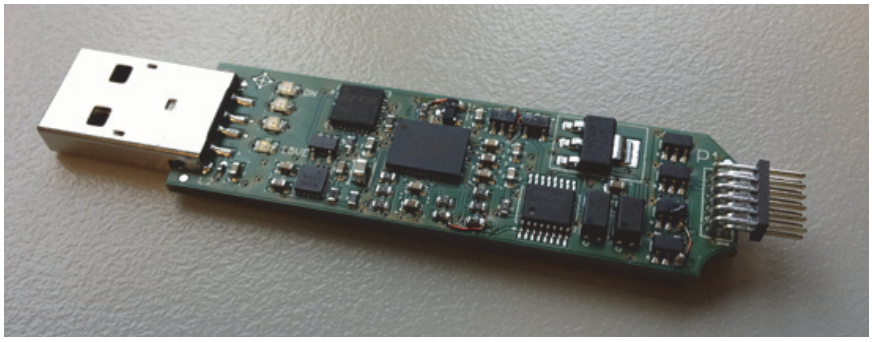


Fig. 1–1: HAL USB Programming Tool v1.0

1.2. Introduction

The HAL USB programming tool v1.0 is a board for programming the Micronas Hall-effect sensor families with analog and digital output formats. The board is equipped with a Micronas microcontroller HVC 2480B. It provides application software supporting a command interface for the communication with a PC. This allows the implementation of specific PC software for engineering purposes only.

Note: The HAL USB Programming Tool v1.0 is only for engineering purposes. For production Micronas recommend the HAL-APB V1.5 as programming hardware.

1.2.1. Supported HAL Sensors

The HAL USB programmer tool supports the sensors listed in Table 1–1.

Table 1–1: Supported sensors

Sensor	Remark
HAL 1820	Linear sensor with analog output
HAL 24xy	Linear sensor with analog/PWM output
HAL 3625	Direct angle sensor with analog output
HAL 3675	Direct angle sensor with PWM output

Table 1–1: Supported sensors

Sensor	Remark
HAL 37xy	2D position/direct angle sensor with analog/PWM/SENT 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

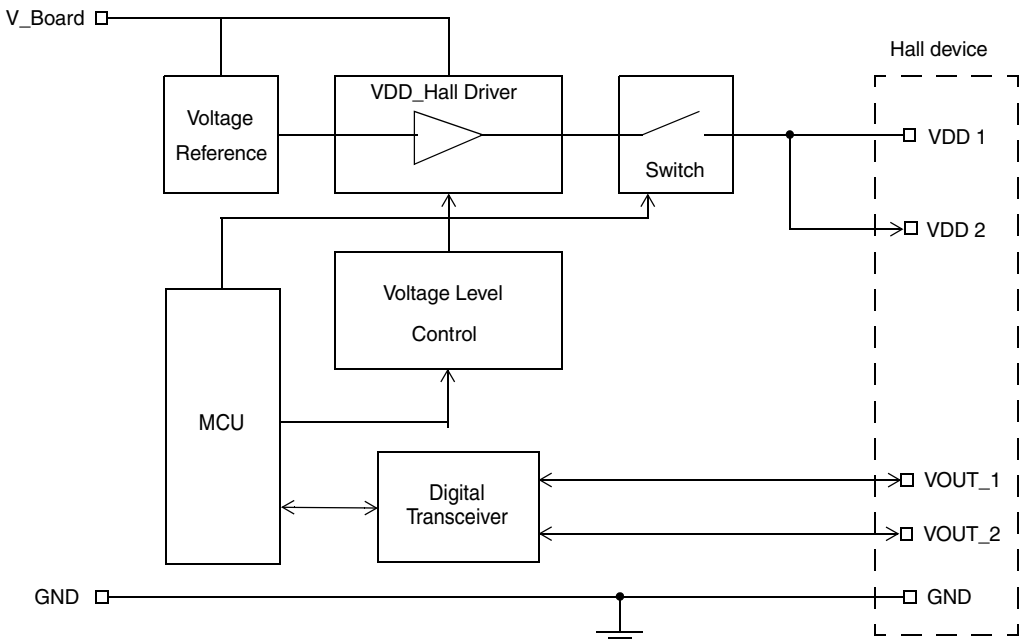


Fig. 1–2: Detailed view of HAL Interface

2. Getting started

2.1. USB Driver Installation

Note: The USB driver is not necessary for PC systems running with Windows7.

Installing the USB VCP Drivers

Plug in the HAL USB programming tool into a spare USB port.

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.

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

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

As an alternative you can open the device manager of your PC system to check if the system has configured the HAL USB programmer tool correctly.

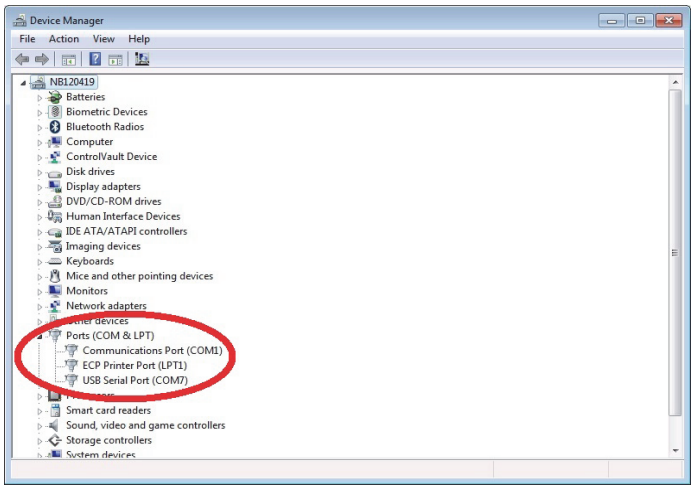


Fig. 2–3: Device manager

2.2. First Steps

2.2.1. Check HAL USB Programming Tool

– Connect the HAL SUB programming tool to the USB-port at your PC.

– Check if the power-on self-test was passed successfully. (After a very short blinking of the four LEDs they shut down.

2.2.2. Check Communication with PC and Hall Sensor Connection

Connect a Hall sensor with the HAL USB programming tool.

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 5 on page 12),

– switch Vdd on using the “vho1” command (see Section 6.2 on page 14),

– try to read out a register (see chapter of the used sensor type).

3. Board Configuration

HAL Interface Connector

Depending on the sensor type, up to two sensors can be connected to the board. For this purpose, a 12-pin connector is provided.

The following pin's are connected in parallel Pin No. 5 (V_{SUP} Sensor 1) and 6 (V_{SUP} Sensor 2) and Pin No. 3 (Common Sensor GND) and Pin No. 4 (Common Sensor GND). The pinning of the interface is described in Table 3–2.

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

Pin No.	Description
1	OUT Sensor 1
2	OUT Sensor 2
3	GND
4	GND
5	VDD Sensor 1
6	VDD Sensor 2
7	reserved
8	reserved
9	reserved
10	5 V
11	GND
12	SDAT

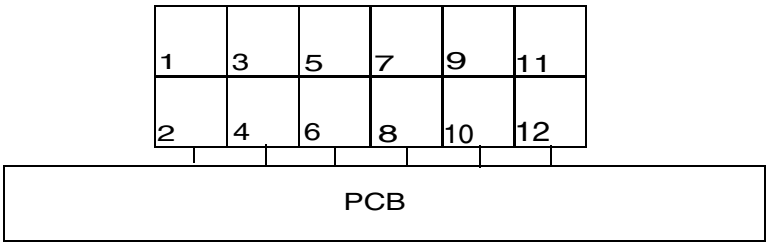


Fig. 3–4: Connector HAL1/2, front view

4. Specification

4.1. 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 close to power switches, electromagnetic actuators, and the like, EMI-compliant layout of the wiring is mandatory.

For recommended cable parameters, please refer to Table 4–3.

4.2. Maintenance and Calibration

The HAL 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 HAL USB programming tool. Otherwise, the board may be damaged, the sensor programming may be insufficient, and the reliability of the sensor reduced.

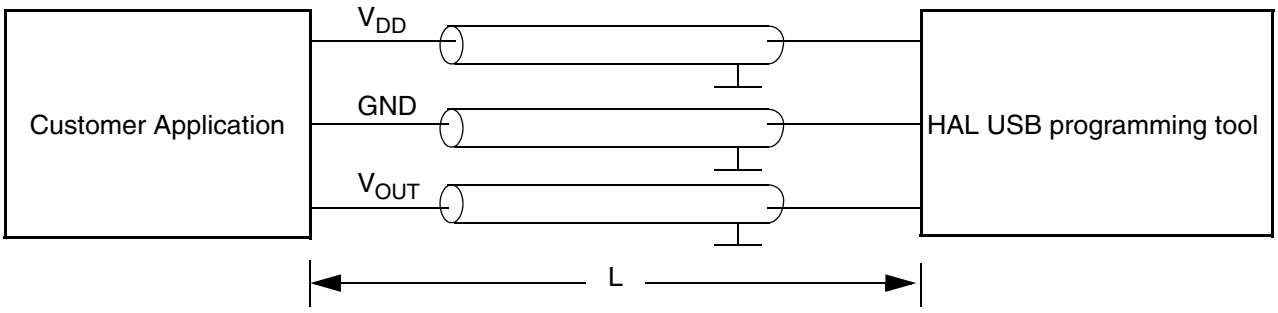


Fig. 4–5: Recommended wiring – schematic sketch

Table 4–3: 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	

4.3. Characteristics

Table 4–4: Board characteristics

Symbol	Parameter	Connector	Limit Values			Unit	Test Conditions
			Min.	Typ.	Max.		
I_{SUP_HAL}	Output Load Current	HAL 1/2	–	–	40	mA	Supply current per device
V_{OUT_HAL}	Output Voltage of Hall Device	HAL 1/2	0	–	5	V	Standard configuration (default)
$V_{SUP\_HAL\_NORM}$	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.

5. Board Functions

5.1. Serial Command Interpreter

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

The serial communication protocol uses a software handshake:

– The PC acts as a master, the HAL USB programming tool as slave,

– The HAL USB programming tool responds to each master **COMMAND** frame with a **RESPONSE** frame.

5.1.1. Serial Interface Configuration

When using a hyperterminal communication please set the following parameters.

Table 5–5: parameter settings of serial interface

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

5.1.2. 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).

5.1.3. 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 5–6)

The Rx-characters contain the received data depending on the command (see device-dependent command lists in section 7, 8, 9).

5.1.4. Error Codes

Table 5–6: Error codes

STATUS	Error
0	no error
1	acknowledge error
2	2’nd acknowledge error
3	invalid command for selected mode
4	reserved
5	reserved
6	reserved
7	reserved
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

6. Board Mode Settings

6.1. Board Operation Modes

In order to meet the different requirements of the various Hall devices, the board can run in different operation modes. When a particular device is used, the corresponding board mode and sensor type has to be selected first.

Table 6–7: Board modes

Mode	Description
A	HAL 1820 – Biphase via V_{SUP} - Pin HAL 24xy – Biphase via V_{SUP} - Pin HAL 36xy – Biphase via V_{SUP} - Pin HAL 37xy – Biphase via V_{SUP} - Pin HAL 38xy – Biphase via V_{SUP} - Pin
C	HAL 24xy – Biphase via OUT - Pin HAL 36xy – Biphase via OUT - Pin HAL 37xy – Biphase via OUT - Pin HAL 38xy – Biphase via OUT - Pin

6.2. Board Configuration Commands

The board configuration commands shall be used to

– select the board mode
– read firmware version

Table 6–8: Board configuration

Action	Command	Parameter	Remarks
get firmware version	?v	return <ST>:*[Version]*	firmware release version Example => ?v <= 0:v2.32
set board mode	sm**N**	**N** see Table 6–7 for details return value: <ST>:0000**N**	switch board to device specific mode Example => smA <= 0:0000A

Table 6–8: Board configuration

Action	Command	Parameter	Remarks
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
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.
[1)]Available with firmware versions greater than 2.32.. <ST> = Board Status character see from Section 5.1.4 on page 13 onwards for details			

7. HAL 1820

The HAL 1820 is a universal magnetic field sensor with a linear output, based on the Hall effect. A magnetic field perpendicular to the branded side of the sensor causes an output voltage directly 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.

7.1. 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. 7–1).

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

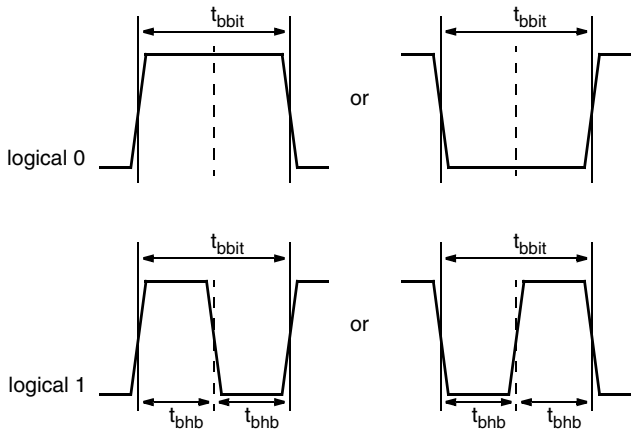


Fig. 7–1: Definition of logical 0 and 1 bit

7.2. Command Structures of Protocol

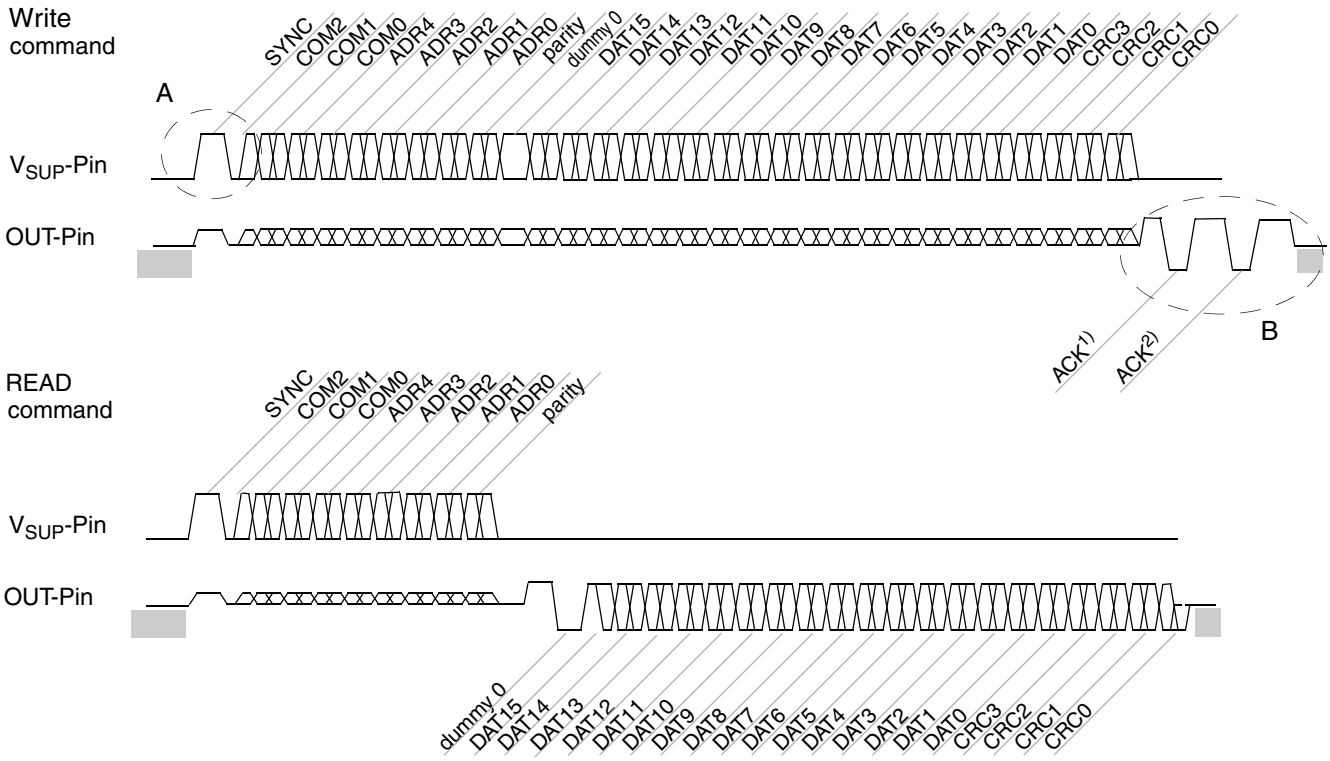
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

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

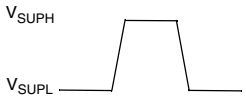
CRC: CRC bit

ACK: acknowledge
: Analog Output

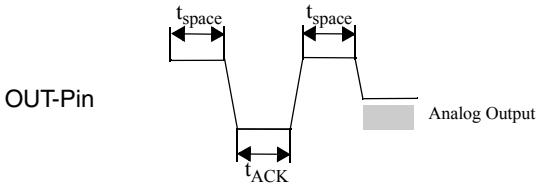
Communication via V_{SUP}-Pin (Biphase-In = V_{SUP}-Pin / Biphase-Out = OUT-Pin)



Detail A

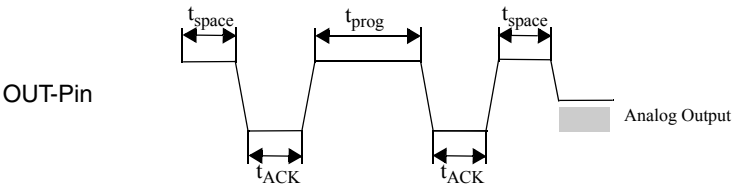


Detail B



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

Detail B



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

7.3. Telegram Parameters

Table 7–9: 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	

7.4. 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 7–10: 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

7.4.1. 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.

7.4.2. 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.

7.4.3. 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

7.4.4. 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

7.4.5. 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.

7.4.6. 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.

7.5. HAL 1820 – Board Commands

Note: For general board commands see Table 6–8 on page 14

Table 7–11: 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 5–6 on page 13 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 5–6 on page 13 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

7.6. 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.

8. HAL 24xy

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.

8.1. 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. 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.

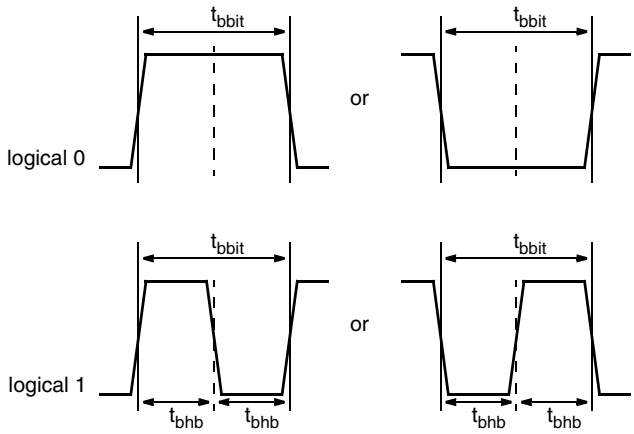


Fig. 8–1: Definition of logical 0 and 1 bit

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

command structure

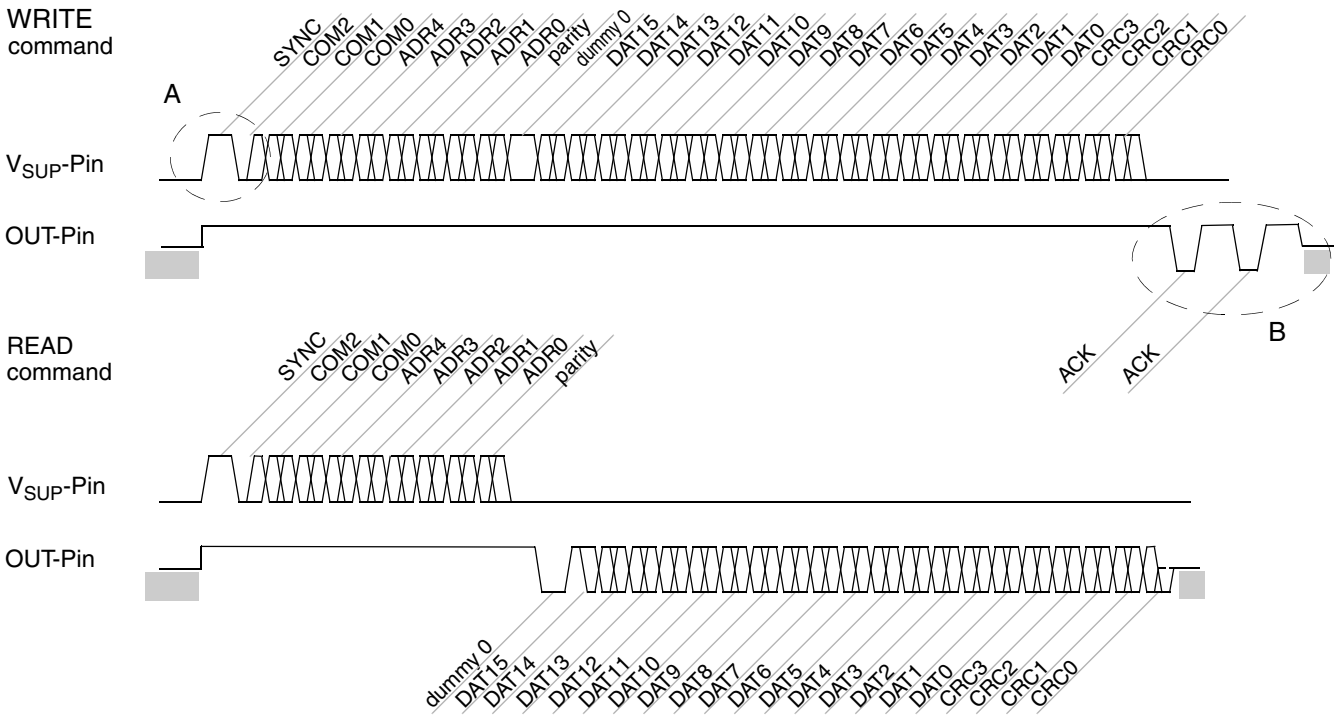
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

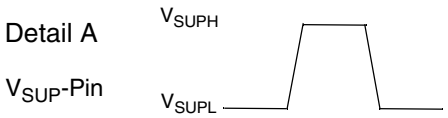
SYNC: start bit (always 0)
DAT: data bit

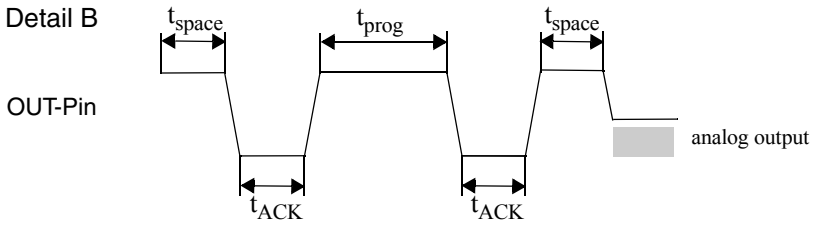
CRC: CRC bit

ACK: acknowledge
: Analog output

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







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

command structure

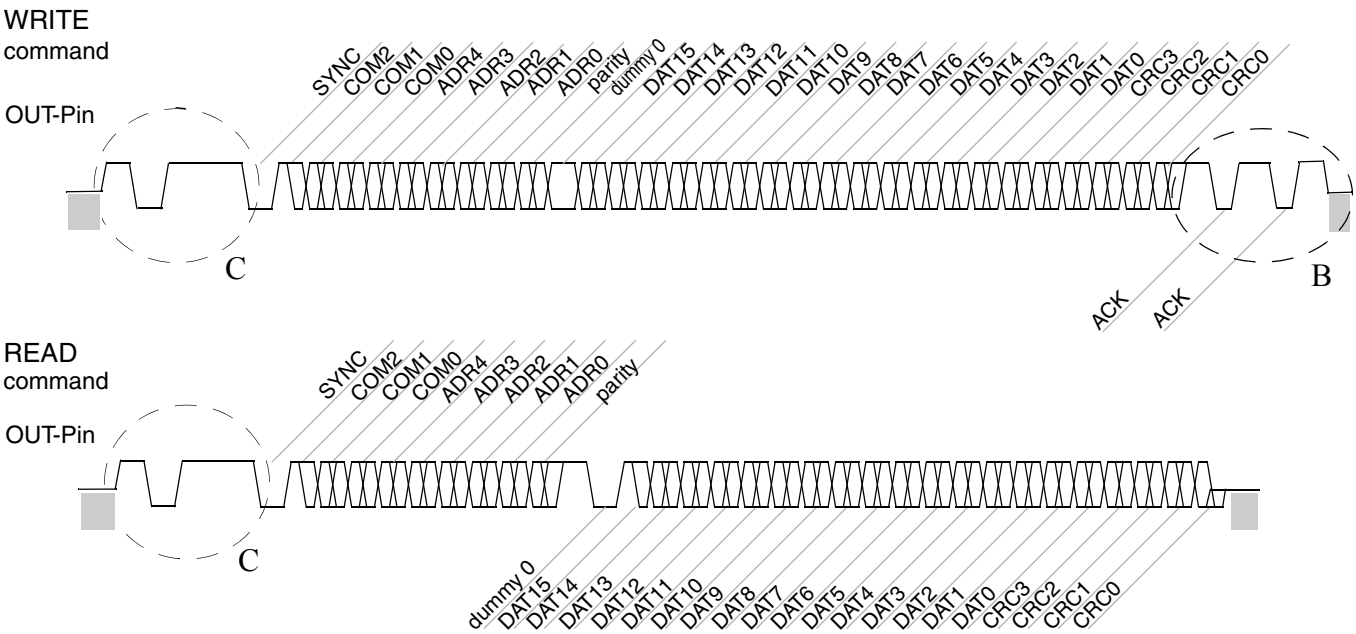
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

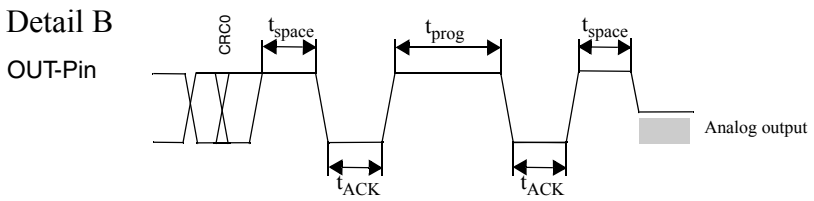
SYNC: start bit (always 0)
DAT: data bit

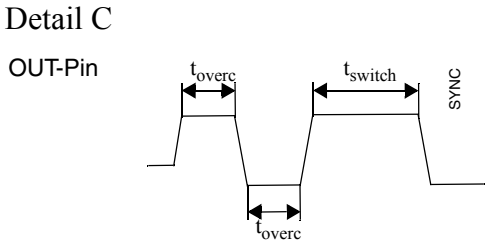
CRC: CRC bit

ACK: acknowledge
: Analog output

Communication via Out-Pin / Bidirectional on Out-Pin







8.4. Telegram Parameter

Table 8–12: Telegram Parameters HAL 24xy

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.

8.5. Available Sensor Commands

The sensor supports three 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 8–13: 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

8.5.1. 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.

8.5.2. 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]).

8.5.3. 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.

8.5.4. 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.

8.5.5. 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”.

8.5.6. 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.

8.6. HAL 24xy – Board Commands

Table 8–14: 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 5–6 on page 13 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 5–6 on page 13 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 5–6 on page 13 for details) <R> received data as 4-digit hex No. *Example:* read address b => xxr0b <= 0:3333E

8.7. 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.

Electrostatic 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.

9. HAL 3625, HAL 3675, HAL 37xy, HAL 385x and HAL 387x

The HAL 36x5, HAL 37xy, 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.

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.

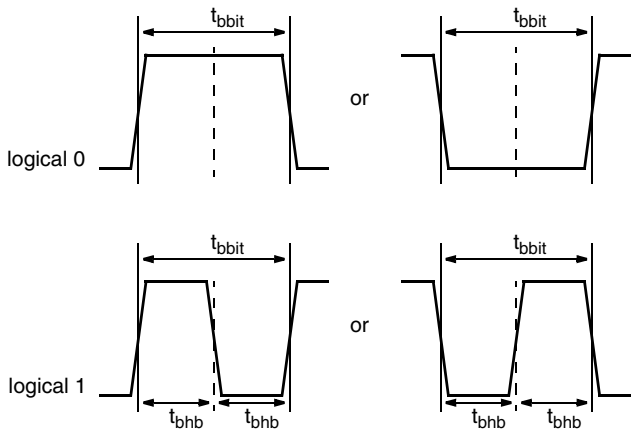


Fig. 9–1: Definition of logical 0 and 1 bit

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

command structure

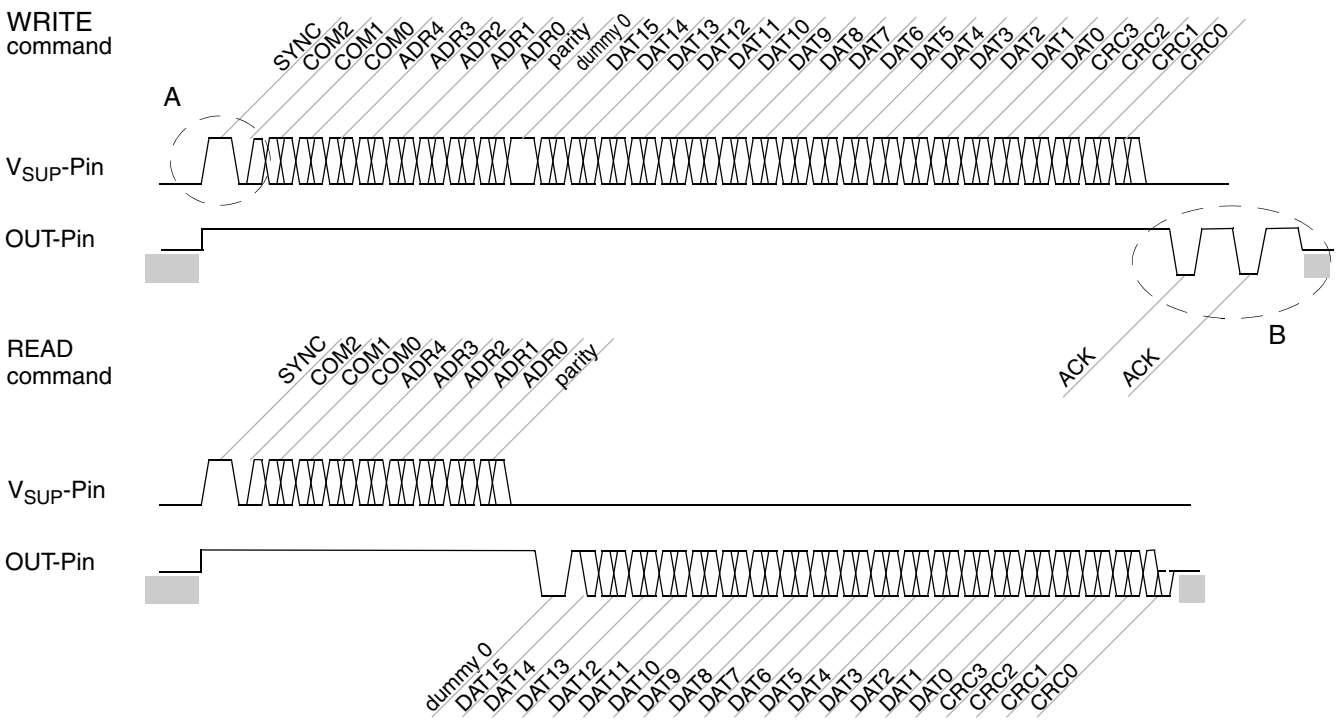
COM: command bit
ADR: address bit
parity: command and address check bit
dummy: dummy bit (always 0)

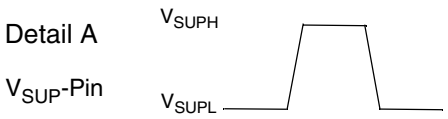
SYNC: start bit (always 0)
DAT: data bit

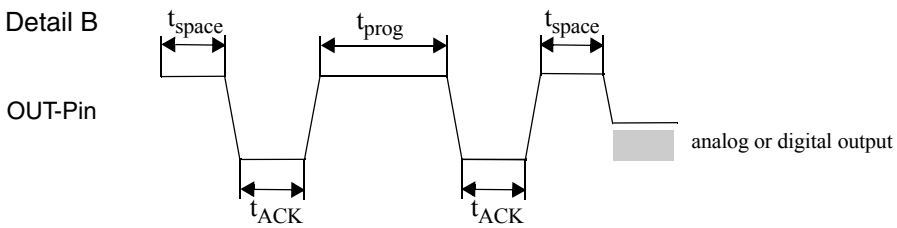
CRC: CRC bit

ACK: acknowledge
: analog or digital output

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







9.3. Telegram Parameters for communication via V_{sup}

Table 9–15: Telegram Parameters for communication via V_{sup}

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 acknowl-edge		–	1	–	t_{s_bbit}	
t_{ACK}	Acknowledge time		–	1	–	t_{s_bbit}	
t_{prog}	Programming time		–	4.5	–	ms	

9.4. 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)

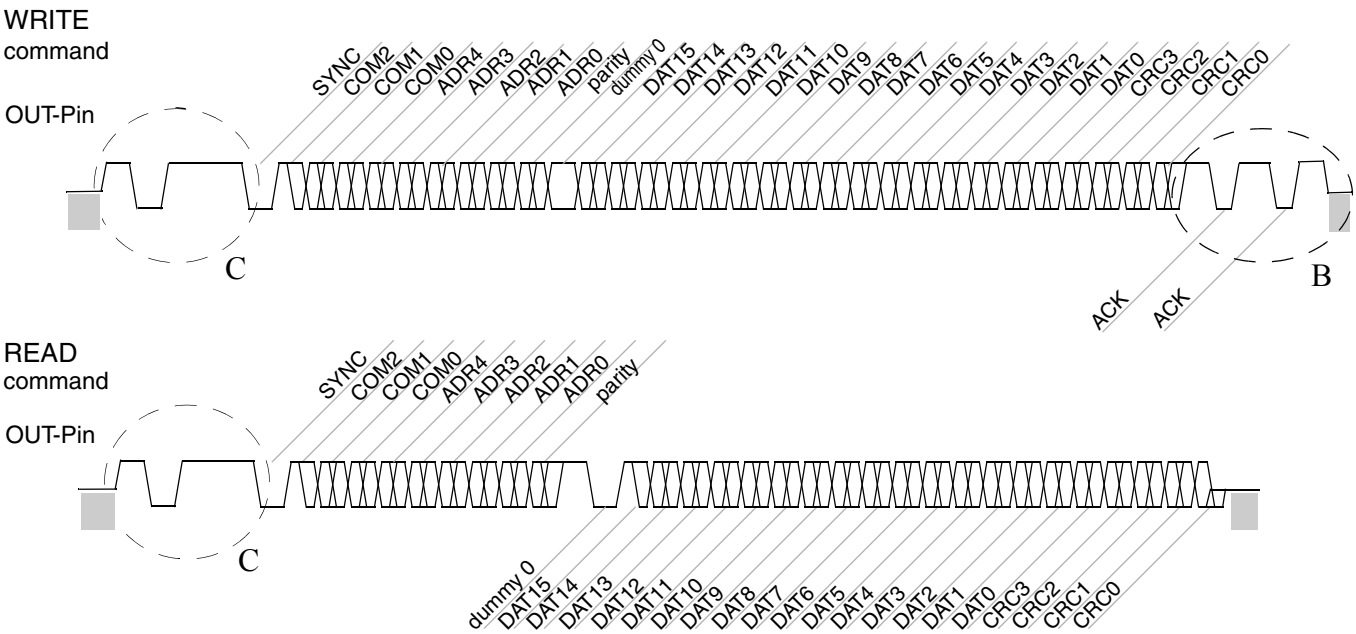
SYNC: start bit (always 0)
DAT: data bit

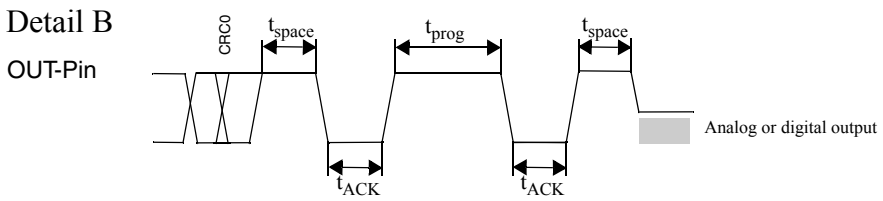
CRC: CRC bit

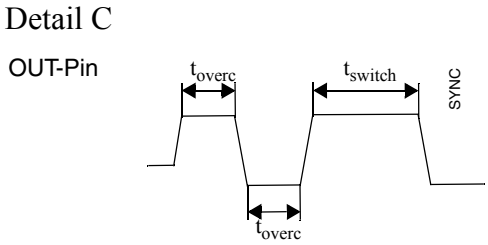
ACK: acknowledge

: Analog or digital output

Communication via Out-Pin / Bidirectional on Out-Pin







9.5. Telegram Parameters for communication via OUT-Pin

Table 9–16: Telegram Parameters for communication via OUT-Pin

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.

9.6. 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 9–17: 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

9.6.1. 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.

9.6.2. 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]).

9.6.3. 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.

9.6.4. 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.

9.6.5. 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”.

9.6.6. 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.

9.7. HAL 3625, HAL 3675, HAL 37xy, HAL 385x and HAL 385x – Board Commands

Table 9–18: HAL 3625, HAL 3675, HAL 37xy, 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 5–6 on page 13 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 5–6 on page 13 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 5–6 on page 13 for details) <R> received data as 4-digit hex No. *Example:* read address 8 => xxr08 <= 0:37B75

9.8. 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.

10. Application Note History

1. HAL1820, HAL24xy, HAL36xy, HAL37xy, HAL38xy HAL USB Programming Tool v1.0, Jan. 20, 2014; APN000093_001EN. First release of the application note.